Autodesk® Topobase™ Administrator Guide

Autodesk®

© **2009 Autodesk, Inc. All Rights Reserved.** Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

Trademarks
The following are registered trademarks or trademarks of Autodesk, Inc., in the USA and other countries: 3DEC (design/logo), 3December, 3December.com, 3ds Max, ADI, Alias, Alias (swirl design/logo), AliasStudio, Alias|Wavefront (design/logo), ATC, AUGI, AutoCAD, AutoCAD Learning Assistance, AutoCAD LT, AutoCAD Simulator, AutoCAD SQL Extension, AutoCAD SQL Interface, Autodesk, Autodesk Envision, Autodesk Insight, Autodesk Intent, Autodesk Inventor, Autodesk Map, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSnap, AutoSketch, AutoTrack, Backdraft, Built with ObjectARX (logo), Burn, Buzzsaw, CAiCE, Can You Imagine, Character Studio, Cinestream, Civil 3D, Cleaner, Cleaner Central, ClearScale, Colour Warper, Combustion, Communication Specification, Constructware, Content Explorer, Create>what's>Next> (design/logo), Dancing Baby (image), DesignCenter, Design Doctor, Designer's Toolkit, DesignKids, DesignProf, DesignServer, DesignStudio, Design|Studio (design/logo), Design Web Format, Discreet, DWF, DWG, DWG (logo), DWG Extreme, DWG TrueConvert, DWG TrueView, DXF, Ecotect, Exposure, Extending the Design Team, Face Robot, FBX, Filmbox, Fire, Flame, Flint, FMDesktop, Freewheel, Frost, GDX Driver, Gmax, Green Building Studio, Heads-up Design, Heidi, HumanIK, IDEA Server, i-drop, ImageModeler, iMOUT, Incinerator, Inferno, Inventor, Inventor LT, Kaydara, Kaydara (design/logo), Kynapse, Kynogon, LandXplorer, LocationLogic, Lustre, Matchmover, Maya, Mechanical Desktop, Moonbox, MotionBuilder, Movimento, Mudbox, NavisWorks, ObjectARX, ObjectDBX, Open Reality, Opticore, Opticore Opus, PolarSnap, PortfolioWall, Powered with Autodesk Technology, Productstream, ProjectPoint, ProMaterials, RasterDWG, Reactor, RealDWG, Real-time Roto, REALVIZ, Recognize, Render Queue, Retimer,Reveal, Revit, Showcase, ShowMotion, SketchBook, Smoke, Softimage, Softimage|XSI (design/logo), SteeringWheels, Stitcher, Stone, StudioTools, Topobase, Toxik, TrustedDWG, ViewCube, Visual, Visual Construction, Visual Drainage, Visual Landscape, Visual Survey, Visual Toolbox, Visual LISP, Voice Reality, Volo, Vtour, Wire, Wiretap, WiretapCentral, XSI, and XSI (design/logo).

The following are registered trademarks or trademarks of Autodesk Canada Co. in the USA and/or Canada and other countries: Backburner,Multi-Master Editing, River, and Sparks.

The following are registered trademarks or trademarks of MoldflowCorp. in the USA and/or other countries: Moldflow, MPA, MPA (design/logo),Moldflow Plastics Advisers, MPI, MPI (design/logo), Moldflow Plastics Insight,MPX, MPX (design/logo), Moldflow Plastics Xpert.

All other brand names, product names or trademarks belong to their respective holders.

Disclaimer
THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. "AS IS." AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

Published by:
Autodesk, Inc.
111 Mclnnis Parkway
San Rafael, CA 94903, USA

Contents

Chapter 1 **Topobase Administrator Guide** 1

Administration Introduction 1
Starting Topobase 1
Starting Topobase Administrator 2
Password 4
Starting Topobase Client 5
Administrator Explorer 6
Document Settings 7
Administrator Workspace Menu 8
Administrator Document Menu 9
Administrator Setup Menu 12
Setting Up Users and User Groups 13
Setting up the Client Commands 14
Customizing the Client Menu Bar 15
Setting Application Options 16
Setting Common Application Options 16
Setting Map Options 17
Setting COGO Application Options 19
Setting Jobs Application Options 21
Setting Document Explorer Options 22
Setting Generate Graphic Application Options 22
Setting Other Application Options 24
Setting Document Options 24
Common Document Options 25

Setting Document Map Options . . . 25
COGO and Dimensioning Document Options . . . 26
Setting Job Topology Document Options . . . 27
Setting Wastewater Document Options . . . 28
Setting Electric Document Options . . . 28
Document Administration . . . 29
Setting Up General Shortcut Menus . . . 30
Setting Up Specific Shortcut Menus . . . 32
Configuring Topobase Commands . . . 32
Customizing Form Toolbars and Menus . . . 33
Exporting Document User Interface Settings . . . 34
Document: FlyIns . . . 34
Topobase Data Modelling . . . 35
Data Model Administrator . . . 36
Data Model Templates . . . 36
Data Model Code . . . 37
Topobase Data Checker . . . 37
Creating a Data Check . . . 39
Scheduling a Data Check . . . 41
Topobase Point Numbering . . . 42
Point Numbering Per Document . . . 45
Point Numbering Per Group . . . 46
Point Numbering Per Grid 100m . . . 47
Point Numbering Per Raster Plan . . . 49
Understanding Users and Workspaces . . . 50
User Logins . . . 51
Topobase Administration: User Groups . . . 52
Creating Workspaces . . . 53
Adding and Removing Documents . . . 53
Security Administrator . . . 57
Managing Roles . . . 59
Add a Role . . . 60
Permission Templates . . . 60
Edit a Role . . . 62
Setting Up Topobase Web . . . 63
Assigning Display Models for Topobase Web . . . 65
Configuring the Document Explorer . . . 67
Setting Up Electric Explorer Profiles . . . 70
Topologies . . . 71
Job Management . . . 71
Topobase System Tables . . . 71
System Tables: Data Model Code . . . 73
System Table TB_DICTIONARY . . . 74
System Table TB_GN_DIALOG_HIGHLIGHT . . . 75
System Table TB_GN_DIRECTORY_ALIAS . . . 75
System Table TB_GN_INFO_REDIRECT . . . 77

System Table TB_INFO 78
System Table TB_INTERSECTION 79
System Tables TB_JOB_* 81
System Table TB_JOB_VERSION 81
System Table TB_JOB_STATE 84
System Table TB_JOB_STATE_TRANSITION 85
System Table TB_JOB 86
System Table TB_JOB_PERIMETER 87
System Table TB_LABEL 88
System Table TB_LABEL_DEF 88
System Table TB_NET_* 89
System Table TB_NET_TRACE_TPL 89
System Table TB_NET_SCOND_DEF 91
System Table TB_POSITION_FINDER 92
System Table TB_RELATIONS 94
TB_RELATIONS: Parent, Child, Sibling 95
TB_RELATIONS: Master-Detail Relations 96
System Tables TB_RULE* 98
System Table TB_RULE_PARAM 99
System Table TB_RULE_DEF 100
System Table TB_TOPOLOGY* 101
System Table TB_WORKFLOW 102
Topobase Advanced Topics 103
Explorer Configurator 103
Creating SQL Expressions 103
Using the SQL Assistant 105
Using the SQL Assistant in the Form Designer 108
Select Statement Validation 109
Workflow Definition 110
Feature Rule Definition 110
Report Designer 111
Display Models 111
Topobase Administration Tutorials 111
Creating a Topobase Document 111
Creating a Label Definition 113
Creating a Display Model 116
Creating a Workspace 116
Exploring a Data Model 118
Creating a Report 121
Creating a Report Head 124
Report Definition: Adding a Chart 124
Creating a Master-Detail Report 126
Creating and Using Dimensions 127
Creating A Profile Definition 128
Adding the Profile Extension 129
Using the SQL Sheet 130

SQL Sheet: Standard SQL Commands 133
SQL Sheet: SQL Plus like Commands 133
SQL Sheet: Special Commands 134
Creating a Plot Using AutoCAD Blocks 136

Chapter 2 Data Model 143

Data Model Administrator Introduction 143
Starting the Data Model Administrator 143
Data Model Administrator: Creating a Database 144
About Data Models 148
Data Model Description 149
About Coordinate Systems 150
Updating the Coordinate System 151
Data Model Administrator: Shortcut Menus 152
Document Shortcut Menu 153
Topics Shortcut Menu 156
Feature Class Shortcut Menus 157
Data Model Administrator: Creating a Feature Class 160
Editing Feature Classes 162
Setting a Feature Class Read Only 163
Creating a Feature Class From a View 164
Customizing the Feature Explorer 165
Adding Attributes 167
Attribute Type, Precision and Scale 170
Attribute Relation Settings 170
Modifying Attribute Properties 171
Deleting Attributes 172
Data Model: Feature Class Types 172
Feature Class Type: Attribute 173
Feature Class Type: Centroid 174
Feature Class Type: Collection 174
Feature Class Type: Compound Polygon 175
Feature Class Type: Compound Line String 176
Feature Class Type: Dimension 177
Feature Class Type: Label 180
Feature Class Type: Line String 181
Feature Class Type: Point 181
Feature Class Type: Polygon 182
Data Model: Domain Tables 183
Editing a Domain Table 184
Data Model: Model Tables 184
Updating Model Tables 186
Data Model: Labels 187
Label Properties 189
Label Properties: Label Information 190
Label Properties: Regeneration Definition 191

Label Properties: Positioning 192
Label Properties: Orientation 193
Label Properties: Positioning and Orientation 195
Label Properties: Alignment 197
Label Properties: Scaling 198
Label Properties: Label ID 199
Label Properties: ORIGIN conversion 199
Label Select Statements 200
Label Formatting With SQL 202
Label Geometry Select Statements 202
Sample Labels 202
Working with Complex Labels 203
Label Commands 204
Data Model: Utilities 204
Data Model: Dimensioning 205
About Aligned Dimensioning 206
About Orthogonal Dimensioning 208
Data Model: Compounds 211
Compound Triggers 212
Compound Commands 213
Data Model: Topologies 215
Data Model: Jobs 215
Data Model: Feature Templates 216
Data Model: Schematics 216
Getting Schematics Started 218
Creating a Schema Plan Template 218
Schema Plan Settings 220
Editing a Schema Plan Template 222
Creating a Generalization 223
Editing a Generalization 224
Data Model: Intersections 224
Creating an Intersection 226
Intersections: Adjustment 228
Intersections: Getting Started 229
Intersections: Results 231
Intersections: Setting up 2D / 3D Intersections 232
Data Model: Spatial Relationship Settings 233
Data Model: Oracle Schema Converter 235
Converting Oracle Tables 238
Topobase Data Transfer (Oracle Data) 239
Creating a Mapping Definition 240
Editing a Mapping Definition 241
Using Oracle Data Import 242
Configuring Oracle Data Import 243
Oracle Data Import: Mapping Feature Classes and Domains 246

Oracle Data Import: Edit Feature Class Properties 248
Oracle Data Import: Edit Feature Attribute Properties . . . 250
Oracle Data Import: Domain Settings 251
Oracle Data Import: SQL Statements and Placeholder 252
Mapping Definition for Interlis Data Import 252
Importing an Oracle Data Import Mapping Definition 253
Exporting an Oracle Data Import Mapping Definition 254

Chapter 3 Topology Reference 255

Topology Introduction 255
Getting Topology Started 256
Creating Topologies 256
Area Topology Properties 258
Arc Linearize Tolerance 259
Logical Topology Properties 260
Flow Direction Management 261
Connection Tolerance Radius 261
Utility Topology Properties 262
Area Topology Introduction 264
Area Topology Tables 266
Topology: _TSUR State Codes (Area) 267
Area Topology Initialization 268
Area Topology Checker 269
Logical Topology Introduction 269
Topology Checker 270
Network Tracer 270
Logical Topology: Tracing Templates 272
Logical Topology: Tracing Types 274
Defining Tracing Conditions 275
Tracing Condition SQL Statements 276
Logical Topology Initialization 277
Network Topology Update 278
Updating Tracing Conditions 279
Topology Offline Initialization Processing 281
Topology Online Processing 281
Topology Tutorial 281
Defining a Tracing Template for a Logical Topology 282
Defining a Tracing Condition 282

Chapter 4 Utility Reference 285

Utility Model Introduction 285
Utility Data Model 287
Creating a Utility Model 288
Utility Feature Rules 290

Utility Model Tutorial . . . 291
Creating a Utility Data Model . . . 291
Defining a Label for a Utility Object . . . 292

Chapter 5 Jobs Reference . . . 295

Jobs Introduction . . . 295
Getting Jobs Started . . . 296
Jobs Update . . . 297
Job Administrator . . . 298
Configuring Jobs . . . 299
Job Enabling (Document Settings) . . . 299
Creating a Job Enabled Document . . . 300
Job Enabling a Document . . . 300
Job Disabling a Document . . . 301
Jobs Feature Locking . . . 301
Assigning a Job Perimeter . . . 302
Creating a Job Template (Job Administrator) . . . 304
Job Leasing . . . 305
About Jobs And Job Templates . . . 306
About Job States . . . 308

Chapter 6 Form Designer Reference . . . 309

Form Designer Introduction . . . 309
Starting Form Designer . . . 309
Form Designer: Managing Multiple Forms . . . 310
Form Designer: Creating Generic Forms . . . 311
Form Designer: Menu Bar . . . 313
Designer Menu: Options . . . 316
Designer Menu: Master-Detail . . . 317
Designer Menu: Tab Pages . . . 318
Form Designer: Controls . . . 318
Form Designer: Default Forms . . . 319
Form Designer: Toolbar . . . 320
Form Designer: General Control Properties . . . 324
Form Designer: Advanced Combobox - Properties . . . 326
Form Designer: Advanced Form Launcher - Properties . . . 327
Form Designer: Basic ComboBox - Properties . . . 328
Form Designer: Basic Form Launcher - Properties . . . 332
Form Designer: CheckBox - Properties . . . 334
Form Designer: Color ComboBox - Properties . . . 335
Form Designer: Coordinates Editor - Properties . . . 337
Form Designer: Document Linker - Properties . . . 338
Form Designer: Editable SQL TextBox - Properties . . . 339
Form Designer: Feature Highlighter - Properties . . . 341
Form Designer: FID Form Launcher - Properties . . . 342

Form Designer: FID Linker - Properties . . . 343
Form Designer: Formula TextBox - Properties . . . 344
Form Designer: Horizontal Line - Properties . . . 345
Form Designer: Picture Combobox - Properties . . . 346
Form Designer: Picture Linker - Properties . . . 347
Form Designer: Property TextBox - Properties . . . 349
Form Designer: Read-Only SQL TextBox - Properties . . . 351
Form Designer: Script Programmable - Properties . . . 353
Form Designer: SQL Form Launcher - Properties . . . 353
Form Designer: Static Picturebox - Properties . . . 355
Form Designer: Static Text - Properties . . . 356
Form Designer: Model Selector - Properties . . . 356
Form Designer: Examples . . . 357
Form Designer Example: Auto Tab Order . . . 357
Form Designer Example: Validation Code . . . 358
Form Designer Example: Formula TextBox . . . 359
Formula TextBox: Sample Formulas . . . 360
Form Designer Example: Read-Only SQL TextBox . . . 362
Form Designer Example: Editable SQL TextBox . . . 364
Form Designer Example: Advanced Combobox . . . 365
Form Designer Example: Coordinates Editor with SQL . . . 369
Form Designer Tutorial . . . 370
Creating Controls . . . 370
Showing the Coordinates of Utility Features . . . 372
Defining Tab Pages for Forms . . . 374
Defining Form Toolbars . . . 374
Defining Shortcut Menus for Feature Class Forms . . . 376
Assigning a Feature Class Form to a User Group . . . 376
Assigning Help Files to Feature Class Forms . . . 377

Chapter 7 Working with Display Models . . . 379

Understanding Display Models . . . 379
Storing Display Models . . . 381
Modifying Display Models . . . 382
Creating Display Models . . . 383
Using Display Model Samples . . . 387
Overview of the Display Manager . . . 387
Using the Display Manager with Autodesk Topobase . . . 388
Bringing In Features From Topobase . . . 388
Creating LAYER Files . . . 390
Styling Features . . . 391
Topobase Line Styles . . . 393
Styling Label Features . . . 393
Displaying Label Features with Orientation . . . 397
Styling Annotation Layers . . . 398
Displaying Label Features with Prefix and Suffix . . . 398

Map Drawing Units . . . 399
Using Enhanced Stylization . . . 400
Enabling Enhanced Stylization . . . 400
Enhanced Point Stylization . . . 400
Enhanced Line Stylization . . . 401
Enhanced Polygon Stylization . . . 403
Feature Text Labels . . . 404
Feature Symbol Labels . . . 405
Building If, Lookup, and Range Expressions . . . 406
If Expression . . . 407
Lookup Expression . . . 407
Range Expression . . . 407
Creating Topobase Calculations . . . 408
Topobase Text Function TB_SQL . . . 409
Topobase Text Function TB_VARIABLE . . . 410
Styling Job-Enabled Features . . . 412

Chapter 8 Plot Reference . . . 417
Understanding Plot . . . 417
Getting Plot Started . . . 417
Configuring Plots . . . 417
Plot Settings . . . 418
Plot Display Models . . . 419
Plot Extension . . . 421
Plot Map Placeholder . . . 422

Chapter 9 Report Designer Reference . . . 425
Report Designer Introduction . . . 425
Report Designer: Layout Components . . . 425
Getting Report Designer Started . . . 427
Report Designer Menu Bar . . . 428
Report Designer: Adding Controls . . . 429
Report Designer Options . . . 433
Report Designer Wizard . . . 434
Report Designer Control Properties . . . 435
Report Designer: Page Controls . . . 437
Report Designer: Text Control . . . 439
Text Control: Format . . . 441
Report Designer: DB Column Control . . . 441
Report Designer: Picture Control . . . 442
Report Designer: Line Control . . . 443
Report Designer: Chart Control . . . 444
Using the Chart Wizard . . . 445
Report Designer: SQL Statements . . . 447
Report Designer: Functions . . . 448

Report Designer: Functions - Example 449
Report Designer: SQL Values 450
Report Designer: Unit Placeholder 451
Report Designer: Parameter 453
Report Designer Import Report Definition 454
Report Designer ASCII Export 454
Report Designer: Advanced Topics 455
Report Designer: Polygon and Line Definition Report 455
Report Designer: Spatial Selection Report 456
Report Designer: Coordinate Report 458
Report Designer: Suppress a Zone 458
Report Designer: Create a Crystal Report 459
Report Designer: Call a Report via API 460
Report Designer: Change Colors at Runtime 460
Report Designer: Reading Data from the Memory 461

Chapter 10 Position Finder Reference 463
Position Finder Introduction 463
Position Finder Administrator 464
Position Finder Sequential Search 465
Sequential Search: Input Types 466
Sequential Search: Placeholders 468
Defining a Sequential Search 468
Sequential Search: Executing and Validating 470
Sequential Search: Example 470
Position Finder Flat Search 472
Defining a Flat Search 472
Flat Search: Example 473
Position Finder PlugIn Search 474
Position Finder Tutorial 475
Defining a Search to Find a Building 475
Defining a Search to Find an Object 477
Defining a Flat Search (Example) 478

Chapter 11 Workflow Reference 481
Workflow Introduction 481
Workflow Definition 481
Workflow Explorer Configuration 483
Workflow Execution 484

Chapter 12 Profiles 485
Overview of Profiles 485
Profile Designer 486
Create a Profile Definition 487

General Settings . . . 488
Features Settings . . . 489
Decoration Settings . . . 492
Labels Settings . . . 494
Working With Profile Templates . . . 495
Profile Data Model . . . 496
Profile System Tables . . . 497
System Table TB_PROFILE . . . 498
System Table TB_PROFILE_AXIS . . . 500
System Table TB_PROFILE_DEFINITION . . . 501
System Table TB_PROFILE_DETAIL . . . 502
System Table TB_PROFILE_SUB_DETAIL . . . 505
Profile Feature Classes . . . 506
Profile Feature Class Attributes . . . 508
Profile Manager . . . 509

Chapter 13 **Feature Rules . . . 511**
About Feature Rules . . . 511
Feature Rule Properties . . . 512
Edit Feature Rule Properties . . . 513
Feature Rule Definition . . . 514
Feature Rule Group . . . 516
Feature Rule Parameters . . . 517
Assigning Feature Rules . . . 517
Exploring Feature Rules . . . 517
Exploring Feature Rule Groups . . . 518

Glossary . . . 521

Index . . . 527

Topobase Administrator Guide

1

Administration Introduction

This guide for Topobase™ administrators includes the following subjects:

- Data model administration (Creating/Editing Data Models)
- Form Designer (Customizing feature class forms)
- Application and document options
- User administration: User group (page 526) and user interface
- Display models for Topobase
- Job management
- Topology
- Report designer
- Using the SQL Assistant
- Workflow management

Starting Topobase

You can start Topobase Client and Topobase Administrator separately.

After the login you must open a workspace.

Topobase User And Topobase System User

When you log in, you must enter your personal user name. Depending on which Topobase user group you belong to, you have user group specific access rights to the database. See also Topobase Administration: User Groups (page 52).

Also, you must be connected to a Topobase system user. Mostly a default connection is specified during the configuration process. You can either be connected to the Topobase System User (default is TBSYS) or to the Topobase Main User, which has restricted rights (default is TBMAIN).

To view the connected Topobase system user

1. Start either Topobase Administrator or Topobase Client.
2. In the Topobase Login dialog box, click Options >>> to display the Topobase System/Main User section.

To work with Topobase Client, we recommend that you connect to the Topobase Main User TBMAIN. When you are connected to TBMAIN, administrator functionality is not available.

Topobase Administrator—You must be connected to the Topobase System User TBSYS and you must be member of the ADMIN user group if you want to start Topobase Administrator. A member of the ADMIN user group who is connected to the Topobase Main User TBMAIN cannot start Topobase Administrator.

Topobase Client—You must be connected to the Topobase System User and you must be a member of the ADMIN user group to use the administrator functionality of Topobase Client, such as editing all viewports, administering filter definitions or reference records.

See also:

- User Logins (page 51)
- Starting Topobase Administrator (page 2)
- Starting Topobase Client (page 5)

Starting Topobase Administrator

You use Topobase Administrator to manage administration tasks. Use the tools to specify workspace (page 526), document (page 521), and configuration settings.

To start Topobase Administrator

1. Click Start menu ➤ Programs ➤ Autodesk ➤ Autodesk Topobase 2010 ➤ Autodesk Topobase Administrator 2010.
2. In the Topobase Login dialog box, enter your user name and password. The default user name is Topobase. You can leave the password empty.
3. At the first login, you must connect to the Topobase System User. Click Options>>> and enter the Topobase System User credentials. You specify these settings only once. See also the Topobase Installation and Configuration Guide, section CEV_ProdName_CEV Configuration.
4. Click Connect.
5. Click Workspace menu ➤ Open.
6. Select a workspace and click Open.
7. If no workspaces are available, click Cancel.

If you do not have an open workspace, only the basic setup commands are available in the Administrator explorer.

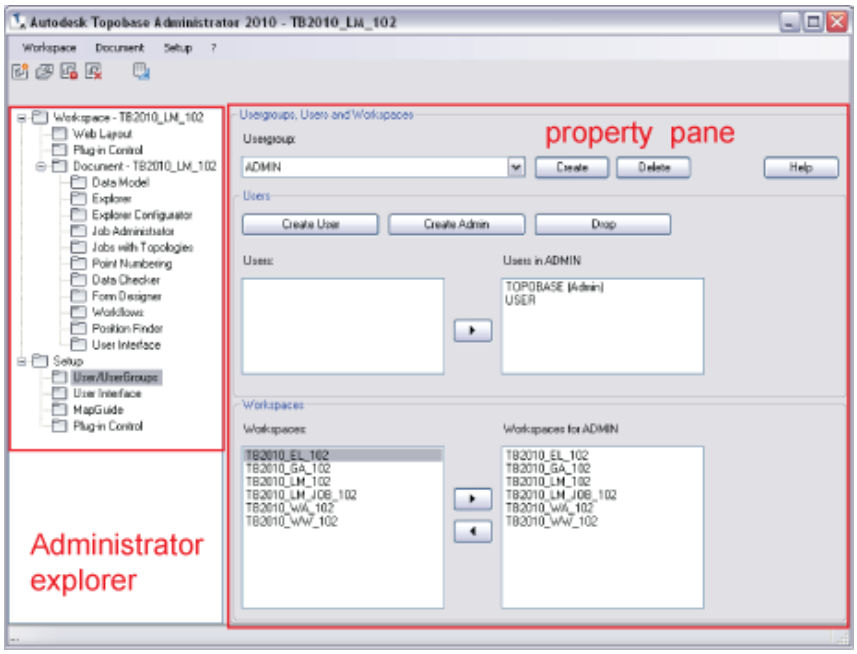
If you are prompted to update the structure of the document, click Yes. See also Data Model Code (page 37)

To create a workspace

- Click Workspace menu ➤ Create.
- See Creating a Workspace (page 116)

You can now work on all workspace, document, and configuration settings.

You start by clicking a command on the Administrator menus or by clicking an item in the Administrator explorer.



Topobase Administrator with explorer pane, menu bar, and property pane.

Variant: To start Topobase Administrator from Topobase Client

1. Click Start menu ➤ Programs ➤ Autodesk ➤ Autodesk Topobase Client ➤ Autodesk Topobase Client.
2. Log in as Administrator.
3. Click Setup menu ➤ Administrator.

See also:

- Administrator Explorer (page 6)

Password

If the password for the document (Oracle database) has been changed, for example, using an external tool, you can no longer open the workspaces with this document. In the Manage Document Password dialog box, either set a new password or remove the document from the workspace.

By default you can specify the document password when you import a document into a workspace.

NOTE For Oracle 11 g, by default, passwords expire after 180 days, and you must set a new password. Your Oracle administrator can change this behavior. To set a new password, for example, use the SQL Sheet, log in as SYSTEM user, and execute the command *alter user <name> identified by <new password>*. When you open your workspaces, in the Manage Document Password dialog box, set the new password accordingly.

See also:

- Adding and Removing Documents (page 53)

Starting Topobase Client

Topobase Client is the basic module of the Topobase desktop version.

To start Topobase Client

1. Click Start menu ➤ Programs ➤ Autodesk ➤ Autodesk Topobase 2010 ➤ Autodesk Topobase Client 2010.
2. When prompted to log in, enter your user name and password. In the Topobase Login dialog box, enter your user name and password. The default user name is Topobase. You can leave the password empty.
3. At the first login, you must connect to the Topobase Main or System User. Click Options>>> and enter the Topobase Main Or System User credentials. You specify these settings only once. See also the Topobase Installation and Configuration Guide, section CEV_ProdName_CEV Configuration.
4. Click Connect.
5. In the Open Workspace dialog box select a workspace and click Open.

If no workspace exists, click Cancel. Open the Topobase administrator to create a new workspace.

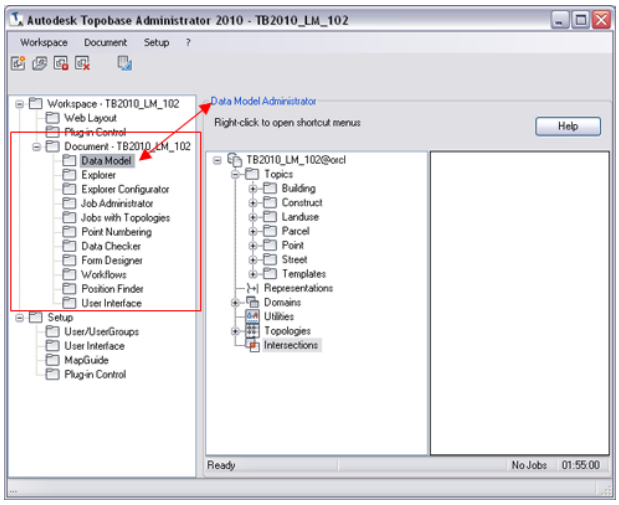
If you are prompted to update the structure of the document, click Yes. See also Data Model Code (page 37)

See also:

- Getting Started with Topobase Client
- Using the Standalone Client
- Starting Topobase Client (page 5)
- Setting Application Options (page 16)

Administrator Explorer

In Topobase Administrator you can use a number of workspace, document, and configuration tools.



The Topobase Administrator commands can be started from the Administrator explorer.

The Workspace node is only available if a workspace is open. It contains the workspace-specific settings and document nodes with document-specific settings. See also Document Settings (page 7).

The Setup node provides basic administration tools.

See also:

- Starting Topobase Administrator (page 2)
- Administrator Workspace Menu (page 8)

- Administrator Document Menu (page 9)
- Administrator Setup Menu (page 12)

Document Settings

In the Administrator explorer, click Document menu ➤ Settings to view the Document Settings. Alternatively, click the Document node.

The document settings are read only, except for the document password, and the Jobs Settings.

When you create a new document, you specify these settings. For more information about the settings see Data Model Administrator: Creating a Database (page 144).

Modules

You can add any module to the current document. However we recommend that you add only one module per document.

To add a module to the document

1. In the Document Settings, on the navigation pane, click the Modules link. Select the module to add. Click Save. Topobase Administrator adds the data model code to the system table TB_DATAMODELCODE.
2. When the document is reloaded, you will receive a prompt asking whether you want to update the database structure.

To remove a module from a document

➤ In the Document Settings, on the navigation pane, click the Modules link. Clear the module to remove.

This will not delete the feature class tables of this data structure. However the module will not be updated any more.

Extensions

You can add extensions to the document. Extensions are database enhancements such as COGO, Dimensions, Profiles, or Plot.

To add the Dimensioning extension to the document

1. In the Document Settings, on the navigation pane, click the Extensions link. Select Dimension Extension. Click Save. Topobase Administrator adds the topic Dimension to the document.
2. When the document is reloaded, you will receive a prompt asking whether you want to update the database structure.

See also Adding Dimensions.

To add the Plot Extension

1. In the Document Settings, on the navigation pane, click the Plot link.
2. See Configuring Plots (page 417).

To add the Oracle Data Import Extension

1. In the Document Settings, on the navigation pane, click the Oracle Data Import link.
2. See Topobase Data Transfer (Oracle Data) (page 239).

Jobs

To configure jobs

1. In the Document Settings, on the navigation pane, click the Jobs link.
2. See Job Enabling (Document Settings) (page 299).

Administrator Workspace Menu

Use the Workspace menu commands to configure workspaces, as shown in the following table:

Workspace menu commands	Description
Create	Creates a new workspace. You can also create a new document (page 521) at the same time.
Open	Opens a workspace.

Edit	Administers workspaces and documents. See also Adding and Removing Documents (page 53).
Close	Closes the workspace.
Drop	Deletes the selected workspace. This command deletes only the workspace. You can also choose to remove Oracle database users.
SQL Sheet	Opens the SQL Sheet. See Using the SQL Sheet (page 130).
Web Layout	Specifies the Web layout. See also Setting Up Topobase Web (page 63).
Plug-In Control	You can switch off Plug-Ins for the workspace to improve performance. **NOTE** You can also switch the Plug-In controls for all workspaces or documents using Setup menu ➤ Plug-In Control. If you switch off a Plug-In for all workspaces, the Plug-In will not be available in specific workspaces, even if it is switched on.
Exit Administrator	Quits the Administrator.

Administrator Document Menu

Use the Document menu commands to create and customize Topobase documents.

The Document menu is available only when a workspace is open. A workspace can contain more than one document (page 521). Select the appropriate document in the tree view before starting the commands, as shown in the following table:

Document menu commands	Description
Settings	Opens the Document Settings panel. See also Document Settings (page 7).
Data Model	Starts the data model administrator.

	See also Data Model Administrator (page 36).
Explorer	Starts the Explorer Manager. See also Configuring the Document Explorer (page 67).
Explorer Configurator	Defines the document explorer tree views. Detailed API knowledge is required. See also Explorer Configurator (page 103).
Job Administrator	Starts the Job administrator to create and edit job templates and transition passwords. See also Jobs Introduction (page 295)
Jobs With Topologies	Sets document job options. See also Setting Job Topology Document Options (page 27).
Point Numbering	Configures automatic numbering for point features. See Topobase Point Numbering (page 42).
Data Checker	Checks for data errors. See Topobase Data Checker (page 37).
Form Designer	Starts the form designer. See also Form Designer Introduction (page 309).
Plot	Available for plot enabled documents. Specifies plot and island plot settings. See Plot Settings (page 418).
Profile Designer	Starts the Profile Designer. Sets up profile definitions for the Topobase Client, Profile Manager. See Overview of Profiles (page 485). Available for documents that contain the Profile Extension.
Workflows	Defines business rules for data acquisition, analysis, or reporting. See also Workflow Introduction (page 481).
Position Finder	Defines search statements. See also Position Finder Introduction (page 463).
User Interface	Defines the document specific user interface for each user group (page 526). See Document Administration (page 29)

	NOTE You must re-open the workspace after you have modified the settings.
Report Designer	Creates and Designs reports. See also Report Designer Introduction (page 425).
Electric Explorer	Available for Electric documents. Defines user group specific profiles for the Electric Explorer. See Setting Up Electric Explorer Profiles (page 70).
SQL Sheet	Opens the SQL Sheet. See Creating SQL Expressions (page 103).
Form Manager >	Administers feature class forms.
Form Import Form Export	Exports or imports form definitions. Select the topics and feature classes whose forms you want to export. Select Others to export a form definition of a common table that belongs to no feature class or domain and is not a TB_ system table. The forms are exported into an XML File.
Form Check	Removes unused form definitions. All form layouts and definitions without a parent feature class are removed.
Form Performance Hints	Checks the form definitions (database indices) and provides hints to improve performance. Hints are written into a log file.
Export	Exports the following components:
Document Settings	Exports the document options to an XML file: Menus (Basis), Menus (Single), toolbars, form toolbars and menus, FlyIns and reports. See also Exporting Document User Interface Settings (page 34).
Oracle Dump	Exports the document as an Oracle dump file (*.*dmp*).
Import	Imports the following components:
Document Settings	Imports document options that have been exported to an XML file.

Interlis Structure Import	Opens the Interlis Structure Import wizard. See Interlis Import.
Oracle Data	Transfers data between two Topobase documents. See Topobase Data Transfer (Oracle Data) (page 239). The Oracle Data Import is only available for documents that contain the Oracle Data Import Extension. See Document Settings (page 7).

Administrator Setup Menu

Use the Setup menu commands to configure users and user groups (page 526), as shown in the following table:

Setup menu commands	Description
Security Administrator	Opens the Security Administrator. See Security Administrator (page 57).
User/User Groups	Configures Topobase users and user groups. See also Topobase Administration: User Groups (page 52) Setting Up Users and User Groups (page 13).
User Interface	Customizes the application related Client user interface for each user group. Click the tabs to configure the Standalone Client toolbar, and the TB commands. See also Customizing the Client Menu Bar (page 15). Setting up the Client Commands (page 14).
MapGuide	Specifies the Autodesk MapGuide credentials. See also Setting Up Topobase Web (page 63).
Plug-In Control	Switches off Plug-Ins. For example, To improve performance, clear the Plug-Ins that you do not want to be loaded.

See also:

- Understanding Users and Workspaces (page 50)

Setting Up Users and User Groups

You can create user groups (page 526) with specific rights and add users to the group.

To set up users and groups

1. In the Administrator explorer, expand the Setup node and click User/Usergroups.
2. In the User Groups section, select the user group to edit, for example ADMIN.
3. In the Users section, click the user and click the arrow button to add the user to the user group.
4. In the Workspaces section, click the workspace and click the arrow button to assign it to the user group.

Adding/removing users and user groups

Also, you can create or remove users and user groups.

To create a user

1. In the Administrator explorer, expand Setup and User/Usergroups.
2. Select the user group for the new user, for example ADMIN.
3. In the Users section click Create User.
4. Enter a name, for example, Jimmy.

The user is created and shown in the Users in <user group> list.

To create a user group

1. In the Administrator explorer, expand Setup. Expand User/Usergroups.
2. In the User Groups section click Create.
3. Enter a name, such as VIEWER.

4 Assign a user, either an existing one or create a new user, for example, by entering Jimmy.

To remove a user

1 In the Administrator explorer, expand Setup. Click User/Usergroups.

2 Select the user group you want to remove the user from, for example VIEWER.

3 In the Users section, select the user, for example Jimmy and click Drop.

To remove a user group

1 In the Administrator explorer, expand Setup. Click User/Usergroups. Click Remove.

2 Select the user group, for example VIEWER, and click Delete.

See also:

- Topobase Administration: User Groups (page 52)
- User Logins (page 51)
- Creating Workspaces (page 53)

Setting up the Client Commands

You can customize the Client user interface by enabling or disabling application related TB commands. When you disable a TB command, the command cannot be executed. That means, if you enter the command in the command line, or if you click the button on the Ribbon, a message notifies you that the command cannot be executed.

To customize the Client TB commands

1 In the Administrator explorer, expand Setup. Click User Interface.

2 Click the TB Commands tab.

3 Select the user group.

4 Select the commands you want to enable.

You can customize the Standalone Client toolbar for different user groups. For example, you can hide icons that are not available for a user group.

To customize the Standalone Client toolbar

1. In the Administrator explorer, expand Setup. Click User Interface.
2. Click the Toolbars tab.
3. Select the user group.
4. Expand the toolbar you want to customize, for example Main.
5. Select the icons you want to provide. Clear the check boxes for icons you do not want.

See also:

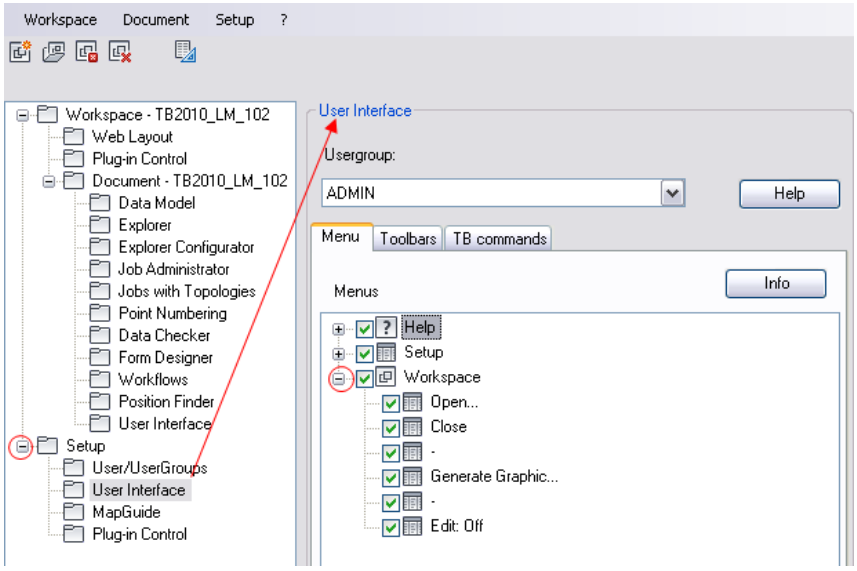
- Configuring Topobase Commands (page 32)

Customizing the Client Menu Bar

You can customize the Client user interface for different user groups. For example, you can hide or disable commands for a user group.

To customize the Standalone Client menu bar

1. In the Administrator explorer, expand Setup. Click User Interface.
2. In the User Interface section, in the Usergroup list, select the user group, for example ADMIN.
3. Click the Menu tab.
4. Expand the menu you want to customize, for example Workspace.
5. Select the menu items you want to provide. Clear the check boxes for items you do not want.



Customize the Standalone Client menu bars for different user groups.

Setting Application Options

Application options are stored in the Topobase system database schema TBSYS (page 524). Use Topobase Client to set application options.

To set application options

1. Start Topobase Client, and open the workspace.
2. Click Settings tab ➤ Setup panel ➤ Application Options.
3. Click an option in the tree view.
4. In the right pane, edit the option settings.

Setting Common Application Options

You use the common application options to specify layout, directories for log files and user settings.

To set common application options

1. Click Settings tab ➤ Setup panel ➤ Application Options.
2. In the tree view, click Common.
3. In the right pane, edit the options.

Common options	Description
Load Last Open Forms	Upon login, opens feature class forms that were previously open until you closed the workspace.
Remember Last Workspace	Upon login, opens the last workspace you used.
Report Root Directory	Specifies the folder in which to save report templates. The default directory is <topobase>\Reports.
Temp. and Log Directory	Specifies the temporary and log directories. The default directories are <topobase>\temp and <topobase\log. For example, the log files of document imports are stored in the log directory.
Data Directory	Specifies the directory for data. The default directory is <topobase>\Data. For example, the word serial letter templates are stored in the data directory.

Setting Map Options

Map Options apply to the AutoCAD Map connection. Use Map Options to specify options for all documents (page 521).

To set Map Options application options

1. Click Settings tab ➤ Setup panel ➤ Application Options.
2. In the tree view, click Map Options.
3. In the right pane, edit the options, as shown in the following table:

NOTE Changes to Map Options apply immediately. Document level options override some Map Options.

Map Options	Description
Display Info about Label's Parent	Redirects the feature information request from the label feature class to the parent. If this check box is selected, you can click a label in the map to view information about the related object. For example, click a point number to view information about the point. If you clear the check box, only label information is displayed. See also System Table TB_GN_INFO_REDIRECT (page 77).
Open Form In Edit Mode After Digitizing	Opens a feature class form in edit mode. When you use Create A Feature With Form the form is opened after digitizing. This option makes the form ready for additional attributes.
Vicinity For Digitizing Points [m]	Searches for points that already exist in the database, when creating a new point feature. This option finds points that are in the vicinity of the new point and that have defined values for "Z (height)" and "QUALITY". If points are found, these attribute values are applied to the new point. In the map document options, you can specify feature classes for the vicinity search. Search results are displayed in the command window.
Always Display Highlighted Features	Regenerates the graphic before highlighting the features. Use this option to generate graphics to display the features that are currently excluded by a filter setting, such as a viewport, or any other filter entered in Display Managers context menu Query to filter data. For example, when you highlight tracing results.
Zoom For Feature Highlight/Zoom GoTo	Specifies the zoom factor for highlighting the current feature of a feature class form or in the feature explorer using the highlight button.
Configuration For Point Feature Classes	Specifies the viewport. Enter fix values.

Configuration For Other Feature Class Types	Specifies the viewport. Enter a magnification factor as follows: ■ 0 = The current viewport does not change when you highlight a feature. ■ between 0 and 1: Zooms in. For example, enter 0.5 to zoom to the features and then zoom in with a factor of 2. ■ > 1: Zooms out. For example, enter 2 to zoom to the features with a factor of 2. The viewport size is calculated based on the feature extension.
Perimeter Display Options	Specifies how a job perimeter is displayed in the map. You can highlight the perimeter with a solid or hatch pattern. See also Configuring Jobs (page 299).
Vicinity Search	Specifies the feature classes to which the vicinity search shall be applied.

NOTE To set options for individual documents, use Document Map Options. See also Setting Document Map Options (page 25).

Setting COGO Application Options

You can use COGO application options to define settings for Topobase construction functions.

To set COGO application options

1. Click Settings tab ➤ Setup panel ➤ Application Options.
2. In the tree view, expand COGO.
3. In the right pane, edit the options, as shown in the following table:

COGO	Description
Right Angle Course	Specifies the mode for right angle course calculation. Select either:

	■ Point Mode—The right angle course is defined by an arbitrary sequence of known points and measured sections. ■ Base Line Mode—The right angle course is defined by two existing points that specify the base line section followed by measured sections. See also the Topobase Client User Guide, section Constructions: Right Angle Course.
Precision	Specifies calculation precision. The default is 0.001.
Split Tolerance	Specifies whether a line can be considered straight after inserting or extracting a point. This value specifies the maximum distance of a point from the line, which must be greater than the spatial tolerance. See also the Topobase Client User Guide, section Constructions: Line Processing (Vertices).
Split Line After Inserting Point	Splits the line after you use the Insert Point command.
Point Insertion Settings	Specifies how the Precision and Reliability attributes are set by the Survey application when the Insert Point command is used to insert a point into a line. When an existing point is projected onto a line, its reliability is improved and its coordinates are changed.
Precision	Defines the feature attribute where the precision of an inserted point is saved.
Reliability	Defines the feature attribute where the reliability of the inserted point is saved. This attribute usually refers to a domain *_TBD. You can specify the value (*_TBD.ID) for the two states Reliable and Not Reliable. For example, reliability values could be stored in the domain table RELIABILITY_TBD where ID = 1 is not reliable ID = 2 is reliable

Reliable	Specifies the reliable value ID. For example 2.
Is Not Reliable	Specifies the not reliable value ID. For example 1.

See also the Topobase Client User Guide, section Construction Introduction.

Setting Jobs Application Options

You use Jobs application options to specify the behavior of Topobase Client when you select a job to work with.

To set Jobs application options

1. Click Settings tab ➤ Setup panel ➤ Application Options.
2. In the tree view, click Jobs.
3. In the right pane, edit the options, as shown in the following table:

Jobs Options	Description
Lease Options	Defines whether you want to automatically lease jobs.
Automatically Lease the Selected Job	Leases a job automatically when selected. If you select this option, you can enter a leasing period.
Duration of Job Leasing	Specifies the period for automatic job leasing. The leasing period starts as soon as the job is selected.

NOTE You can lease jobs manually using the Job Manager.

See also:

- Jobs Introduction (page 295)
- Job Leasing (page 305)
- Job Manager and Job Toolbar

Setting Document Explorer Options

You use the Topobase document explorer (page 521) options to control general user interface behavior.

To set document explorer options

1. Click Settings tab ➤ Setup panel ➤ Application Options.
2. In the tree view, click Document Explorer.
3. In the right pane, edit the options, as shown in the following table:

These settings take effect when you reopen the workspace.

Document Explorer options	Description
Double Click Opens The Form	Opens feature class forms or starts workflows by double-clicking the feature class or workflow item.
Display Information About Selected FeatureClasses	Displays the table name and F_CLASS_ID in the status bar at the bottom of the Document explorer. This information is useful for administrators.

Setting Generate Graphic Application Options

You use the Generate Graphic application options to control the Topobase viewport (page 526).

To set Generate Graphic application options

1. Click Settings tab ➤ Setup panel ➤ Application Options.
2. In the tree view, click Generate Graphic.
3. In the right pane, edit the options, as shown in the following table:

These settings take effect when you reopen the workspace.

Generate Graphic options	Description

Viewport Creation	Specifies default values for the extension of a viewport. These values can be modified later in the Advanced Generate Graphic dialog box. See also the Topobase Client User Guide, section Displaying Features In Your Map.
Configuration For Point Feature Classes	Specifies the default height and width when you define a viewport using a point in the center of the drawing.
Configuration For Other Feature Class Types	Specifies a magnification factor when you define a viewport with a non point feature in the center of the drawing. The viewport size is calculated based on the feature extension. Enter a magnification factor as follows: ■ 0 = A selected polygon feature is used. ■ between 0 and 1: Zooms in. The filter rectangle is smaller than the bounding box of the selected geometry. ■ 1 = No change. ■ > 1: Zooms out. The filter rectangle is greater than the bounding box of the selected geometry.
Display Model	Specifies settings for display models.
Repository Base Path	Specifies the path where the display model files such as (**.tbdm*) and (**.layer*) are stored. See also Storing Display Models (page 381).
Drawing Options When Generating A Display Model	Specifies whether to open a new drawing each time you select a new display model, and generate graphic.
Close Current Drawing	Closes the current drawing. Opens a new drawing for each display model map of the selected Display Model. For example, you select a display model DM1, and generate graphic to populate *Drawing1.DWG*. Then, you select another display model DM2. *Drawing1.DWG* is closed, and a new drawing *Drawing2.DWG* is populated.
Keep Current Drawings Open	Keeps the current drawings open, and opens a new drawing for each display model map of the selected Display Model.

Reuse Active Drawing	Loads the first display model map of the selected Display Model into the active drawing (**.DWG*), such as an AutoCAD drawing (**.DWG*) that contains projected parcels. Then, you select a display model, and generate graphic. The display model adds the feature layers to your active drawing, and you can use the command Create New Feature From Geometry to transfer the projected parcel geometry into the database.

Setting Other Application Options

In addition to the basic application options, you can specify other customer-specific options.

To set other application options

1. Click Settings tab ➤ Setup panel ➤ Application Options.
2. In the tree view, click Plug-Ins.
3. In the right pane, edit or specify other options.

Setting Document Options

You use the Document options to control settings that are specific to only one document (page 521). These settings are saved either in the document or in the system database schema TBSYS (page 524). In order to edit the document options, the workspace must be open.

To set document options

1. Start Topobase Client, and open the workspace.
2. Click Settings tab ➤ Setup panel ➤ Document Options.
3. Click an option in the tree view.
4. In the right pane, edit option settings.

Document options work in addition to application options, and are specific to each document.

NOTE Some document options are only available to members of the Admin user group (page 526).

Common Document Options

You cannot edit common document options. They provide general document information.

To view common document options

1. Click Settings tab ➤ Setup panel ➤ Document Options.
2. In the tree view, click Common. The options in the following table are displayed:

Common options	Description
Database Schema	Name of the Oracle® database schema: <schema>@<oracle service>
Document ID	Unique identifier.
Version	Version of the document structure. This value changes whenever the data model is updated.
TBSYS Version	Version of the (*.jar) file that controls server-side functionality. This value can be different from the document structure version. However the first number has to be identical in both.
Keys/Data Model Codes	These values identify different data models. For example, the DM Code (Data Model Code) controls the scope of necessary data model updates. See also System Tables: Data Model Code (page 73).

Setting Document Map Options

You use the document Map options for settings, that are specific to only one document. These settings override the general Map application options.

NOTE You set general Map options using the Map application options. Using the document Map options, you can override the Map application options.

To set document Map options

1. Click Settings tab ➤ Setup panel ➤ Document Options.
2. In the tree view, click Map Options.
3. In the right pane, edit option settings.

Vicinity Search—In the document map options you can specify single feature classes to which the search shall be applied, in addition to the application map options.

See also:

- Setting Map Options (page 17)

COGO and Dimensioning Document Options

You use the COGO and dimensioning document options to control construction issues, that are specific to only one document.

To set COGO and dimensioning options

1. Click Settings tab ➤ Setup panel ➤ Document Options.
2. In the tree view, click COGO And Dimensioning.
3. In the right pane, edit option settings.

COGO and Dimensioning area	Description
Delete Construction Features From The Document	Deletes the temporary COGO features to clarify the construction workflow. Normally these are deleted as soon as Topobase COGO is finished.
Dimension Feature Class For	Specifies the dimension feature classes to store dimensioning that can optionally be created when you use the COGO commands.

COGO and Dimensioning area	Description
Orthogonal Calculation	Selects a dimension feature class that stores the orthogonal dimensioning.
Arc Intersection	Selects a dimension feature class that stores the arc dimensioning.
Dimensioning area	Settings for dimensioning.
Constant Offset For Aligned Dimensioning.	Specifies an offset value for aligned dimensioning.

NOTE You set general COGO options using COGO application options.

See also:

- Setting COGO Application Options (page 19)

Setting Job Topology Document Options

You can specify when area topologies can be checked. Use the Jobs With Topologies option to specify at which transitions a topology check shall be executed. It is recommend that you check the topologies when the job transition state changes from Pending to Live.

To set job topology options

1. Start Topobase Administrator and open the workspace.
2. Click Document menu ➤ Jobs With Topologies.
3. Select the Topology, and select the state changes, at which to check the topologies.
4. Optionally select Allow Job State Change In Case Of Invalid Topologies, if you want to change the job state, even with topology errors detected.

See also:

- About Job States (page 308)
- Setting Jobs Application Options (page 21)

Setting Wastewater Document Options

The Wastewater options item is only available in wastewater documents.

To set wastewater options

1. Click Settings tab ➤ Setup panel ➤ Document Options.
2. See Set Wastewater Document Options.

Setting Electric Document Options

The Electric options items are only available in electric documents.

To set electric options

1. In Topobase Client, open an Electric NA or Electric CE workspace.
2. Click Settings tab ➤ Setup panel ➤ Document Options.
3. In the Document Options dialog box, in the tree view, click Electric Explorer or Electric CE Explorer depending on your version of the Electric module.
4. Set any of the following options:
   - General: Specify how many features to retain in the Last Used feature lists. Specify whether to show a form automatically upon feature creation. Specify the amount of time to pause before moving to the next cross section when creating conductors.
   - "Transmission" Feature Containers: In Electric NA, specify the display of Transmission feature containers.
   - "Underground" Feature Containers: In Electric CE, specify the display of Underground feature containers.
   - Station Internal: Specify the display model to use for the station internal drawing. Define the default station internal bounding box.
5. If you are using Topobase Electric CE, click the Electric CE node and define the behavior of multi conductors.

   Select or clear the checkbox as needed for The Connection Layout Is Only Applied On Structural Features. If you select the checkbox, the layout is used by multi conductors whose start and end points are located at

Structure Features. If you clear this checkbox, the layout is used for all multi conductors whose start and end points are located anywhere.

6 When finished, click OK.

See also the Topobase Electric User Guide, section About This Guide.

See also:

- Setting Up Electric Explorer Profiles (page 70)

Document Administration

You can customize the document-specific user interface by specifying which tool bars are available for specific user groups. For more information, see the following sections.

To customize the document-specific user interface

1 Start Topobase Administrator and open a workspace.

2 In the Administrator explorer, expand the document. Click User Interface.

3 In the User Interface section, define the interface by doing one or more of the following:

- Click the Menus (Basis) tab to configure general shortcut menus for Document explorer items. See also Setting Up General Shortcut Menus (page 30).
- Click the Menus (Single) tab to configure specific shortcut menus based on the general settings. See also Setting Up Specific Shortcut Menus (page 32).
- Click the Toolbars tab to configure tool bars for the Topobase Standalone Client, and the Topobase Web version.
  - For Topobase Client and Topobase Web - Basic Layout, use the Construction Toolbar, and the Main Toolbar setting.
  - For Topobase Web - Flexible Layout, use Fusion Common Toolbar, and Fusion COGO Toolbar.
- Click the Forms tab to configure forms tool bars and shortcut menus. See also Customizing Form Toolbars and Menus (page 33).

- Click the FlyIns tab to specify the FlyIns. See also Document: FlyIns (page 34).
- For Topobase Electric documents: Click the Feature Container tab to specify the menu items that are available for the selected user group in a feature container, such as the Electric Explorer.
- Click the TB Commands tab to specify which TB commands are enabled for the selected user group. See also Configuring Topobase Commands (page 32)

NOTE You must re-open the workspace after you have modified the settings.

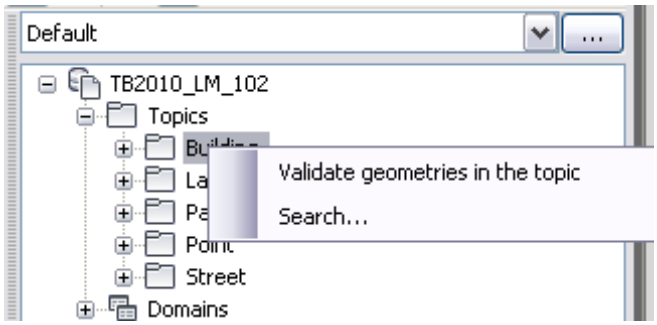
For more information about user groups, see Topobase Administration: User Groups (page 52).

Setting Up General Shortcut Menus

You use the Menus (Basis) settings to customize shortcut menus for items in the document explorer (page 521). You can control the functionality available for each user group (page 526). For example, you can limit the commands that members a user group can use.

The document explorer includes items such as feature classes, topologies, or intersection. See also Configuring the Document Explorer (page 67)

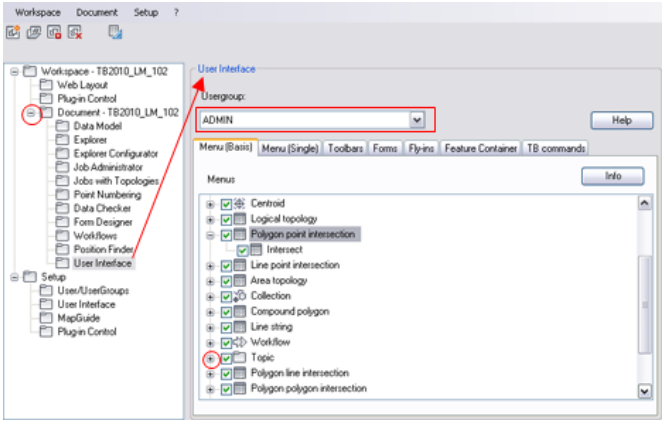
In general, you can perform all operations using shortcut menus.



For each user group, you can define which shortcut menus are available for each item.

To set up general shortcut menus

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, expand the document. Click User Interface.
3. In the User Interface section, click the Menu (Basis) tab. The list displays all types of explorer items.
4. Select a user group, for example ADMIN.
5. Select an item, for example Topic and expand it.
6. Select the menu commands that you want to be available.



Enabling the commands for the Topic shortcut menu for the ADMIN user group.

There are two steps in customizing shortcut menus. First, use Menus (Basis) to define which menus are available for each type of Document explorer item, such as a point feature class or a topology. Next, use Menus (Single) to further define which specific menus are available for each individual item, such as a specific feature class.

See also:

- Setting Up Specific Shortcut Menus (page 32)

Setting Up Specific Shortcut Menus

You can customize the shortcut menus for each specific item in the Document explorer. In addition to general settings that apply to all items of one type (such as all topics) you can specify shortcut menus for individual items (such as an individual topic).

To set up specific shortcut menus

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, expand the document. Click User Interface.
3. In the User Interface section, click the Menu (Single) tab. The list displays all explorer items of the selected document.
4. Select a user group, for example ADMIN.
5. Select a menu item, for example the topic Building, and expand it.
6. Select the menu commands that you want to be available.

Configuring Topobase Commands

For the Topobase Client version, you can customize the document related TB commands that are available for each user group.

When you disable a TB command, the command cannot be executed. That means, if you enter the command in the command line, or if you click the button on the Ribbon, a message notifies you that the command cannot be executed.

To disable or enable TB commands

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, expand the document. Click User Interface.
3. In the User Interface section, click the TB Commands tab.
4. Select a user group, for example ADMIN.
5. Select the commands that you want to provide.

See also:

- Setting up the Client Commands (page 14)

Customizing Form Toolbars and Menus

Topobase Client provides feature class forms, containing icons, tools, and shortcut menus to process data. Use the Form Toolbar/Menus option to define the icons, tools, and shortcut menus that can later be assigned to different feature class forms and customized for a user group.

To customize form tool bars and menus

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, expand the document. Click User Interface.
3. In the User Interface section, click the Forms tab.
4. Select a user group, for example ADMIN.
5. Select a form type, for example Domain.
6. Select the icons you want to be available on the toolbar.

Topobase includes the following default toolbar and menu configurations:

- Domain—In a domain table form you do not need a Create Feature icon, only an Add New icon.
- Feature—The feature class forms toolbar/menu contains Create Feature and Digitize New Geometry icons/commands.
- Label—In the label feature class forms toolbar/menu you need neither the Create Feature nor the Add New icon/command, because labels usually are created using the Label Placer (page 523).
- Others—You can define your own types of form tool bars/menus. Use the Add button.

NOTE For technical reasons, the configuration list of icons and menu commands shows all possible items even though some are hidden by the application. For example, the Domain toolbar list displays the Highlight and Digitize commands as selected. However, they will never be included in the Domain toolbar.

For more information about the tools and menu commands, see the Topobase Client User Guide, Using Feature Class Forms.

Exporting Document User Interface Settings

You can export the document user interface configuration. For example, you have set up the document user interface for different user groups, such as ADMIN, EDITOR, and VIEWER. You want to share this configuration with other offices or customers who work with the same document. User groups that are not available on the target system are created during the import.

Use the Document Settings export command to export the following settings (**.xml*).

- User group specific document explorer shortcut menus, such as shortcut menus for the VIEWER user group that contains only the Show Forms command.
- User group specific document tool bars, such as a Main toolbar for the VIEWER user group that does not contain the Delete Feature icon.
- User group specific feature class form tool bars and shortcut menus, such as a toolbar for the VIEWER user group that does not provide edit functionality.

To export document user interface settings

1. Start Topobase Administrator and open the workspace.
2. In the Administrator explorer, select the document.
3. Click Document menu ➤ Export ➤ Document Settings.
4. In the Document Settings Export dialog box, select the settings to export.
5. Click Export.

You can import the settings file (**.xml*) using Document menu ➤ Import ➤ Document Settings.

Document: FlyIns

FlyIns are components of a Topobase task pane. Like feature class forms, they can be either of the following:

- docked in = included in the application window
- docked out = independently flying on the desktop.

Some examples of FlyIns:

- Document Explorer
- Job Management
- Workflow Explorer

For each document and user group, you can define which FlyIns are available.

To view available FlyIns

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, expand the document. Click User Interface.
3. In the User Interface section, click the FlyIn tab.
4. Select a user group, for example ADMIN.
5. Select the FlyIns you want to provide.

You can either display or hide the available FlyIns with the Document > Toolbars/FlyIns command. Also, you can switch between the Workflow and Document explorer with the icons on the Document toolbar.

NOTE Some FlyIns are only available in the Topobase Web version.

Topobase Data Modelling

This section describes Topobase data models. A data model includes topics, feature classes, topologies and intersections. You create data models using the Topobase data model administrator.

See also:

- Exploring a Data Model (page 118)

Data Model Administrator

Topobase data model administrator provides a user interface where you can create, process and administer data structures. It provides tools to do the following:

- Create or delete a document (page 521).
- Create and edit topics and feature classes.
- Create domains.
- Create label feature classes and define labels.
- Update label text.
- Enable versioning of feature classes.
- Add attributes to feature classes.
- Build compounds.
- Create topologies.
- Initialize and edit topologies.
- Define intersections.
- Create Schematics and Generalizations.
- Define feature rules.

See also:

- Starting the Data Model Administrator (page 143)

Data Model Templates

When you create a Topobase document, you can use a data model template, which is a predefined data structure.

Create a new document using a predefined structure

1. Start Topobase Administrator.
2. Click Workspace menu ➤ Create.

3 Enter a name, for example WS_Water.

4 In the right pane, in the Workspace Manager section, click New.

5 In the Create New Document dialog box, enter the General Settings.

6 On the navigation pane, click Modules, and select a module such as Water Data Model.

7 See Creating a Topobase Document (page 111)

NOTE You can also create a new document by importing a dump file with a predefined database schema.

See also:

- Creating Workspaces (page 53)
- Creating a Topobase Document (page 111)
- Data Model Administrator: Creating a Database (page 144)

Data Model Code

The data model code contains update information about the data model. The code is read when you open a workspace. If the data model is not up-to-date, you are prompted to start update routines.

NOTE The data model code is stored in the user data base schema in TB_DATAMODELCODES.

Topobase Data Checker

Use the Topobase Data Checker to validate the quality of your data. For example, define any set of consistency checks that can be executed after data has been imported into a document.

You can start the data checker either manually on demand or on a scheduled basis. For example, you run a set of checks when you are ready to change the job state.

In Topobase Client, when you start the Data Checker, the result of the checks, is displayed in the feature explorer, for example features with wrong or

inconsistent data. Optionally, you generate a report that logs the inconsistencies.

In a scheduled mode, the result is provided as a report that is sent via E-mail to a specified recipient. Then, you can resolve the conflicts on the whole.

To configure the data checker in Topobase Administrator

1. Open the workspace.
2. In the administrator explorer, select the document.
3. Click Document menu ➤ Data Checker.
4. Under Data Checker, use the General tab to create and edit checks. Use the Scheduled tab, to schedule the execution of the checks.

On the General tab use the icons and shortcut menus as show in the following table.

Data Checker General tab	Description
	New Check. Opens the Data Check dialog box, and creates a new data checker definition. See Creating a Data Check (page 39).
	Remove Check. Removes the selected check.
	Edit Check. Opens the Data Check dialog box for the selected data check. You can modify the description and the select statement.
	Import Check. Open the file selector. Select an (*.XML) file to import.
	Export Selected Checks. Exports the selected data checks into (*.XML) files.

On the Schedule tab use the icons and shortcut menus as show in the following table.

Data Checker Scheduled tab	Description
	New Scheduled Task. Opens the Scheduled Data Check dialog box, and creates a new scheduled task. See Scheduling a Data Check (page 41).
	Remove Scheduled Task. Removes the selected task.
	Edit Scheduled Task. Opens the Scheduled Data Check dialog box.

To start the data checker in Topobase Client

1. Open the workspace.
2. Click Tools tab ➤ Modify panel ➤ Data Checker.
3. In the Data Checker dialog box, select one or more checks.
4. Click Execute.

Creating a Data Check

Use Topobase Administrator to define and configure data checks.

Data Check dialog box	Description
Name	Specifies a name.
Parent Check	Specifies a parent data check for the selected data check.
Description	Specifies a description.

Check Statement	Specifies the SQL select statement that will be executed by the selected data check.
SQL Assistant	Starts the SQL Assistant. See also Using the SQL Assistant (page 105).
Row Level Security	Specifies the Row Level Security.
Data Check Applies To The Selected Job Or To The Live Job (For Scheduled Tasks)	Recommended. Select this option ■ To execute the data check in the current job - in case of manual start. ■ To execute the data check in the Live Job - in case of scheduled start.
Data Check Applies To The Entire Database	Select this option to execute the data check in the entire database.
Check SQL	Checks whether the SQL select statement is valid.

Check statements

A check statement is an SQL select statement that analyzes the data and returns errors. If the check statement does not return any rows, the data is valid.

The SQL select statement must return a set of FIDs, and should match the following skeletons.

- For simple data checks, use
  SELECT fid FROM <feature class> WHERE <checking condition> ORDER BY fid;
- For advanced data checks, use
  REPEAT <select statement> FOR <select statement>;

  This SQL select statement repeats the first select statement as long as the second select statement returns a result.

  In the first <select statement> you can use placeholders that represent columns of the second <select statement>.

Example

```
REPEAT
SELECT fid from $f_class_name WHERE <checking condition>
FOR
SELECT f_class_name from TB_DICTIONARY where f_class_name like
'EL%';
```

TIP Define any database view to perform the check, and use the Check statement to execute a SELECT on this view. For example: *SELECT fid FROM <database view> ORDER BY fid;*

Scheduling a Data Check

The Scheduled tab displays the scheduled task. Select a task in the data grid, and click Edit Scheduled Task to view the properties.

Data Check Scheduled tab	Description
Taskname	Displays the task name.
Checks	Displays the number of selected data checks.
Weekdays	Displays the weekday when check will be executed.
Start Time	Display the start time of the scheduled data check.
Report Recipient	Display the E-mail address the report will be send to.

Define a scheduled task as shown in the following table.

Scheduled Data Check dialog box	Description
Task name	Specifies the task name. The task name is used as E-mail subject.
Schedule	Specifies the days and the time when the data checks will be executed.
Report area	Specifies the E-mail addresses of originator and recipient.

SMTP area	Specifies the SMTP server, such as Host Name, Port, and authentication.
Checks	Selects the data check that will be executed in the selected task.

NOTE For Oracle 11: The Oracle 11 security model requires that you configure an ACL (Access Control List). Then, the scheduled job can send an E-mail with the data checker results. For more information, see Oracle 11 Users Guide and Reference.

Topobase Point Numbering

Automatic point numbering helps to manage long and complex point numbers, and makes point acquisition more efficient and less error-prone for the field crew.

Use Topobase Administrator to set up automatic point numbering for your documents. Automatic point numbering applies to new point features that are either imported, calculated, or digitized. By default, no point numbering is applied. Some Topobase applications, such as Survey require a point number.

Requirements: Automatic point numbering applies to point feature classes that have the default attribute TB_POINTNUMBER. See also Feature Class Type: Point (page 181).

NOTE Automatic point numbering does not support manual updates of the point number.

A Topobase point number consists of two parts.

- Repetitive prefix
- Increment that is auto generated, and incremented by 1

For example, the following point numbers have the prefix CT219611, followed by a 5 digit increment.

```
CT219611 50005
CT219611 50006
CT219611 50007
CT219611 50008
```

You configure the prefix, using four different point numbering types. You configure the increment by creating feature class groups.

Point numbering types

- Per Document
- Per Group
- Grid 100m
- Raster Plan

IMPORTANT We recommend that you do not change the point numbering type once it has been assigned.

To set up the automatic point numbering

1. Start Topobase Administrator, and open the workspace.
2. In the administrator explorer, expand the document node, and click Point Numbering.
3. On the right pane, under Point Numbering, configure the numbering type as shown in the following table.

PointNumbering settings	**Description**
Select The Point Numbering Type	Selects the numbering type. By default, no automatic numbering is used. Automatic point numbering is often used by the Land Management modules. For point numbering types, see Point Numbering Per Document (page 45) Point Numbering Per Group (page 46) Point Numbering Per Grid 100m (page 47) Point Numbering Per Raster Plan (page 49).
Settings	Specifies the selected numbering type.
Number Of Digits For Increment Value	Specifies the number of digits for the increment part of the point number. The number of digits must match the increment minimal and maximal value.

Exception Prefix	Specifies a prefix that is used when the maximum increment value is exceeded. For example, your increment minimal value is 1, and your increment maximal value is 999, so the system can generate point numbers for 999 points. For any additional point, the system uses the exception prefix. Then you can filter for these points, and modify the numbers accordingly.
Standard Prefix	For numbering type Per Document. Specifies the repetitive number part.
Group Settings	Selects the point feature classes for automatic point numbering. The explorer displays all point feature classes. Point feature classes that do not have the TB_POINTNUMBER attribute, are highlighted in red. Point feature classes that have not been grouped, are listed under the Unassigned FC node. You create one or more groups, and specify the point numbering rules for each group. Within a group, the same increment settings are applied, and the increment is unique.
Settings For Selected Group <name>	Specifies the numbering rule for the selected group.
Increment Minimal Value	Specifies the first value of the increment.
Increment Maximal Value	Specifies the maximum value for the increment.
Prefix Per Group	For numbering type Per Group. Specifies the repetitive number part for the selected group.
Number Example	Displays a sample point number using the specified settings.
Create	Creates a feature group. For example a group named Cadastre, or Construction. Use the arrow buttons, or drag an unassigned feature class to the group.
Rename	Renames the selected group.

TIP Press F2 to rename a group. Press F5 to refresh the tree view.

Point Numbering Per Document

Use the point numbering type Per Document to generate point numbers that are unique per document. All point number use the same repetitive number part. For the increment, you can define groups of feature classes, where each group uses a specific number range.

Example: You use the repetitive number part CT2196 for all points, however for projected points, you provide the numbers 1-49999, and for topographic points, you provide the numbers 50000-99999.

```
CT219611 50001 -> topographic point
CT219611 00001 -> projected point
CT219611 00002
CT219611 50002
CT219611 99999
ERCT 00001 -> number range exceeded
```

To set up a point numbering per document

In this example, you define point numbering for two feature class groups that use the same repetitive part, but different increments.

1. Start Topobase Administrator, and open the workspace.
2. In the administrator explorer, expand the document node, and click Point Numbering.
3. On the right pane, under Point Numbering, Select The Point Numbering Type, select Per Document.
4. Expand the Settings, and specify the repetitive part.

   Number Of Digits For Incremental Value: For example, 5.

   Exception Prefix: For example, ERRCT.

   Standard Prefix: For example, CT219611.
5. Expand the Group Settings to select the point feature classes.
6. Click Create, and enter the name, such as Control Points.

7 From the Unassigned FC node, select the feature classes to enable point numbering. Use the arrow buttons, to move the selected feature class under the group node.

8 Create other groups as needed, such as Topographic Points.

9 Select the feature class group, such as Control Points.

10 Under Settings For Selected Group, specify the increment number part.

Increment Minimal Value: For example, 1.

Increment Maximal Value: For example, 49999.

11 Select the next feature class group, such as Topographic Points.

12 Under Settings For Selected Group, specify the increment number part.

Increment Minimal Value: For example, 50000.

Increment Maximal Value: For example, 99999.

Point Numbering Per Group

Use the point numbering type Per Group to generate point numbers for feature class groups. For each feature class group, you specify the repetitive number part, and the increment.

To set up a point numbering per group

1 Start Topobase Administrator, and open the workspace.

2 In the administrator explorer, expand the document node, and click Point Numbering.

3 On the right pane, under Point Numbering, Select The Point Numbering Type, select Per Group.

4 Expand the Settings, and specify the repetitive part.

Number Of Digits For Incremental Value: For example, 5.

Exception Prefix: For example, Error.

5 Expand the Group Settings to select the point feature classes.

6 Click Create, and enter the name, such as Control Points.

7 From the Unassigned FC node, select the feature classes to enable point numbering. Use the arrow buttons, to move the selected feature class under the group node.

8 Create other groups as needed, such as Topographic Points.

9 Select the feature class group, such as Control Points.

10 Under Settings For Selected Group, specify the increment number part.
Increment Minimal Value: For example, 1.
Increment Maximal Value: For example, 99999.
Prefix Per Group: For example CO.

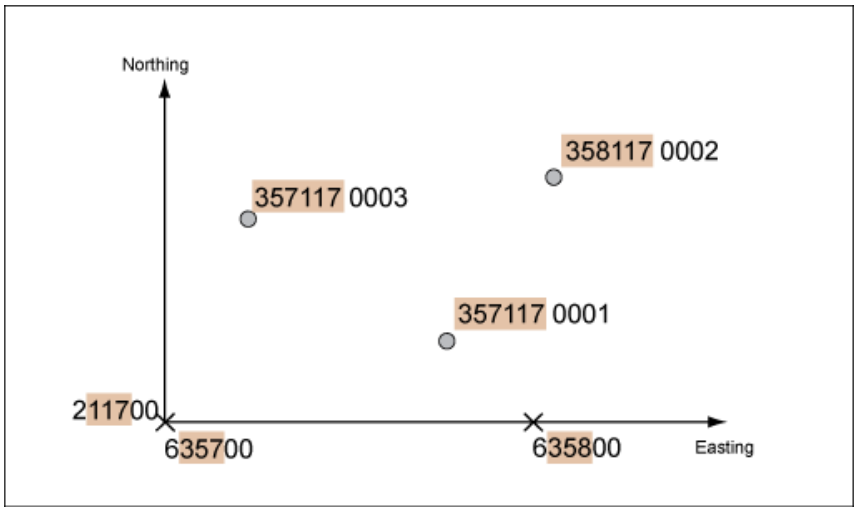
11 Select the next feature class group, such as Topographic Points.

12 Under Settings For Selected Group, specify the increment number part.
Increment Minimal Value: For example, 1.
Increment Maximal Value: For example, 99999.
Prefix Per Group: For example TOP.

Point Numbering Per Grid 100m

Select the point numbering type Grid 100m to generate a point number where the prefix is depending on the point coordinates.

- The repetitive number part is generated from the 100 <length unit> value of the coordinates: 3 digits of the easting value + 3 digits of the northing value. That means the repetitive part always consists of a 6 digit value.
- The increment part can be configured for different feature class groups.



To set up a point numbering per 100m grid

1. Start Topobase Administrator, and open the workspace.
2. In the administrator explorer, expand the document node, and click Point Numbering.
3. On the right pane, under Point Numbering, Select The Point Numbering Type, select Grid 100m.
4. Expand the Settings, and specify the repetitive part.

   Number Of Digits For Incremental Value: For example, 6.

   Exception Prefix: For example, Error.
5. Expand the Group Settings to select the point feature classes.
6. Click Create, and enter the name.
7. From the Unassigned FC node, select the feature classes to enable point numbering. Use the arrow buttons, to move the selected feature class under the group node.
8. Create other groups as needed.
9. Select the feature class group.
10. Under Settings For Selected Group, specify the increment number part.

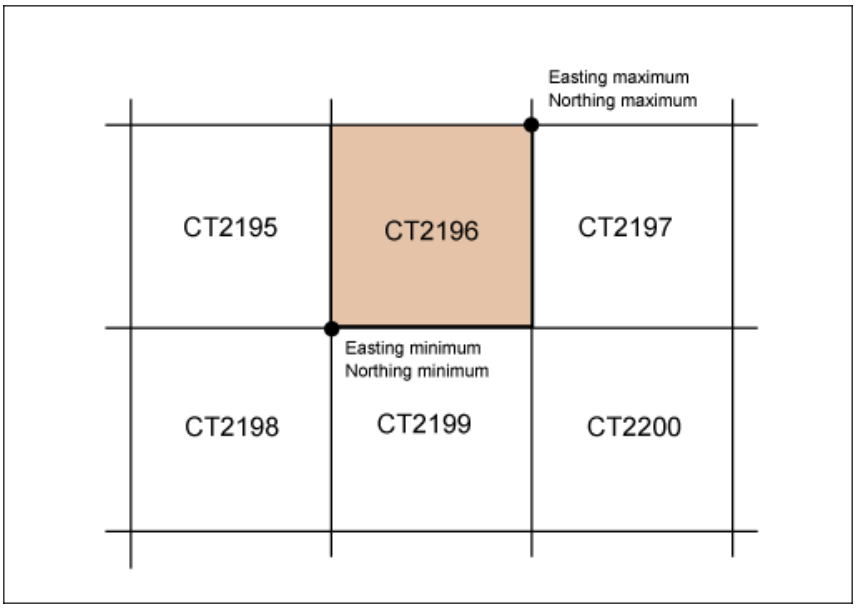
    Increment Minimal Value: 1

    Increment Maximal Value: 99999.

Point Numbering Per Raster Plan

Select the point numbering type Raster Plan to generate a point number where the prefix is depending on the location of the point in a specified raster.

- You specify the repetitive number part per raster. You either define your own raster, or you load a raster plan from the Land Management (CH) document.
- The increment part can be configured for different feature class groups.



To set up a point numbering per raster plan (Land Management (CH))

1. Start Topobase Administrator, and open a workspace that has access to the Land Management (CH) document.
2. In the administrator explorer, expand the document node, and click Point Numbering.
3. On the right pane, under Point Numbering, Select The Point Numbering Type, select Raster Plan.
4. Expand the Settings, and specify the repetitive part.

   Number Of Digits For Incremental Value: For example, 6.

Exception Prefix: For example, Error.

Under Raster Plan Definition, click Load.

For each raster, specify the prefix.

The Load button is only available for documents with the Land Management (CH) data model. Then, the prefix is loaded from LM_PL_PLAN.PLAN_NUMBER, and the values for Easting Minimum, Northing Minimum, Easting Maximum, and Northing Maximum are loaded from the plan topology feature class LM_PLAN_TSUR. This requires that the topology LM_PLAN needs to be initialized and without errors.

5 Expand the Group Settings to select the point feature classes.

6 Click Create, and enter the name.

7 From the Unassigned FC node, select the feature classes to enable point numbering. Use the arrow buttons, to move the selected feature class under the group node.

8 Create other groups as needed.

9 Select the feature class group.

10 Under Settings For Selected Group, specify the increment number part.

Increment Minimal Value: For example, 1.

Increment Maximal Value: For example, 999999.

Understanding Users and Workspaces

The database administrator prepares and configures the Oracle® database and prepares workspaces for Topobase users.

To prepare a workspace, do the following:

- Create workspaces for single or multiple documents, which are Oracle database schemas.
- Create one or more Oracle database schemas, such as Land Management, Waste Water, or Water.
- Use Document Administration to specify the functionality for different user groups. Specify menu bars and tool bars for documents.

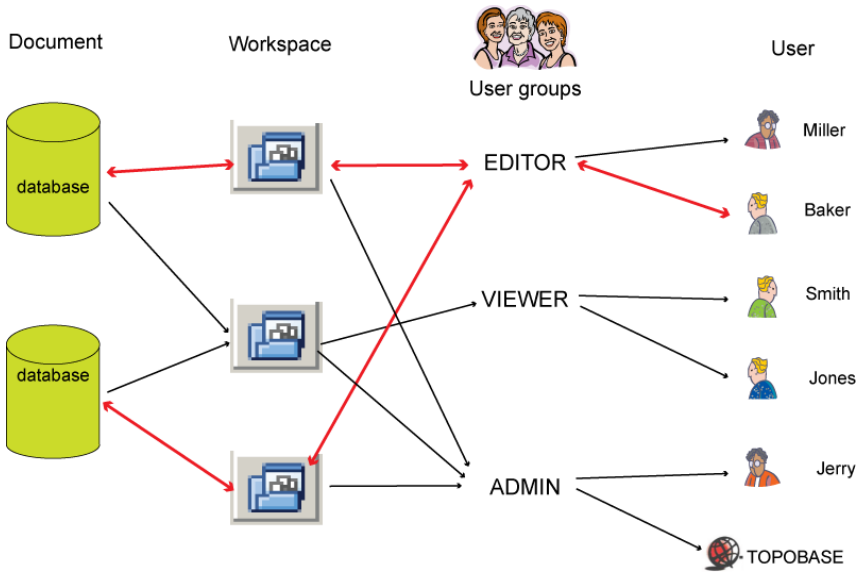
After preparing the workspaces, the database administrator makes them accessible for the different user groups.

To make workspaces accessible, do the following:

- Define user groups, such as ADMIN, EDITOR, and VIEWER.
- Assign the appropriate workspaces to user groups.
- Define unique personal logins for each user and assign users to appropriate user groups.

See also:

- Topobase Administration: User Groups (page 52)



Mr. Baker is a member of the EDITOR group. He can access two workspaces.

User Logins

Topobase provides a virtual personal login for each user. After installation you can start Topobase with the following default user:

- User name: Topobase
- Password: TOPOBASE (or empty).

Each user has a personal login. By default, the password is the user name (in upper case). If you leave the password empty, the user name is taken as the password.

NOTE The password is case sensitive. Note, that the default password, which is the same as the user name, is upper case.

To change a password

1. Log in using the old password.
2. Click Settings tab ➤ Setup panel ➤ Change Password.

See also:

- Starting Topobase (page 1)

Topobase Administration: User Groups

Topobase application users are grouped into user groups that have rights to access the database. The rights are defined by the administrator and typically, these include the following user groups:

- ADMIN User—Has unrestricted access to all data. All commands are available.
- EDITOR User—Can edit all or parts of the data. All editing commands are available.
- VIEWER User—Can view and analyses the data. No editing or administrator commands are available. The user interface provides only the icons and menus that are needed.

You can create as many user groups as necessary and define their rights in detail. In addition to data access rights, a user group has access to specific workspaces.

Use the Topobase Administrator Setup options to create user groups, add users, and assign workspaces.

See also:

- Setting Up Users and User Groups (page 13)

Creating Workspaces

As a user, you access the documents by opening a workspace.

To create a new document (page 521), you must create a new workspace or select an existing workspace.

To create a workspace

1. Start Topobase Administrator.
2. Click Workspace menu ➤ Create.
3. Enter a workspace name, for example WS_water, and click Create.

In the Workspace Manager section of the Administrator window, define the documents in the workspace.

Workspace Manager	Description
Workspace Name	Specifies the name.
Documents	Creates, imports or removes documents.
New	Creates a new document with an individual data model. The data model administrator is started, and you can specify the document properties. See also Data Model Administrator: Creating a Database (page 144)
Import	Imports a document. Two import methods are available. See also Adding and Removing Documents (page 53)
Drop	Removes a document from TBSYS (page 524). You can only remove documents that are not part of any workspace.

Adding and Removing Documents

You use the Workspace Manager to add to or remove documents from a workspace. You can only process documents if they are assigned to a workspace.

To add or remove documents from a workspace

1. Start Topobase Administrator and open the workspace.
2. In the Administrator explorer, click the workspace. In the right pane, the Workspace Manager is displayed.

   The documents of the current workspace are displayed in the Workspace Manager.
3. Specify options, as shown in the following table:

Workspace Manager	Definitions
Documents	Creates, imports or removes documents.
Available Documents In TB-SYS	Lists all existing database schemas.
Documents In Workspace <name>	Lists all documents of the current workspace.
Arrow buttons	Removes or adds documents to a workspace. Use the up and down arrow you can change the document order, which affects the display order of the tabs in the document explorer.
New	Creates a new Topobase database with an individual data model. The Data model administrator is started and you can specify document properties. See also Data Model Administrator: Creating a Database (page 144)
Import	Imports a document. Two import methods are available.
Import From Existing Oracle Schema	Adds an existing document, such as an Oracle database schema that has been imported with Oracle tools before.
Import From Oracle Dump File	Creates a document by importing an Oracle (*.dmp) file. See also Data Model Templates (page 36).

Drop	Removes a document from TBSYS. You can only remove documents that are not part of any workspace.

Importing a document into a workspace

Use the Create Workspace command to import documents from Oracle (**.dmp*) files. Workspaces cannot be imported.

Import a document into a new workspace

1. Start Topobase Administrator.
2. Click Workspace menu ➤ Create.
3. Enter a workspace name and click Create.
4. In the Workspace Manager section, click Import.
5. In the Import dialog box, select Import From Oracle Dump File.
6. In the File Selection dialog box select the (**.dmp*) file.
7. In the Dump Import dialog box, specify import options, as shown in the following table:

Dump Import options	Description
Dump File	Database templates are provided in the <topobase>\templates folder.
Oracle User	Specifies the database schema.
User name	Specifies the name of the Oracle database schema.
Password	Specifies the Oracle password.
Tablespaces	Specifies Oracle tablespaces.
Default	Specifies the default. Default is USER_DATA
Temporary	Specifies the default. Default is TEMPORARY DATA
Index	Specifies the default. Default is INDEX
Import Parameters	Controls the dump import.

Executable	Specifies the Oracle import function. Default is <ora>\bin\imp.exe
Arguments	Specifies the Oracle import arguments.

8 Click Import to continue.

9 Log in as Oracle user with DBA rights, for example SYSTEM.

10 Click Login.

The import log files are stored in the Log folder (default is <topobase>\Log.

The (**.dmp*) file is imported. If necessary, the document structure is updated. Click YES to update the document structure. It is recommended, that you compute the statistics after the import. Click Yes to compute the statistics.

11 View the import log files. They are saved in the same directory as the dump files.

The document is opened and shown both in the document explorer pane and the Administrator explorer.

NOTE During the import the spatial indexes are re-created automatically. If any problems occur, a message appears. In this case, you must validate the geometries manually, resolve the problems, and create the spatial index manually. Use Topobase Client to validate the geometries.

Export Topobase documents into a (.dmp*) file.**

1 Click Setup menu ➤ Administrator.

2 In the Administrator explorer, click the document to export.

3 Click Document menu ➤ Exports ➤ Oracle Dump.

4 Verify the export parameters, as shown in the following table:

Dump Export	Description
Output Directory	Specifies the directory for the (**.dmp*) file.
Export Parameters	

Executable	Specifies the Oracle export function. The default is <ora>\bin\exp.exe
Arguments	Specifies the Oracle export arguments.

5 Click Export.

6 View the log file.

Security Administrator

Use Topobase Security Administrator to define and setup Oracle users/roles with customized access rights on a Topobase document (main document). Autodesk Topobase provides several permission templates, such as Full Access, or Read Only.

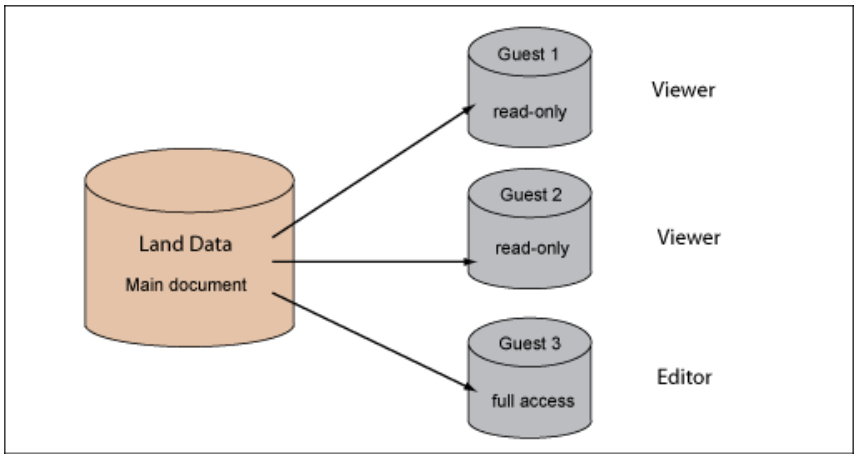
NOTE To use the Security Administrator, you must have DBA rights. You must be familiar with Oracle, and the Topobase database schema.

Main document—Document (Oracle schema) that contains your data.

Guest document—Document that accesses the main document by a role. You can create different roles to provide different levels of access to the main document. A guest document does not have any data tables, but only synonyms and privileges on the main document.

Guest user—Oracle user <guest> that has been created using the Security Administrator, and that has a role TBR_<guest> that defines the relation between the main document and the guest document. A guest document has some additional roles named TOPOBASE_* for internal use.

The role TBR_<guest> has specific rights on your main document, for example restricted access to some or all feature classes.



IMPORTANT You must manually maintain each role associated with the main document.

For example, when you introduce new tables to your main document, for example by adding a new feature class, domain, topology, utility, extension, or intersection, or by running a data structure update, you must adjust the settings for each new table, using the Security Administrator.

TIP In the main document, use the SQL Sheet to query the privileges on that user.

```
select * from user_tab_privs;
select * from user_tab_privs_made;
Select * from user_tab_privs_recd;
```

To create a guest document

1. Start Topobase Administrator.
2. Click Setup menu ➤ Security Administrator.
3. Log in as Oracle user with DBA rights, for example SYSTEM
4. In the Security Administrator dialog box, under Roles, click Add.
5. Enter a name for the Oracle guest user, and specify the password.
6. Select a permission template, such as Read Only.
7. Under Data To Access, select the document you want to define the role for.

8 Click Save.

9 Under Roles, select the guest user, and click Edit. On the Data tab, review the access rights of the role TBR_<guest>.

10 Click Close.

11 Click Workspace menu ➤ Open.

12 Select the workspace you want to add the guest user to.

13 Under Available Documents in Tbsys, select the guest user, and click the arrow icon to add it to your workspace.

NOTE A guest document is based on a role on the main document. Therefore you cannot edit the document related settings. To edit the document related settings, open a workspace containing the main document.

Managing Roles

Use the Security Administrator to create and edit Oracle roles.

In the Security Administrator dialog box, in the explorer, select the Role node, and click a button to add a guest user, or to edit a TBR_<guest> role.

Security Administrator - Roles	Description
Roles explorer	Displays the guest users.
Add	Select the Role node, and click Add to create a guest user with a role. See also Add a Role (page 60).
Edit	Edits the role of the selected guest user. See Edit a Role (page 62).
Duplicate	Duplicates the selected guest user and the role. Creates a guest user and a role using the same settings as the selected user. Enter a name for the duplicate guest user.
Delete	Deletes the selected guest user and the role.

Add a Role

You use the Security Administrator to create an Oracle guest user with a role.

To create a guest user with a role

1. In the Security Administrator explorer, select the Role node.
2. On the right pane, click Add.
3. Under Add Role, specify the permission settings as shown in the following table.

Add Roles	Description
Name	Name of the Oracle guest user, for example TB2010_LM_Guest.
Access	Specifies the password, and the table spaces for the guest user.
Data	Selects the main document to grant the access rights to.
Permission Template	Selects the permission template. See also Permission Templates (page 60).
Data To Access	Selects the main document.
Save	Creates the user and the role in the database. To edit the role, select the guest user, and click Edit. See Edit a Role (page 62).

Permission Templates

The Topobase Security Administrator provides permission templates, such as Full Access, or Read Only. You can define and reuse your own permission settings. Permission settings for the database objects View, Synonym, Package are stored in (**.SecDef*) files. For example:

```
SIUD--- LM_LANDUSE
SIUD--- LM_LANDUSE_L
SIUDEA- *
```

The letters indicate the rights, followed by a blank, and the object name. Use the object name *, to set the rights for all objects that are not yet defined in the previous lines. Also, you use * to filter LIKE.

(*.SecDef) key	Description
*	Indicates a comment line.
S	Select
I	Insert
U	Update
D	Delete
A	Alter
E	Execute
G	Direct Grant

Optionally you save your permission settings as permission templates. Permission templates can be used, when you add a new role. The templates are stored in the directory <topobase_administrator>\Data\SecurityAdministrator, as soon as you have used the Add command once.

To create a permission template

1. In the Security Administrator explorer, select the role.
2. On the right pane, click the Data tab.
3. Edit the settings.
4. In the grid, click File menu ➤ Save As Permission Template.
5. In the Save Permission Template dialog box, under New Data Setting, enter a name for the permission template.
6. Click Save.

Edit a Role

Use the Security Administrator to edit the role settings.

To edit a role

1. In the Security Administrator explorer, select the guest user/role.
2. Edit the permission settings as shown in the following table.

Security Administrator - Edit Roles	Description
Access tab	Displays the Oracle settings of the selected guest user. To change the password, click Edit.
Data tab	Displays the database objects, and their data permission settings. In the grid, use the menu commands. Click the column heading to sort the values. For each database object, select the permission type.
File menu ➤ Open	Loads permission settings that have previously been saved from a (**.SecDef*) file. See also Permission Templates (page 60).
File menu ➤ Save As	Saves the permission settings into a (**.SecDef*) file.
File menu ➤ Save As Permission Template	Saves the permission settings as a permission template that can be used when you create a role. See also Permission Templates (page 60).
File menu ➤ Redirect Data Connection	Redirects the data permission settings to another main document that has the same data structure. Opens the Redirect Data Connection dialog box. For example, if the main document is being replaced by another main document with the same data structure.
Edit menu ➤ Clear Selection	Clears the selected rows.
Edit menu ➤ Select All	Selects all rows.

Edit menu ➤ Enable Advanced Editing	Enables the Advanced Edit mode. In Advanced Edit mode, you can modify the permission settings for all database objects, such as system tables. **WARNING** Changes to system tables can corrupt the database.
Filter	Displays or hides the filter area. To filter database objects by name, or by type.
Save button	Activated after you have made any modifications. Saves the modifications.
Reset	Restores the original settings.

Setting the permission

For each database object, you specify the following permissions: Select, Insert, Update, Delete, Execute, Alter, Direct.

NOTE If you assign one or all of the following permissions to a feature class or a package, the system automatically assigns the Select permission: Insert, Update, Delete, Execute.

Setting Up Topobase Web

With Topobase Administrator, you can provide workspaces, and display models for the Topobase Web version that is based on Autodesk MapGuide®. For complete instructions refer to the Installation and Configuration Guide.

To specify Map Guide credentials

The credentials are required for some Web functions to log in, and to set up the Web layout.

1. Start Topobase Administrator, and open the workspace.
2. Click Setup menu ➤ MapGuide.

3 Enter the credentials as shown in the following table.

Map Guide	Description
MapGuide Site	Specifies the MapGuide site, for example http://127.0.0.1/mapguide2010. **NOTE** You must enter the IP number. Do not use the machine name such as *localhost*.
MapGuide User Name	Specifies the user name to log into MapGuide® Studio. For example Administrator.
MapGuide Password	Specifies the password to log into MapGuide Studio. For example admin.

To set up the Web layout

You select the Web layout elements for the selected workspace.

1 Start Topobase Administrator, and open the workspace.

2 Click Workspace menu ➤ Web Layout.

3 Do one or more of the following:

Web Layout	Description
Web Layout	Select a Web layout. For example Library://Topobase Land/TB2010_LM_B.WebLayout, if you use the AJAX viewer. For example Library://Topobase Land/TB2010_LM_F.ApplicationDefinition, if you use the Fusion viewer.
Web Symbol Library	Select the symbol library, for example Library://Topobase Land/Data/Prototype_LM.SymbolLibrary.
Web Symbol Name	Select the symbol name. For example BAV28. This symbol is used to highlight the position finder search result. See also Position Finder Introduction (page 463).
Web Viewer	Select a Web viewer.

See also the Topobase Installation and Configuration Guide.

Assigning Display Models for Topobase Web

In Topobase Web, you can use several display models to display your map. To provide display models for the use in Topobase Web, you perform the following steps.

- In Topobase Client, you create the display models.
- You publish the display models to MapGuide to create the MapGuide Definitions. If a display model contains several maps, you publish each display model map to a separate Map Definition. You publish each display model map into a different folder.
- In Topobase Administrator, you set up the Web Layout. See Setting Up Topobase Web (page 63).
- In Topobase Client, you assign the display model maps to the Map Guide Definitions.

To publish a display model map to MapGuide

1. Start Topobase Client, and open the workspace.
2. Select the display model, and generate graphic. If the display model contains several display model maps, select the drawing to publish.
3. In Display Manager, click Tools ➤ Publish To MapGuide.
4. In the Publish To MapGuide dialog box, create a new folder, and click Publish.
5. Publish each display model map to a separate folder.
6. In MapGuide Studio, use the Map Editor to assign a coordinate system for each Map Definition. Under Settings, select a coordinate system. If you are unsure which coordinate system to select, locate and select XY-M in the Arbitrary X-Y Coordinate System category.

   IMPORTANT In the Map Definition, the coordinate system must not be empty.

7. In MapGuide Studio, we recommend that you configure the Feature Resource using a connection to the Topobase system user TBSYS.

To publish a Plot display model to MapGuide

Usually, you have a separate Plot display model that is used to display the plot features, such as the frame, north arrow, legend, or scalebar.

1. Open the Plot display model.
2. Publish to MapGuide.
3. Assign the display model for use in Topobase Web.

To assign display models for Topobase Web

Required: The display models have been published to MapGuide, and the Topobase Administrator Web Layout is set up.

1. Start Topobase Client, and open the workspace.
2. Click Home tab ➤ Display Panel.
3. Select the display model, and click Generate Graphic.
4. From the Save Display Model list, select Assign Display Model For Use In Topobase Web.
5. In the Assign Display Model For Use In Topobase Web dialog box, for each map of the display model, assign the MapGuide definition.
6. Click Save.

   To modify the settings, click Edit.

Assign Display Model For Use In Topobase Web	Description
Display Model	Lists the display models that are available in the document. These are all display models that are available in the Display Model select box.
<display model> (nr)	Name of the display model, and number of maps that are defined in the display model.
<map>	Name of the display model map.
MapGuide Definition	Displays the assigned MapGuide Definition. For each map, you assign a layout.

Edit	Edits the assignment.
Save	In Edit mode: Saves the settings.

NOTE Topobase Web uses the Web Layout that you have set up in Topobase Administrator as the initial view.

Configuring the Document Explorer

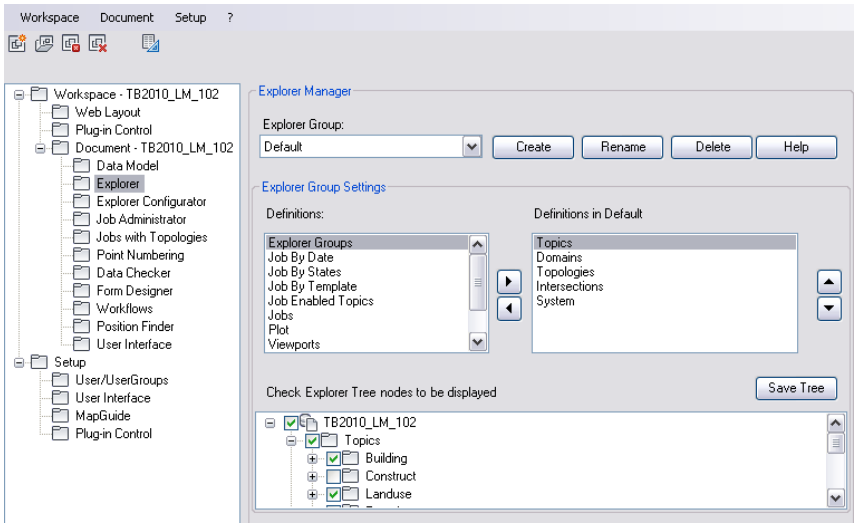
In the Topobase task pane, you use the Topobase document explorer to view and edit data. Use the Topobase administrator to define document explorer groups. You can define which items are displayed in the document explorer and configure document explorers for each document.

To configure a document explorer

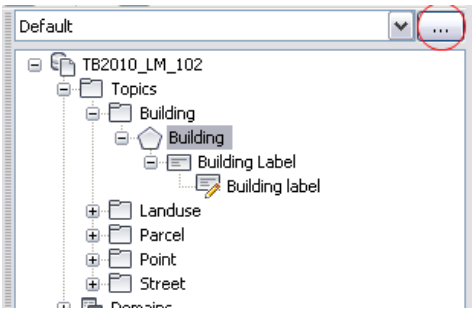
1. Start Topobase Administrator and open a workspace.
2. Click Document menu ➤ Explorer.
3. In the right pane, configure the document explorer content and layout, as shown in the following table:

Explorer Manager	Description
Explorer Group	Specifies the group to edit, for example Default.
Create, Rename, Delete	Creates, removes or renames the explorer group. Click a button.
Explorer Group Settings	Specifies the document explorer items.
Arrow buttons	Adds or removes a selected item to or from the document explorer. Click the arrow.
Definitions in <explorer group name>	Displays the order of the items in the tree view. Use the arrow buttons to change the display order.
Check Explorer Tree Nodes To Be Displayed	Displays the tree, where you can select or clear the nodes. **NOTE** You can also edit this tree in the Topobase task pane by clicking the explorer edit button.

Save Tree	Saves changes, you make in the explorer tree.
Explorer Group Security	Specifies access rights for explorer groups. You can assign explorer groups to user groups. For example, define an explorer group that displays all workflows, and make it unavailable for the VIEWER user group. The settings are stored in the document, in the table TB_EXPLORER_GROUP_UG_RESTR.
User Group With Access To <explorer group>	Displays the user groups that can access the current explorer group.
User Group With No Access To <explorer group>	Displays the user groups that cannot access the current explorer group.



Configuring the Topobase document explorer.



Click the Explorer edit button to edit the document explorer.

Definitions and objects that can be shown in the document explorer are:

Explorer Definition	Description
Topics	Processes feature classes.
Domains	Processes domain tables.
Topologies	Lists the topologies.
Intersections	Starts intersection definitions
System	Lists the Client system tables.
Explorer Groups	Specifies an explorer group.
Job By	Displays jobs ordered by date, state or type.
Job Enabled Topics	Displays job-enabled feature classes.
Jobs	Displays jobs
Plot	Lists the plot feature classes. We recommend that you do not display the plot feature classes in the document explorer.
Viewports	Lists the viewports.
Workflows	Lists the workflows.

Setting Up Electric Explorer Profiles

In Topobase Electric you use the Electric explorer to process the electric features. Use Topobase Administrator to set up profiles that control which feature containers are displayed in the electric explorer.

To set up a profile for the Electric explorer

1. Start Topobase Administrator and open the workspace. Select the document.
2. Click Document menu ➤ Electric Explorer.
3. In the right pane, click New.
4. Under Available Containers, select the containers you want to display.
5. Under Profile Assignment, select the user group, and under Available Profiles, select or clear the profiles.

Electric Explorer	Description
Profiles	Specifies the profiles and containers.
Available Profiles	Displays the profiles of the selected document.
Available Containers	Displays the containers of the Electric explorer, such as Conductor, Devices, Duct.
New	Creates a new profile.
Delete	Deletes the selected profile.
Profile Assignment	Assigns a profile to a user group.
User Group	Selects the user group.
Available Profiles	Displays the profiles. that are assigned to the selected user group.

NOTE Use the Electric document options to specify the attributes that are displayed in each container. See Setting Electric Document Options (page 28).

For more information about Topobase Electric, see the Topobase Electric User Guide.

Topologies

Topobase topologies are described in a separate chapter.

See also:

- Topology Introduction (page 255)

Job Management

Based on the Oracle database, Autodesk provides a long transaction and versioning framework.

See also Jobs Introduction (page 295).

Topobase System Tables

Topobase stores system information in client and server system tables.

The client system tables are stored in the Topobase document. They store document-specific settings.

WARNING Do not edit the Topobase system tables manually. If you do this, your application might not work properly any longer.

The server system tables are stored in TBSYS (page 524) (Topobase database server). They store settings that are valid for the whole application, such as application menu settings.

The system tables must not be edited manually, however they contain information that can help administrators in case of trouble shooting, as shown in the following table:

Table	Description
TB_ATTRIBUTE	Stores attribute names for translation.
TB_FORM_LOCATION	Describes size and location of Topobase.Forms.Form (non generic dialog boxes) for Web and desktop version.

TB_DATAMODELCODE	See System Tables: Data Model Code (page 73).
TB_DATAMODELCODES	Stored in TBSYS. See System Tables: Data Model Code (page 73).
TB_DIALOG_USER	Stored in TBSYS. Defines the default mode for every user.
TB_DIALOG	Stored in TBSYS. Stores the default value for DefaultMode.
TB_DICTIONARY	Stores the description of the feature class tables. See also System Table TB_DICTIONARY (page 74).
TB_DIMENSION	Stores the dimensioning configuration. Dimensioning consists of a parent dimensioning feature class, a point feature class for symbols, a line feature class, a label feature class, and a child label line feature class. See also Data Model: Dimensioning (page 205)
TB_INFO	Stores project settings.
TB_INTERSECTION	See System Table TB_INTERSECTION (page 79).
TB_JOB_*	Manages jobs. See also System Tables TB_JOB_* (page 81).
TB_LABEL_DEF	See System Table TB_LABEL_DEF (page 88)
TB_NET_*	See System Table TB_NET_* (page 89).
TB_POSITION_FINDER	Defines searches. See also System Table TB_POSITION_FINDER (page 92).
TB_RELATIONS	See System Table TB_RELATIONS (page 94).
TB_RULE_*	Stores feature rules. See also System Tables TB_RULE* (page 98).
TB_SURFACE_*	Stores the configuration for the Polygon And Line Definition report. For configuration, see also Report Designer: Polygon and Line Definition Report (page 455).
TB_TOPOLOGY	See System Table TB_TOPOLOGY* (page 101).
TB_WORKFLOW	See System Table TB_WORKFLOW (page 102).

TB_VARIABLE	Stores user defined variables that can be used for stylization. See Topobase Text Function TB_VARIABLE (page 410).
TB_GN_*	System tables for toolbar and menu settings.
TB_GN_CONTROL	Stores properties of all controls used in the generic forms of the document.
TB_GN_DIALOG	Specifies the default position/size of forms (tables); for example the height of the detail of a master-detail form
TB_GN_DIALOG_BAR_ITEM	Stores settings for form menus and tool bars Example: If the form toolbar contains a highlight button, you will find a row with the following value in TB_TB_GN_DIALOG_BAR_ITEM.NAME: Topobase.Map.MapDialog/TBHighlightBounded
TB_GN_DIALOG_HIGHLIGHT	Stores configuration of the Highlight Related Features command. See also System Table TB_GN_DIALOG_HIGHLIGHT (page 75).
TB_GN_DIALOG_MACHINE_USER	Stores user/machine dependent position/size.
TB_GN_DOCUMENT_BAR_ITEM	Stores settings for document toolbar buttons.
TB_GN_DIRECTORY_ALIAS	Stores file names for Topobase Web. See also System Table TB_GN_DIRECTORY_ALIAS (page 75).
TB_GN_INFO_REDIRECT	Stores redirection information. See also System Table TB_GN_INFO_REDIRECT (page 77).
TB_GN_MENU_ITEM	Stores settings for application menu items. Stored in TBSYS.
TB_GN_MENU_ITEM	Stores settings for document menu items.
TB_GN_TOOLBAR_ITEM	Stores settings for application toolbar buttons. Stored in TBSYS.

System Tables: Data Model Code

Topobase has several system tables that store the data model code.

TB_DATAMODELCODES

TBSYS.TB_DATAMODELCODES contains information about which data model codes currently exist, which means all Topobase data model codes.

Attributes of TB_DATAMODEL-CODES	Description
ID	Stores the unique key, which is system generated.
FIRM	Stores the company code.
CODE	Stores the data model code. 0 = undefined.
NAME	Stores the name.
SUB CODE	Stores a sub code for variations on a data model, for example, county specific distinction.

TB_DATAMODELCODE

The TB_DATAMODELCODE table resides in the Topobase document (page 521) and contains the users valid data model codes. A document can have several data model codes, for example, if the document contains both a survey and profile structure.

In Topobase Client, you can view the data model codes of the current document.

See also:

- Common Document Options (page 25)
- Data Model Templates (page 36)

System Table TB_DICTIONARY

This table stores the description of the feature class tables.

Attributes of TB_DICTIONARY	Description
F_CLASS_ID	Primary key.
F_CLASS_NAME	Name of the feature class.

CAPTION	Caption of the feature class. This caption is displayed on the UI, for example in the document explorer.
F_CLASS_TYPE	Feature class type. O = polygon, P = point, L = line string, T = attribute, A = label, E = centroid.
READ_ONLY	Specifies whether the feature class is read only.
TABLE_DOES_NOT_EXIST	Specifies whether this feature class is a view. 1 = This feature class is a view. The default is 0 = This feature class is a table of the current data base schema.
FEATURE_REPRESENTATION	Specifies the string that is displayed in the feature explorer. Default is {FID}. See also Customizing the Feature Explorer (page 165).

System Table TB_GN_DIALOG_HIGHLIGHT

This table specifies linked features to highlight. For example, if you highlight a manhole, the appropriate manhole cover is also highlighted. The highlight button is only available on the form toolbar if there are appropriate settings in the system table TB_GN_DIALOG_HIGHLIGHT.

Example

Attribute	Value
HIGHLIGHT_FEATURECLASS	MYPOINT
HIGHLIGHT_FILTER	fid={parent_fid} Use any SQL expression in the filter.

System Table TB_GN_DIRECTORY_ALIAS

This table specifies the links to file or picture folder names for Topobase Web version. It is used for form designer and report designer controls that specify a file name or a picture file. The first part of the file name will be replaced using this table.

NOTE If your pictures are stored in the <topobase>\Pics folder or a subfolder, you need not to specify an alias in TB_GN_DIRECTORY_ALIAS.

Control properties that use file/folder names:

- Text box
- Picture box
- Picture combo box
- Static picture

Example with filename "c:\dat\mydocuments\house55.bmp"

1. Create an alias in the IIS for "c:\dat\mydocuments\". For example, "MyPictures")
2. Create a new record in TB_GN_DIRECTORY_ALIAS and set

   Directoryname="c:\dat\mydocuments\"

   AliasName= "http://mycompany.com/MyPictures/"

All Web file names that begin with "c:\dat\mydocuments\" are preceded by "http://mycompany.com/MyPictures/".

There are two ways to specify the alias name. You can either enter a complete URL or a relative address. For example, when you specify a complete URL, such as *http://mycompany.com/MyPictures*, the alias name is the absolute path. When you specify an alias without the HTTP part, such as *MyPictures/*, the alias name is related to the IIS alias TBWEB.

Also, you can create a virtual directory that is not located under MapGuide and specify the complete URL as alias.

NOTE Restart Topobase after making new entries or changing older entries.

See also:

- Form Designer: Property TextBox - Properties (page 349)
- Form Designer: Picture Linker - Properties (page 347)
- Form Designer: Picture Combobox - Properties (page 346)
- Form Designer: Static Picturebox - Properties (page 355)
- Report Designer: Picture Control (page 442)

System Table TB_GN_INFO_REDIRECT

The following table defines redirection information:

Attribute of TB_GN_INFO_REDIRECT	Description
ID	Stores the unique key, which is system-generated.
F_TABLE_NAME	Specifies the feature class that is selected in the map.
DIALOG_TABLE_NAME	Specifies the name of the form that is opened instead of the feature class form.
F_COLUMN_NAME	Specifies an attribute of the feature that is selected in the map.
DIALOG_COLUMN_NAME	Specifies an attribute of the form to be opened that is related to F_COLUMN_NAME.
INFO_FILTER	Specifies a complex filter, for example, if you have relations between more than one table.
DIALOG_DOCUMENT_NAME	Opens a form in another document. If the document is not found in the currently open workspace, the system searches all open documents and opens the first form it finds.

In the following cases, a redirection definition in the TB_GN_INFO_REDIRECT table is not necessary:

- For labels and their parent feature class. Because redirection for labels is common, you can activate label redirection by using a document option.
- The utility application has a default redirection to show the utility objects when you select a network geometry.

Example 1: To click a manhole but show the section form, set the following values:

Attribute	Value

F_TABLE_NAME	MANHOLE
F_COLUMN_NAME	FID_SECTION
DIALOG_TABLE_NAME	SECTION
DIALOG_COLUMN_NAME	FID

Example 2 shows the use of INFO_FILTER:

Attribute	Value
F_TABLE_NAME	Specify the name of the feature class that is selected in the map.
F_COLUMN_NAME	(leave this blank)
DIALOG_TABLE_NAME	Specifies the name of the form to open instead of the feature class form
DIALOG_COLUMN_NAME	(leave this blank)
INFO_FILTER	FID in (select FID_PARENT from PARCEL_TBL where {Fids})

The word {Fids} is replaced with the FIDs of the graphic. The whole expression is set as a filter on the form to open.

See also:

- Utility Model Introduction (page 285)
- Setting Map Options (page 17)

System Table TB_INFO

The following table stores system information. The table resides in the document.

Attributes of TB_INFO	Description
ID	Unique key

PARAM	Parameter.
VALUE_CHAR	Value, if the parameter is of type CHAR.
VALUE	Value, if the parameter is numeric.
COMMENTARY	Commentary

ID	**Some important parameters are the following:**
1001	Version of Topobase structure.
1002	Name of the Topobase System User (TBSYS database schema)

System Table TB_INTERSECTION

The following table stores intersection definitions.

NOTE The properties are set with the data model administrator.

Attributes of TB_INTERSECTION	**Description**
ID	Stores the unique key, which is system-generated.
NAME	Specifies the name of the intersection.
TYPE	OO = Polygon/Polygon => Polygon OL = Polygon/Line => Line OP = Polygon/Point => Point LL = Line/Line => Point LP = Line/Point => Point
ADJUST_SECOND	True/false: Adjusts the area of the second feature class to the area of the first feature class. Only useful for polygon/polygon intersections.
ADJUST_EXCLUDE	Specifies the condition that excludes features from an adjustment. Example: s.area <100.

BUFFER_PRIM_FC	Defines a perimeter for a buffer. All features inside the perimeter are returned.
F_CF_CLASS_NAME_PRIM	Primary feature class.
F_CLASS_NAME_SEC	Secondary feature class.
F_CLASS_NAME_INTER	Specifies the name for intersection feature class. If the table does not exist, it is created automatically. The name must not contain blanks.
PRIM_JOBID SEC_JOBID	Specifies the job used to get data from DB. If "JOBID= 0", the currently selected job is used.
PRIM_UPDATE_STATE-MENT SEC_UPDATE_STATEMENT	Selects values from other tables to store in the intersection feature class. These values can be used for reports. This method is faster than creating a view that joins information from other tables and from the intersection results.
STORE_GEOMETRY	Stores "AREA" and "AREA_NOMINAL" on OO intersections (polygon/polygon). Set to true or false.
TOL_AREA_NOMINAL	Stores the maximum value of the difference between the area of the primary and summary of the secondary feature class where no adjustment is started.
TOLERANCE	Sets the spatial tolerance for the intersection. This tolerance is equal to the smallest tolerance of primary and secondary feature class.
WHERE_CLAUSE_SEC WHERE_CLAUSE_PRIM	Restricts features participating in the intersection. The restriction can be defined as a WHERE Clause.

TB_STATE_INTERSECTION defines the state in which the intersection applies. If an intersection is called with the same primary and secondary feature classes, it must be executed before the merge job. All other intersections can be called after the merge.

See also:

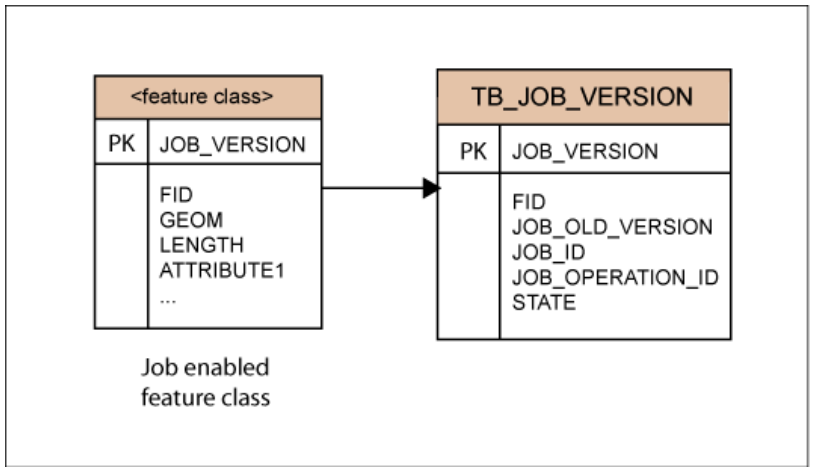
- Creating an Intersection (page 226)

System Tables TB_JOB_*

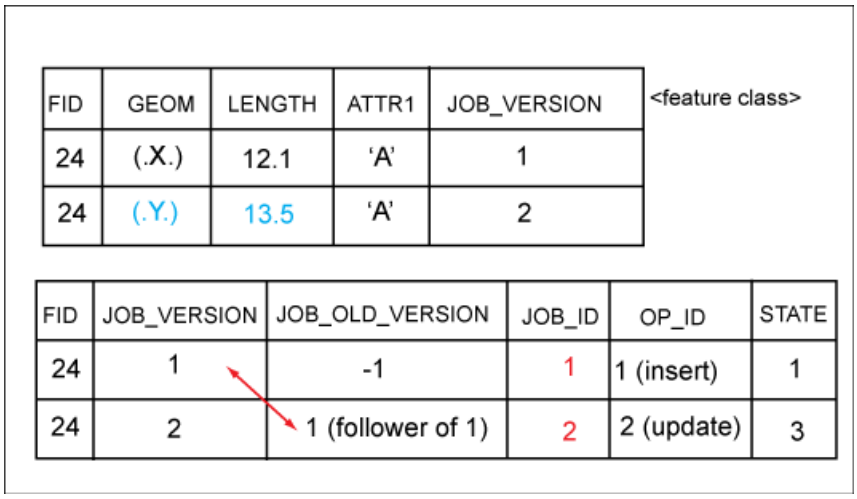
WARNING Do not edit Topobase Job system tables manually. This causes data loss and can corrupt jobs. Values are updated using triggers. For example, if you made modifications using SQL, manual editing is not necessary.

System Table TB_JOB_VERSION

The table TB_JOB_VERSION stores the complete history of every feature. The feature table has one job relevant attribute: JOB_VERSION. The table TB_JOB_VERSION also controls which features are visible.



TB_JOB_VERSION holds the history of every feature and controls which features are visible.



With JOB_VERSION the FID is not unique, however with the Virtual Private Database it is.

Example: A "SELECT * from <feature class> returns the following values:

With Job 1: 24, (.X.), 12.1, 'A', 1

With Job 2: 24, (.Y.), 13.5, 'A', 2

The JOB_OPERATION_ID shows the operation performed on the feature, for example, Insert, Modify, or Delete.

Attributes of TB_JOB_VERSION	Description
CONFLICT	
JOB_VERSION	Specifies the version of an object (feature). Mandatory.
JOB_OLD_VERSION	Specifies the previous version of an object. Optional.
JOB_ID	Specifies the job ID. Mandatory.
JOB_OPERATION_ID	Specifies the job operation ID. Mandatory. 1 = INSERT,

	2 = UPDATE, 3 = DELETE - 1 = Indicates that this feature has already been existing, at the moment of the job enabling of the feature class.
OS_USER_NAME	Stores the OS user.
OPERATION_DATE	Stores the operation date.
STATE	Defines the state of the feature, either: 1 = live 2 = pending 3 = open 4 = deleted (additional states can be defined in TB_JOB_STATE)

After a table has been job enabled, a new value for JOB_VERSION is created for every object (sequence TB_JOB_VERSION_S):

- JOB_OLD_VERSION is set to -1.
- JOB_ID will is 1 = live, for every feature.
- JOB_OPERATION_ID is -1 for every feature.
- STATE is 1 = live, for every feature.

If a user works on a feature within a certain job (JOB_ID) and for example modifies a feature, the operation is registered in TB_JOB_VERSION as follows:

- JOB_OLD_VERSION is set to the number, the feature had before the modification.
- JOB_ID is < current Job ID.
- JOB_OPERATION_ID is 2 = "modification".
- STATE is 3 = open.

You can trace the history of every feature, including the operation performed, the date, user, and initial appearance.

TB_JOB_VERSION.JOB_OPERATION_ID can be used in Display Manager to highlight modified or newly created features, so they can be distinguished

from life features. Also, you can define a special style for the deleted features. See Styling Job-Enabled Features (page 412).

System Table TB_JOB_STATE

The system table TB_JOB_STATE stores the job state configuration. To define additional job states do this at the very beginning of all job configuration and data capture. After working with jobs respective job-enabled features do not add any new job states.

Attributes for TB_JOB_STATE	Description
ID	Stores the unique key, which is system-generated.
NAME	Specifies the job name.
INITIAL_STATE	Indicates whether the state is the initial state for each job. 0= No. 1 = Yes. **NOTE** There can only be one initial state.
FEATURES_EDITABLE	Indicates whether the features can be modified using insert, update, or delete.
USE_JOBID	Specifies the job state whose features are displayed additionally to the features of the current job state. For example, this might specify that all live features are displayed, even if they do not belong to the current job state.

The following table shows the default job states:

ID	Name	Initial state	Editable	Use Job ID
1	Live	No	No	No
2	Pending	No	No	No
3	Open	Yes	Yes	1 (live)
4	Deleted	No	No	No

System Table TB_JOB_STATE_TRANSITION

The system table TB_JOB_STATE_TRANSITION stores the job state transition rules. You can define additional transition rules. Besides, you must do this at the beginning of all job configuration and data capture.

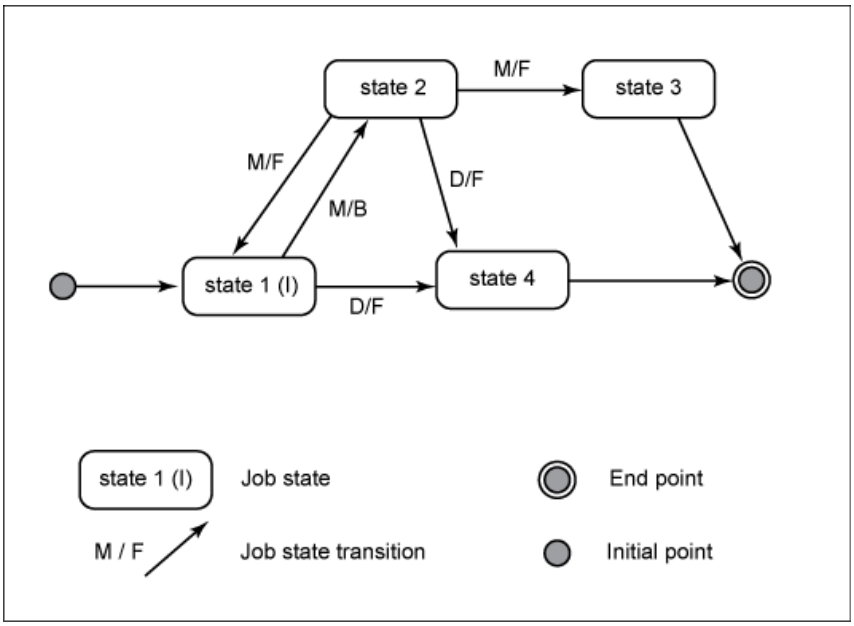
Attributes of TB_JOB_STATE_TRANSITION	Description
PREDECESSOR_ID	Stores the foreign key to the previous state (TB_JOB_STATE) in the state diagram.
SUCCESSOR_ID	Stores the foreign key to the next state (TB_JOB_STATE) in the state diagram.
TRANSITION_TYPE	Specifies the value indicating what the software must do when the state transition is executed. ■ M—Merge the features. ■ D—Delete the job. ■ N—Do nothing.
TRANSITION_DIRECTION	Specifies the value indicating whether the transition is made forward (F) or backward (B). Example: A forward transition could be a merge process (open -> pending). A backward transition could be an undo of a merge process (pending -> open).

NOTE A transition type with D for delete can never be backward.

The following table shows the default job state transition rules:

ID	Predecessor	Successor	Direction	Type
1	Open (3)	Pending (2)	Forward (F)	Merge (M)
2	Pending (2)	Open (3)	Backward (B)	Merge (M)

3	Pending (2)	Live (1)	Forward (F)	Merge (M)
4	Open (3)	Deleted (4)	Forward (F)	Delete (D)
5	Pending (2)	Deleted (4)	Forward (F)	Delete (D)



Job states and job state transitions. The direction of a transition can be forward F or backward B. M merges the features.

The initial point can only have one initial state as successor. There can only be one initial state.

System Table TB_JOB

The system table TB_JOB contains the list of all jobs.

WARNING There are two default (virtual) jobs in every database: Pending Job and Live Job. They are necessary for job management, even if you do not work with Topobase jobs. Do not delete these jobs, as this will cause malfunctions.

Attributes of TB_JOB	Description

USE_STATE	For internal use only. Defines the job state for live and pending virtual jobs.

System Table TB_JOB_PERIMETER

The system table TB_JOB_PERIMETER administers the job perimeters.

Attributes of TB_JOB_PERIMETER	Description
JOB_ID	Specifies the job the perimeter belongs to (TB_JOB).
JOB_PERIMETER_TYPE_ID	Specifies the perimeter type. ■ **Type 1**—Stores the perimeter that has been digitized or selected. For type 1 perimeters no F_CLASS_ID is stored. ■ **Type 2**—For each perimeter feature class that is used in the job template, one type = 2 perimeter is created.
F_CLASS_ID	Specifies the ID of the perimeter feature class.
TOPO_ID	Specifies the ID (TB_TOPOLOGY.ID) of the topology that builds the geometry of the perimeter. For JOB_PERIMETER_TYPE_ID=1 perimeters no TOPO_ID is stored.

Example: The job template contains the topics Parcel and Land Cover. Each topic is part of a topology and has its own perimeter feature class. In a job, when you select the perimeter using the parcel topology, an intersection with the land cover topology is performed and for each perimeter feature class a record with JOB_PERIMETER_TYPE_ID=2 and the TOPO_ID is added.

System Table TB_LABEL

The system table TB_LABEL contains the IDs of feature classes and their related label feature classes. This table lists all feature classes that have a label feature class. The IDs refer to TB_DICTIONARY.F_CLASS_ID.

Attributes of TB_LABEL	Description
ID	Stores the unique key, which is system-generated.
TBL_F_CLASS_ID	Specifies the ID of the label feature class table (child). This table contains the attribute data.
PARENT_F_CLASS_ID	Specifies the ID of the feature class table (parent). This table contains the text.

System Table TB_LABEL_DEF

The table TB_LABEL_DEF contains SQL Select statements that return text when placing or updating labels.

NOTE Do not edit this table manually. Use the Topobase data model administrator (Label Properties).

Attributes of TB_LABEL_DEF	Description
ID	Stores the label definition ID. User defined label definitions start with ID 10.000. IDs < 10.000 are used for Autodesk applications.
LABEL_ID	Stores the label ID. All label definitions of one feature class are grouped by this label_ID. For example, if you have defined four labels for a certain label feature class, those for definitions will all have the same label_ID.
NAME	Specifies the label name.
SELECT_STATEMENT	Stores the label select statement. See alsoLabel Select Statements (page 200).
EXAMPLE	Specifies a sample label.
Position_along_object	Defines the position of the label. 1 = Start; 2 = Middle; 3 = End. Default is Middle.

Absolute_offset_*	Specifies the absolute text coordinates offset.
Relative_offset_*	Specifies the relative text coordinates offset.
ORIENTATION	Stores the text orientation.
IS_RELATIVE_ORIENTATION	Specifies whether the text orientation is absolute or relative. Either 0 = absolute or 1 = relative.
Autolabel	Specifies whether the label is created when the feature was created. Either 1 = Yes or 0 = No.
User_selects_position	Specifies whether the user selects the label position. Either Yes or No.
Overwrite_label_text_with_null	Specifies whether the label text will be NULL if the select statement returns NULL. Either 1 = Yes or 0 = No.
Scale_select_statement	Specifies a select statement to query the scale.
Pre	Specifies a prefix for label text.
Suf	Specifies a suffix for label text, for example a unit of measurement could be used.
Select_statement_geometry	Specifies a select statement to find the coordinates of a parent feature.
Active	Specifies whether the label is active. Either 1 = Yes or 0 = No.

System Table TB_NET_*

The system tables TB_NET_* reside in the document. They store the network tracing system information.

System Table TB_NET_TRACE_TPL

This table stores the network tracing definitions.

See also:

- Logical Topology: Tracing Templates (page 272)

Attributes of TB_NET_TRACE_TPL	Description
ID	Primary key
ACTION	Specifies the actions to be applied on the resulting features: 0 = no action 1 = highlight 2 = show results 3 = highlight and show result
ACTIVE	Specifies that the template can be selected by the user.
COMMENTARY	Commentary
DIRECTION	Specifies the direction of the trace: 1 = up; forward 2 = down; backward 3 = both directions
NAME	Specifies the name, that is displayed in the network explorer.
TRACE_TYPE	Specifies the type of network trace: 1 = shortest path 2 = minimum cost spanning tree 3 = reachability 4 = within cost analysis (not yet implemented) 5 = nearest neighbors analysis (not yet implemented) 6 = all paths between two nodes (not yet implemented) 7 = traveling salesman problem analysis (not yet implemented)

TOPOLOGY_ID	Specifies the topology for which the tracing template has been defined.

System Table TB_NET_SCOND_DEF

This table stores the search conditions for network analysis. The table resides in the document.

See also:

- Defining Tracing Conditions (page 275)

Attributes of TB_NET_SCOND_DEF	Description
ID	Primary key
ACTIVE	Specifies, that the condition can be selected by the user. Either 0 = deactivated or 1 = active.
COMMENTARY	Commentary.
IS_COST	Specifies the type of the condition: 0 = stop condition 1 = cost calculation algorithm
NAME	Specifies the name of the condition.
SQL_STMT	Specifies a select statement.
TOPOLOGY_ID	Identifies the topology for which the condition has been defined.
TOPO_TYPE	Specifies to which component of the topology the condition applies: 1 = nodes, 2 = edges.

System Table TB_POSITION_FINDER

The system table TB_POSITION_FINDER resides in the document.

Attributes of TB_POSITION_FINDER	Description
Base attributes	
NAME	Specifies the name or header of the search that will be displayed in the Position Finder group box.
GUITYPE	Specifies the search type: 1 = sequential (Default) 2 = flat 3 = PlugIn
PRIORITY	Defines the order of the entries. Default = 0.
Sequential search	The following four attributes are needed to define a sequential search.
STMT_TITLE_*	Stores the title and short explanation that will be returned with the n th query.
SELECT_STMT_*	Stores the search query. This is a standard SQL statement. **NOTE** The first search query has no placeholders.
GEOM_STMT_*	Stores the geometry query. Always optional, except when it is the last query. `select geom from <table> where fid = {x}.`
INPUTTYPE_*	Stores the input type for the results in the n th query: 1 =select list (Default); 2 = text box 3 = text box with auto complete

NOT_FOUND_MSG_*	Defines an optional message to be used as a prompt if the search query returns no rows.
Flat Search	The following attributes are needed for flat search: With these attributes, the select statement is composed as follows: `select <pk_column>, <input_colums>from <table_name>where upper(<input_column>) like upper(<param>)order by <order_column>`
TABLE_NAME	Stores the name of table or view containing the data.
PK_COLUMN	Stores the name of the attribute containing the primary key. This is the key attribute.
GEOM_COLUMN	Specifies the attribute containing the geometries. This is the geometry attribute.
ORDER_COLUMN	Specifies the order of the query results. May contain more than one attribute (comma separated). Optional.
INPUT_COLUMN_*	Stores the input attributes (1-5) that will be used for the WHERE Clauses.
STMT_TITLE_*	Stores a short explanation of the corresponding input column_* of the data table.
NOT_FOUND_MSG_1	Stores the message to be used as a prompt if the search query returns no rows. All other 'NotFound Messages' (2-4) will be ignored.
Assembly Name Namespace Class Name	Stores the properties used in a Plug-In search.

See also:

- Position Finder Introduction (page 463)

System Table TB_RELATIONS

The system table TB_RELATIONS table stores relations between feature classes. For example, you can define relations when you add an attribute to a feature class, using the Data model administrator.

Attributes of TB_RELATIONS	Description
ID	Stores the unique key, which is system-generated.
PARENT_TABLE_NAME	Stores the name of the parent table, for example a domain table *_TDB. Domain tables usually are parent tables.
PARENT_COLUMN_NAME	Stores the parent attribute name. Example: ■ FID, if the parent is a feature class table. ■ ID, if the parent is a domain table.
CHILD_TABLE_NAME	Stores the name of the child table, for example a label feature class table.
CHILD_COLUMN_NAME	Stores the name of the child attribute. Example: ■ FID_xxx, if the child is a feature class table. ■ ID_xxx, if the child is a domain table.
DELETE_CHILD	Specifies whether the child feature is deleted when the parent feature is deleted. N—Set Relation to Parent to NULL when Parent is deleted. Default. L—Do not alter child when parent is deleted. D—Delete Feature when Parent is Deleted. C—Don't delete Feature when a parent exists. P—Don't delete Feature when a child exists.

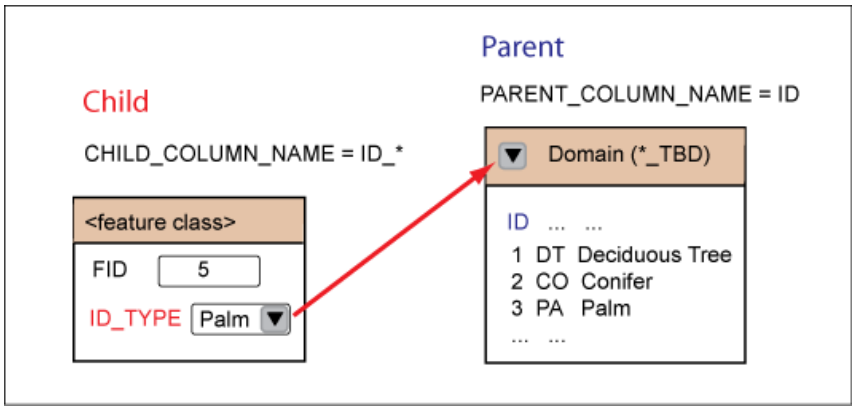
CREATE_CHILD	Specifies whether a child feature is created. Either 1 = Yes or 0 = No. Default = 0.
ACTIVE	Specifies that the record is active. Either 1 = Yes or 0 = No. Default = 1.

TB_RELATIONS: Parent, Child, Sibling

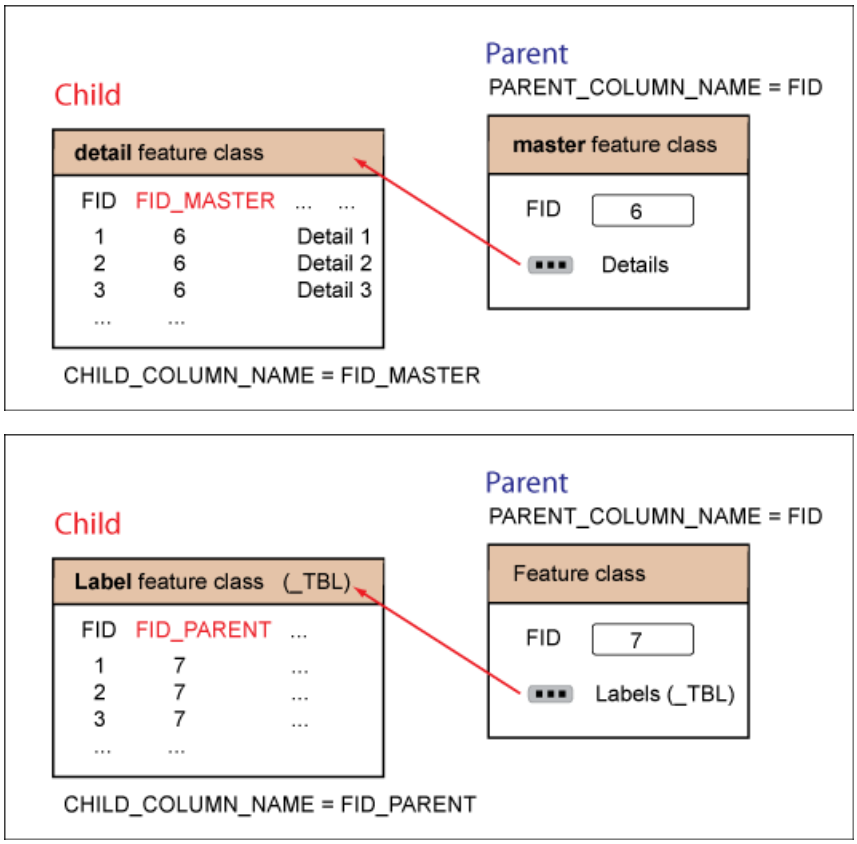
Topobase data models frequently use two types of relations:

- relation between a feature table and a domain table.
- relation between two feature tables, one of which is a master table and one is a detail table.

The following illustration shows the relation between a feature table and a domain table:



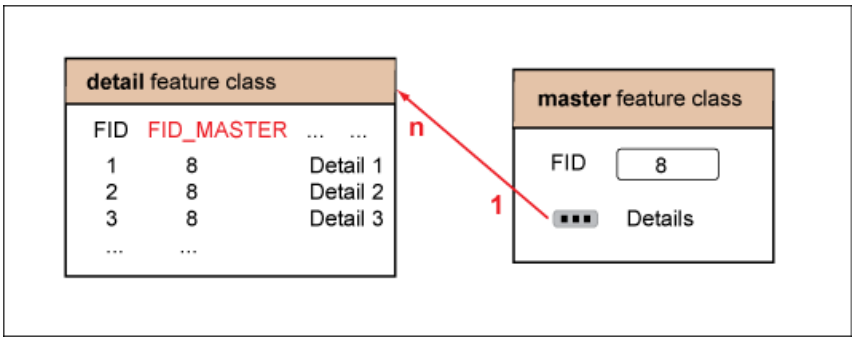
The following illustration shows the relation between a master table and a detail table:



Topobase Data Model: The most frequent master-detail relation is —between a feature class and a label feature class.

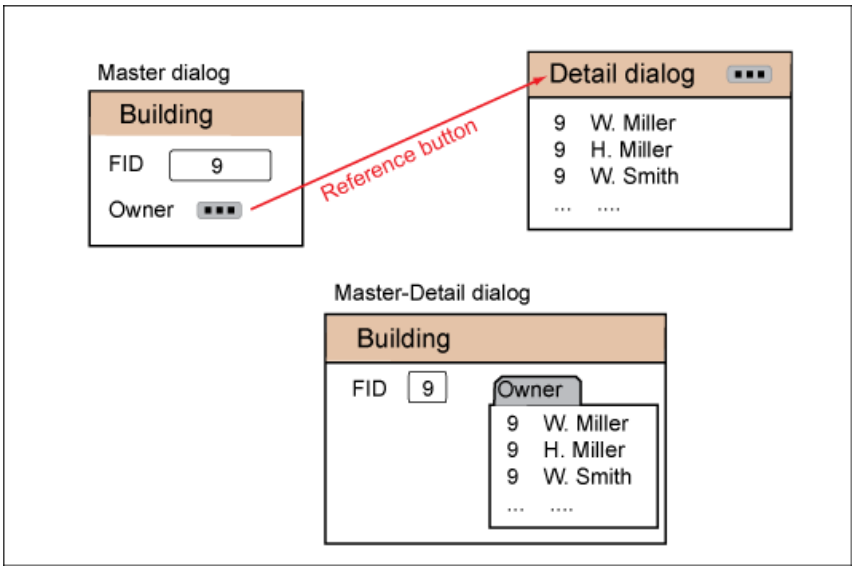
TB_RELATIONS: Master-Detail Relations

In a Master-Detail relation, one feature class can have multiple relations to another feature class. For example, a master feature, such as a building can have several detail features. Similarly, each feature can have several labels.



Topobase Data Model: Master-Detail relation

In Topobase Client feature class forms, you can either show the detail features in a separate tab or you can display a reference button to open the related form and all related features in the filter. You configure this in Topobase Form Designer.



Topobase Data Model: Display of related features in feature class forms

See also:

- Designer Menu: Master-Detail (page 317)

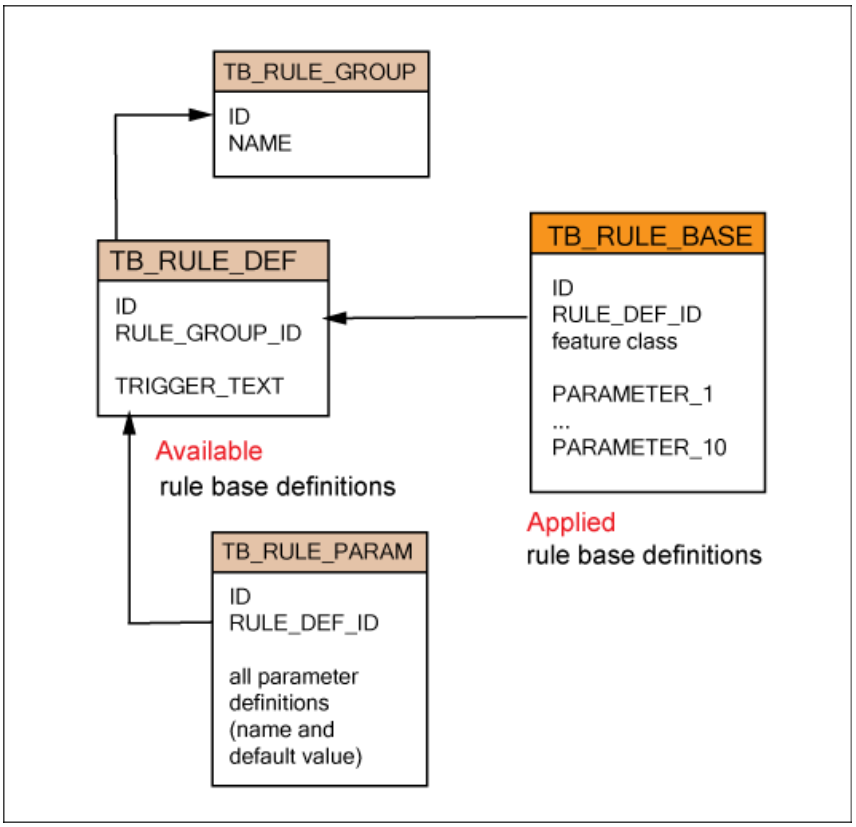
System Tables TB_RULE*

The system tables TB_RULE* store the Topobase feature rules.

See also:

- About Feature Rules (page 511)

Table	Description
TB_RULE_PARAM	Stores feature rule parameters.
TB_RULE_DEF	Stores all rule base definitions. See also System Table TB_RULE_DEF (page 100).
TB_RULE_GROUP	Lists rule base groups. Feature rules can be grouped.
TB_RULE_FCLASSTYPE	Specifies which feature rules are applied to a feature class when it is created (L = for line string, P = for point, T = for label feature classes). If F_CLASS_TYPE is empty, the rule is assigned to all feature classes.
TB_RULE_BASE	Stores the configuration of individual rules for each feature class (applied rule base definitions). **NOTE** Only rules that are listed here can be activated.
TB_TMP_FEATURE TB_RMP_FEATURE_TLB	Stores temporary data for server-side feature rules and topology or compound feature classes.



Topobase feature rule system tables

System Table TB_RULE_PARAM

This table stores the parameters of the feature rule definitions. Only the stored parameters are available in the Edit Rule Base dialog box, where you can assign values.

Attributes of TB_RULE_PARAM	Description
ID	Stores the primary key.
RULE_DEF_ID	Stores the ID of the rule definition in TB_RULE_DEF.

NAME	Specifies the name.
DEFAULT_VALUE	Specifies a default value.
POSITION	Stores an index (1 - 10). The parameters can be used in the trigger text using the alias %P<index>, for example %P1 or %P2.

System Table TB_RULE_DEF

This table stores the feature rule definitions.

For server-side feature rules, the definition includes the actual program. The rules can be identified by their IDs.

For client-side feature rules, the definition includes the rule name, class name and assembly name, as shown in the following table:

Attributes of TB_RULE_DEF	Description
ID	Stores the primary key.
NAME	Specifies a short name for the rule.
CLASSNAME	Specifies the name of the class where the method is defined. For client-side feature rules.
ASSEMBLY	Specifies the name of the assembly where the method is defined. For client-side feature rules. **NOTE** For server-side feature rules this attribute must be NULL.
TRIGGER_TEXT	Specifies the PL/SQL block forming the body of the rule. For server-side feature rules. **NOTE** For client-side feature rules this attribute must be NULL.
IS_ROW	Specifies the trigger type: Either 1 = row trigger or 0 = statement trigger.

IS_SYSTEM	Specifies the trigger property: Either 1 = rule body is immutable or 0 = rule body can be edited.
IS_DEFAULT	Specifies when the rule is activated. 1 = The rule is a default rule that is activated when a matching feature class is created. 0 = The rule is not activated by default.
DEFAULT_PRIORITY	Specifies the default priority for this rule.
REQUIRES_ID	Specifies whether chaining is allowed. 1 = Allows chaining of dependent triggers, for example, when a row trigger requires a statement trigger. For server-side feature rules.
EXECUTE_COLUMN	Specifies when a trigger is executed. 0 = trigger is always executed. Not NULL = The trigger only executes when an attribute matching the specified name is modified.
RULE_GROUP_ID	Specifies whether the rule belongs to a group. Is NULL for standalone rules.
RULE_VERSION	Specifies the version of the rule in the format. For client-side feature rules.
DESCRIPTION	Concise description of the rule.
BI, BU, BD, AI, AU, AD	These fields map to an event (before/after insert/update/delete). A value of 1 means that the rule can react on that kind of event.

System Table TB_TOPOLOGY*

Topologies require the following some system tables, which reside in the document:

TB_TOPOLOGY—Lists the available topologies.

TB_TOPOLOGY_DEF—Stores the relationship to participant feature classes.

System Table TB_WORKFLOW

The system table TB_WORKFLOW stores the workflow definitions. It resides in the document.

Attributes of TB_WORKFLOW	Description
ID	Stores the unique key, which is system-generated.
IS_SYSTEM	Sets values related to the definition in the Workflow administrator. Set this value = 1 to write protect the definition. this case only the caption can be modified.
CAPTION	Specifies the unique name that is displayed in the document explorer (page 521).
NAME	Specifies the name of the workflow
PARENT_ID	Stores the TB_WORKFLOW.ID of the root workflow. This value is used for the sort order in the workflow explorer tree.
PICTURE	Stores the name of the icon in the workflow explorer. The pictures must be saved in the <topobase>\pic folder or a sub folder.
PRIORITY	Controls the sort order of the node items in the explorer tree.
SCRIPTCODE	Stores the VB .NET script code (up to 4000 characters).

See also:

- Workflow Introduction (page 481)

Topobase Advanced Topics

Explorer Configurator

Application developers can customize the Topobase document explorer groups for each document, using the Explorer Configurator.

To create explorer statements

1. Start Topobase Administrator and open a workspace.
2. In the Administrator window, click Document menu ➤ Explorer Configurator.

Create new definitions and statements that are displayed dynamically in the Topobase document explorer, in the same way the default topics, feature classes, and so on are displayed.

On the Statements tab, under Select Statement, us the SQL Assistant to create the select statements.

The review example demonstrates some settings that can be made

1. In the Explorer Configurator, expand the Workflows item.
2. Select Workflows and click the Statements tab.

Notice the explorer icon definition, the node item captions, and the SQL statements.

See also:

- Creating SQL Expressions (page 103)
- Select Statement Validation (page 109)

For more information see the Topobase Developer's Guide.

Creating SQL Expressions

Topobase Administrator contains an SQL assistant to simplify the creation of SQL select statements. In all dialog boxes, where you can enter an SQL statement, you can use the SQL assistant, for example when you create a label

definition, or when you customize feature class forms. Depending on the condition in which you start the SQL assistant, specific information, such as valid parameters, is provided.

IMPORTANT To use the SQL assistant you have to be familiar with Oracle, SQL, and the Topobase database schema. This knowledge is not subject of this manual.

You use the SQL assistant in the following components and tasks.

- Report Designer, see also Report Designer Introduction (page 425).
- Form Designer, see also Form Designer Introduction (page 309).
- Defining tracing conditions, see also Logical Topology: Tracing Templates (page 272).
- Position Finder, see also Position Finder Introduction (page 463).
- Explorer Configurator, see Explorer Configurator (page 103).
- Creating label definitions, and defining rules for label regeneration. See also Data Model: Labels (page 187)
- Defining Intersections, see also Data Model: Intersections (page 224).

NOTE The SQL Assistant is specific to the condition in which it is started. For general purposes, use SQL Sheet. SQL Sheet is the standalone version of the SQL assistant. See also Using the SQL Sheet (page 130).

The SQL assistant facilitates the entry of table names, attribute names and relations. It provides context specific information about the database schema that you are working with. For example, you can list tables, views, feature class attributes, feature relations, and explore the values are actually stored in the database. Then, you click or double-click to add the values to the SQL expression, so that no misspellings will occur.

Validating the SQL expression

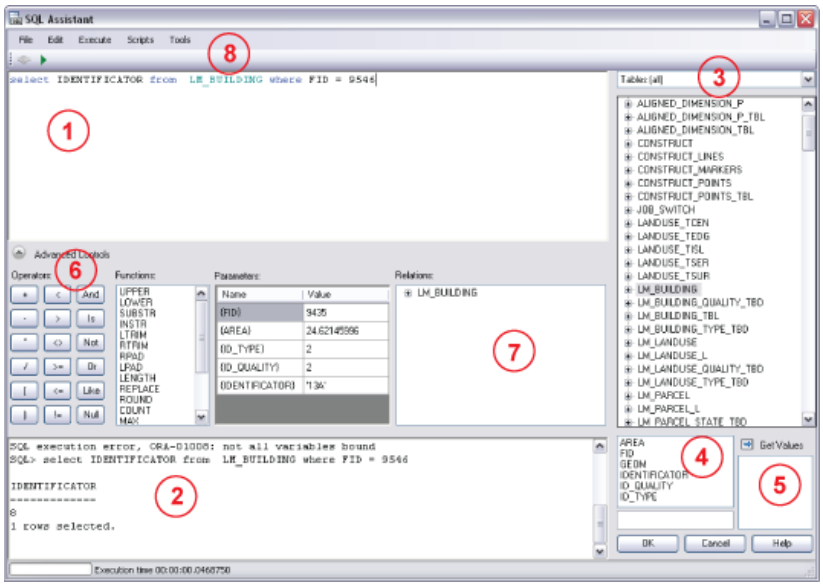
Before saving the select statement, you can validate the syntax. Click Execute menu ➤ Execute or click to execute the select statement. The result is displayed in the script output box. Syntax errors are highlighted and marked with a "^" sign.

TIP Press <ESC> to abort the select statement while it is executed. For example, if the select statement returns a large number of records.

Using the SQL Assistant

The SQL Assistant dialog box consists of the following components.

NOTE The SQL Sheet provides some of the components of the SQL Assistant as a standalone version, such as the Relation Wizard, as described in the following table. See also Using the SQL Sheet (page 130).



Number	Description
1	Expression input box. Displays the select statement. Use the controls or type directly to edit the select statement. Click File menu ➤ Save to store the select statement. Click File menu ➤ Open to load an existing (*.sql) file.
2	Executed script output box. Displays the execution records. Click File menu ➤ Save Output As to store the content in a file.

3	Tables and database objects list. Displays current database objects, for example tables, synonyms or views. Select an object to display more information. In the list, right-click an object, and use the commands. For example, right-click a table name, and click Grid, to view the table content in a data grid.
4	Attribute list. Displays the attributes and the data type of the selected objects.
5	Get Values. Displays the values of the selected attribute. Select an attribute from the list, and click Get Values to display the unique values of the attribute. Duplicate entries are not listed here (SELECT DISTINCT).
6	Advanced Controls section. Displays controls that can be used to create SQL expressions. Click to display all controls.
Operators	Click to add the operator at the cursor point in the expression input field.
Functions	Double-click to add the function at the cursor point in the expression input field.
Parameters	Displays the parameters that are available in the current task. For example, for label definition select statements, the parameter $ID is available. Under Value, enter a value that will replace the parameter, when you run the select statement using the Execute command.
7	Relation wizard. Displays related tables. Expand the explorer nodes, and click to add the relation statement at the cursor point in the expression input field. The expression queries a value **NOTE** The relation wizard is also available from the table list. Right-click the table, and click Relation Wizard.
8	SQL Assistant menu bar. See also SQL Assistant Menu Bar (page 107).

SQL Assistant Menu Bar

The SQL Assistant menu bar provides commands to edit and run the SQL scripts. Most of the commands are standard SQL commands. The following table shows some useful commands.

NOTE Some commands and icons are deactivated in the SQL Assistant. They are available if you use the SQL Sheet. See also Using the SQL Sheet (page 130)

Menu	Description
File	Commands to store and reuse the SQL expressions. The default folder for (**.sql*) files is <topobase_administrator>\Data\SQLSheet\Scripts. ■ Open—Opens an existing (**.sql*) file and adds it to the Expression input box. You can edit the expression, and then execute it. ■ Save—Saves the content of the expression input box. ■ Save Output As—Saves the content of the Executed Script output box.
Edit	Commands to edit the content of the Expression input box and the Executed Script output box. ■ Clear—Removes the content of the Expression input box, or the Executed Scripts output box, depending on where the cursor is.
Execute	Commands to execute and commit the select statements. ■ Execute At Cursor—Executes the select statement beginning at the cursor position. Use this command, if you if you have a large SQL expressions that you want to be executed partly. ■ Spool—Use the spool commands to create a spool file. The default spool file name is <topobase_administrator>\Data\SQLSheet\Log\Spool.log. ■ History—Displays all select statements that have been executed. To re-use, select the select statement, and click Copy To SQL Sheet.

Scripts	Starts SQL scripts that are stored in (**.sql*) files. The menu shows all (**.sql*) files that are stored under <topobase_administrator>\Data\SQLSheet\Scripts. Scripts that are stored in a subfolder will appear in a sub menu. Copy your own SQL scripts to this location to make them available through this menu.
Tools	Commands to start tools. ■ Gather Table Statistics—For example, use this command to update the record numbers that are displayed with the table names in the table list. ■ Spool File—Opens the spool file. ■ Log File—Opens the log file. ■ Options—Settings to control the SQL Assistant, such as autocommit setting, import and export command parameters. For example, set EnableF4 to False, if you do not want the content of the input or output box to be removed when you press [F4].

Using the SQL Assistant in the Form Designer

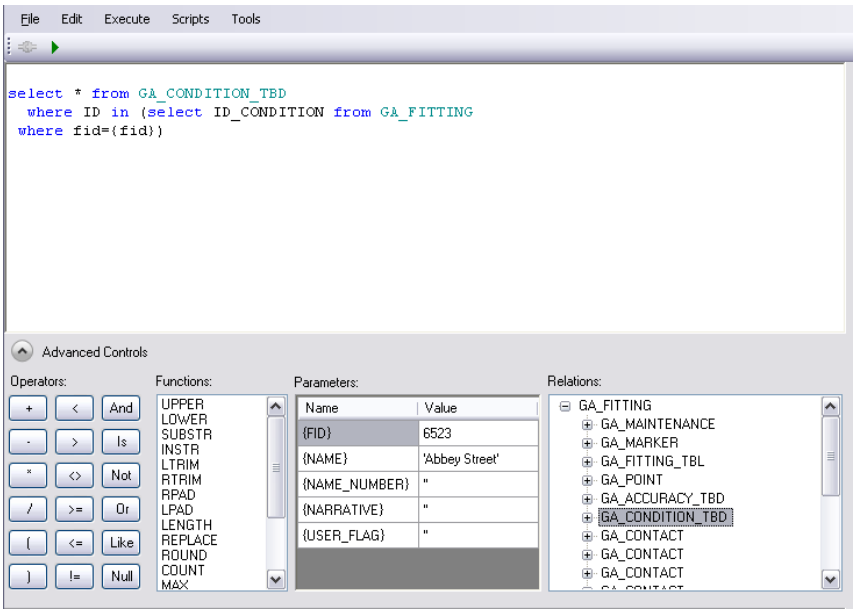
In the Form Designer, you use the SQL Assistant when defining control properties that require SQL statements.

When you click the item in the property pane, a button appears. Click this button to start the SQL Assistant.

Example - Querying relations

In the Gas database schema, the feature class GA_FITTING has a relation to the GA_CONDITION_TBD feature class. The example shows how to query the CONDITION and to display it on the form.

1. Open a workspace that has access to a Gas database schema.
2. Start the Form Designer.
3. Add a Read-Only SQL Text Box. Click the SQL property, and start the SQL Assistant.



Under Relations, the related feature classes are displayed. Select one in the tree view, and click, to insert the select statement that retrieves the related value into the expression input box. Then, replace the * with the attribute you need, such as VALUE.

See also:

- Form Designer Example: Read-Only SQL TextBox (page 362)
- Form Designer Example: Editable SQL TextBox (page 364)
- Form Designer: Editable SQL TextBox - Properties (page 339)
- Form Designer: Feature Highlighter - Properties (page 341)

Select Statement Validation

During your configuration work, you sometimes need to define SQL select statements. In most UIs, you can execute and validate your definitions, such as the definition syntax. You use:

- Position Finder Administrator
- Explorer Configurator.

The execution results are shown in the SQL Info dialog box. In case of errors or unwanted results, you can edit the SQL statement and repeat the validation, as long as it is successful.

See also:

- Explorer Configurator (page 103)
- Sequential Search: Executing and Validating (page 470)

Workflow Definition

Application developers can define workflows. Topobase Workflows provide basic functionalities for workflows that are used in Topobase applications, such as Water, Waste Water and Gas.

See also:

- Workflow Introduction (page 481)
- System Table TB_WORKFLOW (page 102)

Feature Rule Definition

Application developers can define feature rules in the data model administrator.

To edit feature rules

1. Start Topobase Administrator and open a workspace.
2. Click Document menu ➤ Data Model.
3. In the data model explorer, select a feature class, right-click and click Edit Feature Rules.

See also:

- Topobase Feature Rules Reference
- About Feature Rules (page 511)

Report Designer

Using Topobase report designer, you can design your own reports and modify other reports.

See also:

- Report Designer Introduction (page 425)
- Creating a Report (page 121)

Display Models

In Topobase, you use display models to administer thematic views. A display model definition specifies which layer files (feature layers) are loaded into the Display Manager. Also, they define multi map windows.

See also:

- Working with Display Models (page 379)

Topobase Administration Tutorials

Creating a Topobase Document

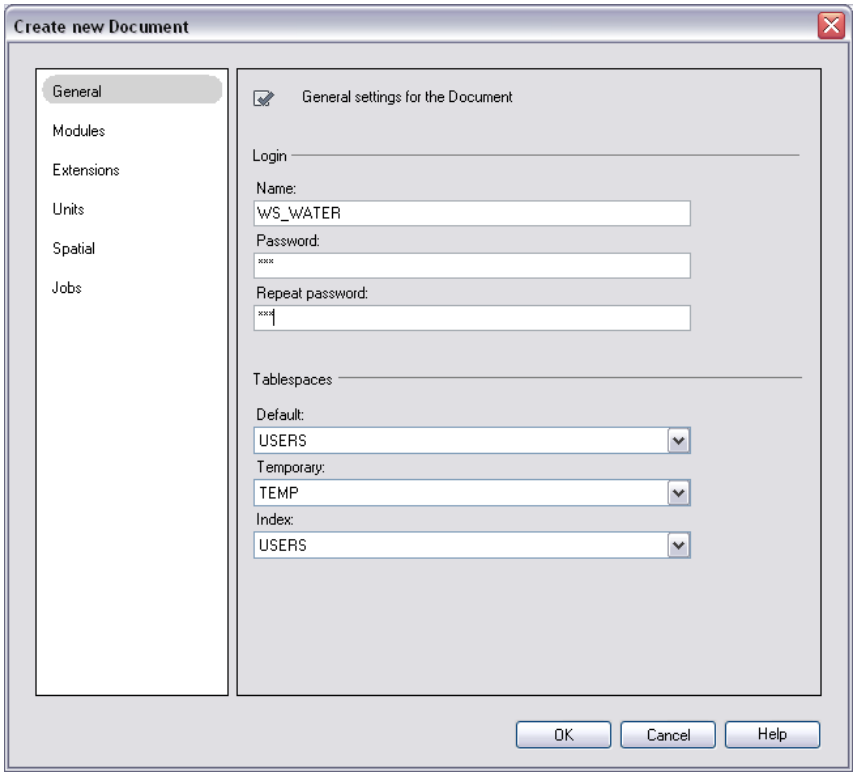
In this tutorial you will use Topobase administrator to create a new Topobase document (page 521). You can define later your own application tables.

To create a Topobase document

1. Start Topobase Administrator.
2. In the Administrator window, click Workspace menu ➤ Create.
3. Enter a name, such as WS_Water, and click Create.
4. In the right pane, in the Workspace Manager section, under Documents, click New.
5. On the navigation pane, click the Modules link.

6 On the right pane, select the Module, such as Water Data Model. Select No Module if you want to create your own data model from scratch.

7 On the navigation pane, click the General link. Enter the name and password for the new document. Also, specify the tablespace names.



Topobase data model administrator: Creating a new document is the same as creating an oracle database schema.

8 Click the Spatial link. These settings are used for spatial storage and operations. Notice the tolerance, which is important for accuracy of coordinates.

9 Click the Units link. These settings are used for calculations.

10 Click the Extensions link. These settings are used to add data model extensions such as COGO, Dimensioning, or Plot.

11 Click the Jobs link. These settings are used for job enabling the document.

12 Click OK. The document is created. The basic structure is displayed in the data model administrator. Now you can enhance the data structure.

See also:

- Data Model Administrator: Creating a Database (page 144)

Creating a Label Definition

In this tutorial you will create a label feature class, a label definition and add the new label feature class to the document explorer.

Labels display the attribute data associated with features in maps, and generate text elements. Labels are stored in a special feature class type, Label Feature Class.

The following example demonstrates point number labels in a map, using data stored in a parent feature class.

To create a label feature class (Topobase Administrator)

1 Start Topobase Administrator and open the workspace.

2 In the Administrator explorer select the document and click Document menu ➤ Data Model. The data model explorer is shown.

3 If possible, expand the feature class and see whether a label feature class exists. If so, you can go to step 7. A feature class can only have one label feature class. If no label feature class exists, you can create one.

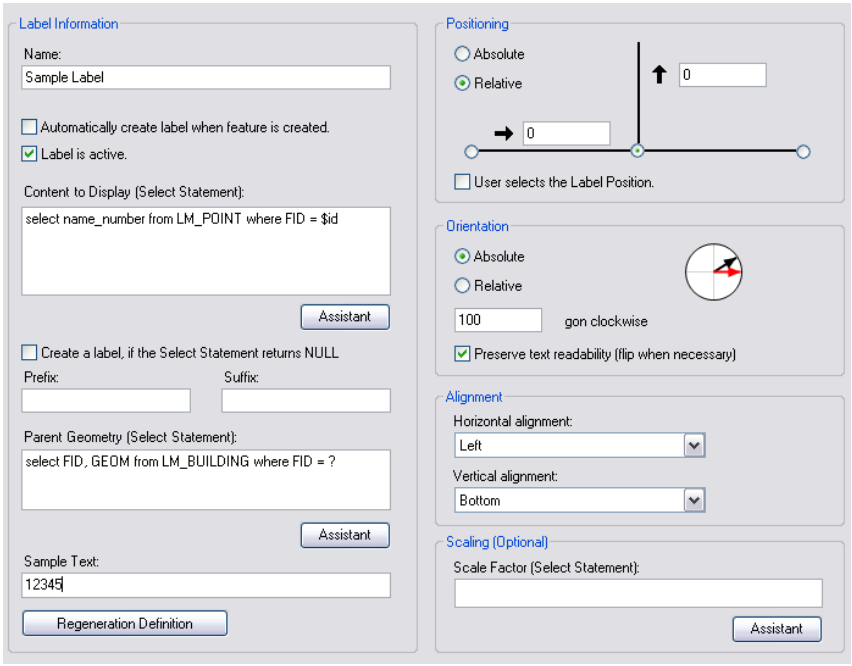
4 Right-click the parent feature class and click Create Feature Class.

5 Enter a name, caption, and for feature class type select Label. Click OK.

NOTE In Topobase Client, in the Display Manager, only label feature classes with the name extension _TBL can be used as Annotation Layers. See also Styling Annotation Layers (page 398).

6 The label feature class is displayed in the data model explorer.

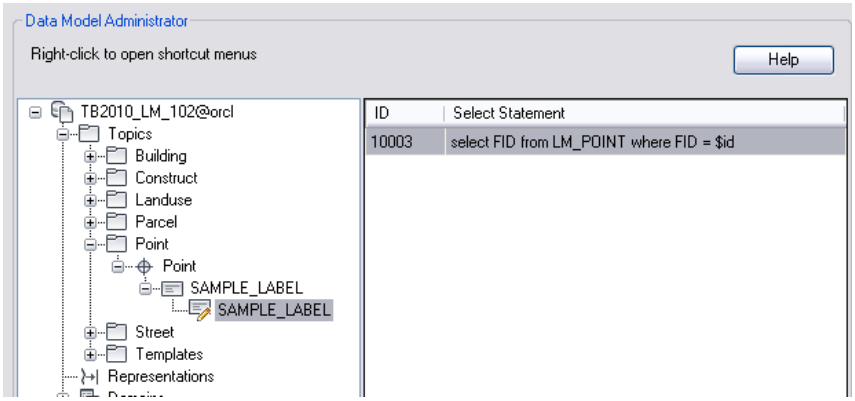
7 Right click the label feature class and click Create Label Definition.



Defining a label for point numbers

To create a label definition (Topobase Administrator)

1. In the Label Properties dialog box, specify the text. In the sample select statement, the label text is the value of the attribute NAME_NUMBER of the parent feature class LM_POINT.
2. Click OK to save the label definition. A label item is displayed in the data model explorer. The select statement is displayed in the right pane.



3 To define more label definitions, right-click the label feature class and click Create Label Definition.

4 Define the properties in the Label Properties dialog box. For example, use the default select statement. Click OK to save the definition.

5 Exit Topobase Administrator.

To add the new label feature class to the document explorer (Topobase Client).

It the new feature classes are not visible in the document explorer, do the following.

1 Start Topobase Client and open the workspace.

2 In the Topobase task pane, click Groups to show the Document explorer group.

3 Click , and expand the parent feature class. Select the new label feature class.

4 Click again. The new feature class is now displayed.

To create a label (Topobase Client)

1 Click Home tab ➤ Display panel ➤ Generate Graphic.

2 In the drawing, select some points belonging to the parent feature class.

3 Click Home tab ➤ Quick Access panel ➤ Attributes to open the feature class form.

4 On the form toolbar, click .

5 In the Choose Label Definition dialog box select the label definition and click OK.

NOTE The Choose Label Definition dialog box is not displayed if there is only one label definition. When you create a label, it is generated immediately. In some cases, the AutoCAD command line might prompt you to click the label position.

For stylization with the Display Manager, make sure that you have established an FDO connection to the new label feature class.

See also:

- Using the Display Manager with Autodesk Topobase (page 388)
- Styling Label Features (page 393)

Creating a Display Model

You create a display model either by modifying the display models that are provided with the demo data sets, or you create a default display model with standard stylization.

See also:

- Creating Display Models (page 383)
- Using the Display Manager with Autodesk Topobase (page 388)

Creating a Workspace

In this tutorial you will create a workspace that contains existing documents and assign the workspace to a user group and create a default display model.

To create a workspace (Topobase Administrator)

1. Start Topobase Administrator.
2. In the Administrator window, click Workspace menu ➤ Create.
3. Enter a name, such as WS_Water.
4. In the right pane, in the Workspace Manager section, click Import.
5. Select Import From Existing Oracle Schema, click OK and select the schema.
6. If prompted to update the structure of the document, click Yes. The workspace is opened.

Assigning the Workspace to a User Group

You must assign a workspace to a user group (page 526). The default user group is the ADMIN group.

To assign a workspace to a user group

1. In the Administrator window, click Setup menu ➤ User/User Group.
2. In the Usergroup section, select a user group, such as VIEWER.
3. In the Workspaces section, select the workspace and assign it to the user group. Either use the button or double-click the item.
4. Exit the Administrator.

Creating a Display Model

You must create a display model that defines the stylization for your feature layers.

To create a display model

1. Start Topobase Client, and open the workspace, such as TB2010_LM_102.
2. Click Home tab ➤ Display panel ➤ Display Model ➤ Open Default Display Model.
3. Click OK to draw the complete document with default stylization. All feature layers are loaded in the Display Manager.
4. Use the Display Manager to stylize the layers.

5 When the layers are styled, click Home tab ➤ Display panel ➤ Save Display Model As.

6 In the Default Display Model dialog box, under Layer List, select an Autoload layer, such as LANDUSE_TSUR.

7 Click OK, and specify a name and folder to save the display model. We recommend that you store each display model in a separate folder.

8 Exit Topobase Client.

See also:

- Understanding Users and Workspaces (page 50)
- Overview of the Display Manager (page 387)

Exploring a Data Model

In this tutorial you will use the Topobase data model administrator to explore the data structures of a document. The following symbols and icons represent structures and relations:

Icon	Explanation
	Topobase root node (expanded), also called document, Oracle user, project, and dump.
	Topic node
	Topic node (not expanded)
	Point feature class with no related objects
	Point feature class with related objects, but not expanded
	Point feature class with related objects, expanded
	Attribute feature class

	Compound Polygon feature class
	Compound Linestring feature class
	Linestring feature class
	Label feature class
	Label definition
	Polygon feature class
	Centroid feature class
	Dimension feature class
	Representations node
	Geoschematic Plan
	Orthogonal
	Generalization
	Domain node
	Domain table
	Topology node
	Area Topology, not yet initialized

	Area Topology, initialized without errors
	Logical Topology. Expand the logical topology item to view the related conditions and tracing templates (network analysis).
	Intersection node, with sub entries
	Intersection
	Utility node
	Utility

To explore the document for an overview of tables, attributes, and relations

1. Start Topobase Administrator and open the workspace.
2. In the Administrator window, click Document menu ➤ Data Model.
3. In the data model explorer, expand the Topics node.
4. Click a topic name to view its feature classes. Feature classes are shown on the right in the property pane.
5. Move the cursor to a feature class row. A tool tip displays information about the feature class, such as the number of features.
6. In the data model explorer, expand a topic and select a feature class name. Attributes and their definitions (Oracle columns) are displayed in the property pane. The caption and the Oracle table name are displayed in the status bar.
7. Expand a label feature class and select a label definition. The SQL Select statement is displayed in the property pane.
8. Expand the Domain node and click a domain name. All values of this domain table are displayed in the property pane. The caption and the Oracle table name (_TBD) are displayed in the status bar.
9. Expand the Topology node and click a topology name. The icon indicates whether the topology was initialized correctly or not. Feature classes that are part of the topology are displayed in the property pane.

Right-click an item to view more details or edit the item.

See also:

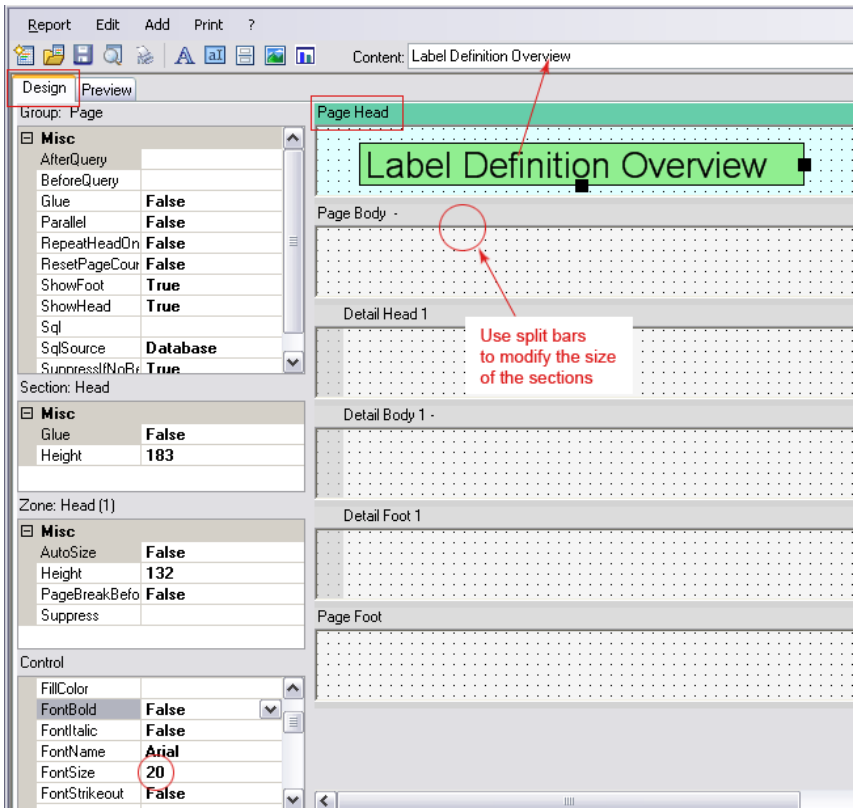
- Data Model: Feature Class Types (page 172)

Creating a Report

In this tutorial you will use Topobase report designer to create reports for Topobase Web and Client.

To create a report

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, select the document.
3. Click Document menu ➤ Report Designer.
4. Click Report menu ➤ New ➤ Blank Report.
5. Enter a report name, such as Label Definition Overview. In the right pane the Page Head section is highlighted.
6. Click Add menu ➤ Text and enter Label Definition Overview. The text control is added in the right pane.



Topobase Report Designer: Creating a new report.

7 Click the new text. The properties are displayed in the left pane on the Design tab.

8 On the Design tab, in the Control category, change Font Size to 20.

9 In the right pane, use the mouse to move and resize the text box and change the height of the page head section.

10 Click the Detail Body 1 section.

11 In the left pane, in the Group: Detail category, select the SQL property and click .

12 In the SQL Assistant, enter the following SQL statement:

```
select * from TB_LABEL_DEF where {Parameter.Filter}
```

Use the tables list, and the Advanced Controls to enter the SQL statement. For example, instead of typing the table name, select the table, and double click. Instead of typing the parameter expression, select the parameter in the list, and double click.

13 In the SQL Assistant, click Execute to validate the SQL statement.

14 Click OK to exit the SQL Assistant.

15 Click Add menu ➤ DB Column.

16 In the Add DB Column dialog box, Ctrl-click NAME and SELECT_STATEMENT. Click OK. Two controls are added to the Detail Body 1 section.

17 You can now position the controls with the mouse.

The application replaces the value {Parameter.Filter} at runtime with the current filter of the relevant form. Only the data of the current filtered records is printed.

18 Click the Page Body section.

19 Click Add menu ➤ Text.

20 Add two text controls: “Name” and “Label Select Statement”.

21 Click the Page Foot section.

22 Click Add menu ➤ Special ➤ Page X Of Y to add a page counter.

23 Click Add menu ➤ Line to add a line to separate the foot from the body.

24 Click Report menu ➤ Save.

Next, you will assign the new report to the feature class form TB_LABEL_DEF.

25 In the left pane, click the Preview tab to display the report. You can print or export the report to HTML or ASCII.

26 Click Report menu ➤ Quit to save the report.

27 In Topobase Client, in the document explorer, expand the System Tables node and open the form TB_LABEL_DEF.

28 On the form toolbar, click the Print icon.

29 Right-click the Report dialog box. Click Add ➤ Topobase Report.

The Title is displayed when a user selects a report. The Report Name is the name of your newly created report.

30 Click OK to save the report.

To view the report, select it in the list and click OK.

Creating a Report Head

In this tutorial you will design reports with different layouts for each page, such as a report with a heading.

To create a report head

1 Define a report as described in Creating a Report (page 121).

2 Click Report > Open to open the report designer.

3 Click Add > Page.

In the right pane, for each page, a separate layout tab is displayed.

4 Click the new empty page. Click Edit menu ➤ Move ➤ Move Page Up. Now, the new page is the first page.

5 Design the title page. For example, add a title (text control) and a logo (picture control) to the page head section.

6 Click Report menu ➤ Options, to draw a frame round the title page.

7 In the Options dialog box, click the Page 1 tab. Select Draw Frame and click OK.

8 Save the report and click the Preview tab, to view the new report.

Report Definition: Adding a Chart

In this tutorial you will insert a chart to a report, displaying the land use distribution.

To create a report that contains a chart

1 Start the report designer.

2 Click Report menu ➤ New ➤ Wizard.

3 Enter the report name, such as Land Use Distribution, and click OK.

4 In the New Report dialog box, under Table/Form, select the land use feature class LM_LANDUSE.

5 Under Title, enter Land Use. Click OK.

6 In the report designer, enlarge the Page Body section by dragging the section.

7 On the toolbar, click .

8 In the Chart Wizard dialog box, under Chart Type, select Bar.

9 On the navigation pane, click the Title tab, and under Text, enter Land Use. Select a font, and a color.

10 On the navigation pane, click the Chart tab, and select Show Legend.

11 Click the Axes tab, and for the Axis Y, select Automatic.

12 Click OK to finish the Chart Wizard.

13 In the Page Body section, select the chart control, and modify size and position.

14 In the left pane, in the Control property grid, select the SQL property. Click to open the SQL Assistant.

15 In the expression input box, enter the following select statement.

```
select b.value, count (a.id_type)
from LM_LANDUSE a,
LM_LANDUSE_TYPE_TBD b
where a.id_type=b.id
group by a.id_type, b.value
```

16 Click Execute menu ➤ Execute to check the select statement.

17 Click OK.

18 In the report designer, on the left, click the Preview tab.

19 Click the Design tab, and in the Control property grid, select Config, and click .

20 In the Chart Wizard dialog box, click the Axis tab.

21 Under Axis Y, clear Automatic, and adjust the scaling of the Y axis. For example, enter the following values. Minimum: 10, Maximum: 1300, Minor Step: 50, Major Step: 100.

22 Click OK.

23 Click the Preview tab.

24 Click Report menu ➤ Save.

The select statement returns a value for each land use type (LM_LANDUSE_TYPE_TBD.VALUE), and counts the number of features (LM_LANDUSE) for each type.

See also:

- Report Designer: Chart Control (page 444)

Creating a Master-Detail Report

In this tutorial you will define reports that print master-detail information. These reports print information in a report, using related information that is stored in another feature class.

To create a master-detail report

1 Open the report, using the report designer.

2 In the right pane, click the Detail Body 1 section.

3 Click Add menu ➤ Child Group. The child group with Head 1.1, Body 1.1, and Foot 1.1 is inserted between the Detail Body 1 and Detail Foot 1 sections.

4 In the right pane, select the Detail Body 1.1 section.

5 In the left pane, in the Group:Detail property grid, select the SQL property to define the SQL statement. Click and enter the following statement:

```
select * from ...
where ... = {Id}
```

At runtime, the application replaces the value {Id} with the value of the column ID from the record in the parent group.

6 Click Add menu ➤ DB column. Select the columns you want to print. You can select several items (multi selection) by pressing and holding <SHIFT> or <CTRL>.

7 Click the Detail Head 1.1 section.

8 Click Add menu ➤ Text. Add the text controls for the caption.

9 Click Report menu ➤ Save.

10 Click the Preview tab. Each master display model followed by the detail definitions and colors is displayed in the report.

For more information about master-detail forms refer to TB_RELATIONS: Master-Detail Relations (page 96).

Creating and Using Dimensions

In this tutorial you will, acting as a Topobase database administrator, add the dimensioning feature classes. Then, as a user, you will create and delete dimensions.

Dimensioning must be prepared by the database administrator, who adds the Dimension Extension to the database. Users can then create dimensions in Topobase Client.

To add the Dimension Extension (Topobase Administrator)

1 Start Topobase Administrator and open a workspace.

2 In the Administrator explorer, select the document.

3 On the right pane, under Document Settings, on the navigation pane, click the Extensions link.

4 Select Dimension Extension.

5 Click Save to add the topic Dimension. This topic groups dimensioning feature classes for aligned dimensioning and orthogonal dimensioning, and the label definitions.

The label definitions are used to inscribe the dimension values. The values are stored in the dimension line feature class (CALCULATED_x and MEASURED_x). The label definitions use the LINE_TYPE attribute , and the POINT_TYPE attribute to distinguish between the dimensioning types.

6 Make sure that the Dimensioning feature classes are added to the document explorer.

7 Define the stylization of the dimensioning layers in the Display Model.

To create dimensions (Topobase Client)

1 Start Topobase Client, open a workspace and generate graphic.

2 See Adding Dimensions.

To delete dimensions (Topobase Client)

A dimensioning consists of several features. You can either delete each feature on its own or delete all related features.

To select all related features, select the abscissa. To select a single feature, select the feature, such as the ordinate. Click Delete Feature.

See also:

- Data Model: Dimensioning (page 205)

Creating A Profile Definition

In this tutorial you learn about creating a profile definition. The Wastewater demo data set provides a sample profile definition. Use the sample to get familiar with profile definitions.

You can create your own profile definitions in any document to draw longitudinal profiles of any feature classes. Profile definitions are based on the Profile data model. The following steps are required:

Creating a profile definition from scratch

- Add the Profile Extension (profile data model) to the document. See Adding the Profile Extension (page 129).
- Design a profile, identifying which features will be included, such as layouts, scale ranges, and labels.
- Use the data model administrator to create the profile feature classes. Usually you have one corresponding profile feature class for each real world feature class. See Profile Feature Classes (page 506)
  Example: You want to draw a wastewater profile with manholes and sections (feature classes WW_MANHOLE and WW_SECTION). Then, you need the profile feature classes WW_MANHOLE_PRO and WW_SECTION_PRO.
- Use the Profile Designer to create the profile definitions. See Profile Designer (page 486).

The profile definitions are available in Topobase Client in the Profile Manager. In the Profile Manager, you can select the definition, modify the default settings, select the features and the profile axis, and create and store the profiles.

See also:

- Overview of Profiles (page 485)

Topobase Client User Guide, section Profile Manager.

Adding the Profile Extension

In this tutorial you will use the data model administrator to add the Profile Extension to a document.

To add the Profile Extension (Topobase Administrator)

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, select the document.
3. Click Document menu ➤ Settings.
4. Under Document Settings, on the navigation pane, click the Extensions link.

5 Select Profile Extension.

6 Click Save. The document is loaded again.

7 If prompted, click Update to create the profile tables.

See also:

- Profile Data Model (page 496)

Using the SQL Sheet

In this tutorial you get familiar with the SQL Sheet. The SQL Sheet is the standalone version of the SQL Assistant.

See also Creating SQL Expressions (page 103).

IMPORTANT To use the SQL Sheet you have to be familiar with Oracle and SQL. This knowledge is not subject of this manual.

To start SQL Sheet

1 Start Topobase Administrator

2 Click Workspace menu ➤ SQL Sheet.

NOTE You can start the Topobase SQL Sheet without Topobase Administrator. The SQL Sheet is installed with Topobase Administrator in the <topobase_admin>\Bin folder. Create a shortcut of the file <topobase_admin>\Bin\SQLSheet.exe. Then, you start the SQL Sheet by double-click.

To log in - connect

- Under User, select an Oracle user from the list, and enter the password and the Oracle service name.
- Optionally, enter the Oracle system user name and password. These are required, if you want to import or export Oracle (**.dmp*) files, or drop users or tables.

To connect to another database

- Click Connect to open the Login dialog box.
- Variant: In the list of database objects, select Users.
- In the user list, right-click the user, and click Connect.

To export an Oracle user to a (.dmp*) file**

- Optionally set export options: Click Tools menu ➤ Options.
- In the list of database objects, select Users.
- In the user list, right-click the user, and click Dump Export.

To view the database size

- From the database objects list, select User Spaces. The list displays all database users with their file size.
- From the database objects list, select Tablespaces. The list displays the size of the tablespaces. Expand the node items, and use the shortcut menu to change the size of a table space.

To export the current Oracle user to a (.dmp*) file**

- Optionally set export options: Click Tools menu ➤ Options.
- Click Tools menu ➤ Dump Export.

To create an Oracle user

You can use the SQL Sheet to create an create Oracle user. Note, that this is not a Topobase database schema. To create a Topobase database schema, use Topobase Administrator.

- Click Tools menu ➤ Create User.

To view database objects

From the database objects list, you select a database object. Then, select an item, and right-click to open shortcut menus. For tables, you can apply several predefined filters.

- Users—Displays all Oracle users.
- Tables (All)—Displays all table names, including the size of the stored data. The data size is calculated automatically when you connect to the user.
- Tables (User Tables With Data)—Displays all user tables that contain data. User tables are non system tables. For example, in a Topobase document, user table names start with a prefix that indicates the module, such as WW for Wastewater, or GA for Gas.
- Tables (User Tables All)—Displays all user tables.
- Tables (Topobase System)—Displays the Topobase system tables. See also Topobase System Tables (page 71).
- Tables ($)—Displays the spatial tables.
- Tables (Filtered)—Displays tables using a filter expression that you enter, such as *TB%*.
- Other database objects, such as Views, Synonyms, Sequences, Indexes, Triggers.

To transfer the attribute names to the expression input box

- When yo select a table, the column names are displayed. Select one or more items from the list, right-click and click Select ? From Table.

SQL Sheet menu bar

- See also SQL Assistant Menu Bar (page 107).

To start the SQL Spy and SQL Error Spy

Use the SQL Spy, and the SQL Error Spy to analyze problems. The SQL Spy displays all select statements that are currently executed against the Oracle database by any Topobase or other application. The SQL Error Spy displays select statements that could not be executed.

- Click Tools menu ➤ SQL Spy to start the SQL Spy.
- Click Tools menu ➤ SQL Error Spy to start the SQL Error Spy.

SQL Sheet commands

The SQL Sheet provides three types of commands. As appropriate, the commands can be started through the menu bar, the shortcut menus, or you can type the commands directly in the expression input box.

- See SQL Sheet: Standard SQL Commands (page 133).
- See SQL Sheet: SQL Plus like Commands (page 133).
- See SQL Sheet: Special Commands (page 134).

SQL Script Examples

For more examples, view the sample scripts that are provided under <topobase_administrator>\Data\SQLSheet. To add the scripts to the expression input box, do one of the following.

- Click Scripts menu ➤ <script name>.
- Click File menu ➤ Open, and select the file.

SQL Sheet: Standard SQL Commands

IMPORTANT To use the SQL Sheet you have to be familiar with Oracle and SQL. This knowledge is not subject of this manual.

For example, standard commands are: COMMIT, ROLLBACK, SELECT, INSERT, UPDATE, DELETE, CREATE, DROP.

SQL Sheet uses transactions, and the COMMIT is executed either automatically when you quit the SQL Sheet, or when you start it explicitly through the Execute menu ➤ Commit. You can use the AutoCommit option, if you want to your changes to be committed immediately. Click Tools menu ➤ Options.

SQL Sheet: SQL Plus like Commands

IMPORTANT To use the SQL Sheet you have to be familiar with Oracle and SQL. This knowledge is not subject of this manual.

- DEFINE <variable name>—Creates a user defined variable. Note that in SQL Sheet, the command must be finished with a ";".

- DESC <table name>—Displays the table definition in the Executed script output box.
- SPOOL commands—See SQL Assistant Menu Bar (page 107).
- SPOOL <file name>—Creates a spool file with the given name.
- SET HEADING ON/OFF— Switches off the headings of the output result. For example, use SET HEADING OFF and Set ECHO OFF if you want to create a script from the output, and you only need the data without any comments.

SQL Sheet: Special Commands

IMPORTANT To use the SQL Sheet you have to be familiar with Oracle and SQL. This knowledge is not subject of this manual.

- ABORT—Ignores all select statement that follow this command. For example, if you have several select statements, use this command to execute only the first part.
- CONNECT—Connects to the database. Optionally, use parameters, for example CONNECT <user name/password>, or CONNECT <user name/password@service>.
- EXPORTDUMP <user name> <filename>— Creates a (**.dmp*) file. To specify the export options, click Tools menu ➤ Options, and set the CommandLineDumpExport option.
- EXCEL <select * from <table name>>–Opens MS Excel and displays the result of the select statement in an (**.xls*) file. Use the command EXCELFILE <file name> <select * from <table name>> to store the result of the select statement in a file without opening the Excel file.
- ASCIIFILE <file name> <select statement>–Stores the result of the select statement in a TAB separated ASCII (**.txt*) file.
- GRID select * from <table name>–Displays the result in a grid dialog box. You can edit the values.

  NOTE To commit the changes, click Update, Commit And Close, or start the COMMIT command.

- REM–Marks a comment. Or enter "--"

- FIND <value> [,Filter]–Searches all columns of all tables for the given value. Optionally, set a filter.
- CLS or CLEAR–Clears the output window.
- EXECUTE <procedure name>– Executes a stored procedure.
- SELECTDOUBLES <select statement> CONDITION <condition> KEY <key>–Finds duplicate values. For example: `SELECTDOUBLES * from MY_FEATURE condition TB_POINT_NUMBER KEY FID` <condition> contains the column names to be analyzed. <key> contains the primary key.
- DELETEDOUBLES <select statement> CONDITION <condition> KEY <key>–Deletes duplicate entries.
- SETDELETEDDOUBLES <...> CONDITION <condition> KEY <key>–Use this command for previous Topobase versions (TB2). Sets the attribute DELETED to 1 for duplicated values. Example: `SETDELETEDDOUBLES * from MY_FEATURE CONDITION attribute1,attribute2`.
- SHELL Filename [,wait]–Executes a system command. Example: `SHELL notepad.exe`. Example: `SHELL notepad.exe,true`.
- EXPLAINPLAN <select * from <table name>>–Find the execution plan of a select statement, so that you can analyze which indexes are used, and find out how to optimize the select statement.
- REPEAT <...> FOR <...>— Repeats a select statement, and inserts parameters. The first select statement is repeated, and the values from the second select statement are inserted into the first select statement. The results are stored in a temporary table TB_SQL_RESULT.
- GATHERTABLESTATS—Collects statistical information for all tables of a user. The command executes the following: `BEGIN DBMS_STATS.GATHER_TABLE_STATS('username', 'tablename', cascade=>TRUE); END;`
- PACKDATA <select expression>—Writes all column values of a child table to a column PACKDATA1 of a parent table. The column PACKDATA1 is of the data type VARCHAR2. Example: `packdata select p.fid, c.name from parent p, children c where p.fid=c.FID_parent order by p.fid;`
- PACKDATA2—Writes all column values of a child table to a column PACKDATA2 of a parent table. The column PACKDATA2 is of the data type CLOB. Use this command for large data sizes.

- COMPDBWITH <user name>—Compares the data structure of the current Oracle user with another user, and displays the differences.
- DEF <table name>—Displays the table definition in the Executed script output box. The command displays more details that the DECSC command.

Creating a Plot Using AutoCAD Blocks

In this tutorial, you will create a plot that uses an AutoCAD block as a frame, instead of using the normal Topobase plot decoration features.

Prerequisites

To perform this tutorial successfully, you should be familiar with the following aspects of AutoCAD, AutoCAD Map 3D, and Topobase:

- What AutoCAD blocks and AutoCAD block references are, and how to use AutoCAD's block editor
- How to use the AutoCAD Map 3D Display Manager
- How to use Topobase workspaces, documents, feature classes, features, and generic forms
- How to create a Topobase display model and set up a repository path
- How to create and edit a Topobase plot, and use the Plot Library

Assumptions

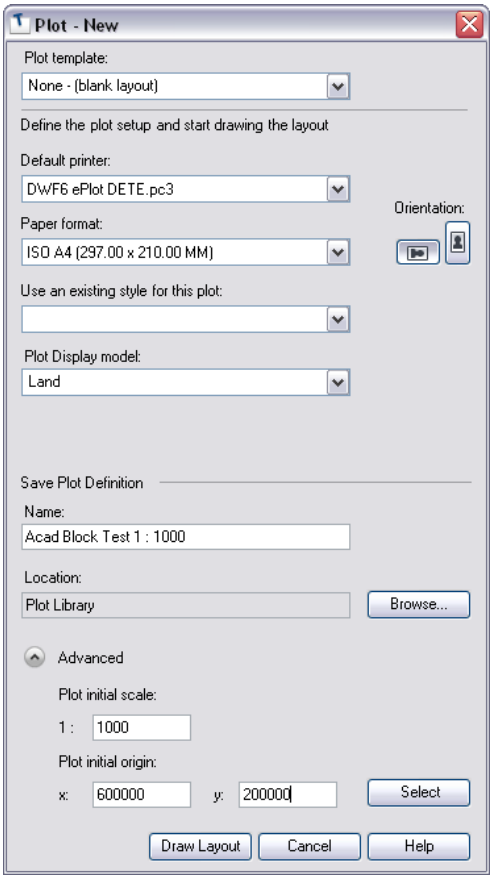
Before performing this tutorial, the following should be true:

- You have created a workspace.
- You have imported the Land Management demo dump shipped with Topobase 2010 and named the document PLOT_ACADBLOCKS.You have enabled the Topobase Plot Extension for that document.
- You have set up the repository path for your display models correctly.
- You have copied the default display model for plots, shipped with Topobase 2010, to a sub-directory of your display model repository path, and adjusted all settings.

To create a plot using AutoCAD blocks

1 In Topobase Client, create a new plot.

- Use the settings shown below.

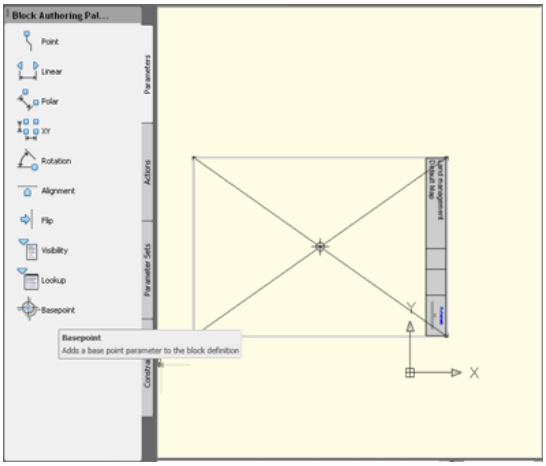


- Click Create to open the plot for editing.
- Close all open drawings except the one that contains the newly created plot.

2 In the newly created plot drawing, type *BEDIT* to open the AutoCAD Block Editor.

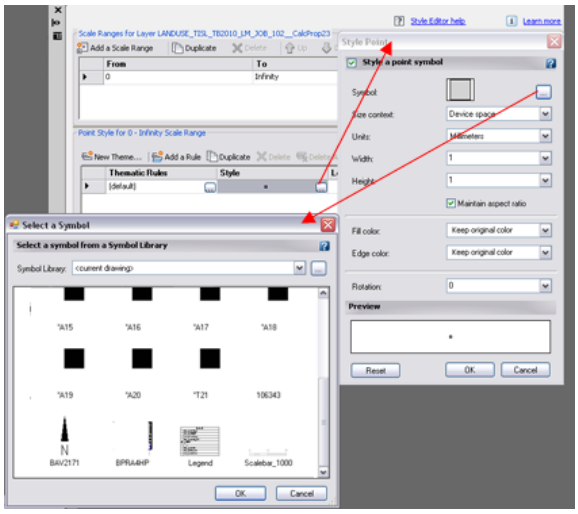
- Create a new AutoCAD block that is 297 units wide and 210 units tall (A4).

- Set the block's base point to its center.



- Close the block editor.

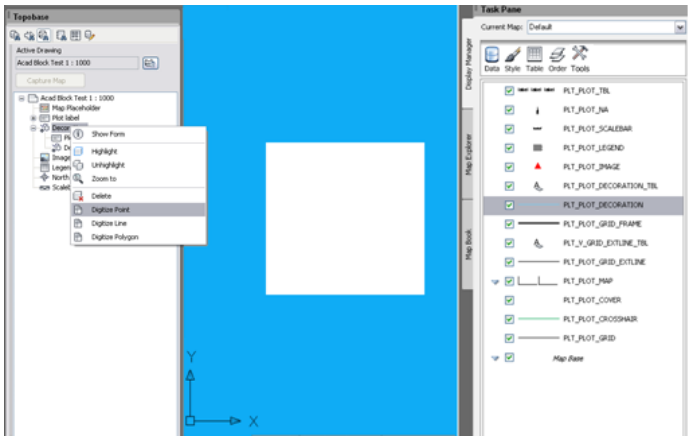
3 Style the layer PLT_PLOT_DECORATION.
   - In the Display Manager, select the layer and click Style.
   - In the Style Editor, under Point style 0 - Infinity Scale Range, click the browse button in the column labeled Style.Set the parameters as follows:
      - Symbol: Select the block you just created from the current drawing.
      - Size Context: Map space
      - Units: Meters
      - Width: 297 (A4, 1:1000)
      - Height: 297 (this is set automatically when you set the width)
      - Fill color: Keep the original color.
      - Edge Color: Keep the original color.
      - Rotation: Enter the expression ORIENTATION * 0.9 + 90



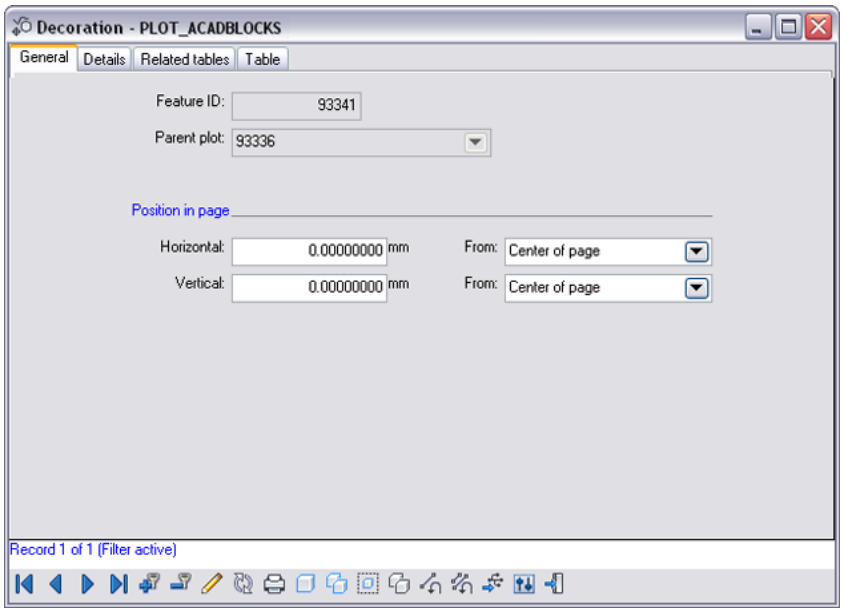
4 Save the current display model, overwriting the existing plot display model.

This creates a new drawing named BlockDefinitions.dwg, which contains the block you just created.

5 Digitize a new plot point decoration feature, as shown below.



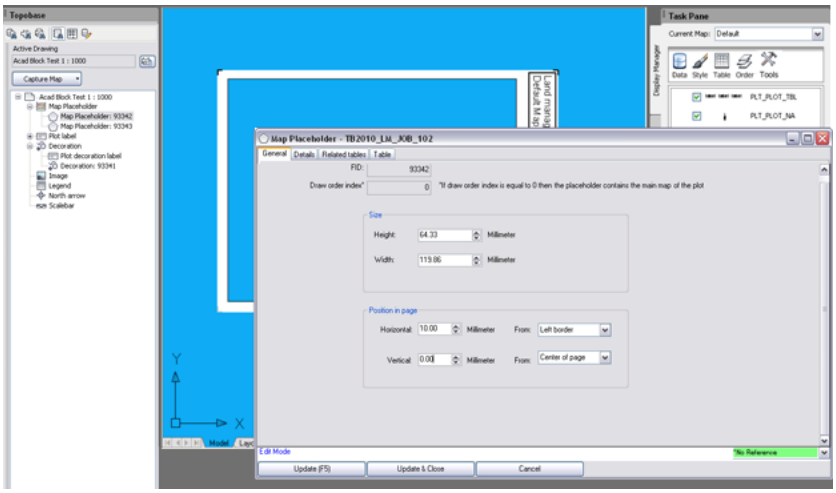
6 Open the generic form for this point feature, and set its position to the center of the page, as shown below.



Once updated, the AutoCAD block will be centered on the paper.

The point decoration feature is styled using the AutoCAD block you created. This AutoCAD block has its base point set at its center. Because the point decoration feature is centered on the paper, the AutoCAD is correctly positioned.

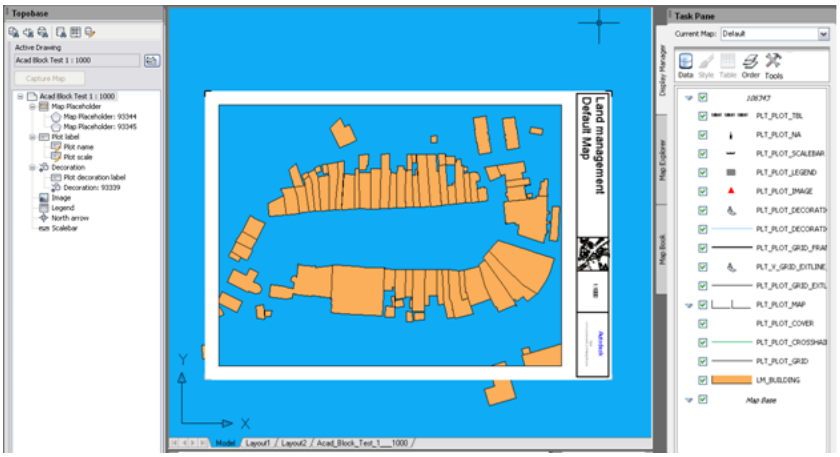
7 Optionally, create a (primary) map placeholder and set it to a pre-defined width and position.



- Capture the map for this map placeholder.
- Specify an existing display model.
- Apply a rotation value, if necessary.

8 Optionally, extend the plot by adding a label for the plot's scale, an overview map, and any other desired elements.

The plot shown here is scaled at 1:1000.



Data Model 2

Data Model Administrator Introduction

Use the data model administrator to create a Autodesk Topobase™ document (page 521) and to configure feature class tables within a document.

To learn more about administrative tasks, refer to the following topics:

- Topobase Data Modelling (page 35)
- Data Model Administrator: Creating a Database (page 144)
- Data Model: Feature Class Types (page 172)
- Data Model: Labels (page 187)
- Data Model: Topologies (page 215)

Starting the Data Model Administrator

Only members of the administrator user group (page 526) can start the Topobase Administrator module!

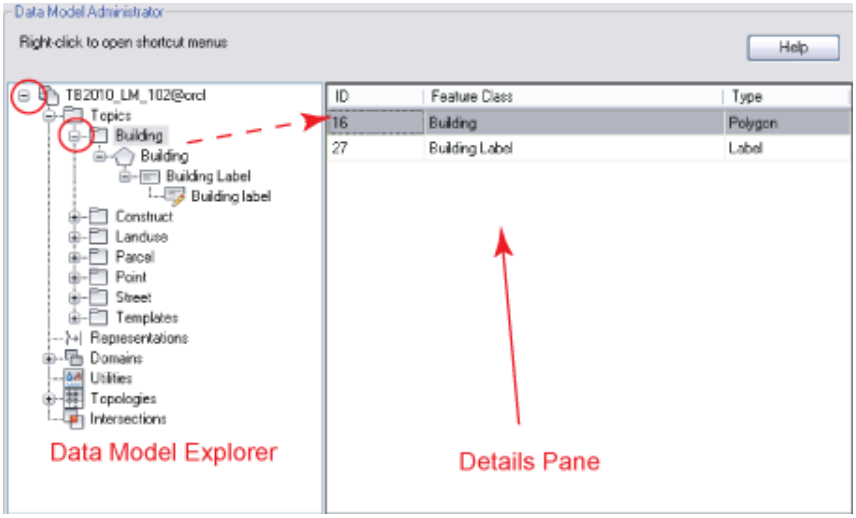
To start the data model administrator

1. Click Start menu ➤ Programs ➤ Autodesk ➤ Autodesk Topobase 2010 ➤ Autodesk Topobase Administrator 2010.
2. Log in as administrator.
3. Click Workspace menu ➤ Open, and open a workspace (page 526).

4 In the Administrator explorer in the left pane expand the workspace item and select the document (page 521).

5 In the Administrator window, click Document menu ➤ Data Model.

In the right pane, the data model administrator is opened.



Topobase data model administrator with data model explorer.

If you select a topic (page 524), the details pane shows the feature classes of this topic. If you select a feature class (page 522), the details pane shows the attributes (database columns).

If you point to an attribute in the details pane, the attribute properties are displayed in a tool tip.

Data Model Administrator: Creating a Database

You use the Workspace Manager to create a new Topobase document (page 521).

Define the new document in the Create New Document dialog box, using the following settings:

NOTE For existing documents the Document Settings are read only. To view the settings in the Administrator explorer, click the document node. See also Document Settings (page 7).

Create New Document dialog box	Use the links on the navigation pane to access the document settings.
General settings	Specifies the database names and table spaces
Login	Specifies the name and password of the Oracle database schema. Specify the name according to Oracle conventions. Recommendation: Use only the characters a-z, 0-9 and _ (underscore). Do not specify a name that starts with a number.
Tablespaces	Specifies the tablespaces. Select default, temporary, and index tablespaces from the objects available in your Oracle database.
Modules settings	Selects the module database schema. Do not select any module if you want to create a new database schema from scratch. Select a module, such as Water, Wastewater or Electric to create a pre-defined data structure. Performs the following steps: Stores the data model code in the system table. Creates the database schema, using the unit settings. Imports pre-configured forms. See also About Data Models (page 148).
Extensions settings	Displays data model extensions to be applied. The following extension can be added to any module or database. ■ COGO Extension—Adds the COGO feature classes. See also *Construction Introduction*. ■ Dimension Extension—Adds the default feature classes for orthogonal and aligned dimensioning. See also Data Model: Dimensioning (page 205).

- Interlis Export Extension—Adds feature classes to store the Interlis export configuration. See the Land Management CH User Guide.
- Plot Extension—Adds the plot feature classes. See also Plot Reference (page 417).
- Profile Extension—Adds the data structure to manage profile data. See also Overview of Profiles (page 485).
- Oracle Data Import Extension—Adds system tables TB_MIG_* that are required for the Oracle Data Import. See also Topobase Data Transfer (Oracle Data) (page 239).
- Templates Extension—Adds the template feature classes. See also Data Model: Feature Templates (page 216).

Units settings	Displays settings that specify default units for the attributes, such as for ORIENTATION or LENGTH. Also, you can define specific units for each attribute when you create it. The settings are used when the document is created either for the default attributes or, for example, during the creation of a water module: WA_TANK.CAPACITY is a volume. Depending of the default settings, the corresponding unit will be set. The settings must never be modified at any later time. The settings are saved in the system table TB_INFO. The default orientation is dependent on the angle unit. Examples: Selected: Degree and Clockwise -> Default orientation = 90 Selected: Gon and Clockwise -> Default orientation = 100
Spatial settings	Settings that specify spatial storage and operations and values for calculations.
Dimensions settings	Selects the dimension setting.

2D	Selects 2D to store the plane geometry (easting, northing) in the GEOM attribute, and the elevation (height) in the Z attribute.
3D	Selects 3D to enable 3D support, and to store the coordinates (easting, northing, and elevation) in the GEOM attribute. **IMPORTANT** We recommend that you use Oracle 11g, if you enable 3D support.
Tolerance	Oracle Spatial Tolerance defines the maximum precision for numeric data; the default value is 0.0005. **IMPORTANT** This setting is critical for data processing performance and the size of spatial indices. For best results, we recommend it be set to a consistent value between 0.01 and 0.005 for all tables in a schema. For example, a setting of 0.005 means that measurements in meters can be precise to 5 mm. Project architects are encouraged to set this value before creating any feature classes in Topobase. The spatial tolerance value can be altered for a specific feature class.
Min, Max	Defines the range of the spatial index. For faster performance, set these values close to the actual used range of coordinates, that is used.
Spatial Reference ID	Assigns a spatial reference ID (SRID). For spatial indexing you can assign a spatial reference system (Oracle spatialSRID). The SRID is saved in the MDSYS User (CS_SRS). Some valid settings are: Switzerland: SRID = Swiss National System (352257) Germany: SRID = GK Zone x (DHDN) This parameter is important for later spatial operations, such as transformation into another system. All feature classes will inherit this SRID. See also About Coordinate Systems (page 150)

Jobs Settings	Specifies settings for versioning, such as job enabling, feature locking, and job perimeter. See also Creating a Job Enabled Document (page 300).

See also:

- Creating a Topobase Document (page 111)
- Data Model Administrator: Creating a Feature Class (page 160)
- Data Model: Spatial Relationship Settings (page 233)
- Updating the Coordinate System (page 151)
- Overview of Profiles (page 485)

About Data Models

When you create a Topobase document, you can specify whether to add a predefined data structure. In the Document Settings, under Modules, you select the data model to add to the document.

Data Model	Description
Electric Central Europe Data Model	Data model for Topobase Electric Central Europe (CE).
Electric North America Data Model	Data model for Topobase Electric North America (NA).
Gas Data Model	Data model for Topobase Gas.
Land Switzerland Data Model	Data model for Topobase Land Management CH.
Land Germany Data Model	ALKIS data model (based on GeoInfoDoc V6).
Survey Module	Database schema for Topobase Survey (Survey document). Use one central Survey document to control your calculation projects. See Survey Database.

Wastewater Data Model	Data model for Topobase Wastewater. See the Topobase Wastewater User Guide.
Wastewater Data Model (For Import Document)	Data model for the import of inspection data. The Topobase Wastewater User Guide.
Water Data Model	Data model for Topobase Water. See the Topobase Water User Guide.

Data Model Description

Print a Data Model Description report to get a complete overview about topics, feature classes, and attributes.

To print a data model description

1. Start Topobase Client, and open the workspace.
2. In the Document Explorer, select the document.
3. Click Output tab ➤ Reports And Profiles panel ➤ Open Report.
4. In the Report dialog box, select the report Data Model Description.
5. Click Preview.
6. In the preview window, select one of the output options.

Additional information

Entity Relationship Diagrams are available for the module data models. In addition, explanations of the different feature classes are available in the module documentation.

Module	Entity Relationship Diagram	Data Model Description
Water	*Water Data Model Diagram*	Topobase Water Data Model
Wastewater	*Wastewater Data Model Diagram*	Topobase Wastewater Data Model
Gas	*Gas Data Model Diagram*	Topobase Gas Data Model

Module	Entity Relationship Diagram	Data Model Description
Electric North America	*Electric NA Data Model Diagram*	Topobase Electric NA Data Model
Electric Central Europe	*Electric CE Data Model Diagram*	Topobase Electric CE Data Model
Land CH	*Land Management CH Data Model Diagram*	Topobase Land Management CH Data Model

About Coordinate Systems

Topobase supports multiple coordinate systems. This means that you can join different database schemas with non-identical coordinate systems. Also, this means that a workspace (page 526) can contain several documents where each document (page 521) has a different coordinate system (SRID). The Topobase support covers graphic generation and graphic editing (digitizing with back transformation of the digitized coordinates).

NOTE A document can have either one SRID assigned or none. The document SRID automatically applies to all feature classes of the document.

During graphic generation, when a workspace contains several documents with different coordinate systems, the SRID is used for on-the-fly coordinate transformation, representing the geometry data in AutoCAD Map in one global coordinate system.

Use Topobase data model administrator to assign a coordinate system:

- Assign an SRID when you create the document. See Data Model Administrator: Creating a Database (page 144).
- Set the SRID of a document at any time. See also Updating the Coordinate System (page 151).

NOTE These settings apply to the database. In addition there are coordinate system settings, such as the coordinate system of a drawing (Map Coordinate System) and of a feature layer (Display Manager - FDO provider).

If you represent the data in a drawing, Topobase Client validates the configuration of the coordinate systems and performs the transformation

when necessary. For example, a workspace has document SRID A and the AutoCAD Map drawing contains the Global Coordinate System B. For representation in the map, a transformation from A to B is performed. If one of the coordinate systems has not been set, no transformation and no generation of graphics is possible.

Although the representation is transformed in the drawing, all calculations and geometry updates are made in the database coordinate system. Also, the coordinate values shown in the Topobase dialog boxes and feature class forms are the internal database values.

See also Topobase Client User Guide, *Working with Multiple Coordinate Systems*.

For more information about SRID, see *Oracle Spatial Users Guide and Reference*.

Updating the Coordinate System

You can set the SRID (coordinate system) for all feature classes of a document (page 521). For example, you have imported a document whose coordinates are stored in system B, and no SRID is set. If you want to process this document in a workspace that contains a document with a distinct coordinate system, you must set the SRIDs accordingly.

Restrictions

- If the document contains views with relations to external tables, you cannot update the SRID.

IMPORTANT The Update Coordinate System command does NOT transform the coordinates in the database.

The Update Coordinates System command updates the metadata: SDO_SRID of all geometries, spatial indices, and the Oracle SRID. The document length unit will be set according to the selected SRID. However, if the coordinates do not correspond to the SRID, you must transform the geometry in an additional step.

To set the SRID

1. Start the data model administrator.
2. In the data model explorer, select the document node item.
3. Right-click and click Spatial menu ➤ Update Coordinate System.

4 In the Update Coordinate System dialog box, select the SRID, using either of the following settings:

Update Coordinate System Dialog Box	Description
Current Coordinate System	Displays the current coordinate system.
New Coordinate System	Specifies a system that is provided in the Oracle Spatial System user (MD-SYS.SDO_CS_SRS.CS_NAME). Select a system.
WKT	Displays the well-known text (WKT) description of the SRS.

Topobase does not support the update of the SRID for a single feature class or topic.

See also:

- About Coordinate Systems (page 150)

Data Model Administrator: Shortcut Menus

The data structure of the selected document is displayed in the data model explorer. The items are shown in a tree view, starting from the Topobase root node in the following order:

Document (<name>@<datasource>) (root node)

Topics (page 524) (Topics node)

Domains (page 522) (Domains node)

Utilities (Utilities node)

Topologies (Topologies node)

Intersections (Intersections node)

In the data model explorer, you can right-click any object to see a shortcut menu of commands related to that object. If available, further details concerning the selected item are shown in the details pane on the right.

Document Shortcut Menu

The following commands on the document root node shortcut menu affect the whole document. Some of the commands can also be started from the topic or feature class level.

To display the shortcut menu, right-click the document node item in the data model explorer and click a command, as shown in the following table:

Shortcut Menu **Document Node**	**Description**
Empty Topobase	Empties the tables (TRUNCATE) of feature data, but retains the tables and schema.
Create Grants	Creates the required grants for the current Topobase (Oracle commands), in case of trouble shooting.
Set Environment Variables	Reads some settings from TB_INFO and saves them for the session: - Index table space (TB_TMP_MGMT) - name of TBSYS. This command creates all user synonyms on TBSYS; sets all required settings for Job package and recreates the package; creates a context that corresponds to user names; and re-creates ON_LOGON trigger.
Compound	Commands that use sub-commands to process compounds. See also Data Model: Compounds (page 211).
UFID	Commands that use the following sub-commands to process indices: UFID (unique feature ID) and FID (feature ID)
Reset FID Sequence	Resets the FID sequence, so the sequence can start from the beginning. For example, this is necessary, when a document (page 521) has been emptied. You can also reset the FID sequence if the sequence does not match the maximum FID value from the TB_UFID table.

Cleanup UFID Table	Deletes all FIDs from TB_UFID that have no corresponding feature. If there are many unnecessary entries this may improve performance.
Restore UFID Table	Restores missing entries in TB_UFID, for example, if values have been deleted manually.
Spatial >	Command that uses the following sub-commands to process spatial indices.
Insert Or Update Spatial Metadata	Inserts or updates spatial metadata. This is data about the minimum and maximum of X and Y, the SRID, spatial Tolerance and dimension; Stored in TB_DICTIONARY.
Create Or Replace All Spatial Indices	Creates or replaces spatial indices.
Drop All Spatial Indices	Deletes spatial indices.
Check Spatial Config. For System Tables	Writes spatial metadata and spatial indices for TB_PERIMETER and TB_JOB_PERIMETER. Stored in TB_INFO.
Update Coordinate System	Updates the spatial reference ID (SRID). See also Updating the Coordinate System (page 151)
Round Geometry On All Feature Classes	Rounds the geometry. For example, if you have the Round Geometry feature rule enables, and want to adjust imported data. See also Round Geometry.
Triggers >	Command that uses the following sub-commands to process triggers.
All Triggers	Creates or replaces, enables or disables all triggers.
Compound Triggers	See Data Model: Compounds (page 211).
Optimization	Command that uses the following sub-commands for database optimization (DBMS_STATS*). These are important for the Oracle optimizer to generate optimal execution plans.

	For more information refer to *Oracle Spatial Users Guide and Reference*.
Optimize Topobase Schema	Runs the DBMS_STATS.GATHER_SCHEMA_STATS(USER,cacade=>true) command. This procedure gathers (not estimates) statistics for all objects in a schema and on the indexes. We recommend that you optimize the Topobase schema after you have imported a document from a dump file. It is recommended that you run the Optimize command whenever you have extensive changes to the data, such as imports or major updates.
Optimize Feature Classes	Runs the DBMS_STATS.GATHER_TABLE_STATS([USER], [Table], cascade=>true) command for the feature class tables. This procedure gathers table, column, and index statistics.
Optimize Topobase System Tables	Runs the DBMS_STATS.GATHER_TABLE_STATS([USER], [Table], cascade=>true) command for the following tables: TB_UFID and TB_JOB_VERSION. This procedure gathers table, column, and index statistics.
Update Label Texts	Updates all label texts of all label feature classes. See also Label Commands (page 204).
Jobs >	Command that uses the following sub-command to process jobs.
Choose Job	Selects a job. Available for job enabled documents. For commands that process feature classes you must select a job. For example, for Update Area Attribute, or Update Label Texts.
Erase Job History For Live Features	For live jobs, removes all historical elements that are stored in the feature class and the TB_JOB_VERSION table. See also System Tables TB_JOB_* (page 81)
Convert Tables To Feature Classes	Converts Oracle tables to Topobase feature classes. Analyzes the document schema, and opens the Convert To Topobase Schema dialog box. Data Model: Oracle Schema Converter (page 235).

SQL Protocol	This command opens the SQL Protocol window, where you can create a log file of SQL statements that the data model administrator executes during your work. You can control which type of statements you want to log. You can copy or save the statements for future use. Once you close the SQL Protocol window the logged statements are removed.
Refresh	Reloads the data structure tree from the tables, displaying any changes you have made. Use this command or press F5.

Topics Shortcut Menu

Related Topobase feature classes are combined into a collection called Topics. You can edit or create topics and subtopics using shortcut menus, as shown in the following table:

Shortcut Menu **Topic Node**	**Description**
Create Topic	Creates a new topic (main level). Topic Name: Specify a name for the topic. Caption: Enter a caption. The caption is displayed in the various Topobase explorers. Maximum 255 characters.

Right-click the topic item you want to process in the data model explorer and choose a command, as shown in the following table:

Shortcut Menu **Topic Item**	**Description**
Create Topic	Creates a new topic (sub level).
Create Feature Class	Creates a new feature class in the current topic.
	See also Topobase Data Modelling (page 35).
Edit Feature Classes	Sets shared properties for all feature classes of the current topic. These properties can also be

	set individually, either when you create or when you edit the feature class. See also Data Model Administrator: Creating a Feature Class (page 160).
Rename Topic	Renames the topic, changing the caption.
Empty Topic	Clears the topic from all features, emptying all feature classes in the current topic (TRUNCATE); Oracle feature class tables are maintained.
Drop Topic	Drops all feature classes and features in the current topic. All Oracle feature class tables are deleted (DROP).
Update Label Texts	Updates all label texts of the label feature class in the current topic. See also Label Commands (page 204).
Job Enabled	Enables jobs for all feature classes in the current topic. See also Data Model: Jobs (page 215).

Feature Class Shortcut Menus

In the data model explorer, you can right-click the feature class you want to process and choose a command. You can also use shortcut menus for label definition items, as shown in the following table:

Shortcut Menu **Feature Class Item**	**Description**
Create Feature Class	Creates a new feature class under the current one (sub feature class). See also Data Model Administrator: Creating a Feature Class (page 160).
Create Label Definition	Defines a new label definition. This command is only available for label feature classes. See also Label Properties (page 189).
Create Auto Labels	Creates auto labels. You can create labels for features that have auto label definitions but are missing labels.

	For example, if you have disabled label triggers during a data import, you can create the missing auto labels. This operation is not available for label feature classes.
Add Attributes	Adds a new attribute, which means you add a column to the Oracle table. See also Adding Attributes (page 167).
Edit Feature Class	Edits the feature class properties. See also Editing Feature Classes (page 162)
Rename Feature Class	Renames the feature class, changing the caption.
Empty Feature Class	Deletes all features of the feature class (TRUNCATE); Oracle feature class table is maintained.
Drop Feature Class	Drops all features in the current feature class. The Oracle feature class table is also deleted (DROP).
Edit Feature Rules	Defines a set of feature rules (rule base) for each feature class. See also About Feature Rules (page 511).
Compound	Commands to process compound feature classes. See also Data Model: Compounds (page 211).
Spatial	Commands to process Spatial indices
Create Or Replace Spatial Index	Creates or replaces spatial indices.
Drop Spatial Index	Drops spatial indices.
Triggers	Commands to process triggers of the feature class tables.
Triggers	Sub-commands that create or replace, drop, enable or disable all triggers.
Compound Triggers	See also Data Model: Compounds (page 211).
Optimization	Commands that use the following sub-commands for database optimization (DBMS_STATS*). For more in-

	formation see *Oracle® Spatial User's Guide and Reference*.
Optimize Feature Class	Runs the DBMS_STATS.GATHER_TABLE_STATS([USER], [Table], cascade=>true) command for the feature class tables. This procedure gathers table, column, and index statistics.
Update Label Text	Updates the label texts. Available only for label feature classes. See also Data Model: Labels (page 187).
Update Length Attribute	Calculates the length again; available only for feature classes with LENGTH column.
Update Area Attribute	Calculate the area again. Available only for feature classes with AREA attribute.
Intersect With	Calculates the intersection of a feature class with another. Available only for feature classes that are allowed to be intersection primary feature classes (which are at the moment polygon and line string). See also Data Model: Intersections (page 224).
Form Designer	Starts the Topobase form designer, to customize the form. See also Starting Form Designer (page 309)
Job Enabled	Job enables. A hook indicates that the selected feature class is job enabled.

Shortcut Menu **Label Definition Item**	**Description**
Properties	Displays the properties that you can edit. Right-click the label definition. Click Properties. See also Label Properties (page 189).
Drop	Drops the label definition. Right-click the label definition. Click Drop.

Data Model Administrator: Creating a Feature Class

When you create a feature class, you create an Oracle table with certain default columns (attributes). After a feature class is created, you can add any number of additional custom attributes.

When creating a feature class, you always attach it either to a topic or to another feature class, in which case, it is a sub-feature class, or a child class.

To create a feature class, right-click the parent topic, or feature class and click Create Feature Class. The Create Feature Class dialog box is displayed.

After finishing the default settings of a feature class, you can add any required customer attributes. See also Adding Attributes (page 167).

Also, you can register a database view as feature class. See Creating a Feature Class From a View (page 164).

Create Feature Class Dialog Box **General tab**	**Description**
Create New	Selects the creation of a new feature class. On the General tab, enter the name, caption, and type of the new feature class, such as LM_POINT, Sample Point, Point.
Name	Specifies a table name. Enter a name that complies with Oracle format restrictions. If you are planning to use jobs, limit the name to 19 characters. Note: Names for Oracle tables can consist of 30 characters maximum. Since the table names for Topobase for identification of the system provide certain initials and extensions, fewer letters are available for the actual names. Avoid national or special characters such as ä, â, ö, ü, $,%, &.
Caption	Specifies the caption that is displayed in explorer tree views. It can contain up to 255 characters. You can use blanks.
Type	Specifies the type. Select the type in the list. If you select a geometry feature class type, the Geometry Settings

	tab is active. See also Data Model: Feature Class Types (page 172).
Topic	Displays the topic of the feature class. **NOTE** If the Create Feature Class dialog box has been opened from the Create Utility Model dialog box, the Create New Topic command is available.
Feature Class Is Read Only	Sets the feature class to read-only. See also Setting a Feature Class Read Only (page 163)
Create From Existing View	Selects the creation of a feature class from an existing database view. Lists the views that exist in the current document. See also Creating a Feature Class From a View (page 164).
Model Feature Class	Assigns a model feature class. See Data Model: Model Tables (page 184).
Geometry Settings tab	For feature classes with geometry, some attributes for the spatial reference system (spatial column GEOM) are displayed in the Geometry Settings tab. The default settings here were defined during the creation of the Topobase. You can modify these settings for each feature class, but be aware that this requires more modifications to keep the system consistent. Data Model Administrator: Creating a Database (page 144).
Spatial Relate Settings	Controls the behavior of each feature class during spatial selections. If you use a polygon or rectangle to select features, these settings specify the conditions for element selection. See also Data Model: Spatial Relationship Settings (page 233)
Point Feature Class Settings tab	For point feature classes, you can add attributes to store the point's accuracy and reliability. See also Feature Class Type: Point (page 181). Select Digitize With User Defined Orientation if you do not want to use the default orientation. When digitizing

a new point, you will be prompted to specify the orientation.

NOTE You can enable or disable the prompt for the orientation using the Edit Feature Class command. See Editing Feature Classes (page 162).

Editing Feature Classes

In the data model administrator you can modify the following feature class properties:

- **General tab**—Feature class caption. Set or reset the feature class read-only.
- **General tab**—Digitizing Options.
  - Select Prompt For Orientation With Each New Feature if you want to enter the orientation value for each new feature when digitizing in Topobase Client.
  - Select Prompt Z Value With Each New Feature if you want to enter the Z value for each new feature when digitizing in Topobase Client. Only available, if 3D is enabled.
- **General tab**—Model Feature Class. Assigns a model feature class. See Data Model: Model Tables (page 184).
- **Geometry Settings tab**—Edit Spatial Relate Settings.
- **Job Settings tab**—For topics: Assign a perimeter feature class and buffer to all feature classes of the topic. The Job Settings tab is only available if the document is job enabled.
- **Display Settings tab**—Define which feature information will be displayed in the feature explorer. See Customizing the Feature Explorer (page 165).

NOTE The Digitizing Options for the Z value are valid in AutoCAD Map 3D, if you digitize a Topobase feature. If you are digitizing an AutoCAD entity, use the Map 3 D Options dialog box, 3D Modelling tab to enable the Z value input.

See also:

- Setting a Feature Class Read Only (page 163)

- Assigning a Job Perimeter (page 302)
- Data Model: Jobs (page 215)
- Jobs Introduction (page 295)

Setting a Feature Class Read Only

In the data model administrator, you can set a feature class to read-only so that users cannot update and delete the data.

Examples

- Use this option for feature classes that store imported data, which is not managed by the Topobase application.
- We recommend that you use this option when you create a feature class from a view.

For read-only feature classes the following commands are deactivated:

Administrator functions

- Empty Feature Class
- Update length/area attribute
- Compound processing: eliminate/build/convert
- Update Label Texts

Client functions

- Digitizing
- Creating new features
- Dimensioning

NOTE This read-only property is achieved by deactivating Topobase commands in the user interface and not by Oracle rights. Oracle commands, such as insert, are still available from the Oracle side.

See also:

- Data Model Administrator: Creating a Feature Class (page 160)
- Editing Feature Classes (page 162)

Creating a Feature Class From a View

You can include Oracle views in Topobase documents and use them in the same way as the document's own feature class tables. For example, you can use views to combine data from different feature classes in one table. Then, you define reports containing this data or you define position finder searches based on the view. Also, you can use views to stylize features in the Display Manager.

In the Topobase demo data set TB2010_LM you find a sample view LM_V_LANDUSE_TSUR. It is used for stylization.

NOTE Alternatively, you can use Dynamic Stylization to query attributes from multiple feature classes. See Creating Topobase Calculations (page 408).

To create a feature class from an existing view

1. Create the view using any oracle tool. It is recommended that the name starts with a prefix indicating the module in addition to a prefix such as _V indicating that this is a view. For example, LM_V_.
2. Start the data model administrator, and start the Create Feature Class command.
3. In the Create Feature Class dialog box, on the General tab, select Create From Existing View.
4. From the Existing Views list, select the view to register. The name of the feature class is the same as the name of the view, and you cannot edit the name. When you hover over the name of the view, the view definition is displayed.
5. Enter the caption to be displayed in the explorer tree views.
6. Optionally edit the type: By default the feature class type is the same as for the view. For example, edit the type to create a Collection feature class from a view that has been defined as a line type feature class.
7. Select Feature Class Is Read Only. See also Setting a Feature Class Read Only (page 163).

8 Click OK to register the view.

NOTE You can only register a view once. If a view has already been registered as feature class, it is not displayed in the Existing Views list.

View definitions

When you define a view that can be used as feature class, the following conditions must be met.

- The views must contain the attribute FID. A view that does not contain the FID is not displayed in the Existing Views list.
- If you want to be able to edit the features, the view must contain FID and GEOM that come from the same feature class. Then, the view will be registered as a feature class that handles geometries.

Customizing the Feature Explorer

For each feature class, you can define a representation string that will be displayed in the feature explorer. You can display any feature class attribute, or the topic. Also you can display units. By default, the feature explorer displays the FID of the features.

For example, when displaying the results of a road network tracing in the feature explorer, you can specify that the road name is displayed instead of the FID. You define the representation string in the data model administrator.

For example, when displaying the results of a utility network tracing in the feature explorer, you can specify that both the attribute feature class and the related geometry feature class is displayed.

To define the representation string for the feature explorer.

1 Start the data model administrator.

2 Select the feature class.

3 Right-click, and click Edit Feature Class.

4 In the Edit Feature Class dialog box, click the Display Settings tab.

5 On the Display Settings tab, use the options as shown in the following table.

Display Settings tab	Description
Element To Display	Selects an element to display in the feature explorer. Select one of the following. ■ Attribute ■ Feature Class Information
Value	Specifies values for the selected element. The list displays all valid elements. For example, if you select Attribute, the list displays the feature class attributes.
Add	Adds the selection to the String Editor.
String Editor	Displays the selected elements in braces {}. Use the String Editor to edit the representation string. The string is validated immediately.
Examples	To display the area unit, enter: *{AREA} {AREA.Unit.Symbol}*
Feature: 123	To display any static text, enter the text. For example: *Feature: "{FID}"*
Control Cable: 123	*{TB_DICTIONARY.Caption}: {FID}*. Displays feature class caption, a colon, a space, and the FID of the feature.
	To display a related attribute, enter: *WW_MANHOLE_MODEL.ID_MATERIAL*.
	To display a utility feature class along with its related geometry feature class, for the utility feature class, enter: *{TB_DICTIONARY[F_CLASS_ID_ATTR].Caption}*.

Pipe: 123 (Line 124)	For a utility feature class: To display the feature class caption and the FID of both the attribute feature and the geometry feature. *{TB_DICTIONARY[F_CLASS_ID_ATTR].Caption}: {FID_ATTR} ({TB_DICTIONARY.Caption} {FID})*
Control Line 123 (15.60 m)	For a line feature class: To display the feature class caption, FID, length of the line with two decimals, and the unit. *{TB_DICTIONARY.Caption} {FID} ({LENGTH:0.00} {LENGTH.Unit.Symbol})*

The representation string is stored in the system table TB_DICTIONARY.

See also Topobase Client User Guide, section *Feature Explorer*.

Using the String Editor

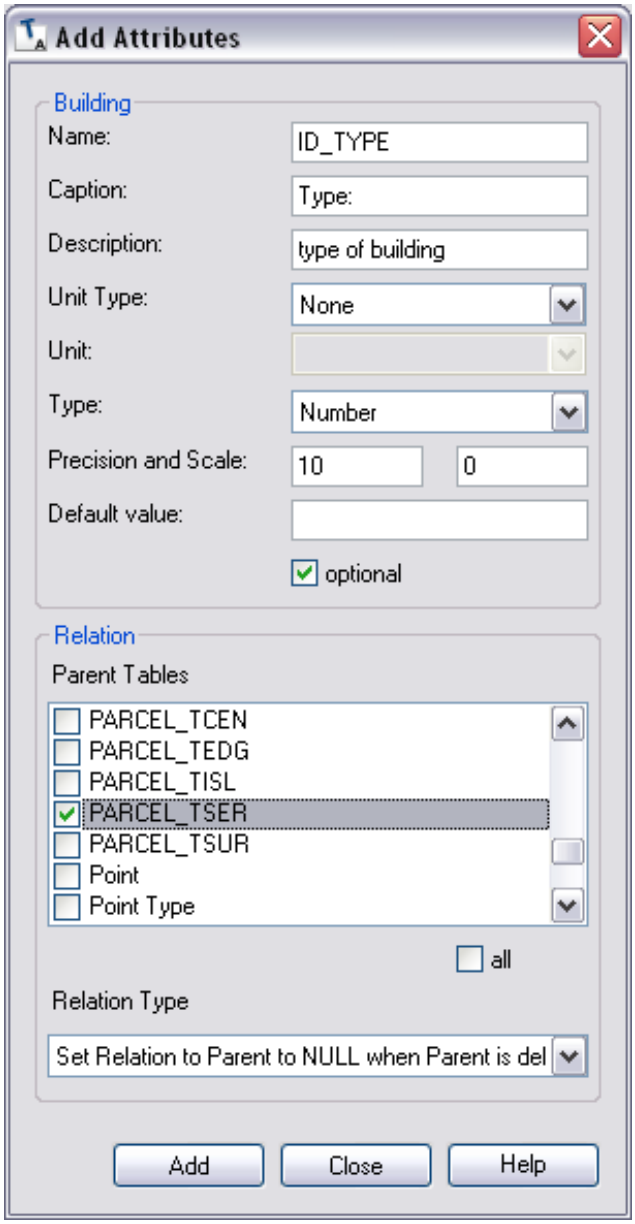
Use the string editor to customize the feature explorer.

- To add any static text, type the text in the string editor.
- To add dynamic text, enter a value in braces {}. Valid values are listed in the Value list.

Adding Attributes

When you create a feature class with the data model administrator, it has default attributes that are specified by the feature class type. However, you can add more attributes.

Right-click the feature class to which you want to add a column, and click Add Attributes to open the Add Attributes dialog box. Add attributes as shown in the following illustration and tables:



Data Model Administrator: Adding an attribute to a feature class.

Add Attribute Dialog Box	Description
Name	Specifies the column name. Enter a name that complies with Oracle format restrictions. If the column relates to data in another table, it is recommended that you use a prefix to indicate this relationship. Example: FID_<parent> or ID_<domain>.
Caption	Specifies the caption to be displayed in the feature class form. You can modify this caption later with the Topobase Form Designer.
Description	Description. The description is shown as tool tip when the user points to the attribute in the detail pane of the data model administrator.
Unit Type	Selects a unit type, if required, such as Angle or Area.
Unit	Selects an appropriate unit, such as Gon Clockwise or Square Meter.
Type Precision and Scale	Specifies type and precision. In the list box, select the appropriate type, subject to Oracle specifications. See also Attribute Type, Precision and Scale (page 170).
Default Value	Specifies a default value. Optionally, assign a default value and specify whether the attribute is mandatory or optional.
Optional	Specifies whether the attribute is optional.
Relation	Attribute relation settings, including Parent Tables and Relation Type. See also Attribute Relation Settings (page 170).
All	Select All if you want to define a multi-relation. See also Attribute Relation Settings (page 170).

For more information refer to *Oracle Administrator's Guide*.

Attribute Type, Precision and Scale

You can specify the attribute type, precision, and scale using the following values.

Attribute Type	Description
VARCHAR2 (n)	Variable length character string -> relations
NUMBER (p,s)	Number having precision p and scale s -> relations
DATE	Valid date range
BFILE	Pointer to binary file on disk
BLOB	Binary large object
CHAR	Fixed length character data CHAR(20)
CLOB	Character Large Object
Long	Long
NChar	National
NCLOB	National Character Large Object
Nvarchar2 (n)	National Character.
Raw (n)	Raw binary data
TimeStamp (n)	Number of digits in the fractional part of the SECOND date time field

For more information refer to *Oracle Administrator's Guide*.

Attribute Relation Settings

For data types "NUMBER" and "VARCHAR2" you can specify a relationship between the attribute and another table. To do this, you create an attribute

that is a foreign key to a primary key in another table (the parent table). These settings are stored in a system table TB_RELATIONS. YOu set the following:

- Multi-relations—Select one or more parent tables to connect the attribute to several feature classes.
- Parent Tables—Select the parent tables from all feature classes of the current Oracle database schema. The relation is directed to the unique FID/ID column of the parent table.
- Relation Type—Specify whether deleting a feature also deletes a related child feature or deletes only the foreign key.

	TB_RELATIONS.DELETE_CHILD
N	The default is Set Relation To Parent To NULL When Parent Is Deleted. Example: For domains (lists), this value must be selected, otherwise the list items are also deleted.
L	Don't Alter The Child When Parent Is Deleted.
D	Delete Feature When Parent Is Deleted. Example: Select this value for slope hatches that have a foreign key to the slope edge. Then the hatch lines are deleted when the edge is deleted.
C	Don't Delete Feature When A Parent Exists.
P	Don't Delete Feature When A Child Exists.

See also:

- System Table TB_RELATIONS (page 94)
- TB_RELATIONS: Parent, Child, Sibling (page 95)

Modifying Attribute Properties

You use the data model administrator to modify attribute properties. At any time, you can modify the Caption and Description. For user defined attributes, you can also modify the Unit Type and Unit attributes.

In the details pane, right-click the attribute and click Edit Attribute.

Deleting Attributes

You use the data model administrator to drop feature class attributes. In the details pane, right-click the attribute and click Drop Attribute. The command deletes all values and drops the attribute from the database table.

Data Model: Feature Class Types

Each feature class consists of one feature table. This table has a fixed basic structure, but you can create additional columns (attributes).

Basic Structure: Every feature has an identifier FID (feature identifier) that is unique for each database schema (document (page 521)).

The feature geometry is saved in Oracle Spatial (standard attribute GEOM).

For more information, refer to the *Oracle Spatial Users Guide and Reference*.

Feature Class Types	Description
Attribute	For attribute tables (features without geometry).
Centroid	For centroids. A special point associated with a polygon that is defined by line strings. A centroid is surrounded by the features of a line string feature class that builds the polygon. Examples: parcels, land use.
Collection	For features with any geometry (lines, point, or polygon). Normally, a feature class contains features of only one geometry type. In a collection feature class, you can store features with different geometry, which means that you can store point features, line features, and so on in one single feature class.
Compound LineString	For polylines. A group of two line string feature classes.

Compound Polygon	For areas. Built as a group of a line string and a polygon feature class.
Dimension	For dimensioning. Consists of related tables: Dimension feature class <name>, line string feature class <name>_L, label feature class <name>_TBL with line string <name>_HL and point feature class <name>_P.
Label	For text. Is related to a parent feature class and therefor can only be created from feature class level. **NOTE** In Topobase Client, in the Display Manager, only label feature classes with the name extension _TBL can be used as Annotation Layers.
LineString	For polylines.
Polygon	For (closed) polygons. Consists of arcs as well as lines or polylines.
Point	For points.

See also:

- Data Model: Domain Tables (page 183)
- Data Model: Model Tables (page 184)

Feature Class Type: Attribute

This feature class is for features without geometry. Examples of attribute feature classes are denotations, lists, and so on.

Default Attributes of Attribute Feature Class

FID	Primary key

Feature Class Type: Centroid

Centroids usually are part of area topologies. Create area topologies using the data model administrator.

See also:

- Data Model: Topologies (page 215)

Default Attributes of the Centroid Feature Class	
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
ORIENTATION	
AREA	Area calculated by topology function.

Feature Class Type: Collection

Feature class for objects that can be either point, line, or polygon.

Default Attributes of the Collection Feature Class	
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
AREA	
LENGTH[2D,3D]	
ORIENTATION	

Feature Class Type: Compound Polygon

A compound polygon is like a polygon, except that the line segments that form the polygon can have different line attributes. For example, a compound polygon can show a building (closed polyline) with one subterranean wall segment that is drawn with a dotted line.

PARENT feature class = Polygon feature class: Data of this feature class is saved redundantly. It is a standard polygon feature class. If the associated line strings do not frame a polygon, no polygon is saved. The essential feature attribute data is stored in the parent feature class.

Default Attributes of the Compound Polygon Feature Class (Parent Polygon)	
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
AREA	Calculated area.
INVALID_GEOMETRY_ERROR	Oracle error code if polygon geometry is not valid.

CHILD feature class = Line string feature class (child, single segments): This is a standard line string feature class with an additional column for a reference to the polygon feature class (FID_PARENT). Its name indicates the related polygon, for example "Single Segment [<compound name>]" or <compound name>_L. Normally the child line strings have few attributes, but can have one to control representation in graphics, such as "LINE_TYPE".

Default Attributes of the Compound Polygon Feature Class (Child Line String)	
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
LENGTH	
FID_PARENT	Foreign key to parent polygon feature.

See also:

- Data Model: Compounds (page 211)

Feature Class Type: Compound Line String

A compound line string is a type of line string in which individual line strings can have different line attributes. The polyline can be drawn with distinct line types for each segment. For example, compound line strings are used for pipelines, where particular segments have special attributes to display the segments at crossing points.

The essential feature attribute data is stored in the parent feature class.

Default Attributes of the Compound Line String Feature Class (Parent Line String)	
FID	Primary key
GEOM	Feature geometry polyline. For more information, see *Oracle Spatial Users Guide and Reference*.
LENGTH	Calculated length of all segments.
INVALID_GEOMETRY_ERROR	State of line in relation to the polyline: (TRUE = valid, Lines not connected)

Child feature class (single segments): Normally, the child line strings have few attributes, but can have one to control representation in graphics, such as "LINE_TYPE".

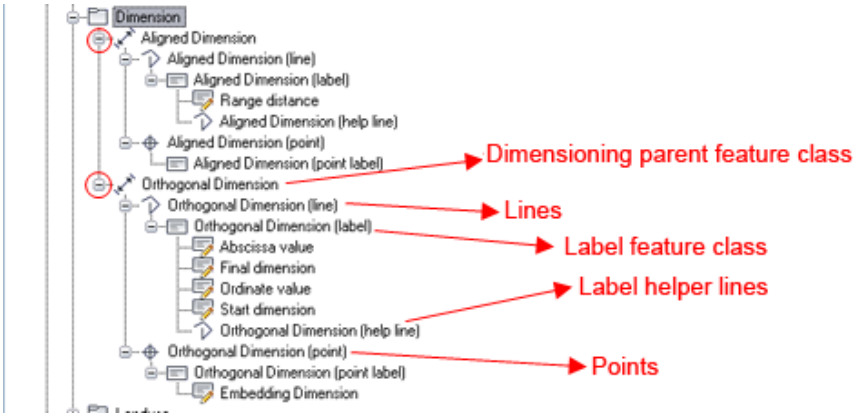
Default Attributes of Compound Line String Feature Class (child line string)	
FID	Primary key
GEOM	Feature geometry polyline. For more information, see *Oracle Spatial Users Guide and Reference*.
LENGTH	Length of a single segment.
FID_PARENT	Foreign key to parent polyline.

See also:

- Data Model: Compounds (page 211)

Feature Class Type: Dimension

A Topobase dimensioning consists of several related feature classes. When you add the Dimension Extension to a document, the dimensioning feature classes are created, as shown in the following illustration. See also Document Settings (page 7).



Topobase Data Model: A dimensioning feature class consists of several related feature classes.

Dimension Parent: This attribute table does not store any geometry.

Dimension Table	Description
	<dimension>
FID	Primary key
DIMENSION_TYPE	Type of dimensioning: AA = Aligned Abscissa Only AT = Aligned Text Only AC = Aligned Complete OC = Orthogonal Complete OE = Orthogonal Extension Baseline

Dimension Children: These tables store the geometry.

Dimension Point Child	Description
	<dimension>_P
	Stores the points at the start and endpoints of the measure line, such as abscissas and ordinates.
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
FID_PARENT	Foreign key to dimension parent table.
POINT_TYPE	AS = Start of orthogonal abscissa. AE = End of orthogonal abscissa. AE_ = End of orthogonal abscissa, not to be displayed. OS = Start of orthogonal ordinate. OE = End of orthogonal ordinate. S = Start of aligned line, abscissa. E = End of aligned line, abscissa.

Dimension Line Child	Description
	<dimension>_L
	Stores the measure line.
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
FID_PARENT	Foreign key to dimension parent table.
LENGTH	
LINE_TYPE	AL = Aligned line, abscissa. AL_ = Aligned line, abscissa, not to be displayed. HL = Help line for dimension.

	HL_ = Help line for dimension, not to be displayed. OO = Orthogonal ordinate. OA = Orthogonal abscissa. OAE = Orthogonal abscissa extension. OAB = Orthogonal baseline abscissa.
OFFSET	Stores the offset of the measure line.
MEASURED_A	Measured dimension of aligned line or orthogonal abscissa.
MEASURED_O	Measured dimension of orthogonal ordinate.
CALCULATED_A	Calculated dimension of aligned line or orthogonal abscissa.
CALCULATED_O	Calculated dimension of orthogonal ordinate.
Label	
	<dimension>_TBL
	The label feature class stores the measure label texts. See also Feature Class Type: Label (page 180)
Dimension Label Line	Description
	<dimension>_HL
	Stores lines that are related to labels, for example lines that point to the feature where the label belongs to.
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
FID_PARENT	Foreign key to dimension label table.

See also:

- Data Model: Dimensioning (page 205)

Feature Class Type: Label

A label feature class can only be created from the feature class level. Each feature class can have only one label feature class.

NOTE In Topobase Client, in the Display Manager, only label feature classes with the name extension _TBL can be used as Annotation Layers. See also Styling Annotation Layers (page 398).

The default attributes of a label feature class are as follows:

Default Attributes of the Label Feature Class

FID	Primary key
FID_PARENT	Foreign key to parent table's FID
GEOM	Label geometry = insertion point. For more information, see *Oracle Spatial Users Guide and Reference*.
HORIZONTAL_ALIGNMENT	Horizontal alignment of the label. The default is Center.
LABEL_DEF_ID	Foreign key to TB_LABEL_DEF.ID.
LABEL_TEXT	Text of the label.
ORIENTATION	Text orientation.
ORIGIN	This value is no longer used. Use alignment attributes instead. See also Label Properties: ORIGIN conversion (page 199)
PRE	Label prefix.
SUF	Label suffix.

VERTICAL_ALIGNMENT	Vertical alignment of the label. The default is Halfline. See also Label Properties: Alignment (page 197)

A label feature class always belongs to a parent feature class (relation FID_PARENT). Normally a label feature class has no additional columns and the application user (page 526) will never edit it. The attributes of a label feature class are generated when a user places a label.

See also:

- Data Model: Labels (page 187)
- Creating a Label Definition (page 113)

Feature Class Type: Line String

Default Attributes of the Line String Feature Class	
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
LENGTH	

See also:

- Feature Class Type: Compound Line String (page 176)

Feature Class Type: Point

Default Attributes of the Point Feature Class	
FID	Primary key
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.

ORIENTATION	Orientation of the point geometry.
Z	Height, for example, terrain elevation.
QUALITY	
Optional Point Feature Class Attributes	When creating a point feature class, you can optionally add the following attributes. For example, these attributes are used by survey applications to store accuracy and reliability values, which are usually set automatically by the application. For example the Topobase Construction functions automatically set values. See also Data Model Administrator: Creating a Feature Class (page 160) See also Setting COGO Application Options (page 19)
TB_POINTNUMBER	Stores a point number.
TB_ACCURACY_POSITION	Stores the accuracy of the ground representation.
TB_ACCURACY_HEIGHT	Stores the accuracy of the height representation.
TB_RELIABILITY_POSITION	Stores the reliability of the ground representation. This value indicates the quality of the measuring method.
TB_RELIABILITY_HEIGHT	Stores the reliability of the height representation. This value indicates the quality of measuring method.
TB_POSITION_RELIABLE	Stores the name of the attribute that specifies whether the ground representation of the point is reliable or not. The attribute is of type check box (1=yes, 0 = no).
TB_HEIGHT_RELIABLE	Stores the name of the attribute that specifies whether the height coordinate of the point is reliable or not. The attribute is of type check box (1=yes, 0 = no).

Feature Class Type: Polygon

Default Attributes of the Polygon Feature Class	
FID	Primary key

GEOM	Polygon geometry. For more information, see *Oracle Spatial Users Guide and Reference.*
AREA	The area is calculated by Topobase.

See also:

- Feature Class Type: Compound Polygon (page 175)

Data Model: Domain Tables

You can create and edit domain tables in the Topobase data model administrator. In the data model explorer, right-click the Domains item and click Create Domain.

NOTE The suffix _TBD is added automatically to the Domain Table Name you enter. Domain tables have a set of standard attributes. We recommend that you do not add attributes to a domain table.

The data model explorer displays the most important domain values as shown in the following table.

Domain Tables Attribute	Description
Key	Displays the unique key. <domain_TBD.ID>
Value	Displays the value. <domain_TBD.VALUE>
Short Value	Displays the short value. <domain_TBD.SHORT_VALUE>
Active	Specifies that the record is active. Selected by default. Clear to hide or de-activate the record in the domain list.

Editing a Domain Table

To edit a domain table

1. In the data model explorer, expand the Domains item.
2. Click the domain table that you want to edit.
3. In the grid view on the right, edit the table.Click the record you want to edit and enter the new values.
4. To add a new record, right-click anywhere in the grid and click Create.

If you add a new record, the Key value is filled automatically with the next free ID (unique).

To delete a record, select it in the grid, right-click and click Delete.

NOTE If you want to add or update a large number of domain values, open the feature class form, export the domain table to Excel and edit and re-import the data.

See also *Topobase Client User Guide*

Data Model: Model Tables

You can use model driven attributes to provide a consistent and smart way of data acquisition. Using model tables, you constrain attributes based on other attributes. These attributes are called model driven.

Example: To use a model table for diameter and material

1. In the model table, specify valid combinations of diameters and material, such as Iron - 50; Iron - 100, Concrete - 400, Concrete - 300.
2. Assign the model table to the main feature class, such as Pipe.
3. In Topobase Client, in the main feature class form, use the Model Selector control to enter the attribute values, or to open the Model table.

The following instructions show the concept of model tables. For example, in Topobase Gas, you use the Pipe Model table, to acquire Pipe attributes.

To use a model table for data acquisition

1. Start Topobase Client, and open a workspace with access to the Gas demo data set.
2. In the document explorer, expand the topic Pipe. Right-click Pipe, and click Show Form.
3. In the Pipe form, select a Pipe.
4. Besides Model, click to show the Pipe Model form.

   The Pipe Model form consists of two sections: At the top is the filter section for the model attributes. Use the horizontal scroll bar in the middle of the form to view all attributes. The lower section displays the models (model table records) that match the current feature. For example, if not all model related attributes of the main feature are filled, several models would be applicable.
5. Use the filter area to search for any other models.
6. In the data grid, select a model, and click Apply.
7. The main feature class changes into Edit mode. Check the values, and click Update to insert the attribute values as specified in the selected model.
8. In the main feature class, click again. The model table filter is set to the model you just applied. That means that the main feature attributes match the model attributes. To apply another model, use the filter area.
9. Close the Pipe Model form.

Alternatively, you select a model from the model list. In the Pipe form, on the General tab, select a model from the list. The list displays the Model Name (MODEL_NAME). If no model name is assigned, the list shows an empty entry. If the model table does not contain the attribute MODEL_NAME, the value of the VALUE attribute is shown. If no attribute VALUE is available, the list shows the FID.

To assign a model table to an existing feature class

1. Start the data model administrator.
2. Select the main feature class.

3 Right-click, and select Edit Feature Class.

4 In the Edit Feature Class dialog box, click the General tab.

5 Under Model Feature Class, select the attribute feature class you want to assign to the feature class.

6 Click OK.

You can use any attribute feature class as model table. If the model feature class contains attributes that are not in the main feature class, you can optionally add these attributes.

In the parent feature class form, the model table can be accessed using the Model Selector. Use the Form Designer to add the Model Selector control.

To add the Model Selector.

1 Start the Form Designer.

2 Select the parent feature class.

3 Click Designer.

4 See Form Designer Reference (page 309).

Updating Model Tables

Topobase 2009 introduced a flexible management of the model feature classes (model tables). In previous versions, some Topobase applications used model tables named <name>_MODEL. When you open a document that has the old model table structure, the system updates existing model tables, so they support the new concept of model driven attributes.

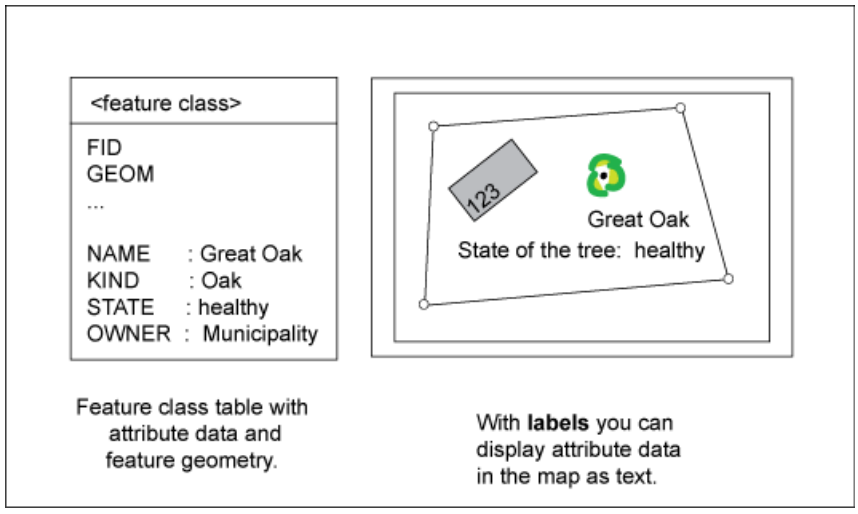
The data structure update performs the following steps.

- **Main feature class**—The attribute FID_MODEL that links to the model table, will be removed.
- **Main feature class**—All attributes of the related model table will be added. The values of these model attributes are filled with the model information of the model table, so that no data is lost.
- **Main feature class form**—A model selector will be added. Use the model selector to access the values of the model table.
- **Model table**—The attribute NAME_NUMBER is renamed to MODEL_NAME.

NOTE The model feature class is still being used, according to the new concept. However the model attributes are now stored in the main feature class, and you use the model table to support data acquisition.

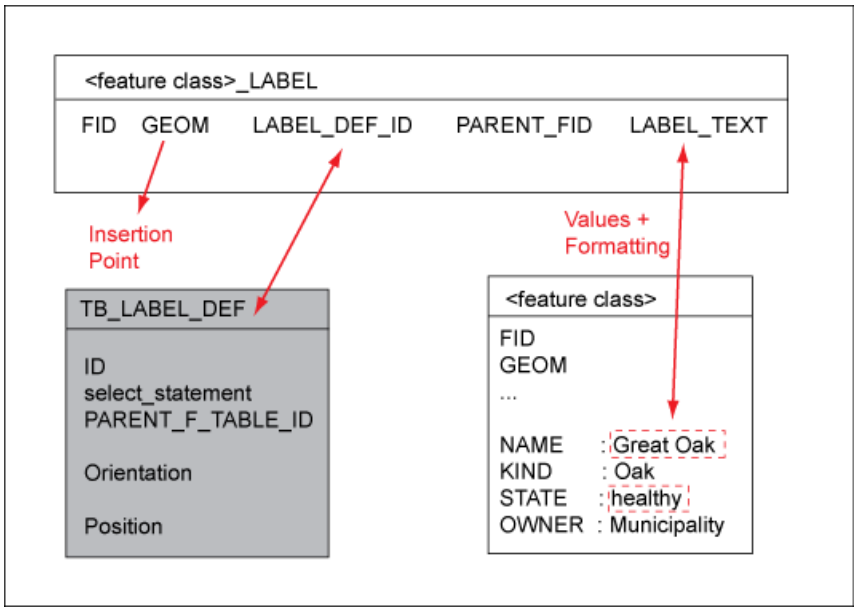
Data Model: Labels

In Autodesk Topobase applications, you can display any attribute data of a feature as label text, and align the label consistently in relation to the displayed feature. Labels are stored in a special label feature class <feature class>_TBL that is a child of the parent feature class whose attribute data is to be displayed. Each feature class can have only one label feature class.



Topobase Data Model: Labels display attribute data as text.

Example: The label feature class TREE_TBL contains text that displays species and vitality of trees. The parent feature class that stores the attribute data is TREES.



Topobase Data Model: Relations between feature table and label feature class. The attribute itself (name of the tree = Great Oak) is saved with the feature. To display this value as text in graphics, generate labels.

The label definition consists of a select statement to query the attribute values from the database, in addition to rules about how to display the text (offset, insertion point). The select statement is executed each time the feature is regenerated. Therefore, the label always displays the current value and if you update a value in the database, labels that use this data are updated.

You can use the data model administrator to create and define label feature classes and custom labels. Use Topobase Client or Web to create and position labels for selected features.

NOTE Each feature class can have only one label feature class, but can have any number of label definitions.

See also:

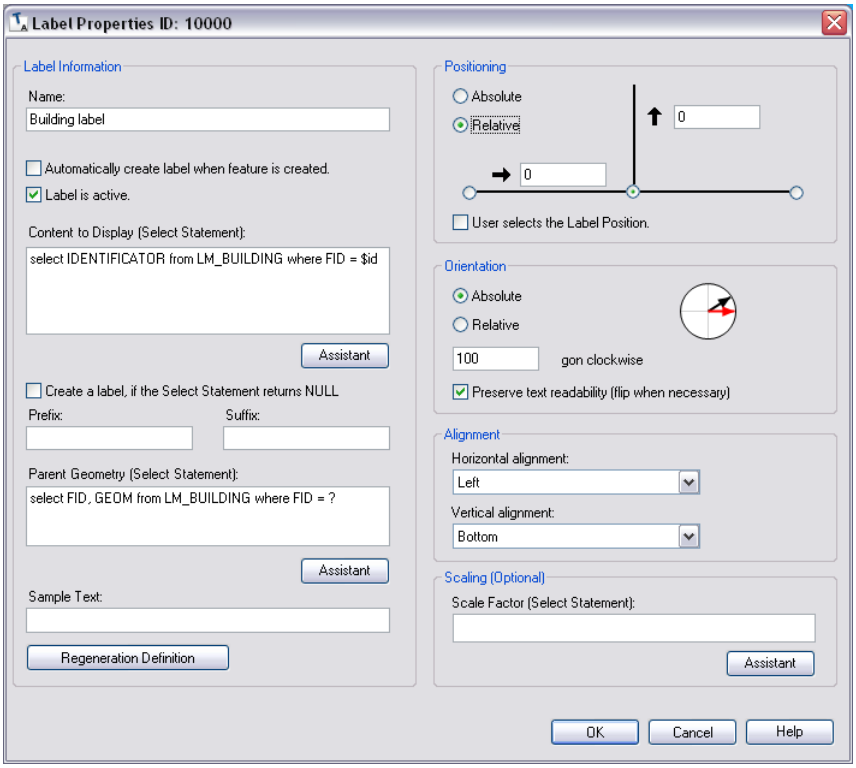
- Creating a Label Definition (page 113)

Label Properties

Label properties define how the label text is selected from the database and how the label is positioned.

To edit label properties

1. Start the data model administrator.
2. In the data model explorer, select the label definition.
3. Right-click and click Properties.



Topobase data model administrator: Label Properties dialog box

Label Properties: Label Information

Label Properties Dialog Box **Label Information**	**Description**
Name	Default is <label feature class name> Label: You can change the default value to any clear name. The label name is displayed in the explorer tree views and the Choose Label Definition dialog box. It is saved in TB_LABEL_DEF.NAME. The label name can have up to 255 characters and may contain blanks (and special characters).
Automatically Create Label When Feature Is Created	Creates a label upon feature creation. These labels are called Auto Labels.
Label Is Active	Specifies that the label is displayed by the label placer. Selected by default. Clear this check box if you do not want to display this label for selection with the label placer (page 523).
Content To Display (Select Statement)	Specifies the text to be displayed by an SQL select statement. Default statement: `select FID from <table name> where FID = $ID` This statement returns the FID. The easiest change is to replace the column name FID with the name of another attribute that you want to display as label text. See also Label Select Statements (page 200).
Assistant button	Opens the SQL Assistant. Optionally, you use the SQL Assistant to create SQL expressions. See Creating SQL Expressions (page 103).
Display NULL If The Select Statement Returns NULL	Creates an "empty" label even if there is no corresponding value in the feature table. If necessary, you can update the data for these labels later, using the Update Label Texts command. If this check box is cleared and the select statement returns nothing, no label record is created.

Prefix/Suffix	Standard text included before or after the selected text, such as an abbreviation or a unit of measurement. For stylization in Display Manager see Displaying Label Features with Prefix and Suffix (page 398). Alternatively you can use the CONCAT statement in the label SQL select statement.
Parent Geometry (Select Statement)	Defines the auto labels for attribute feature classes. See also Label Geometry Select Statements (page 202)
Sample Text	Optionally displays a sample text. Enter an example of label text.
Regeneration Definition	Defines regeneration rules for label definitions. See also Label Properties: Regeneration Definition (page 191)

Label Properties: Regeneration Definition

You can define regeneration rules for label definitions. Normally, labels are updated (regenerated) automatically when an attribute is modified. However, this only updates the attributes of the main feature class, which is the feature class the label definition belongs to.

Regeneration rules are applied, if the label definition contains text that is stored in a secondary related feature class. For example, a label definition of the building feature class that displays informations that is stored in the street feature class.

In the Label Properties dialog box, click Regeneration Definition to define regeneration rules.

Label Regeneration Definition dialog box	Description
Regeneration Rules	Selects a rule. The default name for a rule is the name of the current label definition.
Create New Rule	Creates a new rule.
Selected Regeneration Rule	Displays the properties.

Feature Class	Selects the related feature class (secondary feature class). If a feature of this feature class is modified, the regeneration rule is applied.
If One Of The Following Is Changed	Displays the attributes of the selected feature class. Select an attribute whose modification will cause the regeneration rule to be executed.
Run This Statement To Get The Label Parent FIDs	Defines a selection to get the label parent ID. The default statement contains the main and the secondary feature class. The alias $id has to be used.
Activate This Rule	Activates the rule.
Assistant button	Opens the SQL Assistant. Optionally, you use the SQL Assistant to create SQL expressions. See Creating SQL Expressions (page 103).

NOTE If you define a label regeneration rule, some RegnerateLabel feature rules are added to the secondary feature class. You can view these in the feature rule editor.

IMPORTANT Do not deactivate these rules in the feature rule properties. Use the Activate This Rule option in the Regeneration Definition dialog box to deactivate or activate the rule.

Label Properties: Positioning

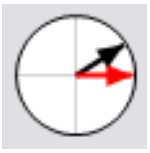
Positioning	Description
Absolute Relative	See also Label Properties: Positioning and Orientation (page 195)
User Selects The Label Position	Specifies that when labels are created, the user is prompted to enter the label position by mouse click. Text box insertion point and orientation are used as defined.

Label Properties: Orientation

Orientation	Description
Absolute Relative	See also Label Properties: Positioning and Orientation (page 195)
Preserve Text Readability (Flip When Necessary)	Specifies that the labels are rotated so they are readable. This option can be required if the label positioning properties create upside down labels for some features. When the label is created, the orientation is either calculated relative to the feature geometry or set to the absolute value specified in the label properties. The orientation of upside down labels is greater or equal to 200 Gon (180 degree). Select this option to rotate those labels at 200 Gon, to make them readable, and to store the adjusted orientation in the database. **NOTE** This option does not flip existing texts, if you change the map units or the label styles. It applies to the label orientation that is stored in the database. **NOTE** Do not select this option for flow direction arrows. The label (symbol) orientation would not show the correct flow direction. **NOTE** If this option is not selected, you must select the option "Labels Are Fixed (Not Dynamic)" in the Style Editor, if you want to display labels that are upside down. See also Styling Label Features (page 393). If you deselect both options "Labels Are Fixed (Not Dynamic)" and "Preserve Text Readability (Flip When Necessary)", labels that are upside down would not be displayed.

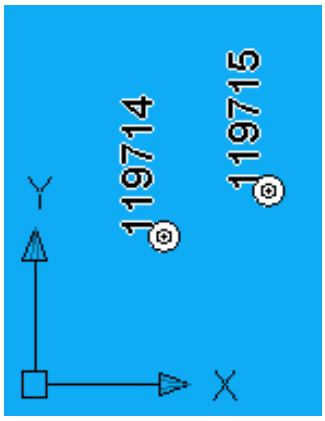
In the Orientation section, a circle indicates the label orientation.

Example—Absolute orientation = 100 Gon.

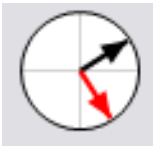

The red arrow indicates the label orientation. For absolute orientation it references the Base Angle = North. For example, select Absolute and enter 100 Gon to display the labels horizontally. This is the default setting.

Example—Absolute orientation = 0 Gon.

The red arrow indicates the label orientation. For absolute orientation it references the Base Angle = North. For example, select Absolute and enter 0 to display the labels that are aligned with North.



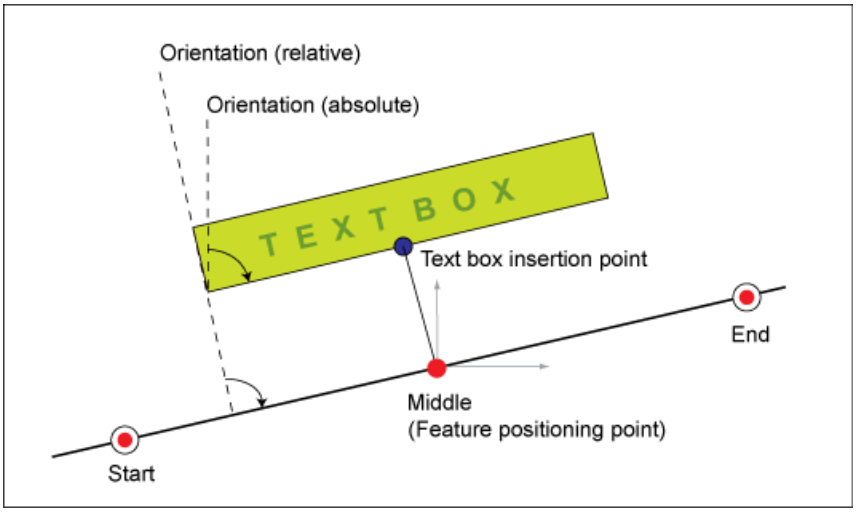
Example—Relative orientation = 100 Gon.


The black arrow indicates the parent feature orientation. The red arrow indicates the label orientation. For relative orientation it references the parent feature orientation. For example, select Relative and enter 100 Gon to display the labels perpendicular to the parent feature.

NOTE In Topobase documents the Base Angle is North. In Topobase Client, when you are prompted to digitize the label orientation, the Base Angle of the drawing will automatically be set to North, so you can digitize the appropriate value. See also Map Drawing Units (page 399).

Label Properties: Positioning and Orientation

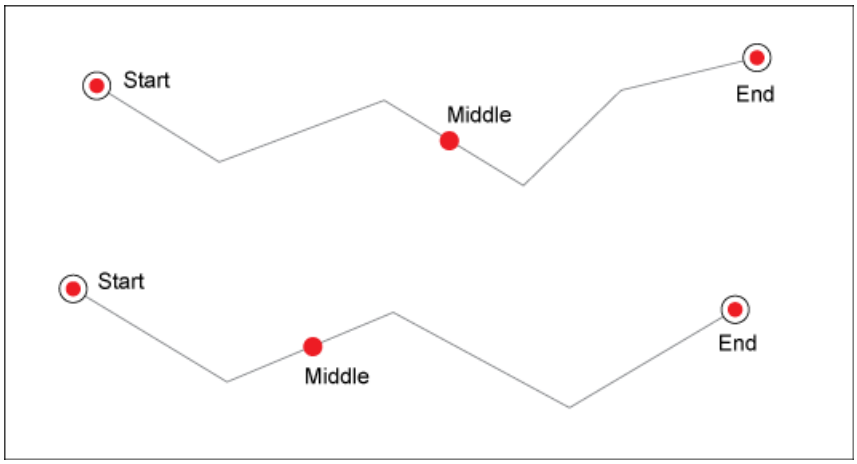
Use the label positioning and orientation settings to specify the label location relative to the feature geometry. These settings establish the location of a text box insertion point with reference to a start, middle, or endpoint of the feature.



Topobase Data Model: Defining label positioning and orientation.

Positioning values specify the location of the text box insertion point, relative to the feature.

You click the start, middle, or end point on the line to specify a feature positioning point (reference):



Topobase Data Model: Defining the feature reference for a label

- For lines with more than one segment, middle is the middle of the mid segment or the segment prior to this if the number of segments is even.
- For points the reference is the point.
- For polygons the reference is the center.
-

In the Label Properties dialog box, for positioning offset values, select one of the following:

- Absolute: Specifies the text box insertion point position relative to true north and east.
- Relative: Specifies the text box insertion point position relative to a line perpendicular to the feature direction.

Enter northing and easting values to specify the offset from the feature positioning point to the text box insertion point.

- For example, absolute offset values for the figure above are: approximately dY = - 1 (easting); dX = 2 (northing).
- For example, relative offset values for the figure above are: approximately dY = 0 (easting), dX = 3 (northing).

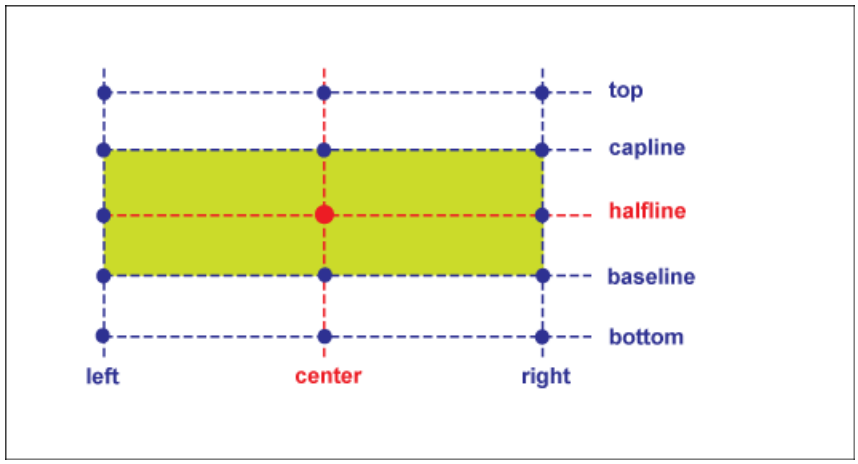
Orientation values specify an angle of rotation for the text box. Select one of the following:

- **Absolute orientation**: References to true north. For example, values for the figure above are: approximately 90 Gon.
- Relative orientation: References to a line perpendicular to the feature geometry. For example, values for Figure 22 are approximately 100 Gon. For points, the value refers to a possible stored orientation.

Label Properties: Alignment

For the Alignment properties, you can specify the location of the text box insertion point (origin) in relation to the text box. These values are also used in the Display Manager stylization.

Alignment	Description
Horizontal Alignment	Specifies the text box insertion point horizontally related to the text box. Select Right, Center, Left. Default value is Center.
Vertical Alignment	Specifies the text box insertion point vertically related to the text box. Select Bottom, Baseline, Halfline, Capline, Top. Default value is Halfline.

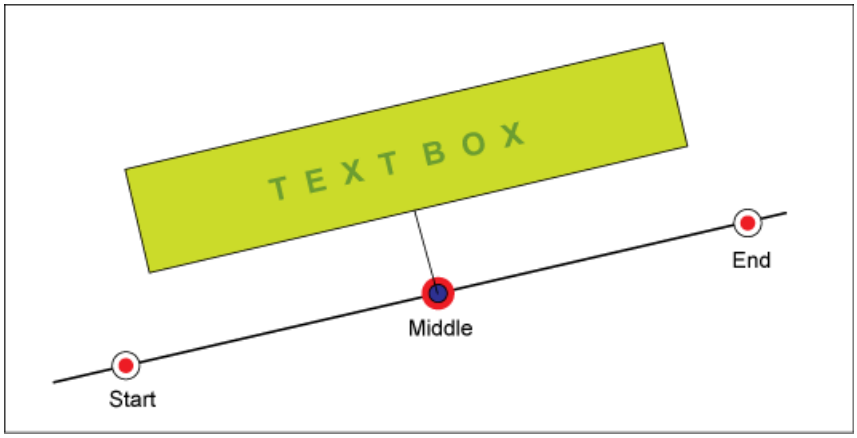


Alignment, is set to the defaults Center and Halfline

Example: Positioning a label parallel to a line

- Feature positioning point: Middle.
- Horizontal Alignment: Center.
- Vertical Alignment: Bottom.

With these properties, the text is always positioned in the middle of the line, regardless of its length.



Topobase Data Model: Defining the label location along a line.

See also:

- Styling Label Features (page 393)
- Overview of the Display Manager (page 387)

Label Properties: Scaling

Scale select statements are used to position a label text regarding the scale of a possible map. A scale select statement can use complete SQL select statements and returns a numerical value of the scale (for example 500 for 1:500 scale), which is taken into consideration using relative offset when calculating label position.

Scaling (Optional)	Description

Scale Factor (Select Statement)	Optionally, defines a scale factor select statement in order to position the label text relative to the scale.
Assistant button	Opens the SQL Assistant. Optionally, you use the SQL Assistant to create the scale factor select statement. See Creating SQL Expressions (page 103).

Label Properties: Label ID

Each label definition has a unique identifier that is generated when you create a new label definition. It is shown in the title bar of the Label Properties dialog box, for example Label Properties ID 1000.

NOTE The LABEL_ID is a group identifier that marks label definitions belonging to the same parent feature class.

For better overview, and to separate common application labels from individual labels, all labels created with the data model administrator by any user (page 526) have IDs starting with 10000. This is also important for updates. Labels with IDs less than 10000 can be updated when installing a new Autodesk Topobase version. Labels with IDs greater than 10000 are not affected by updates in a newer version.

See also:

- System Table TB_LABEL_DEF (page 88)

Label Properties: ORIGIN conversion

In previous Topobase versions, the text box insertion point was specified using a 5x5 point matrix related to the text box. The value was saved in the label feature class attribute ORIGIN. This attribute is no longer used. The attribute is replaced by the horizontal and vertical alignment property (3x5 matrix). The document update automatically converts existing ORIGIN values, as shown in the following table.

ORIGIN	Alignment
1,2	Bottom, Left

3	Bottom, Center
4,5	Bottom, Right
6,7	Baseline, Left
8	Baseline, Center
9,10	Baseline, Right
11,12	Halfline, Left
13	Halfline, Center
14,15	Halfline, Right
16,17	Capline, Left
18	Capline, Center
19.20	Capline, Right
21,22	Top, Left
23	Top, Center
24,25	Top, Right

Label Select Statements

When you define labels, the select statements are the basic part of the definition. You can define or edit select statements with the data model administrator user interface (UI) using the Create Label Definition dialog box or the Label Properties dialog box. You either type the select statement directly in the input box, or you use the SQL Assistant to create the SQL expression. See also Creating SQL Expressions (page 103).

The select statements are saved in TB_LABEL_DEF.SELECT_STATEMENT.

The data model administrator uses the following default select statement to return a feature identifier (FID).

```
select FID from <table name> where FID = $id
```

This select statement can be changed or extended, for example, using static commentaries, additional queries, formats and so on.

In the following expression, the f is the alias for the feature class table (that is the f behind the table name, separated by a blank).

```
select f.fid from <table name> f where f.fid = $id
```

It is important, that each select statement must contain the placeholder $id. During label creation, this term is replaced by the FID of the parent feature, whose attribute data is to be displayed.

The following example from the Land demo data set extends the default select:

```
„select 'Point number: '|| f.name_number||'Orientation: '||round
(f.orientation,2) from LM_POINT f where f.fid = $id“
```

Examples of useful SQL additions:

- Commentary text (with apostrophes)
- Query the point number (f.name_number)
- Query the orientation (f.orientation)
- SQL conform formatting command (round, n, m)

When defining select statements, note the following definition rules:

No.	Definition Rules for Label Select Statements
1	Static words or formats must be entered with apostrophes ' ' and separated by two pipe signs || from one another and from column names.
2	You can use SQL formatting statements, such as TRUNC, or ROUND. See also Label Formatting With SQL (page 202).
3	If an alias is defined, it must be used. Otherwise, this selection cannot be performed.
4	For label definitions that use numerical fields: if there is no record (NULL), do not write "0" when used, for example: `decode (year),Null,'',Year`
5	The expression $s suppresses a description when repeated.

If a selection provides several records with repeating data, for example, a manhole name when selecting inlets, the '$s' command immediately after the field name suppresses description repetition.

Label Formatting With SQL

You can use SQL conform format statements such as ROUND and TRUNC.

Examples:

... select round (area,0) from ...

... select round (area,2) from ...

... select trunc (area,-2) from ...

Label Geometry Select Statements

Geometry select statements are used to define auto labels for attribute feature classes. Since attribute features do not have geometry, you can use a geometry select statement to query the geometry of the parent feature.

The Topobase utility applications use geometry select statements to generate auto labels for the network objects. Example:

```
SELECT FID, GEOM FROM WA_LINE WHERE FID_ATTR = ?
```

If you create a label definition, the sample geometry select statement is:

```
select FID, GEOM from [parent] where FID = ?
```

Note, that every geometry select statement has to be finished by the expression "..where FID = ?)".

See also:

- Defining a Label for a Utility Object (page 292)

Sample Labels

Example—To inscribe the area of a polygon feature class, such as LM_BUILDING.

Modify the select statement as follows:

```
select round (f.area,2) from LM_BUILDING f where f.fid = $id
```

After the “SELECT”, you must specify the column <area>, which contains the data that you want to inscribe. You can use SQL format commands as "ROUND" to specify the number of digits as in the example above. Separator is a ",". Note, that you can use the character following the table name <LM_BUILDING f> as an alias in the SQL statement.

Example—To inscribe the building type, which is saved in a related domain table, you must include this table and relate it to a WHERE clause.

select u.value from LM_BUILDING f, LM_BUILDING_TYPE_TBD u where f.id_type = u.id and f.fid = $id

When you save the label definition, the syntax of the SQL statement is checked. When you use the SQL Assistant, you can execute and check the SQL statement at any stage of the definition.

If you execute the SQL statement separately, delete the "f.fid=$id" at the end of the statement.

In case you make any modifications to a select statement, you can update the existing label text:

- In the data model explorer, select the label feature class.
- Right-click and click Update Label Texts

Working with Complex Labels

In a label definition select statement, you can use the CONCAT_LIST function to define multiline labels, such as in the gas demo data set.

To define a multiline label

1. Start Topobase Administrator and open a workspace that contains the gas data model.
2. In the data model explorer, under the topic Pipe, select the feature class Pipe Label.
3. Right-click and select Create Label.
4. In the Label Properties dialog box, under Content To Display (Select Statement), enter the following select statement:

```
select concat_list(CURSOR(
select 'Value: '||c.value||'\n'
||'Diameter: '||a.DIAMETER_NOMINAL||'\n'
||'Date of Installation: '||to_char(a.date_installation,'YYYY')
from ga_pipe a, ga_material_tbd c
where a.ID_MATERIAL = c.id and a.FID = $id
))myCursor from dual
```

Label Commands

Shortcut Menu **Label Item**	**Description**
Update Label Texts	Updates the label texts in the drawing after you change a label definition or the corresponding attribute in feature tables. Updates the LABEL_TEXT attribute of the label feature class tables (***_TBL.LABEL_TEXT) by performing the select statements again. This command can be performed for all label texts, either in the current document (page 521) or in a topic or in a feature class. This depends on the tree view level from which you start the command.

Data Model: Utilities

You can create and edit utility data models in the Topobase data model administrator. Right-click the Utilities item and click a command.

Utility data models provide network components for the Topobase utility applications, such as Water, Gas or Waster Water.

Shortcut Menu **Utility Item**	**Description**
Create Utility Model	Creates a utility model (network). The concept is to separate the infrastructure from networks. See also Utility Model Introduction (page 285).

Data Model: Dimensioning

Topobase provides special feature classes to store dimensioning in the database. A dimensioning consists of several related feature classes. A dimensioning feature class (parent attribute feature class) is related to several child feature classes that store the geometry components, such as points (symbols) and lines, and labels.

The default topic Dimension contains two dimensioning feature classes, Aligned Dimension, and Orthogonal Dimension. Dimensioning either uses the calculated dimension or the measured dimension.

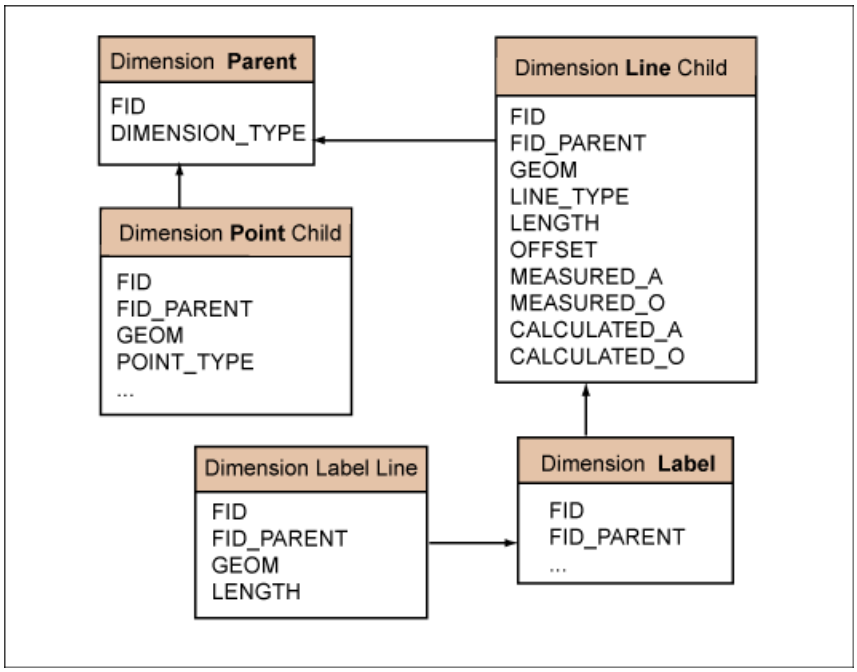
See also About Aligned Dimensioning (page 206).

See also About Orthogonal Dimensioning (page 208).

Use the Document Settings to add the dimensioning feature classes to any document. See also Document Settings (page 7).

We recommend that you use the Dimension Extension, and the default dimension feature classes. Optionally, you can create dimensioning feature classes manually in the data model administrator.

NOTE Autodesk provides a display model for the Dimension extension. Copy the files from <topobase_client>\Template\Extensions\Dimension\DisplayStyle to your Repository Base Path. See also Working with Display Models (page 379).



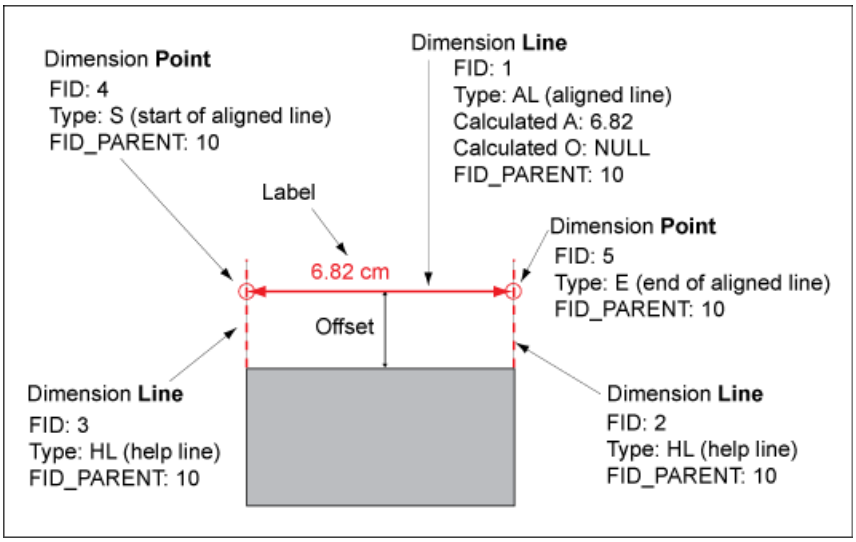
Topobase Data Model: Dimensioning feature classes. The dimension parent feature class is related to the geometry feature classes, Dimension Point Child and Dimension Line Child.

See also:

- Creating and Using Dimensions (page 127)
- Document Settings (page 7)
- Feature Class Type: Dimension (page 177)

About Aligned Dimensioning

Topobase provides several types of aligned dimensioning. Depending on the dimensioning type, the system uses the LINE_TYPE attribute and the POINT_TYPE attribute to distinguish the points and lines. In the display models, you use these attributes to stylize the dimensioning layers.



Topobase Data Model: Components of an aligned dimension.

An aligned dimensioning consists of points and lines, as shown in the following table.

Aligned Dimensioning type	Description
AC = Aligned Complete	Default aligned dimensioning including lines and labels. Consists of: ■ Line of type AL: abscissa. ■ Lines of type HL: ordinate help lines. ■ Point of type S: start point of abscissa. ■ Point of type E: end point of abscissa.
AA = Aligned Abscissa Only	Aligned dimensioning where only the abscissa is drawn, but no ordinate lines. Consists of: ■ Line of type AL: abscissa. ■ Lines of type HL_: ordinate help lines. Notice the special line type. In the Display Manager, use the line type HL_ to hide this helper line in the drawing. ■ Point of type S: start point of abscissa.

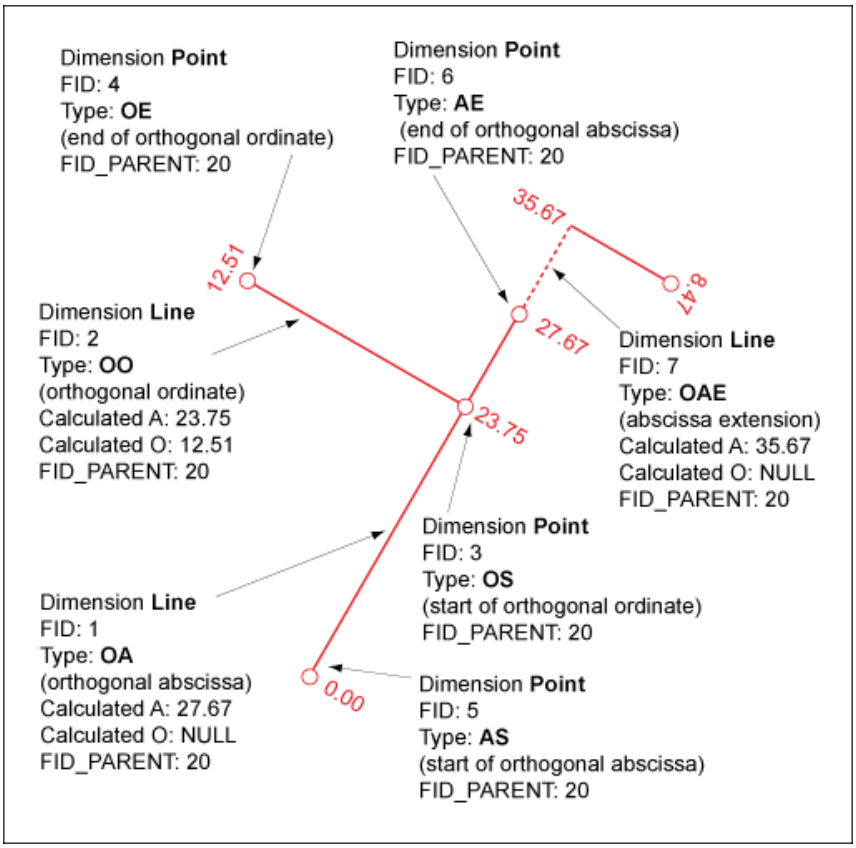
	■ Point of type E: end point of abscissa.
Aligned Text Only	Aligned dimensioning where only the label is, but no abscissa, and no ordinate lines. For example, to display the length of a building edge in the drawing without any lines. Consists of: ■ Line of type AL_: abscissa. Notice the special line type. In the Display Manager, use the line type AL_ to hide the abscissa in the drawing. ■ Point of type S: start point of abscissa. ■ Point of type E: end point of abscissa.

NOTE In the document options you can specify a constant offset value. See COGO and Dimensioning Document Options (page 26).

About Orthogonal Dimensioning

Topobase provides several types of orthogonal dimensioning. Depending on the dimensioning type, the system uses the LINE_TYPE attribute and the POINT_TYPE attribute to distinguish the points and lines. In the display models, you use these attributes to stylize the dimensioning layers.

For example, if the abscissa has been extended, because an orthogonal point is located behind the end point of the abscissa, the line is of type OAE, and you can style the line as a dashed line.



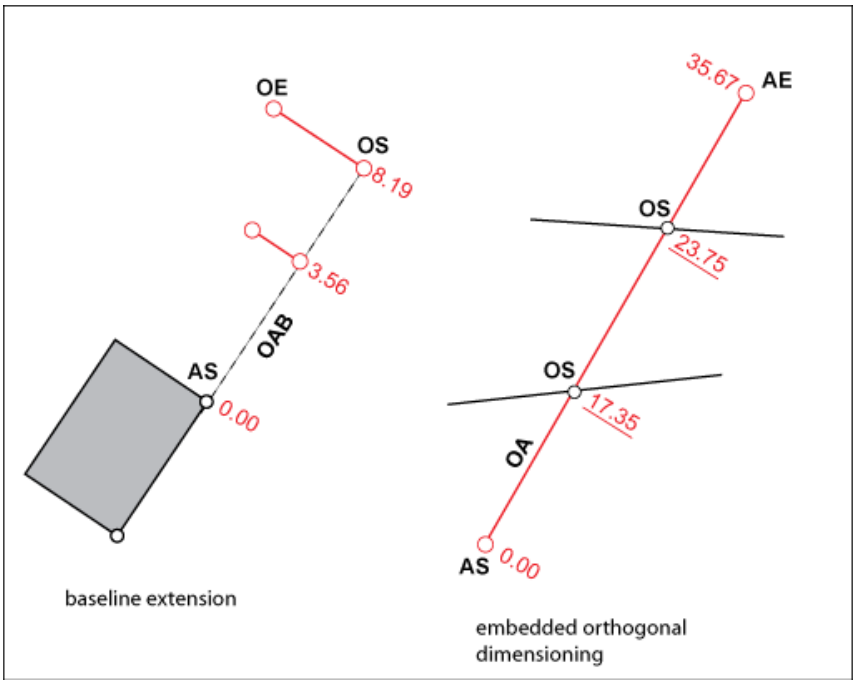
Topobase Data Model: Components of an orthogonal dimension.

An orthogonal dimensioning consists of points and lines, as shown in the following table.

Orthogonal Dimensioning type	Description
OC = Orthogonal Complete	Default orthogonal dimensioning including lines and labels. Consists of: ■ Line of type OA = orthogonal abscissa. ■ Lines of type OAE: Orthogonal Abscissa extension. Only if an ordinate lies behind the abscissa start or end point. ■ Point of type AS = start point of abscissa.

	■ Point of type AE = end point of abscissa. ■ Multiple lines of type OO: ordinates. ■ For each ordinate: point of type OS: start point of ordinate. ■ For each ordinate: point of type OE: end point of ordinate.
OE = Orthogonal Extension Baseline	Orthogonal Dimensioning based on the extension of a baseline. For example, you use the edge of a building as abscissa. Consists of: ■ Line of type OAB = orthogonal abscissa. ■ Lines of type OAE: Orthogonal Abscissa extension. Only if an ordinate lies behind the abscissa start or end point. ■ Point of type AS = start point of abscissa. ■ Point of type AE_ = end point of abscissa. Notice the special line type. In the Display Manager, use the line type AE_ to hide the end point in the drawing. ■ Multiple lines of type OO: ordinates. ■ For each ordinate: point of type OS: start point of ordinate. ■ For each ordinate: point of type OE: end point of ordinate.

Both OC and OE dimensioning support embedded feature dimensioning. An embedded feature is a line that intersects the orthogonal abscissa. The intersection point can be embedded as any other orthogonal point.

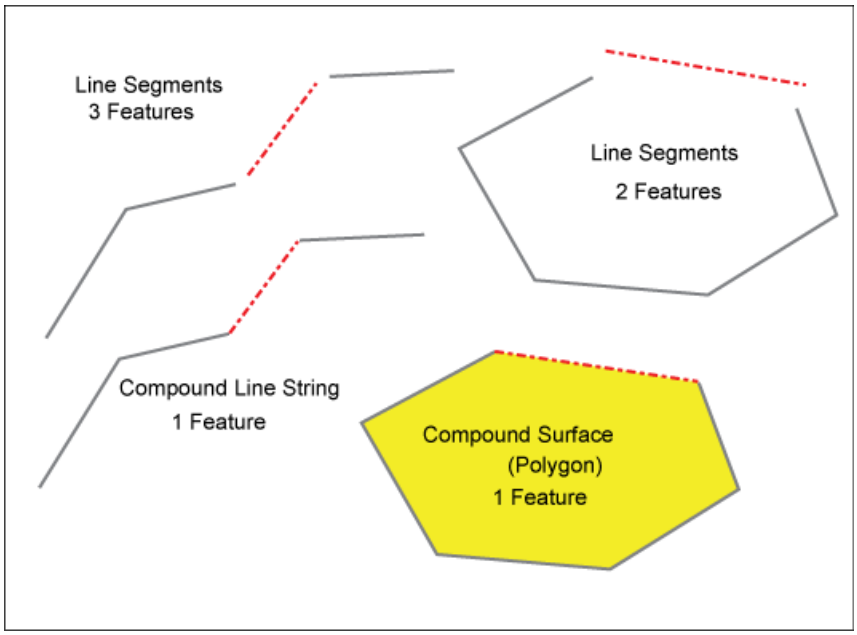


Components of an orthogonal baseline dimension, and an embedded dimension.

Data Model: Compounds

A compound is a feature class type that consists of a logical group of feature classes, which include a parent and a child feature class.

Compounds are used for polygon or line string objects where particular line segments can have different attributes. Therefore a compound represents a group of line segments as one object. For example, several child line segments are combined as a parent polyline or several child line segments are combined as parent polygon.



Topobase Data Model: Compound feature classes

When you digitize features of a compound feature class, you digitize the child line strings, and the parent polygon is generated.

See also:

- Feature Class Type: Compound Polygon (page 175)
- Feature Class Type: Compound Line String (page 176)

Compound Triggers

When a compound child is created, triggers perform the automatic generation of the compound parent features. For this action, triggers do the following:

- Create the parent feature.
- Calculate the polygon area (if closed).
- Calculate the length of the polyline.

Triggers can be enabled or disabled for any reason. You can start compound trigger commands in the data model administrator either from the document (page 521) or from the feature class level.

See also:

- Data Model Administrator: Shortcut Menus (page 152)

.

Compound Commands

You can start the compound commands in the data model administrator either from the document (page 521) or from the feature class level.

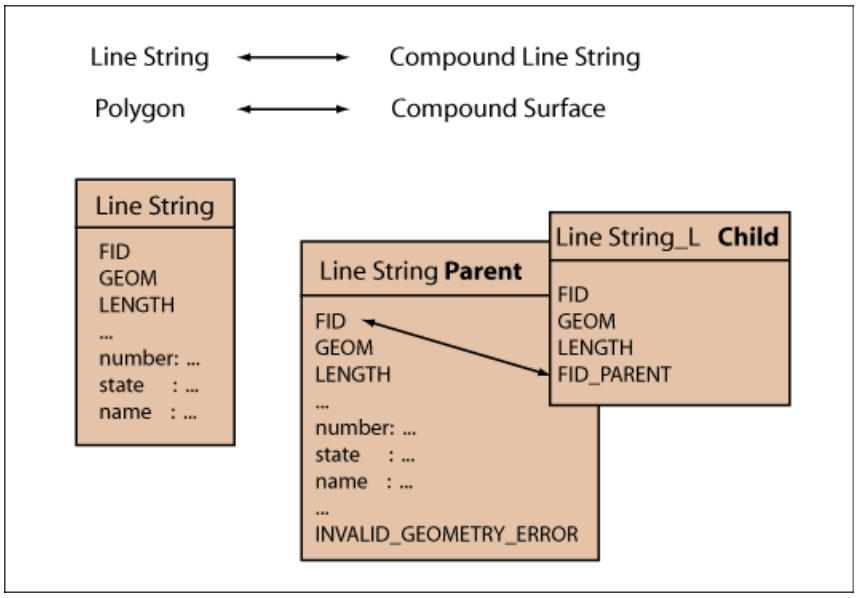
See also:

- Data Model Administrator: Shortcut Menus (page 152)

.

Convert to a Compound

You can convert a line string or polygon feature class into a compound feature class. The existing line string or polygon feature class is the parent feature class and a new child feature class is created. All attributes of the existing parent class are maintained.



Topobase Data Model: Converting a feature class to a Compound.

Eliminate a Compound

You can convert a compound feature class into a line string or polygon feature class. The parent feature class persists, but the child feature class (single segment) and all subordinate feature classes are dropped.

Build Compounds

You can build compounds to combine the data of two related feature classes. This can be useful if you are migrating data from a foreign system to Topobase. Example:

You can build a compound by combining the following:

- Parent feature class with all attribute data, but no geometry.
- Child feature class with a relation to parent feature class, no attribute data, but with geometry, such as line strings.

You use the Build Compound command to save the line string geometry of the related children for each parent and thus build a compound polygon feature class.

Data Model: Topologies

You can create and edit topologies in the Topobase data model administrator. To work on a document's topology (page 525), you must open the corresponding document. In the data model explorer, expand the Topology node and right-click the topology you want to work on.

Shortcut Menu **Topology Item**	**Description**
Initialize Topology	Initializes the topology. See Area Topology Initialization (page 268). See Logical Topology Initialization (page 277). Normally you start the initialization after the creation of the topology, and when a huge number of topology enabled features is stored in the database. Then the initialization creates the corresponding secondary objects, such as areas, and stores the connectivity.
Clear Topology	Clears a topology. You can empty the topology system tables, and then initialize the topology again.
Drop Topology	Drops the topology definition and tables.
Properties	Displays the topology properties.

See also:

- Topology Introduction (page 255)

Data Model: Jobs

Working with Topobase jobs (page 525) requires settings and definitions that must be made with the data model administrator. You can job enable or disable a document. Open the corresponding document, select an item in the data model explorer and right-click to see the job state of each topic or feature class.

NOTE You cannot job enable a single feature class; you must job enable the whole document.

See also:

- Jobs Introduction (page 295)

Data Model: Feature Templates

You use feature templates to efficiently create recurrent arrangements of features. A template comprises geometry, attributes, labels and internal connectivity. Also, you can define templates for different types of equipment that vary in their attribute values (reference records).

You define a template by selecting existing features in a drawing, for example prototype features. After having created the template, you can delete the prototype features. Templates are defined in a local coordinate system. When you instantiate a template in the map, you digitize the origin and specify an orientation.

Optionally a template can be grouped. Then, when you move or rotate a feature of the group, the group is treated as a unit and all other features are moved or rotated the same way.

To work with templates, you must add the Template extension to the document. See also Data Model Administrator: Creating a Database (page 144).

The Template extension adds a topic Template that contains the template system tables TB_FEATURE_GROUP and TB_FEATURE_GROUP_FEATURE.

See also:

- Working with Templates

Data Model: Schematics

Topobase supports multiple locations (representations). You can visualize a feature in different locations, and store the geometries in the database. For generalization, you can digitize features with multiple locations. Also, Topobase maintains consistency, when you update or delete features with multiple locations.

Topobase supports two types of multiple locations.

Schema Plan

A schema plan represents real world features by transforming the original feature geometry to an alternative location, for example by applying a coordinate offset. For example, you create geoschematic or orthogonal schema plans to represent an electric network.

For example, use a geoschematic plan to create a georeferenced cable map for electric utilities that is derived from pathway geometry.

Generalization

Generalization provides a set of generalization feature classes that can be used to digitize secondary features at one or more alternative location. For example, you displace features to create an alternative representation for an overview map with a scale bigger than 1:10'000.

To create a representation template, start the Topobase data model administrator. Right-click the Representations node and click a command.

Shortcut Menu **Representations node**	**Description**
Create Schematic	Creates a schema plan template. See also Creating a Schema Plan Template (page 218).
Create Generalization	Creates generalization feature classes. See also Creating a Generalization (page 223).

To view available representations, click the Representations node. Each representation type is indicated by a special icon.

- —Geoschematic diagram
- —Orthogonal diagram
- —Generalization

Getting Schematics Started

Use Topobase Administrator to

- Create schematics feature classes.
- Create schema plan templates. See Creating a Schema Plan Template (page 218).
- Create generalization feature classes.

Use Topobase Client to

- Select the schema plan template.
- Select the features and build a schema plan, that means to create the schematic features.
- Digitize generalization features, using the generalization feature classes.

Building a schema plan means deriving new geometries from existing geometries based on their existing logical connectivity, and based on the defined rules.

Creating a Schema Plan Template

The Create Schematics command creates the schema plan template that comprises the schematic feature classes, and rules how to derive the schematic features from the real world features.

To create a schema plan template

1. Start Topobase data model administrator.
2. Expand the document node.
3. Right-click the Representations node, and click Create Schematics.
4. Specify the properties as shown in the following table.
5. Click Create.

Create Schematics	Description

General area	Specifies general settings.
Schema Plan Name	Specifies the name. This name will be displayed in Topobase Client, when you select the representation.
Schema Plan Type	Specifies the type of schema plan. Select either ■ Orthogonal ■ Geoschematic The type of schema plan specifies the rules that apply when you create the schematic features.
Topology	Selects the topology to be associated to the schema. When you create the schematic features, the topology is used to determine the connectivity of the real world features. The list displays all logical topologies of the document.
Schema Point Feature Class	Specifies a point feature class to store the schematic point features. Either select an existing point feature class from the list, or click Create New to create a new feature class.
Schema Line Feature Class	Specifies a line string feature class to store the schematic line features. Either select an existing line string feature class from the list, or click Create New to create a new feature class.
Settings area	Specifies the rules to create the schematic features. Click Details to display a description of the settings. Some settings only apply to particular types of schema plans.
Drawing Priority	Specifies the priority of the direction in which the schematic features are drawn. See Schema Plan Settings (page 220). ■ 0 = Horizontal. ■ 1 = Vertical. Applies to Orthogonal schema plans.

Multiline Offset	Applies to orthogonal and geoschematic schema plans in the Electric CE module. See Multiline Offset and Distribution (Electric CE) (page 221).
Multiline Distribution	Applies to orthogonal and geoschematic schema plans in the Electric CE module. See Multiline Offset and Distribution (Electric CE) (page 221).

Schematic features can be point or line features. They are stored in two feature classes.

- Point location feature class—Point feature class.
- Line location feature class—Line string feature class.

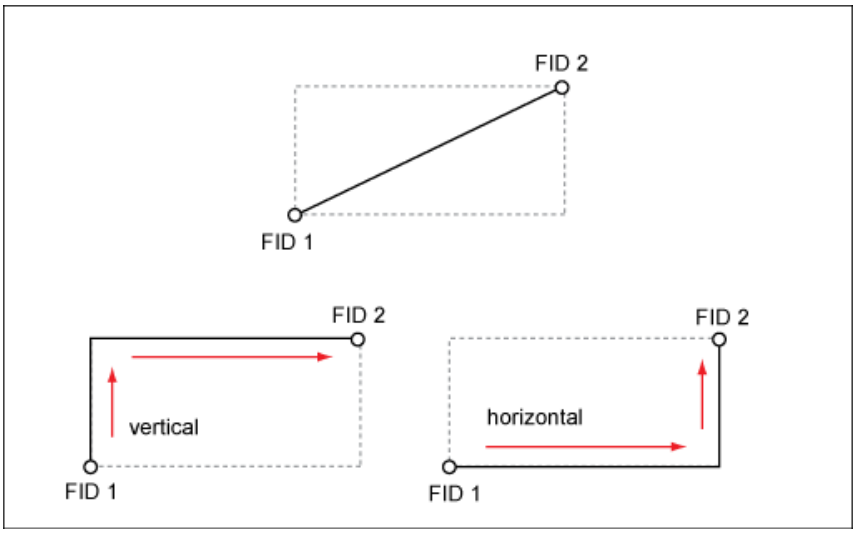
When you create the schema plan template, a new item is added under the Representation node. Click the schema plan to view the properties in the right pane.

Schema Plan Settings

Depending on the schema plan type, and depending on the module, in the Create Schematic dialog box, you specify additional settings. The Settings area only displays the settings that are valid for the selected schema plan type.

Drawing Priority

For orthogonal schema plans, the drawing priority specifies the priority of the direction in which the schematic features are drawn. Can be either 0 = horizontal or 1 = vertical.

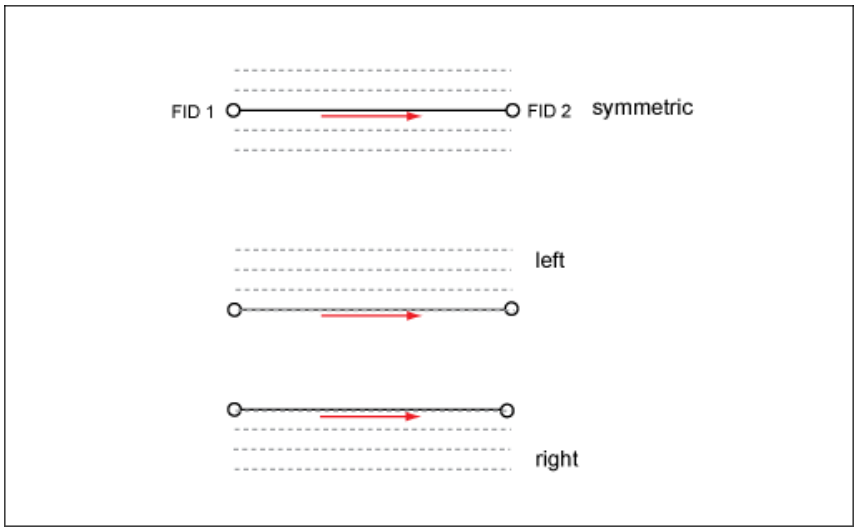


The direction of the lines between two points is derived from the FID. For example, a line will be created from FID 1 to FID 2, either starting vertically, or horizontally.

Multiline Offset and Distribution (Electric CE)

In the Topobase Electric CE data model, you can specify the multiline offset and distribution, both for orthogonal and geoschematic plans.

- **Multiline Offset**— Specifies the offset distance between overlapping multiline conductor features and the parent single line feature. The offset line is drawn in parallel to the original feature. The offset is the distance between the parallel features. Unit is the document length unit.
- **Multiline Distribution**—Specifies the distribution of multiline conductor features around the parent line feature. Can be either
  - 1 = left
  - 0 = symmetric
  - 2 = right



The left or right distribution direction is defined by the FID.

Editing a Schema Plan Template

In the data model explorer, use the shortcut menus to edit a schema plan template.

To edit a schema plan template

1. In the data model explorer, expand the Representation node, and select the schema plan.
2. Right-click, and click one of the following commands.

Schema plan shortcut menu	Description
Edit Schema	Opens the Create Schema dialog box. You can modify the name and the settings. You cannot modify the feature classes and the topology.
Empty Schema	Deletes all associated schematic features, but preserves the schema plan template.

Drop Schema	Deletes the schema plan template. Optionally you can also delete all associated schematic features, and the schematic feature classes.

Creating a Generalization

The Create Generalization command creates generalization feature classes for different types of geometry.

To create a generalization

1. Start Topobase data model administrator.
2. Expand the document node.
3. Right-click the Representations node, and click Create Generalization.
4. Specify the properties as shown in the following table.
5. Click Create.

Create Generalization	Description
Generalization Name	Specifies the name. This name will be displayed in Topobase Client on the Home tab, in the Data Source panel, when you select the representation.
Generalization Point Feature Class	Specifies a point feature class to store the generalization features. Either select an existing feature class from the list, or click Create New to create a new feature class.
Generalization Line Feature Class	Specifies a line string feature class to store the generalization features. Either select an existing feature class from the list, or click Create New to create a new feature class.
Generalization Polygon Feature Class	Specifies a polygon feature class to store the generalization features. Either select an existing feature class from the list, or click Create New to create a new feature class.

Generalization Collection Feature Class	Specifies a collection feature class to store the generalization features. Either select an existing feature class from the list, or click Create New to create a new feature class. Use a collection feature class to store features of any geometry type.

Editing a Generalization

In the data model explorer, use the shortcut menus to edit a generalization.

To edit a generalization

1. In the data model explorer, expand the Representation node, and select the generalization.
2. Right-click, and click one of the following commands.

Generalization shortcut menu	Description
Edit Generalization	Opens the Create Generalization dialog box. You can modify the name.
Empty Generalization	Deletes all generalization features, but preserves the generalization feature classes.
Drop Generalization	Deletes the generalization, including the generalization features. Optionally you can also delete all generalization feature classes.

Data Model: Intersections

Use the Topobase intersection routines to calculate intersections between two feature classes. Intersections can be defined in the Topobase data model administrator and added to the document explorer (page 521) in the Topobase task pane.

Examples: An intersection of the parcel feature class with the land use feature class determines the polygon sections of different land use for each parcel.

An intersection of the parcel feature class with the control point feature class determines all parcels where a control point is located.

NOTE You can execute intersections between 3D feature classes, or between a 2D and a 3D feature class. See also Intersections: Setting up 2D / 3D Intersections (page 232).

The following table shows the different types of intersections:

Intersection Type	Description
OO	Polygon with polygon Result: polygon feature class
OL	Polygon with line Result: line string feature class
LL	Line with line Result: point feature class
OP	Polygon with point Result: point feature class
LP	Line with point Result: point feature class

An intersection requires a primary and a secondary feature class. These feature classes can be identical. Therefore, for example, you can compare parcel areas before and after a modification.

Intersection feature class—The results of intersections are stored in the database in special target feature classes. Each intersection type has its own target feature class with certain default attributes. The intersection feature class is either created when the intersection is defined, or it is created explicitly by the user.

Intersections can be performed with a job, or without a job:

Intersection within a job—If the intersection is part of a job, the performance is dependent on the job change state of the feature class.

Intersection without jobs—In this case, you can start the process manually from the intersection shortcut menu.

NOTE The intersection feature class is not job-enabled due to dependencies between topics, if the primary and secondary feature classes are not in the same topic.

Creating an Intersection

To create an intersection

1. Start the data model administrator.
2. Right-click the Intersection root item and click Create Intersection.
3. In the Create Intersection dialog box, specify the properties, as shown in the following table:

Create Intersection Dialog Box Properties	Description
Definition	
Intersection Name	Specifies the intersection name. The definition of the intersections are saved in the database and can be identified by this name.
Feature Class Name	Specifies the name of the intersection feature class that stores the intersection results. For each intersection, the system creates a feature class corresponding to the intersection type.
Execute The Intersection When New Job State Is	Specifies the job state when the intersection is executed, such as Pending, or Open.
Store Geometry	Select this option to store the intersection geometry.By default, the resulting geometry is not saved.
Primary Feature Class area	
Primary Feature Class	Specifies the name of feature class. Select an available valid feature class in the list box. At the moment valid feature classes are line string and polygon feature classes.

Where Clause Of Primary Feature Class	
Assistant button	Opens the SQL Assistant. Optionally, you use the SQL Assistant to create the WHERE Clause. See Creating SQL Expressions (page 103).
Update The Values Of Primary Feature Class	Specifies the values from other tables to be stored in the intersection feature class. These values can be used for reports, which is much faster than making a view and joining the information from other tables and the intersection result in the view.
Use This Job	Job used to get data from DB. If "JOBID" is "0", the currently selected job is used.
Secondary Feature Class area	
Secondary Feature Class	Specifies the name of the feature class. Select an available valid feature class in the list box.
Where Clause For Secondary Feature Class	
Assistant button	Opens the SQL Assistant. Optionally, you use the SQL Assistant to create the WHERE Clause. See Creating SQL Expressions (page 103).
Update the Values Of Secondary Feature Class	Specifies the values from other tables to be stored in the intersection feature class. These values can be used for reports, which is much faster than making a view and joining the information from other tables and the intersection result in the view.
Use This Job	Specifies the job used to get the data from DB. If "JOBID" is "0", the currently selected job is used
Adjustment area	**NOTE** These settings are for intersections of two polygon feature classes only. See Intersections: Adjustment (page 228).

Adjust Area_Nominal	Performs an adjustment. Edit the following settings:
Use Only Specified Features For Adjustment (Where Clause)	Defines a WHERE clause that filters the features to be adjusted. Example: `s.area>100`. s is the alias for the secondary feature class.
Tolerance Area_Nominal	See Intersections: Adjustment (page 228).
Tolerance [m] For Intersection	If the result is a polygon, defines a tolerance to clean up the resulting polygon. If two neighboring points lie within this tolerance, one point will be removed. If the pitch of an arc is less than this tolerance, the arc will be linearized.
Buffer	Specifies a buffer for the inside selection of the primary feature class. To find the primary features, all features INSIDE the perimeter are returned.

The values are stored in the system table TB_INTERSECTION.

See also:

- System Table TB_INTERSECTION (page 79)

Intersections: Adjustment

When you intersect two polygon feature classes, the sum of all resulting intersection areas (AREA_NOMINAL) should be equal to the area of the primary feature class. To achieve this, you can perform an adjustment of areas.

The adjustment function calculates the difference between one area of the primary feature class and the related sum of areas of the intersection feature class. Then, the function distributes the difference to the largest resulting area of the intersection feature class.

Example—The difference between the sum of the intersection area_nominal (234) and the area_nominal of the primary feature (235) is greater than the Tolerance Area Nominal. Then, the difference of 1 must be distributed to the intersection area_nominal.

The distribution is executed twice. The first distribution is executed only for the features that have not been excluded using the WHERE clause. Then, if there is still a difference, all features are used for the second distribution.

Example—Intersection between the primary feature class Parcel and the secondary feature class Landuse. Intersection properties and result areas are the following:

- 235 square meter of parcel area
- 150 square meter of street landuse section
- 50 square meter forest land use section
- 34 square meter grassland.
- Sum of areas is 234, which is < parcel area.
- Difference to be distributed = 1
- Tolerance Area Nominal = 0.5
- Use Only Specified Features For Adjustment (Where Clause) = s.type != 1(streets)

In the example the difference must be distributed. Without the WHERE clause it would be added to the largest area of intersection, which is street. With the WHERE clause you exclude the streets from the adjustment. Then, the difference is added to the area of the forest section, because this is the second largest area. If the parcel has only streets sections, the first distribution can not be applied. In the second distribution all features will be considered, and the difference is added to the intersection area_nominal of the largest street section on the parcel.

See also:

- System Table TB_INTERSECTION (page 79)

Intersections: Getting Started

To work with intersections, you must have the following:

- Intersection definition (in the data model administrator).
- Intersection feature classes and intersections to be shown in the document explorer in the Topobase task pane

You can start a predefined intersection manually. Use the predefined intersections for frequent intersections. To start an intersection in the Topobase data model administrator, right-click the intersection item and click Intersect.

Alternatively, in the Topobase Client or Web task pane, in the document explorer, right-click the Intersection node and click Intersect. You can either intersect the whole area or select the areas of the primary feature class.

To start an intersection from the document explorer

1. In the document explorer, expand the Intersection node and select the intersection.
2. Right-click and click Intersect.
3. In the Select The Perimeter For The Intersection dialog box, select one of the following:

   Intersect Over Area (select features) and click Select.

   Intersect Over All (all features) and click OK.

NOTE When selecting the perimeter in the drawing, we recommend to use an appropriate Object Snap. For example, to select line features, use OSNAP = NEAREST.

Also, you can start an intersection and perform temporary and occasional intersections, by entering the parameters in dialog mode.

To start a temporary intersection

1. Start the data model administrator.
2. In the data model explorer, select the primary feature class. Right-click and click Intersect With.
3. Select the secondary feature class and double-click.

Under the topic Intersection, the target feature class is created with a name containing the names of the feature classes.

NOTE If you want to repeat a temporary intersection with the same feature classes, you must drop the topic of the target feature classes beforehand.

Intersections: Results

The following table describes the target feature class of a polygon –polygon intersection (OO). The name of the target feature class is specified in the Create Intersection dialog box. The system creates a topic Intersection with sub topics <intersection name> for each definition.

Polygon—Polygon Intersection Target Feature Class Column	Description
AREA	Only polygon feature class
AREA_NOMINAL	
ATTRIBUTE_*	1 - 4
FID	Primary key
FID_PRIM	Foreign key of primary feature class
FID_SEC	Foreign key of secondary feature class
GEOM	For more information, see *Oracle Spatial Users Guide and Reference*.
JOB_ID	
LENGTH	Only line string feature class
ORIENTATION QUALITY Z	Only point feature class
PRIM_*	1 - 3
SEC_*	1 - 3

*_TSER feature class: Each target feature class has a related collection feature class *_TSER. A trigger (feature rule) on the intersection feature class will validate the new features and copy invalid features to _TSER table.

The perimeter is defined by the selected job or by selection of the user.

Before intersection, all old and invalid features inside the perimeter must be deleted.

Intersections: Setting up 2D / 3D Intersections

You can execute mixed intersections between a 2D and a 3D feature class. For example, you intersect a 2D feature class with a view on a 3D feature class. The intersection is calculated using the 2D coordinates where the Z value is ignored, and results are stored as 2D geometry, or 3D geometry with Z=0. An exception is n intersection in a 3D document, if the secondary feature class is a point. In that case, the result is the original point with X,X, and Z coordinates.

To execute mixed intersections, you must set up the intersection feature classes manually. As a Topobase document can either be 2D or 3D, the 2D and 3D feature classes are stored in different documents. In the document where you want to create the intersection (target document), you must create a view to access the feature class of the source document.

To set up 2D/3D intersections

1. In the target document, create the view.
   - Create a synonym to the feature class in the source document.
   - Grant SELECT on the synonym.
   - Create a view on the synonym.
   - In Topobase Administrator, create a feature class from the view. See Creating a Feature Class From a View (page 164).
2. Create the intersection. See Creating an Intersection (page 226).
3. If your target document is 2D, set the dimension of the feature class (view) back to its original value TB_DICTIONARY.DIMENSION = 3. This is required, because you can only create the intersection if the two feature classes have the same dimension.
4. Re-open the workspace.
5. Execute the intersection.

NOTE To improve performance in a 3D document, make sure that your primary feature class is 2D, and your secondary feature class is 3D.

Data Model: Spatial Relationship Settings

Autodesk Topobase applications can analyze spatial relationships. With the data model administrator, while creating feature classes, you can set spatial properties to control the spatial behavior during spatial selections.

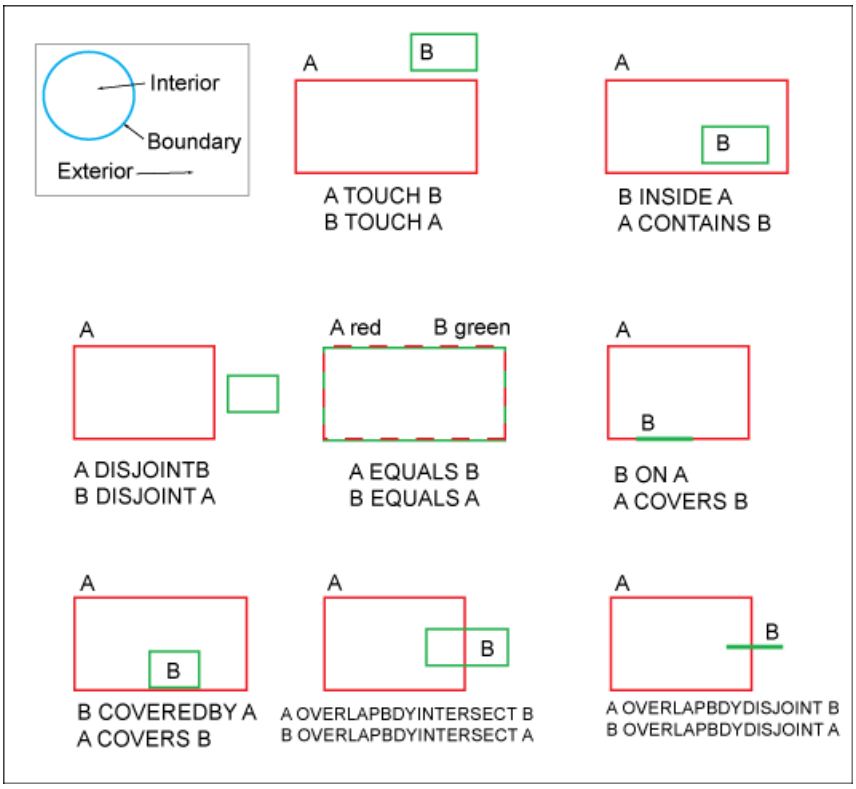
You can define geographic selections with topology queries or spatial operators. For example, in Topobase Client in the Advanced Generate Graphic dialog box, use the Advanced Settings button.

The SDO_RELATE operator performs both primary and secondary filter operations, where the mask parameter specifies the topological relation of interest.

Examples: Spatial Operator SDO_RELATE

Topological relations:

- SDO_RELATE: spatial operator for evaluation of topological relations.
- Each spatial object has an interior, boundary, and exterior.
- The boundary separates interior and exterior.



Oracle Spatial SDO_RELATE operator: Intersection patterns

Names of topological relationships:

These are the following spatial intersection patterns:

Name	Topological relationship
DISJOINT	The boundaries and interiors do not intersect.
TOUCH	The boundaries intersect, but the interiors do not intersect.
OVERLAPBDYDISJOINT	The interior of one object intersects the boundary and interior of the other object, but the two boundaries do not intersect. Example: When a

	line originates outside a polygon and ends inside that polygon.
OVERLAPBDYINTERSECT	The boundaries and interiors of the two objects intersect.
EQUAL	The two objects have the same boundary and interior.
CONTAINS	The interior and boundary of one object is completely contained in the interior of the other object.
COVERS	The interior of one object is completely contained in the interior of the other object, and their boundaries intersect.
INSIDE	The opposite of CONTAINS. A INSIDE B implies that B CONTAINS A (Boundary and interior of one object are completely inside the other object).
COVEREDBY	The opposite of COVERS. A COVEREDBY B implies that B COVERS A (Demarcations of two objects which partially cover themselves, and the interior of the other object is completely contained within the object).
ON	The interior and boundary of one object is on the boundary of the other object (and the second object covers the first object). Example: When a line is on the boundary of a polygon.
ANYINTERACT	The objects are non-disjoint, which means all.

For more information, see *Oracle Spatial Users Guide and Reference*.

Data Model: Oracle Schema Converter

Autodesk Topobase provides a schema converter that converts any Oracle structure into a Topobase database schema. For example, use the schema converter to convert an Oracle table in a Topobase feature class. In the first

step, the schema converter analyzes all non Topobase tables, and lists any conflicts. After resolving the conflicts, the tables are ready to be converted. The conversion adds Topobase specific columns, renames the geometry column to GEOM, creates and stores the Topobase metadata.

To analyze and convert an Oracle schema

1. Start Topobase Administrator.
2. Create a workspace.
3. Under Documents, click Import.
4. Do one of the following.
   - Select Import From Existing Oracle Schema. In the Add Existing Schema dialog box, clear Show Only Topobase Schemas. Select the schema. Enter the password.
   - Select Import From Oracle Dump File. Select the dump file. Specify the settings. Enter the password.
5. If you select a non-Topobase schema, or a non Topobase dump file, you are prompted whether to create a Topobase structure, and to open the Topobase Converter now. Click Yes.
6. In the Create New Document dialog box, under Extensions, check the Extensions, and under Spatial, check the SRID. Click OK.

NOTE Topobase supports only one SRID per document. When you create a new document to import a non Topobase schema, make sure that you specify the SRID in the Document Settings - Spatial. Then, the schema converter will convert all tables that match the document's SRID.

The non-Topobase schema is analyzed, and the results are displayed in the Convert To Topobase Schema dialog box. Use the settings as shown in the following table.

Convert To Topobase Schema	Description
Tables	Displays the analysis results. Use the data grid to filter for tables where a conflict has been detected.
Analyze And Convert	Analyses and Converts the Oracle schema.

Analyze	Analyzes the data structure. If you have resolved any conflict, click Analyze to restart the analysis. You can start the analysis as often as needed.
Convert	Starts the conversion. See also Converting Oracle Tables (page 238).
Display	Selects whether to show all tables in the data grid, or tables with conflicts only.
data grid	Displays the results of the analysis. Click a row, to display more details in the right pane.
Table Name	Displays the table name.
State	Displays the result. ■ Ready = Table can be converted. ■ Resolve Conflicts = Table cannot be converted. Conflicts must be resolved.
State Description	Displays the analysis details of the selected table. Select a row, and use the options in the right pane to resolve simple conflicts, or use any Oracle tool to resolve other conflicts. If appropriate, select multiple rows to resolve the conflicts.
Table Name	Displays the table name. If the table name has more than 25 characters, enter a shorter table name.
Table Type	Specifies the table type to resolve the following simple conflicts. **Table has no geometry column** ■ Do not Convert. ■ Convert To Attribute Feature Class. ■ Convert to Domain Table. **Table has a geometry column, but the geometry type cannot be determined** ■ Attribute

- Point
- Line
- Polygon
- Collection

Primary Geometry Column	For feature classes that have multiple GEOM columns, selects the primary geometry column.

When you have resolved the conflicts, you start the conversion of the tables that have the state Ready.

To convert the schema

1. In the Convert To Topobase Schema dialog box, click Convert.
2. The data grid displays the converted tables, and the State. The State can either be Conversion Succeeded, Resolve Conflicts, or Conversion Errors.

For a Conversion Error, the right pane shows more details. Use any Oracle tool to resolve the errors, and start the analysis and the conversion again.

NOTE To analyze and convert single Oracle tables that are part of a Topobase document, start the schema analysis in the document. In the data model explorer, right-click the document node and click Convert Tables To Feature Classes.

For more information, see *Oracle Spatial Users Guide and Reference*.

See also:

- Data Model: Feature Class Types (page 172)
- Data Model: Domain Tables (page 183)

Converting Oracle Tables

The Topobase schema converter converts tables that store geometry according to the GTYPE of the geometry data, as shown in the following table.

Geometry type GTYPE	Feature Class Type
Point	Point feature class.

Multipoint (2D)	Collection feature class.
Line Multiline	Line feature class.
Polygon Multipolygon	Polygon feature class.
Unsupported geometry types, such as LRS, 3D Point Cloud, Composite Surface, Solid.	Attribute feature class.

After the Conversion

Notice, that the conversion of an Oracle schema to a Topobase document does not perform the following.

- Create new spatial indices. The conversion re-creates the existing spatial indexes.
- Validate geometry.
- View conversion: Views pointing to renamed geometry columns can become invalid.
- Handle job functionality.
- Distinguish between label feature class, centroid feature class, and point feature class. These feature classes have point geometry, and will be converted into point feature classes.
- Set feature rules, other than the default feature rules, such as to calculate the length, and the area.
- Set different topics. All converted feature classes are assigned to one topic.

Topobase Data Transfer (Oracle Data)

The Topobase Oracle Data Import tool transfers Oracle data from any Oracle user to a Topobase document. Oracle Data Import requires an Oracle user (Intermediate document) that stores the data to be transferred. However, it is not required that the data structure of the intermediate document matches the final data structure. For example, the intermediate document can contain

data that has been imported from some external sources, without changing its original data structure (flat port).

For the data transfer into your final document, you use the Oracle Data import. You specify the mapping definition to distribute the data from the intermediate document to the final document. The mapping definition for the Oracle Data import is stored in the final document.

Oracle Data Import—Migration tool to transfer Oracle data to Topobase documents.

Source document—Topobase document from which the data is transferred. For example an intermediate Interlis document, to which the Interlis data has been imported 1:1.

Final document—Topobase document to which the data is transferred.

NOTE Oracle Data import is an integrated part of the Topobase Interlis Import. See also Mapping Definition for Interlis Data Import (page 252).

IMPORTANT The final document must contain the Oracle Data Import Extension. See also Document Settings (page 7).

To transfer data between two documents

1. Start Topobase Administrator, and open the workspace that contains the final document.
2. In the Administrator explorer, select the final document.
3. Click Document menu ➤ Import ➤ Oracle Data.
4. In the Source Document dialog box, select the source document, and enter its credentials. Click OK.
5. Specify the mapping definition. See Creating a Mapping Definition (page 240).
6. Test the mapping definition.
7. Import the data.

Creating a Mapping Definition

The Oracle Data import configuration consist of a mapping definition that describes how each feature of the source document is transferred to the final

document. The mapping definition is stored in the final document (TB_MIG* tables), and can be exported to an (*.*XML*) file for reuse in multiple documents.

To create a mapping definition

1. Start Topobase Administrator, and open the workspace that contains the final document.
2. In the Administrator explorer, select the final document.
3. Click Document menu ➤ Import ➤ Oracle Data.
4. In the Source Document dialog box, select the source document, and enter its credentials. Click OK.
5. In the Import Oracle Data dialog box, click New.
6. In the Create New Mapping Definition dialog box, configure the mapping file. See Configuring Oracle Data Import (page 243).

Editing a Mapping Definition

A mapping definition is the Oracle Data import configuration that describes how each feature of the source document is transferred to the final document. The mapping definition is stored in the final document, and can be exported to a (*.*XML*) file for reuse in multiple documents.

To edit a mapping definition

1. Start Topobase Administrator, and open the workspace that contains the final document.
2. In the Administrator explorer, select the final document.
3. Click Document menu ➤ Import ➤ Oracle Data.
4. In the Source Document dialog box, select the source document, and enter its credentials. Click OK.
5. In the Import Oracle Data dialog box, click Edit.
6. In the Edit Mapping Definition dialog box, configure the mapping file. See Configuring Oracle Data Import (page 243).

Using Oracle Data Import

You use the Import Oracle Data dialog box to manage the data transfer.

Import Oracle Data dialog box	**Description**
Select Mapping Definition	Selects the Oracle Data import configuration that defines the data transfer.
New	Creates a new mapping definition. Opens the Create New Mapping Definition dialog box. You either start from scratch, or you import and amend an existing mapping definition (**.XML*) file. See also Importing an Oracle Data Import Mapping Definition (page 253).
Edit	Opens the Edit Mapping Definition dialog box.
Delete	Deletes the selected mapping definition.
Temporary Directory	Specifies a temporary directory to save files, such as Log files, Error files, or SQL Loader files.
Status	Displays status messages.
Test Mapping Definition	Validates the mapping definition without migrating any data. The test ■ Creates access permissions (Oracle grants). ■ Runs the SQL select statements. ■ Revokes the permissions. ■ Writes the results into the log files *OracleDataImport.log*, and *OracleDataImport.err*. The *OracleDataImport.err* file stores the SQL select statements that need to be corrected.
Import	Starts the Oracle Data Import, and transfers the data to the final document.

Configuring Oracle Data Import

Use Oracle Data import to configure the data transfer, and to create or edit the mapping definition. The Create New Mapping Definition dialog box, and the Edit Mapping dialog box consist of several tabs: Overview; Feature Classes; Domains; Feature Attributes; Domain Attributes.

Create New Mapping Definition dialog box	**Description**
Overview tab	Displays a high level overview of the mapping definition.
Mapping Definition Name	Specifies a name. For example, the name is displayed in the Import Oracle Data dialog box, or when you select the mapping definition for the Oracle Data import in the Interlis Data Import dialog box.
Created From	If the mapping definition has been imported from a file, displays the (**.XML*) file name. See also Importing an Oracle Data Import Mapping Definition (page 253).
Mapping Definition Status	Compares the selected configuration to the data structure of the current document. Indicates whether the configuration of each element is complete. Click View to display the detail tabs. **Linked**—Indicates completed mapping definitions. **Not Linked**—Indicates missing mapping definitions. For example, if you have not added attributes to your feature classes.
Advanced Settings Status	Indicates whether SQL select statements that are executed before and after the Oracle Data import, or placeholders have been defined. Click View to open the dialog boxes to edit the settings. See Oracle Data Import: SQL Statements and Placeholder (page 252).

Import	Imports a mapping definition from an (*.XML) file.
Export	Exports the mapping definition to an (*.XML) file. For example to backup your configuration.
Close	Quits the dialog box. The mapping definition is automatically saved.
Advanced Settings	Opens dialog boxes to create or edit SQL select statements, and placeholders. See Oracle Data Import: SQL Statements and Placeholder (page 252).
Feature Classes tab	Links the feature classes of the source document with the feature classes of the final document. See also Oracle Data Import: Mapping Feature Classes and Domains (page 246).
Link Feature Classes area	Displays the feature classes of the source, and of the final document. To link two feature classes, select the source feature class, then select the final feature class, and click the Link Feature Classes icon.
	Links the selected feature classes.
	Unlinks the selected feature classes.
	Automatically links all feature classes with the same name.
	Creates a copy of the selected feature class in the final document.

Linked Feature Classes (Shown In The Order They Will Be Mapped) area	Displays the feature classes that have been linked in the order they are mapped in the mapping definition. Use the icons on the right to modify the order. Properties: Displays the migration method, such as SQL Loader, or API.
	Unlinks the selected feature classes.
	Opens the Edit Properties dialog box. See Oracle Data Import: Edit Feature Class Properties (page 248).
Domains tab	Links the domains of the source document with the domains of the final document.
Link Domains area	Displays the domains of the source, and of the final document. To link two domains, select the source domain, then select the final domain, and click the Link icon.
Linked Domains area	Displays the domains that have been linked. Use the icon to unlink the domains.
Feature Attributes tab	Links the feature attributes of the source document with the feature attributes of the final document.
Prioritized List Of Linked Feature Classes area	Displays the feature classes that have been linked in the Feature Classes tab. For each feature class, an icon indicates whether all attributes have been linked.
Link Feature Attributes	Displays the attributes of the selected feature classes.
Linked Feature Attributes (Shown In The Order They Will Be Mapped) area	Displays the feature attributes that have been linked. Displays properties that have been set. See also Oracle Data Import: Edit Feature Attribute Properties (page 250).

	Opens the Edit Properties dialog box. See also Oracle Data Import: Edit Feature Attribute Properties (page 250).
	Automatically links all attributes as specified in the option. ■ Link all attributes with the same name. ■ Link all attributes that are not linked, to a NULL attribute, so that all attributes can be considered as Linked. ■ Link all attributes that contain no data to a NULL attribute.
	Switches to the Domain Values tab. If the item is related to a domain table, activates the corresponding domain.
Domain Values tab	Links the domain values of the source document with the domain values of the final document.
Linked Domains	Displays the domains that have been linked in the Domains tab. For each domain, an icon indicates whether all values have been linked.
Link Domain Values	Displays the values of the selected domains.
Linked Domain Values	Displays the domains that have been linked. Use the arrow buttons to modify the order of the domain values.
Domain Settings	Opens the Domain Settings dialog box. See Oracle Data Import: Domain Settings (page 251).

Oracle Data Import: Mapping Feature Classes and Domains

The following sections explain details about the settings that you specify in the Create New Mapping Definition dialog box.

To link feature classes

1. Click the Feature Classes tab.

   Feature classes that have not yet been linked, are marked with a special icon .

2. Select the source feature class, and the final feature class.

3. Click Link Feature Classes.

   The linked feature classes are added to the Linked Feature Classes list. Icons indicate that some attributes need to be mapped.

4. Click the Feature Attributes tab.

5. Select the linked feature classes row, and under Link Feature Attributes, link the attributes.

When you specify a mapping, you can use the table name / attribute name NULL to explicitly not migrate the item. Then the system does not mark the item as Not Linked. For example, when you share your configuration with another user, the NULL items would not be marked as Not Linked. However, any additional feature classes and attributes of the other document are marked as Not Linked, and missing mapping definitions can be added.

Feature Attributes

When you link feature attributes, the system checks the attribute type. For example, you cannot link an attribute of type Number(10) to an attribute of type Number(1), or you cannot link an attribute of type SDO_GEOMETRY to an attribute of type Varchar2.

If the final attribute is of type NOT NULL, a DECODE command inserts a default value.

If both the source attribute name, and the final attribute name is ORIENTATION, and if the source attribute is of type Number(20,8), and the final attribute is of different type, the system changes the final attribute type to Number(20,8).

FID attribute

When you link the source attribute FID to the final attribute FID, the application creates a system attribute T2T_FID. In the list, T2T_FID is displayed in parentheses, next to the FID attribute. T2T_FID stores the source FID. The final FID will be newly generated during the data transfer, in order to generate

consistent FIDs. T2T_FID (the source FID) is used to restore any relations between the features.

See Oracle Data Import: Edit Feature Attribute Properties (page 250)

Feature relations

Optionally, relations to other features are restored. For this purpose, the source FID is stored in system attribute T2T_<attribute name>, and an update SQL script (Post SQL) is created. The Post SQL will be run after the migration. In the Linked Feature Attributes list, under Properties, the entry PostScript indicates that a Post SQL has been created.

To edit the Post SQL, click the Edit icon. See also Oracle Data Import: SQL Statements and Placeholder (page 252).

Domain table relations

If both the source attribute, and the final attribute has a relation to a domain table, you have to map the domain tables. That means, you assign the values of the source domain table to the values of the final domain table.

Example: A source domain table specifies colors, such as 1=Red, 2=Blue, 3 = Black. You map the source domain table to a final domain table, where the following values are stored: 53=Red, 54=blue, 55=Black. Then you map the domain values, so the migration changes the values: 1 to 53, 2 to 54, 3 to 55.

To copy a value of the source domain table to the final domain table, select the value, and click the Copy icon. For example, to add missing values. This creates a KEY value, starting at 10001 for numeric keys, and copies the SHORT_VALUE.

Oracle Data Import: Edit Feature Class Properties

In the mapping definition for a feature class, you specify properties as shown in the following table.

Property	Description
Include In Mapping Definition	Includes the settings in the mapping definition.
Migration Model	Specifies the migration method. ■ Add As A New Feature Using Topobase API: Inserts each feature separately. This method guarantees data consistency. No post pro-

	cessing is required. Use this method to migrate a small amount of data. ■ Add A New Feature Using SQL Loader: This method inserts the data from SQL Loader files. For each feature class, Insert statements are created, so that client-side feature rules would not be applied. For example, use this method to find a single record that stops the data transfer. ■ Add As A New Feature Using Insert Statement: This method creates an insert statement per feature class, so that client-side feature rules are not applied. This is the fastest method. For example, if you migrate a large amount of data. This method usually requires post processing. ■ Update Existing Feature Using Filter: Define an SQL filter expression that is added in the WHERE part of the SQL select statement. You can use the alias "a" for the table name. You can use placeholders for parameters. See Oracle Data Import: SQL Statements and Placeholder (page 252).
Filter	For example, use a filter to import an Interlis import project. For the Interlis data import, for each feature class, use the filter *IMPORT_ID = {IMPORT_ID}*. Then, the Interlis import, part 2, replaces the value of the parameter {IMPORT_ID} with the value corresponding to the selected Interlis Import project. See also Import Projects.
Comments	Stores any comments, for example, that explain why a method has been chosen.

Migration Model - Using a filter—For example, select the Update Existing Feature Using A Filter option, if you need to update existing data instead of inserting data.

Sample scenario: Database A stores parcel data, database B stores additional data for the same parcels. To migrate the parcel data, you execute two migration steps.

- Execute a standard Insert for database A.
- Execute an Update statement to add the additional data from database B. Enter the SQL select statement that defines the relation of the source table and the final table.

Use the alias "s" for the source table and the alias "t" for the final table. For Example:

```
s.fid = t.fid
s.flur=t.flur and s.stammnummer=t.stammnummer and s.unternum
mer=s.unternummer
```

Using these expressions, the following Update statement is used.

```
 update AL_F_FLUR t SET (NAME,KENNUNG) =
(SELECT NAME,KENNUNG FROM AL_F_FLUR s WHERE s.fid = t.fid)
```

In this case the attributes NAME and KENNUNG have been selected in the dialog box.

You can use placeholders, see Oracle Data Import: SQL Statements and Placeholder (page 252).

Oracle Data Import: Edit Feature Attribute Properties

In the mapping definition for feature attributes, you specify properties as shown in the following table.

Property	Description
Source Attribute Or SQL Statement For <source document name>	Displays the source attribute.
Final Attribute For <final document name>	Displays the final attribute.
Temporary Final Attribute Name	Specifies an alternative attribute name. For example, for FID relations and FID attributes.

	To store the data in a temporary attribute instead of the final attribute.
Include In Mapping Definition	Includes the settings in the mapping definition.
Map Attributes To Domains	Maps the selected attribute to a domain.
Post SQL	Defines an SQL select statement that is executed after the migration, using the connection to the final user. For example, use this option for FID relations. Use the following placeholders: ■ {SOURCEUSERNAME}—replaces the user name of the source user. ■ {TARGETUSERNAME}—replaces the user name of the final user. For example, to maintain the relation between a feature and its label.
Comments	Stores any comments.

Oracle Data Import: Domain Settings

Topobase domain tables have a fixed data structure. See Data Model: Domain Tables (page 183).

In the mapping definition for domains, you specify which attributes are displayed in the domain values list of the data model administrator.

Domain Settings dialog box	Description
Source Key Column	Specifies the primary key of the source domain table.
Final Key Column	Specifies the primary key of the final domain table. For a Topobase domain table *_TBD: ID.
Source Caption Column	Specifies the source caption.

Final Caption Column	Specifies the attribute that stores the caption in the final domain table. For a Topobase domain table *_TBD: VALUE.

Oracle Data Import: SQL Statements and Placeholder

Optionally you define SQL select statements that are executed before and after the Oracle Data import. For example, before the import, you disable triggers. After the import, you enable the triggers.

Placeholder—Inside the Filter statements, you can use placeholders. or example:

```
FID={MY_PARAMETER}
```

The placeholder name has to be in capital letters. You define the placeholder in the Set Placeholder dialog box, or you set the values in the API. API values overwrite the values that are set in the dialog box.

See also:

- Oracle Data Import: Edit Feature Class Properties (page 248)

Mapping Definition for Interlis Data Import

For Interlis data import to the Land CH data model, Autodesk provides customized Oracle Data import mapping definitions. You import the files (**.XML*) into your final document. If necessary, you can edit the configuration, save the modifications, and export the settings into an (**.XML*) file for further reuse.

To import a Oracle Data import mapping definition

1. Start Topobase Administrator, and open the workspace that contains the final document.
2. In the Administrator explorer, select the final document.
3. Click Document menu ➤ Import ➤ Oracle Data.
4. In the Source Document dialog box, select the source document, and enter its credentials. Click OK.
5. In the Import Oracle Data dialog box, click New.

6 In the Create New Mapping Definition dialog box, click Import.

7 Select the Oracle Data import mapping definition (*.XML) file.

8 Click Open.

9 If necessary, modify the configuration settings.

10 Click Close.

See also:

- Interlis Import
- Configuring Oracle Data Import (page 243)

Importing an Oracle Data Import Mapping Definition

You can import an Oracle Data import mapping definition (*.XML) file for reuse and modification.

IMPORTANT The document you import the file to must contain the Oracle Data Import Extension. See also Document Settings (page 7).

To import an Oracle Data import mapping definition

1 Start Topobase Administrator, and open the workspace that contains the final document.

2 In the Administrator explorer, select the final document.

3 Click Document menu ➤ Import ➤ Oracle Data.

4 In the Source Document dialog box, select the source document, and enter its credentials. Click OK.

5 In the Import Oracle Data dialog box, click New.

6 In the Create New Mapping Definition dialog box, click Import.

7 Select the mapping definition (*.XML) file.

8 Click Open.

9 If necessary, modify the configuration settings.

10 Click Close.

Exporting an Oracle Data Import Mapping Definition

You can export an Oracle Data import mapping definition into an (**.XML*) file.

To export an Oracle Data import mapping definition

1. Start Topobase Administrator, and open the workspace that contains the final document.
2. In the Administrator explorer, select the final document.
3. Click Document menu ➤ Import ➤ Oracle Data.
4. In the Source Document dialog box, select the source document, and enter its credentials. Click OK.
5. In the Import Oracle Data dialog box, select the mapping definition, and click Edit.
6. In the Edit Mapping Definition dialog box, click Export.
7. Specify a file name.
8. Click OK.
9. Click Close.

Topology Reference 3

Topology Introduction

Autodesk Topobase uses topologies to model objects and phenomena of the real world. Typically, a GIS user needs information about relationships between objects with topological characteristics. Modeling of land and other flat surfaces is a perfect application for area topology. Logical topology is appropriate wherever real world objects are connected to each other in networks.

- **Area topologies** are usually invoked to represent parcels in cadastral applications. Parcels are defined by their borders and should incorporate data specific to the entire parcel. The border in this model corresponds to a set of lines and the area to a polygon. With an area topology, you build areas (polygons) from a set of lines.
  An intersection can be performed with polygons as the primary feature class and polygons, line strings, and points as secondary feature classes. For intersections, you can use area topologies to build polygons from a set of lines for the primary feature class. See also Data Model: Intersections (page 224)

- **Logical topologies** connect features of any feature classes. The features need not to be spatially connected, and they need not to have geometry. For example, a logical topology connects points with points or lines with lines or lines to points or attribute features with attribute features. For example, a logical topology can represent a waste water network or electrical transmission lines.
  A **network topology** is a sub type of a logical topology that is based on an arc-node model and is based on geometry. Topobase utility models are based on logical topologies that connect points and lines.

- Topobase utility applications, such as Water, Gas, or Wastewater, use logical topologies that comply with the Topobase utility model. See also Utility Model Introduction (page 285).

Topologies are maintained at the client-side using feature rules.

See also:

- Area Topology Introduction (page 264)
- Logical Topology Introduction (page 269)

Getting Topology Started

Use Topobase data model administrator to

- Create feature classes.
- Create the topology and enable topologies for the feature classes.
- Initialize the topology, using an offline batch operation mode.

Use Topobase Client or Web to

- Explore and validate the topology using the topology checker.
- Process topologies in online operating mode.

See also:

- Data Model Administrator: Creating a Feature Class (page 160)
- Creating Topologies (page 256)
- Topology Offline Initialization Processing (page 281)

Creating Topologies

You create topologies in the Topobase data model administrator.

To create an area topology

1. Start the data model administrator.

2 In the data model explorer, select the Topologies node item, right-click and click Create Topology ➤ Area.

3 In the Area Topology dialog box, define the topology.

4 Define the properties and click OK.

The new topology is displayed in the data model explorer.

To create a logical topology

1 Start the data model administrator.

2 In the data model explorer, select the Topologies node item, right-click and click Create Topology ➤ Logical.

3 In the Logical Topology dialog box, define the topology.

4 See Logical Topology Properties (page 260)

For a Utility model, we recommend that you create the logical topology during the creation of the utility model. See also Creating a Utility Model (page 288).

After creating a topology, the participating feature classes become topology-enabled.

To initialize a topology

1 Select the topology. Notice, that in the right pane, the related system feature classes are displayed.

2 Right-click the topology and click Initialize Topology.

After the initialization, the topology is ready and can be processed in Topobase Client. From now on, the topology is maintained in online mode.

See also:

- Topology Online Processing (page 281)
- Topology Offline Initialization Processing (page 281)
- Utility Model Introduction (page 285)

Area Topology Properties

An area topology consists of two feature classes, centroid and line feature class, along with its various properties, as shown in the following table:

Area Topology Dialog Box	Description
Topology area	Contains general settings for the topology.
Name	Specifies the name of the topology.
Centroid Feature Class	Selects a centroid feature class (FC). A centroid stores the essential information of a polygon. Multiple topologies can share a centroid.
Line Feature Class	Selects the edge FC. One edge FC can participate in several topologies.
Tolerances area	
Arc Linearize	Sets the arc linearize value. The default value is 0.08. See also Arc Linearize Tolerance (page 259)
Advanced area	
Centroids Filter Expression	Selects the centroids to be used to build the area topology. Enter a a filter for the centroid feature class as follows: <column> = <value>, for example: ID_TYPE=24.
Lines Filter Expression	Selects the edges to be used to build the area topology.

See also:

- Area Topology Introduction (page 264)

Arc Linearize Tolerance

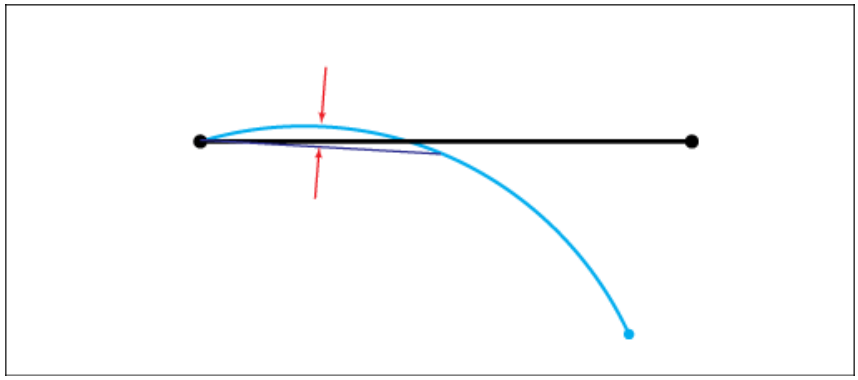
You can modify the default value for the arc linearize tolerance during area topology creation. After the topology is created, the setting cannot be altered without dropping and recreating the topology.

To view the arc linearize tolerance

1. Start the data model administrator.
2. In the data model explore, select the topology.
3. Right-click and click Properties.

The arc linearize tolerance helps to overcome impurity of the data, especially minor overlapping of arc segments. If overlapping is detected, then the arc segment is linearized in the same way that the respective segments, at the beginning or the end of the arc, are replaced by linear segments. If such linearization overcomes the topology error and overlapping disappears, the polygon is stored. Otherwise a topology error is returned.

There is a recommended value for arc linearize tolerance, which cannot be smaller than the spatial tolerance for the line feature class. It is recommended that the reference setting is about 2 – 5 times the spatial tolerance of the line feature class.



Topobase area topology: By setting the arc linearization tolerance, you can control overlapping of arcs and lines.

Logical Topology Properties

A logical topology consists of several feature classes, along with its various properties, as shown in the following table. To view the properties, in the data model administrator, select the topology, right-click, and select Properties.

Logical Topology Dialog Box	Description
Topology Name	Specifies the name of the topology.
Connection Tolerance Radius	Specifies the distance within which two features are considered as spatially connected. See also Connection Tolerance Radius (page 261).
Default Flow	Specifies the default flow between two features. You can modify this value at any time, using the Manage Connection dialog box. The default value is used, if the topology creates a new connection between two features. For example, if you digitize a new feature and the topology automatically connects it to existing features.
Define Logical Topology area	
Feature Classes In The Topology tab	Specifies the feature classes that are part of the logical topology. Select one or more feature classes.
Flow Per Feature Class	Specifies the attribute and the attribute values that store the flow information. See also Flow Direction Management (page 261).
Connectivity Between Feature Classes tab	Specifies the possible connections. For each feature class in the list on the left side, select one or more possible connected feature classes in the list on the right side.

For each feature class that is part of a logical topology, a button is added to the feature class form. Use the Connection <topology name> button to open the connectivity system table <topology name>_CONN.

NOTE For topologies that are used in a utility model, restrictions apply. See also Utility Topology Properties (page 262).

See also:

- Logical Topology Introduction (page 269)

Flow Direction Management

When you define a logical topology, you specify a default flow direction that is valid for all features in the topology. For example, you set the default flow direction to Backward, and in Topobase Client you use the Manage Connections dialog box to set specific flow directions.

Depending on your data, the flow information might already be stored with the feature data. Then, you can specify the feature class attribute that stores the flow direction along with the values that indicate the flow options. For example a road feature class uses the attribute DIRECTION to store the flow. Then you can specify that the attribute DIRECTION is used to define the flow direction.

To use flow information from a feature attribute

1. In the Logical Topology dialog box, click the Feature Classes In The Topology tab.
2. Select the feature class.
3. Under Flow Per Feature Class, select the feature class attribute that stores the flow direction.
4. In the data grid, for each flow option, enter a value that indicate the flow direction.
5. Click OK.

Connection Tolerance Radius

The connection tolerance radius determines when a feature is spatially connected to another feature. For example, when you digitize features using Topobase Client, the connection is created automatically if they lie within the tolerance.

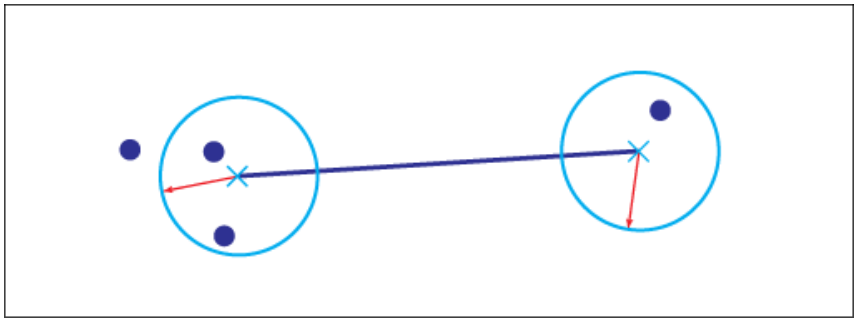
Also, during topology initialization, the connections of spatially connected features are automatically added to the connectivity table. See also Logical Topology Initialization (page 277).

You can modify the connection tolerance radius. The unit corresponds to the default length unit of the document. Example: If your spatial reference system (SRID) uses meters, the unit is meters.

If you modify the connection tolerance radius, you must initialize the topology again.

To modify the connection tolerance radius

1. Start the data model administrator.
2. In the data model explorer, select the topology.
3. Right-click and click Properties.
4. Modify the tolerance value.
5. Click OK and click Yes to initialize the topology.



The Connection Tolerance Radius determines whether a feature is connected to another feature. It is applied during topology initialization and online processing. For example, if there are two features in the tolerance radius, both features will be connected to the line. However, for a topology that is related to a utility model, this would cause a topology error of the type Line Has Two Start Points.

Utility Topology Properties

The Topobase utility model (page 526) requires a logical topology that is based on an arc-node model.

NOTE During the creation of a utility model, you can create an appropriate logical topology with properties as shown in the following table.

To view the properties, in the data model administrator, select the topology, right-click, and select Properties.

Logical Topology Properties (utility model)	Description
Topology Name	Specifies the name of the topology. The name corresponds to the name of the utility model.
Connection Tolerance Radius	Specifies the distance within which two features are considered as spatially connected.
Default Flow	Cannot be edited.
Define Logical Topology area	
Feature Classes In The Topology	Displays the utility feature classes. For a utility model, the topology consists of one line string feature class, and one point feature class. These feature classes store the edges and the nodes. Cannot be edited.
Flow Per Feature Class	Cannot be edited.
Connectivity Between Feature Classes	Displays the attribute feature classes that are related to the point geometry, and the line geometry feature classes. When you create the utility model, the points are connected to the lines, and the lines are connected to the points. After the utility has been created, when you edit the logical topology properties, you can define the connections using the attribute feature classes. However you cannot connect points to points, or lines to lines.

If you want to perform network tracing, the features must be connected, that means that they must have an entry in the connectivity table <topology name>_CONN.

See also:

- Utility Model Introduction (page 285)

Area Topology Introduction

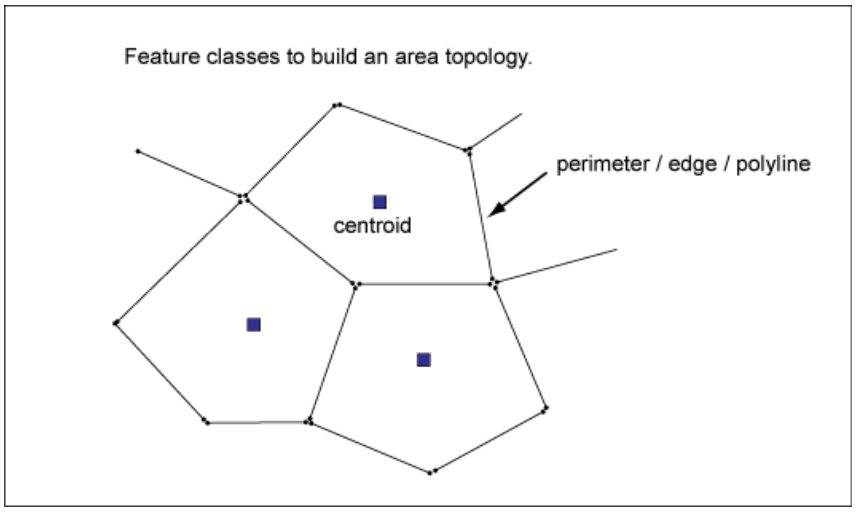
Autodesk Topobase defines area topologies with a line feature class and a centroid feature class. The line features are also referred to as edges. They are the master objects that store the primary data that determines the topological relationship. Topology routines create secondary redundant objects that can be recreated from master objects when needed. However, despite their redundancy, these secondary objects play important role in the representation and analysis of the topological configuration.

Topobase area topology supports two operation modes: online and offline.

The online, or interactive, mode is a response of the area topology subsystem to certain user actions. For example, an area is built when the edges make a topologically valid configuration after the user creates new edges or edits existing ones.

In the offline, or initialization mode, areas are built from all available edges as a batch. Offline mode is used mostly at topology creation time, when huge number of line strings and centroids are related topologically, but corresponding secondary objects, such as areas, are still missing.

An area topology consists of several lines or polylines that build an enclosed polygon. The polygon contains a centroid, which is the representative of the polygon. One specific quality of area polygons is the fact that they do not overlap and cover the whole area with an exhaustive and complete net, such as the way that land parcels cover the surface of the earth.



Area topologies consist of centroids and polylines. The polygon geometry (area) of the edges is generated with the topology.

To build an area topology you need two feature class types.

- Centroid (page 521) feature class where attributes of the polygon are stored.
- Line String feature class, which normally only comprises the geometry and has one attribute that can be used for stylization using the Display Manager. If you name the line string like the associated centroid and with the suffix _L, you can easily identify it. You can also define the line string feature class as a sub feature class.

You create both the centroid feature class, and the line feature class. Both feature classes will not be modified by the system.

When you create a topology, the system creates several support feature classes, such as a polygon (area) of the line strings <topology name>_TSUR. Do not store any polygon related information in the support feature class. These feature classes are maintained by the topology.

NOTE There is no entry in TB_RELATIONS. Centroids and line strings have no explicit relation.

See also:

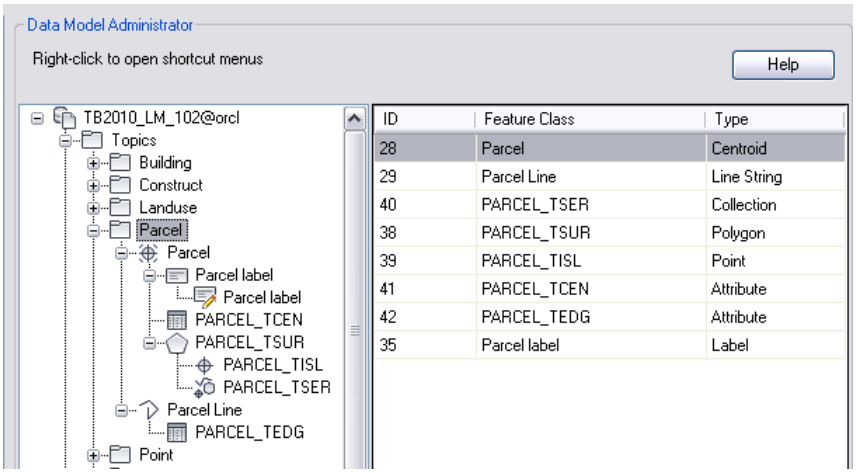
- Creating Topologies (page 256)

- Topology Online Processing (page 281)
- Area Topology Initialization (page 268)

Area Topology Tables

In addition to the line and centroid feature classes, each topology requires some volatile tables that hold secondary objects, such as areas generated, error reporting, and analysis facilities.

When a topology is defined in the data model administrator, a set of secondary feature classes is created. They all bear the name given to the topology, such as Parcel, and are differentiated by the suffix, such as Parcel_TCEN.



Topobase Data Model: Master and secondary objects of an area topology

Area Topology Tables	Description
Suffix _TSUR	Suffix, used to name a secondary feature class that contains generated areas (polygons). The table is updated by feature rules on the base feature class. See also Topology: _TSUR State Codes (Area) (page 267).
Suffix _TSER	Suffix used to name a secondary feature class that saves polygons with an invalid geometry, such as a Line Crosses Or Touches Self error. As long as invalid objects cause problems with spatial selects, such objects are

	moved from _TSUR to _TSER. There is no spatial index on the table, and invalid geometries do not cause any problems. **NOTE** Only polygons are processed here. if lines are not in a closed loop, no polygon can be created and there will be no entry in either the _TSUR or _TSER table.
Suffix _TEDG and _TCEN	Suffix used to name attribute type feature classes where information about topological edges and centroids is stored.
Suffix _TISL	Suffix used to name a point feature class that contains redundant data: polygons built in improper direction (clockwise instead of CCW). The geometry field matches the arbitrary vertex in the invalid polygon. These polygons are hole templates for building a polygon that contains an island.

Topology: _TSUR State Codes (Area)

State codes are saved in *_TSUR.STATE.

State Code	Description
NULL or 0	OK
17	General error (occurs when built polygon is heavily inconsistent)
	Any other value corresponds to an Oracle Spatial error code, for example:
13349	Polygon boundary crosses itself

More topology errors and their descriptions are displayed in the topology checker in Topobase Client or Web.

See also:

- Area Topology Checker (page 269)

Area Topology Initialization

Topobase area topologies can be processed in offline mode, during initialization. Before you can process an area topology in online mode, it has to be initialized. The initialization can be started manually in the data model administrator after you create a topology.

To initialize a topology

1. Start the data model administrator.
2. In the data model explorer, select the topology.
3. Right-click and click Initialize Topology.
4. In the Area Topology Initialization dialog box, optionally check one or all of the following options:
   - Create Missing Centroids—Creates a centroid feature that lies inside of the polygon.
   - Create Logfile Of Topology Errors—Logs the topology errors.
5. Click OK to start the initialization.

NOTE If the topology is job enabled, missing centroids cannot be created.

Topology initialization is mostly used at topology creation time, when a huge number of line strings and centroids are related topologically, but corresponding secondary objects, such as areas, are still missing. The procedure of topology initialization (= building the topology) includes the following steps:

- Filling a table *_TEDG with information about the edges
- Building the polygons *_TSUR from the edges
- Handle the islands of the polygons
- Mapping centroids to polygons (*_TCEN)

The polygon geometry is created for each area in counter clock-wise direction, except for the perimeter polygons which are clock-wise. Every island is surrounded by such a perimeter. Upon successful completion of the initialization, a spatial index on the _TSUR table is created. The *_TSER table stores topological constructs that failed to make a valid topological configuration.

Logging topology errors

During the initialization you can optionally log topology errors. The default location of the log file is *<topobase_client>\Log\<topology name>.log*. The log file contains the same errors that would be detected when you run the topology checker, however with better performance, and including intersecting lines. For example, select this option when you initialize the topology after you have imported a large amount of data. Then you can assess the quality of the data, without having to start Topobase Client.

For job-enabled topologies, errors are only logged for live data, but not for open and pending jobs.

Missing centroids

A Topobase area topology requires centroids, and the topology checker will identify missing centroids as errors. For example, centroids can be missing if you have imported data from another system that does not support centroids. Or if you have imported only the line features. To resolve this topology error, do one of the following.

- Digitize the missing centroids manually.
- In Topobase Administrator, initialize the topology.
- In Topobase Client, run the Area Topology Checker, and use the Create Missing Centroids command. See also Area Topology Checker.

A missing centroid is only created, if the polygon geometry is valid. For invalid polygons, correct the polygon geometry, and then create the centroid manually.

Area Topology Checker

See Topobase Client User Guide, section Area Topology Checker.

Logical Topology Introduction

Autodesk Topobase logical topologies support both spatial and attributive connectivity. For example, using logical topologies you can administer networks that have no geometry stored.

The Topobase applications such as Water, Wastewater, Gas, and Electricity are based on logical topologies that represent the utility networks.

Before you start working with a logical topology, you must configure the logical topology using the data model administrator.

To configure a logical topology (data model administrator)

- Create a logical topology. See Creating Topologies (page 256)
- Initialize the logical topology using the data model administrator. In the data model explorer, right-click the topology and click Initialize Topology. See also Logical Topology Initialization (page 277).

To work with logical topologies (Topobase Client)

- Start Topobase Client and open the workspace.
- Generate graphic.
- In the drawing, select a feature. On the main toolbar, click the Manage Connections icon.

See also the Topobase Client User Guide, section Manage Logical Connections.

See also:

- Topology Offline Initialization Processing (page 281)
- Utility Model Introduction (page 285)

Topology Checker

See Topobase Client User Guide, section Topology Checker (page 270).

Network Tracer

Tracing Topologies

You can define tracing templates for the topology feature classes. The following functionality is available:

- Shortest Path
- Minimum Spanning Tree
- Reachability

■ Cycle Detection

For topology analysis, you can use stop conditions and cost functions. Autodesk provides some predefined tracing models.

Logical topology analysis (tracing) can be performed either within predefined workflows or manually using the network tracer. The network tracer is available in both Topobase Client and Topobase Web.

To analyze a topology (in Topobase Client)

1. Start Topobase Client and open the workspace.
2. Generate graphic.
3. In the document explorer, expand the Topologies item.
4. Right-click the topology you want to trace and click Network Tracer.
5. In the Network Tracer dialog box, in the Tracing Template list, select a template.
6. Follow the on-screen instructions in the Features section. Depending on the tracing template, there are various workflows.
7. When you finish, click Trace.
8. The resulting features are highlighted in the map and listed in the Tracing Result dialog box.

The Tracing Result dialog box displays the features in an explorer tree, from which you can process the features. Expand the items, select a feature and use the buttons on the toolbar.

NOTE Tracing templates can be defined in the data model administrator.

Tracing Result Dialog Box	Description
	Opens the feature class form and shows feature information.
	Highlights the selected feature in the drawing.

	Unhighlights the feature.
	Zooms to the selected feature.
	Removes the selected feature from the explorer tree.
Tracing Template	Specifies a predefined tracing model (propagation). Tracing templates can be defined in the data model administrator. See also Logical Topology: Tracing Templates (page 272).
Start Features	Specifies a start node where the tracing shall begin.
Visit/End Features	Specifies an end node for the shortest path analysis.

If you are tracing an attributive topology, you can specify the start and end features in the attribute feature class form.

Logical Topology: Tracing Templates

You can define tracing templates and tracing conditions in the data model administrator. In Topobase Client, you can then use them as templates to execute network tracing, for example, in utility applications such as water and wastewater.

Start Topobase data model administrator, expand the topology, and right-click Tracing Templates.

Tracing Template dialog box	Description
Tracing Template area	General settings
Name	Specifies the template name. Use this name to select the tracing in the network tracer.

Tracing Type	See Logical Topology: Tracing Types (page 274).
Tracing Direction	Specifies the tracing direction: ■ Both Directions. ■ Forward ■ Backward The tracing direction configures which connection can be followed while tracing. For example, in the Manage Connections dialog box, set one or more connections to Forward or Backward. If you set a connection from feature A to feature B to Forward, the tracing can only follow one direction. If you set the connection to Both, the tracing can follow both directions.
Comment	Specifies a description of the tracing.
Highlight Features On Screen After Tracing	After the tracing, highlights the results in the drawing.
Show Feature Explorer After Tracing	After the tracing, displays the results in the feature explorer.
Template Is Enabled	Specifies that the template is available in the network tracer.
Tracing Conditions area	Lists current tracing conditions. Select one or more conditions which will be applied to the tracing.
New	Opens the Tracing Condition dialog box, where you can define a new tracing condition. See also Defining Tracing Conditions (page 275).

See also:

- Defining a Tracing Template for a Logical Topology (page 282)

Logical Topology: Tracing Types

You can define tracing templates using the following tracing types.

- **Shortest Path**—Use this type to find the shortest path between two nodes in an edge network. In this context "shortest" means " with the least effort" as you can use any attribute (cost attribute) to be minimized. If you minimize the length attribute, the result is the shortest path. If you set the cost attribute to a constant, the result is the path with the fewest connections.
- **Minimum Spanning Tree**—Use this type to find the minimum-cost tree that connects all nodes of a network or part of a network. This tracing type is available only for arc-node models. The minimum-cost tree minimizes the number of connections needed to connect all nodes with each other. If you assign a cost attribute, the tracing considers those values in finding the lines with minimum cost. For example, use this type when designing electrical networks.
- **Reachability**—Use this type to find all edges and nodes that can be reached from a start node, either directly (direct connecting edge) or indirectly (using several edges).
- **Cycle Detection**—Use this type to identify loops in a topology. For example, use this tracing type in Topobase Electric.

For each tracing type, you can define stop conditions or cost functions.

If no cost functions are defined, and if the feature has a LENGTH attribute (2D document) or a LENGTH3D attribute (3D document), the length is used for the cost. Otherwise, the feature has no costs (zero).

The tracing templates are stored in the system table TB_NET_TRACE_TPL.

See also:

- Logical Topology: Tracing Templates (page 272)
- Defining Tracing Conditions (page 275)
- System Table TB_NET_TRACE_TPL (page 89)

Defining Tracing Conditions

You can define any number of tracing conditions. They are applied in the tracing template definition.

Tracing Condition Dialog Box	Description
Tracing Condition area	General Settings
Name	Specifies the name. Use this name to select the condition in the Tracing Template dialog box.
Condition Type	Defines the type of the condition: ■ Select Cost Function for the Shortest Path or Minimum Spanning Tree analysis. For example, if you use the length of a line as a cost attribute, you can find the shortest path. ■ Select Stop Condition to determine a feature to constrain the tracing. For example, a valve that is closed.
Comment	Specifies a description.
Condition Is Enabled	Includes the condition in the trace. If you clear the check box, the trace ignores the condition.
SQL Statement	Select the feature classes you want to use in your condition. If the topology is used in utility model, the list displays the attribute feature classes. For standard topologies, the list displays the feature classes of the topology.
SQL Statement For Condition	Generates and edits the SQL statement. For editing click Generate to generate a skeleton SQL query. It is generated depending on the condition type and the selected feature class. See Also Tracing Condition SQL Statements (page 276).
Assistant	Opens the SQL Assistant: Optionally, you use the SQL Assistant to create SQL expressions. See also Creating SQL Expressions (page 103).

Validate	Validates the SQL select statement.
OK	Saves the condition.

The tracing conditions are stored in the system table TB_NET_SCOND_DEF.

See also:

- Logical Topology: Tracing Templates (page 272)
- Defining a Tracing Template for a Logical Topology (page 282)
- System Table TB_NET_SCOND_DEF (page 91)

Tracing Condition SQL Statements

Use the Tracing Condition dialog box to define SQL statements that control the network tracing. SQL and data model knowledge is required.

Enter the statement directly or click Generate to generate a skeleton SQL query. Then, click Assistant to open the SQL Assistant.

- **Cost function**—SQL statement that returns a number; it may return NULL if the cost does not apply. This is rated as zero costs.
- **Stop condition**—SQL statement that returns NULL if the condition does not apply. Then, the feature is not a stop feature. If the condition returns any other number, the feature is a stop feature.

SQL Skeleton query	Description
Example: Utility tracing with stop condition	`case /* Point */ when &f_class_id=33 then ( select case /* Armature */ when f_class_id_attr=3 then ( select 1 from GA_ARMATURE where fid=ga_point.fid_attr and <CONDITION>) else null end from GA_POINT ga_point where fid=&fid) else null end`
Example: Non-utility tracing with stop condition:	`case /* <feature class> */ when &f_class_id=<id> then ( select 1 from`

	`<feature class> where fid=&fid and <CONDITION>) else null end`
<CONDITION>	Place holder used in a stop condition. Replace this value with the attribute that contains the information when you want to stop the tracing. Example: The attribute OPEN = YES indicates open armatures. `case /* Point */ when &f_class_id=33 then ( select case /* Armature */ when f_class_id_attr=3 then ( select 1 from GA_ARMATURE where fid=ga_point.fid_attr and not OPEN = 'yes') else null end from GA_POINT ga_point where fid=&fid) else null end`
Example: Utility tracing with cost function	`case /* Point */ when &f_class_id=33 then ( select case /* Armature */ when f_class_id_attr=3 then ( select <VALUE> from GA_ARMATURE where fid=ga_point.fid_attr) else null end from GA_POINT ga_point where fid=&fid) else null end`
Example: Non-utility tracing with stop condition	`case /* <feature class> */ when &f_class_id=137 then nvl(( select <VALUE> from <feature class> where fid=&fid ),0) else 0 end`
<VALUE>	Placeholder used in a cost function. Replace this value with the cost attribute to be minimized in the tracing. Example: `PIPE_LENTH`
select 1	Helper select for a stop condition to get a return value that is not NULL if the conditions applies.

Logical Topology Initialization

Topobase logical topologies must be initialized before you can use them, for example, in a utility model (page 526). Start the initialization manually in the data model administrator.

IMPORTANT Initialization only considers spatial connectivity. It is recommended that after the initialization you check the logical topology in Topobase Client using the Manage Connection dialog box.

To initialize a logical topology

1. Start the data model administrator.
2. In the data model explorer, select the topology.
3. Right-click and click Initialize Topology

For each possible connection, the system tries to spatially connect the features when they are digitized and lie within the Connection Tolerance Radius.

Spatial initialization—When you initialize the topology, the system fills the <logical topology name>_CONN system table with the connectivity information.

Spatial online mode—In Topobase Client, when you digitize new features in Topobase Client, the connectivity table is maintained automatically, where spatial connection is possible.

To check the connectivity table after initialization

1. Start Topobase Client and open the workspace.
2. Generate graphic.
3. Click Manage Connections, and explore the connectivity of the features.

Optionally, in the feature class form, click the Connection <topology name> button to open the connectivity table <topology name>_CONN, and check for logical, non-spatial connectivity.

Network Topology Update

The topology concept has been updated with the previous version Topobase 2009. When you open a workspace that has an old topology structure for the first time, the data structure will be updated. Network topologies and attributive network topologies are automatically upgraded into logical topologies. With Autodesk Topobase 2009, network topologies have been replaced by logical topologies.

The self-acting upgrade of a network topology <topology name> into a logical topology performs several steps.

- Creating the connectivity table <topology name>_CONN.
- Transferring connectivity information that is stored in the system tables <topology name>_NLNK and <topology name>_NNOD, and in the FID_<name>_FROM and FID_<name>_TO attributes to the connectivity table <topology name>_CONN.
- Dropping the system tables and attributes.
- Defining the following logical topology properties.
  - All point attribute feature classes can only be connected to the line attribute feature classes.
  - All line attribute feature classes can only be connected to the point attribute feature classes.
  - Default flow: Forward.
- Adjusting the forms of the participating feature classes.

IMPORTANT After the upgrade, we recommend that you check the following.

- **Tracing Conditions**—Check the feature class names and attributes names that are used in the tracing condition SQL statement. See Updating Tracing Conditions (page 279).
- Reports
- Display Models.

Updating Tracing Conditions

Network tracing conditions cannot be updated automatically. If you are working with a document that has been updated from a previous version, you must manually convert existing tracing conditions. For example, you must convert the tracing conditions, if you import an existing Topobase 2008 document.

To convert a network tracing template (cost function)

1 Open the data model administrator.

2 In the data model explorer, select the tracing condition, and click Properties.

3 Review the SQL Statement For Condition, and find the feature class name and the SQL expression. For example, a previous tracing condition is: `select PIPE_LENGTH from WA_PIPE where FID = L.FID_ATTR`, where PIPE_LENGTH is the expression, and WA_PIPE is the feature class.

4 Copy the expression to the clipboard.

5 Under SQL Statement, select the feature class, such as Pipe. Click Generate.

6 Replace the placeholder <VALUE> by the expression that you copied.

```
case /* Line */ when &f_class_id=22 then
( select case /* Pipe */ when f_class_id_attr=30 then
( select PIPE_LENGTH from WA_PIPE where fid=wa_line.fid_attr
 ) else null end from WA_LINE wa_line where fid=&fid
) else null end
```

7 Click Validate.

To convert a network tracing template (stop condition)

1 Open the data model administrator.

2 In the data model explorer, select the tracing condition, and click Properties.

3 Review the SQL Statement For Condition, and find the feature class name and the SQL expression. For example, a previous tracing condition is: `select 1 from WA_VALVE where FID = N.FID_ATTR AND ID_VALVE_STATUS = 3`, where ID_VALVE_STATUS = 3 is the expression, and WA_VALVE is the feature class.

4 Copy the expression to the clipboard.

5 Under SQL Statement, select the feature class, such as Valve. Click Generate.

6 Replace the placeholder <CONDITION> by the expression that you copied.

```
case /* Point */ when &f_class_id=31 then
( select case /* Valve */ when f_class_id_attr=45 then
( select 1 from WA_VALVE where fid=wa_point.fid_attr
AND ID_VALVE_STATUS = 3
 ) else null end from WA_LINE wa_line where fid=&fid
) else null end
```

7 Click Validate.

See also:

- Logical Topology: Tracing Templates (page 272)

Topology Offline Initialization Processing

Topobase topologies support two operation modes: Online and Offline.

In offline or initialization mode, you build topologies from all available elements as a batch. Offline mode is used mostly at topology creation time, when a huge number of features are related topologically, but corresponding secondary objects, such as areas, are still missing.

See also:

- Area Topology Initialization (page 268)
- Logical Topology Initialization (page 277)

Topology Online Processing

Topobase topologies support two operation modes: Online and Offline.

In online or interactive processing mode, the topology subsystem responds to certain user actions, causing a specific result. For example, an area is built when the edges make a topologically valid configuration, after the user creates new edges or edits existing ones.

Client-side feature rules control the online processing of area topologies that occur during user interaction with topology-enabled master feature classes (centroids and/or edges).

Topology Tutorial

In this tutorial, you will define a template and a condition for tracing a network.

Defining a Tracing Template for a Logical Topology

You can define templates for network tracing.

Required: logical topology (page 523). It is also useful to define the tracing conditions before you start to define the tracing templates.

See also:

- Defining Tracing Conditions (page 275)

To define a tracing template

1. Start Topobase data model administrator.
2. In the data model explorer, select the logical topology, expand it, right-click Tracing Templates and click Create Tracing Template.
3. Enter a name and comment.
4. Select the tracing type and tracing direction.
5. Select the option Template Is Enabled.
6. Select the tracing conditions. If no conditions are available, click New to define a condition.
7. Click OK to finish the definition. The tracing template is added to the data model explorer.

The tracing can be executed in Topobase Client and Web from the network tracer.

See also:

- Network Tracer (page 270)

Defining a Tracing Condition

To define a tracing condition

1. Start Topobase data model administrator.

2 In the data model explorer, select the topology, expand it, right-click Conditions and click Create Condition. (Alternatively, click the New button in the Tracing Template dialog box).

3 Enter a name and comment.

4 Select the Condition Type.

5 Select the feature classes that will be used in the condition.

6 Click Generate to generate a skeleton SQL query.

7 Click Assistant, and edit the query.

8 If you are finished selecting and creating conditions, click OK.

See also:

- Tracing Condition SQL Statements (page 276)

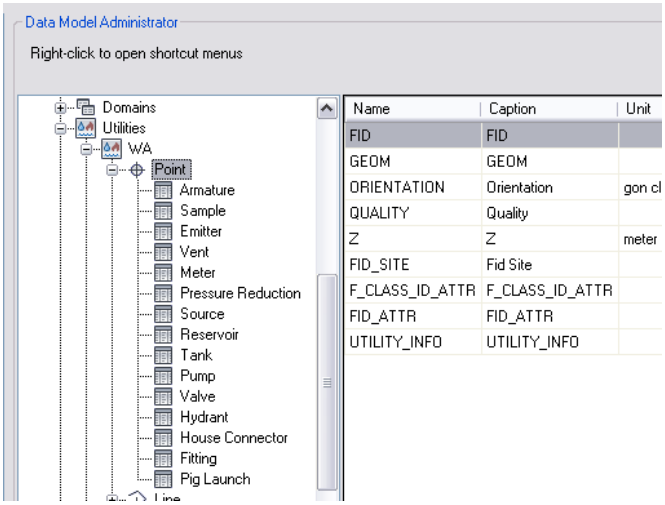
Utility Reference 4

Utility Model Introduction

The Topobase utility model provides basic utility network components, which are used in the utility applications, such as Water, Gas and Wastewater. It splits the network geometry (connectivity) from the attribute data.

The utility network consists of nodes and edges that represent the connectivity. The utility model stores this network in one point feature class (nodes) and one line string feature class (edges) and relates the geometry to the different utility objects (attribute data). When you create a utility model, you choose to create the appropriate logical (network) topology at the same time.

Example: In the Water application, the utility node feature class has a multi-relation to all point attribute feature classes, such as armature, emitter, vents, meters and so on. The utility edge feature class has a relation to the line attribute feature class Pipe.



Utility data model: Water node feature class Point with its relations to the object feature classes; shown in the data model explorer.

This structure is not displayed to the user. However, the user can digitize the attribute feature because of special utility feature rules. Then the geometry features are processed.

The concept of separating geometry and attribute storage allows for two different line splitting methods, soft and hard split. These methods determine how the system deals with the attribute data of the two resulting features.

- Soft Split—The line is split into two separate features. Each has its own geometry, but both use the same attribute data.
- Hard Split—The line is split into two separate features. Both have their own geometry and both have their own attribute data.

For more information about the utility applications refer to the respective Topobase Application Guides, such as the Topobase Water User Guide.

See also:

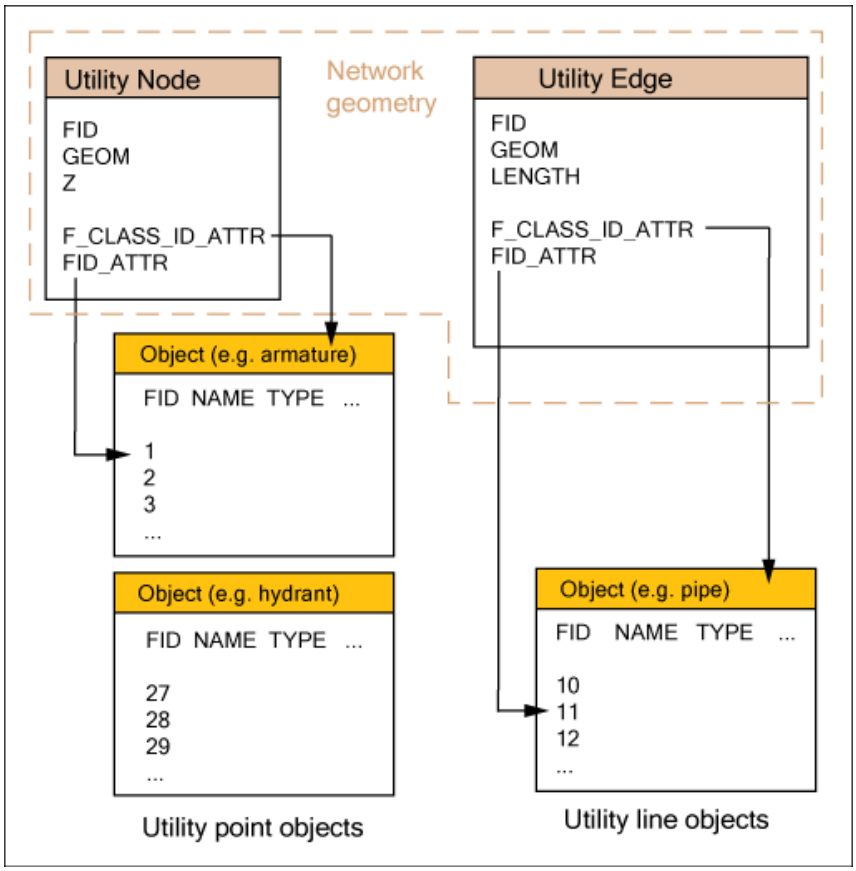
- Logical Topology Introduction (page 269)

Utility Data Model

A Topobase Utility data model consists of nodes and edges that are stored in two geometry feature classes and attribute data that is stored in attribute feature classes. Each utility feature is described by a geometry feature, which has no specific attributes (and should not be enhanced) and its related attribute features. This means that the node type is determined by the related attribute feature.

The two geometry feature classes consist of the following:

- Utility edges—One utility edge feature class stores all network lines. Every edge has a relation to its related attribute feature.
- Utility nodes—One utility node feature class stores all network points. Every node has a relation to its related attribute feature.



Utility data model

Creating a Utility Model

Use the Topobase data model administrator to create utility data models. The Create Utility Model command creates the relation between the geometry and the attribute feature classes and adds utility system attributes.

Requirements

- Attribute feature classes for the utility point and line objects, such as water house connectors and armatures
- Point feature class for the nodes (geometry)

- Line string feature class for the edges (geometry)
- Logical topology, see Utility Topology Properties (page 262).

These components can be created during the Create Utility model process, as shown in the following table:

Create Utility Model Dialog Box	Description
General area	General settings for the utility model
Name	Specifies a name for the utility model.
Feature Classes area	Specifies the utility feature classes. Use the tabs to select the feature classes that store the network geometry.
Point tab	Specifies the point feature class that stores the nodes.
Line String tab	Specifies the line string feature class that stores the edges.
Attribute Feature Classes	Specifies the attribute feature classes that store the utility objects (attribute data). Select one or more.
Create New <…> Feature Class	Creates a new feature class. Opens the Create Feature Class dialog box, where you can create a new feature class. Default names are <model name>_POINT <model name>_LINE
Logical Topology area	Specifies a logical topology.
Topology	Lists all logical topologies in which the selected geometry feature classes participate. See also Logical Topology Introduction (page 269).
Create New Logical Topology	Creates a logical topology with the geometry features. Opens the Logical Topology dialog box, where you can define a new topology. Default

	name is <model name> with the connectivity table <model name>_CONN.
OK	Creates the model. This button is only active if all necessary entries are made.

In addition to the default attributes, the utility node and utility edge feature classes have some system attributes, as shown in the following table:

Attributes of Node and Edge Feature Classes	Description
	Utility edge and node feature class (network geometry)
F_CLASS_ID_ATTR	Foreign key: indicates the related attribute feature class; TB_DICTIONARY.F_CLASS_ID
FID_ATTR	Foreign key: indicates the related attribute feature
	Utility edge feature class (network geometry)
UTILITY_INFO	Defines the appearance in the drawing (only lines)

Do not add more customer attributes because these tables are processed automatically.

NOTE The foreign key to the attribute tables is defined as multi relation. With this we can relate one geometry feature class to several attribute feature classes. Example: A water point feature can be related to either of the point attribute feature classes: armature, valve, hydrant.

See also:

- Creating a Utility Data Model (page 291)

Utility Feature Rules

The Topobase utility feature rules are used to keep the database consistent. You can view and edit the applied feature rules in the data model explorer.

To view the feature rules

1. Start the data model administrator.
2. In data model explorer, select the feature class, such as Pipe in the Water application.
3. Right-click and click Edit Feature Rules.
4. Click the Client Side (.Net) tab.

For more information about the utility feature rules, see the Topobase Feature Rule Reference, section Utility Feature Rules.

See also:

- About Feature Rules (page 511)

Utility Model Tutorial

Creating a Utility Data Model

To create a simple utility data model

1. Start Topobase Administrator.
2. Open the workspace (page 526) that contains the document (page 521).
3. In the administrator explorer, in the left pane select the document.
4. Click Document menu ➤ Data Model.
5. Create two feature classes to hold the geometries:
   One feature class of type line string and one of type point.
6. Create attribute feature classes to hold the data. They can be in the same topic as the geometries or in different topics.
7. Select the Utilities item. Right-click and click Create Utility Model.
8. Enter a name, such as Gas.
9. Select a point feature class and the matching attribute feature classes, such as Gas_point and Armature, Filter, Siphon.

10 Select a line feature class and the matching attribute feature classes, such as Gas_line and Pipe.

11 Select a logical topology. If no topology is available, click the Create New Logical Topology link and click OK in the Logical Topology dialog box.

12 Click OK to create the utility model.

13 Initialize the newly created topology. In the data model administrator, select the topology, right-click and click Initialize Topology.

14 Define tracing conditions. In the data model administrator, expand the topology, and select Conditions. Right-click and click Create Condition. See also Defining Tracing Conditions (page 275)

15 Define a tracing template. the data model administrator, expand the topology, and select Tracing Templates. Right-click and click Create Tracing Templates. See also Logical Topology: Tracing Templates (page 272)

Defining a Label for a Utility Object

You can define labels to inscribe the attribute data of the network objects. Notice, that in the utility model, a network object itself has no geometry, but is related to a geometry feature class (node or edge).

To place label texts using the auto label function, you must define the labels using a geometry select.

To define a label for a utility object

1 Start Topobase Administrator and open the workspace.

2 In the administrator explorer, select the document.

3 Click Document menu ➤ Data Model.

4 In the data model explorer, select the network feature, for example Pipe.

5 Select the label feature class. Right-click and click Create Label Definition.

Set the following label properties:

Property	Description
Content To Display (Select Statement)	This select statement queries the FID. Example: `Select FID from GAS_PIPE where FID = $id`
Parent Geometry (Select Statement)	This select statement queries the geometry of the related feature (FID) that has been found by the select statement above. Example: `Select FID, GEOM from GAS_LINE where FID_ATTR = ?`
Label Information	Select Automatically Create Label When Feature Is Created.
	Select Label Is Active.
Orientation	Select Preserve Text Readability.

See also:

- Label Select Statements (page 200)

Jobs Reference 5

Jobs Introduction

Based on the Oracle® database, Autodesk delivers a long transaction and versioning framework that meets the specifics of the cadastral applications, Topobase Jobs.

Topobase Jobs are based on Oracle's VPD (Virtual Private Database) technology. This makes it possible to have multiple states for an object in the database, such as live, pending, and open.

Data may change only within a job; whereas data outside a job is read-only. Within the Topobase Jobs data may be Live, Pending, Open (project data). Live data is read-only and common for entire enterprise (visible identically in all jobs, and outside jobs, also).

In addition, any user defined job state can be defined.

- **Open**—Once a data modification process is initiated, any data change is isolated within that job and is not visible in other jobs and outside jobs. The job is set to open. Feature locking is either Pessimistic Feature Locking, or Optimistic Feature Locking.
- **Pending**—When changes are ready for review and approval, the job (including all altered data) is set to Pending.
- **Live**—When the work is approved and the job is ready for production, the job is set to Live.
- **Other states**—You can define additional job states.

Topobase provides different UIs to configure and process jobs.

- Topobase Administrator provides Job Administrator.
- Topobase Client and Web provide Job Manager and Job Explorer.

See also:

- Getting Jobs Started (page 296)
- Jobs Feature Locking (page 301)
- About Jobs And Job Templates (page 306)
- About Job States (page 308)

Getting Jobs Started

Before you start working with jobs in a document, you must configure jobs using Topobase Administrator, and create jobs, using Topobase Client or Web.

To configure jobs (Topobase Administrator)

1. Start Topobase Administrator.
2. **Job enable** the document. Optionally assign a perimeter feature class. See also Job Enabling (Document Settings) (page 299).
3. Use the job administrator to create **job templates**. See also Creating a Job Template (Job Administrator) (page 304).
4. Use Display Models to style the features that are part of a job. See Styling Job-Enabled Features (page 412).

To work with jobs (Topobase Client)

1. Start Topobase Client.
2. Start the Job Manager.
3. Create a job, using a job template. After a job is created, it has the job state Open and you can start working.

NOTE The job toolbar is only visible if the document has job enabled feature classes.

See also:

- Job Administrator (page 298)

See also Topobase Client User Guide, section Job Manager Menus and Icons.

Jobs Update

The previous version Topobase 2009 provided a new job concept. When you open a Topobase workspace containing the old job concept, you can update the jobs to the new concept.

Topobase Job Concept

- **Job Enabling**—You can only job enable the whole document. Job enabling of single feature classes or topics is no longer possible.
- **Job Perimeter**—You can use one perimeter feature class for the whole document. Or you can assign a perimeter feature class for each topic. If necessary, use the Edit Feature Class command to assign a perimeter to a single feature class.
- **Feature Locking**—You can set the feature locking type either to Pessimistic Locking, or Optimistic Locking. The previous version supported Optimistic locking.

When you open an existing workspace for the first time, the document structure is updated automatically. For job enabled documents, if the job configuration does not match the new job concept, a message informs you that you must repair the job configuration.

You use the Document Settings to repair an invalid job configuration.

To repair the Job Document Settings

1. Start Topobase Administrator, and open the workspace.
2. In the administrator explorer, select the document.
3. Click Document menu ➤ Settings. On the navigation pane, click Jobs.

   Topobase checks the job configuration. For existing documents, that have been processed using Topobase 2008, the Feature Locking Type is set to Optimistic Feature Locking.

4 If different perimeter feature classes are detected in one topic, the Perimeter Feature Class is set to Advanced Settings (See Below). Click to open the Advanced Settings. Select one perimeter feature class per topic.

If necessary, you are prompted to repair the job configuration. Click Repair to job enable the whole document.

To change the feature locking type, see Jobs Feature Locking (page 301).

See also:

- Assigning a Job Perimeter (page 302)
- Jobs Feature Locking (page 301)

Job Administrator

Using the Topobase Job Administrator, you can define and edit job templates, assign job templates to user groups, and administer job transition passwords.

To start the Job administrator

1 Start Topobase Administrator.

2 Open the workspace.

3 Click Document menu ➤ Job Administrator.

4 On the right pane, in the explorer tree, expand Jobs and Job Templates to see all available templates.

In the explorer tree, use the buttons or shortcut menus to process the job templates.

Buttons and Commands	Description
Create	To create a job template. See also Creating a Job Template (Job Administrator) (page 304)

Edit	To edit a job template, to assign the template to a user group, and select feature classes.
Delete	To delete the job template.
Password	To change the passwords that allow a user to change the job states. For each transition you can define different passwords. See also About Job States (page 308)
Assign Templates To User Groups	Opens the Set User Groups For Job Templates dialog box. Select a user group, and assign the templates. See also About Jobs And Job Templates (page 306).

Configuring Jobs

Job Enabling (Document Settings)

To work with jobs, you must job enable the document. When you job enable the document, you lock the features so that they can only be modified within a job.

Job enabling comprises the following steps.

- Setting the document to job enabled. You either create a job enabled document, or you job enable an existing document later. Also, you can modify this setting.
- Specifying the feature locking type.
- Optionally, assigning perimeter feature classes.

NOTE You job enable a document when you create the document. However the perimeter settings must be set after job enabling, because only job enabled feature classes can be assigned as a perimeter feature class.

Job enabling starts certain routines. For example, it adds a JOB_VERSION attribute to each feature class and writes Topobase Jobs metadata (TB_JOB* system tables).

WARNING Do not edit the Topobase Job system tables manually.

You use the job enabled feature classes to define any number of job templates. Job templates are defined in the Topobase Job Administrator.

See also:

- System Tables TB_JOB_* (page 81)
- Creating a Job Template (Job Administrator) (page 304)

Creating a Job Enabled Document

Use the Job Document Settings to job enable a document.

1. Create the database. See Data Model Administrator: Creating a Database (page 144).
2. Click Document menu ➤ Settings. On the navigation pane, click Jobs.
3. On the right pane, select Enable Jobs (Versioning) For This Document.
4. Under Feature Locking Type, select the locking type. See also Jobs Feature Locking (page 301).
5. Click OK to create the document.

Optionally, after the document has been created, you can assign the perimeter feature classes. See Assigning a Job Perimeter (page 302)

Job Enabling a Document

You can job enable a document at any time. Use the Job Document Settings to job enable a document.

To job enable a document

1. In the administrator explorer, select the document.
2. Click Document menu ➤ Settings. On the navigation pane, click Jobs.
3. On the right pane, select Enable Jobs (Versioning) For This Document.
4. Under Feature Locking Type, select the locking type. See also Jobs Feature Locking (page 301).
5. Click Save.

Optionally, assign the perimeter feature classes. See Assigning a Job Perimeter (page 302)

Job Disabling a Document

If all jobs are Live, you can job disable a document. That means that no job must be Open or Pending. For example, if you want to modify the feature locking type, you must job disable the document.

Use the Job Document Settings to job disable a document.

To job disable a document

1. In the administrator explorer, select the document.
2. Click Document menu ➤ Settings. On the navigation pane, click Jobs.
3. On the right pane, clear Enable Jobs (Versioning) For This Document.
4. Click Save.

NOTE If the check box Enable Jobs (Versioning) For This Document is disabled, the document contains open or pending jobs.

See also:

- Jobs Feature Locking (page 301)

Jobs Feature Locking

Topobase supports two types of feature locking, where the main difference is the handling of job conflicts. A job conflict occurs if the same feature is modified in different jobs.

Pessimistic Feature Locking

Feature Locking type where a feature can only be modified in one job. Use Pessimistic Feature Locking to detect and solve any job conflicts as early as possible. For example, in Job A you add a transformer on a pole, and in Job B another user B selects the same pole to delete it from the database. Then, with pessimistic feature locking, user B immediately receives the warning that the feature is locked, rather than at job commit time. User B cannot modify the feature.

To solve the job conflict, user B would contact you, and you can jointly decide how to proceed. For example, you can use the Partial Posting command to extract the respective features from your Job A, so that it can be modified in Job B.

You use Partial Posting to extract a subset of features from the current job in order to change its state to Live immediately. Then the modified features can be modified again in other jobs.

For more information about Partial Posting, see Topobase Client User Guide.

Optimistic Feature Locking

Feature locking type, where a feature can be modified in different jobs. Use Optimistic Feature Locking, if you want to allow modifications, even if conflicts occur. With Optimistic Feature Locking, job conflicts will not be detected immediately, but only when you change the job state. After the first job has been merged forward, and when the second job is being merged forward, the Job Conflict Manager appears.

To change the feature locking type

Requirement: No job must be Open or Pending.

1. In the administrator explorer, select the document.
2. Click Document menu ➤ Settings. On the navigation pane, click Jobs.
3. On the right pane, clear Enable Jobs (Versioning) For This Document. If the document contains Open or Pending jobs, this option is deactivated.
4. Click Save to job disable the document.
5. Select Enable Jobs (Versioning) For This Document.
6. Select the new Feature Locking Type.
7. Click Save.

Assigning a Job Perimeter

A job perimeter is a polygon feature class that limits the area where a certain job shall be processed (job perimeter (page 523)). Perimeters are mainly used in Survey applications; however, they may be introduced in arbitrary applications as well. The perimeter feature class stores the job perimeter of the respective jobs. You either assign one perimeter for all feature classes, or you assign a different perimeter feature classes for single topics.

NOTE The use of job perimeters is optional. However, if you assign a perimeter feature class and then create a job, you also must select a job perimeter.

With job perimeters, you can control the spatial area within a certain job, where modifications are allowed. Also, you can define feature rules to be applied on the objects with a perimeter, such as a perimeter check.

We recommend that you assign a polygon feature class that is used only for perimeters. If necessary, create a new polygon feature class for that purpose.

To assign perimeter feature classes

1. In the administrator explorer, select the document.
2. Under Document, on the navigation pane, click Jobs.
3. Under Perimeter Settings, enter the options as shown in the following table.

Perimeter Settings	Description
Perimeter Feature Class	Optional: Assigns a perimeter feature class. Lists all available polygon feature classes of the current document. Feature Classes that belong to a topology or an intersection are not listed.
Additional Buffer Around Perimeter	Assigns a buffer. Notice, that the unit is the length unit as set in the document settings. You can enhance the perimeter polygon with the buffer.
Advanced Settings	Assigns a perimeter feature class to a topic. Click if you do not want to use the same perimeter for all topics, but if you want to use different perimeter feature classes for some or all topics. The advanced settings override the general setting.

If you have assigned a perimeter feature class, the Perimeter Check feature rule has been activated automatically. This is one of the predefined Topobase server-side feature rules. It checks whether a feature is inside the selected perimeter.

To view the perimeter check feature rule

1. In the data model explorer, select the feature class.
2. Right-click and click Edit Feature Rules.
3. Click the Server Side (PL/SQL) tab. The Check Perimeter rule is applied.

See also:

- About Feature Rules (page 511)

Topobase Client User Guide, Defining the Job Perimeter

Creating a Job Template (Job Administrator)

Before you start working with a job, you must specify which of the job enabled feature classes you want to work with. Normally, you work with the same grouping of feature classes, so it is helpful to pre-select those feature classes. Do this by creating job templates.

NOTE Use the job templates in the Topobase Job Manager to create the jobs.

You can define job templates with the Topobase Job Administrator. It is required that before you define job templates, you must job enable the document.

To create a job template

1. Start Topobase Administrator and open the workspace.
2. Select the document and click Document menu ➤ Job Administrator.
3. In the Job Administrator explorer, expand Jobs and select Job Template.
4. Right-click and click Create.
5. In the Job Template dialog box, click the Feature Classes tab.
6. Select the topics and feature classes you want to add to the template.
7. Click the User Groups tab.
8. Assign the job template to the user groups that are allowed to use the template.
9. Click OK. The new template is added in the explorer tree.

To change the user group of a job template

1. Start Topobase Administrator, and open the workspace.
2. Select the document, and click Document menu ➤ Job Administrator.
3. In the explorer tree, select the job template, and click Edit.
4. In the Job Template dialog box, click the User Groups tab, and assign the user group.

Creating a job template means combining job enabled feature classes to process together in one job. You can combine feature classes from different topics into one job template. When you work with a job that has been created from this template, you can change the state of each topic separately.

See also:

- Job Enabling (Document Settings) (page 299)
- About Jobs And Job Templates (page 306)

Topobase Client User Guide, section Job Manager Menus and Icons

Job Leasing

By default, data that is processed within a job can only be accessed by the Topobase application. Using Job Leasing, you can make data available to other applications also, for example to SQL plus or import routines. Without job leasing, data is not available until the job state has been set to Live.

Use the Topobase Job Manager to manually lease a job.

To lease a job manually

1. Start either Topobase Client or Web and open the workspace.
2. Start the Job Manager.
3. In the Job Manager explorer, select the job, right-click and click Lease Job.
4. In the Lease Job dialog box, specify the end date of the lease.

To enable automatic job leasing

Use the Jobs application options to specify that the selected job is always leased automatically.

1. Start either Topobase Client or Web.
2. Click Setup menu ➤ Options.
3. In the tree view, click Jobs.
4. Select Automatically Lease The Selected Job.
5. Define the duration of job leasing.

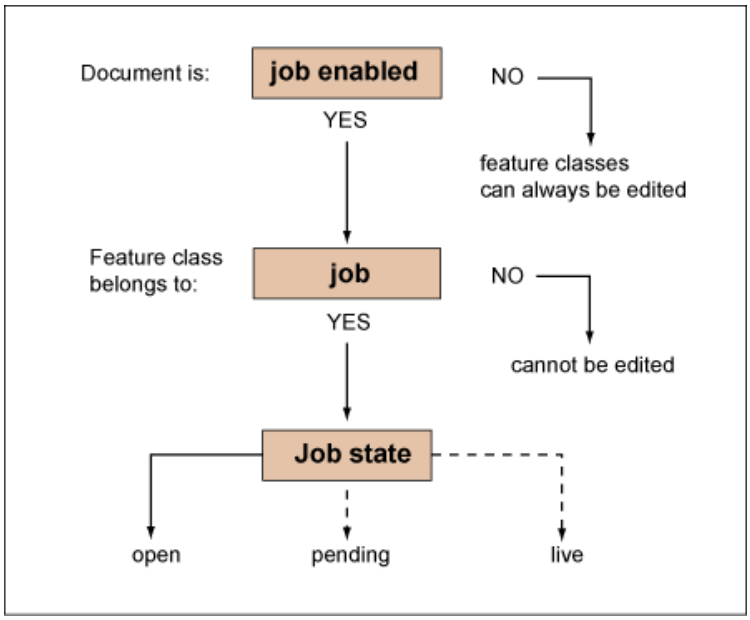
See also:

- Setting Jobs Application Options (page 21)

About Jobs And Job Templates

To work with job-enabled feature classes you must create a job that contains these feature classes. You can use job templates to create the jobs.

A job template contains pre-defined feature classes that can be processed in the jobs that are created from this template. Other job enabled feature classes, which are not in this job, may not be modified or deleted in that job.



Job enabled feature classes can only be edited if they belong to the current job. A job can only be processed if its state is open.

You can define an arbitrary number of job templates with any combination of job enabled feature classes.

Job templates and user groups

You can assign job templates to one or more user groups. For example, to improve security, a user group is only allowed to use a limited set of job templates. In Topobase Client, members of this user group can only work with the assigned jobs templates, and with the jobs that have been created using these templates.

NOTE Members of the Administrator group can access all job templates, and process all jobs. When you create a new user group, by default all job templates are available to the members of the new group.

In the Job Administrator, use the Assign Templates To User Groups command to assign unassigned job templates to user groups.

NOTE Existing jobs from previous versions will not be assigned to a user group, so they can be used by everyone.

See also:

- Job Administrator (page 298)
- Creating a Job Template (Job Administrator) (page 304)

About Job States

Job states and their transitions can freely be defined. They are stored in the system tables TB_JOB_STATE and TB_JOB_STATE_TRANSITION.

When you job enable the document, you lock the features so that they can only be modified within a job (long-term transaction). Any feature modification is isolated in the job, and after approval you can change the job state of all modified features to Live.

Default job states

- Live
- Pending
- Open
- Deleted: This is a pseudo state. Deleted jobs are not stored in the database.

A job may contain one or more topics and each topic has its own job state. The topic's job states are stored in TB_JOB_TOPIC_STATE.

You can use the document options to specify whether to perform a job topology (page 525) check when the job state is changed.

Change the job state in Topobase Client. See the Topobase Client User Guide, section Changing the Job State.

See also:

- Job Administrator (page 298)
- Setting Job Topology Document Options (page 27)
- System Table TB_JOB_STATE (page 84)
- System Table TB_JOB_STATE_TRANSITION (page 85)

Form Designer Reference 6

Form Designer Introduction

Autodesk Topobase Client is a basic module, used to process both geometry and attribute data that is stored in the database. With Topobase Form Designer you can customize the feature class forms.

Use the Form Designer to define multiple forms for a single feature class, and to assign different forms to different user groups.

NOTE Also, you can customize the forms of any tables, views, synonyms, or domain tables with the form designer.

To learn more about customizing Topobase feature class forms, you should start with the following topics:

See also:

- Starting Form Designer (page 309)
- Form Designer: Controls (page 318)
- Form Designer: Toolbar (page 320)

See also the Topobase Client User Guide, section Using Feature Class Forms.

Starting Form Designer

Topobase Form Designer is a tool that you use to design forms for processing feature data (feature class forms). For example, you can hide or show text boxes,

create controls, check boxes, radio buttons, and combo boxes in the forms. Arrange elements to create a form that suits your needs.

You must have Topobase Administrator login rights (member of the ADMIN user group (page 526)) to use the form designer.

To start the form designer

1. Start Topobase Administrator and open a workspace (page 526).
2. In the Administrator explorer, select the document (page 521).
3. In the Administrator window, click Document menu ➤ Form Designer. Alternatively, you can click the Form Designer item in the administrator explorer.
4. The form designer is displayed in the right pane. The topics and feature classes are displayed in the form explorer.
5. In the form explorer, select a feature class. In the right pane, the forms of the selected feature class is displayed.
6. Under Form Types, select a form, such as Default.
7. Click Edit. The Designer dialog box opens.

If you have selected a master detail form (page 523), you are prompted to select the form to edit.

See also:

- Form Designer: Creating Generic Forms (page 311)
- TB_RELATIONS: Master-Detail Relations (page 96)

Form Designer: Managing Multiple Forms

Use the Form Designer to define multiple forms for a single feature class, and to assign different forms to different user groups.

When you start the form designer, in the designer pane, you administer multiple feature class forms. For example, you define a form that only displays the attributes that are allowed to be edited by the VIEWER user group. Then you assign this form type to the user group VIEWER.

Designer pane	Description

Form Types	Displays the form definition that are available for the selected feature class. The default form provides a basic form definition containing controls for all feature class attributes.
Add	Adds a form definition. Opens the Designer dialog box for the selected feature class. See Form Designer: Creating Generic Forms (page 311).
Edit	Opens the Designer dialog box to edit the selected form. See Form Designer: Creating Generic Forms (page 311).
Duplicate	Duplicates the selected form.
Preview	Displays the selected feature class form.
Delete	Deletes the selected feature class form. You cannot delete the default form.
Assign a form type area	Assigns a form type to a user group.
User Groups	Displays the user groups.
Assigned Form Type	Assigns the form to the user group.

Form Designer: Creating Generic Forms

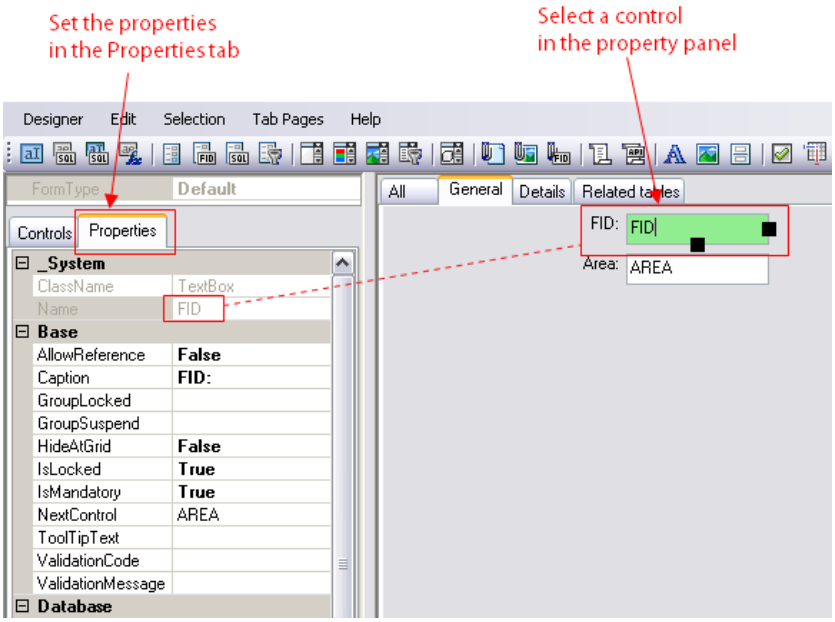
Use the Topobase form designer to customize feature class forms.

NOTE Make sure you know all about the components of Topobase feature class forms.

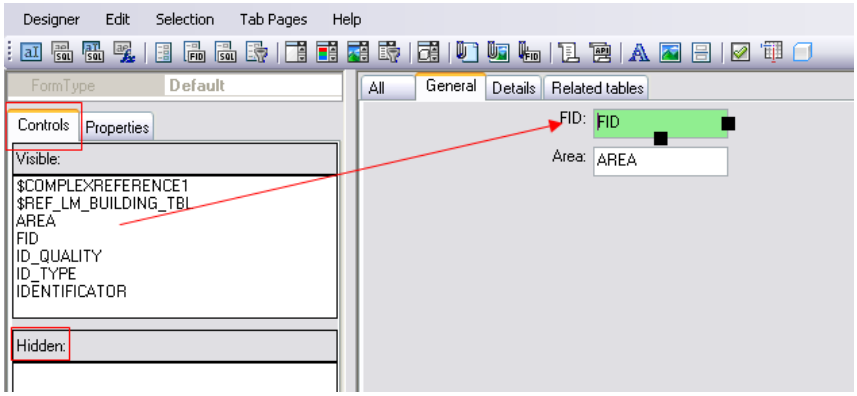
In the Form Designer, when you edit a feature class form, two windows are opened: the feature class form and the Designer dialog box. In the Designer dialog box, in the right pane, you can see all the form's controls (input fields, controls, buttons, and so on). In the left pane, you can process the particular elements.

Use the Controls tab to define the database attributes and controls that are shown in the form.

Use the Properties tab to define the properties of each control.



Topobase Form Designer: In the left pane, on the Properties tab, a list of the properties of the control that is selected in the right pane is displayed.



Topobase Form Designer: On the Controls tab, a list of visible and hidden controls is displayed.

- **Visible controls**—This list is an overview of all controls that are visible in the current form. To hide a control, drag it from the Form to the Hidden list box.

- **Hidden controls**—This list contains all controls that are currently hidden. To display a control in the form, double-click or drag the entry.

Hidden controls—This list contains all controls which currently are hidden. To show a control in the form, double-click (or drag-and-drop) the entry.

Also, you can use the Edit menu commands to hide or show controls.

NOTE To hide a control in the form, drag it to the Hidden list.

See also:

- Starting Form Designer (page 309)
- Form Designer: Menu Bar (page 313)

Form Designer: Menu Bar

Designer menu	Contains the following commands to control the appearance of the feature class forms:
Options	Controls the general appearance and assigns Help files or Help documents.
Master-Detail	Controls the appearance of master-detail forms.
GridOrder	Changes the order of the attributes or hides/shows certain controls in table view mode (in the grid). Select Automatic to order the controls by their position (Top, Left) as defined in the form designer. For forms containing several tabs, the order is Tab ID, Top, Left. Clear Automatic to define your own grid order. Select or clear the controls and use the arrows to move them.
Exit Without Saving	Cancels all changes that you have made.
Exit	Exits the form designer and saves all layout changes and configurations.

Edit menu	Contains the following commands to process the controls of the current form. Note that in some cases, you can use hotkeys or drag a control.
Copy Paste	You can copy and paste controls with all settings, such as for the position and caption. Controls that are related to an attribute are only pasted, if the attribute exists in the target form. If you paste a control that is related to an attribute, you are notified that some controls already exist and that you can replace them. If you do not want to replace them, only the new controls are pasted.
Hide All Show All	Hides hide or shows controls. By default, all controls are displayed. All hidden controls are listed in the left pane on the Controls tab (Hidden). All others are listed in Visible. **NOTE** Use the Hide command to remove a control that refers to a database attribute.
Hide Controls Without Data	Hides hide all controls that have no data (NULL) in the attribute of the table. You can remove the controls of unused attributes. Use Close Gaps to close the gaps in the layout.
Hide All Reference Controls	Hides all reference controls.
Create Gap Close Gaps	Removes or creates gaps between controls. If you hide some of the default controls, there can be gaps between the remaining controls.
Auto Tab Order	The tab order defines the order, in which the cursor will move when you press <TAB> or <ENTER>. Default setting is Auto Tab Order is ON = From left to right and from top to bottom. **NOTE** The tab order affects only the Form view mode. You can set the Grid Order in Table View mode.

Move Control	Changes the position of a control by moving it upwards or downwards. The command refers to a single highlighted control.
Delete Control	Deletes the selected control. If the control refers to a database attribute, for example FID, delete does not affect the database. The next time you start the form designer, the control is listed as Hidden. Controls with no relation to database attributes are completely deleted. **NOTE** Use this command to delete controls, such as for picture box or static label.
Selection menu	Contains commands to select and process the controls of the current form.
Select All	Selects and highlights the controls of the current form. Note: You can also select several controls at once (multi-selection), using the following: **By rectangle**—Drag a rectangle containing the controls you want to select. By click—Press Ctrl or Shift+click.
Hide Selected Delete Selected	Hides or deletes the selected controls. Notice the difference between hide and delete.
Move Selected To Tab Page	Moves a control from one tab to another.
Tab Pages menu	Contains commands to process tab pages.

See also:

- Designer Menu: Options (page 316)
- Designer Menu: Master-Detail (page 317)
- Designer Menu: Tab Pages (page 318)
- Form Designer Example: Auto Tab Order (page 357)

Designer Menu: Options

In the Designer dialog box, click Designer menu ➤ Options.

Option Dialog Box	Description
System Category	These settings cannot be edited.
ID	TB_GN_DIALOG.ID, can help to find the record in trouble-shooting.
Tablename	Oracle table name of the current feature class.
Name	Alias = caption for the oracle table (as defined in Topobase Administrator).
Title	Defines the title to be shown in the forms title bar. **NOTE** By default the title of a form corresponds to the title that has been set in Topobase Administrator.
Toolbar	Assigns a special toolbar to the form. Usually you will define tool bars for the different types of tables, such as feature class, domain and so on. For example, a domain table toolbar does not need any tools concerning graphic connection. Therefore you need to define only a single toolbar for a group of forms. Select a toolbar from the Toolbar box.
Startup SortOrder	Defines the order in which features are listed, such as according to point number or according to FID.
Help	Assigns a Help topic to the form. **NOTE** You can also assign Help topics to each control of the form. See also Form Designer: General Control Properties (page 324)
Filename	For desktop Help: Enter any help file (**.chm*, **.hlp* or any Windows document), as for example *Topobase.chm*. The files must be located in the directory <topobase>\Help or in a subdirectory of this. Always set the filename without the directory name. If you specify a (**.chm*) help file, you can specify a topic ID in the Context property.

Context	For desktop and Web help: Context ID or URL of the Help topic, as for example *topobase_introduction.htm*. Example: topic.htm#bookmarkname or topic.htm. **NOTE** If you leave the URL empty, this context is also used for the Web Help. In this case the Web Help location will be *<topobase_web>/help/context*.
Url	For the Web Help: URL of the Help topic, as for example *indexweb.html*. Default folder is <topobase_web>\help. Example: To use a file in a subfolder of the default directory, you have to specify the file starting with the subfolder name, as for example *customized/help.html*. **NOTE** Without the "*http://*" the files are searched on the Topobase Web server.

See also:

- Defining Form Toolbars (page 374)
- Assigning Help Files to Feature Class Forms (page 377)

Designer Menu: Master-Detail

In the Designer dialog box, click Designer menu ➤ Master-Detail.

Master-Detail Dialog Box	Description
Current Details	Lists the relations that are defined as master-detail.
Available	Lists all related tables: <table_name of related table>, <column of related table>, <column of table>. Click to select an item and then click an arrow button to move from one list to the other or to change the order.

Height of detail form: One default value for the height is stored in the system table TB_GN_DIALOG.DefaultDetailHeight. Do not edit this table manually. If someone who is not an ADMIN user (page 526) opens a master-detail form for the first time, this size is applied.

See also:

- TB_RELATIONS: Master-Detail Relations (page 96)

Designer Menu: Tab Pages

For feature class forms in form view mode, you can apply several tabs (Tab Pages) to display information.

In the Designer dialog box, click Tab Pages menu and click commands to create and edit tabs.

Click Tab Pages menu ➤ Group Suspend to specify that the tab page is invisible for one or more user groups. Tab pages can be suspended for certain user groups.

See also:

- Defining Tab Pages for Forms (page 374)

For more information about feature class form view modes, see the Topobase Client User Guide, section View Mode.

Form Designer: Controls

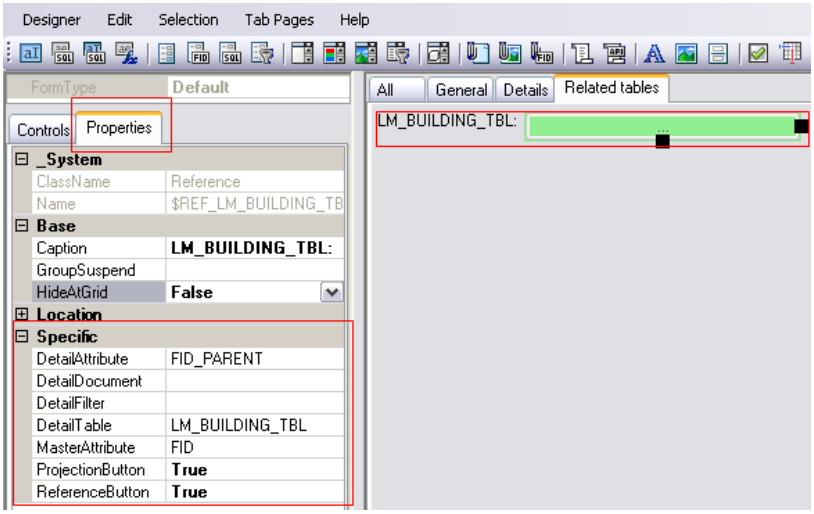
Topobase Form Designer provides different types of controls to customize the feature class forms.

The first type of controls show the content of a particular database attribute, for example FID, POINT_NUMBER, and so on. Each attribute of a table may have exactly one control of one of the following types:

- Property text box (TEXTBOX)
- Basic combo box (COMBOBOX)
- Check box (CHECKBOX)
- Color combo box (COLORBOX)
- Picture linker (PICTUREBOX)
- Picture combo box (PICTURECOMBOBOX).

These controls have a property category Database that shows the attribute name and dimension.

The second type of controls allow access to records related to the current record. These are created for each relation, but you can define additional controls. For example, by default, a reference control is defined for label feature classes.



Form Designer: Reference type controls look like a button and allow accessing related data. Click the control in the right pane and set its properties.

The third type of controls add layout elements, such as text, pictures, buttons, and lines. They have no counterpart in the database. For example, they are used to write commentary, illustrate the form, or add a company logo.

Form Designer: Default Forms

When you start the Topobase form designer for the first time, a default form is created. It normally contains controls of the following type:

- **Property text boxes**—For database attributes without any relation
- **Basic combo boxes**—For database attributes with relation to another table
- **Check boxes**—For database attributes declared as number (1)
- **Basic form launcher**—For relations pointing to the current table, for example for a relation to a label feature class (page 523)

The maximum InputLength of an attribute corresponds to the data structure and input is limited to this size.

The sort order of the attributes is defined by the COLUMN_ID of USER_TAB_COLUMNS. This order should correspond to the order of creation in Oracle.

Form Designer: Toolbar

The toolbar of the Designer dialog box provides distinct commands for creating and editing forms. The toolbar icons are grouped according to their function.

Click an icon to add a control to the form, as shown in the following table:

Icon	Description
	NOTE The controls TEXTBOX, CHECKBOX, COMBOBOX, COLORBOX, PICTUREBOX and PICTURECOMBOBOX refer to a database attribute. For each attribute, you can apply one appropriate control. If all database attributes have one of these controls, no further control of this type can be created. The message "All Attributes Have A Control! Would You Like To Replace An Existing Control ?" indicates that there are no attributes without a control in the current form. Click Yes to replace the control. In the Select Control To Be Replaced list you can select the attribute.
	Controls, used to show a text box, referencing a value
	Property TextBox Class name: TEXTBOX Adds a text box to show the content of the database attribute. Use a text box to show the content of a database attribute. The Message: All Attributes Have A Control ! means: In the current form, there are no attributes remaining without control.
	Read-Only SQL TextBox Class name: SQLLABEL Adds an SQL Label. Use this to show the result of an SQL query. You can define a select statement to query any information. The result is shown in the form, but cannot be edited. Example: Use SQL label to calculate the total area of all related parcels.

Editable SQL TextBox
Class name: SQLTEXTBOX
Adds an SQLTextBox: Use this to show the result of an SQL query, for example, to show the area which is stored in a different table. The result can be edited and the value is saved (update). Use this type of control only if you query a single attribute.

Class name: FORMULA
Adds a formula to show the result of a calculation using the database attributes. Use this to calculate values. You can use all database attributes and all mathematical expressions (VB .NET code).

Controls that open an editable form, showing all properties of a referenced feature or set of features.

Basic Form Launcher
Class name: REFERENCE
Adds a Reference. Use this to show data related to the current record. Reference controls are created for all relations that point to the current table, for example, the relation to the label feature class.

NOTE This control is set by default, according to TB_RELATIONS

FID Form Launcher
Class name: FIDREFERENCE
Adds a FID-Reference to open a form that belongs to a certain FID (dynamic relation, multi-relation). An FID Reference is not linked to a database attribute of the current table. Use FID References to show data with a special FID.

SQL Form Launcher
Class name: COMLEXREFERENCE
Adds a ComplexReference to create references using complex SQL statements, for example to open a form of another table. A Complex Reference is not linked to a database attribute of the current table. Use this to open the form of another table with a filter. You can define this filter with complex SQL statements.

Advanced Form Launcher
Class name: MULTIREFERENCE
Similar to the Reference control, you can use multi-relations.

Combo boxes

Basic Combo Box.
Class name: COMBOBOX
Adds a combo box that enables acquisition through a dropdown list. A combo box is always linked to a database attribute. Use this control to select your entry through a combo box from a domain table (*_TBD) or any other related table.

Color Combo Box.
Class name: COLORBOX
Adds a color combo box that can be used to select a color from a list.

Advanced Combo Box.
Class name: MASTERBOX
Adds an advanced combo box (master box). The master box control is similar to an add-on to the combo box control. Use this control to define one or more filters (pre-selections) for a combo box. For example, to enter a street name, you define a filter for the city. In the form, the master box looks like a combo box.

Picture ComboBox
Class name: PICTURECOMBOBOX
Provides picture file names in a list box.

Model Selector
Class name: MODELBOX
Adds a control for acquisition of model driven attributes. See also Data Model: Model Tables (page 184).

Linkers

Document Linker
Class name: DOCUMENTMANAGER
Opens the document manager and shows whether any documents are available.

Picture Linker
Class name: PICTUREBOX
Displays record-related pictures.

FID Linker
Class name: FIDBUTTON
Adds an FID button that can be used to select a feature in the map and fill an attribute with this value (data acquisition).

	Programmable controls
	Script-Programmable Control Class name: SCRIPTBUTTON Inserts a control that is programmed using VB script.
	API-Programmable Control Class name: APICONTROL Inserts a control that is programmed as a DLL.
	Static content
	Static Text Class name: LABEL Adds a label. Use this to write any text, notes, or remarks in the form.
	Static Picture Box Class name: STATICPICTURE Adds a StaticPicture: Use this to position a picture in the form, for example, a company logo.
	Horizontal Line Class name: LINE Draws a line. Use this to paint horizontal or vertical lines into the form.
	Other controls
	Coordinates Editor Class name: GEOMETRY Adds a Geometry control to show coordinates and to enable edition of coordinates. Use this to show coordinates (Y, X, Z) in the form.
	Checkbox Class name: CHECKBOX Adds a Yes/No CheckBox. A check box is always linked to a database attribute. Use this control to show the content of a database attribute which can only be 1 or 0, respectively Yes/No, or Open/Closed. A check box control is assigned to database attributes which are declared as num (1).
	Feature Highlighter Class name: HIGHLIGHTBUTTON Highlights a feature in the map.

See also:

- Creating Controls (page 370)
- Form Designer Example: Advanced Combobox (page 365)

Form Designer: General Control Properties

The following general control properties belong to all types of controls:

Property	Description
Form Type	Specifies the name of the form. For each feature class, Topobase provides a default form. You can define multiple forms per feature class. See also Form Designer: Managing Multiple Forms (page 310).
System Category	System category properties cannot be changed.
Name	All controls related to a database attribute are named like the attribute. For example, the text box FID. All other names start with the suffix "$", such as $FIDREFERENCE or $FORMULA1 or $FORMULA2.
ClassName	For example: TextBox
Base Category	
Caption	This caption is displayed in front of the input box.
GroupLocked	Specifies that the controls are visible but locked for one or more user groups. Select the group whose members are not allowed to use this control.
GroupSuspend	Specifies that the controls are invisible for one or more user groups. Controls can be suspended for certain user groups. Select the group whose members are not allowed to use this control.
HideAtGrid	Hides a control in Table View mode. True = hidden, false = shown.

NextControl	Defines the tab order. Each time a user presses <TAB> or <ENTER>, the cursor moves to the control (input field) that you can define here. This way, you can control the order of inputs.
ToolTipText	Help text that is displayed when the cursor hovers over the input field.
Database Category	
ColumnDesc	Description of the attribute
ColumnName	Attribute name
Help Category	Assigns a help topic to a control. The topic is displayed if you press F1 while the cursor is on the control. The properties are the same as for the form Help. See also Designer Menu: Options (page 316)
Help Context	Assigns a help topic, such as *topobase_introduction.htm*.
HelpFilename	For the desktop version: Specifies the help file. For example, *Topobase.chm*. **NOTE** If you define no HelpFilename for a control, Topobase Client will use the HelpFilename of the form.
HelpURL	For the Web version: URL of the Help topic. For example, *index-web.html*.
Specific Category	
ClientSideUniqueCheck	True = if the user enters a value, the system checks whether the value is unique for the feature class. Note, that this property does not apply to the Oracle database property. Example: Set this value to True to import data from excel, that may contain identical values. In this case, you would have set the Oracle property to Non Unique to allow the import from excel. However, you want to prevent the user from entering new values that already exist. The default is False.

More properties are shown in the following sections.

See also:

- Form Designer Example: Auto Tab Order (page 357)
- Designer Menu: Options (page 316)

Form Designer: Advanced Combobox - Properties

An Advanced Combobox (master box) control is an add-on to a combo box. You can relate an advanced combo box both to a basic combo box or to another advanced combo box in the current form. Therefore, you can catenate several dropdown lists and trace multiple relations, as shown in the following table:

Property	Description
System Category	
ClassName	MasterBox
Name	Name, for example $MASTERBOX7
Base	
IsLocked	Default = false
Specific Category	
DetailControl	Name of the ComboBox control, for example FID_STREET
DetailAttributeName	Detail attribute with relation to master table (child_column_name), for example FID_CITY
DetailFilter	
FidButton	With the FidButton, you can select a feature in the map. Then, the record of the master table is selected with the FID of the selected feature and the dependent value of the record is selected in the list.

HighlightButton	True = show a highlight button
MasterAttribute-Name	Master attribute of the master table (parent_column_name), for example FID
MasterFilter	Filter: you can define a filter for the master box dropdown list.
MasterList	Display attributes: Shown in dropdown list of the master box, such as NAME
MasterOrderBy	Orderby. You can sort the items in the dropdown list.
MasterTable	Table from which you choose the pre-selection for the detail control (parent_table_name), for example LM_CITY.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer Example: Advanced Combobox (page 365)

Form Designer: Advanced Form Launcher - Properties

You use Advanced Form Launcher (Multi Reference) controls in the same way as Basic Form Launcher (Reference) controls. The difference is that can use multi–relations. That is, with one button, you can open more than one form at the same time.

The Advanced Form Launcher generates two buttons: the Reference button and the Projection button. If the user presses the Reference Button, the form of all the related tables is opened, with a filter set to the current record.

NOTE There is a difference between Reference and Projection: The Reference Button shows related features of the current record. The Projection Button shows the related features of all records that are in the current filter.

Property	Description
System Category	
ClassName	MultiReference

Name	Name, default = $MULTIREFERENCE9
Specific Category	
DetailRelations	Select the relations/forms
ProjectionButton	Default = true
ReferenceButton	Default = true

Example with Feature classes Parcel, SmallHouse and LargeHouse with relations in TB_DICTIONARY

- Parcel.FID -> SmallHouse.FID_Parcel
- Parcel.FID -> LargeHouse.FID_Parcel

If you click the FID button, all related forms that have at least one record that matches the relation, are opened.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer: Basic Form Launcher - Properties (page 332)

Form Designer: Basic ComboBox - Properties

This control is frequently used for domains (*_TBD). When a form is created, every attribute with a relation to a domain receives a basic combo box. All entries in TB_RELATIONS are analyzed.

Basic combo box properties: A combo box always has a drop-down arrow that you use to select an entry. If the related feature has geometry, you can optionally set an FID Selection button to identify and assign the related object in the map and a Highlight button to highlight the related object in the map.

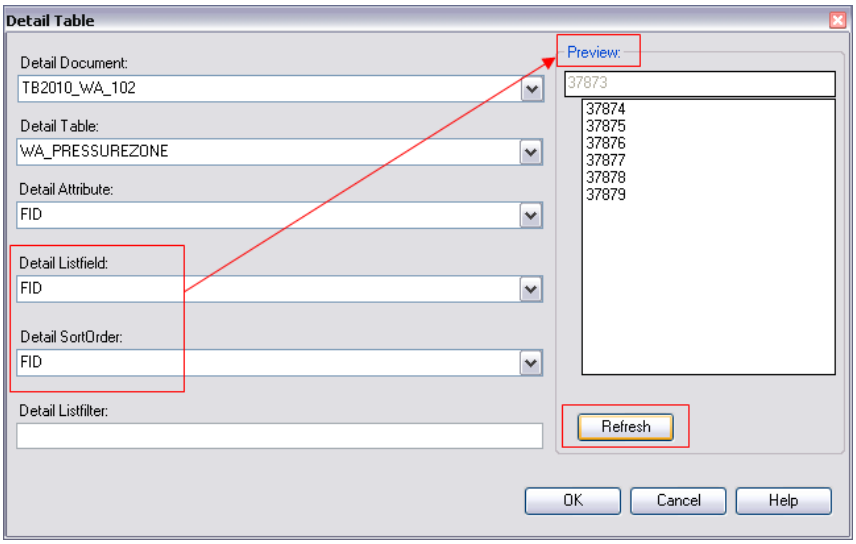
Property	Description
System Category	
ClassName	ComboBox

Name	Name = attribute name, for example ID_STATUS
Base	
AllowReference	If True, the value can be entered by using a reference record.
IsLocked	True = cannot be edited
IsMandatory	True = must always be acquired. In edit mode these attributes are highlighted by a red background color.
ValidationCode	Defines a validation code for each input field. During validation, values from other attributes can also be queried.
ValidationMessage	Defines a message to display if validation fails.
Specific Category	
AllowDoubleClick	True = default = If you double-click, the form of the related table will appear. False = If you double-click, no table will appear.
Detail	All Detail entries belong together. They can be entered together in a separate dialog box that appears when you click any of the respective drop-down arrows.
DetailDocument	The detail table may also reside in a document that is not the current document.
DetailFieldName	Detail attribute that is stored, for example, ID; usually the attribute is set according to TB_RELATIONS.
DetailFilter	Default filter = empty. Detail Listfilter: You can define which records of the detail table are available, for example, ACTIVE=1. Also, you can define an additional filter.
DetailList	Detail Listfiled: attribute to show in the form, for example, VALUE. For example, use *ID* || ' - ' || *VALUE* to show both ID and VALUE.

	For domains usually the ID is listed. It is reasonable to change this into VALUE, as this is more clarifying.
DetailOrderBy	Defines the sort order of entries in the detail table, for example, VALUE.
DetailTable*	The related table. For example *_TBD, which is usually the detail table that is set according to TB_RELATIONS.
FidButton**	True = an FID selection button is shown next to the input box. Use this only if the relation points to a geometry feature that can be identified in the map.
HighlightButton	True = shows a highlight button
UseBuffer***	True = default; by default, all values of the specified detail table are loaded once in a session. If set to False, loads the detail table every time you open the combo box (by pressing the drop-down arrow button) or you change (update, or navigate to) the record. Note: A False setting negatively impacts performance. Always set this option to True if you work in Table View mode, because in this case records are changed frequently in order to show all records.
ValueDefault	Specifies a default value to insert when you add a new record. You can also use this for the AutoIncrement value or date/time and to create GUIDs.

* DetailTable: The preview pane shows the detail list. Click refresh to check whether your definition has the desired results.

** FID button: The Fid Selection button is used to assign the related object by clicking in the drawing. It is visible even if the attribute is locked. That means that the drop down list will not open, and the user will always need to use the button to select the feature in the map. He cannot select one from the list. In this special case, locked means that the user cannot enter any values using the keyboard.



Combo box detail table settings: Normally you would change the Detail Listattribute or Detail Sort Order only, because the Detail Table and Detail Attribute values are read from TB_RELATIONS.

*** UseBuffer: Normally, you set the UseBuffer property to True for best performance. In the case of real domain tables (*_TBD), the entries do not change much, so they are not loaded every time you change the record. It may be necessary to set the UseBuffer property to False, for example, if the related table is not a real domain, but an ordinary feature table, whose contents frequently change.

Multi-relations, which are relations to several tables, can be set in the data model administrator. If a multi-relation is defined, you can select those tables from the Detail Table dropdown list as well. In this case, you will see an entry in the dropdown list in {} as the first item.

If you have a complex expression instead of a simple attribute, you can use multi-relations, which means that you can have a list of FIDs from several defined feature classes. For example: 'FID: ' || FID || ', NR: ' || NR

To get this working, the expression in the property box has to look like this: 'FID: ' || FID || ', NR: ' || NR, NR. Note, that you should list all attribute names used in the expression after the expression and separated by commas. The attribute name FID is an exception and should not to be listed.

See also:

- Form Designer: General Control Properties (page 324)

- Form Designer Example: Validation Code (page 358)
- Form Designer: FID Form Launcher - Properties (page 342)

For more information about Refresh Records see the Topobase Client User Guide, section Using Feature Class Forms.

Form Designer: Basic Form Launcher - Properties

The Basic Form Launcher (Reference) control generates two buttons: the Reference button and the Projection button. If the user presses the Reference button, the form of the related table is opened, with a filter set to the current record.

NOTE There is a difference between Reference and Projection: The Reference Button shows related features of the current record. The Projection Button shows the related features of all records that are in the current filter.

The first time you open a form, Topobase Client creates a Basic Form Launcher according to the entries in TB_RELATIONS. Most of the Basic Form Launchers are of this type. However, in very special cases you can define your own references.

Property	Description
System Category	
ClassName	Reference
Name	Name, default = $REFERENCE1
Specific Category	
Detail	The Detail properties can be edited in the Reference Options dialog box. Click the property row and then click .
DetailDocument	Shows details or forms that belong to another document (page 521).

DetailAttribute-Name	Child_column_name. For example, FID_PARENT.
DetailFilter	Default filter = empty = all related records. You can define an additional filter to this filter.
DetailTable	Child_table_name. For example, POINT_TEXT.
MasterAttrib-uteName	Parent_column_name. For example, FID. Note: The current form is the parent_table.
ProjectionBut-ton	True = a projection button is generated. This is the default.
ReferenceBut-ton	True = a reference button is generated. This is the default.

For DetailFilter, you can define which of the related records are to be shown. For example, in the feature class form Water Pipe, you can create a Reference button that shows all related points of a certain type (TYPE):

- Define DetailFilter as: TYPE = 'Fireplug'

You can use any SQL expression (that is the part following the WHERE), for example:

TYPE = 'Fireplug' or TYPE = 'any' OR

TYPE = 'Fireplug' and TYPE in (select ty from tytable where v1='any'

For DetailDocument, you can open a form that belongs to a table of another document (page 521). This document has to be open so that you can select it from the dropdown list. For example, a workspace (page 526) with two documents is open, and you want to show Real Estate data in a Water form. In the Water form, you need to select Real Estate for DetailDocument.

NOTE The detail document might not exist, or not be opened in the user's current workspace. This can happen if the form designer selects his document name (template), but the user has checked in the document with another name (for example, the template with the name real_estate was checked in as document my_real_estate). In this case, Topobase Client will search all open documents for this form.

The form designer uses the DetailDocument property to select the form of the document in a list box. This property is also helpful if the same table name occurs in several document (page 521)s (for example, TB_* tables), which is an extremely rare case. Normally, all Topobase applications use a prefix to distinguish table names (W = water, RE = real estate, and so on) so the problem of opening the wrong table when the detail document name is not correct, will not occur.

If the DetailDocument property is not specified, the system searches in the current database, and if the DetailDocument is specified (even with a nonsense name), searches in all open databases.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer: Advanced Form Launcher - Properties (page 327)

Form Designer: CheckBox - Properties

Property	Description
System Category	
ClassName	CheckBox
Name	Specifies the name of appropriate database attribute.
Base	
AllowReference	If True, the value can be entered by using a reference record.
IsLocked	If True, the value cannot be edited.
IsMandatory	If True, attributes must always be acquired. In edit mode, these attributes are highlighted by a red background color.
ValidationCode	Defines a validation code for each input field. During validation, values from other attributes can also be queried.

ValidationMessage	Defines a message to display if the validation fails.
Specific Category	
FalseValue	Defines a value to be displayed in the form if the database value is 0 or NULL. For example, No, Close, or False.
TrueValue	Defines a value to be shown in the form if the database value is 1. For example Open, True, or Yes.
ValueDefault	Specifies a default value that is inserted when you add a new record. You can also use this for the AutoIncrement value or date/time, and to create GUIDs.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer Example: Validation Code (page 358)

Form Designer: Color ComboBox - Properties

With the Color ComboBox (ColorBox) control, you can select a color from a list. The control displays each color in a small box in front of each color name. This box is not available in the Web version. The Web version displays only the name.

By default, all database attributes have a control. To set a color combo box, you are prompted to replace an existing control. Select the existing control in the Select Control To Be Replaced dialog box.

Property	Description
System Category	
ClassName	ColorBox
Name	Specifies the attribute name.
Base	

AllowReference	If True, the value can be entered by using a reference record.
IsLocked	If True, the value cannot be edited.
IsMandatory	If True, attributes always be acquired. In edit mode, these attributes are highlighted by a red background color.
ValidationCode	Defines a validation code for each input field. During validation, values from other attributes can also be queried.
ValidationMessage	Defines a message to display if the validation fails.
Specific Category	
ColorSchema	Specifies the table in which the colors are defined. Default is TB_COLOR_ACAD. You can create your own color schema table in order to provide different color definitions (in the TBSYS user, arbitrary name). The table must have the following attributes: - ID (varchar2): value of the color; - VALUE (varchar2): name displayed for the color - RGB_VALUE (varchar2(6)): RGB color, to display the color in the control and the list. For Web applications, you can use TB_COLOR_WEB, which stores W3C color definitions.
RGBButton	If set to True, generates an additional button that displays a free RGB color selection dialog box, to select a RGB color.
ValueDefault	Specifies a default value to insert when you add a new record. You can also use this for the AutoIncrement value or date/time and to create GUIDs.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer Example: Validation Code (page 358)

Form Designer: Coordinates Editor - Properties

Use the Coordinates Editor (Geometry) control to display geometry and coordinates of points, lines and polygons, as shown in the following table:

Property	Description
System Category	
ClassName	Geometry
Name	Specifies the name. For example, $GEOMETRY4.
Base	
IsLocked	If True, the value cannot be edited.
Specific Category	
AvailableValue	Instead of the coordinates, you can show a text. If no geometry exists, the UnAvailableValue text is displayed. If geometry exists and the AvailableValue is not empty this text is displayed. If the AvailableValue is empty (default), the coordinates are shown.
UnAvailableValue	
PrecisionXY	Defines the precision of the values shown in the form.
PrecisionZ	
ShowEndPointOnly	
ShowInterpretation	Shows the start or end or mid point of an arc.
ShowStartPointOnly	If False, in the case of lines/polygons, all coordinates of all vertices are shown. If True, only the coordinates of the first or last vertex are shown.
ShowX	X = easting, Y = northing, Z = elevation. If True, the respective value X, Y, or Z is shown in the text box/list box. The order of display is X (easting), Y (northing), Z (elevation).
ShowY	

ShowZ	If False, the respective value (x,y,z) will not be shown in the text box/list box.
SQL	Default: empty. You can define an SQL statement to show the geometry of related child features, which is stored in a different table. For example, this happens, if the table is of the attribute type, as is the case in the Topobase utility model (page 526). Note the parallel to label definitions. With a geometry control in addition to SQL, you can display the coordinates of the related geometry. With a Parent Geometry select statement, you can query the geometry of a related feature if the parent label is of the attribute type.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer Example: Coordinates Editor with SQL (page 369)
- Showing the Coordinates of Utility Features (page 372)
- Using the SQL Assistant (page 105)
- Label Geometry Select Statements (page 202)

Form Designer: Document Linker - Properties

The Document Linker (DocumentManager) control is not linked to a database attribute of the current table. This control inserts a button with the number of assigned documents on it (documents that have been assigned using the document manager).

When the user clicks the button, the document manager is opened, with the assigned documents filtered.

NOTE Using this control will make the performance of the form slower.

Property	Description
System Category	
ClassName	DocumentManager

Name	Specifies the name. For example, $DOCUMENTMAN-AGER1.
Specific Category	
AvailableColor	The color, if documents are available.
AvailableValue	Message that is displayed on the button. For example, *{0} Document(s) Available.*
UnAvailableColor	The color, if no documents were found.
UnAvailableValue	The returned message, when no documents are found. For example *No documents available.*

See also:

- Form Designer: General Control Properties (page 324)

Form Designer: Editable SQL TextBox - Properties

Use the Editable SQL TextBox (SqlTextBox) control to display the result of an SQL query, for example, showing the area which is stored in a different table. It is recommended that you use this type of control only when you query a single attribute.

If you do not want to edit the value and just show it, it is recommended that you use the Read-Only SQL TextBox control, which has a better performance.

Property	Description
System Category	
ClassName	SqlTextBox
Name	Specifies the name. For example, $SQLTEXTBOX1.
Base	
IsLocked	If True, the value cannot be edited.
Specific Category	

AlignRight	If True, aligns the value. Specifies whether the value is aligned to the right edge of the input box.
AllowMultiRowUpdate	If False, checks whether the update tries to update more than one row in the linked table. If so, the whole update is ignored, and a message will appear. Otherwise with a wrong SQL statement, you could destroy your data when your SQL would create unwanted updates.
DetailAttributeName	Specifies the attribute (to be selected from the detail table) that you want to display.
DetailFilter	The filter expression for the select statement (without a WHERE expression). Click to open the SQL Assistant. Use the SQL Assistant to build the filter statement. However, you must cut off the first part, including the WHERE.
DetailTable	Specifies the table from which you want to select data.

Note the difference between Read-Only SQL TextBox and Editable SQL TextBox. Both controls allow you to query data, but the Read-Only SQL TextBox results in a read-only label, that cannot be edited. The Editable SQL TextBox results in a text box, where the value can be edited and saved (updated) in the database.

For this reason, the SQL statements differ in the way they are defined. While the Read-Only SQL label statement can be defined as one expression in the SQL property, the Editable SQL text box statement consists of three components (DetailAttributeName, DetailFilter and DetailTable properties). However, both statements have the same result.

Because of the splitting, the system is able to update the record. When the user saves the record, the system updates the related detail table, using the following command:

Update `<detailtable name>` set `<detailfield name> ='<your input>'` where `<detail filter>`.

See also:

- Form Designer: General Control Properties (page 324)
- Using the SQL Assistant (page 105)

- Form Designer Example: Editable SQL TextBox (page 364)
- Form Designer: Static Text - Properties (page 356)

Form Designer: Feature Highlighter - Properties

Use a Feature Highlighter (HighlightButton) control to highlight a feature in the map that is not in the current table, but is linked to the database attribute with a relation. This control works in a similar way to the FID Button control, but it will highlight a feature. It is a special variation of the combo box highlight button.

Example from a Wastewater application: Use the Feature Highlighter in the section feature class, which has a relation to the manhole at the section start point, to highlight the manhole.

You can also use a filter to highlight a feature that is different from the one where you get the FID.

Property	Description
System Category	
ClassName	HighlightButton
Name	Specifies the name. For example, $HIGHLIGHTBUTTON1.
Specific Category	
DetailAttributeName	Specifies the name of the database attribute that contains the FID. For example, FID_DETAIL.
DetailFilter	SQL filter that highlights another feature. You can use the SQL Assistant to define the SQL statement.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer: FID Linker - Properties (page 343)
- Form Designer: Basic ComboBox - Properties (page 328)
- Using the SQL Assistant (page 105)

Form Designer: FID Form Launcher - Properties

You use the FID Form Launcher (FidReference) control to generate an FID button. When a user clicks this button, the form the FID belongs to is opened, with the filter set to the FID.

NOTE The FID is unique in the whole database. Therefore, with the FID Form Launcher you can create dynamic relations. Use table TB_UFID to see which FID belongs to which feature class.

Property	Description
System Category	
ClassName	FidReference
Name	Specifies the name. For example, $FIDREFERENCE.
Specific Category	
MasterAttribute	Selects the attribute name (of the current table) where the FID is saved.

If the current record contains attributes that have FIDs saved, for example, those that have a relation to another table, you may wish to show this related record. If this relation is a multi-relation, you can use the FID Reference to create a reference button that will open the form the FID belongs to. The filter is set to the FID. Because the FID is unique in the whole database, you can handle dynamic relations.

Example: In relation to a point, where the relation is dynamic (to several point feature classes), you can either use the basic combo box or FID form launcher to show the related record. Also, you can define the field as a simple text box and add an FID form launcher. In addition, you can hide the text box and use only the FID form launcher.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer: Basic ComboBox - Properties (page 328)

Form Designer: FID Linker - Properties

An FID Linker (FidButton) control is not linked to a database attribute belonging to the current table. With this control, you can select a feature in the map and fill an attribute with this value (This is the same as acquiring related features simply by selecting them in the map). However, normally, you would select a related feature from a basic combo box, for example, you could select a detail that belongs to a manhole.

Property	Description
System Category	
ClassName	FidButton
Name	Specifies the name. For example, $FIDBUTTON1.
Specific Category	
DetailAttribute	Specifies the name of the database attribute that should be filled with the FID. For example, FID_DETAIL.
DetailFilter	Filter statement (optional): The SQL statement that gets another FID as a result. Example: You want to select a feature in the map but fill the FID of its parent into the attribute. Use the SQL Assistant to define the filter statement. *Select FID from parent where FID_CHILD={Fid}* Note: {FID} is case sensitive. The {FID} is replaced by the selected FID at runtime.
DetailTable	Specifies the feature classes that should be selectable in the map. Use this, when you need more than one feature class, such as *{FeatureC1,FeatureC2,FeatureC3}*

See also:

- Using the SQL Assistant (page 105)
- Form Designer: General Control Properties (page 324)
- Form Designer: Basic ComboBox - Properties (page 328)

Form Designer: Formula TextBox - Properties

Property	Description
System Category	
ClassName	Formula
Name	Specifies the name: For example, $FORMULA3.
Specific Category	
Formula	Clicking defines a formula and opens the formula editor. You can calculate formula, using the database attributes and all valid VB .NET codes.

NOTE The formula expressions obey the same rules as validation statements.

Functions (excerpt)	
	In the formula, you can use the expression “result=me.” to see a list box with all the available functions that you can use in the formula. In the formula pane where a list box appears you can select the desired function by double-clicking.
Me.Connection	Returns the Current Connection (Type Topobase.Data.TBConnection).
Me.ConnectionTools	Returns the Current ConectionTools (Type Topobase.Data.Tools.ConnectionTools).
Me.GeometryAvailable	Returns whether Geometry is available or not.
Me.DocumentCount	Returns the number of Microsoft Documents (DoumentManager) that are related to this feature.
Me.BackColor= "" Me.ForeColor=""	Sets the color (HTML color).

NOTE In the formula editor, you can select the available functions in a list box that appears after you enter "me.". The result of the formula is always assigned to the variable “result".

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer Example: Formula TextBox (page 359)
- Formula TextBox: Sample Formulas (page 360)

Form Designer: Horizontal Line - Properties

Use the Horizontal Line (Line) control to draw horizontal and vertical lines with various properties, as shown in the following table:

Property	Description
System Category	
ClassName	Line
Name	Specifies the name. For example, $LINE6.
Specific Category	
ForeColor	Specifies the color of the line.
LineSize	Specifies the size of the line.
ThreeD	If True, the line is three dimensional.
Vertical	If False, the line is horizontal.

See also:

- Form Designer: General Control Properties (page 324)

Form Designer: Picture Combobox - Properties

This control is linked to a database attribute. You use the Picture ComboBox control to display picture files, which are stored in one directory, in a combo box.

Property	Description
System Category	
ClassName	PictureComboBox
Name	Specifies the attribute name.
Base	
AllowReference	If True, the value can be entered by using a reference record.
IsLocked	If True, the value cannot be edited.
IsMandatory	If True, the value must always be acquired. In edit mode, these attributes are highlighted by a red background color.
ValidationCode	Defines a validation code for each input field. During validation, values from other attributes can also be queried.
ValidationMessage	Defines a message to display if validation fails.
Specific Category	
ItemHeight	Specifies the size (height) of picture in the list.
ItemWidth	Specifies the size (width) of picture in the list.
PictureDirectory	Specifies the directory containing the picture files you want to provide in the list box.
ValueDefault	Specifies a default value to insert when you add a new record. You can also use this for the AutoIncrement value or date/time, and to create GUIDs.

NOTE Although the picture files can be stored in any folder, it is recommended that you store them in the <topobase>\Pics folder or a subfolder. Then, they are available with the dynamic path <topobase>\Pics and you need to enter the subfolder/filename only. If you use picture files with a static path, they are not available on a different computer unless the path is exactly the same. Alternatively, you can use *//Servername* to store the picture files on a server.

NOTE To show the pictures in the Web version, if you do not use the <topobase>\Pics folder, you must map the folder names in the system table TB_GN_DIRECTORY_ALIAS.

See also:

- System Table TB_GN_DIRECTORY_ALIAS (page 75)
- Form Designer: General Control Properties (page 324)
- Form Designer Example: Validation Code (page 358)

Form Designer: Picture Linker - Properties

The Picture Linker (Picture Box) control is linked to a database attribute. It shows record-related pictures (**.gif*, **.bmp*, and so on). In the form, the user can use a list box to select a picture. When the picture is selected, the directory or path names of the pictures are saved in the database.

NOTE Although the picture files can be stored in any folder, it is recommended that you store them in the <topobase>\Pics folder or a subfolder. Then, they are available with the dynamic path <topobase>\Pics and you need to enter the subfolder/filename only. If you use picture files with a static path, they are not available on a different computer unless the path is exactly the same. Alternatively, you can use *//Servername* to store the picture files on a server. The dimension of the respective database attribute must be large enough to store the directory names.

TIP Make sure that the size of the control is large enough to display the picture.

NOTE To show the pictures in the Web version, if you do not use the <topobase>\Pics folder, you must map the folder names in the system table TB_GN_DIRECTORY_ALIAS.

Property	Description

System Category	
ClassName	PictureBox
Name	Specifies the name. For example, PICTURE.
Base	
AllowReference	If True, the value can be entered by using a reference record.
IsLocked	If True, the value cannot be edited.
IsMandatory	If True, the value must always be acquired. In edit mode, these attributes are highlighted by a red background color.
ValidationCode	Defines a validation code for each input field. During validation, values from other attributes can also be queried. See also Form Designer Example: Validation Code (page 358).
ValidationMessage	Defines a message to display if validation fails.
Specific Category	
HideText	Specifies whether the list box text is hidden. Set to True to hide the list box with the file name and show the picture only. This property only works in View mode. Default = False.
StretchImage	Specifies how the image is displayed. If True, the picture is stretched to the size of the control. If False, the picture is shown in its original size.

See also:

- System Table TB_GN_DIRECTORY_ALIAS (page 75)
- Form Designer: General Control Properties (page 324)

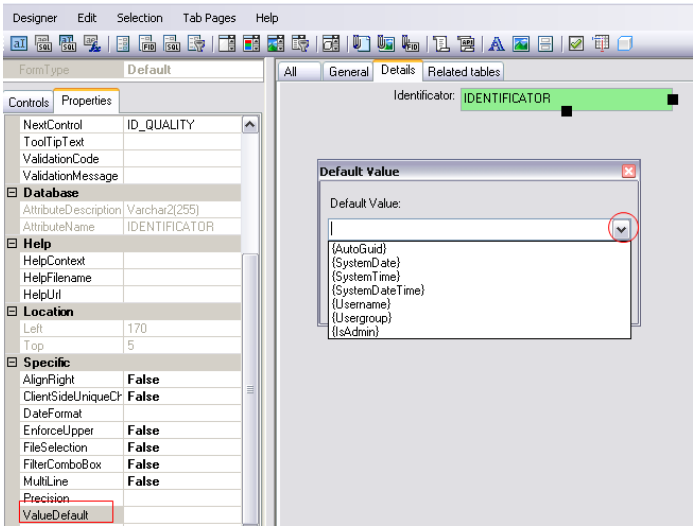
Form Designer: Property TextBox - Properties

Property	Description
System Category	
ClassName	TextBox
Name	Specifies the database attribute name. For example, FID, COMMENTARY.
Base	
AllowReference	If True, the value can be entered via reference record.
IsLocked	If True, the value cannot be edited. In the form, locked input fields are highlighted with light gray color.
IsMandatory	If True, the value must always be acquired. In edit mode, these attributes are highlighted by a red background color.
ValidationCode	Defines a validation code for each input field. During validation, values from other fields can also be queried.
ValidationMessage	Defines a message to display if validation fails.
Specific Category	
AlignRight	If True, aligns the value. Specifies whether the value is aligned to the right edge of the input box.
DateFormat	Specifies the date format to be shown in the form. Only available for Oracle attributes of Date type. For text boxes where DateFormat is set, a Calendar Button is displayed. However, if DateFormat is set to "yyyy" the calendar button is invisible.
EnforceUpper	Specifies whether all characters that are entered in the text box are converted to upper-case characters.
FileSelection	Specifies that a reference is made to an external file. Default = False. Set this property to True to refer to an extern-

	al file, for example a picture. Two additional buttons are displayed. One selects the file, the other shows the file with its related application (for example a **.doc* with Microsoft® Word).
FilterComboBox	Specifies that a filter list box is displayed. Default = False. Set this property to True to define that in filter mode a list box will be displayed, where the user can select the items. This means that in filter mode, the user can find a record by its name or another value, in a list.
MultiLine	Specifies whether to enable multi line text. If True, creates text of more than one line. Press <RETURN> to insert a line break. If this property is set to False, your cursor will move to the next input field, when you press <RETURN>.
Precision	Specifies the precision of the value, which is shown in the form. As a rule, this will differ from the precision saved in the database. By default, set to the precision of the DB definition.
ValueDefault	Specifies a default value that is inserted when you add a new record. You can also use this for the AutoIncrement value or the date/time and to create GUIDs.
WordWrap	Specifies that automatic word wrap is enabled. **NOTE** If you want to use this option in the Web version, you must also enable the MultiLine property.

Example for the FilterComboBox property: Filter mode of a form Streets with a text box Name. If the user clicks the button, he will see a list of all street names in the whole table. He can select one in the list, and then click Filter to select the record. The FilterComboBox property executes a Select distinct to avoid the possibility that items are shown twice.

Example for the ValueDefault property: ValueDefault can either be used for AutoIncrement, AutoGUID or date/time values.



Topobase Form Designer: Assigning a default value.

NOTE FileSelection in the Topobase Web version: Use the table TB_GN_DIRECTORY_ALIAS to handle the link to a file name. In view mode, the first part of the file name is replaced (mapped) using this table. In this way it is possible to show the linked files in the Web browser.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer Example: Validation Code (page 358)
- System Table TB_GN_DIRECTORY_ALIAS (page 75)

Form Designer: Read-Only SQL TextBox - Properties

Use the Read-Only SQL TextBox (SqlLabel) control to display text in the feature class form that is queried with an SQL statement. For example, you can display the area of the related polygon in a centroid feature class. You can also query and display data from different attributes and tables.

Test the SQL statement in the SQL Assistant.

Property	Description

System Category	
ClassName	SqlLabel
Name	Specifies the name. For example, $SQLLABEL1.
Specific Category	
AlignRight	If True, aligns the value. Specifies whether the value is aligned to the right edge of the input box.
FontName	Selects a windows font including style and size.
ForeColor	Specifies the color of the text.
Precision	Specifies the precision of numeric values.
Sql	Defines SQL statements to query any information.
UnAvailableValue	If the SQL returns no result, this value is displayed.
UseBuffer	Specifies whether the buffer for the data can be switched off. If True, you can switch the buffer off the data. In some cases, it is necessary to see real data. Default = True.

Notice the difference between Read-Only SQL TextBox and Editable SQL TextBox. Both controls allow you to query data, but in one case the result is displayed as a label, which means that it cannot be edited. In the other case, the result is displayed in a text box, and the value can be edited and saved (updated) in the database.

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer: Editable SQL TextBox - Properties (page 339)
- Using the SQL Assistant (page 105)
- Form Designer Example: Read-Only SQL TextBox (page 362)
- Form Designer Example: Editable SQL TextBox (page 364)

Form Designer: Script Programmable - Properties

Use the Script Programmable (ScriptButton) control to execute small scripts. Its function is similar to the Formula TextBox control. You can use VB .NET and the Topobase API in the scripts.

Property	Description
System Category	
ClassName	ScriptButton
Name	Specifies the name. For example, $SCRIPTBUTTON1.
Specific Category	
ButtonCaption	Specifies the text that is displayed on the button.
ScriptCode	Click the property row to open the editor. Opens a script code editor, where you can edit the code.

Sample Script:

```
dim c as Topobase.Forms.DialogControls.ComboBox

c = Me.Dialog.Controls.Item("Fid_Street")

Msgbox( "The Street has the Fid:" & c )
```

See also:

- Form Designer: General Control Properties (page 324)
- Form Designer: Formula TextBox - Properties (page 344)

Form Designer: SQL Form Launcher - Properties

An SQL Form Launcher (ComplexReference) control is not linked to a database attribute of the current table. With Complex Reference, you can open the form of another table, using a filter. You can define this filter using complex SQL statements.

Test the SQL statement in the SQL Assistant.

Property	Description
System Category	
ClassName	ComplexReference
Name	Specifies the name. For example, $COMPLEXREFERENCE1.
Specific Category	
DetailTable	Specifies the table you want to open. Select the table.
Filter	Defines the filter that will be executed. Set the values that should be replaced in {}. Click to open the SQL Assistant. Examples: *FID = {FID_CHILD}.* *name = (select typename from specialtable where height = {objectheight} and width={objectwidth})*

You can use the ComplexReference control with input parameters, such as a radius, which the user is prompted to enter. An input box will appear. If the parameter is a numeric value, you must use the # character. If it is a text value, use the $ character.

Example: {#Radius} {$Name}

```
SDO_WITHIN_DISTANCE(a.Geom,
(select geom from LM_POINT where fid={fid}) ,
'distance = {#Radius}') = 'TRUE'
```

See also:

- Form Designer: General Control Properties (page 324)
- Using the SQL Assistant (page 105)

Form Designer: Static Picturebox - Properties

Use the Static Picturebox (StaticPicture) control to show pictures in the feature class form, for example, your company logo.

Property	Description
System Category	
ClassName	StaticPicture
Name	Specifies the name. For example, $STATICPICTURE4.
Specific Category	
FileName	Specifies the directory and file name of the picture to be shown in the form (**.gif*, **.bmp*, and so on).
StretchImage	Specifies how the image is displayed. If True, the picture is stretched to the size of the control. If False, the picture is shown in its original size.

NOTE Although the picture files can be stored in any folder, it is recommended that you store them in the <topobase>\Pics folder or a subfolder. Then, they are available with the dynamic path <topobase>\Pics and you need to enter the subfolder/filename only. If you use picture files with a static path, they are not available on a different computer unless the path is exactly the same. Alternatively, you can use *//Servername* to store the picture files on a server.

To show the pictures both in Topobase Client and Topobase Web, make sure that the pictures are stored both in the <topobase_client>\Pics folder and the <topobase_web>Pics folder or subfolder.

NOTE To show the pictures in the Web version, if you do not use the <topobase>\Pics folder, you must map the folder names in the system table TB_GN_DIRECTORY_ALIAS.

See also:

- System Table TB_GN_DIRECTORY_ALIAS (page 75)
- Form Designer: General Control Properties (page 324)

Form Designer: Static Text - Properties

Use the Static Text control to display static text in the feature class form, for example, explanations.

Property	Description
System Category	
ClassName	Label
Name	Specifies the name. For example, $LABEL5.
Specific Category	
AlignRight	If True, aligns the value. Specifies whether the value is aligned to the right edge of the input box.
FontName	Selects a windows font including style and size.
ForeColor	Color of the text
Text	Specifies the text you want to write onto the form. Enter the text.

See also:

- Form Designer: General Control Properties (page 324)

Form Designer: Model Selector - Properties

Use the Model Selector control to add the Model selector to a feature class form.

Property	Description
System Category	
ClassName	
Name	Specifies the name. For example, $MODELSELECTOR.

Form Designer: Examples

The following examples provide information you need to know to work with the form designer, including:

- Setting the tab order
- Defining validation code
- Defining a formula or functions (some sample formulas are provided)
- Using a read-only SQL text box
- Using an editable SQL text box
- Showing geometry with SQL

See also:

- Using the SQL Assistant (page 105)

Form Designer Example: Auto Tab Order

Using the Auto Tab Order option

When entering data, you press the <TAB> or <ENTER> key to move to the next input field. You use the Auto Tab order option to define the order in which the cursor will move (in Form view mode).

To define the tab order

1. Start Topobase form designer.
2. In the form explorer, select the feature class, select the Form Type, and click Edit.
3. In the Designer dialog box, click Edit menu ➤ Auto Tab Order. If you select Auto Tab Order, the cursor will move according to default settings: from left to right and from top to bottom.
4. Clear Auto Tab Order to define your customized order.

5 In the right pane, click the first control.

6 In the left pane, click the Properties tab.

7 Select NextControl and click . Select the next control in the list.

Form Designer Example: Validation Code

You can define validation code (check rules) according to the same rules as those for defining formulas (using VB .NET). The code defines the value "result", which can either be True or False. If True, the value is accepted. If False, the modification is blocked and the user has to correct the input until it fits the rules. The application stays in Edit mode until the input is correct.

The following example from the water utility application assumes that a hydrant has two elevation attributes: ELEVATION is the height of the point where the hydrant is connected to the water network, GROUND_ELEVATION is the height of the hydrant above sea level.

To define a check rule to ensure that the GROUND_ELEVATION is always higher than the ELEVATION

1 Start Topobase form designer.

2 In the form explorer, select the feature class Hydrant, select the Form Type, and click Edit.

3 In the Designer dialog box, in the right pane, click the General tab and select the GROUND_ELEVATION text control.

4 In the left pane, click the Properties tab.

5 Select the ValidationCode property and click .

6 In the Validation Code dialog box, enter the validation code. You can double-click the attribute names in the list to insert them into your code.

```
if {ELEVATION} < {GROUND_ELEVATION} then

Result=True

Else

Result=False

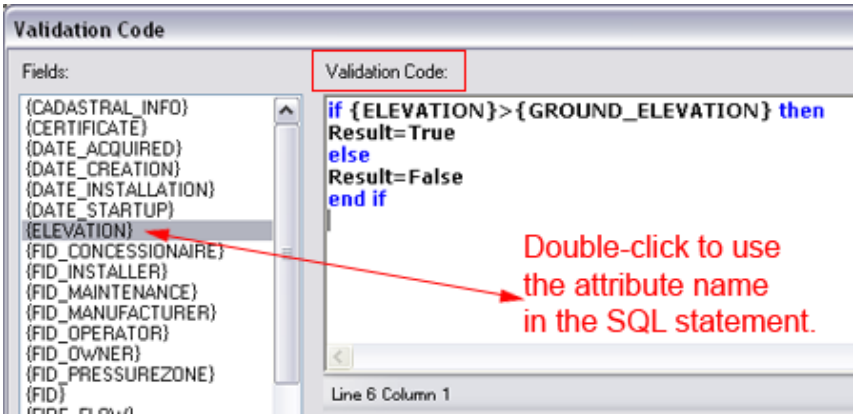
End if
```

7 In the Validation Code dialog box, enter the following text that is displayed if the validation code returns False:

```
Elevation must not be higher than ground elevation!
```

8 Click OK to close the Validation Code window.

To see the validation, open the Hydrant feature class form and enter values for ELEVATION and GROUND_ELEVATION. The error message is displayed if you enter an ELEVATION value that is higher than the GROUND_ELEVATION value.



Control input with a validation code. If the user enters a value that does not match, the specified error message is displayed.

See also:

- Form Designer Example: Formula TextBox (page 359)

Form Designer Example: Formula TextBox

You can define formulas or functions according to the VB .NET rules. For more information, see *http://msdn.microsoft.com/vbasic*.

Use the Formula control to add, subtract, multiply or divide values.

To define a formula control

1 Start Topobase form designer.

2 In the form explorer, select the feature class, such as DOC_GREENSPACE.

3 Select the Form Type, such as Default, and click Edit.

4 In the Designer dialog box, on the designer toolbar, click the Formula Textbox icon.

5 Click OK for the name.

6 In the left pane, click the Properties tab.

7 Select the Formula property.

8 Click to open the formula editor. In the formula editor, all current database attributes are listed in the attributes pane.

9 Double-click an attribute name to use it in the formula. In the formula, you may use all valid VB .NET expressions, for example, you can use *result = {AREA} * {MAINTENANCE_COSTS}*.

10 In the formula editor, click Check Code to check the formula for syntax errors, using the current record.

11 Click Test to check the result with the current record set.

12 Click OK.

13 Exit the form designer.

Formula TextBox: Sample Formulas

Showing the FID

```
result="The FID is " & {FID}& "."
```

Showing the result of any selection

```
result = Me.ConnectionTools.LngValue("select count(*) from <table>
 where column=" & LngValue("parent_fid"))
```

Showing whether geometry is available and to set the color of the box

```
if Me.GeometryAvailable then

result="Geometry available"

 Me.ForeColor=""

Me.BackColor=""
```

```
else
 result="No Geometry available"
Me.ForeColor="white" 'These are HTML colors !
Me.BackColor="red"
end if
```

Showing how many Microsoft documents (DocumentManager) are available and to set the color of the box

Notice, that the temp variable c is used in order to avoid calling the function twice:

```
dim c as integer
c = me.DocumentCount
if c=0 then
 result="No Documents available"
Me.BackColor=""
else
 result=c & " Document(s) available"
 Me.BackColor="red"
end if
```

NOTE This example shows the concepts. However, it is much easier to use the Document Manager control for this task.

See also Form Designer: Document Linker - Properties (page 338).

Calculating with the attribute values

```
Result = {HEIGHT} + 20
Result = {HEIGHT} * 20
Result = {HEIGHT} / 20
Result = {HEIGHT} - 20
```

Combinations with other attribute names are also possible, for example:

```
Result = {HEIGHT_1} + {HEIGHT_2} *10
```

```
Result = {attribute name1} * {attribute name3} *10
```

Assembling the contents of two text attributes. Delete or add the characters of text fields (data type varchar2)

```
result = {TEXT1} & {TEXT2}
```

Appending a prefix and a suffix

```
result = "Approx. " & {AREA} & "m2"
```

Generating a manhole name from an FID

```
result = "MH " & {FID}
```

Calculating the remaining days until next maintenance (using a feature class attribute MAINTENANCE_NEXT_DATE)

```
Dim d As Date

Dim t As System.TimeSpan

d = Me.Value("maintenance_next_date")

t = d.Subtract(now)

result=t.Days
```

Form Designer Example: Read-Only SQL TextBox

You can display text in the feature class form that is queried with an SQL statement. The following example shows how to display the label text that is stored in the label feature class (page 523) *_TBL directly in the parent feature class form.

To display the label text in the parent feature class

1. Start Topobase form designer.
2. In the form explorer, select the feature class, select the Form Type, and click Edit.
3. In the Designer dialog box, on the designer toolbar, click

.
4. Click OK for the name.

5 In the left pane, click the Properties tab.

6 Select the UnAvailablValue property and enter the following text: *No label text available*.

7 Select the SQL property and click to open the SQL Assistant.

8 In the SQL Assistant dialog box, under Relations, expand the feature class, and click the label feature class *_TBL. The SQL expression is inserted into the expression input box.

9 Replace the asterisk (*) in the SQL expression with LABEL_TEXT. Result is:

```
select LABEL_TEXT from <featureclass>_TBL
where FID_PARENT in (select FID from <featureclass> where
fid={fid})
```

10 Click Execute to run the SQL statement, and to check the syntax.

11 Click OK to exit the SQL Assistant.

12 Exit the form designer.

13 In the feature class form, browse for the records. If a record has no label, the UnAvailableValue is displayed instead of the label text.

NOTE A feature can have more than one label, which is not considered in the above example.

Example: SQL label statement to display the area of an area topology in the centroid feature class:

```
select ROUND(bts.EXACT_AREA,6)
from BF_TSUR bts, BF_TCEN btc
where btc.FID_TSUR = bts.FID
and btc.FID_CENTROID = {FID}
select exact_area from LM_T_LANDUSE_TSUR
 where FID in (select FID_TSUR from LM_T_LANDUSE_TCEN
 where FID_CENTROID in (select FID from LM_LANDUSE
```

```
where fid={fid}))
```

See also:

- Form Designer: Read-Only SQL TextBox - Properties (page 351)
- Using the SQL Assistant (page 105)
- Data Model: Labels (page 187)

Form Designer Example: Editable SQL TextBox

You can display text in the feature class form that is queried with an SQL statement. The following example shows how to use an editable text box to display data.

This example shows how to display the maintenance date of the green space the tree belongs to.

1. Start Topobase form designer.
2. In the form explorer, select the feature class, select the Form Type, and click Edit.
3. In the Designer dialog box, on the designer toolbar, click the Editable SQL Text Box icon.
4. Click OK for the name.
5. In the left pane, click the Properties tab.
6. Select the DetailTable property, click and select the feature class.
7. Select the DetailAttribute property, click and select the attribute.
8. Select the DetailFilter property and click to open the SQL Assistant.
9. In the SQL Assistant dialog box, under Relations, expand the feature class nodes and click the detail feature class that contains the information you want to display. The SQL expression is inserted into the expression input box.

```
select * from DOC_GREENSPACE where FID in (select FID_GREENSPACE
from DOC_TREE where fid={fid})
```

10 In the expression input box, delete the first part of the select statement including the WHERE. The result is:

```
FID in (select FID_GREENSPACE from DOC_TREE where fid={fid})
```

11 Click Execute to run the SQL statement, and to check the syntax.

12 Click OK to exit the SQL Assistant.

13 Select the DetailFilter property again, and click . Note that the expression input box shows how the select statement has been put together from the three parts.

The select statement result is shown in a text box. If a user modifies this value, the related table is updated.

See also:

- Form Designer: Editable SQL TextBox - Properties (page 339)
- Using the SQL Assistant (page 105)

Form Designer Example: Advanced Combobox

The form designer uses Advanced Combobox (MasterBox) controls to control the filter of a combo box. For example, instead of selecting a street in a large dropdown list showing all streets of all cities, select the city first and then the street from a smaller list of pre-selected items.

In a feature class form, the advanced combo box is controlled by the combo box. For example, the advanced combo box has no relation to any attribute of the form. It is related only to the combo box. If you browse, you will always see the corresponding values, such as city, district, or street. If you change any of these values, the corresponding location is emptied.

The specific properties can be set in a special Master Box properties window.

Click in any of the master box property rows to open this window.

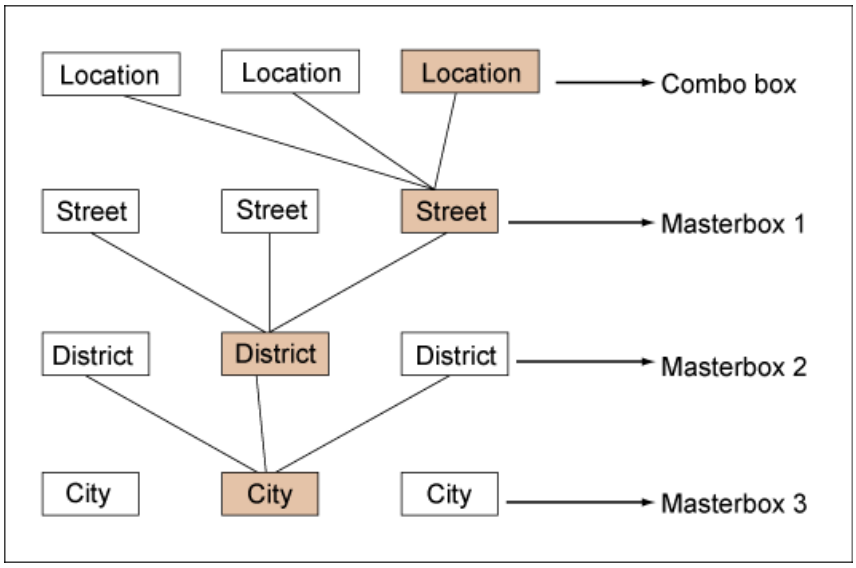
Advanced Combobox properties example

- Detail Control: FID_STREET
- Master Table: LM_CITY
- Detail Table:
- Master Attribute: FID
- Detail Attribute: FID_CITY
- Display Attributes: NAME
- OrderBy: NAME

Example: Multi-relation Advanced Combobox: City ->District -> Street ->Location

Target: To assign a location (house number) to a building with the help of a combo box (BUILDING.FID_LOCATION <- LM_LOCATION.FID).

The user does not want to select the location from the table LM_LOCATION. He wants to select the city first, then the district and street, and finally the location. Each drop down list will only contain the pre-selected items.



Form Designer: Using an advanced combo box control, you can set one or more filters to assign the location. In this example, the user can restrict the selection of the location by city, district, and street.

The following relations exist: There are several cities; every city has several districts and every district has streets and every street has locations (house numbers).

```
LM_LOCATION { FID, ..., Number, FID_STREET }

LM_STREET { FID, ..., Name, FID_DISTRICT }

LM_DISTRICT { FID, ..., Name, FID_CITY }

LM_CITY { FID, ..., Name }

PARENT_TABLE_NAME.PARENT_COLUMN_NAME

-> CHILD_TABLE_NAME.CHILD_COLUMN_NAME

LM_CITY.FID -> LM_DISTRICT.FID_CITY

LM_DISTRICT.FID -> LM_STREET.FID_DISTRICT

LM_STREET.FID -> LM_LOCATION.FID_STREET

LM_LOCATION.FID -> BUILDING.FID_LOCATION
```

The above relations must be defined in TB_RELATIONS. Now you can configure the master boxes for the feature class form Building. Master boxes are created for the FID_LOCADTION combo box.

To define an advanced combo box control

1. Start Topobase form designer.
2. In the Designer dialog box, in the right pane, select the control FID_LOCATION. Change the DetailList attribute property to HOUSE_NUMBER and the DetailSortOrder property to HOUSE_NUMBER. With this you make certain that the house number is displayed instead of the FID. Note that the Detail ListFilter will not work in this special case, so do not use it.
3. On the Designer toolbar, click the AdvancedComborBox icon. A control named MASTERBOX1 is created. Change its properties in the left pane as follows:
   - Caption = Street:;
     Detail Control = FID_LOCATION = name of the combo box
     Master Table = LM_STREET
     Master Attribute = FID
     Detail Attribute = FID_STREET

Display attributes = NAME (attribute with street name)

Orderby = NAME (sorts by street name)

4 Click to add another advanced combo box. A control named MASTERBOX2 is created. Change its properties in the left pane as follows:

- Caption = District:
  Detail Control = MASTERBOX1 (the above Street box)

  Master Table = LM_DISTRICT

  Master Attribute = FID

  Detail Attribute = FID_DISTRICT

  Display attributes = NAME (attribute with name of district)

  Orderby = Name (sorts by district names).

5 Click to add another advanced combo box. A control named MASTERBOX3 is created. Change its properties in the left pane as follows:

- Caption = City:
  Detail Control = MASTERBOX2 (the above District box)

  Master Table = LM_CITY

  Master Attribute = FID

  Detail Attribute = FID_CITY

  Display attributes = NAME (attribute with the city name)

  Orderby = Name (sorts by city names)

6 Now, relocate the controls as you want, and exit the form designer.

In case any SQL errors occur, check the settings of the above steps. Also, check to be certain that the indices of the attributes you have used are set up correctly.

NOTE Any use of combo boxes, master boxes and so on will debase performance of the feature class form, because additional SQLs must be executed each time you change the record. This is especially true for Table View mode, which requires many record moves to show the visible records of the grid. Therefore, in table view mode you will notice quite a slowdown if you use basic combo boxes or advanced combo boxes.

TIP To optimize the speed, check whether you have set indexes on the related attributes.

See also:

- Form Designer: Advanced Combobox - Properties (page 326)

Form Designer Example: Coordinates Editor with SQL

You can use the SQL property to show geometry that is not stored in the current table, but in a related table. The current table may also be an attribute table without inherent geometry, but with a related feature.

By default, the SQL property of a geometry control is empty. In this case, the geometry of the current form is shown. You can use an SQL statement to show the geometry of a feature, which is saved in a related table (a "related child" feature).

Some rules:

- See Using the SQL Assistant (page 105)
- An SQL statement always returns an FID.
- Attribute names must be written in braces, for example {FID}, {Y1}.
- You can use SQL statements that follow the relation of up to ten tables. If the SQL returns more than one FID, only the first FID is taken. Geometry controls using the SQL property always are read-only (locked).
- All attributes in {} are replaced by the actual value. The {} attributes are existing in the current form (in the following example: the Line table).

Geometry Controls with SQL - Examples

Showing the geometry of a related point

The feature class form (table) WA_LINE has an attribute FID_STARTPOINT that is related to a table WA_POINT from which the geometry shall be shown.

```
select FID from WA_POINT where fid={FID_STARTPOINT}
```

or:

```
{FID_StartPoint}
```

The FID is unique in the whole database, therefore the system can find the feature class (table) with this FID without knowing the table name.

Showing the coordinates of the manholes at the beginning and the end of a section in the WW_SECTION form

The manhole FIDs are saved in the WW_SECTION form in the attributes {FID_first_manhole} and {FID_last_manhole}.

In the WW_SECTION form: Create a geometry control and define the SQL:

```
{fid_first_manhole}
```

The detailed SQL is:

```
select FID from WW_MANHOLE where fid={FID_first_manhole}
```

The SQL statement has to yield an FID. The geometry of this FID is shown.

Showing the pipe geometry in a sewage works form

SEWAGE_WORKS.FID_STRUCTURE; STRUCTURE.FID_PIPE; PIPE.GEOM

```
select FID_PIPE from STRUCTURE where FID = {FID_structure}
```

Selecting

```
select FID_<t1> from t2 where FID in (select FID from <t3> where
FID_<t4>={t4} and structuretype ='iron')
```

See also:

- Form Designer: Coordinates Editor - Properties (page 337)
- Showing the Coordinates of Utility Features (page 372)

Form Designer Tutorial

Creating Controls

To create a control

1. Start Topobase Administrator and open the workspace.
2. Start Topobase form designer.

3 In the form explorer, select a feature class, and do one of the following.
   - Select the Form Type, such as Default, and click Edit.
   - Select the Form Type, and click Duplicate.
   - Click Add.

4 In the Designer dialog box, on the left pane, notice the Form Type name.

5 If the Designer dialog box shows different tabs, click the tab you want to add the control to. Usually there is one All tab. All controls on this tab are available in all other tabs.

6 On the Designer toolbar, click a control icon, such as Static Text .

7 Enter the name of the control and click OK. The control is added to the Form pane.

8 Select the control in the right pane.

9 In the left pane, click the Properties tab and define the control's properties.

Defining Properties

Each control has its own set of properties which can be edited directly. Select (click) the control you want in the form pane. On the Properties tab, on the left, you can either enter the values directly or click to continue.

Example 1: Modifying a caption:

- In the Form pane, click a caption. You can enter the new value directly in the Form pane.

Example 2: Defining a validation code:

- Select a control.
- In the Base category, click the ValidationCode row in the left pane. A button appears. Click it to open the Validation Code dialog box and define the validation code.
- See also Form Designer Example: Validation Code (page 358)

Example 3: Removing controls from the form.

- Select the control in the Form pane.
- Drag it from the right to the left pane.

The control is added to the Hidden list on the Controls tab on the left. From there you can drag it back to the Form pane.

NOTE New database attributes that have recently been added are not visible in the form. You can drag them from the Hidden list into the form pane.

Showing the Coordinates of Utility Features

Use the Coordinates Editor control to show the coordinates of a feature, even if they are not stored in the feature table itself. The Topobase utility model (page 526) stores the utility objects separately from the geometry. There are several attribute feature classes that share one geometry feature class. The following example is from the water utility model:

Example: All point objects, as for example the point object WA_ARMATURE, use the feature class WA_POINT and all line objects use the feature class WA_LINE to store the geometry. Therefore no coordinates are stored in the WA_ARMATURE feature class.

The following steps show the concept and explain how to display the coordinates in an Armature feature class form. To reproduce the steps, use the Water demo data set.

To show the geometry of a utility object

1. Start Topobase form designer.
2. In the form explorer, select the feature class Armature, select the Form Type, such as Default, and click Edit.
3. In the Designer dialog box, click the General tab.
4. On the Designer toolbar, click Coordinates Editor.
5. Click OK for the name.
6. Select the control and in the left pane edit the properties:
   UnAvailableValue = No coordinates available.

7 Exit the form designer.

8 Select an armature in the feature class form. The new geometry field shows the text you entered earlier, because the coordinates are not stored in WA_ARMATURE.

With a geometry select statement you can retrieve the coordinates from the related geometry table WA_POINT.

9 In the form explorer, select the feature class Armature, select the Form Type, such as Default, and click Edit.

10 In the Designer dialog box, click the General tab.

11 In the right pane, select the coordinates control.

12 In the left pane click, the Properties tab.

13 Under Specific, click the SQL property and click to open the SQL Assistant.

14 In the SQL Assistant dialog box, under Relations, expand the feature class WA_ARMATURE, and click the related feature class WA_POINT. The SQL expression is inserted into the expression input box.

```
select * from WA_POINTwhere FID_ATTR in (select FID from
WA_ARMATUREwhere fid={fid})
```

15 In the SQL Assistant, click Ex ecute to check the syntax.

16 Click OK to set the property.

17 Exit the form designer.

If you select the same armature as before, you can see the coordinates in the feature class form.

Variant

To display the X (easting) and Y (northing) coordinate in two separate fields, add another geometry control and use the following attributes:

- Set one Control to True the other to False. Either:
- ShowX=True and ShowY=False
  or:

- ShowX=False and ShowY=True

Defining Tab Pages for Forms

You can assign several tab pages to a form, and customize the tabs. You use controls on the tab pages to set up the information that is displayed in a form.

Using the Tab Pages commands on the Designer menu, you can create or edit the tabs. (Example can be reproduced in workspace TB2007_LM).

To define additional tab pages

1. Start the Form Designer
2. In the form explorer, select the feature class Point, select the Form Type, such as Default, and click Edit.
3. In the Designer dialog box, click Tab Pages menu ➤ Add Tab.
4. Enter a name, such as "Special Form" and click OK.
5. The new tab page is added to the right pane. Note that the All tab shows all controls and is only visible in the form designer. Controls that are visible in all tab pages are indicated by a blue caption.
6. In the left pane, click the Controls tab.
7. In the right pane, click the All tab and remove all controls that you want to display on other tab pages. For example, do not remove the FID control if you want to display this on all tab pages. Shift+click to select multiple controls. In the left pane, in the Hidden list you can see the controls that you removed.
8. To add controls to a tab, in the right pane, click the tab and drag the controls that you want to add from the left pane to the tab on the right.
9. Next, define the Special Form tab.
10. Exit the Designer.

Defining Form Toolbars

You can assign a special toolbar to each form and each user group. Usually you will define toolbars for the different types of tables, such as feature class (page 522), label feature class (page 523), domain (page 522), and so on. For

example, a domain table toolbar does not need any tools related to graphic connection.

By default, there are some toolbar definitions: Feature, Domain, Label, and Others. When you create a new form, one of these toolbars is assigned:

- Feature class form: Feature toolbar type
- Label feature class form: Label toolbar type
- Domain: Domain toolbar type
- system tables and other tables: Others toolbar type

By configuring toolbars, you can hide or show certain icons.

To assign a toolbar to a form

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, select the document.
3. In the Administrator, click Document menu ➤ Form Designer.
4. In the form explorer, select the feature class, select the Form Type, and click Edit.
5. In the Designer dialog box, click Designer menu ➤ Options.
6. In the Options dialog box, assign a toolbar to the form.

See also:

- Designer Menu: Options (page 316)

To configure a form toolbar

1. Start Topobase Administrator and open the workspace.
2. In the Administrator, click Document menu ➤ User Interface.
3. In the right pane, click the Forms tab.
4. Select the user group whose form toolbars you want to configure.
5. Click Add to create a new type of form toolbar.

6 Expand the items by clicking the + in front of the selected entry. All available tools are listed. Select or clear the buttons that you want to hide or show in the form.

NOTE Hiding a button in a toolbar definition affects all forms that use this toolbar.

NOTE The toolbar definition list also includes the setting that specifies the commands that are in the related shortcut menus.

See also:

- Defining Shortcut Menus for Feature Class Forms (page 376)
- Customizing Form Toolbars and Menus (page 33)

Defining Shortcut Menus for Feature Class Forms

You can customize the shortcut menus of the feature class forms. You perform the same steps as for configuring form toolbars.

See also:

- Defining Form Toolbars (page 374)

To define a shortcut menu for a form

1 Start Topobase Administrator and open the workspace.

2 In the Administrator, click Document menu ➤ User Interface.

3 In the right pane, click the Forms tab.

4 Select the user group whose form shortcut menus you want to configure.

5 Select the form type you want to configure, such as Feature.

6 Expand the list. All available commands are listed. Select or clear the commands that you want to show or hide in the form.

Assigning a Feature Class Form to a User Group

For each user group you can assign specific feature class forms. See also Form Designer: Managing Multiple Forms (page 310).

To assign a form to a user group

1. Start Topobase Administrator and open a workspace.
2. In the Administrator explorer, select the document.
3. In the Administrator, click Document menu ➤ Form Designer.
4. In the form explorer, select the feature class.
5. Under User Groups, for each user group, select the appropriate form type.

Assigning Help Files to Feature Class Forms

You can configure a Help system for either desktop or Web version. Also, you can assign Help files to each form and each control.

To assign Help topics to a form

1. Start Topobase Administrator and open the workspace.
2. In the Administrator explorer, select the document.
3. In the Administrator, click Document menu ➤ Form Designer.
4. In the Administrator, click Document menu ➤ Form Designer.
5. In the form explorer, select the feature class. Select the Form Type, and click Edit.
6. In the Designer dialog box, click Designer menu ➤ Options.
7. In the Options dialog box, in the Help section, enter the properties, as for example:

   File Name: *topobase.chm*

   Context: *form_designer_introduction.htm*

The Help file and topic are displayed when the user presses F1 and the cursor is not on a control.

You can also assign a Help topic to each control. This topic is displayed when the user presses F1 while the cursor is on that control. In this case, you must select the control and edit the Control properties. After editing the properties, you can enter the Help file name and Help context again. See also Form Designer: Property TextBox - Properties (page 349)

Working with Display Models

7

Understanding Display Models

In Topobase, you use display models to administer thematic views. A display model definition specifies which set of layer files (feature layers) is loaded into the Display Manager. Use the AutoCAD Map 3D Display Manager to create and edit the styles for the feature layers, see also Overview of the Display Manager (page 387).

Also, in the display model you define multi map windows and autoload layers.

You apply different display models to render the data of a workspace. Display models are stored in the file system, and you can export and import the display models to share with other users and across workspaces. Display models are stored in the file system in (*.*xml*) files.

Autoload Layer

In a display model, you optionally specify autoload layers that are displayed automatically when you select a Display Model. You do not need to generate graphic explicitly. Use autoload layers as context for the spatial selection of your viewport. For better performance, autoload layers should not contain too many details. For example, set the administrative boundaries as Autoload layer. In the Display Model definition, you can set one or more layers as autoload layers.

Multi Windows

You can define display models that display the features in multiple drawings. That means that your display model can consist of several display model maps.

For each drawing window you can define different stylizations. For example, define one window to display all feature layers, using basic stylization, and another window to display a selection of feature layers, using a thematic style. In Topobase Electric, you might use a second window to display the network in a schematic view. Both windows are connected to the database. Operations such as highlighting a feature apply to both windows.

Default Display Model

Topobase provides a temporary default display model that allows easy and quick rendering of the workspace data. Temporary means that the display model is loaded in memory, but not yet stored in the file system. However, you can modify the temporary display model and save it as display model for further use. When you generate graphic using the default display model, a default style applies to each feature class. You cannot influence the styling of a default display model. You can create new display models and save them for all your stylization requirements.

The default display model contains all layers of the workspace, that means all feature layers of all documents. When you open a workspace, and select Open Default Display Model, it is created automatically. You use the default model, if you have not yet defined any stylization, which also means that you have not yet specified any Autoload layers.

NOTE If you open the default display model in a job-enabled document, you can optionally track feature deletion. See also Styling Job-Enabled Features (page 412).

To use the default display model

1. Select the workspace.
2. Click Home tab ➤ Display panel. In the Display Model list, click Open Default Display Model.

   This loads all feature layers of the document into the memory.
3. Click Generate Graphic.

   All feature layers are displayed in the Display Manager. For documents with many feature classes, this may take some time.
4. Click OK to continue.

In the Display Manager, stylize the feature layers. Under Display Model, click Save. We recommend that you save each display model in a separate folder.

NOTE In the default display model, label feature classes with the extension _TBL are loaded as feature layers. See also Styling Annotation Layers (page 398).

A display model consists of several files. See also Storing Display Models (page 381).

For detailed information about Generate Graphic and defining viewports, see the Client User Guide, section Displaying Features In Your Map.

Display options

You can control the behavior of display models and Generate Graphic by several application options. See Setting Generate Graphic Application Options (page 22).

- Number of open drawings
- Highlight features

Saving Drawing Objects

In Topobase Client, the Generate Graphic command creates FDO features. When you save the graphic as **.dwg* using the File menu ➤ Save command, the file contains only the FDO connection information but no drawing objects. For example, when you open this **.dwg* file on another computer, you have to reconnect to the database.

You can save the graphic as a **.dwg* containing drawing objects. In Display Manager, click the Tools icon and select Save Current Map To DWG. Using this command, the FDO features are exported as drawing objects. This command does not export 3D data, and it does not export feature attribute data.

To export the graphic into a **.DWG* file that contains the feature attribute data as Xdata, you use the Export To DWG command. This command exports drawing objects, including 3D data. This command requires that the FDO styles have corresponding AutoCAD styles. For more information see the Topobase Client User Guide, section Export to DWG.

Storing Display Models

A Topobase display model consists of several files as shown in the following table.

Display model file	Description

<display model name>.tbdm	XML file that stores the name, related display model maps, Z order of the display model maps. A display model consists of one or more display model map files.
<window name>.tbdmmap<nr>	Display model map file: XML file that stores the size, position, and references to the feature layers of each window. If you define multi windows, for each window one display model map file is used.
<feature class name>.LAYER	Layer file (XML file). For each layer that is referenced in the display model map file.
<prototype>.DWG	DWG file containing blocks

We recommend that you store the display models in a file system that represents the structure. For example, save each display model (set of files) in a separate folder. Or create separate folders for layer files that you want to use in multiple display models.

In the application options, you specify the Repository Base Path to the central location of the display model files. See also Setting Generate Graphic Application Options (page 22).

Modifying Display Models

If you Generate Graphic before saving modifications of your layer settings, you are prompted how to proceed. In the Layer Display Settings Changed dialog box, select one of the following.

- **Keep**—To Generate Graphic using the modified layer settings. The changes are not saved in the display model.
- **Discard**—To discard the modifications, and to Generate Graphic using the selected display model.

To save the changes in the display model, click Home tab ➤ Display panel ➤ Save Display Model.

Creating Display Models

You create a display model either by modifying the default model, or by using the Display Manager to style the feature layers from scratch.

You can easily extend an existing display model by adding new layer files. The display model files use references to store the layers. For example, open an existing display model, then add new layer files to the Display Manager, and save the files as a new display model.

To create a display model from scratch

1. Use the Display Manager to style the feature layers. See Bringing In Features From Topobase (page 388).
2. Click Home tab ➤ Display panel ➤ Save Display Model As.
3. In the Display Model dialog box, specify the properties as shown in the table below.
4. We recommend that you save each display model in a separate folder. For example, save it to *<topobase_client>\DisplayModel\<name>*.

Display Model Properties

In the Window List, and in the Layer List, use drag-and-drop to change the order of the columns. Right-click the title row to select the columns you want to show or hide. Show and hide the columns by right clicking and checking and unchecking

Display Model	Description
General area	
Name	Specifies the name of the display model.
Display Deleted Features	This option is only available, if the document is job enabled. And this option only applies, if the display model has been created using the Display Deleted Features option. For display models that have been created using the Display Deleted Features option, you clear this option, to display only the features of the job state Live. Then, you do not to display features that have been deleted, or modified, or created in the current job. With this option is enabled, the Display

	Manager uses the Topobase text function TB_SQL for TB_JOB_OPERATION _ID. See also Styling Job-Enabled Features (page 412).
Window List area	Displays one or more drawing windows that are included in the display model. For each window, stores the display model map properties such as size, location and state. In the data grid, click the Name field, to modify the name. See also Multi Windows (page 379).
Layer List area	Displays the feature layers that are loaded in the Display Manager.
Name	Displays the layer caption, as displayed in the Display Manager. Related to <Name> in the (**.tbdmmap*) file.
Feature Class	Displays the feature class name.
Group	Displays the Display Manager group.
Autoload	Specifies whether the layer is an Autoload layer. Select this option if you want to draw the feature layer automatically upon selection of the display model. See also Autoload Layer (page 379).
Filter	Saves a filter that has been applied to the selected feature layer. ■ Example—In Display Manager, select a feature layer, right-click Query To Filter Data, and define a filter. ■ Example—Zoom into the drawing, and click Generate Graphic. Then, the filter is set to the extents of the current Window viewport. When you save the display model, the check box will be selected, and the filter will be saved in the LAYER file. Clear this option, if you do not want to save the filter.
Confine To Viewport	Specifies whether the layer will have its viewport updated on Generate Graphics or not. Use this option to control which feature classes are represented in the current viewport and which are not.

	Select Confine To Viewport to draw feature data based on the current viewport. When you zoom out, features outside the viewport are not visible. Clear Confine To Viewport to draw all feature data without clipping. When you zoom out, features outside the viewport are visible, and can be used as context for another Generate Graphic. For example, for feature layers that represent the base map content, or a map grid. See also Setting Map Options (page 17), Always Display Highlighted Features.
Selectable	Specifies whether the layer is marked as Make Layer Selectable in the Display Manager. For example, to change multiple layers without having to right-click on each layer and without editing the (**.tbdmmap*) file.
Source	Displays the document name for display models that contain layers from different documents with the same name. For example, in a workspace with two water documents, you can add two feature layers WA_POINT to the display model, one of each document. Then, the Source of each document is displayed. **NOTE** If your display model has feature layers with a Source assigned, you cannot share the display model with other documents, because in this case, the document names are stored with the display model.
Ref.	This setting can only be changed if you are saving a display model. Specifies whether the layer is a referenced layer. Clear this option to create a copy of the layer (LM_Building_2.layer), leaving the original referenced layer file untouched. For example, to open a shared Display Model on a read-only network share, and to save it as a working copy in your own repository. See also Sharing layers across display models (page 386).
Path	Displays the path where the LAYER file is stored. The path is relative to the Repository Base Path that is specified in the application options; see Setting Generate Graphic Application Options (page 22).

NOTE The path also indicates whether the display model shares layers with other display models.

Sharing layers across display models

To share layer stylization, add any existing LAYER files from other display models to your display model. You can either copy existing layer files into your display model directory, or you can reference an existing layer.

To share layers by reference

1. Click Home tab ➤ Display panel. In the Display Model list, click Open Display Model and open the display model.
2. In the Windows explorer, browse to your Repository Base Path, select the existing LAYER files, and drag them to the Display Manager.
3. Click Save Display Model As.
4. In the properties dialog box, notice the path of the existing layers, and the Ref. option. When you drag an existing layer into the Display Manager, and save the display model, the layer will not be saved in a new location. The layer will be referenced in the display model file, and the path to the original layer will be stored.
5. Specify the properties, and click OK. You are prompted, that the display model contains external layers, and that any modifications you have made on these files will be overwritten.

IMPORTANT If you use references layers, be aware that if you modify the stylization, the modifications will apply to all display models that use this layer.

Creating a display model from a drawing template

You can save existing drawing templates (**.dwt*) that have been used with Topobase 2008 as display models.

To save a drawing template as display model

1. Start Topobase Client, and open the workspace.
2. Open the drawing template (**.dwt*), such as TB2008_LM_102.dwt.
3. Click Home tab ➤ Display panel ➤ Save Display Model As.

4 In the New Display Model dialog box, enter a name. Click OK, and save the display model in a separate folder, such as Land102.

5 Generate Graphic.

NOTE The label feature classes <name>_TBL are loaded as feature layers, so you can reuse existing stylization. Optionally you can add the label feature classes as annotation layers. See Styling Annotation Layers (page 398).

Using Display Model Samples

Autodesk Topobase Client provides sample display models for the demo data sets, and for the following extensions.

- COGO
- Dimension
- Plot

Use the samples to modify the stylization as needed.

To use the sample display models

1 Copy the Display Model files <topobase_client>\Template\Extensions\<name of the extension>\DisplayStyle* to your display model repository.

2 Add the layer files you need to your display models, and modify stylization.

3 Save your display model.

Overview of the Display Manager

Autodesk Topobase uses the AutoCAD Map 3D Display Manager to create, edit, and display styles for feature source data in your map. You can use the standard stylization functionality available in AutoCAD Map 3D, or you can use Enhanced Stylization available only in Topobase. Both stylization procedures are available through the Display Manager Style command.

NOTE You must enable Enhanced Stylization through a special data connection. For more information, see Using Enhanced Stylization (page 400).

In addition to styling feature source data, you can change the display order and specify different styles for different views based on scale. You can also create themes for feature layers, to identify different types of data.

For example, you can highlight all the roads in a parcel that need to be paved or identify all the lots within 100 meters of a proposed highway.

You can also use themes to change colors, line types, symbols, text, or other properties that can help you present map information and tell a story. Themes can help make information more visible, illustrate how data is distributed, make data easier to analyze and interpret, and reveal patterns.

NOTE The styles you create with the Display Manager can be reused in Topobase Web by publishing the map containing those styles to Autodesk MapGuide first. See the MapGuide Studio Help for more information.

For detailed information about using the Display Manager to create and apply the styles, see the following sections in the AutoCAD Map 3D help:

- *Styling Features*
- *Creating Themes*

For additional Topobase specific information, read the following sections:

- Using Enhanced Stylization (page 400)
- Styling Label Features (page 393)
- Topobase Line Styles (page 393)
- Creating Topobase Calculations (page 408)

Using the Display Manager with Autodesk Topobase

Before using the Display Manager to style data in Autodesk Topobase, and to create feature layers, you need to connect to the Topobase provider.

Bringing In Features From Topobase

You can create a feature layer in Display Manager for each feature class that is stored in Topobase. Then, you style the feature layers.

To bring in features from Topobase

1. Start Topobase Client.
2. In Display Manager, click Data ➤ Connect To Data.
3. In the Data Connect window, select Add Topobase Connection in the Data Connections By Provider list.
4. Under Connection Name, type a name for this connection. You can give the connection any name you like.
5. Under Service Name, enter the service name for the Oracle data store. For example, ORCL.
6. Under Topobase Main Or System User Name, enter the name for the Topobase Main or System User. Also, enter the password. The default for the Topobase System User is TBSYS.
7. Under Document, select the document name in the list.
8. Click Connect.
9. In the Add Data To Map section, select the feature classes you want to add.
10. Do either of the following:
    - Click Add To Map to use standard AutoCAD Map 3D stylization.
    - Click Add To Map Using Enhanced Styles to use enhanced stylization. For more information on using enhanced stylization, see Using Enhanced Stylization (page 400).
11. Close the Data Connect window.

Then, style and edit the feature layers, and save the display model.

Optionally, save the settings as LAYER files, and create the display model from LAYER files. LAYER files can easily be modified using a text editor.

NOTE If you use the Oracle provider (Add Oracle Connection), you cannot use TBSYS to log in, you must log in with DBA rights.

See also:

- Styling Features (page 391)

Creating LAYER Files

After you have set up the styles and feature source information, you can optionally save this information as LAYER files.

Saving Styled Feature Layers

After you have styled feature layers, you can save the connection and styling information to LAYER files that you can share with other users, or add to any display model. A LAYER file contains connection and style information only. It contains no feature data.

To save LAYER files

1. In the Display Manager task pane, right-click the layer containing the information and styles you want to save and select Save Layer.

   In the Save Layer dialog box, under Files Of Type, your new file is given a (**.layer*) extension and a default file name from the name of the layer you selected.

2. You can rename the file or not. Then, save it to a convenient location on your system.

NOTE Autodesk Topobase provides sample layer files for the demo data sets. See also Topobase Installation and Configuration Guide.

When you select a display model, Autodesk Topobase adds the source files to the AutoCAD Map Explorer, creates the connection, adds the feature layer to the Display Manager (page 521). Click Generate Graphic to style the layers correctly.

To add a LAYER file to a display model

1. Navigate to where you saved your LAYER files. For example, for the Water demo data set, select the LAYER files in the folder *<topobase_client>\Template\Modules\Water\DisplayStyle* .
2. Select the files you want to include in the template, and drag them to the Display Manager task pane.
3. In Display Manager, click Data ➤ Connect To Data.
4. In the Data Connect window, under Data Connections By Provider, select the Topobase connection the feature layers belong to.

5 Check the connection parameters. After you have entered the Topobase Main User Password, under Document, select the document (datastore) containing the feature classes, whose LAYER files you just created. For example, select the Water demo data set TB2010_WA_102.

6 Click Connect and close the Data Connect window.

7 In the Display panel, click Display Model icon ➤ Save As.

Styling Features

You can specify properties that control how features appear on a map. For example, you can specify the scale ranges at which a feature is visible, set line color, and add labels.

Styling Point Features

Read the following AutoCAD Map 3D help topic:

■ *Styling Point Features*

NOTE Autodesk Topobase provides sample prototype drawings with predefined symbols for the demo data sets. See also Topobase Installation and Configuration Guide.

To use symbols of a prototype drawing

1 In the Display Manager, in the Style Point dialog box, click .

2 In the Select A Symbol dialog box, click .

3 In the Open Block Symbol File dialog box, select the prototype drawing (**.dwg*). For example, *<topobase_client>\Template\Modules\Water\DisplayStyle MAP2008\Prototype_WA.dwg*.

4 Click Open. Select a symbol.

Styling Line Features

Specify the thickness, color, and pattern of polyline features. Style a single line or build a composite line with several components and then style each component. For example, to illustrate a highway, create a thick black line and add a thinner, dashed, yellow line.

Read the following AutoCAD Map 3D help topic:

- *Styling Line Features*

NOTE Autodesk Topobase provides additional line styles. See Topobase Line Styles (page 393).

Styling Area Features

Specify the fill style and color, background color, edge style and color, and line thickness used to draw area (polygon) features.

Read the following AutoCAD Map 3D help topic:

- *Styling Area Features*

Adding Labels to Features

Add labels to features on feature layers. The label is placed near the line, point symbol, or polygon.

Read the following AutoCAD Map 3D help topics:

- *Adding Labels to Features*
- *Allowing Labels to Obscure Points*
- *Displaying Fixed Labels at Point Locations*

IMPORTANT In addition to the method of Adding Labels to Features, which is provided by the Display Manager, Autodesk Topobase provides another approach of labeling. Using Topobase you can create label features that are stored in the database. You use label features to create complex labels with a precise positioning.

See also:

- Styling Label Features (page 393)

Topobase Line Styles

For line stylization you can use the AutoCAD Map default line styles. Additionally, Autodesk Topobase provides Topobase line styles that are stored in the <topobase_client>\linx folder in style definitions files (*.*linx*).

You can edit the style definition files with any XML editor. For example, you can use a files as template for your own enhancements. Also, you can create additional style definition files.

A (*.*linx*) file contains two parts. The first part stores decorations and the second part stores the current line style definitions referencing those decorations. To draw filled symbols, you can use pre-defined symbols for filled circles and squares that are defined in the style definition file.

You can use decorations, and definitions that use those decorations. AutoCAD Map 3D provides four default decorations. You can add new decorations following a simple MoveTo/LineTo pattern.

For a complete format description, see the schema definition file *<topobase_client>\linx\linestyles.xsd*.

NOTE Rendering time significantly increases with more pixel runs used in a line style definition.

Styling Label Features

You use Topobase label features to create inscription with a precise and fix positioning relative to the features. Also, label feature text can be queried using complex SQL select statements and therefore display information of multiple feature attributes.

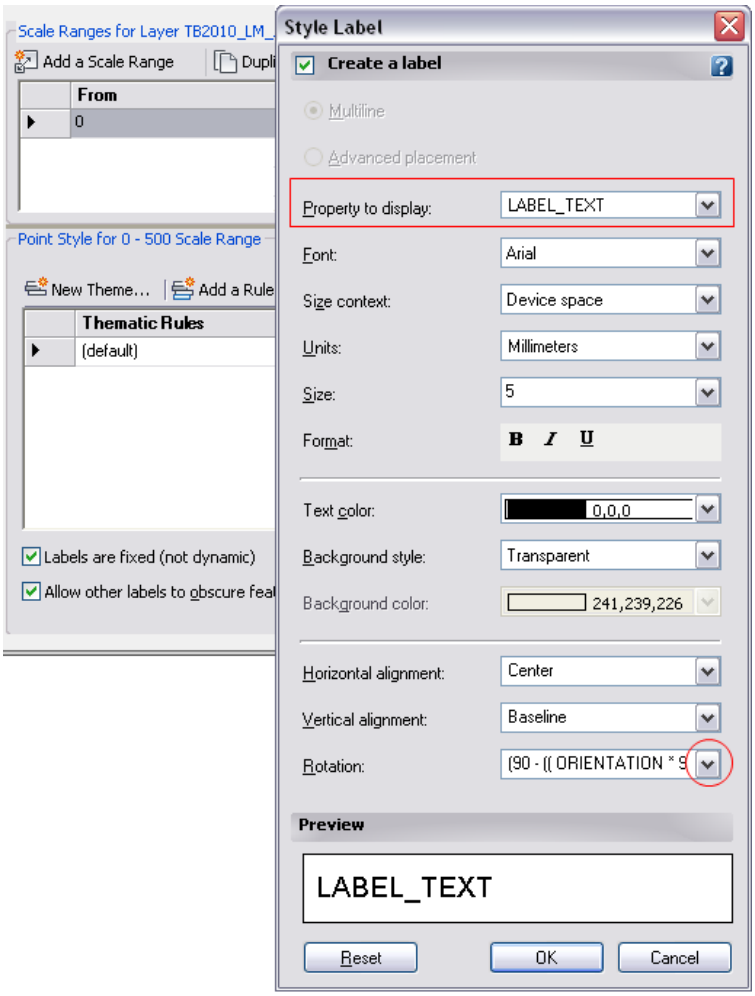
Label features are stored in the database in a special feature class type. The label text is stored in the attribute LABEL_TEXT. In Display Manager, label features are represented using separate feature layers.

Optionally label features can be represented in annotation layers. See Styling Annotation Layers (page 398).

IMPORTANT When styling label features in the Style Editor, you must enable the following options:

- **Allow Other Labels To Obscure Feature Symbols On This Layer**—Select this option to specify that labels on other layers be allowed to obscure points on the selected layer.
- **Labels Are Fixed (Not Dynamic)**—Select this option to specify that the labels are always displayed according to the draw order.

NOTE By default, a label feature is displayed using a symbol. Style the label feature as follows:



Styling a label feature

To style a label feature

1. In Display Manager, select a feature layer containing the label feature class, such as LM_STREET_TBL. By default, a label feature class is named <name>_TBL.
2. Click the Style And Theme Layers icon.

3 In the Style Editor, select Allow Other Labels To Obscure Feature Symbols On This Layer and Labels Are Fixed (Not Dynamic).

   These options make sure that the labels are displayed exactly at the position that is stored in the database.

4 In the Style Editor, under Scale Ranges, select a scale range.

5 Under Point Style For Scale Range, click the box under Style.

6 In the Style Point dialog box, select Style A Point Symbol and specify a symbol. Click OK.

7 Under Point Style For Scale Range, click the box under Feature Label.

8 In the Style Label dialog box, select the Create A Label check box.

9 For Property To Display, select LABEL_TEXT.

10 Use the Horizontal Alignment and Vertical Alignment lists to specify the position of the text origin relative to the feature.

   You can select the alignment values that have been specified in the label definition. In the list, select Expression. In the Text Expression dialog box, under Property, select HORIZONTAL_ALIGNEMENT or VERTICAL_ALGINEMENT.

11 For Rotation you use the value that is stored in the database. Select Expression, depending on the document unit settings. See also Displaying Label Features with Prefix and Suffix (page 398)

IMPORTANT You must style a point symbol for the label feature, if you want to edit the label in the drawing, such as changing the orientation or the position.

If you do not style a point symbol for the label feature, you cannot select the label in the drawing.

Read the following AutoCAD Map 3 D help topics:

- Adding Labels to Features
- Allowing Labels to Obscure Points
- Displaying Fixed Labels at Point Locations

See also:

- Data Model: Labels (page 187)

■ Displaying Label Features with Prefix and Suffix (page 398)

Displaying Label Features with Orientation

You can use the orientation that is stored in the database to specify the rotation of Topobase label features. The orientation is stored in the label feature class attribute ORIENTATION. Note, that the Display Manager interprets the ORIENTATION as degrees counterclockwise, regardless of the document units and Autodesk Map drawing settings. Also, the document base angle is North, and the Display Manager base angle is East. To display properly, you must convert the orientation from document units to degrees counterclockwise

To convert the label orientation

1. In Display Manager, select a feature layer containing the label feature class. Click the Style And Theme Layers icon.
2. In the Style Editor, specify an expression for the rotation. In the Style Label dialog box, select Rotation = Expression. Use the expressions exactly as shown in the following table:

Document Unit	Rotation Unit	Expression
Degree	counterclockwise	ORIENTATION + 90
Gon	counterclockwise	((ORIENTATION * 9/10) + 90)
Radian	counterclockwise	((ORIENTATION * 1800000/31415) + 90)
Degree	clockwise	(90 - ORIENTATION)
Gon	clockwise	(90 - (ORIENTATION * 9/10))
Radian	clockwise	(90 - (ORIENTATION * 1800000/31415))

Styling Annotation Layers

You can either load a label feature class as feature layer or as annotation layer.

- **Annotation layer**—Under Data ➤ Connect To Data, select a feature class that has the name extension _TBL. In the Display manager, annotation layers are marked by a special icon .
- **Feature layer**—On the Home tab ➤ Display panel, open the default display model. All label feature classes are loaded as feature layers.

To style an annotation

1. In Display Manager, select the annotation layer. Click the Style And Theme Layers icon.
2. In the Style Editor, under Symbolization Style For Scale Range, click the box under Style.
3. In the Style Text Layer dialog box, style the annotation.

Displaying Label Features with Prefix and Suffix

You can add a static prefix/suffix to your label definition, such as a unit or a caption. The prefix and suffix values are stored in the label feature class in the attributes PRE and SUF. To display the labels including the prefix/suffix, you must use an expression.

To display a label with prefix and suffix.

1. In Display Manager, select a feature layer containing the label feature class. Click the Style And Theme Layers icon.
2. In the Style Editor, under Point Style For Scale Range, click the box under Feature Label.
3. In the Style Label dialog box, select the Create A Label check box.
4. For Property To Display, select Expression.
5. In the Text Expression dialog box, click the Advanced icon. For example, to display both prefix and suffix, enter the following expression:

```
concat("PRE", concat("LABEL_TEXT", "SUF"))
```

See also:

- Label Properties: Label Information (page 190)

Map Drawing Units

When generating a graphic, Topobase sets the following drawing units according to the document unit settings:

- **Angle Type**—For example degrees or gons.
- **Rotation**—For example clockwise or counterclockwise.
- **Base Angle**—In Topobase documents the direction is North.

The drawing units apply to all construction commands.

If a workspace contains more than one document with different unit settings, you are notified that the label orientation unit can not be determined clearly when you digitize an orientation.

To view the map unit settings

1. On the Map command line, enter Units.
2. Click Direction for the Base Angle direction.

To view the document unit settings

1. Start Topobase Administrator.
2. In the Administrator explorer, select the document.
3. Click Data Model. In the data model explorer, select the label feature class.

   The ORIENTATION unit is displayed in the properties pane on the right.

NOTE All Topobase documents use the Base Angle direction North. You cannot modify this value.

Using Enhanced Stylization

When you use Data Connect within Topobase Client to add data to your map, you can enable enhanced stylization functionality that is not available in AutoCAD Map 3D. This stylization enables you to build a collection of styles laid one on top of the other for points, lines, polygons, and text and symbol labels.

Enabling Enhanced Stylization

To enable enhanced stylization you must select it when you connect to data.

To enable enhanced stylization

1. In Display Manager, click Data ➤ Connect To Data.
2. Configure the data connection.

   For more information, see Bringing In Features From Topobase (page 388).
3. When you are ready to add the data to the map, click Add To Map Using Enhanced Styles instead of Add To Map.

Enhanced Point Stylization

Add a point or multigeometry layer to your map using Add To Map Using Enhanced Styles in the Data Connect dialog box. For more information, see Enabling Enhanced Stylization (page 400).

To style points using enhanced stylization

1. In Display Manager, right-click a feature layer that contains points.
2. Click Edit Style.
3. In the Style Editor, under Scale Ranges, select the scale range to style.
4. In the Symbolization Style area for the selected scale range, click the box under Style.
5. In the Style And Label Editor dialog box, click the Style tab.

   If you are working with a multigeometry layer, for Geometry Type, click Point, Line, or Polygon to specify the type of geometry you want to style on this layer.

6 For Size Context, specify the type of units:

- Select Device Space to specify symbol widths and heights in screen units. Available units are Points, Inches, Millimeters, or Centimeters.
- Select Map Space to specify symbol widths and heights in Mapping Coordinate System (MCS) units. Available units are Inches, Feet, Yards, Miles, Millimeters, Centimeters, Meters, and Kilometers.

7 For Units, select the type of units to use.

8 Under Symbol And Style, click Add to add a new style component.

9 Select each style component in the Symbol And Style list and style as desired.

Styles are drawn from the bottom of the list to the top so use Move Down and Move Up to control the draw order.

Click Load to open a symbol from a *.dwg* file.

Use Width and Height to adjust the size of the symbol. Click Maintain Aspect Ratio to keep the symbol intact as you modify either of these settings.

Use X Offset and Y Offset to specify the offset of the symbol from the location of the point feature.

For any style setting, click Edit Expression to build styles using expressions based on feature properties. For more information about expressions, see Building If, Lookup, and Range Expressions (page 406).

10 When you are finished styling points, click OK.

To add text labels to this layer, see Feature Text Labels (page 404). To add symbol labels to this layer, see Feature Symbol Labels (page 405).

Enhanced Line Stylization

Add a line or multigeometry layer to your map using Add To Map Using Enhanced Styles in the Data Connect dialog box. For more information, see Enabling Enhanced Stylization (page 400).

To style lines using enhanced stylization

1 In Display Manager, right-click a feature layer that contains lines.

2 Click Edit Style.

3 In the Style Editor, under Scale Ranges, select the scale range to style.

4 In the Symbolization Style area for the selected scale range, click the box under Style.

5 In the Style And Label Editor dialog box, click the Style tab.

If you are working with a multigeometry layer, for Geometry Type, click Point, Line, or Polygon to specify the type of geometry you want to style on this layer.

6 For Size Context, specify the type of units:

- Select Device Space to specify symbol widths and heights in screen units. Available units are Points, Inches, Millimeters, or Centimeters.
- Select Map Space to specify symbol widths and heights in Mapping Coordinate System (MCS) units. Available units are Inches, Feet, Yards, Miles, Millimeters, Centimeters, Meters, and Kilometers.

7 For Units, select the type of units to use.

8 Under Symbol And Style, click Add to add a new style component.

9 Select each style component in the Symbol And Style list and style as desired.

Styles are drawn from the bottom of the list to the top so use Move Down and Move Up to control the draw order.

Click Load to open a line style from a *.lin* file.

Use Width and Height to adjust the size of line pattern elements. Click Maintain Aspect Ratio to keep the pattern intact as you modify either of these settings.

For any style setting, click Edit Expression to build styles using expressions based on feature properties. For more information about expressions, see Building If, Lookup, and Range Expressions (page 406).

10 When you are finished styling lines, click OK.

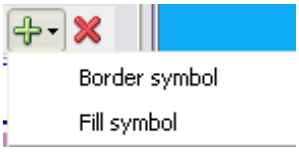
To add text labels to this layer, see Feature Text Labels (page 404). To add symbol labels to this layer, see Feature Symbol Labels (page 405).

Enhanced Polygon Stylization

Add a polygon or multigeometry layer to your map using Add To Map Using Enhanced Styles in the Data Connect dialog box. For more information, see Enabling Enhanced Stylization (page 400).

To style polygons using enhanced stylization

1. In Display Manager, right-click a feature layer that contains polygons.
2. Click Edit Style.
3. In the Style Editor, under Scale Ranges, select the scale range to style.
4. In the Symbolization Style area for the selected scale range, click the box under Style.
5. In the Style And Label Editor dialog box, click the Style tab.

   If you are working with a multigeometry layer, for Geometry Type, click Point, Line, or Polygon to specify the type of geometry you want to style on this layer.
6. For Size Context, specify the type of units:
   - Select Device Space to specify symbol widths and heights in screen units. Available units are Points, Inches, Millimeters, or Centimeters.
   - Select Map Space to specify symbol widths and heights in Mapping Coordinate System (MCS) units. Available units are Inches, Feet, Yards, Miles, Millimeters, Centimeters, Meters, and Kilometers.
7. For Units, select the type of units to use.
8. Under Symbol And Style, click Add. Then click Border Symbol or Fill Symbol to add a new style component. 

9. Select each style component in the Symbol And Style list and style as desired.

   Styles are drawn from the bottom of the list to the top so use Move Down and Move Up to control the draw order.

For polygon border styling, click Load to open a border line style from a *.lin* file.

For polygon fill styling, click Load to open a fill pattern from a *.pat* file.

Use Width and Height to adjust the size of line or fill pattern elements. Click Maintain Aspect Ratio to keep the pattern intact as you modify either of these settings.

For any style setting, click Edit Expression to build styles using expressions based on feature properties. For more information about expressions, see Building If, Lookup, and Range Expressions (page 406).

10 When you are finished styling polygons, click OK.

To add text labels to this layer, see Feature Text Labels (page 404). To add symbol labels to this layer, see Feature Symbol Labels (page 405).

Feature Text Labels

Add a point, line, polygon, or multigeometry layer to your map using Add To Map Using Enhanced Styles in the Data Connect dialog box. For more information, see Enabling Enhanced Stylization (page 400).

To style feature text labels using enhanced stylization

1 In Display Manager, right-click a feature layer that contains the features you want to label.

2 Click Edit Style.

3 In the Style Editor, under Scale Ranges, select the scale range to style.

4 In the Symbolization Style area for the selected scale range, click the box under Feature Label.

5 In the Style And Label Editor dialog box, click the Feature Text Label tab.

If you are working with a multigeometry layer, for Geometry Type, click Point, Line, or Polygon to specify the type of geometry you want to label on this layer.

6 For Size Context, specify the type of units:

- Select Device Space to specify symbol widths and heights in screen units. Available units are Points, Inches, Millimeters, or Centimeters.

- Select Map Space to specify symbol widths and heights in Mapping Coordinate System (MCS) units. Available units are Inches, Feet, Yards, Miles, Millimeters, Centimeters, Meters, and Kilometers.

7 Under Symbol And Style, click Add to add a new style component.

8 Select each style component in the Symbol And Style list and style as desired.

Styles are drawn from the bottom of the list to the top so use Move Down and Move Up to control the draw order.

For line features, Vertical Alignment controls the placement of the text relative to the line feature being labeled.

For any style setting, click Edit Expression to build styles using expressions based on feature properties. For more information about expressions, see Building If, Lookup, and Range Expressions (page 406).

9 When you are finished styling text labels, click OK.

Feature Symbol Labels

Add a point, line, polygon, or multigeometry layer to your map using Add To Map Using Enhanced Styles in the Data Connect dialog box. For more information, see Enabling Enhanced Stylization (page 400).

To style feature symbol labels using enhanced stylization

1 In Display Manager, right-click a feature layer that contains points.

2 Click Edit Style.

3 In the Style Editor, under Scale Ranges, select the scale range to style.

4 In the Symbolization Style area for the selected scale range, click the box under Feature Label.

5 In the Style And Label Editor dialog box, click the Feature Symbol Label tab.

If you are working with a multigeometry layer, for Geometry Type, click Point, Line, or Polygon to specify the type of geometry you want to label on this layer.

6 For Size Context, specify the type of units:

- Select Device Space to specify symbol widths and heights in screen units. Available units are Points, Inches, Millimeters, or Centimeters.
- Select Map Space to specify symbol widths and heights in Mapping Coordinate System (MCS) units. Available units are Inches, Feet, Yards, Miles, Millimeters, Centimeters, Meters, and Kilometers.

7 Under Symbol And Style, click Add to add a new style component.

8 Select each style component in the Symbol And Style list and style as desired.

Styles are drawn from the bottom of the list to the top so use Move Down and Move Up to control the draw order.

Click Load to open a symbol from a *.dwg* file.

Use Width and Height to adjust the size of the symbol. Click Maintain Aspect Ratio to keep the symbol intact as you modify either of these settings.

For any style setting, click Edit Expression to build styles using expressions based on feature properties. For more information about expressions, see Building If, Lookup, and Range Expressions (page 406).

9 When you are finished styling symbol labels, click OK.

Building If, Lookup, and Range Expressions

Topobase supports expressions as values for almost all style settings. Use the Expression Builder to style features using expressions based on feature attributes. In addition to standard expressions, Topobase provides three theming functions: If, Lookup, and Range. These functions define a theme for one display attribute, for example, line color or thickness. They work with multiple data types: string, real, integer, boolean, color, date-time.

NOTE Each parameter can be a constant or expression that evaluates to a string or numerical value.

If Expression

```
If('Condition', TrueValue, FalseValue)
```

The ‘Condition’ parameter is a string containing any expression that evaluates to a Boolean value (essentially, a filter). Typically, this might be a single comparison such as 'population > 1000', where ‘population’ is a feature property of the layer. ‘TrueValue’ and ‘FalseValue’ are expressions that are evaluated and returned when the condition is true or false, respectively. Both expressions must evaluate to the same data type (string or number).

This example could be used to rotate an upward-facing arrow symbol to indicate rising or falling populations:

```
If('pop2005 > pop2000', 0.0, 180.0)
```

Lookup Expression

```
Lookup(Expression, DefaultValue, Key1, Value1, … KeyN, ValueN)
```

This function implements a lookup table where each key is associated with a single value. ‘Expression’ is an expression that evaluates to a key (string or number). This key is compared to the remaining constant ‘Key#’ parameters to select the constant ‘Value#’ parameter that is to be returned. If the requested key is not found, ‘DefaultValue’ is returned. The ‘Expression’ and all ‘Key’ parameters must have the same data type, as must the ‘DefaultValue’ and all ‘Value#’ parameters. The number of key-value pairs is not fixed.

This example could be used to assign colors to parcels based on their zoning type:

```
Lookup(zoning, 0xff888888, 'Industrial', 0xffff0000, 'Commercial',
0xff00ff00, 'Residential', 0xff0000ff)
```

Range Expression

The range function compares the ‘Expression’ key to a set of ranges, where a key matches a range if MIN <= KEY < MAX:

```
Range(Expression, DefaultValue, Min1, Max1, Value1, … MinN, MaxN,
ValueN)
```

Ranges are specified by the ‘Min#’ and ‘Max#’ parameters, and each has an associated ‘Value#’ parameter. If the requested key does not match any of the provided ranges, ‘DefaultValue’ is returned. If ranges overlap, the first range

that matches the key will be selected. The 'Expression' and all 'Min' and `Max' parameters must have the same data type, as must the 'DefaultValue' and all 'Value#' parameters. The number of range-value sets is not fixed.

This example could be used to specify symbol size depending on city populations:

```
Range(population, 4.0, 100000, 1000000, 6.0, 1000000, 10000000, 8.0)
```

Creating Topobase Calculations

You use the FDO expression builder to define stylization that is based both on feature class attributes and on attributes of related features, or on any values that you calculate using attribute values.

Use the Expression Builder to create calculated properties that can be used for dynamic stylization. Topobase provides two text functions.

- **TB_SQL**—Use this text function to query attributes from a related feature class. The attributes can be used for thematic rules.
- **TB_VARIABLE**—Use this text function, if you want to define thematic rules that use user defined variables.

To create a calculated text property

1. In the Display Manager, select the feature class.
2. Right-click, and click Create Calculations.
3. See Topobase Text Function TB_SQL (page 409) and Topobase Text Function TB_VARIABLE (page 410).

To modify a calculated text property

1. In the Display Manager, select the feature class.
2. Right-click, and click Manage Calculations.
3. In the Manage Layer Data dialog box, select the calculated property, and click Edit.
4. Modify the calculation, and click OK.

Topobase Text Function TB_SQL

Use the text function TB_SQL to create a calculated text property that queries attributes of a related feature class. Then, you use the calculated property for stylization.

The following example can be reproduced with the demo data set. You use the TB_SQL text function to define a thematic rule for the stylization of the Landuse feature class. Using the calculated property, you can define the area stylization without using an Oracle view.

To create a calculated text property (TB_SQL)

1. Start Topobase Client, and open the workspace.
2. Select the display model, and click Generate Graphic.
3. In the Display Manager, select the feature class, such as LANDUSE_TSUR.
4. Right-click, and click Create A Calculation.
5. In the Create A Calculation dialog box, enter a name, such as LU_Type.
6. Click Text Function, and click TB_SQL.
7. Insert the SQL command in parentheses after the function. Make sure you use apostrophes to get a text string.

   ```
   'SELECT TO_CHAR(MAX(C.ID_TYPE)) FROM LANDUSE_TCEN m,LM_LANDUSE
   c
   WHERE C.FID=M.FID_CENTROID AND M.FID_TSUR=g.FID'
   ```

   The calculated property provides the landuse type that is stored in the centroid feature class LM_LANDUSE.
8. Click OK to create the calculated property.
9. In the Display Manager, select the feature class. Right-click, and click Zoom To Extends.
10. Right-click, and click Edit Style.
11. In the Style Editor, add a Thematic Rule.
12. In the Create/Modify Expression dialog box, click Properties, and under Text Properties select the calculated property, such as LU_Type.
13. Define the rule, such as *LU_TYPE = '27'*. This rule styles the landuse polygons that have the land use type LM_LANDUSE.ID_TYPE = 27 = Building Area.

14 Style the layer.

Defining the SQL command

The SQL command must correspond to the following rules.

- The statement must return one column.
- The statement must return a string. For numbers, use the function TO_CHAR.
- For the feature class, use the alias g.

IMPORTANT To test the SQL command, execute `SELECT g.FID, (SQL command) FROM <feature class name> g`. For example, use SQL Sheet to execute the command.

Topobase Text Function TB_VARIABLE

Use the text function TB_VARIABLE to create a calculated text property that provides a user defined variable for thematic stylization. The variables are stored in the system table TB_VARIABLE. For each personal login, you can define such variables.

For example, in TB_VARIABLE, you define a variable BUILDING whose value represents the FID. In Display Manager you use the variable to style the building. To highlight a single building in the map, you enter the FID of the building in TB_VARIABLE. If you want to highlight another building, you need not to modify the display model, but you can enter the new FID in TB_VARIABLE.

In the following example, you use the TB_VARIABLE text function to define a thematic rule for the stylization of a building.

To create a calculated text property (TB_VARIABLE)

1 Start Topobase Client, and open the workspace.

2 Select the display model, and click Generate Graphic.

3 In the Display Manager, select the feature class, such as LM_BUILDING.

4 Right-click, and click Create A Calculation.

5 In the Create A Calculation dialog box, enter a name, such as VARIABLE_BUILDING.

6 Click Text Function, and click TB_VARIABLE.

7 Insert the Key in parentheses after the function. Make sure you use apostrophes to get a text string.

```
('BUILDING')
```

8 Click OK to create the calculated property.

9 In the Display Manager, select the feature class. Right-click, and click Zoom To Extents.

The key values that are used in the calculated properties, and the user ID must be specified in TB_VARIABLE.

To add the key values in TB_VARIABLE

1 In the Document explorer, under System Tables, right-click TB_VARIABLE, and click Show Form.

2 On the form toolbar, click New Record.

3 Enter the key values as shown in the following table.

TB_VARIABLE attribute	**Description**
USER_ID	Specifies the user. Enter the TB_USER.ID. In Topobase Client, the variable will only apply, if this user is connected to the database. For other users, the variable will not take any effect.
KEY	Specifies the key that is used in the calculated property, such as BUILDING.
VALUE	Specifies a value for the key. For example enter the FID of the specific building you want to represent using a special style.

To style the feature layer

1 In the Display Manager, select the feature class. Right-click, and click Edit Style.

2 In the Style Editor, add a Thematic Rule.

3 In the Create/Modify Expression dialog box, click Properties, and under Text Properties select the calculated property, such as VARIABLE_BUILDING.

4 Insert an expression, such as VARIABLE_BUILDING = TOSTRING(FID).

5 Click OK.

When you use a data connection without Topobase, no Topobase user is set. However you can use the variables, if TB_VARIABLE.USER_ID = Null.

Styling Job-Enabled Features

When you work with Topobase jobs that are in open or pending job state, you can highlight modified or newly created features, so they can be distinguished from life features. Also, you can define a special style for the deleted features.

For job enabled documents, Topobase provides an option that enhances the Default Display Model, so you can easily style the features that have been created, modified, or deleted in a job.

To create a Display Model that tracks feature modification and deletion

The document must be job enabled.

1 Start Topobase Client, and open the workspace.

2 Click Home tab ➤ Display panel.

3 From the Display Model list, select Open Default Display Model.

4 In the Generate Graphic - Default Display Model dialog box, select Enable Deleted Features To Be Displayed.

5 In the Display Manager, style the layers, and save the Display Model.

The Display Model provides a Topobase text function TB_SQL for TB_JOB_OPERATION _ID. You use this text function to define Thematic Rules for the features that have been created, modified, or deleted in the current job.

To style features that are modified or created in a job

1. In Display Manager, select the feature layer, such as Building (LM_BUILDING).
2. Click Style.
3. In the Style Editor, under Polygon Style For Scale Range, click Add A Rule.
4. Click the box under Thematic Rules.
5. In the Create/Modify Expressions dialog box, click Property.
6. In the Property list, select JOB_OPERATION_ID and enter the following condition:
   - For new features: *JOB_OPERATION_ID = '1'*.
   - For modified features: *JOB_OPERATION_ID = '2'*.
   - For deleted features: *JOB_OPERATION_ID = '3'*.
7. Click OK.
8. Click the box under Style. In the Style Point dialog box, specify the style.

For more information about the Topobase text function TB_SQL see the Topobase Administrator Guide, section Creating Topobase Calculations (page 408).

Variant: Style features using Oracle views

Requirements: For each job-enabled feature class that you want to style, you must create an Oracle view that includes the JOB_OPERATION_ID attribute. Job-enabled feature classes have a JOB_VERSION attribute, which is related to the TB_JOB_VERSION table.

- Create an Oracle view, such as the building view LM_V_BUILDING in the land management demo data set (`CREATE VIEW LM_V_BUILDING AS select a.FID, a.AREA, a.GEOM, a.IDENTIFICATOR, a.ID_QUALITY, a.ID_TYPE, a.JOB_VERSION, b.JOB_OPERATION_ID FROM LM_BUILDING a, TB_JOB_VERSION b WHERE a.JOB_VERSION = b.JOB_VERSION`).
- Register the view as feature class. See Creating a Feature Class From a View (page 164).

To style features that are modified or created in a job

1. In Display Manager, select the feature layer of the view, such as LM_V_BUILDING.
2. Click Style.
3. In the Style Editor, under Polygon Style For Scale Range, click Add A Rule.
4. Click the box under Thematic Rules.
5. In the Filter dialog box, under Property Or Expression, select Property.
6. In the Property list, select JOB_OPERATION_ID and enter the following condition:

   For new features: JOB_OPERATION_ID = 1.

   For modified features: JOB_OPERATION_ID = 2.

   For deleted features: JOB_OPERATION_ID = 3.
7. Click OK.
8. Click the box under Style. In the Style Polygon dialog box, specify the style.

NOTE You use the layer of the view for stylization, such as LM_V_BUILDING. Do not use the feature class layer, such as LM_BUILDING.

Settings for Topobase Web

In Topobase Web, you cannot use the Display Deleted Features option.

To style job enabled features in Topobase Web

- Edit the system table TB_SETTINGS. Add a row with the following values.

  ```
  ITEMKEY = SHOW_DELETED_FEATURES_ENABLED
  ```

  ```
  ITEMVALUE = 1
  ```
- Create views over the feature classes that include the JOB_OPERATION_ID from TB_JOB_VERSION joined by JOB_VERSION, and style the views using the value JOB_OPERATION_ID. See Variant: Style features using Oracle views (page 413)

See also:

- System Tables TB_JOB_* (page 81)

- Jobs Introduction (page 295)

Plot Reference 8

Understanding Plot

See Topobase Client User Guide, Understanding Plotting.

Getting Plot Started

Before you start working with plots in a document, you must configure plots using Topobase Administrator, and create plot templates, using Topobase Client.

- Plot enable the document. See Configuring Plots (page 417).
- Use Topobase Client to create plot templates. See Topobase Client User Guide, Creating Plot Templates.
- Use Display Models to style the plot features such as north arrow, legend, scale bar. See also Plot Display Models (page 419).
- In Topobase Client, create the plots, and print the plots. See Working With Plots.

Configuring Plots

This section describes how to plot enable a document. To plot enable a document, you add the Plot Extension.

To plot enable a document

1 Start Topobase Administrator.

2 Open the workspace, and select the document.

3 In the administrator explorer, select the document.

4 In the right pane, click Extensions.

5 Select Plot Extension.

6 Click Save.

7 In the Update Modules And Data Models dialog box, click Update.

The system adds the plot feature classes and system tables to the document. See also Plot Extension (page 421).

To edit and view the plot feature classes in Topobase Client, use the Plot Library. See Understanding Plotting.

Plot Settings

When a document is plot enabled, you can specify plot settings in the administrator explorer.

To specify Plot Settings

1 Open the workspace and select the document.

2 Click Document menu ➤ Plot.

3 On the right pane, specify the settings as shown in the following table.

In Topobase Client, the settings can be modified in the Create New Plot dialog box, under Advanced.

Plot Settings area	Description
Default Scale	Specifies the default scale.
Insertion Point	Specifies the real world coordinates within the Topobase document where the initial plot features will be created. When you create a plot, or a plot template, these coordinates are used to display the page canvas. If these coordinates do not lie within the document's spatial extents, you cannot create any plots. However, in Topobase Client, in the Create New Plot dialog box, you can modify the insertion point.

	IMPORTANT Enter coordinates that are valid for the spatial settings of the document, such as the spatial reference system, and the extents. See also Data Model Administrator: Creating a Database (page 144) **NOTE** The current insertion point of the map place holders will be specified during capturing. For example, for the demo data set, enter (600000, 200000).
Island Map Settings area	See also Topobase Client User Guide, Capturing Islands.
Define The Feature Class And The Buffers Used For Island Maps	Displays all polygon feature classes of the selected document. Select one or more feature classes that can be used to specify the island map perimeter.
Buffer 1	Specifies the default distance for an additional buffer. In Topobase Client, you use the buffers to expand the island out from the selected feature. In the display model, use the drawing order to specify which features shall be displayed in the buffer.
Buffer 2	Specifies the default distance for a second additional buffer to the island.

Plot Display Models

This section describes which feature layers have to be loaded and stylized to display the plot components.

Autodesk provides a plot display model in the folder <topobase_client>\Template\Extensions\Plot\DisplayStyle. To use the plot display model, create a new folder in your Display Model Repository Base Path, such as Plot Default, and copy the files to this location.

Style the following plot features. The features are shown in the order in which they appear in the Display Manager draw order view. For example, the Plot Label is drawn in front of the layers lower down.

Feature class	Description

Plot Label	PLT_PLOT_TBL. Label feature class that stores labels. Define thematic rules that use the LABEL_DEF_ID to style different labels for different types of labels, such as plot name, or plot scale.
North Arrow	PLT_PLOT_NA. Point feature class that stores the north arrow. Style the point using a north arrow symbol.
Scalebar	PLT_PLOT_SCALEBAR. Point feature class.
Legend	PLT_PLOT_LEGEND. Point feature class. Style the point using a block that represents the legend.
Image	PLT_PLOT_IMAGE. Point feature class.
Plot Decoration Label	PLT_PLOT_DECORATION_TBL. Label feature class.
Decoration	PLT_PLOT_DECORATION. Collection feature class.
Grid / Cross Hair Extension Lines Frame	PLT_PLOT_GRID_FRAME. Line feature class.
Grid / Cross Hair Extension Line	PLT_PLOT_GRID_EXTLINE. Line feature class.
Map Placeholder	PLT_PLOT_MAP. Polygon feature class that represents the map placeholder. The plot function requires that you style the map placeholder depending on the DRAWORDER. For DRAWORDER=0 do not apply a fill pattern. For DRAWORDER>0 apply a fill pattern. See the sample display model.
Plot Cover	PLT_PLOT_COVER. Collection feature class that represents the canvas.
Crosshair	PLT_PLOT_CROSSHAIR. Collection feature class.
Grid Line	PLT_PLOT_GRID. Line feature class.

Use the application options to specify a Repository Base Path. See also Setting Generate Graphic Application Options (page 22).

See also:

- Plot Extension (page 421)

Plot Extension

This section describes the plot feature classes and system tables. These tables are added to the document, when you add the Plot Extension. See also Document Settings (page 7).

To view the plot feature classes

1. Start Topobase Administrator.
2. Open the workspace, and select the document.
3. Click Document menu ➤ Data Model.
4. In the data model explorer, expand the topic Plot.

The Plot Extension consists of two topics, as shown in the following table. Plot table names have the prefix PLT_<table name>.

Feature classes and topics	Description
Plot	Topic that groups all plot feature classes.
Plot	PLT_PLOT. Point feature class that stores the plot origin, and plot attributes, such as the name.
Plot Cover	PLT_PLOT_COVER. Collection feature class.
Map Placeholder	PLT_PLOT_MAP. Polygon feature class. See also Plot Map Placeholder (page 422).
Crosshair	PLT_PLOT_CROSSHAIR. Collection feature class.
Grid Line	PLT_PLOT_GRID. Line feature class.
Grid / Cross Hair Extension Line	PLT_PLOT_GRID_EXTLINE. Line feature class.

Grid / Cross Hair Extension Lines Frame	PLT_PLOT_GRID_FRAME. Line feature class.
Island Plan Cover	PLT_PLOT_ISLAND_COVER. Polygon feature class.
Predefined Plot Scale	PLT_PLOT_SCALE. Attribute feature class. Stores the scales that can be selected when creating a plot.
Plot Label	PLT_PLOT_TBL. Label feature class.
Plot Decoration	Sub topic that groups geometry feature classes and label feature classes that store the plot components.
Decoration	PLT_PLOT_DECORATION. Collection feature class.
Image	PLT_PLOT_IMAGE. Point feature class.
Legend	PLT_PLOT_LEGEND. Point feature class.
North Arrow	PLT_PLOT_NA. Point feature class.
Scalebar	PLT_PLOT_SCALEBAR. Point feature class.

Plot Map Placeholder

The plot feature class PLT_PLOT_MAP stores the map placeholder properties as shown in the following table. Some attributes are set automatically, when you create the plot template, and add the map placeholder. To edit the attributes manually, open the Map Placeholder feature class form.

PLT_PLOT_MAP attribute name	Caption	Description
ID_GRIDCROSS_TYPE	Display	Specifies the type of grid that is added to the map. Can be Crosshair, Grid, or No Grid Or Crosshair. Relation to domain table PLT_PLOT_GRIDCROSS_TBD.
GRIDCROSS_EASTING_INTERVALL GRIDCROSS_EASTING_INTERVALL	Intervals	Specifies the distance between the grid crosses.

PLT_PLOT_MAP attribute name	Caption	Description
GRIDCROSS_SHOWEXTLINES	Grid Extension Lines	Creates extension lines for the grid, or the cross hairs.
ID_GRIDCROSS_LABELTYPE	Extension Lines Labels	Specifies the label type. Can be No Labels; Align Along The Grid Extension Line; Align Along The Map Frame. References the label definition.
GRIDCROSS_SUPPRESS_LABELS	Suppress Labels Tolerance	Specifies a tolerance that suppresses the creation of grid extension lines. For example, if you create a rotated plot, the labels for easting and northing might overlap. The tolerance specifies a radius around an existing label where no other label can be created. Enter a distance. ■ NULL, 0, or a value <0. Each grid extension line label will be created. This may cause overlapping labels. ■ Value > 0. This value specifies the paper space value. If a grid extension line label is located within this distance, another label would not be created, to prevent overlapping text.
GRIDCROSS_FRAME_INSIDE	Frame Inside	Specifies whether the frame for the grid extension line is created inside or outside the map placeholder. Default is No, so the frame does not overlap any GIS data.

Report Designer Reference 9

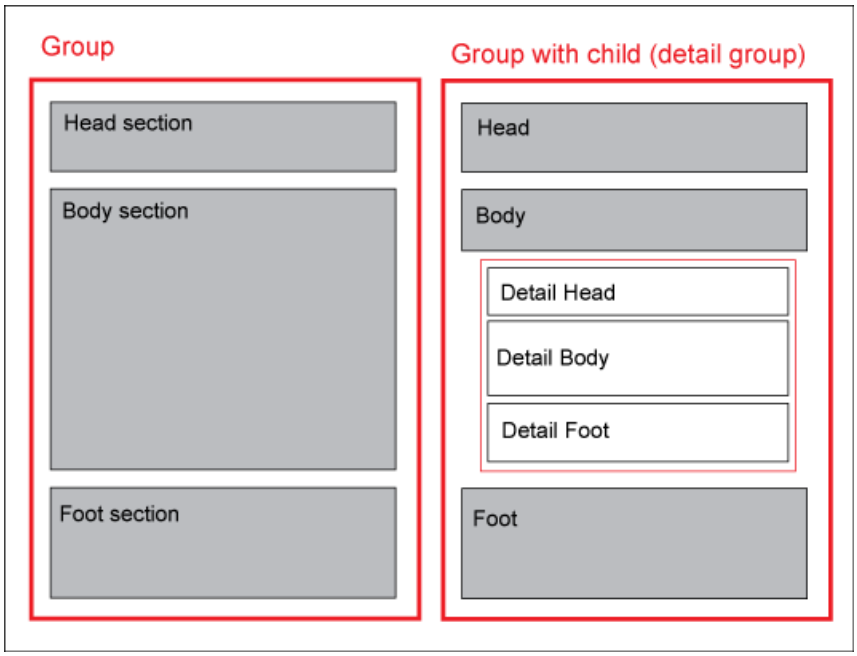
Report Designer Introduction

Using Topobase report designer, you can design your own reports and assign them to any Topobase feature class form. The report definitions are stored in the document (page 521) (TB_RPT_*tables).

Report Designer: Layout Components

A Topobase report layout consists of different parts:

Group—A group consists of three sections: head, body and foot.



Report Designer: Report layout showing a group with head, body, and foot sections and a group with a child group

Child Group—You can insert a child group with detail head, body and foot sections. Child groups are added between the body and foot section.

Zones—Optionally, each section can be divided into different zones. A default section has only one zone. Note that there is a difference between zone and section. You can define zones that can for example be suppressed under certain conditions.

Pages—You can define different pages for a report, for example, to provide a title page (report head) that should look different.

Controls—You can use a number of control elements to design the content for each layout section.

You use the sections to design report layout as shown in the following table:

Section	Description
Page Group	
Page Head	The content of this section is repeated on top of every page.

Page Foot	The content of this section is repeated at the bottom of every page.
Detail Group	
Detail Head	The content of this section is printed only once, above the data in the preceding body section.
Detail Body	The content of this section is repeated for every record of the SQL statement that is set in the Group properties.
Detail Foot	The content of this section is printed after the records of the body section.

See also:

- Report Designer Control Properties (page 435)

Getting Report Designer Started

You start the report designer in the Topobase Administrator.

To start the report designer

1. Start Topobase Administrator and open a workspace.
2. Select the document (page 521).
3. On the Administrator toolbar, click the Report Designer icon.

The handling of the designer is similar to the Topobase form designer.

See also:

- Form Designer Introduction (page 309)
- Creating a Report (page 121)

Report Designer Menu Bar

Some commands of the menu are only available after a report has been opened.

Report menu	Description
New >	These commands are used to create a report definition.
Wizard	Creates a report, using page-by-page procedures. See also Report Designer Wizard (page 434).
Blank Report	Starts the definition from the draft, using a a blank template.
Open	Opens an existing report (from the current document (page 521)).
Save	Saves the definitions.
Remove	Deletes a report definition.
Rename	To rename the report.
Import/Export >	These commands are used to exchange report definitions with XML files.
Export XML Report Definition Import XML Report Definition	For importing/exporting.
Options	You use these options to define paper size and orientation for Web and Desktop reports. See also Report Designer Options (page 433)
Edit menu	
Move >	Moves the layout components relative to each other.
Move Group	Moves the selected child group.
Move Zone	Moves the selected zone.

Move Page	Moves the selected page.
Remove >	These commands are used to delete groups, zones, and pages. Select the respective item and start the command.
Delete Controls F11	Deletes the selected controls.
Copy Controls	Copies the selected controls.
Paste Controls	Pastes the controls (to the selected section).
Add menu	These commands are used to add controls. See also Report Designer: Adding Controls (page 429).
Print menu	These commands are used to generate reports in different formats, such as Print or HTML. These commands are also available on the Preview tab. See also Report Designer Options (page 433)
Preview F5	Displays a preview. If errors in SQL statements, formulas, and functions occur, these are displayed in an error list.
Print	Prints the report on a printer.
PDF	Creates a report in PDF.
HTML	Creates a report in HTML.
ASCII	Exports data into an ASCII text file. See also Report Designer ASCII Export (page 454).

Report Designer: Adding Controls

You can customize the report layout, using a number of controls to add elements, such as static text, line, and pictures.

To add a control to a report definition

1 Start Report Designer and open a workspace.

2 Click Add menu and select a control, as shown in the following table:

Add menu >	**Description**
DB Column	Adds a database attribute. See also Report Designer: DB Column Control (page 441).
Text	Adds a text control. Use this control to add static text, such as headings, annotations, and comments. See also Report Designer: Text Control (page 439)
Picture	Adds a picture to the report. **TIP** To embed a WMS URL, add a picture control and enter the URL as file name.
Line	Adds a line. You can modify the line parameters either in the right pane by dragging the line or in the left pane by editing the properties. See also Report Designer: Line Control (page 443).
Chart	Adds a chart. See also Report Designer: Chart Control (page 444).
Function >	These commands are used to process functions. You use functions to display the results of calculations in the report. See also Report Designer: Functions (page 448).
Manage	Opens the function manager to define and edit functions.
<name>	Adds an existing function <name> to the current section. You should only add functions to body sections that contain records (SQL statements).
SQL >	These commands are used to process SQL values. See also Report Designer: SQL Values (page 450).
Manage	Opens the SQL manager to define and edit SQL statements.

<name>	Adds an existing SQL value <name> to the current section.
Special >	These commands are used to insert special controls to be used in the layout sections.
Page Count	Displays the total number of pages to print {Report.PageCount}.
Page Index	Displays the page number (starts with one) {Report.PageIndex}.
Page X Of Y	Displays the current page number and the total number of pages.
Continue Page	Displays the page number of the following page {Report.ContinuePage}. Example: Continues on page {Report.ContinuePage}. The complete text is suppressed on the last page.
Record Count	Displays the total number of records in the current body group {Report.RecordCount}.
Record Index	Displays the index of the current record {Report.RecordIndex}.
Not First Record	**NOTE** Do not use this as a stand-alone. It is listed here to show you that this function is available. Use it in functions to identify the first record and, for example, to apply a differing style or to suppress data. The value returns -1 (true) or 0 (false) This function {Report.RecordNotFirst} is normally used together with the Suppress property. For example, in the Suppress property of a text control, enter {Report.RecordNotFirst}. Then, the control is only shown for the first Record and suppressed for all following records.
Print Date	Displays the date {Report.PrintDate}.
Print Time	Displays the time {Report.PrintTime}.

Sum	Displays the sum of values listed in the previous body section, for example, displays the length of all listed pipes. You can use this control only in foot sections, either in detail or page foots. The Sum function is inserted as a text control with the content {Report.Sum(ColumnName)}. Replace ColumnName with the name of the attribute for which you want to calculate the sum, for example, LENGTH.
Average	Displays the average of the listed values. The Average function is inserted as a text control along with the content {Report.Avg(ColumnName)}. Replace ColumnName with the name of the attribute you want to calculate the average of. See also Report Designer: Page Controls (page 437).
Unit >	These commands are used to add unit placeholders. See also Report Designer: Unit Placeholder (page 451)
Plugin Function>	These commands are used to add functionality using the Topobase API.
Manage	Opens the Plug-Ins manager to add and remove Plug-Ins.
<name>	Adds an existing Plug-In <name>.
Parameter >	These commands are used to process parameters. See also Report Designer: Parameter (page 453).
Manage	Opens the parameter manager.
<name>	Adds an existing parameter <name>.
Form Control	Adds a form control in a body section that contains an SQL query. All the DB controls of the queried table are displayed. You can select one or more controls. Note the difference between DB Column and Form Control: A DB attribute displays the value that is stored in the database. A form control displays the database value as it has been defined in the form designer. See also Report Designer: DB Column Control (page 441)

	Example: A database attribute is displayed with a check box control. It can have only the values 0 and 1. In the form, the display of 1 is mapped to the string “closed” and 0 to “Open”. If you use the DB column control, the report shows only the values 1 or 0. If you use the form control, the strings are shown.
Child Group	Inserts a group with head, body and foot section. You can add a child group to the master group or to any child group (between body and foot section). You can define nested groups (child group with child groups) to print master-detail information. See also Creating a Master-Detail Report (page 126)
Zone	Inserts a zone into a section. Select the section to which you want to add a zone. For example, use zones to define conditions to suppress. See also Report Designer: Suppress a Zone (page 458).
Page	Inserts a page. In the right pane, each page is shown on a separate tab. You can add multiple pages with different layout and content to your report. For example, you can create a report head or title page.

Report Designer Options

You can define paper size and orientation for Web and desktop reports.

To set report options

1. Start Report Designer and open a workspace.
2. Click Report menu > Options. Set options, as shown in the following table:

Report Designer Options	Description
Format	Selects a printer format. The values for Paper Width and Height are set automatically but can be modified. It is recommended that you use the A4 Portrait format in the Internet Explorer IE (Web version). Adjust the paper

	height to 2800 and in Internet Explorer click File menu ➤ Page Setup. In the Page Setup dialog box, delete all entries in the header and footer section. If necessary, set the paper height smaller.
Module Name	Sets a module name to be used by the application API. Default: Empty.
HTML Window Title	Specifies a title for the Web preview window.
Page X	Draws a frame. For each page, you can specify the size of a frame to be drawn. For example, with this option you can design a report head (title page).

Report Designer Wizard

You can use the Report Designer wizard to create a report with a basic layout.

In the Report Designer window, click Report menu ➤ New ➤ Wizard. Enter a name. In the New Report dialog box, you can specify the layout of the report, as shown in the following table:

New Report Dialog Box	Description
Table/Form	Selects the feature class or view that contains the data to be printed.
Title	Specifies the name of the report.
Style	Specifies the Page Break property, This property can also be modified on the designer tab. Select Forms to print every record on a new page (Page Break Before = true). Select Table/List to print the records continuously (Page Break Before = False). See also Report Designer: Page Controls (page 437).
Format	Selects the paper format. See also Report Designer Menu Bar (page 428), Options menu.
Attributes	Selects the feature class attributes that you want to add to the report.

Click OK to create the report definition. The definition contains the page head with title and print date, the page body with captions for the selected attributes, and a child group with detail body.

You can use the menu commands to modify these report definitions.

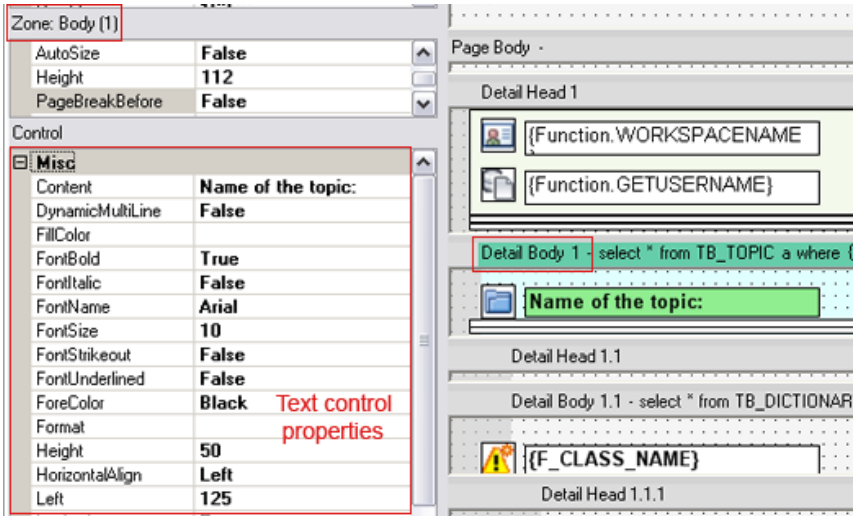
See also:

- Report Designer Menu Bar (page 428)

Report Designer Control Properties

You can define the report layout and content with a number of control elements that can be applied to the different sections.

To work on the different sections, select a section in the right pane of the Report Designer window. The selected section is highlighted. In the left pane the control properties are displayed in the property grid (Design tab).



Report Designer: Specifying text control properties in the property grid

The property grid on the left is categorized: The upper category shows the properties of the group that the selected control belongs to. Following are the section and zone properties, followed by the control properties.

Property Category	Description

Group	Properties for the whole page.
Zone	Properties for the particular head, body, or foot section or a zone.
Control	Properties of the current control, for example, text or picture control properties.

Report Designer Toolbar

Icon	Description
	Creates a new report.
	Opens an existing report definition. You are prompted whether to save the current report definition.
	Saves the current report definition.
	Displays a preview.
	Prints the report.
	Adds a text control. Use this to add static text comments to any section. See also Report Designer: Text Control (page 439).
	Adds a database attribute in a body section that contains an SQL query. See also Report Designer: DB Column Control (page 441)
	Adds a line. See also Report Designer: Line Control (page 443).
	Adds a picture to the report. See also Report Designer: Picture Control (page 442).
	Adds a chart to the report. See also Report Designer: Chart Control (page 444).

More commands are available on the menu bar.

See also Report Designer Menu Bar (page 428).

Report Designer: Page Controls

Note that the positioning properties, such as height, left, or top, can be modified by dragging the element in the right pane.

Page Control Property	Description
Group:	Properties for the whole group (page, detail section).
After Query Before Query	Defines Plug-Ins or functions that are executed before or after the query. For example, you can write data to a temporary file and then print this file. Alternatively, you can change to another job state to print a parcel list with the state before and after the modifications.
Glue	Prints the content together on the same page, as far as this is possible. If no space is left on the current page, the "glued" group is printed on the next page. Select True to print all group sections (head, body, foot, and their children) on the same page. The default is False.
Parallel	Prints one group on the left and another group on the right side of the report. This works like a layer where the second group is printed over the first one. Set to True to print a group parallel to another group.
Repeat Head On New Page	Prints the head section of a group on every new page.
Reset Page Count	Sets the page count to 1.
Show Foot Show Head	Shows or hides the foot or head section. Hide unused sections to make the designer view more clear.

SQL	Defines an SQL statement to retrieve the data to be printed in the particular section. Click the property row, and then click to open the SQL Assistant. See also Using the SQL Assistant (page 105) and Report Designer: SQL Statements (page 447).
SQL Source	Selects the source of the data to be printed. **Database**—Default is Database. **Memory, Plug-In**—Select either of these, if the application developer fills data into the memory in a DataTable object and you want to use this as data instead of reading the data from a database table or database view. This is useful if you need to do some preprocessing and you cannot directly use a database table for the output. Memory can only be used together with the API.
Suppress If No Records	Suppresses the head and foot section if the body section is empty because there are no records available.
Section	Specifies properties for the head, body and foot section
Glue	Prints the section together on the same page as far as this is possible. If no space is left on the current page, the "glued" section is printed on the next page. Select True to print the section (head, body, foot or their children) on the same page. The default is False.
Height	Specifies the height of the section.
Zone:	Specifies the properties of the section zone.
Autosize	Controls the height of the zone. Select True to calculate the height of the zone based on the controls that are inside. Note that this takes effect only at runtime, but not during the design.
Height	Specifies the eight of the zone. Optionally, use the AutoSize property.
Page Break Before	Starts the section on a new page. The default is False.

Suppress	Suppresses the zone under certain conditions. The condition can be a function or a select that produces either true or false (0 or 1). If the condition is True, the zone is suppressed.
Control	Specifies the properties of the particular controls. See also Report Designer: Text Control (page 439) Report Designer: Picture Control (page 442). Report Designer: Line Control (page 443).

Report Designer: Text Control

You can use text controls to add static text, such as headings, annotations, comments, and to define the text, and the styles of the text, as shown in the following table:

NOTE Basically all control types, except picture and line control are text controls, some of which contain functions instead of plain text. For example, a DB Column control used to display the NAME attribute is a text control with the text "{NAME}".

Note that the positioning properties, such as height, left, or top, can be modified by dragging the element in the right pane.

Text Control Property	Description
Content	Defines static text to be displayed. The text can also be modified in an input field in the upper section of the window or by clicking in the content row in the right pane.
Dynamic Multiline	Prints text with variable height. If the text you want to display is a multi-line text with an unknown number of rows, you need this option to dynamically enhance the section. Without the Dynamic Multiline option, the text would probably be printed into the following section.
Fill Color	Defines a fill color for the text box.
Font	Specifies the font attributes.

Fore Color	Defines the text color. You can either select a color clicking or enter either an SQL statement or a function to specify the color. See also Report Designer: Change Colors at Runtime (page 460).
Format	Specifies a format, for example 0.000, for numbers. You can use any VB .NET format. See also Text Control: Format (page 441).
Locked	Locks the control on the grid. A locked control cannot be moved with the mouse. With this option, you can prevent moving a control unintentionally. Note that you can move a locked control with the Left and Top property by specifying the offset values.
Horizontal Align, Vertical Align	Aligns the text in the text box. Select either left, center, right (horizontal) and top, center, bottom (vertical).
Left, Top	Positions the text box relative to the section borders.
Priority	Controls the drawing priority of text and pictures. For example, this would prevent a picture covering a text control. Controls with a low priority value are drawn first. Example: A picture with priority 10 covers a text with priority 1. **TIP** Save and reopen the report to see the result in the preview.
Suppress	Suppresses the control under certain conditions, such as a function or a select statement that produces either True or False (0 or 1). If the condition is True, the text is suppressed. Example: To print the word “Warning” if the status of a tree is “damaged”. Write a function, for example, *Fun.Tree* that queries the attribute and returns True if the tree is healthy. Enter the Suppress property in braces. For example, *{fun.tree}*.
Suppress Duplicates	Suppresses the value if it has the same content as the previous one. This property works only in body sections.

Width	Defines the width of the text box. You can align the text content horizontally and vertically, in relation to this box.
Width ASCII	Specifies formatted attributes for an ASCII export. You can define the width of the text box as number of characters. Additionally, you can align the text to the left or right. See also Report Designer ASCII Export (page 454).

Text Control: Format

You can use any VB .NET format, either in a predefined format or with your own definitions. This means that there is a flexible format specification for different types of data (strings, numbers, date).

TIP Syntax validation is not available, but you can switch to the Preview tab to check that the data is displayed correctly in the report.

Example: Formats for a DB Column control (number (2) attribute)

Value	Format	Result displayed
2	($#,##0)	($2)
2	($#,##0.00)	($2.00)
2	0.00	2.00
2	0.00%	200.00%

Report Designer: DB Column Control

Use this control to add a database attribute. You can select any of the attributes that are retrieved by the body section SQL statement.

NOTE This control is identical to the text control. The difference is that you can select the DB attribute from a list. However, if you know the attribute name, you can enter it as text control content as well. Make sure the attribute name is in {} braces. You can also mix parameters, such as {Column} with static text, functions, and parameters.

You can also use a form control, which is similar to the DB Column control. Click Add menu ➤ Form Control. See also Report Designer: Adding Controls (page 429).

For control properties, see Report Designer: Text Control (page 439)

See also:

- Report Designer: SQL Statements (page 447)

Report Designer: Picture Control

Note that the positioning properties, such as height, left, or top, can be modified by dragging the element in the right pane.

Picture Control Property	Description
Height	Specifies the height of the picture.
Keep Size	Scales the picture uniformly (in X and Y). This effect is visible only in the preview.
Left, Top	Positions the picture relative to the section borders.
Locked	Locks the control on the grid. A locked control cannot be moved with the mouse. With this option, you can prevent moving a control unintentionally. Note that you can move a locked control with the Left and Top property by specifying the offset values.
Picture File	Selects the picture to be displayed. The default path is <topobase>\pics. **NOTE** Although the picture files can be stored in any folder, it is recommended that you store them in <topobase>\Pics or a subfolder. Then, they are available with the dynamic path <topobase>\Pics. If you use picture files with a static path, they are not available on a different computer unless the path is exactly the same. For variants and Web version, see Form Designer: Picture Linker - Properties (page 347)

Priority	Controls the drawing priority of text and pictures. For example, this would prevent a picture covering a text control. Controls with a low priority value are drawn first. Example: A picture with priority 10 covers a text with priority 1. **TIP** Save and reopen the report to see the result in the preview.
Suppress	Suppresses the control under certain conditions, such as a function or a select statement that produces either True or False (0 or 1). See also Report Designer: Text Control (page 439).
Width	Specifies the width of the picture. Can also be specified with the mouse.

Report Designer: Line Control

Note that the positioning properties, such as height, left, or top, can be modified by dragging the element in the right pane.

Line Control Property	Description
Height	Specifies the length of a vertical line.
Left, Top	Positions the line relative to the section borders.
Line Size	Specifies the line weight.
Locked	Locks the control on the grid. A locked control cannot be moved with the mouse. With this option, you can prevent moving a control unintentionally. Note that you can move a locked control with the Left and Top property by specifying the offset values.
Priority	Controls the drawing priority of text and pictures. For example, this would prevent a picture covering a text control. Controls with a low priority value are drawn first. Example: A picture with priority 10 covers a text with priority 1.

Suppress	Suppresses the control under certain conditions, such as a function or a select statement that produces either True or False (0 or 1). See also Report Designer: Text Control (page 439).
Vertical	Draws a vertical line. Select True to draw the line.
VerticalAuto Height	Draws a line to the bottom of the page. Select 1 to draw the line. The default is 0 (Off). Enter a value to be added to the line height. For example, enter "-100" to draw the line almost to the bottom of the page (where Pageheight -100).
Width	Specifies the length of a horizontal line.

Report Designer: Chart Control

Note that the positioning properties, such as height, left, or top, can be modified by dragging the element in the right pane.

Chart Control Property	Description
Config	Opens the chart wizard to define the chart layout. Click the property, and click . See also Using the Chart Wizard (page 445).
Height, Width	Specifies the height and the width of the chart.
Left, Top	Positions the chart relative to the section borders.
Locked	Locks the control on the grid. A locked control cannot be moved with the mouse.
Priority	Controls the drawing priority.
Script	Adds a script. For example, to modify the chart at runtime. Click the property, and click .

SQL	Edits the SQL expression that queries the data to be visualized. Click the property, and click to open the SQL Assistant. See also Using the Chart Wizard (page 445) and Creating SQL Expressions (page 103).
Suppress	Suppresses the control under certain conditions, such as a function or a select statement that produces either True or False (0 or 1). See also Report Designer: Text Control (page 439).

Using the Chart Wizard

You use the chart wizard to design the chart. A chart can be of type Bar, Line, Point, or Pie. Use the tabs on the left side to specify the layout. On the right side, a preview of the chart is displayed.

To open the chart wizard

1. Start the Report designer, and open the report definition.
2. Do one of the following:
   - Click Add Chart To The Report to insert a new chart.
   - On the Design tab, select an existing chart control. On the left pane, in the property grid, select Config, and click .
3. In the Chart Wizard dialog box, specify the chart layout.
4. Click OK.
5. In the Report Designer, on the Design tab, define the SQL select statement to query the data to be visualized.
6. Select the chart control. In the Control property grid, select Sql, and click .

7 In the SQL Assistant dialog box, define the SQL select statement that returns the data to be visualized in the chart.

8 Click the Preview tab, to generate a preview of the report.

Select Statements for Charts

When you add a chart control, a sample SQL select statement is provided. Depending on the chart type, the query for the chart representation, must return an appropriate number of values.

Return values for the different chart types

- Line chart—Series (legend text), X, Y
- Point chart—Series (legend text), X, Y
- Bar chart—Legend, value
- Pie chart—Legend, value

The sample SQL select statement shown below visualizes how many feature classes of each type are stored in the system table TB_DICTIONARY. Use a chart of type Bar.

```
select F_CLASS_TYPE,
count(F_CLASS_TYPE) from TB_DICTIONARY group by F_CLASS_TYPE
```

For example, in a Line chart, each line corresponds to a series, and the select statement could return the following <series>,<X>,<Y>.

```
1,15,20
1,16,30
1,17,31
...
2,15,20
2,16,30
2,18,19
...
3,15,20
3,16,30
3,17,31
...
```

In the chart, 3 lines would be drawn.

See also:

- Report Definition: Adding a Chart (page 124)

Report Designer: SQL Statements

To retrieve the data to be printed in the report, you can define SQL statements in the body sections. You can use record values in body sections and SQL statements.

NOTE Record values are written in braces { }.

Record Value	Description
{ColumnName}	Displays an attribute name. The value is replaced at runtime by the record value of the respective SQL statement that is defined for the group. You can also use record values with text. Example: The display model {Name} has the FID {Fid}.
{Parameter.Filter}	Filters records. The filter is replaced at runtime with the current filter of the respective form, so that only the data of the current filtered records is printed.
{Parent.ColumnName}	Finds the record value of a parent group.
{Parent.Parent.ColumnName}	Finds the record value of the parent of a parent.
{Report.Sum(ColumnName)}	Calculates the sum of the attribute values. You can use this function only in the body and foot section. Note the differences between the sections: In the Body section, calculates the current sum. In the Foot section, calculates the total sum. **TIP** To calculate the sum in a head section you can use SQL values. Example: `select sum(money) from table where parent={key}.`

{Report.Avg(ColumnName)}	Calculates the average. You can use this function only in the body and foot section.
{fun.xxx}	Executes functions. See also Report Designer: Functions (page 448).
{sql.xxx}	Runs a complex SQL statement. See also Report Designer: SQL Values (page 450).
{Parameter.xxx}	See also Report Designer: Parameter (page 453).

NOTE If a record value that is used in an SQL statement returns a NULL value, this is translated to 0 or ''.

See also:

- Report Designer: DB Column Control (page 441)

Report Designer: Functions

You can use functions in body sections to calculate any values to be printed in your report. Functions must be written in VB .Net. These functions provide all possibilities that are available in VB .NET, for example, IF – THEN statements. In addition, the functions provide all Topobase API functions.

To define a function, click Add menu ➤ Function ➤ Manage. In the Function manager dialog box, you can define and edit the functions.

Function Manager	Description
Add	Adds a function. Click the button and enter a function name, for example, MyTest.
Remove	Deletes a function.
Syntax Check	Checks the syntax of the function.
Function	Enters the function code. If you add a function, a sample code is displayed: <name> `= Me.Record.LngValue("Fid") * 10.`

	This code reads the value of the attribute FID and multiplies it with 10. *Me.record* is of type Topobase.Data.Tools.RecordReader and can be used in the same way.
Record.LngValue("Fid")	Reads a numeric value from VB data type Long
Record.DblValue("Scale")	Reads a numeric value from VB data type Double
Record.StrValue("Name")	Reads a text value from VB data type String

To add a function to the report, click Add menu ➤ Function ➤ <name>.

NOTE You can also add a text control and enter the text {fun.name}, for example, *{fun.MyTest}*.

See also:

- Report Designer: Text Control (page 439)
- Form Designer: Formula TextBox - Properties (page 344)
- Report Designer: Suppress a Zone (page 458)

Report Designer: Functions - Example

In the example you create a function to print rounded values. The functions uses parameters. With the parameters you can apply the function for different database columns. The function prints

- "--" if the attribute value is empty.
- "Infinite" if the value is 999999.
- the rounded value, for all other values.

To create a function

1. Start the Report designer, and open a report.
2. Click Add menu ➤ Function ➤ Manage.
3. In the Functions dialog box, click Add.
4. In the New Function dialog box, enter a name, such as MyRound. Click OK.

5 Under Function, enter the code.

```
dim columnName AS String
dim scale as integer
columnName = Me.FunctionParameters.Item(0) ' First parameter
scale = Me.FunctionParameters.Item(1)' Second parameter
if Me.Record.IsDBNull(columnName) then
MyRound = "--" 'Attribute value is empty
else
dim value as double
value = Me.Record.DblValue(columnName)
if value = 999999 then
MyRound = "Infinite"
else
MyRound = System.Math.Round(value, scale)
end if
end if
```

6 Click Syntax Check.

7 Click Close.

8 In the report, add the function Control and enter the parameters, for example:

- {Fun.MyRound("column_name1",2)

9 {Fun.MyRound("column_name2",7)}

Report Designer: SQL Values

You can run complex SQL statements that are related to the current records, and retrieve a single value. This is similar to the SQL label controls in the Form Designer.

For example, you can define an SQL value to display the value of a domain table (_TBD) instead of the value that is stored in the related table.

To define an SQL value, click Add menu ➤ SQL ➤ Manage. In the SQL Manager dialog box, you can define and edit the functions.

SQL Manager	Description
Add	Adds an SQL statement. Click the button and enter a name, for example: MySQL.

Remove	Deletes a statement.
Assistant	Starts the SQL Assistant to build the statement.
SQL	Shows the SQL statement. You can edit the statement directly in this dialog box or click Assistant. Example: `select value from SYMBOL_TBD where id={id_symbol}` The value {id_symbol} is replaced at runtime by the value of the current record.

To add an SQL value to the report, click Add menu ➤ SQL ➤ <name>.

NOTE You can also add a text control and enter the text {SQL.name}, for example, *{SQL.MySQL}*.

In the report, the value in braces {} is replaced at runtime by the result of the SQL statement.

TIP If you have a Check Box value (Yes/No value), you can define an SQL as follows: `select decode({Active},0,'No',1,'Yes') from dual.`

See also:

- Creating SQL Expressions (page 103)
- Using the SQL Assistant in the Form Designer (page 108)

Report Designer: Unit Placeholder

You can use unit placeholders to display unit names, such as foot, inch, centimeters, for length values.

All available units and their translations are stored in the Topobase system. If you use a placeholder instead of a static text to display the unit name, the name will always appear in the appropriate language. This is useful if a report is used in different countries, using different languages.

There are different placeholders:

- **Default unit**—Displays the document default unit as specified in the unit settings during the creation of the document.

- **Unit item**—Displays an individually selected unit.
- **Unit from attribute**—Displays the unit as specified in the attribute's unit type during the creation of the feature class attribute in the data model administrator.
- **Unit from SQL field**—Displays the unit that is stored with a feature in a special feature attribute.

Unit placeholder	Description
For default units: {Unit.Area} {Unit.Length} {Unit.<name>}	Adds a default unit to your report. Click Add menu ➤ Unit ➤ <name>. This adds a text control containing the appropriate placeholder.
For unit items: {Unit.Item("Centimeter")} {Unit.Item("Foot")} {Unit.Item("<name>")}	Adds a unit item to your report,. Click Add menu ➤ Unit ➤ Units. Select a unit from the list. This adds a text control containing the placeholder.
For attribute units: {Unit.Attribute("Feature-ClassName","Attribute-Name")} {Unit.Attribute("Feature-ClassName","Attribute-Name", "DefaultUnitType")}	Adds an attribute unit to your report. Click Add menu ➤ Text and enter the placeholder manually in the content property. Note that the expression must be enclosed in braces.
For SQL field units: {Unit.Field("ColumnName")}	Adds an SQL field unit to your report. Click Add menu ➤ Text and enter the placeholder manually in the content property. Note that the expression must be enclosed in braces. ColumnName is the attribute of the current feature class that stores the unit ID. It has to be part of the query result of the particular section.

See also:

- Report Designer Menu Bar (page 428)
- Report Designer: Text Control (page 439)
- Data Model Administrator: Creating a Database (page 144)

Report Designer: Parameter

Use parameters to set values using the API at runtime. The expression *{Parameter.Filter}* is used as default. It is replaced by the current filter of the form. The value is used if the report is shown in the report designer.

To edit the default {Parameter.Filter}

1. Start the report designer and open a report.
2. Click Add menu ➤ Parameter ➤ Manage.
3. In the Parameter dialog box, select FILTER and edit the value. The default is "(1=1)".

The following are examples of parameters:

Filter	Expression	Replacement
fid=10	"and h. {Parameter.Filter}"	"and h. fid=10"
fid=10 or fid=13	"and h. {Parameter.Filter}"	"and h. fid=10 or fid=13"
fid=10 or fid=13	"and h.fid in (select fid from WA_HYDRANT where {Parameter.Filter})"	"and h.fid in (select fid from WA_HYDRANT where fid=10 or fid=13"

To create a new parameter

1. Start the report designer and open a report.
2. Click Add menu ➤ Parameter ➤ Manage.
3. In the Parameters dialog box, click Add.
4. In the New Parameter dialog box, enter a name and click OK.
5. In the Parameters dialog box, enter a value.

You can also specify a default filter value for a stand-alone start of a Topobase report. An application developer can set the filter by using the API if the report is called without a Topobase feature class form.

Report Designer Import Report Definition

You can import Topobase XML report definitions. XML report definitions can be created with the report designer using the export command.

Use XML report definitions to share report definitions with other users.

To import a report definition

1. Start Topobase Administrator and open the workspace.
2. In the Administrator explorer, select the document to which you want to import the report definition.
3. Click Document menu ➤ Report Designer.
4. Click Report menu ➤ Import/Export ➤ Import XML Report Definition.
5. Select the XML file and click OK.
6. Click Report menu ➤ Open and select the report. You can customize the report definition if necessary.

Report Designer ASCII Export

You can use Topobase reports to create ASCII text files. With this you can use it as an export interface that can be customized.

To export data in an ASCII file

1. Create a report that contains the database attributes you want to export.
2. Make sure that the Width ASCII property matches the desired character count of your output file. You can also set Width ASCII to 0 (not NULL). In this case the width is set to <number of characters + 1 space>.
3. In the Preview tab, click the ASCII button to start the export.

Report Designer: Advanced Topics

Report Designer: Polygon and Line Definition Report

Using Topobase Client and Web, you can print detailed polygon or line definition reports, such as a parcel report that lists the area, the distance between the parcel border points, the parcel borders, the points, and parcel number. These detailed reports require a special configuration. If you have installed the Land demo data set, you can reproduce the following steps.

To configure a polygon/line definition report

1. Start Topobase Client and open the workspace.
2. In the document explorer, right-click the system table TB_SURFACE_CONFIG and click Show Form.
3. Add a new record with the following attributes:

   ID: feature class ID of the line string or polygon feature class, as for example 29 (LM_PARCEL_L).

   NAME: The name of the configuration will be displayed in the Report For Lines/Polygons dialog box.
4. In the document explorer, right-click the system table TB_SURFACE_POINT and click Show Form.
5. For each point feature class you want to include in the report, add a new record with the following attributes:

   SURFACE_CONFIG_ID: TB_SURFACE_CONFIG.ID of the record you created before.

   POINT_F_CLASS_ID: feature class ID of the point feature class, as for example 37 (LM_POINT).

   PRIORITY: You can include several point feature classes. Only the points of the feature class with the highest priority will be used. If a vertex has no such point, the point feature class with the next priority will be used.

   Make sure to use point feature classes that store points that are located on the specified polygon or line features.

NOTE When you select a feature class in the Topobase Client document explorer, you can see the ID of that feature class in the status bar of the Topobase task pane. See also Setting Document Explorer Options (page 22).

To create the report definition

1. Start Topobase Administrator and open the workspace.
2. Start the report designer and create a new report definition for the table TB_SURFACE_DEF.
3. In the report definition, set the module name. Click Report menu ➤ Options. In the Other Options section, enter the Module Name: SURFACE_DEFINITION.

To print a polygon/line report

1. Start Topobase Client and open the workspace.
2. See Topobase Client User Guide, section Printing Reports.

Report Designer: Spatial Selection Report

In Topobase applications you can perform spatial selections for reports to print. For example, you can select all features that lie within a job perimeter or within a specified area. For this type of reports a special definition is required.

To create the report definition

1. Start Topobase Administrator and open the workspace.
2. In the Administrator explorer, select the document for which you want to define the report, such as TB2010_GA_102.
3. Click Document menu ➤ Report Designer.
4. Click Report menu ➤ New.
5. Create the report definition.
6. To set the module name, click Report menu ➤ Options.
7. In the Other Options section, enter the Module Name: *Topobase.SpatialExportTool*.

In the report definition you must use the parameter PERIMETERID.

To add the parameter PERIMETERID to the report definition

1. In the Report Designer, click Add menu ➤ Parameter ➤ Manage.
2. In the Parameters dialog box, click Add.
3. In the New Parameter dialog box, enter the parameter name PERIMETERID.
4. Under Value, enter 1.
5. Click OK. Click Close.

PERIMETERID contains the ID of the perimeter definition in TB_PERIMETER. It is always set. TB_PERIMETER stores the perimeter that you digitize when you use the Spatial Export tool in Topobase Client. For example, in the Spatial Export dialog box, when you select No Spatial Selection, TB_PERIMETER.ID = 0.

Your report definition must contain a query for the PERIMETERID, as shown in the following sample.

```
WHERE
{parameter.PERIMETERID}=0
OR SDO_RELATE(T.geom, (SELECT geom FROM TB_PERIMETER
WHERE id={parameter.PERIMETERID}),
'mask=anyinteract query=window')='TRUE'
```

To define the report definition

1. Click the Detail Body 1 section.
2. On the Design tab, in the Group Detail category, click the SQL property.
3. Define a query that selects the features which are located in the perimeter.
4. Click Add ➤ DB Column, and select the attributes you want to display.
5. If needed, add more controls, and save the report.

Example: Query to select features within a perimeter.

```
select P.FID from GA_PIPE P, GA_LINE T
WHERE P.FID = T.FID_ATTR AND
({parameter.PERIMETERID}=0
 or SDO_RELATE(T.geom, (SELECT geom FROM TB_PERIMETER
 WHERE id={parameter.PERIMETERID}),
'mask=anyinteract query=window')='TRUE')
GROUP BY P.FID ORDER BY P.FID
```

The parameter SPATIALMASK is only set if it is in the report definition. Sample SQL statement:

```
WHERE
{parameter.PERIMETERID}=0
OR SDO_RELATE(T.geom, (SELECT geom FROM TB_PERIMETER
WHERE id={parameter.PERIMETERID}),
'mask={parameter.SPATIALMASK} query=window')='TRUE'
```

For more information about using the spatial selection reports, see the Topobase Client User Guide, section Spatial Export.

Report Designer: Coordinate Report

Topobase provides templates to print coordinate report. You use the Coordinate Export tool to print coordinate reports. See Topobase Client User Guide - Printing Reports.

When you create your own coordinate reports, you must use the parameter filter `FilterCoordinateFID`.

See also:

- Report Designer: Parameter (page 453)

Report Designer: Suppress a Zone

You can suppress a zone of a report, if that zone does not display any content. For example, if the attribute value to be printed is empty.

To suppress a zone with no content to print

1. Create a zone containing the attribute, for example COMMENT.
   - Select the section, and click Add menu ➤ Zone.
   - Add the attribute to be printed, such as COMMENT.

2 Define the condition for the zone to be suppressed. For example, suppress the zone if the attribute COMMENT is empty.

- Click Add menu ➤ Function ➤ Manage.
- Add a new function *IsEmpty*.
- ```
  dim columnName as string
  columnName = Me.FunctionParameters.Item(0)
  return Me.Record.IsDBNull(columnName)
  ```

3 Select the zone. Under Suppress, enter the condition *{Fun.IsEmpty("COMMENT ")}*.

See also:

- Report Designer: Functions (page 448)

Report Designer: Create a Crystal Report

In Topobase Client, you can use reports that have been generated using Crystal Reports™.

When you generate a report, follow the using

Guidelines for creating a report

- At least, select the table TB_CRYSTAL_REPORTS, and the feature class.
- Both TB_CRYSTAL_REPORTS, and the feature class must have a column FID.
- In the Crystal Reports designer, link the FID columns.

To add the report to a feature class form

1 Copy the Crystal Reports (*.rpt) file to your reports rook directory.

You specify the report root directory in the Application Options. See Setting Common Application Options (page 16).

2 Start Topobase Client, and log in as Admin user.

3 Open the feature class form.

4 On the form toolbar click the Print icon.
  ```

5 In the Report dialog box right-click anywhere and click Add ➤ Crystal Reports.

6 Enter a title, and select the Crystal Reports (*.rpt) file. Click OK.

Report Designer: Call a Report via API

Create a reference to the assembly "Topobase.Forms.Report.dll":

```
Dim report As New Topobase.Forms.Report.Report(connection, _
"Name of the Report")
report.SetParameter("Filter", "Fid=17)")
report.SetParameter("Title", "Annual Report 2005")
report.PreView() ' Show the Report
You can also save the report as HTML via the API.
report.SaveHtml("c:\test.html")
```

NOTE Take care to always define the parameters in the report before setting a value to a parameter.

Report Designer: Change Colors at Runtime

The following example shows how to change colors at runtime.

To change colors at runtime

1 Enter a function name in the ForeColor property of the text control instead of the color name (or HTML color) itself. For example,
`{Fun.TreeColor}`

2 Create a function that calculates the color you want to set for each single record.

3 In the report designer window, click Add menu ➤ Function ➤ Manage.

4 Click Add to add a new function. Enter the name, for example, TreeColor.

5 In the Function pane, define:

```
if Me.Record.LngValue("status") = 1 then
 TreeColor = "red"
```

```
else
 TreeColor = "black"
end if
```

6 In the designer window, select the text box and set the ForeColor property as follows: {fun.TreeColor}.

See also:

- Report Designer: Functions (page 448)

Report Designer: Reading Data from the Memory

Example:

```
Dim table As New DataTable
```

‘Define a Table in Memory

table.Columns.Add("FID", System.Type.GetType("System.Int64"))

```
table.Columns.Add("NAME", System.Type.GetType("System.String"))
Dim i As Integer
‘Fill the Table with Data
For i = 1 To 20
Dim newRow As DataRow
newRow = table.NewRow
newRow.Item("FID") = i
newRow.Item("NAME") = "a" & i & "bbb" table.Rows.Add(newRow)
Next
Report.SetDataSource("MyMemoryTest", table)
```

In the report definition, you then can treat this table like a database table and apply a SQL statement:

```
select * from MyMemoryTest where Fid={Fid}
```

Set the SQLSource property to Memory.

Position Finder Reference 10

Position Finder Introduction

Using the Topobase position finder, you can quickly find the location of a certain object, such as a building, parcel, or any other type of feature that has geometry. The detected geometry becomes the center of a graphic generation or a zoom GoTo.

Topobase position finder is based on search definitions that can arbitrarily be defined. The definitions are stored in the document (page 521) in the system table TB_POSITION_FINDER. That means that you can define special search definitions for each document.

Both Topobase Client and Web use the position finder to generate graphic using geographical selection.

Topobase Client - Advanced Generate Graphic dialog box: In the Position Finder group box, you can specify a map center by selecting a certain object. The searches can be defined with the position finder. This search is of the search type sequential with input type combo box.

There are three types of position finder search:

- **Sequential Search**—By performing a sequential search, you improve the search from one level to another. Finally, you get the feature you want. The sub-searches can be defined with selection lists or text boxes. For example, you first select a city, then the street, and finally a building.
- **Flat Search**—The flat search is well known from other address or phone number searches. You can fill in what you know about the location to find. The position finder searches the data source for all matching results. You must select the result for the center of the graphic generation process.
- **PlugIn search**—This can be used if the geometries do not exist in the database. The coordinates for the bounding box come from the memory. It is possible to code PlugIns, which return coordinates from given parameters.

In the Web version, you can highlight the result with a symbol. See also Setting Up Topobase Web (page 63).

See also:

- Position Finder Administrator (page 464)
- Defining a Sequential Search (page 468)
- Defining a Flat Search (page 472)
- Position Finder PlugIn Search (page 474)
- System Table TB_POSITION_FINDER (page 92)

Position Finder Administrator

Use the Position Finder Administrator to define and edit search definitions.

To start the Position Finder Administrator

1. Start Topobase Administrator.
2. In the Administrator explorer, select the document (page 521).
3. Click Document menu ➤ Position Finder.

To view or edit a position finder definition

1. In the Definition list, select the definition you want to edit.
2. Click Rename to rename the search name.
3. Click Delete to delete the search definition.
4. Click Create to define a search definition.

See also:

- Position Finder Sequential Search (page 465)
- Position Finder Flat Search (page 472)
- Position Finder PlugIn Search (page 474)

Position Finder Sequential Search

When you use sequential search, you use several steps to find the center feature. Each step improves the results with different SQL statements.

- Each step has a select statement and a title for the query. The geometry query is optional.
- The select statement returns some result rows.
- The first attribute contains the key (mandatory); the second attribute contains a text (optional).
- The values of these rows will usually be elements in the next searches. They can be used in the following select statements with the help of placeholders.
- Every select statement (apart from the first) can access the values of the rows returned by the previous search, which is also true for the geometry queries. When a result row of a select statement has a geometry attribute, it is possible to create a geometry query for this search. This geometry query will always look as follows:

```
Select geom from <table> where fid = {x}
```

In this query, you only select the geometry attribute. The WHERE clause is composed of the primary key attribute and the current ID of the previous

returned row. The placeholder {X} is replaced by the ID of the selected result row (X=0 for the first select statement).

See also:

- Defining a Search to Find a Building (page 475)
- Defining a Search to Find an Object (page 477)

Sequential Search: Input Types

In sequential searches, the input type specifies how the user communicates the next sub searches. The main difference is in the execution of the search, as in the following input types:

- Selection List (default)—You can select further search items from a combo box. In this case, the select statement has already been executed.
- Text Box—You have to enter the search text. Then, the query is executed. The search text is part of the WHERE clause in the SQL statement.
- Text Box With Auto Complete— You do not have to enter the complete search text. The search starts as soon as you start to enter the text. The search text is part of the WHERE clause in the SQL statement.

IMPORTANT For Text Box and Text Box With Auto Complete, the SQL statement must contain a LIKE statement. Also, you must use the placeholder {0}.

Text Box example—To find table names, define the following SQL statement:

```
select f_class_name from tb_dictionary where upper(f_class_name)
like upper('{0}')
```

Using this definition in Topobase Client, for example, you enter "lm_b*". At runtime, the placeholder {0} is replaced by the character the user has entered. The * is replaced with a "%" and all tables starting with LM_B* are displayed.

Text Box With Auto Complete example— To find table names, use the same SQL statement:

```
select f_class_name from tb_dictionary where upper(f_class_name)
like upper('{0}')
```

For example, using this definition in Topobase Client, you start to enter any character. At runtime, the placeholder {0} is replaced by the character the user

has entered plus a prefix %. The result is all tables that start with A, as shown in the following SQL statement:

```
select f_class_name from TB_dictionary where UPPER (f_class_name)
 like UPPER ('A%')
```

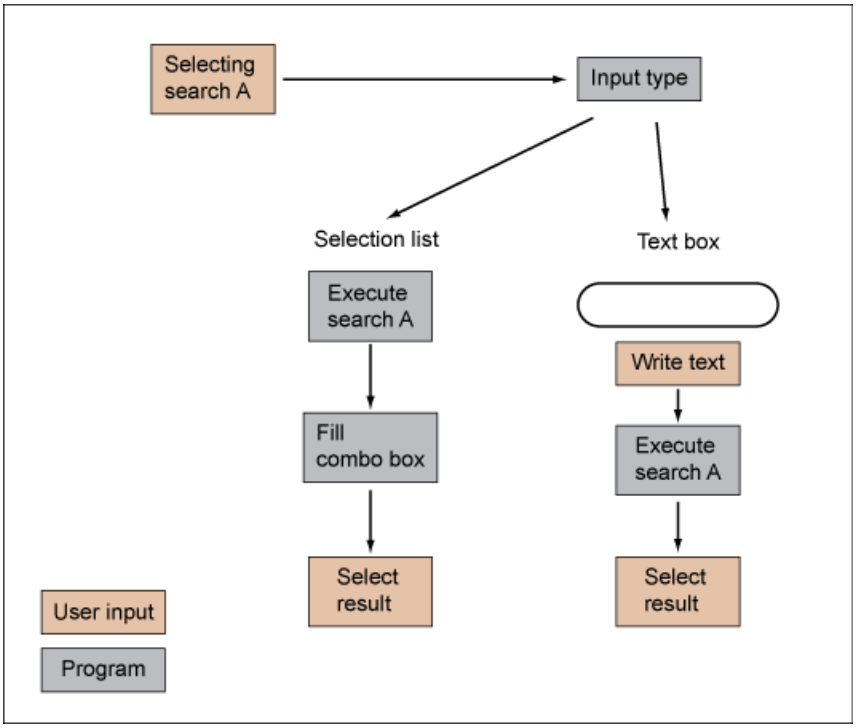
Selection List example— To find street names, use the following SQL statement:

```
select distinct LABEL_TEXT from LM_STREET_TBL where GEOM is not
NULL order by LABEL_TEXT
```

Text Box example— To find street names with a text box, use the following SQL statement that includes a LIKE statement and uses the placeholder {0}.

```
select distinct LABEL_TEXT from LM_STREET_TBL where LABEL_TEXT
like ('{0}') and GEOM is not NULL order by LABEL_TEXT
```

The following diagram shows the workflow with different input types.



Position Finder: Sequential search workflow with different input types.

Sequential Search: Placeholders

When you define a sequential search, you can use placeholders to access the values that were returned in the previously executed SQL statements. This is very helpful if you need the ID of the selected row in the next select statement.

For example, to select the buildings in a certain street, you need the ID of this street in the buildings query.

Placeholders are written in {} braces: {X}, where X is the number of the select statement that the result was returned from (index).

NOTE The index always begins with 0 (zero). This means that the results of the first query must be accessed with {0}.

Example:

Statement 1:

```
Select id, name from City order by name;

Result rows for 1:

ID NAME

 1 Athens

 2 Berne

 3 Berlin
```

Statement 2:

```
Select id, name from Street where id_city = {0} order by name
```

In this example, the query for statement 2 depends on the previously selected result, because it contains a placeholder '{0}'. When you select the city Berlin, the following query is executed:

```
Select id, name from Street where id_city = 3 order by name
```

NOTE The index has to be smaller than the number of search queries.

Defining a Sequential Search

Use the Topobase Position Finder Administrator to administer sequential search definitions.

To define sequential search

1. Start the position finder administrator
2. Click Create to define a new search.
3. Enter a name, such as Streets.
4. For Position Finder Type, select Sequential Search.
5. Click OK.
6. In the Position Finder Administrator, click the Statement 1 tab and specify the first query, for example, select (demo data set):

   Title: Street Name.

   Input Type: Selection List.

   Select Statement: `select distinct LABEL_TEXT from LM_STREET_TBL where GEOM is not NULL order by LABEL_TEXT`

   Geometry Select Statement: `select GEOM from LM_STREET_TBL where LABEL_TEXT like '{0}'`
7. If necessary, click each of the Statement 2 to Statement 4 tabs and specify further queries.

For each statement, enter a title, input type, and select statement. Optionally, enter a geometry statement, using options, as shown in the following table:

Input	Description
Definition	Specifies the name of the definition, which appears in the Topobase Client selection list.
Type	Specifies the input type that configures how the user will interact for the next searches. See also Sequential Search: Input Types (page 466)
Validate Execute	Validates or executes the select statement. See also Sequential Search: Executing and Validating (page 470)
Assistant	Opens the SQL Assistant. See also Creating SQL Expressions (page 103)

The second attribute of the select statement is shown in the position finder selection list. If there is no second attribute, you are prompted for the first. The first attribute usually is used for the primary key.

It is recommended that you do not select more than two attributes.

NOTE You can define a Not Found message in the TB_POSITION_FINDER system table.

See also:

- System Table TB_POSITION_FINDER (page 92)

Sequential Search: Executing and Validating

When defining the search statements in the position finder administrator, you can test them either by executing the search or by validating the statements.

To execute a sequential search

1. Click the Statement_x tab.
2. Click Execute.

To validate a sequential search definition

1. Click the Statement_x tab.
2. Click Validate.

See also:

- Select Statement Validation (page 109)

Sequential Search: Example

The following position finder definition example is also provided with the demo data set. It demonstrates a sequential search that is used to find a feature by selecting a feature class, then selecting an attribute, and then selecting a value for the attribute.

Input	Value

Statement 1 tab	
Name	Feature over Attribute. This definition is a sequential search.
Title	Feature Class
Select Statement	`select f_class_name, caption from TB_Dictionary where f_class_type <> 'T' order by f_class_name`
Input Type	Selection list
Statement 2 tab	
Select statement	`select COLUMN_NAME from cols where TABLE_NAME = '{0}' and COLUMN_NAME not in (select COLUMN_NAME from user_sdo_geom_metadata where TABLE_NAME = '{0}' or TABLE_NAME = '{0}_HOST')`
Title	Attribute
Input Type	Selection list
Statement 3 tab	
Title	Value
Select Statement 3	`select distinct {1} from {0}`
Input Type	Selection list
Statement 4 tab	
Title	Feature
Select Statement 4	`select fid from {0} where {1}='{2}' order by fid`
Geometry Select Statement	`SELECT GEOM FROM {0} WHERE FID={3}`
Input Type	Selection list

Position Finder Flat Search

Using a flat search, you can find a location, using only one step. You need to fill text boxes 1 - 5 with all the information you know about the target location. The position finder will then dynamically compose a select statement and execute it against a flat data table containing the locations.

The generated search statement is as follows.

```
select <pk_column>, <input_colums>
from <table_name>
where upper(<input_column>) like upper(<param>)
order by <order_column>
```

This statement is composed by the position finder when you start the flat search. If the query returns more than one row, the results are presented in a separate dialog box, and you can select an item to be the center of the next graphic generation.

If the query returns only one row, the Selection dialog box is not shown.

After you have selected a result (or when only one row has been returned), the position finder finds the geometry of the selected result. The geometry select statement is as follows:

```
select <geom_column> from <table_name> where <pk_column> = key
```

This statement is composed and executed by the position finder. The geometry (coordinates) are saved in the View port.

See also:

- System Table TB_POSITION_FINDER (page 92)

Defining a Flat Search

Use the Topobase position finder administrator to administer flat search definitions.

To define a flat search

1. Start the Position Finder Administrator
2. Click Create to define a new search.
3. Enter a name, such as Streets - flat.

4 For Position Finder Type, select Flat Search.

5 In the Settings tab, you can specify the search query, using options as shown in the following table:

Input	Description
Basic	Defines a data source. You must define a view or select a table containing all data. This data source must contain a Key Attribute (primary key), a Geometry Attribute, and at least one input attribute Attribute 1. Optionally, you can specify a sort attribute to order the result rows. The data in this table or view is normally in the first normal form (1NF).
Search Attributes	Specifies the search attribute. The description appears on the Topobase Client GUI when you execute the search.
Additional Where Clause	Adds a WHERE clause to the automatically composed select statement.
Validate	Validates the select statement. See also Select Statement Validation (page 109)

NOTE Because the data source that has not been normalized is very slow, you need to optimize the data with indices and compute the statistics.

Flat Search: Example

The following input is used in a flat search to find a parcel marking by either point number, marker, surveyor, reliability or dossier.

Input	Value
Name	Parcel Markings
Table Name	LM_BORDERPOINT (This table stores all filter attributes. If they are stored in different tables, you can create a view containing the attributes.)

Attribute 1 Description	Point Number
Attribute 1	BORDER_POINT_NUMBER
Attribute 2 Description	Marker
Attribute 2	ID_BORDER_POINT_TYPE
Attribute 3Description	Dossier
Attribute 3	DOSSIER
Key Attribute	FID
Geometry Attribute	GEOM
Sort Attribute	BORDER_POINT_NUMBER

Position Finder PlugIn Search

Use the Topobase position finder administrator to administer PlugIn search definitions.

To define a PlugIn search

1. Start the Position Finder Administrator.
2. Click Create to define a new search.
3. Enter a name, such as Tiles.
4. For Position Finder Type, select PlugIn Search.
5. On the Settings tab, specify the assembly name, class name, namespace, and parameters, as shown in the following table:

Plug-In Search Settings	Description

Assembly Name	Name of the DLL. The DLL must be available in the <topobase>\bin folder. Example: Topobase.Customers.Autodesk.OSTile.dll
Namespace	Example: Topobase.Customers.Autodesk.OSTile
Classname	Example: OSTile
Parameter Description	To define the prompts in the Advanced Generate Graphic dialog box. Example: Tile Name.

For more information refer to the Topobase Developer's Guide.

Position Finder Tutorial

Defining a Search to Find a Building

Using the position finder you can define search statements that find the center coordinates of an arbitrary object. In the following example, you can find a building that has a certain address.

To find a building, if you know the address

1. Start Topobase Client and open the workspace with the document you want to search.
2. Click Setup menu > Administrator.
3. In the administrator explorer, select the document and click Position Finder Administrator.
4. Click Create.
5. Enter a name, such as Sample Sequential Search.
6. For Position Finder Type, select Sequential Search.
7. Click OK.
8. Define the search with the following settings:

Attribute	On the Statement 1 tab

Title	City
Input Type	Selection list
Select Statement	`Select id, name from City order by name`
Geometry Select Statement	Leave this empty.

Result rows of the first select statement are ID and Name. For example:

- 1, Newcastle
- 2, Bern
- 3, Sidney

If you select ID 1, Newcastle in the combo box, then:

Attribute	On the Statement 2 tab
Title	Street
Input Type	Selection list
Select Statement	`Select id, name from Street where id_city = {0} order by name`
Geometry Select Statement	`Select geom from Street where id = {1}`

The geometry query will find the position of the street, even if you do not select a building.

If you have selected Newcastle, that means {0} = 1.

Result rows of the second select statement are ID and Name. For example:

- 1, Nile Street
- 2, Bathwick Street
- 3, Bennet Street

If you select ID 3, Bennet Street in the combo box, then:

Attribute	On the Statement 3 tab

Title	Building
Input Type	Selection list
Select Statement	`Select id, number from Buildings where id_street = {1} order by number`
Geometry Select Statement	`Select geom from Buildings where id = {2}`

The geometry query uses the result {2} of the select statement.

Result rows are ID and Number. For example:

- 1, 15
- 2, 17
- 3, 17A

Defining a Search to Find an Object

Using the position finder you can define search statements that find the center coordinates of an arbitrary object.

To find an object when you know the feature class and the FID

1. Start the position finder administrator.
2. Click Create and select Sequential.
3. Enter a name, such as Sample Search 2.
4. On the Statement X tabs, enter the following values.

Attribute	On the Statement 1 tab
Title	Feature Class
Input Type	Selection list
Select Statement	`select f_class_name, caption from TB_Dictionary where f_class_type <>`

	`'T' order by f_class_name; (feature class type 'T' = attribute)`
Geometry Select Statement	Leave empty.

Result rows are F_Class_Name and Caption. For example:

- SW_BUILDING Building
- SW_MANHOLE Manhole
- SW_SEWER Sewer

Attribute	On the Statement 2 tab
Title	Feature
Input Type	Text box
Select Statement	`select fid from {0} where fid = {1} order by fid`
Geometry Select Statement	`Select geom from {0} where fid = {1}`

Note that this select statement works with input type Text Box. The select statement uses the value (FID) that you have entered: {1}.

Result rows: FID … 100 … 101 … 102

Defining a Flat Search (Example)

To find a parcel in the Swiss survey data model

1 Create a view to compose the data (locations) in a flat table, as follows:

```
SELECT
d.fid, trim(a.value) district, b.nom commune, c.numero, d.geom
FROM
district_tbd a, commune b, immeuble c, bf_tsur d, bf_tcen e,
bien_fonds f
WHERE
```

```
a.id = b.no_district AND
substr(c.identdn,4,3) =
decode(length(b.numcom),1,'00'||b.numcom,2,'0'||b.numcom,b.num
com) AND
f.fid_bien_fonds_de = c.fid AND
e.fid_centroid = f.fid AND
e.fid_tsur = d.fid
ORDER BY
a.value, b.nom, length(c.numero), c.numero
```

This view can be very complex. Be careful about performance. The order of the input attributes does not matter, but you should have at least one input attribute.

2. In the Administrator explorer, select the document and click Position Finder Administrator.
3. Click Create.
4. Enter a name.
5. For Position Finder Type, select Flat Search.
6. In the Settings tab, enter the following values: (You need to define the different attributes, so Position Finder knows which attributes contain which data.)

Input	Value
Table Name	V_Immeuble
Attribute Description 1	District
Attribute 1	DISTRCIT
Attribute Description 2	Commune
Attribute 2	COMMUNE
Attribute Description 3	Number
Attribute 3	NUMERO
Key Attribute	FID

7 Using the previous definitions, when you start the search, the program will execute the following select statement: (The example is filled with some sample input data).

```
SELECT FID,DISTRICT,COMMUNE,NUMERO
FROM V_IMMEUBLE
WHERE UPPER(DISTRICT) LIKE UPPER('Lau%') AND
UPPER(NUMERO) LIKE UPPER('1002')
```

8 Executed geometry statement for the result row:

```
SELECT GEOM FROM V_IMMEUBLE WHERE FID=20407749
```

Workflow Reference 11

Workflow Introduction

Topobase workflows provide basic functionality for workflows that are used in Topobase applications, such as Water, Wastewater, Gas, Electric, or Land Management. Workflows guide you through tasks like:

- Acquisition
- Analysis
- Reports

Autodesk provides convenient workflows with applications, such as water, gas, and wastewater.

For each workflow that has been started, a separate workflow area is displayed. The workflow area contains embedded information about the further procedures and displays options and check boxes to control the execution of the workflow.

For more information about the use of these workflows, refer to the respective Topobase Application Guides, such as Wastewater User Guide or Water User Guide.

Workflow Definition

You can define workflows for each document (page 521) using the Topobase administrator. Note that to do this, VB .NET and Topobase API (Application Programming Interface) knowledge is required. This knowledge is not presented in this manual.

To define a workflow

1. Start Topobase Administrator and open a workspace (page 526).
2. In the administrator explorer, select the document.
3. Click Document menu ➤ Workflows.
4. The workflow property pane shows a tree view of the defined workflows.
5. In the workflow explorer, select the root node and click Create. Enter a name, such as Network Point Acquisition.
6. In the workflow explorer, select the new workflow. In the right pane, enter the workflow properties.
7. It is recommended that you sort the workflows by categories. To create a category, use Create Workflow. Leave the script code property empty. Then select the category item and click Create Workflow.

Workflow Administrator	Description
Name	Specifies the name of the category or the workflow.
Caption	Specifies the caption of the category or the workflow. The caption is displayed in the document explorer tree. Define a caption that describes the function clearly.
Picture	Specifies an icon for the explorer tree. Click to select a picture. The file has to be either in the <topobase>\pics\Explorer folder or a <topobase>\pics sub-folder. Examples: IconWorkflow.ico: <topobase>\pics\Explorer\ \Dialog\WFStep.jpg: <topobase>\pics\Dialog\
Create	Creates a new category or workflow.
Delete	Deletes a workflow. **WARNING** If you delete a category, all workflows are deleted.
Save	Saves the workflow definitions.

Target Application	Specifies the target application that the workflow definition is designed for. Use this option to mark workflows that have been created especially for the Topobase Web or the Topobase Client version. Depending on this option, the workflow will be available in the Client or Web document explorer.
Script Code	Specifies the script code. Complete the skeleton in the script code window. You can use VB .NET and the Topobase API. Enter “me.” to view valid expressions.
Validate	Checks the syntax.

The definition is stored in the document in the system table TB_WORKFLOW.

The default icons of the workflow explorer tree nodes are assigned in the administrator explorer statement configuration.

See also:

- System Table TB_WORKFLOW (page 102)
- Explorer Configurator (page 103)

Workflow Explorer Configuration

You can provide the workflows, using the Topobase document (page 521), where users can start the workflows. Consider creating an explorer group for all workspaces.

To show a workflow in the document explorer

1. Start Topobase Administrator and open a workspace.
2. Define the workflow.
3. Select the document and click Document menu ➤ Explorer.
4. In the Explorer manager, add the workflow item to the appropriate document explorer definition or create a new explorer group for the workflows.
5. In the explorer tree, check all workflows that you want to provide.
6. Click Save Tree.

Workflow Execution

To start a workflow from the Topobase document explorer.

1. Start Topobase Client and open a workspace.
2. Select the document explorer that contains the workspaces.
3. In the document explorer, expand a workflow category.
4. Select the workflow you want to start, right-click and click Execute.

Depending on the task for which the workflow has been designed, a workflow area is displayed. The workflow area guides you through the selected workflow and displays Help text and options.

For more information about using the application workflow, s refer to the appropriate manual:

- Topobase Water User Guide
- Topobase Wastewater User Guide
- Topobase Gas User Guide
- Topobase Electric User Guide
- Topobase Land Management User Guide

Profiles 12

Overview of Profiles

You use Autodesk Topobase profiles to generate three dimensional views of utility features, for example, in the Wastewater, Water or Gas application.

A profile data model consists of five primary system tables and the corresponding profile feature classes. Profiles are stored in the document (page 521).

Profiles provide a detailed view of the network. You can query network information by selecting a feature in the profile or in the drawing.

Work with profiles

- **Profile data structure**—In Topobase Administrator, you use the data model administrator to add the Profile Extension to your document. The Profile data structure is a system of tables that define the underlying structure of the profile. See Adding the Profile Extension (page 129)
- **Profile designer**—Use the Profile designer to set up your profile definitions. Based on the profile data structure you can create any profile definitions for different profiles. See Profile Designer (page 486).
- **Profile templates**—Use the Profile designer to save a profile definition as template, and to reuse the template for modifications. See also Working With Profile Templates (page 495).
- **Creating and drawing profiles**—In Topobase Client you can use the Profile Manager to create and draw profiles using the profile definitions.

NOTE The Wastewater, Water, and Gas demo data sets provide a sample profile definition.

Profiles can be created for linear objects, such as pipes, by projecting features onto an axis. The axis can be selected in the drawing or it can be defined through network tracing.

Profiles are stored in a local spatial reference system, with 0,0 as the base coordinates. These coordinates can be placed anywhere in the global coordinate system.

In order to create a profile, profile features will be calculated and stored in the document. All features and feature classes can be edited and styled using the Display Manager.

See also:

- Profile Data Model (page 496)

Profile Manager

Profile Designer

Use the Profile designer to set up your profile definitions for use in the Topobase Client Profile Manager. Based on the profile data model you create your profile definitions for different profiles. The Profile designer guides you through the process, and it provides schematic overviews as reference guide for profile elements.

To start the Profile designer

The Profile designer is available in all documents that contain the Profile Extension.

1. Start Topobase Administrator, and open the workspace.
2. Select the document, and click Document menu ➤ Profile Designer.
3. On the Profile designer pane, the profile explorer displays the profile definitions of the selected document.

Use the Profile designer to view and edit the profile system tables, and the profile feature classes.

Profile Explorer

- In the profile explorer, expand the document node, and the Profile node to display the profile definitions.

- Expand a profile definition node to display the associated feature classes.
- Select the profile definition node to display the profile system table TB_PROFILE_DEFINITION in the right pane. See also Profile Data Model (page 496).
- Select an associated table to display table preview in the right pane.

Do not edit the profile feature classes, or the profile system tables manually. You use the Profile designer. When you create a profile definition from scratch, the profile explorer helps you to review the profile feature classes and their relations. See also Profile System Tables (page 497) and Profile Feature Classes (page 506).

Create a Profile Definition

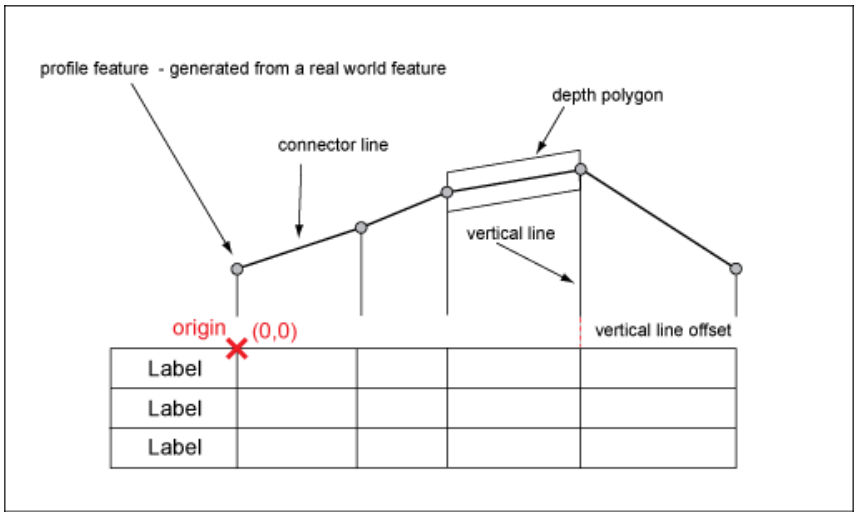
Use the Profile designer to create a profile definition.

1. Start the Profile designer.
2. In the profile explorer, select the Profile node, and click Create.
3. In the Profile Designer dialog box, specify the profile settings. Follow the order that is indicated in the navigation pane on the left.

NOTE When your cursor hovers over an input field for more than 2 seconds, an extended tooltip displays additional information.

Specify the following profile components.

- General Settings (page 488)
- Features Settings (page 489)
- Decoration Settings (page 492)
- Labels Settings (page 494)



Profile feature classes store the profile drawing.

General Settings

In the Profile designer, click the General tab. Under General, you specify profile properties, such as profile origin, scale factor, title, or template.

In Topobase Client, Profile Manager, when you create a profile from the profile definition, you can modify the height scale factor, length scale factor, distance offset, and elevation offset.

Profile Designer dialog box - General settings	Description
Title	Specifies the name of the profile definition.
Template	Selects a profile template. The selected template overrides all fields in all pages of the profile definition with the settings stored in the template. To reset all fields to NULL, select None. See also Working With Profile Templates (page 495).
Height	Specifies the scale factor in height. Ratio between the elevation of real world features in the planimetric view and height

Profile Designer dialog box - General settings	Description
	of profile features in the profile view. For example, set the height as 2 to specify that meter in the real world view is equal to two meters in the profile view.
Length	Specifies the scale factor in length. Ratio between distance in the real world and distance in the profile.
Distance Offset	Specifies the distance offset. Distance from the first digitized axis vertex in the planimetric view that the profile origin x axis is referenced from.
Elevation Offset	Reference altitude. Height that the profile origin is referenced from, this height is adjusted according to the scale factor. When you use the profile definition in Topobase Client, you can adjust this value.

Features Settings

In the Profile designer, click the Features tab, and the sub tabs Lines, Polygons, and Attributes.

Under Features, you map the real world feature classes to the profile details.

The settings are stored in the profile feature class TB_PROFILE_DETAIL.

Profile Designer dialog box - Feature settings	Description
Profile Features	Defines the mapping of the profile features and the real world features. You can add as many mapping definitions as needed.
Detail Title	Specifies the name of the detail, such as Fitting, or Manhole.

Profile Designer dialog box - Feature settings	Description
Main Feature Class	Selects the real world feature class, such as WW_POINT. The list displays all non profile feature classes of the document.
Profile Feature Class	Selects the profile feature class, such as WW_POINT_PRO. The list displays all profile feature classes of the document. See also Profile Feature Class Attributes (page 508).
Distance Attribute	Selects a distance attribute of the selected main feature class on which the profile distances are based.
Where Clause	Defines a WHERE Clause one the selected main feature class that is used as a filter when the corresponding profile feature is generated. For example, to only add network points of type fitting to the profile. Click Assistant to open the SQL Sheet, and click Validate to validate the WHERE clause.
Select Statement	Defines an SQL select statement on the geometry of the selected main feature class to filter profile features according to geometric relationship. For example, to only add points to the profile that are topologically associated with a section. Click Assistant to open the SQL Sheet, and click Validate to validate the WHERE clause.

Under Features - Lines, you define the connector lines, and the vertical lines that can be applied to the profile features.

Feature - Lines settings	Description
Profile Features	Displays the mapping definitions that have been created under Feature Settings. Select a mapping definition, and specify the connector line, or the vertical line.

Feature - Lines settings	Description
Connector Lines	Specifies a line that connects the profile point features.
Feature Class	Selects the profile feature class to store the line. The list displays all profile feature classes of the document. See also Profile Feature Class Attributes (page 508).
Attribute	Selects an attribute of the connector line feature class that can be used for stylization.
Value	Specifies a value for the selected attribute.
Vertical Lines	Specifies a vertical line for the profile features.
Feature Class	Selects the profile feature class to store the vertical line. The list displays all profile feature classes of the document. See also Profile Feature Class Attributes (page 508).
Offset	Specifies the distance (in document units) between the X axis (easting) and the start of the vertical base line.
Attribute	Selects an attribute of the vertical line feature class that can be used for stylization.
Value	Specifies a value for the selected attribute.

Under Features - Polygons, you define the depth polygons. You use depth polygons to add depth related information to your profile.

Feature - Polygon settings	Description
Profile Features	Displays the mapping definitions that have been created under Feature Settings. Select a mapping definition, and specify the polygon.
Depth Polygon	Specifies a polygon.

Feature - Polygon settings	Description
Where Clause	Defines a WHERE Clause on the main feature class. Click Assistant to open the SQL Sheet, and click Validate to validate the WHERE clause.
Feature Class	Selects the profile feature class to store the polygon. The list displays all profile feature classes of the document. See also Profile Feature Class Attributes (page 508).
Depth	Specifies the height (in document units) of the polygon.

Under Features - Attributes, you define up to 5 attribute values for each profile detail.

Feature - Attributes settings	Description
Profile Features	Displays the mapping definitions that have been created under Feature Settings. Select a definition, and specify the attributes.
Feature Class Attributes	
Attribute	Selects an attribute of the selected main feature class.
Value	Specifies a value for the attribute.

Decoration Settings

In the Profile designer, click the Decoration tab. Under Decoration, you specify layout elements such as legends, profile borders, graticules, guide lines, or axis markers.

Profile Designer dialog box - Decoration settings	Description
Decoration Features	Specifies the profile decoration feature classes. You can add as many feature classes as needed.

Profile Designer dialog box - Decoration settings	Description
Detail Title	Specifies a name for the decoration, for example “Static Text”.
Profile Feature Class	Selects the profile feature class to store the decoration features, such as WW_PROFILE_STATIC_TEXT. The list displays all profile feature classes of the document. See also Profile Feature Class Attributes (page 508).
Distance Attribute	
Decoration Feature Setting	
Decoration Type	Specifies the type, such as Single Point, Single Line, or Repeated Horizontal Line. The list depends on the selected profile feature class type. For example, use single elements to represent decorative elements such as legend frames or axis borders. Use repeated elements to represent elements within the profile such as label frames or tables which are comprised of features repeated many times at a prespecified distance.
Interval	For repeated features. Specifies the vertical or horizontal distance (in document units). Enter 0, or a positive or a negative value.
Reference Origin	Specifies the relative position of the decoration elements in the local profile coordinate system.
Start Easting End Easting Start Northing End Northing	Specifies the position of a decoration feature in the profile. If the decoration type is a single element, the start and end coordinates define the absolute position of the feature in the local profile coordinate system.

Profile Designer dialog box - Decoration settings	Description
	If the decoration type is a repeated element, the start and end coordinates define the absolute position of the first feature in the repeated set, and also the reference origin from which the repeated features are placed.
Add Profile Width	Select this option to add the profile width to the start / end easting value. For example, to place the axis border around the profile.
Add Profile Height	Select this option to add the profile height to the start / end northing value. For example, to place the axis border around the profile.

Labels Settings

In the Profile designer, click the Labels tab.

Profile Designer dialog box - Labels settings	Description
Label Features	
Sub Detail Title	Specifies the name of the sub detail, such as Section Label.
Profile Feature Class	Selects the profile label feature class, such as WW_POINT_PRO_TBL. The list displays all profile feature classes of the document. See also Profile Feature Class Attributes (page 508).
Reference Feature	Selects the reference feature (label parent), such as Manhole. The list displays the titles of all profile feature classes of the document.

Profile Designer dialog box - Labels settings	Description
Label Definition	Selects the label definition. The list displays all label definitions of the document.
Reference Origin	Specifies the relative position of the label features in the local profile coordinate system.
Start Easting End Easting Start Northing End Northing	Specifies the position of a label feature in the profile. If you do not specify any coordinates here, the are placed according to the geometry of the reference feature.
Add Profile Width	Select this option to add the profile width to the start / end easting value. For example, to place the profile title.
Add Profile Height	Select this option to add the profile height to the start / end northing value.

Working With Profile Templates

You can save your current profile definition as template that can be reused.

To save a profile definition template

1. Start the Profile designer.
2. In the profile explorer, select the profile definition, and click Edit.
3. In the Profile Designer dialog box, under Template, click Save As.
4. Enter a Template Name, and click OK.

To delete a profile definition template

1. Start the Profile designer.
2. In the profile explorer, select the Profile node, and click Create.
3. In the Profile Designer dialog box, select the template, and click Delete.

The system stores the profile definition templates in the system tables TB_*_TPL.

Profile system tables that store the templates

- TB_PRO_DEFINITION_TPL
- TB_PRO_DETAIL_TPL
- TB_PRO_SUB_DETAIL_TPL

Profile Data Model

Topobase Administrator creates the underlying structure for the profiles that users will create, using their feature classes and attribute data.

The profile data model consists of two sets of tables that are stored in the document.

Profile system tables

Adding the Profile Extension adds the profile system tables to your document. See also Adding the Profile Extension (page 129).

- TB_PROFILE
- TB_PROFILE_AXIS
- TB_PROFILE_DEFINITION
- TB_PROFILE_DETAIL
- TB_PROFILE_SUB_DETAIL

Profile feature classes

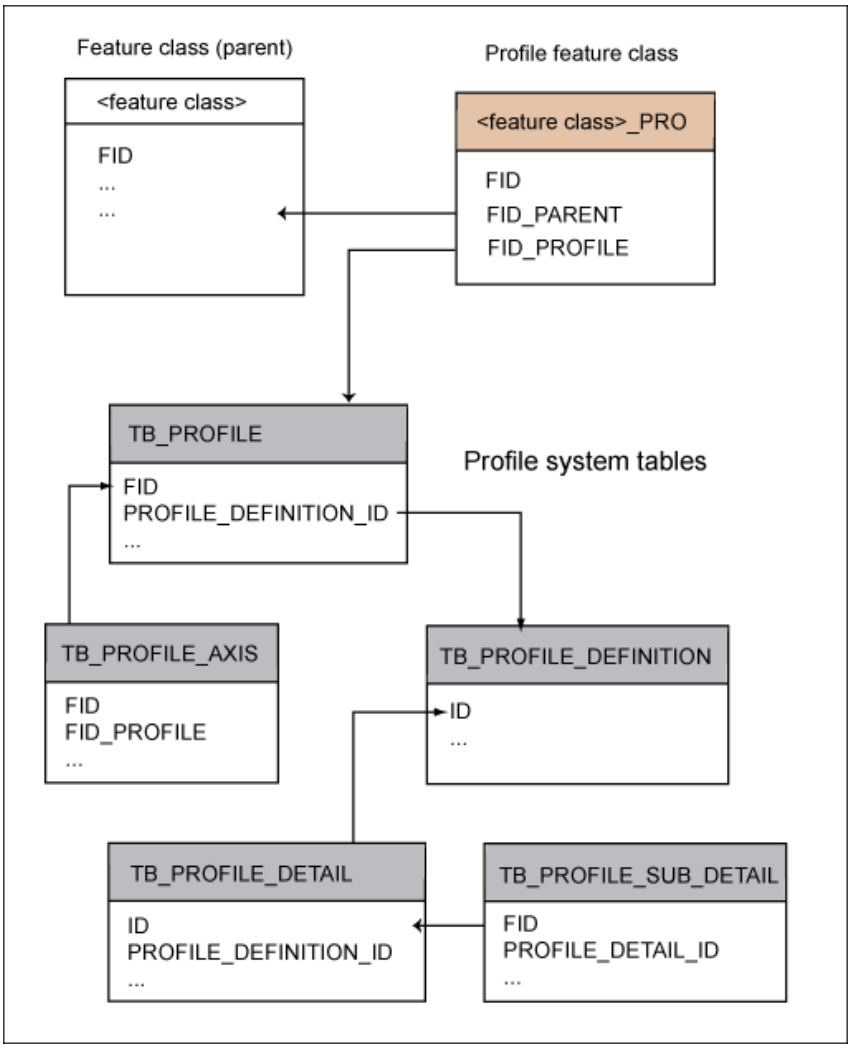
Use the Topobase Data Model Administrator to create your profile feature classes.

- xx_featureclassname_PRO
- The names of the profile feature classes are arbitrary. We recommend that you use the data model code as a prefix and add the suffix "_PRO".
- Mandatory attributes, see Profile Feature Class Attributes (page 508).

Profile System Tables

Profile system tables store the configuration of the profiles in the document, as shown in the following table. You use the Profile Designer to edit the configuration. See Create a Profile Definition (page 487).

Table	Description
TB_PROFILE	Point feature class that stores the origin of a profile.
TB_PROFILE_AXIS	Line string feature class that stores the axis of the profile.
TB_PROFILE_DEFINITION	Stores the profile definition.
TB_PROFILE_DETAIL	Stores profile details.
TB_PROFILE_SUB_DETAIL	Stores profile sub details.



System Table TB_PROFILE

The point feature class TB_PROFILE stores the created profiles. All profile features are related to this table.

When you create a profile in Topobase Client, you are prompted to position the profile in the drawing, which means that you can digitize the origin in the global coordinate system. This origin is stored in TB_PROFILE.

All profile features are defined in a local coordinate system, with the origin of the profile as point 0/0. The width and height of the profile depend on the profile length and height difference in combination with Z_ORIGIN, SCALE_FACTOR_HEIGHT, SCALE_FACTOR_LENGTH.

TB_PROFILE attribute	Description
FID	Primary key.
GEOM	Origin of the profile in the global coordinate system. This is the local 0,0 coordinate.
ORIENTATION	Orientation of the profile.
QUALITY	Quality of the point.
Z	Height of the point. A standard point attribute.
USER_ID	User who creates the profile. Corresponds to TBSYS.TB_USER.ID
PROFILE_TITLE	Title of the profile.
PROFILE_DEFINITION_ID	Profile definition used to create this profile. Related to TB_PROFILE_DEFITION.ID
	NOTE The values for the following attributes are set when you create a profile. If you change a PROFILE_DEFINITION_ID, the default values of the definition are set.
SCALE_FACTOR_HEIGHT	Scale factor for the height, used to define the relationship between the global coordinate system and the profile coordinate system for the height information. NORTHING_PROFILE = (Z - Z_ORIGIN) / SCALE_FACTOR_HEIGHT + GEOM.NORTHING
SCALE_FACTOR_LENGTH	Scale factor for the length, used to define the relationship between the global co-

TB_PROFILE attribute	Description
	ordinate system and the profile coordinate system for the length information EASTING_PROFILE =(LENGTH_GLOBAL - DISTANCE_ORIGIN) / SCALE_FACTOR_LENGTH +GEOM.EASTING
Z_ORIGIN	Specifies the origin of the profile. Reference altitude of the profile. You can enter this value in the Profile Manager dialog box.
DISTANCE_ORIGIN	Specifies the reference distance of the profile. The origin of the profile is at this distance.

System Table TB_PROFILE_AXIS

The line string feature class TB_PROFILE_AXIS stores the axis or alignment of the profiles.

Data represented in the profile is determined by an axis that can be selected by using the logical topology between a start and end point or by digitizing a polyline.

TB_PROFILE_AXIS attribute	Description
FID	Primary key
GEOM	Geometry of the axis
LENGTH	Length of the line string
FID_PROFILE	Foreign key of the profile that has been generated with this axis; related to TB_PROFILE.FID

System Table TB_PROFILE_DEFINITION

The feature class TB_PROFILE_DEFINITION stores the main profile definition.

TB_PROFILE_DEFINITION attribute	Description
ID	Primary key.
USER_ID	User who creates the profile definition. This corresponds to TBSYS.TB_USER.ID.
DEFINITION_TITLE	Title of the profile that you define when you create or edit a profile in the Profile Manager.
ACTIVE	Specifies that the profile definition is available in the Profile Manager: ■ 1 = Yes (default) ■ 0 = No
The following attributes store default values to define the relationship between the global coordinate system and the profile coordinate system for height and length information.	
SCALE_FACTOR_HEIGHT	Default scale factor for the height: NORTHING_PROFILE = (Z_GLOBAL - Z_ORIGIN) / SCALE_FACTOR_HEIGHT + GEOM.NORTHING
SCALE_FACTOR_LENGTH	Default scale factor for the length: EASTING_PROFILE =(LENGTH_GLOBAL - DISTANCE_ORIGIN) / SCALE_FACTOR_LENGTH +GEOM.EASTING
Z_ORIGIN	Default reference altitude of the profile definition. The origin of the profile is at this altitude.
DISTANCE_ORIGIN	Default reference distance of the profile definition. The origin of the profile is at this distance.

System Table TB_PROFILE_DETAIL

The feature class TB_PROFILE_DETAIL stores the definition details of profile lines, profile points, or frame lines.

Each record in TB_PROFILE_DETAIL that corresponds to a feature class will create one profile feature per feature.

Each record in TB_PROFILE_DETAIL that does not correspond to a feature class will create one profile feature only.

For example, the profile detail containing the profile WW_MANHOLE_PRO corresponds to the WW_MANHOLE feature class. For each manhole, one profile feature will be stored. The detail WW_VERTICALLINE_PRO has no relation to the manhole feature class, so this profile detail only defines the one line.

TB_PROFILE_DETAIL attribute	Description
ID	Primary key.
USER_ID	User who creates the profile definition. This corresponds to TBSYS.TB_USER.ID.
DETAIL_TITLE	Title of the profile detail.
PROFILE_DEFINITION_ID	Profile definition related to this detail, related with TB_PROFILE_DEFINITION.ID
F_CLASS_NAME_PROFILE	Name of the feature class containing the features shown in the profile.
F_CLASS_NAME_MAIN	Name of the feature class corresponding to the profile feature class: If F_CLASS_NAME_MAIN is null, only one feature is created as detail. If F_CLASS_NAME_MAIN is not null, one feature per main feature is created as detail.
WHERE_CLAUSE_MAIN	WHERE clause assigned to F_CLASS_NAME_MAIN in order to make a sub-selection of the features. `select * from F_CLASS_NAME_MAIN f where WHERE_CLAUSE_MAIN` The alias f can be used in the WHERE clause.

SELECT_STATEMENT_GEOM_MAIN	Selection of the geometry of the main feature. The query must use the column alias FID, EASTING, NORTHING, HEIGHT. For point feature classes, 0 or 1 row can be returned by this statement. For line feature classes, 0, 1 or more rows can be returned.
FEAT_CONN_LINE_F_CLASS_NAME	Name of the line feature class containing the features used to connect the point features in the profile.
FEAT_CONN_LINE_ATTR1	Attribute name of the line connection feature class
FEAT_CONN_LINE_VALUE_1	Value set for the attribute FEAT_CONN_LINE_ATTR1
VERT_CONN_LINE_F_CLASS_NAME	Name of the line feature class containing the features used to make a vertical line from the feature to the reference height. If F_CLASS_NAME_PROFILE is a point feature class, one vertical line is created. If F_CLASS_NAME_PROFILE is a line feature class, one line is created at the start and one at the end. If two lines are geometrically connected, only one vertical line is created at the connection point.
VERT_CONN_LINE_ATTR1	Attribute name of the line connection feature class.
VERT_CONN_LINE_VALUE_1	Value set for the attribute VERT_CONN_LINE_ATTR1.
VERT_CONN_LINE_ORIGIN_OFFSET	Offset between the reference height and the vertical line. Has the global coordinate system unit.
POLYGON_NEXT_PT_F_CLASS_NAME	Name of the polygon feature class that can only be used for F_CLASS_NAME_PROFILE point feature classes. A polygon is created from the current point feature to the next point feature.
POLYGON_NEXT_PT_HEIGHT	Height of the polygon.
POLYGON_NEXT_PT_WHERE_CLAUSE	WHERE clause used to make a sub selection of the point features. Creates a polygon only if the following statement gives a row back: select 1 from F_CLASS_NAME f

	where POLYGON_NEXT_PT_WHERE_CLAUSE, the alias f can be used in the where clause.
DISTANCE_ATTRIBUTE	Attribute name of F_CLASS_NAME_PROFILE in which the distance along the axis is stored. Only if F_CLASS_NAME_PROFILE is a point feature class.
LOCAL_EASTING_1	First easting coordinate in the local coordinate system.
LOCAL_NORTHING_1	First northing coordinate in the local coordinate system.
LOCAL_EASTING_2	Second easting coordinate in the local coordinate system.
LOCAL_NORTHING_2	Second northing coordinate in the local coordinate system.
LOCAL_EASTING_1_ADD_WIDTH	Flag that defines whether the profile width has to be added to LOCAL_EASTING_1.
LOCAL_NORTHING_1_ADD_HEIGHT	Flag that defines whether the profile height has to be added to LOCAL_NORTHING_1.
LOCAL_EASTING_2_ADD_WIDTH	Flag that defines whether the profile width has to be added to LOCAL_EASTING_2.
LOCAL_NORTHING_2_ADD_HEIGHT	Flag that defines whether the profile height has to be added to LOCAL_NORTHING_2.
REPEATED_DISTANCE	Object is placed every REPEATED_DISTANCE distance; only for point feature class and if F_CLASS_NAME_MAIN is null.
SET_ATTRIBUTE_1 - 5	Attribute name of the table F_CLASS_NAME_PROFILE
SET_VALUE_1 - 5	Value set to the corresponding feature attribute

A LOCAL* value replaces the information of the feature and is considered an absolute value in the profile coordinate system.

If F_CLASS_NAME_MAIN is null and F_CLASS_NAME_PROFILE is a point feature class, then LOCAL_EASTING_1 and LOCAL_NORTHING_1 must be not null and are used as absolute values.

If F_CLASS_NAME_MAIN is null and F_CLASS_NAME_PROFILE is a line feature class, then LOCAL_EASTING_1, LOCAL_NORTHING_1, LOCAL_EASTING_2 and LOCAL_NORTHING_2 must be not null and are used as absolute value.

For line feature class, if a point is not exactly on the line, it is projected to the nearest position of the line.

The relation between the feature of F_CLASS_NAME_PROFILE and FEAT_CONN_LINE_F_CLASS_NAME must be stored in TB_RELATIONS. This relation is automatically filled during the profile generation process.

The relation between the features of F_CLASS_NAME_PROFILE and VERT_CONN_LINE_F_CLASS_NAME must be stored in TB_RELATIONS. This relation is automatically filled during the profile generation process.

The relation between the features of F_CLASS_NAME_PROFILE and POLYGON_NEXT_PT_F_CLASS_NAME must be stored in TB_RELATIONS. This relation is automatically filled during the profile generation process.

System Table TB_PROFILE_SUB_DETAIL

The feature class TB_PROFILE_SUB_DETAIL stores sub-detail definitions. Only labels can be sub-details, all other feature classes are details. Label sub-details correspond to the detail feature classes.

TB_PROFILE_SUB_DETAIL attribute	TB_PROFILE_SUB_DETAIL Description
ID	Primary key
USER_ID	User who creates the sub-detail. This corresponds to TB-SYS.TB_USER.ID.
SUB_DETAIL_TITLE	Title of the sub-detail.
PROFILE_DETAIL_ID	Detail definition related to this sub-detail; related to TB_PROFILE_DETAIL.ID.
F_CLASS_NAME_PROFILE	Name of the feature class containing the label features shown in the profile.
LABEL_DEF_ID	Label definition used to create the label; related to LA-BEL_DEF.ID.

LOCAL_EASTING_1	First easting coordinate in the local coordinate system.
LOCAL_NORTHING_1	First northing coordinate in the local coordinate system.
LOCAL_EASTING_2	Second easting coordinate in the local coordinate system.
LOCAL_NORTHING_2	Second northing coordinate in the local coordinate system.
LOCAL_EASTING_1_ADD_WIDTH	Flag that defines whether the Profile width has to be added to LOCAL_EASTING_1.
LOCAL_NORTHING_1_ADD_HEIGHT	Flag that defines whether the Profile height has to be added to LOCAL_NORTHING_1.
LOCAL_EASTING_2_ADD_WIDTH	Flag that defines whether the Profile width has to be added to LOCAL_EASTING_2.
LOCAL_NORTHING_2_ADD_HEIGHT	Flag that defines whether the Profile height has to be added to LOCAL_NORTHING_2.
SET_ATTRIBUTE_1	Attribute name of the class F_CLASS_NAME_PROFILE.
SET_VALUE_1	Value set to the corresponding feature attribute.

A LOCAL* value replaces the information of the feature and is considered as an absolute value in the profile coordinate system.

One sub-detail feature is created per detail feature. The only exception is for a line profile feature class.

For a line feature class, if a point is not exactly on the line, it is projected to the nearest position of the line.

Profile Feature Classes

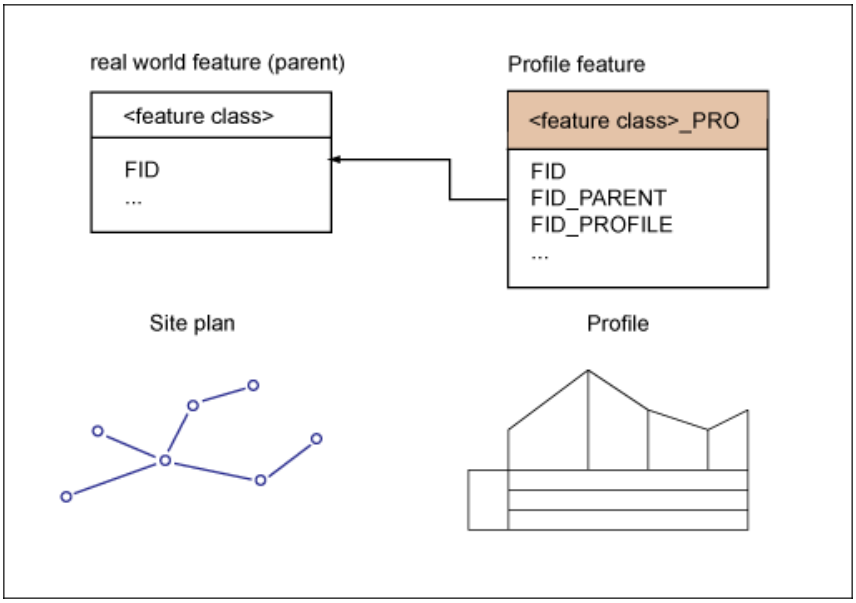
Profile feature classes store the points, lines, and labels for each profile. Profile feature classes represent two types of features:

- A real world feature (parent), such as a manhole cover. Each real world feature corresponds to a profile feature. Real world features are profile objects that are normally displayed in the global reference system. The

profile feature class stores the relation to the real world feature in the attribute FID_PARENT.

- Decoration elements of the profile drawing, such as frame lines, legend, and fix labels. Decoration elements do not correspond to any real world features.

For example, a profile feature of WW_LINE_PRO stores the profile line that corresponds to a real world feature of WW_LINE.



The following table is an example of how a wastewater profile could be stored.

Profile component	Profile feature class (example)	level of detail
Network line	WW_LINE_PRO	detail
Network point	WW_POINT_PRO	detail
Feature label	WW_LINE_PRO_TBL WW_POINT_PRO_TBL	sub detail
Vertical line	WW_PROFILE_LINE_PRO	detail

Profile component	Profile feature class (example)	level of detail
Connector line	WW_PROFILE_LINE_PRO	detail
Symbol such as origin, reference height:	WW_PROFILE_POINT_PRO	detail
Polygon	WW_PROFILE_POLYGON_PRO	detail
Static text such as height, legend labels.	WW_PROFILE_STATIC_TEXT	detail and sub detail

Profile Feature Class Attributes

Use profile feature classes to store the profile elements of the related parent feature classes.

WW__featureclass_PRO Attribute	Description
FID	Primary key.
FID_PARENT	Parent feature class, related with <feature class>.FID; can be null.
FID_PROFILE	Profile, for which the feature has been created, related with TB_PROFILE.FID; not null. This attribute is mandatory. In the Profile Designer, all feature classes that have an attribute FID_PROFILE, are recognized as profile feature class.
GEOM	Geometry of the feature in the profile.
ATTRIBUTE_1	Specific profile feature class attribute. The name of this attribute can be freely defined. ATTRIBUTE_1 is only an example.

For profile points, the height values are taken from the Z attribute. If Z is null, the attribute Z_ORIGIN is used.

For profile lines, the height is obtained from the point on the line or it can be queried with an SQL statement.

Detail profile feature class properties

- Point or line feature class in the topic Profile, containing an attribute FID_PROFILE.
- Point or line feature class in any topic, containing an attribute FID_PROFILE that has a child relation to TB_PROFILE.FID.

Sub Detail feature class properties

- Point or line feature class in the topic Profile, containing an attribute FID_PROFILE.
- Point or line feature class in any topic, containing an attribute FID_PROFILE that has a child relation to TB_PROFILE.FID.
- Label feature class in the topic Profile.

See also:

- System Table TB_PROFILE (page 498)

Profile Manager

In Topobase Client, you can use the Profile Manager to create and draw profiles based on the profile definitions.

NOTE The Wastewater, Water, and Gas demo data sets provide a sample profile definition.

Feature Rules 13

About Feature Rules

Using Topobase feature rules (rule base), you can execute procedures, such as automatic data correction or adding functionality, whenever changes are made to the database.

Feature rules are methods and triggers that process data accordingly to predefined algorithms. For example, rules perform complex consistency and dependency checks at certain events, such as before inserting or after modifying a feature.

Feature rules are provided by Autodesk Topobase. However, users with some experience with PL/SQL triggers can define their own server-side feature rules or to create their own client-side feature rule in .NET.

The feature rules configuration is stored in the document (page 521) in the system tables TB_RULE_*.

Topobase provides client-side and server-side feature rules.

See also:

- Autodesk Topobase Feature Rules
- Server Side Feature Rules
- Client Side Feature Rules
- System Tables TB_RULE* (page 98)

Feature Rule Properties

In Topobase data model administrator, you select the feature class in the data model explorer. Right-click the feature class and click Edit Feature Rules.

In the Feature Rules Properties dialog box you can assign the feature rules, as shown in the following table:

Feature Rule Properties	Description
Feature Class	Name of the feature class
Server Side (PL/SQL) Client Side (.NET)	Feature rules can be server-side or client-side. Click the respective tab. Note, that you can only define new server-side feature rules.
Applied Rule Bases	Lists feature rules that are currently assigned to the feature class. The assigned feature rules are recorded in the system table TB_RULE_BASE. The TB_RULE_FCLASSTYPE table defines which feature rule is available for which feature class type. Example: Calculating the length is valid for LineString feature classes but not for point feature classes.
Edit	Edits the feature rules. Select a row and click the button. See also Edit Feature Rule Properties (page 513)
Priority	Defines the execution order of the feature rules. A feature rule with a lower priority value is executed before a rule with a higher value. Example: It makes sense to validate the geometry of a feature, before calculating the area. Use the Up and Down button to change the order and modify the priority.
Rule Base Definitions	Lists all feature rules (TB_RULE_DEF) and rule groups (TB_RULE_GROUP) that are available for the current feature class. If you select a group, all included rules are added to the Applied Rule Bases list. **NOTE** Only feature rules that are not part of a group are available individually.

Add	Adds a feature rule definition. Select a row and click Add.
Remove	Removes an applied feature rule.
Edit	Edits your own feature rule definitions. Displays the server-side feature rule definitions.
New	Creates a new server-side feature rule definition.
New Group	Creates a feature rule group. See also Feature Rule Group (page 516)
Event	Event on which a feature rule reacts.
BI	Before insert
BU	Before update
BD	Before delete
AI	After insert
AU	After update
AD	After delete

After you have assigned the feature rule, you can modify some properties.

Edit Feature Rule Properties

You can view and edit the properties of an assigned feature rule. Select it in the Applied Rule Bases list and click the Edit button to open the Edit Rule Base dialog box.

Edit the following properties, as shown in the following table:

Property	Description
Active	Activates or deactivates the rule. You can temporarily disable a rule without losing the configuration.

ID	This is the primary key in TB_RULE_BASE. Cannot be edited.
Priority	Modifies the priority. In most cases, you can keep the default priority of the feature rule definition. See also Feature Rule Priorities
Feature Class	Displays the feature class caption.
Definition	Defines server-side feature rules only. Click to view the feature rule definition (trigger text). See also Feature Rule Definition (page 514)
Before Trigger After Trigger	Selects the event at which the rule shall be executed. Modify the regular settings here, if needed. For example, deactivate the perimeter check before an update.
Parameters	Assigns a value to the parameters that are used in the trigger text. For example, most server-side feature rules have a default parameter for the feature class name. See also Feature Rule Parameters (page 517)

Feature Rule Definition

You can define your own server-side feature rules if you know Oracle triggers and the Topobase API. Please refer to the respective manuals.

Feature rules are stored in the system table TB_RULE_DEF in the document (page 521). The following topics cover the configuration process in feature rule definition.

Configuration process

1. In Topobase data model administrator, select a feature class in the data model explorer.
2. Right-click and click Edit Feature Rules
3. Click the Server Side tab.

4 Click the New button next to the Rule Base Definitions area.

In the Feature Rule Definition dialog box, you can define the rules, as shown in the following table:

Feature Rule Definition	Description
Row Trigger	Specifies either selected (1) for row trigger or deselected (0) for statement trigger.
ID	Primary key in TB_RULE_DEF. See also the Topobase Feature Reference, section Feature Rule ID.
Name	Specifies the name of the feature rule. It is recommended that you assign a unique name to each feature rule. Also, add a suffix to indicate the event, for example, add AU for after update. A name for a rule that is part of a group should contain the group name <GROUPNAME>_<ACTION>. For example MoveConnectedLines_BU and MoveConnectedLines_AU.
Description	Specifies a description for the new feature rule. enter a description.
Execute Col:	Specifies when the trigger executes. The trigger executes only when an attribute matching the specified name is modified. If the text box is empty, it will always execute.
Default Priority	Sets the default priority. If assigned to a feature class, this default value can be modified for each feature class. See the Topobase Feature Rule Reference, section Feature Rule Priorities.
Template Values	These primary settings can be modified later for each feature class to which the rule has been assigned.
Parameters	Specifies parameters that can be used in the trigger text. You can use up to 10 parameters. While you are entering the trigger text all used parameters (%P1 - %P10) will be shown and you can enter a name (1st parameter) and a default value (2nd parameter).

Trigger Text	Specifies the trigger text. Enter the trigger text.

NOTE To create client-side feature rules, you need a .NET developer kit and .NET knowledge.

Feature Rule Group

A group can contain several feature rules that define other rules, such as one Before and one After rule.

Server-side rules must adhere to the Oracle trigger limitations. In particular, row triggers cannot query or modify tables in jobs, except for the changed row. Because of this, most server-side rules are implemented as a group of rules. A group of feature rules consists of a before-action rule on the row that inserts the ROWID of modified features into a temporary table, and an after-action statement rule doing the actual work.

You can edit a feature rule group. In the Feature Rules Properties dialog box, select the group in the Feature Rule Definitions list and click the Edit button.

Edit Definition Group Dialog Box	Description
ID	Group ID (TB_RULE_GROUP).
Name	Group name.
Available Definitions	Selects the feature rules that belong to the group. **NOTE** You can only add feature rules that do not yet belong to another rule group. Also, you cannot add the system feature rules to a new customer rule group.
Edit Rule Definition	Displays a rule definition to edit. Note that you can edit only server-side definitions.

You can create new feature rule groups. First, create the new feature rules. Then, in the Feature Rules Properties dialog box, click New Group.

See also:

- Label rule group

Feature Rule Parameters

You can specify up to 10 parameters that can be used in the trigger texts. The parameters are stored in the TB_RULE_PARAM system table.

In the Edit Rule Definition dialog box, you can edit the parameters. To introduce a parameter, enter the parameters %P1, %P2 through %P10 in the trigger text. For each parameter, a text box is displayed, where you can define a name and a default value.

See also:

- System Table TB_RULE_PARAM (page 99)

Assigning Feature Rules

You can assign feature rules in Topobase data model administrator.

To assign a feature rule

1. Select the feature class you want to assign a rule to. Right-click and click Edit Feature Rules.
2. Click either the Server Side (PL/SQL) or Client Side (.net) tab. The upper part of the tab displays the rules that are currently applied. In the lower part. all available rules are displayed.
3. In the lower part, select a rule.
4. Click Add to assign the rule.
5. Click the Edit button next to Applied Rule Bases list to edit the applied feature rule. This is only available for the Server Side rules.

NOTE With the Edit command you can edit the applied feature rule to specify feature class dependent settings.

Exploring Feature Rules

You can view and modify feature rule definitions in Topobase data model administrator.

To explore the feature rules

1. Select a feature class. Right-click and click Edit Feature Rules.
2. Click either the Server Side (PL/SQL) or the Client (.net) tab.
3. In the Applied Rule Bases list, select a rule.
4. Click Edit. In the Edit Rule Base dialog box, click next to the Definition field (only available for server-side feature rules).
5. Click Cancel to close the Edit Rule Definition dialog box.
6. Click Cancel to close the Edit Rule Base dialog box.
7. In the Rule Base Definitions list, select a feature rule.
8. Click Edit to open the Edit Rule Definition dialog box.
9. Click Cancel to close the Edit Rule Definition dialog box.
10. Click Cancel to close the Feature Rules Properties dialog box.

NOTE The Rule Base Definition list only shows the feature rules that are applicable for the selected feature class.

See also:

- Edit Feature Rule Properties (page 513)
- Feature Rule Definition (page 514)

Exploring Feature Rule Groups

You can view feature rule groups and their definitions in Topobase data model administrator.

To explore a feature rule group

1. Select a feature class. Right-click and click Edit Feature Rules.
2. Click the Server Side (PL/SQL) tab.
3. In the Rule Base Definitions list, select a group. Feature rule group names are highlighted.

4 Click Edit. In the Edit Definition Group dialog box select one of the selected feature rules and click Edit Rule Definition.

See also:

- Feature Rule Group (page 516)
- Feature Rule Definition (page 514)

Glossary

area topology Description of spatial relationship between geographic area features. Area topologies contain line strings and centroids. In Topobase™, the polygons are generated automatically from the surrounding line strings. Examples of area topologies are parcels, land use, land cover and political boundaries. See also Topology.

centroid A point that indicates a polygon (approximately in the center). In Topobase, centroids are part of area topologies and belong to the surrounding edges (line string feature class). The centroid normally holds the polygon's attribute data. See also Area Topology.

COGO Abbreviation for Coordinate Geometry. COGO functionality provides calculation routines, such as for intersection, projection, orthogonal survey, offset lines, and right angle course.

Display Manager For stylization in Autodesk Map 3D, applies custom styles to selected features and objects. To view the Display Manager task, select Display Manager in the list at the top of the Task Pane. Also used for stylization of Topobase features.

display model In Topobase, you use display models to administer thematic views. A display model definition specifies which set of layer files (feature layers) is loaded into the Display Manager. Also, the display model defines multi map windows and autoload layers.

document In Topobase, a document is an Oracle® database schema with additional settings in the Topobase System user (database server schema TBSYS). A document is an Oracle database user plus settings for menu bars, toolbars and forms. A document must be assigned to a workspace to be accessible by Topobase Client or Topobase Web. You can create, edit, and configure the documents using the Topobase Administrator.

document explorer Control element in the Topobase Client task pane and in the Topobase Web layout. Use the tree view to show the objects that are stored in the database. For different requirements and more clarity these objects can be grouped into explorer groups. Provides a document-specific

view to process the following objects: Topics (and feature classes), domains, topologies, intersections, system tables, and workflows. You can define a different document explorers for each document. Also called Topobase explorer.

domain Sets of values. For example, a domain defines the values that are allowed for a feature attribute. Topobase data models store domains in domain tables (*****_*TBD*). Domain tables are created using the Topobase data model administrator.

explorer group In Topobase Administrator, a configuration that specifies which objects are to be shown in the document explorer. These settings are saved as Explorer Groups.

feature In Topobase, an entity of a feature class. Each feature in a feature class represents a row or record in the feature class table.

feature class In Topobase, the basic class for objects. For example, a parcel is a feature class. In a database, each feature class corresponds to one Oracle table. A feature class can have any number of attributes (Oracle columns), one of which can be of type "geometry". There are general types of feature classes, such as the following:

- Attribute (feature class without geometry)
- Line String
- Polygon
- Point
- Centroid
- Label
- Compound Polygon
- Compound Linestring

You can group several feature classes for each topic. Each feature class contains many entities/instances or records, which are called features.

feature class form Database form to view and edit attribute data stored in Topobase. Forms can be customized with the Topobase form designer.

feature explorer Control element used to display a set of features in a tree view, resulting from a selection, a validation, or a tracing.

graphic connection A connection between Topobase and Autodesk Map or Autodesk MapGuide to display the features. Topobase Client has a graphic

connection to Autodesk Map. Topobase Web has a graphic connection to Autodesk MapGuide.

job perimeter Spatial area where a job can be processed. You can use job perimeters to control where the modifications of the current job is allowed. Features outside the job perimeter cannot be processed. Also, you can define feature rules to be applied on the objects within the perimeter.

label In Topobase, any attribute data of a feature can be displayed as text, using label features. Label features are generated by arbitrary select statements that can be defined by the customer and therefore are a flexible way to add inscriptions to the objects. Label definitions (select statements and other settings) are stored in the system table TB_LABEL_DEF. Label definitions can be created or edited using the Topobase data model administrator. Label features can be stylized with the Display Manager by displaying the LABEL_TEXT property.

label definition Select statements that create labels. The label definition 1) queries data from the database and 2) specifies positioning and text orientation. This information is used in the Display Manager for stylization.

label feature class Feature class type used to store label features in the database <feature class name>_TBL. Each feature class (parent feature class) can have exactly one label feature class. The label feature class contains default attributes only, and contains no other specific attributes. It stores a relation to the parent feature class.

label placer Places a label to generate a label feature, to write information into the drawing or the map.

logical topology Description of the relationship of features of any feature classes, both attribute or geometry feature classes. The features need not to be spatially connected. For example, a logical topology connects points with points, lines with lines,lines to points, or attribute features to attriute features. Utility networks are based on logical topologies that connect points (nodes) and lines (edges).

For example, a logical topology can represent a waste water network or electrical transmission lines.

master-detail form Type of feature class form where related records are shown in an embedded sub-form on one or more tabs.

network topology See logical topology.

Polygon Object built of line segments that form an enclosed area. In Topobase, polygons are stored in a polygon feature class.

Position Finder Finds the location of a certain object, such as a building, a parcel, or any other type of feature that has geometry. The geometry found will be the center of a graphic generation or a zoom GoTo. There are several types of search such as Sequential search and Flat search.

profile In Topobase, a longitudinal section of line features. A profile is created by projecting features on an axis.

profile data model Data model consisting of a set of profile system tables and an arbitrary set of profile feature classes. Profile system tables store the basic configuration and settings. Profile feature classes store the components of each profile drawing.

prototype drawing Drawing that stores all used blocks, symbols, regular and other lines, text styles and dimensioning styles.

schema plan A schematic diagram that represents real world features by transforming the original feature geometry to an alternative location, for example by applying a coordinate offset. The schema plan represents the real world features in a clear structure, and preserves topology. A schema plan can either be displayed as overlay to the original features, or in a secondary window. You use Topobase Administrator to define schema plans.

schematic feature In Topobase, schematic features are derived from real world features. They are stored in the database and are used to draw the schema plan. Each schematic feature is associated to its real world feature.

TBMAIN Topobase Main User, which is a Topobase System User with restricted rights. The default name for the Topobase Main User is TBMAIN. To start the application, non-administrator users can optionally connect to the Topobase Main User.

TBSYS Topobase System User. Topobase system database schema. Topobase server component that stores application settings and server-side stored procedures. The default name for the Topobase System user is TBSYS. Application users must be connected to the Topobase System user or to the Topobase Main User TBMAIN to start the application.

template In Topobase, a template is an arrangement of recurrent features. The arrangement includes feature attributes, geometry and connectivity. In Topobase Client, you use templates to place a feature arrangement in one single step.

topic In Topobase, a group of several feature classes. A topic is a collection of feature class tables. Topics can be thought of as containers used to organize feature classes. Considering a data transfer, topics are fully independent of one another. Each topic may have sub-topics.

To build a clear and transparent data structure, you can group feature classes into topics, group several topics into main topics, and define feature classes with sub-feature classes. These relations between topics and feature classes serve only as an illustration of the data structure. There is not necessarily an actual relation between the tables.

Topobase Administrator A Topobase basic module, used for Topobase administration. Topobase Administrator contains several components, such as:

- Topobase data model administrator
- Topobase form designer
- Topobase report designer
- Topobase job administrator

Topobase Client A component of Topobase Administrator, used for registering and processing data stored in Topobase through forms and using Autodesk Map for graphic processing.

Topobase data model administrator A Topobase module used to process and establish data structures (topics, tables, attributes, topologies, and intersections) in Topobase. Also, you can define label definitions and dimensioning.

Topobase job All changes in the Topobase data pool can be controlled and performed by exact reports on appropriate processing steps, if they are performed inside a job. Using jobs allows you to control the version. A job includes various processing states (live, pending, project). For each processing state, an application exactly defines which actions are allowed.

Topobase System User Database schema that stores application settings and server-side stored procedures. Default name is TBSYS. Application users must be connected to the Topobase System User to start the application. They can either connect directly to the Topobase System User or use a Topobase Main User which has restricted rights. See also TBSYS and TBMAIN.

Topobase Web The web version of Topobase applications.

topology A geometric shape property in which metrical relations play no role. Topology describes how lines, nodes, and polygons connect and relate to each other, and it forms the basis for spatial analysis as network tracing. In Topobase, topologies can be administered within individual groups of feature classes. They can be defined with the Topobase data model administrator.

topology checker Tool to check topologies. The results are displayed in a tree view, displaying all feature errors.

user A person who works with any Topobase application. Each application user belongs to a group called a user group, with certain tasks and rights. Examples of users: Mr. MILLER, Mr. SMITH, Ms. BAKER; BILLY, TONY, LARRY.

user group A group of users having certain task and rights. Examples of user groups: ADMIN, VIEWER, EDITOR, and so on. A member of a user group can access workspaces only if he has a permission. He may use certain tools and functions and he has certain rights to edit or view the data.

utility model Topobase data model component for utility applications, which provides feature classes and rules for utility networks.

viewport Area of interest for the generate graphic process. A Topobase viewport is a spatial filter on the map.

workflow In Topobase applications, a guide for the user through tasks like acquisition, analysis, and reports.

workspace The central workflow unit in Topobase. For different user groups, such as EDITOR, VIEWER or ADMIN, you can define the appropriate workspaces, with respective roles and rights. A workspace comprises one or more documents.

From the user's point of view, a workspace is the starting point of his work. He must open a workspace to work with any Topobase application. By selecting a workspace, he will load all necessary objects, including the appropriate menus and toolbars, with a single mouse-click. Therefore, he can even access data from different applications, such as land management and wastewater, in arbitrary combinations.

Index

A

access rights
    read only feature class 160
accuracy (point feature class) 182
Add Attributes dialog box 169
Advanced Combobox control 326
Advanced Form Launcher control 327
ALGNMENT
    label text 197
alias
    in label definition select statements 201
    path (web configuration) 75
    path to file name (web configuration) 351
aligned dimensioning 205
analysis
    network - 271
annotation
    see also labels 187
annotation layer 180, 398
application options 16
Application Options dialog box 16
arc linearization (area topology) 259
area topology
    centroid 264
    edge 264
    initialization 268
    introduction 264
Area Topology dialog box 258
areas
    calculating (topology) 268
    label definition to inscribe - 202
    printing reports 455
    see area topology 264
arithmetical calculation
    in forms - example 359
Assistant
    SQL - example 108
attribute
    data types 170
    relation settings 170
auto labels 190
    creating - later 157
AutoGUID 350
AutoIncrement 350

B

Basic ComboBox 328
Basic Form Launcher control 332
buffer
    intersection 228
Build Compounds 214
business rules
    see feature rules 511

C

calculation
    arithmetic - in forms - example 359
    define -s in forms 344
    in forms (example) 359
centroid feature class 174
centroids
    area topology 258, 264
    creating missing 268
chart 124, 444
Chart Wizard 444
Check Box control 334
COGO
    add topic to document 7
COGO features
    delete temporary - 26
Color ComboBox 335
ColorsShema (forms) 336
ComplexReference 353
compound
    feature class attributes 175–176
compound polygon feature class 175
compounds
    building 214

convert to - 213
data model 211
eliminate - 214
configuration
document explorer 103
job state transition 85
job states 84
jobs 296
link to file name (web) 351
link to picture folder (web) 75
polygon and line reports 455
position finder 463
search definitions 463
Topobase Web 63
topologies 256
user groups 12
connection radius
logical topology 260
connection tolerance radius (logical topology) 261
connextion tolerance
logical topology 260
construction features
delete temporary - 26
context help
assigning context - to forms 377
controls
creating form - (form designer) 370
Convert to Compound 213
coordinate system 150
updating SRID 151
coordinates
displaying - of utility features 372
displaying feature - 369
show - in a form 337
Coordinates Editor control 337
cost functions (tracing) 276
Create Feature Class dialog box 160
Create Generalization dialog box 223
Create Intersection dialog box 226
Create New Document dialog box 144
Create Schematics dialog box 218
credentials
Topobase login 2
customizing
date format in forms 349
customizing Administrator toolbar 14
customizing Client menus 15
customizing Client toolbar 14
customizing Electric explorer 29
customizing form menu bar 33
customizing form toolbar 33
customizing form toolbars 374
customizing forms 311
displaying a select list 350
specifying a default value 350
customizing reports 425
customizing shortcut menus (document explorer) 30, 32
customizing TB commands 32
customizing the document explorer 67
customizing user interface (document) 29
cycle detection (tracing) 274

D

data base schema
view name 25
Data Checker 37
data model 35
add to document 7
administrator 36
compounds 211
creating 144
delete feature class attribute 172
dimensioning 205
exploring 118
labels 187
modify attribute property 171
see also documents 36
template 36, 145
utility 287
data model administrator
introduction 143
data model code 37, 73
add to document 7
data model template 145
view code 25
data model explorer 152
data transfer 239

database
optimization (DBMS_STATS) 154
date
customizing the format 349
form designer - default value 350
decimals
number of - in a report 441
default
attributes (feature class) 167
forms (form designer) 319
job states 308
label orientation 195
setting - values (form designer) 350
unit settings 146
delete
display -d features (jobs) 412
DELETE_CHILD
TB_RELATIONS 94
DetailDocument
form designer 332
Digitizing Options 162
dimensioning
creating 127
data model 205
defining 127
feature classes 177
line types 178
point types 178
types 205
types (codes) 177
Dimensioning
add topic to document 7
Display Manager 387
overview 387
display model
creating - 383
Display Settings
feature explorer 165
document
administration and clean up 153
creating 144
transfer data between two -s 239
document explorer
configuration 67, 103
customizing shortcut menus 30, 32
Document Linker control 338
Document Manager control 338
document options 24
export 11
import 11
documents
creating 111
creating - from template 36
import - into workspace 55
view data model codes 25
view name of data base schema 25
view version 25
domain
creating - tables 183
displaying - values in a combo box 328
editing - values 183
drawing templates
assigning to a workspace 116
dropdown list
add related - (form designer) 326
dump
import - 55
DWG
export 381
dynamic relation 342
Dynamic Stylization 408

E

edge
area topology 264
logical topology 269
Edit Attributes dialog box 171
Edit Rule Base dialog box 513
Editable SQL TextBox control 339
editing
domain values 183
Electric CE
schema plan settings 220
Electric explorer
customizing commands 29
profiles 70
elevation
calculation - wastewater 28
Eliminate Compound 214
Enable Jobs dialog box 299

enabling jobs 299
Enhanced Styles 387
export
  ASCII text files 454
  document 11
  document into Oracle dump file 56
  document options 11
  DWG 381
  line definitions 455
extension
  add to document 7
  template 216

F

feature class
  adding attributes 167
  creating 160
  creating a label feature class 113
  geometry attribute 172
  label - 187
  make - read only 160, 163
  model table 184
  naming limitations 160
  relations 94
  types 172
feature explorer
  customizing 165
  zoom factor for highlight 17
Feature Highlighter control 341
feature locking 301
  settings 148
feature redirection 77
  examples 77
Feature Rule Definition dialog box 515
feature rule group
  view definitions 518
feature rules
  assigning - 517
  data model administrator 158
  defining 514
  exploring - 517
  group 516
  introduction 511
  parameters 517
  see business rules 110
  system table 98
  utility model 290
  viewing 512
  viewing and editing 110
Feature Rules Properties dialog box 512
feature styles
  saving 390
features
  bounded highlight settings 75
  rendering 111
FID Button control 343
FID Form Launcher control 342
FID Linker control 343
FID Reference control 342
FilterComboBox
  example 350
flow direction management 261
flyIns
  document options 34
  view available 34
form
  assigning toolbars 316
form designer
  assigning toolbar 316
  AutoGUID 350
  AutoIncrement 350
  calculate values in a form 344
  controls 318
  customizing forms 311
  defining form toolbars 374
  defining multiple tab pages 374
  displaying a combo box 328
  displaying a RGB color button 336
  drawing lines (layout) 345
  dynamic relation 342
  editing tab pages 318
  enabling highlight for features 328
  general control properties 324
  hiding controls 313
  hiding controls for user groups 324
  inscribe dynamic editable text 339
  inscribe dynamic text (SQL) 351
  inscribe static text 356
  introduction 309
  link to picture folder (web) 75
  locking controls for user groups 324

open a second form 353
related dropdown lists 326
selecting a color from a list 335
show coordinates 337
starting 309
toolbar 320
yes/no check box 334
forms
customizing 311
double click opens the - 22
import/export 11
locking input fields 349
master-detail 95–96
specify mandatory fields 349
specify validation for input fields 349
formula
-control example 359
Formula control 344

G

generalization 216
GEOM 172
Geometry control 337
grid cross settings 422
GUID
form designer - Auto- 350

H

hard split 285
help
assigning context - to forms 377
Highlight Button control 341
history
erasing job - 155
see jobs 295
history (job version) 81

I

import
document into a new workspace 55
document options 11
dump (document) into workspace 55
report definition 454
initialization
area topology 268
logical topology 277
input type
position finder 466
insertion point (label) 197
intersection
adjusting areas 228
creating 226
feature class 225
introduction 224
meta data 79
results 225
types 224
invalid job configuration 297

J

job administrator 298
job conflict 301
Job Enabled 299
job state
defaults 84
introduction 308
transition options 27
transition state configuration 85
job state configuration 84
job transition 308
job version 81
jobs
assigning perimeter feature class 299
creating templates 304
enabling 299
erasing job history 155
feature locking 301
getting started 296
introduction 295
invalid configuration 297
lease options 21
leasing 305
perimeter 17, 302
perimeter settings 148
repair and update configuration 297

selecting feature classes 306
settings 148
states 295
style modified and new features 412
template 298, 306

L

Label control (form designer) 356
label definition
content to display 190
creating 113
example 201
inscribe area 202
inscribing utility attributes 292
parent geometry select statement 202
rules 201
scale factor select statement 198
select statements 200
SQL format commands 202
label feature class
creating 113
label features
display orientation 397
display prefix and suffix 398
style - 393
label ID
data model administrator 199
Label Properties dialog box 189
Label Regeneration Definition dialog box 191
labels
creating 113
data model 187
defining the textbox insertion point 197
feature class attributes 180
horizontal alignment 197
insertion point 195
orientation 195
origin 197
origin conversion 199
properties 189
redirecting to parent feature 17
regeneration rules 191
update label text 155, 204
vertical alignment 197
LDF files
loading 390
saving 390
Lease Job dialog box 305
lease options (jobs) 21
leasing jobs 305
license
assigning 9
Line control 345
line definition
configuring a report 455
line style
using Topobase -s 393
lines
drawing - in a form (layout) 345
splitting 285
loading
feature styles 390
location 216
locking input fields 349
logical topology
connection radius 260
initialization 277
introduction 269
Logical Topology dialog box 260
login
changing password 51
credentials 2
default user 51
document password 4
Topobase Administrator 2
Topobase Client 1, 5
long transaction
see jobs 295

M

Manage Document Password dialog box 4
mandatory
specify - fields 349
maps
styling 387
Master Box control 326

master-detail 95–96
properties (form designer) 317
Master-Detail dialog box 317
menu
customizing form shortcut -s 376
menus
customizing form - 33
migration method
TB2TB 248
minimum spanning tree (network tracing) 274
missing centroids 268
model driven attribute 184
Model selector (form designer) 356
model table 184
module
add to document 7
creating a - from template 145
Multi Reference control 327
multi-relation
reference 327
use with combo box controls 331
multi-selection
form designer 315
multi-user
read only feature class 160
multiple
coordinate systems 216
location 216
multirelation
data model administrator 170

N

network analysis 271
network topology
see logical topology 285
Network Tracer dialog box 271
node
area topology 256
logical topology 269

O

optimistic feature locking 297, 301
options
application - 16
electric 28
form designer 316
job topology - 27
setting document - 24
wastewater - 28
Oracle
convert Oracle schema 235
view name of data base schema 25
Oracle dump
exporting a document 11
orientation
default setting 146
define default - for labels 195
enter when digitizing 161–162
orthogonal dimensioning 205

P

parent geometry (select statement)
example (utility model) 293
password 2, 4–5
job transition 298
user login 51
pending
jobs state 295
perimeter
check job - 303
display options 17
feature class 302
job - 302
perimeter check
enabling 304
perimeter feature class 162
pessimistic feature locking 301
pessimistic locking 148
Picture Box control 75
Picture ComboBox control 346
Picture Linker control 347
pictures
display static - in a form 355
use in form designer and report designer 75
picures
display record-related - 347

Plot
add to document 7
plots
creating with AutoCAD blocks 136
point number 182
automatic generation 42
points
feature class attributes 181
feature class optional attributes 182
polygon feature class 182
polygons
compound - 175
printing reports 455
see also area topology 264
position finder 463
symbol 63
primary feature class (intersection) 226
printing
page setup 433
reports in Web version 433
profile
Electric explorer container 70
profile definition
creating 128
profiles
add to data model 129
data model 496
introduction 485
projection
displaying a - button 327, 332
propagation
see tracing 271
Property TextBox control 349
prototype drawing 391

Q

quality
point measurement 182

R

reachability (topology tracing) 274
read only
feature class 163
Read-Only SQL TextBox control 351
Read-Only SQL TextBox example 362
reference button
displaying a - 327, 332
Reference control 332
Reference Options dialog box 332
relation type 170
relation wizard 105
reliability (point
feature class) 182
rendering features 111
Repair job configuration 297
report definition
adding a chart 124
creating 121
creating a head 124
creating a master-detail report 126
import 454
number of decimals 441
report designer
adding layout controls 429
charts 444
creating a report (wizard) 434
example- call via API 460
example- change colors 460
getting started 427
handling see also form designer 427
introduction 425
link to picture folder (web) 75
options 433
printer format 433
report head 124
reports
creating (report desiger wizard) 434
defining 425
designing layout 425
number of decimals 441
with spatial selection 456
representation 216
right angle course
setting the mode 19
rule base
see feature rules 158, 511

S

saving feature styles 390

scale factor select statement
label with - 198
schema
convert Oracle schema 235
schema converter 235
schema plan 216
Script Button control 353
Script Programmable control 353
scripts
assigning to a form 353
search
defining a flat search (example) 478
defining a flat search (position finder) 472
defining a sequential search (example) 475, 477
defining a sequential search (position finder) 468
position finder 463
types (position finder) 463
secondary feature class (intersection) 226
Set User Group For Job Template dialog box 298
shortest path (topology tracing) 274
snap radius
logical topology 260
soft split 285
spatial intialization
logical topology 277
spatial operator SDO_RELATE 233
spatial reference ID 147
spatial relationship settings 233
spatial selection
enable - for reports 456
spatial tolerance 147
split lines
feature rules 290
soft and hard split 285
SQL assistant 103
SQL Error Spy 132
SQL Form Launcher 353
SQL Info dialog box 109
SQL Label control 351
SQL label example 362
SQL protocol 156
SQL Sheet 130
SQL Spy 132
SQL Text Box control 339
SRID 147, 150
updating 151
Static PictureBox control 355
Static Text control 356
statistics
optimizing Topobase schema 155
stop condition
tracing 276
string editor 165
styled maps 387
styling
label features (Topobase) 393
lines 393
maps 387
symbol
assigning a - for position finder 63
symbol library
prototype drawing 391

T

tab order
defining the - (form designer) 357
Task Pane
Display Manager overview 387
TB_COLOR_ACAD 336
TB_COLOR_WEB 336
TB_DATAMODELCODE(S) 73
TB_GN_DIALOG_HIGHLIGHT 75
TB_GN_INFO_REDIRECT 77
TB_INFO 78
TB_LABEL 88
TB_LABEL_DEF 88
TB_POINTNUMBER 42
TB_POSITION_FINDER 92, 463
TB_RELATIONS 94
TB_RELATIONS.DELETE_CHILD 170
TB_SQL 408
TB_VAR 408
TB_VARIABLE 408
TB2TB 239
migration method 248
TBEXPORTDWG 381

TBSYS
view version 25
template
creating a job - 304
feature arrangement 216
job - 306
Templates
add to document 7
terrain elevation 182
text
assembling contents of -attibutes 362
TextBox control 349
tolerance
arc linearization (topology) 259
area topology 258
connection tolerance radius (topology) 261
intersection 228
setting the split - 19
spatial - 147
topology 259, 261
toolbar
assigning form - 316
customizing Administrator - 14
customizing Client - 14
customizing form -s 374
document main - 29
topic 35
creating 156
template 216
topology
creating 256
flow direction 261
getting started 256
initialization 215
introduction 255
node 256
overlapping lines 259
system tables 101
tracing 271
conditions 276
cost functions 276
defining - conditions 275
defining conditions 272, 282
defining templates 282
stop condition 276
templates 272
types 274
visualize results 18
tracing result
configuring the feature explorer 165
Tracing Result dialog box 271
Tracing Template dialog box 272
trigger
compounds 212
feature rule 511

U

unit support
reports 451
set attribute units 169
settings 146
Update Coordinate System dialog box 151
user admininstration 13
user groups
adding 13
concept 52
creating 13
removing 13
user interface
customize Administrator toolbar 14
customize Client toolbar 14
customizing Client menus 15
customizing form shortcut menus 376
defining form toolbars 374
hiding form controls for user groups 324
locking form controls for user groups 324
toolbar for feature class form 316
users
adding 13
creating 13
removing 13
utility model
creating 288, 291
data model 287
feature rules 290

introduction 285

V

validation
defining - code for data acquisition 358
version
document structure 25
TBSYS (topobase.jar) 25
versioning
feature locking 301
see jobs 295
vicinity 17
view
registration 164
using an Oracle - in a document 164
viewport
default size 22

W

wastewater options 28
web
position finder symbol 63
web layout
assigning a workspace 63
workflows
definition 481
introduction 481
show - in document explorer 483
system tables 102
workspace manager 53
workspaces
adding - to a document 53
assigning a web layout 63
creating 53, 116
setting up - (concept) 50

X

XYZ
show - values in a form 337

Z

Z value
enter when digitizing 162
zoom
factor for feature highlight 17