Configuring

Calling a REST web service from a workflow

To call a REST web service, identify the parameters used to call the REST web service. Test the exchange of data between RICOH ProcessDirector and the application. Then add a step based on the **CallRESTService** step template to the workflow.

To call a REST web service from a workflow:

1. Learn the requirements for communication with the REST web service for the application:
   - The values for authenticating with the application
   - The values for requesting data from the REST web service
   - The format of the data provided in the response

   Refer to the documentation for the application or consult with the company that hosts the application.
2. To prepare RICOH ProcessDirector to communicate with the application, do these tasks:
   - If the application requires a security certificate, install the certificate on the RICOH ProcessDirector primary computer.
   - If your environment requires a proxy server to communicate with web services, set up the system to use it.

   For more information, see the related tasks.
3. Run a manual test that authenticates with the application and requests a response from the REST web service. Verify that the web service returns the response that you want.
   Many browsers have plug-ins, such as Boomerang for Google Chrome, that test web service calls to REST clients.
4. If the web service includes a payload, decide how you want to provide that information:
   - You can enter the payload as the value of the **Request payload** property.
   - You can store the payload in a text file that the **CallRESTService** step can access during job processing.
5. Click the **Workflow** tab.
6. Click the name of the workflow you want to modify.
7. **Optional:** Disable the workflow by clicking the switch to the left of the workflow name.
   If you do not disable the workflow while you edit it, jobs that use this workflow continue to move through steps. When you save, the workflow is momentarily disabled then enabled again. Jobs that are processing in the workflow could move into error.
8. Review the steps that are included in the workflow and the default values that they set.
9. Add a step based on the **CallRESTService** step template to the workflow in the appropriate place.
10. Set values for the job properties.
    1. Set the **Request URL** property to the URL of the web service that the step calls.
       The URL can use either the HTTP or HTTPS protocol.
    2. Set the **Request method** property to the value required by the web service.
    3. Set the **Password** property to the password that the step provides to the REST web service to authenticate with the application.
       You enter the password as the value of this property. To provide the password to the REST web service, specify the **Job.WebService.Password** property as a symbol in the **Request payload**, **Request header**, or **Request parameters** property: `${Job.WebService.Password}`. To determine which property to use, refer to the documentation of the web service.
    4. Set the **Request payload type** property:
       - To specify the body of the web services request as the value of the **Request payload** property, select **TEXT**.
       - To specify the body of the web services request in a text file that the **CallRESTService** step can access during job processing, select **FILE**.

       > **Note:**
       > - If you do not specify a payload, RICOH ProcessDirector ignores the value of this property.

    5. If the web service includes a payload, set the **Request payload** property:
       - If you set the **Request payload type** property to **TEXT**, enter the payload as the value of this property.
       - If you set the **Request payload type** property to **FILE**, enter the full directory path, name, and extension of the text file as the value of this property.
         Enter the payload as the content of the text file.

       To send the password in the payload, specify the **Job.WebService.Password** property as a symbol.
       In this XML example, order ID is stored as the value of the **Custom 3** job property (**Job.Info.Attr3**). The payload includes the password as the value of the `<Token>` element and order ID as the value of the `<OrderID>` element:
       `<Token>${Job.WebService.Password}</Token><OrderID>${Job.Info.Attr3}</OrderID>`
       The step resolves the symbols when it calls the REST web service.

6. Set the values for the **Request header** and **Request parameters** properties, as required.
   Each header field or parameter is a keyword/value pair. Each pair must appear on a separate line and must be separated using a colon (:) or equals sign (=).
   The keyword and value can be multiple words. RICOH ProcessDirector uses the first colon or equals sign on each line to split the words into the keyword/value pair.
   This example contains two parameters: `token` and `orderId`. The value of each keyword is a RICOH ProcessDirector symbol. Each symbol contains one of these properties: **Job.WebService.Password** and **Job.Info.Attr3**.
   `token:${Job.WebService.Password}orderId:${Job.Info.Attr3}`
   The step resolves the symbols when it calls the REST web service.
7. If your environment requires a proxy server to communicate with web services, set the **Use proxy** property to the correct proxy server.
8. Set the **Response file** property to the full directory path and name of the file that RICOH ProcessDirector uses to store the response from the application.

   Typically this value is `${getFileName(print,json,write)}` or `${getFileName(print,xml,write)}`, depending on the type of response that the application returns.
   After the response is stored in the file, other steps in the workflow can process the response.

11. When you finish, click **OK**.
12. Save and enable the workflow.

When you finish setting up your workflows, test the exchange of data between RICOH ProcessDirector and the application.

Example

The Web Services Enablement feature includes a supplied **RestfulWebServiceWF** workflow with a **CallRESTService** step.

Parent topic: Preparing to use workflows