Red Hat Integration 2020-Q4

Getting Started with APIs in Red Hat Integration

Getting Started with APIs in Red Hat Integration

Last Updated: 2021-02-07

Red Hat Integration 2020-Q4 Getting Started with APIs in Red Hat Integration

Getting Started with APIs in Red Hat Integration

Legal Notice

Abstract

Use the Task Management API Provider quickstart (booster) to learn how to develop and deploy REST APIs using Red Hat Fuse Online, Apicurito, and 3scale API Management.

Table of Contents

PREFACE 3

CHAPTER 1. OVERVIEW OF RED HAT INTEGRATION COMPONENTS 4

CHAPTER 2. VIEWING, PUBLISHING, AND TESTING THE TASK API QUICKSTART IN FUSE ONLINE 5

2.1. BEFORE YOU BEGIN 5

2.2. OVERVIEW OF THE QUICKSTART 5

2.2.1. Overview of steps for using the Quickstart 5

2.3. DOWNLOADING THE API PROVIDER QUICKSTART FILES 5

2.4. VIEWING THE API DEFINITION IN API DESIGNER (OPTIONAL) 6

2.5. IMPORTING AND PUBLISHING THE EXAMPLE API PROVIDER QUICKSTART INTEGRATION 7

2.6. TESTING THE EXAMPLE API PROVIDER QUICKSTART INTEGRATION IN FUSE ONLINE 9

CHAPTER 3. PUBLISHING, DISCOVERING, AND TESTING THE API SERVICE IN 3SCALE 11

3.1. DISCOVERING THE PUBLISHED API IN 3SCALE 11

3.2. TESTING THE API SERVICE IN 3SCALE 12

3.3. NEXT STEPS 14

PREFACE

Use the Task Management API Provider quickstart (an example that you import) to learn how to develop and deploy REST APIs with Red Hat Integration.

CHAPTER 1. OVERVIEW OF RED HAT INTEGRATION COMPONENTS

For developing and deploying REST APIs, you can use these Red Hat Integration components:

Fuse Online

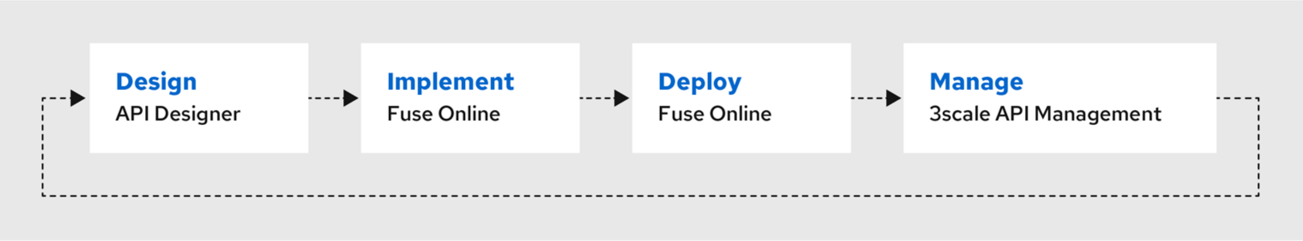
Red Hat Fuse is a distributed, cloud-native integration platform. Fuse Online is Red Hat’s web-based Fuse distribution. It is pre-installed on the OpenShift Online Professional tier. You can also install it on an OpenShift (on-premise) cluster. Fuse Online is for business users who prefer minimal code development.

API Designer

Red Hat provides a lightweight version of the API designer. You can use it to create API definitions in OpenAPI (Swagger) format. You can access the API Designer from within the Fuse Online API provider.

3scale API Management

Red Hat 3scale API Management manages security, performance, control, and monetization for your APIs.



66_RHI_0220

For more details, see Developing and Deploying API Provider Integrations .

CHAPTER 2. VIEWING, PUBLISHING, AND TESTING THE TASK API QUICKSTART IN FUSE ONLINE

2.1. BEFORE YOU BEGIN

Before you can use Red Hat Integration to run the Task Management API Provider quickstart integration, make sure that your OpenShift environment meets the following configuration requirements:

- Your OpenShift Container Platform (OCP) on premise administrator must deploy Fuse Online and 3scale API Management on the same cluster.

- The 3scale API Management administrator must configure 3scale for service discovery .

- 3scale discovery is enabled as described in Configuring Fuse Online to enable 3scale discovery of APIs. When 3scale discovery is enabled, Fuse Online automatically adds the necessary 3scale annotations to the API.

- You can log into the Fuse Online console and the 3scale Admin Portal.

2.2. OVERVIEW OF THE QUICKSTART

In Red Hat Integration, a quickstart is an example integration that you can import into your Fuse Online environment.

The Task Management API Provider quickstart helps you to quickly learn how to configure, publish, and test an API provider integration. It simulates an API that sales consultants might use to track the tasks that they must complete when interacting with customer contacts. Some example "to-do" tasks include "create an account for a new contact" or "place an order for an existing contact". The API provider integration stores the task data in Fuse Online’s **PostgresDB** sample database.

2.2.1. Overview of steps for using the Quickstart

1. Download the Task Management API Provider quickstart files.

2. (Optional) View the API definition in API Designer by importing the quickstart OpenAPI JSON file.

3. Import the quickstart ZIP file into Fuse Online and then publish it.

4. Test the published Task Management API by invoking **curl** commands.

5. Enable 3scale discovery and then discover the published API service in 3scale API Management.

6. Test the API service in 3scale by invoking **curl** commands.

2.3. DOWNLOADING THE API PROVIDER QUICKSTART FILES

You must download the Task Management API Provider quickstart files so that you can import them into Fuse Online.

Go to https://github.com/syndesisio/syndesis-quickstarts/tree/1.11/api-provider and then download these files:

task-api.json (raw)

The Task Management API definition in OpenAPI format. You view this file in the API Designer to explore the API structure. The API definition is a starting point for an API Provider integration.

> NOTE
>
> You must download the raw version of this file.

TaskAPI-export.zip

The Task Management API Provider integration. This file contains the complete API Provider integration (based on the API definition in the **task-api.json** file). You import this file into Fuse Online, edit it in the Fuse Online integration editor, and then publish it as an API service that 3scale can discover.

2.4. VIEWING THE API DEFINITION IN API DESIGNER (OPTIONAL)

You can use the API Designer to create new API definitions in OpenAPI format. You can also use the API Designer to view and edit existing API definitions. You can use an API that you create or import into the API Designer as a starting point for building a Fuse Online integration.

This task is optional because you use the API Designer to peruse the Task Management API paths, operations, and data type (that is, you are not creating or editing the API specification). After you become familiar with the API specification, exit the API Designer and return to Fuse Online to import the pre-built integration that is provided in the other quickstart file (**TaskAPI-export.zip**).

Prequisites

- You are familiar with REST APIs.
- You downloaded the raw version of the **task-api.json** file as described in Section 2.3, “Downloading the API Provider quickstart files”.
- You are logged into the Fuse Online dashboard.

Procedure

Follow these steps to view the API definition in the API Designer:

1. In Fuse Online, in the left navigation panel, click **Integrations**.
2. Click **Create Integration**.
3. On the **Choose a connection** page, click the **API Provider** card.
4. Drag the raw version of the **task-api.json** file that you downloaded to the **Upload an OpenAPI file** section. Fuse Online displays a message that it has successfully imported the file.
5. Click **Next**.

   > NOTE
   >
   > You can safely ignore the warning message.

6. Click **Review/Edit**.

The API opens in API Designer.

7. Explore the structure of the API. It includes the following components:
   - The / path with these operations:
     - **GET** - Get a list of all tasks
     - **POST** - Create a new task
   - The /**{id}** path with these operations:
     - **GET** - Fetch a task by its ID
     - **PUT** - Update a task by its ID
     - **DELETE** - Delete a task by its ID
   - The **Task** data type with three properties:
     - **completed** (**integer**, **optional**)
     - **id** (**integer**, **required**)
     - **task** (**string**, **required**)
8. To save time, rather than building the integration from this OpenAPI specification, exit the API Designer and return to Fuse Online so that you can import the entire pre-built integration that is provided in the other quickstart file (**TaskAPI-export.zip**) that you downloaded in Section 2.3, “Downloading the API Provider quickstart files”.
   To exit the API Designer, click **Cancel** three times and then **Confirm** the cancellation until you return to the Fuse Online console.

2.5. IMPORTING AND PUBLISHING THE EXAMPLE API PROVIDER QUICKSTART INTEGRATION

The **TaskAPI-export.zip** file contains the Task Management API Provider integration. It is based on the API definition in the **task-api.json** file.

After you import the quickstart integration, you can examine the operation flows and then publish the integration.

Prerequisites

- Fuse Online is open in a browser.
- The Fuse Online environment must include the **Todo** sample app as described in Adding a sample app to a Fuse Online environment running on OCP.

Procedure

1. Import the **Task API** quickstart integration:
   a. In Fuse Online, in the left navigation panel, click **Integrations**.
   b. In the upper right, click **Import**.

   c. Drag the **TaskAPI-export.zip** file that you downloaded to the **Import** page. Fuse Online indicates that it has successfully imported the file.

   d. In the left navigation panel, click **Integrations** to see an entry for the **Task API** integration that you just imported. Although the entry indicates that configuration is required, this integration is ready to publish.

2. Click **Edit Integration** to view the list of the operations that this API provides.
   Each operation has a flow defined for it. In Fuse Online, a *flow* defines the connections and other steps to execute for each REST operation in the integration. The imported quickstart provides pre-defined flows for each operation.

3. To examine the flows for each operation:

   a. Click its **Edit flow** button to display the visualization for that flow.
      Each flow already has a database connection, one or more data mapper steps, and a **Provided API Return Path**step that finishes the flow.

   b. For the **Invoke SQL** step, click **Configure** to see the SQL statement that the connection executes. Then click **Cancel** to return to that operation's visualization flow.

   c. For a data mapper step, click **Configure** to see the mappings. Then click **Cancel** to return to the visualization.

   d. For the **Provided API Return Path**step, which is the last step in every operation's flow, click **Configure** to see the HTTP return codes that the operation might send to the caller. Click **Cancel** to return to the visualization.

   e. After examining one operation's flow, click the **Integrations**> **Task API**> **Operation** drop down menu and then select another operation.

   f. Repeat this subset of steps to examine each flow.

4. Click **Save** and edit the integration name if you want to.

5. Disable 3scale discovery initially, so that you can test the API before you publish it to 3scale:

   a. Click **Cancel** and confirm that you want to exit from editing the integration.

   b. Click **View** to open the Integration's summary page.

   c. If there is a **Disable Discovery** button, click it to disable 3scale discovery. If the button label is **Enable discovery**, continue to the next step.
      Note: If you do not see an **Enable Discovery** or **Disable Discovery** button on the summary page, it means that the OpenShift administrator has not configured service discovery.

   d. Click **Publish** and then confirm that you want to start the integration.
      The summary page for the integration shows publication progress as it assembles, builds, deploys, and starts the integration.

6. When the **Task API** integration summary page displays **Running**, Fuse Online displays the external URL for the Task API service. It looks something like this:
   https://i-task-api-fuseonline.apps.openshift.com/

   This URL is where Fuse Online makes the Task API service available. REST API calls specify URLs that start with this base URL.

2.6. TESTING THE EXAMPLE API PROVIDER QUICKSTART INTEGRATION IN FUSE ONLINE

When the Fuse Online **Task API** quickstart integration is running, you can invoke **curl** utility commands that send HTTP requests to the Task API service.

Prerequisites

- Fuse Online indicates that the **Task API** integration is **Running**.
- You disabled discovery for the **Task API** integration.

Procedure

1. Copy the integration's external URL.
2. In a terminal, invoke a command such as the following to assign the integration's external URL to the **externalURL** environment variable. Be sure to replace the URL in this sample command with the URL that you copied.

```
export externalURL="https://i-task-api-fuseonline.apps.openshift.com"
```

3. Invoke a **curl** command that triggers execution of the flow for the **Create new task** operation:

```
curl -k --header "Content-Type: application/json" --request POST --data '{"id":1,
"task":"Create an account for new customer"}' $externalURL
```

- **-k** allows **curl** to proceed and operate even for server connections that are otherwise considered insecure.
- **--header** indicates that the command is sending JSON format data.
- **--request** specifies the HTTP **POST** command, which stores data.
- **--data** specifies the JSON format content to be stored. In this example, the content is **{"id":1, "task":"Create an account for new customer"}**. Note that the **completed** field is not required.
- **$externalURL** is the URL to invoke.

This command sends an HTTP **POST** request to the Task API service, which triggers execution of the **Create new task** operation's flow. Flow execution adds a new task to the sample database and returns a message such as the following to indicate what it did:

+

```
{"completed":null,"id":1,"task":"Create an account for new customer"}
```

4. Invoke the following **curl** commands to create two more tasks:

```
curl -k --header "Content-Type: application/json" --request POST --data '{"id":2, "task":"Place
order for customer"}' $externalURL
```

```
curl -k --header "Content-Type: application/json" --request POST --data '{"id":3, "task":"Send
confirmation email to customer' $externalURL
```

5. Invoke a **curl** command that triggers execution of the flow for the **Fetch task** by ID operation:

   ```
   curl -k $externalURL/1
   ```

   To obtain a task, the **curl** command needs to specify only the URL. The HTTP **GET** command is the default request. The last part of the URL specifies the ID of the task to get.

6. Invoke the following **curl** command that triggers execution of the flow for the **Fetch all tasks** operation:

   ```
   curl -k $externalURL
   ```

7. Optionally, invoke a **curl** command that triggers execution of the flow for the **Delete task** for ID operation:

   ```
   curl -k -X DELETE $externalURL/3
   ```

   This command invokes the HTTP **DELETE** command with the same URL as the command that obtained a task by its ID.

8. Optionally, you can view the **PostgresDB** sample database table contents:

   a. View a list of the Fuse Online pods to obtain the sample database podname:

   ```
   oc get pods
   ```

   b. Type the following command, replacing **syndsis-db-1-xxxxx** with your sample database podname.

   ```
   oc rsh syndesis-db-1-xxxxx
   ```

   c. Access the sample database and view the **todo** table:

   ```
   sh-4.2$ psql -Usampledb

   sampledb=> select * from todo;
   ```

   You should see a result similar to the following:

   ```
   id |          task                      | completed
   ----+-----------------------------------+-----------
    3 | Send confirmation email to customer |           |
    2 | Place order for customer           |     1
    1 | Create an account for new customer  |
   (3 rows)
   ```

CHAPTER 3. PUBLISHING, DISCOVERING, AND TESTING THE API SERVICE IN 3SCALE

3.1. DISCOVERING THE PUBLISHED API IN 3SCALE

After you publish an API service in Fuse Online with 3scale discovery enabled, you can discover it in the 3scale API admin portal.

Prerequisites

- The Fuse Online API Integration is deployed on the same OpenShift cluster where 3scale is installed.
- You know the API's service name and its namespace (OpenShift project).
- Your 3scale user or service account (depending on the configured authentication mode) has the necessary privileges to view the API service (for example **i-task-api**) and its namespace.
- 3scale discovery is enabled as described in Configuring Fuse Online to enable 3scale discovery of APIs. When 3scale discovery is enabled, Fuse Online automatically adds the necessary 3scale annotations to the API.

Procedure

1. Enable discovery for the Fuse Online API integration:
   a. In Fuse Online, select **Integrations** and then click **View** for the Task API integration.
   b. On the integration's summary page, click **Enable Discovery**.
   c. Click **Publish** and then confirm that you want to start the integration.
      The summary page for the integration shows the publication progress as it assembles, builds, deploys, and starts the integration.
   d. Wait until the **Task API** integration summary page displays **Running**.
2. To discover the API in 3scale:
   a. Log in to the 3scale Administration Portal
   b. In the Admin Portal's **Dashboard**, under the **APIs** section, click **New Product**.
   c. Choose **Import from OpenShift**.
   d. In the **Namespace** field, the OpenShift project that contains the API, for example **fuseonline**.
   e. In the **Name** field, select the name of the API service within that namespace, for example **i-task-api**.
   f. Click **Create Product**. (In 3scale, a *product* is a customer-facing API that also includes an internal *backend* API).
      A page for the **fuseonline** product opens.

3.2. TESTING THE API SERVICE IN 3SCALE

When you import a discoverable service, 3scale generates a new customer-facing API product (or service). It also generates the API service's corresponding internal backend URL by using the Fuse Online namespace.

Before you can promote the API service to the 3scale staging area for testing, you must do the following tasks in 3scale:

- Create an application plan that establishes the rules (access rights, limits, pricing, features) for using the API service.
- Create an application that subscribes to the plan. An application accesses the API within the constraints of the application plan. The application provides credentials for making calls to the API.
- Create mapping rules for calls to the API. Mapping rules define the metrics or methods that you want 3scale to gather and report. For example, a mapping rule might track how many GET operations are received by the API service.

Prerequite

- You have discovered the API in 3scale.

Procedure

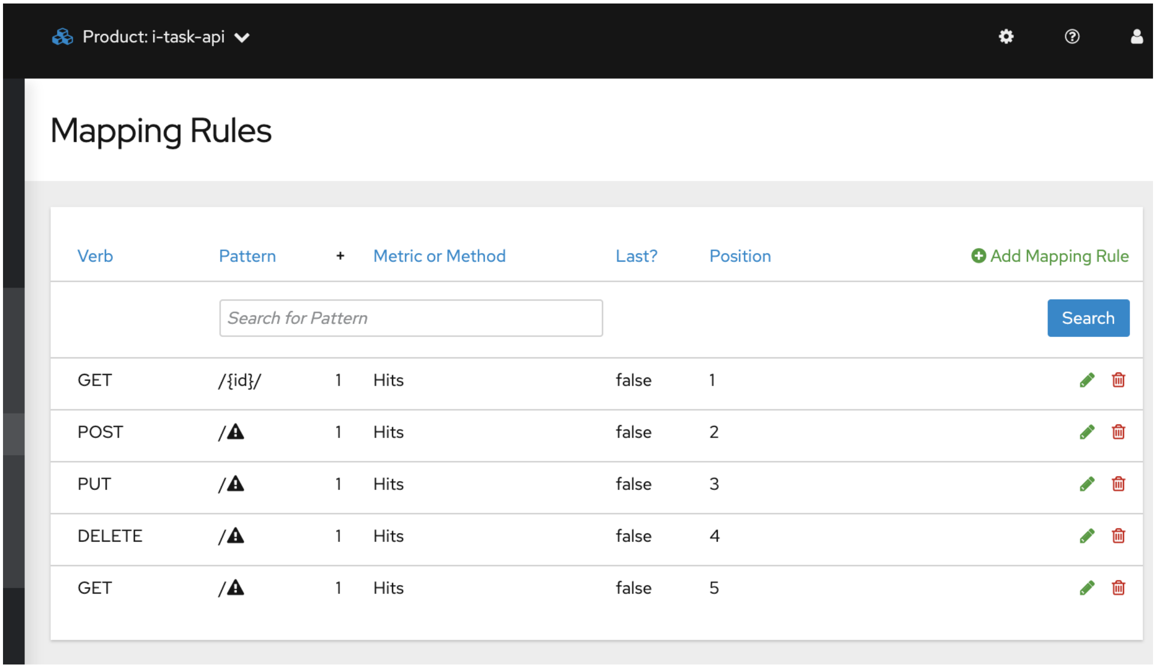
1. Create an application plan:
   a. Click **Create** application plan and in the **Name** field, type **Basic Task Plan**. For the purposes of this quickstart example, you can leave the other fields blank.
   b. Click **Create application plan**.
   c. Click **Publish**.
2. Create an application and associate it with the application plan:
   a. From the top menu, select **Audience**.
   b. Click **Developer** for the **John Doe** default user.
   c. Click **Applications** and then **Create Application**.
   d. Click inside the **Application** Plan field to open the drop-down selection list of application plans and then click the **Basic Task Plan** under **i-task-api**.
   e. In the **Name** field, type a name, for example, **Task App**.
   f. In the **Description** field, type some text for example: **The Fuse Online Task quickstart application.**
   A summary page for the Task App appears. It includes the **User Key** value that you will use when running test **curl** commands in a later step.
3. Set up mapping rules for the API service:
   a. From the top menu, select **i-task-api**.

b. Under **Integration**, select **Mapping Rules**.
   Note: In 3scale, *integration* has a different meaning than in Fuse Online. In 3scale, *integration* refers to how an API integrates with 3scale, for example, to manage and secure client access to the API.

c. Add a mapping rule:

   - Click **Add mapping rule**.
   - Select a **Verb**, for example **GET**.
   - Type a **Pattern**, for example / or /**{id}**/
   - Click **Create Mapping Rule**.

d. Repeat substep **c** until you have defined the mapping rules shown here:



Mapping Rules

Verb	Pattern	+	Metric or Method	Last?	Position	⊕ Add Mapping Rule
	Search for Pattern					Search
GET	/{id}/	1	Hits	false	1	
POST	/⚠	1	Hits	false	2	
PUT	/⚠	1	Hits	false	3	
DELETE	/⚠	1	Hits	false	4	
GET	/⚠	1	Hits	false	5	

4. To update the staging environment for the **i-task-api** Product:

   a. Select **Integration** > **Configuration**.

   b. Click **Promote v.x to Staging APIcast**

5. Use the staging environment to make **curl** calls to test the API operations. An example is provided for you in the **Staging APIcast** section with this format:

```
curl -k "https://fuseonline-i-task-api-3scale-apicast-
staging.apps.test.lab.upshift.rdu2.redhat.com:443/{id}/?
user_key=4466490a15df83a6b36d2838fb19a030"
```

Note: You replace **{id}** with a valid value. When you tested the API in Fuse Online, you created three tasks that are stored in the sample database. Therefore, you can replace **{id}** with **1**, **2**, or **3** (if you did not delete the third task).

Here are some example **curl** commands that you can try:

- To make the commands easier to read, you can create variables to hold the **URL** and **user key**:

Note You must replace the URL and user key values with values that are specific to your 3scale staging environment:

```
export threescaleURL=”https://fuseonline-i-task-api-3scale-apicast-staging.apps.test.lab.upshift.rdu2.redhat.com:443”

export KEY=”4466490a15df83a6b36d2838fb19a030"
```

- Obtain a task by ID 1:

```
curl -k GET $threescaleURL/1/?user_key=$KEY
```

- Update a task by ID 1:

```
curl -k --header "Content-Type: application/json" --request PUT --data '{"id":1,
"task":"Update to task 1 - to do by Friday"}' $threescaleURL/1/?user_key=$KEY
```

- Create a new task:

```
curl -k --header "Content-Type: application/json" --request POST --data '{"id":4,
"task":"Check the staus of the customer order"}' $threescaleURL/?user_key=$KEY
```

- Get all tasks:

```
curl -k $threescaleURL/?user_key=$KEY
```

- Delete a task:

```
curl -k -X DELETE $threescaleURL/4/?user_key=$KEY
```

6. To see how 3scale tracks the API interactions, click **Analytics**> **Traffic** and view the metrics that 3scale gathered based on the mapping rules that you defined in Step 3.

3.3. NEXT STEPS

See the Red Hat 3scale API Management documentation for information about managing the API.