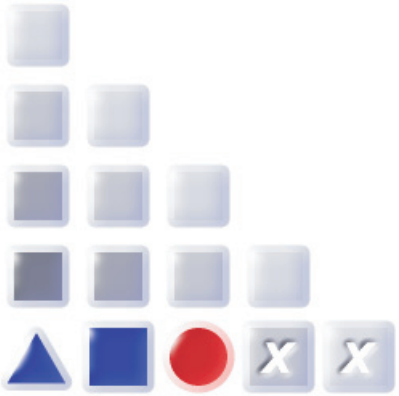
ADOxx®

Version 1.0



User Manual

www.boc-group.com

Books included in this documentation:

Introduction

User Manual

Administration Manual

Method Manual

Installation and Database Management

BOC Group – Addresses:

Austria:	BOC Asset Management GmbH A-1010 Vienna, Baeckerstr. 5
	BOC Information Technologies Consulting AG A-1010 Vienna, Wipplingerstr. 1
	BOC Information Systems GmbH A-1010 Vienna, Wipplingerstr. 1
	BOC Unternehmensberatung GmbH A-1010 Vienna, Rabensteig 2
Germany:	BOC Information Technologies Consulting GmbH Mosse-Palast am Leipziger Platz D-10117 Berlin, Vossstr. 22
Ibérica:	BOC Business Objectives Consulting Ibérica, S.A. E-28006 Madrid, Velàzquez, 71
Ireland:	BOC Information Technologies Consulting Ltd. 80 Haddington Road Dublin 4, Ireland
Greece:	BOC Information Technologies Consulting EPE GR-10680 Athens, Mavromichali 16
Poland:	BOC Information Technologies Consulting Sp. z o.o. PL-02-01 Warschau, Al. Jerozolimskie 109/26
Hotline:	hotline@boc-eu.com
Homepage:	http://www.boc-group.com

© User documentation for ADO*xx*® Version 1.0 2008
All rights reserved.

No part of this work covered by the copyright hereon may be reproduced or used in any form or by any means – graphic, electronic, or mechanical, including photocopying, recording, or information storage and retrieval systems – without the prior written approval of BOC Asset Management GmbH.

Table of Contents

Preface 1
1. Using the ADOxx Manuals 2
2. About this Manual 3
   2.1 Structure of the Manual 3
   2.2 Contents of the Manual 3
3. New Features in ADOxx Version 1.0 4
   3.1 New Features in the Modelling Toolkit 4
      3.1.1 General Innovations 4
      3.1.2 Modelling 5
      3.1.3 Import/Export 6
   3.2 New Functionality in the Administration Toolkit 6
      3.2.1 General news and improvements 6
      3.2.2 User Management 6
      3.2.3 Library Management 7
      3.2.4 Model Management 7
      3.2.5 LEO, AdoScript and Expressions 8
4. Note 9

I. ADOxx 10
1. General Notes 12
2. Trademarks 13
3. ADOxx Administration Toolkit 15
4. ADOxx Modelling Toolkit 17
5. ADOxx Product Palette 19
   5.1 ADOxx Business Edition 19
   5.2 ADOxx Professional Edition 20
   5.3 Additional ADOxx Components 20
      5.3.1 ADOxx Process Cost Analysis 20
      5.3.2 Personnel and Capacity Management 20
      5.3.3 Call Centre Management 21
      5.3.4 case/4/0 Interface 21
      5.3.5 objectiF Interface 21
   5.4 ADOxx Modelling Methods 22
   5.5 ADOxx Interfaces 22
   5.6 Support of Standards and Management Methods 24

II. Fundamentals of modelling 25
1. Introduction to modelling with ADOxx 26
   1.1 Identification of Business Processes 26
   1.2 Process Flow 26
      1.2.1 Preparation for modelling 27
      1.2.2 Modelling objects 27
      1.2.3 Inserting connectors 28
      1.2.4 Naming objects and connectors 28
      1.2.5 Further object modelling 28
      1.2.6 Control 28
   1.3 Organisational structure 28
   1.4 Entering model information 29
2. Example models 30

III. General information about the Modelling Toolkit 31
1. Terms and Context 32
2. ADOxx User Interface 36
   2.1 Window Bar 36

2.2 Menu Bar ... 37
2.3 Component Bar ... 37
2.4 Quick-Access Bar ... 37
2.5 Workspace ... 38
2.6 Explorer ... 38
2.7 Navigator ... 38
2.8 Inspector ... 39
2.9 Search Results ... 39
3. Control Elements ... 41
3.1 Selection Control Elements ... 41
3.1.1 Model Types ... 42
3.1.1.1 Model Type Filter ... 42
3.1.2 Refresh ... 43
3.1.3 Item Search ... 43
3.1.3.1 Display Search Results ... 44
3.1.4 Save as ... 45
3.1.5 Shrink/Expand ... 45
3.1.6 Select All Items ... 46
3.1.7 Deselect ... 46
3.1.8 Selected Items ... 46
3.1.9 Shrink/Expand Version Threads ... 47
3.2 ADOxx Notebook Control Elements ... 48
3.2.1 "Save as" Icon ... 49
3.2.2 "Print" Icon ... 49
3.2.3 "Info" Icon ... 49
3.2.4 "LargeText Field" Icon ... 49
3.2.5 "Dialog" Icon ... 49
3.2.6 "Expression Dialog" Icon ... 50
3.2.7 "Add" Icon ... 50
3.2.8 "Delete" Icon ... 50
3.2.9 "Execute/Follow" Icon ... 50
3.2.10 "File Dialog" Icon ... 50
3.2.11 "Notebook" Icon ... 50
3.2.12 Input Window ... 51
3.2.12.1 Input Dialogue for Text ... 51
3.2.12.2 Text Input Field ... 52
3.2.12.3 Input Dialogue for Expressions ... 52
3.2.13 Accelerators ... 53
4. ADOxx Explorer ... 54
4.1 Models in the Database ... 56
4.2 Opened Model Windows ... 57
4.3 Refresh ... 57
4.4 Options for the Opening of Models ... 57
4.5 Set Visible Model Types ... 58
4.6 Show / Hide Branches (Levels) ... 58
4.7 Create New Models / New Model Groups ... 58
4.8 Open / Activate Models ... 59
4.9 Close Models ... 59
4.10 Save Models ... 59
4.11 Rename Models / Model Groups ... 59
4.12 Print Models ... 60
4.13 Cut / Copy / Paste / Delete Models and Model Groups ... 60
4.14 Copy / Delete Model References ... 60
4.15 Manage model groups ... 61
4.16 Item Search ... 61
4.17 Save Tree Structure ... 61
5. ADOxx Navigator ... 62
6. ADOxx Inspector ... 64
7. ADOxx Serach Result Window ... 65
8. ADOxx Browser ... 66
8.1 Classification ... 66

8.2 Structure ... 68
8.2.1 Editable ADOxx Browser ... 69
8.2.2 Hierarchical ADOxx Browser ... 70
8.2.3 Representation of Attribute Values ... 71
8.2.3.1 Representation of Programcall Attribute Values ... 72
8.2.3.2 Edit References ... 73
8.3 Enter Column Width ... 74
8.4 Adjust Column Width ... 75
8.5 Enter Row Height ... 75
8.6 Adjust Row Height ... 76
8.7 Expand / Shrink All ... 76
8.8 Select Attributes/Columns ... 77
8.9 Sort ... 78
8.9.1 Sort by Attribute Columns ... 78
8.10 Align Attribute Values ... 79
8.11 Show/Edit (Attribute) Values ... 79
8.12 Search ... 80
8.13 Save ... 81
8.13.1 Properties ... 82
8.13.2 Formats ... 82
8.13.2.1 Save as spreadsheet (*.txt) ... 82
8.13.2.2 Save as comma-separated value (*.csv) ... 83
8.13.2.3 Save as text (*.txt) ... 83
8.13.2.4 Save as RTF file (*.rtf) ... 83
8.13.2.5 Save as HTML file (*.htm) ... 84
8.13.2.6 Save as comparable representation(*.acr) ... 84
8.13.2.7 Save as AmiPro text (*.txt) ... 84
8.14 Copy to Clipboard ... 84
8.15 Print ... 84
8.16 Diagram ... 85
8.16.1 Bar Diagram ... 87
8.16.1.1 Settings for the Bar Diagram Display ... 88
8.16.2 Pie Chart ... 89
8.16.3 Generate Graphics ... 90
9. Swimlanes ... 91
10. Versioning ... 92
10.1 Basics ... 92
10.1.1 Model-related Versioning ... 92
10.1.2 Time-related Versioning ... 92
10.2 Effects ... 93
11. Inter-model References ... 94
11.1 Reference Settings ... 97
11.1.1 Edit Reference Depth ... 98
11.2 Effects of the Versioning on Inter-model References ... 98
12. Attribute Profiles ... 100
13. Record Attributes ... 104
IV. Components of the Modelling Toolkit ... 107
1. Information acquisition ... 108
1.1 HOMER (Overview) ... 108
1.2 Start HOMER ... 109
1.3 Select HOMER Scenario ... 109
1.4 Transform ADL Data to Excel Sheets ... 110
1.5 Transform Excel Sheets to ADL Files ... 115
1.6 Creating Acquisition Table ... 118
1.7 Consolidating Acquisition Tables ... 121
2. Modelling ... 126
2.1 Introduction ... 126
2.2 The Model Editor ... 127
2.2.1 The Working Area of the Model Editor ... 127
2.2.2 Modelling Bar ... 128
2.2.2.1 Class Panel ... 129

2.2.2.2 Relation Panel ... 129
2.2.3 Model Window ... 130
2.2.4 Modelling modes of the Model Editor ... 131
2.2.4.1 Editing Mode ... 131
2.2.4.2 Drawing Mode ... 131
2.2.5 Drawing Objects ... 132
2.2.6 Drawing Connectors ... 132
2.2.6.1 Bendpoints ... 133
2.2.6.2 Adding New Relations via the Context Menu ... 134
2.2.6.3 Adding New Relations ... 134
2.2.6.4 Relation "is inside" ... 135
2.2.7 Editing Objects and Connectors ... 136
2.2.7.1 Undo ... 137
2.2.7.2 Redo ... 137
2.2.7.3 Select Objects and Connectors ... 138
2.2.7.4 Focus ... 139
2.2.7.5 Move Objects ... 139
2.2.7.6 Change Object Size ... 141
2.2.7.7 Define Object Size and Position ... 142
2.2.7.8 Change Connector Course ... 144
2.2.7.9 Re-allocate Connectors ... 145
2.2.7.10 Rename Connector Marks ... 146
2.2.7.11 Manage Connectors in ADOxx Notebooks ... 146
2.2.7.12 Edit Attributes in the ADOxx Notebooks ... 146
2.2.7.13 Change Visualised Attribute Values ... 148
2.2.7.14 Move Connector Centre ... 148
2.2.7.15 Copy ... 149
2.2.7.16 Cut ... 150
2.2.7.17 Paste ... 151
2.2.7.18 Delete ... 152
2.2.7.19 Search Model ... 152
2.2.7.20 Generate Object Names ... 153
2.2.7.21 Change Values of Object/Connector Attribute ... 154
2.2.7.22 Number Objects ... 155
2.2.7.23 Convert Object ... 155
2.2.7.24 Generate Graphics ... 156
2.2.7.25 Align Objects ... 160
2.2.7.26 Arrangement Function ... 160
2.2.7.27 Arrange Manually - Arrangement Assistant ... 162
2.2.8 Context Menu ... 173
2.2.8.1 Context Menu of the Drawing Area ... 174
2.2.8.2 Context Menu of Objects ... 174
2.2.8.3 Context Menu of Connectors ... 176
2.2.9 Drawing Area ... 176
2.2.9.1 Resize the Drawing Area ... 176
2.2.9.2 Adding/Removing Empty Space ... 178
2.2.10 Tabular Model View ... 181
2.2.10.1 Class Selection ... 182
2.2.10.2 Display Attribute Columns ... 183
2.2.10.3 Edit Attributes ... 184
2.2.11 View Options ... 186
2.2.11.1 View Mode ... 186
2.2.11.2 Mouse Access ... 188
2.2.11.3 Zoom ... 189
2.2.11.4 Snap Grid ... 190
2.2.11.5 Page Layout ... 192
2.2.11.6 Connector Representation ... 197
2.2.11.7 Font Size ... 199
2.2.11.8 Show/Hide Submodels ... 200
2.2.11.9 Redraw ... 201
2.2.12 Input Support Dialogues in ADOxx Notebooks ... 201

2.2.12.1 Edit Record Attributes ... 202
2.2.12.2 Define Colours ... 203
2.2.12.3 Select Enumeration Value ... 204
2.2.12.4 Select Values from an Enumeration List ... 204
2.2.12.5 Enter Program Call ... 205
2.2.12.6 Edit Date Value ... 205
2.2.12.7 Edit Date and Time Value ... 206
2.2.12.8 Editing Times ... 206
2.2.12.9 Add References ... 207
2.2.12.10 Add Attribute Profile Reference ... 208
2.2.12.11 Random Generator ... 211
2.2.12.12 Transition Conditions in the Subsequent Relation ... 215
2.2.12.13 Assigning and Defining Sub Processes ... 221
2.2.12.14 Assigning Performers ... 222
2.2.12.15 Allocating Resources ... 227
2.2.12.16 Performer Calendar ... 228
2.2.12.17 Process Calendar ... 235
2.2.13 Modelling Guidelines for Simulation ... 240
2.2.13.1 Completeness ... 240
2.2.13.2 Modelling Parallel Paths ... 241
2.2.14 Global Attribute Change ... 243
2.2.14.1 Model Selection ... 244
2.2.14.2 Object Selection ... 244
2.2.14.3 Attribute Selection ... 249
2.2.14.4 Changing Attribute Values ... 250
2.3 Model Navigation ... 255
2.3.1 Navigating through open models ... 255
2.3.2 Activate model window ... 255
2.3.3 Activate Referenced Model ... 255
2.4 Model Management ... 256
2.4.1 Create New Model ... 256
2.4.2 Open Existing Model ... 257
2.4.3 Close Model ... 259
2.4.4 Save Model ... 259
2.4.4.1 Auto-save ... 260
2.4.5 Save Model Under a Different Name ... 260
2.4.6 Save Model as New Version ... 262
2.4.7 Export Table ... 262
2.4.8 Rename Model ... 262
2.4.8.1 Version Number ... 263
2.4.8.2 Model Name ... 264
2.4.9 Model Attributes ... 264
2.4.10 Check Cardinalities ... 264
2.4.11 Inter-model References ... 265
2.4.11.1 Outgoing References ... 265
2.4.11.2 Incoming References ... 267
2.4.12 Managing Model Groups ... 269
2.4.12.1 Create Model Group ... 270
2.4.12.2 Rename Model Group ... 270
2.4.12.3 Move Model Group ... 271
2.4.12.4 Delete model group ... 271
2.4.12.5 Assign Models ... 271
2.4.12.6 Move Model Reference ... 272
2.4.12.7 Copy Model Reference ... 273
2.4.12.8 Delete Model Reference ... 273
2.4.13 Model Comparison ... 273
2.4.13.1 Model Selection ... 273
2.4.13.2 Results ... 274
2.4.14 Search Models ... 275
2.4.14.1 Model Search by Model Attributes ... 276
2.4.14.2 Process Hierarchy (tabular) ... 279

© BOC Group 2008

2.4.14.3 Search Result ... 281
2.4.14.4 Model Search for All Child Processes - Search Result ... 281
2.4.14.5 Process Hierarchy (graphical) ... 281
2.4.15 Deleting models ... 283
2.4.15.1 Show Referenced Models ... 283
2.4.16 Application Models ... 284
2.4.16.1 Open Application Model ... 285
2.4.16.2 Add Application Model ... 285
2.4.16.3 Rename Application Model ... 287
2.4.16.4 Edit Application Model ... 287
2.4.16.5 Delete Application Model ... 288
2.4.17 Print Models ... 288
2.4.17.1 Print Models - Model selection ... 289
2.4.17.2 Print several models with identical settings ... 290
2.4.17.3 Select Pages ... 290
2.5 Editing Attribute Profiles ... 291
2.5.1 Add Attribute Profile Folder ... 292
2.5.2 Rename Attribute Profile Folder ... 293
2.5.3 Copy Attribute Profile Folder ... 293
2.5.4 Move Attribute Profile Folder ... 294
2.5.5 Delete Attribute Profile Folder ... 294
2.5.6 Add Attribute profile ... 294
2.5.7 Save Attribute Profile as New Version ... 295
2.5.8 Edit Attribute Profile Values ... 295
2.5.9 Edit several attribute profile values simultaneously ... 296
2.5.10 Show Attribute Profile Values ... 296
2.5.11 Show Several Attribute Profile Values simultaneously ... 297
2.5.12 Copy Attribute Profile ... 297
2.5.13 Move Attribute Profile ... 298
2.5.14 Delete Attribute Profile ... 298
2.5.15 Show Used Attribute Profiles ... 298
2.5.16 Carry out Queries on Attribute Profiles ... 299
2.5.16.1 Standardised Queries on Attribute Profiles ... 299
2.5.16.2 User-defined Queries on Attribute Profiles ... 301
2.5.16.3 Query Results ... 302
2.6 Calculating Times and Costs ... 303
2.6.1 Settings ... 304
2.6.2 Results ... 305
3. Analysis ... 306
3.1 Queries/Reports ... 306
3.1.1 Model Selection ... 307
3.1.2 Carrying out Standardised Queries ... 308
3.1.2.1 Select Attributes for Result Display ... 311
3.1.2.2 Model Information ... 311
3.1.3 Carrying out User Defined Queries ... 312
3.1.4 Query Results ... 313
3.1.4.1 Edit Attribute Values ... 315
3.1.4.2 Sort Query Result ... 315
3.1.4.3 Select Attributes ... 316
3.1.4.4 Display Record Attributes ... 317
3.2 Predefined Queries and Plans ... 318
3.2.1 Model Selection ... 319
3.2.2 Executing Predefined Queries ... 319
3.2.3 Results ... 321
3.2.4 Results of Predefined Plans ... 321
3.2.4.1 Change column titles of predefined plans ... 322
3.3 Relation Tables ... 322
3.3.1 Model Selection ... 322
3.3.1.1 Settings ... 324
3.3.2 Display ... 325
3.4 Analytical Evaluation ... 325

3.4.1 Processing ... 326
3.4.2 Results ... 327
3.4.3 Display Results ... 327
3.4.3.1 Display Results ... 329
3.4.4 Background Information on the Analytical Evaluation ... 329
3.4.4.1 Input for the Analytical Evaluation ... 329
3.4.4.2 Functioning of the Analytical Evaluation ... 330
4. Simulation ... 332
4.1 Introduction ... 332
4.1.1 Overview of the simulation algorithms ... 333
4.1.2 Evaluation Mechanisms ... 334
4.1.3 Time concepts within simulation ... 334
4.1.3.1 The difference between "Company time" and "Real time" ... 335
4.1.3.2 Activity Times ... 335
4.1.3.3 Process times as simulation results ... 339
4.1.4 Cooperative vs. Non-cooperative Activities ... 340
4.1.5 Global End ... 340
4.1.6 Manual random generator ... 342
4.1.7 Period-related versus Process-related View ... 342
4.2 Path Analysis ... 343
4.2.1 Executing a Path Analysis ... 343
4.2.1.1 Numbers ... 345
4.2.1.2 Options ... 346
4.2.1.3 Passive components ... 346
4.2.2 Simulation Run ... 346
4.2.3 Results ... 347
4.2.3.1 Path Results ... 347
4.2.3.2 Business Process Results ... 348
4.2.4 Displaying results ... 348
4.2.4.1 Display business process results ... 349
4.2.4.2 Path results ... 350
4.2.4.3 Transferring results ... 350
4.2.5 Background ... 353
4.2.5.1 Input ... 353
4.2.5.2 Processing ... 353
4.2.6 Application Scenario (Example) ... 354
4.3 Capacity analysis ... 354
4.3.1 Carrying out a capacity analysis ... 355
4.3.1.1 Numbers ... 356
4.3.1.2 Options ... 357
4.3.1.3 Passive components ... 357
4.3.2 Simulation run ... 357
4.3.3 Results ... 357
4.3.4 Displaying Results ... 358
4.3.4.1 Displaying Results in Tabular Form ... 359
4.3.4.2 Personnel Capacity Planning ... 361
4.3.4.3 Representing results graphically (Diagram) ... 361
4.3.4.4 Transferring the Results of the Capacity Analysis ... 362
4.3.5 Background of capacity analysis ... 362
4.3.5.1 Input ... 362
4.3.5.2 Processing ... 363
4.3.6 Application Scenario (Example) ... 366
4.4 Workload analysis (steady-state and fixed time period) ... 367
4.4.1 Carrying out a workload analysis ... 367
4.4.1.1 Numbers ... 369
4.4.1.2 Options ... 370
4.4.1.3 Passive components ... 370
4.4.2 Simulation run ... 371
4.4.3 Task stacks for performers ... 372
4.4.4 Results of the workload analysis ... 373
4.4.5 Representing results ... 373

© BOC Group 2008

4.4.5.1 The display of results in a tabular form ... 375
4.4.5.2 Transferring the results of the workload analysis ... 375
4.4.6 Background of the workload analysis ... 376
4.4.6.1 Input ... 376
4.4.6.2 Processing ... 376
4.4.7 Application scenario (example) ... 385
4.5 Offline animation ... 386
4.5.1 Input ... 386
4.5.2 Carry out an offline animation ... 386
4.5.3 Processing ... 387
4.6 Simulation cache ... 388
4.6.1 Changes and their effects on the simulation cache ... 389
4.6.2 Manual deletion of the simulation cache contents ... 389
4.7 Delete simulation results ... 390
4.8 Agents ... 391
4.8.1 Creating and customising agents ... 391
4.8.2 Agents overview ... 392
4.8.2.1 Define agent types ... 394
4.8.2.2 Adjusting reference objects ... 395
4.8.3 Configuring agents ... 396
4.8.3.1 Basic configuration ... 397
4.8.4 Result functions for agents ... 400
4.8.4.1 Result function "animation" ... 401
4.8.4.2 Result function "Number of changes in responsibility" ... 401
4.8.4.3 Result function "Frequency" ... 402
4.8.4.4 Result function "Execution costs" ... 403
4.8.4.5 Result function "Execution costs (conditional)" ... 404
4.8.4.6 Result function "Workload" ... 405
4.8.4.7 Result function "Capacity" ... 405
4.8.4.8 Result function "Cycle time" ... 406
4.8.4.9 Result function "Special formula" ... 407
4.8.4.10 Result function "Sum over attribute" ... 409
4.8.5 Configure result functions ... 410
4.8.5.1 Basic configuration ... 410
4.8.5.2 "Animation" ... 412
4.8.5.3 "Number of changes in responsibility" ... 413
4.8.5.4 "Frequency" ... 413
4.8.5.5 "Costs of execution" ... 414
4.8.5.6 "Costs of execution (conditional)" ... 415
4.8.5.7 "Workload" ... 416
4.8.5.8 "Capacity" ... 417
4.8.5.9 "Cycle time" ... 418
4.8.5.10 "Special formula" ... 419
4.8.5.11 "Sum over attribute" ... 421
4.8.6 Employing agents in simulations ... 422
4.8.6.1 The animation of results by agents ... 423
4.8.6.2 Agents - Result selection ... 424
4.8.6.3 Represent agents results in tabular form ... 430
4.8.6.4 Represent agents results graphically ... 430
4.8.6.5 Represent result development as function graph ... 431
4.8.6.6 File format of the result development ... 432
5. Evaluation ... 433
5.1 Comparison of results ... 433
5.1.1 result selection ... 434
5.1.1.1 Add result file ... 435
5.1.1.2 Information on the results ... 436
5.1.2 Display results graphically ... 436
5.1.3 Tabular representation ... 437
5.2 Evaluation queries ... 438
5.2.1 Model selection ... 438
5.2.2 Run evaluation queries ... 439

5.2.2.1 Model information ... 440
5.2.3 Results ... 441
5.3 FlowMark Audit Trail Evaluation ... 441
5.3.1 Tabular representation - process-related ... 442
5.3.2 Tabular representation - performer-related ... 443
5.4 Process cost analysis ... 444
6. Import/Export ... 445
6.1 ADL Import ... 445
6.1.1 Model Import ... 446
6.1.2 Selection ... 450
6.1.2.1 Assign model type ... 453
6.1.3 Result ... 454
6.1.4 Options and examples ... 454
6.1.4.1 General options for ADL import ... 454
6.1.4.2 Rename models from ADL file ... 455
6.1.4.3 To paste into existing models ... 455
6.1.4.4 Overwrite existing models ... 460
6.1.4.5 Ignore Models ... 461
6.1.4.6 To increase the version number ... 461
6.1.4.7 Rename attribute profiles from ADL file ... 462
6.1.4.8 Paste into existing attribute profiles ... 462
6.1.4.9 Overwrite existing attribute profiles ... 463
6.1.4.10 Ignore attribute profiles from ADL file ... 463
6.1.4.11 Assign version number ... 464
6.1.5 Import application models ... 464
6.1.6 Model overview ... 467
6.1.7 Result ... 467
6.2 ADL Export ... 468
6.2.1 Model Export ... 468
6.2.2 Export Application Models ... 472
6.3 XML Import ... 473
6.3.1 Model Import ... 474
6.3.2 Selection ... 476
6.3.3 Result ... 480
6.4 XML Export ... 481
6.4.1 Model Export ... 481
6.5 Documentation ... 486
6.5.1 Generate documentation ... 486
6.5.2 Options ... 488
6.5.3 Attribute and class filter ... 493
6.5.3.1 Load mode ... 495
7. Additional components ... 497
7.1 Process Cost Analysis ... 497
7.1.1 Introduction ... 497
7.1.2 Process cost analysis with ADOxx-3c ... 498
7.1.3 Actual and plan analysis ... 499
7.1.4 Input ... 499
7.1.5 Method of working ... 500
7.1.5.1 Simulation ... 500
7.1.5.2 Establishing costs of processes ... 500
7.1.5.3 Acquiring process costs rate ... 501
7.1.5.4 ON processes and idle capacities ... 503
7.1.5.5 Assessment of the cost centers' capacity ... 504
7.1.6 Carrying out the process cost analysis ... 504
7.1.6.1 Starting the process cost analysis ... 504
7.1.6.2 Simulation ... 506
7.1.6.3 Presentation of results ... 508
7.1.6.4 Ending of the Cost-Cutting Component ... 513
7.2 Personnel and capacity management ... 513
7.2.1 Starting Analysis ... 515
7.2.2 Result display options ... 516

© BOC Group 2008

7.2.3 Presentation of results ... 517
7.3 case/4/0 interface ... 517
7.3.1 Interface concept ... 517
7.3.2 Tracking references ... 518
7.3.3 Creating new references ... 519
7.3.3.1 case/4/0 login ... 519
7.3.4 Changing references ... 519
7.3.5 Deleting references ... 519
8. Error Messages ... 520
V. Appendix ... 721
1. Starting ADOxx ... 722
1.1 Start ADOxx ... 722
1.2 Start ADOxx (Single-Sign-on) ... 724
1.3 Start ADOxx WebService ... 726
2. Closing ADOxx ... 727
3. Status Information ... 728
4. Current Configuration ... 729
5. Service Information ... 730
6. Regular Expressions ... 731
7. ADOxx Attribute Types ... 733
7.1 Attribute Profile Reference (ATTRPROFREF) ... 733
7.2 Enumeration (ENUMERATION) ... 733
7.3 Enumeration List (ENUMERATIONLIST) ... 734
7.4 Expression (EXPRESSION) ... 734
7.5 Date (DATE) ... 734
7.6 Date and Time (DATETIME) ... 734
7.7 Integer (INTEGER) ... 735
7.8 Floating Point (DOUBLE) ... 735
7.9 Calendar ... 735
7.10 Long String (LONGSTRING) ... 735
7.11 Program Call (PROGRAMCALL) ... 736
7.12 Inter-model Reference (INTERREF) ... 736
7.13 Record (RECORD) ... 736
7.14 String (STRING) ... 736
7.15 Time (TIME) ... 736
8. Language Codes ... 738
9. Change Password ... 744
10. Messages ... 745
10.1 Send Messages ... 745
10.1.1 Select Receiver ... 746
10.2 Reply Message ... 747
10.3 Forward Message ... 748
10.4 Receive New Message ... 749
10.5 Display Message ... 749
10.6 Received Message ... 749
10.7 Read Message ... 751
10.8 Receive Instant Message ... 752
10.9 Settings ... 752
11. File Formats in ADOxx ... 754
11.1 ADONIS Binary Language (ABL) ... 754
11.2 ADONIS Comparable Representation (ACR) ... 754
11.3 ADONIS Definition Language (ADL) ... 755
11.4 ADOxx Protocol Format (APF) ... 755
11.5 Windows-Bitmap (BMP) ... 755
11.6 Comma Separated Value (CSV) ... 755
11.7 Windows Enhanced Metafiles (EMF) ... 755
11.8 FlowMark Definition Language (FDL) ... 755
11.9 HyperText Markup Language (HTML) ... 756
11.10 JPEG File Interchange-Format (JPG) ... 756
11.11 ZSoft Paintbrush Graphic (PCX) ... 756
11.12 Portable Network Graphics (PNG) ... 756

11.13 Rich Text Format (RTF) ... 756
11.14 Scalable Vector Graphics (SVG) ... 756
11.15 ASCII-Text (TXT) ... 757
11.16 User Definition Language (UDL) ... 757
11.17 Extensible Markup Language (XML) ... 757
12. Expressions ... 758
13. HOMER Scenarios ... 763
13.1 HOMER Scenarios Administration ... 763
13.1.1 Creating HOMER Scenario ... 764
13.1.2 Importing HOMER Scenario ... 764
13.1.2.1 Importing HOMER Scenario ... 765
13.1.2.2 Converting HOMER INI File ... 766
13.1.3 Exporting HOMER Scenario ... 767
13.1.4 Editing HOMER Scenario ... 768
13.1.5 Deleting HOMER Scenario ... 768
13.2 Defining the Structure of Acquisition Tables ... 768
13.2.1 General Model Parameters ... 769
13.2.2 Object Settings ... 771
13.2.2.1 Adding Classes ... 773
13.2.2.2 Editing Class Configuration ... 774
13.2.2.3 Deleting Classes ... 774
13.2.2.4 Adding Attribute Profile Classes ... 774
13.2.2.5 Editing Attribute Profile Classes Configuration ... 775
13.2.2.6 Deleting Attribute Profile Classes ... 775
13.2.2.7 Adding Records ... 776
13.2.2.8 Editing Records ... 776
13.2.2.9 Deleting Records ... 777
13.2.2.10 Adding Attributes ... 777
13.2.2.11 Attribute Types in HOMER ... 780
13.2.2.12 Attribute Formats in HOMER ... 781
13.2.2.13 Modifying Attributes ... 783
13.2.2.14 Deleting Attributes ... 785
13.2.2.15 Adding Model Attribute Classes ... 785
13.2.2.16 Editing Model Attribute Class Configuration ... 786
13.2.2.17 Deleting Model Attribute Classes ... 787
13.2.2.18 Adding Model Attributes ... 787
13.2.2.19 Changing Model Attributes ... 788
13.2.2.20 Deleting Model Attributes ... 788
14. ADOxx Query Language (AQL) ... 790
14.1 Syntax and Semantics of AQL ... 790
14.2 Rules for formulating AQL expressions ... 792
14.3 AQL Examples ... 794
15. colour names ... 801
16. ADOxx standard application library ... 802
17. Ergonomics Compliance Statement for ADOxx ... 803
17.1 Human Criteria ... 803
17.1.1 User Orientation ... 804
17.1.2 Versatility ... 804
17.1.3 Integrity and Signification ... 804
17.1.4 Scope of Action ... 804
17.1.5 Report Messages ... 806
17.2 Dialogue Design ... 806
17.2.1 Task Appropriateness ... 807
17.2.2 Self-description Ability ... 807
17.2.3 Expectation Conformity ... 809
17.2.4 Controllability ... 809
17.2.5 Error Tolerance ... 811
17.2.6 Individuality ... 812
17.2.7 Learn Beneficiality ... 812
17.3 Conformity With the Windows User Interface Style Guide ... 813
17.3.1 Literature ... 814

18. Glossary 815
19. Index 837

Preface

Welcome to

is the extensible, multi-lingual, multi-os platform for product development from the ***Management Office*** of the BOC Group. The ADOxx platform provides fundamental core functionality and a constantly growing library of product building blocks which can be assembled into a final product.

ADOxx is a client/server multi-user system, which has an object-oriented structure. Additionally, **ADOxx** has a remarkable adaptation possibility, so it can be configured according to your needs and developed according to your requirements ("ADOxx-customising").

We hope that our tool meets your requirements and that you have a lot of fun working with ADOxx.

Your BOC Team

Vienna, 2008

1. Using the ADOxx Manuals

Three different user groups who use ADOxx can be distinguished:

- **System Administrators**, (in particular **Database Administrators**),
- **ADOxx Administrators** and
- **ADOxx Users**.

System Administrators install ADOxx and create the ADOxx databases. Additionally they can save and restore ADOxx databases. For database administration please refer to ADOxx Database Administration manual.

ADOxx Administrators are responsible for the configuration of ADOxx ("customising"), as well as for the administration of users, libraries, models and components.

ADOxx Users work with the ADOxx Business Process Management Toolkit. They acquire information, model and document their Business Processes and Working Environment (Organisational structures). In addition, the Business Process and Working Environment models constructed can be analysed, simulated, evaluated and if necessary transformed into an operative basis.

If ADOxx is installed as a stand-alone version, one person may carry out all the above roles.

The following ADOxx manuals exist for the different user groups:

- System-Administrators: **"Installation and Database Administration manual"**
- ADOxx-Administrators:
  - **Administration manual**
  - **Method manual**
- ADOxx Users:
  - **Introduction**
  - **User's manual**
  - **Method manual**

2. About this Manual

The following manual contains an introduction to the ADOxx Modelling Toolkit. This introduction provides a brief description of all the functionality within ADOxx. More detailed information can be found in the ADOxx user manual.

The following manual describes the ADOxx Business Process Management Toolkit version 1.0. With the help of this manual, it will be possible for you to use the functionality of the ADOxx Business Process Management Toolkit.

It is assumed, that the user of this manual possesses basic knowledge of the use of graphical operating systems and the handling of the mouse.

2.1 Structure of the Manual

The manual is split into several parts. Each part consists of chapters and subchapters, which are numbered using the Arabic system. A link to a (sub)chapter consists of the number of the corresponding chapter and the corresponding page number.

The title of the actual part is quoted in the headlines of each page. The pages of the manual are continuously in Arabic numerals (the table of contents in roman numerals).

Each picture in the manual is also numbered. Within the manual a reference to any picture will include its number and corresponding page number.

2.2 Contents of the Manual

The User manual consists of the following sections:

- ADOxx
- Fundamentals of modelling
- General information the Modelling Toolkit
- Modelling Toolkit components
- Appendix

General hints as well as an overview of ADOxx and its features can be found in the "ADOxx" section.

Within the "Fundamentals of modelling" section a background introduction to the (first) modelling of Business Processes is provided.

he "General information on Modelling Toolkit" section contains general information (regarding components) about the Business Process Management Toolkit like e.g. elements, ADOxx browser, user interface etc.

The "Modelling Toolkit components" describes the components of the Modelling Toolkit (acquisition of information, modelling, Analysis, Simulation, Evaluation and Import/Export).

In the "Appendix" general information on the Modelling Toolkit can be found.

© BOC Group 2008

3. New Features in ADOxx Version 1.0

If you have already used ADOxx version 1.0, you will find below a short description:

- of the innovations in ADOxx Modelling Toolkit (see chap. 3.1, p. 4),
- of the innovations in Administration Toolkit (see chap. 3.2, p. 6),

More detailed information can be found in the PDF file "`News in ADOxx 39.pdf`" which can be found on the ADOxx installation CD in the folder "`books`".

3.1 New Features in the Modelling Toolkit

3.1.1 General Innovations

- When using ADOxx 1.0 in Windows XP environments, all graphical elements of the user interface are now shown in the **"XP Look"**.
- ADOxx 1.0 can either be **launched with a German or an English user interface**. An appropriate option can be found in the login dialogue (and as a parameter `-lang` for the command line).

  Note: When using application libraries originating from older ADOxx versions, a library extension is necessary for enabling a multilingual view and unleashing all new possibilities.
- Because of an internal switch to MVC Architecture, **several views at a time of one model** are possible. Furthermore, the model view is now always consistent with the real model contents - it is no longer necessary to refresh the model view manually. The resulting performance gain allows for keeping the model view up-to-date at all times without burdening older computer systems unnecessarily - manual refreshing is no longer necessary. As a consequence, the dialogue "Overview Representation" was removed. All preview windows are now instantly drawn or refreshed.
- With ADOxx 1.0, the traditional "ADOxx Standard Application Library" is replaced with the new **"ADOxx BPMS Modelling Method"**. This library, as its predecessor, is based on the BPMS Paradigm, but contains some new model types and special features.

  Note: Models based on the "ADOxx Standard Application Library" can be transferred trouble-free and effortless to the "ADOxx BPMS Modelling Method" using the import option "Import objects from different library".
- The new tool **"ADOxx Navigator"** shows an overview of the active model in a so-called "docking window" (can be moved across the user interface and docked to special areas of the application window). The part of the model, which is currently visible in the model window, is highlighted an can be used for navigating through the model.

  Note: The Navigator replaces the function "Viewable Area" from the zoom dialogue. At the same time, the accuracy of the preview was improved.
- The new tool **"ADOxx Inspector"** (another "docking window") supplies the user with various details about the active modelling environment.
- **The functional range of the ADOxx Explorer was substanially extended**:
  - Models can be copied via the functions "Copy" (Ctrl+C) and "Paste" (Ctrl+V). This creates a copy of the model, the references pointing to the model remain with the original copy.
  - Model groups can be copied via the functions "Copy" (Ctrl+C) and "Paste" (Ctrl+V).

- Model references can be copied to other model groups via the function "Model reference(s) -> Copy".
- Model references can be removed from model groups via the function "Model reference(s) -> Delete".
- Model groups and models can be moved via the functions "Cut" (Ctrl+X) and "Paste" (Ctrl+V).
- The last model reference of a model can no longer be deleted, to prevent models being moved to the model pool unintentionally. Of course, the model itself can be deleted as ever; this naturally removes the model reference too. (Note: The Pool contains all models not accessible via model groups. They can be moved to the model group structure by the ADOxx administrator.)

- **Editing browser cells in the ADOxx browser is now more comfortable**: For simple attributes (numbers, strings) editing can be simply started by navigating to the cell and starting typing (without preceding double click).
- In the menu "Model", a new menu entry **"Recent models"** provides a list of all models recently opened by the active user.
- Performance has been improved in different areas.

3.1.2 Modelling

- The "Undo" functionality was extended and now offers both **multiple undo and redo** steps. In addition, the scope of undo-able functions was extended; creating objects and connectors and changing the view mode is now also included.
- **Some new functions in the Graphical Model Editor**:
  - The maximum size of the drawing area was increased to 50m x 50m (until now 4,5m x 4,5m).
  - During moving or resizing objects, now the actual state of the object is shown (no longer only a silhouette). This e.g. enables to see during resizing, how the text in a Note will look.
  - The model view was redesigned to prevent jittering during refresh operations.
- **Showing submodels**: An object containing a reference to a different model (e.g. a Subprocess object) can be "expanded". This means that instead of the object, a view of the drawing area of the referenced model is shown.
  - With the "show submodels" function all subordinated models over an arbitrary number of levels can be sown (even recursive!).
  - Showing/hiding submodels is possible both for single objects and for whole models.
  - Submodels can be shown even if they belong to a different model type than the main model.
  - Expanded objects are still accessible (e.g. for opening the notebook).
  - Changes to a model are instantly refreshed in all models having the changed model shown as submodel.
  - Printing and generating graphics from the expanded view is possible.
  - The expanded submodel view can be reset via the context menu or a special hotspot (upper right corner of the expanded view).
- **New features in the Modelling Bar**: Swimlanes are now shown in the Modelling Bar like all other modelling objects. There they can be selected and used as usual. Furthermore, the context menu of the Modelling Bar was substantially extended: View modes can be changed here as well (up to now only via the menu "View") and parts of the contents of the Modelling Bar (e.g. swimlanes or connectors) can now explicitly be hidden without changing the view mode.

- The **search algorithm** was extended and now allows, apart from the classic search for visualised attribute values, **searching all attribute values** (model and object attributes) either in the active or in all loaded models. The search result is shown in a "docking window" (with additional features). On clicking on a search result, the appropriate model is shown and the object containing the searched pattern is highlighted. In addition, all attribute values contained in the search result can be edited directly. The search results remain available as search history. This enables navigating based on different results.
- **Connector representation**: The dialogues "Connector marks" and "Representation of connector edges" from earlier ADOxx versions are now united in one dialogue "Connector representation". Thus, all connector settings are accessible at the same time.
- Now it is possible to **assign Attribute Profile References** in the tabular modelling too.

3.1.3 Import/Export

- **New ADL 3.9 code**: "Class Attributes" of relations (graphical representation, Notebook definition etc.) are no longer exported with the standard ADL export. If this redundant attribute information is needed for some reasons, it is still possible to export in ADL 3.81 code (as it is in ADL 3.0 code).
- The ADL Import can now handle **over-long reference attributes** without problems.
- The XML files generated by the XML Export are now **UTF-8 coded** (Unicode).

3.2 New Functionality in the Administration Toolkit

New features in Administration Toolkit are included in the following components or functions:

- **General news and improvements** (see chap. 3.2.1, p. 6)
- **User Management** (see chap. 3.2.2, p. 6)
- **Library Management** (see chap. 3.2.3, p. 7)
- **Model Managemet** (see chap. 3.2.4, p. 7)
- **LEO, AdoScript and Expressions** (see chap. 3.2.5, p. 8)

3.2.1 General news and improvements

- When using ADOxx 1.0 in Windows XP environments, all graphical elements of the user interface are now shown in the **"XP Look"**.
- ADOxx 1.0 can either be **launched with a German or an English user interface**. The available languages can be found in the file `lang.ini` listed as Windows Language IDs. The first entry is always the default language. This ensures that e.g. in a German version English can be determined as the default language. Via command line a language can be determined using the parameter `-lang` together with either an ISO-639 code or a Windows Language ID.

  Examples: `-langen` for English or `-lang1031` for German.
- Performance has been improved in different areas.

3.2.2 User Management

- Before starting the import of all system user groups and their users of a selected domain, it is now possible to **select certain system user groups** via a new option. Afterwards, only the users of

the selected groups are imported. Especially in large domains with many system user groups this speeds things considerably up.

3.2.3 Library Management

- It is now possible to define **restriction rules for passwords** (e.g. "at least six characters"). This rule is checked when a user changes his password.
- The **definition possibilities for page layouts** were substantially enhanced. Now a lot of new elements, like lines or multiple bitmaps, can be used.
- **Incrementing the change counter** in referencing models after renaming (pool) objects can be disabled.
- Because of the multilingualism feature, **self-made menu entries can be made multilingual** too.
- Set **filters in administration queries** can now be seen via the button text in the settings.
- **The GraphRep features were substantially extended**:
  - In **PEN** elements, pen style (dashed, dotted) and line thickness can now be combined. Furthermore, various styles for line end caps and joints are available.
  - In addition to Bitmaps, the following **graphic file formats** can now be visualised:

    GIF, JPEG, ICO, PNG, TARGA, TIFF, WBMP, XPM.
  - Basic shapes can now be transformed into arbitrary forms using **clipping commands** (CLIP_RECT, CLIP_ROUNDRECT, CLIP_POLY, CLIP_ELLIPSE) and combining operators.
  - **Colour gradients** can now be painted.
  - Using the BEZIER element **Bézier curves** can be drawn.
  - **Loops** (WHILE, FOR) can be programmed.
  - The two **connector marks** of a pair can now be displayed differently - depending on being at the outgoing or the ingoing side of the connector.
  - Direct access to certain **information from Record attributes** now allows for displaying whole tables on the drawing area or showing different attribute content depending on the current context.
  - Displaying different attribute values depending on the user interface language is possible.
- The same attribute can now appear in more than one chapter of a **Notebook**. Furthermore, the Notebook structure can now be made language-dependent: Depending on the chosen user interface language, different attributes can be shown.
- For the **Predefined Analysis/Evaluation Queries**, there now is a new type of input field: the "Relation field", containing all relations of the model type in question.

3.2.4 Model Management

- **New ADL 3.9 code**: "Class Attributes" of relations (graphical representation, Notebook definition etc.) are no longer exported with the standard ADL export. If this redundant attribute information is needed for some reasons, it is still possible to export in ADL 3.81 code (as it is in ADL 3.0 code).
- The ADL Import can now handle **over-long reference attributes** without problems.

3.2.5 LEO, AdoScript and Expressions

- **New Events:** `AfterEditAttributeValue`, `BeforeCreateModel`, `DiscardRelationInstance`.
- **Extended Event:** `RenameModelThread`.
- The **AdoScript** command `REBUILD_DRAWING_AREA` is no longer needed, as the drawing area is now always refreshed automatically whenever necessary. Existing commands in old scripts are ignored.
- It is now possible to **get the current user interface language** via AdoScript in order to deduce language-specific reactions.
- Extensions to the **MessagePort "AdoScript":** `CREATE_OUTPUT_WIN`, `OUT`, `SET_OUTPUT_WIN_SUBTITLE`, `FILE_EXISTS`, `SERVICE`.
- Extensions to the **MessagePort "Core":** `MOVE_INCOMING_INTERREFS`.
- Extensions to the **MessagePort "CoreUI":** `SET_OBJ_FOREGROUND`, `SET_OBJ_BACKGROUND`, `RESET_OBJ_FOREGROUND`, `RESET_OBJ_BACKGROUND`.
- Extensions to the **MessagePort "Application":** `SET_MENU_ITEM_HDL`, `REMOVE_CONTEXT_MENU_ITEM`.
- Extensions to the **MessagePort "Modeling":** `SET_ATTR_ACCESS_MODE`, `GEN_GFX_STR`, `GEN_GFX_FILE`, `GENERATE_GFX`, `COMPUTE_REGION_IMAGE_MAP`, `SET_REPRESENTATION`, `GET_REPRESENTATION`, `CLEAR_UNDO_REDO`.

4. Note

Designation of people within this documentation:

We would like to explicitly state here that wherever the third person singular is used within this manual, this is intended to include female as well as male persons. The use of the male form of description is not intended to be discriminatory in any way but is simply used in order to ensure consistent descriptions.

Pictures within this documentation:

The pictures within this documentation were created using ADOxx 1.0 for Windows 2000 and Windows XP. If you are using different operating system, the appearance of some screens may be slightly different.

ATTENTION: Further publication of this documentation in any form (including copies) is not permitted without the expressed permission of the BOC Group.

Part I
ADOxx

Successful and dynamic enterprises can achieve decisive competitive advantage through the ability to adapt their Business Processes quickly to the rapidly changing market conditions and through the active arrangement of their core competencies. The increasing dynamic, globalisation and increasing competition make efficient Business Process Management an essential goal.

To enable this, the Procedural Modelling, Analysis, Simulation and Evaluation of Business Processes is a decisive success factor. The goals of Business Process Management are the optimisation of both the processes of an enterprise as well as the resources and technology which execute those processes.

The ADOxx Modelling Toolkit, which was developed by BOC in co-operation with the University of Vienna offers essential tool support for re-engineering and reorganisation of projects.

ADOxx was specifically designed for the particular needs of financial services organisations and provides functionality primarily for the following application areas:

- Business Process Optimisation / Business Process Re-engineering (BPR)
- Quality Management / ISO 9000 Certification and Maintenance
- Controlling (Process Costing)
- Personnel Management (Personnel and Resource planning)
- Organisation Management (Enterprise documentation, Job descriptions etc.)
- Information Management (Creation of technical concepts for IT systems, Interfaces to Workflow and CASE systems, Introduction of ERP Systems)
- Creation of electronic handbooks which can be made available over an intranet with the use of powerful multi media functionality
- Evaluation of Business Processes (Benchmarking, Monitoring, "Should-be" comparison)

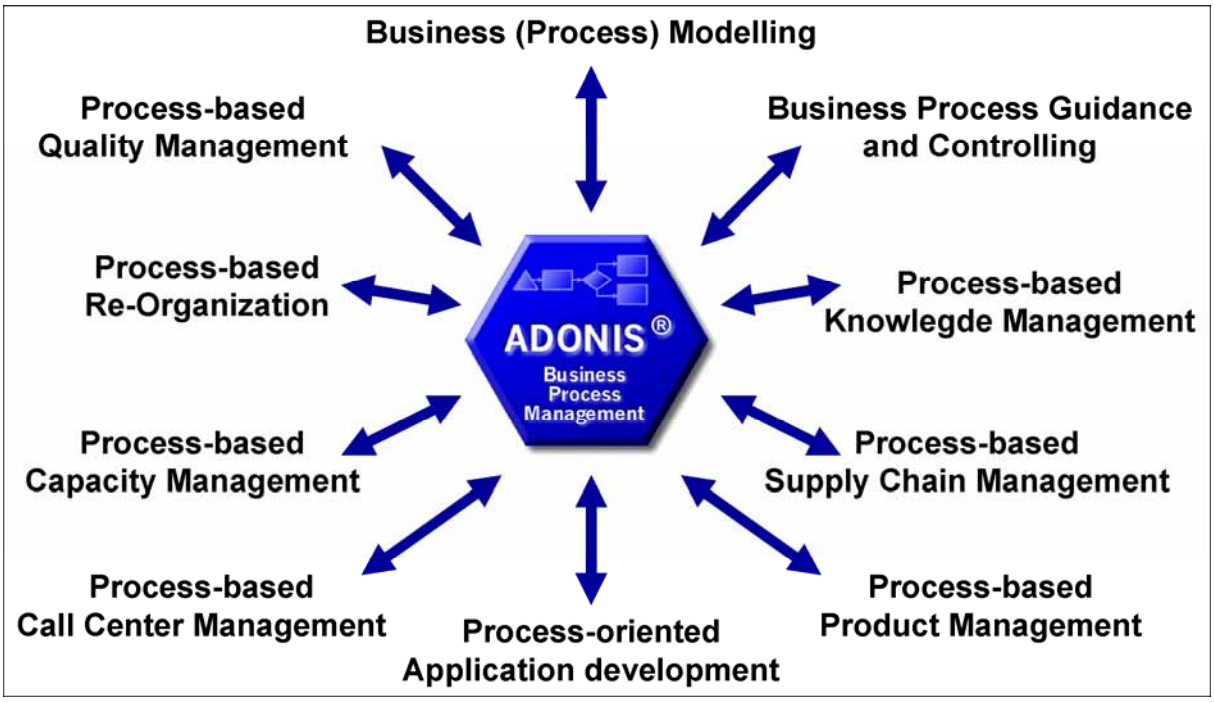


Figure 1: Modelling using ADOxx

The "Meta-concept" of ADOxx means that through customising, the tool can be configured to optimally suit the particular requirements of any user. The ADOxx user can decide by himself how he should build the processes and how he can best use the ADOxx mechanisms. As ADOxx is method independent, the management of Business Processes on different levels is guaranteed.

ADOxx® as a tool for business process management and also complies with the requirements for software ergonomics (see chap. 17., p. 803).

© BOC Group 2008

1. General Notes

The use of the ADOxx software is primarily subject to the conditions of the licence agreement. Insofar as not agreed otherwise in the licence agreement, the following terms and conditions for usage of the ADOxx software apply:

The customer is informed that all rights relating to the ADOxx software, in particular the copyright and inclusion of all documents and information supplied in the course of contract negotiation as well as any which are currently pending belong to BOC.

BOC transfers to the customer, the non-exclusive and time restricted right to use the ADOxx software in accordance with the conditions of usage specified in the licence agreement.

"Usage" is the total or partial copying (inputting) of the machine-readable software into the specified machine for the processing of the instructions or data contained therein.

The customer may produce a copy of the ADOxx software for backup purposes only. No BOC copyright notice displayed on the software or in the documentation may be changed, amended, replaced or deleted. Reproduction of the documentation is not permitted; additional copies can be obtained from BOC at the current fee.

A separate licence is required for the usage of the ADOxx software on a machine other than the specified machine, except in instances where the specified machine is out of order; in this case the temporary usage of the software on another machine is permitted. The customer agrees to inform BOC without delay of the serial number and location of the specific machine on which ADOxx has been installed, if requested to do so by BOC.

The customer is not permitted to wholly or partially translate back into source code, reverse engineer or decompile the software.

The customer agrees not to make the ADOxx software, including any copies thereof, available to third parties at any time.

ADOxx is a registered trademark of BOC. Copyright BOC Group 1996-2008. All rights reserved.

All product and companies names (see chap. 2., p. 13) are subject to the copyright law of each company. The usage of product and brand names in this manual may not be construed as permitting third parties free usage of such names in breach of copyright legislation

Hint: The program ADOxx contains some software, which has been developed by the Apache Software Foundation (http://www.apache.org/).

As outlined in the terms and conditions of the licence agreement, BOC guarantees the functionality of the software. In so far as not agreed otherwise in the licence agreement, the guarantee period extends six months from the end of the software purchase period.

The liability of BOC is as outlined in the licence agreement. Insofar as legally permitted, BOC will not accept liability for atypical damage, solely financial damage, loss or damage to recorded data, direct or indirect damage, lost profits, expected but unachieved savings or for damages arising from third party claims against the customer.

2. Trademarks

Company	Trademark
Apache Software Foundation	***Xerces***
BOC Asset Management GmbH	***ADOit***,
	ADOlog,
	ADONIS,
	ADOscore,
	BPMS,
	Management Office
Intel Corporation	***Pentium***
International Business Machines Corporation	***DB2***,
	DB2 Client Application Enablers,
	DB2 Connect,
	DB2 Universal Database,
	FlowMark,
	Informix,
	Lotus Notes,
	LOVEM,
	MQSeries,
	Notes,
	OS/2 Warp,
	Rational Rose,
	WebSphere
Macrovision Corporation	***InstallShield***
Microsoft Corporation	***BizTalk***,
	Excel,
	Microsoft,
	SharePoint,
	Visio,
	Visual Basic,
	Webdings,
	Windows,
	Windows NT,
	Wingdings
microTOOL GmbH	***case/4/0***
	objectiF

© BOC Group 2008

Oracle Corporation	***Oracle***
SAP AG	***R/3 System***
X/Open Company Limited	***UNIX***

3. ADOxx Administration Toolkit

The ADOxx Administration Toolkit supports the effective use of ADOxx. The administration of ADOxx users and user groups, ADOxx libraries as well as ADOxx models and model groups is carried out within the ADOxx Administration Toolkit.

More detailed information on the Administration Toolkit can be found in the Administration manual as well as in the online documentation of the toolkit.

The ADOxx Administration Toolkit consists of the following components:

User Management

The User Management component allows the ADOxx administrator to create and authorise ADOxx users. By assigning ADOxx users to ADOxx user groups, the access rights of the users to model groups and models can be defined. The creation, alteration and deletion of users and user groups is also possible within this component. In addition, it is possible to export ADOxx users and user groups into **UDL files** as well as to **import them from UDL files**. UDL stands for **U**ser **D**efinition **L**anguage. With the help of UDL Import/Export it is possible to transfer ADOxx users and user groups between ADOxx databases. UDL export can also be used as a backup mechanism for your ADOxx users and user groups.

Library Management

In ADOxx every user has access to an ADOxx application library with which he can create models. The ADOxx-Default-Library is the default group of objects and relations that support modelling. Within the Library Management component, ADOxx application libraries can be imported, exported and deleted. Library and class attributes can also be altered (customised) within this component.

Editing and checking the library and class attributes of the application libraries takes place within this component. Additionally, application libraries can be exported into ABL files and ABL files can be imported into the ADOxx database. (ABL = **A**DONIS **B**inary **L**anguage). With the help of ABL import and export, application libraries can be transferred to different ADOxx databases. ABL export can also serve as a method of backing up your application libraries.

Model Management

Within the Model Management component, the ADOxx administrator can import, export and delete ADOxx models, model groups and application models. In addition, model groups can be created and altered. The access rights of user groups (and therefore users) are assigned to ADOxx model groups. Model management allows you to set up the access rights to the models of the ADOxx database.

Attribute Profile Management

In the Attribute Profile Management, the ADOxx administrator may add, delete, edit, export, or import the ADOxx attribute profiles and attribute profile groups. Attribute profiles represent the repository concept in ADOxx. The attribute profile represents one or more attributes, that may be reused for any

object and can be centrally maintained. Attribute profiles are embedded in the user rights system, to enable unified and controlled care over the attribute profiles.

Component Management

Using the Component Management, additional components of **ADOxx** can be made available to the ADOxx users. Furthermore, the current configuration of the Modelling Toolkit can be checked within this component.

> **Hint:** A more detailed description of the ADOxx Administration Toolkit can be found in the ADOxx Administration manual.

4. ADOxx Modelling Toolkit

You can construct your models using the ADOxx Business process management Toolkit.

The Modelling Toolkit provides a large number of components and modules, which enable yout to enrich, analyse, simulate and evaluate your models.

The ADOxx Modelling Toolkit consists of the following components:

Information Acquisition

Information Acquisition supports you in gathering information, which is important (or necessary) to successfully model your Business Processes and Working Environments. One of the methods provided is the use of the **acquisition tables or HOMER component**, which runs through a link to Microsoft Excel. Data can be entered in these tables (excel sheets) and this data can then be exported to an ADL file and imported into the ADOxx Modelling Toolkit.

Modelling

The Modelling Component is the heart of the ADOxx Business Process Management Toolkit. It allows you to build the models you require (e.g. Business Process or Working Environments). You can create and amend your own models (and the attribute values of the objects used) using the graphical editor (**model editor**) provided. Additionally, it is also possible to input attribute values through a tabular view of the model.

If you have never modelled business processes before, we recommend that you read the section "Fundamentals of Modelling" in the user manual. Some basic tips to ensure the clarity and quality of your models can be found there.

Analysis

Within the Analysis Component, **queries** on your ADOxx models can be run and **relation tables** or **predefined charts** can be produced. Both predefined and user-definable queries are provided in ADOxx. The query language in ADOxx is AQL (**A**DONIS **Q**uery **L**anguage). Creation of queries will be defined by the ADOxx administrator and will be provided to an ADOxx user.

The results of a query can be displayed either as a table or graphically. The results can also be exported to an ASCII file. In this way you can process the results further in another application (e.g. spreadsheet, word processor etc.).

An **Analytical Evaluation** of business process models can also be carried out.

Simulation

The Simulation of Business Processes and Working Environments is executed in the Simulation Component. Four simulation algorithms are available in ADOxx: **Path Analysis** simulates the Business Process models only, while the **Capacity Analysis, Workload Analysis (steady state),**

and Workload Analysis (fixed time period) simulate the Business Process models *and* the corresponding Working Environment models.

With the help of **ADOxx agents** it is possible to calculate non-standard results during simulation.

Evaluation

The Evaluation Component offers mechanisms for the Evaluation of "should-be" models as real running processes.

The Evaluation Component provides the following areas of functionality **comparative representation of results**, **Evaluation of real-time audit trails of the workflow management system MQSeries Workflow / WebSphere MQ** (IBM) and **pre-defined evaluation queries**.

As an additional option, the **ADOxx Process Costing** component as well as the **dynamic Evaluation modules** can be integrated with the toolkit.

The ADOxx Process Costing Component supports the optimisation of costs, especially overhead costs and supports the identification and evaluation of possibilities for savings. Dynamic Evaluation Modules, based on the simulation algorithm "Capacity Analysis", enable the period-related evaluation (e.g. Human Resource Management), for which the calculation algorithms can be customised.

Import/Export

The Import/Export Component provides the option of **exporting** ADOxx models, model groups and application models **into ADL or XML files** as well as **importing** them into the ADOxx database from such files. ADL stands for **A**DONIS **D**efinition **L**anguage. With the help of this component you can transfer ADOxx models, model groups and application models into a different ADOxx database. Additionally, ADL or XML Export can serve as a back-up mechanism for your models, model groups and application models.

The **Documentation** (part of the Import/Export Component) provides you with the possibility to transfer your ADOxx models into an electronic model documentation (e.g. HTML, XML data) or in a paper documentation (e.g. DOC files for Microsoft Word). In this way it is possible to distribute the contents of your models either in document format (through a word processing program) or over an Intranet.

Hint: You will find a detailed description of the ADOxx Modelling Toolkit within the section "Modelling Toolkit" in the ADOxx user manual.

5. ADOxx Product Palette

ADOxx version 1.0 is offered as:

- ADOxx Business Edition (see chap. 5.1, p. 19) or
- ADOxx Professional Edition (see chap. 5.2, p. 20).

The following **additional components** are not contained in the standard configuration of ADOxx (Business or Professional Edition) and can be acquired additionally and integrated into ADOxx (into any existing modelling method):

- "Process Cost Analysis" (see chap. 5.3.1, p. 20)
- "Personnel and Capacity Management" (see chap. 5.3.2, p. 20)
- "Call Centre Management" (see chap. 5.3.3, p. 21)
- "case/4/0 interface" (see chap. 5.3.4, p. 21)

To use ADOxx (Business and Professional Edition) a set of different **modelling methods** (see chap. 5.4, p. 22) is available.

By utilising our **interfaces** (see chap. 5.5, p. 22) data from various applications can be exchanged with ADOxx (Business and Professional Edition).

Additionally ADOxx supports further **standards und management methods** (see chap. 5.6, p. 24) for the process management.

> **Hint:** For further information regarding products and services of the BOC Group please contact your ADOxx consultant.

5.1 ADOxx Business Edition

Within the "**ADOxx Business Edition**" the following components of the ADOxx Modelling Toolkit are available:

- Modelling
- Analysis
- Documentation
- Import/Export

> **Hint:** The components "Process Cost Analysis (Process Costing Component)", "Personal- and Capacity Management", "Call Centre Management" as well as "case/4/0 interface" are additional components and therefore they are not contained in the ADOxx Business Edition.

When needed it is possible to acquire new components or further modules within a component of the ADOxx Modelling Toolkit, which then can be provided for the ADOxx users by entering a new licence key. Alternatively it is also possible to further restrict the composition of the ADOxx Business Edition.

Hint: When using the ADOxx Business Edition the smart-icons of unsupported components will not be displayed in the component bar. The same applies to smart-icons in the quick-access bars. If the ADOxx user or the ADOxx administrator did not change the configuration, only the smart icons corresponding to the functionality of ADOxx Business Edition will be displayed.

5.2 ADOxx Professional Edition

Within the "**ADOxx Professional Edition**" all components of the ADOxx Modelling Toolkit are available.

Hint: The components "Process Cost Analysis (Process Costing Component)", "Personal- and Capacity Management", "Call Centre Management" as well as "case/4/0 interface" are additional components and therefore are not contained in the ADOxx Professional Edition.

5.3 Additional ADOxx Components

The additional components:

- "Process Cost Analysis"
- "Personnel and Capacity Management"
- "Call Centre Management"
- "Case/4/0 interface"
- "ObjectiF interface"

can be integrated into your existing ADOxx installation. They are not contained in the default configuration of ADOxx (Business or Professional Edition).

5.3.1 ADOxx Process Cost Analysis

The **Process Cost Analysis** supports cost optimisation, especially overhead costs. The system is capable to set informative analysis on the available savings possibilities.

The ADOxx Process Cost Analysis simulates the Business Process models. This identifies further potential for analysis and optimisation of the process costs - the course of the control structure can contain decisions and some parallel structures and above this many optional process levels are allowed.

5.3.2 Personnel and Capacity Management

The **Personnel and Capacity Management** supports the time period based evaluation of Business Processes and Working Environments. In particular the periods of validity assigned in the period-related versioning will be used.

The **key figures** determined in the Personnel and Capacity Management **can be corporate-specific configured**. Besides quantitative evaluation **quality evaluation** will also be supported. For instance, questions such as "What are the roles and skills that I need to realise my Business Processes?" can be answered.

5.3.3 Call Centre Management

The **Call Centre Management Component** supports personnel planning (FTEs) and skills management within Call and Multimedia Service Centres.

The processes in these centres are used as the basis and input for crucial information regarding time, costs and knowledge. For instance:

- How long does a helpdesk query or an address change take?
- What capacity is needed for a given time interval and which costs do arise?
- How do you document for your employees which actions to take in different scenarios?

5.3.4 case/4/0 Interface

Through the coupling of **case/4/0** from **microTOOL GmbH**, **Business Process oriented application development** is possible, to create the required building blocks for a new system in case/4/0 from ADOxx and to maintain consistent links to the specific activities of the Business Processes. The link is established online from special Business Process models to an Application Development environment.

Through this direct integration from specialised Business Process models to IT models before the construction of an application system, it is possible to identify the Business Process **affected** by the new system. The functionality of ADOxx allowing personnel capacity and cycle time to be calculated is very useful for deriving the qualifications necessary to carry out the Business Processes. This means that when changing existing applications, we already have an effective **qualification system**.

The **IT company map** together with the Business Processes provide comprehensive documentation. As a result of the links between the two toolkits, changes to either the applications or the Business Processes can be accurately described and evaluated. In this way modern **Information and Migration Management** is possible.

ADOxx and case/4/0 also offers components for **the generation of documentation** which can then be distributed over the Internet or an Intranet. Through the coupling with case/4/0 Business Process and application documentation can be integrated and distributed to target groups within the enterprise.

5.3.5 objectiF Interface

Through the coupling of **objectiF** from **microTOOL GmbH Business Process oriented application development** is possible, to create the required building blocks for a new system in objectiF from ADOxx and to maintain consistent links to the specific activities of the Business Processes. The link is established online from specialised Business Process models to an Application Development environment.

Through this direct integration from specialised Business Process models to IT models before the realisation of an application system, it is possible to identify the Business Processes **affected** by the new system. The functionality of ADOxx allowing personnel capacity and cycle time to be calculated is very useful for deriving the qualifications necessary to carry out the Business Processes. This means that when changing existing applications, we already have an effective **qualification system**.

The **IT company map** together with the Business Processes provide comprehensive documentation. Because of the links between the two toolkits, changes to either the applications or the Business Processes can be accurately described and evaluated. In this way modern **Information and Migration Management** is possible.

ADOxx and objectiF also offers components for **the generation of documentation**, which can then be distributed over the Internet or an Intranet. Through the coupling with case/4/0 Business Process and Application Documentation can be integrated and distributed to target groups within the enterprise.

5.4 ADOxx Modelling Methods

The modelling in ADOxx (Business or Professional Edition) is effected applying a defined methodology (= application library). Within one ADOxx database several methods can be administered and provided to the users.

In addition to the possibility to adapt the modelling methodology to the needs of the customer the ADOxx BPMS modelling method (ADOxx-Default-Library) is contained within ADOxx (Business or Professional Edition).

The following (de-facto standards) for modelling methodologies are offered in addition:

- UML method
- LOVEM method
- E-Business method
- EPC method
- QM method (ISO9000/2000)

Hint: For further information regarding customer-specific ADOxx modelling methods please contact your ADOxx consultant.

In addition to the modelling methods mentioned above, ADOxx supports further standards and management methods (see chap. 5.6, p. 24) for the process management.

5.5 ADOxx Interfaces

Using the interfaces to/from ADOxx (Business and Professional Edition) data can be

1. imported from other applications into ADOxx,
2. edited in ADOxx and
3. exported from ADOxx into other applications.

For the tools listed below interfaces already have been realised and can be integrated and adapted for your modelling method (application library) if required:

- Data acquisition:
  - Excel (Microsoft Corporation)
  - HOMER (BOC Asset Management GmbH) - contained in the ADOxx Professional Edition
  - Visio (Microsoft Corporation)
- Workflow Management and Enterprise Integration systems:
  - BizTalk Server (Microsoft Corporation)
  - EasyFlow (Heyde AG/ TOPAS InformationsTechnologien GmbH)
  - Fabasoft Components (Fabasoft)
  - IBM MQSeries Workflow (IBM)
  - IBM WebSpere MQ Workflow (IBM)
  - ORACLE BPEL Process Manager (Oracle)

  - Staffware 2000 (TIBCO Software Inc.)
  - TIBCO Staffware i10 (TIBCO Software Inc.)
  - Unisys Workflow (Unisys)
  - WPDL (WfMC - Workflow Management Coalition)
  - XPDL export
- Simulation:
  - ARENA (Rockwell Software Corporation
- CASE tools:
  - case/4/0 (microTOOL) - interfaces provided directly in ADOxx (see chap. 5.3.4, p. 21)
  - IBM Rational Rose (IBM)
  - objectiF (microTOOL) - interfaces provided directly in ADOxx (see chap. 5.3.5, p. 21)
  - XMI Import/Export
- Software Engineering:
  - SQS-TEST (SQS Software Quality Systems AG)
- ERP systems:
  - R/3 (SAP) - reference models
  - R/3 (SAP) - Online interface to integrate R/3 calls
  - R/3 (SAP) - HR integration
  - JuraPack (PELI) - Claims management
- Groupware systems:
  - IBM Lotus Notes (IBM) - Export interface for company documentation
  - IBM Lotus Notes (IBM) - Online interface to call and integrate Notes documents
  - SharePoint Team Services für Microsoft Project 2002 Server (Microsoft Corporation)
- BPM tools:
  - ibo (ibo)
- Relational databases:
  - via an ODBC interface
- Document management systems (DMS):
  - DOMEA (OpenText Corporation)
  - IBM Lotus Notes (IBM)
  - PCDOCS (Hummingbird)
  - SharePoint Team Services for Microsoft Project 2002 Server (Microsoft Corporation)
  - Document management systems supporting ODMA

Hint: For further information regarding customer-specific ADOxx interfaces please contact your ADOxx consultant.

5.6 Support of Standards and Management Methods

ADOxx supports the following standards and management methods.

> **Hint:** Supporting these standards and management methods is performed by extension and adaption of existing modelling methods (ADOxx application libraries). For detailed information please contact your ADOxx consultant.

- Management methods:
  - TQM (Total Quality Management)
  - BSC (Balanced Scorecards)
  - ERM (Enterprise Risk Management)
  - Six Sigma
  - HL7 (Health Level Seven)
  - Basel II
- Business process management/Business modelling:
  - SOX (Sarbanes-Oxley-Act)
  - BPMN (Business Process Modelling Notation)
  - eEPK (Ereignisgesteuerte Prozeßketten)
  - ISO 900x
  - Zachmann Framework
- Enterprise Application Integration (EAI) and Workflow systems:
  - BPEL (Business Process Execution Language)
  - BPMN (Business Process Modelling Notation)
  - WPDL (Workflow Process Definition Language)
  - XPDL (XML Process Definition Language)
- Software development:
  - MDA (Model Driven Architecture)
  - UML2 (Unified Modelling Language)
  - XMI (XML Metadata Interchange)
- Standards for data exchange:
  - XML (Extensible Markup Language)
  - ODBC (Open Database Conectivity)
  - ODMA (Open Document Management API)

Part II
Fundamentals of modelling

In this chapter you will find short instructions how to model business processes and working environments with use of ADOxx. There is also a description of exemplary models (see chap. 2., p. 30).

This section is a a good introduction for novice ADOxx users. It will assist the user in understanding how to properly model business processes.

The introduction chapter should be helpful to when trying to understand the modelling techniques. In the beginning, the created models can be incomplete. To avoid this, the user can find in the description of the modelling component (see chap. 2., p. 126) some helpful instructions what model types are necessary to be modelled.

Additionally, basic tips to ensure the clarity and quality of the Business Process models are provided.

With the help of the example models, the creation of the ablility to simulate models in ADOxx is presented.

1. Introduction to modelling with ADOxx

The comprehensive and integrated modelling of Business Processes takes into account the following aspects:

- Identification of Business Processes (see chap. 1.1, p. 26)
- Modelling of the flow of business processes (see chap. 1.2, p. 26)
- Modelling of the organisational structure (see chap. 1.3, p. 28)
- Entering the model information in the attributes of objects (see chap. 1.4, p. 29).

The model type Business Process Model is used to model flows of processes, and accordingly, the model type Working Environment to model organisational structure.

1.1 Identification of Business Processes

According to the BPMS Methodology, during the organisational projects following approach is followed]>:

- Identify the products or services produced by the enterprise or its part being examined
- Identify the Business Processes that answer the question "with which processes or activities are these products or services realised?"

Workshops and brainstorming techniques can be used to identify these aspects.

1.2 Process Flow

The process flow is shown in a Business Process Model. The creation of Business Process Models can be carried out in either a top-down or bottom-up.

In **bottom-up modelling** sections of the process (possibly later sub processes) are modelled in great detail. These excerpts - sub processes and single activities - are aggregated into a Business Process Model.

Top-down modelling begins at a higher level with the depiction of the overall process to which more detail is then added through further activities and sub processes.

During modelling within ADOxx both methods can be used.

Flowing modelling guidelines are provided a Business Process:

1. Preparation for modelling (see chap. 1.2.1, p. 27)
2. Modelling objects (see chap. 1.2.2, p. 27)
3. Inserting connectors (see chap. 1.2.3, p. 28)
4. Naming objects and connectors (see chap. 1.2.4, p. 28)
5. Further object modelling (see chap. 1.2.5, p. 28)
6. Control (see chap. 1.2.6, p. 28)

1.2.1 Preparation for modelling

Before starting modelling, please prepare a draft containing your working plan. The plan should contain activities and timetable concerning the work to be done. Thus, it serves as a basis throughout the actual modelling. This is especially useful for large processes. It will also be useful when deciding where to implement sub processes.

The introduction of sub processes can be used, especially when:

- the sub processes can be re-used in other business processes,
- to provide a reasonable overview of the Business Processes (guideline: ca. 20 activities per model) or
- the part of the main model, which is shown as sub process should be modelled in detail.

Your plan can also be used to design a standard naming system for activities and sub processes, which will improve clarity (e.g. infinitive + subject such as "Write letter").

1.2.2 Modelling objects

Open the Model editor (see chap. 2.2, p. 127) and place a manageable number of objects (see chap. 2.2.5, p. 132) (activities, decisions) on the drawing area.

To draw an object select it in the class list (by clicking on it with the left mouse button) and then place it on the drawing area by clicking on the required position there.

When beginning modelling, the objects "Activity" and "Decision" are the most important, because these objects enable you to model a Business Process.

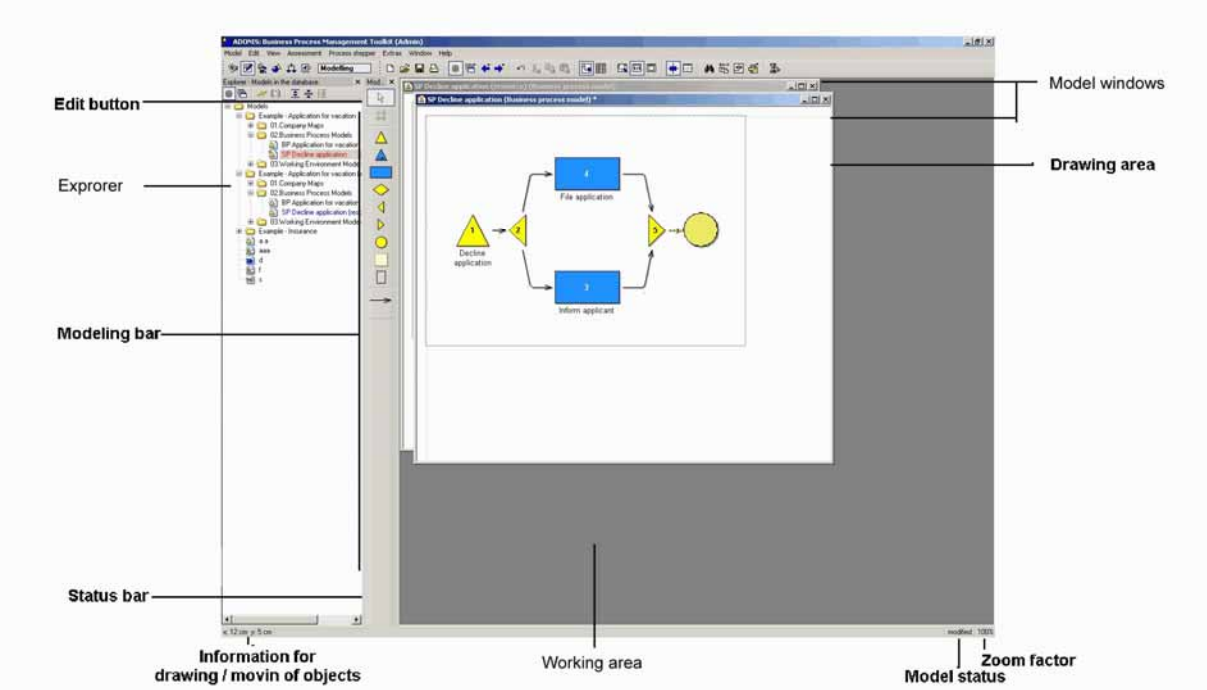


Figure 2: Model-Editor

Place an object of a class "Process start" at the beginning of your process and an object of class "End" at its end. These two objects are placed on the modelling surface not only to improve clarity of

the process, but also to be used by other ADOxx components within the Modelling Toolkit. A "Process start" and an "End" object must be defined within a Business Process Model.

Please rearrange the objects according to the flow in your previously prepared plan. Remember that you can always change the order of objects if required.

Concerning the modelling direction (horizontal/vertical), the horizontal ordering of objects has proved itself in terms of clarity in providing an overview or comparison of the Business Processes. Detailed Business Processes are more legible when created in a horizontal direction over a number of pages.

1.2.3 Inserting connectors

The flow logic is shown through the insertion of connectors, in particular through the relation "Subsequent". Through this relation, which joins any object with the next object, the logical flow of your Business Process can be identified.

To add a connector to the model, select the relevant relation from the relation list by placing the mouse on it and clicking with the left mouse button. Then click (again with the left mouse button) on the two objects in the drawing area, which this relation should join.

At the beginning you can ignore relations such as: "Sets" and "Sets variable". These are necessary for the Simulation (see chap. 4., p. 332) of Business Processes.

1.2.4 Naming objects and connectors

The next step is to name the objects (see chap. 2.2.7.12, p. 146) (activities and decisions) and the relations. The relation "Subsequent" should only be labelled when it starts at the object "Decision" and then holds the conditions under which the following activities will be executed.

1.2.5 Further object modelling

For modelling additional objects, see "object modelling" (see chap. 1.2.4, p. 28) which describes the method.

1.2.6 Control

Finally, you should check your model. An efficient manual method is to follow the flow within your model. Check the order of activities and decisions as well as the object and relation labels (see chap. 1.2.4, p. 28). Correct any invalid objects, relations and labels and insert any of them as required.

1.3 Organisational structure

The Working Environment model presents the organisational structure connected with the modelled business process. You must identify which performers are available for the execution of the Business Process. ADOxx offers the possibility of modelling the Working Environment and then connecting it with the Business Process Models. The Business Processes modelled are executed within this Working Environment. The different effects of the Working Environment on business processes can be compared in this way.

The actual structure of the enterprise (or part of the enterprise) which carries out the important Business Processes is modelled in a Working Environment model. First model the performers and

define their roles (e.g. Specialist, Secretary etc.). When modelling the organisational units (e.g. departments, enterprise areas) you can identify which performers belong to which organisational units.

1.4 Entering model information

In addition to the labels, it is possible to enter information into the attributes of the objects and relations modelled. This can be done by double-clicking on the required object or relation. This opens the ADOxx notebook, which contains many input fields, divided over various pages and chapters.

The information, which you enter in the ADOxx Notebooks provides on the one hand a description (information and documentation for you and other ADOxx users), and is also used by other components of the Business Process Management Toolkit (e.g. in the Simulation Component).

© BOC Group 2008

2. Example models

During the installation of the ADOxx Business Process Management Toolkit several ADL files with example models are installed in the ADOxx directory. These files contain a few examples of models which can be imported into ADOxx.

The example models serve to illustrate the modelling of business processes and Working Environments with ADOxx .

ATTENTION: The example models are based on the **ADOxx standard application library**.

Hint: The description of the example models is to be found in volume III of the user documentation ("ADOxx standard method manual") or the online documentation "ADOxx standard method" ("Extras" menu, "ADOxx standard method") menu item.

Part III
General information about the Modelling Toolkit

Hint: All examples and descriptions in this part of the user manual refer exclusively to the ADOxx standard application library (see chap. 16., p. 802). Text and numeric entries in the pictures may differ from those on your screen.

1. Terms and Context

In this chapter you will find an introductory description of the terms used in ADOxx as well as their context. For a better understanding, these terms and context are graphically represented as follows (see fig. 3).

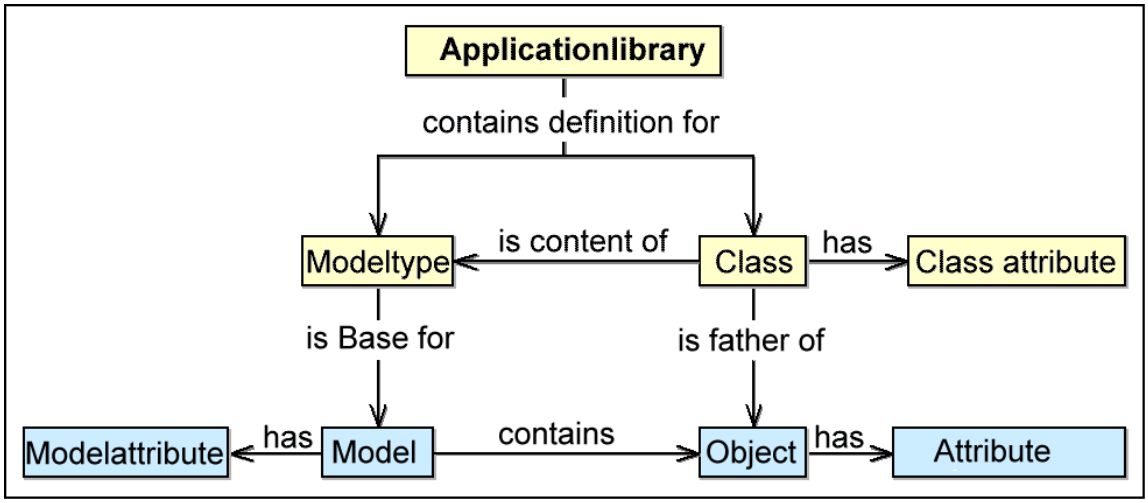


Figure 3: Terms and context in the Modelling Toolkit

Hint: You can find the name of the application library which you are assigned to in the state information (see chap. 3., p. 728) .

Application libraries contain the definitions for **model types** and **classes**.

A **model type** is a grouping of classes. The picture below shows for instance the model types of the ADOxx-Default-Library (see fig. 4).

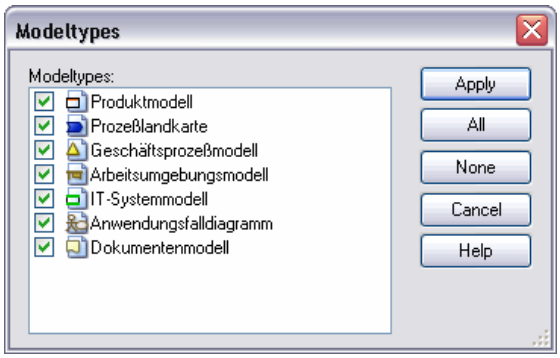


Figure 4: Model types of the ADOxx application library

Classes represent the pattern for the objects created by an ADOxx user (in ADOxx Modelling Toolkit). Classes have **class attributes**, which control for instance the graphical display of an object or the arrangement of the object attributes in the ADOxx Notebook in ADOxx Modelling Toolkit. In addition, the **(object) attributes** are also defined in classes. Each **(object) attribute** will be assigned to an attribute type (see chap. 7., p. 733) and a standard value during the definition.

Models based on a model type (e.g. Business Process Models) will be created in the ADOxx Modelling Toolkit. Models have **model attributes**, which provide general information about the model (e.g. date of creation, status).

A model is made up of **objects**, which are derived (= instanced) from classes. Objects have **(object) attributes**, which contain the information describing the model and its contents.

Example from the Modelling Toolkit:

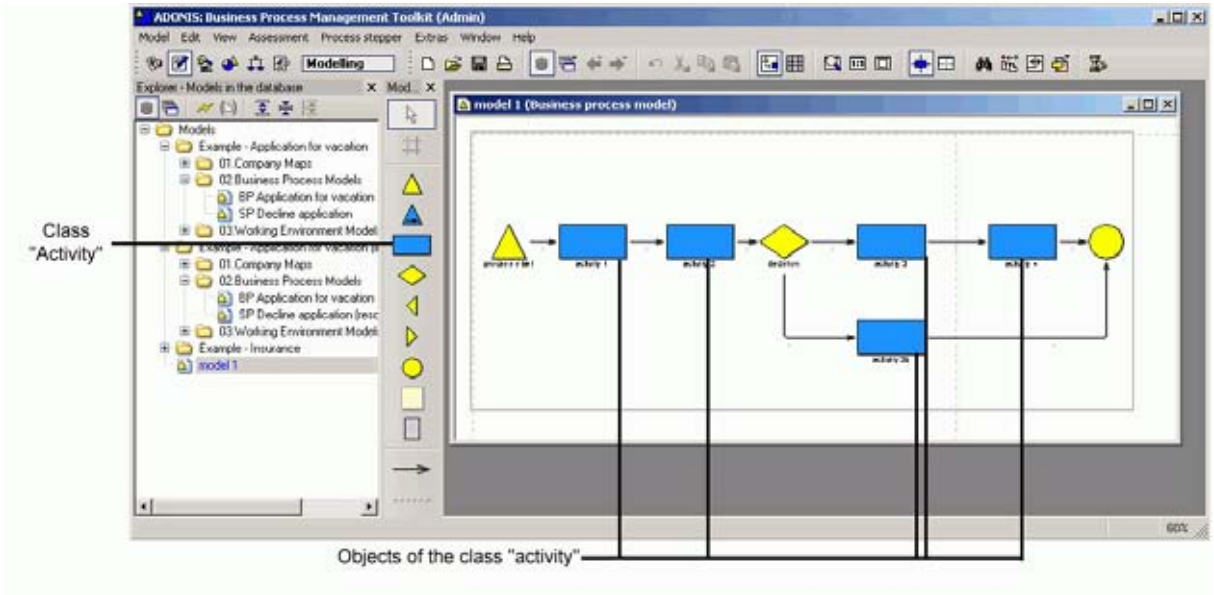


Figure 5: Model, classes and objects in ADOxx

The representation above (see fig. 5) shows the model "BP model 1" of the model type "Business Process Model".

All classes available for the modelling of models are shown in the modelling bar (see chap. 2.2.2, p. 128) on the left side, like for instance the class "activity".

Hint: The display of the available classes depends on the view mode, i.e. some classes and relations can be shrunk or expanded for the modelling.

In the model "BP model 1", objects of the available classes are modelled (= instanced), such as the objects "activity-1", "aktivity-2a", "activity-2b" and "activity-3" of the class "activity".

By double-clicking on an object with the left mouse button, you will open its ADOxx Notebook, which contains all the (object) attributes defined for this class (see fig. 6).

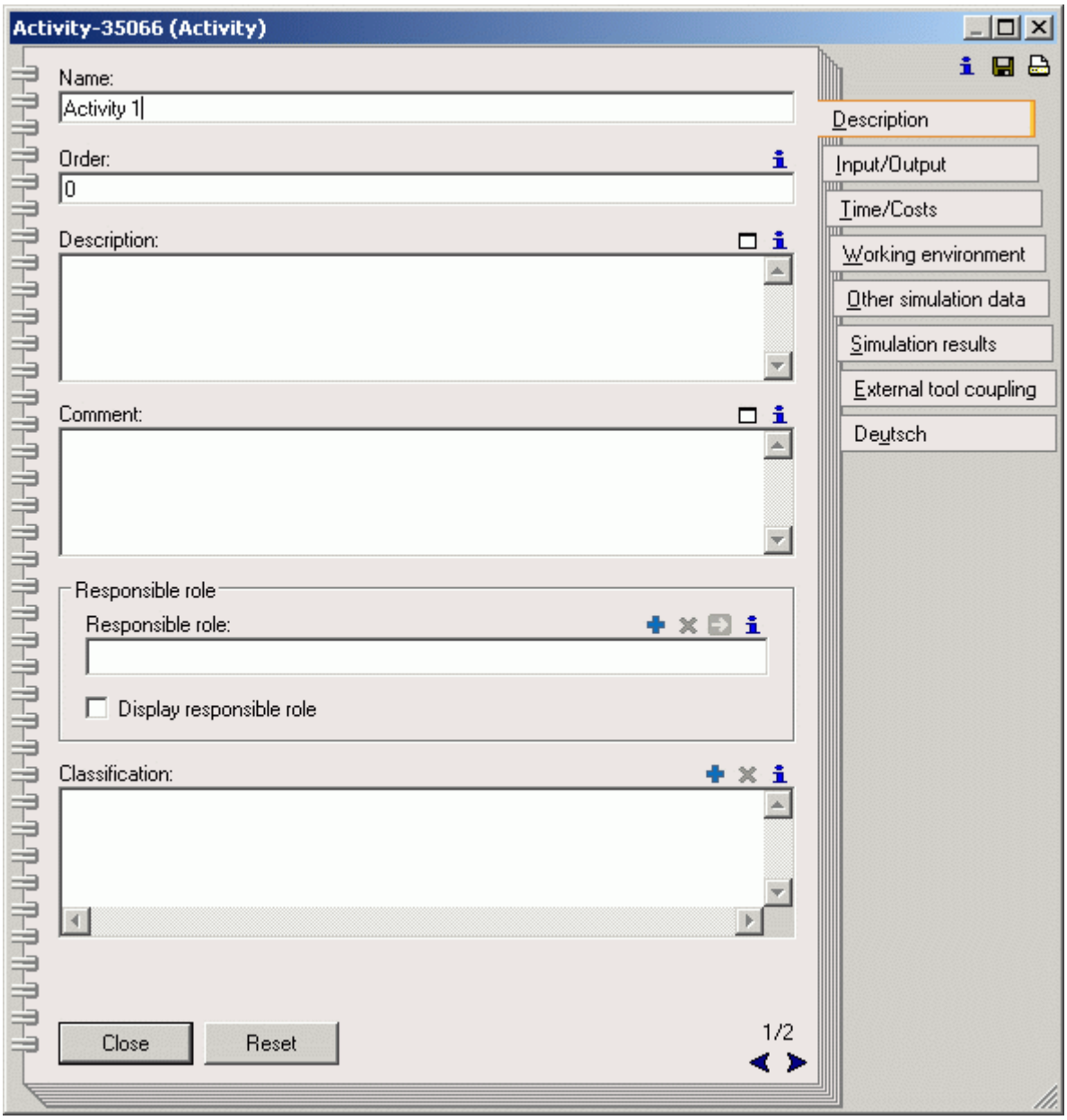
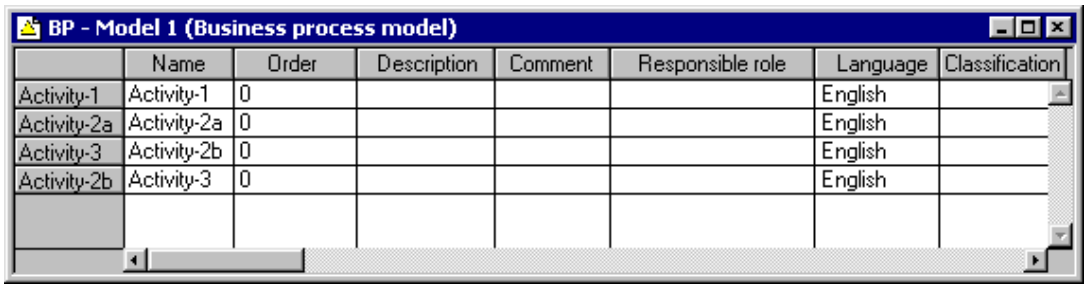


Figure 6: ADOxx-Notebook for the object "activity-1" of the class "activity"

The values for the available (object) attributes will be entered into the ADOxx Notebook according to their type of attribute (see chap. 7., p. 733).

The tabular model display enables viewing and editing of the attribute values of all the objects of a class (see fig. 7).



	Name	Order	Description	Comment	Responsible role	Language	Classification
Activity-1	Activity-1	0				English	
Activity-2a	Activity-2a	0				English	
Activity-3	Activity-2b	0				English	
Activity-2b	Activity-3	0				English	

Figure 7: Tabular display of all the objects of the class "activity"

In this way, the editing of attribute values in several objects of a class is quicker and clearer in the tabular display than in the graphical display, where you have to open and close the ADOxx Notebook of each single object.

© BOC Group 2008

2. ADOxx User Interface

The ADOxx user interface follows the structure of other applications. However, there are some additional features.

The ADOxx window (see fig. 8) appears on the screen after the ADOxx Modelling Toolkit (see chap. 1.1, p. 722) has been started. This window consists of the following six parts:

- Window bar (see chap. 2.1, p. 36)
- Menu bar (see chap. 2.2, p. 37)
- Component bar (see chap. 2.3, p. 37)
- Quick-access bar (see chap. 2.4, p. 37)
- Workspace (see chap. 2.5, p. 38)
- Explorer (see chap. 2.6, p. 38)

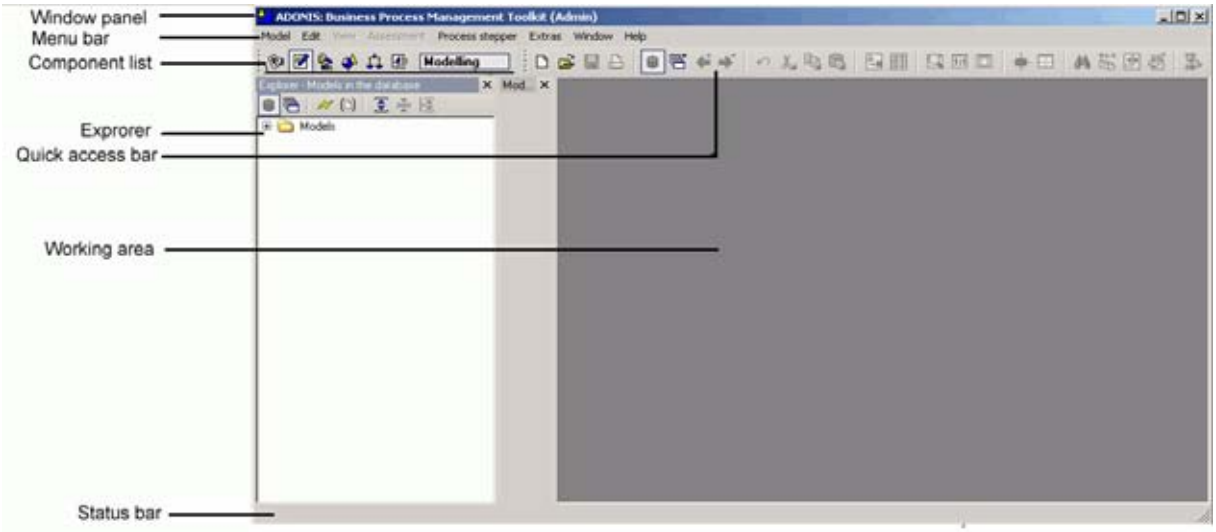


Figure 8: Working area in the Modelling Toolkit

Explorer, components and the quick-access bar, can be freely moved by means of "Drag and Drop". (hold mouse on the bar, drag it and drop it in a desirable place). They can be docked to the edge of the window or set in any place. At the end of the work, settings will be stored and used in the next program session. By pressing key ***<F11>***, all opened tools can be hidden and afterwards shown again.

2.1 Window Bar

The window bar (header bar) consists of the exit button (small program symbol), information on the working area (see chap. 2., p. 36) (ADOxx component and the user name in brackets) and the buttons "Minimize", "Maximize" and "Close" (see fig. 9).

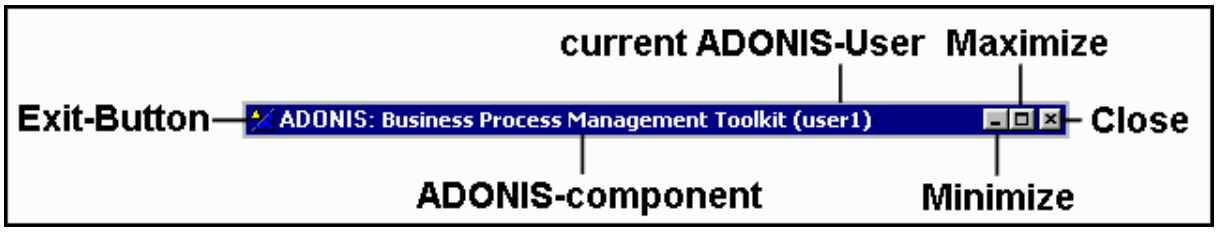


Figure 9: Window bar in the Modelling Toolkit

2.2 Menu Bar

The menu bar is placed below the window bar (see chap. 2.1, p. 36).

When starting the Modelling Toolkit, the modelling component is activated and its menus are displayed in the menu bar (see fig. 10).

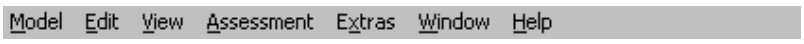


Figure 10: Menu bar (Modelling Component)

Hint: Depanding on the ADOxx configuration, the menu bar can contain more entries than shown here.

2.3 Component Bar

You can find the component bar below the menu bar (see chap. 2.2, p. 37) at the left hand side.

Figure 11: Component bar (Modelling Component)

The component bar (see fig. 11) holds the symbols (smart icons) for the single components of the Modelling Toolkit (see chap. 4., p. 17) (the components "Acquisition", "Modelling", "Analysis", "Simulation", "Evaluation" and "Import/Export". You can activate a component by clicking on the appropriate smart icon.

Alternatively you can also activate a component by clicking on the component bar (to the right of the component icons) with the right mouse button. By selection of the required component a popup menu appears. Alternatively, you can open this popup-menu by pressing the function key <F9> and activate a component by using its accelerator (=underlined letter in the component name).

The name of the component currently active is displayed to the right of the component symbols.

2.4 Quick-Access Bar

The quick-access bar is located below the menu bar (see chap. 2.2, p. 37) right next to the component bar (see chap. 2.3, p. 37)

Figure 12: Quick-access bar (Modelling Component)

The quick-access bar (see fig. 11) offers symbols for actions frequently used when working with this component.

2.5 Workspace

The workspace is the part of the window directly below the component and quick-access bar. (see chap. 2.3, p. 37). It is used especially when working within the modelling component of the Modelling Toolkit.

2.6 Explorer

ADOxx Explorer is a universal tool, that provides easy access to models and their status information. It has its own quick-access bar with special smart-icons (see fig. 13).

A detailed description can be found in the chapter "ADOxx Explorer" (see chap. 4., p. 54).

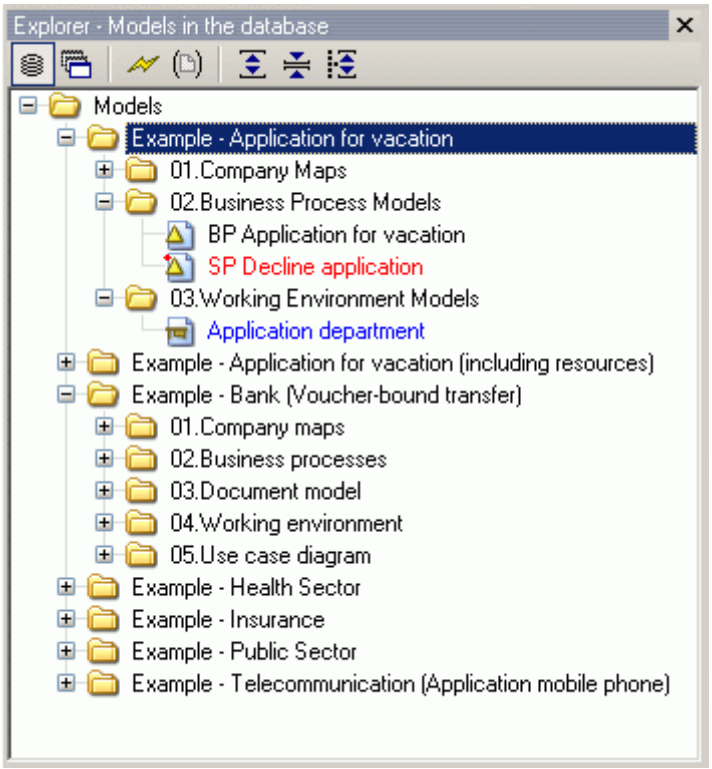


Figure 13: The ADOxx Explorer

2.7 Navigator

The ADOxx Navigator is a tool showing an overview over the active model and offering various zoom and navigation possibilities (see fig. 14).

A detailed description can be found in the chapter "ADOxx Navigator" (see chap. 5., p. 62).

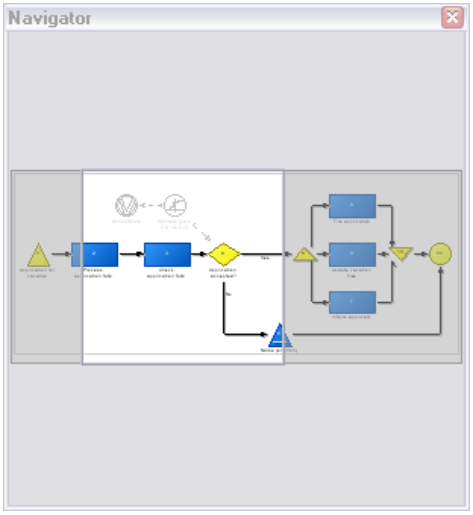


Figure 14: The ADOxx Navigator

2.8 Inspector

The ADOxx Inspector is a tool which supplies the user with various details about the active modelling environment (see fig. 15).

A detailed description can be found in the chapter "ADOxx Inspector" (see chap. 6., p. 64).

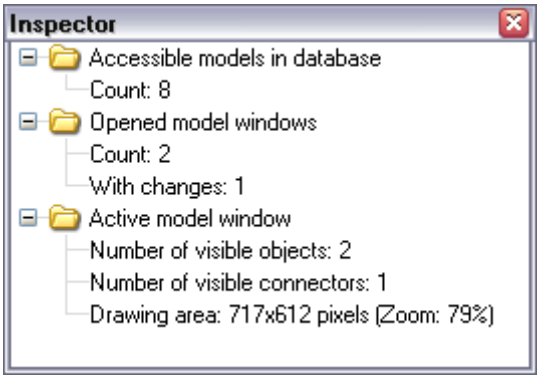


Figure 15: The ADOxx Inspector

2.9 Search Results

The ADOxx Search Results window is a tool which allows accessing both the search function and all search results of the current session. It has its own quick-access bar with special smart-icons (see fig. 16).

A detailed description can be found in the chapter "ADOxxsearch results" (see chap. 7., p. 65).

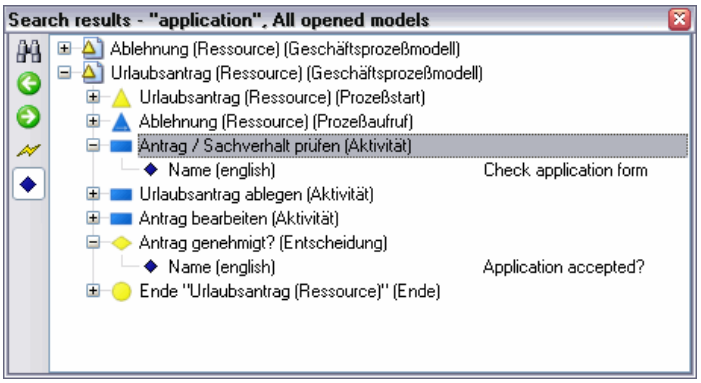


Figure 16: Der ADOxx-Suchergebnisse

3. Control Elements

Control Elements (buttons, accelerator keys) are found in a number of places in ADOxx in order to provide support.

Control Elements are available:

- **in selection windows** (see chap. 3.1, p. 41) and
- **in ADOxx Notebooks** (see chap. 3.2, p. 48).

3.1 Selection Control Elements

The global selection control elements are available in the context men (right mouse button) of every selection window (e.g. Open Model, Queries, ADL Export).

The following general functions are available in most windows as an addition to the context-specific functions:

- "Modeltypes" (see chap. 3.1.1, p. 42)

  (only available in model selection lists)
- "Manage model groups" (see chap. 2.4.12, p. 269)

  (only available in model selection lists)
- Refresh (see chap. 3.1.2, p. 43)
- Item search (see chap. 3.1.3, p. 43)
- Save as (see chap. 3.1.4, p. 45)
- "Shrink/Expand" (see chap. 3.1.5, p. 45) with the following options
  - "Show all"
  - "Hide all"
  - "Show selected"
  - "Show selected with sublevels" and
  - "Shrink selected"

  (only available in hierarchical selection lists)
- "Select all items" (see chap. 3.1.6, p. 46)
- "Deselect" (see chap. 3.1.7, p. 46)
- "Selected items" (see chap. 3.1.8, p. 46)
- "Shrink/Expand time related versioning" (see chap. 3.1.9, p. 47)

  (This function is only available in model and attribute profile selection lists if time related versioning is being used.)

3.1.1 Model Types

A list of all model types defined in the application library is displayed in the window "Modeltypes" (see fig. 17).

Hint: If your application library contains a customised model type filter (see chap. 3.1.1.1, p. 42), this filter will appear instead.

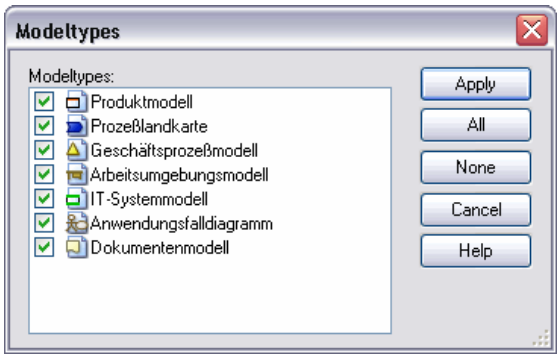


Figure 17: Model types of the ADOxx-Default-Library

To the left of each model type, a checkbox indicates, whether the model type is currently displayed (checked ☑) or not (unchecked ☐). Click into the checkbox to change the status.

If you click **"All"**, all model types are activated. By clicking on **"None"**, all model types will be deactivated.

After the successful change, click **"Assign"** to display the updated model selection list, i.e. only the models of the activated model types are displayed.

Hint: By default, all model types are activated.

Hint: The "model types" menu item is only available in model selection lists.

ATTENTION: Changed settings will be reset when exiting ADOxx.

3.1.1.1 Model Type Filter

The model type filter is a special kind of model type representation. It is not a standardised ADOxx function but is - by request - added to an application library instead. The model type filter allows the user to choose from various predefined views on the available model types (see fig. 18).

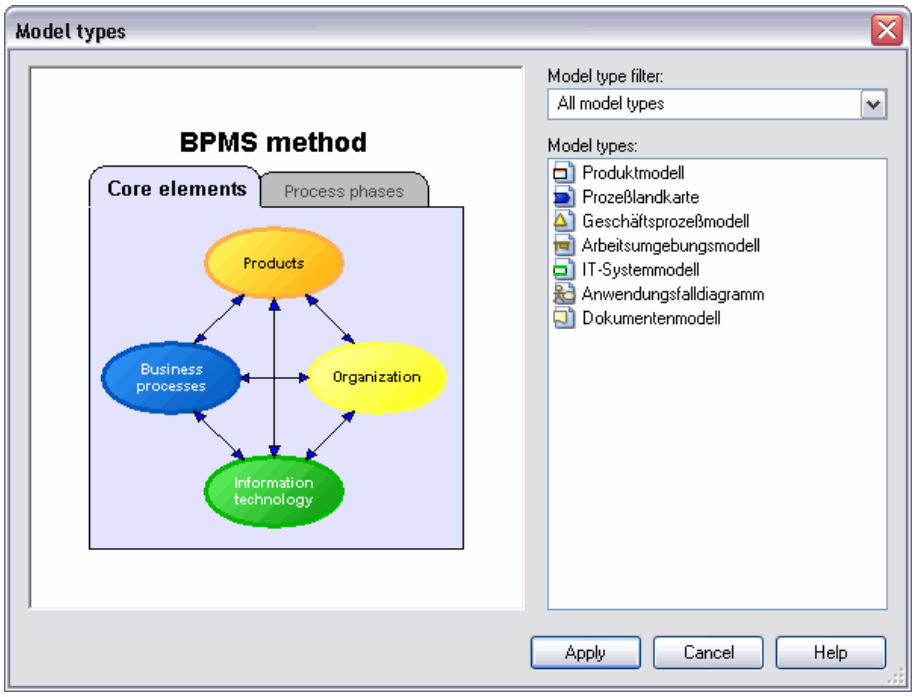


Figure 18: Model type filter of the ADOxx-Default-Library

The left part of the filter window usually has an embedded graphics with hotspots - **the scenario diagram**. Dependin on which graphical element is clicked, the model type list to the right displays exactly those model types which are used in the selected scenario (part). Thus even complex method definitions with more than ten model types are presented tailor-made to the use case on the user interface.

Instead of clicking the hotspots in the scenario diagram you can use the dropdown-list **"Model type filter"** to select one of the scenarios provided. Furthermore, this list contains an additional entry: **"Custom"**. If you choose this variant, an extra button **"Change"** appears. Click on this button to open the model type list (see chap. 3.1.1, p. 42) and select your desired model types manually.

ATTENTION: Changed filter settings will be reset when exiting ADOxx.

3.1.2 Refresh

By selecting "Refresh" (Icon or key **<F5>**), the selection list will be updated with the current status of the ADOxx database.

3.1.3 Item Search

By selecting "Item search" (shortkey **<Ctrl>+F**) you can search for entries in the list. When this button has been selected, the window "Entry search" is displayed (see fig. 19).

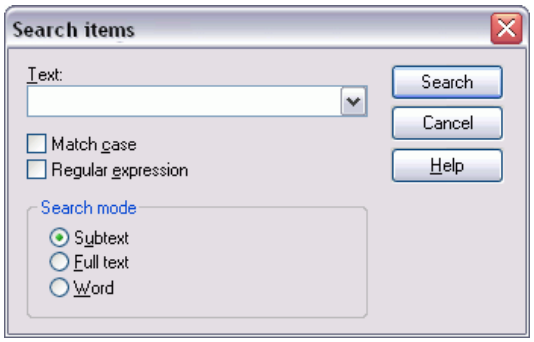


Figure 19: Entry search

Enter the text for which you wish to search into the field "Text" (or select an item already in the list).

The option "Match case" determines whether or not the actual case of the text for which you are searching should be taken into account.

The option "Regular expression" allows a search for regular expressions (see chap. 6., p. 731).

Additionally you can specify the type of search you are carrying out:

- **"Subtext"**: means that the text being searched for can be part of a larger word or sentence.
- **"Full text"**: searches for an entire piece of text, which matches the text for which you are searching.
- **"Word"**: assumes that the piece of text you are searching for is an entire word and will only locate matching words. A word must be a single grouping of text, flanked by blanks, a full-stop or the like.

Start the search by clicking "Search". If entries matching the search criteria are found in the list, the search results will be displayed (see chap. 3.1.3.1, p. 44).

3.1.3.1 Display Search Results

All entries matching the search criteria ae displayed in the window "Found entries "<Search criteria>"" within their hierarchy (see fig. 20).

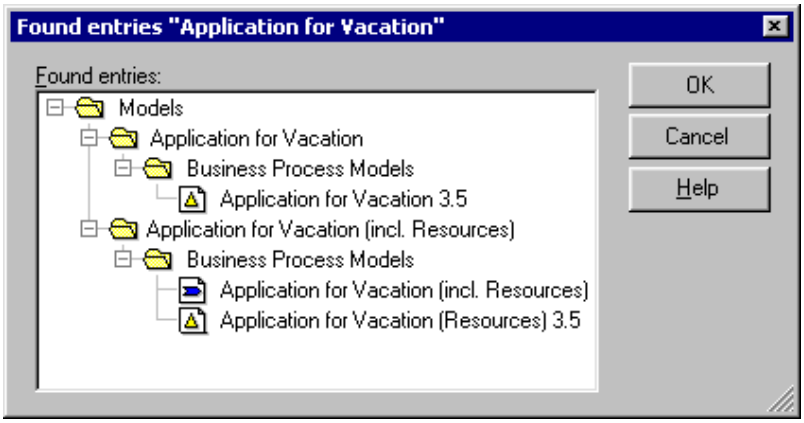


Figure 20: Search results

Select any result in the "Found entries" and then click "OK". The search results window is closed and the selected entries copied into the original selection list. Therefore you can carry out the action (e.g. open model, export models) on the selected search result.

3.1.4 Save as

By selecting "Save as" you can save the contents of the list to a text file. Once this menu item has been selected the "Save as" window is displayed (see fig. 21).

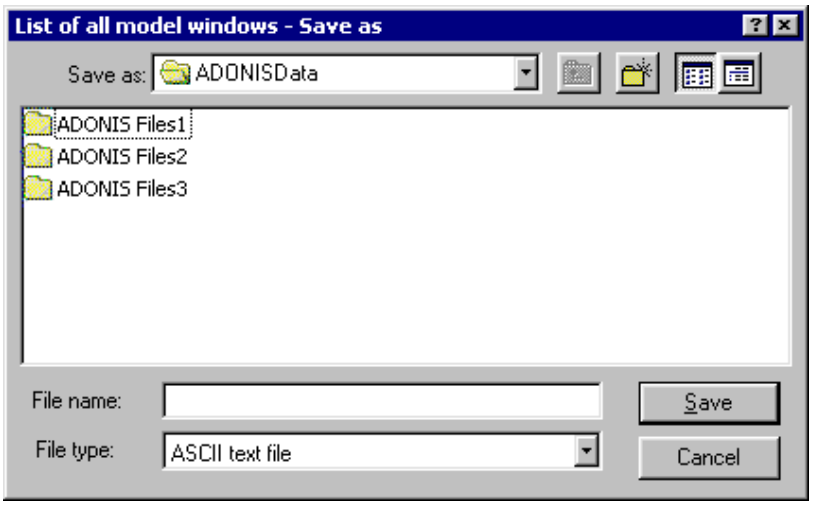


Figure 21: Save as

Enter a file name and select a path where the contents of the list are saved.

3.1.5 Shrink/Expand

Hint: "Expand/Shrink" functions are available only in lists with hierarchical levels

The menu entry "Expand/Shrink" leads to a sub menu, where the hierarchical view in the selection window can be expanded or condensed, the following options are available:

- "Show all"
- "Hide all"
- "Show selected"
- "Show selected with sublevels"
- "Shrink selected"

The options in a nutshell:

Show all:

All hierarchical levels of the list are displayed.

Hot key: **<Ctrl>**+**Enter**

Hide all:

Only the topmost hierarchical levels of the list are shown.

Hot key: **<Ctrl>**+**<Shift>**+**Enter**

Show selected:

This option has the same effect as clicking on the symbol and is only available when at least one list entry is selected. Entries that are immediately below the next hierarchy level will remain collapsed.

Hot key: **+**

Show selected with sublevels:

This function is also only available if at least one entry is selected. All entries on all hierarchical levels below this one will be selected - regardless of structure depth.

Hot key: *****

Shrink selected:

This option has the same effect as clicking on the symbol and is only available when, at least one list entry is selected. All entries below the selected entry are collapsed.

Hot key: **-**

3.1.6 Select All Items

By selecting the "Select all items" menu item or using the hot key **<Crtl>+<A>** all selectable items in the list are selected.

Hint: Only the selectable items of the selection list will be selected, i.e. the items shrunk are not selected. To select all the items of a selection list, first select the "expand all" menu item (see chap. 3.1.5, p. 45) and then the menu item "Select all items".

3.1.7 Deselect

With "Deselect", all selected list entries are deselected.

3.1.8 Selected Items

By selecting the "Selected items" menu item all items selected in the list will be displayed in the window "All selected items" (see fig. 22).

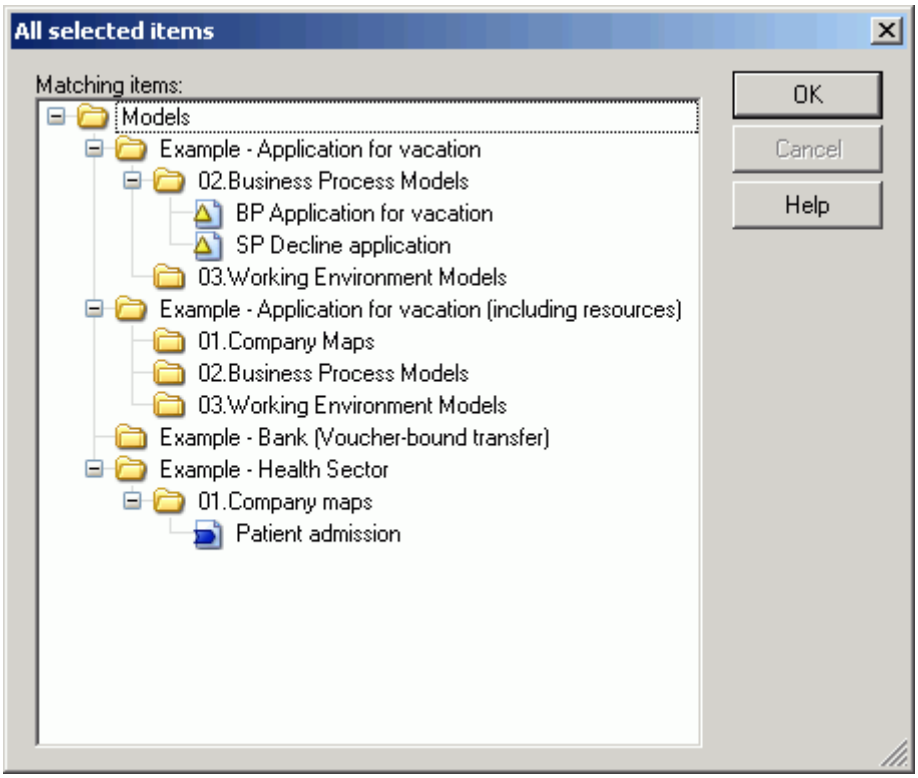


Figure 22: Display selected items

3.1.9 Shrink/Expand Version Threads

Hint: The "shrink/expand version threads" function is only available if a **time-related versioning** (see chap. 10.1.2, p. 92) is defined in the application library.

Expanding version threads is possible by clicking on the icon (right above the model/attribute profile selection list) and causes the displaying of all the versions of a model/attribute profile saved in the ADOxx database (see fig. 23).

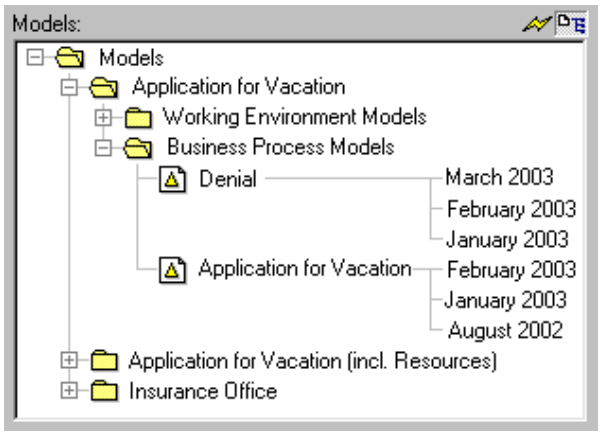


Figure 23: Model selection list with expanded version threads

In case of the hidden version thread, only the latest version of a model/attribute profile will be displayed in the selection list (see fig. 24).

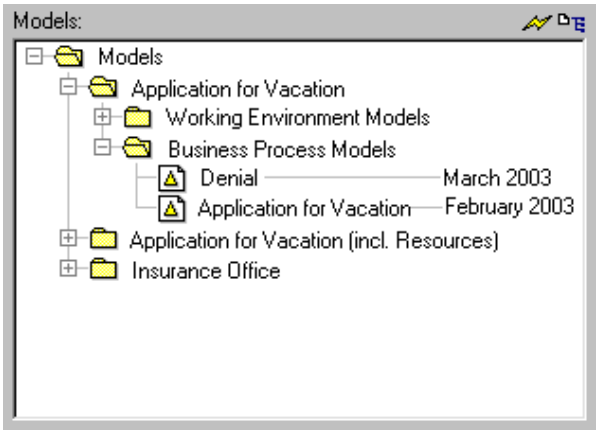


Figure 24: Model selection list with shrunk version threads

3.2 ADOxx Notebook Control Elements

The ADOxx Notebooks in the Modelling Toolkit are used for the attribute management of objects, connectors and models.

Each ADOxx Notebook consists of one or more chapters, which in turn contain one or more pages. Tabs on the right hand side of the Notebook indicate the chapters. If a chapter contains more than one page, then the pages are numbered. By clicking on the tabs, the first page of the selected chapter is displayed. The arrow buttons and in the bottom right hand corner enable you to flip forwards and backwards through each page of the ADOxx Notebook.

The ADOxx Notebooks contain a number of icons, which carry out specific functions and assist in the input of various attribute values. There are also accelerators (underlined letters) in the attribute and chapter names - these in combination with the Alt-key provide another method for navigating through the notebooks.

The ADOxx Notebooks contain the following functions:

- "Save as" icon (see chap. 3.2.1, p. 49)
- "Print" icon (see chap. 3.2.2, p. 49)
- "Info text" icon (see chap. 3.2.3, p. 49)
- "Large text field" icon (see chap. 3.2.4, p. 49)
- "Dialog" icon (see chap. 3.2.5, p. 49)
- "Expression dialog" icon (see chap. 3.2.6, p. 50)
- "Add" icon (see chap. 3.2.7, p. 50)
- "Delete" icon (see chap. 3.2.8, p. 50)
- "Execute/Follow" icon (see chap. 3.2.9, p. 50)
- "File dialogue" icon (see chap. 3.2.10, p. 50)
- "Notebook" icon (see chap. 3.2.11, p. 50)

A Accelerators (see chap. 3.2.13, p. 53)

The "Save" and "Print" icons can be found in the top right hand corner of the ADOxx Notebook.

The "Info" icon can be found on the top right hand corner of the ADOxx Notebook to provide information on the specific class and can also be found over a number of attributes within the Notebook to provide information on the particular attribute.

The "Large text field" icon can be found on the right hand side above attribute fields of this type.

3.2.1 "Save as" Icon

By clicking on the icon the entire contents of the ADOxx Notebook can be saved to a TXT file (see chap. 11.15, p. 757) (ASCII-Text).

3.2.2 "Print" Icon

By clicking on the Print icon the entire contents of the ADOxx Notebook can be printed.

An information window is displayed when the contents have been successfully sent to the printer.

3.2.3 "Info" Icon

When the "Info" icon is selected , a window with additional information on either the class or the particular attribute as appropriate is displayed. The contents of this window can be printed.

Hint: The "Info" icon is only available in the ADOxx Notebooks of objects and connectors.

Hint: The "Info" icon will be displayed only when information on either the class or the particular attribute is defined in the application library.

For the attribute "Name" no "Info" icon will be displayed.

3.2.4 "LargeText Field" Icon

When you click on this icon , a large input field is displayed for the particular attribute. In this way an overview of the input for long attribute values can be taken. The "Large text field" icon is available for all multi-line attributes.

It is also possible to print out the contents of these fields.

3.2.5 "Dialog" Icon

When you click on this icon , a support dialogue for the particular attribute value is displayed. It supports the entering of complex attributes (e.g. performer assignment, statistical distributions for the generation of variables, graphical representation of objects and connectors etc.).

Hint: The "Dialogue" icon is only available in the ADOxx Notebooks of objects and connectors in the Modelling Toolkit as well as in the Class Attribute "GraphRep" within Library Configuration in the Administration Toolkit.

3.2.6 "Expression Dialog" Icon

When you click on this icon , an input window for the expressions (see chap. 3.2.12.3, p. 52) for each attribute value will be displayed. This makes the activity of entering expressions easier.

3.2.7 "Add" Icon

When you click on this icon , a window, in which the attribute values to be added are listed, will be displayed.

Hint: The "Add" icon is only available in the ADOxx Notebook of objects for enumeration lists (see chap. 2.2.12.4, p. 204), creation of inter-model references (see chap. 11., p. 94) and attribute profile references (see chap. 12., p. 100) .

3.2.8 "Delete" Icon

When you click on this icon , a previously selected attribute value will be deleted.

Hint: The "Delete" icon is only available in the ADOxx Notebook of objects for enumeration lists (see chap. 2.2.12.4, p. 204), creation of inter-model references (see chap. 11., p. 94) and attribute profile references (see chap. 12., p. 100) .

3.2.9 "Execute/Follow" Icon

When you click on this icon , a program call will be executed or the referenced model will be shown - depending on the attribute type.

Hint: The "Execute/Follow" icon is only available in the ADOxx Notebook of objects and connectors.

3.2.10 "File Dialog" Icon

If you click on this icon , a window for the input of a file name to call a program is displayed. The file entered will automatically be loaded when carrying out the program call.

Hint: The "File dialogue" icon is only available in the ADOxx Notebook of objects and connectors.

3.2.11 "Notebook" Icon

If you click on this icon , the ADOxx Notebook of the previously selected connector will be displayed and you can edit the attribute of this connector (see chap. 2.2.7.12, p. 146).

Hint: the "notebook" icon is only available in the ADOxx Notebooks of objects, which contain the chapter "relations".

3.2.12 Input Window

3.2.12.1 Input Dialogue for Text

In the input dialogue for text (see fig. 25), texts (e.g. attribute values) are displayed in a field changeable in size and further information and functions are available.

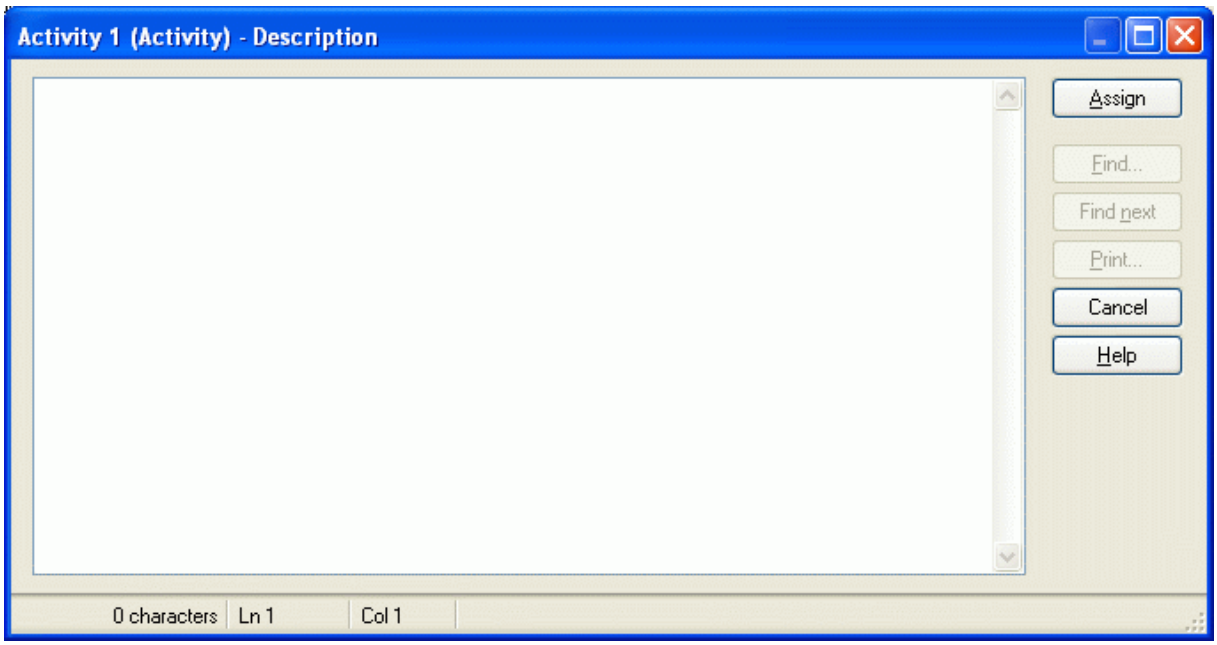


Figure 25: Input dialogue for text attributes

The first field of the status row (at the bottom of the window) shows the number of characters contained in the text. This supports you in entering attribute values, since for example attribute of type "text" (STRING) must not contain more than 3700 characters. In the second field of the status row, the position of the cursor is displayed (row, column).

Once you have edited the text, click **"Assign"** to confirm your entries.

Hint: The "Assign" button is only available if the text displayed can be edited.

By clicking on **"Find"** you can search for entries in the text (see chap. 3.1.3, p. 43). According to the definition of a search text, further entries will be found in the text, if you click **"Find next"**.

You can print (see p. 51) the contents of this input field by clicking on **"Print"**.

Print Text Field / Diagram

Before printing the text, the window "Print text field" is displayed (see fig. 26).

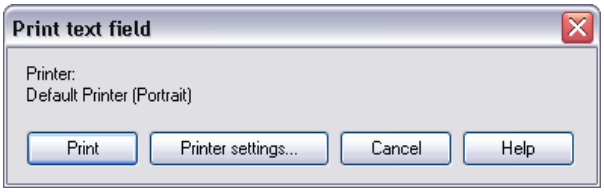


Figure 26: Print text field

Click on the **"Print"** button to send the text to the indicated printer.

Click on **"Printer settings"** to change the settings for the printout.

Hint: If you pront a diagram instead of a text field, the window bears the title "Print diagram". The functionality stays the same.

3.2.12.2 Text Input Field

In the text input field (see fig. 27) a single-line field text input is available.

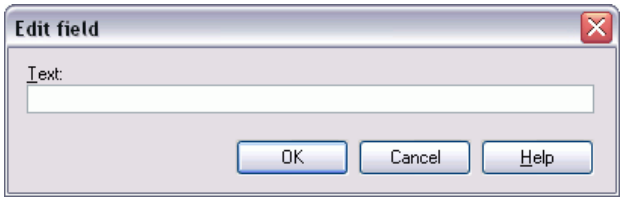


Figure 27: Text input field

Hint: The label of the text input field (window title and field label) as well as some standard text entries are defined in the application library.

3.2.12.3 Input Dialogue for Expressions

A special input dialogue for changeable attributes of the type "Expression" (see chap. 7.4, p. 734) (see fig. 28) is available.

Hint: The changeability status of expression attributes are defined in the library.

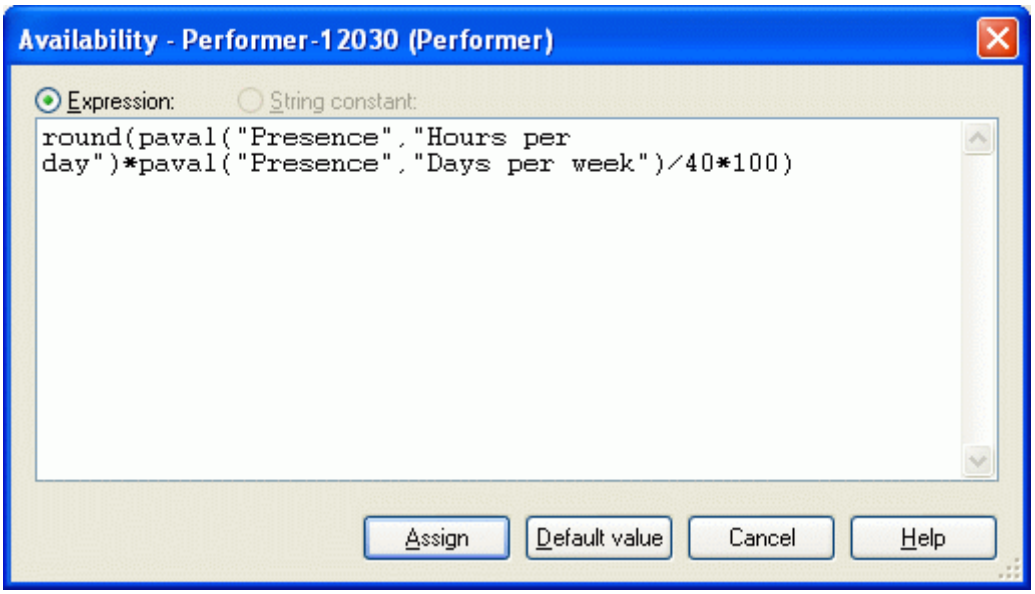


Figure 28: Input dialogue for expressions

You can enter either an expression (see chap. 12., p. 758) or a string. Click on the **"Standard value"** button to enter the standard value defined in the application library.

Click **"Assign"** to confirm your entry.

Hint: The entered value must not contain more than 3600 characters.

3.2.13 Accelerators

The accelerators in the ADOxx Notebooks support you in navigating and entering attribute values.

The accelerators are identified by underlined letters and are contained in all chapter names and attribute names (see fig. 29).

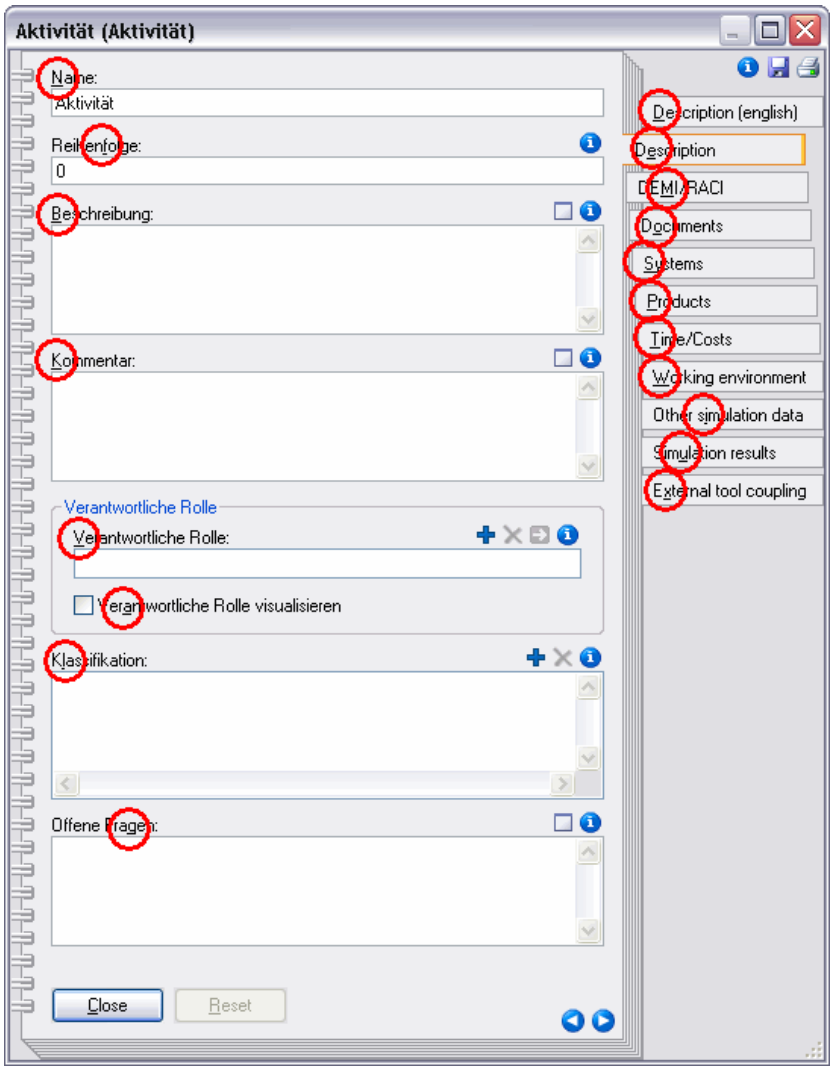


Figure 29: Accelerators in a ADOxx Notebook

Navigation using accelerators is carried out using a hot key combination of **<Alt>** and the desired underlined letter. On entering the accelerator of a chapter, the first page of the chapter is displayed. By entering the accelerator of an attribute name, the cursor is placed in the relevant attribute field so that you can input a value or edit the existing value.

For turning the pages of the Notebook, you can use either the arrow buttons in the bottom right corner of the Notebook or the key combination **<Alt>**+**<+>**for the next page and**<Alt>**+**<->** for the previous page.

Hint: In Windows environments it is possible to hide accelerators by default. However, as soon as you press the **<Alt>** key they become visible.

4. ADOxx Explorer

The Explorer is a window, that displays ADOxx models and their status information (see fig. 30). It has its own quick-access bar, which provides fast access to the functions and views. Additional functions are available via context menu.

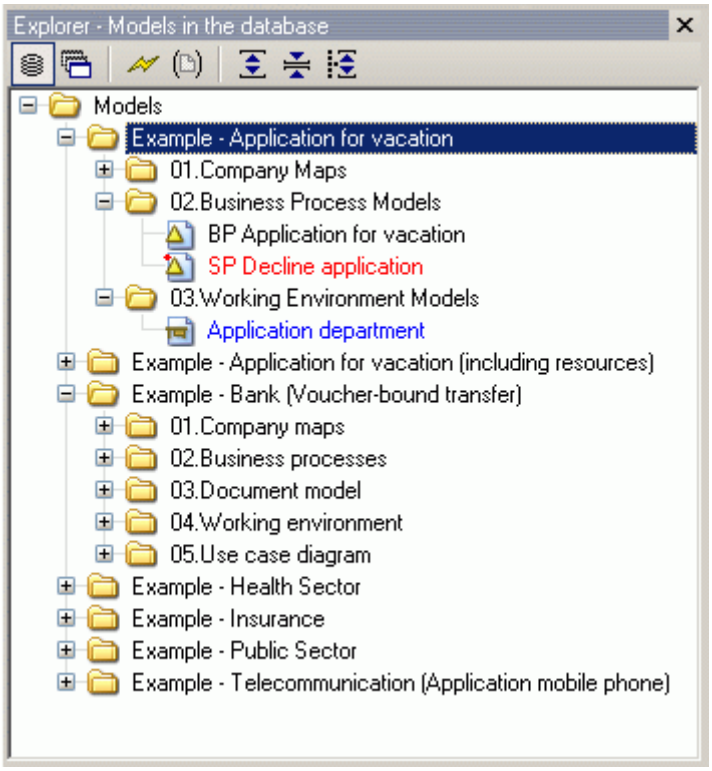


Figure 30: The ADOxx Explorer

The Explorer can be made visible or hidden by pressing **<Crtl>+E** or selecting the option **"Explorer"** in the menu **"Window" -> "Tools"**. It can be moved and docked to any edge or can float in any position. Using the <Tab> key, a user can change the focus between the Explorer and the active model window.

Text colour in the Explorer:

- **Black** text displays model group names and models that are closed.
- **Blue** text displays models that are currently open and have no unsaved changes.
- **Red** text displays models that are open and contain changes that have not been saved.

The Explorer smart-icons and their functions:

"Models in the database" (see chap. 4.1, p. 56)

"Opened model windows" (see chap. 4.2, p. 57)

"Refresh view" (see chap. 4.3, p. 57)

"Options for the opening of models" (see chap. 4.4, p. 57)

"Set visible model types" (see chap. 4.5, p. 58)

"Show all" (see chap. 4.6, p. 58)

"Hide all" (see chap. 4.6, p. 58)

"Show selected with sublevels" (see chap. 4.6, p. 58)

The Explorer's context menu (view "Models in the database") provides the following functions:

New	to create a new model or model group (see chap. 4.7, p. 58)
Open / Activate	to load selected models from the database or bring a loaded model to the top (see chap. 4.8, p. 59)
Rename	to rename selected models or model groups (see chap. 4.11, p. 59)
Cut	to cut selected model or model groups (see chap. 4.13, p. 60)
Copy	to copy selected model or model groups (see chap. 4.13, p. 60)
Paste	to paste selected model or model groups (see chap. 4.13, p. 60)
Delete	to delete selected model or model groups (see chap. 4.13, p. 60)
Model references	to copy or delete selected model references (see chap. 4.14, p. 60)
Model types	to filter the visible model types (see chap. 4.5, p. 58)
Manage model groups	to manage the model group structure (see chap. 2.4.12, p. 269)
Refresh	to update the Explorer view (see chap. 4.3, p. 57)
Item search	to search for entries in the Explorer (see chap. 4.16, p. 61)
Save as	to save the current structure to a text file (see chap. 4.17, p. 61)
Shrink/Expand	to show or hide branches and selected levels (see chap. 4.6, p. 58)
Select all items	to select all models and model groups in the current view (see chap. 3.1.6, p. 46)
Deselect	to cancel all selections (see chap. 3.1.7, p. 46)
Selected items	to display selected models and model groups clearly laid out (see chap. 3.1.8, p. 46)

The Explorer's context menu (view "Opened model windows") provides the following functions:

Activate	to bring a loaded model to the top (see chap. 4.8, p. 59)
Close	to close selected models (see chap. 4.9, p. 59)
Save	to save selected models (see chap. 4.10, p. 59)
Rename	to rename selected models or model groups (see chap. 4.11, p. 59)
Print	to print selected models (see chap. 4.12, p. 60)
Refresh	to update the Explorer view (see chap. 4.3, p. 57)
Item search	to search for entries in the Explorer (see chap. 4.16, p. 61)

© BOC Group 2008

Save as	to save the current structure to a text file (see chap. 4.17, p. 61)
Shrink/Expand	to show or hide branches and selected levels (see chap. 4.6, p. 58)
Select all items	to select all models and model groups in the current view (see chap. 3.1.6, p. 46)
Deselect	to cancel all selections (see chap. 3.1.7, p. 46)
Selected items	to display selected models and model groups clearly laid out (see chap. 3.1.8, p. 46)

4.1 Models in the Database

Clicking on the icon or selecting menu **"Window"** -> **"Navigation"** -> **"Models in the database"** or pressing the keys **<Crtl>+D** causes the explorer to show the model group structure including all models from the database. The active model from the workspace is preselected:

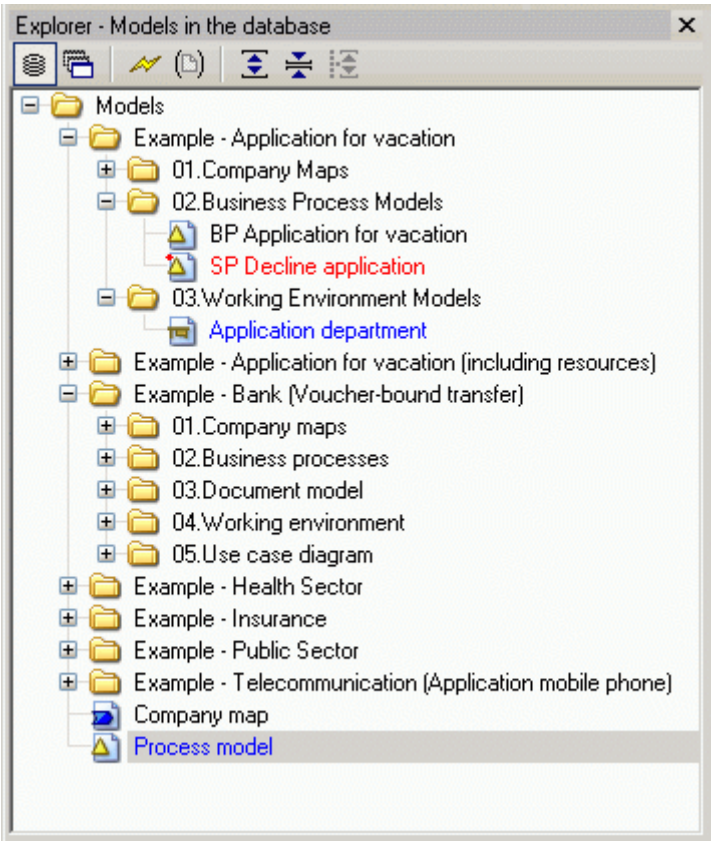


Figure 31: View option: model stock

Hint: By using these menu commands or hot keys, a potentially hidden explorer will be activated.

4.2 Opened Model Windows

Clicking on the icon or selecting menu **"Window"** -> **"Navigation"** -> **"Opened models"** or pressing keys **<Crtl>+L** causes the Explorer to show the currently open models sorted by model type. The active model from the workspace is preselected:

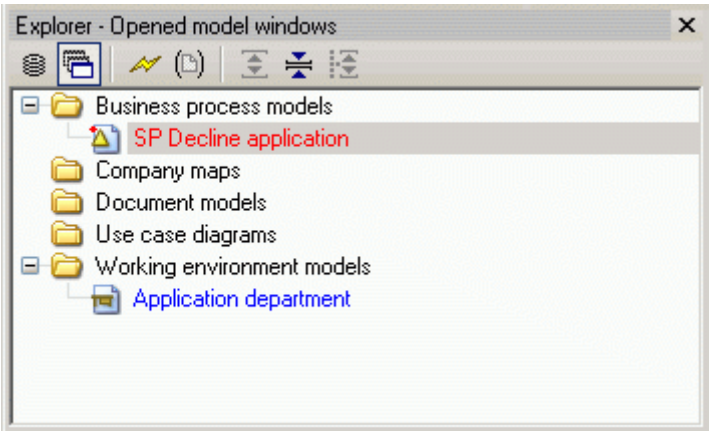


Figure 32: View option: Opened model windows

Hint: By using the menu commands or hot keys of these two functions, a potentially hidden explorer will be activated.

4.3 Refresh

The icon (or *context menu* -> **"Refresh"**) is used to make a connection to a ADOxx database to display the newest status there. Regular updates are suggested especially in the distributed environments, where many users can access the same model stock at the same time. The active model from the workspace is preselected after refreshing.

4.4 Options for the Opening of Models

After clicking on the icon a dropdown menu appears, offering various model opening options. To the left of the menu entry a tick (✔) indicates that the function is currently active. These variants are available:

Include referenced models
In addition to the selected models, all models referenced from these are opened too.

Open write-protected
The models are opened without write access.

Minimise new windows
The opened models are not displayed in the workspace but placed there as minimised windows.

Hint: These three options can be combined freely.

References
The support dialogue References - Settings (see chap. 11.1, p. 97) is opened.

4.5 Set Visible Model Types

For both views "Models in the database (see chap. 4.1, p. 56)" and "Opened model windows (see chap. 4.2, p. 57)" it is possible to filter the displaying of given model types. The icon (or *context menu* -> menu entry **"Modeltypes"**) opens a dialogue, where the user can select preferred model types to be displayed:

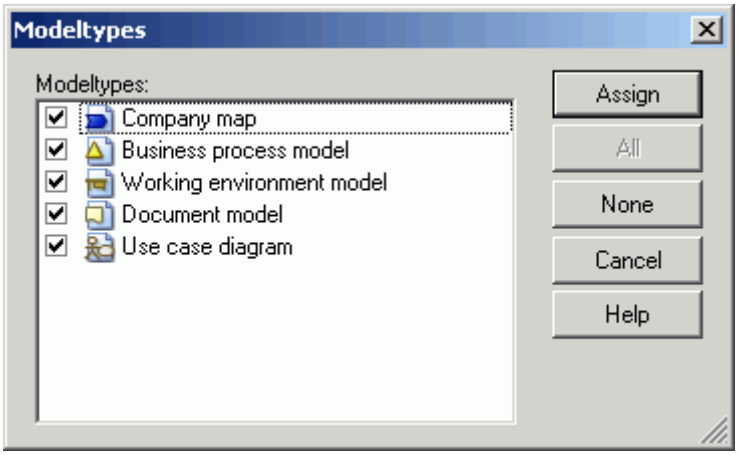


Figure 33: Showing or hiding model types

By clicking on the checkboxes next to the model types, the user can choose which type to show ( ) or hide ( ). After clicking **"Assign"**, the view is adjusted accordingly.

> **Hint:** If your application library contains a special model type filter (see chap. 3.1.1.1, p. 42), it will be shown instead of the model type selection dialogue.

4.6 Show / Hide Branches (Levels)

In ADOxx models are displayed and managed in a hierarchical tree structure. As in all such structures, clicking on or respectively, expands or shrinks the desired branch or level.

On clicking on the icon the complete tree structure will be expanded.

After clicking on the icon , the view "Models in the database" is collapsed to the main model groups or the view "Opened model windows" is reduced to a model type list.

If you selected collapsed model group, you can expand the complete substructure by clicking .

> **Hint:** All above functions are located in the Explorer's **context menu** under **"Expand / Shrink"** too.

4.7 Create New Models / New Model Groups

A new model group can be created in the Explorer immediately. Select the model group where the new model or the new model group should be created and choose the menu entry **"New"** from the *context menu* and the desired sub menu entry.

A new model group (sub menu entry "Model group") will be named "New model group". If you chose a model type from the sub menu the new model of this type will be named "New model" and will be displayed on the workspace.

Hint: Modells and model groups can be renamed (see chap. 4.11, p. 59) at any time.

Hint: The functions "Create new models" and "Create new model groups" are available in the view "Models in the database" (see chap. 4.1, p. 56) only.

4.8 Open / Activate Models

A selected **model** can be opened by double-clicking it, selecting **"open"** from the *context menu* or by pressing the **"Enter"** key. The name of the opened model is displayed in blue in the ADOxx Explorer.

An already open model can be "activated" (brought to the foreground in the workspace) by double-clicking the model name, choosing **"Activate"** from the *context menu* or hitting the **"Enter"** key.

Hint: During opening, the Options for the Opening of Models (see chap. 4.3, p. 57) are taken into account.

Hint: The function "Open models" is available in the view "Models in the database" (see chap. 4.1, p. 56) only.

4.9 Close Models

A selected **model** can be closed (see chap. 2.4.3, p. 259) by selecting **"Close"** from the *context menu*.

Hint: The function "Close models" is available in the view "Opened model windows" (see chap. 4.2, p. 57) only.

4.10 Save Models

A selected **model** can be saved (see chap. 2.4.4, p. 259) by selecting **"Save"** from the *context menu*.

Hint: The function "Save models" is available in the view "Opened model windows" (see chap. 4.2, p. 57) only.

4.11 Rename Models / Model Groups

A selected **model** can be renamed (see chap. 2.4.8, p. 262) by selecting **"rename"** from the *context menu* or by pressing **<F2>**.

A selected **model group** can be renamed (see chap. 2.4.12.2, p. 270) by selecting **"rename"** from the *context menu* or by pressing **<F2>**.

Hint: The function "Rename model groups" is available in the view "Models in the database" (see chap. 4.1, p. 56) only.

4.12 Print Models

A selected **model** can be printed (see chap. 2.4.17, p. 288) by selecting **"Print"** from the *context menu*.

Selecting several models and selecting **"Print"** from the *context menu* afterwards enables printing of these models with identical settings (see chap. 2.4.17.2, p. 290).

> **Hint:** The function "Print models" is available in the view "Opened model windows" (see chap. 4.2, p. 57) only.

4.13 Cut / Copy / Paste / Delete Models and Model Groups

Like in almost all modern programs containing treelists, the Explorer offers the widely-known features **"Cut"** (**<Ctrl>**+**X**), **"Copy"** (**<Ctrl>**+**C**) und **"Paste"** (**<Ctrl>**+**V**) for manipulating models and model groups. These commands can also be found in the context menu.

A selected **model** can be deleted (see chap. 2.4.15, p. 283) by selecting "delete" from the context menu or by pressing the key**<Del>**.

A selected **model group** can be deleted (see chap. 2.4.12.4, p. 271) by selecting "delete" from the context menu or by pressing the key**<Del>**.

> **Hint:** Model groups can only be deleted when they are totally empty.

> **Hint:** The functions "Cut, "Copy", "Paste" and "Delete" are available in the view "Models in the database" (see chap. 4.1, p. 56) only.

4.14 Copy / Delete Model References

Strictly speaking, the ADOxx Explorer does not show models, but *model references* (references to the real model in the database). For most of the Explorer features (open, rename, etc.) this difference to classic file systems is extraneous. Nevertheless, the model references principle has a big advantage: By copying model references into different model groups, the same model can be accessed from various places in the model hierarchy. This enables efficient work even in structures with very strict access restrictions.

With the function **"copy model reference"** (*contex menu* - entry "Model reference(s)" - subentry "Copy") the selected model references are copied to the ADOxx clipboard. Afterwards, they can be inserted into another model group (*context menu* - entry "Paste" or keys **<Ctrl>**+**V**).

The function **"delete model reference"** (*contex menu* - entry "Model reference(s)" - subentry "Delete") removes the selected references from their model groups. The models themselves remain untouched.

Difference Copy model - Copy model reference:

- When you copy a *model* and paste it elswhere into the treelist, a new model is created that, while containig the same objects and connectors, bears a different name than the copied model. The number of models in the database increases.
- When you copy a *model reference* and paste it elswhere into the treelist, another reference to an existing model is created, bearing the same name than the copied reference. The number of models in the database remains the same.

ATTENTION: The last model reference of a model cannot be deleted, in order to prevent models being moved to the model pool unintentionally. Of course, the model itself can be deleted as ever; this naturally removes the model reference too. (Note: The Pool contains all models not accessible via model groups. They can be moved to the model group structure by the ADOxx administrator.)

Hint: The functions "Copy model reference" and "Delete model reference" are available in the view "Models in the database" (see chap. 4.1, p. 56) only.

4.15 Manage model groups

Via the *context menu* - entry **"Manage model groups"**, the "Model group management" (see chap. 2.4.12, p. 269) is started.

Hint: The function "Manage model groups" is available in the view "Models in the database" (see chap. 4.1, p. 56) only.

4.16 Item Search

Via the *context menu* - entry **"Item Search"**, the general ADOxx search function is started, in order to search for text patterns in the Explorer.

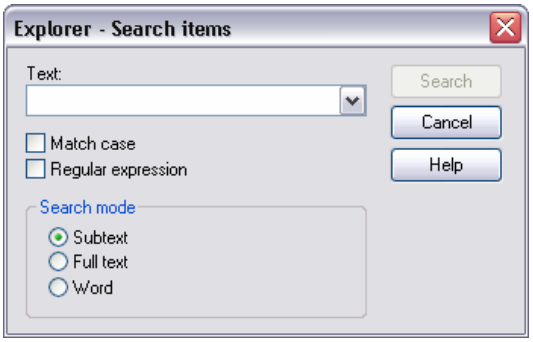


Figure 34: An Example: Looking for a model containing the string "application" in the name

The following options are available:

- Match case: capitals and small letters will be distinguished, while finding the given term.
- Regular expression: searched entry will be interpreted as a Regular Expression, not a string.
- Subtext: a searched term can be found in the text, that does not have to be a whole word.
- Full text: only the results that exactly match the search string will be taken into consideration.
- Word: A searched term must be a word. The "word" is defined as a string of arbitrary length, flanked by blanks, a full-stop or the like.

4.17 Save Tree Structure

To save the tree structure of the Explorer to a file, select **"Save as"** from the *context menu*. In the appearing window, enter the desired path and filename for the export file. The data is stored in ASCII text format.

5. ADOxx Navigator

The Naviagtor is a window showing an overview of the active model (see fig. 35). The part of the model which is currently visible in the workspace is displayed with white background, the "invisible" part has a grey background in the Naviagtor window.

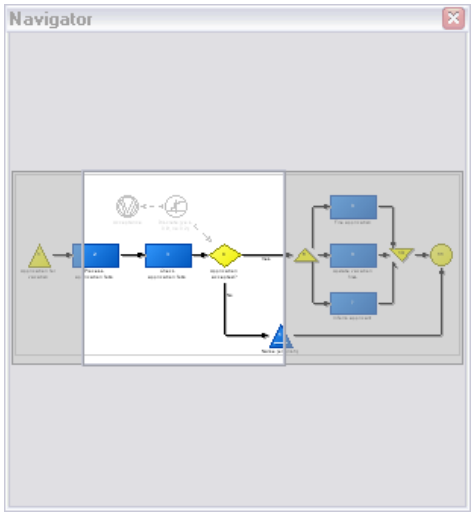


Figure 35: The ADOxx Navigator

The Navigator can be made visible or hidden by pressing **<Crtl>+R** or selecting the option **"Navigator"** in the menu **"Window" -> "Tools"**. It can be moved and docked to any edge or can float in any position.

The Navigator offers these functions:

Move visible area	to move the area visible in the workspace to a different position
Change visible area	to jump to a different part of the model and display it in the workspace
Zoom	to resize the visible area

Move visible area:

The visible part of the model can be moved using the Naviagtor. For this, click into the white-back area of the model overview (the mouse pointer changes to) and drag it to the desired site in the model. During moving, the view in the workspace is adjusted permanently.

Change visible area:

With this function, far-away parts of a model can be reached with one click. Click at an arbitrary grey-back point in the model preview. The visible area both in the Naviagtor and the workspace jump to the selected position (with the click point as the centre).

Zoom:

In addition to the extensive zoom function (see chap. 2.2.11.3, p. 189), the Navigator offers a possibility to resize the visible area of the model. For this, move the mouse pointer to a border of the visible area in the Navigator (the mouse pointer changes to or respectively).Drag the border into the desired direction. On releasing the mouse button, the new zoom factor is calculated and the model views in Navigator and workspace are adjusted accordingly.

6. ADOxx Inspector

The Inspector is a window which informs the user about various details of the current modelling environment (see fig. 36). The information is clearly arranged as a treelist.

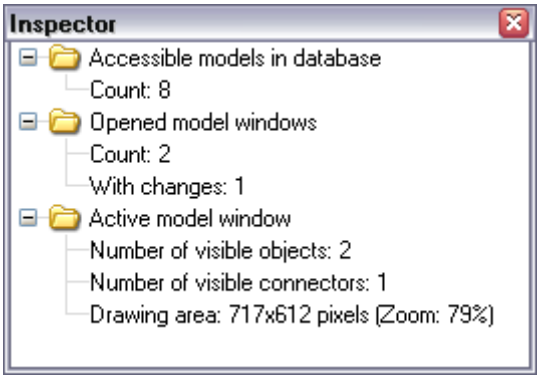


Figure 36: The ADOxx Inspector

The Inspector can be made visible or hidden by pressing **<Crtl>+I** or selecting the option **"Inspector"** in the menu **"Window" -> "Tools"**. It can be moved and docked to any edge or can float in any position.

Like in all tree structures, by clicking on ⊞ or ⊟ single hierarchy levels can be expanded or collapsed repectively. Furthermore, some hot keys are available: **<Ctrl>+<Enter>** shows all hierarchy levels, **<Ctrl>+<Shift>+<Enter>** hides all but the topmost hierarchy levels, **"+"** expands the selected branch, **"-"** hides the selected branch, and **"*"** expands the selected level plus all sub levels.

The Inspector offers the following details:

Accessible models in database Shows the number of models in the ADOxx database which the active user may acces.

Opened model windows Shows the number of models the active user currently holds open plus how many of them contain unsaved changes.

Active model window Lists the number of visible objects and connectors from the active model. Furthermore, the size of the drawing space at the current zoom factor is shown (in pixels).

7. ADOxx Serach Result Window

The ADOxx search result window is a tool which gives both access to the search function (see chap. 2.2.7.19, p. 152) and all model content search results of the current session. It has its own quick-access bar with special smart-icons (see fig. 37).

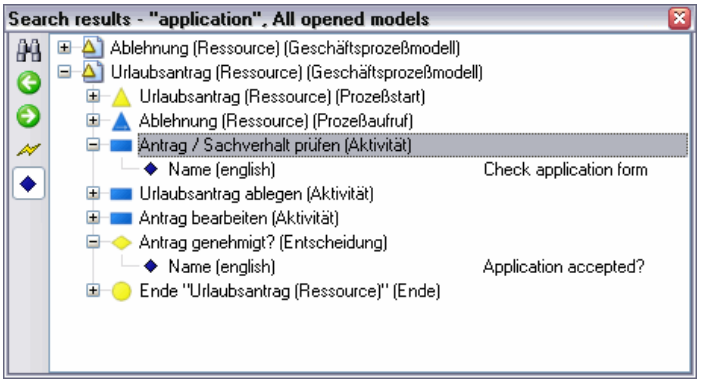


Figure 37: Search results

The search result window can be made visible or hidden via the menu **"Window"** - entry **"Tools"** - **"Search results"** or the hot key **<Ctrl>+<Shift>+F**. It can be moved and docked to any edge or can float in any position.

After every search, the result is displayed as a **treelist** in this window. The visualised hierarchy is: Model(s) - Object(s) - Attribute(s) with their values. The title bar of the search result window shows the active search with the most important search parameters.

Click on a **model** in the search result window to call and display this model in the workspace. Click on an **object** or **attribute** to call and display its model in the workspace with the appropriate object preselected and centered in the middle of the visible area.

You can of course **modify attribute values** in the search results: Double-click the desired attribute in the search result window to open an input support dialogue.

Click on the Icon to perform **another search**. The search function (see chap. 2.2.7.19, p. 152) is called.

A continuous **history** allows navigating through all model search results. Click on the Icon to jump to the previous or to jump to the next search result.

If you want, you can **update a serch result**. For this, click on the icon in the search result window. The currently visible search is repeated and the result window updated accordingly.

You can make the search result **more clearly arranged**. Click the toggle icon . The attributes are removed from the view and only two hierarchy levels (models and objects) remain visible. Of course, in this view no attribute values can be modified. Click once more to show the attributes again.

8. ADOxx Browser

A browser included in ADOxx provides an overview, work can be distributed and further work (printing, saving...) can be carried out. Regardless of where you are in ADOxx, the ADOxx browser provides a standard layout.

The ADOxx browser is used in the modelling Component (see chap. 2., p. 126), Analysis Component (see chap. 3., p. 306), Simulation Component (see chap. 4., p. 332) and the Evaluation Component (see chap. 5., p. 433).

8.1 Classification

In ADOxx different types of browsers are available. They can be divided into:

- **hierarchical** and
- **non hierarchical** browsers

and depending whether the information can be changed into

- **editable** and
- **non editable** browsers.

The information in an **editable browser** (see chap. 8.2.1, p. 69) can be edited and saved while the information in a non editable browser will only be used to show results.

The information in a **hierarchical browser** (see chap. 8.2.2, p. 70) is shown in different levels (e.g. models and sub models) while information in a non hierarchical browser is shown in a one-dimensional structure.

Both groups of browsers (hierarchical and editable) can be combined in the actual ADOxx browser depending on the information shown.

Examples:

- editable/hierarchical:

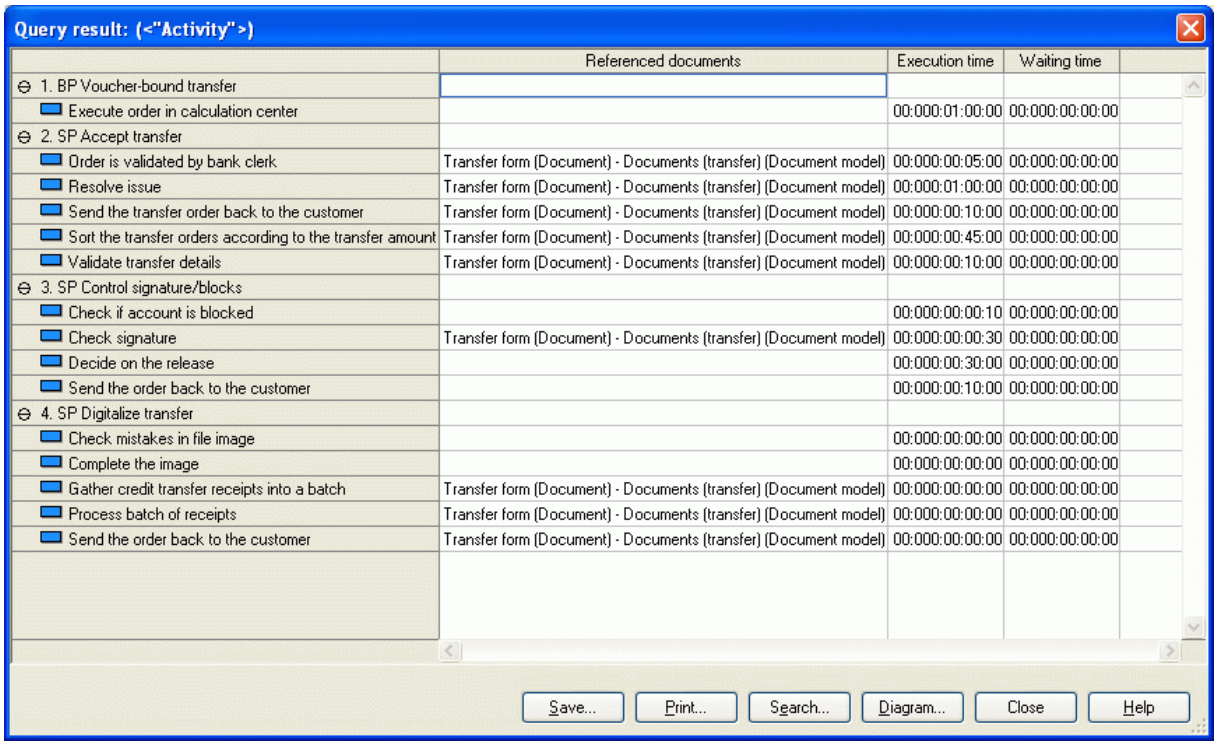


Figure 38: Example of an editable, hierarchical browser (analysis result)

- editable/non hierarchical:

	Execution time	Waiting time	Costs
Order is validated by bank clerk	00:000:00:05:00	00:000:00:02:00	4,00
Validate transfer details	00:000:00:10:00	00:000:00:00:40	22,00
Resolve issue	00:000:01:00:00	00:000:00:10:00	7,00
Sort the transfer orders according to the transfer amount	00:000:00:45:00	00:000:00:41:00	6,00
Send the transfer order back to the customer	00:000:00:10:00	00:000:00:00:03	9,00

Figure 39: Example of an editable, non hierarchical browser (tabular model representation)

- non editable/hierarchical:

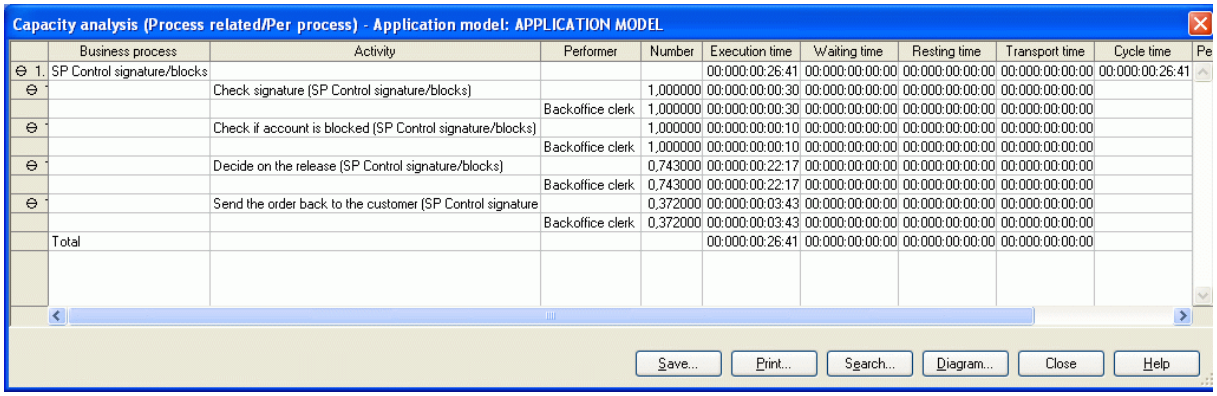


Figure 40: Example of a non editable, hierarchical browser (simulation result capacity analysis)

- non editable/non hierarchical:

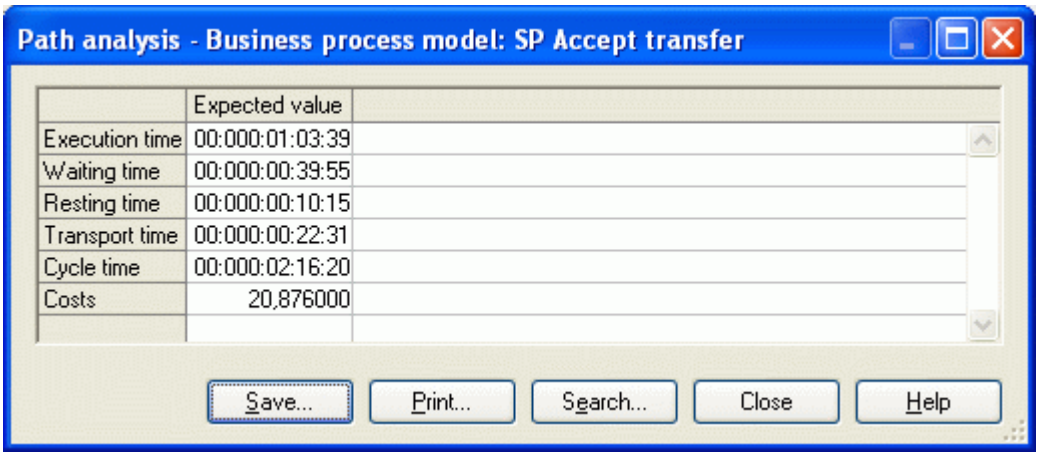


Figure 41: Example of a non editable, non hierarchical browser (simulation result path analysis)

8.2 Structure

The window of the ADOxx browser displays data (results) in tables and offers the functionality to save, print and display the data graphically.

The column headings (first row, grey background) show the type of data in each column (e.g. "attribute"). The first **column** (grey) contains the structure of the rows or the row names for each of the data sections (lines).

The **rows** group the results in individual data sections. Each data section contains one or more cells.

Within the **cells** the corresponding attribute values or calculation results are displayed.

Hint: In hierarchical ADOxx browsers (see chap. 8.2.2, p. 70) empty cells will also be shown.

The picture below (see fig. 42) shows the results of a Capacity Analysis in the ADOxx browser. Some information about the results is also shown.

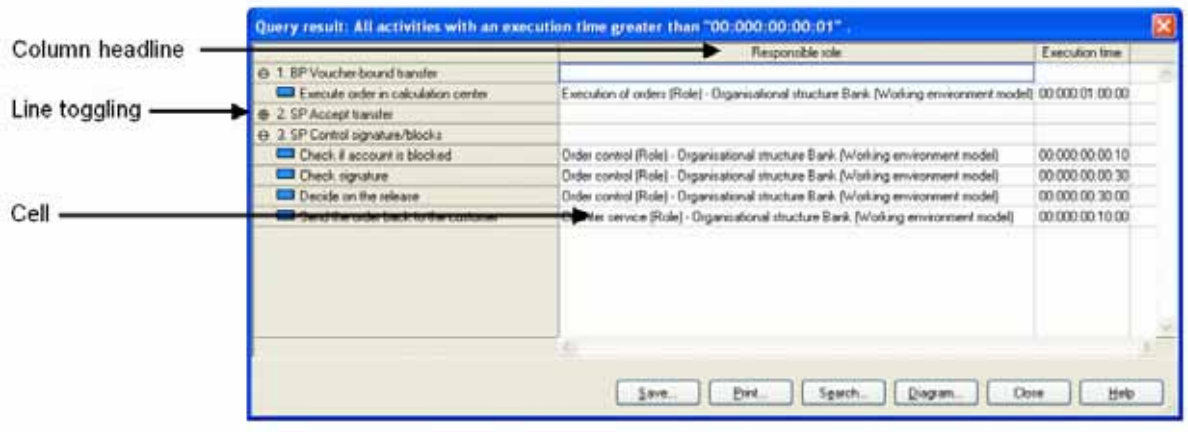


Figure 42: ADOxx browser structure

In a **hierarchical ADOxx browser** (see chap. 8.2.2, p. 70) a number indicating the level will be shown in the (grey) header of each row indicating the hierarchy level of the current row.

The display area (white background) contains the results.

The structure of the browser in terms of the rows and columns labels and contents varies according to the way results are displayed.

In an **editable ADOxx browser** (see chap. 8.2.1, p. 69) you can change the values displayed in the cells. The availability of this function depends on the one hand on the data displayed and on the other hand on the write access to models from which the data is taken from. For example, you can edit the results of an Analysis query directly in the ADOxx browser, if you have previously opened the models for the query with write access.

Hint: The active (current) cell or the selected area will be indicated by a frame using the colour for the frame as defined in the system settings.

Hint: Write protected attributes in editable ADOxx browsers will be shown in grey font.

The **context menu** (right mouse button) in the ADOxx browser contains among other things the following functions:

- **Expand all (see chap. 8.7, p. 76)**: show the entire contents of the browser by expanding all areas,
- **Shrink all (see chap. 8.7, p. 76)**: only shows the highest level,
- **Save (see chap. 8.13, p. 81)**: save the contents of the browser in a file,
- **Print (see chap. 8.15, p. 84)**: print the contents of the browser as currently displayed (WYSIWYG),
- **Find (see chap. 8.12, p. 80)**: search for specific text within the browser,
- **Copy to clipboard (see chap. 8.14, p. 84)**: copy the contents of the browser to the clipboard,
- **Select attributes/columns (see chap. 8.8, p. 77)**: select the columns, which have to be shown (attributes),
- **Sort (see chap. 8.9.1, p. 78)**: sort the browser contents,
- **Column width (see chap. 8.3, p. 74)**: enter the column width,
- **Adjust column width (see chap. 8.4, p. 75)**: adjust the width of the columns to the optimum size,
- **Row height (see chap. 8.5, p. 75)**: enter the row height.
- **Adjust row height (see chap. 8.6, p. 76)**: adjust the height of the rows to the optimum size.

Hint: The functions "Expand all (see chap. 8.7, p. 76)" and "Shrink all (see chap. 8.7, p. 76)" are only available in the hierarchical ADOxx browser (see chap. 8.2.2, p. 70) .

The **context menu** (right mouse click) **for the column header** in general contains the following functions:

- **align attribute values** (see chap. 8.10, p. 79): align the attribute value of the current column to the left, centre it or to the right,
- **sort ascending/descending** (see chap. 8.10, p. 79): ascending or descending sorting of the attribute values of the current column.

8.2.1 Editable ADOxx Browser

The editing of the displayed value in an editable ADOxx browser is possible by:

- directly entering a value in any cell,
- copying and inserting values inside the browser,
- inserting values which have been copied onto the clipboard, for instance from a table calculation program.

For simple attribute types (figures, text) simply enter an attribute value **directly into the appropriate cell**, by clicking on the cell or selecting the cell using the cursor keys. Type in the new content and confirm with the "Enter" key.

Complex attributes (enumerations, expressions etc.) need an input support dialogue (see chap. 2.2.12, p. 201). In this case, double-click the cell or select it and press the "Enter" key to call the support dialogue.

If you want to **edit several (related) values of the same attribute** (i.e. in the same column), hold down the <Shift> key, click on the cell with the first (highest) value (the mouse pointer changes to) and mark - with another mouse click - the domain of the value to edit. Press the <Enter> key to start the input. If you press the <Enter> key again, you will finish the entry of the current value and start the entry of the next value.

> **Hint:** You can abort the editing of several values at any time by pressing the <Esc> key.

Copy an attribute value, by clicking the cell with the value to copy or select it using the cursor keys and then select the "Copy" menu item in the context menu (right mouse button) or press the key combination <Ctrl>+<C>.

Paste the copied attribute value, by clicking on the cell in which the value should be inserted or select it using the cursor keys and then select the "Paste" menu item in the context menu (right mouse button) or press the key combination <Ctrl>+<V>

> **Hint:** The copy and paste function is also possible for several values (areas) at the same time, by selecting the appropriate areas with the mouse and then carrying out the functions "Copy" and "Paste".

If you want to **copy the value of a cell into several (related) cells** of the same attribute (i.e. in the same column), press and hold the <Ctrl> key, then click on the cell with the value to copy (the mouse pointer will change to) and select the target area - by pressing the mouse button again for each cell while still holding the <Ctrl> key- .

Additionally, it is also possible to insert cell values **from another application** (e.g. Microsoft Excel) into ADOxx browser .

> **Hint:** When copying and pasting values, note that the type of attributes must match with each other, i.e. the type of attributes of the copied value must correspond to the type of attribute in which you want to insert the value.
>
> **Exception:** In attributes of the type "String" or "Longstring", you can insert the values of other types of attributes.

8.2.2 Hierarchical ADOxx Browser

The hierarchical ADOxx browser enables the representation of hierarchical structures (tree structures), making it possible to shrink and expand each tree (see fig. 43). Furthermore a number indicating the level will be shown in the (grey) header of each row indicating the hierarchy level of the current row.

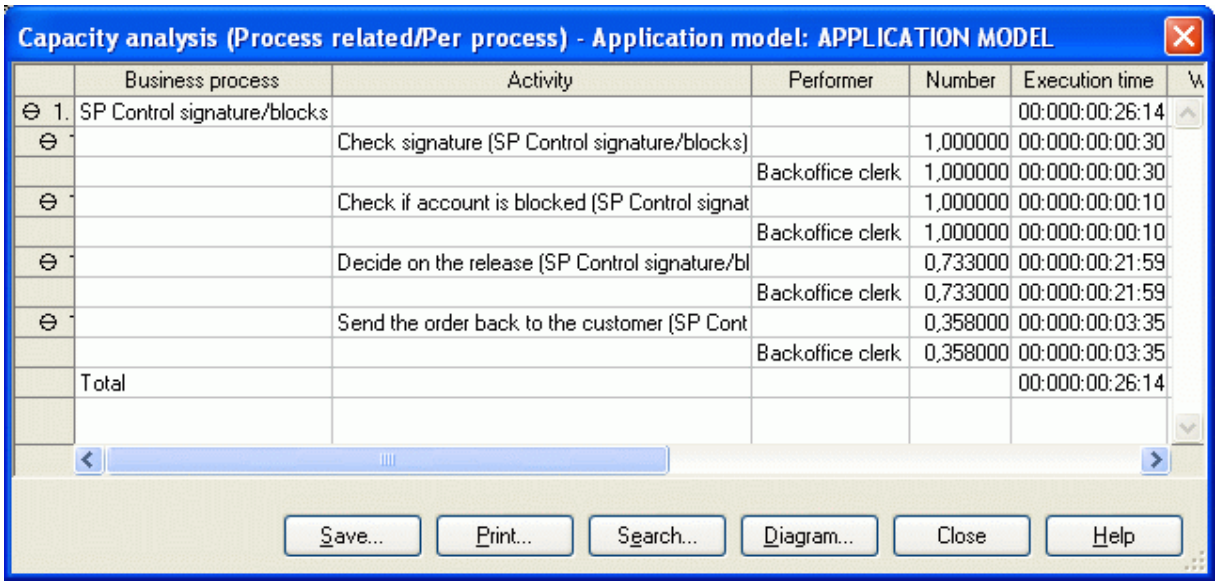


Figure 43: Hierarchical ADOxx browser (Example simulation result)

Before each section either ⊕ or ⊖ is displayed in the first column. This icon ⊕ indicates that the current row contains a summary of results and can be broken down further by clicking on the symbol. This symbol ⊖ means that all information is displayed. By clicking on this symbol only summary information is displayed.

Hint: Lines without a symbol represent the lowest hierarchy (tree) and therefore have no other subordinated data.

Hint: Due to the hierarchical representation the ADOxx browser will also show empty cells.

8.2.3 Representation of Attribute Values

The attribute values shown in a ADOxx browser are shown as text whereas the representation of the following attribute types differs from the normal representation in the ADOxx Notebook:

Records A record in the tabular modelling is represented by the word "`[Record]`" and in the analysis results represented by the symbol ⊕

Program call The representation of a program call attribute (see chap. 8.2.3.1, p. 72) depends on the definition of your application library.

Reference A reference to a **model** is represented by "<*model name*> (<*model type*>)" whereas multiple references within one attribute are shown using several lines.

A reference to an **object** within a model is represented by "<*object name*> (<*class name*>) - <*model name*> (<*model type*>)" whereas multiple references within one attribute are shown using several lines.

Enumeration list The enumeration values selected from the list will be displayed separated by ";".

Calendar A process or performer calendar is represented by the word "`[Calendar]`".

To **edit the displayed attribute values** click on the respective cell. Depending on the attribute type the following input support dialogues are available:

Edit record attributes (see chap. 2.2.12.1, p. 202)
To edit attributes of type "Record" (RECORD).

Colour definition (see chap. 2.2.12.2, p. 203)
To select a colour (for the graphical representation of an object).

Choose enumeration value (see chap. 2.2.12.3, p. 204)
To choose an attribute value (in the tabular representation).

Selection enumeration value from list (see chap. 2.2.12.4, p. 204)
To select an attribute value from an enumeration list.

Edit external program call (see chap. 2.2.12.5, p. 205)
To edit an external program call (in the tabular representation).

Edit date (see chap. 2.2.12.6, p. 205)
To edit attributes of type "Date" (DATE).

Edit date/time (see chap. 2.2.12.6, p. 205)
To edit attributes of type "Date and time" (DATETIME).

Edit time (see chap. 2.2.12.8, p. 206)
To edit attributes of type "Time" (TIME).

Add references (see chap. 2.2.12.9, p. 207)
To create model and object references of the type "Reference" (INTERREF).

Distribution definition (see chap. 2.2.12.11, p. 211)
To define a distribution function to assign to variables.

Transition condition (see p. 215)
To define transition conditions in Subsequent relations.

Performer assignment (see chap. 2.2.12.14, p. 222)
To define performer assignments.

Resource assignment (see chap. 2.2.12.15, p. 227)
To define resource assignments.

Definition of a performer calendar (see chap. 2.2.12.16, p. 228)
To define a performer calendar and therefore when the performer will be present.

Definition of a process calendar (see chap. 2.2.12.17, p. 235)
To define a process calendar.

Hint: For referencing an attribute profile (see chap. 2.5, p. 291) move the mouse pointer to the respective cell, open the context menu (right mouse button) and select the menu entry "attribute profile".

8.2.3.1 Representation of Programcall Attribute Values

The attribute values of program calls can be represented differently depending on the definition in your application library. The possible values are as follows:

- "**-> [*Executable*]**"

  Instead of ***Executable*** the name of the defined program will be shown. The program is started by double clicking on the cell.

  Note: The name of the program and the parameter are defined by the ADOxx administrator and cannot be changed.

- "**-> [<automatically>]**"

  Double clicking on the cell will start the program associated with the parameter.

  Note: The parameter is defined by the ADOxx administrator and cannot be changed.

- "***Executable***"

Double clicking on the cell will show the input dialogue for the program call (see chap. 2.2.12.5, p. 205) while not displaying the parameter.

Note: The parameter is defined by the ADOxx administrator and cannot be changed.

- "***Parameter***"

  Double click on the cell to show the input dialogue for the program call (see chap. 2.2.12.5, p. 205).

- "**[*Parameter*]**"

  Double clicking on the cell will show the input dialogue for the program call (see chap. 2.2.12.5, p. 205) whereas the executable program has been predefined.

  Note: The executable program is defined by the ADOxx administrator and cannot be changed.

8.2.3.2 Edit References

By clicking on a cell with references a window showing the reference targets is displayed.

References to models in the list "Reference targets" (see fig. 44) will show the following information:

- Reference status
- Model type icon of the referenced models
- Name of the referenced models

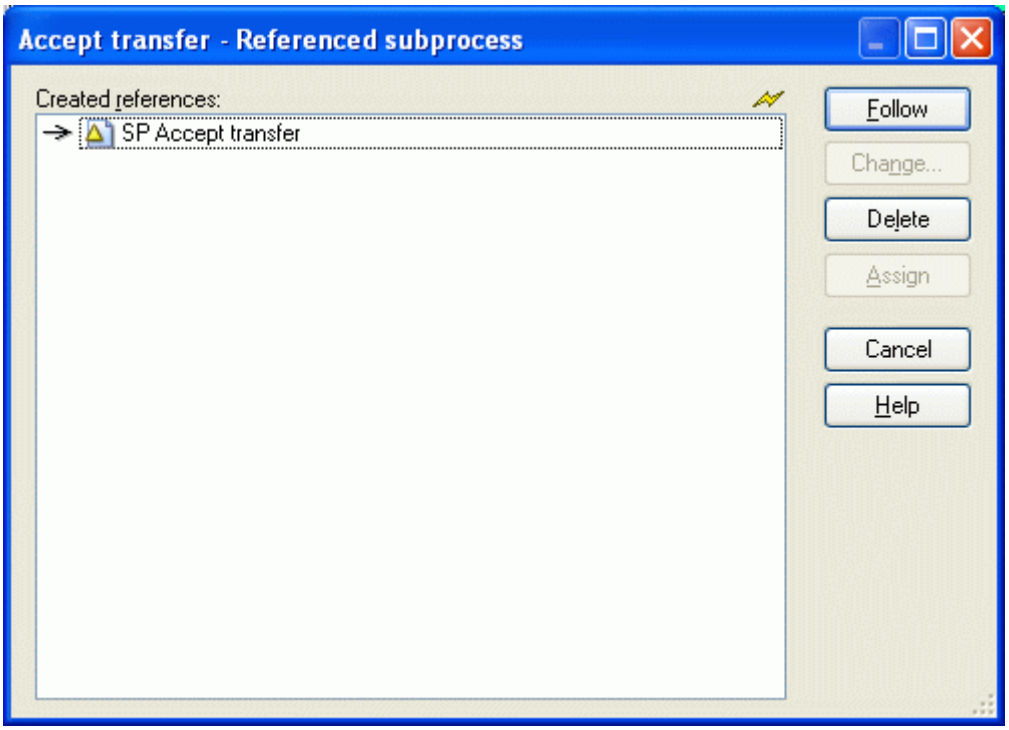


Figure 44: Editing a model reference attribute

References to objects in the list "Created references" (see fig. 45) will show the following information:

- Reference status
- Class symbol of the referenced object

- Name of the referenced object
- Model type icon of the model containing the referenced object
- Name of the model containing the referenced object

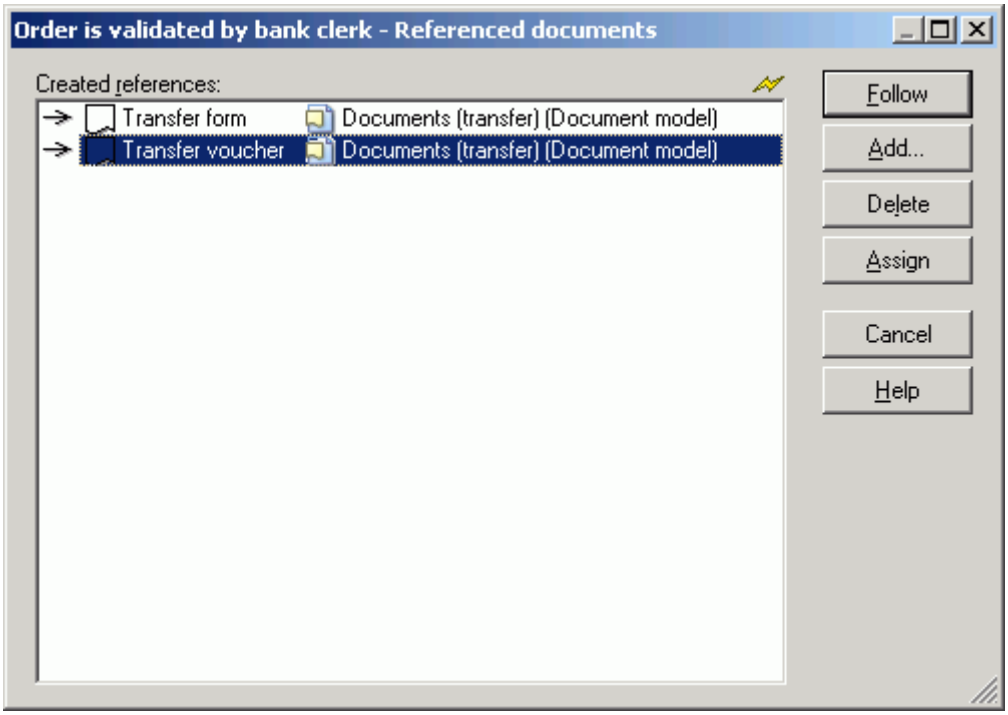


Figure 45: Editing an object reference attribute

The status of the reference at the beginning of each line indicates whether the reference is valid ( ) or broken ( ).

By clicking on the button:

"Follow"	you can follow the previously selected reference and display the referenced model or object;
"Change"	enables you to edit (see chap. 2.2.12.9, p. 207) the reference;
"Add"	helps you to define (see chap. 2.2.12.9, p. 207) new references;
"Delete"	removes the previous references;
"Assign"	confirms the changes performed.

8.3 Enter Column Width

The ADOxx browser enables the user to define the width of any column. To enter a column width, place the mouse pointer on the appropriate column, open the context menu (right mouse button) and select the "Column width" menu item.

In the "Set column width" (see fig. 46), the current column width is shown in the "width" field.

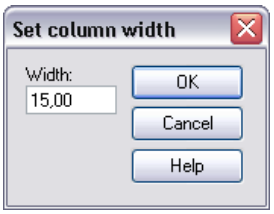


Figure 46: Set column width

Enter the value of the column width and then click on the OK button.

Hint: Every value between 0 and 200 is allowed for the column width.

Hint: The column width will be given in so-called norm signs, i.e. the entered value defines the number of characters, which will be displayed in the standard font in the cell.

8.4 Adjust Column Width

Within the ADOxx browser it is possible to dynamically adjust the width of any row. In order to do this, place the mouse pointer at the division point between two columns (see fig. 47) in the header column.

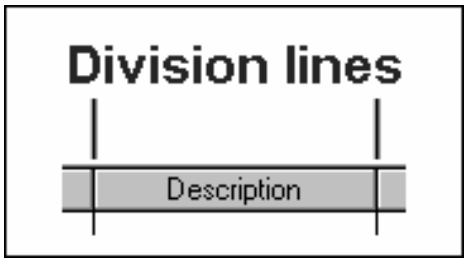


Figure 47: Column separator

The mouse pointer will change to when it is placed in the correct place. Once the mouse pointer changes, it is possible to increase or decrease the width of a column by pressing and holding the left mouse button and moving the mouse in the required direction. In this way it is possible to display the results in columns wider than the standard maximum.

It is also possible to adjust the width of a particular column by double-clicking in the header of that column (this will adjust it to the optimal width).

Hint: By selecting "Adjust columns" in the context menu (right mouse button) all the columns in the ADOxx browser are adjusted to their optimal width (wide enough to display the entire contents of the largest value in the column to a maximum of 80 characters).

8.5 Enter Row Height

The ADOxx browser enables the user to define the height of any rows. To enter a row height, place the mouse pointer on the appropriate row, open the context menu (right mouse button) and select the "Row height" menu item.

In the "Set row height" window (see fig. 48) , the current number of visible rows is shown in the "Row" field.

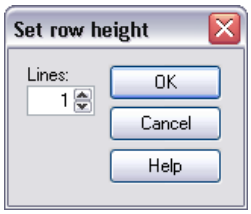


Figure 48: Set row height

Enter the number of rows you want to display and then click on the OK button.

8.6 Adjust Row Height

With the ADOxx browser it is possible to dynamically adjust the width of any column, so that cells that contain several lines can be shown completely.

To adjust a row height, place the mouse pointer on the division line between two rows (see fig. 49) in the first column.

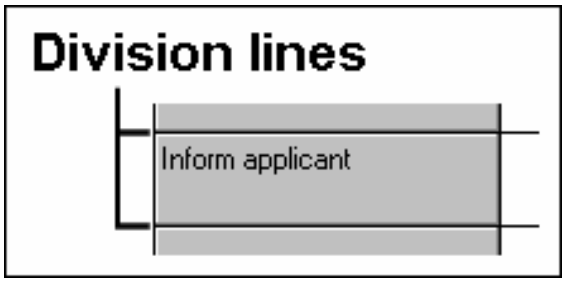


Figure 49: Row separator

The mouse pointer will change to when it is placed in the correct place. Once the mouse pointer changes, it is possible to change the height of a row by pressing and holding the mouse button and moving the mouse in the required direction. In this way it is possible to display results containing line breaks.

Hint: By selecting the **"Adjust row height"** menu item in the context menu (right mouse button), the row will be adjusted to it's optimal height.

Hint: By selecting the **"Adjust all rows height"** menu item in the context menu (right mouse button), all rows will be adjusted to their optimal height.

8.7 Expand / Shrink All

The ADOxx browser stores the output of queries, Capacity and Workload Analysis in groups (e.g. models). In order to present an overview to the user and to allow the user to find particular results quickly. These groups can then be expanded (or shrunk) to display more detail.

Unexpanded rows are characterised by the symbol . By clicking on this symbol, the rows will be expanded and this symbol will be shown.

To restore to the original state click on the symbol again and it will change from to .

Hint: By selecting the menu item **"Expand all"** in the context menu (right mouse button) all rows of the ADOxx browser will be displayed.

By selecting the menu item **"Shrink all"** in the context menu (right mouse button) only the highest-level row will be displayed.

8.8 Select Attributes/Columns

With the ADOxx browser it is possible for specific displays to shrink or expand any columns with attribute values or evaluation results

Hint: The function for the selection of attributes/columns to display is not available in all ADOxx browsers.

Hint: If you want to sort the browser contents according to a specific attribute, this attribute must be expanded before you call the "Sort" function (see chap. 8.9.1, p. 78).

Select attributes

To select the attributes to display (i.e. columns with attribute values), open the context menu of the browser (right mouse button) and select the "Select attributes" menu item. A window will be displayed (see fig. 50), in which all the attributes that can be displayed are shown following the notebook structure.

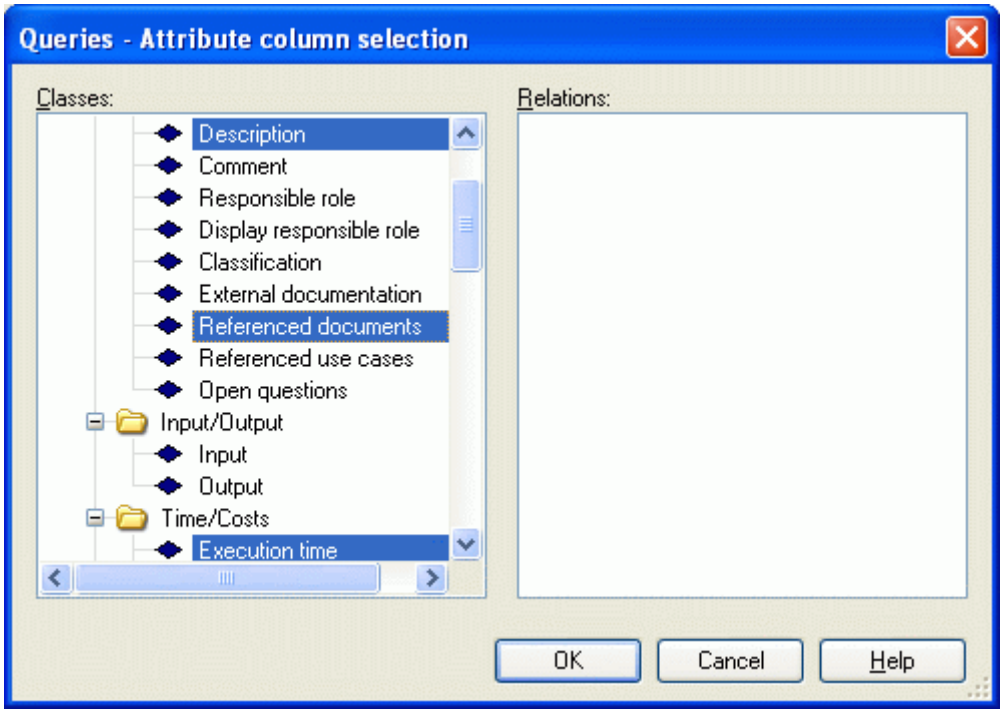


Figure 50: Select attributes (Example from the "Model search")

Hint: In the browser, the editable attributes are marked with the ◆ symbol, and the write-protected attributes with the ◇ symbol.

Select the attributes that the sort should be applied to and then click on the OK button.

Select columns

To select the columns to display, open the context menu of the browser (right mouse button) and select the "Select columns" menu item. All columns, that can be displayed in the browser are shown in the "Select visible columns" window (see fig. 51).

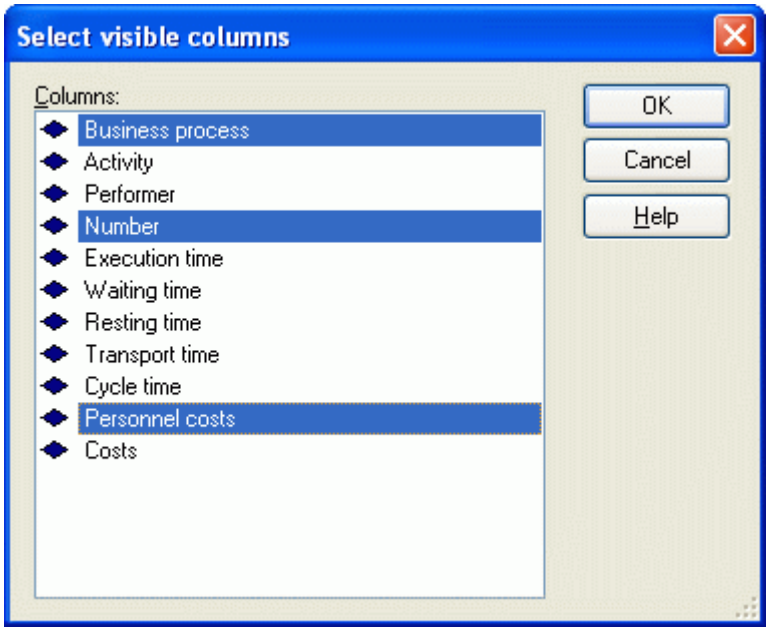


Figure 51: Select columns (Example from the "Analytical evaluation")

Select the columns you want to display and click on the OK button.

8.9 Sort

In the ADOxx browser, it is possible for specific displays to sort the browser contents using the attribute values displayed (within a column).

Hint: The function to sort is not available in every ADOxx browser.

To sort the browser contents displayed move the mouse to the top of the column and select the menu item "Sort (ascending)" or "Sort (descending)" from the context menu of the browser (right mouse button).

In addition it is possible to select the attribute to use (= column) for the sort using a dialogue for selection (see chap. 8.9.1, p. 78). In this case open the context menu of the browser (right mouse button, the mouse pointer must not be on the header of a column) and choose the menu point "Sort".

8.9.1 Sort by Attribute Columns

After selecting the menu point "Sort" within the context menu of the browser, all attribute columns within the respective notebook structure are displayed (see fig. 52).

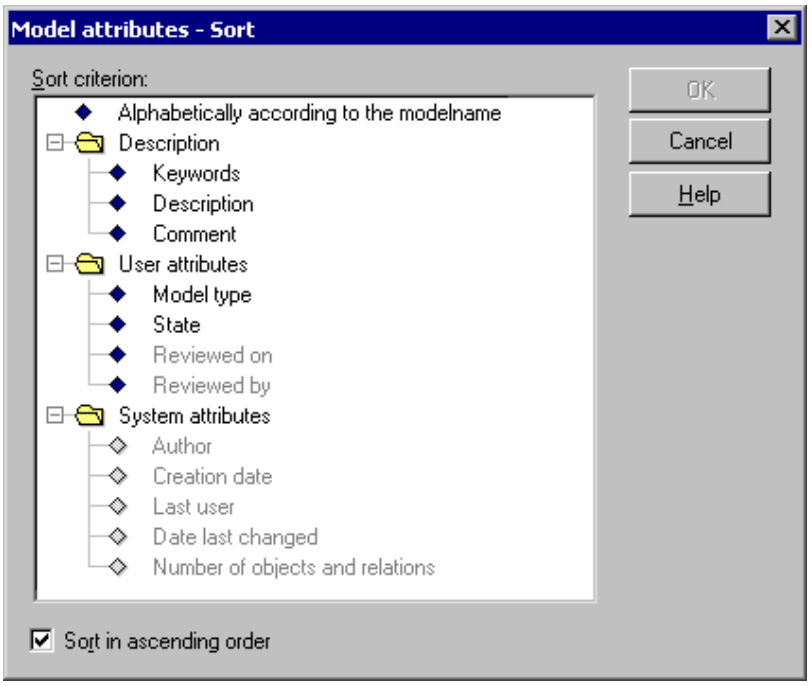


Figure 52: Attribute selection to sort (Example from tabular model display)

Hint: It is only possible to sort with columns being displayed in the browser. Columns/attributes not shown in the browser will be shown in the selection dialogue in grey font. If you want to sort using a column currently not displayed you will have to include the column beforehand (see chap. 8.8, p. 77).

Hint: In editable browsers, changeable attributes are marked with the symbol, and write protected attributes with the symbol.

Select the attribute, according to which the sort should be ordered by and then click on the OK button.

8.10 Align Attribute Values

The ADOxx browser offers the possibility to align the displayed attribute values of a column

- left,
- centred or
- right.

Align the values of a column by moving the mouse to the top of the column, open the context menu (right mouse click) and select the alignment ("Left", "centred" or "Right").

Hint: The current setting is indicated by a hook in the context menu.

Hint: The default setting for text values is left, while numbers will be shown right justified.

8.11 Show/Edit (Attribute) Values

For improved readability, for editing or for showing additional information, you can display (attribute) values in a separate window.

Hint: The possibilities for displaying (attribute) values in a separate window depends on the type of the attribute (see chap. 2.2.12, p. 201) and on the type of the browsers (see chap. 8.1, p. 66).

If you want to show a (attribute) value, double-click on the cell with the (attribute) value to display and the following window will be opened (see fig. 53).

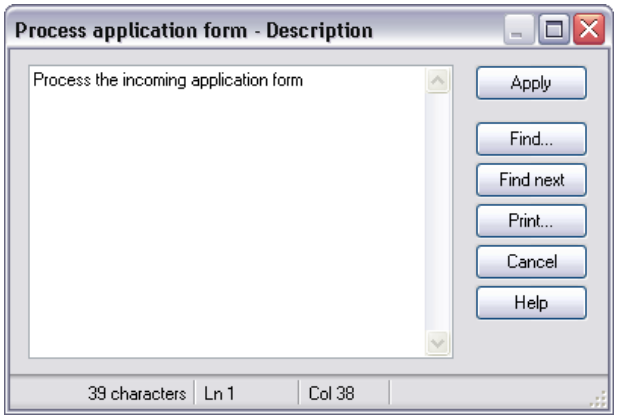


Figure 53: Show (attribute) values

You can save the displayed (attribute) value to a file ("Save" button) or print it out (button "press").

Hint: (Attribute) values can be changed in **editable browsers** (see chap. 8.2.1, p. 69) only.

8.12 Search

Using the button "Search" (or the corresponding item in the popup menu) you can search for a specific piece of text in the browser (in all expanded rows). The window "Search for browser contents" (see fig. 54) is displayed after clicking on this button.

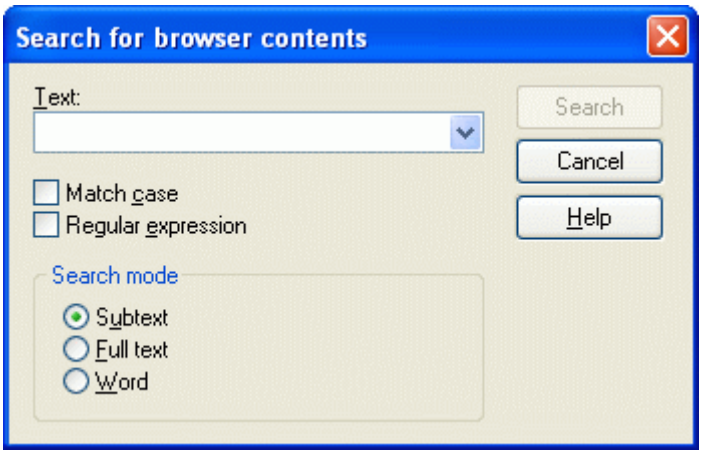


Figure 54: Search for browser content

Enter the text (character or expression) for which you would like to search for, into the input field "Text".

The following options are available for your search:

- **Match case**: when this option is selected the search will differentiate between upper and lower case letters.

- **Regular expression**: the default is to deactivate this option when searching for standard text. However by activating "Regular expression" (see chap. 6., p. 731) you can search the contents with the help of special characters.

  Hint: It is possible to use wildcards in the field "Text" ("*" for any number of characters, "?" for exactly one character).

  In order to use wildcards, the option "Regular expression" must be deactivated.

In addition, the following options are also available when searching for text:

- **Subtext**: means that the text being searched for can be part of a larger word or sentence.
- **Full text**: this is the default option - finds all cells which match exactly the full text being searched for.
- **Word**: finds all texts where the text being searched occurs as a single word (may be bounded by blanks or a full stop).

Clicking on the button "Search" begins the actual search. Each row, which is matched successfully during a search, will be highlighted in red.

8.13 Save

With the "Save" button you can save the displayed results to a file. The window "Save - options" (see fig. 55) is displayed once the button is selected.

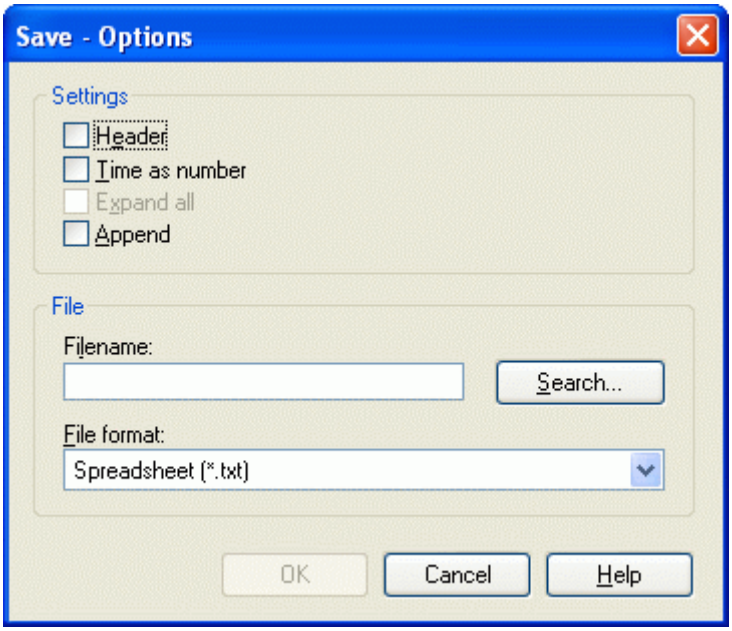


Figure 55: Save - properties

Within the "Save - properties" window you can choose:

- Properties (see chap. 8.13.1, p. 82)
- File formats (see chap. 8.13.2, p. 82) for external data

that have an effect on format and contents of the data.

ATTENTION: The available options depend on the type of results being displayed (e.g. Query results), i.e. not all options are always available.

In addition you must enter the filename and path to which the results should be saved in the input field "Filename".

Click on the OK button to continue.

8.13.1 Properties

The "Properties" allow you to influence the contents of the file being saved. The following properties are available:

Header:
When this option is selected, a header is added to the beginning of the file being created, which contains general details on the results (including date, time, the query run and model name(s)).

Time as number:
If this option is selected, then attributes which are currently saved in the ADOxx time format YY:DDD:HH:MM:SS will be converted to seconds in the file (e.g. the expression 00:000:00:30:00 will be converted to 1800 seconds).

Expand all:
Selecting this option will cause the entire contents of the browser to be saved regardless of how many levels of a hierarchical ADOxx browser (see chap. 8.2.2, p. 70)are actually expanded.

Note: This option is only available when the browser tree is not entirely expanded.

Append:
The contents of the browser will be appended to an existing file.

8.13.2 Formats

By clicking on the file format list, you can select the type of file in which the results should be saved. It is very important to choose the correct format especially when you wish to work further on the results in a spreadsheet program (e.g. Microsoft Excel) or when you wish to carry out a Comparison of results (see chap. 5.1, p. 433).

You can save the results

- for Spreadsheet (*.txt),
- for Spreadsheet (*.csv),
- for Word processor (*.txt),
- in Rich Text Format (*.rtf),
- as HTML file (*.htm) or
- as comparing representation (*.acr)

8.13.2.1 Save as spreadsheet (*.txt)

When the "**Spreadsheet (*.txt)**" format has been selected, you can specify user-defined cell separators to enable the resulting file to be easily used in spreadsheet applications.

After selecting the path and filename in the "Save - options" (see fig. 55) window and clicking on the OK button, the window "Separators for spreadsheet" is displayed (see fig. 56).

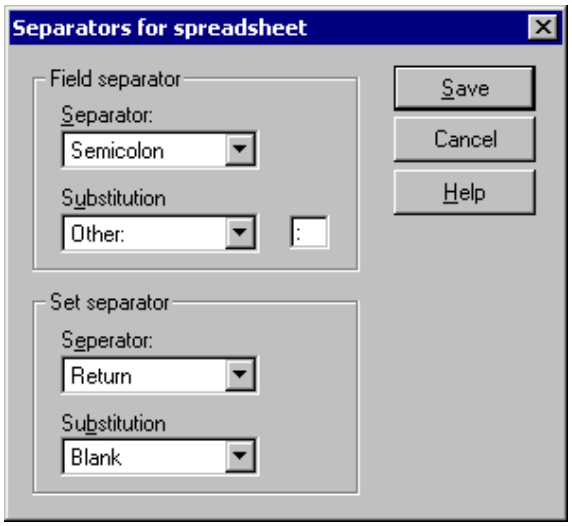


Figure 56: Save as spreadsheet

An entire row in the ADOxx browser - including the row description - is defined as a set. The "Set separator" identifies what character is used to indicate the end of a single row. A single cell is defined as a field and the "Field separator" is used to indicate an appropriate character to separate all cells.

In case the separator characters are already used within the data in the ADOxx browser, a "Substitution" character must also be defined which will be used to replace any separator characters that already exist.

The separator for the decimal point indicates which character should be used to separate decimal numbers (i.e. comma, full stop or other).

Hint: To guarantee that also the cells which contain line breaks will be correctly copied into a spreadsheet program, you can also save the contents of the browser to a CSV-file (see chap. 8.13.2.2, p. 83).

8.13.2.2 Save as comma-separated value (*.csv)

When you have selected the format "**Spreadsheet (*.csv)**", you can edit the contents of the browser window without losses for a later integration into a spreadsheet program.

Like the file format **spreadsheet (*.txt)** (see chap. 8.13.2.1, p. 82), the CSV format enables the definition of a whole line as a record. Additionally the line breaks which may appear in the cells will be correctly interpreted.

8.13.2.3 Save as text (*.txt)

"**Word processor**" saves the contents of the ADOxx browsers in this format as pure text to the selected file (file extension .TXT). The single lines will end in the file with a line break.

8.13.2.4 Save as RTF file (*.rtf)

The format "**Rich Text Format**" saves the content of the ADOxx browser as an RTF file (file extension .RTF). The table format is kept and this RTF file can be opened and further edited in a word processing program.

8.13.2.5 Save as HTML file (*.htm)

By selecting the format "**HTML file**" the contents of the browser are saved in a file in HTML format (file extension .HTM) which can then be viewed through a HTML browser (e.g. Internet explorer).

8.13.2.6 Save as comparable representation(*.acr)

When the format **"Comparable representation"** is selected, the results are written in ACR format (**A**DONIS **C**omparable **R**epresentation). The results can then be compared using the Comparison of results (see chap. 5.1, p. 433) option in the Evaluation component (see chap. 5., p. 433).

8.13.2.7 Save as AmiPro text (*.txt)

"**AmiPro text**" saves the contents of the ADOxx browsers in this format as text to the selected file (file extension .TXT) for a later using in AmiPro (IBM Lotus). The single cell will be separated with '**;**', single lines will end in the file with a line break.

8.14 Copy to Clipboard

When you select the menu item "Copy to clipboard" in the ADOxx browsers popup menu (right mouse button), the contents currently displayed in the ADOxx browser are copied to the clipboard and can then be pasted into Microsoft Excel for example in order to carry out some further work or calculations.

8.15 Print

In order to print the results displayed in the ADOxx browser, click on the button "Print". This will cause the window "Print" (see fig. 57) to be displayed:

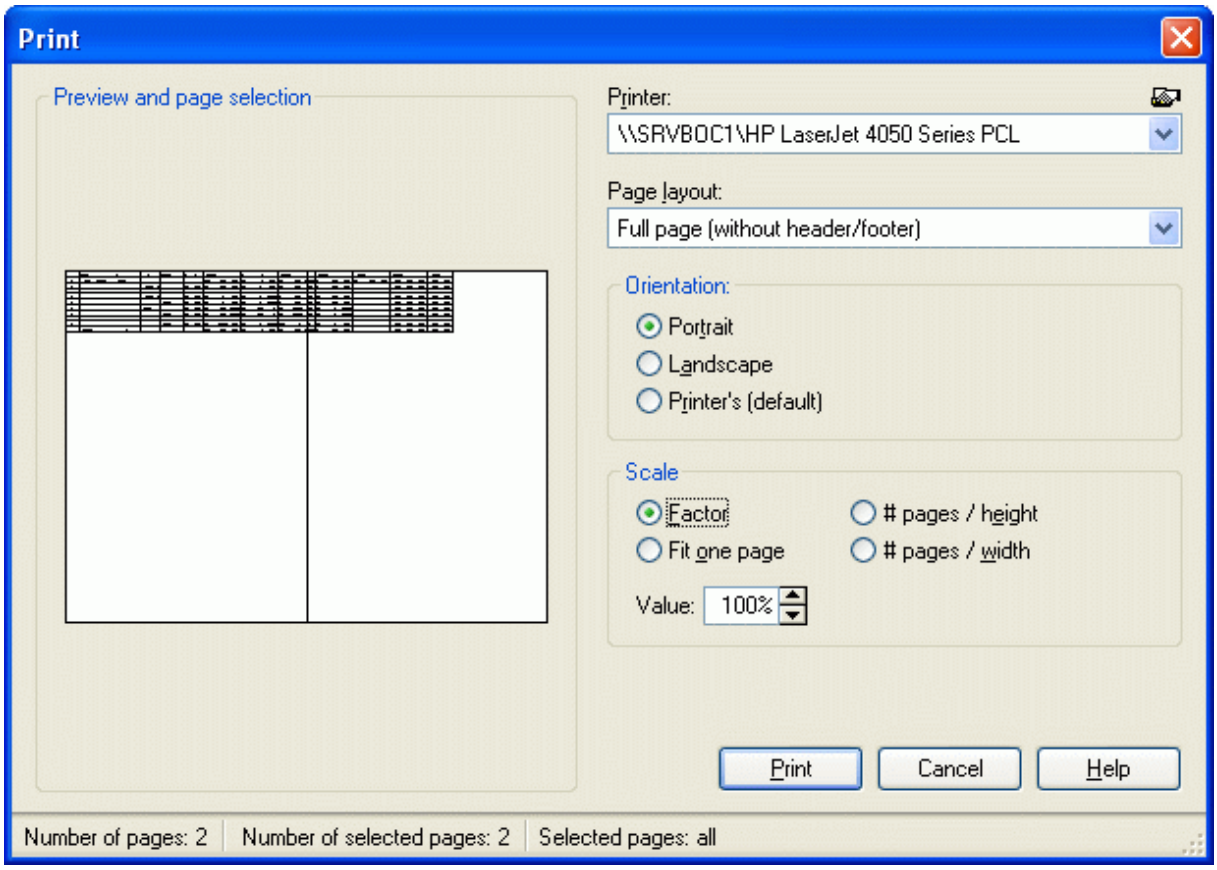


Figure 57: ADOxx-Browser-Print

Print:

The field **"Print"** contains a list of all available printers. Select a printer from the list (default printer is pre-selected). The printer settings can be checked and if necessary changed by clicking on the smart icon . If any problem with printers occur contact your system administrator.

Choose your settings for **"Page layout"**, **"Layout parameters"** and **"Scale"**. Check the "Layout parameters" (see p. 194) chapter for details.

The window **"preview and page selection"** is updated with each change. This print preview helps to choose pages (see chap. 2.4.17.3, p. 290) for the printout.

In the **status line** you will find information about the printout of several pages depending on the current layout and scale (total number of pages, number of selected pages, numbering of selected pages).

Hint: The numbering of pages is done by the line from the left to the right.

To start printing click **"Print"**. In the status window, you will find information about current printing status. By clicking on the "Cancel" button, the printing process will be cancelled immediately.

8.16 Diagram

In the window "<Browser content> - Diagram settings" (see fig. 58) you can define the settings for the graphical display of results.

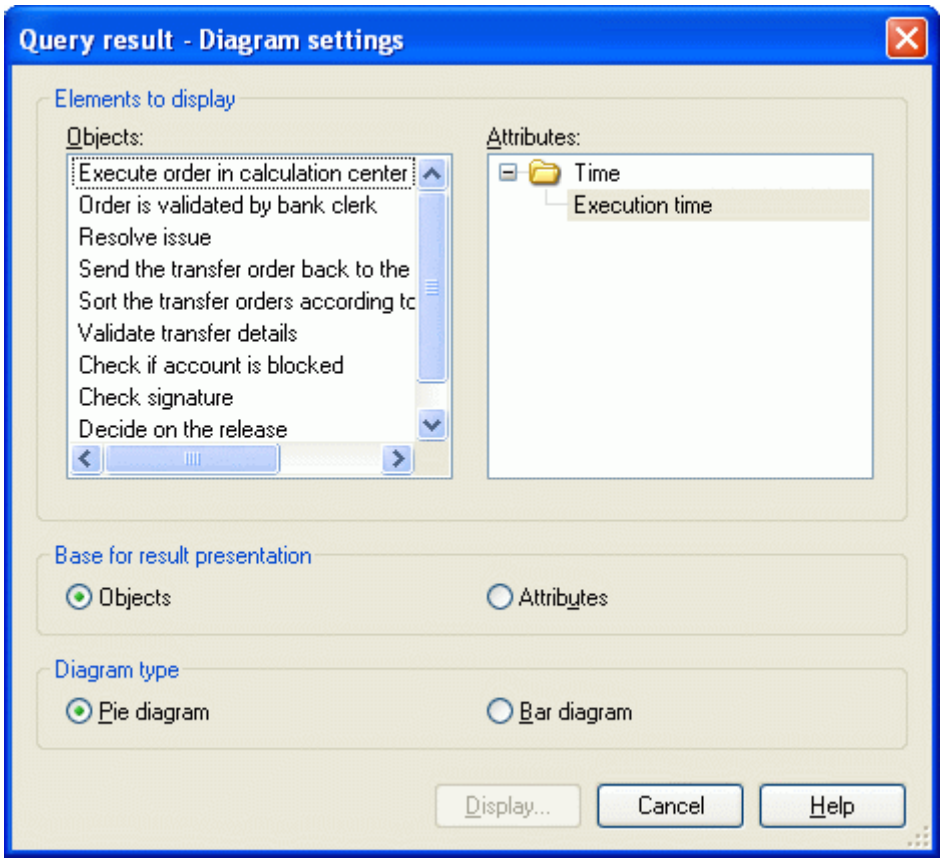


Figure 58: Diagram settings

Hint: Only numerical values (numbers) and times will be shown in the graphical display of the results comparison (see chap. 5.1.2, p. 436).

Select the objects and attributes to be displayed.

Hint: You can only select time or number attributes.

Define either an "object" or an "attribute" **basis for the results display**. The "object" option basis shows the selected attribute per object. The "attribute" option basis gives the selected objects per attribute.

The **type of diagram** sets the type of graphical display. It is possible to display the results either as a bar diagram (see chap. 8.16.1, p. 87) or a Pie chart (see chap. 8.16.2, p. 89). The bar diagram shows a group per value with a bar for each result. The pie diagram shows the part of each result for the selected value.

ATTENTION: Only one attribute can be displayed as a "Pie chart".

Once you have chosen the selected settings, click on the **"Display" button**. In the "Result comparison - graphical display" window, the results will be shown as graphics. If the graphical representation is bigger than the drawing area in the window, a scroll-bar will also be available to enable you to see the entire diagram.

On the right of the graphical result display you will find options for the following general layout settings, which are available for both types of diagram:

- "**Value**": When this option is selected, the appropriate values - in time or number format - are included in the graphical display.
- "**Legend**": By activating this option, the diagram will be described more precisely using a legend.

- "**Axis labels**": By activating this option, the axis of the diagram will be labelled.
- "**Items per page**": This refers to the number of items shown on one page. Changing this value can give you a better overview of the diagram. By reducing the number of items per page, the undisplayed items will be moved onto the next page.
- "**Page**": The page number of the current page is displayed. Changes to the **field "Items per page"** causes the total number of pages to change. If, for example the number of entries per page is reduced from 15 to 10, then the number of pages may be increased from 1 to 2.

You can copy (see chap. 8.16.3, p. 90) or print (see p. 51) the displayed diagram to the clipboard or to a graphic file.

By clicking on the "cancel" button you will close the graphical result display.

8.16.1 Bar Diagram

Results displayed in a bar diagram (see fig. 59) show several attributes of a type, (time or number) with their attribute values. The attributes will be displayed in a legend on the left upper side of the graphical results window.

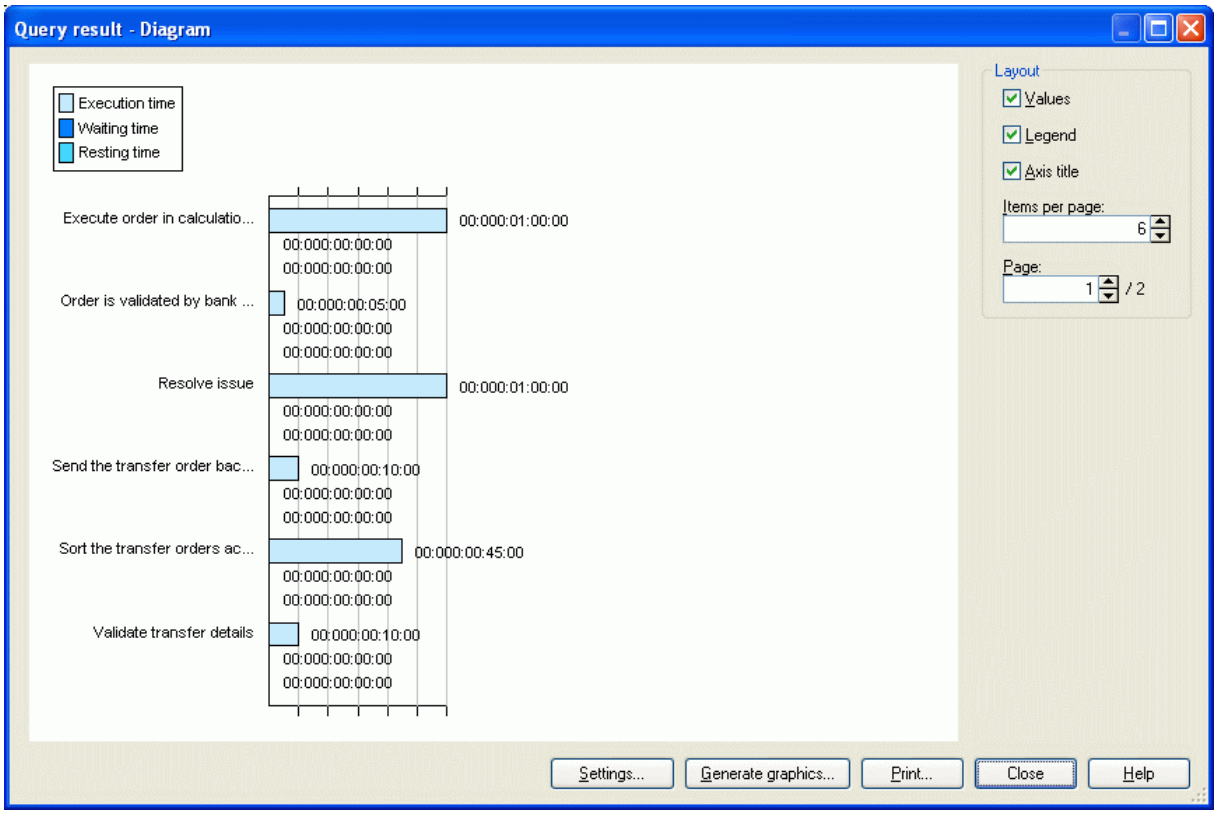


Figure 59: Bar diagram

On the right of the graphical result display you will find options for the following general layout settings:

- "**Value**": When this option is selected, the appropriate values - in time or number format - are included in the graphical display.
- "**Legend**": By activating this option, the diagram will be described more precisely using a legend.
- "**Axis labels**": By activating this option, the axis of the diagram will be labelled.

- "**Items per page**": This refers to the number of items shown on one page, changing this value can give you a better overview of the diagram. By reducing the number of items per page, the remaining items will be moved onto the next page.
- "**Page**": The page number of the current page is displayed. Changes to the **field "Items per page"** causes the total number of pages to change. If, for example, the number of entries per page is reduced from 15 to 10, then the number of pages may be increased from 1 to 2.

Click on the "Settings" button, to change the settings for the bar diagram display (see chap. 8.16.1.1, p. 88).

You can save the displayed diagram to the clipboard or copy (see chap. 8.16.3, p. 90) or print (see p. 51) it to a graphic file.

8.16.1.1 Settings for the Bar Diagram Display

You can change the settings concerning the display of the bar diagram in the "Diagram settings" window.

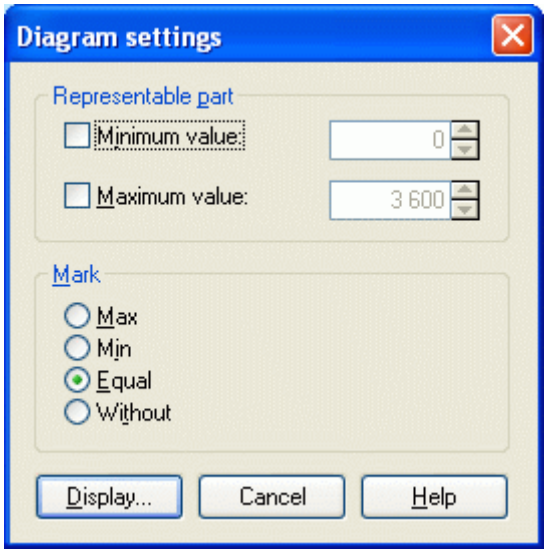


Figure 60: Settings for the bar diagram

In the **"Part to display"** window you can limit the bar diagram to a specific area by entering a lower and/or an upper limit. (For "time" attributes, the lower and the upper limit will be measured in seconds.)

The **"Mark" field** options enables better graphical editing of the files. You can click and select one of the following options for the display of the scale:

- "**Equal**" divides the length of the bar diagram in equal sections.
- "**Min**" marks the smallest attribute value.
- "**Maximum**" marks the largest attribute value.
- "**Without**" shows the diagram without divisions.

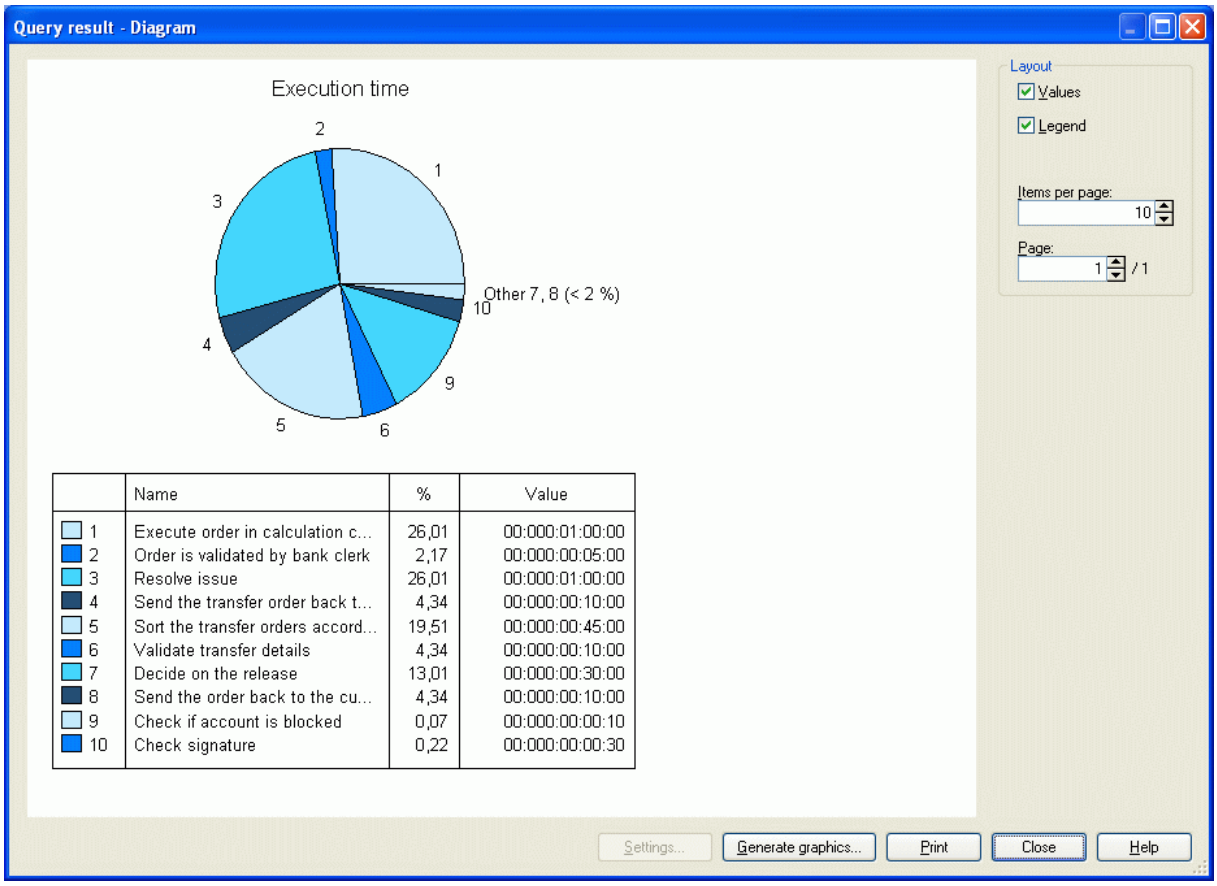
Click on the "Display" button to show the bar diagram with the changed settings.

8.16.2 Pie Chart

Only one attribute can be displayed within a pie chart (see fig. 61).

Hint: A graphical representation of the "Pie chart" is only available for the results of the "Capacity Analysis" (see chap. 4.3, p. 354) and "Workload Analysis" (see chap. 4.4, p. 367), Simulation algorithms as well as the Comparison of Results (see chap. 5.1, p. 433).

Hint: A graphical representation of "Pie chart" is only available for the results of the "Capacity Analysis" and "Workload Analysis" Simulation algorithms as well as the comparison of results.



	Name	%	Value
1	Execute order in calculation c...	26,01	00:000:01:00:00
2	Order is validated by bank clerk	2,17	00:000:00:05:00
3	Resolve issue	26,01	00:000:01:00:00
4	Send the transfer order back t...	4,34	00:000:00:10:00
5	Sort the transfer orders accord...	19,51	00:000:00:45:00
6	Validate transfer details	4,34	00:000:00:10:00
7	Decide on the release	13,01	00:000:00:30:00
8	Send the order back to the cu...	4,34	00:000:00:10:00
9	Check if account is blocked	0,07	00:000:00:00:10
10	Check signature	0,22	00:000:00:00:30

Figure 61: Pie chart

Each section of the pie chart contains a number, which refers to the name of the entry, its value and its percentage of the total sum. They are displayed in a tabular form, below the pie chart.

To the right of the graphical representation the following options are available:

- "**Values**": When this option is selected, the appropriate values - in time or number format - are included in the graphical display.
- "**Legend**": By activating this option, the diagram will be described more precisely using a legend.
- "**Page**": The page number of the current page is displayed. Changes to the **field "Items per page"** causes the total number of pages to change. If, for example, the number of entries per page is reduced from 15 to 10, then the number of pages may be increased from 1 to 2.

8.16.3 Generate Graphics

The button "Generate graphics" allows you to copy the displayed diagram onto the clipboard or to store it as either a BMP file (1-bit or 24-bit), a PCX file(8-bit or 24-bit) or a JPG file (24-bit), PNG file (24-bit) or. EMF file (24-bit).

Clicking on this button causes the window "Generate graphics" (see fig. 62) to be displayed.

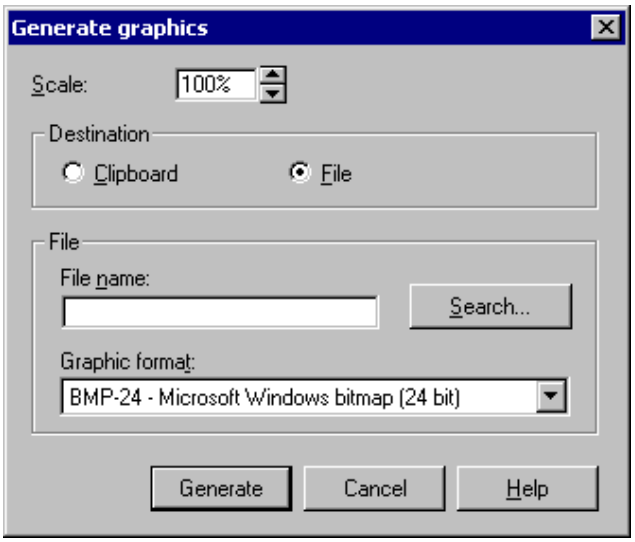


Figure 62: Generate graphic

Select the destination to which the graphic should be copied. If the clipboard is selected, no more options need to be set.

If copying to a file, then you must enter a path and filename in the field **"Filename"**. Additionally, you can select the size of the graphic to be generated in the field **"Scale"** and the type of graphic in the field **"Graphic format"**.

Click on the button "Generate" to actually generate the graphic.

9. Swimlanes

Swimlanes are special modelling objects stretching horizontally or vertically over the entire drawing area of a model.

> **Hint:** The availability of swimlanes depends on the definition of your application library. Please contact your ADOxx administrator for further information.

Swimlanes in ADOxx have the following characteristics:

- Horizontal swimlanes start at the top border of a model and are seamlessly placed below each other.
- Vertical swimlanes start at the left border of a model and are seamlessly placed next to each other.
- Swimlanes are always on the lowest possible level, i.e. all other modelling objects will always be visible when they are placed on a swimlane.
- The modelling objects on a swimlane are automatically connected to the swimlane via the relation "is inside".
- All modelling objects are usually either entirely inside or outside of a swimlane (exceptions can be made possible by extending the application library):
  - If a modelling object is placed on the edge of a swimlane the size of the swimlane automatically adjusts.
  - When moving a selected modelling object with the cursor the size of the swimlane is extended accordingly when the object is moved out of the existing boundaries of the swimlane.
- Swimlanes cannot be copied.
- Swimlanes cannot be moved.
- When deleting a swimlane all objects placed on it and all connectors leading to these objects will be deleted as well.

© BOC Group 2008

10. Versioning

10.1 Basics

ADOxx supports the following types of model and attribute profile versioning:

- **model-related versioning** (see chap. 10.1.1, p. 92)
- **time-related versioning** (see chap. 10.1.2, p. 92)

Hint: In the application library, one versioning type is defined and used for all model types available.

10.1.1 Model-related Versioning

With model-related versioning, a version number can be assigned to each model. This is the case when creating (see chap. 2.4.1, p. 256), saving as another version (see chap. 2.4.5, p. 260) or renaming (see chap. 2.4.8, p. 262) the model.

ADOxx neither demands a specific format nor a specific semantic for the version number. According to the use case, the user can define the format and semantics.

It is possible to create a new model version by creating a new model (see chap. 2.4.1, p. 256) and entering the same model name with a different version number.

Hint: In model-related versioning you can only create version numbers for models, not for attribute profiles (see chap. 12., p. 100) .

10.1.2 Time-related Versioning

In time-related versioning, a **date of validity** will be assigned during the creation of every model and every attribute profile. This indicates the date from which the attribute profile version becomes valid. The validity of a version ends with the start of validity of the next version. The youngest version of a model is the valid one.

Dates of validity are of significance especially for the use of inter-model references (see chap. 11.2, p. 98) and attribute profile references. References can only be created on models or on attribute profiles which are valid at the same time as the referencing model.

The format of the version number can consist of the elements "Day", "Month" and "Year".

Example:

If you do not want to create a version with a day level, the format of the version number can for instance consist only of month and year (see fig. 63).

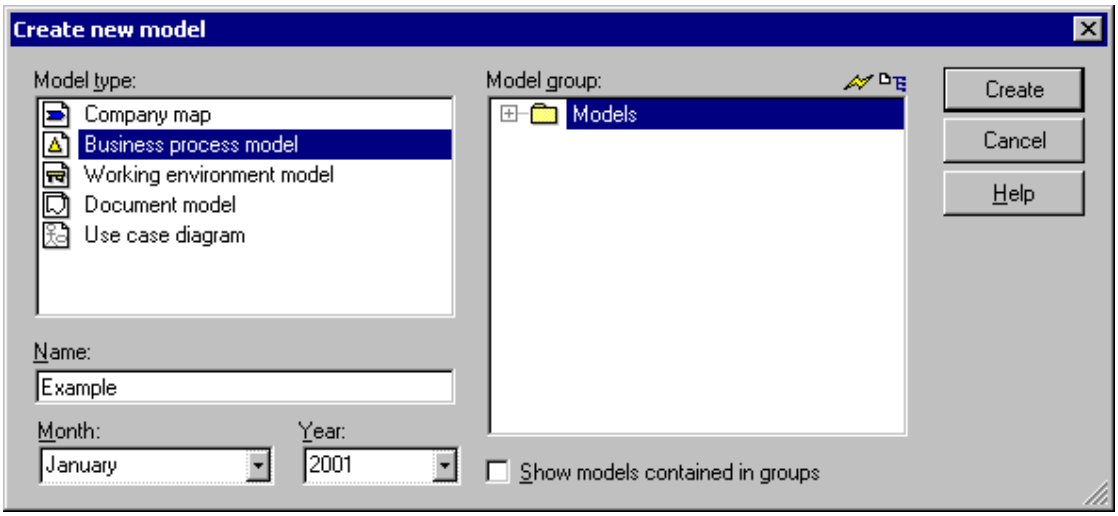


Figure 63: Example of time-related versioning when creating models

10.2 Effects

The versioning of models and attribute profiles has effects on the following functions:

- **ADOxx Modelling Toolkit:**
  - Create new model (see chap. 2.4.1, p. 256)
  - Create new version of an already existing model (see chap. 2.4.6, p. 262) (only for time-related versioning)
  - Rename models (see chap. 2.4.8, p. 262)
  - Save model with a new name (see chap. 2.4.5, p. 260)
  - Define application models (see chap. 2.4.16, p. 284) (only for time-related versioning)
  - Create and follow inter-model references (see chap. 2.2.12.9, p. 207) (only for time-related versioning)

    The order of inter-model references concerns all functions, for which models or attribute profiles have been selected for further editing (e.g. Analysis, Simulation, Evaluation, Import, Export).
  - Create attribute profile references (see chap. 2.5.6, p. 294) (only for time-related versioning)
  - Edit attribute profiles (see chap. 2.5, p. 291) (only for time-related versioning)
- **ADOxx Administration Toolkit**
  - Edit attribute profiles (only for time-related versioning)
  - Import models/Attribute profiles
  - Export models/Attribute profiles export (only for time-related versioning)

11. Inter-model References

With inter-model references, it is possible to reference from an object of one model into:

- other models, or
- objects in those other models.

In ADOxx 1.0 you can classify the inter-model references. This is of importance in all model selection lists, which contain the "Included referenced models" (e.g. "Open model", "Query", "ADL export"), since all referenced models of the models selected in these list will be determined and further edited according to their settings.

A "Reference tree" will be created with all models referenced by the settings. The figure below shows an example of a simple reference tree (see fig. 64).

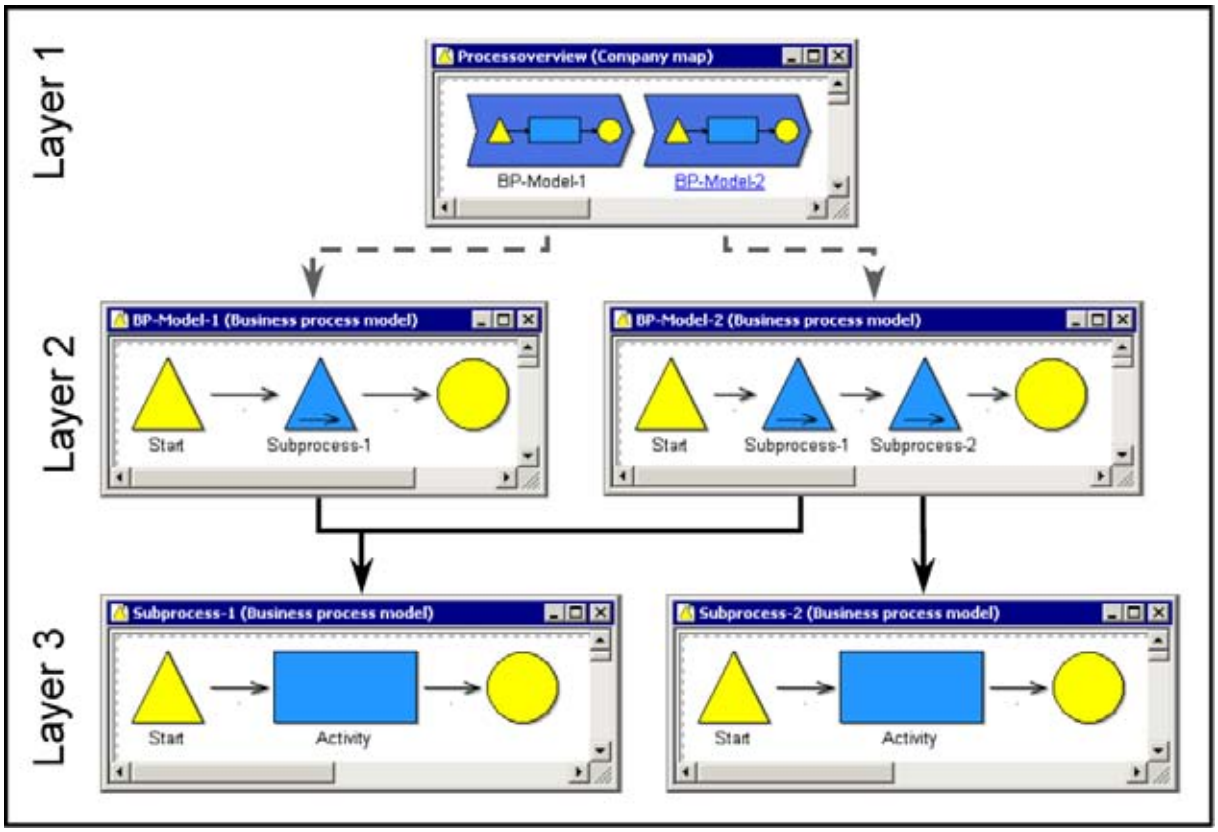


Figure 64: Example of reference tree

In the "main model" (Level 1) the models "Model 1" and "Model 2" (level 2) are referenced, they contain the models "referenced model A" and "referenced model B" (Level 3). Starting from the "main model", you can form the reference tree shown, i.e. by selecting the "main model" and activating the option "Including referenced models", the five models will be loaded for editing.

With the settings (see chap. 11.1, p. 97) for the inter-models referenced you will determine,

- the depth (level) of which the references should be taken and
- whether they are main or side references.

The **depth** indicates of which level (starting from the selected model) the references will be taken and the referenced models taken into account. In the example with the reference tree (see fig. 64) , a depth of 1 starting from the '"main model" would give us as a result only the models "model 1" and "model 2".

The classification **main reference** and **side reference** determines how to proceed after having followed a reference .

Main referenced models will be treated like the (selected) starting model, i.e. all references contained in this model will be followed, if the limit of the depth allows.

For **side referenced models**, the references contained in these models will only be followed if:

1. they start from the same attribute of an object of the same class and
2. the limit of the depth allows it.

In contrast to the main references, the limit of the depth of side references will be evaluated each time the side references are used (see fig. 65). The depth of main references is verified globally, over all levels.

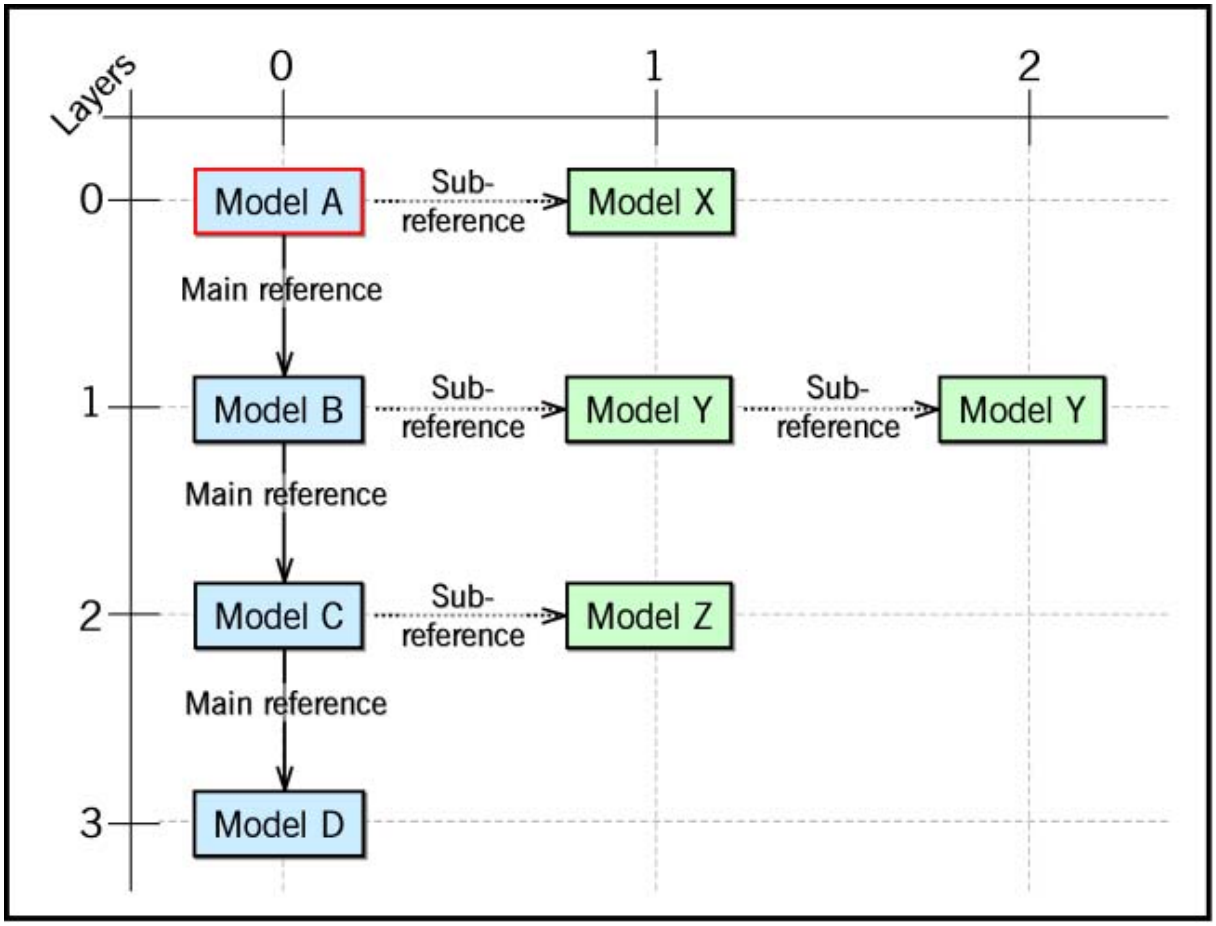


Figure 65: Schematic display of reference evaluation

In the schematic display of reference evaluation (see fig. 65) the "model A" is the original model, i.e. the model from which the reference tree is formed. The models: "model B", "model C" and "model D" will be referenced from the "model A" with a main reference (e.g. sub processes) and form together the reference tree of the main reference (main tree). A depth of "2" for this main reference means, that the "model B" will not be taken into account in the reference tree to be created.

© BOC Group 2008

The side references (e.g. documents) will be considered per level, i.e. the depth Evaluation is always done starting from a model of the main tree and the count starts again at 0. A depth by "1" for the side references means - looking at the figure above - that the "model N" is not considered when creating the tree.

Differences between main and side references are explained in the graphic below (see fig. 66).

Hint: For the references in this example, the depth is set to <unlimited>.

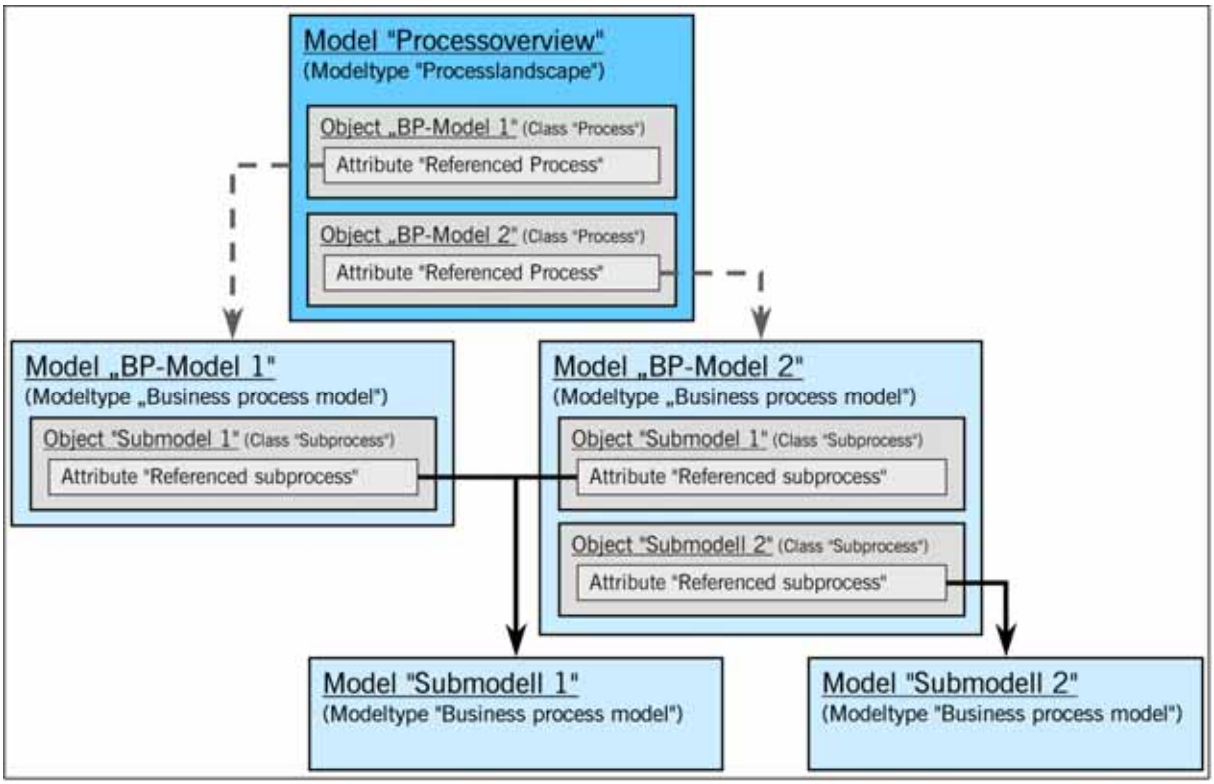


Figure 66: Example of a reference tree

The reference attributes "called process" in the class "process call" (in the models of the type "Business Process Model") and "referenced process" in the class "process" (in the models of the type "Process map") are defined within the ADOxx-Default-Library (see chap. 16., p. 802).

The references of the attribute "referenced process" are shown by - - ➤, and those of the attribute "called process" by ➤.

If the references of the attribute "referenced process" (- - ➤) are defined as side references, the results are - starting from the "process overview" process map - the Business Process Models "BP model 1" and "BP model 2", irrespective of the definition of the references in the attribute "called process" (➤), since the results of side references are taken into account, only if the same side references refer to further models.

If the references of the attribute "referenced process" (- - ➤) are defined as main references, the results are - starting from the "process overview" process map - all Business Process Model drawn, irrespective of the definition of the references of the attribute "called process" (➤).

11.1 Reference Settings

In the window "references - settings" (see fig. 67), you can edit the classification (main or side references) and the depth for all inter-model references defined in the application library.

Hint: You can edit the reference settings in all the model selection lists (e.g. "Open model", "Query", "ADL export"), in which the option "including referenced models" is available and activated, by clicking on the "References" button.

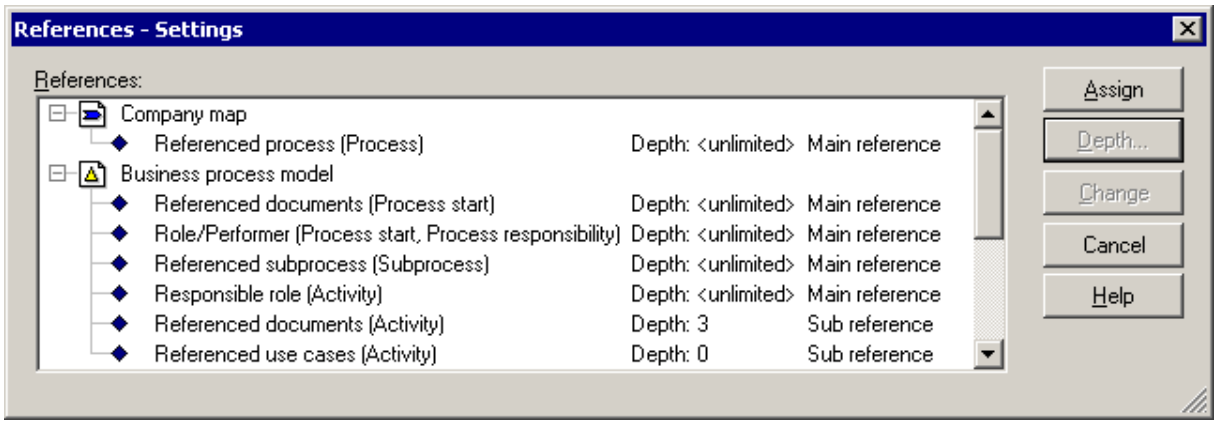


Figure 67: References - Edit settings

Select the reference to edit and adjust the **depth** (see chap. 11.1.1, p. 98) by clicking on the button "Depth".

Change the classification of a selected reference by clicking on the button "Change".

Alternatively, you can edit the selected reference via the context menu (see fig. 68) (right mouse button).

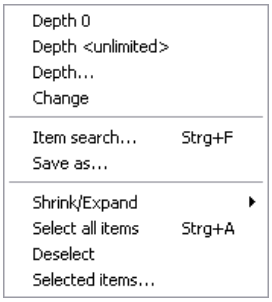


Figure 68: context menu to edit the reference settings

The following menu items are available to change the reference depth:

Depth 0 sets the value to "0" (the menu item will only be displayed, if the value of the selected reference is not 0).

Depth <unlimited> sets the value by "<unlimited>" (the menu item will only be displayed, if the value of the selected reference is not <unlimited>).

Depth opens the window "Edit reference depth" (see chap. 11.1.1, p. 98).

By selecting the "Change" menu item, you will change the classification (main reference <-> side reference) of the selected reference.

11.1.1 Edit Reference Depth

The reference depth gives the depth (down to which level) - starting from a selected model - you will follow the references and take the referenced models into account.

Enter the depth for the previously selected inter-model reference by selecting any predefined value (0 - 1 - 2 - 3 - 4 - 5 - <unlimited>) or by entering any value in the list "reference depth" (see fig. 69).

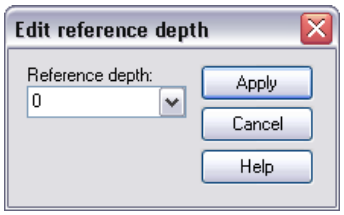


Figure 69: Edit reference depth

Once you have selected the reference depth, click on the "Apply" button to assign the value of the reference.

11.2 Effects of the Versioning on Inter-model References

Inter-model references for model-related versioning

For model-related versioning, inter-model references always point at concrete model versions or at objects in concrete model versions. The version number of the model only has an effect on the inter-model references, in so far as it (together with the model name) clearly identifies the target-model or the target-object.

Inter-model references for time-related versioning

For time-related versioning, inter-model references always point at version threads. The actual goal of a reference will be dynamically determined via the current version number: Inter-model references always point at the version of the target threads, which is valid at the same time as the source model.

Example:

1. Starting from "model A (April 2001)", an inter-model reference points at "model B". "Model B" is to be found only in the versions "January 2001" and "February 2001". In April 2001, the version "February 2001" is therefore still available. Therefore the target of the reference is "model B (February 2001)".

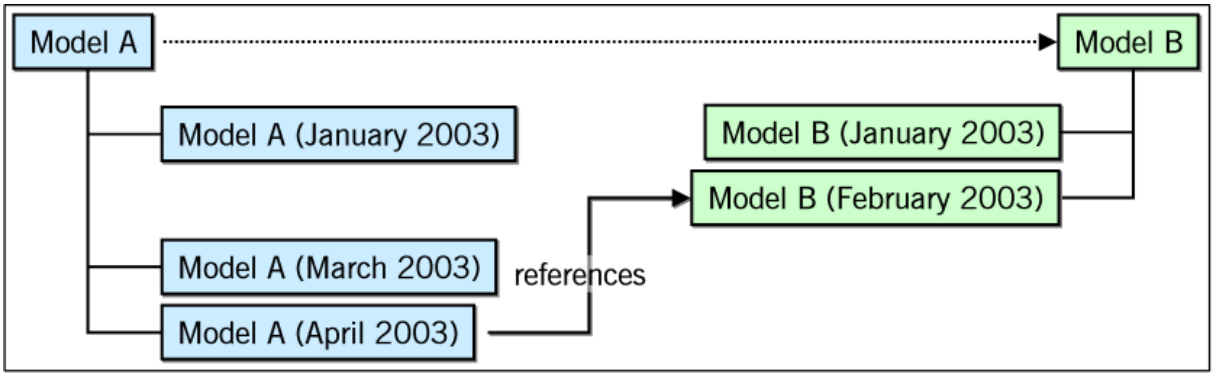


Figure 70: Reference from "model A (April 2001)" to "model B (February 2001)"

2. If a new version for "March 2001" is created from the "model B", it will be valid in April 2001. The inter-model reference points therefore automatically at "model B (March 2001)".

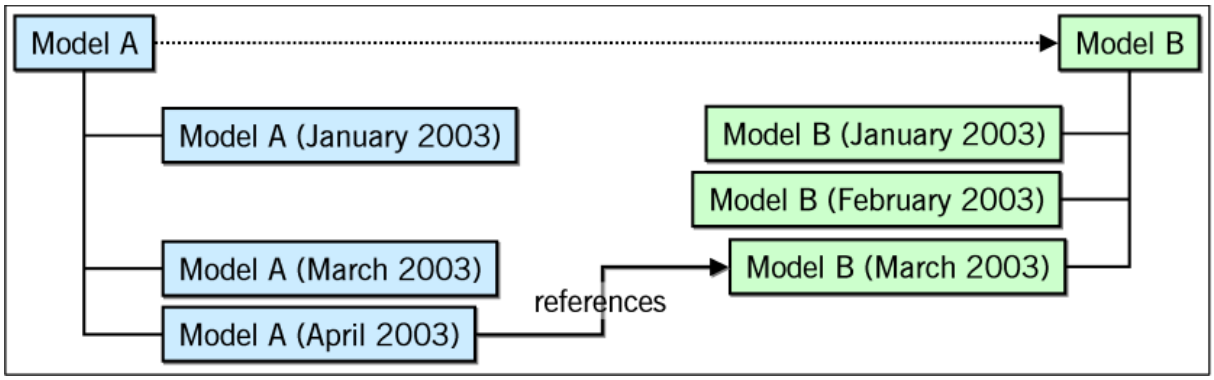


Figure 71: Reference from "model A (April 2001)" to "model B (March 2001)"

3. If moreover a new version for "May 2001" is created from "model B", this has no effect on the inter-model reference. In April 2001 "model B (March 2001)" is still valid. Therefore the inter-model reference still points at this version.

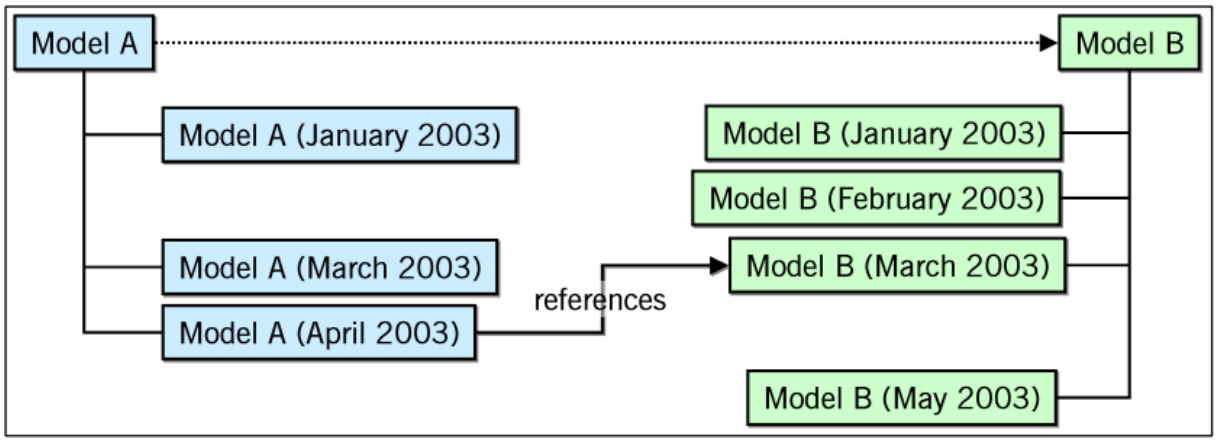


Figure 72: Reference from "model A (April 2001)" to "model B (March 2001)"

4. If a new version for "May 2001" is created from "model A", the intermodel- reference of "model A (May 2001)" points at "model B (May 2001)". The inter-model reference of "model A (April 2001)" still points at "model B (March 2001)".

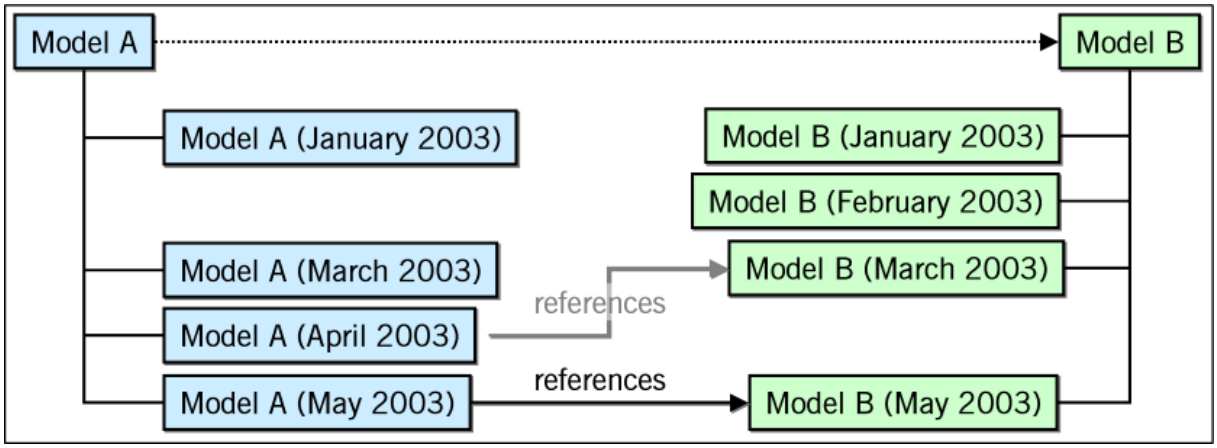


Figure 73: Reference from "model A (Mai 2001)" to "model B (May 2001)"

12. Attribute Profiles

With ADOxx attribute profiles, it is possible to create a repository, i.e. attribute structures which you can reuse in ADOxx.

With attribute profiles, you can define and change recurrent attribute values at a specific place (attribute profile administration). This way, the consistency of the deposited information is guaranteed.

An attribute profile consists of any number of attributes of any type (see chap. 7., p. 733).

> **Hint:** The attribute profile administration is a component of the Administration Toolkit. Additionally, authorised users can administrate attribute profiles in the Modelling Toolkit (modelling component).

Editing in the Attribute Profile Management

Concrete attribute profiles - i.e. attribute sequences determined with values - will be created, edited and deleted in the attribute profile administration (see fig. 74).

The creation of an attribute profile means that an attribute profile class has been instanced to an attribute profile (like instancing of an object of a class in the modelling in Modelling Toolkit).

Additionally in the attribute profile administration, it is possible to create so-called (attribute) profile group hierarchy, in order to set up user-specified structures for the attribute profiles.

Usage in modelling

The use of attribute profiles is possible in the modelling component through the creation of references to defined attribute profiles (see fig. 74).

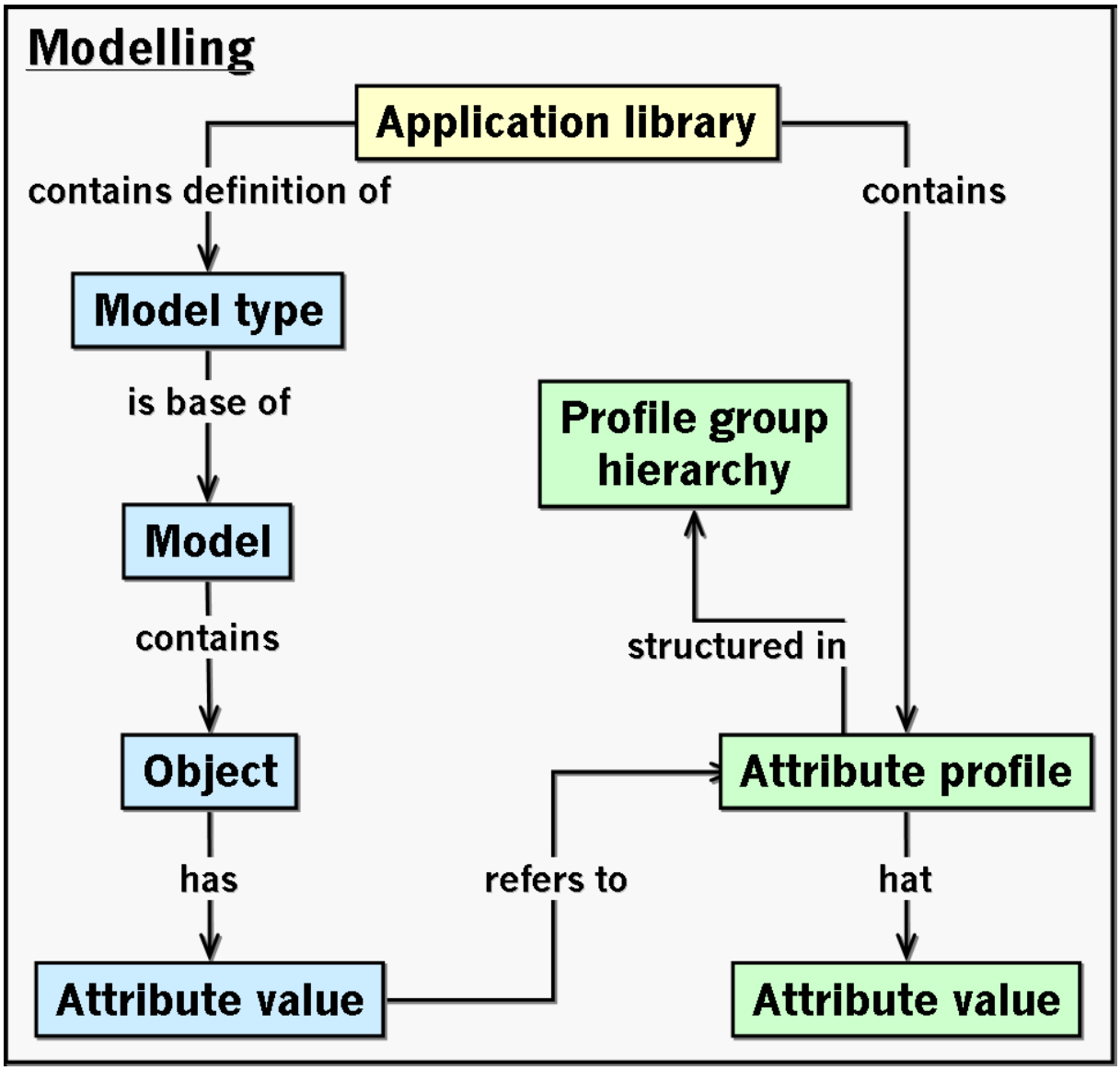


Figure 74: Attribute profiles in Modelling Toolkit

As previously presented in the description of the terms and context (see chap. 1., p. 32), model types and classes will be defined in the application library.

Models of a specific type will be created in the modelling component. The objects of the available classes will be created again in a model. The (object) attributes will be determined with attribute values in an object.

If an object contains an attribute of the type "attribute profile reference" (see chap. 7.1, p. 733), the assignment of values is done via an attribute profile contained in the profile group hierarchy and the value it contains.

Example of attribute profiles and attribute profile references

The following picture shows the context of classes, objects, attributes, models, model types, attribute profile classes, attribute profiles and profile groups in graphics (see fig. 75).

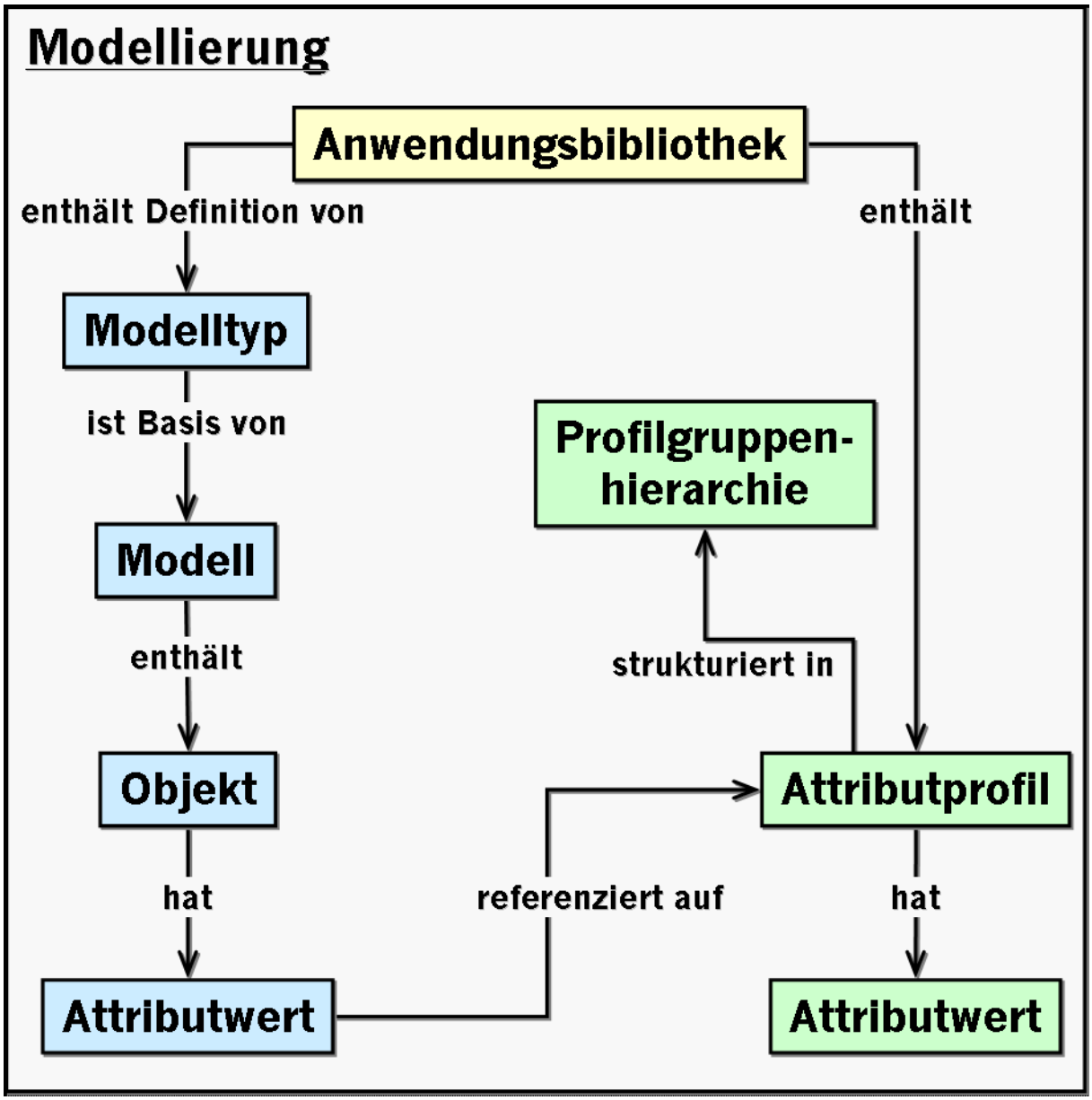


Figure 75: Schematic representation for the use of attribute profiles

Definition in the application library:

The class "performer" and the attribute profile class "presence" are defined in the application library.

The attribute profile class "presence" contains the definition of the (object) attributes "name" (from the attribute type "text") as well as "day/week" and "hours/day" (both from the attribute type "float").

In the class "performer", the (object) attribute "name", "first name" (of the attribute type "text") and "presence" (of type "Attribute profile reference") are defined. When defining the attribute "presence", you will enter additionally to the attribute type the referenced attribute profile class "presence".

Editing in the Attribute Profile Management:

In the attribute profile administration you will find the structure of the profile group hierarchy ("profile group 1" and "profile group 2") which contain the attribute profile "full time", "part time" and "temporary

help" (instanced by the attribute profile class "absence"). The values "day/week" and "hours/day" are assigned for each of these attribute profiles.

Usage in modelling:

In ADOxx Modelling Toolkit, the model "Sales department" of the type "Working Environment models" is modelled with the three objects "Huber", "Müller" und "Schmidt" (all from the class "performer").

In the ADOxx Notebook of the three objects, the (object) attributes "name" as well as "day/week will be shown instead of the attribute "presence". The value assignment of these three attributes is done unlike the attributes "name" and "first name" by selecting an attribute profile, i.e. the attribute profile values entered in the attribute profile administration will be referenced into the ADOxx Notebook of the objects.

When you change the value of an attribute profile (see chap. 2.5.8, p. 295), e.g. the number of hours per day is reduced in the "full time" attribute profile from eight to seven, this has a effect on all the object which reference this attribute profile (object "Huber" and "Müller").

13. Record Attributes

Record attributes allow for the realisation of complex attributes. A record attribute is built like a table, i.e. with a lot of rows (0 is possible) and columns. Single rows can be added and deleted accordingly. The rows are numbered starting from 1. The columns display the attributes from which the record attribute is built. All types of files are allowed as possible attribute types inside a record attribute, except a complex attribute and an attribute profile reference. Default values can be defined for the single fields, so that they are already filled when adding a new row. A so-called multiplicity can be defined in each record attribute, which gives the maxiumum number of rows, which can constitute this record attribute.

The arrangement is graphically displayed in the picture below and described as follow (see fig. 76):

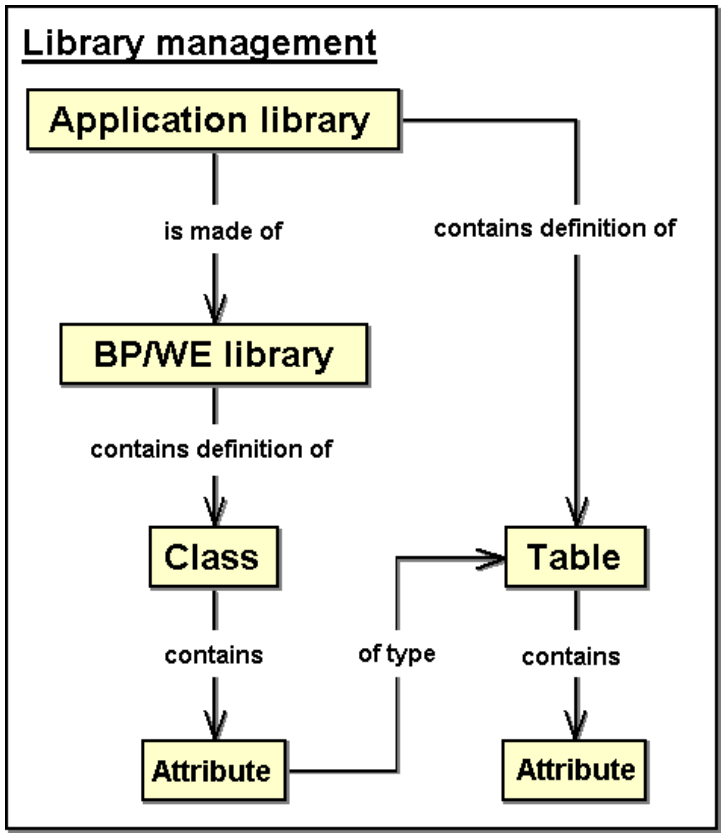


Figure 76: Schematic arrangement of record attributes in ADOxx

Unlike modelling or relation classes (see chap. 1., p. 32), record attribute classes will be defined in an application library and so can be used in all model types.

Modelling and relation classes (= classes) will be defined in a business process or Working Environment library for the model types of this library and are thus available exactly in these model types (for modelling in the Modelling Toolkit).

Record attribute classes and modelling/relation classes equally contain definitions of (object) attributes, which enable the connection of an (object) attribute of a class with a record attribute class through the definition of record attributes.

Usage in modelling

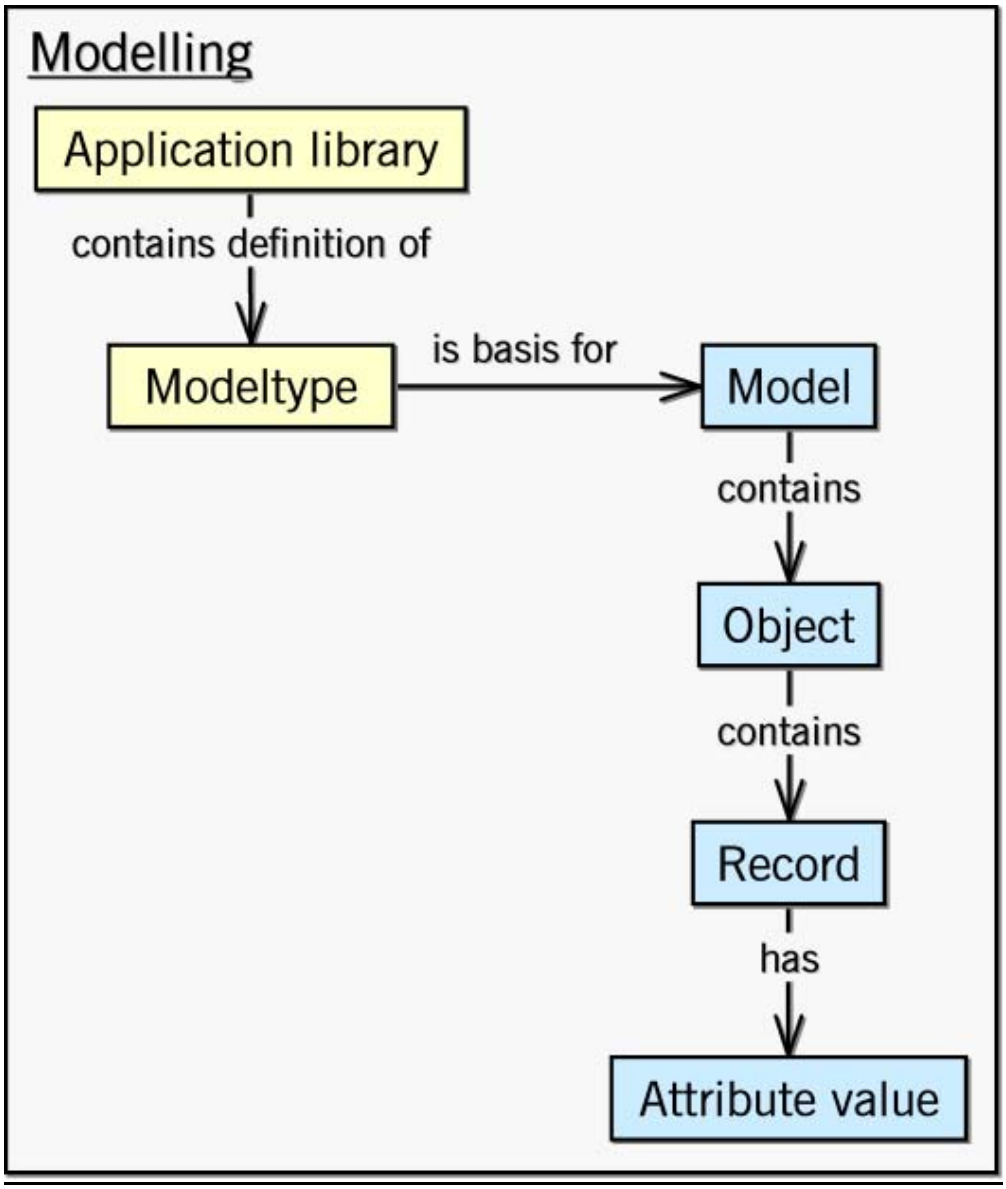


Figure 77: Record attributes in modelling

As previously mentioned in the description of the Terms and context (see chap. 1., p. 32), model types and classes will be defined in the application library.

Models of a specific type will be created in the modelling component. The objects of the available classes will be created again in a model. The (object) attributes will be determined with attribute values in an object.

If an object contains an attribute of the type "table" (see chap. 7.13, p. 736), the default value assignment is done via a record attribute class and the value it contains.

Record attributes are displayed graphically in the ADOxx browser (see fig. 78). You can edit the single fields the same way as for tabular modelling, i.e. support dialogues will be given for special attribute types. The icons and are used to add or delete selected rows.

Hint: Both functions are also available in the context menu of the ADOxx browser.

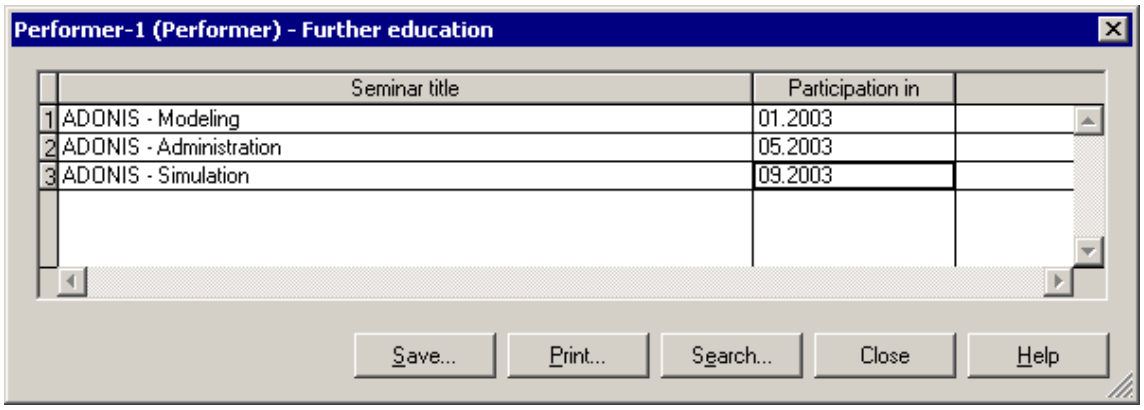


Figure 78: Display of a table attribute (example)

Each addition will cause a row to be added to the end of the table. The fields of the rows will be filled with the default values of the attributes.

To delete one or more rows from the table, select the row(s) by clicking in the numbered headers (first, grey column). You can also select several rows by pressing the control key and the shift key at the same time. Then click on the icon or select the appropriate menu item from the context menu. Confirm the appearing security query to delete the selected rows.

Part IV
Components of the Modelling Toolkit

The **ADOxx Modelling Toolkit** is the main component of ADOxx. By using the Modelling Toolkit you can model (see chap. 2., p. 126) your business processes as well as your working environment (organisational structures).

The Modelling Toolkit offers you extensive functionality which allows you to acquire (see chap. 1., p. 108), analyse (see chap. 3., p. 306), simulate (see chap. 4., p. 332) and evaluate (see chap. 5., p. 433) your business processes and working environments in a cost-effective way.

The Modelling Toolkit offers you extensive functionality which allows you to analyse (see chap. 3., p. 306) your key figures in a cost-effective way.

> **Hint:** All examples and descriptions in this part of the user manual refer exclusively to the ADOxx standard application library (see chap. 16., p. 802). Text and numeric entries in the pictures may differ from those on your screen.

1. Information acquisition

The "Acquisition" component of the ADOxx Modelling Toolkit supports you during the acquisition of data, which can later be imported into ADOxx Business Process Models.

This component uses HOMER acquisition tables, in which you can gather data on product families and products, on activities and their attributes, on organisational units and roles.

The availability of the "Acquisition" component (and the "Acquisition tables" module) within this component depends on the actual configuration of the Modelling Toolkit. (see chap. 4., p. 729)

Clicking on the appropriate smart icon in the components bar activates the "Acquisition" component.

It is also possible to activate this component by clicking with the right mouse button beside the component icons and then selecting the menu item "Acquisition" from the popup menu, which appears. This popup menu can also be activated through the function key <F9> and the Acquisition Component can be selected by pressing the (shortcut) key <a>.

Once the Acquisition Component is selected, a quick-access bar with these smart-icons is displayed:

HOMER-Acquisition tables (see chap. 1.1, p. 108)

"Models in the database" (see chap. 4.1, p. 56)

"Opened model windows" (see chap. 4.2, p. 57)

"Previous model window" (see chap. 2.3.1, p. 255)

"Next model window" (see chap. 2.3.1, p. 255)

1.1 HOMER (Overview)

The product process acquisition module HOMER is a toolkit enabling you to quickly and easily collect data on products, business processes as well as the activities related to them and the organisational units in which they are carried out.

The acquisition tables in HOMER support you to structure your modelling projects and to collect data "on the spot" (e.g. in the specialist divisions).

HOMER is a program independent of ADOxx, which uses the tabular and function-orientated structure of Microsoft Excel, in order to collect and administrate information as easily as possible.

> **Hint:** HOMER version 3.6 or higher needs Microsoft Excel version 8.0 (Microsoft Office 97) or higher.

Using HOMER:

1. **Start HOMER** (see chap. 1.2, p. 109)
2. **Select HOMER scenario** (see chap. 1.3, p. 109)

   The structure of the acquisition tables is described in the HOMER scenario, i.e. which data can be acquired.

3. **Transform ADL data to Excel sheets** (see chap. 1.4, p. 110)

   (The existing ADOxx model will be exported into an ADL file (see chap. 6.2, p. 468), which will be transformed again to an Excel table.

4. **Create acquisition tables** (see chap. 1.6, p. 118)

   (The created acquisition table can be created either empty or with the data from an already existing acquisition table.)

5. **Fill in acquisition tables**

6. **Consolidate acquisition tables** (see chap. 1.7, p. 121)

   By consolidating acquisition tables, you will bring together several acquisition tables with collected data into one and, respectively, the newly created model (on the basis of the consolidated acquisition table) into one acquisition table.

7. **Transform Excel sheets to ADL data** (see chap. 1.5, p. 115)

   The ADL data created from the Excel acquisition tables will be imported into ADOxx (see chap. 6.1, p. 445) and will replace the respective data in the existing models or in the newly created ones.

1.2 Start HOMER

Hint: HOMER must have been duly installed on your computer. If necessary, refer to the HOMER installation description in the volume V of the ADOxx user documentation ("Installation and Database administration").

You can call HOMER from the acquisition component level of the ADOxx Modelling Toolkit or directly, by the Window start menu ("Programs" menu item - "HOMER 3.7" submenu). A proper installation of Microsoft Excel, as well as the correctness and the completeness of the Microsoft Excel system paths and register entries are required to use HOMER. You can use Microsoft Excel from the Version 97 / 8.0

To start HOMER from the ADOxx Modelling Toolkit, move to the Acquisition Component and click on the icon , or select in the **"Table"** menu the **"Acquisition tables"** menu item.

Start HOMER over Windows, by selecting in the "Start" menu the "Programs" menu item, the submenu with your ADOxx Version and then select the program file "HOMER" and the program "HOMER".

Before starting the data acquisition, you should define structure of the HOMER acquisition tables and thus, the attributes to acquire (see chap. 13.2.2.10, p. 777). This simplifies the subsequent Evaluation of the data. After having filed in the HOMER acquisition tables, you can take them into the ADOxx Modelling Toolkit.

1.3 Select HOMER Scenario

A HOMER scenario describes the settings of an acquisition table (see chap. 13.2, p. 768) for a certain use (e.g. acquisition of the execution time, acquisition of the documents used). If you want to work with the acquisition tables, you must previously select the appropriate scenario.

Select in the menu **"Administration"** the menu item **"Scenario manager"**.

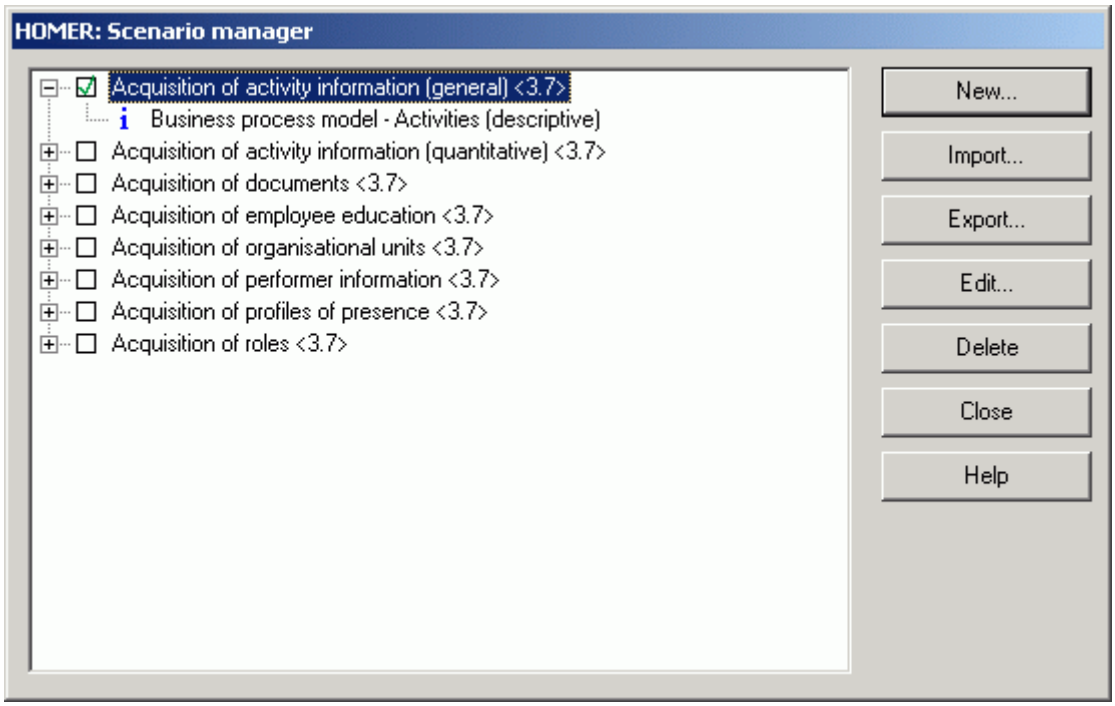


Figure 79: HOMER: Scenario manager

You will carry out the **scenario selection** by clicking on, and thus activating, the checkbox of the appropriate scenario. An activated scenario is characterised by the symbol ☑ and appears in the status bar of HOMER.

Hint: The loading of the activated scenario can take some seconds.

The buttons "New", "Import", "Export", "Edit" and "Delete" help you administrating the scenarios (see chap. 13.1, p. 763).

1.4 Transform ADL Data to Excel Sheets

Transform an ADL data to Excel sheets by selecting in the **"File"** menu the **"Transform ADL to Excel-Sheets"** menu item. In the following window (see fig. 80) you can define the two files as well as the other transformation options.

Hint: Only the attributes which are defined in the table structure (see chap. 13.2, p. 768) will be transformed in HOMER .

Tab **"ADL files settings"**:

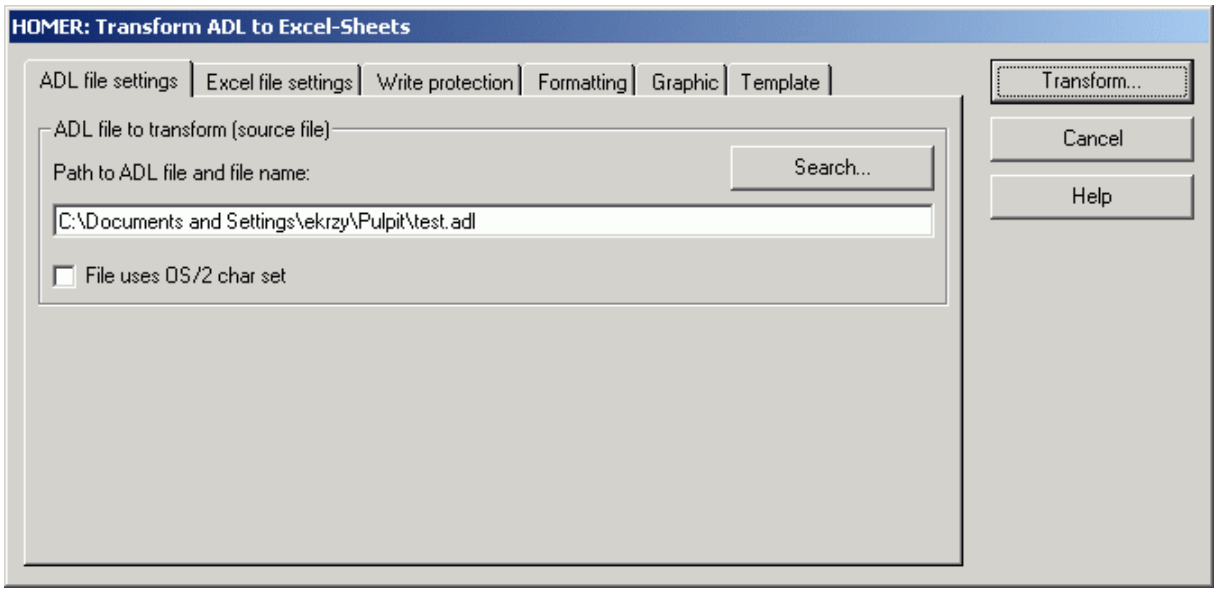


Figure 80: HOMER: Transform ADL files to Excel sheets - ADL file settings

Enter in the **"Path to ADL file and file name"** field the path and the name of the ADL file, which you want to import. Clicking on the button **"Search"** will provide you with support.

ATTENTION: Always provide the absolute (full) path information.

Activate the **"File uses OS/2 char set"** option if you want to import am ADL file, which has been exported from ADOxx for OS/2.

Tab **"Excel file settings"**:

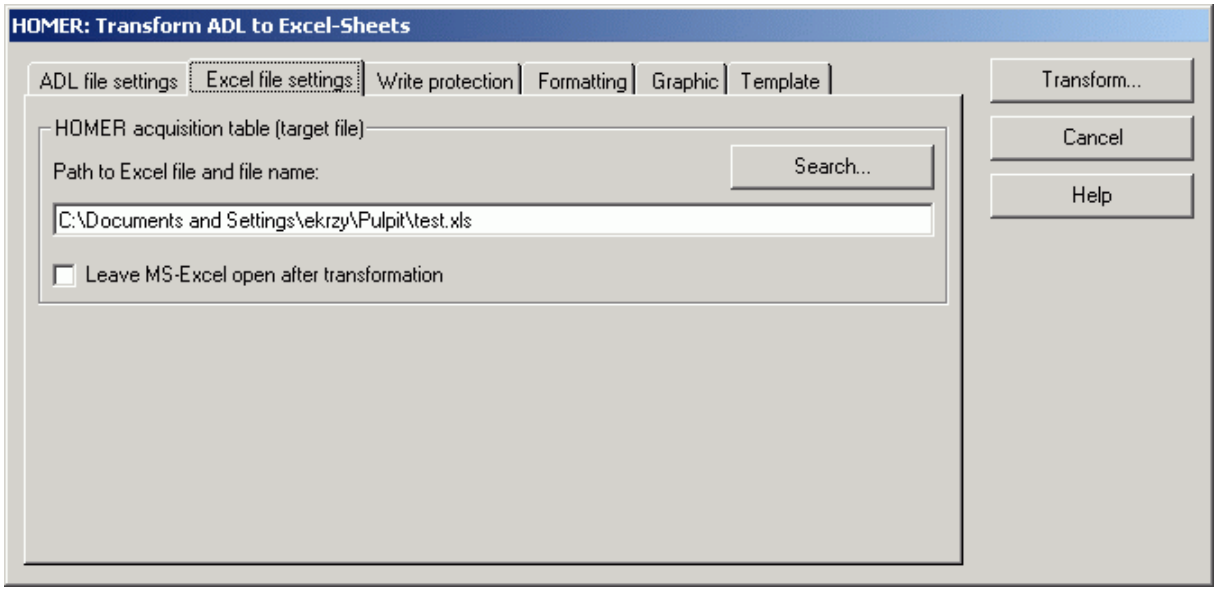


Figure 81: HOMER: Transform ADL data to Excel sheets - Excel file settings

To save the Excel file, provide the path and the name of the Excel file in the field **"Path to Excel file and file name"**. Clicking on the **"Search"** button will provide you support.

ATTENTION: Always provide the absolute (full) path information.

Activate the **"Leave MS-Excel open after transformation"** option if you want to view the table in Excel directly after the transformation.

Tab **"Write protection"**:

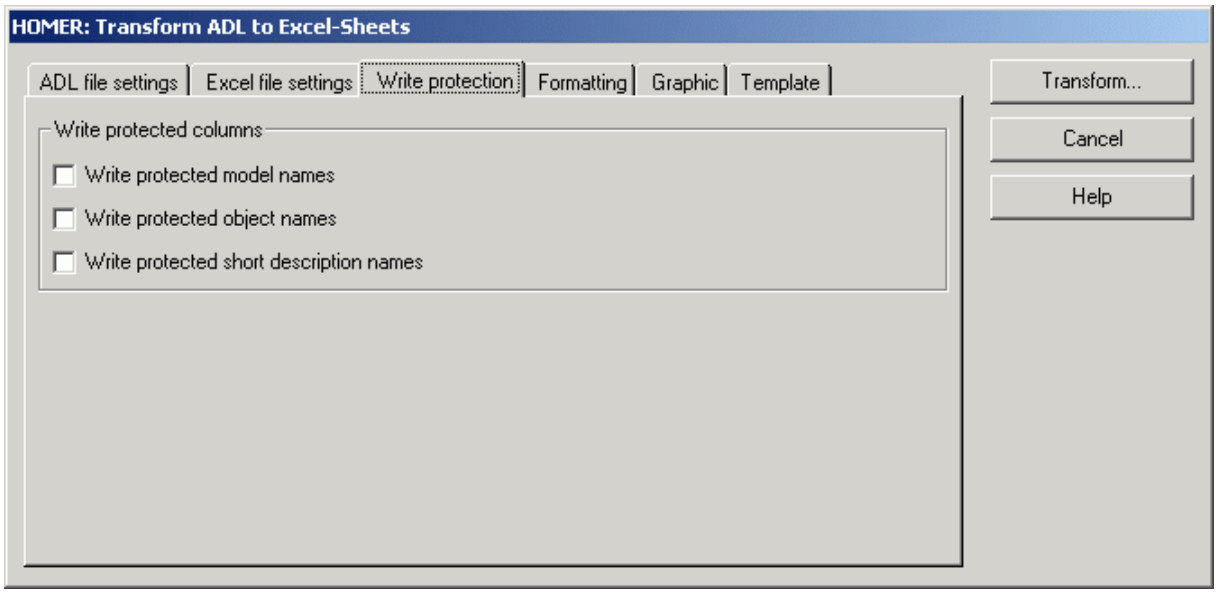


Figure 82: HOMER: import ADL - write protection

Within this register you can define whether the cells for the model name, the object name and the short name should be set write protected.

Hint: The write protection can be cancelled by selecting in Microsoft Excel "Tools" menu the "Protection" menu item and then the "Unprotect sheet" submenu. To protect the write protection with a password, import the appropriate cell into Excel as write protected, cancel the sheet protection in Microsoft Excel by selecting the "Unprotect sheet" ("Tools" menu, "Protection" menu item), select the "Protect sheet" submenu ("Tools" menu, "Protection" menu item) and enter the proper password in the "Password" field .

Tab **"Formatting"**:

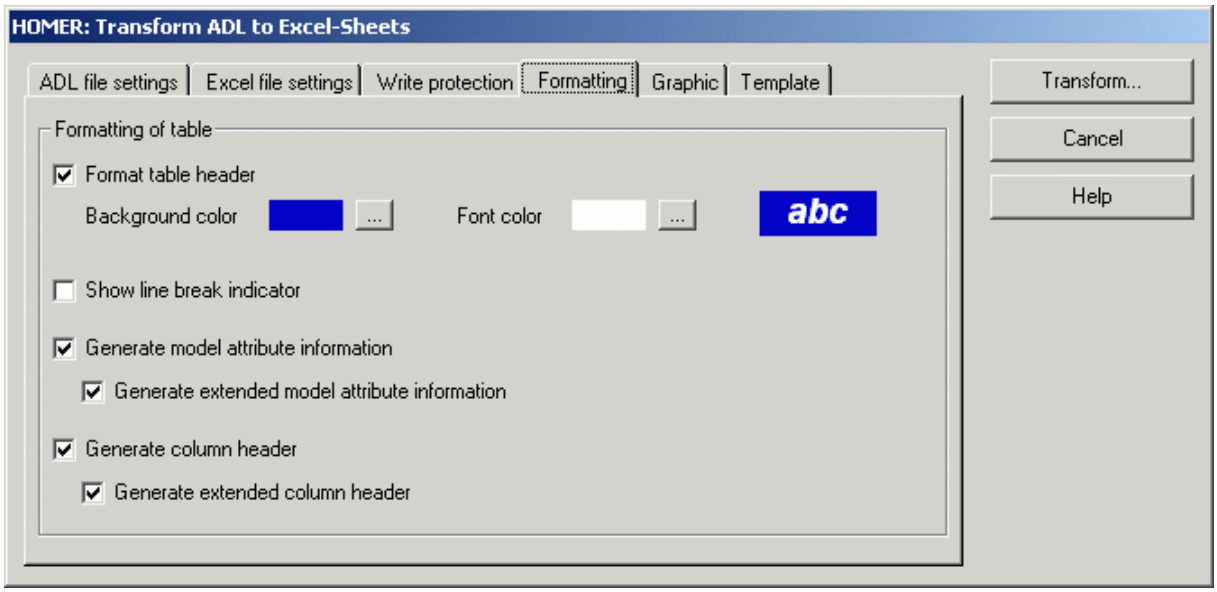


Figure 83: HOMER: transform ADL files to Excel sheets - Formatting

Format table header
creates a table with a formatted header (rows until the first object). You can define a **background colour** and **Text colour** using a colour selection dialogue; the default **font** is set to Arial 12pt if the option is enabled.

Create column for line break indicator
will show the column for the line break indicator for multi-line attributes.

Create model attribute information
Adds the attribute name to the model attributes.

Show extended model attribute information
adds the attribute type as well as the text "(Model attribute)" for all model attributes.

Create column headings
adds a header for the object name, short description and attribute columns.

Create extended column headings
adds column headings with increased information for attributes, consisting of attribute name, attribute type and the object name the attribute belongs to.

Tab **"Graphic"**:

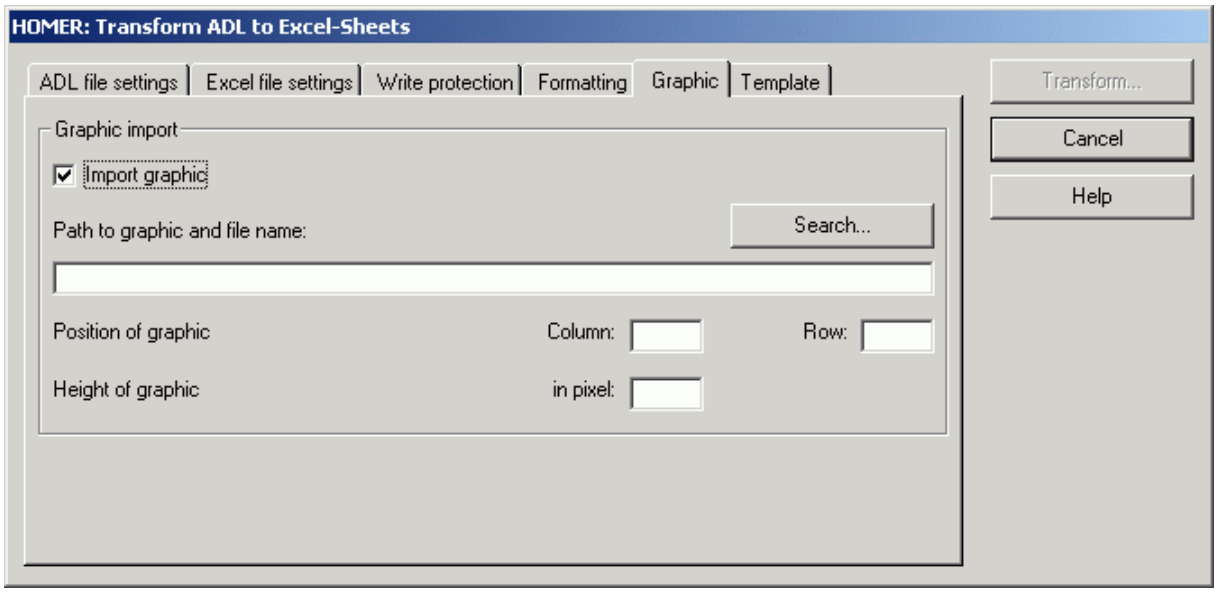


Figure 84: HOMER: transform ADL files to Excel sheets - Graphic

This is useful if you wish to integrate a graphic (e.g. company logo) in a defined cell of the generated Excel sheet. You can use the following graphic formats:

You can import the following graphic formats:

- Windows Bitmap (BMP)
- Device Independent Bitmap (DIB)
- Windows Meta File (WMF)
- Windows Enhanced Meta File (EMF)
- Icon File (ICO)
- Cursor File (CUR)

Enter in the **"Path to graphic and file name"** field the path and the name of the graphic file, which you wish to import. Clicking on the "Search" button will provide you support.

ATTENTION: This option is only meant to import **small** graphics. You should not import model graphics, as this could lead to main memory problems in Microsoft Excel.

ATTENTION: Always provide the absolute (full) path information.

Position of the graphic

Column and **row** of the cell you want to import the graphic to.

Height of graphic in pixels

The desired height of the graphic. The maximum height for graphics is 397 pixel; we recommend 90 pixel.

Hint: The colour settings and the option **"Import graphic"** with all their properties will only be available when the option **"Format table"** has been activated.

Tab **"Master"**:

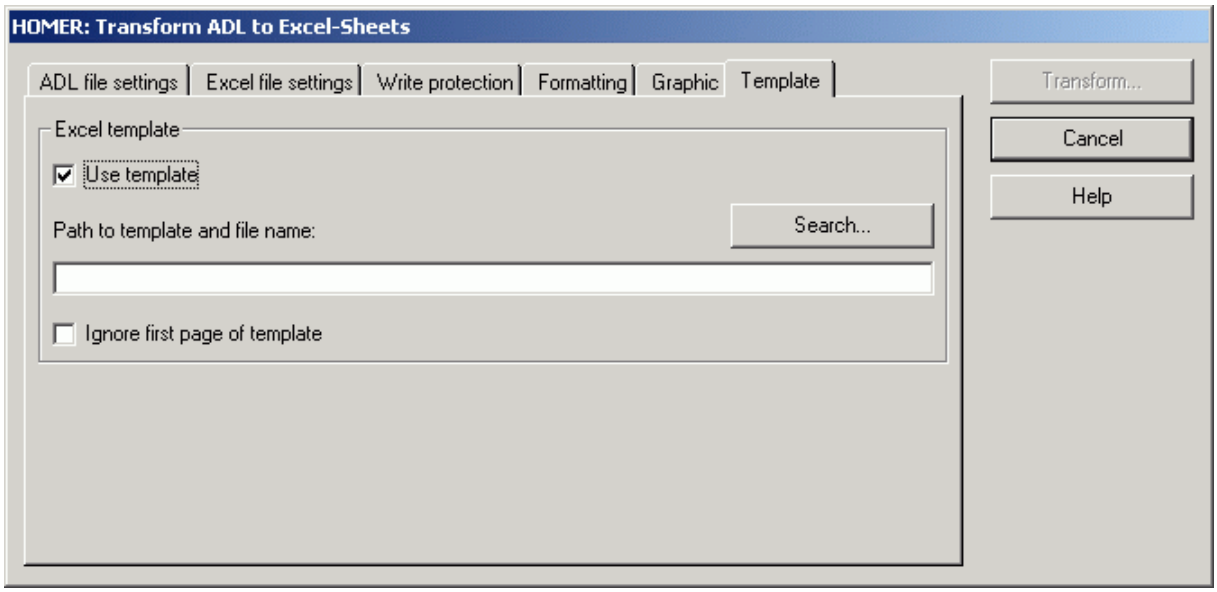


Figure 85: HOMER: Transform ADL files to Excel sheets - Model

Activate the option **"Use masters"** if you want to use an already existing acquisition table as a model for the formatting.

> **Hint:** When you activate the option **"Use masters"**, the option "Format table" in the register "Formatting" will automatically be deactivated (see fig. 83), since the formats will be taken over in the model.

Enter in the field **"Path to master and file name"** the path and the name of the excel file, which you want to use as model. Clicking on the **"Search"** button will provide you support.

> **ATTENTION:** Always provide the absolute (full) path information.

Activate the option **"Ignore first page of master"** to ignore the first page of the Excel table. This can be useful for instance when the first page contains additional explanatory information or generated statistics.

Click on the button **"Transform"**. In the following window select the models ( ) and attribute profiles ( ) to be transformed.

The successful transformation will be confirmed with a message.

1.5 Transform Excel Sheets to ADL Files

Transform Excel sheets to ADL files, by selecting the "Transform Excel-Sheets to ADL" menu item in the "File" menu. The "HOMER: ADL Export" window (see fig. 87) will appear, in which you can carry out the following settings and then start the transformation by clicking on the "Transform button".

> **Hint:** HOMER can only transform attributes, which are defined in the table structure (see chap. 13.2, p. 768).

Tab **"Excel file settings"**:

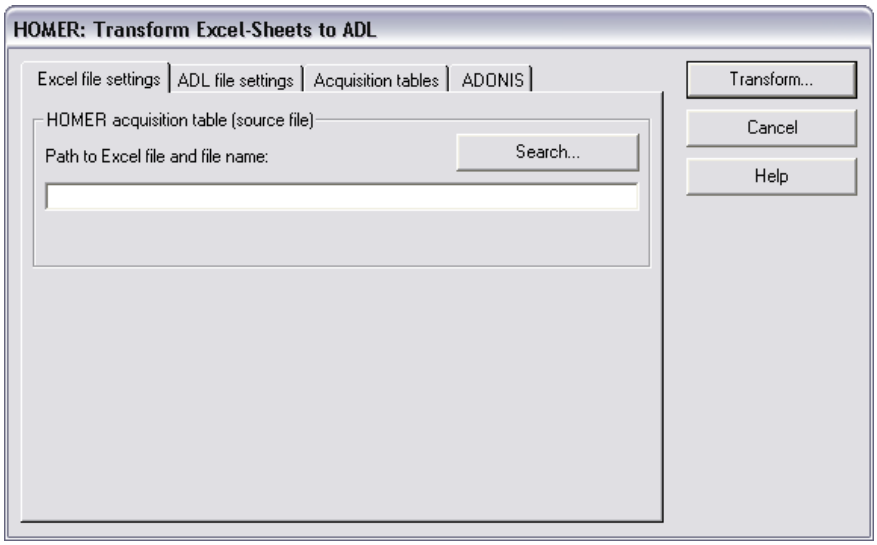


Figure 86: HOMER: Transform Excel sheets to ADL files - Excel file settings

Enter in the **"Path to Excel file and file name"** field the path and the name of the Excel file you want to export. Clicking on the "Search" button will provide you support.

ATTENTION: Always provide the absolute (full) path information.

Tab **"ADL file settings"**:

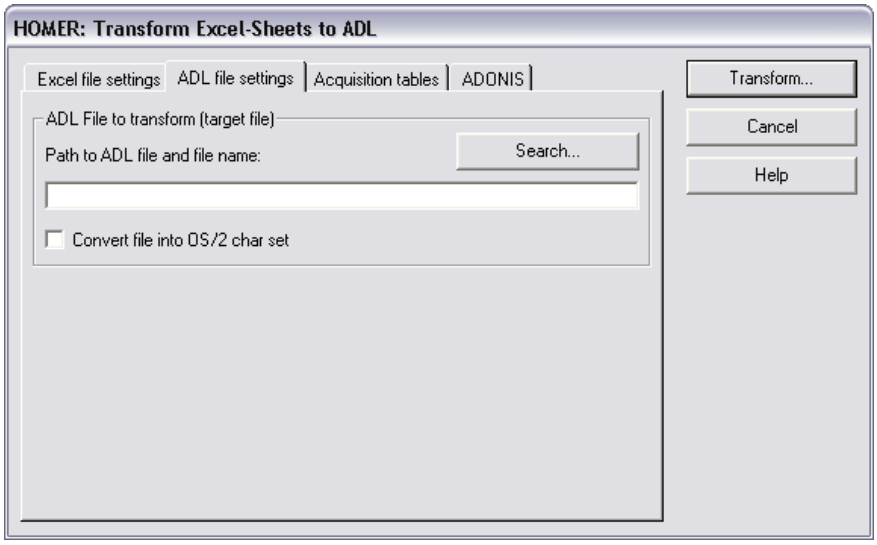


Figure 87: HOMER: transform Excel sheets to ADL files - ADL file settings

Enter in the **"Path to ADL file and file name"** field, the path and the name of the ADL file, which you want to create. Clicking on the "Search" button will provide you support.

ATTENTION: Always provide the absolute (full) path information.

Activate the **"Convert file into OS/2 char set"** option to create an ADL file, which should be imported to ADOxx for OS/2.

Tab **"Acquisition tables"**:

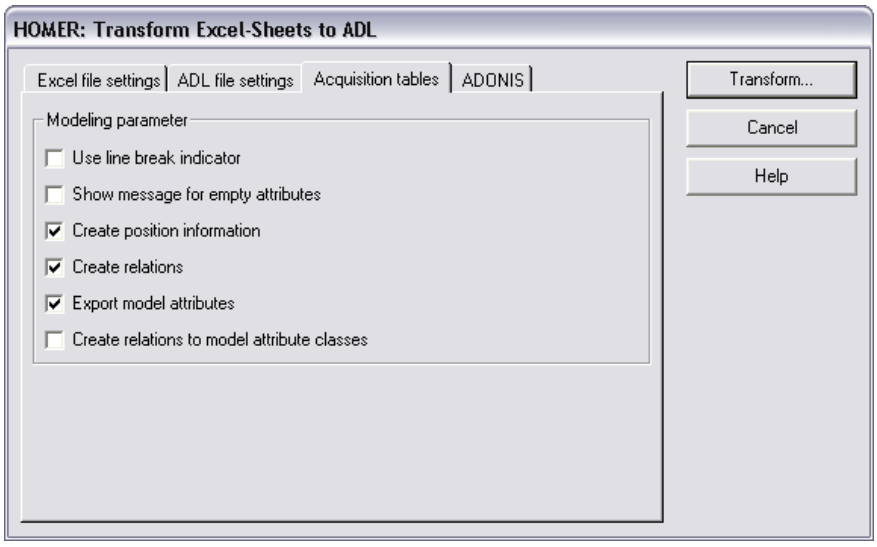


Figure 88: HOMER: Transform Excel sheets to ADL files - Acquisition tables

Activate the **"Use line break indicator"** option to use the line break indicator for attributes of the type "several lines long string".

Activate the **"Show message on empty attributes"** option to display all attributes containing no value.

Activate the **"Create position information"** option when you want to arrange objects in a model you have created.

ATTENTION: If you want to take over acquisition information in already existing models, the "Create position information" must be **deactivated**.

Activate the **"Create relations"** option if you want to connect all the objects of a new model with the relations defined in your settings .

ATTENTION: If you want to take over acquisition information in already existing models, the "Create relations" must be **deactivated**.

Activate the **"Export model all attributes"** option if you want to take over model information in an ADL.

Activate the **"Create relations to model attribute classes"** option if you want to connect all the objects of a model attribute class of a new model with the rest of the objects using the relation defined in your settings.

Hint: If you have not defined any relation in your settings, then the option is inactive.

Tab **"ADOxx"**:

Hint: The "ADOxx" tab will only be available, once you have chacked the checkbox "Create position information" (tab "Acquisition tables") .

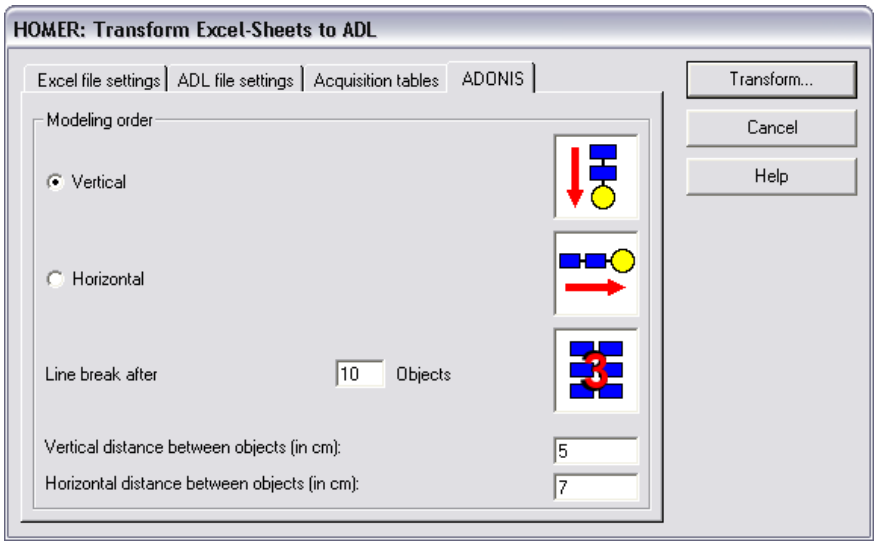


Figure 89: HOMER: Transform Excel sheets in ADL files - ADOxx

Select the **"Vertical"** setting, if you want to arrange the objects to be exported vertically.

Select the **"Horizontal"** setting, if you want to arrange the objects to be exported horizontally.

Enter in the **"Line breaker after ... objects"** field the number of objects after which a new "row/column" should start in the ADOxx model.

Enter in the **"Vertical distance between objects (in cm)"** and **"Horizontal distance between objects (in cm)"** fields the distance between the objects to be exported.

Finally, click **"Transform"**. In the dialogue window following, select the models ( ) and attribut profiles ( ) to be transformed.

A successful transformation will end with an appropriate message.

1.6 Creating Acquisition Table

If you want to create acquisition tables, select in the "File" menu the **"Create acquisition tables"** menu item. The "HOMER: Acquisition tables" window (see fig. 90) will then be displayed, in which you will carry out the following described settings and then create the tables by clicking on the OK button.

Tab **"General settings"**:

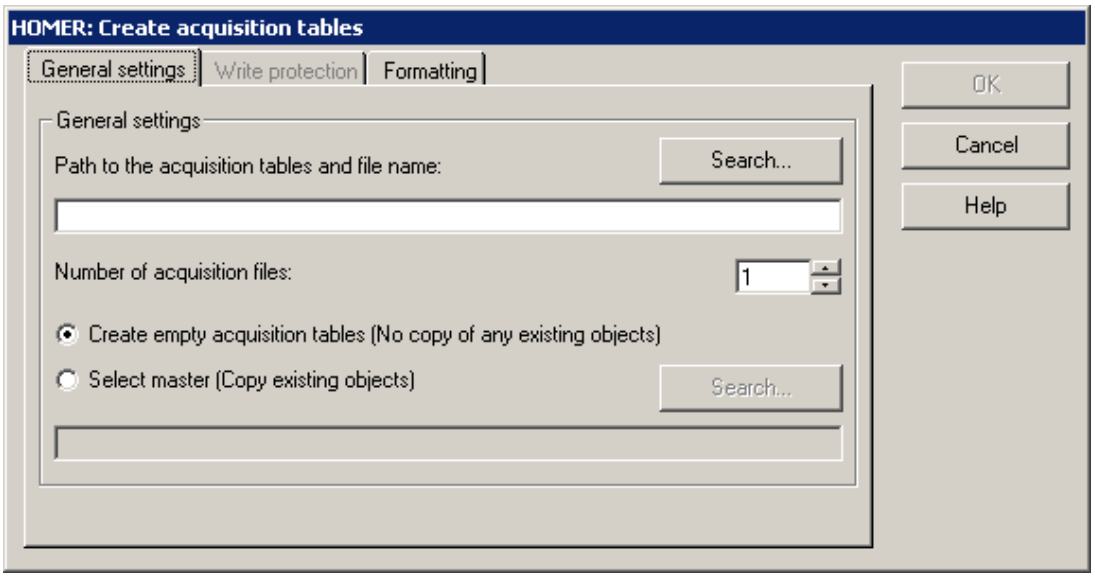


Figure 90: Create acquisition table - General settings

Enter in the **"Path to acquisition tables and file name"** field the path and the name of the acquisition tables to be created. Clicking on the **"Search"** button will provide you with support.

ATTENTION: Always provide the absolute (full) path information.

Enter in the **"Number of acquisition files"** field the number of files with acquisition tables, which you wish to generate.

Hint: You can create a maximum number of 300 acquisition files in one go.

Select the **"Create empty acquisition tables (No copy of any existing objects)"** setting, if you want to create empty acquisition tables (without objects and attributes), or select the **"Select master (copy existing objects)"** setting and enter the path for a workbook, if you want to use one of these as master for the acquisition tables you wish to create. Clicking on the **"Search"** button will provide you with support.

Tab **"Write protection"**:

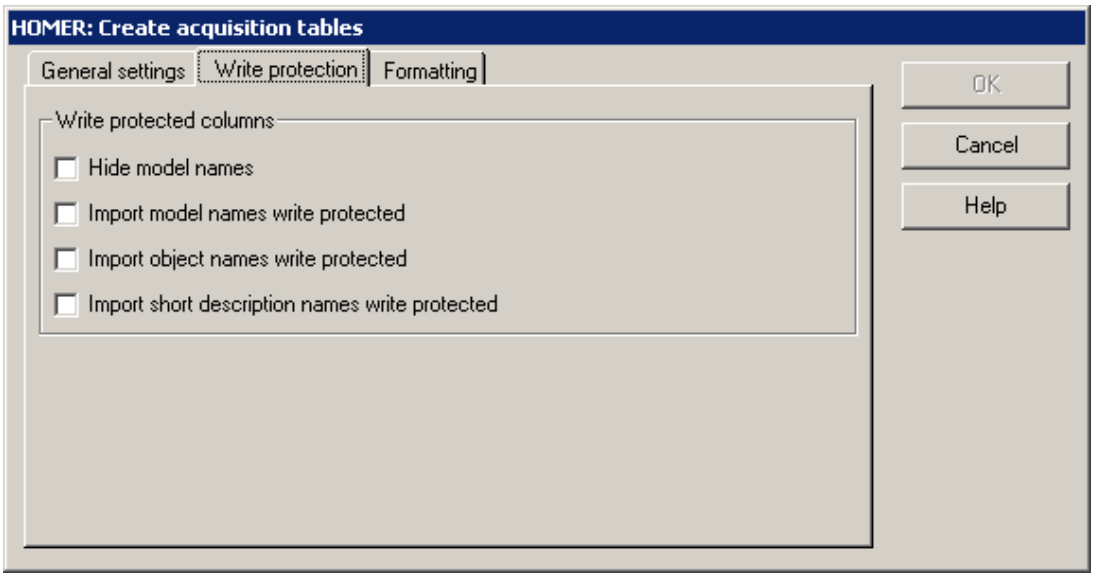


Figure 91: Create acquisition tables - Write protection

Within this tab you can define whether the cells for the model name, the object name and the short name shall be set write-protected. Furthermore, the cell containing the model name can be made invisible for the user.

Hint: The write protection can be cancelled by selecting in Microsoft Excel "Tools" menu the "Protection" menu item and then the "Unprotect sheet" submenu. To protect the write protection with a password, import the appropriate cell write protected, cancel the sheet protection in Microsoft Excel by selecting the "Unprotect sheet" ("Tools" menu, "Protection" menu item), select the "Protect sheet" submenu ("Tools" menu, "Protection" menu item) and enter a password in the "Password" field .

Hint: The options described above are only available if the setting "Select master (copy existing objects)" has been selected in the "General settings" (see fig. 90).

Hint: The write-protection options are particularly recommended in a distributed acquisition environment to ensure that the most necessary and elementary data of the acquisition tables are not changed or deleted by mistake.

Tab **"Formatting"**:

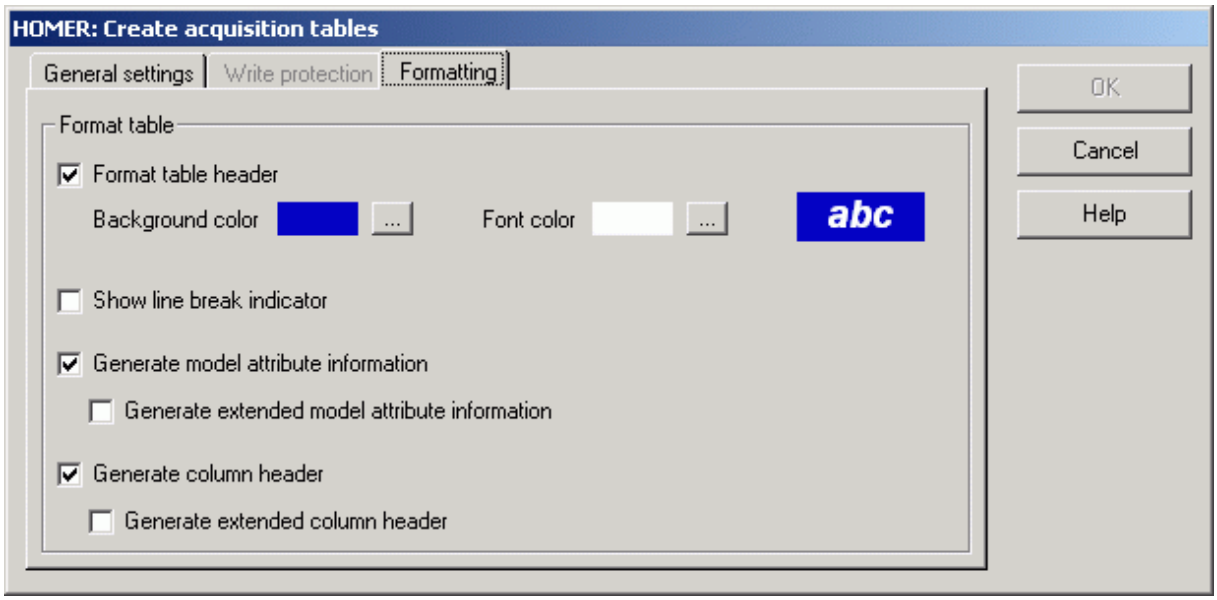


Figure 92: Create acquisition tables - Formatting

Format table header
creates a table with a formatted header (rows until the first object). You can define a **background colour** and **Text colour** using a colour selection dialogue; the default **font** is set to Arial 12pt if the option is enabled.

Create column for line break indicator
Will show the column for the line break indicator for multi-line attributes.

Create model attribute information
adds the attribute name to the model attributes.

show extended model attribute information
adds the attribute type as well as the text "(Model attribute)" for all model attributes.

Create column headings
adds a header for the object name, short description and attribute columns.

create extended column headings
adds column headings with increased information for attributes, consisting of attribute name, attribute type and the object name the attribute belongs to.

Hint: The options described above are only available if the setting "Select master (copy existing objects)" has been selected in the "General settings" (see fig. 90).

Start the creation of the acquisition tables by clicking on the OK button.

Hint: After a successful creation you will be shown a message, e.g. that the files "C:\xy(#).xls" have been created. Replace the sign (#) with an index number. If you have chosen "3" for the number of acquisition tables, the files "C:\xy(01).xls", "C:\xy(02).xls" and "C:\xy(03).xls" have been created.

1.7 Consolidating Acquisition Tables

During (distributed) data acquisition it may become feasible to consolidate several complete acquisition tables in order to obtain representative average values.

If you want to consolidate acquisition tables, select in the **"File"** menu the **"Consolidate acquisition tables"** menu item. The "HOMER: Consolidate acquisition tables" window (see fig. 93) will then be

displayed, in which you will carry out the following described settings and then consolidate the tables by clicking on the OK button.

Register **"File settings"**:

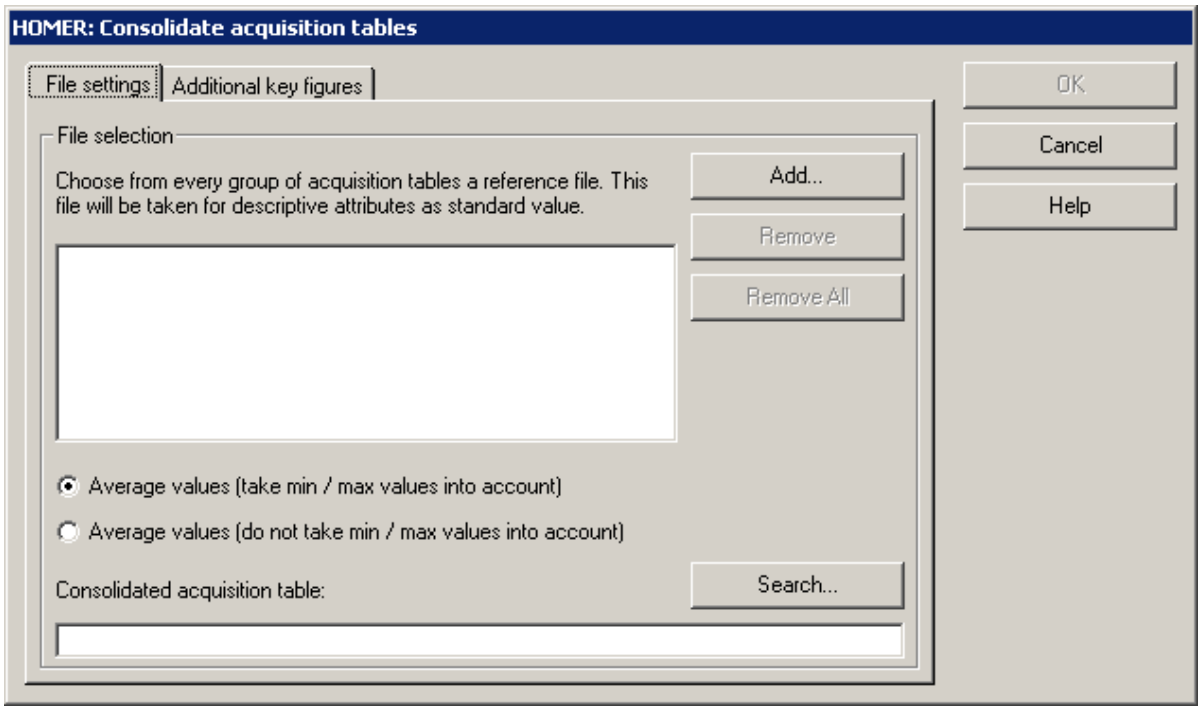


Figure 93: HOMER: Consolidate acquisition tables - file settings

Add to the "list of acquisition tables" the acquisition tables which you want to consolidate by clicking the **"Add" button**.

Choose from every group of acquisition tables a reference file. This file will be taken as a template for *descriptive attributes*; these are attributes which are not (cannot be) consolidated. Should several of these descriptive attributes "collide" (since several tables will be integrated into a single one), the values from the selected "Reference file" will be used.

Descriptive attributes are:

- Strings
- Enumeration
- Enumeration list
- Model reference
- Object reference
- Attribute profile reference
- Program call
- Expression

Evaluate-able attributes are:

- Number
- Time

- Discrete random generator
- Date
- Date and time

Hint: The attributes to be evaluated must be defined in the table structure (see chap. 13.2, p. 768).

ATTENTION: The structure of the tables must be absolutely identical. The scenario selected in the Scenario manager (see chap. 1.3, p. 109) will be used for the Evaluation. There shall be no discrepancy in the object names, discrete variables forms etc.

Hint: In the following, **"Sum1"** means that when building the sum, the maximum and minimum value are taken into account, i.e. **all** attributes will be added.

Hint: In the following, **"Sum 2"** means that when building the sum, the maximum and minimum value are not taken into account, i.e. all attribute values **less the largest and the smallest** will be added.

Select the **"Average values (take min/max values into account)"** setting, if you want to obtain as average value of an attribute the sum of all attribute values divided by the number of attribute values .

Formula: `AvV1 = (Sum1 [AttV]) ÷ n1`

Select the **"Average values (do not take min/max values into account)"** setting, if you want to obtain as average value of an attribute the sum of all attribute values, less the largest and the smallest, divided by the number of attribute values minus two .

Formula: `AvV2 = (Sum2 [AttV]) ÷ n2`

Tab **"Additional key figures"**:

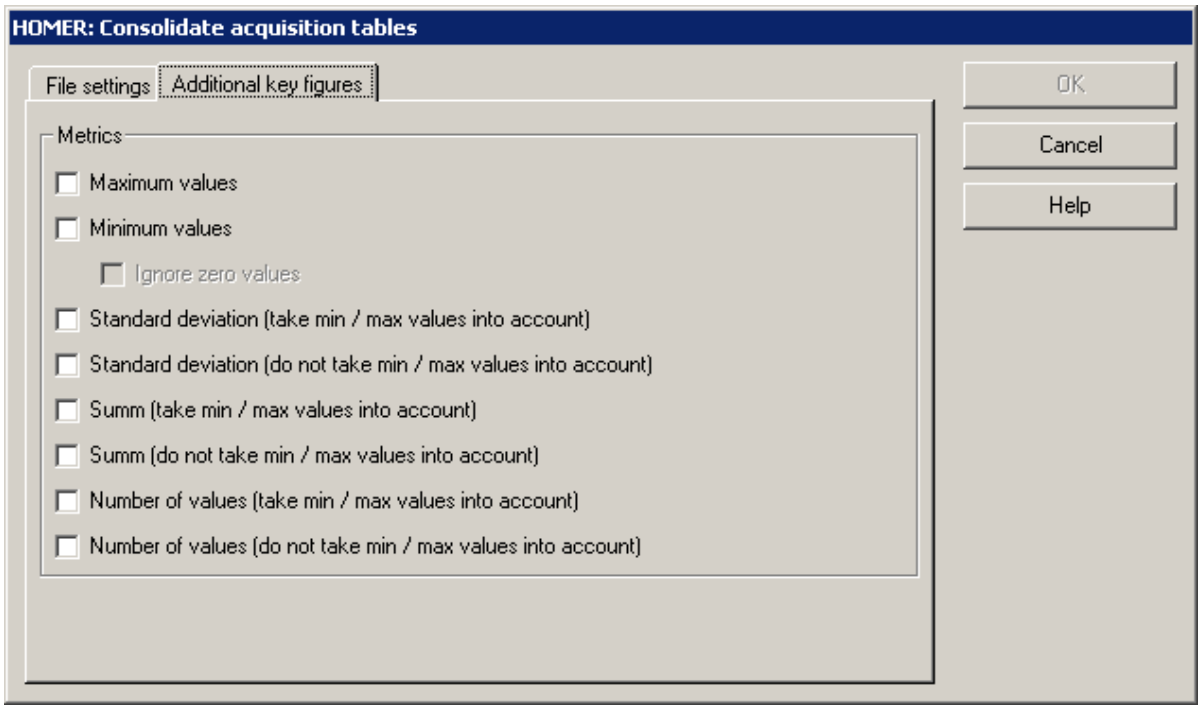


Figure 94: HOMER: Consolidate acquisition tables - Additional key figures

Activate those values you want to list in additional statistical evaluation tables:

Maximum value:
The highest attribute value (Maximum).
Formula: `MAX = Max {AttV1; ..; AttVn}`

Minimum value (without "Ignore zero values"):
The smallest attribute value (Minimum)
Formula: `MIN = Min {AttV1; ..; AttVn}`
Hint: The present minimum value will also be taken into account during the average calculation, i.e. if you have clicked the "average values (do not take min/max values into account)" option, the present value will be subtracted from the sum of all attribute values.

Minimum value (with "Ignore zero values")
The smallest attribute value (Minimum), but ignoring all attributes with the value zero (0)
Formula: `MIN = Min {NOT [AttV1 = 0]; ..; NOT [AttVn = 0]}`

Hint: The present minimum value will also be taken into account during the average calculation, i.e. if you have clicked the "Average values (do not take min/max values into account)" option, the present value will be subtracted from the sum of all attribute values.

"Standard deviation (take min/max values into account)"
Standard deviation of all attribute values
Formula: `StD = Sqrt ( [ {n1 × Sum1 (Quad [AttV] ) } - {Quad (Sum1 [AttV] ) } ] ÷ [n1 × [n1 - 1] ] )`

"Standard deviation (do not take min/max values into account)"
Standard deviation of all attribute values excluding the smallest (minimum) and highest (maximum) attribute values
Formula: `StD = Sqrt ( [ {n2 × Sum1 (Quad [AttV] ) } - {Quad (Sum1 [AttV] ) } ] ÷ [n2 × {n2 - 1} ] )`

"Sum (take min/max values into account"
Sum of all attribute values

Formula: `SUM1 = [AttV1 + AttV2  + .. + AttVn]`

"Sum (do not take min/max values into account"

Sum of all attribute values excluding the smallest (minimum) and highest (maximum) attribute values

Formula: `SUM2 = [AttV1 + AttV2 + .. + AttVn] - [Min {AttV1; ..; AttVn}] - [Max {AttV1; ..; AttVn}]`

Number of collected values (take min/max values into account)"

Formula: `N1 = n`

Number of collected values (do not take min/max values into account)"

Formula: `N2 = n - 2`

Start the consolidation by clicking on the OK button.

2. Modelling

2.1 Introduction

The modelling of business processes and working environments plays a central role within ADOxx. **Modelling** is **the basis for further work** on your Business Processes and Working Environments.

Activating the Modelling Component:

There are 3 possibilities to start the "modelling" component:

- By clicking on the smart icon in the component bar
- By clicking with the right mouse button on or beside the component list and then selecting the menu item "modelling" from the popup menu, which appears
- By pressing the hot key **<F9>** followed by the button **"M"**.

Once the Modelling Component has been activated, the following icons are displayed in the quick-access bar:

Model Management:

"Create new model" (see chap. 2.4.1, p. 256)
"Open an existing model" (see chap. 2.4.2, p. 257)
"Save model" (see chap. 2.4.4, p. 259)
"Print model" (see chap. 2.4.17, p. 288)

Model Navigation:

"Models in the database" (see chap. 4.1, p. 56)
"Opened model windows" (see chap. 4.2, p. 57)
"Previous model window" (see chap. 2.3.1, p. 255)
"Next model window" (see chap. 2.3.1, p. 255)

Model Editing:

"Undo" (see chap. 2.2.7.1, p. 137)
"Redo" (see chap. 2.2.7.2, p. 137)
"Cut" (see chap. 2.2.7.16, p. 150)
"Copy" (see chap. 2.2.7.15, p. 149)
"Paste" (see chap. 2.2.7.17, p. 151)
"Find" (see chap. 2.2.7.19, p. 152)
"Global change" (see chap. 2.2.14, p. 243)
"Drawing area" (see chap. 2.2.9, p. 176)
"Generate graphics" (see chap. 2.2.7.24, p. 156)

Model View:

"Graphical model view" (see chap. 2.2, p. 127)

"Tabular model view" (see chap. 2.2.10, p. 181)

"Zoom" (see chap. 2.2.11.3, p. 189)

"Scale 1:1" (see chap. 2.2.11.3, p. 189)

"Show all" (see chap. 2.2.11.3, p. 189)

"Snap grid - active/inactive" (see chap. 2.2.11.4, p. 190)

"Snap grid - visible/invisible" (see chap. 2.2.11.4, p. 190)

Model Valuation:

"Time and Costs" (see chap. 2.6, p. 303)

Hint: The appearance of the smart icons in the quick-access list depends on the particular configuration (see chap. 4., p. 54) of the Modelling Toolkit. ADOxx administrators can modify the quick-access bar ad lib.

Hint: For any additional information and explanation in the chapter please refer to use **ADOxx-Default-Library** (see chap. 16., p. 802).

2.2 The Model Editor

The model editor is used to create and work on ADOxx models. Each ADOxx model is displayed in a Model window (see chap. 2.2.3, p. 130) and contains objects and connectors.

The model Editor is a graphical Editor. It is necessary to use the mouse to work with the model Editor. To facilitate the input of data a Tabular Model View (see chap. 2.2.10, p. 181) is also available.

2.2.1 The Working Area of the Model Editor

The Working area (see chap. 2.5, p. 38) of the model editor is that part of the ADOxx window "ADOxx: Modelling Toolkit (<*Username*>)" (see fig. 95), which is below the Component bar (see chap. 2.3, p. 37) and when the Modelling Toolkit is first started is shown as a grey area.

When a new model is created or an existing model opened, it is displayed in a model window on the working area.

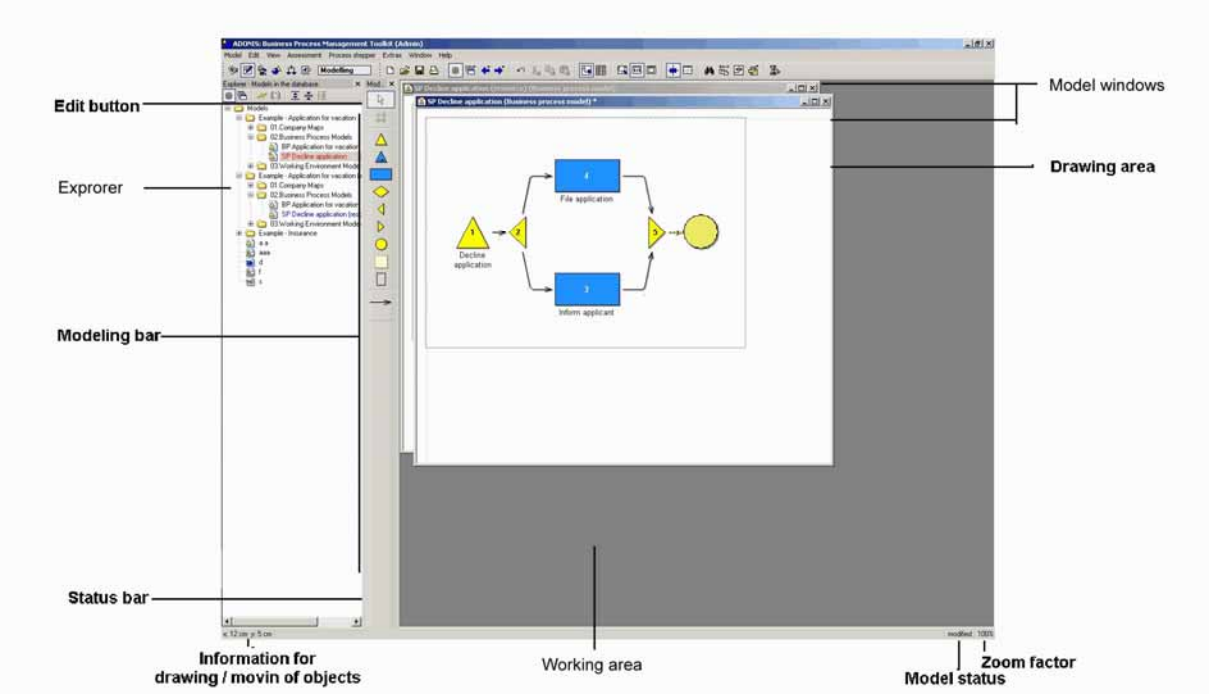


Figure 95: Working area of the model editor

Once a model is open, the modelling bar (see chap. 2.2.2, p. 128) is displayed on the left hand side of the working area.

The **Class panel** contains the objects and relations, available for modelling. The content of the class panel depends on the application library used as well as the model type which you are currently working on. By selecting a different view on the model through the option "Mode" (see chap. 2.2.11.1, p. 186) in the "View" menu, the objects available in the class panel can be increased or reduced.

The **Status row** displays in each activated model window information such as the coordinates during the drawing and the moving of objects, the view mode or the model state (e.g. the model has been altered since the last storage) . The co-ordinates show the distance from the left (x-value) top (y-value) corner of the drawing area (see chap. 2.2.9, p. 176). On the right side of the status row, there are three status fields. The left field shows the scale of the current model. The field in the middle shows whether the model has been altered since it was last saved. If it has been altered, the word "modified", is shown in this field, otherwise it is empty. The field on the right (expression monitor) shows how many attributes of the type "expression" still have to be updated.

2.2.2 Modelling Bar

The modelling bar is by default located to the left of the working area. It contains the "empty space tool" (see chap. 2.2.9.2, p. 178), the class panel (see chap. 2.2.2.1, p. 129) and the relation panel (see chap. 2.2.2.2, p. 129). The modelling bar can be shown/hidden via menu **"Window"** - entry **"Tools"** - **"Modelling"** or via the hot key **<Ctrl>+M**.

While the "Edit" icon and the "empty space tool" are always visible, however depending on the space available (e.g. if the ADOxx window is not at its maximum size) only parts of the class and relation panels may be visible. When this occurs, the arrow icons and can be used to scroll through the contents.

2.2.2.1 Class Panel

The **Class panel** contains all the classes available for modelling.

> **Hint:** The objects displayed in the class panel depend on the Mode (see chap. 2.2.11.1, p. 186) currently selected.

The picture below shows the class panel of the **ADOxx-Default-Library** for all model types (see fig. 96).

Figure 96: Class panel of the ADOxx-Default-Library

2.2.2.2 Relation Panel

The **Relation panel** contains all the relations, available for modelling.

> **Hint:** The relations displayed in the relation list depend on the current Mode (see chap. 2.2.11.1, p. 186) selected.

The picture below shows the relation panel for the **ADOxx-Default-Library** for all model types (see fig. 97).

Figure 97: Relation panel of the ADOxx-Default-Library

2.2.3 Model Window

A model window (see fig. 98) consists of a window bar and a drawing area.

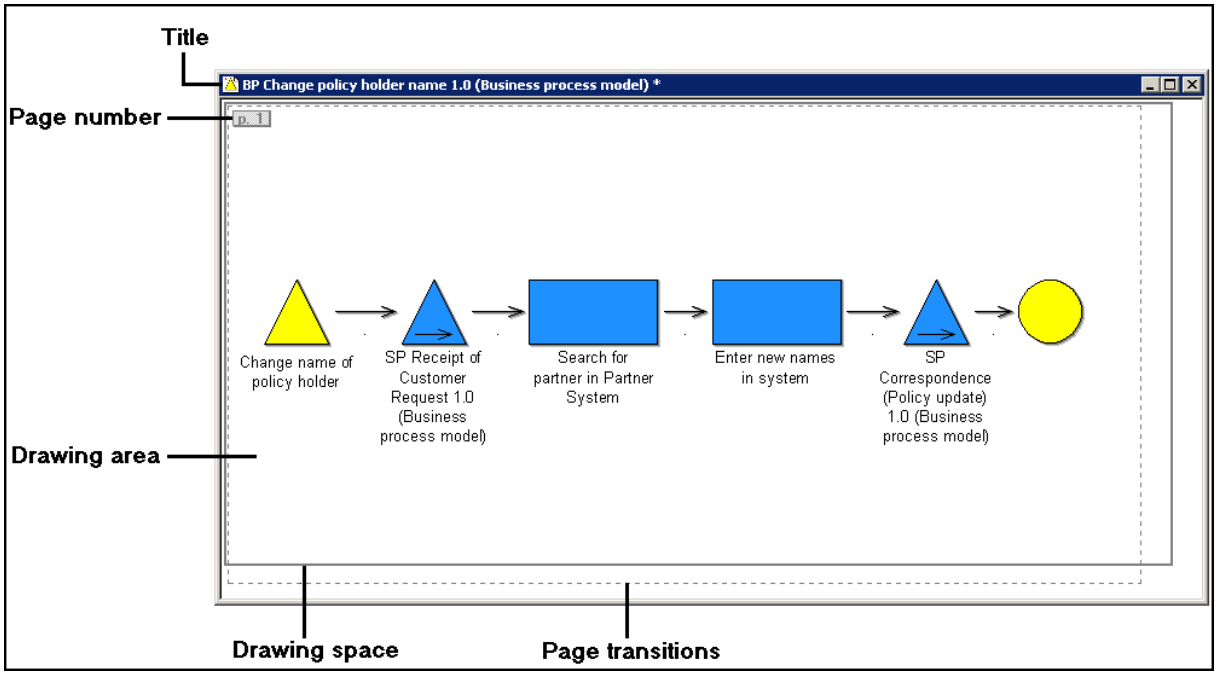


Figure 98: Model window

The **Window bar** of each model window contains the exit icon, the name and type of the model as well as the buttons to minimize, maximize and close the window.

The appearance of the exit icon depends on the model type. For example of a business process model it looks like this and for a working environment model like this .

When you change a model the exit icon is highlighted with a red star until the changes have been saved in the model. Similarly an asterisk "*" is added to the title bar of the model window.

The **Drawing area** is used to create and work on your models.

A grey rectangular border shows you the size of the actual **Drawing area** (see chap. 2.2.9, p. 176). The drawing area is at least 10cm×10cm in size and will grow in size to include all the objects and relations and to print the entire model on a graphical printer. Placing an object below or to the right of the current drawing area can increase the size of the drawing area. You can also pull the rectangular borderline with the mouse in order to increase/decrease the size of the drawing area.

The **Page borders** (dashed lines) on the drawing area show the actual page layout which will be produced when printing (see chap. 2.4.17, p. 288) or generating graphics (see chap. 2.2.7.24, p. 156). Additionally the page border line also shows the paper orientation as well as the paper size defined in the printer settings.

The **Page number** numbers all pages continuously by the line starting from the top left.

> **Hint:** It is possible to switch on/off the display of the page border line and of the page number in the "Page layout" window (see chap. 2.2.11.5, p. 192).

It is possible to open more than one model (of the same or different model types) at a time. The status row and the title bar of the model window refer to the model window which is currently active.

The open modelling windows can be ordered and activated using the options available in "Window" menu, its corresponding quick-access icon or ADOxx Explorer (see chap. 4., p. 54).

2.2.4 Modelling modes of the Model Editor

The following two modelling modes are available within the model Editor:

- Editing mode (see chap. 2.2.4.1, p. 131)
- Drawing mode (see chap. 2.2.4.2, p. 131).

> **Hint:** The model Editor's modelling modes differ from the modes available in the menu View (see chap. 2.2.11.1, p. 186), which refer to the modelling objects available for the user.

2.2.4.1 Editing Mode

Editing mode allows you to select objects and connectors already placed on the drawing area, move them and Edit their attributes for example through the ADOxx Notebook. When you are in Editing mode, the mouse pointer is shaped like an arrow when in the drawing area and the Edit button is highlighted in the Class panel (see chap. 2.2.2, p. 128).

2.2.4.2 Drawing Mode

Drawing mode allows you to place new objects and connectors on the drawing area. This mode is activated by clicking on a class or a relation in the Class panel (see chap. 2.2.2, p. 128). When in drawing mode, the mouse pointer is shaped like a pen and the selected object in the Class panel (see chap. 2.2.2.1, p. 129) or Relation panel (see chap. 2.2.2.2, p. 129) is highlighted. An icon from the modelling bar (see chap. 2.2.2, p. 128) can also be selected through a popup menu, which is activated by right-clicking on the modelling panel.

To change from drawing mode to editing mode, click on the drawing area with the **right** mouse button or else click on the Edit icon in the modelling panel.

2.2.5 Drawing Objects

Objects are placed according to the "point and click" principle. This means that you first select an object from the Class panel (see chap. 2.2.2.1, p. 129) by clicking on it. This causes the object to be highlighted and changes the model Editor into Drawing mode (see chap. 2.2.4.2, p. 131).

Now move the mouse pointer to the location within the drawing area at which you would like to place the object. You will notice that the appearance of the mouse pointer is now a pen. By clicking with the left mouse button the object is placed on the drawing area. You can draw any number of objects of the selected class in this way.

If you keep the left mouse button pressed when positioning a new object you can see the object. By releasing the left mouse button the object will be drawn into the current position. (Note: by moving the mouse when this outline is shown, the object can be positioned elsewhere in the drawing area.) If you wish to abort the drawing of this object, press the left and right mouse button simultaneously. In this way the drawing mode is not deactivated. To deactivate drawing mode, press once with the right mouse button on the drawing are (or select the "Edit" icon from the class panel with the left mouse button). When the object is displayed on the drawing area, the corresponding co-ordinates of the location are displayed in the status bar.

If you wish to place a different object within your model, select it by clicking in the class panel and then continue as described above.

Option: Hold down the **<Alt>** key while positioning an object to activate the empty space tool (see chap. 2.2.9.2, p. 178). This enables you to make room in crowded model areas.

> **Hint:** If the object is not placed exactly as you wish on the drawing area, the Snap grid (see chap. 2.2.11.4, p. 190) is activated.

2.2.6 Drawing Connectors

Connectors are drawn in a similar way to objects. Click on the required relation in the Relation panel (see chap. 2.2.2.2, p. 129).

Place the mouse pointer (Drawing mode (see chap. 2.2.4.2, p. 131)) on the object from which the new connector should begin.

> **Hint:** The Focus (see chap. 2.2.7.4, p. 139) will be put on the start object, if it is an allowed start object (according to the definition in the application library) for the selected relation class.

Click on that object and then place the mouse pointer on the object, which should be joined to the first object by the connector.

> **Hint:** The Focus (see chap. 2.2.7.4, p. 139) will be put on the target object, if it is an allowed end object according to the definition in the application library) for the selected relation class.

The connector is now shown in a magenta colour. This indicates that the connector has not yet been created. The connector will only be created when you left click on the target object. As a result the connector will be shown in the defined colour.

You can also draw a connector by moving the mouse pointer Drawing mode (see chap. 2.2.4.2, p. 131) onto the start object, left clicking and then keeping the left mouse button pressed, move the mouse pointer to the target object where you release the button.

It is also possible to create a connector between two objects, which are not visible in the window at the same time. After you have clicked on the start object, move the mouse pointer to the appropriate position. The window will scroll automatically.

You can cancel the drawing of a connector by clicking on the right mouse button before clicking on the target object. In this way drawing mode is not deactivated. To change back to Editing mode click once on the drawing area with the right mouse button or select the "Edit" icon.

By following the method described above, connectors will be drawn in a straight line from the start to the target object.

It is also possible to create **bendpoints** (see chap. 2.2.6.1, p. 133) within a connector.

If you attempt to create a connector between two classes for which this connector is invalid or for which a connector of the same type already exists, an appropriate error message will be displayed. If other connectors are possible between the objects you wish to connect, you can click on the button "other" in the error message and then select a valid connector (see chap. 2.2.6.3, p. 134).

Connectors are usually created using the "point and click" method described above. However it is also possible to create relations using the **Object context menu** (see chap. 2.2.8.2, p. 174) of the object where the connector shall start or end.

For the objects, which ADOxx Notebook contains the chapter "relations" (it depends on the definition of their application library) you can additionally **administrate connectors from the ADOxx Notebook** (see chap. 2.2.7.11, p. 146).

Grouping objects, i.e. objects that are used to hold other objects inside themselves, e.g. aggregations, swimlanes, create automatically the **relation "is inside"** (see chap. 2.2.6.4, p. 135) for the objects they contain.

2.2.6.1 Bendpoints

It is also possible to create **Bendpoints** within a connector. These are points at which the connectors bend (see fig. 99). Bendpoints are inserted by clicking anywhere within the drawing area between the selection of the start and the target object.

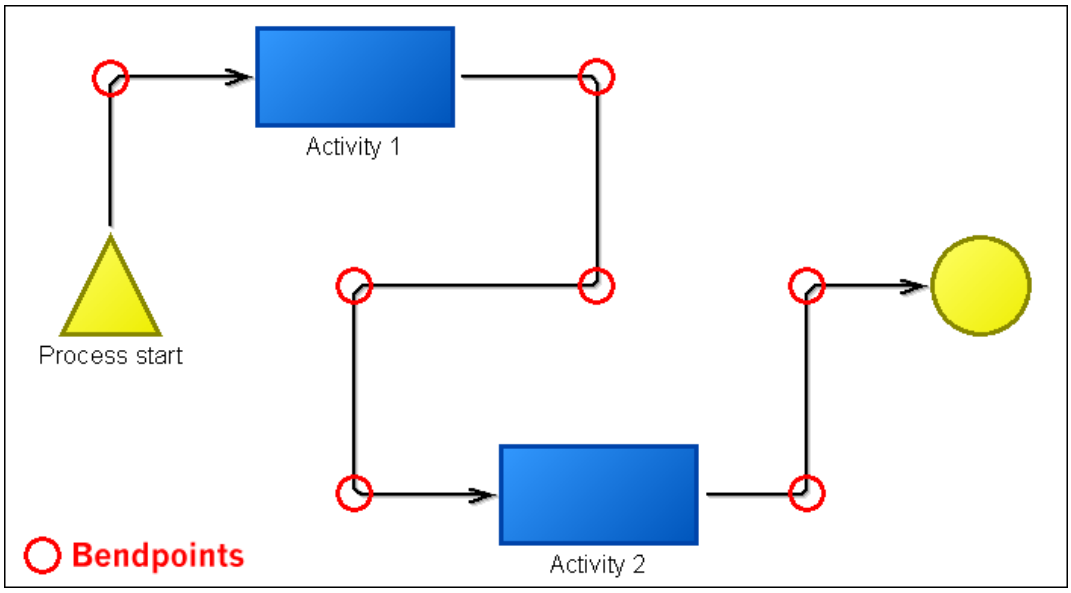


Figure 99: Bendpoints

Hint: The bendpoints of a connector are always positioned on one of the intersections of the snap grid (see chap. 2.2.11.4, p. 190).

2.2.6.2 Adding New Relations via the Context Menu

Adding a relation outgoing or incoming to an object with the context menu is possible if you click on the context menu of an object (right button of the mouse), then on the menu item "new relation" and then select the expected relation in the appearing submenu.

Es erscheint das Fenster "Neue Beziehung anlegen - <*Art der Beziehung*>" (see fig. 100), in welchem Sie, je nachdem, ob die Beziehung hin- oder wegführend ist, entweder das Ausgangsobjekt oder das Zielobjekt selektieren.

Then the "New relation - <*Type of relation*>" window appears (see fig. 100), in which you can select the start object or the target object, according to the type of relation needed.

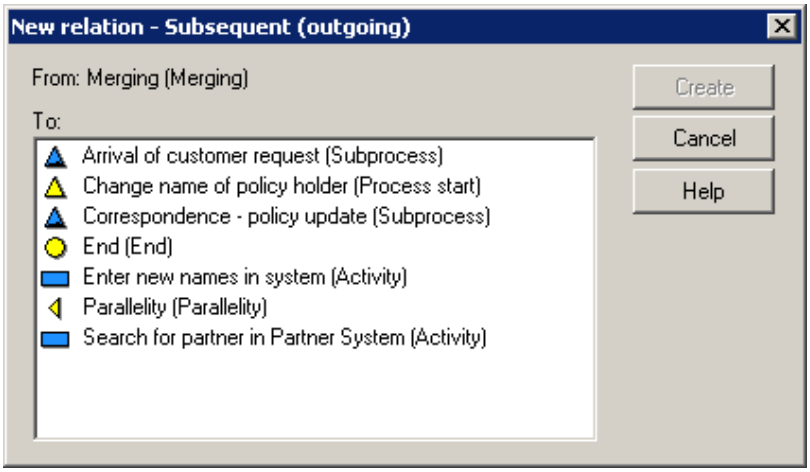


Figure 100: Create new relation

By clicking on the **button "create"** the relation will be made and the connector will be drawn on the model.

2.2.6.3 Adding New Relations

If you attempt to create a connector between two classes for which this connector is invalid or for which a connector of the same type already exists, an appropriate error message ([amodraw-05]) will be displayed.

If other connectors are possible between the objects you wish to connect, you can click on the button "other" in the error message and create a new relation.

After clicking the button "other" the error message will close, both objects you wish to connect will be selected and the "new relation" window (see fig. 101) where the valid relations will be displayed.

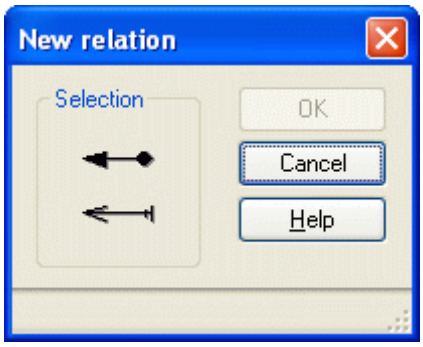


Figure 101: New relation

Select in the **field "select"** the corresponding connector and click afterwards on the OK button.

Hint: The orientation of the connector in the field "select" is related to the originally selected start object, i.e. a connector (directed) **from the left to the right creates an outgoing relation**, a connector (directed) **from the right to the left creates an incoming orientation**.

2.2.6.4 Relation "is inside"

The relation "is inside" will be automatically made between grouping (aggregated) objects and the objects created within these objects.

The relation "is inside" can be evaluated for instance with queries in the Analysis Component (see chap. 3.1, p. 306) , i.e. you can define queries, which results are all objects that are inside a specific domain (e.g. swimlane, aggregation).

The determination of an object, which is located inside a specific domain happens - depending on the definition of the application library- either on the position or on the size of this object. An object is inside a domain, when:

1. The position of this object is inside this domain
2. The object is completely on the area of this domain or
3. The object touches on the area of the domain.

Hint: The position of an object can be displayed and changed (see chap. 2.2.7.7, p. 142) using the context menu of this object (right button of the mouse).

Example from the ADOxx-Default-Library:

The picture below (see fig. 102) shows a part from a Business Process Model with a grouping object of the class "Aggregation" and three objects of the class "Activity".

In addition to the coordinates, the position (defined as object centre) of the objects of the class "Activity" , in the object of the class "Aggregation" the position (which is defined as from the left or from the top) and the size (width "w" and height "h") will be shown.

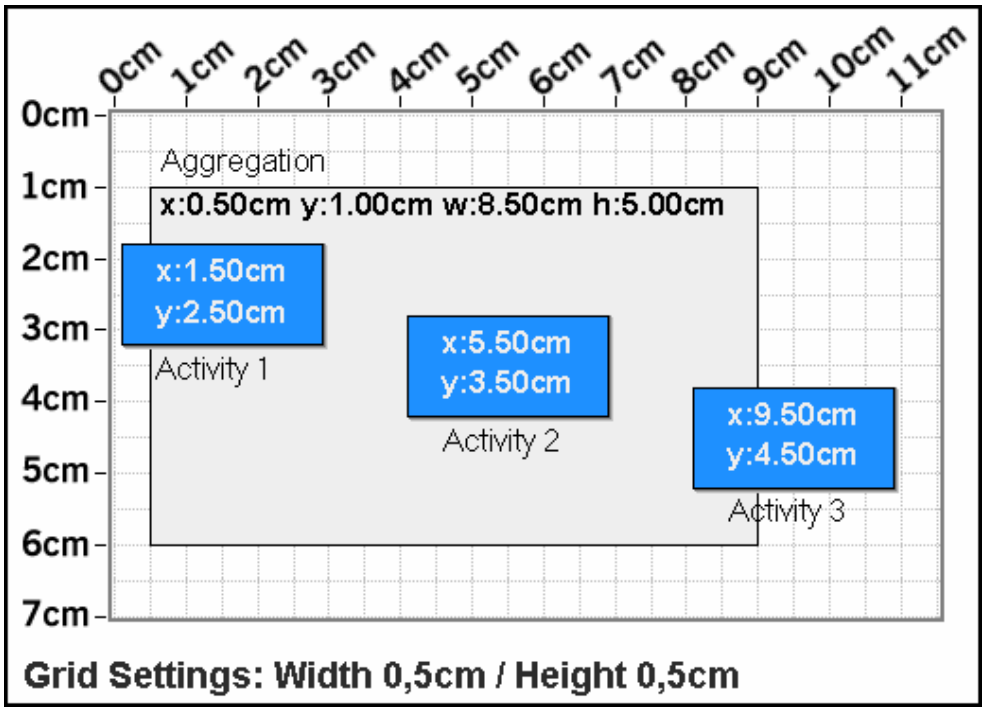


Figure 102: Example of the definition of the relation "is inside"

The following area of the aggregation is the result of these data: `x1:0,5cm/y1:1.0cm`to `x2:9.0cm/y2:6.0cm`.

In the **definition of the relation according to the position** (1.) all objects, which position is on this area - in the example the objects "activity 1" and "activity 2" (see fig. 102) - will be automatically connected to the aggregated object according to the relation "is inside".

In the **definition of the relation according to the complete placing** (2.) all objects which are completely on the area - in the proposed example, the object "activity 2" (see fig. 102) - will be automatically connected to the aggregated object according to the relation "is inside".

In the **definition of the relation according to the partial placing** (3.) all objects, which are touching on this area - in the proposed example, the object "activity 1", "activity 2" and "activity 3" (see fig. 102) - will be automatically connected to the aggregated object according to the relation "is inside".

Hint: The connector of the relation "is inside" will not be shown in the model graphic.

2.2.7 Editing Objects and Connectors

After drawing objects and connectors it is possible to edit them. If the drawing mode is still active, you must first change into the Editing mode (click once with the right mouse button on the drawing area or select the Edit icon). The following functions are available when editing objects and connectors:

- **Undo** (see chap. 2.2.7.1, p. 137)
- **Redo** (see chap. 2.2.7.2, p. 137)
- **Select** (see chap. 2.2.7.3, p. 138)
- **Focus** (see chap. 2.2.7.4, p. 139)
- **Move objects** (see chap. 2.2.7.5, p. 139)
- **Change object size** (see chap. 2.2.7.6, p. 141)

- **Define object size and position** (see chap. 2.2.7.7, p. 142)
- **Change connector course** (see chap. 2.2.7.8, p. 144)
- **Re-allocate connectors** (see chap. 2.2.7.9, p. 145)
- **Rename connector mark** (see chap. 2.2.7.10, p. 146)
- **Change connector representation** (see chap. 2.2.11.6, p. 197)
- **Manage connectors in ADOxx Notebooks** (see chap. 2.2.7.11, p. 146)
- **Edit attributes in the ADOxx Notebooks** (see chap. 2.2.7.12, p. 146)
- **Change visualised attribute values** (see chap. 2.2.7.13, p. 148)
- **Move connector centre** (see chap. 2.2.7.14, p. 148)
- **Copy** (see chap. 2.2.7.15, p. 149)
- **Cut** (see chap. 2.2.7.16, p. 150)
- **Paste** (see chap. 2.2.7.17, p. 151)
- **Delete** (see chap. 2.2.7.18, p. 152)
- **Search model** (see chap. 2.2.7.19, p. 152)
- **Global change** (see chap. 2.2.14, p. 243)
- **Generate object names** (see chap. 2.2.7.20, p. 153)
- **Change values of object/connector attributes** (see chap. 2.2.7.21, p. 154)
- **Number objects** (see chap. 2.2.7.22, p. 155)
- **Convert objects** (see chap. 2.2.7.23, p. 155)
- **Generate graphics** (see chap. 2.2.7.24, p. 156)
- **Align objects** (see chap. 2.2.7.25, p. 160)
- **Arrangement function** (see chap. 2.2.14.2, p. 244)
- **Check Cardinalities** (see chap. 2.4.10, p. 264)

2.2.7.1 Undo

Editing actions can be undone immediately after the particular editing action has been carried out. For this, select menu **"Edit"** - entry **"Undo"**, press the hot key **<Ctrl>+Z** or click the smart-icon .

Hint: The action to be undone is displayed in the option "Undo" within the "Edit" menu. If the function "Undo" is not available, this is shown by the text "Not possible" beside the menu item "Undo" in the menu "Edit".

Hint: Undone actions can be redone using the "Redo" functionality (see chap. 2.2.7.2, p. 137).

2.2.7.2 Redo

Undone actions can be redone. For this, select menu **"Edit"** - entry **"Redo"**, press the hot key **<Ctrl>+Y** or click the smart-icon .

Hint: The action to be redone is displayed in the option "Redo" within the "Edit" menu. If the function "Redo" is not available, this is shown by the text "Not possible" beside the menu item "Redo" in the menu "Edit".

2.2.7.3 Select Objects and Connectors

Any object or connector can be selected simply by clicking on it. Selected objects (see fig. 103) and connectors (see fig. 104) are coloured red.

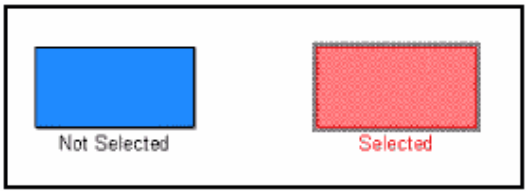


Figure 103: Not selected/selected object

Figure 104: Not selected/selected connectors

You can select **multiple objects and connectors** following these methods:

- Keep the **<Shift>** key or the **<Ctrl>** key **pressed** while clicking one by one on the objects and connectors that should be selected
- Select an area (with the form of a rectangle), which contains the objects and connectors that should be selected, by:
  - Placing the mouse pointer on one corner of the area
  - Pressing the left mouse button and, keeping it pressed
  - Dragging the mouse pointer until the region (limited by a red broken line) is big enough to enclose all the objects and connectors you wish to select.

  All objects, which are completely inside this area will be selected. If the mouse pointer is moved over the edge of the window during the selection, i.e. while the left mouse button is being pressed, then the window content will be moved, however the start point of the area being selected will of course stay the same.

  Note: During the selection of all objects and connectors by area selection, a connector which is inside the area will only be selected if both starting and target objects are selected .
- Select all objects and connectors in a model by:
  - Selecting in the context menu of the drawing area (right mouse button) the menu item "select all"
  - Selecting in the menu Edit the menu item "select all" or
  - Pressing the key combination**<Ctrl>+A**.

Once you have selected a number of objects you can still make changes to this selection by keeping the (<Shift> key) or the <Ctrl> key pressed and clicking on any objects or connectors you wish to select or deselect. Moreover you can also select several areas one by one by keeping the (<Shift>key) or the <Ctrl>key **pressed**

Concerning **objects and connectors lying on top of each other** you can bring the object/connector lying above in the background by selecting in the context menu of the object/connector (right mouse button) the menu item "layer" and then the submenu "to background". This way it is possible to select and edit the objects/connectors lying under.

Hint: Aggregated objects will generally - depending on the definition in the application library - be placed on the lower layer, i.e. if some within objects in these aggregated objects are placed to the background, they will still be (visible) on the aggregated object.

To completely deselect, click on an empty point of the drawing area, or select "deselect" in the menu "Edit".

2.2.7.4 Focus

This shows which single object or which single connector currently has the focus within the model window. In a model window, either no object or exactly one object can have the focus at any time.

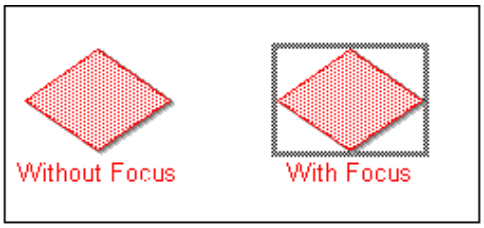


Figure 105: Focused object

A focused object has an outline drawn around it (see fig. 105). A focused connector is shown with both endpoints (○, with its middle point (◆) as well as all Bendpoints (see chap. 2.2.6.1, p. 133)(□) highlighted (see fig. 106).

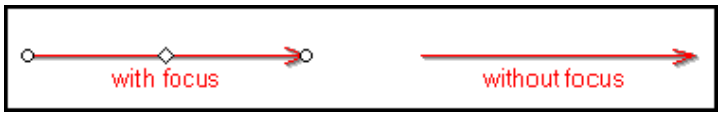


Figure 106: Focused connector

Focus is necessary in order to use the editing functionality "Change object size" (see chap. 2.2.7.6, p. 141), "Change the appearance of connector" (see chap. 2.2.7.14, p. 148) and "Align" (see chap. 2.2.7.25, p. 160). You set the focus by clicking on an object or connector. To remove the focus from all objects, click on some empty space in the drawing area.

2.2.7.5 Move Objects

Moving objects within a modelling window supports "drag and drop" feature. To move a specific object click on it with the left mouse button (the object is now selected and has focus) and then move the mouse (dragging the object) while keeping the left mouse button pressed.

While moving an object you can see that its outline is coloured blue and dashed. When you release the left mouse button the object will be moved to the specified place in the drawing area. If you would like to move more than one object at a time you should select (see chap. 2.2.7.3, p. 138) them first. Then all selected objects can then be moved together as described above. While moving a single object, the other selected objects will also be moved.

You can also move single objects with the context menu (right mouse button) selecting the menu item "position/size" and then defining the new position (see chap. 2.2.7.7, p. 142).

Selected objects can also be moved using the cursor key on the keyboard. With deactivated snap grid (see chap. 2.2.11.4, p. 190) each time you press a cursor key the object will be moved by a step size of one millimetre, with activated snap grid the step size corresponds to the snap height or the snap width.

Hint: To move aggregated objects and all objects they contain, press the <Shift> key before starting the moving and keep it pressed while moving the aggregated object.

Effects on related connectors:

- **Start and target objects of a connector are selected:**

  The connector will be moved unchanged with its object.

- **Starting or target object of a connector is selected:**

  - Case 1: The part of the connector of the object you wish to move and the bendpoint laying beneath this object goes horizontally or vertically.

    The object and the next bendpoint in a horizontal (vertical) direction will by horizontal movement create a new position for both.

    Example from the ADOxx-Default-Library:

    In the model detailed below (see fig. 107) the object "request authorised?" (of the class "Decision") in horizontal position will be moved to the right. The bendpoint in the related connector will also be automatically moved to the right.

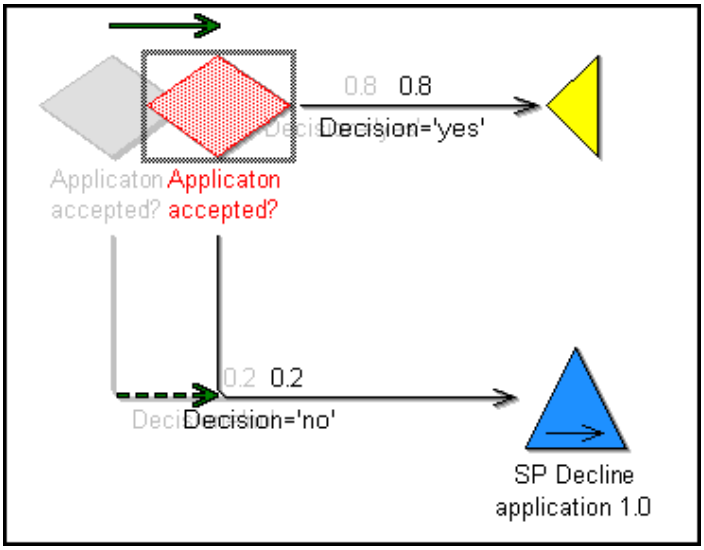


Figure 107: Example of the moving of an object and of its related bendpoints

  - Case 2: The part of the connector between the object you wish to move and the bendpoint laying next to this object does not run horizontally or vertically.

    Only the object will have a new position

    Example from the ADOxx-Default-Library:

    In the model detail below (see fig. 108) the object "request authorised?" (of the class "Decision") in horizontal position will be moved to the right. The position of the direct neighbouring bendpoints in the related connector (from the decision "request authorised?" to the sub process "rejection1.0") stays the same.

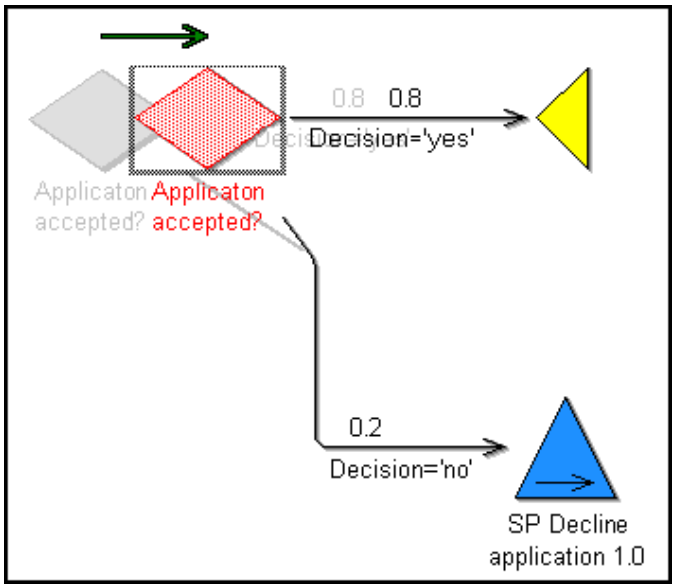


Figure 108: Example of the moving of an object

You can move back the objects to their original position using the function "Undo" (see chap. 2.2.7.1, p. 137).

2.2.7.6 Change Object Size

For certain modelling classes, the size of the objects within the drawing area can be adjusted. In the **ADOxx-Default-Library** this applies to the classes "Aggregation" and "Note".

To adjust the size of an object, click on it to focus it (see chap. 2.2.7.4, p. 139). If the size of the selected object can be adjusted, you will see eight points highlighted □ on the outline which can be used to increase/decrease its size (see fig. 109).

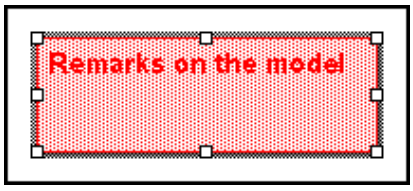


Figure 109: Change object size

Change the size of an object by moving one of its highlighted points. For this, move the mouse pointer on one of the highlighted points. Then the mouse pointer changes into the double arrow which symbolizes the moving direction of the marked points of the object ( ). Press the left mouse button and at the same time move the mouse pointer. As you move the mouse you will see a blue new outline of the object. When you release the left mouse button, the object will take its new size and will correspondingly be represented on the drawing area.

Additionally you can change the size of an object using its context menu (right mouse button) by selecting the menu item "position/size" define (see chap. 2.2.7.7, p. 142).

This change to an object size can be undone using the function "Undo" (see chap. 2.2.7.1, p. 137).

Hint: If the snap grid is activated, the change in size will be carried out in steps corresponding to the grid. Therefore it is easy to scale and align different objects to

identical dimensions and alignment. If you do not want this, deactivate the snap grid or hold the <Crtl> key while drawing.

2.2.7.7 Define Object Size and Position

The size and position of objects can be defined by inputting the exact values (in cm) .

Hint: The change of size is only possible for objects of a specially defined class (=objects changeable in size), like for instance the objects of the class "aggregation" and "notice" from the **ADOxx-Default-Library**.

The position of an object is defined with the values "**x (horizontally from the left)**" and "**y (vertically from the top)**",and its size with the values "**width**" and "**height**" (see fig. 110).

The reference point for the position depends on the definition of the graphical representation of each class. The reference point of objects changeable in size is generally on the upper left corner of the object and for objects with fixed size it is generally at the centre point of the object.

Example from the ADOxx-Default-Library:

The picture below (see fig. 110) represents an object changeable in size of the class "notice" (left) and an object of the class "Activity" (right) from the ADOxx-Default-Library (see chap. 16., p. 802), the reference point for the notice is in the left upper corner of the object and for the Activity it is the middle point of the object.

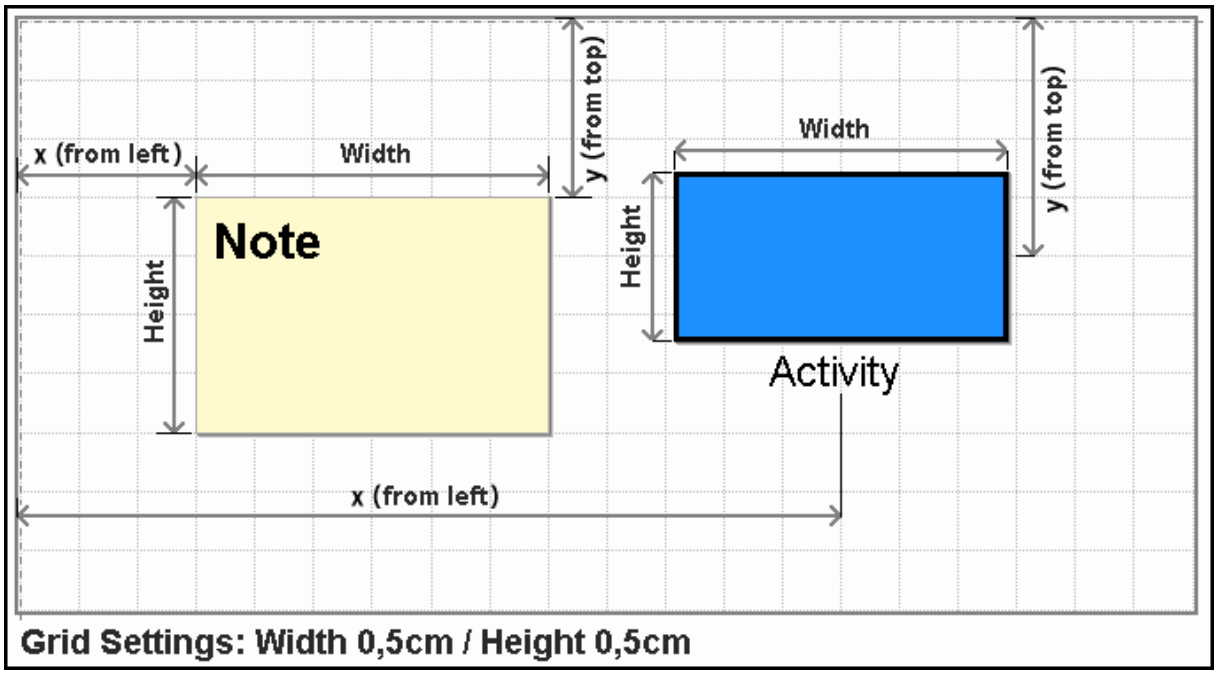


Figure 110: Reference point for the object position and size

The definition of the position and/or the size of an object is possible by selecting the menu item "position/size" in the context menu (right mouse button) of the object.

For **objects changeable in size**, the window "position/size" (see fig. 111) with changed settings for the values "**x (horizontally from the left)**", "**y (vertically from the top)**", "**width**" and "**height**" will be shown.

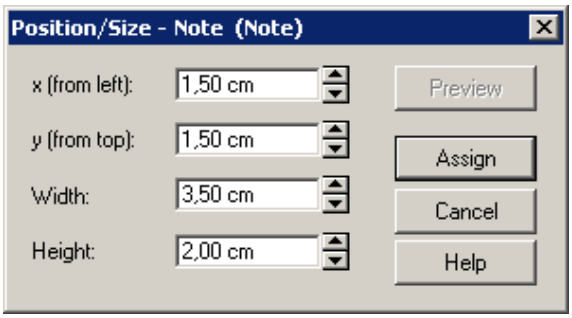


Figure 111: Define size and position (for objects changeable in size)

For **objects with fixed size**, the window "position/size" (see fig. 112) with changed settings for the values "**x (horizontally from the left)**", "**y (from top)**" as well as the fixed values "**width**" and "**height**" will be shown.

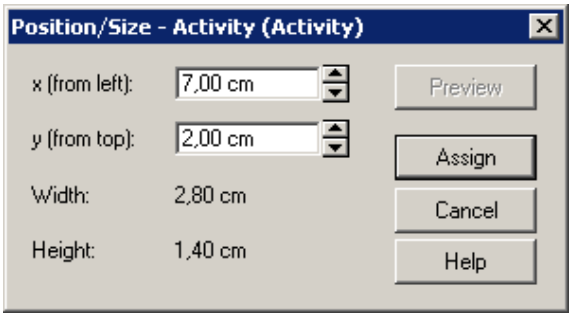


Figure 112: Define position (for objects with fixed size)

For **swimlanes**, the window "position/size" (see fig. 112) with changed settings for the value "**height**" as well as the fixed value "**y (vertically from the top)**" (for horizontal swimlane) or the changed value "**width**" as well as the fixed value" **x (horizontally from the left)**"(for vertical swimlanes) will be shown.

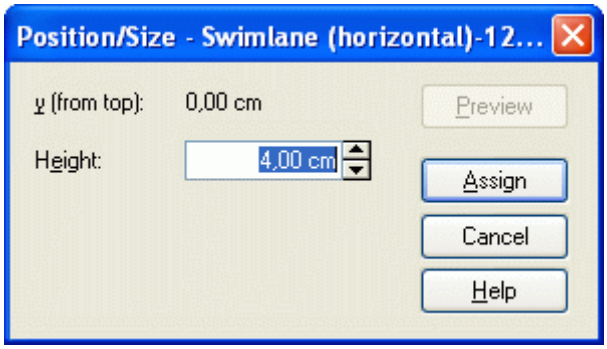


Figure 113: Define height (for swimlanes)

Change the values by direct input in the field or by clicking the buttons by steps of 0,1 cm.

By clicking on the button "preview" you will have a look at the effects of the values changed, clicking on the button "assign" " confirms the new position of the object.

2.2.7.8 Change Connector Course

The appearance of a connector can be changed by :

- adding new bendpoints (see chap. 2.2.6.1, p. 133), or
- moving existing bendpoints, or
- removing existing bendpoints or
- moving parallel connector edges.

Add a new bendpoint:

To add a new bendpoint:

- place the mouse pointer on a part of the connector without other elements,
- hold down the left mouse button (the connector is selected and focussed) and
- drag the connector to the place you want the bendpoint to appear.

Once you release the left mouse button the connector will be placed in its new position (□).

Move a bendpoint:

Moving bendpoints works in the same way as moving objects:

- place the mouse pointer on the bendpoint you want to move,
- hold down the left mouse button (the bendpoint is selected and focussed) and
- drag the bendpoint to the place you want the bendpoint to appear and release the mouse button.

Remove a bendpoint:

To remove a bendpoint, just move it into such a position that a straight line between the two neighbouring bendpoints is created.

Remove all bendpoints of a connector:

To remove all bendpoints of a connector:

- open the context menu of the connector,
- select the menu entry "Remove all bendpoints".

Moving connector edges parallel:

Horizontal and vertical connector edges can also be moved parallely. For that purpose a "handle"(grey bar) is provided along the edge. To move the connector in a parallel motion, take the following steps:

- Select a connector (by clicking on it),
- Point the mouse pointer on the "handle" (the mouse pointer will change into or) and

- Holding the mouse button, move the connector edge into the desired position.

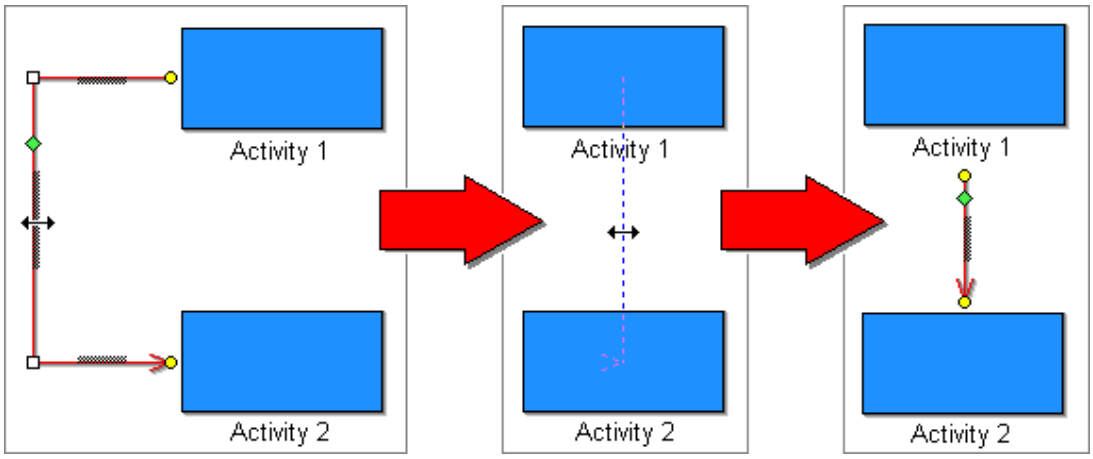


Figure 114: Connector edges moved in parallel

After releasing the mouse button, the connector edge will be set in the new position.

To undo a connector modification, click "Undo" (see chap. 2.2.7.1, p. 137) ( ) .

2.2.7.9 Re-allocate Connectors

Re-linking connectors means that either the start or target object of an existing connector is changed. To re-link a connector first focus (see chap. 2.2.7.4, p. 139), click on the start or endpoint ( ) of this connector (with the left mouse button) and move it to the new starting or target object you wish, while keeping the left mouse button pressed (see fig. 115).

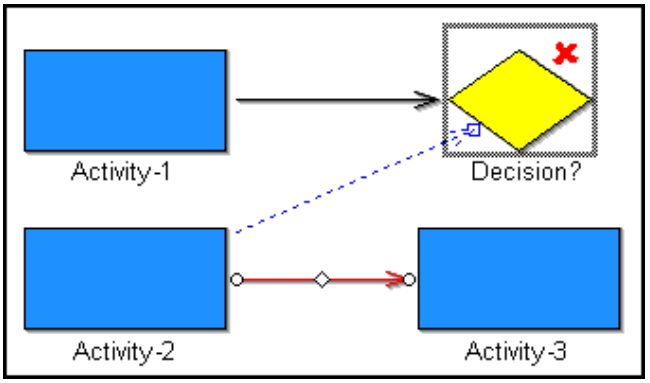


Figure 115: Example of the re-linking of a connector

Hint: Only authorised start or target objects will be focused (see chap. 2.2.7.4, p. 139), (accordingly to the definition in the application library) for the selected relation class.

Re-linking connectors is only possible where the relation is permitted between the start and target objects or where a connector of the same type does not already exist between the two objects. If you attempt to re-link a connector in the invalid manner an appropriate error message will be displayed and the connector will remain the same.

The re-linking of connectors can be undone using the function "Undo" (see chap. 2.2.7.1, p. 137).

2.2.7.10 Rename Connector Marks

The value of a connector mark (see chap. 2.2.11.6, p. 197) displayed in a model can be edited by double clicking on the connector mark. Within the window "Rename connector mark" (see fig. 116) the current value is shown in the field "New name".

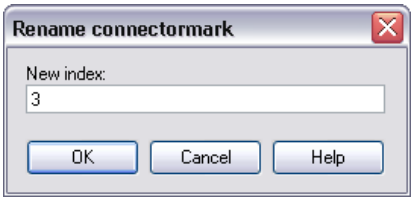


Figure 116: Rename connector mark

Enter a new value and click on the OK button to assign the new value.

Hint: The new value has to be either numeric or alphabetical, depending on the definition and has to be unique within the connector marks of one model.

Hint: When performing an automatic generation of connector marks (see chap. 2.2.11.6, p. 197) previously assigned values may be overwritten.

2.2.7.11 Manage Connectors in ADOxx Notebooks

Accordingly to the definition in your application library, connectors can be administrated in the chapter "Relations" of object's notebook.

Relations leading to or leading from an object are different. For each possible relation leading to or leading from an object there is an entry with a list of all corresponding existing connectors.

The lists will be automatically updated when any changes to the model concerning the connectors are made.

On the right above each list there are three buttons for the editing of the connectors:

- "Add relation" (see chap. 3.2.7, p. 50)
- "Delete relation" (see chap. 3.2.8, p. 50)
- "Display notebook" (see chap. 3.2.11, p. 50)

Hint: To delete connectors via ADOxx Notebook, select either in the menu "Edit" the menu item "Undo", click the key combination **<Ctrl>+Z** or click on the smart icon .

2.2.7.12 Edit Attributes in the ADOxx Notebooks

Double-clicking on an object or connector causes its ADOxx Notebook (see fig. 117) to be opened. Within an ADOxx Notebook, you can edit the attributes of an object or of a connector.

Hint: The ADOxx Notebook will only be opened, if it is defined in the application library of the relevant object or connector. Otherwise an appropriate indication will be shown.

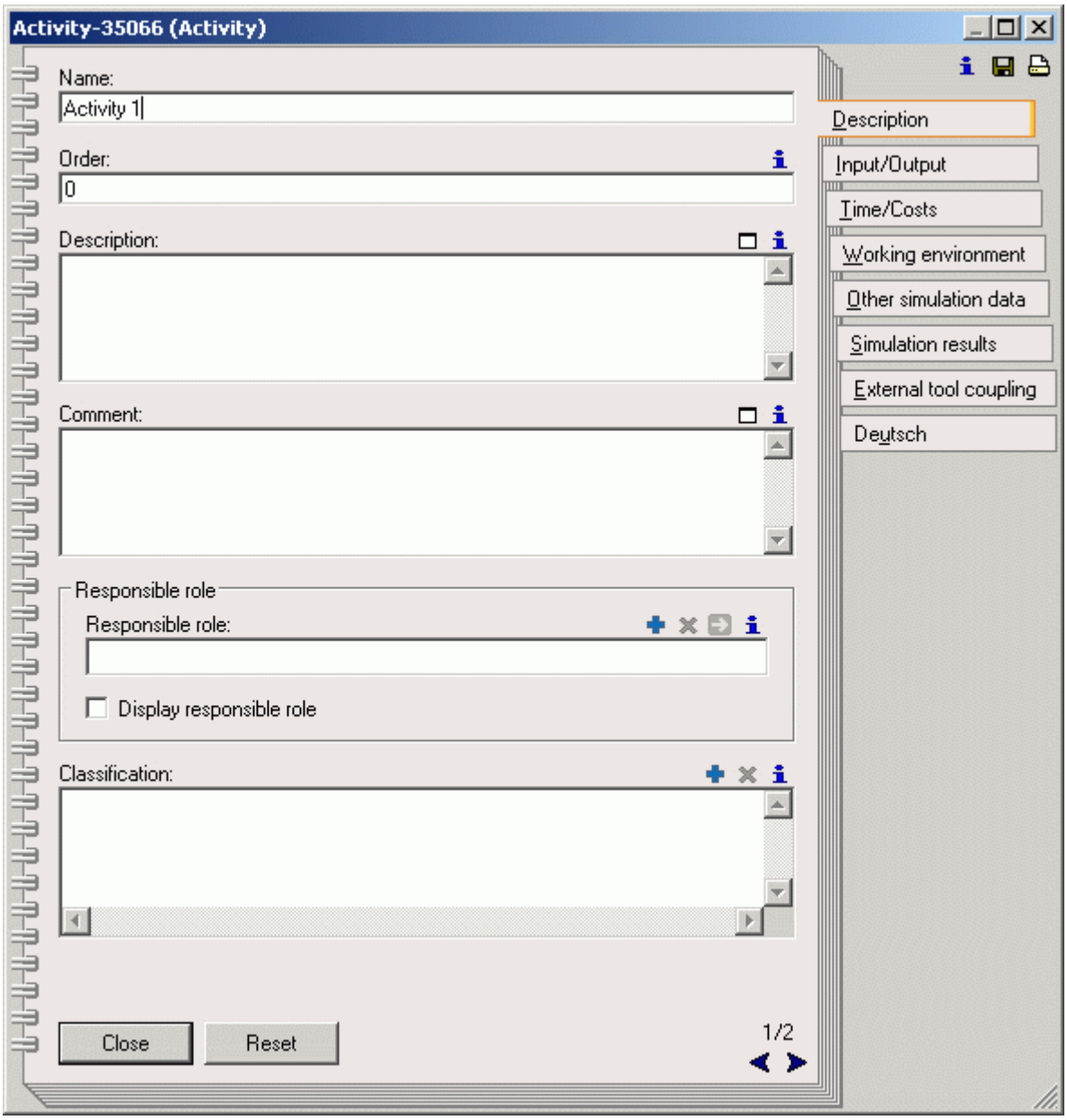


Figure 117: ADOxx-Notebook of the object of the class "Activity" (from the ADOxx-Default-Library)

The ADOxx Notebooks are structured like office files with a register tabs. The register tabs (chapters) are on the right-hand side of the ADOxx Notebook. By clicking on the tabs, the first page of the selected chapter is displayed. If a chapter contains more than one page, then you will see in the right lower corner of the displayed page the number of the page and the total amount of pages in this chapter (e.g. 1/3).

Click the buttons and in the right lower corner of the ADOxx Notebooks or stroke the keys <Alt>+<-> or <Alt>+<+>to leaf page by page through the ADOxx Notebooks.

Hint: The paging within a chapter of an ADOxx Notebook depends on the screen resolution and on the font size and can be different from the pictures in this chapter.

The contents of an ADOxx Notebook can be saved in a datum (Button) or printed out (Button).

To edit the attributes of an object, enter the expected values in the fields of the attribute. The values you have entered will be immediately taken over when you leave the corresponding editing field. Close

the ADOxx Notebook after the entry by clicking on the close button. To fetch back the original value of the last edited input field of a page, click on the button "put back". If you want to undo changes in several notebook pages, use the function "Undo" (see chap. 2.2.7.1, p. 137).

The ADOxx Notebooks contain controls (see chap. 3.2, p. 48), which can support you in adding attributes values.

Hint: The ADOxx Notebook of an object can also be opened using the context menu of this object (right mouse button) by selecting the menu item "notebook".

Hint: ADOxx Notebooks are defined for each class and for the relation 'Subsequent" in the **ADOxx-Default-Library** (see chap. 16., p. 802).

2.2.7.13 Change Visualised Attribute Values

By displaying of attribute values (visualised) in the model graphic, you can represent the information you want to (e.g. object name) in the drawing area.

Hint: The visualised attribute values are defined in the application library.

When you wish to change an attribute value, first select the object and then click on the attribute value. An input window will be opened, (see fig. 118), in which the current attribute value will be displayed.

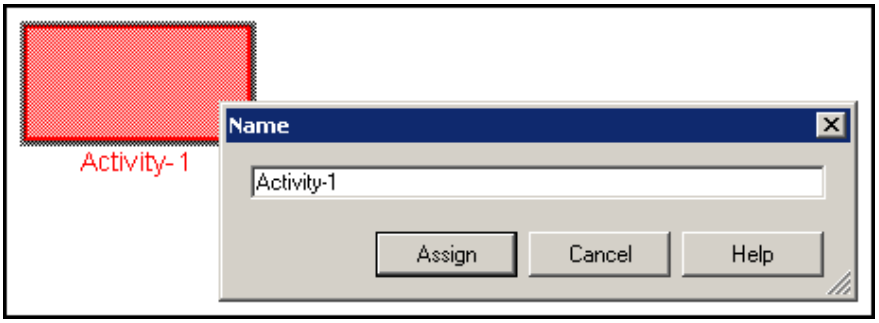


Figure 118: Change visualised attribute value

Hint: Depending on the type of visualised attribute, another input window - e.g. input support field (see chap. 2.2.12, p. 201) - may be displayed.

Change the attribute value and confirm by clicking on the button "assign". The change will be taken over and the attribute value will be visualised on the drawing area.

You can also call the input window using the context menu of the object (right mouse button). If only an attribute value is visualised, then its name is entered as menu item in the context menu. If they are several, the menu item "visualised attribute values" will be shown with a sub menu item for each attribute.

2.2.7.14 Move Connector Centre

As well as the start point (), endpoint () and bendpoints () a focused connector also shows the edge middle point () (see fig. 119).

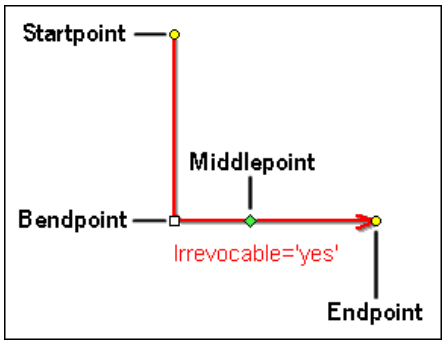


Figure 119: Edge middle point

In the **ADOxx-Default-Library** visualised connectors attributes of the relation "subsequent" will be aligned. As you can move the edge middle point, it is also possible to adjust the label position of the connector.

> **Hint:** If you are using a customised application library, the connector labels can also be displayed at the start or the endpoint of the connector. It is also possible to assign a graphical symbol to the edge middle point (such as the arrows at the start or end point).

If the edge middle point of a connector has not yet been moved, then ADOxx will place the midpoint in the geometric centre of the connector. If on the other hand it has already been moved, then ADOxx will try to maintain a position. The chosen position follows any object or bendpoint movements.

To move an edge middle point first select the corresponding connector in order to give it focus (see chap. 2.2.7.4, p. 139). Then click on the edge middle point mark ◆ with the left mouse pointer and pull it along the connector to the position it should have. By releasing the left mouse button, the connector will be assigned the new position of the edge middle point and the connector will be drawn with the related moved elements (connector label).

If the edge middle point of a connector has been moved and you would like to replace it in the geometric centre, then right click on the connector and select the menu item "edge middle point" from the context menu and then the submenu "auto". The edge middle point will be calculated according to the changed position and the connector will be drawn with the related moved elements.

The last movement of the "Edge middle point" can be undone by using the "Undo" (see chap. 2.2.7.1, p. 137) function.

2.2.7.15 Copy

You can Copy objects and connectors by selecting (see chap. 2.2.7.3, p. 138) them and then using the "Copy" function. This function can be called in a number of ways:

- Select the menu item "Copy" from the "Edit" menu.
- Use the key combination <Ctrl>+<c>.
- Select the menu item "Copy" from the context menu (see chap. 2.2.8, p. 173) of a selected object". If you only wish to Copy one object, it is not necessary to select it beforehand.
- Click on the smart-icon in the quick-access bar.

Object and connectors copied remain in the clipboard until other objects or connectors are copied or cut or until you exit ADOxx. Objects and connectors can be copied into any model of the same type, i.e. which has the same classes and relations.

> **Hint:** A connector will only be copied if it is selected with the two objects it connects.

Hint: When you copy and then paste referenced objects (see p. 151) you can choose to take over the references into the pasted object or leave them in the original object.

ATTENTION: The options "Copy", "Cut" (see chap. 2.2.7.16, p. 150) and "Paste" (see chap. 2.2.7.17, p. 151) can only be applied within ADOxx. If you wish to Copy the graphic model representation into any other applications, you must use the function "Generate graphics" (see chap. 2.2.7.24, p. 156)".

2.2.7.16 Cut

You can cut objects and connectors by selecting (see chap. 2.2.7.3, p. 138) them and then calling the "Cut" function. This function can be called in a number of ways:

- Select the menu item "Cut" from the "Edit" menu.
- Press the key combination <Ctrl>+<x>.
- Select the menu item "Cut" from the context menu (see chap. 2.2.8, p. 173) of a selected object". If you are cutting only one object, marking is not necessary.
- Click on the "Cut" smart-icon in the quick-access bar.

Objects and connectors that were cut remain active in the clipboard until other objects or connectors are copied or cut until you exit ADOxx. Objects and connectors cut can be copied into any model of the same type, i.e. which has the same classes and relations. The function "Cut" is a combination of the functions "Copy" (see chap. 2.2.7.15, p. 149) and, afterwards, "Delete" (see chap. 2.2.7.18, p. 152)".

Once the "Cut" function has been used, it is possible to undo it using the "Undo" (see chap. 2.2.7.1, p. 137) function.

Hint: A connector can only be cut when it and the two objects it connects are selected.

Hint: When you cut referenced objects (see p. 150) the references to this object will be invalid.

ATTENTION: The options "Cut", "Copy" (see chap. 2.2.7.16, p. 150) and "Paste" (see chap. 2.2.7.17, p. 151) can only be applied within ADOxx. If you wish to copy the graphic model representation into any other applications, you must use the function "Generate graphics" (see chap. 2.2.7.24, p. 156)".

Cut Referenced Object

When you cut an object, which is referenced by an other object, an appropriate indication appears before deleting this object (see fig. 120), since the references of the referenced object become invalid after it has been cut.

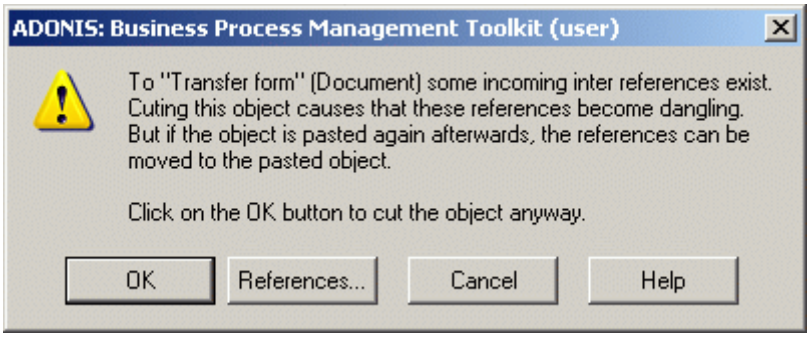


Figure 120: Indication before cutting a referenced object

Hint: When you paste a referenced object (see p. 151), you can take over the references to this object, i.e. the references remain valid.

Click on the button "references", to get the list of all objects, which reference the object you wish to delete (see p. 268).

Clicking on the button OK will cut.

2.2.7.17 Paste

You can insert copied (see chap. 2.2.7.15, p. 149) or cut (see chap. 2.2.7.16, p. 150) objects and connectors into models of the same type, i.e. with the same defined classes and relations, by calling the "Paste" function. This function can be called in a number of different ways:

- Select the menu item "Paste" from the "Edit" menu.
- Press the key combination <Ctrl>+<v>.
- Select the menu item "Paste" from the context menu (see chap. 2.2.8, p. 173)(Key <F12> followed by <p>).
- Click on the appropriate smart icon in the quick-access bar.

Once the "Paste" function has been called, place the mouse pointer on the position within the drawing area at which you would like the clipboard contents to be inserted. If you move the mouse pointer over the edge of the window or the drawing area, scrolling starts.

Click on the spot where you would like the objects to be pasted. The pasted objects and connectors are selected. Instead of confirming the position, you can cancel the paste process by right-clicking.

Hint: Copied or cut objects and connectors can only be pasted into models of the same type, i.e. in which the classes and relations for the object or connector to be pasted are defined!

Hint: When you paste referenced objects (see p. 151) you can choose to take over references to the pasted object or leave them to the original object.

Paste Referenced Objects

If you want to paste an object, which is referenced by another object, (see fig. 121).

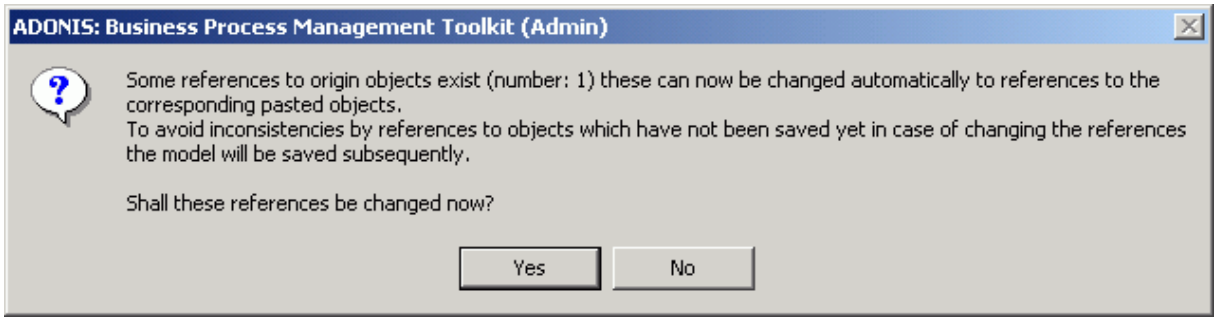


Figure 121: Query before pasting a referenced object

Click on the **yes-button**, to apply changes, i.e. the **references** on the copied or cut original object will be deleted and **set in the pasted object**.

Click on the **no button**, to cancel changes, i.e. the **references on the copied or cut original object will remain the same** and the pasted object will not be referenced by another object.

2.2.7.18 Delete

First selecting (see chap. 2.2.7.3, p. 138) the required object(s) and then calling the "Delete" function can delete objects and connectors. This function can be called in a number of ways:

- Select the menu item "Delete" from the "Edit" menu.
- Press the button <Del>.
- Select the menu item "Delete" from the context menu (see chap. 2.2.8, p. 173) of a previously selected object.

Objects and connectors, which were deleted can be restored by using the "Undo" (see chap. 2.2.7.1, p. 137) function.

Hint: When you delete referenced objects (see p. 152) the references to this object will become invalid .

Delete Referenced Object

When you delete an object, which is referenced by another object, an appropriate indication appears before deleting this object (see fig. 122), since this reference is not valid after you have deleted this referenced object.

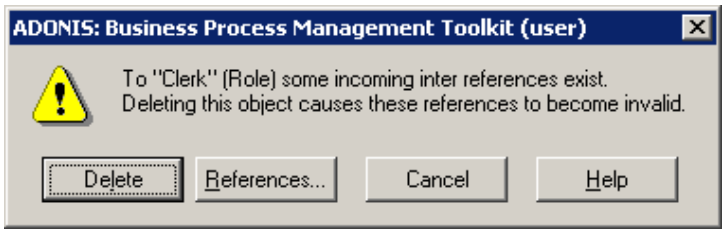


Figure 122: indication before deleting a referenced object

Click on the button "reference", to get a list of all objects, which are referenced to the object you wish to delete (see p. 268).

By clicking on the button "delete" you will delete the object and all the references to this object will be invalid.

2.2.7.19 Search Model

Using the function "find", it is possible to search within a model for all labels -i.e. in visualised attribute values (see chap. 2.2.7.13, p. 148).

The function "Find" can be called in a number of ways:

- Select the menu item **"Find"** from within the **"Edit"** menu.
- Press the hot key **<Ctrl>+F**.
- Click on the appropriate smart icon in the quick-access bar.

Once the function "Find" has been called, the window "Find in models" (see fig. 123) is displayed. In this window you can enter the text or the subtext, which should be searched for.

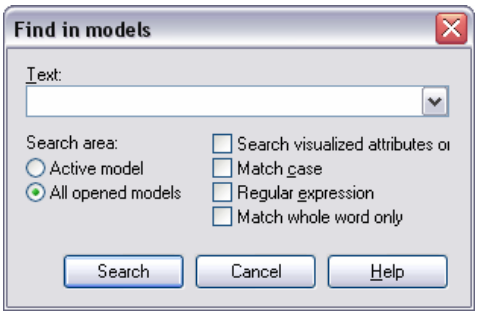


Figure 123: Search in models

Select, whether the search shall be performed only on the **active model** or in **all opened models**.

Activate **"Search visualised attributes only"**, if you only want to look for visualised attribtue values. A deactivated option extends the search base to all attributes.

If you wish to distinguish between capital and small letters and want the search to only return exact matches, then activate the option **"Match case"**.

If the option **"Regular expression"** is activated, then the search text will be interpreted as a regular expression (see chap. 6., p. 731).

Activate the search criterion **"Match whole word only"**, to seach for words. A word is a piece of text flanked by blanks or punctation marks.

Wildcards can be used when the option "Regular expression" is deactivated. "?" is for one character exactly, "*" for any number of characters (For regular expressions this is expressed by "." or ".*").

Click **"Search"** to start the search. The search results are displayed in the search result window (see chap. 2.9, p. 39).

If no matches are found, an information window pops up explaining that the particular expression being searched for was not found.

2.2.7.20 Generate Object Names

When generating object names, starting from a start object, all objects connected with relations will be provided in an appropriate attribute with a number.

> **Hint:** The function for object name generation is defined in the application library. There it is determined which object classes are affected and which attribute (=start attribute) provides the object name.
>
> If necessary, contact your ADOxx consultant.

To generate object names, open or activate the related model and select in the "Edit' menu the menu item "Update attributes" and them the submenu "Generate object names".

The names of the objects of the current model in the model editor will be generated and the model representation updated.

Conditions/Results:

- the name will be generated only for the objects, which names have not been already changed manually.

  The name entered by the ADOxx user will not be overwritten.

  Note: If the name entered by the ADOxx user is equal to the name set by ADOxx automatically ("<*class name*>-<*digit*>") nevertheless an object name will be generated.

- The generated object names will be automatically numbered all the way through in ascending order, in order to make the model as clear as possible.
- If no value has been assigned to the attribute, from which the object name to be generated has to be taken (start attribute), the class name (with an ordinal number) will be entered as the object name.
- If you want to realize the function "generate object name" in a start attribute in which you have formerly changed the value, then this change will be correspondingly taken over and a new name will be generated.

  Hint: If you call the function "generate object name" in a start attribute where no value is assigned, then a new generation of the name will be realised (possibly with an other ordinal number).

2.2.7.21 Change Values of Object/Connector Attribute

Additionally to the possibility to edit the objects and connectors attribute in the ADOxx Notebooks (see chap. 2.2.7.12, p. 146) you can change at the same time the attribute values of several objects and connectors.

Hint: It is only possible to change the attribute values of several objects or connectors for objects of the same class or for connectors of the same relation.

To change the attribute values of several objects (connectors), highlight the objects (connectors) to be changed and then select the menu item "Edit attributes" in the menu "Edit". All attributes of the previously highlighted objects or connectors will be listed in the window "change attributes - attribute selection" (see fig. 124).

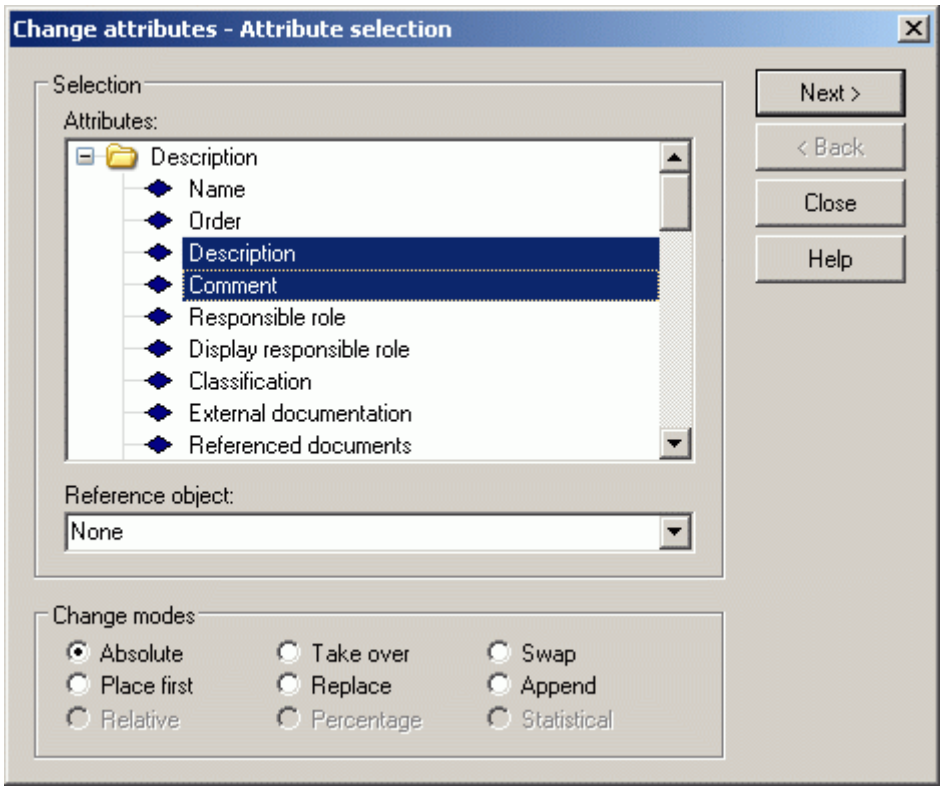


Figure 124: Change the values of object/connector attribute - attribute selection

All highlighted objects are contained in the field **"Reference object"**. A reference object is useful when you select the change mode "Absolute" to pre-assign the value of the attribute you wish to change.

Select in the list **"Attributes"** all the attributes, which should be changed within the listed objects.

To realize attribute changes, select one of the change modes listed in the lower part of the window. The possibilities of selection of the change mode depend on the type of the selected attribute, i.e. for certain attribute types (e.g. figure, time format) only special, meaningful change modes are offered.

As in the Global change (see chap. 2.2.14, p. 243) function, the following **change modes** are at your disposal:

- **"Absolute"** (see p. 251)
- **"Take over"** (see p. 251)
- **"Swap"** (see p. 251)
- **"Place first"** (see p. 252)
- **"Replace"** (see p. 252)
- **"Append"** (see p. 253)
- **"Relative"** (see p. 253)
- **"Percentage"** (see p. 254)
- **"Statistical"** (see p. 254)

Hint: If none of the change modes are available, it means that you have selected attributes, which, because of their type, cannot be changed in one process.

Continue the attribute changes by clicking **"Change"**. Depending on the change mode, an appropriate information window is displayed.

After having done the changes, an information window containing all attributes which values have been changed, will be displayed

2.2.7.22 Number Objects

When numbering objects, all objects connected by relations are supplied with a number in a certain attribute, starting at a start object without a predecessor.

Hint: The numbering function and its availability is defined in the application library. It is possible to define which classes via which kind of relation should be numbered, the attribute in which the number calculated is entered, and if the numbering is to be carried out consecutively over all classes or within the single classes.

If necessary, please contact your ADOxx administrator.

To number the objects, open or activate the appropriate model and select the menu item "Update attributes" from the "Edit" menu. Within this menu item, you can now select "Objects numbering".

The objects of the current model in the model editor are numbered and the model representation is updated.

2.2.7.23 Convert Object

When you convert objects, an object of a special class will be converted into an object of another class (= target class).

Hint: The function "convert" is defined in the application library. It is possible to define which objects should be converted and which attribute of these objects should be taken over by this conversion.

Hint: In the **ADOxx-Default-Library** you can convert in the Business Process Models the objects of the class "Activity" into objects of the class "Subprocess" (and vice versa).

To convert an object, highlight an object and select in the context menu of this object (right mouse button) the menu item "Convert" and then the submenu with the name of the target class.

The object will be converted in the target class and correspondingly displayed on the drawing area.

ATTENTION: The conversion of objects **cannot** be undone.

2.2.7.24 Generate Graphics

You can save models or model parts as graphic files or copy them into the clipboard by using the function "Generate graphics". You can then import these graphics into other applications (word processing, graphic programs etc.) ADOxx supports the following formats when saving graphics:

Generation possibilities:

- A **model section** (see p. 156),
- Only **selected pages** (see p. 158),
- The **whole model** (via both other functions).

Supported file formats:

- **"BMP-1" - Microsoft Windows Bitmap (monochrome)**; bitmap graphic with one bit colour-depth (black-and-white).
- **"BMP-24" - Microsoft Windows Bitmap (24-bit)**; bitmap graphic with 24-bit colour depth (True Colour, 16,7 mill. colours).
- **"PCX-8" - ZSoft PC-Paintbrush (8-bit, compressed)**; PCX graphic with 8-bit colour depth (256 colours).
- **"PCX-24" - ZSoft PC-Paintbrush (24-bit, compressed)**; PCX graphic with 24-bit colour depth (True Colour, 16,7 mill. colours).
- **"JPG" - JPEG-Graphics Formate**; JPEG graphic (True Colour, 16,7 mill. colours).
- **"PNG" - Portable Network Graphics**; PNG graphic with 24-bit colour depth (True Colour, 16,7 mill. colours - Windows only).
- **"EMF" - Windows Enhanced MetaFile**; EMF graphic with 24-bit colour depth (True Colour, 16,7 mill. colours - Windows only).
- **"SVG" - Scalable Vector Graphics**; SVG graphic with 24-bit colour depth (True Colour, 16,7 Mill. colours, no click regions - Windows only).

Model Area

ATTENTION: If you want to generate a graphic from a model area, you must mark the part of the model from which the graphic should be generated **before** you

call the function "Generate graphics". Otherwise, a full model graphic is exported.

Prepare generation:

Determine a section by dragging a frame around the desired area by holding down the **<Alt>** key and moving the mouse with pressed **left mouse button**. A blue-green rectangle frame appears, indicating the area:

After releasing the mouse key you can adjust the frame size and position by clicking one of the borders and dragging it into another position.

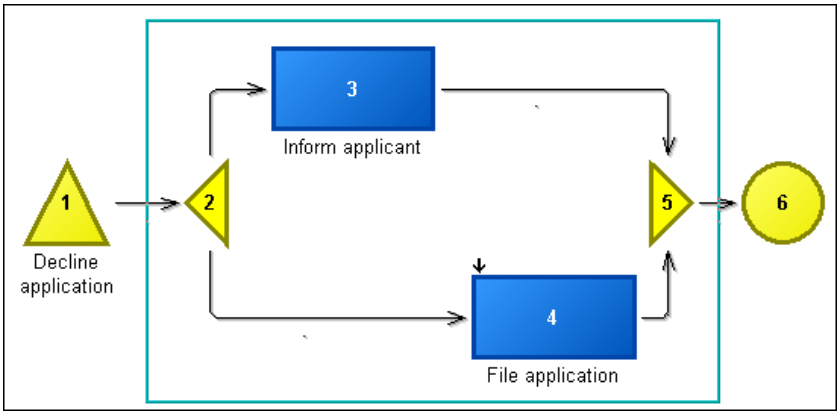


Figure 125: selecting a model section

Select the option **"Generate graphics"** from the **"Edit"** menu click the smart-icon in the quick-access bar. The window "Generate graphics (Region)" appears (see fig. 126).

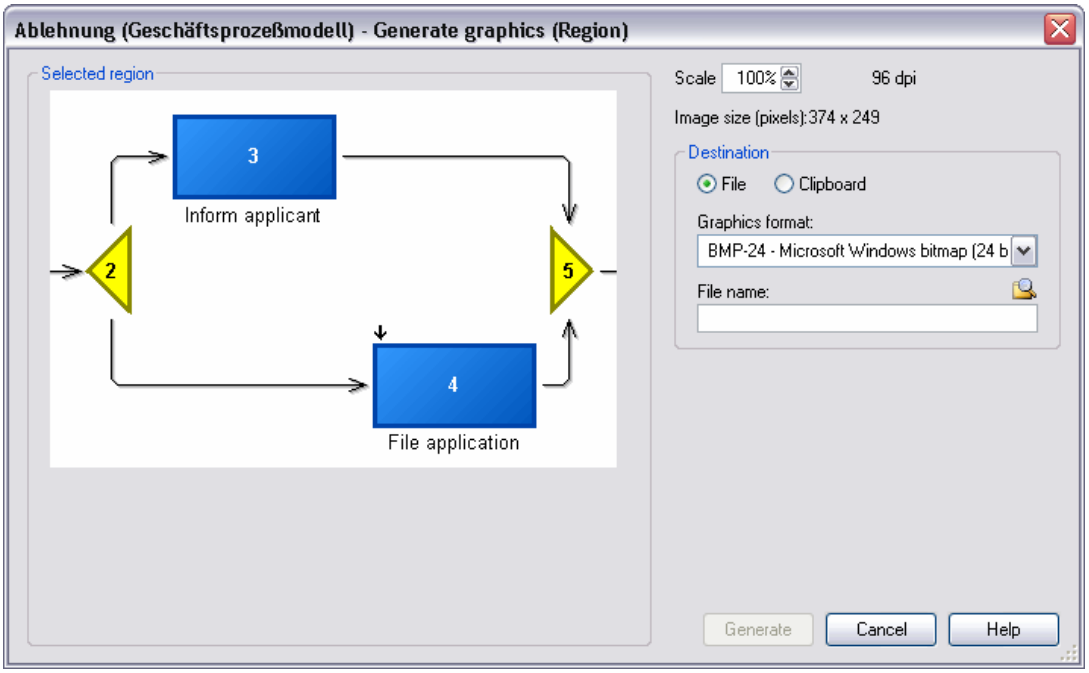


Figure 126: Generate graphics - region

Generate:

In the field **"Selected area"** you can see a preview of the graphic in the current scaling, which will be automatically adjusted with every change.

With the **"Scale"** percentage, you determine the final size of the graphic (default: 100%). The graphic will be saved with the ideal print resolution 96 dpi (dots per inch). The size of the picture in pixels will be shown as well.

In the function group **"Destination"** you decide, whether the graphic should be exported to a **file** or the **clipboard** of the operating system. In case of copying into the clipboard, no additional settings are necessary. After copying it into the clipboard, you can insert the graphic into other applications immediately. If you wish to save the graphics to a file, you have to specify the graphic format, name and target path as well.

> **Hint:** The contents of the clipboard are lost as soon as other graphics, texts etc. are copied into the clipboard or when the computer is switched off.

Select one of the provided file formats from the dropdown list **"File format"**.

Enter the target path and file name into the field**"File name"**. Click to call a support dialogue.

Click **"Generate"** to start the generation process. A status window informs you about the progress.

Model Pages

With this function, you can separately select pages of the layout to be exported as a graphics or not a graphics. Graphic of the entire model, will be generated if all the pages were selected.

> **ATTENTION:** The selected pages will be always saved as separate data. For the whole part of graphic of the model, use the function "Generating part of graphic"!

Generate:

Select the **"Graphic generation"** option, with the sub option **"Pages"** from the **"Edit"** menu. The following dialogue will be displayed:

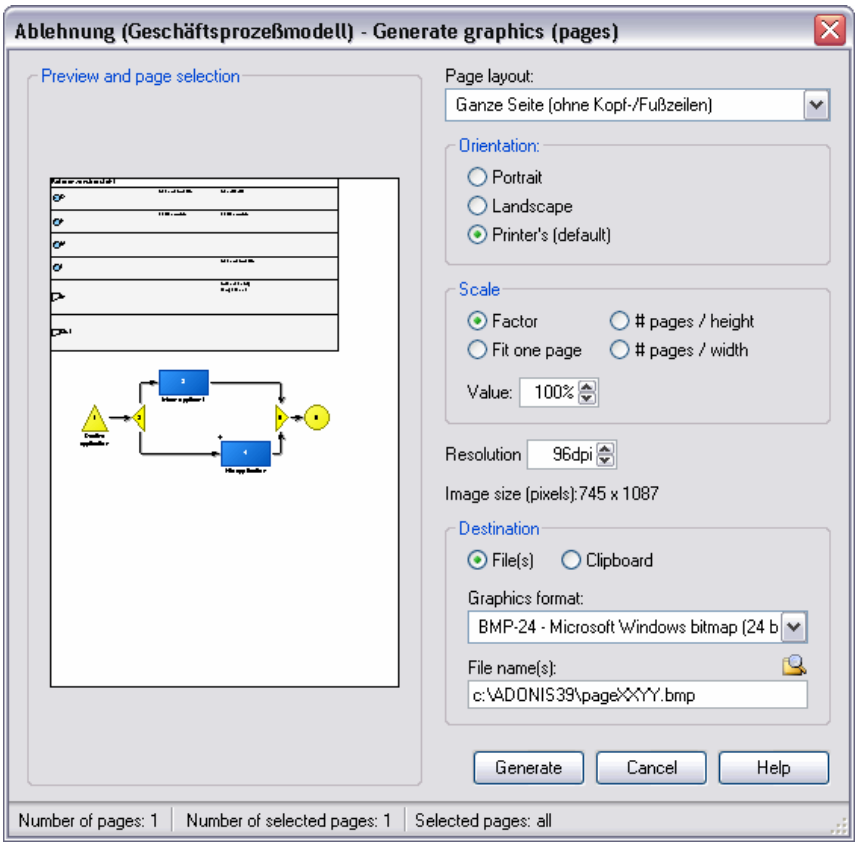


Figure 127: Generate graphics - pages

Set your settings for **"Pages layout"**, **"Layout parameters"** and **"Scale"**. Check the Layout parameters (see p. 194) chapters for details.

The window **"Preview and page selection"** will be updated with every change. This preview allows to select particular pages for the generation of graphics: Click on the pages for which no graphics should be generated. The pages selected will be highlighted grey. If you click on them one more time, you will deselect marked pages.

In the status line you will find information about the multiple pages model print out that is independent from the actual page layout and scale (total number of pages, number of pages to print, numbers of selected pages to print).

In the **"Scale"** field you can define a figure which scales the picture size in relation to its original size by entering a number value. The default value is 100% (the original size of the model). The picture size in pixels will be displayed.

The graphic will be saved with the ideal print resolution 96 dpi (dots per inch.) You can change this number in the **"Resolution"** field.

Hint: To print with the optimal quality, set the scale to 100% and the resolution to 300 dpi.

By clicking on the **"Function"** in the function group, you decide whether the graphic should be exported to the *System's clipboard* or to the *Data*. In case of coping into the clipboard, no additional settings are necessary. If you wish to save the graphics to the data file, you have to provide file's graphic format, name and destination.

ATTENTION: It is not possible to copy more than one page of the model to the clipboard. The graphic will be generated only for the first selected (not grey highlighted) page!

Hint: The contents of the clipboard will be lost as soon as other graphics, texts etc. are copied into the clipboard or when the computer is switched off.

If you wish to save the selected area to a file, you can select one of the available graphic formats in the **"Graphics format"** field. You must enter the path and name of the file in the field **"File name"**. The smart icon supports you in selecting a path and file name.

Click on the **"Generate"** button to generate the graphics. During generation a status window informs you about the current state of the process.

Hint: Please note that a four-figure number will be attached to the file name you select. This ensures that the file names generated are unique. Please take into consideration that 16-Bit Windows and DOS applications can only process files with names consisting of a maximum of 8 characters.

The file names generated this way consist of <*file name*> XXYY.BMP or <*file name*> XXYY.PCX, where "XX" indicates the horizontal page number and "YY" the vertical page number of the appropriate graphic within the full model.

2.2.7.25 Align Objects

Aligning implies that all selected objects (see chap. 2.2.7.3, p. 138) should be placed along the same x-coordinate or y-coordinate. **"Focus"** (see chap. 2.2.7.4, p. 139) is very important here.

Align objects by selecting them and focus (see chap. 2.2.7.4, p. 139) the referenced object to which they should be aligned.

Then select in the "Edit" menu the "Align" menu item as well as the "Horizontally" submenu or the "Vertically" submenu, you can also click on the smart icon (for horizontal alignment) or. (for vertical alignment) in the quick-access bar.

The alignment of objects can be undone using the "Undo" (see chap. 2.2.7.1, p. 137) function.

2.2.7.26 Arrangement Function

When arranging objects and connectors automatically, a predefined function is run on the current model.

ATTENTION: The arrangement function must be defined in the application library for the model type of the model you wish to arrange.

If you want to arrange the objects of a model automatically, follow these steps:

1. Open or activate the model you wish to arrange.
2. Select the menu item "Arrange" from the "Edit" menu.
3. Choose the required arrangement function from the sub menu that appears.

Hint: If no arrangement functions are defined for the model type of the current model, then no arrangement functions will be displayed in the sub menu and you will not be able to automatically re-arrange the objects within the model.

Once you have selected an arrangement function, the automatic model arrangement will be carried out. The single steps of the process and the status are displayed in the window "Arrangement function - status" (see fig. 128).

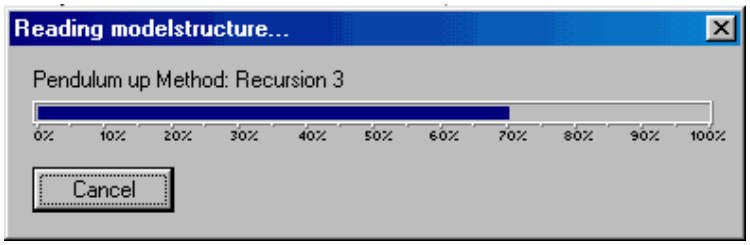


Figure 128: Automatic arrangement - Status

When the model has been arranged, you can see a preview of the new arrangement in the window "Arrangement function - designed arrangement" (see fig. 129).

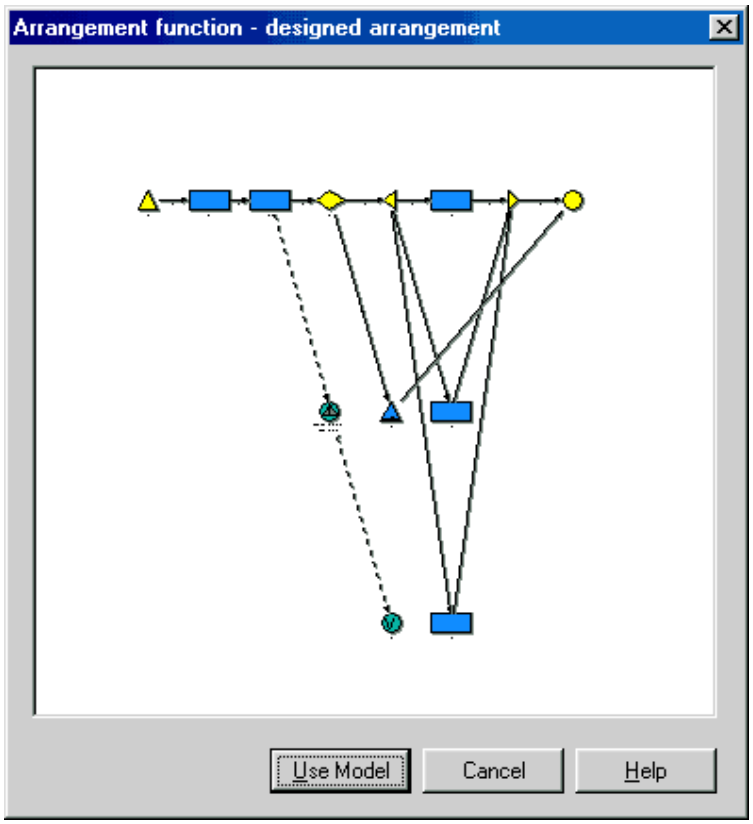


Figure 129: Automatic arrangement - Designed arrangement

Click on the button "Use Model" to adopt the new object positions as displayed in the preview screen. You will then be asked, if you want to replace the current model with the arranged model.

After the model has been successfully arranged, it will be displayed with the new object positions in the model window.

Hint: The start of the arrangement function causes a topological arrangement of the whole graph. This is necessary to assign each object of the model to hierarchically structured layers and to link them with relations. The relative position of the objects within the layers is chosen randomly (see fig. 130).

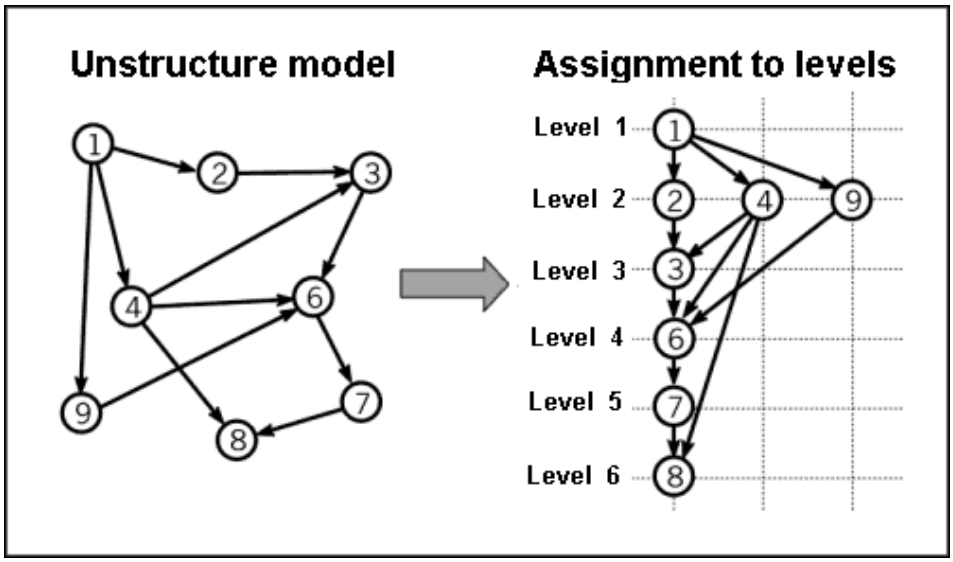


Figure 130: Schematic representation of the automatic arrangement

Hint: The automatic model arrangement can be undone by using the "Undo" (see chap. 2.2.7.1, p. 137) function.

2.2.7.27 Arrange Manually - Arrangement Assistant

The difference between the manual and automatic arrangement functions is that with the manual arrangement function each step of the arrangement is displayed and can be controlled. You can set parameters for each of these steps, which will influence the appropriate step.

If you want to arrange a model manually, follow these steps:

1. Open, activate or create the model you want to arrange (the model currently active will be arranged).
2. Select the menu item "Arrange" from the "Edit" menu.
3. Select the option "Arrangement assistant" from the appearing sub-menu.

The window "<model name> - Select arrangement function" (see fig. 131) appears.

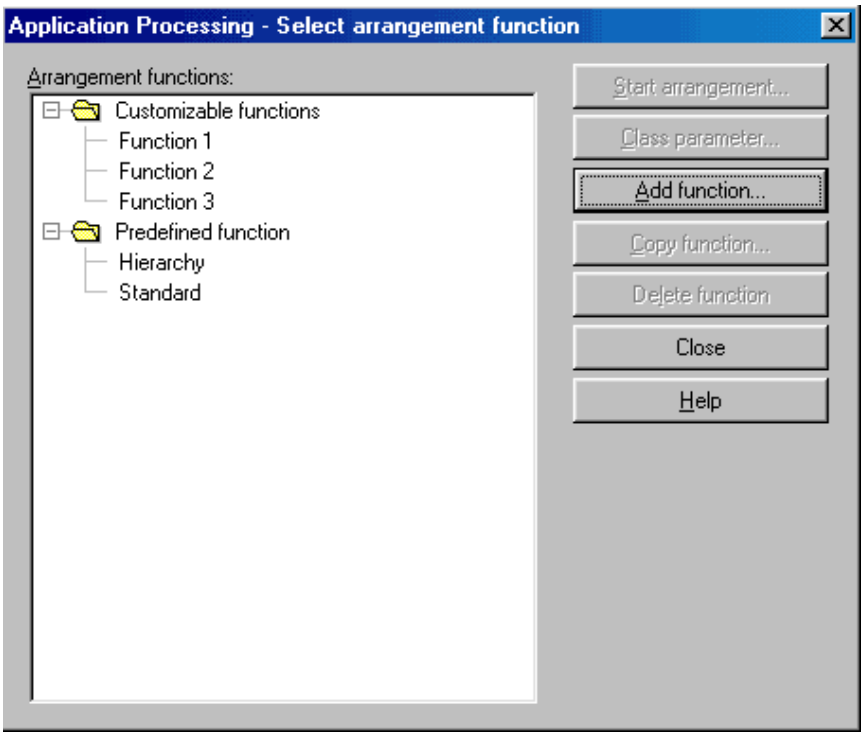


Figure 131: Arrange manually - Select function

The **"Arrangement functions"** list contains customisable and predefined functions. The **customisable functions** can be selected for automatic model arrangement, whereas the **predefined functions** represent the arrangement functions defined in the application library and act as "models" for the customisable functions.

Select one of the customisable functions and click on the button:

- **"Start arrangement"** to start the manual model arrangement, which comprises the following steps:
  - **Bendpoints** (see p. 164),
  - **Minimize crossings** (see p. 165),
  - **Optimise positions** (see p. 166),
  - **Orientation** (see p. 168),
  - **Preview** (see p. 169).
  - **Use model** (see p. 170).
- **"Class parameter"** to edit the class specific parameters (see p. 171).
- **"Add function"** to create a new function (see p. 173).
- **"Copy function"** to copy an existing function (see p. 173).
- **"Delete function"** to delete an existing function.

Hint: The start of the arrangement function causes a topological arrangement of the whole graph. This is necessary to assign each object of the model to hierarchically structured layers and to link them with relations. The relative position of the objects within the layers is chosen randomly (see fig. 130).

Bendpoints

In the "Parameters - bendpoints" window (see fig. 132) settings for the insertion of additional bendpoints can be selected.

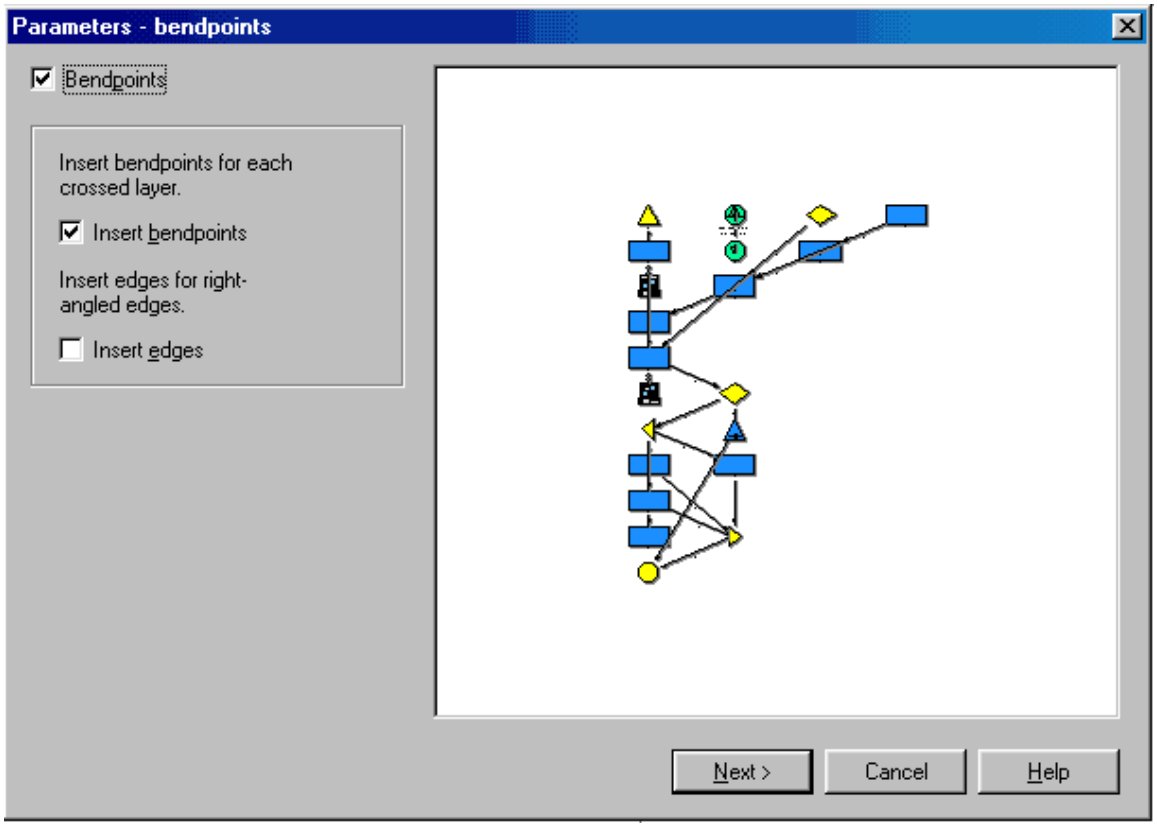


Figure 132: Arrange manually - Bendpoints

- **Bendpoints:** Bendpoints are added to the relations of the model you wish to arrange.
- **Insert Bendpoints:** It may happen during the initial topological arrangement of the model that relations need to be drawn across several layers. Activating the option "Bendpoints" will cause the relations avoid objects of layers in-between the start and the target object. Thus the probability of overlapping objects and relations is minimised. We recommend that this option should be activated.
- **Insert edges:** This option causes the relations of the model to be represented as far as possible only with horizontal and vertical lines. Additional bendpoints have to be inserted in order to achieve this.

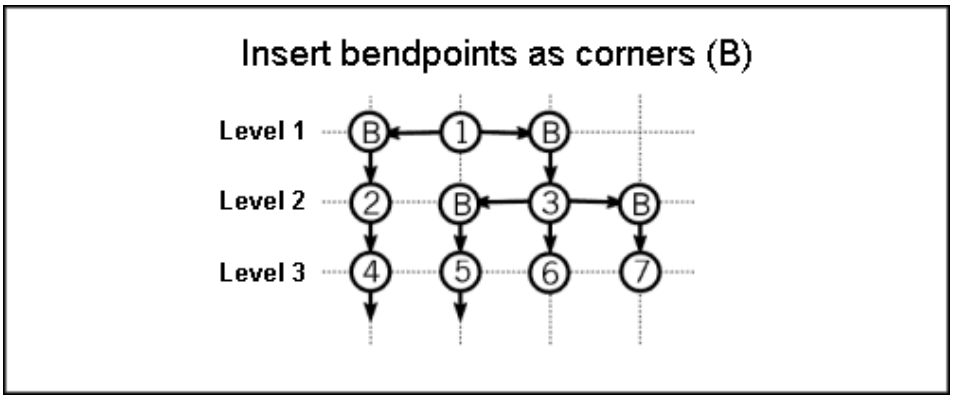


Figure 133: Schematic representation - Paste bendpoints

Crossing Minimisation

You can define the settings for the minimisation of the crossings of relations in the window "Parameters - minimize crossings" (see fig. 134).

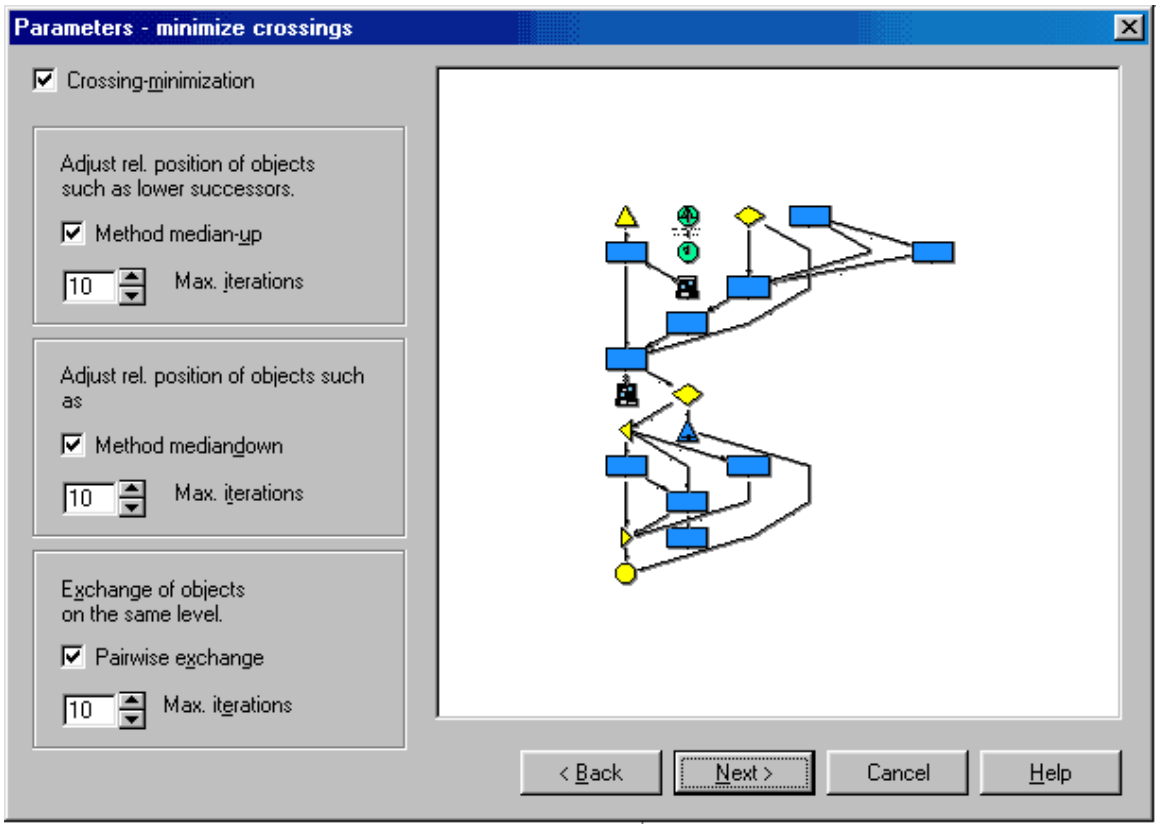


Figure 134: Arrange manually - Crossing-minimisation

- **Crossing-minimisation:** the objects assigned to layers can be moved within the appropriate layer in order to minimize the number of crossings of relations (see fig. 135).

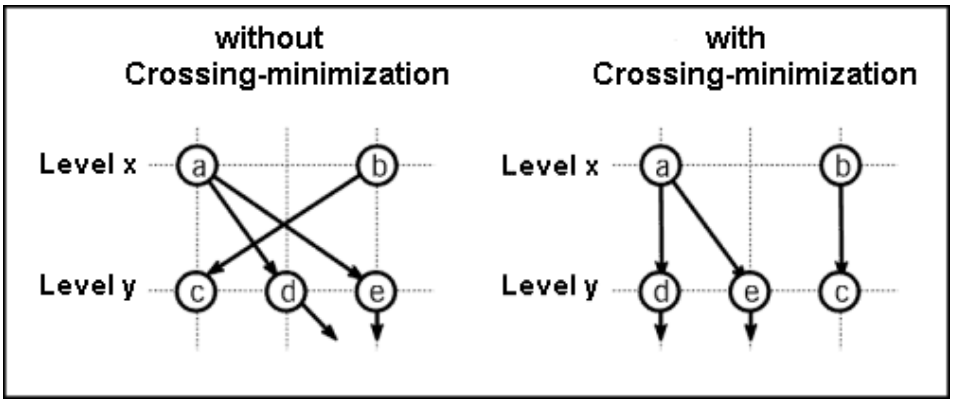


Figure 135: Schematic representation - Crossing-minimisation

- **Method median-up:** crossing-minimisation is carried out from lower hierarchy layers to higher layers. The maximum number of iterations serves to limit the calculation time when arranging very large models.

  Hint: A low number of iterations may influence the quality of the arrangement in a negative way. We recommend a maximum of 10 iterations.

- **Method median-down:** crossing-minimisation is carried out from higher to lower hierarchy layers. The maximum number of iterations serves to limit the calculation time when arranging very large models.

  Hint: We recommend that both the median-up and the median-down options should be activated. In this way, maximum quality within the arrangement will be reached.

- **Pair wise exchange:** This option exchanges two adjacent objects at a time and checks if the number of crossings is thus lowered. Although this option means that the arrangement function takes a little more time, it is recommended. The maximum number of iterations serves to limit the calculation time when arranging very large models. We recommend a maximum of 10 iterations.

Optimise Positions

Settings for optimising the positions of the objects to be arranged can be selected in the "Parameters - optimise positions" (see fig. 136) window.

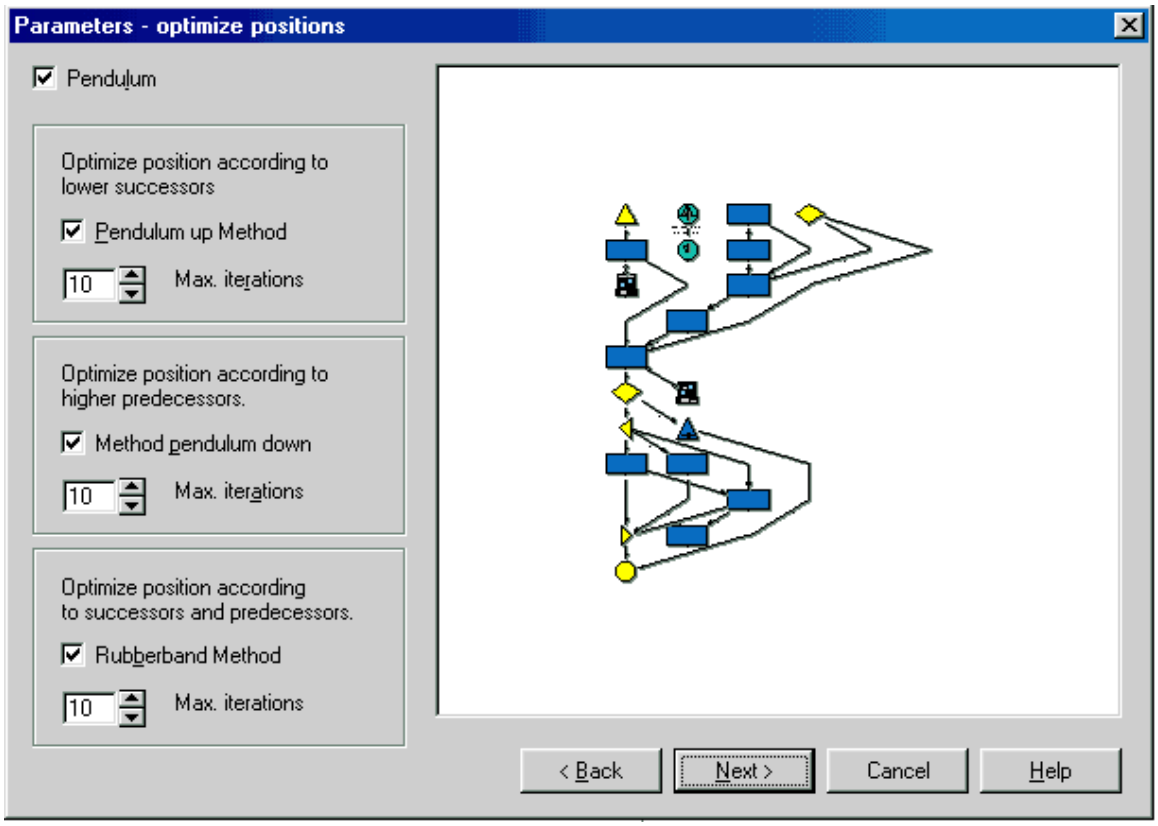


Figure 136: Arrange manually - Optimise position

- **Method pendulum up:** The horizontal position of an object at one layer is optimised according to the position of objects at hierarchically lower layers which are linked with it (as far as possible). We recommend that this option should be activated with a maximum of 10 iterations.

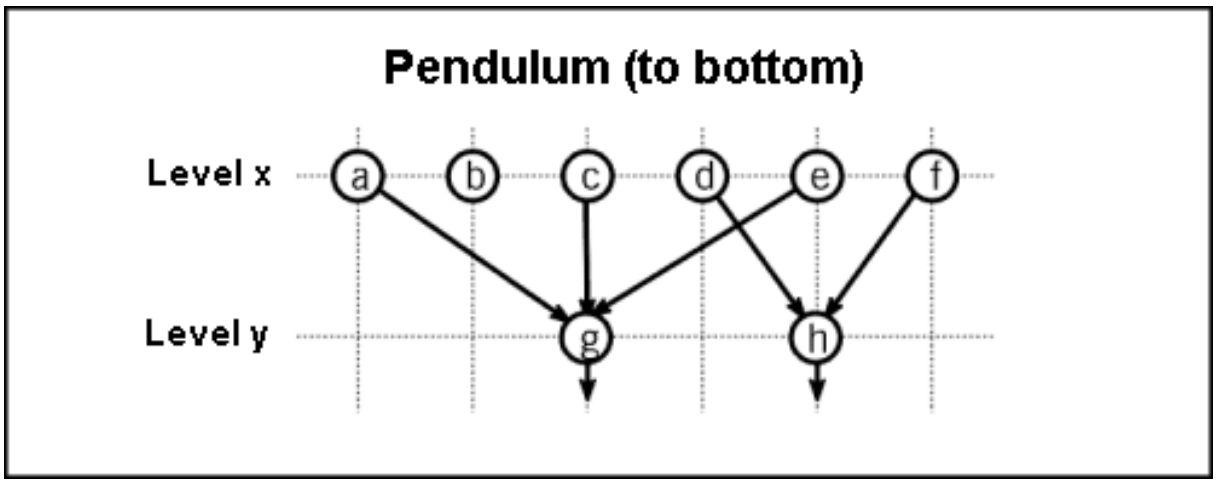


Figure 137: Schematic representation - Method pendulum down

- **Method pendulum down:** The horizontal position of an object at one layer is optimised according to the position of objects at hierarchically higher layers, which are linked with it (as far as possible) (see fig. 137). We recommend that this option should be activated with a maximum of 10 iterations.
- **Method Rubber band:** The position of an object is optimised according to the position of objects linked with it of both hierarchically higher and lower layers (as far as possible). The maximum number of iterations serves to limit the calculation time when arranging very large models.

Orientation

Additional settings to determine the layout of your model are available in the "Parameters - orientation" (see fig. 138) window.

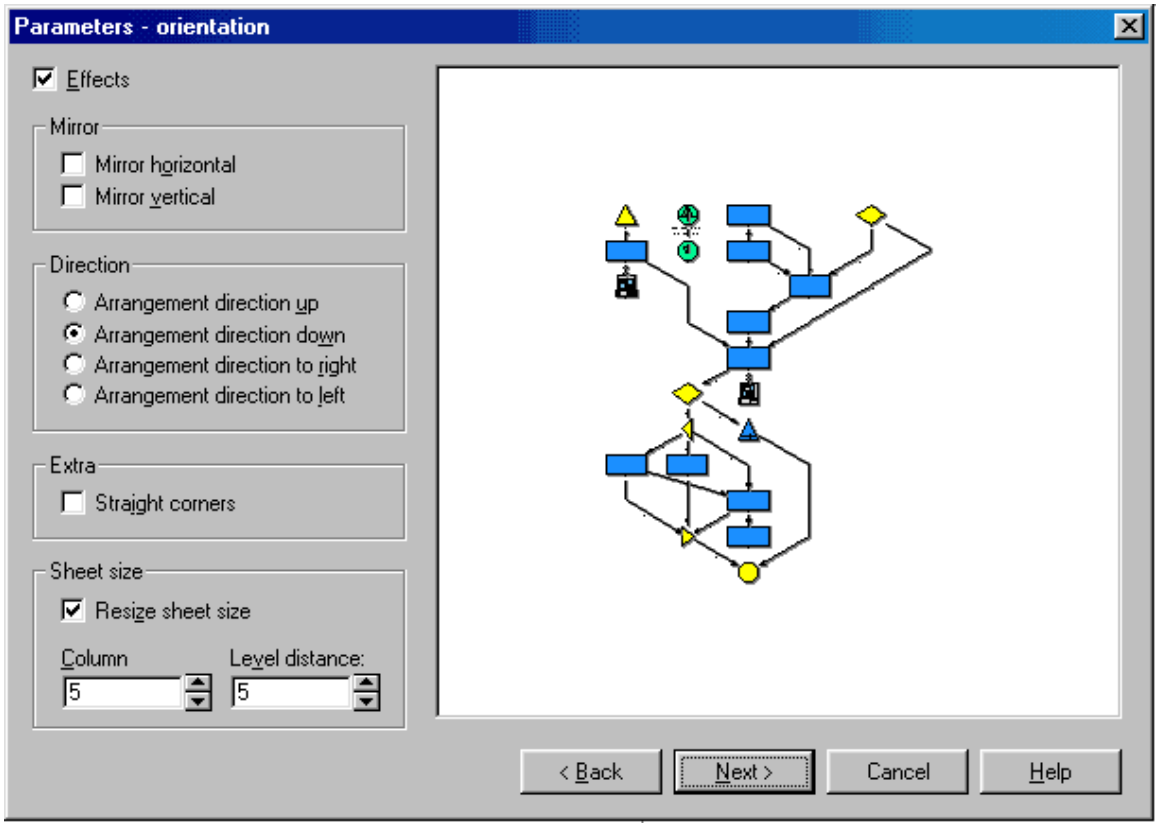


Figure 138: Arrange manually - Orientation

- **Effects:** The model arranged might be additionally manipulated. If this parameter is deactivated, you cannot select any further parameter settings and the model will be arranged according to the standard settings. The standard settings comprise the options "Arrangement direction down" and "Resize sheet size".
- **Mirror horizontal:** The model is mirrored horizontally.
- **Mirror vertical:** The model is mirrored vertically.
- **Arrangement direction up:** The model is arranged from bottom to top.
- **Arrangement direction down:** The model is arranged from top to bottom.
- **Arrangement direction to right:** The model is arranged from left to right.
- **Arrangement direction to left:** The model is arranged from right to left.
- **Straight corners:** Additional horizontal or vertical segments are inserted in all relations stretching over one or more layers, in order to further minimize the overlapping of objects and relations (see fig. 139).

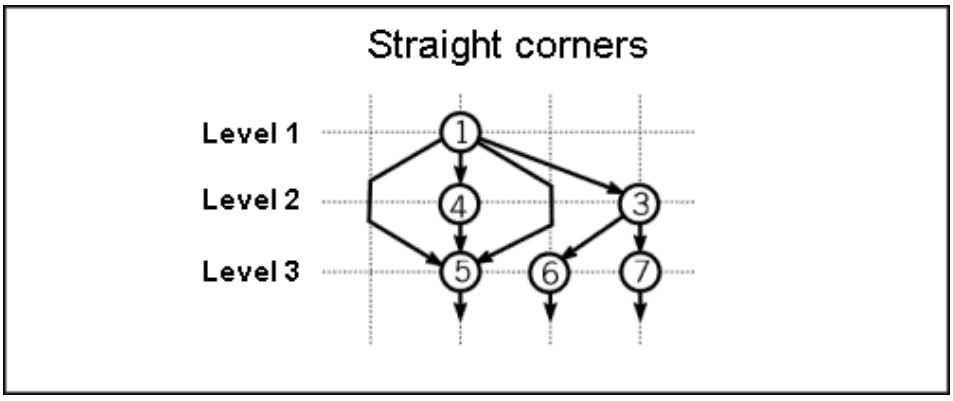


Figure 139: Schematic representation - Straight corners

- **Resize sheet size:** When this option is active the size of the drawing area is automatically adjusted to the space required by the model arranged. The required space results from the number of objects and the column and layer distance defined. If this option is deactivated, the distance between the objects is calculated to fit the objects into the current drawing area.

  Hint: If the size of the drawing area is too small and not resized, the arranged objects may overlap.

- **Column distance:** Defines the horizontal minimum distance between two adjacent objects in cm (will be carried out when "Resize sheet size" is active).
- **layer distance:** Defines the vertical minimum distance between two adjacent objects in cm (will be carried out when "Resize sheet size" is active).

Preview

Once the final arrangement step ("Orientation") has been completed, the window "Arrangement function - designed arrangement" (see fig. 140) appears. It shows a simplified representation of the resulting arrangement.

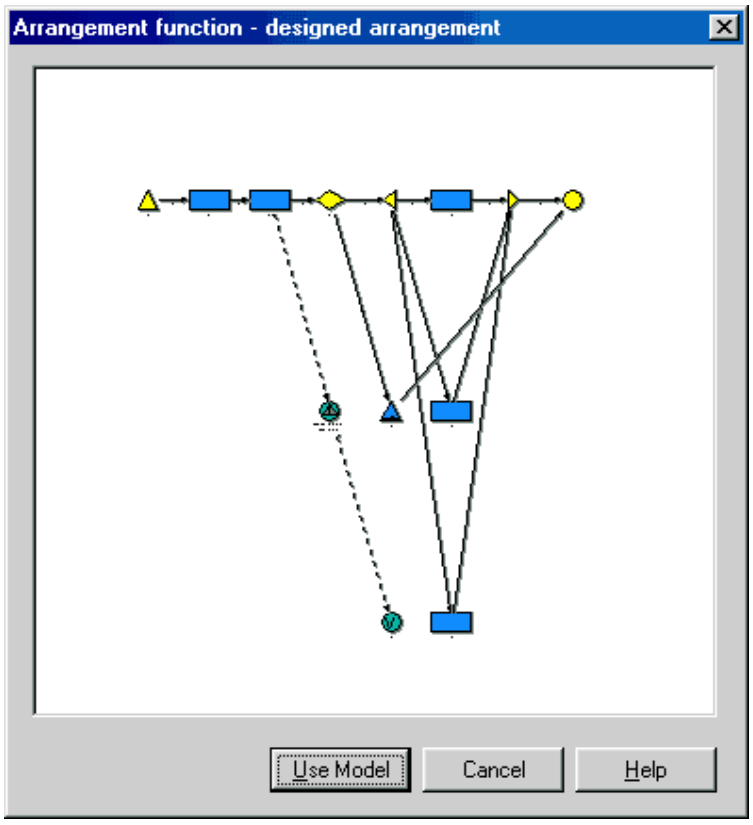


Figure 140: Arrange manually - Designed arrangement

Click on the button "Use model", if you want the current model to be changed according to the arrangement displayed in the preview window.

Use Model

If you choose to use the model, a window (see fig. 141) opens up, in which you can activate the following options:

- **"Save arranged model":** the model shown in the preview window is used.
- **"Save arrangement function":** the parameter settings of the current arrangement function are saved.

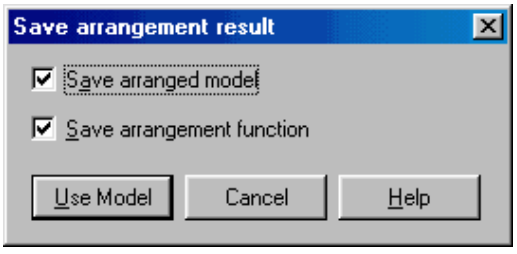


Figure 141: Arrange manually - Save arrangement

Click on the button "Use model" to use the model and/or to save the arrangement function.

Editing Class-specific Parameters

You can edit the class specific parameters by clicking on the button "Class parameter" in the window "Select arrangement function". The window "<function name> - Parameter configuration" (see fig. 142) will be displayed.

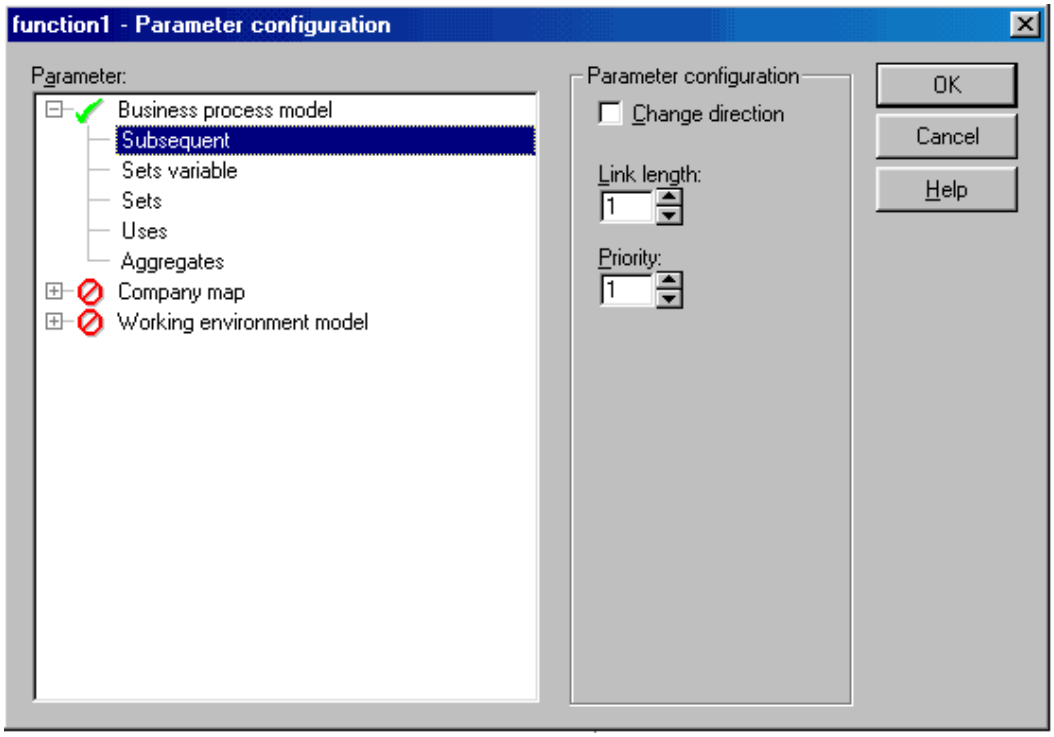


Figure 142: User specified arrangement function - Parameter settings

Define which model types should be able to use this arrangement function here. Assign a model type to an arrangement function by double-clicking on the appropriate model type contained in the list of model types. The icon in front of the name of the model type indicates the current assignment status. The icon ✓ means "assigned" and the icon ⊘ means "not assigned".

Select the relations of an assigned model type and edit the following parameter settings:

- **Change direction:** By activating this parameter setting the direction of an instance of this relation class is turned around internally (see fig. 143). If two objects A and B, which are linked through the relation class X have the order "B followed by A" and the order "A followed by B" is actually desired, then direction change should be activated for class X.

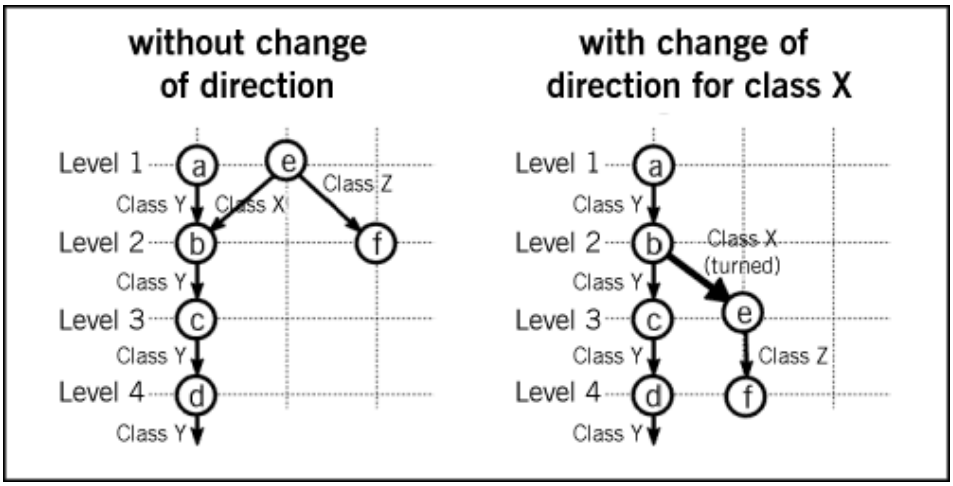


Figure 143: Schematic representation - Change direction

- **Link length:** By providing an integer value n for this parameter you can define how many layers (n-1) should be spanned by an instance of this class (see fig. 144). If instances of different relation classes meet at an object, the maximum number of layers is calculated.

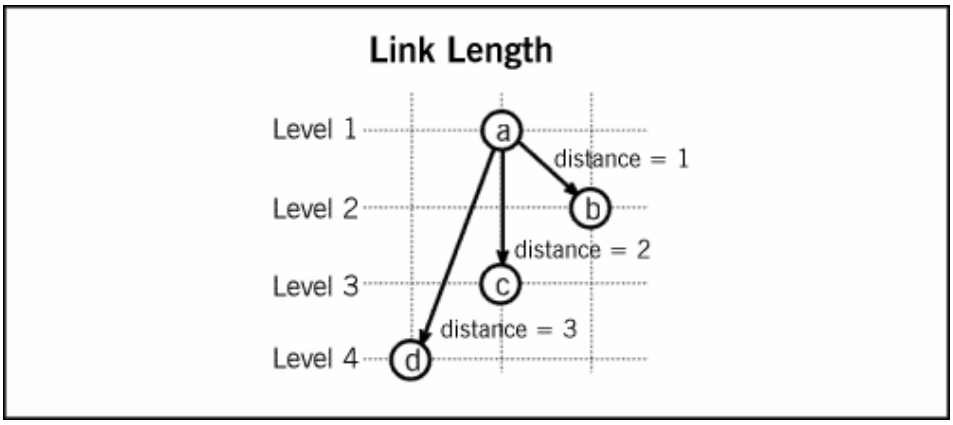


Figure 144: Schematic representation - Level length

- **Priority:** By providing an integer value n for this parameter you can specify how important an instance of this relation class is in comparison with instances of other relation classes. When arranging the objects within a model it is more likely that two objects joined with this relation will be horizontal or parallel with each other (see fig. 145). E.g. objects, which are joined by a relation of high priority are more likely to be positioned above or beside each other.

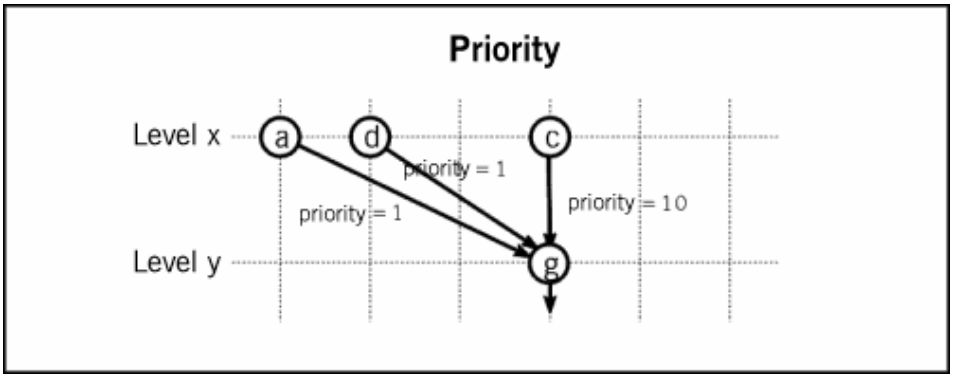


Figure 145: Schematic representation - Priority

Add Function

Enter the name of the new function into the field "New function name" in the window "Add arrangement function" (see fig. 146). Then click on the "Insert" button to create the new arrangement function.

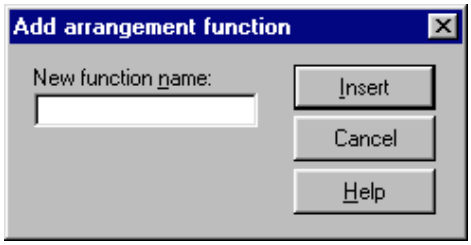


Figure 146: Add arrangement function

All common parameters are initialised with default values for the new arrangement function. The class specific parameters will be initialised for all model types of the current application library and all classes of each model type.

Copy Function

Enter the name of the new arrangement function into the field "New function name" in the window "Copy arrangement function" (see fig. 147). Then click on the button "Insert" to create a copy of the arrangement function previously selected.

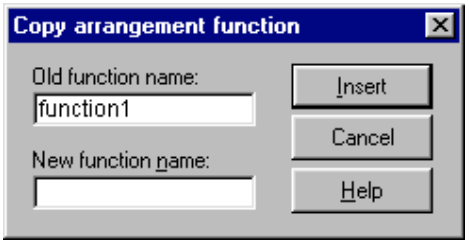


Figure 147: Copy arrangement function

2.2.8 Context Menu

The context menu is to be opened by clicking on the right mouse button. There you can call functions from this current menu, i.e. frequently used functions will be proposed depending on the position of the mouse pointer at the moment of the menu display.

You can open up context menus for objects (see chap. 2.2.8.2, p. 174), connectors (see chap. 2.2.8.3, p. 176) and the drawing area (see chap. 2.2.8.1, p. 174). Right-click on an object, a connector or on a blank area of the drawing area to open the appropriate context menu.

The menu only displays functions, which can currently be called. For example the function "Paste" is only displayed in the context menu of the drawing area, if objects of the same model type have previously been copied or cut.

Hint: Additionally to the (above described) context menus in the modelling, other context menus are available in the Selection (see chap. 3.1, p. 41) and in the ADOxx browser (see chap. 8., p. 66).

2.2.8.1 Context Menu of the Drawing Area

You can open the context menu of the drawing area (see fig. 148) by right clicking on the (blank) drawing area.

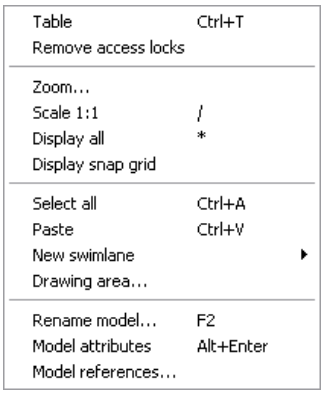


Figure 148: context menu of the drawing area

The context menu contains the following functions:

- **Table** for switching to tabular model display (see chap. 2.2.10, p. 181).
- **Remove access locks** for unlocking the mouse access (see chap. 2.2.11.2, p. 188).

  (This menu item is only available if the mouse access of at least one class is locked.)
- **Zoom** for model display using a zoom factor (see chap. 2.2.11.3, p. 189).
- **Scale 1:1** for model display using a zoom factor (see chap. 2.2.11.3, p. 189) of 100%.

  (This menu item is only available if the zoom value does not currently equal 100%.)
- **Display all** to display the whole model in the model window.
- **Display snap grid** to display the snap grid (see chap. 2.2.11.4, p. 190).
- **Select all** to select all objects and connectors in the model (see chap. 2.2.7.3, p. 138).
- **Paste** to paste copied or cut objects and/or connectors (see chap. 2.2.7.17, p. 151).
- **New simlane** to place a new swimlane (see chap. 9., p. 91) into the model.

  (This menu item is only available if placing swimlanes is allowed.)
- **Drawing area** to open the window "Drawing area" (see fig. 151), in which you can increase or decrease the size of the drawing area (see chap. 2.2.9, p. 176).
- **Rename model** to change the name of the model (see chap. 2.4.8, p. 262).
- **Model attributes** to open the notebook which contains the model attributes (see chap. 2.4.9, p. 264).
- **Model references** to display the incoming inter-model references of the model (see chap. 2.4.11.2, p. 267).

2.2.8.2 Context Menu of Objects

You can open the object context menu (see fig. 149) by clicking on an object with the right mouse button.

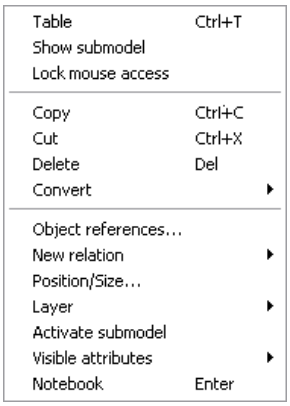


Figure 149: Object context menu (example)

The context menu contains the following functions:

- **Table** to switch to tabular model display (see chap. 2.2.10, p. 181).
- **Show submodel** to display the referenced submodel within the current model (see p. 200).

  (This menu item is only available if displaying of a submodel is possible.)
- **Lock mouse access** to lock or unlock the mouse access (see chap. 2.2.11.2, p. 188).
- **Copy** to copy (see chap. 2.2.7.15, p. 149) the object.
- **Cut** to cut (see chap. 2.2.7.16, p. 150) the object.
- **Delete** to delete (see chap. 2.2.7.18, p. 152) the object.
- **Convert** to convert an object into an other object class (see chap. 2.2.7.23, p. 155).

  (This menu item is only available if the conversion is possible.)
- **Object references** to display a list of all models in which references (see chap. 2.4.11, p. 265) to the object selected exist. (This menu option is only available if inter-model references on objects are defined in the application library.)
- **case/4/0** for the creation and management of references on objects of the CASE Tool case/4/0 (see chap. 7.3, p. 517)

  (This function is only available if the additional component "case/4/0 coupling" has been acquired and the application library has been defined appropriately.)
- **New relation** to create new relations to or from the object selected (see chap. 2.2.6, p. 132).
- **Position/Size** to define the position and size of objects (see chap. 2.2.7.7, p. 142)
- **Layer** to place an object on the drawing area in foreground or in background.
- **Activate submodel** to activate a referenced submodel (see chap. 2.3.3, p. 255).

  (This menu item is only available if activating of a submodel is possible.)
- **Visible attributes** opens a window in which the visualised attribute value can be edited (see chap. 2.2.7.13, p. 148).

  (Depending on the number of visualised attributes for a particular object either the attribute name (if only one attribute is visualised) or else the term "Visible attributes" which contains a submenu for each attribute actually displayed, will appear in the context menu).
- **Notebook** to open the ADOxx Notebook, in which object attributes (see chap. 2.2.7.12, p. 146) can be edited.

2.2.8.3 Context Menu of Connectors

The connector context menu (see fig. 150) can be opened by right clicking on a connector.

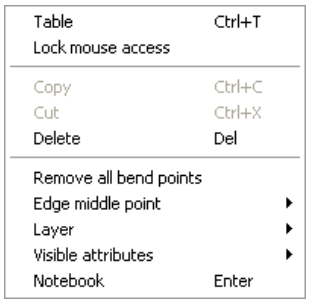


Figure 150: Connector context menu

The context menu contains the following functions:

- **Table** to switch to the tabular model display (see chap. 2.2.10, p. 181).
- **Lock mouse access** for locking or unlocking the mouse access (see chap. 2.2.11.2, p. 188).
- **Delete** to delete the connector (see chap. 2.2.7.18, p. 152).
- **Delete all bendpoints** to delete all bendpoints defined within the connector.
- **Edge middle point** to define the edge middle point of the connector (see chap. 2.2.7.14, p. 148).
- **Layer** to place the connector on the drawing area in foreground or in background.
- **Visible attributes** opens a window in which the visualised attribute value can be edited (see chap. 2.2.7.13, p. 148).

  (Depending on the number of visualised attributes for a particular object either the attribute name (if only one attribute is visualised) or else the term "Visible attributes" which contains a submenu for each attribute actually displayed, will appear in the context menu).
- **Notebook** will open the ADOxx Notebook in which connector attributes can be edited (see chap. 2.2.7.12, p. 146).

2.2.9 Drawing Area

The drawing area (see chap. 2.2.1, p. 127) is a rectangle enclosing all objects and connectors of a model. It is bordered by a dark grey outline, which can be seen in the model window. The drawing area is automatically extended when an object or connector is placed to the right or below the drawing area.

- **Resize:** (see chap. 2.2.9.1, p. 176) The edges can be almost freely moved.
- **Add an empty area:** (see chap. 2.2.9.2, p. 178) It is possible to create a space in the middle of given model.

  Hint: The minimum size of the drawing area is 10x10 cm. It is determined by the arrangement of objects and connectors. The maximum width and height is 50m.

2.2.9.1 Resize the Drawing Area

The drawing area can be increased or decreased in a two different ways:

Automatic expansion

The drawing area will be automatically expanded, as soon as one object or connector is placed on the right or below the boarder of the drawing area.

Manually:

To change sizes of the drawing area manually, place the cursor in editing mode on the frame of the drawing area; the cursor then changes to a black double headed arrow (, , or). and you can enlarge or reduce the drawing area by pressing the left mouse button and dragging the mouse in the appropriate direction (while keeping the mouse button pressed).

The "Drawing area" function:

You can change the drawing area by calling the function "Drawing area". There are several ways to call this function:

- Select the option **"Drawing area"** in the **"Edit"** menu.
- Select the option **"Drawing area"** from the **context menu** of the drawing area.
- Click on the appropriate **smart icon** in the quick-access panel.

Once this function is called, the window "Drawing area" (see fig. 151), in which you can increase or decrease the size of the drawing area, is displayed.

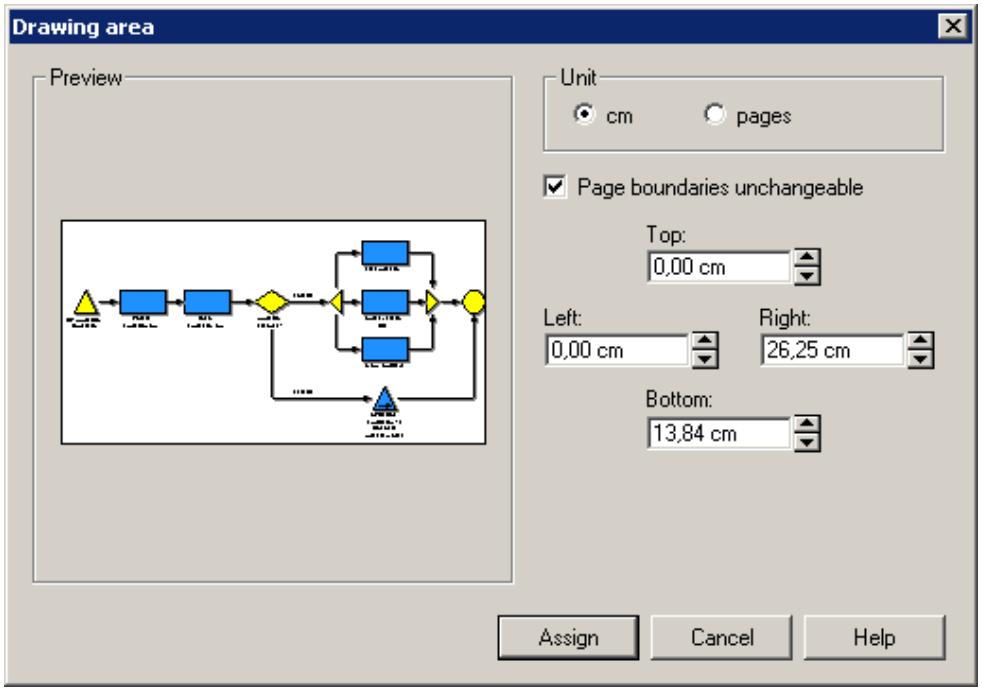


Figure 151: Drawing area

The "Drawing area" window shows the current coordinates of the model's margins. You can move the drawing area's four boundaries by clicking on the appropriate buttons. The extent of the changes depends on the page layout selected and on the arrangement of the objects modelled as well as the space they require. You can also directly change the drawing area by entering the required values into the corresponding input fields (top, bottom, left, right).

The input of a negative value into the fields "Left" and "Top" leads to an extension of the drawing area, as a coordinate value higher than the current one always means "further right" or "further down" respectively.

> **Hint:** Should you enter invalid values, they will be corrected automatically. When calling the function "Drawing area" again, the values in the input fields "Top" and "Left" are set to "0", since this is the coordinate system's origin.

The option **"Unit"** allows you to choose between "cm" and "pages". If you are working with "pages", changes to the drawing area depend on the size of the pages as defined in the page layout.

The option **"Page transitions unchangeable"** provides support for the organization of the drawing area and the objects arranged in it. When this option is activated (standard setting) the current page number is maintained, i.e. the system will always insert whole pages when the drawing area is extended. The page related arrangement of the model will not be changed, which is important especially when you extend the drawing area upwards or to the left.

Click **"Assign"** to close the window **"Drawing area"** and store the settings.

> **Hint:** Changes to the drawing area can be undone by calling the function "Undo" (see chap. 2.2.7.1, p. 137) in the menu "Edit" or by clicking on the respective smart icon on the quick-access bar.

2.2.9.2 Adding/Removing Empty Space

ADOxx provides the "Empty spacing tool", that easily adds or removes any size of empty space on the drawing area. With use of this tool, it is easy to obtain empty space within a model in a fast manner.

Start Empty space tool:

The empty spacing tool can be started in two different ways:

- By clicking on the icon in The modelling bar or
- by pressing **<*Alt*> + double click** in the model window.

The grey cross-hair appears, whose intersection is located on the clicked point (see fig. 152).

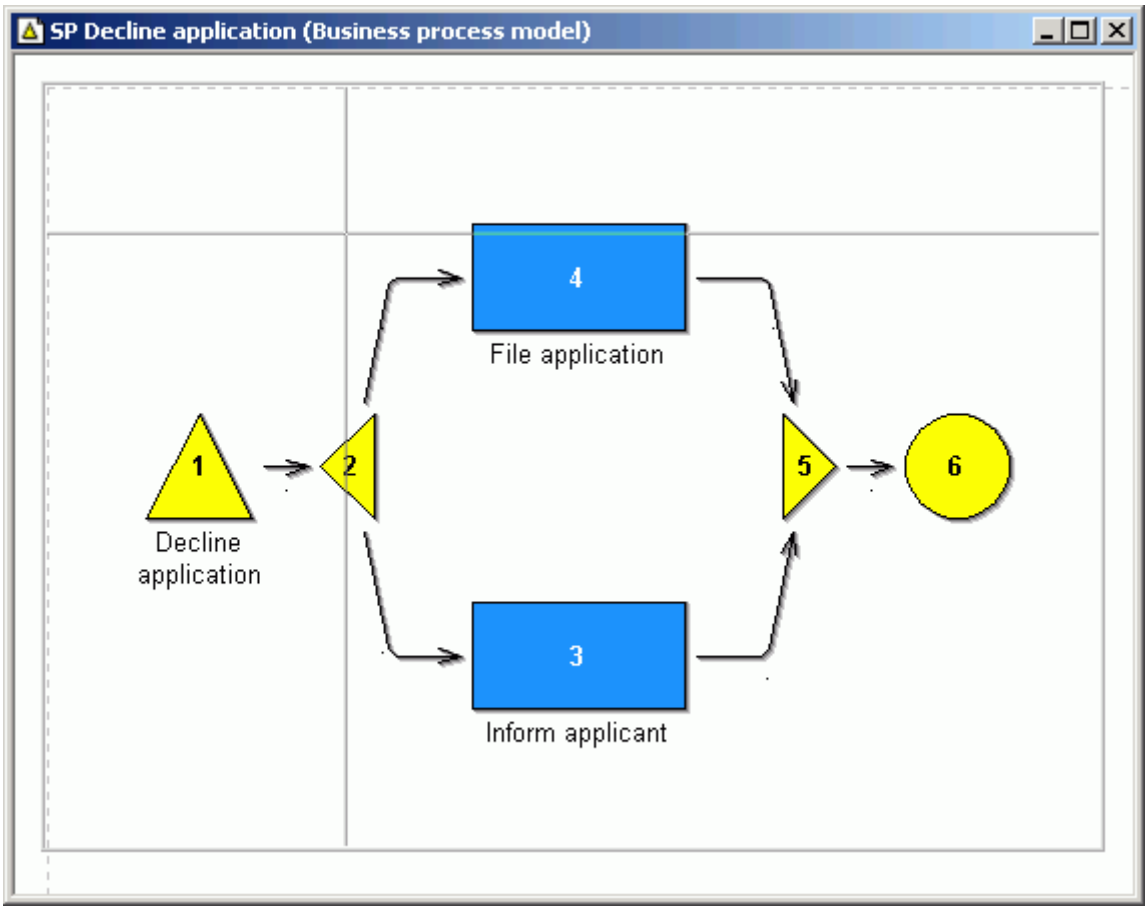


Figure 152: The empty spacing tool

It doesn't make any difference, whether you click on the empty area or on the object. The "empty spacing tool" remains active and visible until you click on any other place in the model window.

Creating empty space:

The tool can create empty space in three different ways:

- stretching the entire height of the drawing area
- stretching the entire width of the drawing area
- freely defined rectangular

For the **empty space with the same height**, you point the mouse pointer on the vertical line of the tool's threat (the symbol changes into ↔. Drag the line to the right with pressed mouse key. The thread will change the colour to blue, and the starting position, will be marked with the dashed line. As soon as you release the mouse key, all objects that were located right from the original position will be moved further right, so that they will be located on the right from the new position. In between a new area the empty space was created. The existing connectors will be respectively extended.

The same situation is with the **empty space with the same width**. With the horizontal threat of the tool (the mouse symbol ↕), objects are moved further down.

The similar mechanism is used, when you want to add **free defined rectangle**. To do so, pull the intersection point of the cross hair (mouse symbol) to the right and down. The blue rectangle will be created:

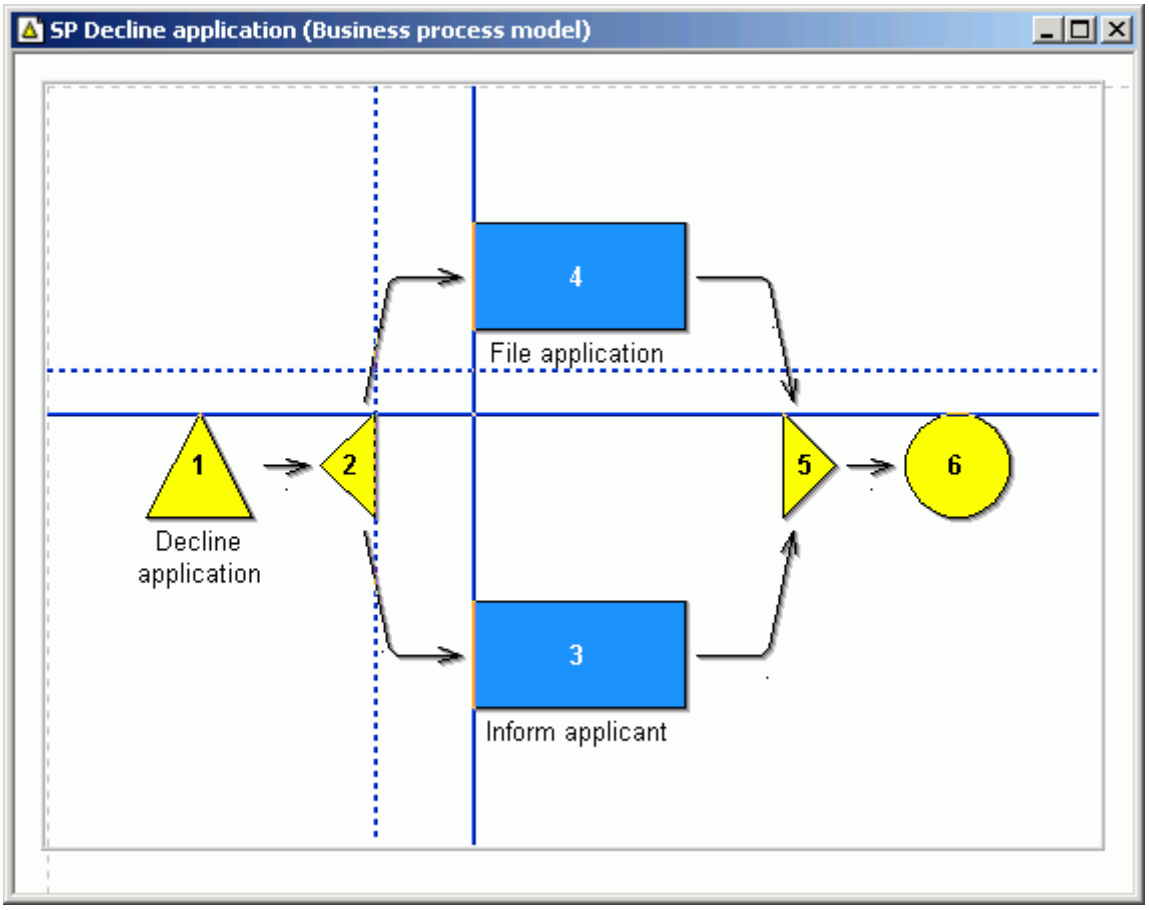


Figure 153: Generating space in the rectangle shape

By releasing the mouse key, the objects will be moved to the right *and* down.

Hint: Activate the "empty spacing tool" so that cross hair's intersection point will be at the beginning of the object.

Removing an empty area:

If there is too much space, it can be removed with use of empty spacing tool. This is done in the opposite way: Through **shifting the cross hair to the left and/or up** the object located right, and below will be moved to the left and up. Existing connectors will be shortened respectively.

Hint: Empty space that is created or removed within swimlanes or aggregations, will enlarge or reduce them.

Hint: If the snap grid (see chap. 2.2.11.4, p. 190) is active then the enlargement or reduction happens in corresponding steps. If this is not wanted, deactivate the snap grid or stretch while holding the <*Crtl*> key.

Hint: To undo the empty space editing, select the option "Undo" (see chap. 2.2.7.1, p. 137) from the menu "Edit" or click on the corresponding smart icon in the quick-access bar.

2.2.10 Tabular Model View

Tabular model display enables you to easily edit the attribute values of objects of a specific class or of connectors of a specific relation type for the current model. If you have a lot of information to add to a model - for example you are adding execution times or descriptions to all activities - it is quicker and easier to do this using the tabular view of the model rather than the notebooks.

To call the tabular model display, activate the model and carry out one of the following steps:

- Select the option **"Table"** from the **"View"** menu.
- Select the option **"Table"** from the context menu of the drawing area (see chap. 2.2.8.1, p. 174).
- Click on the smart icon in the quick-access bar.
- Press the hot key**<Ctrl>+T**.

The tabular model display is shown in the model window (see fig. 154) on the working area of the ADOxx browser (see chap. 8., p. 66).

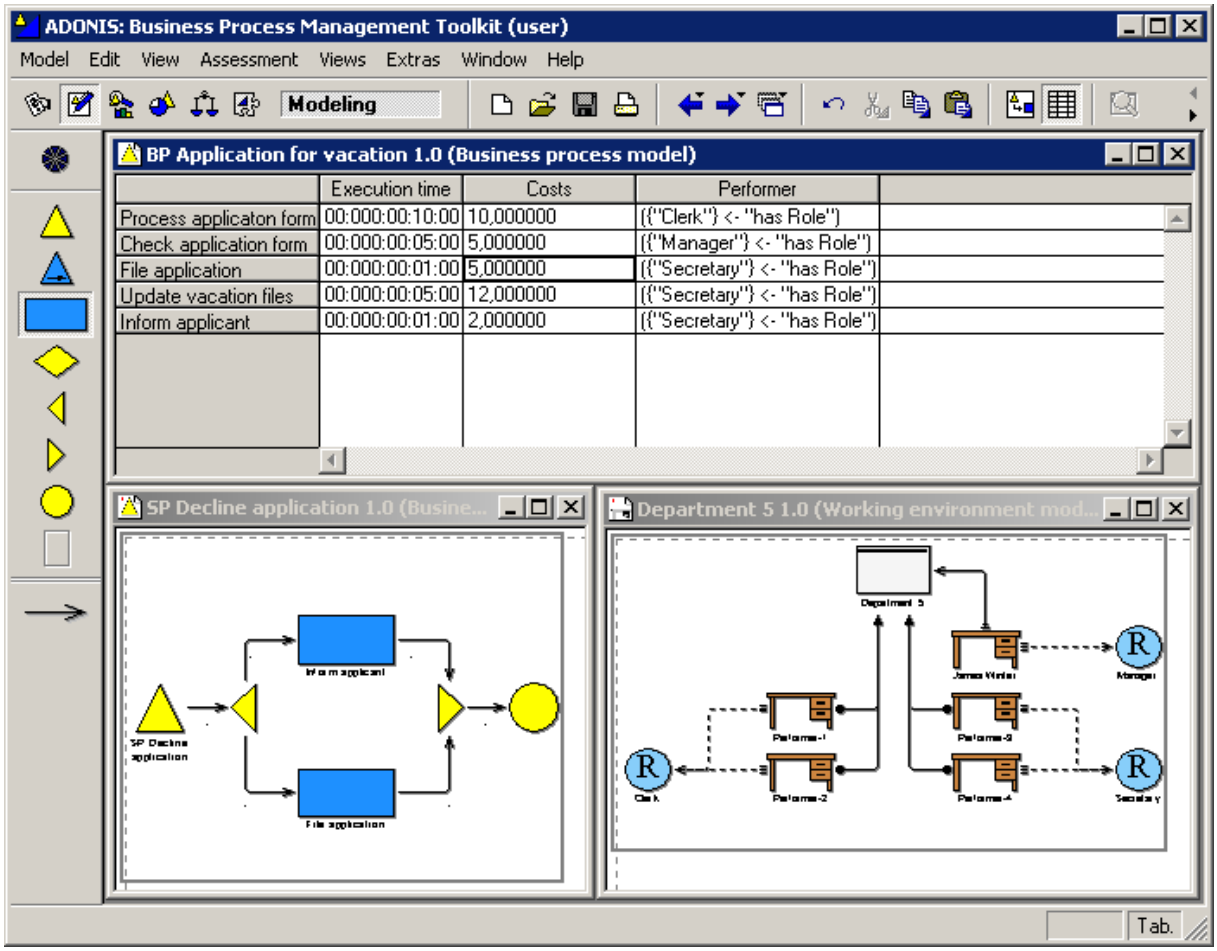
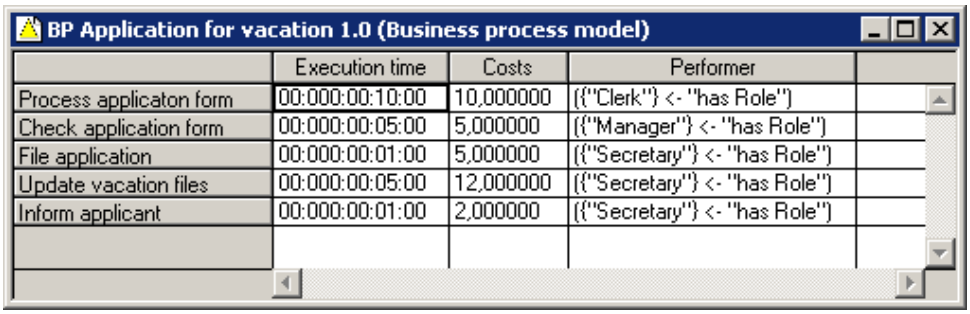


Figure 154: Tabulated model view (example)

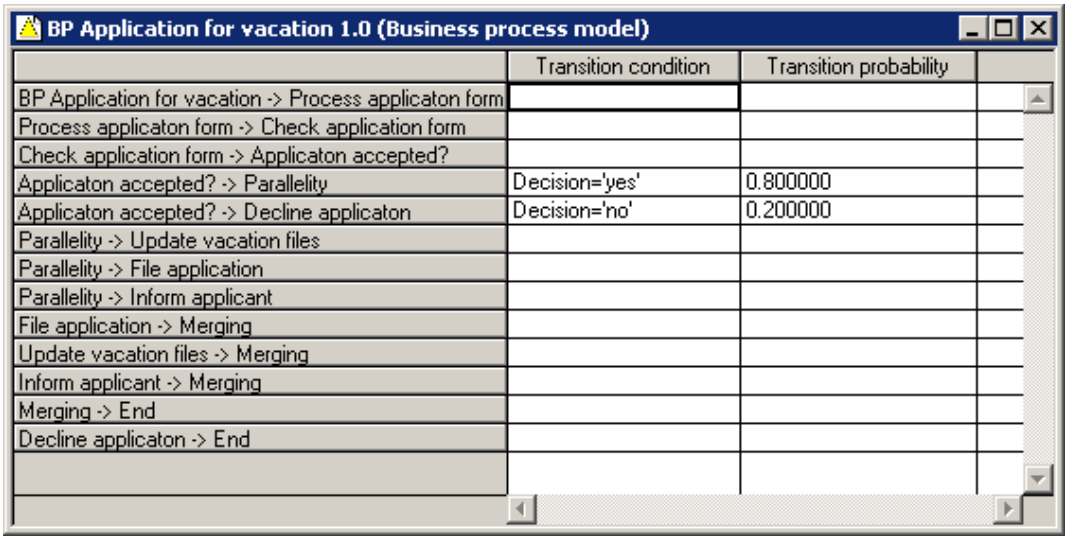
Once you have selected a class (see chap. 2.2.10.1, p. 182) in the modelling panel, all attributes (in the columns) of the objects of the selected class contained in the module (see fig. 155) or of the connectors of the selected relation (in the lines) (see fig. 156) will be displayed. The order of the attributes from the left to the right corresponds to the order of the attributes in the ADOxx Notebook.



	Execution time	Costs	Performer
Process applicaton form	00:000:00:10:00	10,000000	({"Clerk"} <- "has Role")
Check application form	00:000:00:05:00	5,000000	({"Manager"} <- "has Role")
File application	00:000:00:01:00	5,000000	({"Secretary"} <- "has Role")
Update vacation files	00:000:00:05:00	12,000000	({"Secretary"} <- "has Role")
Inform applicant	00:000:00:01:00	2,000000	({"Secretary"} <- "has Role")

Figure 155: Tabular model representation- Display objects of a class

The first (grey) column contains either the name of the objects (see fig. 155) or the starting- and target object of the connectors (see fig. 156).



	Transition condition	Transition probability
BP Application for vacation -> Process applicaton form		
Process applicaton form -> Check application form		
Check application form -> Applicaton accepted?		
Applicaton accepted? -> Parallelity	Decision='yes'	0.800000
Applicaton accepted? -> Decline applicaton	Decision='no'	0.200000
Parallelity -> Update vacation files		
Parallelity -> File application		
Parallelity -> Inform applicant		
File application -> Merging		
Update vacation files -> Merging		
Inform applicant -> Merging		
Merging -> End		
Decline applicaton -> End		

Figure 156: Tabular model view- Display connector of a relation

To edit attribute values either click on the corresponding cells or steer the cells using the cursor key and press the **<Enter>** key.

Additionally (through context menu - right mouse button), you can save the attribute values to a file, copy it into the clipboard or print it. You can find (see chap. 8.12, p. 80) attribute values or sort (see p. 185) the objects or connectors according to a particular attribute.

Hint: Write protected attributes are marked with a minus (-) attached to the name; they **cannot** be edited.

2.2.10.1 Class Selection

To change from the currently displayed class or relation, click on the symbol of a different class or relation in the modelling bar (see chap. 2.2.2, p. 128). The objects or connectors of that class will then be displayed.

Hint: It is only possible to select classes or relations, which exist within your model.

Figure 157: Overview button

Alternatively, you can click on the "Overview" button (see fig. 157) in the modelling bar. An overview of all available classes and relations, including the corresponding number of instances will be displayed (see fig. 158). Double-click on a class or relation to launch the appropriate display.

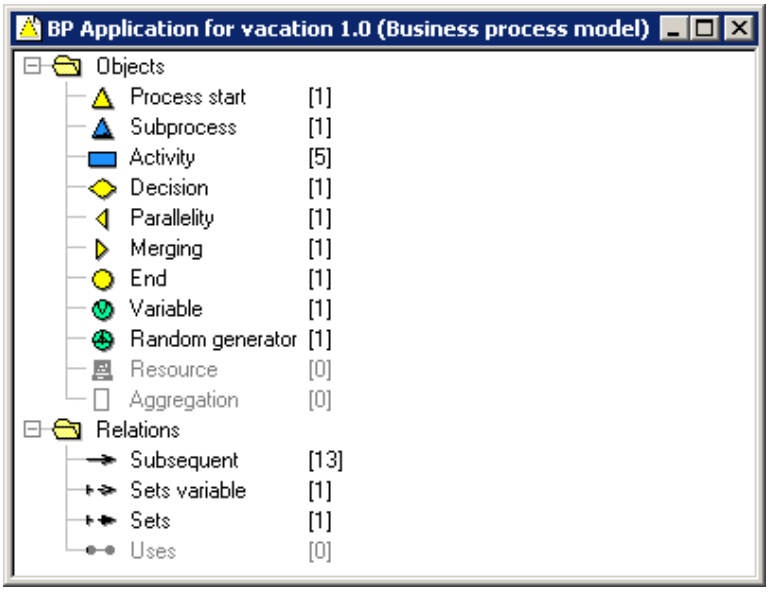


Figure 158: Overview in the tabular modelling

If you select the tabular model display in the object context menu (see chap. 2.2.8.2, p. 174) or in the connector context menu (see chap. 2.2.8.3, p. 176), then the table will list all objects or connectors of the selected class. The class or connector originally selected will be highlighted (see p. 185).

When you first view a class or relation as a table, all attributes of the object or of the connector will be displayed. If you select the "select attributes" menu item ("view" menu or context menu) you can change from attribute columns (see chap. 2.2.10.2, p. 183) displayed at any time.

Hint: The selection of displayed attributes will be saved and remains valid for all the models of the same type, until you select other attributes or exit ADOxx.

2.2.10.2 Display Attribute Columns

The window "Attribute columns - <*class name*>" (see fig. 159), in which you can select the attributes you wish to be displayed, appears.

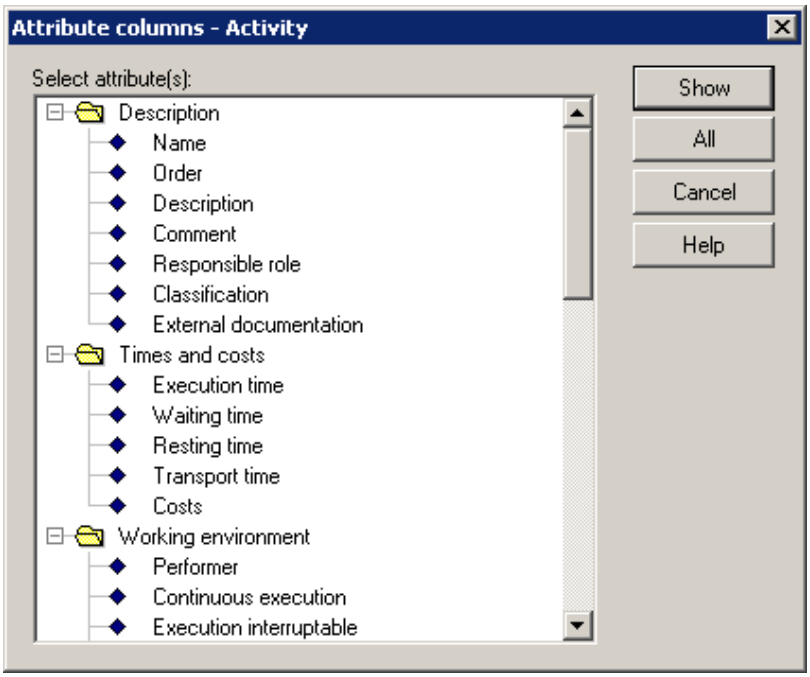


Figure 159: Select attributes for the display

The attributes displayed in this window reflect the structure of the ADOxx Notebook, i.e. the attributes are divided into chapters.

Each attribute is preceded by a symbol which indicates if the attribute is write protected ( ) or not ( ).

Select those attributes you want to be displayed in the ADOxx browser and click the "Show" button.

Hint: The selection of the attributes to be displayed remains saved and valid for all models of the same type, until you select other attributes or exit ADOxx.

2.2.10.3 Edit Attributes

To edit attribute values you can either:

- Double-click the corresponding cells, or
- Place the (black) cell frame on the corresponding cells using the cursor keys or the tabulator key and then press the enter key.

Depending on the attribute type the attribute value can either be edited directly in the ADOxx browser (see chap. 8., p. 66) or in an appropriate input window.

The support dialogs available in the ADOxx Notebooks for editing attribute values are also available here. You can complete direct editing of a particular field in the ADOxx browser by pressing the enter key or by mouse-clicking on any space within the ADOxx browser. Should the attribute value entered be invalid (for example due to the input of non-numerical instead of numerical characters), an information message will be displayed and you will have the option to continue editing the attribute or to reset the original value.

Additionally special input windows, which are different from the display in the ADOxx Notebook, are displayed for editing the attributes of type enumeration (see chap. 2.2.12.3, p. 204), enumeration list (see chap. 2.2.12.4, p. 204) and program call (see chap. 2.2.12.5, p. 205).This input window will be displayed when you double-click on the corresponding cell.

Hint: The attribute values for these attribute types can be entered directly into the notebook in the graphical model display.

Undo

The value edited last in the tabular model display can be undone, i.e. the original value will be restored. Select the option "Undo" in the "Edit" menu, or press the key combination <Ctrl>+<Z> or click on the corresponding smart icon .

Sort

In the tabular model display it is possible to change the order of the objects or connectors of the class or relation displayed in the ADOxx browser (see chap. 8., p. 66). The objects can be ordered based on their attribute values. Select the function "Sort" in the "View" menu. The window "Sorting according to attribute columns" appears, in which you can select an attribute. The objects or connectors will then be sorted according to the attribute selected.

Hint: It is only possible to sort objects by an attribute, which is currently displayed in the ADOxx browser.)

Click on the Sort button or press enter in order to confirm the new sorting order.

Select Objects/Connectors

You can select each objects or connectors simply by clicking on the first (grey) column. Selected objects and connectors are highlighted in red (see fig. 160)

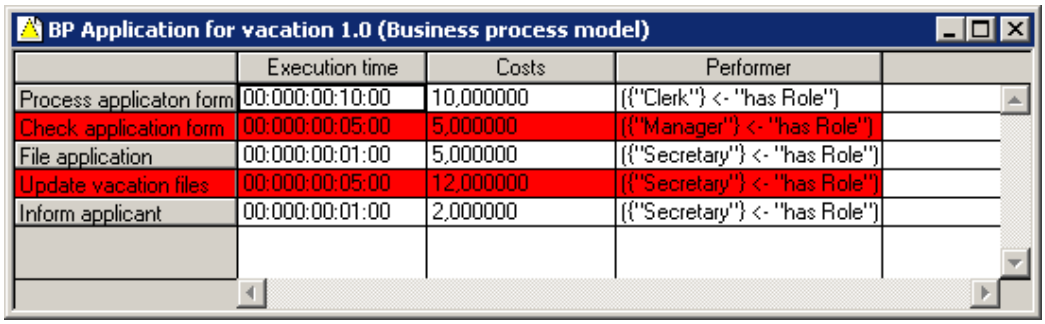


Figure 160: Selection of "object lines"

You can select several objects or connectors by clicking on them one after the other while keeping the <Shift> key pressed.

Once you have selected a number of objects you can still make changes to this selection by keeping the <Shift> key pressed and clicking on any objects or connectors you wish to select or deselect. If you wish to deselect the items you have selected, click on any free space in the ADOxx browser (see chap. 8., p. 66).

This selection of objects or connectors will be maintained when switching between graphical and tabular model display, if the objects selected in the graphical display belong to the class selected in the tabular display.

Create New Objects

To do so, at first the object class (see chap. 2.2.10.1, p. 182) has to be selected. At the end, click on the **"Position new object"** in the **table's context menu** (right mouse button) .

A new object, with a random name will be added at the end of the list. Attributes can be edited (see chap. 2.2.10.3, p. 184) as in already existing objects already existing objects.

> **Hint:** Connectors cannot be crated in that way.

A new object will be placed in the free spot in the drawing area. Later on it has to be correctly position in the graphical representation and connected with connectors.

2.2.11 View Options

View options influence the way the model is displayed - both on the screen and when it is printed. Since view options are part of the model level attributes, the current settings are saved in the ADOxx database.

The view options available are combined in the "View" menu. Some of them can be activated via the drawing area's context menu or by mouse clicking on the corresponding smart icon in the quick-access bar. The following options are available:

- **Graphic/Table** - for switching between graphic and tabular model display (see chap. 2.2.10, p. 181)
- **Mode** (see chap. 2.2.11.1, p. 186) - predefined view modes on objects and connectors
- **Views** (see p. 187) - various view options
- **Mouse access** (see chap. 2.2.11.2, p. 188) - for setting mouse access to objects and connectors
- **Zoom** (see chap. 2.2.11.3, p. 189) - scale for the display of the model on the screen
- **Snap grid** (see chap. 2.2.11.4, p. 190) - grid which can be defined in order to support modelling and arrangement of objects
- **Page layout** (see chap. 2.2.11.5, p. 192) - page size and print layout for ADOxx models
- **Representation of connectors** (see chap. 2.2.11.6, p. 197) - graphical tools for connectors
- **Font size** (see chap. 2.2.11.7, p. 199) - font size of visualised attributes
- **Select attributes** (see chap. 2.2.10.2, p. 183) - selection of the attribute columns to be displayed (only in the tabular modelling)
- **Sort** (see p. 185) - Description of an attribute, according to which the table will be sorted (only in the tabular modelling)
- **Redraw** (see chap. 2.2.11.9, p. 201) - the model window will be redrawn

2.2.11.1 View Mode

The (view) mode determines which classes and relations are visible. All non-visible classes and relations are removed from the modelling bar so that no new objects or connectors of these classes and relations can be created.

Objects and connectors belonging to the non-visible classes, which have already been created, will not be shown in the model window, i.e. the model editor behaves as if these objects and connectors were non-existent. This applies in particular to the printing and graphic generation functions.

The option "Mode" in the "View" menu contains a sub menu listing all the modes available. A tick marks the current mode. You can switch to another mode by selecting it from the sub menu - the modelling bar and the drawing area of the active model window will be rearranged accordingly.

Hint: The available modes depend on the definition of your application library.

Hint: A description of the modes of the **ADOxx-Default-Library** is to be found in the volume III of the user documentation ("ADOxx standard method manual") or in the online documentation "ADOxx standard method" ("Extras" menu, "ADOxx standard method "menu item).

Hint: The settings of the (view) mode only affect the way the model is displayed on the screen and when printing. For further processing (Analysis, Simulation etc.) all the objects and connectors modelled are of course taken into account.

Hint: Should you need further modes defined, please contact your ADOxx administrator or your ADOxx consultant.

Display Options

While modelling it can be helpful to display the hidden objects on the drawing area. Otherwise, it could easily happen that objects and connectors are created in the same place or overlapping each other.

For this reason the non-visible objects and connectors may be visualised on the screen in a light grey outline. In order to select the screen display options for the current mode, select the option "Display" from the menu "View", sub menu "Mode".

This will cause the window "Mode - display" to be opened. It is possible to define here when objects and connectors, which have been made non-visible by a certain mode, should nonetheless be shown (in light grey outline) in the model windows.

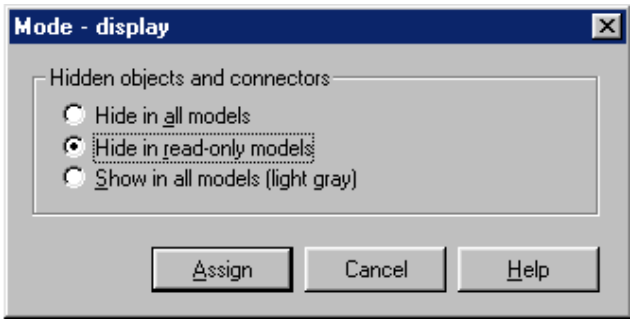


Figure 161: Display options

Choose one of the following options by clicking on it:

- **"Hide in all models"**: The items made non-visible by the current mode will not be displayed in any modelling window.
- **"Hide in read-only models"**: In models with write access the non-visible items are displayed with light grey outlines. If the model is being viewed "read only" (i.e. write protected models) the non-visible items will not be displayed at all.
- **"Show in all models (light grey)"**: Non-visible items are shown as light grey outlines in all modelling windows.

  Hint: The option selected does not influence printing or graphic generation. When printing and generating graphics, the system never takes non-visible objects into account.

Click on the "Assign" button to close the dialogue window and to confirm the display option selected. Clicking on the "Cancel" button will close the window without executing any changes.

The display option selected will be saved in your user profile in the database. This means that the setting will be maintained after you quit and re-start ADOxx.

2.2.11.2 Mouse Access

"Mouse access" describes the ability, to edit in the graphic modelling objects and connectors in a model by clicking in them (e.g. move, change attributes values). When you lock the mouse access, you can disable changes on locked objects and connectors.

Hint: The mouse access lock happens for all objects of a class, even if you activate the lock using the context menu of an object or of a connector.

You can define the mouse access either by using the context menus of objects (see chap. 2.2.8.2, p. 174) and connectors (see chap. 2.2.8.3, p. 176) or by choosing the menu item "Mouse access" from the "View" menu and selecting the option "List" from the sub menu which is displayed.

The window "Mouse access" (see fig. 162), in which all classes and relations are listed, appears. The classes or relations with locked mouse access are displayed as a light grey outline and with a "(-)" attached to the class or relation name.

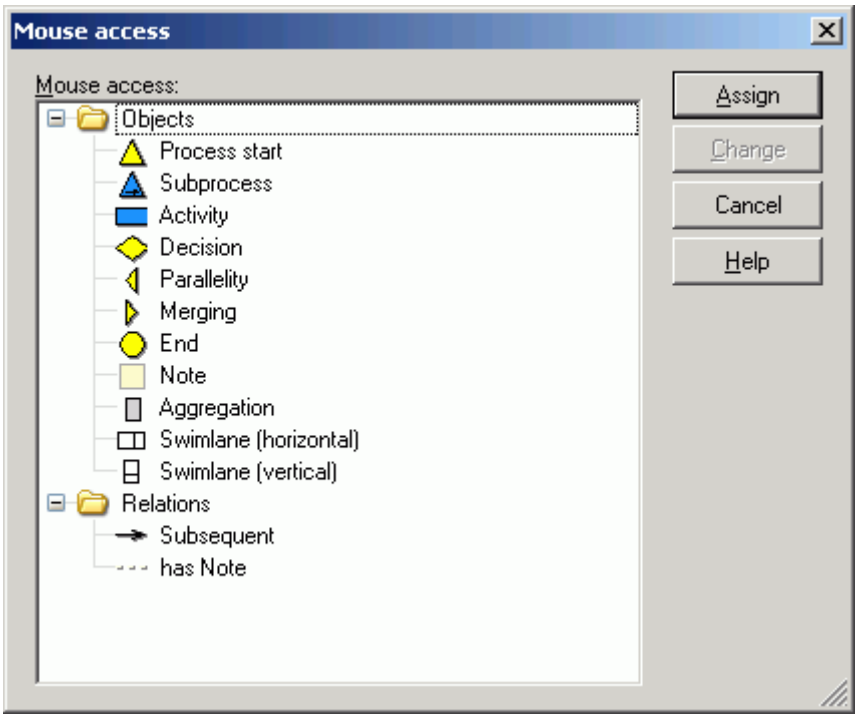


Figure 162: define mouse access

You can change the mouse access for objects of the selected classes or for the connectors of the selected relations by double-clicking on the appropriate element or by pressing the space bar. Additionally, it is possible to select a class or a relation and then to click on the "change" button, which will also change the status (locked/unlocked).

Click on the "Assign" button to confirm the changes you have made.

Select the menu item "Mouse access" from the "View" menu and choose the option "Representation" from the sub menu to define how locked objects and connecters are displayed (see p. 189).

Display Options

In the "Mouse access - representation" window (see fig. 163) you will define how locked objects and connectors will be displayed in the graphic modelling.

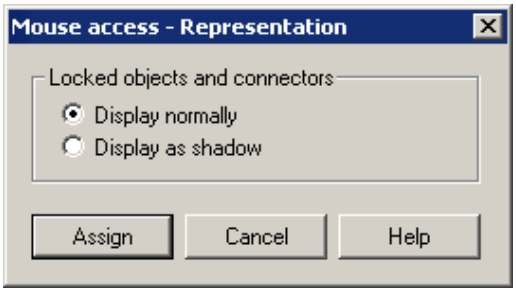


Figure 163: Representation of locked objects/connectors

The **"Display normally"** representation form causes no changes, but if you choose the option "**Display as shadow**" all objects of a locked class will be displayed as outlined.

The selected representation of locked objects will be saved in the database of you user profile. It will remain the same, if you exit ADOxx and start again.

2.2.11.3 Zoom

The "Zoom" function changes the scale of display in the modelling window. Values between 5% and 200% are possible. The value 100% displays the model on a 1:1 scale. Smaller zoom values help you get an overview of bigger models.

The current zoom factor is indicated in the zoom field of the status line. The zoom field is the left of the two fields placed on the right side of the status line.

Change the zoom value by following one of these steps:

- Right-click on the zoom field of the status line. A popup-menu (see fig. 164) listing possible zoom values for selection appears. A tick marks the current value. The value, which allows the whole model to be displayed in the modelling window, is marked by an asterisk.

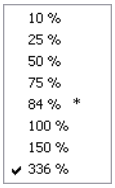


Figure 164: Zoom popup menu

- **Select an area** by placing the cursor in a corner of the area, pressing the **right** mouse button and dragging the mouse - keeping the right mouse button pressed down (the mouse pointer will change into a) - until the green outlined area has the size you want. If you let go of the mouse button, the zoom value will change so that this **area fills the drawing area**.
- If the **zoom value does not equal 100%**, the function **"Scale 1:1"** enables you to set it to 100%. This function can be called either:
  - By clicking on the respective smart icon in the quick-access bar.
  - By selecting the "scale 1:1" menu item in the context menu of the drawing area.
  - By pressing the right mouse button and keeping it pressed and then press the left mouse button.

If the zoom value already equals 100%, the smart icon will be displayed as a pressed button.

- If you want to display the whole drawing area (see chap. 2.2.3, p. 130) in the model window, select the function "Display all". To call this function you can either:
  - Click on the smart icon in the quick-access bar.
  - Select the "Display all" menu item in the context menu of the drawing area.
  - Press the right mouse button and keeping it pressed and then double-click on the left mouse button (Note: When the zoom value equals 100%, single click on the left mouse button).
- You can increase or decrease the zoom value by 10% by pressing the keys "Page up" and "Page down" respectively.
- You may also set the zoom value using the function "Zoom" (see p. 190). Call this function by:
  - Clicking on the appropriate smart icon in the quick-access bar.
  - By selecting the menu item "Zoom" from the context menu of the drawing area.
  - By selecting the menu item "Zoom" from the "View".
  - By double-clicking (left mouse button) on the zoom field in the status line in the right lower corner of the ADOxx window.

Hint: The changes can be undone by clicking on the "undo" button in the quick-access bar.

Hint: The changes can be cancelled using the "undo" function (see chap. 2.2.7.1, p. 137).

Define Zoom Value

You can change the zoom factor in the zoom window "Zoom" (see fig. 165).

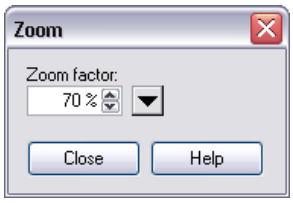


Figure 165: Zoom

You can enter the zoom value into the field **"Zoom factor"**. You can open a popup menu (the same that appears when right-clicking on the zoom field in the status line) by clicking on the menu button . If you select a value from this list, it will be assigned to the field "Zoom factor".

Click on the button "Close", to close the window and to display the visible area with the zoom setting in the model window.

2.2.11.4 Snap Grid

The snap grid helps you when arranging the available objects and connectors and can be.

- **Active** or **inactive** or.
- **Visible** or **non-visible**.

If you have selected the function **"active snap grid"**, objects and connectors when modelling will be bound to the snap grid, i.e. the objects and the bendpoints of the connectors will be placed at the intersection of a horizontal and a vertical line of this grid.

If you have selected the option **"visible snap grid"**, the snap grid will be displayed on the graphic model view as a grey grid. The snap grid now supports you when arranging a model.

Hint: The lines of the snap grid are visible on the screen but not on a print out of the model or on a generation of the model graphic.

You can activate or deactivate the snap grid by:

- Clicking on the smart icon in the quick-access bar.
- Selecting the submenu "active" ("View" menu, "Snap grid" menu item).
- Defining the snap grad settings (see p. 191).

Hint: If the snap grid is active, the smart icon in the quick-access bar appears as pressed.

You can make the snap grid visible or non-visible by:

- Clicking on the smart icon in the quick-access bar.
- Selecting the "visible" submenu ("View" menu, "Snap grid" menu item) .
- Selecting the "Display snap grid" menu item in the context menu of the drawing area.
- Editing the snap grid settings (see p. 191).

Hint: If the snap grid is visible, the smart icon in the quick-access bar appears as pressed.

Additionally you can change the spacing between the lines of the snap grid in the snap grid settings (see p. 191).

Settings

In the "Snap grid - settings" window (see fig. 166), you can activate or deactivate the snap grid (option "Active"), make it visible or invisible (option "Visible") or change the horizontal and vertical settings of the grid.

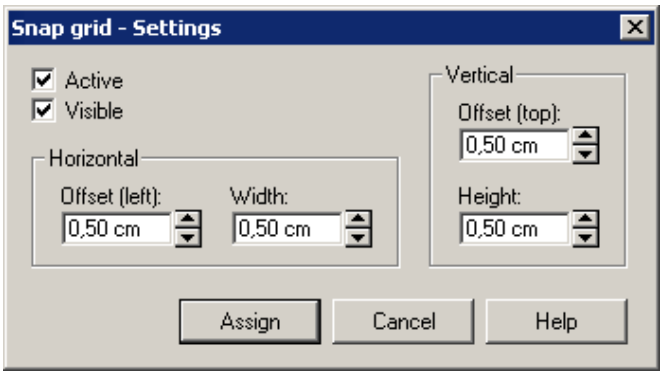


Figure 166: Snap grid -Settings

Over the **"Active" option** you may activate or deactivate the snap grid. Over the **"visible" option** you may make the snap grid visible or non-visible .

Defining the width in horizontal direction or the height in vertical direction can change the **spacing of the snap grid lines** (see fig. 167).

If you enter an offset, the representation and the arrangement of the snap grid will be changed to the right or bottom by the defined spacing starting from the top left corner of the drawing area (see fig. 167).

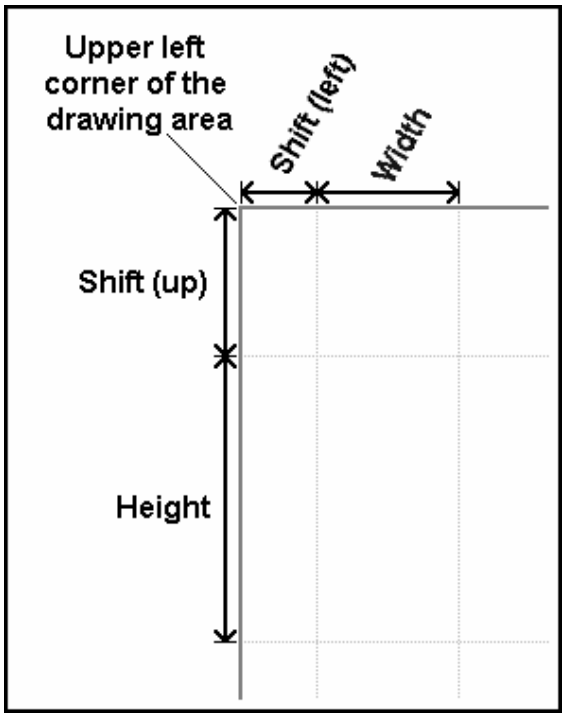


Figure 167: Example of the snap grid settings

Hint: The value of the horizontal offset will only be taken into account if it is smaller than the width, and the value of the vertical offset only if it is smaller than the height.

When you have changed the settings, click on the "Assign" button to save the settings and close the window. The snap grid settings are saved in the ADOxx database separately for each model.

Hint: The standard settings of the snap grid (activity, visibility, height, width and offset) are defined in the application library.

2.2.11.5 Page Layout

Page layouts can be assigned to each ADOxx model. Apart from the sheet size of a printout, the page layout also determines its layout and structure. In each page layout you can define user-specific headers and footers, integrate graphics such as company logos and display model information.

Select the option **"page layout"** in the **"view"** menu. The window "page layout" appears (see fig. 168).

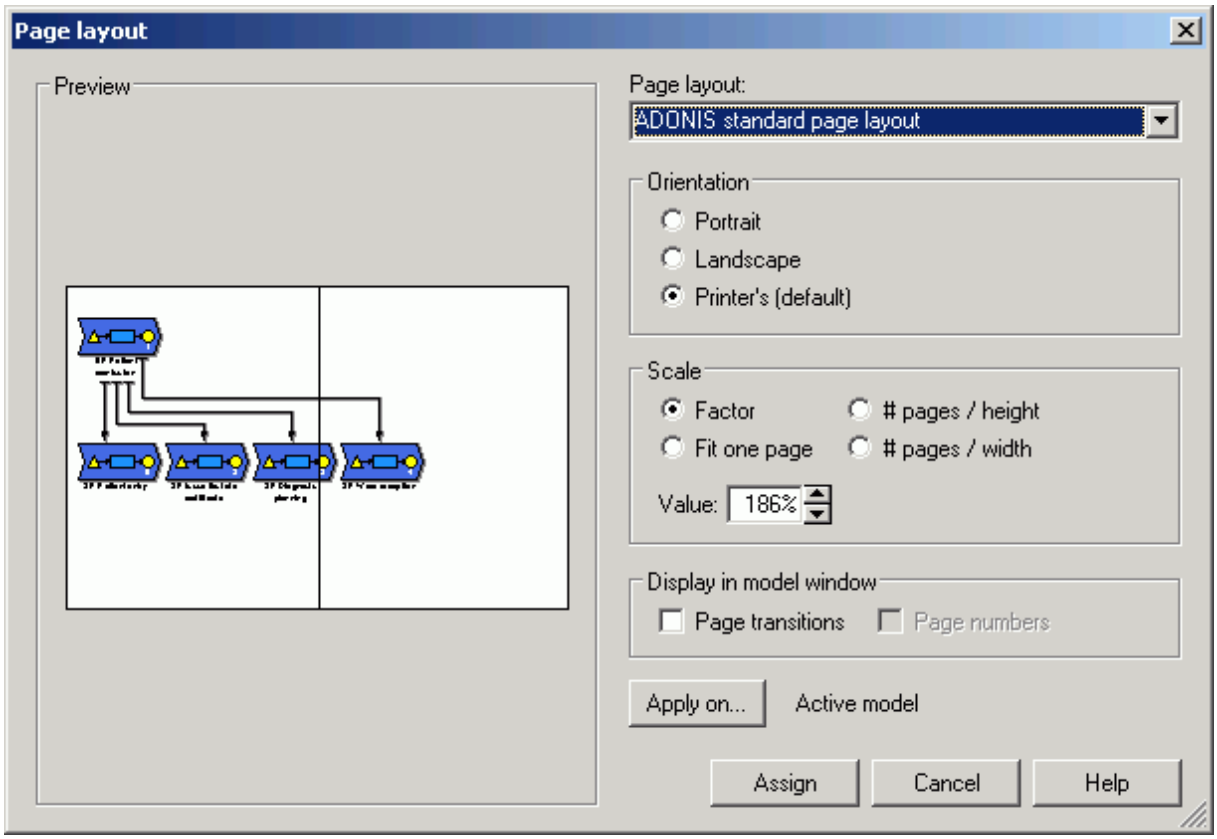


Figure 168: Page layout

Choose your settings for **"page layout"**, **"layout parameters"** and **"scale"**. Details about those settings can be found in the chapter Layout parameters (see p. 194).

In the window **"display model window"**, following options can be activated:

- **Change pages:** The pages area will be displayed in the Model window (see chap. 2.2.3, p. 130)
- **Page number:** Next to the page edge the corresponding page number will be displayed

By clicking on the button "Apply to", a small dialogue with opened models will appear. Here you can select the model, for which, you wish to apply your settings:

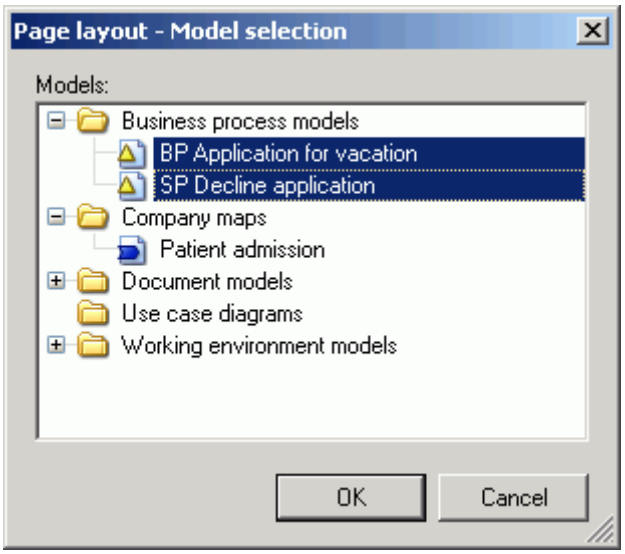


Figure 169: Page layout

Hint: By default, the layout changes will apply only for the active model

Click on the **"Apply"** button, to apply the settings. The dialogue window will be closed.

Layout Parameters

The layout parameters arrange the format of models. It can be defined in a different places or modified and saved in the models.

Define layout parameters:

Select the desirable **"page layout"** from the list .

Hint: The different page layouts are defined in the application library.

A few default page layouts can be found in all application libraries: The **ADOxx standard page layout** (see p. 195) and the poster layout **Poster (with flaps)** (see p. 196) for graphical model representations and the **ADOxx standard page layout for tables** (see p. 196) only for printing from the ADOxx Browser.

In the field **"layout parameters"** you can select the following orientations: **portrait**, **Landscape format** or **from the printer** (according to the printing settings).

The representation size is set in the **"Scale"** section:

- **Factor:** The percentage of the original size (by default 100%).
- **On one page:** The whole model will be zoomed to fit on one page regardless of the actual size.
- **number of pages - height:** Determine, the optimal model representation in terms of height (in pages). The width is calculated automatically.
- **Number of pages - width:** Determine, the optimal model representation in terms of width (in pages). The height is calculated automatically.

The **preview** window will be updated with every change.

The changes in the pages layout affect as well the models representation (the pages boundary will change), as the model printout and the graphics generation. Additionally the existing connector marks (see chap. 2.2.11.6, p. 197) might be moved to a new place.

ADOxx Standard Page Layout

The ADOxx standard page layout is defined in ADOxx and is always available to printout models.

The **ADOxx standard page layout** (see fig. 170) contains:

- A headline with:
  - the ADOxx logo
  - The model name
  - The date on which the model was created
  - The name of the ADOxx user, who created the model
  - The value of the model attribute "model type"
  - The date on which the model was last changed
  - The name of the ADOxx user who last changed the model
  - The value of the model attribute "State"
- An area for the model graphic
- A foot line with:
  - The copyright
  - The printout date
  - The page number

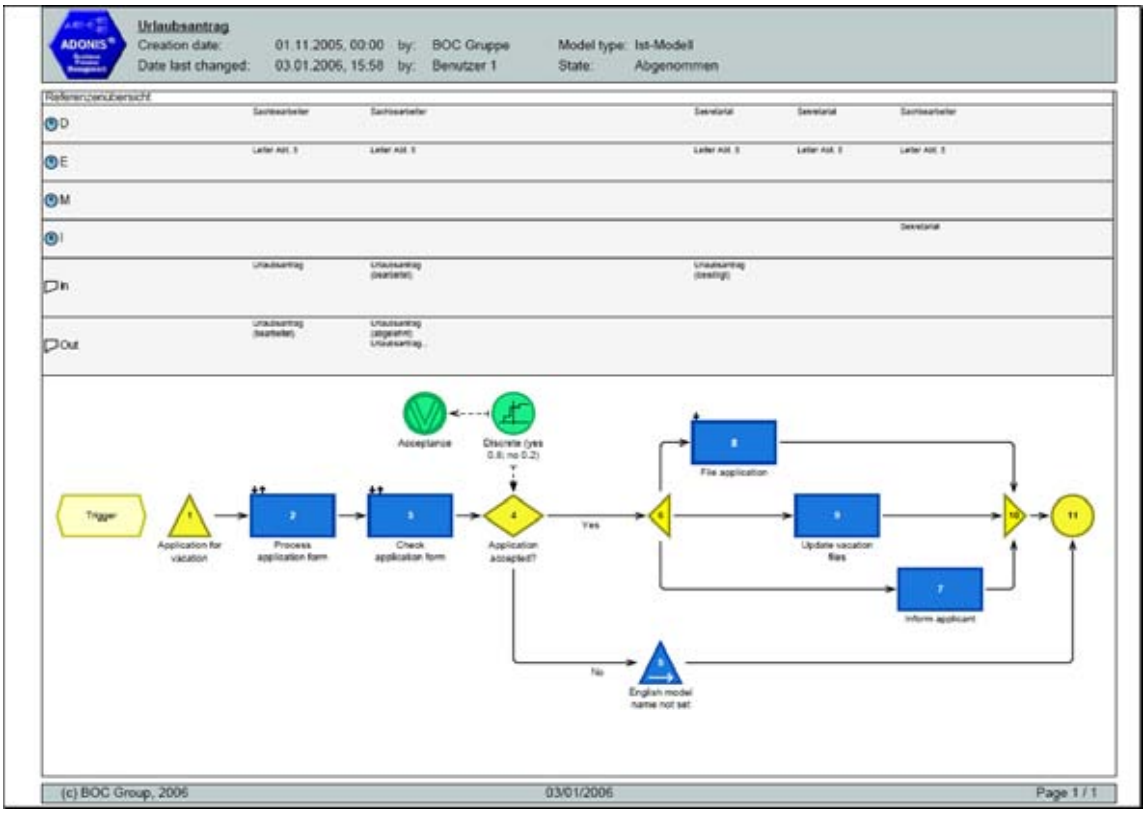


Figure 170: ADOxx standard page layout (example)

Poster Printout

The **Poster (with flaps)** page layout is defined in ADOxx and is thus always available. It is designed for the printout of larger models where the single pages are sticked together.

A special feature of this layout is that on the right and bottom page border flaps are printed, giving room for glue.

ADOxx Standard Page Layout for Tables

The ADOxx standard page layout for tables is defined in ADOxx and is always available for the printout from the ADOxx-Browser (see chap. 8., p. 66) after the realisation of Evaluation (e.g. Analysis, Simulation).

The **ADOxx standard page layout for tables** (see fig. 171) contains:

- A headline with:
  - dem ADOxx-Logo
  - The date of the Evaluation (date, time, type of Evaluation etc.)
- The area for the tables
- A footer with:
  - The copyright
  - The printout date
  - The page number

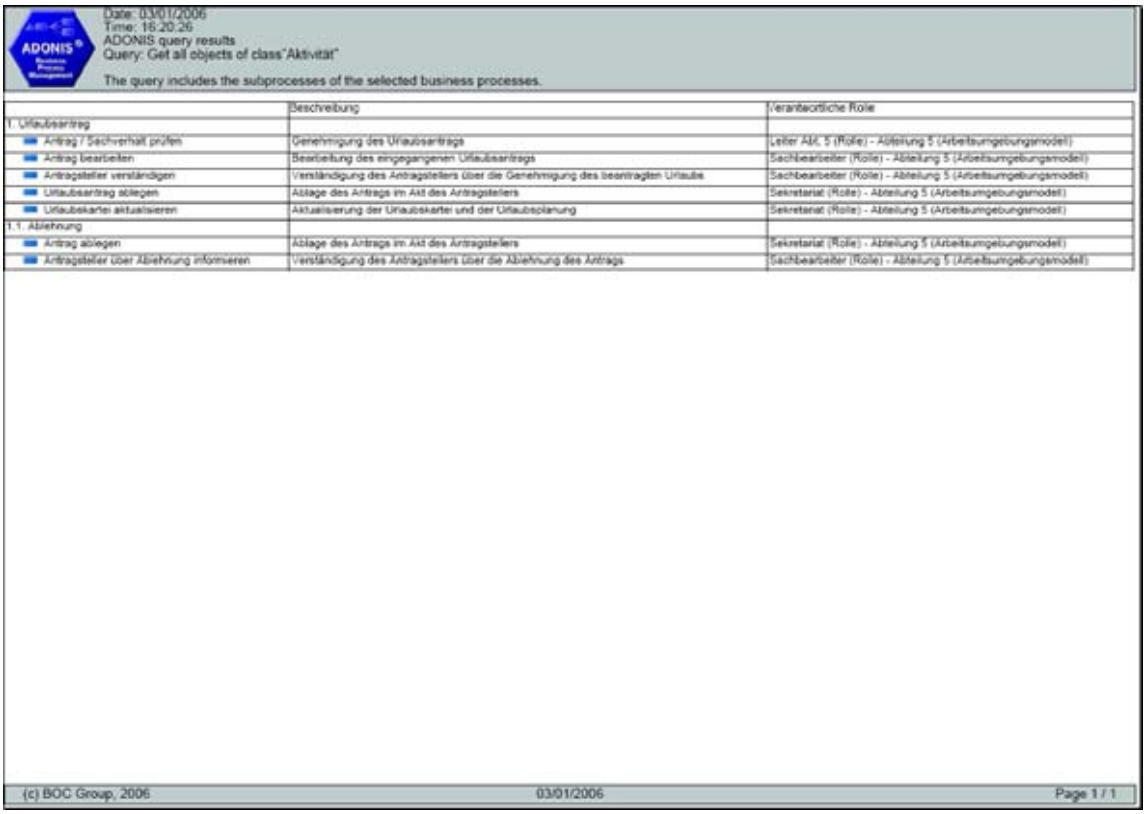


Figure 171: ADOxx standard page layout for tables (examples)

2.2.11.6 Connector Representation

The basic look of connectors is determined during the application library definition. However, it is also possible for a user to make personal adjustments that will be stored in the particular models:

Connector marks: Useful display variant for page-spanning connectors - the ends on every page are marked to make locating the pairs easier.

Bendpoints: Depending on the user's preferences, bendpoints can be shown either sharp or rounded.

Crossings: If connectors cross over, the junctions can be represented either with little bridges or just flat.

To call these functions, select **"Change connector representation"** from the **"View"** menu. The following dialogue window appears:

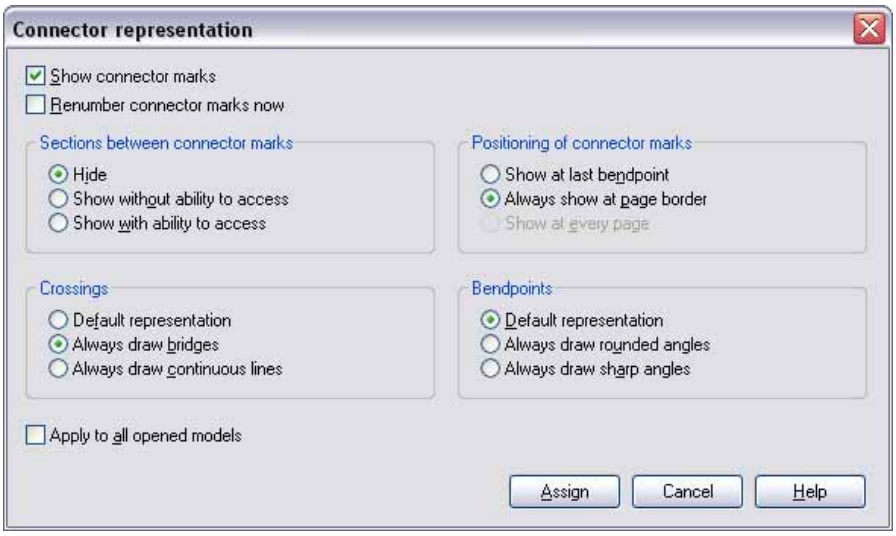


Figure 172: Connector representation options

Choose your settings and click **"Assign"**. The settings are applied and the dialogue window closed.

Connector marks:

ADOxx provides so-called **connector marks** in order to make printouts of large multi-page models more readable. These are graphical aids indicating page-spanning connectors.

Activate the option **"Show connector marks"** (☑), to assign connector marks to all connectors extending over a page border. This checkbox activates all other connector marks options too.

Connector marks are numbered upon creation to make them unique. If you add page-crossing connectors while modelling, the corresponding connector mark is duly numbered with the next available number. When deleting connector marks, their number is maintained internally and not reassigned. To regian consecutive numbering, select the option **"Renumber connector marks now"**.

In addition you have to choose one of the options for displaying the **sections between connector marks**:

- **Hide:** The part of the connector between the two connector marks is invisible (default).
- **Show without ability to access:** The part of the connector between the two connector marks is shown as a thin grey line. The course of this line cannot be changed.

- **Show with ability to access:** The part of the connector between the two connector marks is shown in black. The connector's course can be modified with bend points (see chap. 2.2.6.1, p. 133).

These variants are possible for the **Positioning of connector marks**:

- **Show at last bendpoint:** The connector mark appears at the last bend in the connector before it leaves the page (default).
- **Always show at page border:** Regardless of the connector course, the connector mark always appears at the page border.
- **Show at every page:** The connectors are shown on all pages they touch, even if they just cross them (and have no connection to an object there).

Hint: Size, colour, numbering and shape of the connector marks are defined in the application library. If you want tailor-made connector marks, please contact your ADOxx administrator or your ADOxx consultant.

Bendpoints:

For bendpoints (see chap. 2.2.6.1, p. 133), three possible looks are available:

Default representation:
The bendpoints will be displayed as in the ADOxx administrator settings.

Always draw rounded angles:
All bendpoints will be rounded, regardless of the administrator settings.

Always draw sharp angles:
All bendpoints will be sharp.

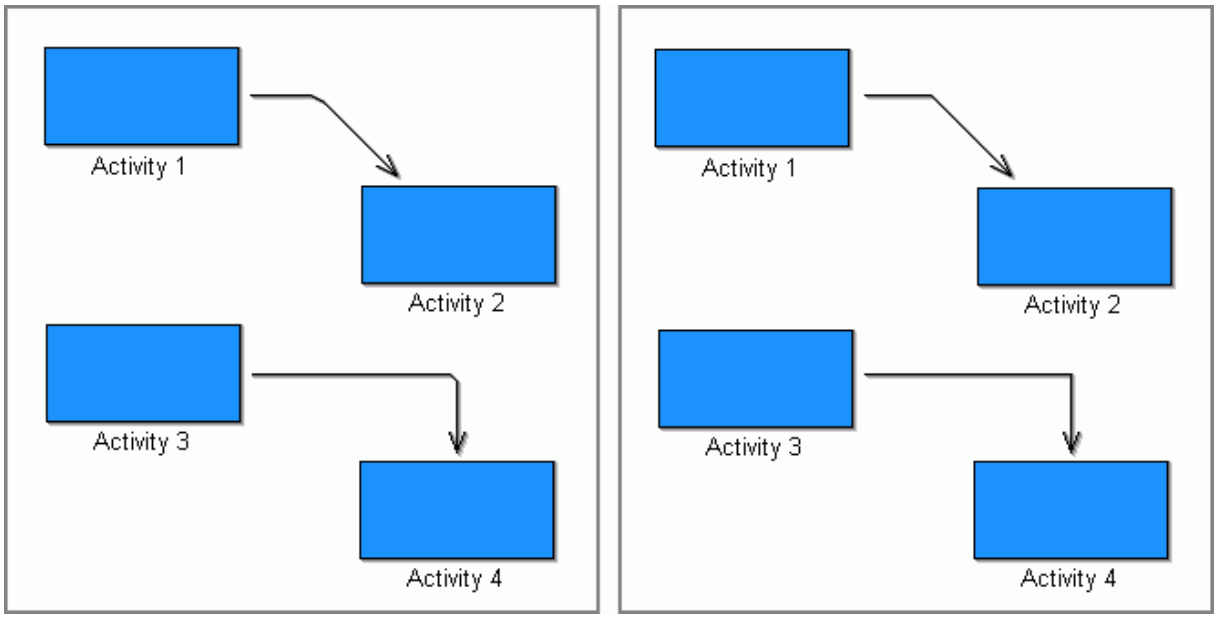


Figure 173: Representation with round/sharp bend points

Crossings:

There are also three variants to display connector crossings:

Default representation:
Junctions will be displayed as in the ADOxx administrator settings.

Always draw bridges:
The connector lying on top is drawn with a little bridge at the junction, regardless of the administrator settings.

Always draw continous lines:
Junctions are represented only by straight lines.

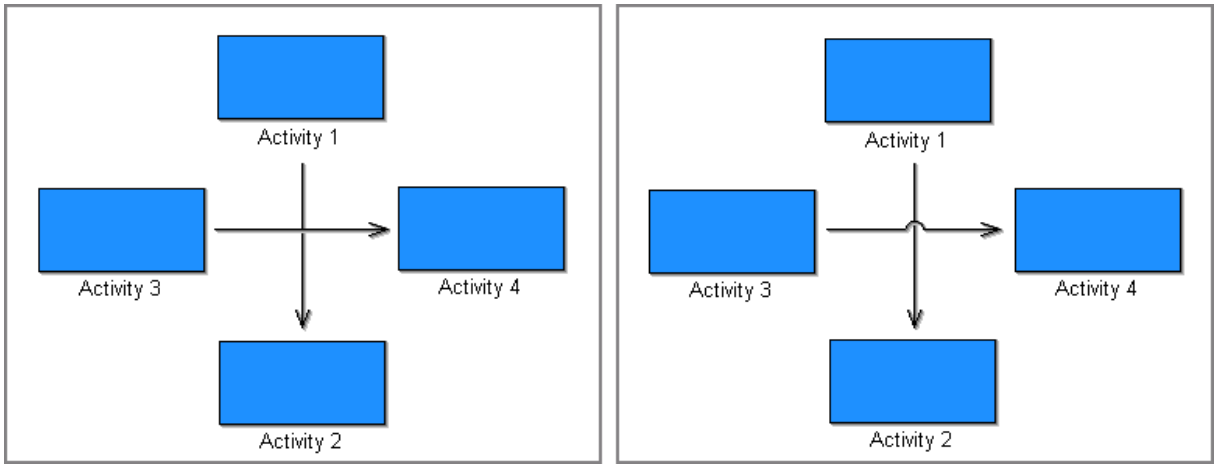


Figure 174: Representation of crossings - with or without bridges

Apply to all opened models:

If this option is activated, these settings will not only be saved in the active model but in *all* opened models.

Hint: ADOxx administrators can define the appearance of bendpoints and junctions for each type of connector *separately*. However, the manual settings here concern *all* connectors in the model.

2.2.11.7 Font Size

Change the font size of visualised attributes by selecting the option "Font size" from the "View" menu. The window "Font size" appears (see fig. 175).

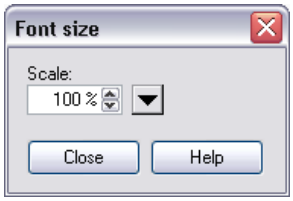


Figure 175: font size

Set the font size at a value between 50% and 200% of the original size by:

- directly entering a new value in the **"Scale" field**,
- editing the value in the "scale" field using the spin buttons (step size: 1%),
- clicking the button and selecting a predefined value.

Click on the "assign" button, to assign the changed value of the font size for the visualised attribute in the model.

Hint: As visualised attributes on different objects may be displayed with different font sizes, it is only possible to change the font size of a model relatively.

2.2.11.8 Show/Hide Submodels

With the function "show submodels" it is possible to expand the view of the active model by replacing submodel referencing objects (e.g. "Subprocess") with a view on the model graphic of the referenced model itself.

Showing submodels is performed:

- for **all referencing objects** in the active model with the function "Show submodels" (see p. 200),
- for **a single referencing object** in the active model by opening the context menu of the referencing object (right mouse button) and selecting the menu entry "Show submodel".

Hiding submodels is performed:

- for **all shown submodels** in the active model with the function "Hide all submodels" (see p. 200),
- ffor **a single referencing object** in the active model by opening the context menu of the particular submodel view (right mouse button) and selecting the menu entry "Hide submodel" or by clicking the "Close" icon ( ) in the upper right corner of the submodel view.

Show Submodels

For showing all submodels of the current model select the menu **"View"** - entry **"Show/Hide submodels"** - **"Show submodels"**.

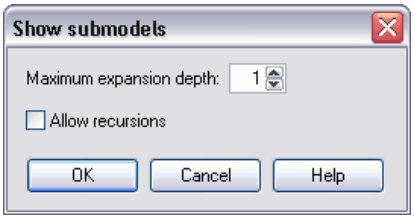


Figure 176: Show submodels

In the window "Show submodels" (see fig. 176) you can adjust these settings:

Maximum expansion depth:
Define through how many depth levels the submodels shall be shown.

Allow recursions:
Activate this option, if you want to include recursions: Already shown models (including the parent model itself) will be shown again, if they are called more than once and the maximum expansion depth allows it.

Hide All Submodels

Zum Ausblenden aller Submodelle im aktuellen Modell wählen Sie im Menü "Ansicht" den Menüpunkt "Submodelle ein-/ausblenden" und den Untermenüpunkt "Alle Submodelle ausblenden".

2.2.11.9 Redraw

Selecting the option **"Redraw"** from the "View" menu makes ADOxx redraw the current model window.

2.2.12 Input Support Dialogues in ADOxx Notebooks

ADOxx provides input support dialogues for entering complex attribute values and formulae into the ADOxx Notebooks and in the tabular modelling (see chap. 2.2.10, p. 181). Whether such a supporting dialogue is available for a given attribute is indicated by the "Dialog" button above the attribute's input field.

> **Hint:** The availability of the input support dialogs depends on the definition in your application library .

These input support dialogues are available:

Edit table attributes (see chap. 2.2.12.1, p. 202)
for the editing of attributes of the type "record" .

Colour definition (see chap. 2.2.12.2, p. 203)
for the selection of a colour (for the graphic representation of the object).

Select enumeration value (see chap. 2.2.12.3, p. 204)
for the selection of the attribute value (in the table modelling).

Select enumeration values from a list (see chap. 2.2.12.4, p. 204)
for the selection of attribute values from an enumeration list.

Enter external program call (see chap. 2.2.12.5, p. 205)
for the input of an external program call (in the table modelling).

Edit date attributes (see chap. 2.2.12.6, p. 205)
for the editing of attribute of the type "date" (DATE).

Edit date/time attributes (see chap. 2.2.12.6, p. 205)
for the editing of attributes of the type "date and time" (DATETIME).

Edit time attributes * (see chap. 2.2.12.8, p. 206)
for the editing of attributes of the type "time" (TIME).

Add references * (see chap. 2.2.12.9, p. 207)
for the creation of model and object references in attribute of the type "reference" (INTERREF).

Distribution definition * (see chap. 2.2.12.11, p. 211)
for the definiton of a distribution function for the distribution of variables.

Transition condition definition * (see p. 215)
for the definition of a transition condition in a subsequent relation.

Performer assignment * (see chap. 2.2.12.14, p. 222)
for the definition of a performer assignment expression.

Resource assignment * (see chap. 2.2.12.15, p. 227)
for the definition of a resource assignment expression.

Definition of the performer calendar * (see chap. 2.2.12.16, p. 228)
for the definition of the performers time of presence.

Definition of the process calendar * (see chap. 2.2.12.17, p. 235)
for the definition of the appearance profiles of processes.

Hint: The attributes "Performer calendar" and "Process calendar" are edited through a button (called "Performer calendar" or "Process calendar") in the ADOxx Notebook instead of the "Dialog" button and a separate input field.

Hint: For reference attributes, open the supporting dialogue over the "add" icon .

Hint: All input supporting dialogs marked with a "*" are used to get models that are simulable. You will find a precise description in the Chapter "modelling guide-lines for Simulation" (see chap. 2.2.13, p. 240).

A special input supporting dialogue (see chap. 3.2.12.3, p. 52) is available for the definition of attribute values in attributes of the type "expression" (EXPRESSION). This input supporting dialogue will be started using the "expression dialogue" icon .

2.2.12.1 Edit Record Attributes

ADOxx provides you with a support dialogue for editing record attributes, which is to be opened in the ADOxx Notebook by clicking on the "dialogue" button , on the right above the record display.

Once you have clicked on the button, the record will appear in the window "<*Object name*> - <*Name of the record attribute*>" (see fig. 177), in which you can edit attribute values, as well as add and delete rows.

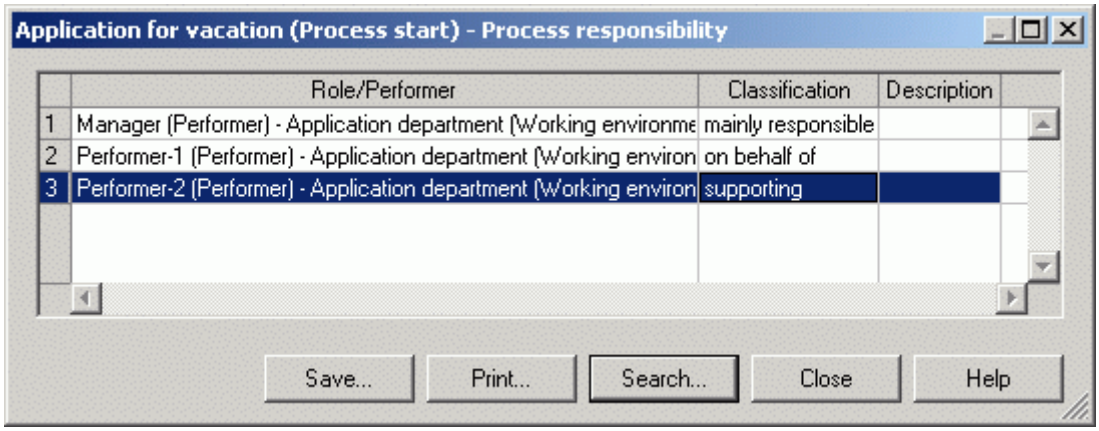


Figure 177: Edit record attributes

To **edit an attribute value**, first select the appropriate cell and then the menu item "Edit attribute" in the context menu (right mouse button). The input is done directly into the cell or an input window depending on the type of the selected attribute.

To **copy the value of an attribute**:

1. Select the cell with the attribute value to be copied
2. Select in the context menu (right mouse button) the menu item "Copy"
3. Select the cell into which you want to paste the attribute value and
4. Select in the context menu (right mouse button) the menu item "Paste".

Hint: Copying values within a record is only possible for attributes of the same type or in attributes of type "String".

To **copy the value of a row**:

1. Mark the row to copy by clicking on the row number (grey column)

2. Select in the context menu (right mouse button) the menu item "Copy rows"
3. Select the row into which the copied one has to be pasted and
4. Select in the context menu (right mouse button) the menu item "Paste".

Hint: When copying rows, the values of the copied rows will be inserted into an already existing row and so will overwrite the values contained in this row.

To **add a row to a record**, select in the context menu the menu item "Insert row". This new row will be inserted at the bottom of the record.

To **move a row in the record**, mark the row to be moved by clicking on the row number (grey column), then select in the context menu the menu item "Move row" - the mouse pointer will change to - and click on the row above where the new row has to be inserted.

To **delete a row from the record**, mark the row to delete by clicking on the row number (grey column) and then select in the context menu the menu item "Delete rows".

2.2.12.2 Define Colours

ADOxx provides you with an input window for the selection of a colour (e.g. for the graphical display of the object), which you can open in the ADOxx Notebook by clicking on the "dialogue" button , on the right above the corresponding attribute field.

Once you have clicked on the button, the window "colour" (see fig. 178) will appear.

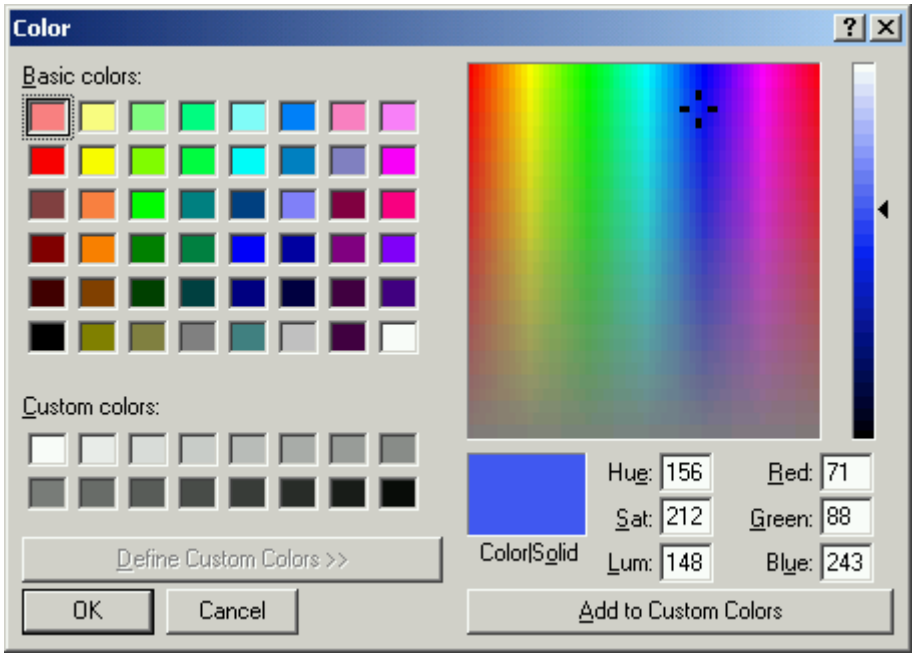


Figure 178: Edit colours

Select one of the **basic colours** or one of the **user-defined colours**. Alternatively you can edit a colour by entering the colour value (colour/saturation/brightness or red/green/blue) or by directly clicking on the colour spectrum. If you click on the **"Add colour"** button, the edited colour will be inserted into the "user-defined colours" and is available for the current ADOxx session.

Click on the **OK button**, to copy the selected colour.

2.2.12.3 Select Enumeration Value

When double-clicking on an attribute of the type "Enumeration" (see chap. 7.2, p. 733) the window "<*object name - attribute name*>" (see fig. 179) containing the available attribute values will be displayed.

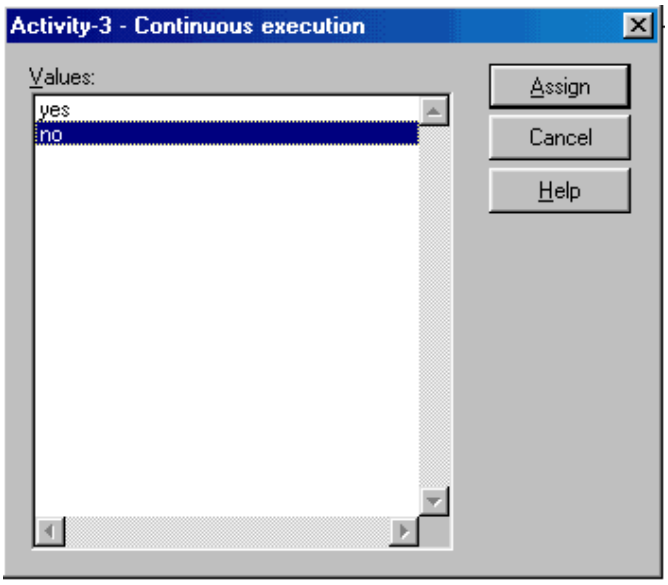


Figure 179: Enumeration

Choose an attribute value from the list **"Value"** and then click on the **"Assign"** button to take over the change.

Hint: The window for the input of an enumeration value (see fig. 179) is only shown in the tabular display.

2.2.12.4 Select Values from an Enumeration List

When double-clicking on an attribute of the type "Enumeration list" (see chap. 7.3, p. 734) the window "<*Object name - Attribute name*>" (see fig. 180) containing the available attribute values will be displayed.

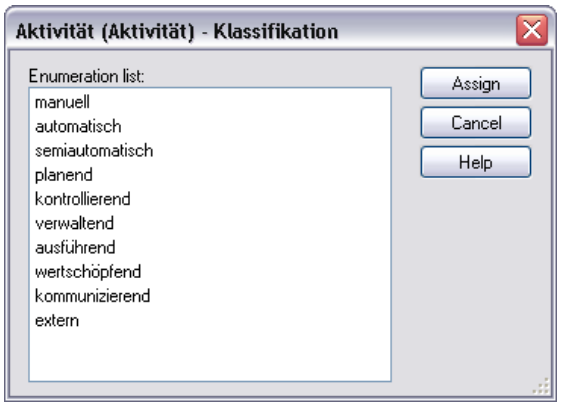


Figure 180: Enumeration list

Select the attribute value required from the field **"Enumeration list"** or mark it and click on the **"Assign"** button to assign the changes to the tabular model display.

Hint: To select several attribute values hold the <Ctrl> key additionally.

2.2.12.5 Enter Program Call

When double-clicking on an attribute of the type "Program call" (see chap. 7.11, p. 736) the window ""<*Object name - Attribute name*>"" (see fig. 181) showing the current settings will appear.

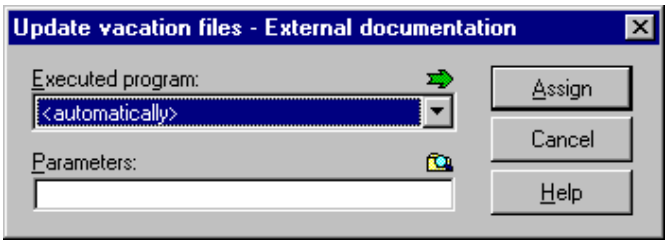


Figure 181: Program call

Select a program in the field **"Executed program"** and optionally enter the path and name of the file that should be opened on calling the program into the field **"Parameters"**. Then click on the **"Assign"** button to confirm the changes.

Hint: The representation of the program call attribute (as in the figure above) may depend on the definition of your application library.

Instead of the selection list "Executable program" a **button** with the name of the program or the text "<automatically>" can be shown.

It is possible to fade out the **field "Parameter"**.

2.2.12.6 Edit Date Value

ADOxx provides you with a support dialogue for the assignment of date attributes, which you can open from the ADOxx Notebook by clicking on the "dialogue" button , on the right above the attribute field.

Once you have clicked on the button, the window"<*Object name*> - <*Name of the date attribute*>" (see fig. 182)opens, where attribute value can be edited.

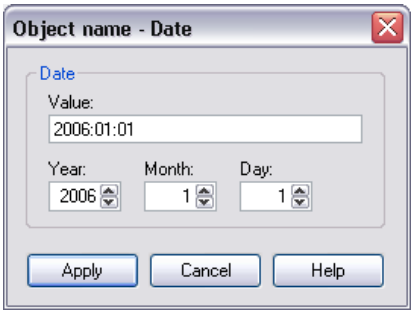


Figure 182: Enter date

The numbers given in the fields "year", "month" and "day" will automatically be converted into the date format (YYYY:MM:DD) and displayed in the field "value". By clicking on the button "Assign", the current value will be transferred to the ADOxx Notebook.

Hint: The attribute type "Time" (see chap. 2.2.12.8, p. 206) is available for the coverage of a time period.

2.2.12.7 Edit Date and Time Value

ADOxx provides you with a support dialogue for the assignment of date and time attributes, which you can open from the ADOxx Notebook by clicking on the "dialogue" button at the right above the attribute field.

Once you have clicked on the button, the window "<*object name*> - <*name of the date and time attribute*>" (see fig. 183) will appear, in which you can edit the attribute value.

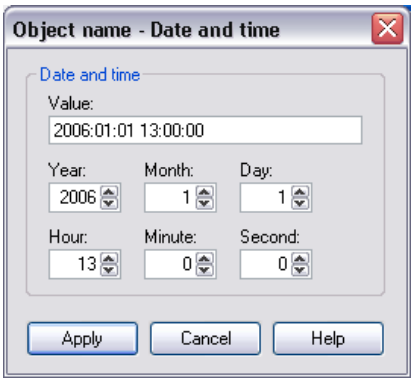


Figure 183: Enter date and time

The numbers given in the fields "year", "month" and "day" will automatically be taken over in the date format (YYYY:MM:DD) and displayed in the field "value". By clicking on the button "Assign", the current value will be transferred to the ADOxx Notebook.

Hint: The attribute type "Time" (see chap. 2.2.12.8, p. 206) is available for the coverage of a time period.

2.2.12.8 Editing Times

ADOxx provides you with a support dialogue for entering time attributes (time period), which you can open from the ADOxx Notebook by clicking on the "dialogue" icon at the right above the attribute field.

The window "<*object name*> - <*name of the time attribute*>" (see fig. 184), in which you can edit the attribute value, appears.

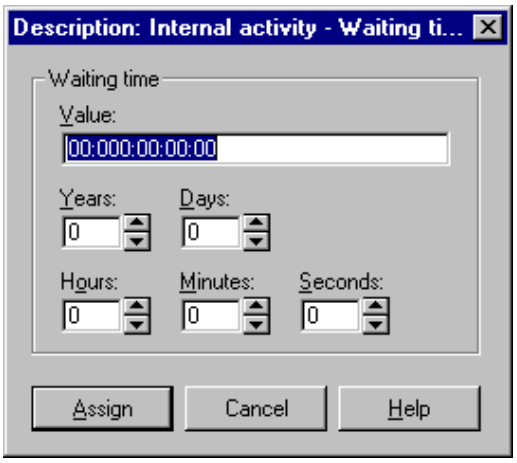


Figure 184: Enter a time period (example "Execution time")

The values entered into the fields "Years", "Days", "Hours", "Minutes" and "Seconds" are automatically transformed into the ADOxx time format (YY:DDD:HH:MM:SS) and displayed in the "Value" field. Confirm the current value by clicking on the "Assign" button.

Hint: The provision of **times** (see chap. 4.1.3.2, p. 335) in the models is **not a precondition** for the execution of the evaluation.

However, if this data is not provided the results may be incomplete (e.g. cycle time may be zero).

Hint: The attribute types "Date" (see chap. 2.2.12.6, p. 205) and "Date and time" (see chap. 2.2.12.7, p. 206) are available for the coverage of a time period.

2.2.12.9 Add References

You can add references:

- In the **graphical modelling**, by clicking on the "Add" icon of the corresponding reference attribute,
- In the **tabular modelling**, by double-clicking on the corresponding cell (in the row of the starting object and in the column of the reference attribute).

All models which can be referenced and are stored in the ADOxx database are available when adding a reference to a model (see p. 207).

It is possible to add a reference to an object (see p. 208) only if the model containing this object is opened.

Add Model References

In the window "<*Object name (class name) - Attribute name*> - Add reference" (see fig. 185), the model group hierarchy is displayed with all models you can reference.

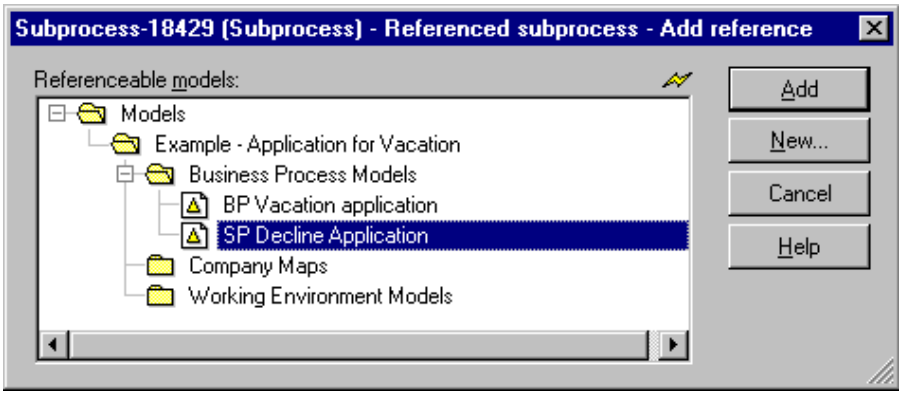


Figure 185: Add model reference

Select the model to refer to reference and click on the **"Add"** button. The window will be closed and the name of the referenced model will be registered in the text field of the ADOxx Notebook.

Click on the button **"New"** to create a new model (see chap. 2.4.1, p. 256). Once you have created the new model, the "models that can be referenced" list will be updated and you can put a reference to this model.

Hint: The modelling of the new created model is carried out later.

By clicking on the icon , you can update the "models that can be referenced" list.

Add Object References

In the window "<*object name (class name) - attribute name*> - Add reference" (see fig. 186), all opened models that can be referenced - sorted according to their model type - and the current reference targets will be displayed.

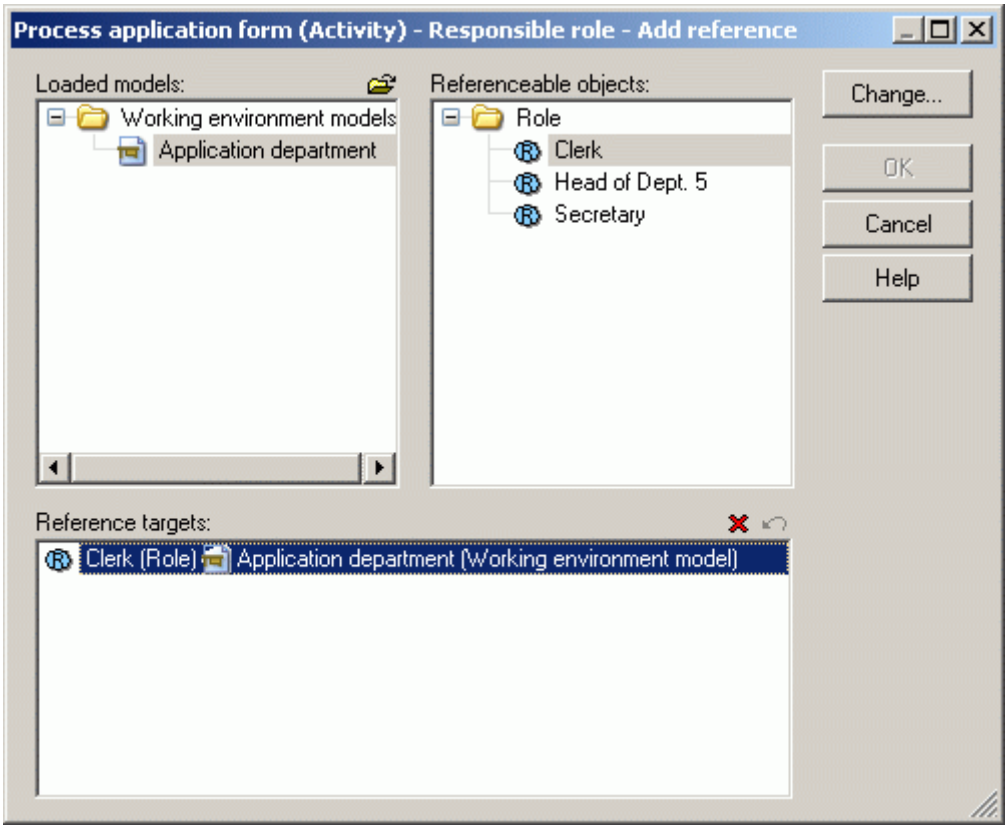


Figure 186: Add object reference

First select in the field "Opened models" the model, which contains the target object. In the field "objects that can be referenced", all objects available will be displayed according to their classes. Select the object that can be referenced and click on the button **"Add"**, to include it in the list "Reference target". Click on the OK button to close the window and to take over the just created object reference to the reference attribute (e.g. to the ADOxx Notebook).

Click on the "Open" icon , to open a saved model (see chap. 2.4.2, p. 257) in the ADOxx database. After having opened the model, the "Opened models that can be referenced" list is updated and you can put a reference to the objects contained in this model.

You can delete the existing object references shown in the "Reference targets" list, by selecting the reference to delete and clicking on the "Delete" icon .

If you want to undo the changes carried out, click on the cancel icon above the list of the reference targets.

2.2.12.10 Add Attribute Profile Reference

You can add an attribute profile reference in the ADOxx Notebook of the start object by clicking on the "Add" icon of the appropriate attribute of the attribute profile reference.

All already defined attribute profiles will be listed in the profile group structure in the window "<*Name of application library*> - Attribute profile selection" (see fig. 187).

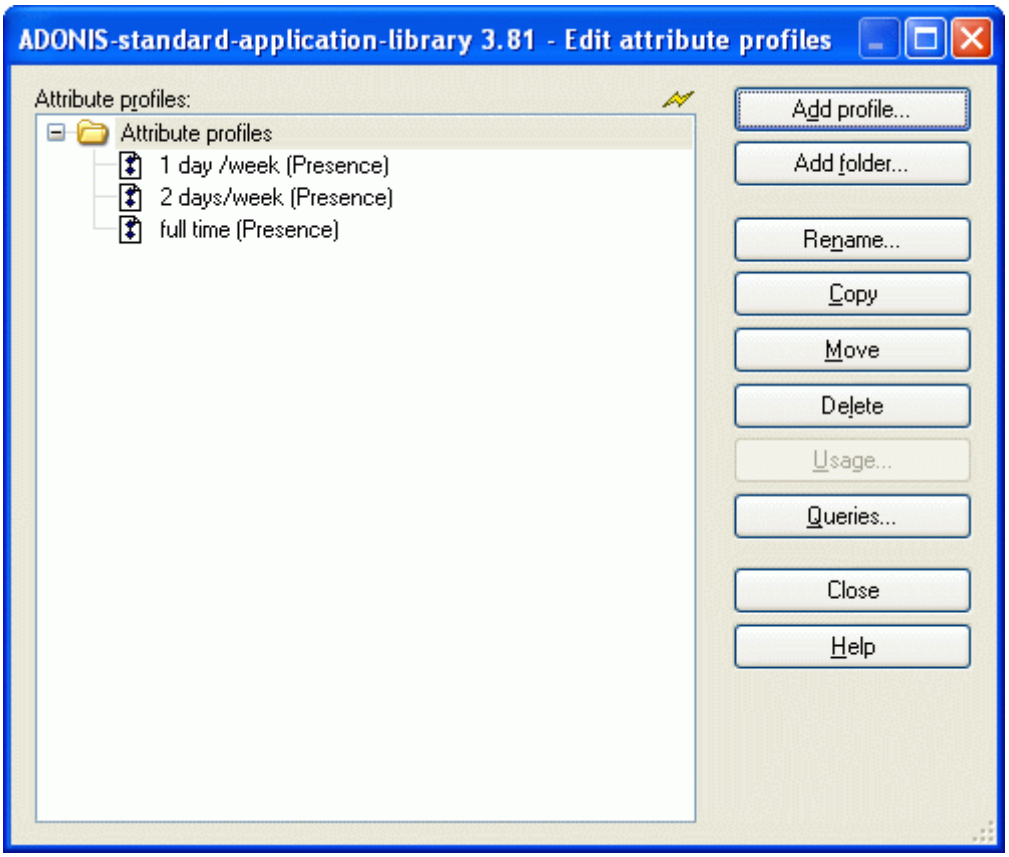


Figure 187: Add attribute profile reference

Select the attribute profile you wish to put a reference to and click on the button "Assign", to create the reference and to close the window. The attribute values defined in the attribute profile are taken over to the ADOxx Notebook of the current object.

Additionally, before the assignment of the attribute profile, you can, by clicking on the button:

"View" Display (see chap. 2.5.10, p. 296) the value of a previously selected attribute profile or display (see chap. 2.5.11, p. 297) at the same time the values of several previously selected attribute profiles.

"Utilisation" Display a list of the objects and models, in which the previously selected attribute profiles will be given a reference (see chap. 2.5.12, p. 297).

If you are entitled to edit attribute profiles, additional functions for the administration attribute profiles will be available in the window "<*Application library name*> - Attribute profile selection" (see fig. 188).

Hint: If you want to know more about the authorisation to edit attribute profiles, contact your ADOxx administrator.

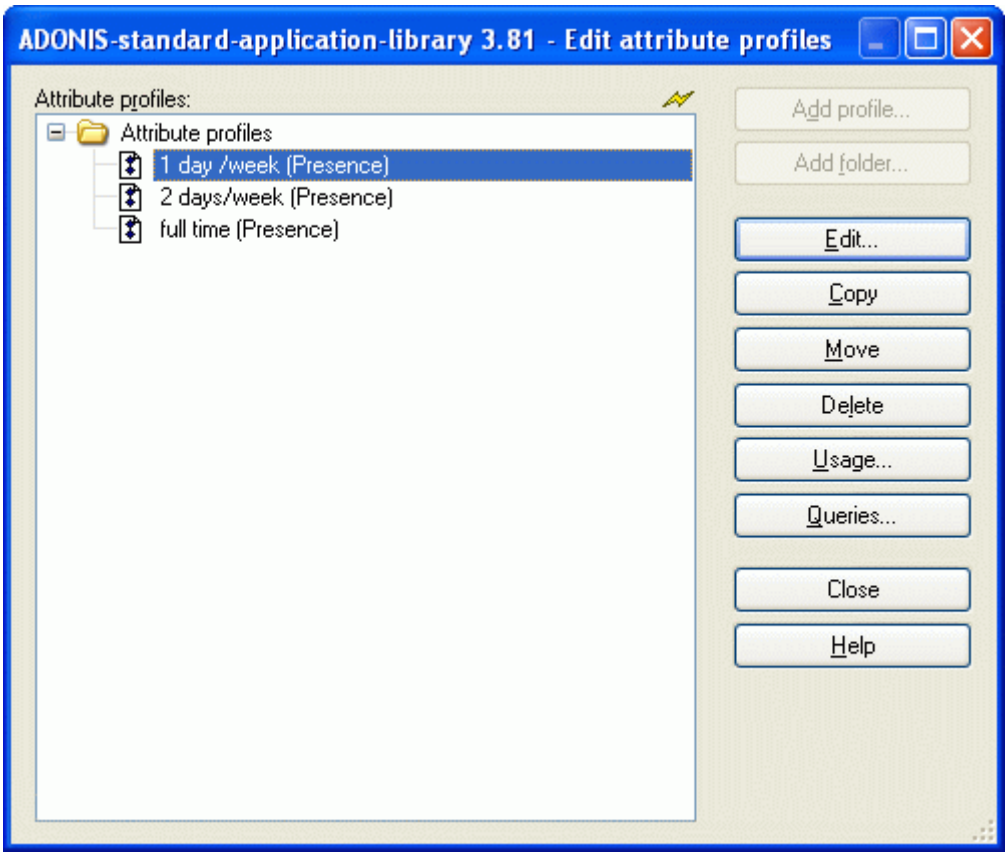


Figure 188: Add attribute profile reference (incl. Attribute Profile Management)

Click on the button:

"New attribute profile" To create a new attribute profile (see chap. 2.5.6, p. 294) in a previously selected attribute profile group.

"New version" To save a previously selected attribute profile group as a new version (see chap. 2.5.7, p. 295).

Note: This button is only available, if the attribute profiles are defined in an application library with time-related versioning (see chap. 10.1.2, p. 92).

"New group" To create a new attribute profile group in a previously selected attribute profile group or to create a new (main) attribute profile group (see chap. 2.5.1, p. 292) without previous selection.

"Rename" To assign a new name (see chap. 2.5.2, p. 293) to a previously selected attribute profile group.

"Edit" To edit (see chap. 2.5.8, p. 295) the values of a previously selected attribute profile or the values of several previously selected attribute profiles at the same time.

"Copy" To copy a previously selected attribute profile group (see chap. 2.5.3, p. 293) or a previously selected attribute profile (see chap. 2.5.12, p. 297).

"Move" To move a previously selected attribute profile group (see chap. 2.5.4, p. 294) or a previously selected attribute profile (see chap. 2.5.13, p. 298) to another attribute profile group.

"Delete"	To delete a previously selected attribute profile group (see chap. 2.5.5, p. 294) or a previously selected attribute profile (see chap. 2.5.14, p. 298).
"Utilisation"	To display a list of the objects and models, in which the previously selected attribute profiles will be referenced (see chap. 2.5.12, p. 297).

2.2.12.11 Random Generator

A Random generator sets a value to a variable to which it is connected. The value depends on an expression.

> **Hint:** ADOxx provides you with a support dialogue (see p. 213) for the definition of these expressions
>
> **Hint:** The attribute **"manual random generator"** (class "random generator") enables during the simulation (see chap. 4., p. 332) the manual assignment of values instead of the automatic assignment through the assignment expression.

The syntax of the assignment expressions (*ValAssign*) is based on the following grammar:

***ValAssign* ::=** `MathExpr { ; MathExpr }`

***MathExpr* ::=** `Term | ( MathExpr ) | Term Operator MathExpr`

***Term* ::=** `String | Numeric | Variable | Distrib`

***Operator* ::=** `+ | - | * | /`

- **Terms:**

  In the simplest case an assignment expression consists of a single term. There are three possibilities when defining a term:

  A term can be a **constant**. Depending on the type of the variable this can either be a constant of the type **"enumeration"** - e.g. 'standard case' - or of the type **"float"** - e.g. 365.

  A term can also be **another variable** - e.g. X - of the same type. So the value of the variable can be assigned to another variable.

  A term can also be a **distribution** (see p. 212). For variables of type enumeration a distribution type - the discrete distribution - is defined in ADOxx. Three different distribution types exist for variables of type float: exponential, uniform and normal distribution.

- **Arithmetical expressions:**

  Using the terms listed above, complex **arithmetical expressions** (*MathExpr*) can be formed. Terms of type float can be combined with the arithmetical operators +,-,* and brackets. The operators * and / take priority over the operators + and -. Terms of type enumeration can be combined using the operator + and brackets if necessary.

- **Separating a list of assignment expressions:**

  It is also possible to use more than one expression, by separating them with a semicolon. During the first simulation run, the first expression will be used, during the second run the second, and so on. When there are more runs than expressions, the last expression will be used repeatedly.

Examples for valid assignment expressions:

Some examples of valid expressions are now listed. X and Y are variables of type float, while S and T are variables of type enumeration.

Uniform(4; 10) : the variable is assigned with a random value, based on the distribution used in the example.

X+1 : the variable is assigned the value X plus 1.

X+Uniform(4; 10) : the variable is set to the value of the variable X increased by a random value, based on a uniform distribution.

X*Exponential(4; 10)+Y : the value of the variable X is multiplied by a random value (based on an exponential distribution), to this product the variable Y is added.

S+'a' : the character 'a' is added to the string S and the result is assigned to the variable.

'a';S+'b';S+'c' : during the first run the variable is assigned the character 'a'. During the second run the character 'b' is added to the variable S and the result assigned to the variable. During the third and following runs the character 'c' is added to the variable S and the result assigned to the variable.

Hint: A correct random generator is a precondition to perform analytical evaluation (see chap. 3.4, p. 325), assessment (see chap. 2.6, p. 303), process cost analysis (see chap. 5.4, p. 444) and human resource management (see chap. 7.2, p. 513).

Variables

Variables will be assigned (see chap. 2.2.12.11, p. 211) with values in ADOxx Business Process Models through assignment objects and will be queried during the process. This way, after branching in the process, paths can be passed according to the assignment of variables - defined in the transition conditions (see p. 215).

The display of variables depends on the definition in your application library. In the ADOxx-Default-Library, variables will be shown over objects of the class "variables".

ATTENTION: The variable names shall not start with **numerals from 0 to 9**, the **figures + - : = < > * / ( ) . , '** as well as a **blank** or a **return**. Moreover the **figures + - : = < > * / ( ) . , '**as well as **blanks** or **returns** are not authorised in variable names. The key words **AND**, **OR**, **NOT** and **TRUE** are not possible for variable names.

Distribution

Variables are assigned (see chap. 2.2.12.11, p. 211) with values in ADOxx using objects of type Random generator. This can be done using distributions. For variables of type float three **continuous distribution functions** are offered: exponential, uniform and normal distribution. For variables of type enumeration a **discrete distribution** is available.

The syntax for **continuous distribution** is:

- for **normal distribution**: normal (<number1>;<number2>)

  Enter the expected value and the standard deviation for the normal distribution, where <number1> represents the expected value and <number2> stands for the standard deviation.

  Example: `Normal(1200;100)`The variable has a normal distribution with an expected value of 1200 and a standard deviation of 100.

- For **exponential distribution**: exponential (<number>)

  Enter the expected value of the exponential distribution, letting <number> be 1 divided by the expected value.

  Example: `Exponential(0,002)`The variable has an exponential distribution with an expected value of 500.

- For **uniform distribution**: uniform (<number1>;<number2>)

  Enter the boundaries for the uniform distribution, where <number1> indicates the lower boundary and <number2> the upper boundary.

  Example: `Uniform(0;100)`The variable is uniformly distributed between the boundaries 0 and 100.

The syntax for a **discrete distribution** is:

Discrete (<Symbol1> <number1>;<Symbol2> <number2>; ...)

You can define two or more symbols with their corresponding probabilities (number1, number2,...). The sum of the probabilities must always equal one!

ATTENTION: The entry of symbol names is case-sensitive. The symbol names shall not **start** with **numerals from 0 to 9**, **blanks** or **returns** and the **figures : ( ) . , ; '**. Also **blanks** and **returns**, as well as the **figures: ( ) . , ; '** are not authorised in symbol names.

Examples:

Discrete (YES 0.6;NO 0.4):
The variable is assigned with a probability of 0.6 of being 'YES' and a probability of 0.4 of being 'NO'. Therefore two transition conditions <variable name>='YES' and <variable name>='NO' are valid and possible.

Discrete (a 0.5;b 0.3;c 0.1;d 0.1):
The variable is assigned with a probability of 0.5 of taking the value 'a', a probability of 0.3 of being 'b', a probability of 0.1 of taking the value 'c' and a probability of 0.,1 of taking the value 'd'. The four possible transition conditions therefore are: <variable name>='a', <variable name>='b', <variable name>='c' and <variable name>='d'.

Support Dialogue for Distributions

ADOxx provides you with help to enter a distribution in the ADOxx Notebook of objects of class "Random generator". This is found by clicking on the dialogue button next to the field "Value" . The window "<*class name*>: <*object name*> - Value" (see fig. 189) will appear.

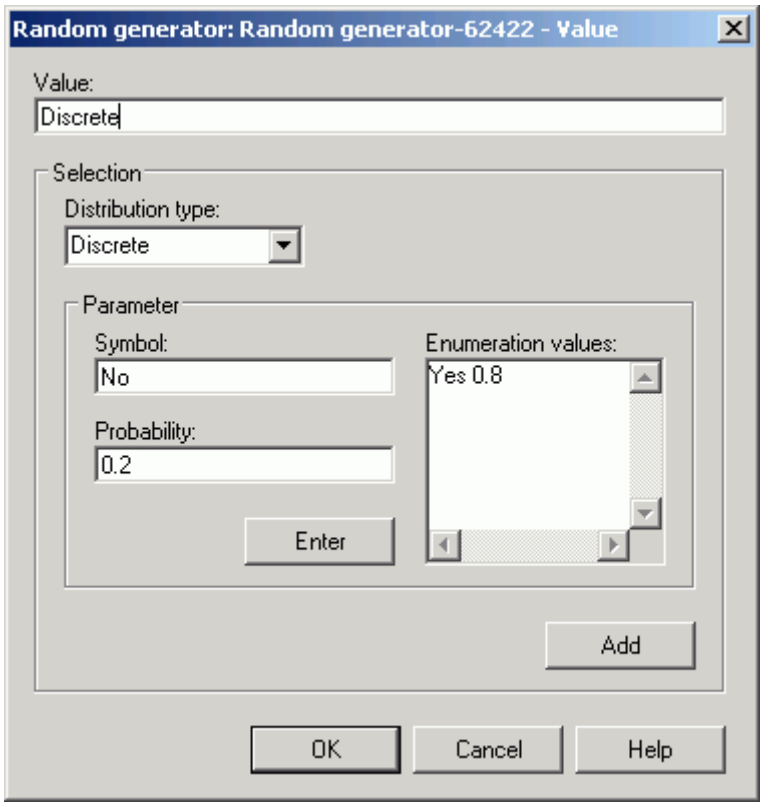


Figure 189: Variable assignment

The current distribution (see chap. 2.2.12.11, p. 211) can be found in the field **"value"**.

You can add distributions to any position within the **"Value"** field. Positioning the cursor within this field at the position at which you would like to place a new distribution does this. If you do not explicitly set the position of the cursor, then a new distribution will be added at the end of the field (separated by a';').(Please note that the first time you add a new distribution to a Random generator it will over-write the default distribution).

Choose the distribution (see p. 212) from the list box **"Distribution type"**. You can choose from the following distribution types:

- **"Discrete"**: discrete distribution function.
- **"Exponential"**: continuous distribution function, exponential distribution.
- **"Uniform"**: continuous distribution function, uniform distribution.
- **"Normal"**: continuous distribution function, normal distribution.

Depending on the distribution chosen you will be asked to enter different parameters.

By clicking the OK button the expression in the field **"Value"** will be syntactically checked and copied into the ADOxx Notebook.

ATTENTION: Arithmetical expressions (e.g. "X + 1") must be directly entered into the ADOxx-notebook

2.2.12.12 Transition Conditions in the Subsequent Relation

Connectors leaving decision objects must be assigned with transition conditions (see p. 215) or transition probabilities (see p. 220). These will be used (by the simulation algorithms and Analytical Evaluation) to determine under which conditions (or probabilities) a path will be chosen.

Transition conditions will be evaluated during the **simulation** (see chap. 4., p. 332), the **valuation** (see chap. 2.6, p. 303), the **process cost analysis** (see chap. 7.1, p. 497) and the **human resource planning** (see chap. 7.2, p. 513).

Transition probabilities will be raised during the **analytical Evaluation** (see chap. 3.4, p. 325) to determine paths.

> **Hint:** Connectors from paralleled objects can optionally be assigned with both transition conditions and transition probabilities (to define a conditional parallelism).

Transition Conditions

Transition conditions indicate under which conditions a subsequent connector will pass through a process model during the simulation.

A valid transition condition consists of:

- An expression (see p. 216)
- A probability (see p. 220) or
- A macro (see p. 217).

Transition conditions will be assigned in the ADOxx Notebook of the subsequent connector. By clicking on the "dialogue" icon for the attribute "transition condition", an input dialogue for the definition of a valid condition (see p. 216) will be displayed.

Examples for expressions:

X='YES': the condition is fulfilled, if the variable X has the value 'YES' (X is given a value based on a discrete distribution!).

(X='YES') AND (Y='YES'): the condition is fulfilled, if the variables X and Y both have the value 'YES' (X and Y are given a value based on a discrete distribution).

Z<1000: the condition is fulfilled, if the variable Z has a value greater than or equal 1000 (Z is given a value based on a continuous distribution!).

Z>=1000: the condition is fulfilled, if the variable Z is equal to or greater than 1000 (Z is continuously distributed!).

(X='NO') OR (Z!=1000): the condition is fulfilled, if X equals 'NO' or Z takes a value not equal to 1000 (X is discretely, Z continuously distributed!).

(X='YES') AND (Y='YES') AND (Z<1000): the condition is fulfilled, if the variables X and Y take the value 'YES' and Z a value less than 1000 (X and Y are discretely, Z continuously distributed!).

Z<=1000: the condition is fulfilled, if the value of variable Z is less than or equal to 1000 (Z is continuously distributed!).

$current process time$ > 00:000:00:12:00 the condition is fulfilled, if more than 12 minutes process time (i.e. cycle time) are passed.

Hint: Correct transition conditions are the prerequisite for the valuation (see chap. 2.6, p. 303), the simulation (see chap. 4., p. 332), the process cost analysis (see chap. 7.1, p. 497) and the Human Resource planning (see chap. 7.2, p. 513).

Define Transition Condition

To define transition conditions in the connectors of the relation "subsequent", ADOxx provides you with an input dialogue, which is to be opened from the notebook of each connector by clicking on the "dialog" icon on the right above the attribute field. Once you have clicked on the button, the window "Subsequent - Transition condition" will appear (see fig. 190).

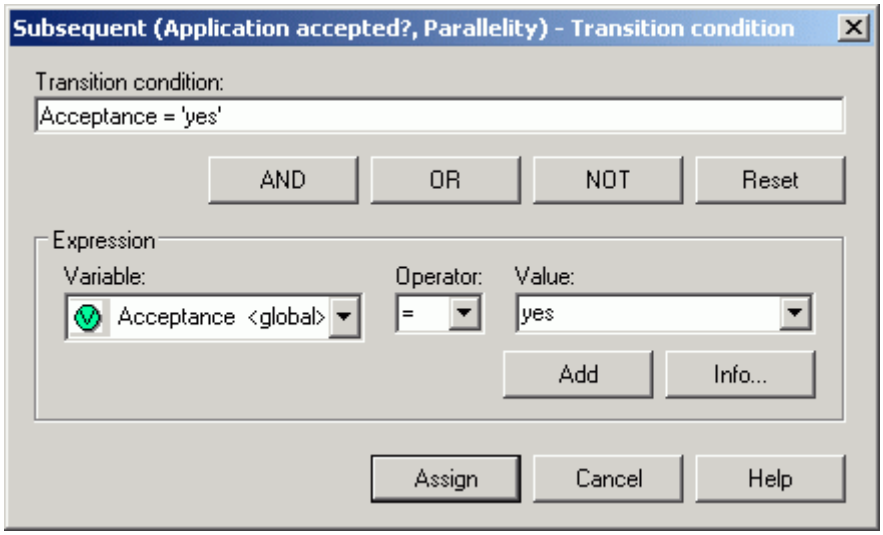


Figure 190: Define transition condition

The current attribute value will be displayed in the **"Transition condition" field**.

Define a valid expression by:

1. Selecting a variable available in the **"Variable" list**.

   This list contains all variables contained in the current model as well as all the variables from all other opened models, which have been defined as global. Moreover, the list contains the system variables <current process time>, <time of day> and <weekday>.

   The system variable **<current process time>** stands for the cycle time past since the process time. This variable can be used in all simulation algorithms.

   The system variable **<time of day>** stands for the current time of day. This variable can be evaluated only in the Workload Analysis (see chap. 4.4, p. 367).

   The system variable **<weekday>**stands for the current week day. This variable can be evaluated only in the Workload Analysis (see chap. 4.4, p. 367).

2. Selecting a comparison operator in the **"Operator" list**.

   If the selected variable is discretely distributed, only the comparison operators = and != (for unequal) are available. In other cases the operators <, <=, > and >= can also be used.

3. Select or enter an authorised value in the **"Value" field** (for discrete random generator).

By clicking on the **"Add" button**, you will transfer the defined expression to the "transition condition" field.

If you click on the **"Info" button**, a message window containing information about the current variable will appear.

Using the button **AND**, **OR** and **NOT**, you will be able to connect several expressions with another. Click on the **Button "Reset"**, to delete the entry in the field "transition condition".

Clicking on the **"Assign" button** will transfer the value in the field "Transition condition" in the attribute of the subsequent connector.

Macros

Macros supports you for complex and/or recurrent transition values and will be defined in the transition conditions of the subsequent relations.

You can **define a macro**, by adding a colon followed by the name of the macro to the condition (or probability).

Examples:

- `x='yes' : normal`

  A macro with the name "normal" is defined for the transition condition x = 'yes'.

- `(amount>10000) AND (x='no') : exemption`

  A macro with the name "exemption" is defined for the transition condition (amount>10000) AND (x='no').

- `0.95 : most likely`

  A macro with the name "most likely" is defined for the probability 0.95 (i.e. 95 %).

Hint: Macros can be defined only in the attribute "transition condition" in objects of the relation class "subsequent".

Hint: In this attribute, they can be used both in connection with transition conditions, which refer to variable values, and with probabilities. Macros for transition conditions, which refer to variable values, will be interpreted unlike macros for probabilities. This will be described in the following sections.

ATTENTION: The name of a macro must not start with **numerals between 0 and 9**, the **figures + - : = < > * / ( ) . , '** as well as a **blank** or a **return**. Moreover the **figures + - : = < > * / ( ) . , '** as well as **blanks** or **returns** are not authorised in the name of a macro.

If **macros are defined for transition conditions**, they are interpreted as aliases for the corresponding transition condition. Therefore, if a particular transition condition is used frequently, this means that it can be summarised in a short and concise term. This will be particularly helpful when using long and complicated transition conditions.

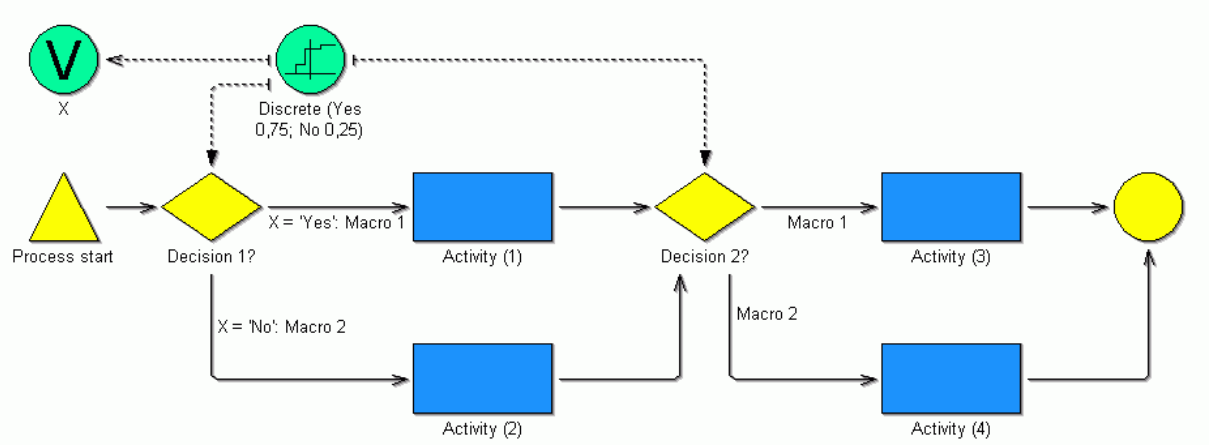


Figure 191: Example: macros of variables

In the example above (see fig. 191) "Macro1" replaces the transition condition "X = 'Yes'" and "Macro2" the transition condition "X = 'No'". After each decision, the transition conditions will be evaluated again. There are **4 possible paths** for the process, which are:

- Process start -> decision 1? -> **activity [1]** ->
  -> Decision 2? -> **activity [3]** -> End
- Process start -> decision 1? -> **activity [1]** ->
  -> Decision 2? -> **activity [4]** -> End
- Process start -> decision 1? ->**activity [2]** ->
  -> Decision 2? -> **activity [3]** -> End
- Process start -> decision 1? ->**activity [2]** ->
  -> Decision 2? -> **activity [4]** -> End

Macros are globally valid, comparable to global variables. A macro, which was defined on sub process level, is valid in the process models directly or indirectly **calling** the sub process and the process models directly or indirectly **called** by the sub process.

Macros can be defined independently of how they will be used, i.e. they only have to be defined once in the main process (including all sub processes).

If **macros for probabilities** are defined, they will always be used after branchings. This way, it is possible, to show **depending probabilities**.

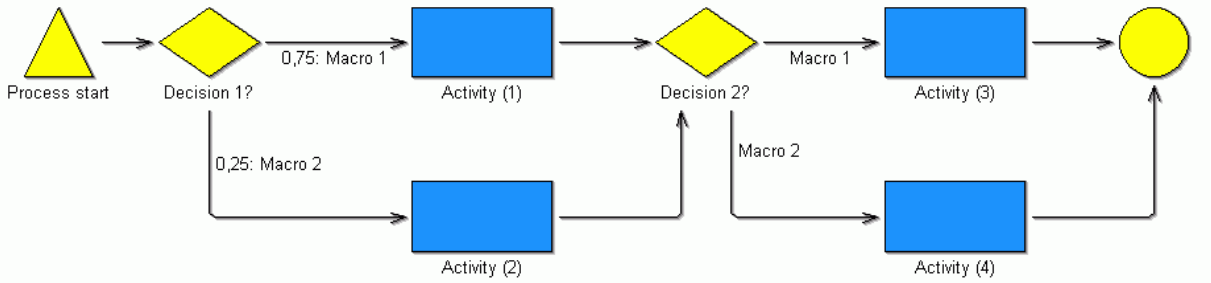


Figure 192: Example: Macros for probabilities

In the example above (see fig. 192) the probabilities will be evaluated after "Decision-1" for each process run. With a probability of 75%, the path leading to "activity-1" will be taken and "macro1" will be defined. With a probability of 25%, the path leading to "activity-2" will be taken and "macro2" will be

defined. - After "decision-2", the path which macros has been previously defined will always be taken. So there are **only 2 possible paths** through the process, which are:

- Process start -> decision 1? -> **activity [1]** ->
  -> Decision 2? -> **activity [3]** -> End
- Process start -> decision 1? -> **activity [2]** ->
  ->Decision 2? -> **activity [4]** -> End

Macros for probabilities must be defined at their first appearance in the process flow.

Syntax of Transition Condition

The syntax of the transition condition is based on the following grammar:

Condition or macro **:** *Empty* | *Condition* | *Condition* **:** *Macro name* | *Macro name*

Hint: An empty transition condition is interpreted as TRUE.

ATTENTION: The macro name shall neither start with **numerals between 0 and 9**, the **figures + - : = < > * / ( ) . , '**, nor **blank** or **return**. Moreover the **figures + - : = < > * / ( ) . , '** or **blanks** and **returns** are not authorised for macro names.

Condition:

TRUE |
NOT *Condition* |
(*Condition***)** |
Comparison |
Condition **AND** *Condition* |
Condition **OR** *Condition*

Hint: When NOT, AND or OR are used, "complex" transition conditions (i.e. which contain NOT, AND or OR) need to be put in brackets.

Comparison:

StringTerm StringOp StringTerm |
NumTerm NumOp NumTerm |
ProcessTimeTerm SysOp ProcessTimeTerm |
WeekdayTerm SysOp WeekdayTerm |
DaytimeTerm SysOp DaytimeTerm

StringTerm: *Variable* | *StringConstant*

StringOp: **=** | **!=**

NumTerm: *Variable* | *NumConstant*

NumOp: **=** | **!=** | **>** | **<** | **>=** | **<=**

SysOp: **=** | **!=** | **>** | **<** | **>=** | **<=**

ProcessTimeTerm: **$current process time$** | *Variable* | *ProcessTimeConstant*

WeekdayTerm: **$weekday$** | *Variable* | *WeekdayConstant*

DaytimeTerm: **$time of day$** | *Variable* | *DaytimeConstant*

ProcessTimeConstant: *Year* **:** *Day* **:** *Hour* **:** *Minute* **:** *Second*

DaytimeConstant: *Hour* **:** *Minute* **:** *Second*

Weekday Constant: **:Mon:** | **:The:** | **:Wed:** | **:Thu:** | **:Fri:** | **:Sat:** | **:Sun:**

Year: *Numerical value between 00 and 99*

***Day*:** *Numerical value between 00 and 365*

***Hour*:** *Numerical value between 00 and 23*

***Minute*:** *Numerical value between 00 and 59*

***Second*:** *Numerical value between 00 and 59*

Hint: The values ***Year***, ***Hour***, ***Minute*** and ***Second*** must be two-digit, the value ***Day*** three-digit.

ATTENTION: The variable name shall neither start with **numerals between 0 and 9**, the **figures + - : = < > * / ( ) . , '** nor with a **blank** or a **return** . Moreover the **figures + - : = < > * / ( ) . , '** and **blanks** or **returns** are not authorised for variable names. The key words **AND**, **OR**, **NOT** and **TRUE** are not authorised for variable names.

***StringConstant*:** **'** *Symbol name* **'**

ATTENTION: The symbol name (for the value assignment in a discrete distribution) shall neither start with **numerals between 0 and 9**, the **figures: / ( ) . , ; '** nor with a **blank** or a **return**. Moreover the **figures: / ( ) . , ; '** and **blanks** or **returns** are not authorised for symbol names.

***NumConstant*:** [*Sign*] *Digit+* |
[*Sign*] *Digit+ Point Digit**

Hint: If no sign has been given, it will be interpreted as '+'.

***Sign*:** **-** | **+**

***Digit*:** **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9**

***Point*:** **.** | **,**

Examples:

- (<elementary expression1>) AND (<elementary expression2>)
- (<elementary expression1>) OR (<elementary expression2>)
- NOT (<elementary expression2>)
- (<elementary expression1>) AND (NOT (<elementary expression2>))

An elementary expression is defined as:

<Variable name> <operator> <constant>, whereby:

- **<Variable name>** represents available one, already assigned.
- **<Operator>** takes on the forms "**<**", "**>**" or "**=**".
- **<Constante>** is a numerical or alphanumeric expression (for numerical expression only integers are taken into account, alphanumeric expressions must stand in simple quotation marks!).

Transition Probabilities

Transition probabilities are represented by a floating number > 0 and < 1 (e.g. **0.7**).

Hint: The **transition probabilities** defined in the "transition probabilities" attribute will be used in the **analytical valuation** (see chap. 3.4, p. 325) to determine the path.

Hint: The **transition probabilities** defined in the "transition conditions" attribute will be used in the **simulation** (see chap. 4., p. 332) and the **valuation** (see chap. 2.6, p. 303), the **process cost analysis** (see chap. 7.1, p. 497) as well as the **human resource planning** (see chap. 7.2, p. 513) to determine the path.

Hint: The **transition probabilities** defined in the "transition probabilities" attribute will be used in the **analytical valuation** to determine the path.

Hint: The **transition probabilities** defined in the "transition conditions" attribute will be used in the **simulation**, the **valuation**, the **process cost analysis** as well as the **human resource planning** .

2.2.12.13 Assigning and Defining Sub Processes

In the simulation (see chap. 4., p. 332) of business processes, all referenced sub processes in the Business Process Model must refer to existing sub processes . (When using hierarchical Working Environment models, the organisational units in the Working Environment model must refer to existing sub models.)

Hint: The correct use of referenced sub processes is also a pre-condition for running Analytical valuation (see chap. 3.4, p. 325), assessment (see chap. 2.6, p. 303), process cost analysis (see chap. 5.4, p. 444) and human resources planning (see chap. 7.2, p. 513) successfully.

Sub processes enable you to keep your business process models clearly structured or to reuse models.

ADOxx supports the following proceedings to model process models through the assignment and definition of sub processes in referenced sub process models:

- **Bottom-up modelling (see p. 221)** and
- **Top-down modelling (see p. 222)**.

The assignment of sub processes is done using a reference attribute (e.g. "called process"), into which you will add a reference to a model (see chap. 2.2.12.9, p. 207). This reference attribute will be appropriately defined in the application library.

Hint: Using sub processes will help you make your models more readable. Whenever you model sub processes, take care to design them in such a way that they are self-contained parts of the whole model. You should also use sub processes when describing routine or common processes, so that a common process need only to be modelled once ("re-usability").

Hint: ADOxx also allows recursive calls when processes are assigned to an object of the class "Subprocess", i.e. a process may call itself. Before you use such a construction, check it thoroughly. It is essential to have a suitable condition for exiting a recursive loop. When a recursive process call occurs, the system will therefore ask you, if you really want the process to call itself.

Bottom-up Modelling

Bottom-up modelling means that you start the model by modelling the sub processes (the processes of the lowest level), while modelling the higher processes later on. This means that you model your business process models from bottom to top.

Top-Down Modelling

Top-down modelling means that you begin modelling at the highest level (the main process). The main process is modelled completely. Possible sub process calls are integrated but the corresponding sub processes are modelled at a later stage. Your business process model is therefore modelled from top to bottom.

> **Hint:** In the ADOxx-Default-Library (see chap. 16., p. 802), the top-down Modelling will be additionally supported through the function "Transform", using which an object of the class "Activity" in Business Process Models can be transformed to an object of the class "Subprocess" .

2.2.12.14 Assigning Performers

Business process models and working environment models are linked on the one hand by defining application models (see chap. 2.4.16, p. 284) and, on the other hand by assigning performers to activities. Such links enable the capacity (see chap. 4.3, p. 354) and Workload (see chap. 4.4, p. 367) simulations.

> **Hint:** Assigning performers is also a precondition for Process Cost Analysis (see chap. 5.4, p. 444).

Assigning performers means that one or a group of performers are entered into the attribute "Performer" of each object of the class "Activity". These performers can execute the particular activity within the simulation. The performers are assigned to activities by AQL expressions (AQL = ADOxx Query Language (see chap. 14., p. 790)).

ADOxx provides a support dialogue for assigning performers in objects of the class "Activity". Open this dialogue window by clicking on the "Dialog" button above the attribute field in the notebook.

The window "<*class name*>:<*object name*> - Performer" (see fig. 193), in which you can define the activity's performer, is opened.

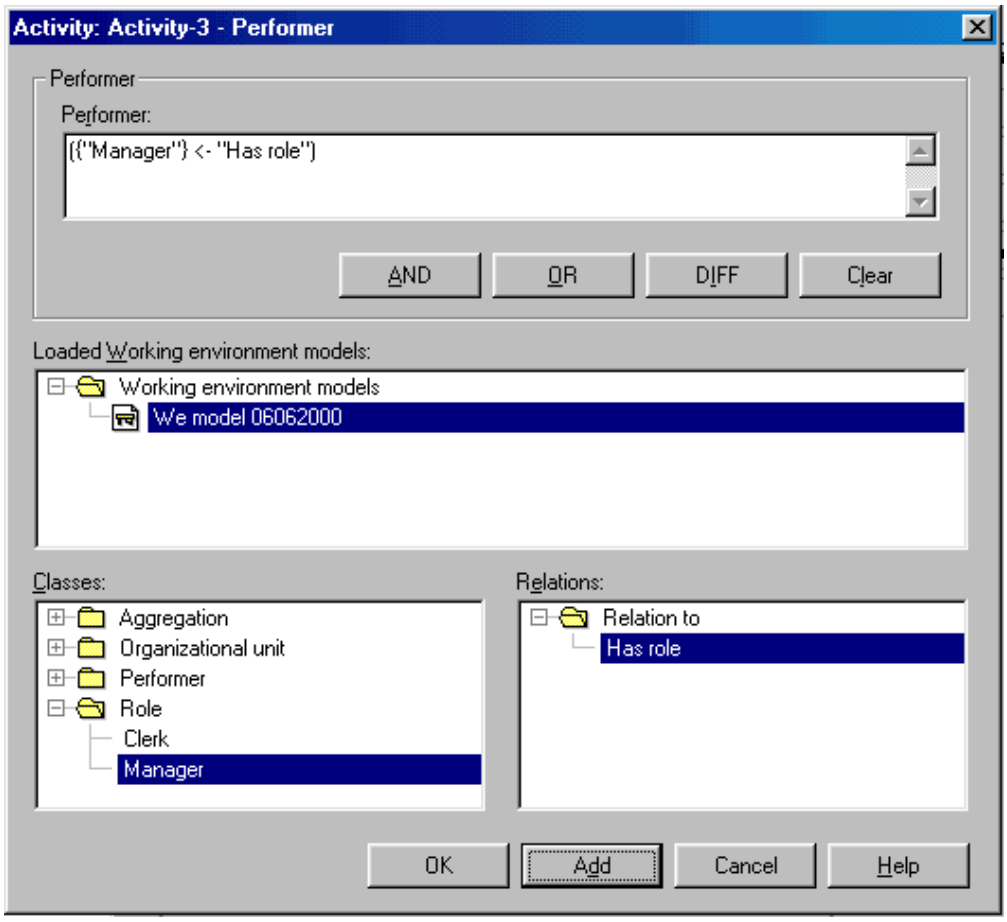


Figure 193: Assigning performers

In order to define the performer(s) or groups of performers who can execute a particular activity, you must first select the working environment model, in which the performers to be assigned, their roles and their organisational units have already been modelled.

ATTENTION: The working environment model must be opened in order to enable the assignment of performers!

As soon as you have selected the working environment model, the performers, groups (=organisational units, roles etc.) and resources are displayed in the field **"Classes"**.

Exactly one performer executes an activity. If several performers are listed in the field "Performer", ADOxx selects one performer (see p. 363) from this group during simulation.

To assign one or more performers to an object of the class "Activity", select the appropriate performer(s) in the field **"Classes"**. Clicking on the button **"Add"** causes the performers selected to be entered as valid AQL (see chap. 14., p. 790) expressions into the field **"Performer"**.

If an "Activity" is to be executed by performers of a group characterised by a relation in the working environment model (e.g. all performers with the role "Clerk" or all performers belonging to the organisational unit "Department A"), select the respective group by clicking on it. The field **"Relations"** shows the relations leading to or from the grouping selected. Click on the relation required. When selecting a grouping which has only one relation, you must still select both the grouping and the relation in order to generate a valid AQL expression.

When clicking on the **"Add"** button, the grouping and the relation selected are entered as valid AQL expressions in the field **"Performer"**.

The buttons "**AND**", "**OR**", "**DIFF**" enable you to combine several AQL (see chap. 14., p. 790) expressions (self-defined or added). Delete the current AQL expression in the field **"Performer"** by clicking on the button **"Clear"**. It is possible to type the required AQL expression directly into the Performer field.

When defining a performer the expression you entered or added are displayed in the field **"Performer"**. Clicking on the OK button causes the system to check the expression's syntax and then enter it into the object's notebook.

Clicking on the "Cancel" button closes the window "Performer assignment" while maintaining the previous value of the "Performer" attribute.

Assigning performers from hierarchical working environment models

If your current application library supports hierarchical Working Environment models (see chap. 2.2.12.13, p. 221), the position of the performer must be specified more precisely (see fig. 194) once you have clicked on the button **"Add"** (see fig. 193).

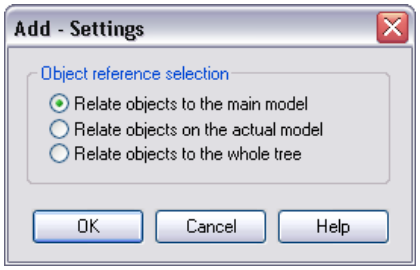


Figure 194: Add a performer from a Working Environment hierarchy

Select the option **"Refer objects to the main model"**, if the performer has to be searched for in the main model of the model hierarchy. The main model is the Working Environment model, which was given during the definition of the application model (see chap. 2.4.16.2, p. 285) (see fig. 195).

Select the option **"Refer objects to the current model"**, if the performer has to be searched for in the currently selected model of the field "Loaded Working Environment models" (see fig. 195).

Select the option **"Refer objects to the whole tree"**, if the performers have to be searched for in the whole model hierarchy (see fig. 195).

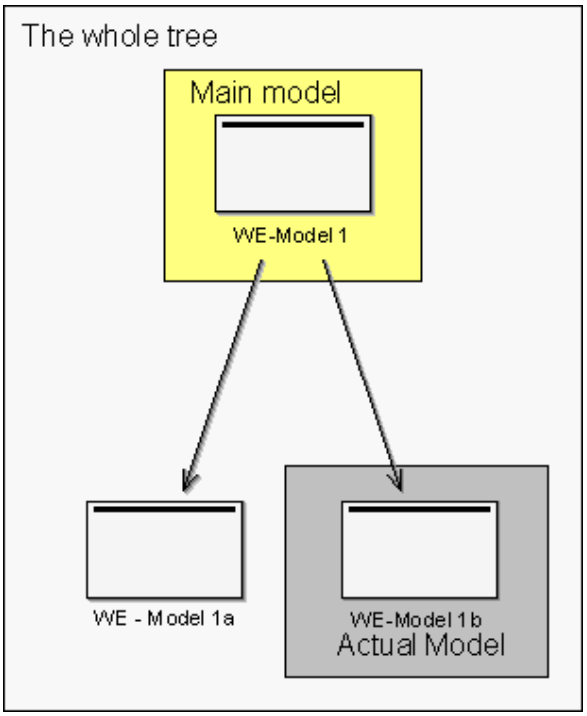


Figure 195: Terms in the hierarchical Working Environment models

Hint: For the simulation of hierarchical Working Environment models, the main Working Environment model must be contained in the application model (see chap. 2.4.16, p. 284). The sub Working Environment models referenced in this model will be automatically loaded during the simulation.

Assigning performers to sub processes

The performer assignment is - according to an appropriate definition in your application library - also possible in sub process objects.

This way you can define a "standard performer assignment expression", which will be analysed during the simulation, if no performer has been assigned to the activities of the referenced process.

Example:

In the following example (see fig. 196), three Business Process Models (main model, referenced model and referenced model of the second level) are represented. Additionally the performers of the activities are shown.

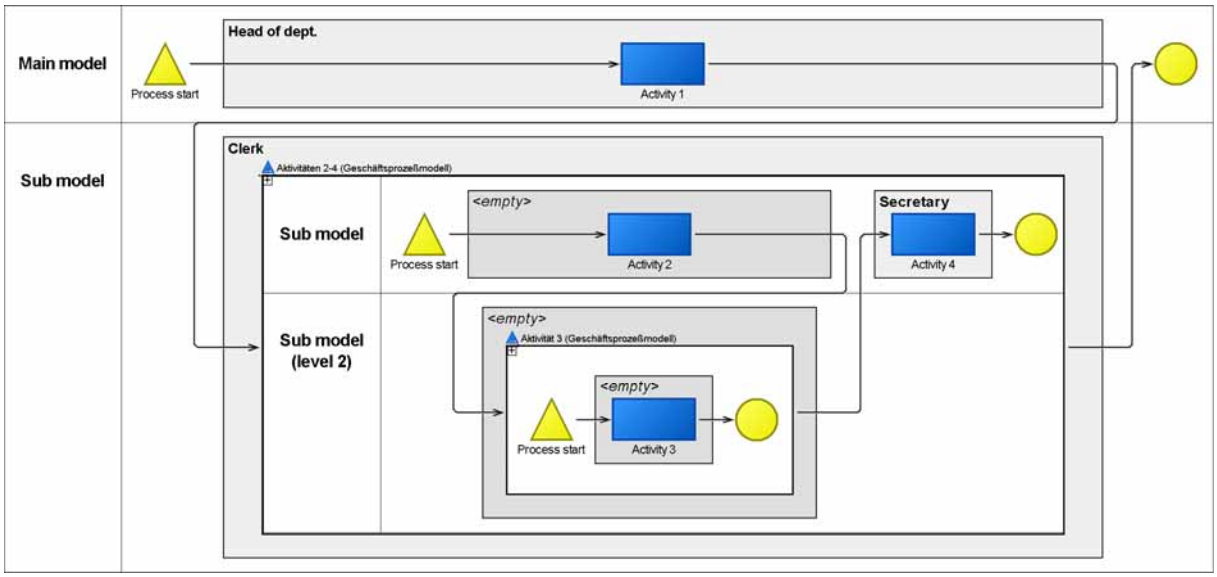


Figure 196: Assigning performers to subprocesses

In the main model the performer expression "Responsible person" will be defined in the sub process object of a submodel. In the sub model, no performer is defined in the activity "Activity 2", i.e. the standard value from the sub process object in the main model will be used during the simulation and so assigned to the "Activity 2" of the performer "Responsible person".

No performer is assigned in the activity of the sub model of the second level. During the simulation, the standard value from the sub process object will be used in the submodel. Since this standard value is also not assigned in the submodel, the standard value from the sub process object in the main model will be used and so assigned to the "Activity 3" of the performer "Responsible person".

Defining Probabilities

In the simulation algorithms Capacity Analysis (see p. 363) and Workload Analysis (see p. 381), the performer assignment expression will be evaluated and the activity will be randomly (uniformly distributed) assigned to one of the resulting performers. The standard uniformly distributed random selection of the performer can however be influenced through the entry of probabilities in the performer assignment expression.

For this, adhere to the following syntax:

```
{ <AQL expression> <Probability>; }
<AQL expression> <Probability>
```

Example:

An activity should be assigned with a probability of 20% to a performer of the role "Clerk" and with a probability of 30% to a performer of the role "Secretary" and with a probability of 50% to a performer of the role "Temporary help".

Give the following performer assignment expression:

```
({"Clerk": "Role"} <- "has role") 0.2;
({"Secretary": "Role"} <- "has role") 0.3;
({"Temporary help": "Role"} <- "has role") 0.5
```

ATTENTION: The single AQL expressions (see chap. 14., p. 790) will be separated by the figure **';'** . **There shall be no ';' at the end of the whole expression!**

Hint: The sum of the probabilities must equal 1.

2.2.12.15 Allocating Resources

Allocating resources means that one or more actual resources are assigned to each object of the class "Resource" via the "Selection" attribute. These resources are required when executing an activity. Allocated resources are evaluated in the Simulation Component during Capacity Analysis (see chap. 4.3, p. 354) and Workload Analysis (see chap. 4.4, p. 367).

The allocation of resources is described by AQL expressions (AQL = ADOxx Query Language (see chap. 14., p. 790)).

ADOxx provides a support dialogue for the allocation of resources. Call this support dialogue by opening the ADOxx Notebook of an object of class "Resource" in a business process model and then clicking on the "Dialog" button next to the input field "Selection". The window "<*class name*>:<*object name*> - Selection" (see fig. 197) appears.

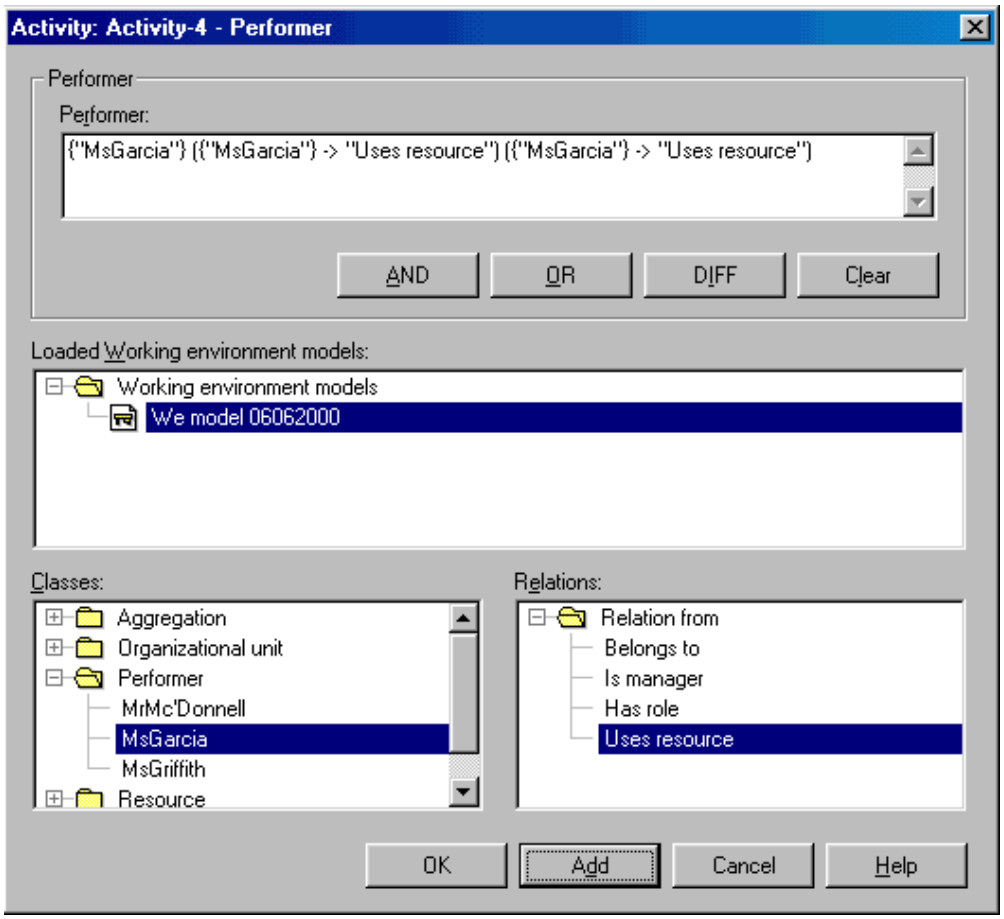


Figure 197: Allocating resources

ATTENTION: Resources can only be allocated in business process models. The resources in the working environment models represent resources, which actually exist in the company. The resources in the business process models are references to the resources in the working environment model and represent the resources used while a business process is performed.

The field **"Resource"** holds the current resource expression. Before a simulation runs, this expression is the attribute value entered into the attribute "Selection" of the ADOxx Notebook from which the dialogue has been called.

Before resources can be allocated, you must first select the working environment model in which the resources to be allocated are modelled.

> **ATTENTION:** The working environment model from which you wish to allocate resources must be loaded to enable the assignment of resources!

After the working environment model has been selected, the performers, groups (=organisational units, roles etc.) and resources are shown in the field **"Classes"**.

Exactly one resource is allocated to an activity. If the "Selection" attribute contains more than one resource, ADOxx selects one resource (see p. 364) during simulation. In order to allocate one or more resources to an object of class "Resource", click on the resources in the field **"Classes"**. Click on the **"Add"** button to enter the resources selected as a valid AQL expression into the field **"Resource"**.

If an object of the class "Resource" is to be allocated via a performer or a group using certain resources, please select the appropriate performers or groups by mouse click. The field **"Relations"** shows the relations leading to or from the performer or group selected. Click on the required relation.

> **Hint:** If you select more than one performer or a group of performers, the relations shown in the relation list represent the intersection set of the relations defined for the performers or groups selected.

By clicking on the **"Add"** button the performer or group as well as the relation selected are entered as a valid AQL expression into the **"Resource"** field.

The special performer "current performer" represents the current performer of the activity at the time of its execution (during a simulation). This means that the resource actually used by that performer can be specified. The "current performer" refers to a single entry and cannot be combined with other expressions (see AQL syntax).

The buttons "**AND**", "**OR**", "**DIFF**" enable you to combine several AQL expressions (see chap. 14., p. 790). You can delete the current AQL expression in the **"Resource"** field by clicking on the button **"Clear"**.

When defining a resource the expressions entered are displayed in the **"Resource"** field. Clicking on the OK button causes the system to check the expression's syntax and to copy it to the object's notebook.

Clicking on the "Cancel" button closes the window "Resource Selection" while maintaining the previous value of the attribute "Selection" in the notebook.

> **Hint:** The allocation of resources is also a prerequisite for process cost analysis (see chap. 7.1, p. 497).

2.2.12.16 Performer Calendar

The performer calendar allows you to define working time profiles for objects of class "Performer". This calendar is evaluated by the "Workload Analysis" (see chap. 4.4, p. 367) and specifies, for a one year period, the presence of a performer.

> **Hint:** The performer calendar is a standardised calendar. This means that it does not refer to a specific year and that it comprises **365 days**. Leap years are not considered and the first day is always a Monday, the 1st of January.

Open the calendar by clicking on the button **"Calendar"** in the "simulation data" chapter of the ADOxx Notebook of an object of class "Performer".

The window "<performer's name> - Calendar" (see fig. 198, p. 229) appears.

Display Performer's Calendar

The window "<*performer's name*> - Calendar" (see fig. 198) shows the current working time profile of the performer selected. The default setting for the class "Performer" is that Monday to Friday are working days, Saturday and Sundays are days off (bank holidays, vacation etc. are not taken into account).

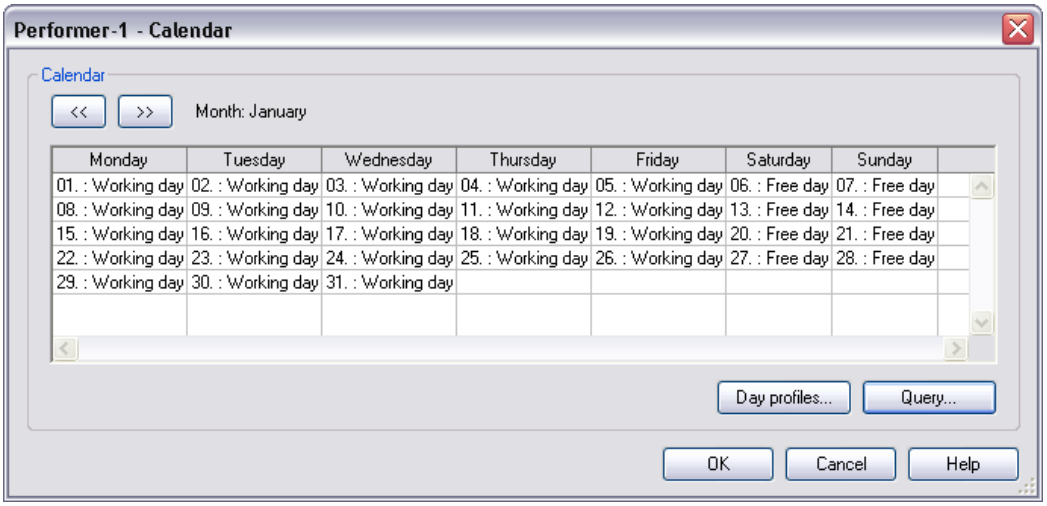


Figure 198: Performer's calendar

The working time profiles are displayed for one month. You can show the previous and following months by clicking on the buttons **"<<"** or **">>"** respectively. The name of the current month is shown to the right of the two buttons.

If the names of the day profiles are too long to fit in the standard column width of the calendar, you can adjust the column width accordingly. Double-click on the column's header (e.g. "Monday"). The width of the corresponding column will be set in such a way that all day profile names fit into the column.

Click on the OK button to close the calendar and to confirm the changes made.

In addition to the existing day profiles "Working day" and "Free day" you can define new day profiles (see p. 230) and change (see p. 231) or delete (see p. 232) existing profiles. By clicking on the **"Day profiles"** button, the window "Day profiles" (see fig. 199) listing all available day profiles will appear.

Click on the button **"Query"** if you wish to calculate the actual availability of the performer (see p. 234).

Displaying Day Profiles

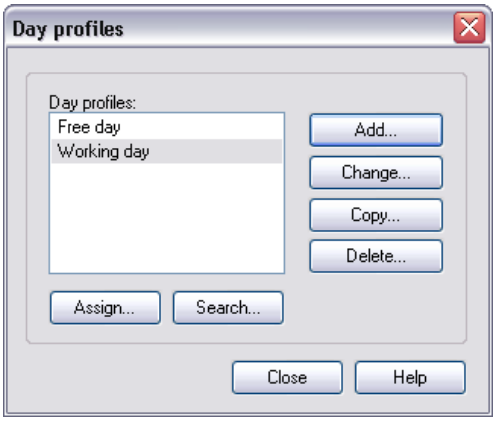


Figure 199: Displaying day profiles

The window "Day profiles" (see fig. 199) enables you to:

- Add new day profiles (see p. 230).
- Change existing day profiles (see p. 231).
- Copy day profiles from other objects of the class "Performer" (see p. 233) - also from other models.
- Delete existing day profiles (see p. 232).
- Assign existing day profiles to days or periods of time (see p. 232).
- Search for days to which day profiles have already been assigned (see p. 234).

Hint: The "Global change" (see chap. 2.2.14, p. 243) function allows you to copy the settings of one performer calendar into the calendars of other performers.

Adding Day Profiles

You can add new day profiles by clicking on the button **"Add"** in the window "Day profiles" (see fig. 199, p. 230) . The window "New day profile" (see fig. 200) opens up, into which you must enter a unique name for the new day profile (a name which has not yet been used for another day profile).

Figure 200: New day profile

Click on the OK button or press the enter key to continue the process. Clicking on the "Cancel" button closes the window and returns to the window "Day Profile".

Once you have entered a unique name for the day profile to be added and clicked on the OK button, the window 'Day profile "<name of the day profile>" - time intervals' (see fig. 201) appears.

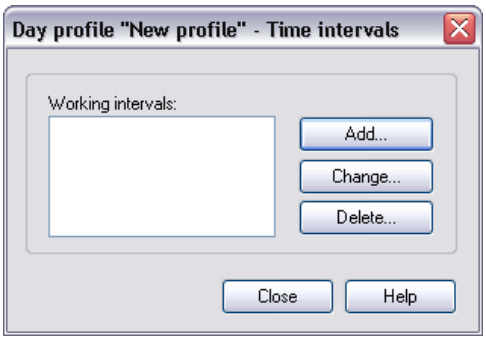


Figure 201: Time intervals

Click on the **"Add"** button and enter the start and end time of the interval in the window "New interval" (see fig. 202).

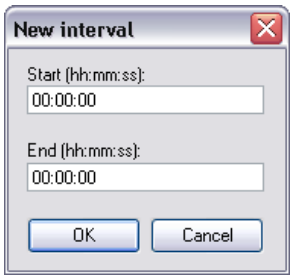


Figure 202: Add interval

Please make note of the following points when entering the interval:

- Time intervals must be entered in the time format "**hh:mm:ss**".
- The starting time must be **earlier** than the ending time.
- If more than one time interval is defined for a day profile, the intervals **may not** overlap.

After entering the interval, click on the OK button to copy the new interval to the list of working times in the window 'Day profile "<name of the day profile>" - to copy time intervals' (see fig. 201, p. 231).

Changing Day Profiles

If you wish to change an existing day profile, select the appropriate profile in the "Day profiles" (see p. 230) window and then click on the button **"Change"**. The window 'Day profile "<*name of the day profile*>" - time intervals' (see fig. 203) appears, showing the intervals already defined in the field **"Working intervals"**.

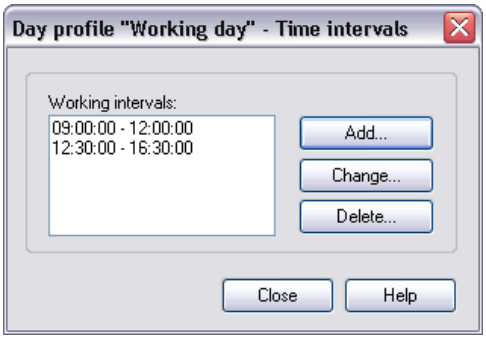


Figure 203: Changing day profiles

If you wish to **change an interval**, select the interval to be changed and then click on the **"Change"** button.

Enter the new starting time and/or the new ending time in the window "Change interval".

Please make note of the following points when entering the interval:

- Time intervals have to be entered in the time format "**hh:mm:ss**".
- The starting time must be **earlier** than the ending time.
- If more than one time interval is defined for a day profile, the intervals **may not** overleap.

After entering the interval, click on the OK button to copy the new interval to the list of working times in the window 'Day profile "<name of the day profile>" - to copy time intervals'.

To **delete an interval**, select the appropriate interval in the window 'Day profile "<*name of the day profile*>" - time intervals' and click on the **"Delete"** button.

A message asking you if you really wish to delete the time interval appears.

Deleting Day Profiles

If you wish to delete an existing day profile, select it by a mouse click in the window "Day profiles" (see fig. 199, p. 230) and then click on the button **"Delete"**.

A window appears, in which you have to confirm that you really wish to delete the day profile selected.

Click on the "Yes" button to continue deleting the profile. Click on the "No" button to stop the process.

Assigning Day Profiles

You can assign the day profiles defined to certain days and/or periods of time. There are two ways to do this:

- For **a certain day**.
- For **certain days of the week and/or periods of time during a year**.

In order to assign a day profile to **a specific day**, double-click on the particular day in the window "<*performer's name*> - Calendar" (see fig. 198, p. 229). The window "<Date>" (see fig. 204), showing the current day profile and the available (assignable) day profiles, are displayed.

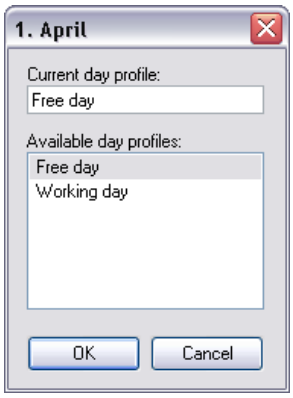


Figure 204: Assign profile for a certain day

Select the day profile to be assigned from the list **"Day profiles"** and click on the OK button. The selected day profile will then be assigned. Clicking on the "Cancel" button cancels the process.

If you wish to assign a day profile to **certain days of the week and/or periods of time during a year**, select the day profile to be assigned in the window "Day profiles" (see fig. 199, p. 230) and then click on the "Assign" button. The window "<name of day profile>" - assign to days' (see fig. 205) opens.

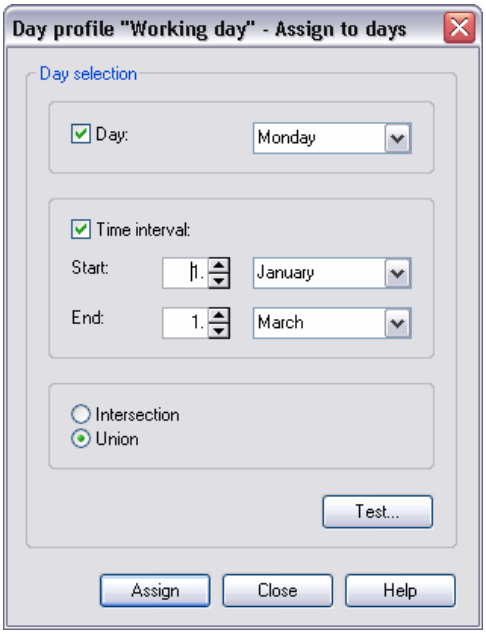


Figure 205: Assign day profiles

Activate the option **"Day"** and choose the day that you want to assign the day profile to. If you click on the OK button, the current day profile is assigned to all days (within the year) of the type selected.

You can define a period of time by selecting the option **"Time interval"**. After you have entered the period, click on the OK button to assign the current day profile to all days within the period selected.

In addition, you can form a set of days the current day profile will be assigned to by activating the options "Day" **and** "Time interval". The following options are at your disposal:

- **Intersection**: AND operator (standard setting); All days identical to the day of the week selected within the period defined will be determined (e.g. all Fridays from March, 1st to April, 30th).
- **Union**: OR operator; All days identical to the day of the week selected **or** within the period of time specified are determined (e.g. all Fridays of a year and all days from March, 1st to April, 30th).

After you have selected the day of the week, entered the period of time and the chosen operator, click on the "Assign" button to update the days in the calendar.

A preview of the days selected (see p. 233) can be seen before you assign the current day profile to those days. Click on the **"Test"** button to call this preview.

Show Specified Days

A preview of the days specified can be seen in the window "Specified days".

It is possible to save the content of the window "Specified days" to a file or print it out.

Copying Day Profiles

If you wish to copy an existing day profile, go to the "Day profiles" (see fig. 199, p. 230) window, select the day profile and click on the **"Copy"** button.

The window "Copy day profiles" (see fig. 206) appears. There you can select a reference object (specific performer, who has the same working time or the same days off) from another opened working environment model as well as the day profiles to be copied.

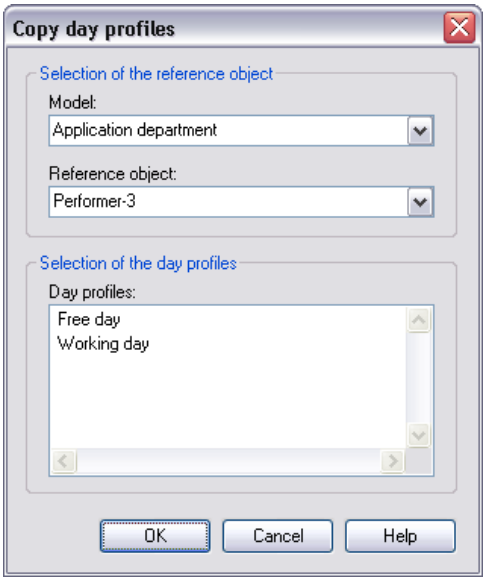


Figure 206: Copy day profiles

Confirm the day profile(s) to be copied by clicking on the OK button or close the window by clicking on the cancel button.

The window "Day profiles" will appear again, containing the copied day profile.

Searching for day profiles

If you wish to see all the days to which an existing day profile has been assigned, click on the respective day profile in the window "Day profiles" (see fig. 199, p. 230) and then click on the "Search" button.

The window 'Assignment of the day profile "<name of the day profile>"' lists all days in chronological order to which the previously selected day profile has been assigned.

You can save the contents of the window 'Assignment of the day profile "<name of the day profile>"' to a file or print it out.

Calculating the Time of Presence

The evaluation of the performer calendar enables you to calculate the time of presence (quantity of time during which a particular performer is actually available for work) of the respective performer during a certain period of time, based on the calendar defined.

Clicking on the button "Query" in the window "<*performer's name*> - Calendar" (see fig. 198, p. 229) opens the window "Calculate time of presence" (see fig. 207).

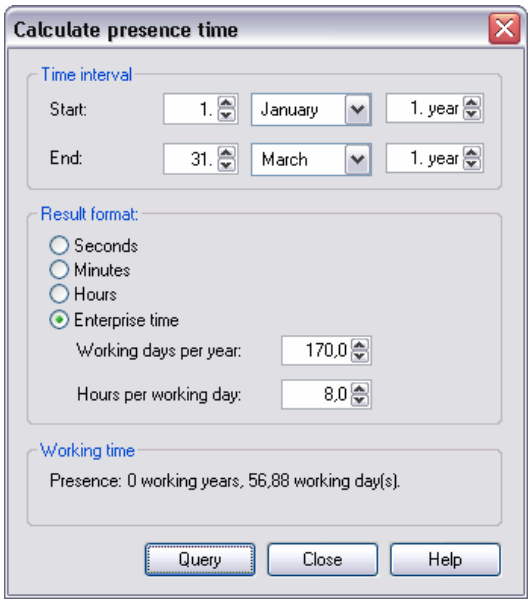


Figure 207: Calculate the time of presence

Specify the time interval and the result format and click on the "Query" button. The time of presence is shown in the format selected in the field **"Working time"**.

2.2.12.17 Process Calendar

The process calendar offers you the possibility to define occurrence profiles for every business process. The calendar is evaluated by the simulation algorithm "Workload Analysis" (see chap. 4.4, p. 367) and states the period of occurrence and the probability of occurrence of a given business process referring to one year.

> **Hint:** The process calendar is a standardised calendar. This means that it does not refer to a specific calendar year and consists of **365 days**. Leap years are not considered and the first day is always Monday, 1st of January.

Click on the button **"Process calendar"** in the chapter "simulation data" in the ADOxx Notebook of an object of the class "Process start" to open the process calendar.

When clicking on the button the window "<process start name> - Process calendar" will appear.

Display Process Calendar

The window "<*process start name*> - Process calendar" (see fig. 198) presents the current occurrence profile of the process selected. The default setting for the class "Process start" is that the days from Monday to Friday are defined as working days, Saturdays and Sundays are days off (bank holidays etc. are not taken into account).

The working time profiles are displayed for one month. You can view the previous and following months by clicking on the buttons **"<<"** or **">>"** respectively. The name of the current month is shown to the right of the two buttons.

If the names of the day profiles are too long to fit in the standard column width of the calendar, you can adjust the column width accordingly. Double-click on the column's header (e.g. "Monday"). The width of the corresponding column will be set in such a way that all day profile names fit into the column.

Click on the OK button to close the calendar and to confirm the settings.

Click on the "Cancel" button to close the calendar and to discard any changes.

If you click on the cancel button, after making some changes to the calendar, a second message will appear asking you to confirm that you definitely want to exit without saving your changes. Click on the "Yes" button to close the calendar without executing any changes. You can return to the window "Process start: <process start name> -- calendar" by clicking on the "No" button.

In addition to the existing day profiles "Working day" and "Free day" you can define new day profiles, change or delete them. By clicking on the **"Day profiles"** button, the window "Day profiles" (see fig. 199) listing all available day profiles (see p. 236) will appear.

Click on the **"Query"** button to calculate the number of processes (see p. 239).

Day Profiles

The following functions can be carried out in the "Day profiles" (see fig. 199) window:

- Add (see p. 236) new day profiles.
- Change (see p. 236) existing day profiles.
- Copy (see p. 239) day profiles from other objects of the class "Process start", also from other models.
- Delete (see p. 238) existing day profiles.
- Assign (see p. 238) existing day profiles to days and/or time periods.
- Search (see p. 239) for days already assigned to an existing day profile.

Hint: The function "Global change" (see chap. 2.2.14, p. 243) allows you to copy the settings of the process calendar into other process calendars.

Adding Day Profiles

You can define new day profiles by clicking on the **"Add"** button in the window "Day profiles". The window "New day profile" (see fig. 200) appears, in which you must enter a unique name for the new day profile (a name that has not yet been used).

Click on the OK button or press the Enter button to continue generating the new day profile. Clicking on the Cancel button interrupts the process and returns you to the window "Day profiles".

Once you have entered a unique name for the day profile to be added and have clicked on the OK button, the window Day profile appears "<name of the day profile>" - time intervals".

Click on the **"Add"** button. The window "New interval" will be displayed.

Changing Day Profile

If you wish to change an existing day profile, select the appropriate profile in the window "Day profiles" and then click on the **"Change"** button. The window 'Day profile "<*name of the day profile*>" - time intervals' (see fig. 208) appears, showing the intervals already defined in the field **"Intervals"**.

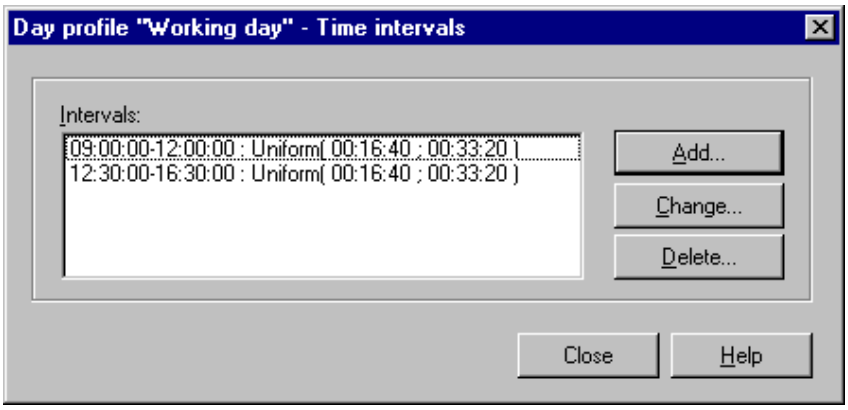


Figure 208: Change day profiles

In this window you can:

- Add (see p. 237) a new occurrence (or frequency) interval.
- Change (see p. 238) an existing interval of occurrence.
- Delete (see p. 238) an existing interval of occurrence.

Adding Intervals

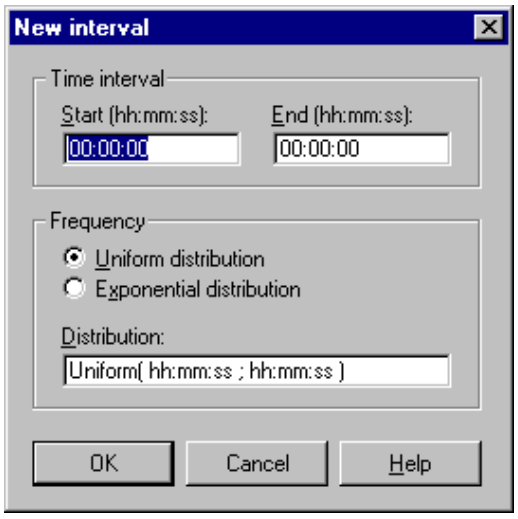


Figure 209: Adding intervals

Enter the start and end time of the interval into the field **"Time interval"** (see fig. 209). When doing this, consider the following points:

- Time intervals must be entered in the time format "**hh:mm:ss**".
- The starting time must be **earlier** than the ending time.
- If more than one time interval is defined for a day profile, the intervals **may not** overlap.

Define the probability of occurrence within the time interval in the field **"Frequency"** (see fig. 209). You can choose between the following:

- **Uniform distribution**: A business process' probability of occurrence within the given interval is distributed uniformly between a lower and upper boundary (period of time).

Example:

If you enter the frequency "Uniform (00:30:00; 02:00:00)", the business processes will be triggered from every 30 minutes (minimum) to every two hours (maximum).

- **Exponential distribution**: A business process' probability of occurrence within the interval given is distributed exponentially with the expectancy 1/E (E = period of time in seconds).

  Example:

  If you enter the frequency "Exponential (0,00027778)", the business processes will be triggered with an expectancy of 1/3600 (corresponds to one hour) for the intervals of time.

After you have entered the interval and the frequency, click on the OK button to copy the new interval into the list of working times in the window 'Day profile "<name of the day profile>" - time intervals'.

Changing Intervals

If you wish to change an interval, select the respective interval and click on the **"Change"** button. The window "Edit interval" (see fig. 210) appears.

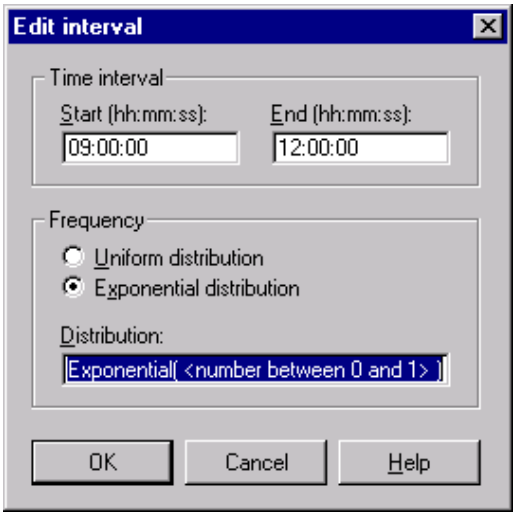


Figure 210: Edit interval

You now have the same options as when adding an interval (see p. 237).

Deleting Intervals

In order to delete an interval, select the interval to be deleted and then click on the **"Delete"** button. The system will then ask you if you really want to delete the respective interval.

Deleting Day Profile

To delete an existing day profile, select the respective day profile in the window "Day profiles" (see p. 236) and then click on the **"Delete"** button.

You will be asked to confirm the deletion of the day profile.

Assigning Day Profiles

Please refer to the information provided for the performer calendar (see p. 232).

Displaying Specified Days

Please refer to the information provided for the performer calendar (see p. 233).

Copying Day Profiles

As well as defining new day profiles for each object of the class "Process start" you can also copy the existing day profiles into other "Process start" objects. Click on the **"Copy"** button in the window "Day profiles" (see fig. 199). The window "Copy day profiles" (see fig. 206) will be displayed.

A day profile is copied according to the three steps described below:

1. Select from the list **"Model"** the business process model that contains the object of the class "Process start" whose day profile you wish to copy. (**Note**: The list only contains models, which are currently open)
2. Select the reference object of the class "Process start" from the list **"Reference object"**, i.e. that object of the class "Process start" whose profile you wish to copy.
3. Select the day profiles to be copied from the list **"Day profiles"**. After you have carried out the first two steps, the day profiles defined for the reference object will be shown in the list "Day profiles".

   Hint: You can only copy those day profiles which have a name different to the day profiles already defined in the Process start to which you are copying (as the day profiles are identified by their names).

Searching for Day Profiles

Please refer to the information provided for the performer calendar (see p. 234).

Calculating Average Number of Processes

The evaluation of the process calendar enables you to calculate the average number of processes during a specified period of time, based on the calendar defined.

When you click the "Query" button in the window "*<process start name>* - Process calendar" (see fig. 198) the window "Calculate average number of processes" (see fig. 211) is displayed.

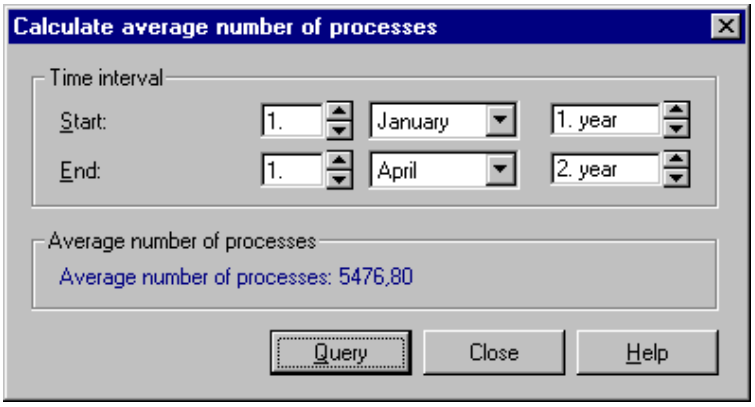


Figure 211: Calculate average number of processes

Define the period for the calculation and then click on the "Query" button. The average number of processes will be displayed in the window **"Average number of processes"**.

2.2.13 Modelling Guidelines for Simulation

Every ADOxx model you want to analyse in the Simulation Component (see chap. 4., p. 332) must meet certain requirements. In order to meet these requirements, ADOxx supports you with modelling guidelines in the following areas:

- **Completeness** (see chap. 2.2.13.1, p. 240)
- **Parallelism** (see chap. 2.2.13.2, p. 241)
- **Times** (see chap. 2.2.12.8, p. 206)
- **Random generator** (see chap. 2.2.12.11, p. 211)
- **Transition conditions and probabilities** (see chap. 2.2.12.12, p. 215)
- **Sub processes** (see chap. 2.2.12.13, p. 221)
- **Performer assignment** (see chap. 2.2.12.14, p. 222)
- **Resource allocation** (see chap. 2.2.12.15, p. 227)
- **Performer calendar** (see chap. 2.2.12.16, p. 228)
- **Process calendar** (see chap. 2.2.12.17, p. 235)

Hint: These guidelines also partly apply to Analytical Evaluation (see chap. 3.4, p. 325), Assessment (see chap. 2.6, p. 303) and Process Cost Analysis (see chap. 5.4, p. 444).

2.2.13.1 Completeness

Before starting their Simulation, business process models must meet the following requirements:

- Each model must **have exactly one start object**.
- Each model must **have at least one end object**.
- The start object and the end object(s) must be **completely and logically linked** by other modelling objects and/or connectors (e.g. activities, decisions etc.).
- The **transition conditions** or **transition probabilities** (see chap. 2.2.12.12, p. 215) after a decision **must be defined correctly**.

There are additional requirements for the Simulation algorithms "Capacity Analysis" (see chap. 4.3, p. 354) and " Workload Analysis" (see chap. 4.4, p. 367):

- A **performer assignment** (see chap. 2.2.12.14, p. 222) must be **defined in every activity**.
- An **application model** (comprising at least one business process model and exactly one working environment model) **must be defined**.

Hint: The provision of **times** (see chap. 2.2.12.8, p. 206), **costs** and **quantities** in the models is **not a precondition** for the Simulation. However, if this data is not entered, the Simulation results may be incomplete.

The completeness check regarding the attribute values assigned can be called using the appropriate query (see chap. 3.1, p. 306). Additionally you can provide the application library with predefined queries (see chap. 3.2, p. 318), using which corporate/project specific queries for completeness could be executed.

The Cardinality check (see chap. 2.4.10, p. 264) will enable you to check the authorised number of objects and connectors within a model.

2.2.13.2 Modelling Parallel Paths

For the modelling of parallel flows within Business Process Models, the following conditions must be met:

1. Each parallelism (branching of simultaneous activities) must be reconnected to the process.

   This condition is met in the example below (see fig. 212), as the parallelism "A" is united in the merging "A" and the parallelism "B" in the merging "B".

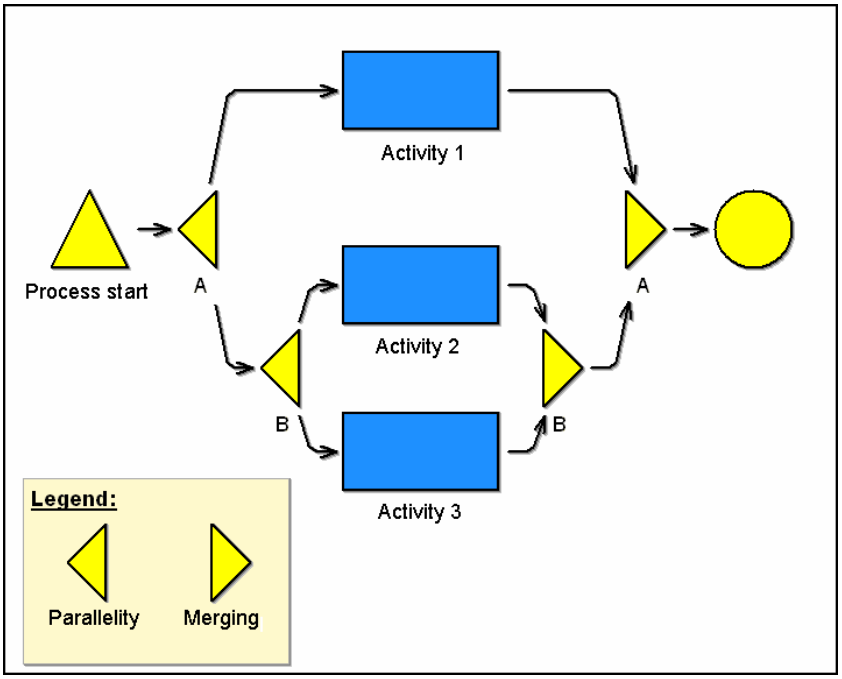


Figure 212: Modelling of parallelism - Merging and nesting

2. Parallel process parts can be nested into one another and an inner parallelism must be merged before it can merge to an external parallelism.

   This condition is met in the example above (see fig. 212), as the inner parallelism (parallelism "B" - "activity 2", "activity 3" - merging "B") is united, before to synchronise the external parallelism "A" in the merging "A".

3. The number of paths starting from a parallelism must equal the number of the logical paths coming into the merging.

   The expression "logical" refers to the flow logic during the Simulation. There is only a logical path after the decision, because there is always only a logical path running out after a decision in a process running.

   This condition is met in the example below (see fig. 213), as the number of the paths starting from the parallelism "A" (two) equals the number of the logical paths coming in the merging "A" (in a process running, either the "activity 2" or the "activity 3" will be carried out after the decision).

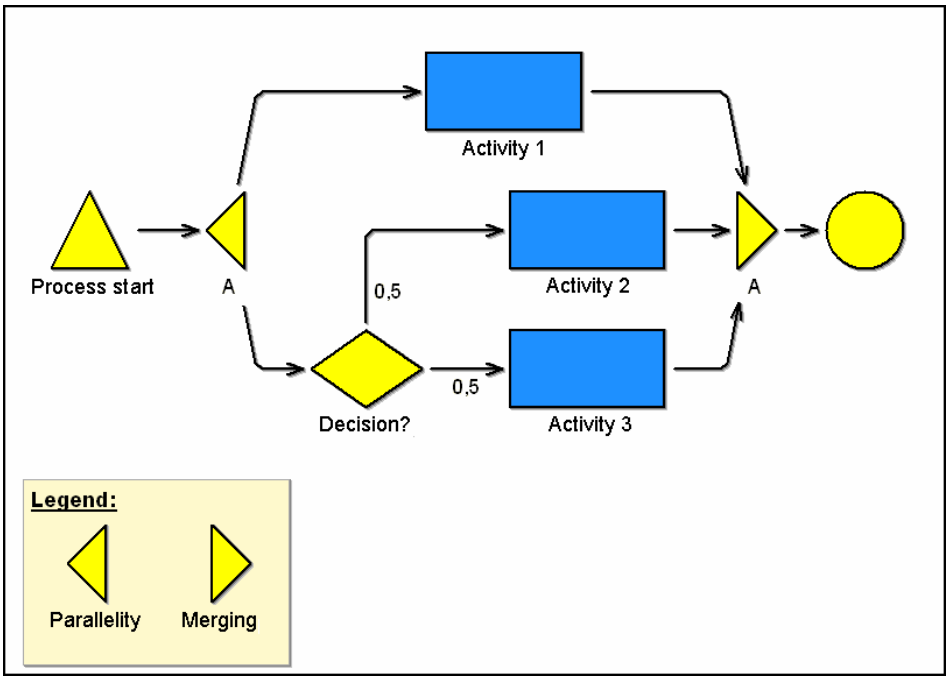


Figure 213: Modelling parallelism - Same paths number

Note: This condition presupposes also that the parallel paths inside the parallelism cannot be united before the merging point (i.e. in the example above, it would not be authorised to model a subsequent connector from the decision to the "activity 1" or from the "activity 1" to the "activity 2").

4. A connection of objects external to the parallelism with objects inside the parallelism is exclusively authorised over the parallelism object .

   This condition is **not** met in the example below (see fig. 214), as the path leads directly from the "activity 4" into the parallelism to "activity 3".

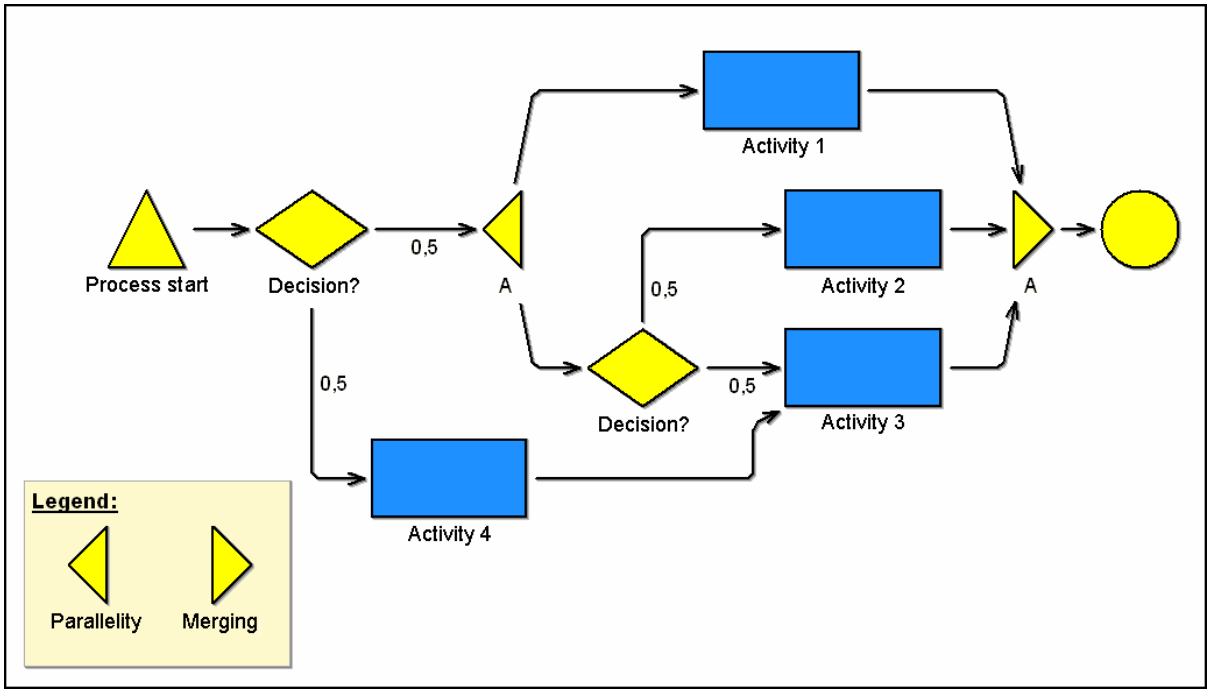


Figure 214: Modelling parallelism - Defined entrance point

5. A connection of objects inside the parallelism with objects outside the parallelism is exclusively authorised over the connecting object.

   This condition is **not** met in the example below (see fig. 214), as a path leads directly from the decision, which is placed within the parallelism, to the process end.

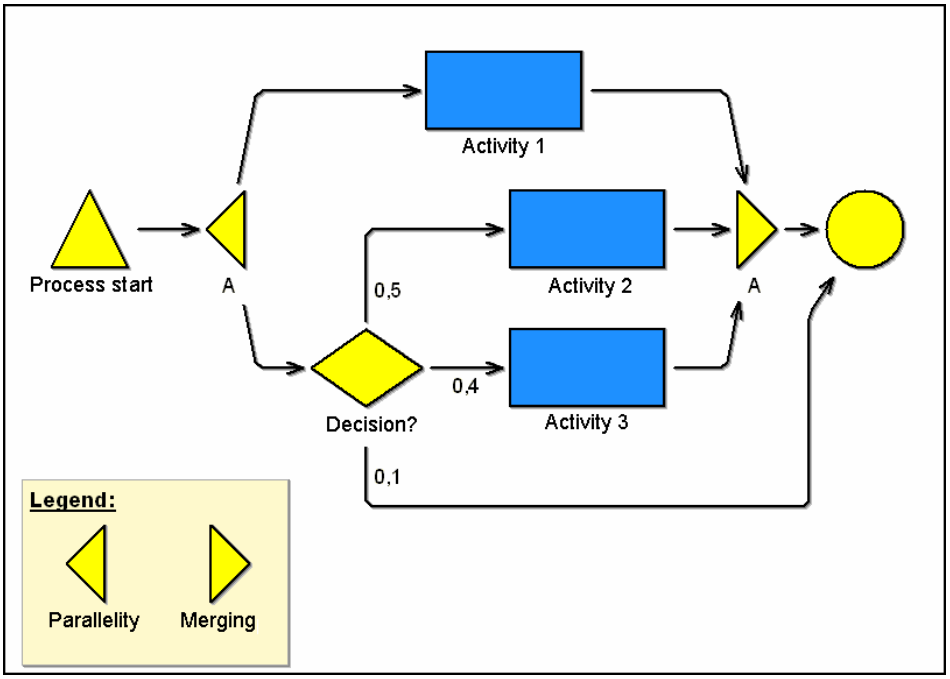


Figure 215: Modelling parallelism - Defined exit point

2.2.14 Global Attribute Change

The function "Global change" allows you to change the attributes of several objects/connectors in one or more models at the same time without having to open each object/connector notebook and carry out the changes individually.

You can change all attributes of an object/connector at the same time and assign them a certain value. The value of an object/connector attribute may be copied to an attribute of the same object. Numerical values (e.g. costs, execution time etc.) can be changed relatively, by percentage or by carrying out mathematical operations (such as sums and averages). For alphanumerical attributes (e.g. descriptions, comments etc.) it is possible to place first, append or replace characters or strings.

Start the function "Global change" by selecting the menu item "Global change" from the "Edit" menu or by clicking on the smart icon in the quick-access bar.

To execute global changes in attributes, follow these steps:

1. **Model selection** (see chap. 2.2.14.1, p. 244)
2. **Object selection** (see chap. 2.2.14.2, p. 244)
3. **Attribute selection** (see chap. 2.2.14.3, p. 249)
4. **Attribute changes** (see chap. 2.2.14.4, p. 250)

Hint: The global change of attribute values is only possible in opened models.

2.2.14.1 Model Selection

The window "Global attribute changes - model selection" holds all models currently loaded (see fig. 216).

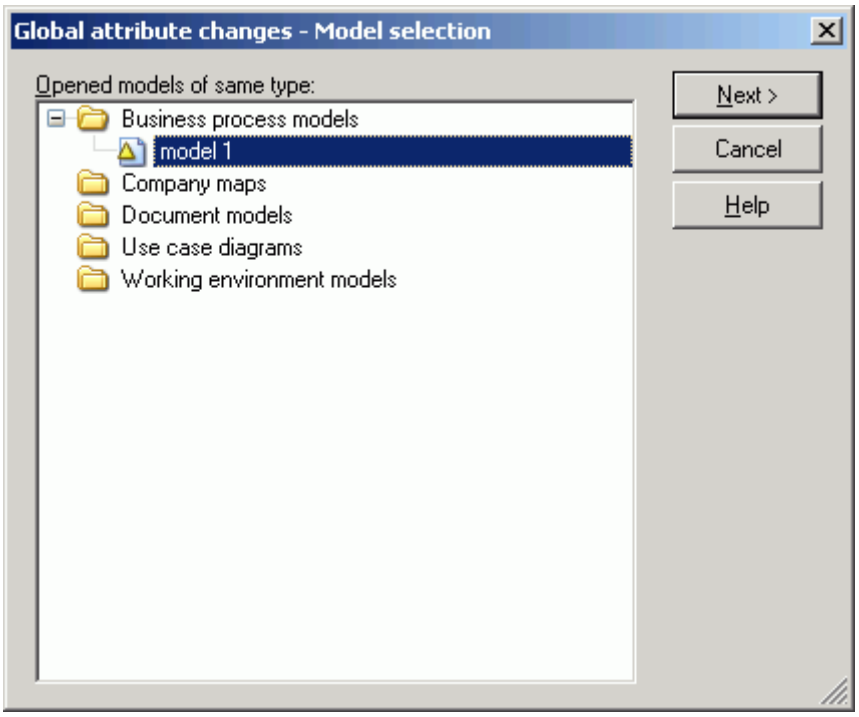


Figure 216: Global attribute changes - Select model

To select the model, which object attributes you wish to change, select the appropriate model name in the model selection list. Now click on the button " Next >" to continue with the object selection (see chap. 2.2.14.2, p. 244).

Hint: For the global changes of attribute values, you can only select models of model types, which are defined in the same library .

2.2.14.2 Object Selection

The window "Global attribute changes - object selection" (see fig. 217) allows you to define a query. The system will then search for all objects of the previously selected models corresponding to the query entered.

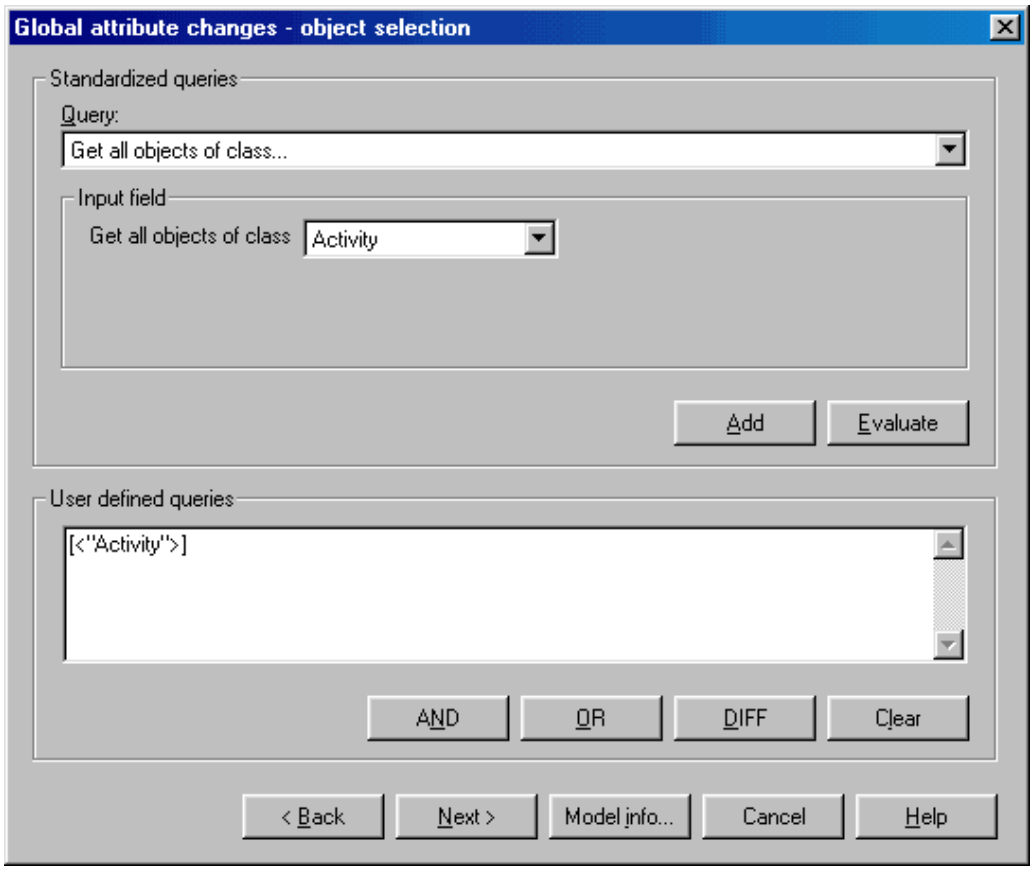


Figure 217: Global attribute changes - Object selection

For the object selection, you can either enter a standardised (see p. 245) or a user defined (see p. 248) query.

As soon as you have selected or entered a query, click on the "Next>" button. The window "Global attribute changes - attribute selection" (see chap. 2.2.14.3, p. 249) is displayed (see fig. 221).

If you click on the "Model info" button, the list of the selected models (see p. 248) will be displayed.

Standardised Query

To carry out standardised queries for the object selection, select one of the standardised queries in the "query" list (see fig. 218). Then an appropriate blank text appears within the input field of the selected query. Enter the necessary information (e.g. select object/attribute name) or select the value you want in the list of the classes, attributes or comparative operators.

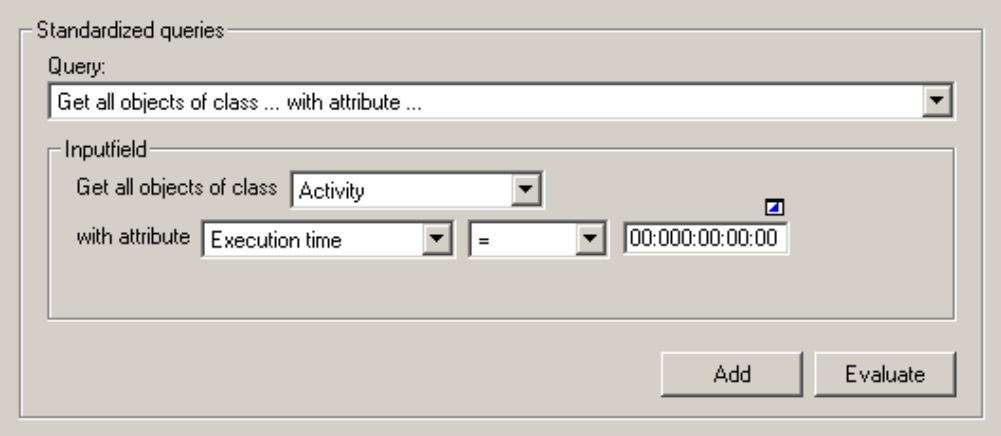


Figure 218: Standardised query (part of the window "Global attribute changes - Object selection")

The standardised query will be explained as following:

- **Get all objects of the class** [*selection*]**.**

  This query displays all objects of the selected class.

  Example: By selecting the class "Activity" you will obtain all activities in the models selected.

- **Get all object of the class** [*selection*] **with attribute** [*selection*][*operator*][*value*]**.**

  This query displays all objects of the selected class, with specified attribute value.

  Note: Select the class first, then the attribute of the class and the operator and then enter the attribute value.

  Example: the class "Activity" with the attribute "Costs", the operator ">" the value "1000" display all objects of the class "Activity" with costs that are greater than 1000.

- **Get all object of the class** [*selection*] **with number of rows in the table attribute** [*selection*][*operator*][*value*]**.**

  This query displays all the object of the selected class, that contain the number of rows (defined by the user) in the table attribute.

  Note: This query will only be displayed if table attributes are defined in the assigned application library.

  Example: By selecting the class "Process start" with the "process responsibility" table attribute, the ">" operator and the input value "2", the system will display all objects of the class "Process start" in the table "process responsibility" with more than two rows.

- **Get all the objects of the class** [*selection*] **with table attribute** [*selection*] **and with the column** [*selection*][*operator*][*value*]**.**

  This query displays all objects of the selected class, that contain the specified attribute value.

  Note: This query will only be displayed if table attributes are defined in the assigned application library.

  Note: This query will only provide the result if the table attribute contains at least one row, and the search condition is fulfilled. For example: if you wish to find all objects, that contain a value (Wildcard "*"), in the column of the table attribute, then the only objects that will be displayed are those which have at least one defined row.

  Example: Selecting the "Process start" class with the table attribute "process responsibility" and the "classification" column, the "=" operator and the input value "on behalf of" will display all the

object of the class "Process start", that contain the value "on behalf of" in the "classification" column of the "process responsibility" class.

- **Get object** [*object name*] **of the class** [*selection*]**.**

  This query allows to verify in what models given objects occur.

  Example: By the instruction of the class "Activity" and the object "edit application form" the system will detect the activity "edit application form" within the previously selected models.

- **Get all the objects, that are connected with the object** [*Object name*] **of the class** [*selection*]
  by the relation [*selection*] **.**

  Using this query you can follow defined relations.

  Note: The name of the object provided by the user must exactly match the name of the object in the model (capital and small letters are distinguished). The input from wildcards is not allowed.

  Example: The selection of the object "find worker" of the class "role" and relation "has role" will display all searched persons related to that role in the selected model.

- **Get all connectors of the class** [*selection*]**.**

  This query displays all the connectors of the selected relation.

  Example: By selecting the relation "subsequent", all the subsequent-connectors in the selected model will be reported.

- **Get all the connectors of the relation** [*selection*] **with**

  attribute [*selection*][*operator*][*value*]**.**

  This query shows all connectors of the relation type selected, which contain the specified attribute value.

  Note: First select the relation, then the attribute of the relation and the operator and finally enter the comparison value of the attribute.

  Example: The query: select the relation "Subsequent" with the attribute "Transition probability", the operator ">" and the input value "0.5", displays all connectors of the type "Subsequent" with transition probability greater than 0.5 (50%).

Hint: Instead of **[*object name*]** the window "Query" shows an input field into which you may enter the name of an existing object.

Instead of **[*value*]** there is a selection field into which you may enter a text.

Instead of **[*number*]** there is a selection field from which you may choose a number .

Instead of **[*selection*]** there is a selection field from which you may select a class, an attribute of a class or a relation.

Instead of **[*operator*]** there is a selection field from which you may choose a number.

Carry out the object selection using the complete standardised query by clicking on the **"Evaluate" button**, to select the attributes (see chap. 2.2.14.3, p. 249) you wish to change.

If you click on the add button **"Add" button**, the standardised query will be converted into an AQL-Expression (see chap. 14., p. 790) and displayed within the **"user-defined queries" field** (see fig. 219). From this query, you can create a (complex) user defined query (see p. 248) by entering further AQL expressions and by adding further standardised queries.

Hint: If an AQL expression is entered in the **"User defined queries" field**, you can carry out this user defined query by clicking on the button "Next >", otherwise the current selected standardised query will be carried out.

User Defined Query

You will carry out the creation of user defined queries for the object selection in the **"User defined queries" field** (see fig. 282), by connecting standardised queries, defining queries in the query language AQL (see chap. 14., p. 790) or extending standardised queries with AQL expression.

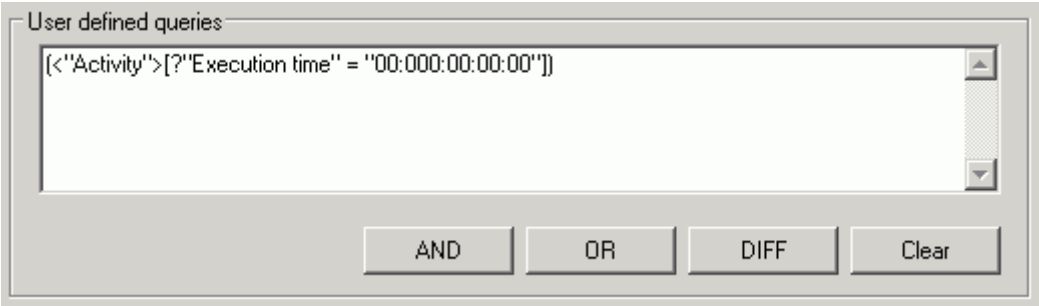


Figure 219: User-defined queries (Part of the window "Global attribute changes- object selection")

As a support for the definition of user defined queries, you can also put **them together from the standardised query**.

For this, define a standardised query and then click on the "Add" button. This way an AQL expression, which corresponds to your standardised query, will be generated. This AQL expression will be entered into the **"User defined query" field**.

By clicking on the **"AND"**, **"OR"** and **"DIFF"** buttons, you will combine or connect queries. By clicking on the **"Clear" button** you will clear the content of the **"User defined query" field**.

Hint: If an AQL expression is entered in the **"User defined queries" field**, you can carry out this user defined query by clicking on the button "Next >", otherwise the current selected standardised query will be carried out.

Click in the window "Global attributes changes - select object" (see fig. 217) on the "Next >" button, to carry out the object selection and to select the attributes (see chap. 2.2.14.3, p. 249) you wish to change.

Model Information

The previously selected models (see chap. 2.2.14.1, p. 244) will be listed in the window "Global changes - Model information" (see fig. 220).

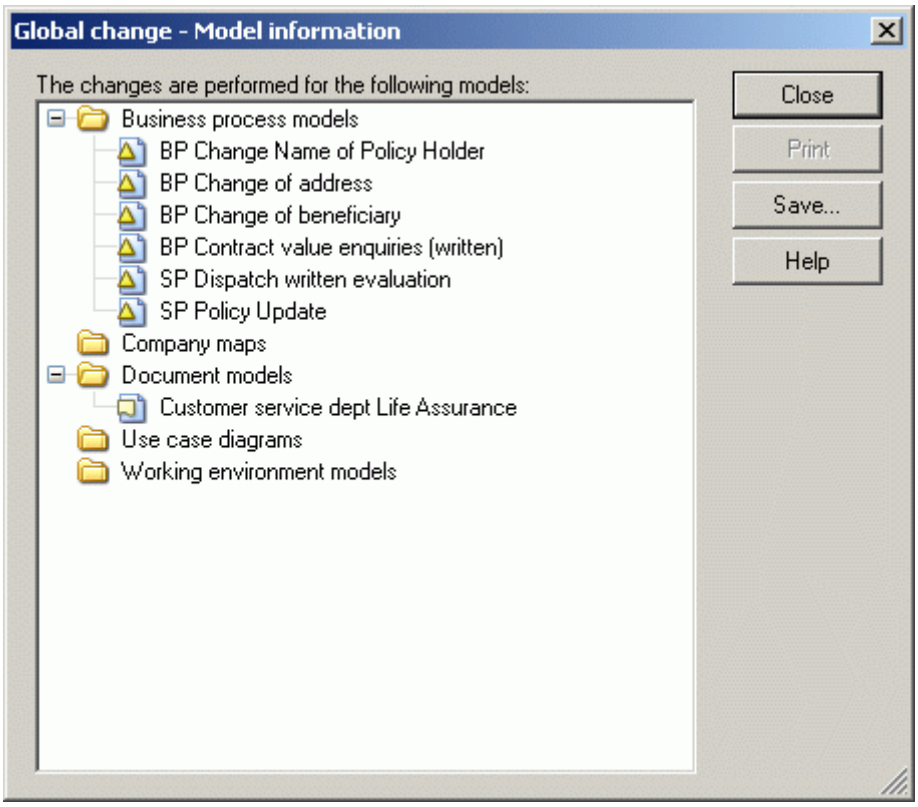


Figure 220: Global change - Model information

You can save the content of this window in a file.

2.2.14.3 Attribute Selection

Select the attributes you wish to change in the window "Global attribute change - attribute selection" (see fig. 221). The attribute values you select here will be changed in all the objects previously selected.

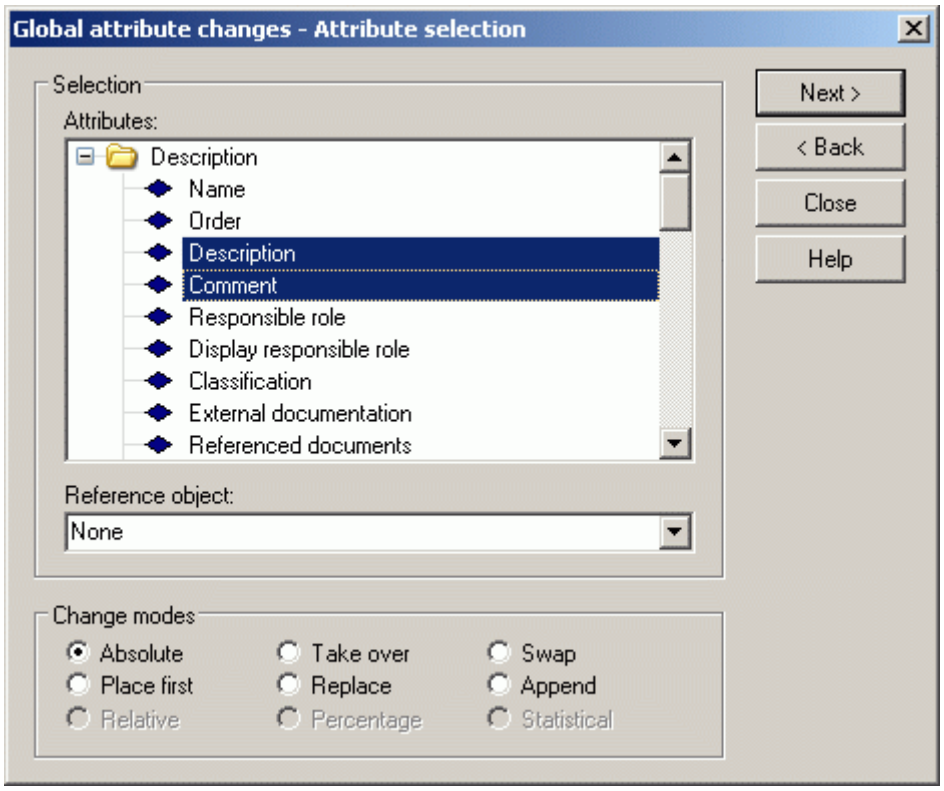


Figure 221: Attribute selection

In the list **"Attributes"** select the attributes you wish to change in the objects found.

The field **"Reference object"** lists all objects found in which the global change will be carried out. A reference object is used when selecting the change mode "Absolute" for the pre assignment of the value of the attribute you wish to change.

After the selection of the attributes to be changed and if necessary of a reference object, you will define the type of change and click on the button "Next >", to change the attributes (see chap. 2.2.14.4, p. 250).

2.2.14.4 Changing Attribute Values

To execute a global attribute change, you must also select one of the change modes displayed in the lower part of the window. The types of change mode available for selection depend on the type of the attributes marked. For certain types of attributes (e.g. number, time format etc.) only some types of change make sense.

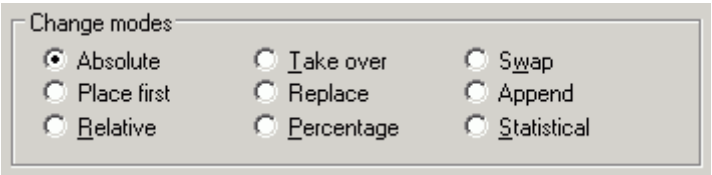


Figure 222: Global attribute change - Change modes

The following **change modes** are available:

- **"Absolute"** (see p. 251)
- **"Take over"** (see p. 251)

- **"Swap"** (see p. 251)
- **"Place first"** (see p. 252)
- **"Replace"** (see p. 252)
- **"Append"** (see p. 253)
- **"Relative"** (see p. 253)
- **"Percentage"** (see p. 254)
- **"Statistical"** (see p. 254)

Hint: If none of the change modes are available, you have probably selected attributes, which due to their type(s) cannot be changed in one process.

Continue the global attribute by clicking on the "Change" button. Depending on the change mode selected an appropriate window will be displayed, in which you can carry out the changes.

After the change has been carried out, an information window will appear. It contains all attributes whose attribute values have been changed.

Absolute

The change mode "Absolute" is possible for **all attribute types except the attribute "Name"**.

The change mode "Absolute" means that you enter new attribute values with which you want to replace all existing attribute values. You may choose an object from the list **"Reference object"**. The selected object's attribute values will then serve as reference values for the changes of the other objects' attribute values and will be listed in the notebook "Changing attribute values - absolute", which opens when clicking on the "Change" button.

All previously selected attributes are shown in this notebook. If you have selected a reference object, its current attribute values will be displayed. If no reference object has been selected, then the attributes of this notebook will be blank. You can now either enter new attribute values or use the reference object's attribute values for the global change.

Hint: If you do not enter an attribute value in the notebook, global changing (clicking on the "Change" button) causes the respective attribute values in the objects selected to be deleted or (in case of numbers and time attributes) to be set to 0 or 00:000:00:00:00 respectively.

Take Over

The change mode "Take over" **is possible for all attributes except the attribute "Name"**. It is only applicable, however, to exactly two attributes of the same type.

The change mode "Take over" means that the attribute value of an object is copied to another attribute of the same object. A prerequisite for copying is that exactly two attributes have been selected from the list and that these attributes are of the same type. The direction of the copy ("from attribute x to attribute y") is determined by one of the two options in the window **"Changing attribute values - take over"**.

Swap

Swapping attribute values **is possible for all attributes except the attribute "Name"**. It is only applicable, however, to two attributes of the same type.

The change mode "Swap" means that the values of two attributes are swapped. A prerequisite for this exchange is that only two attributes have been selected from the list and that these two attributes are of the same type.

Clicking on the "Change" button executes the swap. As soon as it is completed, a window listing the attributes whose values have been swapped appears.

Close the window by clicking on the OK button or by pressing the enter key.

Place First

It is possible to append attribute values to the front of **all alphanumerical attributes** (e.g. description, comment etc.).

The change mode "Place first" inserts a character or string at the beginning of the current attribute value in all attributes selected. Enter the value which is to pre cede the attributes selected into the field **"String to place first"** in the window **"Changes of attribute values - place first"** and click on the button **"Change"**.

Replace

Replacing attribute values is possible for **all alphanumerical attributes and for distribution attributes** (e.g. description, comment or value).

The change mode "Replace" replaces existing characters or strings in the selected attributes.

Enter the text you wish to be replaced into the field **"Old string"** and the new text into the field **"New string"** in the window **"Changes of attribute values - replace"** and click on the **"Change"** button. The text will be replaced in all objects, which contain exactly the "Old string".

> **Hint:** A special input window (see p. 252) is available to replace program call attributes.

Replace (Program Call)

In the selected program call attributes, the change mode "replace" replaces the program to carry out and/or the entered parameters.

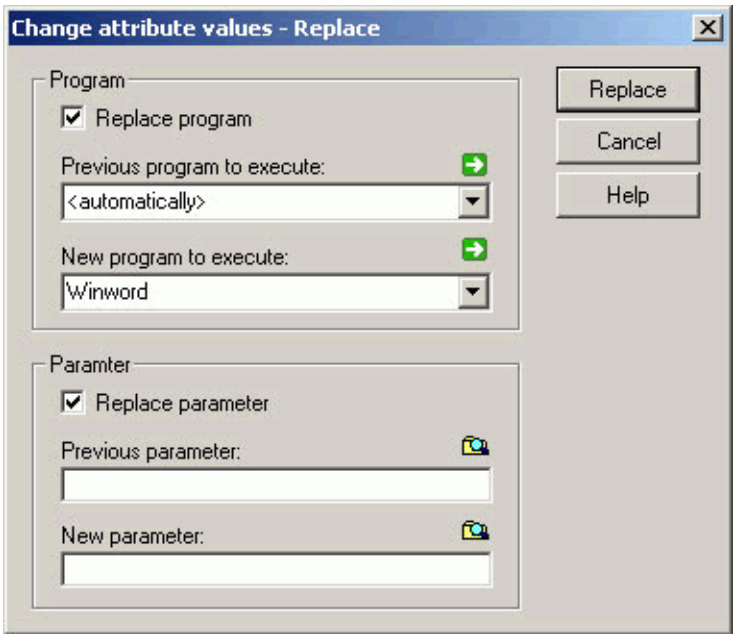


Figure 223: Replace program call attribute values

In the **"Change attribute values - replace "** window (see fig. 223) you can activate the option **"Program replace"**, to change the program to carry on, or the option **"replace parameters"**, to change the indicate parameter.

Select in the **"Old program to carry out "** list the program to be changed and the new program in the **"New program to carry out"**.

Enter in the **"Old parameter"** field the parameter to be replaced and the new parameter in the **"New parameter"** field.

Click on the **"Change"** button. The program to be carried out and/or the entered parameter will be replaced in all the objects which fit to the selection or the instruction.

Append

Attribute values can be appended to **all alphanumerical attributes** (e.g. description, comment etc.).

The change mode "Append" enables you to add characters or strings to the attributes selected.

Enter the text to be appended to the previously selected attributes into the field **"String to append"** in the window **"Change of attribute values - append"**.

Relative

The relative change of attributes is possible for **attributes in numeric or time format** (e.g. wages per hour, execution time etc.).

The change mode "Relative" causes the values of the attributes marked to change by one numerical value positively or negatively.

Once you have clicked on the OK button, the **"Change attribute values - relative"** window will appear, in which you will set the arithmetical operator (**"Operator"** field) and the operand (**"Value"** field).

The following operators are available for the relative attribute change:

+ Addition

- Subtraction

Note: the substraction will be indicated within the "Value" field, i.e. each value which is subtracted from the value in the previously selected attribute.

***** Multiplication

Note: The multiplication will be indicated within the "Value" field, i.e. each value, which will be multiplied by the value in the previously selected attribute (=multiplicand).

/ Division

Note: The divisor will be indicated within the "Value" field, i.e. each value, by which the value in the previously selected attribute will be divided (=dividend).

Hint: When changing attribute values in time format (e.g. execution time), only the operators "addition" and "subtraction" are available. Enter the operand in ADOxx time format into the **"Value"** field, you can find a support to this input by clicking the "dialogue" icon .

Percentage

The percentage change of attributes is possible for **attributes in numeric or time format** (e.g. wages per hour, execution time etc.).

The change mode "Percentage" causes the values of the attributes selected to be changed by a certain percentage.

Clicking on the "Change" button opens the window **"Changes of attributes - percentage"**, in which you determine if the existing attribute values are to be increased or decreased (field **"Operator"** with the options "**+**" for increase and "**-**" for decrease). You must also enter the percentage by which the attribute values should be changed into the field **"Percent"**.

Statistical

The statistical change of attributes is possible for **attributes in numeric or time format** (e.g. wages per hour, execution time etc.).

The change mode "Statistical" causes the values of the attributes in a selected object to be copied to a target attribute of the same object after a statistical operation has been carried out.

Once you have clicked the OK button the **"Change attribute values - Statistical"** window will appear, in which you will define the change mode, the input attributes and the target attribute.

Both statistical operations "**Sum**" and "**Average**" are available as change mode options.

"**Sum"** means that the values of a number of attributes are summed up and inserted in a target attribute.

"Average" also sums up the values of a number of attributes, but then divides the sum by the number of attributes summed up and thus calculates the average. The average value is then assigned to a target attribute.

To select the target attribute, double-click on the appropriate attribute in the list **"Input attribute"** or select the attribute from the list **"Take over <change mode> to"** and then click on the **"Change"** button.

2.3 Model Navigation

The model navigation provides support while working on several models, possibly connected by references.

2.3.1 Navigating through open models

This function enables you to navigate through several model windows. The model windows opened are ordered according to the sequence in which they were opened. The navigation function enables you to bring the previous or next model window to the foreground of the working area.

There are various ways to navigate through the model windows:

- Select the option "**Back**" in the menu "Window", to display the model window that was **prior** to the actual one or choose the options **"Previous"** or **"Next"** to display the previous or next model window respectively in the foreground. These options are available under the Windows menu.
- Press the key combination <Alt>+<Left arrow key> to display the **previous** model window or the combination <Alt>+<Right arrow key> to display the **next** model window.
- Click on the smart icon or in the quick-access bar.

The function "Activate sub model" (see chap. 2.3.3, p. 255) allows you to open sub models referenced in a model (or to bring them to the foreground if already open). Calling a sub model causes the models to be re-ordered in such a way that the model called will be placed immediately behind the model, which is referencing it.

The models are also re-ordered when a model is opened (see chap. 2.4.2, p. 257) or brought to the foreground via the ADOxx explorer (see chap. 4., p. 54). The model opened or brought to the foreground is placed immediately behind the model previously active.

2.3.2 Activate model window

If you activate a model window, this window will be displayed in the foreground on the working area.

To activate a model window:

- Select in the "window" menu the menu item with the model name of the window you wish to activate.
- Click on the smart icons , or and keep the mouse button pressed, until a pop-up menu with the name of the opened model is displayed and then select from this menu the model name of the window you wish to activate.
- Select the model in the ADOxx Explorer.

Hint: There is a special method to activate model windows known as activating a referenced model (see chap. 2.3.3, p. 255) from a (main) model.

2.3.3 Activate Referenced Model

The activation of sub models allows you to open referenced models quickly (or activate their model windows).

To activate a sub model

- **click** on the **visualised object reference** (e.g. name of the sub model or referenced object), which is displayed as a hyperlink (see fig. 224),

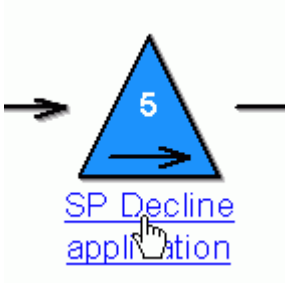


Figure 224: Hyperlink display (Example "subprocess")

- **Double click, while keeping the control key (<Ctrl>) pressed on the object**, from which the reference is starting, in the drawing area (Function "Ctrl+double click").
- **Select in the context menu of the object the "activate submodel" menu item.**
- **Select the registered reference in the ADOxx Notebook and click on the icon** .

Hint: The hyperlink function (see fig. 224) is only possible for non selected objects.

Hint: The "Activate sub model" menu item in the context menu of an object as well as the function "Ctrl + double click" is only available, if a "model pointer" as been defined for the class of this object in the application library.

If the sub model is already opened, only the corresponding model window will be activated, i.e. displayedc in the foreground.

If you "Ctrl + double-click" on an object holding no reference, an appropriate message will be displayed and a window will open up, in which you can create an inter-model reference.

When activating a sub model, the models opened will also be re-ordered within the navigation function (see chap. 2.3.1, p. 255): The sub model will be placed immediately behind the model, which is referring it.

2.4 Model Management

Model management is an umbrella term for the model administration and control functions available in ADOxx.

The model management functionality can be found under the Model menu within the Modelling Component.

2.4.1 Create New Model

If you wish to create a new model, select the menu item "New" from the "Model" menu. Alternatively you can use the key combination <ctrl>+<n> or the appropriate smart icon in the quick-access bar.

The window "Create new model" (see fig. 225), in which you can select the model type, the model group, the model name and (optionally) the version number of the model to be created, will be displayed.

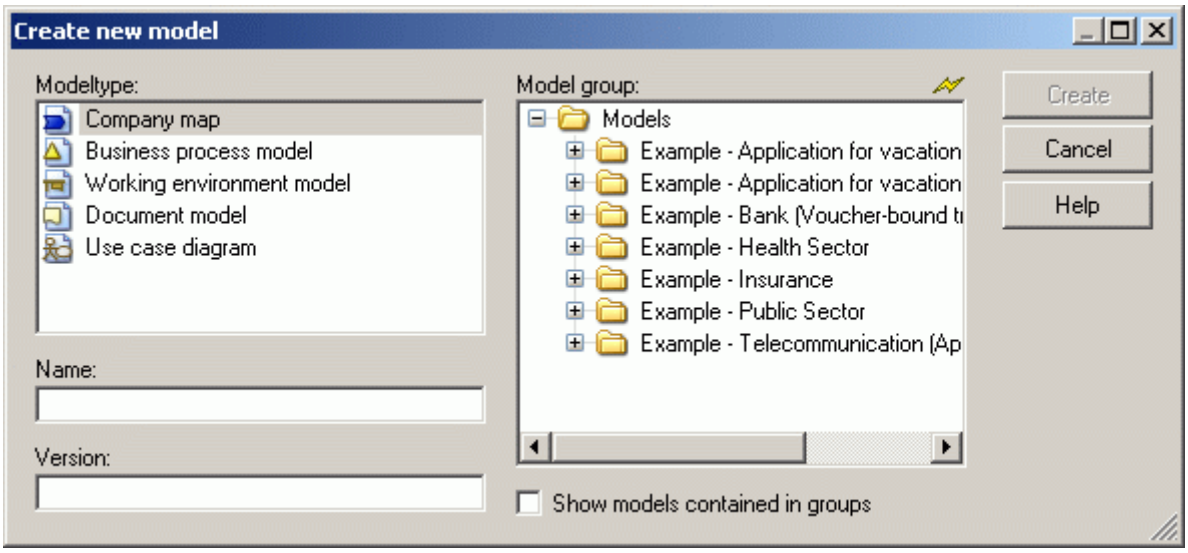


Figure 225: Create new model

Hint: Depending on the definition of the application library, this dialogue may additionally contain a model type filter (see chap. 3.1.1.1, p. 42).

Select in the **"Model type" field** the type of the new model and in **"Model group" field** the model group, in which you wish to create the new model.

Then enter the name of the new model in **"Name" field** and if necessary, for model-related versioning, a version number in the **"Version" field**.

Hint: The model name must contain at most 220 characters. The version number for the model-related versioning (see chap. 10.1.1, p. 92) must contain at most 20 characters.

Hint: As in the "Create a new model" dialogue window, you can expand or collapse models in the model groups, by clicking on the appropriate check box.

In time-related versioning, fields for the validity period of the model will be displayed instead of the "Version" field (e.g. "Month" and "Year").

Hint: There must be no model of the same name, type and version number already existing within the ADOxx database. If such a model already exists an appropriate error message will be displayed when you try to create an identically named model.

In the model group list, you can have a look at the models contained in the model groups, for this, activate the **"Show models contained in groups" option**.

Confirm your entries by pressing the button "Create" which will cause the current window to be closed and the new empty model to be displayed. If the button "Create" is faded out in grey, this means that at least one of the three necessary entries "Model type", "Model group" or "Name" has not been selected.

2.4.2 Open Existing Model

If you wish to open a model, which was previously saved within the ADOxx database, then select the menu item "Open" from within the "Model" menu. Alternatively, use the key combination <ctrl>+<o>or the smart icon from the quick-access bar.

This causes the "Model open" (see fig. 226) window, which contains a model selection list to be opened. Within the model group hierarchy all models which have been saved and to which you have access are displayed.

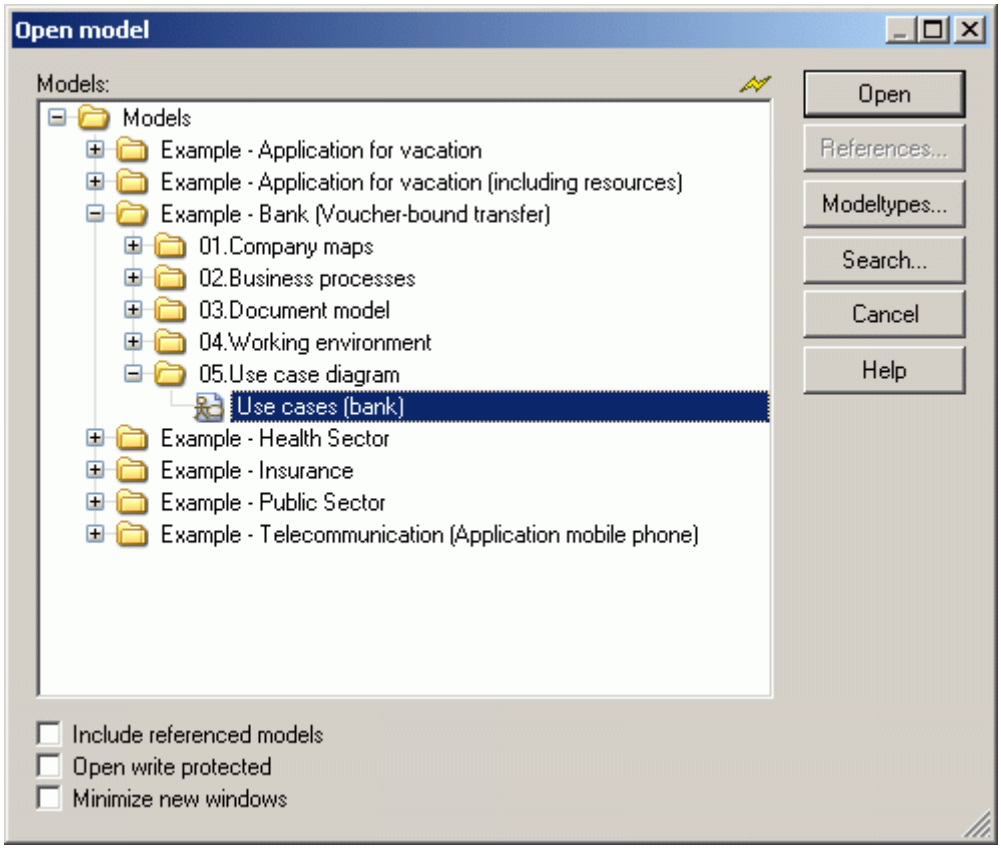


Figure 226: Open models

Select one or more models to be opened and click on the button **"Open"**.

By clicking the **"References"** button, you can fix the References - settings (see chap. 11.1, p. 97) to open models by activating the **option "Include referenced models"**.

By clicking on the button **"Model types"** you can restrict the display of models in the model selection list to certain model types (see chap. 3.1.1, p. 42).

By selecting the button **"Search"** you can search (see chap. 3.1.3, p. 43) the model list for specific models. All models, which match the search are displayed in the "Found entries" window. Selecting one or more of these will then cause them to be highlighted within the "Open model" (see fig. 226) window. Clicking on the "Open" button can then open them.

Clicking on the **"Refresh**" button causes the model selection list to be refreshed.

The following options can also be set within the "Open model" (see fig. 226) window:

- "**Include referenced models**": the subprocesses of each model selected will also be opened. This works recursively.
- "**Open write protected**": the selected models are only opened for reading purposes, i.e. any changes you make to these models will not be saved.
- "**Minimize new windows**": the model window will be minimised in the working area.

Hint: If you attempt to open a model which has already been locked by another ADOxx user, a window will be shown asking you if you wish to open this model with read-only access.

Hint: Models with read-only access will be displayed with a padlock (e.g.) represented in the model type icon.

2.4.3 Close Model

If you wish to close a model, activate the corresponding model window or the icon corresponding to this model window and select from the "Model" menu the menu item "Close". You can also close the model window (see chap. 2.2.3, p. 130) by clicking on the exit button ( ), by opening the window menu (by a single click on the exit button) and then selecting the option "Close" or by the key combination <Ctrl>+<F4>.

If there are unsaved changes in the model, the exit button of the model window will be marked with a red star and a "*" will be appended to the window name. This will be displayed in the model window by a "Change" in the status line. If you close a changed model, without saving it, an appropriate indication window will be displayed.

Now you now have the possibility either to save and close the model (click the "Yes" button), close the model without saving the changes (click the "No" button) or to cancel the previously selected close action (click the "Cancel" button).

Hint: By selecting the menu item "Close all" in the "Window" menu all model windows will be closed.

Hint: If you close a **new empty Model, which has not been saved**, an appropriate indication window will be displayed and you will have the possibility to delete this model or to store the empty model within your ADOxx database.

2.4.4 Save Model

If the contents of a model have changed and these changes have not yet been saved, the exit-button of the model window is marked with a red star and a star is added to the name of the window. The text "modified" also appears in the status row referring to the active model.

In order to save these changes in the ADOxx database, you can select the option "Save" from the "Model" menu. Alternatively use the key combination ***<ctrl>+<s>*** or the smart icon from the quick-access bar.

If you wish to save all changed models within the ADOxx database select the menu item "Save all" from the "Model" menu. It is also possible to save models from within the "ADOxx-Explorer (see chap. 4., p. 54) window.

Models are saved automatically (see chap. 2.4.4.1, p. 260) after a specified number of changes (standard:20) have been made. It is possible to turn this option on or off and to change the number after which an automatic save should be carried out by using the menu item "Auto-save" in the "Extras" menu.

Hint: If you would like to save the models to a floppy disk, then you must use the "Import/Export" (see chap. 6., p. 445) component and export (see chap. 6.2, p. 468) the models to an ADL file.

Hint: If you change only view parameters (e.g. zoom value, view mode), the model will not be considered as "changed", however you can save this new view in the ADOxx database.

2.4.4.1 Auto-save

Using the **Auto-save** function, the changes will be automatically and individually saved by the program.

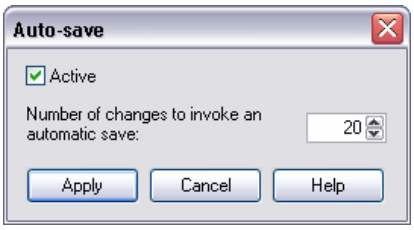


Figure 227: Auto-save

You can switch this function on by activating the "Active" option within the "Auto-save" window (see fig. 227), i.e. the auto-save comes after the entered number of changes. By entering a value you will fix the number of actions, which must be carried out before saving automatically.

If you deactivate the option "active", there will be no auto-save.

Once you have entered the value, click on the "Apply" button to save this setting.

> **Hint:** These settings will be stored in your user profile and will remain valid only for the current ADOxx user.

2.4.5 Save Model Under a Different Name

If - by **model-related versioning** (see chap. 10.1.1, p. 92) - you wish to save an open model with a different name or with a different version number, without having to overwrite the current model, select in the "Model" menu the "save as" menu item.

By **time-related versioning** (see chap. 10.1.2, p. 92) select in the "model" menu the "save as" menu item and the submenu "new model", to save the open model with a different name. To save the open model with a different version number, select the submenu "New Version" (see chap. 2.4.6, p. 262).

> **Hint:** Whether the model will be versioned model-related or time-related depends on the configuration of your application library.

The "Save model as" window appears:

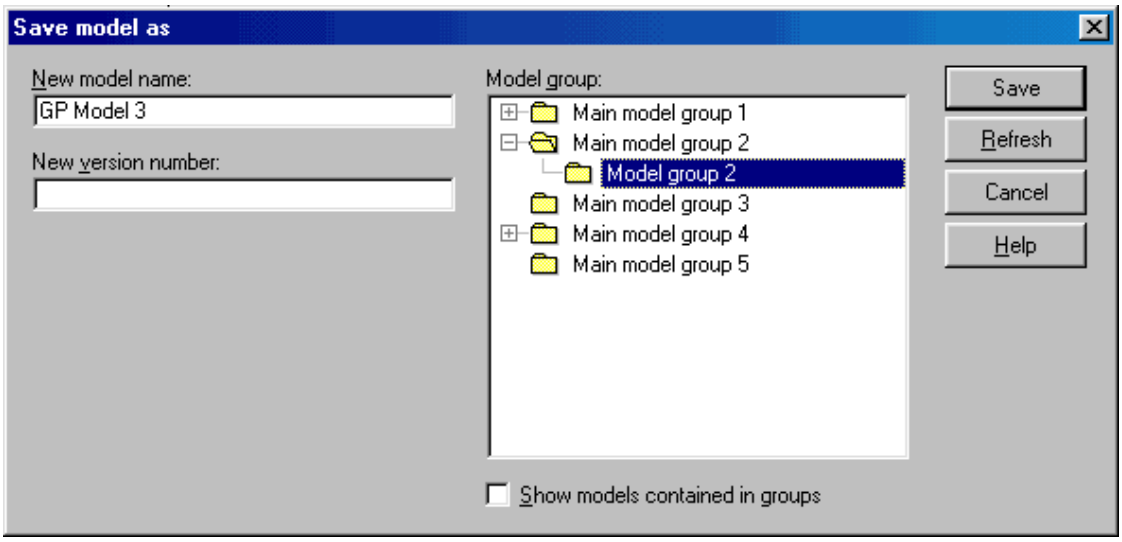


Figure 228: Save model with a new name

The current name and the current version number of the model are provided. Now you can change target model group, name and version number. As soon as you click **"Save"** the model is stored in the ADOxx database with the new parameters. If the model contains unsaved changes, you are also asked whether you want to save these changes in the original model too.

Hint: The model name must contain at most 220 characters. The version number for the model-related versioning (see chap. 10.1.1, p. 92) must contain at most 20 characters.

Hint: As in the "Create a new model" dialogue window, you can expand or collapse models in the model groups, by clicking on the appropriate check box.

By **time-related versioning** (see chap. 10.1.2, p. 92), fields for the validity period of the model (e.g. "Month" and "Year") will be displayed instead of the "version" field.

Hint: There must be no model of the same name, type and version number already existing within the ADOxx database.

Hint: The old model is no longer open - it still exists in the ADOxx database and can be re-opened as normal.

Hint: After saving a read-only model under another name, the new model is opened with read-write access NOT as read-only.

For referred on the model references, additional dialogue appears, that might look like:

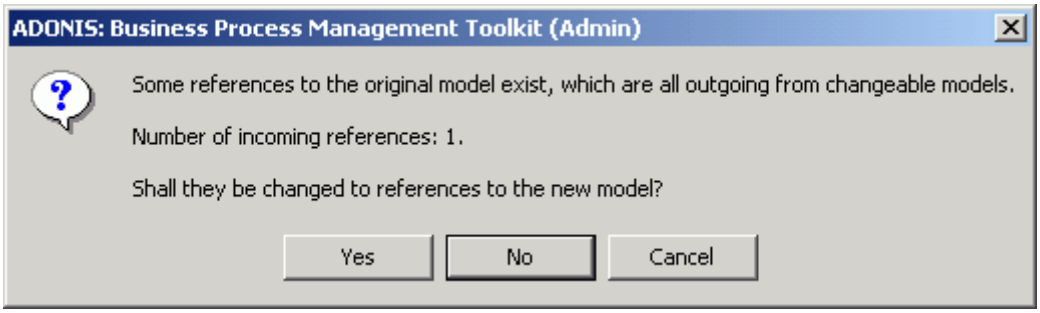


Figure 229: Saving model under the new name

It is checked whether the user has write access to any models referencing the active model, and also verified whether the number of references that lead to the model is displayed. After the verification, the program asks if the references should be re-routed to the new model.

After the user decision and the origin of references, the following options are available:

- The user clicks "no": All references still refer to the original model.
- All references come from models with write access and the decision is "yes": The references will be moved. All the references refer now to the new model.
- The references come partly from alterable, partly from write protected models and the decision is "yes": All the alterable references refer to the new model. References from write protected models, as before refer to the original one.
- All outgoing references come from the models, that are not write protected: It is not possible to make any selection. All references refer to the original model.

Hint: It is possible, to limit above possibilities, in case of client specific configuration in the ADOxx.

2.4.6 Save Model as New Version

For **time-related versioning** (see chap. 10.1.2, p. 92) fix in the "Save model as new version" the new version number, under which the model must be saved in the ADOxx database.

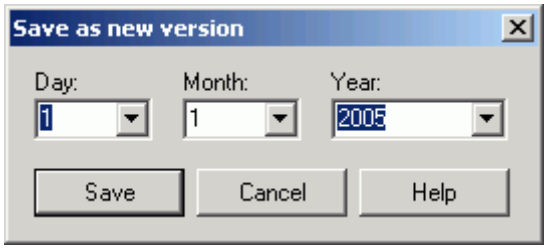


Figure 230: Save model as new version

Hint: There must be no model of the same type in the ADOxx database, which version number corresponds with your entry.

Hint: If for model-related versioning (see chap. 10.1.1, p. 92) you want to save an opened model with a different version number, select in the "Model" menu the "Save as" (see chap. 2.4.5, p. 260) menu item.

2.4.7 Export Table

Hint: The "Export table" menu item is only available, if the current model is displayed as a table.

When exporting a table, the contents of the tabular model display (see chap. 2.2.10, p. 181) of a class or relation are saved to a file (see chap. 8.13, p. 81) in the specified format (TXT, RTF, HTML).

2.4.8 Rename Model

By **model-related versioning** (see chap. 10.1.1, p. 92) -, if you want to give a model a different name and/or a different version number, call the "Rename" function from the "Model" menu.

The dialogue window "Rename model" (see fig. 231) appears, showing the current model name and version number.

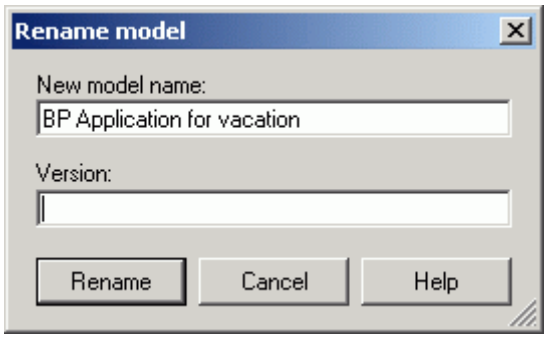


Figure 231: Rename model

Enter a new model name and/or a new version number.

Hint: The new model name and the new version number must be unique, i.e. there must be no model already existing with the same name or version number in the ADOxx database.

Hint: The model name must contain at most 220 characters and the version number at most 20 characters.

Click on the button "Rename" to execute the changes and to close the dialogue window. Clicking on the "Cancel" button closes the window without executing any changes.

Hint: When a model is renamed, any inter-model references to and from this model are automatically adjusted. In some cases it may be necessary to update these references (see chap. 2.4.11.1, p. 265).

ATTENTION: In contrast to the function "Save as" the model saved under the old name (or old version number) will **not** be kept when renamed.

For **time-related versioning** (see chap. 10.1.2, p. 92) you can rename a model by:

- Assigning a new version number (see chap. 2.4.8.1, p. 263) or
- Assigning a new model name (see chap. 2.4.8.2, p. 264).

To rename this model select in the "Model" menu the "Rename" menu item and the "Version" submenu or the "New model name" submenu.

2.4.8.1 Version Number

The version number of the current model is displayed in the "Rename version" window (see fig. 232).

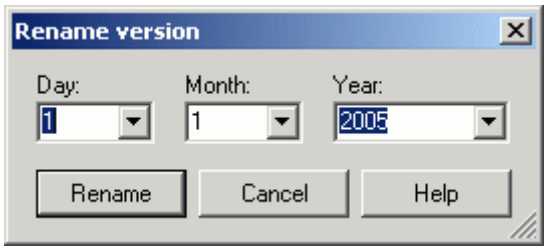


Figure 232: Rename version

Select the version number, which should be assigned to the model and then click on the "Rename" button.

Hint: The new version number must be unique, i.e. there must be no model already existing with the same version number in the ADOxx database.

2.4.8.2 Model Name

The name of the current model is displayed in the "Rename model" window (see fig. 233).

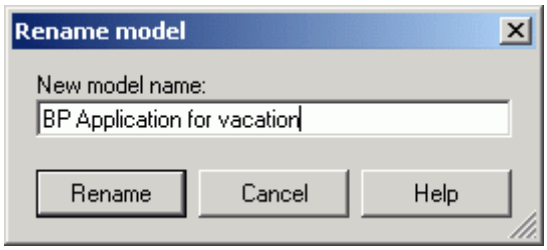


Figure 233: Rename model

Enter a new name for the model and then click on the "Rename" button.

Hint: The new model name must be unique, i.e. there must be no model already existing with the same name in the ADOxx database.

Hint: When you rename models with time-related versioning, all models of the same name (but with a different version number) will also be renamed.

2.4.9 Model Attributes

Each model has a notebook containing attributes describing the model itself. These attributes are identical for all model types. You can look at the model attributes by activating the appropriate model window and selecting the menu item "Model attributes" from the "Model" menu. Alternatively, you may also call this function via the drawing area's context menu.

After selecting the menu option, the ADOxx Notebook appears, that displays the model attributes in the predefined chapter structure.

The model attributes can be changed by the user, if they are not write protected (like i.e. systems attributes, that are automatically maintained by ADOxx .

Hint: Write-protected model attributes are by default shown in grey.

Click on the "Close" button to close the notebook and to save the entries. Clicking on the "Undo" button will delete all changes made since the notebook has been opened.

2.4.10 Check Cardinalities

The cardinalities in ADOxx indicate:

- How many objects of a class are contained in a model.
- How many connectors of a relation are outgoing from an object.
- How many connectors of a relation are incoming into an object.

The field of cardinalities (number of objects, number of connectors) can be limited in the **application library** (cardinality rules), in order to, for instance, be able to check the modelling lines (max. 12 activities in a process model) or the Simulation condition (exactly an outgoing subsequent connector of an activity).

ATTENTION: During the modelling, the compliance with the cardinality rules will **not** be automatically checked.

Check the cardinality of a model, by selecting the "Check cardinalities" menu item in the "Model" menu.

If the cardinality rules are not compliant, an appropriate error message will be displayed. You can abort the check and correct the errors displayed or continue the check.

Once you have successfully checked the cardinalities, i.e. if all cardinality rules are met in the model checked, an appropriate message will be displayed.

2.4.11 Inter-model References

There are two types of inter-model references:

- References **to a model**.
- References **to an object**.

Inter-model references are created (see chap. 2.2.12.9, p. 207) in an attribute of each start object and can be managed via the respective ADOxx Notebook of that start object. Such attributes are called INTERREF attributes (see chap. 7.12, p. 736).

Inter-model references are updated automatically when you rename the referenced models or objects.

To display inter-model references to an active model select between:

- Outgoing references (see chap. 2.4.11.1, p. 265) and
- Incoming references (see chap. 2.4.11.2, p. 267).

2.4.11.1 Outgoing References

You can display all outgoing references of objects of a model in a list. Select the option "Inter-references" from the "Model" menu and call the function "Outgoing from model" from the sub menu.

The window "Outgoing inter-references" (see fig. 234) appears. A folder exists for each INTERREF attribute that occurs in the model. This folder is labelled with the attribute name and the respective class name. All existing references outgoing from the model are listed in the appropriate folder.

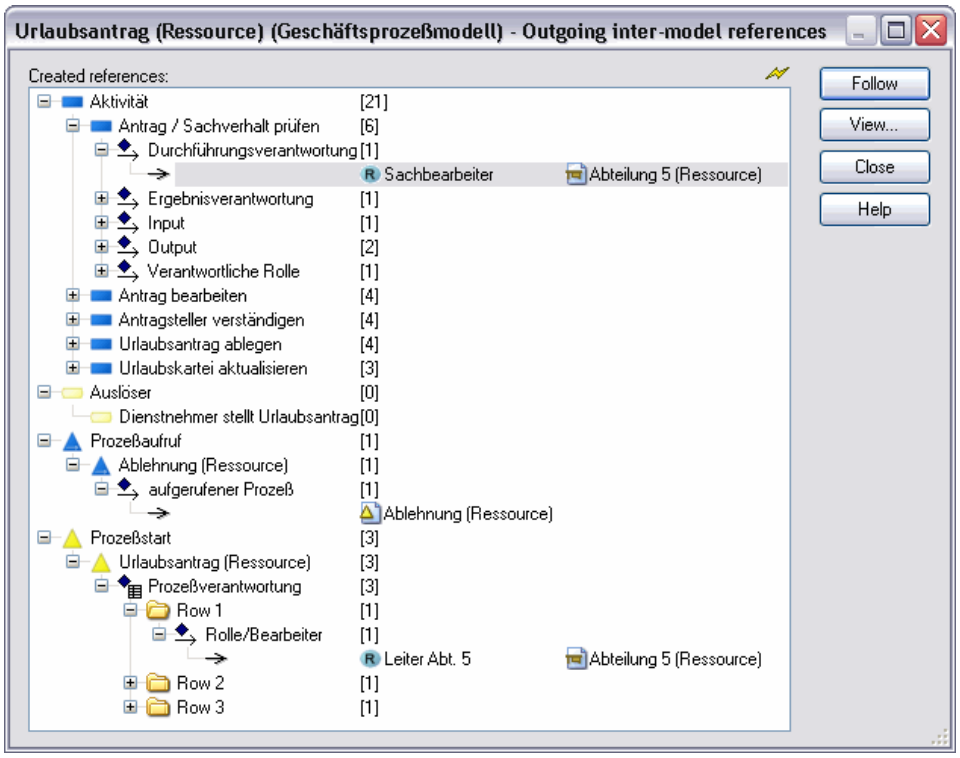


Figure 234: Outgoing references

The following information will be displayed for each existing **Reference to a model**:

- Class name of the starting object with the numer of references in brackets (**class**)
- Name of the starting object with the numer of references in brackets (**object**)
- Name of the reference attribute with the numer of references in brackets (**attribute**) *or*

  if reference attribute is part of a record attribute:
  - Name of the record attribute with the numer of references in brackets
  - Row within the record table with the numer of references in brackets
  - Name of the reference attribute with the numer of references in brackets (**attribute**)
- Reference status and name of target model

For each existing **on the object reference** the following information will be displayed:

- Class name of the starting object with the numer of references in brackets (**class**)
- Name of the starting object with the numer of references in brackets (**object**)
- Name of the reference attribute with the numer of references in brackets (**attribute**) *or*

  if reference attribute is part of a record attribute:
  - Name of the record attribute with the numer of references in brackets
  - Row within the record table with the numer of references in brackets
  - Name of the reference attribute with the numer of references in brackets (**attribute**)
- Reference status and name of target model and target object

The Reference status indicates whether it is a valid inter-model reference (→).

The default hierarchy (class - object - attribute) can be changed clicking on the **"View"** button.

An inter-model reference is invalid (if its target model does not exist in the database or if the target model does not exist in the model, i.e. if the target model has been deleted after the creation of the reference) or when using time-related versioning the target model has a more recent date than the source (start) model.

If models have been renamed or deleted since you logged on, click on the **"Refresh"** button to update the contents of the list.

In order to navigate to one of the models referenced, select the appropriate entry in the list and click on the **"Follow"** button or double-click on the respective entry. The model referenced is loaded and its model window activated. If the reference is to an object, the specific object will be highlighted.

2.4.11.2 Incoming References

You can also display all references incoming to the model or to the object in the model. There are two possibilities:

- Select in the **"model"** menu the **"inter-references"** menu item and then the option **"Incoming to model".**
- Select the option "Model references" in the **context menu** of the drawing area

The window "Incoming inter-references" (see fig. 235) is displayed, showing a list of all start objects of incoming references.

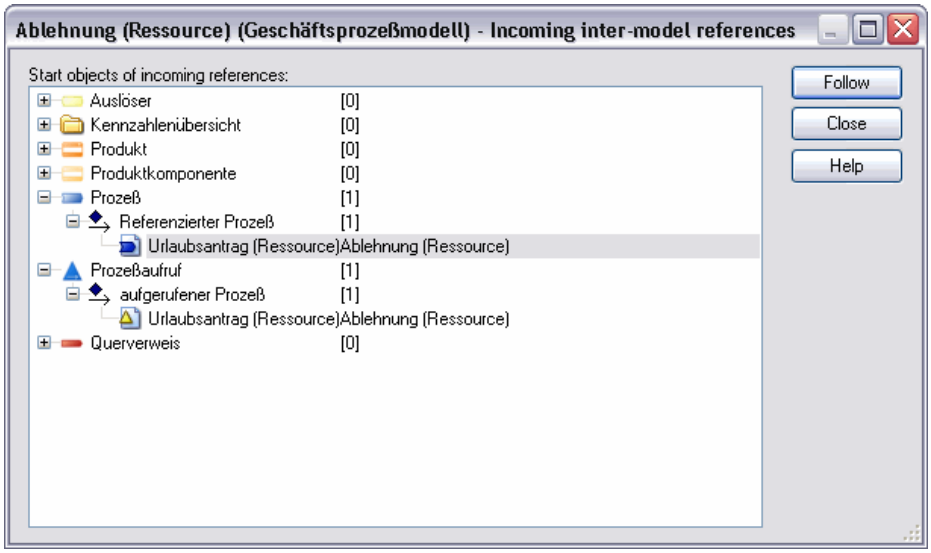


Figure 235: Incoming references

A folder exists for each reference attribute which can receive the references to the current model type. This folder is labelled with the class name and the number of references in brackets. In each class folder the name of the reference attributes with the number of references will be listed as a sub folder. All existing references incoming to the model or to the objects in the model will be displayed in the appropriate link.

The following information will be displayed for each **reference to the active model**:

- Name of the model with the start object.
- Name of the start object.

The following information will be displayed for each **reference to an object in the active model**:

- Name of the model with the start object.
- Name of the start object.
- →.
- Name of the target object.

If you wish to navigate to the start model, select the appropriate entry in the list and click on the **"Follow"** button or simply double-click on the entry. The start model is loaded (if necessary), its model window activated and the start object is highlighted.

As this list is always structured correctly due to the way inter-references are saved in the database, this window does not hold a "Refresh" button. This window will not show invalid references.

Reference to the Active Object

You can display the start objects of references incoming to an object by selecting in the context menu of the object (click on the right mouse button) the "select object reference" menu item.

All objects with a reference to the current object will be listed in the "<object name> - incoming inter-references" (see fig. 236) window. The objects are grouped according to the name of the attribute, in which the reference is entered.

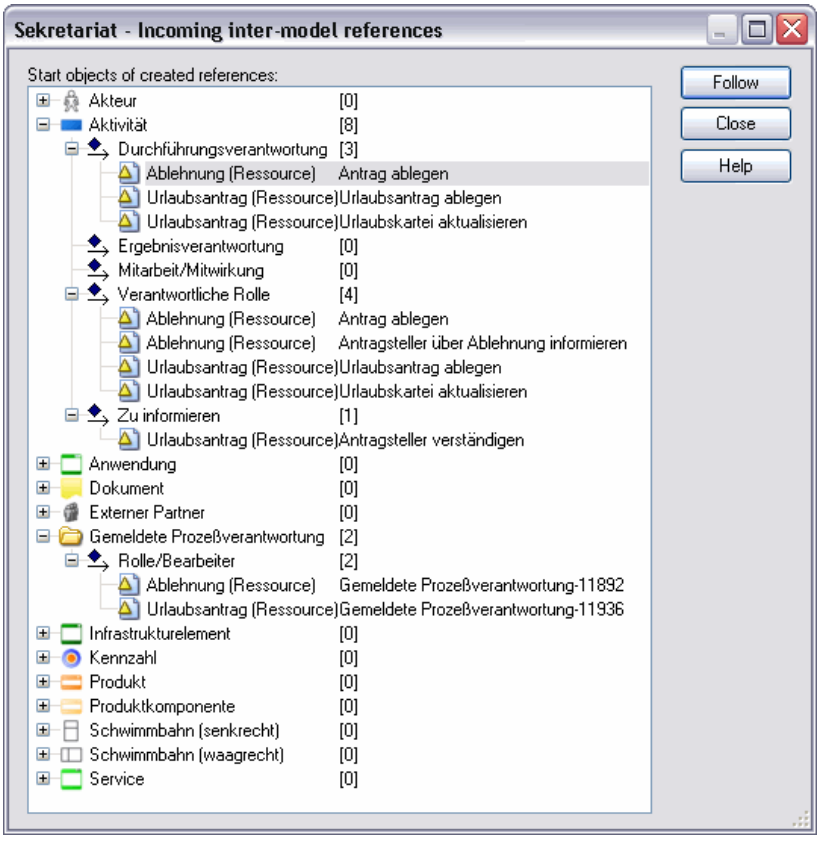


Figure 236: Start object of the incoming reference to the active object

If you select the object and then click on the **"Follow" button**, the model will be displayed with this object.

2.4.12 Managing Model Groups

The model group hierarchy is displayed in the window "model groups list" (see fig. 237).

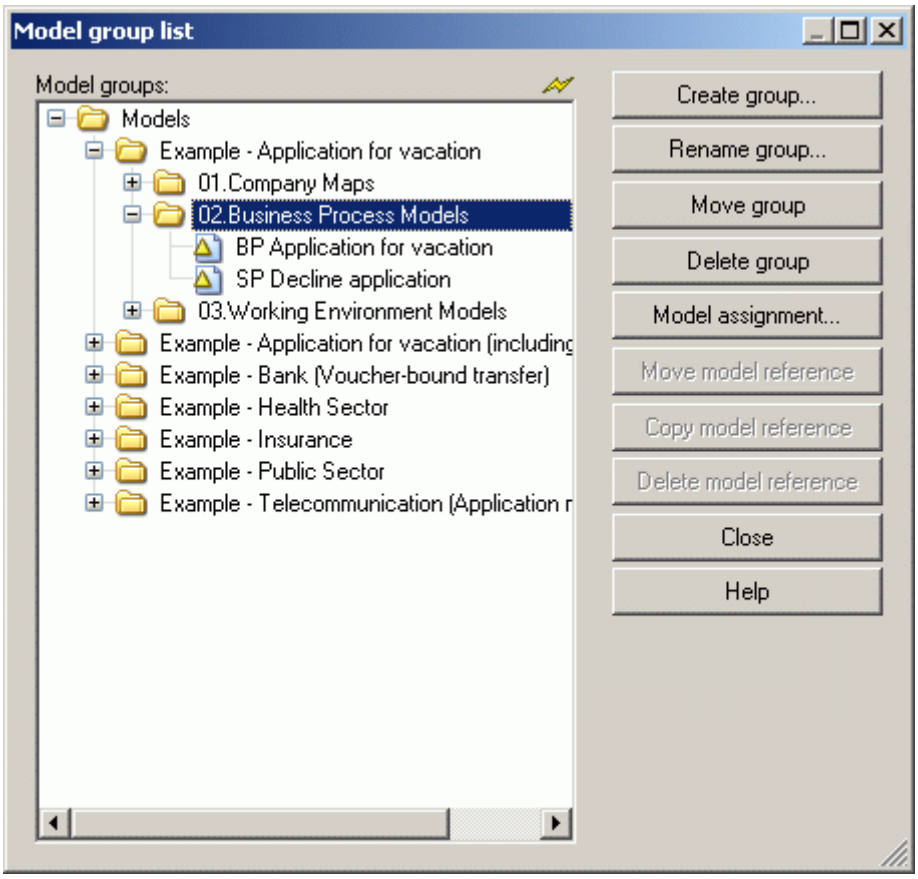


Figure 237: Model groups list

Hint: Model references to locked models (i.e. to models, which are opened by a user with a write access) cannot be edited and thus are displayed in the model group structure in grey font.

The list "Model groups" contains all existing model groups.

You can select the following functions by clicking on the appropriate button:

"Create group"	To create (see chap. 2.4.12.1, p. 270) a new model group.
"Rename group"	To rename (see chap. 2.4.12.2, p. 270) an existing model group.
"Move group"	To move (see chap. 2.4.12.3, p. 271) existing model groups.
"Delete group"	To delete (see chap. 2.4.12.4, p. 271) existing model groups.
"Model assignment"	To assign models (see chap. 2.4.12.5, p. 271) to a model group;
"Move model reference"	To move (see chap. 2.4.12.6, p. 272) model references from one model group to another.
"Copy model reference"	To copy (see chap. 2.4.12.7, p. 273) model references from one model group into anther.
"Delete model reference"	To delete (see chap. 2.4.12.8, p. 273) model references from a model group.

"Refresh" To update the model group list (database update).

"Close" To close the window.

When you open the context menu (right mouse button), the general functions (see chap. 3.1, p. 41), the above listed functions of the model group list as well as the additional functions mentioned below are available:

"Model types" for the model view of specific model types (see chap. 3.1.1, p. 42).

Hint: The availability of the functions depends on whether or not you have previously selected either a model group or a model.

2.4.12.1 Create Model Group

Model groups can be created at every hierarchy level, i.e. you can create a new model group within any model group.

In order to create a new model group, select the model group in which the new model group should be contained and then click on the button "Create group".

The window "Create model group" appears (see fig. 238).

Figure 238: Create model group

Enter the name of the new model group and click on the OK button. The window is closed and the updated model group hierarchy is displayed.

ATTENTION: The name of the new model group must be unique on the level of the main groups and within each model group.

2.4.12.2 Rename Model Group

If you wish to rename an existing model group, select the appropriate model group and click "Rename group". The window "Rename model group" (see fig. 239) appears, in which the current name (field "Old model group name") is shown. The field "New model group name" also holds the current name and you can adjust or over-write this with the new name you require.

Figure 239: Rename model group

Enter the new name of the model group and "OK". The window is closed and the updated model group hierarchy is displayed.

ATTENTION: The name of the new model group must be unique on the level of the main groups and within each model group.

2.4.12.3 Move Model Group

To move a model group, select the appropriate model group and then click on the button "Move group". As soon as you move the cursor over the list of the model hierarchy its shape will change to .

Now click on the model group in which you would like to move the model group previously selected.

If you want to move a model group to the highest hierarchy level (to the main group), you must click on the area below model hierarchy, or alternatively on the left, next to the main models' groups.

Hint: The name of the model group to be moved must be unique within the new model group and within the main group.

Hint: A model group can only be moved when you have write access for this model group und the models within this model group are not opened by another user with write access (the locked models will be shown in grey font).

Before the group is moved you will be asked to confirm this action.

2.4.12.4 Delete model group

Hint: Model groups can be deleted only if they contain either other model groups or models.

Select one or more model groups which you want to delete and click "Delete group".

Before you finally delete the model group, an appropriate security message will be shown.

2.4.12.5 Assign Models

By assigning models to a model group, references to the models stored in the **ADOxx** database are created. The models assigned to a model group can be loaded with read-only or read-write access depending on the access authorisation of the user who attempts to open them.

If you wish to assign models stored in the **ADOxx** database to a model group, select the appropriate model group and click on the button "Model assignment". The window "<*model group name*> - Model assignment" (see fig. 240) is displayed, showing all models saved in the **ADOxx** database, organised according to model types.

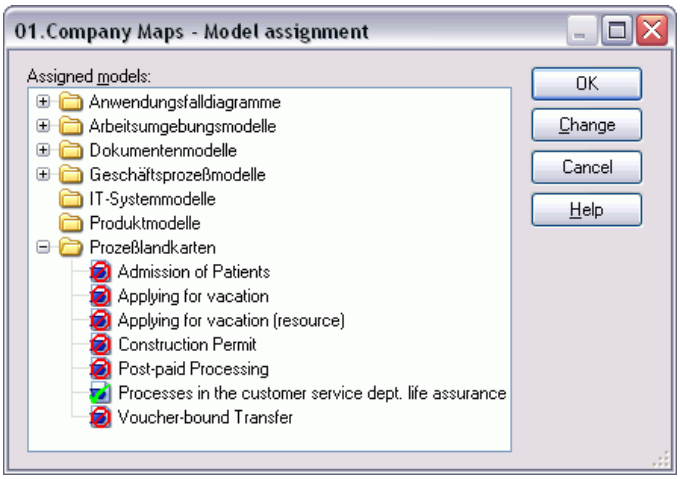


Figure 240: Model assignment

The icon before each model indicates the current status of the model with regard to the model group you have currently selected.

means "not assigned",

means "assigned" and

for "varying assignments".

Change the assignment by double-clicking on the respective model or by single clicking on the "access" or "no access" button.

> **Hint:** If you want to delete the last existing model reference from a model group, an appropriate message will be displayed, as this means that this model will no longer be assigned to any model group within the **ADOxx** database and thus cannot be accessed by any **ADOxx** users. However, unassigned models can be re-assigned by the **ADOxx** administrator.

After changing the model assignment as required (by double-clicking on the appropriate models to change the assignment mode or by selecting one or more models and then clicking the button "Change mode"), close the window by clicking on the OK button. The updated model hierarchy will be displayed.

2.4.12.6 Move Model Reference

If you wish to move a model reference to another model group, select the respective model reference and click on the button "Move model reference". As soon as you place the cursor over the list of the model hierarchy its shape changes to: .

Now click on the model group to which you would like to move the previously selected model reference.

> **Hint:** A model group may only contain one model reference to a model stored in the **ADOxx** database. You cannot move a model reference into a model group in which such a reference already exists.

The system will ask you to confirm this action before the model reference is actually moved.

2.4.12.7 Copy Model Reference

If you wish to copy a model reference to another model group, select the appropriate model reference and click on the button "Copy model reference". As soon as you place the cursor over the model hierarchy list its shape will change to .

Now click on the model group into which you want to copy the previously selected model reference.

> **Hint:** A model group may only contain one model reference to a model stored in the **ADOxx** database. You cannot copy a model reference into a model group in which such a reference already exists.

The system will ask you to confirm the action before the model reference is actually copied.

2.4.12.8 Delete Model Reference

To delete a model reference from a model group, select the model reference and then click on the button "Delete model reference".

The model reference will be deleted after a security message.

> **Hint:** It is not possible to delete the last model reference from a model group as this model will not be assigned to any model group of the application library and therefore cannot be accessed by any ADOxx user. Should you try to delete the last model reference in a model group a corresponding message will be shown

2.4.13 Model Comparison

With the option "Model comparison" two models of the same type may be compared with respect to the objects and connectors they contain. The criteria for comparison are the names of objects or the name of the start and target objects of connectors. Different attribute values will be assigned to identical objects or connectors.

Start the comparison of models by clicking on the option "Model comparison" in the "Model" menu. The window "Model comparison - model selection (see chap. 2.4.13.1, p. 273)" opens up, in which you can select the models to be compared.

The results of a model comparison (see chap. 2.4.13.2, p. 274) are displayed **in tabular form** in the ADOxx browser (see chap. 8., p. 66). You can copy the results into the clipboard, save them to a file (see chap. 8.13, p. 81) or print them out (see chap. 8.15, p. 84).

2.4.13.1 Model Selection

Select the models to be compared in the window "Model comparison - model selection" (see fig. 241).

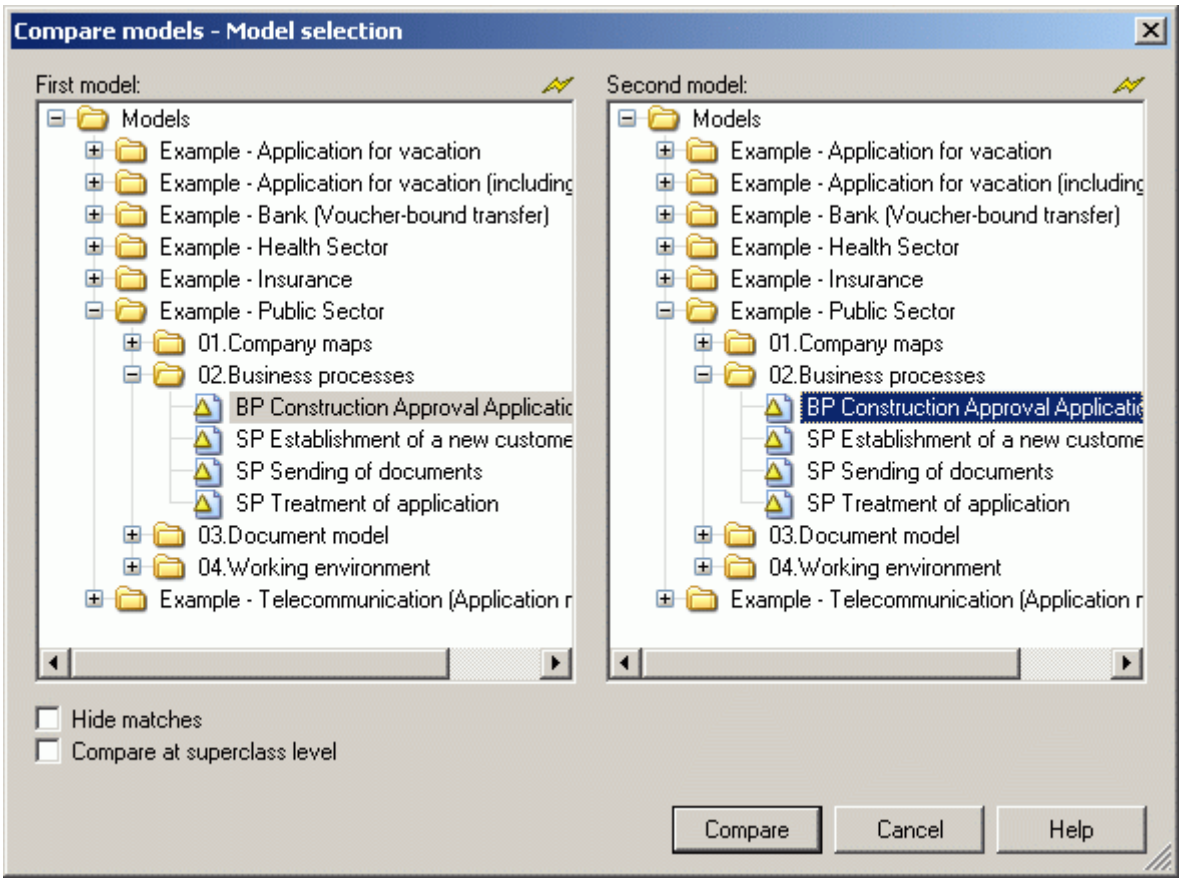


Figure 241: model comparison - model selection

ATTENTION: The two models to be compared must be of the same type.

If the option **"Comparison at super class level"** is not active, the comparison is carried out on all classes that can be instantiated ("reference classes") in the library. The objects or connectors derived from a certain reference class will then be compared.

If the library, in which the models to be compared are defined, has an appropriate hierarchy of classes, it is also possible to compare them at super class level. The classes of reference are then only the uppermost classes of the hierarchy in the library. The set of objects or connectors that is to be compared to this class is made up of those objects or connectors, which are derived from any class of the based on reference class.

If you wish to only display the differences between the models, activate the option **"Hide matches"**. Objects and connectors which appear in both models and whose attribute values do not differ will then not be shown in the result.

Start the comparison by clicking on the OK button. The models will be loaded and the results of the model comparison (see chap. 2.4.13.2, p. 274) will be displayed.

2.4.13.2 Results

Should both models be totally identical (same objects and connectors and equal attribute values), an appropriate message will be displayed.

Otherwise the result of the comparison is displayed in tabular form in the ADOxx browser (see chap. 8., p. 66).

The ADOxx browser (see chap. 8., p. 66) displays the results in the following categories (columns) - divided into columns:

- **In both models**.
- **Different values**.
- **Only in model <first model name.>.**
- **Only in model <second model name>.**

The column "In both models" lists those objects and connectors, which occur in both models (provided that the option **"Hide matches"** is deactivated). Objects are regarded as identical, if they have the same name and belong to the same class. Two connectors are regarded as identical, if the objects they connect have the same name and belong to the same class.

The column "Different values" marks by "**xxx**" all those objects and connectors, which have different attribute values.

The column "Only in model <first model name>" contains all objects and connectors that only occur in the model <first model name>.

The column "Only in model <second model name>" contains all objects and connectors that only occur in the model <second model name>.

If the result contains both objects and connectors, a blank line divides them.

It is possible to view the differing attribute values of those objects and connectors which are listed in the column "In both models" marked by "**xxx**".

Double-click on an object or connector with differing attributes in order to actually view the attributes, which are different. Alternatively, you can list these attributes by double-clicking on the "xxx". The differing attribute values are displayed in the **ADOxx** browser. The differences are shown in the columns "Value in model <first model name>" and "Value in model <second model name>".

If an attribute value is too long to be fully displayed, clicking on its value will call a window in which the value will be displayed in full.

Close the window by clicking on the OK button or by pressing the enter key.

Show Differences in Values

The differences in attribute values are displayed in the columns "Value in model < first model name> " and "Value in model < second model name>".

You can copy the differences to the clipboard, save them to a file or print them out. Clicking on the OK button closes the window.

2.4.14 Search Models

There are three functions available for searching models. Select the menu item "Search models" from the "Model" menu.

- "By model attributes" (see chap. 2.4.14.1, p. 276).
- "Process hierarchy (tabular)" (see chap. 2.4.14.2, p. 279)
- "Process hierarchy (graphical)" (see chap. 2.4.14.5, p. 281).

When **searching for models based on model attributes** you can search in the **ADOxx** database for all those models which meet the criteria you define for a certain model attribute.

The **tabular process hierarchy** enables you to search in the **ADOxx** database for processes meeting one of six pre-defined criteria. The models found are shown in the ADOxx browser (see chap. 8., p. 66). Within the process hierarchy you may only search **for business process models**.

The **graphical process hierarchy** allows the graphical display of a process hierarchy.

Hint: The availability of the "Process hierarchy (graphically)" depends on the definition of your application library.

2.4.14.1 Model Search by Model Attributes

The model search based on model attributes can either be carried out according to standardised or user-defined queries (similarly to the queries of the Analysis Component).

If you wish to search for a model by means of a standardised query, select one of the standardised queries from the list **"Query"** in the window "Model search" (see fig. 242).

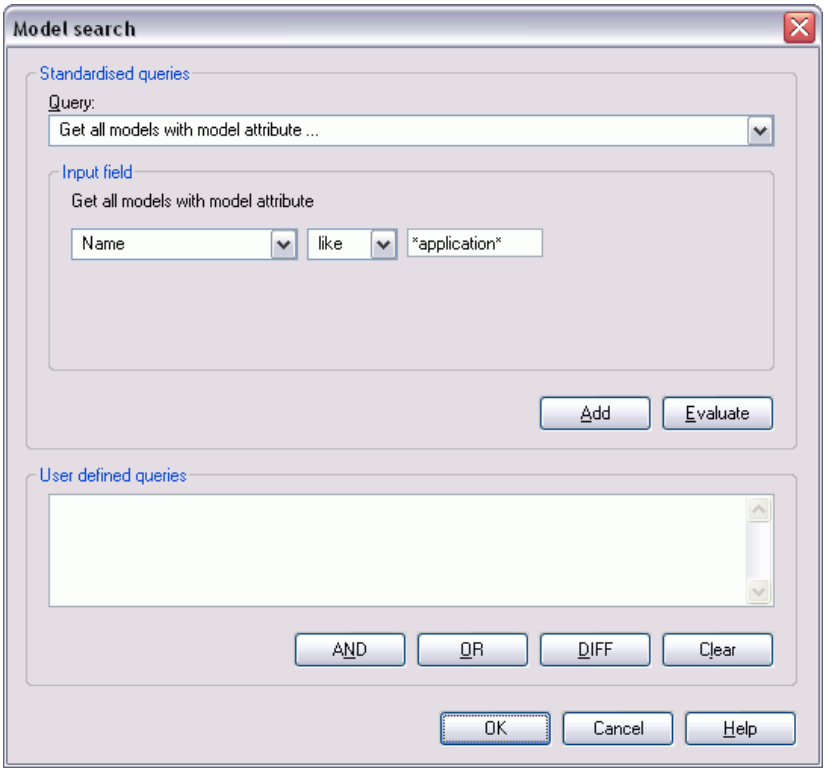


Figure 242: Model search

Enter the necessary information or select the values required from the list of modeltypes, attributes or operators of comparison.

The following pre-defined queries are available:

- **`Get all models.`**

  This query will display all models saved to the database.

- **`Get all models of type`** [*`selection`*]**`.`**

  This query will display all models of the selected type.

- **Get all models with**

  model attribute [*selection*][*operator of comparison*][*input*]**.**

  This query will display all models whose model attributes meet the criteria defined.

- **Get all models of type** [*selection*] **with**

  model attribute [*selection*][*operator of comparison*][*input*]**.**

  When using this query, you will find all models of the specified type which have a specific model attribute.

  Note: First enter the modeltype, then the model attribute, the operator of comparison and then enter the attribute's value to be compared with.

  Example:

  Selecting the model type business process model with the model attribute "Date last changed", the comparison operator "<" and inputting the value "01.01.2000" lists all models that were last changed before January 1st 2000. (Note, the dates stored in the **ADOxx** database include the time as well as the date).

You can carry out a standardised query by clicking **"Evaluate"**. Clicking **"Add"** changes the query into an AQL expression which is then displayed in the field **"User defined queries"**. You can edit this query by entering additional AQL expressions (see chap. 14., p. 790). You can combine or connect the queries by using the buttons **"AND"**, **"OR"** and **"DIFF"**. Delete the contents of the field **"User defined queries"** by clicking on the **"Clear"** button.

You can create user defined queries by yourself in the field **"User defined queries"**. By using AQL you can define any query you like. You may of course use standardised queries to support you when creating user defined queries.

Results

After you have clicked on the OK button, the models found are displayed in the window **"Search result"** (see fig. 243). These models may now be opened or exported. Additionally, you can view the model attributes of the models found in the **ADOxx** browser.

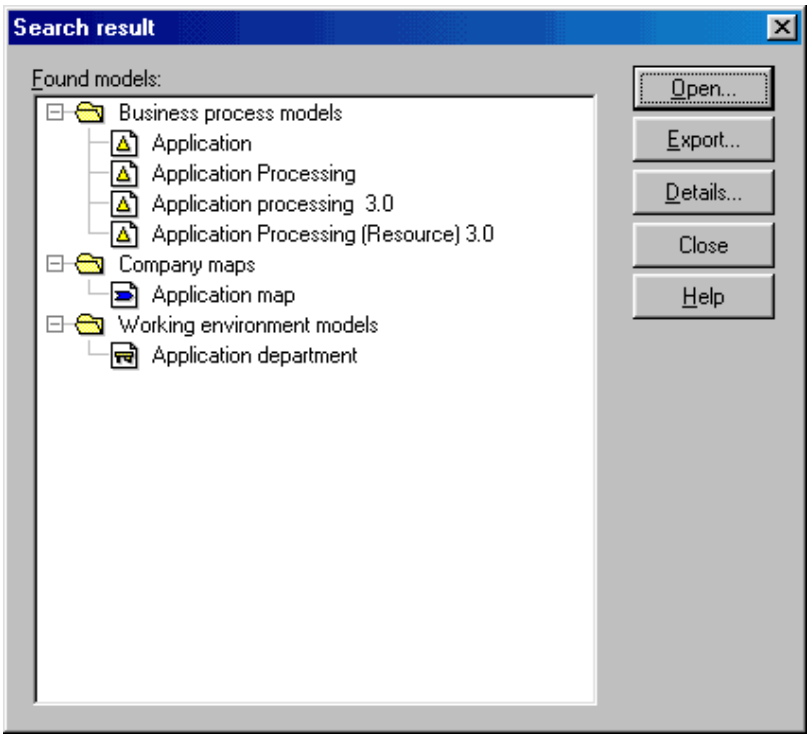


Figure 243: Search result

To open the models, click on the **"Open"** button. The window "Open model" (see fig. 226) is displayed. In this window the models found during the search are already selected and you can define how the selected models should be opened by activating the appropriate options.

If you wish to export the models, click on the **"Export"**button. The window "ADL-Export - model selection" (see fig. 415) opens up. In this window you can see the models found, divided into model types and already selected for export.

Click on the **"Details"** button to display the model attributes. The model attributes of the models found will then be displayed in the **ADOxx** browser. You can select the model attributes you wish to display in the browser's popup-menu, option "Attribute columns". In the window "Model attributes - selection" you can select the attributes you wish to display (see fig. 244).

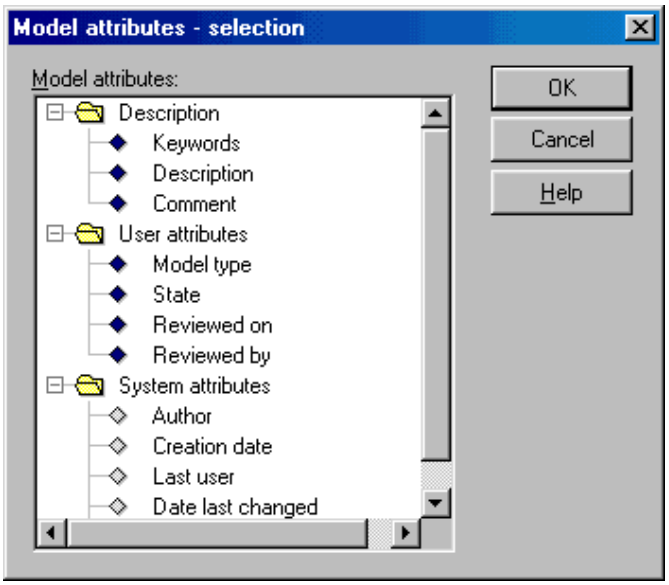


Figure 244: Model attributes - Selection

If no models have been found, you will receive an appropriate message. Close the window by clicking on the OK button or by pressing the enter key.

2.4.14.2 Process Hierarchy (tabular)

The **model search within the process hierarchy** is carried out according to one of six search criteria. The search comprises all business process models stored in the **ADOxx** database.

Hint: The model search within the process hierarchy is only possible for process modules over the reference attribute "called process".

Start the model search within the process hierarchy by selecting the option "Search models" from the "Model" menu, sub menu item "Process hierarchy (tabular)". The window "Process hierarchy - model selection" (see fig. 245) appears, in which you can define search criteria and the start model for the search.

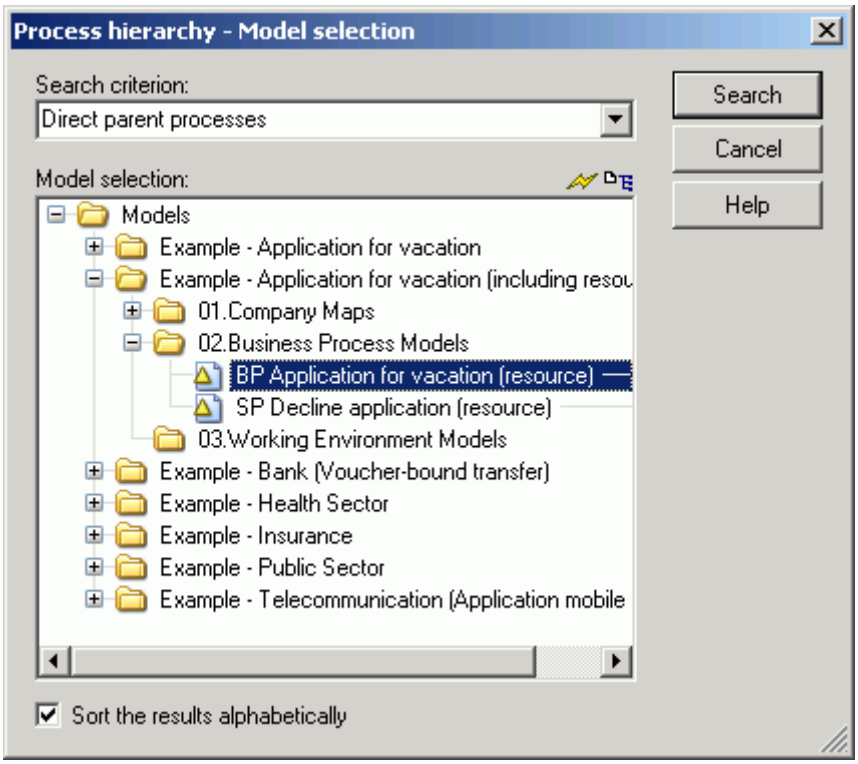


Figure 245: Process hierarchy (tabular) - Model selection

Hint: The model search within the process hierarchy refers exclusively to business process models.

The following search criteria are available in the window "Process hierarchy - model selection" (see fig. 245) :

- "**Direct parent processes**": Starting from one business process model, the system searches for all the business process models which call the initial model **directly** as a sub process.
- "**All parent processes**": Starting from one business process model, the system searches for all the business process models which call the initial model **directly or indirectly** (through other sub processes).
- "**Direct child process**": Starting from one business process model, the system searches for all the business process models, which are called **directly** by the initial model.
- "**All child processes**": Starting from one business process model, the system searches for all the business process models which are called **directly or indirectly** by the initial model. The result will be displayed in a table (see chap. 2.4.14.4, p. 281).
- "**Whole Hierarchy**": Starting from one Business Process Model the system will search for all Business Process Models, which are called **directly or indirectly** by the initial model. The whole hierarchy will be displayed.
- "**Isolated processes**": The system searches for all the business process models which neither call another business process model nor are called by other business process models.
- "**Main processes**": The system searches for all the business process models which are not called by other business process models, but which call other business process models.

Hint: When searching for "**Isolated processes**" and "**Main processes**" no initial model is necessary. That is why the model selection is faded out.

Choose the process from the **BP model list**.

When the **option "Sort the results alphabetically"** is active, the processes found are listed in alphabetical order.

Start the search within the process hierarchy by clicking on the button **"Search"**.

If the search produces results, i.e. existing models meet the search criteria; they will be displayed in the ADOxx browser (see chap. 8., p. 66).

ATTENTION: Since the search is carried out on the processes saved in the **ADOxx** database only, changes, which may have been made but have not yet been saved are not taken into account during the search!

Hint: When searching for direct child processes, for all child processes or for the entire hierarchy of a business process model, it may happen that a business process model, which has not been saved in the **ADOxx** database is referenced as a sub process. In this case "(does not exist)" is appended to the name of this business process model.

If no models, which meet the search criteria are found, an appropriate message window will be displayed. Close the window by clicking on the OK button or by pressing the enter key.

Hint: By analogy with the text in this window, the message "There is no child process" is displayed when searching for direct/all child processes, the message "There is no isolated process" when searching for isolated processes and the message "There is no main process" when searching for main processes.

Close the window by clicking on the OK button or by pressing the enter key.

2.4.14.3 Search Result

For all search criteria (exception: "All child processes") the result is shown as a list containing the model attribute values in the ADOxx browser (see chap. 8., p. 66).

The first, grey column shows the models, the other columns show the values of the model attributes. You can select which model attributes should be displayed by using the option "Attribute columns" from the browser's popup-menu. Select the attributes you wish to display in the window **"Model attributes - selection"**. Click on the **OK button** to close the window.

You may also copy the search results into the clipboard, save them to a file or print them out. Clicking on the button **"Open"** opens the window "Open model" (see fig. 226) where you can open those business process models listed in the **ADOxx** browser. By clicking on the button **"Export"** you can export (see chap. 6.2.1, p. 468) the models found. Close the window by clicking on the "Cancel" button.

2.4.14.4 Model Search for All Child Processes - Search Result

The result of the model search for **all child processes** is displayed as a table in the ADOxx browser (see chap. 8., p. 66). The first column (left-most, coloured grey) contains all calling, the first line (topmost, grey shaded) all called business process models.

An entry (e.g. "xxx") in a given cell means that the BP model of the particular line calls the business process model in the particular column.

2.4.14.5 Process Hierarchy (graphical)

Hint: The availability of the "Process hierarchy (graphical)" function depends on the definition of your application library.

To create a graphical process hierarchy, select in the "Model" menu the "Search models" menu item and then the submenu "Process hierarchy (graphical)".

Hint: If you generate the graphical process hierarchy in a model and want to save it, you must open an empty model of the appropriate model type before calling this function.

The "Select process hierarchy" window will then be displayed (see fig. 246).

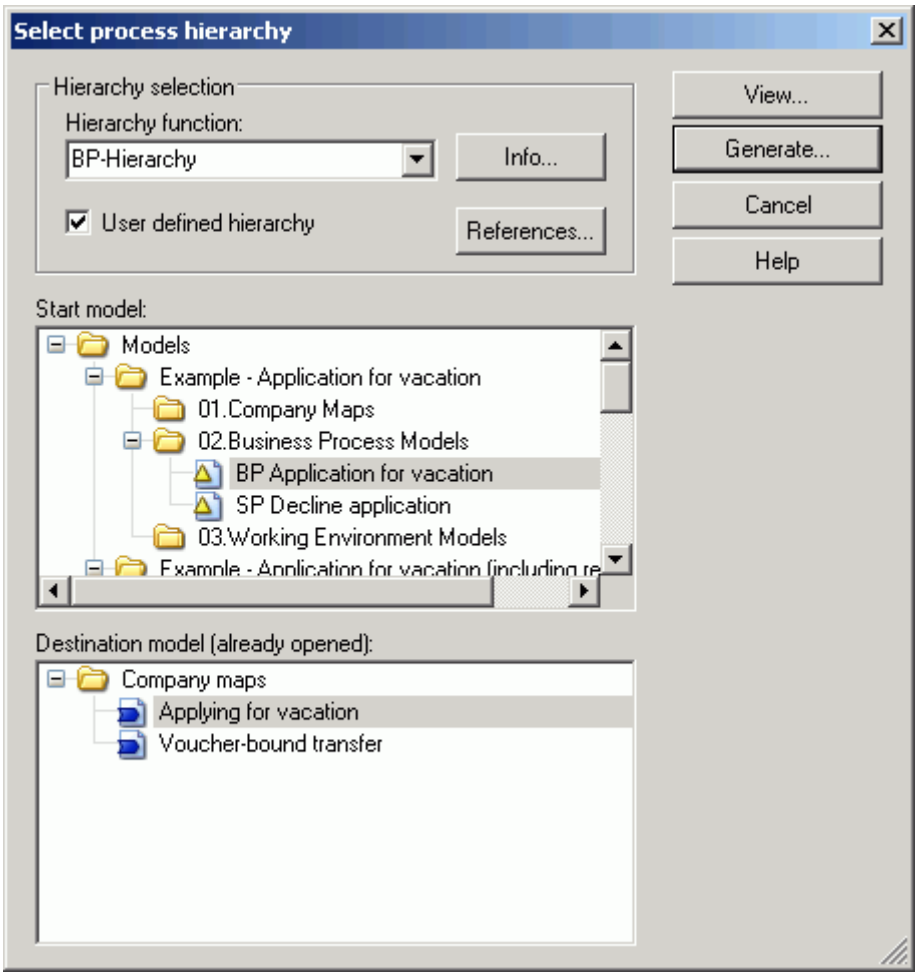


Figure 246: Process hierarchy (graphical) - Model selection

Select in the **"Hierarchy function" field** one of the functions available. If you click on the **"Info" button**, a message window with the model type for the start and target model will be displayed.

Activate the **"User defined hierarchy" option** to change the references - settings (see chap. 11.1, p. 97) predefined in the application library by clicking on the **"references" button**.

Select in the **"Start model" list** the model from which the references to be displayed in graphic should start. If you want to save the process hierarchy, select in **"destination model (already opened)"** the prepared (empty) destination model.

Click on the **"Generate" button** to generate the process hierarchy in the previously selected opened model.

Hint: The "Generate" button is only available, if a model has been selected from the "destination model (already opened)" list. The results will be taken over in this selected model.

Click on the **"View" button** to generate the process hierarchy in the preview window.

Before taking over the graphical process hierarchy in a model or when generating the view, the "Process hierarchy" window with the generated hierarchy will be displayed.

To take over the generated model graphics in the selected destination models, click on the "Assign" button.

2.4.15 Deleting models

All models saved in the ADOxx database are listed in the "Delete models - selection" windows (see fig. 247)-in the model group hierarchy.

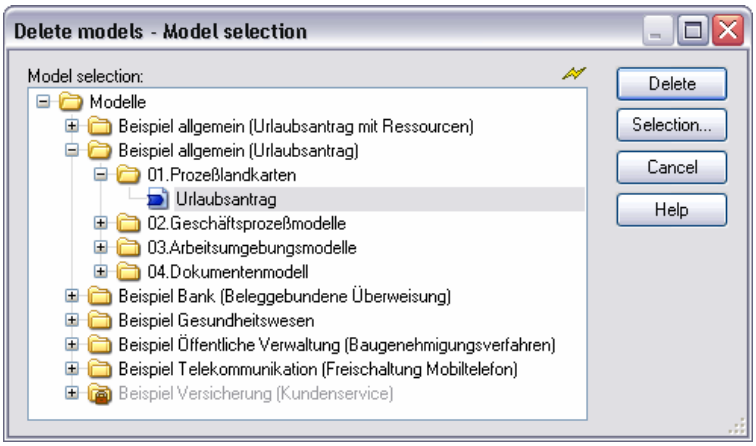


Figure 247: Delete models - selection

Select the models you want to be deleted and click **"Delete"**.

> **Hint:** The models to be deleted, which are referenced by other models, will be displayed in the "Delete models - Information" window before the deletion (see fig. 248).

If you click **"Selection"** the "Find model" window (see fig. 242) will be shown, in which you can limit the model selection via the search (see chap. 2.4.14.1, p. 276) to specific model attributes.

Before the models are finally deleted from the ADOxx database, an appropriate security message is displayed.

On confirming the security message, it will be checked whether the models to be deleted are referenced by other models. If this is the case for at least one model, the window "Delete models - Information" will be displayed. This window lists the models, which are referenced by other models in the ADOxx database. Click **"Delete"** again to continue the deletion. If you hit the **"Cancel"** button the deletion will be cancelled.

ATTENTION: A model can only be deleted if no ADOxx user has opened this model in the ADOxx Modelling Toolkit at that time.

ATTENTION: The deletion of models from the **ADOxx** database cannot be undone!

2.4.15.1 Show Referenced Models

In the window "Delete models - Information" (see fig. 248), all models, which are referenced by other models will be displayed.

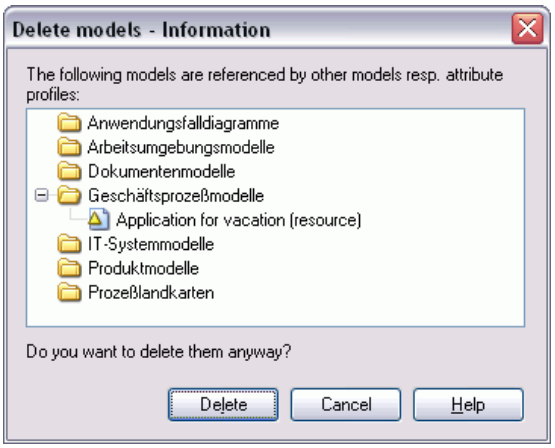


Figure 248: Delete models - referenced models

Click **"Delete"** to continue the deletion of the selected models.

Hint: When deleting referenced models, the references outgoing from the start models can no longer be followed and will be displayed as broken references.

2.4.16 Application Models

Defining application models is necessary if you wish to execute the Simulation algorithms "Capacity Analysis" (see chap. 4.3, p. 354) and "Workload Analysis" (see chap. 4.4, p. 367) in the Simulation Component. An application model combines exactly one working environment model (organisational structure) with one or more business process models.

The assignment of performers (see chap. 2.2.12.14, p. 222) links the activities to be executed in a business process model to a performer or a group of performers in a working environment model. The definition of application models makes it possible, too, that several business processes can be executed by the performers of a working environment model at the same time. This results in a realistic Simulation of the capacity and workload of performers.

Hint: An application model only has to contain main business process models. The Simulation Component automatically loads all sub processes called during Simulation.

Hint: If you are using hierarchical Working Environment models (see p. 224), the main Working Environment model must be contained in the application library. Then sub Working Environment models referenced in this model will be automatically loaded during the Simulation.

Select the option "Application models" from the "Model" menu. The window "Application models" (see fig. 249) appears, listing the application models which are already defined. Click on an application model, if you wish to see the business process models and working environment model it comprises.

When using a time and cost application library, you can switch between these two views:

- Version specific application models
- Sequence version specific application models

Sequence version specific application models contain (implicit) all the versions of the models, which define the application model. The current assignment in the Simulation appears dynamically through the version context.

Version specific application models only contain a certain version of the models, which define the application model.

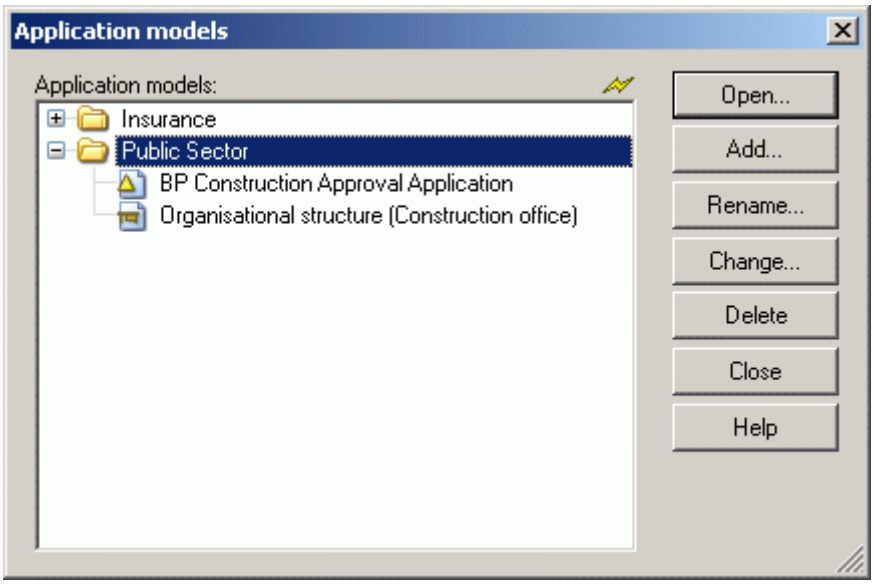


Figure 249: Application models

Starting from the "application models" window you can:

- Open an application model (see chap. 2.4.16.1, p. 285).
- Add a new application model (see chap. 2.4.16.2, p. 285).
- Rename application models (see chap. 2.4.16.3, p. 287).
- Change the composition of an application model (see chap. 2.4.16.4, p. 287).
- Delete application models (see chap. 2.4.16.5, p. 288).

2.4.16.1 Open Application Model

When opening an application model, you will open all the models which are assigned to this application model (optional including referenced models). Select in the "Application models" window (see fig. 249) the application model you wish to open.

Clicking on the button **"Open"** opens the window "Open model" (see fig. 226) and only the models assigned in the application model will be listed - sorted according to the model types and already selected.

The available options correspond to those available when you want to open saved models (see chap. 2.4.2, p. 257).

2.4.16.2 Add Application Model

To define an application model, click in the "application models" window (see fig. 249) on the **"Add" Button** .

The "Application model definition" window (see fig. 250) will then be displayed (see fig. 250), in which all the Business Process Models and the Working Environment models saved in the ADOxx database are listed.

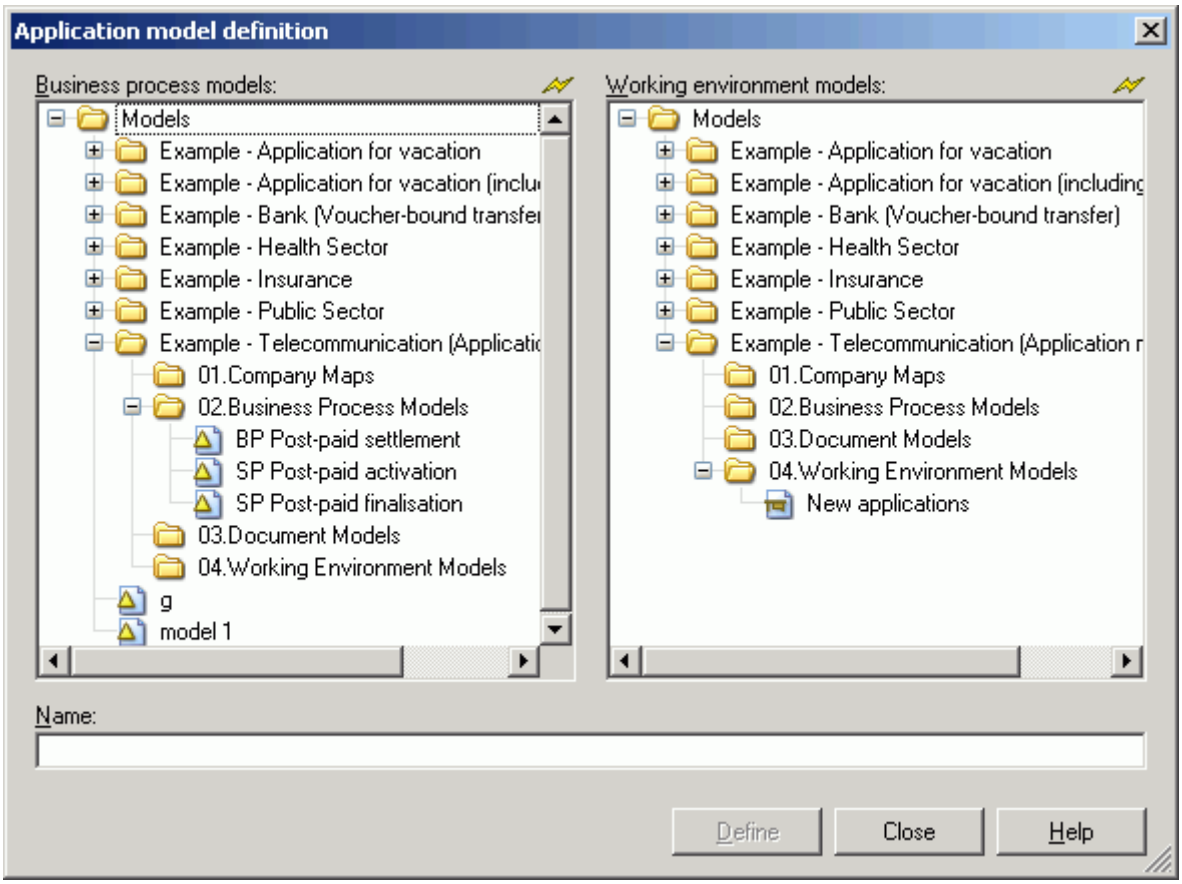


Figure 250: Add application models

Select the models, which you wish to assign to the application model.

ATTENTION: You must not select sub processes, which are called by the business process models selected. They will automatically be loaded and executed during Simulation.

Enter a name for the application model into the field **"Name"**.

ATTENTION: An application model consists of exactly one working environment model and at least one business process model. The name of the application model must not be identical to the name of any other application model.

Clicking on the button **"Define"** causes the application model to be defined according to your specification. A message window will confirm this.

The window "Application model definition" will now be displayed again. You may now define another application model. When clicking on the "Cancel" button the window will be closed and the updated list of application models will be displayed (see fig. 249).

Hint: You may also define application models in the Simulation Component (see chap. 4., p. 332) before you start the capacity (see chap. 4.3, p. 354) or the Workload Analysis (see chap. 4.4, p. 367). However, application models may be changed (see chap. 2.4.16.4, p. 287) or deleted (see chap. 2.4.16.5, p. 288) **only in the Modelling Component**.

2.4.16.3 Rename Application Model

To rename an application model, select from the application model list in the "Application models" window (see fig. 249) the application model, whose name you want to change. Once you have clicked on the **"Rename" button**, the "Rename application model" window (see fig. 251), in which you can enter the new name, will be displayed.

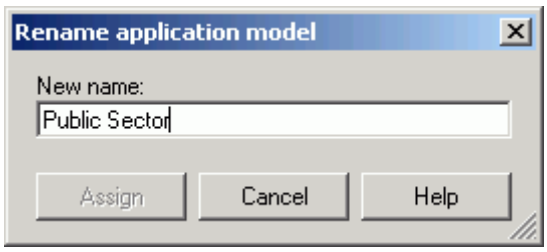


Figure 251: Rename application model

Enter the desired new name.

Confirm the changes by clicking on the **"Assign" button**.

2.4.16.4 Edit Application Model

If you wish to change the composition of an application model, select from the list of application models in the "Application models" window (see fig. 249) the application model you wish to change and then click on the **"Change" button**.

The "Application model <model name>: Change" (see fig. 252) window, in which the models assigned to the application model are highlighted, will now be displayed.

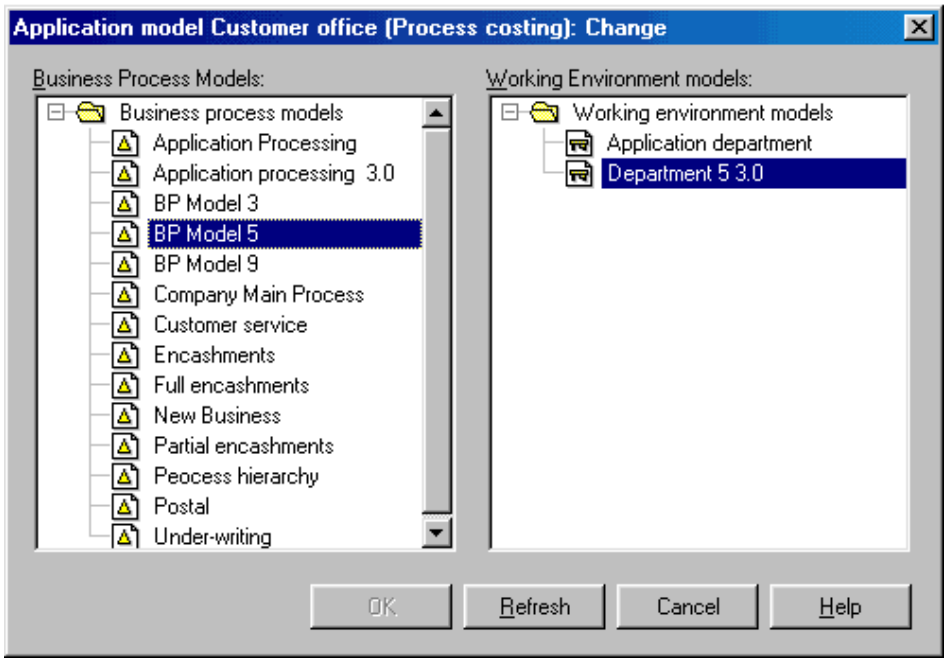


Figure 252: Edit application model

Change the composition by selecting or deselecting the respective business process and working environment models listed.

Confirm the changes by clicking on the OK button.

2.4.16.5 Delete Application Model

If you wish to delete an application model, select the application model to be deleted from the list of application models in the "Application models" window (see fig. 249).

After you have clicked on the button **"Delete"**, a window appears asking you whether you really want to delete the application model selected.

Click on the "Yes" button to proceed with the deletion. Clicking on the "No" button closes the window and shows the window "Application models".

> **Hint:** Deleting an application model only deletes the application model. The business process models and the working environment model, which had been assigned to the application model are **not** deleted.

2.4.17 Print Models

Call print function:

There are three different ways to start printing:

- By clicking on the **Smart-Icon**
- By selecting the **"Print"** option from the **"Model"** menu.
- By pressing the key combination **<Ctrl>+<p>**

The window "<*model name*> (<*model type*>) - Print" (see fig. 253) is displayed.

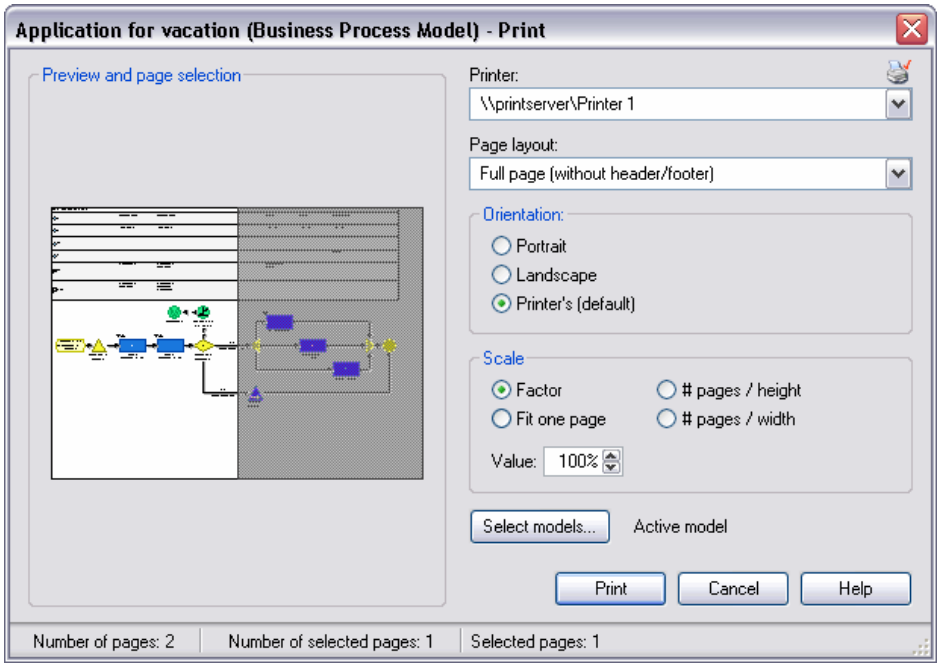


Figure 253: Printing model

Print:

The field **"Print"** contains a list of all available printers. Select a printer from the list (default printer is pre-selected). The printer settings may be displayed and if necessary changed by clicking on the smart icon . If any problem with printers occur contact your system administrator.

Select your settings for **"Page layout"**, **"Layout parameters"** and **"Scale"**. Check the **"Layout parameters"** chapter (see p. 194) for details.

By default, the active model will be printed. By pressing the button **"Select model"** you can select an additional models for printing (see chap. 2.4.17.1, p. 289).

The window **"Preview and page selection"** is updated with each change. This print preview helps to choose pages for the printout (see chap. 2.4.17.3, p. 290).

Hint: For very large models (drawing area wider than 15x15m or more than 4000 pages using factor 100%) the page boundaries in the preview cannot be displayed. In this case the text "<*no*> pages - page boundaries cannot be displayed! will be shown above the preview, the page selection is not available then.

For displaying the page boundaries again you can reduce the scaling factor and/or enlarge the print window.

In the status line, you will find information about the printout of several pages depending on the current layout and scale (total number of pages, number of selected pages, numbering of selected pages).

Hint: The numbering of pages is done by the line from the left to the right.

To start printing click on the **"Print"** button. In the status window, you will find information about current printing status. By clicking on the "Cancel" button, the printing process will be cancelled immediately.

Hint: Depending on your system configuration, the information window may appear after the starting printing operation.

2.4.17.1 Print Models - Model selection

In the window "Print - Model selection" (see fig. 254) all open model windows are displayed.

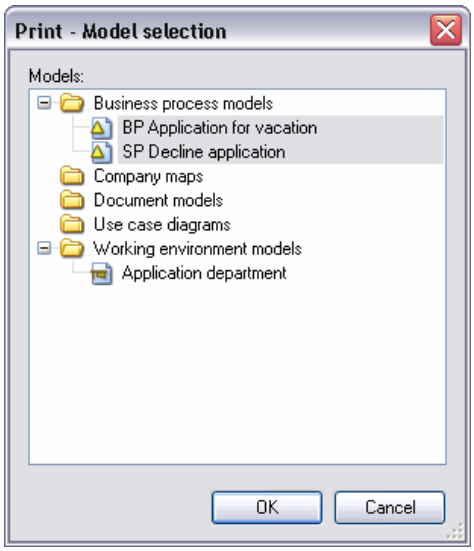


Figure 254: Print - Model selection

Select the models for printing and press OK. You can change the print setting in the window "Print multiple models with identical settings" (see fig. 255, p. 290).

2.4.17.2 Print several models with identical settings

Print several models with identical setting by pressing the button **"Select model"** in the Print dialogue (see fig. 253, p. 288) and by selecting these models (see chap. 2.4.17.1, p. 289).

As an alternative printing of several models can be called from the ADOxx Explorer (see chap. 4., p. 54) using the view "Opened model windows".

In the dialogue "Print multiple models with identical settings" (see fig. 255) the following settings are available.

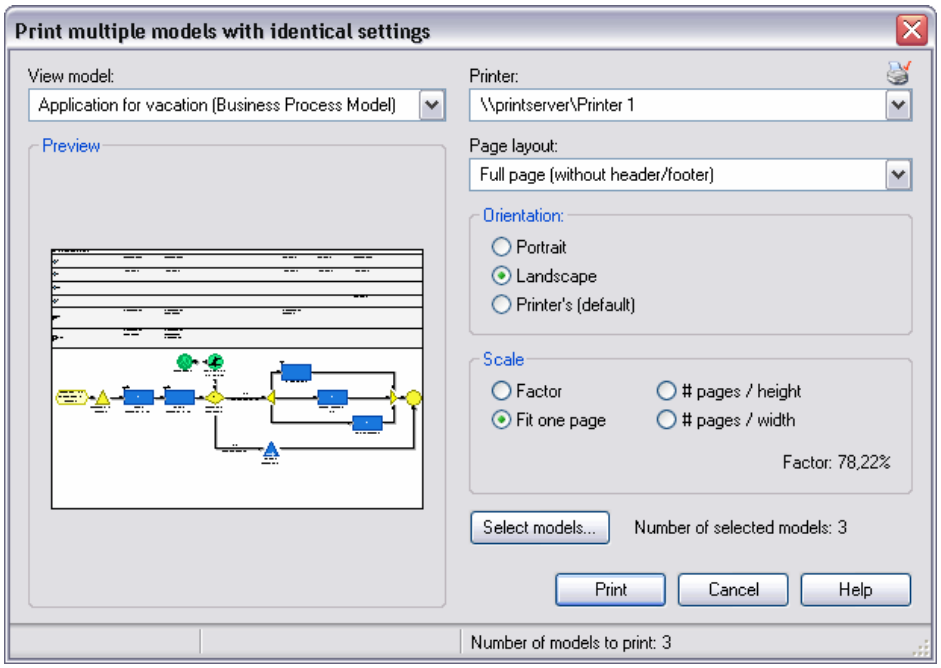


Figure 255: Print several models

In the additional **"Model view"** field, you can switch between all selected models.

The window **"Preview"** is updated with each selection.

Select your settings for **"Page layout"**, **"Layout parameters"** and **"Scale"**. Check the **"Layout parameters"** chapter (see p. 194) for details.

To start printing click on the **"Print"** button. In the status window, you will find information about current printing status. By clicking on the "Cancel" button, the printing process will be cancelled immediately.

2.4.17.3 Select Pages

If a print out needs several pages, you can determine which pages actually shall be printed (by default: all).

To omit particular pages during printing, click on them in the **"Preview and page selection"**. They will be highlighted with grey and omitted during printing. You can un-mark pages by clicking on them again:

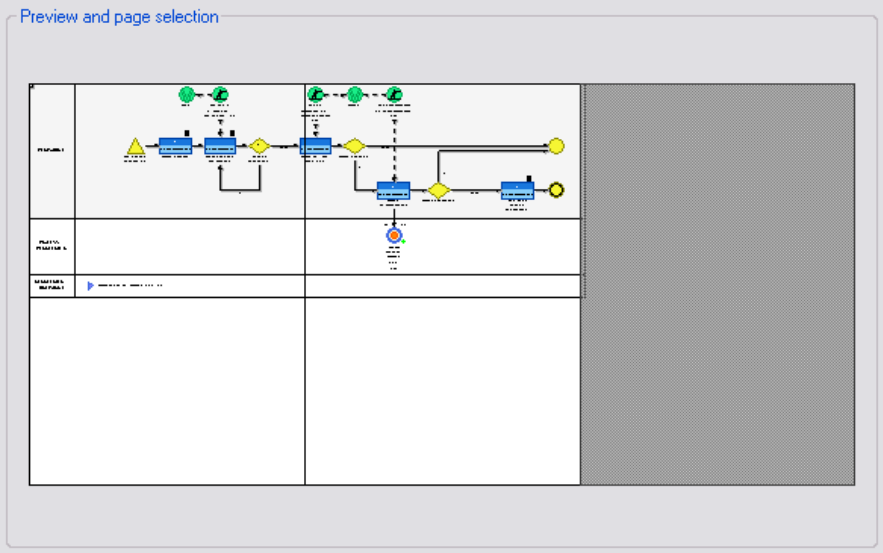


Figure 256: Preview and page selection

2.5 Editing Attribute Profiles

By selecting the menu item "Attribute profiles" in the menu "Edit", it is possible starting from the list of attribute profiles, to edit attribute profiles and attribute profile folders. All existing attribute profiles and attribute profile folders will be shown in the "Attribute profile" list:

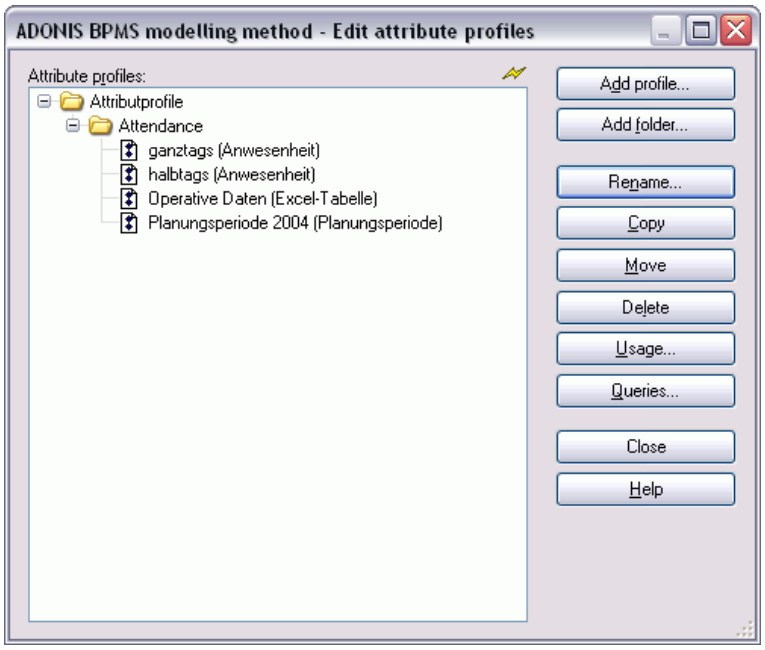


Figure 257: Attribute Profile (group) list

Clicking on the following buttons will enable you to:

"New attribute profile" Create a new attribute profile (see chap. 2.5.6, p. 294) in a previously selected attribute profile folder.

"New version" Save a previously selected attribute profile folder as a new version (see chap. 2.5.7, p. 295).

	Note: This button is only available if the attribute profiles are defined in an application library with time-related versioning (see chap. 10.1.2, p. 92).
"New group"	Create a new (main) attribute profile folder (see chap. 2.5.1, p. 292) in a previously selected attribute profile folder or in a not previously selected attribute profile folder.
"Rename"	Assign a new name (see chap. 2.5.2, p. 293) to a previously selected attribute profile folder.
"Edit"	Edit (see chap. 2.5.8, p. 295) the values of a previously selected attribute profile or the values of several previously selected attribute profiles at the same time.
"Copy"	Copy a previously selected attribute profile folder (see chap. 2.5.3, p. 293) or a previously selected attribute profile (see chap. 2.5.12, p. 297).
"Move"	Move a previously selected attribute profile folder (see chap. 2.5.4, p. 294) or a previously selected attribute profile (see chap. 2.5.13, p. 298) into another attribute profile folder.
"Delete"	Delete a previously selected attribute profile folder (see chap. 2.5.5, p. 294) or a previously selected attribute profile (see chap. 2.5.14, p. 298).
"Usage"	Show a list of the objects and models, in which the previously selected attribute profiles are referenced (see chap. 2.5.15, p. 298).
"Query"	Evaluate (see chap. 2.5.16, p. 299) the contents of the attribute profiles using the criteria you have defined.

The general functionality (see chap. 3.1, p. 41) as well as the above-listed functionality of the attribute profile list are available if you open the popup-menu (right mouse button) .

Hint: The availability of functions depends on the previous selection of an attribute profile or of an attribute profile folder.

2.5.1 Add Attribute Profile Folder

It is possible to create attribute profile folders in each hierarchy level, i.e. you can create a new attribute profile folder in each attribute profile group.

To create a new attribute profile folder, select the attribute profile folder, which should contain the new attribute profile folder and then click on the button "add folder".

Hint: To create an attribute profile folder as a main group, i.e. on the upper hierarchy level, no attribute profile folder should be selected.

The window "Add attribute profile folder" (see fig. 258) appears.

Figure 258: Add attribute profile folder

Enter the name of the new attribute profile folder and then click on the OK button.The window will be closed and the updated hierarchy of the attribute profile folders will be shown.

ATTENTION: The name of the new attribute profile folder must be unique on the level of the main folders and within each attribute profile folder.

2.5.2 Rename Attribute Profile Folder

If you want to give a new name to an already existing attribute profile folder, select this attribute profile folder and then click on the button "Rename". The window "Rename attribute profile folder" will appear (see fig. 259), in which the current name (Field "Old name") is displayed. In the field "New name" the current name is also entered and selected so that you can directly enter the new name.

Figure 259: Rename attribute profile folder

Enter the name of the attribute profile folder and then click on the OK button. The window will be closed and the updated hierarchy of attribute profile folders will be displayed.

ATTENTION: The name of the new attribute profile folder must be unique on the level of the main folders and within each attribute profile folder.

2.5.3 Copy Attribute Profile Folder

Select the attribute profile folder you want to copy into another attribute profile folder and then click on the button "Copy". As soon as you move the mouse pointer onto the list of the model hierarchy, it will be changed into .

Then click on the attribute profile folder to which you want to copy the previously selected attribute profile folder.

If you want to move the attribute profile folder to the upper hierarchy level (main group), click on the free space below the model hierarchy or on the left near the main attribute profile folders.

Hint: When you copy a new attribute profile folder, the whole structure of this folder, i.e. all contained sub-profile folders as well as all attribute profiles contained in the folders, will be doubled.

Hint: When you copy an attribute profile folder at the same place, a number will be automatically appended to the name of the profile folder, in order to make it as clear as possible.

2.5.4 Move Attribute Profile Folder

If you want to move an attribute profile folder, select this attribute profile folder and then click on the button "Move". As soon as you move the mouse pointer onto the list of the model hierarchy, it will be changed into .

Click on the attribute profile folder to which the previously selected attribute profile folder should be moved to.

If you want to move the attribute profile folder to the upper hierarchy level (main folder), click on the free space below the model hierarchy or on the left near the main attribute profile folders.

Hint: The name of the attribute profile folder to be moved shall not be given to another attribute profile folder in the new attribute profile folder or within the main folder.

2.5.5 Delete Attribute Profile Folder

Select one or more attribute profiles that you want to delete and then click on the button "Delete".

Before the final deleting of the attribute profile folder, an appropriate security query will appear.

Should some references to attributes profiles (contained in the attribute profile folder to be deleted) remain, then the window "Delete attribute profile - used attribute profile" (see fig. 260) will be displayed, indicating from which objects in which models these attribute profiles are referenced.

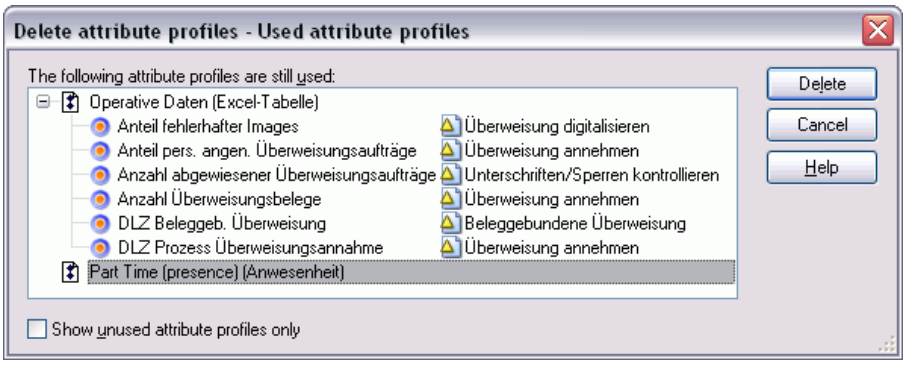


Figure 260: Delete attribute profile folders - used attribute profiles

Select the attribute profiles you want to delete despite its usage, activate the option "Delete selected attribute profiles" and click on the OK button.

2.5.6 Add Attribute profile

When you add an attribute profile, an instance of the attribute profile class defined in the application library will be created.

Hint: Attribute profiles must be created in a previously selected attribute profile folder.

To create a new attribute profile, select the attribute profile folder, in which you would like to create it in and then click on the button "New Attribute profile".

If there are several attribute profile classes defined in your application library, the window "New attribute profile" will appear (see fig. 261).

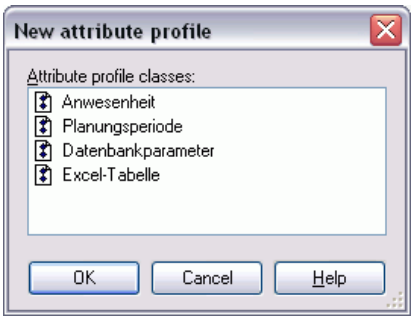


Figure 261: Add attribute profile

Select the attribute profile class from which the instance of the new attribute profile must be created and then click on the OK button.

Hint: When using an application library with **time-related versioning** (see chap. 10.1.2, p. 92), you can give a version number for the new attribute profile.

In the window displayed as a result (see fig. 263), you can define the attribute value of the new attribute profile (see chap. 2.5.8, p. 295).

2.5.7 Save Attribute Profile as New Version

Hint: The functionality "Save attribute profile as a new version" is only available if the attribute profiles are defined in an application library with time-related versioning (see chap. 10.1.2, p. 92).

In the window "Save as new version", you can define a new version number to save an existing attribute profile.

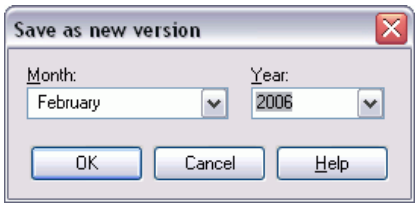


Figure 262: Save as new version

2.5.8 Edit Attribute Profile Values

When editing an attribute profile, you will define the attribute values of this profile. These attribute values will be integrated into the modelling object through the referencing of the attribute profile during modelling in the Modelling Toolkit.

The editing of an attribute profile is carried out in an ADOxx Notebook (see fig. 263)- the same way as for attributes in objects.

Hint: The representation of attributes and the structure of ADOxx Notebooks depends on the cutsomised definition in the application library.

Hint: Select several attribute profiles to edit their values simultaneously in the ADOxx browser (see chap. 2.5.9, p. 296).

Figure 263: Edit attribute profile

Define the value of an attribute profile by entering the appropriate values in the attribute fields.

ATTENTION: The name of an attribute profile must be unique within an attribute profile class.

Then click on the button "Close" to save the attribute values and close the ADOxx' Notebook. The updated list of attribute profiles will be displayed.

2.5.9 Edit several attribute profile values simultaneously

When editing several attribute profiles, you will define the attribute values of these profiles in the ADOxx browser (see chap. 8., p. 66) in tabular form (see fig. 264). Attribute profiles will be displayed in rows and attributes in columns.

Hint: The representation of attributes and the structure of ADOxx Notebooks depends on the customised definition in the application library.

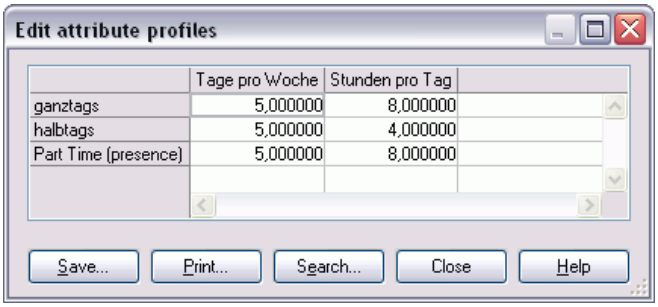


Figure 264: Edit multiple attribute profiles at a time

Define the values of attribute profiles by entering values in the appropriate cells.

ATTENTION: The name for an attribute profile must be unique inside of an attribute profile class.

Then click on the button "Close" to save the attribute values and close the window with the ADOxx browser. The updated list of attribute profiles will be shown.

2.5.10 Show Attribute Profile Values

When viewing an attribute profile, you can display the defined attribute profile values in an ADOxx Notebook (see fig. 265).

Hint: The representation of attributes and the structure of ADOxx Notebooks depend on the customised definition in the application library.

Figure 265: Show attribute profile

Hint: If it is not possible to edit attribute profile values. If necessary, contact your ADOxx consultant.

Click on the button "Close" to close ADOxx Notebook.

2.5.11 Show Several Attribute Profile Values simultaneously

When viewing several attribute profiles, you can display the defined attribute profile values in the ADOxx browser (see chap. 8., p. 66) in tabular form (see fig. 266). The attribute profiles will be displayed in rows and the attributes in columns.

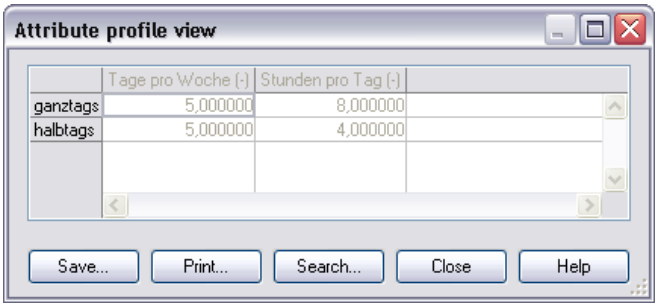


Figure 266: Show several attribute profile values simultaneously

Click on the button "Close" to close the window with the ADOxx browser.

2.5.12 Copy Attribute Profile

If you want to copy an attribute profile to another attribute profile folder, select this attribute profile and then click on the button "Copy". As soon as you move the mouse pointer onto the list of the model hierarchy, it will be changed into .

Then click on each attribute profile folder to which the previously selected attribute profile should be copied.

Hint: Since attribute profile names in an application library must be unique, a generated number will be automatically appended to the name when creating the copy of an attribute profile.

2.5.13 Move Attribute Profile

Select the attribute profile you want to move and then click on the button "Move" .As soon as you move the mouse pointer onto the list of the model hierarchy, it will be changed into .

Click on the attribute profile folder to which the previously selected attribute profile should be moved.

2.5.14 Delete Attribute Profile

Select the attribute profiles you want to delete and then click on the button "Delete".

Before the final deleting of attribute profiles, an appropriate security alert will appear.

If some references to the attribute profiles to be deleted should remain, the window "Deleting attribute profiles - Used attribute profiles" (see fig. 267) will be displayed indicating from which objects in which models these attribute profiles are referenced.

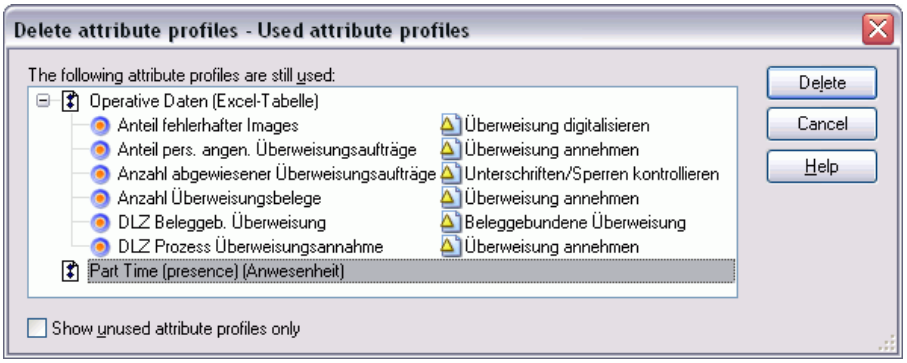


Figure 267: Delete attribute profiles - Used attribute profiles

Select the attribute profiles, which you want to delete despite their usage and click on the OK button.

Activate the option "Show not used attribute profiles only", to only show the attribute profiles which are currently not being referenced.

2.5.15 Show Used Attribute Profiles

The window "Attribute profile - Usage" (see fig. 268) shows all previously selected attribute profiles and the models and objects referencing these attribute profiles.

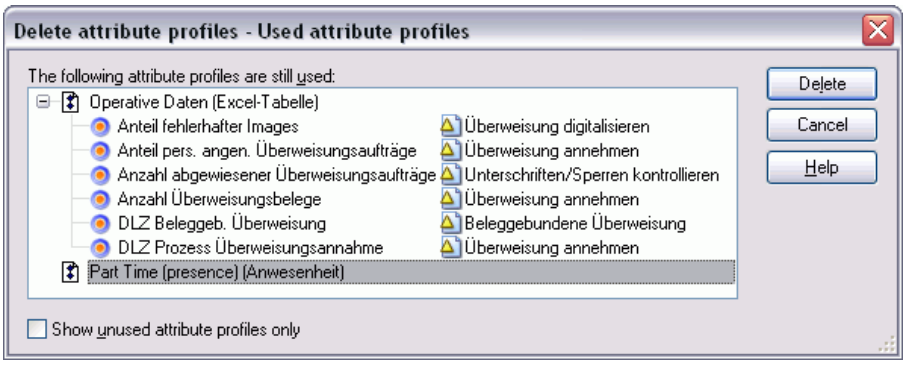


Figure 268: Attribute profile usage

Select a start object and then click on the button "Follow" to open the model containing this object.

Activate the option "Show not used attribute profiles only", to only show the attribute profiles which are currently not being referenced.

2.5.16 Carry out Queries on Attribute Profiles

In the window "Queries on attribute profiles" (see fig. 269) , you can carry out standardised queries (see chap. 2.5.16.1, p. 299) and user-defined queries (see chap. 2.5.16.2, p. 301).

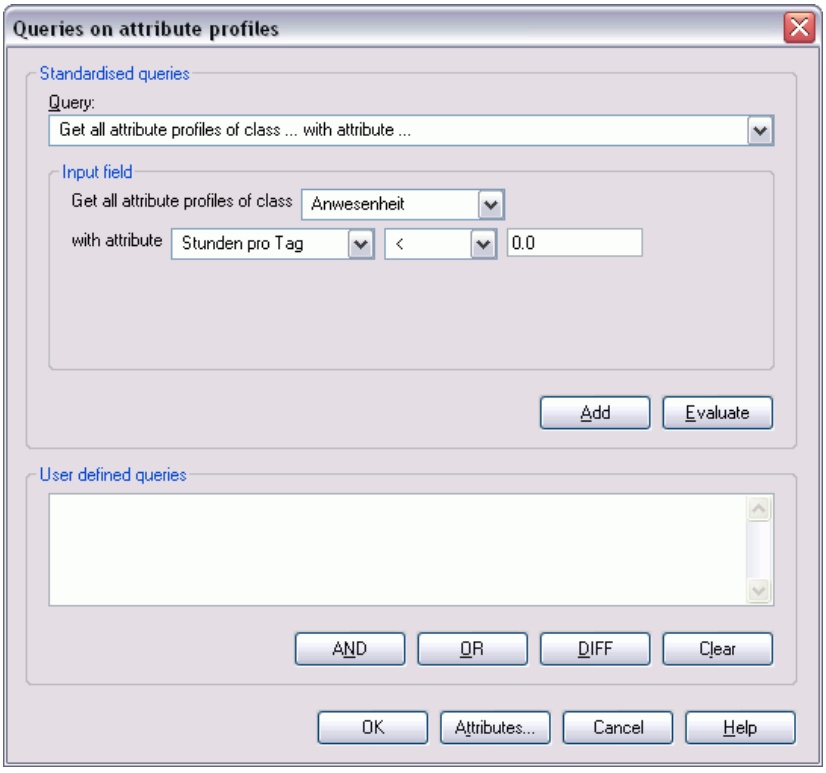


Figure 269: Standardised and user-defined queries on attribute profiles

2.5.16.1 Standardised Queries on Attribute Profiles

To carry out standardised queries, select a standardised query from the list "Query" (see fig. 270). An appropriate text with gaps appears in the input field of the selected query. Enter the required

information (e.g. select name of the attribute profile/attribute) or select any value from the class/attribute/comparison operators lists.

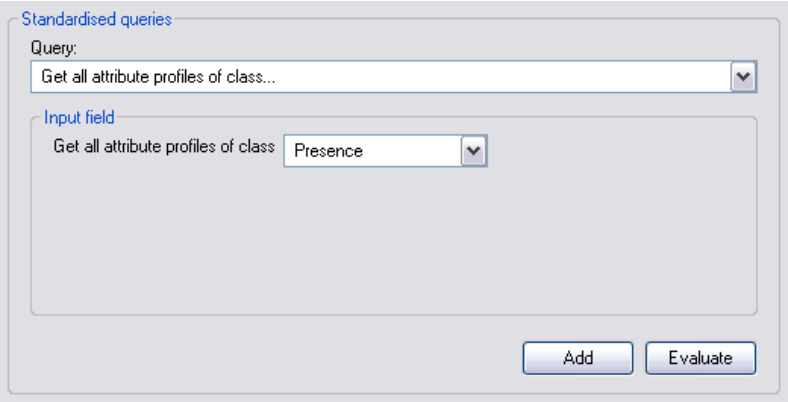


Figure 270: Standardised queries (part of the window "Queries on attribute profiles")

Standardized queries are explained below:

- **`Find all attributes of the class`** [*`Selection`*]**.**

  This query displays all attribute profiles of the selected class.

- **`Find all attributes of the class`** [*`Selection`*] **`with attribute`** [*`Selection`*][*`comparison operator`*][*`entry`*]**.**

  This query displays all attribute profiles of the selected class with a specific attribute value.

  Note: First select the class, then the attribute of the class and the comparison operator and enter the comparison value of the attribute.

- **`Find all attributes of the class`** [*`Selection`*] **`with record attribute`** [*`Selection`*][*`comparison operator`*][*`Number`*]**.**

  This query displays all attribute profiles of the selected class, which contain in their given record attribute the number of entries (rows) you have defined.

  Note: These standardised queries will only be displayed if record attributes are defined in your application library.

- **`Find all attributes of the class`** [*`Selection`*] **`with record attribute`** [*`Selection`*] **`and with column`** [*`Selection`*][*`comparison operator`*][*`entry`*]**.**

  This query displays all attribute profiles of the selected class, which contain in the given table and column a specific attribute value.

  Note: These standardised queries will only be displayed if record attributes are defined in your application library.

You can carry out the completed standardised queries by clicking on the **"Evaluate" button**. The query results (see chap. 2.5.16.3, p. 302) will be displayed in the ADOxx browser .

If you click on the **"Add" button** , the standardised query will be transformed into an AQL expression (see chap. 14., p. 790) and shown in the **"user defined query" field** (see fig. 272). From this query you can create a complex user-defined query (see chap. 2.5.16.2, p. 301) by entering further AQL expression and by adding further standardised queries.

By clicking on the **"Attribute" button** (see p. 301) you will define the attributes, which should be added to the name of the attribute profile found. If you select no attribute, the query results will be displayed **without** attribute.

Hint: Standardised queries will be saved after the completion of the query. When recalling the menu item "Queries" the standardised query will be shown as a user-defined query (see chap. 2.5.16.2, p. 301) using AQL expression. This query remains saved until another standardised or user-defined query is entered or you close ADOxx.

Hint: If an AQL expression is registered in the **"user-defined query" field**, you will carry out this user-defined query by clicking on the OK button, otherwise the current selected standardised query will be carried out.

Select Attributes for Result Display

All attributes of attribute profile classes will be shown in the window "Queries - Attribute column selection" (see fig. 271).

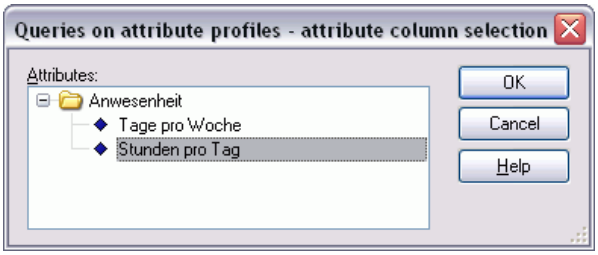


Figure 271: Queries on attribute profiles- Attribute settings

Attributes are always grouped under the attribute profile class in which they are defined. Select the attributes which you want to show in the display of query results in the ADOxx browser (see chap. 8., p. 66) and then click on the OK button.

Hint: The attribute selection remains saved until you select other or additional attributes or you close ADOxx.

2.5.16.2 User-defined Queries on Attribute Profiles

Carry out the creation of user-defined queries within the **"user-defined query" field** (see fig. 272), while connecting standardised queries, defining queries in the query language AQL (see chap. 14., p. 790) or expanding standardised queries with AQL-expressions.

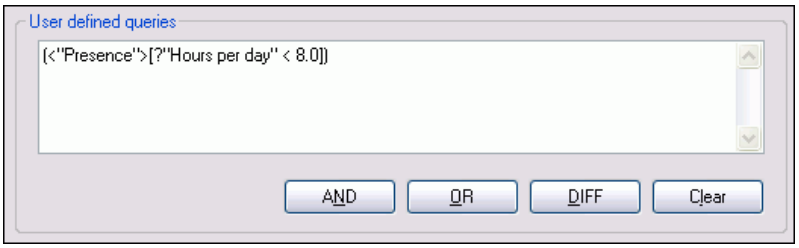


Figure 272: User defined queries (Part of the window "Query on attribute profiles")

You can use and assemble standardised queries (see chap. 2.5.16.1, p. 299) as a support for the definition of user-defined queries.

For this define a standardised query and click on the "Add" button. This will generate an AQL expression, which corresponds to your standardised query. This AQL-expression will be entered in the **"User defined query" field**.

By clicking on the **"AND"**, **"OR"** and **"DIFF"** buttons, you can combine or connect queries. You can also delete the content of the **"user-defined queries" field** by clicking on the **"Cancel" button**.

By clicking on the **"Attributes" button** (see p. 301) you will define the attributes, which should be displayed in addition to the name of the objects or connectors found.

Hint: Standardised queries (see chap. 2.5.16.1, p. 299) will be saved after the completion of the queries. When recalling the "Query" menu item, the standardised query will be displayed as a user-defined query (AQL-expression). This query will remain saved, until you enter another standardised or user-defined query or you close ADOxx.

Hint: If an AQL expression is registered in the **"user-definied query" field**, you will carry out this user-defined query by clicking on the OK button, otherwise the current specific standardised query will be carried out.

2.5.16.3 Query Results

The result of the queries will be displayed in the ADOxx browser (see chap. 8., p. 66) relative to their attribute profile class in the window "query result: <*Search criteria*>" (see fig. 273).

Hint: Instead of<*Search criteria*>, the formulated query will be shown in the window "query". This is either the text of the standardised query or an AQL expression of the user-defined query.

The name of each attribute profile found will be shown in the first (grey) column, where the attribute profiles of a class are gathered under the class name of attribute profile. If additionally you have selected attributes for the result display (see p. 301), then the values of these attributes will be displayed, the name of these attributes will be shown in the first (grey) row (see fig. 273).

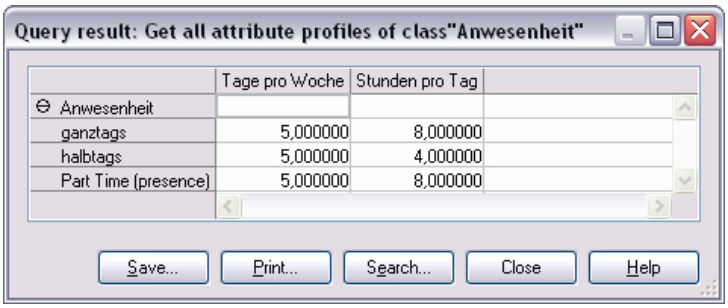


Figure 273: Queries on attribute profiles - Result display

Hint: The display of record attributes (see p. 303) expands the representation in the ADOxx browser to columns (by integrating record attributes) and rows (by integrating the table content).

Select in the context menu (right mouse button) the menu item "Select attributes" to change the display of attributes (see p. 301).

You can save the query result to a file, print it out or display it as a diagram. Moreover, the search for specific contents is possible.

Furthermore, the attribute values of attribute profiles can be edited directly in the query result browser (see chap. 2.5.9, p. 296).

In addition it is possible to sort (see p. 302) the displayed objects and connectors according to specific attribute values using the context menu (right mouse button).

Sort Results

You can sort the displayed attribute profiles according to their attribute values using the menu item "Sort" in the context menu (right mouse button).

Once you have selected the menu item "Sort", the window "Sort - Attribute selection" (see fig. 274) will be shown.

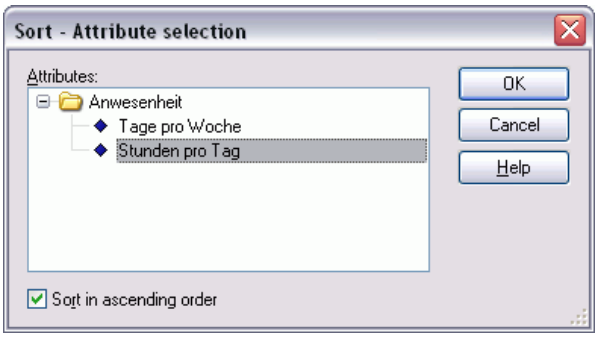


Figure 274: Sort - Attribute selection

Select from the lists the attribute according to which the query result will be sorted. The **"Sort in ascending order" option** enables the choice between ascending and descending order.

> **Hint:** The sort is carried out attribute profile class- related, i.e. the query result will be sorted using the selected attribute within the attribute profile class.

Display Record Attributes

For the result display of the attributes (see fig. 273), the first representation of record attributes is carried out in the ADOxx browser by the symbol ⊕. This means, that it is a short display of tables and that the record attributes can be displayed by a click.

Once you have clicked the cell with the symbol ⊕ the display of the table will be expanded: Either the column will be expanded to all defined record attributes or the table rows will be integrated in the display as additional rows.

> **Hint:** In the expanded display of a table, all record attributes of an object will be outlined in blue.

The symbol ⊖ indicates, that all record attributes are displayed. Clicking on it brings up a compact view of the displayed table.

To **shrink or expand all record attributes**, select in the context menu of the ADOxx browser (right mouse button) the menu item "Shrink all record attributes" or "Expand all record attributes".

2.6 Calculating Times and Costs

The assessment of business process models enables you to calculate average values for the following criteria:

- **Execution time**
- **Waiting time**
- **Resting time**
- **Transport time**
- **Cycle time**
- **Costs**

Hint: The names of the criteria listed above correspond to the names used in the ADOxx Standard Application Library. Customising enables you to employ other names and calculate other values. If necessary, please contact your ADOxx administrator or your ADOxx consultant.

If you wish to assess a business process with respect to time and costs, activate the modelling window of the business process first and then select the option "Time and costs" from the "Assessment" menu. Alternatively, you can start the assessment by clicking on the smart icon in the quick-access bar.

As soon as you have selected the option or clicked on the smart icon, the assessment is started according to the parameters set. A status window informs you about the current status of the process.

You can end the assessment by clicking on the "Cancel" button.

When the assessment is completed, the results are shown in the window "<BP- model name>: Results" (see fig. 276) in the ADOxx browser (see chap. 8., p. 66).

2.6.1 Settings

Business Process Models are assessed with respect to times and costs according to the following parameters:

- **Number of Simulations:** Measure for the accuracy of the results; the higher the number of Simulations, the more exact the results of the assessment will be.
- **Working days per year:** This value is necessary (together with the value "Hours per working day") to convert real time into company time.
- **Hours per working day:** This value is necessary (together with the value "Working days per year") to convert real time into company time (see chap. 4.1.3.1, p. 335).
- **Input parameters:** The input parameters determine which attributes are used as input for the assessment. Input parameters are defined during customising. The ADOxx standard application library provides you with the input parameters "**Standard**" and "**Without resting and transport time**".

You may change the assessment settings by selecting the option "Settings" from the "Assessment" menu. After selecting this menu item the window "Time and costs - settings" opens up (see fig. 275).

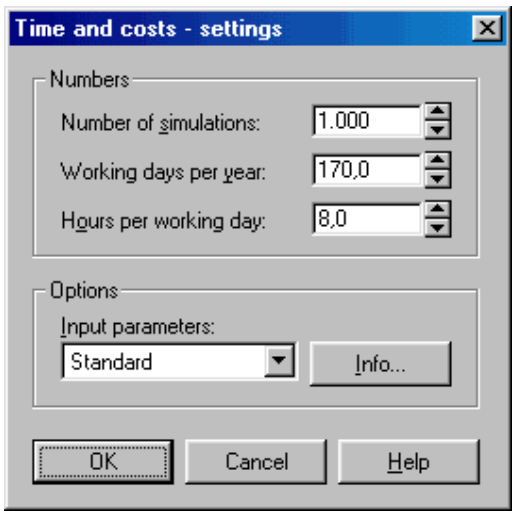


Figure 275: Assessment settings

Click on the OK button to close the window and save the changes.

Clicking on the "Cancel" button closes the window and discards any changes.

2.6.2 Results

The results of an assessment of a Business Process Model with regard to times and costs are shown in the ADOxx browser (see chap. 8., p. 66) (see fig. 276).

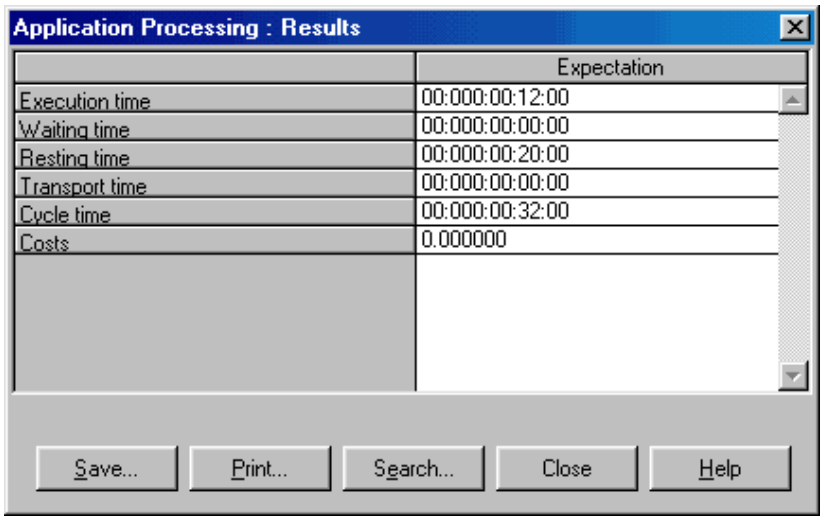


Figure 276: Assessment results

You can save the results to a file by clicking on the button **"Save"** or print them out by clicking on the button **"Print"**.

Close the window "<BP model name>: Results" by clicking on the "Cancel" button.

3. Analysis

The Analysis Component allows you to carry out static evaluation on your ADOxx models (the Simulation component (see chap. 4., p. 332) allows the possibility of dynamic evaluation).

In "Queries / Reports (see chap. 3.1, p. 306)" you can analyse the contents of your models (objects, connectors and their attributes), which contain specific defined criteria. The results of these queries can be displayed either in graphical or tabular form, can be saved in different formats (including TXT, CSV, RTF HTML or ACR) and can be printed.

Additionally - depending on the application library you are using - it is possible to execute "predefined queries" (see chap. 3.2, p. 318) and create "Relation tables" (see chap. 3.3, p. 322). The relation tables display the relations between two classes of a particular model type. The results of the "Predefined queries" and "Relation tables" can also be saved in various formats and printed to a connected printer.

The availability of the "Analysis" component and of the options "Queries/Reports" (see chap. 3.1, p. 306), "Predefined queries" (see chap. 3.2, p. 318), "Relation tables" (see chap. 3.3, p. 322) and "Analytical Evaluation" (see chap. 3.4, p. 325) depend on the current configuration (see chap. 4., p. 729) of the Modelling Toolkit.

Activate the "Analysis" component by clicking on the smart icon in the component list.

You can also activate the Analysis Component, by clicking with the right mouse button in the component list (beside the component smart icons) and then selecting the menu item "Analysis" from the popup menu which is displayed. This popup menu can also be displayed by using the function key <F9> and the Analysis Component can be selected by pressing the key <a>.

Once the Analysis Component has been activated, the quick-access bar contains the following smart icons:

"Queries/Reports" (see chap. 3.1, p. 306)

"Predefined queries" (see chap. 3.2, p. 318)

"Relation tables" (see chap. 3.3, p. 322)

Analytical Evaluation (see chap. 3.4, p. 325)

"Models in the database" (see chap. 4.1, p. 56)

"Opened model windows" (see chap. 4.2, p. 57)

"Previous model window" (see chap. 2.3.1, p. 255)

"Next model window" (see chap. 2.3.1, p. 255)

Hint: The appearance of the smart icons in the quick-access bar depends on the particular configuration of the quick-access bar (see chap. 4., p. 54) and the current configuration (see chap. 4., p. 729) of the Modelling Toolkit.

3.1 Queries/Reports

ADOxx offers the following types of queries/reports:

- Standardised queries (see chap. 3.1.2, p. 308)
- User defined queries (see chap. 3.1.3, p. 312)

When carrying out a **standardised query**, you only need to supply the missing parameters ("text with gaps") **without** having to enter an AQL expression.

A **user-defined query** is carried out by combining standardised queries or using AQL expressions (see chap. 14., p. 790), which can be formulated directly.

3.1.1 Model Selection

If you wish to carry out standardised (see chap. 3.1.2, p. 308) or user defined (see chap. 3.1.3, p. 312) queries on your models, select the option "Queries/Reports" from the "Analysis" menu, or click on the appropriate smart icon in the quick-access bar.

The window "Queries - model selection" (see fig. 277) opens up, showing all available models stored in the ADOxx database.

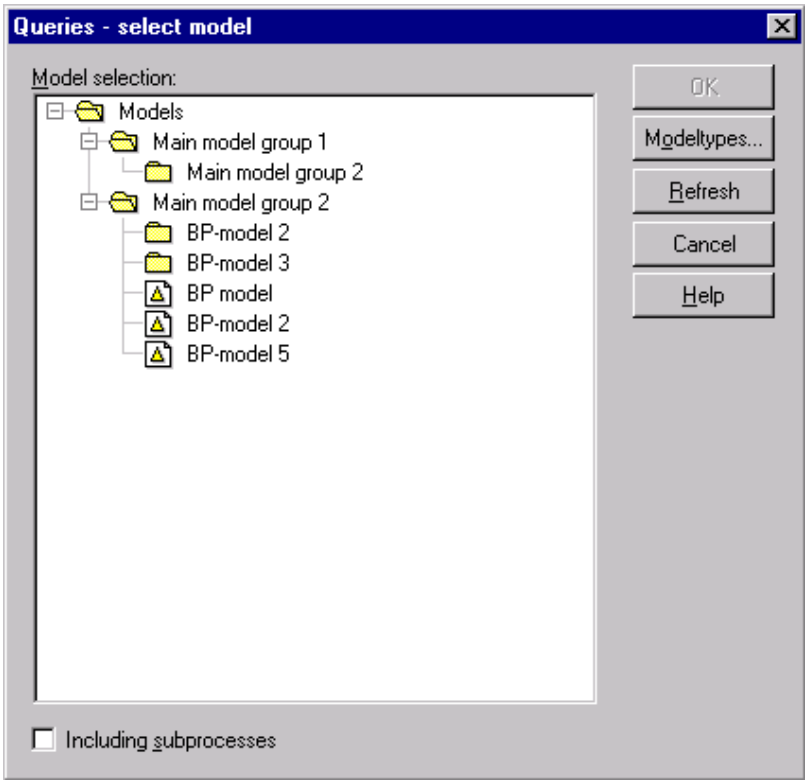


Figure 277: Queries - Model selection

Select the models on which the queries are to be carried out and then click on the OK button to continue the process.

ATTENTION: All models selected must be based on the same library (business process or Working Environment library).

Activating the **"Including referenced models"** option will specify whether the queries should also include the models referenced by the selected models. If including referenced processes automatically, the models selected will be regarded as main processes. Within the representation of results, referenced processes are displayed indented according to their position in the process hierarchy.

Hint: If the option **"Including referenced processes"** is activated, you can change the reference settings (see chap. 11.1, p. 97) by clicking on the "References" button.

Then click on the OK button, to carry out standardised (see chap. 3.1.2, p. 308) or user defined queries (see chap. 3.1.3, p. 312) .

3.1.2 Carrying out Standardised Queries

Once all models to be queried have been loaded, the window "Queries" is displayed (see fig. 278).

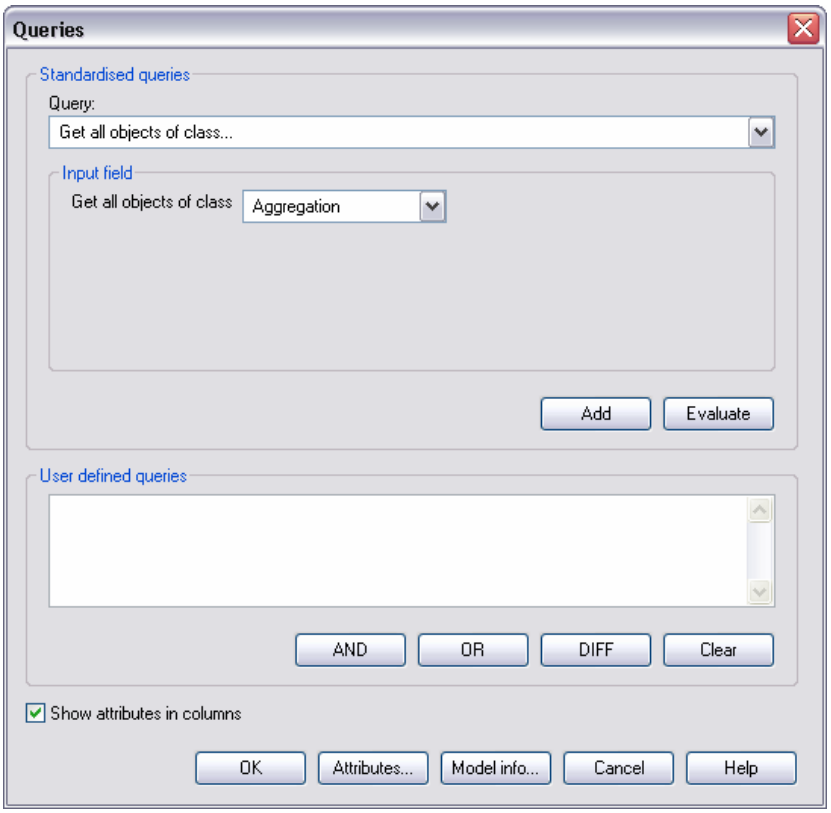


Figure 278: Standardised and user specific queries

To carry out standardised queries select one of the standardised queries in the "Queries" list (see fig. 279). Then an appropriate text with gaps will appear in the input field of the selected query. Enter the required information (e.g. select the object/attribute name) or select the appropriate values in the class, attributes or comparison operators lists.

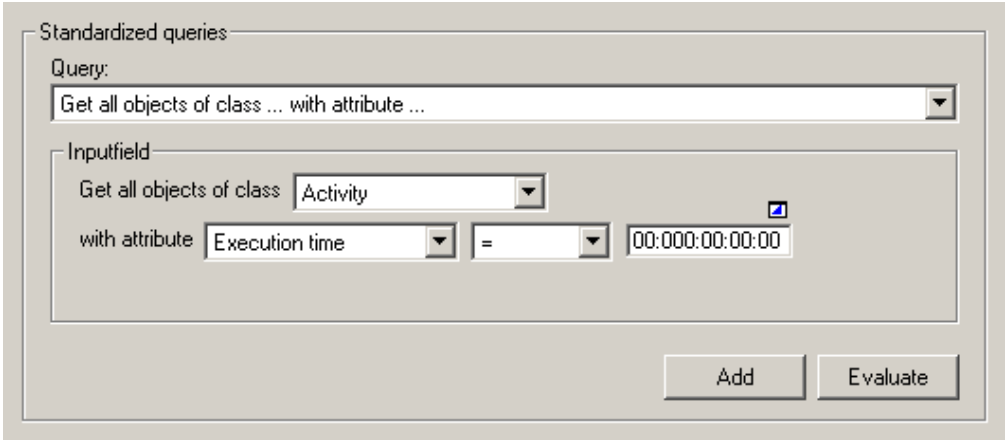


Figure 279: Standardised queries (part of the "Queries" window)

In the following, the standardised queries are explained:

- **Get all objects of the class** [*selection*]**.**

  This query displays all objects of the selected class.

  Example: By selecting the class "Activity" you will obtain all activities in the models selected.

- **Get all object of the class** [*selection*] **with**

  attribute [*selection*][*operator*][*value*]**.**

  This query displays all objects of the selected class, with specified attribute value.

  Note: Select the class first, then the attribute of the class and the operator and then enter the attribute value.

  Example: the class "Activity" with the attribute "Costs", the operator ">" the value "1000" display all objects of the class "Activity" with costs that are greater than 1000.

- **Get all object of the class** [*selection*] **with number of rows**

  in the table attribute [*selection*][*operator*][*value*]**.**

  This query displays all the object of the selected class, that contain the number of rows (defined by the user) in the table attribute.

  Note: This query will only be displayed if table attributes are defined in the assigned application library.

  Example: By selecting the class "Process start" with the "process responsibility" table attribute, the ">" operator and the input value "2", the system will display all objects of the class "Process start" in the table "process responsibility" with more than two rows.

- **Get all the objects of the class** [*selection*] **with**

  table attribute [*selection*] **and with the column**
  [*selection*][*operator*][*value*]**.**

  This query displays all objects of the selected class, that contain the specified attribute value.

  Note: This query will only be displayed if table attributes are defined in the assigned application library.

  Note: This query will only provide the result if the table attribute contains at least one row, and the search condition is fulfilled. For example: if you wish to find all objects, that contain a value (Wildcard "*"), in the column of the table attribute, then the only objects that will be displayed are those which have at least one defined row.

  Example: Selecting the "Process start" class with the table attribute "process responsibility" and the "classification" column, the "=" operator and the input value "on behalf of" will display all the object of the class "Process start", that contain the value "on behalf of" in the "classification" column of the "process responsibility" class.

- **Get object** [*object name*] **of the class** [*selection*]**.**

  This query allows to verify in what models given objects occur.

  Example: By the instruction of the class "Activity" and the object "edit application form" the system will detect the activity "edit application form" within the previously selected models.

- **Get all the objects, that are connected with the object** [*Object name*] **of the class** [*selection*]
  by the relation [*selection*] **.**

  Using this query you can follow defined relations.

  Note: The name of the object provided by the user must exactly match the name of the object in the model (capital and small letters are distinguished). The input from wildcards is not allowed.

Example: The selection of the object "find worker" of the class "role" and relation "has role" will display all searched persons related to that role in the selected model.

- **Get all connectors of the class** [*selection*]**.**

  This query displays all the connectors of the selected relation.

  Example: By selecting the relation "subsequent", all the subsequent-connectors in the selected model will be reported.

- **Get all the connectors of the relation** [*selection*] **with attribute** [*selection*][*operator*][*value*]**.**

  This query shows all connectors of the relation type selected, which contain the specified attribute value.

  Note: First select the relation, then the attribute of the relation and the operator and finally enter the comparison value of the attribute.

  Example: The query: select the relation "Subsequent" with the attribute "Transition probability", the operator ">" and the input value "0.5", displays all connectors of the type "Subsequent" with transition probability greater than 0.5 (50%).

  Hint: Instead of **[*object name*]** the window "Query" shows an input field into which you may enter the name of an existing object.

  Instead of **[*value*]** there is a selection field into which you may enter a text.

  Instead of **[*number*]** there is a selection field from which you may choose a number .

  Instead of **[*selection*]** there is a selection field from which you may select a class, an attribute of a class or a relation.

  Instead of **[*operator*]** there is a selection field from which you may choose a number.

You can perform the finalised standardised query by clicking on the **"Evaluate"** button. The Query result (see chap. 3.1.4, p. 313) will be displayed in the ADOxx browser.

Click **"Add"** to change the standardised query into an AQL expression (see chap. 14., p. 790) and display it in the **"user defined queries"** (see fig. 282) field. Using the standardised query you can create a (more complex) user defined query (see chap. 3.1.3, p. 312) by entering further AQL expressions and adding further standardised queries

By clicking on the **"Attributes"** (see chap. 3.1.2.1, p. 311) button the user can define the attributes, which should be displayed in addition to the name of the object/connector found. Clicking on the **"Model info"** (see chap. 3.1.2.2, p. 311) button allows the user to see models on which queries are based.

Hint: Standardised queries are saved once they have been carried out. When calling the option "Queries" again, the standardised query will be displayed as a user defined query (see chap. 3.1.3, p. 312) (i.e. you will see the AQL expression). Such a query will be maintained until the user enters another standardised or user-defined query or exits ADOxx.

Hint: If an AQL expression is entered in the field **"User-defined queries"**, you can carry this query out by clicking on the OK button. Otherwise, the currently selected standardised query will be executed.

Using the option **"Show attribute in columns"** the user can determine how the query results (see chap. 3.1.4, p. 313) should be displayed. If this option is activated, the objects/connectors found, as well as the models will be displayed on rows and the attributes in columns (see fig. 284).

If this option is deactivated, the query result will always be shown within the five following columns (see fig. 283):

- model

- object/connector name
- class or relation
- attribute
- value

3.1.2.1 Select Attributes for Result Display

You can specify the attributes which you would like to be displayed for any objects or connectors found by the query by clicking on the button "Attributes".

Hint: If you do not select any attributes, the query results are listed **without** attributes.

Clicking on the button "Attributes" opens the window "Queries - attribute settings" (see fig. 280).

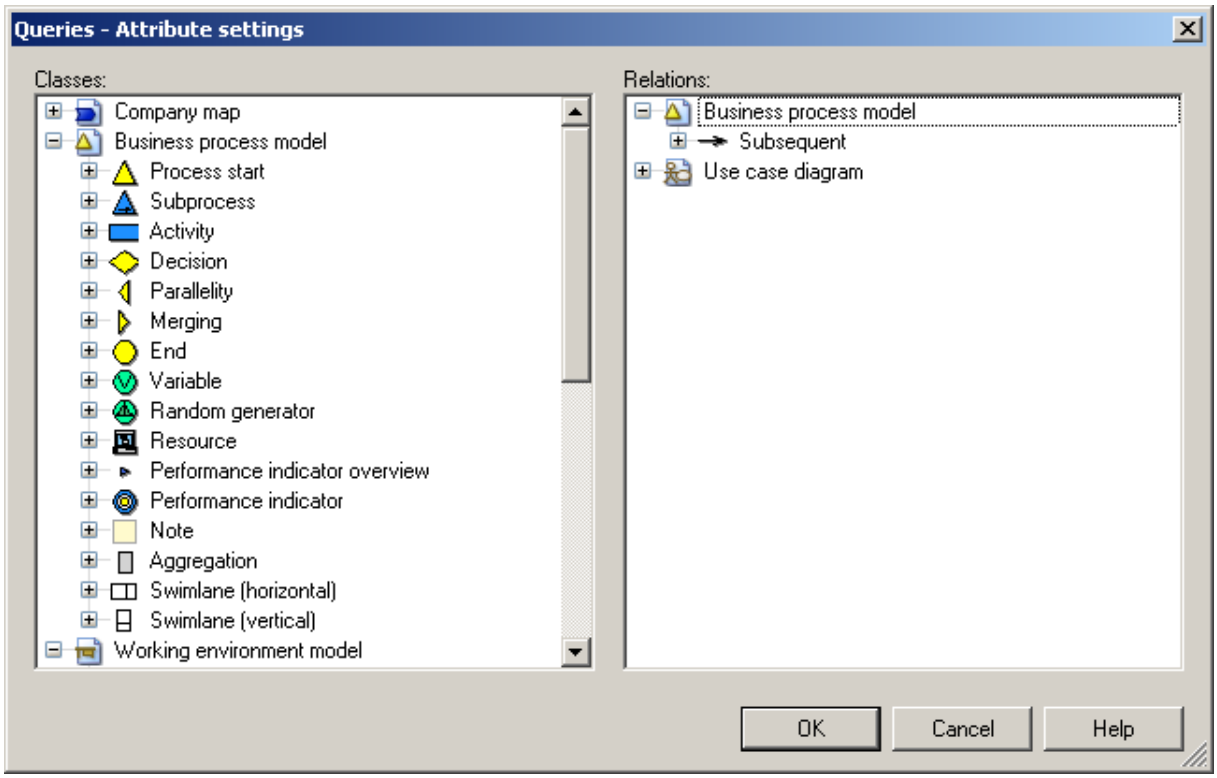


Figure 280: Query - Attribute settings

Select one or more attributes from the "Attributes" list. The attributes selected will be displayed in the ADOxx browser (see chap. 8., p. 66) as part of the query result.

Hint: The attribute selection will be saved until the user selects other or additional attributes or quits the system.

3.1.2.2 Model Information

In the window "Queries - model information" (see fig. 281) all models selected for querying (plus their referenced models, if applicable) are shown.

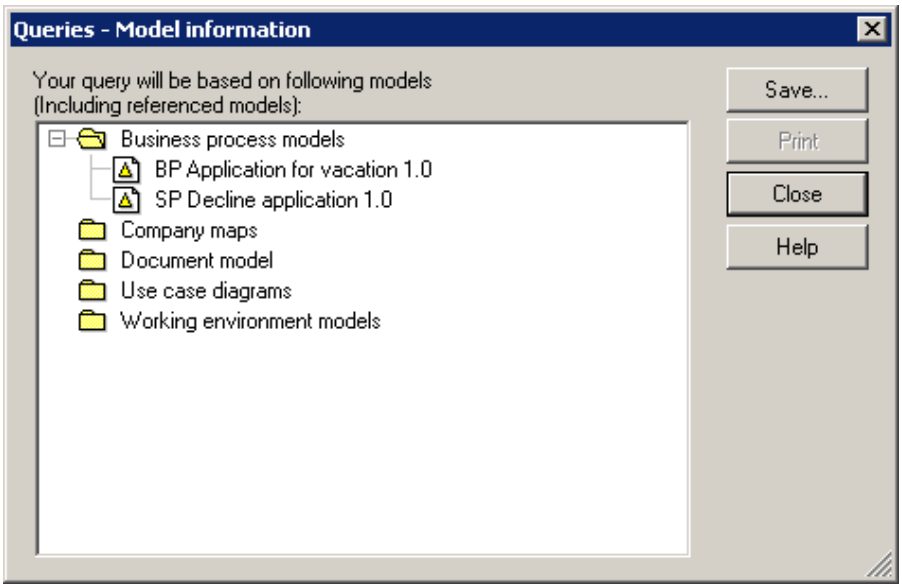


Figure 281: Show all selected (and referenced) models

You can save the contents of the window to a file.

3.1.3 Carrying out User Defined Queries

As soon as all models have been loaded, the "Queries" window is displayed (see fig. 278).

You can formulate user-defined queries in the field **"User defined queries"** (see fig. 282), by combining standardised queries, defining queries in the query language AQL (see chap. 14., p. 790) or expanding standardised queries with AQL expressions.

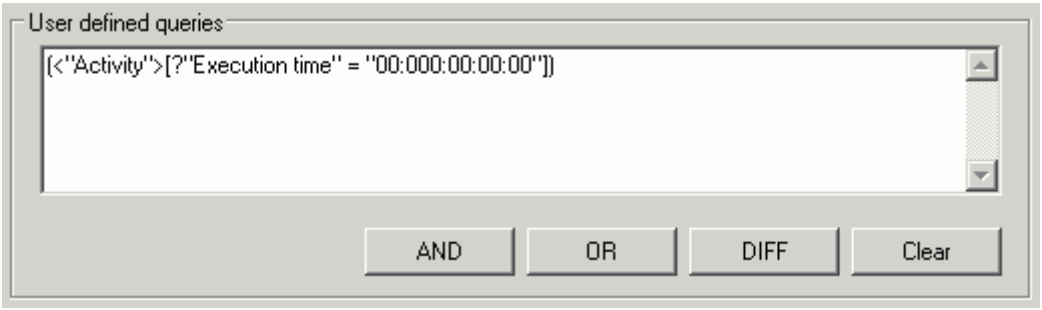


Figure 282: User defined queries (Part of the "Queries" window)

You may also define user-defined queries by combining standardised queries.

Define a standardised query and click on the "Add" button. This generates an AQL expression, which corresponds to your standardised query. This AQL expression is displayed in the field **"User-defined queries"**.

You can combine or connect queries by using the buttons **"AND"**, **"OR"** and **"DIFF"**. Delete the contents of the field **"User-defined queries"** by clicking on the button **"Clear"**.

You can define the attributes that you want to be displayed in addition to the names of the objects or connectors found by clicking on the button "Attributes" (see chap. 3.1.2.1, p. 311). Clicking on the button "Model info" (see chap. 3.1.2.2, p. 311) shows you on which models the query is based.

Hint: Standardised queries (see chap. 3.1.2, p. 308) are saved once the query has been carried out. When calling the option "Queries" again, the standardised query will be shown as a user defined query (see chap. 3.1.3, p. 312) (i.e. you will see the AQL expression). This query will be maintained until you either enter another standardised or user-defined query, or until you quit ADOxx.

Hint: Should an AQL expression be entered in the field **"User-defined queries"**, you can carry this query out by clicking on the OK button. Otherwise, the currently selected standardised query will be executed.

The option **"Show attributes in columns"** allows to define how the query results (see chap. 3.1.4, p. 313) will be displayed. If this option is activated, the objects/connectors found, as well as the models they belong to will be displayed in rows and their attributes in columns (see fig. 284).

If this option is deactivated, the query results will always be shown within these five columns (see fig. 283):

- model
- object/connector name
- class or relation
- attribute
- value

3.1.4 Query Results

The results of the queries are displayed in the ADOxx browser (see chap. 8., p. 66) in the "Query results: <*Search criteria*>" window (see fig. 283).

Hint: The query formulated in the window "Queries" (see fig. 278) will be displayed instead of the <*Search criteria*>, i.e. either the text of the standardised query or the AQL expression of the user-defined query.

Hint: The display configuration depends on whether the **"Show attributes in columns"** option in the window "Query" is activated or not (see fig. 278).

If the option "Including sub processes" was activated during model selection (see chap. 3.1.1, p. 307), the business process models are numbered according to their hierarchy. The first business process is numbered "1.", its direct sub processes are "1.1.", "1.2.", "1.3." etc. The sub processes of model "1.2." will also be numbered accordingly as "1.2.1.", "1.2.2." and so forth. Only the first call of a model, which calls itself directly or indirectly is part of the query result. For any further call, you will find a reference to the first occurrence such as "see. 1.2.1.".

Bei deaktivierter **Option "Attribute in Spalten darstellen"** im Fenster "Abfragen" (see fig. 278), werden stets fünf Spalten ausgegeben (see fig. 283). Die Spalte "Modell" enthält alle Modelle, auf welchen die Abfrage durchgeführt wurde. Jedes Modell wird in der Zeilengliederung (see chap. 8.2, p. 68) numeriert.

If the **"Show attributes in columns"** (see fig. 278) option in the "Queries" window is deactivated, then the same five columns will always be displayed (see fig. 283). Of these, the column "model" contains all models on which the query has been carried out. Each model will be numbered in the row structure (see chap. 8.2, p. 68).

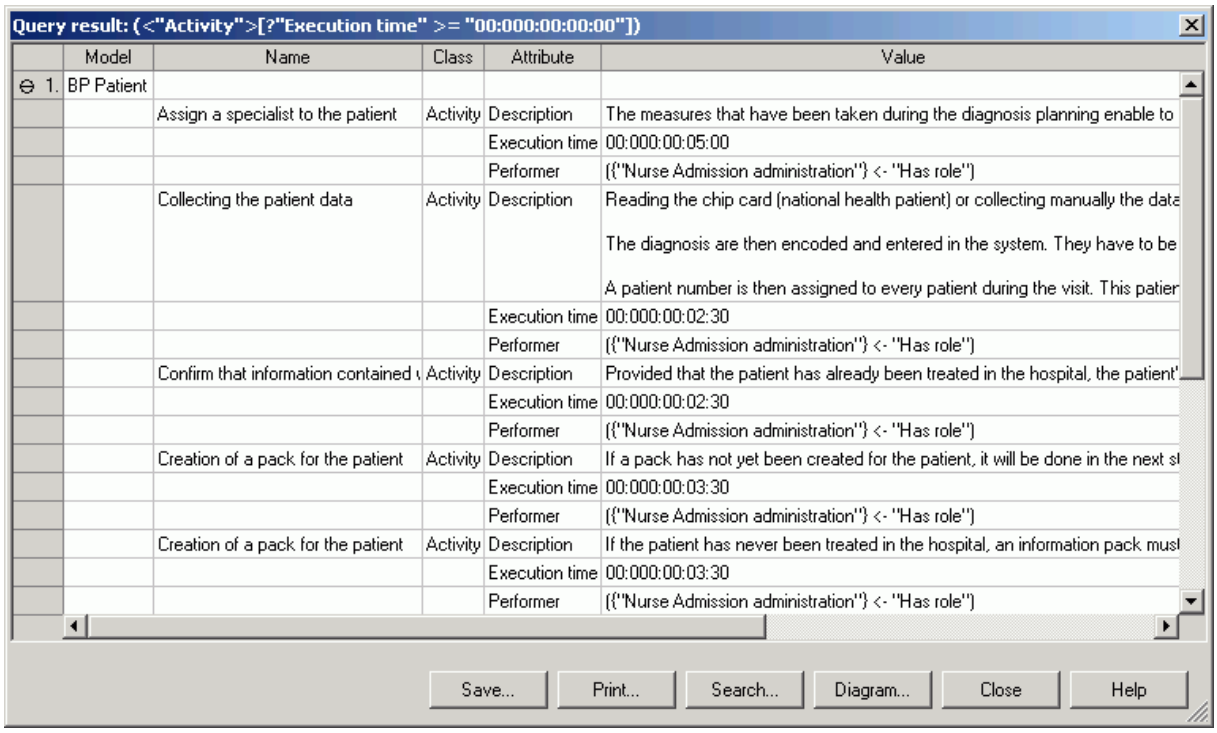


Figure 283: Query results ("Show attributes in columns" option deactivated)

The column "name" lists the name for each object found and the column "class" the names of the appropriate class . If querying relations, this column displays the start and target object of the connector found, as well as the name of the relation. If you have selected attributes for the results display (see chap. 3.1.2.1, p. 311), the values of these attributes will also be shown.

If, in the window "Queries", the option **"Show attributes in columns"** (see fig. 278) is activated, then the results will be arranged so that there is a line for each object or connector found and the attributes are shown in the columns (see fig. 284).

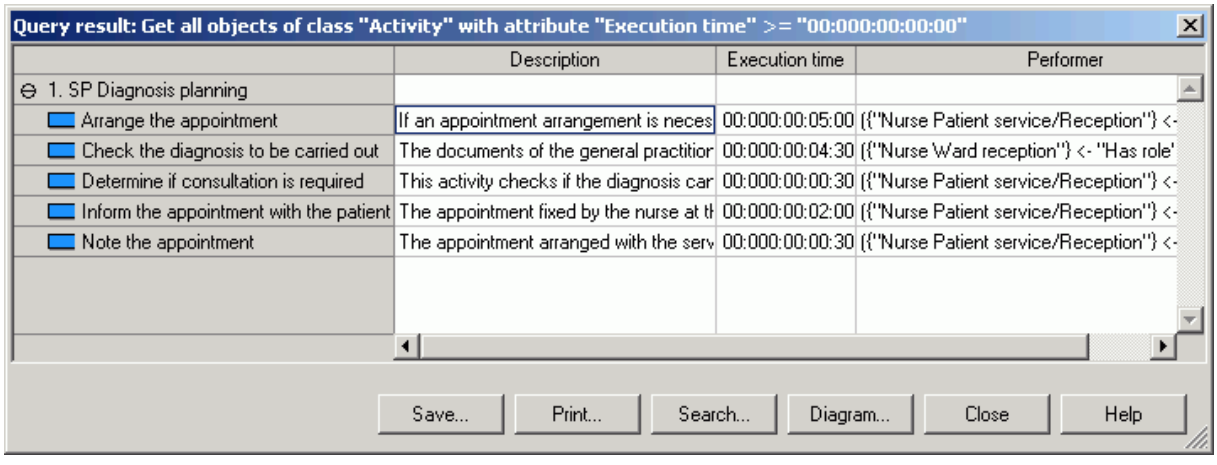


Figure 284: Query result ("Show attributes in columns" option activated)

The object name (or the name of the start and target objects for connectors) is displayed in the first column, as well as an icon, which represents the object class (or the relation of the connector) .

Hint: Displaying table attributes (see chap. 3.1.4.4, p. 317) expands the representation in the ADOxx browser to columns (by adding table attributes) and rows (by adding the table contents).

Select the the "Select attributes" menu item from the context menu (right mouse button) to display the attributes to be changed (see chap. 3.1.4.3, p. 316).

You can save the query result displayed to a file, print it out or display it as a diagram. Moreover, the search for specific content is also possible.

Weiters können die Attributwerte von Objekten bzw. Konnektoren jener Modelle, die geladen sind, direkt im Abfrageergebnis-Browser bearbeitet werden (see chap. 3.1.4.1, p. 315).

Furthermore, the attribute values of the objects/connectors attribute values of the loaded models can be directly edited in the query result browser (see chap. 3.1.4.1, p. 315).

Using the context menu (right mouse button), it is also possible to sort (see chap. 3.1.4.2, p. 315) the displayed objects and connectors according to specific attribute values.

3.1.4.1 Edit Attribute Values

After executing a query you can edit the attribute values in the query result browser in the Analysis Component.

Hint: If you want to edit the attribute values of objects or connectors which are query results, the models containing these objects or connectors must be loaded before you execute the query. Should you try to edit attribute values of a model that is currently not opened, a query will be shown with possibility to open the respective model.

Edit the attribute values in the query result browser by double clicking on the corresponding cell or by selecting in the context menu (right mouse button) the "Edit attributes" menu item and then - depending on the type of attribute - for attributes of the types "integer", "float" and "text", enter the value directly in the cell or for attributes of every other type enter values in the support dialogue window (see chap. 2.2.12, p. 201)

3.1.4.2 Sort Query Result

You can sort the objects and connectors according to specific attribute values via the popup-menu (right mouse button), menu item "Sort".

Once the "Sort" option has been selected, the window "Queries - select sorting criteria" (see fig. 285) is shown.

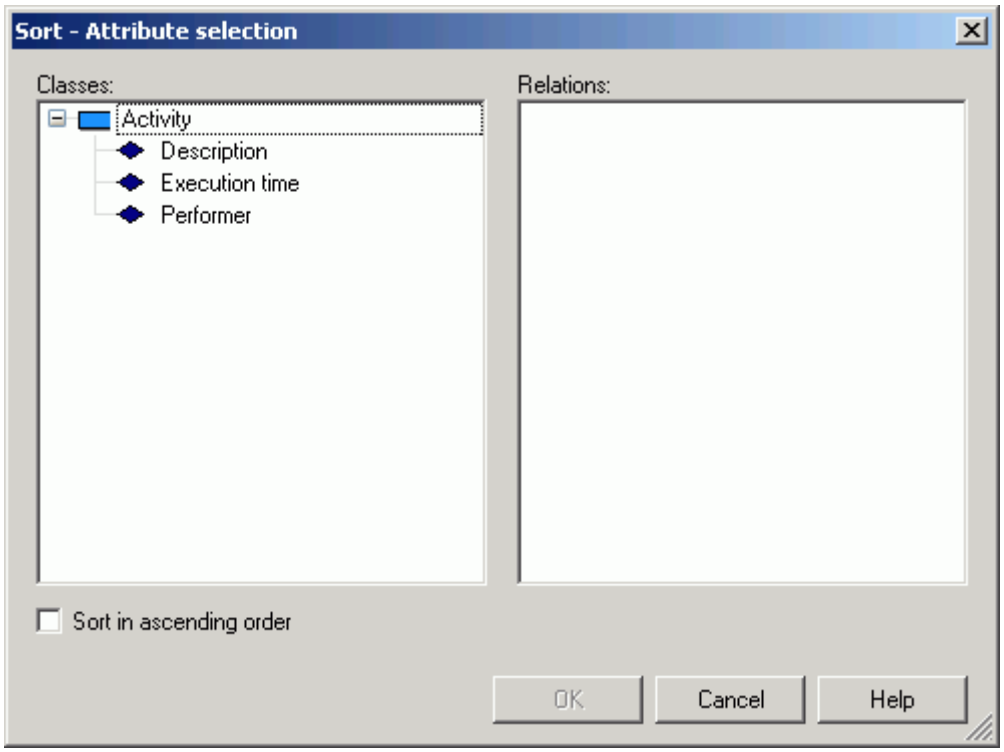


Figure 285: Sort - Attribute selection

Select an attribute from the list "Attribute". The results will then be sorted according to this attribute. The option **"Sort in ascending order"** allows you to choose between ascending and descending order.

Hint: The results will be sorted in a model-related way, i.e. the query result will be sorted using the selected attributes within the result models.

3.1.4.3 Select Attributes

By the result display of the attributes in columns (see fig. 284) you can change the representation of attributes directly in the ADOxx browser by clicking on the "Select attributes" option in the context menu (right mouse button).

The window "Queries - Attribute column selection" (see fig. 286) shows all the classes or relations of the objects or connectors that are contained in the query result. The attributes of these objects or connectors, which have already been selected for the display (see chap. 3.1.2.1, p. 311) before carrying out the query, are highlighted.

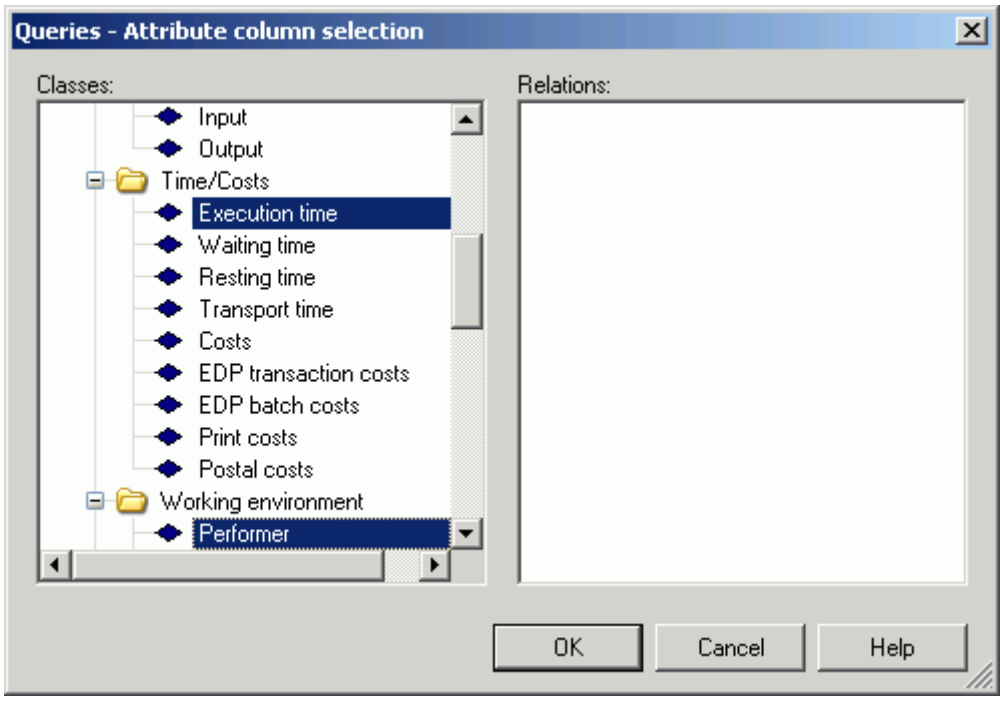


Figure 286: Sort - Attribute selection

Select the additional or other attributes to be displayed and then click on the OK button, to show the selected attributes in the ADOxx browser.

3.1.4.4 Display Record Attributes

With the resulting display of attributes (see fig. 284) it is possible to show Record attributes in the ADOxx browser through the symbol ⊕. In the picture below (see fig. 287) you can see the query result on a Working Environment model, for which the "Training" table, the "Hourly wages" and "Availability" have been selected.

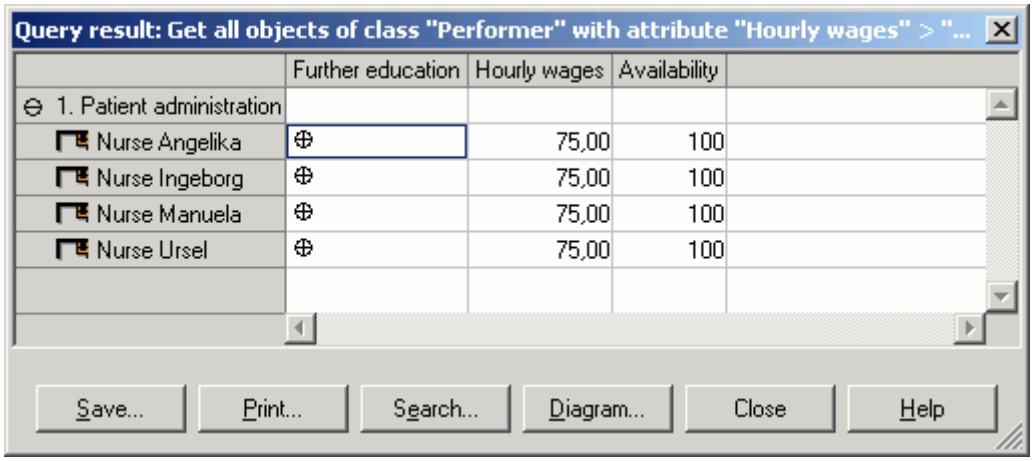


Figure 287: Record attributes faded out in the query result browser

⊕ symbolises, that it is a short display of the table and that the record attributes can be displayed by clicking.

Once you have clicked the cell containing the symbol ⊕ the table display will be expanded: the columns will be expanded to all defined record attributes and table rows will be added to the display (see fig. 288).

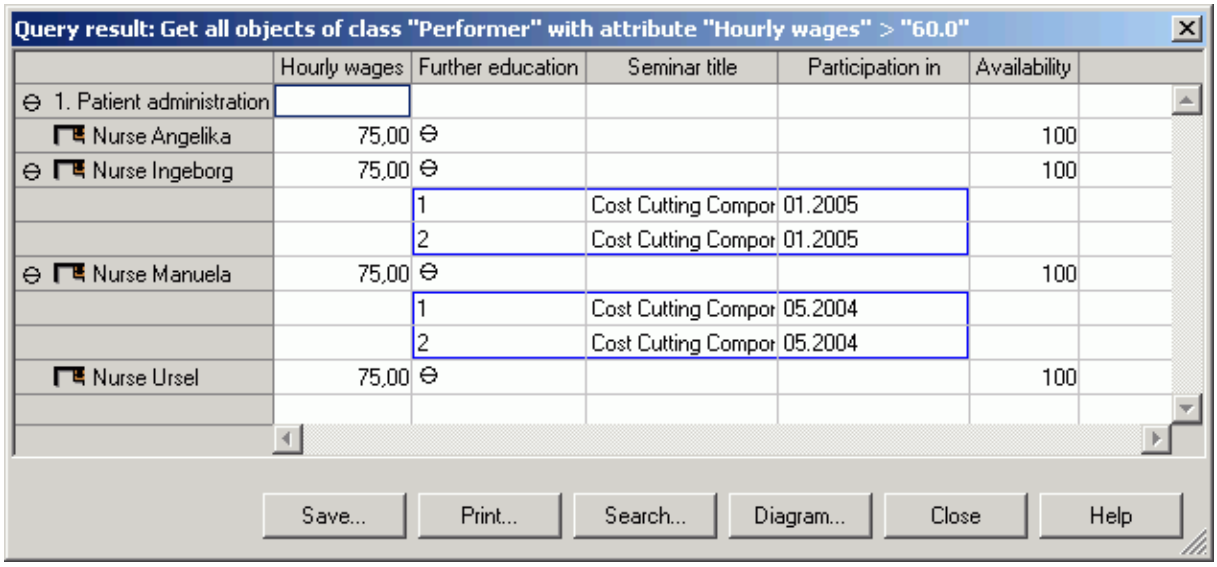


Figure 288: Display of record attributes in the result query browser

Hint: In the expanded display of a table, all the record attributes of an object will be outlined in blue.

The symbol ⊖ signalise that all recorded attributes will be displayed. Clicking on it will show a short display of the table again.

To **shrink or expand all record attributes in the query result**, select in the context menu of ADOxx browser (right mouse button) the "Shrink all" or the "Expand all" option.

3.2 Predefined Queries and Plans

Depending on the definition of your application library, you may execute certain predefined queries or generate predefined plans.

Predefined plans are a special type of output results. Predefined plans generate tables with empty columns. Predefined plans may, for example, be used for defining task plans.

ATTENTION: Predefined queries and plans must be defined in the application library and can be changed or expanded by the ADOxx administrator if needed.

Hint: The following pictures refer to the predefined queries and plans of the **ADOxx standard application library (see chap. 16., p. 802)**. A description of the queries in the application library is to be found in the user documentation volume III "ADOxx' standard method manual" or in the online document "ADOxx standard method " ("Help" menu).

You can find the predefined queries and plans in the Analysis component in the "Analysis" menu between the options "Queries/Reports" and "Calculation". When selecting one of these menu entries a submenu appears, listing the predefined queries for the respective model types.

3.2.1 Model Selection

If you wish to execute predefined queries (see chap. 3.2.2, p. 319) on your models, select the option "Queries on *<model type>*" from the "Analysis" menu and choose your desired query from the appearing submenu. Alternatively, you can select the smart icon from the quick-access bar and select your query from the appearing pulldown menu.

The window "Predefined queries on *<model type>* - Model selection" (see fig. 289) is displayed. This window contains a list of all available models saved in the ADOxx database.

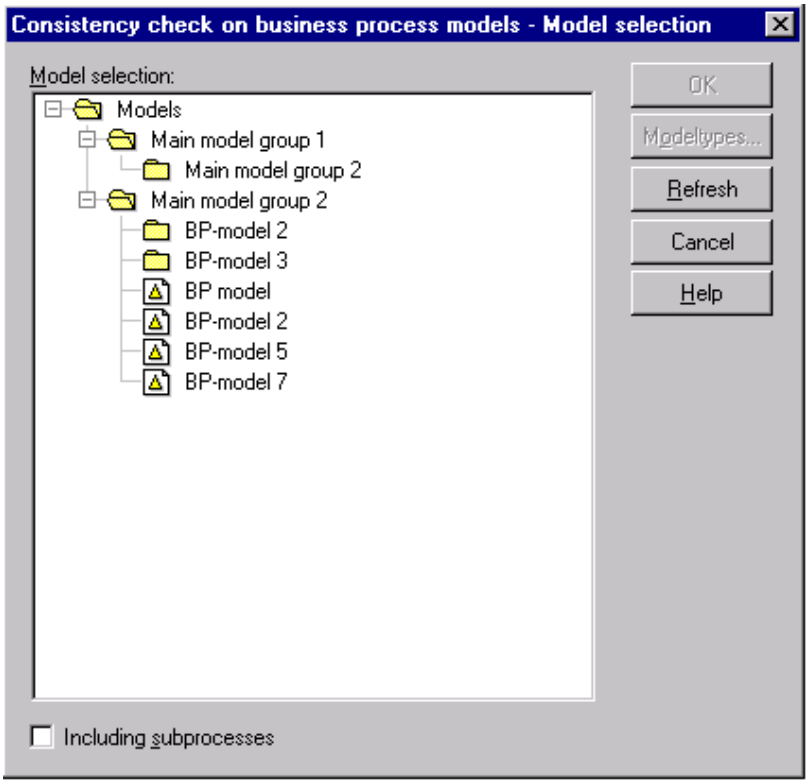


Figure 289: Predefined queries - Model selection (example)

Select the models on which the query shall be executed and then click on the OK button to execute the predefined queries (see chap. 3.2.2, p. 319).

With the **"Including referenced models"** option you can determine whether the queries shall include the models referenced by the selected models or not. If this option is activated then the selected models will be considered as main models, i.e. in the result representation, the referenced models called by these main models will be indented according to their position in the hierarchy.

3.2.2 Executing Predefined Queries

Select (and if necessary complete) a predefined query in the "*<Predefined query>* - Queries" window (see fig. 290).

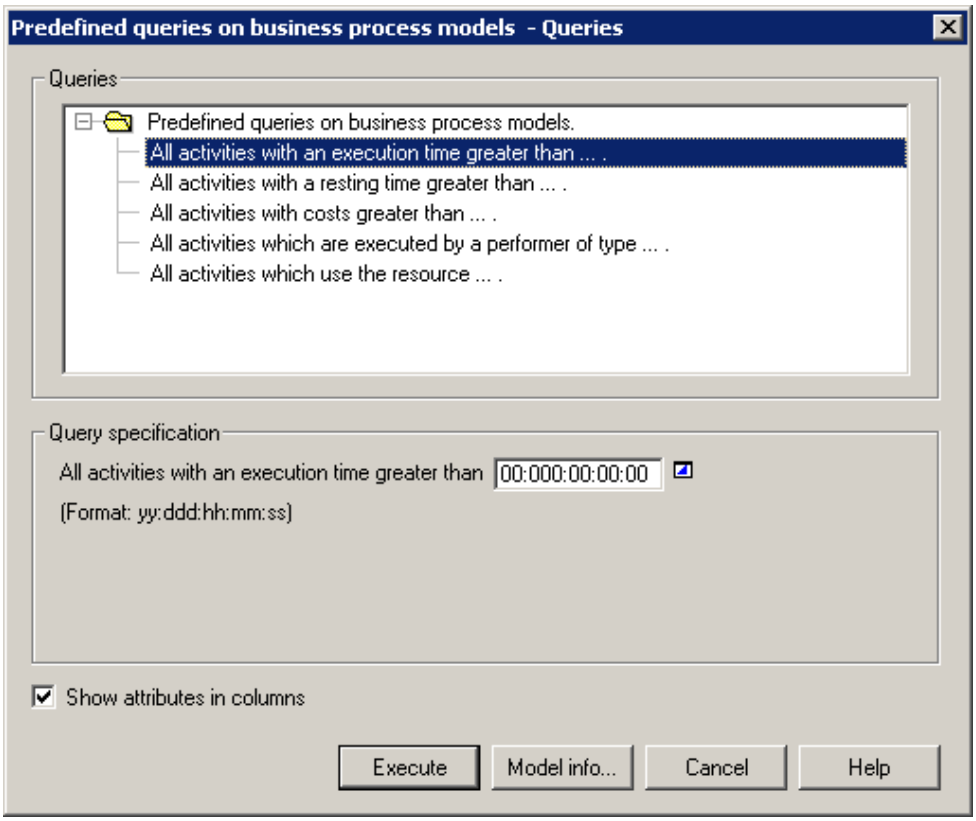


Figure 290: Execute predefined queries (example from the ADOxx-Default-Library

Zur Durchführung vordefinierter Abfragen wählen Sie eine der vordefinierten Abfragen durch Auswahl aus der Liste "Abfragen" aus. Danach erscheint u.U. im Eingabefeld der der gewählten Abfrage entsprechende Lückentext. Geben Sie die notwendigen Informationen (z.B. Name des Objekts) ein bzw. wählen Sie aus der Liste der Klassen, Attribute oder Vergleichsoperatoren die gewünschten Werte aus. Klicken Sie danach auf den **Button "Ausführen"**, um die gewählte Abfrage auszuwerten.

For this, select a predefined query from the "Queries" list. Depending on the query, one or more input fields may appear (as represented by "..." in the query name). Enter the necessary information (e.g. name of the object) or select required values and operators from lists provided. Afterwards, click **"Ececute"** to carry out the query selected.

Clicking on the button "Model info" shows you on which models the query is based.

The results of the predefined queries (see chap. 3.2.3, p. 321) and predefined plans (see chap. 3.2.4, p. 321) are displayed in the ADOxx browser (see chap. 8., p. 66).

Using the **"Show attributes in columns" option** you can determine how the result of the predefined queries (see chap. 3.2.3, p. 321) will be displayed. If this option is activated, the objects or connectors and the models found will be dispayed on rows and the attributes in columns (see fig. 284).

If this option is deactivated, the following five columns will always be displayed in the result (see fig. 283):

- Model
- Object or connector name
- Class or relation
- Attribute
- value

3.2.3 Results

The results of the queries are displayed in the window "Query results - <*query name*>" in a ADOxx browser (see chap. 8., p. 66) similar to the results of the queries/reports (see chap. 3.1.4, p. 313) functions.

You can save the results shown to a file; print them out or display them as a diagram. You may also search for specific contents.

Moreover the attribute values of objects or connectors of the models, which are loaded, can be directly edited in the query result browser (see chap. 3.1.4.1, p. 315).

In addition, you can sort (see chap. 3.1.4.2, p. 315) the objects and connectors according to specific attribute values via the popup menu (right mouse button).

3.2.4 Results of Predefined Plans

The result of a predefined plan is - unlike the result of a predefined query (see chap. 3.2.3, p. 321) - an empty table. This table is shown in the ADOxx browser (see chap. 8., p. 66) in the window "Plan generation: <*predefined plan name*>" (see fig. 291).

Hint: The **"Show attributes in columns" option** is not available in predefined plans.

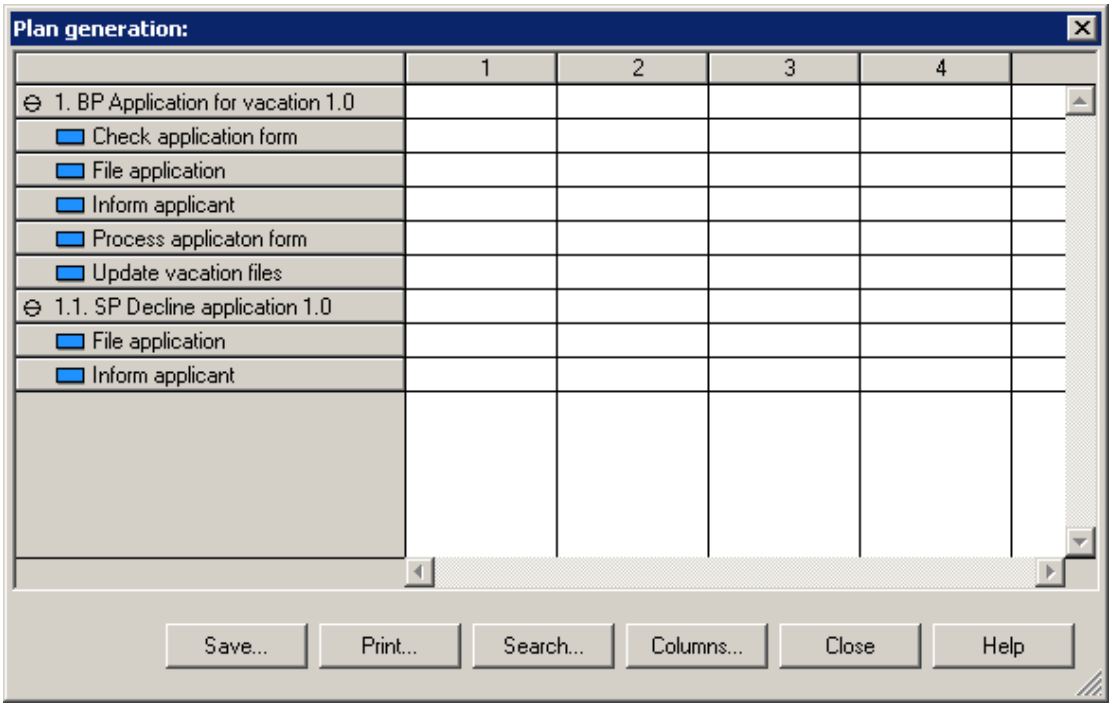


Figure 291: Predefined plan (example)

You can define column titles by clicking on the button "Columns" (see chap. 3.2.4.1, p. 322).

You may save the generated plan to a file or print it out.

3.2.4.1 Change column titles of predefined plans

In the window "Change column titles" (see fig. 292) you can change the column headers generated for the plan. Enter a name for the column into the field "Column names" and then click on the "Change" button.

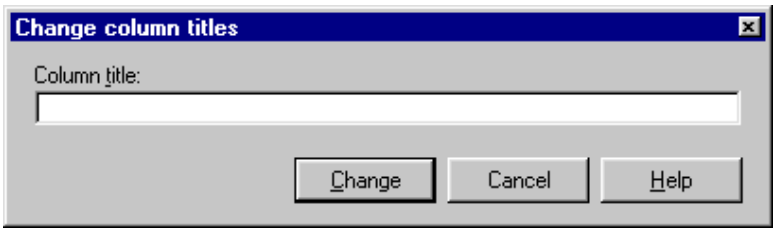


Figure 292: Change column titles of plans

The text entered will be transferred to the table and a column number will be added to it.

3.3 Relation Tables

The system can generate various, predefined relation tables ("relation matrices"), depending on the configuration of your application library.

These relation tables show the existence of relations. In this case, "relation" defines connectors (within a model) as well as inter-model references.

A relation table shows, which start objects (e.g. performer) are linked to which target object (e.g. role).

Hint: Since there is also the possibility to evaluate the relations of inter-model references, the displayed start and target objects can also be contained in different models and in models of different types.

ATTENTION: The ADOxx administrator must redefine relation tables during the configuration of the libraries.

Hint: The following pictures refer to the predefined relation tables in the ADOxx standard application library (see chap. 16., p. 802).A description of the relation tables in the ADOxx-Default-Library is to be found in the user documentation volume III "ADOxx' standard method manual" or in the online document "ADOxx' standard method" ("Help" menu).

Hint: If there are no tables defined in your application library, the text "No relation tables defined" appears when opening the option "Relation tables".

3.3.1 Model Selection

In order to create a relation table, select the option of the appropriate table from the menu "Relation tables". Alternatively, you can click on the smart icon in the quick-access bar and select a relation table from the popup menu, which appears.

The window "<*relation table*> - model selection" is shown.

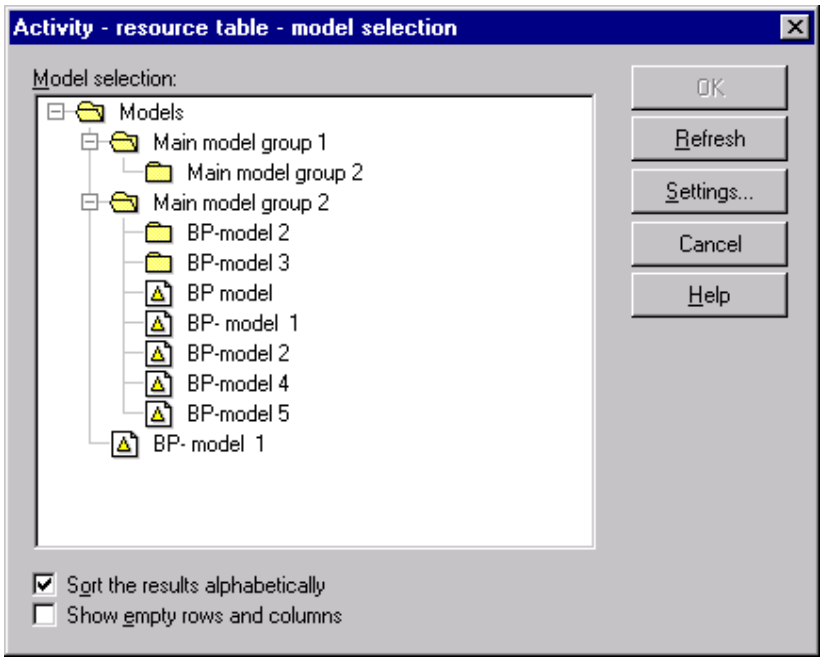


Figure 293: Relation table - Model selection

Select from the list of the models stored in the ADOxx' database the models for which you want to create the appropriate table (i.e. which contains the start objects) and click on the OK button to create and show (see chap. 3.3.2, p. 325) the corresponding table.

Click on the **"Settings"** button to define the display options and the restrictions for the representation (see chap. 3.3.1.1, p. 324).

Activating the option **"Sort the results alphabetically"** shows the start and target objects in the result table in alphabetical order.

> **Hint:** You can also display another attribute other than the names. Should several objects then have the same attribute value, the information will be summed up.

The standard representation of relation table is in "compact form", i.e. only the objects between which there is really a relation are contained in the table. Activate the **"Show empty rows and columns"** option to display also the start and target objects to which no relation is connected (see fig. 297).

> **Hint:** The display of a relation table depends on the existence of relations (connectors or inter-model references) and on the activating of the "Show empty rows and columns" option. The table below shows which conditions must be met to display a relation table (see Table 1).

"Show empty rows and columns"	**Existing connectors**	**Existing inter model references**	**Display possible**
Yes	Yes	-	***Yes***
Yes	No	-	***Yes***
Yes	-	Yes	***Yes***
Yes	-	No	***No***
No	Yes	-	***Yes***
No	No	-	***No***

No	-	Yes	***Yes***
No	-	No	***No***

Table 1: Conditions to display a relation table

3.3.1.1 Settings

You can limit the objects or relations displayed by entering selection criteria. Click on the **"Settings"** button. The window "<*relation table*> - Settings" (see fig. 294) appears.

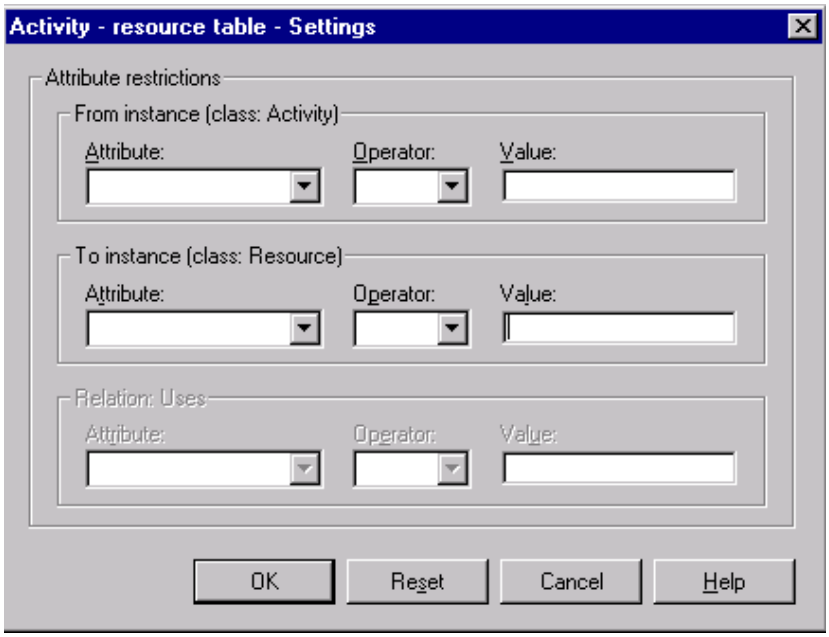


Figure 294: Relation table - Define attribute restrictions

Enter the required restriction for the start object, the target object and/or the relation displayed.

> **Hint:** If no attributes are defined for the start or the target object class, then the corresponding input area is not visible.

When defining filters the operators >, >=, <, <=, =, !=, like or unlike may be used.

For example, if an activity resource table is to contain only those activities whose names contain the word "perform", you must enter the following filter (see fig. 295).

Figure 295: Example of filter (part of the window "Settings")

> **Hint:** A filter is defined only when both an attribute and an operator have been selected for the object or connector! Empty filters are not regarded as restrictions.

If you wish to use the filter(s) entered when displaying the table, click on the OK button. The settings will then be saved and will be displayed when you select the button **"Settings"** again.

> **Hint:** The settings remain saved until you enter new settings, delete the existing settings by clicking on the button **"Reset"** or select another relation table.

3.3.2 Display

The relation table generated is displayed in the ADOxx, browser (see chap. 8., p. 66) in the window "<*relation table*> for model <*model name*>" (see fig. 296).

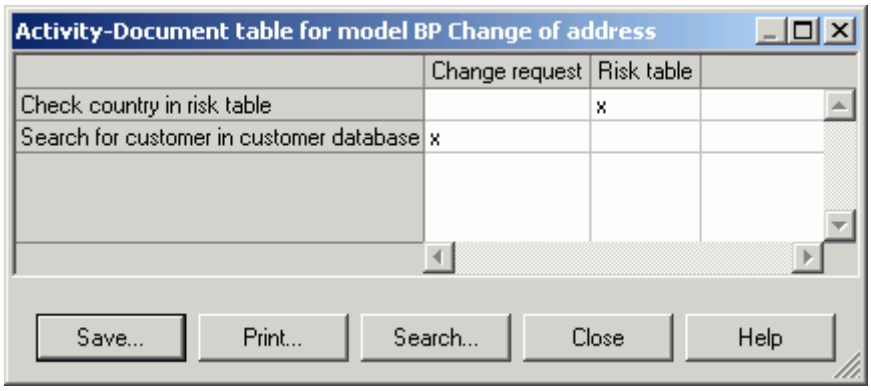


Figure 296: Realtion table (Shrink empty rows and colums)

The first column (shaded gray) shows the start objects, the first row (also shaded grey) shows the target objects. An existing connector between a start and a target object is indicated - depending on the definition in your application library - either by an "x" (standard value) or by the value of a certain attribute of the relation. An entry in a cell means that the start object in that row is connected with the target object in the corresponding column.

Hint: In the picture above the relation table shows the standard representation in "compact form", i.e. empty rows and columns will be shrunk (see fig. 296). You can have a complete representation of the relation table including empty rows and columns - as it is shown in the picture below - (see fig. 297) by activating the "Show empty rows and columns" option in the <*Relation table*> - Model selection" window (see fig. 293).

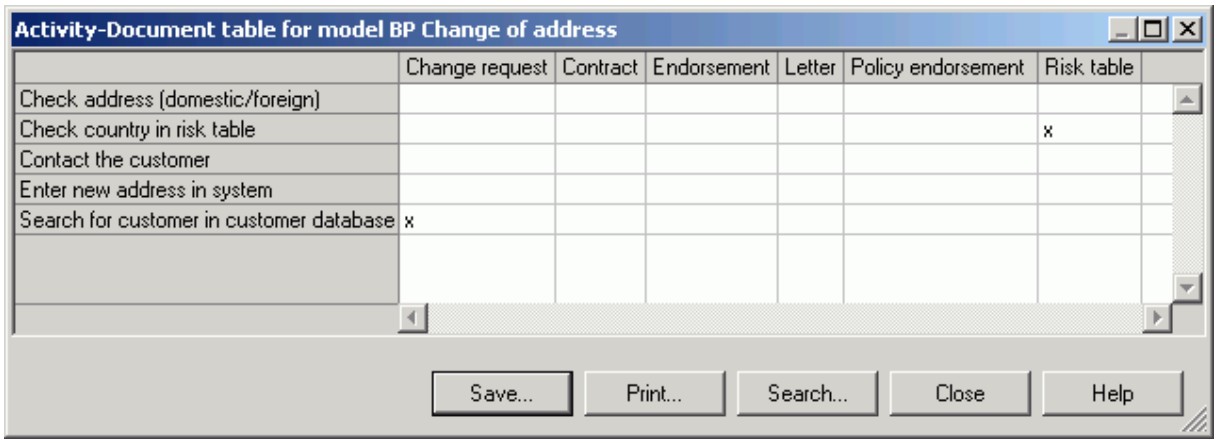


Figure 297: Relation table (Show empty rows and columns)

You may save the relation table to a file, print it out on the standard printer or search for a specific piece of text within the table. To close the displayed table, click on the close button

3.4 Analytical Evaluation

Analytical Evaluation enables you to carry out a static evaluation of your business process models. The working environment is not taken into account.

Analytical Evaluation (see chap. 3.4.1, p. 326) calculates the average times and costs of the business processes in question.

Hint: Note the information about the input (see chap. 3.4.4.1, p. 329) and the functioning (see chap. 3.4.4.2, p. 330) of the Analytical Evaluation.

3.4.1 Processing

Carry out an Analytical Evaluation by selecting the option "Analytical Evaluation" from the "Analysis" menu. Alternatively, you may click on the smart-icon in the quick-access bar.

Select the business process model to be evaluated from the "Model selection" list in the "Analytical Evaluation" (see fig. 298) window.

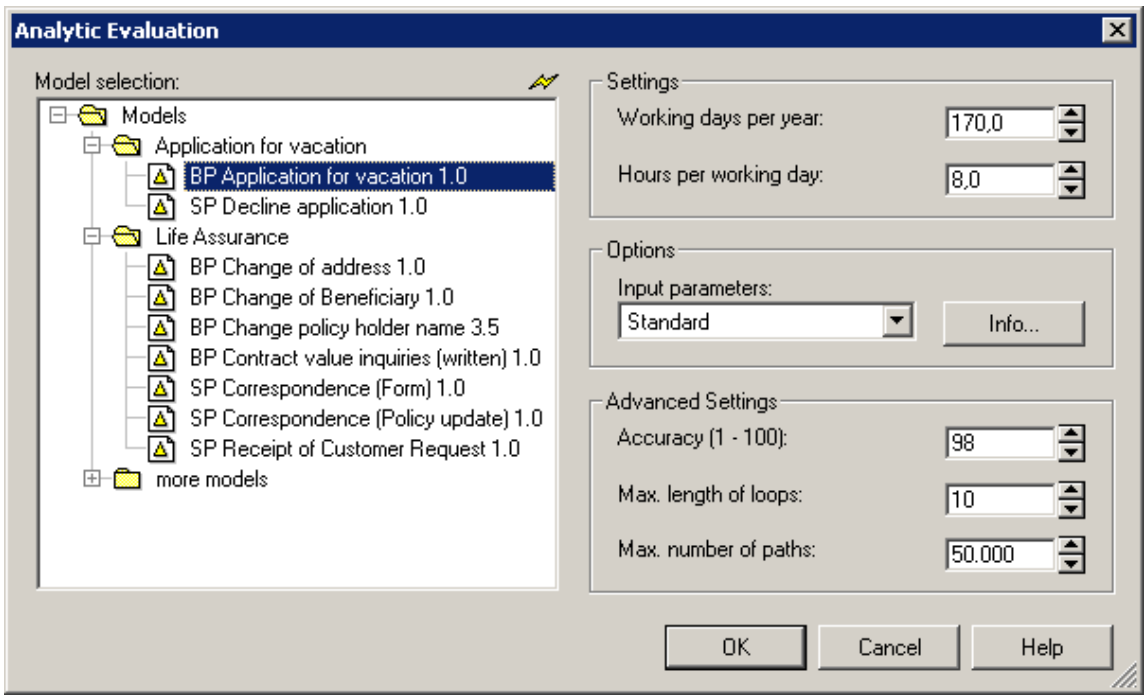


Figure 298: Analytical evaluation - Model selection and settings

Additionally, the following settings may also be defined:

"Working days per year": This value serves the definition of company time (together with the value "Hours per working day").

"Hours per working day": This value serves the definition of company time (together with the value "Working days per year").

Hint: The term "Company time" is explained in the chapter "Time concepts in Simulation" (see chap. 4.1.3, p. 334).

"Input parameters": Select an **"Input parameter"** for the Analytical Evaluation. The "Input parameter" determines which attributes serve as input for the Analytical Evaluation and which evaluation results are calculated. An explanation for the "Input parameter" selected can be seen by clicking on the "Info" button.

Hint: It is possible to define additional "Input parameters". If you require any additional "Input parameter" choices, please contact your ADOxx administrator.

"Accuracy": The parameter "Accuracy" determines how accurately the Evaluation results for complex branchings and loops in the business process model are approximated. You can

select any number between 1 and 100 (standard value: 98). A higher number means a higher calculation accuracy as well as a longer calculation time.

"Max. length of loops": This parameter serves to avoid infinite loops. The maximum loop length defines how many modelling objects may occur within a process loop as a maximum. The loop entry is not included.

Hint: Always enter the loop length of the longest loop of the business process to be evaluated (or a higher number).

"Max. number of paths": This parameter determines at which number of different paths the Analytical Evaluation will be aborted. When the Analytical evaluation is stopped because the number defined has been reached, you will not receive any results.

Hint: When evaluating business process models which do not contain loops, the parameters "Accuracy" and "Max. length of loops" are ignored.

3.4.2 Results

The Analytical Evaluation provides the following results (see chap. 3.4.3, p. 327):

- (average) **Frequency (the frequency with which each activity occurs)**
- (average) **Execution time**
- (average) **Waiting time**
- (average) **Resting time**
- (average) **Transport time**
- (average) **Costs**
- (average) **Execution time in working days**
- (average) **Cycle time.**

Hint: While customising ADOxx, various **"Input parameters"** can be defined. A specific "Input parameter" is selected at the start of each Analytical Evaluation. It defines which results are calculated. Other results than those listed above can be calculated by specifying different "Input parameters". If you need additional "Input parameters", please contact your ADOxx administrator.

After the Analytical Evaluation has run, the company time and the volume of processes (see chap. 3.4.3, p. 327) on which the results are based can be altered. This enables you to view results based on different scenarios.

3.4.3 Display Results

After the Analytical Evaluation has been completed, the window "Analytical Evaluation - results: <*model name*>" (see fig. 299) appears.

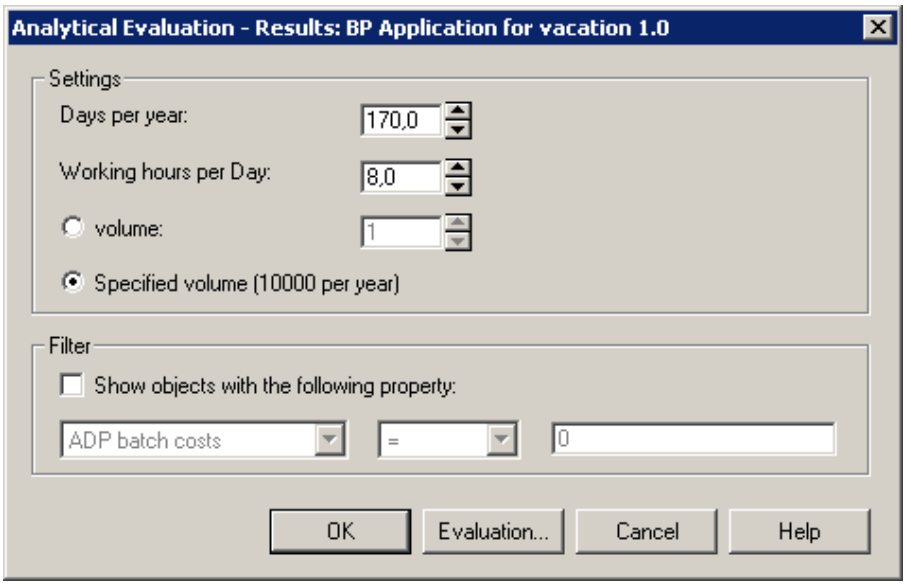


Figure 299: Analytic Evaluation- Settings for the result display

You can choose from the following display options:

"Days per year": Serves the definition of the company time (together with "Working hours per day"). Changing the value displayed causes all results to be adjusted accordingly. The company time defined before the Analytical Evaluation is run is used as the standard setting.

"Working hours per day": Serves the definition of the company time (together with "Days per year"). Changing the value displayed causes all results to be adjusted accordingly. The company time defined before the Analytical Evaluation is run is used as the standard setting.

Hint: When the company time is altered, the results of the Analytical Evaluation are only adjusted according to the new company time. The results are not re-calculated.

"Volume": Enter a value for the volume in order to multiply the results by this value.

"Specified volume": Select the "Specified volume" in order to multiply the results by the value entered in the "Process start" object.

"Filter": Activating the option **"Show objects with the following property"** acts as a filter on the results displayed - i.e. only those activities which match the criteria entered will be displayed.

Select an attribute (left selection field), an operator (centre selection field) and an attribute value (right selection field).

Click on the OK button to display the Evaluation results (see chap. 3.4.3.1, p. 329).

Click on the button **"Evaluation"** to store the results back into the model. Afterwards, evaluation queries (see chap. 5.2, p. 438) can be executed on this data.

Hint: If a filter has been defined, the filter is taken into account when the results are saved, i.e. results will be saved for objects (activities) meeting the requirements of the filter. For objects that do not correspond to the filter, no results will be saved. The results are saved in the start object according to the filter. In order to avoid inconsistencies, any results stored from previous Evaluations will be deleted in **all** objects of the model evaluated before the current results are saved (including sub processes, regardless of the filter).

3.4.3.1 Display Results

The results of the Analytical Evaluation are displayed in a tabular format in the ADOxx browser (see chap. 8., p. 66). The header of the ADOxx browser contains the company time and the volume on which the results shown are based.

The ADOxx browser contains the following columns for Analytical Evaluation results:

Business process: This column shows the name of the business process model and the sub processes.

Activity: This column contains the names of the activities.

Frequency: This column shows the frequency of occurrence of the activities evaluated. This result is only displayed for activities and not for business processes. The frequency is displayed with an accuracy of two decimal places.

If the evaluated business processes contain loops, the expected occurrence frequency is approximated. The higher the calculation accuracy, the more accurate the expected occurrence frequencies will be approximated.

Times and costs: These results are displayed for the activities in the model as well as for the business process itself.

Depending on the "Input parameters" selected for the Analytical Evaluation, additional result columns may be displayed.

If you have defined limitations for the display of the activities, only the results of the activities meeting the requirements of the filter defined will be contained in the aggregated results.

Execution time in working days: This result is shown as a single result for each activity as well as an aggregated result for the business process. It shows the execution time in working days.

ATTENTION: The aggregated results for the evaluated main process also contain the results of the sub processes called. The aggregated results for the sub processes, on the other hand, contain only the activity results of the respective sub process and **not** the results of further referenced sub processes.

Durchlaufzeit: Die Durchlaufzeit wird für den ausgewerteten Geschäftsprozeß, sowie für alle Subprozesse ermittelt.

Hint: Die Durchlaufzeit beinhaltet auch die Dauer der aufgerufenen Subprozesse.

Cycle time: The cycle time is calculated for the business process evaluated as well as for all sub processes.

Hint: The cycle time displayed for the main process also includes the cycle time of all referenced sub processes.

3.4.4 Background Information on the Analytical Evaluation

3.4.4.1 Input for the Analytical Evaluation

Input for the Analytical Evaluation is a Business Process Model. The Business Process Model, as well as any sub processes it calls, must be correctly modelled. In particular, the transition probability, which is set within the subsequent relations following a decision must be set.

In order to run the Analytical Evaluation function, a **transition probability** must be defined for each connector leading away from a decision (in contrast to the simulation, for which transition conditions are defined).(It is also possible to define a transition probability in a connector leading away from a parallelity object. This indicates a conditional probability).

Hint: The total sum of all transition probabilities of the connectors, which lead away from a decision must always be 1.

The model clipping in the picture below (see fig. 300) compares the transition probabilities ("0.8" and "0.2") to the transition conditions ("authorisation='YES' and "authorisation='NO') as an example.

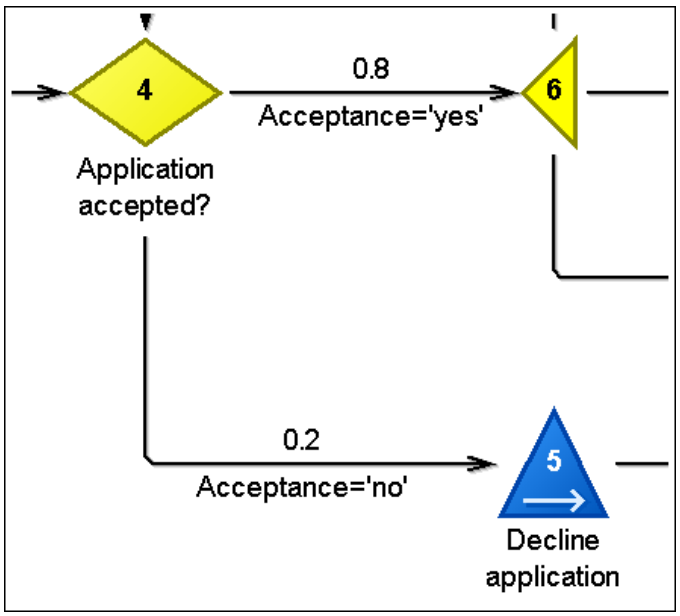


Figure 300: Analytical Evaluation - Example of correct transition probability

Hint: The Analytical Evaluation function cannot evaluate variables or transition conditions.

3.4.4.2 Functioning of the Analytical Evaluation

The Analytical Evaluation analyses all possible paths of a Business Process Modelincluding sub processes and calculates their probability. On the basis of the times and costs stated, provisional results for each path and subsequently the whole process (see chap. 3.4.2, p. 327) are calculated.

ATTENTION: In the Analytical Evaluation, a "global end" (see chap. 4.1.5, p. 340) appearing in the process will not be taken into account, i.e. a global end will be treated as "local end"!

Treatment of parallel paths

As the number of provisional results rises exponentially with the number of parallel paths, you can specify a limit by defining the parameter **"Max number of paths"** (standard value: 50,000) in order to avoid a memory overload. This danger usually only occurs if parallel paths contain a lot of loops. The Analytical Evaluation is aborted as soon as the value defined is reached.

Hint: The definition of the maximum number of paths mainly depends on the memory capacity of your computer.

Treatment of loops

When a Business Process Model contains loops the results are calculated approximately. The parameter **"Accuracy"** (standard value: 98) defines how accurately the provisional result of the loop's path is calculated. As soon as the accuracy defined is reached, the Analysis of the loop ends and the subsequent objects will be carried out.

Additionally, you may specify how many objects may be contained within the loop as a maximum by defining the parameter **"Max length of loops"** (standard value: 10). This enables the system to identify infinite loops. If a loop contains more objects than the number specified, the Analytical Evaluation will be aborted.

Example:

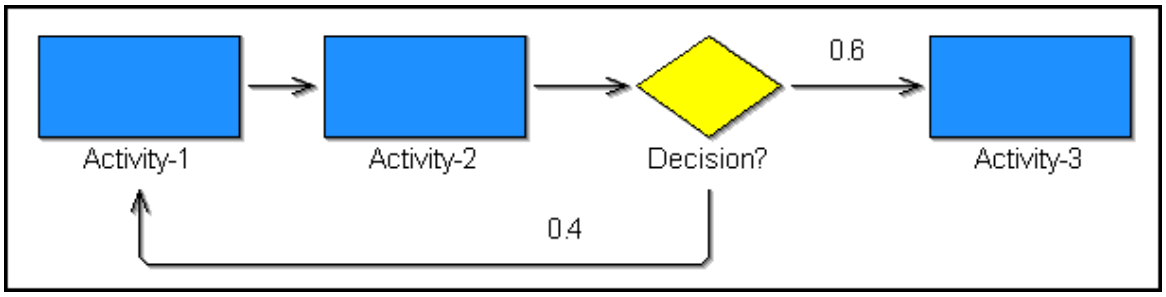


Figure 301: Analytical Evaluation - Example of a loop

In this case (see fig. 301) a maximum loop length of at least 3 must be defined, as the process loop contains three objects ("Activity-2", "Activity-3" and "Decision").

Hint: The loop entry "Activity-1" is not taken into account.

Meaning of the "Accuracy" parameter:

The Analytical Evaluation analyses each possible path through the process separately. Since a process can contain a large number of paths, the results will be approximate, in order to reduce the calculation time. The "Accuracy" parameter controls the precision of this approximation. A higher value in the "Accuracy" parameter means a higher calculation accuracy but also a longer calculation time.

The Analytical Evaluation analyse s all paths through the process and breaks off as soon as a path ends and at the same time the total probability of all paths equals (0.9 + 0.00099999999 * accuracy).

The "Accuracy" parameter is particularly of significance regarding loops in the process. Each loop has a potential to cause an infinity of path, since after each loop running either the process part can be continued or the loop can run again. If the loop runs again (the path increases by 1), the same situation starts after the end of the loops. That is why the number of paths is infinite. Besides loops, branchings also causes additional paths in the process.

© BOC Group 2008

4. Simulation

The availability of the "Simulation" component and the modules "Path analysis" (see chap. 4.2, p. 343), "Capacity analysis" (see chap. 4.3, p. 354), "Workload analysis (steady state)" (see chap. 4.4, p. 367), "Workload analysis (fixed time period)" (see chap. 4.4, p. 367) and Agents (see chap. 4.8, p. 391) depends on the current configuration (see chap. 4., p. 729) of the Modelling Toolkit.

Activate the "Simulation" component by clicking on the smart icon in the component list.

You can also activate the "Simulation" component by clicking with the right mouse button in the component list (near the component smart icons) and then selecting the menu item "Simulation" from the popup menu which is displayed. Alternatively, you can open the popup menu using the function key <F9> and then press the <s> key to activate the simulation component.

After activating the Simulation Component, the following smart icons are available in the quick-access bar:

"Path analysis" (see chap. 4.2, p. 343)

"Capacity analysis" (see chap. 4.3, p. 354)

"Workload analysis (steady state)" (see chap. 4.4, p. 367)

"Workload analysis (fixed time period)" (see chap. 4.4, p. 367)

"Agents" (see chap. 4.8, p. 391)

"Offline animation" (see chap. 4.5, p. 386)

"Delete simulation results" (see chap. 4.7, p. 390)

"Free simulation cache" (see chap. 4.6, p. 388)

"Models in the database" (see chap. 4.1, p. 56)

"Opened model windows" (see chap. 4.2, p. 57)

"Previous model window" (see chap. 2.3.1, p. 255)

"Next model window" (see chap. 2.3.1, p. 255)

"Open model" (see chap. 2.4.2, p. 257)

"Print" (see chap. 2.4.17, p. 288)

"Generate graphics" (see chap. 2.2.7.24, p. 156)

The appearance of the smart icons in the quick-access bar depends on the particular configuration of the quick-access bar as well as the current configuration (see chap. 4., p. 729) of the Modelling Toolkit. You can change the configuration of the quick-access bar by selecting the menu item "quick-access bar" within the "Extras" menu.

4.1 Introduction

Within this section, some general information on ADOxx is found in the following chapters:

- Overview of the simulation algorithms (see chap. 4.1.1, p. 333)
- Evaluation mechanisms (see chap. 4.1.2, p. 334)
- Time concepts in the simulation algorithms (see chap. 4.1.3, p. 334)
- Cooperative vs. non-cooperative activities (see chap. 4.1.4, p. 340)
- Global End (see chap. 4.1.5, p. 340)
- Manual random generator (see chap. 4.1.6, p. 342)
- Time view vs. Process view (see chap. 4.1.7, p. 342)

4.1.1 Overview of the simulation algorithms

By simulating your Business Process Models you can try out potential restructuring measures and examine the effects of such measures from different views.

ADOxx provides you with the following simulation algorithms:

- Path analysis (see chap. 4.2, p. 343)
- Capacity analysis (see chap. 4.3, p. 354)
- Workload analysis (steady state) (see chap. 4.4, p. 367) (with animation)
- Workload analysis (fixed time period) (see chap. 4.4, p. 367) (with animation)

Path analysis allows you to evaluate your Business Process Models without taking the Working Environment (organisational structure) into account, whereas in **capacity analysis** and **workload analysis** there are assignments of performers to the activities. Workload analysis can be run either for a "steady state" or for a "fixed time period".

> **Hint:** The offline animation (see chap. 4.5, p. 386) allows you to re-run simulations which have been saved in protocol (history) files (*.apf).

The difference between **capacity and workload analysis** can be characterised as follows:

"Capacity analysis" (see chap. 4.3, p. 354) determines **how many resources** (performers and machines) are **necessary** for the execution of certain Business Process Models based on how often these Business Process Models are executed in a given period of time.

"Workload analysis" (see chap. 4.4, p. 367) determines the **waiting and cycle times** of business process models and activities based on a given quantity of resources (performers and machines) and their availability as well as the stochastic start rate of the Business Process Models.

The difference **between steady state and fixed time period in the workload analysis** can be characterised as follows:

When executing a **steady state** analysis, a **specific number of process runs** are simulated independent of the time period covered by the execution of the Business Process Models. The results are calculated once a "steady state" has been reached (i.e. once an "average" process run has been completed).

"When executing a **fixed time period** analysis, a **previously determined time period** is simulated independent of how many Business Process Models are executed within this period. The results will relate to the previously defined "Calculation phase".

© BOC Group 2008

4.1.2 Evaluation Mechanisms

The evaluation mechanisms enable you to carry out further evaluation on the simulation results. In order to do this you can **save the simulation results back into the models** and then investigate "non-standard simulation results" using evaluation queries (in the ADOxx Query Language AQL (see chap. 14., p. 790)).

ATTENTION: If any of the models which have just been simulated have been loaded read-only, it will not be possible to save the results back to any of the models.

Reasons:

In a write-protected model, the attribute values of objects in the ADOxx Notebooks may be changed but not saved. Thus, transferring values would cause inconsistencies.

If for example the main process only is loaded in a write-able state, and the subprocesses are only loaded for the simulation and thus are write-protected, transferring the results into the main process and saving them would result in inconsistent overall result values. The subprocesses would still contain the "old" simulation results - since the new results may not be saved in write-protected subprocesses - whereas the simulation attributes of the main process would contain the new results. An analysis of these models would therefore not lead to a sensible result.

Examples of "non-standard simulation results":

- Determine the frequency of occurrence of all those activities with execution time greater than 30 minutes.
- Determine all the processes which have an aggregated execution time of more than 50 days per year.
- Determine all performers which belong to the organisational unit ... and whose capacity is below ....

The descriptions of the simulation algorithms also explain how results can be transferred (see Path Analysis (see chap. 4.2.4.3, p. 350), Capacity Analysis (see chap. 4.3.4.4, p. 362) and Workload Analysis (see chap. 4.4.5.2, p. 375)).

Evaluation queries are carried out in an identical manner for all simulation algorithms. This is described in the Evaluation Component (see chap. 5., p. 433).

With the help of the option "Delete Simulation Results" (see chap. 4.7, p. 390) which is available on the quick-access bar within the Simulation Component, simulation results which have been previously transferred into models can be deleted from the results chapter of the objects within those models.

4.1.3 Time concepts within simulation

When using the Simulation Component you should be aware of the different time concepts which are differentiated between within ADOxx:

- Company time versus real time (see chap. 4.1.3.1, p. 335)
- Activity times (see chap. 4.1.3.2, p. 335)
- Process times (see chap. 4.1.3.3, p. 339)

4.1.3.1 The difference between "Company time" and "Real time"

The fields **"Working days per year"** and **"Hours per working day"** can be found in the start windows of the "Path Analysis" (see chap. 4.2, p. 343) and "Capacity Analysis" (see chap. 4.3, p. 354) algorithms. These two fields enable you to convert real time into your "company time".

> **Hint:** During the ADOxx customising the default values for the fields **"Working days per year"** and **"Hours per working day"** are determined. However, if you need different default values please contact your ADOxx administrator.

Company time means the time during which your Business Process Models are executed.

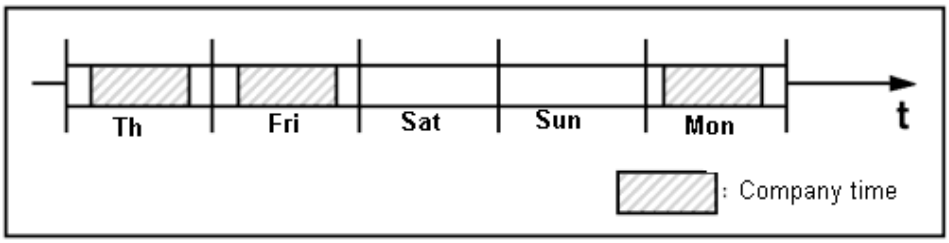


Figure 302: Company time representation

If you for example enter "**220**" into the **field "Working days per year"** and "**8**" into the **field "Hours per working day"**, then one day corresponds to eight hours and one year to 220 days. This means then, that a cycle time of 40 hours will be converted into five working days.

If you enter "**365**" into the field **"Working days per year"** and "**24**" into the field **"Hours per working day"**, ADOxx will calculate in "real time". Please ensure that in this case the times entered into the Business Process Models also refer to "real time".

Example:

Assume an activity with a waiting time (see chap. 4.1.3.2, p. 335) - from the company point of view - of one day exists. This activity is not executed on weekends.

- "**Company Time**": You do not consider the weekend (Saturday and Sunday). In this case, you may enter one day as waiting time. (Attribute Value "00:001:00:00:00")
- "**Real Time**": In this case you should enter 1.4 days as waiting time. (Attribute Value "00:001:09:36:00")

  Reason: 80% of the time, this activity is executed on Tuesday, Wednesday, Thursday or Friday. On these days, the waiting time is 1 day. However 20% of the time, the activity is executed on a Monday where the actual waiting time is 3 days (including the weekend). This results in an average waiting time of 0.8 × 1 + 0.2 × 3 = 1.4 days = 1 day, 9 hours and 36 minutes or - in the ADOxx time format - "00:001:09:36:00".

4.1.3.2 Activity Times

By default, the following activity times - attributes of activities - serve as input for the simulation:

- **Execution Time**
- **Resting Time**
- **Transport Time**
- **Waiting Time**

(Exception: Workload analysis; the waiting time displayed will be the simulation result, i.e. the corresponding activity attribute will not be evaluated)

In the following chapters you will find a description of:

- the standard definition of activity times (see p. 336),
- the input parameters for the simulation (see p. 337),
- an alternative definition of activity times (see p. 337).

Standard definition

Be sure that in the modelling a standardised definition is based to the activity times.

ATTENTION: If different definitions for activity times are used in different models, you can obtain incorrect simulation results.

The selected definitions are also related to the results determined by the simulation algorithms.

The (recommended) standard definition for the single activity time (see fig. 303) is:

Resting Time: This is the time after an activity has been completed by one performer and before it is passed over to the next performer (for example it is lying in an out-tray waiting to be picked up by the internal post).

Transport Time: The time it takes for an activity to be transported from the previous to the next performer. Transport times especially occur - similarly to resting times - when documents must be passed on manually between the performers.

Waiting Time: This is the time when a particular activity is ready to be executed but the performer (or resource) it is waiting for is busy with another activity. Waiting times can also occur when an activity is interrupted and continued at a later point in time.

Execution Time: The time during which the current activity is executed.

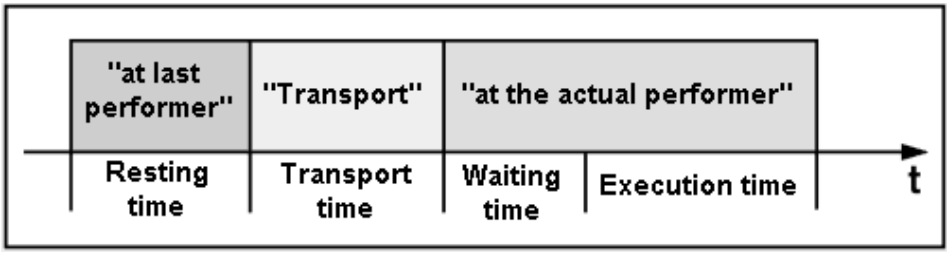


Figure 303: Standard definition of activity times

The following graphic (see fig. 304) represents the standard definition of the activity times in relation to an ADOxx Business Process Model in a laps of time. "Performer 2" is the current performer, "Performer 1" is the last performer and "performer 3" is the next performer.

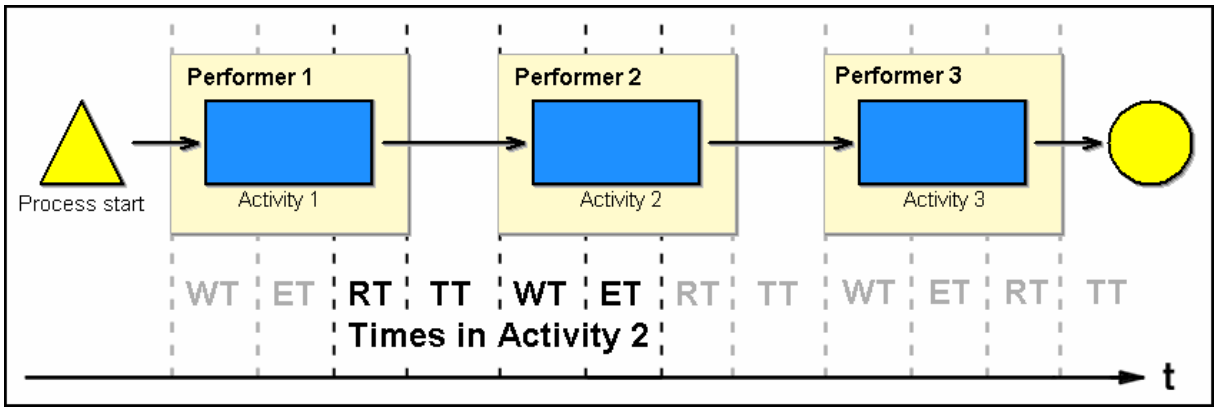


Figure 304: Standard definition of the activity times in a process model

ADOxx also gives the ability to alternatively define single activity times, especially resting and transport times. An alternative definition is presented in an excursus (see p. 337).

Input parameter

In the window for the start of simulation algorithms - for example in the "Simulation - Path analysis" window (see fig. 314) - you can select among the options (see fig. 305) in the "Input parameter" field, which activity attributes allow to present the times mentioned above or defined in your application library, as well as whether and if yes which program calls shall be executed.

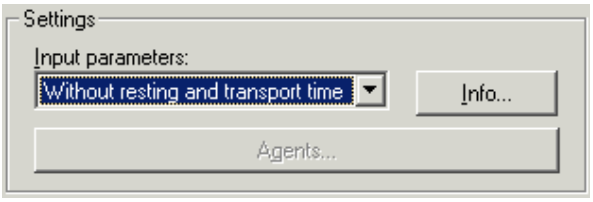


Figure 305: Part of the start window of the simulation

So, for instance, the option "Without resting and transport time" means that these two activity times will not be taken into account during the simulation. In other words, in this case, the resting and transport time will be considered as components of the waiting time. If necessary, - for example when supporting the appropriate business process through an "online processing EDP application - they can also equal zero.

ATTENTION: Since in the workload analysis the waiting time of the activities - in terms of standard definition (see p. 336) - arises as a simulation result, the resting and transport time are (if they do not steadily equal zero), to be placed in the single activities.

Clicking on the info button will give you an explanation of the selected input parameter combination.

Hint: The definition of the single input parameter combinations is carried out during the customising of ADOxx. If you want some further input parameter combinations, contact your ADOxx administrator.

Digression: alternative definition

In contast to the (recommended) standard definition concerning times of activities (see p. 336) (see fig. 306), you can also select the following (shown below) non-standard definition.

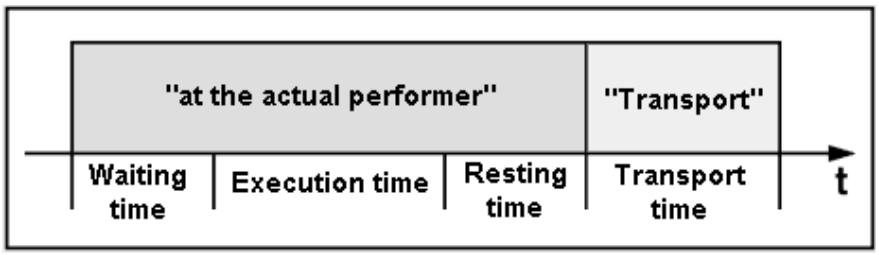


Figure 306: Alternative definition of the activity times

Resting Time: This is the time after an activity has been completed by one performer and before it is passed over to the next performer (for example it is lying in an out-tray waiting to be picked up by the internal post).

Transport Time: The time it takes for an activity to be transported from the previous to the next performer. Transport times especially occur - similarly to resting times - when documents must be passed on manually between the performers.

Hint: The definition of execution and waiting time corresponds to the standard definition of these time attributes (see p. 336).

The following graphic (see fig. 307) represents the alternative definition of the activity times in connection with an ADOxx Business Process Model in a laps of time whereas "performer 2" is the current performer, "performer 1" is the last and "performer 3" the next performer.

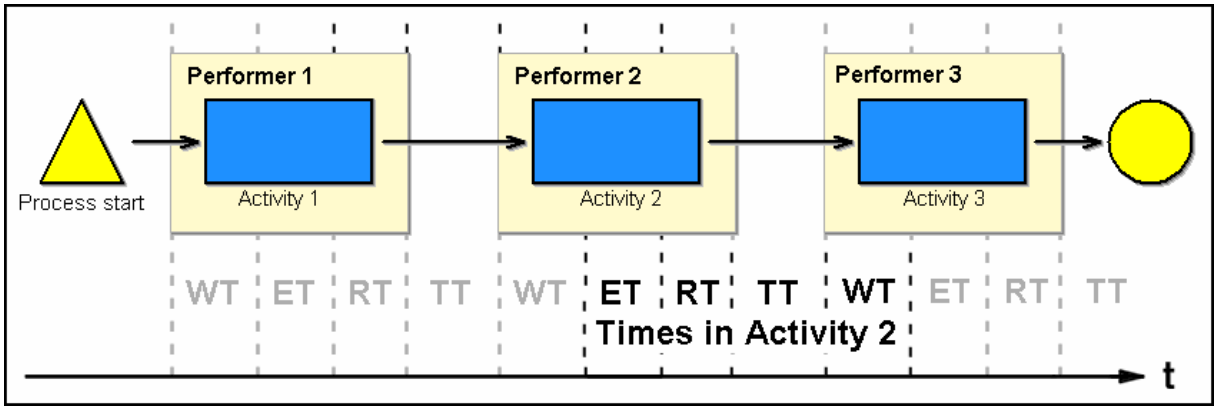


Figure 307: Alternative definition of the activty times in a process model

However this definition has a **disadvantage**, as resting and transport time equal all the activities which will follow. This has effect in the alternative paths (after decisions), since in this case, several activities can follow an activity.

Example:

In the process model shown below (see fig. 308) the disadvantage of the same resting and transport time is clear.

If the resting and transport times should differ between "activity 1" and "activity 2", "activity 1" and "activity 3" as well as "activity 1" and "activity 4", then they cannot be shown, since they are defined in only one attribute in the "activity 1".

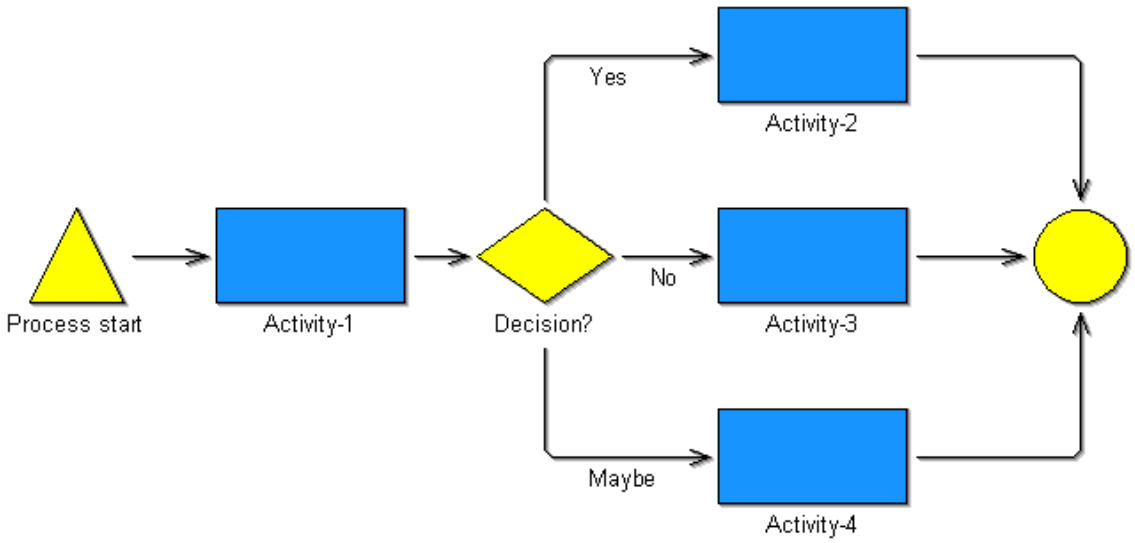


Figure 308: Example process model

The standard definition of the activity times (see p. 336) avoids this problem, since the resting and transport times are defined in the activities "activity 2", "activity 3" and "activity 4".

4.1.3.3 Process times as simulation results

One of the most important results which ADOxx produces as a result of ALL simulation algorithms are the average times (more exactly: "estimated expected values" of the business processes:

Cycle Time	The (average) cycle time of a business process specifies how much time is needed on average from the start of a business process to its end (end therefore determines the response time).
Execution Time	The (average) execution time of a Business Process Model specifies the average time necessary for executing the activities.
Waiting Time	The (average) waiting time of a Business Process Model specifies how much time is needed on average as waiting time prior to execution of an activity or while an activity is interrupted.
Resting Time	The (average) resting time of a Business Process Model specifies the average time in which the activities "rest" at the performers' desk after having been executed.
Transport Time	The (average) transport time of a Business Process Model specifies the average time necessary for transport between the execution of different activities.

Hint: The execution, waiting, resting and transport times are calculated both for activities and for business processes.

Effects of parallelisms on the cycle time

Where a Business Process Model contains/includes parallelisms, the cycle time will **not** be the sum of execution time, waiting time, resting time and transport time, since in this case several activities are executed **simultaneously**.

Example:

Take the business process as illustrated below (see fig. 309):

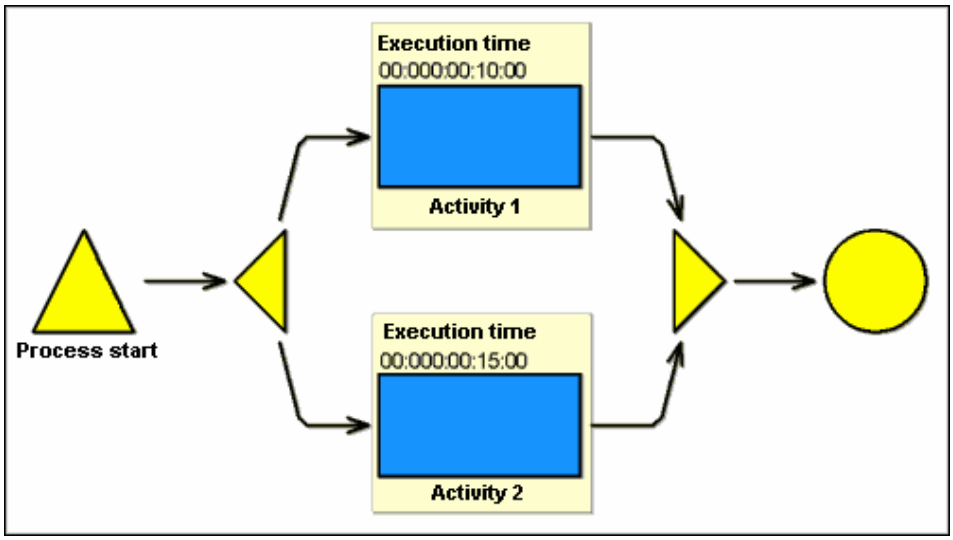


Figure 309: Effects of parallelism on the cycle time

"Activity-1" has an execution time of 10 minutes. "Activity-2" has an execution time of 15 minutes. All other activity times equal 0.

The execution time of the Business Process Model is thus 25 minutes (= 10 + 15 Minutes). The cycle time, however, is **15 minutes** (= maximum of 10 and 15 minutes), since both activities are executed in parallel and so both activities can start at the same time.

4.1.4 Cooperative vs. Non-cooperative Activities

By default, the simulation algorithms regard activities (objects of the class "Activity") as non-cooperative, i.e. they assume an activity is executed by exactly one performer.

You may, however, mark an activity as cooperative (attribute "Cooperative") in the ADOxx Notebook (see chap. 2.2.7.12, p. 146) of "Activity" objects. Cooperative activities (such as meetings) are always executed by several performers. To calculate the simulation results, the execution time (see chap. 4.1.3.2, p. 335) of a cooperative activity is multiplied by the number of participating performers.

The mechanisms by which cooperative activities are processed are explained in detail where the different simulation algorithms are described (see Path Analysis (see p. 354), Capacity Analysis (see p. 364) and Workload Analysis (see p. 382)).

> **Hint:** Should you work with a customer-specific application library, it is possible that the attribute "Cooperative" is not available. In such a case, you may not create cooperative activities as described above.

4.1.5 Global End

The attribute "Type" of an object of the class "End" can mark this end object as a "global end". When a global end is reached, the simulation - with one exception - **immediately finishes the main process** along with the subprocesses called, either directly or indirectly. The global end need not necessarily be defined in a main process model but may occur in any directly or indirectly called process model.

The one and only **exception** is, if the global end occurs within a parallel path. Then the main process and all the subprocesses are not finished immediately but only when the parallel paths are merged again. The following are regarded as parallel paths

- Paths which occur directly (in the same model) between a parallelity and a merging object (i.e. directly within a parallelity) as well as
- Paths within (possibly purely sequential) subprocesses, which are called directly or indirectly from within a parallelity.

If a global end is reached in a parallel path, processing continues until all parallel paths are merged and thus only one active path exists.

That means the highest level parallelity within the call hierarchy must be closed by an appropriate merging object. All activities succeeding this merge (both in the same process model and in any following subprocesses) are then omitted and the main process as well as all the subprocesses active at this time are finished.

Example:

See the business process illustrated below (see fig. 310) with the subprocess (see fig. 311) also illustrated. The End object within the subprocess is a "Global end".

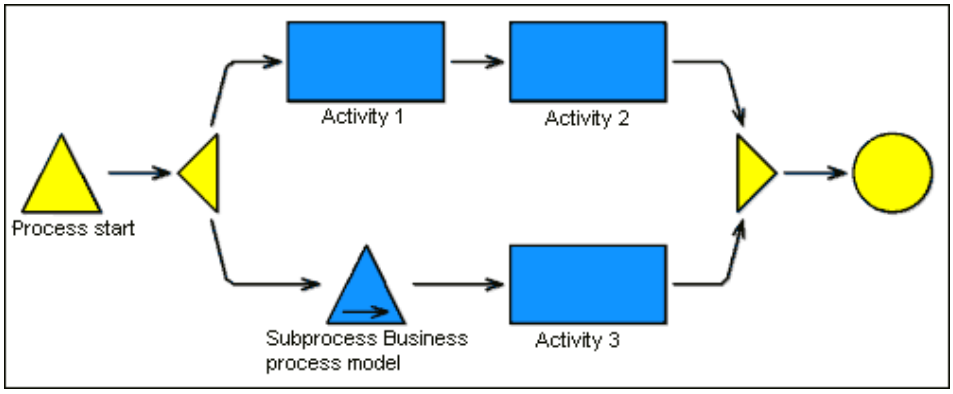


Figure 310: Global end - main model (example)

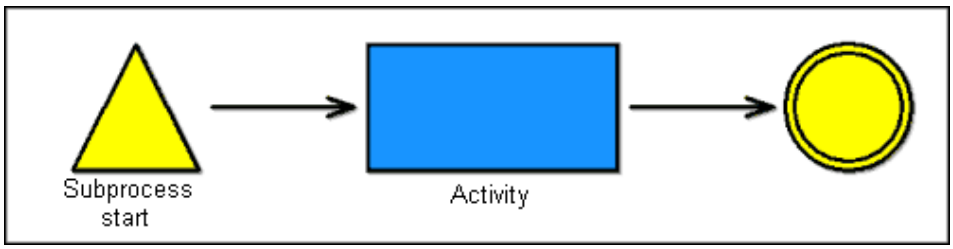


Figure 311: Global end - submodel (example)

Da der Subprozeß innerhalb einer Parallelität aufgerufen wird, wird die Prozeßausführung solange fortgesetzt, bis die Parallelität abgeschlossen ist. D.h. es werden "Aktivität-1", "Aktivität-2" sowie parallel "Aktivität" (im Subprozeß) **und "Aktivität-3" (!)** ausgeführt.

Since the subprocess is called from within a parallelity, this means that process execution continues after the global end is reached until the merging object closes the parallelity. This means that "Activity-1", "Activity-2" and **"Activity-3" (!)** are executed. However should an extra activity exist in the Business Process Model between the merging object and the end, this activity would NOT be executed due to the global end in the subprocess.

Hint: Should you work with a customer-specific application library, it may be possible that you cannot create a "Global end". If necessary, please contact your ADOxx consultant.

4.1.6 Manual random generator

Hint: The function for the manual random generator is defined in the application library for the random generator objects.

A manual random generator means, that during the simulation the actual value of the variable can be set by the user according to the available values defined in the random generator and used in the transition conditions of the subsequent relations. This way the simulation process can be steered by the user.

The function "manual random generator" will be set during the modelling (see chap. 2., p. 126) in the process model by activating the attribute "manual random generator" in the corresponding object of the class "random generator".

During the simulation, you must select in each simulation run for each random generator, which is set on "manual", one of the values previously defined in the window "<*Variable name*> - Value entry" (see fig. 312).

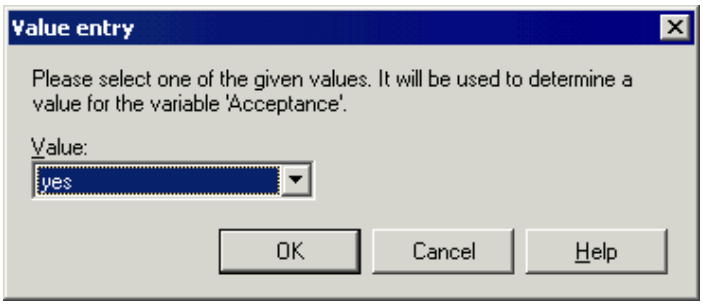


Figure 312: Manual random generator (value entry during the simulation)

The values authorised for the given variable are listed in the **"value" field**. Select any value and then click on the OK button to continue the simulation.

Hint: Clicking on the cancel button will abort the simulation.

4.1.7 Period-related versus Process-related View

Period-related and process-related results are distinguished in the simulation algorithms "Capacity Analysis" (see chap. 4.3, p. 354) and "Workload Analysis" (see chap. 4.4, p. 367).

In the case of process-related results the average values for the Business Process Model (such as "Average cycle time", "Average costs" etc) are calculated. In the case of period-related results these values are scaled with regard to a certain period of time. This scaling is based on the frequency of occurrence of the process.

The following period-related results are available in the simulation algorithms "Capacity Analysis" (see chap. 4.3, p. 354) and "Workload Analysis (steady state)" (see chap. 4.4, p. 367):

- Results **per year** and
- Results **per month**.

The simulation algorithm "Workload Analysis (fixed time period)" (see chap. 4.4, p. 367) offers period-related results **for the evaluation period selected when the simulation starts** (example: results for the period of time from the 21st of February to the 17th of July).

Period-related results inform you about the capacity and the workload of performers and groupings (such as roles or organisational units) within the evaluation period.

ATTENTION: The period-related and process-related results cannot always be directly converted into each other in the case of the "Workload Analysis" (see chap. 4.4, p. 367) simulation algorithm. All the activities executed during the evaluation period are taken into account for the period-related results. This will also include activities from processes which started or ended before or after the evaluation period.

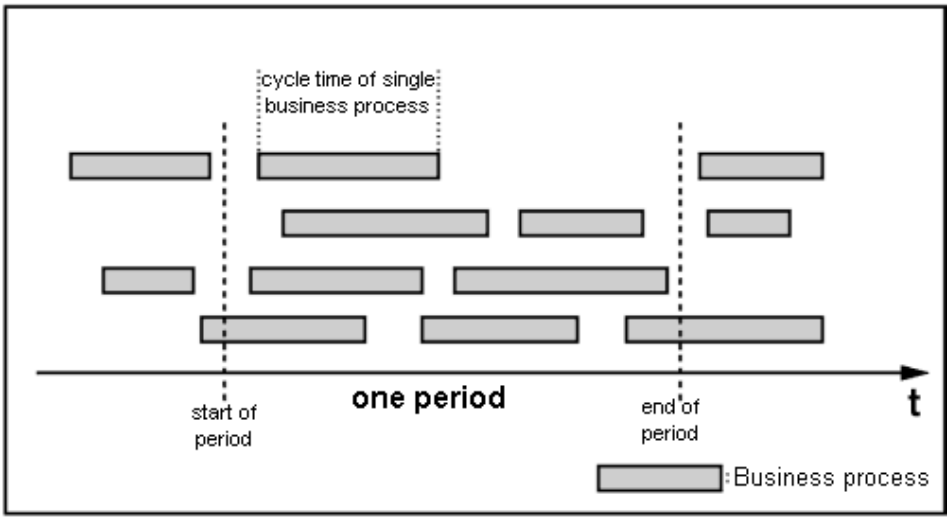


Figure 313: Period-related/process-related View

4.2 Path Analysis

The path analysis algorithm enables you to evaluate your Business Process Models (see chap. 4.2.6, p. 354) without regard to the Working Environment (organisational structure). You can obtain process and path related results for a Business Process Model (including all subprocesses which have been called). The process-related results describe the average times and costs of the Business Process Model. You can assess the paths run through depending on their frequency of occurrence and the time and cost criteria you defined. In addition, the time and cost criteria of each path can be displayed both graphically and in text format.

The path analysis (see chap. 4.2.5.1, p. 353) is based on a process model including all subprocesses.

When you execute a path analysis (see chap. 4.2.1, p. 343), the paths of the Business Process Model are run through and the times and costs (see chap. 4.2.3, p. 347) occurring in the course of this are calculated.

4.2.1 Executing a Path Analysis

To start the path analysis carry out the following steps:

1. Select the option "Path Analysis" from the "Algorithm" menu. Alternatively, click on the smart-icon in the quick-access panel.

   The window "Simulation - path analysis" (see fig. 314) will appear.

2. Select the process model which you wish to simulate from the "Model Selection" list. The model selection list can be refreshed by clicking on the icon .

3. Check the following settings:
   - Numbers (see chap. 4.2.1.1, p. 345)
   - Options (see chap. 4.2.1.2, p. 346)
   - Passive Components (see chap. 4.2.1.3, p. 346)
4. Confirm your entries by clicking on the OK button.

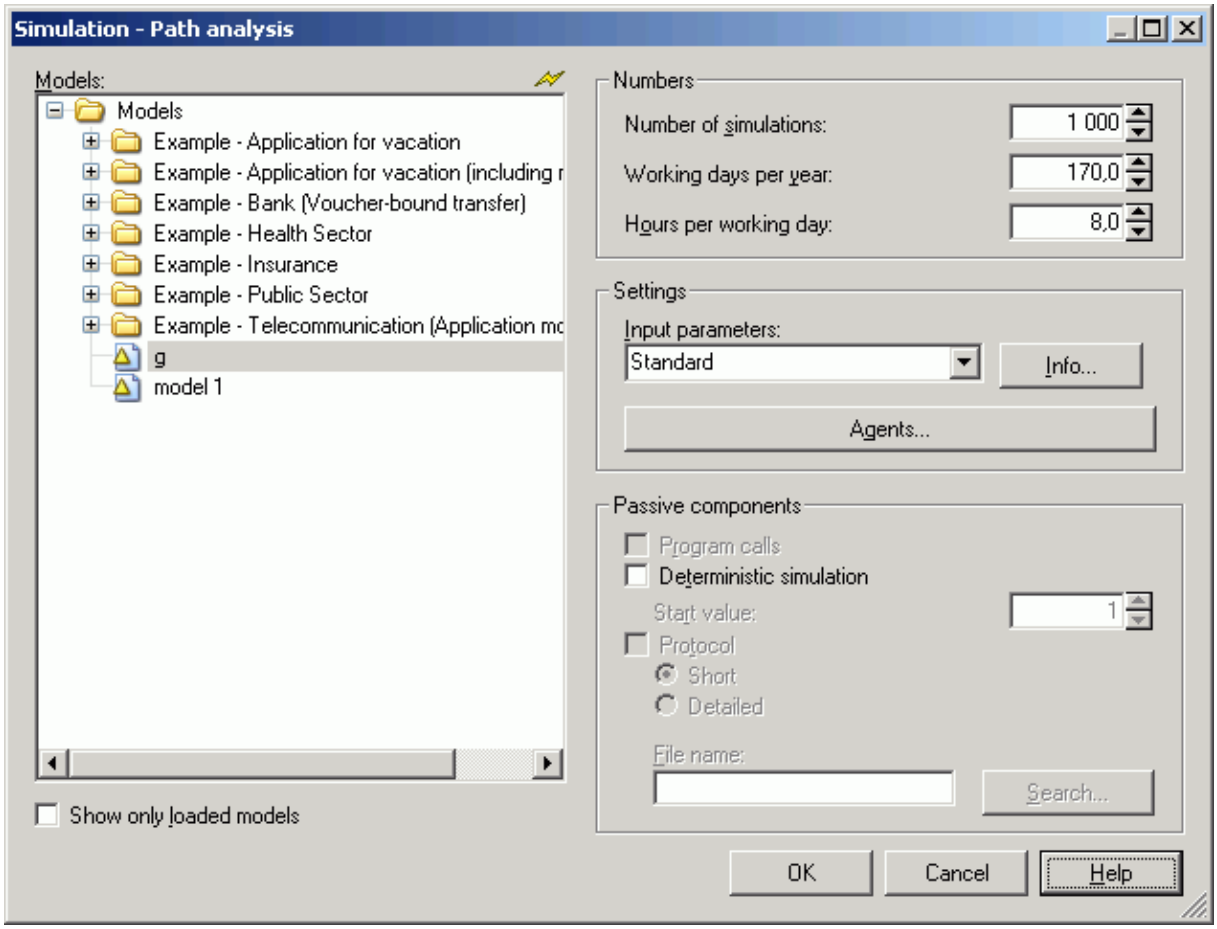


Figure 314: Start window for the path analysis

Activate the **option "Show only loaded models"** to limit the display of models to be simulated to the models currently opened (see fig. 315).

Hint: Write-protected models bear a lock symbol (🔒) next to the model type symbol.

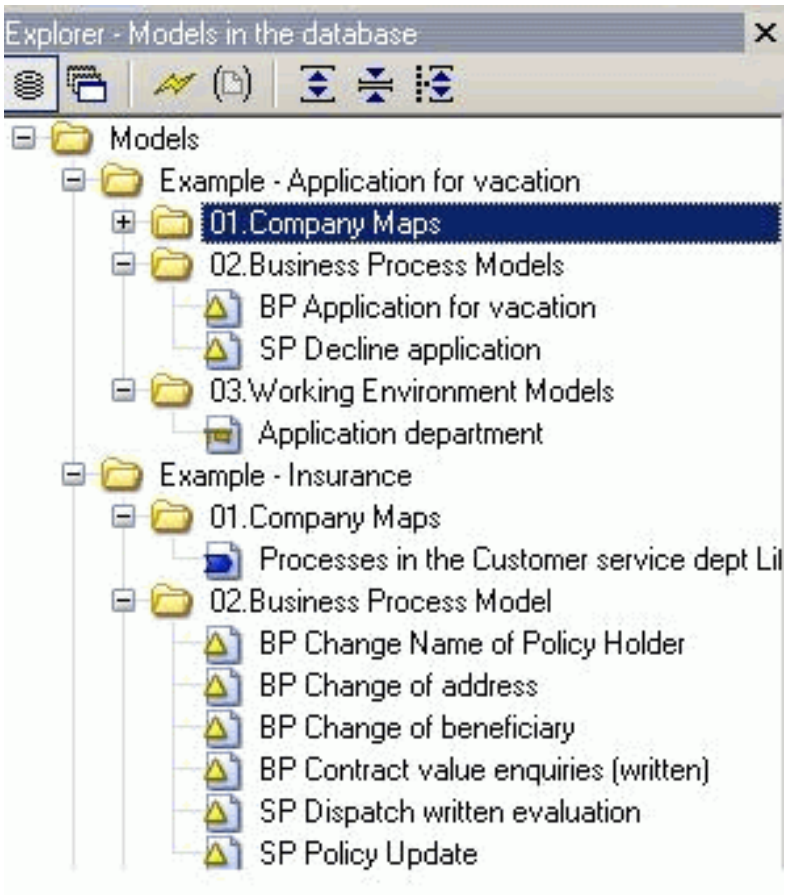


Figure 315: Display of loaded models

This way you can identify the models to which you have a write access and to which it is possible to take over the simulation results (see chap. 4.2.4.3, p. 350) after the simulation.

Hint: By double-clicking on a process model in the model selection list, a path analysis based on this model and the current settings is started.

Hint: The "Passive components" are not available for path analysis.

4.2.1.1 Numbers

"Number of simulations" indicates how many processes are to be "run through". The number of simulations selected affects the accuracy - the higher the number, the more exact the simulation results will be. Obviously the simulation will take longer as the number of simulations to be run is increased.

"Working days per year" together with the value in the field **"Hours per working day"** this field defines company time (see chap. 4.1.3.1, p. 335).

"Hours per Working Day" (together with the value in the field **"Working days per year"**) is used to define company time (see chap. 4.1.3.1, p. 335).

Hint: When ADOxx is customised, default values may be set for the entries in the fields **"Working days per year"** and **"Hours per working day"**. Should you wish to change the defaults, please contact your ADOxx administrator.

4.2.1.2 Options

The **"Input Parameters"** field allows you to select which attributes should serve as input for the path analysis (see chap. 4.1.3.2, p. 335) and thus which results are calculated. When you click on the "Info" button, the current input parameter combination is explained. New input parameters can be configured during customising if required. If necessary please contact your ADOxx administrator.

Click on the button **"Agents"** to open the Agent Overview (see chap. 4.8.2, p. 392) screen which shows any agents assigned to the model currently selected (and if necessary allows you to add existing or create new agents).

4.2.1.3 Passive components

"Program calls" Default setting: deactivated.

If this option is activated, program calls specific to the activity will be carried out during the simulation of each activity. The selected input parameter combination (see p. 337) will determine which program calls will be concerned by this .

"Deterministic simulation" Default setting: disabled.

When enabling this option the simulation is initialised with the same start value. This ensures that with the same start value independent simulation runs will determine the same simulation results. To obtain identical results with the deterministic simulation the content of the models to be simulated must not be changed.

4.2.2 Simulation Run

Loading any necessary models

If not all the Business Process Modells necessary for the path analysis are open, the additional models needed will be loaded **write-protected**.

Once the path analysis is finished, i.e. when the result window has been closed, the system will ask you, whether the models loaded especially for the simulation should also be closed.

> **Hint:** If any necessary models for the path analysis have been loaded write-protected, you may **not** transfer the simulation results (see chap. 4.2.4.3, p. 350) into the models after the simulation.

Should an error occur while the Business Process Models are being loaded, refresh the model lists by clicking on the button **"Refresh"** in the window "Simulation - path analysis" (see fig. 314) and try to load the models again.

Transferring models

Once all necessary models have been loaded, all simulation models are transferred into an internal representation. This is shown by the status window's title.

> **Hint:** Should the models still exist in the simulation cache (see chap. 4.6, p. 388), (maybe they have been loaded previously for a simulation), it will not be necessary to transfer them a second time.

Errors in the course of the simulation

If an error is detected during the simulation, you will be informed about it with a special information window. In this case, you can close the information window by clicking on the "OK" button, or by pressing the "Enter" key. The model in which the error was detected is brought to the foreground where the object is marked graphically. Correct the error and then restart the path analysis.

> **Hint:** Error statements (see chap. 8., p. 520) which may occur during the simulation, their causes and recommended actions are described in another chapter.

Status window

The simulation may take some time, depending on the number of simulation runs and the size of the Business Process Models to be simulated. A status window informs you about the current state of the simulation.

You may cancel the path analysis by clicking on the "Cancel" button and confirming the following query. In this case, **no** results will be shown.

4.2.3 Results

The results of the path analysis are divided into path results (see chap. 4.2.3.1, p. 347) and business process results (see chap. 4.2.3.2, p. 348).

The path results (see chap. 4.2.4.2, p. 350) are displayed as a **textual history (or protocol)**.

Additionally, a **graphical** representation of the path which you are currently viewing is displayed in the business process model itself - the path you are viewing is highlighted in red.

The business process results (see chap. 4.2.4.1, p. 349) are displayed in **tables** (ADOxx Browser (see chap. 8., p. 66)).

4.2.3.1 Path Results

The following results can be seen for each path:

- **Average Execution Time of the Path**
- **Average Waiting Time of the Path**
- **Average Resting Time of the Path**
- **Average Transport Time of the Path**
- **Average Cycle Time of the Path**
- **Average Costs of the Path** (if these are defined in the current input parameter combination)
- **Average Probability of the Path**

These totals result from the sum of the attribute values of the activities included in the path. (i.e. Average execution time for a path is derived from the sum of all "Execution time" attributes in all activities within that path). Cooperative activities (see chap. 4.1.4, p. 340), which the path may include, are excepted.

> **Hint:** By defining input parameter combinations (see chap. 4.1.3.2, p. 335) ("Simmapping") in the ADOxx Administration Toolkit it is possible to calculate additional results. Contact your ADOxx administrator, if necessary.

4.2.3.2 Business Process Results

The path analysis calculates the following totals as business process results:

- **(Average) Execution Time of the Business Process**
- **(Average) Waiting Time of the Business Process**
- **(Average) Resting Time of the Business Process**
- **(Average) Transport Time of the Business Process**
- **(Average) Cycle Time of the Business Process**
- **(Average) Costs of the Business Process** (if these are defined in the current input parameter combination)

These totals specify the average values for the business process as determined by the simulation. The values are the sum of the results for the single paths (see chap. 4.2.3.1, p. 347), each weighted with its path's probability.

> **Hint:** While ADOxx is customised, additional "input parameter combinations" (see chap. 4.1.3.2, p. 335) can be defined. In this way you can obtain additional business process results. If you would like to set up different "input parameter combinations" or additional results, please contact your ADOxx administrator.

4.2.4 Displaying results

After the simulation, the result window of the path analysis will be shown ("Path Analysis - Business Process Model: <*Model Name*>") (see fig. 316). In this window you may select the manner in which the results should be displayed.

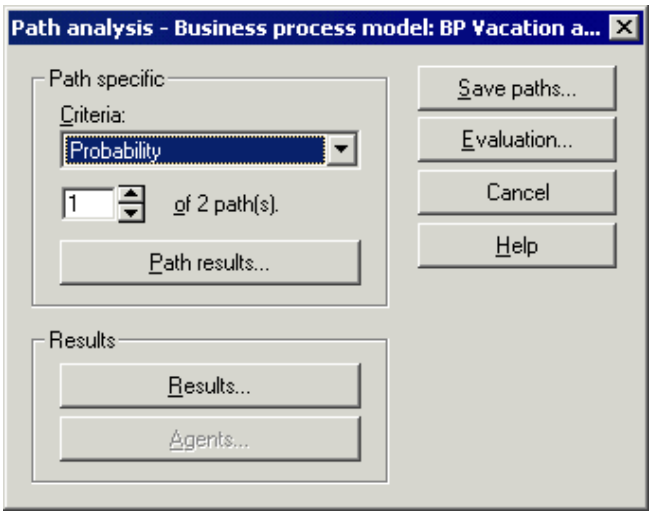


Figure 316: Path analysis - result selection

The result of a path analysis is a set of paths based on the structure of the Business Process Model (including any subprocesses). The path results (see chap. 4.2.3.1, p. 347) are calculated for each path. The list of paths may be ordered according to different criteria - i.e. by probability, by highest execution time etc.

By clicking on the button:

"Path Results" you can display the selected path graphically in the model window and as text (in the window "Path Analysis - Result Paths <*No*>" (see fig. 318) together with the corresponding path results (see chap. 4.2.4.2, p. 350).

From the **Criteria list** you may select a sorting criterion for the path results. The paths are sorted in descending order according to the criterion selected, i.e. the path with the highest value (such as longest cycle time, highest costs and so on) will be placed in the first position - Path 1.

In addition, you can select which path should be displayed from the sorted path list.

Hint: Different sorting criteria may result in a different order of paths. The path with the longest cycle time, for example, is unlikely to also be the path with the highest probability.

"Results" you can display the "business process results" (see chap. 4.2.4.1, p. 349) of the process.

"Agents" you can list the results calculated by agents (see chap. 4.8, p. 391).

By clicking on the button **"Save"** you can save all the path results of the path analysis into a file.

If you wish to transfer the results back into the models (see chap. 4.2.4.3, p. 350) click on the button **"Evaluation"**.

Exit the simulation results by clicking on the "Cancel" button. A window will then inform you that the simulation results will be deleted, if you close the window.

After finishing the path analysis (that is, after closing the result window), you will probably be asked, whether the Business Process Models opened especially for the path analysis should be closed again. Please note that any Business Process Models loaded in addition were loaded **write-protected**.

4.2.4.1 Display business process results

The business process results (see chap. 4.2.3.2, p. 348) are displayed in the window "Path Analysis - Business Process Model: <*Model Name*>" (see fig. 317).

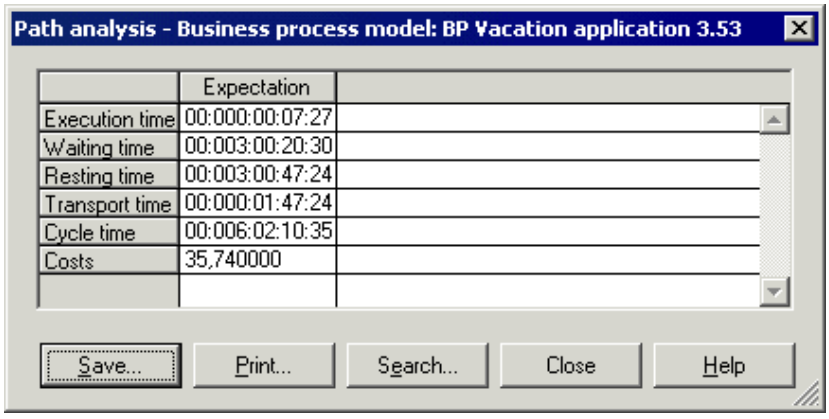


Figure 317: Path analysis - total results

You can save and print out the results displayed in the ADOxx browser (see chap. 8., p. 66).

4.2.4.2 Path results

The path selected is highlighted in the model windows (see the red border around the objects and connectors in the path). Text describing the path results (see chap. 4.2.3.1, p. 347)(protocol) can be viewed in the window "path analysis - result path <*No*>" (see fig. 318). Here information on which activities were processed in this path is shown as well as the overall totals of execution time, cycle time etc. for the path

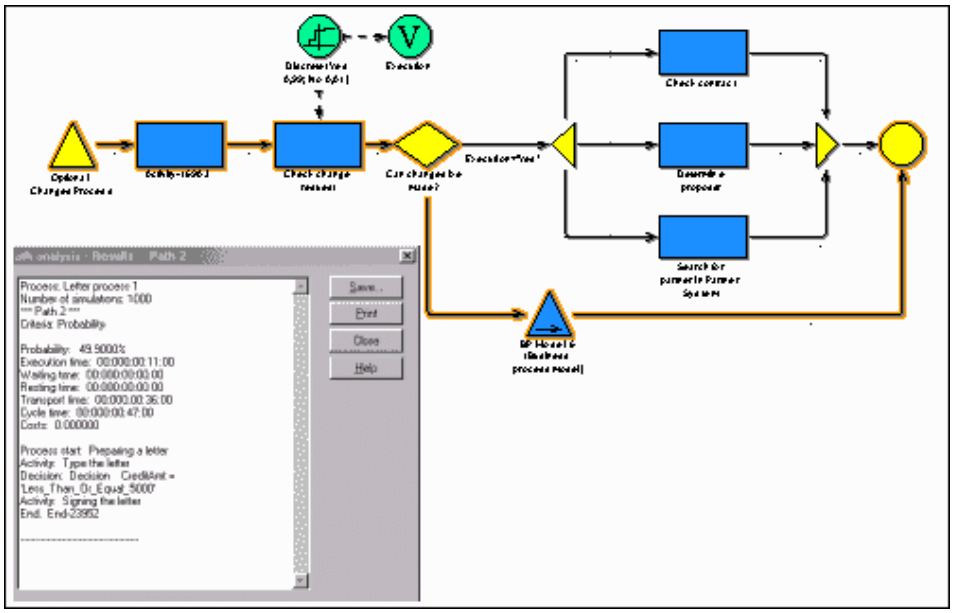


Figure 318: Path analysis - display path results

Furthermore, this protocol contains a listing of all objects run through.

You can save the path analysis protocol of a path to a file by clicking **"Save"** or print it by clicking **"Print"**.

The graphical representation of the selected path can be printed using the "Print model" menu item within the "Algorithm" menu or, by clicking on the smart-icon in the quick-access panel.

4.2.4.3 Transferring results

The simulation results of the path analysis (see chap. 4.2.3, p. 347) can be transferred into the business process models simulated. This means that the values calculated in the simulation can be saved back to a chapter in the Notebook of the "Process start" symbol and/or the "Activity" object. By doing this, you can then apply the available evaluation mechanisms (see p. 352) after the simulation.

Transfer the results of the path analysis by clicking on the button **"Evaluation"** in the window "Path Analysis - Business process model: <*Model Name*>" (see fig. 316).

You will be asked, whether you really want to transfer the simulation results into the models. The **"Info"** button will give you some information about the results currently stored in the models (see p. 351).You may then decide, whether or not to overwrite them.

If you decide to store the results, the window "Evaluation - result adoption" (see fig. 319) appears, in which you can specify the type of results to be saved.

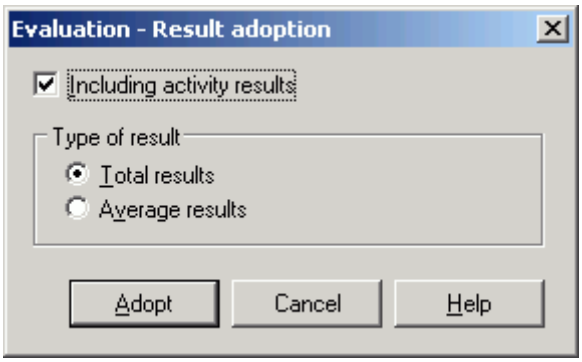


Figure 319: Path analysis - transfer results

Activate the option **"Including activity results"**, to transfer results for each activity into the model.

If you select the option **"Total results"**, the sum of the results of all runs will be transferred. If you select the option **"Average results"**, the average results (in relation to one process run) will be transferred.

The business process results (see chap. 4.2.3.2, p. 348) are transferred into the process start object of the "main process".

If you select the "Including activity results" option the activity results for the processes run through will be transferred into the objects of class "Activity". (See chapter "Aggregated results" in an Activity Notebook).

Transfer the results by clicking on the button "Adopt". Immediately after the transfer you may use the evaluation mechanisms (see p. 352) "Queries" and "Pre-defined evaluation queries". Note, these evaluation queries are also available within the Evaluation component of ADOxx.

Hint: The system will inform you if you attempt to store results from a simulation for the second time - i.e. identical results

Hint: Transferring simulation results changes the contents of the models simulated. These are then marked as "modified".

You can delete the simulation results transferred using the option "Reset simulation results" (see chap. 4.7, p. 390) in the "Edit" menu within the Simulation Component.

Result information for simulated models

Before you transfer the simulation results into the models (see chap. 4.2.4.3, p. 350), you should ensure that you are not over-writing results you may want to keep.

You can see which results are currently stored in the models by clicking on the button **"Info"**. The window "Result information for simulated models" (see fig. 320) will show you the results.

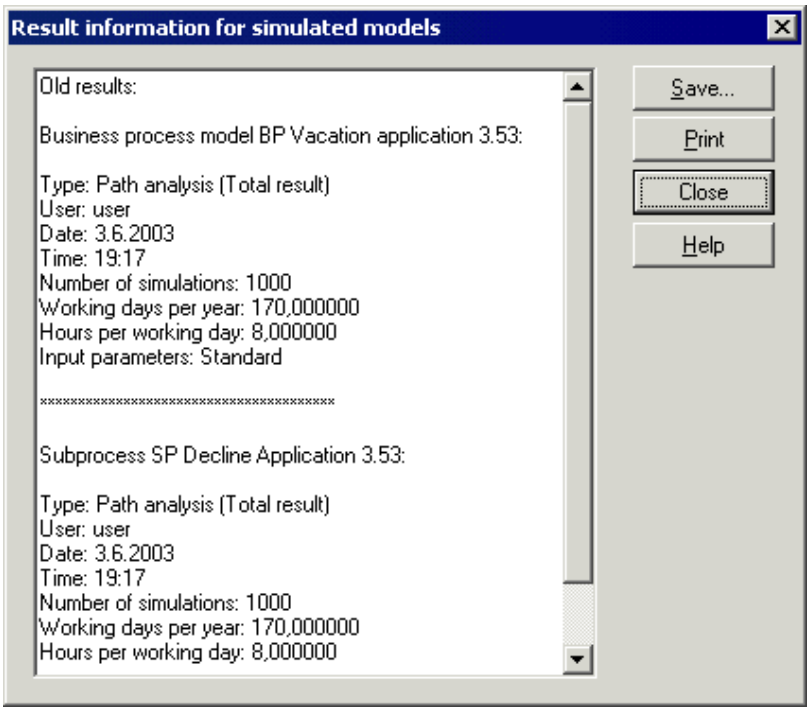


Figure 320: Result information for simulated models

The result information is structured in the following manner:

- **Result Type**
- **User who transferred these results**
- **Transfer Date**
- **Transfer Time**
- **Simulation Settings** (number of simulations etc)

The result information is shown for all models concerned. You have an option to save this information or to print it out.

Evaluation mechanisms

The evaluation mechanisms enable you to analyze simulation results which were transferred into the models.

You can choose between **"Queries"** and **"Pre-defined evaluation queries"** in the window "Evaluation mechanisms" (see fig. 321).

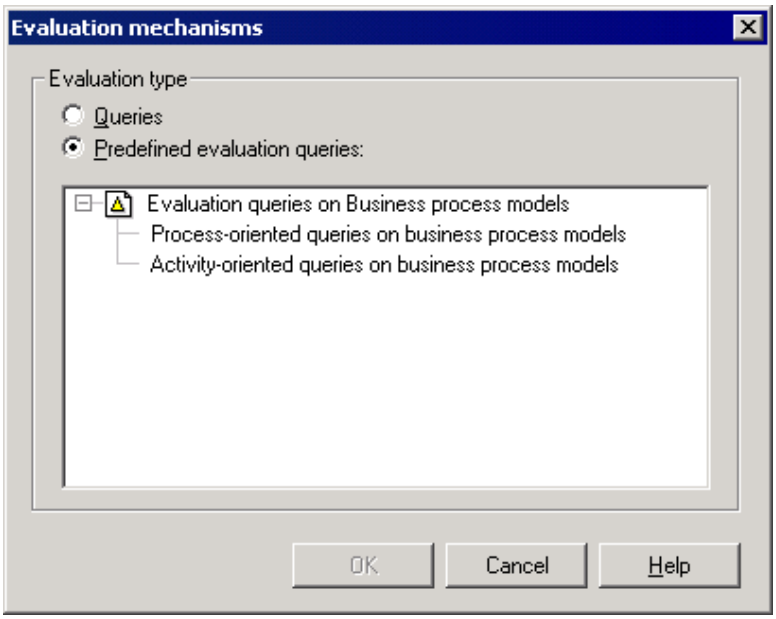


Figure 321: Evaluation mechanisms

Hint: The simulation results must first be transferred into the models (see chap. 4.2.4.3, p. 350) before you can use the evaluation mechanisms.

By selecting the option **"Queries"** you can run analysis queries (see chap. 3.4, p. 325) on the Business Process Models simulated.

By choosing the option **"Pre-defined evaluation queries"** you can run certain predefined evaluation queries (see chap. 5.2, p. 438) - i.e. queries defined while ADOxx was customised - on the simulation results which were transferred into the Business Process Models.

Hint: If there are no predefined evaluation queries, the option "Pre-defined Evaluation Queries" will be deactivated and cannot be chosen. If necessary, contact your ADOxx administrator.

4.2.5 Background

4.2.5.1 Input

During path analysis exactly one Business Process Model (as well as any subprocesses called) is simulated.

This business process must be correctly modelled. The same applies to any subprocesses called. It is especially important that objects of class "Variable" and "Random generator" are used correctly and that the transition conditions in the "Subsequent" connector are syntactically correct.

4.2.5.2 Processing

During path analysis the process (as well as any subprocesses called) is "played through".

The activities are **not** assigned to performers and resources.

How to treat cooperative activities

A cooperative activity is executed by several performers simultaneously. The number of performers participating in a cooperative activity, must be specified in the "Average number of participants" attribute.

When the path and process results are calculated, the times and costs of cooperative activities are multiplied by the value of the attribute "Average number of participants". The execution time of cooperative activities corresponds to that of parallel activities and does not therefore affect the cycle time.

Example:

An activity with an execution time of 3 minutes is marked as cooperative (in the ADOxx Notebook via the attribute "Cooperative") and has the value "4" in the attribute "Average number of participants". The execution time for this activity is twelve minutes (= 4 × 3 minutes). The cycle time for this cooperative activity is of course only 3 minutes as the activity is executed **simultaneously** by all 4 performers.

4.2.6 Application Scenario (Example)

Starting with the business process results (see chap. 4.2.3.2, p. 348) of the path analysis (see chap. 4.2, p. 343) you can calculate the personnel capacity demand for business processes.

Example:

The business process "X" occurs 2000 times per year. A year corresponds to 170 working days in this company. A working day has 8 hours.

The expected execution time calculated by the path analysis for business process "X" is 2 days. (i.e. 00:002:00:00:00).

The personnel capacity demand, then, is calculated as follows:**(Working hours needed for the Process per Year) ÷ ÷ (Working hours of a Performer per Year)**

That is:**{ [(Frequency of Occurrence of the Process per Year) × × (Expected Execution Time)] ÷ ÷ [(Working Days per Year) × (Hours per Working Day)] }**

From these results the personnel capacity demand of the business process "X":`(2000 processes/year × 2 days × 8 hours) ÷ (170 days × 8 hours) = = 23,5 performers`

24 performers are thus necessary for executing business process "X".

4.3 Capacity analysis

The capacity analysis simulates business processes while taking into account the corresponding working environment. If you execute a capacity analysis (see chap. 4.3.1, p. 355), the activities of the business processes simulated are executed by the performers assigned to them. If necessary, resources are also assigned.

Through this, statements related to both the process and the period (see chap. 4.1.7, p. 342) can be made which concern process or job related capacities. Based on the results of the capacity analysis you can plan your personnel capacity demand (see chap. 4.3.4.2, p. 361).

The capacity analysis is based on (see chap. 4.3.5.1, p. 362) an application model (see chap. 2.4.16, p. 284). This consists of exactly one working environment model and at least one Business Process Model.

4.3.1 Carrying out a capacity analysis

Follow the steps below to start the capacity analysis:

1. Select the option "Capacity analysis" in the menu "Algorithm". Alternatively, you can click on the smart-icon in the quick-access panel.

   The window "Simulation - Capacity analysis" (see fig. 322) will be shown.

2. Select the application model (Business Process Model and Working Environment model) on which you want to run the capacity analysis from the list under "Application model selection". If you click on the button **"Add"**, you can define new application models. By clicking on the button **"Refresh"** you can refresh the model selection list.

3. Make the following entries:

   - Numbers (Simulations, Working days per year, Hours per working day) (see chap. 4.3.1.1, p. 356)
   - Options (see chap. 4.3.1.2, p. 357)
   - Passive Components (see chap. 4.3.1.3, p. 357)

4. Confirm your entries by clicking on the OK button.

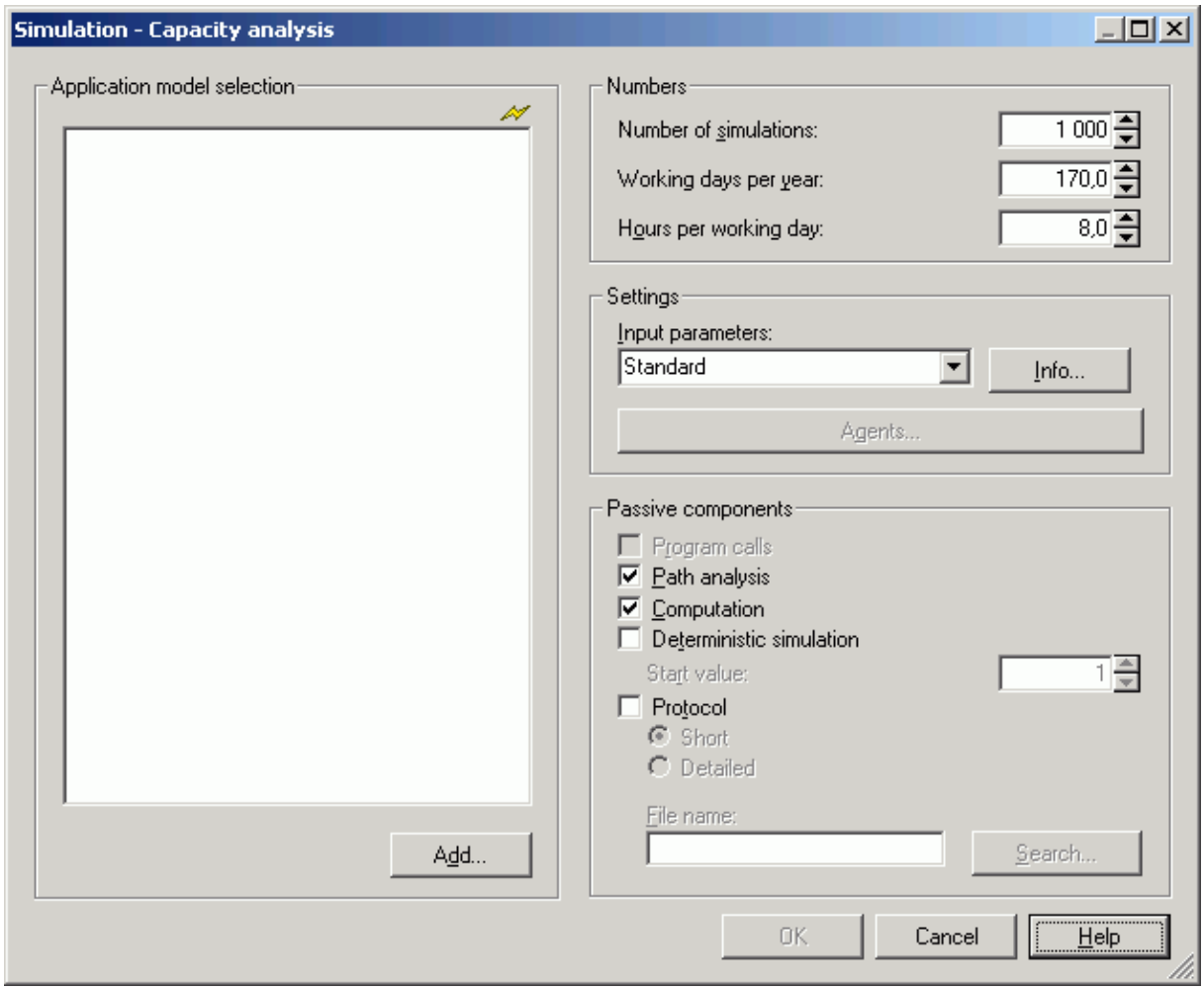


Figure 322: Carry out capacity analysis

Hint: By double-clicking on an application model in the selection list you can start the capacity analysis based on that model and the current numbers, settings and passive components.

4.3.1.1 Numbers

"Number of simulations" this indicates how often each process contained in the application model is to be simulated. The number of simulations is a measure of accuracy, i.e. the higher the number, the more exact the simulation results will be. Note - the higher the number of simulations, the longer the simulation will last.

"Working days per year" serves (in combination with the value in the field "Hours per working day") to convert time into company time (see chap. 4.1.3.1, p. 335).

"Hours per working day" serves (in combination with the value in the field "Working days per year") to convert time into company time (see chap. 4.1.3.1, p. 335).

4.3.1.2 Options

The results to be calculated in the capacity analysis (see chap. 4.1.3.2, p. 335) are determined by the value in the field **"Input Parameters"**. When you click on the Info button, the current input parameter combination is explained.

Click on the button **"Agents"** to open the Agents overview (see chap. 4.8.2, p. 392) of the models belonging to the currently selected application model (in the field "Application model selection").

4.3.1.3 Passive components

"Program calls" Default setting: deactivated.

If this option is activated during the simulation of each activity, program calls specific to the activity will be carried out . The selected input parameter combination (see p. 337) will determine which program calls will be concerned by this.

"Path analysis" Default setting: activated.

If this option is activated, the results of the capacity analysis are calculated. If you deactivate the option **"Path Analysis"**, you will **not** obtain cycle times for the business processes (see chap. 4.1.3.3, p. 339).

"Computation" Default setting: activated.

If this option is activated, the results of the capacity analysis are calculated. If you deactivate the option **"Computation)"**, you will obtain **no** results at all.

Hint: It can be sensible to deactivate the options "Path analysis" and "Computation" in cases in which you need no results but just want to generate a simulation protocol.

"Deterministic simulation" Default setting: disabled.

When enabling this option the simulation is initialised with the same start value. This ensures that with the same start value independent simulation runs will determine the same simulation results. To obtain identical results with the deterministic simulation the content of the models to be simulated must not be changed.

"Protocol" Default setting: deactivated.

If the option is activated, a simulation protocol is generated in the file you specify.

Select the **short version** to use the protocol as basis for the offline animation (see chap. 4.5, p. 386). The short version of a protocol is stored in the compressed APF format (APF = **A**DONIS **P**rotocol **F**ormat).

Select the **detailed version** to generate a detailed protocol. After the simulation you can open the protocol file with a text editor and trace it step by step.

4.3.2 Simulation run

Information regarding the loading of additional models, the transfer of models and the treatment of errors during the simulation is as described in path analysis (see chap. 4.2.2, p. 346).

4.3.3 Results

The results of the capacity analysis are divided into period-related and process-related results (see chap. 4.1.7, p. 342).

The period related results are calculated (see p. 366) based on the business process related results.

The results are further divided into capacity planning and resource results (see chap. 4.3.4.1, p. 359).

The results can be displayed in the form of tables and diagrams, they can be printed or saved to a file in order to edit them further, for example in a spreadsheet calculation program.

4.3.4 Displaying Results

After the simulation run, the result window "Capacity analysis - application model: <*Model Name*>" (see fig. 323) appears.

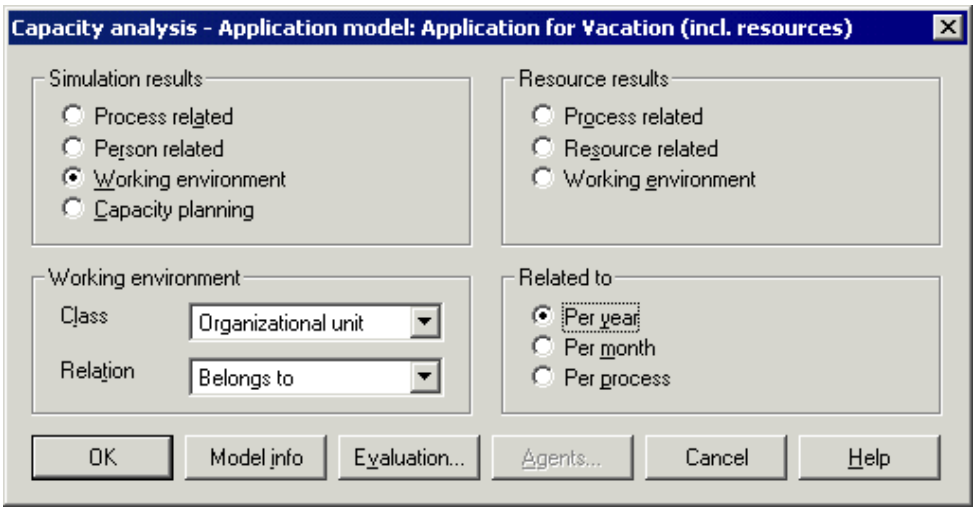


Figure 323

In this window you can define the reference type (per year, per month or per process), the result type (working environment related or resource results (see chap. 4.3.4.1, p. 359)) and the structure (process-related, person-related or working environment-related).

Hint: If no resources have been assigned to the activities of the Business Process Models, there will be **no** resource results and the result window will be displayed without the options for the resource results.

Hint: You can obtain the capacities for the different performers by selecting the settings "Person related" and "per year" or "per month", respectively.

When the option **"Working environment"** (for both the simulation and the resource results) or the option **"Capacity planning"** (see chap. 4.3.4.2, p. 361) is selected, the fields "Class" and "Relation" are activated. Here you can specify the grouping class (such as "Organizational unit" or "Role") and the relation which exists between the performers of the Working Environment model and the grouping object specified (such as "Belongs to"). Results will then be displayed in a table.

If one or more performers do not belong to exactly one grouping, an appropriate statement will be displayed, since this means that the results cannot be unambiguously assigned to one grouping.

Clicking on the button **"Model info"** will show you which Business Process Models and Working Environment model are part of the application model simulated.

Once the settings have been selected, the result table can be seen in the ADOxx browser (see chap. 8., p. 66) by clicking the OK button.

The results can be seen in different hierarchical levels, in graphical format (pie chart or bar chart) and can be saved in different formats to a file.

The simulation results can be transferred (see chap. 4.3.4.4, p. 362) back into your models by clicking on the **"Evaluation"** button.

If you have used agents (see chap. 4.8, p. 391) to calculate additional user-defined results, then the agents' results (see chap. 4.8.6.2, p. 424) can be obtained by clicking on the button **"Agents"**.

By clicking on the **"Cancel"** button you may exit the result representation. A window will inform you then that the simulation results will be deleted, if you close the window.

When you have finished the capacity analysis, that is, after you have closed the result window, you will - if necessary - be asked, whether the Business Process Models opened especially for the capacity analysis are to be closed again. Please note that any Business Process Models which were loaded for this reason were opened **write-protected**.

4.3.4.1 Displaying Results in Tabular Form

The results are divided into **simulation-** and **resource results**.

When examining simulation results, performers are assigned to single activities and processes. When viewing resource results, resources (PC, type-writer and so on) are assigned to the single activities and processes.

The results can be displayed either per process, per year or per month (see fig. 324). The results are shown in the ADOxx browser (see chap. 8., p. 66). Within the ADOxx browser, you may expand and shrink the different levels, search for contents, display the results graphically, save to various file formats and print.

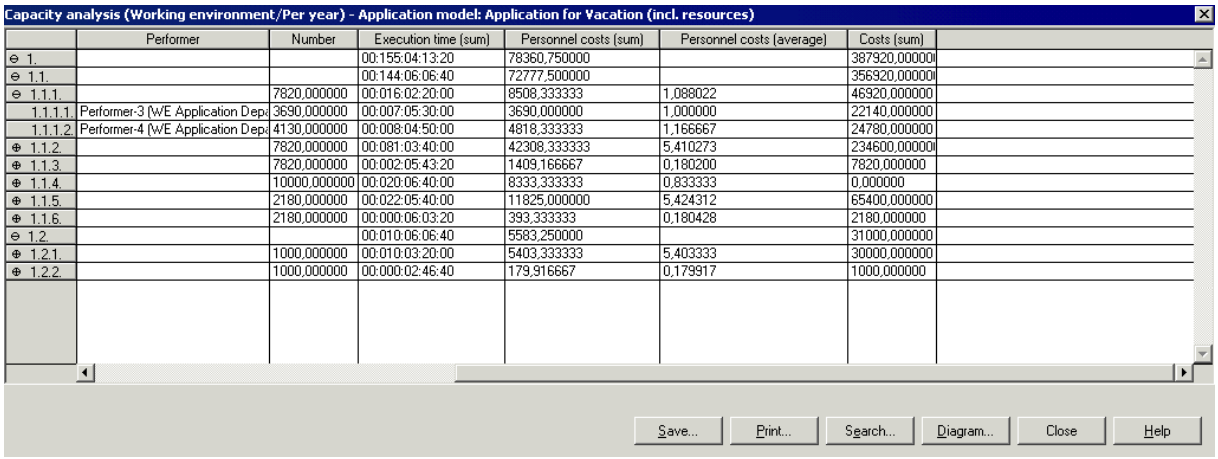


Figure 324: Capacity Analysis - Tabular results display

The **Simulation results** can be displayed as follows:

- Process related (per year, month or process) (see p. 360)
- Person related (per year, month or process) (see p. 360)
- Working environment related (per year, month or process) (see p. 360)

In addition, you can plan your **Personnel Capacity Demand** (see chap. 4.3.4.2, p. 361).

The **resource results** can be displayed as follows:

- Process related (per year, month or process) (see p. 360)
- Working environment-related (per year, month or process) (see p. 360)
- Resource related (per year, month or process) (see p. 360)

The following results are output in the selected hierarchical structure:

- **Number**
- **Resting time**
- **Transport time**
- **Waiting time**
- **Execution time**
- **Cycle time** (only "per Business Process Model")
- **Costs**
- **Personnel costs**

Hint: The names depend on your application library. They can be changed while ADOxx is configured.

You can graphically represent (see chap. 4.3.4.3, p. 361) the times and costs calculated by clicking on the button **"Diagram"**.

Process-related Results

The process-related results are structured as follows:

Level 1: Business Process Models

Level 2: Activities of the Business Process Modells

Level 3: Activities of the Business Process Models executed by single performers or resources

Performer-related Results

The **person-related** results are structured as follows:

Level 1: Performer

Level 2: Business Process Models in which the performer executed activities

Level 3: Activities of the Business Process Models executed by the performer

Working Environment Results

The **working environment-related** results are structured as follows:

Level 1: Grouping class specified - e.g. Organizational unit or Role

Level 2: Business Process Models in which the performers belonging to the grouping executed activities

Level 3: Activities of the Business Process Models executed by the performers belonging to the grouping

Level 4: The actual performers belonging to the specific grouping class who executed the activities

Resource-related Results

The **resource-related** results are structured as follows:

Level 1: Resources

Level 2: Business Process Modells in which the resource was used

Level 3: The activities within the Business Process Models which used the resource

4.3.4.2 Personnel Capacity Planning

As a special **period-related result (see chap. 4.1.7, p. 342)** you can **plan your personnel capacity demand**. This refers to the Business Process Models contained in the application model as well as to a grouping class you may choose (such as "Role" or "Organizational unit").

You can obtain the personnel capacity required for all objects of the grouping class selected. The personnel capacity is derived from the (average) execution time (see chap. 4.1.3.3, p. 339) of the performers assigned to this grouping and the quantities of work modelled (specified by the values in the attributes "Quantity" and "Time period in the objects of class "Process start").

Hint: To view the personnel capacity demand you must select "Per year" or "Per month" in the window "Capacity analysis - application model: <*Model Name*>" (see fig. 323).

The results is displayed in the window "Capacity planning (<*Grouping*>) - application model: <*Model Name*>" (see fig. 325).

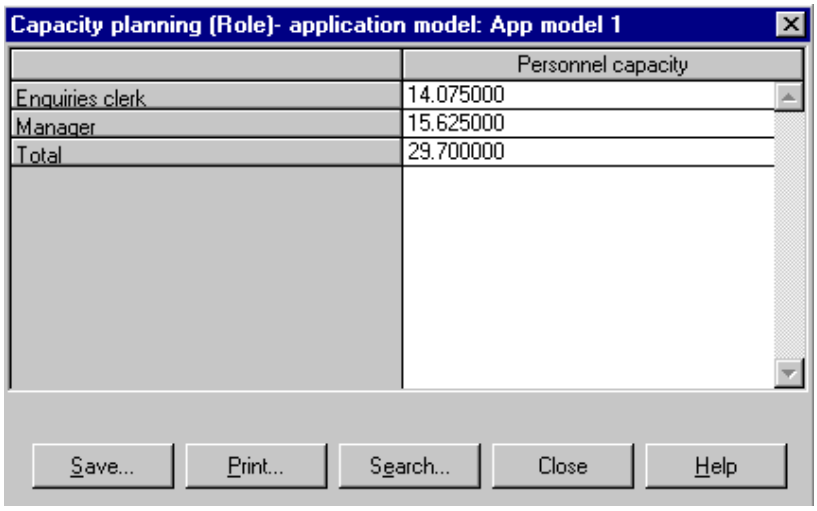


Figure 325: Capacity analysis - Human resources planing

Hint: Please bear in mind that the personnel capacity is a **period-related** (see chap. 4.1.7, p. 342) result which is independent of the length of the period selected. That is, the capacity per year is identical to that of the capacity per month. The reason for this is, that we assume a linear workload. This means that the quantity of work is assumed to increase to the same degree as the working hours available.

4.3.4.3 Representing results graphically (Diagram)

The times and costs calculated by the capacity analysis can be displayed graphically either as bar diagrams (see chap. 8.16.1, p. 87) or as pie charts (see chap. 8.16.2, p. 89).

By using the function "Generate graphics" (see chap. 8.16.3, p. 90) you can copy the diagram into the clip-board, save it into a file or print it out (see p. 51).

4.3.4.4 Transferring the Results of the Capacity Analysis

You can transfer the simulation results of the capacity analysis (see chap. 4.3.3, p. 357) into the business process models simulated. By doing this, you can apply the evaluation mechanisms (see p. 352) available.

Transfer the results of the capacity analysis by clicking on the button **"Evaluation"** in the window "Capacity analysis - application model: <*Model Name*>" (see fig. 323).

You will be asked to confirm that you really want to transfer the simulation results into the models. Clicking on the button **"Info"** gives you some information on the results currently stored in the models (see p. 351).You may then decide, whether or not you want to overwrite them.

The window "Evaluation - get results" (see fig. 319), will be displayed, where you can carry out the settings for taking over.

Activate the option **" Included activity result"**. Activate the option **"included activity result"** to take over every individual result of the activity, that is in the model.

If you select the option **"Complete result"**, the sum of all results will be passed. If you select the option **"Average result"** the average results will be taken over (applied to the process run).

The result of business processes (deactivated option "including the activity result") will be received in the process start object of the "main process".

The activity results (activated option "included activity result") will be entered to the object of the "activity" class of the carried out process if the mentioned option is set.

Once the results have been transferred to the models, (press "Yes" on the confirmation screen), you may use the evaluation mechanisms (see p. 352) "Queries" and "Pre-defined Evaluation Queries".

> **Hint:** If you have already transferred simulation results to the models you are simulating and you are attempting to re-transfer identical results, the system will inform you of this and you will not be permitted to over-write the results-
>
> **Hint:** Transferring simulation results changes the contents of the models simulated. These are then marked as "modified".

You can delete the simulation results transferred using the option "Delete simulation results" (see chap. 4.7, p. 390) in the "Edit" menu or the appropriate smart icon.

4.3.5 Background of capacity analysis

4.3.5.1 Input

Capacity analysis is based on an application model (see chap. 2.4.16, p. 284). An application model can be defined in the "Model" menu using the option "Application Models". It consists of **exactly one** Working Environment model and **at least one** Business Process Model (any subprocesses are automatically included).

> **Hint:** Any subprocesses called from the main Business Process Model(s) are automatically added to the application model.

In order to run a capacity analysis, not only must the process flow be modelled correctly (as in the case of path analysis (see chap. 4.2, p. 343)), but in addition the performers of the various activities must be defined (see chap. 2.2.12.14, p. 222). The latter is done by entering an AQL expression into the attribute "Performer" of each activity. Please note especially the dynamic AQL expressions such as "done by" (see p. 798). It is also possible to assign performers depending on certain current variable values (see p. 795).

If you have modelled resources in your Business Process Models, then you must also assign an actual resource from the working environment model (resource assignment (see chap. 2.2.12.15, p. 227)). This is done by entering an AQL expression in the attribute "Selection" in objects of the class "Resource". Please note especially the dynamic AQL element "current performer" (see p. 798).

4.3.5.2 Processing

A capacity analysis simulates each Business Process Model contained in the application model as many times as specified in the number of simulations (see chap. 4.3.1.1, p. 356). The results can then be examined either as process or period related (see chap. 4.1.7, p. 342).

In contrast to path analysis (see chap. 4.2, p. 343), each activity is assigned to a performer (see p. 363) and, where necessary, resources assigned to activities (see p. 364) are evaluated.

As the activities are assigned to performers, the capacity of each performer can be identified

> **Hint:** Please bear in mind that the personnel capacity is a **period-related** (see chap. 4.1.7, p. 342) result which is independent of the length of the period selected i.e. the personnel capacity per year is identical to the personnel capacity per month. The reason for this is that a linear workload is assumed. This means when examining a longer period of time we assume the quantity of work increases to the same degree as the working hours available.

In addition, the personnel costs (see p. 366) and the resource costs (see p. 366) of the activities and the Business Process Models are calculated. In order to do this, the activities' execution times and the hourly wages of the performers and resources are used.

> **Hint:** The waiting times entered into the notebooks of the activities serve as input to the capacity analysis as they do in the path analysis (see chap. 4.2, p. 343). Only the simulation algorithm "Workload Analysis" (see chap. 4.4, p. 367) calculates the waiting and cycle times dynamically.

Assigning activities

In the course of the capacity analysis, the activities to be executed are assigned to performers of the Working Environment model (objects of class "Performer") according to the entries in the attribute "Performer" (in the objects of the class "Activity").

> **Hint:** If a standard performer assignment (see p. 225) has been carried out for a sub model, all activities without entries will be delegated to the attribute "Performer" according to the standard performer assignment.

Performers are assigned to each activity with the help of AQL expressions (AQL = **A**DONIS **Q**uery **L**anguage) (see chap. 14., p. 790).

Evaluation of the AQL expression in the "Performer" attribute renders a set of performers from the Working Environment model assigned. One of these performers is selected **at random** (see p. 226) from this set and the activity is then assigned to him/her.

The attribute "Availability" in objects of the class "Performer" is evaluated, when activities are assigned to performers. In this attribute you can specify, whether a performer works full-time or part-time. This is specified as a number between 0 and 100 - interpreted as a percentage.

Example:

The value of the attribute "Performer" in the objects "Activity-1" of the class "Activity" is:`{"Specialist"} <- "Has role"`

Evaluating this AQL expression identifies the performers "Performer-1", "Performer-2" and "Performer-3". The values in the attribute "Availability" are *"50%" for "Performer-1"* and *"100%" for "Performer-2" and "Performer-3"* each.

The "probability" of performing the activity, then, is only half as great for "Performer-1" as for "Performer-2" and "Performer-3". "Performer-2" and "Performer-3" each have the same "chance" to have this activity assigned to them for execution.

Thus, "Activity-1" is **at random** assigned to "Performer-1" with a probability of 0.2 and to "Performer-2" or "Performer-3" with a probability of 0.4 each.

Therefore, although the activities are assigned to performers randomly - the availabilities are taken into account.

How to treat cooperative activities

A cooperative activity is an activity which must be executed by more than one performer (e.g. a meeting). In the capacity analysis, the times and costs of cooperative activities are multiplied by the number of participating performers to calculate the results.

ATTENTION: Cooperative activities shall not be assigned to resources (see p. 364), since it would abort the simulation.

Example:

A **cooperative** activity has an execution time of 10 minutes (attribute value: "00:000:00:10:00"). The evaluation of the performer assignment expression during the capacity analysis results in "Performer-1", "Performer-2" and "Performer-3". Therefore 30 minutes is assumed as the execution time of this activity.

Hint: When the cycle time is calculated, only 10 minutes execution time is taken into account as the cooperative activity was executed by all three performers **simultaneously**.

Specifying resources assigned

Resources can be assigned if required to any activities within your Business Process Models. This is done by connecting the activity to the resource object with the "Uses" relation.

Hint: It is possible that the application library you are using has been customised to meet your specific needs and therefore contains several classes of resources such as PC, telephone, fax, type-writer and so on. All resource classes are connected to objects of the class "Activity" by the relation "Uses".

The resources in the Business Process Models contain the attribute "Selection", into which an AQL expression (see chap. 14., p. 790) is entered. This AQL expression describes which type of resource (from the Working Environment model) is needed to execute the activity. Please note especially the dynamic AQL element "current performer" (see p. 798).

ATTENTION: Resources shall not be assigned to cooperative activities (see p. 364) resources, since it would abort the simulation.

For the capacity analysis (and also for the Workload Analysis (see chap. 4.4, p. 367)) the resources are evaluated as follows during the simulation run:

If the current activity is connected to a resource by the relation "Uses", we distinguish between two cases:

- **Case 1:**

When the resource object is reached for the first time in the process, the AQL expression in the attribute "Selection" is evaluated. This evaluation identifies a number of resource objects from the Working Environment model. From this set one resource is selected **randomly** (in the same way that activities are assigned to performers (see p. 363)).

ATTENTION: In the case of the workload analysis (see chap. 4.4, p. 367) the resource with the shortest waiting queue (determined as sum of the execution times of waiting activities) is selected.

Now a concrete resource is assigned to the resource object in the Business Process Model. This assignment is kept until the business process just simulated is finished. This mechanism works similarly to that which assigns values to variables.

- **Case 2:**

  If a concrete resource from the Working Environment has already been assigned to the resource object (in the Business Process Model), this resource is used for the execution.

The mechanism described above ensures that for different activities one and the same resource is used. This can be compared to the AQL expression and "done by" (see p. 798) used in the assignment of performers (see chap. 2.2.12.14, p. 222) to activities.

ATTENTION: Please notice the difference between the resources in Business Process Models and in Working Environment models. The resources in the Business Process Models serve as references for the resources needed and are filled with concrete values (= resources from the Working Environment model) during the simulation. The resources in a Working Environment model symbolise "real" resources (PC, fax, type-writer and so on), which are available for the performers.

Calculating the performers' capacity

A performer's capacity is determined by the relation between his **workload** and his **working time** It is a period-related (see chap. 4.1.7, p. 342) result of the capacity analysis (see chap. 4.3, p. 354).

A performer's **workload** is the sum of the execution times of the activities he executed in the given period of time.

A performer's **working time** is calculated by multiplying the company time (see chap. 4.1.3.1, p. 335) by the performer's Availability.

Hint: Please bear in mind that capacity is a **period-related** (see chap. 4.1.7, p. 342) result which is independent of the length of the period selected. That is, the capacity per year is identical to the capacity per month. The reason for this is, that we assume a linear workload. This means that over a longer period, the quantity of work is supposed to increase to the same degree as the working hours available.

Example:

The performer "Performer-1" has an availability of 50%. The company time (see chap. 4.1.3.1, p. 335) is "180 days per year" and "8 hours per working day". The performer "Performer-1" must carry out activities with a total execution time of 60 days (attribute value "00:060:00:00:00") each year. The performer's personal capacity is calculated as follows:**`(Personal Workload per Year) ÷ (Working Days per Year)`**That is`60 Days ÷ (180 Days) × 0,5 = 0,66`

Hint: Capacity may be greater than 1 where the performer is overloaded.

Calculating Personnel and Resource Costs

Each performer (object of the class "Performer") in a Working Environment model has the attribute "Hourly wages". This attribute is used to calculate the personnel costs of activities and Business Process Models.

The personnel costs of an activity during the simulation are calculated as follows:**(Execution Time in Hours) × (Hourly Wages of the Current Performer)**

The **expected personnel costs** of an activity result from the average personnel costs of all the activities executed during the simulation.

The **expected personnel costs** of a Business Process Model are derived from the sum of the activities' expected personnel costs weighted against the activities' average frequency of occurrence.

The resource costs of activities and Business Process Models are calculated similarly to the personnel costs. Instead of the attribute "Hourly wages" in objects of the class "Performer" the attribute "Hourly wage" in objects of the class "Resource" is evaluated.

Converting process-related into period-related results and vice versa

Each object of the class "Process start" in a Business Process Model contains the attributes "Quantity" (whole number) and "Time period" (Values: "Per day", "Per month", or "Per year").

The period-related results (see chap. 4.1.7, p. 342) of a Business Process Model are calculated based on the values of the attributes "Quantity" and "Time period".

Example:

A process has personnel costs of 200. The company time is 220 days per year. The values in the attributes "Quantity" and "Time period" are "10" and "per day".

Therefore, the personnel costs for the process per year are calculated as follows:**(Costs for a Process) × (Number per Year)** = = 200 × 220 × 10 = 440,000

4.3.6 Application Scenario (Example)

The capacity analysis enables you to calculate the **costs for a business process - divided into personnel costs and other costs**.

Each object of the class "Performer" in a Working Environment model contains the attribute "Hourly wages". Each object of the class "Activity" contains the attribute "Execution time" which specifies how long it takes to execute this activity. Other costs such as fixed costs which are directly related to the activity can be entered into the "Costs" attribute.

When the capacity analysis is run, the activities are assigned to the available performers.

Therefore, the total costs for executing a concrete activity are:**(Value of the Attribute "Costs" of the Activity) + + [(Execution Time of the Activity) × × (Hourly Wages of the Performer selected)]**

Therefore you can obtain the personnel and other costs as a direct simulation result.

4.4 Workload analysis (steady-state and fixed time period)

The workload analysis supports you when you want to analyze the dynamic behaviour (see chap. 4.4.7, p. 385) of your organization. Waiting times for activities are not defined in advance but calculated by the simulation, based on a waiting queue model. The stochastic arrival times (starting frequency) of each process are described in the process calendar (see chap. 2.2.12.17, p. 235) of the various processes. Additionally, the exact availability of each performer can be entered into a calendar (see chap. 2.2.12.16, p. 228) - see Calendar attribute in the Performer object in a Working Environment model. Depending on the processes' frequency of occurrence and the assignment of performers to objects of the class "Activity" it is possible to identify workload profiles of organisational units and roles as well as to recognise bottlenecks and idle times. You can obtain both process and period related (see chap. 4.1.7, p. 342) results.

The input (see chap. 4.4.6.1, p. 376) for the workload analysis is an application model (see chap. 2.4.16, p. 284). If you execute a workload analysis (see chap. 4.4.1, p. 367), the Business Process Models of the application model selected are simulated in the working environment model that is part of the application model.

The **difference between the workload analysis (steady-state) and the workload analysis (fixed-time period)** can be characterised as follows:

"In the case of the **steady-state view** a **fixed number of process runs** are simulated, regardless of how much time is needed until the processes are executed. The results calculated are the average values with regard to one process run."

"In the case of the **fixed-time period view** a **predefined period of time** is simulated, regardless of how many processes will be executed during this period of time. The results can be examined per process or over the evaluation phase specified"

4.4.1 Carrying out a workload analysis

The following steps can be followed to run a Workload analysis:

1. Select either "Workload analysis (steady-state)" or "Workload analysis (fixed-time period)" in the "Algorithm" menu. Alternatively click on the appropriate smart-icon for either workload analysis (steady-state) or workload analysis (fixed-time period).

   The window "Simulation - Workload analysis" (see fig. 314) will appear.

2. Select the application model you wish to simulate from the "Application Model Selection" list. Clicking on the **"Add"** button, allows you to define new application models. Clicking on the **"Refresh"** button will refresh the model selection list.

3. Make the following entries:

   - Numbers (see chap. 4.4.1.1, p. 369)
   - Options (see chap. 4.4.1.2, p. 370)
   - Passive Components (see chap. 4.4.1.3, p. 370)

4. Confirm your entries by clicking on the OK button.

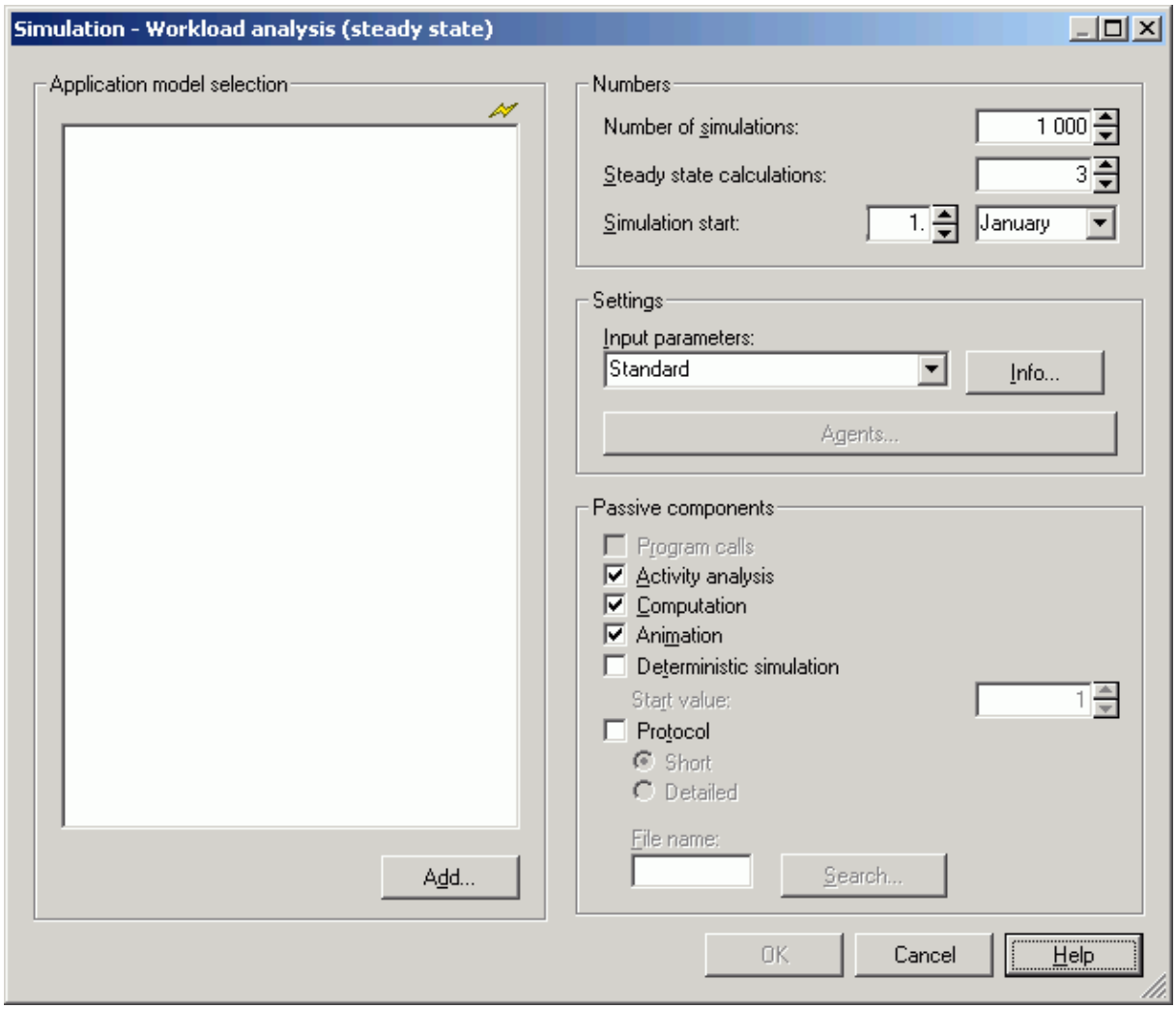


Figure 326: Carrying out a workload analysis (steady state)

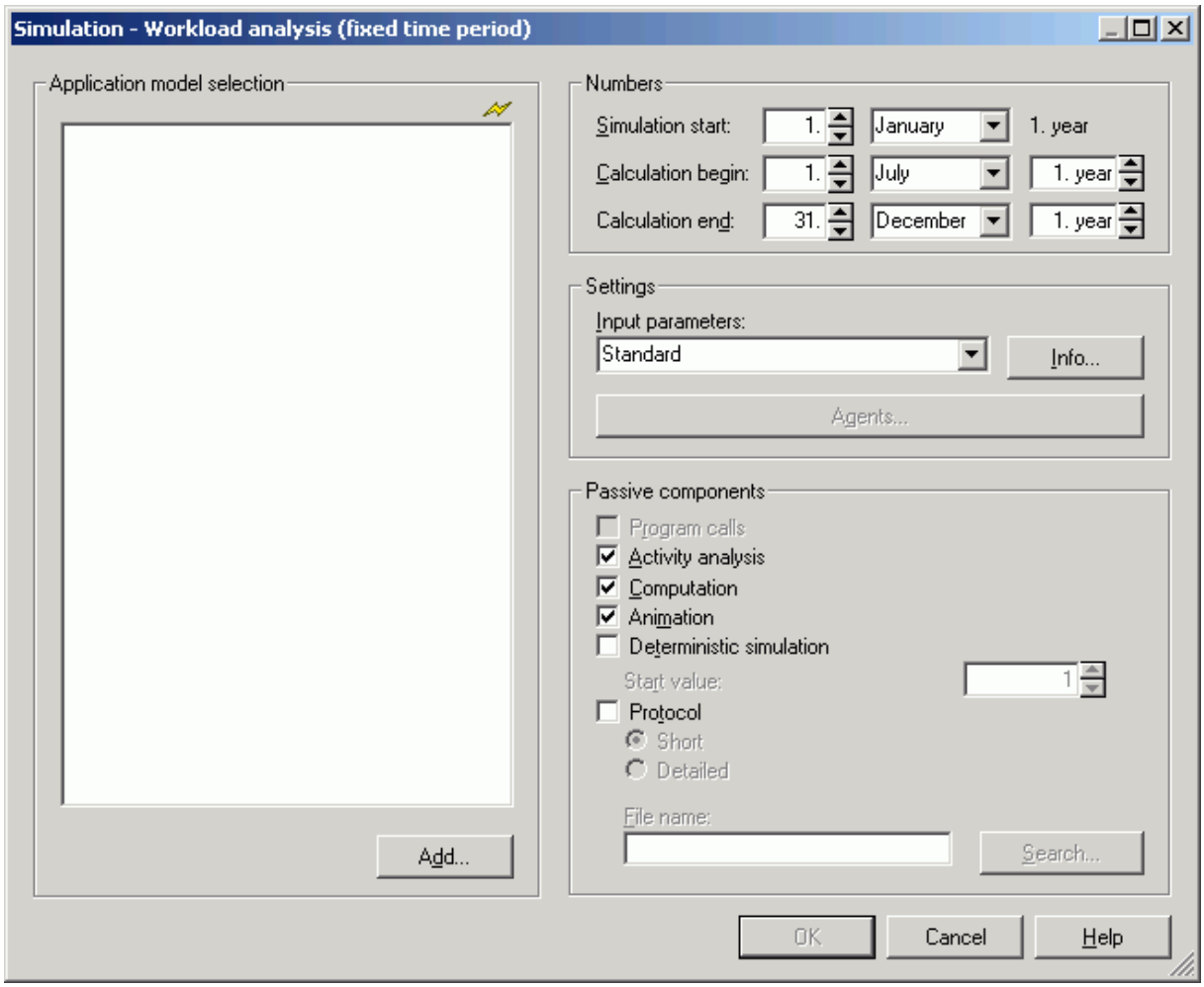


Figure 327: Carrying out a workload analysis (fix time period)

Hint: By double-clicking on an application model in the model selection list you can start the workload analysis based on this model and the current numbers, settings and passive components.

4.4.1.1 Numbers

Steady state View:

"Number of simulations" this indicates how often each process contained in the application model is to be simulated. The number of simulations is a measure of accuracy, i.e. the higher the number, the more exact the simulation results will be. Note - the higher the number of simulations, the longer the simulation will last.

"Steady state calculations" Indicates how many attempts must be made to reach a "steady state". During each attempt (oscillation phase) the number of simulations specified above will be run (see p. 379). There must be at least 3 oscillation phases.

"Simulation start" Specifies the (fictitious) start of the simulation (i.e. the first oscillation phase).

Hint: The simulation results are not measured (see p. 380) from the time the simulation starts - the actual results are measured AFTER the last oscillation phase (see p. 379) is completed.

Fixed time period view:

"Simulation start" specifies the start date of the simulation.

"Calculation begin" specifies the start date for the result measuring phase. The period between the "Simulation start" and the "Calculation start" is the initialisation phase (see p. 380). The system simulates during the initialisation phase but no results are measured at this point.

"Calculation end" defines the end of the simulation. The simulation results are calculated in the period between "Calculation start" and "Calculation end". The period selected should be at least long enough to completely execute some processes. The longer the period is, the more accurate the results of the simulation will be. It is possible that no results will be calculated after the simulation if the period of time selected is too short.

Hint: February has 28 days, i.e. leap years are not taken into account.

Hint: The "Simulation start" must be earlier than the "Calculation start" which must be earlier than the "Measuring End".

4.4.1.2 Options

The field **"Input Parameters"** allows you to select which attributes should serve as input (see chap. 4.1.3.2, p. 335) to the workload analysis (see chap. 4.4, p. 367) and therefore which results will be calculated. When you click on the **Info** button, the current input parameter combination is explained.

Click on the button **"Agents"** to open the Agent Overview (see chap. 4.8.2, p. 392) of the models belonging to the currently selected application model.

4.4.1.3 Passive components

"Program calls" Default setting: deactivated.

If this option is activated, program calls specific to the activity will be carried out during the simulation of each activity. The selected input parameter combination (see p. 337) will determine which program calls will be concerned by this .

"Activity analysis" Default setting: activated.

If this option is activated, the results of the workload analysis are calculated. If you deactivate the option **"Activity Analysis"**, you will **not** obtain results "per process" (see chap. 4.1.3.3, p. 339).

"Computation" Default setting: activated.

If the option is activated, the results of the workload analysis are calculated. If you deactivate the option **"Computation"**, you will obtain **no** results at all.

Hint: It can be sensible to deactivate the "Activity analysis" and "Computation" options in situations where you simply want to generate a simulation protocol but do not need to see the specific results.

"Animation" Default setting: activated.

If the option is activated, the execution of activities in the Working Environment model is visualised (see chap. 4.4.2, p. 371) during the workload analysis by task stacks.

"Deterministic simulation" Default setting: disabled.

When enabling this option the simulation is initialised with the same start value. This ensures that with the same start value independent simulation runs will determine the same simulation results. To obtain identical results with the deterministic simulation the content of the models to be simulated must not be changed.

"Protocol" Default setting: deactivated.

If the option is activated, a simulation protocol (log) is generated in the file you specified.

Select the **short version** if you would like to use the log file generated in the offline animation (see chap. 4.5, p. 386) later on. The short version of a protocol is stored in the compressed format APF (APF = **A**DONIS **P**rotocol **F**ormat).

Select the **detailed version** to generate a more detailed protocol. After the simulation you can open the protocol file with a text editor and trace each step.

4.4.2 Simulation run

Loading of additional models and the treatment of any errors during the simulation is, as described for path analysis (see chap. 4.2.2, p. 346). Transferring simulation results back to the model is as described for capacity analysis (see chap. 4.3.4.4, p. 362)

Oscillation Phases (only in the case of the **steady-state** view)

Once the models have been loaded, the oscillation phases (see p. 379) are carried out. This is shown by the title of the status window.

If a steady state has not been reached after the number of attempts you specified, you will be asked, whether you wish to carry out the simulation anyway.

Click on the "Yes" button to calculate the simulation results of the workload analysis (steady-state view) (see chap. 4.4.4, p. 373). When clicking on the "No" button to cancel analysis.

Hint: Please note that the simulation results calculated based on an unsuccessful steady-state analysis may be distorted and should only be examined with care.

Animation

If you selected the passive component "Animation" (see chap. 4.4.1.3, p. 370) in the start window of the workload analysis, the task stacks of the different performers are represented graphically in the Working Environment model during simulation (see fig. 328).

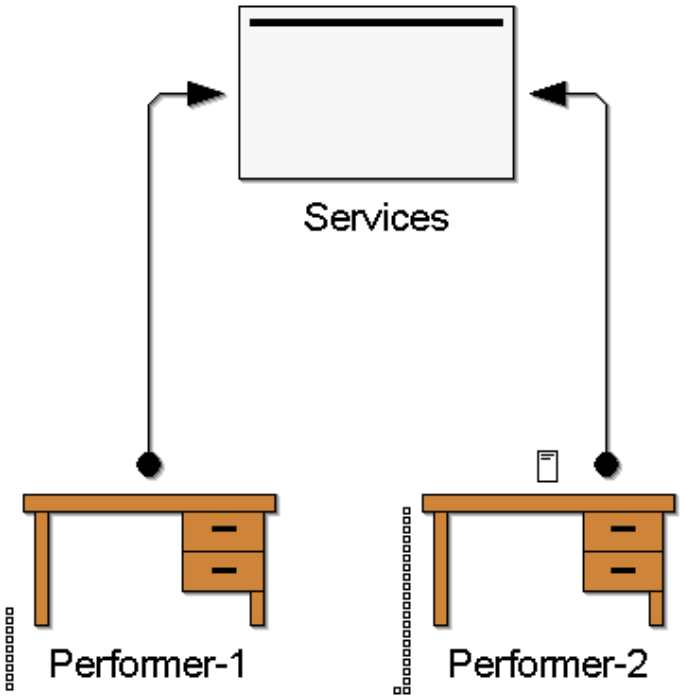


Figure 328: Workload analysis - animated task stack

This shows you the performers' workload during the simulation run. When you double-click on a performer, the content of his task stack (see chap. 4.4.3, p. 372) is shown.

If necessary, you can bring the Working Environment model to the foreground using the menu "Window".

In the **status window** of the workload analysis you can control the animation's speed.

A delay factor can be set in the field "Delay". By default, this is set to "**0**".

By clicking on the button "Pause" you can interrupt the simulation and then continue it step by step by clicking on the button "Step". Clicking on the button "Pause" again will cause the simulation to continue as normal.

4.4.3 Task stacks for performers

If you wish to obtain further information about a performer's task stack during the simulation, interrupt the simulation by clicking on the option "Pause" and then double-click on the object of class "Performer" in which you are interested.

The task stack is shown in its current order in the ADOxx browser (see chap. 8., p. 66) - i.e. item No. 1 has been in the queue for the longest length of time and should be executed next. The different activities which are part of the performer's task stack are listed (see fig. 329).

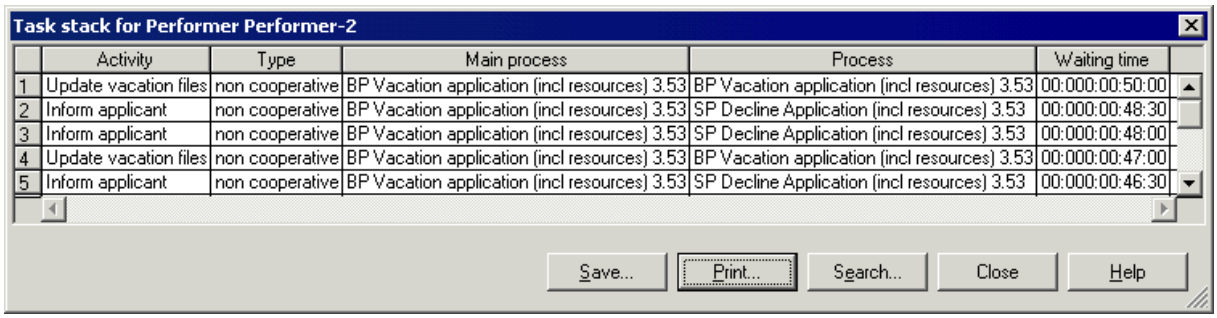


Figure 329: Task stack display

For each activity, its name and type, its main process, its actual process (i.e. could be in a subprocess) and its waiting time are shown. An activity's type may be "non cooperative", "coop ., synchronous" or "coop., asynchronous".

The column "Main Process" shows theBusiness Process Model contained in the application model. The column "Process" specifies the process model in which the activity exists.

If no task stack is assigned to the performer at this point of the simulation (i.e. his queue is empty), or if you have clicked on an object that does not belong to the class "Performer", an appropriate message will be displayed.

4.4.4 Results of the workload analysis

The results of the workload analysis are structured similarly to those of the capacity analysis (see chap. 4.3.3, p. 357).

Hint: You can obtain the performers' workloads in the "person-related results" with a period-related (see chap. 4.1.7, p. 342) period of time.

Hint: In contrast to the capacity analysis, no direct personnel capacity planning is possible, since this depends on the criteria of the user. The user has to define which waiting and cycle times (see chap. 4.1.3.3, p. 339) are acceptable from his point of view and at what point additional performers are necessary.

4.4.5 Representing results

After the simulation run the result window "Workload Analysis - Application Model: <*Model Name*>" (see fig. 330) appears.

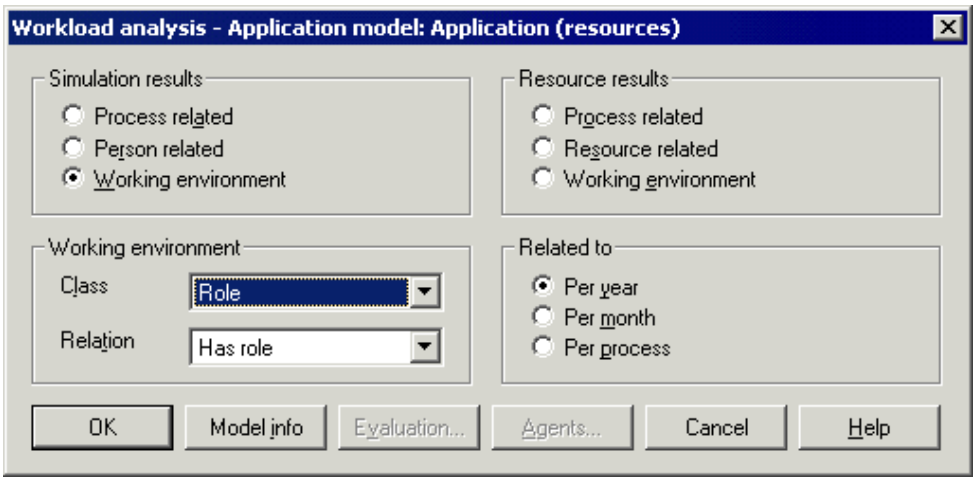


Figure 330: Workload analysis - select results

In this window you can define the reference type (per year, per month or per process), the result type (simulation or resource related (see chap. 4.3.4.1, p. 359)) and the structure (process related, person related or working environment).

For the **workload analysis (steady-state view)** you can select as reference **"Per process"**, **"Per month"** or **"Per year"**.

For the **workload analysis (fixed time period)** you can select as reference the options **"Per process"**, or **"Evaluation period"**. If you select the option "Evaluation period", you obtain the results referring to the period of time between "Calculation begin" and "Calculation end" which you specified before the simulation started. This period of time is again shown beside the option "Evaluation Period".

> **Hint:** If no resources have been assigned to the activities of the Business Process Models, there will be **no** resource results and the result window will appear without the options for the resource results.

By clicking on the different options you can define the result type, your preferred structure and the reference.

When you select the option **"Working Environment related"** (both for simulation and for resource results), the fields "Class" and "Relation" in the "Working Environment" group are activated. Select the grouping class (e.g. "Organizational Unit") and the relation which exists between the performers of the Working Environment model and the grouping objects (e.g. "Belongs to"). The results of those performers which belong to the grouping selected are then shown in aggregated form.

If one or more performers do not belong to exactly one grouping, an appropriate statement will be displayed, since this means that the results shown may not include all performers and therefore should be interpreted carefully.

Clicking on the button **"Model Info"** informs you which Business Process Models and which Working Environment model are contained in the application model simulated.

Once you are happy with the settings, click on the "OK" button. The results will then be displayed in tabular form in the ADOxx browser (see chap. 8., p. 66).

You can represent the results in different hierarchy levels (see chap. 8.7, p. 76), display them and save (see chap. 8.13, p. 81) them in various formats to a file.

By clicking on the button **"Evaluation"** you can transfer the simulation results into the models.

Should you have defined agents (see chap. 4.8, p. 391) to calculate user-defined results, you can obtain these agents' results (see chap. 4.8.6.2, p. 424) by clicking on the button **"Agents"**.

By clicking on the "Cancel" button in the result window you can leave the representation of results. An informatory message will be displayed to tell you that the simulation results will be deleted, if you close the window.

Once the **workload analysis is finished**, (i.e. after the result window has been closed), you will, if necessary, be asked, whether any models loaded specifically for the simulation should be closed again. Please note that any models loaded for this reason are loaded **write-protected**.

4.4.5.1 The display of results in a tabular form

The results of the workload analysis are divided - as are those of the capacity analysis - into simulation and resource results.

In the case of the simulation results performers are assigned to the different activities and Business Process Models. In the case of the resource results, resources (such as PC, type-writer etc) are assigned to the different activities and Business Process Models.

The results may be displayed (see fig. 331) in the ADOxx browser related to a single process or to a period of time. You may expand and shrink the levels, search for contents, display the results graphically, save them to various file formats and print them out.

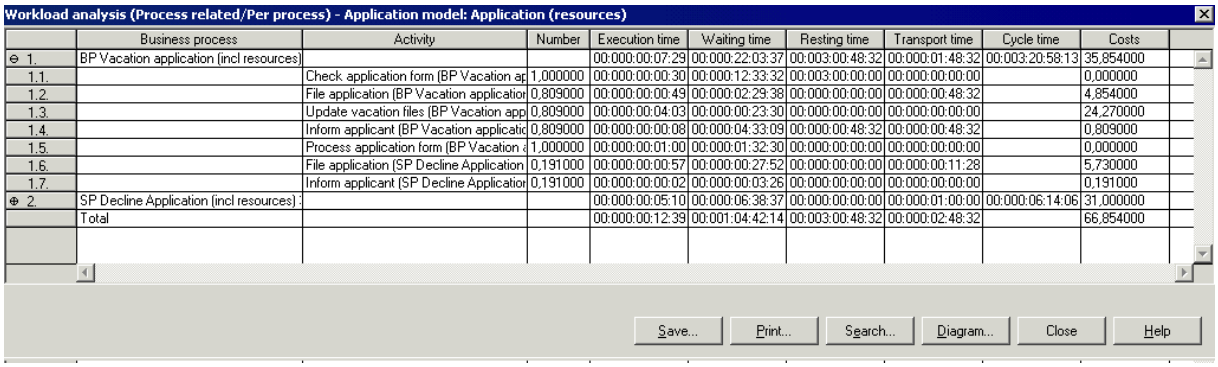


Figure 331: Workload analysis - tabular result display

The **simulation results** can be displayed as follows:

- Process elated (per time period or per process) (see p. 360)
- Person related (per time period) (see p. 360)
- Working Environment-related (per time period) (see p. 360)

The **resource results** can be displayed out as follows:

- Process related (per time period) (see p. 360)
- Resource related (per time period) (see p. 360)
- Working Environment related (per time period) (see p. 360)

4.4.5.2 Transferring the results of the workload analysis

The results of the workload analysis are taken over in the same way as the results of the capacity analysis (see chap. 4.3.4.4, p. 362).

4.4.6 Background of the workload analysis

4.4.6.1 Input

The input for the workload analysis is, similarly to the capacity analysis (see chap. 4.3, p. 354), an application model (see chap. 2.4.16, p. 284).

The principal difference between the capacity and the workload analysis is that the workload analysis calculates the **waiting times of the activities** and thus also cycle times of the processes using the **workloads of the performers and resources**. If an activity is assigned to a performer or to a resource, while he is still busy executing a different task, waiting time is caused.

The workload analysis needs further functionality (see chap. 4.4.6.2, p. 376) in addition to the mechanisms of the capacity analysis (see chap. 4.3.5.2, p. 363)

4.4.6.2 Processing

The workload analysis simulates on the "**time axis**", in contrast to the "Path Analysis" (see chap. 4.2, p. 343) and "Capacity Analysis" (see chap. 4.3, p. 354) algorithms.

Further, in contrast to the capacity analysis (see chap. 4.3, p. 354), the waiting period of the activity and process run over the work load of the person or the resource activity will be calculated in the workload analysis. If a person or a resource receives an activity, while the others have already been edited, waiting periods occur (see p. 336).

The **workload analysis (steady-state)** simulates each Business Process Model contained in the application model (see chap. 2.4.16, p. 284) at least as many times as you specified. This number is a measure for the accuracy of the results.

The **workload analysis (fixed time period)** simulates each Business Process Model contained in the application model (see chap. 2.4.16, p. 284) for the period of time you specified: you enter the date for the start of the simulation, for the start of the result measuring period and for the end of the simulation.

The results can be examined as either process or period related (see chap. 4.1.7, p. 342).

Simulation on the time axis allows the number of activities a performer has to execute vary as the processes are started at random. For this reason, further functionality is needed in addition to the mechanisms of the capacity analysis (see chap. 4.3.5.2, p. 363):

- Triggering Processes (see p. 377) (Process calendar)
- Processing of Activities (see p. 378) (Performer calendar)
- Oscillation Phases (see p. 379) (only for steady-state view)
- Length of the Oscillation Phases/Result Measuring (see p. 380) (only for steady-state view)
- Initialisation Phase (see p. 380) (only for fixed time period view)
- Tolerance waiting time (see p. 381)
- Assignment of Activities to Performers (see p. 381)
- Continuous Execution (see p. 382)
- Treatment of Cooperative Activities (see p. 382)
- Treatment of uninterruptible activities (see p. 383)
- Determining Resources Assigned (see p. 384)
- Processing of Activities (see p. 384) (Priorities and Maximum Resource Waiting Times)

- Calculation of the Performers' Workloads (see p. 385)

Trigger off processes: process calendar

To trigger off the processes, the "Process Calendars" (see chap. 2.2.12.17, p. 235) defined in objects of the class "Process start" is evaluated.

A **process calendar** refers to one year. A year consists of 365 days. A "day profile" (see fig. 332) is assigned to each day.

A **day profile** consists of a series of time intervals to each of which a distribution is assigned. The possible distributions available for use are Uniform distribution or Exponential distribution.

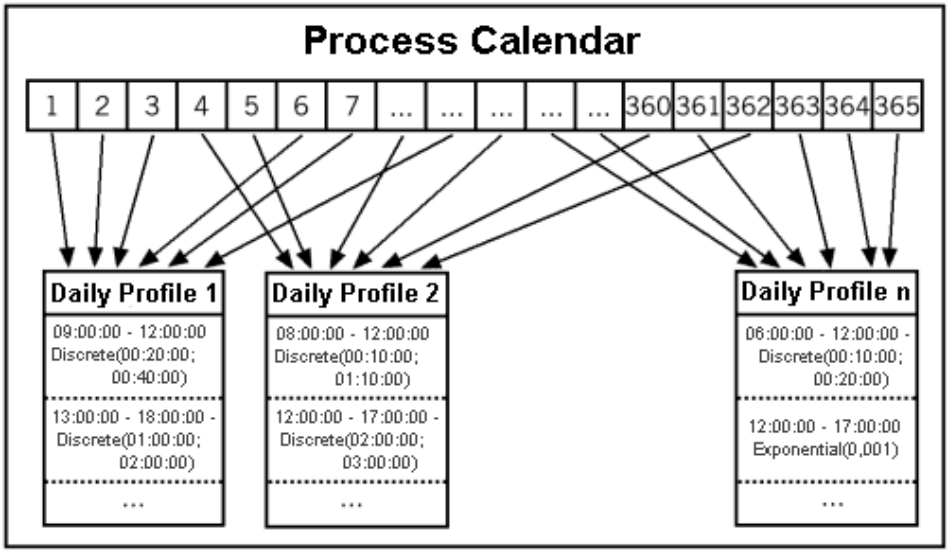


Figure 332: Process calendar (schematic display)

In the course of the simulation, the processes are started according to the distribution that is assigned to the current time interval (see fig. 333).

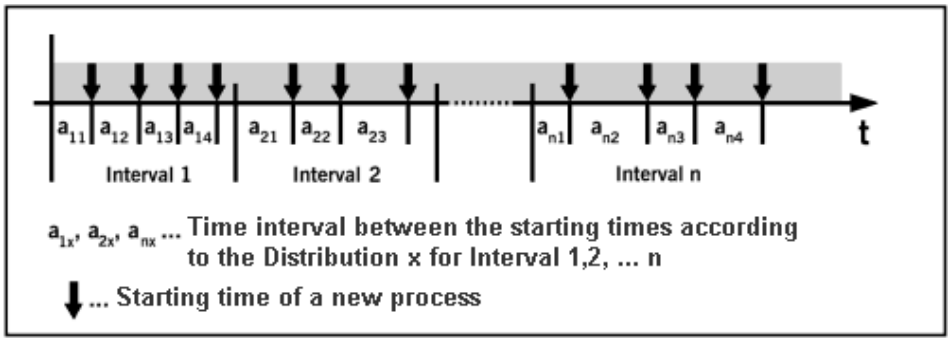


Figure 333: Start time for processes

Example:

"Day Profile 1" is assigned to the first day (see fig. 332). Between 0 and 9 o'clock no process is started, since no time interval is defined here.

At 9.00 the uniform distribution "Uniform (00:20:00;00:40:00)" specified is evaluated. The number 00:35:00, for example, is generated at random. That means the process is started at 9.35. Now the

next number is generated at random, let us say 00:25:00, and added to the last start point. The next process will thus be started at 10.00.

The processes will be started until a start point exceeds the current time interval. In the example above, this would be a start point later than 12:00.

We distinguish two cases:

- **Case a):** The new start time exceeds a time interval of the daily profile (e.g.:12.30)
- **Case b):** The new start time lies within a later time interval (in this example the interval 13:00 - 18:00).

In **case a)** the process is started at the time calculated and then continued in the next time interval (see fig. 334).

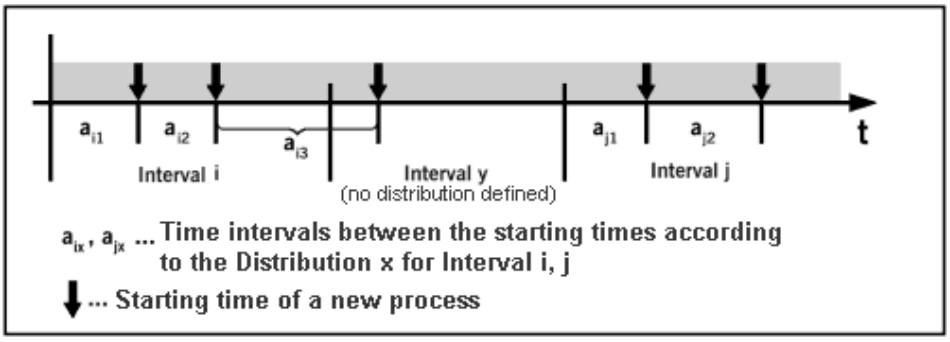


Figure 334: Example of process impulse (case a)

In **case b)**, starting at the new interval start a start point is generated for all the time intervals "in between" and the minimum of these start points generated is chosen for starting the process (see fig. 335).

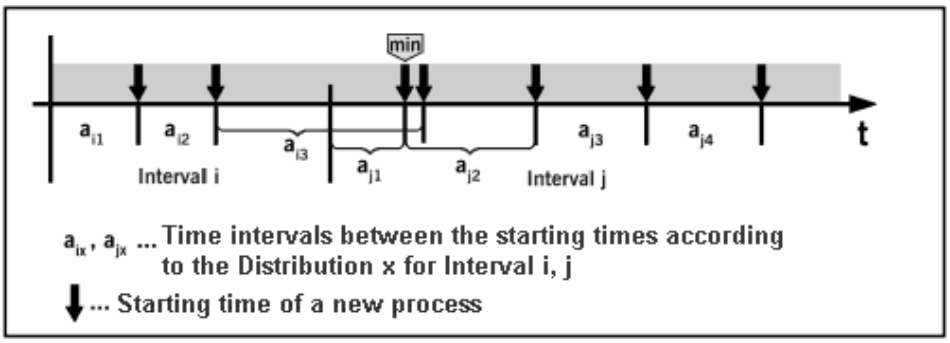


Figure 335: Example of process impulse (case b)

Hint: The process calendar is evaluated for both the **steady-state view** and the **fixed time period view**.

Processing activities: agent calendar

In the same way that process calendars (see p. 377) are assigned to the processes, calendars are also assigned to the different performers.

These calendars have the same structure as the process calendars (see chap. 2.2.12.17, p. 235). However, no distributions are assigned to the time intervals. A time interval describes the time during which a performer is present - i.e. available for work.

A performer can only execute activities while he is present. If a performer is absent, the activities in the performer's task stack will accumulate waiting times (see chap. 4.1.3.2, p. 335).

If a performer begins an activity but is not able to complete it in one sitting - e.g. it is the end of the days work or the beginning of the weekend as indicated by his calendar, then the activity is interrupted and continued at that point in time at which the performer is available again.

Hint: The agent calendar is evaluated both for the **steady-state view** and for the **fixed time period view**.

Oscillation phases (steady-state view)

When running a workload analysis, the simulation begins in an "empty system", i.e. the different performers of the working environment assigned have no activities to be executed on their "desks". For this reason, it does not make sense to start measuring the results as soon as the simulation starts.

Therefore for the **workload analysis (steady-state view)** the results are calculated following the so-called **"oscillation phases" (see p. 380)**. During these oscillation phases the simulation runs but no results are calculated.

Hint: In the **workload analysis (fixed time period)** an initialisation phase (see p. 380) replaces the oscillation phases.

As soon as a steady state is reached, the results are measured. The term "**median waiting stack sum**" (MWSS), which is defined as follows, is needed to determine the steady state (see fig. 336).

$$MQS = \frac{\sum_{\text{Activity}} (RT + TT + WT + ET)}{(\text{Convergence phase})}$$

Figure 336: Median waiting stack sum (MWSS)

The MWSS is calculated by taking the sum of the "life times" of all activities which occurred during the oscillation phase - that is the sum of resting time plus transport time plus waiting time plus execution time - divided by the length of the oscillation phase.

Therefore the important quantities for the MWSS are the "length of the oscillation phase" (see p. 380) and the waiting times of the activities.

The **steady state** is defined as follows:

For each phase the "MWSS of the Working Environment" is determined. If the MWSS lies between the minimum and the maximum (calculated based on the previous oscillation phases), the system is regarded as having reached a steady state and measuring of results starts.

After the first oscillation phase, the minimum and the maximum are set to the MWSS calculated. After the second oscillation phase, either the minimum or the maximum is updated. If the MWSS lies between the minimum and the maximum after the third oscillation phase, the system is regarded as having reached a steady state and the measuring of results starts. In any other case, the minimum or the maximum is updated and the fourth oscillation phase takes place. This procedure is carried out until a steady state is reached or the number of attempts to reach it exceeds the number you specified (see fig. 337).

Hint: The procedure described above explains why at least three oscillation phases have to be executed.

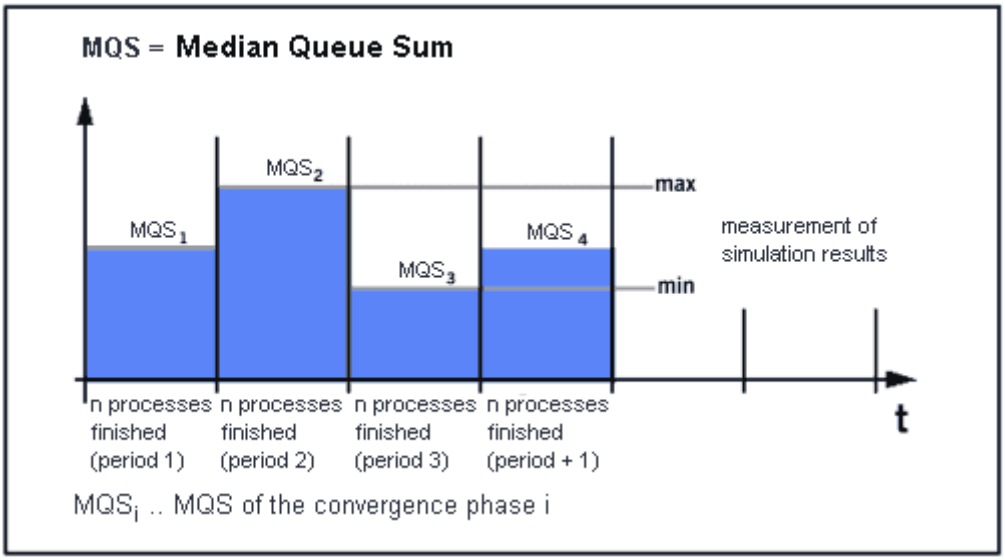


Figure 337: Oscillation phases

Example:

The picture above illustrates the MWSS of the oscillation phases of a simulation run. After the first oscillation phase the MIN is set as the minimum MWSS, and the MAX is set as the maximum MWSS. Then, all of them are set as MWSS1,that is the MWSS of the first oscillation phase. After the second oscillation phase MAX is set to MWSS2, the MWSS of the second oscillation phase. After the third oscillation phase MIN is set to MWSS3, the MWSS of the third oscillation phase. The MWSS of the fourth oscillation phase, MWSS4, lies between MIN and MAX. Thus, the steady state is reached and the result measuring starts.

If it is not possible for the system to reach a steady state (meaning that either the minmum or the maximum is re-set after every oscillation phase), you will be asked, whether you want to run the simulation (measuring of results) anyway.

If the simulation system (represented by the Working Environmentmodel from the application model) is overloaded, a steady state can never be reached, as the performers' and resources' task stacks will continuously increase. This, of course, also results in increasing waiting and cycle times.

Length of the oscillation phases and result phase (steady-state view)

You determine both the length of the oscillation phases (see p. 379) and the length of the result measuring by the "number of simulations" (see chap. 4.4.1.1, p. 369) you enter prior to the start of the simulation. An oscillation phase (or a results phase) is finished when each process contained in the application model has been executed at least *n* times, where *n* is the number of cycles entered.

Initialisation phase (fixed time period view)

In the case of the workload analysis (see chap. 4.4, p. 367), the simulation again starts with an "empty system", i.e. the different performers of the working environment assigned have no activities to be executed on their "desks". For this reason, it does not make sense to start measuring the results as soon as the simulation starts.

Therefore the results are measured in the **workload analysis (fixed time period view)** after the so-called **"initialisation phase"**. The processes are simulated during this initialisation phase but no results are measured yet.

You determine how long the initialisation phase lasts by entering numbers (see chap. 4.4.1.1, p. 369) prior to the start of the simulation. The initialisation phase will last from the **"Simulation start"** entered to the **"Calculation begin"** date.

The initialisation phase replaces the oscillation phases (see p. 379) of the workload analysis (steady-state view). Since the length of the initialisation phase depends on what you enter, there is no guarantee that the system will have reached a steady state (see p. 379) at the time the initialisation phase finishes.

Tolerance waiting time

You can assign a **"Tolerance waiting time"** to a Business Process Model in the Process start. When the attribute **"Use time until abandon"** is set on the value "yes", each process instance will be abandoned, if the waiting time between its start (see p. 377) and the moment when you start to edit the first activity is longer than the assigned tolerance waiting time.

Example:

In the simulation of call centre processes, an incoming phone call will be broken off after a tolerance waiting time of a few seconds, if the call has not previously been answered (i.e. has started with the execution of the first activity in the process).

Hint: Business Process Models, whom a tolerance waiting time has been assigned to, shall not start with a parallelity.

Hint: The consideration of the tolerance waiting time is carried out the same way in the **steady-state** and the **fixed time period** .

Assign activities

The activities are assigned to the performers as in the capacity analysis (see p. 363).

Each activity can be controlled via the attribute "task stack", whether the activity is directly assigned to the performer ("personal task stack") or to a "central task stack".

Personal task stack

An AQL expression (see chap. 14., p. 790) is evaluated and renders a number of performers. This way, the performer, whose waiting queue is the smallest (in terms of the sum of the execution time of the activity in the waiting queue), is selected to perform this activity:

The activities in the personal task stack of a performer will be exclusively executed by this performer. They will remain in his task stack as long as he executes them, even if in the meantime another potential performer is unoccupied.

Central task stack

Activities in the central task stack are initially not assigned to a specific performer. As soon as a potential performer is free, he can take an activity from the central task stack and carry it out. The selection of the next activity by a performer (either from the personal or from the central task stack) is done according to the priority of the activities (see p. 384). In case of matching priorities, the activities in the personal task stack take precedence over the activities in the central task stack.

Hint: The activities are assigned to performers in this way both in the **steady-state view** and in the **fixed time period view**.

Continuous execution

Processes which imply direct contact to a customer are often executed without interruption, such as banking business processes like "Opening an account".

In the case of such business processes it has to be ensured that following activities are executed directly. This is modelled using the attribute "Continuous execution" in objects of the class "Activity" (possible values "yes" / "no"). If the value is "yes", the following activity is executed directly, in case the performer is the same, regardless of his task stack. If the value is "no", the performer may execute an activity that belongs to a different business process.

The attribute "Continuous execution" also applies to subprocesses.

ATTENTION: The "continuous execution" shall not be set on "yes", if an activity is assigned to the "central task stack" (see p. 381).

Example:

There is a sequence of two activities: First the activity "execute application", then the activity "authorisation".

The value of the attribute "continuous execution" is "yes" in the activity "authorisation".

The value of the attribute "performer" of the following activity "authorisation" is "'execute application' done'. This ensures that the activity "authorisation" is executed by the same performer who executed the activity "execute application".

During the simulation the activities are processed as follows:

In the course of the simulation, the activity "execute application" is performed, let us say, by "performer-1". This performer has a task stack of activities to be executed. After "performer-1" has executed the activity "execute application", he also receives the activity "authorisation". Since "continuous execution" is activated, the activity "authorisation" is executed immediately, independent of "performer-1"'s task stack.

Hint: The "continuous execution" works in the same way both in the **steady-state view** and in the **fixed time period view**.

Cooperative activities

Objects of the class "Activity" can be marked as **"cooperative"** using the attribute "Cooperative". The attribute "Cooperation mode" specifies whether a particular cooperative activity can be carried out asynchronously or not.

- **Asynchronous Cooperative Activities** are assigned to all qualified performers. (Depending on the value in attribute "Performer" and the value in attribute "Average number of participants"). The process is continues AFTER all performers have executed this activity. Each performer may execute this activity at different times - depending on their current task stack.
- **Synchronous Cooperative Activities** are executed by all performers simultaneously. The attributes "Min. quota of presence" and "Max. start period" are evaluated to determine how many performers need to be available in order to execute the activity and how long the process can wait for performers to become available.

Hint: Synchronous cooperative activities are executed at the earliest possible time. If necessary, other activities will be interrupted and continued after the cooperative activity has been finished.

ATTENTION: Cooperative activities shall not be assigned to resources (see p. 384), since it would abort the simulation.

The attribute **"Min. quota of presence"** indicates how many performers must be available in order to execute the activity. This attribute contains a number between 0 and 100 which is interpreted as a percentage.

Example:

Evaluation of performer assignment (see p. 381) (AQL (see chap. 14., p. 790) expression in the attribute "Performer") in a synchronous cooperative activity renders a set of four performers. The attribute "Min. quota of presence" has the value "**75**". This means the activity can be executed as soon as three performers are available (75% of four performers).

The attribute **"Max. start period"** defines the period of time, in which the cooperative activity should take place. If it is not possible to find a time for the synchronous cooperative activity within the "Max. start period" defined, an appropriate message will be issued. Click on the "Yes" button or press the "Enter" key to close the window and continue the simulation. The simulation is continued, the period of time in which the cooperative activity is to take place is extended by the value of the attribute "Max. start period". If it is still impossible to find a time within this maximum start time period, the statement will be issued again.

Selecting the "No" button will cancel the simulation. You can then check the availability of the performers participating in this activity, reduce the minimum number of performers with whom the activity can be processed or possibly increase the "Max. start period":.

Hint: Cooperative activities are treated in the same way in the **steady-state view** and in the **fixed time period view**.

Uninterruptible activities

Waiting time can arise in activities for various reasons. For instance, waiting time can appear when the relevant performer is already executing another activity and so the current activity has to wait until its execution. Waiting time can also arise when the activity is interrupted because the working hours of the performer are to the end. The execution of this activity will be continued as soon as the performer starts work again.

Using the **"interruptible execution"** attribute, you can choose that no waiting time shall arise for some specific activities. If this attribute is set on the "No" value, you will try to execute this activity if possible without waiting time.

This means:

1. If an uninterruptible activity is placed in the personal task stack (see p. 381) of a performer, he will interrupt the execution of every other activity to start executing this uninterruptible activity.
2. If an uninterruptible activity is placed in the central task stack (see p. 381), it will directly be assigned to a potential performer. Then the process is the same as for a personal waiting queue.
3. If during the execution of an uninterruptible activity the performer finishes his working hours, he will still continue the execution. As soon as the execution of the activity is finished, you will verify whether other inunterruptable activities are still waiting in the personal task stack of the performer. If this is the case, they will also be executed. The performer, who has already finished his working hours, will no longer check the central waiting queue.

Hint: uninterruptible activities must be executed at the same time and by the same performer, then the priority (see p. 384) decides, which activity will be executed first. For the other activity, a waiting time will arise despite its definition as being uninterruptible.

Determine Assigned Resources

The resources are assigned in the same way as in the capacity analysis (see p. 364).

If evaluating the AQL expression (see chap. 14., p. 790) renders more than one resource, the following procedure is applied to select a certain resource:

The activity is assigned to the resource with the shortest waiting queue (task stack) - size of the queue is determined by the sum of the execution times of the activities in the queue.

ATTENTION: Resources shall not be assigned to cooperative activities (see p. 382), since it would abort the simulation.

Hint: The resources are determined in the same way for the **steady-state view** and for the **fixed time period view**.

Processing Activities: Priorities and maximum Resource Waiting Times

If a performer is overloaded while the workload analysis is executed, a task stack of activities to be executed builds up beside his desk. Arriving activities are placed at the end of the task stack.

To ensure that the task stack is processed in a realistic manner, the attributes "Priority", "Continuous execution", "Interrupted execution" "Co-operative", "Max. resource waiting time" in the objects of the class "Activity" are evaluated to determine the next activity.

Then the performer's task stack is sorted as follows:

1. Uninterruptible activities (see p. 383):

   Uninterruptible activities will always be executed with the utmost priority and can even interrupt the execution of other activities.

2. Synchronous cooperative activities (see p. 382):

   Synchronous cooperative activities usually represent among other things conferences, meetings. They have the second utmost priority and can also interrupt the execution of other activities.

3. Activities with continuous execution (see p. 382):

   Activities with continuous execution are at the third place and they can **not** interrupt the execution of activities.

4. Interrupted activities, which will now be continued:

   Interrupted activities have priority over "normal", not interrupted activities. They can **not** interrupt the execution of activities.

5. "Normal" activities:

   All activities which have none of the above-mentioned characteristics.

In this category, the activities are sorted according to their priority (attribute "priority"). The higher the value the attribute "priority" is, the further up the activity is ranked within the task stack. Activities with the same priority will be sorted according to the already incurred waiting time (activities with the longest waiting time first).

If the first activity in the task stack is assigned to one or more resources (see p. 364), then the task stack of the resource will be evaluated (if necessary a resource will be previously determined by the evaluation of the attribute "selection"): If the sum of the execution times for the activities in the task

stack of the resource is bigger than the value given in the attribute "max. resource waiting time", then the next activity in the task stack of the performer will be given priority.

Each time, the activity on top of the task stack (at the first rank) will be executed.

Hint: Prioritising activities will be treated in the same way for **steady-state view** and **fixed time period view** .

Determining the performers' workloads

The relationship between the "amount of work" a performer is assigned and his available working time determines his workload. The workload is a period-related (see chap. 4.1.7, p. 342) result which is described by a floating point number.

A performer's workload is calculated by the workload analysis (see chap. 4.4, p. 367). It is the sum of the execution times of the activities the performer has executed related to the selected period of time.

A performer's working time is the time during which he is "present", i.e. available for executing activities. The working time is described by the (Performer) Calendar (see p. 378).

Hint: The workload must not be greater than 1.

4.4.7 Application scenario (example)

Hint: In order to run an accurate workload analysis, more information is needed than to run either a Path Analysis (see chap. 4.2, p. 343) or a Capacity Analysis (see chap. 4.3, p. 354). This includes additional information such as the occurrence frequency of processes (see p. 377) and the real availability of a performer (see p. 378)). For this reason, the workload analysis is especially useful for comparative propositions (such as "Is one model better or worse than another one?"), as the acquisition effort is smaller here than when running a workload analysis for quantitative purposes.

The workload analysis enables you to detect bottlenecks in your Business Process Models.

In the Working Environment model the activities to be executed are visualised during the course of the simulation as "task stacks" (see chap. 4.4.3, p. 372) belonging to each performer. If the performers have very large task stacks, this indicates that they are overloaded with work.

With the help of person related evaluation you can identify which activities are executed by which performers. By examining the execution times of the activities carried out by "overloaded" performers, you can identify for example where improvements are possible.

The activities which are executed mainly by the "overloaded" performer may be a good starting point for improvement. Bottlenecks may be defused by the following changes:

- **Changing the Performer assignment (see chap. 2.2.12.14, p. 222) of an activity**

  If the number of performers who can execute an activity is increased, then an "overloaded" performer will carry out the activity less frequently. This should reduce his workload.

- **Add Additional Performers**

  If you add a new performer to the Working Environment which has exactly the same characteristics as the overloaded performer (same organisational unit, same role and so on), the workload of the overloaded performer should also be reduced.

4.5 Offline animation

Offline animation enables you to rerun a protocol (logfile) created during a simulation. As in a workload analysis simulation, the performers' task stacks are animated in the Working Environment while running offline animation.

Hint: It is not possible to list the contents of a task stack during the offline animation

Hint: By default, results are **not** calculated during offline animation. However, should you have defined agents (see chap. 4.8, p. 391), these will calculate results during the offline animation.

The protocol file contains a history of all the relevant events of the simulation run. With the help of offline animation you can replay the protocols from the Capacity Analysis (see chap. 4.3, p. 354), Workload Analysis (steady-state view) (see chap. 4.4, p. 367) and Workload Analysis (fixed time period view) (see chap. 4.4, p. 367).

Offline animation has the following advantages compared to a "normal" simulation (see chap. 4., p. 332):

- Replaying the simulation can be compared to a video. You can examine certain parts of the period simulated, fast forward or rewind the simulation protocol and replay the animation in single steps (both forward and backward).
- Offline animation is usually faster than the simulation (depending on your models' structure).
- It is possible to investigate specific aspects of your models more closely (once you have the results from a "normal" simulation run). You may still define appropriate agents (see chap. 4.8, p. 391) and repeat the previous simulation run exactly using the offline animation. The agents' results are thus based on the same simulation events as the simulation results. There is therefore no problem in comparing them.

4.5.1 Input

The inputs for offline animation (see chap. 4.5, p. 386) are the **short version** of a simulation protocol (apf file) and the application model during the simulation of which the protocol has been generated.

Hint: The long version of the simulation protocol **cannot** be evaluated by the offline animation.

Hint: The models contained in the application model should **not** have been **substantially changed**, since the simulation protocol was generated.

Additional objects may be added and attributes may be changed. However, objects such as the performers must not have been renamed or deleted. If the protocol refers to an object that no longer exists in the simulated models, the animation will be cancelled with an appropriate error message.

4.5.2 Carry out an offline animation

To start offline animation, proceed according to the following steps:

1. Select the option "Offline Animation" from the "Algorithm" menu. Alternatively, click on the smart-icon in the quick-access panel.
2. Enter the path and name of the protocol file (APF-file) to be opened in the screen "Open simulation protocol" which is now displayed and click on the "OK" button.

3. If your models contain active agents (see chap. 4.8, p. 391), you will now be asked, whether you wish to employ these.

   Hint: If agents (see chap. 4.8, p. 391) are active during the offline animation, it is not possible to rewind the protocol or to step backwards. However the function "Back to the start" (see chap. 4.5.3, p. 387) will rewind to the beginning of the simulation.

After any models which are not currently open have been loaded, the window "Offline Animation" (see fig. 338) is displayed.

4.5.3 Processing

Processing is controlled via the "Offline Animation" (see fig. 338) window.

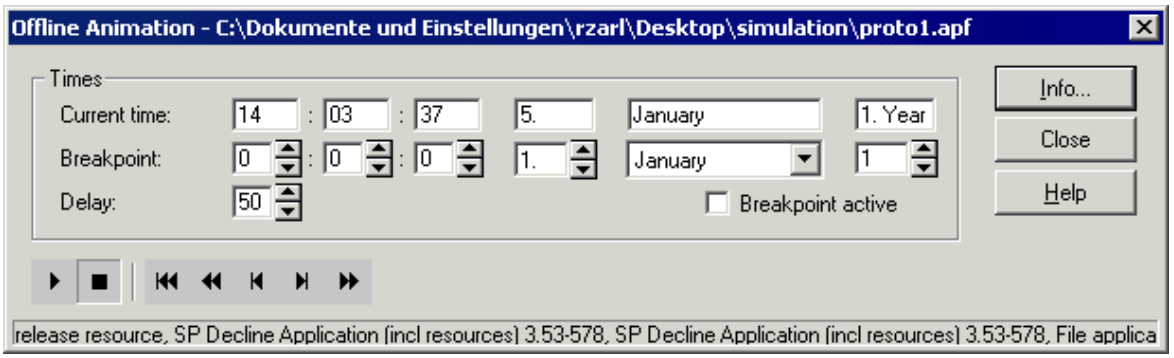


Figure 338: Offline animation - control

Clicking on the button **"Info"** will provide you with information on which models will be animated. The **"Current Time"** indicates the time at which a step was executed. In addition, the following settings can be made:

Breakpoint: Here you can enter a time at which the current operation (play, fast forward or rewind) should be interrupted if the option "Breakpoint active" is also activated.

Breakpoint active: Default Setting: deactivated

If the option "Breakpoint active" is activated, the offline animation is interrupted at the breakpoint specified.

Delay: This option allows you to slow down the animation. The higher the value you enter, the higher the delay, i.e. the slower the offline animation.

The following functions are available to support you in replaying the simulation protocol:

Play

By clicking on the Play button, offline animation will be started or continued as appropriate. To view the animation of events, you must activate the model window of the working environment mode simulated. The animation is interrupted when the end of the protocol is reached, an activated interruption point is reached or if you click on another button (such as "Stop").

Stop

Clicking on the "Stop" button interrupts the current operation (such as Fast forward or Rewind).

Back to the start

By clicking on this button the current operation is interrupted and the system jumps back to the start of the simulation protocol.

Fast rewind

If you click on this button, the protocol is rewound. Activated interruption points are taken into account.

Note: The delay entered is not taken into account when the system rewinds.

Single Step Backward

By clicking on this button the animation steps back to the previous event.

Single Step Forward

By clicking this button, the next event is animated.

Fast Forward

Click on this button to fast forward the protocol. Activated interruption points are taken into account.

Note: The delay entered is not taken into account when the system fast forwards.

Should agents (see chap. 4.8, p. 391) be active during offline animation, you can obtain the agents' results (see chap. 4.8.6.2, p. 424), as soon as the offline animation is finished by clicking on the button "Close". It is important that at least part of the protocol is played through in order that the agents should have an opportunity to calculate results. If your agents have not calculated results after the offline animation, this indicates that too short a part of the protocol was played through.

ATTENTION: Please notice that all the agents' results are also reset (i.e. deleted), if you jump back to the beginning of the simulation protocol (button "Back to the start"). You must then re-run the offline animation to obtain the agents' results (see chap. 4.8.6.2, p. 424).

4.6 Simulation cache

In order to simulate (see chap. 4., p. 332) Business Process Model and Working Environment models in ADOxx, an internal representation of the models must exist in the main buffer/cache. For this reason, all models to be simulated are transferred into this internal representation prior to the simulation run and stored in the **simulation cache**.

If the models are very complex, it may take some time to transfer them. If you wish to run a sequence of simulations with the same application model, the models are only transferred prior to the first simulation run. Afterwards, the representation which exists in cache is used again.

This assumes that the application model to be simulated has not been substantially changed in the meantime.

A change may have the following effects, depending on its type (types of changes (see chap. 4.6.1, p. 389)):

- A completely new transfer of the application model into cache
- A partial transfer of single main processes within the application model
- A partial transfer of single subprocesses within the application model
- The re-use of an application model already transferred

If you wish, you can also delete (see chap. 4.6.2, p. 389) the contents of the simulation cache manually.

4.6.1 Changes and their effects on the simulation cache

A **completely new transfer** of the whole application model is necessary under the following circumstances:

- If the simulation algorithm is changed (e.g. first capacity analysis (see chap. 4.3, p. 354) and then workload analysis (see chap. 4.4, p. 367)).
- If at least one of the parameters "Working days per year" or "Hours per working day" has been changed.
- If the application model is changed, because the **Working Environment model** is replaced by a different one.
- If the Working Environment model is changed.
- If the starting point of a workload analysis simulation is changed.
- If a new input parameter combination is selected.

A completely **new transfer of a main process** (a Business Process Model contained in the application model, including any directly or indirectly called subprocesses) is necessary under the following circumstances:

- If the main process is not contained in the simulation cache.
- If the number of subprocesses, which a main process calls directly or indirectly, changes.
- If new global variables, which have not yet been transferred to cache are defined in the main process or in one of its subprocesses.

If changes other than those outlined above are carried out, ADOxx will transfer **only the changed process model**, as in the following examples:

Example 1: An additional activity is added to a process model.

Example 2: The order of two activities in a process model is changed.

Example 3: A subprocess call is removed from a process model, but the same subprocess is still called from another place.

Example 4: Performer assignment is changed for a subset of activities in a process model.

If none of the models in the application model were changed and none of the above parameters of the simulation were changed, then the simulation can be run using the models which already exist in cache **without a new transfer**.

4.6.2 Manual deletion of the simulation cache contents

By manually deleting the simulation cache, the internal representation of the simulation models is deleted and the cache is emptied.

This is sensible, if you want to enforce a new transfer of the models to be simulated, if no further simulation runs are to be carried on or simply to free buffer space.

The steps below should be followed in order to delete the contents of the simulation cache

1. Switch into the Simulation Component (see chap. 4., p. 332).
2. Select the option "Free simulation cache" form the "Edit menu". Alternatively, click on the smart-icon in the quick-access panel.

An appropriate information message will be displayed when you have freed the simulation cache.

4.7 Delete simulation results

The functionality "**Delete Simulation Results**" enables you to delete simulation results which had previously been transferred into your models (see chap. 4.1.2, p. 334) (E.g. contents of chapters "Simulation results" or "Aggregated results" in the Process start and Activity objects). When you delete simulation results, any numerical values will be reset to 0 and any alphanumeric (text) values will be blanked.

ATTENTION: **All models**, from which you would like to delete simulation results must be loaded **read-write**, otherwise an appropriate information window will be displayed.

In order to delete the simulation results, select "Reset simulation results" from the "Edit" menu in the Simulation Component. Alternatively, you can click on the smart-icon in the quick-access panel.

The window "Delete simulation results - model selection" (see fig. 339) is now displayed. You can now select the models from which you would like to delete simulation results.

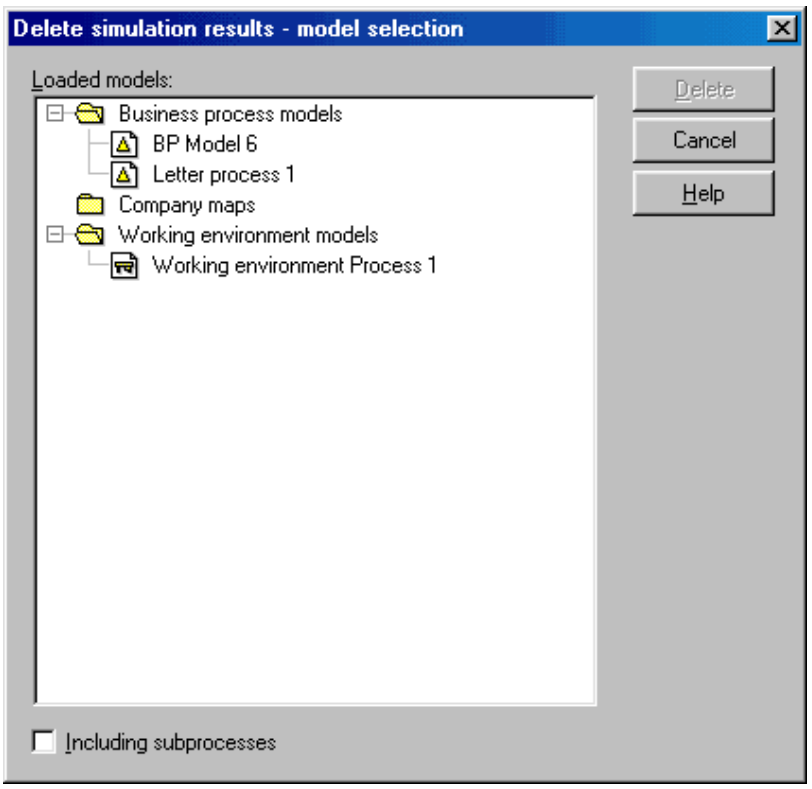


Figure 339: Delete simulation results - model selection

Hint: In the Business Process Models selected the values of the simulation result attributes are deleted in all objects of the class "Process start" and of the class "Activity". If you select Working Environment models, the simulation result attributes are reset in all objects of the class "Performer".

When deleting simulation results from Business Process Models, you can specify whether or not simulation results should also be deleted from any called subprocesses by checking **"Including subprocesses"**. In this case the models selected are interpreted as main processes.

Once you have selected the models, click on the button **"Delete"**, to start the deletion. The simulation results in the models selected will be deleted.

Hint: Once the simulation results have been deleted, the models concerned are automatically saved in the database.

4.8 Agents

The task of agents is to support you if you wish to calculate additional non-standard simulation results. During the simulation the agents collect information about the behaviour of the Business Process Models and thus produce additional simulation results.

The difference between the standard simulation results and the **agents' results** is that you can **freely define** the results the agents are to determine.

While the simulation calculates **"general process measures"**, such as cycle times and execution times, you can employ agents to calculate **"process-specific measures"** such as the cycle time up to a specific activity, the execution times of all manual activities or other company-specific measures.

Using agents you can determine evaluation results for **process parts**, **subprocesses** and different organisational structures in your Working Environment models. If you instruct an agent to investigate only certain selected objects within a model, it will ignore all other objects of this model and not take them into account when the results are calculated.

Agents can also determine evaluation results **for selected periods of time** (such as "expected costs in the first quarter of a year").

The following settings must be made when configuring an agent (see chap. 4.8.3, p. 396):

- Result which the agent is to determine, (see p. 399),
- Objects (such as activities, performers...), which are to be taken into account (see chap. 4.8.3.1, p. 397) when the results are calculated,
- In **workload analysis** (see chap. 4.4, p. 367):

  the period of time (see chap. 4.8.3.1, p. 397), during which the agent is to calculate the results.

An agent is always created (see chap. 4.8.1, p. 391) in a certain model. The model may be of any type that can be simulated. The agent is stored together with the model in the database and will also be automatically exported or imported during an ADL-Export (see chap. 6.2, p. 468) or. ADL-Import (see chap. 6.1, p. 445).

4.8.1 Creating and customising agents

To create new agents or change existing agents, proceed according to the following steps in the simulation component:

1. Open the models in which you wish to edit or create agents.

   Note: If you open models which are write-protected, you may edit the agents, but you will not be able to save the changes.

2. Select the option "Agents" in the menu "Edit". Alternatively, you can click on the following smart-icon .

   It is also possible to select the "Agents" button from the start window of one of the simulation algorithms.

The window "Agents Overview" (see fig. 340) provides you with an overview of the existing agents of the model.

Hint: It is possible to define agents while configuring ADOxx (see chap. 4.8.2.1, p. 394). These agents can be defined as only valid for certain simulation algorithms. If you open the window "Agents Overview" (see chap. 4.8.2, p. 392) from the start window of a simulation, only agents which can be evaluated during this type of simulation will be displayed.

4.8.2 Agents overview

The window "Agents overview" (see fig. 340) contains a section called **"Agent List"** in which any agents which already exist on the selected models are displayed.

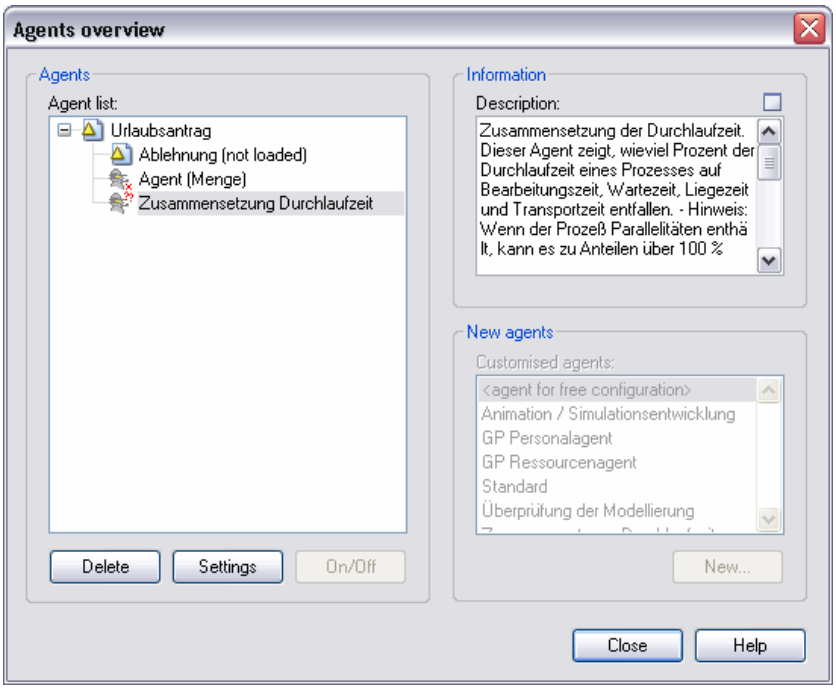


Figure 340: Agents overview

Hint: Agents are part of the models in which they were created. During an ADL export or import they are exported or imported together with the other model contents.

The subprocesses of the models selected are shown in the model hierarchy if they are also open.

Hint: If a Subprocess object references a non-existing subprocess, this is indicated by :

The **state of each agent** is indicated by the following icons:

Active:

Active agents evaluate the simulation and produce evaluation results.

Deactivated:

Inactive agents do not collect information during the simulation and therefore produce no evaluation results until you reactivate them.

Error:

An agent with errors is not activated during the simulation. It does not produce evaluation results. A faulty agent can be created when for example the names of the modelling objects which the agent observes are changed, but the agent is not adjusted (see chap. 4.8.2.2, p. 395) to reflect this.

Hint: If you change the names of modelling objects, the window "Agents - Change observed objects" (see fig. 341)) will be displayed when you call the Agent function. The references can be updated in this window.

If an agent is selected, the field "Information" will display the following information:

- An **information text**, written by the user who originally created the agent.
- The **results** which this agent calculates.

Creating a new agent

To create a new agent, proceed according to the following steps:

1. Select a model from the **"Agent List"**, in which you would like to create the new agent.

   Note: If you create agents in write-protected models, you cannot save these. All settings made for this agent will be deleted once you close the model.

2. Select an agent type from the list of **Customised agents**

   The agent type "**<agent for free configuration>**" is always available for selection. This is an agent which has no tasks assigned to it and which you can customise as desired.

   Note: Agent types are predefined "templates" for agents. If you employ agents frequently for certain consistent tasks, then appropriate agent templates can be defined during the configuration of ADOxx.

   Please contact your ADOxx Administrator for more information.

3. Click on the **"New"** button. The window "Agent configuration - <*Name of Agent*>" (see fig. 342), in which you can assign tasks to the new agent, will now be displayed.

   Hint: If there are several agent classes defined in your application library, you must first select the agent class you require in the window "Agent - Class Selection".

Changing an existing agent

To change an existing agent, first select it in the field **"Agent list"**. One of the following buttons can then be selected (clicked on):

"Settings"	opens the window "Agent configuration - <*Name of Agent*>" (see fig. 342). You may then change some of the existing settings or add new tasks for the agent to carry out.
"On/Off"	activate or deactivate the selected agent(s).
"Delete"	delete the agent(s) selected.

Agents in write-protected models

New agents may be created and existing agents altered in write-protected models. The models can then be simulated and the agents will calculate results as specified.

ATTENTION: It is impossible to save write-protected models (and therefore changes to their agents) in the ADOxx database. All agent changes you have made on a write-protected model will be lost when the model is closed.

4.8.2.1 Define agent types

Agent types can be defined during ADOxx customisation. Agent types are "templates" which are used when you wish to create new agents. By using agent types, the characteristics and tasks of an agent are automatically defined - to whatever extent of detail is required - when it is created. You can of course change and adjust agents created in this way later on.

The Administration Manual (Library Management chapter) contains a description on how to create agent templates. Contact your ADOxx administrator if you need further agent types defined.

The ADOxx-Default-Library already contains a number of predefined agents (see p. 394).

Pre-defined agent types in the ADOxx standard application library

New agents, based on predefined agent templates, can easily be created in the "Agents overview" (see fig. 340) screen (see chap. 4.8.2, p. 392)

The ADOxx-Default-Library contains the following predefined agent types:

Animation / Simulation progress: An agent of this types shows the development of its process's average cycle time during the simulation in the form of a function graph. Towards the end of the simulation, the cycle time should not contain large variations. If it does, this means that there were too few simulation runs and therefore the simulation results are not meaningful.

By default, agents of this type take all activities and all subprocesses into account when calculating the cycle time. However, if required, you can restrict the agent's reference objects in such a way that only the cycle time of certain process parts or subprocesses are animated.

WE Agent: Personnel: A 'WE Agent: Personnel' calculates the personnel capacity and the personnel costs of all performers of the Working Environment model, in which it is created. The personnel capacity and the personnel costs refer here to all Business Process Models simulated and can be aggregated for roles and organisational units.

This agent can only be employed during the capacity analysis. You may restrict the reference objects of the agent to certain performers, organisational units or roles.

BP Agent: Personnel: 'BP Agent: Personnel' agents are created in business process models and calculate personnel capacity, personnel costs and changes in responsibility for all activities and subprocesses in the model. The results for subprocesses are calculated in aggregated form (not separately for each activity).

A 'BP Agent: Personnel' can only be employed during the capacity analysis. You may restrict the reference objects of the agents to certain process parts or subprocesses.

BP Resource Agent: 'BP Resource Agents' are created in Business Process Models and calculate the resource capacity, resource costs and the number of resource changes for all activities and subprocesses in their model. The results for subprocesses are calculated in aggregated form (not separately for each activity).

'BP Resource Agents' can only be employed during the capacity analysis. You may restrict these agents' reference objects to certain process parts or subprocesses.

Standard: Agents of this type are created in Business Process Models and calculate standard results. The results calculated depend on the simulation algorithm run. The following results can be calculated by a "standard" agent:

- Relative frequency of occurrence of all activities and subprocesses
- Average Execution time
- Average Waiting time (*only Path and Capacity analysis*)
- Average Resting time
- Average Transport time

- Average capacity (*only Capacity analysis*)
- Average workload (*only Workload analysis*)
- Average Personnel costs (*only Capacity analysis*)
- Average Personnel costs (limited (*only Workload analysis*)
- Average Resource Costs (*only Capacity analysis*)
- Average Resource Costs (limited (*only Workload analysis*)
- Average Costs
- Average Cycle Time

By default, these results are calculated for all activities in the agent's model and all its subprocesses (arbitrary depth). It is possible to restrict the extent of detailed calculation (e.g. calculate only the results for the main process). It is also possible to restrict the agent's reference objects to parts of the process or subprocesses.

Modelling check Agents of this type check all their model's activities for modelling errors. By default, subprocesses are not checked. If you wish to check them, too, you must either extend the reference objects by the subprocess(es) or create an agent in each subprocess you wish to check.

The agent checks, whether the attributes "Execution time", "Waiting time", "Resting time", "Transport time" and "Costs" are set in all activities. This agent also recognises activities in "dead paths", i.e. in paths which will never be executed during the simulation of the process.

In order to use agents of this type you must execute a path analysis. This agent can only be created in Business Process Models.

Cycletime ratios Agents of this type calculate the cycle time and how it is made up (in percent) of "Execution time", "Waiting time", "Resting time" and "Transport time".

The cycle time is calculated separately for all activities in the agent's model and in all its subprocesses (regardless of the grade of depth).

It is only possible to use agents of this type in business process models and only in the path and the capacity analysis.

ATTENTION: Please note that the total percentage may exceed 100%, where the process evaluated contains parallelism.

4.8.2.2 Adjusting reference objects

By adjusting the reference objects you can correct any reference objects of your agents which have become invalid.

Hint: The adjustment only affects your agents' configuration, not the objects modelled within your models.

The window "Agents - Change observed objects" (see fig. 341) will - after an appropriate query - be displayed if the names of objects observed by an agent have been changed and the agent function has been called.

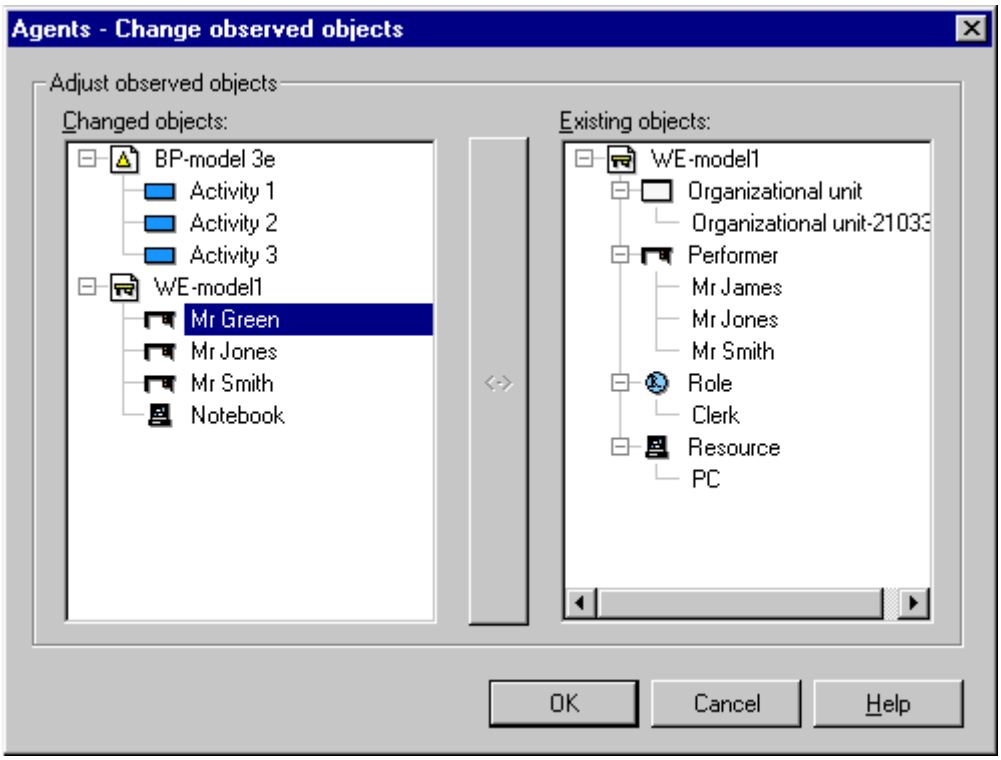


Figure 341: Agents - adjusting reference objects

The list **"Changed objects"** contains all objects, to which agents had been assigned for evaluation purposes, but which no longer exist under their original name (as they were renamed or deleted). These objects are shown within the model in which they had originally been placed.

As soon as one of these objects has been selected, the **"Existing objects"** list shows all objects of the same model which are potential reference objects for the agents. Select the object, which corresponds to the object in the field "Changed objects" and click on the button **"<->"**. By doing this, the selected existing object is assigned to all agents in place of the changed object.

After adjusting the reference objects, click on the "OK" button. If all the "Changed objects" have not been adjusted, a security message will be displayed. This will allow you to either delete any remaining "Changed objects" or return to the "Agents - Change observed objects" screen.

If you click on the button "Cancel", you will be returned to the "Agents - Change observed objects" screen where you may re-assign the remaining changed object(s) objects are not adjusted. Alternatively clicking OK will cause any remaining changed object(s) to be deleted from the list of objects currently observed by the agent. Please note that invalid agents are indicated by the symbol in the window "Agent Overview" (see fig. 340).

4.8.3 Configuring agents

An agent's behaviour is designed in the window "Agent Configuration - <*Agent Name*>" (see fig. 342)). The window contains a tab for **"Basic configuration" (see chap. 4.8.3.1, p. 397)** and can contain tabs for the Evaluation Results (see chap. 4.8.4, p. 400) the agent is to calculate.

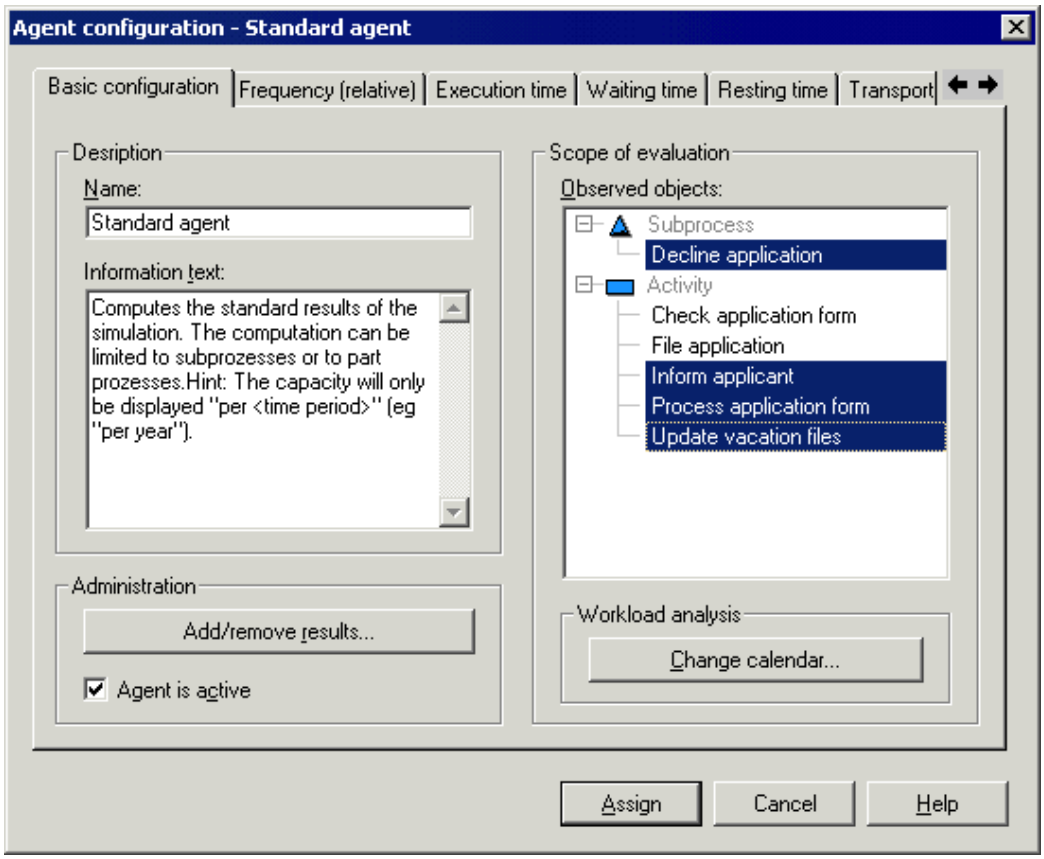


Figure 342: Configuring agents (example)

4.8.3.1 Basic configuration

It is possible to specify which modelling objects the agent should take into account when calculating results and for which periods of time the agent is to calculate the results in the "Basic configuration" (see fig. 342) tab.

The following entries can be made on this screen:

- **Name**

  A name can be entered for the agent.

- **Description**

  The agent can be described as you require here. This description is to help both you and other ADOxx users to recognise this agent's tasks quickly. We recommend that you include a reference to the objects observed by this agent within the description.

- **Add/remove Results**

  Click on this button, to add or remove result functions (see p. 399).

  Note: For each result the agent is to calculate an additional tab is added to the window "Agent configuration". Select a particular tab to instruct the agent regarding calculation details for that result.

- **Agent is active**

  This option enables you to activate or de-activate the agents.

- **Observed objects**

The reference objects (objects to be observed) for the agent can be selected here. The agent will only take objects specified here into account when calculating the results.

When a subprocess call is selected, the agent will monitor the model that is referenced by this subprocess call provided that the model was called by the simulation via the subprocess call selected.

If you select grouping objects (e.g. roles or organisational units) in working environment models, the agent will monitor all objects which are directly related to the grouping object selected by **any relation** (such as all the performers assigned to an organisational unit).

Clicking with the right mouse button in the field "Observed objects" will cause a pop-up menu to be displayed. Select the option **"AQL Specification" (see p. 399)**. This will allow you to select the objects to be observed via the ADOxx Query Language AQL (see chap. 14., p. 790).

You may assign reference objects of the following modelling classes to an agent:

- Activity (in models of the BP library)
- Subprocess (in models of the BP library)
- Performer (in models of the WE library)
- Grouping (in models of the WE library)
- Resource (in models of all types)

When calculating the result, the agent observes only the modelling objects directly assigned to it and ignores all other (invalid) modelling objects. In this way, you can assign **all** modelling objects in a model to an agent for observation, even if invalid modelling objects should be among them.

- **Change calendar**

  This setting is only evaluated in the **Workload Analysis** (see chap. 4.4, p. 367). Click on this button to open (see p. 398) the agent's evaluation calendar.

Changing the evaluation period

In the window "Agent - evaluation calendar" (see fig. 343) you can instruct the agent to monitor the simulation only during certain periods of time. By doing this, you can obtain results for restricted periods of time (such as the costs in the first quarter of a year).

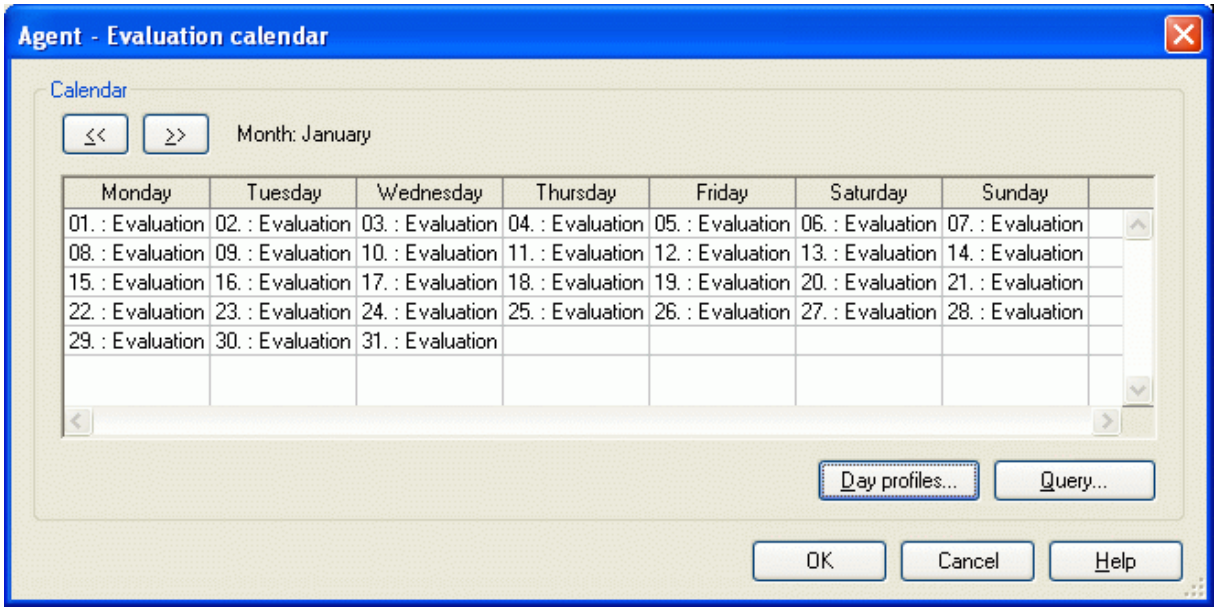


Figure 343: Agents - evaluation period

Hint: The "agent for free configuration" evaluates the simulation over the total simulation period by default (i.e. Monday to Sunday, from 0 until 24 hours each day). You need only change the evaluation calendar, if you want the agent to evaluate the simulation at other times.

AQL specification

The window "AQL Specification" (see fig. 344) allows you to select the agent's reference objects via the ADONIS Query Language AQL (see chap. 14., p. 790).

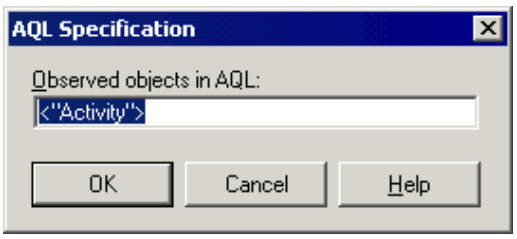


Figure 344: Direct input of referenced objects in AQL

Enter an AQL expression to select the required reference objects into the field **"Observed objects in AQL"**.

Examples:

- AQL expression to assign all activities of the agent's model to the agent:
  `<"Activity">`
- AQL expression to assign all subprocesses of the agent's model to the agent:`<"Subprocess">`
- AQL expression to assign all activities **and**(!) all subprocesses of the agent's model to the agent:
  `<"Activity"> OR <"Subprocess">`
- AQL expression to assign all "manual" activities to the agent:
  `<"Activity">[?"Classification"="manual"]`

Selection of result functions

The results to be calculated by the agent can be defined in the window "<*Agent Name*> - Select results to be generated" (see fig. 345).

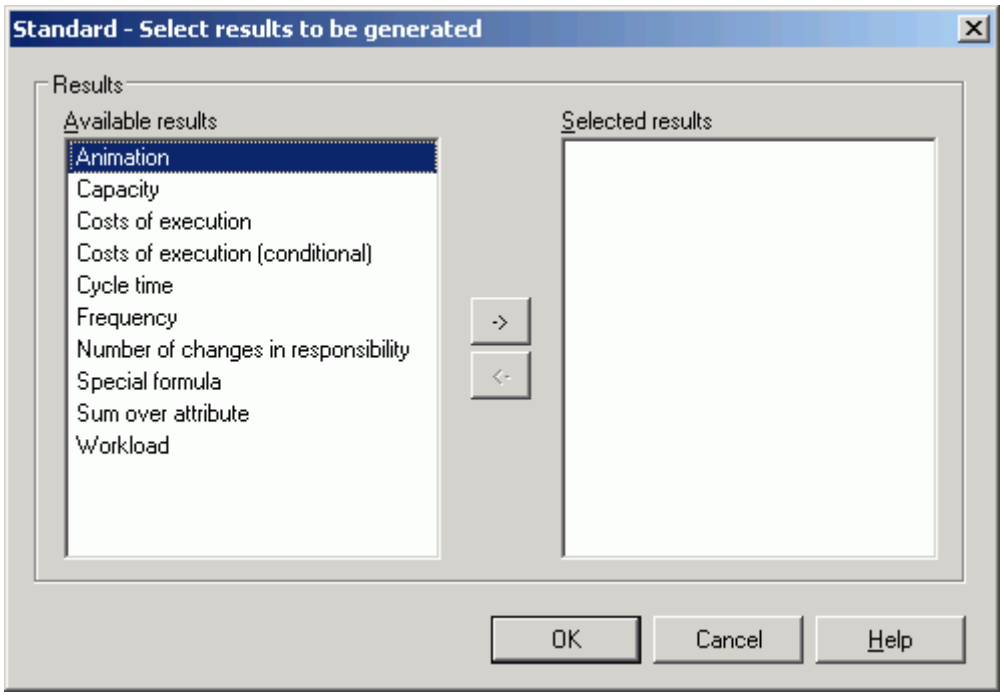


Figure 345: Agent - selection of result functions

The list **"Available results"** contains results functions which can be assigned to the agent for calculation. The list **"Selected results"** contains those results which the agent is currently instructed to calculate.

> **Hint:** The list "Selected results" contains the particular result functions with the names currently assigned to them. As names can be freely assigned according to your wishes (see chap. 4.8.5.1, p. 410), this list may contain entries which differ from those in the list "Available results".

Select one of the result functions available and click on the button **"->"** to assign it to the agent.

Select one of the result functions selected and click on the button **"<-"** to remove the result function selected from the agent.

> **Hint:** Some result functions can be assigned to an agent more than once.

4.8.4 Result functions for agents

You can instruct the agent to calculate the following results:

- **Animation** (see chap. 4.8.4.1, p. 401)
- **Number of changes in responsibility** (see chap. 4.8.4.2, p. 401)
- **Frequency** (see chap. 4.8.4.3, p. 402)
- **Costs of execution** (see chap. 4.8.4.4, p. 403)
- **Costs of execution (conditional)** (see chap. 4.8.4.5, p. 404)
- **Workload** (see chap. 4.8.4.6, p. 405)
- **Capacity** (see chap. 4.8.4.7, p. 405)
- **Cycle time** (see chap. 4.8.4.8, p. 406)

- **Special formula** (see chap. 4.8.4.9, p. 407)
- **Sum over attribute** (see chap. 4.8.4.10, p. 409)

Theoretically, each agent can be instructed to calculate as many results as you wish. In some cases, it can be sensible to assign the same result function a number of times to one agent.

Hint: Calculating several results takes up more time and may prolong the simulation considerably.

4.8.4.1 Result function "animation"

The result function **"Animation"** does not produce results itself but enables you to graphically represent the development of *other results* during the simulation.

The results are represented as function graphs (see chap. 4.8.6.1, p. 423) which each show the current result values during the course of the simulation.

With the help of the animation you can see whether the result values still vary strongly towards the end of the simulation or whether they have found their level in an average value.

Hint: This result function can be assigned to agents in all model types that can be simulated.

Define the results the agent is to animate in the "Animation" (see chap. 4.8.5.2, p. 412) tab in the "Agent configuration" (see fig. 342) window.

ATTENTION: As the "Animation" result function represents *other* results graphically, it is not sensible to instruct an agent to execute this function alone.

Hint: For each animation function you instruct an agent to execute, a window is opened during the simulation which shows the animated results. If you wish to represent different results in *separate* windows, you must **assign** the "Animation" result function **several times** to the agent.

4.8.4.2 Result function "Number of changes in responsibility"

The result function **"Number of changes in responsibility"** is employed to determine how frequently the executing performer or the resource employed is changed during the execution of a process.

Whenever there is a performer or a resource included in the execution of a process who was not previously directly involved in the execution of the process, this is counted as a change in responsibility.

Hint: When the first performer or the first resource starts executing the process, this does not count as a change in responsibility.

During the **parallel** execution of several activities, changes of responsibility occur due to an increase in participants involved who were prior to this not directly involved in the process's execution.

It also counts as a change of responsibility when activities are executed **cooperatively** by several performers and through this performers who were prior to this not involved in the process's execution take up work.

Hint: This result function can only be assigned to agents in models of the BP library. Reference objects may be activities and Subprocesses.

Hint: The result "Number of Changes of Responsibility" can be calculated in the simulation algorithms "Capacity analysis" (see chap. 4.3, p. 354) and "Workload analysis" (see chap. 4.4, p. 367).

Within the "Number of changes in responsibility" (see chap. 4.8.5.3, p. 413) tab, within the window "Agent configuration" (see fig. 342) you must define whether the changes of responsibility are to be calculated for performers, resources or for other classes of your application library.

Should your agent monitor modelling objects which are not directly related, a change in responsibility is only found, if the next **reference object** is executed by a performer or resource, different from the one which executed the last **reference object**.

Example:

Let us assume the following model. "Activity-1" and "Activity-3" are executed by "Performer-1", "Activity-2" is executed by "Performer-2". An "Agent X" has the reference objects "Activity-1" and "Activity-3" and calculates the "Number of changes in responsibility" (see fig. 346).

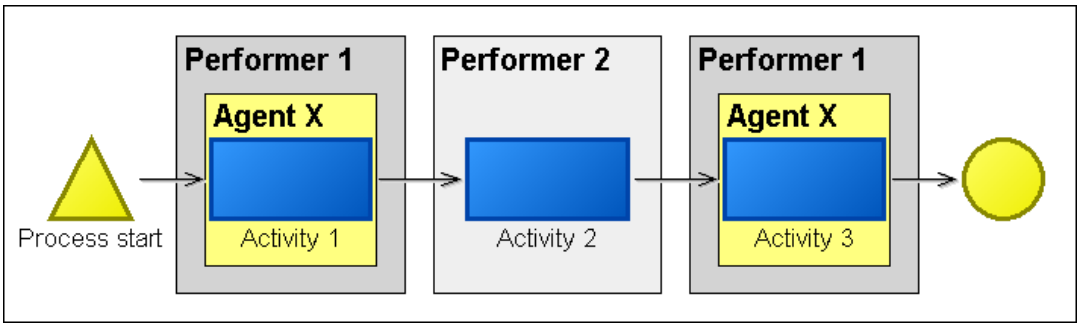


Figure 346: Example of changes in responsibility

"Agent X" renders '0' as result, since from its point of view no change in responsibility took place, as both of the activities observed were executed by the same performer.

However, if the same agent was to monitor all three activities, the resulting "number of changes in responsibility" will be 2.

If a change of responsibility takes place when **a subprocess is called**, this change is charged to the subprocess. However, if an agent *only* evaluates the subprocess, this change of responsibility will not count at all, since the agent did not take into account what happened *before* the subprocess started.

If an agent is to calculate the changes of responsibility with regard to several classes (e.g. for performers **and** for resources), you must **assign** this result function to the agent **several times**.

4.8.4.3 Result function "Frequency"

The result function "Frequency" causes an agent to calculate the frequency of occurrence of processes and modelling objects. You may instruct the agent to calculate both the relative frequency (in %) and the absolute frequency (i.e. how often the objects monitored were executed in total during the simulation).

Example:

The process-X is simulated with 1000 runs. In the course of the simulation, Activity-A of this process is executed 500 times. The **relative frequency** of Activity-A is thus 0.5 (i.e. 50 %). The **absolute frequency** of Activity-A is 500.

In addition, you can instruct an agent in the result function to calculate all those objects which are **never** executed during the simulation. In this way, you can discover "dead paths" and modelling errors in your processes. Of course, the agent will only search for objects which are never executed among those which are assigned to it for observation.

Specify whether the agent should calculate the relative or the absolute frequency of occurrence in the "Frequency" (see chap. 4.8.5.4, p. 413) tab in the window "Agent Configuration" (see fig. 342). You can also extend the settings concerning result calculation there.

Hint: The result function "Frequency" can only be assigned to agents in Business Process Models. Reference objects can be activities and Subprocesses.

If an agent is to calculate both the absolute and the relative frequency, you must **assign** the "Frequency" result function **twice** to this agent.

ATTENTION: An agent always perceives only its own model (i.e. the model in which it was created) and all the subprocesses of this model. Therefore, if you simulate a model which contains an agent **in one of its subprocesses**, this agent does **not** determine the frequency of occurrence from the point of view of the main model, but only **from the point of view of its own model** (which is a subprocess of the main model simulated). The agent's model therefore always has a relative frequency of occurrence of 1 (i.e. of 100%) - independent of the probability with which it is called by the main model of the simulation. Corresponding to this, the frequencies of occurrence of activities and subprocesses within a deeper level must be regarded in relation to the agent's model.

4.8.4.4 Result function "Execution costs"

The result function "Execution costs" enables you to calculate those costs that occur at the executing objects within your organization while the process is executed. Typical examples for this are personnel costs or resource costs.

Hint: The differences between this function and the result function "Execution costs (conditional)" (see chap. 4.8.4.5, p. 404) are described in the chapter "Execution costs" vs. "Execution costs (conditional)" (see p. 404).

The execution costs are calculated by multiplying the execution times of the performers or resources by their hourly wages. In addition, the execution costs are weighted against the frequency of occurrence of the objects executed.

The result "Execution Costs" can only be calculated in the simulation algorithm "Capacity analysis" (see chap. 4.3, p. 354).

Hint: If you also wish to calculate the execution costs in the "Workload analysis" (see chap. 4.4, p. 367) you must employ the result function "Execution costs (conditional)" (see chap. 4.8.4.5, p. 404).

Hint: The result function "Execution costs" can be assigned to agents in all models that can be simulated. Reference objects can be activities, Subprocesses, performers and resources.

You can define whether the execution costs are to be calculated for performers, resources or other class of your application library in the "Execution costs" (see chap. 4.8.5.5, p. 414) tab in the "Agent configuration" (see fig. 342) window, and extend the settings (see chap. 4.8.5.5, p. 414) concerning the calculation of the results.

If your agent only observes **parts of the process simulated**, it will calculate the execution costs only with regard to the process part monitored.

Each result function "Execution costs" can only calculate the costs with regard to one single class (e.g. performer, resource, etc.). If you wish to calculate the execution costs simultaneously for several classes, you must **assign** this result function **several times** to this agent.

4.8.4.5 Result function "Execution costs (conditional)"

The result function "Execution costs (conditional)" enables you to calculate the costs that occur for the executing objects within your organization while the process runs. Typical examples for this are personnel costs or resource costs.

> **Hint:** The differences between this function and the result function "Execution costs" (see chap. 4.8.4.4, p. 403) are described in the chapter "Execution costs" vs. "Execution costs (conditional)" (see p. 404).

The execution costs are calculated by multiplying the execution times of the executing performers or resources (in hours) by the corresponding hourly wages.

> **Hint:** The result "Execution costs (conditional)" can be calculated in the simulation algorithms "Capacity analysis" (see chap. 4.3, p. 354) and "Workload analysis" (see chap. 4.4, p. 367).

> **Hint:** The result function "Execution costs (conditional)" can be assigned to agents in all model types that can be simulated. Reference objects can be activities, Subprocesses, performers and resources.

You can define whether the execution costs are to be calculated for performers, resources or another class of your application library in the "Execution costs (conditional)" (see chap. 4.8.5.6, p. 415) tab within the window "Agent configuration" (see fig. 342). In addition, you can extend the settings concerning the calculation of results.

If your agent only observes **parts of the process simulated**, it will calculate the execution costs only with regard to the process part actually monitored.

Each result function "Execution costs (conditional)" can only calculate the costs with regard to one single class (e.g. performer, resource, ...). If you wish to calculate the execution costs simultaneously for several classes, you must **assign** this result function **several times** to this agent.

"Execution costs" vs. "Execution costs (conditional)"

The result function "**Execution costs**" (see chap. 4.8.4.4, p. 403) calculates the expected and weighted execution costs for your organization's objects (such as performers, resources etc.). The *expected* execution costs of a resource, for example, which is only used in every second process run are lower than those of a resource used in every run.

The result function "**Execution costs (conditional)**" (see chap. 4.8.4.5, p. 404) calculates the execution costs for your organization's objects *provided that the particular object is in fact used*. That is, the total execution costs are summed up only from those process runs during which the particular object was actually employed.

Example:

In the following model: (see fig. 347) an agent observes "Activity-3" and calculates both the "Execution costs" and the "Execution costs (conditional)" for the class "Resource". The "Resource-1" pictured has hourly wages set to 100. "Activity-3" has an execution time of 1 hour.

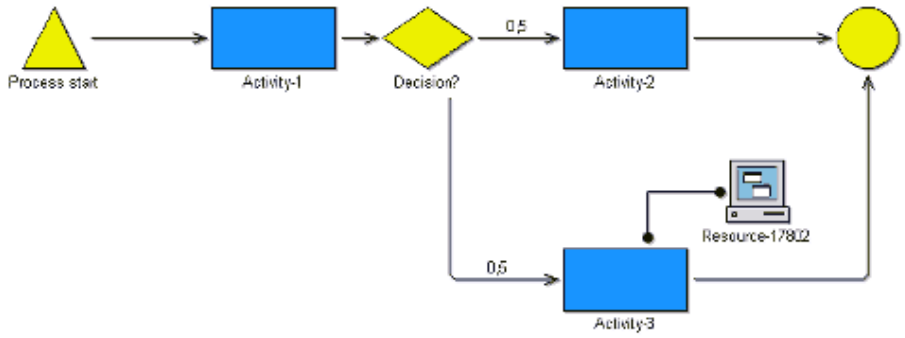


Figure 347: "Execution costs" vs. "Execution costs (conditional)

During the simulation "Activity-3" is on average only executed in every second process run. The "Execution Costs" calculated (i.e. the execution costs that are on average to be expected) are thus 50.

The "Execution Costs (conditional)" calculated, however, are 100, since only those process runs during which "Activity-3" was really executed and thus "Resource-1" was in fact employed were evaluated.

Hint: If you instruct an agent to calculate the "Main Process Results (total)" (see chap. 4.8.5.1, p. 410), all those process runs in which at least one reference object was involved are taken into account for calculating the result "Execution Costs (conditional)".

4.8.4.6 Result function "Workload"

Using the result function "Workload", you can instruct an agent to calculate the workload of the executing objects within your organization. These may be performers or resources. If you instruct an agent to calculate the workload of resources, it assumes that these resources are available 24 hours a day.

Hint: The result "Workload" can only be calculated in the simulation algorithm "Workload Analysis" (see chap. 4.4, p. 367).

Hint: The result function "Workload" can be assigned to agents in all model types that can be simulated. Reference objects can be activities, Subprocesses, performers and resources.

You can define whether the workload should be calculated for performers, resources or another class of your application library in the "Workload" (see chap. 4.8.5.7, p. 416) tab within the window "Agent configuration" (see fig. 342), . In addition, you can extend the settings concerning the calculation of results.

If your agent only observes **parts of the process simulated**, it will calculate the workload only with regard to the process part observed.

Each "Workload" result function can only calculate the workload with regard to one single class (e.g. performer, resource, etc.). If you wish to calculate the workload simultaneously for several classes of objects, you must **assign** this result function **several times** to the agent.

4.8.4.7 Result function "Capacity"

The "Capacity" result function calculates the capacity of performers and/or resources caused by the process execution. This is tantamount to the personnel or resource capacity demand required.

The "Capacity" result can only be calculated in the simulation algorithm "Capacity analysis" (see chap. 4.3, p. 354).

Hint: The "Capacity" result function can be assigned to agents in all model types that can be simulated. Reference objects can be activities, Subprocesses, performers and resources.

You can define whether capacity should be calculated for performers, resources or another class of your application library in the "Capacity" (see chap. 4.8.5.8, p. 417) tab within the window "Agent configuration" (see fig. 342) . In addition, you can extend the settings concerning the calculation of results.

If your agent only observes **parts of the process simulated**, it will calculate the capacity only with regard to the process part observed.

Each "Capacity" result function can only calculate the capacity with regard to one single class (e.g. performer, resource, etc.). If you wish to calculate the capacity simultaneously for more than one object class, you must **assign** this result function **several times** to the agent.

4.8.4.8 Result function "Cycle time"

Using the result function "Cycle Time", you can instruct an agent to calculate the cycle time of your Business Process Models.

Hint: The result function "Cycle time" can only be assigned to agents in models of the BP library. Reference objects can be activities and Subprocesses.

It is possible to extend the settings concerning result calculation within the "Cycle time" (see chap. 4.8.5.9, p. 418) tab in the window "Agent configuration" (see fig. 342).

If you instruct the agent to observe only **partial processes or subprocesses**, it calculates the cycle time only for these partial or subprocesses.

If you instruct the agent to observe **modelling objects which are not directly related to each other** (such as an activity at the beginning of the process and an activity at the end of the process, but no activities in between), the agent calculates the cycle time of each process part separately and renders as total the sum of all cycle times. parallelism are taken into account.

Example:

The model pictured below (see fig. 348) contains the following times:

Object	Execution time	Waiting time
Activity 1	10 Minutes	5 Minutes
Activity 2	10 Minutes	5 Minutes
Activity 3	2 Minutes	2 Minutes
Activity 4	2 Minutes	2 Minutes
Subprocess 1	30 Minutes	10 Minutes

Table 2: Execution and waiting times in the example "Cycle Time"

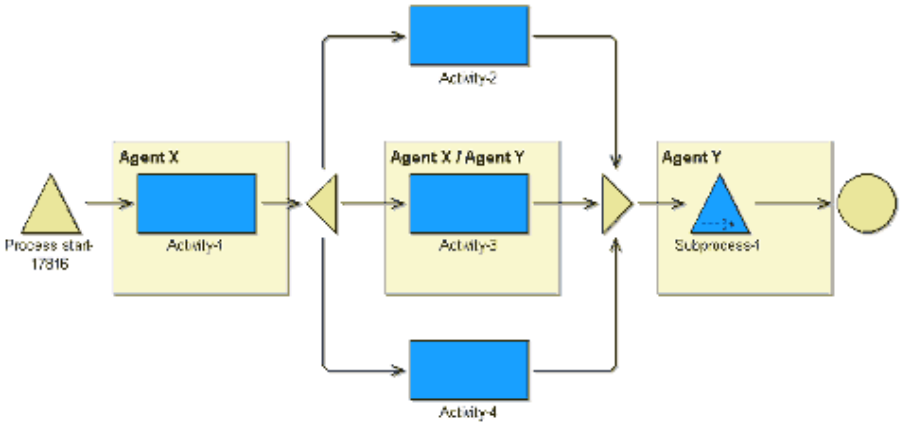


Figure 348: Agents - Cycle time (example)

- *Case 1:*

  "Agent X" calculates the cycle time for the following process part: **"Activity 1"** and **"Activity 3"**.

  The result is `(10 + 5) + (2 + 2) =` **`19 minutes`** In this case, the parallelity does not influence the cycle time of the process part.

- *Case 2:*

  "Agent Y" calculates the cycle time for the following process part **"Activity 3"** and **"Subprocess"**.

  "Activity 3" takes (2 + 2) = 4 minutes. Simultaneously, "Activity 2" (which is not being observed) is executed. This delays the execution start of the "Subprocess" by 11 minutes. (As Activity 2 takes (10 + 5) = 15 minutes which is an additional 11 minutes over the time taken by "Activity 3". The "Subprocess 1" then needs another (30 + 10) = 40 minutes. In total, the cycle time of the process part thus amounts to`(2 + 2) + 11 + (30 + 10) =` **`55 minutes.`**

4.8.4.9 Result function "Special formula"

The result function "Special formula" enables you to instruct the agent to calculate completely individual results. You may specify the calculation formula yourself for the desired result in the agent.

This result function produces new results for you by combining existing results. You may combine the results of other result functions and constant numbers using the following mathematical operators:

+	Addition
-	Subtraction
*****	Multiplication
/	Division
abs (*x*)	AbsoluteSum
acos (*x*)	Arcus Cosinus
asin (*x*)	Arcus Sinus
atan (*x*)	Arcus Tangens

cond (*Condition* , *x* , *y*) If the condition is true, *x* is the result of this function. Else, *y* is the result.

cos (*x*)	Cosinus
cosh (*x*)	Cosinus Hyperbolicus

exp (*x*)	e raised to the x-th power
log (*x*)	Logarithm
log10 (*Value*)	Logarithm on base 10
max (*x* , *y*)	Maximum; The result of this operator is the greater of the two values x or y.
min (*x* , *y*)	Minimum; The result of this operator is the smaller of the two values x or y.
pow (*x* , *Power*)	x raised to the power specified
sin (*x*)	Sinus
sinh (*x*)	Sinus Hyperbolicus
tan (*x*)	Tangens
tanh (*x*)	Tangens Hyperbolicus
sqrt (*x*)	Square Root

You may insert the following values as parameters **"x"** or **"y"** of the mathematical operators:

- **Names of Result Functions**,

  `such as sin("Personnel costs")`

- **Numbers**,

  `such as sin(250)`

- **Valid Mathematical Expressions**, i.e. closed calculation formulas (to which the same rules apply),

  `such as sin( max("Personnel costs", "Resource costs") )`

  Hint: You may **not** use the result functions "Special formula" and "Animation" in the calculation formula.

You can enter the calculation formula to be used in the "Special formula" (see chap. 4.8.5.10, p. 419) tab within the "Agent configuration" window (see fig. 342).

Hint: This result function will only produce results when used in a simulation algorithm within which all the result functions used in the calculation formula also produce results.

If an agent is to calculate several individual results, you must assign the "Special formula" result function several times to it and specify an appropriate calculation formula each time.

Examples:

- *Example 1:*

  In order to add "Process costs" and "EDP costs" (provided the agent also calculates these results), you have to specify the following calculation formula:

  `"Process Costs" + "EDP Costs"`

- *Example 2:*

  To find the square root of "Execution time" (provided the agent also calculates "Execution time"), you have to specify the following calculation formula:

  `sqrt("Execution time")`

- *Example 3:*

To calculate using the number "1", where the "Execution time" is greater than 0 or the number "0" where "Execution time" is less than or equal to 0, you can specify the following expression in the calculation formula:

```
cond("Execution time" > 0, 1, 0)
```

4.8.4.10 Result function "Sum over attribute"

The result function "Sum over attribute" can calculate the expected value of any numerical or time attribute (such as "Costs" or "Execution time") for one process run. The attribute must belong to the class "Activity".

You can define which attribute is to be evaluated in the "Sum over Attribute" (see chap. 4.8.5.11, p. 421) tab within the "Agent Configuration" window (see fig. 342). In addition, you can extend the settings concerning the result calculation.

Hint: This result function can only be assigned to agents in models of the BP library. Reference objects can be activities and Subprocesses.

By means of this result function you can for example instruct an agent to calculate the expected times and costs of your business processes. If you let the agent observe only **modelling objects along certain process paths**, you will obtain the particular expected value for the path monitored. If you let the agent observe only certain process parts (such as a certain subprocess call), you will obtain the isolated result for the process part.

Example:

If you only assign **"Activity-2"**, **"Activity-3"** and **"Activity-4"** as reference objects to an agent in the model represented below (see fig. 349), you will only obtain results for the process part included in the parallelity.

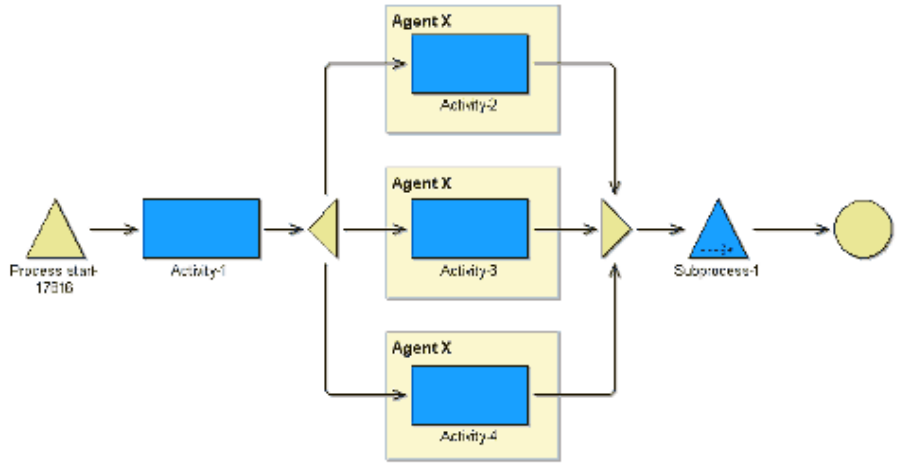


Figure 349: Agents - Sum over attribute

If different attributes are to be evaluated by the same agent, you must **assign** this result function **several times** to the agent.

4.8.5 Configure result functions

For each result function you instruct an agent to execute, a separate tab is added in the "Agent Configuration" window (see fig. 342). Within this tab, more detailed instructions can be specified for the agent with regard to the calculation of the specific results (see chap. 4.8.4, p. 400).

4.8.5.1 Basic configuration

The settings listed below (see fig. 350) can be made for most result functions.

The only exceptions are the "Animation" (see chap. 4.8.6.1, p. 423), "Capacity" (see chap. 4.8.5.8, p. 417) and "Special formula" (see chap. 4.8.5.10, p. 419) tabs which only allow you to make some of the settings explained below.

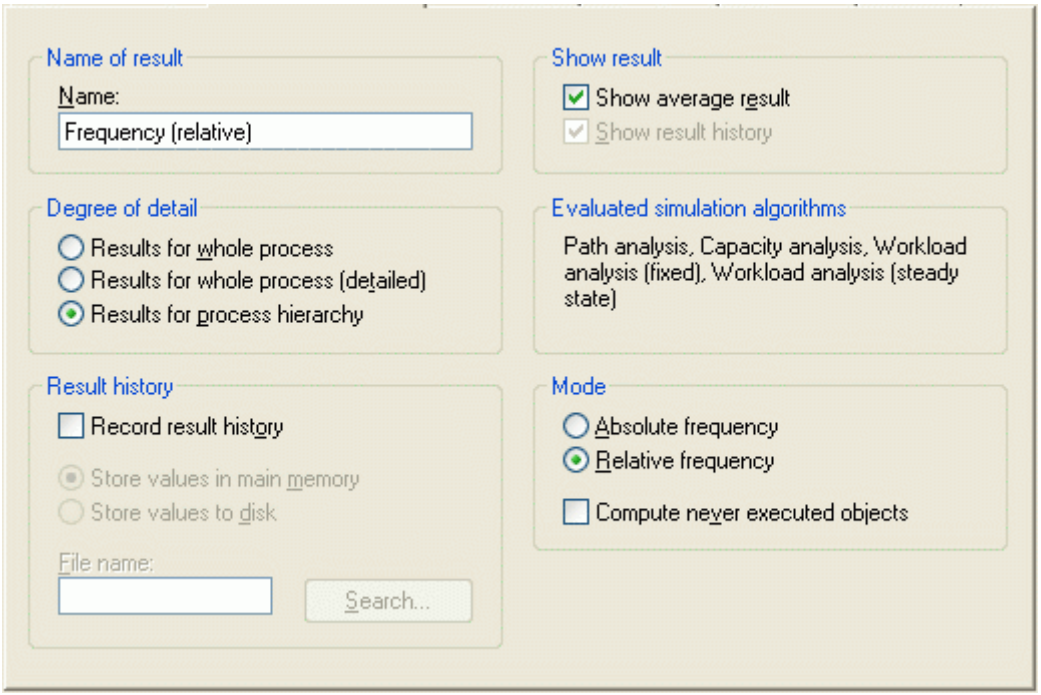


Figure 350: Agents - General result settings

- **"Name of result"**

  In this field you can enter a name for the result calculated. The title of this tab is automatically adjusted to the new name.

- **"Degree of detail"**

  The degree of detail of the evaluation indicates to which extent the agent calculates the particular result. Selecting a high degree of detail ("Results for process hierarchy") can considerably slow down the simulation.

  - *"Results for whole process"*

    If you select this option, the agent calculates a single number as result. This number is the average total result of all the agent's reference objects.

  - *"Results for whole process (detailed)"*

    If you select this option, the agent calculates a number *for each modelling object observed in its model*. If the agent monitors subprocess calls, it will calculate the average total result of the subprocess referenced for the object that called the subprocess.

- *"Results for process hierarchy"*

  If you select this option, the agent calculates a result *for each modelling object observed in its model and in all subprocesses observed*. If the agent monitors subprocess calls, a separate result is calculated for each activity in the subprocess referenced (and in any further subprocesses on lower levels). The results are calculated separately for different process contexts: i.e. if a subprocess on a lower level is called from different models, you will obtain a separate result for each path on which this subprocess was reached.

ATTENTION: The results produced when the degree of detail is "**Results for process hierarchy**" can be extremely extensive (especially in the case of large models and where there are complex interrelations between models). We recommend that you select this degree of detail only when you need separate information about specific process parts or subprocesses. To restrict the amount of results generated, we recommend that you assign few referenced objects to the particular agent.

- **"Result history"**

  You can instruct your agent to draw the development of the result in the course of the simulation. This can then be displayed after the simulation in the form of a function graph (similar to the animation shown by an agent). It informs you whether the result values have found their level in an average value towards the end of the simulation. If you simulate using "Workload Analysis" (see chap. 4.4, p. 367), you can also deduce variations over the course of time (such as seasonal variations) from the development of the results. You can also save the development of results as a file and investigate it using other software products (such as spreadsheet calculation).

  Activate the option "Record result history", to store the development of the results and then select one of the following options:

  - **"Store values in main memory"** or
  - **"Store values to disk"**

  When you select the option **"Store values to disk"**, you must also enter the path and filename into the field "File name".

  Note: Saving values to a file takes up additional time during the simulation. However by doing this, you can edit the data later on using other software products. If you select the option "Store values in main memory", the simulation is a bit faster, but you cannot re-use the data later. In the latter case, more storage space is required on your computer during the simulation. The extent of the additional capacity needed depends on the number of reference objects and the degree of detail specified for the evaluation.

- **"Show result"**

  Activate the option **"Show average result"**, to have the result calculated by this result function displayed after the simulation.

  In some cases you may wish to calculate but not to view the result, for example when the agent calculates a result which only serves as input for a "Special formula" (see chap. 4.8.5.10, p. 419) result function.

  Activate the option **"Show result history"**, to have the result history displayed after the simulation.

  Note: This option is only available if you also selected the option "Record result history".

- **"Evaluated simulation algorithms"**

  The entries in this field are to inform you during which simulation algorithm this result function is used. If an agent is instructed to calculate several results, it will always only calculate those results which it can during the current simulation algorithm and ignore all other result functions.

4.8.5.2 "Animation"

Settings concerning the result function "Animation" (see chap. 4.8.4.1, p. 401) can be made on the Animation (see fig. 351) tab.

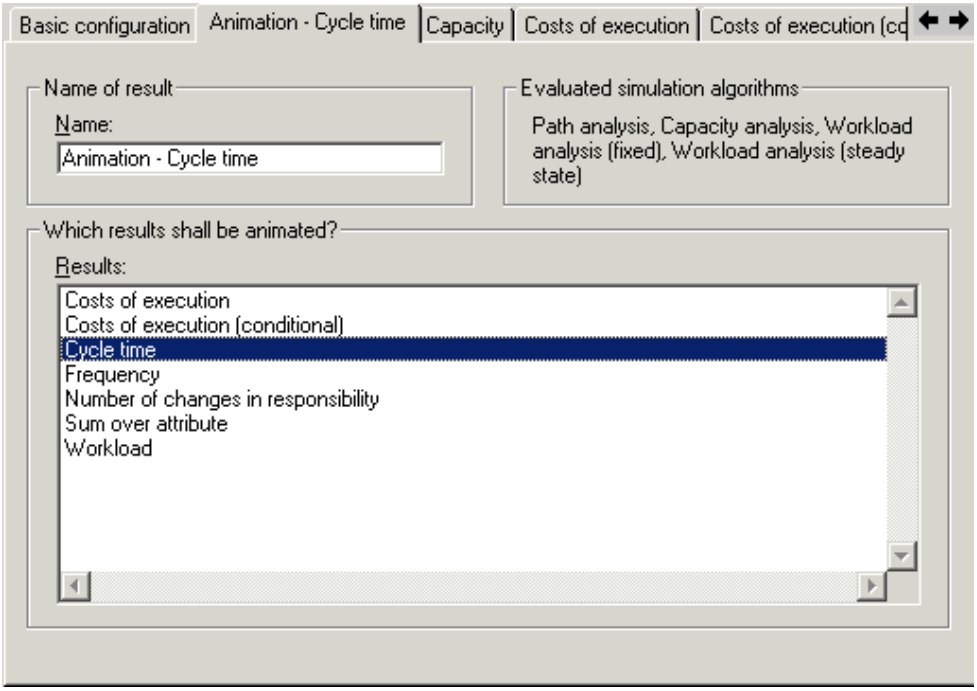


Figure 351: Agents result function - Animation

Enter a name for the animation function (such as "Animation of the Cycle Time") into the field **"Name"**.

Hint: The name of the tab card is adjusted automatically if you change the name.

The field **"Evaluated simulation algorithms"** lists those simulation algorithms during which the animation is executed.

Hint: If the results to be animated are not calculated during a certain simulation algorithm, then the animation will not be executed. This is not explicitly taken into account in the group "Evaluation simulation algorithms".

In the **"Results"** list you can select those results which your agent is to represent graphically in the course of the simulation. The list "Results" contains all this agent's result functions for which an "Animation" (see chap. 4.8.4.1, p. 401) is possible. Select as many results as you wish from the list.

ATTENTION: The animation may become slow and confusing when you select many results.

Should the "Results" list be empty, you must first assign further result functions to the agent which you then can select for animation. In order to do this, select the file/record card "Basic configuration" (see chap. 4.8.3.1, p. 397) and click on the button "Add/Remove Results" (see p. 399).

Hint: The results "Special formula", "Animation" and "Capacity" **cannot** be represented by the animation.

4.8.5.3 "Number of changes in responsibility"

The "Number of changes in responsibility" (see fig. 352) tab is used to instruct the agent in detail concerning the calculation of the result "Number of changes in responsibility" (see chap. 4.8.4.2, p. 401).

Hint: The name of the tab is adjusted automatically if you change the name.

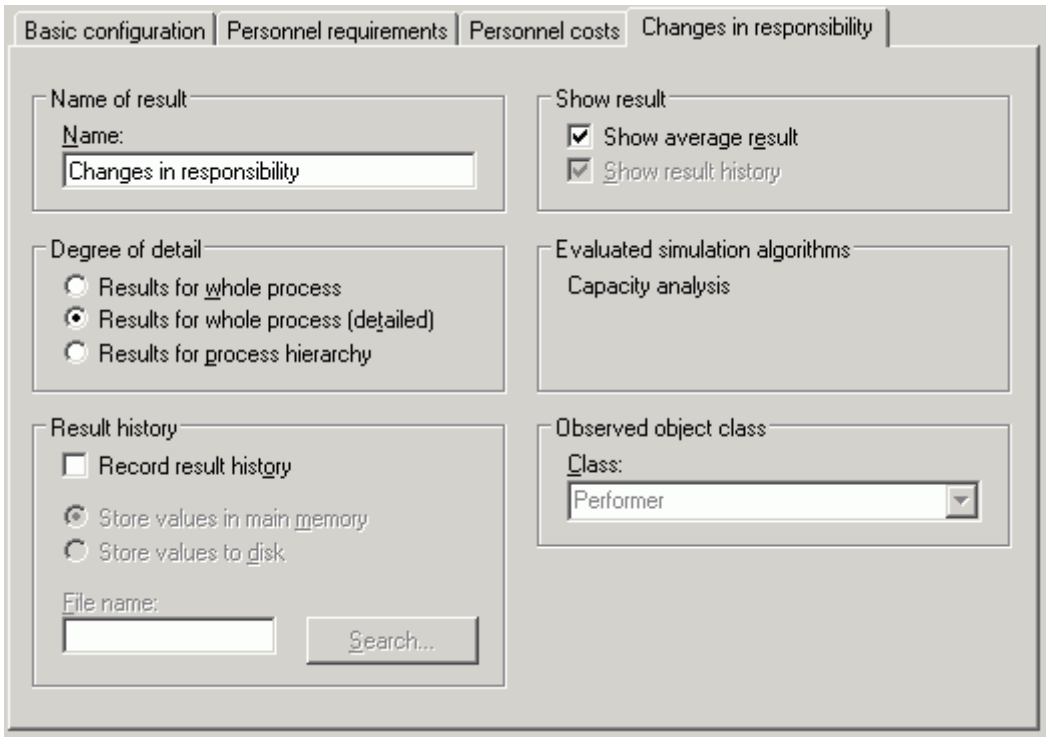


Figure 352: Agents result function - Number of changes in responsibility

In addition to the "Basic configuration" (see chap. 4.8.5.1, p. 410) you must specify the class for which the number of changes of responsibility are to be calculated. To do this, select a class from the list **"Class"**.

Hint: It is only possible to calculate "Number of changes in responsibility" for objects of the classes "Performer" and "Resource".

Example:

If your agent is to determine the number of changes of performers, you have to select the class "Performer" from the list.

4.8.5.4 "Frequency"

You can instruct the agent in detail with regard to the calculation of the result "Frequency" (see chap. 4.8.4.3, p. 402) on the "Frequency" (see fig. 353) screen.

Hint: The name of the tab is adjusted automatically, if you change the name.

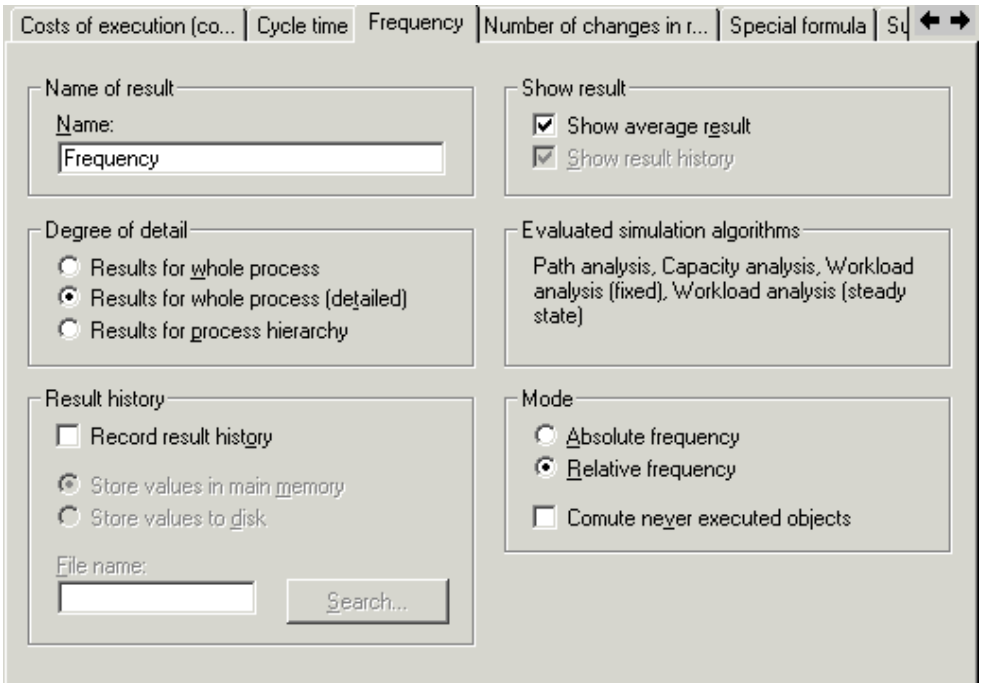


Figure 353: Agents result function - Frequency

In addition to the "Basic configuration" (see chap. 4.8.5.1, p. 410) one of the following modes can be specified:

- **"Absolute frequency"**

  Select this option to instruct the agent to calculate the absolute frequency of occurrence of the models and modelling objects. The absolute frequency of occurrence indicates how often during the simulation an object was executed. For example, if you simulate with a higher number of process runs, the absolute frequency of occurrence also increases. With the help of the absolute frequency you may for example find out, how often a certain process path was run through.

- **"Relative Frequency"**

  Select this option to instruct the agent to calculate the relative frequency of occurrence of the models and modelling objects. The relative frequency of occurrence is the frequency of occurrence of these objects in percent *with regard to the agent's model*. The model in which the agent was created always has a relative frequency of occurrence of 1 (i.e. of 100%).

By activating the option **"Identify unexecuted objects"** you can instruct the agent to search among the modelling objects observed for objects which were never executed in the course of the simulation. In this way, you can detect "dead activities" and "dead paths" in your processes.

> **Hint:** "Dead paths" are paths which are never executed in the course of a simulation run. Since the simulation works with random generators, it is possible, however, that these paths would be executed in other simulation runs.

If you do not select this option, the agent ignores objects which are never executed. They are not taken into account in the agent's result representation (*not even* with a frequency of occurrence of "0").

4.8.5.5 "Costs of execution"

You can instruct the agent in detail regarding the calculation of the result "Execution costs" (see chap. 4.8.4.4, p. 403) on the "Costs of execution" tab (see fig. 354).

> **Hint:** The name of the tab is adjusted automatically if you change the name.

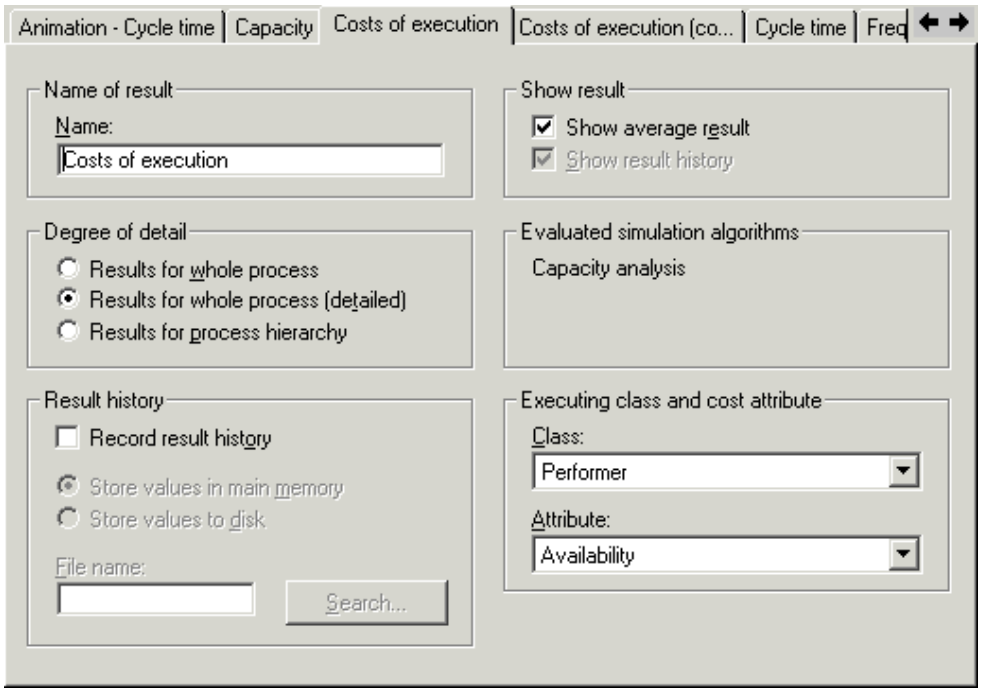


Figure 354: Agents result function - Costs of execution

In addition to the "Basic configuration" (see chap. 4.8.5.1, p. 410) settings, you can select the class for which the execution costs are to be calculated from the **"Class" list**. Then select the attribute which represents the costs per hour or the hourly wages.

Hint: You can only select from performer and resource classes.

Example:

To calculate the personnel costs, select the class "Performer" from the "Class" list and the attribute "Hourly Wages" from the "Attribute" list.

Hint: The difference between this result function and the result function "Execution Costs (conditional)" (see chap. 4.8.4.5, p. 404) are described in the chapter "Execution Costs" vs. "Execution Costs (conditional)" (see p. 404).

4.8.5.6 "Costs of execution (conditional)"

You can instruct the agent in detail with regard to the calculation of the result "Execution Costs (conditional)" (see chap. 4.8.4.5, p. 404) on the "Costs of execution(conditional)" tab (see fig. 355).

Hint: The name displayed on the tab is adjusted automatically if you change it.

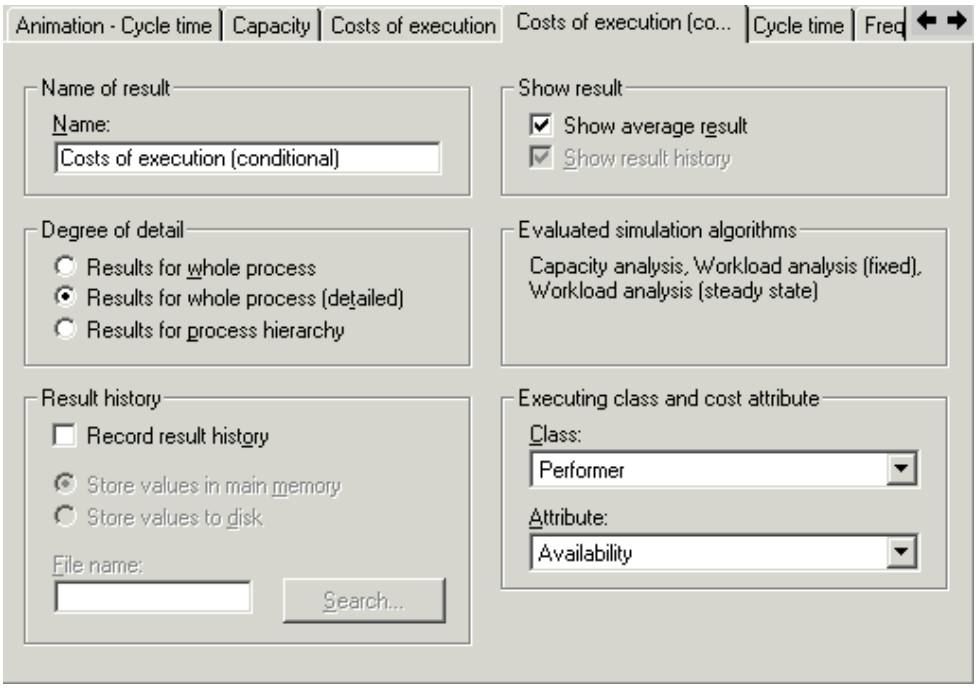


Figure 355: Agents result function - Costs of execution

In addition to the "Basic configuration" (see chap. 4.8.5.1, p. 410) settings, you can select the class for which the execution costs are to be calculated from the **"Class" list**. Then select the attribute which represents the costs per hour or the hourly wages.

Hint: You can only select from among performer and resource classes.

Example:

To calculate personnel costs, select from the list "Class" the class "Performer" and from the list "Attribute" the attribute "Hourly Wages".

Hint: The differences between this result function and the result function "Execution Costs" (see chap. 4.8.4.4, p. 403) are described in the chapter "Execution Costs" vs. "Execution Costs (conditional)" (see p. 404).

4.8.5.7 "Workload"

You can instruct the agent in detail regarding the calculation of the "Workload" (see chap. 4.8.4.6, p. 405) result on the "Workload" tab (see fig. 356).

Hint: The name displayed on the tab is adjusted automatically if you change it.

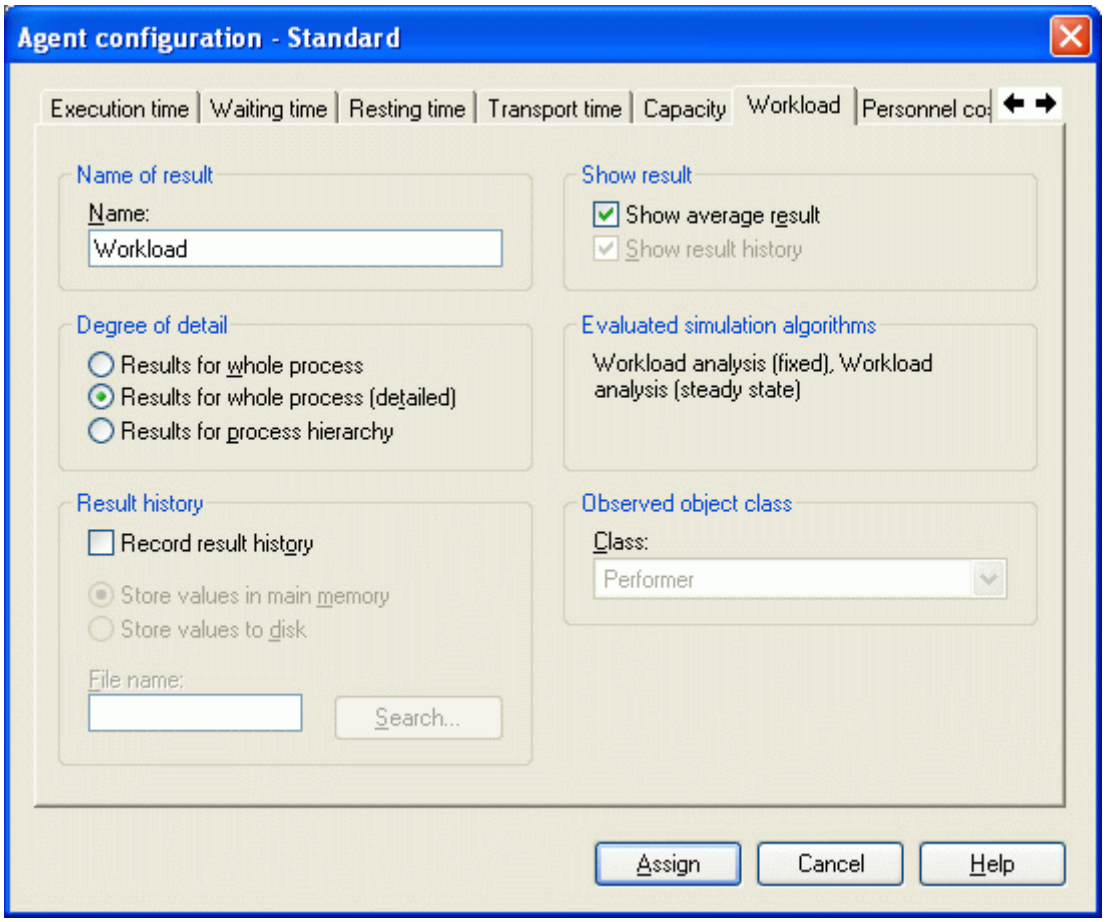


Figure 356: Agents result function - Workload

In addition to the "Basic configuration" (see chap. 4.8.5.1, p. 410) settings, you can select the class for which the workload is to be calculated from the **"Class"** list.

Hint: You can only select from among performer and resource classes.

Example:

Select the class "Resource" from the list in order to calculate the workload of resources.

4.8.5.8 "Capacity"

You can instruct the agent in detail regarding the calculation of the result "Capacity" (see chap. 4.8.4.7, p. 405) on the "Capacity" (see fig. 357) tab.

Hint: The name displayed on the tab is adjusted automatically if you change it.

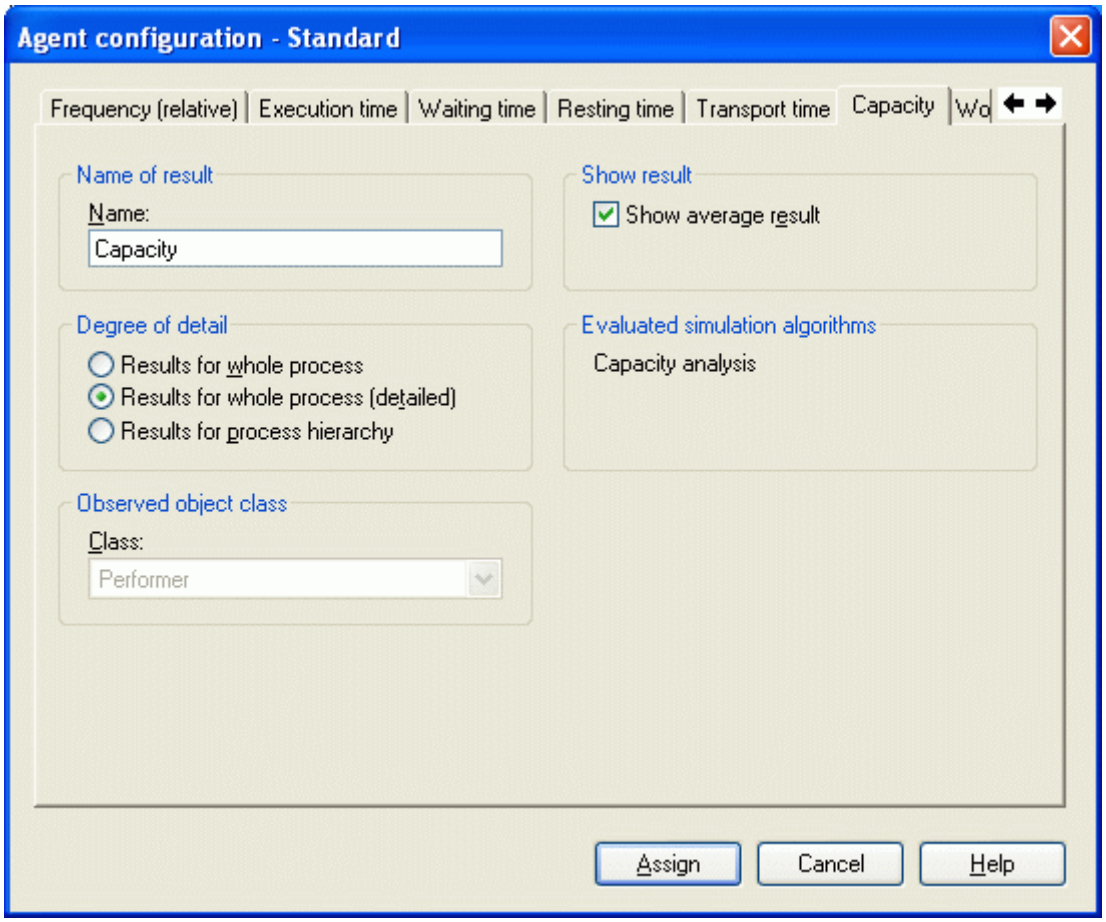


Figure 357: Agents result function - Capacity

The usual "Basic configuration" (see chap. 4.8.5.1, p. 410) settings can be made apart form the following exception: the capacity cannot be calculated in a sensible manner before the simulation is finished and is thus not continuously calculated during the simulation. Therefore, it is impossible to represent the development of the result for capacity.

In addition, you can select the class for which the capacity is to be calculated from the **"Class"** list.

Hint: You can only select from among performer and resource classes.

Example:

If you wish to calculate the capacity of the resources employed, you must select the class "Resource" from the list.

4.8.5.9 "Cycle time"

Only the usual "Basic configuration" (see chap. 4.8.5.1, p. 410) settings can be made on the "Cycle time" (see fig. 358) tab.

Hint: The name displayed on the tab is adjusted automatically if you change it.

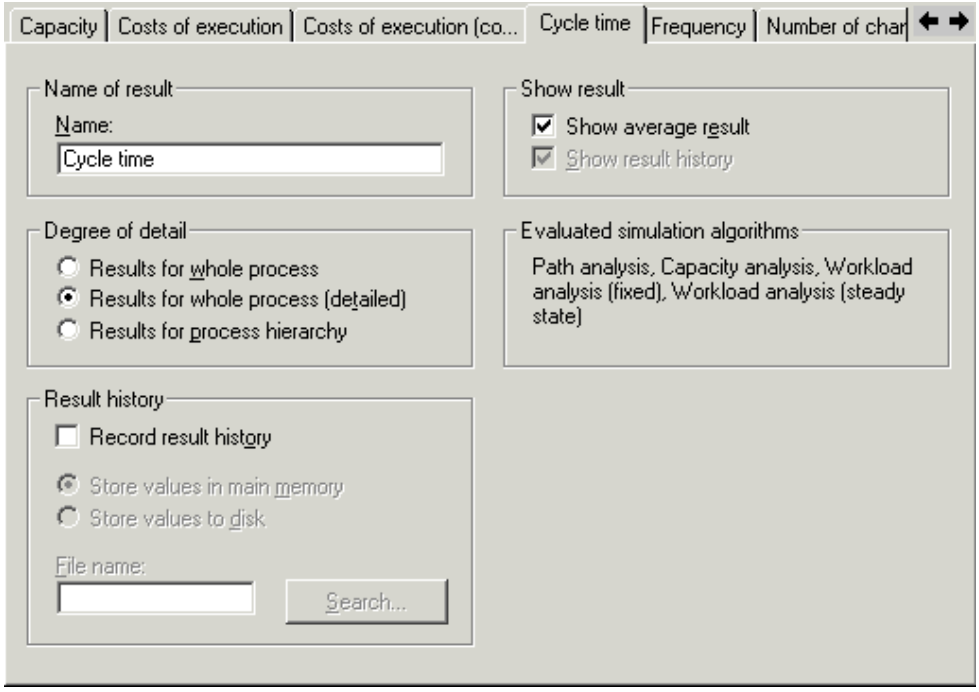


Figure 358: Agents result function - Cycle time

4.8.5.10 "Special formula"

You can instruct the agent in detail regarding the calculation of the result "Special formula" (see chap. 4.8.4.9, p. 407) on the "Special formula" (see fig. 359) tab.

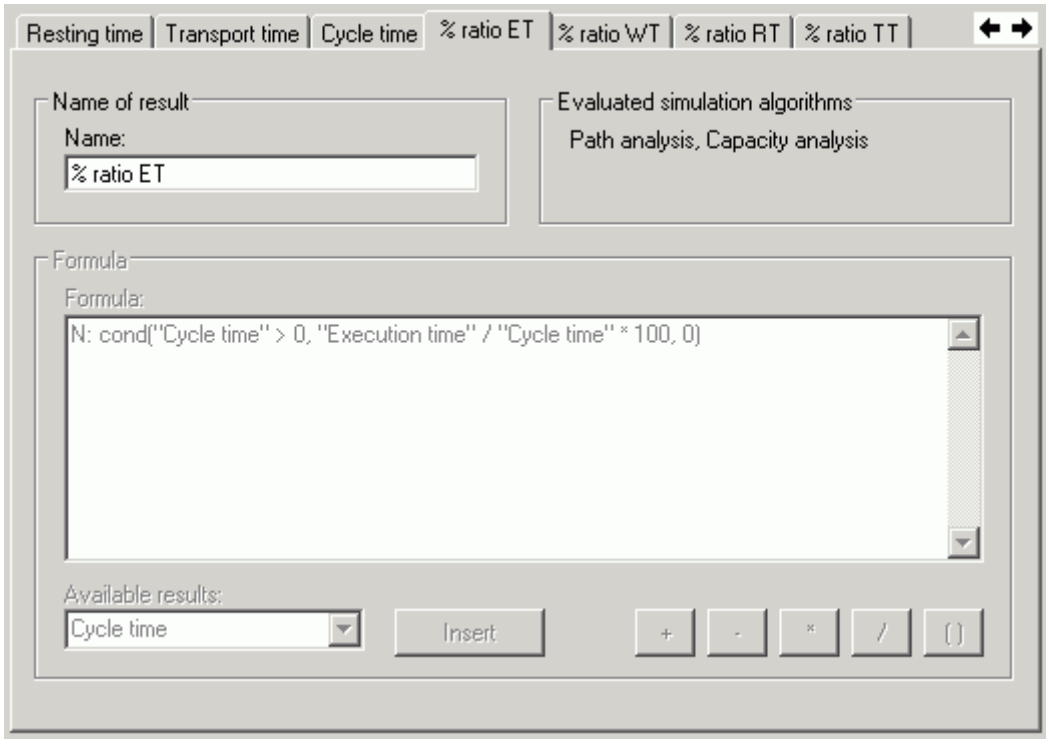


Figure 359: Agents result function - Special formula

Enter a name for the "Special formula" into the field **"Name"**.

Hint: The name displayed on the tab is adjusted automatically if you change it.

The group **"Evaluated simulation algorithms"** lists all simulation algorithms in which the "Special formula" can be calculated.

Hint: If a result on which the "Special formula" is based is not calculated during a certain simulation algorithm then this "Special formula" will not be calculated during this simulation algorithm.

The field **"Formula"** is used to define the formula which this function uses to calculate a result. A calculation formula consists of result names, mathematical operators and, if necessary, numbers.

To define a calculation formula, select a result on which your identification number is to be based from the list **"Available results"** and then click on the button "Insert".

Click on the buttons "+", "-", "×", "/" or "( )" to insert mathematical operators into the calculation formula.

To edit the formula directly, click on the field "Formula". In this way, you may also insert constant numbers into the formula as well as specific operators (see chap. 4.8.4.9, p. 407). **Combine results, numbers and mathematical operators to create the formula you require.**

Hint: If you change the names of results which you are currently using within this calculation formula, you must also adjust the formula accordingly. Should this be necessary you will be informed via an error message while closing the window.

If the list "Available results" is empty, this means that you must first assign result functions to the agent, which can then be used in the formula. In order to do this, select the "Basic configuration" (see chap. 4.8.3.1, p. 397) tab and then click on the button "Add/Remove Results" (see p. 399).

Hint: You cannot insert the result function "Animation" or other "Special formula" numbers into this function.

Syntax of the calculation formula

The following rules apply when creating a calculation formula:

- Your formula must contain at least one result name. This result name must represent a result which is also calculated by this agent. It serves as a basis for the "Special formula".
- All result names used must be placed between inverted commas. If the result names contain the symbols **"** or ****, these have to be "masked" by placing a back slash **** in front of them. If you add results to the formula using the button "Insert", these adjustments are carried out automatically.
- The formula entered must be a valid mathematical expression, i.e. there must be a closing parenthesis for each opening parenthesis and result names and numbers must be linked by mathematical operators etc.

If you enter a formula that is syntactically incorrect, you will be informed about this as soon as you close the window. You must then correct the formula.

Example of a Calculation Formula

You have assigned the result functions "Costs of execution", "Sum over attribute" and "Special formula" to your agent and renamed the "Costs of execution" function as "Personnel Costs" and the "Sum over Attribute" function as "Process Costs (Expected Value)".

You want your agent to calculate the total costs using the "Special formula".

Enter the name "Total costs" into the field **"Name"**. Enter the following into the field **"Formula"**:

```
"Personnel Costs" + "Process Costs (Expected Value)"
```

Data Type

By default, the data type of the first result function used in the formula is assumed to be the data type ("Time" or "Numerical Value") of the "Special formula". You may, however, force the system to output the ADOxx time format by placing a **T:** in front of the calculation formula. Similarly placing an **N:** in front of the calculation formula causes a numerical value to be produced.

Example:

You want your agent to calculate which share of the cycle time is taken up by the execution time and thus enter the following calculation formula:

```
"Execution time" / "Cycle time" × 100
```

The first result function used in the calculation formula is "Execution time", thus a time value. By default, the result of the calculation formula will be issued in time format. To force the system to output a numerical value, you would have to change the formula as follows:

```
N: "Execution Time" / "Cycle Time" × 100
```

ATTENTION: When time is calculated, it is always converted into seconds.

To calculate the "Execution time" in hours, for example, you have to enter the following calculation formula:

```
N: "Execution Time" / 3600
```

4.8.5.11 "Sum over attribute"

You can instruct the agent in detail regarding the calculation of the result "Sum over Attribute" (see chap. 4.8.4.10, p. 409) on the "Sum over attribute" tab (see fig. 360)

Hint: The name displayed on the tab is adjusted automatically if you change it.

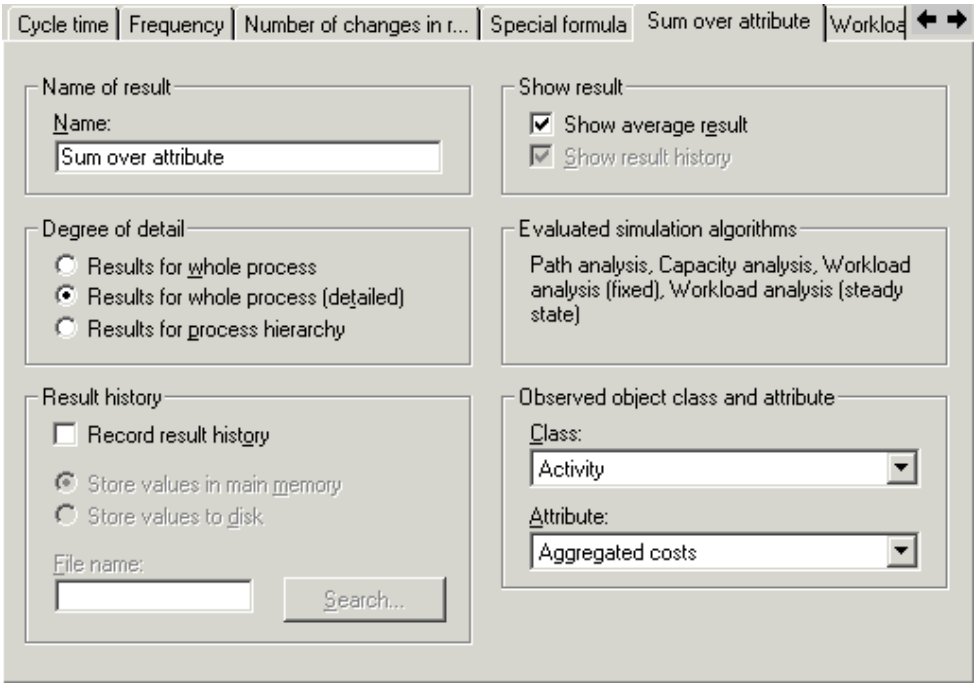


Figure 360: Agents result function - Sum over attribute

In addition to the "Basic configuration" (see chap. 4.8.5.1, p. 410) settings must enter the class to be observed and the attribute to be evaluated. To do this, select a class from the **"Class"** list and select the attribute for which the agent should calculate the expected value from the **"Attribute"** list.

Hint: You can only select from "Activity" classes.

Example:

Select the class "Activity" and the attribute "Costs" to calculate the average costs of a process.

4.8.6 Employing agents in simulations

Employing agents in simulations produces additional results. Proceed according to the following steps to employ agents:

Employ agents in simulations

1. Create agents in the models to be evaluated and/or activate existing agents (see chap. 4.8.2, p. 392).
2. Start the simulation with the models to be evaluated.
3. Click on the button **"Agents"** in the result window of the simulation. This opens a window in which you can select the agent results which you wish to see (see chap. 4.8.6.2, p. 424).

Employing agents in the offline animation

1. Create agents in the models to be evaluated and/or activate existing agents (see chap. 4.8.2, p. 392).
2. Start the Offline Animation (see chap. 4.5, p. 386) with a simulation protocol referring to the models to be evaluated (APF File).
3. Answer "Yes" to the question *"Some agents were detected in the simulated models. Do you want the agents to evaluate the offline animation?"* (Please note that this question only appears, if activated agents were found in the models to be evaluated.)

   Note: If agents are employed in an offline animation, it is impossible to rewind the animation.
4. Let the offline animation run through once. It is not necessary to let the whole simulation protocol run through. However, if you cancel the offline animation too soon, it is possible that the agents will not yet have calculated results (or will only have calculated imprecise results). We recommend that you instruct an agent to animate the results. In this way, you will be able to see from what point in the offline animation, exact results are available (i.e. from when the results begin to vary only slightly).
5. Close the offline animation. Then the Agent - result selection window (see chap. 4.8.6.2, p. 424) will open up automatically.

   Hint: If you specified that an agent should animate (see chap. 4.8.4.1, p. 401) the results, an appropriate animation window (see chap. 4.8.6.1, p. 423) will have been opened during the offline animation (see fig. 361).

4.8.6.1 The animation of results by agents

If you have assigned the result function "Animation" (see chap. 4.8.4.1, p. 401) to an agent, an animation window (see fig. 361), in which the development of the result is represented as a function graph, is shown during the simulation.

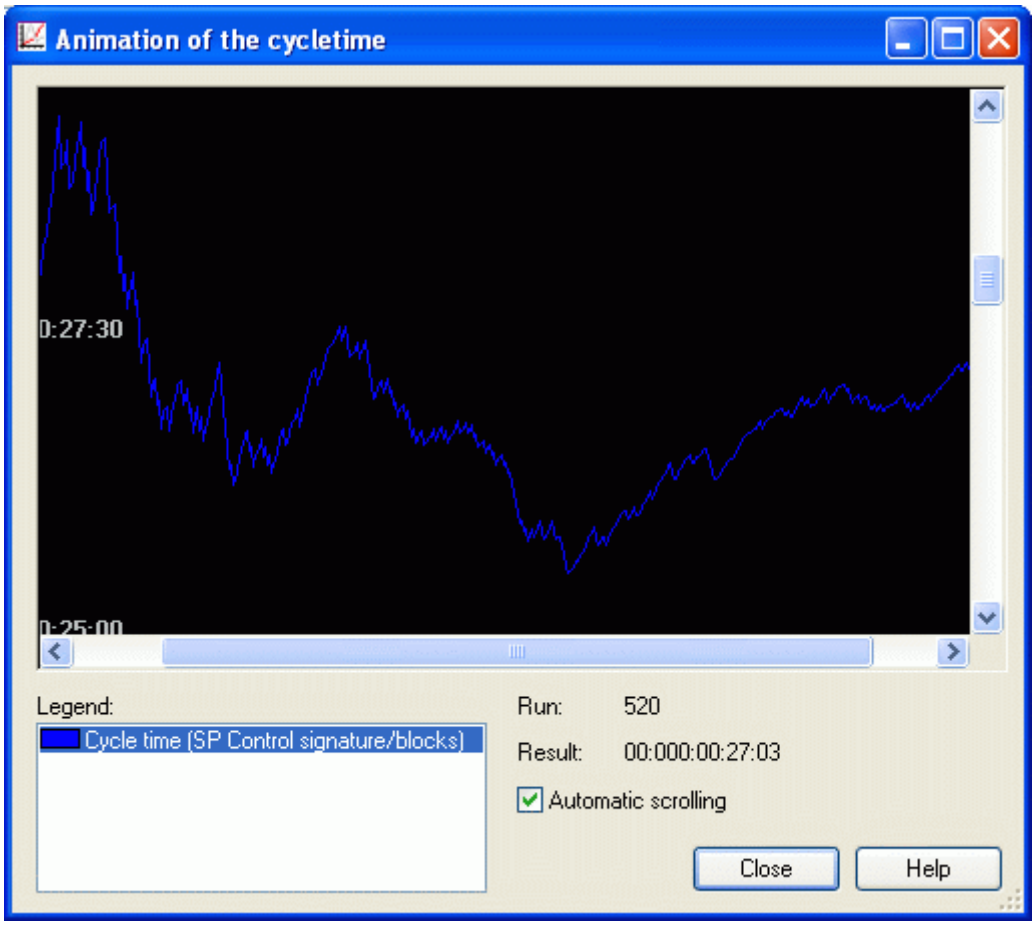


Figure 361: Animation of results by agents (example)

During the Path analysis (see chap. 4.2, p. 343) and the Capacity analysis (see chap. 4.3, p. 354) the animation window is shown as soon as the simulation has started.

In the case of the workload analysis (see chap. 4.4, p. 367) the animation window is shown no sooner than all oscillation phases are finished. In most cases, however, it will be shown some time after the result phase has started.

If you instructed the agent to monitor the simulation only during a certain period of time (you can configure this via the agent's evaluation calendar (see chap. 4.8.3.1, p. 397)), the animation window will not open prior to the start of the evaluation period.

You can find the following information in the animation window:

Title: The window's title is the name of the animation function (see chap. 4.8.5.2, p. 412).

Representation of the Graph: In the upper part of the window the development of the results is shown as a coloured function graph on a black surface. On the x-axis the number of completed simulation runs is shown whereas the y-axis shows the particular result value.

Hint: The size of the animation window can be changed. The drawing area is adjusted to the new window's size.

Legend: The legend contains an entry for each function graph represented on the drawing area. If you select an entry from the legend, the corresponding function graph is automatically centred on the drawing area.

If an animated result is calculated on the level of activities, you will obtain a function graph and an entry in the legend for each single activity result.

Hint: The maximum number of function graphs that can be represented simultaneously is 50 (independent of the number of animation windows). If this number is exceeded, you will be informed accordingly and no further function graphs will be created. In exceptional cases, this can lead to empty animation windows.

Run: The simulation runs evaluated are counted in this field.

Hint: Please note that this number does not necessarily match the number of runs set at the beginning of the simulation. The number of runs set at the beginning of the simulation determines how often the main process simulated is to be run through. However, if your agent evaluates a subprocess, it is possible - depending on the process structure - that this subprocess is run through more often or less often than the main process.

In the case of Workload analysis (see chap. 4.4, p. 367) it is also possible, that a main process run was started prior to the beginning of the result measuring phase (and is thus not counted by the simulation as "validly simulated") while a subprocess of this main process starts later, therefore during the result phase. If your agent only observes this subprocess, the subprocess run is counted by the agent and shown in the animation window whereas the run of the complete (i.e. main) process is not counted by the simulation.

Result: This field always shows the current evaluation result. The display refers to the entry selected in the legend.

Automatic Scrolling: By activating this option, you can cause the visible area of the drawing area to move automatically together with the function graph selected in the legend.

4.8.6.2 Agents - Result selection

In the window "Agents - Result selection" (see fig. 362) you select those agents the results of which are to be displayed.

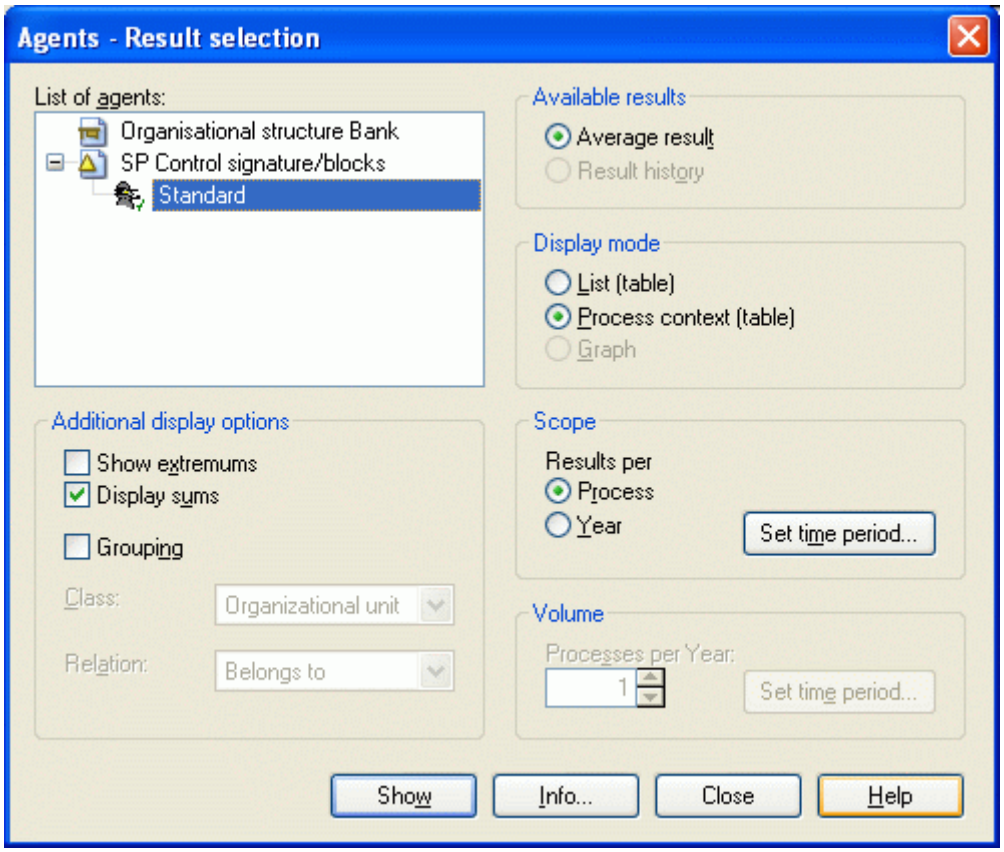


Figure 362: Agents - Result selection

All models simulated with the agents they contain can be seen in the field **"Agent list"**. If you select a particular agent, certain fields and buttons in the window are activated or deactivated, respectively, depending on the configuration and the results calculated by the agent.

Hint: Only agents which have actually calculated results are shown. If an agent is not active during the path analysis, for example, then it is not available for selection after the path analysis. Agents which you deactivated prior to the simulation or which are configured incorrectly, do not calculate results and are not shown here.

Right-click on an agent or a model to open the popup menu (see fig. 363) for the entry selected.

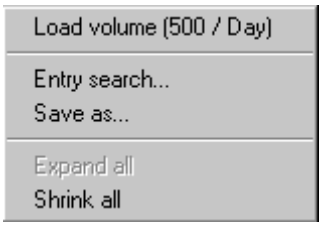


Figure 363: Popup menu for the loading of the quantity

Select the option **"Load volume"** to automatically transfer the process quantity of the model selected into the field **"Volume" (see p. 427)**. If an agent is selected, the process quantity of the model which contains this agent will be transferred.

Hint: This option is not available for models of the WE library.

Click on the button **"Show"** to display the results of the agent selected (see chap. 4.8.6.3, p. 430). The results are displayed according to the other settings you have made in this window.

Clicking on the button **"Info"** displays a window (see fig. 364) which contains a description of the agent selected and the results it calculates (see p. 428).

In addition, you can make the following settings:

- Available Results (see p. 426)
- Display mode (see p. 426)
- Scope (see p. 427)
- Volume (see p. 427)
- Additional Display Options (see p. 428)

Available results

Agents can calculate two types of results:

- **Average results** and
- **Result history.**

Average results are average final results (expected values), calculated from the simulation evaluation.

Average results are displayed in the "ADOxx Browser" (see chap. 8., p. 66).

Result histories show the progressive development of a result over the course of the simulation. A result history is a series of numbers displayed in the form of function graphs.

Each agent in the field "Agent list" (see p. 426) within the "Agents - result selection" window (see fig. 362) may have calculated a number of average results and/or a number of result histories. Select whether you wish to see the average results or the result history calculated by the agent in the "Available results" group. If one of these options is deactivated, the agent currently selected did not calculate any results of this type.

> **Hint:** Agents only record results histories if they have been configured accordingly (see chap. 4.8.5.1, p. 410).

Display mode

Depending on the Result type (see p. 426) selected, the window "Agents - result selection" (see fig. 362) offers you the following modes of representation:

For **Average results** you can choose between

- "Process context (table)" or
- "List (table)"

For **Result histories** the only option available is

- "Function graph"
- **"Process context (table)"**

  The option "Process context (table)" is the default display mode for agent results. Results are represented here in their process context, i.e. activity results are assigned to the processes in which they occur, subprocesses are assigned to their main processes and so on. If a subprocess is called by a number of different Business Process Models, the results are listed separately for each branch.

- **"List (table)"**

  In this mode of representation the results are listed without their process contexts in a flat list. Values which occur in different contexts are summarised.

  Whether the mode of representation "List (table)" is applicable to a certain result or not depends on the user's **interpretation** of the values listed in the table. When this mode of representation is used, you must always bear in mind that it shows **an aggregated view** on the results of the "Process context(table)" mode, but that the results are not re-calculated.

- **"Function graph"**

  This mode of representation is designed for result developments. These are always shown as function graphs in a coordinate system. The x-axis shows how the simulation progresses, the y-axis shows the particular result.

Scope

The period of time to which the results listed should refer can be defined in the "Scope" group.

You can select **"Results per process"**, to see the results referring to one single process run.

Alternatively, you can select **"Results per <time period>"**, to see the results referring to a specified period of time. You can select any period of time (see p. 429). The default period of time is "per year". To change the period of time referenced, click on the button **"Set time period"** (see p. 429).

> **Hint:** If you select "Results per <time period>", you must also specify a number of processes in the group "Volume" (see p. 427).

Volume

In the group "Volume" within the window "Agent - result selection" (see fig. 362) you enter the number of processes occurring in a certain period of time. This number is needed to calculate the "Results per <time period>" (such as the "Results per year").

> **Hint:** You can only specify a number, if you selected the option **"Results per <time period>"** in the group "Scope" (see p. 427).

> **Hint:** After a Workload analysis (see chap. 4.4, p. 367) the group "Volume" is **not** available, since the number of processes is determined by the simulation and thus cannot be changed.

Enter the particular number into the field **"Processes per <time period>"**. You can select any reference period of time for the number of processes. To do this, click on the button **"Set time period"** (see p. 429). The default period of time is "per year".

Usually, you will have already specified the number of processes and the reference period of time in the Process start object of the model simulated. You can automatically read this data back by selecting the agent from the field **"Agent list"** with the right mouse button and then choosing the option "Load volume" from the popup menu which appears.

Alternatively, you may experiment with different numbers and periods of time to see in which way the changed entries affect the evaluation results.

> **Hint:** If you are displaying the results of an agent which evaluated a model of the WE library, the number entered applies equally to all business process models simulated.

Additional Display Options

The group "Additional display options" in the "Agents - result selection" window (see fig. 362) contains a number of additional options which refer to the output of the agent results.

> **Hint:** The availability of the elements described below depends on the simulation previously executed and on the currently selected agent. It is possible that the content of this group changes, when you select a different agent.

The following elements may be contained in the group "Additional display options":

- **"Show minimum and maximum"**

  If this option is activated, the minimum and maximum value for each result of the agent selected is listed at the end of the result table in two separate lines.

- **"Display sums"**

  If this option is activated, the sum over all reference objects for each result of the agent selected is listed in a separate line of the result table.

- **"Group Working Environment"**

  This option is only available if the agent selected has calculated results for objects from your working environment model. By activating the option "Grouping" these results are shown grouped (e.g. Personnel Costs for "Organizational units" instead of "Performer").

  After activating this option you define the class and the relation which define the grouping (e.g. "Organizational unit" and "Belongs to" in the fields **"Class**" and **"Relation"**).

  Note: The option "Group "Working Environment" is not available after Path analysis (see chap. 4.2, p. 343).

Info

The information window (see fig. 364) shows the information text on an agent and the names of all result functions which the agent has been instructed to calculate. Results, which are calculated but not displayed are not shown.

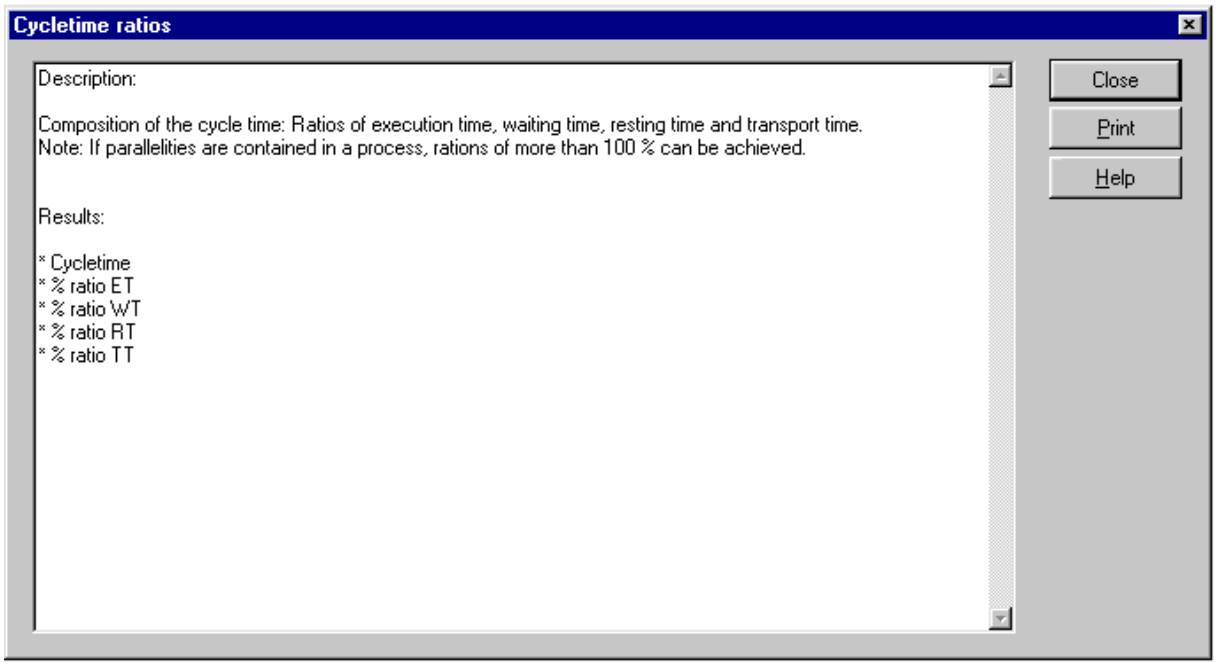


Figure 364: Information window for result function

The content of this window can be printed out.

Set time period

The reference period of time for agent results and for the process numbers on which the agent results are based can be entered in the window " Set time period" (see fig. 365).

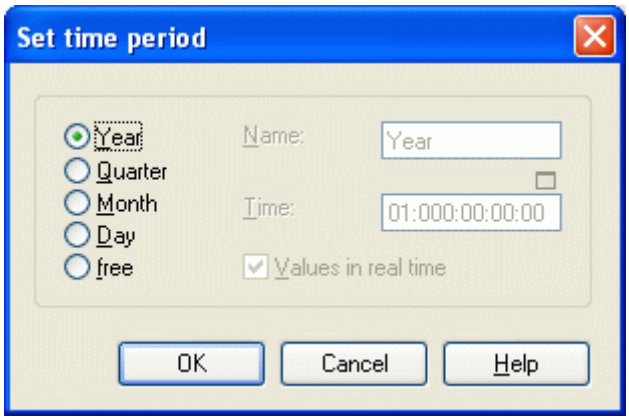


Figure 365: Set time period for result

Either select one of the pre-defined periods of time ("Year", "Quarter", "Month" or "Day") or define a new period of time according to your wishes ("free").

Select the option "free" to define a new period of time. Please use the ADOxx time format. This period of time will be interpreted as the fraction of a year (i.e. a "Month" would be 365 days / 12, that is 30 days and 10 hours).

Enter whatever name you would like for the period of time you are defining into the **"Name"** field.

> **Hint:** The period of time you defined will be saved until you either quit ADOxx or define another period of time.

Should the entry in the field "Period of Time" refer to a "normal" year of 365 days per year and 24 hours per day, the option **"Values in real time"** must be activated. If the time specification refers to the company time (see chap. 4.1.3.1, p. 335) currently set in ADOxx, this option must not be set. By default, this option is activated.

Example:

A company time of 170 days per year and 8 hours per day is defined and you wish to enter "Six months" as a period of time.

Enter the words "Six months" into the field "Name".

If you wish to specify the period of time **referring to one year in real time**, enter **00:182:12:00:00** (= 365 days of 24 hours each divided by 2) into the field "Time" and **activate** the option "Values in real time".

If you wish to specify the period of time **referring to one year in company time** enter **00:085:00:00:00** (= 170 days / 2) into the field "Time" and **deactivate** the option "Values in real time".

4.8.6.3 Represent agents results in tabular form

An agent's results are represented in the ADOxx browser (see chap. 8., p. 66). The browser's headline informs you which agent calculated these results, to which period of time these results relate and on which number of processes these results are based (see fig. 366).

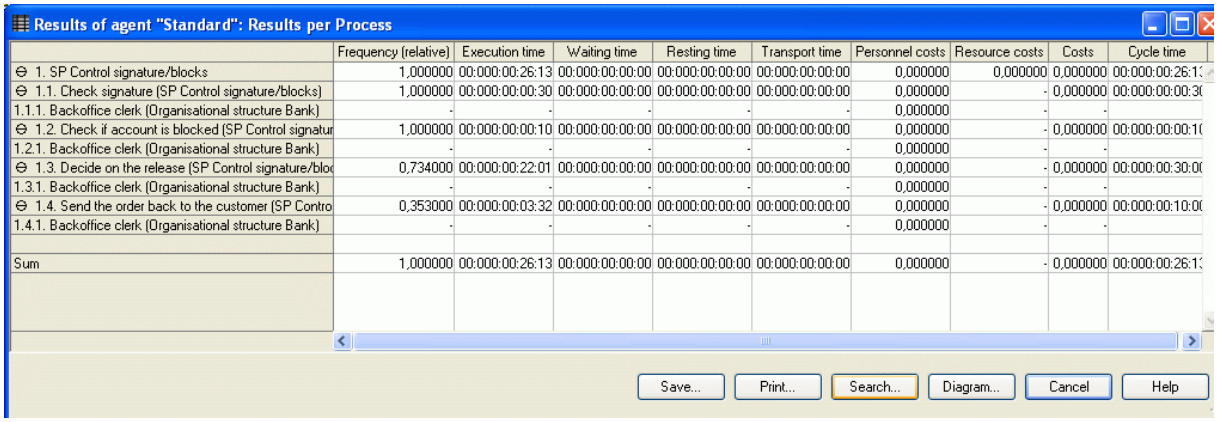


Figure 366: Represent agents results in tabular form

The contents of the ADOxx browser can be displayed graphically (see chap. 4.8.6.4, p. 430) by clicking on the button "Diagram".

4.8.6.4 Represent agents results graphically

In the window "Agent - Diagram" (see fig. 367) you must select the results and reference objects as well as the type of diagram for the graphic representation.

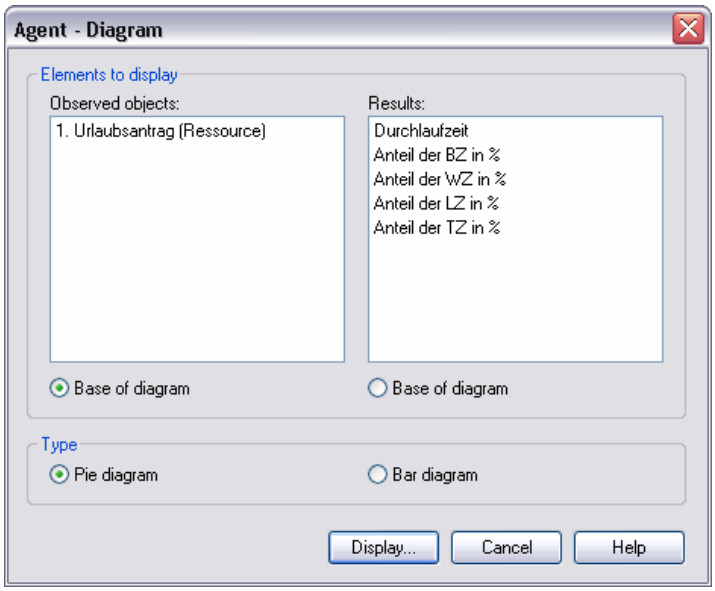


Figure 367: Represent agents results graphically (example)

Select one or more reference objects, for which you would like to create a diagram, in the field **"Observed objects"**. Select those results you want to be represented in the diagram in the field **"Results"**.

Define either the results selected or the reference objects selected as basis for the diagram to be displayed by selecting the particular option below one of the lists.

The diagram focuses on displaying those elements selected as the basis for the diagram.

Then select the type of diagram (**"Bar diagram"** (see chap. 8.16.1, p. 87) or **"Pie chart"** (see chap. 8.16.2, p. 89)) and click on the button "Display" to have the result displayed.

Hint: You can only select the diagram type "Pie chart", if the diagram's basis is exactly one element.

4.8.6.5 Represent result development as function graph

If an agent has traced one or more results in the course of the simulation, this result development will be represented as a function graph.

To display the result development (see chap. 4.8.5.1, p. 410) select the options "Result history" and "Graph" in the window "Agents - Result Selection" (see fig. 362) and then click on the button "Show". The "Agent - Result Development" (see fig. 368) window will be shown.

Hint: If you ran a large number of simulation runs, it will take quite some time to build the function graph.

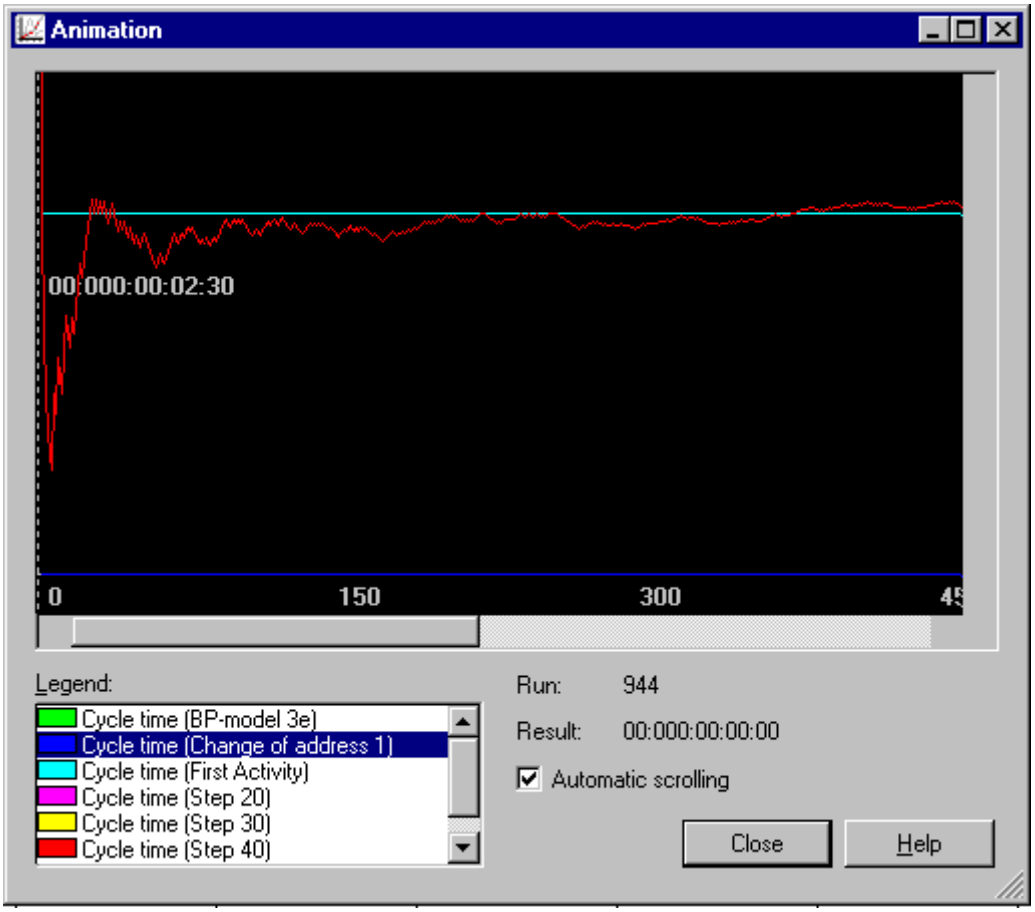


Figure 368: Function graph of the result development

The black area in the upper part of the window shows the different function graphs. Each function graph shows the development of a certain result in the course of the simulation. Should it have carried out a Workload analysis (see chap. 4.4, p. 367), each function graph shows the development of the result over the course of time.

The x-axis of the coordinate system shows each step of the simulation progress whereas the y-axis shows the corresponding result value.

The **Legend** contains all the function graphs pictured. If you select a graph here, it will be centred on the drawing area.

Hint: The maximum number of function graphs that can be represented simultaneously is 50 (referring to all windows opened). If this number is exceeded, you will be informed accordingly and no further function graphs will be constructed., In this case, we recommend that you close several windows containing function graphs and then try to display the graphs you could not see first again.

4.8.6.6 File format of the result development

If an agent had been instructed to trace the result development of a file, you may not only display the file content in the form of a function graph in ADOxx, but also process it further in other applications (e.g. to create Microsoft Excel diagrams from the result development).

The agents keep to the following format when generating a result development file:

Comment Lines always begin with the symbol '#'. The file starts with a file header, which specifies the result, the model and the reference objects monitored.

Each **line which contains data on the result development**, has the following format:

```
<ReferenceObject-ID>;
<Time>;
<ResultValue>;
<NameOfReferenceObject>
```

ReferenceObject-ID: An ID that is non-ambiguous for each reference object. If two reference objects have the same name, you can distinguish between them with the help of the ReferenceObject-ID.

Time: This element contains the simulation time, which refers to the particular line. Should you have carried out a Path analysis (see chap. 4.2, p. 343) or a Capacity analysis (see chap. 4.3, p. 354), the number of the corresponding simulation run is stored here. If you have carried out a workload analysis, the element contains the respective simulation time in seconds.

ResultValue: This element contains the results value that was current at the time indicated. If the result is a time (such as "Cycle time"), the result value is given in seconds.

NameOfReferenceObject: This element contains the name of the reference object, to which the line refers.

A resulting development file contains a consecutive sequence of such lines. The lines are listed in ascending order according to the entries in the column "Time" (the earliest first).

5. Evaluation

The availability of the "Evaluation" component and its modules "FlowMark audit trail" (see chap. 5.3, p. 441), Comparison of results (see chap. 5.1, p. 433) and "Evaluation queries" (see chap. 5.2, p. 438) depends on the current configuration (see chap. 4., p. 729) of the Modelling Toolkit.

The availability of additional components such as the "Process cost analysis (Cost-cutting component)" (see chap. 7.1, p. 497) with its modules "Actual analysis" and "Plan analysis" as well as "Human resource management" (see chap. 7.2, p. 513) depends on the acquisition of an additional licence.

The **"Evaluation" component** can be activated by clicking on the appropriate smart icon in the component list.

Alternatively, activate the Evaluation Component by right-clicking on the component list (beside the component smart icons) and then selecting the menu item "Evaluation" from the appearing popup menu. The popup menu can also be displayed using the function key <F9> and then the Evaluation Component is activated by pressing the key <v>.

Once the Evaluation Component has been activated the following icons should be displayed in the quick-access bar:

"FlowMark audit trail evaluation" (see chap. 5.3, p. 441)

"Comparison of results" (see chap. 5.1, p. 433)

"Predefined queries" (see chap. 5.2, p. 438)

"Models in the database" (see chap. 4.1, p. 56)

"Opened model windows" (see chap. 4.2, p. 57)

"Previous model window" (see chap. 2.3.1, p. 255)

"Next model window" (see chap. 2.3.1, p. 255)

If the Cost-cutting component is included in your configuration then the following extra smart icons will also be displayed in the quick-access bar:

"Actual-process cost analysis" (see chap. 5.4, p. 444)

"Plan-process cost analysis" (see chap. 5.4, p. 444)

The additional component "Human resource management" is to be called from the "Evaluation" menu.

> **Hint:** The appearance of the icons in the quick-access bar depends on the configuration of the quick-access bar (see chap. 4., p. 54) and the current configuration (see chap. 4., p. 729) of the Modelling Toolkit.

5.1 Comparison of results

You may compare the results of various ADOxx simulations (see chap. 4., p. 332) or analysis's (see chap. 3., p. 306) to each other.

Hint: In order to compare results, they must first have been saved in the format "Comparing Representation" (ACR (see chap. 11.2, p. 754) files).

It is possible to save results both from the Analysis Component (see chap. 3., p. 306) and the Simulation Component (see chap. 4., p. 332) (Path Analysis (see chap. 4.2, p. 343), Capacity Analysis (see chap. 4.3, p. 354) and Workload Analysis (see chap. 4.4, p. 367)) in ACR format. You can for example compare the personnel capacity demand of the as-is process to that of the should-be-process. The results can be displayed as a diagram or in tabular mode.

To compare the results, click on the smart-icon in the quick-access panel or select the option "Comparison of results" in the menu "Evaluation".

The window "Comparison of results - result selection" (see fig. 369) will appear on your screen, in which you should select the results to be compared (see chap. 5.1.1, p. 434).

5.1.1 result selection

Previously created results (see chap. 5.1.1.1, p. 435) can be selected in the window "Comparison of results - selection of results"" (see fig. 369).

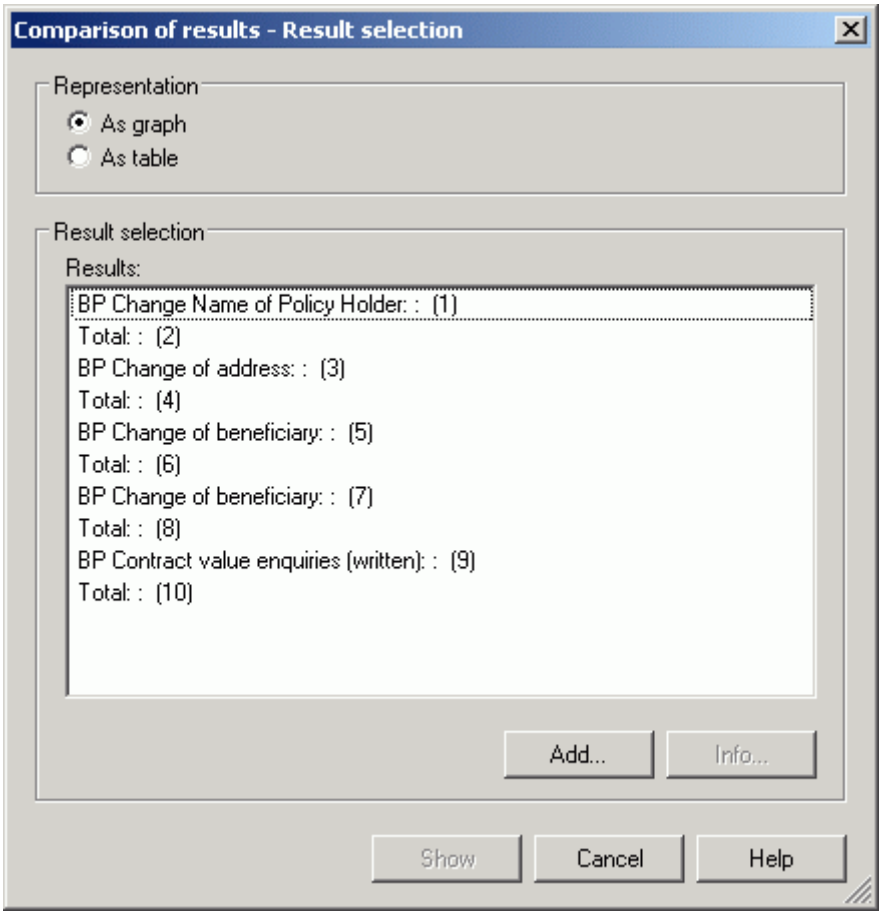


Figure 369: Selection of the results to compare

Click on the button **"Add"** to select simulation or analysis results (see chap. 5.1.1.1, p. 435) to be added to the "Results" list.

By clicking on the button **"Info"** when you have selected a specific result from the list, information regarding that result is displayed (see chap. 5.1.1.2, p. 436).

Select those results you wish to compare from the **"Results"** list. Two "Representation" options are available:

Graphical graphic representation (see chap. 5.1.2, p. 436) (i.e. bar or pie chart).

Tabular tabular representation in the ADOxx browser (see chap. 8., p. 66).

Click on the button **"Show"** after you have selected the results to be compared and how they are to be represented. The results to be selected will then be displayed either graphically (see chap. 5.1.2, p. 436) or in tabular (see chap. 5.1.3, p. 437) form.

5.1.1.1 Add result file

The window "Comparison of results - add file" (see fig. 370) shows information on the file previously selected. As well as date and time of creation, the following details are displayed:

- In the case of **Query Results**:
  - the description of the result to be compared (name of the object and name of the model which contains these objects),
  - the query's AQL expression (see chap. 14., p. 790),
  - the name of the models on which the query was executed.
- In the case of **Simulation Results**:
  - the description of the result to be compared (name of the business process model for the path analysis (see chap. 4.2, p. 343); name of the application model for the capacity analysis (see chap. 4.3, p. 354) or the workload analysis (see chap. 4.4, p. 367)),
  - the type of simulation algorithm (path analysis (see chap. 4.2, p. 343), capacity analysis (see chap. 4.2, p. 343) or workload analysis (see chap. 4.2, p. 343)),
  - the simulation parameters (number of simulation runs, input parameters and so on).

The results to be compared are numbered in ascending order in brackets as the description of the results may not necessarily be unique. This Id number does not change when new results are added.

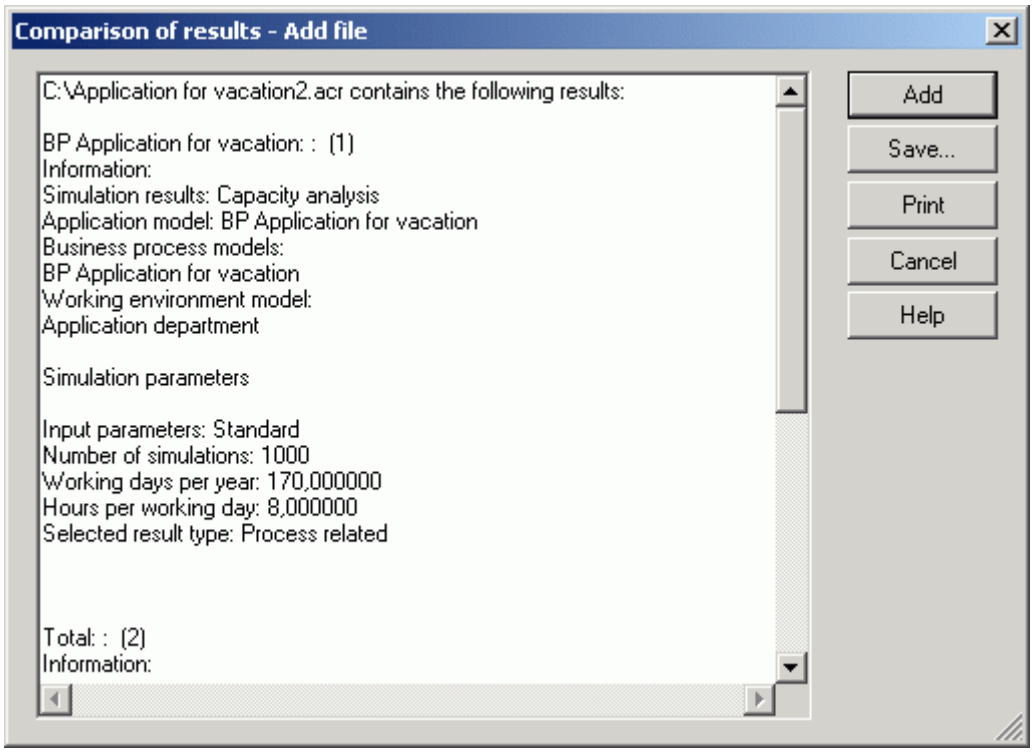


Figure 370: Add result file (Information about contents of the ACR file)

The information on each query or simulation which is held within the ACR files can be saved to a file or printed out for further evaluation. By clicking on the button **"Add"** you can add the results displayed to the "Results" list in the window "Comparison of results - result selection" (see fig. 369).

5.1.1.2 Information on the results

The window "Comparison of results - information" shows details of the result selected previously.

These details match those shown after a result file has been added (see chap. 5.1.1.1, p. 435).

5.1.2 Display results graphically

The settings for the graphical result representation are arranged in the "Result comparison - value selection" window (see fig. 371)

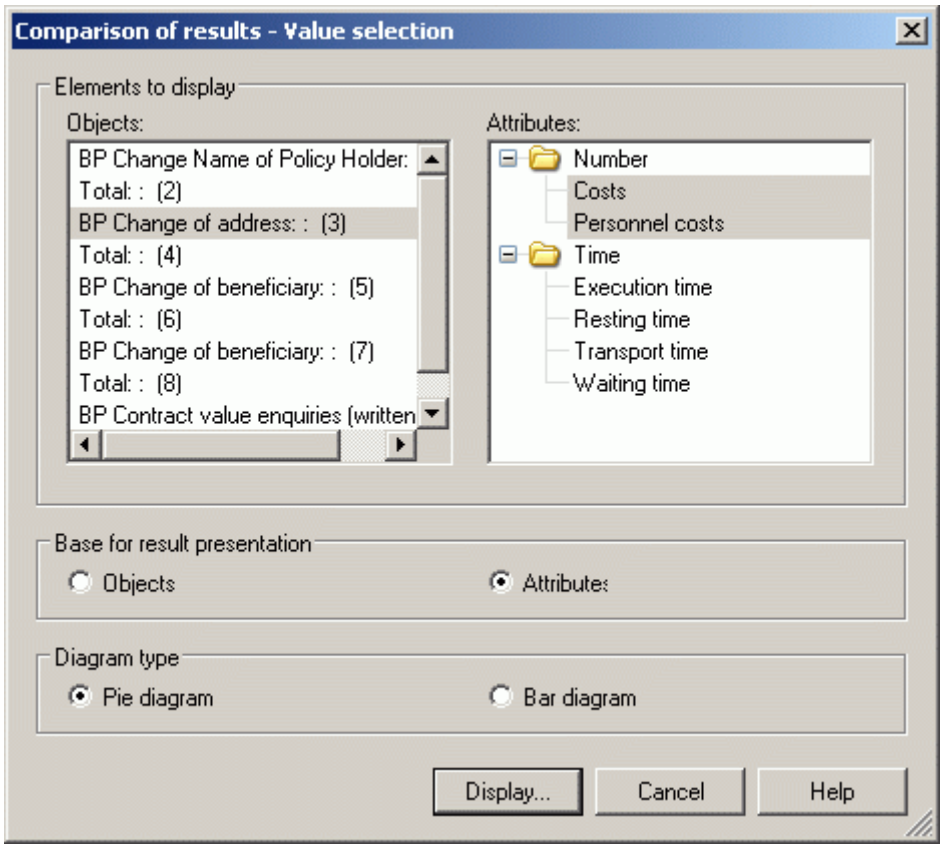


Figure 371: "Comparison of results - result selection

Hint: The graphical representation (see chap. 8.16, p. 85) of the comparison of results only shows the numerical values and times.

Select the objects and attributes you wish to display.

Hint: Either time or cost attributes are allowed in the attribute selection.

Define the **basis for the result display** by selecting either "objects" or "attributes" as basis. In the basis "objects" the selected attributes will be displayed according to their object. In the basis "attributes" the selected objects will be displayed according to their attributes.

The **type of diagram** defines the type of graphic representation. You can represent the values in a bar diagram or in a pie chart. In the bar diagram a group will be displayed per value, with one bar for one result. In the pie diagram the parts of the single results will be displayed for the selected values.

ATTENTION: The pie chart can only represent one single value.

Once you have selected your required settings, click on the **"Diagram"** button. The graphical representation will be shown (see chap. 8.16, p. 85) in the window "Comparison of results - graphical representation.

5.1.3 Tabular representation

If you set the representation type to **"As table"** and then press the **"Show"** button in the "Comparison of Results - Result selection" (see fig. 369) window, the ADOxx browser will be displayed.

The ADOxx browser (see chap. 8., p. 66) shows the selected results in columns. The rows contain all comparable values of these results.

Hint: Only those values which occur in all the results to be displayed are listed!

5.2 Evaluation queries

Various evaluation queries can be run. This depends on the definition of your application library.

ATTENTION: Evaluation queries will be defined in the application library and, if necessary, can be changed or expanded by the ADOxx administrator .

Hint: The pictures below refer to the evaluation queries of the **ADOxx standard application library (see chap. 16., p. 802)** A description of the queries in the application library is to be found in the user documentation volume III "ADOxx' standard method manual" or on the online document "ADOxx' standard method" ("Help" menu).

For the Evaluation queries, menu entries are added to the menu "Evaluation" of the Evaluation component. For every model type having at least one Evaluation query assigned, an extra menu entry is shown. On selecting one of these menu entries, a submenu cotaining all appropriate queries appears.

5.2.1 Model selection

If you want to run an evaluation query (see chap. 3.2.2, p. 319), select the option "Evaluation Queries on <*Model Type*>" in the menu "Evaluation" and then in the sub menu the pre-defined query type you want. Alternatively, you can click on the smart-icon in the quick-access panel and then proceed further as above.

The window "<*Pre-defined Evaluation Query type*> - Model selection" (see fig. 372) appears, showing all models within the ADOxx database to which you have access.

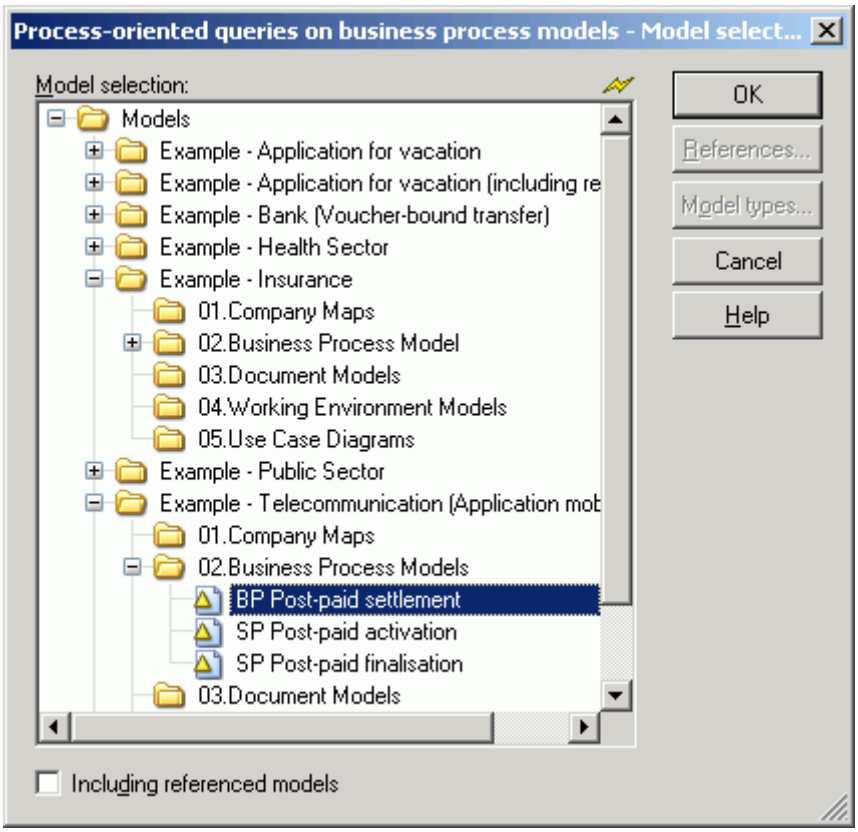


Figure 372: Evaluation queries - model selection

Select the model(s), on which you want to run the evaluation queries and then click on the "OK" button to select and carry out a query (see chap. 5.2.2, p. 439).

If you are running a query on Business Process models, then you can specify if referenced processes should be included automatically by activating the option **"Including referenced models"**. In this case, the models selected will be interpreted as main business processes and any referenced processes called will be indent in the result display according to their place in the "hierarchy".

Hint: When you activate the **"Including referenced models" option** you can change the reference settings (see chap. 11.1, p. 97) by clicking on the "references" button.

5.2.2 Run evaluation queries

The available queries will be displayed in the "<*Evaluation query type on model type*> - queries" window (see fig. 373).

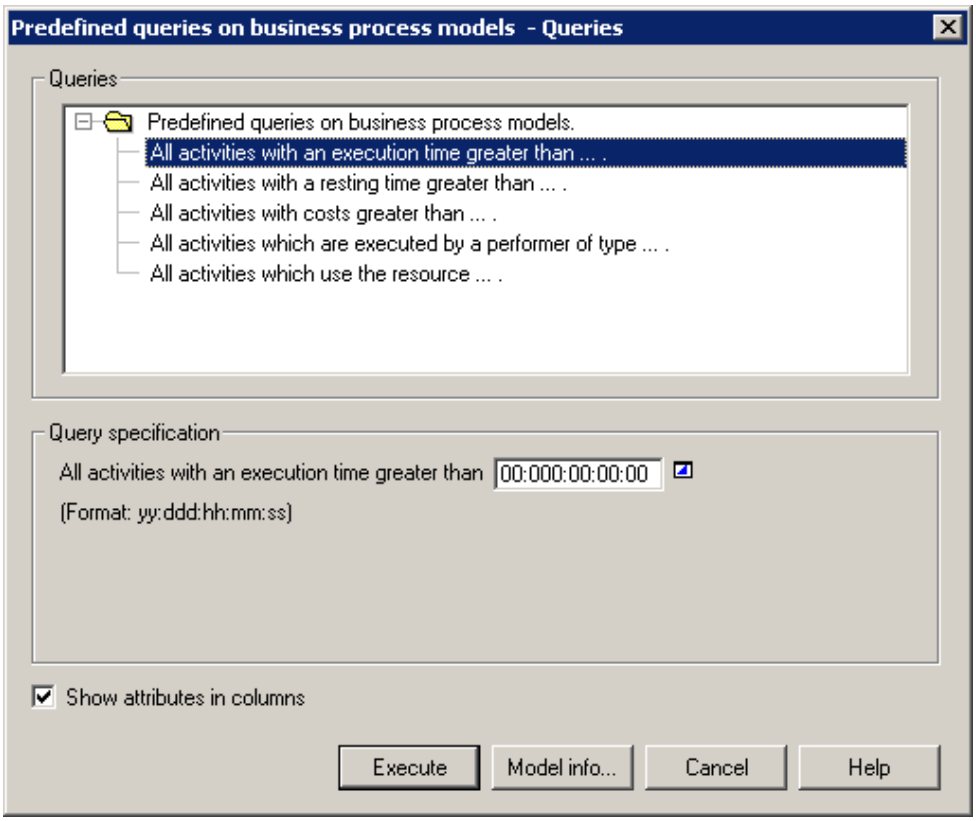


Figure 373: Run evaluation queries

Select one of the pre-defined evaluation queries from the **"Queries"** list. This will cause the "Query specification" section of the screen to be updated with a text and an input field. The text will describe the query and the input field should be updated by you as required. Now click on the "OK" button to evaluate the query selected.

Enter the necessary information (e.g. name of the object) or select in the class, attribute or comparison operator list the appropriate value and then click on the OK button to evaluate the selected evaluation query.

If you click on the "Model Info" (see chap. 5.2.2.1, p. 440) button, the name(s) of the model(s) on which you are running the query will be shown.

The results of the pre-defined evaluation queries (see chap. 5.2.3, p. 441) are displayed in the ADOxx browser (see chap. 8., p. 66).

5.2.2.1 Model information

If you click on the button "Model Info" in the window "<*Evaluation query type on model type*> - Queries" (see fig. 373), the models selected will be shown in the window "Evaluation queries - model info" (see fig. 374).

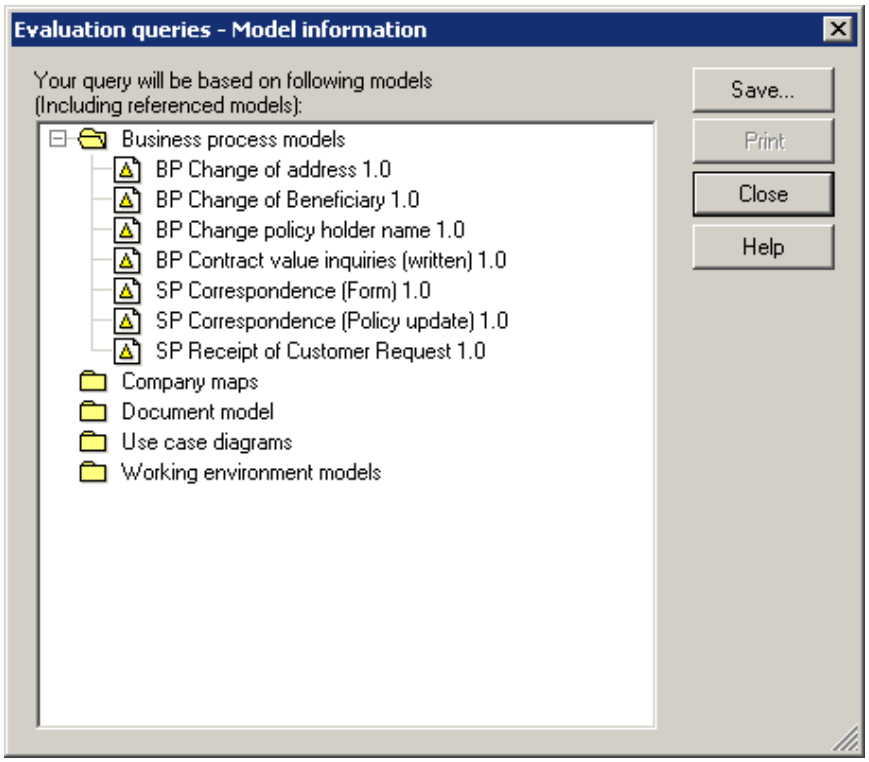


Figure 374: Show all selected (and referenced) models

The contents of this window may be printed out or saved to a file.

5.2.3 Results

The results of the evaluation queries are represented in the ADOxx browser (see chap. 8., p. 66) in the window "Query Results - <*Pre-defined Evaluation Query*>", as are the results of the queries/ /reports (see chap. 3.1.4, p. 313).

You may save the result in a file, print it out or display it as a diagram. It is also possible to search for text in the results.

Using the popup menu (accessed by clicking with right mouse-button) you may also sort (see chap. 3.1.4.2, p. 315) the objects and connectors displayed according to certain attribute values.

5.3 FlowMark Audit Trail Evaluation

The "FlowMark Audit Trail Evaluation" module enables you to evaluate processes and activities in a way similar to that of the Simulation Component (see chap. 4., p. 332) (cycle, waiting and execution times of processes and "Activity" objects). This equips you to recognise bottlenecks and other problem areas in a business process which has been transformed into **FlowMark** (Workflow Management system; producer: IBM). The results can be displayed in graphical or tabular form.

The FlowMark audit trails are generated by the FlowMark Workflow Management system.

Hint: The FlowMark documentation contains detailed descriptions of the formats of the FlowMark audit trails.

To begin evaluation of FlowMark audit trails select the option "FlowMark Audit Trail" in the menu "Evaluation" . Alternatively, you can click on the smart-icon in the quick-access panel.

The window "Evaluation: Open FlowMark audit trails" ((see fig. 375)) appears on your screen. Select the name (file extension: .DEL) and location of the FlowMark audit trail to be evaluated and click on the "OK" button. In the following window you will be asked to choose between the options **"Process related"** (see chap. 5.3.1, p. 442) for evaluating the activities and **"Person related"** (see chap. 5.3.2, p. 443) for evaluating the performers. Then click on the "OK" button to continue the evaluation.

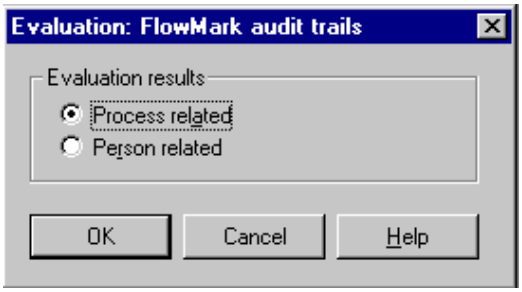


Figure 375: FlowMark Audit Trail Evaluation - select display type

Hint: The results are structured in a similar way to the simulation results after the capacity analysis (see chap. 4.3, p. 354) and after the workload analysis (see chap. 4.4, p. 367).

5.3.1 Tabular representation - process-related

The **process-related evaluation** calculates the execution and waiting times of activities and the cycle times of business processes which were executed in FlowMark. In addition, it evaluates which activities in one business process have been executed, how often and by which performers.

The evaluation results are displayed in tabular form in the ADOxx browser (see chap. 8., p. 66) (see fig. 376).

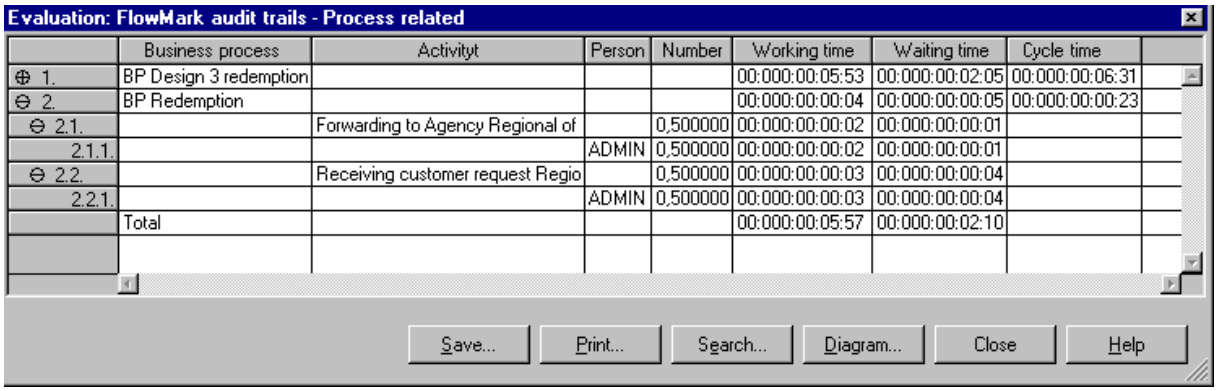


Figure 376: FlowMark audit trail evaluation - tabular display (process-related)

The waiting time of an Activity is the difference between the time at which the activity could have been executed (FlowMark audit trail event: "Activity ready") and the time at which it was actually started (FlowMark audit trail event: "Activity started").

The execution time of an activity (see fig. 377) is the difference between the time at which the Activity was started (FlowMark audit trail event: "Activity started") and the time at which the Activity was completed (FlowMark audit trail event: "Activity ended normally").

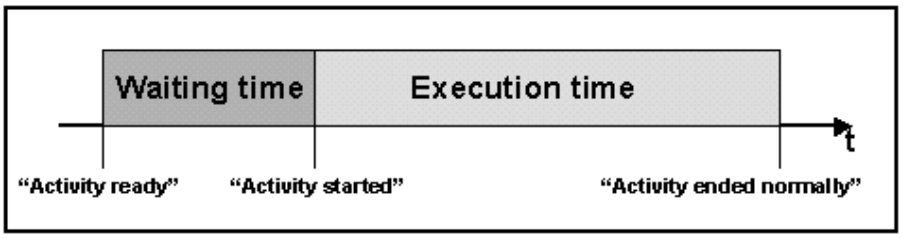


Figure 377: Definition of the execution and waiting time

The cycle time of a business process is the difference in time between the process start (Flowmark audit trail event: "Process started") and the process end (FlowMark audit trail event: "Process ended normally") (see fig. 378). The sum of the execution and waiting times of a business process may not match the cycle time of the FlowMark process, as a business process may be interrupted by the FlowMark administrator (FlowMark audit trail event: "Process suspended") and then later resumed (FlowMark audit trail event: "Process resumed").

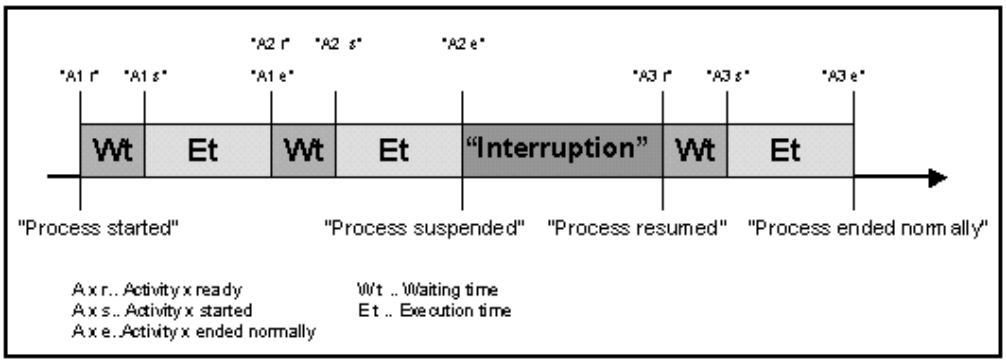


Figure 378: Definition of the cycle time of a process

5.3.2 Tabular representation - performer-related

The **person related evaluation** evaluates the performers who have executed activities in the business processes executed within FlowMark. It calculates the execution and waiting times of the activities executed by each performer and also informs you which performers executed which activity in a business process.

The results are displayed in tabular form in the ADOxx browser (see chap. 8., p. 66) (see fig. 379).

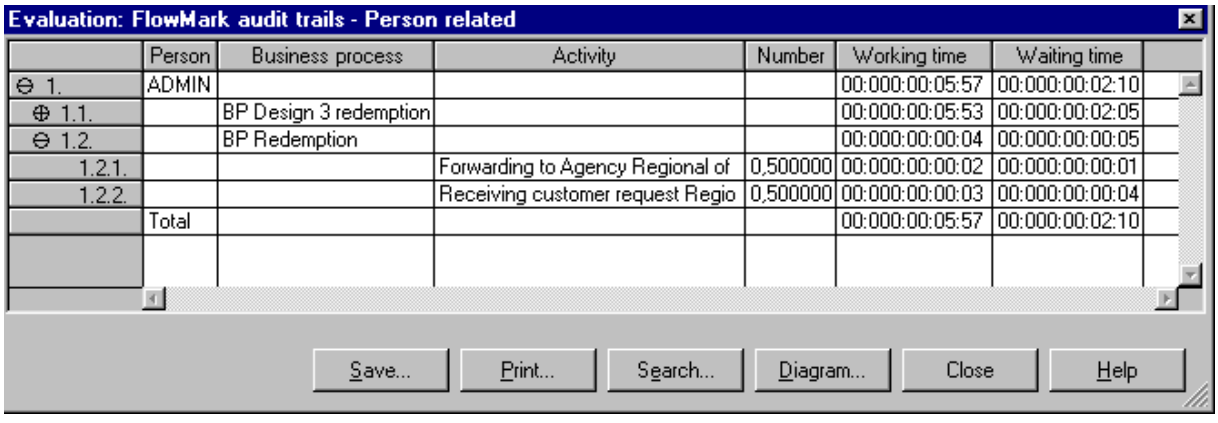


Figure 379: FlowMark audit trail evaluation - tabular display (performer-related)

5.4 Process cost analysis

The ADOxx Process Cost Analysis supports cost optimisation, especially in the area of overhead costs. The system is able to provide you with an evaluation concerning potential cost cutting measures.

The ADOxx process cost analysis simulates Business Process Models. This opens up further possibilities for the analysis and optimisation of process costs, as Decisions and Parallelitys can be parts of the control structure. There may be any number of process levels.

ATTENTION: The component "Process cost analysis" is neither included in ADOxx standard nor in ADOxx light. It can be purchased additionally from BOC Group.

Hint: A detailed description of the "Process costing component" (see chap. 5.4, p. 444) can be found in the chapter on additional functionality.

6. Import/Export

The availability of the "Import/Export" component and its modules "ADL Import" (see chap. 6.1, p. 445), "ADL Export" (see chap. 6.2, p. 468)", "XML Import" (see chap. 6.3, p. 473), "XML Export" (see chap. 6.4, p. 481) and "documentation" (see chap. 6.5, p. 486) depends on the current configuration (see chap. 4., p. 729) of the Modelling Toolkit.

The "Import/Export" component enables you to export to (see chap. 6.2, p. 468)ADL (see chap. 11.3, p. 755) or XML files the models and the model groups, the attribute profiles and the attribute profile groups as well as the application models, that are saved in the ADOxx' database. Also you can copy these files onto a floppy disk as an additional back up of your data or to transport your models to another computer (e.g. in a place, which is not connected to a network).

Via the "documentation" (see chap. 6.5, p. 486) you can generate from models saved in the ADOxx' database HTML or RTF files .

The **"Import/Export" component** can be activated by clicking on the appropriate smart icon within the quick-access bar.

It can also be activated by clicking with the right mouse button on the component list (near the component smart icons) and then selecting the menu item "Import/Export" from the popup menu shown. The popup menu can also be displayed by pressing the function key <F9> and then the Import/Export component can be selected by pressing the key<i>.

Once the Import/Export component has been activated the following smart icons should be displayed in the quick-access bar:

ADL Import (see chap. 6.1, p. 445)

ADL Export (see chap. 6.2, p. 468)

HTML Generation (see chap. 6.5.1, p. 486)

RTF Generation (see chap. 6.5.1, p. 486)

Options - documentation (see chap. 6.5.2, p. 488)

"Models in the database" (see chap. 4.1, p. 56)

"Opened model windows" (see chap. 4.2, p. 57)

"Previous model window" (see chap. 2.3.1, p. 255)

"Next model window" (see chap. 2.3.1, p. 255)

Hint: The appearance of the smart icons in the quick-access bar depend on the particular Configuration of the quick-access bar/panel (see chap. 4., p. 54) and the current configuration (see chap. 4., p. 729) of the Modelling Toolkit. The configuration of the quick-access bar can be changed by selecting the menu item "Quick-access bar" within the "Extras" menu.

6.1 ADL Import

The ADL import function allows the integration of data (models, attribute profiles, application models ...) in one ADOxx database. The data to be imported must be saved using the ADL Export (see chap. 6.2, p. 468) function prior to import.

ADL import is possible not only for **Models/Model groups and attribute profiles/attribute profile groups** (see chap. 6.1.1, p. 446) but also for **application models** (see chap. 6.1.5, p. 464).

6.1.1 Model Import

This function allows for the import of models, model groups, attribute profile and attribute profile groups. This process contains more steps:

- Start import
- Establish import settings and options
- Establish the aims of import and things to be imported
- Import

Start ADL Import of models/attribute profiles:

It is possible to start import in three ways:

- Click *shortly* on the smart-icon ,
- Click and *hold on* the smart-icon and choose item from the drop-down menu **"models/attribute profiles"**,
- Choose **"model"** from a menu item **"ADL Import"** sub menu item **"models/Attribute profiles"**.

Window "ADL Import - Settings" will be shown:

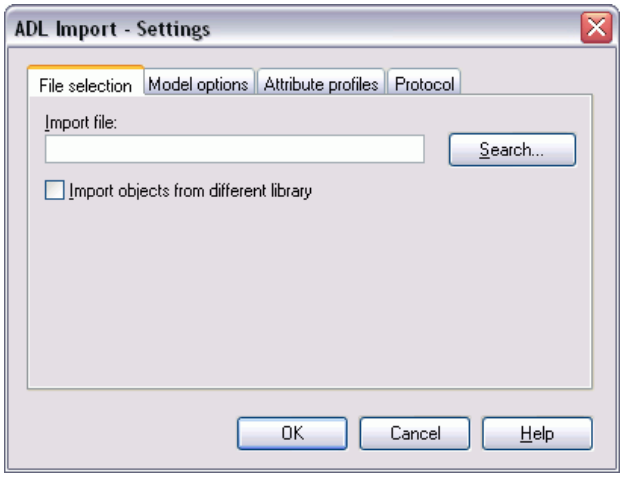


Figure 380: ADL-Import

This dialogue consists of four parts, which can be identified with tabs.

ADL Import - Settings

In the window "ADL Import - Settings" you can define settings concerning import in the tabs "File selection", "Model options", "Attribute profiles" and "Protocol".

> **Hint:** The tab "Versioning" is only available after an appropriate definition in the application library (Customising) .

Tab "File selection":

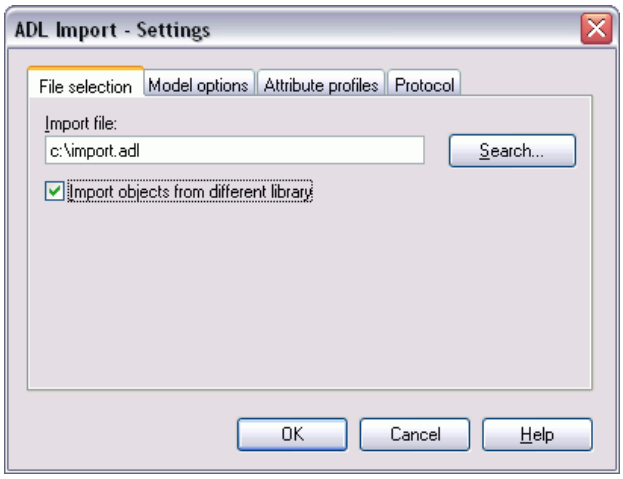


Figure 381: ADL Import - File selection

Enter the path and name of the ADL file with the model you wish to import into the **"File selection"** tab. The Button **"Search"** opens a window for the user to searching for the location of the file to be imported.

The option **"Import object from different library"** is available to import models which are based on a different application library than the one that is curently selected.

ATTENTION: The import of objects from different libraries is reasonable only if most of the classes and relations are the same in both libraries. This is because only the classes and relations which exist in the library assigned to the user (current library) are imported. Similarly, only attributes which appear in the user library (current library) will be imported. Attributes that are in the current library and are missing in the objects of models to be imported will be added and filled in with default values of the current library. Attributes which exist in objects of the models to be imported and are not available in the current library will not be imported.

Hint: If you do not choose this option, it will not be possible to import external models. An information window will appear and import will be cancelled. If you are ussure in which application library your models are based, choose this option to successfully import the models.

Tab "Model options":

Here you can find settings for the import of models or model groups:

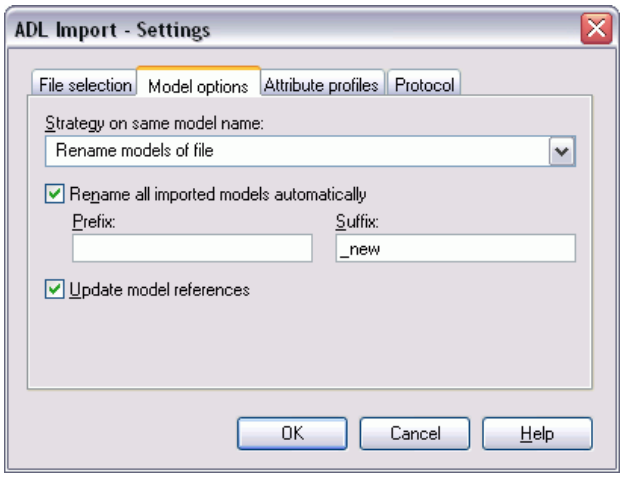


Figure 382: ADL Import - Model options

To import models, which have the *same name as the models stored in the ADOxx database*, you can select one of the following options from the list "Strategy on same model name":

- "Rename models of file" (see chap. 6.1.4.2, p. 455)
- "Paste into existing models" (see chap. 6.1.4.3, p. 455)
- "Overwrite existing models" (see chap. 6.1.4.4, p. 460)
- "Ignore models of file" (see chap. 6.1.4.5, p. 461)
- "Increase version number" (see chap. 6.1.4.6, p. 461)

The **option "Rename all imported models automatically"** assigns a prefix to the name of all imported models and/or a suffix. In the case that a name generated already exists in the database it will be processed according to the configuration of the option "Strategy on same model name".

> **Hint:** If you have chosen "Rename models of file" or the option "Rename all imported models automatically" and assigned a prefix/suffix, then it is recommended to also select the option "Update model references".

The **option "Update model references"** is available for all options and should always be selected, when the imported models are renamed. By updating the model references the references to imported models are automatically updated in case one of the imported models is being renamed.

> **Hint:** The strategy "Increase version number" as well as the additional action "Include version number in the model names" are available only for ADL Import to an application library with time-related versioning (see chap. 10.1.2, p. 92).

"Attribute profiles" tab:

In the "Attribute profiles" tab you can select one of the options for the import of attribute profiles:

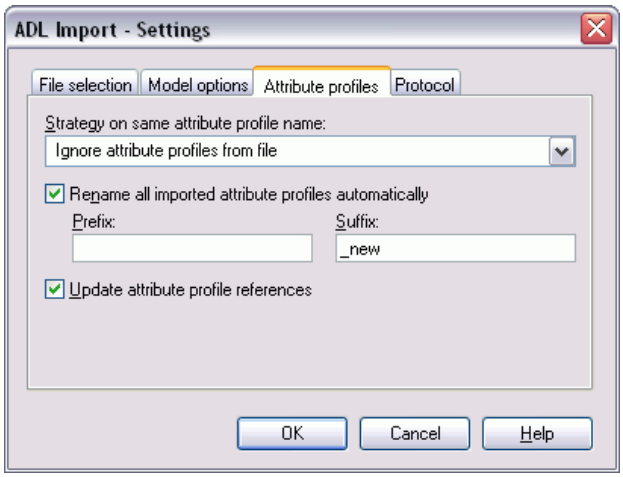


Figure 383: ADL Import - Attribute profiles

To import attribute profiles which have the *same name as the models stored in the ADOxx database*, you can select one of the following options from the list **"Strategy on same attribute profile"**:

- "Rename attribute profiles from ADL file" (see chap. 6.1.4.7, p. 462)
- "Paste into existing attribute profiles" (see chap. 6.1.4.8, p. 462)
- "Overwrite existing attribute profiles" (see chap. 6.1.4.9, p. 463)
- "Ignore attribute profiles from ADL file" (see chap. 6.1.4.10, p. 463)
- "Increase version number" (see chap. 6.1.4.6, p. 461)

The **option "Rename all imported attribute profiles automatically"** allows assigning a prefix to the name of all imported attribute profiles (it will be put in front of the attribute profile name) and/or a suffix (it will be put behind the attribute profile name). In the case that a name generated already exists in the database it will be processed according to the configuration of the option "Strategy on same attribute profile name".

> **Hint:** If you have chosen as a strategy "Rename attribute profile from the file" or the option "Rename all imported attribute profiles automatically" and assigned a prefix/suffix, then it is recommended to select the option "Update attribute profile references" as well.

The **option "Update attribute profile references"** is available for all attribute profile options and should always be selected when the imported attribute profiles are renamed. By updating the attribute profile references, the references to imported attribute profiles are automatically updated in case one of the imported attribute profile is being renamed.

Tab "Protocol":

It will enable you to generate a protocol for the ADL Import:

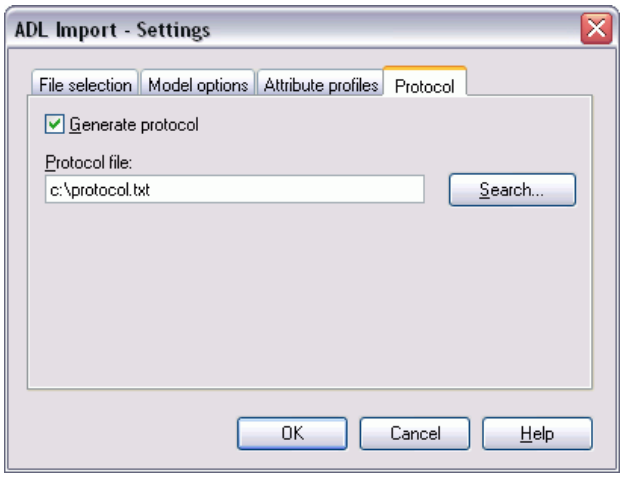


Figure 384: ADL Import - Protocol

Activate the option **"Generate protocol"** and enter into the **"Protocol file"** field the path and the name for the import protocol (TXT file). The button **"Search"** opens a window for the user to locate a file to be chosen.

After selecting all the options, click on **"OK"**, to start ADL Import of models or attribute profiles.

6.1.2 Selection

After selecting an ADL file and determining the required settings (see chap. 6.1.1, p. 446) the imported data is read; All the contained models (see fig. 385) and/or attribute profiles (see fig. 387) will be shown in the **"ADL Import - selection"** window.

Hint: The display of tabs "Models" and "Attribute profiles" depends on the input of the ADL file, i.e. only models in tab "Models", or only attribute profiles in tab "Attribute profiles" can be shown, exclusively one of two tabs can be available.

Tab "Models":

Models contained in an ADL file will be shown here:

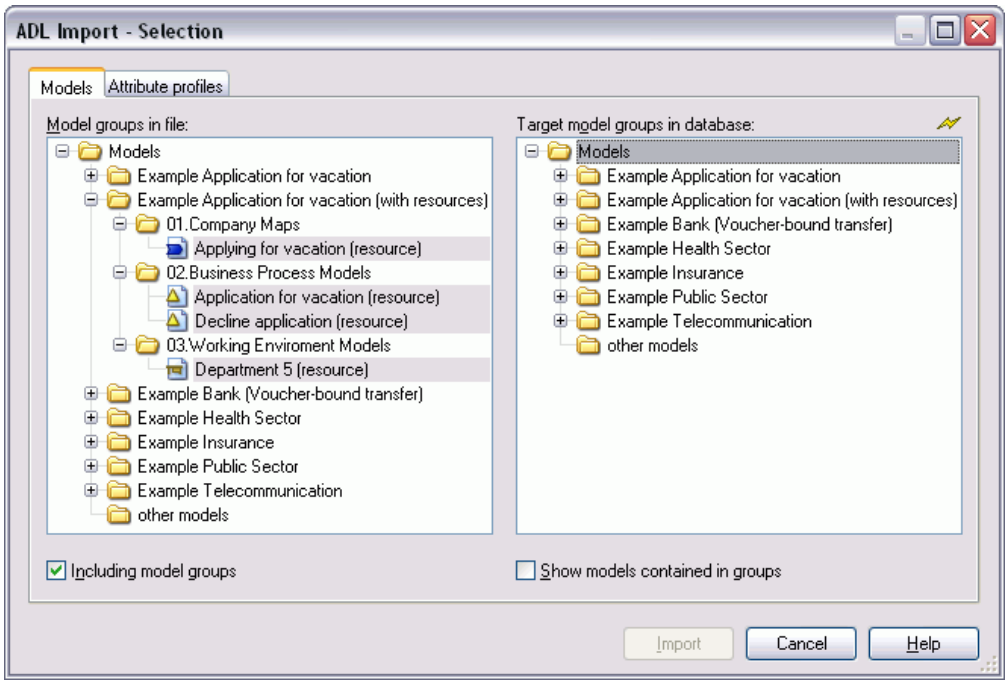


Figure 385: ADL Import - Model selection including model groups

If the option **"Including model groups"** is activated in the "Models" tab, the models contained in the ADL file will be displayed according to their model hierarchy (see fig. 385). When this option is selected the model group hierarchy will be imported as well as the selected models.

Hint: If no models are selected to be imported, only the model group hierarchy will be imported.

If you deactivate the option **"Including model groups"**, the models contained in the ADL files are displayed according to their model type. Only the selected models will be imported while using ADL Import:

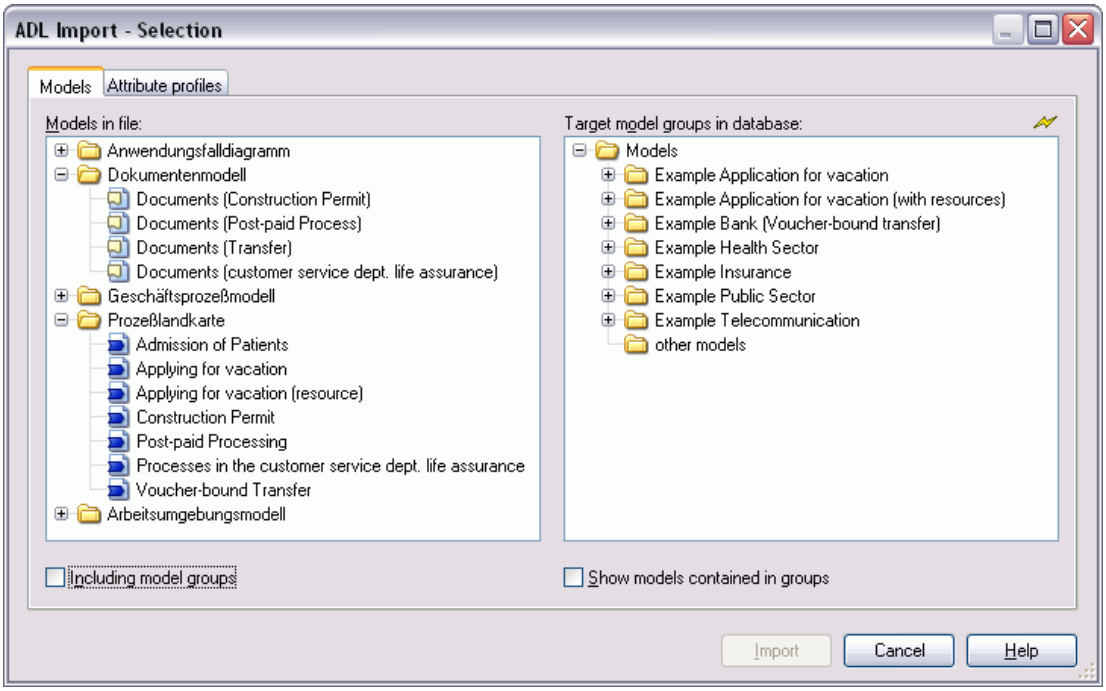


Figure 386: ADL Import - Model selection including model groups

Select the option **"Show models in groups"** to see lists of models contained in all model groups.

To choose imported models: Select all the models you want to import along with the model groups which you want to create references to the imported models.

Tab "Attribute profiles":

All the attribute profiles contained in the ADL file will be displayed here:

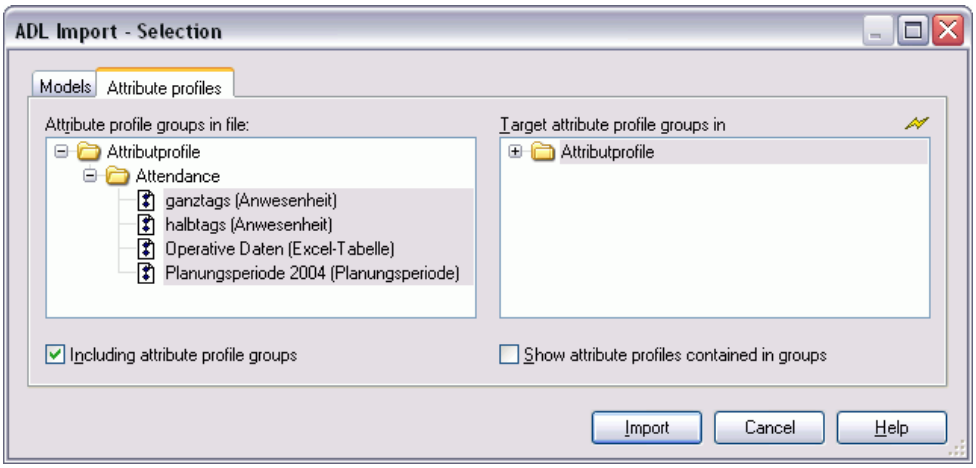


Figure 387: ADL Import - Attribute profile selection including attribute profile groups

If you *activate* the option **"Including attribute profile groups"**, the attribute profiles contained in the ADL file will be displayed according to their hierarchy (see fig. 387). When this option is selected the attribute profile group hierarchy will be imported as well as the selected attribute profiles.

Hint: If no attribute profiles are selected to be imported, only the attribute profile group hierarchy will be imported.

If you *deactivate* the option **"Including attribute profile groups"** in the tab "Attribute profiles", the attribute profiles contained in the ADL file will be displayed with no structure. Only the attribute profiles selected will be imported during the ADL Import:

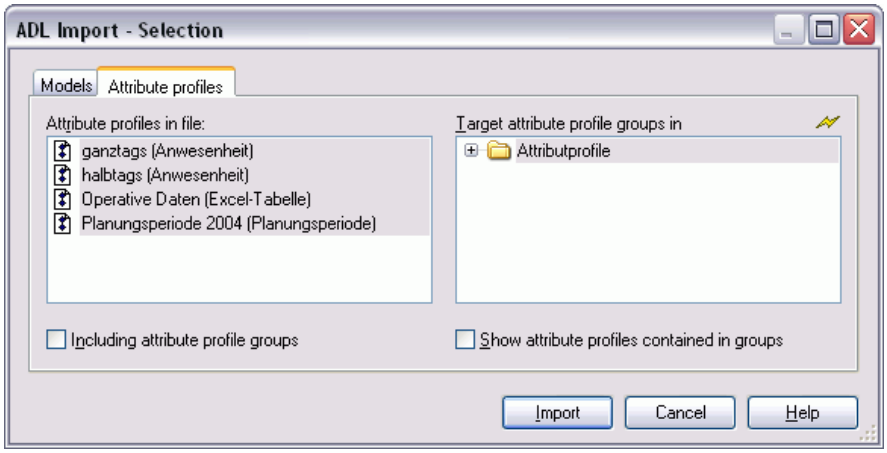


Figure 388: ADL Import - Attribute profile selection excluding attribute profile groups

When you select the option **"Show attribute profiles contained in groups"** the attribute profiles contained in the ADOxx Database will be displayed.

To choose import profiles: Select all the attribute profiles you want to import and the attribute profile groups in which you want to create a reference to the imported profile.

To import:

After choosing model and attribute profiles click on **"Import"**, to start the ADL Import.

At the beginning of the import, the ADOxx database is first checked for models or attributes which have the same name as the models or attributes to be imported.

> **Hint:** Only the name of the models or of the attribute profiles to be imported is compared to the models or attribute profiles stored in the ADOxx database, **not** their contents. Therefore, an imported model or attribute profile may have the same content as one stored in the ADOxx database, but under a different name. This means it is very important to select the appropriate options as listed in the window "ADL Import - Settings" (see fig. 382) for importing models (see chap. 6.1.1, p. 446) with identical names or in the window "ADL Import - Settings (see fig. 383)for importing attribute profiles (see chap. 6.1.1, p. 446) with identical names.

A status window keeps you informed about the current status of the import.

> **Hint:** During the import, if the ADL file contains a model of a type which is not defined in the current application library and the option "Import models from different library" is activated in the window "ADL Import - Settings" (see fig. 380), it is possible to assign a model type (see chap. 6.1.2.1, p. 453) defined in the application library to this model .

6.1.2.1 Assign model type

If a model's type as specified in the ADL file being imported does not exist in the current application library, then the window "ADL Import - Assign model type" is displayed:

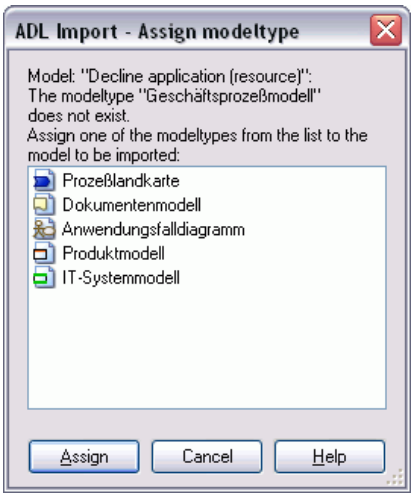


Figure 389: ADL Import - Assign model type

Select one of the suggested model types and assign it to the model being imported by clicking on the **"Assign"** button. The ADL Import will continue.

6.1.3 Result

After the ADL Import has been successfully completed, the window "ADL Import - Results" will inform you about the conclusion of the import. This window lists the names under which the imported models are stored in the ADOxx database:

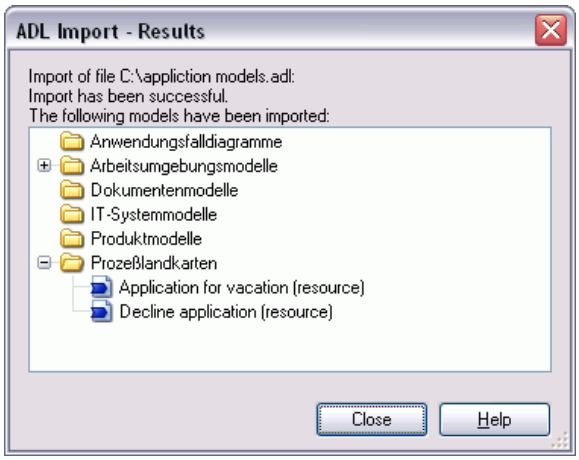


Figure 390: ADL Import - Result display (with protocol)

If in the window "ADL- Import - Settings" - Tab "Protocol" (see fig. 381) the option "Generate protocol" is active and a path and the name for the protocol file to be created is provided, then the button "Protocol" will be shown in the "ADL Import - Results" (see fig. 390) window. By clicking on this button the generated protocol will be shown in a text editor.

Hint: If you have imported exclusively model groups, the successful ADL Import will be indicated through an appropriate message window.

6.1.4 Options and examples

Hint: This chapter lists all ADL and XML import options as well as examples. Please note that not each option is available for both functions.

6.1.4.1 General options for ADL import

Activate in the tab "model options" the option **"Rename all imported models automatically"** and enter a prefix (field **"Prefix"**) and/or a suffix (field **"Suffix"**) that will be added to the model name during import.

Activate in the tab "model options" the option **"Update model references"** to ensure that all references within the model you wish to import will be updated according to the change of name.

Option when using an application library with time-related versioning (see chap. 10.1.2, p. 92):

Activate in the tab "model options" the option **"Take over version numbers of model names"** to inherit model-oriented version numbers contained in the model names (see chap. 10.1.1, p. 92) which now would have time-related versioning. As a result each imported models will gain its own version order.

6.1.4.2 Rename models from ADL file

The model to be imported is renamed, if a model of the same name and model type is already stored in the ADOxx database.

Hint: Activate in the window "ADL Import - Settings" the option **"Update model references"** to ensure that all references within the model you wish to import will be updated according to the change of name.

Hint: Activate in the window "ADL Import - Settings" the option **"Take over version number in the model name"** to take over the model-related version number (see chap. 10.1.1, p. 92) of a model to the name of a model in a time-related version. This way each imported model will have it's own version thread created.

The option "Take over the version number in the model name" is available only for ADL Import to an application library with time-related versioning (see chap. 10.1.2, p. 92) .

When the name and model type are the same as of a model in the database, a window will open with the question of whether a new name should be assigned.

Hint: If the model with the same name stored in the database is not shown, it means that it is not assigned to a model group or it is referenced to a model group to which the user has neither read nor write access.

If you click on **"Yes"**, a window **"Rename model"** will be displayed:

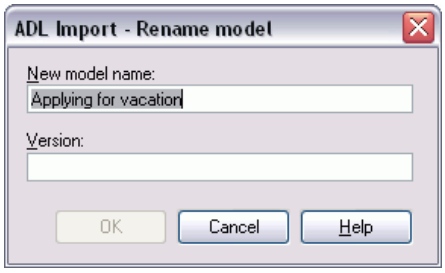


Figure 391: Import - Rename model

Enter a new name for the model to be imported and click on "OK" or press the Enter key to start importing.

Hint: Using the **"Cancel"** option in the window "Rename model" causes the import to abort as in the ADOxx database, only models with unique names can be stored.

6.1.4.3 To paste into existing models

Models to be imported will be aggregated in the ADOxx database with all models which have already been stored there. In case of objects with the same names and connectors, attribute values of the imported objects or connectors will be taken over. Other objects and connectors of the imported models will be newly created.

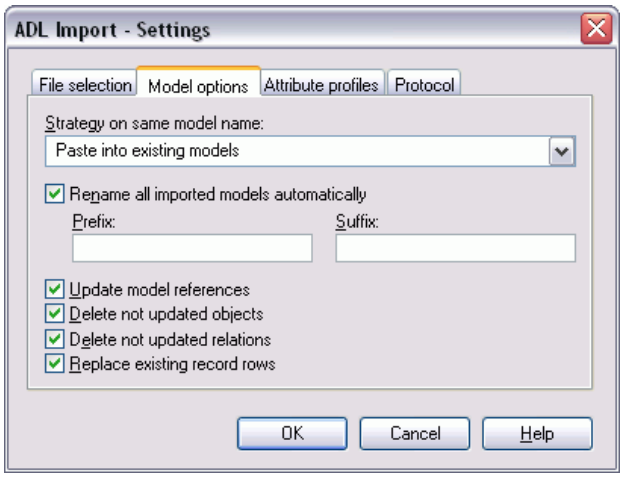


Figure 392: ADL import settings - extended model options

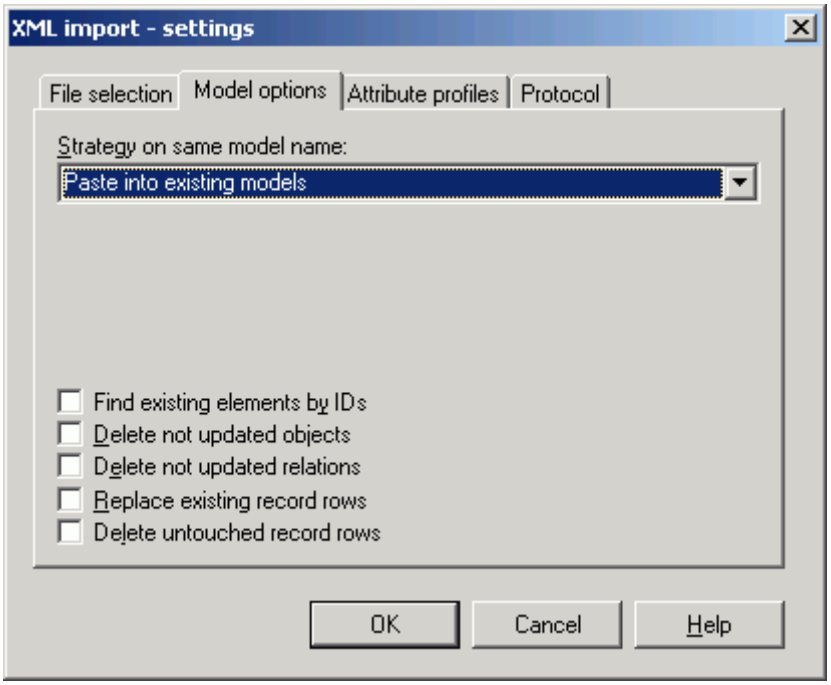


Figure 393: XML import settings - extended model options

The XML import option **"Find existing elements by their ID numbers"** allows to compare objects not by using their names but through their internal ID numbers. Thanks to this option, it is also possible to assign correctly renamed objects. If the names of an object differ from one another in a file to be imported and in a database, the name from the database is valid.

When the option **"replace existing record rows"** is *active*, the existing rows in record attributes will be cancelled and replaced with rows from the models to be imported. If the option is *deactivated*, the rows of record attributes in the attribute profiles to be imported will be appended to the lines in the existing attribute profiles. The effects are shown in an example (see p. 457) .

If the XML import option **"delete untouched record rows"** is *active*, all rows of the record attributes that remain unaffected during import will be deleted. If the option is *deactivated*, they remain unchanged.

If you *activate* the options **"Delete not updated objects"** or the **"Delete not updated relations"**, all objects/connectors in the existing model will be deleted, if the objects/connectors with matching names are not contained in the model to be imported. The effects are illustrated in an example

(see p. 459). Objects, which are stored in the ADOxx database and are not contained in the model to be imported, are left unchanged in the model, if the "Delete not updated objects" option is *deactivated*. A connector and its attribute values are transferred, if the objects and their links have matching names and types in the model stored as well as in the model to be imported.

If there are objects or connectors with matching names in both the model to be imported and the existing model, the window **"ADL Import - replace values"** will be displayed for each object (see fig. 394) or for each connector (see fig. 395).

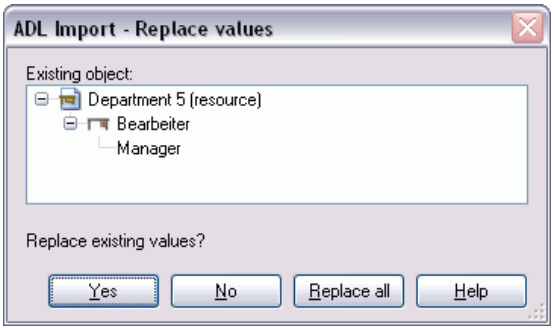


Figure 394: ADL Import - Replace value of an object

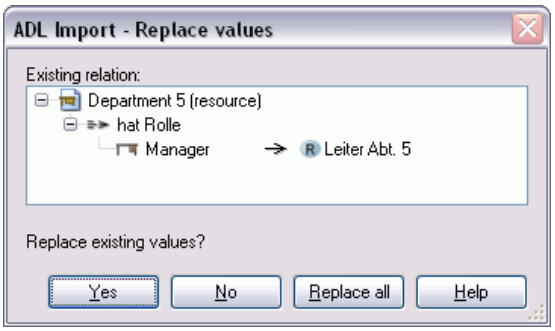


Figure 395: ADL Import - Replace value of a connector

Click **"Yes"** if you want to replace the existing attribute values with the values of the object or connector in the import model. By clicking **"Replace All"**, all attribute values of objects and connectors with matching names will be replaced by the values of the imported model. Clicking **"No"** means that the existing attributes of the object being queried will not be changed.

Example of the option "Replace existing record rows"

The effects of the option "Replace existing record rows" is shown in the picture below (see fig. 396).

Assumption:

A model "Example model 1" of the type "BP model" stored in the database contains the object "Activity 1" with an attribute of type record. The values of the attributes are shown in the picture below (see fig. 396).

	Attribute1	Attribute2
1	10.000000	Value1
2	20.000000	Value2
3	30.000000	Value3
4	40.000000	Value4

Figure 396: Attribute value of the model saved in the database

A model "Example model 1" of the type "BP model" is stored in an ADL file. This model contains the object "Activity 1" with an attribute of type Record. The values of this attribute (see fig. 397) are different from those stored in the database.

	Attribute1	Attribute2
1	100.000000	ValueA
2	200.000000	ValueB

Figure 397: Attribute value of the model contained in the file

Results:

When the option **"Replace existing record rows" is activated**, the entries in the table in the model stored in the database will be deleted and the values will be transferred from the model to be imported. The result of the ADL Import therefore equals the values of the attribute from the model in the ADL file (see fig. 397).

When the option **"Replace existing record rows" is deactivated**, the entries remain in the record in the model stored in the database and the values from the model to be imported will be appended. The result of the ADL Import is shown below:

	Attribute1	Attribute2
1	10.000000	Value1
2	20.000000	Value2
3	30.000000	Value3
4	40.000000	Value4
5	100.000000	ValueA
6	200.000000	ValueB

Figure 398: Results of the deactivated option "Replace existing record rows"

ATTENTION: Record attributes can be limited to a certain number of rows (entries). Should the maximum number of entries be exceeded while having the option "Replace existing record rows" disabled, the import will be aborted.

Example of the option "Delete not updated objects/connectors"

The effects of the option "Delete not-updated objects/connectors" are illustrated in the following example.

Assumption:

A model "Example model 1" of type "WE model" stored in the database contains the objects displayed in the picture below (see fig. 399).

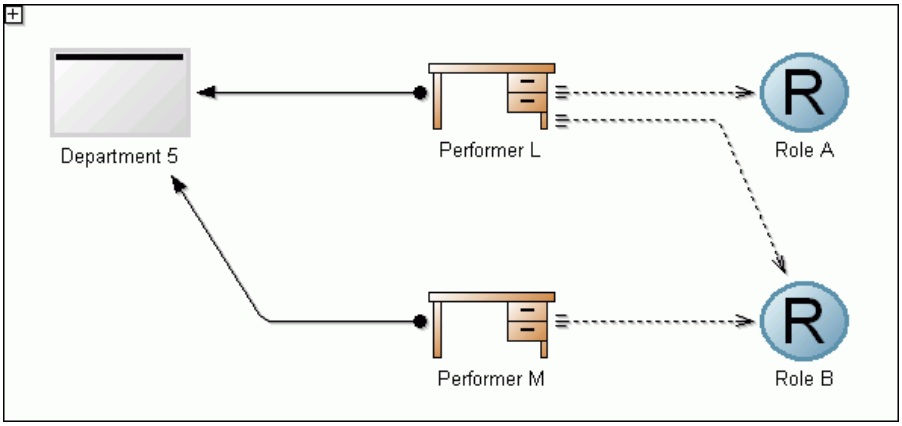


Figure 399: Example model in the ADOxx database

Variant 1: A model "Example model 1" of the type "WE model" is stored in the ADL file, which is different from the model stored in the database, since it does not contain the object "Role A" (see fig. 400).

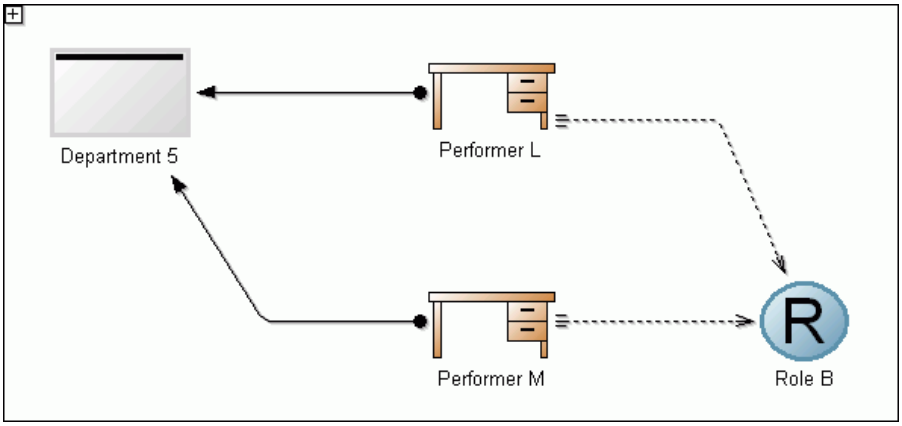


Figure 400: Example model in the ADL file (Variant 1)

Variant 2: A model "Example model 1" of the type "WE model" is stored in the ADL file, which is different from the model stored in the database, since it does not contain the connector from the object "Performer L" to the object "Role B" (see fig. 401).

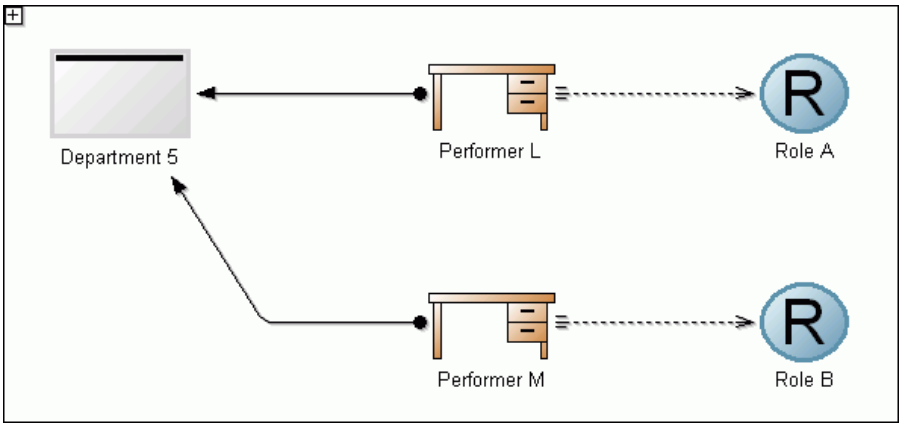


Figure 401: Example model in the ADL file (Variant 2)

Results:

When the option **"Delete not updated objects" is activated**, the result of the **Variant 1** equals the contents of the model stored in the file (see fig. 400) since the object "Role A" is not contained in the model to be imported.

When the option **"Delete not updated connectors" is activated**, the result of the **Variant 1** is as shown below (see fig. 402), since the connector from the object "Performer L" to the object "Role B" is not contained in the model to be imported.

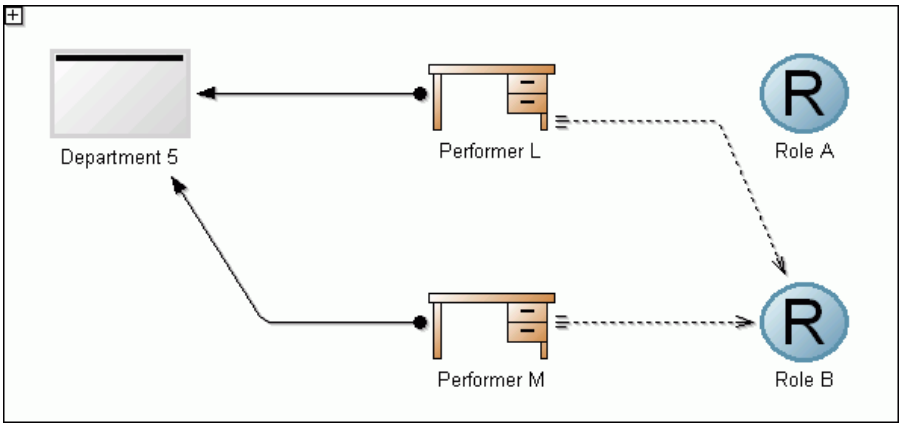


Figure 402: Import results by activated option "Delete non updated connectors"

When the **option "Delete not updated objects" is activated**, the result of the **Variant 2** equals the contents of the model stored in the database (see fig. 399).

When the option **"Delete not updated connectors" is activated**, the result of the **Variant 2** equals the contents of the model stored in the file (see fig. 401), since the connector from the object "Performer L" to the object "Role B" is not contained in the model to be imported.

6.1.4.4 Overwrite existing models

An existing model which is stored in the ADOxx database is overwritten by the model to be imported.

> **Hint:** In addition the general options for import (see chap. 6.1.4.1, p. 454) **"Update model references"** and - when using time-related versioning - **"Take over version numbers of model names"** are available.

ATTENTION: All application models (see chap. 2.4.16, p. 284) that contains one of the imported models will be deleted during the ADL import in the overwriting mode. Inthis case a saftey query enables you to cancel the import.

Hint: Activate in the window "ADL Import - options" the option **"Update model references"** to ensure that all references within the model you wish to import will be updated according to the change of name during the import.

Hint: Activate in the window "ADL Import - options" the **"Take over version number in the model name"** to take over the model-related version number (see chap. 10.1.1, p. 92) of a model to the name of a model in a time-related version. This way each imported model will have it's own version thread created.

The option "Take over the version number to the model name" is available only for ADL Import to an application library with time-related versioning (see chap. 10.1.2, p. 92) .

6.1.4.5 Ignore Models

If you select this option, any models to be imported from the ADL file which have matching names with existing models will in fact not be imported. Should an ADL file contain **only** models with matching names, an appropriate window will be shown:

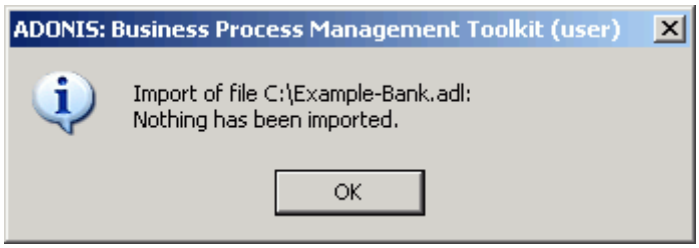


Figure 403: Import - all models were ignored

Click **"OK"** or hit the "Enter" key to close the window.

Hint: In addition the general options for import (see chap. 6.1.4.1, p. 454) **"Update model references"** and - when using time-related versioning - **"Take over version numbers of model names"** are available.

6.1.4.6 To increase the version number

Hint: The option "To increase version number" is only available in application libraries which have time-related versioning (see chap. 10.1.2, p. 92).

Models which should be imported and have the same names as the ones already stored in the database will be saved there under a new model version. The version number will then be assigned automatically i.e. such a model gets the next available validity date.

Hint: The version number of a model to be imported is not valid if it is based on an ADOxx-1.0 application library with model-related versioning (see chap. 10.1.1, p. 92) or when it comes from an ADOxx Version lower than 1.0 or when the version number does not correspond to the syntax definition in the current application library.

Hint: When importing models coming from a model-related application library into the time-related one or when importing models with invalid version numbers from a time-related library (i.e. with syntax different from the used one) the user has to assign version number to every imported model (see chap. 6.1.4.11, p. 464).

Hint: If a model with the same name and type as the imported one has already been stored in a database, the option **"Increase version number"** will be used and the model will be automatically assigned the next available version number.

For models with matching names and model type and different invalid version numbers, the version numbers will be given increasingly (i.e. "January 2000", "February 2000"). The initial value will be assigned to the model with the oldest date of creation.

6.1.4.7 Rename attribute profiles from ADL file

The attribute profile to be imported will be renamed, if an attribute profile with the same name and from the same attribute profile class is already stored in the ADOxx database. In the case of matching names and classes, a window will be displayed asking if a new name should be assigned.

If you click **"Yes"**, the window **"ADL Import - Rename attribute profile"** appears:

Figure 404: Import - Rename Attribute profile

Hint: Clicking on the **"No"** button in the query dialogue will abort the import, since only attribute profiles with unique names can be stored in an ADOxx database.

Enter a new name for the attribute profile to be imported and click on the **"OK"** button or press the enter key to continue the ADL Import.

Hint: Clicking on the **"Cancel"** button in the window "ADL Import - Rename attribute profile" causes the abort of the import, since only attribute profiles with unique names can be stored in an ADOxx database.

6.1.4.8 Paste into existing attribute profiles

Attribute profiles to be imported will be aggregated in the ADOxx database with all attribute profiles from the same attribute profile class, which have already been stored there. In doing so, you can use the additional option "To replace existing record rows":

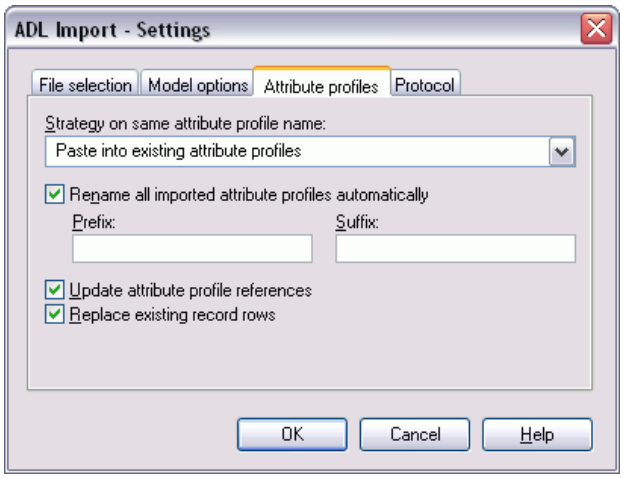


Figure 405: ADL Import - Extended Attribute profile options

Hint: If the option **"Replace existing record rows"** is *active*, the record rows of record attributes in the existing attribute profiles will be deleted and replaced by the rows in the attribute profiles to be imported. If the option is *deactivated*, the rows of record attributes in the attribute profiles to be imported will be appended to the lines in the existing attribute profiles . The effects are shown in an example (see p. 457).

In case of matching names and attribute profiles in the file and in the ADOxx database, the window "ADL Import - Replace values" appears for each of these attribute profiles.

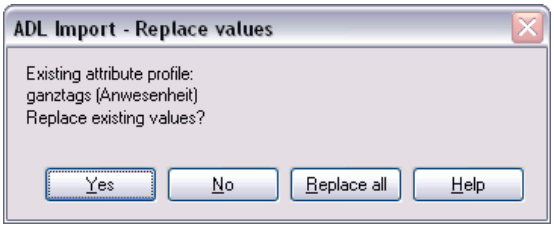


Figure 406: To Import attribute profiles - replace values

Click on the **"Yes"** button, if you want to replace the existing attribute value of the attribute profile by the value of the ADL profile in the ADL file.

If you click on the button **"Replace all"**, all attribute values will be replaced by the value of the imported attribute profile.

Clicking on the **"No"** button causes no changes to the attribute values of the existing attribute profile.

6.1.4.9 Overwrite existing attribute profiles

The attribute profile to import will be pasted into an attribute profile of the same class stored in the ADOxx database. While doing so, the attribute values will be taken over from the attribute profile to the file, if they differ from the values stored in the ADOxx database.

The attributes of these attribute profiles, which are stored in the ADOxx database and not contained in the model to be imported, will be set back to the standard value.

6.1.4.10 Ignore attribute profiles from ADL file

When selecting this option, attribute profiles with matching names will be ignored, i.e. not imported. Should there be **exclusively** attribute profiles with matching names in an ADL file, then an appropriate window will be displayed:

Figure 407: ADL Import - All attribute profiles have been ignored

Close the window by clicking on the **"OK"** button or pressing the Enter key.

6.1.4.11 Assign version number

Hint: It is possible to assign a version number only to models of an application library, in which a time-related versioning (see chap. 10.1.2, p. 92) has been defined.

During the ADL Import of models, the window **"ADL Import - assign version number"** will be displayed with the defined initial value for each model with an invalid version number (i.e. not corresponding to the syntax or not available):

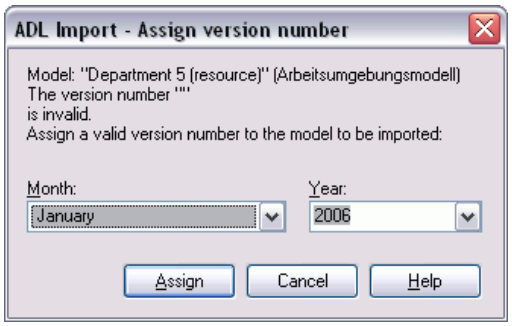


Figure 408: ADL Import - Assign version number

Select a version number and click on the button **"Assign"**.

Hint: Clicking on the "Cancel" button in the window "ADL Import - Assign version number" will abort the import, since models, which are based on the current application library, are only authorised with a valid version number.

6.1.5 Import application models

This function allows the user to import models, this process contains the following steps:

- Start import
- Establish import settings and options
- Establish the aims of import and items to be imported
- Import

To start ADL Import of application models:

It is possible to import in two ways:

- Click and *hold on* the smart-icon and choose from the drop-down menu. You can see the menu item **"Application models"**,
- Choose from the menu **"Model"** menu item **"ADL Import"** and then its sub menu item **"Application models"**.

Window "Application models import - Settings" will be shown:

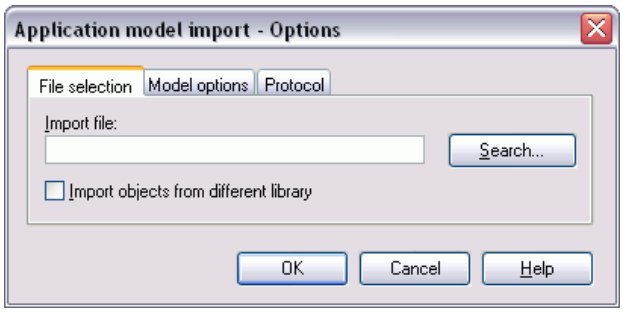


Figure 409: ADL Import of application models

This dialogue consists of three parts, which can be identified with tabs.

ADL Import - Settings

In the window "ADL import - Settings" you can define the settings concerning the import in the tabs "File selection", "Model options", and "Protocol".

Tab "File selection":

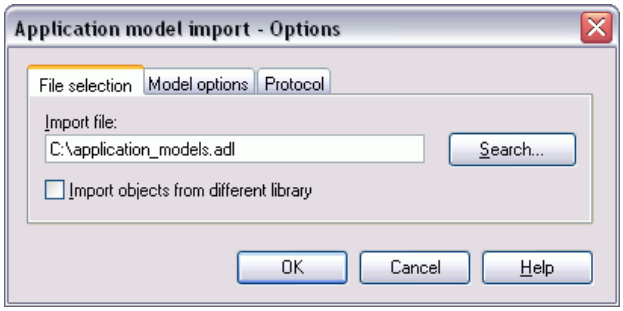


Figure 410: Application model import - File selection

Enter the path and the name of the ADL file with the model you wish to import into the **"File selection"** tab. The button **"Search"** opens a window allowing the user to search for the file to be chosen.

The option **"Import object from different library"** is available to import models which are based on an application library which differs from the one assigned to the user.

Tab "Model options":

Here you define settings for import of application models:

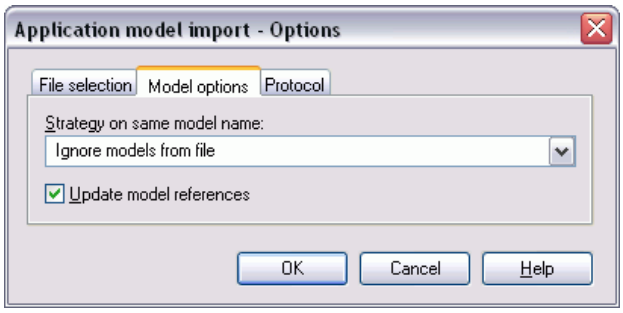


Figure 411: Import of application models - Model options

In the case of application models in the file to be imported, which have *the same names as the models stored in the ADOxx database*, you can always use the option **"Ignore models from file"** as the **"Strategy on same model name"**.

> **Hint:** When importing application models, which come from an application library with time-related versioning, you can choose the option "Increase version number" as reaction when the names of the models are the same. As a result, when you get an application model of the same model type and with the same model name, the next version number will be assigned to it during import automatically.

Activate the option **"Update model references"** to correspondingly update the references within the imported models during the change of the name of the imported models. Thanks to this option all the references to the imported models will be automatically updated.

> **Hint:** The option "Update model references" only has an affect when importing application models to an application library with a time-related version where you have selected the option "Increase version number" as a response.

Tab "Protocol":

In this tab you can have a protocol generated for the ADL Import.

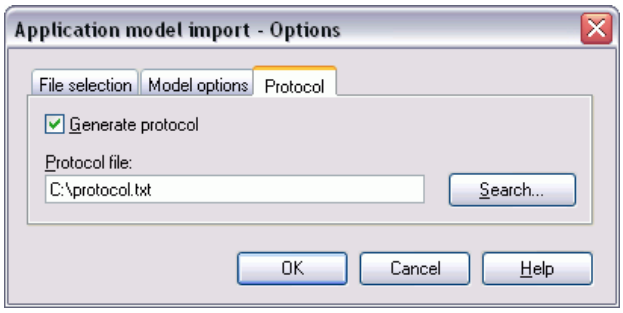


Figure 412: Application model import - Protocol

For this, activate the option **"Generate protocol"** and enter the path and the name for the import protocol (TXT file) into the **"Protocol path file"**. The Button **"Search"** allows the user to search for the ADL file.

Once you have successfully entered the settings, click on the **"OK"** button to assign them.

6.1.6 Model overview

The content of the ADL file to be imported is read and the application models contained in the file are listed in the window **"Application Model Import - Model overview"**:

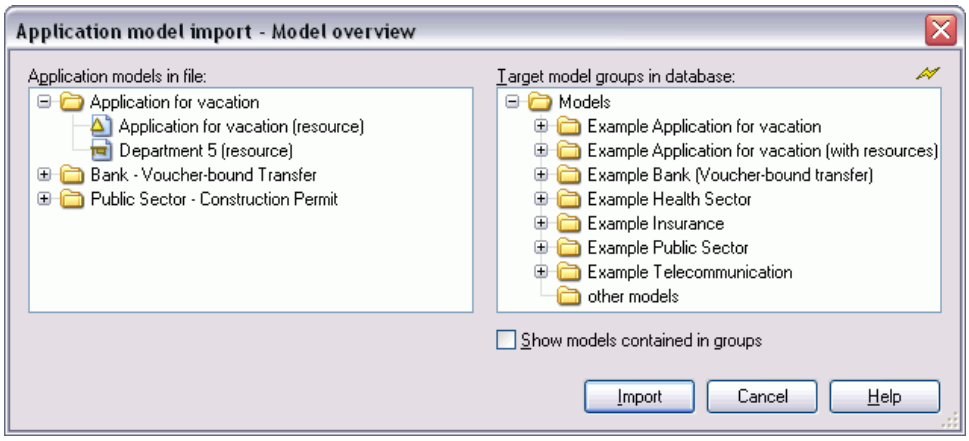


Figure 413: Application model import - Model overview

Select the model group into which a reference to the models defined in the application models imported should be generated.

> **Hint:** If you don't select a destination model group, the imported models will not be assigned to a model group and so can not be opened by an ADOxx user. Before the import, a security query will inform you about this.

Click on the button **"Import"** to import the models.

A status window will inform you on the progress of the ADL import.

> **Hint:** If the ADL file contains a model of a type which is not defined in the current application library, you can assign a model type (see chap. 6.1.2.1, p. 453) defined in the application library to this model during the import.

6.1.7 Result

After the ADL Import has been successfully completed, the window **"ADL Import - Results"** will inform you about the conclusion of the import. This window lists the names under which the import models are stored in the ADOxx database.

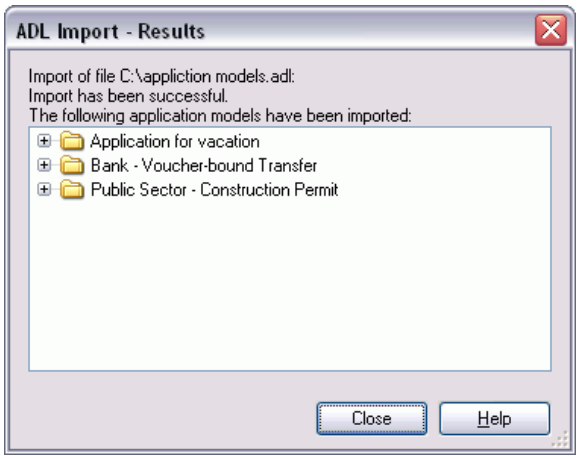


Figure 414: Application model import - result display (with protocol)

If you activate the option "Generate protocol" in the window "ADL Import - Options" - "Protocol" tab (see fig. 410) and give the path and the name for the protocol file to be created, then the button **"Protocol"** will be shown in the "ADL Import - Results". By clicking on this button the generated protocol will be shown in a text editor.

6.2 ADL Export

The ADL Export functionality allows you to save your models in text format (ASCII) and, if necessary, import them into a different ADOxx database.

You may export either **models/attribute profiles** (see chap. 6.2.1, p. 468) (and model groups/attribute profiles groups) or **application models** (see chap. 6.2.2, p. 472).

6.2.1 Model Export

With this function you can export models, model groups, attribute profiles and attribute profile groups.

Start an ADL export for models/attribute profiles:

There are three options how to begin exporting:

- Click on the smart-icon ,
- Click and hold the smart-Icon and select from drop down menu the option **"Models/attribute profiles"**,
- Select from menu **"Model"** the option "ADL Export" **from the sub menu select the option "Models/Attribute profiles"**.

The window "ADL Export - Selection" with model group structure of ADOxx-Database will be shown:

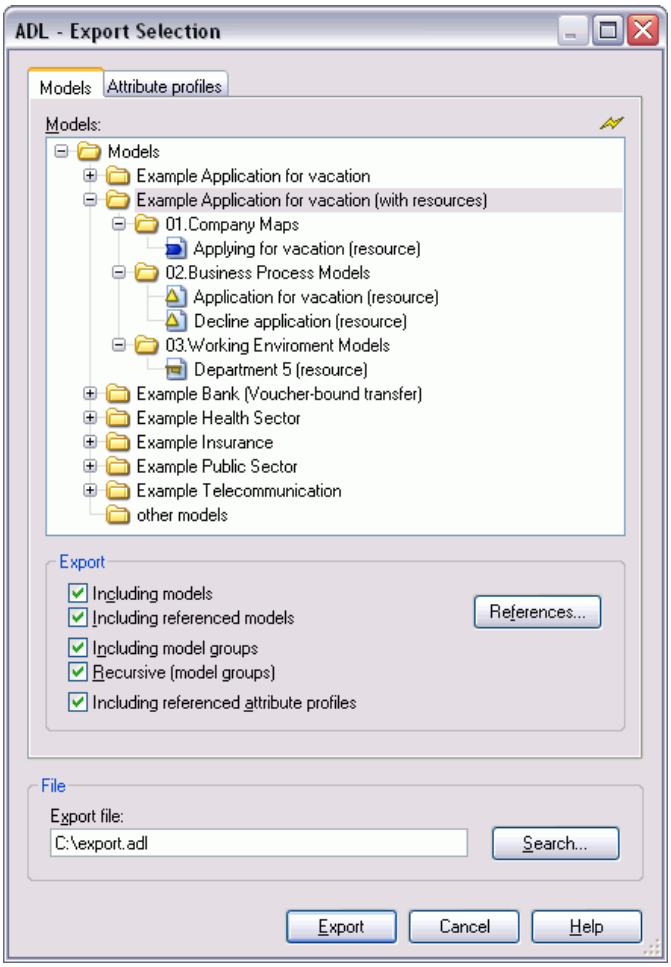


Figure 415: ADL export - Model selection

Through **tabs** "Models" and "Attribute profiles" you can toggle between model and attribute profile selection. In the window **"Selection"** the complete hierarchy of models, model groups or attribute profiles is visable. The function **"Export"** contains export options.

Prepare model export:

Select the tab **"Models"**, the model hierarchy will be shown:

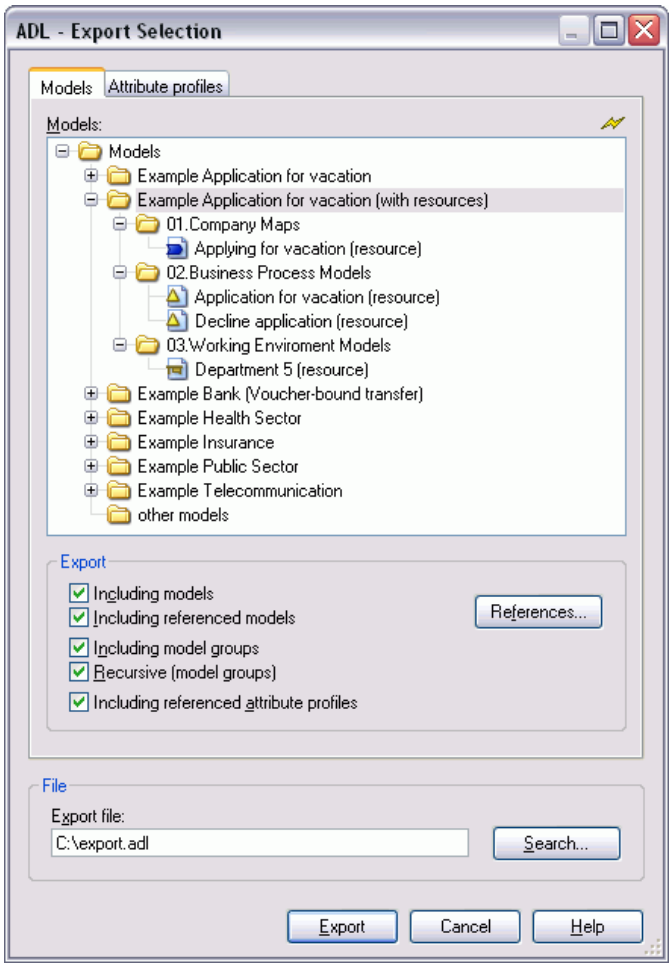


Figure 416: Model and model group selection

Select in the window **"ADL Export - Selection"** the models and/or model groups to be selected.

Define the required export options:

Including models The selected models will be imported.

Including referenced models

Apart from the selected models to be exported, the models referenced to these selected models will be exported - regardless of their selection. Clicking on the option "References" (see chap. 11.1, p. 97), will enable you to change the settings . ("inclusive referenced models" is only accessible, if option "Models" is active.)

Including model groups The selected model groups will be exported.

Recursive (model groups)

Apart from the selected model groups to be exported, all the sub groups of all hierarchy levels will be exported. ("Recursive (model groups)" is only accessible, if option "Model groups" is active.)

Including referenced attribute profiles

All referenced attribute profiles of the models to be exported will also be exported. ("Including referenced attribute profiles" is only accessible, if the option "Models" is active.)

Prepare attribute profile export:

If it is not necessary that you also export automatically all referenced attribute profiles, you can always select the ones to be exported manually. In order to do this choose the tab **"Attribute profile"**. Attribute profile hierarchy will be shown:

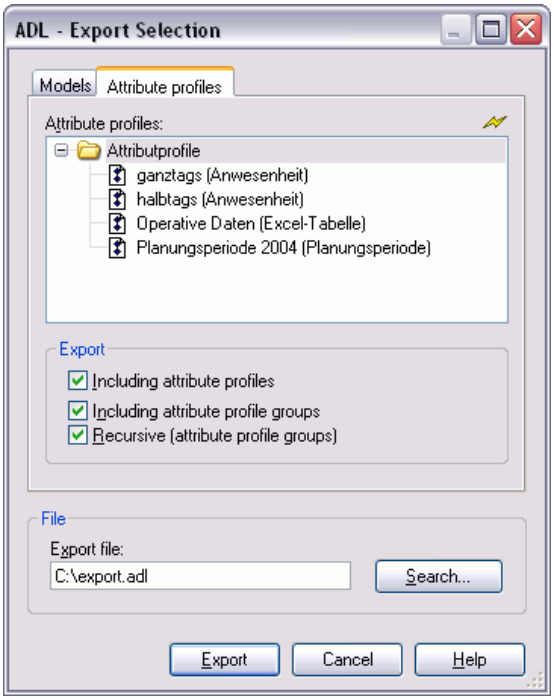


Figure 417: Select attribute profiles and attribute profile groups

ATTENTION: You can select attribute profiles to be exported, only if in tab "Models" the option "Referenced attribute profiles" is *deactivated*.

Mark in the window **"Selection"** the attribute profiles and/or groups to be exported.

Define required export options:

Including attribute profiles Selected attribute profiles will be exported.

Including attribute profile groups Selected attribute profile groups will be exported.

Recursive (attribute profile groups)

Apart from the selected attribute profile groups, all the sub groups of all hierarchy levels will be exported. ("Inclusive all sub groups" is only accessible, if option "Attribute profile groups" is active.)

Export:

Define in **"File name"** the path and the name for the ADL file to be created. The button **"Search"** opens the supporting dialogue window.

Click on button **"Export"** to start exporting. A status window will keep you informed about the progress. Another window will inform you when the ADL export has been completed successfully.

Special case: ADL Export for ADOxx Version 3.0x

If you want to export models and/or model groups as an ADL file for ADOxx Version 3.0x open the window **"Browse"** to enter the path and the name for the file. In the drop down field **"Data type"** choose **"ADL 3.0 files"**.

> **Hint:** When exporting ADL to ADOxx Version 3.0x you will lose all model information, which was implemented from Version 3.5 (e.g. attributes in a new attribute type).

Special case: ADL export for ADOxx version 3.81

During ADL export in ADOxx 1.0, redundant class attribute values of connectors are no longer exported. If you want to include these attribute values into an ADL export, open the window **"Browse"** to enter the path and the name for the file. From the drop-down field **"Data type"** choose **"ADL 3.81 files"**.

> **Hint:** The class attribute values of connectors are defined in the application library and are not modified by the ADL file (up to version 3.81). Not exporting the class attribute values (default in ADOxx 1.0) does not have any negative impact on data backup or data transfer between ADOxx databases.

6.2.2 Export Application Models

This function allows for the export of application models stored in a database.

Start ADL export of application models:

There are two possible ways to do it:

- Click and hold on the smart-Icon and select from the drop-down menu the option **"Application models"**,
- Select from menu **"Model"** the menu item **"ADL Export"** and from its sub menu **"Application models"**.

Window "ADL Export - Selection of application model" with current application models will be shown:

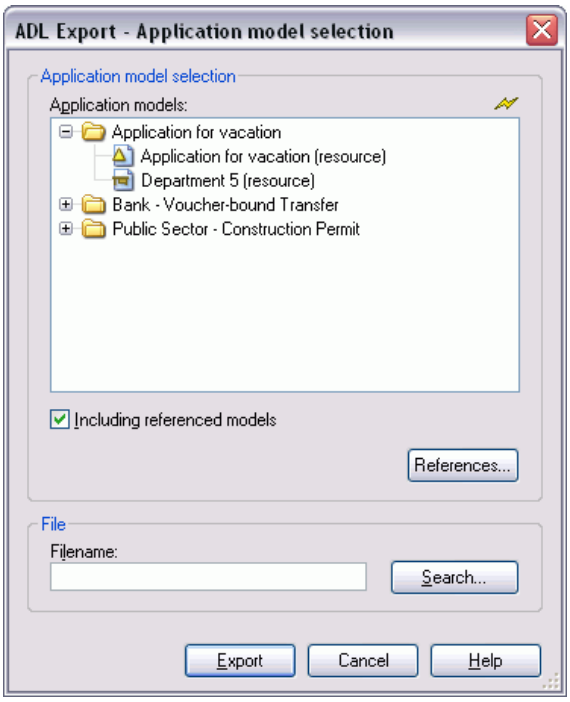


Figure 418: Export application models

All application models are displayed in the window **"Application models"**. Mark all models which shall be exported.

Activate the option **"Including referenced models"**, to export selected application models as well as all their referenced models. With the button "References" (see chap. 11.1, p. 97) these settings can be changed.

Export:

Define in **"File name"** the path and the name for the ADL file. With button **"Search"** you can open the supporting dialogue window.

Click on the button **"Export"** to start exporting. A status window will keep you informed about the progress. Another window will inform you when the ADL export has been completed successfully.

Special case: The use of a sequence version and a version-specific application library

When using a time-based application library, you can switch between the options *Export of version specific application models* and *Export of sequence version application models*.

When exporting *Version specific application models*, only models with the same model versions as the models contained in the application model will be exported. When exporting *Sequence version application models*, all versions of models contained in application model will be exported.

6.3 XML Import

The option XML Import (see chap. 6.3.1, p. 474) allows for the import into database models and attribute profiles, which were earlier externally stored with XML Export (see chap. 6.4, p. 481).

6.3.1 Model Import

This function allows the import of models, model groups, attribute profiles and attribute profile groups. This process consists of three steps:

- To start import
- To define import settings and options
- To define aim and input to be imported
- To import

To start XML Import of Models/Attribute profiles:

Select from menu **"Model"** menu item **"XML Import (Standard)"** with sub menu item **"Models/Attribute profiles"**.

Window "XML Import - Settings" (see fig. 419) will be shown.

This dialogue consists of four parts, which can be identified with tabs.

XML import - Settings

In the window "XML Import - Settings" you can define in tabs "File selection", "Model options", "Attribute profiles" und "Protocol" settings concerning import.

> **Hint:** The tab "Versioning" is only available after an appropriate definition in the application library (Customising).

Tab "File selection":

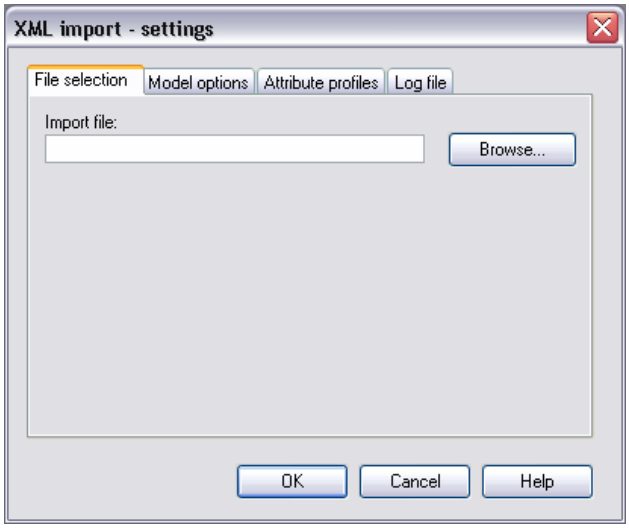


Figure 419: XML import settings - file selection

Enter the path and the name of the XML file with the model you wish to import into the **"File selection"** tab. The Button **"Search"** opens a window supporting the user to find the location of the file to be chosen.

Tab "Model options":

Here you can find settings for the import of models or model groups:

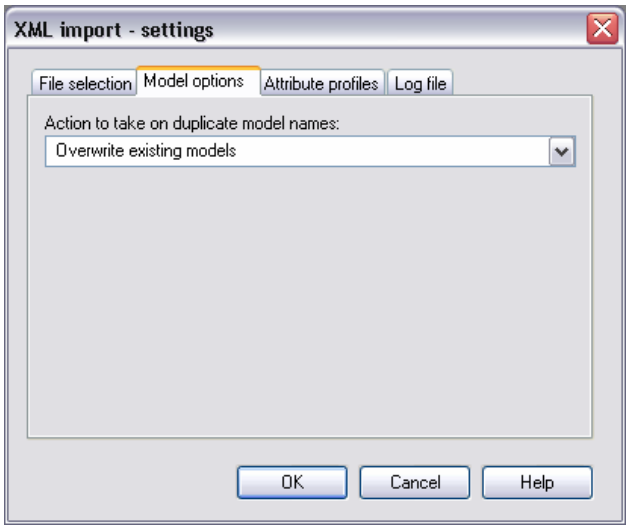


Figure 420: XML import settings - model options

To import models, which have the *same name as the models stored in the ADOxx database*, you can select one of the following options from the list "Strategy on same model name":

- "Overwrite existing models" (see chap. 6.1.4.4, p. 460)
- "Paste model contents into existing models" (see chap. 6.1.4.3, p. 455)
- "Ignore models from file" (see chap. 6.1.4.5, p. 461)

Tab "Attribute profile options":

In the "Attribute profiles" tab you can select one of the options for the import of attribute profiles:

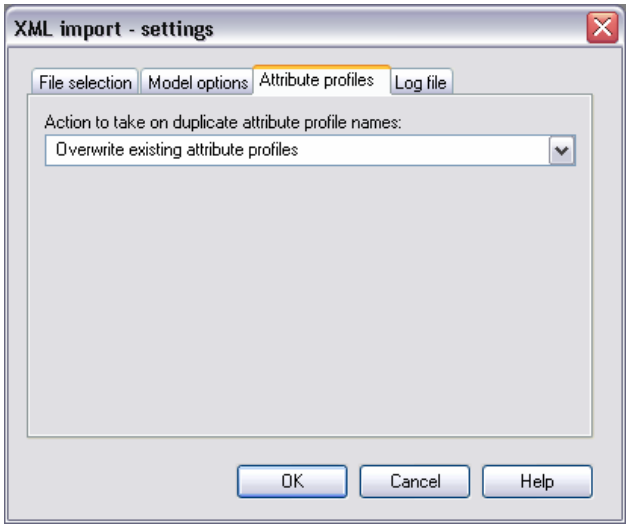


Figure 421: XML import settings - attribute profile options

To import attribute profiles which have the *same name as the profiles stored in the ADOxx database*. You can select one of the following options from the list **"Strategy on same attribute profile names"**:

- "Overwrite existing attribute profiles" (see chap. 6.1.4.9, p. 463)
- "Ignore attribute profiles from the file" (see chap. 6.1.4.10, p. 463)

Tab "Protocol":

This tab will enable you to generate a protocol for the XML import:

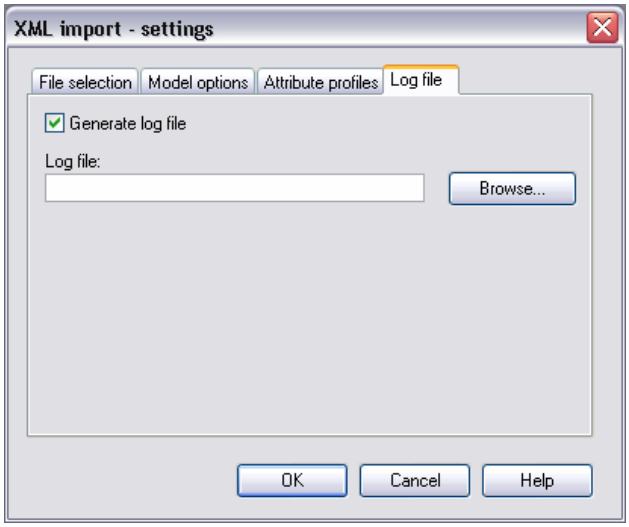


Figure 422: XML import settings - log file

Activate the option **"Generate protocol"** and enter into **"Protocol file"** field the path and the name for the import protocol (TXT file). Button **"Search"** opens a window allowing the user to search for the file to be chosen.

After selecting all the options click on **"OK"**, To start XML import of models or attribute profiles.

6.3.2 Selection

After selecting the XML file and determining the required settings (see chap. 6.3.1, p. 474) the imported data is read; All the contained models (see fig. 423) and/or attribute profiles (see fig. 425) will be shown in the **"XML Import - selection"** window.

Hint: The display of tabs "Models" and "Attribute profiles" depends on input of XML file, i.e. only models in tab "Models", or only attribute profiles in tab "Attribute profiles" can be shown, exclusively one of two tabs can be available.

Tab "Models":

Models contained in XML file will be shown here:

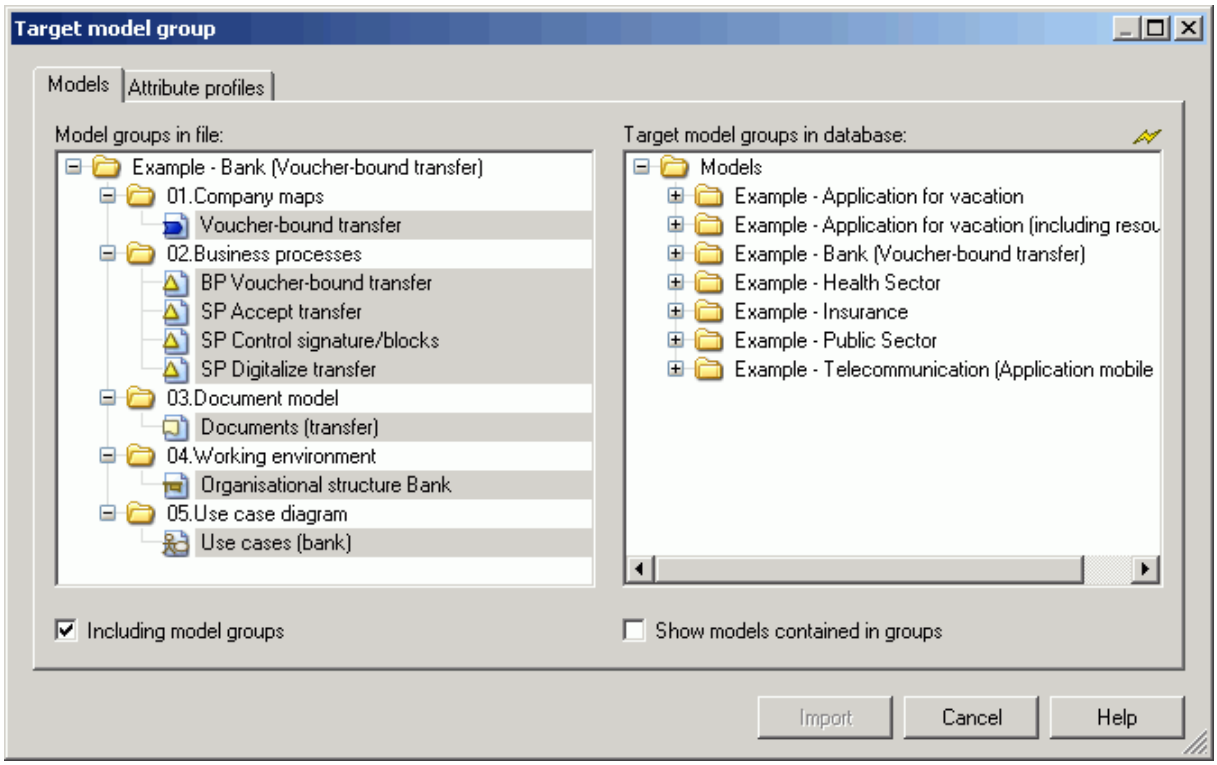


Figure 423: ADL Import - Model selection including model groups

If the option **"Inclusive model groups"** is *activated* in the "Models" tab, the models contained in the XML file will be displayed according to their model hierarchy; When this option is selected the model group hierarchy will be imported as well as the selected models.

Hint: If no models are selected to be imported, only the model group hierarchy will be imported.

If you deactivate the option **"Including model groups"**, the models contained in the XML files are displayed according to their model type. Only the selected models will be imported while using XML import:

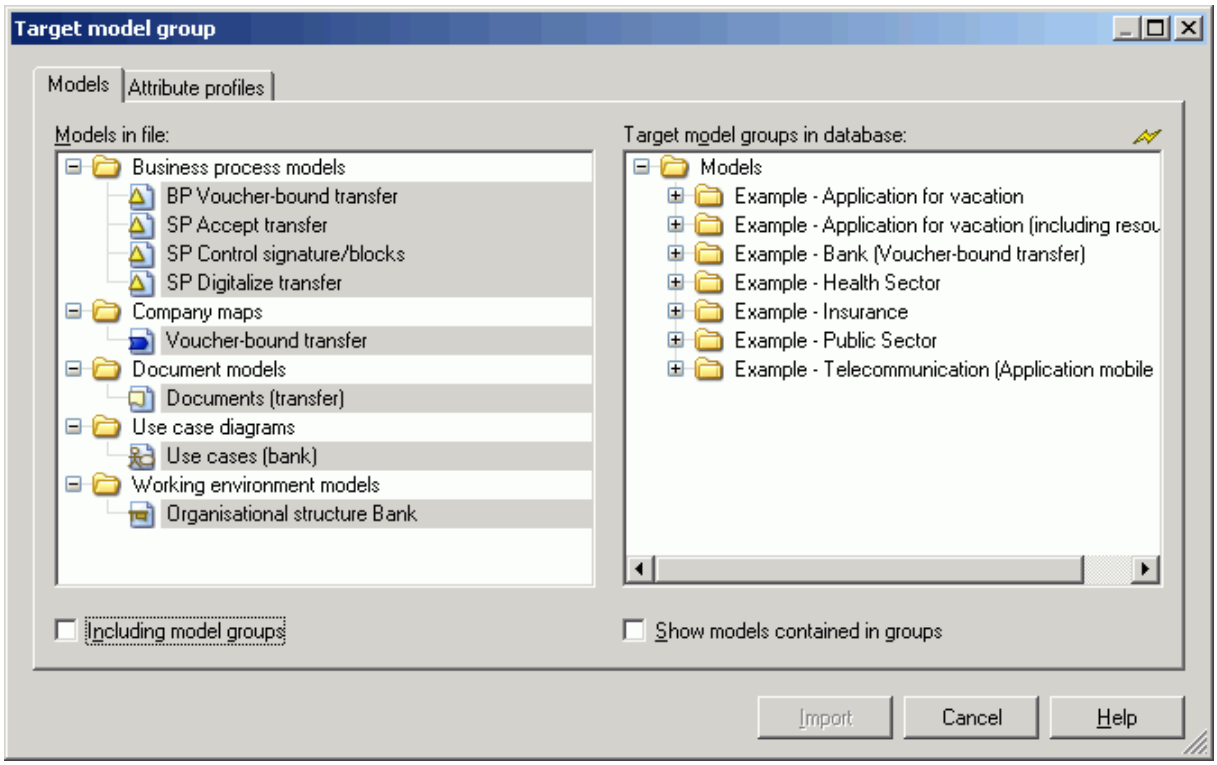


Figure 424: ADL Import - Model selection excluding model groups

Select the option **"Show models in groups"** to see lists of models contained in all model groups.

Select imported models: Select all the models you want to import and the model groups in which you want to create references to the imported models.

"Attribute profiles" tab:

Here all attribute profiles contained in an XML file will be displayed:

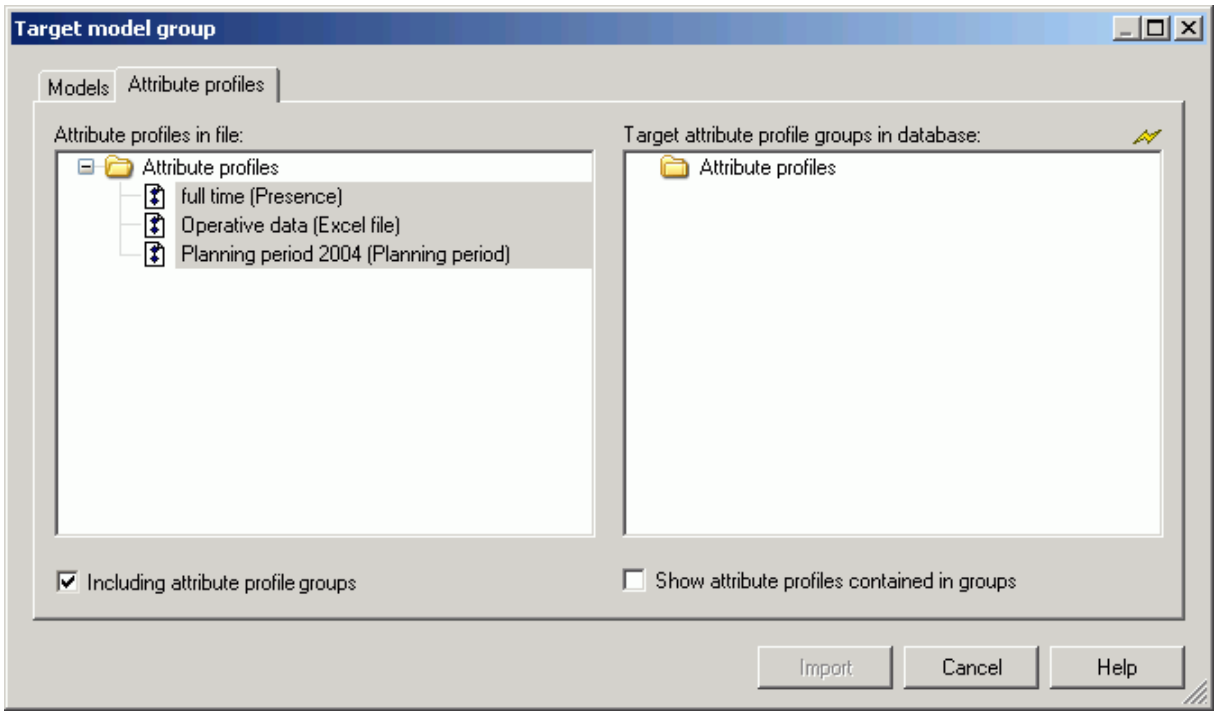


Figure 425: XML Import - Attribute profile selection inclusive Attribute profile groups

If in tab "Attribute profiles" the option **"Inclusive Attribute profile groups"** is *active*, you can see all the attribute profiles (in profile hierarchy) contained in an XML file (see fig. 425). During XML import not only selected attribute profiles, but also attribute profile groups hierarchies will be imported.

Hint: If you do not choose attribute profiles to import, only attribute profile group hierarchies will be imported.

If the option **"Inclusive Attribute profile groups"** in tab "Attribute profiles" is *deactivated*, all attribute profiles form XML file will be presented without arrangement. Hence, only selected attribute profiles will be imported:

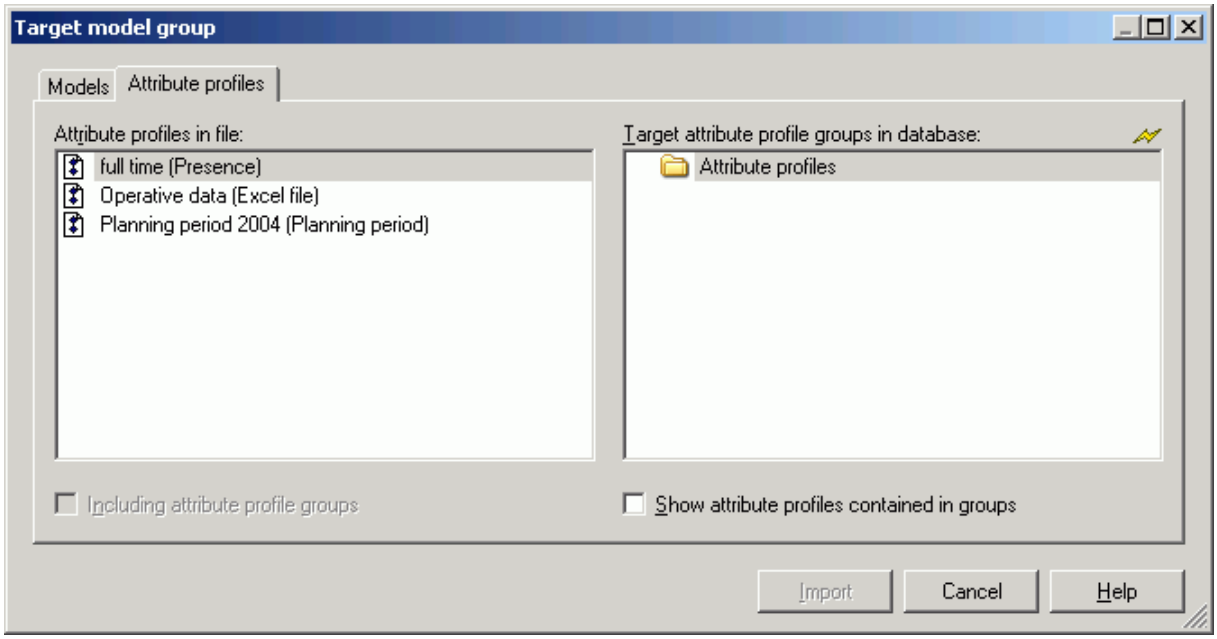


Figure 426: XML Import - Attribute profile selection excluding attribute profile groups

If you select the option **"Show attribute profiles contained in groups"** the attribute profiles contained in the ADOxx Database will be displayed.

Selection of import profiles: Select all the attribute profiles you want to import and attribute profile groups in which you want to create reference to the imported profile.

Import:

After choosing models and attribute profiles click on **"Import"**, to start XML import.

At the beginning of the import, the ADOxx database is first checked for models or attributes which have the same name as the models or attributes to be imported.

> **Hint:** Only the name of the models or of the attribute profiles to be imported is compared to the models or attribute profiles stored in the ADOxx database, not their contents. Therefore, an imported model or attribute profile may have the same content as one stored in the ADOxx database, but a different name. This means it is very important to select the appropriate options as listed in the window "XML Import - Settings" (see fig. 420) for importing models (see chap. 6.3.1, p. 474) with identical names or in the window "ADL Import - Settings (see fig. 421)for importing attribute profiles (see chap. 6.3.1, p. 474)with identical names.

A status window keeps you informed about the current status of the XML import.

6.3.3 Result

After the XML import has been successfully completed, the window "XML Import - Results" will inform you about the conclusion of the import. This window lists the names under which the imported models are stored in the ADOxx database:

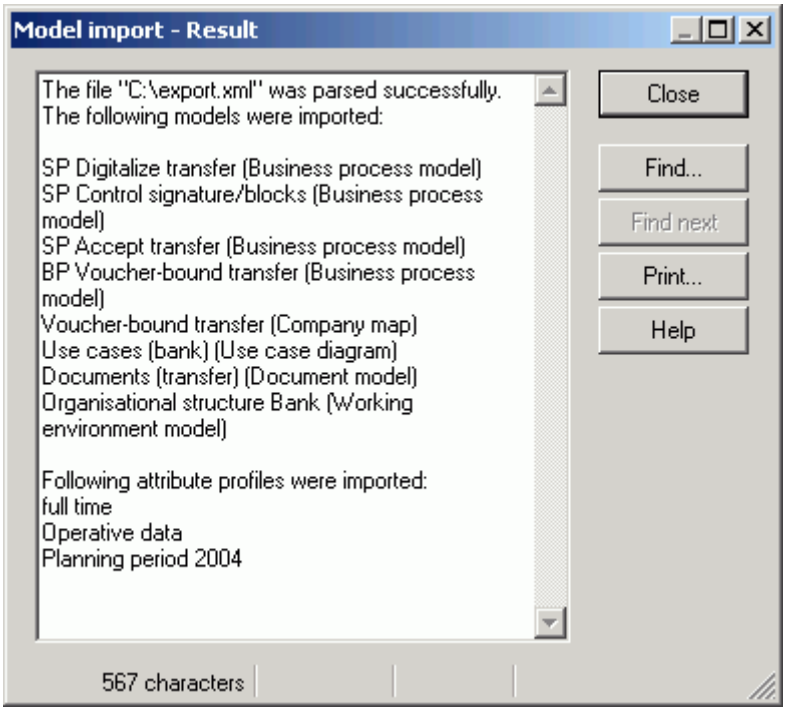


Figure 427: XML Import - Result display (with protocol)

If you activate in window "XML - Import - Settings" - tab "Protocol" (see fig. 381) the option "Generate protocol" and give the path and the name for the protocol file to be created, then the button "Protocol" will be shown in the "XML Import - Results" (see fig. 390) window. By clicking on this button the generated protocol will be shown in a text editor.

Hint: If you have imported exclusively model groups, the successful XML import will be indicated through an appropriate message window.

6.4 XML Export

The XML Export functionality allows you to save your models in text format (ASCII) and, if necessary, import them into a different ADOxx database.

XML Export is possible for **Models/Attribute profiles** (see chap. 6.4.1, p. 481) (also together with Model groups/Attribute profile groups).

Hint: Use ADL Export (see chap. 6.2.2, p. 472) to save **application models** externally.

Hint: In ADOxx 1.0 XML files will be created with encoding "UTF-8".

6.4.1 Model Export

This function allows for the export of models, model groups, attribute profiles and attribute profile groups.

To start XML Export for models/attribute profiles:

Select from menu **"Model"** menu item **"XML Export (Standard)"** and then its sub menu item **"Models/Attribute profiles"**:

The window "XML Export - Selection" with the model groups structure of ADOxx-database will be shown:

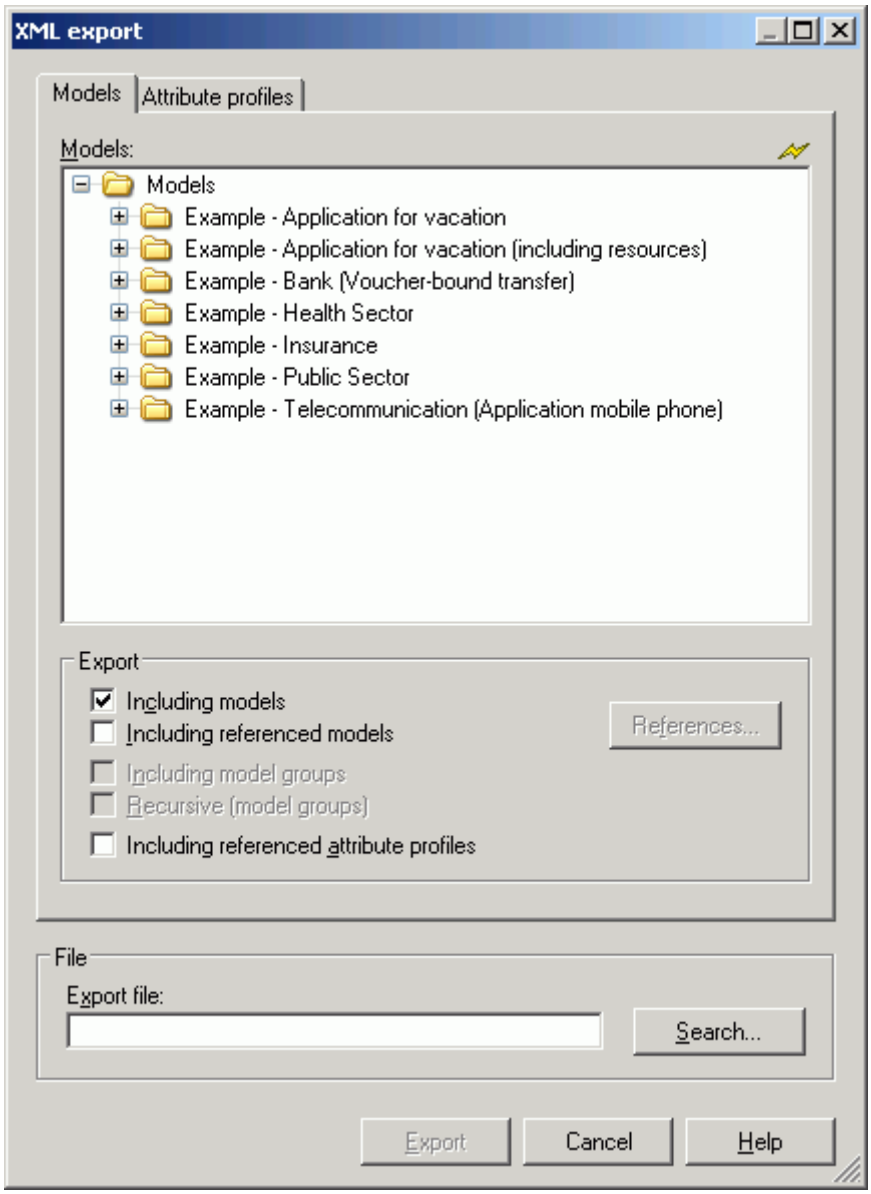


Figure 428: XML Export

Through tabs "Models" and "Attribute profiles" you can toggle between model and attribute profile selection. In the window **"Selection"** the complete hierarchy of models, model groups or attribute profiles is apparent. The function **"Export"** contains export options.

To prepare model export:

Select Tab **"Models"**. Model hierarchy will be shown:

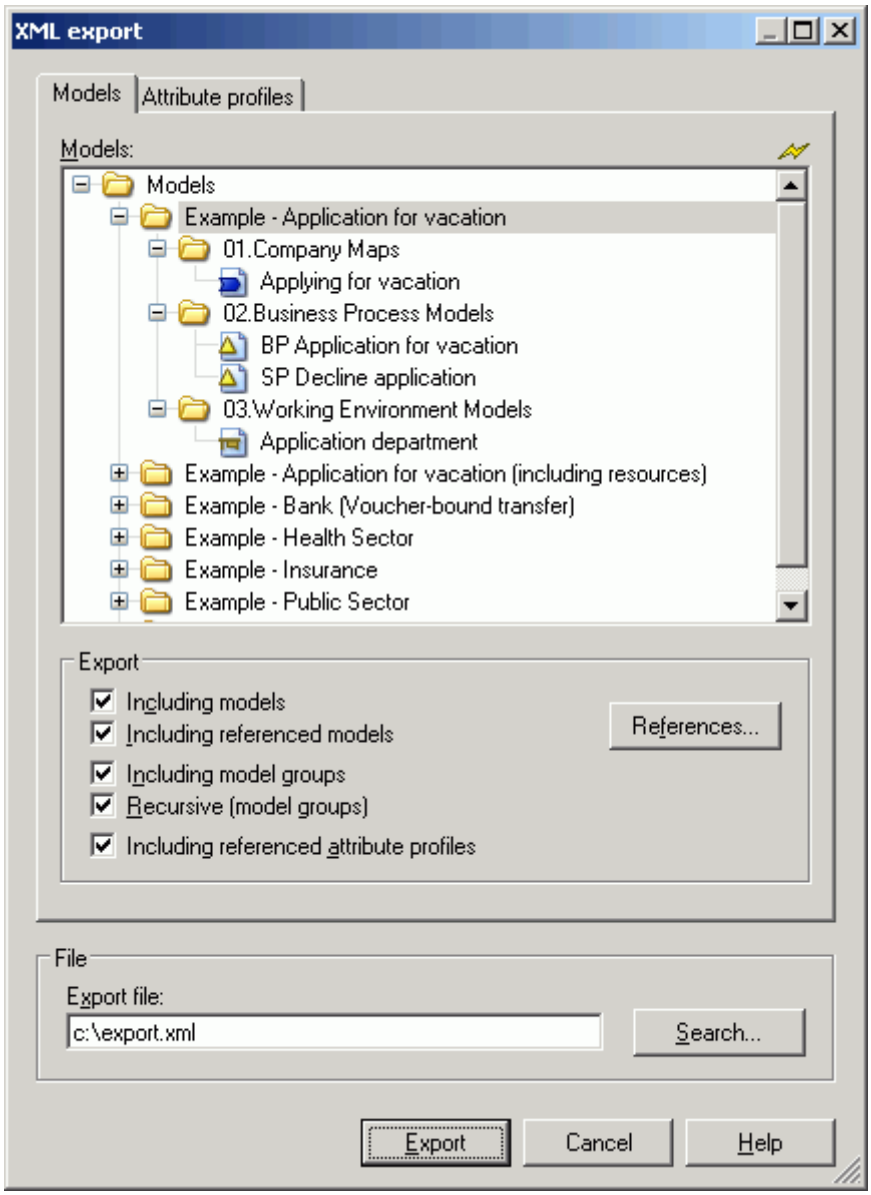


Figure 429: Model and model group selection

In the window **"Selection"** select models and/or model groups to be exported.

Define the required export options:

Models The selected models will be exported.

Including referenced models

Apart from the selected models to be exported, the models referenced by these selected models will be exported - regardless of their selection. Clicking on the "References" (see chap. 11.1, p. 97) button, will enable you to change the settings . ("inclusive referenced models" is only accessible, if the option "Models" is active.)

Model groups The selected model groups will be exported.

Inclusive all contained model groups

Apart from the selected model groups to be exported, all the sub groups of all hierarchy levels will be exported. ("Inclusive all models contained" is only accessible, if the option "Model groups" is active.)

Referenced attribute profiles

All referenced attribute profiles of the models to be exported will also be exported . ("Referenced attribute profiles" is only accessible, if the option "Models" is active.)

To prepare attribute profile export:

When it is not required to automatically export all attribute profiles, which are referenced by the exported models, you can manually select the attribute profiles to be exported. To do this choose tab **"Attribute profile"**. Attribute profile hierarchy will be shown:

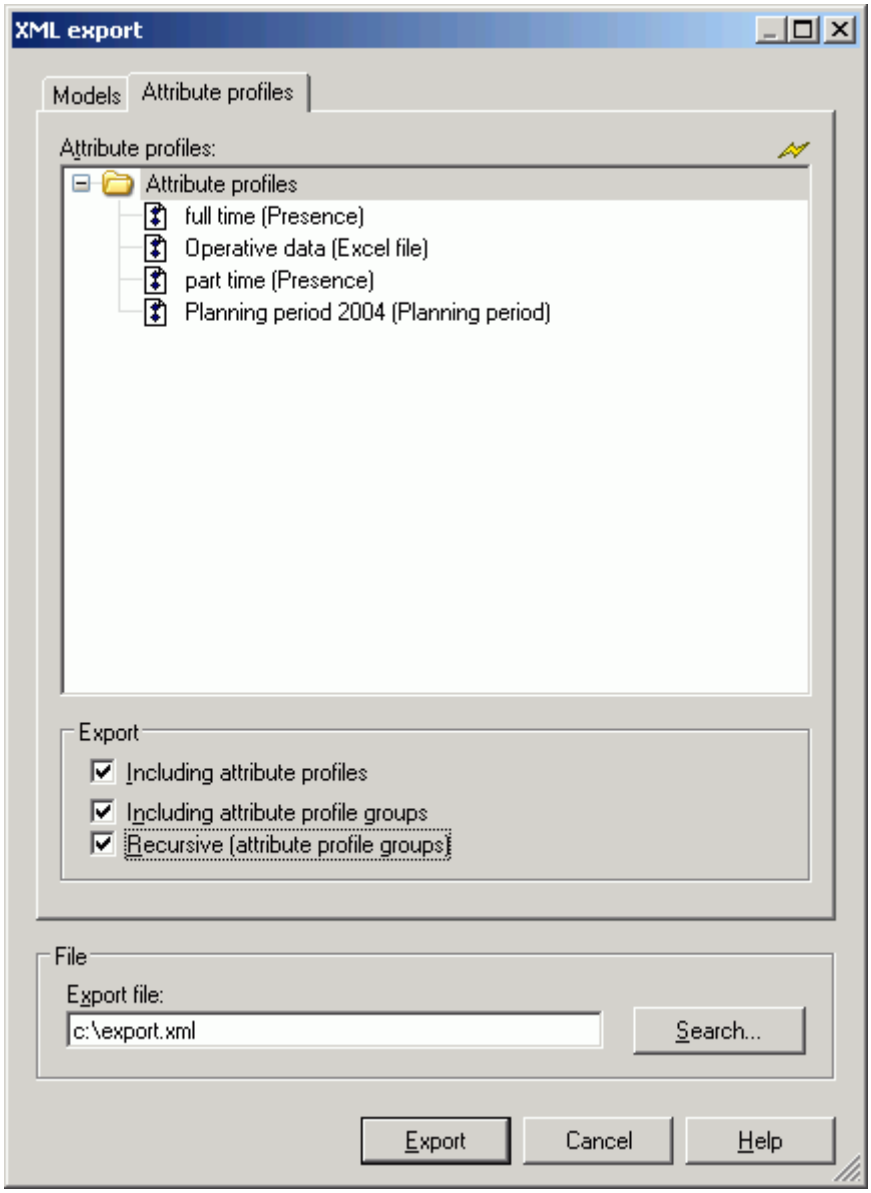


Figure 430: Select attribute profiles and attribute profile groups

ATTENTION: You can select attribute profiles to be exported, only if in tab "Models" the option "Referenced attribute profiles" is *deactivated*.

Mark in the window **"Selection"** the attribute profiles and/or groups to be exported.

Define the required export options:

Including attribute profiles The selected attribute profiles will be exported.

Including attribute profile groups The selected attribute profile groups will be exported.

Inclusive all contained model groups Apart from the selected attribute profile groups, all the sub groups of all hierarchy levels will be exported. ("Inclusive all sub

groups" is only accessible, if option "Attribute profile groups" is active.)

Export:

Define in **"File name"** the path and the name for the ADL file to be created . The **search** button opens window to chhose location of file.

Click on button **"Export"** to start exporting. A status window will keep you informed about the progress. Another window will inform you when the ADL export has been completed successfully.

6.5 Documentation

Using the Documentation Component, you can transform ADOxx models to files (e.g. HTML, RTF, XML). This way it is possible to integrate the model contents including graphical representation into documents (via the word processing program) or to distribute them on intranet.

Hint: The menu items described below refer to the ADOxx-Default-Library and can vary according to your ADOxx' configuration.

Start the generation of documentation (see chap. 6.5.1, p. 486) via the appropriate menu item in the "Documentation" menu.

The documentation will be created using the current valid **settings**. In the settings for the documentation you will define among other things which model contents (model graphics, objects, attributes) will be taken over in the document you are generating.

To define the settings, follow these two steps:

1. **Options** (see chap. 6.5.2, p. 488):

   Selecting the appropriate menu item (in the "Documentation" menu) or by clicking on the smart icon will enable you to display and change the basic settings.

   In the basic settings you will define whether and how the model graphics as well as which classe and attribute information - using predefined modes- will be taken over in the documentation.

2. **Attribute and class filter** (see chap. 6.5.3, p. 493):

   You can show and change the settings of the attribute and class filter by selecting the "attribute and class filter" menu item ("Documentation" menu) .

   In the attribute and class filter, starting from the predefined modes, you can individually shrink or expand classes and relations as well as their attributes , i.e. include or exclude them in the documentation to be generated.

These settings (basic settings and settings of the attribute and class filter) will be saved user-specific .

6.5.1 Generate documentation

To start the generation of documentation files, select the appropriate menu item **(e.g. "HTML generation")** in the **"Documentation"** menu.

Once you have clicked on the menu item, the model window (see fig. 431), which contains all the models stored in the ADOxx database, will appear.

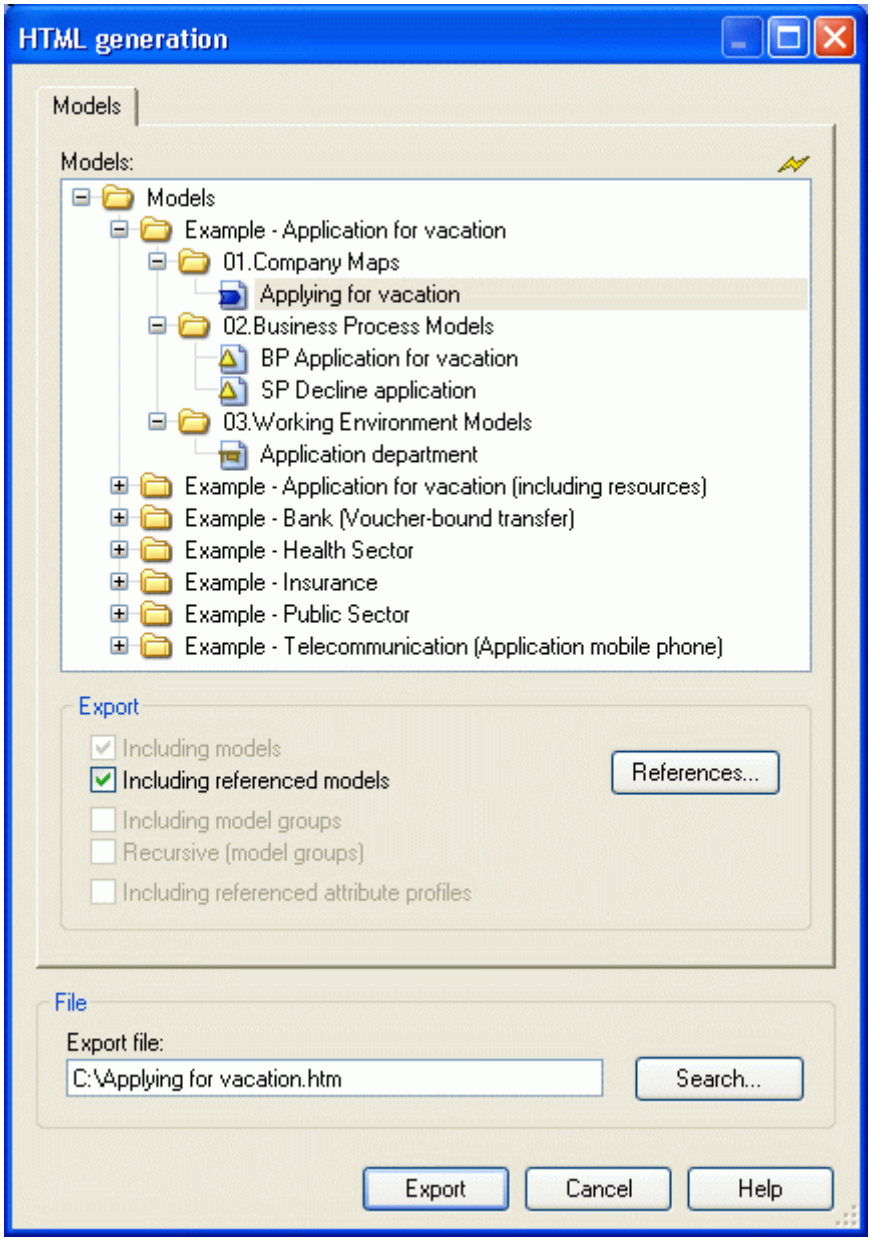


Figure 431: Model selection for documentation (example HTML generation))

Select the models you want to include in the output documentation.

Activate the option **"Including referenced models"**, if all selected models and the models connected to these models are to be generated in the documentation. Clicking on the "References" (see chap. 11.1, p. 97) will enable you to change the settings.

Enter in the **"Export file"** field the path and the name of the file to which the selected models shall be generated and click on the OK button to start the documentation process with the current settings (see chap. 6.5.2, p. 488) or the settings in the attribute and class filter (see chap. 6.5.3.1, p. 495) .

ATTENTION: Note that most of the time not only one file will be created, but several files and/or graphic files. The file names (e.g. of the graphic files) will be created by appending a suffix to the file names you have selected.

Hint: If you want to ensure that all files will meet the "8+3" name convention, choose basic names which do not contain more than two characters (e.g. "dd.htm").

Once your documentation has been successfully generated a corresponding message window is displayed.

6.5.2 Options

Select the **"Options"** menu item from the **"Documentation"** menu to open the ADOxx Notebook in which the settings for the documentation can be made (see fig. 432).

ATTENTION: The following descriptions refer to the settings in the ADOxx-Default-Library. When using an other application library, the available settings can differ from the one shown here.

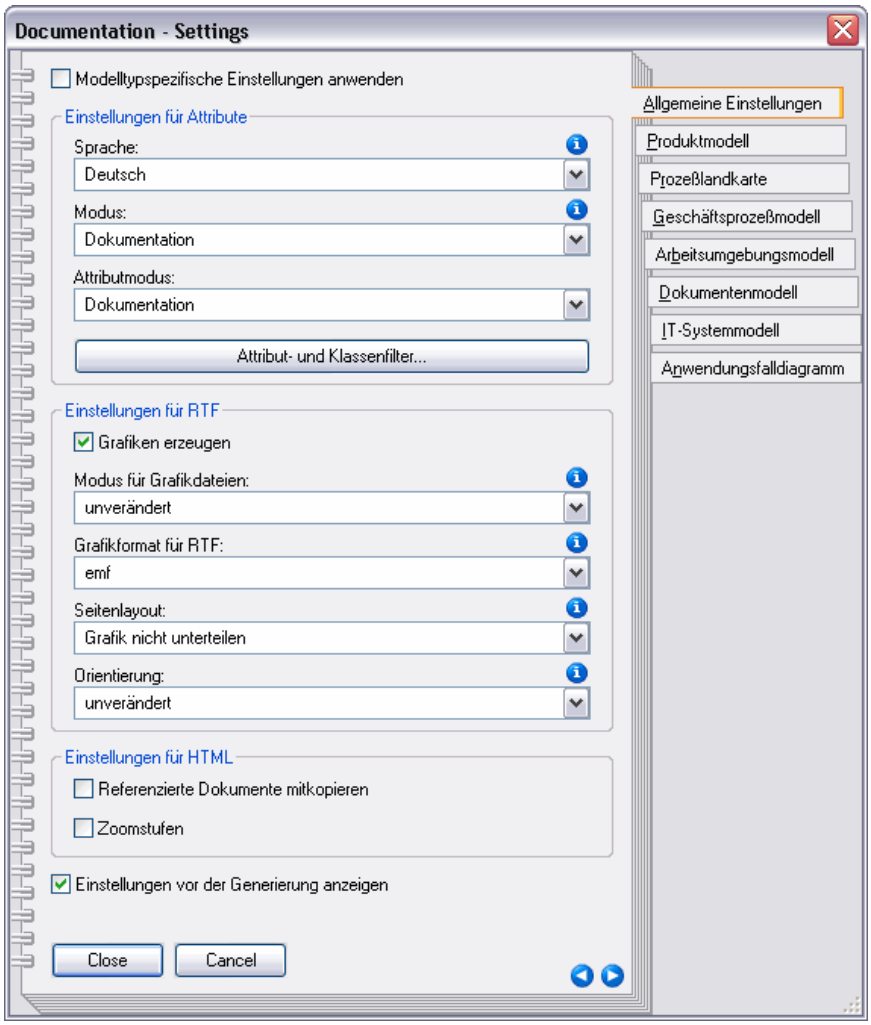


Figure 432: Documentation options of the ADOxx-Default-Library

Hint: The selection possibilities for settings which are available within the "Options - documentation" dialogue (see fig. 432) depend on the application library which you are using.

Beside the chapter "General settings", you will find in the notebook a chapter for each model type.

Hint: The settings "Mode", "Attribute mode", "Graphic file mode", "Page layout" and "Orientation" as well as the option "Create graphics" are available in each notebook chapters. The option "Apply model type specific settings" is only proposed in the chapter "General settings".

You can carry out the following settings when using the ADOxx-Default-Library:

General settings:

Option "Apply model type specific settings"

Normally the settings carried out refer to models of all types. By clicking on the option "Apply model type specific settings" you can also carry out specific settings for each model type. If the option is activated, the settings will be drawn up from the appropriate chapter according to the model type for the documentation generation.

Option "Display settings before generation"

Normally selected settings (e.g. language) will be displayed before the generation is started. The displaying of these settings can be disabled by unchecking the option "Display settings before generation".

Settings for objects and attributes:

Language

Here you can select the language that will be used for creating documentation.

Hint: The selected language does not work in the graphical representation of exported models. To arrange the language also in the graphical representation, select the option "view" in the "view" menu and choose "select the language for the attribute display". Then follow instructions.

Mode

The Documentation Component supports the model view, i.e. the result of the documentation generation can be displayed in the same way as the models in the Modelling Component via the View mode (see chap. 2.2.11.1, p. 186) .When selecting a mode, only objects of a specific modelling class will be taken in the generated documentation.

Example:

In Working Environment models, the objects of the class "Organizational unit", "Performer", and "Role" will be shown in the "Standard" mode, whereas the mode "Role diagram" limits the display to objects of the class "Performer" and "Role" (see fig. 433). During the setting of the "Role diagram" mode for the documentation, only objects of the class "Performer" and "Role" as well as their attributes will be taken over in the documentation.

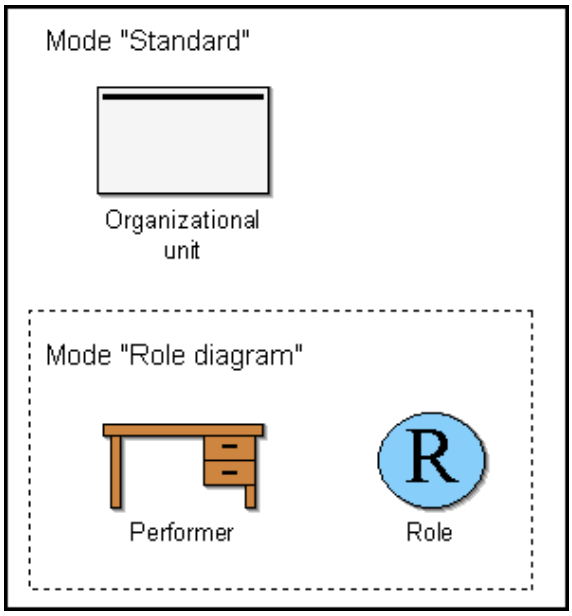


Figure 433: Modes in Working Environment models (Example from the ADOxx-Default-Library)

Hint: The mode option has an effect exclusively on the textual representation of the model contents. The graphical representation of the model will be controlled via the setting "Graphic file mode".

Attribute mode

The attribute mode set defines which attributes of an object will be taken in the generated documentation.

Example:

The attributes represented below are contained in the objects of the class "Role" (see fig. 434).The attribute mode "Documentation" limits the output to the attributes "Name", "Description" and "Comment".

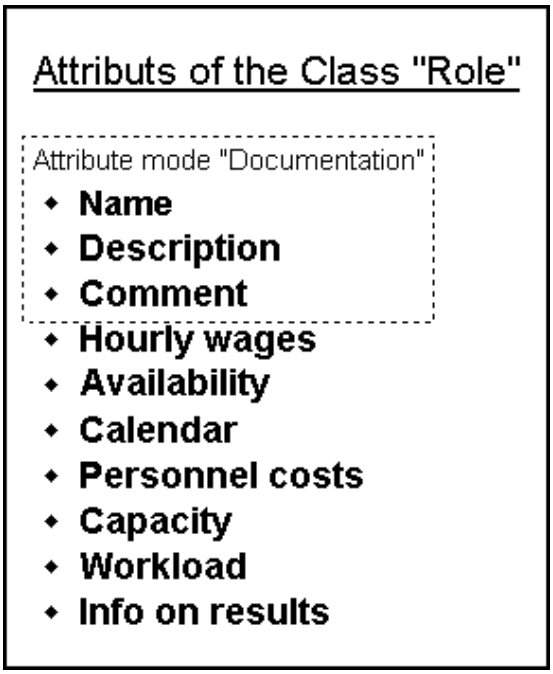


Figure 434: Attributes of the class "Role" (Example from the ADOxx-Default-Library)

Attribute and class filter

Via the Attribute and class filter (see chap. 6.5.3.1, p. 495) you can determine, which information (class, relations and attributes) may be exported during the documentation generation.

ATTENTION: If you activate the "Apply attribute and class filter settings" in the attribute and class filter, only the settings defined in the attribute and class filter (see chap. 6.5.3.1, p. 495) will be applied for the determination of the classes and relations to be taken over in the documentation.

Settings for Word:

"Create graphics"

If the documentation shall contain no graphics, deactivate this option. The result is that no graphic file will be created and the generated document will contain no reference to graphic.

Graphic file mode

The Documentation Component supports the model view in the same way as the Modelling Component via the view modes (see chap. 2.2.11.1, p. 186). When selecting a mode, only objects of a specific modelling class will be taken in the model graphics generated in the documentation.

Hint: Differing from the application of a mode view in the Modelling Component separate to each object, the setting selected here will be applied to all models (of a model type).

The selected graphic file mode has no effect on the mode view in the modelling.

Example:

In Working Environment models, the objects of the class "Organizational unit", "Performer", and "Role" will be shown in the "Standard" mode, whereas the mode "Role diagram" limits the display to objects of the class "Performer" and "Role" (see fig. 433). If setting the "Role diagram" graphic file mode for the documentation, the model graphics in the generated documentation will contain only objects of the class "Performer" and "Role" .

Hint: The "Graphic file mode" option has an effect exclusively on the graphical representation of the model contents. The textual representation of the model will be controlled via the setting "Mode".

Graphics format for Word

The graphics for documentation can be generated in different formats. Select the desired format.

Page layout

Select the page layout to apply to the generated graphics.

ATTENTION: The page layout selected here refers exclusively to the generated model graphics, not to the page layout of the documentation in itself.

Example:

The following picture (see fig. 435) shows an application example of the page layout of graphics in the documentation. In the example of the RTF generation, the model graphic has been created in ADOxx' standard page layout, whereby the graphic is shown separated on two pages according to its size.

Hint: The connector marks displayed in the picture must be created explicitly (see chap. 2.2.11.6, p. 197) .

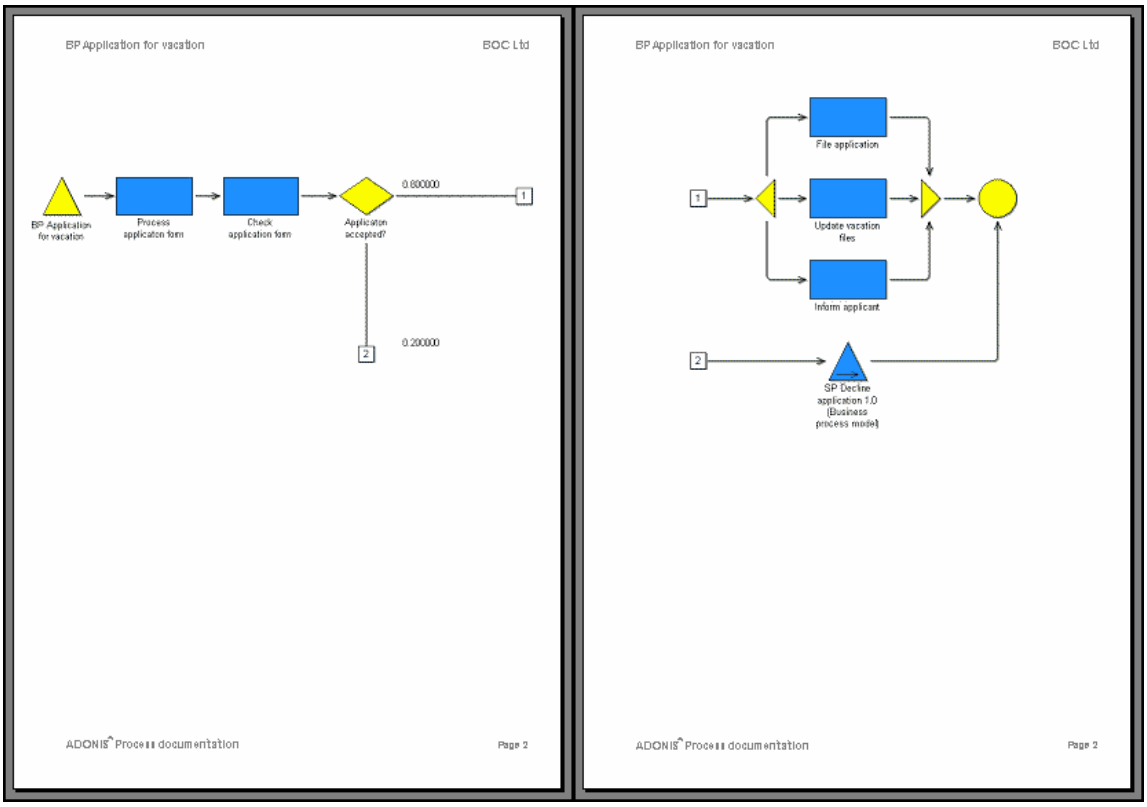


Figure 435: Example of the consideration of a page layout for the RTF generation

Orientiation

Graphic files can be created with or without rotation, by 90° to the left (counter clockwise), by 90° to the right (clockwise) or by 180°. Select the orientation of the graphic files according to the page layout.

Example:

The following picture (see fig. 436) shows an application example of the orientation of graphics in the documentation. In the example of the RTF generation, the model graphic has been created first not rotated (left page) and then rotated by 90° (counter clockwise)(right page). Using the rotation, a larger model will be displayed, which is easier to read .

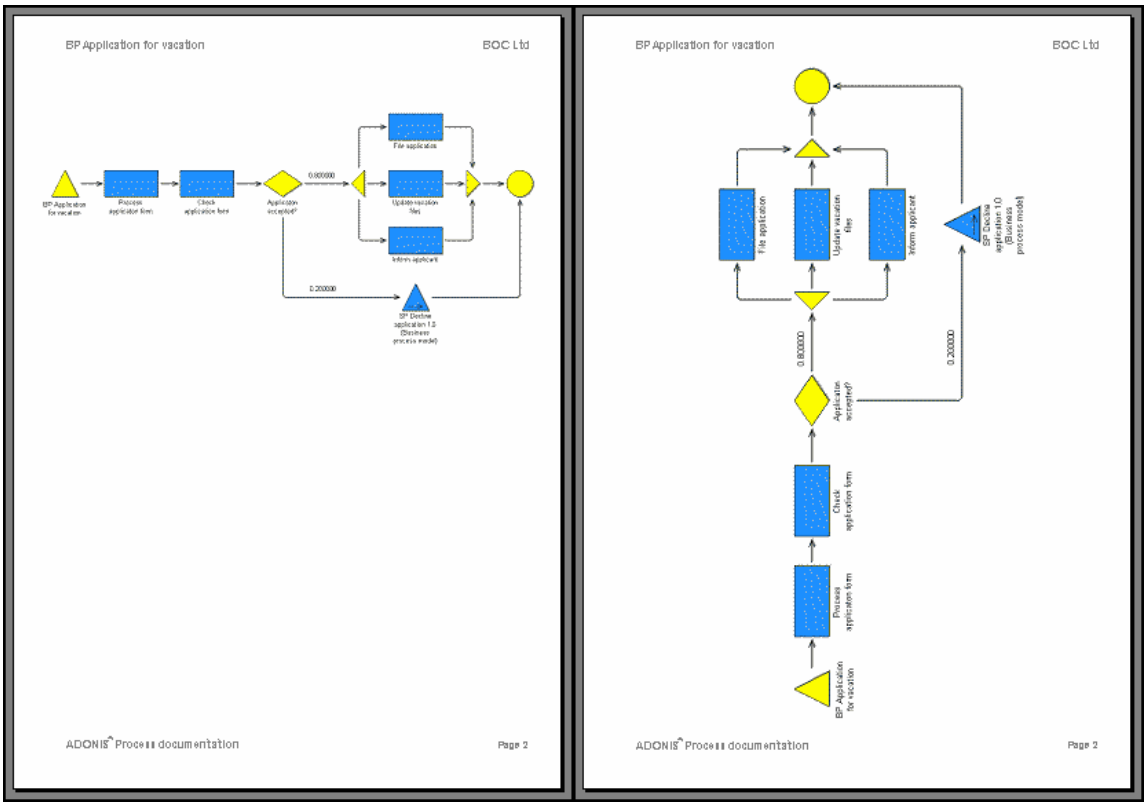


Figure 436: Example of non-rotated and by 90° left rotated graphic for RTF generation

Settings for HTML:

Copying of referenced documents

This function allows the generation to include external documents, that are coming out of referenced models. By doing this, the documents will have an increased data volume, however it is guaranteed to have all necessary documents available in the defined state.

Zoom levels

This function enables zoom levels (50%, 75% and 100%) for sizing the model graphics in the HTML documentation.

ATTENTION: Settings that are defined here will always be used independently from the option "Use the settings from the attribute and class filter" (see chap. 6.5.3.1, p. 495).

Click on the "Close" button to change the current settings. If you click on the "cancel" button, the performed changes will not be saved.

6.5.3 Attribute and class filter

You will determine in the window "Documentation - Attribute und class filter" (see fig. 437) the information (class, relation and attributes) which should be exported during the documentation generation .

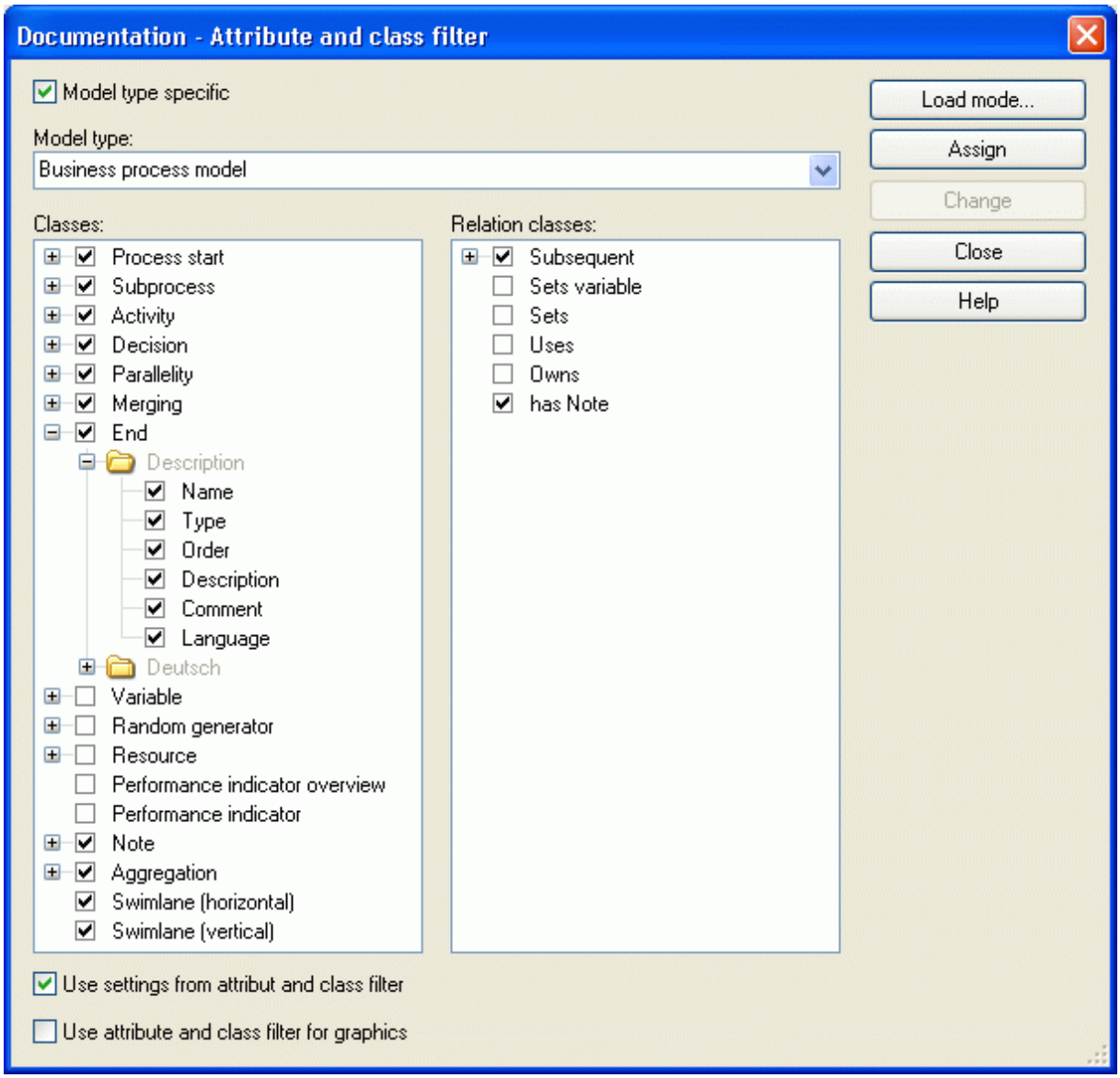


Figure 437: Attribute and class filter for the documentation (settings in the ADOxx-Default-Library)

The **option "Modeltype specific"** controls the selection possibilities in the "Modeltype" list. If this option is activated, you can select a modeltype defined in the application library in the "Modeltype" list. If the option is deactivated all classes and relations will be displayed.

Select in the **"Modeltype" list** the modeltype for which you want to fix the information to be exported.

In both **"classes"** and **"relations"** lists, classes and relations of the selected models will be displayed with their accompanying attributes . The attributes of the particular classes and relations will be represented in the chapter structure of the ADOxx Notebook, i.e. Under each class/relation, the notebook chapters will be listed as files and will contain the attribute of the class/relation.

The symbol ☑ before a class/relation or an attribute indicates that this class/relation or attribute has been taken over in the documentation. Classes, relations and their attributes, which will not be exported into the documentation are marked with the symbol ☐.

Click on the symbol ☑ (the box will be unchecked) to deselect the corresponding classes, relations and attributes for the documentation, i.e. the respective element will not be generated in the documentation.

Click on the symbol ☐ (the box will be checked) to select the corresponding classes, relations and attributes for the documentation, i.e. the respective element will be generated in the documentation.

If you select several elements and click on the **"Change" button**, the status (checked or unchecked) will be changed for all this elements.

Click on the **"Load mode" button** to load (see chap. 6.5.3.1, p. 495) pre-settings to the mode (for the definition of class and relation) and the attribute mode (for the definition of attributes) . The display in the "classes" and "relations" list will be arranged according to the selection.

Activate the **"Apply attribute and class filter settings" option** to use the settings selected to generate the document.

ATTENTION: If you deactivate the "Apply attribute and class filter settings" option, only the settings (see chap. 6.5.2, p. 488)defined in the "Documentation - settings" window (see fig. 432) will be applied.

Activate the **option "Use attribute and class filter for graphics"** to apply your settings also for the generation of graphics within the documentation.

Hint: The option "Use attribute and class filter for graphics" is only available when the options "Use settings from attribute and class filter" has been enabled.

Click on the **Button "Assign"** to save the current settings.

6.5.3.1 Load mode

When loading a mode, you can take over the settings defined in this mode for the documentation in the attribute and class filter (see chap. 6.5.3, p. 493) and then shrink or expand classes and relations as well as their attributes.

The current model type as well as the mode (for the class and relation definition) and the attribute type (for the definition of attributes) will be displayed in the window "Attribute and class filter - Load mode" (see fig. 438).

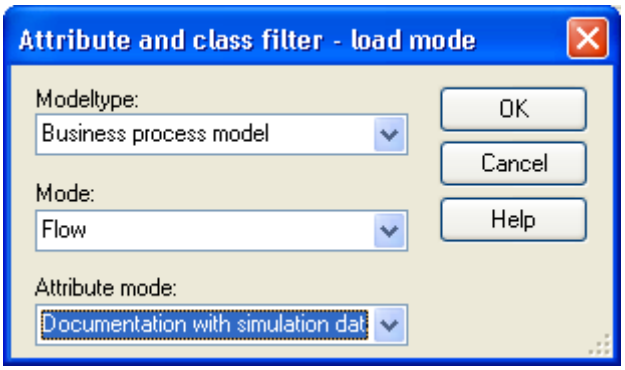


Figure 438: Attribute and class filter - Load mode

Select in the **"Model type" list** the model type for which you want to set the information to be exported.

Select in the **"Class mode" list** the mode in which you want to export the documentation. The option corresponds to the View mode (see chap. 2.2.11.1, p. 186) of models in the Modelling Component, i.e. specific classes and relations will be displayed depending on the mode. Regarding the documentation, this means that only the classes and relations, which are shown in the selected mode (in the Modelling Component) will be drawn up for the generation.

Select in the **"Attribute mode" list** the attribute mode, in which you want to export the documentation. The attribue mode defines the attributes of an object which will be taken in the generated documentation.

Click on the **OK button** to take over the settings in the Attribute and class filter (see chap. 6.5.3, p. 493).

7. Additional components

The additional components "Process cost analysis" (see chap. 7.1, p. 497), "Dynamic evaluation modules" (see chap. 7.2, p. 513) and "Case/4/0 coupling" (see chap. 7.3, p. 517) give the ability to extend the ADOXX standard configuration.

Hint: The additional components are not contained in the package of ADOxx or of ADOxx light.

7.1 Process Cost Analysis

ATTENTION: The component "Process cost analysis" is not contained in the package of ADOxx or of ADOxx light and can be acquired at BOC Group.

Hint: The process cost analysis is integrated into the Evaluation Component of the Modelling Toolkit during the acquisition.

7.1.1 Introduction

The economical general conditions for companies and public organizations have heavily changed in the last years. A rapid evolution of technologies, ever more demanding customers, the globalisation of markets and an ever more tightened competition engendered a drop of profit making related to business volumes. Companies have thus to find new ways to improve their competitiveness. Reconsidering old structures and processes is a promising solution. Considering this we must pay even more attention to business processes within the organization and thus on a way to visualize the processes of business falls in all departments in order to optimize them (Business Process Reengineering). This is why for instance most of the financial service providers are endeavoured to tighten and simplify their business processes and thus to clearly reduce their cycle time and cost rates.

The fundamental part of the process cost analysis is to assign the company's overheads to the calculation object according to the real utilization of the company's activities and not as it used to be, to distribute them to the products using relatively vague surcharges.

Introducing an additional cost account analysis essentially expresses the intention to make up for the lack of certain information regarding costs, which are necessary to support management in their decision making process. The requirement of additional information regarding costs is caused by three factors:

- Changes in company's strategic position owing to technology innovation, increasing competitive pressure on globalize markets and a changed structure of value added.
- Changes of information needs and new calculation structures are required, as well as a costs control for indirect domains and more information regarding costs in order to support strategic decisions.
- Defects of traditional methods for cost accounting, as well as overheads settled without regard to causes, simplistic surcharges and inappropriate distribution, wrong information and therefore wrong indicated signals.

7.1.2 Process cost analysis with ADOxx-3c

The component ADOxx-3c (cost cutting component) enhances ADOxx' capacities towards an even more effective and exact analysis and localization of costs. This component supports you to reach your targets with regard to cost cutting. Using an appropriate structuring and aggregation of calculated values will show among other things the costs caused by particular processes and enable to analyse how efficient the different costs centers are.

Unlike conventional process cost systems - which are exclusively based on analytical methods - ADOxx-3c makes use of the simulation of Business Process Models. This provides you a wide potential to analyse and optimize process costs. You may represent your processes in as many details and on as many details as you wish, since the process structures may also include control structures such as decisions and parallelism and since there is no limit to the number of levels a process may have.

ADOxx-3c supports you for:

- **the transparency in the field of overheads**

  This is particularly suitable for the utilization of existing activities and resources and to provide support to continuous overhead costs management. For example ADOxx-3c can bring transparency to the settlement of internal services by assigning the appropriate costs to all the processes - including so called internal processes - with regard to their causes. This way the users, hence the internal customers of this process, will take into consideration the real costs arising from these processes. An increased transparency and so a more convincing information regarding the costs origin has to be considered during the development of new products or when adapting some processes. ADOxx-3c gives the ability to accurately calculate in advance the costs of a new product or of a changed process.

  The process cost calculation intends to have an influence on the behaviour of employees. A responsible and cost-conscious behaviour regarding both services causing overheads and the concrete execution of these services shall be reached. This more extended transparency, which ADOxx-3c enables, will make activities or processes affecting costs and their consequences clearer for all employees.

- **increasing the efficiency of processes with regard to profitability, quality and time**

  The information, which has been collected using ADOxx-3c are particularly useful to control profitability. This applies first to activities or processes within single cost centers and - this is the essential innovation in process cost calculation - also to inter costs centre business processes. By focusing on processes, a holistic company wide process cost analysis is set up, which does not only focus on separated cost centers but takes into account the various interfaces between different departments and cost centers. It is important to emphasise and make transparent that some improvements in one cost centre can lead to negative effects in others and therefore impact negatively on the value chain of a company as a whole.

- **the process orientated and strategic calculation to support decisions**

  ADOxx-3c supports you not only for decisions about different product variations, but also about customers, marketing, prices and other sales decisions .

The idea when extending ADOxx to a process cost analysis toolkit is to avoid a double work during the acquisition of business processes and sub-processes within the company. If the processes have already been acquired in the framework of a (re)organization project in your company, then it would be easy to improve them with the information which is necessary for process cost analysis. If ADOxx is used as a process management toolkit in your whole company, and if the processes are continuously updated in the framework of "permanent Re-Engineering", then the periodical effort regarding process cost analysis will be enormously facilitated. Using this procedure it is possible to minimize the coverage expenses, which are a drawback in the system of process cost calculation. A consequence of it is a general cost-profits-improvement of the process cost calculation work within the company.

You could also represent the processes as a pure sequence of activities and so find out process costs . If later you want to carry out a more detailed analysis of this process, or of a part of it, this would be possible just by adding further activities, decisions or sub-processes .

7.1.3 Actual and plan analysis

In order to analyse the costs caused by business processes, ADOxx-3c puts two systems at your disposal, the actual and the plan process cost analysis.

- **actual process cost analysis**

  By means of the actual cost analysis you can search for the costs caused by your business processes during a previous period. It is based on the known quantities, such as cost centers budget, cost types, execution times and number of processes. The actual process cost analysis shows where the costs originate and let you determine their amount as well as the average process costs.

- **Plan process cost analysis**

  The Plan-process cost analysis supports you when planning the cost centers budget. By entering the expected number of processes and the time and costs information belonging to them, you will determine the cost centers budget necessary to fulfil the assignment

7.1.4 Input

The input for the process cost analysis ADOxx-3c is an application model (see chap. 2.4.16, p. 284). You can define an application model within the Modelling Component, by selecting in the "Model" menu the "Application model" menu item. It is made of **exactly one** Working Environment model and **at least one** Business Process Model (excluding the sub-processes called).

> **Hint:** The sub-processes called within the Business Process Models will be internally added to the application model during the process cost analysis.

Additionally to the correct modelling of the process and of the structure organization, it is necessary to determine during the process cost analysis the performer assignment for each activity by analogy with the capacity analysis (see chap. 4.3, p. 354).

This performer assignment (see chap. 2.2.12.14, p. 222) is made in the "Performer" attribute of the "Activity" class, in which you will enter an AQL expression (see chap. 14., p. 790). You can also let the performer assignment depend on current variable values.

The following modelling guide lines must be met in the Working Environment model (see fig. 439):

- Each performer of the Working Environment model must be assigned to exactly one cost centre using the "is charged to" relation.
- Several performers can be assigned to one cost centre using the "is charged to" relation.
- Using the "is manager" relation you can define a performer as the manager of a cost centre.

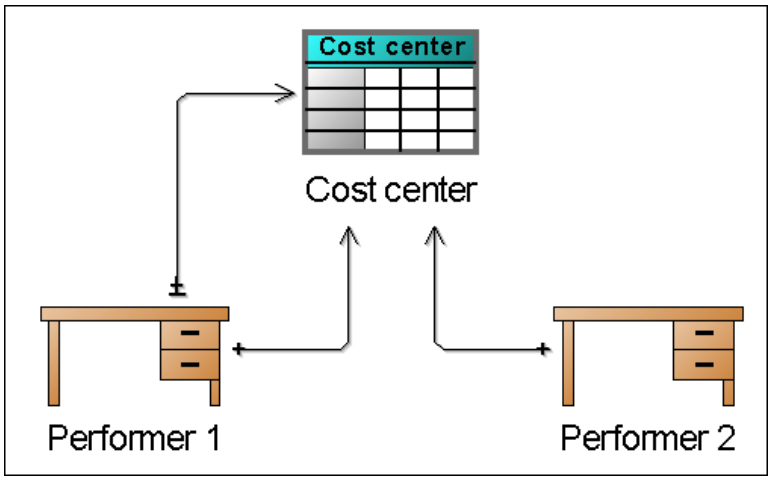


Figure 439: modelling of a cost centre in a Working Environment model

Hint: When you startADOxx-3c, an error message will appear if one of the above defined modelling guide lines is not met .

Resources are not taken into account in this present version of ADOxx-3c Process Cost Analysis component, but can be contained in the Business Process Models and in the Working Environment model without affecting the results

7.1.5 Method of working

To reach the linear relation between cost driver and the amount of cost generated for each process, which is necessary for process cost analysis, a simulation mechanism is used in ADOxx-3c.

7.1.5.1 Simulation

The characteristic feature of processes modelled in ADOxx is their relatively high level of detail. It means that a process should present not only a sequence of activities but it may also contain controlling objects like decisions or parallelisms.

Consequently, a Business Process Model proceeds not only by one predetermined way but, because of "Decision" objects, a variety of different possibilities is available in the business process. In consequence, costs of process cycle, according to the type of the case, may vary. This sometimes impedes or prevents pure analytical calculation of process costs. In order to avoid it, in the AD0NIS-3c component the simulation is available. In this method all the cases in the specified period are taken into consideration and the arising costs are summed up. The same simulation algorithm as in the capacity analysis (see chap. 4.3, p. 354) is used during the process cost analysis.

Within the process cost analysis every business process of the application model is simulated as many times as is defined before the start. The number of simulations will influence their precision. The results are converted by AD0NIS-3c into a realistic number of business processes executed and they are presented within the chosen period. Additional description of the simulation algorithm can be found in the description of the capacity analysis (see chap. 4.3, p. 354) in ADOxx-user's manual.

7.1.5.2 Establishing costs of processes

Calculation of costs of processes within the simulation proceeds through addition of all costs generated in activities. The costs are projected, on the basis of the simulation, according to the real number of the processes executed. Moreover, suitable output ON (output neutral) costs and possible idle costs of processes are added.

During the simulation a number of cases, defined by the user, are executed for each business process from the application model. A case wanders also from one process constructs (i.e. Activity, Process reference) to another.

If an activity is reached in the cycle, analysis of AQL expressions (see chap. 14., p. 790) in the attribute "Performer" are executed. The allocation algorithm selects, from the number of potential performers, a performer for this activity. This choice is made by a random generator taking into consideration availability of all the possible performers. The higher the value in the attribute "Availability" of a performer, the higher the possibility that the performer will be chosen for execution of the activity.

Carrying out an activity by the performer generates costs, which are divided in ADONIS-3c as follows:

- Execution time costs
  - **execution costs** are calculated by multiplying wages per hour of the performer by the execution time of the activity
  - **Time - dependent costs** are calculated by multiplying time dependent costs by the execution time of the activity
- **Activity-fixed costs** are costs generated while executing a single activity. Users can define infinitely many different activity-fixed costs using ADOxx customising. Costs, which are generated while executing a single activity, must be entered.

  Hint: If all activity-fixed costs change, for example, due to inflation rate, they can be raised quickly, by a defined interest rate, using the function "Global change" (see chap. 2.2.14, p. 243) in the Modelling Component.

Total costs, which are generated by executing this activity, are added to an adequate performer and his cost centre.

This procedure is the same for every activity executed in the business process cycle. The sum of all costs, which are generated by all the cases of the process, determines output induced (0l) process costs. Total process costs are calculated by adding output neutral (ON) process costs and idle costs to 0l process costs.

7.1.5.3 Acquiring process costs rate

The quantitative result of AD0NIS-3c process costs analysis is the **process costs rate**. Process cost rates are calculated for all the analysed processes as well as for the referenced sub-processes. AD0NIS-3c determines output induced process costs rate (0l-PCR) as well as overall process costs rate (PCR).

Process costs rate defines average costs for the single execution of the process: `costs rate = ( Process costs ÷ Process quantity )`

Process costs

Process costs result from:

- **Execution costs** (execution time x execution costs for a time unit of a performer)
- **Time dependent costs** (time dependent costs × execution time of the activity)
- **Total activity-fixed costs**,
- **ON costs** (if taken into account) and
- **Idle costs** (if taken into account).

Quantity of the process costs

Quantity of the process costs is variably calculated by AD0NIS-3c after choosing the option **"Cost driver quantity"** in the window "Process cost analysis - Actual analysis" (see fig. 440) optionally in the window "Process cost analysis - Plan analysis".

In general, AD0NIS-3c uses the quantity defined in the attribute "Quantity" of the Process start object of the process as "Cost driver quantity" for calculating process costs rates.

However special application scenarios require, that the value 1 must be written in the attribute mentioned. If you want to calculate process costs rate, the option "Cost driver quantity" must be activated in the start window of the process cost analysis (see chap. 7.1.6.1, p. 504) and the attribute "Cost driver quantity" should be set accordingly.

> **Hint:** You must make sure, that the attribute "Quantity" in the Process start object of the process defined in the application model, has the value higher than 0.
>
> When cost driver quantity as quantity of the process costs is used, the attributes "Cost driver quantity" of the process and sub-processes must have the value higher than 0.
>
> In other case, an error message is displayed by AD0NIS-3c **after** the simulation.

Conversion between process oriented and period oriented results

Every object of the class "Process start" in a Business Process Model has attribute Quantity" (whole number) and "Reference period" (values: "For a day", "For a month" or "For a year". These values define how often a business process is executed in the chosen period. When the value "0" is selected in the process of the application model, the analysis is stopped and the adequate message turns up.

> **Hint:** When the option "Cost driver quantity" is not activated in the start window of the process cost analysis, the number in the attribute "Quantity" in the Process start object of the process is acquired by cost driver quantity of the model. When the option is activated, the value in the attribute "Cost driver quantity" of the Process start object of the process is used

According to these entries, AD0NIS-3c projects the number of business processes started for the company's year. If the process cost analysis should be carried out for a period shorter than one year, the cycle duration can be chosen at the start of ADONIS-3c. The number of processes initiated is calculated for the period accordingly. The cycle duration can last:

- a year
- a quarter
- a month
- a day

Example:

A business process appears ten times in one day. It means that assuming 170 working days during the year, the process appears 1700 times within the year. When ADONIS-3c takes into consideration a quarter period, then the process is initiated 1700/4 = 425 times during the chosen period.

> **Hint:** The attribute "Budget" and "ON fixed costs" of the class "Cost centre" and the attribute "Cost driver quantity" of the class "Process start" relate to the periods chosen in the start window of the process cost analysis and so are also not converted. However, the attribute "Quantity" of the class "Process start" is converted, taking into consideration the "Reference period" for the chosen cycle duration .

7.1.5.4 ON processes and idle capacities

Hint: ADOxx-3c takes into consideration ON processes and idle capacities in the process cost analysis only when suitable settings are activated in the start window.

Output neutral (ON) processes are executed in one cost centre and do not apply for other cost centers. Usually these are administrative procedures, like for example personnel management, planning and organisational activities. However processes, which are not yet mapped in ADOxx by 0l processes, can be also included.

Input data of 0l processes is entered into suitable cost centers according to the place of origin of costs. In AD0NIS-3c the ON-processes can by classified in two ways:

1. **ON times**: overall time needed for all the incidental ON processes, which were executed in the cost centre.
2. **ON fix costs**: If for some ON processes time needed may not or should not be defined, identified costs may also be defined in the cost centre. For example, if certain capacities of the cost centre are integrated with a project, spending for the project may be registered here.

   Hint: To avoid reckoning ON-processes twice, the procedures, for which time needed is defined, should not be additionally evaluated with the amount of money.

Actual process cost analysis:

Actual process cost analysis comes from cost centre budget spent in the considered previous period. Components of budget which were not consumed by 0l processes are assigned to the ON processes and idle capacities `ON costs + idle costs = cost centre budget - OI costs`.

Idle time of a cost centre is calculated as a difference between performer's capacity, which is at the disposal of the cost centre, and time needed for 0l processes and ON processes.

In order to attach proper costs to the ON costs, you need the following calculation:

If you take the assumed budget of a cost centre and subtract the costs of the 0l processes, what you get are the costs that need to be divided between the ON processes and the idle time. You subtract from this sum of money all possible ON fixed costs and add them to the ON processes. The outstanding amount of money is then divided in relation to the time demand (ON and idle times). Finally, you get the costs of the ON processes and the idle capacity.

Plan-process cost analysis:

At the beginning of the Plan process cost analysis (see chap. 7.1.6.1, p. 504) the user can input in the field "Idle times in %" the estimated percentage relation between the idle capacity and the execution time in an 0l process .

Example:

Let the total execution time of all 01 processes in a cost centre amount to 120 days (ADOXX Time Format: 00:120:00:00:00) and let's assume the additional needed idle time amounts to 20%. In that way we get in the process cost analysis an idle time of 24 days (ADOXX Time Format: 00:024:00:00:00). These calculations take place always with the preset company time (see chap. 4.1.3.1, p. 335)in mind.

As far as the Plan process analysis is concerned, AD0NIS-3c assumes that in case of ON and idle times the average hourly cost rate (execution costs and time dependent costs) may be applied as opposed to the Actual process cost analysis. The hourly cost rate is calculated as an average based

on the 0l processes. Further on, this rate is multiplied by the ON and idle times to obtain the ON and idle costs. The ON costs are then expanded by the ON fixed costs.

When the ON and idle costs are already calculated, they will be attached to appropriate processes in the cost centre in both Actual process and Plan process analysis, according to execution time for each process in cost centre.

7.1.5.5 Assessment of the cost centers' capacity

The capacity of a cost centre is obtained by summing up all the performers allotted to this cost centre by relation "is charged to". The percentage availability of a performer in the period of analysis may be set in the ADOxx Notebook of the performer's object in the attribute "Availability". This value is set by default to 100%, which describes a full-time employee.

In the Actual process cost analysis the cost centre's budget is required as input and is obtained from the budget attribute of the cost centre object.

The cost centre's budget required for the next period results from the Plan process cost analysis. It is made up of the 0l processes' costs and the costs generated by ON processes and idle capacities.

7.1.6 Carrying out the process cost analysis

To start the process cost analysis please follow these steps:

1. Switch to the Evaluation Component by clicking the smart icon .

   Alternatively, you may activate the Evaluation Component by right-clicking the component bar (to the right of the components' smart icons), which activates a popup menu, and selecting the "Evaluation" item. You can also activate the pop-up menu using the <F9> and then press the <e> key to trigger the Evaluation Component.

2.

   - If you want to carry out the **Actual process cost analysis** please select the item "Actual analysis" from the "Cost-Cutting component" menu. Alternatively, you may click on the smart icon in the quick-access bar.
   - If you want to carry out the **Plan process cost analysis** please select the item "Plan analysis" from the "Cost-Cutting Component" menu. Alternatively, you may click on the smart icon in the quick-access bar.

This will cause a "Cost-Cutting Component - Actual analysis" (see fig. 440) or a "Cost-Cutting Component - Plan analysis" window to show up, where you can start your process cost analysis (see chap. 7.1.6.1, p. 504).

7.1.6.1 Starting the process cost analysis

Select from the list "Application model selection" the application model (Business process models and Working environment model) that you want to use as input for the process cost analysis (see chap. 7.1.4, p. 499) By clicking the "Add" button you may define new application models (see chap. 2.4.16.2, p. 285). You may refresh the list by clicking the "Refresh" button.

In the "Cost-Cutting Component - Actual analysis" (see fig. 440) or "Cost-Cutting Component - Plan analysis" window please submit the following data :

- Numbers (see p. 505)
- Period length (see p. 505)
- Settings (see p. 506)

Confirm the submission by clicking the OK button.

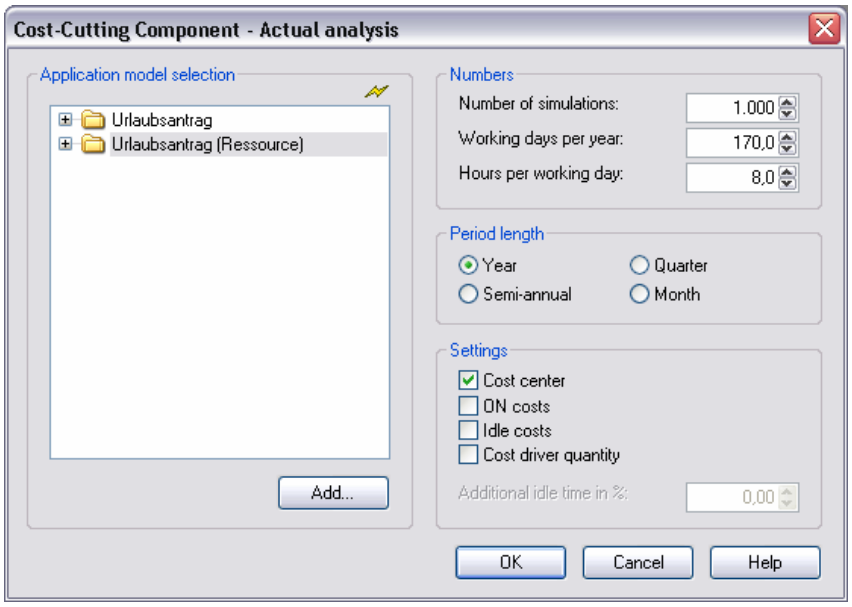


Figure 440: Starting process cost analysis

After clicking the OK button, the library definitions for process cost analysis will be checked.

If the library definition has no assigned values or contains improper syntax, you will be presented with an appropriate message box and the process cost analysis will be carried out with standard values. Please contact your ADOxx administrator in such a case.

Numbers

Number of simulations — Specifies how many times each application model in the business process should be simulated. Number of simulations is an indicator of accuracy of the simulation, which means that the higher this number is, the more precise the results are (and the more time the simulation takes up).

Working days per year — This value combined with the "Hours per working day" value is necessary to convert real time into company time (see chap. 4.1.3.1, p. 335).

Hours per working day — This value combined with the "Working days per year" value is necessary to convert real time into company time (see chap. 4.1.3.1, p. 335).

Period length

Period length specifies how long should be the examined time period.

The basis for period length is the company year, specified in the "Numbers" dialogue box through "Working days per year" and "Hours per working day" variables. AD0NIS-3c processes the available period lengths internally as follows:

- year = company year
- half-year = company year ÷ 2
- quarter = company ÷ 4

- month = company year ÷ 12

Hint: The "Cost centre" class1 attributes "Budget" and "ON fixed costs" and the "Cost driver quantity" attribute of the "Process start" class refer to the period chosen by you before and therefore do not get converted. Whereas the attribute "Quantity" of the "Process start" class is converted taking into consideration "Reference period" for the chosen cycle duration.

Settings

Cost centre Default setting: activated.

By deactivating this option you exclude the cost centers, with activity's performers assigned to them from the calculation. That means the total cost of a process will not include ON nor idle times. This option may be altered only by "Actual process cost analysis". By "Plan process cost analysis" it is impossible to deactivate "Cost centre" option.

ON costs Default setting: activated.

This option may be altered only with "Cost centre" option active. By activating this option, you include in the calculation the costs and the possible ON fixed costs related to ON times.

Idle costs Default setting: activated.

This option may be altered only with "Cost centre" option active. By activating this option, you include the costs related to idle times in the calculation.

Cost driver quantity Default setting: deactivated.

Activate this option to substitute the real quantity of business processes' cycles with the cost driver quantity submitted in the Process start object.

Additional idle time in % Activated only by "Plan process cost analysis". You can provide here the assessed percentage relation of execution to idle times (see chap. 7.1.5.4, p. 503). For example, if you expect the idle times in the analysed period to amount to 20% of the 0l processes' execution times input 0.2 in this field.

7.1.6.2 Simulation

Loading of additional necessary models

If not all models necessary for the process cost analysis are loaded, the missing ones are opened in **read-only** mode. When the process cost analysis is done, that is after results windows are all closed, you will be asked whether the models loaded exclusively for the purpose of the simulation should be closed.

If you encounter an error when loading models, please refresh the model lists in the "Cost-Cutting Component - Actual analysis" (see fig. 440) (or "Cost-After the additional necessary models get loaded, all simulation models are transferred into an internal view. It is displayed in the title of the status window.

Simulation errors

If an error occurs during the simulation, an error message box shows up. Close the message box by clicking on the OK button or by pressing enter key. After that, the object that caused problems is - if it's possible - displayed in the foreground and graphically marked. Solve the problem and start the process cost analysis again.

Hint: The simulation error messages (see chap. 8., p. 520) their causes and solutions are described separately.

Status window

Depending on the number of simulations, the whole process may take up some time. You are informed about the progress of the simulation in the status window.

You can stop the process cost analysis by clicking the Cancel button and accepting the pop-up message box. In such a case no results are displayed.

After the simulation

This is when the ADONIS-3c process cost calculation takes place. This process may take some time, depending on the number of simulations and the size of the model. The approximate time of calculation is displayed

Calculation errors

If an error occurs during the calculation, a question box shows up. You may choose to stop the calculation and leave the process cost analysis by clicking on the "Yes" button or continue the calculation by clicking on the "No" button.

Hint: If you decide to continue, some of the results may be incorrect!

If the total of execution costs, time dependent costs, "activity-fixed" costs and ON costs **exceeds the cost centre's budget**, you will be presented with an appropriate hint.

To solve the problem you may try the following changes:

- Decrease one of these costs:
  - execution costs,
  - time dependent costs,
  - "activity-fixed" costs.
  - ON costs
- Increase - if it's possible - the cost centre's budget or decrease the process quantity.

If the capacity available in the cost centre is insufficient to carry out the 0l or ON processes, you will be presented with an appropriate hint .

To solve the problem you may try the following changes:

- Decrease the execution times of the particular processes.
- Decrease the quantity of processes.
- Increase the capacity of a cost centre by adding performers or rising the availability values.
- Decrease the ON times
- Increase - if it's possible - the "Hours per working day" or "Days per year" values.

Hint: of the process cost analysis, their causes and solutions are described separately. (see chap. 8., p. 520)

© BOC Group 2008

7.1.6.3 Presentation of results

When the calculations are done, the "Cost-Cutting Component - Actual analysis -Reports" (see fig. 441) (or "Cost-Cutting Component - Plan analysis - Reports") results window is displayed.

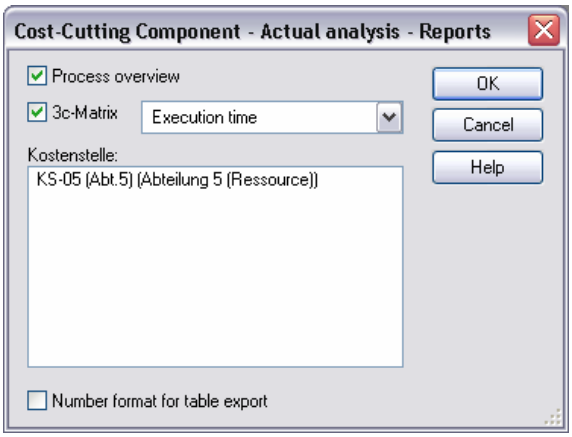


Figure 441: Process cost analysis - Report selection

In this window you can select which reports should be opened.

You can choose from the following:

Process overview Default setting: active.

Lists all processes of the application model and their sub-processes with relevant data.

3c-Matrix Default setting: deactivated.

Clearly presents the relations between cost centers and processes. You can choose an attribute (for example, a type of costs or execution time) and in this way analyse the places where costs of a particular type are generated in a process. When the 3c-Matrix option is active, you can choose an attribute in the select box located to the right.

Cost centre Default setting: deactivated.

Presents the cost centre with all relevant data. The list contains all the cost centers that were used in the simulation of an application model. In the list you can choose as many cost centers as you wish.

Number format for table export

Default setting: deactivated.

When this option is deactivated the particular reports in the ADOxx Browser (see chap. 8., p. 66) present the data aligned to the right with a decimal point as separator. Such a presentation guarantees good readability on the screen but makes it difficult to process the data in case of export to a spreadsheet application. The numbers are then treated as text and cannot be used for further calculations. Activate this option to enable the processing of the data in a spreadsheet application.

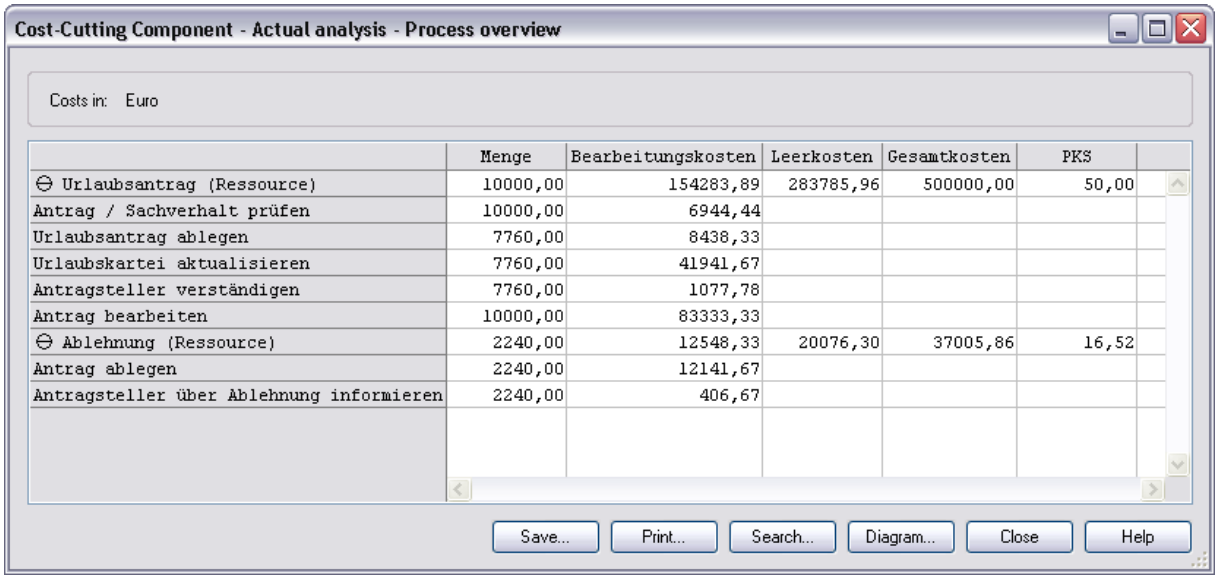
Hint: It is recommended not to open all available reports at once because too many windows could make your desktop unclear. The "Cost-Cutting Component - Actual analysis - Reports" (see fig. 441) (or "Cost-Cutting Component - Plan analysis - Reports") window remains open until the end of the process cost analysis, which means you always have the possibility to open additional windows or reopen the ones you closed.

Process overview

In the process overview you are presented with all the processes that were analysed in the simulation. Process cost rates and all other data useful for process cost analysis is displayed for each process.

The currently used currency and possibly an amount are displayed over the table. You may define the currency using ADOxx customising.

The results are displayed in the ADOxx-Browser (see chap. 8., p. 66) in a "Cost-Cutting Component - Actual analysis - Process overview" (see fig. 442).



	Menge	Bearbeitungskosten	Leerkosten	Gesamtkosten	PKS
⊖ Urlaubsantrag (Ressource)	10000,00	154283,89	283785,96	500000,00	50,00
Antrag / Sachverhalt prüfen	10000,00	6944,44			
Urlaubsantrag ablegen	7760,00	8438,33			
Urlaubskartei aktualisieren	7760,00	41941,67			
Antragsteller verständigen	7760,00	1077,78			
Antrag bearbeiten	10000,00	83333,33			
⊖ Ablehnung (Ressource)	2240,00	12548,33	20076,30	37005,86	16,52
Antrag ablegen	2240,00	12141,67			
Antragsteller über Ablehnung informieren	2240,00	406,67			

Figure 442: Process cost analysis - Process overview

You can show and hide particular levels, search for contents, present the results graphically, save and print them.

The results in the process overview contain the following structuring elements:

- Business processes
- Activities

In the ADOxx-Browser all business processes are displayed at the beginning of the list, followed by their referenced sub-processes.

The results of the processes are displayed in an aggregated form, which means that for example, the total costs of a business process contain also the costs of its referenced sub-processes.

Following information and data is displayed for items of the "Business process" type:

- Cost driver
- Cost driver quantity
- Quantity
- Execution time
- Execution costs
- Time dependent costs
- Activity-fixed costs

© BOC Group 2008

- Total fixed costs
- ON costs
- Idle costs
- Total costs
- 0l process costs rate
- Process costs rate

Following information and data is displayed for items of the "Activity" type:

- cost driver
- Quantity
- Execution time
- Execution costs
- Time dependent
- Activity-fixed costs
- Total fixed costs

Hint: The information and data mentioned above (i.e. the columns within the table) for the items "Business process" and "Activity" can be hidden for each column by selecting the menu item "Hide this column" from the context menu of this column or can be hidden or unhidden by selecting the menu item "Select columns" from the context menu.

Cost centre report

For each cost centre, that was involved in process execution, a report is created.

The presentation of results is carried out in ADOxx-Browser (see chap. 8., p. 66) in the "Cost-Cutting Component - Actual analysis - < *Cost centre name*>" (see fig. 443).

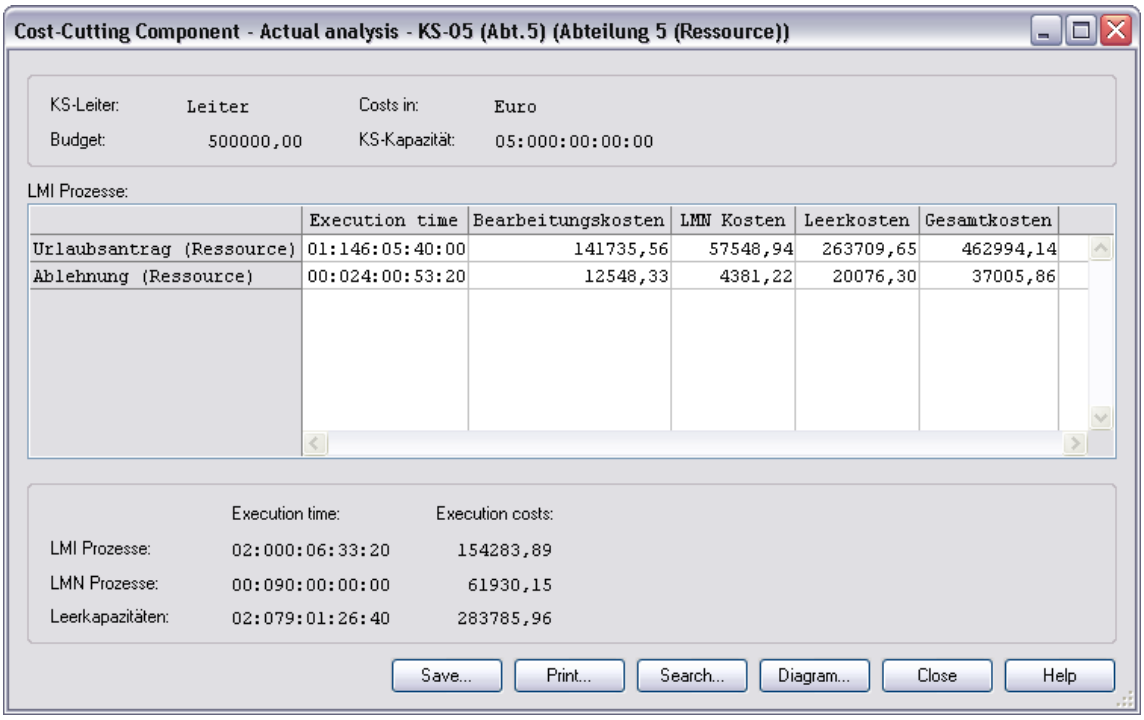


Figure 443: Cost-Cutting Component - cost centre report

Within the cost centre report you can show or hide single layers and save, print or present the results in a graphical form.

The "Cost-Cutting Component - Actual analysis - <*Cost centre name*>" (see fig. 443)) window consists of three category groups:

- In the upper part of the window you can find main information about cost centre (information part)
- ADOxx-Browser (see chap. 8., p. 66) which contains the list of all processes executed in the cost centre, is located in the centre part (browser part)
- The lower part of the window displays the costs assigned to 0l processes and idle capacity

The information part includes:

CC manager Cost centre manager; performer's name, who is connected with cost centre through "is cost centre manager" relation.

Cost in Displays used currency and the possible measured value (e.g. thousand euro). The currency can be defined using ADOxx-Customising.

Budget Is overtaken for Actual process cost analysis from cost centre class object. Budget for Plan process cost analysis is the result of the preceding evaluation.

CC capacity Cost centre capacity; cost centre capacity is determined, similar to calculating of the cost centre in the cost-cutting component - cost centre calculating Cost-Cutting Component - Cost centre calculating (see chap. 7.1.3, p. 499) from the user defined values.

In browser part in first sentenceADOxx-Browser are presented those 0l processes, in which performer, assigned by cost centre, executes at least one activity in the simulation. The processes of the application model, including referenced sub-processes, are presented at the beginning of the list. Opposed to the process overview, the value of processes is not displayed aggregated in this case.

The results in process overview contain items of "Business process" type. The following information is displayed for each business process:

- Cost driver
- Execution time
- Execution costs
- Time dependent costs
- Activity-fixed costs
- Total fix costs
- 0l costs
- Idle costs
- Total costs

The lower part of the window contains:

0l processes

Total execution time and execution costs of 0l processes.

ON processes

time required for ON processes in the course of period (which is specified in cost centre notebook) and assigned costs including possible 0l fixed costs.

Note: Execution time and execution costs will be displayed only if the proper settings (see fig. 440) in the " Cost-Cutting Component - Actual analysis (see p. 506) window are chosen.

Idle capacity

Idle time and assigned costs.

Note: Execution time and execution costs will be displayed only if " (see fig. 440) proper settings (see p. 506) in the "Cost-Cutting Component - Actual analysis window are chosen.

3c-Matrix

The 3c Matrix presents, in chart form, relationships between processes and cost centers. This gives a fast and clear overview of the main process cost areas.

The presentation of the results in the ADOxx-Browser (see chap. 8., p. 66) is carried out in the "Cost-Cutting Component - Actual analysis - 3c Matrix" (see fig. 444).

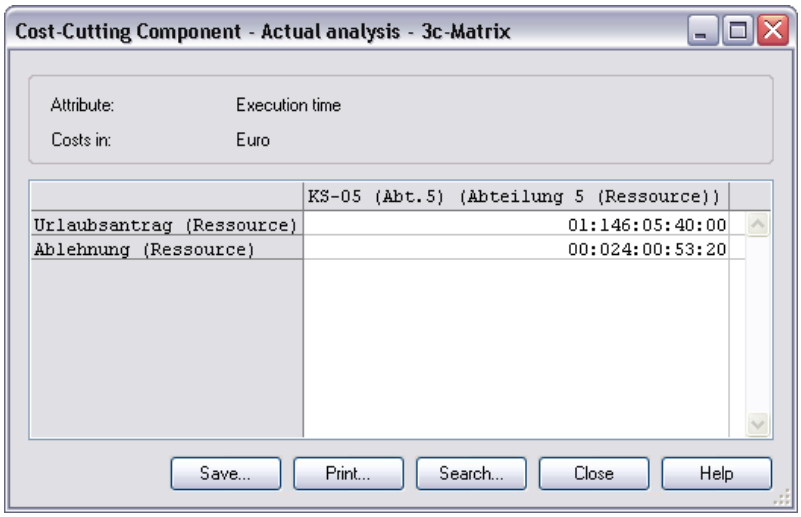


Figure 444: Cost-Cutting component - 3c Matrix

You can search for a content or save and print the results.

In the upper part of the window you can find an attribute name, whose value is shown in the chart cells. Used currency and the possible measured value is specified in the next row. The currency can be defined using ADOxx-Customising.

The processes are listed in the ADOxx Browser and cost centers are displayed in the chart columns. The processes of the application model, including referenced sub-processes, are presented at the beginning of the list. Opposed to the process overview, the values of processes are not displayed aggregated in this case.

7.1.6.4 Ending of the Cost-Cutting Component

Close ADONIS 3c, by clicking the "Close" button in the "Cost-Cutting Component - Actual analysis - Reports" (see fig. 441) (or "Cost-Cutting Component - Plan analysis -Reports")window.

A window shows up, which informs you, that all simulation results will be deleted when closing ADONIS 3c window.

Click "Yes" to close the window and finish Cost-Cutting Component. If you click "No", the "Cost-Cutting Component - Actual analysis - Reports" (or "Cost-Cutting Component - Plan analysis - Reports") window will reopen. After closing ADONIS 3c, you will be asked, if the models, that were loaded exclusively for Cost-Cutting Component analysis, are to be closed, too. Be aware, that the additionally loaded models were opened in read-only mode.

Click "Yes" to close the window and the models that were loaded for simulation. When you click "No" you will close the window but all loaded models will remain in **read-only** mode.

7.2 Personnel and capacity management

The quantity of running processes fluctuates all the time (from period to period) and the personnel in organisational units are subject to ongoing changes. This is also true when it comes to other important company data.

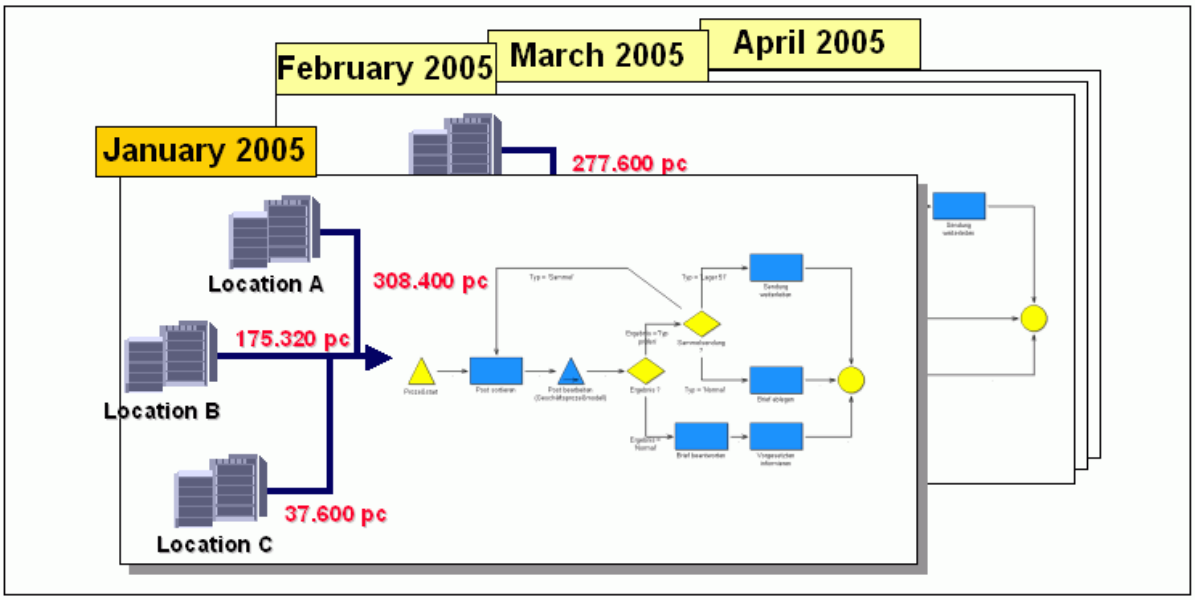


Figure 445: Dynamics in personnel and capacity management

These dynamics have to be also included in personnel management arithmetic. ADOxx offers with its personnel and capacity management components, a tool for personnel and capacity planning in a dynamic environment. The personnel and capacity components allow you to calculate period's comprehensive personnel key indicators, which take into account extraordinariness of each period.

Business Process Model, process quantities and company organisation structure are versioned, modelled and compared with each other (see fig. 446). The personnel and capacity management components acquire adequate data for each period and analyse them with help of the ADOxx simulation mechanism.

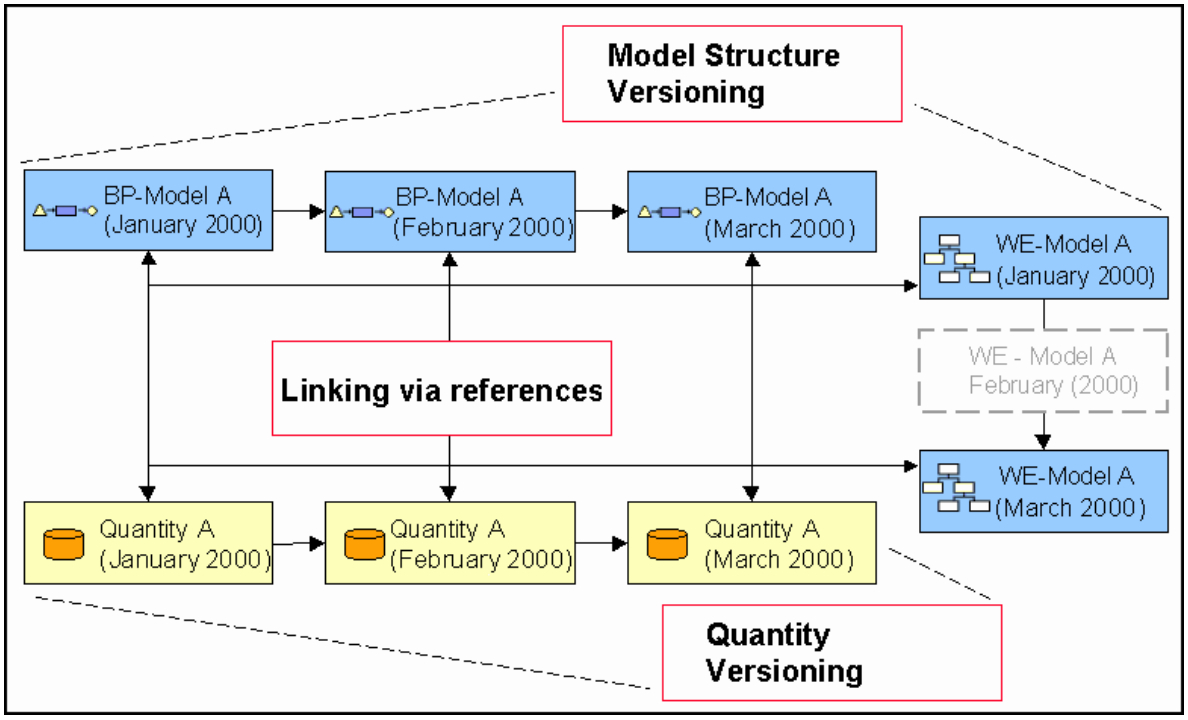


Figure 446: Personnel and capacity management concept

ATTENTION: Comprehensive period analysis requires time-related versioning of models (see chap. 10., p. 92).

Typical results of personnel and capacity management components are:

- Execution quantities
- Demand for personnel per organisation unit and period
- Productivity key indicators (e.g. how much of the process execution time is used in unproductive way)
- Gross and net execution time
- Personnel cost
- etc.

Hint: The personnel and capacity management components are mostly customisable. Which performance indicators are actually evaluated and what evaluating prescripts are used during assessment, depend on the actual configuration of the users application library. Thus it is possible to adapt personnel and capacity management components to internal controlling and management systems established in a company.

A Differentiation between personnel and capacity management components simulation algorithms:

The simulation algorithms (see chap. 4.3, p. 354) allow determining of basic personnel key indicators. Although personnel and capacity management components make use of internal simulation's mechanisms, they offer more possibilities. Moreover personnel and capacity management components support additional **period comprehensive analysis** and allow **comparing key indicators**.

ATTENTION: The personnel and capacity management components are not included with the standard ADOxx configuration.

7.2.1 Starting Analysis

Select the Evaluation Component (see chap. 5., p. 433) and choose "Capacity management" from the "Evaluation" menu to display "Capacity management".

In the field "Application model selection" choose the Application model (see chap. 2.4.16, p. 284), that is to be evaluated.

Hint: Only sequence version specific application models can be analysed.

Set in the field "Numbers" values, known from the capacity analyse (see chap. 4.3.1.1, p. 356).

If you activate the "Reproducible attitude" field, the personnel and capacity management component will always supply the same results by equating input parameters. If You do not activate this field, the results could insignificantly fluctuate due to specifics of the simulation's mechanisms.

Select from the "Analysis period" group required start and end point of analysis.

Click "OK" to start analysis. If necessary, models that are to be analysed will be automatically loaded before analysing process.

7.2.2 Result display options

"Capacity management - Key indicators <*Application model name*>" window allows you to present the calculated personnel key indicators in different views by choosing the following options: Capacity management - Key indicators (Views) and Capacity management - Key indicators (Grouping).

Presentation of key indicators ("Views" tab)

Process oriented:
The presentation is carried out for business processes, structured by activities and then by performers.

Working environment oriented:
The presentation is carried out for performers, structured by business processes and then by activities.

Reference to key indicators ("Views" tab)

Period of time:
Select key indicator you would like to be presented from the evaluated periods of time. If you have more than one period evaluated, you can choose one of the following options: "All periods (average)" or "All periods (comparison)".

Reference to process flow:
The key indicators will be referenced at a time of a single process flow (e.g. personnel cost for a single execution of "Reporting a loss by a telephone" business process.)

Reference to the total period of time:
The key indicators will be referenced to the total selected periods of time (e.g. personnel cost for all "Reporting a loss by a telephone" business processes in June 2001.)

Process object grouping ("Grouping" tab)

Class:
Select a grouping class for activities. Activity results will be displayed aggregated accordingly.
Hint: Your current ADOxx configuration determines whether or which grouping classes are available.

Relation:
Select the relation class, which makes the connection between activities and the chosen grouping class.

Working environment grouping ("Grouping" tab)

Class:
Select grouping class for performers. Performers' results will be displayed aggregated accordingly.
Hint: Your current ADOxx configuration determines whether or which grouping classes are available.

Relation:
Select the relation class, which makes connection between performer and the chosen grouping class.

Furthermore there is "Info" button available. When you click this button, actual analysis parameters will be displayed.

7.2.3 Presentation of results

The key indicators will be displayed in the ADOxx Browser (see chap. 8., p. 66).

The particular processes and organisation structure are reflected in cells. Columns contain key indicators.

Apart from the common functions of ADOxx Browsers (see chap. 8., p. 66) Pop-up menues allows you to show or hide single columns and also sort the entire chart in increasing or decreasing order by selected key indicator.

7.3 case/4/0 interface

Case/4/0 is a CASE-Tool, which was developed and is distributed by **microTOOL GmbH**. It has been proved and widely used in developing mainframes and client/server applications by integrating analysis and design system tools. It offers support for different programming languages in generation of the code and Reverse Engineering. Furthermore case/4/0 includes publisher, which generates documentation prepared for Intranet/Internet publishing.

To use case/4/0 interface, you must meet following requirements:

- The relevant component accesses have to be activated.
- The "External coupling" library attribute must be appropriately configured. It describes, from which class references case/4/0 objects are allowed. This configuration is carried out by your ADOxx consultant.
- case/4/0 (Version 4.3 or higher) must be installed properly.

  Hint: case/4/0 is available for Windows NT and Windows 95/98 only.

7.3.1 Interface concept

An online - connection between Business process model in ADOxx and IT - functions and data structure of system analysis in case/4/0 interface (see fig. 447) is established in this case. The connections are available and operate in both directions. Through the m :n assignment it is possible to achieve maximum flexibility in process modelling. During process modelling, user is able to indicate, which functions of system applications support each technical activities and which data is relevant for the implementation.

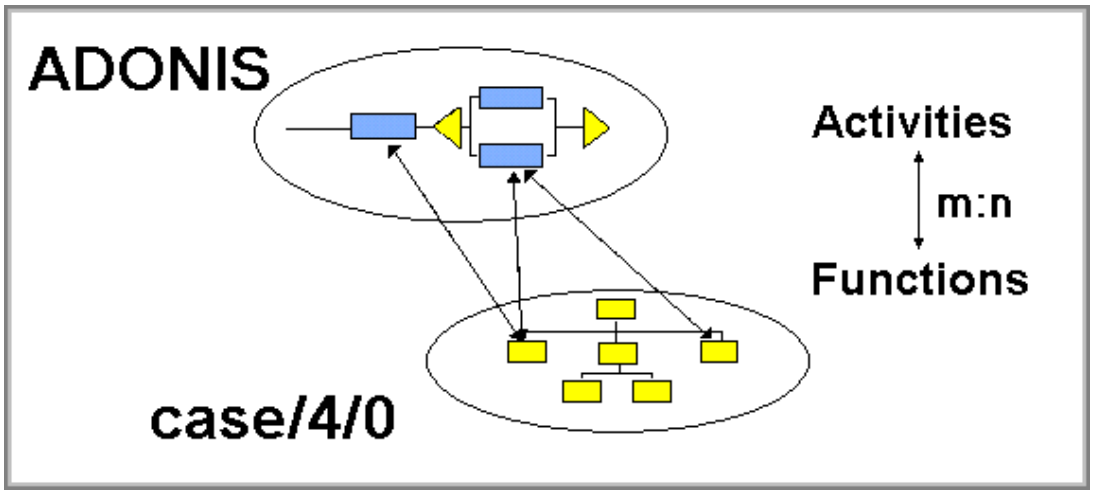


Figure 447: Interface concept

In the following "Reference" means always a reference between ADOxx and case/4/0 and "Object" means an object in ADOxx.

Reference management (creating, modifying and deleting) is conducted only in ADOxx. It goes always out of the object, that belongs to the class for which references are allowed. The existing references can be tracked by jumping to the object in case/4/0.

Reference management is carried out always based on the initial object. If you want to manage object references or create a new reference, you have to click the right mouse button on that object in the drawing area and then select from the context menu the "case/4/0" option. This option is available only for objects, from a class, for which references are available.

The analysis of references in other directions, i.e. starting from case/4/0 is also possible. You can for example, ask in which business processes and activities are application system's functions and data used. For more details see case/4/0 user's manual documentation.

After selecting "case/4/0" option from the context menu the "*<Activity name>* - case/4/0" window will be displayed.

You can then, by clicking:

"Track" track (see chap. 7.3.2, p. 518) pre-selected reference to case/4/0 object,

"New" create new (see chap. 7.3.3, p. 519) reference to case/4/0 object,

"Change" change (see chap. 7.3.4, p. 519) pre-selected reference to case/4/0 object,

"Delete" delete (see chap. 7.3.5, p. 519) pre-selected reference to case/4/0 object.

7.3.2 Tracking references

While tracking a reference the case/4/0 model which contains the referenced object will be displayed in the foreground. Mark in "*<Activity name>*" the window, those references that are to be tracked, and click the "Track" button. Alternatively you can double-click on the relevant entry on the list.

Case/4/0 will be activated and the window with referenced object will be displayed or opened in foreground. The dialogue window will be closed.

7.3.3 Creating new references

To create a new reference click "New" in the "<*Activity name*> - case/4/0" window. The "case/4/O-System selection" window will be displayed. Select a system from the list and then click "OK".

> **Hint:** If case/4/0 interface is not running the case/4/0 login (see chap. 7.3.3.1, p. 519) window will appear.

The "case/4/0 function references" window shows up, in which you can select the referenced object.

Select an object from the list and then click "OK".

The reference is now created and is indicated in the "<*Activity name*> - case/4/0" window on the "Applied references" list.

7.3.3.1 case/4/0 login

If the case/4/0 interface is not running, when attempting to create (see chap. 7.3.3, p. 519), change (see chap. 7.3.4, p. 519) or delete (see chap. 7.3.5, p. 519) references the "case/4/0 login" window will appear.

Enter login name and correct password and then click "OK" to proceed with login process .

7.3.4 Changing references

By changing references you replace existing references with new ones. Mark in "<*Activity name*> - case/4/0" the window the reference that is to be replaced and click the "Change" button (this button is active only when one reference is properly marked).

Select in the "case/4/0 System selection" the desired system and finally, in the "case/4/0 Function references" window, select a relevant object from the list.

New reference will be displayed in the "<*Activity name*> - case/4/0" window on the "Applied references" list, instead of the old reference.

7.3.5 Deleting references

To delete references, select them from the references list the in the "<*Activity name*> - case/4/0" window and then click "Delete".

> **Hint:** If case/4/0 interface is not running, the case/4/0 login (see chap. 7.3.3.1, p. 519) window will appear.

Selected references will be deleted and erased from the "Applied references" list in the "<*Activity name*> - case/4/0" window.

8. Error Messages

You will find a detailed description of each error message (including cause and action) in:

- the context menu for each error message by clicking on the help button in the window of the error message,
- in the online documentation (menu item "ADOxx help" in the menu "Help") in the chapter "Error messages",
- in the PDF documentation in the chapter "error messages" (PDF files are to be found on the ADOxx installation CD in the directory \BOOKS).

This chapter lists the error messages that may occur in the ADOxx Modelling Toolkit. The error messages are divided across the following groups:

aacquiui	**aactana**	**aalkwins**	**aanabrw**
aanactrl	**aanadlg**	**aanaeval**	**aanaud**
aap	**aapedit**	**aappmod**	**aapqury**
aapview	**abmpsup**	**abrsimpl**	**acaledit**
acalui	**acard**	**acase40**	**acccber**
acccdlg	**acccinit**	**acnumchk**	**acoexhan**
acoexman	**acoexpar**	**acompare**	**acomps**
aconfui	**acregchk**	**adbacc**	**adbadm**
adbsess	**adistrib**	**adocase**	**adynattr**
adynpred	**aeadsply**	**aeagents**	**aeagnrl**
aearslt	**aeasubag**	**aevalrel**	**aevkres**
aexpappl	**aexpdlg**	**aexplore**	**aexport**
aexpsort	**afile**	**agdt**	**agenpar**
aglochan	**agogo**	**agolay**	**agraph**
ahaconv	**ahagrob**	**ahamot**	**ahanote**
aicondef	**aimport**	**aleo**	**alib2sgm**
alibload	**alogin**	**amoddiff**	**amodlru**
amodraw	**amodtab**	**amomain**	**amoshell**
anamegen	**anbbrow**	**anotebk**	**aoffanim**
aoutgen	**apercalc**	**aperctrl**	**aperout**
aperpar	**aperress**	**apersui**	**aplangen**
apredct	**aprint**	**aproobjs**	**apwchgui**
areena	**aregex**	**arelrdef**	**areposui**
aresattr	**as2rdlg**	**ascript**	**asetref**
asetsel	**asgmlexp**	**asik**	**asim1dlg**
asim1end	**asim2dlg**	**asimmap**	**asimobj**
asimtext	**asrchmod**	**asrchtlb**	**astarth**

astdcfg	**asubsrch**	**asuibrw**	**avedbox**
averdate	**awrjpeg**	**axml**	**hacevalacq**
hacexp	**hacfile**	**hacimp**	**hacini**
hacinit	**svxwins**		

An error message consists of an error code (in square brackets - []) that is shown at the beginning of the error message and of the message's text (below the error code). The error code is a combination of letters and numbers. The letters represent the error group, the numbers serve to number the messages within an error group. The error messages are listed in alphabetical order according to groups and in ascending order inside each group. Should you have problems finding an error message, please use the online help or check the index.

Hint: The list of errors is ordered alphabetically according to error groups and the groups are numbered in ascending order. The numbering need not necessarily be continuous.

You can close the window containing the error message by clicking on the OK button or by pressing the "Enter" key. Then execute the actions described under "Action".

[aacquiui-02]

Message: The acquisition module HOMER could not be started.

Maybe HOMER has not been installed correctly.

Cause: The file "acqui.dat" in the ADOxx installation directory does not contain a valid path for HOMER or HOMER was not installed in due form respectively.

Action: Make sure that HOMER is installed properly (e.g. try to start HOMER via the "Start" menu of the operating system). If HOMER starts withot error that way, the acquisition module is installed properly (If this produces an error too, the installation procedure was faulty and has to be repeated). Correct the call path for "homer.exe" in the file "acqui.dat".

[aactana-01]

Message: Fatal error during activity analysis!

Delete the simulation cache and restart the simulation.

Cause: A fatal error has occurred from which the system cannot automatically recover.

Action: Try to run the simulation again. If the same error re-occurs, then export your models to an ADL file and contact your ADOxx consultant.

[aalkwins-01]

Message: Invalid filename.

Cause: An invalid filename, the name of a write-protected file or a directory that does not exist has been specified for the import file.

Action: Enter a valid name of a file that is not wirte-protected in an existing directory.

[aalkwins-02]

Message: Enter a filename.

Cause: You have not specified a filename for the ADL file to be imported.

Action: Enter the filename of the ADL file.

[aalkwins-08]

Message: The models contained in the application model "<*model name*>" have been deleted. Therefore this application model is no longer valid.

Cause: The application model you have specified for export no longer contains any models.

Action: Specify a different application model which contains the models you require.

[aalkwins-10]

Message: Username must be at least three characters!

Cause: When renaming a username during UDL import, a name of less than three characters was specified.

Action: Please enter a new username which is at least three characters long.

[aalkwins-11]

Message: An unknown model type has been used!

Cause: For the export using AdoScript an unknown model type was provided. The export of models of this type is not possible.

Action: Use the name of an existing model type in the AdoScript.

[aanabrw-01]

Message: Error during the copy: All values could not be pasted!

Cause: An error has occurred when pasting in the ADOxx browser. This is why the pasting process has been cancelled.

Action: Verify whether the area to be pasted is not to large and whether the attribute types go together. Also check whether it is allowed to be pasted in the selected area.

[aanabrw-02]

Message: Error during the copy: No value could be pasted!

Cause: An error has occurred when pasting in the ADOxx browser. This is why the pasting process has been cancelled.

Action: Verify whether the area to be pasted is not to large and whether the attribute types go together. Also check whether it is allowed to be pasted in the selected area.

[aanactrl-02]

Message: The AQL expression is syntactically not correct!

Cause: The AQL expression you have entered does not correspond with the defined AQL syntax.

Action: Enter a syntactically correct AQL expression.

[aanadlg-04]

Message: The models to be analysed must be defined on the same application library.

Cause: The models selected for analysis are based on different application libraries.

Action: Select a number of models to be analysed which are based on the same application library.

[aanadlg-05]

Message: An error has occurred during the storage!

Cause: An error has occurred during the storage of the model information of the models analysed.

Action: The data medium is possibly write protected (e.g. floppy disk).

[aanadlg-06]

Message: The entered value "<*value*>" is not valid!

<*Format*> is expected.

Cause: While creating a standardised query, you have entered a value which is not compatible with the selected attribute.

Action: Enter a value compatible with the entry type.

[aanaeval-01]

Message: The input parameters for the analytic evaluation are corrupt. The evaluation cannot continue.

Cause: There is a problem with the sim mapping specifications.

Action: Contact your ADOxx administrator. The sim mapping specified within the application library contains errors. This can be corrected within the library management component of the ADOxx Administration Toolkit.

[aanaeval-02]

Message: Enter valid values in all fields of the group "Filter".

Cause: An invalid value exists in at least one field of the group "Filter".

© BOC Group 2008

Action: Enter a valid value or else deactivate the group "Filter".

[aanaeval-03]

Message: The given filter doesn't apply to any one object. Change the "Filter" settings.

Cause: The required action cannot be carried out as no object (Activity) can be found to match the filter settings.

Action: Enter a valid value or else deactivate the group "Filter".

[aanaeval-04]

Message: Either no simresult mapping is defined or the mapping defined is invalid. Adopting the evaluation results is therefore not possible.

Cause: Simresult mapping is either invalid or not defined.

Action: Contact your ADOxx administrator. The "simresult mapping" must be corrected in the Administration Toolkit.

[aanaeval-05]

Message: There are no input parameter defined. The analytic evaluation cannot be carried out.

Define an input parameter configuration in ADOxx Administration Toolkit or contact your ADOxx administrator.

Cause: The Sim-Mapping is deficient or not defined.

Action: Contact your ADOxx administrator. The Sim-Mapping defined in the ADOxx Administration Toolkit must be corrected.

[aanaeval-06]

Message: The model contains no object, for which results can be calculated.

Cause: The analytic evaluation has been carried out with a model which contains no object of the class Activity.

Action: Carry out the analytic evaluation with a model which contains at least one object of the class Activity.

[aanaeval-07]

Message: Some of the models to be evaluated are loaded with a write protection. The automatic transfer of the evaluation results to the models would lead to inconsistencies because of the mix of results and therefore is not possible.

The evaluation will be cancelled.

Cause: The analytic evaluation has been started via an AdoScript. Then the option of the automatic storage of the evaluation results to the models has been activated. (Note: The option has been possibly activated automatically.) When this option is activated, it

must be possible to load all the models to be evaluated (main or submodels) with write access. If this is not the case, the evaluation cannot be continued.

Action: Make sure that all models to be evaluated can be loaded with write access. For instance, you can manually load the models with write access before starting the evaluation.

[aanaud-01]

Message: Error in application library.

Cause: The query definitions in your application library contain errors.

Action: Contact your ADOxx administrator or your ADOxx consultant.

[aanaud-03]

Message: Error in library attribute "<*attribute name*>":

<*Number*>. query:

Unknown model type ("<*model type name*>")!

Cause: An unknown model type has been used in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-04]

Message: Error in library attribute "<*attribute name*>":

<*Number*>. query:

Invalid context!

Cause: A key word has been used in a wrong context in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-05]

Message: Error in library attribute "<*attribute name*>":

<*Number*>. query:

Unknown key word ("<*Name*>")!

Cause: An unknown key word has been used in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-06]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Invalid menu item!

Cause: The menu item is missing in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-07]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Invalid query!

Cause: The query text is missing in the query in the above-mentioned library attribute .

Action: Contact your ADOxx administrator.

[aanaud-08]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Invalid AQL expression!

Cause: A syntactical invalid AQL expression is used in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-09]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Unknown class ("<*class name*>") in the <*context*>!

Cause: An unknown class is used in the query in the above-mentioned library attribute. The context can be the input fields or the result attributes.

Action: Contact your ADOxx administrator.

[aanaud-10]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Unknown attribute ("<*attribute name*>") in the <*context*>!

Cause: An unknown attribute is used in the query in the above-mentioned library attribute. The context can be the input fields or the result attributes.

Action: Contact your ADOxx administrator.

[aanaud-11]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Key word missing("<*Name*>")!

Cause: The named key word is missing in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-12]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Context missing!

Cause: A required context is missing in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-13]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
Input field empty!

Cause: The input elements are missing in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-14]

Message: Error in library attribute "<*attribute name*>":
<*Number*>. query:
AQL expressions empty!

Cause: The AQL expressions are missing in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-15]

Message: The entered value "<*Value*>" is invalid!
Expected is <*Format*>.

Cause: While carrying out a predefined query, you have entered a value, which does not match to the defined input type.

Action: Enter a value which suits the input type.

[aanaud-16]

Message: Error in library attribute "<*attribute name*>":

<*Number*>. query:

Result attribute empty!

Cause: A plan without result attributes has been defined in the query in the above-mentioned library attribute.

Action: Contact your ADOxx administrator.

[aanaud-17]

Message: Error in library attribute "<*attribute name*>":

The length of the attribute value exceeds the maximum allowed length of 32.000 characters!

Cause: The definition for the mentioned query attribute exceeds 32.000 characters and therefore can not be saved.

Action: Reduce the attribute value by deleting not used parts of the text.

[aap-01]

Message: The attribute possesses no facet of the type "<*Type name*>".

Cause: The attribute possesses no facet with the names "AttributeProfileRefDomain" and so cannot be used.

Action: Contact your ADOxx administrator.

[aap-02]

Message: The expression is faulty:

"<*LEO error message*>"

Cause: The value of the facet "AttributeProfileRefDomain" is faulty. The LEO expression entered contains a syntactical error. Note the appropriate [aleo] error message.

Action: Rectify the value of the facet "AttributeProfileRefDomain" and take into account the specified [aleo] error message or contact your ADOxx administrator.

[aap-03]

Message: The LEO expression is empty.

Cause: The facet "AttributeProfileRefDomain" is empty.

Action: Rectify the value of the facet "AttributeProfileRefDomain" or contact your ADOxx administrator.

[aap-04]

Message: The key word "APREF" is missing.

Cause: The LEO expression in the facet "AttributeProfileRefDomain" does not contain the required key word "APREF".

Action: Correct the value of the facet "AttributeProfileRefDomain" by adding the key word "APREF" or contact your ADOxx administrator.

[aap-05]

Message: The parameter "c" of the key word "APREF" is missing.

Cause: The parameter "c" of the key word "APREF" is required but has not been found.

Action: Rectify the value of the facet "AttributeProfileRefDomain" by adding the parameter "c" to the key word "APREF" or contact your ADOxx administrator.

[aap-06]

Message: The parameter "c" of the key word "APREF" contains an unknown class name.

Cause: The class entered in the parameter "c" does not exist.

Action: Rectify the value of the facet "AttributeProfileRefDomain" by adding a valid class name or contact your ADOxx administrator.

[aap-07]

Message: The specified class in the parameter "c" of the key word "APREF" is not an attribute profile class.

Cause: The specified class in the parameter "c" is not the name of an attribute profile class but the name of another ADOxx class.

Action: Rectify the value of the facet "AttributeProfileRefDomain" by inserting a valid attribute profile class name or contact your ADOxx administrator.

[aap-08]

Message: The specified class in the parameter "c" of the key word "APREF" is not instanceable.

Cause: The class specified in the parameter "c" may be an attribute profile class, but it has been defined as "abstract" and so cannot be instanced.

Action: Rectify the value of the facet "AttributeProfileRefDomain" by inserting a valid attribute profile class name or set the referenced class on "not abstract" or contact your ADOxx administrator.

© BOC Group 2008

[aap-09]

Message: The LEO expression contains too many key words.

Cause: The facet "AttributeProfileRefDomain" can only reference one attribute profile class.

Action: Rectify the value of the facet "AttributeProfileRefDomain" by referencing only one attribute profile class or contact your ADOxx administrator.

[aap-10]

Message: The LEO expression can contain a maximum of one reference.

Cause: An attribute value refers to more than one attribute profile.

Action: Change the attribute value by referencing only one attribute profile or contact your ADOxx administrator.

[aap-11]

Message: The key word "REF" is missing.

Cause: The attribute value is faulty because the key word "REF" is missing.

Action: Change the attribute value by inserting the key word "REF" or contact your ADOxx administrator.

[aap-12]

Message: The parameter "i" of the key word "REF" is missing.

Cause: The attribute value is faulty because the parameter "i" of the key word "APREF" is not available.

Action: Change the attribute value by adding the parameter "i" to the key word "REF" or contact your ADOxx administrator.

[aap-14]

Message: The specified instance in the parameter "i" of the key word "REF" is invalid.

Cause: The referenced attribute profile has not been found.

Action: Change the attribute value so that a valid attribute profile is shown or contact your ADOxx administrator.

[aap-15]

Message: The specified instance in the parameter "i" of the key word "REF" does not have the suitable class.

Cause: The referenced attribute profile is not from the class specified in the facet "AttributeProfileRefDomain".

Action: Change the attribute value so that a valid attribute profile is shown or contact your ADOxx administrator.

[aapedit-01]

Message: The entry "<*entry text*>" is not authorised for the attribute "<*attribute name*>"!

<*Additional error text*>

Would you like to further edit the attribute?

Cause: During the editing of attribute profiles, an invalid attribute value has been entered.

Action: Enter a valid value for this attribute.

[aapedit-02]

Message: The name "<*object name*>" is already assigned to another object of the class "<*Class name*>" !

<*Additional error text*>

Would you like to further edit the attribute?

Cause: When changing the name of an object in the Attribute Profile Management, you have selected a name which is already given for this class .

Action: Select another name for the object.

[aapedit-03]

Message: Not all values could be inserted!

Cause: When pasting in the ADOxx browser, an error has occurred. This cancelled the pasting.

Action: Check if the area to be pasted is not too large and if the attributes go together. Then verify if it is allowed to paste in the area you have selected.

[aappmod-03]

Message: The application model <*application model name*> already exists. Please use another name.

Cause: Application models are identified by their names. Therefore each application model must have a unique name.

Action: Enter a different name for the required application model.

[aappmod-10]

Message: The application model <*model name*> could not be deleted. Please refresh the model lists and try again.

Cause: Possibly an error with the connection to the ADOxx database occurred or - if you are working in a client/server environment - the application model may already have been deleted by another ADOxx user.

Action: Refresh the model lists by clicking on the "Refresh" button. If the error re-occurs, then exit ADOxx and restart the application. If the error re-re-occurs, please contact your ADOxx administrator.

[aappmod-11]

Message: The application model <*model name*> could not be created. Refresh the model lists and try again.

Cause: Possibly an error occurred with your connection to the ADOxx database or - if you are working in a client/server environment - it is possible that one of the models you have selected was already deleted by another ADOxx user.

Action: Refresh the model lists by clicking on the "Refresh" button. If the error re-occurs, then exit ADOxx and restart the application. If the error re-occurs, then please contact your ADOxx administrator.

[aappmod-12]

Message: The application model <*model name*> could not be changed. Please refresh the application model list and try again.

Cause: Possibly a problem occurred with your connection to the ADOxx database or - if you are working in a client/server environment - one of your selected models may already have been deleted by another ADOxx user.

Action: Refresh the model lists by clicking on the "Refresh" button. If the error re-occurs, exit from ADOxx and restart the application. If the error re-occurs, then please contact your ADOxx administrator.

[aappmod-13]

Message: The models contained in the application model "<*model name*>" have been deleted. This is why the application model is no longer valid. Delete the application model "<*model name*>" and then actualise the model lists.

Cause: You have tried to open or change a no longer existing model.

Action: Redefine the application library after having updated the model lists .

[aapqury-01]

Message: The AQL expression is syntactically not correct!

Cause: The AQL expression you have entered does not correspond with the AQL syntax defined.

Action: Enter a syntactically correct AQL expression.

[aapqury-02]

Message: "<*Attribute profile name*>" is already being edited by another ADOxx user and thus can not be changed!

Cause: The specified attribute profile cannot be changed as it is already being edited by another ADOxx user.

Action: Carry out the changes later.

[aapqury-03]

Message: Error during the copy: Not all values could be inserted!

Cause: During the insertion in the ADOxx browser, an error has occurred. The pasting has been cancelled.

Action: Check if the area to be pasted is not too large and if the attribute types go together. Then check if it is possible to paste in the area you have selected.

[aapqury-04]

Message: Error during the copy: No value could be inserted!

Cause: During the insertion in the ADOxx browser, none of the values to be inserted could be assigned.

Action: Check if the area to be pasted is not to large and if the attribute types go together. Then check if it possible to paste in the area you have selected.

[aapview-01]

Message: Error when following an attribute profile reference:

"<*Model name*>" (<*model type*>)

could not be opened.

The referenced model has meanwhile possibly been deleted.

Cause: The original model of the attribute profile reference cannot be opened, because it has been meanwhile deleted by another ADOxx user or your access right to this model has been withdrawn by the ADOxx administrator.

Action: Speak with other ADOxx users or with your ADOxx administrator.

[abmpsup-01]

Message: File "<*filename*>" cannot be opened!

Cause: One of the specified files cannot be opened with write access. Possibly an incorrect path has been entered or there may be too little space available in the directory or disk on which the file should be stored.

Action: Please check that the file and path were specified correctly and check the space available on the data medium on which the file should be stored.

[abmpsup-02]

Message: No file name specified!

Cause: You wish to save the generated graphic in a file, but have not specified a filename.

Action: Please enter a filename in which the graphic should be stored.

[abrsimpl-06]

Message: The column width must be of a number greater than 0 and smaller than 200!

Cause: You have entered no number or a number outside the valid area (between 0 and 200, excluding limits).

Action: Enter a valid column width.

[acaledit-01]

Message: Invalid entry of the start time (format hh:mm)!

Cause: The start time has been entered in an invalid format.

Action: Enter the start time in format hh:mm (hours:minutes).

[acaledit-02]

Message: Invalid entry of the end time (format hh:mm)!

Cause: The end time has been entered in an invalid format.

Action: Enter the end time in format hh:mm (hours:minutes).

[acaledit-03]

Message: The starting time must be before the ending time!

Cause: The start time you have entered is after the input end time.

Action: Define the start and end times so that the start time is earlier than the end time.

[acaledit-04]

Message: Invalid initialising value!

Cause: The internal value for the calendar is invalid, since it contains an unknown distribution (only uniform or exponential distribution are allowed).

Action: Correct the calendar value.

[acalui-01]

Message: The day profile <*day profile name*> cannot be deleted because it is still being used. Please click on the search button to identify the assigned days.

Cause: You are attempting to delete a day profile which is still assigned to a number of days in the calendar. A day profile can only be deleted when it is no longer assigned to any days.

Action: If you really want to delete the selected day profile, then you must first ensure that it is no longer assigned to any days in the calendar. Using the "Search" button, you can identify the days to which this day profile is still assigned. Using the "Assign" button you can assign a day profile to a number of different days. Alternatively, by double-clicking on a day directly within the calendar, you can change the assigned day profile.

[acalui-02]

Message: Please enter a name for the new day profile.

Cause: You have not specified a name for the day profile you are trying to create. As day profiles are identified by their names, it is essential to enter a unique name.

Action: Enter a valid name for the day profile.

[acalui-03]

Message: The day profile name <day profile name> is not allowed. Either a day profile with this name already exists or the name specified contains restricted characters. Please use a different name.

Cause: An invalid name has been entered for the day profile. Either a day profile with this name already exists or the name specified contains the character "@".

Action: Enter a valid name for the new day profile which does not contain the character "@".

[acalui-04]

Message: The maximum number of day profiles has been reached. Before defining a new day profile, please delete at least one.

Cause: There is a maximum permitted number of day profile (approx. 50). This number has been reached.

Action: Delete at least one of the existing day profiles in order to define a new one.

[acalui-05]

Message: Please select a day profile.

Cause: You have clicked on a button (Change, Delete, Assign or Search) which requires a day profile to be specified before the relevant action can be carried out.

Action: Select a day profile from the list and click on the button again.

[acalui-06]

Message: The input in the start field is not time.

Cause: You have entered a value into the Start field which is not in the correct time format. Note that the required format is "hh:mm:ss", where "hh" is a number between "00" and "23", and both "mm" and "ss" require a number between "00" and "59". A valid value is for example "08:00:00".

Action: Enter a time as described in "Cause" above.

[acalui-07]

Message: The input in the End field is not time.

Cause: You have entered a value into the End field which is not in the correct time format. Note that the required format is "hh:mm:ss", where "hh" is a number between "00" and "23", and both "mm" and "ss" require a number between "00" and "59". A valid value is for example "17:00:00".

Action: Enter a time as described in "Cause" above.

[acalui-08]

Message: The start time cannot be later than the end time.

Cause: When defining time intervals, the start time must be earlier than the end time. Please note that the time intervals within a day must be between "00:00:00" and "23:59:59".

Action: Enter values in the Start and End fields which make up a valid time interval.

[acalui-09]

Message: There are intersections with intervals that are already defined in the day profile.

Cause: You have attempted to define a new time interval which overlaps with existing intervals. The time intervals within a day profile must be distinct - i.e. they cannot overlap with any other time intervals.

Action: Enter a valid time interval which doesn't overlap with any other time intervals. If necessary, delete or change existing time intervals.

[acalui-10]

Message: Please select an interval.

Cause: You have clicked on a button (Change or Delete) which requires a time interval on which to work.

Action: Select the time interval on which you want to carry out the action, and then click on the relevant button again.

[acalui-13]

Message: Please consider that the start date must be before the end date.

Cause: You have entered start and end dates which don't define a time interval. Please note that the time interval must lie within a single year (January - December).

Action: Enter a start and end date which define a valid time interval.

[acalui-14]

Message: The days set is empty. Please activate at least one input type.

Cause: You have not selected an input type ("Day" or "Time interval"). This means that no dates can be selected to which a day profile can be assigned.

Action: Activate at least one input type (by clicking) and then define the days to which the day profile should be assigned.

[acalui-15]

Message: The calendar is too large. Therefore it cannot be stored in the database. Please delete some day profiles.

Cause: There is an internal limit defined for the complexity of the calendar. When a very high number of day profiles are defined (>50) the calendar can become so complex that it can no longer be saved.

Action: Delete any unnecessary day profiles - i.e. day profiles which are not assigned to any days.

[acalui-16]

Message: The distribution is not correct.

Cause: You have entered a value into the input field which does not meet the conventions of the distribution.

Action: Once you select your required distribution type, the syntax is shown in the 'Distribution' field. When using uniform distribution, please be careful that the minimum and maximum bounds are in the correct time format.

[acalui-17]

Message: The maximum number of day profiles has been defined. Before copying a day profile, please delete at least one.

Cause: A limit exists for the maximum number of day profiles which can be defined (approx. 50).

Action: Delete at least one day profile.

[acalui-18]

Message: Please specify the reference object and select at least one day profile.

Cause: In order to copy a day profile into the current calendar, you must first specify the model from which you want to copy, the reference object which contains the day profile you want to copy and finally the actual day profile itself.

Action: Select the reference object and the day profile.

[acalui-19]

Message: A day profile named<*day profile name*> already exists. Therefore it cannot be copied.

Cause: Day profiles must have a unique name within a calendar. Therefore if a day profile with the same name already exists in the calendar to which you are attempting to copy, the system will not allow the new profile to be copied.

Action: If you are sure that you want to copy the new day profile into your calendar, you must first delete the existing day profile with the same name.

[acalui-20]

Message: The calendar has the wrong format. Therefore the day profiles cannot be copied.

Cause: If the calendar of the selected reference object has an invalid format, then no day profiles can be copied.

Action: Select a different reference object. In order to correct the invalid calendar of the reference object, you should open the ADOxx notebook of that object - the calendar will then automatically be corrected to contain standard values.

[acard-02]

Message: "<*Object*>": A maximum of <*number*> connectors is allowed (<*direction*>).

Should the verification of the cardinalities be continued?

Cause: The verification of the cardinalities of the model has determined that the maximum number of allowed connectors in the specified object is too high .

The possibly shown direction ("incoming", "outgoing" or "all together") indicates whether the maximum number of incoming or outgoing connectors has been reached or if the number of connectors in general is too high.

Action: According to the defined cardinality restrictions, only a certain number of connectors is allowed. To adhere to these restrictions, you have to delete connectors which start from or end in the specified object.

[acard-03]

Message: "<*source*>": A maximum of <number> connectors of the type "<*relation class*>" is allowed (<direction>).

Should the verification of the cardinalities be continued?

Cause: The verification of the cardinalities of the model has determined that the maximum number of allowed connectors of the specified relation type in the specified source is too high.

The *source* is about a concrete object, an object of the class displayed a relation between a concrete object and objects of the specified class or about the relation between objects of the class shown and a concrete object.

The possibly shown direction ("incoming", "outgoing" or "all together") indicates whether the maximum number of incoming or outgoing connectors has been reached or if the number of connectors in general is too high.

Action: According to the cardinality restrictions, only a certain number of connectors of the specified relation class are permitted. To adhere to these restrictions, you have to delete connectors which start from or end in the specified object.

[acard-04]

Message: Unexpected value in the attribute "<*attribute*>" ("<*class*>"): "<*error expression*>" is not a valid value. "<*correct expression*>" is expected.

Cause: The specified attribute of the specified class contains an invalid value. The valid expression listed is expected instead.

Action: Contact your ADOxx administrator. The Administration Toolkit should be used to correct the invalid value.

[acard-05]

Message: The model contains<*number*> objects of the class "<*class name*>". A maximum of <*number*> objects are allowed.

Should the verification of the cardinalities be continued?

Cause: The verification of a model has determined that the maximumnumber of allowed objects of the specific class is too high.

Action: According to the defined cardinality restrictions, only a certain number of objects of the specified class is allowed. To fulfill these restrictions, you must delete the appropriate number of objects of this class.

[acard-06]

Message: The model contains<*number*>objects of the class "<*class name*>". <At least *number*> object(s) is/are required.

Should the verification of the cardinalities be continued?

Cause: The verification of a model has determined that theminimum number of required connectors of the specific class has not been reached.

Action: According to the defined cardinality restrictions, a minimum number of objects of the specified class is required. To fulfill these restrictions, you must create the corresponding number of objects of this class.

[acard-07]

Message: <*Object or class name*>:At least <*number*> connector(s) is/are required (<*direction*>).

Should the verification of the cardinalities be continued?

Cause: The verification of a model has determined that the minimal number of required connectors has not been reached.

The possibly shown direction ("incoming", "outgoing" or "all together") indicates whether the minimum number of incoming or outgoing connectors has not been reached or if the number of connectors in general is too low.

Action: According to the defined cardinality restrictions, a minimum number of connectors is required. To fulfill these restrictions, you must create additional connectors, which start from or end in the specified object.

[acard-08]

Message: <*Object or class name*>: At least<*number*> connector(s) of the type "<*relation class name*>" is/are required (<*direction*>).

Should the verification of the cardinalities be continued?

Cause: The verification of a model has determined that for the minimum number of required connectors of the specified class for the specified object has not been reached.

The possibly shown direction ("incoming", "outgoing" or "all together") indicates whether the minimum number of incoming or outgoing connectors has not been reached or if the number of connectors in general is too low.

Action: According to the defined cardinality restrictions, a minimum number of connectors of the specified relation class is required. To fulfill these restrictions, you must create additional connectors of the relation class, which start from or end in the specified object.

[acard-09]

Message: Error in the assignment of the attribute "<*attribute name*>" ("<*class name*>"): "<*expression*>" is not applicable to the relation class "<*relation class name*>"!

Cause: The specified cardinality restriction is not meaningful.

This message will be delivered for instance if the number of **incoming** connectors are restricted to a class, although the corresponding connectors of this class can only be **outgoing** (and not incoming).

Action: Remove this restriction.

[acase40-01]

Message: Error while following the selected reference!

Cause: The referenced object in case/4/0 has been deleted or case/4/0 could not be started

Action: Delete or change the reference or install case/4/0 correctly

[acase40-02]

Message: An error has occurred during the new creation of a reference!!

Cause: This is a serious error on the part of case/4/0. Possibly case/4/0 is not correctly installed.

Action: Install case/4/0 or check the existing case/4/0 installation.

[acase40-03]

Message: An error has occurred while attempting to change the selected reference!

Cause: This is a serious error on the part of case/4/0. Possibly case/4/0 is not correctly installed.

Action: Install case/4/0 or check the existing case/4/0 installation.

[acase40-04]

Message: An error has occurred while deleting a reference!

Cause: This is a serious error on the part of case/4/0. Possibly case/4/0 is not correctly installed.

Action: Install case/4/0 or check the existing case/4/0 installation.

[acccber-01]

Message: The quantity is 0 in at least one model of the application model!

Cause: The 'Quantity' attribute in a 'Process start' object is 0 for at least one of the business process models belonging to the application model.

Action: Check the business processes in the application model and specify a 'Quantity' greater than 0 for each 'Process start' object.

[acccber-02]

Message: The cost driver quantity is 0 in at least one model of the application model!

Cause: The attribute 'Cost driver quantity' in objects of the class 'Process start' is 0 in at least one of the business process models belonging to the application model.

Action: Check the business processes in the application model and specify a 'Cost driver quantity' greater than 0 for each 'Process start' object.

[acccdlg-01]

Message: Internal error "Internal error number" while processing the application model "application model name"!

Cause: An internal error occurred while processing this application model.

Action: Please contact your ADOxx administrator.

[acccdlg-02]

Message: The input parameters are not defined correctly. Please contact ADOxx customer support.

Cause: The input parameters specified for the ADOxx-3c simulation are not consistent with the class and attribute definitions of the application library.

Action: Correct the "sim mapping" definition in the Administration Toolkit or contact your ADOxx consultant.

[acccdlg-03]

Message: There is no quantity defined in at least one model of the application model!.

<*Model name*>

Cause: The 'Quantity' attribute in a 'Process start' object is 0 for at least one of the business process models belonging to the application model.

Action: Check the business processes in the application model and specify a 'Quantity' greater than 0 for each 'Process start' object.

[acccinit-01]

Message: At least one performer is not assigned to a cost centre!

Cause: At least one of the performers from the working environment model specified as part of the application model is not assigned to a cost centre.

Action: A performer must belong to exactly one cost centre. Connect the performer through the relation "Is charged to" with the corresponding cost centre.

[acccinit-02]

Message: At least one performer is assigned to more than one cost centre!

Cause: At least one of the performers from the working environment model specified as part of the application library is assigned to more than one cost centre.

Action: A performer must be assigned to exactly one cost centre. Delete the incorrect relations from the model.

[acccinit-03]

Message: At least one cost centre has more than one manager!

Cause: At least one cost centre in the working environment model is connected to more than one performer with the relation "Is cost centre manager".

Action: Each cost centre must have at most one manager - so delete the unnecessary relations.

[acccinit-04]

Message: The attribute <*attribute name*> is not defined!

Cause: The attribute specified is not defined in the application library.

Action: Please contact your ADOxx consultant.

[acccinit-05]

Message: The class<*class name*> is not defined in the application library!

Cause: The class specified is not defined in the application library.

Action: Please contact your ADOxx consultant.

[acnumchk-01]

Message: The key word "DOMAIN" is missing.

Cause: The LEO expression in the facet "AttributeNumericDomain" does not contain the required key word "DOMAIN".

Action: Rectify the value of the facet "AttributeNumericDomain", by adding the key word "DOMAIN" or contact your ADOxx administrator.

[acnumchk-02]

Message: The parameter "message" of the key word "DOMAIN" is missing.

Cause: The LEO expression in the facet "AttributeNumericDomain" does not contain the required parameter "message" for the key word "DOMAIN".

Action: Rectify the value of the facet "AttributeNumericDomain", by adding the parameter "message" to the key word "DOMAIN" or contact your ADOxx administrator.

[acnumchk-03]

Message: After the key word "DOMAIN", only the key word "INTERVAL" is authorised.

Cause: The LEO expression in the facet "AttributeNumericDomain" contains an invalid key word. After the key word "DOMAIN", only the key word "INTERVAL" is authorised.

Action: Rectify the value of the facet "AttributeNumericDomain", by also using the key word "INTERVAL" after the key word "DOMAIN" or contact your ADOxx administrator.

[acnumchk-04]

Message: The parameter "lowerbound" of the key word "INTERVAL" is missing.

Cause: The LEO expression in the facet "AttributeNumericDomain" does not contain the required parameter "lowerbound" for the key word "INTERVAL".

Action: Rectify the value of the facet "AttributeNumericDomain", by adding the parameter "lowerbound' to the key word "INTERVAL" or contact your ADOxx administrator.

[acnumchk-05]

Message: The parameter "upperbound" of the key word "INTERVAL" is missing.

Cause: The LEO expression in the facet "AttributeNumericDomain" does not contain the required parameter "upperbound" for the key word "INTERVAL".

Action: Rectify the value of the facet "AttributeNumericDomain", by adding the parameter "upperbound" to the key word "INTERVAL" or contact your ADOxx administrator.

[acnumchk-06]

Message: The value of the parameter "upperbound" is smaller than the value of the parameter "lowerbound".

Cause: The value of the parameter "upperbound" of the key word "INTERVAL" is smaller than the value of the parameter "lowerbound" of the same key word.

Action: Change the value of the parameter "lowerbound" and "upperbound" so that the value of the parameter "upperbound" is greater than or equal to the value of the parameter "lowerbound" or contact your ADOxx administrator.

[acnumchk-07]

Message: The value is not in one of the specified intervals.

Cause: The specified value is not within the area valid for the attribute.

Action: Change the value so that it is in an interval specified in the facet "AttributeNumericDomain".

[acoexhan-01]

Message: No expression!

Cause: The value of the edited EXPRESSION attribute does not contain any expression, i.e. no formula or possible constant. This way no result can be calculated for the EXPRESSION attribute.

Action: Specify an expression for the calculation of the result.

[acoexhan-02]

Message: The result of the expression is not of the type specified in the attribute definition!

Expected type: "<*result type1*>"

Delivered type: "<*result type2*>".

Cause: The expression of the EXPRESSION attribute delivers a result, which is of another type than the one specified in the attribute definition. This is why you cannot use the result.

Action: Specify an expression which delivers a result matching to the type. If necessary, insert a conversion operator (VAL/STR) in this formula .

[acoexhan-03]

Message: The calculated value is too long/big to save in the database!

Cause: The result of an EXPRESSION attribute returned a value which is too long/big to save in the database.

Action: Define an expression resulting in a shorter/smaller value.

[acoexhan-04]

Message: The expression is too long to store in the database!

Cause: The EXPRESSION is too long and therefore cannot be saved into the database.

Action: Define a shorter expression to calculate the result.

[acoexman-01]

Message: The syntax definition is EXPR type:<ResultType> [expr: [fixed:]<expression>, however the attribute value is "<*attribute value*>"!

Cause: The value defined for the EXPRESSION attribute is syntactically wrong.

Action: Change the value so that the syntax instructions come up.

[acoexpar-01]

Message: Internal error!

Cause: An unexpected error code has been delivered in the EXPRESSION attribute during the call of a core function. This can occur when a number which is not an ID is given instead of an ID.

Action: Verify and correct the formula.

[acoexpar-02]

Message: Attribute "<*attribute name*>" (<*Class name*>) does not exist!

Cause: In the EXPRESSION attribute, a string, whose value is not an attribute name in the specified class, has been assigned during the call of a function, which expects an attribute name as a parameter.

Action: Verify and correct the formula.

[acoexpar-03]

Message: The attribute "<*attribute name*>" (<*class name*>) cannot be accessed because of its types in the expressions!

Cause: You are trying to sort an attribute of the type RECORD or PROFILEREF.

Action: Verify and correct the formula.

[acoexpar-04]

Message: Error in the referenced attribute "<*attribute name*>" of the objects "<*object name*>" (<*class name*>) in "<*model name*>" (<*model type*>)!

Cause: The EXPRESSION attribute accesses to another EXPRESSION attribute, in which an error is delivered instead of a value.

Action: Verify and correct the formula of the referenced EXPRESSION attribute specified in the error message.

[acoexpar-05]

Message: At the moment, the object "<*Object name*>" (<*class name*>) in "<*model name*>" (<*model type*>) cannot be accessed!

Cause: The EXPRESSION attribute attempts to access an object which is not in the memory yet. This conflict can only occur during the loading of a model and will disappear itself after the loading process.

Action: No action required or action is done automatically.

[acoexpar-06]

Message: No object "<*object name*>" (<*class name*>) exists in "<*model name*>" (<*model type*>)!

Cause: In the EXPRESSION attribute, a string, with a value which is not the name of an object of the specified class, has been assigned during the call of a function, which has an object name as parameter.

Action: Verify and correct the formula.

[acoexpar-07]

Message: In the model type "<*model type*>", there is no class "<*class name*>"!

Cause: In the EXPRESSION attribute, a string whose value is not the name of a class in the specified model type, has been assigned during the call of a function, which expects a class name as a parameter.

Action: Verify and correct the formula.

[acoexpar-08]

Message: Access to the specific attribute "<*attribute name*>" is not authorised!

Cause: In the EXPRESSION attribute, you have attempted to sort this attribute, which leads to an endless recursion.

Action: Verify and correct the formula.

[acoexpar-09]

Message: The attribute "<*attribute name*>" (<*class name*>) is not of numerical type!

Cause: In the EXPRESSION attribute, you have attempted to calculate an arithmetical aggregate function via a non numerical attribute.

Action: Verify and correct the formula.

[acoexpar-10]

Message: The ID list is faulty!

Cause: In the EXPRESSION attribute, a string, which contains no or not only IDs, is assigned to a function, which expects a string with a list of IDs as parameter.

Action: Verify and correct the formula.

[acoexpar-11]

Message: <*Function name*> cannot be evaluated without access to the Modelling Component!

Cause: In the EXPRESSION attribute, you have attempted to access the "modelling" message port, which is not available.

Action: None. This error does not occur in the Modelling Toolkit.

[acoexpar-12]

Message: No column "<*column name*> exists in the record attribute "<*attribute name*>" (<*class name*>)!

Cause: In the EXPRESSION attribute, a string, whose value is not the name of a column of the specified RECORD attribute, is assigned during the call of a function, which expects as a column name of a RECORD attribute as parameter, .

Action: Verify and correct the formula.

[acoexpar-13]

Message: The column "<*column name*>" of the record attribute "<attribute name>" (<*class name*>) is not of numerical type!

Cause: In the EXPRESSION attribute, you have attempted to calculate an arithmetical aggregate function using a non numerical attribute.

Action: Verify and correct the formula.

[acoexpar-14]

Message: The column "<*column name*>" of the record attribute "<*attribute name*>" (<*class name*>) is not of the type <*type name*>!

Cause: In the EXPRESSION attribute, a RECORD column is accessed, whose attribute type is different from the attribute type expected.

Action: Verify and correct the formula.

[acoexpar-15]

Message: Syntactical error in the AQL expression!

Cause: In the EXPRESSION attribute, a function is called, which expects an AQL expression with a string as parameter. The value assigned is not a syntactically correct AQL expression.

Action: Verify and correct the formula.

[acoexpar-16]

Message: Reference missing in the object "<*Object name*>" (<*class name*>), record attribute "<*attribute name*>", column "<*column name*>" [<*column index name*>]!

Cause: In the EXPRESSION attribute, an INTERREF attribute, which must contain a valid reference, is accessed. However the INTERREF attribute contains either no reference or no valid reference.

Action: Change the specified INTERREF attribute, so that it contains a valid reference.

[acoexpar-17]

Message: The attribute "<*attribute name*>" (<*class name*>) is neither of numerical nor of time type!

Cause: In the EXPRESSION attribute, you have attempted to calculate an arithmetical aggregate function via a non numerical and non time attribute.

Action: Verify and correct the formula.

[acoexpar-19]

Message: The AQL expression cannot be evaluated!

Cause: In the EXPRESSION attribute, you have attempted to evaluate an AQL expression, without having the message port required for it available.

Action: None. This error does not occur in the Modelling Toolkit.

[acoexpar-20]

Message: Cyclic dependence when accessing to the attribute "<*attribute name*>" of the object "<*object name*>" (<*class name*>) in "<*model name*>" (<*model type*>)!

Cause: Several EXPRESSION attributes cyclically access another, so that an endless recursion appears.

Action: Verify and correct the formulas of the concerned EXPRESSION attribute.

[acoexpar-21]

Message: In the model type <*model type*> there is no relation type "<*class name*>"!

Cause: In the EXPRESSION attribute, a string, which is not the name of a relation type, is assigned to a function, which expects the name of a relation type as parameter.

Action: Verify and correct the formula.

[acoexpar-22]

Message: The ID list when calling the <*function name*> is empty!

Cause: In the EXPRESSION attribute, an empty string is assigned to a function which expects a non-empty ID list as a parameter. The reason can be, that an INTERREF attribute is accessed in which no valid reference is contained.

Action: Verify and correct the formula and the included INTERREF attributes.

[acoexpar-23]

Message: The attribute "<*attribute name*>" (<*class name*>) is not of the type <*type name*>!

Cause: In the EXPRESSION attribute, an attribute which is different from the expected attribute type, is being accessed using a function.

Action: Verify and correct the formula.

[acoexpar-24]

Message: At least one object is referenced by the INTERREF attribute "<*attribute name*>" of the object "<*object name*>" (<*class name*>) and you cannot access this object as "<*model name*>" (<*model type*>) is not loaded!

Cause: In the EXPRESSION attribute, you have attempted to find IDs referenced objects, without having opened the target model.

Action: Open the target model. If the expression could be correctly calculated once, the last valid result will be delivered instead of this error.

[acoexpar-25]

Message: An object "<*object name*>" (<*class name*>) is referenced by the INTERREF attribute "<*attribute name*>" of the object "<*object name*>" (<*class name*>) and this object does not exist in "<*model name*>" (<*model type*>)!

Cause: In the EXPRESSION attribute, you have attempted to find IDs referencing, no longer existing objects.

Action: Edit the INTERREF attribute concerned or create new objects, so that the INTERREF attribute contain valid references again.

[acoexpar-26]

Message: The INTERREF attribute "<*attribute name*>" of the object "<*object name*>" (<*class name*>) contains errors!

Cause: In the EXPRESSION attribute, you have accessed an INTERREF attribute, which contains an invalid value. This means that the database is corrupt.

Action: The ADOxx database may be recreated.

[acoexpar-27]

Message: The call of the <*function name*> can not be evaluated, as "<*model name*>" (<*model type*>) is not loaded!

Cause: In the EXPRESSION attribute, you have attempted to sort information from a model which is not opened.

Action: Open the specified model. If the expression could be correctly calculated once, the last valid result will be delivered instead of this error.

[acoexpar-28]

Message: The call of the <*function name*> can not be evaluated, as there is no scope or referenced model!

Cause: You have attempted to evaluate an object-related expression without having specified the scope. This can occur during the call of EVAL_EXPRESSION, when functions have been called, which refer to a specific object id and the parameter object has not been specified.

Action: Specify the object id when calling EVAL_EXPRESSION.

[acoexpar-29]

Message: The record attribute "<*attribute name*>" (<*class name*>) contains no row with index <*row number*>!

Cause: In the EXPRESSION attribute, you have attempted to access a RECORD row which does not exist.

Action: Verify and correct the formula or edit the specified RECORD attribute.

[acoexpar-30]

Message: The attribute profile referenced by "<*attribute name*>" (<*class name*>) does not exist!

Cause: In the EXPRESSION attribute, you have attempted to access to a referenced attribute profile, which has broken the attribute profile reference .

Action: Verify and correct the formula or the attribute profile reference specified in the error message.

[acompare-01]

Message: Please specify a filename.

Cause: No filename has been specified.

Action: Please enter the filename to which you wish to write.

[acompare-02]

Message: Invalid filename.

Cause: An invalid filename has been specified for the result file.

Action: Enter a valid filename.

[acompare-03]

Message: Error while opening the result file.

Cause: The result file could not be opened - perhaps no access was possible to the location on which the file is stored at the time (e.g. diskette).

Action: Check the data medium or location and begin again.

[acompare-04]

Message: Error while reading the result file.

Cause: The specified result file was not in the expected ACR format.

Action: While saving this file, a serious error must have occurred. Use a different result file.

[acompare-07]

Message: A representation is not possible because there are no common values available.

Cause: The selected results have no value in common. Therefore a comparable representation is not possible.

Action: Select different results to be compared.

[acompare-13]

Message: The result <*result name*> does not exist.

Cause: You have selected a non-existent result.

Action: Enter the name of an existing result.

[acompare-14]

Message: Error while writing to file.

Cause: An error occurred while saving information on the result comparison to a file.

Action: Possibly there is not enough space on the data medium. Please check the space available on the location to which you are saving the file and then try again.

[acomps-01]

Message: No control component specified!

Cause: A fatal error has occurred.

Action: If this error re-occurs after attempting simulation again, export your models to an ADL file and contact your ADOxx consultant.

[aconfui-01]

Message: An invalid customer number has been entered! You must use a combination of uppercase letters (A-Z), lowercase letters (a-z) or numbers (0-9) with a minimum of 5 and a maximum of 17 characters.

In the expanded mode, you can use a minimum of 5 and a maximum of 24 characters.

Cause: The customer number does not meet the specified conditions.

Action: Enter a valid customer number.

[aconfui-03]

Message: An error has occurred when saving the licence number!

Cause: The licence number could not be saved to the file. There may not be enough free memory space available or the file is write-protected.

Action: Create some free memory space or save the licence number to another file.

[aconfui-04]

Message: No file name has been entered!

Cause: No file name has been entered!

Action: Enter a file name.

[aconfui-05]

Message: Cannot load current configuration!

Cause: An error occurred while the current configuration was read from the ADOxx database.

Action: Quit ADOxx and re-start it. Should this error re-occur, contact your system administrator.

[aconfui-10]

Message: Licence number does not correspond with customer number!

Cause: The licence number entered cannot be used for configuring components when combined with the customer number listed.

Action: Check the licence number entered. Should this error still occur, contact your system administrator or your ADOxx consultant.

[aconfui-12]

Message: No licence number entered!

Cause: You did not enter a licence number.

Action: Enter the licence number for the new configuration of components of the ADOxx Modelling Toolkit.

[aconfui-13]

Message: New configuration could not be saved in database.

Cause: An error occurred when the new configuration was to be saved in the ADOxx database.

Action: Quit ADOxx and re-start it. Should this error re-occur, contact your database or system administrator.

[aconfui-14]

Message: The generation of the new licence number is not possible!

Cause: An unexpected error has occurred during the generation of the licence number.

Action: Contact your ADOxx administrator.

[aconfui-15]

Message: No test licence number has been entered!

Cause: No test licence number has been entered!.

Action: Enter a test licence number.

[aconfui-16]

Message: The test licence number could not be decoded!

Cause: The licence number is not valid.

Action: Enter a valid licence number.

© BOC Group 2008

[aconfui-17]

Message: You must configure 1-99 users to specify the maximum number of users!

In the extended mode, you can configure 1-9999 users.

Cause: An invalid number of users has been entered.

Action: Enter a valid number of users.

[aconfui-18]

Message: The Configuration Toolkit cannot be started, since an ADOxx application is already active. Close the other ADOxx application in order to start the Configuration Toolkit.

Cause: An ADOxx application is already active.

Action: First close the other ADOxx application. Then start the Configuration Toolkit again.

[acregchk-01]

Message: The key word "REGEXP" is missing.

Cause: The LEO expression in the facet "AttributeRegularExpression" doesn't contain the required key word "REGEXP".

Action: Rectify the value of the facet "AttributeRegularExpression", by adding the key word "REGEXP" or contact your ADOxx administrator.

[acregchk-02]

Message: The parameter "expression" of the key word "REGEXP" is missing.

Cause: The LEO expression in the facet "AttributeRegularExpression" does not contain the required parameter "expression" for the key word "REGEXP". In the parameter "expression", you must have the regular expression, which the attribute values must meet.

Action: Correct the value of the facet "AttributeRegularExpression", by adding the parameter "expression" to the key word "REGEXP", or contact your ADOxx administrator.

[acregchk-03]

Message: The parameter "message" of the key word "REGEXP" is missing.

Cause: The LEO expression in the facet "AttributeRegularExpression" does not contain the parameter "message" required for the key word "REGEXP". In the parameter "message", you must have the description text for the regular expression, which the attribute values must meet.

Action: Correct the value of the facet "AttributeRegularExpression", by adding the parameter "message" to the key word "REGEXP" or contact your ADOxx administrator.

[acregchk-04]

Message: The regular expression in the parameter "expression" is invalid.

Cause: The parameter "expression" in the facet "AttributeRegularExpression" contains an invalid regular expression.

Action: Rectify the value of the facet "AttributeRegularExpression", by giving a valid regular expression as value to the parameter "expression" of the key word "REGEXP" or contact your ADOxx administrator.

[acregchk-05]

Message: The expression does not meet the regular expression in the parameter "expression".

Cause: You have attempted to assign a value, which doesn't correspond to the defined regular expression.

Action: Change the value in such a way, that it meets the regular expression. If necessary, have a look at the help text of the corresponding attribute to find out if there is a description of the possible values.

[adbacc-01]

Message: <*Database error message*>Class: ADBACC, Stmt handle: <*Handle*>, Function: <*Function Name*>

Cause: A general database error occurred. The error message gives information about the cause of the error and contains the database error-code, which can be used to gain further information from the database error documentation.

Action: Exit ADOxx and start it again. If the error re-occurs, please write down the <*Database error message*>, <*Handle*> and <*Function Name*> and contact your system administrator.

[adbadm-01]

Message: <*Database error message*> Class: ADBADM, Stmt handle: <*Handle*>, Function: <*Function Name*>

Cause: A general database error occurred. The error message gives information about the cause of the error and contains the database error-code, which can be used to gain further information from the database error documentation.

Action: Quit ADOxx and start it again. If the error still occurs, please write down the <*Database error message*>, <*Handle*> and <*Function Name*> and contact your system administrator.

[adbsess-01]

Message: The network or database system is not available at the moment. The database connection has been aborted.

In the current session, no further database access is possible.

It is recommended to start the application again.

Cause: An error has occurred in the network connection.

Action: Try to start the application again. Should this error appear again, contact your system administrator.

[adbsess-02]

Message: Because of an unexpected database message, the connection with the database has been aborted.

In the current session, no further database access is possible.

It is recommended to start the application again.

Cause: Demands to the database system have been rejected by an error message.

Action: Click on the button "Show database message" and note the displayed error message and contact a system administrator or an ADOxx administrator.

Further access to the database is only possible once you have restarted the application.

[adistrib-01]

Message: Enter a numeric value for the expected valueperiod;

Cause: No number has been entered for the expected value in a normal distribution.

Action: Enter a numeric value for the expectation.

[adistrib-02]

Message: Enter a number greater than 0 for the standard deviation.

Cause: Either no number or a number less than or equal to 0 has been entered for the standard deviation value of the normal distribution.

Action: Enter a positive number for the standard deviation.

[adistrib-04]

Message: Enter a number greater than 0,00000001 for the expected value.

Cause: Either no number or a number less than or equal to 0.00000001 has been entered for the expected value in the exponential distribution.

Action: Enter an authorised number (i.e. greater than 0.00000001) for the expected value.

[adistrib-05]

Message: Enter a number for both the lower and upper bounds.

Cause: No number has been entered for either the lower or upper bound of the uniform distribution.

Action: Enter a numeric value for the lower and upper bounds.

[adistrib-06]

Message: The lower bound cannot be greater than the upper bound.

Cause: A smaller value has been entered for the upper bound of a uniform distribution compared to that entered for the lower bound.

Action: Enter numeric values for the upper and lower bounds such that the upper bound is greater than the lower bound.

[adistrib-08]

Message: The sum of the probabilities must add up to 1.

Cause: You have entered probability values for the discrete distribution where the sum of all probabilities is not 1.

Action: Specify the probabilities such that they add up to 1.

[adistrib-09]

Message: Enter a valid symbol name.

Cause: An invalid symbol name has been specified for the discrete distribution (i.e. one which contains invalid characters such as @ or is completely numeric.

Action: Enter an alphanumeric value for the symbol.

[adistrib-10]

Message: Enter a unique symbol name.

Cause: You have entered a symbol name which you have already used within this discrete distribution.

Action: Each symbol name within a discrete distribution must be unique - select a different symbol name.

[adistrib-11]

Message: Enter a value between 0 and 1. Please note that format is "0,n" or "0.n" - NOT ".n".

Cause: Either no number, or a number less than zero or greater than 1 has been entered for the probability value in the discrete distribution.

Action: Please enter a number between 0 and 1. (Format 0,n or 0.n).

[adistrib-13]

Message: Syntax error.

Cause: The distribution defined is not correct.

Action: Check the correct syntax for the distribution type.

[adistrib-14]

Message: Invalid number.

Cause: A non-numeric value has been entered as part of the distribution instead of a numeric value.

Action: Enter a numeric value.

[adistrib-15]

Message: Invalid symbol.

Cause: An invalid symbol name has been specified for the discrete distribution (i.e. one which contains invalid characters such as @ or is completely numeric.

Action: Enter an alphanumeric value for the symbol.

[adistrib-16]

Message: Ambiguous symbol.

Cause: A symbol name which is not unique within this discrete distribution has been entered.

Action: Ensure that every symbol name is unique within a discrete distribution.

[adistrib-17]

Message: No object selected!

Cause: No object was selected in resource assignment.

Action: Select the object(s) to be assigned to the resource.

[adistrib-19]

Message: Expression is syntactically incorrect!

Cause: The AQL expression for resource assignment is syntactically incorrect.

Action: Correct the AQL expression or use the input support to create an AQL expression with the correct syntax.

[adistrib-20]

Message: The current performer can only be selected alone.

Cause: You have attempted to use the expression "current performer" with other objects.

Action: Either select "current performer" or the other objects.

[adistrib-23]

Message: Expression syntactically incorrect!

Cause: The AQL expression in the "Performer" field is syntactically incorrect.

Action: Check the AQL syntax or re-enter the performer required using the input support.

[adistrib-28]

Message: Invalid value.

Times must be entered in the format yy:ddd:hh:mm:ss.

Cause: You have entered a time value in an invalid format.

Action: Enter a time value in the correct format (yy:ddd:hh:mm:ss).

[adistrib-29]

Message: Invalid value.

The year may not be higher than 999999.

Cause: Too high a value has been entered for the year.

Action: Enter value less than or equal to 999999 for the year.

[adistrib-30]

Message: Error while creating a new inter-reference:

The maximum number of references for this attribute to model type <*model type*> is <*number*>.

Cause: The maximum number of references for this attribute to models of the specified type has been exceeded.

Action: Delete another reference or contact your ADOxx administrator or consultant if you would like to increase this maximum number.

[adistrib-31]

Message: Error while creating a new inter-reference:!

The maximum number of references for this attribute

to objects of class<*class name*> in model type <model type> is <*number*>.

Cause: The maximum number of object references from this attribute has been exceeded.

Action: Delete any unnecessary object references or contact your ADOxx consultant.

[adistrib-34]

Message: Error when creating a new inter model reference:

The maximum number of references authorised for this attribute is <*number*>!

Cause: You have attempted to create a new inter model reference, although the INTERREF attribute already contains the maximum number of references authorised for this attribute.

Action: If there is an unnecessary reference in this INTERREF attribute, you must delete it first.

[adistrib-35]

Message: The transition condition contains errors:

<*Error*>

Cause: The transition condition is syntactically not correct.

Action: Correct the error.

[adistrib-36]

Message: The value for the process time contains errors:

The expected format is YY:DDD:HH:MM:SS!

Cause: The format of the process time in comparison within the transition condition is not correct.

Action: Enter a process time in ADOxx time format YY:DDD:HH:MM:SS.

[adistrib-37]

Message: The value for the day time contains errors:

The expected format is hh:mm:ss!

Cause: The format of the day time in comparison within the transition condition is not correct.

Action: Enter a day time in format hh:mm:ss.

[adistrib-38]

Message: The value for "Date-time" is invalid. The value must be in format "YYYY:MM:DD hh:mm:ss" and you must keep within the following value range:

Year: 0001 - 9999, month: 01 - 12, day: 01 - 31

Hour: 00 - 23, minute: 00 - 59, second: 00 - 59.

Cause: You have entered an invalid value in an attribute of the type "Date-time" (i.e. invalid format or outside the authorised value range .

Action: Enter the value using the format "YYYY:MM:DD hh:mm:ss" and observe the above described value range.

[adistrib-39]

Message: The value for "Year" is too big. Values between 0001 and 9999 are authorised.

Cause: In an attribute of type "Date-time", you have entered a value for "Year", which is outside the authorised value range.

Action: Enter a value between 0001 and 9999 for the year.

[adistrib-40]

Message: The value for "Date" is invalid. The value must be in format "YYYY:MM:DD" and you must keep to the following value range:

Year: 0001 - 9999, month: 01 - 12, day: 01 - 31.

Cause: You have entered an invalid value for the date in an attribute of type "Datum" (DATE) (i.e. invalid format or outside the authorised value range.

Action: Enter the value in format "YYYY:MM:DD" and observe the above described value range.

[adistrib-41]

Message: The value for "Days" is too big.

The maximum allowed value is 364 days.

Cause: In an attribute of type "Time", you have entered a value for "Days", which is outside the valid range.

Action: Enter a value between 0 and 364 for the days.

[adistrib-42]

Message: The value for "Hours" is too big.

The maximum allowed value is 23 hours.

Cause: In an attribute of type "Time", you have entered a value for "Hours", which is outside the valid range.

Action: Enter a value between 0 and 23 for the hours.

[adistrib-43]

Message: The value for "Minutes" is too big.

The maximum allowed value is 59 minutes.

Cause: In an attribute of type "Time", you have entered a value for "Minutes", which is outside the valid range.

Action: Enter a value between 0 and 59 for the minutes.

[adistrib-44]

Message: The value for "Seconds" is too big.

The maximum allowed value is 59 seconds.

Cause: In an attribute of type "Time", you have entered a value for "Seconds", which is outside the valid range.

Action: Enter a value between 0 and 59 for the seconds.

[adocase-01]

Message: Error initialising Case/4/0.

Check the following information:

<error-specific description>

Cause: A system error occurred while trying to start the case/4/0 interface or to initialise case/4/0. Possibly your system does not fulfill the software requirments.

Action: Note the error description. Check that your system fulfills all software requirements.

[adynattr-01]

Message: Error with access to a dynamic attribute!

Cause: A fatal error has occurred.

Action: If this error re-occurs on starting simulation again, please export your models to an ADL file and contact your ADOxx consultant.

[adynattr-02]

Message: Illegal type for dynamic attribute!

Cause: A fatal error has occurred.

Action: If this error re-occurs on starting simulation again, please export your models to an ADL file and contact your ADOxx consultant.

[adynattr-03]

Message: Error during the access to internal representation of a dynamic attribute!

Cause: A fatal error has occurred.

Action: If this error re-occurs on starting simulation again, please export your models to an ADL file and contact your ADOxx consultant.

[adynpred-01]

Message: Activity was not yet executed!

<*Activityclass name*>: <*object name*>

Model: <*model name*>

Cause: In the specified activity, the attribute "Performer" contains an AQL expression which references "performer of" another activity. However the activity referenced here has not yet been executed and so the performer cannot be evaluated.

Action: Correct the entry in the "Performer" attribute of the specified activity.

[aeadsply-01]

Message: The diagram cannot be displayed, the names of the columns and rows are not unique. The following names are not unique.

<*terms*>

Cause: The names of the columns and rows are not unique.

Action: Update the entries so that the names of the columns and rows are unique.

[aeadsply-04]

Message: A result with this name already exists. You may not use duplicate names for results.

Cause: You have specified a name for a result function which is not unique within this agent.

Action: Enter a different name for the result.

[aeadsply-05]

Message: This is not a valid filename. Please select another file name.

Cause: An invalid filename was entered for the result.

Action: Enter a valid filename or deactivate the calculation of this result.

[aeadsply-06]

Message: The result history cannot be displayed.

Cause: You want to show the development of the agents results graphically. An error occurred while building this graphic.

Action: Contact your ADOxx consultant. If you know in which file the results are saved, then also send this file to your ADOxx consultant.

[aeadsply-07]

Message: Please enter a valid time string. Time values have to be entered in the following format: yy:ddd:hh:mm:ss where "yy" specifies the number of years, "ddd" specifies the number of days, "hh" the number of hours, "mm" the number of minutes and "ss" the number of seconds.

Cause: You have entered a time value in an invalid format.

Action: Please enter the time in a valid format.

[aeadsply-08]

Message: The name of a result that should be animated has changed. Please select the result again.

Cause: You have specified a result function for animation within this agent. However following the creation of this agent, the function has been changed or deleted.

Action: Select the result functions which should be animated.

[aeadsply-09]

Message: Please select an object class or remove the result "<*result name*>" from this agent.

Cause: You must select a class for the result function.

Action: Select a class or delete the result function.

[aeadsply-10]

Message: The name of this agent is not unique. Please select another name.

Cause: You are attempting to define two agents with the same name within your model.

Action: Select a different name for this agent.

[aeadsply-11]

Message: You have not selected any objects to be observed yet.

Cause: You have not selected any objects to be observed for this agent.

Action: Select the register "Basic configuration" and select the objects which you would like this agent to observe.

[aeadsply-12]

Message: You have to select at least one result and at least one object.

Cause: You have selected no results or no objects. Therefore no diagram can be shown.

Action: Select at least one object and at least one result.

[aeadsply-13]

Message: "<*result term*>". Syntax error in the specified formula.

Cause: There is a syntax error in the specified formula.

Action: Correct the formula.

[aeadsply-14]

Message: "<*result term*>". The specified formula contains an invalid result name

Cause: An invalid result formula was entered or the result formula was changed after it was specified.

Action: Correct the formula.

[aeadsply-16]

Message: No agents may be created in this model.

Cause: You have attempted to create agents in a model which cannot be simulated.

Action: Insure that the active model window refers to a model that can be simulated.

[aeadsply-17]

Message: Error in agent customising: The keyword AGENT must be followed by the name of the defined agent type.

Cause: A predefined agent contains errors.

Action: Contact your ADOxx administrator. The agent definition must be corrected in the library management component of the ADOxx Administration Toolkit.

[aeadsply-19]

Message: A maximum of 50 graphs can be displayed at the same time. Any surplus graphs will be ignored.

Cause: The animation of agent results and the presentation of the result history is in the form of function graphs. A maximum of 50 graphs can be displayed at the same time. However under the current conditions, more than 50 graphs are required to be displayed.

Action: End the simulation and open the configuration window of the relevant agent. Decrease the detail grade of the result functions which should be displayed.

[aeadsply-20]

Message: Agent could not be copied.

Cause: The selected agent could not be copied.

Action: Export the model in which the Agent is specified to ADL and then contact your ADOxx consultant.

[aeadsply-21]

Message: The agent could not be inserted.

Cause: Insertion of a previously copied agent is not possible.

Action: Save the model from which the agent was copied and the model in which you want to insert the agent to an ADL file, note the name of the agent and contact your ADOxx consultant.

[aeadsply-23]

Message: Please select an attribute or remove the result function "<*result*>" from this agent.

Cause: You must select an attribute for the agent specified.

Action: Select an attribute or delete the agent.

[aeagents-02]

Message: Error in a combining function of an agent.

Cause: A result function of type "Animation" or "Special formula" could not be initialised. The agent concerned is configured incorrectly. Perhaps the agent was imported from an ADL file which contained errors.

Action: Cancel the simulation, open the start window of the simulation again and click on the button "Agents". Open the configuration window of each agent marked here as incorrect and correct the configuration errors. As soon as you close the configuration windows again you will be informed if you have overlooked possible configuration errors. Correct these and start the simulation again. **Note:** To avoid similar problems in the long run, you should export the models simulated to an ADL file and inform your ADOxx consultant.

[aeagents-05]

Message: All specified result functions of the current agent could not be saved. (Too many functions have been specified). The configuration window of the current agent will show you which functions were saved successfully.

Cause: You tried to assign too many result functions to an agent. On average, about 15 functions can be assigned to an agent.

Note: The exact number depends on both the types of result functions assigned and the manner in which result functions, which were already assigned, were configured. There is no fixed upper limit for the number of possible result functions in an agent.)

Action: Check which result functions had been assigned successfully and which were not by using the tab cards in the configuration window of the agent and those which were not. Remove unnecessary result functions.

[aeagents-06]

Message: The result names of the following agent are not unique: "<*Agent name*>"

Cause: An agent is configured incorrectly. Perhaps the agent was imported from an invalid ADL file or you used a pre-defined agent, the definition of which is incorrect.

Action: Open this agent's configuration window and change the result name of all result functions so that all result names are unambiguous.

[aeagents-07]

Message: The following result will not be generated: "<*Name of the result function concerned*>: <*Name of the result not generated*>"

Cause: A result, which serves as basis for the result function listed, cannot be generated. The result function may have been configured incorrectly.

Action: Cancel the simulation and then open the start window of the simulation again. Click on the button "Agents". and correct the configuration of all the agents marked as incorrect in the window "Agents overview".

[aeagents-08]

Message: The configuration string of an agent could not be decoded. The following result type was not recognised: "<Unknown result type>"

Cause: An agent is configured incorrectly. Perhaps the agent was imported from an invalid ADL file or you used a predefined agent, the definition of which is incorrect.

Action: Correct the configuration of all the agents marked here as incorrect in the window "Agents Overview" (menu "Edit" - option "Agents").

[aeagents-09]

Message: An agent result cannot be determined due to one of the following reasons: the agent is located in a recursively called process, the agent is located in a subprocess which was called in a parallel way within the execution of one and the same main process.

Result: "<*Result name*>"

(Process: "<*Process name*>")

Cause: The model structure simulated cannot be evaluated by one of the agents created. There are two possible reasons for this:

1. the model in which the agent was created is called recursively. That is, the model calls itself again (maybe via other subprocesses).

2. the agent was created in a subprocess and this subprocess is called from a super ordinated process in a parallel way.

To put it in a more general context this means that two (or more) instances of a (sub)process which orginate from the same main process instance, are executed simultaneously.

Action: Avoid recursive model structures or change your model structures accordingly.

Avoid calling subprocesses in a parallel way within a model structure. One way to avoid a parallel structure is to move the agent from the subprocess, which is called in a parallel way, to the directly super ordinated process and assign the subprocess call objects as observed objects to the agent.

[aeagents-10]

Message: Error when evaluating the simulation through an agent. The generation of the following result has been deactivated to avoid further error messages: "<*Result name*>"

Cause: An unexpected error occurred while the result listed was calculated. Generation of this result has been cancelled.

Action: Note the result name listed, export the models simulated to an ADL file and inform your ADOxx consultant.

[aeagnrl-01]

Message: An error has occurred. Please contact your ADOxx consultant.

Cause: An unexpected error occurred in the agent function.

Action: Please note under which circumstances the error occurred and inform your ADOxx consultant.

[aeagnrl-02]

Message: A result function could not be identified.

Cause: The configuration data of an agent is incorrect.

Action: Open the window "Agent overview" and correct the configuration of all the agents maked as incorrect.

[aeagnrl-04]

Message: Agent specific settings could not be decoded.

Cause: An agent is configured incorrectly.

Action: Open the window "Agent Overview" and correct the configuration of all the agents marked as incorrect here.

[aeagnrl-06]

Message: The evaluation calendar of the agent is corrupt.

Cause: An agent is configured incorrectly. Perhaps the agent was imported from an invalid ADL file or you use a predefined agent, the definition of which is incorrect.

Action: Open the window "Agent Overview" and correct the configuration of all the agents marked as incorrect here.

Hint: To avoid this error in the future, you should inform your ADOxx consultant. The agent definition in the ADOxx Administration Toolkit is probably incorrect and must be corrected.

[aeagnrl-08]

Message: No simulation results are available.

Cause: The agent selected did not calculate results.

Action: You have probably already received other error messages concerning this agent. Correct these errors. As soon as all errors are corrected, the agent will calculate results again.

[aeagnrl-11]

Message: All result functions could not be applied to the agent.

Cause: You tried to assign further result functions to an agent. Not all the result functions selected could really be assigned. Perhaps you tried to assign too many result functions to the agent.

Action: Check which of the result functions were assigned successfully and which were not by using the tab cards in the agent's configuration window. Should you have assigned a large number of result functions to an agent, consider whether you can remove some of them. Only assign those result functions which are really needed to the agent.

[aeagnrl-13]

Message: This model was loaded in read-only mode.

This is why the action cannot be carried out. Load the model with write access.

Cause: You loaded the model concerned in read-only mode. You can only execute the action if you have read-write access to the model.

Action: Close the model and then reload it with read-write access. Then repeat the action.

[aeagnrl-14]

Message: The configuration values of this agent are corrupt.

Cause: The agent you are currently editing is configured incorrectly.

Action: Should you have just tried to create a predefined agent, please contact your ADOxx administrator. The agent definition is incorrect and must be corrected in the Library Management of the ADOxx Administration Toolkit. If this is not the case, open this agent's configuration window and check the configuration. As soon as you close the window again, you will be informed about possible configuration errors.

[aeagnrl-15]

Message: The definition of an agent's observed objects is invalid.

Cause: An agent is configured incorrectly. Perhaps the agent was imported from a corrupt ADL file or you are use a predefined agent, the definition of which is incorrect

Action: Cancel the simulation, re-open the start window of the simulation and click on the button "Agents". In the window "Agent Overview", assign new objects to be observed to all agents marked as incorrect. Start the simulation again.

[aeagnrl-18]

Message: The settings for the agent are too extensive. Please select fewer objects to be evaluated or results to be created. You can also distribute the tasks to several agents.

Cause: The agent has been assigned too many reference objects and/or tasks. The configuration of the agent therefore is too extensive and cannot be saved.

Action: Assign less tasks to the agent. Distribute/split the tasks among several agents.

© BOC Group 2008

[aearslt-01]

Message: The following file could not be opened with write access: "<*filename*>"

Cause: An agent was instructed to record the result history of one of its results. The agent could not open the file in which the result history should be saved. This could be due to one of the following reasons:

1. The file exists already and is write-protected.

2. The filename is incorrect.

3. The disk is full or has not been inserted.

Action: Depending on the cause, act according to one of the following recommendations:

1. Remove the write-protection from the existing file. (Attention: This means the file can be over-written any number of times in the future!)

2. Correct the filename in the configuration window of the agent concerned.

3. Clear some storage space on the disk or insert a disk.

Hint: If you close the error message and do not correct the error, the result history will not be recorded.

[aearslt-02]

Message: Error when writing the file "<*filename*>". The disk may be full.

Cause: An agent was instructed to record the result history of one of its results. An error occurred during the simulation, while the result history was being saved. The disk is either full or was removed. Note: In the case of very large models and long simulation duration the file containing the result history can also become very large. Even if you thought prior to the simulation that there was enough empty space on the disk, this may have been utilised already.

Action: Cancel the simulation and then empty some storage space for the result history file.

Note: If you close the error message and don't correct the error, the result history will not be recorded.

[aearslt-03]

Message: Error reading the file "<*filename*>".

Cause: During the simulation an agent recorded the result history of one of its results in the file listed. Now this result history is to be displayed graphically. For this reason it is being read in again from the file. An error occurred, while the file was read in.

Perhaps the disk on which the result history was saved has been removed.

Action: Insert the disk again and restart the display of the result history.

[aeasubag-02]

Message: The following result generating function of an agent has not been configured completely and is thus deactivated: "<*result name*>"

Cause: You use a pre-defined agent. This agent's definition is incorrect

Action: Cancel the simulation, then open the start window of the simulation again and click on the button "Agents". The window "Agent overview" opens up. Correct the configuration of all agents marked as incorrect here.

Hint: To avoid this error in the future, you should note down the result name listed (and the agent type concerned, if you know it) and inform your ADOxx consultant. The agent definition in the Library Management of the ADOxx Administration Toolkit is probably incorrect and must be corrected.

[aeasubag-03]

Message: The initialisation of an agent failed. The result "<*result name*>" cannot be initialised because "<*name*>" is not a valid name. Please contact your ADOxx administrator.

Cause: You are using a predefined agent. An invalid name was used within the definition of this agent.

Action: Note down the result and the name listed (and also the agent type concerned, if you know it) and inform your ADOxx consultant. The agent definition in the Library Management of the ADOxx Administration Toolkit must be corrected.

[aeasubag-04]

Message: The following result cannot be generated because not all results on which this result is based are available:

"<*Name of the Result Function Concerned*>"

Cause: The calculation formula in the result function listed is incorrect. It references other result functions which do not exist

Action: Open the configuration window of the concerned Result Function and correct the calculation formula.

[aeasubag-05]

Message: The following result cannot be generated because of a syntax error in the required formula:

"<*Result function name*>"

Cause: The calculation formula for the result function listed is incorrect. It contains a syntax error.

Action: Open the configuration window of the agent concerned and correct the calculation formula.

[aeasubag-06]

Message: Some specified calendars are too complex. The following result cannot be generated:

"<*Result name*>"

Cause: The calendars of performers, resources and/or agents in the models simulated are too complex. The result listed(which is always of the type "Workload") therefore cannot be calulated

Action: Use a smaller number of different day profiles in the calendars.

[aeasubag-07]

Message: "<*Result name*>"

This result cannot be generated. (Possibly other results which are required for the generation are not available under the current settings.)

Cause: An agent's result could not be generated. This may be for one of the following reasons:

1. The result listed could not be initialised. In this case, you already received an appropriate error message at the beginning of the simulation.

2. The result was calculated based on other results. One or more of these other results is not available.

Either an error occurred while the result not available was being calculated (in this case, you already received an appropriate error message at the beginning of the simulation), or the result not available cannot be displayed under the current settings of the result representation (That is the most probable cause.)

Action: Depending on the cause:

1. Correct the error that was originally reported.

2. Change the settings concerning the result representation.

Please note that some results can only be represented "per <*time period*>" (such as "per year"). As long as the option "per process" is selected, these results will not be available. Then display the results of the agent concerned again.

[aeasubag-08]

Message: The formula for the result "<*result name*>" contains too many different result names. The formula cannot be evaluated.

Cause: A calculation formula may contain at most 26 different results. This upper limit was exceeded by the calculation formula of the result function listed.

Action: Simplify the calculation formula.

[aevalrel-02]

Message: Wrong value at the from object.

Cause: You have entered a non-numeric value as the filter for the from-object in the relation table for a numeric attribute.

Action: Enter a number.

[aevalrel-03]

Message: Wrong value at the to-object.

Cause: You have entered a non-numeric value as the filter for the to-object in the relation table for a numeric attribute.

Action: Enter a number.

[aevalrel-04]

Message: Wrong value at the relation.

Cause: You have entered a non-numeric value as the filter for the relation in the relation table for a numeric attribute.

Action: Enter a number.

[aevalrel-05]

Message: Error in application library.

Cause: The definition of the selected relation table contains errors in the application library.

Action: Contact your ADOxx administrator or your ADOxx consultant.

[aevkres-01]

Message: Error in audit trail (line <*line number*>):
The process "<*process name*>" was not found.

Cause: An error occurred while reading the audit trail. An unknown process was referenced.

Action: Replace the invalid process with a valid one in the audit trail.

[aevkres-02]

Message: Error in audit trail (row <*line number*>):
The process instance "<*instance name*>" was not found.

Cause: An error occurred while reading the audit trail. An unknown instance was referenced.

Action: Replace the invalid instance name with a valid one in the audit trail.

[aevkres-03]

Message: Error in audit trail (line <*line number*>):
The activity "<*activity name*>" could not be found.

Cause: An error occurred while reading the audit trail. An unknown activity was referenced.

Action: Replace the invalid activity name with a valid one in the audit trail.

[aevkres-04]

Message: Error in audit trail (line <*line number*>):
The activity instance "<*instance name*>" could not be found.

Cause: An error occurred while reading the audit trail. An unknown activity instance was referenced.

Action: Replace the invalid instance name with a valid one in the audit trail.

[aevkres-05]

Message: Error in audit trail (line <*line number*>):
The file is not a valid audit trail.

Cause: A syntax error occurred while reading the audit trail.

Action: Correct the syntax in the specified line of the audit trail.

[aevkres-06]

Message: Error in audit trail (line <*line number*>):
This line is too long. A maximum of 1023 characters is allowed.

Cause: The audit trail contains a line with more than 1023 characters.

Action: Check the specified line and reduce it.

[aexpappl-01]

Message: The attribute "Documentation configuration" contains errors (row <*no:*>)!
The format model has been ignored, since a required attribute is missing: <*attribute name*>

Cause: The content of the library attribute "Documentation configuration" is faulty: An attribute of the class "__LibraryMetaData__" of the BP library is referenced but does not exist.

Action: Contact your ADOxx administrator.

[aexpappl-06]

Message: Error during the creation of the help file.

Cause: These temporary files are created in the current directory or in the directory at which the enviroment variable TEMP is pointing.

Action: Ensure that the enviroment variable TEMP exists and points to a directory in which enough memory space is available and in which you are allowed to create files.

[aexpappl-08]

Message: The file <*File name*> does not exist!

Cause: During the ADOxx customising, it was determined that one or more files (e.g. graphic logo files) should be copied from the installation directory to the target directory. The file specified in the error message could not be opened.

Action: Verify your ADOxx installation using the file list and create or complete if necessary missing or damaged files.

Hint: The documentation process will continue despite this error but the generated documentation contents will not contain data related to the damaged or missing files e.g. a logo.

[aexpappl-10]

Message: The attribute "documentation configuration" is faulty!

The name of the format model is missing, the format model has been ignored.

Cause: In the attribute "documentation configuration", no name was specified during the definition of a format model (using a FORMAT or an EXPORT element).

Action: Correct this error using the Administration Toolkit or contact your ADOxx administrator.

[aexpappl-13]

Message: The attribute "documentation configuration" contains errors (row <*no:*>)!

The element "<*element name*>" is faulty.

Cause: In the attribute "Documentation configuration", an invalid name was entered.

Action: Correct the spelling of the element name or delete the invalid element using the Administration Toolkit or contact your ADOxx administrator.

[aexpappl-14]

Message: The attribute "Documentation configuration" contains errors (row <*no:*>)!

SOURCE elements must always be inside an EXPORT element.

Cause: In the attribute "documentation configuration", an element was entered at the wrong position.

Action: Correct the error using the Administration Toolkit or contact your ADOxx administrator.

[aexpappl-15]

Message: The attribute "Documentation configuration" contains errors (row <*no:*>)!

The SOURCE element "<*element name*>" does not exist.

Cause: In the attribute "Documentation configuration", an invalid element name was given to a SOURCE element.

Action: Correct the element name using the Administration Toolkit or contact your ADOxx administrator.

[aexpappl-16]

Message: The attribute "Documentation configuration" contains errors (row <*no:*>)!

LIBRARY elements must be inside SOURCE "Model" elements.

Cause: In the attribute "Documentation configuration", an element was entered at the wrong position.

Action: Correct the error using the Administration Toolkit or contact your ADOxx administrator.

[aexpappl-17]

Message: Error in the attribute "Documentation configuration".

The attribute "<*attribute name*>" does not exist.

Cause: The content of the library attribute "Documentation configuration" is faulty: An attribute from the class "__LibraryMetaData__" of the BP library is referenced but does not exist.

Action: Contact your ADOxx administrator.

[aexpappl-18]

Message: Error in the attribute "Documentation configuration".

Characteristics <*Characteristics name*>. The attribute <*attribute name*> has the wrong type.

Cause: In the library attribute "Documentation configuration", an attribute of the class "__LibraryMetaData__" of the BP library is referenced but has the wrong type.

Action: Contact your ADOxx administrator.

[aexpappl-19]

Message: Error in the attribute "Documentation configuration".

Characteristics <*characteristics name*>.

The value "<*value denomination*>" is not allowed in the attribute "<*attribute name*>".

Valid values are: <*value area*>

Cause: In the library attribute "Documentation configuration", an enumeration attribute of the class "__LibraryMetaData__" of the BP library is referenced but its value area contains invalid values.

Action: Contact your ADOxx administrator.

[aexpappl-20]

Message: Error in the attribute "Documentation configuration".

The value "<*value denomination*>" is not allowed in the attribute "<*attribute name*>.

Valid values are: <*value area*>

Cause: In the library attribute "Documentation configuration", an enumeration attribute of the class "__LibraryMetaData__" of the BP library is referenced but its value is not in the valid area.

Action: Contact your ADOxx administrator.

[aexpappl-21]

Message: Error in the attribute "Documentation configuration".

Characteristics <*characteristics name*>.

The attribute "<*attribute name*>" is not a class attribute.

Cause: The content of the attribute "Documentation configuration" is faulty: An attribute from the class "__LibraryMetaData__" of the BP library is referenced but is not a class attribute.

Action: Contact your ADOxx administrator.

[aexpappl-22]

Message: The file "<*file name*>" could not be opened!

Cause: A file required for the documentation generation could not be opened.

Action: Verify your ADOxx installation using the file list and replace or complete if necessary missing or damaged files.

[aexpappl-23]

Message: The file "<*file name*>" could not be overwritten! Make sure that the directory to which the file should be written does exist and that you own the required write access rights.

Cause: The target file specified for the documentation could not be opened with write access.

Action: Verify the write access to the target directory, in which the file will be generated. You can create documentation only in directories for which you own write access. Then verify, if a file with the specified name exists already in the target directory and delete this file if necessary.

[aexpappl-25]

Message: Invalid Value:

The value "<*Value*>" could not be saved.

Cause: The configuration of the documentation generation is faulty.

Action: Contact your ADOxx administrator.

[aexpappl-26]

Message: Error in the application library:

No documentation export was defined for the menu item "<*name*>".

Cause: No format model exists in the attribute "Documentation configuration" for the menu item called (using a FORMAT or EXPORT element).

Action: Control the spelling of the menu items called and if necessary correct errors or adapt the attribute "documentation configuration" using the Administration Toolkit or contact your ADOxx administrator.

[aexpappl-27]

Message: The class "__LibraryMetaData__" does not exist.

Cause: The class "__LibraryMetaData__" required for the documentation generation does not exist in the BP library .

Action: Contact your ADOxx administrator.

[aexpappl-28]

Message: The user settings could not be saved to the database, since the user profile exceeds the maximum length.

Cause: The database attribute to save the user-specific settings is not large enough to store all settings.

Action: The user settings could not be saved to the database.

Hint: The settings carried out for the documentation are lost after you close ADOxx. But the settings are valid for the documentation which will be immediately generated.

[aexpappl-29]

Message: An error has occurred during the conversion of the library attribute "Documentation configuration": The attribute "<*attribute name*>" could not be created in the class "_Library MetaData_".

Cause: An attribute with the specified name already exists in the class "_Library MetaData_".

Action: Contact your ADOxx administrator.

[aexpappl-30]

Message: Error in the command "EXEC_ACFILTER"! The attribute "<*attribute name*>" does not exist within the class "_Library MetaData_".

Cause: The specified attribute is referenced within the AdoScript command EXEC_ACFILTER, but does not exist.

Action: Contact your ADOxx administrator.

[aexpappl-31]

Message: The converted library attribute "Documentation configuration" could not be saved to the database as it is too long.

Cause: The configuration of the Documentation Component is too big to be converted.

Action: Carry out a manual conversion of the library attribute or simplify the configuration by deleting single menu items.

[aexpappl-32]

Message: Error in the application library:

No settings dialogue has been defined.

Cause: No format model (using the DIALOG element) exists in the attribute "documentation configuration" for the called menu item.

Action: Correct the error if necessary (adapt the attribute "documentation configuration" using the Administration Toolkit) or contact your ADOxx administrator.

[aexpappl-33]

Message: The settings for the attribute- and class filter could not be saved, as the length of the settings exceeds the maximum length allowed for the attribute "<*Attribute name*>" of the class "__LibraryMetaData__".

Cause: The settings you have defined in the attribute and class filter cannot be saved to the ADOxx database, because of their size.

Action: When carrying out the settings in the attribute and class filter, use the predefined mode (button "Load mode") to reduce the complexity and therefore the size.

[aexpappl-34]

Message: Error in the attribute "documentation configuration",

Characteristics "<*characteristics*>".

The value "<*value*>" of the attribute "<*attribute name1*>" has not been defined in the attribute "<*attribute name2*>".

Cause: The specified value can also be the name of an attribute (*attribute name2*) which contains the domain.

Action: Define the value in the domain or change the value of the attribute to a value from the domain.

[aexpappl-35]

Message: The target directory "<*directory name*>" is not a valid directory.

Cause: The specified directory name is invalid.

Action: Specify a valid directory name.

(The characters **/ \ : * ? < >** and **|** are not allowed in directory names.)

[aexpappl-36]

Message: No LOG file could be created!

Documentation generation will continue.

Cause: The creation of the LOG file is wrong.

Action: Verify if a file with this name exists already and if it is write-protected or already opened.

If necessary, verify if you possess write rights on the target directory and if there is enough memory available in the hard disc.

[aexpappl-37]

Message: The file "<*file name*>" is referenced in the "documentation configuration" with "copy", but has not been specified in the "requirefile" area.

Contact your ADOxx administrator.

Cause: The specified file has been listed in the library attribute "documentation configuration" as a copy parameter and not as a "requirefile" parameter.

Action: In the ADOxx Administration Toolkit, define the specified file in the library attribute "documentation configuration" as a "requirefile" parameter or contact your ADOxx administrator.

[aexpappl-38]

Message: The ADOxx installation directory cannot be used as a target directory.

Cause: You have selected the ADOxx installation directory as the target directory for the documentation generation or the export. This is not allowed for security reasons.

Action: Select another target directory. If you have write access rights only on the ADOxx installation directory, create a sub directory in it and select a file as target file in this sub directory.

[aexpappl-39]

Message: The format of the font table "<file name>" is not valid.

The default font table will be used.

Cause: A codepage conversion table you wanted to use contains an error and is thus invalid.

Action: Contact your ADOxx administratorParallelity

[aexpdlg-01]

Message: The user specific settings are inconsistent.

Standard values will be loaded.

Cause: The user profile is in an inconsistent state. This can happen e.g., if a user is assigned to a new application library.

Action: Close the error message, standard values will be loaded and carry out the settings in the attribute and class filter again.

[aexplore-01]

Message: Error while moving a model group!

"<*Model group*>" can not be moved to "<*target model group*>".

Cause: Either the target model group is write-protected or a subgroup of the group to be moved.

Action: Move model groups only into other groups where you have write-access and are not into sub groups of the group to be moved.

[aexplore-02]

Message: Error while copying a model group!

"<*Model group*>" can not be copied to "<*target model group*>".

Cause: Either the target model group is write-protected or a subgroup of the group to be moved.

Action: Copy model groups only into other groups where you have write-access and are not into sub groups of the group to be moved.

[aexplore-03]

Message: Error while copying model groups!

Just disjoint model groups can be copied in one go.

Cause: When copying a model group, all sub groups are automatically copied with it. Thus when selecting model groups for copying, it is neither allowed nor expedient to select sub groups.

Action: Adjust your model group selection accordingly.

[aexplore-04]

Message: Error while creating a new model group!

Access permissions to the parent model group may have been changed or the parent model group may have been deleted by another user.

Refresh the model explorer, then try again.

Cause: Creating a new model group is not possible, as your selected parent group is no longer accessible. Either this group was deleted in the meantime by another user or the access rights were changed by the ADOxx-Administrator.

Action: Refresh the model group view before creating new model groups.

[aexport-01]

Message: Unable to open export file.

Cause: The file to be exported could not be created, since the data carrier/medium specified (e.g. floppy disk) is write-protected or does not contain enough free space.

Action: Remove the write-protection from the data carrier/medium or create some new space or assign a valid file name and restart the export.

[aexport-05]

Message: No models were exported.

Cause: Due to errors during the loading process no models were exported.

Action: Quit ADOxx and restart it. If the error message re-occurs during the next attempt, please inform your database administrator.

[aexport-06]

Message: Unknown model type.

Cause: An unknown model type was encountered during the export process. Possibly, the connection to the ADOxx database was aborted.

Action: Quit ADOxx and re-start it. If the error message re-occurs during the next attempt, please inform your database administrator.

[aexport-07]

Message: Unknown library type.

Cause: An unknown library type was encountered during the export process. Possibly, the connection to the ADOxx database was aborted.

Action: Quit ADOxx and re-start it. If the error message re-occurs during the next attempt, please inform your database administrator.

[aexport-08]

Message: Error in export: <*Error*>

Cause: The error listed occurred during the export process. Possibly, the connection to the ADOxx database has been aborted.

Action: Quit ADOxx and re-start it. If the error message re-occurs during the next attempt, please inform your database administrator.

[aexport-10]

Message: Error writing to file "<*filename*>".

Cause: During the export process an error occurred while writing to the file specified.

Action: There is possibly not enough space left on the data carrier/medium. If there is still enough space, start the export process again.

[aexport-11]

Message: Error in library "<*library name*>".

Do you want to continue anyway?

Cause: During the export process an error occurred either while a library was about to be loaded or while determining the libraries assigned.

Action: Cancel the export process and re-start it. If you continue the export process, the ADL file will not contain the relevant model groups access information.

[aexport-12]

Message: Error when exporting files belonging to the library!

Cause: When loading a file belonging to the library from the ADOxx database, a serious error has occurred. The connection to the database may be aborted.

Action: Close ADOxx, start it again and repeat the export. Should the message appear again, contact your database administrator.

[aexpsort-01]

Message: Class <*class name*>

The attribute specified as sort criteria

<*Attribute name*>

does not exist.

The default value will be used.

Should this message be shown again in the future?

Cause: Your application library contains errors.

Action: Contact your ADOxx administrator.

[aexpsort-02]

Message: Class <*class name*>

The attribute specified as sorting criteria

<*attribute name*>

is of an invalid type.

The default value will be used.

Should thus message be shown again in the future?

Cause: Your application library contains errors.

Action: Contact your ADOxx administrator.

[aexpsort-03]

Message: Class <*class name*>:
The attribute specified as a duplicate criterion
"<*attribute name*>"
does not exist.
The default value will be used.
Should this message be shown again in the future?

Cause: Your application library contains errors.

Action: Contact your ADOxx consultant.

[aexpsort-04]

Message: Class <*class name*>:
The attribute specified as a duplicate criterion
"<*attribute name*>"
is of an invalid type.
The default value will be used.
Should this message be shown again in the future?

Cause: An attribute of an invalid type is defined as duplicate criterion in your application library.

Action: Contact your ADOxx consultant.

[afile-01]

Message: The file <*file name*> could not be opened with read access.

Cause: An error occurred, when attempting to open the specified file with read access.

Action: Verify, whether the specified directory as well as the specified file exist and whether you possess the required access rights to this file.

[afile-02]

Message: The file <*file name*> could not be opened with write access.
The file may be used by another program.

Cause: An error occurred, when attempting to open the specified file with write access.

Action: Verify, whether the specified file is currently opened, whether the specified directory as well as the specified file exist and whether you possess the required access rights for this file.

[afile-03]

Message: Error when reading the file "<*file name*>".

Cause: An error occurred when reading the specified file.

Action: Repeat the process. Should the error appear again, manually copy the specified file to the target directory. If successful, verify your ADOxx installation using the file list. If necessary, reinstall ADOxx.

[afile-04]

Message: Error when reading the file "<*file name*>".

Cause: An error occurred when writing the specified file to the target directory.

Action: Verify if there is enough memory space available in the target directory. If the target directory is a network drive, check if there is still a connection. Then repeat the process.

[afile-05]

Message: The file "<*file name*>" could not be deleted.

Cause: An error has occurred when deleting the specified file.

Action: Verify, if the specified directory and/or the specified file exist, if you have the required rights to delete this file and if the file is already opened.

[afile-06]

Message: The file "<*file name*>" does not exist.

Cause: The specified file could not be found.

Action: Verify if the specified directory and/or the specified file exist and whether you have the required rights to access this file.

[afile-07]

Message: The file "<*file name*>" exists already .

Cause: You have attempted to create or copy the specified file.

Action: The file will be created only if the already existing file has been renamed or deleted.

[afile-08]

Message: The file "<*file name*>" cannot be closed.

Cause: An error has occurred when trying to close the specified file.

Action: Repeat the process or contact your ADOxx administrator.

[afile-09]

Message: No temporary file could be created.

Cause: An error has occurred when trying to create a temporary file in the temporary folder.

Action: Verify if enough memory space is available in the hard disk, if the system variable "TEMP" is set and if you have the required access rights for the directory specified in the system variable "TEMP".

[afile-10]

Message: The file name "<*file name*>" is invalid.

Cause: The specified file name is not in a correct format or contains invalid characters.

Action: When entering file names, make sure that total length of paths and file names does not exceed 120 figures and no more than seven directories are contained in the path. Moreover, the characters **/ \ : * ? < >** and **|** are not valid.

[afile-11]

Message: The file "<*file name*>" could not be renamed.

Cause: An error has occurred when attempting to rename the specified file.

Action: Verify if a file with the same name already exists . If necessary assign another file name.

[afile-12]

Message: The file "<*file name*>" could not be moved.

Cause: An error has occurred when attempting to move the specified file.

Action: Verify if the specified directory exists and if a file with the same name is contained in this directory.

[afile-13]

Message: The specified database directory is invalid.

Cause: The database directory you have specified could not be found.

Action: Verify the specified database directory and correct your entry accordingly.

[afile-14]

Message: The file "<*file name*>" could not be accessed, since it is not open!

Cause: You have tried to access a file which is not open.

Action: Open the file before you access it.

[afile-16]

Message: The file "<*file name*>" already exists for the library "<*library name*>".

Cause: You have tried to import or to copy the specified file to the folder of the specified application library.

Action: The file can only be imported or copied with the specified name once the file already existing in the folder of the application library has been renamed or deleted.

[afile-17]

Message: The name of the file "<*file name*>" is too long.

Cause: The file <*file name*> could not be stored in the database becuase the name of the file is too long.

Action: Adjust the length of the file name accordingly. Note that the length of the external file name within the library (including the file extension) and the length of the library name must not exceed 249 characters.

[agdt-01]

Message: An error has occurred when sorting the start model!

Cause: The model could not be sorted. Connectors are possibly missing in the model to be arranged.

Action: Check the model to be arranged for missing connectors and repeat the process. Should the error appear again, close ADOxx and start it again. If the error is not corrected, contact your ADOxx administrator.

[agdt-02]

Message: Positions could not be assigned!

Cause: An error occurred, while positions were being assigned

Action: Quit ADOxx and start it again. Should this error re-occur, contact your ADOxx administrator.

[agdt-03]

Message: Bendpoints could not be inserted!

Cause: An error occurred while the bendpoints were being inserted.

Action: Should this error re-occur, align the connectors without inserting bendpoints or contact your ADOxx administrator.

[agdt-04]

Message: Positions could not be assigned with bendpoints!

Cause: An error occurred while positions with bendpoints were being assigned.

Action: Quit ADOxx and start it again. Should this error re-occur, contact your ADOxx administrator.

[agdt-05]

Message: Method median-down could not be applied!

Cause: An error occurred while the median-down method was being applied.

Action: Quit ADOxx and start it again. Should this error re-occur, align the connectors without applying the median-down method or contact your ADOxx administrator.

[agdt-06]

Message: Median-up method could not be applied!

Cause: An error occurred while the median-up method was being applied.

Action: Quit ADOxx and start it again. Should this error re-occur, align the connectors without applying the median-up method or contact your ADOxx administrator.

[agdt-07]

Message: Pairwise exchange could not be applied!

Cause: An error occurred while pair wise exchange was being applied.

Action: Quit ADOxx and start it again. Should this error re-occur, align the connectors without applying pairwise exchange or contact your ADOxx administrator.

[agdt-08]

Message: Pendulum-down method could not be applied.

Cause: An error occurred while the pendulum-down method was being applied.

Action: Quit ADOxx and restart it. Should this error re-occur, align the connectors without applying the pendulum-down method or contact your ADOxx administrator.

[agdt-09]

Message: Pendulum-up method could not be applied!

Cause: An error occurred while the pendulum-up method was being applied.

Action: Quit ADOxx and start it again. Should this error re-occur, align the connectors without applying the pendulum-up method or contact your ADOxx administrator.

[agdt-10]

Message: Horizontal mirror could not be applied!

Cause: An error occurred while applying the horizontal mirror.

Action: Quit ADOxx and start it again. Should this error re-occur, contact your ADOxx administrator.

[agdt-11]

Message: Vertical mirror could not be applied!

Cause: An error occurred while applying the vertical mirror.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-12]

Message: It was not possible to arrange downwards!

Cause: An error occurred while arranging downwards.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-13]

Message: It was not possible to arrange upwards!

Cause: An error occurred while arranging downwards.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-14]

Message: It was not possible to arrange to the right!

Cause: An error occurred while arranging to the right.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-15]

Message: It was not possible to arrange to the left!

Cause: An error occurred while arranging to the left.

Action: Quit ADOxx and start it again, contact your ADOxx administrator.

[agdt-16]

Message: The rubber band method could not be applied!

© BOC Group 2008

Cause: An error occurred while applying the rubber band method.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-18]

Message: Cycle minimisation could not be applied!

Cause: An error occurred while applying cycle minimisation.

Action: The model contains too many cycles and therefore may not be arranged in the current form. Try to simplify the model and to reduce the number of cycles.

[agdt-19]

Message: New model could not be written!

Cause: An error occurred while the newly arranged model was about to be transferred.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-20]

Message: At least one relation couldn't be inverted according to the parameter value!

Cause: An error occured while attempting to turn the relations of the model according to the parameter value.

Action: Change the parameter values of the alignment function or contact your ADOxx administrator.

[agdt-22]

Message: Process-hierarchy could not be generated because two objects would have the same name!

Cause: An attempt was made to create an object with a name currently assigned to an existing object.

Action: Delete the model, into which the process hierarchy was to be generated and then start generating the process hierarchy again.

[agdt-23]

Message: Could not position one object of the model to be arranged!

Cause: An error occurred while positioning the object.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-24]

Message: Could not number the objects of the model!

Cause: An error occurred, while the model objects were being numbered.

Action: Quit and restart ADOxx. Should this error re-occur, contact your ADOxx administrator.

[agdt-26]

Message: The selected model contains no model object to be arranged! No arrangement could be carried out.

Cause: The arrangement function has been used on a model which contains no object and therefore has been closed.

Action: Use the arrangement function only on models with objects you want to re-place.

[agdt-27]

Message: The selected model contains no model references to be displayed as a process-hierarchy!

Cause: You selected a model without model references for generating a graphical process hierarchy.

Action: Insert the necessary model references or select a different model.

[agdt-28]

Message: The selected model involves no objects for numbering! Cannot number objects.

Cause: You selected a model which does not contain any objects or objects which are not defined for numbering when attempting to number the objects.

Action: Model the object to be numbered or select a different model.

[agdt-29]

Message: Name missing after the key word "<*profile*>"!

Cause: In the configuration of the arrangement function, a profile and a name have been defined. The profile is either an arrangement profile **PROFILE**, a hierarchy profile **HIERPROFILE** or an enumeration profile **ENUMPROFILE**.

Action: Once you have specified a profile in the configuration of the arrangement function, you need to give a name to the profile. Specify the name between double quotation marks or if necessary contact your ADOxx administrator.

[agdt-30]

Message: Faulty initialising with expression for the key word "<*element*>"!

Cause: The specified element could not be initialised. The error appears, when an element of the same name has already been defined, when a previous definition refers to the same model type/the same class, relation etc. or when the specified model type/the specified class/relation class does not exist.

Element is one of the following syntax elements:

PROFILE, **CLASSPAIRPAR**, **HIERPROFILE**, **HIERmodel type**, **ENUMPROFILE**, **ENUMmodel type**, **ENUMCLASS** or **ENUMREL**.

Action: Verify the configuration of the arrangement function and if necessary contact your ADOxx administrator.

[agdt-31]

Message: Wrong key word "<*key word*>"!

Cause: The specified key word is not authorised in this context or makes no sense here.

Action: Verify the configuration of the arrangement function and if necessary contact your ADOxx administrator.

[agdt-32]

Message: Faulty value after the key word "<*key word*>"!

Cause: This error appears for one of the following reasons:

The name of the model type has not been specified or is invalid (for **HIERmodel type**, **ENUMmodel type**, **CLASSmodel type**, **DEFmodel type** or **CLASSPAIRmodel type**).

The name of the attribute, the class or the relation is missing or is invalid (for **CLASSPAR**, **CLASSPAIRPAR**, **HIERATTRIB**, **HIERCLASS**, **HIERUSECLASS**, **HIERUSEREL**, **HIERUSEATTRIB**, **ENUMCLASS** or **ENUMREL**).

The parameter **dist:** for **DOUBLEDP** has not been specified.

The parameter **vertdist:** for **CHNGSIZE** has not been specified.

The parameter **use:** for **HIERPROFILE** or **usetype:** for **HIERmodel type** has not been specified, the profile or the model type does not exist.

Action: Check/complete the configuration of the application function and if necessary contact your ADOxx administrator.

[agenpar-01]

Message: Syntactical error! '(' expected!

Cause: An opening bracket is expected for the parse of a random generator.

Action: Check if an opening bracket is missing in the expression you have specified and if necessary insert it. The second row of the error message indicates where in ADOxx a bracket is expected.

[agenpar-02]

Message: Syntax error!

')' expected!

Cause: When translating a random generator, a closing bracket was expected.

Action: Check that the syntax of your given expression is correct and insert the required closing bracket. The second line of the error message gives you an idea of where in ADOxx a bracket is expected.

[agenpar-03]

Message: The argument for an exponential distribution cannot be 0!

Cause: You have entered 0 as the argument for an exponential distribution. This value is not permitted.

Action: Replace the 0 with another value.

[agenpar-04]

Message: The first argument of a uniform distribution must be smaller than the second!

Cause: The first argument which you have entered for a uniform distribution is smaller than the second.

Action: Check that you have entered the arguments in the correct order.

[agenpar-05]

Message: Syntax error! Character string or ' expected!

Cause: When translating a random generator, a single quotation mark or a character string was expected.

Action: Check if you are missing a character string or quotation mark from your expression and add in whatever is missing. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-06]

Message: Syntax error! ' expected!

Cause: During translation of a random generator a quotation mark was missing.

Action: Check if you are missing a quotation mark from your expression and add it in where necessary. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-07]

Message: Syntax error! Constant, variable or distribution expected!

© BOC Group 2008

Cause: When checking a random generator, a constant, variable or distribution was expected.

Action: Check if you have left out a constant, a variable or a random generator from your expression. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-08]

Message: Syntax error! Multiplication or division are not defined for character strings!

Cause: You are attempting to multiply or divide variables or constants which are characters within the expression you have entered.

Action: Change the type of the variable referenced by the expression, replace the constants with numerical constants or replace the multiplication or division operator with one which is valid for character strings (e.g. +).

[agenpar-09]

Message: Syntax error! Addition or subtraction of two terms of different types!

Cause: The expression you have entered attempts to add or subtract two variables or constants of different types.

Action: Change the type in the variable referenced by this expression so that the operation can be carried out.

[agenpar-10]

Message: Syntax error! Subtraction is not defined for character strings!

Cause: You are attempting to subtract variables or constants which are strings within the expression you have defined.

Action: Change the type in the variable referenced by this expression or change the constant to a numeric constant or change the operator to one which is permitted for character strings in order that the operation can be carried out.

[agenpar-11]

Message: Syntax error! Semicolon or end expected!

Cause: ADOxx expects an end to your expression or a semicolon as a divider between two expressions.

Action: If you wish to specify more than one expression in your random generator, specify these by semicolons. If this is not required, check that you have not forgotten to enter an operator between two terms. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-12]

Message: Syntax error! Numeric parameter expected!

Cause: ADOxx expected a numeric parameter as an argument for the distribution.

Action: Replace the specified argument with a numeric constant. Variables or constants are not permitted in this case. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-13]

Message: Syntax error! Semicolon expected!

Cause: When translating a random generator, a semicolon was expected.

Action: Check where you are missing a semicolon from your expression and add it in where necessary. The second line of the error message gives you an idea of where in ADOxx this is expected. A semicolon can also be used to separate a number of expressions in a random generator.

[agenpar-14]

Message: Syntax error! Character string expected!

Cause: ADOxx expected a character string as parameter to the distribution.

Action: Replace the appropriate argument with a character string. Variables or constants which begin with a figure are not permitted in this case. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-15]

Message: Syntax error! Probability expected!

Cause: ADOxx expected a probability as argument to the distribution.

Action: Replace the appropriate argument with a probability. Variables or characters are not allowed in this case. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-16]

Message: A probability must be between 0 and 1!

Cause: You have entered a probability which is smaller than 0 or greater than 1.

Action: Replace the probability with a valid value. Please note that probability must be entered as 0.n or 0,n. The second line of the error message gives you an idea of where in ADOxx this is expected.

[agenpar-17]

Message: The sum of the probabilities must be 1!

Cause: The sum of the probabilities of a discrete distribution must be exactly 1.

Action: Change the probabilities of the created symbols of the discrete distribution so that they sum to 1.

[agenpar-18]

Message: The arrival time cannot be negative.

Cause: If a uniform distribution is used then the lower bound must not be smaller than 0.

Action: When using a uniform distribution to determine the arrival times, the arguments used must be positive.

[agenpar-19]

Message: The upper bound must be greater than 0!

Cause: When using a uniform distribution to specify arrival frequency, then upper bound must be greater than 0.

Action: Increase the value of the upper bound (second argument) so that it is greater than 0.

[aglochan-01]

Message: No objects matching the criteria were found.

Cause: The evaluation of the predefined or user-defined criteria did not find any result objects.

Action: Formulate the selection criteria so that at least one object is found.

[aglochan-02]

Message: Select at least one attribute.

Cause: No attribute to be changed was selected from that attribute list.

Action: Select at least one attribute so that the change can be executed.

[aglochan-04]

Message: Attributes could not be changed.

Cause: An error occurred while trying to update the attributes.

Action: Close the model and open it again. Try to update the attributes once more. If the error re-occurs contact your system administrator.

[aglochan-05]

Message: The selection contains objects or connectors of different types or relations. Please change criteria to select only objects or connectors of the same type.

Cause: The selection criteria contains objects of different types. The global change is only possible for attribute of objects of a class or connectors of a relation class.

Action: Change the selection criteria so that only objects of one type or connectors of one relation class are selected.

[aglochan-06]

Message: Invalid input for attribute<*attribute name*>: <*attribute value*>

Cause: An invalid value was entered for the attribute given - e.g. a value of the wrong type.

Action: Enter a valid value.

[aglochan-08]

Message: The selected attribute combination is not allowed. Please select an attribute combination with at least one activated change mode.

Cause: The selected attributes have no possible changes in common.

Action: Select only attributes which have at least one change mode in common.

[aglochan-09]

Message: Please select one attribute which should be changed.

Cause: When "Taking over" an attribute, the attribute to which the value should be copied has not been specified.

Action: Please select an attribute to which the input attribute should be "taken over".

[aglochan-11]

Message: The AQL expression is syntactically incorrect!

Cause: The AQL expression to find the objects to be changed is syntactically incorrect.

Action: Enter a syntactically correct AQL expression.

[aglochan-12]

Message: There are no changeable attributes!

Cause: The class of the object to be changed contains no attributes. A change is therefore not possible.

Action: Select other objects.

[aglochan-13]

Message: Wrong floating point value!

Cause: The entered value is not a valid floating point.

Action: Enter a valid floating point value.

[aglochan-14]

Message: Wrong time value!

Cause: The entered time value does not correspond with the ADOxx time format (yy:ddd:hh:mm:ss).

Action: Enter a value in ADOxx time format.

[aglochan-15]

Message: The value 0 is not allowed for the division!

Cause: A division by 0 is not possible.

Action: Enter a value for the divisor which is not 0.

[agogo-01]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

The keyword "GRAPHREP" must appear first but the attribute value begins with "<*Text*>"!

Cause: The specified attribute contains errors.

Action: This error can be fixed in the ADOxx Administration Toolkit or please contact your ADOxx administrator.

[agogo-02]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

"<*Text*>" is not a keyword!

Cause: The value assigned to the specified attribute during customising contains errors.

Action: This error can be fixed in the ADOxx Administration Toolkit or please contact your ADOxx administrator.

[agogo-03]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

"<*Text*>" is not the name of an attribute!

Cause: The value assigned to the specified attribute during customising contains errors.

Action: This error can be fixed in the ADOxx Administration Toolkit or please contact your ADOxx administrator.

[agogo-04]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

<*Keyword*> cannot be used together with COMPOUND!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

Only LINE, POLYLINE and CURVE can be used together with COMPOUND.

Action: Correct the error in the Library Management of the ADOxx administration Toolkit or contact your ADOxx administrator.

[agogo-05]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

Illegal numeric parameter in <*keyword*> element!

An integer value > 0 has to be specified.

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

The number of points in the case of POLYLINE / POLYGON must be greater than 0.

Action: Correct the error in the Library Management of the ADOxx administration Toolkit or contact your ADOxx administrator.

[agogo-06]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

<*Keyword*>without IF!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

ELSIF, ELSE or ENDIF used without previous IF.

Action: Correct the error in the Library Management of the ADOxx administration Toolkit or contact your ADOxx administrator.

[agogo-07]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

ENDIF is missing!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

Action: Correct the error in the Library Management of the ADOxx administration Toolkit or contact your ADOxx administrator.

[agogo-08]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

<*keyword*> is just defined for relations!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

EDGE, START, MIDDLE or END in the definition of the graphic representation of a class.

Action: Correct the error in the Library Management of the ADOxx administration Toolkit or contact your ADOxx administrator.

[agogo-10]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

IF with multiple ELSE branches!

Cause: The customised attribute value contains the syntax/semantics error described.

A defined IF sequence contains more than one ELSE branches. While multiple ELSIF statements are possible, only one ELSE condition is allowed.

Action: Fix the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agogo-11]

Message: Error with the value of the attribute "<*attribute name*>" (<*context*>):

<*Keyword*> without block - element sequence enclosed in curly braces is missing!

Cause: The customised attribute value contains the syntax/semantics error described.

FOR or WHILE loops without element sequence in curly braces ("**{ .. }**").

Action: Fix the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agogo-12]

Message: Error with the value of the attribute "<*attribute name*>" (<*context*>):

ENDWHILE without WHILE!

Cause: The customised attribute value contains the syntax/semantics error described.

ENDWHILE was used without preceeding **WHILE** statement.

Action: Fix the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agogo-13]

Message: Error with the value of the attribute "<*attribute name*>" (<*context*>):

"<*Name*>" is not the name of a model type group!

Cause: The customised attribute value contains the syntax/semantics error described.

The model type group name specified in the HOTSPOT definition of the Method Diagram does not comply with any existing name of a model type group.

Action: Fix the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agolay-03]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

"<*text*>" is not a keyword!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agolay-04]

Message: Error in the value of the attribute "<*attribute Name*>" (<*context*>):

Missing layout for "<*keyword*>"!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

HEAD, BITMAP, PAGE or FOOT, without having previously introduced LAYOUT.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agolay-05]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

Region "<*keyword*>" defined twice!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

HEAD, BITMAP, PAGE or FOOT definition twice in LAYOUT.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agolay-06]

Message: Error in the value of the attribute "<*attribute name*>" (<*context*>):

Missing region for "<*keyword*>"!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

TEXT, ATTR or FONT, without having previously introduced a region (HEAD, ...).

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agolay-07]

Message: Error in the value of the attribute "<*attribute name*>" (<*context*>):

Necessary region "PAGE" is missing in layout "<*Name*>"!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

A layout must always contain a PAGE definition.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agolay-08]

Message: Error in the value of the attribute "<*attribute name*>" (<*context*>):

At layout "<*name of layout*>" specified attribute "<*attribute name*>" is not existing or is of the wrong type!

Cause: The customised attribute value contains the syntax/semantics error described.

In the page layout definition <*name of layout*> the GraphRep attribute <*attribute name*> was specified. However, this attribute does not exist in the class "__ModelTypeMetaData__".

Action: Fix the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[agraph-01]

Message: Print-Error: Unable to connect to printer.

Cause: The operating system could not establish a connection to the printer.

Action: Check the connection to the printer or the printer itself. Contact your system administrator if necessary.

[agraph-02]

Message: All values 0: Diagram cannot be displayed.

Cause: All attribute and result values selected for the representation are 0. A diagram can therefore not be represented.

Action: At least one of the attribute or result values to be represented must be greater than 0. Select a set which contains at least one attribute or result value greater than 0 from the sets of results to be represented.

[agraph-03]

Message: Error while copying the graphic to the clipboard: There is not enough available memory.

Cause: There is not enough memory available to copy the graphic to the clipboard.

Action: Exit some other programs and try again.

[agraph-04]

Message: An error has occurred when writing the file "<*file name*>".

The file name is possibly not allowed or the data carrier is full.

Cause: An error has occurred when writing on the specified file. The cause is either there is not enough memory in the target drive or an error when accessing to the file or a faulty target medium.

Action: Verify the file name and make sure that there is enough memory space available in the target drive (you need at least, depending on the output format, from 100KB to 10MB). Make some free space in memory or check if a not write- protected medium is enclosed in the target drive as long as you are dealing with an interchangeable data medium.

[ahaconv-01]

Message: "<*Token ()*>" is not a key word!

Cause: There is a syntactical error in a class attribute "__Conversion__", which is required for the transformation of objects to another class.

Action: Verify and correct the "__Conversion__" attribute.

[ahaconv-02]

Message: "<*Name*>" is not the name of class that can be instanced!

Cause: A not existing class is specified in a class attribute "__Conversion__", which is required for the transformation of objects to another class.

Action: Verify and correct the "__Conversion__" attribute.

[ahaconv-03]

Message: The attribute "<*attribute name*>" (<*class name*>) does not exist!

Cause: A not existing attribute is specified in a class attribute "__Conversion__", which is required for the transformation of objects to another class.

Action: Verify and correct the "__Conversion__" attribute.

[ahaconv-04]

Message: The attribute "<*attribute name 1*>" (<*class name 1*>) and "<*attribute name 2*>" (<*class name 2*>) are not of the same type!

Cause: A conversion indication, which is not realisable, is specified in a class attribute "__Conversion__", which is required for the transformation of objects to another class.

Action: Verify and correct the "__Conversion__" attribute.

[ahaconv-05]

Message: The attributes are not based on the same record class:

"<*Attribute name 1*>" (<*Class name 1*>): "<*Record class 1*>"

"<*Attribute name 2*>" (<*Class name 2*>): "<*Record class 2*>"

Cause: In the class attribute "__Conversion__", used for converting objects from one class to another, a statement has been made which cannot be executed.

Action: Check and correct the "__Conversion__" attribute.

[ahaconv-06]

Message: The attributes are not based on the same attribute profile class:

"<*Attribute name 1*>" (<*Class name 1*>): "<*Attribute profile class 1*>"

"<*Attribute name 2*>" (<*Class name 2*>): "<*Attribute profile class 2*>"

Cause: In the class attribute "__Conversion__", used for converting objects from one class to another, a statement has been made which cannot be executed.

Action: Check and correct the "__Conversion__" attribute.

[ahaconv-07]

Message: The following attribute cannot be converted as not all enumeration values are defined in the target attribute:

From "<*Attribute name 1*>" (<*Class name 1*>) to "<*Attribute name 2*>" (<*Class name 2*>)

Cause: In the class attribute "__Conversion__", used for converting objects from one class to another, a statement has been made which cannot be executed.

Action: Check and correct the "__Conversion__" attribute.

[ahaconv-08]

Message: The following attribute cannot be converted as the name of the target attribute is already assigned to another object:

From "<*Attribute name 1*>" (<*Class name 1*>) to "<*Attribute name 2*>" (<*Class name 2*>)

Cause: In the class attribute "__Conversion__", used for converting objects from one class to another, a statement has been made which cannot be executed. The name used for the instance to be converted cannot be assigned as this name has already been assigned to an object of this class in the model.

Action: Check and correct the "__Conversion__" attribute.

[ahagrob-01]

Message: "<*text*>" is not a keyword!

Cause: In the class attribute "Allowed objects" of a swimlane class an unknown keyword is defined.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahagrob-02]

Message: "<*class name*>" is not the name of a class!

Cause: In the class attribute "Allowed objects" of a swimlane class the name of a undefined class is used with the INCL or EXCL keyword.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahagrob-03]

Message: Objects of class "<*class name*>" may be placed over multiple swimlanes.

Therefore class "<*class name*>" cannot be excluded from swimlane class "<*class name of swimlane*>"!

Cause: In the class attribute "Allowed objects" of the specified swimlane class the specified class is excluded although this class can be placed above several swimlanes. Because of the invalid combination (exclusion in the swimlane class - class allowed to be placed above several swimlanes) Klasse in mehreren Schwimmbahnen erlaubt) the exclusion is ignored.

Action: Correct the class attribute "Allowed objects" in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahamot-01]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

"<*text*>" is not a keyword!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahamot-02]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

Missing model type for "<*keyword*>"!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

INCL, **EXCL**, **OR_ASSIGN**, **AND_ASSIGN** or **MODE**, without having previously introduced model type.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahamot-03]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

Name for "<*keyword*>" is missing!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

Name missing after **model type**, **MODE**, **OR_ASSIGN** or **AND_ASSIGN**.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahamot-04]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

Base model type "<*name*>" is not defined!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

The model type (name after **from:**, **OR_ASSIGN** or **AND_ASSIGN**) was not defined.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahamot-05]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

Base mode "<*name*>" is not defined!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

The mode (name after **from:**, **OR_ASSIGN** or **AND_ASSIGN**) was not defined.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahamot-06]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

"<*name*>" is not the name of a class or relation class that can be instanced!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

The name after **INCL** or **EXCL** is not a class or relation name.

Common mistake: Trying to include or exclude a class which does not belong to the model type at all, while a mode is defined.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahamot-08]

Message: The file "<*file name*>" could not be found!

Cause: The Bitmap file assigned to a model type does not exist.

Action: Create the appropriate file or correct the file name.

[ahanote-01]

Message: Error in the value of attribute "<*attribute name*>" (<*context*>):

"<*text*>" is not a keyword!

Cause: The value defined during customising of the attribute listed contains the syntax error specified.

Action: Correct the error in the Library Management of the ADOxx Administration Toolkit or contact your ADOxx administrator.

[ahanote-02]

Message: "<*attribute name*>" is not the name of an attribute of this class!

Cause: The specified attribute is contained in the class attribute "AttrRep", but it is not defined for this class.

Action: Enter in the class attribute "AttrRep" only the attribute, which is also defined in this class.

[ahanote-04]

Message: The attribute "<*attribute name*>" is contained several times!

Cause: In the class attribute "AttrRep" for the assembly of the ADOxx notebook, the mentioned attribute is defined more than once. In the ADOxx notebook, an attribute cannot be displayed more than once.

Action: Delete in the class attribute "AttrRep" the concerned ATTR elements, so that the specified attribute appears only once.

[ahanote-05]

Message: The user "<*User name*>" has no access rights for the notebook!

Cause: You have not sufficient rights to access the ADOxx notebook of the object/relation.

Action: Contact your ADOxx administrator.

[aicondef-01]

Message: Cannot load icon configuration!

Cause: Possibly the connection to the ADOxx database server has been aborted.

Action: Repeat the action. Should the error re-occur, please contact your database or system administrator.

[aicondef-02]

Message: Cannot save icon configuration!

Cause: Possibly the connection to the ADOxx database server has been aborted.

Action: Repeat the action. Should the error re-occur, please contact your database or system administrator.

[aicondef-03]

Message: Unknown user!

Cause: Possibly the connection to the ADOxx database server has been aborted.

Action: Repeat the action. Should the error re-occur, please contact your database or system administrator.

[aicondef-04]

Message: Cannot load quick-access bar configuration from application library!

Cause: Possibly the application library is defective.

Action: Repeat the action. Should the error re-occur, please contact your ADOxx administrator.

[aimport-01]

Message: Unable to open import file.

Cause: The import file could not be opened; the data medium (e.g. floppy disk) could not be accessed.

Action: Check the medium and re-start the import.

[aimport-02]

Message: Import of file <*path+file name*>:

error in line <#:

<*semantic/syntax error*>

The import process will be aborted.

Cause: During the import, the specified file contains a semantically or syntactically incorrect definition. This might be due to a default in the specified file or to manual changes within the file.

Action: The content of the file cannot be imported. If necessary contact your ADOxx consultant.

[aimport-03]

Message: Import of file <*path+file name*>: contains libraries which are not allowed.

Cause: The ADL-file contains library definitions, which are not allowed.

Action: The models included in the file cannot be imported.

[aimport-04]

Message: A serious error occurred while deleting the erroneous library. Possibly the connection to the database has been aborted. Please exit ADOxx.

Cause: An error occurred during deletion of an erroneous library.

Action: Exit ADOxx and inform your database administrator.

[aimport-05]

Message: A serious error occurred while deleting the erroneous model. Possibly the connection to the database has been aborted. Please exit ADOxx.

Cause: A serious error occurred during the deletion of an erroneous model.

Action: Exit ADOxx, re-start it and redo the import. If the error re-occurs, please contact your database administrator.

[aimport-06]

Message: A serious error occurred while loading the old model. Possibly the connection to the database has been aborted. Please exit ADOxx.

Cause: After an error occurred during the model import (option "Paste into existing models"), the attempt to reload the old model led to a serious error.

Action: Exit ADOxx, re-start it and redo the import. If the error re-occurs, please contact your database administrator.

[aimport-07]

Message: A serious error occurred while restoring the old model. Possibly the connection to the database has been aborted. Please exit ADOxx.

Cause: After an error occurred during the model import (option "Overwrite existing models"), the attempt to restore the model to be overwritten led to a serious error.

Action: Exit ADOxx, re-start it and redo the import. If the error re-occurs, please contact your database administrator.

[aimport-08]

Message: Import of file<*file name*>: File contains models which are not allowed.

Cause: The ABL file contains model definitions, which are not allowed.

Action: The libraries included in the file cannot be imported.

[aimport-09]

Message: Import of file <*file name*>: invalid format.

Cause: You tried to import an ADOxx library or an ADL file which is not compatible with your ADOxx version. ADOxx libraries are not directly exchangeable within different ADOxx versions and/or different operating systems.

Action: Please contact your ADOxx consultant.

[aimport-10]

Message: Import of file <*file name*>:

The read-only model <*model name*> cannot be changed by import.

Cause: You tried to change a read-only model via model import (option "Overwrite existing models" or "Paste into existing models????").

Action: If the model was loaded read-only, please close it and import the model again. If you have read-only access for the model, please contact your ADOxx administrator.

[aimport-11]

Message: No application libraries are defined in the database. Therefore user import is impossible.

Cause: You tried to import users, without previously importing an application library.

Action: Import the application library needed and re-start the user import.

[aimport-12]

Message: Error when opening the protocol file!

Cause: An error has occurred when opening the protocol file. The data medium (e.g. floppy disk) is possibly write-protected.

Action: Specify another protocol file name and/or control the write-protection of the data medium.

[aimport-13]

Message: Error when writing the protocol file "<*file name*>"!

Cause: When creating the import protocol file an error has occurred.

Action: The data medium is possibly full.

[aimport-14]

Message: The model group "<*model group name*>" does not exist!

Would you like to skip the setting of the corresponding access right and continue the import?

Cause: The model group referenced for the UDL import does not exist, i.e. in the UDL file, the access right defined to the specified model group cannot be set.

Action: Click on the "Yes" button to skip the setting of the access rights of the specified model group and continue the import.

Click on the "All" button to skip the setting of the access rights of all non existing model groups and continue the UDL import.

If you click on the "No" button, the UDL import will be aborted.

[aimport-15]

Message: The libraries contain no classes! An import is therefore not possible.

Cause: The libraries of the imported ABL file contain no classes.

Action: Select another ABL file.

[aimport-16]

Message: Import of the file<*file name*>:

The user has no access to the model "<*model name*>"!

This cannot be changed by the import.

Cause: You have attempted to change a model you can not access by ADL import into the data base. This is not permitted.

Action: Select another import option (e.g. rename) or do not import the specified model.

[aimport-17]

Message: Unknown target library!

The connection to the database is possibly aborted.

Exit ADOxx.

Cause: When searching for information in the library, into which the ADL file should be imported, a serious error has occurred.

Action: Exit ADOxx, restart it and repeat the import. Should the message re-appear, contact your database administrator.

[aimport-18]

Message: The ADL file is possibly from a versioned library.

The import of such a file to a non-versioned library is only possible by explicitly specifying the option 'import-versioned-file'.

Cause: You have tried to import an ADL file from a versioned library to a non versioned library. In the silent mode of the ADL import via an AdoScript, this is only possible by explicitly specifying the option 'import-versioned-file'.

Action: Insert the option 'import-versioned-file' to the AdoScript.

[aimport-19]

Message: The ADL file is possibly from a versioned library.

The import of such a file with the option 'Transfer version number to the model name' is only possible by explicitly specifying the option 'import-versioned-file'.

Cause: When searching for information of the library to which the ADL file should be imported, a serious error has occurred.

Action: Insert the additional option 'import-versioned-file' to the AdoScript.

[aimport-20]

Message: Invalid target model group!

Cause: A target model group must be specified for the ADL import via an AdoScript in silent mode, when models or model groups should be imported. You have specified either no or invalid target model groups.

Action: Specify a valid target model group.

[aimport-22]

Message: Invalid target attribute profile group!

Cause: A target attribute profile group must be specified for the ADL import via an AdoScript in silent mode, when attribute profile groups should be imported. You have specified either no or an invalid target attribute profile group.

Action: Enter a valid target attribute profile group.

[aimport-24]

Message: A serious error has occurred when re-creating the old attribute profile.

The connection with the database is possibly aborted.

Exit ADOxx.

Cause: After an error in the attribute profile import (option "Overwrite attribute profiles") you have tried to re-create the overwritten attribute profile, which has led to a serious error.

Action: Exit ADOxx, restart it and repeat the import. If the message appears again, inform your database administrator.

[aimport-25]

Message: A serious error has occurred when deleting the faulty attribute profile.

The connection with the database is possibly aborted.

Exit ADOxx.

Cause: After an error in the attribute profile import, you have tried to delete the faulty attribute profile, which has led to a serious error.

Action: Exit ADOxx, restart it and repeat the import. If the message appears again, inform your database administrator.

[aimport-26]

Message: Import of the file <*ABL file name*>:

Error when saving the file belonging to the library "<*file name*>"!

Cause: A serious error has occurred during the library import when saving the specified file to the ADOxx database.

Action: Exit ADOxx, restart it and repeat the import. Note that the length of the external file name within the library (including the file extension) and the length of the library name must not exceed 249 characters in total.

If the message appears again, inform your database administrator.

[aimport-27]

Message: System user administration could not be initialised!

Cause: You have been trying to import system users into an ADOxx database with Single-Sign-on functionality, but the operating system (Windows ME/98/95) does not support Single-Sign-on.

Action: To import system users into a ADOxx database use an operating system supporting Single-Sign-on (i.e. Windows XP/2000/NT 4.0).

Please pay attention to the software pre-requisites for Single-Sign-on within the ADOxx installation manual.

[aleo-01]

Message: Character expected after backslash '\'!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-02]

Message: Closing quotation mark " is missing!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-03]

Message: Invalid character: <*character*>.

key word (next element) or Name (next attribute) expected!

<*Context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-04]

Message: <*figure*> ':' expected!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-05]

Message: Value expected after ':' !

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-06]

Message: <*Text*>: Measure unknown!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-08]

Message: <*Text*>: key word expected!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-09]

Message: Unauthorised figure after closing quotation marks ""!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described

Action: Correct the error described or contact your ADOxx administrator.

[aleo-10]

Message: Unauthorised figure after a bracket closing an expression ')'!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described

Action: Correct the error described or contact your ADOxx administrator.

[aleo-13]

Message: Operand expected!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described

Action: Correct the error described or contact your ADOxx administrator.

[aleo-14]

Message: Closing bracket ')' is missing!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described

Action: Correct the error described or contact your ADOxx administrator.

[aleo-15]

Message: Operator expected!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described

Action: Correct the error described or contact your ADOxx administrator.

[aleo-16]

Message: Closing bracket ')' without opening bracket '('!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described

Action: Correct the error described or contact your ADOxx administrator.

[aleo-17]

Message: Unauthorised figure!

<*context*>

Cause: The attribute listed (context) was changed during customising (or in the ADL file) and contains the syntax error described

Action: Correct the error described or contact your ADOxx administrator.

[aleo-18]

Message: Division by 0!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-19]

Message: Over flow!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-20]

Message: This expression shall not contain a variable!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-21]

Message: Type incompatibility!

<*function call*>

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-22]

Message: <*Text*>: Unknown function!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-23]

Message: Wrong parameter number!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-24]

Message: <*Text*>: Unknown operator!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-25]

Message: <*Text*>: Lowercase letters in a key word!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-26]

Message: Under flow when accessing to string!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-27]

Message: Over flow when accessing to string!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-28]

Message: <*figure*>: unauthorised figure after a key word!

<*context*>

Cause: The attribute listed (context) was changed during customisation (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-29]

Message: Unauthorised figure after an attribute name!

<*context*>

Cause: The attribute listed (context) was changed during customisation (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-30]

Message: Unauthorised figure after an attribute value!

<*context*>

Cause: The attribute listed (context) was changed during customisation (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-31]

Message: The target of a value assignment must be a variable!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-32]

Message: Call of the function error()!

<*context*>

Cause: The described runtime error has occurred during the evaluation of an expression, which is contained in the value of the specified attribute.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-34]

Message: Closing bracket '}' is missing!

<*context*>

Cause: The attribute listed (context) was changed during customisation (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-35]

Message: Closing bracket '}' without opening bracket '{'!

<*context*>

Cause: The attribute listed (context) was changed during customisation (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-36]

Message: The value for <*attribute name*> is missing or is of the wrong type!

Expected: <*Attribute type1*>, found: <*Attribute type2*>.

<*context*>

Cause: The attribute listed (context) was changed during customisation (or in the ADL file) and contains the syntax error described.

Action: Correct the error described or contact your ADOxx administrator.

[aleo-37]

Message: Over flow when concatenating two strings!

The maximum possible length of a string contains 65529 characters.

Cause: The string made from the concatenation is too long to be saved to a string variable.

Action: Change the concerned command or expression, so that the created string will not be too long.

[aleo-38]

Message: Overflow while calculating an expression!

The expression is too complex and cannot be calculated.

Cause: The expression has a nesting depth and/or a number of operands greater than 250.

Action: Simplify the expression or contact your ADOxx administrator.

[aleo-39]

Message: <*Name of variable*>: is not initialised!

<*Name of function*>

Cause: Calling the function <*Name of function*> changes an array (areplace, aappend). However the variable <*Name of variable*> is not defined as an array.

Action: Initialize the variable listed as an array before the function call or contact your ADOxx administrator.

[aleo-40]

Message: Parameter list without function!

Cause: A part of the expression is interpreted as a parameter list without a function name in front of it.

Action: Correct the error described or contact your ADOxx administrator.

[alib2sgm-01]

Message: The file could not be opened.

Cause: The file specified for the storage of browser contents could not be opened.

Action: The data carrier may be write protected. Delete the write protection and try to save the browser contents again. If necessary contact your system administrator.

[alib2sgm-02]

Message: No file name has been entered!

Cause: You have not specified a file name for the export file of a library.

Action: Enter the name of an export file.

[alib2sgm-03]

Message: The directory name entered is invalid!

Cause: You have entered a not existing path for the export of a library.

Action: Enter a valid (=existing) path for the export.

[alibload-01]

Message: The library "<*library name*>"could not be loaded. Possibly the connection to the database was lost.

Cause: The library listed could not be loaded from the database.

Action: Repeat the action. Should the error re-occur, close/quit ADOxx, re-start it and repeat the action again. Should the error still continue to occur, please contact your system administrator.

[alibload-02]

Message: Initialisation of the application library "<*library name*>" has failed. Please check the attributes of all libraries which are attached to this application library.

Cause: The library attributes of a library which is part of the application library listed contains errors.

Action: Please check the library attributes of all libraries which are part of the application library listed. You can execute this check using the Administration Toolkit (component "Library Management"). Correct all errors which are reported.

Message: Initialisation of the application library "<*library name*>" has failed:

<*Number of error message*>

<*Error description*>

Cause: The attributes of the library contain errors.

Action: Please check the attributes of the libraries shown. Within the message window a further error code (*Number of an error message*) will be shown specifying the type of error (*Error description*). To deal with this error please refer to the error groups ahamot, asimmap or aleo.

If necessary contact your ADOxx consultant.

Message: Initialisation of the application library "<*library name*>" has failed due to the following error:

Error in relation redefinition.

Cause: The attributes in the relation redefinition of the library contain errors.

Action: Check the attributes of the library.

If necessary contact your ADOxx consultant.

Message: Initialisation of the application library "<*library name*>" has failed due to the following error:

Unexpected error (Internal error code: <*Error code*>).

Cause: The attributes of the library contain errors.

Action: Note the error and contact your ADOxx consultant.

[alibload-03]

Message: The library "<*library name*>" is currently being edited by the user "<*user name*>" and therefore cannot be loaded.

Cause: Another ADOxx administrator is editing the mentioned library at the same time.

Action: Wait until the library is released or speak with the other ADOxx administrator about it.

(You can also send an appropriate message to the specified user via the ADOxx mail (menu "Extras", menu item "Messages").)

[alogin-05]

Message: Data base "<*database name*>" does not exist!

Cause: A database of the name you entered has not been catalogued on your computer.

Action: Check the database name entered or apply to your database administrator to have the respective database catalogued.

[alogin-06]

Message: ADOxx could not connect to the database! Please try again.

Cause: The connection to the ADOxx database server may have been lost or the server may have broken down.

Action: Repeat the action. Should the error occur again, please contact your system administrator.

[alogin-09]

Message: The user <*user name*> is already logged into ADOxx! The login process was stopped.

Cause: A user may not be logged in more than once. A user of the user name you entered is already logged in.

Action: Login using a different user name or wait until the user logged in logs out of ADOxx.

[alogin-10]

Message: Login impossible - a user name must be specified!

Cause: You did not enter a user name into the field 'Login'.

Action: Please enter a user name.

[alogin-11]

Message: Login impossible - a password must be specified!

Cause: You did not enter a password into the field 'Password'.

Action: Please enter the user's password.

[alogin-12]

Message: Login impossible - a database name must be specified!

Cause: You did not enter a database name into the field 'database Name'.

Action: Please enter a database name.

[alogin-13]

Message: Wrong password specified for user < *user name*> of database <*database name*>!

Cause: The password you entered is not the correct one for the user and database listed.

Action: Please enter the correct password.

[alogin-14]

Message: In the database <*database name*>, there is no user <*user name*>!

Cause: The user name you entered is not part of the list of possible users for the database listed.

Action: Please enter a user name that is permitted for the database listed or ask the ADOxx administrator to enter the user name you used before into the user list of the ADOxx database.

[alogin-15]

Message: The user <*user name*> has no execution access for the Modelling Toolkit!

Cause: The user of the user name you entered has no execution access for the Modelling Toolkit.

Action: Enter the user name of an ADOxx user who has execution access or ask your ADOxx administrator to grant you execution access.

[alogin-16]

Message: The user <*user name*> has no execution access for the Administration Toolkit!

Cause: The user of the user name you entered has no execution access for the Administration Toolkit.

Action: Enter the user name of an ADOxx user who has execution access or ask your ADOxx administrator to grant you execution access.

[alogin-17]

Message: The database system is not valid!

Cause: The database system you entered does not exist.

Action: Enter one of the database systems permitted (parameter -sDB2, -sORACLE, -sINFORMIX or -sSQLSERVER).

[alogin-18]

Message: ADOxx could not connect to the database - task stopped! Please contact your system administrator.

Cause: The connection to the ADOxx database server may have been lost or the ADOxx database server may have broken down.

Action: Repeat the action. Should the message occur again, please contact your system administrator.

[alogin-19]

Message: At the moment the maximum number of <*number*> users is logged in to the ADOxx database <*data base name*>. Do you want to log into another ADOxx database?

Cause: You tried to log into an ADOxx database which had the maximum number of users permitted already logged in.

Action: Click on the "Yes" button to log into another ADOxx database or the "No" button to cancel.

[alogin-20]

Message: The ADOxx database <*database name*> is configured for single user mode and a user is already logged in at the moment! Do you want to log into another ADOxx database?

Cause: You tried to log into an ADOxx database which is configured for single user mode (maximum number of users permitted: 1).

Action: Click on the "Yes" button to log into another ADOxx database or the "No" button to cancel the ADOxx message.

[alogin-21]

Message: The selected database cannot be accessed from this application!

The database <*database name*> was created with the version <*version number1*> You currently use <*version number2*>.

Note: You may migrate your models using the import/export tool.

Do you want to login to another ADOxx database?

Cause: You tried to log into a database which was not generated by your current ADOxx version (*version number2*).

Action: Generate a new database using your current ADOxx version. Then migrate the models and libraries from the database with the help of the ADOxx version listed (*version number1*). If necessary contact your ADOxx administrator.

[alogin-22]

Message: A character set error has been detected with the selected database!

Please contact your ADOxx administrator.

Do you want to login to another ADOxx database?

Cause: The database system and your ADOxx client use different character sets.

Note: If the database system is "Microsoft SQL Server", then there are probably different ODBC settings regarding the option "Transform OEM into ANSI" on the various clients.

Action: Contact your ADOxx administrator.

Hint: When the database system "Microsoft SQL Server" is used, the option "Transform OEM into ANSI" (program "ODBC32" in the Windows system directory "Settings") must be set to the same value for all clients with ADOxx applications. This value must also agree with the one on that client from which the ADOxx database was installed.

[amoddiff-02]

Message: No further information available!

Cause: You have selected a cell in the table in which there is no further information.

Action: Select the attribute you would like to compare in the column 'In both models'.

[amoddiff-08]

Message: The model <*model name* >could not be loaded. Please refresh the models list and try again.

Cause: An error occurred while attempting to load the specified model. It is possible that this model was deleted by another ADOxx user.

Action: Refresh the model list and select a different model.

[amoddiff-09]

Message: An error occurred while evaluating attributes.

Cause: An error occurred while evaluating attributes.

Action: Try again or select a different object whose attributes you would like to compare.

[amoddiff-10]

Message: The models to be compared must have the same model type.

Cause: You have selected two models for comparison of different model types. Only models of the same type can be compared.

Action: Select two models of the same type.

[amodlru-01]

Message: Model <*model name*>, Instance <instance name>
The reference model does not exist.
The process is continued.
Should future messages of this type be displayed?

Cause: The specified instance refers to a model which does not exist in the database.

Action: Check whether the model exists and correct/recreate the model reference if necessary.

[amodlru-02]

Message: Internal error<*error number*>

Cause: An unexpected error has occurred.

Action: Quit ADOxx, restart it and try this again. If the error re-occurs, contact your ADOxx consultant.

[amodlru-03]

Message: Model <*Model name*>, Instance <*Instance name*>
The referenced model could not be loaded.
The process is continued.
Should future messages of this type be displayed?

Cause: An unexpected error occurred.

Action: Exit ADOxx, restart it and repeat the task. Should the error occur again please contact your ADOxx consultant.

[amodlru-04]

Message: Model <*Model name*>, Instance <*Instance name*>
The referenced model could not be loaded as you do not have sufficient access rights.
The process is continued.
Should future messages of this type be displayed?

Cause: The instance points to a model in a model group in which you do not have the required access rights(read access).

Action: Contact your ADOxx administrator in order to obtain sufficient access rights.

[amodraw-02]

Message: The input "<*input text*>" is invalid for the attribute "<*attribute name*>"!
<*Information about valid inputs*>

Cause: The input text cannot be converted into a value valid for this attribute.

Action: Correct your input accordingly. Consider the given information while doing this.

[amodraw-03]

Message: Clipboard error: Not enough free memory!

Cause: The selected objects and connectors could not be copied onto the clipboard as there is not enough free memory.

Action: Attempt to increase the space available by closing applications and model windows. Then try to copy to the clipboard again.

[amodraw-04]

Message: Error on creating a new relation:
A relation of type <*relation type*>
from object "<*object name*>" (<*class name*>)
to object "<*object name*>" (<*class name*>)
already exists !

Cause: You have attempted to link two objects by a connector when they are already connected by a relation of the same type.

Action: Only one connector of the given type can exist between two objects.

[amodraw-05]

Message: Error on creating a new relation:
A relation of type "<*relation type*>"
from object "<*object name*>" (<*class name*>)

to object "<*object name*>" (<*class name*>)

is not allowed!

< *Information on what relations are possible between the two selected objects and which objects can be linked by the selected relation is also provided here.* >

Cause: You are attempting to create a relation between two objects which cannot be linked by the relation chosen.

Action: If another relation can be used to link the two objects, then join the objects using this relation instead.

[amodraw-08]

Message: Error while writing file "<*filename*>"!

The filename may be invalid or the disk is full or write protected or currently being used by another application.

Cause: The file cannot be created because the filename you have entered is invalid or there is not enough space to create the file or the disk is write-protected or or currently being used by another application.

Action: Check the filename selected and the space available. Close the file in the other application or close the application blocking the access to the file

[amodraw-09]

Message: Error in the value of attribute "<*attribute name*>" (<*model name*>):

"<*Text*>" is not a keyword!

Cause: The value of the specified model attribute was possibly altered in the ADL file and now contains an invalid value.

Action: Correct the ADL file or contact your ADOxx consultant.

[amodraw-10]

Message: No value specified in attribute "<*attribute name*>"!

Cause: Ctrl and double-click cannot be used to bring the referenced model to the foreground in this case as no value was specified in the attribute.

Action: Close the error message and then select that model to which you would like to link this reference attribute.

[amodraw-11]

Message: Error while following an inter-reference!

"<*model name*>" (<*model type*>)

could not be opened. Probably the referenced model has been deleted or renamed and the inter-references need to be updated.

Cause: You have attempted to follow a model inter-reference where the model referenced no longer exists

1. the model was altered by another ADOxx user in the meantime or
2. the model was deleted by another ADOxx user.

Action: 1. Refresh the inter-model references (Menu "Model" - entry "Inter-references" - select incoming or outgoing - and then "Refresh").
2. Recreate the deleted model or delete the reference.

[amodraw-12]

Message: Error while following an inter-reference!

The referenced object

"<*object name*>" (<*class name*>

in model

"<*model name*>" (<*model type*>)

could not be found!.

The referenced model has probably been deleted or renamed and the inter-references need to be updated.

Cause: You have attempted to follow a model inter-reference where the model referenced no longer exists. It has probably been altered or deleted by another ADOxx user.

Action: Recreate the model or delete the reference.

[amodraw-13]

Message: No model pointer is specified for class "<*class name*>"!

Cause: It is not possible to link to the referenced model through ctrl and double-click as the class attribute "Model pointer" must first be updated to hold the correct model description.

Action: Define the model pointer in the ADOxx Administration Toolkit or contact your ADOxx administrator.

[amodraw-14]

Message: The name "<*object name*>" is already assigned to another object of class "<*class name*>"!

Cause: You have attempted to give a name to an object which is the same as the name already given to an object of the same class in the same model.

Action: Enter a name which is unique within the model for this object.

[amodraw-15]

Message: Error while creating graphics: Not enough free memory!

Cause: For large graphics (greater than 1000x1000 pixel) the space available may not sufficient especially when the colour settings are very high on the local computer (e.g. 16 mill. colours).

Action: Set the colour resolution to 256 colours or produce the graphic with fewer colours or smaller proportions.

[amodraw-16]

Message: Error when following the model pointer:

"<*model name*>" (<*model type*>)

does not exist!

The error can only occur when the attribute specified in the model pointer is of the type STRING. We recommend to switch to INTERREF .

Cause: The specified model has since been deleted or there is a spelling mistake.

Action: Re-create the model and correct the model name accordingly.

[amodraw-17]

Message: None of the copied objects (initial model type: "<*model type1*>") can be pasted into the models of type "<*model type2*>"!

Cause: The models which you want to copy from one model cannot be pasted into a model of another model type. Either the class of the models is not defined in the target model or the target model is not based on the same library (BP or WE library) as the starting model.

Action: It is not possible to copy these objects.

[amodraw-18]

Message: Error when reconnecting a relation:

A relation of the type <*relation type*>

of object "<*object name*>" (<*class name*>)

to object "<*object name*>" (<*class name*>)

already exists!

Cause: You have attempted to put a relation between two objects, for which there is already an appropriate connector.

Action: Only a connector of the same type is possible between two objects.

[amodraw-19]

Message: Error when reconnecting a relation:

A relation of the type <*relation type*>

of object "<*object name*>" (<*class name*>)

to object "<*object name*>" (<*class name*>)

is not authorised!

<*Information concerning which relation is possible between the selected objects and between which classes the selected relation is authorised.*>

Cause: The created connector is missing since the underlying relation is, according to its definition, not possible between the classes of the selected objects.

Action: If there is another relation authorised between the objects, you can use this relation to connect them with each other.

[amodraw-20]

Message: Error when following an inter model reference!

"<*model name*>" (<*model type*>)

could not be opened because of an inappropriate version number.

The referenced model may have been deleted in the meanwhile or the inter model references must only be updated once the referenced model has been updated.

Cause: You have attempted to follow an inter model reference, but the referenced model no longer exists as a result of it possibly being deleted in the meantime by another ADOxx user.

Action: Update the inter model references (menu "Edit" - entry "inter model references" - button "Update") and repeat the action.

Should the error re-occur, restore the deleted model and delete the reference.

[amodraw-21]

Message: The object cannot be renamed as it is referenced by at least one model you do not have access to.

Cause: You have tried to rename an object which is referenced by another model you do not have access to.

Action: Update the model list and try again. Should the error re-occur then contact your ADOxx administrator in order to gain access to the referenced models. A list of all models referencing the target object can be obtained using the dialogue "Show incoming references".

[amodraw-22]

Message: The input "*Connector number*" is invalid for the connector mark!

Enter a positive numeric value.

Cause: You have tried to rename a connector mark and assigned an invalid value.

Action: Enter an integer value greater than zero in order to rename the connector mark or click on the button Cancel to exit the renaming.

[amodraw-23]

Message: The *Connector number* provided is not valid for the connector mark! Please provide an index consisting of no more than three characters using characters A-Z.

Cause: You have tried to rename a connector mark and assigned an invalid value.

Action: Enter a valid name for the connector mark: valid values are strings consisting of characters from A to Z and having no more than three characters(A, AB, ABC).

[amodtab-01]

Message: The input "<*input text*>" is not valid for the attribute "<*attribute name*>"

<*additional error information*>

Would you like to work further on this attribute?

Cause: An invalid value was added via the modelling tabular view.

Action: Enter a valid value.

[amodtab-02]

Message: The name "<*object name*>" is already assigned to another object of class "<*class name*>"!

<*additional error text*>

Would you like to continue working on this attribute?

Cause: When changing an objects name in the tabular modelling view, a name was given to one object which had already been assigned to another object of the same class.

Action: Select a different name for the object.

[amodtab-03]

Message: Not all values could be pasted!

Cause: When pasting to the ADOxx browser, an error has occurred. Therefore not all values could be inserted.

Action: Check if the area to be pasted is not too large and if the attribute types go together. Then check if the pasting is allowed for the area you have selected. You should also pay attention to the uniqueness of the object names during the pasting.

[amodtab-04]

Message: No values could be pasted!

Cause: When pasting to the ADOxx browser, an error has occurred. Therefore no values could be inserted.

Action: Check if the area to be pasted is not too large and if the attribute types go together. Then check if the pasting is allowed for the area you have selected. You should also pay attention to the uniqueness of the object names during the pasting.

[amomain-01]

Message: Error while unlocking "<*model name*>" (<*model type*>)!

Possibly the connection to the database was aborted.

Cause: Possibly the connection to the ADOxx database server has been lost.

Action: Contact your system administrator.

[amomain-02]

Message: Error while renaming the "<*model name*>" (<*model type*>):

"<*model name*>" (<*model type*>) already exists!

Cause: When renaming an object you have attempted to give it a name which already belongs to another object of the same type in the model.

Action: Enter a different name for the model.

[amomain-03]

Message: Error while saving "<*model name*>" (<*model type*>) as "<*model name*>" (<*model type*>):

A model with the desired name already exists!

If you cannot see the model with this name, then either it does not belong to a model group or it belongs to a model group to which you have no access. Contact your ADOxx administrator.

Cause: You have attempted to save a model under a name already held by another model.

Action: Save the model under a different name.

[amomain-05]

Message: "<*model name*>" (<*model type*>) cannot be deleted because it is currently locked by user <*User name*>.

Cause: The specified model is currently being edited by another ADOxx user and so is locked.

Action: You can delete this model when it is no longer locked by any ADOxx users.

[amomain-06]

Message: "<*model name*>" (<*model type*>) could not be opened!

Internal error code: <*Code*>

Cause: A connection could not be established to the ADOxx database.

Action: Note the error code and contact your database administrator.

[amomain-07]

Message: "<*model name*>" (<*model type*>) could not be saved in the database!

Cause: Either there is not enough space in the ADOxx database to save this model or no connection could be established to the database.

Action: Contact your database administrator.

[amomain-08]

Message: "<*model name*>" (<*model type*>) contains an illegal process reference!

Possible causes for an illegal process reference:

Either no model with the specified name exists or a future model version is referenced.

or

Message: "<*model name*>" (<*model type*>) contains <*number*> illegal process references!

Possible causes for an illegal process reference:

Either no model with the specified name exists or a future model version is referenced.

Cause: You have attempted to open any subprocesses contained in the model automatically. However there are one or more Subprocess objects which reference non-existent models.

Action: Correct these references or create the appropriate models.

[amomain-10]

Message: An error has occurred while deleting from the database!

Cause: No connection to the ADOxx database could be established.

Action: Contact your system administrator.

[amomain-11]

Message: It is not possible to close model windows while the simulation is running!

Cause: You have attempted to close a model window while the simulation is still running.

Action: Close the window once the simulation has been completed.

[amomain-12]

Message: "<*model name*>" (<*model type*>) has the access state "read access" and so cannot be deleted.

Cause: You have attempted to delete a model for which you have no update rights.

Action: Contact your ADOxx administrator.

[amomain-16]

Message: Error while creating a new model:

"<*model name*>" (<*model type*>) exists already

If this model does not appear in the model select box, it is not contained in an accessible model group. In this case, please contact your ADOxx administrator.

Cause: You have entered a name for this new model which is already assigned to another model.

Action: Enter a different name.

[amomain-17]

Message: Error while creating a new model!

Internal error code: <*code*>

Cause: No connection to the ADOxx database could be established.

Action: Please contact your system administrator.

[amomain-20]

Message: "<*model name*>" (<*model type*>) has the access state "no access" and therefore cannot be opened!

Cause: Possibly the access status of this model was changed or it was created by a user from a different user group with more access than you.

Action: Contact your ADOxx administrator in order to change your access rights.

[amomain-21]

Message: Error while opening a submodel of "<*model name*>" (<*model type*>)! The submodel has the access state "no access" and therefore cannot be opened!

Cause: Possibly the access status of this submodel was changed or it was created by a user from a different user group with more access than you.

Action: Contact your ADOxx administrator in order to change your access rights.

[amomain-22]

Message: "<*model name*>" (<*model type*>) could not be opened - this model has been deleted or renamed! The models which currently exist in the database are shown after clicking the "Refresh" button.

Cause: The selected model was renamed or deleted previously.

Action: Click on the "Refresh" button in order to refresh the list of model names.

[amomain-25]

Message: The new model could not be created!

The new model cannot be assigned to any of the selected model groups because the user rights have been changed in such a way that no write access is possible on these model groups any more or the selected model groups have been deleted by another user in the meantime!

Cause: You have attempted to create a new model but you no longer have access to the chosen model group.

Action: Refresh the model list and save the model to a group which you have access to or create a new model group in which to save the model.

[amomain-26]

Message: "<*model name*>" (<*model type*>) could not be saved to the database, since the user rights have been changed in such a way that the write access to the associated model groups is no longer possible!

Cause: The write rights on the model groups have been withdrawn by another user.

Action: Contact your ADOxx administrator or save the model with a new name to another model group.

[amomain-27]

Message: Error when saving "<*model name 1*>" (<*model type 1*>) as new version "<*model name 2*>" (<*model type 2*>):

The selected new version already exists!

Cause: During the storage of a model as a new version, you have entered a version number, which is already given to the current model.

Action: Select a version number which is not already given.

[amomain-28]

Message: Error when renaming "<*model name 1*>" (<*model type 1*>) in "<*model name 2*>" (<*model type 2*>):

The selected new version already exists!

Cause: When renaming the version number of a model ,you have entered a version number which already exists for the current model.

Action: Select a version number which has not already been assigned.

[amomain-29]

Message: Error when saving the "<*model name 1*>" (<*model type 1*>) as new version "<*model name 2*>" (<*model type 2*>):

No write access rights for the associated model group!

Cause: The model cannot be saved as a new version, since there are no write access rights for the model group.

Action: Contact your ADOxx administrator.

[amomain-30]

Message: The connection to the database is aborted!

Exit ADOxx.

Cause: There is no connection to the ADOxx database.

Action: Export not-saved models to an ADL file, exit ADOxx and then start again. Import the models from the ADL file, to restore the status before the loss of the database connection.

[amomain-31]

Message: "<*Model name*>" (<*Model type*>) can not be saved in the database, because the model has violated at least one of the cardinality conditions!

Cause: The automatic cardinality check has detected an error in the model.

Action: Check the model with help of modelling guidelines and eliminate the error.

[amomain-32]

Message: Error while changing the name from <*Model name1*> (<*Model typ1*>) to <*Model name2*> (<*Model typ2*>):

The new model name is improper!

Cause: The model name contains improper line feed.

Action: Change the model name .

[amomain-33]

Message: Error while saving <*Model name1*> (<*Model typ1*>) under <*Model name2*> (<*Model type2*>):

A new model name is improper!

Cause: The model name contains improper line feed.

Action: Change the model name.

[amomain-34]

Message: Error while saving "<*Model name1*>" (<*Model type1*>) under "<*Model name2*>" (<*Model type2*>):

A new version number is improper!

Cause: The version number contains the improper line feed.

Action: Change the Version number.

[amomain-35]

Message: Error while renaming "<*Model name1*>" (<*Model type1*>) in "<*Model name2*>" (<*Model type2*>):

A new version number is improper!

Cause: The version number contains the improper line feed.

Action: Change the Version number.

[amomain-36]

Message: "<*Model name*>" (<*Model type*>) could not be renamed, because it is currently locked by user <*user name*>!

Cause: For renaming a model it must not be locked by any other ADOxx user.

Action: Rename the model later, after the other user closed the model.

[amomain-37]

Message: "<*Model name*>" (<*Model type*>) could not be opened - this model has been deleted in the mean time!

Cause: Opening the model from the window "Search results" was not successful as the model had been deleted in the meantime.

Action: Update the contents of the window "Search results" - the deleted model will not be listed any more.

[amomain-38]

Message: "<*Model name*>" (<*Model type*>) could not be renamed, because it is opened read-only!

Cause: To rename the model it has to be closed or opened with write-access.

Action: Close the model and rename it afterwards.

[amomain-39]

Message: A new model cannot be created, because the maximum number of 300 models opened at the same time is already reached!

Cause: A new model cannot be created, because the maximum number of 300 models opened at the same time is already reached!

Action: Close at least one of the model windows to open other model windows.

[amomain-40]

Message: "<*Model name*>" (<*Model type*>) cannot be opened, because the maximum number of 300 models opened at the same time is already reached!

Cause: The specified model cannot be opened, because the maximum number of 300 models opened at the same time is already reached!

Action: Close at least one of the model windows to open other model windows.

[amoshell-01]

Message: Error when executing the open command:

No model type has been specified!

Cause: An OPEN_MODEL-command has been sent to the "modeling" message port without having specified the model type.

Action: Correct the call of the OPEN_MODEL-command.

[amoshell-02]

Message: Error when executing the open command:

"<*model type*>" is not a model type!

Cause: An OPEN_MODEL-command with a non existing specified model type has been sent to the "modeling" message port.

Action: Correct the call of the OPEN_MODEL command.

[amoshell-03]

Message: Error when executing the open command:

"<*model name*>" (<*model type*>) does not exist!

Cause: An OPEN_MODEL-command with a non existing specified model has been sent to the "modeling" message port.

Action: Correct the call of the OPEN_MODEL command.

[anamegen-01]

Message: The definition of the name generation is erroneous:

Class "<*class name*>" does not have an attribute "<*attribute name*>"!

Cause: The attribute mentioned is not defined in the shown class and thus cannot be used for the name generation.

Action: When define a name generation, choose an existing attribute.

[anbbrow-01]

Message: Not all values could be inserted!

Cause: When inserting in the ADOxx browser, an error has occurred. The insertion has been aborted.

Action: Check if the area to be pasted is not too large and if the attribute types go together. Then check if the insertion in the area you have selected is allowed.

[anbbrow-02]

Message: The entry "<*entry text*>" is not authorised for the attribute "<*attribute name*>"!
<*Additional error text*>
Do you want to further edit the attribute?

Cause: When editing in a record attribute an invalid attribute value has been entered.

Action: Enter a valid value for this attribute.

[anotebk-01]

Message: An error occurred while attempting to save the contents of the notebook!

Cause: Either the filename under which you are saving is invalid or there is not enough space on the disk or drive to which you are saving.

Action: Check filename and available space.

[anotebk-02]

Message: An error occurred while attempting to print the contents of the notebook!

Cause: Possibly there is an error with your printer configuration.

Action: Contact your system administrator.

[anotebk-03]

Message: Starting the process
<*program name*>
failed!

Cause: The value referenced in the PROGRAMCALL attribute cannot be resolved successfully as an external program.

Action: Possibly the "Path" environment variable is incomplete or the program has not been installed.

[aoffanim-01]

Message: Error while reading the protocol file!

Cause: ADOxx could not open the file you specified.

Action: Check and correct the name and/or path listed of the protocol file to be opened.

[aoffanim-02]

Message: Invalid file format!

Cause: The file opened does not meet the format requirements expected by ADOxx.

Action: Check that you have entered the correct file name. If necessary generate a new protocol file.

[aoffanim-03]

Message: Model not found!

Cause: ADOxx could not open a model, which is referenced in the protocol file to be opened. Perhaps the model was deleted after this file was generated or it belongs to a different database.

Action: Check, whether the model named exists. If not, could re import it or open a different database, which still contains this model.

[aoffanim-04]

Message: Instance not found!

Cause: ADOxx could not find an instance, which is referenced in the protocol file to be opened. Perhaps the instance was deleted from the model or renamed after the record file was generated.

Action: Check whether the instance named still exists in the model. If not you will probably need to re import the original model or generate a new protocol file.

[aoffanim-05]

Message: Class not found!

Cause: ADOxx could not find a class which was referenced in the protocol file to be opened. Perhaps the record file was generated using a model which belongs to a different application library than the current one.

Action: Check whether the record file belongs to the model of a different application library. If not, generate a new record file.

[aoffanim-06]

Message: Syntax error in the protocol file! Incorrect time format!

Cause: A time specified in the protocol file does not meet the expected format requirements. Perhaps the protocol file was changed.

Action: Generate a new protocol file. Should the error continue to occur, please contact your ADOxx consultant.

[aoffanim-07]

Message: Subprocess does not exist!

Cause: An error occurred.

Action: Repeat the action (probably after having restarted ADOxx or after having deleted the simulation cache). Should the error continue to occur, please contact your ADOxx consultant.

[aoffanim-08]

Message: Syntax error in the protocol file! Numeric value expected!

Cause: ADOxx expects a numeric value in a certain position of the protocol file. Perhaps the protocol file was changed.

Action: Generate a new protocol file. Should the error continue to occur again, please contact your ADOxx consultant.

[aoffanim-09]

Message: Syntax error in the protocol file! Line too long!

Cause: A line in the protocol file is longer than allowed. Perhaps the protocol file was changed.

Action: Generate a new protocol file. Should the error continue to occur again, please contact your ADOxx consultant.

[aoffanim-10]

Message: Syntax error in the protocol file! Unknown event!

Cause: An event occurring in the protocol file is not known to ADOxx. Perhaps the protocol file was changed.

Action: Generate a new protocol file. Should the error re-occur, please contact your ADOxx consultant.

[aoffanim-11]

Message: The protocol file has been created with the detailed format.

Cause: The protocol file you specified has the detailed protocol format. The offline animation cannot read this format.

Action: Start another simulation run with protocol switched on and select the option "short".

[aoffanim-12]

Message: Attribute not found!

Cause: ADOxx could not find an attribute, which is referenced in the protocol file to be opened. Perhaps the protocol file was generated using a model that belongs to a different application librrary than the current one.

Action: Check whether the protocol file belongs to the model of a different application library. If not, generate a new protocol file.

[aoffanim-13]

Message: Error when accessing model!

Cause: An error occurred.

Action: Repeat the action (probably after having restarted ADOxx or after having deleted the simulation cache). Should the error continue to occur, please contact your ADOxx consultant.

[aoffanim-14]

Message: The task stack cannot be displayed in the Offline Animation.

Cause: During the Offline Animation you have clicked on a task stack.

Action: No task stack can be displayed during the Offline Animation. To obtain this type of information you must carry out a workload analysis.

[aoutgen-01]

Message: Invalid filename.

Cause: An invalid filename was given to which the browser contents should be saved.

Action: Enter a valid filename.

[aoutgen-02]

Message: The file could not be opened.

Cause: The file specified into which the browser contents should be saved could not be opened.

Action: Possibly the disk is read-only. Remove the access restriction and attempt to save the browser contents again. Contact your system administrator if necessary.

[aoutgen-03]

Message: Error writing to file.

Cause: The file specified into which the browser contents should be saved could not be written.

Action: There is possibly no longer enough space on your disk. Contact your system administrator if necessary.

[aoutgen-04]

Message: Error when starting the postprocessors.

(command line: <*command*>)

Cause: When saving the browser contents in the format RTF or HTML, an error has occurred during the start of the postprocessor (jade).

Action: Verify if the postprocessor is installed in the ADOxx installation directory or contact your ADOxx administrator.

[apercalc-01]

Message: At least one model is not archived. The monthly closure will be aborted.

Cause: To execute a monthly closure, all models concerned must previously have been archived. This is not the case.

Action: Archive all concerned models.

[aperctrl-02]

Message: The evaluation cannot be executed, since the current ADOxx configuration contains no time-related versioning!

Cause: You have attempted to carry out an inter-period evaluation, although your ADOxx configuration does not support this.

Action: Contact your ADOxx administrator.

[aperout-01]

Message: The current result display is empty!

Cause: No results are available for the current settings.

Action: Select other settings.

[aperpar-01]

Message: Error in the dynamic evaluation module "<*module name*>": The key word '<*key word name*>' is invalid or is used at an invalid location.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-02]

Message: Error in the dynamic evaluation module "<*module name*>": After the key word '<*key word name*>', the name of the key figure is missing.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the attribute "dynamic evaluation module" must be corrected.

[aperpar-03]

Message: Error in the dynamic evaluation module "<*module name*>": No formula is defined for the key figure '<*key figurename*>' .

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-04]

Message: Error in the dynamic evaluation module "<*module name*>": The definition of the key figure '<*key figure name*>' is incomplete. The correct definition of the formula is missing behind the key word '<*key word name*>' .

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-05]

Message: Error in the dynamic evaluation module "<*module name*>": The definition of the key figure '<*key figure name*>' contains errors. The expression specified behind the key word '<*key word name*>'

'<*expression*>

is syntactically incorrect.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-06]

Message: Error in the dynamic evaluation module "<*module name*>": The definition of the key figure '<*key figure name*>' contains errors. A result file type has been specified as a formula behind the key word '<*key word name*>'.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-07]

Message: Error in the dynamic evaluation module "<*module name*>": The redefinition of the function '<*function name*>' contains errors. No valid function indicator has been specified behind the key word '<*key word name*>'.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-08]

Message: Error in the dynamic evaluation module "<*module name*>": The following key figures are not defined, although they are referenced in at least one formula:

<*Name of the key figures*>

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-09]

Message: Error in the dynamic evaluation module "<*module name*>": The definition of the key figure '<*key figure name*>' contains errors. The expression '<*value*>' specified behind the key word '<*key word name*>' is invalid.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-10]

Message: Error in the dynamic evaluation module "<*module name*>": The name of the following key figures is not unambiguous:

<*name of the key figures*>

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-11]

Message: No key figures are defined for the dynamic evaluation module "<*module name*>".

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-12]

Message: Error in the dynamic evaluation module "<*module name*>": The expression "<*value*>" is not defined.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-13]

Message: Error in the dynamic evaluation module "<*module name*>" behind the key word "<*key word name*>": "<*value*>" does not exist.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-14]

Message: Error in the dynamic evaluation module "<*module name*>" behind the key word "<*key word name*>": "<*value*>" has the wrong type (expected type: "<*Type name*>").

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-15]

Message: An unknown dynamic evaluation module has been discovered.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperpar-16]

Message: The current configuration contains errors.

No dynamic evaluation module has been defined.

Cause: Your ADOxx application library contains errors. The library attribute "Dynamic evaluation module" contains a value which does not define a correct dynamic evaluation module.

Action: Contact your ADOxx administrator. The value of the library attribute "Dynamic evaluation module" has to be corrected.

[aperress-03]

Message: The calculation of the key figure "<*key figure name*>" is cyclic.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperress-04]

Message: The key figure "<*key figure name*>" could not be calculated for "<*object name*>" . Reason: <*reason*>

Cause: The specified key figure can not be calculated. Your ADOxx Application Library could possibly be defective.

Action: Contact your ADOxx administrator.

[aperress-05]

Message: The key figure "<*key figure name*>" could not be calculated, since the evaluation of the formula '<*formula value*>' is wrong.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[aperress-06]

Message: The key figure "<*key figure name*>" could not be calculated, since the time basis belonging to it could not be found.

Cause: Your ADOxx application library contains errors.

Action: Contact your ADOxx administrator. The value of the library attribute "dynamic evaluation module" must be corrected.

[apersui-01]

Message: The ending time of the evaluation cannot be before the starting time of the evaluation!

Cause: You have specified an ending time, which is before the starting time.

Action: Specify an ending time, which is greater/the same as the starting time.

[apersui-02]

Message: The evaluation cannot start with "<*date*>", since the model "<*model name*>" only exists from "<*date*>" !

Cause: At least one of the models to be evaluated does not exist at the specified starting time of the evaluation. (i.e. its version number is greater than the starting time.)

Action: Select a later starting time or create an earlier model version of the model concerned.

[aplangen-01]

Message: Error in library attribute (wrong number).

Cause: The definition of the selected plan in the application library contains errors.

Action: Contact your ADOxx administrator.

[apredct-01]

Message: Error during the evaluation:
<*error*>!

Cause: The specified error has occurred during the evaluation of the transition condition.

Action: Correct the error.

[apredct-02]

Message: Error during the transfer:
<*error*>!

Cause: The specified error has occurred during the transfer of the transition condition.

Action: Correct the error.

[aprint-01]

Message: Error while trying to start the print job!

Cause: There could possibly be a problem with your printer configuration.

Action: Contact your system administrator.

[aprint-02]

Message: "< *filename* >" is not a valid filename!

Cause: A file with the specified name could not be opened. Either the path does not exist or there is an invalid character in the name.

Action: Correct your input.

[aproobjs-01]

Message: Error in time attribute!
< *activity subclass*>: <*activity name*>
Model: <*model name*>

Cause: A fatal error has occurred.

Action: If this error re-occurs when the simulation is run again, export your models to an ADL file and contact your ADOxx consultant.

[aproobjs-02]

Message: Time attribute expected!

<*activity subclass*>: <*activity name*>

Model: <*model name*>

Cause: A fatal error has occurred.

Action: If this error re-occurs when the simulation is run again, export your models to an ADL file and contact your ADOxx consultant.

[aproobjs-03]

Message: Random generator for Variable of type enumeration expected!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The referenced random generator contains a distribution of type normal, uniform or exponential but is connected to a variable of type enumeration.

Action: Correct the attribute "value" in the Random generator or change the type of the Variable.

[aproobjs-04]

Message: Random generator for a variable of type float expected!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The referenced random generator contains a distribution of type discrete but is connected to a variable of type enumeration.

Action: Correct the attribute "value" in the Random generator or change the type of the Variable.

[aproobjs-05]

Message: Invalid index!

Model: <*model name*>

Cause: A fatal error has occurred when accessing a variable.

Action: If this error re-occurs when the simulation is run again, export your models to an ADL file and contact your ADOxx consultant.

[aproobjs-06]

Message: No subsequent relation found!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: No subsequent connector was found or no transition condition evaluates to true.

Action: Possibly a subsequent connector is missing. In this case create one. If this object is a Decision or a Parallelity, it is also possible that no transition condition evaluates to true. Check the transition conditions and the random generator of the variables used.

[aproobjs-07]

Message: No unique subsequent exists!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: No transition condition in the subsequent relations of the specified objects evaluates to true.

Action: Only objects of the class "Parallelity" may have many subsequent connectors which evaluate to true. Delete any unnecessary connectors or update the transition conditions.

[aproobjs-08]

Message: No subprocess model given!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: No valid subprocess model is referenced in the "Referenced subprocess" attribute of the Subprocess object.

Action: Correct the value of the "Referenced subprocess" attribute in the specified Subprocess object.

[aproobjs-09]

Message: No subsequent exists or no transition condition fulfilled!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The specified object of class "Parallelity" has no subsequent connectors or has no subsequent connectors whose transition conditions evaluate to true.

Action: Possibly a subsequent connector is missing. In this case create one. It is also possible that no transition condition evaluates to true. Check the transition conditions and the random generator of the variables used.

[aproobjs-10]

Message: Merging without Parallelity!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: No corresponding object of class Parallelity can be found for the specified object of class "Merging".

Action: Change the model appropriately.

[aproobjs-11]

Message: Unexpected stack error!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: A fatal error has occurred from which the system cannot recover.

Action: If this error re-occurs when the simulation is run again, export your models to an ADL file and contact your ADOxx consultant.

[aproobjs-12]

Message: Many conditions fulfilled!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: More than one transition condition was evaluated to true for the subsequent relations following the specified object.

Action: Change the transition conditions or the distributions as appropriate.

[aproobjs-13]

Message: No transition conditions are true!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: No transition condition of the specified decision's "Subsequent" connectors produced the result "true" on evaluation.

Action: Change the transition conditions and/or the distributions of the variables used accordingly.

[aproobjs-16]

Message: Performer not found! Model: <*model name*>

Cause: A runtime error occurred while the performers were assigned.

Action: Should the error occur again when the simulation is restarted, export the model into an ADL file and contact your ADOxx consultant.

[aproobjs-17]

Message: Availability must contain a positive value!

<*subclass performer*>: <*name performer*>

Model: <*model name*>

Cause: The attribute "Availability" of the object specified must contain a value greater or equal to zero.

Action: Check the value of the attribute "Availability" of the object specified.

[aproobjs-18]

Message: Unknown subprocess model!

<*object class*>: <*object name*>

Subprocess model: <*name of subprocess model*>

Model: <*model name*>

Cause: The model specified in the attribute "Referenced subprocess" does not exists.

Action: Change the value of the attribute "Referenced subprocess" in the object specified.

[aproobjs-19]

Message: Error following a relation!

<*object class*>: <*object name*>

Relation: <*relation class name*>

Model: <*model name*>

Cause: A runtime error occurred which the system could not correct.

Action: Should the error occur again, when the simulation is restarted, export the model into an ADL-file and contact your ADOxx consultant.

[aproobjs-21]

Message: Syntax error during resource assignment!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The value in the attribute "Selection" of the object specified does not satisfy the conventions of AQL.

Action: Correct the value in the attribute "Selection" of the object specified.

[aproobjs-22]

Message: No resource found! Perhaps an unknown name was used.

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The AQL expression in the attribute "Selection" did not render any resources. Perhaps an unknown name was used.

Action: Check if the names used in the AQL expression are valid. Should the error occur nonetheless, export the model into ADL and inform your ADOxx consultant.

[aproobjs-23]

Message: Resource Assignment.

Model: <*model name*>

Cause: A runtime error occurred while the resources were being assigned.

Action: Should the error occur again when the simulation is restarted export the model into ADL and inform your ADOxx consultant.

[aproobjs-24]

Message: No resource found!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The AQL expression listed did not return any resources.

Action: Check the AQL expression in the attribute "Selection" and ensure that the working environment model concerned contains the necessary resources.

[aproobjs-25]

Message: Dynamic role variable does not exist!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: A runtime error occurred which the system could not correct.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-26]

Message: Syntax error during performer assignment!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The value in the attribute "Performer" of the object specified does not satisfy the conventions of AQL.

Action: Correct the value of the attribute "Performer" in the object specified.

[aproobjs-27]

Message: No performer found! Perhaps an unknown name was used

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The AQL expression in the attribute "Performer" did not render any performer. Perhaps an unknown name was used.

Action: Check if the names used in the AQL expression are valid. Should the error occur nevertheless, export the model into ADL and inform your ADOxx consultant.

[aproobjs-28]

Message: The performer assignment contains errors.

model: <*model name*>

When specifying the performer assignment, an unexpected error has occurred! Clear the simulation memory and re-start the simulation. Should the error re-occur, contact your ADOxx administrator.

Cause: A runtime error has occurred during the performer assignment.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-29]

Message: The assignment contains errors. An unknown variable has possibly been specified.

<*object class*>: <*object name*>

Model: <*model name*>

or

Message: The assignment contains errors. An unknown variable has possibly been specified.

Assignment: <*random generator*>

Model: <*model name*>

Cause: A variable is used but has not been defined.

Action: Check the location in which the variable with the name listed is used.

[aproobjs-30]

Message: The random generator expression contains errors!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: The random generator expression contains an error. The expression is probably syntactically not correct.

Characters, which are added to the random generator, like for instance, **"Discrete (Yes 0.5; No 0.5) values correct? Must still be checked."** are a frequent cause for this error.

In this example the completion "values correct? must still be checked." should have been removed before the simulation. Otherwise the error concerned appears.

Action: Correct the random generator expression. Remove possibly added characters.

[aproobjs-31]

Message: Process has no unique start object!

Business process model: <*model name*>

Cause: A business process model must contain **exactly one** start object.

Action: Delete the superfluous start objects in the business process model specified.

[aproobjs-32]

Message: Error when accessing working environment model: Working Environment Model: <*model name*>

Cause: A runtime error occurred while the working environment model was being accessed.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-33]

Message: Missing start object in instance table!

Cause: A runtime error occurred which the system could not correct.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-34]

Message: Variable table of subprocess not found!

Model: <*model name*>

Cause: A runtime error occurred which the system could not correct.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-35]

Message: Error searching for relation instance!

Model: <*model name*>

Cause: A runtime error occurred which the system could not correct.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-36]

Message: Missing object in instance table!

Model: <*model name*>

Cause: A runtime error occurred which the system could not correct.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-37]

Message: Index not found!

<*object class*>: <*object name*>

Index: <*index*>

Model: <*model name*>

Cause: When a subprocess was called in the model specified, the indices were not consistent.

Action: Check the indices of the "Parameter" and "Call parameter" relation of the subprocess object and the start object in the business process called.

[aproobjs-38]

ATTENTION: The error message [aproobjs-38] contains different text parts which are all shown in the following. Should there be no further explanations, then the transition condition contains no correct elements.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

Missing **')'**

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent"-connector does not meet the syntactic conventions. A closing bracket is missing.

Action: Check whether there is a matching closing bracket for each opening bracket.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

Unknown operator of comparison!

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent" connector does not meet the syntactic conventions. ADOxx expects an operator of comparison in the place indicated. For constants and variables of the type "Floating Point Number" the operators "<", ">" and "=" are permitted.

For constants and variables of the type "Enumeration" the operator of comparison "=" may be used.

Action: Use a permitted operator.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

Unknown variable!

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent"-connector does not meet the syntactic conventions. An undefined variable was used.

Action: Check whether the variable used has already been defined as a global variable in the main process or as a local variable in the respective subprocess model. If so, check the variable's name.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

Illegal type of variable!

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent" connector does not meet the syntactic conventions. An undefined variable was used.

Action: Check whether the variable used has already been defined as a global variable in the main process or as a local variable in the respective subprocess model. If so, check the variable's name.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

No term identifier!

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent" connector does not meet the syntactic conventions. Possibly characters which are not permitted were used for a term. The characters permitted are any numbers, letters and the signs "_", "-", and "+".

Action: Replace the characters which are not permitted.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

Illegal operator of comparison!

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent" connector does not meet the syntactic conventions. An illegal operator of comparison was used. For constants and variables of the type "Floating Point Number" the operators "<", ">" and "=" are permitted. For constants and variables of the type "Sequence" the operator of comparison "=" may be used.

Action: Use a permitted operator.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

Comparison of incompatible types!

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent" connector does not meet the syntactic conventions. Constants or variables respectively, are compared which are of different types.

Action: Change the type of the variable used or replace the constants accordingly.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

Logical operator or end expected!

<*Correctly interpreted part of the condition*>

Cause: The value in the attribute "Transition Condition" of a "Subsequent" connector does not meet the syntactic conventions. The logical expression can only be extended using one of the logical operators "AND" or "OR".

Action: Use one of the operators mentioned or remove the additional part from the condition accordingly.

Message: Error in condition!

From: <*object name*>

To: <*object name*>

Model: <*model name*>

The types of variables or constants on the left and on the right side of the comparison do not match!

<*Correctly interpreted part of the condition*>

Cause: In the value of the attribute "Transition Condition" of a "Subsequent" connector the types of variables and constants do not match. A continuously distributed (exponentially, equally or normally distributed) variable must be compared to a numerical constant (e.g. 100), a discretly distributed variable to an alphanumeric constant (e.g. 'YES').

Action: Change either the constant in the transition condition or the value of the attribute "Variable Type" in the object belonging to the class "Variable" accordingly

[aproobjs-39]

Message: Illegal assignment to dynamic role variable!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: A value is assigned to a role variable which ADOxx has already filled with a performer by a "done by" expression (attribute of an activity). (see AQL Rule No. 9)

Action: Use a different variable name.

[aproobjs-40]

Message: Error while translating a process calendar:

<*object class*>: <*object name*>

Model: <*model name*>

Cause: A runtime error occurred while a process calendar was being translated.

Action: Should the error occur again when the simulation is restarted, export the model into ADL and inform your ADOxx consultant.

[aproobjs-41]

Message: Invalid process calendar or no process calendar specified.

<*object class*>: <*object name*>

Model: <*model name*>

Cause: No process calendar was specified for the object listed or a process calendar which does not meet the syntactic conventions was specified.

Action: Should the error re-occur when the simulation is restarted, open the model listed and try to edit the process calendar in the ADOxx notebook of the object specified.

Should an error also occur here, export the model into ADL and inform your ADOxx administrator.

[aproobjs-42]

Message: Incorrect process calendar or no process calendar specified.

<*object class*>: <*object name*>

Model: <*model name*>

Cause: No process calendar was specified for the object listed or a process calendar which does not meet the syntactic conventions was specified.

Action: Should the error re-occur when the simulation is restarted, open the model listed and try to edit the process calendar in the ADOxx notebook of the object specified.

Should an error also occur here, export the model into ADL and inform your ADOxx administrator.

[aproobjs-43]

Message: You must specify a number 0 and 100 for the minimum quota of presence.

<*object class*>: <*object name*>

Model: <*model name*>

Cause: A minimum quota of presence outside the range from 0 to 100 was specified for the object listed

Action: Enter a value between 0 and 100 for the minimum quota of presence.

[aproobjs-44]

Message: You must specify a number greater than 0 for the average number of participants.

<*object class*>: <*object name*>

Model: <*model name*>

Cause: An average number of participant less than or equal to 0 was specified for the object listed.

Action: Enter a value greater than 0 for the average number of participants.

[aproobjs-45]

Message: Expected merging object before the end of the process! If there is a merging object, you possibly modelled a loop that leads somewhere before the corresponding parallelity object.

<*object class*>: <*object name*>

Model: <*model name*>

Cause: An attempt was made to leave the process model specified at the object indicated, although at least one merging object was still expected.

Action: Check whether all parallel paths introduced by the parallelism objects are rejoined by merging objects.

[aproobjs-46]

Message: Parallel execution paths initiated by the same split object must be merged by the same join object!

<*object class*>: <*object name*>

Model: <*model name*>

Cause: Parallel paths introduced by the same split object are brought together by different merge objects.

Action: Use the same merging object to join the parallel paths of a split object (parallelism) again. If this is not what you intended to do, you must introduce the parallel paths via different split objects.

© BOC Group 2008

[aproobjs-47]

Message: Index used more than once!

<*object class*>: <*object name*>

Index: <*Index*>

Model: <*model name*>

Cause: The index indicated was used for several instances of relations which have their origin in the object mentioned.

Action: Use unambiguous indices for the relations originating at the object mentioned.

[aproobjs-48]

Message: A cooperative synchronous activity must not employ resources!

Cause: Your model contains a cooperative synchronous activity which employs resources. This is not permitted.

Action: Remove the relation to the resource instance or change the activity into a non cooperative or a cooperative asynchronous one.

[aproobjs-49]

Message: The probability of a "Subsequent" relation must be in the range between 0 and 1!

Cause: The transition condition contains a probability outside the range permitted.

Action: Change the probability to a value in the range between 0 and 1.

[aproobjs-50]

Message: The outgoing relations of a decision contain both conditions and probabilities!

Cause: The transition conditions of a decision's outgoing edges must be uniformly filled either with conditions or with probabilities.

Action: Change the transition conditions of the decision listed in such a way that they contain either conditions or probabilities.

[aproobjs-51]

Message: The sum of the probabilities of all outgoing edges of a Decision must be 1!!

If necessary, check if the correct attribute is assigned (attribute "transition condition" for the simulation or attribute "transition probabilities" for the analytic evaluation).

Cause: There are either no transition conditions specified at the outgoing edges of the specified decisions (or transition probabilities) or the sum of the specified probabilities is not the value 1.

Action: Change the probabilities so that the sum makes 1.

Note that each different attributes are relevant for the simulation and for the analytic evaluation: The simulation uses the value of the attribute "transition conditions". The

analytic evaluation uses the value of the attribute "transition probabilities". You must assign the right attributes, according to what you are using, either the simulation or the analytic evaluation.

[aproobjs-53]

Message: A macro of the name specified is not defined!

Macro: <*macro name*>

Model: <*model name*>

Cause: You are attempting to use a macro name which is not defined.

Action: Check whether the macro name used is already defined anywhere else. If not, replace it with a macro that is already defined or define a macro in another place with the name listed above.

[aproobjs-55]

Message: The process model is being used recursively!

Cause: In contrast to the simulation, the Analytical evaluation function does not allow recursions.

Action: Make sure that no process model calls itself either directly or indirectly as a subprocess.

[aproobjs-56]

Message: Jumping from a parellel path back before the corresponding split object occured!

Cause: It is not permitted to jump back before the split object (parallelity) from a path which follows a split and which has not yet been brought together by a merge object with the parallel paths from the same split.

Action: If you wish to repeat this same parallelism, there are two possibilities:

1. You copy the respective parallelism.

2. You first bring the parallel paths together and then jump back before the split object. If this is not what you intend to do, remove the relation which caused the error.

[aproobjs-57]

Message: The process model may contain an infinite loop! If necessary, put up the maximum loop length!

Model: <*model name*>

Cause: The specified process model apparently contains an infinite loop. The analytic evaluation can not be continued.

Action: Verify the specified model and remove the infinite loop . If there is no infinite loop, increase the value "maximum loop length" in the window "analytic evaluation".

[aproobjs-58]

Message: The number of paths exceeds the upper limit! Either extend the upper limit or use the simulation!

Cause: The actual number of different process paths exceed the "Max. number of paths" specified in the window "Analytical evaluation". Therefore, the Analytical evaluation is cancelled.

Action: Either increase the upper limit for the maximum number of paths or use simulation for evaluating the processes.

[aproobjs-59]

Message: A process model to which a maximum waiting time is assigned cannot start with a parallelism!

Cause: A model to be simulated starts with a parallelism, although a value greater than 00:000:00:00:00 is entered in the attribute "maximum waiting time" (Process start).

Action: Remove the parallelism or set the maximum "Waiting time" by the value 00:000:00:00:00.

[aproobjs-60]

Message: The "continuous execution" cannot be activated at the same time as the "central task stack"!

Cause: The "continous execution" implies that the activity of a specific performer is executed, whereas "central task stack" means that any performerwho has time will execute the activity. Both concepts are contradictory and therefore cannot be activated at the same time.

Action: Deactivate either the "continuous execution" or the "central task stack" option.

[aproobjs-61]

Message: The option "cooperative" cannot be activated at the same time as "central task stack"!

Cause: Cooperative activities will always be executed by a group of specific performers. The "central task stack" implies on the contrary, that any performer,who has time, will execute the activity. Both concepts are contradictory and so cannot be activated at the same time.

Action: Deactivate either the "cooperative" or the "central task stack" option.

[aproobjs-62]

Message: The current simulation path is exceeding the maximum length allowed! Either one of the models being simulated contains an infinite loop or the current simulation path is too long!

Cause: A path through the simulated model is exceeding the maximum length allowed of some 8000 - 9000 objects following each other (the maximum path length cannot be

exactly specified, since it depends on the situation.). Paths of this length are normally due to infinite loops contained in the simulated models.

Action: Correct the model accordingly and restart the simulation.

[apwchgui-01]

Message: Password must not be shorter than three characters.

Cause: You entered a password shorter than three characters.

Action: Enter a password which is at least three characters in length.

[apwchgui-02]

Message: Passwords are not consistent.

Cause: You entered different passwords in the two input fields.

Action: Enter the same password in both input fields.

[apwchgui-03]

Message: The old password is wrong.

Cause: You entered an incorrect password.

Action: Enter your previous password correctly.

[apwchgui-04]

Message: ADOxx could not connect to the database. Task stopped.

Cause: The connection to the ADOxx database server may have been interrupted.

Action: Repeat the action. Should the error re-occur, please contact your system administrator.

[apwchgui-05]

Message: The new password contains restricted characters. Please use characters in the ASCII (resp. ANSI) range 32 to 126 only (exceptions: ' \ { } and @).

Cause: You entered illegal restricted characters or umlauts when entering the password.

Action: The characters permitted for the password are only the numbers **0 to 9**, the letters **a to z** and **A to Z**, blanks and the symbols **! # $ % & ( ) * + , - . / : ; < = > ?** @ **[ ] ^ _ { | } ~**. Umlauts, ß and other symbols must not be used. Please enter the password according to the restrictions above.

[apwchgui-06]

Message: The new password does not match the password rule:

<*Rule*>

Cause: The password you entered does not match the password rule defined by the ADOxx administrator

Action: For the password certain rules have been defined.

The characters permitted for the password are only the numbers **0 to 9**, the letters **a to z** and **A to Z**, blanks and the symbols **! # $ % & ( ) * + , - . / : ; < = > ?** @ **[ ] ^ _ { | } ~**. Umlauts, ß and other symbols must not be used.

Please enter a password according to the restrictions above.

[areena-03]

Message: The Modelling Toolkit could not be started because another ADOxx application is already active.

Close the other ADOxx application in order to be able to start the Modelling Toolkit.

Cause: No more ADOxx applications can be run under OS/2. In order to start the Modelling Toolkit, you must first exit another application.

Action: When you exit another ADOxx application, you will be able to start the Modelling Toolkit.

[areena-04]

Message: The user-specific configuration of ADOxx could not be loaded.

Cause: Due to a database error, the user-specific configuration of ADOxx could not be loaded.

Action: Restart ADOxx. If the message re-occurs, contact your system administrator.

[areena-05]

Message: The relation redefinition contains errors. Contact your ADOxx administrator.

Cause: The configuration of your library contains errors.

Action: Contact your ADOxx administrator.

[areena-07]

Message: The initialising of ADOxx is wrong!

An error has occurred when searching for the user specific library!

The login process will be aborted.

Cause: The configuration of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[areena-08]

Message: The initialising of ADOxx is wrong!

Error on loading the meta model structures!

The login process will be aborted.

Cause: The configuration of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[areena-09]

Message: The initialising of ADOxx is wrong!

Error on creating the meta model!

The login process will be aborted.

Cause: The configuration of your ADOxx database contains errors.

Action: Note down the internal error code and contact your ADOxx administrator.

[areena-10]

Message: The initialising of ADOxx is wrong!

Error on loading the user-specific library!

The login process will be aborted.

Cause: The configuration of your application library or of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[areena-11]

Message: ADOxx can't be used with the desired language <*language*> since

the licence number is not configured for this language.

The login process will be aborted.

Cause: The access rights defined in the licence number do not allow a login in the selected language.

Action: Contact your ADOxx administrator.

[aregex-01]

ATTENTION: The error message [aregex-01] contains different text parts which are all shown below. Further information can be retrieved from the chapter "Regular Expressions" (see chap. 6., p. 731).

Message: The entered search pattern is not a valid regular expression!

Please note the following information:

<*error-specific message*>

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry accordingly.

Message: The entered search pattern is not a valid regular expression!

Wrong masking (trailing '\').

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using the masking (back slash) properly.

Message: The entered search pattern is not a valid regular expression!

Missing bracket ('[' without ']' or vice versa).

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using the brackets properly.

Message: The entered search pattern is not a valid regular expression!

Missing bracket ('(' without ')' or vice versa).

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using the brackets properly.

Message: The entered search pattern is not a valid regular expression!

Missing bracket ('{' without '}' or vice versa).

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using the brackets properly.

Message: The entered search pattern is not a valid regular expression!

Wrong numerical range specified (not a number, number too large, more than two numbers or first number larger than second).

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using the numerical range properly (e.g. {2,3}).

Message: The entered search pattern is not a valid regular expression!

Invalid range specified.

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using a valid range (e.g. [a-z]).

Message: The entered search pattern is not a valid regular expression!

Missing expression before '?', '*' or '+'.

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using an expression before '?', ' *' oder '+'.

Message: The entered search pattern is not a valid regular expression!

Invalid collating element.

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using a valid collating element.

Message: The entered search pattern is not a valid regular expression!

Invalid character class.

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using a valid character class (e.g. [:alpha:]).

Message: The entered search pattern is not a valid regular expression!

Invalid backreference number.

Cause: The search pattern entered does not meet the syntactic requirements of regular expressions.

Action: Please correct the entry using a backreference number to an existing pattern.

[aregex-02]

Message: Not enough memory!

Search could not be performed.

Cause: The search cannot be executed because of not enough free memory.

Action: Exit other programs and try again.

[arelrdef-02]

Message: Error in library attribute "*name*" ("*library name*"):

"*name*" is not a valid relation name!

Cause: There are no existing relations with the specified name.

Action: Change the corresponding library attribute so that it uses an existing relation.

[arelrdef-03]

Message: Error in library attribute "*name*" ("*library name*"):

There is an error in attribute "*attribute name*" !

Cause: An error has occurred in the specified attribute. Possible causes:

The attribute has been specified without its class or relation.

The specified attribute does not exist in the specified class or relation.

The attribute type specified in the parameter "type" is invalid.

The value specified in the parameter "value" could not be assigned to the new attribute.

Action: The following error actions are possible:

Before you specify an attribute, you must specify the class/relation it belongs to.

Verify the name of the attribute, since it has not been found in the previously specified class/relation.

Verify if the value specified in the parameter "type" is available in the following list: "integer", "double", "string", "distribution", "time", "enumeration", "enumerationlist", "longstring" or "programcall".

Change the value of the parameter "value" in such a way that it meets the type specified in the parameter "type".

[arelrdef-04]

Message: Error in library attribute "*name*" ("*library name*"):

There is an error in the facet "*facet name*" !

Cause: An error has occurred in the specified facet. Possible causes:

The facet has been specified without its belonging attribute.

The parameter "value" has not been specified.

The specified facet does not exist in the specified attribute.

The value specified in the parameter "value" could not be assigned to the facet.

Action: The following error actions are possible:

Before you specify a facet, you must specify the belonging attribute.

Verify if the parameter "value" has been specified.

Verify the name of the facet, since it has not been found in the previously specified attribute.

Change the value of the parameter "value" in such a way that it corresponds to the type of the facet.

[arelrdef-05]

Message: Error in library attribute "*name*" ("*library name*"):

"*class name*" is not a valid class name!

Cause: When creating a new relation or changing an existing meta model class, an invalid class name has been specified.

Action: Verify the name of the origin and target classes defined for the relation. Check the names of the meta model classes, which should be changed.

[areposui-01]

Message: "*name*" already exists. Select another name!

Cause: An attribute profile or an attribute profile group with the specified name already exists. Attribute profiles must have globally unambiguous names (i.e. you cannot have an attribute profile with the same name in no other attribute profile group), the name of attribute profile groups must only be unambiguous inside your own group.

Action: Specify an unambiguous name.

[areposui-02]

Message: "*attribute profile (groups) name*" is currently being edited by another ADOxx user and so cannot be edited!

Cause: You have attempted to edit the displayed attribute profile or the displayed attribute profile group, but the profile or the profile group is currently edited by another user.

Action: Wait until the other user has finished the editing. If necessary, you can send an express-message to the user.

[areposui-05]

Message: The entry "*value*" is not allowed for the attribute "*name*"

Cause: You have entered an invalid attribute value.

Action: Enter a valid value.

[areposui-06]

Message: The changes could not be saved! Update the attribute profile list and execute the action again.

(Error code: *code*)

Cause: The attribute profile or the attribute profile groups concerned have perhaps in the meanwhile been deleted by another user. The connection to the database may also be interrupted.

Action: Update the attribute profile list and then carry out the action again. Pay attention to the error messages (e.g. database error), which could also be deleted. If the problem reoccurs, note the specified error code and contact your ADOxx administrator.

[areposui-08]

Message: The attribute profile has already been deleted by another ADOxx user. Update the attribute profile list.

Cause: The attribute profile has already been deleted by another ADOxx user.

Action: Update the attribute profile list.

[areposui-09]

Message: "*name with version number*" already exists. Select another version number!

Cause: An attribute profile with this name and version number already exists.Attribute profiles must have globally unambiguous names (i.e. there shall not be an attribute profile group with the same name in another attribute profile group).

Action: Give another name or another version number.

[areposui-10]

Message: Not all attribute profile groups could be deleted. Some of the attribute profiles contained in the groups are possibly already used and so could not be deleted. New attribute profiles may also have been created meanwhile by another ADOxx user in the group concerned. Update the attribute profile list and try again to delete the groups.

Cause: You have attempted to delete one or more attribute profile groups. This is not possible according to the specified reasons.

Action: Update the attribute profile list and then try again to delete the groups.

[areposui-11]

Message: The attribute profile group "*name*" is currently updated by another ADOxx user. This is why no new attribute profiles could be created at the moment. Try again later.

Cause: An other user is currently editing the current attribute profile group. To avoid inconsistencies, it is not possible in the meantime to create new attribute profiles in this attribute profile group.

Action: Wait until the other user has finished his editing and try it again.

[areposui-12]

Message: The selected attribute profiles are not based on the same class!

Cause: The attribute profiles selected for editing are not instances from the same class.

Action: For the editing, select attribute profiles from only one class.

[areposui-13]

Message: The move in the attribute profile group "<*group*>" is wrong!

This can be due to the following causes:

- the attribute profile group has meanwhile been deleted by another user.

- you have attempted to move a super ordinated group to a group which is one of its subordinated groups.

- An attribute profile group with the same name as the attribute profile group to move already exists in the target group.

If necessary update the attribute profile list and repeat the action.

Cause: 1. The attribute profile group has been meanwhile deleted by another user.

2. You have attempted to move a subordinated group to one of its subordinated groups.

3. An attribute profile group with the same name as the attribute profile group to move already exists in the target group.

Action: Depending on the cause, the move to the selected target group is under impossible conditions.

[areposui-14]

Message: The attribute profile group concerned has been already deleted by another ADOxx user.

Update the attribute profile list and then repeat the action.

Cause: The attribute profile group concerned does not exist any more.

Action: Update the attribute profile list and repeat the action for as long as necessary.

[aresattr-02]

Message: Server error while saving.

Cause: After the simulation results were deleted an error occurred while saving.

Action: Try to delete the simulation result attributes again. If the error occurs again, contact your ADOxx Administrator.

[aresattr-03]

Message: Error in the application library attribute sim result mapping.

Cause: The application library's definition is incorrect in the attribute sim result mapping.

Action: Contact your ADOxx administrator or if necessary your ADOxx consultant.

[arightmg-01]

Message: Could not define new user group!

Cause: An error occurred while a user group was about to be created.

Action: Quit ADOxx and restart it. Should this error re-occur, please contact your system administrator.

[arightmg-02]

Message: Could not delete selected user group!

Cause: An error occurred while a user group was about to be deleted.

Action: Quit ADOxx and restart it. Should this error re-occur, please contact your system administrator.

[arightmg-03]

Message: No new name was entered!

Cause: You did not enter a new name when renaming the group.

Action: Please enter a new name when renaming the group.

[arightmg-05]

Message: An error has occurred when loading the user group containing the user !

Cause: Not all user groups could be loaded.

Action: Quit ADOxx and restart it. Should this error re-occur, please contact your ADOxx administrator.

[arightmg-06]

Message: All user groups to which the user is assigned could not be loaded!

Cause: An error occurred while the user groups to which the user is assigned were about to be loaded.

Action: Quit ADOxx and restart it. Should this error re-occur, please contact your ADOxx administrator.

[arightmg-07]

Message: An error has occurred when creating the model group <*model group name*>!

Cause: An error occurred while a new model group was about to be defined/generated.

Action: Update the model group list and repeat the action. Should the error reappear, exit ADOxx and re-start the application. Should the error appear again, contact your ADOxx administrator.

[arightmg-08]

Message: An error has occurred when deleting the model group <*model group name*>!

Cause: An error occurred while deleting a model group.

Action: Update the model group list and repeat the action. Should the error appear again, exit ADOxx and re-start the application. Should then error occur repeatedly, contact your ADOxx administrator.

[arightmg-11]

Message: An error has occurred when assigning the user <*user name*> to the user group <*user group name*>!

Cause: An error occurred while users were about to be assigned to a user group.

Action: Quit ADOxx and restart it. Should this error re-occur, please contact your ADOxx administrator.

[arightmg-12]

Message: An error has occurred when assigning the user group <*user group name*> to the user <*user name*>!

Cause: An error occurred while user groups were about to be assigned to a user.

Action: Quit ADOxx and restart it. Should this error re-occur, please contact your ADOxx administrator.

[arightmg-13]

Message: An error has occurred during the assignment of <*model name*> (*model type*) to model group <*model group name*>!

The model may have possibly been deleted by another user .

Cause: An error has occured during the assignment of a model to a model group, as this model group may have meanwhile been deleted by another user.

Action: Close the error message - the model group list will automatically be updated - and then repeat the action. Should the error appear again, exit ADOxx and restart the application. Should the error occur repeatedly, contact your ADOxx administrator.

[arightmg-17]

Message: An error has occurred when assigning the user group to the <*user group name*> model group <*model group name*>!

Cause: The access rights of the user group could not be set to the model group. The user or model group has possibly been deleted by another ADOxx user in the meanwhile.

Action: Update the user group list or the model group list and repeat the action. Should the error appear again, exit ADOxx and restart the application. Should the error occur repeatedly, contact your ADOxx administrator.

[arightmg-19]

Message: The selected user group <*user group name*> is not empty and cannot be deleted!

Cause: A user group which still contains users may not be deleted.

Action: Remove all user references from the user group, either by changing the assignment of users or by deleting users from the ADOxx database. Then start the deletion process again.

[arightmg-20]

Message: The selected model group <*model group name*> is not empty and cannot be deleted!

Cause: A model group, which still contains other model groups or models, may not be deleted.

Action: Remove all model references and model groups from the model group, either by changing the assignment of models or by deleting the models - or model groups, respectively - from the ADOxx database. Then start the deletion process again.

[arightmg-21]

Message: A model group with the name <*model group name*> already exists!

Cause: All model groups contained in a model group must have a non-ambiguous name.

Action: Update the model group list to be aware of changes of other ADOxx users. Rename the model group with a new name that is non-ambiguous within the hierarchically superordinated group and restart the action.

[arightmg-22]

Message: A user group with the name <*user group name*> already exists!

Cause: Each user group must have a non-ambiguous name.

Action: Rename the user group with a new non-ambiguous name and start the action anew.

[arightmg-23]

Message: An error has occurred when renaming the user group <*user group name*> could not be renamed!

Cause: An error occurred while one of the user groups was about to be renamed.

Action: Quit ADOxx and restart it. Should this error re-occur, please contact your ADOxx administrator.

[arightmg-24]

Message: An error has occurred when renaming the model group <*model group name*>!

Cause: An error occurred while a model group was about to be renamed.

Action: Update the model group list and repeat the action. Should the error appear again, exit ADOxx and re-start the application. Should the error appear repeatedly, contact your ADOxx administrator.

[arightmg-25]

Message: An error has occurred when moving the reference to the model <*model name*> (<*model type*>)!

Cause: An error occurred while a model was about to be moved.

Action: Update the model group list and repeat the action. Should the error appear again, exit ADOxx and restart the application. Should the error appear repeatedly, contact your ADOxx administrator.

[arightmg-27]

Message: An error has occurred when copying the reference in the model <*model name*> (<*model type*>) to the model group <*model group name*>!

Changes have possibly been carried out by another user. Update the model group list.

Cause: An error has occurred when copying a model reference. The target model group may have been deleted by another user in the meanwhile.

Action: Update the model group list and repeat the action. Should the error appear again, exit ADOxx and re-start the application. Should the error appear repeatedly, contact your ADOxx administrator.

[arightmg-29]

Message: When moving the model group <*model group name*> into model group <*target-model group name*> an error has occurred!

Cause: An error has occurred when moving a model group.

Action: Update the model group list and repeat the action. Should the error appear again, exit ADOxx and re-start the application. Should the error appear repeatedly, contact your ADOxx administrator.

[arightmg-30]

Message: The model group <*target model group name*> already contains a model group with the name <*model group name*>!.

Cause: A model group has been moved into a group which already contains a group of the same name.

Action: Please move the model group to a different group.

[arightmg-31]

Message: Cannot create a new main model group!

Cause: An error occurred while a new main model group was about to be created.

Action: It is not possible to create new main model groups in the Modelling Toolkit. Contact your ADOxx administrator, who can create new main model groups with the Administration Toolkit.

If this error occurs in the Administration Toolkit, update the model group list and repeat the action. Should the error re-occur, exit ADOxx and re-start the application. Should the error appear repeatedly, contact your ADOxx administrator.

[arightmg-32]

Message: An error has occurred when deleting the reference of a model <*model name*> (*model type*)!

Cause: An error occurred while a model reference was about to be deleted.

Action: Update the model group list and repeat the action. Should the error appear again, exit ADOxx and re-start the application. Should the error appear repeatedly, contact your ADOxx administrator.

[arightmg-33]

Message: The target model group <*model group name*> is write protected and cannot be changed!

Cause: You do not have sufficient rights to change the respective target model group you wish to change.

Action: Contact your ADOxx administrator. He is responsible for assigning access rights on models and model groups to users.

[arightmg-34]

Message: The reference to <*model name*> (*model type*) cannot be deleted, since the model is no longer contained in any model group!

Cause: There shall always be at least one reference to each model.

Action: Delete the last reference to a model using the Administration Toolkit or contact your ADOxx administrator.

[arightmg-35]

Message: The model <*model name*> (*model type*) cannot be assigned to a model group!

Cause: The specified model could not be assigned to the selected model group.

Action: Update the model group list and repeat the action. Should the error occur again, exit ADOxx and restart the application. Should the error appear again, contact your ADOxx administrator.

[arightmg-36]

Message: The reference from the model <*model name*> (*model type*) cannot be reassigned, since the model is currently opened with write access by the user <*user name*> !

Cause: The model is currently being edited by another user. The reference cannot be moved.

Action: Make sure that the user which is currently editing the model, closes it or you can move the model reference later.

[arightmg-37]

Message: The reference to the model <*model name*> (*model type*) cannot be deleted, since the model is currently opened with write access by the user <*user name*>!

Cause: The model is currently edited by another user. The reference cannot be deleted.

Action: Make sure that the user which is currently editing the model, closes it or you can delete the model reference later.

[arightmg-38]

Message: The reference to the model "<*model name*>" (<*model type*>) cannot be copied, since the model is currently opened with write access by the user <*user name*>!

Cause: The model is currently edited by another user. The reference cannot be copied.

Action: Make sure that the user which is currently editing the model, closes it or you can copy the model reference later.

(You can also send an appropriate message to the specified user via ADOxx mail (menu "Extras", menu item "Message").)

[arightmg-39]

Message: The model group "<*model group name*>" cannot be moved, since the following models it contains are currently opened with write access by the following users:

<*model name*> (<*model type*>) - <*user name*>

Cause: Models from the specified model groups are being edited by other users. The model group cannot be moved.

Action: Make sure that the user, who is currently editing the model, closes it or you can move the model group later.

(You can also send an appropriate message to the specified user via ADOxx mail (menu "Extras", menu item "Message").)

[arightmg-40]

Message: The reference to the model "<*model name*>" (<*model type*>) cannot be assigned, since the model is currently opened with write access by the user <*user name*>!

Should the action be continued because of the expanded administrator rights?

Cause: The model is currently being edited by another user. The model reference cannot be reassigned.

Action: Make sure that the user, who is currently editing the model, closes it or you can arrange the model reference later.

(You can also send an appropriate message to the specified user via ADOxx mail (menu "Extras", menu item "Message").)

[arightmg-41]

Message: The reference to the model "<*model name*>" (<*model type*>) cannot be deleted, since the model is currently opened with write access by the user <*user name*>!

Should the action be continued because of the expanded administrator rights?

Cause: The specified model is currently being edited by another user. The model reference cannot be deleted.

Action: Make sure that the user, who is currently editing the model, closes it or you can delete the model reference later.

(You can also send an appropriate message to the specified user via ADOxx mail (menu "Extras", menu item "Message").)

[arightmg-42]

Message: The reference to the model "<*model name*>" (<*model type*>) cannot be copied, since the model is currently opened with write access by the user <*user name*>!

Should the action be continued because of the expanded administrator rights?

Cause: The specified model is currently edited by another user. The model reference cannot be copied.

Action: Make sure that the user, who is currently editing the model, closes it or you can copy the model reference later.

(You can also send an appropriate message to the specified user via ADOxx mail (menu "Extras", menu item "Message").)

[arightmg-44]

Message: Insufficient rights for at least one of the model groups selected!

Cause: You have no write access on at least one of the model groups selected.

Action: Update the model group list and/or select only the model groups for which you have write access rights.

[arightmg-45]

Message: You have no write access to the super ordinated model group of the model "<*model name*>"! Update the model group list or select another model group.

Cause: You have no write access to the super ordinated model group of the selected model!

Action: Update the model group list or select another model group.

[arightmg-46]

Message: You do not have sufficient rights to delete or move the model group "<*model group name*>"!

Cause: You have no write access rights to the specified model group and therefore cannot delete or move it.

Action: If necessary, contact your ADOxx administrator, to obtain write access rights to the specified model group.

[arightmg-47]

Message: The user group structure has been changed by another "ADOxx" user.

Update the user group list!

Cause: The user group structure has been changed by another "ADOxx" user.

Action: Update the user group list!

[arightmg-48]

Message: The model group "<*model group name*>" cannot be moved to a specific subgroup.

Cause: You have attempted to move a model group to one of your subgroups.

Action: Move the specified model group to another model group, which is not a subgroup of the specified model group.

[arightmg-49]

Message: The reference to the model <*model name*> (*model type*) cannot be moved, since you do not have enough access rights in the target model group.

Cause: You have attempted to move a model reference to a model group, in which you have no write access rights.

Action: You cannot move the reference to the selected model group. If necessary, contact your ADOxx administrator to adapt your rights.

[as2rdlg-01]

Message: Specify the class and relation.

Cause: You have selected a simulation representation, which enables you to make statements about Working Environment groupings (roles, organisational units etc.). These results require the selected grouping to be specified.

Action: Specify the class and the relation.

[as2rdlg-02]

Message: Some of the models into which the results are to be transferred have been loaded write-protected. The transfer is not possible, because it would cause the simulation results to be mixed up and thus cause inconsistencies.

Cause: You simulated at least one model that is loaded in read-only state and want to transfer the simulation results for evaluation purposes into these models. This is not possible, since models which were loaded in read-only state cannot be saved.

Action: Close the simulation result display and the models which were loaded as read-only. Then load them again, this time without write-protection. After this, start the evaluation again.

[as2rdlg-03]

Message: The model "<*model name*>" has been closed during the simulation. It is therefore not possible to transfer the simulation results.

Cause: You have prematurely closed the model mentioned. This is why you cannot transfer the simulation results to the model. For consistency reasons, the transfer of simulation results to other models will also be stopped.

Action: Carry out again the simulation and close the models only after the transfer of the simulation results.

[as2rdlg-04]

Message: There is no sim result mapping defined. The simulation results can therefore not be transferred.

Cause: Since the entries of the library attribute sim result mapping are empty, the results cannot be transferred.

Action: Contact your ADOxx administrator.

[as2rdlg-05]

Message: Model "<*model name*>": There are no objects to which the grouping specified could apply. Please select a different class or a different relation.

Cause: In the model specified, there is no performer which is linked to the class selected by the relation selected. For this grouping, no results can be displayed.

Action: Select a different class and/or a different relation in the group "Working Environment".

[as2rdlg-06]

Message: The simulation results could not be calculated.

The result calculation period (workload analysis) was possibly too short or the simulated model contain no object, for which you can calculate the results.

Cause: For the last selected models, you have entered a result calculation period which is too short or the simulated models contains no object, for which you can calculate the results.

Action: Define a longer results calculation period or expand the models to the appropriate objects and repeat the simulation.

[ascript-01]

Message: <*Token*> is neither a key word nor a defined procedure!

Cause: The AdoScript contains no command, which is either defined as internal or as a procedure.

Action: Check and correct the AdoScript.

[ascript-02]

Message: <*Token*> without CASE or OBJECTIF!

Cause: The attribute contains an element, in which a preceding CASE or OBJECTIF element is missing.

Action: Check and correct the AdoScript.

[ascript-03]

Message: <*Token*> without ITEM!

Cause: The attribute contains AdoScript commands, which are not assigned to a menu item.

Action: Check and correct the AdoScript.

[ascript-05]

Message: <*Token1*> without <*Token2*>!

Cause: The AdoScript contains a branching or a loop end, in which the opening token is missing.

Action: Check and correct the AdoScript.

[ascript-06]

Message: <*Token1*> without <*Token2*>!

Cause: The AdoScript contains a branching or a loop end, in which the ending token is missing.

Action: Check and correct the AdoScript.

[ascript-09]

Message: CC without message element!

Cause: The AdoScript contains a CC call, which is not followed by a message element.

Action: Check and correct the AdoScript.

[ascript-10]

Message: CALL with defective attribute function: "<*Text*>"!

Cause: The value of the parameter function is not a valid function specification.

Action: Check and correct the AdoScript.

[ascript-11]

Message: No function "<*function name* is existing in the DLL "<*file name*>">"!

Cause: Using CALL in the AdoScript, you have attempted to call a function which is not defined in a DLL.

Action: Check and correct the AdoScript.

[ascript-12]

Message: "<*Token*>" is not a command in message port "<*message port*>"!

Cause: In the AdoScript, using SEND or CC, a command has been sent to the specified message port, where it is not defined .

Action: Check and correct the AdoScript.

[ascript-13]

Message: message port "<*Name*>" does not exist!

Cause: In the AdoScript, using SEND or CC, a command has been sent to the specified, non existing message port.

Action: Check and correct the AdoScript.

[ascript-14]

Message: defective call of <*procedure name*>!

Cause: When calling this procedure, the parameters specified are not the exact same parameters that are specified in the procedure definition.

Action: Check and correct the AdoScript.

[ascript-15]

Message: PROCEDURE without definition element!

Cause: The AdoScript contains a PROCEDURE element, which is not followed by a definition element.

Action: Check and correct the AdoScript.

[ascript-16]

Message: Type incompatibility in a procedure call!

<*function call*>

<*context*>

Cause: When calling the procedure, at least one parameter is not of the type defined in the procedure definition.

Action: Check and correct the AdoScript at the specified place.

[ascript-17]

Message: The old syntax (IF...ENDIF) cannot be used together with the new syntax (IF...{...}) !

Cause: The AdoScript contains both constructs of old and of new syntax.

Action: Check and correct the AdoScript.

[ascript-18]

Message: Starting the process

<*Process name*>

failed!

Cause: The START command has been called from an AdoScript but the program could not be started.

Action: Check and if necessary correct the AdoScript. Make sure that the program to be started exists in the respective directory.

[ascript-19]

Message: LEO call with unauthorised index (set-cur-elem-index:<*index*>)!

Number of parsed elements: <*number*>

Cause: The LEO command has been called in an AdoScript with the index specified with set-current-index outside the authorised area (from 0 to (number of parsed elements) - 1).

Action: Check and correct the AdoScript.

[ascript-20]

Message: LEO call with <*indication* > without previous parse!

Cause: The LEO command has been called in an AdoScript, where no text to be parsed has been specified. However, this must be the case for each call of LEO.

Action: Check and correct the AdoScript.

[ascript-21]

Message: The external DLL "<*DLL name*>" could not be loaded!

Cause: Within an AdoScript the specified DLL file is called from a CALL command although the file dos not exist or is not accessible.

Action: Enable access to the specified DLL file or correct the CALL command.

[ascript-22]

Message: Abnormal terminition of function "<*Funktionsname*>" in DLL"<*DLL name*>"!

Please contact your ADOxx consultant if this problem persists.

Cause: Within an AdoScript an executed CALL command was aborted due to a fatal error within the specified DLL file.

Action: Correct the CALL command or update the DLL file with a corrected version.

[ascript-23]

Message: Double definition of ON_EVENT <*event name*>!

Cause: In the attribute "external linking" there are more than one event handlers defined for one event.

Action: Check and correct the AdoScript.

[ascript-25]

Message: PROCEDURE <*procedure name*> is currently being executed and therefore cannot be overwritten!

Cause: An AdoScript with the specified procedure is executed. The procedure exists already and is executed simultaneously. Therefore the procedure overwrites itself.

Action: Ensure that the specified procedure is defined only once (globally).

[asetref-01]

Message: The entered value is invalid!

Cause: The value specified for depth of the reference is invalid. (only non negative, whole numbers smaller than 65535 are authorised.)

Action: Enter a valid value.

[asetref-02]

Message: The maximum depth of 65535 has been exceeded!

Cause: The value specified for the depth of the reference cannot exceed 65535.

Action: Enter a smaller value.

[asetsel-01]

Message: The parameter of the selected arrangement function could not be changed!

Cause: An error occurred while attempting to change the parameter of the selected arrangement function.

Action: Close ADOxx and re-start it. If this error re-occurs, contact your ADOxx administrator.

[asetsel-02]

Message: No quotation marks may be used in the name of the arrangement function and the name may not be blank!

Cause: Quotation marks were specified in the name of the arrangement function or the name was left blank.

Action: Enter a new name.

[asetsel-04]

Message: The new arrangement function could not be added!

Cause: An error occurred when attempting to add a new arrangement function.

Action: Close ADOxx and re-start it. If this error re-occurs, contact your ADOxx administrator.

[asetsel-05]

Message: The arrangement function could not be deleted!

Cause: An error occurred when attempting to delete an arrangement function.

Action: Close ADOxx and re-start it. If this error re-occurs, contact your ADOxx administrator.

[asetsel-08]

Message: An arrangement function with this name already exists !

Cause: A unique name must be entered for each arrangement function.

Action: Retry the action but enter a different name.

[asetsel-10]

Message: The process hierarchy function is not defined for <*model type*>!

Cause: The arrangement function is not defined for this model type.

Action: Open a model or create a model for which it is possible to build a hierarchy. If the model type required for this is not known, then contact your ADOxx administrator, who is competent in the configuration of the (not user-definable) process hierarchy.

[asgmlexp-01]

Message: Error while writing the export data.

Cause: An error occured while writing the data. Possibly there is not enough space on the disk.

Action: Check if there is enough space on the disk where the data is stored or on the disk where the "TEMP" folder is located.

[asgmlexp-02]

Message: Cannot load model.

Please refresh the model list.

Cause: One of the models selected for export could not be loaded. Possibly this model was renamed or deleted by another user.

Action: Click on the button "Refresh" before selecting your models.

[asgmlexp-05]

Message: Model <*model name*>, Instance <*instance name*>:
The subprocess called does not exist.
The program will continue.
Do you want to be notified of further identical errors?

Cause: The specified instance refers to a model which cannot be loaded from database (does not exist).

Action: Correct the model reference.

[asgmlexp-06]

Message:
Model <*model name*>, instance <*instance name*>:
The referenced object does not exist.
The program will continue.
Do you want to be notified of further identical errors?

Cause:
The specified instance refers to a model inter-reference. This model no longer exists.

Action:
Correct the model reference.

[asgmlexp-07]

Message:
Model <*model name*>, instance <*instance name*>:
Invalid value for program and/or parameter.
The program will continue.
Do you want to be notified of further identical errors?

Cause:
The content of an attribute of type "Programcall" (e.g. External documentation) is invalid.

Action:
Correct the attribute.

[asgmlexp-08]

Message: Model <*model name*>, instance <*instance name*>:
Error when determining the target model.
The process will be continued.
Should messages of this type still be shown?

Cause: The specified instance has an inter model relation with no target instance.

Action: Correct the defective reference accordingly.

[asgmlexp-09]

Message: Error in the attribute "documentation configuration".

Invalid sorting mode: "<*sorting mode*>"

Cause: In the library attribute, an invalid value has been specified as sorting mode.

Action: Repair the error using the Administration Toolkit or contact your ADOxx administrator.

[asgmlexp-10]

Message: Error in the attribute "documentation configuration".

Invalid target directory for the referenced documents: "<*target directory*>"

Cause: The specified target directory does not exist and cannot be created.

Action: Verify you have write access rights in the target directory and if there is enough memory space in the target directory.

[asgmlexp-12]

Message: Error in the attribute "documentation configuration".

Invalid model type: "<*model type*>"

Cause: The specified model type does not correspond to any model type in the library.

Action: Verify the exact spelling of the model type. Correct the error using the Administration Toolkit or contact your ADOxx administrator.

[asgmlexp-13]

Message: Error in the attribute "documentation configuration".

No default LIBRARY element is defined in a SOURCE Block.

Cause: In the library attribute "documentation configuration", no standard settings are defined in an documentation export.

Action: Correct the error using the Administration Toolkit or contact your ADOxx administrator.

[asgmlexp-14]

Message: Error in the attribute "documentation configuration".

The value "<*value*>" in the LIBRARY element of the row <*no.*> is not a whole number.

Cause: The value specified in the library attribute "documentation configuration" not a whole number.

Action: Correct the error using the Administration Toolkit or contact your ADOxx administrator.

[asgmlexp-15]

Message: Model <*model name*>, instance <*instance name*>:

Create program and/or parameter: An external file was referenced but could not be found.

The process will be continued.

Should messages of this type still be shown?

Cause: In an attribute, either an external file has been referenced, which has been meanwhile deleted or moved or an internet address has been entered.

Action: Correct the reference in the model or copy the missing file to the specified directory. If you have specified an internet address in attributes click on the "No" button to suppress messages of this type in the future.

[asgmlexp-16]

Message: Program error: <*error code*>(<*error number*>).

Cause: A general program error has occurred.

Action: Note the error code and error number and contact your ADOxx administrator.

[asgmlexp-17]

Message: Error in the attribute "Documentation configuration".

No LIBRARY element has been defined for the model type "<*model type name*>".

Cause: No element "LIBRARY" has been defined in the library attribute "Documentation configuration" of an element EXPORT.

Action: Add an element LIBRARY for the specified model type.

[asgmlexp-18]

Message: File "<*file name*>" could not be copied.

Cause: The file could either not be found or not be created.

Action: Verify if the file in the specified directory exists, if there is enough free hard disk memory available and if you have the required access rights.

[asgmlexp-19]

Message: Graphic for the model "<*model name*>" could not be generated.

Cause: There is either not enough available memory space in the target directory or you do not have sufficient access rights to the target directory.

Action: Free up memory space or have your access rights extended. Alternatively select another target directory before the generation without the above-mentioned restrictions.

[asgmlexp-20]

Message: The specified path "<*path*>" is invalid for the referenced documents.

The process will be continued.

Cause: The path contains invalid characters or an invalid structure.

Action: Make sure during the entry of path names, that the total amount does not exceed 120 characters and that no more than seven directories are contained in the path. Moreover the characters **/ \ : * ? < >** and **|** are not allowed.

[asik-01]

Message: Failure while querying an attribute!

Cause: A fatal error has occurred from which the system cannot recover.

Action: If this error re-occurs when the simulation is run again, export your model to an ADL file and contact your ADOxx consultant.

[asik-02]

Message: Error when searching for instances of a model!

Cause: A fatal error has occurred from which the system cannot recover.

Action: If this error re-occurs when the simulation is run again, export your model to an ADL file and contact your ADOxx consultant.

[asik-03]

Message: Business process model has no start object!

Business process model: <*model name*>

Cause: The specified model does not contain a start object.

Action: Check the model and insert an appropriate start object.

[asik-04]

Message: Resource could not be found! Model: <*model name*>

Cause: No resource could be found when evaluating the given AQL command.

Action: Check the content of attribute "Selection" and check that your working environment model contains corresponding resources.

[asim1dlg-03]

Message: The input parameters are not defined correctly. A simulation cannot be executed. Please contact your ADOxx consultant.

Cause: The input parameters for the simulation are not consistent with the class and attribute definitions of the application library.

Action: Define the attribute "sim mapping" in the Administration Toolkit correctly or contact your ADOxx consultant.

[asim1end-01]

Message: The results values are too large to be transferred to the models.

Cause: The results values are larger than what the values area of the results attribute allows. This is probably due to the fact that the evaluation was carried out with unusually large input values (e.g. very large performance times and very large process quantities).

Action: Carry out the evaluation with smaller input values.

[asim2dlg-07]

Message: The protocol file contains an invalid path.

Cause: The protocol file's filename contains an invalid path.

Action: Correct the filename.

[asim2dlg-15]

Message: The application model "<*model name*>" is no longer valid. Please refresh the application models list.

Cause: You selected an application model which is no longer valid. One or more models of the application model could not be found.

Action: Refresh the model list by clicking on the button "Refresh" and repeat the action.

[asim2dlg-16]

Message: The application model "<*model name*>" does not exist. Please refresh the application models list.

Cause: You selected an application model which does not exist in the database any more. Presumably, it was deleted by another ADOxx user.

Action: Refresh the model list by clicking on the button "Refresh" and select a different application model.

[asim2dlg-17]

Message: The date of the simulation start has to be before the date of the results gathering start!

Cause: You entered a simulation start date later than the calculation start date into the start window of the workload analysis (fixed-time-period view).

Action: Define the simulation start date in such a way that it lies **before** the calculation (result) start date.

[asim2dlg-18]

Message: The date of the results gathering start has to be before the date of the results gathering end!

Cause: You entered a calculation start date later than the calculation end date into the start window of the workload analysis (fixed-time-period view).

Action: Define the calculation start date in such a way that it lies **before** the calculation (result gathering) end date.

[asim2dlg-21]

Message: No input parameters for the simulation are defined.

(The simulation cannot run.)

Please contact your ADOxx customer support

Cause: An error occurred while the sim mapping was read. The sim mapping is invalid or not defined.

Action: Define an input parameter configuration in the ADOxx administration Toolkit or contact your ADOxx administrator or customer support.

[asim2dlg-22]

Message: The protocol file "<*File Name*>" contains a directory, not a filename.

Cause: You entered the directory name listed instead of a file name.

Action: Correct your entry accordingly.

[asimmap-01]

Message: The sim mapping could not be interpreted correctly.

Cause: A non-identifiable error occurred during the interpretation of the sim mapping.

Action: Check the sim mapping in the library configuration in the ADOxx administration Toolkit and correct possible errors or contact your ADOxx administrator.

[asimmap-02]

Message: Syntax errors were found in the sim mapping.

Cause: There is a syntax error in the sim mapping. Perhaps a closing inverted comma is missing or not all of the keywords are written in capital letters.

Action: Check the sim mapping in the library configuration in the ADOxx administration Toolkit and correct possible errors or contact your ADOxx administrator.

© BOC Group 2008

[asimmap-03]

Message: The expression "<*indicator*>" (library attribute "sim mapping") contains errors.

Cause: An invalid indicator has been used in the sim mapping.

Action: Correct the specified indicator in the library attribute "sim mapping" or contact your ADOxx administrator.

[asimmap-04]

Message: The attribute "<*attribute name*>" of the sim mapping is corrupt.

Cause: The sim mapping is incomplete.

Action: Add the attribute listed to the sim mapping or contact your ADOxx administrator.

[asimmap-05]

Message: No entry parameters have been defined. Contact your "ADOxx" administrator to define entry parameter combinations in the library attribute "sim mapping".

Cause: Your ADOxx application library contains errors. The attribute "sim mapping" must be corrected.

Action: Contact your ADOxx administrator.

[asimobj-01]

Message: No (available) performer found! <*object class*>: <*object name*>

Model: <*model name*>

Cause: The AQL expression in the attribute "Performer" did not render an (available) performer.

Action: Check the AQL expression in the attribute "Performer" and make sure that the corresponding working environment model contains at least one suitable performer. In addition the performer has to be available (i.e. the attribute "Presence" has to be set to a value greater than zero).

[asimobj-02]

Message: No process calendar supplied for any process model of the application model.

Cause: None of the process models of the application model has a process calendar. Therefore, there is nothing to be simulated. A workload analysis cannot take place.

Action: If you wish to run a Workload Analysis, you have to define process calendars in the main processes to be simulated.

[asimobj-03]

Message: Some of your simulation results have exceeded the maximum integer value. The simulation will be aborted!

Cause: The simulation time is exceeding the period of time which ADOxx is able to represent.

Action: Check whether you entered the correct times everywhere in your business process models. If the times are okay, try it with a smaller number of runs. Should this be impossible, contact your ADOxx consultant.

[asimobj-04]

Message: You did not enter any times of presence for the performers in your working environment model. Model: <*model name*>

Cause: All performers of your working environment model are regarded as permanently absent. Therefore, there is nothing to simulate.

Action: Enter times of presence for the performers of your working environment model via the performers' calendars.

[asimobj-05]

Message: Error reading a performer calendar.

Cause: A runtime error occurred while a performer calendar was read.

Action: If the error occurs again after re-starting the simulation, export the model into an ADL file and inform your ADOxx consultant.

[asimobj-06]

Message: No further times of presence exist for any performer of your working environment. The simulation will be aborted.

Cause: According to the calendars of the performers in your working environment, there are no times of presence after the present time. Since there is nothing left to be simulated, the simulation is aborted.

Action: Either enter further times of presence which exceed the current point of time in the simulation or shorten the simulation time by decreasing the number of runs or the number of oscillation phases.

[asimobj-07]

Message: Illegal interruption of a synchronous cooperative activity!

Cause: During the simulation is has been tried to illegally interrupt a synchronous cooperative activity. This is not possible.

Action: Correct the model to be simulated.

[asimtext-01]

Message: The library attribute "SimText" is not defined.

Cause: The library attribute "SimText" is not defined.

Action: Define the library attribute "SimText" in the Administration Toolkit. If necessary, contact your ADOxx administrator.

[asimtext-02]

Message: The library attribute "SimText" is not defined.

The expression "<*Expression*>" is incomplete or undefined.

Cause: The expression listed could not be found in the library attribute "Simtext". The library attribute "Simtext" of the BP library either contains a value which is incorrect or incomplete or is not defined.

Action: Check the library attribute "Simtext" in the Administration Toolkit and correct the definition of the expression listed. If necessary, contact your ADOxx administrator.

[asrchmod-01]

Message: Error while copying: Not all values could be pasted!

Cause: An error has occurred during the paste in the ADOxx browser. Not all values to be inserted have been assigned.

Action: Check if the area to be pasted is not too large and if the attribute types correspond. Check if it is possible to paste in the area you have selected. An insertion is not possible if the model, which model attribute should be changed by the insertion, is not loaded or if it is loaded only with read access.

[asrchmod-02]

Message: Error while copying: No value could be inserted!

Cause: During the pasting in the ADOxx browser, none of the value assigned could be pasted.

Action: Check if the area to be pasted is not too large and if the attribute types correspond. Check if the pasting is allowed within the area you have selected. An insertion is not possible if the model, which model attribute should be changed by the insertion, is not loaded or if it is loaded only with read access.

[asrchtlb-01]

Message: Saving the contents of the model select box failed!

Cause: An illegal file name was entered in the "Save as" window of the model select box.

Action: Please enter a file name that is permitted and make sure that there is sufficient storage space on the data medium.

[astarth-01]

Message: The start of the process<*command rows*> went wrong!

Cause: The attempt to start the specified program went wrong.

Action: Contact you system administrator. The program is either to be installed or system settings must be adjusted.

[astdcfg-01]

Message: Error when creating the *<%mm>*:
The class "*<class name>*" could not be created!
Super class: "*<super class name>*"
Error code: *<error code>*
ADOxx will be closed!

Cause: The configuration of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[astdcfg-02]

Message: Error when creating the *<%mm>*:
The relation class "*<relation class name>*" could not be created!
From class: "*<starting class name>*"
To class: "*<target class name>*"
Error code: *<error code>*
ADOxx will be closed!

Cause: The configuration of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[astdcfg-03]

Message: Error when creating the *<%mm>*:
The attribute "*<attribute name>*" could not be created!
Owner: "*<owner name>*"
Error code: *<error code>*
ADOxx will be closed!

Cause: The configuration of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[astdcfg-04]

Message: Error when creating the *<%mm>*:
No value could be assigned to the attribute "*<attribute name>*"!
Owner: "*<owner name>*"
Value: "*<value>*"
Error code: *<error code>*

ADOxx will be closed!

Cause: The configuration of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[astdcfg-05]

Message: Error when creating the <*%mm*>:
No value could be assigned to the facet "<*facet name*>"!
Attribute: "<*attribute name*>"
Value: "<*value*>"
Error code: <*error code*>
ADOxx will be closed!

Cause: The configuration of your ADOxx database contains errors.

Action: Contact your ADOxx administrator.

[asubsrch-01]

Message: The Business process model <*model name*> does not exist.

Cause: The specified business process model does not exist.

Action: Enter the name of an existing model.

[asuibrw-01]

Message: Select the attributes which should be displayed in the diagram!

Cause: You have not selected any attributes for the diagram.

Action: Select the attributes which should be displayed in the diagram.

[avedbox-01]

Message: Figures chain "<*Text*>" not found!

Cause: The figures chain searched for does not appear in the text field.

Action: The search has possibly not been carried out from the beginning of the text. If necessary place the cursor at the beginning of the text field before starting the search.

[averdate-01]

Message: Internal error: the version handler has not been initialised.

Cause: An internal error has occurred.

Action: Contact your ADOxx administrator.

[averdate-02]

Message: A syntactical error has occurred when parsing the expression:

<LEO expression>

Cause: The value of the library attribute "versioning format" contains errors. The specified LEO expression contains syntactical errors. Pay attention to the appropriate [aleo] error message.

Action: Correct the value of the library attribute "versioning format" and pay attention to the specified [aleo] error message or contact your ADOxx administrator.

[averdate-04]

Message: The key word "VERSIONING" has not been found.

Cause: The first key word of the LEO expressions is not "VERSIONING".

Action: Correct the value of the library attribute "versioning format" by inserting the key word "VERSIONING" at the first position or contact your ADOxx administrator.

[averdate-05]

Message: An invalid key word has been used.

Cause: The LEO expression contains an invalid key word.

Action: Correct the value of the library attribute "versioning format" by deleting all invalid key words or contact your ADOxx administrator. Valid key words are "VERSIONING", "START_DATE", "TEXT_FIELD", "DAY_FIELD", "MONTH_FIELD" and "YEAR_FIELD".

[averdate-06]

Message: The start date can only be defined once.

Cause: The LEO expression contains the key word "START_DATE" more than once.

Action: Correct the value of the library attribute "versioning format" by deleting all unnecessary "START_DATE" key words or contact your ADOxx administrator.

[averdate-07]

Message: The day field can only be defined once.

Cause: The LEO expression contains the key word "DAY_FIELD" more than once.

Action: Correct the value of the library attribute "versioning format" by deleting all unnecessary "DAY_FIELD" key words or contact your ADOxx administrator.

[averdate-08]

Message: The month field can only be defined once.

Cause: The LEO expression contains the key word "MONTH_FIELD" more than once.

Action: Correct the value of the library attribute "versioning format", by removing all unnecessary "MONTH_FIELD" key words or contact your ADOxx administrator.

[averdate-09]

Message: The year field can only be defined once.

Cause: The LEO expression contains the key word "YEAR_FIELD" more than once.

Action: Correct the value of the library attribute "versioning format", by removing all unnecessary "YEAR_FIELD" key words or contact your ADOxx administrator.

[averdate-10]

Message: The value of the day attribute in the start date is too small.

It is only allowed within the area <*1*> to <*31*>.

Cause: An error has occurred in the key word "START_DATE": The value of the attribute "day" is too small.

Action: Correct the value of the library attribute "versioning format", by specifying the attribute "day" of the key word "START_DATE" with a valid value between 1 and 31 or contact your ADOxx administrator.

[averdate-11]

Message: The value of the day attribute in the start date is too high.

It is only allowed within the area <*1*> to <*31*>.

Cause: An error has occurred in the key word "START_DATE": The value of the attribute "day" is too high.

Action: Correct the value of the library attribute "versioning format", by specifying the attribute "day" of the key word "START_DATE" with a valid value between 1 and 31 or contact your ADOxx administrator.

[averdate-12]

Message: The value of the month attribute in the start date is too small.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "START_DATE": the value of the attribute "month" is too small.

Action: Correct the value of the library attribute "versioning format" by specifying the attribute "month" of the key word "START_DATE" with a valid value between 1 and 12 or contact your ADOxx administrator.

[averdate-13]

Message: The value of the month attribute in the start date is too high.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "START_DATE": the value of the attribute "month" is too high

Action: Correct the value of the library attribute "versioning format", by specifying the attribute "month" of the key word "START_DATE" with a valid value between 1 and 12 or contact your ADOxx administrator.

[averdate-14]

Message: The value of the year attribute in the start date is too small.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "START_DATE": the value of the attribute "year" is too small.

Action: Correct the value of the library attribute "versioning format", by specifying the attribute "year" of the key word "START_DATE" with a valid value between 0 and 9999 or contact your ADOxx administrator.

[averdate-15]

Message: The value of the year attribute in the start date is too high.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "START_DATE": the value of the attribute "year" is too high.

Action: Correct the value of the library attribute "versioning format", by specifying the attribute "year" of the key word "START_DATE" with a valid value between 0 and 9999 or contact your ADOxx administrator.

[averdate-16]

Message: The minimum value of the day field is greater than the maximum value.

Cause: An error has occurred in the key word "DAY_FIELD": the value of the attribute "minimum" is greater than the value of the attribute "maximum".

Action: Correct the value of the library attribute "versioning format" by specifying in key word "DAY_FIELD" the value of the attribute "minimum" with a value, which is smaller than the value of the attribute "maximum" or contact your ADOxx administrator.

[averdate-17]

Message: The minimum value of the month field is greater than the maximum value.

Cause: An error has occurred in the key word "MONTH_FIELD": the value of the attribute "minimum" is greater than the value of the attribute "maximum".

Action: Correct the value of the library attribute "versioning format", by specifying in key word "MONTH_FIELD" the value of the attribute "minimum" with a value, which is smaller than the value of the attribute "maximum" or contact your ADOxx administrator.

[averdate-18]

Message: The minimum value of the year field is greater than the maximum value.

Cause: An error has occurred in the key word "YEAR_FIELD": the value of the attribute "minimum" is greater than the value of the attribute "maximum".

Action: Correct the value of the library attribute "versioning format", by specifying in key word "YEAR_FIELD" the value of the attribute "minimum" with a value, which is smaller than the value of the attribute "maximum" or contact your ADOxx administrator.

[averdate-19]

Message: The default value of the day field is too small.

It is only allowed within the area *<1>* to *<31>*.

Cause: An error has occurred in the key word "DAY_FIELD": the value of the attribute "default" is either not in the area 1 to 31 or the value is smaller than the value of the attribute "minimum".

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "default" in the key word "DAY_FIELD" with a value, which is both between the values of the attributes "minimum" and "maximum" and between 1 and 31 or contact your ADOxx administrator.

[averdate-20]

Message: The default value of the day field is too high.

It is only allowed within the area *<1>* to *<31>*.

Cause: An error has occurred in the key word "DAY_FIELD": the value of the attribute "default" is either not in the area 1 to 31 or the value is higher than the value of the attribute "maximum".

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "default" in the key word "DAY_FIELD" with a value, which is both between the values of the attributes "minimum" and "maximum" and between 1 and 31 or contact your ADOxx administrator.

[averdate-21]

Message: The minimum value of the day field is too small.

It is only allowed within the area *<1>* to *<31>*.

Cause: An error has occurred in the key word "DAY_FIELD": the value of the attribute "minimum" is too small.

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "minimum" in the key word "DAY_FIELD" with a value between 1 and 31 or contact your ADOxx administrator.

[averdate-22]

Message: The minimum value of the day field is too high.

It is only allowed within the area <*1*> to <*31*>.

Cause: An error has occurred in the key word "DAY_FIELD": the value of the attribute "minimum" is too high.

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "minimum" in the key word "DAY_FIELD" with a value between 1 and 31 or contact your ADOxx administrator.

[averdate-23]

Message: The maximum value of the day field is too small.

It is only allowed within the area <*1*> to <*31*>.

Cause: An error has occurred in the key word "DAY_FIELD": the value of the attribute "maximum" is too small.

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "maximum" in the key word "DAY_FIELD" with a value between 1 and 31 or contact your ADOxx administrator.

[averdate-24]

Message: The maximum value of the day field is too high.

It is only allowed within the area <*1*> to <*31*>.

Cause: An error has occurred in the key word "DAY_FIELD": the value of the attribute "maximum" is too high.

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "maximum" in the key word "DAY_FIELD" with a value between 1 and 31 or contact your ADOxx administrator.

[averdate-25]

Message: The default value of the month field is too small.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "MONTH_FIELD": the value of the attribute "default" is either not in the area 1 to 12 or the value is smaller than the value of the attribute "minimum".

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "default" in the key word "MONTH_FIELD" with a value, which is both between the values of the attributes "minimum" and "maximum" and between 1 and 12 or contact your ADOxx administrator.

[averdate-26]

Message: The default value of the month field is too high.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "MONTH_FIELD": the value of the attribute "default" is either not in the area 1 to 12 or the value is greater than the value of the attribute "maximum".

Action: Correct the value of the library attribute "versioning format" by specifying the value of the attribute "default" in the key word "MONTH_FIELD" with a value, which is both between the values of the attributes "minimum" and "maximum" and between 1 and 12 or contact your ADOxx administrator.

[averdate-27]

Message: The minimum value of the month field is too small.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "MONTH_FIELD": the value of the attribute "minimum" is too small.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "MONTH_FIELD" the value of the attribute "minimum" with a value between 1 and 12 or contact your ADOxx administrator.

[averdate-28]

Message: The minimum value of the month field is too high.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "MONTH_FIELD": the value of the attribute "minimum" is too high.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "MONTH_FIELD" the value of the attribute "minimum" with a value between 1 and 12 or contact your ADOxx administrator.

[averdate-29]

Message: The maximum value of the month field is too small.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "MONTH_FIELD": the value of the attribute "maximum" is too small.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "MONTH_FIELD" the value of the attribute "maximum" with a value between 1 and 12 or contact your ADOxx administrator.

[averdate-30]

Message: The maximum value of the month field is too high.

It is only allowed within the area <*1*> to <*12*>.

Cause: An error has occurred in the key word "MONTH_FIELD": the value of the attribute "maximum" is too high.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "MONTH_FIELD" the value of the attribute "maximum" with a value between 1 and 12 or contact your ADOxx administrator.

[averdate-31]

Message: The default value of the year field is too small.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "YEAR_FIELD": the value of the attribute "default" is either not within the area 0 to 9999 or the value is smaller than the value of the attribute "minimum".

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "YEAR_FIELD" the value of the attribute "default" with a value which is both between the values of the attributes "minimum" and "maximum" and between 0 and 9999 or contact your ADOxx administrator.

[averdate-32]

Message: The default value of the year field is too high.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "YEAR_FIELD": the value of the attribute "default" is either not within the area 0 to 9999 or the value is greater than the value of the attribute "maximum".

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "YEAR_FIELD" the value of the attribute "default" with a value which is both between the values of the attributes "minimum" and "maximum" and between 0 and 9999 or contact your ADOxx administrator.

[averdate-33]

Message: The minimum value of the year field is too small.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "YEAR_FIELD": the value of the attribute "minimum" is too small.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "YEAR_FIELD" the value of the attribute "minimum" with a value between 0 and 9999 or contact your ADOxx administrator.

[averdate-34]

Message: The minimum value of the year field is too high.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "YEAR_FIELD": the value of the attribute "minimum" is too high.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "YEAR_FIELD" the value of the attribute "minimum" with a value between 0 and 9999 or contact your ADOxx administrator.

[averdate-35]

Message: The maximum value of the year field is too small.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "YEAR_FIELD": the value of the attribute "maximum" is too small.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "YEAR_FIELD" the value of the attribute "maximum" with a value between 0 and 9999 or contact your ADOxx administrator.

[averdate-36]

Message: The maximum value of the year field is too high.

It is only allowed within the area <*0*> to <*9999*>.

Cause: An error has occurred in the key word "YEAR_FIELD": the value of the attribute "maximum" is too high.

Action: Correct the value of the library attribute "versioning format", by specifying in the key word "YEAR_FIELD" the value of the attribute "maximum" with a value between 0 and 9999 or contact your ADOxx administrator.

[averdate-37]

Message: The value of the day field is too small.

It is only allowed within the area <*1*> to <*31*>.

Cause: The day of the specified date is either smaller than the value of the attribute "minimum" of the key word "DAY_FIELD" or less than 1.

Action: Change the date, so that the day is between Minimum and Maximum or between 1 and 31.

[averdate-38]

Message: The value of the day field is too high.

It is only allowed within the area <*1*> to <*31*>.

Cause: The day of the specified date is either greater than the value of the attribute "maximum" of the key word "DAY_FIELD" or greater than 1.

Action: Change the date, so that the day is between Minimum and Maximum or between 1 and 31.

[averdate-39]

Message: The value of the month field is too small.

It is only allowed within the area <*1*> to <*12*>.

Cause: The day of the specified date is either smaller than the value of the attribute "minimum" of the key word "MONTH_FIELD" or smaller than 1.

Action: Change the date, so that the month is between Minimum and Maximum or between 1 and 12.

[averdate-40]

Message: The value of the month field is too high.

It is only allowed within the area <*1*> to <*12*>.

Cause: The day of the specified date is either greater than the value of the attribute "maximum" of the key word "MONTH_FIELD" or greater than 1.

Action: Change the date, so that the month is between Minimum and Maximum or between 1 and 12.

[averdate-41]

Message: The value of the year field is too small.

It is only allowed within the area <*0*> to <*9999*>.

Cause: The day of the specified date is either smaller than the value of the attribute "minimum" of the key word "YEAR_FIELD" or smaller than 0.

Action: Change the date, so that the year is between Minimum and Maximum or between 0 and 9999.

[averdate-42]

Message: The value of the year field is too high.

It is only allowed within the area <*0*> to <*9999*>.

Cause: The day of the specified date is either greater than the value of the attribute "maximum" of the key word "YEAR_FIELD" or greater than 9999.

Action: Change the date, so that the year is between Minimum and Maximum or between 0 and 9999.

[averdate-43]

Message: The internal display of the date is not correct.

Cause: An internal error has occurred: the format of the date is invalid.

Action: Contact your ADOxx administrator.

[averdate-44]

Message: The internal display of the date contains a day although no day field is defined.

Cause: An internal error has occurred: the format of the date is invalid.

Action: Contact your ADOxx administrator.

[averdate-45]

Message: The internal display of the date contains a month although no month field is defined.

Cause: An internal error has occurred: the format of the date is invalid.

Action: Contact your ADOxx administrator.

[averdate-46]

Message: The internal display of the date contains a year although no year field is defined.

Cause: An internal error has occurred: the format of the date is invalid.

Action: Contact your ADOxx administrator.

[averdate-47]

Message: The specified date cannot be increased.

Cause: The specified date is the last possible date.

Action: Use an earlier date since this is the last possible date or contact your ADOxx administrator.

[averdate-48]

Message: Error when parsing the string:

The content of the text field does not correspond with the defined content.

Cause: A string has been found which does not correspond to the one in the library attribute "versioning format".

Action: Adapt the string that it corresponds with the format defined in the library attribute "versioning format". Pay attention to the order! If necessary, contact your ADOxx administrator.

[averdate-49]

Message: Error when parsing the string:

The month name is not valid.

Cause: The name of a month is expected but could not be found.

Action: Possible month names are "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November" and "December".

[averdate-50]

Message: Error when parsing the string:

The sting contains additional characters.

Cause: The string contains excess characters.

Action: Adapt the string so that it corresponds with the format defined in the library attribute "versioning format" or contact your ADOxx administrator.

[averdate-51]

Message: Error when parsing the string:

The combination of day and month is invalid.

Cause: You have attempted to use an invalid combination of day and month.

Action: Change the string so that it only contains valid combinations of day and month, invalid combinations include "29" and "2" or "31" and "4".

[averdate-52]

Message: Error when parsing the string:

No number has been found at the current place.

Cause: A number is expected in the string but could not be found.

Action: Adapt the string so that it corresponds with the format defined in the library attribute "versioning format" or contact your ADOxx administrator.

[averdate-53]

Message: Error when parsing the string:

The number found is too small.

Cause: A number found in the string is too small.

Action: Adapt the string so that it corresponds with the format defined in the library attribute "versioning format" or contact your ADOxx administrator.

[averdate-54]

Message: Error when parsing the string:

The number found is too high.

Cause: A number found in the string is too high.

Action: Adapt the string so that it corresponds with the format defined in the library attribute "versioning format" or contact your ADOxx administrator.

[averdate-55]

Message: Error while parsing string:

Between two dynamic values at least one TEXT field must exist.

Cause: When defining the library attribute "Versioning format" two dynamic fields (for day, month or year) have been entered consecutively.

Action: Change the definition by inserting a text field between the two dynamic fields or contact your ADOxx administrator.

[awrjpeg-01]

Message: <*Error-specific Message*>

Cause: When generating the graphic file, an unexpected error was encountered.

Action: Retry.

[awrjpeg-02]

Message: Too little space!

Cause: Not enough main memory is available.

Action: Close a number of applications and try again.

[awrjpeg-03]

Message: Error on opening file "*filename*".

Cause: The file specified could not be opened with write-access.

Action: Ensure that enough space exists in the directory to which you are writing and that you have access to update that directory.

[awrjpeg-04]

Message: Error while closing file.

Cause: An error occurred while closing the graphics file.

Action: Ensure you have enough space in the directory to which you are writing. If this directory is on the network, ensure that you are still connected. Try again.

[axml-01]

Message: Invalid message: "*message*"

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-02]

Message: Error when initialising a new parser!! A file has already been parsed.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-03]

Message: Error in the XML interface! No parser has been initialised.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-04]

Message: The file "*file name*" does not exist!

Cause: The specified file is required but could not be found.

Action: Control if the specified file actually exists.

[axml-05]

Message: Error in the function XML_ADD_CALLBACK! The type "*type*" does not correspond with any valid type.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-06]

Message: Error in the function XML_SET_SCRIPT! The procedure for the XML import has already been initialised.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-08]

Message: Error in the function "*function name*"! A required parameter is missing.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-09]

Message: Error in the function XML_ADD_CALLBACK! The procedure "*procedure name*" does not exist.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-10]

Message: Error in the function XML_ADD_CALLBACK! A procedure already exists for the element "*element name*".

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-11]

Message: An error has occurred during the initialising of the XML parser: *Error message*

Cause: Pay attention to the error message for more information.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-12]

Message: An error has occurred when parsing the XML file.

Cause: The XML file does not respond to the prescribed syntax.

Action: Control the XML file to be read in.

[axml-13]

Message: An error has occurred when parsing the XML file. File *file name*, row *row number*, column *column number*

Cause: The XML file does not respond to the prescribed syntax.

Action: Control the XML file to be read in.

[axml-15]

Message: The element with the ID *ID* could not be stored in the memory.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-16]

Message: The element "*element name*" does not contain any attribute with the name "*attribute name*".

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-17]

Message: The element "*element name*" is already kept in the memory under the ID *ID*.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-18]

Message: The format of the search string "*search string*" is invalid. Valid format: "name=value"

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-19]

Message: The searched element could not be kept in the memory.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-20]

Message: The element "*element name*" has no parent node as it is the root element.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-21]

Message: The element has no child node.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-22]

Message: The parse has been aborted.

Cause: The XML import has been aborted.

Action: Pay attention to the preceding error messages.

[axml-23]

Message: Error in the function XML_OPEN! A file has already been opened to be written.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-24]

Message: Error in the function XML_OPEN! The file "*file name*" already exists.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-25]

Message: Error in the function XML_OPEN! The file "*file name*" could not be written.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-26]

Message: Error in the function XML_WRITEATTRIBUTE! Attributes can only be specified in start nodes.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-27]

Message: Error in the function XML_OPEN! The value "*code page*" does not correspond with any valid code page.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-28]

Message: Error in the XML interface! No file has been opened to be written.

Cause: An invalid message has been sent to the documentation message port.

Action: There is an error in an AdoScript . Contact your ADOxx administrator.

[axml-29]

Message: Error in the XML interface! There is no more free memory space in the target drive.

Cause: During the XML export, it was not possible to write in the file, since there is not enough memory space available in the target data carrier.

Action: Verify the memory space in the target drive. Select a target driver with enough memory space and repeat the export.

[hacevalacq-02]

Message: The acquisition tables are not consistent. The consolidation will be aborted.

Cause: Corresponding processes have been found but they are inconsistencies in the values of the variables, names of objects or sequences of objects .

Action: Verify the processes to be consolidated and repair the inconsistencies.

[hacevalacq-03]

Message: The selected file is already contained in the file.

Cause: You would like to add a file for the consolidation but it has already been added.

Action: Add another file.

[hacevalacq-04]

Message: The selected file does not correspond with the options set for the current scenario.

Cause: You would like to add a file for the consolidation which does not correspond with the settings of the selected HOMER scenario.

Action: Select a file for the consolidation which corresponds to the current HOMER settings or change the scenario settings according to the file you wish to use.

[hacexp-01]

Message: You have not specified a model name. The generation will be aborted. The model name should be in the below-specified cell. Cell:<cell name>

Cause: There is no model name in the specified cell.

Action: Enter the model name in the specified cell.

[hacexp-05]

Message: A missing attribute has been found. If you want you can abort the generation. Defective attribute:<Attribute name> Object:<Object name>

Cause: You have decided to abort the generation, if a missing attribute is found.

Action: Fill the specified attribute and repeat the process.

[hacexp-06]

Message: A wrong number format has been found. The generation will be aborted. Defective attribute:<attribute name> Object:<object name>

Cause: A number value is in the specified attribute of the specified object, which does not correspond with the attribute settings for this attribute.

Action: Correct the number value according to the selected attribute settings.

[hacexp-08]

Message: You have created the same object twice. Rename or remove one of the objects. Object:<object name>

Cause: You have created an object twice in your table.

Action: Delete one of the objects or change the name of one of these objects.

[hacexp-10]

Message: An internal error has occurred. Try again. Should the error appear again, note the error value and contact your ADOxx administrator.

Cause: An unexpected error has occurred.

Action: Repeat the action. Should the error appear again, note the error value and contact your ADOxx administrator.

[hacexp-14]

Message: An unknown short description has been found in the mentioned cell below. You can change the short description of the object to an already defined one or redefine it. Cell:<cell name>

Cause: A short description of class, which is not defined, has been found.

Action: Change the short description of class in the specified cell to an already available short description of class or define a new class with the available short description of class.

[hacexp-17]

Message: An unknown short description has been found in the mentioned cell below. You can change the short description of the object to an already defined one or redefine it. Cell:<cell name>

Cause: A short description of class, which is not defined, has been found.

Action: Change the short description of class in the specified cell to an already available short description of class or define a new class with the available short description of class.

[hacfile-01]

Message: The file cannot be opened. Check if the file corresponds with the selected options.

Cause: You have attempted to open a file which cannot be opened. Verify if you have full access to this file and if the file is not blocked or being used by another user.

Action: Once you have checked the causes, repeat the process.

[hacimp-05]

Message: The ADL file does not correspond with the HOMER settings and so cannot be imported. The import will be aborted. Verify if the library type and if necessary the model type correspond with the settings.

Cause: You have selected an ADL file for the import, which library properties do not correspond with the HOMER settings.

Action: Select an ADL file which conforms to the HOMER settings, an INI file of INI administration which is adapted to the selected ADL file or adjust the current INI settings to the ADL file.

[hacini-04]

Message: An unexpected error has occurred. Save your changes and try it again.

Cause: An unexpected action of the user or of another program has led to an error.

Action: Close HOMER and re-start it. Should the error appear again, note the error code and description and contact your ADOxx administrator.

[hacini-06]

Message: There is already an attribute in the column you have selected. Select another column.

Cause: You have selected a column for the current attribute which is already assigned.

Action: Select an available column. The furthest right free column is specified in the field 'next free column'.

[hacini-14]

Message: The specified cell is already used. Select another cell.

Cause: You have selected a cell which is already assigned.

Action: Select another cell or abort the process.

[hacini-15]

Message: The below-mentioned class already exists. Class: <class description>

Cause: The entered class description already exists.

Action: Change the class description or abort the process.

[hacini-16]

Message: The short description mentioned below already exists. Change either the short description of this class or before you repeat this process the class mentioned below. Class short description: <class short description>Class: <Class description>

Cause: The class short description you have selected already exists for the specifed class.

Action: Change the current short description or the description of one of the specified classes.

[hacini-17]

Message: The specified column is already used for the object name. Column: <Column name>

Cause: The column you have selected will be used for the object name.

Action: Select another column or abort the process.

[hacini-18]

Message: The specified column is already used for the short description of the objects. Column: <column name>

Cause: The column you have selected is used for the class short description.

Action: Select another column or abort the process.

[hacini-24]

Message: The specified column is already used for the attribute profile version. Column:<column name>

Cause: You have attempted to place an attribute in a column, which is already used for the attribute profile version.

Action: Select another free column or abort the process.

[hacini-26]

Message: The specified column is already used for the short description of the ADOxx tables.

Cause: You have attempted to place an attribute in a column, which is already used for the ADOxx tables.

Action: Select another free column or abort the process.

[hacini-27]

Message: All fields must be filled. You have forgotten to fill the following field.

Cause: You have attempted, to close the dialogue without having filled all required fields.

Action: Fill the specified field and continue.

[hacini-28]

Message: There is already an attribute with this name for this class. Change the name of the attribute.

Cause: An attribute with the selected name has already been created for the current class.

Action: Change the name of this attribute or of the already created attribute.

[hacini-29]

Message: The ADOxx record mentioned below already exists.

Cause: An ADOxx record with the name mentioned already exists.

Action: Change either the short description of this ADOxx record or the mentioned ADOxx record, before you repeat the process.

[hacini-30]

Message: The below mentioned short description already exists. Change either the short description of this ADOxx record or, before you repeat the process, the below mentioned ADOxx record.

Cause: The short description you have selected already exists.

Action: Change either the short description of this ADOxx record or, before you repeat the process, the mentioned ADOxx record

[hacini-31]

Message: The cell entered is already used for the model name.

Cause: You have attempted to use a cell for the current model attribute, which is already used for the model name.

Action: Select another cell.

[hacini-32]

Message: The cell entered is already used for the version number.

Cause: You have attempted to use a cell for the current model attribute, which is already used for the version number.

Action: Select another cell.

[hacini-33]

Message: The cell entered is already used for the attribute below.

Cause: You have attempted to use a cell for the current model attribute, which is already used for the mentioned attribute.

Action: Select another cell.

[hacinit-01]

Message: The below specified INI field is not compatible with the HOMER version used. Import the file to convert the data.

Cause: You have attempted to use HOMER with an INI file, which is not compatible with the current version.

Action: Use the INI file, which is delivered with the version used and import the old INI file using the scenario manager.

[svxwins-01]

Message: No file name has been specified!

Cause: A file name must be specified for the selected action.

Action: Enter a valid file name.

[svxwins-02]

Message: The file does not exist!

Cause: You have attempted to open a file to read which does not exist.

Action: Enter a valid file name.

[svxwins-03]

Message: An error has occurred during the storage!

Cause: An error has occurred during the storage of the file.

Action: There may be not enough space on the data medium or the specified path is not valid. Check the data medium or the path.

Part V
Appendix

1. Starting ADOxx

The login to ADOxx depends on the settings of the ADOxx database:

- exclusively as ADOxx user (see chap. 1.1, p. 722) or
- alternatively as ADOxx user or as system user with Single-Sign-on functionality (see chap. 1.2, p. 724).

The login to ADOxx WebService is described in chapter "Start ADOxx WebService" (see chap. 1.3, p. 726).

1.1 Start ADOxx

Start the Modelling Toolkit by selecting the program from the "Start" menu.

Figure 448: ADOxx login

Enter your user name, your password, the name of the ADOxx database with which you intend to work and click on the button "Login".

> **Hint:** Depending on the settings of your ADOxx installation it might be possible to select the name of the database from a list of possible values in a drop-down list. The availability of such a list is indicated by the symbol at the end of the input field.

After a successful login the Modelling Toolkit is launched.

As an alternative, the ADOxx application can be started from a **command line window** (Windows command line prompt). For that, change to the ADOxx installation directory.

Enter "`areena`" to start the ADOxx Modelling Toolkit.

The following parameters can be used:

-u<*User name*> replace <*User name*> with the name of the ADOxx/system user.

-p<*Password*> replace <*Password*> with the password of the ADOxx/system user.

-d<*DB-Name*> replace <*DB name*> with the name of the ADOxx database.

-lang<*Language code*> replace <*Language code*> either with the Windows Language ID or the ADOxx code of the desired language (see chap. 8., p. 738) for the user interface.

Note: For being able to use -lang<*Language code*>, your ADOxx installation must provide this language. If this parameter is not used, ADOxx will start in the basic language "German".

-f<*Font type name*> replace <*Font type name*> with the name of the font to be used inADOxx.

The default font for Windows is "MS Sans Serif".

-h<*Font size*> replace <*Font size*> with a number between 4 and 48 for the size of the font in ADOxx.

The default font size is 12pt.

-x causes that the ADOxx login screen will be shown despite the fact that parameters for the user name (**-u**), the password (**-p**) and the database name (**-d**) are shown.

-winlogin commands to log in as the currently active system user.

Note: To use the parameter -winlogin the Single-Sign-on functionality has to be installed.

-no_login_screen determines that in the case of an unsuccessful login the login screen will not be displayed after the error message.

Note: An unsuccessful login means that all the parameters for the user name (**-u**), the password (**-p**) and the database name (**-d**) have been stated but at least one of the parameters was incorrect.

-no_printer_warning causes that the error message "No default printer installed" will not be displayed in the case of no default printer being installed.

Hint: If you pass on the user name (-u), the password (-p) and the database name (-d) correctly as parameters, the ADOxx login screen will not be shown.

ATTENTION: Should the user name, password, database name, or font type names contain spaces, the entire parameter has to be put within quotation marks (e.g. "-uUser 1" for the user "User 1").

ATTENTION: User name and password are case sensitive!

Examples:

```
-uAdmin -ppassword -dadoxxdb
```

logs in the user "Admin" with the password "password" to the ADOxx database "adoxxdb".

```
-uAdmin -ppassword -dadoxxdb -x
```

shows the ADOxx login screen (see fig. 448) where the field "User name" contains the value "Admin", the field "Password" the encrypted string "password" and the field "Database name" the value "adoxxdb".

```
"-uUser 1" "-pUser 1" -ddatabase -h14
```

logs in the user "User 1" with the password "User 1" to the ADOxx database "database" using a font size of 14 pt in the ADOxx application window.

1.2 Start ADOxx (Single-Sign-on)

Hint: The functionality for Single-Sign-on, i.e. the login to ADOxx as a system user without providing your username and password is only possible with Windows XP/2000/NT 4.0 and has to be enabled for the respective ADOxx database. For further information please contact your ADOxx consultant.

Start the Modelling Toolkit by selecting the program from the "Start" menu.

Figure 449: ADOxx login (Single-Sign-on)

To log into ADOxx you can choose from the following options:

1. as the **current system user**,
2. as a **different system user** or
3. as ordinary **ADOxx user**.

As a **current system user** enter the name of the ADOxx database you want to work with, select your desired user interface language and the option "as system user" and click on the button "Login" (it is not necessary to provide a user name or password).

As **different system user** enter your system user name, your system password and the name of the ADOxx database you want to work with. Then choose your desired user interface language, select the option "as system user" and click on the button "Login".

Hint: The login as a different system user is only possible within the current domain.

Hint: To log-in as a system user you will have to arrange for your system user information to be imported into the ADOxx user management. Please contact your ADOxx administrator if necessary.

As **ADOxx user** enter your ADOxx user name, your ADOxx password and the name of the ADOxx database you want to work with. Choose a user interface language, select the option "as ADOxx user" and click on the button "Login".

After a successful login the Modelling Toolkit is started.

As an alternative, the ADOxx application can be started from a **command line window** (command line prompt when using Windows). For that, change to the ADOxx installation directory.

Enter "`areena`" to start the ADOxx Modelling Toolkit.

The following parameters can be used:

-u<*User name*>	replace <*User name*> with the name of the ADOxx/system user.
-p<*Password*>	replace <*Password*> with the password of the ADOxx/system user.
-d<*DB-Name*>	replace <*DB name*> with the name of the ADOxx database.
-lang<*Language code*>	replace <*Language code*> either with the Windows Language ID or the ADOxx code of the desired language (see chap. 8., p. 738) for the user interface. *Note:* For being able to use -lang<*Language code*>, your ADOxx installation must provide this language. If this parameter is not used, ADOxx will start in the basic language "German".
-f<*Font type name*>	replace <*Font type name*> with the name of the font to be used inADOxx. The default font for Windows is "MS Sans Serif".
-h<*Font size*>	replace <*Font size*> with a number between 4 and 48 for the size of the font in ADOxx. The default font size is 12pt.
-x	causes that the ADOxx login screen will be shown despite the fact that parameters for the user name (**-u**), the password (**-p**) and the database name (**-d**) are shown.
-winlogin	commands to log in as the currently active system user. *Note:* To use the parameter -winlogin the Single-Sign-on functionality has to be installed.
-no_login_screen	determines that in the case of an unsuccessful login the login screen will not be displayed after the error message. *Note:* An unsuccessful login means that all the parameters for the user name (**-u**), the password (**-p**) and the database name (**-d**) have been stated but at least one of the parameters was incorrect.
-no_printer_warning	causes that the error message "No default printer installed" will not be displayed in the case of no default printer being installed.

Hint: If you pass on the user name (-u), the password (-p) and the database name (-d) correctly as parameters, the ADOxx login screen will not be shown.

ATTENTION: Should the user name, password, database name, or font type names contain spaces, the entire parameter has to be put within quotation marks (e.g. "-uUser 1" for the user "User 1").

ATTENTION: User name and password are case sensitive!

Examples:

```
-dadoxxdb -winlogin
```

logs in the current system user to the ADOxx database "adoxxdb".

```
-uAdministrator -ppassword -dadoxxdb -winlogin
```

logs in a different system user "Administrator" with the system password "password" to the ADOxxdatabase "adoxxdb".

```
-uAdmin -ppassword -dadoxxdb
```

logs in the user "Admin" with the password "password" to the ADOxx database "adoxxdb".

```
-uAdmin -ppassword -dadoxxdb -x
```

shows the ADOxx login screen (see fig. 448) where the field "User name" contains the value "Admin", the field "Password" the encrypted string "password" and the field "Database name" the value "adoxxdb".

```
"-uUser 1" "-pUser 1" -ddatabase -h14
```

logs in the user "User 1" with the password "User 1" to the ADOxx database "database" using a font size of 14 pt in the ADOxx application window.

1.3 Start ADOxx WebService

Start the ADOxx WebServer application from a **command line window** (Windows command line prompt) out of the ADOxx installation directory with the following command:

```
  echo CC "AdoScript" SERVICE start port:80 | areena -u<user name> -p<password> -
d<DB name> -e -no_dialogs
```

Replace <*user name*> with the name of the ADOXX/system user, <*password*> with the password of the ADOXX/system user and <*DB name*> with the name of the ADOXX database.

Hint: This command only starts the ADOxx WebServcie. Additional functionality for applying the ADOxx WebServe has to be implemented accordingly.

2. Closing ADOxx

To close the Modelling Toolkit, double click on the exit button (to the left on the window bar) or click on the .

Within each component of the ADOxx Business Process Management Toolkit, it is also possible to close ADOxx by selecting the menu item "Quit ADOxx" from the menu list on the left.

The Modelling Toolkit checks if there are still opened models and if they contain unsaved changes. If all changes have been saved already, the safety questions appears:

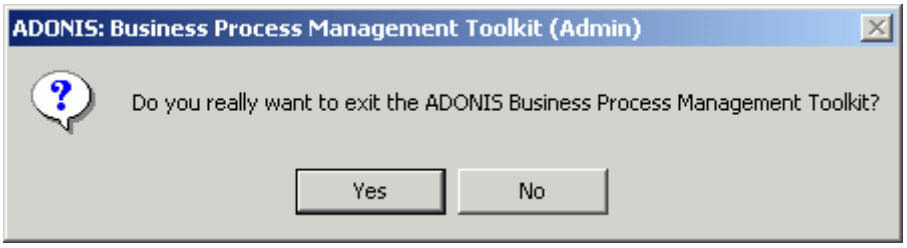


Figure 450: ADOxx Closing: A safety question

Click on **"Yes"** to finish ADOxx while selecting **"No"**, you will come back to the working area.

If one of the models contain unsaved changes, the following safety questions appears:

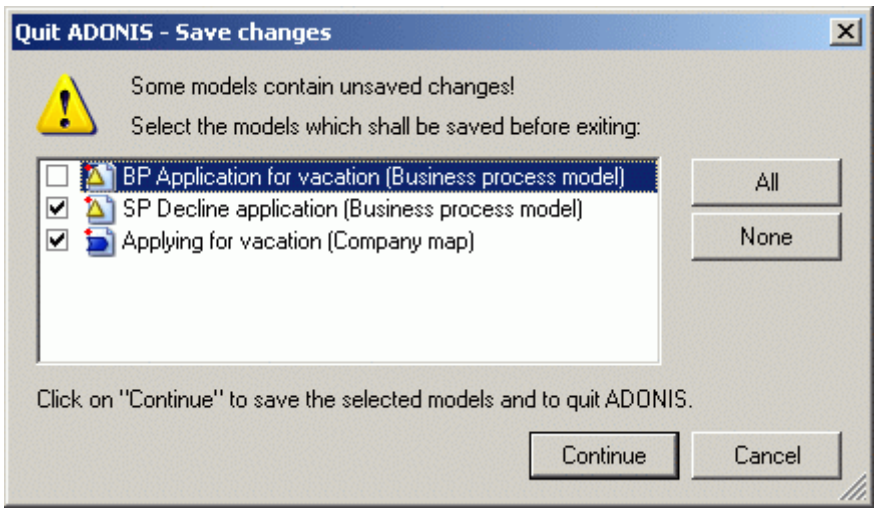


Figure 451: ADOxx Closing with unsaved changes

Remove "checks" from models, that should not be saved, simply by clicking on them. **"None"** will remove all "checks". **"All"** will check all models. Afterwards click on "Continue". The selected models will be saved and ADOxx will be closed.

3. Status Information

Select the option "Status" from the "Extras" menu in the Modelling Toolkit to view the current status of the ADOxx session. Clicking on the option opens the window "ADOxx: Status information" (see fig. 452).

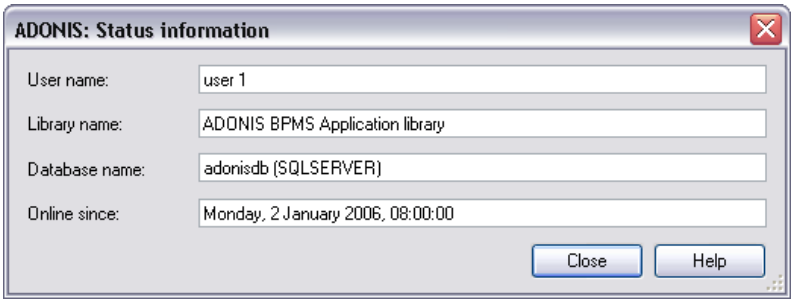


Figure 452: ADOxx status information in the Modelling Toolkit

The field "User name" shows the authorised ADOxx user, the field "Library name" the library assigned to the authorised ADOxx, user and the field "Database name", which contains the ADOxx database of the current ADOxx session. The field "Online since" shows the date and time of last connection.

4. Current Configuration

Select the option "Configuration" in the "Extras" menu to show the available components of the ADOxx Business Process Management Toolkit (current configuration).

After selecting the menu item, the window "<*name of database*>: Current configuration" (see fig. 453) is displayed.

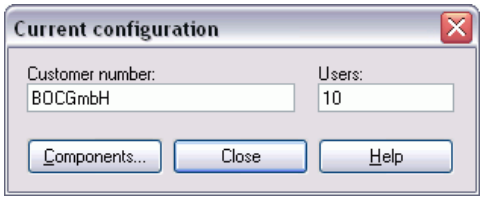


Figure 453: Current configuration of ADOxx

In addition to the customer number and the maximum number of simultaneous ADOxx users, you can view the current configuration of the ADOxx Modelling Toolkit by clicking on the "Components" button.

The current configuration of the ADOxx Business Process Management Toolkit is shown in an ADOxx Notebook. Every chapter represents a component.

The available components of the ADOxx Business Process Management Toolkit are marked by a tick.

© BOC Group 2008

5. Service Information

If you have problems or questions concerning ADOxx, select the option "Service" from the "Help" menu to view contact addresses. The window "Service" appears (see fig. 454).

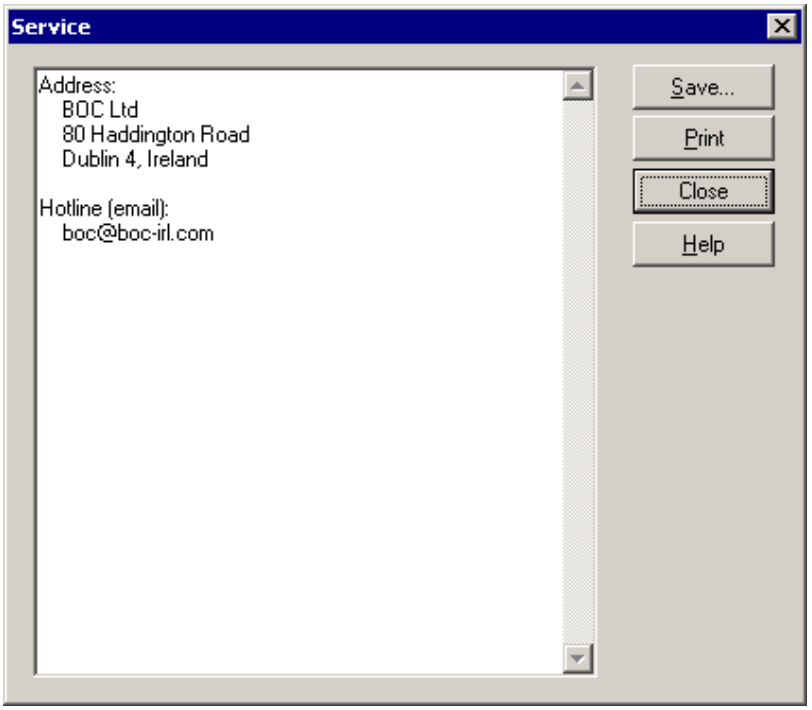


Figure 454: ADOxx service information

You can save this information to a file or print it out.

6. Regular Expressions

A regular expression is a specific search text, in which special strings are used in order to make it applicable to text patterns in a file. A regular expression consists of one or more non-empty *branches* (=strings). These branches are connected or separated by the character "|".

Example: *branch* | *branch* | *branch*;

If a string within the text searched meets the criteria defined by a branch, an appropriate result is displayed (see fig. 455).

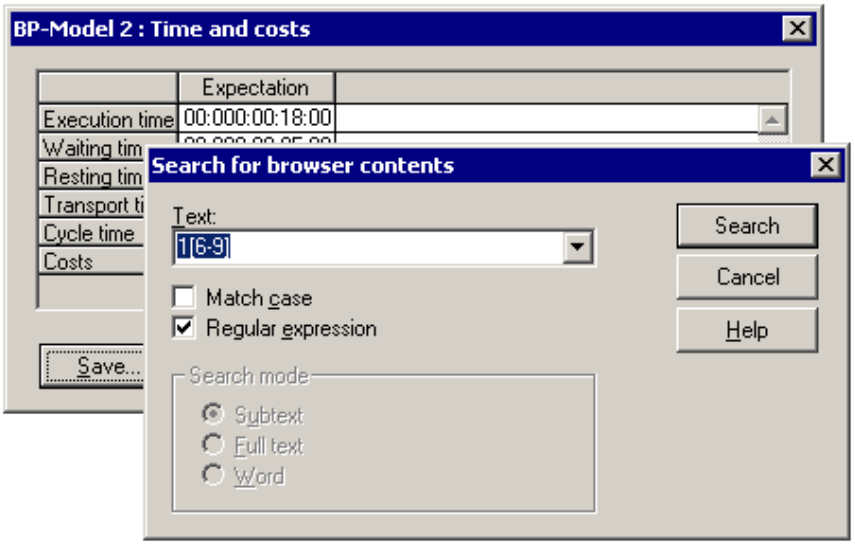


Figure 455: Search using regular expressions

Regular expression - meta-sign	**Description**
* ? . | () [] + \ and string(s) to be searched for	Search for all characters (letters, figures, other) except *** ? . | () [] + ** (=meta-signs or operators); the special function of the above signs is cancelled by putting a **** in front. **Example:** Search with *a*ctivity* for a*ctivity.
.	Instead of the **.** any character may occur in the result. **Example:** *w.rd* finds *ward, w4rd, wgrd, word,* etc.
[]	One of the characters within the brackets must be contained in the result, but the characters within the brackets may not be combined in the result. Please note that the order of the characters searched is of no importance when using square brackets (in contrast to round brackets). **Example:** *w[aeiou]rd* finds *ward, werd, wird, word, wurd* but, for example, not *waerd*.
[^]	The "caret - ^" within square brackets excludes the following string from the search result. **Example:** *wo[^0-9]rd* cannot find a result like *wo3rd*.
^	The result contains those lines starting with the character (combination) following the caret. **Example:** *^word* finds all lines starting with *word*.
$	The result shows all lines ending with the character (combination) in front of the "$". **Example:** *less$* finds all lines ending with *less*. If a line ends with *less.$*, the full stop must be cancelled by *less \ .$*.

The three following operators always refer to the character immediately preceding the operator:

?	The character preceding the "?" occurs either once or not at all (? represents the quantity 0). **Example:** *wo?rd* finds the result *wrd* or *word*.
*	The character preceding the "*" occurs either not at all (* represents the quantity 0) or an indefinite number of times. The search result for *wo*rd* is either *wrd* or *word, woord, wooord* etc.
+	The character preceding the "+" occurs either once or an indefinite number of times. The search result for *wo+rd* is either *word* or *woord, wooord, woooord, ...*.
()	Round brackets serve the grouping of character (sequences). The search result refers to the expression within the brackets. **Example:** *(wo)+rd* finds *wowoword* or *www**wo**rd*, but not *wooord*!
|	You can combine search criteria by describing several regular expressions. These *branches* are connected or separated by the character |. **Example:** *(w.rd | less | .)s*; a string is displayed if it meets the criteria of at least one *branch*, e.g. *words, wards, less.s*

7. ADOxx Attribute Types

ADOxx offers you the following types of attributes:

- Attribute Profile Reference (ATTRPROFREF) (see chap. 7.1, p. 733)
- Calendar (see chap. 7.9, p. 735)
- Date (see chap. 7.5, p. 734)
- Date and Time (DATETIME) (see chap. 7.6, p. 734)
- Double (see chap. 7.8, p. 735)
- Enumeration (see chap. 7.2, p. 733)
- Enumeration List (see chap. 7.3, p. 734)
- Expression (see chap. 7.4, p. 734)
- Integer (see chap. 7.7, p. 735)
- Interref (see chap. 7.12, p. 736)
- Long String (see chap. 7.10, p. 735)
- Program Call (see chap. 7.11, p. 736)
- Record (see chap. 7.13, p. 736)
- String (see chap. 7.14, p. 736)
- Time (see chap. 7.15, p. 736)

Hint: The attribute types are specified during the definition of the application library.

7.1 Attribute Profile Reference (ATTRPROFREF)

An attribute of the type "Attribute profile reference" (ATTRPROFREF) is defined for references to attribute profiles (see chap. 12., p. 100). An attribute profile reference attribute is made of the attributes defined in the referenced attribute profile.

The standard value for attributes of this type depends on the attribute types defined in the referenced attribute profile.

Hint: An attribute profile can contain attributes of all ADOxx attribute types except the type "attribute profile reference".

The value of an attribute of type "Attribute profile reference" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

7.2 Enumeration (ENUMERATION)

An attribute of the type "Enumeration" is characterised by a defined set of values. An "Enumeration" attribute has exactly one value of this set.

The standard value of this type is specified in the library definition.

Hint: The set of values of an "Enumeration" attribute can be extended in the library configuration (Library Management in the Administration Toolkit).

The value of an attribute of type "Enumeration" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

7.3 Enumeration List (ENUMERATIONLIST)

An attribute of the type "Enumeration list" is characterised by a defined set of values. An "ENUMERATIONLIST" attribute has either no value, a single or several values of this set.

The standard value of this type is specified in the library definition.

Hint: The set of values of an "Enumeration List" attribute can be extended in the library configuration (Library Management in the Administration Toolkit).

The value of an attribute of type "Enumeration list" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

7.4 Expression (EXPRESSION)

An attribute of type "Expression" is characterised by the possibility to determine its value via a flexible calculation rule.

The standard value for the attribute of this type is defined in the library definition.

Hint: You can define the possibility of the value entry of an expresion attribute in the library configuration (library administration in Administration Toolkit).

7.5 Date (DATE)

An attribute of the type "date" is defined for the entry of a date in format "YYYY:MM:DD" (year:month:day).

The standard value for attributes of this type is "2002:01:01" or a value defined in the application library.

The value of an attribute of type "Date" will be aligned right by default in the ADOxx browser (see chap. 8., p. 66).

7.6 Date and Time (DATETIME)

An attribute of type "Date time" is defined for the entry of a date in format "YYYY:MM:DD hh:mm:ss" (year:month:day hour:minute:second).

The standard value for attributes of this type is "2002:01:01 00:00:00" or a value defined in the application library.

The value of an attribute of type "Date time" will be aligned right by default in the ADOxx browser (see chap. 8., p. 66).

7.7 Integer (INTEGER)

An attribute of the type "Integer" is defined as an integer from -1,999,999,999 to 1,999,999,999.

The standard value of attributes of this type is "0" or a value defined in the application library.

Hint: The range of values of an "Integer" attribute can be limited during the library definition. An appropriate error message is displayed when a value outside the range is entered.

The value of an attribute of type "Integer" will be aligned right by default in the ADOxx browser (see chap. 8., p. 66).

7.8 Floating Point (DOUBLE)

An attribute of the type "Double" is defined for a float within +/-999,999,999,999,999 for an integer (without decimal places) or +/-999,999,999.999999 for figures with 6 decimals. The corresponding attribute value is displayed to 6 decimal places.

The standard value of attributes of this type is "0.000000" or a value defined in the application library.

Hint: The range of values of a "Double" attribute can be limited during the library definition. An appropriate error message is displayed when a value outside the range is entered.

Hint: The number of figures after the dot can be set in the library definition.

The value of an attribute of type "Double" will be aligned right by default in the ADOxx browser (see chap. 8., p. 66).

7.9 Calendar

An attribute of the type "Calendar" is defined for a performer calendar (=time at which the performer is available at work), for process calendar (=occurrence frequency and probability of business processes) and for agent calendars (=evaluation period of agents).

In the case of the performer calendar, the standard values for attributes of this type are defined from Monday to Friday from 09:00 - 12:00 and from 12:30 to 16:30 (excluding: 1.1., 1.5., 15.8., 26.10., 25.12. and 26.12.). In case of the process calendar it is set up similarly that is from Monday to Friday, between 09:00 - 12:00 and from 12:30 to 16:30 (excluding: 1.1., 1.5., 15.8., 26.10., 25.12. and 26.12.). However the probability with which the process will begin between these times is also specified. The default value for this is the uniform distribution between 15 and 30 minutes. In case of agents, the standard values for attribute of this type are defined from Monday till Saturday, from 00:00 to 24:00 o'clock.

7.10 Long String (LONGSTRING)

An attribute of type "Longstring" is defined for texts up to 32000 characters of any type.

The standard value of attributes of this type is "" (no entry) or a value defined in the application library.

The value of an attribute of type "Longstring" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

© BOC Group 2008

7.11 Program Call (PROGRAMCALL)

An attribute of the type "Program Call" is characterised by a defined value set of programs to be called or by AdoScripts. A program call has either no value or just one value from this set.

The standard value of attributes of this type is specified in the library definition.

> **Hint:** The value set of a "Program Call" attribute can be extended in the library configuration (Library Management in the Administration Toolkit).

The value of an attribute of type "Program call" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

7.12 Inter-model Reference (INTERREF)

An attribute of the type "Interref" is defined for inter-model references on models or objects.

The standard value of attributes of this type is "" (no entry).

The value of an attribute of type "Interref" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

7.13 Record (RECORD)

An attribute of the type "Record" (RECORD) is defined by a flexible list/record administration of assembled attribute types.

The standard value for the attributes of this type depends on the attribute type defined in the record class.

> **Hint:** A record can contain attributes of all ADOxx attribute types except the type "Record" and "Attribute profile reference" .

The value of an attribute of type "Record" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

7.14 String (STRING)

An attribute of the type "String" is defined for texts up to 3700 characters of any type.

> **Hint:** The maximum number of characters for name is 250.

The standard value of attributes of this type is "" (no entry) or a value defined in the application library.

The value of an attribute of type "String" will be aligned left by default in the ADOxx browser (see chap. 8., p. 66).

7.15 Time (TIME)

An attribute of the type "Time" is defined for the entry of a time period in the ADOxx time format YY:DDD:HH:MM:SS (years:days:hours:minutes:seconds).

The standard value of attributes of this type is "00:000:00:00:00" or a value defined in the application library.

The value of an attribute of type "Time" will be aligned right by default in the ADOxx browser (see chap. 8., p. 66).

8. Language Codes

In the following tables, all Windows languages are listed. Every table contains the languages of one codepage.

Hint: In ADOxx, both the ISO 639 codes and the Windows Language IDs can be used.

If more than one Windows Language IDs exists for an ISO 639 code (e.g. "German (Germany)" and "German (Austria)", they are combined into one Windows Language ID. These combinations are displayed in bold letters in the tables (German - for example - with the ISO 639 code "de" is always available in ADOxx with the Windows Language ID "1031").

ATTENTION: The ADOxx installation must be available in the specific languages for being able to use the language codes.

Language	ISO 639 Code	Windows Language ID
Thai	th	1054

Table 3: Languages within the codepage 874

Language	ISO 639 Code	Windows Language ID
Japanese	ja	1041

Table 4: Languages within the codepage 932

Language	ISO 639 Code	Windows Language ID
Chinese (Singapore)	zh	4100
Chinese (People's Republic of China)	**zh**	**2052**

Table 5: Languages within the codepage 936

Language	ISO 639 Code	Windows Language ID
Korean	ko	1042

Table 6: Languages within the codepage 949

Language	ISO 639 Code	Windows Language ID
Chinese (Hongkong S.A.R.)	zh	3076
Chinese (Macao S.A.R.)	zh	5124
Chinese (Taiwan)	zh	1028

Table 7: Languages within the codepage 950

Language	ISO 639 Code	Windows Language ID
Albanian	sq	1052
Bosnian (Bosnia and Herzegovina)	bs	5146
Croatian (Croatia)	**hr**	**1050**

Croatian (Bosnia and Herzegovina)	hr	4122
Czech	cs	1029
Hungarian	hu	1038
Polish	pl	1045
Romanian	ro	1048
Serbian (Latin)	**sr**	**2074**
Serbian (Latin, Bosnia and Herzegovina)	sr	6170
Slovak	sk	1051
Slovenian	sl	1060

Table 8: Languages within the codepage 1250

Language	**ISO 639 Code**	**Windows Language ID**
Azeri (Cyrillic)	az	2092
Belarussian	be	1059
Bulgarian	bg	1026
Macedonian (FYR)	mk	1071
Kazakh	kk	1087
Kyrgyz	ky	1088
Mongolian	mn	1104
Russian	ru	1049
Serbian (Cyrillic)	sr	3098
Serbian (Cyrillic, Bosnia and Herzegovina)	sr	7194
Tatar	tt	1092
Ukrainian	uk	1058
Uzbek (Cyrillic)	uz	2115

Table 9: Languages within the codepage 1251

Language	**ISO 639 Code**	**Windows Language ID**
Afrikaans	af	1078
Basque	eu	1069
Catalan	ca	1027
Danish	da	1030
Dutch (Belgium)	nl	2067
Dutch (Netherlands)	**nl**	**1043**
English (Australia)	en	3081
English (Belize)	en	10249
English (Canada)	en	4105
English (Caribbean)	en	9225

English (Ireland)	en	6153
English (Jamaica)	en	8201
English (New Zealand)	en	5129
English (Republic of the Philippines)	en	13321
English (South Africa)	en	7177
English (Trinidad and Tobago)	en	11273
English (United Kingdom)	**en**	**2057**
English (United States)	en	1033
English (Zimbabwe)	en	12297
Faroese	fo	1080
Finnish	fi	1035
French (Belgium)	fr	2060
French (Canada)	fr	3084
French (France)	**fr**	**1036**
French (Luxembourg)	fr	5132
French (Principality of Monaco)	fr	6156
French (Switzerland)	fr	4108
Galician	gl	1110
German (Austria)	de	3079
German (Germany)	**de**	**1031**
German (Liechtenstein)	de	5127
German (Luxembourg)	de	4103
German (Switzerland)	de	2055
Indonesian	id	1057
Italian (Italy)	**it**	**1040**
Italian (Switzerland)	it	2064
Icelandic	is	1039
Malay (Brunei Darussalam)	ms	2110
Malay (Malaysia)	ms	1086
Northern Sotho	ns	1132
Norwegian (Bokmal)	nb	1044
Norwegian (Nynorsk)	nn	2068
Portuguese (Brazil)	pt	1046
Portuguese (Portugal)	pt	2070
Quechua (Bolivia)	qu	1131
Quechua (Ecuador)	qu	2155
Quechua (Peru)	qu	3179

Sami (Inari, Finland)	se	9275
Sami (Lule, Norway)	se	4155
Sami (Lule, Sweden)	se	5179
Sami (Northern, Finland)	se	3131
Sami (Northern, Norway)	se	1083
Sami (Northern, Sweden)	se	2107
Sami (Skolt, Finland)	se	8251
Sami (Southern, Norway)	se	6203
Sami (Southern, Sweden)	**se**	**7227**
Swedish	**sv**	**1053**
Swedish (Finland)	sv	2077
Spanish (Argentinia)	es	11274
Spanish (Bolivia)	es	16394
Spanish (Chile)	es	13322
Spanish (Costa Rica)	es	5130
Spanish (Dominican Republic)	es	7178
Spanish (Ecuador)	es	12298
Spanish (El Salvador)	es	17418
Spanish (Guatemala)	es	4106
Spanish (Honduras)	es	18442
Spanish (Colombia)	es	9226
Spanish (Mexico)	es	2058
Spanish (Nicaragua)	es	19466
Spanish (Panama)	es	6154
Spanish (Paraguay)	es	15370
Spanish (Peru)	es	10250
Spanish (Puerto Rico)	es	20490
Spanish (Spain)	**es**	**3082**
Spanish (Spain)	es	1034
Spanish (Uruguay)	es	14346
Spanish (Venezuela)	es	8202
Swahili	sw	1089
Tswana	tn	1074
Welsh	cy	1106
Xhosa	xh	1076
Zulu	zu	1077

Table 10: Languages within the codepage 1252

Language	ISO 639 Code	Windows Language ID
Greek	el	1032

Table 11: Languages within the codepage 1253

Language	ISO 639 Code	Windows Language ID
Azeri (Latin)	az	1068
Turkish	tr	1055
Uzbek (Latin)	**uz**	**1091**

Table 12: Languages within the codepage 1254

Language	ISO 639 Code	Windows Language ID
Hebrew	he	1037

Table 13: Languages within the codepage 1255

Language	ISO 639 Code	Windows Language ID
Arabic (Algeria)	ar	5121
Arabic (Bahrain)	ar	15361
Arabic (Egypt)	ar	3073
Arabic (Iraq)	ar	2049
Arabic (Jordan)	ar	11265
Arabic (Kuwait)	ar	13313
Arabic (Lebanon)	ar	12289
Arabic (Libya)	ar	4097
Arabic (Morocco)	ar	6145
Arabic (Oman)	ar	8193
Arabic (Qatar)	ar	16385
Arabic (Saudi Arabia)	**ar**	**1025**
Arabic (Syria)	ar	10241
Arabic (Tunisia)	ar	7169
Arabic (U.A.E.)	ar	14337
Arabic (Yemen)	ar	9217
Farsi	fa	1065
Urdu	ur	1056

Table 14: Languages within the codepage 1256

Language	ISO 639 Code	Windows Language ID
Estonian	et	1061
Latvian	lv	1062

Lithuanian	lt	1063

Table 15: Languages within the codepage 1257

Language	ISO 639 Code	Windows Language ID
Vietnamese	vi	1066

Table 16: Languages within the codepage 1258

9. Change Password

You can use ADOxx after you have received a user name and a password from your ADOxx administrator.

When you start ADOxx for the first time, you should change your password. In addition, it is recommended that you change your password frequently for security reasons.

You can change your password in any active component by selecting the option "Change password" from the "Extras" menu. The window "Change password (User: <*user name*>)" (see fig. 456) appears.

Figure 456: Change user password

Enter your old password once and the new one twice and continue by clicking on the OK button or by pressing the enter key. If all entries are correct, the password will be changed.

ATTENTION: The password you enter is case sensitive, i.e. it distinguishes between capital and lower case characters. The password must consist of at least 3 characters. Valid characters for defining a password are the **figures 0 to 9**, the **letters a to z** and **A to Z**, **blanks** and the **characters ! # $ % & ( ) * + , - . / : ; < = > ? @ [ ] ^ _ { | } ~**.

Hint: The passwords are not displayed on the screen. For every character entered an asterisk is displayed.

10. Messages

The ADOxx email function enables you to send (see chap. 10.1, p. 745) messages to other ADOxx users created on the same ADOxx database, to receive (see chap. 10.4, p. 749) messages from other users and to reply (see chap. 10.2, p. 747) to received messages and/or to forward them (see chap. 10.3, p. 748).

Hint: The addresses of your messages will not be logged in ADOxx when the message is sent. Additionally, you can only send instant messages (see chap. 10.8, p. 752) to ADOxx users who are currently logged in.

The user can change settings (see chap. 10.9, p. 752) concerning messages in order to adapt message functionality to the user needs.

10.1 Send Messages

Send a message to one or more ADOxx users by selecting in the menu "Extras" the menu item "Messages" and the submenu "Send new message".

The window "Messages - Send message" (see fig. 457) will be displayed, in which you can enter the addressee, the subject and the text of the message.

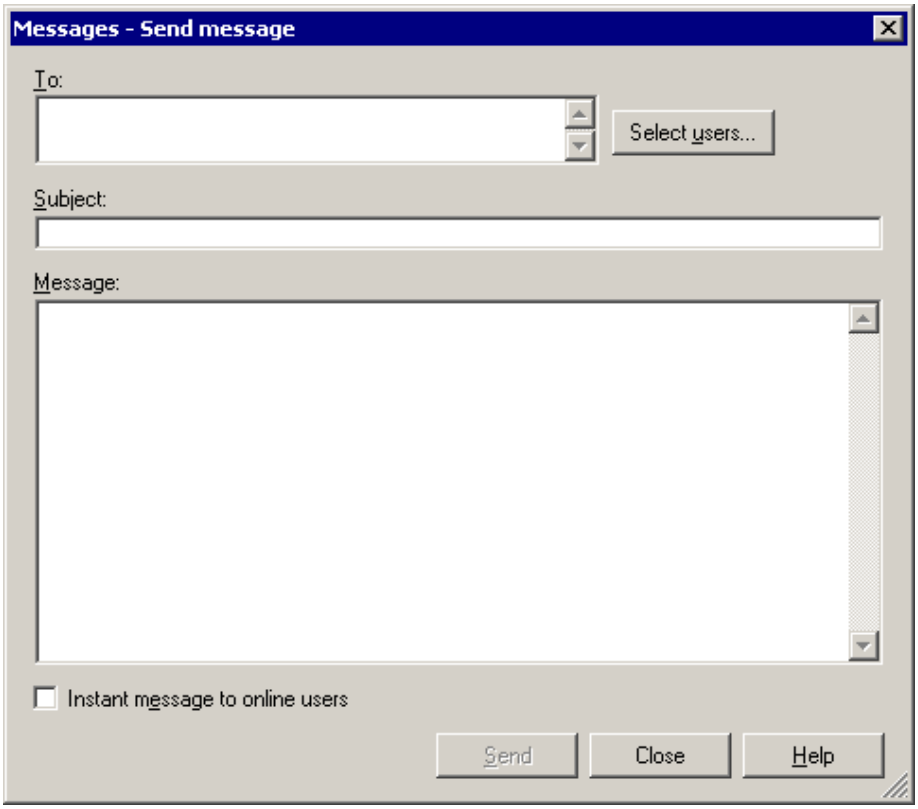


Figure 457: Send message

Enter in the field **"To"** the name(s) of the addressee(s)(= ADOxx user names).

Hint: When entering several ADOxx user names, separate them using a semicolon (";").

Hint: Pay attention to the correct spelling (particularly upper and lower case letters) of the addresses. Clicking on the button "User selection" (see chap. 10.1.1, p. 746) will provide you with support for the entry of the ADOxx user names.

In the field **"Subject"**, you can enter the subject of your message.

In the field **"Message"**, enter the text of your message.

Once you have successfully entered the information, click on the button "Send", to send your message. All addressees logged in will receive information (see fig. 461) about the new message. The addressees not logged in when the message is send will receive the information at their next login.

Select the option **"Instant message to online users"** to send relevant messages to logged in ADOxx users. Instant messages will be directly delivered, i.e. the addressee receives the whole message in a pop-up window (see fig. 464) on her/his screen.

10.1.1 Select Receiver

The support window for the selection of the email receiver(s) shows all ADOxx users created in the ADOxx database (see fig. 458).

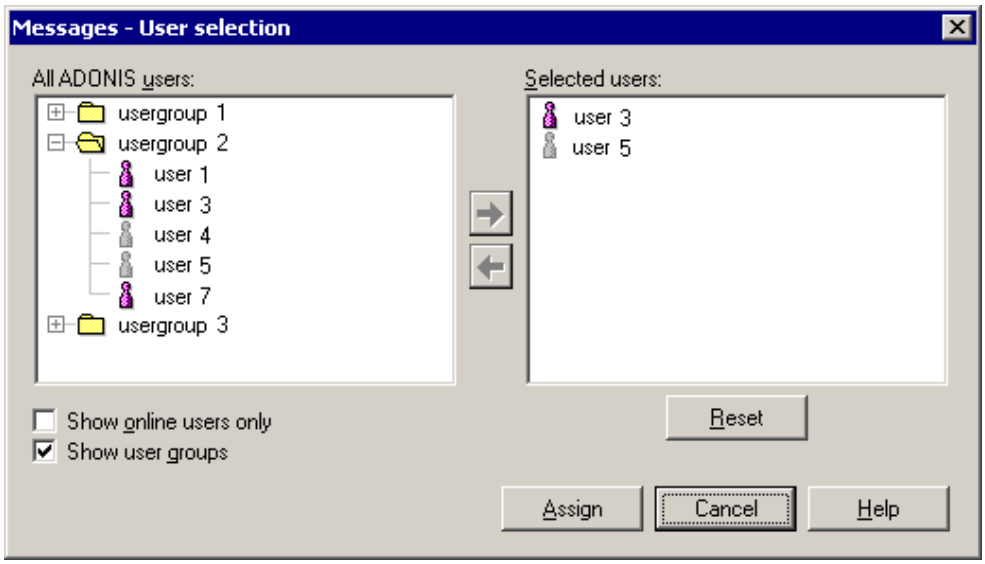


Figure 458: Select user as message receiver

All users are displayed in the **list "All ADOxx users"**. The icon before the user name indicates whether the user is already logged into the ADOxx database or not . Depending on the selected options, the display changes as follows:

Show online users only — If this option is activated, only the users currently logged in to the ADOxx database will be listed.

Show user groups — If this option is activated, the users will be shown in the structure of the user group they are assigned to. If this option is deactivated, you will see an alphabetically sorted list of all ADOxx users.

Select the receiver of the message in the list "All ADOxx users" and click on the arrow button , whereby the selected user will be transferred to the list **"Selected users"**.

When a user group is selected and the arrow button is clicked, all users, which are assigned to the user group, will be transferred to the list "Selected user".

You can remove single users from the list "Selected users", by selecting them and clicking on the arrow button . Clicking on the button **"Reset"** deletes all selected users.

Once you have successfully selected the message receivers, click on the button "Assign". As a result, the selected user will be transferred to the field "To" in the window "Messages - Send message" (see fig. 457).

10.2 Reply Message

Reply to messages of an ADOxx user, by selecting them in the window "Messages - received messages" (see fig. 462) and clicking on the "reply" button.

Hint: You can reply on an instant message directly from the message window (see fig. 464).

The window "Messages - Reply to messages" (see fig. 459), in which you enter the reply, will be displayed. The receiver and the subject field will be set automatically.

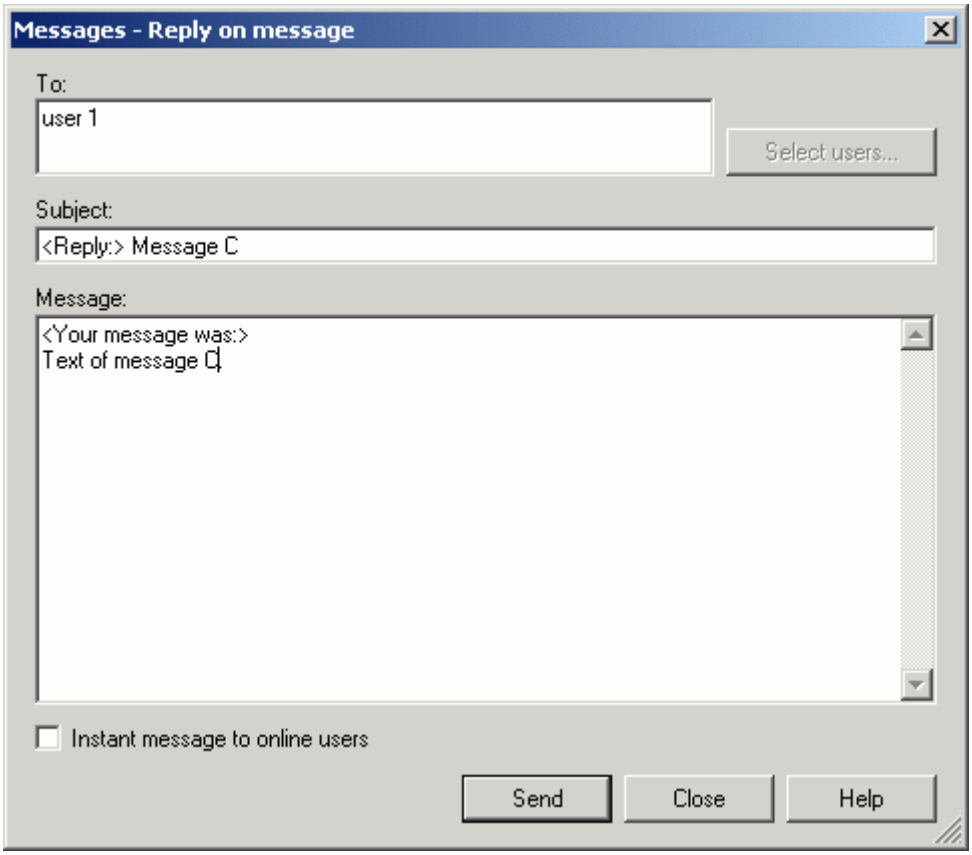


Figure 459: Reply to messages

The name of the receiver (=sender of the previously selected message) has been transferred to the field **"To"**.

The title of the previously selected message has been transferred to the **field "Subject"**, the title is preceded by "**<Reply:>**".

The text of the previously selected message has been transferred to the field **"Message"**, whereby the message is preceded by "**<Your message was:>**". Enter the text of your reply.

Once you have successfully entered the message, click on the button "Send", to send your message.

Select the option **"Instant message to online users"** to send the reply as an instant message to the ADOxx users currently online.

10.3 Forward Message

Select in the window "Messages - received messages" (see fig. 462), the messages you want to forward to one or more ADOxx users and click on the button "Forward".

The window "Messages - forward messages" (see fig. 460), in which you can, if necessary, enter the receiver and a comment, which will be displayed. The subject and the text will be set automatically.

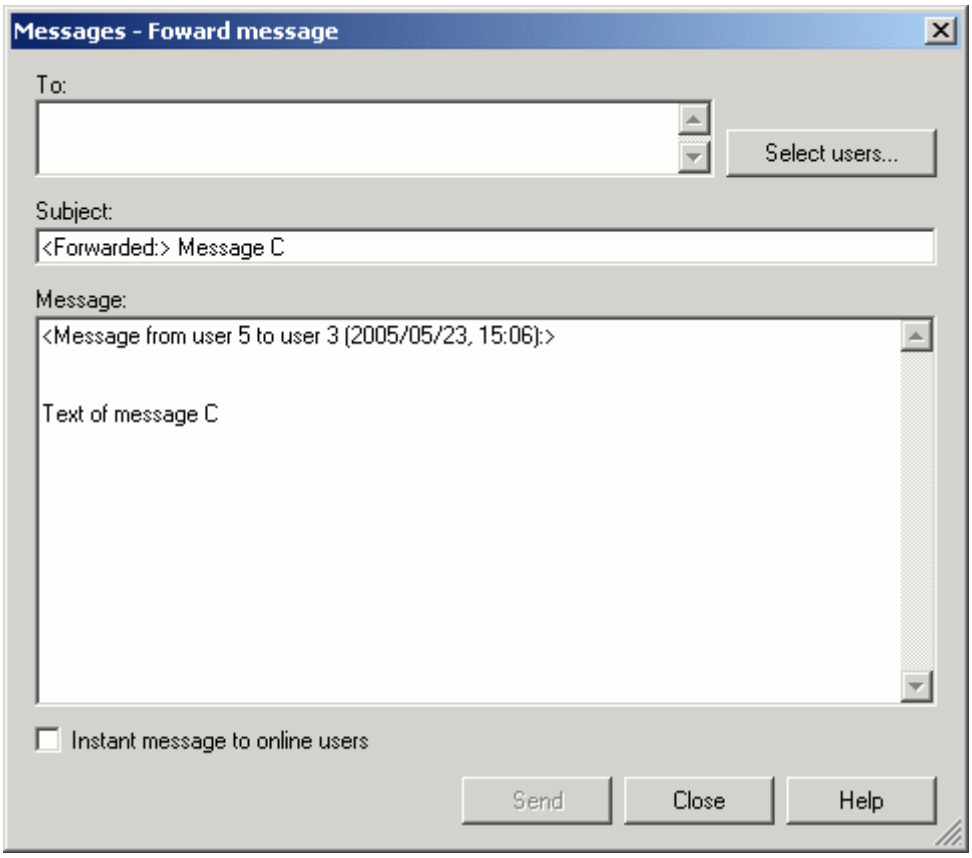


Figure 460: Forward message

Enter in the field **"To"** the name(s) of the addressee(s) (= ADOxx user names).

> **Hint:** When entering several ADOxx user names, you have to separate them with semicolons (";") .

> **Hint:** Pay attention to the spelling (incl. capitalisation) of the addresses. Clicking on the button "User selection" (see chap. 10.1.1, p. 746) you will get a support window for the entry of the ADOxx user names.

The title of the previously selected message has been transferred to the field **"Subject"**, the title is preceded by "**<Forwarded:>**".

The title of the previously selected message has been transferred to the field **"Subject"**, whereby the text is preceded by "**<message from *<user name>* to *<user name>* (*<date>*, *<time>*):>**". you can also enter text in the message to be forwarded.

After the successful entry click on the button "Send" to forward the message.

Select the option **"Instant message to online users"** to forward the message as an instant message to the ADOxx users.

10.4 Receive New Message

New messages arriving will be announced by the information window (see fig. 461).

Figure 461: Indication of new message

Click on the OK button to close the window and to have the received message displayed (see chap. 10.5, p. 749) at a later time.

Clicking on the button "Open" will show the window "Messages - Received messages" (see fig. 462) with the new messages (see chap. 10.6, p. 749) in ADOxx.

Hint: When the option "Info box on new messages" is deactivated in the settings (see chap. 10.9, p. 752), the arrival of new messages will be announced by a prompt sound (instead of the info box).

Hint: By deactivating the option "Messaging system active" in the settings (see chap. 10.9, p. 752), the arrival of new messages will **not** be announced.

10.5 Display Message

To display new messages, select the menu item "Messages" and the submenu "Read messages" from the "Extras" menu.

The window "Messages - received messages" (see fig. 462) containing all received messages (see chap. 10.6, p. 749) will be opened in ADOxx.

10.6 Received Message

The window "Messages - received messages" (see fig. 462) lists the received messages in a table.

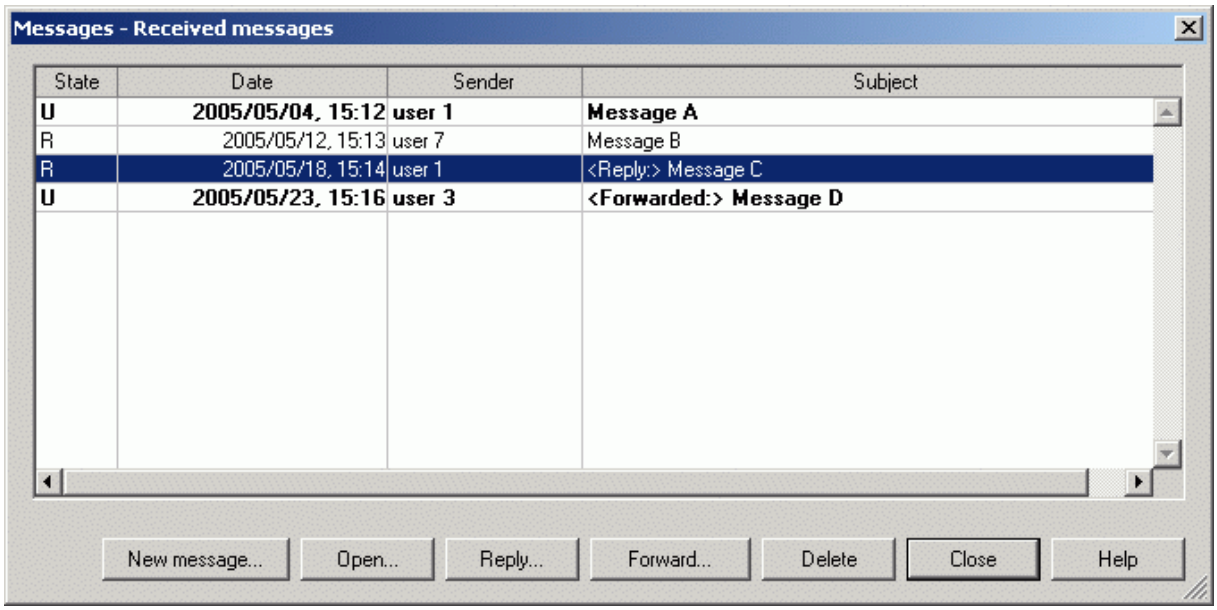


Figure 462: List of received messages

The list of received messages contains the following information:

- State
- Date
- Sender
- Subject

The column **"State"** can display the following entries:

N New messages, i.e. messages received just now or during the display of the window "Messages - received messages" (see fig. 462).

Note: New messages will be shown as unread messages (U) when closing the window "Messages - Received messages".

U Unread messages, i.e. messages that have been received but not yet read.

R Read messages.

Hint: New and unread messages will be brought out in bold.

The column **"Date"** contains the day and the time when the message was sent.

The column **"Sender"** contains the name of the ADOxx user who sent the message.

The column **"Subject"** will show the title of the message.

Hint: The title of the message will be preceded by "**<Reply:>**" if it is a reply to a previously sent message.

Hint: The title of the message will be preceded by "**<Forwarded:>**" if it is a forwarded message.

The following buttons are available under the list of received messages:

"New message" to create and send a new message (see chap. 10.1, p. 745).

"Open" to open and read a selected message (see chap. 10.7, p. 751).

"Reply" to reply to a selected message (see chap. 10.2, p. 747).

"Forward" to forward a selected message (see chap. 10.3, p. 748).

"Delete" to delete previously selected messages.

Hint: You can also use the functions "Open", "Answer", "Forward" and "Delete" after having selected a message if you open the context menu (right mouse button).

Hint: You can select several messages for the function "Delete".

10.7 Read Message

The text of a message will be displayed in a window (see fig. 463).

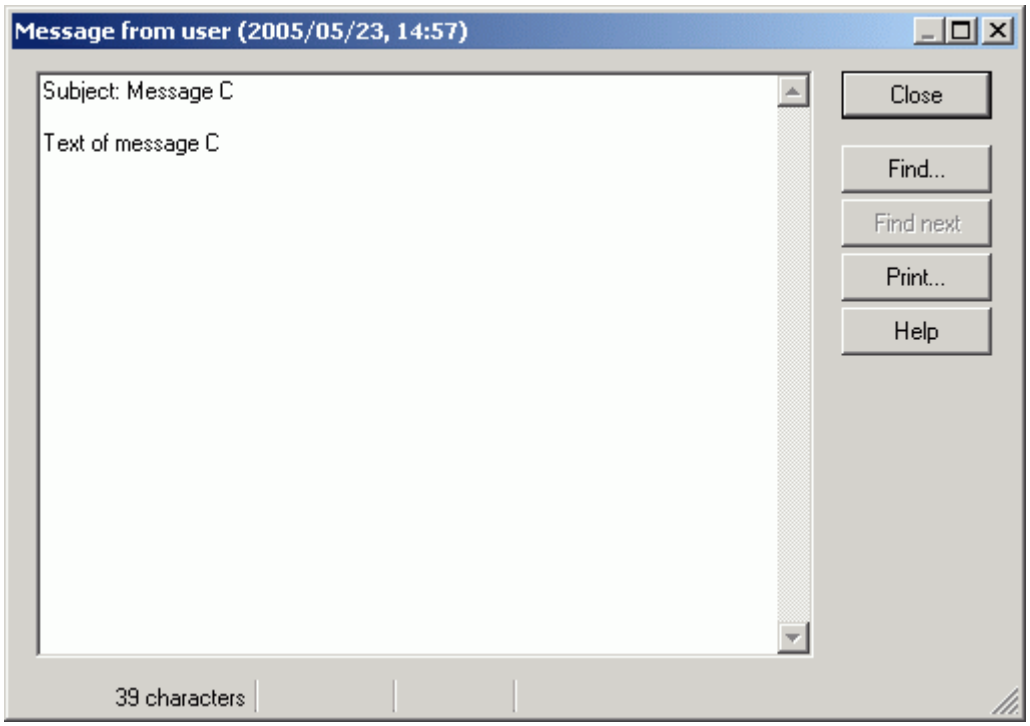


Figure 463: Display of received message

You can take the sender as well as the date and time of sending from the **title line** of the window.

The subject will be specified in the first line of the **text field** and the text of the message is displayed underneath.

Hint: If replying to a message, the information "**<Reply:>**" in the subject line as well as "**<Your message was:>**" will be displayed before the original message.

Hint: If forwarding a message, the information "**<Forwarded:>**" will be displayed in the subject line as well as "**<Message from *<user name>* to *<user name>* (*<date>*, *<time>*):>**" in the text field before the forwarded message.

By clicking on the **button "Press"** you can print the message, the **buttons "Find"** and **"Find more"** enable the finding of text within the message.

10.8 Receive Instant Message

When an instant message has been received, the whole message, i.e. sender, sending time, title and text will be displayed in a window (see fig. 464).

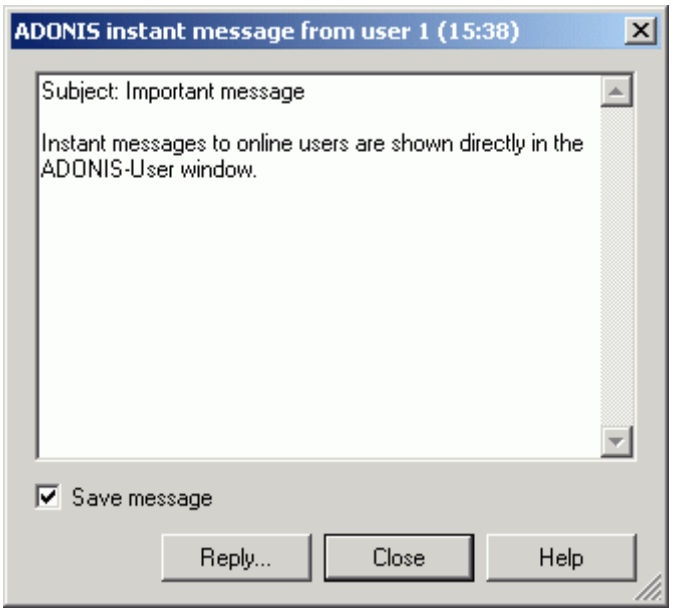


Figure 464: Instant message display

By clicking on the **button "reply"**, you can reply to the message (see chap. 10.2, p. 747).

Click on the button: **"Close"** to close the window. By activating the option **"Save message"**, the instant message will be sent and can be read again later.

Hint: By deactivating the option "Show instant message immediately" in the settings (see chap. 10.9, p. 752), the appropriate window (see fig. 461) will be displayed instead of the instant message.

10.9 Settings

In the window "Messages - Settings" (see fig. 465), you can carry out the settings described as follows.

Figure 465: Message settings

Option "Messaging system active"

If this option is activated, new messages will be announced by an appropriate info box (see fig. 461).

If this option is deactivated, the arrival of new messages will not be announced.

Option "Show instant message immediately"

If this option is activated, instant messages received, i.e. the whole message (sender, sending time, title and text), will be immediately shown in a window (see fig. 464).

If this option is deactivated, the arrival of instant message will be announced by an info box (see fig. 461).

Option "Info box on new messages"

If this option is activated, new messages will be indicated by an appropriate info box (see fig. 461).

If this option is deactivated, the arrival of a message will be announced by a sound prompt.

"Time interval for quering for new message (minutes)"

Specify the frequency (in minutes) to check for new message arrivals in the ADOxx database.

Hint: We recommend to deactivate the message system during long-lasting processing in ADOxx (e.g. simulation runs, import/export of models), since the announcement of new messages can abort the process, which can only be continued once the message window has been closed.

11. File Formats in ADOxx

The following file formats can be generated and are supported within ADOxx:

- ADONIS Binary Language (ABL) (see chap. 11.1, p. 754)
- ADONIS Comparable Representation (ACR) (see chap. 11.2, p. 754)
- ADONIS Definition Language (ADL) (see chap. 11.3, p. 755)
- ADONIS Protocol Format (APF) (see chap. 11.4, p. 755)
- Windows Bitmap (BMP) (see chap. 11.5, p. 755)
- Comma Separated Value (CSV) (see chap. 11.6, p. 755)
- Windows Enhanced Metafiles (EMF) (see chap. 11.7, p. 755)
- FlowMark Definition Language (FDL) (see chap. 11.8, p. 755)
- HyperText Markup Language (HTML) (see chap. 11.9, p. 756)
- JPEG File Interchange-Format (JPG) (see chap. 11.10, p. 756)
- ZSoft Paintbrush (PCX) (see chap. 11.11, p. 756)
- Portable Network Graphics (PNG) (see chap. 11.12, p. 756)
- Rich Text Format (RTF) (see chap. 11.13, p. 756)
- Scalable Vector Graphics (SVG) (see chap. 11.15, p. 757)
- ASCII-Text (TXT) (see chap. 11.15, p. 757)
- User Definition Language (UDL) (see chap. 11.16, p. 757)
- Extensible Markup Language (XML) (see chap. 11.16, p. 757)

Hint: The extensions listed below are suggested by the program. They are optional i.e. other extensions may be used instead. Note however that when searching for files, the filters are preset in order to search for the appropriate extension as listed below. Also using the preset extension makes it easier to see which type of file you are dealing with.

11.1 ADONIS Binary Language (ABL)

ABL files are generated when exporting application libraries from within the Administration Toolkit.

An application library which has been exported into an ABL file may be imported into the ADOxx Administration Toolkit (see chap. 3., p. 15) on another computer.

11.2 ADONIS Comparable Representation (ACR)

ACR files are also generated when saving simulation (see chap. 4., p. 332) and query (see chap. 3.4, p. 325) results to be later used in the "Comparing of results" option within the evaluation (see chap. 5., p. 433) component.

11.3 ADONIS Definition Language (ADL)

ADL files are generated when exporting (see chap. 6.2, p. 468) models (and/or model groups). ADL files can then be imported (see chap. 6.1, p. 445) and edited within the ADOxx Modelling Toolkit.

11.4 ADOxx Protocol Format (APF)

APF files can be created while running the "Capacity analysis" (see chap. 4.3, p. 354) or "Workload analysis" (see chap. 4.4, p. 367) simulation algorithms.

They can then be read by the Offline animation (see chap. 4.5, p. 386) function which allows you to play through the simulation again observing the times and optionally adding breakpoints at certain times.

11.5 Windows-Bitmap (BMP)

BMPs can be created when generating (see chap. 2.2.7.24, p. 156) graphics of models, model sections or the graphical representation (see chap. 8.16, p. 85) of the ADOxx browser (see chap. 8., p. 66) results. In addition BMP files can be created when generating a documentation (see chap. 6.5, p. 486).

BMP files may be opened or inserted into graphics programs and may be edited further there or integrated into other applications such as word processor applications.

11.6 Comma Separated Value (CSV)

CSV files will be created when saving the contents of the ADOxx browser (see chap. 8., p. 66).

CSV files can be further edited in spreadsheet calculation programs (without loss).

11.7 Windows Enhanced Metafiles (EMF)

EMF files will be created when generating (see chap. 2.2.7.24, p. 156) graphics from models or model parts and when copying the graphic result display (see chap. 8.16, p. 85) of the ADOxx browser (see chap. 8., p. 66). In addition EMF files can be created when generating a documentation (see chap. 6.5, p. 486).

EMF files can be inserted into graphic programs and be further edited there or connected to any applications (e.g. spreadsheet calculation programs).

11.8 FlowMark Definition Language (FDL)

FDL files are created when ADOxx models are transformed within the Import/Export component. FDL files can be imported into the Workflow Management system WebSphere MQ/MQSeries Workflow/FlowMark from IBM.

11.9 HyperText Markup Language (HTML)

HTML files can be created when saving the contents of the ADOxx browser (see chap. 8., p. 66) and also when generating the HTML documentation from model contents within the "Documentation component" (see chap. 6.5, p. 486).

HTML files can be viewed in a Web Browser.

11.10 JPEG File Interchange-Format (JPG)

JPG files can be created by generating (see chap. 2.2.7.24, p. 156) graphics from models, model sections or the graphical representation (see chap. 8.16, p. 85) of the ADOxx browser (see chap. 8., p. 66) results. In addition JPG files can be created when generating a documentation (see chap. 6.5, p. 486).

JPG files may be opened or inserted into graphics programs and may be edited further there or integrated into other applications such as word processor applications.

11.11 ZSoft Paintbrush Graphic (PCX)

PCX files will be created by generating (see chap. 2.2.7.24, p. 156) graphics from models, model sections or the graphical representation (see chap. 8.16, p. 85) of the ADOxx browser (see chap. 8., p. 66) results.

PCX files may be opened or inserted into graphics programs and may be edited further there or integrated into other applications such as word processor applications.

11.12 Portable Network Graphics (PNG)

PNG files will be created when generating (see chap. 2.2.7.24, p. 156) graphics from models or model parts and when copying the graphic result display (see chap. 8.16, p. 85) of the ADOxx browser (see chap. 8., p. 66). In addition PNG files can be created when generating a documentation (see chap. 6.5, p. 486).

PNG files can be inserted to graphic programs and then further edited or be connected to any application (e.g. spreadsheet programs).

11.13 Rich Text Format (RTF)

RTF files can be created when saving the contents of the ADOxx browser (see chap. 8., p. 66) and also when using the HTML generation within the "Documentation Component" (see chap. 6.5, p. 486).

RTF files can be further edited in spreadsheet calculation programs.

11.14 Scalable Vector Graphics (SVG)

SVG files are used when generating (see chap. 2.2.7.24, p. 156) graphics from models or model parts and when copying the graphical result representation (see chap. 8.16, p. 85) of the ADOxx browser

(see chap. 8., p. 66). In addition SVG files can be created during documentation generation (see chap. 6.5, p. 486).

In general, SVG files are used in HTML browsers.

Hint: SVG files can be displayed using Microsoft Internet Explorer (Version 5.0 and up, with SVG Viewer plugin) or Mozilla Firefox (Version 1.5 and up).

However, it is recommended to use Microsoft Internet Explorer with SVG Viewer Plugin, as Mozilla Firefox does not interpret text and some other SVG objects correctly.

11.15 ASCII-Text (TXT)

TXT files are generated when saving simulation (see chap. 4., p. 332) and analysis (see chap. 3., p. 306) as well as the contents of an ADOxx Notebook (see chap. 3.2, p. 48). TXT files can be inserted in other applications and edited further.

11.16 User Definition Language (UDL)

UDL files are created by exporting users (and/or user groups) in the User Management component of the Administration Toolkit. UDL files can then be imported into the Administration Toolkit on a different computer. (This can be useful for example where the database system is changing).

11.17 Extensible Markup Language (XML)

XML files will be created when exporting models. The models which have been exported to XML files, can for instance be imported to other computers, where they can be edited.

Hint: In ADOxx 1.0 XML files will be created with encoding "UTF-8".

12. Expressions

Expressions are - rougly speaking - formulae for investigating or calculating something. This can be both a fixed calculation or the processing of attribute values. The syntax of the language for EXPRESSION attributes is based on the following grammar:

ExprDefinition **:** **EXPR type:** *ResultType* [**format:***FormatString*]
expr:[**fixed:**] *CoreExpression* .

ResultType **:** **double** | **integer** | **string** | **time** .

Each definition of the EXPRESSION attribute starts with the keyword **EXPR**. The result type will be fixed by means of the attribute **type:** and a standard fomula expr. Each time an instance is created (object, model or connector), the formula will be used to calculate the result value of the expression.

By setting the modifier as **fixed:** the expression will be unchangeable in the Modelling Toolkit. It means that the EXPRESSION attribute evaluates the same formula everywhere.

Hint: The formula cannot be longer than 3.600 characters.

In the expressions where you can change the instances, each evaluation instance will be assigned to an appropriate expression. Then the specification of an expression in the "Expression-Definition" is optional and is of use for initial assignment of an expression to evaluation instances. For each evaluation instance, you can open an input window from the ADOxx Notebook, which enables the editing of the expression. The changeable expression will be saved to the evaluation instance under **expr**.

In ADOxx, three different types of expressions can be found:

- LeoExpressions**:** LeoExpressions are a set of basic functions and operators, from which an expression can be aggregated. They are used to calculate values, manipulate strings and perform other similar basic tasks.
- **CoreExpressions:** A CoreExpression is an ADOxx-specific extension of a LeoExpression. When creating a CoreExpression, both functions and operators of LeoExpressions and core-specific functions designed specifically for ADOxx can be utilised. CoreExpressions can only be used in attributes of type **EXPRESSION** (see chap. 7.4, p. 734).
- **AdoScriptExpressions:** An AdoScriptExpression is an extension of a LeoExpression. Consequently, all functions and operators of the LeoExpressions can be used here as well. Furthermore, specifically designed AdoScript functions can be utilised. AdoScriptExpressions can only be used within AdoScripts in the ADOxx Administration Toolkit.

The CoreExpressions functions:

Hint: The word "this" always refers to the context where the expression is evaluated.

aval (*instID***,** *attrName***)**

provides the value of the instance object specified by both parameters.

aval (*attrName***)**

provides the attribute value *attrName* of the instance.

aval (*instName***,** *attrName***)**

provides in this model of this class the instance with an attribute value *attrName*.

aval (*className***,** *instName***,** *attrName***)**

provides the attribute value of the instance in this model which is specified by the three parameters.

avalf (*format*, *instID*, *attrName***)**

avalf (*format*, *attrName***)**

avalf (*format*, *instName*, *attrName***)**

avalf (*format*, *className*, *instName*, *attrName***)**

These functions correspond to the call of aval without the first parameter, whereby INTERREF attribute will be formatted. The string *format* will be called for each reference, whereby the following replacements will be carried out: **%o** by the object name of the target object, **%c** by the class name of the target object, **%m** by the model name of the target model, **%t** by the model type name of the target model. The results will be joined together separated by a line break, which is the result of the call.

maval (*attrName***)**

provides the value of a model attribute *attrName* of this model.

irtmodels (*attrName***)**

provides the models of this instance referenced by the INTERREF attribute *attrName*.

irtmodels (*instID*, *attrName***)**

provides the models referenced in this model by the INTERREF attribute of this instance.

irtmodels (*instName*, *attrName***)**

provides the models referenced in this model by the INTERREF attribute of the instance of this class.

irtobjs (*attrName***)**

provides the objects referenced by an INTERREF attribute of this instance.

irtobjs (*instID*, *attrName***)**

provides the objects referenced in this model by the INTERREF attribute *attrName* of the instance *instID*.

irtobjs (*instName*, *attrName***)**

provides the objects referenced in this model by the INTERREF attribute *attrName* of the instance *instName* of this class.

objirsobjs ()

provides the objects referencing this instance. The usage of this function is not recommended due to great preformance losses.

objirsobjs (*targetName***)**

provides the objects referencing the instance *targetName*. The usage of this function is not recommended due to great preformance losses.

objirsobjs (*targetID***)**

provides the objects referencing the instance with the ID *targetID*. The usage of this function is not recommended due to great preformance losses.

modelirsobjs ()

provides the objects referencing this model. The usage of this function is not recommended due to great preformance losses.

profile (*attrName***)**

provides the attribute profile referenced by the INTERREF attribute *attrName* of this instance.

paval (*profilerefAttrName***, *attrNameInProfile*)**

determines the attribute profile referenced by the ATTRPROFREF attribute *attrNameInProfile* of this instance and provides the value of an attribute it contains *attrNameInProfile*.

pavalf (*format***, *profilerefAttrName***, *attrNameInProfile*)**

determines the attribute profile referenced by the ATTRPROFREF attribute *profilerefAttrName* of this instance and provides the value of an INTERREF attribute *attrNameInProfile* it contains, whereby the format is the same as in **paval**.

ctobj()

provides the target object of this connector.

cfobj()

provides the start object of this connector.

ctobjs (*relnName*)

provides the objects connected with this object in an outgoing direction via the specified relation *relnName*.

ctobjs (*instid***, *relnName*)**

provides the objects connected with the specified object in an outgoing direction via the specified relation *relnName*.

cfobjs (*relnName*)

provides the objects connected with this object in an incoming direction via the specified relation *relnName*.

cfobjs (*instid***, *relnName*)**

provides the objects connected with the specified object *instid* in incoming direction via the specified relation *relnName*.

conn (*relnName***, *fromObjs***, *toObjs*)**

provides the connectors of the class *relnName* between the specifed objects. The objects per parameter can be seen as one ID, or many IDs containing strings.

objofrow (*rowID*)

provides the instance, which contains the specified RECORD row *rowID*.

rcount (*recordAttrName*)

provides the number of rows of a RECORD attribute *recordAttrName* of this instance.

row (*recordAttrName***, *rowIndex*)**

provides a row *rowIndex* of a RECORD attribute *recordAttrName* of this instance. The numbering of rows starts with **1** and finishes with **rcount()**.

rasum (*recordAttrName***, *columnName*)**

provides the sum of a record column *columnName* of the attribute *recordAttrName* of this instance.

rasum (*idStr***, *recordAttrName***, *columnName*)**

determines the sum of the specified RECORD columns *columnName* for each specified instance and provides the sum of all the results.

prasum (*profileAttrName***, *recordAttrName***, *columnName*)**

determines the sum of a RECORD column *columnName* in the attribute profile referenced by an ATTRPROFREF attribute of this instance.

ATTENTION: Please note that all the summed values with **rasum()** or **prasum()** must be of the same type. The accepted types are integers and floating point numbers; invalid values are ignored. EXPRESSION attributes, which provide values in both formats, can also be summed up.

isloaded (*modelid***)**

returns '1' if the model *modelID* is loaded, otherwise '0'.

allobjs (*modelID***, *className***)**

provides all objects of the class *className* in the specified model *modelID*.

aql (*aqlExpr***)**

evaluates the AQL expression and provides the result objects in form of an ID string.

prevsl()

provides the predecessors of this swimlane.

nextsl()

provides the successors of this swimlane.

asum (*idStr***, *attrName***)**

provides the sum of all attribute values of the specified attribute *attrName* for the specified instances *idStr*.

amax (*idStr***, *attrName***)**

provides the maximum number of all attribute values of the specified attribute *attrName* for the specified instances *idStr*.

awsum (*instIDStr***, *objAttrName***, *connIDStr***, *connAttrName***)**

provides the weighted sum for all the specified instances *instIDStr* of all attribute values of the given attribute *objAttrName*. This sum is weighted by attribute values *connAttrName* of the connectors *connIDStr*.

pmf (*className***, *profileAttrName***, *recAttrName***, *orgunitColName***, *quantityColName***, *factorAttrName***)**

Human resource management function.

pmf (*className***, *recAttrName***, *orgunitColName***, *quantityColName***, *factorAttrName***)**

Human resource management function (old version).

class (*instID***)**

provides the classes ID of an instance *instID*.

class (*className***)**

provides the ID of the class having the name *className*.

mtype ()

provides the model type of this model.

mtype (*modelID***)**

provides the model type of the specified model *modelID*.

mtclasses (modelTypeName)

provides the classes allowed in the model type **modelTypeName**.

mtrelns (*modelTypeName***)**

provides the relation classes, which are allowed in the model type **modelTypeName**.

`allcattrs (`*`classID`***`)`**

provides all class attributes of the specified class *classID*.

`alliattrs (`*`classID`***`)`**

provides all instance attributes of the class *classID*.

`allrattrs (`*`relationID`***`)`**

provides all attributes of the specified relation *relationID*.

`attrname (`*`attrID`***`)`**

provides the name of the specified attribute *attrID*.

`atttype (`*`attrID`***`)`**

provides the type name ("INTEGER", "DOUBLE", "STRING", "DISTRIBUTION", "TIME", "DATETIME", "ENUMERATION", "ENUMERATIONLIST", "LONGSTRING". "PROGRAMCALL", "INTERREF", "EXPRESSION", "RECORD" or "ATTRPROFREF") of the specified attribute *attrID*.

Additionally, the following variables are preset in every EXPRESSSION attribute:

- **attrid** with the ID of this attribute
- **objid** with the ID of this object/connector
- **classid** with the ID of this class/relation class
- **modelid** with the ID of this model

For every evaluation instance the result of its expression is calculated and saved to the attribute values of the EXPRESSION attribute under **val**. A new calculation is triggered every time the result could possibly have changed. This is the case when the expression was changed or when an attribute value necessary for the expression has been changed in the core component.

13. HOMER Scenarios

A HOMER scenario describes the settings of an acquisition table (see chap. 1.1, p. 108) for a specific use (e.g. acquisition of performance times, acquisition of the documents used).

In HOMER the user can create different scenarios for the purpose of acquisition.

You can manage the scenarios (see chap. 13.1, p. 763) using the scenario manager.

The definition of the table structures (see chap. 13.2, p. 768), i.e. the definition of the settings of the acquisition tables is carried out over the scenario settings.

13.1 HOMER Scenarios Administration

Select in the menu "Options" the menu item "Scenario Manager", to open the window "HOMER: Scenario Manager" (see fig. 466).

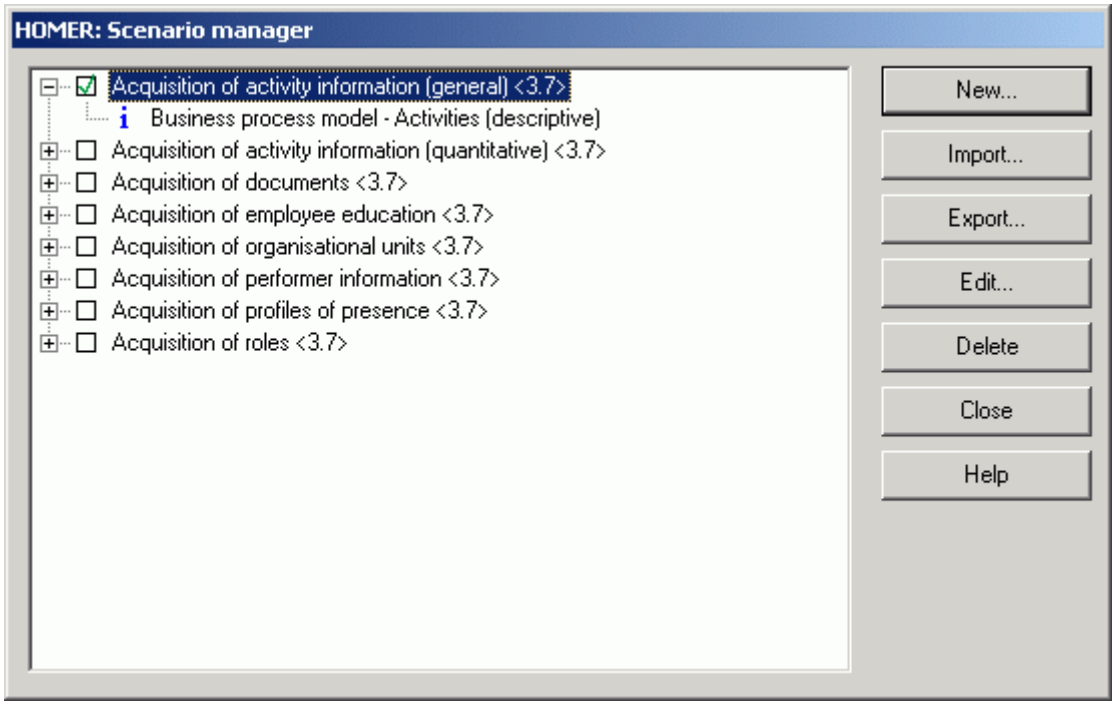


Figure 466: HOMER: Scenario Manager

Click on the respective button to do the following:

Create to create a new scenario (see chap. 13.1.1, p. 764),

Import to import scenarios (as well as INI files of earlier HOMER versions) (see chap. 13.1.2, p. 764),

Export to export scenarios (see chap. 13.1.3, p. 767),

Edit to edit the name, the verison and the keywords of the selected scenario (see chap. 13.1.4, p. 768),

Delete to delete the selected scenario (see chap. 13.1.5, p. 768),

Exit to exit the scenario manager.

The **selection of a scenario** is done when you activate the checkbox of the appropriate scenarios by clicking on it. An activated scenario is represented by the symbol ☑ and appears in the status bar of HOMER.

13.1.1 Creating HOMER Scenario

To create a scenario, click in the "HOMER: Scenario manager" window (see fig. 466) on the button "New". Enter the name, a version number as well as a keyword for the new scenario in the window "HOMER: Create new scenario" (see fig. 467).

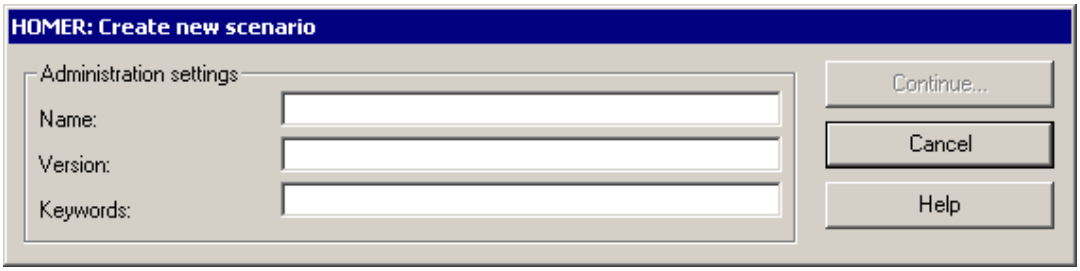


Figure 467: Create HOMER scenario

Hint: The fields "Name" and "Version" have to be filled.

Then click on the button "Continue", the scenario will be created and then you can define the structure of the acquisition table (see chap. 13.2, p. 768).

13.1.2 Importing HOMER Scenario

By importing the HOMER scenarios, you can

- import (see chap. 13.1.2.1, p. 765) scenarios created and exported with HOMER 3.6 or higher (XLS files) or,
- convert (see chap. 13.1.2.2, p. 766) scenarios (INI files) created in earlier HOMER versions to the format of HOMER 3.7.

ATTENTION: The conversion of existing INI files cannot be aborted. Depending on the size of the scenario this might take a few minutes.

Hint: You should transfer all existing acquisition tables to ADL first with the old HOMER version, since columns will be moved during the conversion. Once you have successfully converted the INI files, you can **automatically** transfer the acquisition tables converted in ADL into HOMER.

ATTENTION: Scenarios from XLS files as well as scenarios from converted INI files can only be used with HOMER version 3.6 and higher.

To import scenarios or existing INI files, click in the window "HOMER: Scenario manager" (see fig. 466) on the button "Import".

The window "Select HOMER scenario" will appear (see fig. 468).

ATTENTION: INI files converted for HOMER 3.7 can only be used with HOMER 3.7.

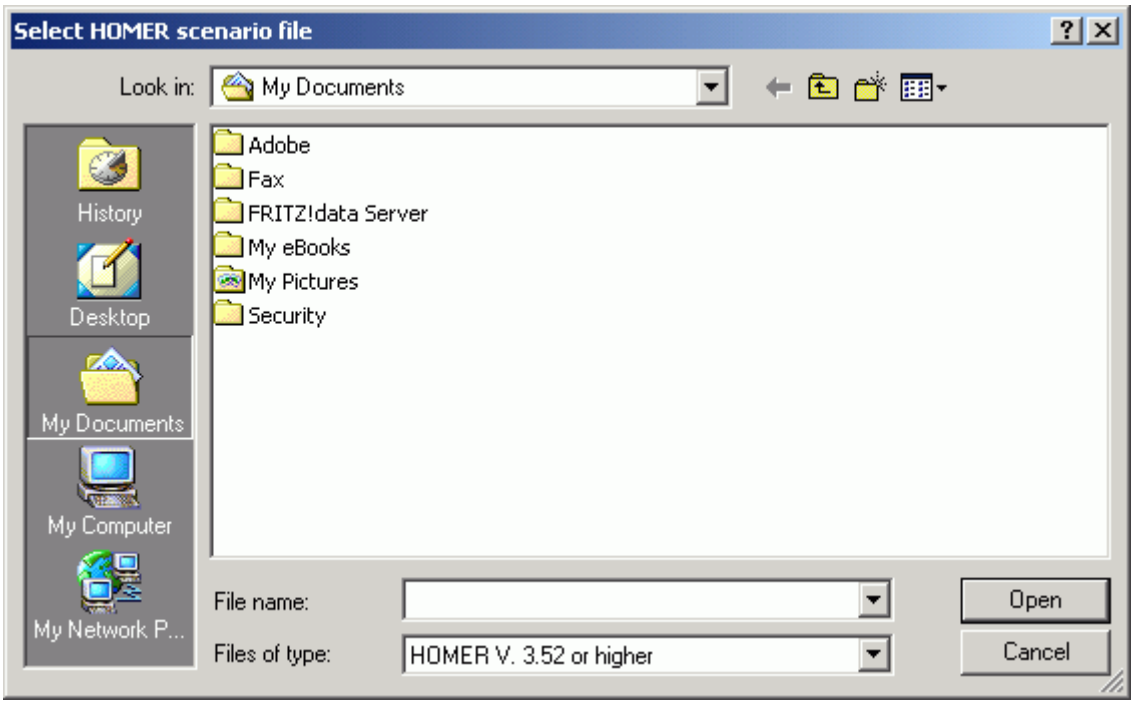


Figure 468: Import HOMER scenario - Select INI file

Select from the **list "Type of file"** the entry "HOMER V. 3.52 or higher", to import (see chap. 13.1.2.1, p. 765) a HOMER scenario, or "HOMER V. 2.0 to V. 3.5", to convert (see chap. 13.1.2.2, p. 766) an INI file.

13.1.2.1 Importing HOMER Scenario

ATTENTION: Scenarios from XLS files as well as scenarios from converted INI files can only be used with HOMER version 3.6 and higher.

Select the XLS file to import and then click on the button "Open". The window "HOMER: Select scenario for the scenario import" (see fig. 469).

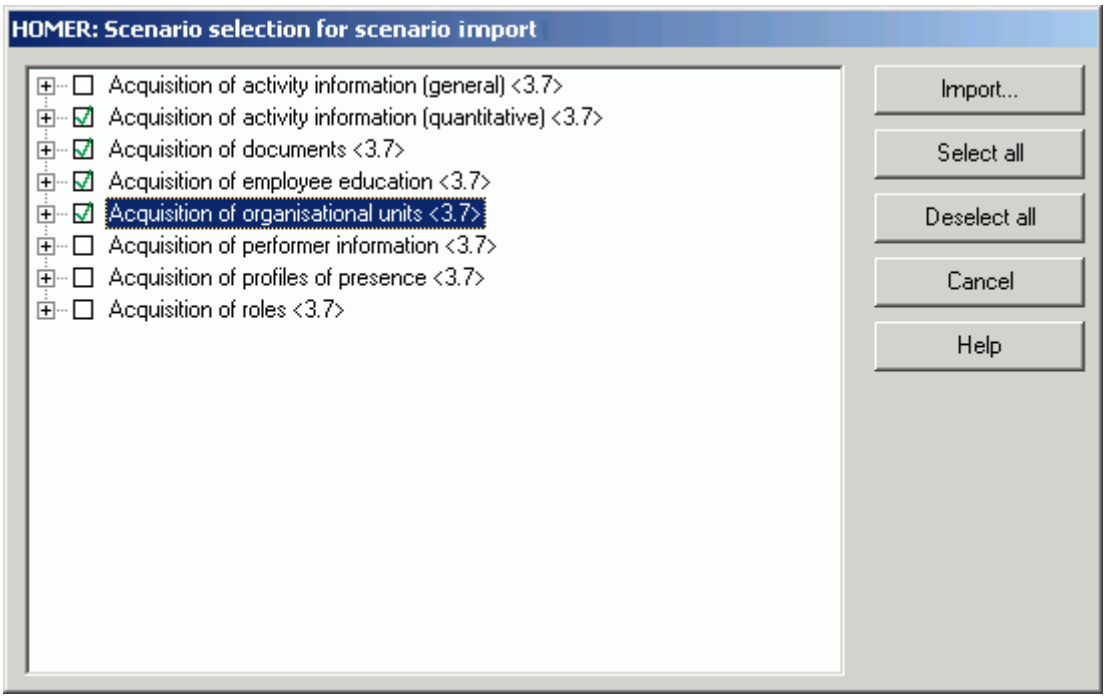


Figure 469: HOMER scenario import - Scenario selection

Select the scenario to be imported (multiple selection is possible) and then click on the button "Import", to start the import.

Hint: When you import an existing scenario, the error message [hacini-34] will be displayed. Close this message and then change the name and/or the version number of the scenario you want to import in the window "HOMER: edit scenario" (see chap. 13.1.4, p. 768).

The successful import will be indicated by a specific message.

Hint: If the file HOMER.xls (in the HOMER program directory) is write-protected no new scenarios are created in the window "HOMER: scenario manager" even when receiving the message that the import has been successful. Remove the write-protection to successfully import the scenarios.

13.1.2.2 Converting HOMER INI File

ATTENTION: The conversion of existing INI files cannot be aborted. Depending on the size of the scenario this might take a few minutes.

Hint: You should transfer all existing acquisition tables to ADL first with the old HOMER, since columns will be moved during the conversion. Once you have successfully converted the INI files, you can **automatically** transfer the acquisition tables converted in ADL again into HOMER.

Select the INI file you want to import and then click on the button "Open". The window "HOMER: Create new scenario" (see fig. 470) will appear.

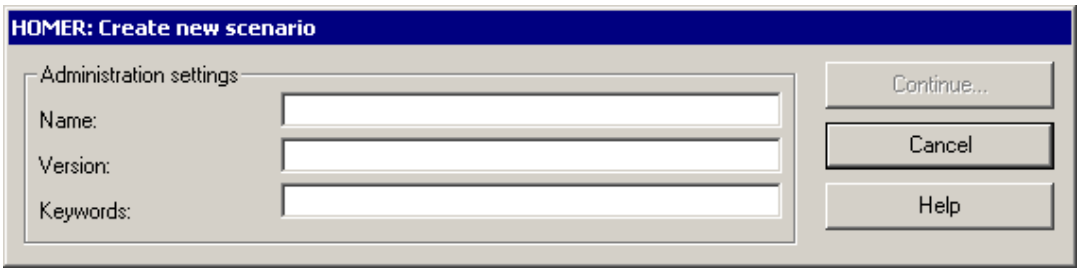


Figure 470: Import HOMER scenario - create scenario

Enter the name and the version number of the scenario you want to create and, if necessary, some keywords and then click on the button "Next".

Hint: The fields "Name" and "Version" must be filled.

After the conversion, the window "HOMER: scenario settings" with the defined table structure (see chap. 13.2, p. 768) will appear.

Check the settings (especially the version number of your current ADOxx version as well as the name of the basis library). To finish the conversion, click on the button "Close".

13.1.3 Exporting HOMER Scenario

When exporting scenarios, you can export an existing scenario to an XLS file, for instance to save this scenario or to transmit it to other HOMER users. The XLS file is protected by a password and can only be edited using HOMER.

To import scenarios or to import existing INI files, click in the window "HOMER: Scenario manager" (see fig. 466) on the button "Export".

The window "HOMER: Select scenario for the scenario export" (see fig. 471), with all previously defined scenarios will be opened.

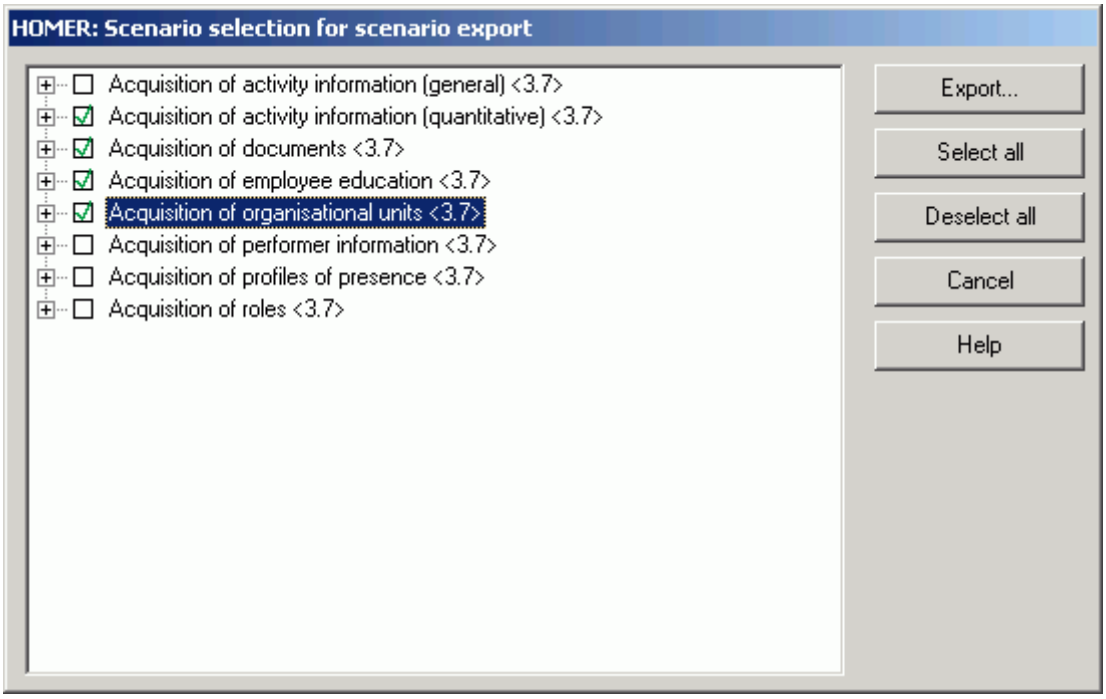


Figure 471: HOMER-export scenarios - scenario selection

Select the scenario you want to export and then click on the button "Export".

In the window "Export scenario", enter the file name as well as the path for the XLS file, to which the selected scenarios should be exported, and then click on the button "Save".

The successful export will be indicated by an appropriate message.

13.1.4 Editing HOMER Scenario

To edit a scenario, select in the window "HOMER: Scenario manager" (see fig. 466) the appropriate scenario and then click on the button "Edit". In the window "HOMER: edit scenario" (see fig. 472), you can change the name, the version number as well as the keyword.

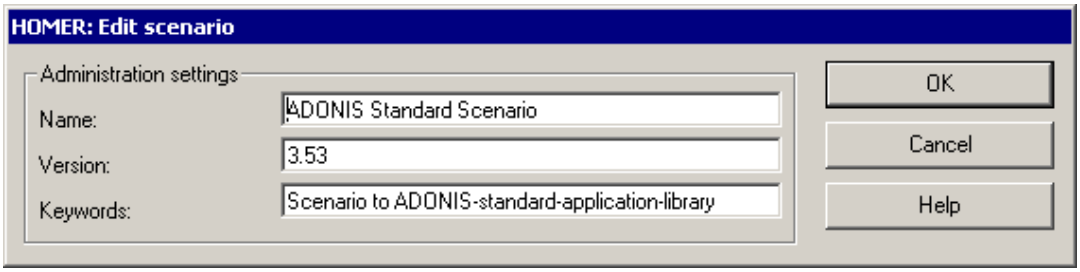


Figure 472: Edit HOMER scenario

Hint: The fields "Name" and "Version" have to be filled.

Then click on the OK button, to adopt the changes.

13.1.5 Deleting HOMER Scenario

To remove **one** scenario from HOMER, select this scenario in the window "HOMER: Scenario manager" (see fig. 466) and then click on the button "Delete".

Hint: Should there be only one scenario available, the button "Delete" is not available.

Before deleting the selected scenario, an appropriate security query will appear.

13.2 Defining the Structure of Acquisition Tables

The definition of the structure of the HOMER acquisition tables consists of fixing the parameter and creating the required attributes.

The **model parameters** (see chap. 13.2.1, p. 769) refer to the basic model information for instance the name of the library used, the name of the relation to create etc. This information will be used for the settings of HOMER.

The **object settings** (see chap. 13.2.2, p. 771) determine the features (attributes) of your activity. Some examples are for instance (specialised) descriptions, type of performer or also quantitative attributes such as performance and waiting times.

13.2.1 General Model Parameters

To adjust, the general model parameter of HOMER, select in the menu "Options" the menu item "Scenario settings". The window "HOMER: Scenario settings" (see fig. 473) will appear, in which you will carry out the following described settings and then save it by clicking on the "OK" button.

Register **"General configuration"**:

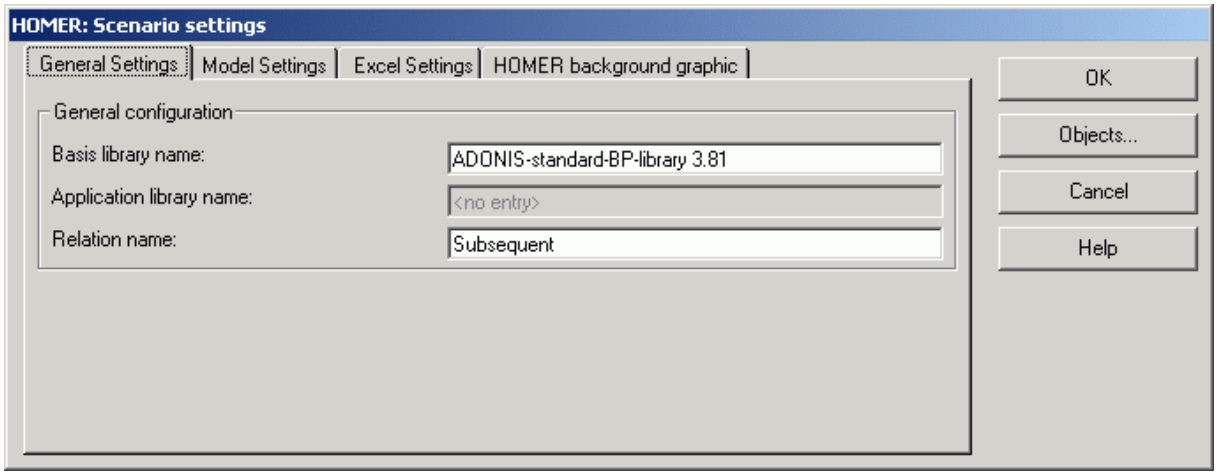


Figure 473: HOMER: scenario settings - general configuration

Enter in the field **"Basis library name"** the name of the BP or WE library, in which the modelling objects that you want to generate or to read, are defined.

Enter in the field **"Application library name"** the name of the application library, in which the modelling objects that you want to generate or to read, are defined.

> **Hint:** In the field "Name of the application library", it is only possible/required to have one entry, in case you want to administrate attribute profiles.

Enter in the field **"Relation name"** the name of the relation by which you want to connect the single objects to each other.

> **Hint:** If you don't want to generate a relation, enter the value "None".

Register **"Model settings"**:

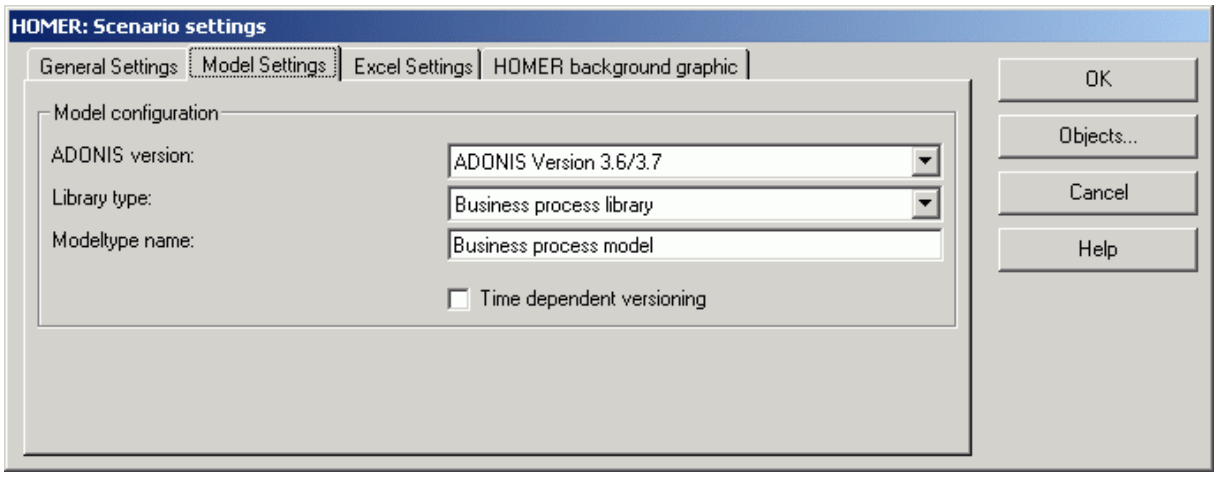


Figure 474: HOMER: scenario settings - model settings

Select from the list **"ADOxx version"** the version of ADOxx for which you want to create or read ADL files.

Select from the list **"library type"** the type from which you want to create or read ADL files.

Enter in the field **"model type name"** the name of the model type you want to generate or read.

> **Hint:** Model types can only be generated for ADOxx from version 3.0.

Click on the option **"time-related versioning"**, if you use an application library with activated time-related versioning (see chap. 10.1.2, p. 92).

> **Hint:** The time-related versioning can only be realised with ADOxx from version 3.5, this is why this field is inactive, when you have selected the option "ADOxx version 2.x" or "ADOxx version 3.x" in the list "ADOxx version".

Register **"Excel settings"**:

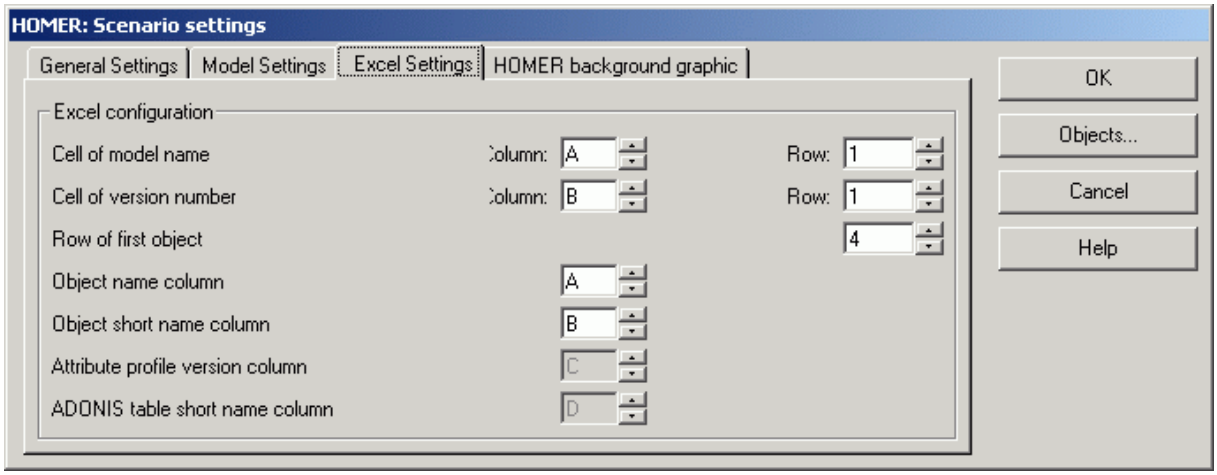


Figure 475: HOMER: scenario- settings - Excel settings

Please define the Excel fields, in which information from the generated Excel fields and specific for model and attribute profiles are found:

Cell of model name **Column** and **row** containing the model name

Cell of version number **Column** and **row** containing the model version

Row of first object Row containing the first modelling objects

Object name column Column containing the name of the modelling objects

Object short name column Column containing the class short name of the modelling objects

Attribute profile version column Column containing the version information for attribute profiles

ADOxx record short name column Column containing the short names for ADOxx records

Hint: The column of the attribute profile version can only be specified, if you have previously created an attribute profile. The column of the ADOxx table short name can only be specified, if you have previously created an ADOxx table .

Register **"HOMER background graphic"**:

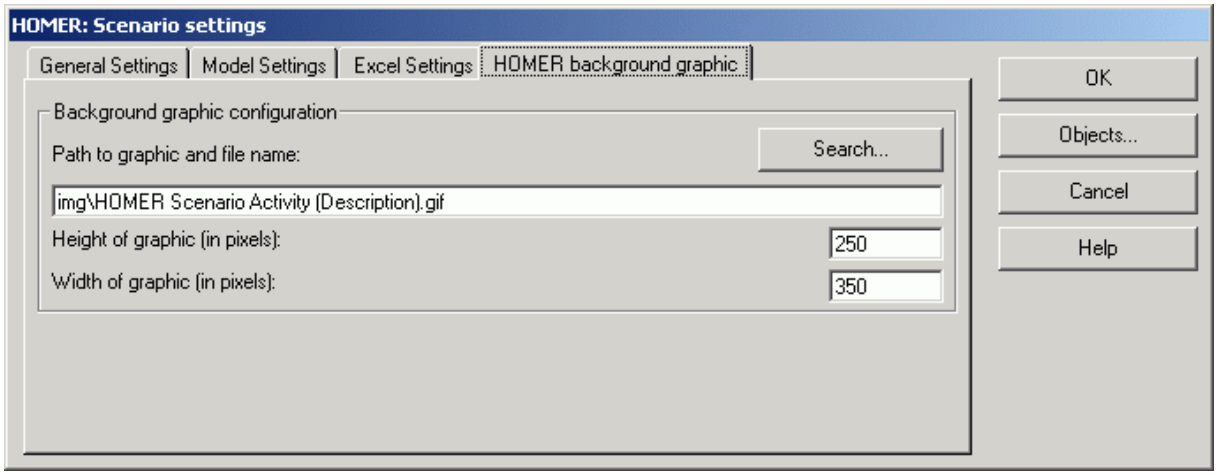


Figure 476: HOMER: scenario-settings - HOMER background graphic

Enter in the field **"Path to graphic and file name"** the path and the name of the graphic file you want to have displayed as background picture for the current scenario in the main HOMER window. Clicking on the button "Search" will provide you support for this.

ATTENTION: Always use the absolute (full) path.

The following graphic formats are available as background graphic:

- Windows Bitmap (BMP)
- Windows Meta File (WMF)

Enter in the fields **"Height of the graphic (in pixels)"** and **" Width of the graphic (in pixels)"** the desired height and width of the selected graphic.

Hint: The maximum height for a background graphic is 550 Pixels, the maximum width is 705 Pixels. Please note that graphics bigger than these limits will be cut.

13.2.2 Object Settings

To adapt the object settings of HOMER, select in the menu "Options" the menu item "Scenario settings". The window "HOMER: Scenario settings" (see fig. 473) will appear. Click on the button "Objects" of this window. The window "HOMER: Object settings" (see fig. 477) will appear.

Alternatively you can start from the window "HOMER: Scenario settings" (see fig. 473) and click on the button "Objects" to the window "HOMER: Object settings".

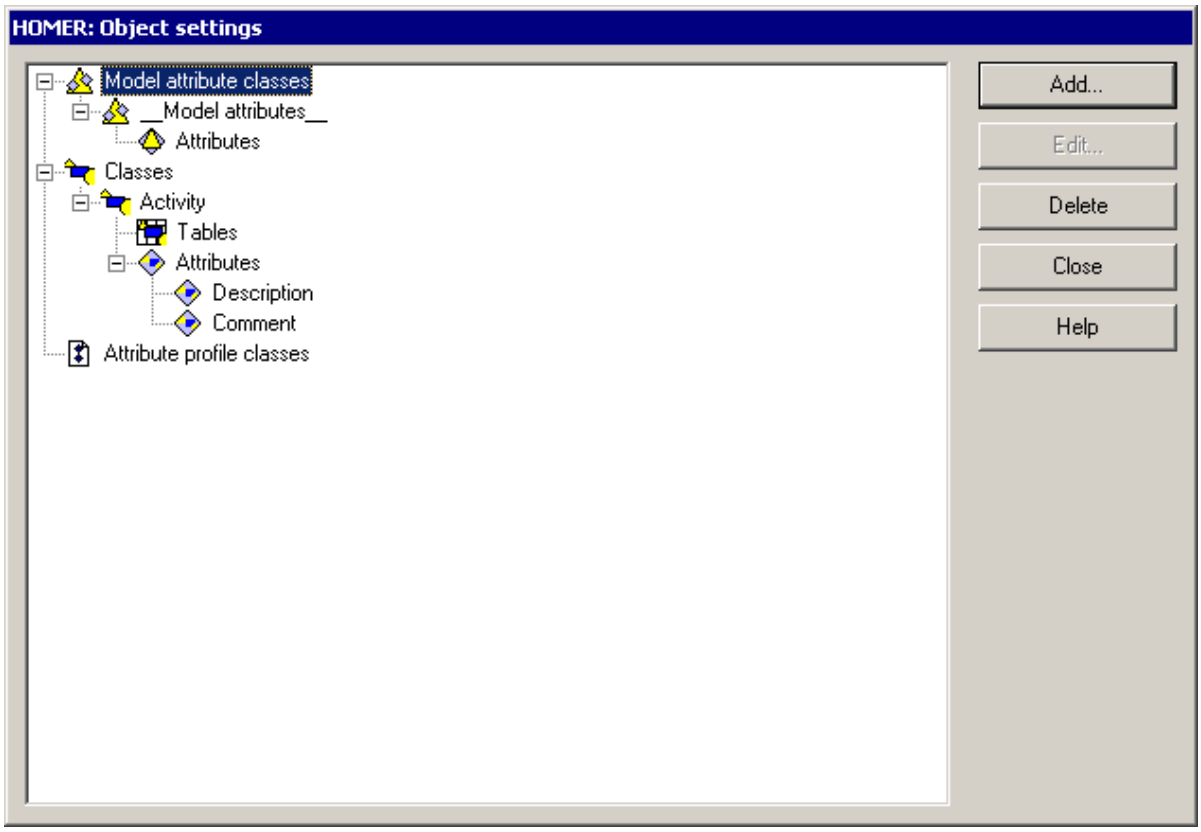


Figure 477: HOMER: classes

Starting from the window "HOMER: Object settings", you can perform the following operations:

- **Classes**
  - add (see chap. 13.2.2.1, p. 773),
  - edit (see chap. 13.2.2.2, p. 774) or
  - delete (see chap. 13.2.2.3, p. 774),
- **Attribute profile classes**
  - add (see chap. 13.2.2.4, p. 774),
  - edit (see chap. 13.2.2.5, p. 775) or
  - delete (see chap. 13.2.2.6, p. 775),
- **Records**
  - add (see chap. 13.2.2.7, p. 776),
  - edit (see chap. 13.2.2.8, p. 776) or
  - delete (see chap. 13.2.2.9, p. 777),
- **Attributes**
  - add (see chap. 13.2.2.10, p. 777),
  - edit (see chap. 13.2.2.13, p. 783) or
  - delete (see chap. 13.2.2.14, p. 785).

- **Model attribute classes**
  - add (see chap. 13.2.2.15, p. 785),
  - edit (see chap. 13.2.2.16, p. 786) or
  - delete (see chap. 13.2.2.17, p. 787),
- **Model attributes**
  - add (see chap. 13.2.2.18, p. 787),
  - edit (see chap. 13.2.2.19, p. 788) or
  - delete (see chap. 13.2.2.20, p. 788).

13.2.2.1 Adding Classes

Hint: The definition of a class can be found at "Terms and definitions" (see chap. 1., p. 32).

To create a class, click on the object "Classes" in the object list of the window "HOMER: Object settings" (see fig. 477) and then on the button "add". The window "HOMER: Add object" (see fig. 478) will appear.

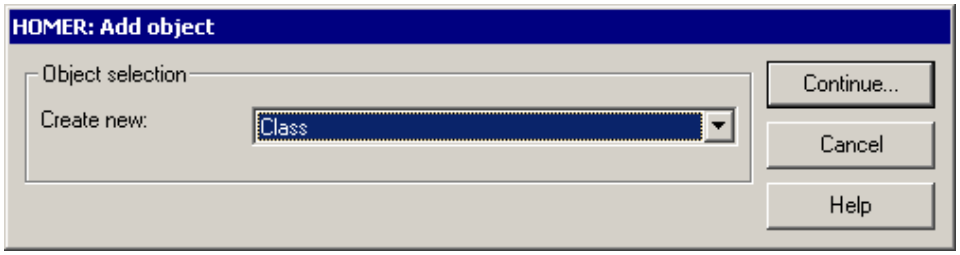


Figure 478: HOMER: add object

Select from the list **"Create new"** the object "class" and then click on the button "Continue". The window "HOMER: Create class" (see fig. 479) will appear.

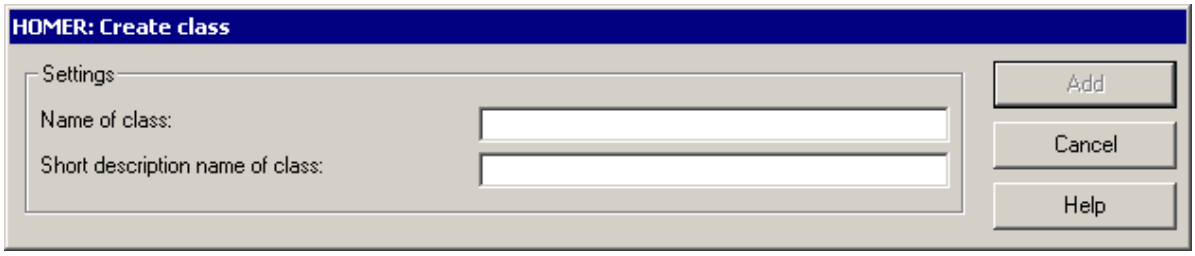


Figure 479: HOMER: add class

Enter in the field **"Name of class"** the name of the class of which objects you want to administrate in HOMER.

Enter in the field **"Short name of class"** any short name for the class of which objects you would like to administrate in HOMER.

ATTENTION: The name of the short name must be clear.

Inside the class, tables (see chap. 13.2.2.7, p. 776) and attributes (see chap. 13.2.2.10, p. 777) can be now created.

13.2.2.2 Editing Class Configuration

To change the configuration of a class, select the class to change from the object selection of the window "HOMER: Object settings" (see fig. 477) and then click on the button "Edit". The window "HOMER: Edit class" (see fig. 480) will appear.

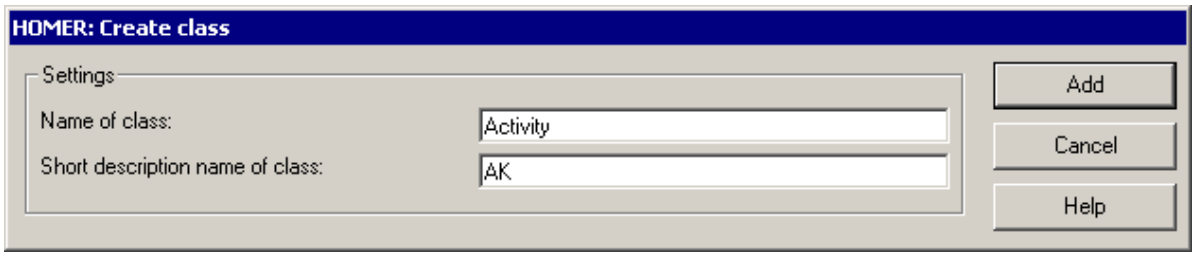


Figure 480: HOMER: edit class

Enter in the field **"Name of class"** the name of the class of which objects you want to administrate in HOMER.

Enter in the field **"Short name of class"** the short name of the class of which objects you want to manage in HOMER.

ATTENTION: The name of the short term must be explicit.

13.2.2.3 Deleting Classes

To remove **one** class, first select it in the window "HOMER: Object settings" (see fig. 477) and then click on the button "delete".

To remove **all** classes, select in the window "HOMER: object settings" (see fig. 477) the object "Classes" and then click on the button "delete".

An appropriate security query will appear before the deletion of the selected classes.

13.2.2.4 Adding Attribute Profile Classes

Hint: The definition of the attribute profile can be found in the "Attribute profile" (see chap. 12., p. 100) section.

To create an attribute profile class, click on the object "Attribute profile classes" in the object list of the window "HOMER: object settings" (see fig. 477) and then on the button "Add". The window "HOMER: Add object" (see fig. 481) will appear.

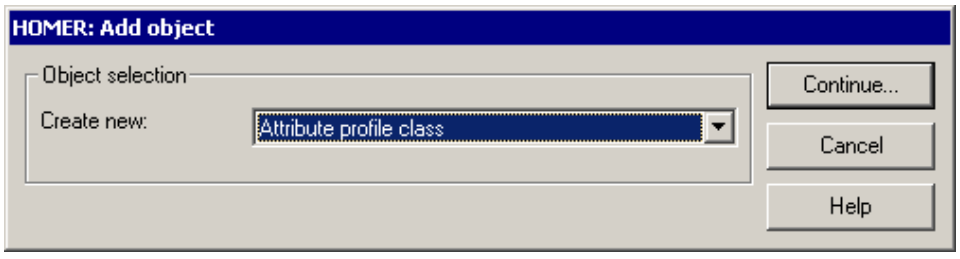


Figure 481: HOMER: add object

Select the object "Attribute profile class" from the list **"Create new"** and then click on the button "Continue". The window "HOMER: Create attribute profile class" (see fig. 482) will appear.

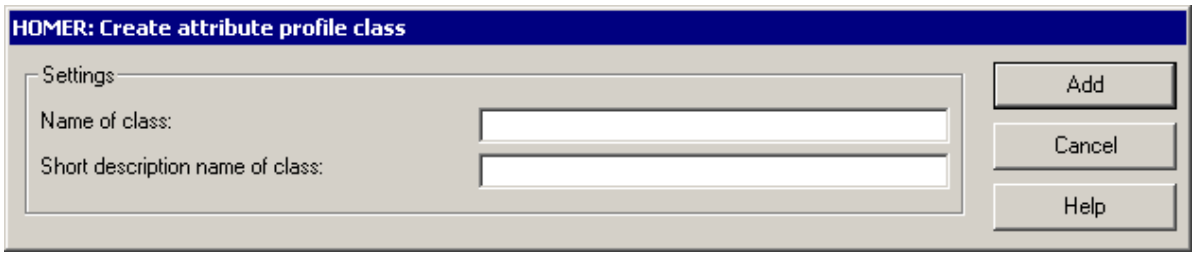


Figure 482: HOMER: create attribute profile class

In the field **"Name of class"** enter the name of the attribute profile class of the objects you want to administrate in HOMER.

Enter in the field **"Short name of class"** any short name for the attribute profile class of which objects you would like to administrate in HOMER.

ATTENTION: The name of the short description must be clear.

13.2.2.5 Editing Attribute Profile Classes Configuration

To change the configuration of an attribute profile class, select the attribute profile class to be changed in the object selection of the window "HOMER: object settings" (see fig. 477) and then click on the button "Edit". in the window "HOMER: Edit attribute profile class" (see fig. 483).

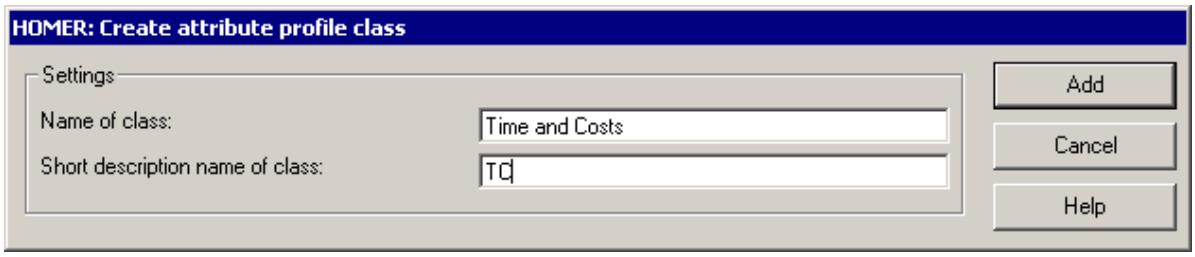


Figure 483: HOMER: edit attribute profile class

In the field **"Name of class"** enter the name of the attribute profile class of which you want to administrate in HOMER.

In the field **"Short name of class"** enter the short name for the attribute profile class of which objects you want to administrate in HOMER.

ATTENTION: The name of the short description must be clear.

13.2.2.6 Deleting Attribute Profile Classes

In order to delete **one** attribute profile class, first select it in the window "HOMER: object settings" (see fig. 477) and then click on the button "delete".

To remove **all** attribute profile classes, select the object "Attribute profile classes" in the window "HOMER: object settings" (see fig. 477) and then click on the button "delete".

An appropriate security query will be displayed before the deletion of the selected attribute profile classes.

13.2.2.7 Adding Records

Hint: You can find the definition of a record, or of a record attribute in the "Record attribute" (see chap. 13., p. 104).

To create a record (for ADOxx attributes of the type "RECORD"), select in the window "HOMER: object settings" (see fig. 477) a record object of a class or an attribute profile class, for which you want to add this record and then click on the button "Add". The window "HOMER: add object" (see fig. 484) will appear.

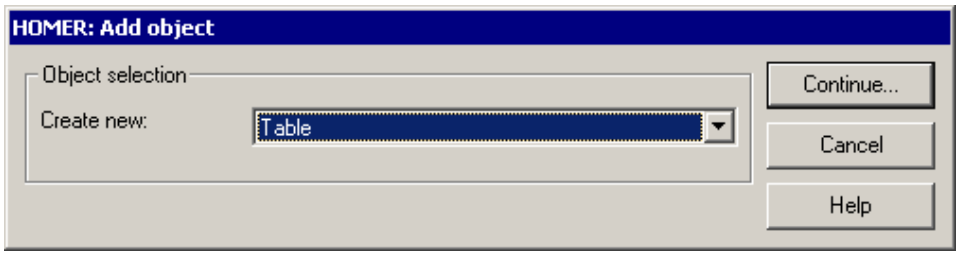


Figure 484: HOMER: add object

Select the object "Record" in the list **"Create new"** and then click on the button "Continue". The window "HOMER: create ADOxx record" (see fig. 485) will appear.

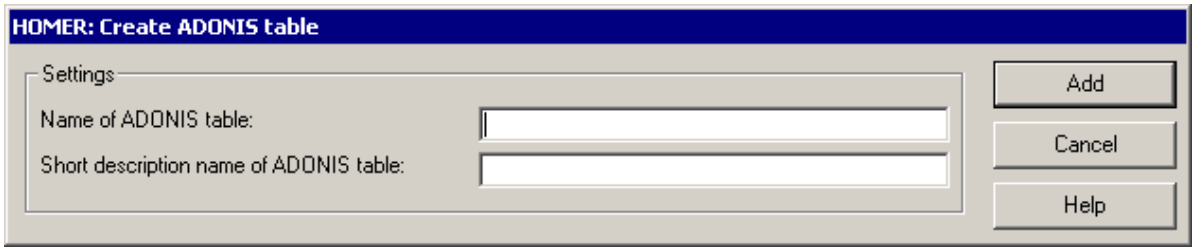


Figure 485: HOMER: add record

Enter in the field **"Name of ADOxx record"** the name of the record of which attributes you would like to administrate in HOMER.

Hint: The name of the attribute of the class, which contains the reference to the record class must be specified.

Enter in the field **"Short name of the ADOxx record"**, the short name for this record.

13.2.2.8 Editing Records

To change the configuration of a record (for ADOxx attribute of type "RECORD"), select in the window "HOMER: object settings" (see fig. 477) the records to change and then click on the button "Edit". The window "HOMER: edit ADOxx record" (see fig. 486) will appear.

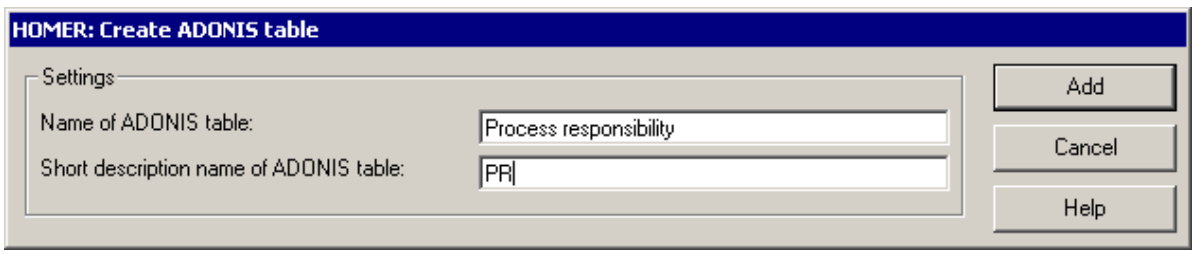


Figure 486: HOMER: edit record

Enter in the field **"Name of ADOxx record"** the name of the selected record.

Hint: The name of the attribute of the class, which contains the reference to the record class, must be specified.

Enter in the field**"Short name of ADOxx record"** the short name for this record.

13.2.2.9 Deleting Records

To remove **one** record, first select it in the window "HOMER: object settings" (see fig. 477) and then click on the button "Delete".

To remove **all** records of a class or attribute profile classes, first select the object "Records" of the appropriate class or attribute profile class in the window "HOMER: object settings" (see fig. 477) and then click on the button "Delete".

An appropriate security query will be displayed before the deletion of the selected records.

13.2.2.10 Adding Attributes

Hint: You can find the definition of an (object) attribute in the section "Terms and connections" (see chap. 1., p. 32).

To create an attribute or a model attribute, select a model attribute , an attribute of a class , an attribute profile class or a record and then click on the button "Add" in the window "HOMER: object settings" (see fig. 477). The window "HOMER: add object" (see fig. 487) will appear.

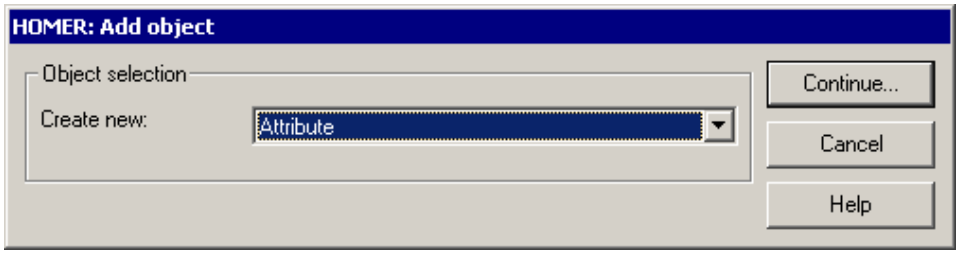


Figure 487: HOMER: configurate attributes

Select in the list **"Create new"** the object "Attribute" and then click on the button "Continue". The window "HOMER: create attribute " (see fig. 488) will be displayed.

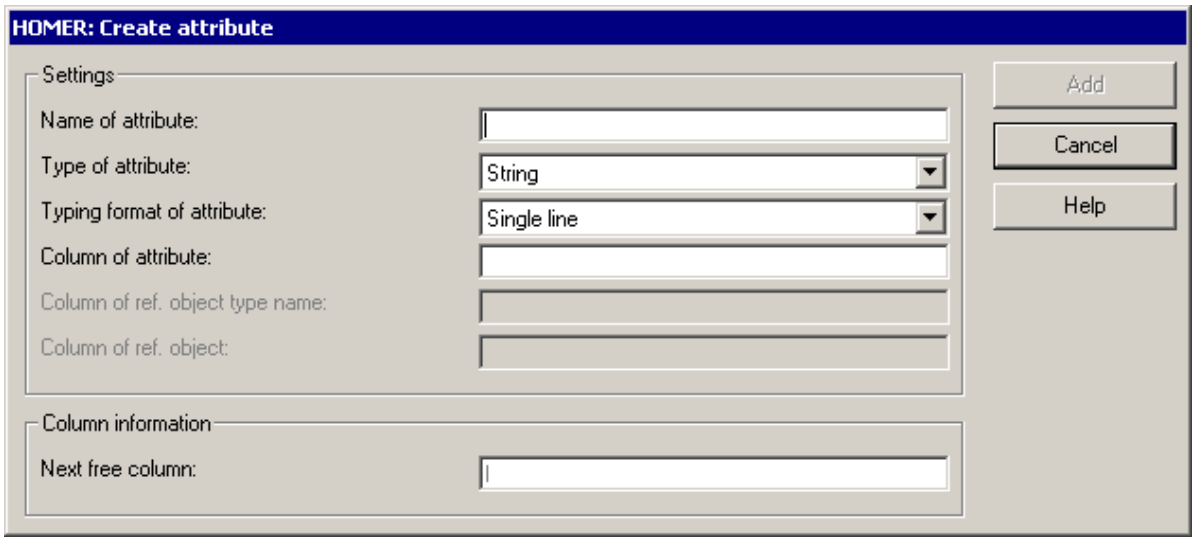


Figure 488: HOMER: create attribute

Enter in the field **"Name of attribute"** the name of the attribute of what data you want to acquire in HOMER.

Select the type of the current attribute (see chap. 13.2.2.11, p. 780) in the list **"Type of attribute"**.

Hint: The following fields and pull-down menus appear depending on the selected attribute type and task format. The figure below illustrates the membership of the attributes and their task possibilities

Hint: In the following fields, the abbreviation "ref." stands for "referenced".

Attribute type	Attribute format	Attribute value column	How breaking point	Separate sign	Column of referenced model	Column of referenced model	Column of referenced object name	Column of ref. attribute profile	Default name of called program	Column of called file	Column of the form name	Form column
String	One line	X										
	Multiple lines	X	X									
Number	Integer	X										
	Integer	X										
Time	ADONIS (yy:dd:hh:mm:ss)	X										
	Year (Floating point number)	X										
	Day (Floating point number)	X										
	Hour (Floating point number)	X										
	Minute (Floating point number)	X										
	Second (Floating point number)	X										
	mm:ss	X										
	[m]:ss	X										
	[h]:mm	X										
	[h]:mm:ss	X										
Enumeration		X										
Enumeration list		X		X								
Model reference					X	X						
Object reference					X	X	X	X				
Attribute profile reference									X			
Programm call	(fixed program name)									X		X
	(variable program name)										X	X
Discrete random generator											X	X
Expression	One line	X										
	Multiple lines	X	X									
Date	ADONIS (yy:mm:dd)	X										
Date and time	ADONIS (yy:dd:hh:mm:ss)	X										

Figure 489: Membership of the attributes and their task possibilities

Explanation:

When choosing attribute type "string", the user has the possibility to choose between input formats "single row" and "many rows" as well as the ability to enter columns of attribute values and line break indicators.

Hint: The field **"Next free column"** contains the name of the next free column (i.e. not assigned) - starting from the end of the record.

Hint: The field **"Next free column"** is shrunk when creating model attributes, since they are not entered in columns but in cells.

ATTENTION: All free columns between two assigned columns will be ignored.

Select in the field **"Typing format of attribute"** the format of the current attribute (see chap. 13.2.2.12, p. 781).

Enter in the field **"Column of attribute"** the name of the column, in which the current attribute should be.

ATTENTION: If the entered column is not available, you have the possibility to move all attributes concerned to the right. Make sure for attributes with multiple columns, that you can separate the neighbouring columns of the attribute.

Enter in the field **"Separator"** a separator sign to separate possible enumeration values of the attribute.

Enter in the field **"Column of ref. model type name"** the name of the column in which the model type name of the referenced model/object should be.

Enter in the field **"Column of ref. model"** the name of the column in which the model name of the referenced model/object should be.

Enter in the field **"Column of ref. object type name"** the name of the column in which the name of the class of the object reference should be.

Enter in the field **"Column of ref. object name"** the name of the column in which the name of the object of the object reference should be.

Enter in the field **"Column of ref. attribute profile"** the name of the column in which the name of the referenced attribute profile should be.

Enter in the field **"Standard name of called program"** the name of the program that should be used by default for the called file.

Hint: If you have selected the type "program call" in the list "type of attribute" and this field is not active, you must activate the radio button in front of the field name.

Enter in the field **"Column of called program"** the name of the column in which the program, that should be used with the called file, should be.

Hint: If you have selected the type "program call" in the list "type of attribute" and this field is not active, you must activate the radio button in front of the field name.

Enter in the field **"Column of called file"** the name of the column in which the called file should be.This field is only relevant for the attribute type "program call".

Hint: The file to call has to be entered with its path.

Enter in the field **"Column of domain name"**, the name of the column in which the name of each variable domain should be. This field is only relevant for the attribute type "Discrete variable setting".

Enter in the field **"Column of domain value"** the name of the column in which the variable domain should be.

13.2.2.11 Attribute Types in HOMER

The following attribute types will be directly supported by HOMER:

String means that you can enter any text of one or several-lines in this attribute of ADOxx.

Number means that you can enter whole numbers or floating points in this attribute of ADOxx.

Time means that you can enter times in the ADOxx time format (YY:DDD:HH:MM:SS) in this attribute.

Enumeration means that by using this attribute of ADOxx you can select **one** value from a list, click on **one** check box or mark **one** value over a radio button.

Enumeration list means that you can select **several** values from a list over this attribute of ADOxx.

Model reference means that you can reference one or more models over this attribute of ADOxx.

Object reference means that you can reference one or more objects over this attribute of ADOxx.

Attribute profile reference means that you can reference one or more attribute profiles over this attribute of ADOxx.

ATTENTION: Attribute profile classes will be supported by ADOxx from version 3.5. If you have **not** selected the option "ADOxx version 3.5x" in the pull-down menu "ADOxx Version" in the general model parameters (see chap. 13.2.1, p. 769), then you cannot create attributes of type "Attribute profile reference".

Program call means that you can reference one or more files and their programs over this attribute in ADOxx.

Discrete random generator means that you can produce the **discrete** random generator for a branching of the process path over this attribute in ADOxx.

Expression means that in ADOxx you can enter values with flexible formulas within this attribute.

Date means that in ADOxx you can enter date values using the ADOxx date format (YY:DDD:HH:MM:SS) within this attribute.

Date and time means that in ADOxx you can enter time values using the ADOxx date and time format (YYYY:MM:DD hh:mm:ss) within this attribute.

Hint: If you require an attribute type not specified here, it is possible to define it as a string (text) and to enter the required ADL syntax manually in the acquisition table. Using an Excel macro (which has to be created), you can change the format of the attribute fields to the appropriate value, in order to create the partially large-scale ADL syntax in a proper way.

Hint: Further information for the HOMER attribute types can be found in chapter "ADOxx attribute types" (see chap. 7., p. 733)

13.2.2.12 Attribute Formats in HOMER

The following formats are available depending on the selected attribute type:

1. ***string***
   - **Single line**

     Set this format for any single line text.

     All symbols will be edited except line breaks.

     A maximum of 255 symbols are allowed in HOMER
   - **Multi line**

     Set this format for any multi line text.

     All symbols will be edited.

     A maximum of 255 symbols are allowed per cell in HOMER. If there are more symbols used, then the below-standing cell will be used.

   ATTENTION: **Line** breaks will be changed to **cell** breaks, i. e. the next deeper cell will be used.

   Note: The new line indicator is useful to differentiate, whether a new cell has started, because there is a line break in the original text or because the maximum number of symbols allowed per cell has been exceeded. If the new line indicator has the value "1", then a new line has started, because there is a line break in the original source attribute at this position. If the value is "0", a new cell has started, because the maximum number of symbols has already been reached.
2. ***Number***
   - **Floating point number**

     Set this format for floating point numbers (e. g.: 0.9 or 1.0).

     Only numbers and one dot will be accepted.

     A maximum of 255 symbols are allowed in HOMER.
   - **Integer (whole numbers)**

     Set this format for floating point number (e. g.: 4 or 17).

     Only numbers will be edited.

     A maximum of 255 symbols are allowed in HOMER.
3. ***Time***
   - **ADOxx (yy:ddd:hh:mm:ss)**

     Set this format for times you want to enter directly in the ADOxx time format.

     Enter the time value in the format "yy:ddd:hh:mm:ss" (Year:Days:Hours:Minutes:Seconds). Three days, five hours and ten seconds must be entered as follows: 00:003:05:00:10.

     A maximum of 15 symbols are allowed in HOMER.
   - **Year (floating point number)**

     Set this format, if you want to enter the time calculated in years as a floating point number. Three days, five hours and ten seconds must be entered as follows: 0.0087902714358193810

     Only numbers and one dot will be accepted.

     A maximum of 255 symbols are allowed in HOMER.

- **Day (floating point number)**

  Set this format, if you want to enter the time calculated in days as a floating point number. Three days, five hours and ten seconds must be entered as follows: 3,208449074

  Only numbers and one dot will be accepted.

  A maximum of 255 symbols are allowed in HOMER.

- **Hours (floating point number)**

  Set this format, if you want to enter the time calculated in hours as a floating point number. Three days, five hours and ten seconds must be entered as follows: 77,0027778

  Only numbers and one dot will be accepted.

  A maximum of 255 symbols are allowed in HOMER.

- **Minutes (floating point number)**

  Set this format, if you want to enter the time calculated in minutes as a floating point number. Three days, five hours and ten seconds must be entered as follows: 4620,16667

  Only numbers and one dot will be accepted.

  A maximum of 255 symbols are allowed in HOMER.

- **Seconds (floating point number)**

  Set this format, if you want to enter the time calculated in seconds as a floating point number. Three days, five hours and ten seconds must be entered as follows: 277210

  Only numbers and one dot will be accepted.

  A maximum of 255 symbols are allowed in HOMER.

- **mm:ss**

  Set this format, if all the times for this attribute are smaller than one hour and you want to enter minutes and seconds as whole numbers.

  Enter the minutes (two-digit) followed by a colon and the seconds (two-digit). Five minutes and ten seconds must be entered as follows: 05:10.

  A maximum of 5 symbols is allowed in HOMER.

- **[m]:ss**

  Set this format, if all the times for this attribute are **not** smaller than one second and you want to enter minutes and seconds as whole numbers.

  Enter the minutes followed by a colon and the seconds (two-digit). Three days, five hours and ten minutes must be entered as follows: 4630:13

ATTENTION: Note that for this format all times greater than 60 minutes must be calculated in minutes, i.e. three days corresponds for instance to 4620 minutes.

  A maximum of 255 characters are allowed in HOMER.

- **[h]:mm**

  Set this format, if all the times for this attribute are **not** smaller than one minute and you want to enter hours and minutes as whole numbers.

  Enter the hours followed by a colon and the minutes (two-digit). Three days, five hours and ten minutes must be entered as follows: 77:10

ATTENTION: Note that for this format all times greater than 24 hours must be calculated in hours, i.e. three days corresponds for instance to 72 hours.

A maximum of 255 characters are allowed in HOMER.

- **[h]:mm:ss**

  Set this format, if you want to enter hours, minutes and seconds as whole numbers.

  Enter the hours followed by a colon and the minutes (two-digit), followed by a colon and the seconds (two-digit). Three days, five hours, ten minutes and thirteen seconds must be entered as follows: 77:10:13

ATTENTION: Note that for this format all times greater than 24 hours must be calculated in hours, i. e. three days correspond for instance to 72 hours.

A maximum of 255 characters are allowed in HOMER.

Hint: If you require an attribute type not specified here, it is possible to define it as a string (text) and to enter the required ADL syntax manually in the acquisition table. Using an Excel macro (which has to be created), you can change the format of attribute fields to the appropriate value, in order to create the large ADL syntax in the correct way.

13.2.2.13 Modifying Attributes

To change the configuration of an attribute, select the attribute to be changed in the window "HOMER: object settings" (see fig. 477) and then click on the button "Edit". The window "HOMER: edit attribute" (see fig. 490) will appear.

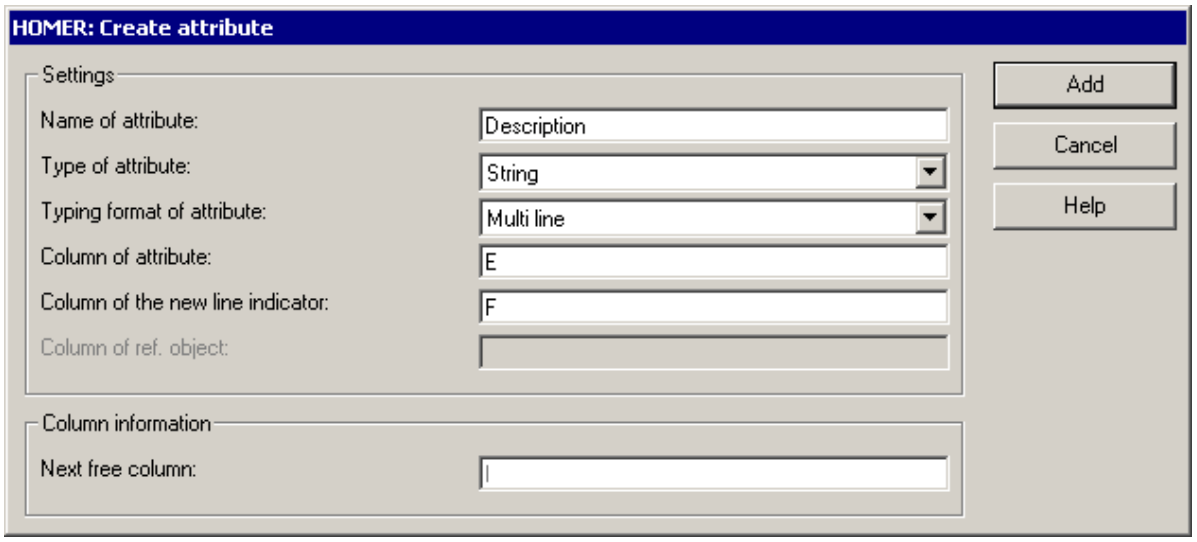


Figure 490: HOMER: edit attributes

Enter in the field **"Name of attribute"** the name of the attribute of which data you want to acquire in HOMER.

Select from the list **"Type of attribute"** the type of the current attribute (see chap. 13.2.2.11, p. 780).

Hint: The following fields and pull-down menus appear depending on the selected attribute type and task format. The figure below illustrates the membership of the attributes and their task possibilities

Hint: In the following fields, the abbreviation "ref." stands for "referenced".

Attribute type	Attribute format	Attribute value column	Row breaking point	Separate sign	Column of referenced model n.	Column of referenced model	Column of referenced object name	Column of refrenced object	Column of ref. attribute profile	Default name of called program	Column of called file	Form column
String	One line	X										
	Multiple lines	X	X									
Number	Integer	X										
	Integer	X										
Time	ADONIS (yy:dd:hh:mm:ss)	X										
	Year (Floating point number)	X										
	Day (Floating point number)	X										
	Hour (Floating point number)	X										
	Minute (Floating point number)	X										
	Second (Floating point number)	X										
	mm:ss	X										
	[m]:ss	X										
	[h]:mm	X										
	[h]:mm:ss	X										
Enumeration		X										
Enumeration list		X		X								
Model reference					X	X						
Object reference					X	X	X	X				
Attribute profile reference									X			
Programm call	(fixed program name)									X		X
	(variable program name)										X	X
Discrete random generator												
Expression	One line	X										
	Multiple lines	X	X									
Date	ADONIS (yy:mm:dd)	X										
Date and time	ADONIS (yy:dd:hh:mm:ss)	X										

Figure 491: Membership of the attributes and their task possibilities

Explanation:

When selecting the attribute type "String" you have the choice of two input formats "Single line" and "Multi line" as well as the option to define the columns for the attribute value and the line-break indicator.

Hint: The field **"Next free column"** contains the name of the next available (i.e. not already used) column.

Hint: The field **"Next free column"** is not available when creating model attributes, as these are not maintained in columns but in cells.

ATTENTION: Free columns between two already used columns will be ignored.

Select in the field **"Format of attribute"** the format of the current attribute (see chap. 13.2.2.12, p. 781).

Enter in the field **"Column of attribute"** the name of the column, in which the current attribute should be.

ATTENTION: If the entered column is not available, you have the possibility to move all attributes concerned to the right. Insure that for attributes with multiple columns, you can separate the neighbouring columns of the attribute.

Enter in the field **"Separator"** the character used to separate the possible enumeration values of the attribute.

Enter in the field **"Column of ref. model type name"** the name of the column in which the model type name of the referenced model/object should be.

Enter in the field **"Column of ref. model"** the name of the column in which the model name of the referenced model/object should be.

Enter in the field **"Column of ref. object type name"** the name of the column in which the name of the class of the object reference should be. This field is only relevant for the attribute type "object reference".

Enter in the field **"Column of ref. object name"** the name of the column in which the name of the object reference should be.

Enter in the field **"Column of ref. attribute profile"** the name of the column in which the name of the referenced attribute profile should be.

Enter in the field **"Standard name of called program"** the name of the program, which should be used by default for the called file.

> **Hint:** If you have selected the type "program call" in the list "type of attribute" and this field is not active, you must activate the radio button in front of the field name.

Enter in the field **"Column of called program"** the name of the column in which the program, that should be used with the called file, should be.

> **Hint:** If you have selected the type "program call" in the list "type of attribute" and this field is not active, you must activate the radio button in front of the field name.

Enter in the field **"Column of called file"** the name of the column in which the called file should be.

> **Hint:** The file to call has to be entered with its path.

Enter in the field **"Column of domain name"**, the name of the column in which the name of each variable domain should be.

Enter in the field **"Column of domain value"** the name of the column in which the variable domain should be.

13.2.2.14 Deleting Attributes

To remove **one** attribute from HOMER, first select it in the window "HOMER: Object settings" (see fig. 477) and then click on the button "Delete".

To remove **all** attributes from HOMER, select in the window "HOMER: object settings" (see fig. 477) the object "Attributes" of any object and then click on the button "Delete".

Before deleting the selected scenario, an appropriate security query will appear.

13.2.2.15 Adding Model Attribute Classes

To create a model attribute class, click on the object "Model attribute class" in the object list of the window "HOMER: object settings" (see fig. 477) and then on the button "Add". The window "HOMER: add object" (see fig. 478) will appear.

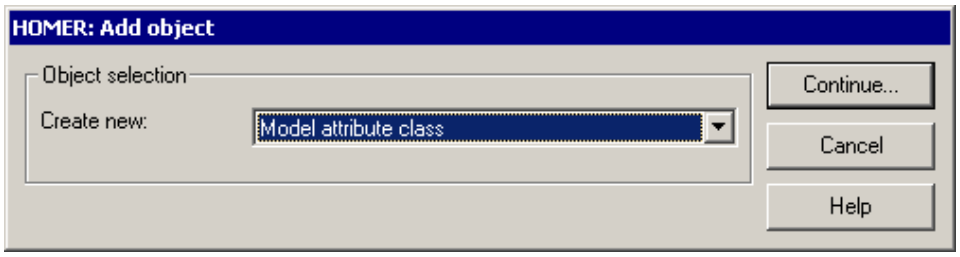


Figure 492: HOMER: add object

Select from the list **"Create new"** the object "model attribute class" and then click on the button "Continue". The window: "HOMER: create model attribute class" (see fig. 493) will appear.

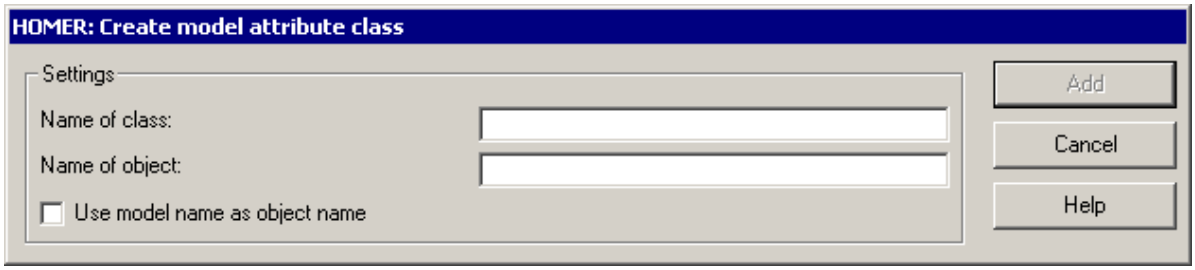


Figure 493: HOMER: create model attribute classes

Enter in the field **"Name of class"** the name of the class of the objects you would like to administrate in HOMER.

Enter in the field **"Name of object"** the name of the object to create which should contain the value for the model attribute.

In case you want to use, for the object name, the name of the model where the current object is, click on the option **"Use model name as object name"**.

13.2.2.16 Editing Model Attribute Class Configuration

To change the configuration of a model attribute class, select the model attribute class you want to change in the object selection of the window "HOMER: object settings" (see fig. 477) and then click on the button "Edit" in the window "HOMER: edit model attribute class" (see fig. 494) will appear.

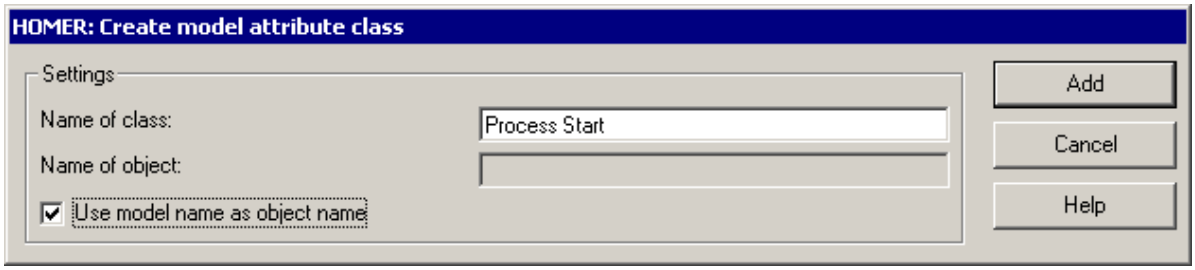


Figure 494: HOMER: edit model attribute class

Enter in the field **"Name of class"** the name of the model attribute class of which objects you would like to administrate in HOMER.

Change in the field **"Name of object"** the name for the object you want to create, which should contain the value for the model attribute .

In case you want to use the name of the model where the current object is as the object name, click the option **"Use model name as object name"**.

13.2.2.17 Deleting Model Attribute Classes

To remove **one** model attribute class from HOMER, first select them in the window "HOMER: Object settings" (see fig. 477) and then click on the button "Delete".

To remove **all** model attribute classes from HOMER, select in the window "HOMER: object settings" (see fig. 477) the object "Attributes" of any object and then click on the button "Delete".

Before deleting the selected scenario, an appropriate security query will appear.

13.2.2.18 Adding Model Attributes

Hint: The definition of the model attributes can be found in "the terms and context" (see chap. 1., p. 32) chapter.

To create a model attribute, select a model attribute and click on the button "Add" in the window "HOMER: object settings" (see fig. 477). The window "HOMER: add object": (see fig. 495) will appear.

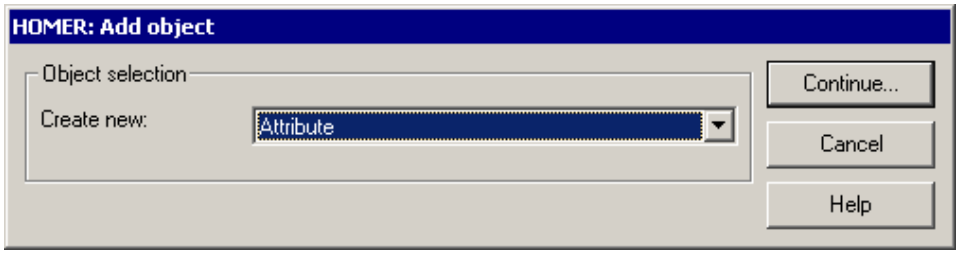


Figure 495: HOMER: configurate attributes

Select from the list **"Create new"** the object "Attribute" and then click on the button "Continue". The window: "HOMER: create attribute" (see fig. 493) will appear.

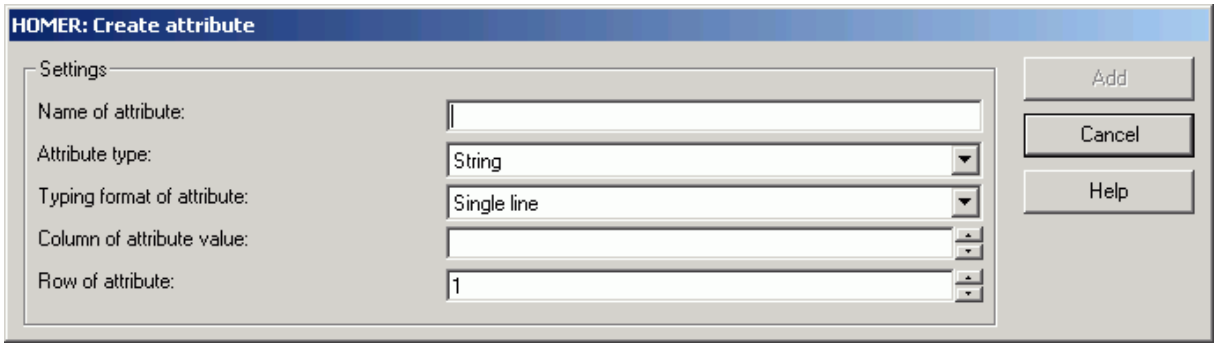


Figure 496: HOMER: create model attribute

Enter in the field **"Name of attribute"** the name of the model attribute you want to acquire in HOMER.

Enter in the list **"Type of attribute"** the type of the current attribute (see chap. 13.2.2.11, p. 780).

Hint: The following fields and pull-down menus appear depending on the selected attribute type and will be listed in the order of the attribute type dependence, i.e. first the field of the attribute type "String" in the order of its appearance and then those of the attribues of type "number" etc.

Select in the field **"Typing format of attribute"** the format of the current attribute (see chap. 13.2.2.12, p. 781). This field is only relevant for attributes of types "String", "Number" and "Time".

Enter in the field **"Column of attribute"** the name of the column for the current attribute.

Enter in the field **"Row of attribute"** the number of the row for the current attribute.

13.2.2.19 Changing Model Attributes

To change the configuration of a model attribute, first select it in the window "HOMER: object settings" (see fig. 477) and then click on the button "Edit". the window "HOMER: edit attribute" (see fig. 497) will appear.

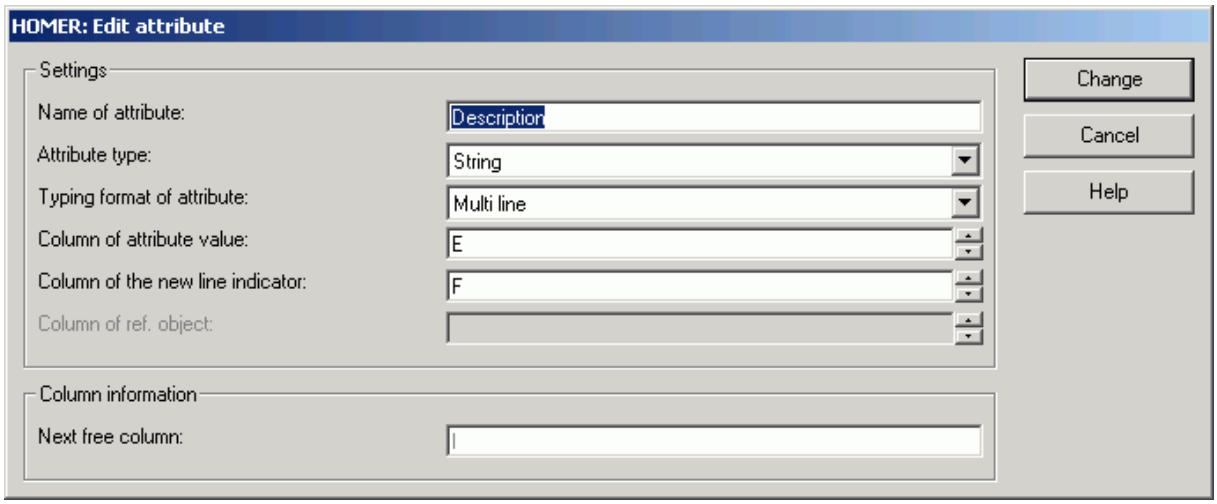


Figure 497: HOMER: edit model attributes

Enter in the field **"Name of attribute"** the name of the model attribute value you want to acquire in HOMER.

Select from the list **"Type of attribute"** the type of the current attribute (see chap. 13.2.2.11, p. 780).

> **Hint:** The following fields and lists appear depending on the selected attribute type and will be listed in the order of the attribute type dependence, i.e. first the field of the attribute type "String" in the order of its appearance and then those of the attribute type "number" etc.

Select in the field **"Typing format of attribute"** the format of the current attribute (see chap. 13.2.2.12, p. 781). This field is only relevant for attributes of type "String", "Number" and "Time".

Enter in the field **"Column of attribute value"** the name of the column, in which the current attribute should be.

Enter in the field **"Row of attribute"** the number of the row, in which the current model attribute should be.

13.2.2.20 Deleting Model Attributes

To remove **one** model attribute from HOMER, first select it in the window "HOMER: object settings" (see fig. 477) and then click on the button "Delete".

To remove **all** model attributes from HOMER, select in the window "HOMER: object settings" (see fig. 477) the object "Attribute" for any object and then click on the button "Delete".

Before deleting the selected scenario, an appropriate security message will appear.

14. ADOxx Query Language (AQL)

The ADOxx query language AQL (**A**DONIS **Q**uery **L**anguage) enables you to run queries on Business Process Models or Working Environment models. The result of an AQL expression is a set of objects or connectors which meet the search criteria you specified.

AQL is used in the following places in the ADOxx Modelling Toolkit:

- when Querying (see chap. 3.1, p. 306) the notebook information of any modelling object in the analysis component (see chap. 3., p. 306)

  (See examples 1, 2, 8, 12, 13, 14, 15 and 33 for **BP models** or examples 3, 4, 5, 6, 7, 9, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27 and 28 for **WE models**.)

- when linking Business Process Model and Working Environment models in the Modelling Toolkit (see chap. 4., p. 17) by the attribute "Performer" - in class "Activity". The Performer Assignment (see chap. 2.2.12.14, p. 222) is needed for the Simulation (see chap. 4., p. 332) (Capacity Analysis (see chap. 4.3, p. 354) and Workload Analysis (see chap. 4.4, p. 367)).

  (See examples 7, 12, 15, 16, 22, 23, 27, 28, 29, 30, 31 and 34.)

- when allocating resources to activities or to performers in the Modelling Toolkit (see chap. 4., p. 17). The Resource Allocation (see chap. 2.2.12.15, p. 227) takes place in the business process models with the help of the attribute "Selection" in the class "resource".

  (See examples 36 and 37.)

- when defining a selection criterion for the Global Change of attributes (see chap. 2.2.14, p. 243) in models in the modelling component (see chap. 2., p. 126).

  (See examples 1, 2, 8, 12, 13, 14, 15 and 33 for **BP Models** or examples 3, 4, 5, 6, 7, 9, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27 and 28 for **WE Models**.)

- when entering selection criteria for a model search according to model attributes in the modelling component (see chap. 2., p. 126).

  (See example 32.)

The query language AQL contains static elements which can be evaluated at any time in the Analysis Component. Static elements are allowed to be assigned in the simulation only if one or more performers are removed from the used Working Environment model. Dynamic elements are evaluated during the capacity and workload analysis via the performer assignment. Additionally the dynamic elements allow the access to variable values or performers of previous activities during the simulation.

> **Hint:** All explanations and descriptions refer to models based on the **ADOxx standard application library** (see chap. 16., p. 802).

14.1 Syntax and Semantics of AQL

Extended Backus Naur Form (EBNF) Notation

The EBNF notation specifies by so-called (production) rules which expressions meet the requirements of the AQL syntax.

Non-terminal symbols are included in <...> and serve to formulate the rules (see chap. 14.2, p. 792). Each rule starts with a non-terminal symbol, followed by "::=" and the symbol's definition. The non-terminal symbols occurring in the definitions are defined in other rules.

Terminal symbols (= symbols which cannot be split up further) are included by inverted commas and are interpreted directly.

The symbols {...}, [...] and | serve to formulate rules in a more compact form.

Summary of the EBNF notation:

<...> non-terminal symbol

'...' terminal symbol

{...} arbitrary number of iterations (even 0-times)

[...] optional (0- or 1-time)

| alternative

Terminal symbols and key words

The syntax of AQL requires that names of classes, objects, relations, attributes, variables and alphanumerical constants are denoted by inverted commas. White-spaces of any kind (blank, tabulator or return) are allowed in any place where they make sense.

<Class> ::= The name of a class or relation. The classes and relations such as "Process start", "Activity", "Decision", "Subsequent" and so on are available to the user via the modelling panel in the model editor, so that he can design business process models. The model, that contains a class or relation, arises from the context of the analysed or simulated models. If another model is desired (i.e. because a class or relation should be referenced in particular model), it has to be entered explicitly (through names and model types):

<Class>':'<Model name>':'<Model type>

<Object> ::= The name of an object within a concrete business process or working environment model. "Write letter", for example, could be the name of an object of the class "Activity". Should objects be ambiguous within a model, the name of the class has to be appended to the object's name as in <Object>':'<Class>.The model, that contains a class or relation, arises from the context of the analysed or simulated models. If another model is desired (i.e. because a class or relation should be referenced in particular model), so it has to be entered explicitly (through names and model types):

<Object>':'<Model name>':'<Model type>, or

<Object>':'<Class>':'<Model name>':'<Model type>

<Relation> ::= The name of a relation, e.g. "Subsequent" in business process models or "Has role" in working environment models.

<Attribute> ::= The name of an attribute, e.g. the class "Activity" has - among others - the attribute "Execution time".

<Operator> ::= '>' | '>=' | '=>' | '=' | '<=' | '=<' | '<' |'!='| 'like' | 'unlike' are comparison operators. 'like' and 'unlike' are used for alphanumerical signs and allow the use of wild cards (? and/or *) in symbol chains. The other operators are used for comparing numerical values.

The operator 'contains' is available where complex attribute values (e.g. intermodel-references) can be interrogated. The operator represents on the one hand, in the sense of a set operation, something comparable with a set inclusion (subset feature), on the other hand the adherence will be expressed this way in the sense of a subtext. However this operator is not an AQL key word!

<Value> ::= <Constant> | '!' <Variable> | '?' <Attribute>

<Constant> ::= A constant expression such as 1000, Yes or others.

Logical Operators serve to link two (or more) AQL expressions:

<Logical Operator> ::= 'AND' | 'OR' | 'DIFF'

- 'AND' links two AQL expressions. The result is the intersection set of the two expressions, i.e. the result consists of all those objects which are both the result of the first and the result of the second expression.
- 'OR' links two AQL expressions. The result is the union set of the two expressions, i.e. the result consists of all objects which are either the result of the first or the result of the second expression or both.
- 'DIFF' links two AQL expressions. The result is the difference of the two expressions, i.e. the result consists of all those objects which are the result of the first expression, but not the result of the second expression.

'AND' links more strongly than 'OR' and 'OR' more strongly than 'DIFF', i.e. the expression "a DIFF b OR c AND d" is evaluated correctly like this: a DIFF (b OR (c AND d)).

14.2 Rules for formulating AQL expressions

The following rules are used while formulating AQL expressions (Analysis Component and static performer assignment of the Simulation Component) or dynamic performer assignment (Simulation Component):

Rule 1: **AQL expression> ::= <AQL expression> {'AND' | 'OR' | 'DIFF' <AQL expression>}**

Each expression can be linked to one or more expressions by logical operators.

See examples 16, 20, 29, 30, 31, 32 and 34.

Rule 2: **<AQL expression> ::= '(' <AQL expression> ')'**

Expressions can have parentheses.

See examples 12, 16, 20, 22, 29, 30, 31, 32 and 34.

Rule 3: **<AQL expression> ::= '{' <object> [',' <object>] '}'**

The result is a set of objects. These are not necessarily derived from a class.

See examples 1, 2. As far as the allocation of resources to activities is concerned, see example 36.

Rule 4: **<AQL expression> ::= '<' <class> '>'**

The result is all objects of the class specified.

See examples 3, 4, 5, 17, 18, 19, 20, 21, 23, 27, 28, 31, 33, 39 and 40.

Rule 5: **<AQL expression> ::= <AQL expression> '->' <Class>**

The result is all objects which are of the specified class.

See example 11.

Rule 6: **<AQL expression> ::= <AQL expression> '->' <Relation>**

The result is all objects which are linked as direct targets of the given relation with at least one object from the AQL expression.

See example 6.

Rule 7: **<AQL expression> ::= <AQL expression> '<-' <Relation>**

The result is all objects which are linked as a direct start object of the relation specified with at least one object from the AQL expression.

See examples 7 and 8.

Rule 8: **<AQL expression> ::= <AQL expression> '->>' <Relation>**

The result is all objects which are linked transitively as a target object of the relation specified with at least one object from the AQL expression.

See example 12.

Rule 9: **<AQL expression> ::= <AQL expression> '<<-' <Relation>**

The result is all objects which are linked transitively as a start object of the relation specified with at least one object from the AQL expression.

See examples 9, 10 and 11.

Rule 10: **<AQL expression> ::= <AQL expression> '->' '<' <Relation> '>'**

The results are all connectors of the specified relation which have one of the specified objects as a start object.

See example 13.

Rule 11: **<AQL expression> ::= <AQL expression> '<-' '<' <Relation> '>'**

The results are all connectors of the specified relation which have one of the specified objects as an end point.

See example 14.

Rule 12: **<AQL expression> ::= <AQL expression> '[' <Value> <Operator> <Value> ']'**

The result is all objects of the start query, which attributes fulfil the defined criteria. Constants (numbers, string constants) can only be at the right of the operator. At the left of the operator, there are only attributes or variable references.

See examples 17, 18, 19, 20, 31 and 33.

Note: Queries with variable references as dynamic components in the performer assignment are only allowed in the simulation. See example 16.

Rule 13: **<AQL expression> ::= <AQL expression> '[' <Value> ']' '[' <Value> <Operator> <Value> ']'**

The result is all objects of the start query where their record attribute profile attribute fulfils the defined criteria. The first value is an attribute reference and specifies the name of the record or attribute profile attribute. In the second expression, constants (numbers, string constants) can only be at the right side of the operator. At the left side of the operator, there are only attributes or variable references.

Note: In case of a record attribute, the criteria is always fulfilled, if at least a table row of the record attribute meets the defined criteria.

See example 21 for record attributes and 41 for attribute profile attributes.

Rule 14: **<AQL expression> ::= 'done by' <Activity>**

Performer assignment. The result is the performer who has carried out the activity during a simulation run. This way, you will ensure, that an activity will be carried out by the same performer as defined in the performer assignment for this activity. The dynamic performer assignment is useful only for the simulation and is only possible within a Business Process Model.

See example 15.

Rule 15: **<AQL expression> ::= 'done by' <Variable>**

Performer assignment. This dynamic performer assignment can be used in an intermodel assignment . The result is the performer who has carried out the activity referenced through the variable during a simulation run. To use this construct it is necessary to enter the name of the variable in the attribute "done by" of the referenced activity.

See example 35.

Rule 16: **<AQL expression> ::= 'current performer' '->' <Resources relation>**

Resources assignment. This way you will ensure that the performer who carries out the current activity uses the resources given to him (resources assignment to performer).

See example 37.

Rule 17: **<AQL expression> ::= <AQL expression> '-->' <Attribute>**

The result is all objects which are referenced in the specified attribute of the object.

See example 39.

Rule 18: **<AQL expression> ::= <AQL expression> '-->>' <Attribute>**

The result is all objects which are transitively referenced in the specified attribute of the class.

Rule 19: **<AQL expression> ::=<AQL expression> '<--'**

The result is all objects which refer the specified object.

See example 40.

14.3 AQL Examples

Hint: The AQL expressions in the examples listed below contain line-breaks. These are not necessary, they are just used to improve the readability of the examples.

Example 1

List all the objects "Request accepted?", "Work on request" and "Take holiday request" of a model:

AQL expression:
```
{"Request accepted?","Work on request",
"Take holiday request"}
```

Example 2

List all the objects "Request accepted?", "Work on request" of the model "Holiday request":

AQL expression:
```
{"Request accepted?":"Holiday request"
:"Business Process Model","Work on request"
:"Holiday request":"Business Process Model"}
```

Example 3

List all the activities of a Business Process Model:

AQL expression: `<"Activity">`

Example 4

List all the activities of the model "Holiday request":

AQL expression: `<"Activity":"Holiday request":"Business Process Model">`

Example 5

List all the roles of a Working Environment model:

AQL expression: `<"Role">`

Example 6

List all the roles of the performer "Maier":

AQL expression: `{"Maier"}->"Has role"`

Example 7

List all the performers who belong to the organisational unit "Distribution":

AQL expression: `{"Distibution"}<-"Belongs to"`

Example 8

List all the performers who have the role "Clerk":

AQL expression: `{"Clerk"}<-"Has role"`

Example 9

Display the whole organisation (i.e. all the organisational units), if it is hierarchically structured and if the organisational unit "Management" is in the upper hierarchy level:

AQL expression: `{"Management"}<<-"is subordinated"`

Example 10

List all the objects within a Business Process Model which are connected with the relation "Subsequent":

AQL expression: `<"End"><<-"Subsequent"`

Example 11

List all the objects of the class Activity within a Business Process Model which are connected with the relation "Subsequent":

AQL expression: `<"End"><<-"Subsequent" >"Activity"<`

Example 12

List all the performers who belong to the organisational unit "Distribution" or an organisational unit subordinated to the distribution:

AQL expression: `({"Distribution"}->>"Is subordinated") <- "Belongs to"`

Example 13

List all the connectors of the class "subsequent" within a Business Process Model which start from objects of the class "Decision":

AQL expression: `<"Decision"> -><"Subsequent">`

Example 14

List all the connectors of the class "Has role" of a Working Environment model which lead to the role "Clerk":

AQL expression: `{"Clerk"} <- <"has role">`

Example 15

During the simulation, the activity "Write letter" should be carried out by the same performer, which has executed the activity "Fill form". Enter the following in the attribute "Performer" of the activity "Write letter":

AQL expression (Performer assignment):

```
Has done "Fill form"
```

Example 16

Example of a variable reference

During the simulation, the activity "Control" should be done by different performers depending on the variables "Credit level": 1. for a credit level above 500.000 by the manager of the organisational unit "Credit department", 2. for a credit level under or equal to 500.000 by performers with the role "Clerk".

Enter the following AQL expression for Activity attribute "Performer" of the Activity "Control":

AQL expression (Performer assignment):

```
(({"Clerk"} <- "Has role")
[!"Credit level" <= 500000]) OR
(({"Credit department"} <- "Is manager")
[!"Credit level" > 500000])
```

Example 17

List all the activities with execution time of five minutes:

AQL expression: `<"Activity">[?"execution time" like "00:000:00:05:00"]`

Example 18

List all the activities for which the attribute "Description" contains no entry:

AQL expression: `<"Activity">[?"Description" like ""]`

Example 19

List all the variables that contain the variable type "Float":

AQL expression: `<"Variable">[?"Variable type" like "Float"]`

or: `<"Variable">[?"Variable type" like "G*"]`

(Since the attribute variable type can only be assigned with "Float" or "Enumeration", the first letter is enough for the entry.)

Example 20

List all the activities which are carried out by clerks and for which the activity costs are at least 10:

AQL expression:
```
(<"Activity">[?"Performer" like "*Clerk*"]) AND
(<"Activity">[?"Costs" >= 10])
```

(Only the activities which are carried out by performers who contain the text "Clerk" in exactly this spelling in the role description. "*clerk*" will not work for this query!)

Example 21

List all the objects of the class "Process start", which have entered the value "Major responsible" in any table row of the record attribute "Process ownership" in the attribute "Classification" :

AQL expression:
```
<"Process start">[?"Process ownership"]
[?"Classification" = "Major responsible"]
```

Example 22

List all the performers with the role "Clerk" who do not belong to the organisational unit "Distribution":

AQL expression:
```
({"Clerk"}<-"Has role") DIFF
({"Distribution"}<-"Belongs to")
```

Example 23

List all the performers of a Working Environment model:

AQL expression: `<"Performer">`

Example 24

List all the roles of the performers "Maier", "Sommer" and "Winter":

AQL expression: `{"Maier", "Sommer", "Winter"} -> "Has role"`

Example 25

List all the oganisational units which are managed by Mr. Müller:

AQL expression:
```
{"Müller"} -> "Is manager"
```

Example 26

List all the organisational units, for which the organisational unit "Department" is subordinated:

AQL expression:
```
{"Department"} -> "Is subordinated"
```

Example 27

List all the managers of the organisational unit "Department":

AQL expression:
```
<"Department"> <- "Is manager"
```

Example 28

List all the employees who have a role:

AQL expression:
```
<"Role"> <- "Has role"
```

Example 29

List all the performers who have the role "Clerk" and belong to the organisational unit "Department":

AQL expression:
```
({"Clerk"} <- "Has role") AND
({"Department"} <- "Belongs to")
```

Example 30

List all the performers who have the role "Clerk" and belong to the organisational unit "Org" or who are managers of the organisational unit "Org":

AQL expression:
```
(({"Clerk"} <- "Has role") AND
({"Org"} <- "Belongs to")) OR
({"Org"} <- "Is manager")
```

Example 31

List of all the performers whose name starts with "M" and who's hourly wages are not higher than 20:

AQL expression:
```
(<"Performer">[?"Name" like "M*"]) AND
(<"Performer">[?"Hourly wages" <= 20])
```

Example 32

List all the organisational units and roles which have the entry "Test" in the attribute "Description":

AQL expression:
```
(<"Organizational unit"> OR
<"Role">)[?"Description" = "Test"]
```

Example 33

List all the performers whose name contains six letters and the fifth letter is an "e":

AQL expression:
```
<"Performer">[?"Name" like "????e?"]
```

Example 34

List all the performers who have the role "Clerk" and belong to the organisational unit "Org" or who are manager of the organisational unit "Org", less the performers who have the role "Secretary" or who are manager of this organisational unit:

AQL expression:
```
((({"Performer"} <- "Has role") AND
({"Org"} <- "Belongs to")) OR
({"Org"} <- "Is manager")) DIFF
(({"Secretary"} <- "Has role") OR
({"Org"} <- "Is manager"))
```

Example 35

During the simulation, the Activity "Write letter" must be carried out by the performer who has executed the Activity "Fill form". In the attribute "Done by" of the Activity "Fill form", define e.g. the variable "Maier". In the Activity "Write letter", enter the following expression in the performer assignment:

AQL expression (Performer assignment):
```
Done by "Maier"
```

Example 36

To carry out the current activity, the resource defined in the Working Environment, e.g. a colour printer, must be used. For this Activity enter the following AQL expression in the resource assignment:

AQL expression (resource assignement):
```
{"colour printer"}
```

Example 37

The performer who carries out the current activity must use the reources that are assigned to him, e.g. his PC. Enter the following AQL expression in the resource assignement of this Activity:

AQL expression (resource assignement):
```
Current performer -> "Uses resource"
```

Example 38

All the Business Process Models which have been changed up to the 1st January 2001.

AQL expression:
```
<"Business Process Model">
[?"Last change on" like "01.01.2001*"]
```

Example 39

All objects of the class "Organizational unit" which are refered to in aBusiness Process Model.

AQL expression: `<"Activity"> --> "ORG unit"`

Example 40

All the objects which refer to an object of the class "Role".

AQL expression: `<"Role"> <--`

Example 41

All the performers who are working at least 3 days a week. (The attribute "Days per week" is defined in the attribute profile attribute "Presence" .)

AQL expression: `<"Perfomer"> [?"Presence"][?"Days per week" >= 3]`

15. colour names

The following colour names can be used in ADOxx (see fig. 498):

aliceblue | antiquewhite | aqua | aquamarine | azure | beige | bisque | black | blanchedalmond | blue | blueviolet | brown | burlywood | cadetblue | chartreuse | chocolate | coral | cornflowerblue | cornsilk | crimson | cyan | darkblue | darkcyan | darkgoldenrod | darkgray | darkgreen | darkkhaki | darkmagenta | darkolivegreen | darkorange | darkorchid | darkred | darksalmon | darkseagreen | darkslateblue | darkslategray | darkturquoise | darkviolet | deeppink | deepskyblue | dimgray | dodgerblue | firebrick | floralwhite | forestgreen | fuchsia | gainsboro | ghostwhite | gold | goldenrod | gray | green | greenyellow | honeydew | hotpink | indianred | indigo | ivory | khaki | lavender | lavenderblush | lawngreen | lemonchiffon | lightblue | lightcoral | lightcyan | lightgoldenrodyellow | lightgreen | lightgray | lightmagenta | lightpink | lightsalmon | lightseagreen | lightskyblue | lightslategray | lightsteelblue | lightyellow | lime | limegreen | linen | magenta | maroon | mediumaquamarine | mediumblue | mediumorchid | mediumpurple | mediumseagreen | mediumslateblue | mediumspringgreen | mediumturquoise | mediumvioletred | midnightblue | mintcream | mistyrose | moccasin | navajowhite | navy | oldlace | olive | olivedrab | orange | orangered | orchid | palegoldenrod | palegreen | paleturquoise | palevioletred | papayawhip | peachpuff | peru | pink | plum | powderblue | purple | red | rosybrown | royalblue | saddlebrown | salmon | sandybrown | seagreen | seashell | sienna | silver | skyblue | slateblue | slategray | snow | springgreen | steelblue | tan | teal | thistle | tomato | turquoise | violet | wheat | white | whitesmoke | yellow | yellowgreen

aliceblue	antiquewhite	aqua	aquamarine	azure	beige	bisque
black	blanchedalmond	blue	blueviolet	brown	burlywood	cadetblue
chartreuse	chocolate	coral	cornflowerblue	cornsilk	crimson	cyan
darkblue	darkcyan	darkgoldenrod	darkgray	darkgreen	darkkhaki	darkmagenta
darkolivegreen	darkorange	darkorchid	darkred	darksalmon	darkseagreen	darkslateblue
darkslategray	darkturquoise	darkviolet	deeppink	deepskyblue	dimgray	dodgerblue
firebrick	floralwhite	forestgreen	fuchsia	gainsboro	ghostwhite	gold
goldenrod	gray	green	greenyellow	honeydew	hotpink	indianred
indigo	ivory	khaki	lavender	lavenderblush	lawngreen	lemonchiffon
lightblue	lightcoral	lightcyan	lightgoldenrodyellow	lightgreen	lightgray	lightmagenta
lightpink	lightsalmon	lightseagreen	lightskyblue	lightslategray	lightsteelblue	lightyellow
lime	limegreen	linen	magenta	maroon	mediumaquamarine	mediumblue
mediumorchid	mediumpurple	mediumseagreen	mediumslateblue	mediumspringgreen	mediumturquoise	mediumvioletred
midnightblue	mintcream	mistyrose	moccasin	navajowhite	navy	oldlace
olive	olivedrab	orange	orangered	orchid	palegoldenrod	palegreen
paleturquoise	palevioletred	papayawhip	peachpuff	peru	pink	plum
powderblue	purple	red	rosybrown	royalblue	saddlebrown	salmon
sandybrown	seagreen	seashell	sienna	silver	skyblue	slateblue
slategray	snow	springgreen	steelblue	tan	teal	thistle
tomato	turquoise	violet	wheat	white	whitesmoke	yellow
yellowgreen						

Figure 498: ADOxx-colour names

© BOC Group 2008

16. ADOxx standard application library

The **ADOxx standard application library** is automatically installed during the installation of ADOxx.

The **ADOxx standard application library** consists of the **ADOxx standard BP library** (contains the classes and relations for the modelling of business process models) and of the **ADOxx standard WE library** (contains the classes and relations for the modelling of working environment models).

> **Hint:** You will find more detailed information about the ADOxx standard application library and the ADOxx standard method (including contained library-specific functionality) in the user documentation "**Volume III - ADOxx standard method manual**".

17. Ergonomics Compliance Statement for ADOxx

The increased usage of information and communication technology has changed the working day completely. In order to avoid additional psychological stress it is important that the work and tools for the work are designed accordingly. ADOxx is a tool for business modelling and globalisation which fully complies with the requirements of software ergonomics as the following research will show.

The ergonomics analysis for ADOxx contains:

- **Human criteria** (see chap. 17.1, p. 803)
- **Dialogue design** (see chap. 17.2, p. 806)
- **Conformity with the Windows User Interface Style Guide** (see chap. 17.3, p. 813)

17.1 Human Criteria

The **seven human criteria for the work design according to DIN EN ISO 9241-2** establishes requirements, recommendations and principles for the design of interactive Multimedia User Interfaces that integrate and synchronize different media (static media such as text, graphs, images, and dynamic media such as audio, animation and video). By taking those criteria into consideration while designing the work tasks become not only harmless, restrictionless and reasonable, but also what's most important for the work design - personally beneficial.

Those criteria are:

User orienation
The work design should *take into account experiences and skills of the user group*.

Versatility
The design of the work assignments should consider, *that an appropriate variety of skills and activities are applied*.

Integrity
The work design should guarantee, *that the tasks to be achieved will be recognised as integrated work units and not fragments of those*.

Signification
The design of the work assignments should guarantee that the tasks to be achieved provide *a clear and understandable description of the system's functions*.

Scope of action
The work design should accommodate *an appropriate scope of action with regard to the order and rate of work*.

Report messages
The design of the work assignments should provide *sufficient report messages on task performance, meaningful to the user*.

Development possibilities
The design of the work assignments should *provide possibilities for further development of existing and new skills within the nature of the task*.

In the chapters below you will find examples of how to implement the above criteria in ADOxx.

17.1.1 User Orientation

The software ADOxx is offered as a package togethr with consulting and training days. Hereby BOC Group assures that persons potentially working with ADOxx will be introduced to the way of thinking of the illustrated business management of ADOxx and **ideally implement this in accordance with the tool**.

For support there is a user guide provided, that describes ADOxx in five parts:

- Introduction
- User's guide
- Method manual
- Method definition and administration manual
- Installation and database administration manual

Furthermore the above manuals are available in the context sensitive help that will display appropriate pages with respect to the user's given activity.

Through the metamodelling technique, ADOxx can be fully customised to customers' wishes. In that way the client's specific symbols, modelling segments and model types can be illustrated in ADOxx. In ADOxx the user will find already familiar methods and through them can quickly obtain an effective rate of output.

17.1.2 Versatility

The versatility arises from **the universal use the of capabilities of ADOxx for all activities of business modelling**. The individual activities are again located in the ADOxx components.

- Acquisition of data needed for modelling
- Model creation (processes and structures)
- Analysis of the models
- Evaluation of business processes
- Import-/Export of models to documents and also for exchanging with other tools

17.1.3 Integrity and Signification

The integrity is guaranteed through **Business Process Management Systems (BPMS) Paradigm** enabling a holisitc look at the Company, its processes and structures and completely depicting them inADOxx.

With the client/server architecture and multi-user operation of ADOxx, team project work is supported in order to solve the complex tasks.

The holistic view of the BPMS Paradigm, but also the possibility to immediately see the work results of the other users, allow teamwork on highest level.

17.1.4 Scope of Action

The modelling in ADOxx must be understood as a creative task. A graphical model editor supplies the user with illustrations of the models, while a tabular model editor allows a data-oriented approach.

Hint: In ADOxx there are several methods to achieve the same goal.

In order to comply with these requirements, there are several places in ADOxx, where you can find cross references to other functions. As an example **the call of the model group management** is displayed here:

- The management of the model group can be directly executed by the menu option **"Model" - "Modelgroup"**:

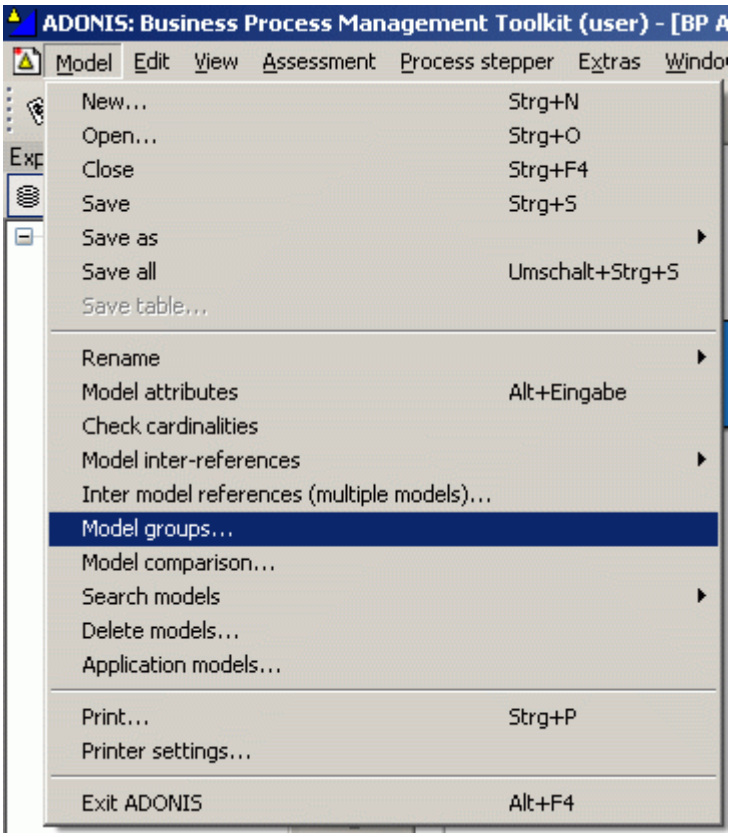


Figure 499: The direct way through the menu

- The model group management can be opened through the context menu in the dialogue where the model list is displayed. The figure shows the "Open model" dialogue with the opened context menu:

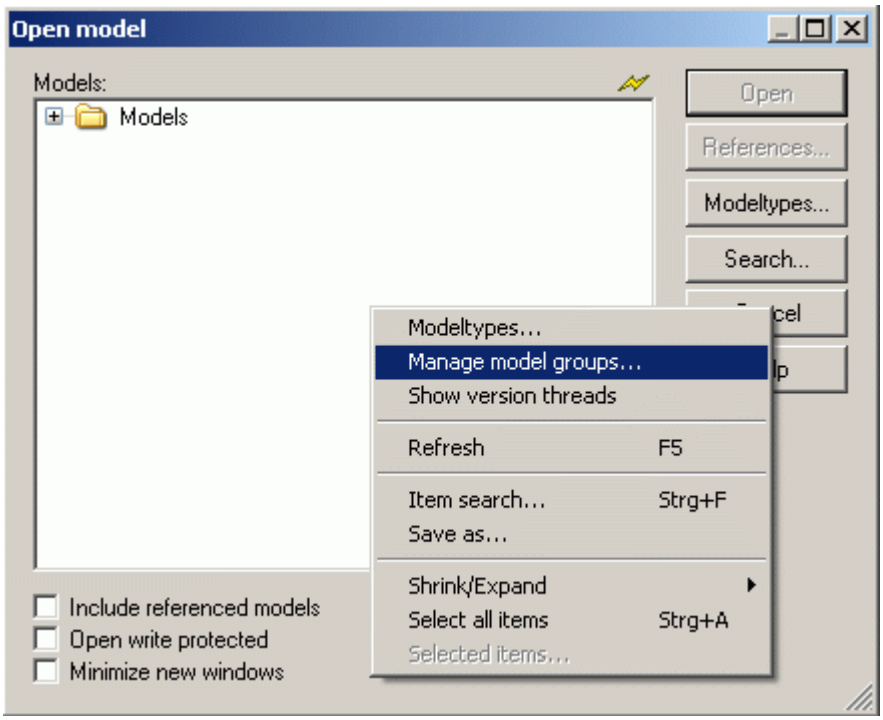


Figure 500: Indirect way through the context menu

17.1.5 Report Messages

In ADOxx the dialogues (see chap. 17.2, p. 806) are constructed in a way that **each user interaction will be answered with the corresponding report message.** Additional help for the report messages is also available in ADOxx

The dialogue commands allow the tool to guarantee or assess the quality of the performed work:

- The modelling method that is the basis for ADOxx meta model is highly defined. It allows the formal assessment of the model structure (i.e. cardinalities of the relations) and stops invalid modelling (i.e. connection from instances, where there is no relations provided).
- Quality checks can be carried out in the form of queries as well as by means of script-extension. Free to formulate queries allow ad-hoc checking of the model state.
- If the model is intended to be simulated, the ability to simulate already counts as the quality criterion that must be granted for carrying out the simulation.

ADOxx also supports manual report messages in the form of review mechanisms. Those mechanisms can be established in the client-specific, special modelling methodology (i.e. model states, notes display, errors visualisation).

17.2 Dialogue Design

The dialogue representation can be assessed in light of formulating principles. The principles are quoted in the **Norm DIN EN ISO 9241-10** (valid for the German speaking region)

Task appropriateness

A dialogue is task appropriate, when it *effectively and efficiently fulfills its tasks*.

Self-desctription ability
The dialogue is able to describe itself, when *every individual dialogue step is immediately understandable* through the report message of the dialogue system or the user's question will be answered.

Expectation conformity
A dialogue has expectations to *conform, when it is consistent and user's features correspond*.

Controllability
A dialogue is controllable, when the user is able to start the dialogue flow, and can influence its speed and position, until the destination is reached.

Error tolerance
A dialogue is error tolerant, when *the intended work result* can be, in spite of recognisable incorrectness of the input, achieved with *either none or minimal correction cost* from a user's point of view.

Individualisation
A dialogue is individualised, when *the dialogue system permits adaptations on the work task request and the individual user preferences*.

Learn ability
A dialogue has learning ability if it *supports the user in learning the dialogue system*.

17.2.1 Task Appropriateness

Efficiency of the work design is achieved in ADOxx through suitable separation of the working steps, dividing dialogues into logically related groups of control elements, and through the selected use of default buttons. Also the state of the default buttons is changed to stop premature closure of the dialogue.

Hint: The default button is the button that will be executed when pressing the <Enter> key.

The figure below shows how the element division occurs in logically related groups in the form of registry cards on the basis of the ADL Import dialogues. It is obvious that the default button "OK", will only be active after the provision of the data name, because a previous continuation of that action would not make sense:

Figure 501: The figure shows the examplorary element division and behavoiur of the default button "OK"

17.2.2 Self-description Ability

By designing the ADOxx user interface, a special emphasis was put on clear timing of the control elements. Buttons are called with verbs where appropriate. Other texts are named in a detailed manner and are consistent with naming conventions.

The following figure illustrates the "open model" dialogue that provides some functions that are clearly explained through unambiguous wording. In every dialogue the context sensitive help is at the user's disposal and immediately provides information to the dialogue.

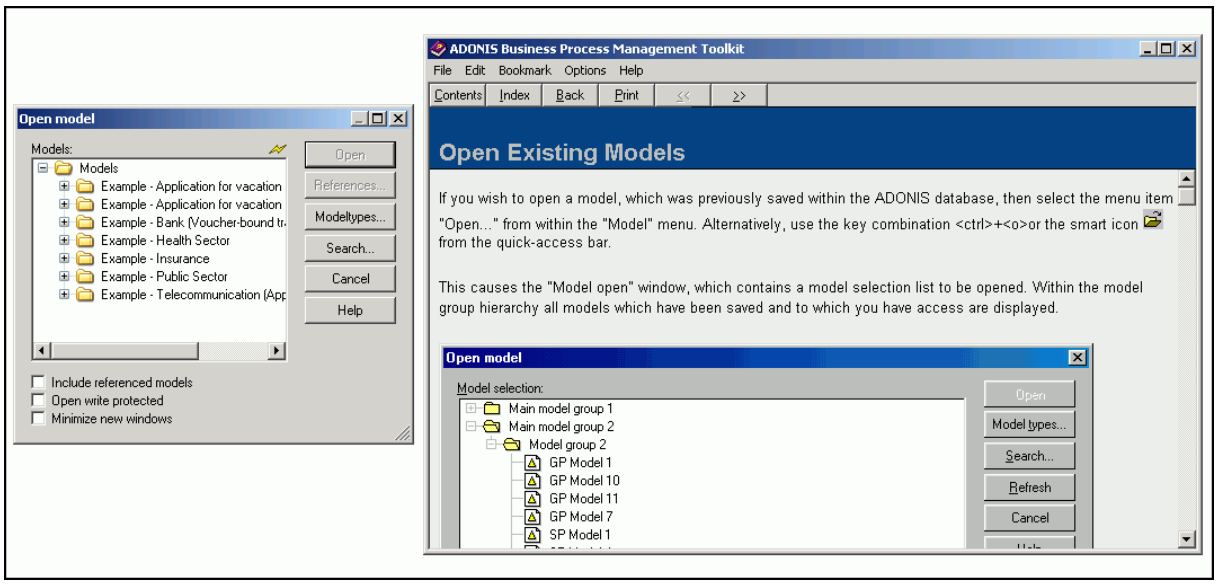


Figure 502: The elements of the "Open model" dialogue

Tooltips will provide comments for ambiguous names. The following figure illustrates a tip of the "Update" button in the "Open model" dialogue.

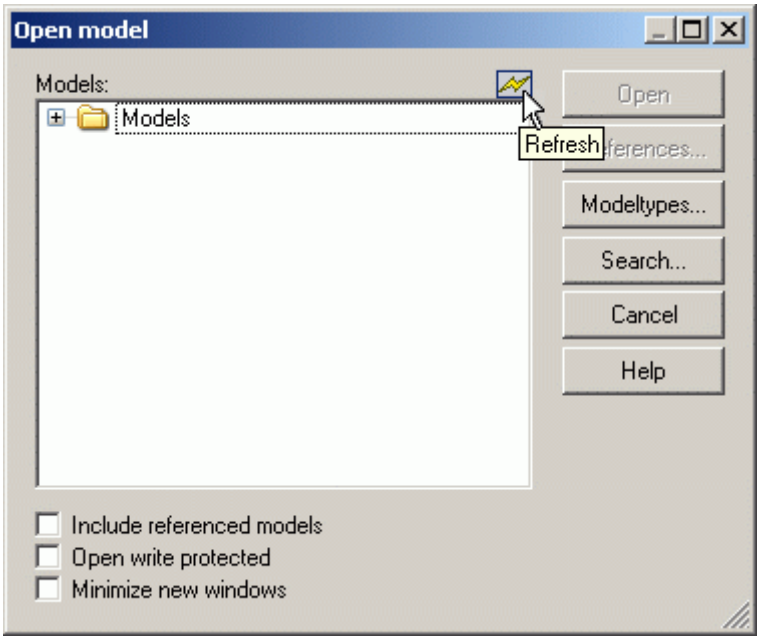


Figure 503: Tooltips

Hint: A Tooltip is a small (classically light yellow) window that contains a short description of the button. It appears when the mouse pointer hovers over the button for some time.

A progress of fairly long actions (like opening a model, import, export, simulation etc.) will be visualised by the status bar. The status bar will display the result at the end of the action.

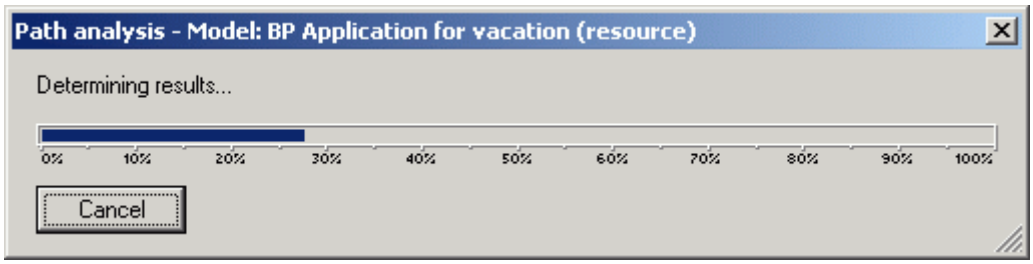


Figure 504: Progress bar for longer operations

If the operation lasts more than ten seconds, the status bar will display time information.

17.2.3 Expectation Conformity

ADOxx conforms to the Windows User Interface Style Guide (see chap. 17.3, p. 813). It also uses the icons and menus of the Microsoft Office products. ADOxx-specific functionality is illustrated through specially designed smart icons. The nomenclature in ADOxx is arranged uniformly; the dialogues consistently use conceptualities. In the same way the structure of the dialogue has been designed homogeneously: dialogues of the same kind represent elements of the same type. The following figure illustrates the comparison of the dialogue management "delete a model" and "model group management". On the left side, the model (group) list will be displayed, on the right side are the buttons with the possible operations. The "help" button is located at the last position. On the right side, above from the model (group) list, there is an icon to update the list.

Hint: Icons are graphical symbols for application functions. Usually they are represented by small buttons on the toolbar.

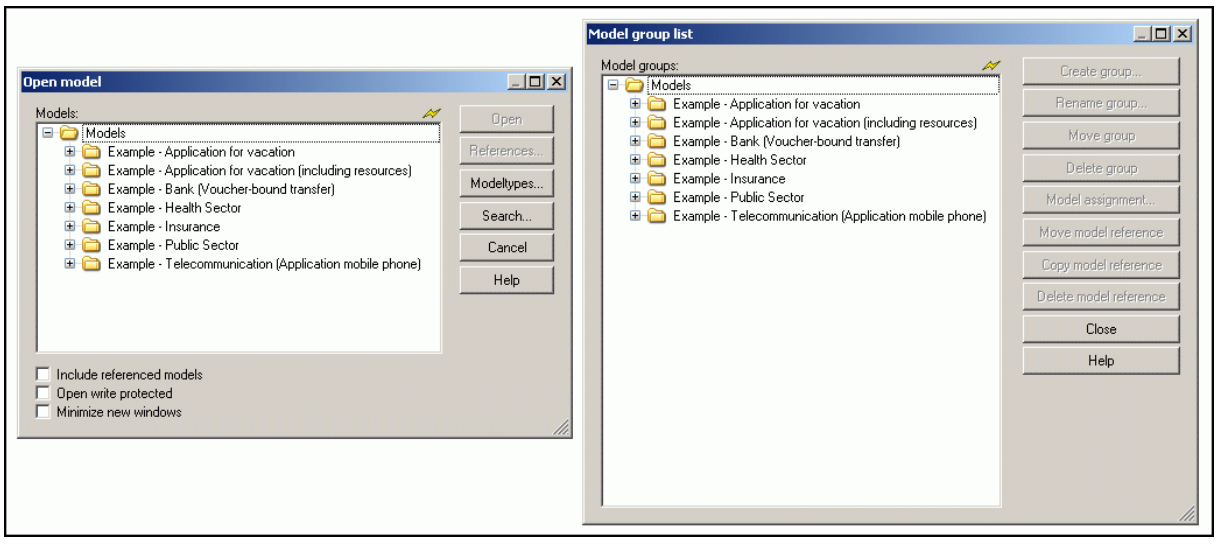


Figure 505: Comparison of the "Open a model" and "List of model groups" dialogues"

17.2.4 Controllability

ADOxx has been developed as a so-called MDI application. This allows simultaneous handling of several models. Arrow keys serve to switch between model windows, a selection window allows direct navigation.

Figure 506: Switching between open model windows

Hint: MDI = Multiple Document Interface, denotes that in the one user interface more than one document (in ADOxx models are documents) can be opened at the same time.

Some dialogues and the ADOxx Notebook are structured by means of register cards. Register cards serve to clearly group the elements from a logistical point of view. To switch between register cards press <Crtl>+<Tab> or alternatively <Shift>+<Crtl>+<Tab>. The ADL import dialogue is an example of a register card (see fig. 501); The ADOxx Notebook is shown in the figure below.

The ADOxx Notebook is implemented as a modeless window. This allows on one hand opening more than one notebook and on the other hand to work on the model while one or more notebooks are open.

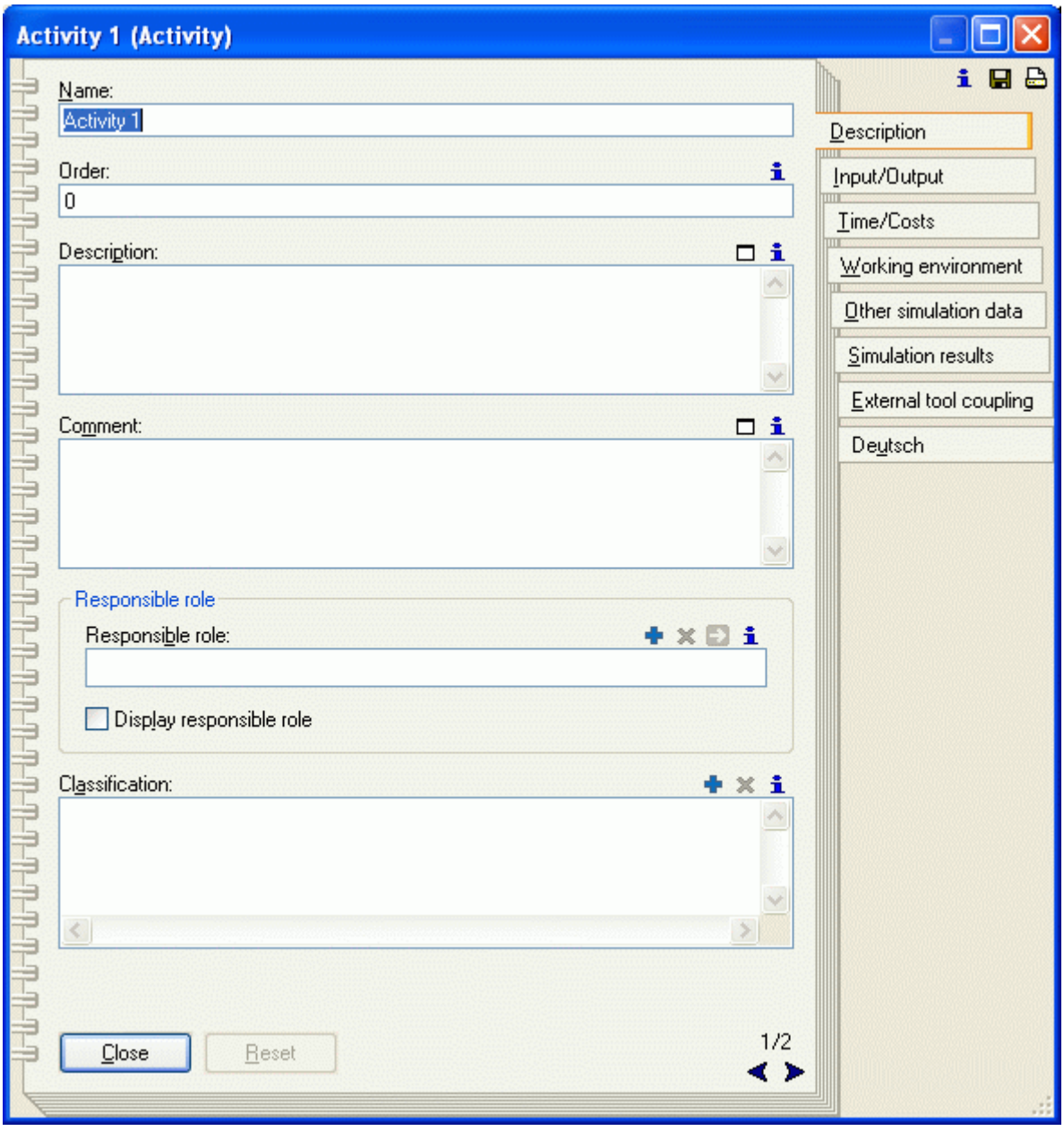


Figure 507: A sample ADOxx Notebook

Hint: Modeless dialogues do not exclusively receive the focus in the application. Thus it is possible to call and click on other windows (for example the main window or other dialogues) while the modeless dialogue remains open.

ADOxx is structured in components. As a result, the "usefulness" (see chap. 17.1.2, p. 804) of menus and toolbars is achieved.

The order of components corresponds to the workflow of the business modelling. Context menus offer advanced functionality. To increace clarity, such functions will remain hidden outside the context menus. Demanding or professional users can use those functions to gain extended controlability (see chap. 17.1.4, p. 804).

17.2.5 Error Tolerance

While carrying out work tasks, the user is supported by the error messages which are identifed by unified error codes (written in the square brackets). By clicking on the "Help" button, a help page with a precise explanation and solution suggestions is provided:

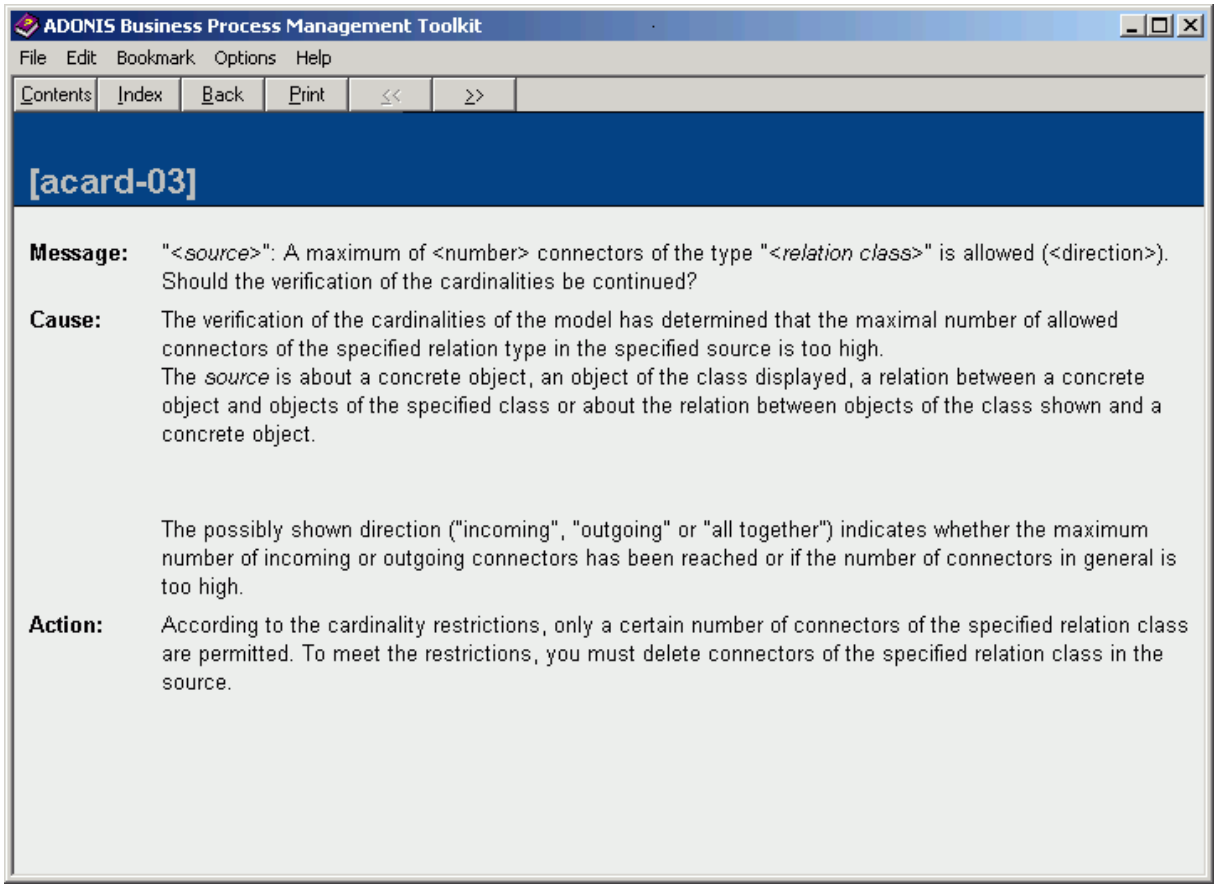


Figure 508: A typical error message

Numbers and date expressions will be automatically formed or corrected:

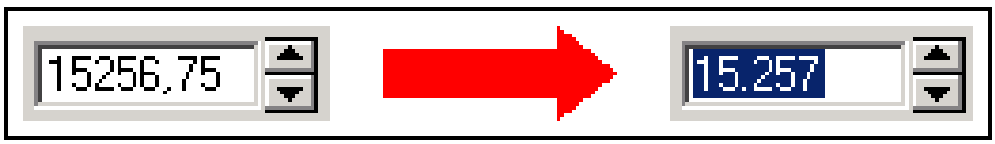


Figure 509: Automatical rounding up to the whole number

False input leads to later actions being not available (i.e. the confirmation button remains deactivated).

17.2.6 Individuality

The meta-model concept of ADOxx allows client-specific adaptations and the development of new modelling methods. For that reason it is possible to adapt graphical visualisations of the modelled objects, layout, size of the generated documents etc.

For individual users, components can be shown or hidden as needed. The toolbar (quick-access bar) can be modified using the AdoScript language.

17.2.7 Learn Beneficiality

ADOxx uses the icons and menu entries of the Micorosoft Office product series. The ADOxx-specific functionality is illustrated through the specially designed smart-icons being consequently re-used. The dialogue icons in dialogues and in the ADOxx Notebook, are a good example of this.

The figure shows the dialogue icons for adding, deleting and following of objects and model references:

Figure 510: Some smart-icons

The learning ability benefits from the clear structure of the components in ADOxx. Furthermore, tooltips and shortcuts which appear directly on the surface area are easy to learn. The used key-combinations correspond to the Windows User Interface Style Guide and also to the stardard key combination of e.g. Microsoft Office:

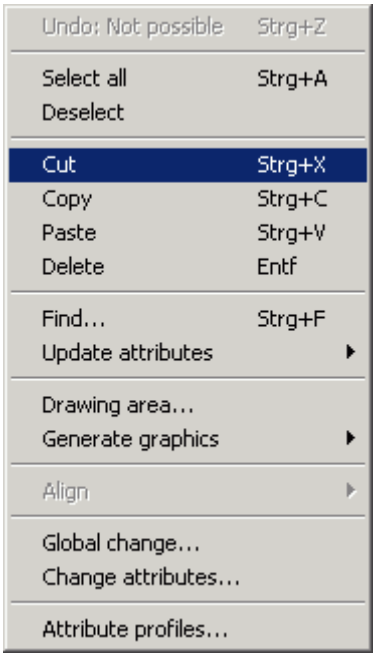


Figure 511: Standard menu and standard key combination

Hint: A shortcut is a key combination that can be used to directly call program functions. Shortcuts are used i.e. for frequently used commands in menus.

ADOxx offers a possibility to preview the printout or the generated graphics, so that the user can see the result before performing the operation:

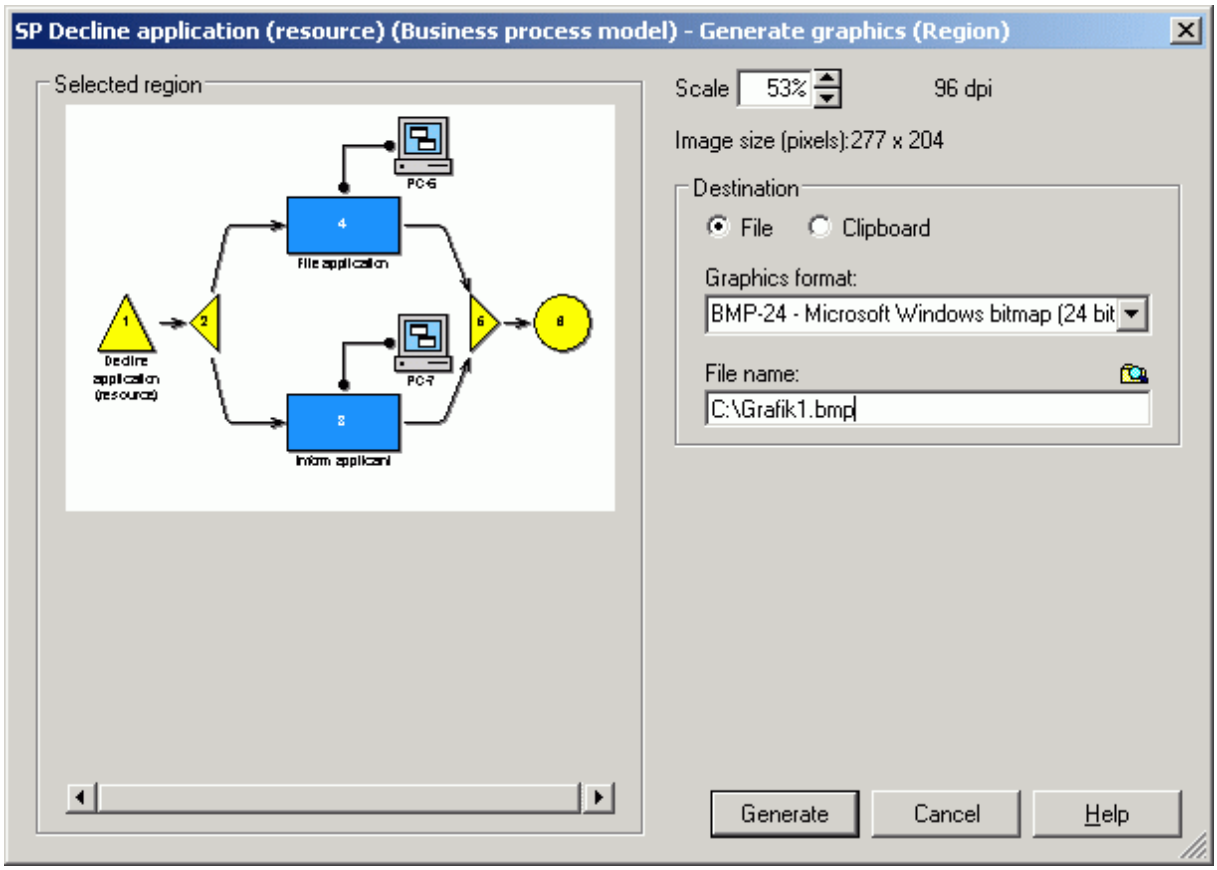


Figure 512: Graphics generation

The input in the Notebook is immediately applied to the visualised models in the model editor. As a result, a very quick feedback on performed changes is provided. Using this feature, the user can easily see the consequences of the input.

17.3 Conformity With the Windows User Interface Style Guide

ADOxx has been and currently is developed as a Microsoft Windows application and the user interface is designed according to the standards of the Microsoft Windows User Interface Style Guide.

Microsoft with its comprehensive **Windows User Interface Style Guide** (see chap. 17.3.1, p. 814) defines how a Windows application should be designed. This is thought to ensure a common "Look & Feel" of Windows applications and also ensures that the thoughtfully developed ergonomics of the style guide are also transferred to applications not developed by Microsoft. The ADOxx user interface takes these requirements into account.

Some of these points listed below illustrate this:

- ADOxx has been designed as a MDI application. This allows for the ability to work with several models at the one time. Each model is shown in a separate window. Using the menu "Window" they all can be accessed and administered.

- Colours, fonts and font sizes as well as international formats are based on the global Windows system settings.
- the font size can be adjusted to the requirements of the respective user. The user can change the font size of the visualised text of a model (on the drawing area) (menu "View", menu item "Font size") and can select the font size used in dialogues and menus when starting ADOxx (command line parameter "-h").
- All menus and dialogues can be accessed using either the mouse or the keyboard. The user is further supported by shortcuts.
- Wherever possible, Windows standard dialogues are used. This is of an advantage as the user will be familiar with these dialogues from other applications and they will always use the language of the operating system:
  - query and status messages
  - file handling (e.g. "Search" button to enter file names during the import, export, documentation generation, protocol files etc.)
  - printer settings

  By using standard dialogues the system settings for sound are also applied (acoustic signals).

17.3.1 Literature

- Microsoft: Official Guidelines for User Interface Developers and Designers. Microsoft Press, 1999.
- Microsoft: The Windows Interface Guidelines for Software Design. Microsoft Press, 1995.

18. Glossary

The glossary explains the expressions used in ADOxx.

The symbol "→" refers to further entries in the glossary.

- A -

ABL. ADONIS **B**inary **L**anguage; file format, in which →application libraries can be saved (exported).

Accelerator. Letter marked by underlining. Occurs in titles of menus and menu options, in field and list names of input fields and in chapter headings and attribute names of →ADOxx Notebooks. By entering the accelerator (sometimes in connection with the <Alt> key) you can call functions and navigate inside both windows and ADOxx Notebooks via the keyboard.

Access Rights. The access rights (read-write/read-only/no access) to an →ADOxx model group are assigned to the →ADOxx user groups by the →ADOxx administrator.

Acquisition (component). →Component of the →Modelling Toolkit. Supports the core activity "Acquisition" in organisation projects with the help of →Acquisition Tables.

Acquisition Tables (HOMER). Functionality in the →Acquisition Component in the →Modelling Toolkit. The acquisition tables help you to store and manage the data acquired and transfer into ADOxx.

ACR. ADONIS **C**omparable **R**epresentation; file format, in which simulation and analysis results, which are displayed in the →ADOxx browser can be saved. ACR-files are needed in the →Evaluation Component for the →comparison of results.

Activity. Smallest entity of a business process. Activities describe what is done in a →business process.

Activity, cooperative asynchronous. An →activity carried out by several →performers who work together but not at the same time.

Activity, cooperative synchronous. An →activity carried out by several →performers who work together at the same time.

ADL. ADONIS **D**efinition **L**anguage; in which →ADOxx models and →ADOxx model groups as well as →ADOxx attribute profile and →ADOxx attribute profile groups can be saved (exported).

ADL Export. Functionality in the →Import/Export Component in the →Modelling Toolkit and in the →Model Management in the →Administration Toolkit. Using the ADL export profiles, you can export →ADOxx models and →ADOxx model groups as well as →ADOxx attribute profile and →ADOxx attribute profile groups to →ADL files.

ADL Import. Functionality of the →Import/Export Component in the →Modelling Toolkit and in the →Model Management in the →Administration Toolkit. Using the ADL import functionality, you can import →ADL files (→ADOxx models and →ADOxx model groups) as well as →ADOxx attribute profile and →ADOxx attribute profile groups into ADOxx and store them in the →ADOxx database.

Administration Toolkit. ADOxx tool, needed by the →ADOxx administrator in order to manage →ADOxx users, →ADOxx libraries, →ADOxx models and →ADOxx attribute profiles. The Administration Toolkit consists of the →User Management, →Library Management, →Model Management, →Attribute Profile Management and →Component Management →components.

ADO*it*. IT Service and IT Architecture Management Tool (manufacturer: BOC Asset Management GmbH). Part of the →Management Office.

ADO*log*. Supply Chain Management Tool (manufacturer: BOC Asset Management GmbH). Part of the →Management Office.

ADONIS. Business Process and Knowledge Management Tool (manufacturer: BOC Asset Management GmbH). Part of the →Management Office.

ADOxx Administrator. Person, who is responsible for the configuration of ADOxx ("→Customising"), the →User Management, →Library Management, →Model Management, →Attribute Profile Management and →Component Management.

ADOxx Attribute Profile. →Attribute profile

ADOxx Attribute Profile Group. →attribute profile group

ADOxx-Default-Library. →Application library, automatically generated during the installation of a →ADOxx database for ADOxx version 3.9 and up and assigned to the standard user "Admin". The ADOxx-Default-Library consists of the →ADOxx-Default-BP-Library and the →ADOxx-Default-WE-Library. The ADOxx-Default-Library is the successor and an extension to the →ADOxx-Standard-Anwendungsbibliothek.

ADOxx-Default-WE-Library Working Environment library, automatically generated during the installation of a →ADOxx database for ADOxx version 3.9 and up. The ADOxx-Default-WE-Library is part of the →ADOxx-Default-Library. With the →classes and →relations defined in the ADOxx-Default-WE-Library, models of the →model type →"Working Environment model" can be created.

ADOxx-Default-BP-Library. , automatically generated during the installation of a →ADOxx database for ADOxx version 3.9 and up. The ADOxx-Default-BP-Library is part of the →ADOxx-Default-Library. With the →classes and →relations defined in the ADOxx-Default-BP-Library, models of the →model types →"Company Map", →"Business Process Model", →"Document Model", →Product Model, →IT System Model and →"Use Case Diagram" can be created.

ADOxx Browser. Special result window used in the →Modelling Toolkit(in the components →Modelling, →Analysis, →Simulation and →Evaluation) as well as in the →Administration Toolkit (in the components →Library Management and →Attribute Profile Management). The results are represented in tabular form and can be displayed graphically. They can also be printed out, saved in a file in →TXT format (table and text), →RTF format or →HTML format.

ADOxx Business Edition. Compared to the complete ADOxx Professional Edition, this is a restricted configuration (→ADOxx configuration) of the →Modelling Toolkit. The ADOxx Business Edition consists of the components →Modelling, →Analysis, →Documentation and →Import/Export.

ADOxx Business Edition. Compared to the complete ADOxx Professional Edition, this is a restricted configuration (→ADOxx configuration) of the →Modelling Toolkit. The ADOxx Business Edition consists of the components →Modelling, →Analysis, →Documentation and →Import/Export.

ADOxx Component. Part of ADOxx software. The standard ADOxx components are the →ModellingToolkit and the →Administration Toolkit as well as the →database administration. In addition, the →Process Cost Analysis,^→Dynamic Evaluation Module, →case/4/0 coupling and →objectiF coupling can be purchased as additional components.

ADOxx Configuration. A version of the →Modelling Toolkit in which the →components are configured according to the customer's needs. The minimal ADOxx configuration consists of the →modelling component.

ADOxx Database. All ADOxx data is stored in the ADOxx database.

ADOxx Database Administration. The ADOxx Database administration contains the programs necessary for generating, saving and restoring →ADOxx databases.

ADOxx Default Language. The language ADOxx starts by default with. The default language is set during installation.

ADOxx Directory. Directory in which ADOxx has been installed. Default: `C:\Program files\BOC\ADOxx10`.

ADOxx Explorer. Tool providing an overview of the →models/model groups stored in the ADOxx database and offering various manipulation possibilities.

ADOxx Inspector. Tool that supplies the →ADOxx user with various details about the active modelling environment.

ADOxx Installation Directory. →ADOxx directory

ADOxx Mail. Messaging function in ADOxx, using which →ADOxx users can exchange (→express) messages. The messages will be saved to the→ADOxx database .

ADOxx Model. Aggregation of →objects of various →classes that serve to represent the essential aspects of the real world for the →ADOxx user. The models serve as a basis for →components such as →analysis, →simulation and →evaluation.

ADOxx Model Group. Group, in which →ADOxx models are referenced. →Access rights for →ADOxx user groups and the →ADOxx users referenced therein are defined for an ADOxx model group.

ADOxx Navigator. Tool which shows an overview of the active →model. The part of the model, which is currently visible in the →model window is highlighted and can be used for navigating through the model.

ADOxx Notebook. Special dialogue window where the attributes of →objects, →connectors, →attribute profiles, →models or →libraries may be changed.

ADOxx Professional Edition. Complete ADOxx version of the →Modelling Toolkit (→ADOxx configuration). The ADOxx Professional Edition in its standard configuration consists of the components →Acquisition, →Modelling, →Analysis, →Simulation, →Evaluation, →Documentation and →Import/Export.

ADOxx Program Folder. Contains the program icons for the installed ADOxx toolkits, the →online manuals and (if applicable) the →ADOxx database administration.

ADOxx Rights. Rights of an →ADOxx user to access the →Administration Toolkit and/or →Modelling Toolkit.

ADOxx Standard Application Library. →Application library that is automatically generated when an →ADOxx database is generated for ADOxx 3.81 or earlier. Since ADOxx 3.9, the →ADOxx-Default-Library is generated automatically. The ADOxx Standard Application Library is assigned to the standard user "Admin". It consists of the →ADOxx Standard BP library and the →ADOxx Standard WE library.

ADOxx Standard BP Library. →Business process library that is automatically generated when an →ADOxx database (up to version 3.81) is generated. The ADOxx Standard BP Library is part of the →ADOxx Standard Application Library. You can create models of types →"Business Process Model"→"Document Model", →"Use Case Diagram" and →"Company Map" with the classes and relations defined in the ADOxx Standard BP Library.

ADOxx Standard WE Library. Working Environment library that is automatically generated when an →ADOxx database (up to version 3.81) is created. The ADOxx Standard WE Library is part of the →ADOxx Standard Application Library.. You can create models of the model type →"Working Environment Model" with the classes and relations defined in the ADOxx Standard WE Library.

ADOxx User. User of →ADOxx components. The ADOxx users stored in the →ADOxx database are displayed in the →user list. ADOxx users may be exported (optionally including →ADOxx user groups) to →UDL files or imported from →UDL files.

ADOxx User Group. Group, in which →ADOxx users are referenced. An ADOxx user group has →access rights to →ADOxx model groups and the →ADOxx models referenced within these. The ADOxx user groups stored in the →ADOxx database are displayed in the →user group list. ADOxx user groups may (optionally including →ADOxx users) be exported into →UDL files or imported from UDL files.

ADOxx User Management (component). A →component of the →Administration Toolkit where →ADOxx users can be created, deleted and their settings can be changed →ADOxx user groups can be created, renamed and deleted. An ADOxx user can be assigned to ADOxx user groups and ADOxx users can be imported and exported.

ADOxx Window. Graphical user interface of ADOxx. You can call the individual →components from within the ADOxx window.

ADO*score*. Strategy and Performance Management Tool (manufacturer: BOC Asset Management GmbH). Part of the →Management Office.

AdoScript. Script language, using which you can implement customised functionality extensions in ADOxx. Moreover AdoScript enables the call of any external programs and functions in DLLs.

Agent. Created by a →ADOxx user in the →Simulation Component. Calculates user-defined simulation results during the simulation. An agent's results refer only to the objects specified by the user.

Agent Calendar. Description of the period of time during which an agent evaluates the simulation algorithm →Workload Analysis.

Alignment Function. Functionality in the →Modelling Component, which allows you to align the →objects and →connectors of an →ADOxx model with the help of pre-defined or user-defined parameters. Available in the "Edit menu".

Analysis (component). →Component of the →Modelling Toolkit, in which →queries and reports can be run on your →models. (i.e. you can evaluate your →business process models. The queries are formulated in a language called →AQL (However, you do not need to understand AQL syntax in order to run analysis queries).

Analytical Evaluation (Calculation). Evaluation mechanism for →business process models in the →Analysis Component of the →Business Process Managament Toolkit. The →models are not simulated by this kind of evaluation but calculated mathematically.

Animation. Passive component in the simulation algorithms →"Workload Analysis (fixed time period)" and →"Workload Analysis (steady state)", which is responsible for graphically representing the →task stacks of the different →performers in the →Working Environment model being simulated. (Do not mix up with →offline animation).

APF. **A**DONIS **P**rotocol **F**ile; file format in which the results of a simulation run can be saved. APF files serve as input for the →offline animation function.

Application Library. Combination of a →Business Preocess and a →Working Environment library. The two →libraries contain the →classes and the →relations for the →models of the →model types defined. The application libraries stored in the →ADOxx database are displayed in the →application library list. Exactly one application library is assigned to each →ADOxx user.

Application Library List. List displaying all application libraries that are stored in the →ADOxx database. From here you can import, export and delete application libraries.

Application Model. Combination of at least one →business process model and exactly one →Working Environment model. The definition of an application model is necessary for the →Simulation Component to carry out the simulation algorithms "→Capacity Analysis" and "→Workload Analysis".

AQL. **A**DONIS **Q**uery **L**anguage; query language which enables you to →assign performers and →allocate resources in the →Modelling Component. You can also enter selection criteria for the functions "global change" and "model search". In addition, you can run →queries in the →Analysis Component and generate →pre-defined queries and →pre-defined evaluation queries when you edit the →library attributes in the →Library Management.

Assessment. Functionality in the →Modelling Component in the →Business Process Management Toolkit. This functionality enables you to evaluate →business process models according to time and cost criteria.

Attribute. Property of a →library, a →model, a →class, an →object, a →relation or a →connector. Attributes are usually filled with values (=attribute values) in ADOxx Notebooks.

Attribute-dependent Graphical Representation. Representation of →objects and →visualised attribute values on the →drawing area depending on concrete attribute values. An attribute-dependent graphical representation can be defined while →customising in the →Administration Toolkit within the →Library Management component. A Resource is an example of an attribute from the ADOxx-Default-Library that has an attribute-dependent graphical representation.

Attribute Filter. Filter function for the →Documentation Component. Using the attribute filter, you can select the attributes you want to transfer to the documentation (e.g. HTML, XML, RTF files).

Attribute Mode. Presetting in the application library for the →Documentation Component. An attribute mode defines, which attributes of a class should be transferred to the documentation (e.g. HTML, XML, RTF files).

Attribute Profile. Concrete instance of an attribute profile class. An attribute profile represents one or more →attributes, which are reused for any →objects and can be maintained in the →Attribute Profile Management).

Attribute Profile Class. Template, according to which the →attribute profiles are generated by →instancing. An attribute profile class or its →attribute profiles are described by a quantity of →attributes, which are assigned with concrete values in an →attribute profile.

Attribute Profile Group. Grouping, in which →attribute profiles are summed up.

Attribute Profile Management. →Component in the →Administration Toolkit, in which you can instance, edit or delete →attribute profiles as well as create, rename and delete →attribute profile groups.

Attribute Profile Reference. ATTRPROFREF; attribute type for referencing an attribute profile.

Attribute Value. Value of an →attribute, being a concrete instance of a characteristic.

Automatic Random Generator. →Random generator, automatic.

- B -

Balanced Scorecard (BSC). Instrument to translate company's mission and strategy to a clear system in order to measure the performance. This system creates the framework for a strategic performance measurement and management system. The Balanced Scorecard focuses above all on the financial targets, but contains also the business drivers of these financial targets.

Bendpoint. Point where a →connector changes its direction.

BMP. Microsoft Bitmap; graphic file format, in which diagrams/graphics which have been generated from →models can be saved.

BPMS Paradigm. (=Business Process Management Systems-Paradigm). Framework for the management of →business processes. This framework consists of five subprocesses: "Strategic Decision Process", "Re-engineering Process", "Resource Allocation Process", "Workflow Process" and "Performance Evaluation Process". It was developed by the BPMS group of the Institute for Applied Computer Science and Information Systems, Department of Knowledge Engineering, University of Vienna.

Browser. →ADOxx browser

Business Process. Number of logically connected →activities whose execution should meet a business objective.

Business Process Library. →Classes for →modelling business processes. The →classes of a business process library are derived from the classes of a →business process meta model. A business process library is part of an →application library. A business process model always refers to

exactly one business process library. Business process libraries are defined while ADOxx is →customised

Modelling Toolkit. →ADOxx component in which the →ADOxx user works. In its standard configuration it consists of the following →components: →Acquisition, →Modelling, →Analysis, →Simulation, →Evaluation and →Import/Export.

Business Process Meta Model. →Classes pre-defined in ADOxx for →modelling →business processes. The →classes of a →business process library are derived from the →classes contained in the business process meta model.

Business Process Model. 1. →ADOxx model of a business process. Comes into existence by →instantiation of →classes of a →business process library.

2. Model type of the →ADOxx-Default-BP-Library.

- C -

Calculation. →Analytical Evaluation.

Calendar. 1. →Agent Calendar, →Performer Calendar, →Process Calendar.

2. CALENDAR; attribute type for defining the period of presence of performers (→Performer Calendar) or the period of occurrence (frequency) of business processes (→Process Calendar).

Capacity Analysis. Simulation algorithm in the →Simulation Component of the →Business Process Management Toolkit which "plays through" →business processes and assigns →activities to the →performers. An →application model serves as input for capacity analysis. This analysis enables you to plan your personnel capacity demand. The results of the capacity analysis are displayed in the →ADOxx browser.

Capacity Management (component). Special definition of the →Dynamic Evaluation Modules. Used for period-related calculation of the organisational/personnel capacity requirements.

case/4/0. Data and function-oriented CASE tool (manufacturer: microTOOL GmbH)

CASE. Computer **A**ided **S**oftware **E**ngineering.

Class. Static template according to which →objects are created by →instantiation. A class or its →objects, are described by a set of →attributes, which are filled with concrete values within an →object. The class "Performer", for example, has among others the →attributes "Name", "Availability" and "Hourly wages".

Class Attribute. →Attribute of a →class, which refers directly to the class and does therefore not occur in →objects of this class such as the attribute "Info text" in an ADOxx modelling class (See: Information icon in Notebooks).

Class Filter. Filter function for the →Documentation Component. Using the class filter, you can select the →classes, in which →objects should be transferred to the documentation (e.g. HTML, XML, RTF files).

Class Panel. Button panel, which contains exactly one button for each →class that can be modelled. The class panel represents the second part of the →modelling bar - between the →edit button and the →relation panel.

Client. A client is a software module which requests services from a →server, usually via a network.

Client/Server. Computer system consisting of a →client and a →server.

Company Map. 1. ADOxx model containing an overview of all →business processes occurring in a company. Comes into existence by →instantiating →classes of a →business process library.

2. Model type of the →ADOxx-Default-BP-Library.

Comparison of Results. Functionality of the →Evaluation Component in the →Modelling Toolkit. Compares results of analysis and simulation in tabular and graphical form. The results are displayed in the →ADOxx browser.

Component. Individual module of the →Administration Toolkit or of the →Modelling Toolkit (e.g. Modelling, Analysis, Model Management).

Component Access. Configuration of the access to the →components of the →Modelling Toolkit. The component access can be defined individually by the →ADOxx administrator for →ADOxx user groups and for each →ADOxx user, whereby the →ADOxx configuration can be customised

Component Bar. Panel, by default below the →menu bar in the →ADOxx window. The smart-icons visible show the available →components. To the right of the component bar, usually the →quick-access bar can be found.

Component Configuration. Functionality of the Component Management in the →Administration Toolkit. Additional ADOxx components can be included here by entering a new licence number.

Component Management. Component of the →Administration Toolkit, in which the components of the →Modelling Toolkit can be configured.

Configuration, current. The current configuration (→ADOxx configuration) specifies which functions of the components of the →Modelling Toolkit are available.

Connector. →Instantiation of a →relation. Connection/link between two →objects (e.g. "Subsequent").

Connector Attribute. →Attribute of a →connector.

Connector Mark. Visualisation of connection joints for connectors that extend over page boundaries. (See →connectors, →page layout). Connector marks are very useful in model documentation.

Connector Numbering. Assigning numerical or alphabetical identifiers to →connector marks. The connector numbering is non-ambiguous within a →model.

Context Menu. Context-depending menu. Appears when you press the right mouse-button. Provides functionality suitable for the particular situation.

Cooperative Activity. An →activity, which is executed by more than one →performer (such as a meeting).

Cost Cutting Component. Additional →ADOxx component, which can be integrated into the →Evaluation Component of the →Modelling Toolkit. The Cost Cutting Component is not part of ADOxx standard configurations. The Cost Cutting Component supports cost optimisation, especially in the area of overhead costs.

Cost Driver. Establishes the relation between costs, processes and calculation objects. The cost driver is a measure of the costs caused by the business processes and subprocesses.

Cost Driver Quantity. The measurable output of a →cost driver. Each →output-induced (OI) process has a cost driver quantity.

CSV. Comma **S**eparated **V**alue; text file format in which contents from the →ADOxx browser can be stored and transferred without loss to a spreadsheet software.

Cursor Key. Arrow keys on the keyboard, using which you can navigate to the left, the right, upward and downward.

Customer Number. An unambiguous alphanumerical identifier for each customer. The customer number must be entered when installing ADOxx.

Customising. Adapting ADOxx to a customer's demands without programming effort. Via customising, definition of the →page layout, the →model types, →pre-defined analysis queries and →plans as well as →pre-defined evaluation queries and simulation mechanisms is performed.

Cycle Time. Time that (on average) passes between the start of a →business process and its end.

- D -

Database Administration. ADOxx component, which the →database administrator uses to administrate the →ADOxx databases. The database administration contains the programs to create, store (only for →DB2 databases) and restore (only for →DB2 databases) →ADOxx databases.

Database Administrator. Person responsible for maintaining, configuring and administering the database system.

Day Profile. Template, which describes a →performer's time of presence or the stochastic occurrence (frequency) of a →business process per day. Day profiles are used in the →Agent Calendar, the →Performer Calendar and the →Process Calendar.

DB2. →DBMS for Windows, Linux, OS/2 and other operating systems (owner: IBM).

DBMS. **D**ata**B**ase **M**anagement **S**ystem.

Default Language. →ADOxx default language.

Depth Search. Process to go through objects, i.e. to automatically process the objects and relations of a model in the RTF generation in the →Documentation Component. Starting from a start object, all objects, which are directly connected to this start object, will be processed. Unlike the →width search the process will not be carried out gradually, but the relations of the objects will be followed until an object, which has no further relation, is reached.

Distribution. DISTRIBUTION; Attribute type for statistic distribution used for assigning variables.

Document Model. Model type of the →ADOxx-Default-BP-Library to illustrate the documents used in →Business Process Models.

Documentation (component). Function in the →Import/Export Component of the →Modelling Toolkit. Using the Documentation Component, you can transport →ADOxx models to specific target formats (e.g. →RTF, →HTML, →XML), whereby you can distribute the model contents including graphic representation integrated into documents or via the →Intranet.

Domain. A domain is a group of computers as part of a network using a common directory database. A domain is organised in different levels and is administered as one unit with common rules and procedures. Every domain has a unique name. (Definition from "Windows Help".)

Drawing Area. Rectangle in the →model editor of the →Modelling Toolkit. The drawing area is at least big enough so that it encloses all →objects and →connectors of a →model. The drawing area is represented in the →modelling window by a grey outline. It is part of the workspace.

Drawing Mode. Mode of the →model editor in which →objects and →connectors can be drawn. You can switch from →editing mode to drawing mode by clicking on a button in the →modelling bar.

Drawing Space. Area of a →model window, where objects and connectors can be placed. The →drawing area is a part of the drawing space.

DSSSL. **D**ocument **S**tyle **S**emantics and **S**pecification **L**anguage (DSL file); DSL files are used in the →Documentation Component, to convert →SGML files to any format (e.g. →HTML, →RTF, →XML).

DTD. **D**ocument **T**ype **D**efinition; DTD are used to define the structure of documents (e.g. in the formats →XML, →SGML) .

Dynamic Evaluation Modules. Additional →ADOxx Component, which can be integrated to the →Evaluation Component of the →Modelling Toolkit. The component "Dynamic Evaluation Modules" is not included in ADOxx standard configurations. The Dynamic Evaluation Modules enable period-related evaluations (e.g. Human Resources Management).

- E -

E-Business. E-Business includes all business activities, i.e. the internal and external business processes of an organisation, which are supported by Internet technology.

Edge Centre. For each →connector the →ADOxx user can move the "centre". By moving the edge centre, the transition condition of a connector of type "Subsequent" may be arranged along this connector as you like. Alternatively, the edge centre may automatically be placed in the geometrical centre of the connector. This is useful to position the text displayed along a connector at a position at which it can be clearly read.

Edit Button. Topmost button within the →modelling bar. With this button the cursor in the →model editor can switch from →drawing mode ("pen") to →editing mode ("arrow").

Editing Mode. Mode in the →model editor, in which it is possible to edit (i.e. select, move, copy, cut, paste, delete etc) →objects and →connectors. You can switch from →drawing mode to editing mode either by clicking on the →editing button in the modelling bar or by right-clicking on the →drawing area in the →modelling window.

EMF. Windows **E**nhanced **M**eta**F**iles; Graphic file format, in which graphics, which have been generated from →models can be stored. EMF is based on vector graphics. This way EMF graphics can be scaled (resized) freely.

Empty Space Tool. Tool in the →Model editor that helps creating empty spaces in crowded models by moving the →objects. Part of the →Modelling bar.

Enumeration. ENUMERATION; attribute type with predefined attribute values. The valid value is exactly one of the set of values supplied.

Enumeration List. ENUMERATIONLIST; attribute type with predefined attribute values. The value valid may be zero, one or more elements of the value range defined.

Evaluation Agent. →Agent

Evaluation (component). →Component of the →Modelling Toolkit. Supports both the core "evaluation" activity in re-organisation projects and the "Performance Evaluation Process" within the →BPMS Paradigm.

Evaluation Queries. Functionality of the →Evaluation Component in the →Modelling Toolkit. Evaluation queries on →models serve to extract the model content (especially with regard to the simulation results). These can then be displayed in a clearly structured way in the →ADOxx browser. Depending on your particular →library configuration various pre-defined evaluation queries may be available.

Evaluation Queries, predefined. Predefined evaluation queries are defined by the →ADOxx administrator for →business process libraries and →working environment libraries. They are made available to the →ADOxx user by special menu options in the Evaluation Component of the →Modelling Toolkit. The results of predefined evaluation queries are displayed in the →ADOxx browser.

Execution Time. Time during which an →activity belonging to a →business process is executed.

Exit Button. Button in the top left of the →window panel, which closes the window when you double-click on it.

Export. →ADL-Export, →XML-Export.

Export Migration Assistant. Functionality in the →Administration Toolkit. The export migration assistant supports you in exporting →application libraries, →ADOxx users, →ADOxx user groups, →ADOxx models and →ADOxx model groups.

Expression. 1. EXPRESSION; attribute type which determines the value of an →attribute using a flexible calculation rule, whereas the attribute value is defined by other attributes.

2. →AQL expression.

Express Message. Specific form of a message (→ADOxx mail) to a logged in →ADOxx user. An express message will be immediately delivered to the receiver and shown in a popup window.

External Coupling. Coupling of an external tool to ADOxx.

- F -

FDL. **F**lowmark **D**efinition **L**anguage; language for describing →FlowMark models.

FDL Generation. Functionality of the →Import/Export Component in the →Modelling Toolkit. This functionality generates FDL files from →models (in order to then import them into →FlowMark).

Floating-Point Number. DOUBLE; attribute type for decimals (up to six decimal places).

FlowMark. →MQSeries Workflow.

Flowmark Audit Trail. Audit trail of →business processes carried out by →FlowMark.

- G -

Global Change. Simultaneous change of attribute values of several objects in one or more models. The attributes to be changed are selected by entering standardised or user-specific search criteria.

Global End. Terminates a →business process including those processes that call it (i.e. may appear within a subprocess). A global end represents the termination of the whole processing.

Groupware System. Computer system supporting teamwork by providing services like e-mail, document management, discussion forum and so on.

- H -

HOMER. →Acquisition Tables.

HTML. **H**yper**T**ext **M**arkup **L**anguage; file format, in which the contents of the →ADOxx browser or model content generated using the →Documentation Component can be saved. HTML files can be displayed in web browsers.

Human Resource Management (component). Specific definition of the →dynamical evaluation modules. Used for the period-related calculation (→time-related versioning) of the organisational/personnel capacity requirements.

- I -

Import Migration Assistant. Functionality in the →Administration Toolkit. The import migration assistant supports you in importing →application libraries, →ADOxx users, →ADOxx user groups, →ADOxx models and →ADOxx model groups.

Import/Export (component). →Component of the →Modelling Toolkit in which →ADL Import, →ADL Export and →FDL Transformation can be called.

Informix. →DBMS for Windows (owner: IBM).

Input Parameter Combination. Parameter for →simulation and analytical evaluation (calculation). When a Simulation or →Analytical Evaluation is started, an input parameter combination must be selected. This specifies on which →attributes the results should be calculated. Depending on the input parameter combination different results will be calculated. Additional input parameter combinations

can be defined in the →Administration Toolkit (→Library Management, →library attribute →"Simmapping").

Inspector. →ADOxx Inspector.

InstallShield. Software, using which you can install ADOxx via Windows (manufacturer: Macrovision Corp.).

Instance. 1. →Object

2. →Attribute Profile

3. →Process Instance.

Instantiation. 1. Creating an →object of a →class or a →connector of a →relation

2. Creating an →attribute profile of an →attribute profile class.

Integer. INTEGER; attribute type for an integer (whole number).

Inter-Model Reference. INTERREF; attribute type used to create a reference to another →ADOxx model or to an →object in a different →ADOxx model.

Internet. Worldwide computer network.

Intranet. Computer network, which is restricted to one organisation (company, department) - in contrast to the →Internet.

ISO9000. This umbrella term is commonly used for the series of norms ISO 9000 ff. These regulations of the International Standards Organisation (ISO) aim at creating a quality assurance system. They include a maximum number of twenty core elements. Depending on the type and size of a company's business operations, the company may acquire a certificate according to the norms ISO 9001, 9002 or 9003. An accredited certifier (= one recognised by the commission) awards this certificate. It certifies that the enterprise quality assurance system conforms with the core elements of the respective norm. The certificate according to ISO 9001-9003 is usually regarded as guarantee for high-quality products and/or services. It is a commonly applied criterion when selecting suppliers especially in the public services and industrial sector.

- J -

JPG. JPEG; graphic file format of the Joint Photographic Experts Group, in which graphics generated from →models can be saved.

- K -

Knowledge Management. Knowledge Management covers all activities of an organisation which deal with supplying and maintaining knowledge.

- L -

LEO. Meta-language for customising and specifying →attributes. Shows the contents of data structures in text form (→AdoScript).

LEOgram. Sequence of →LEO elements.

Library. Number of user-defined →classes for →modelling. Both →ADOxx-Default-BP-Library and →ADOxx-Default-WE-Library can be defined in ADOxx. Libraries are defined during ADOxx →customising.

Library Attributes. →Attributes of libraries.

Library Configuration. Functionality of the →Library Management in the →Administration Toolkit. Using the library configuration, you can edit and check class and library attributes (→customising).

Library Export. Export of an →application library into an →ABL file to transfer data into a different →ADOxx database or for backup purposes. This functionality of the →Library Management is located in the →Administration Toolkit.

Library Import. Import of an →application library from an →ABL file and then save it in the →ADOxx database. Functionality of the →Library Management in the →Administration Toolkit.

Library List. List that shows all the →business process and →Working Environment libraries stored in the →ADOxx database. From the library list you can import, export and delete libraries.

Library Management (component). →Component of the →Administration Toolkit, which is used to import and export →application libraries and to configure library attributes (= library configuration). The Library Management also contains an import and export migration assistant with the help of which the complete contents of the →ADOxx database (models, model groups, application libraries, users and user groups) can be exported (back-up) and imported into a different ADOxx database (data transfer).

Licence Number. Customer-specific number for configuring ADOxx and creating →ADOxx databases. A licence number consists of the encoded →customer number and the key for the available →components of the →Modelling Toolkit (depending on the customer-specific configuration of ADOxx).

Longstring. LONGSTRING; attribute type for text of up to 32.000 arbitrary symbols.

Lotus Notes. →Groupware and document management system (manufacturer: IBM).

LOVEM. **L**ine **o**f **V**isibility **E**ngineering **M**ethodology; business process modelling method of the IBM Corp.

- M -

Main Reference. Classification of →references to determine the referenced models (for instance when opening models including referenced models) outgoing from a model. Main referenced models will be processed like the start model, i.e. all references contained in this model will be followed, if the restriction of depth allows it. (see also →side reference).

Management Office. Software tool suite of BOC Asset Management GmbH. The Management Office consists of the Strategy and Performance Toolkit →ADOscore, the Business Process and Knowledge Management Toolkit →ADONIS, the Supply Chain Management Toolkit →ADOlog and the IT Service and Architecture Management Toolkit →ADOit.

Manual Random Generator. →Random generator, automatic.

Max. Length of Loops. Parameter of the →Analytical Evaluation. Determines how many →objects a loop (in a →model) may contain at most, before the evaluation is assumed as a never-ending loop.

Max. Number of Paths. Parameter of the →Analytical Evaluation. Determines how many different paths a model may contain before the evaluation is cancelled.

Max. Start Time Period. →Attribute of →objects of the →class "Activity". The maximum start time period specifies, for how long a →cooperative activity will at most be delayed until sufficient →performers are available for executing the cooperative →activity. This attribute is only evaluated by the simulation algorithm "→Workload Analysis".

Menu Bar. Panel between the →window panel and the →component bar/→quick-access bar in the →ADOxx window which contains the menu options for the activated →ADOxx component.

Messages. →ADOxx mail.

Meta Model. Describes the modelling classes of a model type and their dependences.

Method Library. →Application library.

Microsoft SQL Server. →DBMS for Windows (owner: Microsoft Corp.).

Migration. Transfer of an →application library, →ADOxx database or ADOxx version to a new or changed ADOxx application library, database or version.

Model. →ADOxx model.

Model Access. →Access right.

Model Attribute. →Attribute of a model.

Model Comparison. Comparison of two →models. Differences found will be displayed in the →ADOxx browser.

Model Editor. Graphically oriented editor in the →Modelling Component, used for creating →models.

Model Group. →ADOxx model group.

Model Group Management. Functionality of the →Model Management in the →Administration Toolkit. With the help of the model group management you can create, rename, move or delete →ADOxx model groups, create, copy, move or delete →model references as well as define →access rights of →ADOxx user groups to ADOxx model groups.

Model History. Supports the navigation between →models. For instance, you may switch between a main model and its sub model.

Model Management (component). →Component of the →Administration Toolkit, in which the →model access can be defined.

Model Pool. Quantity of all →ADOxx models saved in the →ADOxx database which are not assigned to any →ADOxx model group. Can be accessed via →model group management.

Model Reference. 1. Reference of an →ADOxx model group to an →ADOxx model in the →model group management. ADOxx models may be referenced in any number of ADOxx model groups.

2. →Inter-model reference (INTERREF) to an →ADOxx model, created in the →Modelling Component.

Model Search. Functionality of the →Modelling Component of the →Modelling Toolkit. Using the model search, you can search for models that are stored in the →ADOxx database. You can either search according to →model attributes or within the →process hierarchy.

Model Type. In ADOxx version 1.0 one or several model types can be defined in a →ADOxx-Default-BP-Library and in a →ADOxx-Default-WE-Library. In the →ADOxx-Default-Library we distinguish between →Business Process Models, →company maps (business process library) and →working environment models (working environment model).

Model Window. Window in which a →model is displayed and edited.

Modelling (component). →Component of the →Modelling Toolkit, in which →Business Process Models and →Working Environment models are created, using the →model editor.

Modelling, bottom-up. Approach to creating →models. When modelling bottom-up you start with detailed small models ("micro perspective") and gradually combine them to a bird's-eye view on the process in question by reducing their complexity (→modelling, top-down).

Modelling, top-down. Approach to creating →models. When modelling top-down you start with a bird's-eye view on a given process and gradually go into more detail ("micro perspective") (→Modelling, Bottom-Up).

Modelling Bar. Button panel, by default on the left-hand side of the →ADOxx window. The modelling bar consists of the →edit button, the →empty space tool, the →class panel and the →relation panel.

Mode. A mode is a (sub)set of the →objects and →connectors contained within an →ADOxx model. By selecting different modes (menu "View"), some extra objects may be made available and some existing objects may be hidden.

Mouse Access. It is possible to block mouse access for each →class and →relation during →modelling separately. This means that none of the →objects of that particular class or none of the →connectors of the particular relation can be selected.

MQ Series Workflow. →WebSphere MQ.

MSDE. Micro**s**oft **D**atabase **E**ngine; →DBMS for Windows (owner: Microsoft Corp.).

MS SQL Server. →DBMS for Windows (owner: Microsoft Corp.).

- N -

Navigator. →ADOxx Navigator.

Notebook. →ADOxx Notebook.

Notes. →Lotus Notes.

- O -

Object. Instantiation of a →class as for example "Specialist" could be an instantiation of the →class "Role".

Object Attribute. →Attribute of an →object.

Object Reference. Specific form of an →inter-model reference to an →object in the same or in a different →ADOxx model.

objectiF. Object-oriented →UML →CASE tool (owner: microTOOL GmbH).

Offline Animation. The offline animation enables you to exactly replay an earlier simulation run based on a →simulation protocol. During this, the performers' →task stacks are →animated. →Agents may also be employed during the offline animation. When generating the →simulation protocol the format "short version" must be specified in order to create a file (*.apf) that can be used by the offline animation function.

Online Manual. Electronic ADOxx manual, i.e. a manual that can be displayed on the screen by the help system. In ADOxx there are online manuals for the →Modelling-Toolkit, the →Administration Toolkit and the →ADOxx Database Administration.

Output-induced (OI) Process. Process causing costs that vary depending on the amount of work that must be executed by the cost centre. A →cost driver exists.

Output-neutral (ON) Process. Process causing costs which are independent of the amount of work that must be executed by the cost centre. No →cost driver exists.

Oracle. →DBMS for Unix, Windows and other operating systems (owner: Oracle Corp.).

- P -

Page Area. Visualisation of the →page layout in the →model editor of the →Modelling Toolkit. The page area is marked by a broken line. The page area is most important when graphic files are generated.

Page Layout. Description of the size of a printed page. The section available for the graphic model information and the headers and footers per page layout can be defined in the →library configuration of the →Administration Toolkit. The size of a printed page is especially taken into account when graphic files and →connector marks are generated.

Path. Sequence of objects (→activities, decisions etc), which describe one possible way in which a process can be executed.

Path Analysis. Simulation algorithm in the →Simulation Component of the →Modelling Toolkit, which "plays" through →Business Process Models. Serves to calculate process-specific key figures such as "Expected Cycle time" and "Expected Costs" and to determine which paths are critical with regard to various criteria selected by the →ADOxx user.

PCX. ZSoft Paintbrush graphic file format, in which graphics generated from →models can be saved.

PDF. Portable **D**ocument **F**ormat; file format, using which documents can be shown independently of the hardware, the operating system and the application software used for the creation (owner: Adobe Systems Incorporated). The complete ADOxx user documentation is contained in PDF on the ADOxx installation CD.

Performer. Actor within a →business process. Performers are represented in ADOxx as →objects of the →class "Performer" in →Working Environment models. It is possible to specify the performer (or set of performers) who can execute a certain →activity using the "Performer" →attribute in an activity's notebook.

Performer Calendar. Description when a →performer is present and available for executing →activities. A performer's availability can be described in detail down to seconds and refers to one year. Performer calendars are evaluated in the simulation algorithm "→Workload Analysis".

Performer Assignment. In ADOxx, →business processes and →Working Environments are modelled separately. To assign →activities to performers, →AQL expressions are entered in the attribute "Performer" in objects of the class "Activity". These are interpreted during simulation to assign an actual →performer to an →activity.

Plans, pre-defined. Pre-defined plans are a specific type of result representation for →pre-defined queries. If a →query's result are all →objects of a →class within a →model - such as all →activities of a →business process model - the lines of the result table will only show the model name. The columns of the result table will be empty. Pre-defined plans may e.g. be used for generating task plans. They are defined by the →ADOxx administrator (during →library configuration) for →Working Environment libraries and →business process libraries) and are then available to the →ADOxx user, together with the →pre-defined queries in the →Analysis Component of the →Modelling Toolkit.

PNG. Portable **N**etwork **G**raphics; Graphic file format, in which graphics generated from →models can be stored.

Pre-defined Evaluation Queries. →Queries, pre-defined.

Pre-defined Plans. →Plans, pre-defined.

Pre-defined Queries. →Queries, pre-defined.

Prioritisation. Process, using which the →ADOxx user can select the next →activity to be edited during the →simulation.

Process Calendar. Describes the stochastic occurrence of →business processes in time intervals. An example of stochastic occurrence is seasonal fluctuation. Process calendars are evaluated by the simulation algorithm "→Workload Analysis".

Process Hierarchy. Call hierarchy of →Business Process Models. A hierarchy may be structured according to →business processes calling or to →business processes called. Within a process hierarchy, models can be searched (→model search). The process hierarchy can be displayed graphically or in tabular form.

Process Instance. Specific instance of a →business process during the →simulation.

© BOC Group 2008

Program Call. PROGRAM CALL; attribute type for launching an external program (such as Microsoft Excel).

Program Icon. A program icon can be found in the →ADOxx program folder for every →ADOxx component and for every program of the →ADOxx database administration installed. A program is started by clicking on the particular program icon.

- Q -

Quality Assurance. The basic objective of quality assurance measures (or QA systems) is to guarantee a constant quality level with regard to a company's products and/or services. There is a fundamental difference between two different approaches of quality assurance: active vs. reactive methods. The latter attempts to assure the product quality by introducing control mechanisms. These, however, just fight the consequences of faulty processes, i.e. the poor/defective products. Newer approaches have tried to develop optimised procedures to actively affect the products' quality. These active quality assurance approaches are known as →Quality Management and find their expression in a great number of methods, concepts and canons of rules. These include Total Quality Management (TQM) in particular or the series of norms →ISO 9000 ff.

Quality Management. Quality Management is an umbrella term for a number of active quality assurance methods. Active QM approaches - in contrast to approaches of reactive →Quality Assurance - are those the purpose of which is to avoid "quality gaps" in the products and/or services of a company. An essential part of the quality management is thus the documentation and (re-)structuring of →business processes).

Quantity. →Attribute of →objects of the →class "Process start". Specifies how often the →business process occurs. The quantity refers to the value of the →attribute "Time Period". Both attributes are used in the →Capacity Analysis simulation algorithm.

Queries. Functionality of the →Analysis Component in the →Modelling Toolkit. By running queries on →models, the model contents can be extracted and be displayed in a clearly structured way in the →ADOxx browser. The following types of queries are available: →user-defined, →standardised and (depending on the particular →library configuration) →predefined ones.

Queries, predefined. The →ADOxx administrator defines predefined queries for →business process libraries and →Working Environment libraries in the →Library Management component of the →Administration Toolkit. The predefined queries are made available to the →ADOxx user in the →Analysis Component of the →Modelling Toolkit by specific menu options. The results of the predefined queries are displayed in the →ADOxx browser.

Queries, standardised. Standardised queries are available to the →ADOxx user in the →Analysis Component in the →Modelling Toolkit. Standardised queries may be used to create →user-defined queries. The results of the standardised queries are displayed in the →ADOxx browser.

Queries, user-defined. The →ADOxx user defines user-defined queries in the →Analysis Component of the →Modelling Toolkit by using →AQL expressions. The results of the user-defined queries are displayed in the →ADOxx browser.

Quick-access Bar. Panel, by default below the →menu bar in the →ADOxx window. The buttons (smart-icons) visible depend on the →component selected. They provide quick access to certain functions of the →component. To the left of the quick-access bar, usually the →component bar can be found.

- R -

R/3. Standard software suite (manufacturer: SAP).

Random Generator, automatic. The assignment of variables during the →simulation run is automatically using the defined →distributions.

Random Generator, manual. The ADOxx user can select the variable assignment during the →simulation run.

Rational Rose. Object-oriented →CASE tool (owner: IBM).

RDBMS. Relational database management system. →DBMS, which is based on the relational data model.

Read Access. →Access right to an →ADOxx model group and the →ADOxx models referenced within this group. With this type of access right, you may view models **locally** but do not have access to make any changes.

Record. RECORD; attribute type which enables a flexible list/record administration of assembled file types (→attributes).

Reference Model. Consists of standardised reference processes (templates) that represent operational activities generally valid or necessary. Reference models focus on branch or software-specific aspects and can be adapted (customised) to company-specific circumstances.

Reference Process Model. Process from a →reference model.

Relation. General connection between two →classes. →Objects of a →class can be linked by concrete →instantiations) of this →connection), for example, two organisational units may be linked by the relation "Is subordinated".

Relation Attribute. →Attribute of a →relation.

Relation Panel. Button panel that contains exactly one button for each relation that can be modelled in the active context. It is the lowest part of the →modelling bar, following the →edit button and the →class panel.

Relation Table. Functionality available in the →Analysis Component of the →Modelling Toolkit. A relation table represents →connections (of a relation) inside a →model. The →objects (start and target objects), which are linked by the →connectors, are displayed in the row or column headers of the table. Which relation tables are available depends on the respective →library configuration.

Repository. Central database for the administration of all →ADOxx users, →ADOxx application libraries, →ADOxx models etc.

Representation, graphical. Possibility of visualising an ADOxx model in graphical form.

Representation, tabular. Possibility of visualising an ADOxx model in tabular form.

Resource. Part of a →Working Environment which is needed for executing →activities, such as PC, printer or fax.

Resource Assignment. In ADOxx →business processes and →Working Environments are modelled separately from each other. For the →Simulation to take account of →resources, →AQL expression(s) must be entered in the "Selection" attribute of →objects of the →class "Resource" in business processes (support available for this). These are interpreted during the →Capacity and →Workload Analysis s. In this way, the resources in the →Working Environment which are needed to execute an →activity are determined.

Resting Time. Time during which an →activity of a →business process has been executed but has not yet moved on to be transported to the next stage in the process.

Result Function. Function which is assigned to an →agent. By this, the agent is instructed to calculate a certain simulation result.

RTF. Rich **T**ext **F**ormat; file format, in which the contents of the →ADOxx browser or model contents which were generated using the →Documentation Component can be saved. RTF files can be further edited in word-processing programs.

- S -

SAP. **S**ystems, **A**pplications, **P**roducts in data processing. Owner of the standard software →R/3.

Server. A server is a software module which puts its services at the disposal of other software modules, the →clients.

SGML. **S**tandard **G**eneralized **M**arkup **L**anguage; SGML files are used in the →Documentation Component to transfer the documented model contents. In a second phase, the SGML files will be then converted to any target format (e.g. →HTML, →RTF, →XML) using a →DSL file.

Side Reference. Classification of →references to determine the referenced models (for example when opening models including referenced models) starting from a model. Outgoing references from side-referenced models will only be followed, if these references start from the same attribute and if no depth restrictions prevent it.

Sim Mapping. Attribute of a →business process library. In the sim mapping the →input parameter combinations are defined. In addition, the sim mapping defines which →classes should be to totalled for aggregated results, for the →assignment of performers and for the reference objects of the →agents.

Simulation (component). →Component in the →Modelling Toolkit, which simulates →business processes and →Working Environments. ADOxx contains the simulation algorithms "→Path Analysis", "→Capacity Analysis", "→Workload aAnalysis (steady state)" and "→Workload Analysis (fixed time period)".

Simulation Cache. The simulation cache stores the simulated →ADOxx models in the internal representation necessary for →simulation. Thus, the models need not be transferred into the internal representation again, if you start the simulation anew.

Simulation Log. Report (protocol) of a simulation run in the Simulation Component.

Simulation Log, short version. Compressed form of the →simulation log; used by the →offline animation function.

Simulation Log, long version. Detailed form of the →simulation log for replaying each step of the →simulation.

Simulation Run. Processing of a →process instance during the →Simulation.

Single Sign-on (SSO). Functionality in ADOxx which enables that the Windows system user can be imported into the ADOxx user management and therefore allow →system users working and logged in to the →domain to log into ADOxx without providing user name and password again.

Smart-Icon. Graphical buttons in the →component bar, the →quick-access bar etc.

Snap Grid. Grid according to which →objects and →bendpoints can be aligned on the →drawing area.

SQL. **S**tructured **Q**uery **L**anguage. Query language for relational databases.

SQL Server. →Microsoft SQL Server.

Staffware. →Workflow Management System (owner: TIBCO, Inc.).

Stand-alone. A single computer system.

Standard Model Group. →ADOxx model group, which is created automatically with the →ADOxx database. The standard model group has the name "Models".

Standard User. →ADOxx user who is created automatically together with the →ADOxx database. The standard user has the user name "Admin" (default password is "password") and cannot be deleted.

Standard User Group. →ADOxx user group, which is created automatically with an →ADOxx database. The standard user group has the name "ADOxx" and cannot be deleted.

Standardised Queries. →Queries, standardised.

Swimlane. A swimlane is a modelling construct to describe roles, organisational units etc. A swimlane is visualised by vertical and horizontal areas on the drawing area. The characteristic of a swim lane is that when changing its size you will also move the →objects and →relations placed on this swimlane.

System Administrator. Person responsible for the IT infrastructure (hardware, network, software installations and so on).

System User. User of the operating system (Windows) who is authenticated by logging into a →domain. In general the system has to be created by a →system administrator within the system user administration in order to be able to work with and log in to the system. System users can log in to ADOxx via →single sign-on (SSO), if this feature is provided.

System User Administration. Part of an operating system used to create, delete or change the settings of →system users, create, rename and delete →system user groups, as well as assign system users to system user groups.

System User Group. A group containing →system users. A system user group has specific access and execution rights to components of the operating system which are defined by the →system administrator.

- T -

Tabular Representation of a Model. →Representation, tabular.

Task Stack. Stack of →activities a →performer has to execute. The size of the task stack is determined in the simulation algorithm →"Workload Analysis".

Task stack, central. The activities in the central task stack will be executed by the first performer who has time for it (→prioritisation).

Task Stack, personal. The activities in the personal task stack of a performer are exclusively executed by this performer.

Text. STRING; attribute type for text containing up to 3.700 arbitrary symbols (exception: attribute "Name" which is restricted to at most 240 symbols).

Time. TIME; attribute type for the ADOxx time format YY:DDD:HH:MM:SS (Years:Days:Hours:Minutes:Seconds).

Time, company. Division of time referring to the presence of a →performer in the company. A (performer's) company time may for example consists of 170 days a year and 8 hours a day.

Time, real. Conventional division of time, i.e. a year has 365 days and a day has 24 hours.

Time Period. →Attribute of →objects of the →class "Process start". The time period specifies for which period of time the value of the →attribute "→Quantity" is valid. (Used in capacity analysis).

Tolerance Waiting Time. Maximum waiting time before a →process instance is aborted unedited during the →simulation.

Transformation. →FDL-Generation.

Transport Time. Time needed for transporting files, documents etc.

TXT. ASCII-TEXT; file format in which ordinary text files are saved.

© BOC Group 2008

- U -

UDL. **U**ser **D**efinition **L**anguage; file format in which →ADOxx users and →ADOxx user groups are saved (exported) from the →Administration Toolkit.

UDL Export. Functionality of the →User Management in the →Administration Toolkit. With UDL export, →ADOxx users and →ADOxx user groups can be exported into →UDL files.

UDL Import. Functionality of the →User Management in the →Administration Toolkit. With UDL import →UDL files (→ADOxx users and →ADOxx user groups) can be imported into ADOxx and then stored in the →ADOxx database.

UML. **U**nified **M**odeling **L**anguage; method of object-oriented modelling of complex software systems.

Use Case Diagram. **1.** →UML diagram type. Interactions between a system and external people involved are described in Use Case Diagrams.

2. Model type of the →ADOxx-Default-BP-Library.

User. →ADOxx user.

User Account. User name and password for logging into the →DBMS or the operating system.

User Assignment. Assignment of an →ADOxx user to one or more →ADOxx user group(s) in the →User Management of the →Administration Toolkit. This assignment makes it possible for the user to access →ADOxx model groups and →ADOxx models within the Modelling Toolkit.

User Export. Export of →ADOxx users and/or →ADOxx user groups into a →UDL file in order to transfer the data into a different →ADOxx database or make a backup. Functionality of the →User Management in the →Administration Toolkit.

User Group. →ADOxx user group.

User Group List. List of all →ADOxx user groups stored in the →ADOxx database as well as all →ADOxx users referenced in them. With the help of the user list, ADOxx user groups can be added, renamed and deleted. →user assignment can also be edited.

User Import. Import of →ADOxx users and/or →ADOxx user groups from a →UDL file into the →ADOxx database. Functionality of the →User Management in the →Administration Toolkit.

User List. List of all →ADOxx users stored in the →ADOxx database. With the help of the user list, you can add and delete →ADOxx users as well as edit their settings.

User Management (component). →ADOxx User Mangement.

User-defined Queries. →Queries, user-defined.

- V -

Validity Date. Specifies the time from which a model or attribute profile version becomes valid. The validity of a version ends with the validity start of the next version of the same model or attribute profile.

Versioning. Enables the administration of different versions of a →model. in ADOxx, either a →model-related or a →time-related versioning is possible.

Versioning, model-related. A version number is assigned manually to each version of a →model.

Versioning, time-related. A →validity date is assigned to each version of a →model.

Visio. Graphical modelling tool (Owner: Microsoft Corp.).

Visualised Attribute Value. The →attribute value of an →object or a →connector which is displayed on the →drawing area (g.g. Name of an "Activity", transition condition of a "Subsequent" relation).

- W -

Waiting Time. Time during which an →activity of a →business process cannot yet be executed, as the person due to execute it is still busy working on something else; i.e. the activity is ready to be executed but must first queue for a while.

WebSphere MQ. →Workflow Management System, successor of FlowMark and MQ Series Workflow (owner: IBM).

Width Search. Process to go through objects, i.e. to automatically process the objects and relations of a model in the RTF generation in the →Documentation Component. Starting from a "Process start" object, all objects which are directly connected to this start object, will be processed gradually. When all objects which are directly connected to this start object have been processed, one of these directly connected objects will be selected as a new start object and the process will recur. This will be carried out for all objects. (see also →depth search)

Window Panel. Title bar of an →ADOxx component's window (→ADOxx window).

Windows System User. →System user.

Workflow Management System (WMS). Computer system that controls the execution of →business processes. The most prominent task of a Workflow management system is to delegate business process activities to the organisation's employees.

Working Environment. (= organisational structures); structures (e.g. departments and roles), →performers and →resources of an organisation. The working environment of an organisation is represented in ADOxx in →Working Environment models.

Working Environment Library. Contains →classes and →relations for →modelling the →Working Environment as well as description, layout, analysis and simulation definitions. The →classes of a Working Environment library are derived from the classes of the Working Environment meta model. A →Working Environment model always refers to exactly one Working Environment library. Working Environment libraries are defined during ADOxx →customising.

Working Environment Meta Model. Pre-defined →classes for →modelling Working Environments in ADOxx. The classes in a Working Environment library are derived from the classes contained in the Working Environment meta model.

Working Environment Model. 1. →Model of a Working Environment. Results from instantiating →classes of a →Working Environment library.

2. Model type of the →ADOxx-Default-WE-Library.

Workspace (Working Area). Main part of the ADOxx window, usually situated below the →component bar/→quick-access bar. The →modelling windows are displayed in the workspace.

Workload Analysis. Simulation algorithm in the →Simulation Component of the →Modelling Toolkit, based on queuing theory. The simulation runs "on the time axis". The results are the number of →cycle times and →waiting times depending on the capacity/amount of work of →performers and →resources. The input for the Workload Analysis is an →application model. A Workload Analysis can be run based on a →steady state or a →fixed time period basis. The results of the Workload Analysis are displayed in the →ADOxx browser.

Workload Analysis, fixed-time period. →Workload Analysis, in which the length of the initialisation phase and the calculation (result measuring) phase and the types of results calculated (input parameters) are defined by the user.

Workload Analysis, steady-state view. →Workload Analysis, in which the length of the oscillation phases and the calculation period depend on the number of simulation runs.

Write Access. →Access right granted to an ADOxx user group to an →ADOxx model group and the →ADOxx models referenced within it. With this access right, you can locally change as well as save a model's content.

© BOC Group 2008

- X -

XML. E**X**tensible **M**arkup **L**anguage; file format in which →ADOxx models can be stored (exported).

XML Export. Functionality in the →Import/Export Component of the →Modelling Toolkit. Using the XML export, →ADOxx models can be exported to →XML files.

XML Import. Functionality in the →Import/Export Component of the →Modelling Toolkit. Using the XML import,→XML files (→ADOxx models) can be imported to ADOxx and stored in the →ADOxx database.

19. Index

A

ABL files 754
ACR file 754
Activate referened model 255
Activity Times 335
  alternative definition 337
  Input parameter 337
  Standard definition 336
Add attribute profile reference 208
Add model references 207
Add object references 208
Add references 207
  models 207
  objects 208
Additional components 497
ADL export 468
  Application models 472
  Attribute profile groups 468
  Attribute profiles 468
  for ADONIS Version 3.0 468
  Model groups 468
  Models 468
ADL Import 445
  Application Models 464
  Attribute profile groups 446
  Attribute profiles 446
  Examples 454
  Model groups 446
  Models 446
  Options 454
ADL-file 755
ADOxx 10
  component bar 37
  Ergonomics research 803
  Explorer 38
  Hint 12
  Inspector 39
  menu bar 37
  Navigator 38
  New features in version 1.0 4
  Product palette 19
  quick-access bar 37
  Quick-access bar 37
  search results 39
  user interface 36
  window bar 36
  Workspace 38
ADOxx 30
ADOxx Administration Toolkit 15
  Attribute Profile Management 15
  Component Management 15
  Library Management 15
  Model Management 15
  User Management 15
ADOxx attribute types 733
  Attribute profile reference 733
  ATTRPROFREF 733
  Calendar 735
  DATE 734
  Date and time 734
  Dates 734
  DATETIME 734
  DOUBLE 735
  Enumeration 733
  Enumeration list 734
  EXPRESSION 734
  Expressions 734
  Floating point 735
  Integer 735
  Interref 736
  Long string 735
  Program call 736
  RECORD 736
  Records 736
  String 736
  Time 736
ADOxx browser 66
  adjust column width 75
  Adjust row height 76
  Align attribute values 79
  Classification 66
  Copy to clipboard 84
  Diagram 85
  editable 69
  Enter column width 74
  Enter row height 75
  Expand branches 76
  formats (save) 82
  hierarchical 70
  Print 84
  Representation of attribute values 71
  save 81
  search 80
  Select attributes 77
  Select columns 77
  settings (save) 82
  Show (attribute) values 79
  Shrink branches 76
  sort 78
  sort by attribute columns 78
  Structure 68
ADOxx Explorer 54
  Activate model 59
  close model 59
  Copy model 60
  Copy model group 60
  Copy model references 60
  Create new model 58
  Create new model group 58
  Cut model 60
  Cut model group 60
  Delete model 60
  Delete model group 60
  Delete model references 60

Hide all 58
Item Search 61
Manage model groups 61
Models in the database 56
Open model 59
Opened model windows 57
Options for the Opening of Models 57
Paste model 60
Paste model group 60
print models 60
Refresh 57
Rename model 59
Rename model group 59
save model 59
Save tree structure 61
Set Visible Model Types 58
Show all 58
Show selected with sublevels 58
ADOxx Mail 745
Display 749
Forward 748
Read 751
Read instant message 752
Receive instant message 752
Receive new 749
Received message 749
Reply 747
Select receiver 746
send 745
Settings 752
ADOxx manuals 2
ADOxx Modelling Toolkit 17
Acquisition 17
Analysis 17
Evaluation 17
Import/Export 17
Modelling 17
Simulation 17
ADOxx Navigator 62
ADOxx Notebook 48
ADOxx standard application library 802
Pre-defined Agent Types 394
ADOxx standard BP library 802
ADOxx standard WE library 802
ADOxx-3c 498
- Establishing costs of processes 500
Acquiring process costs rate 501
actual process cost analysis 499
Assessment of the cost centres' capacity 504
Carrying out 504
idle capacities 503
Input 499
Method of working 500
ON processes 503
Plan process cost analysis 499
Simulation 500
ADOxx-Inspektor 64
Agent Result Functions 400
Animation 401
Capacity 405
Cycle Time 406
Execution Costs 403
Execution Costs (conditional) 404
Frequency 402
Number of changes in responsibility 401
Special formula 407
Sum over Attribute 409
Workload 405
Agent Results 424
Additional Display Options 428
Available Results 426
Display mode 426
Information 428
Scope 427
Set time period 429
Volume 427
Agents 391
Adjusting Reference Objects 395
Agents Overview 392
Basic configuration 397
change 391
Configure Result Functions 410
Configuring 396
create 391
Define Agent Types 394
Employ in Simulations 422
Represent Agents Results graphically 430
Represent Agents Results in Tabular Form 430
Result Functions 400
Selection of results 424
Align objects 160
allcattrs (expression) 758
alliattrs (expression) 758
allobjs (expression) 758
Allocating resources 227
allrattrs (expression) 758
amax (expression) 758
Analysis 17, 306
Analytical Evaluation 325
on attribute profiles 299
Predefined queries and plans 318
queries 306
Relation tables 322
reports 306
Analytical Evaluation 325
background information 329
display results 327
function 330
input 329
Results 327
Analytical Evaluation - Display results 327
Analytical Evaluation - display results (browser) 329
Animation of Results 423
APF-file 755
Application models 284
Add application model 285
Delete application model 288
Edit application model 287
Open application model 285
rename 287
AQL 790
Examples 794
Rules 792
Semantics 790
Syntax 790
aql (expression) 758
Arrange automatically 160
Arrange manually 162

Add arrangement function 173
Bendpoints 164
Copy arrangement function 173
Crossing-minimisation 165
Editing class specific parameters 171
Optimising positions 166
Orientation 168
Preview 169
Use model 170
Arrangement function (automatic) 160
Asessment 303
Assessing Business Process Models 303
Results 305
settings 304
Assessment 304
Assign models to a model group 271
Assigning performers 222
Assigning performers to sub processes 225
Hierarchical working environment models 224
Probabilities 226
Assigning performers to sub processes 225
Assigning sub processes 221
asum (expression) 758
Attribute and class filter (Documentation) 493
Load mode 495
Attribute profile 291
add 294
Carry out queries 299
copy 297
delete 298
edit 295
edit simultaneously 296
move 298
Query results 302
save as new version 295
show 296
show simultaneously 297
show usage 298
Sort query results 302
Standardised queries 299
user-defined queries 301
attribute profile folder 291
add 292
copy 293
delete 294
move 294
rename 293
Attribute Profile Management 15
Attribute profile reference (Attribute type) 733
Attribute profiles 100
attrname (expression) 758
ATTRPROFREF 733
atttype (expression) 758
aval (expression) 758
avalf (expression) 758
awsum (expression) 758

B

Bar diagram 87
Settings 88
Bendpoints 133
Bendpoints (connectors) 197
Bendpoints in connectors 133
BMP-file 755
Bridges (connectors) 197

C

Calculating times and costs 303
Results 305
settings 304
calculating times and costs (BP model) 304
Calendar 735
Calendar(attribute type) 735
Capacity Analysis 354
Application Scenario (example) 366
Background 362
Carry out 355
Input 362
Processing Mode 363
Results 357
Carrying out an Analytical Evaluation 326
Carrying out the process cost analysis 504
case/4/0 interface 517
Changing references 519
concept 517
Creating new references 519
Deleting references 519
Tracking references 518
cfobj (expression) 758
cfobjs (expression) 758
Change connector course 144
Change object size 141
change password 744
Change values of connector attribute 154
absolute 251
append 253
percentage 254
place first 252
relative 253
replace 252
replace (program call) 252
statistical 254
swap 251
take over 251
Change values of object attribute 154
absolute 251
append 253
percentage 254
place first 252
relative 253
replace 252
replace (program call) 252
statistical 254
swap 251
take over 251
Change visualised attribute values (Quick Edit) 148
Check cardinalities 264
class (expression) 758
Class panel 129
Close model 259
closing ADOxx 727
Codepages 738
colour names 801
Company Time 335
Compare models 273
Model selection 273

Results 274
Show differences in values 275
Comparison of Results 433
Add File 435
Display/represent graphically 436
Information on the results 436
Selection of results 434
Tabular representation 437
Component Management 15
components 107
Configure Result Functions 410
Animation 412
Capacity/Strain 417
Cycle Time 418
Execution costs 414
Execution Costs (conditional) 415
Frequency 413
General Settings 410
Number of Changes in Responsibility 413
Special formula 419
Sum over attribute 421
Workload 416
conn (expression) 758
Connector edges 197
Connector marks 197
Rename 146
Context menu 173
Drawing area 174
The connector context menu 176
The object context menu 174
Control elements 41
"Add" icon 50
"Delete" icon 50
"Dialog" icon 49
"Execute/Follow" icon 50
"Expression Dialog" icon 50
"Info" icon 49
"Large text field" icon 49
"Print" icon 49
"Save as" icon 49
Deselect 46
Hide all 45
Icon "File dialog" 50
Icon "Notebook" 50
in ADOxx Notebooks 48
in selection windows 41
Item search 43
Model types 42
Refresh 43
Save as 45
Selected items 46
Show all 45
Show all selected items 46
Show selected 45
Show selected with sublevels 45
Shrink selected 45
Shrink/Expand 45
Shrink/Expand version threads 47
Convert object 155
Cooperative Activities 340
Copy model graphic in an other application 156
model area 156
Model pages 158
Copy model reference to model group 273
Copy objects/connectors (ADOxx clipboard) 149
Create a new model 256
Create model group 270
Crossings (connectors) 197
CSV file 755
ctobj (expression) 758
ctobjs (expression) 758
current configuration 729
Cut objects/connectors (ADOxx clipboard) 150
Referenced objects 150
Cycle Time 339

D

DATE 734
Date (attribute type) 734
Date and time (attribute type) 734
DATETIME 734
Define colours 203
Define object size/position 142
Define zoom value 190
Defining sub processes 221
Definition of object size and position 142
Delete model group 271
Delete model reference 273
Delete objects/connectors 152
Referenced object 152
Delete Simulation Results 390
Deleting models 283
Show referenced models 283
Diagram 85
generate graphics 90
Display reference to active object 268
Distribution (overview) 212
Documentation 486
Attribute and class filter 493
Generate 486
Options 488
DOUBLE 735
double (EXPR type) 758
drawing area 176
Adding empty space 178
edit 176
increase 176
reduce 176
resize 176
Drawing connectors 132
Drawing mode 131
Drawing objects 132

E

Edges (connectors) 197
Edit attribute profile values 295
Allocating resources 227
Assigning performers 222
colours 203
Date value 205
Distribution (overview) 212
Edit date and time 206
Performer calendar 228
Probabilities 226
Process calendar 235
random generator 211

records 202
Reference 207
Times 206
Transition conditions 215
Transition probabilities 220
Edit date and time value 206
Edit date value 205
Edit record attributes 202
Copy attribute value 202
Copy cell value 202
Delete rows 202
Move rows 202
Paste rows 202
Edit reference depth 98
Edit several attribute profile values 296
Editing connectors 136
Align 160
Arrange manually 162
Automatic object arrangement 160
Bendpoints 197
Bridges 197
Change 144
Change attribute values 154
Change visualised attribute values 148
Connector edges 197
Copy 149
Crossings 197
Cut 150
Define size and position 142
Delete 152
Drawing connectors 132
Editing attributes 146
Generate graphics 156
Managing in ADOxx Notebooks 146
Move connector centre 148
Paste 151
Reallocation 145
Rename marks 146
search in models 152
Select 138
Editing attribute profiles 291
Editing attributes in ADOxx Notebooks 146
Editing mode 131
Editing objects 136
Align 160
Arrange manually 162
Automatic object arrangement 160
Change size 141
Change attribute values 154
Change visualised attribute values 148
Convert 155
Copy 149
Cut 150
Cut referenced objects 150
Delete 152
Delete referenced object 152
Drawing objects 132
Editing attributes 146
Generate graphics 156
Generate names 153
Moving 139
Number 155
Paste 151
Paste referenced objects 151
search in models 152
Select 138
Editing times 206
EMF-file 755
Empty space tool 178
add 178
remove 178
Enter program call 205
Enumeration 733
Enumeration (attribute type) 733
Enumeration list 734
Enumeration list (attribute type) 734
Ergonomics in ADOxx 803
Conformity with the Windows User Interface Style Guide 813
Dialogue design 806
Human criteria 803
Error Messages 520
[aacquiui-02] 521
[aactana-01] 521
[aalkwins-01] 521
[aalkwins-02] 522
[aalkwins-08] 522
[aalkwins-10] 522
[aalkwins-11] 522
[aanabrw-01] 522
[aanabrw-02] 522
[aanactrl-02] 523
[aanadlg-04] 523
[aanadlg-05] 523
[aanadlg-06] 523
[aanaeval-01] 523
[aanaeval-02] 523
[aanaeval-03] 524
[aanaeval-04] 524
[aanaeval-05] 524
[aanaeval-06] 524
[aanaeval-07] 524
[aanaud-01] 525
[aanaud-03] 525
[aanaud-04] 525
[aanaud-05] 525
[aanaud-06] 526
[aanaud-07] 526
[aanaud-08] 526
[aanaud-09] 526
[aanaud-10] 526
[aanaud-11] 527
[aanaud-12] 527
[aanaud-13] 527
[aanaud-14] 527
[aanaud-15] 527
[aanaud-16] 528
[aanaud-17] 528
[aap-01] 528
[aap-02] 528
[aap-03] 528
[aap-04] 529
[aap-05] 529
[aap-06] 529
[aap-07] 529
[aap-08] 529
[aap-09] 530
[aap-10] 530
[aap-11] 530
[aap-12] 530

© BOC Group 2008

[aap-14] 530
[aap-15] 530
[aapedit-01] 531
[aapedit-02] 531
[aapedit-03] 531
[aappmod-03] 531
[aappmod-10] 531
[aappmod-11] 532
[aappmod-12] 532
[aappmod-13] 532
[aapqury-01] 532
[aapqury-02] 533
[aapqury-03] 533
[aapqury-04] 533
[aapview-01] 533
[abmpsup-01] 533
[abmpsup-02] 534
[abrsimpl-06] 534
[acaledit-01] 534
[acaledit-02] 534
[acaledit-03] 534
[acaledit-04] 534
[acalui-01] 535
[acalui-02] 535
[acalui-03] 535
[acalui-04] 535
[acalui-05] 535
[acalui-06] 536
[acalui-07] 536
[acalui-08] 536
[acalui-09] 536
[acalui-10] 536
[acalui-13] 537
[acalui-14] 537
[acalui-15] 537
[acalui-16] 537
[acalui-17] 537
[acalui-18] 538
[acalui-19] 538
[acalui-20] 538
[acard-02] 538
[acard-03] 538
[acard-04] 539
[acard-05] 539
[acard-06] 539
[acard-07] 540
[acard-08] 540
[acard-09] 540
[acase40-01] 540
[acase40-02] 541
[acase40-03] 541
[acase40-04] 541
[acccber-01] 541
[acccber-02] 541
[acccdlg-01] 541
[acccdlg-02] 542
[acccdlg-03] 542
[acccdlg-05] 543
[acccinit-01] 542
[acccinit-02] 542
[acccinit-03] 542
[acccinit-04] 542
[acnumchk-01] 543
[acnumchk-02] 543
[acnumchk-03] 543
[acnumchk-04] 543
[acnumchk-05] 544
[acnumchk-06] 544
[acnumchk-07] 544
[acoexhan-01] 544
[acoexhan-02] 544
[acoexhan-03] 545
[acoexhan-04] 545
[acoexman-01] 545
[acoexpar-01] 545
[acoexpar-02] 545
[acoexpar-03] 545
[acoexpar-04] 546
[acoexpar-05] 546
[acoexpar-06] 546
[acoexpar-07] 546
[acoexpar-08] 546
[acoexpar-09] 547
[acoexpar-10] 547
[acoexpar-11] 547
[acoexpar-12] 547
[acoexpar-13] 547
[acoexpar-14] 547
[acoexpar-15] 548
[acoexpar-16] 548
[acoexpar-17] 548
[acoexpar-19] 548
[acoexpar-20] 548
[acoexpar-21] 549
[acoexpar-22] 549
[acoexpar-23] 549
[acoexpar-24] 549
[acoexpar-25] 549
[acoexpar-26] 550
[acoexpar-27] 550
[acoexpar-28] 550
[acoexpar-29] 550
[acoexpar-30] 550
[acompare-01] 551
[acompare-02] 551
[acompare-03] 551
[acompare-04] 551
[acompare-07] 551
[acompare-13] 551
[acompare-14] 552
[acomps-01] 552
[aconfui-01] 552
[aconfui-03] 552
[aconfui-04] 552
[aconfui-05] 552
[aconfui-10] 553
[aconfui-12] 553
[aconfui-13] 553
[aconfui-14] 553
[aconfui-15] 553
[aconfui-16] 553
[aconfui-17] 554
[aconfui-18] 554
[acregchk-01] 554
[acregchk-02] 554
[acregchk-03] 554
[acregchk-04] 555
[acregchk-05] 555
[adbacc-01] 555
[adbadm-01] 555

[adbsess-01] 555
[adbsess-02] 556
[adistrib-01] 556
[adistrib-02] 556
[adistrib-04] 556
[adistrib-05] 556
[adistrib-06] 557
[adistrib-08] 557
[adistrib-09] 557
[adistrib-10] 557
[adistrib-11] 557
[adistrib-13] 557
[adistrib-14] 558
[adistrib-15] 558
[adistrib-16] 558
[adistrib-17] 558
[adistrib-19] 558
[adistrib-20] 558
[adistrib-23] 559
[adistrib-28] 559
[adistrib-29] 559
[adistrib-30] 559
[adistrib-31] 559
[adistrib-34] 559
[adistrib-35] 560
[adistrib-36] 560
[adistrib-37] 560
[adistrib-38] 560
[adistrib-39] 561
[adistrib-40] 561
[adistrib-41] 561
[adistrib-42] 561
[adistrib-43] 561
[adistrib-44] 562
[adocase-01] 562
[adynattr-01] 562
[adynattr-02] 562
[adynattr-03] 562
[adynpred-01] 562
[aeadsply-01] 563
[aeadsply-04] 563
[aeadsply-05] 563
[aeadsply-06] 563
[aeadsply-07] 563
[aeadsply-08] 564
[aeadsply-09] 564
[aeadsply-10] 564
[aeadsply-11] 564
[aeadsply-12] 564
[aeadsply-13] 564
[aeadsply-14] 565
[aeadsply-16] 565
[aeadsply-17] 565
[aeadsply-19] 565
[aeadsply-20] 565
[aeadsply-21] 565
[aeadsply-23] 566
[aeagents-02] 566
[aeagents-05] 566
[aeagents-06] 566
[aeagents-07] 567
[aeagents-08] 567
[aeagents-09] 567
[aeagents-10] 567
[aeagnrl-01] 568
[aeagnrl-02] 568
[aeagnrl-04] 568
[aeagnrl-06] 568
[aeagnrl-08] 568
[aeagnrl-11] 569
[aeagnrl-13] 569
[aeagnrl-14] 569
[aeagnrl-15] 569
[aeagnrl-18] 569
[aearslt-01] 570
[aearslt-02] 570
[aearslt-03] 570
[aeasubag-02] 570
[aeasubag-03] 571
[aeasubag-04] 571
[aeasubag-05] 571
[aeasubag-06] 571
[aeasubag-07] 572
[aeasubag-08] 572
[aevalrel-02] 572
[aevalrel-03] 572
[aevalrel-04] 573
[aevalrel-05] 573
[aevkres-01] 573
[aevkres-02] 573
[aevkres-03] 573
[aevkres-04] 573
[aevkres-05] 574
[aevkres-06] 574
[aexpappl-01] 574
[aexpappl-06] 574
[aexpappl-08] 574
[aexpappl-10] 575
[aexpappl-13] 575
[aexpappl-14] 575
[aexpappl-15] 575
[aexpappl-16] 575
[aexpappl-17] 576
[aexpappl-18] 576
[aexpappl-19] 576
[aexpappl-20] 576
[aexpappl-21] 577
[aexpappl-22] 577
[aexpappl-23] 577
[aexpappl-25] 577
[aexpappl-26] 577
[aexpappl-27] 578
[aexpappl-28] 578
[aexpappl-29] 578
[aexpappl-30] 578
[aexpappl-31] 578
[aexpappl-32] 579
[aexpappl-33] 579
[aexpappl-34] 579
[aexpappl-35] 579
[aexpappl-36] 579
[aexpappl-37] 580
[aexpappl-38] 580
[aexpappl-39] 580
[aexpdlg-01] 580
[aexplore-01] 581
[aexplore-02] 581
[aexplore-03] 581
[aexplore-04] 581
[aexport-01] 581

© BOC Group 2008

[aexport-05] 582
[aexport-06] 582
[aexport-07] 582
[aexport-08] 582
[aexport-10] 582
[aexport-11] 583
[aexport-12] 583
[aexpsort-01] 583
[aexpsort-02] 583
[aexpsort-03] 584
[aexpsort-04] 584
[afile-01] 584
[afile-02] 584
[afile-03] 584
[afile-04] 585
[afile-05] 585
[afile-06] 585
[afile-07] 585
[afile-08] 585
[afile-09] 586
[afile-10] 586
[afile-11] 586
[afile-12] 586
[afile-13] 586
[afile-14] 586
[afile-16] 587
[afile-17] 587
[agdt-01] 587
[agdt-02] 587
[agdt-03] 587
[agdt-04] 587
[agdt-05] 588
[agdt-06] 588
[agdt-07] 588
[agdt-08] 588
[agdt-09] 588
[agdt-10] 589
[agdt-11] 589
[agdt-12] 589
[agdt-13] 589
[agdt-14] 589
[agdt-15] 589
[agdt-16] 589
[agdt-18] 590
[agdt-19] 590
[agdt-20] 590
[agdt-22] 590
[agdt-23] 590
[agdt-24] 591
[agdt-26] 591
[agdt-27] 591
[agdt-28] 591
[agdt-29] 591
[agdt-30] 591
[agdt-31] 592
[agdt-32] 592
[agenpar-01] 592
[agenpar-02] 593
[agenpar-03] 593
[agenpar-04] 593
[agenpar-05] 593
[agenpar-06] 593
[agenpar-07] 593
[agenpar-08] 594
[agenpar-09] 594
[agenpar-10] 594
[agenpar-11] 594
[agenpar-12] 594
[agenpar-13] 595
[agenpar-14] 595
[agenpar-15] 595
[agenpar-16] 595
[agenpar-17] 595
[agenpar-18] 596
[agenpar-19] 596
[aglochan-01] 596
[aglochan-02] 596
[aglochan-04] 596
[aglochan-05] 596
[aglochan-06] 597
[aglochan-08] 597
[aglochan-09] 597
[aglochan-11] 597
[aglochan-12] 597
[aglochan-13] 597
[aglochan-14] 598
[aglochan-15] 598
[agogo-01] 598
[agogo-02] 598
[agogo-03] 598
[agogo-04] 599
[agogo-05] 599
[agogo-06] 599
[agogo-07] 599
[agogo-08] 599
[agogo-10] 600
[agogo-11] 600
[agogo-12] 600
[agogo-13] 600
[agolay-03] 601
[agolay-04] 601
[agolay-05] 601
[agolay-06] 601
[agolay-07] 602
[agolay-08] 602
[agraph-01] 602
[agraph-02] 602
[agraph-03] 602
[agraph-04] 603
[ahaconv-01] 603
[ahaconv-02] 603
[ahaconv-03] 603
[ahaconv-04] 603
[ahaconv-05] 604
[ahaconv-06] 604
[ahaconv-07] 604
[ahaconv-08] 604
[ahagrob-01] 604
[ahagrob-02] 605
[ahagrob-03] 605
[ahamot-01] 605
[ahamot-02] 605
[ahamot-03] 606
[ahamot-04] 606
[ahamot-05] 606
[ahamot-06] 606
[ahamot-08] 607
[ahanote-01] 607
[ahanote-02] 607
[ahanote-04] 607

[ahanote-05] 607
[aicondef-01] 607
[aicondef-02] 608
[aicondef-03] 608
[aicondef-04] 608
[aimport-01] 608
[aimport-02] 608
[aimport-03] 609
[aimport-04] 609
[aimport-05] 609
[aimport-06] 609
[aimport-07] 609
[aimport-08] 609
[aimport-09] 610
[aimport-10] 610
[aimport-11] 610
[aimport-12] 610
[aimport-13] 610
[aimport-14] 611
[aimport-15] 611
[aimport-16] 611
[aimport-17] 611
[aimport-18] 611
[aimport-19] 612
[aimport-20] 612
[aimport-22] 612
[aimport-24] 612
[aimport-25] 613
[aimport-26] 613
[aimport-27] 613
[aleo-01] 613
[aleo-02] 613
[aleo-03] 614
[aleo-04] 614
[aleo-05] 614
[aleo-06] 614
[aleo-08] 614
[aleo-09] 615
[aleo-10] 615
[aleo-13] 615
[aleo-14] 615
[aleo-15] 615
[aleo-16] 616
[aleo-17] 616
[aleo-18] 616
[aleo-19] 616
[aleo-20] 616
[aleo-21] 617
[aleo-22] 617
[aleo-23] 617
[aleo-24] 617
[aleo-25] 617
[aleo-26] 618
[aleo-27] 618
[aleo-28] 618
[aleo-29] 618
[aleo-30] 618
[aleo-31] 619
[aleo-32] 619
[aleo-34] 619
[aleo-35] 619
[aleo-36] 619
[aleo-37] 620
[aleo-38] 620
[aleo-39] 620
[aleo-40] 620
[alib2sgm-01] 620
[alib2sgm-02] 621
[alib2sgm-03] 621
[alibload-01] 621
[alibload-02] 621
[alibload-03] 622
[alogin-05] 622
[alogin-06] 622
[alogin-09] 622
[alogin-10] 623
[alogin-11] 623
[alogin-12] 623
[alogin-13] 623
[alogin-14] 623
[alogin-15] 623
[alogin-16] 624
[alogin-17] 624
[alogin-18] 624
[alogin-19] 624
[alogin-20] 624
[alogin-21] 625
[alogin-22] 625
[amoddiff-02] 625
[amoddiff-08] 625
[amoddiff-09] 626
[amoddiff-10] 626
[amodlru-01] 626
[amodlru-02] 626
[amodlru-03] 626
[amodlru-04] 627
[amodraw-02] 627
[amodraw-03] 627
[amodraw-04] 627
[amodraw-05] 627
[amodraw-08] 628
[amodraw-09] 628
[amodraw-10] 628
[amodraw-11] 628
[amodraw-12] 629
[amodraw-13] 629
[amodraw-14] 629
[amodraw-15] 629
[amodraw-16] 630
[amodraw-17] 630
[amodraw-18] 630
[amodraw-19] 630
[amodraw-20] 631
[amodraw-21] 631
[amodraw-22] 631
[amodraw-23] 631
[amodtab-01] 632
[amodtab-02] 632
[amodtab-03] 632
[amodtab-04] 632
[amomain-01] 632
[amomain-02] 633
[amomain-03] 633
[amomain-05] 633
[amomain-06] 633
[amomain-07] 633
[amomain-08] 634
[amomain-10] 634
[amomain-11] 634
[amomain-12] 634

© BOC Group 2008

[amomain-16] 634
[amomain-17] 635
[amomain-20] 635
[amomain-21] 635
[amomain-22] 635
[amomain-25] 635
[amomain-26] 636
[amomain-27] 636
[amomain-28] 636
[amomain-29] 636
[amomain-30] 636
[amomain-31] 637
[amomain-32] 637
[amomain-33] 637
[amomain-34] 637
[amomain-35] 637
[amomain-36] 638
[amomain-37] 638
[amomain-38] 638
[amomain-39] 638
[amomain-40] 638
[amoshell-01] 639
[amoshell-02] 639
[amoshell-03] 639
[anamegen-01] 639
[anbbrow-01] 639
[anbbrow-02] 640
[anotebk-01] 640
[anotebk-02] 640
[anotebk-03] 640
[aoffanim-01] 640
[aoffanim-02] 640
[aoffanim-03] 641
[aoffanim-04] 641
[aoffanim-05] 641
[aoffanim-06] 641
[aoffanim-07] 641
[aoffanim-08] 642
[aoffanim-09] 642
[aoffanim-10] 642
[aoffanim-11] 642
[aoffanim-12] 642
[aoffanim-13] 643
[aoffanim-14] 643
[aoutgen-01] 643
[aoutgen-02] 643
[aoutgen-03] 643
[aoutgen-04] 643
[apercalc-01] 644
[aperctrl-02] 644
[aperout-01] 644
[aperpar-01] 644
[aperpar-02] 644
[aperpar-03] 645
[aperpar-04] 645
[aperpar-05] 645
[aperpar-06] 645
[aperpar-07] 645
[aperpar-08] 646
[aperpar-09] 646
[aperpar-10] 646
[aperpar-11] 646
[aperpar-12] 646
[aperpar-13] 647
[aperpar-14] 647
[aperpar-15] 647
[aperpar-16] 647
[aperress-03] 647
[aperress-04] 648
[aperress-05] 648
[aperress-06] 648
[apersui-01] 648
[apersui-02] 648
[aplangen-01] 648
[apredct-01] 649
[apredct-02] 649
[aprint-01] 649
[aprint-02] 649
[aproobjs-01] 649
[aproobjs-02] 650
[aproobjs-03] 650
[aproobjs-04] 650
[aproobjs-05] 650
[aproobjs-06] 650
[aproobjs-07] 651
[aproobjs-08] 651
[aproobjs-09] 651
[aproobjs-10] 651
[aproobjs-11] 652
[aproobjs-12] 652
[aproobjs-13] 652
[aproobjs-16] 652
[aproobjs-17] 652
[aproobjs-18] 653
[aproobjs-19] 653
[aproobjs-21] 653
[aproobjs-22] 653
[aproobjs-23] 654
[aproobjs-24] 654
[aproobjs-25] 654
[aproobjs-26] 654
[aproobjs-27] 654
[aproobjs-28] 655
[aproobjs-29] 655
[aproobjs-30] 655
[aproobjs-31] 656
[aproobjs-32] 656
[aproobjs-33] 656
[aproobjs-34] 656
[aproobjs-35] 656
[aproobjs-36] 656
[aproobjs-37] 657
[aproobjs-38] 657
[aproobjs-39] 660
[aproobjs-40] 660
[aproobjs-41] 660
[aproobjs-42] 660
[aproobjs-43] 661
[aproobjs-44] 661
[aproobjs-45] 661
[aproobjs-46] 661
[aproobjs-47] 662
[aproobjs-48] 662
[aproobjs-49] 662
[aproobjs-50] 662
[aproobjs-51] 662
[aproobjs-53] 663
[aproobjs-55] 663
[aproobjs-56] 663
[aproobjs-57] 663

[aproobjs-58] 664
[aproobjs-59] 664
[aproobjs-60] 664
[aproobjs-61] 664
[aproobjs-62] 664
[apwchgui-01] 665
[apwchgui-02] 665
[apwchgui-03] 665
[apwchgui-04] 665
[apwchgui-05] 665
[apwchgui-06] 665
[areena-03] 666
[areena-04] 666
[areena-05] 666
[areena-07] 666
[areena-08] 666
[areena-09] 667
[areena-10] 667
[areena-11] 667
[aregex-01] 667
[aregex-02] 669
[arelrdef-02] 669
[arelrdef-03] 669
[arelrdef-04] 670
[arelrdef-05] 670
[areposui-01] 670
[areposui-02] 671
[areposui-05] 671
[areposui-06] 671
[areposui-08] 671
[areposui-09] 671
[areposui-10] 672
[areposui-11] 672
[areposui-12] 672
[areposui-13] 672
[areposui-14] 673
[aresattr-02] 673
[aresattr-03] 673
[arightmg-01] 673
[arightmg-02] 673
[arightmg-03] 673
[arightmg-05] 674
[arightmg-06] 674
[arightmg-07] 674
[arightmg-08] 674
[arightmg-11] 674
[arightmg-12] 674
[arightmg-13] 675
[arightmg-17] 675
[arightmg-19] 675
[arightmg-20] 675
[arightmg-21] 675
[arightmg-22] 676
[arightmg-23] 676
[arightmg-24] 676
[arightmg-25] 676
[arightmg-27] 676
[arightmg-29] 677
[arightmg-30] 677
[arightmg-31] 677
[arightmg-32] 677
[arightmg-33] 678
[arightmg-34] 678
[arightmg-35] 678
[arightmg-36] 678
[arightmg-37] 678
[arightmg-38] 679
[arightmg-39] 679
[arightmg-40] 679
[arightmg-41] 679
[arightmg-42] 680
[arightmg-44] 680
[arightmg-45] 680
[arightmg-46] 680
[arightmg-47] 680
[arightmg-48] 681
[arightmg-49] 681
[as2rdlg-01] 681
[as2rdlg-02] 681
[as2rdlg-03] 681
[as2rdlg-04] 682
[as2rdlg-05] 682
[as2rdlg-06] 682
[ascript-01] 682
[ascript-02] 682
[ascript-03] 683
[ascript-05] 683
[ascript-06] 683
[ascript-09] 683
[ascript-10] 683
[ascript-11] 683
[ascript-12] 684
[ascript-13] 684
[ascript-14] 684
[ascript-15] 684
[ascript-16] 684
[ascript-17] 684
[ascript-18] 685
[ascript-19] 685
[ascript-20] 685
[ascript-21] 685
[ascript-22] 685
[ascript-23] 686
[ascript-25] 686
[asetref-01] 686
[asetref-02] 686
[asetsel-01] 686
[asetsel-02] 686
[asetsel-04] 687
[asetsel-05] 687
[asetsel-08] 687
[asetsel-10] 687
[asgmlexp-01] 687
[asgmlexp-02] 687
[asgmlexp-05] 688
[asgmlexp-06] 688
[asgmlexp-07] 688
[asgmlexp-08] 688
[asgmlexp-09] 689
[asgmlexp-10] 689
[asgmlexp-12] 689
[asgmlexp-13] 689
[asgmlexp-14] 689
[asgmlexp-15] 690
[asgmlexp-16] 690
[asgmlexp-17] 690
[asgmlexp-18] 690
[asgmlexp-19] 690
[asgmlexp-20] 691
[asik-01] 691

© BOC Group 2008

[asik-02] 691
[asik-03] 691
[asik-04] 691
[asim1dlg-03] 691
[asim1end-01] 692
[asim2dlg-07] 692
[asim2dlg-15] 692
[asim2dlg-16] 692
[asim2dlg-17] 692
[asim2dlg-18] 693
[asim2dlg-21] 693
[asim2dlg-22] 693
[asimmap-01] 693
[asimmap-02] 693
[asimmap-03] 694
[asimmap-04] 694
[asimmap-05] 694
[asimobj-01] 694
[asimobj-02] 694
[asimobj-03] 694
[asimobj-04] 695
[asimobj-05] 695
[asimobj-06] 695
[asimobj-07] 695
[asimtext-01] 695
[asimtext-02] 696
[asrchmod-01] 696
[asrchmod-02] 696
[asrchtlb-01] 696
[astarth-01] 696
[astdcfg-01] 697
[astdcfg-02] 697
[astdcfg-03] 697
[astdcfg-04] 697
[astdcfg-05] 698
[asubsrch-01] 698
[asuibrw-01] 698
[avedbox-01] 698
[averdate-01] 698
[averdate-02] 699
[averdate-04] 699
[averdate-05] 699
[averdate-06] 699
[averdate-07] 699
[averdate-08] 699
[averdate-09] 700
[averdate-10] 700
[averdate-11] 700
[averdate-12] 700
[averdate-13] 701
[averdate-14] 701
[averdate-15] 701
[averdate-16] 701
[averdate-17] 701
[averdate-18] 702
[averdate-19] 702
[averdate-20] 702
[averdate-21] 702
[averdate-22] 703
[averdate-23] 703
[averdate-24] 703
[averdate-25] 703
[averdate-26] 703
[averdate-27] 704
[averdate-28] 704
[averdate-29] 704
[averdate-30] 704
[averdate-31] 705
[averdate-32] 705
[averdate-33] 705
[averdate-34] 705
[averdate-35] 706
[averdate-36] 706
[averdate-37] 706
[averdate-38] 706
[averdate-39] 707
[averdate-40] 707
[averdate-41] 707
[averdate-42] 707
[averdate-43] 707
[averdate-44] 708
[averdate-45] 708
[averdate-46] 708
[averdate-47] 708
[averdate-48] 708
[averdate-49] 708
[averdate-50] 709
[averdate-51] 709
[averdate-52] 709
[averdate-53] 709
[averdate-54] 709
[averdate-55] 710
[awrjpeg-01] 710
[awrjpeg-02] 710
[awrjpeg-03] 710
[awrjpeg-04] 710
[axml-01] 710
[axml-02] 711
[axml-03] 711
[axml-04] 711
[axml-05] 711
[axml-06] 711
[axml-08] 711
[axml-09] 712
[axml-10] 712
[axml-11] 712
[axml-12] 712
[axml-120 713
[axml-13] 712
[axml-15] 712
[axml-16] 713
[axml-17] 713
[axml-18] 713
[axml-19] 713
[axml-21] 713
[axml-22] 713
[axml-23] 714
[axml-24] 714
[axml-25] 714
[axml-26] 714
[axml-27] 714
[axml-28] 714
[axml-29] 715
[hacevalacq-02] 715
[hacevalacq-03] 715
[hacevalacq-04] 715
[hacexp-01] 715
[hacexp-05] 715
[hacexp-06] 716
[hacexp-08] 716

[hacexp-10] 716
[hacexp-14] 716
[hacexp-17] 716
[hacfile-01] 717
[hacimp-05] 717
[hacini-04] 717
[hacini-06] 717
[hacini-14] 717
[hacini-15] 717
[hacini-16] 718
[hacini-17] 718
[hacini-18] 718
[hacini-24] 718
[hacini-26] 718
[hacini-27] 718
[hacini-28] 719
[hacini-29] 719
[hacini-30] 719
[hacini-31] 719
[hacini-32] 719
[hacini-33] 720
[hacinit-01] 720
[svxwins-01] 720
[svxwins-02] 720
[svxwins-03] 720
Evaluation 17, 433
Comparison of Results 433
Evaluation queries 438
FlowMark Audit Trail 441
Evaluation Mechanisms 334
Evaluation queries 438
Model information 440
standardised 439
Example models 30
Explorer 54
Export application models 472
Export application models (ADL) 468
Export attribute profile groups (ADL) 468
Export attribute profiles (ADL) 468
Export model groups 468
Export models (ADL) 468
EXPR 758
expr 758
fixed 758
type 758
val 758
expr (EXPR) 758
EXPRESSION 734, 758
Expression (attribute type) 734
Expressions 758
allcattrs 758
alliattrs 758
allobjs 758
allrattrs 758
amax 758
aql 758
asum 758
attrname 758
atttype 758
aval 758
avalf 758
awsum 758
cfobj 758
cfobjs 758
class 758
conn 758
ctobj 758
ctobjs 758
irtmodels 758
irtobjs 758
isloaded 758
maval 758
mtclasses 758
mtrelns 758
mtype 758
nextsl 758
objofrow 758
paval 758
pavalf 758
pmf 758
prasum 758
prevsl 758
profile 758
rasum 758
rcount 758
row 758
Exprot table 262

F

FDL file 755
File Format of the Result Development 432
File formats 754
ABL 754
ACR 754
ADL 755
APF 755
BMP 755
CSV 755
EMF 755
FDL 755
HTML 756
JPG 756
PCX 756
PNG 756
RTF 756
SVG 756
TXT 757
UDL 757
XML 757
fixed (EXPR) 758
Floating point (attribute type) 735
FlowMark Audit Trail Evaluation 441
Representation in tabular form - (performer-related) 443
Representation in tabular form - (process-related) 442
Focus 139
Font size 199
Free simulation cache 389

G

general context 32
General information about the Modelling Toolkit 31
Generate documentation 486
Generate graphics 156
model area 156
Model pages 158

Generate object names 153
Global attribute change 243
absolute 251
append 253
Attribute selection 249
Changing attribute values 250
Model information (object selection) 248
Model selection 244
Object selection 244
percentage 254
place first 252
Predifined query (object selection) 245
relative 253
replace 252
replace (program call) 252
statistical 254
swap 251
take over 251
User-defined query (object selection) 248
global change 243
absolute 251
append 253
percentage 254
place first 252
relative 253
replace 252
replace (program call) 252
statistical 254
swap 251
take over 251
Global End 340
Glossary 815

H

Hide all submodels 200
Hierarchical working environment models 224
HOMER 108
Add classes 773
Adding attribute profile classes 774
Adding attributes 777
Adding model attribute classes 785
Adding model attributes 787
Adding records 776
Attribute formats in HOMER 781
Attribute types in HOMER 780
Changing model attributes 788
Consolidating acquisition tables 121
Creating acquisition tables 118
defining structure of acquisition table 768
Delete Attribute profile classes 775
Delete records 777
deleting attributes 785
Deleting classes 774
Deleting model attribute classes 787
Deleting model attributes 788
Editing attribute profile classes configuration 775
editing class configuration 774
Editing model attribute class configuration 786
Editing records 776
Export ADL files 115
General model parameter 769
modifying attributes 783
Object settings 771
Select scenario 109
start 109
survey 108
To administrate HOMER 763
transform ADL data to Excel sheets 110
HOMER scenarios 763
convert 766
create 764
Delete 768
edit 768
export 767
import 764
import (HOMER 3.7) 765
HOMER scenarios (overview) 763
HTML files 756
Hyperlink display (referenced model) 255

I

Import application models 464
Ignore attribute profiles from ADL file 463
Ignore Models 461
Import Application Models (ADL)> 445
import attribute profile (ADL) 445
Examples 454
Options 454
import attribute profile groups (ADL) 445
Examples 454
Options 454
Import attribute profiles (ADL) 446
Import Model Groups (ADL) 445
Examples 454
Options 454
import models (ADL) 445
Examples 454
Options 454
Import models (ADL) 446
Assign Model Type 453
Assign version number 464
General options for ADL import 454
Ignore attribute profiles from ADL file 463
Ignore Import Models 461
Overwrite existing attribute profiles 463
Overwrite existing models 460
Paste into existing attribute profiles 462
Rename import models 455
Rename attribute profiles from ADL file 462
To increase version number 461
To paste into existing models 455
To select attribute profiles 450
To select models 450
Import/Export 17, 445
Incoming references 267
to current object 268
Information acquisition 108
HOMER (survey) 108
Information Acquisition 17
Input dialogue for expressions 52
Input dialogue for text 51
Input support in ADOxx Notebooks 201
Allocating resources 227
Assigning performers 222
colours 203
Date value 205

Distribution (overview) 212
Edit date and time 206
program call 205
random generator 211
records 202
select enumeration value 204
Select values from an enumeration list 204
Times 206
Input window 51
for expressions 52
for texts 51
Print diagram 51
Print text field 51
Inspektor 64
Integer 735
Integer (attribute type) 735
integer (EXPR type) 758
inter-model references 94, 265
Add 207
Edit depth 98
effects of the versioning on inter-model references 98
incoming 267
outgoing 265
Settings 97
Interref 736
irtmodels (expression) 758
irtobjs (expression) 758
is inside (relation) 135
isloaded (expression) 758
Item search 43
Display search results 44

J

JPG files 756

L

Language codes 738
Language tables 738
Layout parameters (page layout) 194
Library Management 15
Load mode (Attribute and class filter) 495
Long string 735
Long string (attribute type) 735

M

Macro name (transition condition) 219
Macros (transition conditions) 217
Managing connectors in ADOxx Notebooks 146
Manual random generator 342
maval (expression) 758
Messages 745
Display 749
Forward 748
New message 749
Read 751
Read instant message 752
Receive instant message 752
Received message 749
Reply 747
Select receiver 746
send 745
Settings 752
Model 255, 256
activate model window 255
Activate referenced model 255
Assign models to a model group 271
Check cardinalities 264
Close 259
Copy model reference to model group 273
Create 256
Create model group 270
Delete 283
Delete model group 271
Delete model reference 273
inter-model references 265
Model attributes 264
Model comparison 273
Model name 264
Model type filter 42
Model types view 42
Move model group 271
Move model reference to model group 272
Navigating through open models function 255
Open 257
Print models 288
Print several models with identical settings 290
printing (model selection) 289
Rename 262
Rename model group 270
Save 259
Save as new version 262
Save with a different name 260
search 275
Version number 263
Model attributes 264
Model comparison 273
Model selection 273
Results 274
Show differences in values 275
Model documentation 486
Attribute and class filter 493
Generate 486
Options 488
Model editor 127
Context menu 173
Model window 130
Modelling Bar 128
Working area 127
Model Export (ADL) 468
Model Export (XML) 481
Model Management 15
Model search by model attributes 276
results 277
Model type filter 42
Model types view 42
Model type filter 42
Model window 130
activate 255
View all open model windows 255
Modelling 17, 126
Editing attribute profiles 291
Model editor 127
Model Management 256

Model navigation 255
Modelling Bar 128
Class panel 129
Relation panel 129
Modelling Component 126
Modelling conditions for Simulation 240
Allocating resources 227
Assigning performers 222
Assigning performers to sub processes 225
Completeness 240
Date value 205
Edit date and time 206
Hierarchical working environment models 224
Parallelism 241
Performer calendar 228
Probabilities 226
Process calendar 235
random generator 211
Times 206
Transition conditions 215
Transition probabilities 220
Modelling modes 131
Drawing mode 131
Editing mode 131
Modelling Toolkit 107
close 727
file formats 754
start 722
start (Single-Sign-on) 724
Model-related versioning 92
Mouse access 188
Display 189
Move connector centre 148
Move model group 271
Move model reference to model group 272
Moving objects 139
mtclasses (expression) 758
mtrelns (expression) 758
mtype (expression) 758

N

Names generation 153
Navigating through open models 255
Navigator 62
Next model window 255
nextsl (expression) 758
Non-cooperative Activities 340
Number objects 155

O

Object arrangement (automatic) 160
objofrow (expression) 758
Offline Animation - Input 386
Offline Animation - Operation 387
Offline-Animation 386
Open existing models 257
Options (documentation) 488
Outgoing references 265

P

Page layout 192
ADOxx standard page layout 195
ADOxx standard page layout for tables 196
Layout parameters 194
Poster printout (with tabs) 196
Parallelisms 339
Paste objects/connectors (ADOxx clipboard) 151
Referenced objects 151
Path Analysis 343
(Input parameters) 353
Application Scenario (Example) 354
Background Details 353
Carry out/Execute 343
How to treat cooperative activities 354
Input 353
Processing 353
Results 347
Path Analysis - Numbers 345
paval (expression) 758
pavalf (expression) 758
PCX files 756
Performer assignment with probabilities 226
Performer calendar 228
adding day profiles 230
Assigning day profiles 232
Calculating time of presence 234
Changing day profiles 231
Copying day profiles 233
Deleting day profiles 232
display 229
Displaying day profiles 230
searching for day profiles 234
Show specified days 233
Period-related View 342
Personal management 513
Personnel Capacity Planning 361
Pie chart 89
pmf (expression) 758
PNG-file 756
prasum (expression) 758
Predefined plans 318
Model selection 319
Results 321
Predefined queries 318
Execute 319
Model information 311
Model selection 319
Results 321
Previous model window 255
prevsl (expression) 758
Print diagram 51
Print models 288
Select pages 290
Print several models with identical settings 290
Print text field 51
Process calendar 235
Adding day profiles 236
Adding intervals 237
assigning day profiles 238
Calculating number of processes 239
Changing day profile 236
Changing intervals of occurrence 238

copying day profiles 239
Day profiles 236
deleting day profile 238
deleting intervals 238
Display 235
displaying specified days 239
Searching for day profiles 239
Process cost analysis 497
actual process cost analysis 499
in ADOxx 498
Introduction 497
Plan process cost analysis 499
Process hierarchy (graphical) 281
Process hierarchy (tabular) 279
All child processes 281
Result 281
Process Times 339
Process-related View 342
profile (expression) 758
Program call 736
Program call (attribute type) 736

Q

queries 306
Model information 311
standardised 308
User-defined 312
Query results 313

R

random generator 211
Distribution (overview) 212
Enter distributions 213
rasum (expression) 758
rcount (expression) 758
Read instant message 752
Real Time 335
Re-allocate connectors 145
Receive instant message 752
Received message 749
RECORD 736
Record (attribute type) 736
Record attributes 104
Redo 137
Reference (attribute type) 736
Reference settings 97
references 265
Add 207
incoming 267
outgoing 265
References 94
Edit depth 98
effects of the versioning on inter-model references 98
Settings 97
Relation panel 129
Relation tables 322
Rename a model 262
Change a model name 264
Change version number 263
Rename model group 270
Represent result development as function graph 431
Results of predefined plans 321
Change titles of columns 322
Results of predefined queries 321
Results of the evaluation queries 441
row (expression) 758
RTF files 756
Run Offline Animation 386

S

Save model as new version 262
Save model under a different name 260
save models 259
Save models automatically 260
Save table contents 262
Search model 152, 275
by model attributes 276
Graphical process hierarchy 281
Tabular process hierarchy 279
Search result window 65
Search results 65
Select enumeration value 204
Select objects/connectors 138
Select values from an enumeration list 204
service information 730
Short introduction to modelling 25
Show attribute profile values 296
Show several attribute profile values simultaneously 297
Show submodels 200
Simulation 17, 332
Activity Times 335
algorithms (overview) 333
Capacity Analysis 354
Company Time 335
Cooperative Activities 340
Employ Agents 422
Evaluation Mechanisms 334
Global End 340
introduction 332
Manual random generator 342
Non-cooperative Activities 340
Path Analysis 343
Period-related view 342
Process Times 339
Process-related view 342
Real Time 335
Time Concepts/Terms of Time 334
Workload Analysis 367
simulation algorithms (overview) 333
Simulation Cache/Buffer 388
Effects on the Simulations Buffer/Cache 389
Snap grid 190
Settings 191
Start ADOxx 722
Start ADOxx (Single-Sign-on) 724
Start ADOxx WebService 726
status information 728
String 736
String (attribute type) 736
string (EXPR type) 758
structure of acquisition tables 768
Add classes 773
Adding attribute profile classes 774
Adding attributes 777

Adding model attribute classes 785
Adding model attributes 787
Adding records 776
Attribute formats in HOMER 781
Attribute types in HOMER 780
Changing model attributes 788
Delete Attribute profile classes 775
Delete records 777
deleting attributes 785
Deleting classes 774
Deleting model attribute classes 787
Deleting model attributes 788
Editing attribute profile classes configuration 775
editing class configuration 774
Editing model attribute class configuration 786
Editing records 776
General model parameter 769
modifying attributes 783
Object settings 771
SVG-file 756
Swimlanes 91
Symbol name (discrete distribution) 212
Symbol name (transition condition) 219

T

Tabular model view 181
Class selection 182
create objects 186
Display attribute columns 183
Edit attributes 184
enumeration list 204
enumeration value 204
program call 205
Select 185
Sort 185
Undo 185
Terms 32
Text input field 52
Time 736
Time (attribute type) 736
time (EXPR type) 758
Time-related versioning 92
To export attribute profile groups(XML) 481
To export attribute profiles (XML) 481
To export attribute profiles(XML) 481
To export model (XML) 481
To export model groups (XML) 481
To import attribute profile groups (XML) 473
To import attribute profiles (XML) 474
To import attributes profile(XML) 473
To import model groups (XML) 473
To import models (XML) 474
To select attribute profiles 476
To select models 476
To import models(XML) 473
Tolerance waiting time 381
Trademarks 13
Transition condition (Subsequent relation) 215
Macros 217
Transition conditions 215
Define 216
Macros 217
Syntax 219
Transition probability 220
TXT files 757
type (EXPR) 758
double 758
integer 758
string 758
time 758

U

UDL files 757
Undo 137
Uninterruptible activities 383
User Management 15

V

val (EXPR) 758
Variable name (transition condition) 219
Variables 212
Versioning 92
Basics 92
Effects 93
Model-related 92
Time-related 92
View 186
Connector marks 197
Font size 199
Hide all submodels 200
Mode 186
Mouse access 188
Page layout 192
Redraw 201
Show submodels 200
Snap grid 190
Zoom 189
View mode 186
Display options 187
View window 51
for texts 51
Print diagram 51
Print text field 51
Visible area 190

W

Workload Analysis 367
Application Scenario (Example) 385
Background details 376
carry out 367
Input 376
Processing Mode 376
Results 373
Task Stacks 372
Tolerance waiting time 381
Uninterruptible activities 383

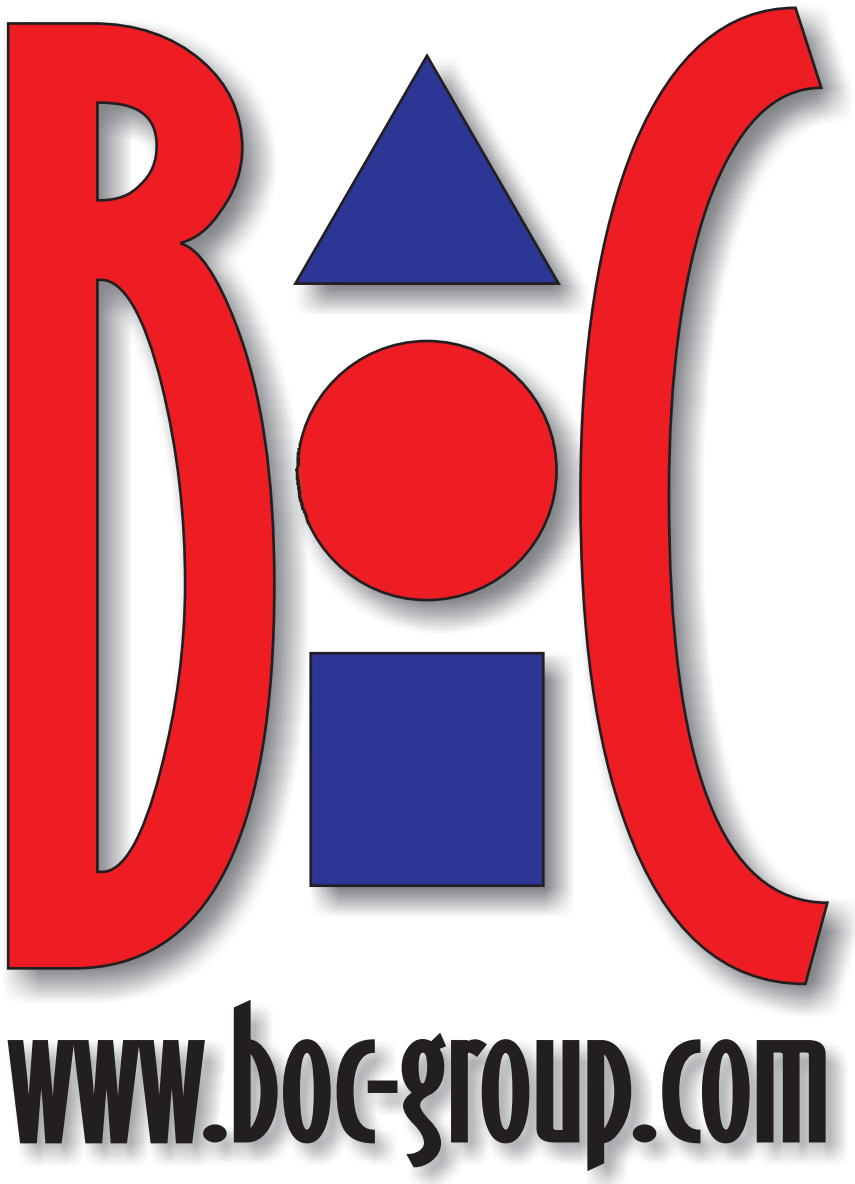
X

XML Export 481
Attribute profile groups 481
Attribute profiles 481
Model groups 481

Models 481
XML Import 473
Attribute profile groups 474
Attribute profiles 474
Model groups 474
Models 474
XML-file 757

Z

Zoom 189



BOC Information Technologies Consulting AG
Wipplingerstr. 1
A-1010 Vienna

BOC Asset Management GmbH
Baeckerstr. 5
A-1010 Vienna