Configuring

Running the sample workflows for processing orders retrieved from MarcomCentral

The **MarcomProcessOrders** and **MarcomProcessJobTicket** workflows show how to process orders retrieved from a sample store at the MarcomCentral website. The sample store serves as a test environment for the RICOH ProcessDirector web services interface to MarcomCentral. A SOAP web service input device, a **CallSOAPService** step, and a SOAP web service notification communicate with the MarcomCentral website. The workflows also include examples of these steps: **ApplyXSLTransform**, **CreateOrdersFromFile**, and **DownloadFile**

The sample input device, **CallSOAPService** step, and notification run a simulation. They do not call the specified MarcomCentral web services. The simulation lets you run the samples without getting your own MarcomCentral credentials and setting up a sample MarcomCentral store that resembles the sample store. The samples use SOAP 1.1 request payloads to retrieve a simulated MarcomCentral order and two job tickets. The value in the static credential and password fields tells RICOH ProcessDirector to run the simulation.

After you run the samples, you can copy them. Then you can adjust the values in the copies to exchange data with your own store at the MarcomCentral website.

> ⬇ Note:
> • The sample objects and files are installed with the MarcomCentral Connect feature.

The simulation includes Web Services Description Language (WSDL) SOAP request objects imported from MarcomCentral. The sample input device, **CallSOAPService** steps, and notification each specify a SOAP request.

This simulation uses these objects and files:

- Workflows:
  - MarcomProcessOrders
  - MarcomProcessJobTicket
- Order property mapping: **MarconOrderSample**
- Input device: **MarcomReceiveOrders**
  The input device retrieves XML orders from the MarcomCentral website by simulating a call to a web service at
  `https://services.printable.com/Trans/1.0/Order.asmx`
  The SOAP request is `Marcom-GetOrdersByDate`.
- Notification: **MarcomCloseoutOrder**
  The notification sends order completion information to the MarcomCentral website by simulating a call to a web service at
  `https://services.printable.com/Trans/1.0/Closeout.asmx`
  The SOAP request is `MarcomSample-CloseoutByOrder`.
- XML orders
  In the simulation, one order is retrieved every 30 seconds from the web service.
  After you run the sample workflows, you can see the contents of a sample order. Go to: `/aiw/aiw1/spool/default/`*`job_number`*
  Replace *job_number* with the number of the job, for example, `10000001`.
  Display the contents of the *`job_number`*`.print.xml` file, for example, `10000001.print.xml`.
- XML job tickets
  In the simulation, two job tickets are retrieved for every order.
  After you run the sample workflows, you can see the contents of the two sample job tickets. Go to:
  `/aiw/aiw1/spool/default/`*`child_job_number`*
  Replace *child_job_number* with the number of the child job, for example, `10000001.1` or `10000001.2`.
  Display the contents of the *`child_job_number`*`.print.xml` file, for example, `10000001.1.print.xml` or `10000001.2.print.xml`.
- Print file: `Savings.pdf`
  This file is downloaded locally from `http://localhost:15080/restapi/`
- XSLT style sheet file: `downloadOverrides.xslt`
  The file is in the `Marcom` directory: `/aiw/aiw1/testfiles/Marcom`
- **CallSOAPService** step:
  - The **CallSOAPService** step retrieves a job ticket from the MarcomCentral website by simulating a call to a web service at
    `https://services.printable.com/Trans/1.0/JobTicket.asmx`
    The SOAP request is `Marcom-GetJobTicketByLineItem`.

To run the sample workflow:

1. Click the **Main** tab.

2. In the Printers portlet, right-click the **Sample** printer and select **Enable**.

3. In the Input Devices portlet, right-click the **MarcomReceiveOrders** SOAP web service input device and select **Enable and Connect**.
   A job named **Marcom Sample Order** appears in the jobs table.
4. Right-click the **MarcomReceiveOrders** input device and select **Disable and Disconnect**.

> ⬇ Note:
> - The sample input device polls for orders every 30 seconds and retrieves an XML job. If you do not disable and disconnect the input device, a new job appears in the jobs table every 30 seconds.

The request payload for the input device includes these RICOH ProcessDirector symbols:
- `${WebService.StaticCredential}`
  Specifies the static credential
- `${WebService.LastSuccessRequestTime}`
  Specifies the date and time when the input device last successfully requested a response from the web service.
- `${WebService.CurrentRequestTime}`
  Specifies the date and time of the current request.

Each time the **MarcomReceiveOrders** input device retrieves an XML job, it sends the job to the **MarcomProcessOrders** workflow.
The job goes through these and other steps in the **MarcomProcessOrders** workflow:
- The **SetJobPropsFromTextFile** step sets the **Job name** property to `Marcom Sample Order`.
- The **CreateOrdersFromFile** step uses the **MarcomOrderSample** property mapping object to identify orders and jobs, then map the values of some XML elements in the XML order file to order and job properties.
- The table lists the properties that are set.

XML element	Database name	User interface name
`/Order/OrderNumber`	Order.Name	Order name
`/Order/ID`	Order.Reference	External order reference
`/Order/OrderDetails/OrderDetail/User/Name`	Order.Customer	Customer name
`/Order/OrderDetails/OrderDetail/SKUDescription`	Order.Description	Description
`/Order/OrderDetails/OrderDetail/Quantity`	Job.Copies	Job copies requested
`/Order/OrderDetails/OrderDetail/OrderNumber`	Job.Marcom.OrderNumber	MarcomCentral order number
`/Order/ID`	Job.Marcom.OrderId	MarcomCentral order ID
`/Order/OrderDetails/OrderDetail/ID`	Job.Marcom.JobTicketId	MarcomCentral job ticket
`/Order/OrderDetails/OrderDetail/ProductType`	Job.Marcom.ProductType	MarcomCentral product type
`/Order/OrderDetails/OrderDetail/ProductName`	Job.Name	Job name

The step identifies one order with two jobs in it. It creates two jobs, **Jensen Bank Baseball Cap** and **Brochure**, and submits the jobs to the **MarcomProcessJobTicket** workflow.
The original order job is sent to the **WaitForRelatedJobs** step.

The jobs start processing in the **MarcomProcessJobTicket** workflow. When the jobs leave the **SetJobPropsFromTextFile** step, the rule on the **[1] Warehouse** connector checks the value of **MarcomCentral product type** property. **Versioned**, **Variable**, and **JobDirect** are product types for printable items.
- If the item does not have a printable product type, the workflow sends the job through the **Warehouse** branch.
- If the item has a printable product type, the workflow sends the job through the **Printable** branch.

Non-printable jobs go through the **ManualStepWithAutoStart** step in the **Warehouse** branch of the workflow. The jobs wait in that step until the warehouse staff indicate that the jobs are ready to ship.
Printable jobs go through these steps and connectors in the **Printable** branch of the workflow:
- When the job arrives at the second **ContinueToNextStep** step, the rule on the **[1] File ready** connector checks whether the **Custom 3** (URL of download file) property is set to a value.
  - If it is, the print file is ready to download. The workflow sends the job directly to the **DownloadFile** step.
  - If it is not, the rule on the **[2] Check for file** connector checks whether the value of the **Custom integer 1** job property is less than `10`.
    The first time that the job arrives at the second **ContinueToNextStep** step, the value of the **Custom integer 1** property is `0`.
    - If the value is `0` through `9`, the workflow sends the job to the **Wait** step, which waits for 30 seconds. The workflow then sends the job to the **CallSOAPService** step.
      The **CallSOAPService** step calls the MarcomCentral web service that retrieves a job ticket for a printable item. The step specifies two RICOH ProcessDirector symbols as part of the value in the **Request payload** property:
      - `${Job.WebService.Password}`
        Specifies the static credential.
      - `${Job.Marcom.JobTicketId}`
        Specifies the job ticket ID.

      The **ApplyXSLTransform2** step uses the `downloadOverrides.xslt` XSLT style sheet to convert an XML element into a RICOH ProcessDirector job property:

XML element	Database name of job property	User interface name of job property
`JobTicket/FinalOutputFileURL/URL`	Job.Info.Attr3	Custom 3

      The step puts the value in an overrides file, and overwrites the overrides file in the spool directory for the job with the new overrides file.

This example shows the contents of the sample overrides file:

```
Job.Info.Attr3=http://localhost:15080/restapi/Savings.pdf
```

The **AssignJobValues** step increments the value of the **Custom integer 1** property by `1`. The workflow then sends the job back to the second **ContinueToNextStep** step, which sends the job through the **Wait** step to the **CallSOAPService** step. The **CallSOAPService** step repeats the call to the MarcomCentral web service that retrieves a job ticket for a printable item. If the **Custom 3** property still does not have a value for the URL, the workflow sends the job through the **CallSOAPService** step up to 9 times.

- If the value is `10`, the job has been sent to the **CallSOAPService** step 10 times. After 5 minutes, the **Custom 3** property still does not have a value for the file to download. The job does not meet the rule on the **[2] Check for file** connector. The workflow sends the job through the **[3] Timeout** connector to the **FailWithMesssage** step and writes a failure message to the job log. The message states: `File was not available to download in the time allowed.`

> ⬇ Note:
> - Because the message is a job property, it appears on the **Information** tab of the job property notebook for all jobs that go through the workflow. The message appears in the job log only when the workflow sends the job through the **[3] Timeout** connector to the **FailWithMesssage** step.

- The **DownloadFile** step downloads the `Savings.pdf` print file from `http://localhost:15080/restapi/` by setting the value of the **URL for download file** property to the `${Job.Info.Attr3}` symbol.
- The **OptimizePDF**, **CountPages**, **CreatePageRanges**, and **PrintJobs** steps process and print the job on the **Sample** printer.

The **WaitForRelatedJobs** step holds the parent job and each of the child jobs until all the child jobs arrive at the step.

5. In the jobs table, right-click the **Jensen Bank Baseball Cap** job and select **Manual Complete**.

6. Select **Move to next step** and click **OK**.
   The job moves to the **WaitForRelatedJobs** step.
   When both jobs reach the **WaitForRelatedJobs** step, the workflow sends all three jobs (the original order job, the non-printable job, and the printable job) to the **RetainCompletedJobs** step. The state of each job changes to **Retained**. The state of the order changes to **Complete**. The **MarcomCloseoutOrder** web service notification calls the web service at
   `https://services.printable.com/Trans/1.0/Closeout.asmx`
   The request payload includes these RICOH ProcessDirector symbols:
   - `${WSNotification.WebService.Credential}`
     Specifies the static credential.
   - The `${Job.Marcom.OrderId}`
     Specifies the order ID.

   If the notification called the web service instead of running the simulation, this change would occur at the sample store on the MarcomCentral website. On the **Display by Item** dialog in the **Order Manager**, the value in the **Order Status** column for each item in order number WS113 would change from **Work in Progress** to **Shipped**.

7. In the jobs table, right-click the **Brochure** job and select **View**.
   The viewer displays the PDF file that was printed as part of the order.

To run the sample workflows again, enable and connect the **MarcomReceiveOrders** input device. Then disable and disconnect it.
Now that you have run the samples, you can copy them. Adjust the values in the copies to exchange data with your own store at the MarcomCentral website. See the related topics.

Parent topic: Setting up to process orders from MarcomCentral