MICRO FOCUS | SERENA

Deployment Automation
Sample Tomcat Deployment Process

Trademarks

U.S. Government Rights

Part number: 6.1.5

Publication date: 2017-09-22

Table of Contents

Sample Tomcat Deployment Process

This document assumes that you have imported the *Sample Tomcat Plugin Using Database 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 consists of two components. The first component delivers a Tomcat application to the Tomcat web server. The second component uses Datical DB to perform database updates (see Deploying a Datical Database Component [page 10]).

The process to deploy an application to the Tomcat server follows these steps:

1. Delete existing files and directories
2. Save the latest artifacts to the destination server
3. Start the Tomcat service
4. Deploy the sample Tomcat application

This sample uses the following plugins:

- FileUtils
- DA Versioned File Storage
- Tomcat

Designing a Process to Deploy the Sample Tomcat Component

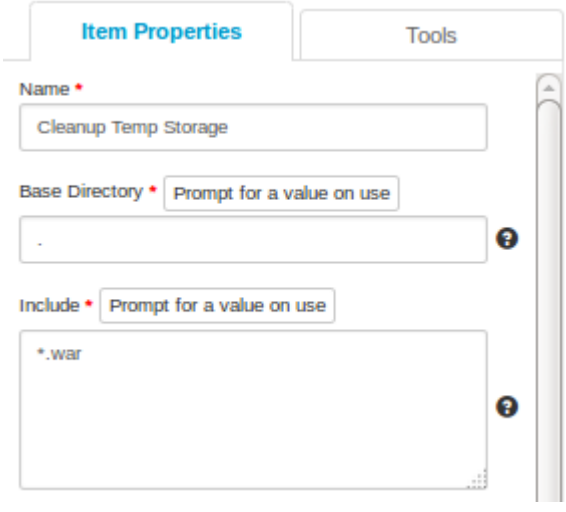
Step 1: Clean up Temp Storage (Delete Files and Directories)

The first step will delete the files from all subdirectories, starting at the base directory for the Sample Tomcat Component.

1. Navigate to **Management > Components**.
2. Select **Sample Tomcat Component** and then select the **Processes** tab.
3. Enter the name `Install Process on Tomcat` and accept the default values for the rest of the fields.
4. Select the process and on the **Tools** tab, expand the menu item **Utilities > FileUtils**.
5. Drag the **Delete Files and Directories** step onto the design space.
6. On the **Item Properties** tab, enter these values for the following fields:

- **Name**: `Cleanup Temp Storage`
- **Base Directory**: `.`
- **Include**: `*.war`

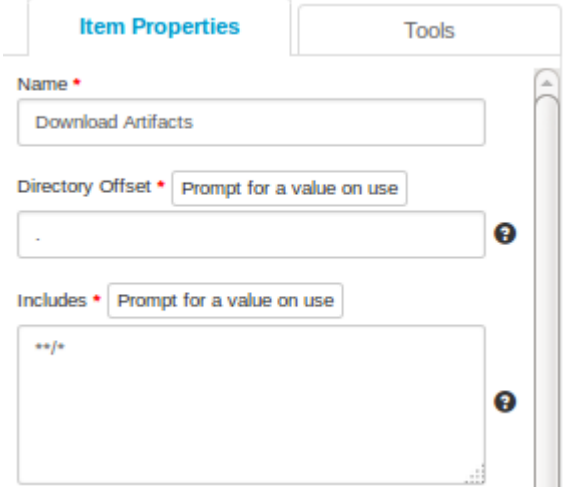
7. Use the default values for the remaining fields.



Step 2: Download Artifacts

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

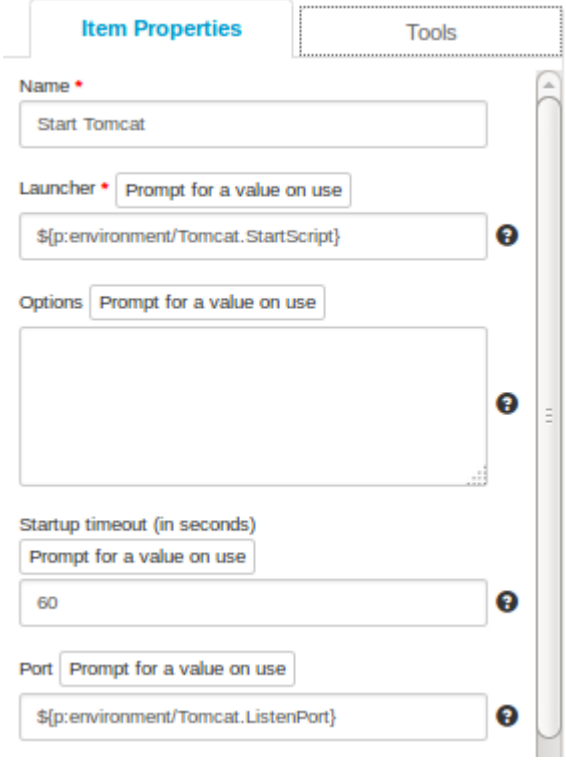
1. On the **Tools** tab, expand the menu item **Repositories > Artifact > DA**.
2. Drag the **Download Artifacts** step onto the design space.
3. Use the default values for all fields.



Step 3: Start Tomcat

The third step will start the Tomcat Service.

1. On the **Tools** tab, expand the menu item **Application Server > Java > Tomcat**.
2. Drag the **Start Tomcat** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Launcher**: `${p:environment/Tomcat.StartScript}`
   - **Startup timeout**: `60`
   - **Port**: `${p:environment/Tomcat.ListenPort}`
4. Use the default values for the remaining fields.



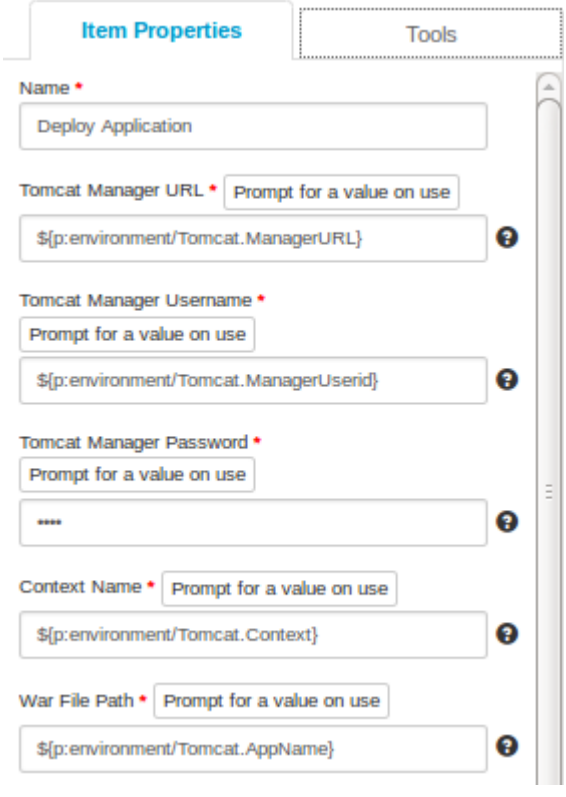
Step 4: Deploy Application

The fourth step will deploy the Sample Tomcat Component.

1. On the **Tools** tab, expand the menu item **Application Server > Java > Tomcat**.
2. Drag the **Deploy Application** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Tomcat Manager URL**: `${p:environment/Tomcat.ManagerURL}`
   - **Tomcat Manager Username**: `${p:environment/Tomcat.ManagerUserid}`

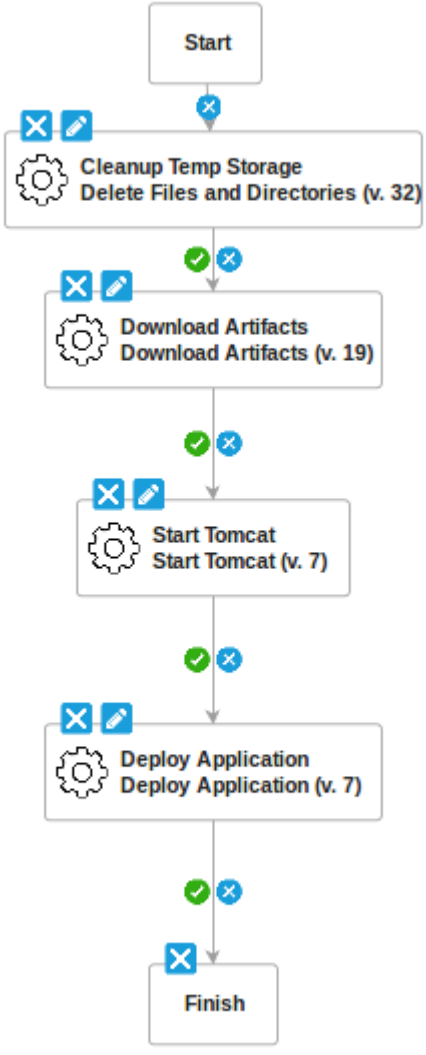
- **Tomcat Manager Password**: *password*
- **Context Name**: `${p:environment/Tomcat.Context}`
- **War File Path**: `${p:environment/Tomcat.AppName}`

4. Use the default values for the remaining fields.



The Sample Tomcat 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 Tomcat 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 Tomcat information.

1. Select **Sample Tomcat 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:
   - Tomcat.AppName
   - Tomcat.Context
   - Tomcat.HostName
   - Tomcat.ListenPort
   - Tomcat.ManagerPassword
   - Tomcat.ManagerURL
   - Tomcat.ManagerUserid
   - Tomcat.StartScript
   - Tomcat.Type

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 Tomcat 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 **Sample Tomcat Component** and then select the **Properties** tab.
6. Click **Edit**.
7. Enter values for the environment properties that you previously defined for the component:
   - Tomcat.AppName: `Qlarius_Online.war`
   - Tomcat.Context: `/Qlarius_Online.war`
   - Tomcat.HostName: `localhost`
   - Tomcat.ListenPort: `8092`
   - Tomcat.ManagerPassword: *`password`*

- Tomcat.ManagerURL: `http://localhost:8092/manager/text`
- Tomcat.ManagerUserid: `tomcat`
- Tomcat.StartScript: `/home/serena/AppServers/apache-tomcat-init/bin/startup.sh`
- Tomcat.Type: `Win`

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

Resource Groups | **Properties**

Name	Label	Required	Value
Tomcat.AppName	The application name	false	Qlarius_Online.war
Tomcat.Context	Tomcat Context	false	/Qlarius_Online.war
Tomcat.HostName	Tomcat Host Name	false	localhost
Tomcat.ListenPort	Tomcat Port	false	8092
Tomcat.ManagerPassword	Manager password for Tomcat	false	****
Tomcat.ManagerURL	Tomcat Manager URL	false	http://localhost:8092/manager/text
Tomcat.ManagerUserid	Tomcat Manager User ID	false	tomcat
Tomcat.StartScript	Tomcat Start Script	false	/home/serena/AppServers/apache-tomcat-int/bin/startup.sh
Tomcat.Type	Tomcat Type	false	Win

Deploying a Datical Database Component

The Datical Database Component process will create a Datical project, create a database step, and deploy the Change Log file. The process to deploy the Datical Database Component follows these steps:

1. Delete existing files and directories
2. Create a Datical project
3. Create the database definition
4. Create a snapshot of the database baseline and current state
5. Save the latest artifacts to the destination server
6. Forecast the impact of the database changes
7. Deploy the database changes
8. Create and publish a report about the database changes

This sample uses the following plugins:

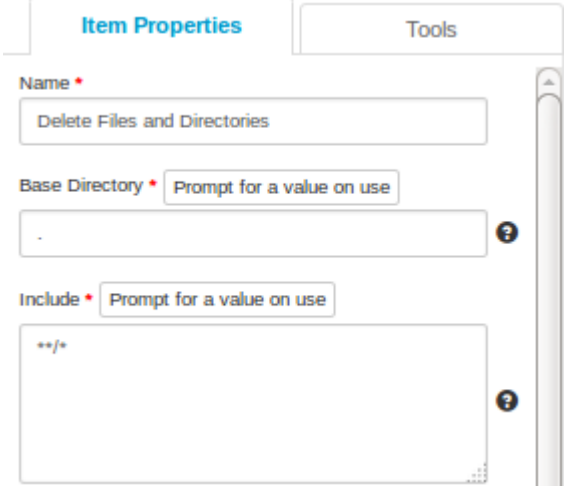
- Datical DB
- DA Versioned File Storage
- FileUtils

Designing a Process to Configure, Create, and Deploy a Datical Database Component

Step 1: Delete Files and Directories

The first step will delete the files from all subdirectories, starting at the base directory for the Sample Database Component (Datical) component. This is a best practice to ensure only the desired change logs are applied.

1. Navigate to **Management > Components**.
2. Select the **Sample Database Component (Datical)** component and then select the **Processes** tab.
3. Click the **Create Process** button.
4. Enter the name `DB Update` 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.

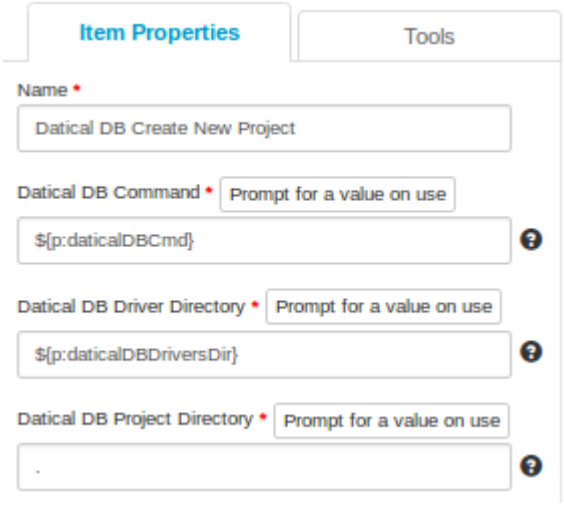


Step 2: Datical DB Create New Project

The second step will create a new Datical DB project.

1. On the **Tools** tab, expand the menu item **Datical DB**.
2. Drag the **Datical DB Create New Project** step onto the design space.

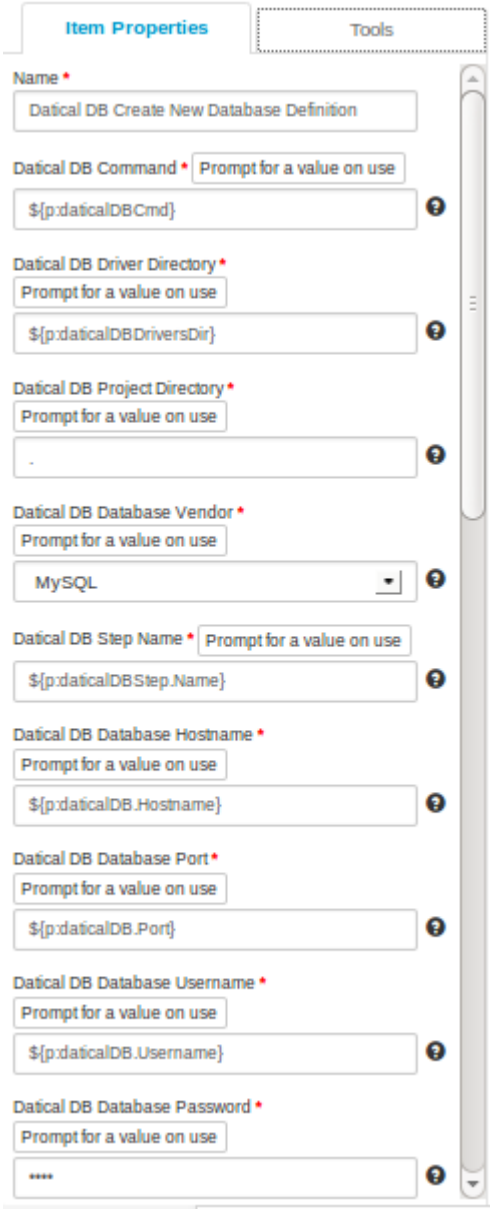
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Datical DB Command**: `${p:daticalDBCmd}`
   - **Datical DB Driver Directory**: `${p:daticalDBDriversDir}`
4. Use the default values for the remaining fields.



Step 3: Datical DB Create New Database Definition

The third step will create the database definition for the selected DB Type for the component.

1. On the **Tools** tab, expand the menu item **Datical DB**.
2. Drag the **Datical DB Create New Database Definition** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Datical DB Command**: `${p:daticalDBCmd}`
   - **Datical DB Driver Directory**: `${p:daticalDBDriversDir}`
   - **Datical DB Database Vendor**: `MySQL`
   - **Datical DB Step Name**: `${p:daticalDBStep.Name}`
   - **Datical DB Database Hostname**: `${p:daticalDB.Hostname}`
   - **Datical DB Database Port**: `${p:daticalDB.Port}`
   - **Datical DB Database Username**: `${p:daticalDB.Username}`
   - **Datical DB Database Password**: *`password`*
4. Use the default values for the remaining fields.

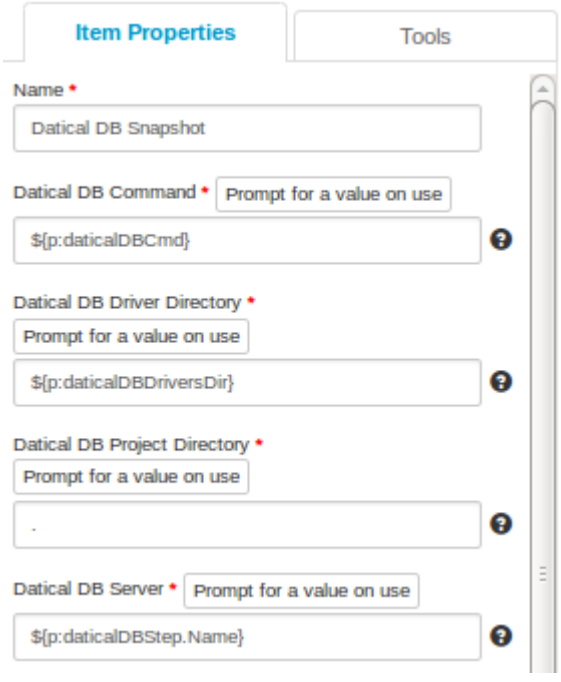


Step 4: Datical DB Snapshot

The fourth step will create a snapshot of the database baseline and current state.

1. On the **Tools** tab, expand the menu item **Datical DB**.
2. Drag the **Datical DB Snapshot** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Datical DB Command**: `${p:daticalDBCmd}`
   - **Datical DB Driver Directory**: `${p:daticalDBDriversDir}`
   - **Datical DB Server**: `${p:daticalDBStep.Name}`

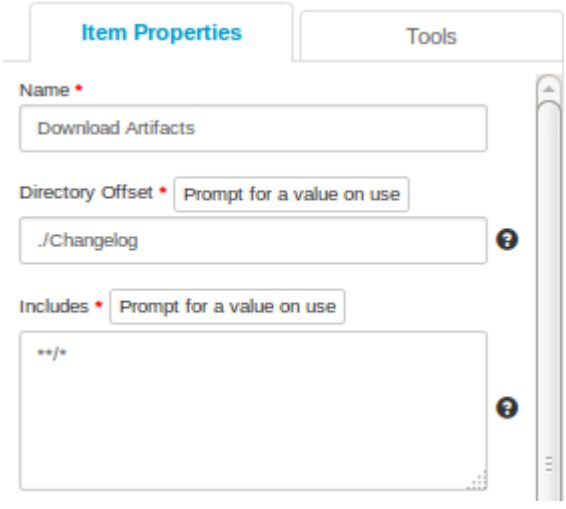
4. Use the default values for the remaining fields.



Step 5: Download Artifacts

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

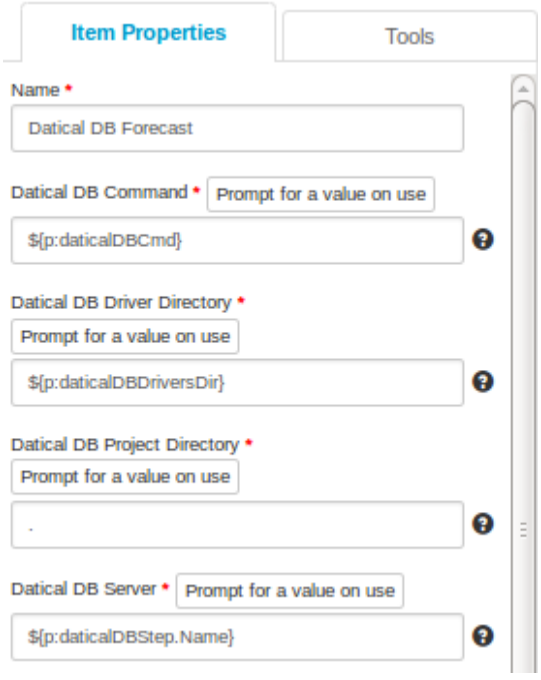
1. On the **Tools** tab, expand the menu item **Repositories > Artifact > DA**.
2. Drag the **Download Artifacts** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Directory Offset**: `./Changelog`
   - **Includes**: `**/*`
4. Use the default values for the remaining fields.



Step 6: Datical DB Forecast

The sixth step will perform a forecast for the database, which will proactively determine the impact of the database changes by simulating the proposed changes to the database model.

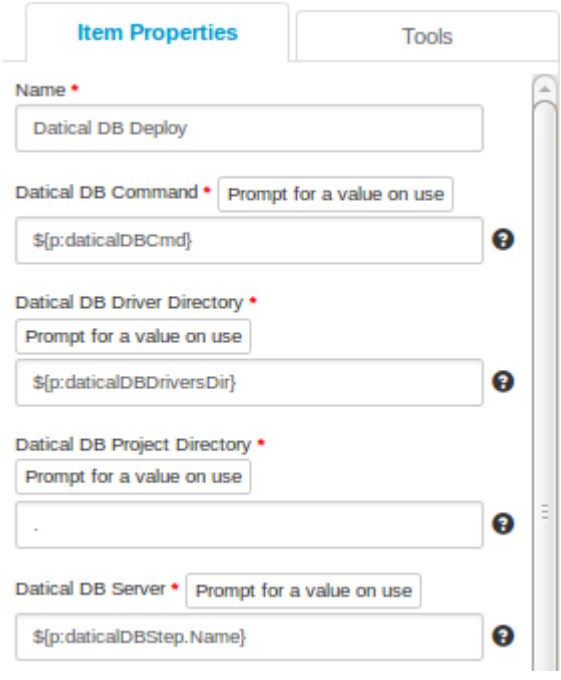
1. On the **Tools** tab, expand the menu item **Datical DB**.
2. Drag the **Datical DB Forecast** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Datical DB Command**: `${p:daticalDBCmd}`
   - **Datical DB Driver Directory**: `${p:daticalDBDriversDir}`
   - **Datical DB Server**: `{p:daticalDBStep.Name}`
4. Use the default values for the remaining fields.



Step 7: Datical DB Deploy

The seventh step will execute the changes to the database.

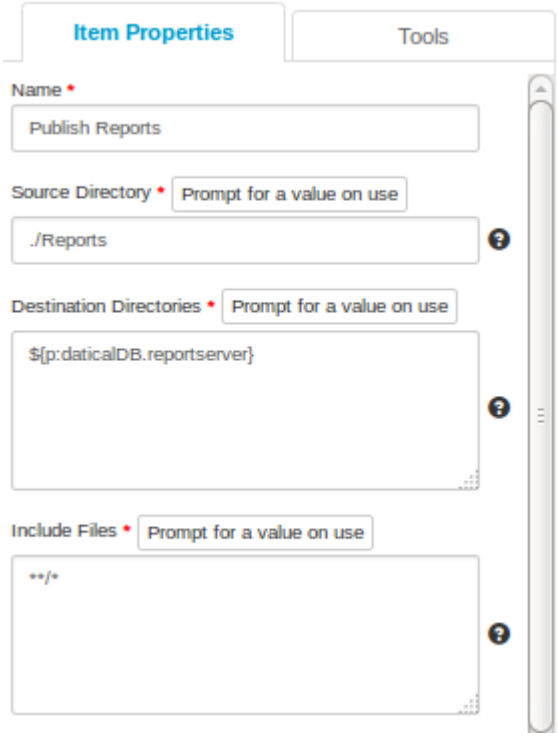
1. On the **Tools** tab, expand the menu item **Datical DB**.
2. Drag the **Datical DB Deploy** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Datical DB Command**: `${p:daticalDBCmd}`
   - **Datical DB Driver Directory**: `${p:daticalDBDriversDir}`
   - **Datical DB Server**: `{p:daticalDBStep.Name}`
4. Use the default values for the remaining fields.



Step 8: Publish Reports

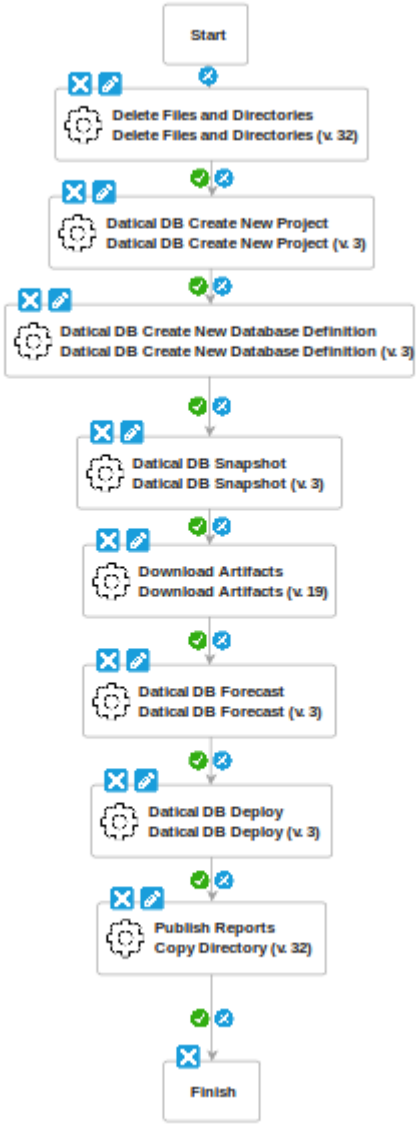
The eighth step will execute a copy of the reports to the appropriate directories.

1. On the **Tools** tab, expand the menu item **Utilities > FileUtils**.
2. Drag the **Copy Directory** step onto the design space.
3. On the **Item Properties** tab, enter these values for the following fields:
   - **Source Directory**: `./Reports`
   - **Destination Directories**: `${p:daticalDB.reportserver}`
   - **Include Files**: `**/*`
4. Use the default values for the remaining fields.



The Sample Database Component (Datical) 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 (Datical) 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 Sample Database Component (Datical) 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 Datical Database information.

1. Select the **Sample Database Component (Datical)** 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:
   - daticalDB.Hostname
   - daticalDB.Password
   - daticalDB.Port
   - daticalDB.reportserver
   - daticalDB.Username
   - daticalDBCmd
   - daticalDBDriversDir
   - daticalDBStep.Name

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 Tomcat 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 **Sample Database Component (Datical)** and then select the **Properties** tab.
6. Click **Edit**.
7. Enter values for the environment properties that you previously defined for the component:
   - daticalDB.Hostname: `localhost`
   - daticalDB.Password: *`password`*
   - daticalDB.Port: `3306`
   - daticalDB.reportserver: `/home/serena/DaticalReports`

- daticalDB.Username: `root`
- daticalDBCmd: `/usr/local/DaticalDB/repl/hammer`
- daticalDBDriversDir: `/home/serena/SampleArtifacts`
- daticalDBStep.Name: `PROD`

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

Resource Groups | **Properties**

Name	Label	Required	Value
daticalDB.Hostname	Datical DB Hostname	false	localhost
daticalDB.Password	daticalDB.Password	false	****
daticalDB.Port	daticalDB.Port	false	3306
daticalDB.reportserver	daticalDB.reportserver	false	/home/serena/DaticalReports
daticalDB.Username	daticalDB.Username	false	root
daticalDBCmd	daticalDBCmd	false	/usr/local/DaticalDB/repl/hammer
daticalDBDriversDir	daticalDBDriversDir	false	/home/serena/SampleArtifacts
daticalDBStep.Name	Datical DB Step Name	false	PROD

Designing an Application Process to Run the Deployment

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

For the Sample Tomcat Application, you will create an application process called **Deploy Sample Tomcat Application**.

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

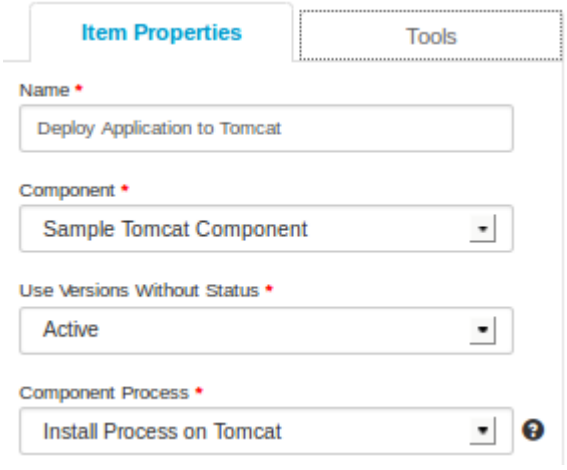
Next, design the application process, **Deploy Sample Tomcat Application**, as follows:

1. Execute the Sample Tomcat Component process.

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

   - **Name**: `Deploy Application to Tomcat`
   - **Component**: `Sample Tomcat Component`
   - **Component Process**: `Install Process on Tomcat`

2. Use the default values for the remaining fields.

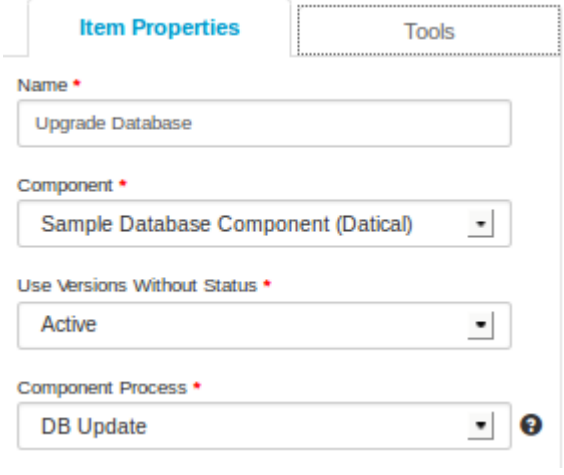


3. Execute the Sample Database Component (Datical) process.

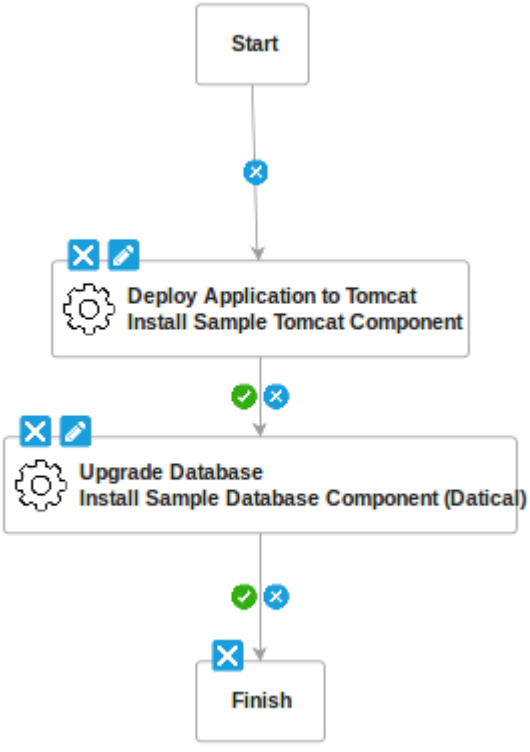
   Drag the **Install Component** step onto the design space and fill in the properties as follows:

   - **Name**: `Upgrade Database`
   - **Component**: `Sample Database Component (Datical)`
   - **Component Process**: `DB Update`

4. Use the default values for the remaining fields.



After selecting the preceding steps and filling in their properties, connect the steps as shown in the following graphical view of the application process, Deploy Sample Tomcat Application.

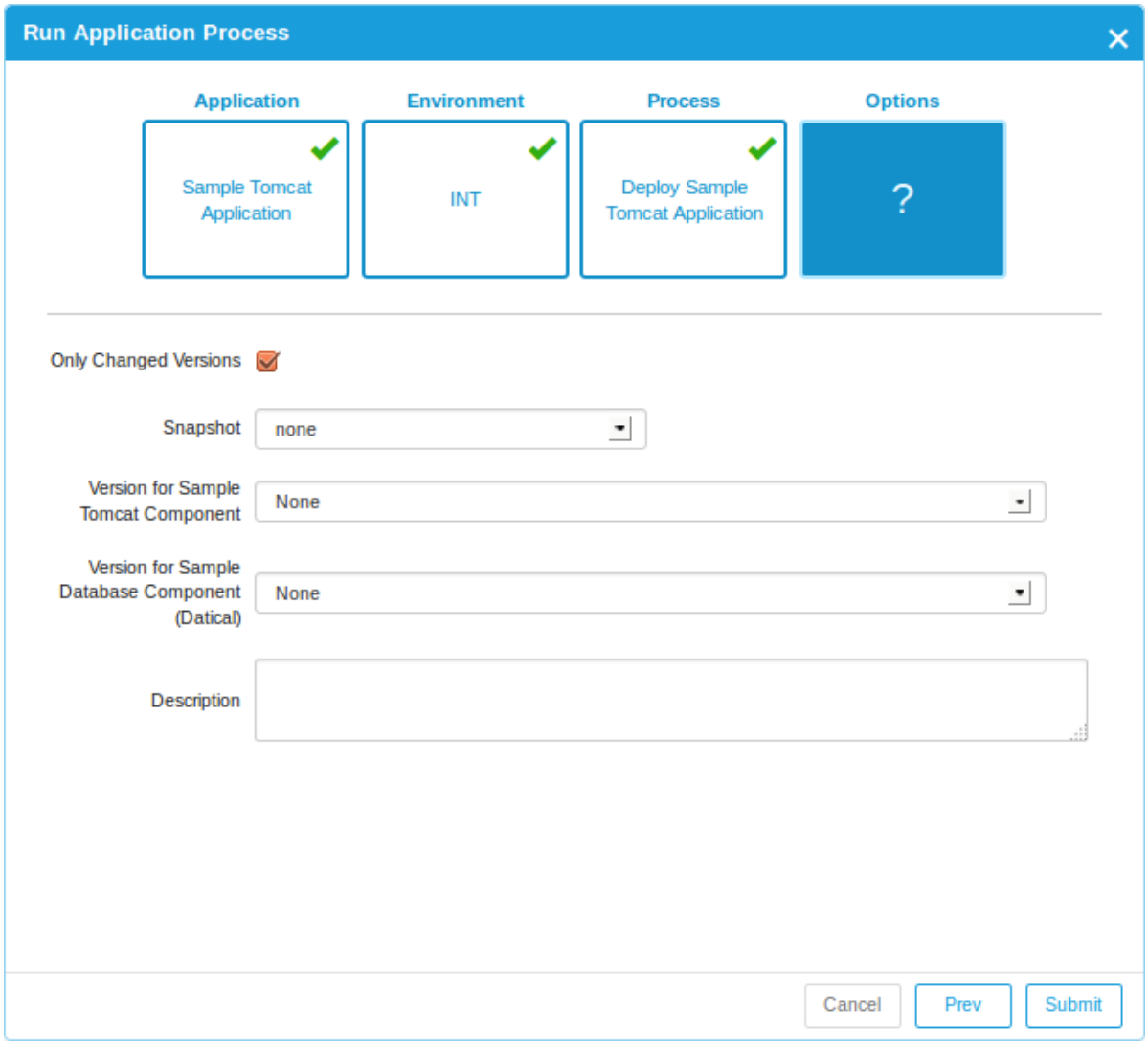


Deploying the Application

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

1. Select **Sample Tomcat Application** and then select the **Environments** tab.
2. Click the **INT** environment and then select **Run Process**.
3. Select **Deploy Sample Tomcat Application** and click **Next**.

   Note: If you have not installed Datical DB, choose **None** as the version for the Datical DB component. Otherwise, the deployment will fail when it cannot find the Datical application.
4. Look for and resolve any warnings at the bottom of the dialog box before submitting.
5. Click **Submit**.


Run Application Process
Application
Environment
Process
Options
Sample Tomcat Application
INT
Deploy Sample Tomcat Application
?
Only Changed Versions
Snapshot
none
Version for Sample Tomcat Component
None
Version for Sample Database Component (Datical)
None
Description
Cancel
Prev
Submit