

Cloud Studio



Jitterbit Cloud Studio

Cloud Studio

Overview

Jitterbit Harmony Cloud Studio is the next-generation, web-based version of Jitterbit's project design application.

Accessible directly through the [Jitterbit Harmony Portal](#), Cloud Studio offers a modern design experience optimized for ease-of-use and ease-of-access by integration designers.

Prerequisites

Cloud Studio is accessed through the [Harmony Portal](#) and does not require any additional hardware or software installation. The only requirements are browser and Jitterbit Harmony Agent version requirements.

Browser Requirements

These major browsers are supported:

- Chrome
- Firefox
- Safari (macOS only)

In order to communicate with the Harmony cloud and access the Harmony Portal, JavaScript needs to be enabled and outbound port 443 (HTTPS) needs to be open. This port is normally allowed by corporate server firewalls.

Agent Requirements

The minimum Jitterbit Harmony Agent version supported with Cloud Studio is [version 9.4.2](#).

Certain Cloud Studio features require an agent [version 10.0](#) or higher, including these:

- Custom connectors built with [Connector Builder](#)
- Custom connectors built with the [Connector SDK](#)

Reference

Cloud Studio reference documentation is organized into sections by topic:

- **Introduction:** This section includes a quick start guide and tutorials, description of the user interface, and terminology and definitions.
- **Projects:** A project is a collection of one or more workflows that comprise and execute an integration use case.
- **Project Components:** Project components are the discrete building blocks of a project.
- **Workflows:** A workflow is a collection of operations used as a tool for your convenience to help segregate different parts of the project.
- **Operations:** An operation is the smallest unit that is independently executed on an agent and recorded by Harmony. Operations are used to define what an integration should do and when it should be done.
- **Connectors:** Connectors provide the interface for entering user-provided input such as credentials to create an authorized connection. Activities associated with those connections can then be added to operations on the design canvas and configured as sources or targets. An endpoint refers to a specific connection and its activities.
- **Transformations:** Transformations are used as steps in an operation to map or *transform* inputs to a resulting output by moving data, cleaning data, or applying business logic. A transformation consists of source and target schemas that have been defined in the transformation and the transformation mapping that generates the output.

On This Page

- [Overview](#)
- [Prerequisites](#)
 - [Browser Requirements](#)
 - [Agent Requirements](#)
- [Reference](#)

Pages in This Topic

- [Introduction](#)
- [Projects](#)
- [Project Components](#)
- [Workflows](#)
- [Operations](#)
- [Connectors](#)
- [Connector Builder](#)
- [Transformations](#)
- [Schemas](#)
- [Scripts](#)
- [Functions](#)
- [Variables](#)
- [Notifications](#)
- [Plugins](#)
- [Recipes](#)

Search in This Topic

Related Topics

- [Connector Builder](#)
- [Connector SDK](#)
- [Design Studio](#)
- [Getting Started](#)

Last updated: Sep 23, 2019

- **Schemas:** Source and target schemas represent the "request" and "response" structures for interaction with a data resource in an operation. These schemas can be either defined within the transformation or provided by an adjacent activity.
- **Scripts:** Scripts provide the flexibility and power to transform data, perform calculations, or perform logic validation beyond simple field mapping. Scripts can be used both as steps of an operation as well as within transformations to apply specified logic or conditions to the data.
- **Functions:** Functions are used within scripts in operations and transformations to enhance and refine data processes.
- **Variables:** Variables are used to allow for the dynamic configuration of endpoints, to support passing of data between operations, and to drive detailed integration logic within transformation scripts.
- **Notifications:** Notifications that send a customized email can be triggered upon success or failure of an operation or called from a script.
- **Plugins:** Plugins are Jitterbit- or user-provided applications that extend Harmony's native capabilities. Jitterbit provides a selection of plugins or you can create your own using Jitterbit's Plugin SDK.
- **Recipes:** A Cloud Studio recipe is an exported Cloud Studio project that serves as a reusable template for starting an integration and demonstrates basic functionality between two endpoints.