SHOP FLOOR EXPERTS, LLC

SyteLine Configuration – SSSync with Multi-Site – White Paper

The purpose of this white paper is to explain how the SSSync Synchronization program functions and the impact that has with Multi-Site customers.

SSSync Overview

During model building, Custom Parts and Part Classes are used to develop the model by "pulling" Items and Work Centers from SyteLine into the model. Attributes values are defined by mapping SyteLine fields to Part Attributes in the model. Common fields mapped are Item, Description, Price, Product Code, etc.

SSSync is used to keep the attribute values in the model in sync with the fields in SyteLine.
This way, values can be maintained in SyteLine, then synced into the model.
SSSync is usually executed periodically to maintain Price or Description changes.

When Custom Part and Part Classes are created, a "sync" record is created which will be used by the SSSync program. This sync record keeps track of the Parts that need to be synced. It also keeps track of the SyteLine database where that part came from. To accomplish that, the sync record stores the Data Source Name (DSN) of the ODBC connection that is used.

During installation of SyteLine Configuration, ODBC connections are setup through Control Panel. The following ODBC connections are usually assigned:

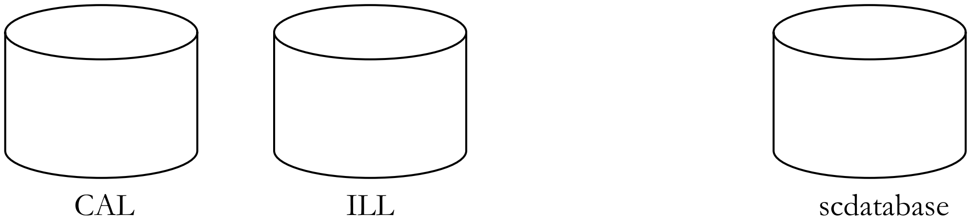
SyteLine	Intersolv driver pointing to the symix database
SyteSelect	Intersolv driver pointing to the symixcfg database
SellingChain	Access or SQL pointing to the scdatabase

The SyteLine DSN is the default ODBC connection for creating Custom Parts and Part Classes.

Typically the above setup is satisfactory for a Single-Site customer.

SyteLine Configuration – SSSync with Multi-Site – White Paper (cont'd)

Multi-Site Considerations



A typical Multi-Site setup might consist of 2 sites, CAL and ILL, and 1 SellingChain database, if all the databases are being stored in a central location.
Dave is developing a model in CAL. Lisa is developing a model in ILL. Both are using the same scdatabase.

The main consideration is the ODBC connections established for Dave and Lisa. Dave could have an ODBC connection called SyteLine pointing to CAL. Lisa could have an ODBC connection called SyteLine pointing to ILL. For model development, this would appear to work fine. When the sync program is executed, it will sync all models in the database. Both models were created with a DSN of SyteLine, but they were pointing to different databases. Therefore, if Dave ran the sync program, it would sync both models using the CAL database. This could improperly change information in Lisa's model. The same is true for Dave if Lisa runs the sync program.

The proper way to setup ODBC connection for the above example is to create an ODBC connection called CAL for the CAL database and one called ILL for the ILL database.
Dave will use the CAL DSN for Custom Parts and Part Classes. Lisa will use the ILL DSN for Custom Parts and Part Classes. The sync records will be created with the correct DSNs.
When the sync program is executed, it will sync the data from the correct database as long as both ODBC connections are setup on the Client PC.