

Conditional Integrations Extension

Last Modified on 06/24/2026 11:58 am EDT

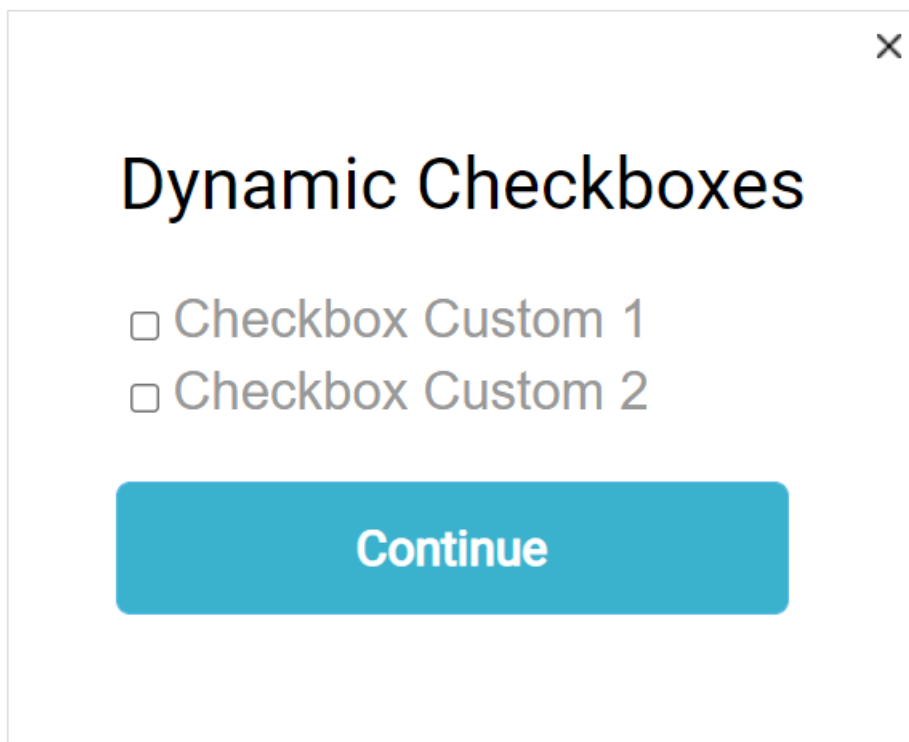
Do you need to activate or suppress specific integrations based on form input data? We have an app for that!

The Conditional Integrations app allows you to activate specific integration IDs based on the "truthiness" of any form field, on any page, even for submissions where the conditional value is on a *prior* page.

Go to Custom JS (Apps) from the profile menu top right, and install "Conditional Integrations".

Then, you can activate specific integrations for form submissions on a given Campaign page using metadata.

Let's say you have this for page 1 of your Campaign, with checkboxes as form Custom fields 1 and 2.



Dynamic Checkboxes

Checkbox Custom 1

Checkbox Custom 2

Continue

Then on page 2 you have a separate (progressive) form:

×

Dynamic Checkboxes 2

Enter your email address

Submit

And you have two Integrations that you want to call based on which checkboxes are checked (none, one, or both):

Integrations

ID	Name	Type
59817	Conditional Integration 2	Real Time Email
59816	Conditional Integration 1	Real Time Email

With the Conditional Integrations app installed, you can configure metadata on the 2nd page (i.e. at the page-level, under Layout) as follows:

Main Page

▼

Apply

Apply Page Layout To

Choose Pages

Apply

Page Metadata

Key	Value	
ci:custom_1	59817	+ -
ci:custom_2	59816	+ -

Page Name

Dynamic Checkboxes 2

Enter your email address

Submit

The format is `ci:fieldName = integrationId`

... where `fieldName` is the specific field name (email, phone, opt_in, custom_1, custom_2, ...), and `integrationId` is the ID of the integrations.

You can use metadata to control the integrations based on form data submissions for the *same* page, or a *subsequent* pages. Typically, you should avoid attaching a (static) integration to the campaign itself. When the `ci:custom_N` page is submitted, any statically attached integrations will be **replaced** by any metadata dynamic integrations. If there are no dynamic integrations activated, static integrations will run as expected.

Note that you can use this app for Custom Fields that are not checkboxes. Integrations are activated if the value is "truthy". Custom fields are "truthy" if they are not blank, and not equal to "false".

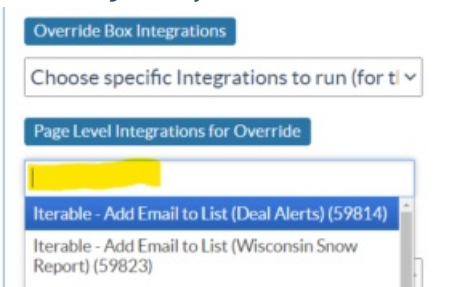
WARNING: this app is not compatible with the "native" UI features for checkbox integrations or form level integration overrides.

Under Page > Form > Fields > (Field), when a checkbox:



The screenshot shows a configuration interface for a checkbox. Under the heading "Form Values", there are two input fields. The first is labeled "true" and has "checked" written below it. The second is labeled "false" and has "unchecked" written below it. Below these fields is an "Integration ID" field, which is currently redacted with a yellow bar.

Under Page > Layout:



The screenshot shows a configuration interface for overriding integrations. Under the heading "Override Box Integrations", there is a dropdown menu labeled "Choose specific Integrations to run (for t...". Below this is a section titled "Page Level Integrations for Override" which contains a list of integrations. The first item is "Iterable - Add Email to List (Deal Alerts) (59814)" and the second is "Iterable - Add Email to List (Wisconsin Snow Report) (59823)". A yellow redaction bar is visible above the list.

If you need to use these features, the Conditional Integration also supports the same functionality. For example, with a checkbox of `custom_1`, you can say in page metadata `ci:custom_1=12345` to get the same effect (where 12345 is integration ID). For form level overrides, you can say in page metadata `ci:any=12345`. The metadata key `ci:any` is a special conditional command that is always true.
