FME Desktop ®

Smallworld 4.3
Training Pathway

FME 2015.1

SAFE SOFTWARE

Revisions

Every effort has been made to ensure the accuracy of this document. Safe Software Inc. regrets any errors and omissions that may occur and would appreciate being informed of any errors found. Safe Software Inc. will correct any such errors and omissions in a subsequent version, as feasible. Please contact us at:

Safe Software Inc.
Suite 2017, 7445 – 132nd Street
Surrey, BC
Canada
V3W1J8

www.safe.com

Safe Software Inc. assumes no responsibility for any errors in this document or their consequences, and reserves the right to make improvements and changes to this document without notice.

Trademarks

FME is a registered trademark of Safe Software Inc.

All brand or product names mentioned herein may be trademarks or registered trademarks of their respective holders and should be noted as such.

Documentation Information

Document Name: FME Desktop Smallworld Training Pathway Manual
FME Version: FME 2015.1 (15510) 32-bit
Operating System: Windows 7 SP-1, 64-bit
Database: Smallworld 4.3.0.7 Core + Cambridge Database
Updated: July 2015

Table of Contents

Introduction

This training material is part of the FME Training Pathway system.

Database Pathway

This training material is part of the FME Training Database Pathway.

The Smallworld training pathway contains advanced content and assumes that the user is familiar with all of the concepts and practices covered by the FME Database Pathway Tutorial, and the FME Desktop Basic Training Course.

FME Version

This training material is designed specifically for use with FME2015.1. You may not have some of the functionality described if you use an older version of FME.

Sample Data

The sample data required to carry out the examples and exercises in this document can be obtained from the training resource page:

- Full FME Training Dataset
- FME for Smallworld Training Data

Smallworld Database

This training includes documented steps for **GE Smallworld Core version 4.3** using the **FME Translator 4.3**. This training uses the Smallworld Cambridge sample database as well as some secondary dataset, see links above.

Older versions of Smallworld or the FME Translator do not support the functionality described in this training. Other Smallworld varieties will differ in their use.

This training assumes the user is familiar with the Smallworld applications for inspecting the Smallworld database and managing *'alternatives'*.
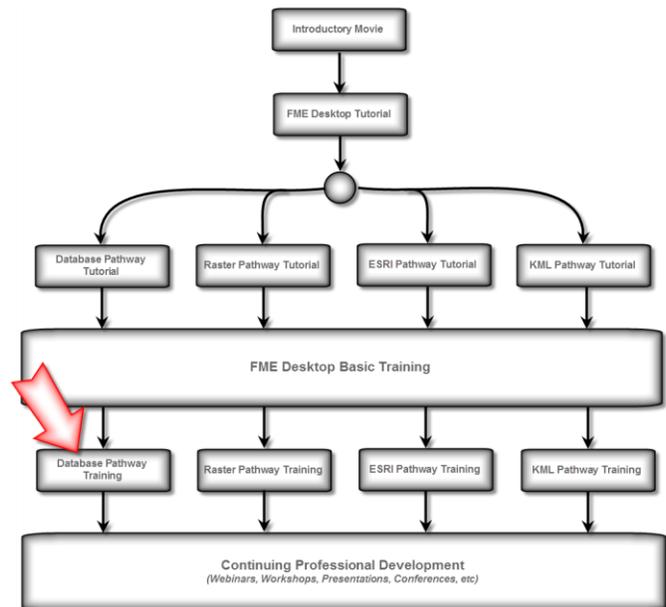
Training Course Resources

The data used in this training is largely based on data from the Smallworld Cambridge Database supplied as part of the Smallworld core product. The Cambridge database is installed on your Amazon AWS computer.

See the section: "**Starting the Smallworld GIS**" on how to start the Smallworld GIS.

Other resources you will be working with are:

Location	Resource
C:\FMEData2015\Resources\Smallworld	Datasets used by this training
C:\FMEData2015\Workspaces\SmallworldTraining	Workspaces used in the exercises
C:\FMEData2015\Output	Good place to direct any exercise output
<documents>/My FME Workspaces	Default location to save FME workspaces

Working with the Smallworld Database and FME

The Smallworld FME translator provides access to Smallworld objects in a form suited to most GIS and CAD applications.

The Smallworld database provides a rich set of functions and capabilities for managing GIS data. Smallworld uses the VMDS database for its data storage. That database structure often needs to be translated into other formats suited to other GIS and CAD applications.

Smallworld objects typically contain multiple geometries, whereas many other GIS formats have limitations on their support for multiple geometries. FME provides a powerful toolkit to translate between Smallworld objects, including multiple geometries, and other GIS formats.

Installation

FME Smallworld Suite consists of several components;
- FME for Smallworld Edition - FME with Smallworld licensing
- Smallworld Translators Suite DVD which contains:
    - Install documentation (swtransins.pdf) and installer
    - Smallworld Translators Suite User Documentation
    - Smallworld FME Translator magik
    - FME dll and FME metafile

Both the Smallworld 4 core and the Smallworld FME Translator must be installed in order for FME to be able to read or write Smallworld data.

The installation components are supplied by GE Smallworld on the Smallworld Translators Suite DVD. Ensure you have the installation pack that matches you're version of Smallworld core product.

Starting the Smallworld GIS
Before you start the FME for Smallworld training, you need to start-up the Smallworld GIS. Here's what you need to do:
1. On your Amazon AWS Remote Desktop – double click or Open the **Start Smallworld** shortcut icon. Two events will happen:
    a. The **emacs** editor will start and open at a MagikSF> command prompt
    b. The **Smallworld Application Manager** will open
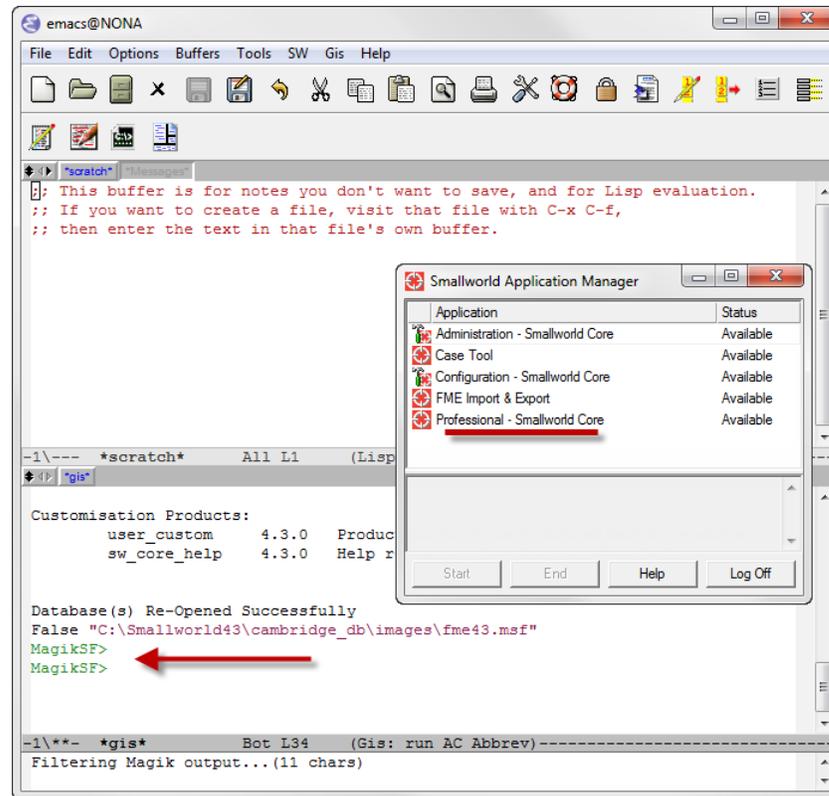
2. Start the Smallworld Connection Service. This service allows the FME 'client' to connect to the Smallworld VMDS database 'server'. In emacs, at the MagikSF> prompt enter:

        MagikSF> **fme_connection_service.start(30000, _unset, :gis)**

   You should see the response:

        fme_connection_service running_on_port: 30000
        a sw:fme_connection_service
        MagikSF>

   Note: For Smallworld 4.3.0.7 you may see the response:

        FME Import & Export has started an fme tics server on port 30000
        a sw:fme_connection_service
        MagikSF>

3. In the Smallworld Application Manager, select the **Professional – Smallworld Core** application and click **Start.**
   Do **not** select the FME Import & Export application - this is the older FME interface

4. In the Smallworld GIS, select the Bookmark tool (or use View – Bookmarks) and select the **Electric Network** bookmark. This will zoom into the area of interest for the training.

5. Open the Version Manager and then select the correct dataset.
   a. In the Smallworld GIS, select **File – Versions…**
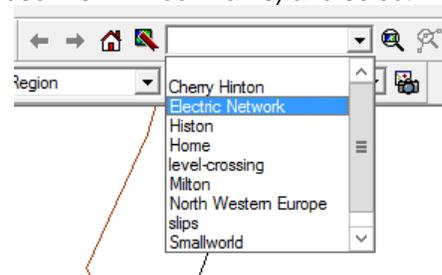
6. Select **Dataset: Electricity**
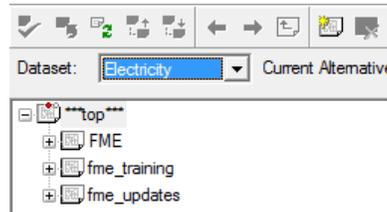
You'll come back to the alternatives and versions later in the training.

Import/Export Errors

If an import/export error occurs, check the FME log first. The traceback messages in Smallworld will generally be misleading. If you can't determine the problem cause in the FME log, next check the emacs traceback to see whether the error occurred in the Smallworld FME translator. The document "Smallworld Translators Suite, Smallworld FME® Reference" from GE does describe some of the typical Smallworld errors which may occur.

If an error occurs during an import to Smallworld it is probably wise to rollback to a checkpoint to ensure that invalid data was not loaded.

SpatialBiz Plug-in Interface Between Smallworld and FME

The SpatialBiz Plug-in for FME from Spatial Business Systems Inc provides another interface between FME and Smallworld, separate from the Smallworld FME Translator. The SpatialBiz Plug-in may offer advantages over the Smallworld FME Translator in certain circumstances

More information about the SpatialBiz Plug-in for FME is available on FME Knowledge Center.

This course does not cover the use of the SpatialBiz Smallworld reader/writer plug-in for FME.

Getting Going

Setup the coordinate system used by the examples and test the connection service between FME and Smallworld

Coordinate Systems

Smallworld generally uses a worldwide coordinate system. The coordinate units are usually in inches, cm or mm, depending on the coordinate resolution required. When translating to other formats, this world coordinate system will most often need to be reprojected into a different coordinate system. FME may not support the coordinate system in the units of the Smallworld world coordinate system. For example, FME supports the British National Grid coordinate system in meters. But the Cambridge database use a British National Grid millimeter coordinate system. So we have created a coordinate system called BritishNatGrid_mm for the FME Smallworld training.

You may wish to create a custom coordinate system to support the units your Smallworld is using refer see the FME User Documentation: FME Coordinate Systems .

Setup: Install Coordinate System	
Objective	Install coordinate system used by Cambridge sample database

Install a Coordinate System

The Cambridge sample database uses a millimeter coordinate system, BritishNatGrid_mm that is not included in the FME installation.

Install this coordinate system in preparation for the examples to follow.

1) Install BritishNatGrid_mm
Locate this coordinate system file in the Smallworld training resources folder at
  *C:\FMEData2015\Resources\Smallworld\Coordinate Systems\
  BritishNatGrid_mm.fme*

Have a quick look at the BritishNatGrid_mm.fme file using NotePad or other text editor.

Copy the BritishNatGrid_mm.fme file, then paste it at
  <My Documents\FME\Coordinate Systems\BritishNatGrid_mm.fme
If it is not already present.

Restart FME Workbench and FME Data Inspector so they will load this new coordinate system.

FME does not know which coordinate system a Smallworld database is using.

You must determine which coordinate system Smallworld is using and be sure to specify that coordinate system explicitly for each FME reader and writer that is connected to Smallworld.

FME might not contain a world coordinate system that matches your Smallworld database.

In that case, you may have to create a custom FME coordinate system to match your Smallworld database. See the FME User Documentation: FME Coordinate Systems > Custom Coordinate Systems > About Custom Coordinate Systems

Connecting to the Smallworld Database

The Smallworld VMDS database must have a connection service configured so that FME can communicate with the database and read or write objects.

FME Connection Service

Before FME can connect to the Smallworld VMDS the FME Connection Service must be running on the Smallworld server. Start the FME Connection Service using the following *magik* command:

```
fme_connection_service.start(30000, _unset, :gis)
```

See the section Starting the Smallworld GIS above.

Test: FME Connection Service	
Objective	Test the connection between FME and Smallworld

Follow these steps to test the connection to the Smallworld database, and inspect the data in the FME Data Inspector.

It is useful to view Smallworld data using the FME Data Inspector because this allows you to:
- Inspect the data prior to translation
- Determine how FME "sees" or "represents" the Smallworld data
- Determine what data structures need to be created to successfully import data back into a Smallworld VMDS.

1) FME Data Inspector
Start FME Data Inspector—Go to Start > All Programs > FME Desktop 2015 > FME Data Inspector or use the FME Data Inspector short cut on your taskbar.

Select **File > Open Dataset.**

Enter the reader format parameters
Reader Format: Smallworld 4

Enter the **Coord. System:** Select BritishNatGrid_mm
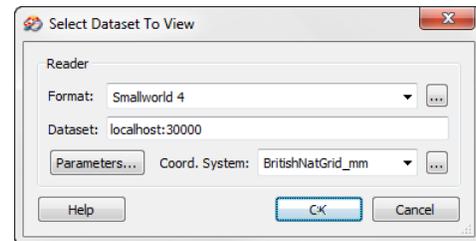
2) Enter Database Connection Parameters
Click on the Parameters button and enter the Smallworld database connection parameters:
Server Host: localhost
Server Port: 30000
Service: FME
Username: *(blank)*
Password: *(blank)*

3) Open Table List
Still in the Parameters dialog, click on the browse button next to Table List
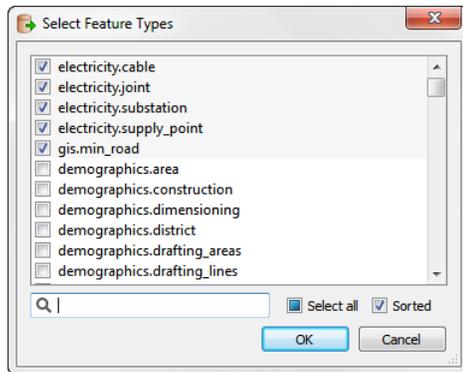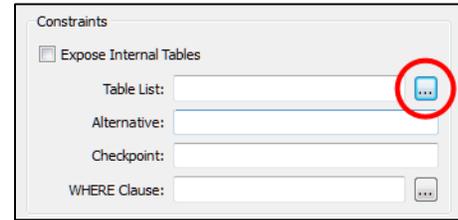
4) Select Tables to View
The Select Feature Types dialog appears.

In the Filter text box, enter "electric" or "gis" to filter the list of feature types.
Select each of these feature types:
- electricity.cable
- electricity.joint
- electricity.substation
- electricity.supply_point
- gis.min_road

Click **OK** to close the Select Feature Types dialog

5) Use Search Envelope
Check the Use Search Envelop and enter:
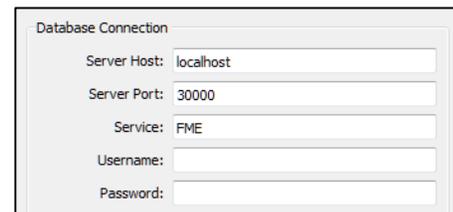- Minimum X: 545000000
- Minimum Y: 259000000
- Maximum X: 547000000
- Maximum Y: 261000000

These search envelope values are in the source data units of measure – millimeters in this case since the Smallworld world coordinate system is British National Grid mm.

You'll probably use these values again so select Defaults – Save as My Defaults.

Click **OK** to close the Parameters dialog. Click **OK** again, to open the dataset in FME Data Inspector
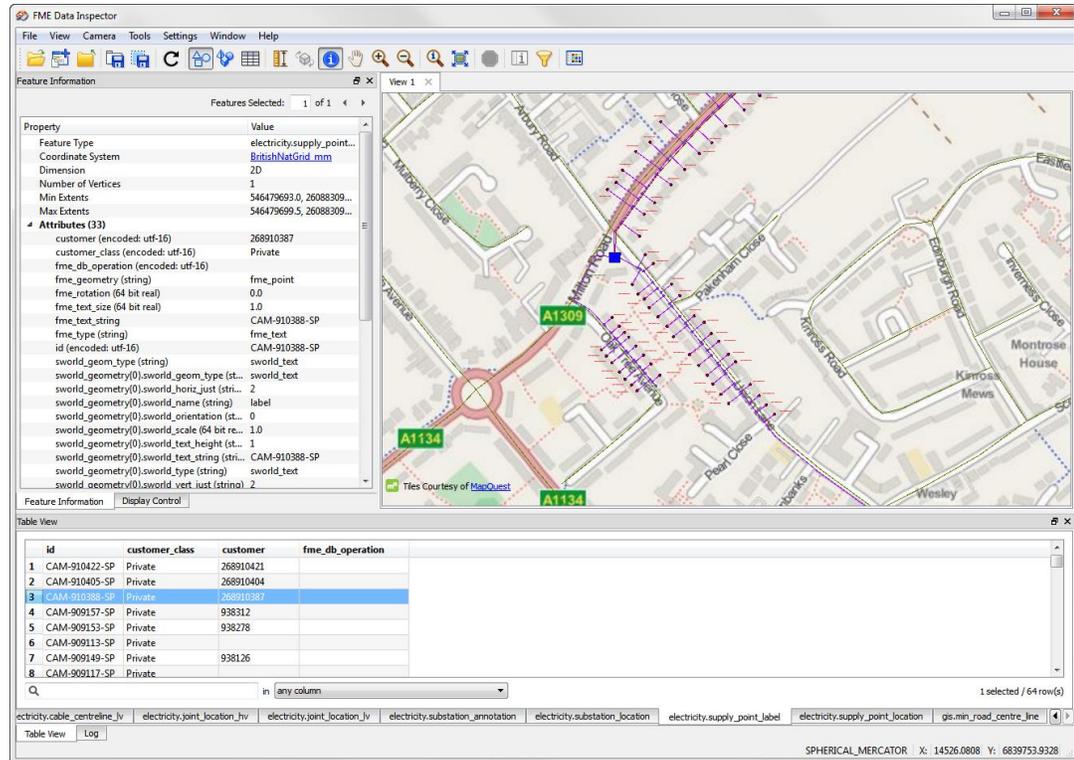
If you don't see the background map of the City of Cambridge then you probably need to configure Background Maps in FME Data Inspector.

6) FME Data Inspector Background Maps
Select Tools -> FME Options. Under Background Map Format select **MapQuest Web Map Tile Service** and under Parameters… select **Open Street Map.**

7) Data Inspection
Select the inspection tool 🛈 and click on the various features to see how FME represents the Smallworld objects with individual attributes and geometry in the Table View and the Feature Information window. In the Feature Information window, note the *sworld_geometry{}* list attribute. More about the sworld_geometry{} list attribute later.

Check the Display Control tab in FME Data Inspector to see the list of feature types.

Display Multiple Geometries

Smallworld objects often have multiple geometry columns, for example centreline & annotation. FME can preserve the multiple spatial column structure, or simplify the geometries.

When you configure the Smallworld reader, the Service parameter lets you select whether you will preserve multiple spatial columns or have FME simplify the geometries. The Service parameter options are:

FME	Simplify geometries
FMENOFACTORY	Preserve multiple spatial columns

Service: FME
Using Service: FME simplifies the Smallworld objects so that they are more compatible with most other GIS and CAD formats. Smallworld objects that contain multiple geometries are split into separate features with different FME feature types, one feature type for each geometry. For example, the Smallworld rail_line object with line and annotation geometries will become the FME feature types gis.rail_line_centre_line and gis.rail_line_annotation. Similarly a Supply Point object will be split into two features, electricity.supply_point_location and electricity.supply_point_label.

Service: FMENOFACTORY
The FMENOFACTORY service produces the "raw" Smallworld objects. The multiple geometries from the Smallworld objects are represented as aggregates on the FME features.

Displaying the "raw" Smallworld objects in FME Data Inspector is one way to determine how you need to structure the data in FME Workbench so that it will upload to Smallworld successfully. It's also useful if you're replicating your data in a database such as Oracle which has multiple spatial columns.

In the previous example you used the default parameter Service: FME. The next example illustrates Service: FMENOFACTORY.

1) Start FME Data Inspector
Back in FME Data Inspector, Select **File > Open Dataset.**

Enter the reader format parameters
Reader Format: Smallworld 4
Coord. System: BritishNatGrid_mm
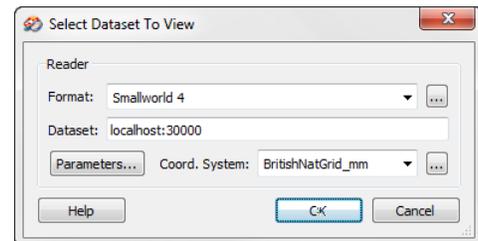
2) Enter Database Connection Parameters
 Click on the Parameters button and enter the Smallworld database connection parameters:
Server Host: localhost
Server Port: 30000
Service: **FMENOFACTORY**
Username: *(blank)*
Password: *(blank)*
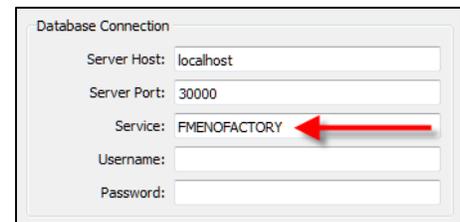
3) Open Table List
Still in the Parameters dialog, click on the browse button next to Table List

4) Select Tables to View
The Select Feature Types dialog appears. Notice that only the main table names are available from this service.

Choose the same objects as before:
- electricity.cable
- electricity.joint
- electricity.substation
- electricity.supply_point
- gis.min_road

Click **OK** to close the Select Feature Types dialog

Select the same Search Envelope values as before:
545000000 259000000 457000000 261000000

Click **OK** to close the Parameters dialog
Click **OK** again, to open the dataset.

6) Inspect the Dataset
Use the table view or the inspection tool to select an electricity.supply_point feature. You can see the feature attributes and the geometry. The geometry of the feature is an fme_aggregate with two parts, a **location** and a **label** (listed under IFMEAggregate):

Property	Value
Feature Type	electricity.supply_point
Coordinate System	BritishNatGrid_mm
Dimension	2D
Number of Vertices	2
Min Extents	545924686.0, 260235479.0
Max Extents	545927668.5, 260239858.0
◢ **Attributes (6)**	
customer (encoded: utf-16)	938074
customer_class (encoded: utf-16)	Private
fme_db_operation (encoded: utf-16)	
fme_geometry (string)	fme_aggregate
fme_type (string)	fme_collection
id (encoded: utf-16)	CAM-909081-SP
◢ **IFMEAggregate (2 Parts)**	
Front Appearance Reference	\<inherited_or_default_appearance>
Back Appearance Reference	\<inherited_or_default_appearance>
◢ ● **Part 0: IFMEPoint**	545924686.0, 260235479.0
Name (encoded: utf-8)	location
◢ T **Part 1: IFMEText**	CAM-909081-SP
Name (encoded: utf-8)	label
Encoding Name	fme-system
Text Rotation (degrees CCW)	0.0
Text Size	1.0
● **Text Location**	545927662.0, 260239857.0

The geometry parts of a feature will vary, depending on the object you select. For linear features such as the electricity.cable, the Smallworld chain and sectors are represented in FME as a path and segments (sometimes called a **complex edge** in network applications). The path is equivalent to the Smallworld **chain** and each sector is represented as a segment of the FME path. Select the cable:

    known_as   = Union Lane LV

This is a multi-sector chain in Smallworld.

If your Smallworld geometries also contain circular arcs, they will be preserved. Other types of arc (elliptical, spline, etc.) are stroked.

Smallworld Geometry

The Smallworld geometry representation in FME is made-up from three parts:
- Smallworld geometry field (geometry name and type)
- FME geometry aggregate
- sworld_geometry{} list attribute

The sworld_geometry{} list attribute describes the structure of the geometry aggregate and is used to link the FME geometry to the Smallworld geometry field when writing objects to Smallworld. In the Supply Point example above the FME Aggregate has two parts:

Part 0   FME Text   label
Part 1   FME Point   location

The sworld_geometry{} list attributes describes the associated geometry name and any geometry characteristics, such as text orientation.

Again, in the Supply Point example above:

`sworld_geometry{1}.sworld_name (string)` **location**

indicates that the second geometry in the aggregate is the location spatial column. Similarly,

`sworld_geometry{0}.sworld_name (string)` **label**

indicates that the first part of the FME aggregate is the label spatial column. The sworld_geometry{} list attributes describe specific characteristics of the text geometry.

Property	Value
Feature Type	electricity.supply_point
Coordinate System	BritishNatGrid_mm
Dimension	2D
Number of Vertices	2
Min Extents	545919896.0, 260087457.0
Max Extents	545922843.5, 260092849.0
⊿ **Attributes (17)**	
customer (encoded: utf-16)	
customer_class (encoded: utf-16)	Private
fme_db_operation (encoded: utf-16)	
fme_geometry (string)	fme_aggregate
fme_type (string)	fme_collection
id (encoded: utf-16)	CAM-909269-SP
sworld_geometry{0}.sworld_horiz_just (32 bit integer)	2
sworld_geometry{0}.sworld_name (string)	label
sworld_geometry{0}.sworld_orientation (64 bit real)	0.0
sworld_geometry{0}.sworld_text_height (64 bit real)	1.0
sworld_geometry{0}.sworld_text_string (string)	CAM-909269-SP
sworld_geometry{0}.sworld_type (string)	sworld_text
sworld_geometry{0}.sworld_vert_just (32 bit integer)	2
sworld_geometry{1}.sworld_name (string)	location
sworld_geometry{1}.sworld_orientation (64 bit real)	-138.13807611427166
sworld_geometry{1}.sworld_scale (64 bit real)	1.0
sworld_geometry{1}.sworld_type (string)	sworld_point
⊿ **IFMEAggregate (2 Parts)**	
Front Appearance Reference	<inherited_or_default_appearance>
Back Appearance Reference	<inherited_or_default_appearance>
⊿ **T Part 0: IFMEText**	CAM-909269-SP
Encoding Name	fme-system
Text Rotation (degrees CCW)	0.0
Text Size	1.0
● **Text Location**	545922837.0, 260092848.0
● **Part 1: IFMEPoint**	545919896.0, 260087457.0

(Text / Point labels on left side of table)

Export Examples

Export examples illustrate some of the typical issues associated with how Smallworld objects can be represented on other formats

Export to FME Feature Store (FFS)

FME Feature Store (FFS) is the internal FME format. It's useful to write Smallworld objects to FFS as this format maintains much of the complex structure of the Smallworld data, including multiple geometries and sectors. If you ever need to send Safe support an example of your Smallworld data, this is the best format to use.

Example 1: Write to FME Feature Store (FFS)	
Scenario	FME user
Data	Electricity facilities and minor roads
Overall Goal	Read Smallworld data and write to FME Feature Store format
Demonstrates	Smallworld multiple geometry objects represented as FME feature aggregates
Starting Workspace	Create from scratch
Finished Workspace	C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld1-Complete.fmw

This example also shows how to preserve the Smallworld multiple spatial columns in an FFS dataset.

1) Generate Workspace
Start FME Workbench. Select the option to **Generate workspace**.

Reader Format: Smallworld 4
Reader Dataset: localhost:30000
Reader Coord Sys: BritishNatGrid_mm
Reader Parameters
Service: <mark>FMENOFACTORY</mark>

2) Select Tables to View
The Select Feature Types dialog appears.

Select each of these feature types:

• electricity.cable	• electricity.supply_point
• electricity.joint	• gis.min_road

• electricity.substation	•

Click **OK** to close the Select Feature Types dialog

3) Use Search Envelope
Check the Use Search Envelope and enter:
- Minimum X: 545000000
- Minimum Y: 259000000
- Maximum X: 547000000
- Maximum Y: 261000000

Note that these values are in mm to match the BritishNatGrid_mm coordinate system.
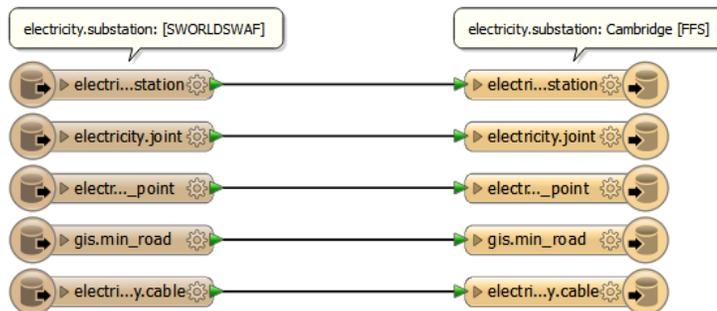
4) Set the Reader Defaults
At the bottom of the Smallworld 4 Parameters dialog, select the Defaults option and set "Save as My Defaults". This will save your parameter settings for future use.

5) Writer Parameters
Writer Format: FME Feature Store (FFS)
Writer Dataset: C:\FMEData2015\Output\Cambridge.ffs

Click OK to generate the workspace.

4) Run and Inspect Output
Run the translation.

Inspect the output in the FME Data Inspector.

Property	Value
Feature Type	electricity.cable
Coordinate System	BritishNatGrid_mm
Dimension	2D
Number of Vertices	34
Min Extents	545899295.0, 259848987.0
Max Extents	546247721.0, 260185703.0
◢ **Attributes (11)**	
fme_db_operation (encoded: utf-16)	
fme_feature_type (string)	electricity.cable
fme_geometry (string)	fme_aggregate
fme_type (string)	fme_line
id (encoded: utf-16)	914782
known_as (encoded: utf-16)	Union Lane LV
length (64 bit real)	488.6231554205155
multi_reader_keyword (string)	SWORLDSWAF_1
multi_writer_id (string)	0
status (encoded: utf-16)	In service
voltage (encoded: utf-16)	LV
◢ **IFMEAggregate (1 Part)**	
Front Appearance Reference	<inherited_or_default_app...
Back Appearance Reference	<inherited_or_default_app...
◢ **Part 0: IFMEPath (28 Segments)**	
Name (encoded: utf-8)	centreline
Closed	No
▷ **Segment 0: IFMELine (2 Coordinates)**	(545899295.0, 260185703.0 ...
▷ **Segment 1: IFMELine (2 Coordinates)**	(545919415.0, 260175262.0 ...
▷ **Segment 2: IFMELine (2 Coordinates)**	(545922638.0, 260173590.0 ...
▷ **Segment 3: IFMELine (2 Coordinates)**	(545926157.0, 260171764.0 ...
▷ **Segment 4: IFMELine (2 Coordinates)**	(545936617.0, 260159458.0 ...

Use the select tool to select the Union Lane LV cable.

Notice the path geometry and the path segments that represent the original chain and chain sectors from Smallworld.

Select one of the Supply Point features Notice how the multiple geometries of the Smallworld object are represented by an FME aggregate.

Writing to FFS is essentially a "dump" of Smallworld objects. If the translation to FFS fails, then there are probably errors in the Smallworld data.

Using FFS files is a good way to send "raw" Smallworld data to the Safe Software support team.

Export to Esri File Geodatabase

The Esri Geodatabase format supports only one geometry per feature class. When the Smallworld FME service is used with the Esri writer, FME Workbench automatically will generate separate reader and writer feature types for each geometry on the Smallworld object.

Example 2: Write to Esri Geodatabase	
Scenario	FME user
Data	Electricity facilities and minor roads
Overall Goal	Read Smallworld data and write to EsriGeodatabase
Demonstrates	Smallworld multiple geometry objects represented with distinct FME feature types
Starting Workspace	Create from scratch
Finished Workspace	C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld2-Complete.fmw

1) Generate Workspace
Start FME Workbench. Select the option to **Generate workspace**.

Reader Format:	Smallworld 4
Reader Dataset:	localhost:30000
Reader Coord Sys:	BritishNatGrid_mm
Reader Parameters	
Service:	**FME**
Table List:	electricity.cable
	electricity.joint
	electricity.substation
	electricity.supply_point
	gis.min_road

*You only need to select the object names; FME will automatically include their multiple geometries.*

Use Search Envelope

Minimum X:	545000000
Minimum Y:	259000000
Maximum X:	547000000
Maximum Y:	261000000

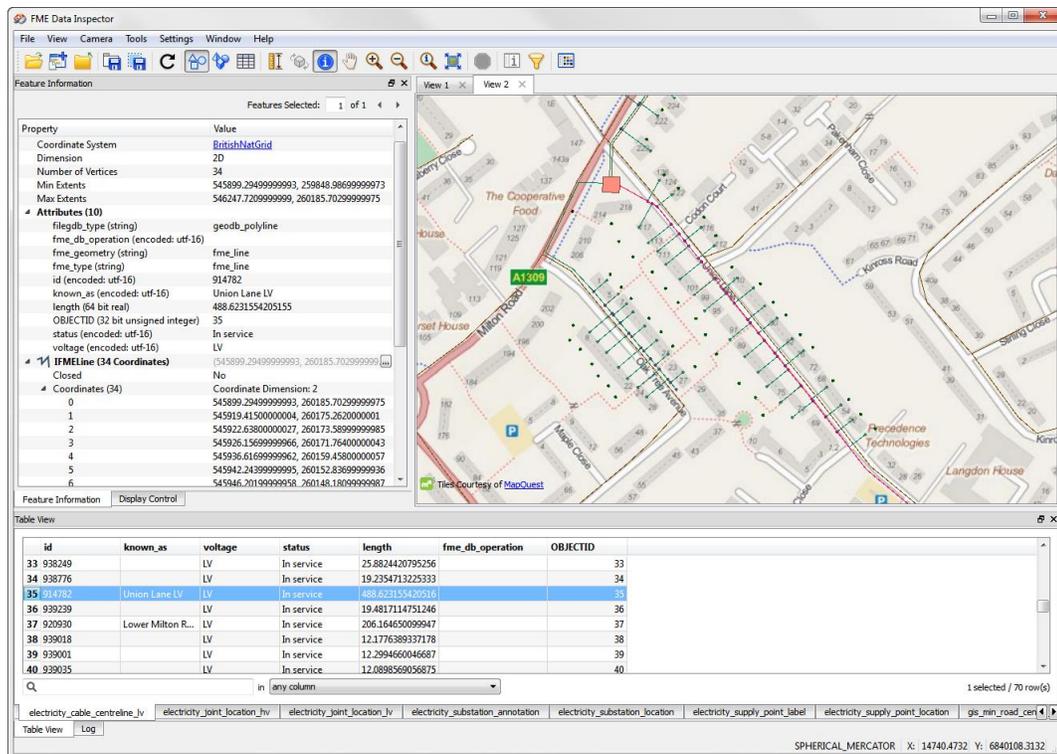|	
Writer Format:	Esri Geodatabase (File Geodatabase API)
Writer Dataset:	C:\FMEData2015\Output\Cambridge.gdb
Writer Coord Sys:	BritishNatGrid

Be sure to specify the Smallworld reader <u>Service: FME</u> and the different reader and writer coordinate systems as shown.

Click **OK** to generate the workspace.
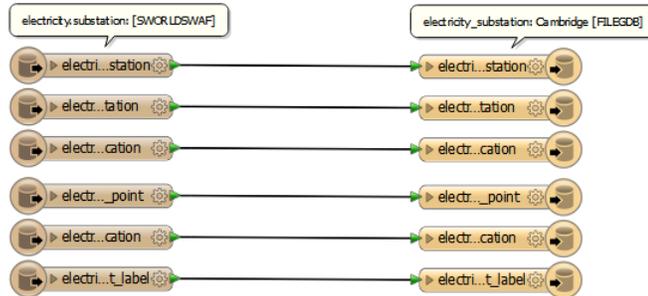
2) Run and Inspect Output
Run the translation. Inspect the output in the FME Data Inspector.

Notice how separate feature types are created for each geometry on the single Smallworld object. Use the inspection tool to select the Supply Point location and then the Supply Point label point. Observe that these two features representing the same Smallworld object are related by *id*.

3) Modify Workspace (Advanced)
This is typically where you'd do some more sophisticated schema mapping to represent your data in a more structured way in the Esri Geodatabase. This might include:

- Merge Supply Point high and low voltage features
- Map HV & LV to subtypes or domains
- Create annotation feature classes
- Relationship feature class

See the workspace:
"C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld2-CompleteAdvanced.fmw"   for some ideas.

The Esri Geodatabase (File Geodatabase API) writer only has limited support for point, lines and areas. To gain the full benefits of the Esri Geodatabase, you can use the Esri Geodatabase (File Geodb ArcObjects) writer, provide you have ArcGIS installed and an ArcGIS license.

Export to Autodesk AutoCAD 3D Map

AutoCAD supports only one geometry per feature, but you can have features with different geometries in the same layer. When the Smallworld FME service is used with the AutoCAD writer, FME Workbench automatically will generate separate reader and writer feature types for each geometry on the Smallworld object.

Example 3: Write to Autodesk AutoCAD	
Scenario	FME user
Data	Electricity facilities and minor roads
Overall Goal	Read Smallworld data and write to AutoCAD
Demonstrates	Ensure that all Smallworld attributes are written to AutoCAD
Starting Workspace	Create from scratch
Finished Workspace	C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld3-Complete.fmw

1) Generate Workspace
Start FME Workbench. Select the option to **Generate workspace**.

Reader Format: Smallworld 4
Reader Dataset: localhost:30000
Reader Coord Sys: BritishNatGrid_mm
Reader Parameters
Service: **FME**   ⟵
Table List: electricity.cable
                    electricity.joint
                    electricity.substation
                    electricity.supply_point
                    gis.min_road

Use Search Envelope
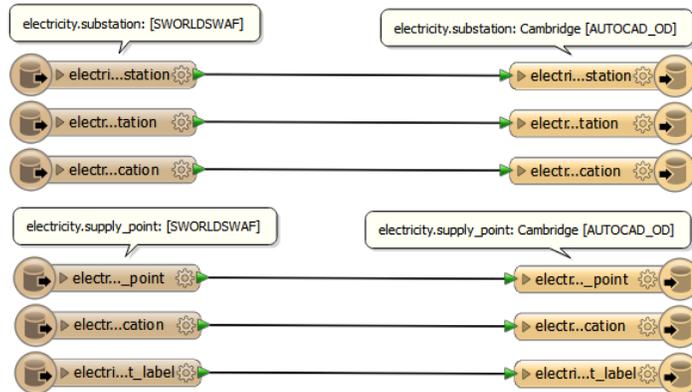   Minimum X: 545000000
              Minimum Y: 259000000
              Maximum X: 547000000
              Maximum Y: 261000000

Writer Format: Autodesk AutoCAD Map 3D DWG/DXF
Writer Dataset: C:\FMEData2015\Output\Cambridge.dwg
Writer Coord Sys: BritishNatGrid

Click OK to generate the workspace.

2) Modify Workspace

You need to truncate some of the long layer names;

electricity.cable_centreline_hv

electricity.cable_centreline_lv

Layer names in AutoCAD are limited to 30 characters so they will need to be renamed.

3) Run Translation

Run the translation.

Inspect the output in the FME Data Inspector.

This is typically where you'd do some more sophisticated schema mapping to represent your data in a more structured way in the AutoCAD file. This might include:

- Merge Supply Point high and low voltage features
- Merge Supply Point labels into the same layer as the Supply Point locations
- Use an AutoCAD template file to define the structure of your AutoCAD file
- Use the DWGStyler to add symbology such as blocks, line styles etc.
- Write a coordinate system .prj

See the Smallworld3-CompleteAdvanced.fmw for some ideas on expanding your workflow.

Data Export Summary

In summary, it's pretty straightforward to extract data from GE Smallworld. Gotchas are:

- Make sure the FME connection service is running in Smallworld
- Coordinate system – using a world coordinate system . And make sure you have set the coordinate system.
- Multiple vs. Single geometry export using Service: FMENOFACTORY or FME respectively
- Schema mapping of the results to the output schema

Import Examples

Import examples illustrate some of the issues associated with importing data into Smallworld, including handling multiple geometries and database operations

Considerations for Import

FME can't create new objects in the Smallworld database. You can only load into existing objects. This means that on import, the FME Smallworld writer data schema must accurately match the existing Smallworld object schema. It's useful to obtain a descriptive CASE report of the Smallworld objects that will receive the imported data.

Steps for Successful Import

1. Identify the correct Smallworld alternative or version to import into.
   FME will make the alternative writable

2. Use FME Data Inspector with the FMENOFACTORY service to inspect existing Smallworld records in the target object. Take note of the data schema that will need to be created in the FME workspace

3. Create the FME workspace in FME Workbench to translate the source dataset schema into the destination schema on the Smallworld objects

4. Test the workspace using the 'Redirect to Inspection Application' menu option, to ensure that the data is restructured as planned

5. Import the data into a test alternative and check the results

6. Import the data into the production alternative or "top" alternative – consider setting a checkpoint before you run FME so you can rollback if needed

7. Inspect the updated Smallworld objects to verify that the imported data is loaded correctly.

Before starting any data import, you should inspect both the source dataset in the FME Data Inspector and the Smallworld objects to obtain a full understanding of the data schemas.

Import Simple Geometries

The Electricity Cable objects in the Smallworld database have a single Centreline geometry. You're going to import some Proposed cables for a new sub-division into the Cambridge database from an AutoCAD 3D Map dataset.

Example 4: Import Single Geometry Objects to Smallworld	
Scenario	FME user
Data	Smallworld Electricity.Cable; Cambridge database inserts
Overall Goal	Insert new Smallworld electricity objects from feature update data
Demonstrates	How to import single geometry objects into Smallworld
Starting Workspace	Create from scratch
Finished Workspace	C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld4b-Complete.fmw

First, inspect the destination Smallworld objects and the source AutoCAD dataset.

1) Inspect Source Dataset
Open the FME Data Inspector. Click on **File > Open Dataset**:

Using FME Data Inspector, open the source dataset:
       "C:\FMEData2015\Resources\Smallworld\sourceData\electric_networkProposed.dwg"
The AutoCAD file cable has a status attribute "In service" and "Proposed". You'll only import the proposed cables as the "In service" cables already exist in the Cambridge database.

In this case, because there is a .prj file in the same directory as the .dwg file, the coordinate system for the AutoCAD drawing can be determined – BritistNatGrid (meters)

2) Compare Electricity Cable in Smallworld and FME Data Inspector
Using the Smallworld Application Manager, open the Professional - Smallworld Core application and inspect the Cambridge sample database.

- Use the bookmark: **Electric Network** to zoom in on the Cambridge electric network data
- Choose the Select Mode tool and click on the "Union Lane LV" cable object
- Check the object Editor for the cable attributes and geometry
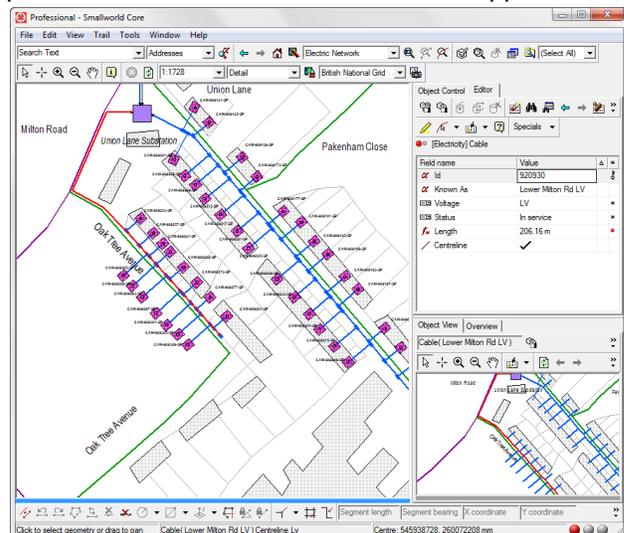
In the FME Data Inspector. Click on **File > Open Dataset**:

Reader Format:	Smallworld 4
Reader Dataset:	localhost:30000
Reader Coord Sys:	BritishNatGrid_mm
Reader Parameters	
Service:	FMENOFACTORY
Table List:	electricity.cable

- Zoom in to the same area as above
- Choose the Select Features tool and click on the "Union Lane LV" cable feature
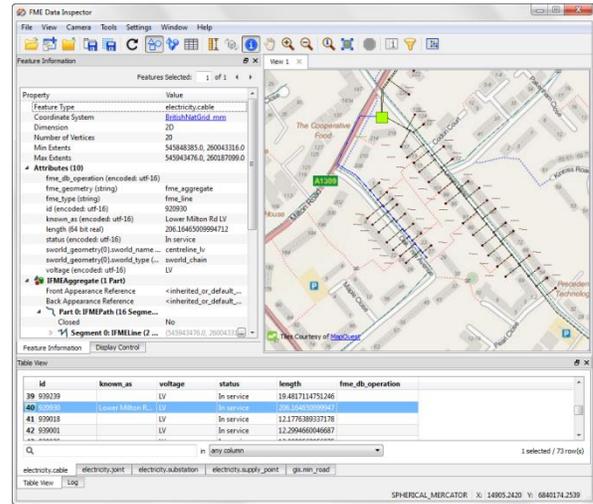
Compare the Smallworld and FME representations of the electricity.cable. In FME the cable geometries are represented by a single geometry (centerline) - in the case of "Union Lane LV" a multi-segment path geometry.

3) Inspect Destination Schema
In the Smallworld GIS, use the Editor to inspect the electricity.cable objects schema. Notice that the sys_id field is named '**id**' is the key field. If you look at a CASE report of the electricity.cable it is a mandatory field.

The Status attribute is an enumerated list with possible values:
```
In service
Abandoned
Accepted
Out of service
Proposed
Removed
```
The voltage attribute is an enumerated list with possible values:
```
11kV
132kV
LV
```
When loading the Status or Voltages attribute, you need to ensure that one of these mandatory enumerated values is set.

Take note how the source schema relates to the destination schema:

Source AutoCAD	Destination Smallworld	Action
id	id	Match FME feature with Smallworld object
length	--	Not updated because calculated field
known_as	known_as	
status	status	Make sure value matches enumerated values
voltage	Voltage	
Geometry		
fme_line	Centreline	

In this example there is not much schema mapping is involved. In most cases you'd need to add some FME schema mapping transformers such as AttributeRenamer, AttributeValueMapper, AttributeCreator or SchemaMapper.

4) Select the Smallworld Destination Alternative
In the Professional - Smallworld Core application, click on **File > Versions** to open the Version Management application.

In the Version Management application, add a new 'alternative', if it doesn't already exist:

Dataset	Name of Alternative
Electricity	fme_training

Double click on the alternative to make it the current alternative.

5) Generate Workspace
Open FME Workbench. Choose the Generate Workspace option.

Reader Format: Autodesk AutoCAD Map 3D Object Data
Reader Dataset:
    "C:\FMEData2015\Resources\Smallworld\sourceData\electric_networkProposed.dwg"
Reader Coord Sys: BritishNatGrid (actually this is optional since the esri_cad.prj exists)
Reader Parameters
Expand Blocks into Entities <mark>uncheck</mark>
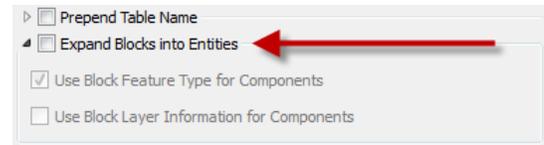
When you uncheck the Expand Blocks into Entities, FME will only read the insertion point for that block, and will not read the block definition.
This is usually what you want when loading CAD data into a GIS.

Writer Format: Smallworld 4
Writer Coord Sys: BritishNatGrid_mm
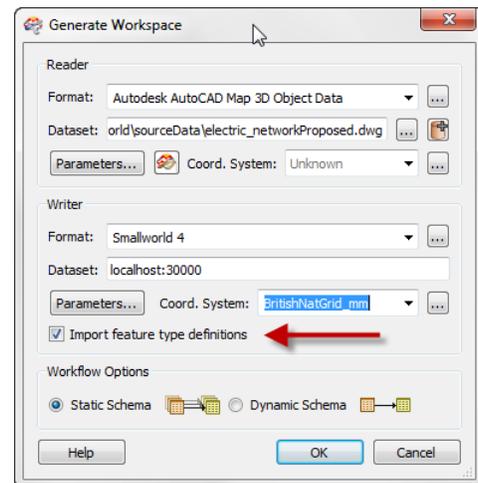**Writer Parameters**
Service: FME
Alternative: |fme_training
(Be sure to use the | symbol as the first character)

In the Generate Workspace dialog, place a checkmark on the
"**Import feature type definitions**" checkbox.

Click OK.

6) Import Writer Feature Types from Smallworld

The "Import Writer Feature Types" dialog will appear. Enter the usual Smallworld connection parameters.
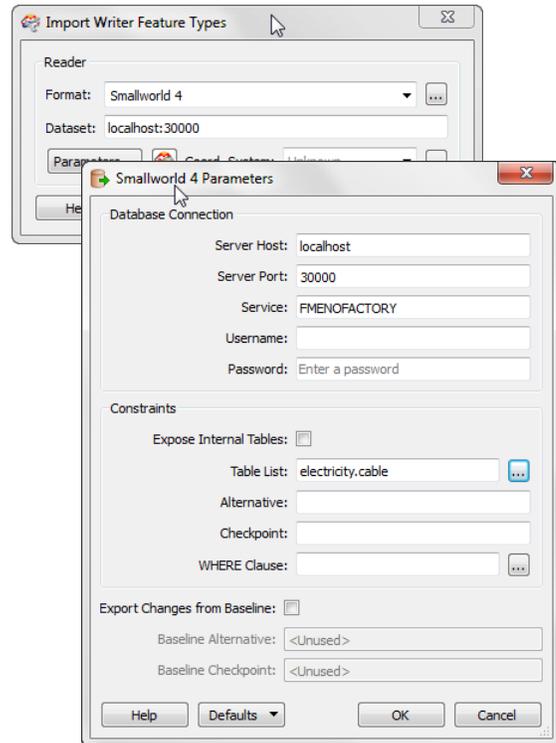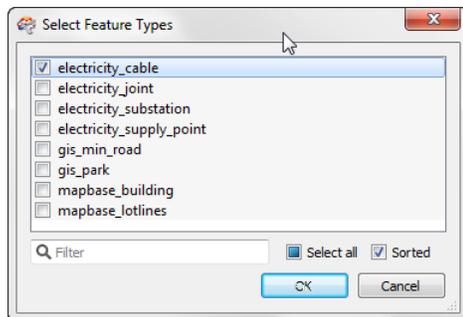
Click on the **Parameters** button, and select:
>   **Table List:**   electricity.cable

Click **OK** to close the import feature types dialog.

7) Select AutoCAD Layers

The **Select Feature Types** dialog will open. This is the AutoCAD layer selection. Select **electricity_cable** and **OK**, to generate the workspace.

> *It is advisable to use the **Import Writer Feature Types** option whenever a Smallworld writer is added.*
>
> *The Smallworld schema will generally be different to the reader schema, and the correct schema can only be obtained by explicitly importing the Smallworld feature type schema.*

---

8) Save Workspace

Save the workspace locally at
>   *<My Documents>\My FME Workspaces\Smallworld4.fmw*

9) Connect Reader and Writer Feature Types

Connect the electricity_cable reader feature type to the new electricity.cable writer feature type. Expand the attribute lists for the reader and the writer.
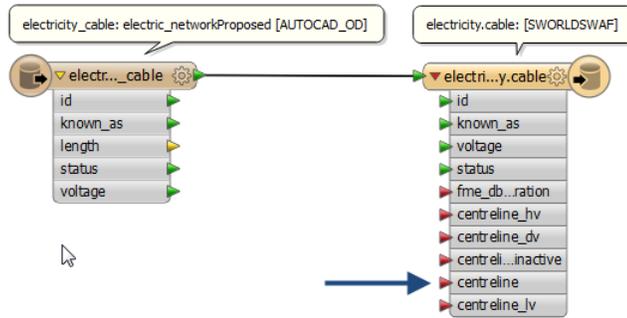
The Smallworld writer exposes all the 'mapped' geometries on the electricity.cable object (*centerline_lv, centerline_hv*, etc.) as well as the physical geometry – *centerline.* FME will only import into the physical geometry, *centerline.*

10) View Geometry Attribute

Click on the parameters button for the electricity.cable writer feature type.

Click on the User Attributes tab. Notice that the data type for the *centerline* attribute is 'sworld_chain', FME's name for the Smallworld 'chain' geometry type.

The centerline attributes represent Smallworld geometry fields and so are never populated directly with attribute values. Instead their values are loaded automatically by FME from the feature geometry. The connection between the Smallworld geometry field (i.e. centerline) and the FME geometry is made using the swoeld_geometry{}.sworld_name attribute that you saw earlier in the FME Data Inspector and is described below.

This means that the input ports for the *centerline,* and other geometry field attributes, will always be unconnected and red.

Click **OK** on the feature type parameter dialog.
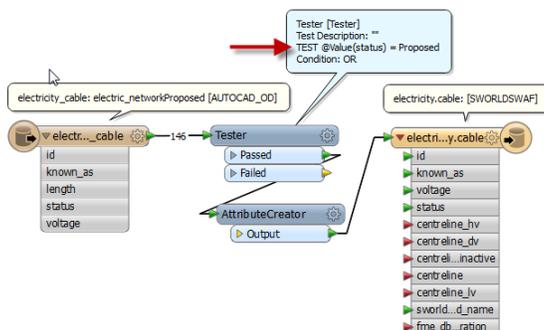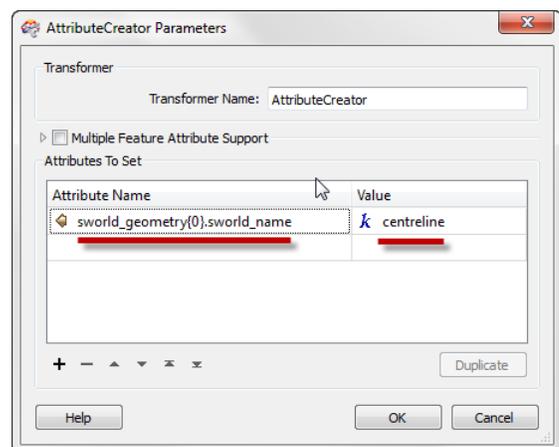
11) Filter the Proposed cables

You only need to insert the proposed cables. Add a Tester transformer to filter out the "In service" cables (or conversely keep the "Proposed" cables).

12) Set Geometry Name

Add an AttributeCreator transformer to your workspace and add the attribute

> sworld_geometry{0}.sworld_name
> > centerline

sworld_name connects the FME geometry to the centerline geometry field.

Notice that the geometry name 'centerline' needs to be spelled in UK/Canada English, not 'cent**er**line'.

Your workspace should look something like this:

---

13) Save and Run
Save the workspace. Run the translation.

Inspect the updated data in the Professional - Smallworld Core application. The proposed cables have been added along Oak Tree Avenue. You might have to select **View – Refresh**.

In the Smallworld GIS, open the Smallworld Explorer (Tools ➔ Explorer…). Open the folders *Electricity*, *Electricity Network* and select *Cable.*
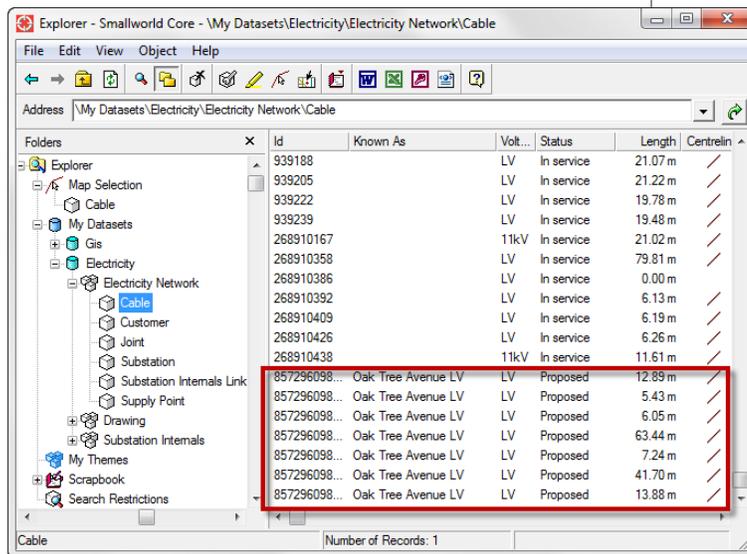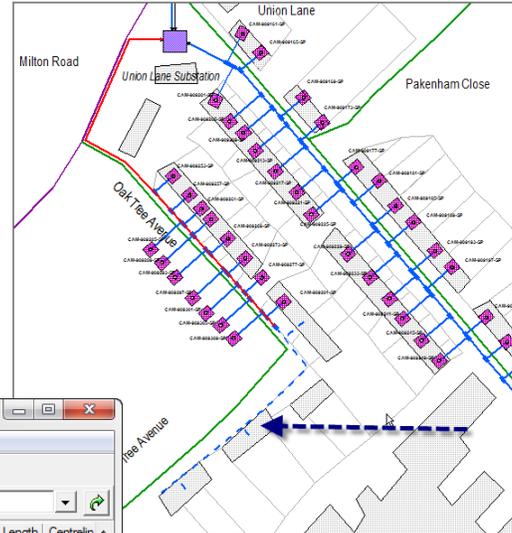
Problems?
The most common data load error is not loading the Smallworld geometry correctly.

In Smallworld Explorer, if the geometry column is not flagged or the Length is 0.0, then you have probably not set the

sworld_geometry{0}.sworld_name correctly which means the link between the FME geometry and the Smallworld geometry field will fail. Under the Writers menu, set the "Redirect to Inspection Application" and re-run the workspace. Inspect the results in Data Inspector to ensure the *sworld_geometry{0}.sworld_name* is configured correctly.

14) Clean-up
In the next section you're going to reload the data. In Smallworld Explorer version manager, select the fme_training alternative, rollback to the *begin* checkpoint by double clicking on the check point and selectin Rollback to Checkpoint. Or in Smallworld explorer select all the **Oak Tree Avenue LV** records you just inserted and **delete** them and commit the deletes. This will give you a fresh start.

Database Operation

At this point you let FME default to a database insert operation. If we wanted to update the existing cable features we'd need to use an update operation. This is what the **fme_db_operation** format attribute is for. FME uses the 'fme_db_operation' attribute on the Smallworld writer features to specify the database operation for each feature. The fme_db_operation attribute can be set to:

```
INSERT
UPDATE
DELETE
```

We're going to discuss the fme_db_operation attribute in more detail in a later section.

In the next steps, you're going to set the value of the known_as attribute to "connector" if it has a null value in the AutoCAD dataset.

1) Edit Workspace
Continue editing the workspace you've been working with.

2) Modify Tester
You do not need to update the named cables such as "Union Lane LV" or "Lower Milton Rd LV" so you can filter them out. Open the Tester properties and add the test:

```
known_as      Attribute Is Empty String
```

3) Add Conditional Tests to AttributeCreator
Now add a couple of conditional expressions to the AttributeCreator. The first conditional expression will set the **known_as** attribute:

if `known_as` is "`attribute is empty sting`" then set value to "`connector`"

and then to set the **fme_db_operation** attribute:

if `status` has value "`In service`" then `fme_db_operation` is `UPDATE`
else if `status` has value "`Proposed`" then `fme_db_operation` is `INSERT`
(Remember that FME is case sensitive for both attribute names and attribute values)
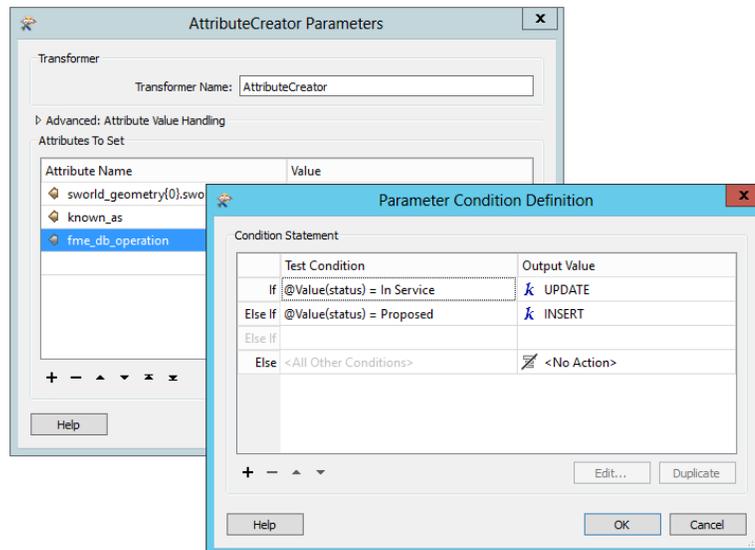
The workspace will now look something like:
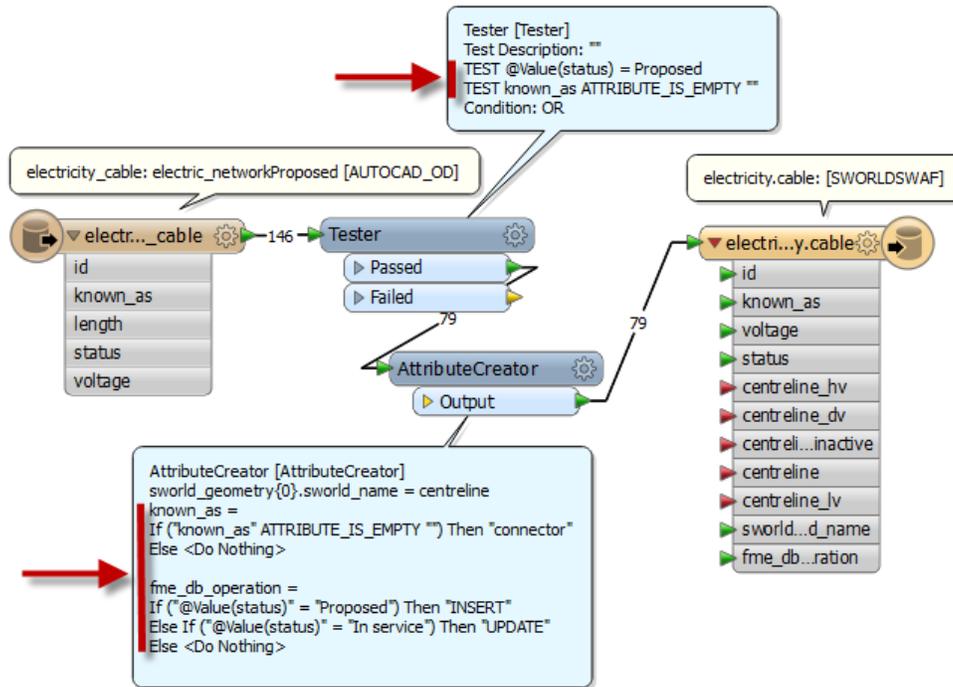
4) Test the workflow
When making more complicated workflows, it's a very good idea to try and test them before attempting to load the data into the database. In workbench, on the Writer menu, set **Redirect to Inspection Application**.

Then Run the workspace and check the feature counts on the workspace connectors and also check the results in FME Data Inspector. You should see all the cables with the exception of those that had a value for "known_as". All the "known_as" values should be either "connector" or "Oak Tree Avenue LV". "In service" cables should have an fme_db_operation = UPDATE. The "Proposed" cables should have an fme_db_operation = INSERT.

5) Save and Run
In workbench, on the Writer menu, unset Redirect to Inspection Application.

Save the workspace. **Run** the translation.

Inspect the updated data in the Professional - Smallworld Core GIS application.
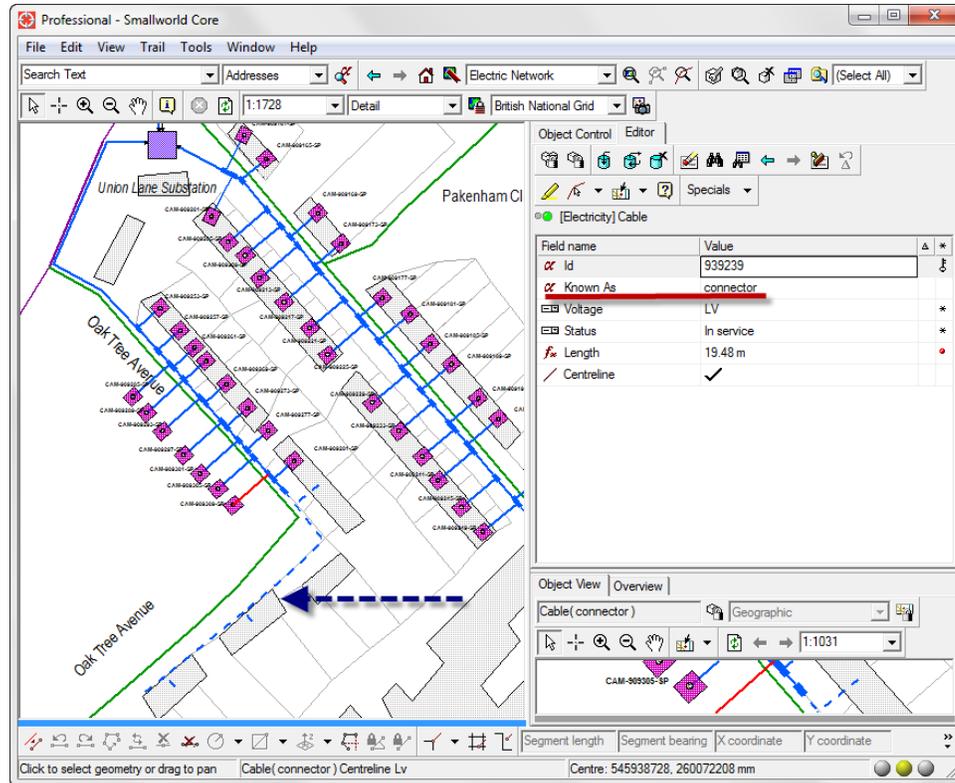
You have loaded some cables into the fme_training alternative and updated some of the existing cables.

Writing to Smallworld Multiple Geometries

The Smallworld data load in the previous example had only a single geometry on the cable object.
Many Smallworld objects have multiple geometries which might include one or more feature geometries (chain, area, point) and an annotation. Annotations often have the same text value as one of the feature attributes, e.g. "name".

Example 5: Write Multiple Geometries	
Scenario	FME user
Data	Smallworld supply point object; feature inserts
Overall Goal	Attach annotations to existing Smallworld Supply Point objects
Demonstrates	How to write text onto Smallworld objects as part of multiple geometries
Finished Workspace	C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld5-Complete.fmw

Load Supply Point Multiple Geometries
In this example you are going to load the Supply Point object which has both a point (location) and text (label) geometries.

1) Generate Workspace
You're going to generate the workspace in the same way you did in the previous section. Open FME Workbench. Choose the **Generate Workspace** option.
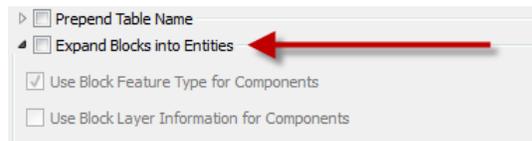
Reader Format: Autodesk AutoCAD Map 3D Object Data
Reader Dataset:
          C:\FMEData2015\Resources\Smallworld\sourceData\electric_networkProposed.dwg
Reader Parameters
Expand Blocks into Entities   uncheck

> ☐ Prepend Table Name
▲ ☐ Expand Blocks into Entities
  ☑ Use Block Feature Type for Components
  ☐ Use Block Layer Information for Components

Writer Format: Smallworld 4
Writer Coord Sys: BritishNatGrid_mm
Writer Parameters
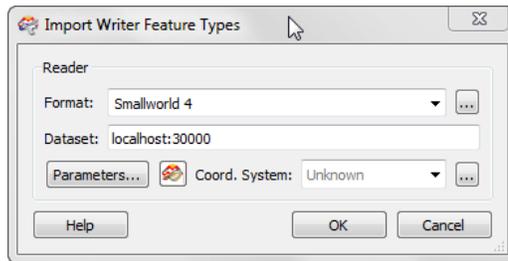**Service:**   FME
Alternative: |fme_training
(Be sure to use the | symbol as the first character)

In the Generate Workspace dialog, place a checkmark on the "**Import feature type definitions**" checkbox.
Click **OK**.

2) Import Writer Feature Types from Smallworld

The "Import Writer Feature Types" dialog will appear.
Click on the **Parameters** button, and select:
Table List: electricity.supply_point

Click **OK** to close the import feature types dialog.

3) Select AutoCAD Layers
The **Select Feature Types** dialog will open. This is the AutoCAD layer selection.
      Select **electricity_supply_poin**
And click **OK**, to generate the workspace.

4) Save Workspace
Save the workspace locally at
      <My Documents>\My FME Workspaces\Smallworld5.fmw

5) Connect Reader and Writer Feature Types
Connect the electricity_supply_point reader feature type to the new electricity.supply_point writer feature type.
Expand the attribute lists for the reader and the writer.

The Smallworld writer exposes the geometry fields on the electricity.supply_point feature type – *location* and *label.*

6) Redirect to Inspection Application
When making more complicated workflows, it's a good idea to try and test them before attempting to load the data into the database. In workbench, on the **Writer menu**, set **Redirect to Inspection Application**. All your output will be redirected to the Data Inspector as you build your workspace.

Also, as you build your workspace, add Data Inspector transformers to inspect the intermediate results.

7) Filter the Proposed Supply Points
You only need to insert the proposed supply points. Add a **Tester** transformer to filter out the "In service" cables (or conversely keep the "Proposed" cables).

8) Filter the Point & Text Geometries
Add a **GeometryFilter** and expose the Point and Text Geometries

9) Set Geometry Name and Database Operation
Add an **AttributeCreator** transformer to your workspace and connect it to the GeometryFilter Point port. In the AttributeCreator add the attributes:

fme_db_operation	INSERT	insert new records only
sworld_name	location	the Smallworld geometry field name
customer_class	New Customer	so we can recognize the new records

Add a second **AttributeCreator** transformer and connect it to the GeometryFilter Text port. In the AttributeCreator add the attributes:

fme_db_operation	INSERT	insert new records only
sworld_name	label	the Smallworld geometry field name
sworld_height	1	some text characteristics
sworld_horiz_just	3	
sworld_vert_just	1	
sworld_orientation	fme_rotation	fme_rotation is derived from the DWG
customer_class	New Customer	so you can recognize the new records

You've changed the customer_class attribute value so we can recognize the new records in Smallworld

10) Combine the Location and Label Geometries - Aggregator

The Aggregator transformer produces a new feature geometry structure that is an FME aggregate. An aggregate is a feature that has multiple geometries sharing the same attributes. In this case the aggregate consists of a text (for the label) and a point (for the location) of the supply point. The aggregator requires a *key* for the aggregation (Group By parameter). For many CAD to GIS problems, there may be no key attribute so a NeighborFinder transformer might be used to make a spatial join. We're lucky – this DWG file has an id field we can use.

Add an ***Aggregator*** transformer and connect both of the *AttributeCreators* to the input port.
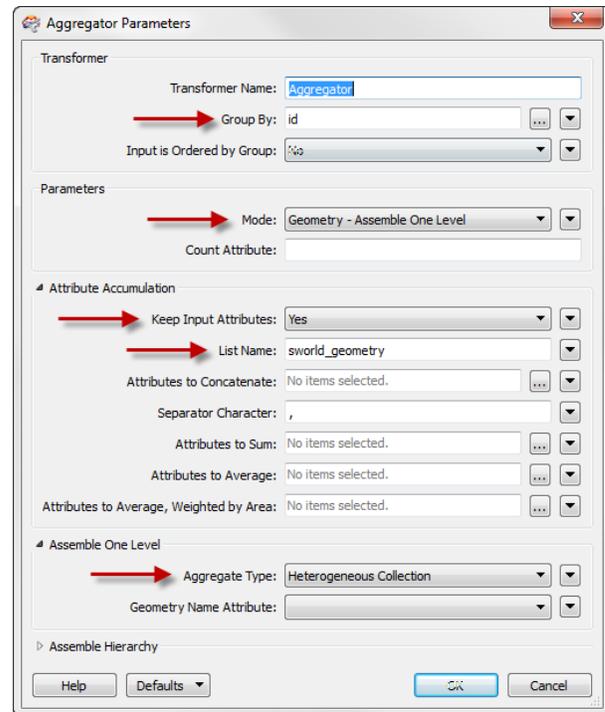
11) Set Aggregator Parameters

In the parameters dialog for the *Aggregator*, enter these settings:

Group By:	id
Mode:	Geometry - Assemble One Level
Attribute Accumulation	
Keep Input Attributes	Yes
List Name:	sworld_geometry
Aggregation Type	Heterogeneous Collection

The Aggregator transformer merges your point (location) and text (label) features using the *id* attribute as the key. The Mode sets the type of merge used for geometry – in this case a simple (one level) geometry structure.

List name: sworld_geometry builds a list for each aggregated feature so you'll end up with something like:

sworld_geometry{0}.sworld_name	location
sworld_geometry {1}.sworld_name	label
sworld_geometry {1}.sworld_height	1
and so on…	

Remember, the sworld_geometry{}.sworld_name links the appropriate part to the FME geometry to the Smallworld geometry field.

The **Aggregate Type** ensures the merged geometries maintain their geometry structure.

Add a Data Inspector to the Aggregator and inspect the results.

12) Clean-up Attributes (Optional)
You can add a couple of **BulkAttributeRemover** transformers – as shown below – to remove redundant AutoCAD attributes using regular expressions:

```
^autocad_
^fme_attrib_info
```

Your workspace should look something like this:
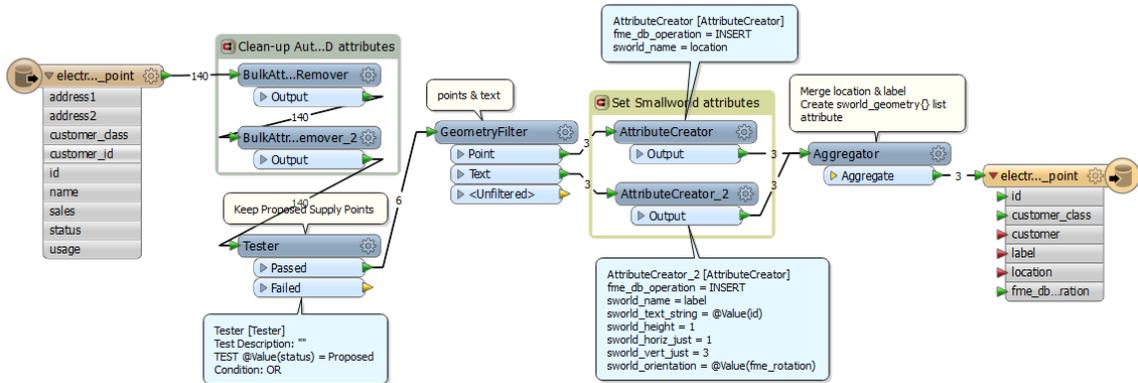
13) Save and Run
Run the workspace and inspect the results in Data Inspector.

On the Writer menu, unset **Redirect to Inspection Application** and run again.

Inspect the updated data in the Professional - Smallworld Core GIS application. If you need to, use the Smallworld Versioning to roll back to the *last* checkpoint and correct your workspace.

Save the workspace.

Summary

In this exercise, you've read point and text features from the AutoCAD dataset and then merged them with the Aggregator transformer. The Aggregator transformer also builds the required sworld_geometry{} list attribute required by the Smallworld writer.

Database Operations on Smallworld

You have used the 'fme_db_operation' format attribute in the previous sections. FME uses the 'fme_db_operation' attribute on the Smallworld writer to specify the database operation INSERT, UPDATE, or DELETE for each feature.

When the operation is UPDATE or DELETE, the reader feature must also supply the join key attribute that matches the Smallworld 'sys_id' on the writer feature type. FME will automatically identify the record to write using this join key attribute. The fme_db_operation can be used to control incremental updates to the database.

When the operation is INSERT, the join key attribute must not be included on the feature. For inserted records, Smallworld creates an internal 'sys_id' automatically, and the insert operation will fail if the writer attempts to supply a join key.

Example 6: Database Operations	
Scenario	FME user
Data	Smallworld electricity database; Cambridge_Electricty_Updates dataset
Overall Goal	Update the electricity database with object changes
Demonstrates	How to write insert, delete, and update Smallworld records
Starting Workspace	Create from scratch
Finished Workspace	"C:\FMEData2015\Workspaces\SmallworldTraining\ Smallworld6-Complete.fmw"

Update Electric Network

New electrical connections were proposed for new customers in the previous sections. These proposed changes have now be Accepted. Some new connections also require that existing connections be changed or removed. This example will update the Smallworld database with these new customers and with the changes to the electrical system.

In this example you'll use a GML file that has been created by a change management system. Changes are tracked by the update_type attribute as either D, U, I (delete, update or insert).

1) Inspect Source Dataset
Inspect the source dataset at
        C:\FMEData2015\Resources\Smallworld\sourceData\Cambridge_Electricity_Updates.gml
Some features have no geometry. All features have an *update_type* attribute that indicates the type of database operation to perform: I (Insert), U (Update) & D (Delete).

2) Add Smallworld Alternative
In the Smallworld Version Management application, add this new alternative or version if it doesn't already exist:

Dataset	Name of Alternative
Electricity	fme_updates

3) Generate Workspace
Start FME Workbench. Choose the option to Generate Workspace.

Reader Format:	GML (Geography Markup Language)
Reader Dataset:	*C:\FMEData2015\Resources\Smallworld\sourceData*
	Cambridge_Electricity_Updates.gml

Writer Format:	Smallworld 4
Writer Dataset:	localhost:30000
Writer Coord Sys:	BritishNatGrid_mm
Writer Parameters	
Service:	FME
Alternative:	|fme_updates

(Be sure to include the | symbol as the first character)

Place a checkmark on the **Import feature type definitions** checkbox.

Click **OK** to generate the workspace.

4) Specify Feature Types
When the *Import Writer Feature Types* dialog appears, click on the Parameters button.
Set these parameters:

Service:	FMENOFACTORY
Table List:	electricity.cable
	electricity.customer
	electricity.supply_point

When the GML **Select Feature Types** dialog appears, select these feature types:

```
electricity_cable
electricity_customer
electricity_supply_point.
```

5) Tidy Workspace
The feature types on the reader may be in a different order than on the writer. To make the feature type names easily visible, you can select the feature types and resize them, or add an annotation can be attached to each feature type.

To add an annotations, select all the feature types. Right-click and choose the **Attach Annotation** option.

Tidy the workspace so that each reader feature type is horizontally aligned with its matching writer feature type; but don't connect them yet.
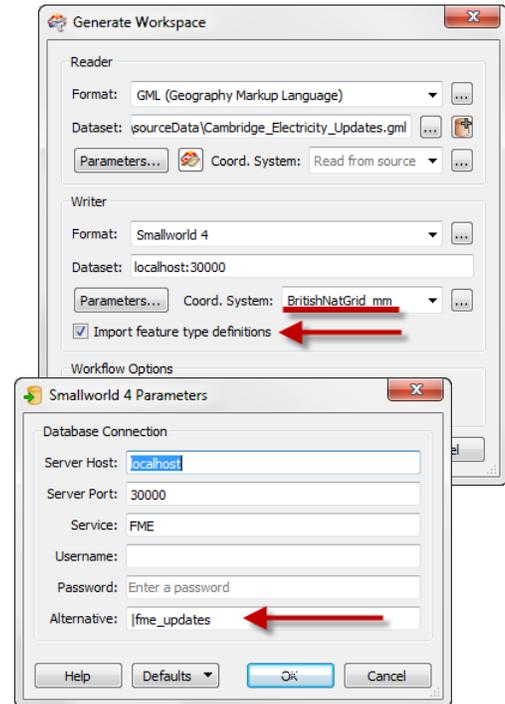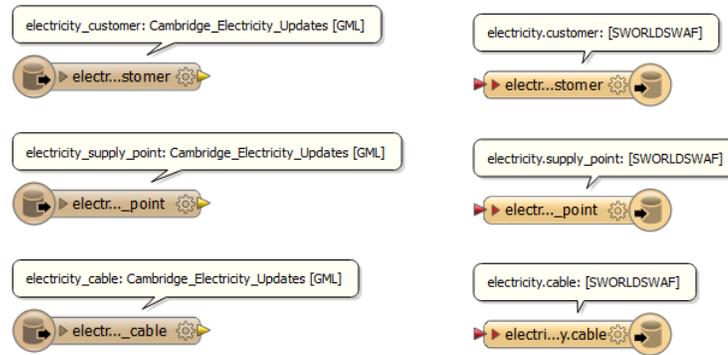
6) Set Database Operation

Add an *AttributeValueMapper* transformer connected to a reader feature type. The update_type attribute identifies which type of operation applies to each feature.

Source Attribute:	update_type
New Attribute Name:	fme_db_operation
Value Mappings	
I	INSERT
D	DELETE
U	UPDATE

Use **Duplicate** transformer and connect the duplicates to the remaining feature types as shown below:
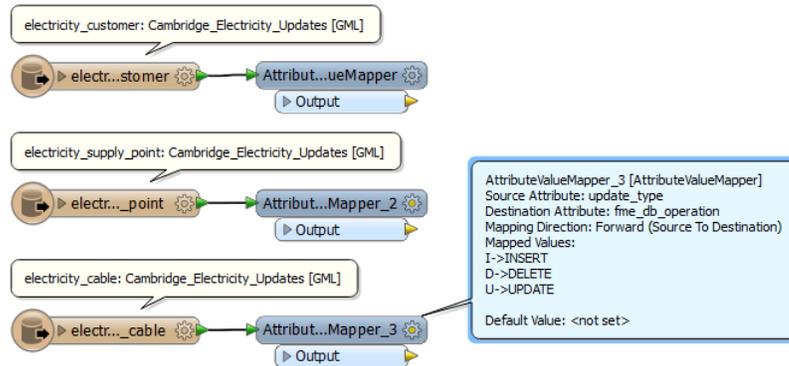
7) Set Geometry Name: centerline & location
Add an AttributeCreator transformer to your workspace and for *supply_point* add the attribute
   sworld_geometry{0}.sworld_name   *location*
Add an AttributeCreator transformer to your workspace and for *cable* add the attribute
   sworld_geometry{0}.sworld_name   *centreline*

Remember the role of the
sworld_geometry{}.sworld.name list attribute. It
associates the correct part of the FME geometry
aggregate to the correct Smallworld geometry
field.

Why don't you need an AttributeCreator for the
customer feature type?

8) Connect to the Writer Feature Type
Connect the *AttributeValueMapper* or
AttributeCreator output ports to its associated
writer feature type.
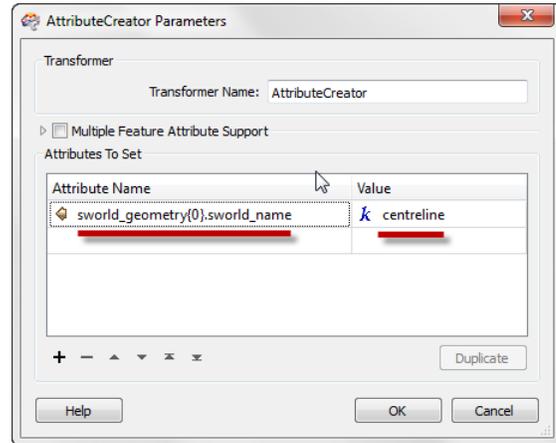
Your workspace will look something like the one below:
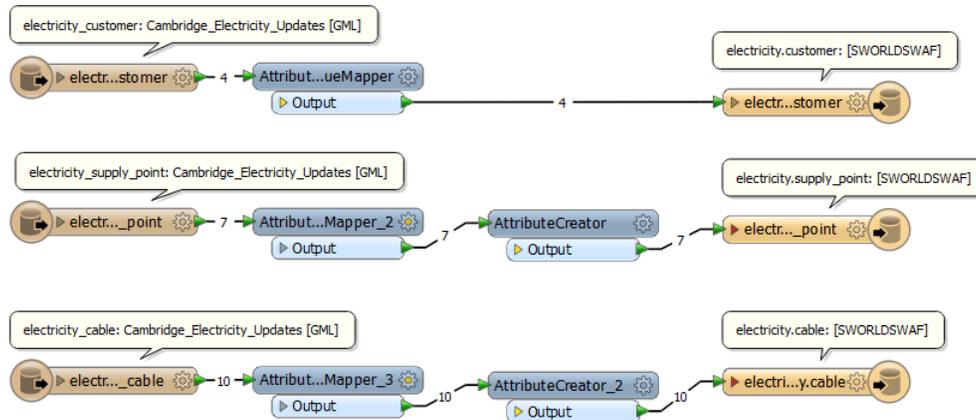
9) Inspect Output
Click on **Writers > Redirect to Inspection Application** to turn on this setting.

Save the workspace. Run the workspace.

Inspect the output in the FME Data Inspector. Use the inspection tool to select cables and supply
points and then verify their attributes. Note that for the INSERT's the *id* (or *customer_id* for
customers) is missing. This is because it's a new feature and does not yet have a 'sys_id" created
yet.
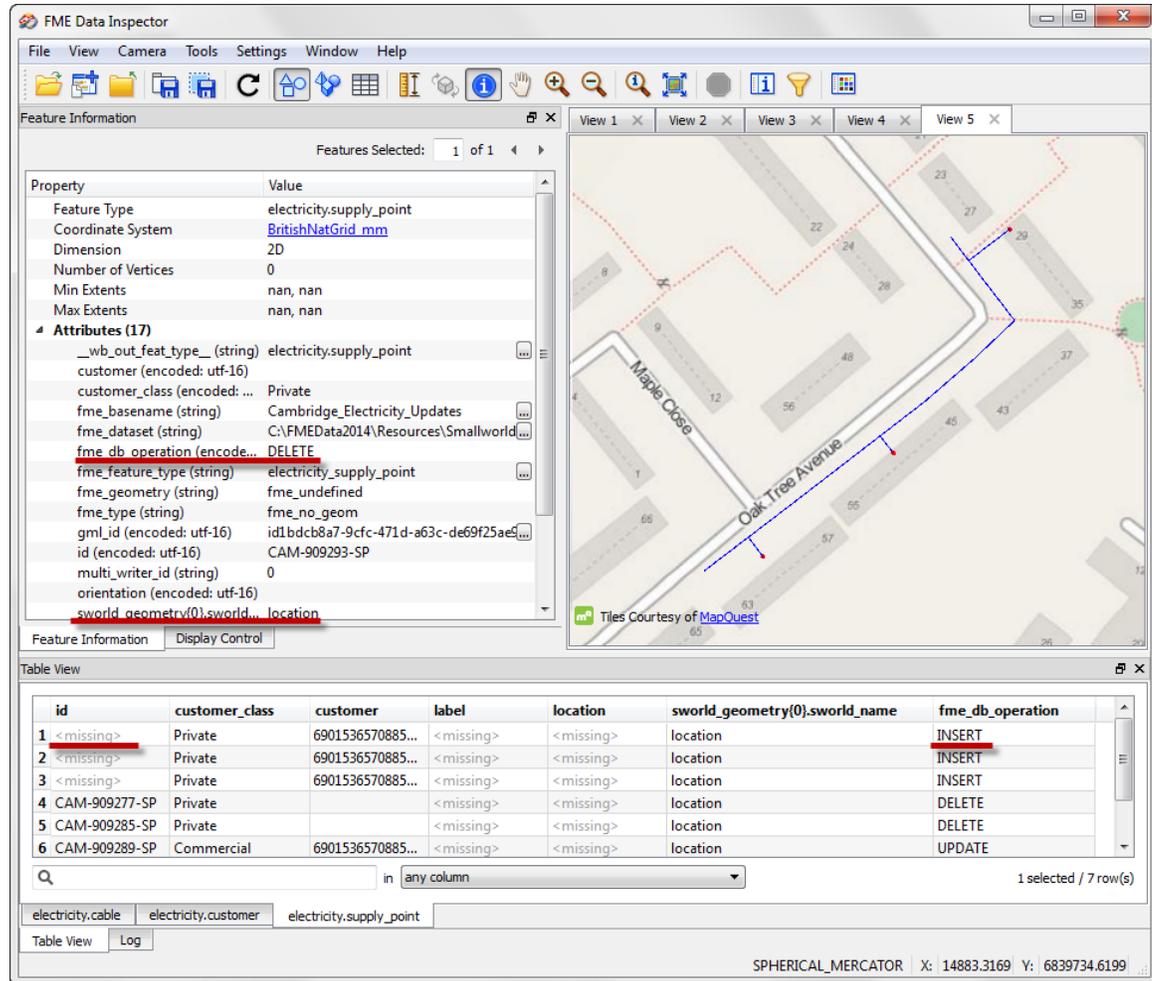
Here's what's going to happen:

| Feature Type	Operation	Feature Count
customer	INSERT	4
cable	DELETE	3
cable	INSERT	7
supply_point	DELETE	3
supply_point	INSERT	3
supply_point	UPDATE	1
	Total:	21

10) Write the Output

In FME Workbench, click on **Writers > Redirect to Inspection Application** to turn off this setting.
Run the workspace.

Open the Professional - Smallworld Core application.

Use the "Electric Network" bookmark to zoom in to see the area of interest around at Oak Tree Avenue.

Inspect the output in Smallworld. In the Smallworld Version Manager double click **fme_updates** to make it the current alternative. You may need to refresh the GIS view.
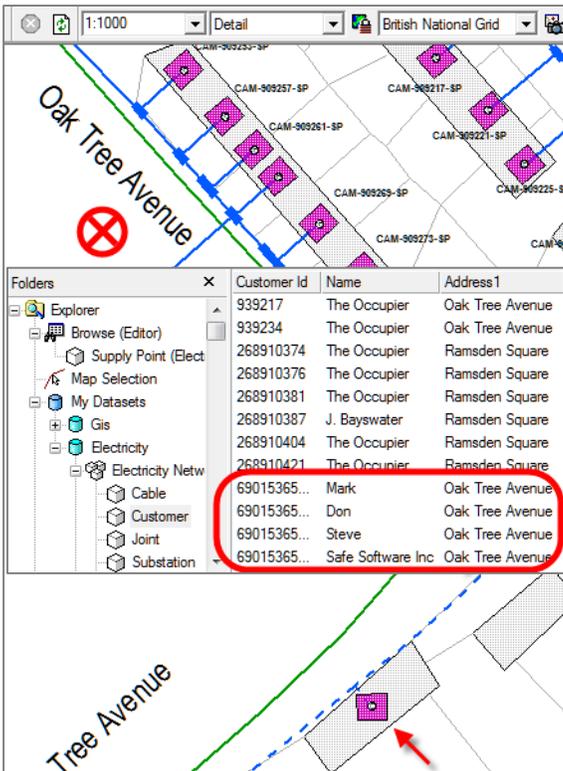
Locate the inserted cables and supply_points. Compare the fme_updates alternative to the base to detect the records that were deleted.

Use the object Explorer to locate the customer records that were inserted, since without geometry they won't appear in the view.

There is a relationship between the supply_point's and the customer table. This relationship uses an intermediate join table.

The GE Smallworld writer cannot create joins through an intermediate join table, so the supply_points have not been joined to their customers. You would have to accomplish this in the Smallworld GIS Object Editor.

FME can be used to import objects that have a direct join (parent/child) relationships. These can be a simple 1:0..n simple joins, 1:0..n heterogeneous joins or 1:0..n multiple joins.
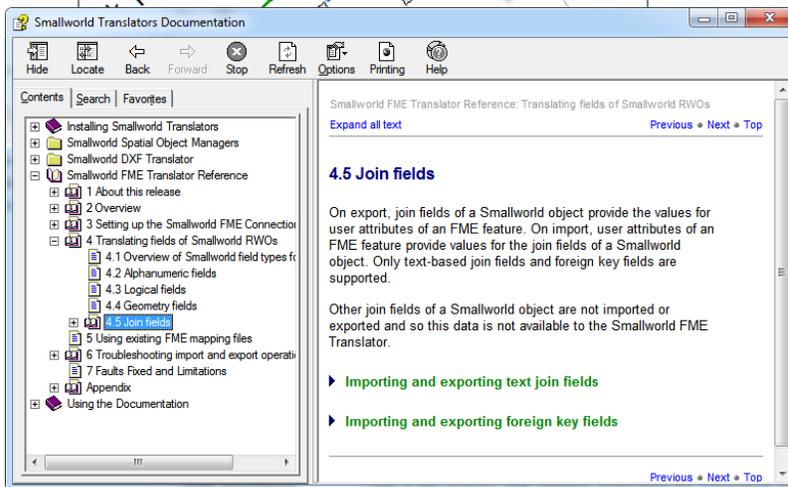
Handling joins is described in the Smallworld FME Translator Reference Section 4.5 Join Fields plus a little more on joins later.

Advanced Task
The supply points that were inserted have no labels.

In the Smallworld Version Management dialog double click on the *begin* checkpoint to rollback to the original data.

Back in FME Workbench, modify the workspace to add label geometry to the supply points. The changes to your workspace will be similar to the previous section where you added labels to the supply points.

Run the workspace and in Smallworld check that the re-inserted supply points now have labels.

Working with Alternatives or Versions

You've already been loading data into different alternatives or versions of the Smallworld Cambridge database. In the previous exercise you updated the electricity network. The GE Smallworld reader can read either all the objects from an alternative, or you can configure the reader to return only the deltas (changes) between alternatives (and checkpoints). This is very useful if you want to synchronize your Smallworld VMDS with another database, and only export the incremental changes.

Example 7: Exporting Changes Between Baseline and Alternative	
Scenario	FME user
Data	Smallworld electricity database updated with changes from Example 6
Overall Goal	Detect changes made to a Smallworld alternative
Demonstrates	Configuring FME to export just the differences between a base and alternative
Starting Workspace	Create from scratch
Finished Workspace	C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld7-Complete.fmw

> *This example requires that the updates from Example 6 have been loaded into Smallworld. If that didn't work, you can use the |FME alternative that was pre-configured with the changes*

Extract Deltas from an Alternative

After the electrical network updates have been made to a Smallworld alternative, you can examine just the differences (deltas) between the baseline and the alternative. This example will export those differences for inspection.

1) Add Smallworld Reader
Open FME Workbench. Start with a Blank workspace.

Add this reader:

Reader Format:	Smallworld 4
Reader Dataset:	localhost:30000
Reader Coord Sys:	BritishNatGrid_mm
Reader Parameters	
Service:	FMENOFACTORY
Table List:	electricity.cable
	electricity.customer
	electricity.supply_point

2) Connect Inspectors
Connect an *Inspector* transformer to each reader feature type.
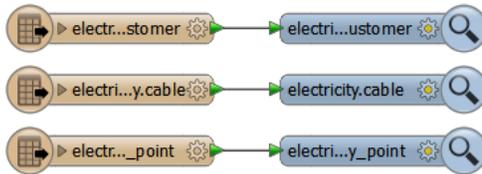
3) Run and Inspect
Run the workspace. You'll see all the objects from the 'top' alternative. You won't see any of the changes you made to the database in the previous exercise.

4) Select an Alternative
In the workspace Navigator, under the Smallworld reader, select Alternative and set the reader alternative:

Alternative: |fme_updates

5) Run and Inspect
Run the workspace. You'll see all the objects from the '*fme_updates'* alternative. You should see the changes you made to the database in the previous exercise.

6) Export Changes
Back in the workspace Navigator, under the Smallworld reader, select set the following reader parameters:

| **Export Changes from Baseline** | **Yes** |
| **Baseline Alternative:** | **\|** (pipe character represents the 'top') |

or use:

Export Changes from Baseline	**Yes**
Baseline Alternative:	**|fme_updates**
Baseline Checkpoint	**begin**

You can restrict the export of changes to a specific checkpoint on the baseline alternative.

7) Run and Inspect
Run the workspace. You'll see only the deltas between the 'top' alternative and the 'fme_updates' alternative.

Note: that the Smallworld reader automatically adds and sets the fme_db_operation attribute when you export changes so you could use these features to update other databases that FME supports.

8) Save Workspace
Save the workspace locally at
    <My Documents>\My FME Workspaces\Smallworld7.fmw
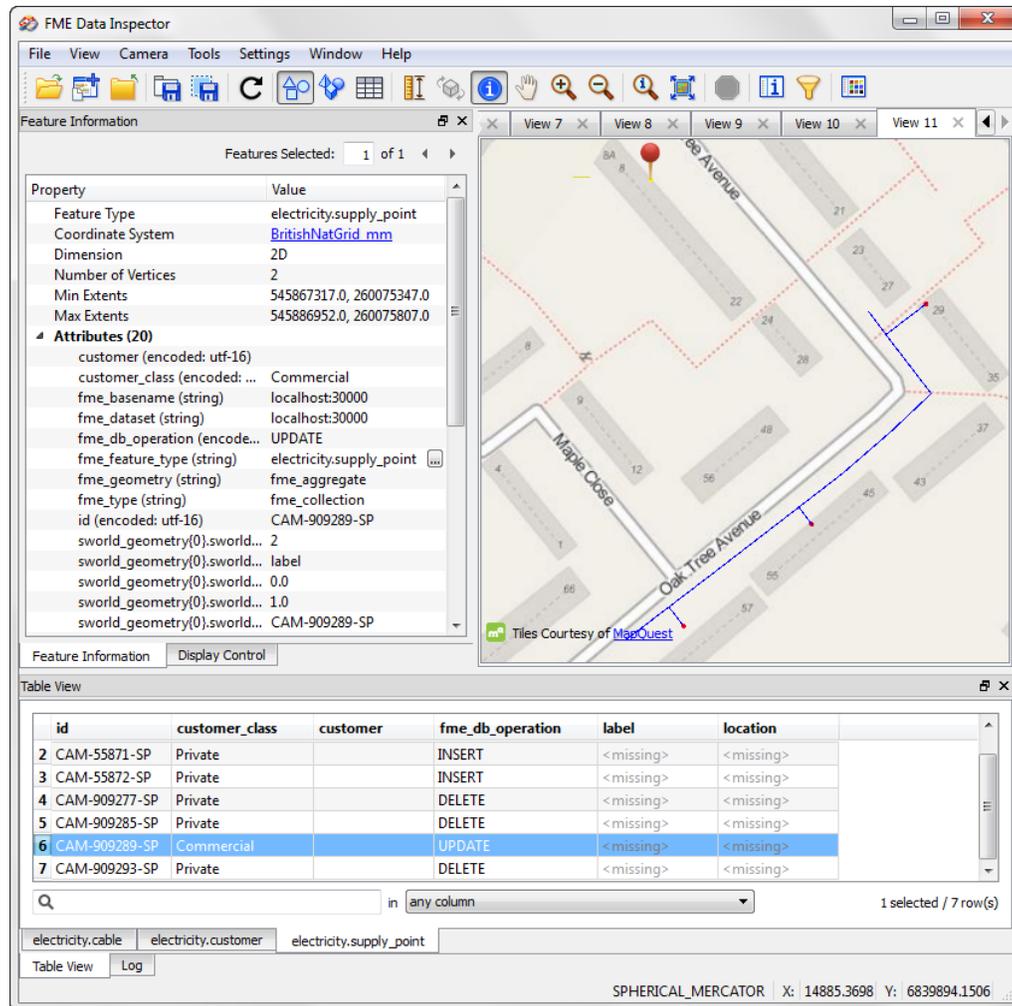
Exporting incremental changes in this way can be used to synchronize a Smallworld VMDS with an Oracle or an SQL Server geodatabase.

Advanced Task—WHERE Predicate

The Smallworld reader supports WHERE predicates that can be used to select a subset of data.

In the Workbench Navigator pane, locate the parameters for the Smallworld reader.
Set these parameters to select only cables whose status is "Proposed":

WHERE Clause: `[Electricity] cable where Status = "Accepted"`
Export Changes from Baseline: No

Run the workspace. Inspect the output. Only the 'Accepted' cables are exported.

Join Fields

FME can read different types of join fields. FME can also read intermediate join tables, for example electricity.int!supply_point_customer. So when exporting from Smallworld you can reconstruct more complex joins.

To export join fields, you may have to make the fields visible in the Administration ACE application. Some fields are not always made visible in the Smallworld GIS, and they must be visible for FME to be able to access the fields.

The GE Smallworld writer can import join fields for text based joins fields and foreign key fields. Other types of join fields are not supported, Refer to the GE Smallworld documentation: Smallworld FME Translator Reference – section 4.5 Join Fields. You may have to make some fields visible using the Administration ACE application

FME can load simple joins. There are three types of joins that can be represented:
- Simple parent – child 1:0..n relationship, i.e. Town - Roundabout
- Heterogeneous relationships. These are parent – child 1:0..n relationships, where there can be different parent objects. i.e. Office or Point of Interest may join to a Road Works object
- Multiple child object joins. These parent – child relationships have one parent but multiple child objects.

Example 8: Viewing Intermediate Join Tables	
Scenario	FME user
Data	Smallworld GIS Electricity database
Overall Goal	Export data with an intermediate join table
Demonstrates	Configuring FME to export more complex joins
Starting Workspace	Data Inspector only

Viewing Intermediate Join Tables
You'll use FME Data Inspector to view more complex join tables. Let's look at the following tables and their related join tables:

Electricity Network
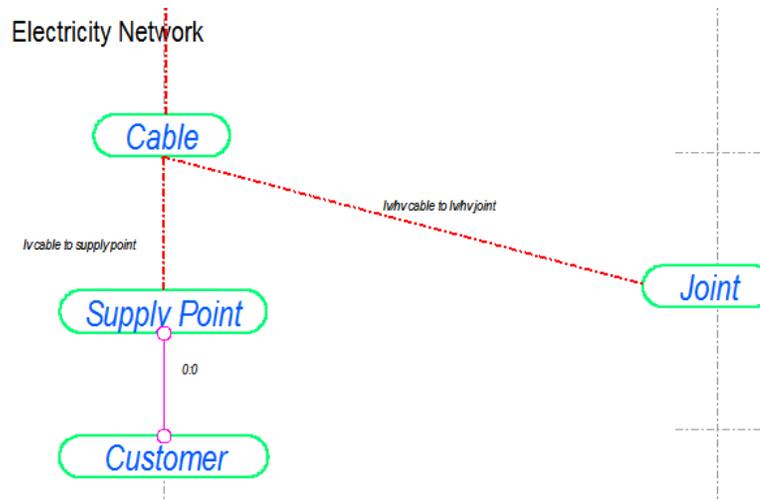
The image above comes from the Smallworld CASE for the Electricity Network. There is a join table for Customer and Supply Point.
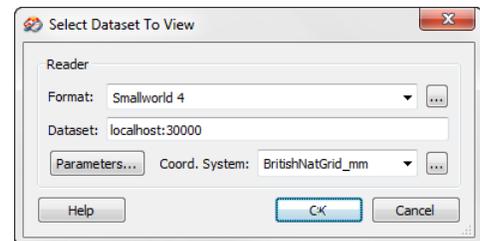
1) FME Data Inspector
Start FME Data

Select **File > Open Dataset.**

Enter the reader format parameters
Reader Format: Smallworld 4
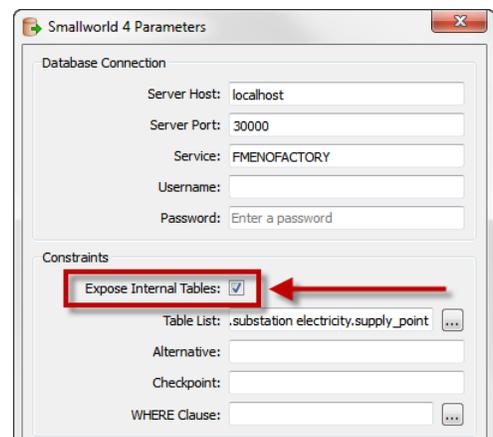
Enter the **Coord. System:** Select BritishNatGrid_mm

2) Enter Database Connection Parameters
Click on the Parameters button and enter the
Smallworld database connection parameters:
Server Host: localhost
Server Port: 30000
Service: FME
Username: *(blank)*
Password: *(blank)*

3) Select Constraints
Still in the Parameters dialog, check **Expose Internal Tables.**

Click on the browse button next to Table List.

In the Filter text box, enter "electric" or "gis" to filter the list of feature types.
Select each of these feature types:
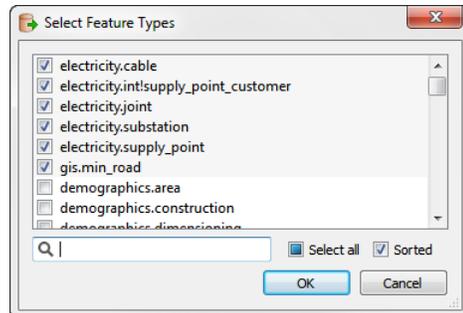
- electricity.cable
- electricity.joint
- electricity.substation
- electricity.supply_point
- electricity.int!supply_point_customer

The int! objects are the intermediate join tables.

Click **OK** to close the Select Feature Types dialog.

Uncheck **Use Search Envelope**.

4) Inspect Results
In the Data Inspector Table View you should see all of the selected objects. The
electricity.int!supply_point_customer is the join table for Customers & Supply Points.

You can use the same steps in Workbench to create a workspace that reads and preserves the
relationships between the Customer and Supply Point objects through the int! tables.

Example 9: Importing with a Simple Join	
Scenario	FME user
Data	Smallworld GIS database updated with changes from MIF Data
Overall Goal	Import a data with a simple join
Demonstrates	Configuring FME to import a child object and update the parent
Starting Workspace	Create from scratch
Finished Workspace	C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld9a-Complete.fmw

Simple Joins

In this example you'll load some new features for Company and Office. This is a 1:n relationship with Company as the parent and Office as the child table.

As mentioned above, to export join fields, you may have to make the fields visible in the Administration ACE application. In the following example, the Company object has had the 'id' field made visible and the Office object has had the office_id and company_id made visible. These fields are not exposed in the default Cambridge database.

To do this in Smallworld: you would use the Administration application to open the 'ACE Configuration' tool. Use the Edit – Object Properties menu to change the appropriate properties, including the field visibility.

First we'll add a new Company record to the Smallworld database:

1) Open Workspace
Start FME Workbench and open the workspace:
        C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld9a-Complete.fmw

2) Run Workspace
Run the workspace. It will add a new record into the Company onject in the |fme_offices alternative.

There is no geometry on the Company object so check the results in the Smallworld Explorer (Tools menu -> Explorer…). Navigate to Gis > Non Topological Objects > Company and browse all features – don't forget to switch to the GIS version |fme_offices

3) Open Workspace
 Open the second workspace for this exercise:
        C:\FMEData2015\Workspaces\SmallworldTraining\Smallworld9b-Complete.fmw
This is a little more interesting. This workspace adds a new Office record and makes the join to the Company object.
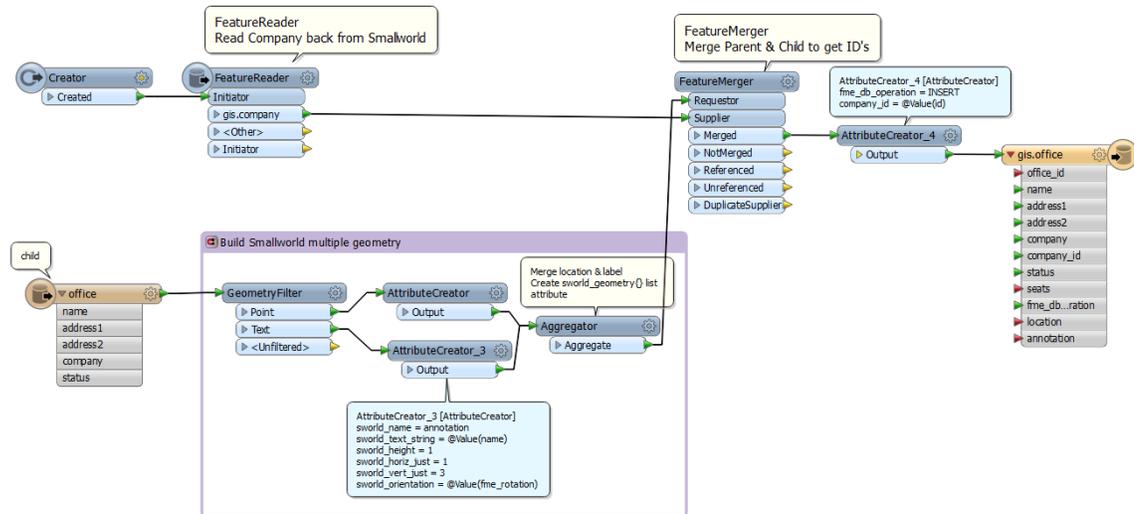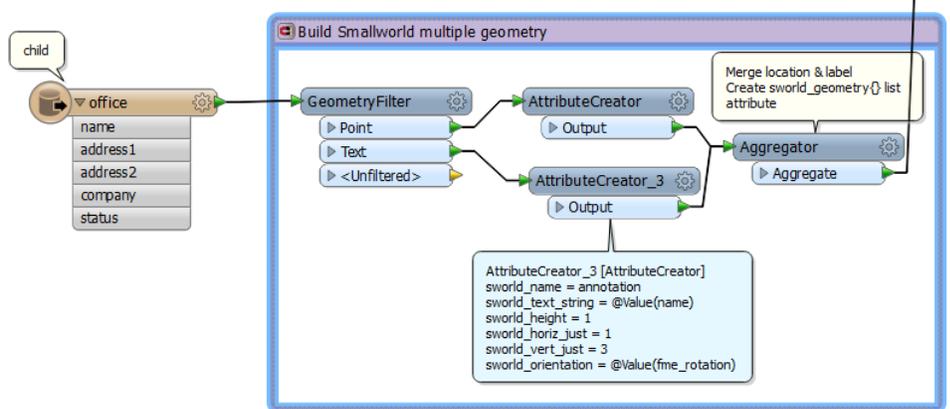

To make the join between the Office and the Company, you have to add the company primary key to the Office object – as the foreign key. The foreign key on the Office table is '**company_id'** which must match the primary key on the Company table which is '**id'**.

In the workspace:

The transformers in the Bookmark "**Build Smallworld multiple geometry**" creates the Office feature with location and annotation geometries, in the same way as the previous exercises.

FeatureReader: This workspace uses the FeatureReader to read back the Company records (we could have used a Smallworld reader). You could also add a predicate to the FeatureReader to limit which Company records are read back.

FeatureMerger: Join the Office object to the related Company. We can make the join using the company name. The office object now has the Company 'id' which can be used as the foreign key.

AttributeCreator: Rename the 'id' field (which is the Company primary key) to 'company_id' which is the foreign key on the Office child object.
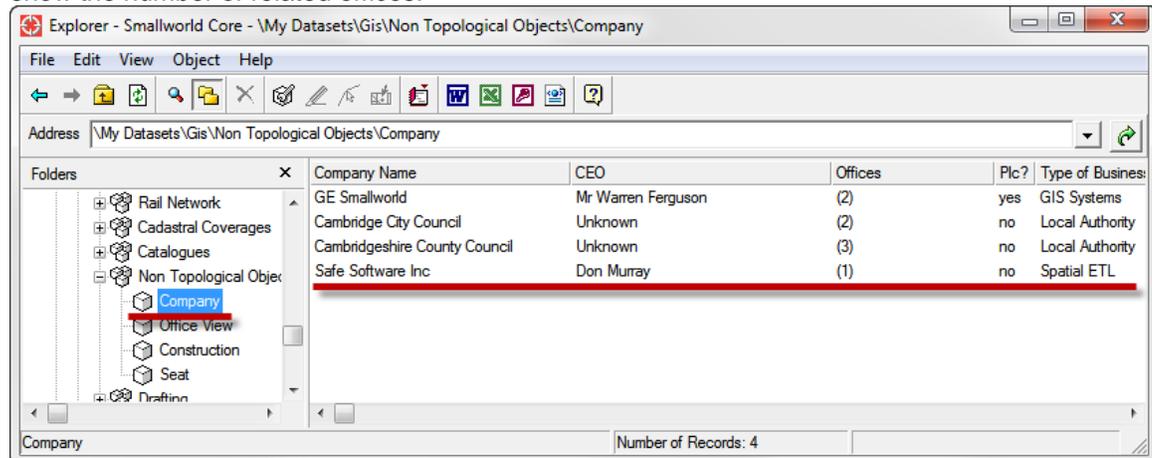
3) Run Workspace
Run the workspace. It will add a new record into the Office table in the |fme_offices alternative

Check the results in the Smallworld Explorer – don't forget to switch to the GIS version |fme_offices.

For the Company table, you should see the original Safe Software Inc company, but it should now show the number of related offices:


As mentioned before, you may have to use the Administration ACE tool to make some of the join fields visible.

In summary, to build a simple join relationship you need to set the foreign key on the child table to match the primary key of the parent table.

Network Topology

FME can export network topology, but to do this you need to expose the network topology fields in the Smallworld database. This requires some magik code installed into your Smallworld image. But once configured, FME would expose the internal topology ids for any objects on the Smallworld network.

Session Review
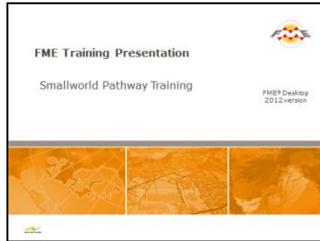
This module demonstrates how to use Spatial ETL to transform Smallworld data for use with other spatial formats

What You Should Have Learned from this Session

The following are key points to be learned from this session:

Theory

- Connecting to a Smallworld database requires the Smallworld 4 Suite - FME and the Smallworld FME Translator to be installed and operational.

- FME can represent Smallworld objects in two ways, in simplified form using FME Service, and in a "raw" form with the FMENOFACTORY Service.

- FME can read and write Smallworld ***multiple geometries***, using the 'sworld_geometry' list attribute

- When importing data to Smallworld, it is essential to inspect the source and destination schemas to ensure an accurate translation. The Smallworld CASE report is useful to show the Smallworld object schema

- FME can export just the differences between a baseline and alternative.

FME Skills

- The ability to use FME Data Inspector to inspect Smallworld data

- The ability to transform Smallworld data into various other formats

- The ability to use FME transformers to restructure data between Smallworld and other formats

- The ability to use the FME and FMENOFACTORY Services

- The ability to read and write single and multiple geometries

- The ability to specify a database operation for each feature when writing to Smallworld

- The ability to extract just the differences between a baseline and alternative.

- Working with simple joins and reading intermediate join tables.