SAP Ariba Metadata Scanner

Installation Requirements

The following document details the installation and configuration requirements to extract the metadata necessary from within SAP Ariba.

This scanner leverages the Swagger | REST open API standard parsing capability in order to discovery entities, schema and relationships in the SAP Ariba tenant.

You must be a licensed customer of SAP Ariba to use this scanner.

Contents of the Installation Guide:

1. This document
2. SQL file that is run as a script to install
    a. Creates the Swagger | REST application registrations
        i. These are needed to create the metadata used to create the system type and Taxonomies.
    b. System Type Model for SAP Ariba
        i. This is needed to create the system type
3. Swagger files for each REST API of Ariba

Installation:

Setup DataSource in Administration:

1. Create Data Source in the Admin → Data Source for underlying database connection
    a. One Data Source needed
2. Enter the following data
    a. Database Type
    b. Username
    c. Password
3. Create Data Source for the Collect Target local SQL database (i.e. dgSAP)
4. Register Data source created in step 3 as a Collect Target
5. Register the Data Source created in steps 1 and 2 as Target Sources
6. Register the required objects in the list above to the appropriate Target Data Source.
7. Build and Refresh each successfully registered object.

8. Confirm data is being returned by looking at the record count on the Target source Table registration.

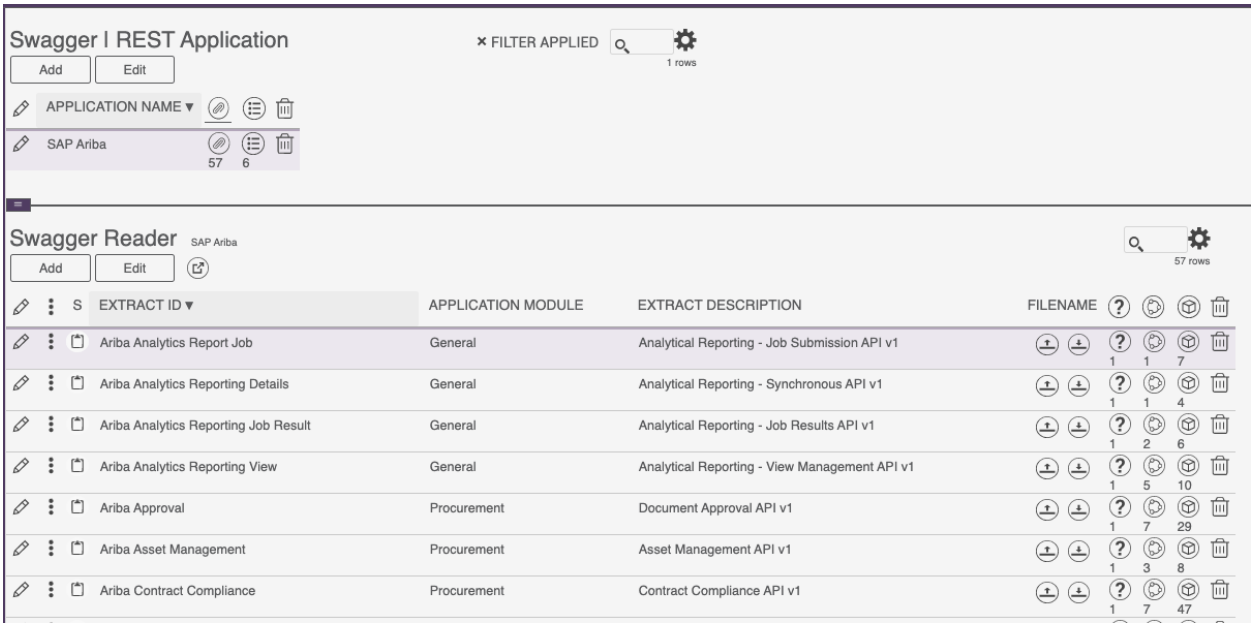
Using the DSP application Collect, download the above tables or views into a Collect Target (i.e. dgSAP; dgSAP_APO)

Install Scanner

1. Run the install script for the SAP Ariba metadata scanner.
    a. *SAP Ariba ScannerTech_Install.sql*
2. Move all the Swagger files in to the file location on the application server where the data source SwaggerFiles point to (i.e. D:\SwaggerFiles\)

Read Swagger Files

1. Navigate to Configuration → Metadata Scanner Technology → Swagger | Rest Applications
    a. SAP Ariba will be there if the install script in SQL was run successfully

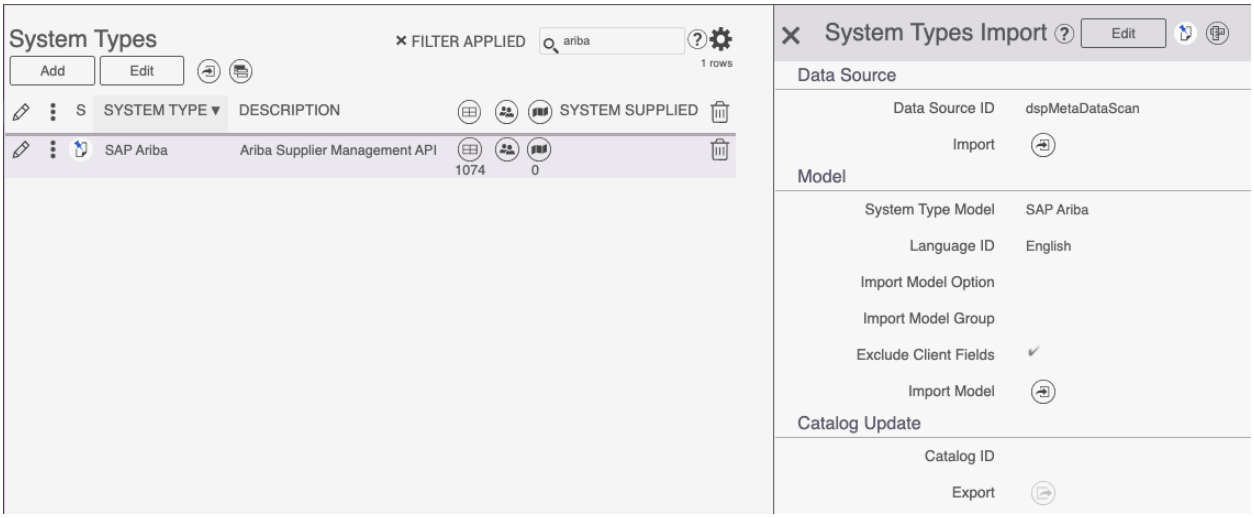


1. For the SAP Ariba Application, you can now read the metadata from all the Swagger Reader records.
2. This can be done one by one by pressing the **Read** button on the toolbar of the Swagger Reader page.

a. Or it can be done in mass by highlighting all the records using the SHIFT and arrows buttons on your keyboard.
b. Once all the desired records are highlighted, press the **Read** button to retrieve the metadata from each Extract ID.
c. Once completed, the System Type can be created and imported from the System Type Model.

Create System Type from Model

1. Create a new System Type (i.e. SAP Ariba)
2. Import the data from the System Type Model SAP Ariba. (SAP Ariba is supplied in the install script).
3. Import Model



Add Data Base, Instance and Schemas

1. Navigate to Syniti Metadata Discovery application, System Type | Extension page.
2. Locate the System Type name created in the steps above.
3. Enter Instance and Database information (Instance is typically the business name of the application often just called Ariba. Database is not relevant but some metadata exchange capabilities require it. A Default value of NA or aribaprod will suffice).
4. Navigate to the Tables and enter schema information in mass update or individually. (Schema doesn't make sense for cloud based applications; a default of NA or ariba can be used).

Build Application Module Hierarchy

1. Navigate to the Vertical View of the System Type| Extension created in the steps above.
2. Edit the following fields to the suggested values:
    a. Scanner Technology = Swagger Reader
    b. Swagger Application = SAP Ariba
3. Navigate to the Action Settings Tab
    a. Under the Analysis Action Label
        i. Click **Create SAP Ariba Taxonomy**
    b. Results will create the following:
        i. Navigation of the Application Module Hierarchical Taxonomy on the Horizontal and Vertical Views
        ii. Searchable flat Taxonomy by Table, Application Module or Transaction (for Ariba's case, API resource paths).