Date :  13-Oct-23

Flight Data (1.0.0)

# Overview

## Real-time data that offers detailed information about flights happening worldwide, including the status, location, altitude, destination, and more. This comprehensive data source allows for the live tracking and monitoring of flights on a global scale.

# Getting Started

## Prerequisites

Before you start using the Flight Data data adapter, ensure you have the following:

* [HCL Foundry](https://manage.hclvoltmx.com/)
* Volt MX Iris
* An account with [Rapid API](https://rapidapi.com/ai-box-ai-box-default/api/flight-data4)

## Platforms Supported

### Mobile

#### iOS

#### Android

### Tablets

### PWA

## Importing the adapter

 **To import the Data Adapter to Volt Foundry, do the following:**

1. Sign in to the  [HCL Foundry](https://manage.hclvoltmx.com/).
2. From the left navigation menu, select **API Management**.
3. In **API Management**, select **Custom Data Adapters**.

4. Click **IMPORT** to import a custom data adapter.

5. On the Import Data Adapter dialog box, click browser to import.

6. Select FLIGHT DATA data adapter zip file and click **IMPORT**.

After importing the data adapter, Volt Foundry opens a window that shows the metadata of the data adapter.



After you import the data adapter, you can view it on the Custom Data Adapters page and use it to create services on Volt Foundry.

After you add the data adapter to your Volt Foundry app. You can refer to the section below for more information.

**Follow the given steps to generate API key.**

1.Go to [Flight Data API portal](https://rapidapi.com/ai-box-ai-box-default/api/flight-data4)

2.Create an account with your email

3.Click on **Subscribe for Free** and get API key.

**Creating an Integration service**

Follow the given steps to create an Integration service using the FLIGHT DATA Data Adapter.

1. Sign in to the [HCL Foundry](https://manage.hclvoltmx.com/).
2. From the left navigation menu, select **API Management**.
3. In **API Management**, select **Integration**.



1. To create a new service, click the **+** button or the **CONFIGURE NEW** button.



1. On the Service Definition tab, select the service type as FLIGHT DATA, and click **SAVE**.



******Creating and Executing operations**

After you create an integration service, you can create and execute operations using the service.

#### Creating an operation

* In **the API Management/Foundry app you created**, in the **Integration** section, select the service that you created.
* After you select the service, navigate to the **Operation List** tab.

* From the drop-down list, select an operation that you want to execute, and click **ADD OPERATION**.



#### Executing an Operation

* From the **Operations List** tab, in the **Configured Operations** section, select the operation you want to execute.



* On the Operation Page, in the Request Input tab, enter a TEST VALUE for all the fields.



* Select a run-time environment and click **Save and Fetch Response** to get a response based on your inputs.



## Publishing your application

If you want to use the services in client applications, you need to publish an app to a run-time environment. You can create the service (as described above) in an application or import the service into an application and publish the application.

# References

## **Endpoint Documentation**

| **Endpoints** | **Input params-Type** | **Description** |
| --- | --- | --- |
| /get\_airline\_flights | airline | Return list of live flights by the given airline code. |
|  / get\_area\_flights | latitudelongitude | Return all flights that within 500 km in radius by the given coordinates. |
| /get\_flight\_info | flight | Return live flight information for given IATA number. |

Top of Form

## These are List of all endpoints, input params of the fields in tabular format.

## Refer [this link](https://rapidapi.com/ai-box-ai-box-default/api/flight-data4) for more information about API endpoints.

# Revision History

Adapter version 1.0.0

## Limitations

Occasionally, when the response doesn't come as expected, refreshing the payload can fetch the response successfully.