MICRO FOCUS | SERENA

Deployment Automation
Sample Database Deployment Process

Trademarks

U.S. Government Rights

Part number: 6.1.5

Publication date: 2017-09-22

Table of Contents

Sample Database Deployment Process

This document assumes that you have imported the *Sample Database Update Using Scripts* sample from the Deployment Automation Samples category in Community website. In Deployment Automation, you can view the pre-configured processes for this sample. This document outlines the steps taken to design the processes.

About the Sample

This sample demonstrates how to update a third-party database using SQL scripts. The process to deploy the sample database component follows these steps:

1. Delete existing files and directories
2. Save the latest artifacts to the destination server
3. Execute SQL scripts using a JDBC driver

This sample uses the following plugins:

- FileUtils
- DA Versioned File Storage
- SQL-JDBC

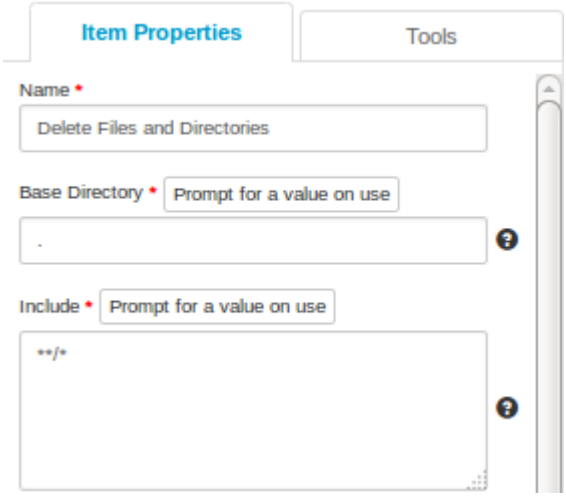
Designing the Process to Deploy the Database Component

Step 1: Clean up Temp Storage (Delete Files and Directories)

The first step will delete the files from all subdirectories, starting at the base directory for the Sample Database Component (script) component.

1. Navigate to **Management > Components**.
2. Select the **Sample Database Component (script)** component and then select the **Processes** tab.
3. Click the **Create Process** button.
4. Enter the name `Deploy Database` and accept the default values for the rest of the fields.
5. Select the process and on the **Tools** tab, expand the menu item **Utilities > FileUtils**.
6. Drag the **Delete Files and Directories** step onto the design space.
7. On the **Item Properties** tab, enter these values for the following fields:
   - **Base Directory**: .
   - **Include**: `**/*`

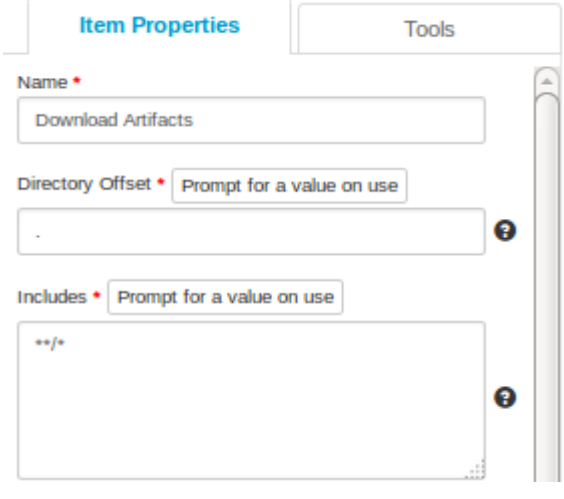
8. Use the default values for the remaining fields.



Step 2: Download Artifacts

The second step will download all of the artifacts for the Sample Database Component (script) version from CodeStation to the component's base directory.

1. On the **Tools** tab, expand the menu item **Repositories > Artifact > DA**.
2. Drag the **Download Artifacts** step onto the design space.
3. Use the default values for all fields.



Step 3: Execute SQL Scripts

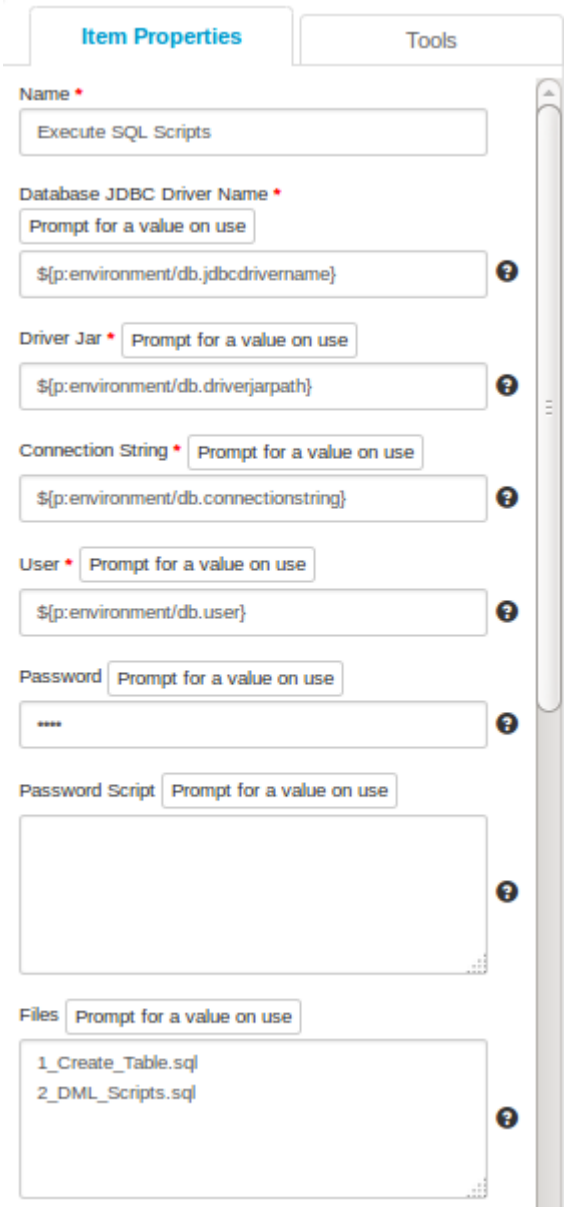
The third step will execute all SQL from all directories, starting at the base directory for the component.

1. On the **Tools** tab, expand the menu item **Database > SQL JDBC**.
2. Drag the **Execute SQL Scripts** step onto the design space.

3. On the **Item Properties** tab, enter these values for the following fields:
   - **Database JDBC Driver name**: `${p:environment/db.jdbcdrivername}`
   - **Driver Jar**: `${p:environment/db.driverjarpath}`
   - **Connection String**: `${p:environment/db.connectionstring}`
   - **User**: `${p:environment/db.user}`
   - **Password**: *password*
   - **Files**:

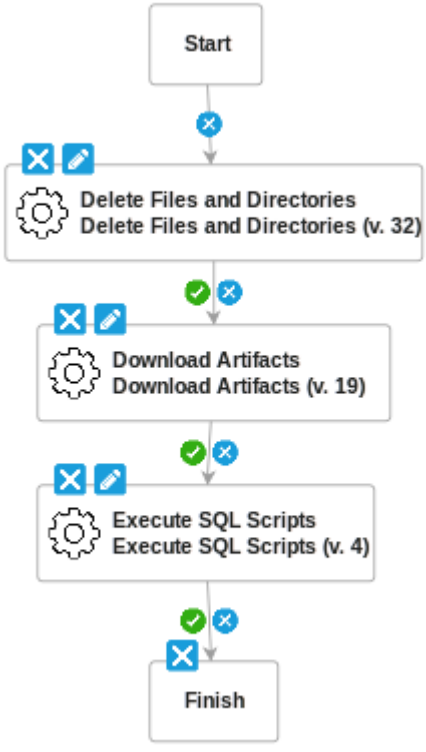
     `1_Create_Table.sql`

     `2_DML_scripts.sql`
4. Use the default values for the remaining fields.



The Sample Database Script Component Deployment Graphic Process

After selecting the preceding steps and filling in their properties, connect the steps as shown in the following graphical view of the Sample Database Component (script) component process.



Configuring the Component Environment Properties

You must configure the environment properties for the component so that the property values are filled in uniquely for each environment. This is done in two parts, as described in the following sections.

Adding the Component Environment Properties

You must first add the environment properties to the component without defining their values. Remember that you used variables in the processes when you referenced the Sample Database Component (script) information.

1. Select the **Sample Database Component (script)** component and then select the **Details** tab.
2. In the selection box, select **Environment Property Definitions**.
3. Click the **Add Property** button.
4. Add the following properties with no values:
   - db.connectionstring
   - db.driverjarpath
   - db.jdbcdrivername

- db.password
- db.user

Designating Values for the Properties in the Environment

After you have added environment properties at the component level, you can set the values for them for each environment to which the component is mapped. Setting values for the properties at the environment level ensures that the variables in the component processes will be appropriately resolved for each environment you run the process against.

1. Navigate to **Management > Applications**.
2. Select **Sample DB Application** and then select the **Environments** tab.
3. Click the **INT** environment and then select **View Details**.
4. Select the **Component Mapping** tab.
5. Select the **Properties** tab.
6. Click **Edit**.
7. Enter values for the environment properties that you previously defined for the component:
   - db.connectionstring: `jdbc:mysql://localhost:3306/INTdb`
   - db.driverjarpath: `/home/serena/SampleArtifacts/mysql-connector-java-5.1.24-bin.jar`
   - db.jdbcdrivername: `com.mysql.jdbc.Driver`
   - db.password: *`password`*
   - db.user: `root`

The properties are now set for the INT environment and are shown in the following figure.

Resource Groups | Properties

Name	Label	Required	Value
db.connectionstring	Database Connection String	false	jdbc:mysql://localhost:3306/INTdb
db.driverjarpath	db.driverjarpath	false	/home/serena/SampleArtifacts/mysql-connector-java-5.1.24-bin.jar
db.jdbcdrivername	JDBC Driver Name	false	com.mysql.jdbc.Driver
db.password	Database Password	false	****
db.user	Database User	false	root

Designing an Application Process to Run the Deployment

Now you will design an application process for the Sample DB Application that will kick off the component process that you have set up.

For the Sample DB Application, you will create an application process called **Upgrade Database**.

1. Select the **Sample DB Application** and then select the **Processes** tab.
2. Click **Create Process**.
3. Enter the name `Upgrade Database` and accept the default values for the rest of the fields.

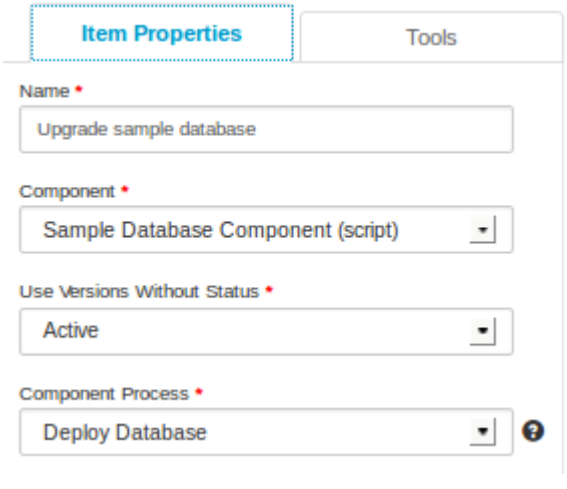
Next, design the application process, **Upgrade Database**, as follows:

1. Execute the Sample Database Component (script) process.

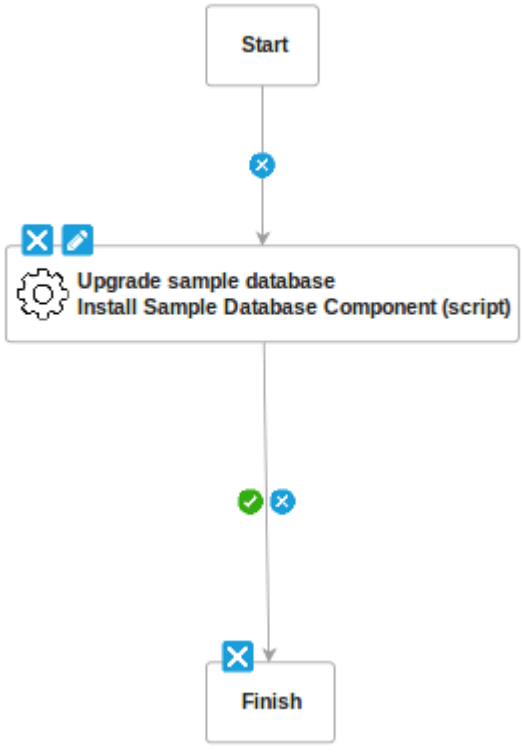
   Drag the **Install Component** step onto the design space and fill in the properties as follows:

   - **Name**: `Upgrade sample database`
   - **Component**: `Sample Database Component (script)`
   - **Component Process**: `Deploy Database`

2. Use the default values for the remaining fields.



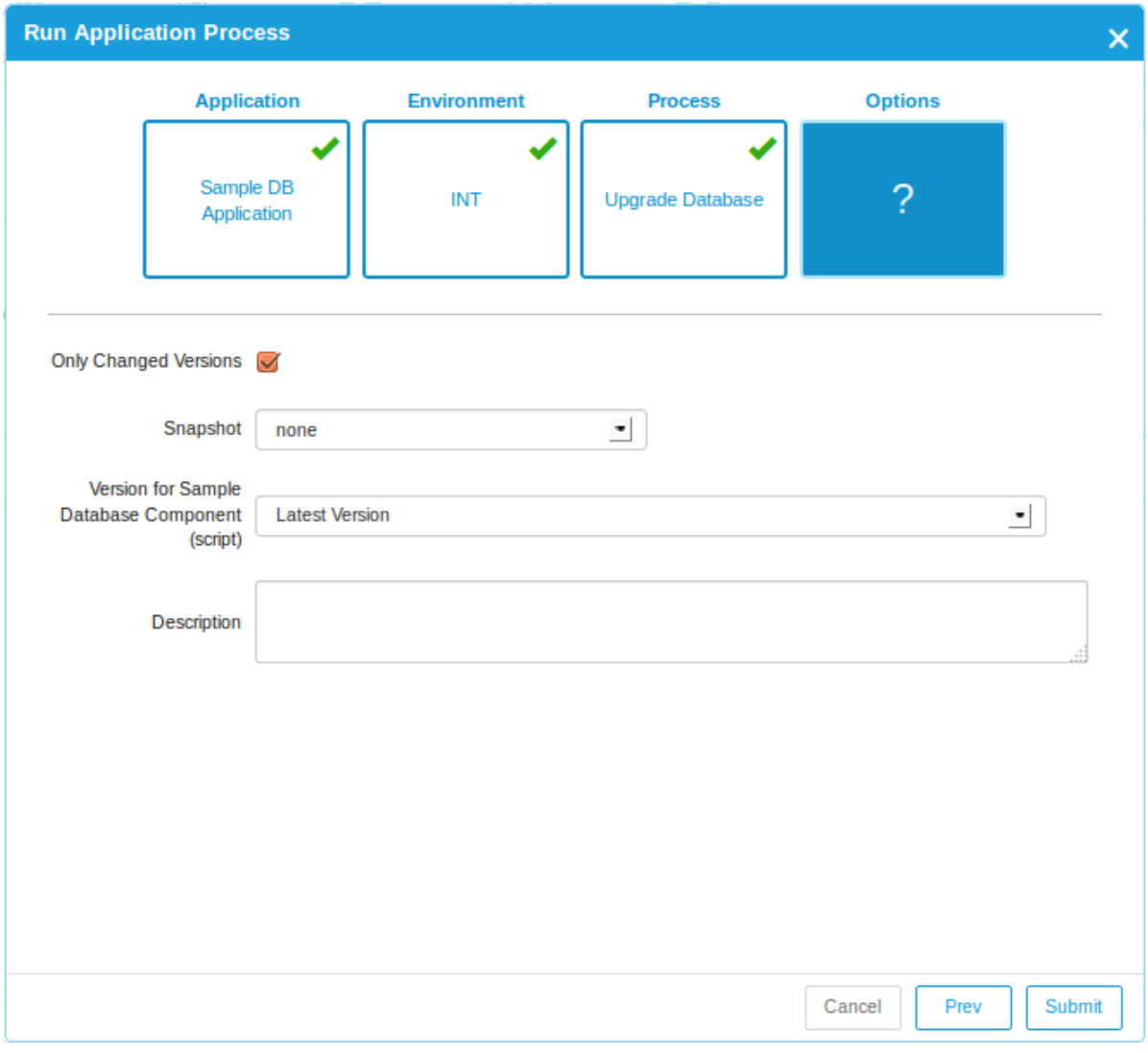
3. Connect the steps as shown in the following graphical view of the application process, Upgrade Database.



Deploying the Application

To perform our example deployment, run the application process from the INT environment as follows:

1. Select **Sample DB Application** and then select the **Environments** tab.
2. Click the **INT** environment and then select **Run Process**.
3. Select **Upgrade Database** and click **Next**.
4. In the **Version for Sample Database Component (script)** field, select **Latest Version**.
5. Look for and resolve any warnings at the bottom of the dialog box before submitting.
6. Click **Submit**.


Run Application Process
Application
Environment
Process
Options
Sample DB Application
INT
Upgrade Database
?
Only Changed Versions
Snapshot
none
Version for Sample Database Component (script)
Latest Version
Description
Cancel
Prev
Submit