Overview / Product overview / Features / Advanced workflow features / Web Services Enablement

Web Services Enablement

This feature lets RICOH ProcessDirector objects and steps use Representational State Transfer (REST) and Simple Object Access Protocol (SOAP) to communicate with web services for applications. The RICOH ProcessDirector implementations of REST and SOAP support Extensible Markup Language (XML). The implementation of REST also supports JavaScript Object Notation (JSON).

The feature adds:

- Input device types
- Notification object types
- Step templates

Input device types

Web Services Enablement provides two types of input devices. REST and SOAP web service input devices communicate with applications by calling web services and retrieving information that RICOH ProcessDirector uses to create jobs.

Notification object types

Web Services Enablement provides two types of notification objects. REST and SOAP web service notifications call web services to update an application when a job or printer event occurs. For example, a notification can update an application when all the items in an order job ship to a customer. A notification also can alert an application when a printing error occurs, or when an input device status changes.

Step templates

Web Services Enablement provides these step templates:

- CallRESTService
- CallSOAPService

Steps based on these step templates let you communicate with applications that provide web service interfaces. These steps call web services from any phase within RICOH ProcessDirector workflows.

Authentication

Web Services Enablement input devices and notifications can communicate with web services for applications that require API key or session authentication. They also can communicate with applications that do not require authentication. Input devices authenticate when they poll for input. Notifications authenticate when they send status to the application.

For API key authentication, you put an authorization code in a **Static credential** property or define an HTTP user ID and password. The object passes the authorization code or the HTTP user ID and password to the web service that exchanges data. The web service then authenticates with the application and returns a response.

For session authentication, you put authentication credentials (user ID and password) and other values in a set of authentication request properties. The input device or notification first calls a REST web service to authenticate with the application. After a successful authentication, the web service returns a token. The input device or notification then transmits the token in the call to the web service that exchanges data.

In this section:

Usage scenario for processing JSON orders with web services

In this scenario, a printing company wants to process orders retrieved from a website for ordering books. Each order consists of 2 job tickets. One job ticket provides information (including the location of the print file) required to print the book. The other job ticket provides information required to print the cover for the book. The book and its cover go through different production processes, and the printing company must report when the whole order is completed. The website provides a REST web services interface.

Parent topic: Advanced workflow features