



Data Synchronization: Autodesk® AutoCAD® Map 3D Enterprise, FME®, and ESRI® ArcGIS®

Drew Burgasser, P.E. – CAD Masters, Inc.

GS2475-R Sacramento Area Sewer District (SASD) chose AutoCAD® Map 3D Enterprise software industry models as the front-end GIS application for creating and editing sewer assets. SASD needed a synchronization solution so that the SASD enterprise GIS (ArcGIS®) could be updated accordingly. The long-transaction functionality (jobs) in AutoCAD® Map 3D Enterprise industry models provides a framework where inserted, edited, and deleted features can easily be queried for synchronization. In this roundtable session, we discuss how to use AutoCAD® Map 3D jobs in conjunction with FME® data conversion software to synchronize ESRI® ArcGIS®.

Learning Objectives

At the end of this class, you will be able to:

- Identify the database tables used in an AutoCAD® Map 3D Enterprise industry model to work with jobs.
- Customize job states to provide a synchronization state.
- Query inserts, edits, and deletes in an AutoCAD® Map 3D Enterprise industry model job.
- Explain how FME® data conversion software can be used in conjunction with Map 3D jobs to update ESRI® ArcGIS®.

About the Speaker

Drew is vice-president of CAD Masters, Inc., an Autodesk partner based in Northern California. He is also a registered professional engineer in the state of California. He has 20 years of experience using Autodesk® software. Drew specializes in implementation of civil engineering and GIS software solutions like AutoCAD® Civil 3D® and AutoCAD® Map 3D™. His combination of expertise in project management, civil engineering, GIS, .NET programming, Oracle, and IT makes him uniquely qualified with planning and implementation of Civil 3D and Map 3D in the most complex of environments. Drew received a BA in Architecture and an MS in Structural Engineering from the University of California at Berkeley.

drew@cadmasters.com

Introduction

Sacramento Area Sewer District (SASD)

Located in Sacramento, CA, SASD is a Sewer Utility providing service to more than 1 million people in the Sacramento region. The mission and visions of SASD are to efficiently collect sewage from homes and businesses within the Sacramento area and to provide the best value of any sewage collection utility in California, as measured by cost and level of service.

Why AutoCAD® Map 3D Enterprise?

SASD chose AutoCAD® Map 3D as a solution to modify sewer GIS data in a multiuser environment. AutoCAD® provides an unparalleled viewing and editing environment that the users were already familiar with. The long transaction (jobs) functionality in AutoCAD® Map 3D provides a QA/QC mechanism that allows SASD to process work through a series of states. Lastly, the AutoCAD® Map 3D API framework allowed SASD to build business rules and workflows from the ground up.

ESRI® ArcGIS®

AutoCAD® Map 3D was never intended to be a replacement for the SASD Enterprise GIS. SASD uses ArcSDE® (Spatial Database Engine) on SQL Server as their Enterprise solution. ESRI® products are used throughout the organization to view and analyze sewer asset.

The Challenge

Allow edits completed in AutoCAD® Map 3D to be copied to ESRI® ArcGIS®. In other words, the two disparate systems must remain synchronized.

Jobs

Job Tables

If you are a GIS administrator for AutoCAD® Map 3D Enterprise, you should become familiar with the job tables used as part of the long transaction functionality.

- TB_JOB – Contains the list of all jobs
- TB_JOB_STATE – Stores the job state configuration
- TB_JOB_STATE_TRANSITION – Stores the job state transition rules
- TB_JOB_VERSION – Stores the *complete history* of every feature.

Job States

As a user works on a job in AutoCAD® Map 3D, the job moves through a series of states. By default these states are as follows:

- Open
- Pending
- Live
- Deleted (optional!)

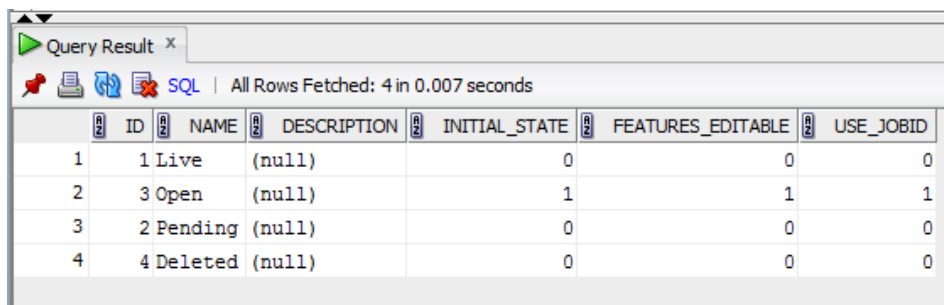
For SASD, it made sense to customize these job states based on their QA/QC processes and synchronization with ArcSDE®:

- Open
- Ready for QC
- QC Completed
- Synchronization
- Live

To modify the job states, modify the TB_JOB_STATE table. We use [Oracle® SQL Developer](#) to connect to Oracle® and query and edit data. You could alternatively use SQL Sheet (SqlSheet.exe) which can be found in the Autodesk Infrastructure Administrator bin folder.

To view job states:

```
select * from tb_job_state;
```



The screenshot shows the 'Query Result' window in Oracle SQL Developer. It displays the results of a query on the TB_JOB_STATE table. The window title is 'Query Result x'. Below the title bar, there are icons for saving, refreshing, and executing, along with the text 'All Rows Fetched: 4 in 0.007 seconds'. The table has 7 columns: ID, NAME, DESCRIPTION, INITIAL_STATE, FEATURES_EDITABLE, and USE_JOBID. The data is as follows:

ID	NAME	DESCRIPTION	INITIAL_STATE	FEATURES_EDITABLE	USE_JOBID
1	1 Live	(null)	0	0	0
2	3 Open	(null)	1	1	1
3	2 Pending	(null)	0	0	0
4	4 Deleted	(null)	0	0	0

To modify the 'Pending' state and rename it to 'Ready for QC':

```
update TB_JOB_STATE set NAME = 'Ready for QC' where ID = 2;
commit; --You need to commit any updates or inserts
```

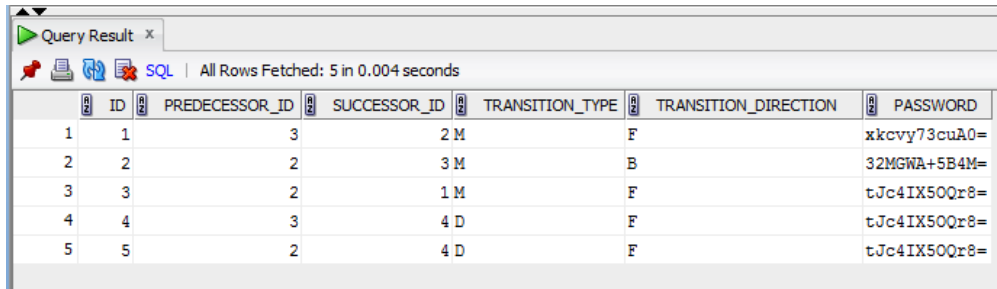
To add a new state named 'QC Completed':

```
insert into TB_JOB_STATE(ID, FEATURES_EDITABLE, INITIAL_STATE, NAME,
USE_JOBID) values ((SELECT MAX(ID) + 1 FROM TB_JOB_STATE), 0, 0, 'QC
Completed Priority', 0);
```

In addition to modifying the TB_JOB_STATE table you also need to modify the transitions between states using TB_JOB_STATE_TRANSITION table.

To view job states transitions:

```
select * from TB_JOB_STATE_TRANSITION;
```



ID	PREDECESSOR_ID	SUCCESSOR_ID	TRANSITION_TYPE	TRANSITION_DIRECTION	PASSWORD
1	1	3	2 M	F	xkcvy73cuA0=
2	2	2	3 M	B	32MGWA+5B4M=
3	3	2	1 M	F	tJc4IX50Qr8=
4	4	3	4 D	F	tJc4IX50Qr8=
5	5	2	4 D	F	tJc4IX50Qr8=

To add a transition from 'Ready for QC' to 'QC Completed':

```
insert into TB_JOB_STATE_TRANSITION(ID, PREDECESSOR_ID, SUCCESSOR_ID,
TRANSITION_DIRECTION, TRANSITION_TYPE) values ((SELECT MAX(ID) + 1
FROM tb_job_state_transition), 2, 5, 'F', 'M');
```

In this update statement, the ID for the 'Ready for QC' state is 2 and we're assuming the ID for the 'QC Completed' state is 5. Query the TB_JOB_STATE table to confirm.

You would also need to add transitions for QC Completed to Synchronization and Synchronization to Live, and remove the transition from Ready for QC to Live:

```
delete from TB_JOB_STATE_TRANSITION where id = ???;
```

Please note that you should do this work at the beginning of your implementation. At the very least, make sure all jobs are posted to live before modifying these two tables.

Using Job Tables to Query Inserts, Updates, and Deletes

To synchronize data, we need to know what features were inserted, updated, and deleted in a job or a group of jobs. SASD chose to use a custom job state named 'Synchronization' as a holding cell for jobs whose edits need to be copied (synchronized) to ArcSDE®.

Assuming we have a feature class named SSLATERALLINE and a job whose ID is 4, we can query the inserts, updates, and deletes in job 4 as follows.

First, if you don't already have function to set a job current, please add:

```
create or replace
function SETJOB(n number) return number is
begin
    INDUSTRYMODELNAME.job3.SETJOB(n);
    return (n);
end;
```

In this function, INDUSTRYMODELNAME is the name of your industry model.

Set job 4 current:

```
select SETJOB(4) from DUAL;
```

Since the data is not yet live (via the long transaction process), we need to set the job *current* to access the data.

The TB_JOB_VERSION table stores the *complete history* of every feature. This table contains a JOB_OPERATION_ID attribute that we can use to determine an insert vs. update vs. delete.

JOB_OPERATION_ID	Specifies the job operation ID. Mandatory. 1 = INSERT, 2 = UPDATE, 3 = DELETE - 1 = Indicates that this feature has already been existing, at the moment of the job enabling of the feature class.
------------------	--

To query all SSLATERALLINE features inserted, updated, or deleted in job 4:

```
Select * from SSLATERALLINE where FID in (select FID from
TB_JOB_VERSION where JOB_ID = 4);
```

Query inserted features from job 4:

```
select * from SSLATERALLINE where fid in (select FID from
TB_JOB_VERSION where JOB_ID = 4 and JOB_OPERATION_ID = 1);
```

Query updated features (simply change JOB_OPERATION_ID from 1 to 2):

```
select * from SSLATERALLINE where FID in (select FID from
TB_JOB_VERSION where JOB_ID = 4 and JOB_OPERATION_ID = 2);
```

How do we query deleted features? Do we simply change the JOB_OPERATION_ID value in the query above to 3? If all you need is the FID value of the deleted features, this will work:

```
Select fid from TB_JOB_VERSION where JOB_ID = 4 and JOB_OPERATION_ID =
3;
```

However, if you need other attributes for the deleted features, `select * from SSLATERALLINE` will return nothing (because they are deleted!). However, if we set the *Live* job current, the features are not yet deleted:

```
select SETJOB(1) from DUAL;
```

The live job always has an ID of 1. Now this will return the deleted features:

```
select * from SSLATERALLINE where FID in (select FID from
TB_JOB_VERSION where JOB_ID = 4 and JOB_OPERATION_ID = 3);
```

This tracing process using the TB_JOB_VERSION table is critical to knowing what features need to be synchronized.

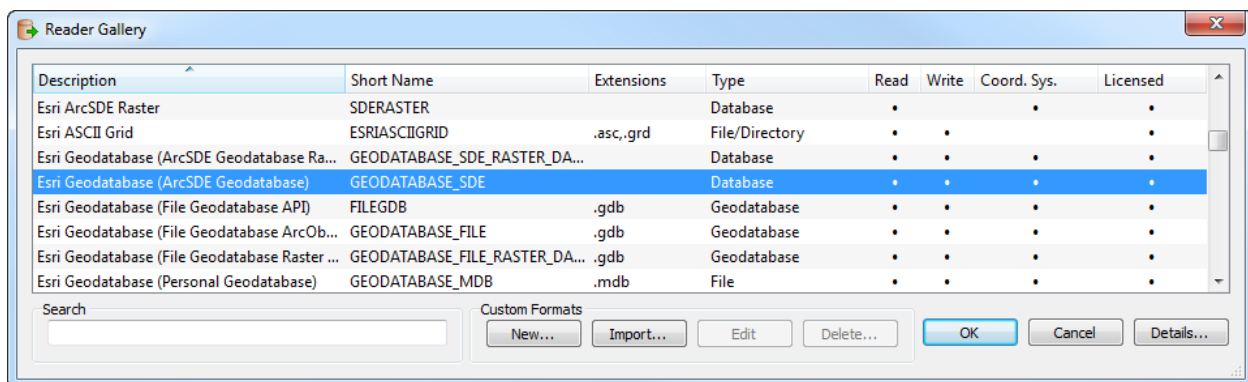
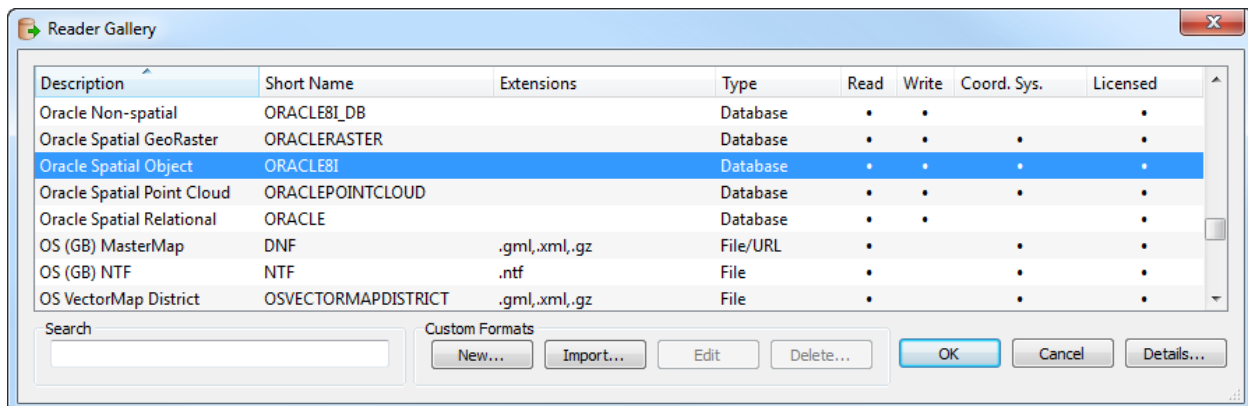
FME® Desktop

In the previous section, we learned how to query the features that we want to synchronize with ArcSDE®. Next, we need to establish how we are going to do the synchronization work. How are we going to communicate with both systems and copy data from one AutoCAD® Map 3D to ArcSDE®? SASD chose to use [FME® Desktop](#) from [Safe Software](#) for this task.

FME® Desktop can translate between over 300 supported formats. It will translate both geometry and attribute data. Transformers are used to configure workflows as data moves from one format to another.

Readers and Writers

FME® can read and write to both AutoCAD® Map 3D (via the Oracle® Spatial Object) and ArcSDE®.



Two separate readers are created for AutoCAD® Map 3D – one for inserts/updates and another for deletes. Remember earlier the need to SETJOB differently for deletes. That's why we need two separate readers.

Reader for Inserts and Updates

Oracle Inserts and Updates

- Coordinate System: <not set>
- Parameters
 - Source Oracle Spatial Object Service: OSERVICE (Linked to 'ORACLE_SERVICE')
 - Username: ORACLE_SEWER_DB (Linked to 'ORACLE_INDUSTRY_MODEL')
 - Password: ***** (Linked to 'ORACLE_PASSWORD')
 - Oracle Workspace: <not set>
- Advanced
 - Persistent Connection: Yes
 - Remove Schema Qualifier: No
 - Read 3D Polygons as Faces: No
 - WHERE Clause: FID in (select FID from TB_JOB_VERSION where JOB_ID = 7) (Linked to 'ORACLE_WHERE_CLAUSE')
 - Minimum X: 0
 - Minimum Y: 0
 - Maximum X: 0
 - Maximum Y: 0
 - Clip to Search Envelope: No
 - Search Envelope Coordinate System: <not set>
 - Relationship To Query Feature: ANYINTERACT
 - Relationship Result Test: = 'TRUE'
 - Rows To Read At A Time: 200
 - SQL Statement To Execute Before Translation: select SETJOB(7) from dual (Linked to 'ORACLE_SQL_BEFORE')
 - SQL Statement To Execute After Translation: <not set>
 - Handle Multiple Spatial Columns: No

Using the queries we learned earlier.

Set the job current.

Reader for Deletes

Oracle Deletes

- Coordinate System: <not set>
- Parameters
 - Source Oracle Spatial Object Service: OSERVICE (Linked to 'ORACLE_SERVICE')
 - Username: ORACLE_SEWER_DB (Linked to 'ORACLE_INDUSTRY_MODEL')
 - Password: ***** (Linked to 'ORACLE_PASSWORD')
 - Oracle Workspace: <not set>
- Advanced
 - Persistent Connection: Yes
 - Remove Schema Qualifier: No
 - Read 3D Polygons as Faces: No
 - WHERE Clause: FID in (select FID from TB_JOB_VERSION where JOB_ID = 7 and JOB_OPERATION_ID = 3) (Linked to 'ORACLE_WHERE_CLAUSE')
 - Minimum X: 0
 - Minimum Y: 0
 - Maximum X: 0
 - Maximum Y: 0
 - Clip to Search Envelope: No
 - Search Envelope Coordinate System: <not set>
 - Relationship To Query Feature: ANYINTERACT
 - Relationship Result Test: = 'TRUE'
 - Rows To Read At A Time: 200
 - SQL Statement To Execute Before Translation: select setjob(1) from dual
 - SQL Statement To Execute After Translation: <not set>
 - Handle Multiple Spatial Columns: No

Using the queries we learned earlier.

Set the *Live* job current to get at the deleted features.

There's only one writer for ArcSDE®.

ArcSDE Writer

ArcSDE

- Coordinate System: <not set>
- Parameters
 - Destination ESRI Geodatabase (ArcSDE) Dataset: SDE_DB_NAME (Linked to 'SDE_DATABASE')
 - Server: SDESERVERNAME (Linked to 'SDE_SERVER')
 - Instance Name: sdesqlserver:SDESERVERNAME (Linked to 'SDE_INSTANCE')
 - Transactional Version: SDE_VERSION_NAME (Linked to 'SDE_VERSION')
 - Template File: <not set>
 - Default Z Value: 0
 - Contains Measures: No
 - Measures Origin: 0
 - Measures Scale: 100
- Advanced
 - Transaction Type: VERSIONING
 - Simplify Geometry: no
 - X Origin: 0
 - Y Origin: 0
 - Scale: 0
 - Contains Z Values: auto_detect
 - Z Origin: 0
 - Z Scale: 1
 - Grid 1 Size: 0
 - Writer Mode: UPDATE
 - Transaction Number: 0

Create version in ArcSDE® and specify here if ArcSDE® is configured for versioning.

Writer assumes UPDATE. We specify INSERTS and DELETES later.

Parameters

Parameters are used in FME so that certain information can be fed to FME when running a batch of synchronization tasks. For example, each job has its own ID and that ID must be fed to FME to do the synchronization work.

User Parameters

- Published Parameters
 - [ORACLE_JOB_ID] Map 3D Job Number : 7
 - [SDE_VERSION] Transactional Version : SDE_VERSION_NAME
 - [ORACLE_SQL_BEFORE] SQL Statement To Execute Before Translation : select SETJOB(\$(ORACLE_JOB_ID)) from dual
 - [ORACLE_WHERE_CLAUSE] WHERE Clause : FID in (select FID from TB_JOB_VERSION where JOB_ID = \$(ORACLE_JOB_ID))
 - [ORACLE_DELETE_WHERE_CLAUSE] WHERE Clause : FID in (select FID from TB_JOB_VERSION where JOB_ID = \$(ORACLE_JOB_ID) and JOB_OPERATION_ID = 3)
 - [ORACLE_SERVICE] Oracle Service : OSERVICE
 - [ORACLE_INDUSTRY_MODEL] Oracle Industry Model : ORACLE_SEWER_DB
 - [ORACLE_PASSWORD] Oracle Industry Model Password : *****
 - [SDE_INSTANCE] SDE Instance Name : sdesqlserver:SDESERVERNAME
 - [SDE_DATABASE] Destination ESRI Geodatabase (ArcSDE) Dataset : SDE_DB_NAME
 - [SDE_SERVER] SDE Server : SDESERVERNAME

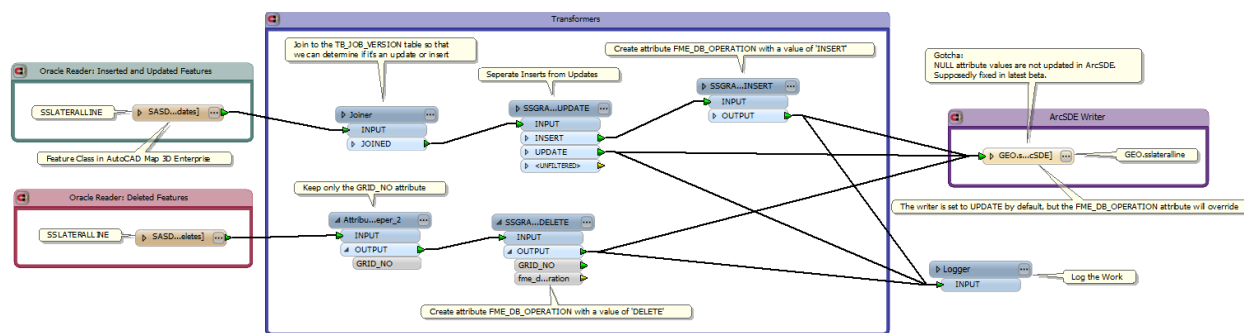
JOB ID

Key Attribute

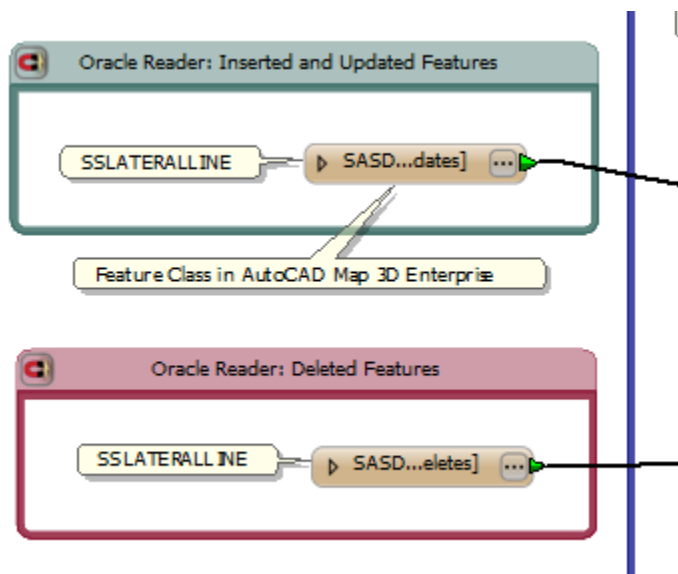
In order to update and delete data, we must establish a *key attribute* that exists in both systems. We can use the AutoCAD® Map 3D FID (unique identifier) attribute as long as that attribute is created in ArcSDE® and the values are updated to match AutoCAD® Map 3D's FID values. In the case of SASD, we used a pre-existing identifier named GRID_NO which is an identifier based on the feature's location in the city grid. A feature rule in AutoCAD® Map 3D assigns this attribute value its proper value when a feature is created.

Workspace

Workspaces are configured to do the data translation. This is entire workspace:



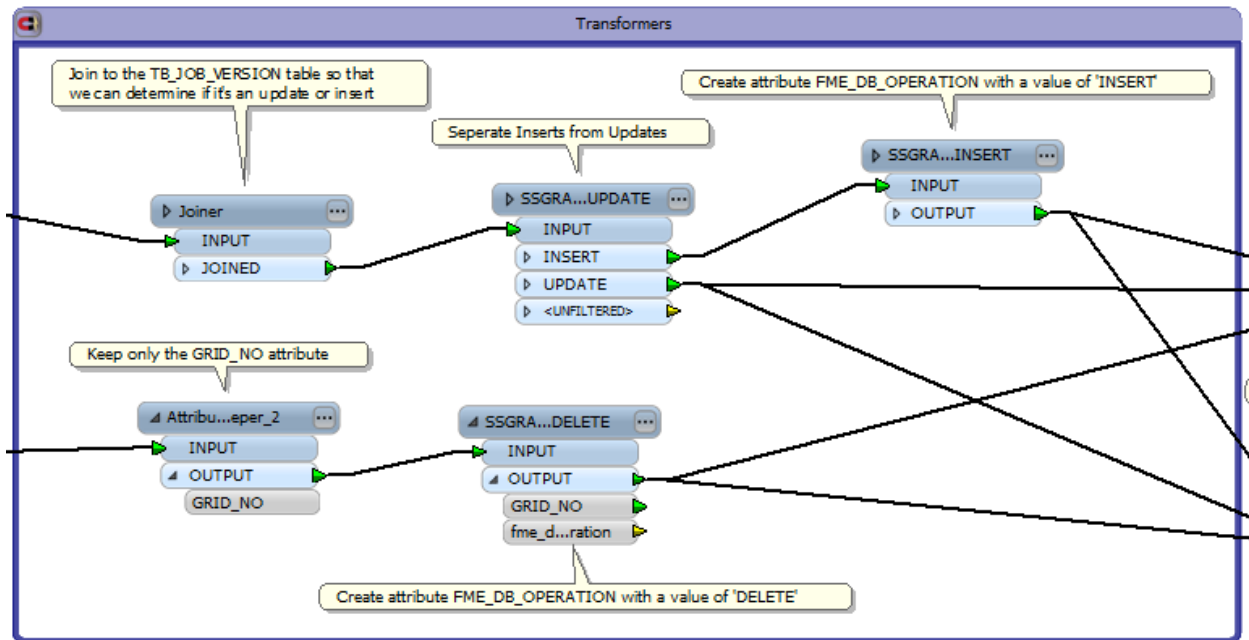
Firstly, we read the data from AutoCAD® Map 3D. In this case, it's just one feature class: SSLATERALLINE.



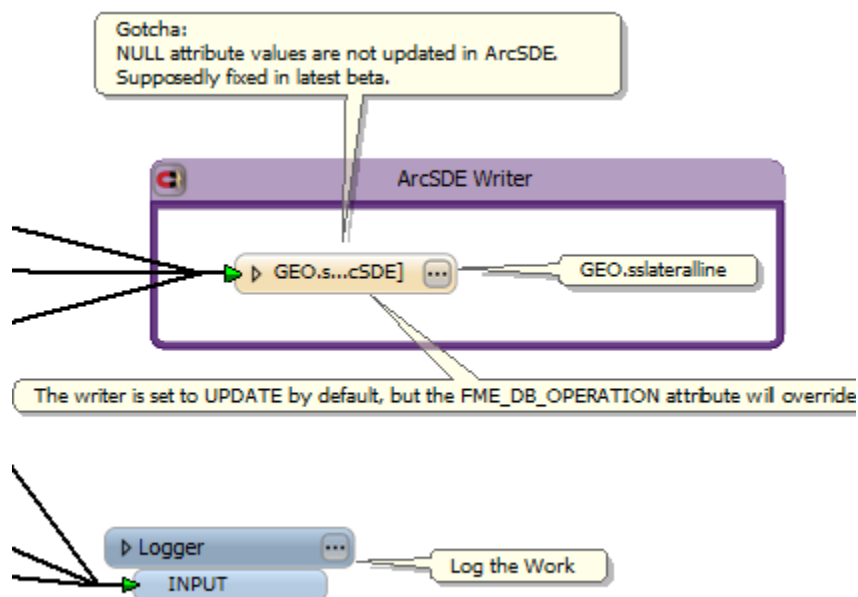
Next, we 'transform' the data. Specifically we're doing the following:

1. Using a Joiner transformer to query TB_JOB_VERSION to determine if the feature is inserted or updated.

- Adding an FME_DB_OPERATION attribute and values to features that are inserted and deleted. By default, the ArcSDE® writer will execute an update statement. The FME_DB_OPERATION attribute will control the writer statement type.



- Writing to ArcSDE® is the last step.



Executing The Synchronization Workspaces

These workspaces can be run in command line batch mode, thus eliminating the need to 'open' FME Workspace each time. A parameter file is configured and then FME.exe is executed with the parameter file as an argument. Please see the following FME® article on batch processing:

<http://fmepedia.safe.com/articles/FAQ/FME-Batch-Processing-Methods-Advantages-Disadvantages>

Conclusion

FME® Desktop can be used to translate data between disparate data sources. For SASD, FME® Desktop is used to synchronize ESRI® ArcSDE® with AutoCAD® Map 3D Enterprise. Using the job tables and queries discussed earlier, FME® can be configured to read job edits and translate those edits to ArcSDE.