

Google Sheets Update Sheets Activity

Introduction

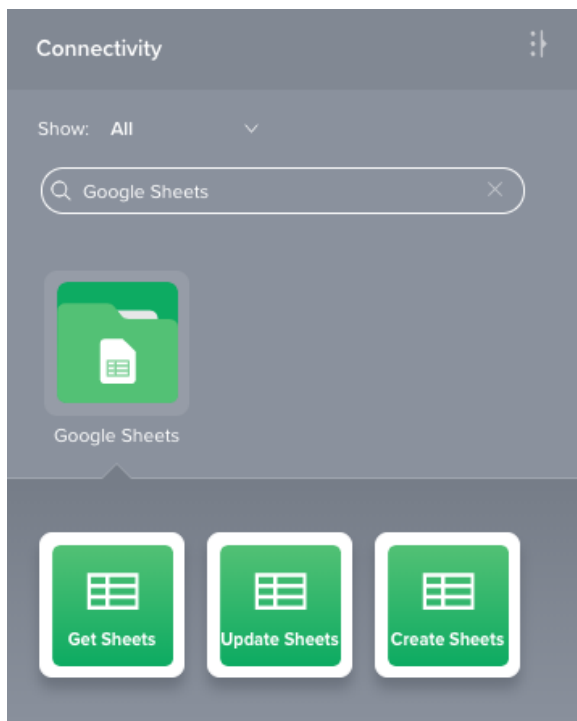
A Google Sheets update sheets activity updates a spreadsheet at a Google Sheets connection and is used as a target in an operation. After configuring a Google Sheets connection, you can configure as many Google Sheets activities as you like for each Google Sheets connection.

Creating a Google Sheets Activity

From the design canvas, open the **Connectivity** tab of the design component palette:



Within the **Endpoints** filter, click the Google Sheets connection block to display activities that are available to be used with a Google Sheets connection:



To create an activity that can be configured, the activity must first be added to an operation on the design canvas. To add an activity to an operation, drag the activity block from the palette to the operation.

For more information about the parts of an operation and adding activities to operations, see [Operation Creation and Configuration](#).

Accessing Menu Actions

After a Google Sheets activity has been added to an operation, menu actions for that activity are accessible from the project pane in either the **Workflows** or the **Components** tabs, and from the design canvas:

- **Project Pane:** In the **Workflows** or **Components** tab of the project pane, hover over an activity name and click the actions menu icon to open the actions menu.
- **Design Canvas:** Within the operation, click an existing activity block to open the actions menu.

Each of these menu actions is available:

- **View/Edit:** This opens the activity configuration screen for you to configure the activity. For details, see *Configuring a Google Sheets Update Sheets Activity* later on this page.
- **Delete:** This is used to permanently delete the activity.

On This Page

- [Introduction](#)
- [Creating a Google Sheets Activity](#)
- [Accessing Menu Actions](#)
- [Configuring a Google Sheets Update Sheets Activity](#)
 - [Step 1 – Enter Basic Information](#)
 - [Step 2 – Review Data Schema](#)
- [Next Steps](#)

Related Articles

- [Component Dependencies, Deletion, and Removal](#)
- [Component Deployment](#)
- [Component Palette](#)
- [Design Canvas](#)
- [Google Sheets Connection](#)
- [Google Sheets Create Sheets Activity](#)
- [Google Sheets Get Sheets Activity](#)
- [Google Sheets Registration](#)
- [Operation Creation and Configuration](#)
- [Operation Options](#)
- [Operation Validity](#)
- [Plugins Added to an Activity](#)
- [Project Pane](#)
- [Transformation Creation and Configuration](#)

Related Topics

- [Cloud Studio](#)
- [Connectors](#)
- [Google Sheets](#)
- [Operations](#)
- [Scripts](#)
- [Variables](#)

Last updated: Dec 20, 2019


- **Rename:** This positions the cursor on the activity name in the project pane for you to make edits.
- **View Dependencies:** This changes the view in the project pane to display any other parts of the project that the activity is dependent on.
- **Remove:** Available only from the actions menu on the design canvas, this removes the activity as a step in the operation without deleting it from the project. When you remove an activity that is adjacent to a transformation, if schemas are provided within that activity, they will no longer be referenced by the transformation. Removed components can be accessed or permanently deleted from the project pane.
- **Deploy:** This deploys the activity and any components it is dependent on.
- **Configurable Deploy:** This opens the deployment screen, where you can select project components to deploy.
- **Add to group:** This opens a prompt to create a new custom group or to add the component to an existing group.
- **Duplicate:** This creates a copy of the activity as a new, unreferenced component. Upon creating the component copy, the cursor is positioned on the component name within the project pane for you to rename the component.

Configuring a Google Sheets Update Sheets Activity

Follow these steps to configure a Google Sheets update sheets activity:

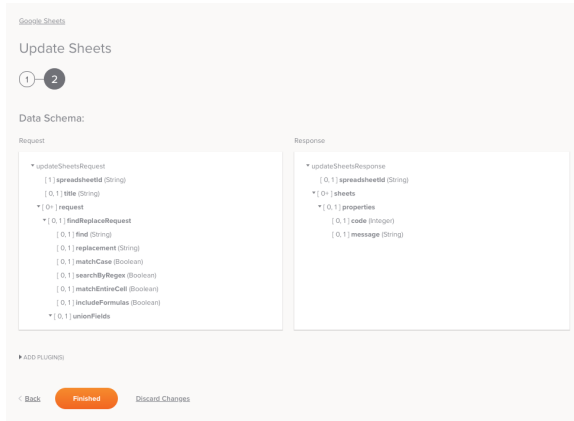
- Step 1 – Enter Basic Information
- Step 2 – Review Data Schema

Step 1 – Enter Basic Information

TIP Fields with a variable icon  support using global variables, project variables, and Jitterbit variables. Begin typing an open square bracket [into the field or click the variable icon to display existing variables to choose from.

- **Name:** Enter a name to use to identify the Google Sheets update sheets activity. The name must be unique for each update sheets activity and must not contain forward slashes (/) or colons (:).
- **Number of Columns:** Enter the number of columns to be used when creating the spreadsheet (optional); if not provided, the default value of 26 is used.
- **Save & Exit:** If enabled, click to save the configuration for this step and close the activity configuration.
- **Next:** Click to temporarily store the configuration for this step and continue to the next step. The configuration will not be saved until you click the **Finished** button on the last step.
- **Discard Changes:** After making changes, click **Discard Changes** to close the configuration without saving changes made to any step. A message will ask you to confirm that you want to discard changes.

Step 2 – Review Data Schema



- Data Schema:** The request and response data schemas from Google Sheets will be displayed. If the operation uses a transformation, the data schemas will be displayed again later during the transformation mapping process, where you can map to target fields using source objects, scripts, variables, custom values, and more.

The Google Sheets connector uses the [Google Sheets API v4](#). Refer to the API documentation for information on the schema fields and practices for manipulating Google Sheets.

• **Request:**

Request Schema Field /Node	Google API Field /Method, type	Notes
updateSheetsRequest	spreadsheets.update	Request to update a spreadsheet
spreadsheetId	spreadsheetId	ID of the spreadsheet to be updated
title	title	Sheet title
request	node	Each request is one of findReplaceRequest , addSheetRequest , updateSheetRequest , deleteSheetRequest , deleteCellRequest , or updateCellRequest
findReplaceRequest	FindReplaceRequest	Finds and replaces data in cells over a range, sheet, or all sheets
find	String	The value to search
replacement	String	The value to use as the replacement
matchCase	Boolean	True if the search is case sensitive
searchByRegex	Boolean	True if the find value is a regex. The regular expression and replacement should follow Java regex rules .
matchEntireCell	Boolean	True if the find value should match the entire cell
includeFormulas	Boolean	True if the search should include cells with formulas. False to skip cells with formulas.
unionFields	node	Union field scope: The scope over which to find /replace; one of SheetID , allSheets , or gridRange must be set
SheetID	Integer	The sheet to find/replace over
allSheets	Boolean	True to find/replace over all sheets
gridRange	GridRange	A range on a sheet
SheetID	Integer	The sheet this range is on

startRowIndex	Integer	The start row (inclusive) of the range, or not set if unbounded
endRowIndex	Integer	The end row (exclusive) of the range, or not set if unbounded
startColumnIndex	Integer	The start column (inclusive) of the range, or not set if unbounded
endColumnIndex	Integer	The end column (exclusive) of the range, or not set if unbounded
addSheetRequest	AddSheetRequest	Adds a new sheet
SheetID	String	The ID of the sheet; must be non-negative; this field cannot be changed once set
title	String	The name of the sheet
index	Integer	The index of the sheet within the spreadsheet
hidden	Boolean	True if the sheet is hidden in the UI, false if it's visible
rightToLeft	Boolean	True if the sheet is an RTL (right-to-left language) sheet instead of an LTR (left-to-right language) sheet
gridProperties	GridProperties	Properties of a grid
rowCount	Integer	The number of rows in the grid
columnCount	Integer	The number of columns in the grid
frozenRowCount	Integer	The number of rows that are frozen in the grid
frozenColumnCount	Integer	The number of columns that are frozen in the grid
hideGridLines	Boolean	True if the grid isn't showing gridlines in the UI
rowGroupControlAfter	Boolean	True if the row grouping control toggle is shown after the group
columnGroupControlAfter	Boolean	True if the column grouping control toggle is shown after the group
updateSheetRequest	node	Sheet to be updated
SheetID	String	The ID of the sheet; must be non-negative; this field cannot be changed once set
title	String	The name of the sheet
index	Integer	The index of the sheet within the spreadsheet
hidden	Boolean	True if the sheet is hidden in the UI, false if it's visible
rightToLeft	Boolean	True if the sheet is an RTL (right-to-left language) sheet instead of an LTR (left-to-right language) sheet
gridProperties	GridProperties	Properties of a grid
rowCount	Integer	The number of rows in the grid
columnCount	Integer	The number of columns in the grid
frozenRowCount	Integer	The number of rows that are frozen in the grid

frozenColumnCount	Integer	The number of columns that are frozen in the grid
hideGridLines	Boolean	True if the grid isn't showing gridlines in the UI
rowGroupControlAfter	Boolean	True if the row grouping control toggle is shown after the group
columnGroupControlAfter	Boolean	True if the column grouping control toggle is shown after the group
deleteSheetRequest	DeleteSheetRequest	The value to search
SheetID	Integer	The ID of the sheet to delete
deleteCellRequest	node	Cells to be deleted
gridRange	GridRange	A range on a sheet
SheetID	Integer	The sheet this range is on
startRowIndex	Integer	The start row (inclusive) of the range, or not set if unbounded
endRowIndex	Integer	The end row (exclusive) of the range, or not set if unbounded
startColumnIndex	Integer	The start column (inclusive) of the range, or not set if unbounded
endColumnIndex	Integer	The end column (exclusive) of the range, or not set if unbounded
shiftDimension	node	How to shift the remaining cells when deleting cells
dimension	Dimension	String, either 'ROWS' or 'COLUMNS'; indicates which dimension an operation should apply to
updateCellRequest	UpdateCellsRequest	Cells to be updated
gridRange	GridRange	A range on a sheet
SheetID	Integer	The sheet this range is on
startRowIndex	Integer	The start row (inclusive) of the range, or not set if unbounded
endRowIndex	Integer	The end row (exclusive) of the range, or not set if unbounded
startColumnIndex	Integer	The start column (inclusive) of the range, or not set if unbounded
endColumnIndex	Integer	The end column (exclusive) of the range, or not set if unbounded
data	GridData	Node representing the data for the sheet for updating
row	RowData	Node representing a row of the sheet for updating
A...Z	n/a	String data for columns of a sheet; by default, 26 columns

• **Response:**

Response Schema Field /Node	Google API Field	Notes
updateSheetsResponse	n/a	Response from creating the spreadsheet

spreadsheetId	spreadsheetId	ID of the newly-updated spreadsheet
sheets	n/a	Node representing the spreadsheet
properties	n/a	Node representing the properties returned
code	n/a	Error code returned (if any)
message	n/a	Message returned upon creation (if any)

- **Add Plugin(s):** Plugins are Jitterbit- or user-provided applications that extend Harmony's native capabilities. To apply a plugin to the activity, click to expand this section and select the checkbox next to the plugin to be used. For additional instructions on using plugins, including details on setting any required variables used by the plugin, see [Plugins Added to an Activity](#).
- **Back:** Click to temporarily store the configuration for this step and return to the previous step.
- **Finished:** Click to save the configuration for all steps and close the activity configuration.
- **Discard Changes:** After making changes, click **Discard Changes** to close the configuration without saving changes made to any step. A message will ask you to confirm that you want to discard changes.

Next Steps

After configuring a Google Sheets update sheets activity, complete the configuration of the operation by adding and configuring other activities, transformations, or scripts as operation steps. You can also configure an operation's operation settings, which include the ability to chain operations together that are in the same or different workflows.

Google Sheets update sheets activities can be used as a target with these operation patterns:

- [Transformation Pattern](#)
- [Two-Transformation Pattern](#) (as the first or second target)

Other patterns are not valid using Google Sheets update sheets activities. See the validation patterns on the [Operation Validity](#) page.

To use the activity with scripting functions, write the data to a temporary location and then use that temporary location in the scripting function.

When ready, deploy and run the operation and validate behavior by checking the operation logs.