Odo User Guide

Author: Dave Abouav <daveabouav@google.com>
Last Updated: May 19, 2023

What is Odo?

Odo[1] is a configurable Workspace Add-on (GWAO) that allows any customer-facing individual (developer, account manager, sales rep, etc) to easily demo to their customer what the Google Workspace extensibility platform is capable of doing for, and with, the customer's in-house tools and processes. No programming is required! The goal is to help the customer quickly visualize how their own tool could be integrated into Google Workspace, and understand the benefit.

Where do I get Odo?

Odo is open-sourced and available at github.com/googleworkspace/gws-odo-addon. You can clone the repository and copy the code into a new Apps Script project. To run it, use either the "Test Deployment" method, or publish it as an internal Add-on within your own Google Workspace organization.

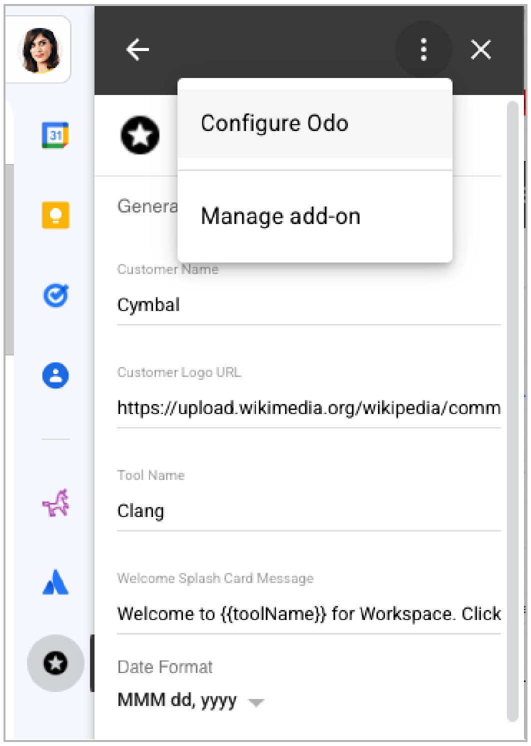
Where do I send feedback + bug reports?

Feedback and bug reports can be sent to daveabouav@google.com.

Configuring Odo

To configure Odo, launch the Add-on from the companion bar and select "Configure Odo" from the Add-on's universal menu. This will take you to the main configuration card.

[1] The name "Odo" comes from a shapeshifter character in Star Trek who can change his appearance at will to adapt to any situation.



General Configurations

In the “General” section you can customize the following fields:

General Configuration		
Configuration Name	**Description**	**Where and how it’s used**
Customer Name	Name of customer this demo is for	Shown in the header of most cards. Also available for use via {{customerName}} merge tag.
Customer Logo Url	Logo image for this customer	Shown as an icon image in the header of most cards. Also shown in larger size in the welcome splash card.
Tool Name	Name of the customer’s tool or integration being demoed	Shown in the header of most cards. Also available for use via {{toolName}} merge tag.
Welcome Splash Card Message	Text to display to the user upon first opening the configured Add-on. Shown each time a new config is saved.	Shown in larger size in the welcome splash card on first use of configured Add-on..
Date Format	Preferred format for displaying dates.	Controls how dates are displayed (i.e. when viewing a record for a Records

		Based integration).

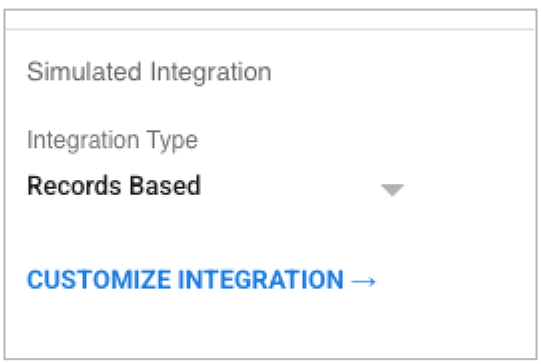
Note that all configuration settings are only stored locally in the context of the installed Add-on for the user who installed it (via Apps Script User Properties).

Configuring the Simulated Integration

In the “Simulated Integration” section you select the type of integration you’d like to demo. It’s called “Simulated” because these are not fully-baked integrations that actually perform any useful task. Their purpose is just to demonstrate how such an integration might look and operate as a GWAO.

The Integration Types section below describes each type of simulated integration, how it can be customized, and in what contexts it can be used.

Each integration type has its own settings that can be customized by clicking the “Customize Integration ⟶” button. If you choose not to customize the integration type, then a set of default settings for the integration type will be used.



Once you are finishing customizing an integration, be sure to click the “⟵ Done” button in the footer, followed by the “SAVE” button in the footer of the main configuration card. Otherwise your customizations will not be retained.

Saving all Configurations

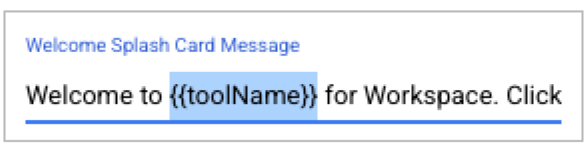
When you are done configuring the General and Simulated Integration sections (including the “Customize Integration” section, if used), be sure to click the “SAVE” button in the footer of the main configuration card.

Resetting all Configurations

To reset all Odo configuration settings back to their defaults (i.e. a “factory reset”), click the “RESET ALL” button in the footer of the main configuration card.

Merge Tags

Odo supports the use of merge tags to make the Add-on more customized and appear more like a real working integration. An example of a merge tag is visible on the main configuration page in the “Welcome Splash Card Message” field.



Here, the {{toolName}} merge tag is replaced with the corresponding value from your configuration before being displayed in your demo Add-on.

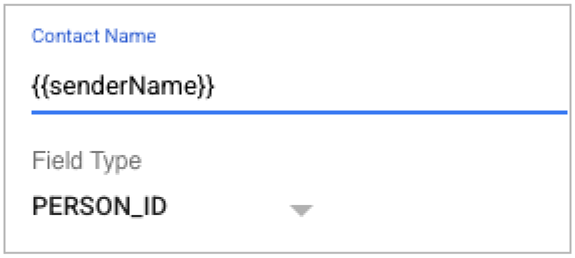
Certain merge tags are always defined, whereas others are only defined in specific contexts. The table below defines the available merge tags in Odo and the contexts in which they are defined.

Merge Tag	**Description**	**Contexts Defined In**	**Sourced From**	**Sample Use-Case / Demo**
{{toolName}}	Name of customer’s tool for	All	General Configuration	Quote name of tool in informative message.

	the demo			
{{customerName}}	Name of customer you're demoing to	All	General Configuration	Quote name of customer in informative message
{{selectedText}}	Currently selected text in Doc or Sheet.	Docs + Sheets	Text in doc or text in spreadsheet cell selected by user. Only uses the first word of selected text. Ignores blank text (just spaces) and empty cells.	Surface a "relevant" record in a Records Based integration based on the selected cell in a spreadsheet (i.e. for "Asset ID" field, enter "{{selectedText}}" merge tag as the value to be used).
{{senderName}}	Name of sender for email currently being viewed.	Opened Gmail message	Gmail header	Surface a "relevant" record in a Records Based integration based on the person ("context") who sent you an email (i.e. for "Contact Name" field, enter "{{senderName}}" merge tag as the value to be used.
{{senderEmail}}	Email address of sender for email currently being viewed.	Opened Gmail message	Gmail header	Surface a "relevant" record in a Records Based integration based on the person ("customer") who sent you an email (i.e. for "Customer Email" field, enter "{{senderEmail}}" merge tag as the value to be used.
{{token}}	Generic identifier	Opened Gmail message Docs + Sheets	If Gmail, same as {{senderEmail}} If Docs or Sheets, same as {{selectedText}}	Can be passed as a URL parameter in a Generic Service integration type to identify a user or entity that the service should act on (i.e. start a whiteboarding session with this user, or call up a record for this entity).
{{fileName}}	Name of file currently selected in Drive	Drive	File currently selected in Drive. If multiple are selected, Odo only uses the first one.	Surface name of file selected in Drive integration (i.e. file to be checked-in to file repo)

Default Merge Values

Merge tags are not guaranteed to be defined in all contexts. For example, say you have a Records Based integration that surfaces Customer Records, and for the “Contact Name” field you have entered “{{senderName}}”.



When you open the Add-on in the context of an open Gmail message, that merge tag is defined. Ergo the Contact Name field will show the name of the sender of that email. But if you load the Add-on in the context of a Spreadsheet, the merge tag is not defined in that context and so no merge tag/value replacement will occur. Ergo, the “Contact Name” field would still show “{{senderName}}” in the record.

To avoid this poor experience, you can define a “default merge value” using the “||” operator inside the merge braces. For example if you instead enter “{{senderName || Wile E. Coyote}}”, then Odo will replace the merge tag with the appropriate value if/when {{senderName}} is defined (i.e. when viewing an email), but will otherwise use the default value (“Wile E. Coyote”) if/when the merge tag is not defined.

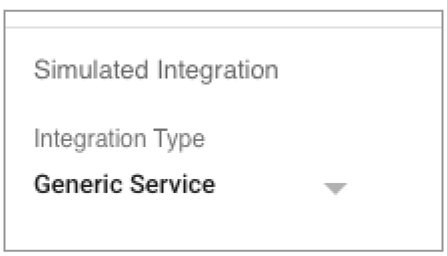
Simulated Integration Types

Odo comes pre-configured with several simulated integration types[2]. These are meant to cover the majority of common types of integrations customers need, with enough configuration options to cover most demos. They are called “Simulated” because these are not fully-baked integrations that actually perform any useful task. Their purpose is just to demonstrate how such an integration might look and operate as a GWAO.

The sections below describe each integration type, how it can be configured, and how it behaves in different contexts.

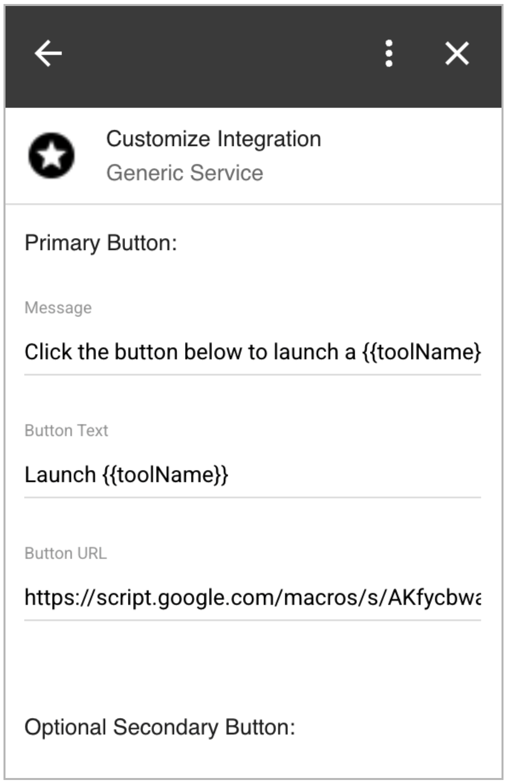
Generic Service

This simulated integration is meant to showcase a very simple integration of a web-based service into Workspace. For example, perhaps a customer has an in-house app (i.e. a whiteboarding tool, a video conferencing app, an employee directory) that they want their employees to be able to easily access from various apps within Workspace.

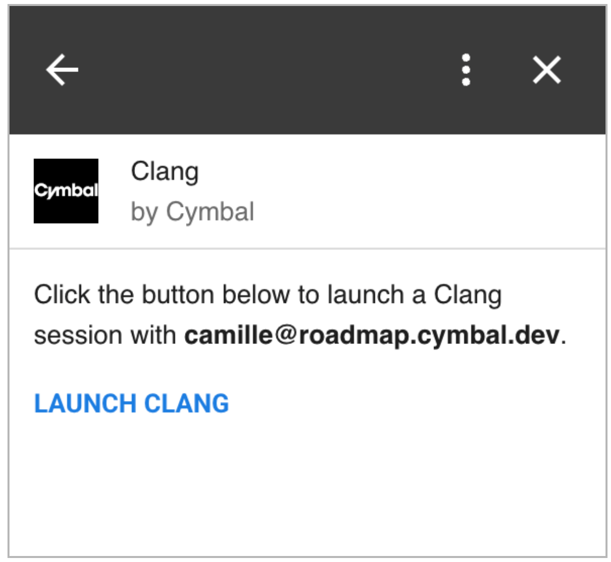


The integration is just a simple card with 1 or 2 buttons. For each button you can configure a message, the button text, and a target URL. The same, simple card is shown in all contexts (i.e. when opened from Gmail, a Doc, a Sheet, etc). Note: See [this tip](#) if demoing in a brand-new editors file.

[2] If you would like to write your own integration type, contact daveabouav@google.com for more information. Odo was written so as to make this fairly straightforward, with the code for each simulated integration type being isolated in its own file. A central framework makes calls to predefined functions in your integration that you must implement, with well defined inputs/outputs for each.



Generic Service Customization



Generic Service Card (only one button)

A secret power of this integration type is the ability to use the {{token}} merge tag. In the context of an open Gmail message, the {{token}} merge tag will expand to the email address of the sender. For example, say you have customized the integration such that buttonUrl = “https://www.whiteboarding.com/new**?invitee={{token}}**”, and you have opened an email from “Bob <bob@company.com>”. When rendered, the buttonUrl will become:

> https://www.whiteboarding.com/new?invitee=**bob%40company.com**[3]

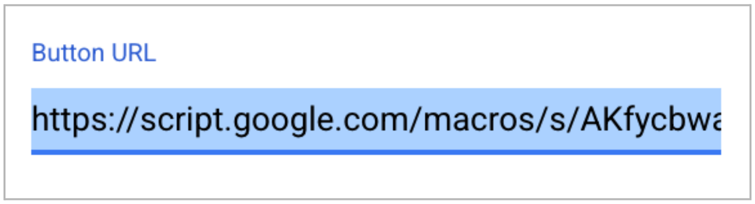
In this way you may be able to actually demonstrate how the integration could be used to initiate some type of session with the email’s sender.

In the context of a Doc or Sheet, {{token}} will expand to the currently selected text or cell. Ergo, if used from a Doc or Sheet, you could point the buttonUrl to some type of service (real or simulated) that looks up the entity selected by the user.

---

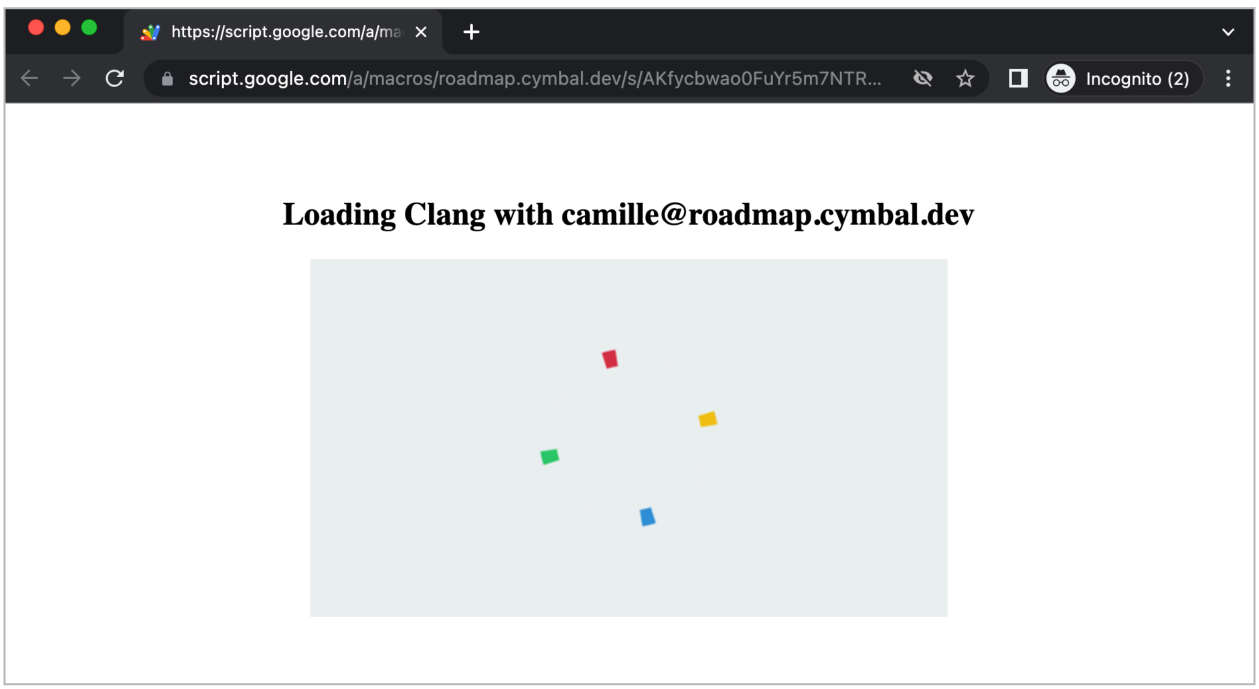
[3] Note that in this integration the {{token}} parameter will get URL encoded for you when used for either buttonUrl.

If you don’t have such a web-based service you can configure for your demo, Odo comes with one pre-configured:



The full URL of which is:
https://script.google.com/macros/s/AKfycbwao0FuYr5m7NTRMIBZcVdEiPZSN7L4cFrILcvQgr5-GRPOe9JKkNcCh5bP9TZaZAAI/exec?**loadingMessage=Loading%20{{toolName}}%20with%20{{token || participants}}**

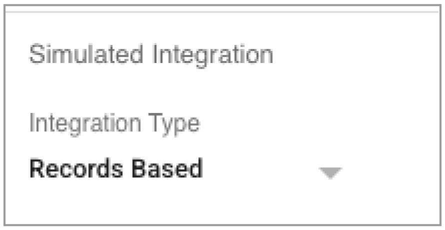
When clicked, this takes the user to a faked service (a public Apps Script web app) that looks like this:



Note that the message shown at the top of the web app can be customized, as it’s embedded in the default buttonUrl for this service.

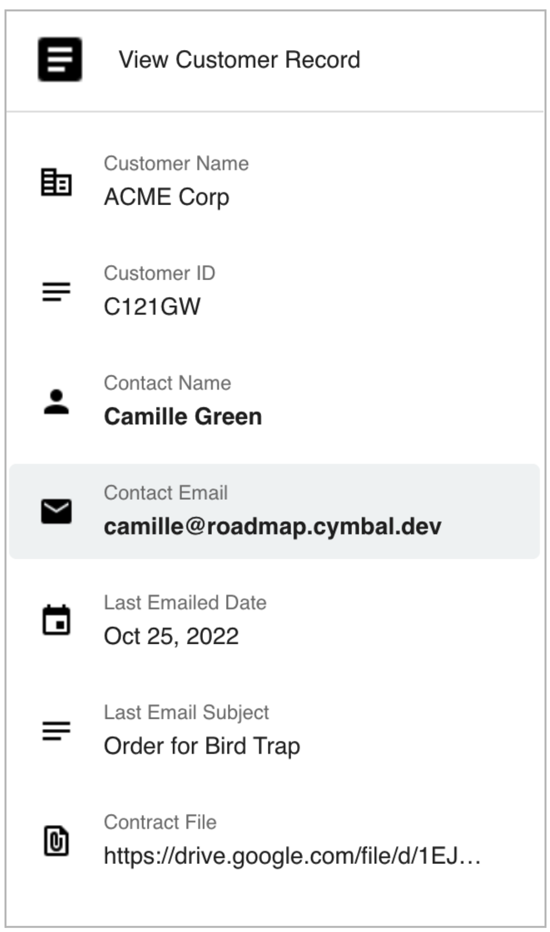
Records Based

This simulated integration showcases how a GWAO can be used to surface various types of records.



The records can be shown in several contexts:

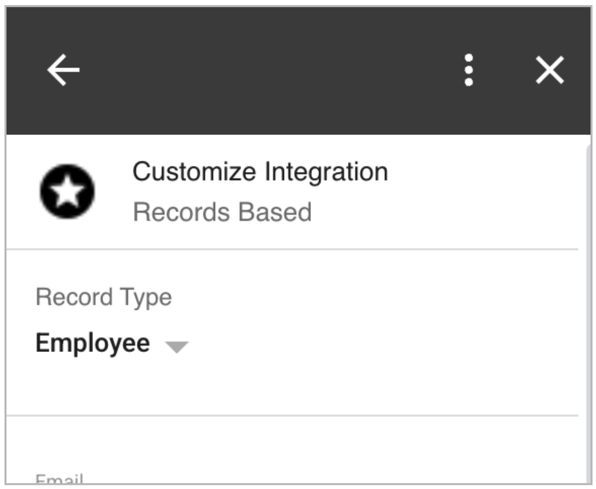
- Gmail message opened: Show record based on sender of email
- Gmail compose message: Show record's file as an attachment to insert
- Docs: Show record based on selected text
- Sheets: Show record based on text in selected cell



Sample of a Customer record shown in Gmail for a Records Based Integration. The name and email address of the sender are shown in the "Contact Name" and "Contact Email" record fields, respectively.

Record Types

The Records Based integration can be customized to use different, pre-configured record types, as described in the table below.



Use the drop-down menu to choose a **Record Type** for Records Based integrations.

Record Types for Records Based Integration		
Record Type	Description	Sample Use-case / Demo
Customer	Surfaces a record of a customer/client	A light-weight CRM integration. In your demo, open an email from a client and then open the Add-on to show the record for that client. The client's name and email address (from the email) will show in the record due to the merge tags used. Reply to the client's email and use the Add-on's compose integration to call up the same record. Click on the button in the footer to insert a copy of their Contract File into the email.
Asset	Surfaces a record of a company asset	A light-weight inventory management integration. In your demo, open a spreadsheet. From the Add-on, click "Start Report", enter

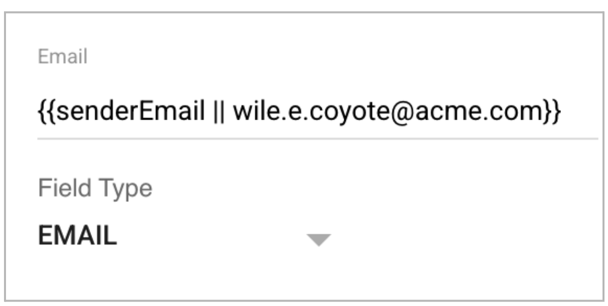
		some dates, and then click “Generate Report”. Click any cell in the Asset ID column of the generated report, and click the “Lookup Record” button to call up the associated record in the Add-on. Note that the Asset ID selected shows in the “Asset ID” field due to the {{selectedText}} merge tag used.
Employee	Surfaces a record of an employee at the customer’s company	A light-weight user directory integration. In your demo, open an email from a fellow employee and then open the Add-on to show the record for that employee. The employee’s name and email address (from the email) will show in the record due to the merge tags used.

Selecting and Configuring Record Type

Each record type comes with several predefined Record Fields, each with a default value. All of these values, and even the name and type of each field, can be customized to your needs.

Each Record Field has 3 parts:

- **Name**: Name to be displayed when showing records or in the header when generating a report
- **Value**: The value to be shown for this field. This value can use merge tags.
- **Type**: The type of field. This field just controls the formatting of the field when shown.



In this example, the Name of the Record Field is “Email”, the Value is “{{senderEmail || wile.e.coyote@acme.com}}”, and the Field Type is “Email”.

More about Record Types

The table below describes how Record Type affects the formatting of the record field.

Record Type	Description	Formatting Example
TEXT	Generic record field.	Customer ID C121GW
EMAIL	Identifies an email address	Contact Email **camille@roadmap.cym**
PERSON_ID	Identifies an individual	Contact Name **Camille Green**
ORG_ID	Identifies an organization	Customer Name ACME Corp
DATE_EPOCH_MS	Date expressed as number of MS since Jan 1, 1970. See here for help.	Last Emailed Date Oct 25, 2022
DATE_OFFSET_DAYS	Date expressed as the whole number of days from today's date. For example, a value of -7 will always show the date from 1 week ago.	Last Emailed Date Oct 25, 2022
FILE_URL	Link to a file	Contract File https://drive.google.cor
FOLDER_URL	Link to a folder	Contract Folder https://drive.google.cor

If in doubt of which field to use, just select a Field Type of "TEXT".

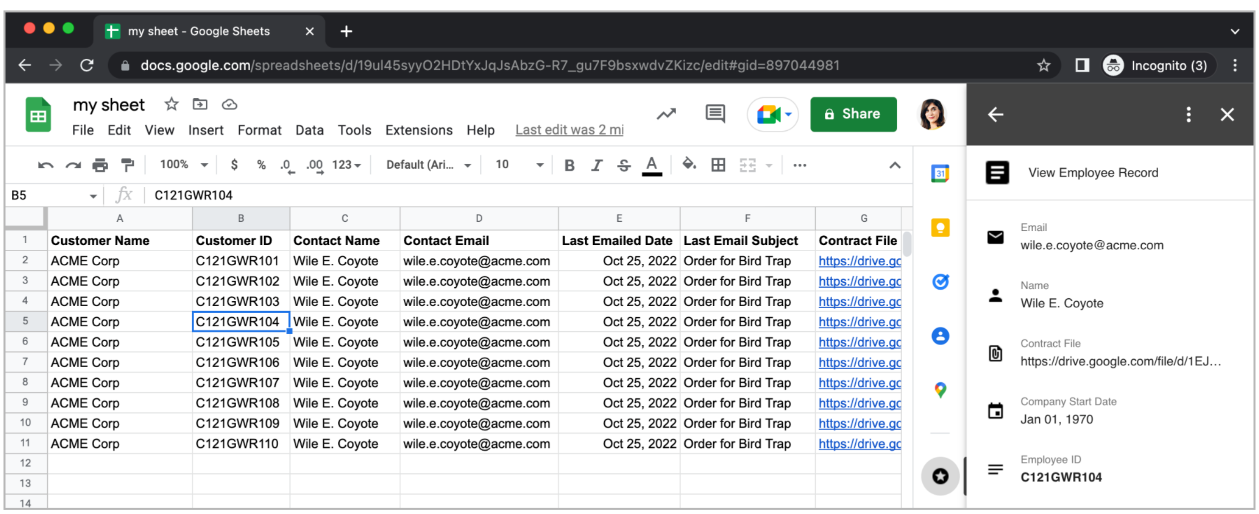
In the future a Generic or "DIY" (Do-It-Yourself) record type may be added, allowing you to start from scratch for your Record Type.

Report Generation

Note: See [this tip](#) if demoing in a brand-new editors file

When generating a report in a Spreadsheet, please note that:

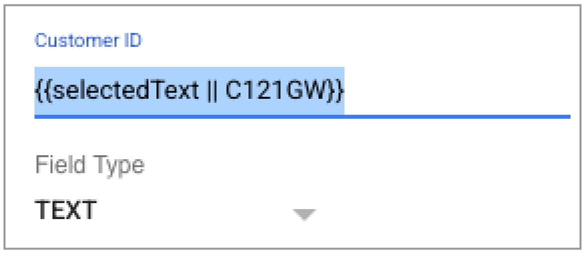
- The “Start Report” button only shows if the currently selected cell is A1. This is a hack to ensure this option shows when the spreadsheet is first loaded. For the same reason, if you click “Lookup Record” when cell A1 is selected, nothing will happen.
- The report dates specified won’t actually affect the report generated.
- A report of 10 rows (plus a header row) will be written to a new sheet, with one column per record field, corresponding to the Record Type selected.
- Each row will be almost identical, simply copied from the values configured for the Record Type. There is one exception - see next bullet.
- In the report, the column corresponding to any record field that uses “{{selectedText || <default merge value>}}” will have a number (based on the row number) appended to its value when written out. This is done so that values that may later be selected to lookup a record will be different from one another.



Customer Name	Customer ID	Contact Name	Contact Email	Last Emailed Date	Last Email Subject	Contract File
ACME Corp	C121GWR101	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR102	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR103	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR104	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR105	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR106	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR107	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR108	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR109	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc
ACME Corp	C121GWR110	Wile E. Coyote	wile.e.coyote@acme.com	Oct 25, 2022	Order for Bird Trap	https://drive.gc

Sample of generated report (for **Customer** Record Type). Notice how the “Customer ID” column changes its value based on the row number, ensuring that not every row is identical.

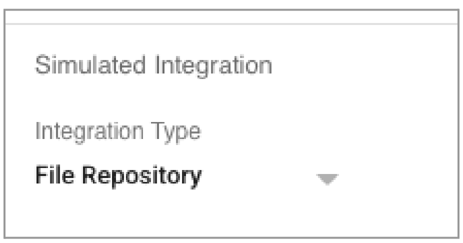
This is because the Customer ID field was configured as {{selectedText || C121GW}}, as shown below:



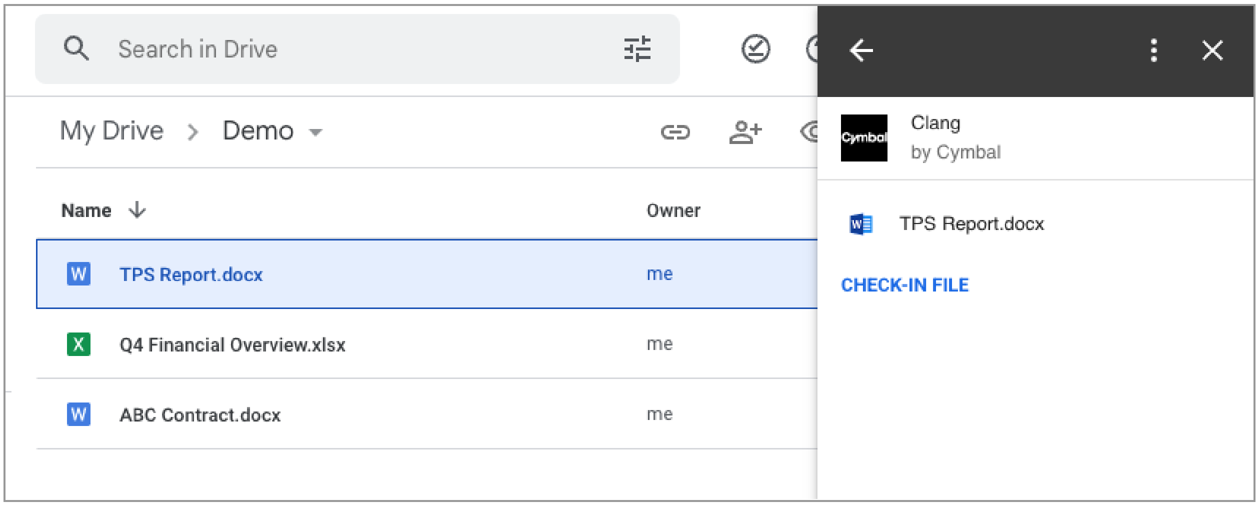
The idea is that any field you'd normally display based on what text has been selected is likely an identifier of some sort, and so it would still make sense with additional numbers appended after it.

File Repository

This simulated integration showcases a Drive integration. It does this by simulating an integration with a 3rd-party file repository that allows the user to check-out files from the repository to Drive, and subsequently check-in files from Drive to the repository. The integration only works with MS Office files.

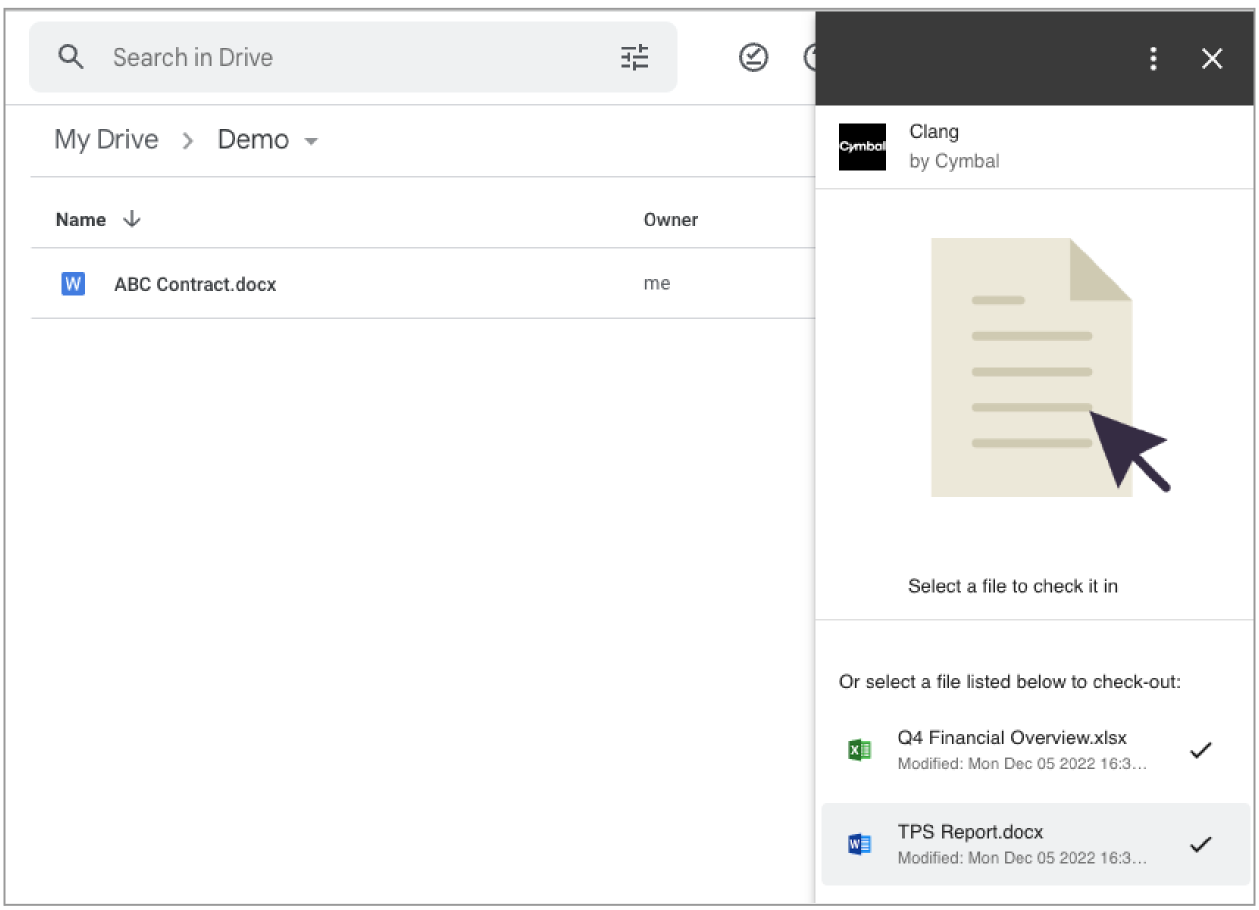


To demo this integration, it is recommended you create a folder in your Google Drive and place one or more MS Office files there. Edit one of the files to make some changes. Then load the File Repository integration and check-in the files by selecting it and then clicking the "Check-In File " button.



Note that the Add-on doesn't actually check-in your file. It just simulates this by hiding it in a new folder called "ODO_DATA" in your My Drive.

When it refreshes, the Add-on will show all files that have been checked-in:



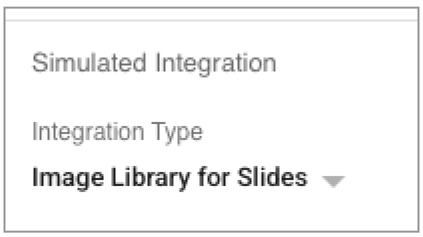
You can check-out a file by clicking the check mark symbol next to it. The integration will "check-out" the file by moving it back from the ODO_DATA folder to the same folder it was "checked-in" from.

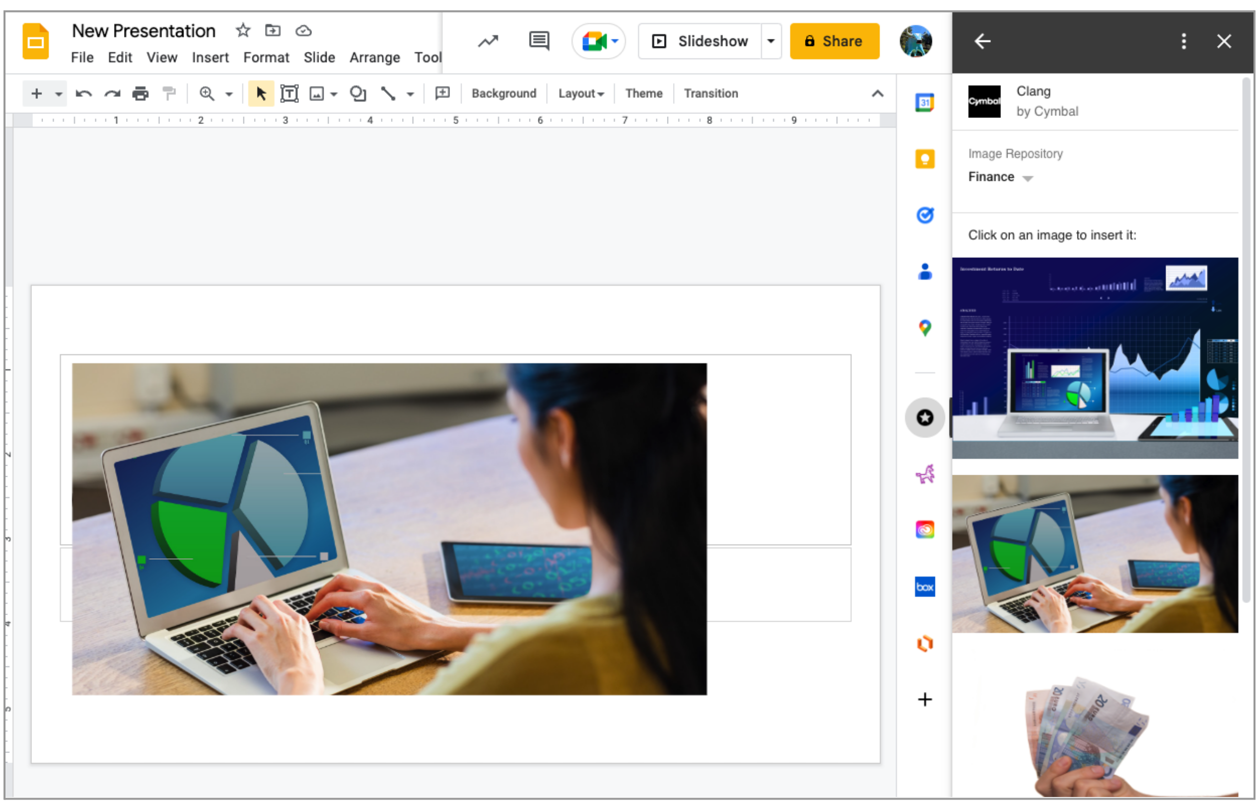
Open the file to show that the changes you made before checking in the file are still there.

Image Library for Slides

Note: See [this tip](#) if demoing in a brand-new file

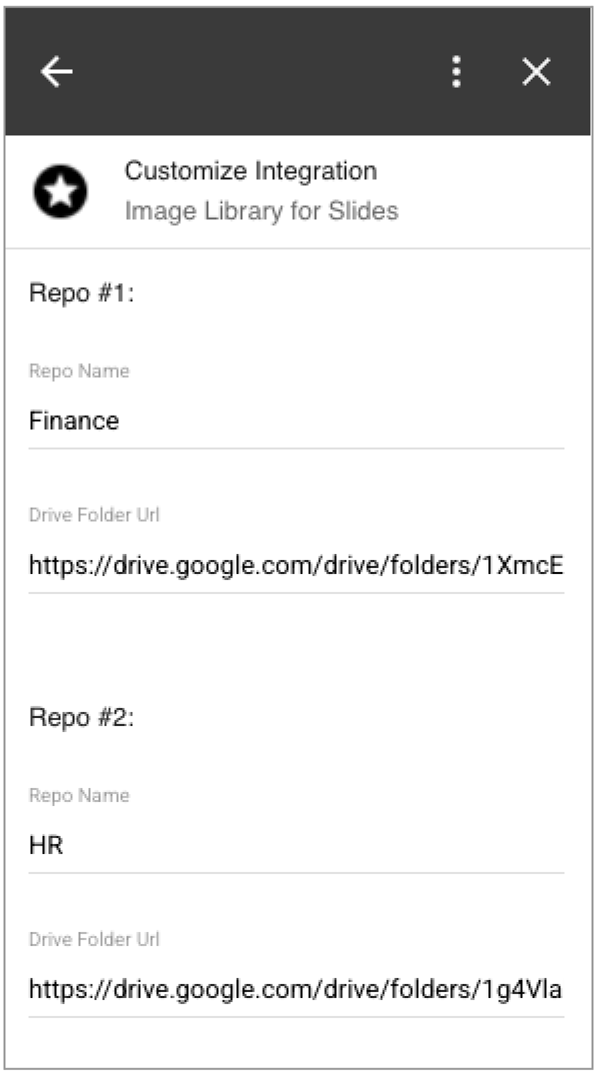
This simulated integration is meant to showcase a Slides integration. It does this by simulating a library of stock images users at a company may use, with several image repositories in the library.





Two image repositories are provided by default (“Finance” and “HR”), with several stock photos in each.

The image repositories are faked using Drive folders with images in them. You can make your own image repositories (i.e. ones with images related to the customer you’re demoing to) by customizing the integration and adding up to 3 repository names and corresponding Drive folder URLs. Make sure the Drive folders are Viewable to “Anyone with the link”.



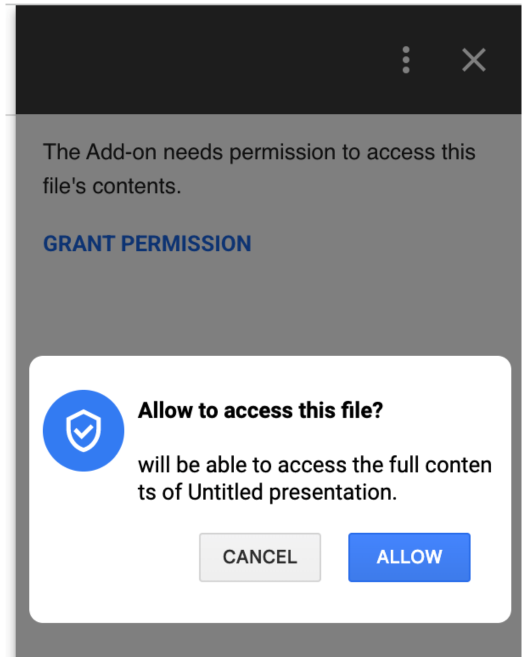
Calendar / Video Conferencing

Coming soon!

Demo Tips & Troubleshooting

Granting Permission on New Files

Opening an integration in an editor file (i.e. "Records Based" in a spreadsheet, or "Image Library" in Slides), will result in a file access dialog appearing (see image on right). You will have to click "Allow" before you can proceed. This is standard for Google Workspace Add-ons.



But note that this process will only work on an existing file that's *already* been saved in Drive. If you try this on a brand-new file with no changes (i.e. new "Untitled Spreadsheet"), it will **continue to fail** to grant access. The reason is that the new editor file doesn't get saved to Drive until some changes are actually made to the file. This also prevents the access grant from being saved.

To avoid this, always make changes to the new file before you try to open Odo, or better, just demo on a pre-existing file in Drive.