TEMPLATES

View online: https://www.construct.net/en/make-games/manuals/construct-3/project-primitives/objects/templates

Templates help managing instances in larger projects. The main uses for them are:

1. Conveniently updating properties for multiple instances across the project in the editor. This can prove to be time consuming and prone to error if done individually. Using templates you can declare a single instance to be the source **template** and other instances of the same object type to be **replicas** of it. After doing so, changing the template will also change all replicas, even across layouts.
2. A template can be used as a preset when creating an instance at runtime using the **Create Object** action. Doing this is can help avoid needing lots of initialization actions and makes it easier to make changes to existing presets or add new ones later on.
3. Using templates it is possible to define different hierarchies in the editor, and then choose which one to create at runtime.

Terms

1. **Template:** an instance that has been set to be the source of property values for other instances to use. These include common properties, plugin properties, instance variables, behavior properties and effect parameters. A template can also be used to decide what values a new instance created at runtime should take when using the **Create Object** action. Modifying a template in the Layout view or through the Properties bar will immediately be reflected in all instances which are replicas of it.
2. **Replica:** an instance that has been set to use an instance already defined as a template. A replica takes its values from the source template unless they are explicitly modified, at which point the replica's own values are used instead.

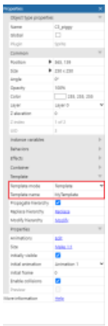
Template properties

The Properties Bar shows relevant properties in the *Template* section. The properties that can be shown there are:

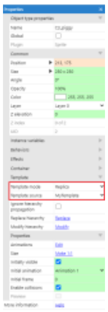
- **Template mode:** an instance can be turned into a template by selecting the *Template* value from the drop-down list. If there are any instances in the object type which have already been set to be a template, then the *Replica* value will also be available in the drop down menu.
- **Template name:** the name to identify a template. This is only shown when **Template mode** is set to *Template*.
- **Template source:** the name of the template a replica is using. This is only shown when **Template mode** is set to *Replica*, and only lists templates of the same object type.

The three main cases of using templates in the Properties Bar are shown in the images below. Click the thumbnails to expand them.

Properties for an instance not using templates.



Properties for an instance set to be a template.

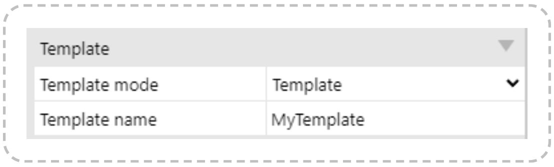


Properties for an instance set to be a replica. Note the highlighting on the properties to identify which are from the template, and which have been overridden for this instance.

Setting up templates and replicas

To start using this feature the first thing you will need to do is set an instance to be a template. To do that, follow these steps:

1. Select the instance you want to be a template in the Layout View.
2. Pick the *Template* option from the **Template mode** dropdown in the Properties bar.
3. The **Template name** input will appear.
4. Give the template a name using the **Template name** text input.

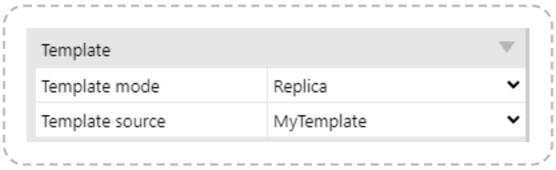


Template	
Template mode	Template
Template name	MyTemplate

That's all you need to do to set an instance to be a template.

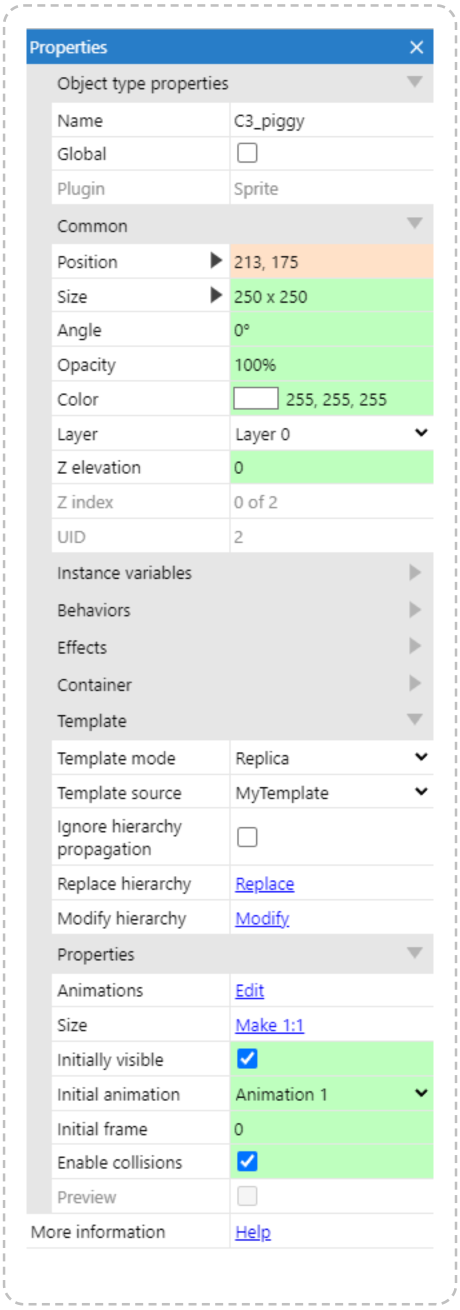
Now let's see how to set other instances to use the template in the editor:

1. Select the instance you want to be a replica of the template in the Layout View. The instance must be of the same object type as the template.
2. Pick the *Replica* option from the **Template mode** dropdown.
3. The **Template source** dropdown will appear.
4. Select the template the replica instance should use from the **Template source** dropdown.



Template	
Template mode	Replica
Template source	MyTemplate

After doing that you will notice that some of the properties in the Properties bar are highlighted. This indicates that the instance is taking that value from the source template.



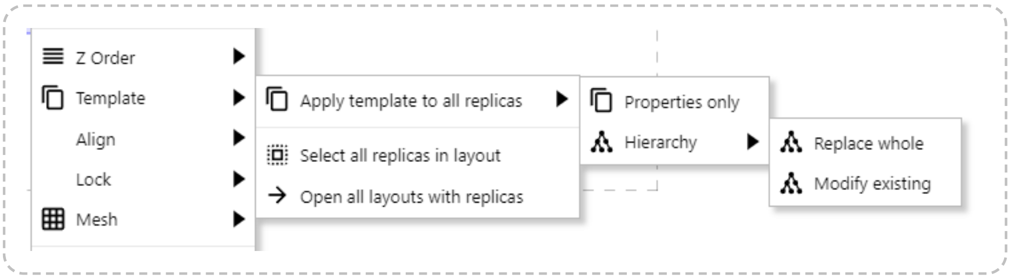
Properties	
Object type properties	
Name	C3_piggy
Global	□
Plugin	Sprite
Common	
Position	213, 175
Size	250 x 250
Angle	0°
Opacity	100%
Color	□ 255, 255, 255
Layer	Layer 0
Z elevation	0
Z index	0 of 2
UID	2
Instance variables	
Behaviors	
Effects	
Container	
Template	
Template mode	Replica
Template source	MyTemplate
Ignore hierarchy propagation	□
Replace hierarchy	Replace
Modify hierarchy	Modify
Properties	
Animations	Edit
Size	Make 1:1
Initially visible	☑
Initial animation	Animation 1
Initial frame	0
Enable collisions	☑
Preview	□
More information	Help

From now on, when the template is modified, all replicas using it will reflect the changes. If a replica is modified individually, the affected property will stop taking the value from the template and instead will take the value from the replica itself. To indicate this, the property will be highlighted in a different color.

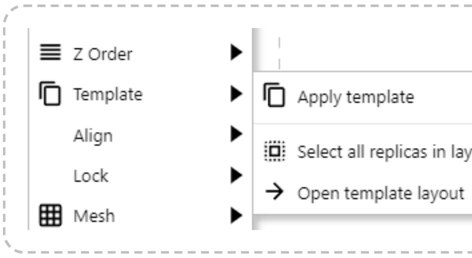
Miscellaneous options

Layout view context menu options

Template context menu


Z Order
Template
Align
Lock
Mesh
Apply template to all replicas
Select all replicas in layout
Open all layouts with replicas
Properties only
Hierarchy
Replace whole
Modify existing

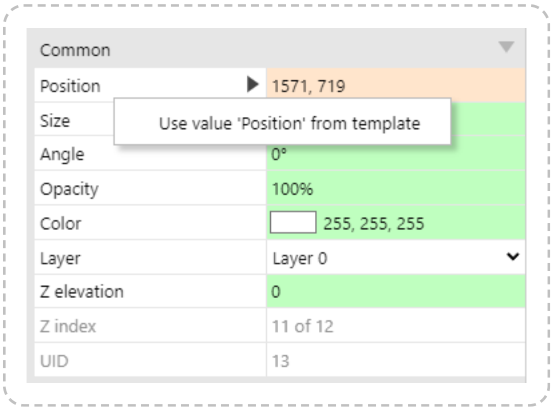
Replica context menu



- **Template instance context menu options:**
  - Apply template to all replicas ▶ Properties only forces all replicas using this template to use the values of the template.
  - Apply template to all replicas ▶ Hierarchy ▶ Replace whole replaces the whole hierarchy of each replica with that of the template. This creates and deletes instances as necessary.
  - Apply template to all replicas ▶ Hierarchy ▶ Modify existing updates the existing hierarchy of each replica to match as best as possible the hierarchy of the template. This does not create or delete any instances.
  - Select all replicas in layout selects all replicas of this template in the layout.
  - Open all layouts with replicas opens all layouts with replicas of the selected template.
- **Replica instance context menu options:**
  - Apply template ▶ Properties only forces the selected replica to use the values from the template.
  - Apply template ▶ Hierarchy ▶ Replace whole replaces the whole hierarchy of the replica with that of the template. This creates and deletes instances as necessary.
  - Apply template ▶ Hierarchy ▶ Modify existing updates the existing hierarchy of the replica to match as best as possible the hierarchy of the template. This does not create or delete any instances.
  - Select all replicas in layout selects all replicas of this template in the layout.
  - Open template layout opens the layout containing the template the selected replica is using.

Properties bar context menu options

When a replica's property is modified and so starts using its own value, it's possible to individually set it back to use the template value by right clicking on the property name.



Common	
Position	1571, 719
Size	
Angle	0°
Opacity	100%
Color	255, 255, 255
Layer	Layer 0
Z elevation	0
Z index	11 of 12
UID	13

Other Properties bar options

Propagate hierarchy

This property is only shown for template instances, if toggled all hierarchy related changes will be propagated automatically to the replicas. Turned on by default.

Ignore hierarchy propagation

This property is only shown for replica instances, if toggled any changes automatically propagated from a template are ignored. Turned off by default.

Replace hierarchy

When used from a template, will cause all the replicas to match the template's hierarchy. When used on a replica, that replica matches it's templates hierarchy. Creates and destroys instances as needed. Using this option ignores **Propagate hierarchy** and **Ignore hierarchy propagation** settings.

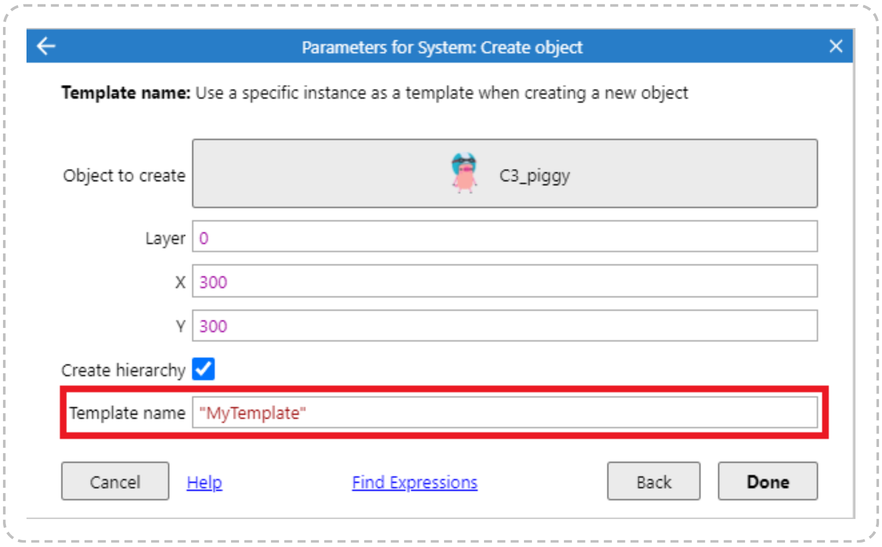
Modify hierarchy

When used from a template, will cause all the replicas to match the template's hierarchy as best as possible. When used on a replica, that replica matches it's templates hierarchy as best as possible. No instances are created or deleted. Using this option ignores **Propagate hierarchy** and **Ignore hierarchy propagation** settings.

Runtime actions

The **Create Object** system action has an additional **Template name** parameter. When specified the new instance being created will be a clone of the template instance, rather than an arbitrary one.

As well as using the properties of the template instance, the new instance will also use the hierarchy from the template if one exists. This means that it is possible to create different hierarchies in the editor that use the same type of instance as a root, while also being able to choose which one to create at runtime.



Parameters for System: Create object

Template name: Use a specific instance as a template when creating a new object

Object to create: C3_piggy

Layer: 0

X: 300

Y: 300

Create hierarchy ☑

Template name: "MyTemplate"

If the provided template name is not available, the *Create Object* action will behave as if the parameter wasn't provided (using an arbitrary instance instead).

Dragging and dropping from the Project Bar

Instances can be created by dragging and dropping an object type in the Project Bar to the Layout View. Normally this creates a default instance based on another instance in the project. However it may be desirable to control which specific kind of instance is created when you do this. Templates can also be used to determine which instance is created in this case.

When you create a template, selecting the object type in the Project Bar will now show two additional properties in the Properties Bar:

- **New instance template:** the template to use when creating a new instance from the Project Bar. You can choose any template in the project for the object type. The created instance will copy all its properties from the template.
- **New instance replica:** check this to set an instance created from the Project Bar to be a replica of the template. This means it will automatically update when the template is modified. If unchecked, the new instance copies the template's properties, but it is not made a replica, so it will not automatically update when the template is modified.

Also note that creating instances from the Project Bar will respect the *Propagate hierarchy* option of the template. If enabled and you create the root instance of a hierarchy from the Project Bar, then it will automatically also create the rest of the hierarchy as well.

Other considerations

- By default, replicas do not use the position properties of the template. In most cases it is more useful for these two properties to remain individual for each replica.
- At runtime, replicas are no longer connected to their templates as they are in the editor. This means that changing a template at runtime will not affect the replicas. This applies for properties and hierarchy changes.