

# Configure IBM Urban Code Deploy to manage Workload Automation objects

By Maria Elena Massino

In this article I will show you step-by-step how to configure IBM® UrbanCode™ Deploy with Workload Automation plug-in to deploy Workload Automation Job Streams to a distributed target environment.

## Contents

- Requirements..... 1
- Prerequisites ..... 1
- Create the input file for your deployment process ..... 2
- Create a component containing your Workload Automation Job Stream Definition ..... 3
- Create a component process to deploy your Workload Automation Job Stream Definition..... 5
- Create an application to manage components component process to deploy your Workload Automation Job Stream Definition ..... 11
- Configure an application environment ..... 12
- Configure the Environment Properties..... 15
- The application process ..... 15
- Deploy the Workload Automation component ..... 17
- Results..... 19
- Lesson checkpoint..... 19

## Requirements

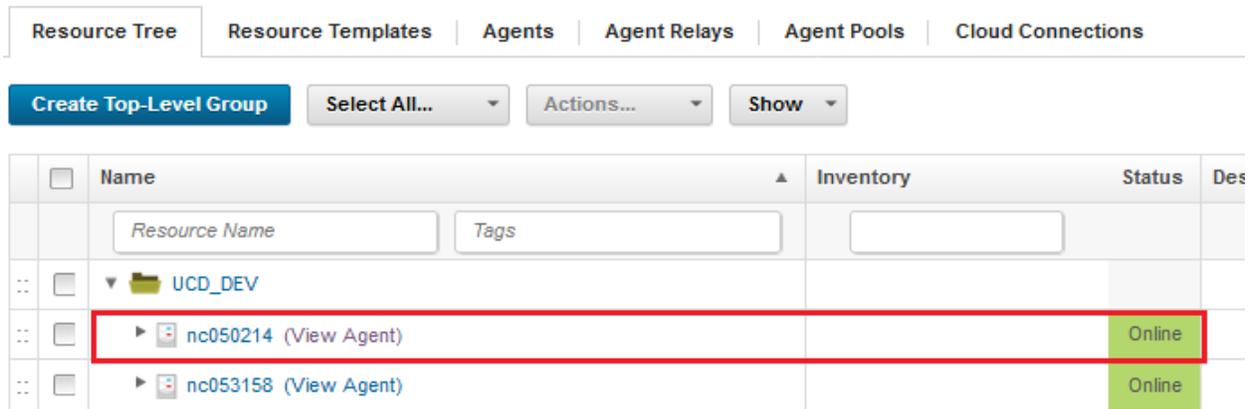
I'm using IBM UrbanCode Deploy 6.2.3 with Workload Automation plug-in installed and Workload Automation 9.4.0.1

## Prerequisites

The deploy is done invoking Workload Automation wappman command so you need to have an IBM UrbanCode Deploy agent installed on the target master or on a machine running a Workload Automation dynamic agent.

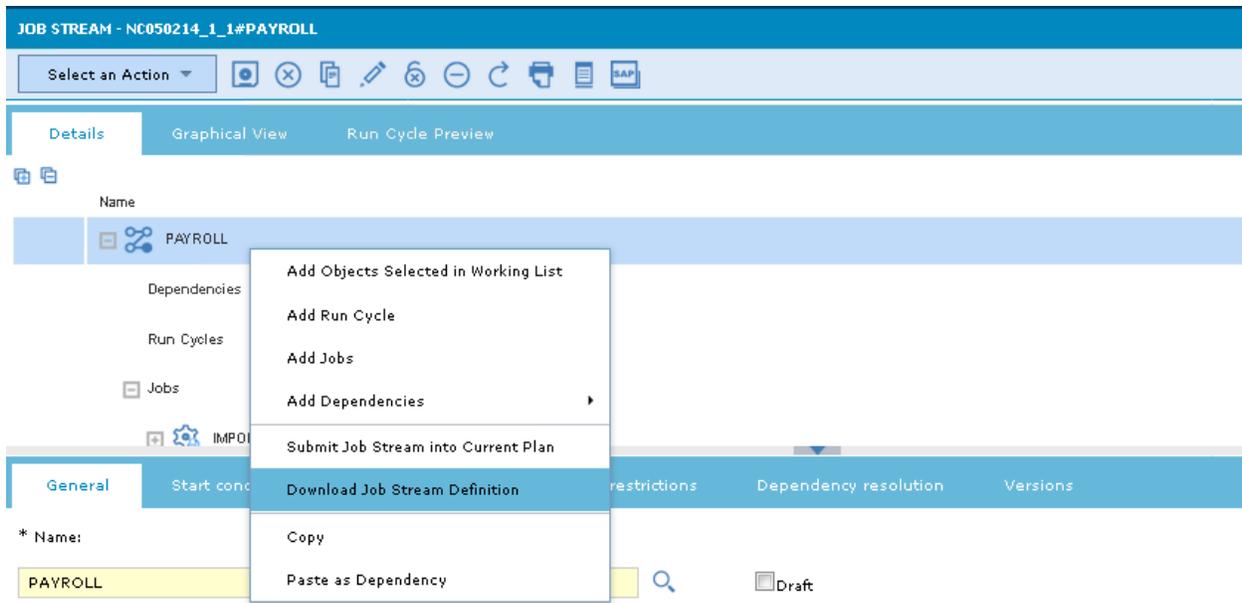
In this article I used an IBM UrbanCode Deploy agent running on the machine where the master is installed (nc050214). See [Installing Agents](#) for details on installing IBM UrbanCode Deploy agents.

Verify that your IBM UrbanCode Deploy agent is installed and running by clicking Resources > Agents. Your agent must be included in the list of agents with a status of **Online**, as shown in the following figure:



### Create the input file for your deployment process

A Workload Automation Job Stream can be downloaded from Workload Designer of Dynamic Workload Console. The zip file created from the "Download Job Stream Definition" action will be the input for our deployment process.



## Create a component containing your Workload Automation Job Stream Definition

In IBM UrbanCode Deployments a component is a container of *artifacts* and *processes*.

Artifacts include any kind of file that is associated with a software project, in our case it consist in the zip file created from Dynamic Workload Console connected to your source environment .

Processes define the activities that components can perform, in our case it will act on the zip files and run a wappman command on Workload Automation target environment to import the objects.

Artifacts are added to a component by connecting the IBM® UrbanCode™ Deploy server to a computer system that hosts the artifacts. The server can import artifacts from build systems, source-code management systems, and file systems. Imported artifacts are stored in CodeStation, the artifact repository.

For simplicity, the artifact for your component comes from the file system where you have installed the IBM UrbanCode Deploy agent.

### Create Workload Automation Component

1. On the machine where you have installed the IBM UrbanCode Deploy agent create a folder named **WorkloadAutomation** and inside it another folder called **1.0**. This will be the base folder for the versioning of your imports. Copy the zip file with the Workload Automation Job Stream Definition into the 1.0 folder
2. Click the Components tab and then click Create Component. In the window that opens, you define the component and specify the location of the artifacts for it.
3. In the Name field, type **Workload Automation**

- In the Source Configuration Type list, select File System (Versioned). This parameter defines the type of artifacts the component uses. All artifacts in a component share the source type. The File System (Versioned) type looks for the artifacts on the file system.
- In the Base Path field, specify the location of the **WorkloadAutomation** folder that you created earlier, such as C:\WorkloadAutomation
- Check the radio for Import new component versions using a single agent and select the IBM UrbanCode Deploy agent where the zip file has been copied
- Accept the default values for the other fields on the page. The Preserve Execute Permissions and Import Versions Automatically check boxes are cleared and the Copy to CodeStation check box is selected. The Default Version Type is set to Full.
- Click Save.

*Import the component version:*

- Click the Versions tab
- Click Import New Versions. The server creates a version of the component that is based on the folder in the WorkloadAutomation folder, and imports the zip file in the WorkloadAutomation/1.0 directory
- Verify that the list of versions includes version 1.0 of the component, as in the following figure:

Home > Components > Workload Automation

### Component: Workload Automation [\(show details\)](#)

Dashboard | Usage | Configuration | Calendar | **Versions** | Processes | Changes

**Import New Versions**

Version	Statuses	Type	Created By	Date	Description	Actions
<input type="text"/>	<input type="text" value="Statuses"/>	Any				
1.0		Full	UC Version Import	6/28/2017, 11:59 AM		<a href="#">Compare</a> <a href="#">Delete</a> <a href="#">Copy</a>

1 record - [Refresh](#) [Print](#)      << < 1 / 1 > >>      Rows 10

- Click the version name, 1.0.
- Verify that the list of artifacts includes the zip file in the WorkloadAutomation/1.0 directory, as in the following figure:

## Version: 1.0 [\(show details\)](#)

Main Configuration History

### Statuses

Add a Status

Status	Description	Created	By	Actions
--------	-------------	---------	----	---------

No statuses have been assigned to this version.

[Refresh](#)

### Artifacts

Total: 2.8 KB (1 files)

Download All

[Expand All](#) [Collapse All](#)

Name	Size	Last Modified	Actions
<input type="text"/>			
 PAYROLL.zip	2.8 KB	5/19/2017, 11:47 AM	<a href="#">Download</a>

[Refresh](#) [Print](#)

## Create a component process to deploy your Workload Automation Job Stream Definition

A component process is a succession of commands that are called steps. Steps can manipulate files, run system commands, set properties, pass information to other steps, and run programs. Steps are provided by automation plug-ins.

Processes are designed with the drag-and-drop process editor where you drag plug-in steps onto the design editor and configure them as you go. We will use three plug-ins, both of which come with the product: FileUtils and IBM UrbanCode Deploy and Workload Automation

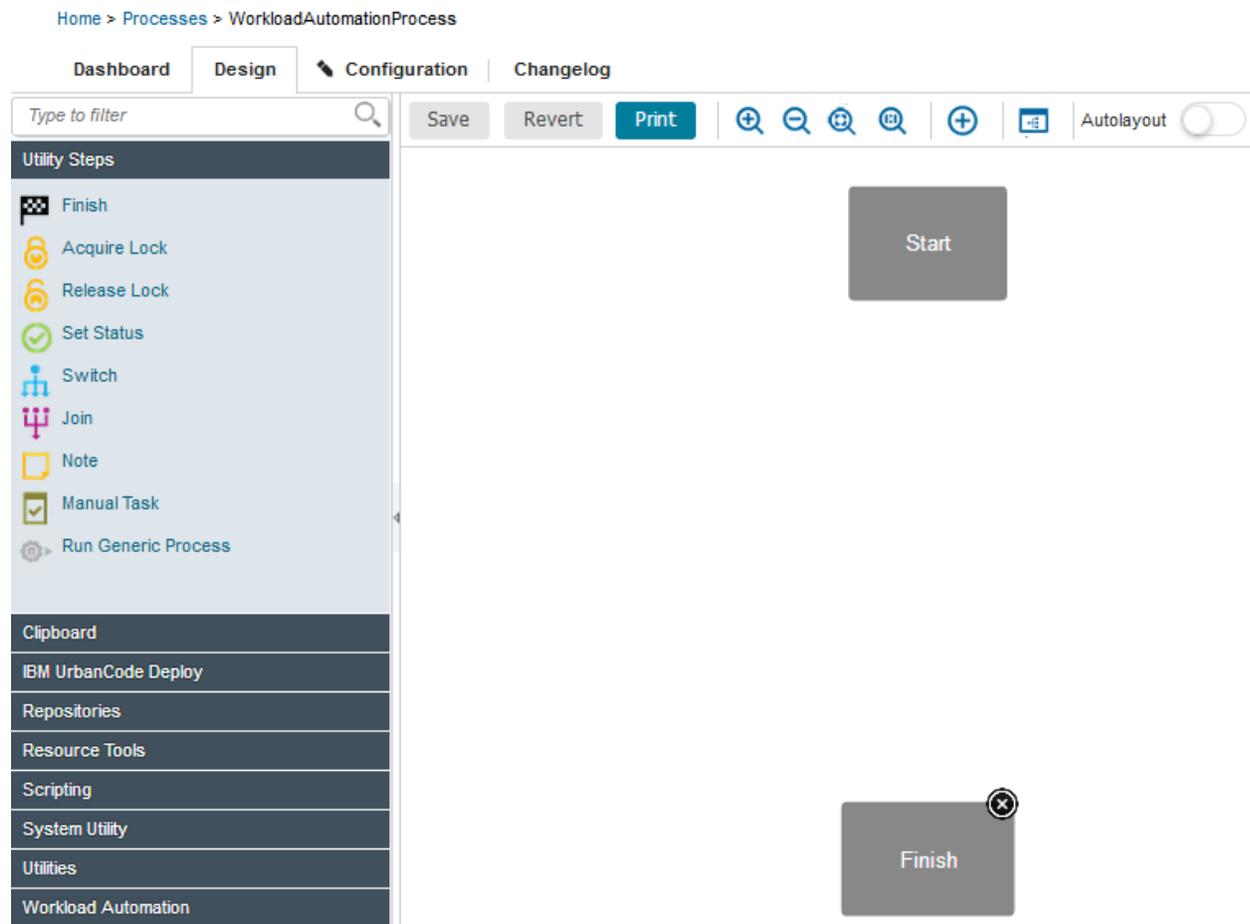
*Create the component process:*

1. On the "Component: WorkloadAutomation" page, click the Processes tab and then click Create Process.
2. In the Create Process window, type WorkloadAutomationProcess in the Name field.
3. In the Process Type list, select Deployment. This list has other options for processes that uninstall or configure components.
4. Accept the default values for the other fields and click Save. The Default Working Directory field becomes the folder that the agent uses to do its work, such as downloading artifacts and creating temporary files. In our case the value becomes `<agent_install>\var\work\WorkloadAutomation`.
5. Click Save

The process is listed on the Processes pane.

From the list of processes, select **WorkloadAutomationProcess**. The process opens in the process editor. The process editor lists plug-ins and steps. The required Start and Finish steps represent the beginning and the end of the process and are automatically placed on the design area. You add steps to

the process by dragging them onto the design area and arranging them between the Start and Finish steps.



Your deploy process will consist in a sequence of 4 steps:

1. Clean the IBM UrbanCode Deploy agent 's working directory. To ensure that the agent works with the most recent files, remove files that remain from previous work
2. Download the artifact from the component into the working directory of the IBM UrbanCode Deploy agent
3. Unzip the file
4. Import the Job Stream Definition into Workload Automation target environment

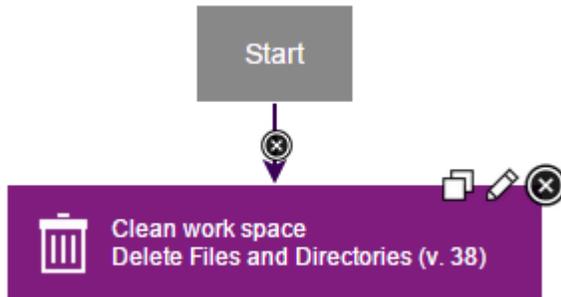
Step1: [clean the IBM UrbanCode Deploy agent 's working directory](#)

- a. In the Step Palette pane on the left, expand Utilities > FileUtils.
- b. Click and drag the Delete Files and Directories step to the process editor. Connect with Start and Finish steps if not done automatically as follows:

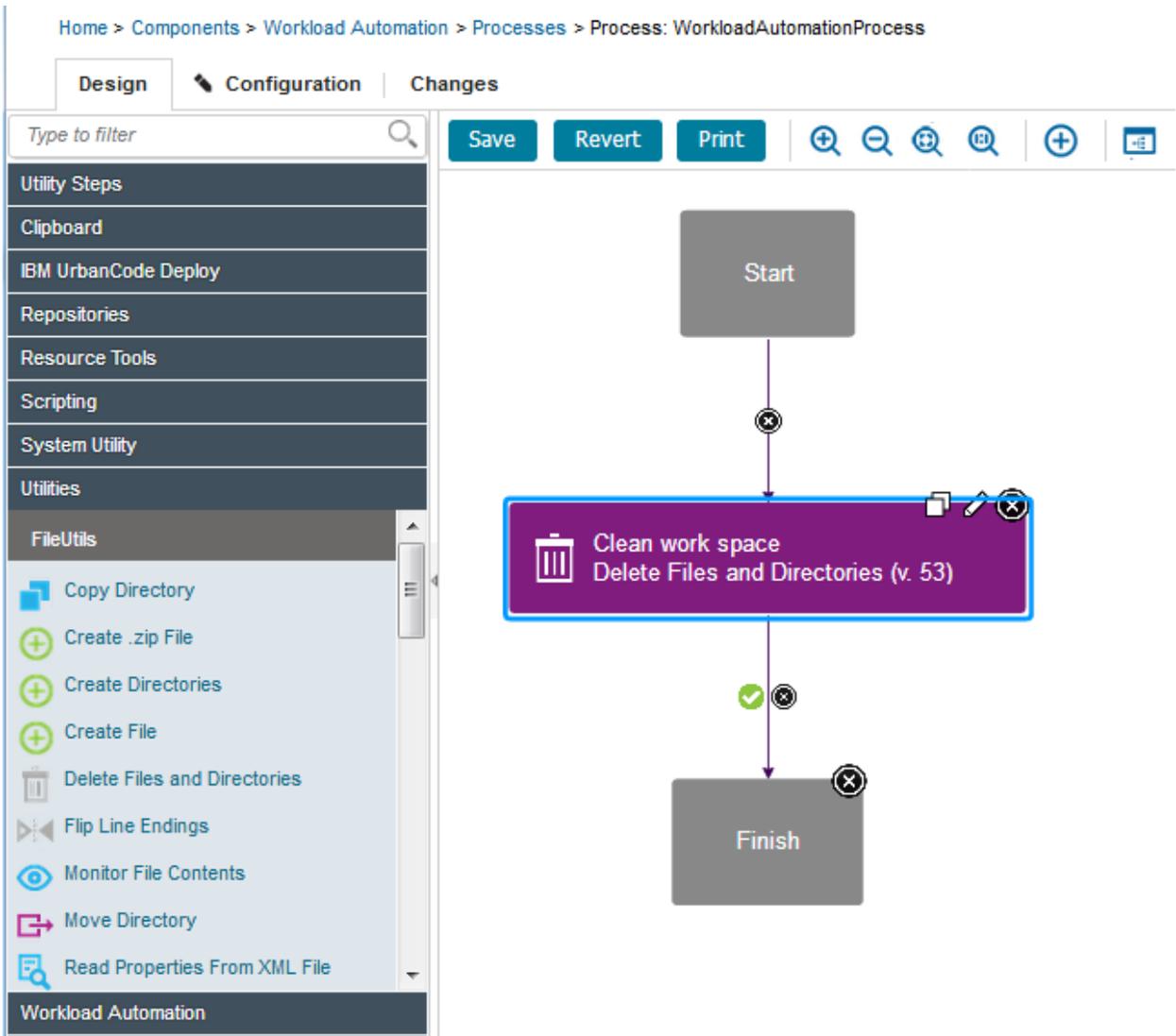
Move the mouse over Start. The arrow icon is displayed, as the following figure shows. You use this icon to connect steps to one another.



Click and drag the arrow icon over the Clean work space step. The connecting arrow links the two steps, as the following figure shows. The direction of the arrow defines the direction of the process flow.



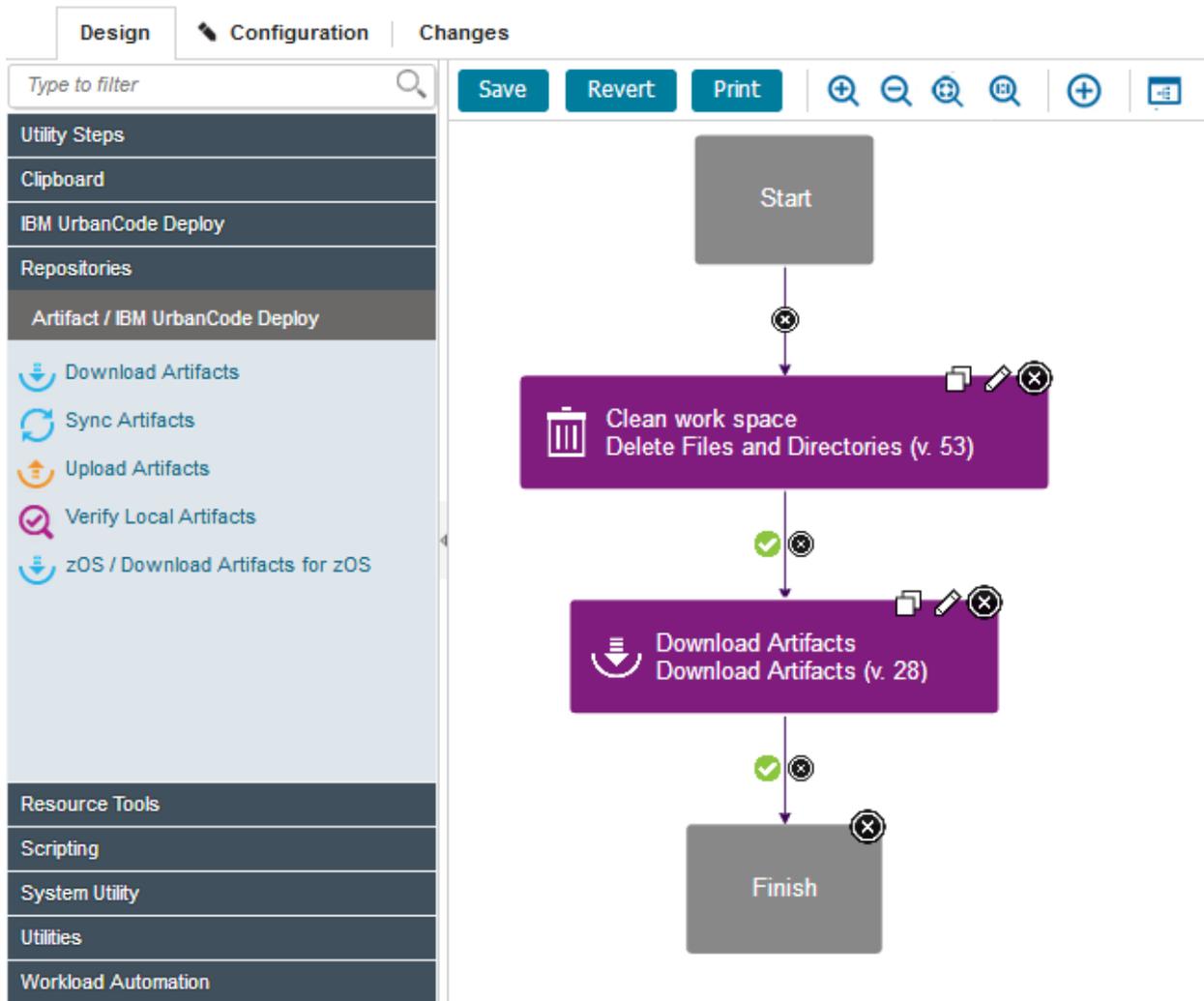
- c. Click the Edit icon  to open the Edit Properties window where you can modify the step properties.
- d. In the Name field, specify a name, for example, **Clean work space**.
- e. In the Base Directory field, specify a single period (.). This value resolves to the folder that you specified as the agent working directory.
- f. In the Include field, specify an asterisk (\*). This parameter instructs the agent to remove all files in the working directory.
- g. Accept the default values for the other properties and then click Save.



Step 2: download the artifact from the component into the working directory of the IBM UrbanCode Deploy agent

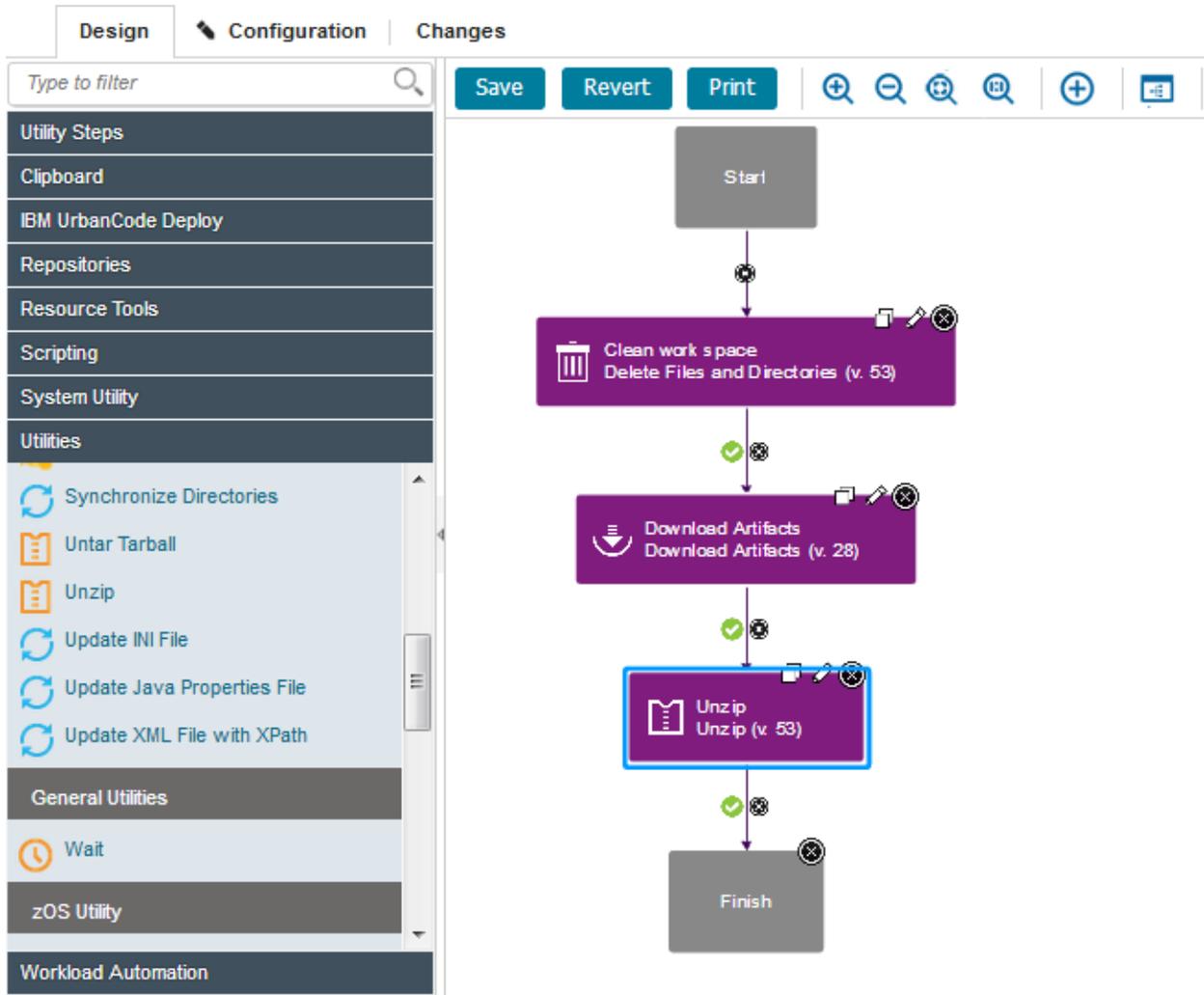
The Download Artifacts step is used in most deployment-type processes. This step downloads the specified version of the component artifacts to the agent's recently cleaned working directory. When you run the process, specify which version of the component artifacts to use.

- a. In the Step Palette expand Repositories > Artifact > IBM UrbanCode Deploy and drag the Download Artifacts step to the process editor between the Clean work space step and the Finish step (If the arrows are not automatically created please proceed as described above) .
- b. In the Edit Properties window, accept the default values and then click Save.



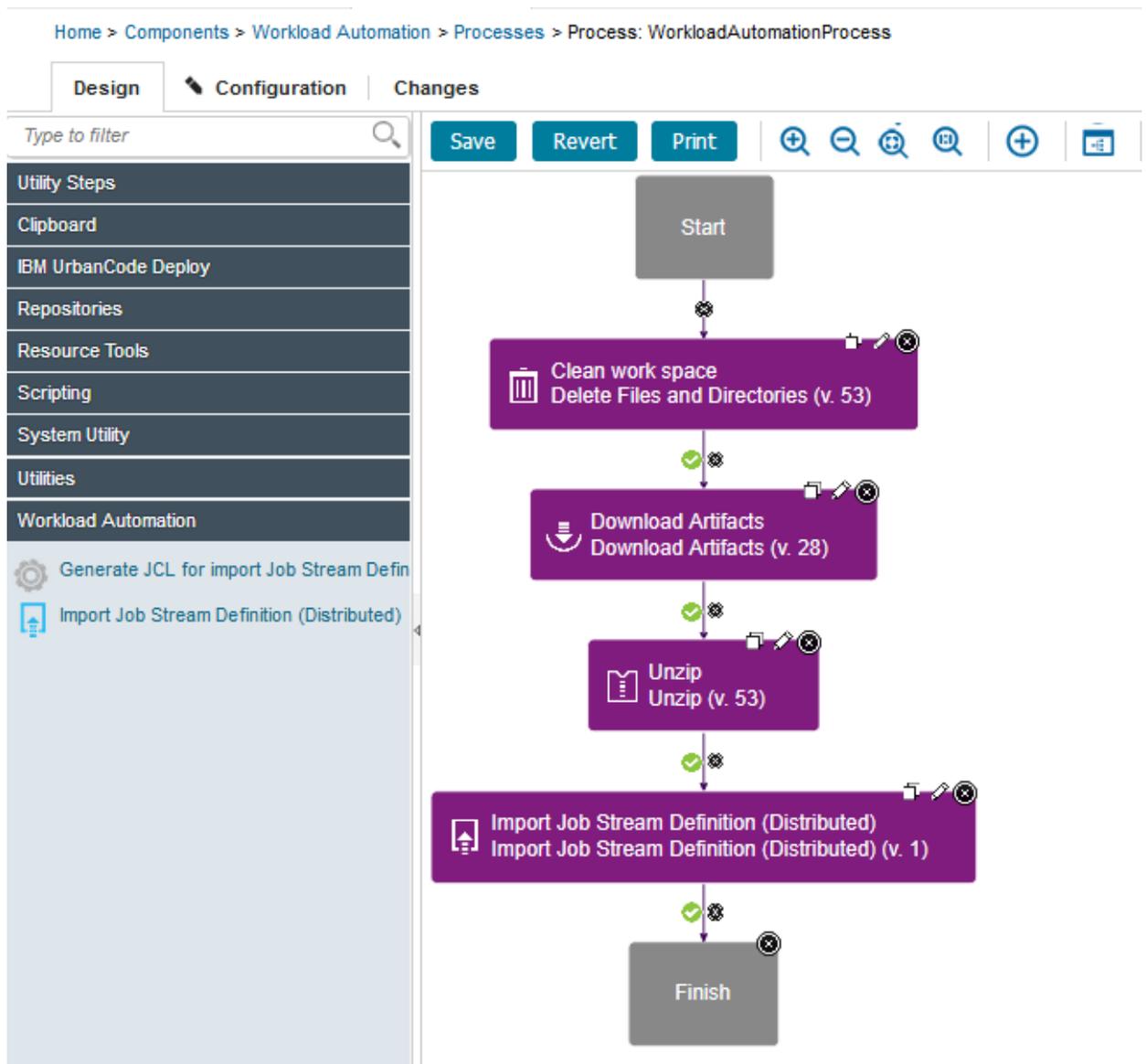
### Step 3: unzip the file

- In the Step Palette expand Utilities and drag the Unzip step to the process editor between the Download Artifacts step and the Finish step (If the arrows are not automatically created please proceed as described above) .
- In the Edit Properties window insert \*.zip in the .zip files field
- Accept the default values for other fields and then click Save.



Step 4: Import the Job Stream Definition into Workload Automation target environment

- a. In the Step Palette expand Workload Automation and drag the Import Job Stream Definition (Distributed) step to the process editor between the Unzip step and the Finish step (If the arrows are not automatically created please proceed as described above) .
- b. Open the Edit Properties window
- c. Insert in the Workload Automation Path the path of your Workload Automation installation: in my case is C:\IBM\TWA\TWS, default is /opt/IBM/TWA/TWS.
- d. Check the checkbox Replace Existing Definition: if checked the plug-in will try a wappman - replace command in case of object already imported
- e. Accept the default values for other fields and then click Save.



## Create an application to manage components component process to deploy your Workload Automation Job Stream Definition

Applications manage components, typically by deploying them into environments.

Your application will manage Workload Automation component; you need to define at least one environment into which the component is deployed; and create a process to do the work. An environment maps the component to agents and handles inventory, among other things. An application process is similar to but not identical to a component process. Application processes are primarily intended to direct underlying component processes and orchestrate multi-component deployments. In out case you create an application and assign the Workload Automation component to it.

### Create the Workload Automation application

1. Click the Applications tab and then click Create Application.

2. Name the new application something like **Workload Automation Application**.
3. Accept the default values for the other fields and click Save.

Add the Workload Automation component to the Workload Automation Application application:

1. From the Application: click on Workload Automation Application just created and then click the Components tab. This is the components tab associated with the application, not the Components tab at the top of the page.
2. Click Add Component.
3. In the Add a Component window, select the Workload Automation component and then click Save.

Home > Applications > Workload Automation Application

## Application: Workload Automation Application [\(show details\)](#)

Environments | History | Configuration | **Components** | Blueprints | Snapshots | Processes | Calendar

**Add Component**

0 Failed Version Import | 0 Importing Version | 0 No Version | 1 Successful

Name	Actions	Last Import	Last Version
Workload Automation	<a href="#">Run Process</a> <a href="#">Remove</a>	Successful	1.0

1 record - [Refresh](#) [Print](#)

1 / 1

## Configure an application environment

An environment is a user-defined collection of resources that identify the components that can be deployed by the parent application, along with the agents that do the work.

### Define the environment

- a. In the Application: Workload Automation Application page, click Environments.
- b. Click Create Environment.
- c. Specify the name to be something like Workload Automation deploy
- d. Accept the default values in the other fields in this window and click Save.

## Application: Workload Automation Application [\(show details\)](#)

Environments | History | Configuration | Components | Blueprints | Snapshots | Processes | Calendar | Changes

Create Environment

Drag environments by their names to re-order them. 1 Environment

Search by Name

or

Search by Blueprint

Expand All

Collapse All



Workload Automation deploy

Snapshot:  
None

Compliance: 0 / 0

### Add resource to the environment

- e. Click the environment name.
- f. Using the Resources tab for the environment, click Add Base Resources. The Add Resource to Environment window shows all available resources.
- g. Select the check box next to resource group containing the agent running on the machine where the Workload Automation is installed and then click OK. When you select the resource group, you automatically select the resources associated with it, such as the agent you assigned to the group. Selecting an agent-type resource identifies the location, usually a computer, where deployments can occur.

## Environment: Workload Automation deploy for Workload Automation Application

Resources | History | Calendar | Configuration | Changes

No Desired Inventory

Add Base Resources

Select All...

Actions...

Show

Expand All Collapse All

<input type="checkbox"/>	Name	Inventory	Status	Description
	<input type="text" value="Resource Name"/> <input type="text" value="Tags"/>	<input type="text"/>		<input type="text"/>
	<ul style="list-style-type: none"> <li>▼ UCD_DEV                             <ul style="list-style-type: none"> <li>▶ <input type="checkbox"/> nc050214 (View Agent) <span style="float: right;">Online</span></li> <li>▶ <input type="checkbox"/> nc053158 (View Agent) <span style="float: right;">Online</span></li> <li>▶ <input type="checkbox"/> TVT5012.svl.ibm.com (View Agent) <span style="float: right;">Online</span></li> </ul> </li> </ul>			

[Refresh](#) [Print](#)

### Map the Workload Automation component to this agent resource

- h. As you hover the mouse over the row with the agent resource, click Actions > Add Component.

## Environment: Workload Automation deploy for Workload Automation Application

Resources | History | Calendar | Configuration | Changes

No Desired Inventory

Add Base Resources

Select All...

Actions...

Show

Expand All Collapse All

<input type="checkbox"/>	Name	Inventory	Status	Description
	<input type="text" value="Resource Name"/> <input type="text" value="Tags"/>	<input type="text"/>		<input type="text"/>
	▼ UCD_DEV			
⋮ <input type="checkbox"/>	▶  nc050214 (View Agent)		Online	
⋮ <input type="checkbox"/>	▶  nc053158 (View Agent)		Online	
⋮ <input type="checkbox"/>	▶  TVT5012.svl.ibm.com (View Agent)		Online	

Refresh Print

- Compare or Synchronize
- Define New Template
- Synchronize With Template
- Add From Template
- Add Group
- Add Component**
- Add Component Tag
- Delete

- Select the Workload Automation component and then click Save

## Environment: Workload Automation deploy for Workload Automation Application

Resources | History | Calendar | Configuration | Changes

No Desired Inventory

Add Base Resources | Select All... | Actions... | Show | Expand All | Collapse

<input type="checkbox"/>	Name	Inventory	Status	Description
	<input type="text" value="Resource Name"/> <input type="text" value="Tags"/>	<input type="text"/>		<input type="text"/>
	▼ UCD_DEV			
<input type="checkbox"/>	▼ nc050214 (View Agent) <input type="button" value="Actions..."/>		Online	
<input type="checkbox"/>	helloWorld (View Component) <input type="button" value="blueCycle x"/>	1.0		
<input type="checkbox"/>	My_Application (View Component)	2.0		
<input type="checkbox"/>	RTC_Component (View Component)	7.0		
<input type="checkbox"/>	WAT_Test (View Component)	3.0		
<input type="checkbox"/>	Workload Automation (View Component) <input type="button" value="Actions..."/>			

### Configure the Environment Properties

#### The application process

An application process, like a component process, consists of steps that are configured with the process editor. In our case the application process will install the Workload Automation component by calling the component process that you created earlier but it could handle multiple components. You must use application processes to deploy components

#### Create the application process

1. Click the Applications tab and then click Workload Automation Application.
2. Click Processes and then click Create Process.
3. In the Create an Application Process window, name the new application process something like **Workload Automation Application Process**.
4. Accept the default values for the other fields and click Save.

## Application: Workload Automation Application [\(show details\)](#)

Environments | History | Configuration | Components | Blueprints | Snapshots | **Processes** | Calendar | Changes

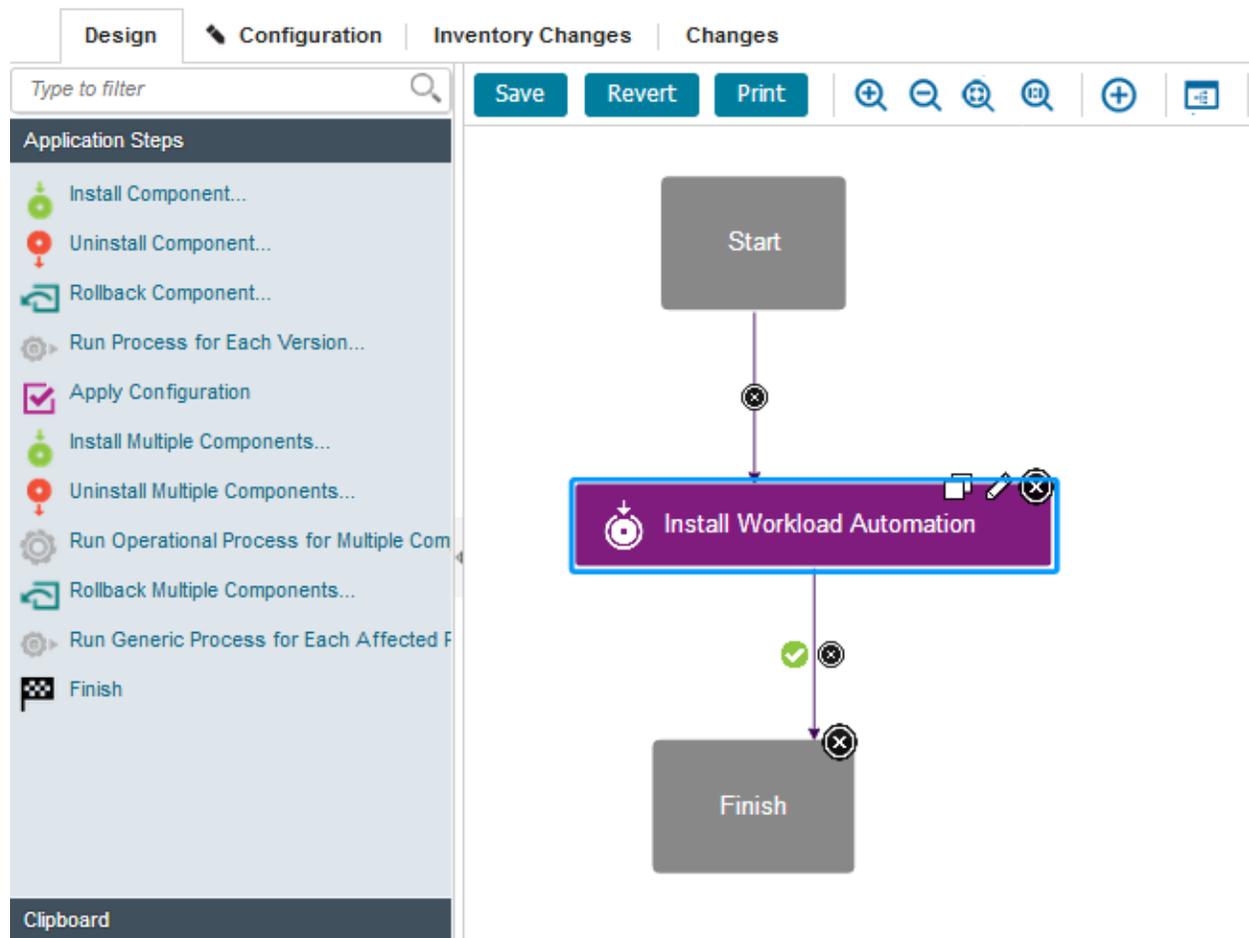
Create Process

Process	Description	Actions
<a href="#">Workload Automation Application Process</a>		<a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>

1 record - [Refresh](#) [Print](#)      << < 1 / 1 > >>      Rows 10

### Add steps to the application process

5. Click on **Workload Automation Application Process** to open the new process in the process editor.
6. Add a step that deploys the **Workload Automation** component:
  - a. From the list of steps, add an Install Component step to the design area.
  - b. In the Edit Properties window, name the step something like **Install Workload Automation**.**
  - c. All the fields should be correctly prefilled anyway assure that in the Component list you have selected the **Workload Automation component**. All components that are associated with an application are available.
  - d. In the Component Process list assure that you have selected the **WorkloadAutomationProcess** component process. If multiple processes were defined for the Workload Automation component, they are listed.
  - e. Accept the default values for the other fields and click OK.
7. Connect the Start step to the Install Workload Automation step.
8. Connect the Install Workload Automation step to the Finish step.
9. Save the process by clicking the Save icon



## Deploy the Workload Automation component

The final step to have the object defined on the target environment is to run the application process on your environment.

1. Open the application page by clicking Applications and then clicking the application name.



2. In the same row as your environment, click the Request Process icon.
3. In the Run Process window, accept the default value for the Only Changed Versions parameter. If this check box is selected, no previously deployed component versions are deployed.
4. In the Process list, select the hello App Process process.
5. Click Choose Versions. The Component Versions window opens.
6. In the Component Versions window, click Select For All, and then select Latest Available. Click OK to return to the Run Process window.
7. Click Submit. The Application Process Request page shows the progress of the request. From here, you can watch as the process runs. The following figure shows the running process.

<span>Log</span>   <span>Properties</span>   <span>Manifest</span>   <span>Configuration Changes</span>   <span>Inventory Changes</span>			
<b>Execution</b>			
<a href="#">Pause</a> <a href="#">Cancel</a>			
Step		Progress	Start Time
▼ 1.  Deploy helloWorld		0 / 1	9:07:25 AM
▶  helloWorld		0 / 1	9:07:25 AM
Total Execution		0 / 1	9:07:25 AM

If the process finishes, the Success status is shown, as in the following figure:

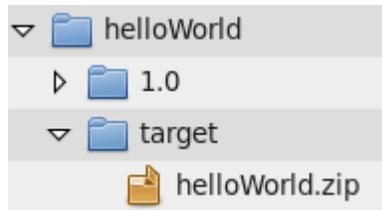
<span>Log</span>   <span>Properties</span>   <span>Manifest</span>   <span>Configuration Changes</span>   <span>Inventory Changes</span>			
▼ <b>Execution</b>			
Start	Progress	Status	
9:07:25 AM	1 / 1	 Success	
	—————		
	—————		
	—————		
Step		Progress	Start Time
▼ 1.  Deploy helloWorld		1 / 1	9:07:25 AM
▶  helloWorld		1 / 1	9:07:25 AM
Total Execution		1 / 1	9:07:25 AM

If the application process execution is not successful, click **Expand All** and in the same row as your application process, click **View Child Execution**. **View Child Execution** does not appear until you hover the mouse over the process. The **Deployment of Component** page shows the log for each step in the component process. From here, you can look at the output log for each step by clicking the **Output Log** icon .

- Open the target directory to confirm that the component artifacts are deployed.

## Results

The `helloWorld.zip` file is moved to the location that you specified in the `helloHome` property, as shown in the following figure:



## Lesson checkpoint

The application process installed the `helloWorld` component into the target environment.

Application processes can also uninstall or update components or run other configuration.

For more information about running deployments, see