

Introduction to Responsive Forms

Responsive Design is a general design principle that describes a user interface that adapts its layout and dimensions to various devices or screen sizes. Responsive Forms on the Aras Innovator platform leverage this design principle to provide Forms that dynamically adjust their structure and content to deliver a seamless user experience.

Responsive Forms vs. Aras Classic Forms

Prior to Aras Innovator 30, there was a single way to define a Form for the Aras Innovator client, now called Classic Forms. This section outlines key differences between Responsive Forms and Classic Forms.

Administrator User Interface

When defining a Classic Form, an administrator creates a new Form item and uses the Classic Form Editor UI to place fields into the Form body. The admin then drags each field to its desired position in the Form. This approach assigns each field an X,Y coordinate that the Aras Innovator platform uses to render a static layout for the Form.

Responsive Forms use an enhanced admin user interface called the Responsive Form Designer. This UI enables admins to define a container-based layout by dragging and dropping structural elements and fields into the Form. Administrators can also configure the layout behavior of the Form and preview how it will look and behave for end users.

Layout Behavior

Classic Forms render content using a static layout that doesn't change based on the size of the user's browser window. Instead, users will be presented with scroll bars when a Classic Form's static content does not fit in frame, or they will see additional whitespace when the static content occupies less space than the current frame.

Responsive Forms use a container-based approach to define the layout of a Form. This means that content can reflow to use the browser window's space more effectively – minimizing the need for scroll bars and reducing unnecessary whitespace.



Data Model

Both Classic and Responsive Forms use the Form ItemType to define Forms for the Aras Innovator client. However, each approach uses a different data structure under the Form item to describe the layout, behavior, and content of the rendered Form.

This parallel architecture enables administrators to adopt Responsive Forms with minimal disruption to packaging and deployment. It also enables customers to continue using their existing Classic Forms.

Important

A single Form item might have both Classic and Responsive definitions present at the same time. However, the Aras Innovator platform will only render the definition identified by the Form's classification. Administrators should use the Classic Form Editor or Responsive Form Designer to choose which definition will be rendered for each Form.

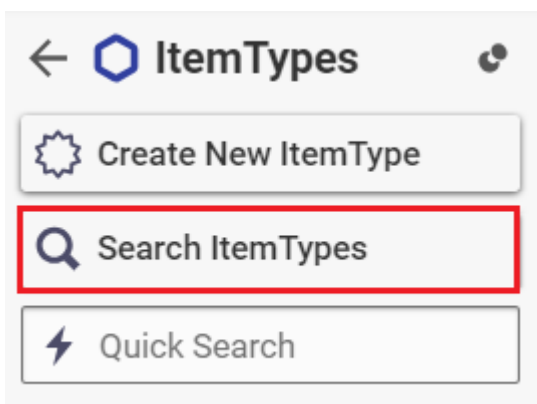
Important

Modifying the layout, behavior, or content of a Classic definition does not affect the Responsive definition, or vice versa.

Create a Responsive Form for an ItemType

The following process outlines how to add a new Responsive Form to an ItemType:

1. From the **Table of Contents**, expand **Administration** and select **ItemTypes**.
2. From **ItemTypes Table of Contents**, click **Search ItemTypes**.



The ItemType grid appears.



3. In the **Name** column, enter required ItemType click **Search**.

4. Select the ItemType.

The ItemType form appears. For this document, Part ItemType is used.



ItemTypes x
Part x

Part

Edit
Refresh
Share
Add
Share
More

^ ItemType

Name
Part

Singular Label
Part

Show Parameters Tab
Never

Class Structure

History Template
Default

Plural Label
Parts

Default Structure View
Tabs On

Small Icon

Versioning
Versionable ☒
Discipline
Automatic
Revisions
Default

Search
Auto Search ☐
Default Page Size
25
Max Records

☐ Unlock On Logout
☐ Dependent
☐ Is Relationship
☒ Enforce Discovery
☐ Use Src Access
☒ Allow Private Permissions

Implementation Type
☒ Single Item
☐ Poly Item
☐ Federated Item

Enable for Secure Social

Large Icon

^ Properties RelationshipTypes Views Server Events Actions Life Cycles Workflows Client Events Can Add Permissions Reports Po

Properties v ☆

Hidden

5. From the Relationship tab, select **Views**.



ItemTypes x
Part x

Part
☆

Edit
Refresh
Link
Share
Add
Share
More

^
ItemType

Name
Part

Singular Label
Part

Show Parameters Tab
Never

Class Structure

History Template
[Default](#)

Plural Label
Parts

Default Structure View
Tabs On

Small Icon

Versioning

Versionable ☒

Discipline
Automatic

Revisions
[Default](#)

Search

Auto Search ☐

Default Page Size
25

Max Records

Large Icon

^
Properties
RelationshipTypes
Views
Server Events
Actions
Life Cycles
Workflows
Cl

Properties
☆

Add
Remove
Search
Hidden
View
Monitor
Share

The existing forms linked to Part ItemType appears.

Properties RelationshipTypes **Views** Server Events Actions Life Cycles Workflows Client Events Can Add Pe

Forms ▾ ☆






Hidden ▾
 ▾
 ▾
 ▾

Form Name ↑ 2	Type ↑ 1	Description	Identity [...]
Part	Default		World
Part AML Editor	Default		World

8. Add a name to the form in the **Form Name** column and select **Default** type in the **Type** column.

Properties RelationshipTypes **Views**

Forms ▾ ☆






Hidden

Form Name ↑ 2	Type ↑ 1
Part	Default
Part AML Editor	Default
 New Responsive Form	Default ▾

Important

Responsive form supports the following form view types: **Default** (since R28), **Preview** (since R30) and **Add, View and Edit** (since R32).

9. Enter the other ItemType View properties as needed and click **Done** to save the changes.

Part ☆

Save Done Discard

Part Parts

Show Parameters Tab Never

Default Structure View

Revisions Default

Max Records

Enforce Discovery

Use Src Access

Allow Private Permissions

Federated Item

Enable for Secure Social

Class Structure

Small Icon

Large Icon

Properties RelationshipTypes Views Server Events Actions Life Cycles Workflows Client Events Can Add Permissions Reports Poly Sources Client Sty

Forms ☆

Form Name ↑ 2	Type ↑ 1	Description	Identity [...]	Classification	Display Priority	Secure Social
Part	Default		World		1	☒
Part AML Editor	Default		World		999	☐
New Responsive Form	Default		World	Software	1	☐

10. Double click on the newly created Form in the Relationship tab.

Part ☆

Edit

Part Parts

Show Parameters Tab Never

Default Structure View

Revisions Default

Max Records

Enforce Discovery

Use Src Access

Allow Private Permissions

Federated Item

Enable for Secure Social

Class Structure

Small Icon

Large Icon

Properties RelationshipTypes Views Server Events Actions Life Cycles Workflows Client Events Can Add Permissions Reports Poly Sources Client Sty

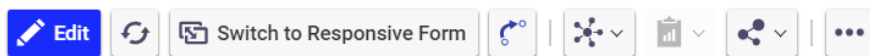
Forms ☆

Form Name ↑ 2	Type ↑ 1	Description	Identity [...]	Classification	Display Priority	Secure Social
New Responsive Form	Default		World	Software	1	☐
Part	Default		World		1	☒
Part AML Editor	Default		World		999	☐

The Classic Form Editor displays the new Form:



New Responsive Form ☆



	Field Type	Field Label	Field Physical	Field Border	Field CSS	Field Event	Form Properties	Form Body	Form Event
ab	button_ok								
ab	cancel								
	grid								
abl	item_number								
abl	state								
	statusbar								

Name button_ok

Field Type Button

Data Source

Background Color

Copy (2) Of Copy Of Part NumberCopy (2) Of Copy Of State

Copy (2) of Copy of Copy (2) of Copy of Cancel

Important

New Forms are created as Classic Forms by default. Classic Forms created from an ItemType are automatically populated with fields generated based on the search grid properties for the parent ItemType.

11. Click **Switch to Responsive Form**.



New Responsive Form ☆



	'New Responsive Fo...	Field Type	Field Label	Field Physical	Field Border	Field CSS
ab	button_ok	Name	button_ok			
ab	cancel	Field Type	Button			
	grid	Data Source				
abl	item_number	Background Color				
abl	state					
	statusbar					

The Form switches to Responsive mode and it's reloaded in the Responsive Form Designer interface:

New Responsive Form ☆

Edit Refresh Switch to Responsive Form Undo Redo Settings

Structure

- 2 Column Set
- 3 Column Set
- 4 Column Set
- Column
- Group Box

Unused Properties

- Changes
- config_id
- Control Type
- Cost
- Cost Basis
- created_by_id

Drag and Drop Elements Here

Select from the "Elements" menu at the left

← →

New Responsiv...

General

Name

New Responsive Form

Item Type

Part

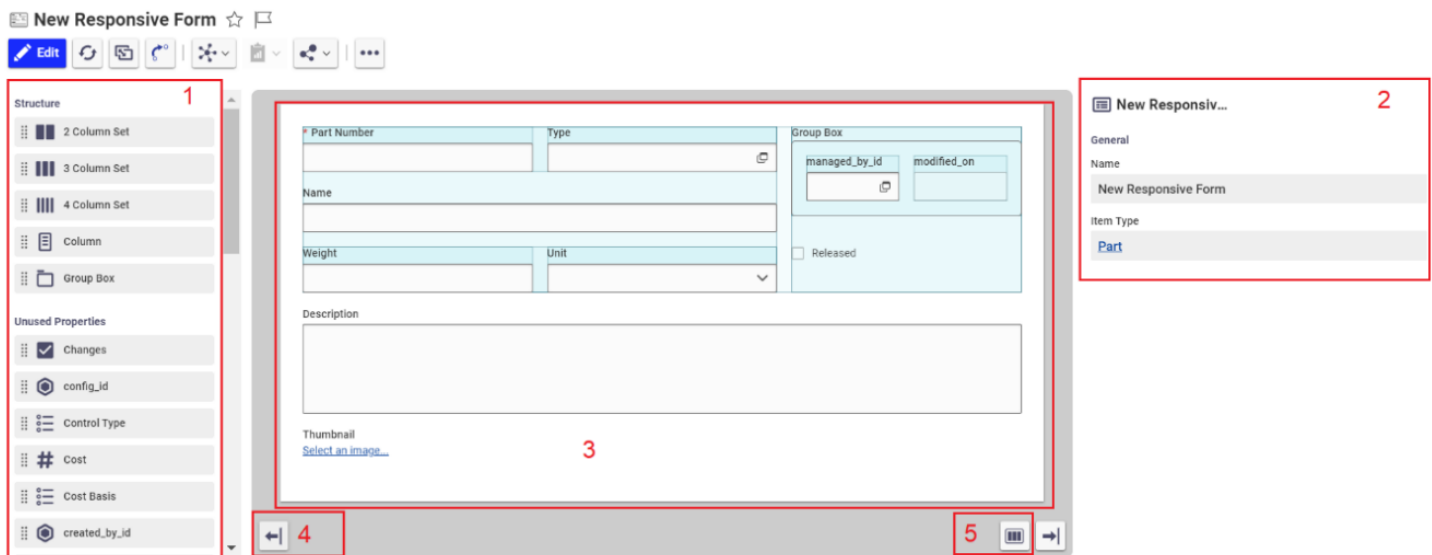
Important



When a Form is switched to Responsive mode for the first time, it receives a new empty Responsive Form definition. Switching a Form to Responsive mode does not migrate, modify, or delete the Form's Classic content. Switching a Form back to Classic mode also does not affect the Form's Responsive content.

Responsive Form Designer

This section explains all the panes, properties and functionalities of the Responsive Form Designer.



There are 3 panes for the Responsive Form Designer:

1. **Element Pane:** This pane presents a list of elements and fields that the admin user can drag and drop into the form. There are two types of objects:
 - **Structure Objects:** These are building blocks for constructing the layout of Responsive Forms. It consists of Column, Column Sets and Group Box.
 - **Column:** A container that wraps a vertical stack of Column Sets, Group Boxes, or fields. All fields in a column will fill 100% of the width of the container. Columns may only be placed inside Column Sets.
 - **Column Set:** A container that wraps one or more Columns. A Column Set can be used to arrange Columns of content side by side, or they can be nested within Columns to give the appearance of a split Column. By default, the Columns inside a Column Set will reflow to stack vertically when the width of the Form shrinks. However, this behavior can be overridden by the administrator as needed. Column Sets may be placed inside a Column or at the root level of the Form.

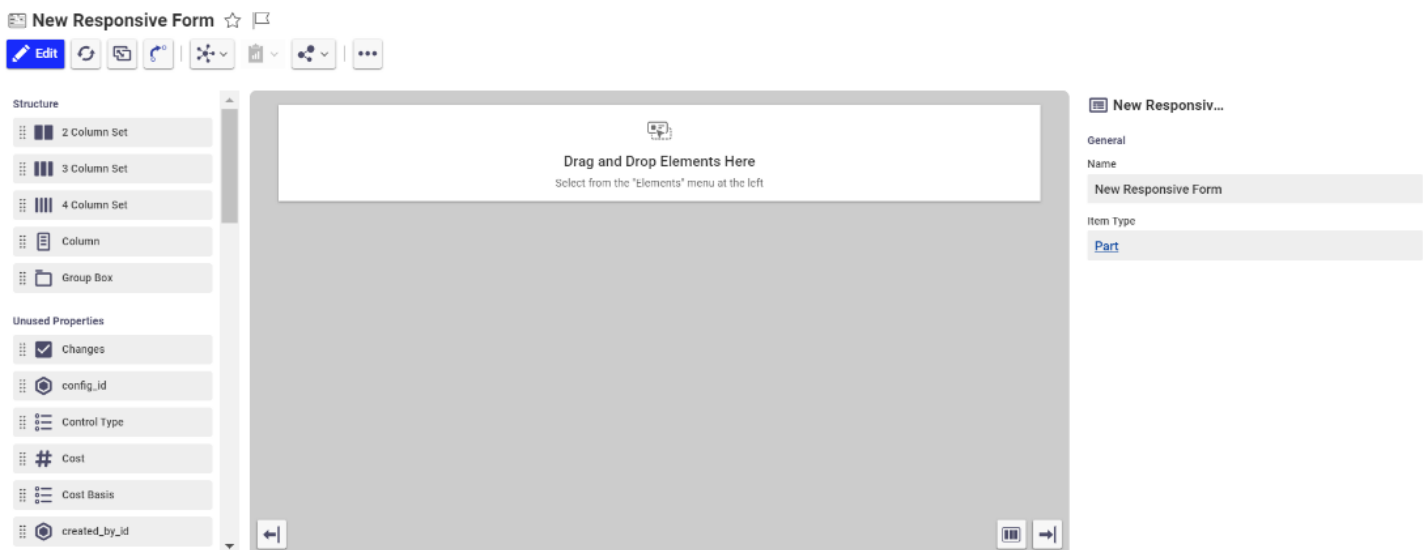


- **Group Box:** A Column Set preconfigured to show a label and border. A Group Box can wrap related content, helping users identify related data even on varying screen sizes. Group Boxes can be used just like Column Sets – inside a Column or at the root level of the Form.
- **Unused Properties:** These are individual fields that are preconfigured to the properties of the Form's parent ItemType. Each field has a default control type such as text box, checkbox, dropdown, etc. As the admin user drags & drops a field into the Form, the corresponding property is removed from the Unused Properties list.

Important

The **Unused Properties** list will only populate when the Form is related to an ItemType. To add a Form to an ItemType, open the desired ItemType and create a new View relationship to the Form.

2. **Detail Pane:** This pane enables the admin user to view and edit the Form's properties. When a structure element or field is selected in the Canvas Pane, the admin can also view or edit the properties of the selected element or field.
3. **Canvas Pane:** This is where the admin user will drag & drop elements to define the content and layout of the Responsive Form. The admin can preview how the content will reflow by changing the size of the browser window or expanding/collapsing the other panes in the designer.



4. **Toggle Pane:** Two buttons on either side of the Canvas enable the user to toggle the designer's side panes on or off. This helps the admin preview the Form's wrapping settings by



changing the size of the Canvas.

5. **Toggle Guides:** Enables the admin to toggle off the blue boxes that show the Form's structure elements in the Canvas if they want to see how the rendered Form will look for an end user.

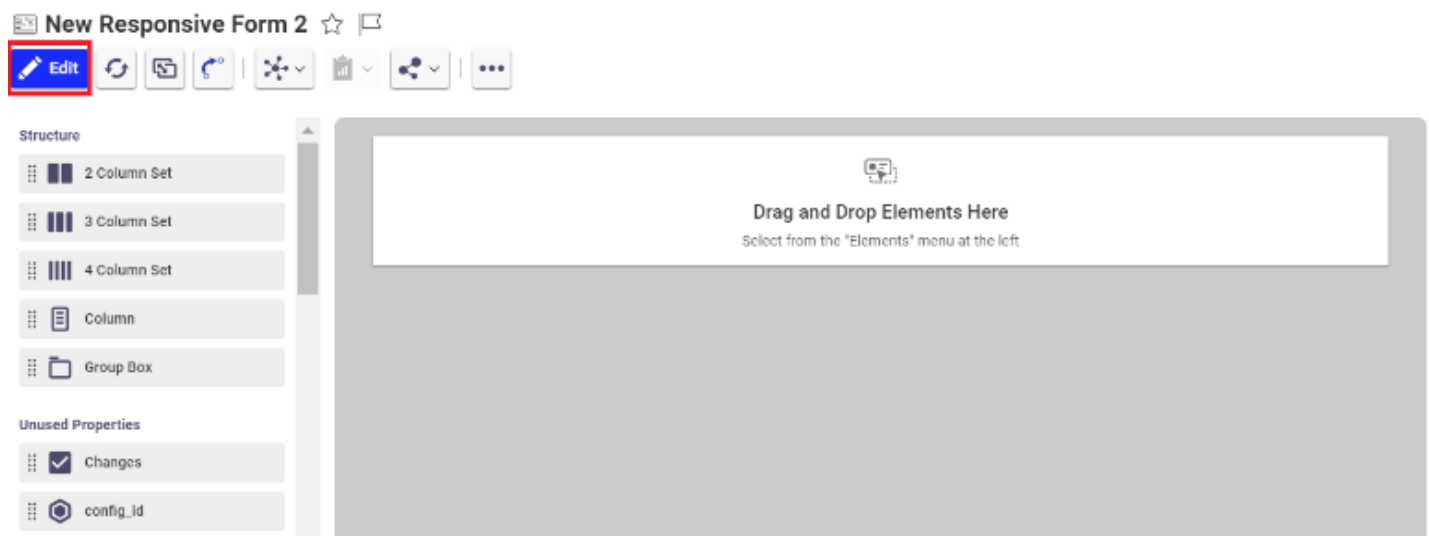
Design a Responsive Form

This section describes how to build a Form with the Responsive Form Designer.

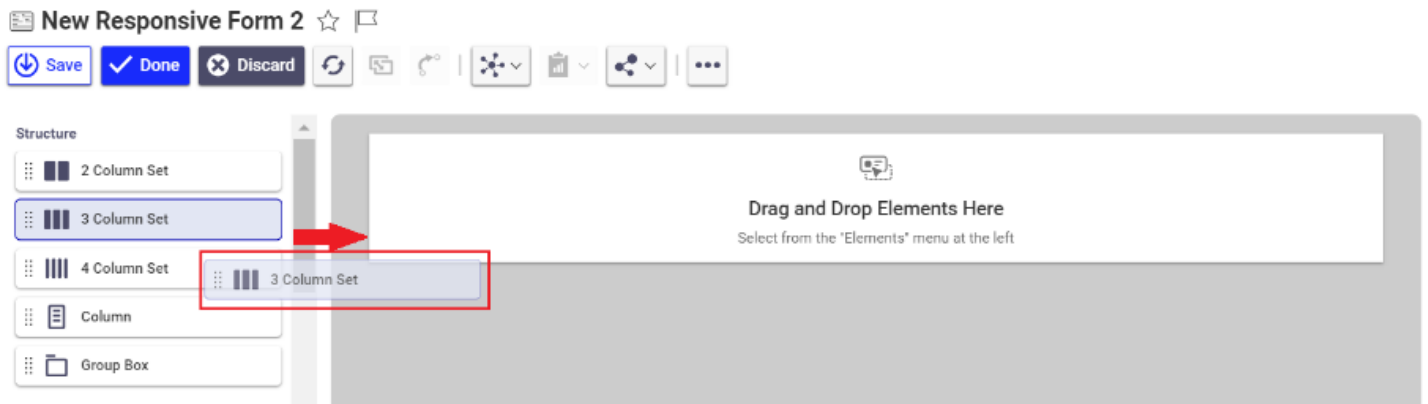
Add Structure with Column Sets

The following steps outline the process of using Column Sets and Columns to create a layout structure for a Responsive Form:

1. Click **Edit** in the Responsive Form Designer toolbar.

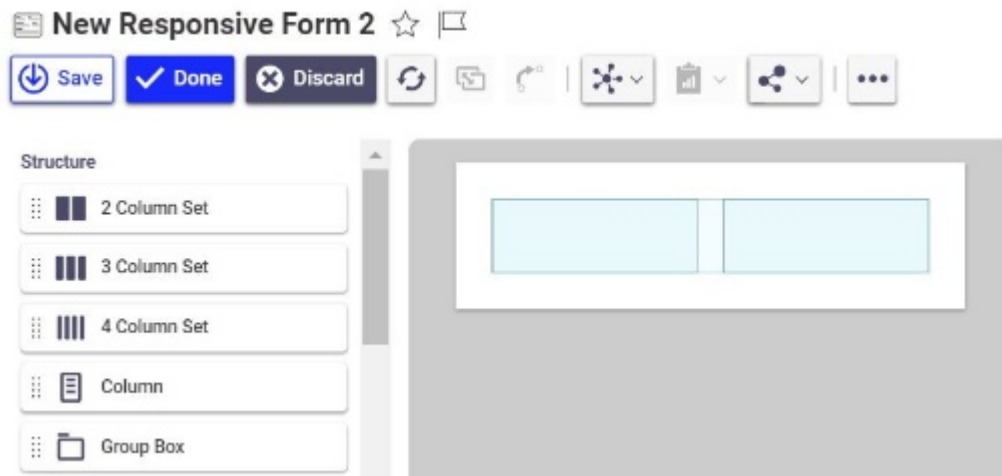


2. From the **Structure** list in the Element Pane, drag and drop the required **Column Sets**.

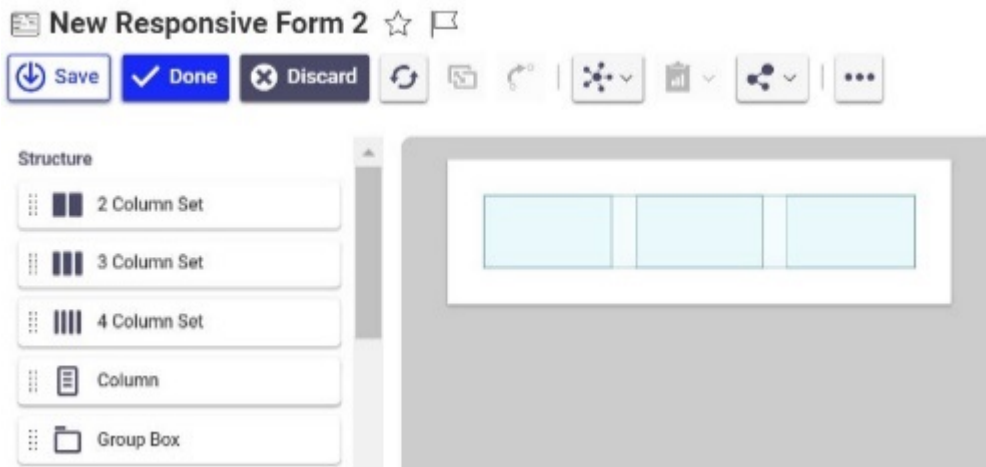


There are 3 Column Sets preconfigured with a number of Columns:

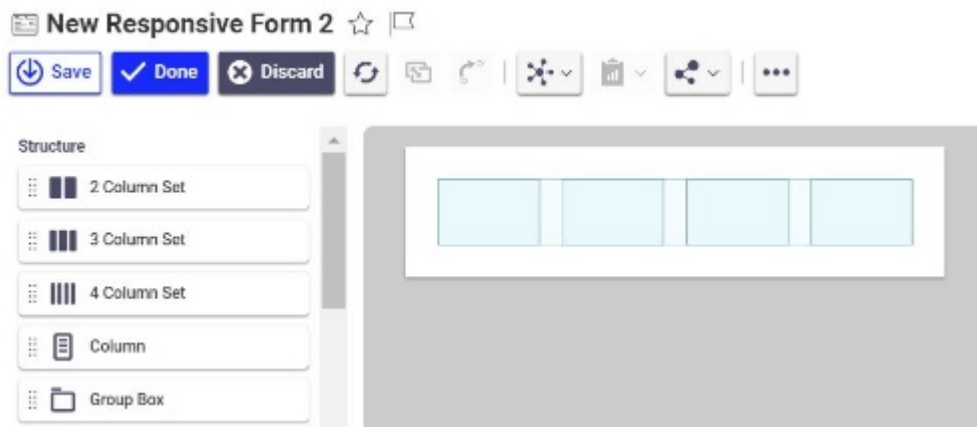
- **2 Column Set:** Adds a Column Set containing 2 Columns.



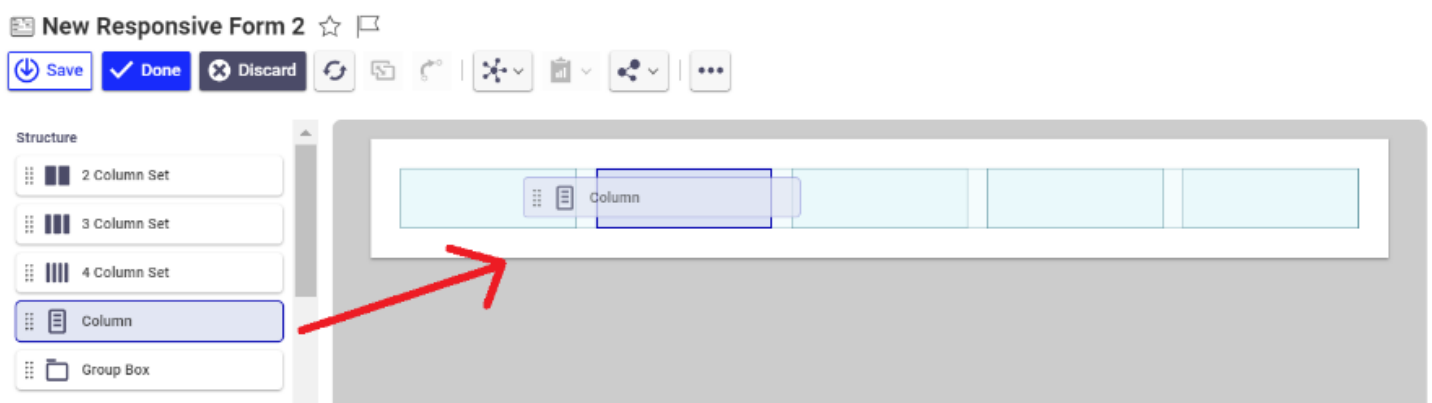
- **3 Column Set:** Adds a Column Set containing 3 Columns.



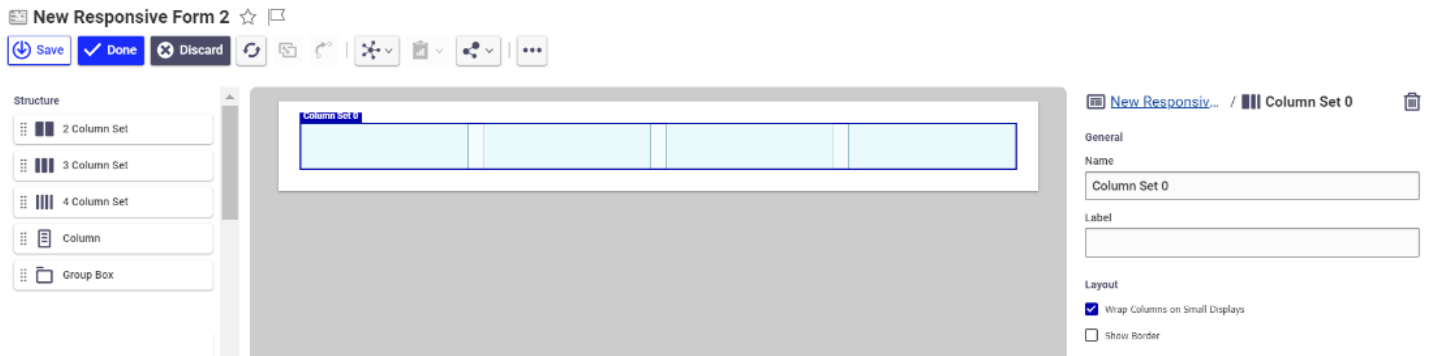
- **4 Column Set:** Adds a Column Set containing 4 Columns.



- To add more Columns to an existing Column Set in the designer, drag & drop a **Column** from the Element pane to a Column Set in the Canvas.



Once the required Column Sets are added, it should appear as follows:



4. Select a **Column Set** in the Canvas to view and edit the following properties in the Detail Pane:

- **Name:** Name to display when the element is selected in the Responsive Form Designer Canvas
- **Label:** Label to display to the user in the Responsive Form Designer Canvas and the rendered Form
- **Wrap Columns on Small Display:** True by default. Enabling this setting allows columns to stack vertically (or "wrap") instead of aligning horizontally when viewed on devices with limited horizontal space. This ensures content remains legible and user-friendly regardless of the screen size.
 - If the **"Wrap Columns on Small Display"** option is selected, the column should appear as follows:
 - On wider screen, column sets appear as follows:



- On smaller screen, column sets appear as follows:





- If set to false, the columns may shrink down to their minimum width when the Form shrinks, but they will not stack/wrap. If the width of the Form becomes smaller than the combined minimum width of the columns, a horizontal scroll bar will appear to allow the user to access all content.
- **Show Border:** False by default. Enabling this field displays a solid border around the Column Set.

Important

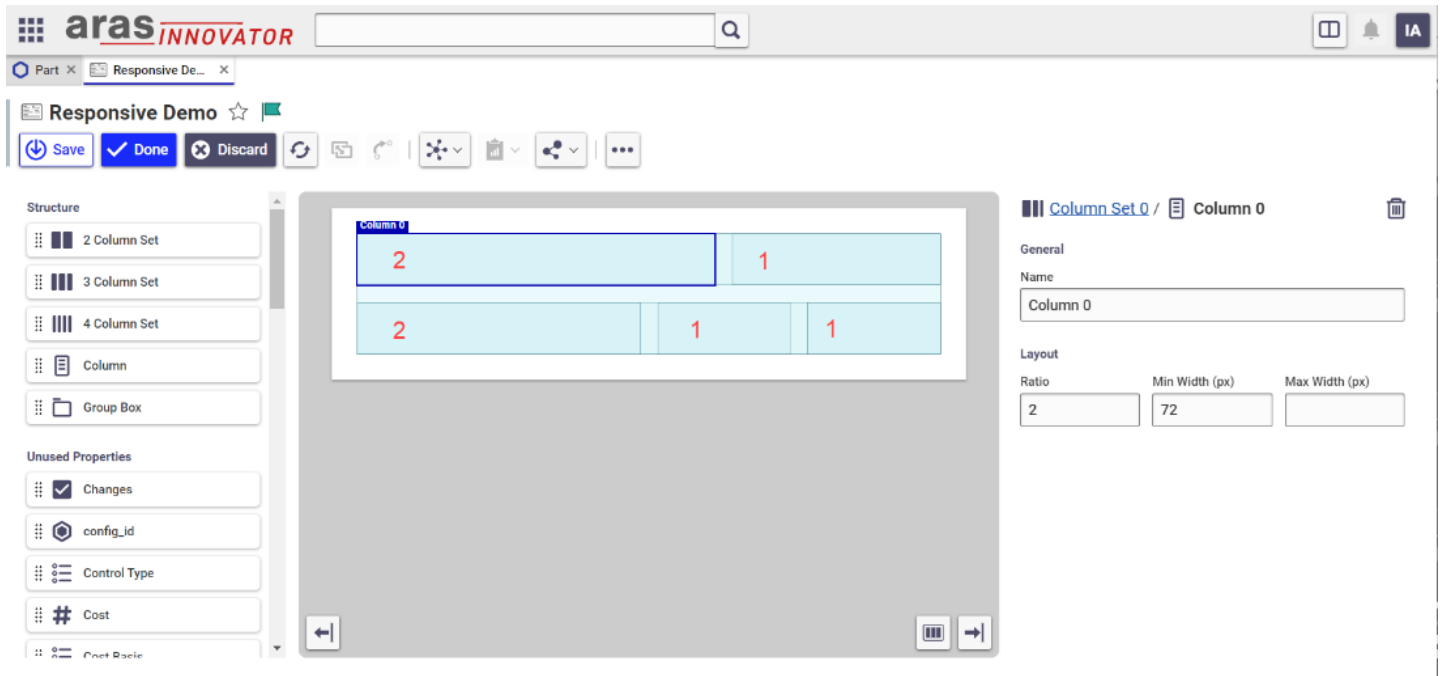
A Group Box is simply a Column Set preconfigured to display a label and a border. All steps described for Column Sets also apply to Group Boxes.

Adjust Layout with Column Properties

Once the Column Sets and Group box are placed, users can adjust the following:

- Name: Name of the column.
- Ratio: Describes the width of a Column relative to the other Columns in the same Column Set.
 - Example: A Column Set contains 2 Columns where one has a ratio of 2 and the other has a ratio of 1 (2:1). This means the first column will be twice as wide as the second column and occupies two thirds of the width of the Column Set.
 - Example: A Column Set contains 3 Columns with a ratio of 2:1:1. In this case the first column will be twice as wide as the second and third columns, meaning it occupies half the width of the Column Set.





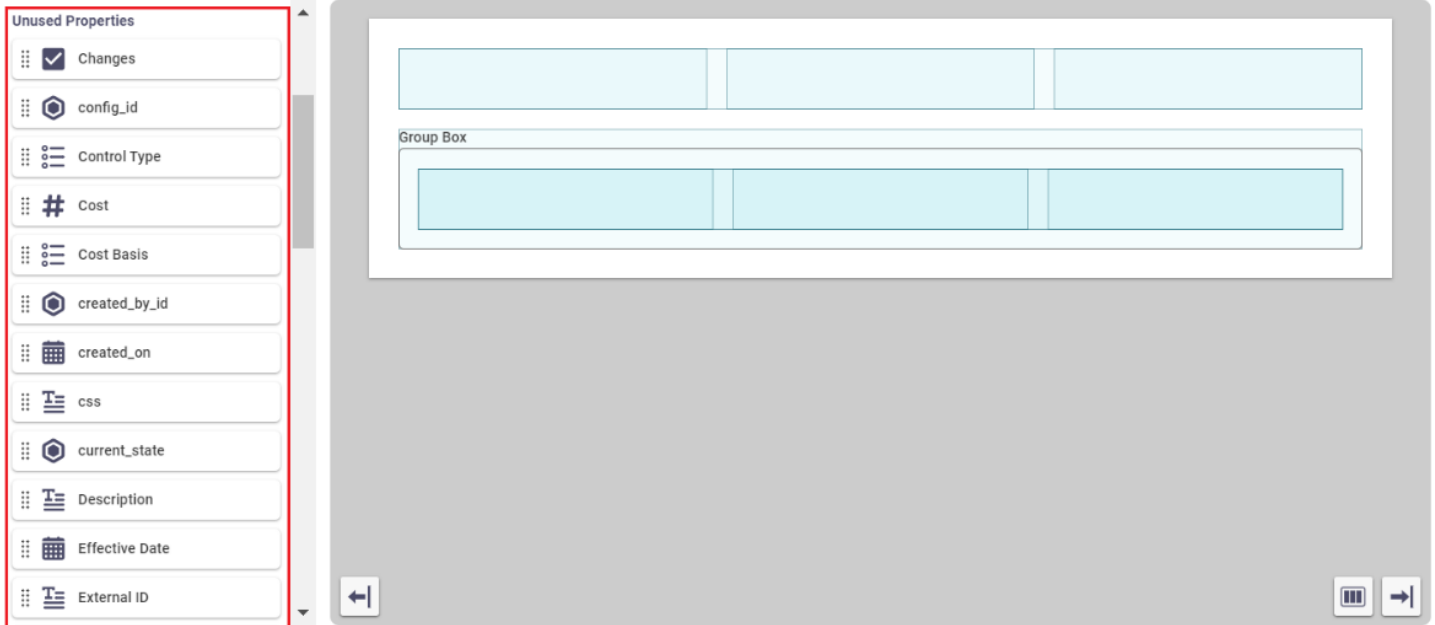
- Min Width (px): Minimum width that a Column will shrink to
- Max Width (px): Maximum width that a Column will expand to

When considering sizing behavior of Columns, it's important to note that a Column's width settings take priority over its ratio. For example, consider a Column Set with two Columns. Column A has a Min Width of 300px and Ratio of 2. Column B has a Min Width of 100px and a Ratio of 1. This ratio of 2:1 indicates that Column A should be twice as wide as Column B. However, the parent Column Set can shrink down until all child Columns hit their respective minimum widths. In this case, Column A will shrink to 300px and Column B will shrink to 100px – resulting in an actual ratio of 3:1. If maintaining the width of Columns in relation to each other is important for a Form's design, be sure any Min or Max Width follows the desired ratio.

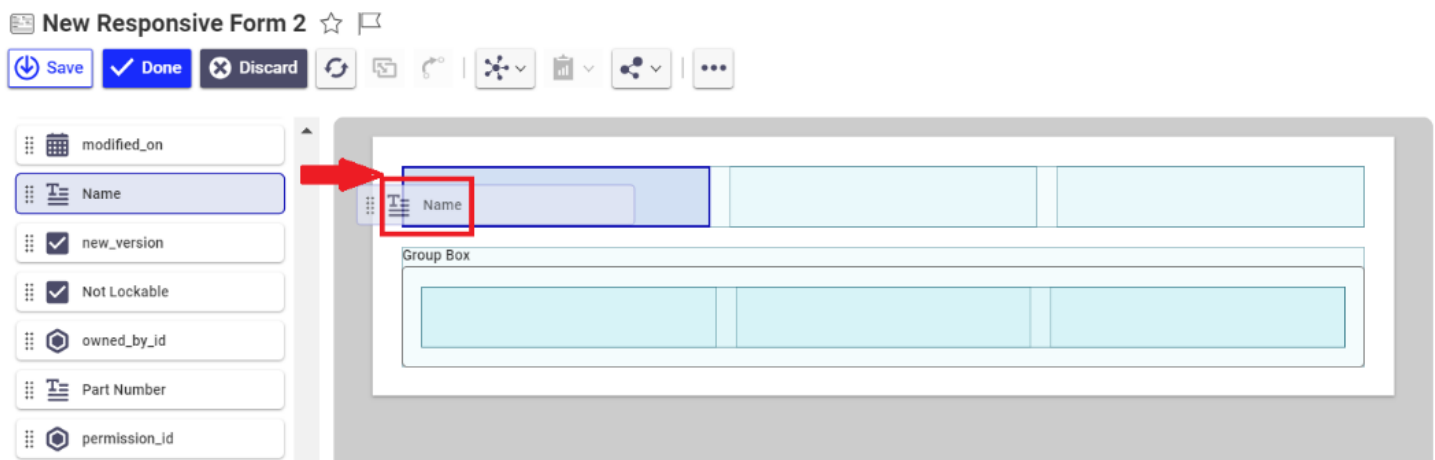
Add Properties to the Form

The following steps outline the process of adding fields to a Responsive Form:

1. In the Element Pane under **Unused Properties**, browse through the properties of the Form's parent ItemType. This list shows properties that have not yet been added to the Form.



2. Drag & drop a property into a Column. This will insert a field that has been pre configured for the selected property.



3. In the Detail Pane, view and update the properties of the field as needed.

- **Name:** Name displayed when the field is selected in the Canvas
- **Label:** Label displayed to the user in the Canvas and the rendered Form. Choose one of the following two options:
 - **Inherit from Property:** Display the label of the property, as defined on the ItemType
 - **Hide:** Don't display a label

- **Help:** Configure whether the field will render a help tooltip. Choose one of the following two options:
 - **Inherit from Property:** Display the property's tooltip, as defined on the ItemType.
 - **Hide:** Don't display a tooltip
- **Field Type:** Configure the type of control to display for the field. Options are limited to controls valid for the linked property's data type. For example, an Item control must be used if the property's data type is "Item."

The following table identifies data types that support multiple field types.

Field Type	Description
A concise input field	Text
A table for short entries (default)	Table
A larger input space for longer text pieces of text or paragraphs	Text Area
A list of options	List
A value selected from a dropdown list	Value List
A field with multiple values selected	Multi Value
A field with a selected value	Value
A field with a selected value and options	Combobox
A field with multiple values displayed as chips in the field. (default)	Multi Value



Options
are
displayed
as
checkbox
list
list
of
checkboxes
classification
Enables
users
to
change
max
display
and
Column
Min
Width
(px)

- **Resize:** Configure the sizing behavior of the field control. Valid only for resizable fields.
 - **Smart:** The field will resize automatically within its Min Height and Max Height parameters. The end user cannot change its size manually. (default)
 - **Manual:** The end user may manually change the field's height.
 - **None:** The field will not resize automatically, and the end user cannot change its size manually.
 - **Min Height:** The smallest height allowed (pixels). Valid only for resizable fields.
 - **Max Height:** The largest height allowed (pixels). Valid only for resizable fields.
4. Repeat the drag & drop process to add more fields to the Form. Columns may contain multiple fields.



The screenshot shows a form design canvas with a grid layout. A red box highlights a column containing two text input fields: 'Part Number' (with an asterisk indicating it's required) and 'Name'. To the right of this column are two more columns. The first column contains 'Unit' (a dropdown menu) and 'Weight' (a text input). The second column contains 'Type' (a text input with a copy icon). Below these columns is a 'Group Box' containing three text input fields: 'created_by_id', 'modified_by_id', and 'managed_by_id' (with a copy icon).

5. If a field should span the entire width of the Form, it may be dropped onto the Canvas, outside of any Column.

The screenshot shows the 'New Responsive Form 2' interface. On the left is a field list with various fields like 'COST BASIS', 'created_on', 'css', 'current_state', 'Description', 'Effective Date', 'External ID', 'External Owner', 'External Type', 'generation', 'id', 'is_current', and 'keyed_name'. The 'Description' field is highlighted. A red arrow points from this field to the canvas. On the canvas, a new 'Description' field is being added, spanning the full width of the form. The canvas also shows the same grid layout as the previous screenshot, with 'Part Number', 'Unit', 'Type', 'Name', 'Weight', 'created_by_id', 'modified_by_id', and 'managed_by_id' fields.

Important

Fields must be wrapped in a Column if they should have a Min Width or Max Width set. If a use case requires a single stack of fields with a minimum or maximum width, add the fields to a Column Set containing a single Column. Then, set the size parameters in the column.



Preview Responsive Form

The following steps outline the process of previewing the Responsive Form in the Responsive Form Designer:

1. To preview the Form without the blue guides for Columns, Column Sets, and Group Boxes, click **Hide Guides**.

New Responsive Form 2 ☆ 🚩

Edit ↺ ↻ ⌂ 🔍 📊 ⚙️ ⋮

modified_on

new_version

Not Lockable

owned_by_id

permission_id

Release Date

Released

Revision

State

Superseded Date

Team

Weight Basis

Part Number

Unit

Type

Name

Weight

Description

Group Box

created_by_id

modified_by_id

managed_by_id

Thumbnail

[Select an image...](#)

←  →

The preview of the Form appears as follows:

*** Part Number**

Unit

Type

Name

Weight

Description

Group Box

created_by_id

modified_by_id

managed_by_id

Thumbnail
[Select an image...](#)

- To preview how the Form will appear when spanning the full width of the Aras Innovator client, click the **Close Pane** buttons at the bottom left and bottom right of the Canvas. This will collapse the Element Pane and Detail Pane, respectively.

New Responsive Form 2

Edit

modified_on

new_version

Not Lockable

owned_by_id

permission_id

Release Date

Released

Revision

State

Superseded Date

Team

Weight Basis

New Responsiv...

General

Name

New Responsive Form 2

Item Type

Part



The Form should appear in the expanded Canvas as follows:

The screenshot shows a web form designer interface titled "New Responsive Form 2". The form is displayed on a canvas with a light blue background. The form layout includes:

- Part Number**: A text input field.
- Unit**: A dropdown menu.
- Type**: A dropdown menu.
- Name**: A text input field.
- Weight**: A text input field.
- Description**: A text area.
- Group Box**: A container with three text input fields: **created_by_id**, **modified_by_id**, and **managed_by_id**.
- Thumbnail**: A section at the bottom left with the text "Thumbnail" and a link "Select an image...".

The top toolbar contains buttons for "Edit", "Undo", "Redo", "Copy", "Paste", and other actions. The bottom toolbar contains buttons for "Open Pane" and "Show Guides".

3. To view the Element Pane and Detail Pane again, click the Open Pane buttons on the bottom left and bottom right of the Canvas.
4. To view the blue guides representing Columns, Column Sets, and Group Boxes, click the Show Guides button.

Save and View Responsive Form

The following steps outline the process of saving the Responsive Form and viewing the rendered Form in an Item View:

1. Once all changes are made to the Responsive Form, click **Done** in the Responsive Form Designer toolbar.



2. In the TOC, navigate to the ItemType associated with the new Responsive Form. Click Search.
3. From the search grid, open an item for viewing. The new tab should display the item with the Responsive Form.

Creating Rules for Responsive Forms

The following steps outline the process of creating Rules for Responsive forms:

1. From the Responsive Forms page, click **Manage Rules**.



The Form Rules appears as follows:

Responsive Part

Edit

↺

📄

🔄

🔗

📊

🔗

⋮

Part Number

Revision

State

Name

Brief description of the Part

Type

Make / Buy

Cost

Unit

Description

Roles

Assigned Creator

Designated User

Effective Date

Form Rules

Condition

Behavior

2. Click **Edit**.

Creating a New Rule

Creating a New Rule with Value of Property

The following steps outline the process of creating a new Rule based on the Value of Property:

1. Click New Rule.



The New Rules screen appears as follows:

New Rule

OK

Cancel

* Name

Condition

 Add Condition  Add Group

Behavior

Add Behavior

Name **The name of the Rule.**

Condition Describes when the rule will be applied.

Behavior Describes the actions that are performed when the rule is applied.

There are 2 options for adding Conditions:

- **Add Condition:** This button allows users to create multiple conditions in a single rule.
- **Add Group:** Allows the user to add a group of conditions under an existing condition with And/Or aggregator. OR is default.

Condition

The screenshot displays the 'Condition' configuration interface. At the top, there are two buttons: 'Add Condition' (with a plus icon) and 'Add Group' (with a group icon). Below these, the conditions are organized into a tree structure. The top-level group contains two conditions, each with a red exclamation mark and a close button. Below this, there is a sub-group labeled 'AND' (with a plus icon and a group icon). This sub-group contains two conditions, also each with a red exclamation mark and a close button. A mouse cursor is pointing at the 'AND' group. The interface is designed to allow users to build complex logical conditions by adding and grouping them.

Important

If a group is deleted, the children conditions will move to the top level.

2. Click Add Condition.



New Rule

OK

Cancel

* Name

Test Rule

Condition

Function

Value of property

Property

Type

Operator

is

Criteria

Assembly

Behavior

Add Behavior

Name	Description
Value of Property	Checks the value of a property of this item.
Property	Property of this item to be checked.
Operator	Type of comparison.
Criteria	Value used by comparison.

3. After selecting the Conditions for the new rule, Click Add Behavior.



 **Edit Rule**

OK

* Name

Hide Make-Buy for Assemblies

Condition

 Type is Assembly

×

Behavior

Property

Make / Buy

Display as

hidden

Add Behavior

Name Description

Property Property to be affected by this behavior.

Display as Hidden – Field is not displayed in the form.Normal – Default Field.Read Only – Field data cannot be edited by user.

- Click OK to add the rule.
- Click Done to save the Responsive Form.

Creating a New Rule with Member of Identity

The following steps outline the process of creating a New Rule based on the identity of the current user:

- Click New Rule.



The New Rules screen appears as follows:

New Rule

OK

Cancel

* Name

Test Rule

Condition

Add Condition

Behavior

Add Behavior

Name **The name of the Rule.**

Condition Describes when the rule will be applied.

Behavior Describes the actions that are performed when the rule is applied.

2. Click Add Condition.

New Rule

OK

Cancel

* Name

Test Rule

Condition

Function



Member of identity



Identity



Behavior

Add Behavior



Name	Description
Member of Identity	Checks whether the current user belongs to this identity
Identity	Identity to be checked.

3. After selecting the Conditions for the new rule, Click Add Behavior.

Behavior

Property

Type

Display as

hidden

Add Behavior

Name	Description
Property	Property to be affected by this behavior.
Display as	Hidden – Field is not displayed in the form.Normal – Default Field.Read Only – Field data cannot be edited by user.

- 4. Click OK to add the rule.
- 5. Click Done to Save the Responsive Form.

Adding Existing Rules

Rules that are created in a Responsive Form will be available to add to any other Responsive Forms.

- 1. Click **Add Rule**.



The Add Rule dialog appears as follows:



Select Items

Rules

Q

Simple

Name ↑	Condition Type	Condition [...]	Behavior Type	Behavior [...]
▼	0 options ...		0 options ...	
Hide Make-Buy for Assemblies	Property	{{"id": "A33DDA..."}}	Display	{{"Property": {"t..."}}
Planning Board Can't Edit Description	Identity	{{"id": "F1E36E..."}}	Display	{{"Property": {"t..."}}

< Prev

Next >

Page: 1 of 1

2 Results

OK

Cancel

- Click the Search button to display all rules that have been created.
- Select the rule that is being added to the Responsive Form.
- Click OK.
- Click Done to save the Responsive Form.

Important

Double-Clicking a rule in the search dialog will add the Rule to the list of Form Rules and dismiss the dialog.

Edit Rules

The following steps outline the process of editing a Rule:

- Select the Rule that needs to be edited.
- Click Edit Rule.

←

Form Rules

Name	Condition	Behavior
1 Hide Make-Buy for Assem...	Type is Assembly	Make / Buy is hidden
2 Planning Board Can't Edit D...	Identity is Planning Board	Thumbnail is read only



3. Make the edits to the rule as needed.

 **Edit Rule** OK

* Name

Hide Make-Buy for Assemblies

Condition

Type is Assembly

Behavior

Property

Make / Buy

Display as

hidden

Add Behavior

Important

Rules may be used in multiple forms. Editing a rule will alter the rule for all forms that currently use it.

4. Click Done to save the Responsive Form.

Removing Rules

The following steps outline the process of removing a Rule from a Form:

1. Select the Rule to remove from the Responsive Form.
2. Click Remove Rule from Form.

  **Form Rules**    

Name	Condition	Behavior
1 Planning Board Can't Edit D...	Identity is Planning Board	Thumbnail is read only
2 Hide Make-Buy for Assem...	Type is Assembly	Make / Buy is hidden



3. Click Done.

Important

Removing a Rule does not permanently delete it. Rules that have been removed are still available by searching in Add Rules.

Reordering the Rules

To reorder the rules, drag and drop the rules in the form rules table.

The screenshot shows a 'Form Rules' interface. At the top, there's a navigation bar with a back arrow, a 'Form Rules' title, and four icons: a plus in a box, a gear with a plus, a box with a pencil, and a box with a red X. Below this is a table with three columns: 'Name', 'Condition', and 'Behavior'. The table contains two rules. The first rule is '2 Hide Make-Buy for A...' with the condition 'Type is Assembly' and the behavior 'Make / Buy is hidden'. The second rule is '1 Planning Board Can't...' with the condition 'Identity is Planning Board' and the behavior 'Thumbnail is read only'. The rules are numbered 2 and 1 respectively, indicating they can be reordered.

	Name	Condition	Behavior
2	Hide Make-Buy for A...	Type is Assembly	Make / Buy is hidden
1	Planning Board Can't...	Identity is Planning Board	Thumbnail is read only

Important

To reorder rules, the form must be in Edit mode.

Form Packaging

Packaging a Responsive Form

Packaging metadata like ItemTypes and Forms is an important part of managing an Aras Innovator instance. The Responsive Form Designer has a dedicated feature to simplify packaging. The following steps outline the process of adding a Responsive Form to a package in a few scenarios.

The Form has already been packaged in Responsive Mode

No action is required if the Form has already been added to a package from Responsive Mode. All data related to the Responsive Form definition has been added to the package definition. The Export Utility will export the current configuration, even if you've updated the Form since it was packaged.

The Form has been packaged in Classic Mode but not Responsive Mode



In some cases, a Responsive Form may have been previously packaged as a Classic Form. In this case, the administrator needs to execute a one-time update to ensure the package definition contains all the necessary metadata for the Form's responsive design. The following steps outline the process of updating a package definition containing a Classic Form that was converted to a Responsive Form: 1. Click on the More icon.

New Responsive Form 2 ☆ 🚩

Edit ↻ 📄 🔗 🌐 ⚙️ 🗑️ 🔄 ⋮

modified_on

☒ new_version

☒ Not Lockable

☐ owned_by_id

☐ permission_id

☐ Release Date

☒ Released

☐ Revision

☐ State

Part Number

Unit

▼

Type

Name

Weight

Description

Group Box

created_by_id

modified_by_id

managed_by_id

2. Select Update Package Definition.

⋮

Create New Form

Properties

Permissions >

Delete >

Add to Package Definition

Update Package Definition

Important



It's only necessary to run the Update Package Definition action once after switching a packaged Classic Form to a Responsive Form. It does not need to be run every time a Responsive Form is modified.

The Form has never been packaged

1. Open the Form in the Responsive Form Designer.
2. Make sure the Form is saved and the designer is in view mode – there will be an Edit button in the toolbar instead of the Save and Done buttons.
3. Click on the **More** icon.

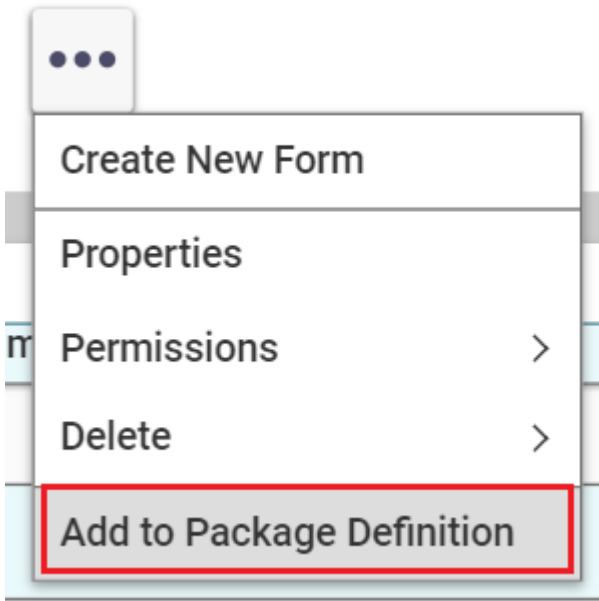
New Responsive Form 2 ☆ 🚩

✎ Edit ↶ ↷ 📄 📋 📌 ⋮

modified_on
new_version
Not Lockable
owned_by_id
permission_id
Release Date
Released
Revision
State

Part Number Unit Type
Name Weight
Description
Group Box
created_by_id modified_by_id managed_by_id

4. Select **Add to Package Definition**.



The **Add selected item(s) to a package** dialog box appears.

Add selected item(s) to a packa... ✕

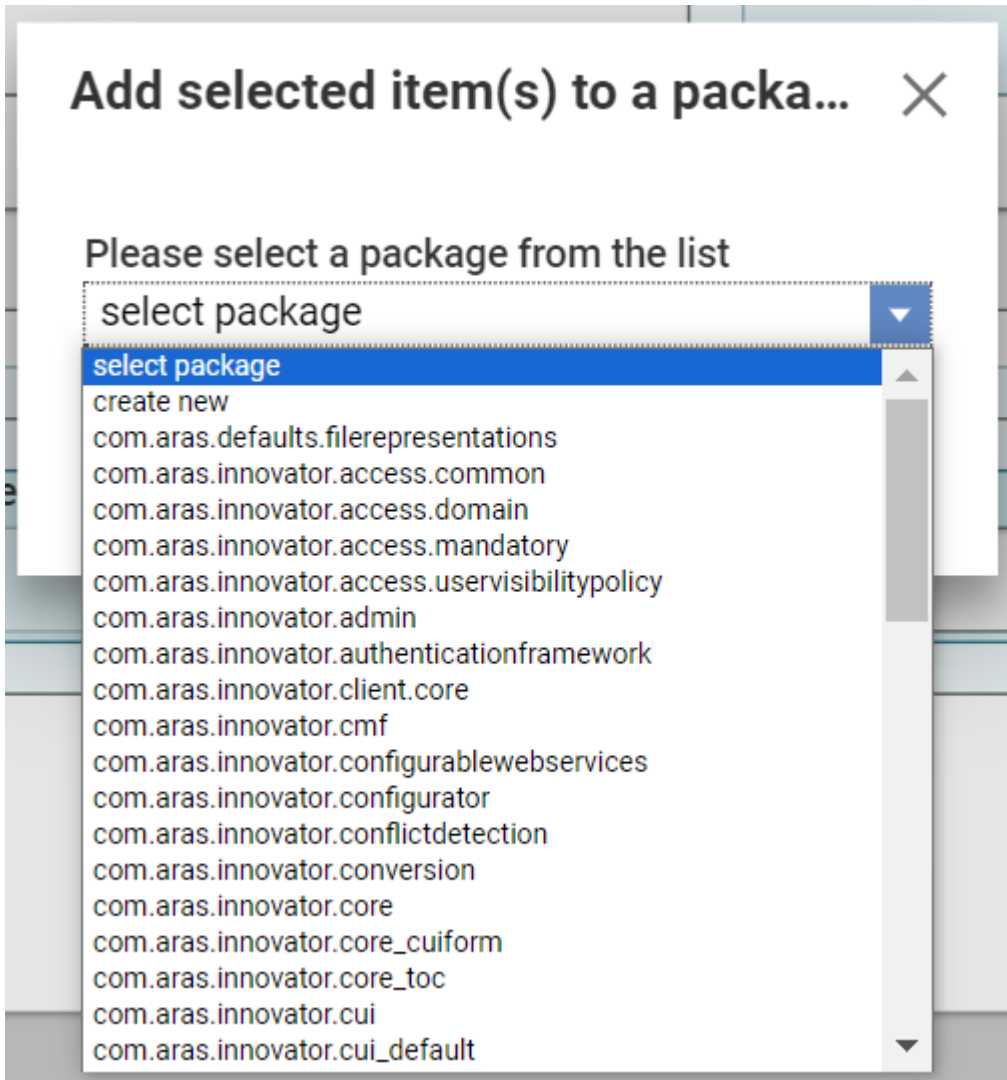
Please select a package from the list

select package ▼

OK

Cancel

5. In the **Select Package** dropdown, either select the existing required package or create a new package.



6. Click **Ok**.

Add selected item(s) to a packa... ✕

Please select a package from the list

 ▼

Now the Form item and the necessary related data will be included next time the specified package is exported.

Important

It's important to use the version of the Aras Export Utility that matches the version of Aras Innovator you are exporting packages from. Starting in Release 28, the Export Utility includes an enhancement that can make exporting and importing Forms easier.

