




WSO2 API Manager 4.1.0 Fundamentals - Integration Profile

Triggering Messages - Part II



WSO2 Training

CC by 4.0



Module Objective

At the end of this module, attendees will be able to:

- Understand how to use Scheduled Tasks to trigger integrations in WSO2 Micro Integrator.
- Understand how to use Inbound Endpoints to trigger integrations in WSO2 Micro Integrator.



Scheduled Tasks

- Configure scheduled jobs that execute internal/external commands.
- Inject a message into a proxy service, main sequence, or a named sequence.



Tasks allow you to configure scheduled jobs in the Micro Integrator and to execute internal and external commands for mediation. The default task implementation injects a message to the Micro Integrator at a scheduled interval. By default, the message goes to the Main sequence but you can also configure it to go to a named sequence or proxy service.

You can also write your own task by creating a custom Java class that implements the Task interface. For example, you could create a task that will read a text file at a specified location and place orders for stocks that are given in the text file.

Working with Scheduled Tasks

Tasks run a piece of code triggered by a timer. This allows users to run scheduled jobs at specified intervals:

- Can use count and interval attributes.
- Can provide the scheduled time as a cron-style entry.
- Can set a task to run right after the server starts up (using just one attribute).

Sample scenario:

On a nightly basis, extract data from a store's sales datasource, transform the data (i.e. data enrichment, or format transformation), and then send a message to a service or a master data repository.



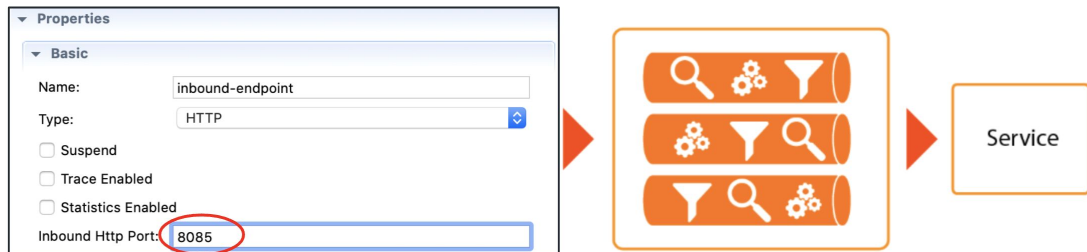
Let's try it out!

Periodic Execution of Integration Processes



Inbound Endpoints

A message source that can be configured dynamically.



Working with Inbound Endpoints

- A message source that can be configured dynamically.
- Behaviour of an inbound endpoint can be:
 - ◉ Listening
 - ◉ Polling
 - ◉ Event based



Listening Inbound Endpoint

- Messages are sent by the client (e.g., HTTP, HTTPS)
- Dynamic configuration



With HTTP Inbound endpoints, you can dynamically change ports on which the client accesses the services.

Polling Inbound Endpoint

- Dynamic configuration
- Polling tasks



Dynamically creates a polling interface. For example, when creating a JMS inbound endpoint, a JMS consumer is created to consume messages from the JMS queue and to inject those messages to the flow. The polling time interval can be configured.

Event-Based Inbound Endpoints

Polls only once to establish a connection with the remote server and then consumes events.

Examples:

- MQTT
- RabbitMQ



Custom Inbound Endpoints

User defined endpoint interfaces. Can be implemented as follows:

- Custom Listening Inbounds
- Custom Polling Inbounds

This is used when you want to implement a new inbound endpoint. This can be implemented in two different ways. For a listening inbound implementation, you need to use the `GenericInboundListener` class, and for a polling inbound implementation, you need to use the `GenericPollingConsumer` class.



Let's try it out!

Using Inbound Endpoints

