Autodesk® Topobase™ Land Management User Guide

Autodesk® Topobase™ Land Management User Guide

© **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 McInnis Parkway
San Rafael, CA 94903, USA

Contents

Land Management 1

Starting Topobase Land Management

When you process Land Management documents, you use Topobase Jobs. Using jobs you clearly separate data that is valid and approved (Live job) from data that is being processed (Open job) and data that is subject to an approval process (Pending job).

See Working With Jobs (page 36).

To start Land Management

1 Start Topobase Client, and open the workspace.

2 Click Display panel ➤ Generate Graphic.

3 Select a job.

Use the Document Explorer to view and edit feature classes.

Use the Workflow Explorer to start the acquisition workflows. See also Using Land Management Workflows (page 33)

Use the Job Explorer to process the features you have created in the current job.

Land Management Reports

The Land Switzerland data model provides report definitions, for example a boundary points report, or a property description report. You use the feature class forms to filter the data, and to print the report.

To print a report

1. Open the feature class form, such as Property.
2. Filter the data you want to print.
3. In the form toolbar click the Print icon.
4. In the Report dialog box, select a report definition.
5. See also Printing Reports

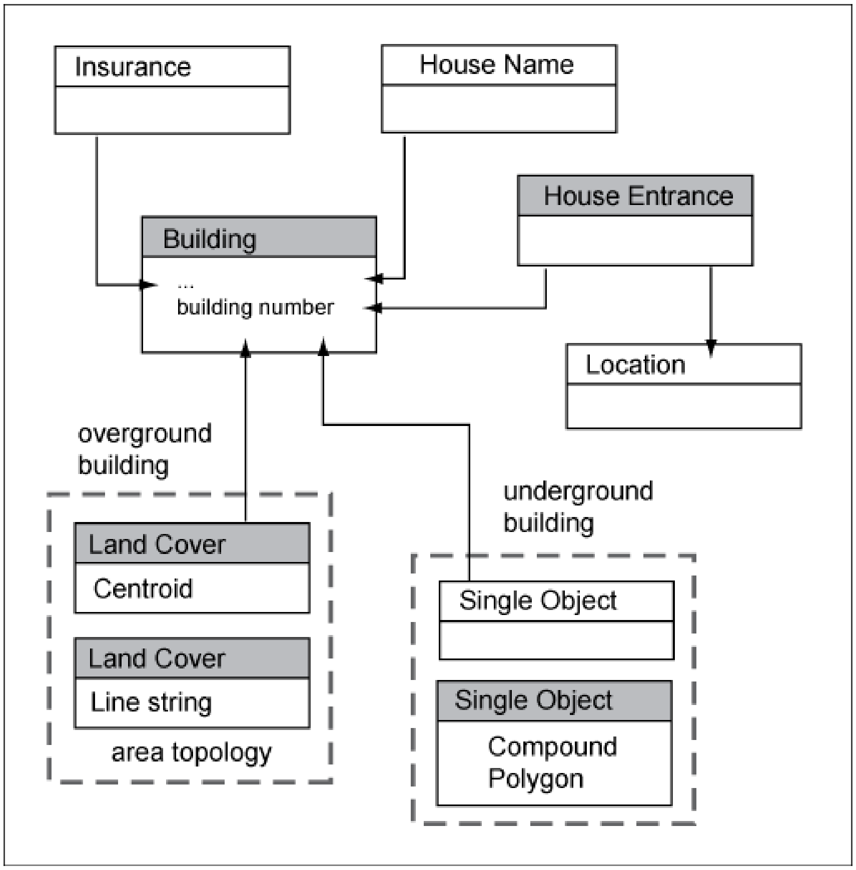
See also:

- Spatial Export
- Coordinate Report

Managing Buildings

Topobase Land Management (Switzerland) provides feature classes to store building information compliant with the Swiss federal norm.

The data model contains one parent feature class, and several related feature classes to store the building information. The parent feature class (LM_BU_BUILDING) stores attributes, and relations to multiple feature classes that store additional information, such as the geometry, building insurance, and house entrance.



Building information consists of the following elements.

- **Building point**—Parent feature. Stores a point within the building. The Swiss standards require that you manage the buildings outside the Land Management module. You can use the building point to attach building information, such as insurance and house entrances. You do not need the building geometry.
- **Building geometry**—In the map, a building is represented by closed polygon. In the database, the building geometry is stored in different feature classes, depending on whether the building is overground or underground.
- **Building insurance**—Insurance data is stored in a related attribute feature class.

- **House entrance**—House entrances are stored in a related point feature class. A house entrance is related to the location. It represents the address of the building in the map.
- Other building attributes, such as name, description.

See also:

- Create Building (page 51)
- Land Management: Building (page 69)
- Managing Land Cover (page 4)

Buildings

Overground buildings—Stored in the Land Cover topic. The geometry of an overground building is stored in the area topology feature classes

- LM_LC_SURFACE—Centroid that stores attribute data, such as building type, and the relation to the parent building.
- LM_LC_SURFACE_L—Lines.

Overground buildings are part of the Land Cover area topology that provides consistent and non overlapping land cover areas. When you digitize a building, the topology requirements must be met, such as no overlapping lines, exact snap of line endpoints, building lines must build a closed polygon.

Underground buildings—Stored in the Single Object topic. The geometry of an underground building is stored in a compound polygon feature class, that means you can style the lines using different line styles. The Single Object parent feature stores the attribute data, such as building type, and the relation to the parent building.

Managing Land Cover

The Land Management data model uses an area topology to provide consistent and non overlapping land cover units. For example, forests, water features, or asphalt areas.

When you digitize land coverage, the topology requirements must be met, such as no overlapping lines, exact snap of line endpoints, exact polygons.

A complete land cover unit consists of

- a centroid feature that stores the attributes.
- line string features that determine the land cover boundary.

Use the Create Land Cover workflow to create land cover features consistently with the data model. See Create Land Cover (page 43).

See also:

- Land Management Data Model (page 63)

Managing Ownership

Topobase Land Management (Switzerland) provides feature classes to store ownership information compliant with the Swiss federal norm.

Ownership information consists of the following elements.

- **Boundary Points**—See Managing Boundary Points (page 6).
- **Properties**—Properties are real estates, mines, and development rights (DPR). The data model provides a Property parent feature class (LM_OW_PROPERTY) that stores basic attributes. One property can consist of several real estates, or DPRs, or mines. In that case, the Property parent feature stores the sum of all partial areas. DPR and mines can overlap the real estates.
  Use the Create Properties workflow to create property features consistently with the data model.
- **Real Estate**—The data model uses an area topology to provide consistent and non overlapping real estates. The real estate centroid stores attribute data, and the relation to the Property parent feature. The real estate boundary lines are determined by Boundary Points, or by Control Points. That means that you cannot create a real estate boundary line without these points. See Create Property: Real Estate (page 40)
- **Development right DPR**—Compound polygon feature. The parent polygon stores the relation to the Property parent feature. See Create Property: DPR (page 42).
- **Mine**—Compound polygon feature. The parent polygon stores the relation to the Property parent feature. See Create Property: Mine (page 41).

For projected properties, the data model provides separate feature classes.

See also:

- Split Real Estate (page 58)
- Merge Real Estates (page 59)

Managing Boundary Points

Boundary points are well determined points that specify the position of boundaries, for example of real estates, cantons, or districts.

For example, you calculate boundary points using the Topobase Survey module, the Topobase COGO commands, or you import boundary points from a file. See the Topobase Survey User Guide.

Projected Boundary Points

Topobase provides rules and workflows to manage projected boundary points. Projected boundary points require a special way to check precision and reliability. Usually you measure the boundary points several times, and you determine the coordinates in an adjustment calculation.

For projected boundary points, you calculate the coordinates in advance, then you export the coordinates to your tachymeter, you transfer the points to the real world (tracing), and you measure the points to control whether the points in the field are as designed. If the measurements confirm that the points are as designed, the initially calculated coordinates are not modified, however the points will flagged as Controlled. During the process, the projected point geometry must be protected from being modified.

To manage projected boundary points, you perform the following steps.

- You calculate the coordinates of the projected boundary points LM_OW_BOUNDARY_POINTS, for example, using COGO functions. You assign the point mark attribute ID_POINT_MARK = Projected Point (ID 14). These points cannot be moved.
- You transfer the points to the field (tracing), and measure the points to control the tracing.
- You process the field measurements using Topobase Survey. You calculate the point coordinates from the field measurements. In the Survey distribution step, the points are compared to the projected points already

stored in the database. If the coordinates of identical points lie within the tolerance, the point are marked as Controlled. However the geometry will not be modified.

Point Numbering

See Automatic point numbering (page 12)

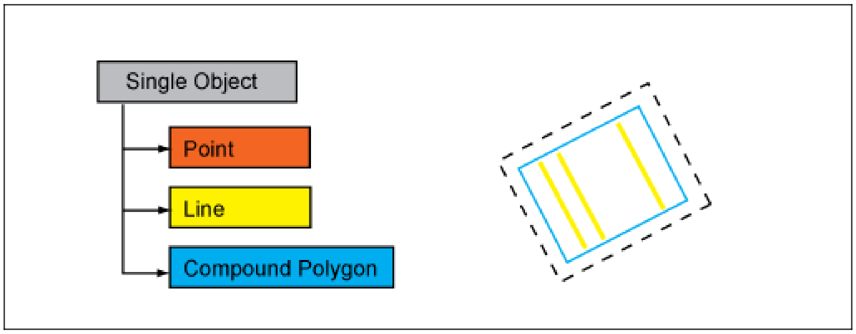
Point Reports

Use the Coordinate Import / Export tool to print a coordinate report. See also Coordinate Report.

Managing Single Objects

Topobase Land Management (Switzerland) provides feature classes to store single objects compliant with the Swiss federal norm. For example, trees, bridges, green houses.

Single objects consist of one parent feature that stores the attributes, and one or more geometry elements. Depending on the type, a single object can be of different geometry type, such as point, line string, or compound polygon. For example, a tree is of geometry type Point. A green house can be of geometry type line string, or compound polygon.



Use the Create Single Object workflow to create single objects consistently with the data model. See Create Single Object (page 45).

Managing Names

Topobase Land Management (Switzerland) provides feature classes to store names compliant with the Swiss federal norm.

- **Local Name**—Local names are associated to an area. Local name areas are part of the Local Name area topology that provides consistent and non overlapping local name areas.
- **Locality**—Name that is not associated to an area.
- **Place**—Name that is associated to a boundary polygon.

Use the Create Name workflow to create names consistently with the data model. See Create Name (page 47).

Managing Locations

Topobase Land Management (Switzerland) provides feature classes to store locations compliant with the Swiss federal norm. For example, roads and places.

A location is stored in a parent feature that stores the attributes, and the relations to the associated geometry element. Depending on the type, a location is associated to different geometry elements.

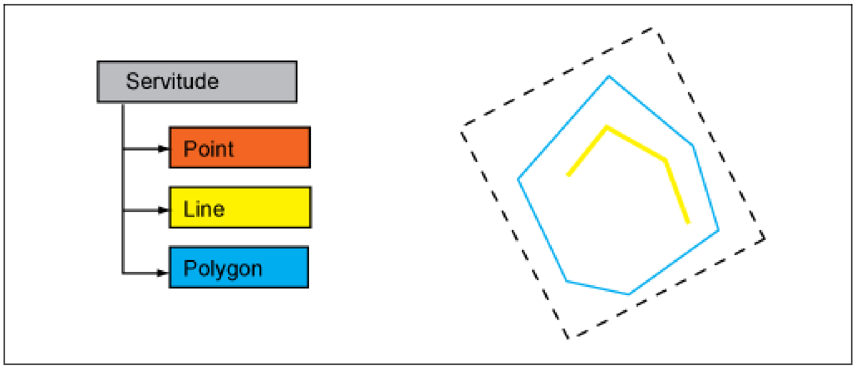
- Named area: Polygon.
- Road sections: Line string.
- Way: Line string.
- Road classification: Polygon.

Use the Create Location workflow to create locations consistently with the data model. See Create Location (page 49).

Managing Servitudes

Topobase Land Management (Switzerland) provides feature classes to store servitudes compliant with the Swiss federal norm. For example, access rights, or third party access.

A servitude is stored in a parent feature that stores the attributes, and the relations to the associated geometry elements.



Servitudes are associated with a geographic location. One servitude can be associated with multiple geometry elements, for example, to display the feature in the map.

Use the Create Servitude workflow to create locations consistently with the data model. See Create Servitude (page 56).

Land Management: Project Settings

2

Land Management: Project Settings

This section describes project settings that control the Land Management application, such as Job settings, and point numbering.

Land Management: Job Settings

When you modify any feature of the Land data base, you use Topobase Jobs. Using jobs you clearly separate data that is valid and approved (Live job) from data that is being processed (Open job) and data that is subject to approval processes (Pending job).

Your Topobase administrator provides the job templates and configures the perimeter feature classes.

Job settings for Land Switzerland

- Job enabled document; using Optimistic Feature Locking. See also Jobs Feature Locking.
- Perimeter: The feature class Mutperimeter (LM_AD_MUTPERIMETER) is the perimeter for all feature classes.
- Job Templates: For example, you define a template that includes the topics Ownership and the Control Points, a template that includes all topics, and a template that includes all topics except Ownership and Control Points. See also Creating a Job Template (Job Administrator).

- Job Templates: The feature class LM_AD_MUTPERIMETER must be part of every job template.
- Job Document options: Use the Jobs With Topologies option to specify at which transitions a topology check shall be executed. See Setting Job Topology Document Options.
- Map Application Options: Use the Perimeter Display Options to specify how the perimeters are displayed in the map. See Setting Map Options.
- Intersections: See Land Management: Intersections (page 14).

See also:

- Working With Jobs (page 36)
- Using Land Management Workflows (page 33)
- Jobs Introduction

Automatic point numbering

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 in your document.

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. See Topobase Point Numbering

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

See also:

- Topobase Point Numbering
- Point Numbering Per Raster Plan

Land Management: Display Model Settings

When you modify any feature of the Land data base, you need to document the modification, so it can be approved by the responsible authority. Usually, you document the changes of your current job in a map that highlights the new, the modified, and the deleted features.

To display the changes in the map, you use an appropriate Display Model. Topobase provides a Display model for the Land Management (CH) demo data set.

To create a Display Model that tracks feature 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 Boundary Point (LM_OW_BOUNDARY_POINT).
2. Click Style.
3. In the Style Editor, under Point 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 Topobase text function TB_SQL see the Topobase Administrator Guide.

To style line features that have been split or joined

- Use the SYSTEM_CREATED attribute. See also Real Estate and Real Estate L (page 96).

Land Management: Intersections

The Land Switzerland data model provides predefined intersections that are executed automatically when the job state changes. For example, intersections between real estate areas and road sections, or real estate areas and local name areas.

Land Management intersections are used to provide a property description report. Swiss land surveyors need to deliver this report for the live state and for the pending state of a real estate.

Use the data model administrator to view the intersection definitions. See

To view intersection definitions

1 Start Topobase Administrator and open the workspace.

2 Click Document menu ➤ Data Model.

3 In the data model explorer, expand the Intersections node.

4 Select an intersection, right-click, and click Properties.

For more information about intersections, see the Topobase Administrator Guide, section Data Model: Intersections

Land Management Data Exchange

3

Land Management Data Exchange

This section describes the Topobase Land Management data exchange tools: Interlis Import, Interlis Export, Coordinate Import, and Coordinate Export.

Interlis Import

Interlis is a standard data exchange description language that is mainly used in Switzerland. An Interlis data set consists of two files.

Interlis data set

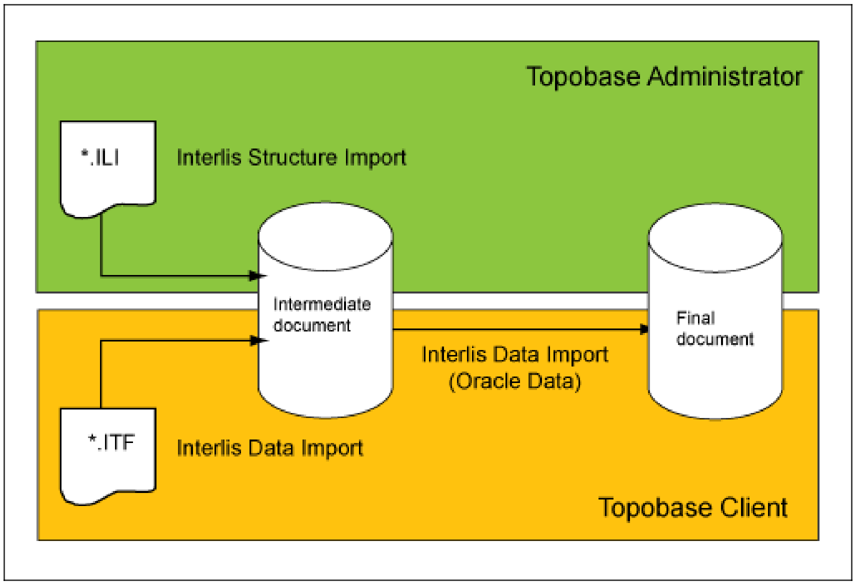
- (**.ili*) file - Interlis data model description file, describes the data structure.
- (**.itf*) file - data transfer file, contains the data.

Topobase Interlis Import supports Interlis V1. For more information about Interlis, see the Interlis Web Site.

Understanding the Concept

Interlis Import uses two Topobase documents, the intermediate document, and the final document.

The intermediate document is an empty Topobase document that is populated in two steps: The first step imports the data structure from the (**.ili*) file. That means the import adds the Interlis topics and feature classes. The second step imports the data from the (*.*itf*) file into the intermediate document. For the first step, you use Topobase Administrator. For the second step you use Topobase Client.



Import Wizard

The Interlis import provides a wizard that guides you through the import process.The wizard controls the sequence of the steps, and validates your input.

On the left side of the wizard, the Task Overview pane displays the status of progress, and indicates which steps to perform next. Click › to show, or ‹ to hide the Task Overview pane.

The navigation bar at the bottom of the window provides buttons to start the steps. When the current step is done, the Next button is activated. Click Next to continue. Use the links on the Task Overview pane, or click the navigation buttons to proceed.

Interlis Structure Import

In the first step on an Interlis import, you import the Interlis data structure into an intermediate document.

BEST PRACTICE Use Topobase Administrator to create one Intermediate document that can be used for all following Interlis data imports.

Creating the Intermediate Document

You create an empty Topobase document for the Interlis data structure to be imported. For Interlis Import, this document does not need any additional data structure, module, or extensions.

However, if you plan to export Interlis data from your intermediate document, you have to add the Interlis Export extension. For example, if you want to create your own (**.ili*) file and want to work directly in the intermediate document.

To create an empty document for Interlis Import

1. Start Topobase Administrator, and create a workspace.
2. Create an empty document. In the Workspace Manager, under Documents, click New.
3. In the Create New Document dialog box, enter the document settings.

   Under Modules, do not select any module.

   Under Extensions, do not select any extension.

   Under Units, Angular Units, select Gon Clockwise.

   Under Jobs, clear Enable Jobs (Versioning) For This Document.

   Click OK to create the document.

To import the Interlis Structure into your empty document.

1. Select the empty document, and import the Interlis data structure. Click Document menu ➤ Import ➤ Interlis Structure Import.
2. In the Interlis Structure Import dialog box, step 1, select the Interlis file (**.ili*).
3. Click Next.

4. Under step 2, you specify the Mapping File.

   Either create a file, or select an existing one. Either case, you can review or modify the settings in the next step.

   Select Create A New Mapping File, and click Next.

5. Under step 3, if you do not want to use the default settings, review the data mapping. See also Review the Data Mapping (page 18).

   - Review Label and Point Definitions.
   - Rename topics, feature classes, and attributes.
   - Review Attribute default values.

6. Click Next.

7. Under step 4, select the label attributes. Click Next. See also Select Label Attributes (page 20).

8. Under step 5: If you want to use the mapping again, save the Mapping File. Click Save As, and enter a file name (**.tbm*).

9. Click Import. Step 6: Import Structure. The data structure is created. After the import, click Close.

   The import adds the Interlis topics and feature classes. After the import, the data structure is displayed in the data model explorer. The intermediate document is now ready for the Interlis data to be imported.

You use Topobase Client to import Interlis data. See Interlis Data Import (page 21).

Review the Data Mapping

When you import the Interlis structure, you map the topics and feature classes of the Interlis structure (*.ili) to Topobase topics and feature classes. You assign topic names, feature class names, and default values. These settings are stored in a Data Mapping file (**.tbm*). The Data Mapping file is an ASCII file that is stored in the file system. We recommend that you store the data mapping file in the same location as the Interlis files (**.ili, *.itf*).

You edit the data mapping file manually, or during the Import, in the Interlis Import Wizard, step 3. The concept is that Interlis Import automatically creates a default data mapping, that can be reviewed, and modified, and saved for reuse.

Review Data Mapping - Label and Point Definitions

In the Interlis data model, there is no distinction between points and labels. Interlis points can either be Topobase points, or Topobase labels. The Interlis import maps Interlis points that have certain properties to label feature classes. To refine the default mapping, select the feature class, and click the arrow buttons.

Review Data Mapping - Renaming

By default, the Interlis import assigns the Interlis names to the Topobase topics, feature classes, and attributes. You can modify the names. For example, if the Interlis structure contains two feature classes with identical names in different topics. In Topobase, feature class names must be unique.

Review Data Mapping - Renaming	Description
Interlis Names	Lists the names of the topics, and feature classes, as described in the Interlis file (**.ili*). ■ To modify a topic name, select the topic node. ■ To modify the feature class name, expand the topic node, and select the topic. By default, Interlis labels are mapped to label feature classes (<name>_TBL). ■ To modify attribute names, expand the topic, and select the feature class. ■ To modify domains, expand the Interlis parent table, and select the domain. To view the mapping of the interlis label attributes, expand the Interlis parent table, and select the Interlis label.
Tables of <name> Attributes of <name> Domains of <name>	Lists the feature classes, attributes, label feature classes, or domains of the selected item. Optionally, for each feature class, enter a new name.

For the label definitions, see also Select Label Attributes (page 20).

Review Data Mapping - Attribute Default Values

In Topobase, some feature class attributes are mandatory. You assign a default value for mandatory attributes. If in a following data export, a record does not contain a value for a mandatory attribute, the assigned default values will be imported.

Review Data Mapping - Attribute Default Values	Description
Topobase Names	Lists the names of the Topobase topics, feature classes, and attributes.
Attributes of <feature class>	Lists the attributes of the selected feature class. Under Default Value, select a value.
Reset	Resets the attribute default values.

Select Label Attributes

By default, for each Interlis label, the import creates a default label definition.

```
select <attribute> from <parent feature class> where FID = $ID
```

You specify the <attribute> in this step. For label definitions, see also Label Properties.

In step 4 of the Interlis import, you review the label properties that are specified in the Interlis file. Under Interlis Labels, select a label, and specify the properties as shown in the following table.

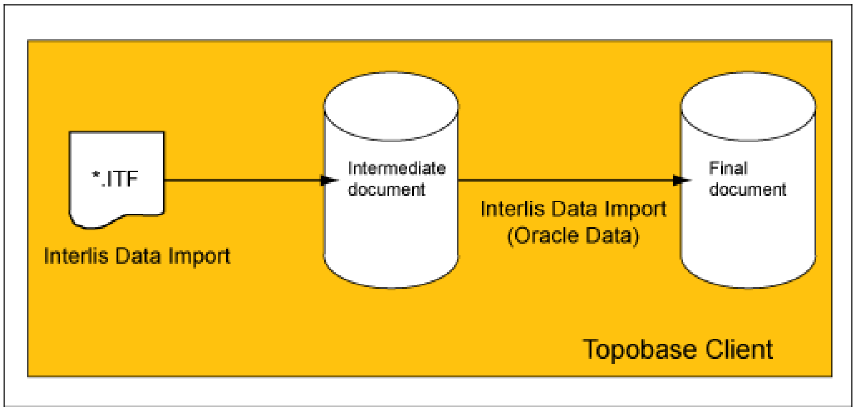
Select Label Attributes	Description
Interlis Labels	Displays Interlis tables that have been identified as Interlis labels (step 3).
Label Definition	Displays label properties as modified in step 3.
Label	Displays the label name.
Parent Table	Specifies the label parent table. The label parent table stores the original information.
Attribute	Specifies the Interlis attribute that is queried.

Orientation	Specifies the Interlis attribute that stores the orientation, and the default value.
Horizontal Alignment	Specifies the Interlis attribute that stores the horizontal alignment, and the default value.
Vertical Alignment	Specifies the Interlis attribute that stores the vertical alignment, and the default value.

For complex labels, such as labels that query multiple attributes from multiple feature classes, you define the label definitions after the Interlis import, using Topobase Administrator.

Interlis Data Import

Use Topobase Client to import Interlis data from an (**.itf*) file into Topobase. Part 1 of the Interlis data import imports the data into an intermediate document. Part 2 of the Interlis data import distributes the data from the intermediate document to your final document, according to the mapping definition for the Oracle data import.



Topobase supports partial and full Interlis Data import, that means that you can execute part 1 and part 2 separately.

- Full import—Part 1 and 2: You import data from the (**.itf*) file into the final document, via Intermediate document.

- Partial import—Part 1: You import data from the *(*.itf)* file into the intermediate document. For example, if you do not need to distribute the data to another document, such as Wastewater, or Electric.

- Partial import—Part 2: You distribute the data from the intermediate document into the final document.

Import Projects

You can import multiple Interlis Data files (**.itf*) into your intermediate document. You mark the imported data by assigning an import project. Using import projects, several users can import data into one intermediate document, and distribute their import projects to different final documents.

Partial import: When you import the Interlis data into the intermediate document, you create an import project. Then you distribute your import project to the final document.

NOTE Your intermediate document must not be job enabled, so you cannot work with jobs. However, you can use import projects to manage multiple imports into the same intermediate document.

When you create an import project, the system creates a feature in the feature class ILI_IMPORT. All imported features are linked to that feature by the foreign-key IMPORT_ID. That means, in the intermediate document, the imported features can be filtered by the import ID.

To configure part 2 of the Interlis import you use the import ID. In the Oracle Data transfer configuration you specify a filter on the current import project.

NOTE To import the features of the selected import project, your configuration must contain the filter for Import_ID. See TB2TB: Edit Feature Class Properties.

Examples: You import a selection of topics. After you have checked the result, you continue to import more data. Or you split the import of a large data file into two parts.

BEST PRACTICE Your system administrator creates one central intermediate document into which all Client users import their Interlis Data files (**.itf*). For example, user A and user B both import data into the same intermediate document. Then, user A distributes his data into the final Wastewater document, and user B distributes his data into the final Gas document.

Importing Interlis Data

Requirements: Your system administrator has created an intermediate document containing the Interlis data structure. The intermediate document provides topics and feature classes for each Interlis feature class or topic. See Creating the Intermediate Document (page 17). The final document contains the Oracle Data Import Extension. See Document Settings.

To import an Interlis data file (**.itf*) into your final document (Full import)

You start the full import from the final document. In the final document, the mapping definition for the Oracle data import must be available.

1. Start Topobase Client, and open the workspace.
2. Select the final document.
3. Right-click, and click Import ➤ Start Interlis Data Import.
4. In the Interlis Data Import dialog box, select the intermediate document.
5. Click Next. Select Create A New Import Project, and enter a name. See also Import Projects (page 22).
6. Click Next. Select the Interlis data file (**.itf*).
7. Show the Advanced Options. Optionally, specify the batch files you want to run before and after the import.
8. Click Next. Specify the data to import. The list displays the topics and feature classes that are stored in the selected (**.itf*) file.
9. Click Next. Specify the options for your final document. By default you import into an empty document. When you import your data into a document that contains data, you optionally select a job.
10. Show the Advanced Options. Select the mapping definition for the Oracle data import. The mapping definition specifies the data transfer between the intermediate and the final document. It is stored in the final document. The mapping definition is provided by your system administrator. See the Topobase Administrator Guide Topobase Data Transfer (TB2TB).
11. Click Import.
12. To check the imported data, generate graphics.

To import an Interlis data file (*.itf) into an intermediate document (Partial import - 1)

This procedure imports the Interlis data into your intermediate document.

1. Start Topobase Client, and open the workspace.
2. Select the intermediate document.
3. Right-click, and click Import ➤ Start Interlis Data Import.
4. Select Create A New Import Project, and enter a name. See also Import Projects (page 22).
5. Click Next. Select the Interlis data file (**.itf*).
6. Show the Advanced Options. Optionally, specify the batch files you want to run before and after the import.
7. Click Next. Specify the data to export. The list displays the topics and feature classes that are stored in the Interlis Data file.
8. To start the import, click Next.
9. After the import, click Close.

To check the imported data, generate graphics.

To distribute the interlis data to the final document (partial import -2)

This procedure distributes your Interlis import project to your final document.

1. Start Topobase Client, and open the workspace.
2. Select the final document.
3. Right-click, and click Import ➤ Complete Interlis Data Import.
4. Select the intermediate document.
5. Click Next. Select the import project to distribute.

   If the list of import projects is empty, the intermediate document does not contain any imported data. Use the Start Interlis Data Import command to import Interlis data into the intermediate document. See also Import Projects (page 22).

6. Click Next. Specify the options for your final document. When you import your data into a document that contains data, you optionally select a job.

7 Under Advanced Options, select the mapping definition for the Oracle data import. The mapping definition specifies the data transfer between the intermediate and the final document. It is stored in the final document. The mapping definition is provided by your system administrator. See the Topobase Administrator Guide Topobase Data Transfer (TB2TB).

Import Data Settings

The Task Overview pane displays the import steps, the settings, and the status of the import progress.

Task Overview	Description
Import Project Name	Displays the name of the import project. Use import projects to manage multiple imports. See also Import Projects (page 22).
Interlis Data (**.itf*)	Displays the name of the Interlis data file (**.itf*).
Intermediate Document	Displays the name of the intermediate document.
Final Document	Displays the name of the final document. When you start the full import from an intermediate document, no final document is needed.
Job	Displays the selected job.
1. Setup Import	Sets up the intermediate document
Select Intermediate Document	For full import: Selects the intermediate document. See also Creating the Intermediate Document (page 17).
Create A New Import Project	For full import: Creates an import project. Enter a name. See also Import Projects (page 22).
Work With An Existing Import Project	Selects an import project. For full import: Use an exiting import project, to add data to. For partial import: Select the import project to distribute to the final document.

Empty Import Project Before Importing New Data	Removes all features that are marked with the selected import project. Select this option to repeat the import, for example, to modify your data selection.
2. Specify Data For Import - Import File	For full import: Selects the data to import.
Specify Interlis Data File (*.itf*)	Selects the Interlis Data File (**.itf*).
Validate Interlis Data File Before Importing	Runs a syntax validation. For example, checks names, and mandatory attributes.
Advanced Options	Selects batch files to be executed and after the import. For example, to drop spatial indices.
Run Batch File Before Import	Selects a file to be executed before the data import. For example, to disable the topology triggers to improve performance.
Run Batch File After Import	Selects a file to be executed after the data import. For example, to enable the topology triggers.
2. Specify Data For Import - Select Import Data	Selects the data to be imported. The list displays the topics of the selected (**.itf*) file.
3. Specify Final Document Options	Sets up the final document.
Perform Fast Import (For Empty Documents Only)	We recommend that you use this option only if your final document does not contain any data. This option is only available for Topobase administrators. Performs the migration according to the Migration Model settings in the Oracle Data transfer configuration. For example, you use the SQL loader to migrate data, which is a fast method. However feature rules are not triggered, so this method usually requires post processing. For more information about the Migration Model settings, see the Topobase Administrator Guide, section Topobase Data Transfer (TB2TB).

Perform Standard Import	Selects the standard migration method. This method guarantees data consistency. No post processing is required.
Please Select A Job	Selects a job. If your final document contains data, we recommend that you use a job for the import. Interlis import does not check whether a feature has already been imported. Using jobs helps you to undo any imports.
Advanced Options	Selects the mapping definition for the Oracle data import. The list displays the mapping definitions that are stored in the selected final document. See Oracle Data Import - Mapping Definition (page 27).
4. Import Data	Imports the selected data.

Oracle Data Import - Mapping Definition

The mapping definition for the Oracle data import describes how the features of the intermediate document are stored in the final document. That means it maps the feature classes, domains, and attributes of the intermediate document to the feature classes, domains, and attributes of the final document.

Interlis Data Import uses the Oracle data import tool to distribute data from the intermediate document to the final document. The distribution process uses an appropriate mapping definition.

NOTE The Swiss Land demo data set contains a mapping definition file.

You use Topobase Administrator to edit, modify, export, and import mapping definitions for the Oracle data import. See the Topobase Administrator Guide Topobase Data Transfer (TB2TB).

Interlis Export

Use Topobase Client to export features into an Interlis data transfer file (**.itf*). Interlis Export supports several options.

- Full Export (all features).

- Export data within a job perimeter.
- Export data within a defined area.
- Export data that has been filtered (topic, feature class).

Topobase Interlis Export supports Interlis V1. For more information about Interlis, see the Interlis Web Site.

IMPORTANT Requirement: Your document must contain the Interlis Export extension that stores the Interlis export configuration.

NOTE The Land CH Data Model contains the Interlis Export Extension, and provides a default configuration.

To export features to a (.itf*) Interlis file**

1. Start Topobase Client, and open the workspace.
2. Select the document.
3. Right-click, and click Export ➤ Interlis Data.
4. In the Interlis Data Export dialog box, specify the following.
   - Data: Filter by topic and classes.
   - File location: (**.itf*) file, log files.
   - Spatial Filter: For example, by job, by topology, by perimeter.

   See also Interlis Export Settings (page 28).
5. Click Export.

Interlis Export Settings

In the Interlis Export dialog box you specify the export as shown in the following table.

Interlis Export dialog box	Description
Specify Data For Export area	Specifies the Interlis data model, and selects topics and feature classes.

Select Interlis Data Model	Selects the Interlis data model. The list displays the export configurations that are stored in the Interlis export extension. See also Interlis Export Extension (page 30).
Select Topics And Feature Classes	Selects the features to be exported. The explorer displays the topics and feature classes that are required for the selected data model.
Specify Export File Name And Location area	Specifies the export files. ■ Interlis Data File (**.itf*) ■ Log File (**.txt*)
Filter Export Data area	Specifies the spatial extent.
Export All	Exports all features of the selected feature classes without any spatial restrictions.
Export Features Within The Job Perimeter	Selects a job. Exports all features of the selected feature classes that lie within the job perimeter. The export intersects the job perimeter with all area topologies. That ensures that topology features are exported consistently, for example, if a centroid does not lie within the job perimeter.
Export Features Within The Selected Topology	Selects a topology. The list displays the area topologies. To select the export perimeter, click Select Features, and pick the areas in the map. The export intersects the export perimeter with all area topologies. That ensures that topology features are exported consistently, for example, if a centroid does not lie within the perimeter.
Export Features Within The Specified Perimeter	Specifies the export perimeter by selecting either an existing polygon, by digitizing a polygon or by digitizing a window.
Retrieve Only Data Inside The Specified Filter Of Topology Or Perimeter	For export within topology, job perimeter, and perimeter. Specifies whether to perform the intersection of the export perimeter with the area topologies.

	Select this option, to export only the features that lie within the export perimeter. **NOTE** Selecting this option may result in invalid Interlis data.
Export	Creates the Interlis data file (**.itf*).

Interlis Export Extension

The Interlis Export command is only available, if the document contains the Interlis Export Extension.

To add the Interlis Export extension

1. Start Topobase Administrator.
2. Select the document.
3. Under Document Settings, click Extensions. Select Interlis Export Extension.
4. Click Save. In the Update Modules And DataModels dialog box, click Update. Click Close.

The Interlis Export extension provides tables that store the export configuration.

- ILI_ATTRIBUTE—Attribute mapping file. Stores the attribute names, corresponding to the names in the Interlis data model description file (**.ili*).
- ILI_EXPORT—Record the exported features.
- ILI_LABELDEF
- ILI_LINEATTR—Manages the line types.
- ILI_MODEL—Stores the model name, corresponding to the name in the Interlis data model description file (**.ili*). The model name is displayed in the Interlis Export dialog box, under Select Interlis Data Model.
- ILI_TABLE—Table mapping file. Stores the table names, corresponding to the names in the (**.ili*) file. For example, specifies whether parent features,

or child features have to be exported. Stores the SQL select statements that are executed to export the features. WHERE Clauses are not allowed.

- ILI_TOPIC—Topic mapping file. Stores the topic names, corresponding to the names in the (*.*ili*) file.

Coordinate Import

See the Client User Guide.

- Importing Point Coordinates

Coordinate Export

See the Client User Guide.

- Exporting Point Coordinates
- Coordinate Report

Land Management Workflows

4

Using Land Management Workflows

Workflows guide you through the most frequently performed tasks. They contain embedded information and options specific to the task, and help you to acquire data consistent and compliant to the external standards.

Data Management workflows

- Create Maintenance Job (page 61)
- Merge Real Estates (page 59)
- Split Real Estate (page 58)

Data Acquisition workflows

- Create Property (page 39)
- Create Land Cover (page 43)
- Create Single Object (page 45)
- Create Name (page 47)
- Create Location (page 49)
- Add Building Element (page 50)
- Create Building (page 51)
- Create Servitude (page 56)

Before starting a workflow, you must generate graphics.

To start a workflow

1. Click Home tab ➤ Display panel ➤ Generate Graphic.
2. Click the Workflow Explorer icon to display the workflows.
3. Do one of the following:
   - Double-click a workflow in the Workflows group.
   - Right-click a workflow. Click Execute.
   - Click a workflow. Click Execute.

Optionally, you start a workflow from the Document Explorer. Select the topic, right-click, and click the workflow command.

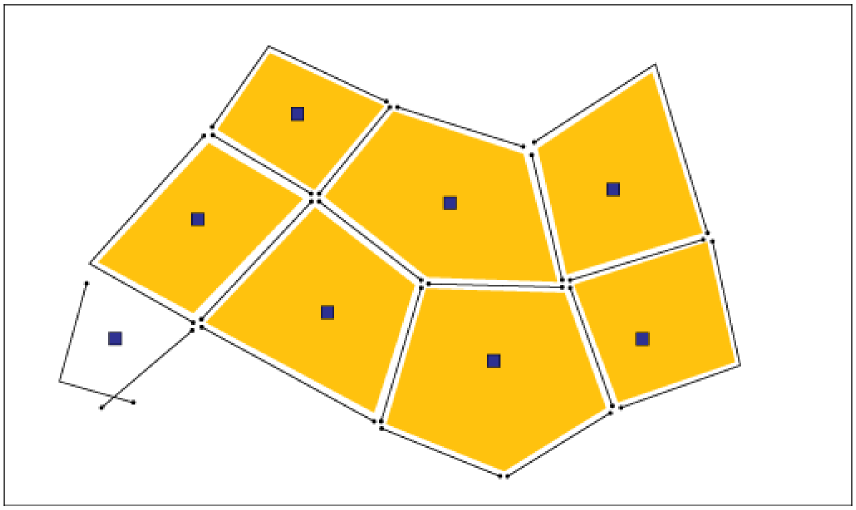
Working with Area Topologies

The Land data model uses area topologies to manage consistent and non overlapping areas for the following features. Normally, these features completely cover the spatial extension of your project.

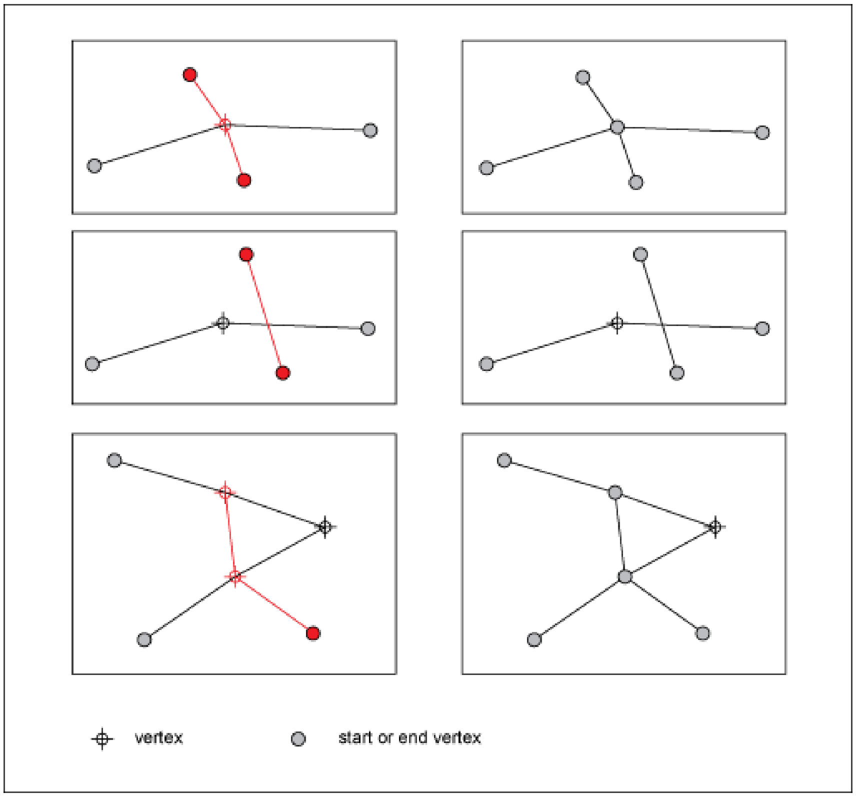
- Land cover
- Real estate (ownership)
- Municipality (administrative)
- Canton (administrative)
- District (administrative)
- Local names
- Tolerance zones
- Plan repartition

When you digitize a feature, the topology requirements must be met, such as no overlapping lines, exact snap of line endpoints, lines must build a closed polygon.

You digitize the centroids, and the boundary lines. If the topology requirements are met, the polygon features are generated automatically. Use the Topology Checker to check for invalid area topologies. See the Topobase Client User Guide, Area Topology Checker.



When you digitize area topology edges, such as real estate boundary lines, or land cover boundary lines, the topology edges are split automatically, for example, when you digitize a line using a vertex of an existing topology edge of the same topology, the line is split automatically.



See also:

- Area Topology Introduction

Working With Jobs

When you modify any feature of the Land data base, you use Topobase Jobs. Using jobs you clearly separate data that is valid and approved (Live job) from data that is being processed (Open job) and data that is subject to approval processes (Pending job).

Your Topobase administrator provides the job templates and configures the perimeter feature classes. See Land Management: Job Settings (page 11).

To create a job

1. Start Topobase Client, and open the workspace.
2. Click Display panel ➤ Generate Graphic.
3. Click Home tab ➤ Data Source panel ➤ Job Manager.
4. In the Job Manager dialog box, click Create Job.
5. In the Create Job dialog box, select the Job Template.
6. Enter a name.
7. Click OK.
8. Click Yes, and select the job perimeter, for example by selecting the real estates in the map.
9. In the Select Perimeter dialog box, select the polygon topology feature class, for example, LM_REAL_ESTATE_TSUR.
10. Click Select. In the map, click the real estates. Select Enter, to finish the selection.
11. Optionally, check the perimeter in the map. In the Select Perimeter dialog box, click the Highlight icon.
12. Close the Select Perimeter dialog box.
13. Note, that in the Job Manager dialog box, your new job is added in the explorer. Expand Jobs ➤ Job States ➤ Open.
14. Close the Job Manager.

Use workflows to create or modify features.

To review the feature modifications (Job Explorer)

1. In the Topobase task pane, click the Job Explorer icon .
2. In the Job Explorer dialog box, select a feature, and use the toolbar buttons to process the features. See also the Topobase Client User Guide, section Job Explorer.

When you have finished your modifications, you change the job state to pendent, and then to Live.

To change the job state (Job Explorer)

1. In the Job Explorer dialog box, select the job root node, and click Change The State Of The Job icon .
2. In the Change Job State dialog box, select the job state Pending.
3. When the job changes have been approved, set the job state to Live.

See also:

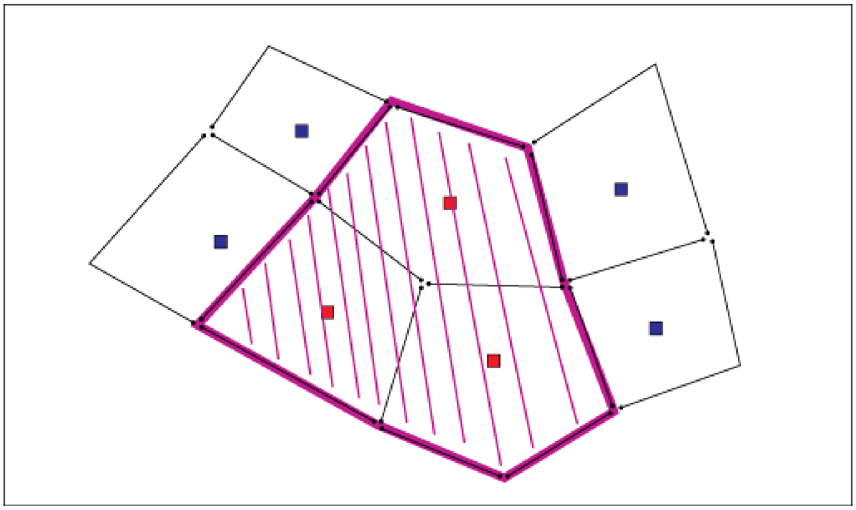
- Using Land Management Workflows (page 33)

See also the Topobase Client User Guide, section Working With Jobs.

Job Mutation Perimeter

When you create a job, you specify the job perimeter. The job perimeter marks the border of the area where you modify your data. Working with a selected job, you cannot modify data that lies outside its job perimeter.

An exception is the Maintenance job. See Create Maintenance Job (page 61).



Each job is assigned to one mutation perimeter. The perimeter is a polygon feature that is stored in the job mutation feature class LM_AD_MUTPERIMETER.

For each modified feature, the relation to the job perimeter is stored in the attribute <feature class>.FID_AD_MUTPERIMETER.

Job perimeters may overlap. Any job conflicts would be detected when you change the job state. This is the standard behavior of the optimistic feature locking type that is used.

You specify the job mutation perimeter either by selecting existing topology polygons, such as real estates, cantons, or districts, or you digitize a new polygon.

To view the job mutation perimeter

1. Select the job.
2. Click Home tab ➤ Data Source panel ➤ Job Manager.
3. In the Job Manager dialog box, select the job.
4. Right-click, and click Select Perimeter.
5. In the Select Perimeter dialog box, near Functions, click the Highlight icon .

See also:

- Create Maintenance Job (page 61)

Create Property

Use the Create Property workflow to create properties such as

- Real estate
- Development right (DPR)
- Mine

See also:

- Managing Ownership (page 5)
- Split Real Estate (page 58)

- Merge Real Estates (page 59)

Create Property: Real Estate

You use the Create Property workflow to create real estate features along with their labels.

Required: Boundary points, control points.

Real estate features are part of an area topology, that means you have to create one centroid along with the edges. The edges must not overlap, and they must build complete polygons. When you digitize the edges, a feature rule automatically splits the lines, where it is necessary. See also Working with Area Topologies (page 34).

When you digitize a centroid, you assign a property number that must be unique per Numbering Domain. If the Numbering Domain is not available, you cannot create the centroid. See also Land Management: Numbering Domain (page 90).

Using the workflow, you either create centroids only, edges only, or both centroids and edges. This allows you to run the workflow according to your method of data acquisition. At any time, you can use the Topology Checker to check the topology. See also Area Topology Checker.

To create a property (real estate - centroid and edges)

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Property workflow.
3. In the workflow pane, select the Property Type: Real Estate.
4. Select Digitize Boundaries And Centroids.
5. Select Show Form To Edit Optional Attributes.
6. Under Property Information enter the Numbering Domain, and the Property Number.
7. Click Start, and in the map, click the boundary points, and the control points that determine the real estate. Use the Osnap setting Node. Follow the prompts in the command line.
8. Click Enter to finish a line. Digitize as many lines as you need. Click Cancel to finish with the real estate lines.

9 Digitize the centroid. Follow the prompts in the command line. You specify the label position and orientation.

10 The Property form opens. Enter your optional attributes, and close the form.

11 To check the Topology, click the Topology Checker icon, and select the LM_REAL_ESTATE topology. See also Area Topology Checker.

12 In the workflow pane, click Close.

Real estate feature classes

Using the Create Property workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_OW_PROPERTY	Property	Property parent feature. Stores the Property Type, and Property Number.
LM_OW_REAL_ESTATE	Real Estate.	Real estate centroid, centroid feature class of the area topology LM_REAL ESTATE. Stores the area, and the relation to the Property parent feature.
LM_OW_REAL_ESTATE_L	Real Estate L	Real estate boundary; line feature class of the area topology LM_REAL_ESTATE.
LM_OW_REAL_ESTATE_TSUR		Real estate polygon. This polygon is automatically generated by the topology.

Create Property: Mine

You use the Create Property workflow to create mine areas. Mine features are stored as compound polygons. That means that you create a polygon that can be composed of single line segments, for example if you want to apply different line styles.

To create a Mine polygon

1 Select the job to work with. See Working With Jobs (page 36).

2 Start the Create Property workflow.

3 In the workflow pane, select the Property Type, such as Mine.

4 Select Show Form To Edit Optional Attributes.

5 Under Property Information enter the Numbering Domain, and the Property Number.

6 Click Start, and in the map, digitize the line segments. Follow the prompts in the command line.

7 Specify the label position and orientation.

8 The Property form opens. Enter your optional attributes, and close the form.

9 In the workflow pane, click Close.

Mine feature classes

Using the Create Properties workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_OW_PROPERTY	Property	Property parent feature. Stores the Property Type, and Property Number.
LM_OW_MINE	Mine	DPR compound parent feature. Stores the relation to the Property parent feature.
LM_OW_MINE_L	Single Segment (Mine).	DPR compound child feature. Stores the single line segments.

Create Property: DPR

You use the Create Property workflow to create areas of DPR (rights). DPR features are stored as compound polygons. That means that you create a polygon that can be composed of single line segments, for example if you want to apply different line styles.

To create a DPR right

1 Select the job to work with. See Working With Jobs (page 36).

2 Start the Create Property workflow.

3 In the workflow pane, select the Property Type, such as DPR, Fish Breeding.

4 Select Show Form To Edit Optional Attributes.

5 Under Property Information enter the Numbering Domain, and the Property Number.

6 Click Start, and in the map, digitize the line segments. Follow the prompts in the command line.

7 Specify the label position and orientation.

8 The Property form opens. Enter your optional attributes, and close the form.

9 In the workflow pane, click Close.

DPR feature classes

Using the Create Properties workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_OW_PROPERTY	Property	Property parent feature. Stores the Property Type, and Property Number.
LM_OW_DPR	DPR	DPR compound parent feature. Stores the relation to the Property parent feature.
LM_OW_DPR_L	Single Segment (DPR)	DPR compound child feature. Stores the single line segments.

Create Land Cover

Use the Create Land Cover workflow to create land cover features that match the data model requirements. See also Managing Land Cover (page 4).

Land cover features are part of an area topology, that means you have to create one centroid along with the edges. The edges must not overlap, and they must build complete polygons. When you digitize the edges, a feature rule automatically splits the lines, where it is necessary. See also Working with Area Topologies (page 34).

Using the workflow, you either create centroids only, edges only, or both centroids and edges. This allows you to run the workflow according to your method of data acquisition. At any time, you can use the Topology Checker to check the topology. See also Area Topology Checker.

To create a land cover area (boundary and centroid)

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Land Cover workflow.
3. In the workflow pane, select the Land Cover Type, such as Garden.
4. Select Digitize Boundaries And Centroids.
5. Select Show Form To Edit Optional Attributes.
6. Under Land Cover Information, select the Quality, such as Other.

   Optionally enter a Land Cover Name.
7. Click Start, and in the map, digitize the land cover boundaries. Follow the prompts in the command line.
8. First, you digitize the boundary lines. Click Enter to finish the line. Digitize as many lines as you need. Click Cancel to finish with the lines.
9. Digitize the centroid.
10. The Surface form opens. Enter your optional attributes, and close the form.
11. To check the Topology, click the Topology Checker icon, and select the LM_LAND_COVER topology. See also Area Topology Checker.
12. In the workflow pane, click Close.

Land Cover feature classes

Using the workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_LC_SURFACE_L	Surface L	Land cover boundary; line feature class of the area topology LM_LAND_COVER.
LM_LC_SURFACE	Surface	Land cover centroid, centroid feature class of the area topology LM_LAND_COVER.

Feature ClassName	Caption	Description
LM_LAND_COVER_TSUR		Land cover polygon. This polygon is automatically generated by the topology.

Create Single Object

Use the Create Single Object workflow to create objects such as trees, bridges, green houses. Single objects consist of one parent feature that stores the attributes, and one or more geometry features. Depending on the type, a single object can be of different geometry type, such as point, line string, or compound polygon. For example, a tree is of geometry type Point. A green house can be of geometry type line string, or compound polygon.

See also Managing Single Objects (page 7).

NOTE The data model stores the underground buildings as Single Objects. However, to create underground buildings, you use the Create Building workflow. See Create Building (page 51).

To create a single object

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Single Object workflow.
3. In the workflow pane, select the Single Object Type, such as Green House.
4. Select Digitize Single Object.
5. Select Show Form To Edit Optional Attributes.
6. Under Single Object Information, select the Quality, such as Other.

   Optionally enter an Object Name. If you enter an object name, you will be prompted to specify the label position and orientation.
7. Click Start, and in the map, digitize the single object geometry elements. Follow the prompts in the command line. If prompted, select the geometry type, and digitize a feature of the selected geometry. Digitize as many elements as needed. Press <ESC> to finish.
8. The Single Object form opens. Enter your optional attributes, and close the form.
9. In the workflow pane, click Close.

A single object can consist of multiple geometry elements, such as lines, points, and polygons. Use the Create Single Object workflow to add elements to an existing single object.

To add an element to a single object

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Single Object workflow.
3. In the workflow pane, select Add Single Element To Existing Object.
4. Deselect Show Form To Edit Optional Attributes.
5. Click Start, and in the map, select the existing object. Follow the prompts in the command line. Digitize the additional element.
6. In the workflow pane, click Close.

Single Object feature classes

Using the workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_SO_SINGLE_OBJECT	Single Object	Parent feature class that stores the attributes of the single object.
LM_SO_LINE_ELEMENT	Line Element	Line elements of the single object. Stores the relation to the Single Object parent feature.
LM_SO_POINT_ELEMENT	Point Element	Point elements of the single object. Stores the relation to the Single Object parent feature.
LM_SO_SURFACE_ELEMENT	Surface Element	Polygon elements of the single object. Stores the relation to the Single Object parent feature.
LM_SO_OBJECT_NAME	Object Name	Attribute feature class. Stores the object name. Related to the label feature class.

Create Name

Use the Create Name workflow to create names. The data model provides three types of name features that differ in the way they are associated with areas. For example:

- **Local Name**—Local names are associated to an area. You digitize the centroids and the boundary lines that are part of an area topology. Local name areas cover the whole area of your project.
  The centroid stores the name, and the Numbering Domain. If the Numbering Domain is not available, you cannot create the centroid. See Land Management: Numbering Domain (page 90)
- **Locality**—Name that is not associated to an area. You enter the name, and you place a label.
- **Place**—Name that is associated to a boundary polygon.

See also Managing Names (page 8).

To create a local name (centroid and boundary line)

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Name workflow.
3. In the workflow pane, select the Name Type: Local Name.
4. Select Digitize Boundaries And Centroid.
5. Select Show Form To Edit Optional Attributes.
6. Under Name Information, select the Numbering Domain, and enter the name.
7. Click Start, and in the map, digitize the local name boundary lines. Follow the prompts in the command line.
8. Click Enter to finish a line. Digitize as many lines as you need. Click Cancel to finish with the local name boundary lines.
9. Digitize the centroid. Follow the prompts in the command line. You specify the label position and orientation.
10. The Property form opens. Enter your optional attributes, and close the form.

11 To check the Topology, click the Topology Checker icon, and select the LM_LOCAL_NAME topology. See also Area Topology Checker.

12 The Local Name form opens. Enter your optional attributes, and close the form.

13 In the workflow pane, click Close.

To create a named locality

1 Select the job to work with. See Working With Jobs (page 36).

2 Start the Create Name workflow.

3 In the workflow pane, select the Name Type: Named Locality.

4 Select Show Form To Edit Optional Attributes.

5 Under Name Information, select the Numbering Domain, and enter the name.

6 Click Start, and in the map, specify the label position and orientation.

7 In the workflow pane, click Close.

To create a Place name

1 Select the job to work with. See Working With Jobs (page 36).

2 Start the Create Name workflow.

3 In the workflow pane, select the Name Type: Place Name.

4 Select Show Form To Edit Optional Attributes.

5 Under Name Information, select the Numbering Domain, and enter the name.

6 Click Start, and in the map, digitize the boundary lines, and the label position and orientation.

7 In the workflow pane, click Close.

Name feature classes

Using the workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_NA_LOCAL_NAME	Local Name	Local Name centroid, centroid feature class of the area topology LM_LOCAL_NAME.
LM_NA_LOCAL_NAME_TBL		Label feature to inscribe the name.
LM_NA_LOCAL_NAME_L	Local Name L	Local Name boundary; line feature class of the area topology LM_LOCAL_NAME.
LM_NA_NAMED_LOCALITY	named Locality	Attribute feature that stores the name. Related to the label feature class LM_NA_NAMED_LOCALITY_TBL.
LM_NA_PLACE_NAME	Place Name	Polygon feature that stores the area of the place.

Create Location

Use the Create Location workflow to create location features such as Street, Named Area, or Place.

See also Managing Locations (page 8).

To create a location

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Location workflow.
3. In the workflow pane, select the Location Type such as Named Area.
4. Select Digitize Location.
5. Select Show Form To Edit Optional Attributes.
6. Under Location Information, select the Numbering Principle, and enter the name. The location name must be unique.
7. Click Start, and in the map, digitize the location polygon. Follow the prompts in the command line.

8 Click Enter to finish a line.

9 In the Choose Label Definition dialog box, select a label definition, such as Location Name Label. Click OK.

10 Select the label position and orientation.

11 The Location form opens. Enter your optional attributes, and close the form.

12 In the workflow pane, click Close.

Location feature classes

Using the workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_LO_LOCATION	Location	Parent feature that stores basic attributes of the location. Related to additional attribute features, or geometry features, such as the Location Name, Named Area, or Road Section.
LM_LO_LOCATION_NAME	Location Name	Attribute feature that stores the name. Related to the label feature class LM_LO_LOCATION_NAME_TBL.
LM_LO_NAMED_AREA	Named Area	Polygon feature. Related to the Location parent feature.
LM_LO_ROAD_SECTION	Road Section	Line string feature. Related to the Location parent feature.

Add Building Element

Use the Add Building Element workflow to add elements to an existing building. For example, you add the following elements.

- Building geometry—Adds geometry to a building parent feature. For example, building parts such as stairs, chimney. Or you add the building geometry to a projected building.
- Building Insurance information
- Building House Entrance Information

You can add multiple building parts (Single Objects), and buildings (Land Cover) to one building parent (LM_BU_BUILDING).

To add building elements

This workflow adds a stair to an existing building.

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Add Elements To Existing Buildings workflow.
3. Select Add Building Geometry.
4. Select Show Form To Edit Optional Attributes.
5. Select a Building Type, such as Other Building Part, Stair.
6. Click Start, and in the map, select the building. Either select a building point, or a building centroid. Follow the prompt in the command line.
7. Select the kind of element you want to add, for example, Line String. Digitize the lines.
8. The Single Object form opens. Enter your optional attributes, and close the form.

Depending on the selected building type, the building element is stored in the Single Object feature class, or in the Land Cover feature class. Building parts can be of geometry type point, line, or compound polygon.

Create Building

Use the Create Building workflow to create both overground and underground buildings. You digitize the geometry, and enter the insurance, and the house entrance attributes. You start the workflow either from

- Workflow explorer.
- Topobase explorer; either from the topic node, or from the feature class node:
  - Topic Building, Land Cover, or Single Object.
  - Feature class Land Cover (LM_LC_SURFACE), Single Object (LM_SO_SINGLE_OBJECT) or Building (LM_BU_BUILDING).

To create an overground building

The workflow creates the building parent (building point), the land cover centroid and line, insurance information, and the house entrance point.

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Building workflow.
3. Select a Building Type, such as Building.
4. Select the Building State, such as Real.
5. Select the Building Number. The number is validated against existing numbers, and format requirements.
6. Enter the EGID.
7. Select Create Building Insurance.
8. Enter the insurance number.
9. Select Create House Entrance.
10. Enter the entrance number.
11. Enter the EGID.
12. Select the Location. Either select a value from the list, or click Select Location In Map.
13. Click Start, and in the map, digitize the building point. Follow the prompts in the command line.
14. Digitize the lines, and a centroid for the building.
15. Select the position of the Insurance label.
16. Digitize the House Entrance, and enter the position for the House Entrance label.
17. In the Building feature class form, add more attributes.

Overground building feature classes

Using the workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_BU_BUILDING	Building	Building point, parent feature that stores the building number, the EGID, and has a relation to the house entrance.
LM_LC_SURFACE	Surface	Land cover centroid, centroid feature class of the area topology LM_LAND_COVER. Has a relation to the building parent feature.
LM_LC_SURFACE_L	Surface L	Land cover boundary; line feature class of the area topology LM_LAND_COVER.
LM_BU_HOUSE_ENTRANCE	House Entrance	House entrance point. Has a relation to the building parent feature class.
LM_BU_BUILD_BUILD_INSUR	Building Insurance	Insurance attribute feature.

Creating an Underground Building

The workflow for underground buildings requires the same entries as the workflow for overground buildings, however the building geometry is stored in the Single Object feature class.

To create an underground building

The workflow creates the building parent (building point), the Single Object features, insurance information, and the house entrance point.

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Building workflow.
3. Select the Building Type: Underground Building.
4. Select the Building State, such as Real.
5. Select the Building Number. The number is validated against existing numbers, and format requirements.
6. Enter the EGID.

7 Select Create Building Insurance.

8 Enter the insurance number.

9 Select Create House Entrance.

10 Enter the entrance number.

11 Enter the EGID.

12 Select the Location. Either select a value from the list, or click Select Location In Map.

13 Click Start, and in the map, digitize the building point. Follow the prompts in the command line.

14 Select the Single Object type, such as Surface. Digitize the lines, for the building.

15 Select the position of the Insurance label.

16 Digitize the House Entrance, and enter the position for the House Entrance label.

17 In the Building feature class form, add more attributes.

Underground building feature classes

Using the workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_BU_BUILDING	Building	Building point, parent feature that stores the building number, the EGID, and has a relation to the house entrance.
LM_SO_SINGLE_OBJECT	Single Object	Single object parent feature. Has a relation to the building parent feature.
LM_SO_LINE_ELEMENT LM_SO_SURFACE_ELEMENT		Geometry features. Related to the Single Object parent feature.
LM_BU_HOUSE_ENTRANCE	House Entrance	House entrance point. Has a relation to the building parent feature class.

Feature ClassName	Caption	Description
LM_BU_BUILD_BUILD_INSUR	Building Insurance	Insurance attribute feature.

Create Building: Options

Create Building	Description
Building Type	Specifies the building type. The list displays all building types of the domain tables Building Category (LM_LC_CATEGORY_TBD), and Single Object Category (LM_SO_OBJECT_CATEGORY_TBD). For overground buildings, the building type is stored in the centroid feature LM_LC_SURFACE.ID_LC_TYPE.
Building Information	Specifies attributes of the parent building feature LM_BU_BUILDING.
State	Specifies the state, such as Real. The list displays the LM_LO_STATE_TBD domain table entries.The state is stored in the parent building LM_BU_BUILDING.
Building Number	Number of the building, as assigned by the cantonal building authorities.
EGID Number	Specifies a unique identifier for houses, compliant with Swiss standards.
Numbering Domain	Specifies the numbering domain that consists of a canton short value. See Land Management: Numbering Domain (page 90).
Create Building Insurance	Adds Insurance information that is stored in the Building Insurance feature class LM_BU_BUILDING_INSURANCE.
Insurance Number	Specifies the insurance number.
Create House Entrance	Adds House Entrance information that is stored in the House Entrance feature class LM_BU_HOUSE_ENTRANCE. A building can have several entrances, and house numbers.

Create Building	Description
Entrance Number	Specifies the house number (assurance number of the building).
EDID Number	Specifies a unique identifier for houses.
Location	Specifies the location of the building. Either select from a list, or click a name in the map (label feature class LM_LO_LOCATION_NAME_TBL.
Show Form To Edit Optional Attributes	Opens the forms to edit the attributes.

Create Projected Building

You can create a projected (planned) building without knowing the exact geometry. Projected buildings are stored in the feature class Pa Projected Surface (LM_PA_SURFACE_PROJ). They have a relation to the feature class Building Point (LM_PA_SURFACE_PROJ.FID_BU_BUILDING).

- You digitize a point at the approximate position of the building (Building Point). Use the Create Building workflow.
- You create the house entrances. Either use the Create Building workflow, or the Add Building Element workflow.
- Optionally, you digitize a Projected Polygon feature (LM_PA_SURFACE_PROJ).
- When the building is existing, you measure its exact location, and create the building. Use the Add Building Element workflow to add the building geometry.

Create Servitude

Use the Create Servitude workflow to create servitude features such as Right Of Access, or Third Party Access. Servitudes are associated with a geographic location. For example, when you create a Right Of Access, you digitize the associated location either as a line string, or as a polygon. One servitude can be associated with multiple geometry elements, for example, to display the feature in the map.

See also Managing Servitudes (page 8).

To create a servitude

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Create Servitude workflow.
3. In the workflow pane, select the Servitude Type such as Right Of Access.
4. Select Digitize Servitude.
5. Select Show Form To Edit Optional Attributes.
6. Under Servitude Information, select the Numbering Domain. Enter a Servitude Number.
7. Optionally, enter a description.
8. Click Start, and in the map, digitize the location polygon. Follow the prompts in the command line.
9. Click Enter to finish a line.
10. In the Choose Label Definition dialog box, select a label definition, such as Servitude Number Label. Click OK.
11. Select the label position and orientation.
12. The Servitude form opens. Enter your optional attributes, and close the form.
13. In the workflow pane, click Close.

Servitude feature classes

Using the workflow you create features as shown in the following table.

Feature ClassName	Caption	Description
LM_SE_SERVITUDE	Servitude	Parent feature that stores basic attributes of the servitude. Related to geometry elements, such as lines, points, and polygons.
LM_SE_LINE_ELEMENT	Line Element	Line string feature. Related to the Servitude parent feature.
LM_SE_POINT_ELEMENT	Point Element	Point feature. Related to the Servitude parent feature.

Feature ClassName	Caption	Description
LM_SE_SURFACE_ELEMENT	Surface Element	Polygon feature. Related to the Servitude parent feature.

Split Real Estate

Use the Split Real Estate workflow to split one real estate into two new real estates. Splitting a real estate includes the following tasks.

- **Property number**—Assign new property numbers.
- **Boundary lines**—Digitize the new real estate lines.

See also Managing Ownership (page 5) and Managing Boundary Points (page 6)

Before you split a real estate, the new boundary points must be available.

To split a real estate

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Split Real Estate workflow.
3. In the map, select the real estate to split, by clicking somewhere within the area.
4. Select Show Form To Edit Optional Attributes.
5. Select Reuse Old Property Number.
6. Under New Property 2, enter the Property Number.
7. Click Start, and in the map, click the boundary points that determine the new boundary line. Use the Osnap setting Node. Follow the prompts in the command line.
8. Click Enter to finish a line. Digitize as many lines as you need. Click Cancel to finish with the real estate lines.
9. Digitize the position (centroid) of the new property 1.
10. Specify the label position and orientation of the new property 1.
11. Digitize the position (centroid) of the new property 2.
12. Specify the label position and orientation of the new property 2.

13 The Property form opens, and both properties are in the filter. Enter your optional attributes, and close the form.

14 To check the Topology, click the Topology Checker icon, and select the LM_REAL_ESTATE topology. See also Area Topology Checker.

15 In the workflow pane, click Close.

Real estate feature classes

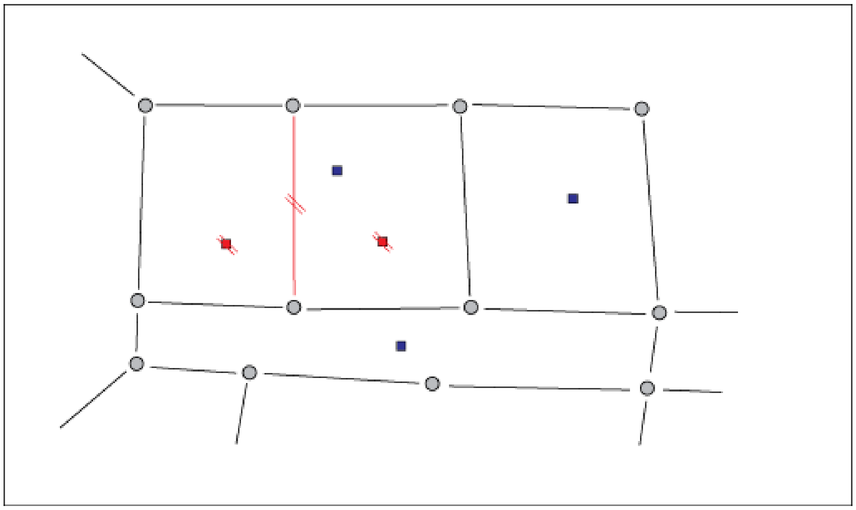
Using the Split Real Estate workflow you update features as shown in the following table.

Feature ClassName	Caption	Description
LM_OW_PROPERTY	Property	Property parent feature. The workflow deletes the property feature of the existing real estates, and creates two new features.
LM_OW_REAL_ES-TATE	Real Estate.	Real estate centroid. If you do not reuse the property number, the workflow deletes the old centroid, and creates two new centroids.

Merge Real Estates

Use the Merge Real Estates workflow to merge two adjacent real estate areas into one. Merging real estates includes the following tasks.

- **Property number**—Assign the property number.
- **Centroids**—Delete the old centroids, and add a new centroid.
- **Boundary lines**—Delete the adjacent real estate lines.



See also Managing Ownership (page 5).

To merge real estates

Follow the steps to merge two real estates, and to maintain the property number of one of them.

1. Select the job to work with. See Working With Jobs (page 36).
2. Start the Merge Real Estates workflow.
3. Select Reuse Old Property Number.
4. In the map, select the real estate whose property number you want to maintain, by clicking somewhere within the area.
5. Select the second real estate.
6. Digitize the position (centroid) of the new real estate.
7. Specify the label position and orientation.
8. The Property form opens. Enter your optional attributes, and close the form.
9. In the workflow pane, click Close.

The workflow removes the old boundary lines, and replaces the two old centroids by the new one, so the topology remains valid. To check the

Topology, click the Topology Checker icon, and select the LM_REAL_ESTATE topology. See also Area Topology Checker.

Real estate feature classes

Using the Merge Real Estates workflow you update features as shown in the following table.

Feature ClassName	Caption	Description
LM_OW_PROPERTY	Property	Property parent feature. The workflow deletes the property features of the existing real estates, and creates a new features.
LM_OW_REAL_ESTATE	Real Estate	Real estate centroid. If you do not reuse the property number, the workflow deletes the old centroids, and creates a new one.
LM_OW_REAL_ESTATE_L	Real Estate L	Real estate boundary lines. The workflow deletes adjacent boundary lines.

Create Maintenance Job

Use the Create Maintenance Job workflow to modify or edit features without using a job perimeter. This is a simplified workflow, for example, to correct topology errors, or you move a label in the map. Then, you change the job state to Pending, and to Live at the same time.

Restrictions: You cannot create or modify features that are related to a job perimeter (LM_AD_MUTPERIMETER). That means, you cannot modify feature classes that have the attribute FID_AD_MUTPERIMETER. For example, you cannot modify property, land cover centroid, canton centroid, or boundary point. However you can modify the land cover edges, the canton edges, the real estate edges, or the labels.

Other than the standard workflows, the Create Maintenance workflow uses Pessimistic Feature Locking. If you modify a feature in a Create Maintenance workflow, the feature is locked, and it cannot be modified in another job until you set the Maintenance Job to Live.

To create a maintenance job

1. Start the Create Maintenance Job workflow.
2. In the workflow pane, under Create New Job, select the job template.

3 Enter a job name, such as Label Adjustment.

4 In the workflow pane, click Create.

The maintenance job is selected on the Home tab ➤ Data Source panel.

See also:

- Working With Jobs (page 36)

Land Management Data Model

5

Explore the Land Management Data Model

The Topobase data model administrator lets you explore land management topics, feature classes, domains, topologies, and intersections.

To explore the Land Management (Switzerland) data model

1. Start Topobase Administrator and open the Land Management workspace.
2. Select the document and click Document menu ➤ Data Model.
3. In the data model explorer, click the topics and feature classes.

The data model stores a description for each attribute. Use the data model explorer to review the description.

To view the feature attribute description

1. In the data model explorer, select the feature class.
2. In the right pane, select the attribute.
3. Right-click, and click Edit Attribute.

The Edit Attributes dialog box displays the description.

Data Model Report

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 Land Management 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.

You can use the SQL Sheet to export the data model description into Excel.

To export a data model description to Excel

1. Start Topobase Administrator and open the Land Management workspace.
2. Click Workspace menu ➤ SQL Sheet.
3. In the SQL Sheet, execute the following command.

```
excel select a.name Topic_Name,
a.caption Topic_Caption,
b.f_class_name FeatureClass_Name,
b.caption FeatureClass_Caption,
c.name Attribute_Name,
c.caption Attribute_Caption,
c.description Attribute_Description
from tb_topic a, tb_dictionary b,
tb_attribute c
where a.id = b.topic_id and b.f_class_id = c.f_class_id order by
a.name, b.f_class_name, c.name
```

Land Management Feature Classes

Topobase Land Management (Switzerland) provides a data model that is compliant with the Swiss federal Interlis data models. The data model contains topics, feature classes, domains, topologies, and intersections.

Use Topobase Administrator to create a document containing the Land Switzerland data model. See also Data Model Administrator: Creating a Database.

See also:

- Data Model Description

Land Management: Administrative

The Administrative topic contains feature classes that store canton boundaries, district boundaries, municipality boundaries, municipality names, boundary points, lots, and the job perimeter.

The data model provides area topologies for each boundary type. That means that for each boundary type several feature classes are provided, a centroid feature class, and a line string feature class. The topology automatically generates polygons that are stored in a polygon feature class.

- Centroid feature class—Stores the meta data.
- Line string feature class—Stores the boundary line.
- Polygon feature class—Stores the polygon. Generated automatically for valid topologies.

District

A district contains several municipalities.

Feature ClassName	Caption	Description
LM_AD_DISTRICT_BOUNDARY	District Boundary	Centroid feature class that stores the attribute data.
LM_AD_DISTRICT_BOUNDARY_L	District Boundary L	Line string feature class that stores the district boundary lines.

Feature ClassName	Caption	Description
LM_AD_DISTRICT_TSUR		Polygon feature class. Generated automatically for valid topologies. Used to display the district areas in the map.

Canton

A canton contains several districts.

Lot

Polygon feature class to store contract sections, or project units.

- Feature class — LM_AD_LOT
- Feature type — Polygon

Municipality Projected

Projected municipality boundaries are stored in the line string feature class LM_AD_COUNTRY_BOUNDARY. They are related to the municipality parent feature class. See Municipality (page 68).

Country Boundary

Country boundaries are stored in the line string feature class LM_AD_MUNICIP_BOUND_PROJ. A projected boundary is related to a job perimeter, and to a municipality name.

Municipality

A municipality is the smallest administrative unit of the land management data model. A district contains multiple municipalities.

Municipality data is stored in three feature classes.

Municipality Boundary

- Feature class — LM_AD_MUNICIP_BOUNDARY

- Feature type — Centroid

Name	Data Type	Description
AREA	Number	The area size, measured in document area units, such as square meters.
DATE_LEGALIZED	Date	Date when the item has been legalized in the register office.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_AD_MUNICIPALITY	Number	Municipality of the item. Relation to LM_AD_MUNICIPALITY.
GEOM	Geometry	Feature geometry point. For more information, see *Oracle Spatial Users Guide and Reference*.
ORIENTATION	Number	
UFID	Number	Unique FID used to identify features during external mutation process.

Municipality Line String

- Feature class — LM_AD_MUNICIP_BOUNDARY_L
- Feature type — Line string

Name	Data Type	Description
GEOM	Geometry	Feature geometry point. For more information, see *Oracle Spatial Users Guide and Reference*.
ID_VALIDITY	Number	Validity of the item. Relation to LM_VALIDITY_TBD.
LENGTH	Number	Length of the line.
UFID	Number	Unique FID used to identify features during external mutation process.

Municipality

- Feature class — LM_AD_MUNICIPALITY
- Feature type — Attribute
- Related table — LM_AD_MUNICIPALITY_TBL; label feature class

Name	Data Type	Description
CANTON_NUMBER	Number	Canton number.
FOSNR	Number	Federal Office of Statistic number
NAME_NUMBER	VarChar2	Name of the item.
UFID	Number	Unique FID used to identify features during External Mutation Process.

Mutation Perimeter

A job mutation perimeter (mutation perimeter) stores the area where a job is performed. See also Job Mutation Perimeter (page 38). For several objects, Interlis data exchange requires that the relation to the job mutation perimeter is stored in the attribute FID_AD_MUTPERIMETER. Interlis data exchange uses the mutation perimeter to track feature modifications.

- Feature class — LM_AD_MUTPERIMETER
- Feature type — Polygon

Name	Data Type	Description
AREA	Number	Area of the polygon.
DATE_CREATED	Date	Date when the item has been created.
DATE_FINALIZED	Date	Date when the item has been finalized.
DATE_LEGALIZED	Date	Date when the item has been legalized in the register office.

Name	Data Type	Description
DATE_REGISTERED	Date	Date when the item has been registered in the register office.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
FID_PARENT_MUT-PERIMETER	Number	Perimeter of this item.
IDENTIFICATION	VarChar2	Identification of the item.
ID_EQUIDISTANCE	Number	Equidistance for the height perimeter.
ID_JOB	Number	Relation to the system job Id.
ID_ORIGIN	Number	Origin of the point
ID_STATE	Number	State of the item
JOB_VERSION	Number	For job enabled feature classes. Specifies the version of the feature.
UFID	Number	Unique FID used to identify features during external mutation process.

Land Management: Building

The Building topic contains feature classes that store building information.

House Name

House names are related to the house entrances. They can be provided for different languages.

- Feature class — LM_BU_HOUSE_NAME
- Feature type — Attribute.
- Related table — LM_BU_HOUSE_NAME_TBL, label feature class.

House Description

House descriptions are related to the house entrances. They can be provided for different languages.

- Feature class — LM_BU_HOUSE_DESCRIPTION
- Feature type — Attribute.

See also:

- Managing Buildings (page 2)
- Create Building (page 51)

Building

LM_BU_BUILDING is the parent feature class that is related to different building elements, such as the building geometry, or house entrance.

See also Managing Buildings (page 2)

- Feature class — LM_BU_BUILDING
- Feature type — Point
- Related table — LM_BU_BUILDING_TBL, label feature class.

Name	Data Type	Description
BUILDING_NUMBER	Number	Building number.
FID_AD_MUNICIPALITY	Number	Municipality of the item. Relation to LM_AD_MUNICIPALITY.
FID_IDENTID	Number	Numbering domain Identifier. Part of a unique identification of all Features. Relation to LM_ND_NUMBER_DOMAIN.
FID_OW_PROPERTY	Number	Property of the item. Relation to LM_OW_PROPERTY.
GEOM	Geometry	Feature geometry.
ID_STATE	Number	State of the item. Relation to LM_STATE_TBD.

Name	Data Type	Description
REGBL_EGID	Number	EGID. Unique identifier for houses.

Building Projected

Projected buildings are stored in the polygon feature class LM_BU_SURFACE_PROJ. They are related to the building parent feature class. See Building (page 70) and Create Projected Building (page 56).

Name	Data Type	Description
AREA	Number	Area of the projected building.
DATE_ACQUIRED	Date	Date when the item has been acquired.
DATE_COMPLETED	Date	Date when the item has been completed.
DATE_CREATED	Date	Date when the item has been created.
GEOM	Geometry	Feature geometry.
FID_BU_BUILDING	Number	Building parent feature.
FID_BU_PLANNING_PERMISSION	Number	Planning permission.

House Entrance

A house entrance stores the location of a building. See also Managing Buildings (page 2).

- Feature class — LM_BU_HOUSE_ENTRANCE
- Feature type — Point.
- Related table — LM_BU_HOUSE_ENTRANCE_TBL, label feature class.

Name	Data Type	Description
ADDRESS_CODE	Varchar2	Address code of the item.

Name	Data Type	Description
CHANGING	Number	Defines if the item is currently being modified or not.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_BU_BUILDING	Number	Building of the item.
FID_LO_LOCATION		Location of the item.
HOUSE_LEVEL	Number	House level.
HOUSE_NUMBER	Varchar2	House number.
ID_APPROVAL	Number	Approval state of the item. Relation to LM__APPROVAL_TBD.
ID_STATE	Number	State of the item. Relation to LM_STATE_TBD.
IS_TEMPORARY_ATTRIBUTE	Number	Temporary attribute of the item.
OFFICIAL_DESIGNATION	Number	Official designation of the item.
POLICE_NUMBER	Varchar2	Police number of the item.
REGBL_EDID	Number	EDID. Unique identifier for houses.
UFID	Number	Unique FID used to identify features during external mutation process.

Building Insurance

Several feature classes administer building insurance information.

- LM_BU_BUILD_BUILD_INSURANCE —System table to administer multiple building insurance features per building.
- LM_BU_BUILIDING_INSURANCE — Stores insurance data.

- Label feature classes.

Building Insurance Data

- Feature class — LM_BU_BUILDING_INSURANCE
- Feature type — Attribute

Name	Data Type	Description
BUILDING_AREA	Number	Area of the building.
BUILDING_CODE	Number	Building code of the item.
BUILDING_CODENAME	Varchar2	Name of the building code.
BUILDING_DISTRICTNUMBER	Varchar2	Building district number of the item.
INSURANCE_NUMBER	Varchar2	Insurance number of the item.
DESIGNATION	Varchar2	Designation of the item.
ID_BUILDING_TYPE1 ID_BUILDING_TYPE2 ID_BUILDING_TYPE3 ID_BUILDING_TYPE4 ID_BUILDING_TYPE5	Number	Building type of the item.
UFID	Number	Unique FID used to identify features during external mutation process.

A building can have multiple insurance contracts. The data model uses a system table to connect the insurance data to a building.

- Feature class — LM_BU_BUILD_BUILD_INSUR
- Feature type — Attribute:

- Related table —LM_BU_BUILD_INSUR_TBL; label feature class.

Name	Data Type	Description
FID_BU_BUILDING	Number	FID of the building. Relation to LM_BU_BUILDING.
FID_BU_BUILDING_INSURANCE	Number	FID of the building insurance. Relation to LM_BU_BUILDING_INSURANCE.

Land Management: Construction

Construction feature classes store DTM (digital terrain model) data, such as DTM edges and points, and construction data such as axe lines and points.

- Feature classes— LM_CO_CONS_*
- Feature classes— LM_CO_DTM_*

Land Management: Control Points

Topobase Land Management provides feature classes for planimetric control points, and for altimetric control points.

Altimetric control point

- Feature class — LM_CP_ACP
- Feature type — Point
- Related table — LM_CP_ACP_TBL, label feature class.

Name	Data Type	Description
DESIGNATION	Varchar2	Designation of the item
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.

Name	Data Type	Description
ID_CATEGORY	Number	Category of the item.
ID_ORIGIN	Number	Origin of the point. Relation to LM_ORIGIN_TBD. Stores how the point has been determined, such as GPS, Project, Constructed, or Digitized.
ID_POINT_MARK	Number	Mark that allows the point to be recovered in the field.
ID_PROJECT	Number	Project category of the point.
MUTATION_NUMBER	Varchar2	Mutation number.
TB_ACCURACY_HEIGHT TB_ACCURACY_POSITION TB_HEIGHT_RELIABLE TB_POSITION_RELIABLE TB_RELIABILITY_HEIGHT TB_RELIABILITY_POSITION	Number	Store the accuracy, and reliability of the point (in position and height). See also the Topobase Administrator Guide, section Feature Class Type: Point.
TEMP_NUMBER	Varchar2	Temporary number of the point.
TB_POINTNUMBER	Varchar2	Point number. See also Topobase Point Numbering.
UFID	Number	Unique FID used to identify features during external mutation process.

Planimetric control point

- Feature class — LM_CP_PCP
- Feature type — Point

- Related table — LM_CP_PCP_TBL, label feature class.

Name	Data Type	Description
DATE_ALTIMETRY_CHANGED	Date	Date when the altimetry of the item has been changed.
DESIGNATION	Varchar2	Designation of the item
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_ACCESSIBILITY	Number	Accessibility of the item.
ID_CATEGORY	Number	Category of the item.
ID_ORIGIN	Number	Origin of the point. Relation to LM_ORIGIN_TBD. Stores how the point has been determined, such as GPS, Project, Constructed, or Digitized.
ID_POINT_MARK	Number	Mark that allows the point to be recovered in the field.
ID_PROJECT	Number	Project category of the point.
ID_PROTECTION_TYPE	Number	Defines the type of the protection for the item.
MUTATION_NUMBER	Varchar2	Mutation number.
NARRATIVE	Varchar2	Commentary / Observation related to the point.
PROTOCOLE	Number	Specifies if there is a protocol which helps to recover the point.
SPECIAL_OLD_MARK	Number	Specifies if the item owns a special old mark.
TB_ACCURACY_HEIGHT	Number	Store the accuracy, and reliability of the point (in position and height). See also the Topobase Ad-

Name	Data Type	Description
TB_ACCURACY_POSITION TB_HEIGHT_RELIABLE TB_POSITION_RELIABLE TB_RELIABILITY_HEIGHT TB_RELIABILITY_POSITION		ministrator Guide, section Feature Class Type: Point.
TEMP_NUMBER	Varchar2	Temporary number of the point.
TB_POINTNUMBER	Varchar2	Point number. See also Topobase Point Numbering.
UFID	Number	Unique FID used to identify features during external mutation process.

Land Management: Intersection

Intersection feature classes store the intersection results. See also Land Management: Intersections (page 14)Data Model: Intersections.

- Feature classes — LM_IS_*

Land Management: Heights

Height information, such as contour lines, points, and edges are stored in the feature classes of the topic Height. For example, to generate a local DTM.

- Feature class — LM_HE_*
- Feature type — Polygon, line string, point

Land Management: Land Cover

The Land Cover topic contains the feature classes that store the land cover information. See also Managing Land Cover (page 4).

Classification

Manages the land cover codes of the land cover centroids.

- Feature class — LM_LC_CLASSIFICATION
- Feature type — Attribute

Code

- Feature class — LM_LC_CODE
- Feature type — Attribute

Object Name

- Feature class — LM_LC_OBJECT_NAME
- Feature type — Attribute
- Related table — LM_LC_OBJECT_NAME_TBL, label feature class.

Object Number

- Feature class — LM_LC_OBJECT_NUMBER
- Feature type — Attribute
- Related table — LM_LC_OBJECT_NUMBER_TBL, label feature class.

Projected Classification

- Feature class — LM_LC_CLASSIFICATION_PROJ
- Feature type — Attribute
- Related table — LM_LC_OBJECT_NAME_TBL, label feature class.

Symbol

Symbols are points that are related to a land cover centroid.

- Feature class — LM_LC_SYMBOL
- Feature type — Point
- Related table — LM_LC_OBJECT_NAME_TBL, label feature class.

Single Point

- Feature class — LM_LC_SINGLE_POINT
- Feature type — Point
- Related table — LM_LC_SINGLE_POINT_TBL, label feature class.

Name	Data Type	Description
DOSSIER	Varchar2	Dossier name of the item.
EXACTLY_DEFINED	Number	Specifies if the point can be exactly defined or not.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_ORIGIN	Number	Origin of the point. Relation to LM_ORIGIN_TBD. Stores how the point has been determined, such as GPS, Project, Constructed, or Digitized.
ID_POINTTYPE	Number	Point type.
ID_QUALITY	Number	Quality, according to Swiss federal categories. Related to LM_QUALITY_TBD.
MUTATION_NUMBER	Varchar2	Mutation number.
NARRATIVE	Varchar2	Commentary / Observation related to the point.

Name	Data Type	Description
TB_ACCURACY_HEIGHT TB_ACCURACY_POSITION TB_HEIGHT_RELIABLE TB_POSITION_RELIABLE TB_RELIABILITY_HEIGHT TB_RELIABILITY_POSITION	Number	Store the accuracy, and reliability of the point (in position and height). See also the Topobase Administrator Guide, section Feature Class Type: Point.
TB_POINTNUMBER	Varchar2	Point number. See also Topobase Point Numbering.
UFID	Number	Unique FID used to identify features during external mutation process.

Surface and Surface L

The Land Cover topic contains an area topology that provides consistent and non overlapping land cover units. For example, forests, water features, or asphalt areas. See also Working with Area Topologies (page 34).

- Feature class — LM_LC_SURFACE
- Feature type — Centroid
- Feature class — LM_LC_SURFACE_L
- Feature type — Line string
- Topology — LM_LAND_COVER.

Surface - centroid

The centroid feature class stores the land cover attributes.

Name	Data Type	Description
AREA	Number	Area. Determined by the topology.

Name	Data Type	Description
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_BU_BUILDING	Number	Building of the item. For buildings, stores the relation to the building parent feature.
ID_LC_TYPE	Number	Land cover type.
ID_QUALITY	Number	Quality, according to Swiss federal categories. Related to LM_QUALITY_TBD.
SYSTEM_CREATED	Number	System attribute, updated by the split functionality. Used in display models to style lines. Managed by a feature rule; see SetSystemCreated_BU.
UFID	Number	Unique FID used to identify features during external mutation process.

Surface L - Line string

The line string feature stores the land cover boundaries. It does not store any land cover attributes, except line styles for the use in the display model.

Name	Data Type	Description
ID_LINETYPE	Number	Line type.
ID_LINETYPE_OV	Number	Line type for the Overview plan.
UFID	Number	Unique FID used to identify features during external mutation process.

See also:

- Managing Land Cover (page 4)

Land Management: Land Slide

The Land Slide topic stores land slide areas. Land slide areas are being legalized in the register office.

- Feature class — LM_LS_LAND_SLIDE
- Feature type — Polygon
- Related table — LM_LS_LAND_SLIDE_TBL, label feature class.

Land Management: Location

The Location topic contains feature classes that store locations. See also Managing Locations (page 8).

Locality

- Feature class — LM_LO_LOCALITY
- Feature type — Polygon
- Related to — LM_AD_MUTPERIMETER, LM_LO_LOCALITY_GROUP.

Locality Group

- Feature class — LM_LO_LOCALITY_GROUP
- Feature type — Attribute

Locality Name

- Feature class — LM_LO_LOCALITY_NAME
- Feature type — Attribute
- Related to — LM_LO_LOCALITY.

Name	Data Type	Description
FID_LO_LOCALITY	Number	Locality of the item.
ID_LANGUAGE	Number	Language
INDEX_TEXT	Varchar2	Index text of the item.

Name	Data Type	Description
SHORT_TEXT	Varchar2	Short text of the item.
TEXT	Varchar2	Text.
UFID	Number	Unique FID used to identify features during external mutation process.

NPA6

- Feature class — LM_LO_NPA6
- Feature type — Polygon
- Related to — LM_LO_LOCALITY.

Name	Data Type	Description
ADDITIONAL_NUMBER	Number	Additional number of the item.
CHANGING	Number	Indicates whether the item is currently being modified or not.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_LO_LOCALITY	Number	Locality parent feature.
ID_STATE	Number	State of the item. Related to LM_LO_STATE_TBD.
NPA	Number	Postal address number.
UFID	Number	Unique FID used to identify features during external mutation process.

Location

Location is the parent feature class that stores general information about locations such as named areas, or road classification.

- Feature class — LM_LO_LOCATION

- Feature type — Attribute

Name	Data Type	Description
CHANGING	Number	Specifies whether the item is currently being modified or not.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_NAME_TYPE	Number	Name type of the item. Relation to LM_LO_NAMETYPE_TBD.
ID_NUMBERING_PRINCIPLE	Number	Way to give a number. Relation to LM_LO_NUMBERING_PRINCIP_TBD.
ID_STATE	Number	State of the item. Relation to LM_LO_STATE_TBD.
ID_TYPE	Number	Type of the item. Relation to LM_LO_STREET_CATEGORY_TBD.
IS_TEMPORARY_ATTRIBUTE	Number	Indicates whether the item is temporary.
LOCATION_NUMBER	Varchar2	Location number.
OFFICIAL_DESIGNATION	Number	Official designation of the item.
UFID	Number	Unique FID used to identify features during external mutation process.

Named Area

The feature class stores the geometry of the named area. Other attributes are stored in the Location parent feature LM_LO_LOCATION.

- Feature class — LM_LO_NAMED_AREA
- Feature type — Polygon

Road Classification

The feature class stores the geometry of the road classification areas, and the road classification type (according to federal regulations). Other attributes are stored in the Location parent feature LM_LO_LOCATION.

- Feature class — LM_LO_ROAD_CLASSIFICATION
- Feature type — Polygon

Road Section

The feature class stores the geometry of the road sections. Other attributes are stored in the Location parent feature LM_LO_LOCATION.

- Feature class — LM_LO_ROAD_SECTION
- Feature type — Line string

Name	Data Type	Description
DISTANCE	Number	Distance.
FID_LO_LOCATION	Number	Location of the item.
IS_AXE	Number	Indicates whether the item is an axe or not.
ORDERING	Number	Order of the item.
UFID	Number	Unique FID used to identify features during external mutation process.

Target Point

The feature class is related to the road sections. LM_LO_LOCATION.

- Feature class — LM_LO_TARGET_POINT
- Feature type — Point

Name	Data Type	Description
FID_LO_ROAD_SECTION	Number	Road section of the item.

Name	Data Type	Description
POINT_NUMBER	Varchar2	Number of the item.

Way

The feature class stores the geometry of the ways. Other attributes are stored in the Location parent feature LM_LO_LOCATION.

- Feature class — LM_LO_WAY
- Feature type — Line string

Name	Data Type	Description
FID_LO_LOCATION	Number	Location parent feature.
ID_WAY_TYPE	Number	Way type of the item. Related to LM_LO_WAY-TYPE_TBD.

Location Name

The feature class Location Name administers the location names of multiple languages.

- Feature class — LM_LO_LOCATION_NAME
- Feature type — Attribute
- Related table — LM_LO_LOCATION_NAME_TBL, label feature class.

Name	Data Type	Description
FID_LO_LOCATION	Number	Location.
ID_LANGUAGE	Number	Language.
INDEX_NAME	Varchar2	Index name of the item.
LOCATION_NAME	Varchar2	Location name of the item.
SHORT_NAME	Varchar2	Short name.

Name	Data Type	Description
UFID	Number	Unique FID used to identify features during external mutation process.

Land Management: Names

The Names topic contains feature classes that store named areas, such as local names, and places, compliant with the Swiss federal norm. See also Managing Names (page 8).

The data model provides an area topology for the local names. That means that several feature classes are provided, a centroid feature class, and a line string feature class. The topology automatically generates polygons that are stored in a polygon feature class.

- Centroid feature class—Stores the meta data.
- Line string feature class—Stores the boundary line.
- Polygon feature class—Stores the polygon. Generated automatically for valid topologies.

Local Name

The feature class Local Name is part of the Local Name area topology that provides consistent and non overlapping local name areas.

- Feature class — LM_NA_LOCAL_NAME
- Feature type — Centroid
- Related table — LM_NA_LOCAL_NAME_TBL, label feature class.
- Topology—LM_LOCAL_NAME

Name	Data Type	Description
AREA	Number	Area. For valid topologies, the area is calculated by the topology triggers.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.

Name	Data Type	Description
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_NAME_ORIGIN	Number	Origin of the item. Related to LM_NA_NAME_ORIGIN_TBD.
LN_NAME	Varchar2	Name.
LN_NUMBER	Varchar2	Number.
UFID	Number	Unique FID used to identify features during external mutation process.

Local Name L

The feature class Local Name L stores the local name boundary geometry.

- Feature class — LM_NA_LOCAL_NAME_L
- Feature type — Line string
- Topology—LM_LOCAL_NAME

Named Locality

The feature class Named Locality stores names that are not related to a geometry feature.

- Feature class — LM_NA_NAMED_LOCALITY
- Feature type — Attribute
- Related table — LM_NA_NAMED_LOCALITY_TBL, label feature class.

Name	Data Type	Description
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.

Name	Data Type	Description
ID_NAME_ORIGIN	Number	Origin of the item. Related to LM_NA_NAME_ORIGIN_TBD.
NL_NAME	Varchar2	Name.
NL_NUMBER	Varchar2	Number.
UFID	Number	Unique FID used to identify features during external mutation process.

Place Name

- Feature class — LM_NA_PLACE_NAME
- Feature type — Polygon
- Related table — LM_NA_PLACE_NAME_TBL, label feature class.

Name	Data Type	Description
CATEGORY	Varchar2	Category.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_NAME_ORIGIN	Number	Origin of the item. Related to LM_NA_NAME_ORIGIN_TBD.
PL_NAME	Varchar2	Name.
PL_NUMBER	Varchar2	Number.
UFID	Number	Unique FID used to identify features during external mutation process.

Land Management: Numbering Domain

The topic stores numbering domain information. For some workflows, the numbering domain is used to determine unique numbers, or names. For example, when you create a property, the numbering domain is determined by the attributes ID_CANTON, NUMBERND, where the category is Municipal Level (ID_NDCATEGORY=4).

See also Create Property (page 39) and Create Name (page 47).

Numbering Domain

- Feature class — LM_ND_NUMBER_DOMAIN
- Feature type — Attribute
- Related table — LM_ND_NUMBER_DOMAIN_TBL, label feature class.

NDGeometry

- Feature class — LM_ND_NDGEOMETRY
- Feature type — Polygon

Land Management: Ownership

The Ownership topic contains feature classes that store ownership information compliant with the Swiss federal norm. See also Managing Ownership (page 5).

Ownership information

- Boundary points that determine the real estate boundaries.
- Properties, such as
  - Development rights (DPR). DPRs can overlap real estate areas.
  - Mines. Mines can overlap real estate areas.
  - Real estates that cover the project area consistently.
  - Projected objects are stored in the LM_OW_*_PROJ feature classes.

Boundary Points

See also Managing Boundary Points (page 6).

- Feature class — LM_OW_BOUNDARYPOINT
- Feature type — Point
- Related table — LM_OW_BOUNDARYPOINT_TBL, label feature class.

Name	Data Type	Description
CANTON_POINT	Number	Defines whether the point is also a canton point.
DESIGNATION	Varchar2	Designation of the item
DISPLAY	Number	Defines whether the point has to be displayed or not.
DISTRICT_POINT	Number	Defines whether the point is also a district point.
EXACTLY_DEFINED	Number	Defines whether the point can be defined exactly.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_BOUNDARY_CONTENT	Number	Defines whether the point is located on a municipality border.
ID_ORIGIN	Number	Origin of the point. Relation to LM_ORIGIN_TBD. Stores how the point has been determined, such as GPS, Project, Constructed, or Digitized.
ID_POINT_MARK	Number	Mark that allows the point to be recovered in the field.
ID_POINTTYPE	Number	Point type.
ID_PROJECT	Number	Project category of the point.

Name	Data Type	Description
MUTATION_NUMBER	Varchar2	Mutation number.
NARRATIVE	Varchar2	Commentary or observation related to the item.
QUALITY	Number	Quality of the point.
SPECIAL_OLD_MARK		Specifies whether the point has a special old mark.
TB_ACCURACY_HEIGHT TB_ACCURACY_POSITION TB_HEIGHT_RELIABLE TB_POSITION_RELIABLE TB_RELIABILITY_HEIGHT TB_RELIABILITY_POSITION	Number	Store the accuracy, and reliability of the point (in position and height). See also the Topobase Administrator Guide, section Feature Class Type: Point.
TEMP_NUMBER	Varchar2	Temporary number of the point.
TB_POINTNUMBER	Varchar2	Point number. See also Topobase Point Numbering.
VALUE	Varchar2	Value of the item.
VALUE_2	Number	Additional value of the item.
UFID	Number	Unique FID used to identify features during external mutation process.

Property

The Property feature class is the parent feature class of several property types.

- Feature class —LM_OW_PROPERTY

- Feature type — Attribute
- Related table — LM_OW_PROPERTY_TBL, label feature class.

Name	Data Type	Description
AREA_TOTAL	Number	Total Area of the item:
EGRIS_EGRID	Varchar2	unique identifier for houses.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_COMPLETENESS	Number	Specifies the completeness of the item.
ID_PROPERTY_TYPE	Number	Property type of the item, such as real estate, mine, DPR. Related to LM_OW_PROP_CATEGORY_TBD.
ID_QUALITY	Number	Quality, according to Swiss federal categories. Related to LM_QUALITY_TBD.
ID_VALIDITY	Number	Validity, such as Proivisional, or Legally Valid. Related to LM_VALIDITY_TBD.
LEGALLY_VALID	Varchar2	Indicates whether the item is legally valid.
ID_ORIGIN	Varchar2	Origin of the item.
PROPERTY_INDEX	Varchar2	Property index of the item.
PROPERTY_NUMBER	Varchar2	Property number.
UFID	Number	Unique FID used to identify features during external mutation process.

Property Address

- Feature class —LM_OW_PROPERTY_ADDRESS

- Feature type — Attribute

Name	Data Type	Description
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
FID_LO_LOCATION	Number	Location of the item.
FID_OW_PROPERTY	Number	Property parent feature of the item.
ID_NAME_TYPE	Number	Name type, such as place name, local name, or street name. Related to LM_LO_NAMETYPE_TBD.
PROPERTY_NUMBER	Varchar2	Property number.
STREETNAME_NUMBER	Varchar2	Street name number of the item.

DPR

The feature class DPR stores areas that are subject to specific development rights. DPR is a compound polygon; see also Data Model: Compounds.

- Feature class —LM_OW_DPR
- Feature type — Compound polygon (parent)
- Feature class — LM_OW_DPR_L
- Feature type — Compound polygon (single segment)

Name	Data Type	Description
AREA	Number	Exact area that is determined by the topology.
AREA_REGISTER	Number	Area value that is used in the register office. Usually this a a rounded value.
DPR_LEVEL	Number	Level of the item.
FID_OW_PROPERTY	Number	Property parent featur of the item.

Name	Data Type	Description
INVALID_GEOMETRY_ERROR	Varchar2	Indicates whether the compound geometry is valid.
PROPERTY_INDEX	Varchar2	Property index of the item.
UFID	Number	Unique FID used to identify features during external mutation process.

Mine

The feature class Mine is a compound polygon; see also Data Model: Compounds.

- Feature class —LM_OW_MINE
- Feature type — Compound polygon (parent)
- Feature class — LM_OW_MINE_L
- Feature type — Compound polygon (single segment)

Name	Data Type	Description
AREA	Number	Exact area that is determined by the topology.
AREA_REGISTER	Number	Area value that is used in the register office. Usually this a a rounded value.
FID_OW_PROPERTY	Number	Property parent featur of the item.
INVALID_GEOMETRY_ERROR	Varchar2	Indicates whether the compound geometry is valid.
PROPERTY_INDEX	Varchar2	Property index of the item.
UFID	Number	Unique FID used to identify features during external mutation process.

Real Estate and Real Estate L

The Ownership topic contains an area topology that provides consistent and non overlapping real estate units. See also Working with Area Topologies (page 34).

- Feature class — LM_OW_REAL_ESTATE
- Feature type — Centroid
- Feature class — LM_OW_REAL_ESTATE_L
- Feature type — Line string
- Topology — LM_REAL_ESTATE.

Real Estate - centroid

The centroid feature class stores the real estate attributes.

Name	Data Type	Description
AREA	Number	Exact area that is determined by the topology.
AREA_REGISTER	Number	Area value that is used in the register office. Usually this a a rounded value.
FID_OW_PROPERTY	Number	Property parent featur of the item.
PROPERTY_INDEX	Varchar2	Property index of the item.
UFID	Number	Unique FID used to identify features during external mutation process.

Real Estate L - Line string

The line string feature stores the real estate boundaries. It does not store any land real estate attributes, except some attributes that are used for line stylization in the display model.

Name	Data Type	Description
DISPLAY_OV	Number	Indicates whether the line has to be displayed in the overview plan.
ID_LINETYPE	Number	Line type. Related to LM_OW_RE_LINETYPE_TBD.

Name	Data Type	Description
ID_VALIDITY	Number	Validity, such as Proivisional, or Legally Valid. Related to LM_VALIDITY_TBD.
SYSTEM_CREATED	Number	System attribute, updated by the split functionality. Used in display models to style lines. Managed by a feature rule; see SetSystemCreated_BU.
UFID	Number	Unique FID used to identify features during external mutation process.

Land Management: Pipelines

The topic Pipelines provides feature classes to store pipe objects compliant with the Swiss federal norm. For example gas pipes, or oil pipes.

Pipe objects consist of one parent feature that stores the attributes, and one or more geometry elements. A pipe object can be of different geometry type, such as point, line string, or compound polygon.

- Feature class — LM_PI_*

Land Management: Plan Repartition

The topic Plan Repartition provides feature classes to administer plans.

Plan attributes

- Feature class — LM_PL_PLAN
- Feature type — Attribute

Plan repartition topology

The Plan Repartition topic contains an area topology that provides consistent and non overlapping plan units. See also Working with Area Topologies (page 34).

- Feature class — LM_PL_PLAN_GEOMETRY
- Feature type — Centroid

- Feature class — LM_PL_PLAN_GEOMETRY_L
- Feature type — Line string
- Topology — LM_PLAN.

Land Management: Planning

The topic Planning provides feature classes to store planned objects.

- Feature class — LM_PA_*

Land Management: Public Ownership

The topic Public Ownership contains a feature class to store forest boundaries.

- Feature class — LM_PO_FOREST_LIMIT
- Feature type — Line string.

Land Management: Servitudes

The topic Servitudes provides <tbd>.

- Feature classes — LM_SE_*

Land Management: Single Objects

The topic Single objects stores objects of different geometry type, such as trees, bridges, green houses. See also Managing Single Objects (page 7).

Single Object

The feature class Single Object is the parent feature class of single objects that can be of different geometry types. The single object geometry is stored in separate feature classes. See also Managing Single Objects (page 7).

- Feature class — LM_SO_SINGLE_OBJECT

- Feature type — Attribute
- Related table — LM_SO_SINGLE_OBJECT_TBL,label feature class.

Name	Data Type	Description
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_BU_BUILDING	Number	Building parent feature. Underground buildings are stored in the Single Object feature class.
ID_APPROVAL	Number	Approval state of the item.
ID_QUALITY	Number	Quality, according to Swiss federal categories. Related to LM_QUALITY_TBD.
ID_TYPE	Number	Type, such as wall, underground building, chimney. Related to LM_SO_OBJECT_CATEGORY_TBD.
UFID	Number	Unique FID used to identify features during external mutation process.

Line Element

- Feature class — LM_SO_LINE_ELEMENT
- Feature type — Line string.

The feature class stores the line geometry of a single object. It is related to the Single Object parent feature, and does not store any single object attributes, except some attributes that are used for line stylization in the display model.

Point Element

- Feature class — LM_SO_POINT_ELEMENT
- Feature type — Point.

The feature class stores the point geometry of a single object. It is related to the Single Object parent feature, and does not store any single object attributes, except some attributes that are used for line stylization in the display model.

Surface Element

- Feature class — LM_SO_SURFACE_ELEMENT
- Feature type — Compound Polygon (parent).
- Feature class — LM_SO_SURFACE_ELEMENT_L
- Feature type — Compound polygon (single segment)

The feature class stores the boundary lines of a single object. The compound parent It is related to the Single Object parent feature, and does not store any single object attributes, except some attributes that are used for line stylization in the display model.

Single Object Point

- Feature class — LM_SO_SINGLE_POINT
- Feature type — Point
- Related table — LM_SO_SINGLE_POINT_TBL, label feature class.

Name	Data Type	Description
EXACTLY_DEFINED	Number	Defines whether the point can be defined exactly.
FID_AD_MUTPERIMETER	Number	Perimeter of this item. Relation to LM_AD_MUTPERIMETER.
FID_IDENTND	Number	Numbering domain Identifier. Part of a unique identification of all features.
ID_ORIGIN	Number	Origin of the point. Relation to LM_ORIGIN_TBD. Stores how the point has been determined, such as GPS, Project, Constructed, or Digitized.
ID_TYPE	Number	Type, such as fountain, chimney, or single point. Related to LM_SO_OBJECT_CATEGORY_TBD.
QUALITY	Number	Quality of the point.
TB_ACCURACY_HEIGHT	Number	Store the accuracy, and reliability of the point (in position and height). See also the Topobase Ad-

Name	Data Type	Description
TB_ACCURACY_POSITION TB_HEIGHT_RELIABLE TB_POSITION_RELIABLE TB_RELIABILITY_HEIGHT TB_RELIABILITY_POSITION		ministrator Guide, section Feature Class Type: Point.
TB_POINTNUMBER	Varchar2	Point number. See also Topobase Point Numbering.
UFID	Number	Unique FID used to identify features during external mutation process.

Symbol

- Feature class — LM_SO_SYMBOL
- Feature type — Point

Name	Data Type	Description
FID_SO_SINGLE_OBJECT	Number	Single Object parent feature.
ID_PLAN_CATEGORY	Number	Plan category of the item. Related to LM_SO_PLAN_CATEGORY_TBD.

Code

The feature class LM_SO_CODE administers code descriptions of different languages.

- Feature class — LM_SO_CODE
- Feature type —Attribute

Object Name

The feature class LM_SO_OBJECT_NAME stores single object names. It is related to the Single Object parent feature class.

- Feature class —LM_SO_OBJECT_NAME
- Feature type —Attribute

Object Number

The feature class LM_SO_OBJECT_NUMBER administers object numbers. It is related to the Single Object parent feature class.

- Feature class —LM_SO_OBJECT_NUMBER
- Feature type —Attribute

Land Management: Tolerance Degree

The topic Tolerance Degree contains an area topology that provides consistent and non overlapping zones of the same tolerance degree. See also Working with Area Topologies (page 34).

- Feature class — LM_TD_TOLERANCEDEGREE
- Feature type — Centroid
- Feature class — LLM_TD_TOLERANCEDEGREE_L
- Feature type — Line string
- Topology — LM_TOLERANCE.

Extension Feature Classes

The Land Management (CH) date model optionally contains data model extensions. The extension feature classes are stored in separate topics.

Topics of the extension feature classes

- **Construct**—Contains the COGO feature classes. See the Topobase Client User Guide, section Construction Introduction.
- **Plot**—Contains the Plot feature classes. See the Topobase Administrator Guide, section Plot Extension.
- **Templates**—Contains system feature classes that administer feature templates. See Data Model: Templates.

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