Mulesoft Anypoint Platform

Application Life Cycle in Anypoint Studio

This article has been generated from the online version of the catalog and might be out of date. Please, make sure to always refer to the online version of the catalog for the up-to-date information.

Auto-generated at January 7, 2026

Overview

Anypoint Studio is Mulesoft's Eclipse-based integration development environment that is used for designing and testing your Applications.

Anypoint Studio provides the following features for building your Applications:

- Instant run of your Mule application inside a local runtime.
- Visual editors to configure API definition files and Mule domains.
- Push changes to a local running app to test changes.
- Integration with Exchange to import templates, examples, definitions and other resources from your Anypoint Platform organization.
- Embedded unit testing framework.
- Built-in support to deploy to CloudHub.

Before following the provided instructions make sure to install Anypoint Studio and deploy the Connector.

Building Applications

Anypoint Studio helps you design and edit your application via the Mule configuration editor, API definitions, properties and APIs, and the `pom.xml` file that handles dependencies.

The Mule configuration editor is opened once you double-click on the application XML file via the following path: `/src/main/mule/`. This editor provides a graphical user interface that allow designing and configuring your application.

Adding Operations

The editor contains a canvas that provides users with visual representations of their selected flows. The editor also allows adding required *event processors* to the canvas as described below:

1. Select the deployed Connector and open the *Mule Palette*.
2. Choose required operation from the Palette.
3. Drag and drop your selected operation to the canvas.

Therefore, you will successfully add your selected processors to your flow.

Endpoint and Token

Specify **Global Element Properties** for the *CloudBlue Connect Config* as follows:

- **Endpoint**: Enter the Connect API endpoint.
- **Token**: Provide your Connect token API Key/Handle.
- **Connection timeout**: Specify required connection timeout.

Click **Test Connection** to test your specified connection properties. Click **OK** to save your adjustments.

Input Data

Depending on your selected operation, it is necessary to configure specific *input data*. This includes filters, product IDs, and so on.

Once all required input data is specified and your application is complete, you can convert your project to a deployable archive and deploy your application as described below.

Export a Studio Project

Convert your designed application to a JAR package by using a wizard provided by Anypoint Studio. Therefore, your created package can be deployed to on-premisis infrastructure or to Anypoint CloudHub.

1. Select **Export** from the **File** menu.
2. In the Import wizard, click to expand the Mule folder, select **Anypoint Studio Project to Mule Deployable Archive (includes Studio metadata)**, and click **Next**.
3. Select the project that you want to export and click **Next**.
4. In the **JAR file** menu, click the ellipses (…) button to explore your local drive, and select the folder to which you want to export your deployable JAR file.

Note that you can choose **Attach Project Sources** to include metadata that Studio requires to reimport the deployable file as an open Mule project into your workspace.

Deploying to CloudHub

The following steps showcase how to deploy your mule deployable archive to Anypoint CloudHub:

1. Right click the project folder on your **Package Explorer View**.
2. Select **Anypoint Platform** and then **Deploy to CloudHub**.
   If you don't have your Anypoint Platform credentials configured in Anypoint Studio, you'll need to provide your log in information before continuing.
3. Configure the **CloudHub** instance to which your application is being deployed.

Thus, Anypoint Studio uses your package name as the name for your application on CloudHub. This name is part of the domain of your deployed application. For example, naming your application "deployment-1" makes your application's public domain

"deployment-1.cloudhub.io".

You can also configure the runtime version. By default, Anypoint Studio uses the application version that you are using in your project during development. To reduce compatibility risks, you should always deploy to the runtime version in which your Mule project is created or the closest alternative. In addition, you can configure the amount of worker and worker size for the CloudHub instance you are going to use, enable Insight, static IPs, configure log levels, and properties placeholders for your application.