

IBM WebSphere® Process Integration V6.0 – Lab Exercise

ManualClean – Human Tasks

What this exercise is about	1
Lab Requirements	1
What you should be able to do	1
Introduction	2
Exercise Instructions	3
Part 1: Initialize the Workspace for this Lab Exercise.....	5
Part 2: Create Manual Cleanse.....	9
Part 3: Prepare for Unit Testing and Assemble Manual Cleanse	16
Part 4: Test Manual Cleanse	19
What you did in this exercise	26
Solution Instructions.....	27
Task: Adding Remote Server to WebSphere Integration Developer Test Environment.....	28

What this exercise is about

In many situations focused on the automation of integration between different technologies, there is still a need for human interaction. WebSphere Integration Developer provides developers with the ability to easily add human interaction into integration applications. Utilizing the Service Component Architecture (SCA) programming model, human tasks can be defined as services (SCA components) and then flexibly wired into integration applications. This lab exercise will show the basic steps for defining a human task as a service component and testing the service using the End-to-End Test Framework.

Lab Requirements

List of system and software required for the student to complete the lab.

- WebSphere Integration Developer V6.0.1 installed
- WebSphere Process Server V6.0 test environment installed
- Sample code in the directory C:\Labfiles60 (Windows®) or /tmp/LabFiles60 (Linux®)

What you should be able to do

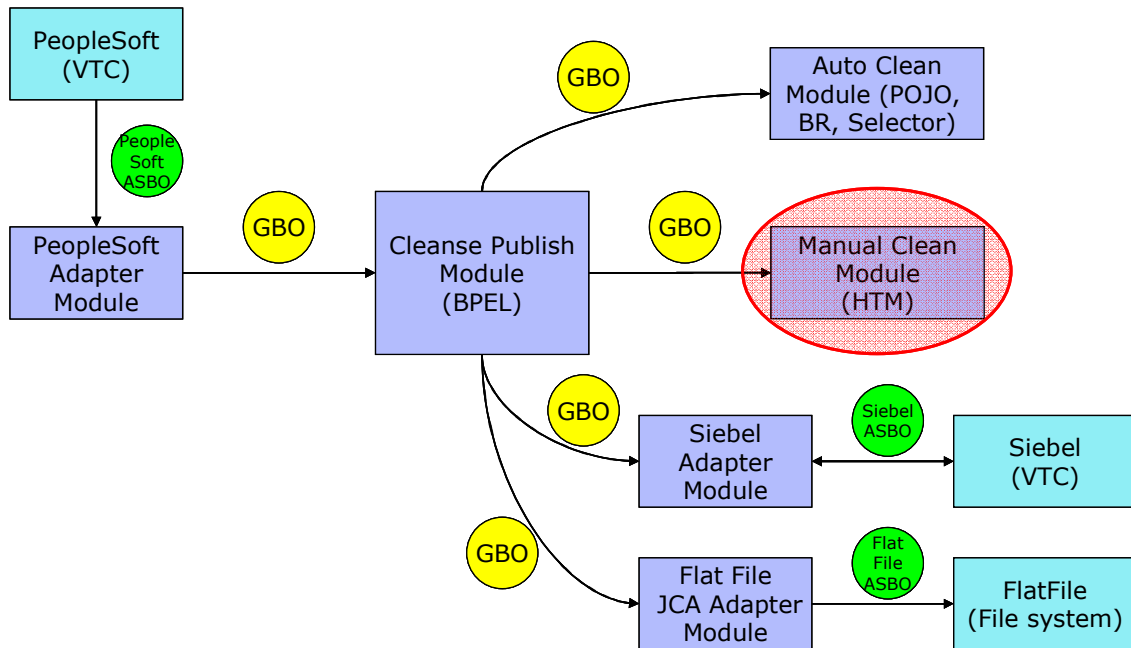
At the end of this lab you should be able to:

- Create a Participating Task

- Assemble the Participating Task following the SCA programming model
- Component test the Participating Task using the End-to-End Test Framework

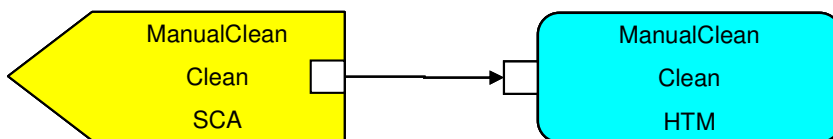
Introduction

The following diagram highlights the part of the overall scenario that will be addressed in this lab. You will create the Manual Clean human task which utilizes the Clean interface. The human task can then be assembled as an SCA Component. The SCA component can then be easily wired as a step in the Cleanse Publish business process. At this step, the human task allows for the general business object (GBO) which was passed into the business process from the PeopleSoft Adapter to be manually updated or cleaned after an automated clean has occurred from a Business Rule, Selector, or Java™ object. After the GBO is manually cleaned and the task is completed, the business process can continue and publish the GBO to the Siebel and Flat File adapters. Because the human task is wired in as an SCA component, at any time the implementation can be changed to any type that matches the clean interface. This provides flexibility to the integration application as the business scenario changes.



Description of the Module

The following diagram shows the ManualClean module. It highlights the parts of the module that will be addressed in this exercise.



Business Objects

The following generic business object (GBOs) will be used in this exercise.

- **Clip** – The parent business object describing characteristics of a particular clip. This generic business object originates from the application specific business object (ASBO) from the PeopleSoft system. In the complete solution, a map and mediation is used to transform the application specific business object from the PeopleSoft system to the generic business object which the Manual Clean human task can operate upon.
- **ClipItem** - The child business object describing a retail packing option for this type of clip. This generic business object is also defined from the application specific business object (ASBO) from the PeopleSoft system and transformed to the generic business object using a map and mediation.

In addition to these business objects, there will be a business graph generated to contain these business objects.

- **ClipBG**

Although this business graph is generated by the tools, it is only associated with the parent business object. The business graph for the child business object is not used in the scenario.

Interfaces

The following interface is generated by WebSphere Integration Developer and uses the business graph and business objects.

- **Clean** – Request/Response interface which passes a single part, named inClipBG of type ClipBG, on the request and a single part on the response, named outClipBG of type ClipBG. With the use of this interface, it is expected that any changes made to the ClipBG will be passed directly back. This interface will be used to define the ManualClean human task and will be part of the exported SCA binding which is called by other SCA components.
-

Exercise Instructions

Some instructions in this lab might be specific for Windows platforms. If you run the lab on a platform other than Windows, you will need to run the appropriate commands, and use appropriate files (for example .sh in place of .bat) for your operating system. The directory locations are specified in the lab instructions using symbolic references as follows:

Reference Variable	Windows Location	Linux Location
<WID_HOME>	C:\Program Files\IBM\WebSphere\ID\6.0	/opt/IBM/WebSphere/ID/6.0
<WPS_HOME>	<WID_HOME>\runtimes\bi_v6	<WID_HOME>/runtimes/bi_v6
<LAB_FILES>	C:\Labfiles60	/tmp/Labfiles60
<WORKSPACE>	C:\Labfiles60\eXchange\ManualClean\workspace	/tmp/Labfiles60/eXchange/ManualClean/workspace
<TEMP>	C:\temp	/tmp
<SOLUTION>	C:\Labfiles60\eXchange\ManualClean\Solution	/tmp/Labfiles60/eXchange/ManualClean/Solution

Windows users note: When directory locations are passed as parameters to a Java program such as wsadmin, you must replace the backslashes with forward slashes to follow the Java convention. For example, C:\LabFiles60\ would be replaced by C:/LabFiles60/.

Note that the previous table is relative to where you are running WebSphere Integration Developer. The following table is related to where you are running remote test environment:

Reference Variable	Example: Remote Windows test server location	Example: Remote z/OS [®] test server location	Input your values for the remote location of the test server
<SERVER_NAME>	server1	cl1sr01	
<WAS_HOME>	C:\Program Files\IBM\WebSphere\AppServer	/etc/cl1cell/AppServerNode1	
<HOSTNAME>	localhost	mvsxxx.rtp.raleigh.ibm.com	
<BOOTSTRAP_PORT>	2809	2809	
<TELNET_PORT>	N/A	1023	
<PROFILE_NAME>	AppSrv01	default	
<USERID>	N/A	cl1admin	
<PASSWORD>	N/A	fr1day	

Instructions for using a remote testing environment, such as z/OS, AIX[®] or Solaris, can be found at the end of this document, in the section "[Task: Adding Remote Server to WebSphere Integration Developer Test Environment](#)".

Part 1: Initialize the Workspace for this Lab Exercise

___ 1. Follow the directions below to initialize the Workspace using the following values:

<WORKSPACE>

C:\Labfiles60\exchange\ManualClean\workspace

<PROJECT_INTERCHANGE>

C:\Labfiles60\exchange\CleansePublishLibrary\import\CleansePublishLibrary_P1.zip

<MODULE>

ManualClean

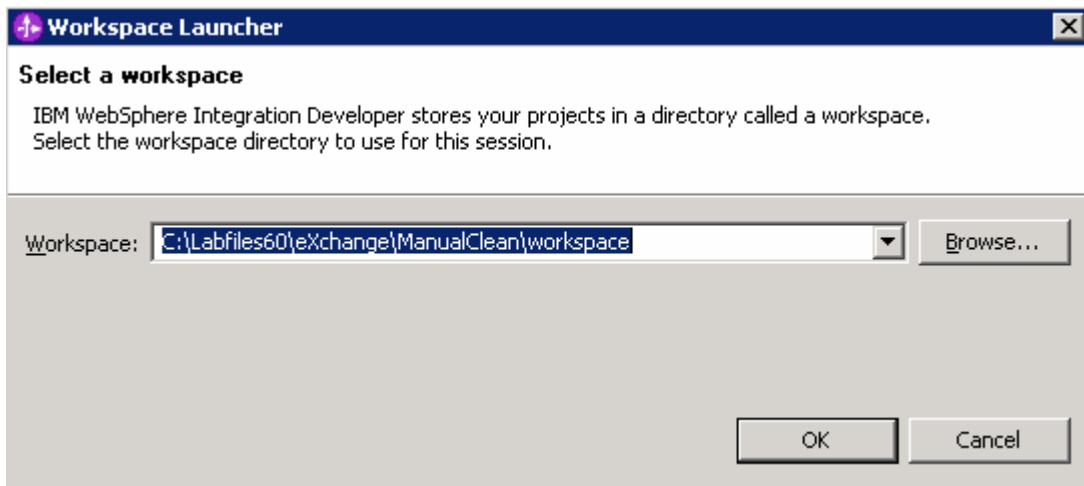
<DEPENDENT_LIBRARIES>

CleansePublishLibrary

___ 2. Start WebSphere Integration Developer V6.0.1 with a new workspace located at **<WORKSPACE>**.

___ a. From Windows Explorer, navigate to the **<WID_HOME>** directory and double click on wid.exe.

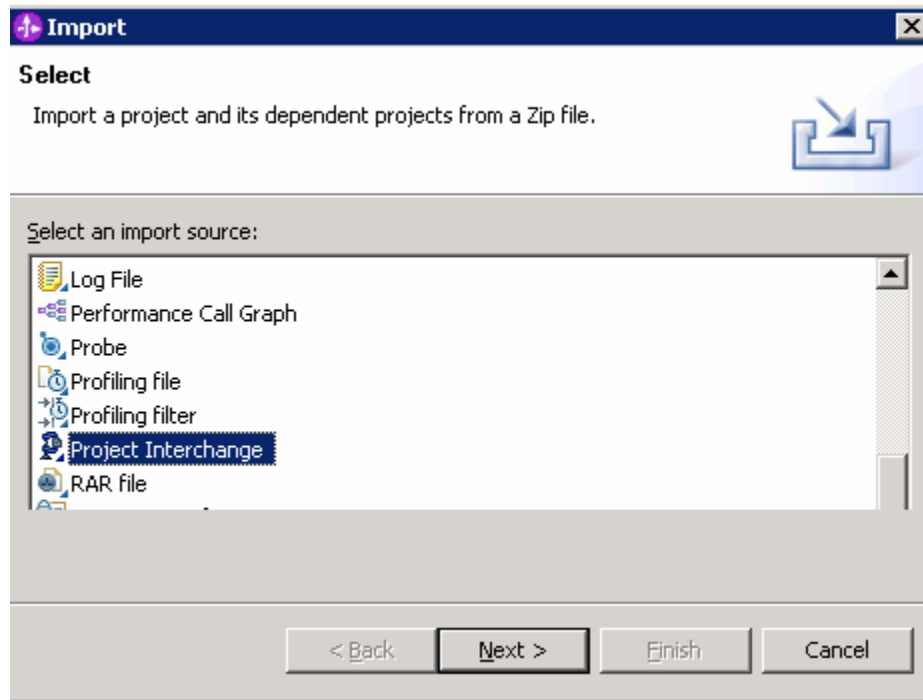
___ b. When prompted for workspace name, enter the value provided by the **<WORKSPACE>** variable for this lab and click **OK**.



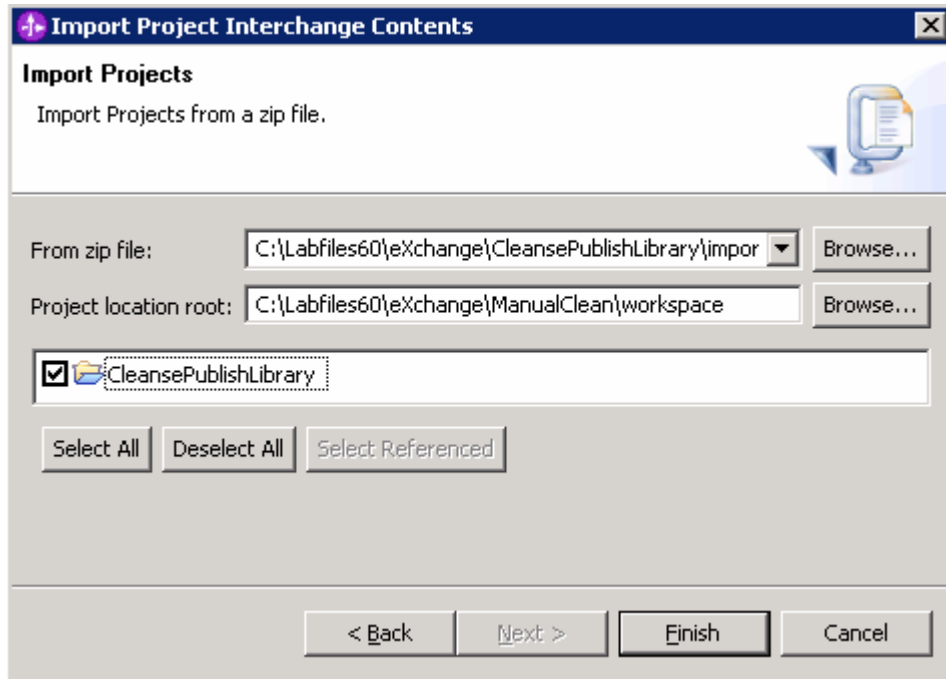
- ___ c. When WebSphere Integration Developer V6.0 opens, close the **Welcome page** by clicking on the Go to the workbench icon (bent over arrow at top-right).



- ___ d. Ensure you are in the **Business Integration** perspective.
- ___ 3. If this lab requires you to import a project interchange file, setup the required libraries and modules for this lab by importing the project interchange file **<PROJECT_INTERCHANGE>**.
 - ___ a. From the menu bar, select **File -> Import...**
 - ___ b. In the Import dialog, scroll down and select **Project Interchange**.



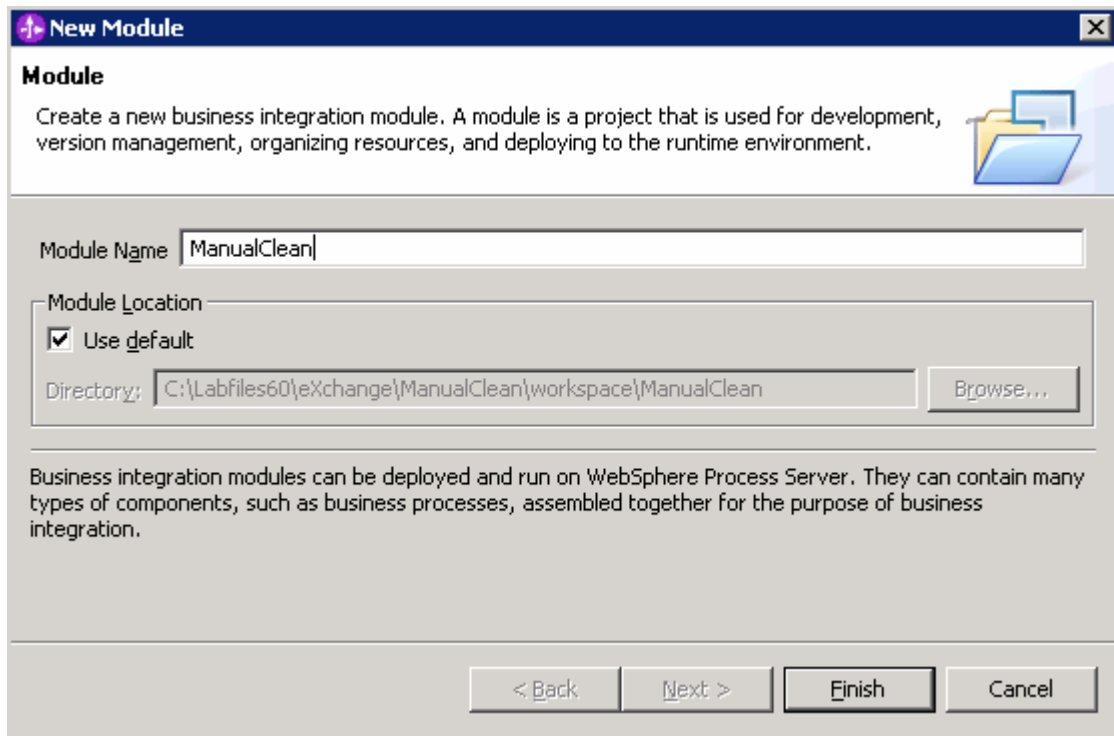
- ___ c. Click **Next**.
- ___ d. In the Import Projects dialog, initialize the From zip file: field to **<PROJECT_INTERCHANGE>**.
- ___ e. Click the **Select All** button.



- ___ f. Click **Finish**.
- ___ 4. If this lab requires that you create a Business Integration module called **<MODULE>**, complete these steps.
 - ___ a. Right click on the background of the Business Integration view to access the pop-up menu.
 - ___ b. Select **New > Module**.



___ c. In the New Module dialog, enter **<MODULE>** for the Module Name.

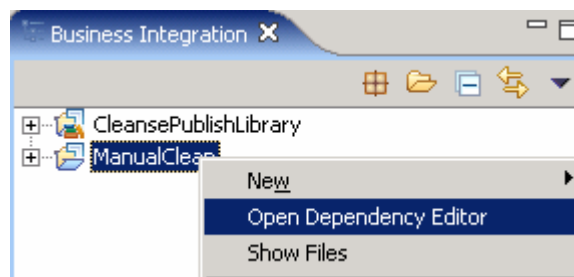


___ d. Click **Finish**.

___ 5. If this lab requires that **<MODULE>** needs any **<DEPENDENT_LIBRARIES>**, complete these steps:

___ a. In the Business Integration view, right click on the **<MODULE>** you just created to access the pop-up menu.

___ b. Select **Open Dependency Editor**.



___ c. In the Dependency Editor, click the **Add...** button.

___ d. In the Library Selection dialog, select from the list the **<DEPENDENT_LIBRARIES>**.

___ e. Click **OK**.

___ f. Press **Ctrl+S** to save the dependencies for this module.

Part 2: Create Manual Cleanse

There are 3 different types of tasks you can create for integration applications.

Participating tasks provide a way for an integration application to create tasks for individuals.

Originating tasks allow a person to invoke a service.

Pure Human tasks allow a person to create tasks for a specific person or a group of people.

Escalations can be set on all of these different tasks if the task is not claimed or completed in a specific time period. In this section you will create a participating task with different levels of permissions and escalation behavior.

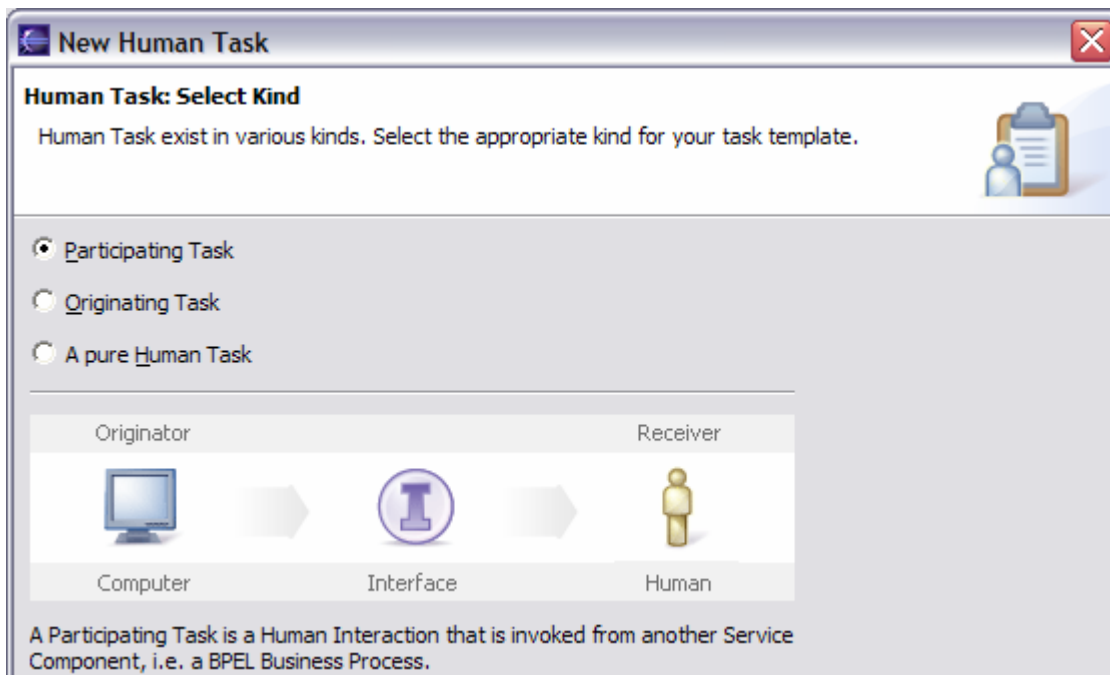
Note: In order to correctly build and test, four user IDs (user1, user2, user3, and adminuser) are assumed to exist on the system and be part of the Administrators group. You can substitute different user IDs in the instructions if different user IDs exist on your system.

- ___ 1. Create a participating human task named ManualClean.
 - ___ a. Expand **ManualClean** and **Business Logic** in the Business Integration view.
 - ___ b. Right click on **Human Tasks** and select **New > Human Task**.
 - ___ c. Enter **com.clipsandtacks.task** for the Folder.
 - ___ d. Enter **ManualClean** for the Name.

The screenshot shows a dialog box titled "New Human Task" with a subtitle "Human Task: Select Destination". Below the subtitle is the instruction "Specify Module, Folder, Namespace and Name for the Human Task". The dialog contains four input fields: "Module" with a dropdown menu showing "ManualClean" and a "New..." button; "Namespace" with a text box containing "http://ManualClean/com/clipsandtacks/task" and a checked "Default" checkbox; "Folder" with a text box containing "com.clipsandtacks.task" and a "Browse..." button; and "Name" with a text box containing "ManualClean".

- ___ e. Click **Next**.

__ f. Accept the default value to create a Participating Task and click **Next**.



__ g. Select the radio button for **Select an existing interface**.

__ h. Drop the interface down and select **Clean {
<http://CleansePublishLibrary/com/clipsandtacks/scm/Clean> }.**
The Operation should be set to **cleanseClip**.

__ i. Click **Finish**. The Human Task editor will open.

___ 2. Set the potential owner and administrator permissions for the task.

There are **5** different permissions that can be set on a participating task. (Other tasks have similar permissions with slight differences).

Potential Instance Creator – The Potential Instance Creator can be set to limit who can create a participating task. With a participating task, the originator of a task is a system and this permission can require that the system is passing the correct token for authorization before the task is created.

Potential Owner - Potential Owners can claim Tasks, or make changes to tasks they claim. Potential Owners are also ultimately responsible for completing tasks which they have claimed or assigned to others as sub-tasks. Potential Owners can also transfer claimed tasks to other Potential Owners of the Staff activity.

Editor - Editors are allowed to view or update tasks that have been assigned to other owners. When a task is claimed, the owner can update the task information without completing it. That information can then be updated by an Editor. Editors cannot complete tasks. For example, at an insurance call center, an employee (the owner) has claimed a task for resolving an insurance claim. In the meantime, another employee (with editor access) finds additional information that is relevant to the claim and updates the task. The owner of the task can then view the additional information and complete the task.

Reader - Readers can view tasks that have been claimed. They can view any information that may have been set before the task is completed. Readers can not change or complete tasks.

Administrator – Administrators can perform all the same tasks as the Potential Owner. The administrator can also complete tasks claimed by another person and transfer a task to someone who is not a Potential Owner.

When setting the Potential Owner role to a user or group of users, you must specify a verb and parameter(s). The Verb is mapped to a specific search which is run against the user registry by the Staff plugin provider for that user registry. By default, there are a number of Verbs and mappings available. These Verbs are also available for the Editor and Reader roles. Some more information on a number of Verbs:

Users - Specify a single user to claim and then complete a task. For a Local OS user registry this can be the user ID. For LDAP user registry, the expected format for this value would be the distinguished name (cn=John Doe, ou=New York, o=Acme, c=US). Best practice is to use the Group Members verb instead of Users and specify a group which contains the user. This best practice allows for installed applications with tasks to remain unchanged when changes occur within the company hierarchy.

Users by user ID - Specify a user by their user ID. For a Local OS user registry this is the same as **Users**. For LDAP user registry, this value would be the **uid** attribute for an inetOrgPerson. Best practice with this Verb is to use the Group Members verb rather than a specific user.

Group Members - Specify a group of users who can claim and then complete a task.

Group Members without Named Users – Specify the group and a user who should be excluded from receiving a task. This verb, with the use of a process context value, can be used to implement a “4-eyes” policy where tasks must be approved by two different users from the same group.

Manager of Employee - Specify an employee by distinguished name (cn=Susan Long, ou=New York, o=Acme, c=US). The manager of the employee will then be able to claim and complete the task. Use of this verb requires a Manager-Employee relationship to exist with the user registry. For LDAP user registry, the manager attribute is used by default for resolving this query. Default support for this verb is for an LDAP user registry only.

Manager of Employee by user ID - Very similar to **Manager of Employee** except the employee user ID is used instead of the full distinguished name. Support for this Verb is only available for LDAP user registries.

Native Query - Specify a query specific to the user registry. Before using Native Query, consider creating a new Verb and appropriate translation to the actual implementation. With a Native Query, changes made to the user registry have the potential to affect the business process and force updates to be made to the process. This forces the business process to be deployed again to the Business Process engine.

Everybody - Use this Verb instead of maintaining tasks for each and every different user

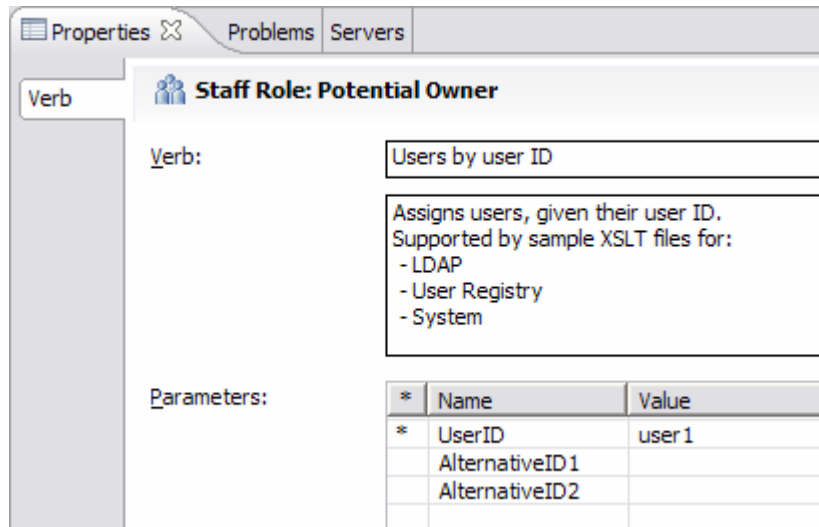
Nobody - Use this Verb to restrict access to a role. Nobody can only be used in Reader roles.

___ a. A Potential Owner is required for any participating task and is created by default under the Receiver Properties. Select the Potential Owner to set the appropriate user. The Properties view and Verb tab should be shown.

___ b. For the Verb, select **Users by user ID**.

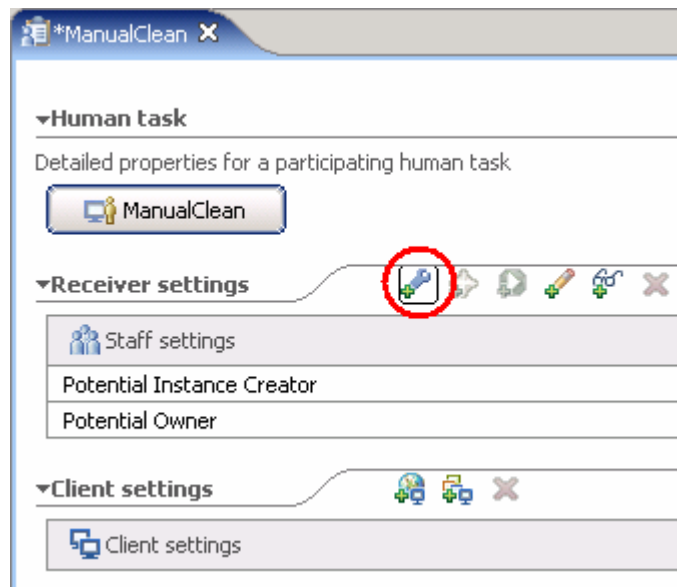
Note: For simplicity, a single user will be specified rather than a group even though using the Group Members verb is a best practice.

___ c. Enter **user1** for the User ID.

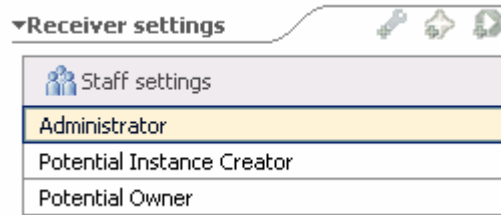


___ d. View the setting for the Potential Instance Creator. The default setting is Everybody and any client will be able to create a Manual Clean task.

___ e. Select the appropriate icon to set an administrator for the task.

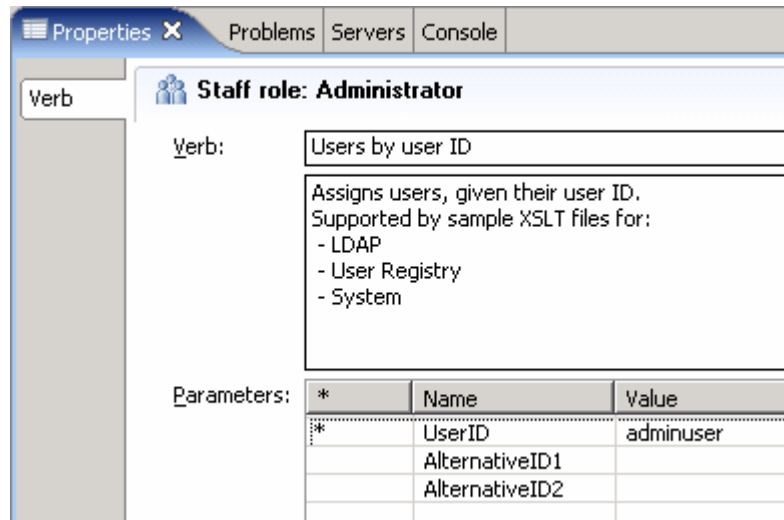


__ f. Select Administrator and then the Properties view if it is not already shown.



__ g. Select **User by user ID** for the Verb.

__ h. Type in **adminuser** for the User ID parameter.

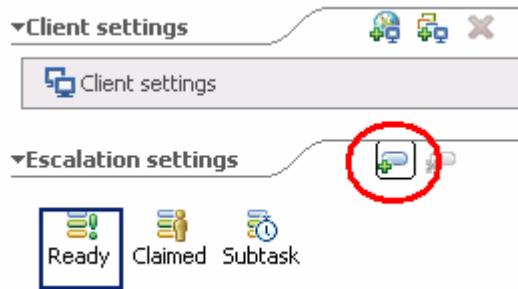


___ 3. Set an escalation on the task to assign a task if it has not been completed in a timely fashion.

A robust set of escalations may be established for a task. Escalations are run based on an exceeded duration and the current state of task. When a state is reached (Ready, Claimed, or Sub-tasks created), the escalation chain begins at that state. The first escalation in the chain will run after the duration has exceeded and will check to see if the required state for the escalation has been reached. If the state has been reached, the escalation will not fire and the escalation chain will continue. If there is another escalation in the chain, it will fire when its duration has been reached, checking to see if the required state has been reached for the task. Escalation times are relative to the preceding escalation durations. For example if escalation is 2 days, and the second escalation is 3 days. The first escalation would occur after 2 days and the second escalation would occur 3 days later or 5 days from the start of the escalation chain.

__ a. Select **Ready** in the Escalation settings section.

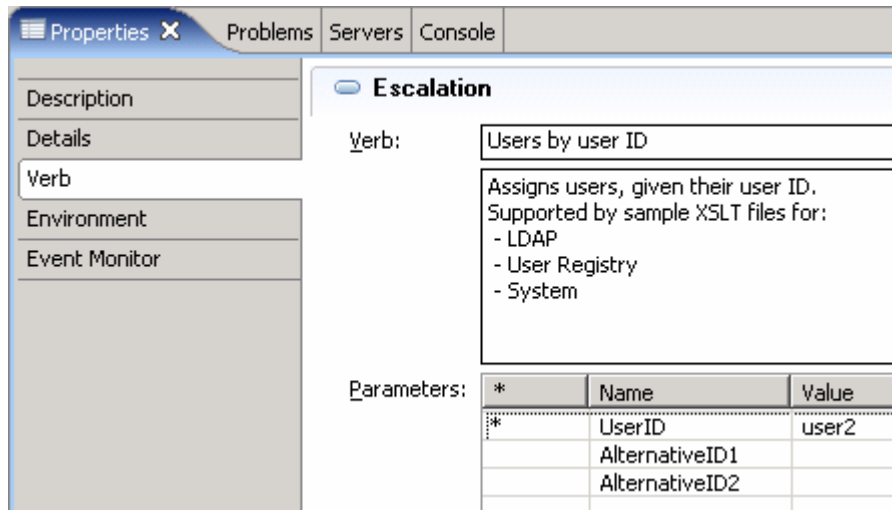
__ b. Click the **Add Escalation** button.



__ c. In the Properties view for the escalation, there are a number of different settings. Escalation can result in an action being performed such as creation of a work item for a task, generation of an e-mail, or an event generation, which can be monitored at a central location. For assigning tasks, a Verb and Parameter must be specified and will be used to search a user registry for the appropriate person to assign the task. Select the **Verb** tab.

__ d. Select **Users by user ID** for the Verb.

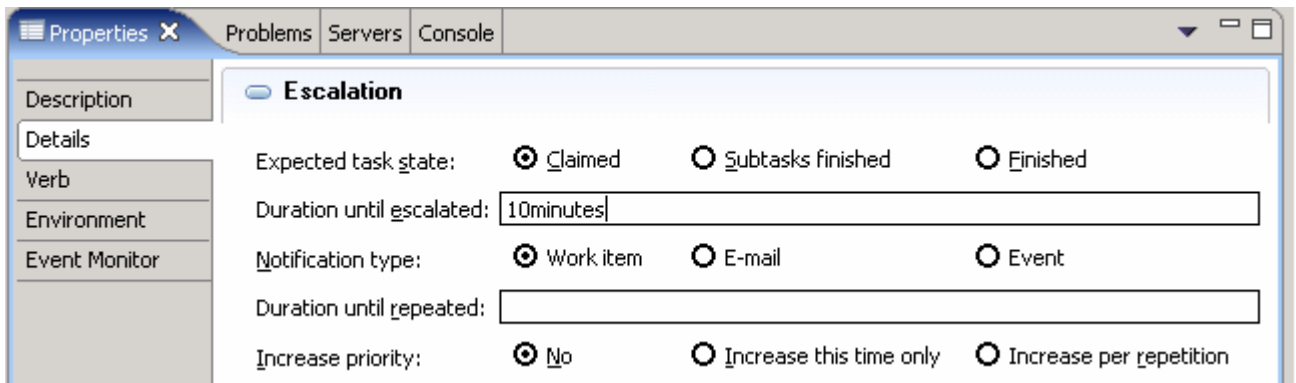
__ e. Enter **user2** for the User ID. This user will be involved in the outcome of this escalation.



__ f. Select the **Details** tab. The combination of the state and the duration are checked at the time the escalation occurs and can be specified on this tab.

__ g. Select the **Claimed** radio button for the **Expected task state**.

- ___ h. Enter **10minutes** for the **Duration until escalated**. This escalation will occur after 10 minutes if the task is not claimed. Ensure the Work item radio button is selected. The escalation is complete.



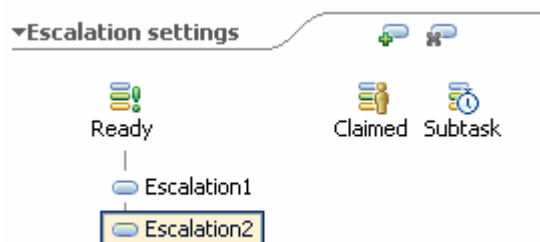
Note: The format for the duration is based on the calendar, which is set on the Details tab for the task. The calendar supports different time formats, such as 10minutes, 5days, or 2months. Custom calendars may also be created to support additional formats.

Note: Escalation occurs when the duration is exceeded. A built-in scheduler checks for escalation approximately every 10 minutes and is not meant for precision firing. Therefore, if an escalation duration is reached 1 minute after the scheduler last ran, it could be over 9 minutes before the escalation occurs. If escalations must fire exactly after the duration has expired, you might want to consider a solution other than the built in escalation capabilities.

Note: For this example a small duration was used in order to easily simulate escalations. Typically, a larger value would be used.

In summary, for this escalation, after the task is created, if it is not claimed in 10 minutes, a task will be created for user2.

- ___ i. A second escalation can be created after the first escalation. Select **Escalation1** and click the **Add Escalation** button again.



- ___ j. For the second escalation, follow similar steps as setting the first escalation and set the correct values to create a task for **user3** after **10minutes** if the task is not **Claimed**. For the second escalation of 10 minutes, this will fire 10 minutes after the first escalation has fired or a total of 20 minutes after the task was created. You could also create an escalation chain which would start when the task was claimed. To create these escalations, follow the same steps as above.

- ___ k. When complete, save the Task.

Part 3: Prepare for Unit Testing and Assemble Manual Cleanse

In this part, you will prepare the task for testing by specifying the Staff plugin provider and assembling it as a SCA component. In order for a participating task to be called by other SCA components, you must create an SCA binding.

1. Specify the Staff plugin provider for the task.

In order for tasks to be assigned, a query must be run against a user registry to locate the correct user(s) based on the Verb and Parameters specified. The query is run by a resource running in WebSphere Process Server known as a Staff plugin provider. Since each user registry is potentially different in schema and connection type, there are different Staff plugin providers for each user registry. WebSphere Process Server includes 3 Staff plugin providers that can be used for locating users for task assignment:

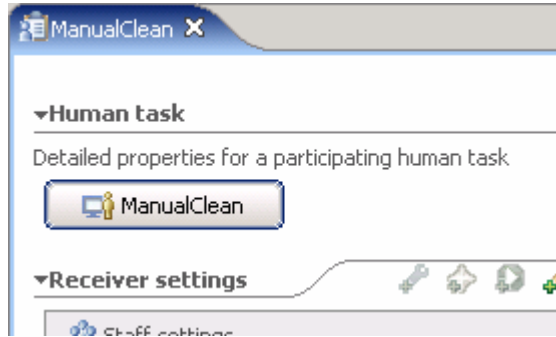
LDAP Staff plugin provider – LDAP servers can be used for locating users for task assignment with this Staff plugin provider. This provider supports the largest number of verbs and provides the highest level of customization because many attributes can be defined in the LDAP server and these can be used in searches from the LDAP Staff plugin provider.

User Registry Staff plugin provider – The User Registry Staff plugin provider directs all searches for users and groups to the security manager running in WebSphere Process Server. When Global Security is enabled, the security manager communicates with a particular user registry (local OS, LDAP, or custom registry) to perform authorization of requests coming in to the server. The User Registry Staff plugin provider can pass searches to the security manager and utilize any of the different registries. Since the security manager only needs to search for users and groups, the number of Verbs available to the User Registry Staff plugin provide is limited to searches for users and groups too. Verbs such as Manager of Employee, Department Members, and Role members are not available for this Staff plugin provider.

System Staff plugin provider – For test environments, the System Staff plugin provider supports a small number of verbs and is ideal for verifying the logic of task assignment without dealing with the complexities of actual user registries. With this Staff plugin provider, no queries are run against a user registry to locate users or groups. Instead, hard-coded values, specified as Parameters are used. The System Staff plugin provider also contains a special configuration that can assign all tasks to anyone (everybody). With this setting, everyone will be able to view, claim, or complete tasks without configuring security on the server. Because of the lack of capabilities and potential security risks, the System Staff plugin provider is only supported for test environments and is not intended for production servers.

You can prepare the ManualClean task for easy testing by setting it to use the Everybody capability on the System Staff plugin provider.

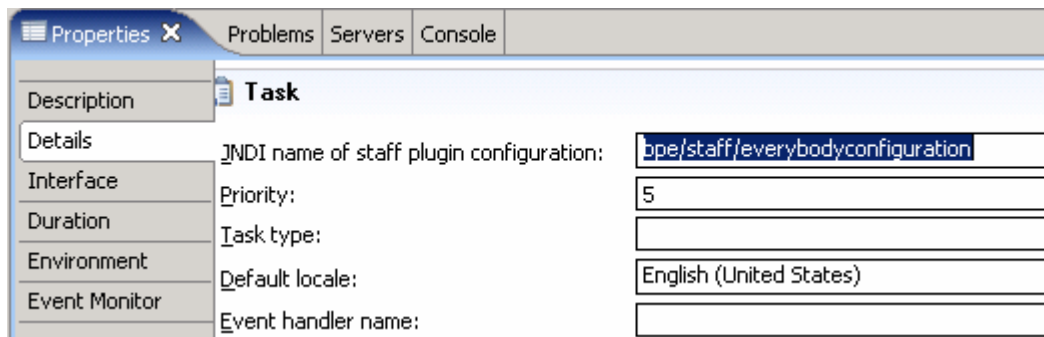
__ a. Select **ManualClean** in the Task editor.



__ b. Select the Properties view if it is not already shown and the **Details** tab.

__ c. Select **bpe/staff/everybodyconfiguration** for the JNDI name of staff plug-in configuration. This is the configuration which provides the ability to make tasks available to everyone and is part of the System Staff plug-in provider which is preconfigured in WebSphere Process Server.

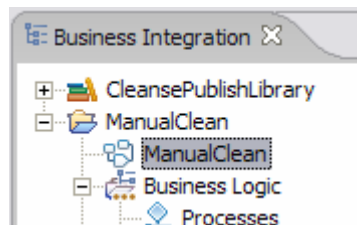
Note: The Staff plugin providers can be viewed in the WebSphere Process Server Administrative console under Resources > Staff plugin provider.



__ d. Save and close the Task.

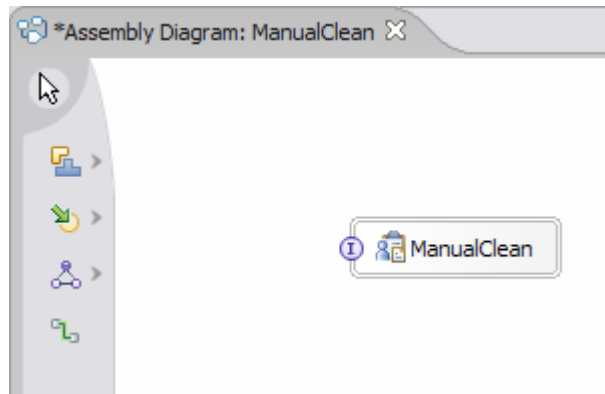
___ 2. Assemble the human task as a SCA component.

__ a. Expand **ManualClean** in the Business Integration view and double-click on **ManualClean** to open the Assembly editor.

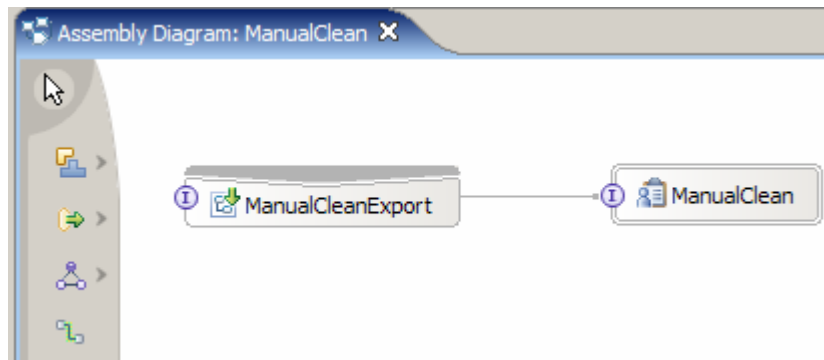


__ b. Expand Business Logic and Human Tasks in the Business Integration view.

__ c. Select and drop the **ManualClean** task onto the Assembly editor.

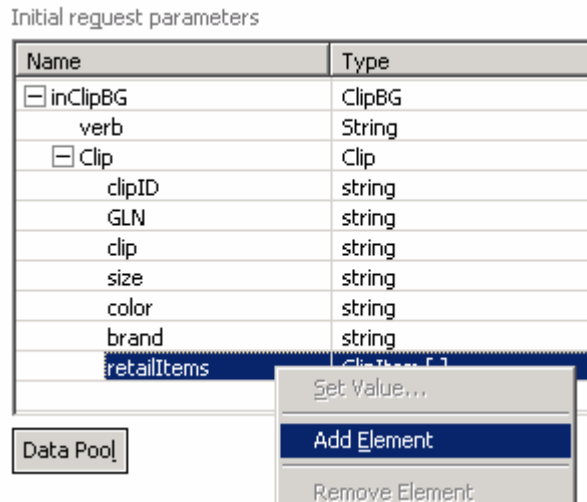


___ d. Right-click on ManualClean and select **Export....** and **SCA Binding**.

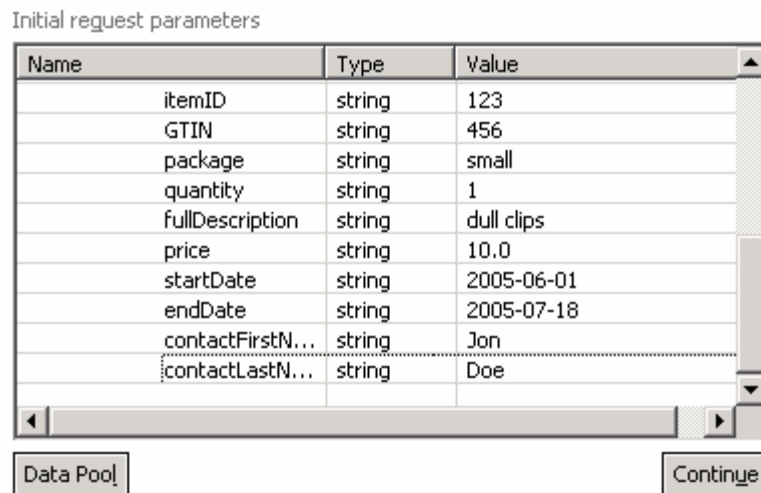


___ e. Save the Assembly editor. The task is now an SCA component and the appropriate code will be generated to expose the ManualClean human task as a SCA component.

___ c. Right-click on the value for retailItems and select **Add Element**.



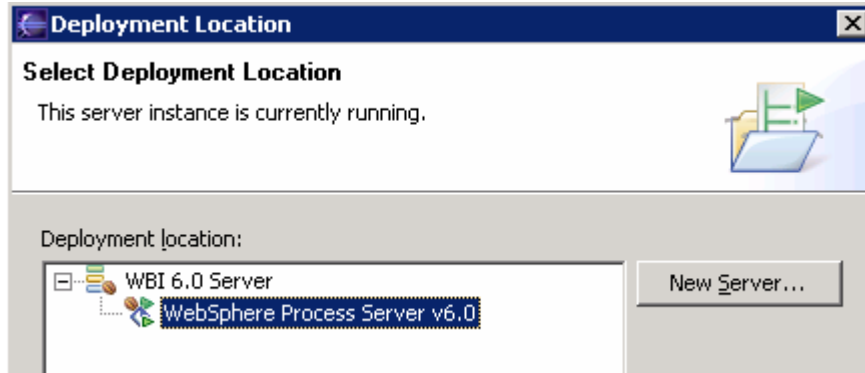
___ d. Scroll down and fill in the values according to the graphic or use values of your choice.



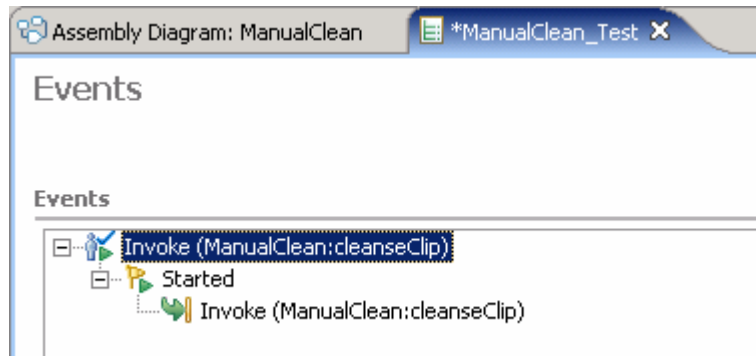
___ 4. Run the test.

___ a. Click the **Continue** button under the list of initial request parameters.

- ___ b. Expand **WBI 6.0 Server** in the Deployment Location window and select the appropriate WebSphere Process Server on which you want to deploy the application. This is especially true for z/OS, AIX, Solaris remote test environment, where the WebSphere Integration Developer will be remote to the test environment. Click Finish



- ___ c. Click **Finish**. The ManualCleanApp application and the Test Connector application which drives the testing on the server will be published to the server. The SCA component will also be invoked.

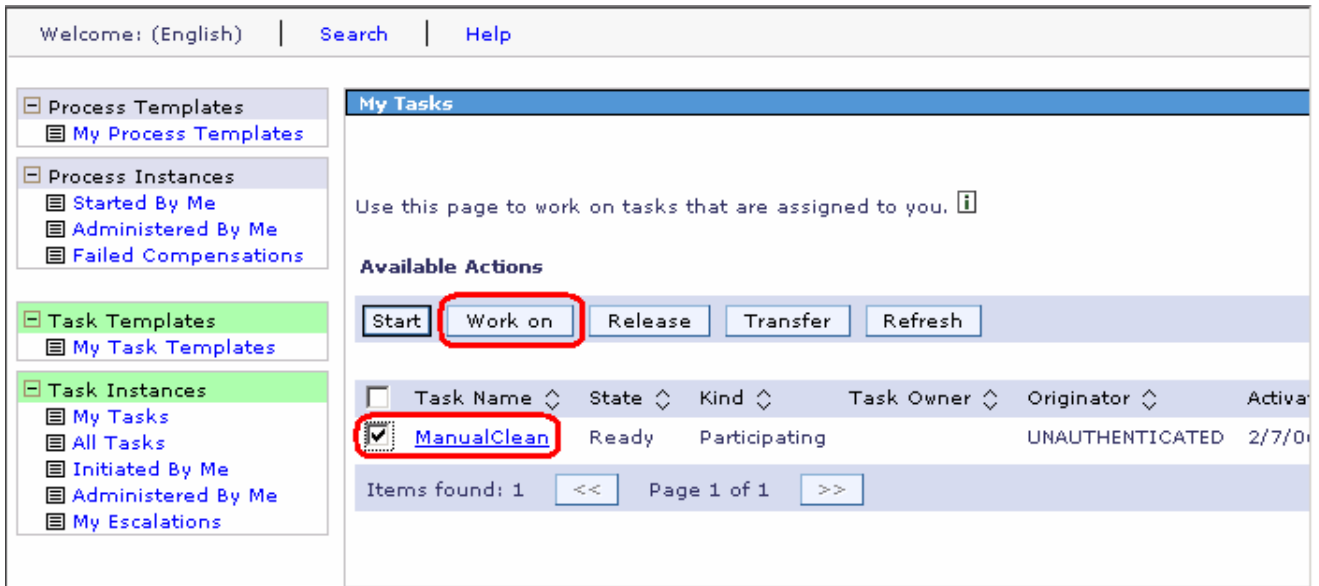


___ 5. Complete the Task implementation.

The SCA interface on the task is called and a task will be created and available to anyone as the Everybody configuration of the System Staff plugin provider was used. WebSphere Process Server includes the BPC (Business Process Choreography) Explorer, which provides a web interface for interacting with business process instances and tasks generated from human tasks. Users can view, claim, or complete tasks, depending on their permission levels for the tasks.

- ___ a. Right-click on WPS Server v6.0 In the Servers view and select **Launch > BPC Explorer**. You can also open a Web browser and navigate to <http://<HOSTNAME>:9080/bpc>. The My Tasks view will be shown with the task ready to be completed.
- ___ b. The task for ManualClean will be listed in the view. Since security is not enabled and the System Staff plugin provider with the Everybody configuration is being used, the tasks are listed for all users. If security was enabled and the Everybody configuration was not used, only those tasks which you have permission to work with will be listed.

- c. Check the box for the instance and click **Work on**. This will claim the task on behalf of the current user. Since security is not enabled and you are unauthenticated, the task is claimed, but not for a specific person.



- d. The task will be displayed with the input message to the task also displayed.

Available Actions



Task ManualClean

Instance Name

Task Input Message

inClipBG	Clip	dipID	1000														
		GLN	1000														
		dip	testClip														
		size	1														
		color	red														
		brand	dull														
		retailItems	<table border="1"> <thead> <tr> <th>itemID</th> <th>GTIN</th> <th>package</th> <th>quantity</th> <th>fullDescription</th> <th>price</th> <th>start</th> </tr> </thead> <tbody> <tr> <td>123</td> <td>456</td> <td>small</td> <td>1</td> <td>dull clips</td> <td>10.0</td> <td>2005-06-0</td> </tr> </tbody> </table>	itemID	GTIN	package	quantity	fullDescription	price	start	123	456	small	1	dull clips	10.0	2005-06-0
itemID	GTIN	package	quantity	fullDescription	price	start											
123	456	small	1	dull clips	10.0	2005-06-0											

- e. There will also be fields displayed where the information for the output message can be entered. Scroll down and enter the following values or any values you choose. The type for the values will be checked before it is submitted if you enter an incompatible value.

Click the “+” button to enter values for the retailItems business object.

Task Output Message

outClipBG

Clip

clipID 1001

GLN 1001

clip testClip2

size 500

color very red

brand very dull


retailItems +

__ f. Enter the values for the retail value.

retailItems	itemID	GTIN	package	quantity	fullDescription	price	startDate	endDate	contactFirstName	contactLastName
	456	789	medium	2	dull clips	20.0	2005-07-01	2005-12-31	Jane	Doel

__ g. Scroll back to the top and click **Complete**. The list of tasks will be displayed again and will show the task as finished.

Task Message

Use this page to provide the data required to complete the task. 

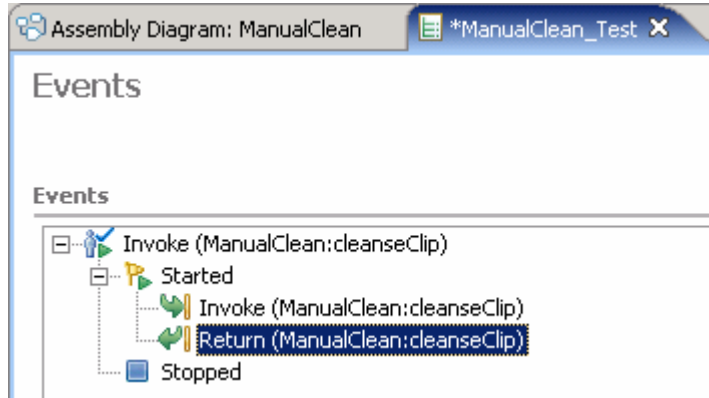
Available Actions

Task Instance Name ManualClean

Task Input Message

inClipBG	Clip	clipID	1000
----------	------	--------	------

___ h. Return to the component test window and view the return response captured.



___ i. Notice in the Detailed Properties section, the response that you filled in when completing the task is shown. The test was successful.

► **General Properties**

▼ **Detailed Properties**

Module: [ManualClean](#)

Component: [ManualClean](#)

Interface: [Clean](#)

Operation: [cleanseClip](#)

Return parameters:

Name	Type	Value
[-] outClipBG	ClipBG	
verb	VerbType	<null>
[-] Clip	Clip	
clipID	ClipIDType	1001
GLN	GLNType	1001
clip	ClipType	testClip2
size	SizeType	500
color	String	very red
brand	BrandType	very dull
[-] retailItems	ClipItem []	
[-] retailItems[0]	ClipItem	

___ j. You can run another test using the same values by clicking **Invoke** or you can close and save the test.

Note: If the task is not claimed within 10 minutes, the first escalation will be run. The Everybody configuration is used with the Staff plug-in provider and when the escalation occurs, a task will be assigned to everybody. Since the task was originally assigned to everybody, no additional work items are created for the task. For this reason, the actual escalation behavior is not exhibited in this exercise. If a different configuration was used with the Staff plug-in provider and global security was enabled, a new and different task would be created for the escalation receiver.

- ____ 6. Clean up test environment.
 - ___ a. Right-click on appropriate WPS Server on which you deployed the applications in the Servers view and select Add and remove projects...
 - ___ b. Click << Remove All.
 - ___ c. Click Finish.
- ____ 7. Stop the Server
 - ___ a. Right click on WebSphere Process Server v6.0 server from the Servers view and select Stop from the context menu.

What you did in this exercise

In this exercise, you built a human task with a potential owner and an administrator. You added escalations to create tasks after a particular duration is exceeded if the task is not claimed. With the End-to-End Test Framework, you tested the task as a SCA component, verifying task creation and completion.

Solution Instructions

- ____ 1. Follow the directions in the task [Initialize the Workspace for a Lab Exercise](#), using the following values:

<WORKSPACE>

C:\Labfiles60\exchange\ManualClean\workspace

<PROJECT_INTERCHANGE>

C:\Labfiles60\exchange\ManualClean\Solution\ManuelClean_P1.zip

<MODULE>

n/a

<DEPENDENT_LIBRARIES>

n/a

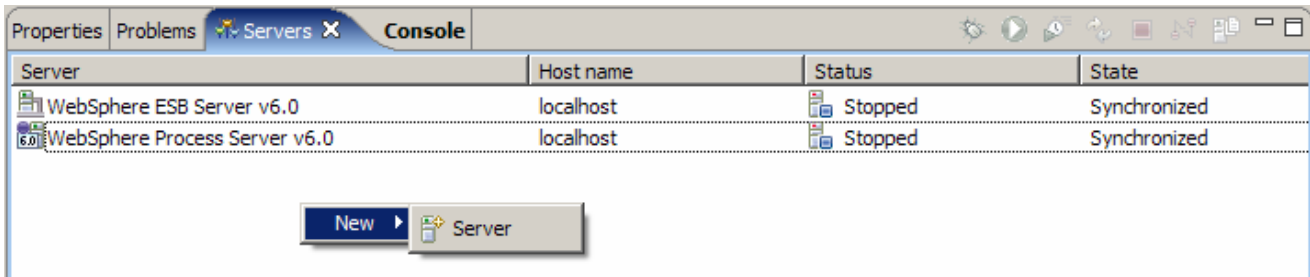
- ____ 2. Continue with **Part 4: Test ManualCleanse.**

Task: Adding Remote Server to WebSphere Integration Developer Test Environment

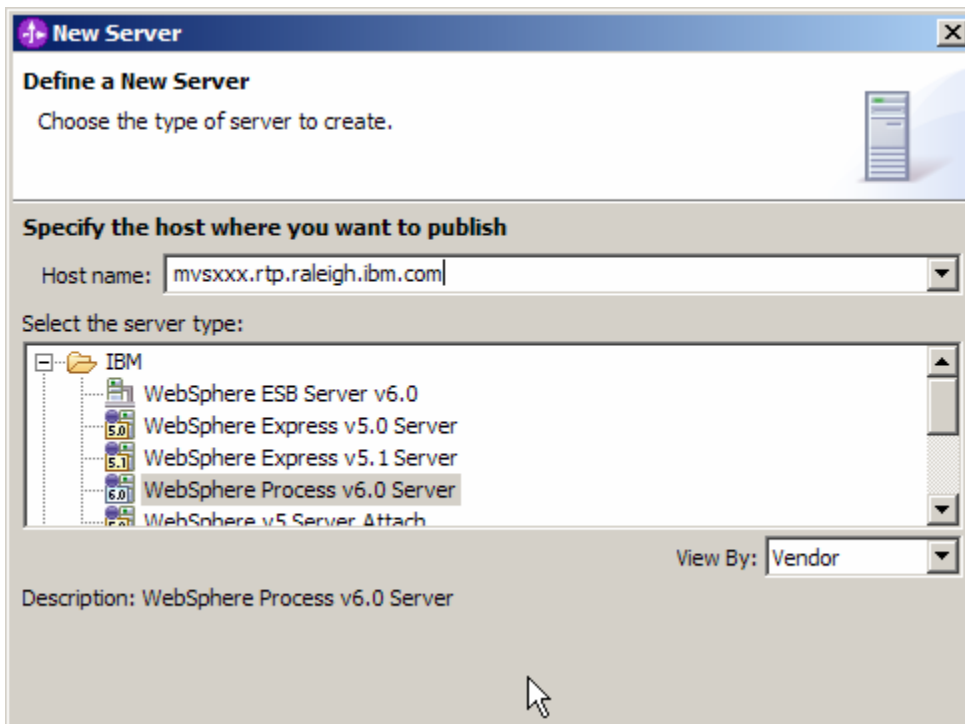
This task describes how to add a remote server to the WebSphere Integration Developer Test environment. This sample will use a z/OS machine.

Create a new remote server.

- ___ 1. Right click on the background of the Servers view to access the pop-up menu.
- ___ 2. Select **New > Server**.



- ___ 3. Specify hostname to the remote server, **<HOSTNAME>**.
- ___ 4. Ensure that **'WebSphere Process v6.0 Server'** is highlighted in the server type list.



- ___ 5. Click **Next**.
- ___ 6. On the WebSphere Server Settings page, select the radio button for **RMI** and change the ORB bootstrap port to the correct setting (**<BOOTSTRAP_PORT>**).

New Server

WebSphere Server Settings

Input settings for the new WebSphere server

WebSphere profile name: []

Server connection type and admin port

RMI (Better performance)

ORB bootstrap port: [9131]

SOAP (More firewall compatible)

SOAP connector port: [8880]

Run server with resources within the workspace

Security is enabled on this server

Current active authentication settings:

User ID: []

Password: []

Server name: [server 1]

Server type

BASE, Express or unmanaged Network Deployment server

Network Deployment server

Network Deployment server name: []

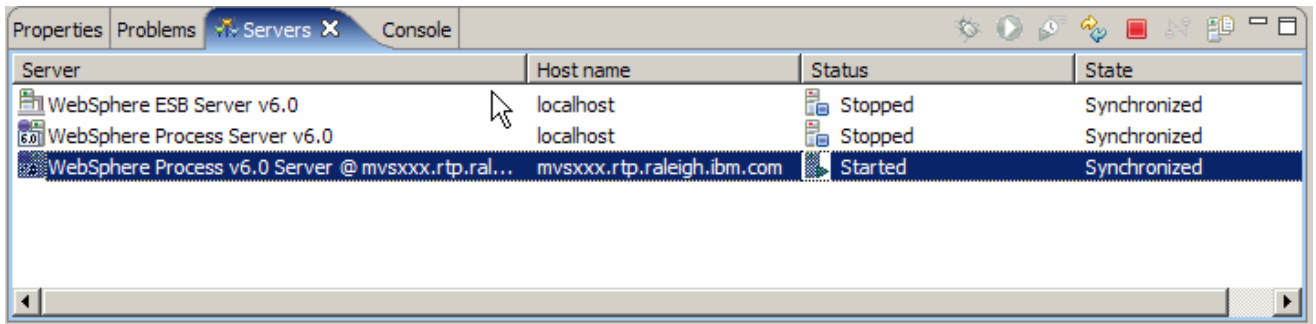
The server name is in the form of:
<cell name>/<node name>/<server name>
For example, localhost/localhost/server1.

Click this button to detect the server type.

< Back Next > Finish Cancel

___ 7. Click **Finish**.

___ 8. The new server should be seen in the Server view.



- ___ 9. Start the remote server if it is not already started. WebSphere Integration Developer does not support starting remote servers from the Server View.
- ___ 10. From a command prompt, telnet to the remote system if needed:
 - 'telnet <HOSTNAME> <TELNET_PORT>'**
 - userid : **<USERID>**
 - pw : **<PASSWORD>**
- ___ 11. Navigate to the bin directory for the profile being used:
 - cd <WAS_HOME>/profiles/<PROFILE_NAME>/bin**
- ___ 12. Run the command file to start the server: **./startServer.sh <SERVER_NAME>**
- ___ 13. Wait for status message indicating server has started:

```

ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server c11sr01 open for e-business; process id is 0000012000000002
    
```