TIMELINE CONTROLLER

View online: https://www.construct.net/en/make-games/manuals/construct-3/plugin-reference/timeline-controller

The **Timeline controller** object allows timelines to be controlled in event sheets.

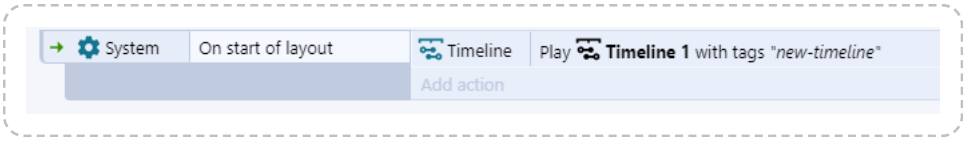
Tagging

Much like tweens, timelines can be optionally tagged when they are played using one of the **Play** actions. Tags are useful to later control a timeline (or multiple timelines sharing the same tags) with some of the other actions, conditions or expressions.

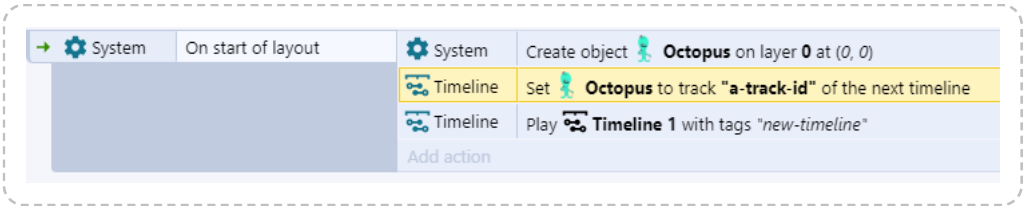
Setting instances to play

In the simplest case, a timeline will affect the instances that were used to create the timeline in the editor. Using the **Set Instance** action it is possible to use different instances to the ones used in the editor. Below are some short examples to help illustrate how this action works.

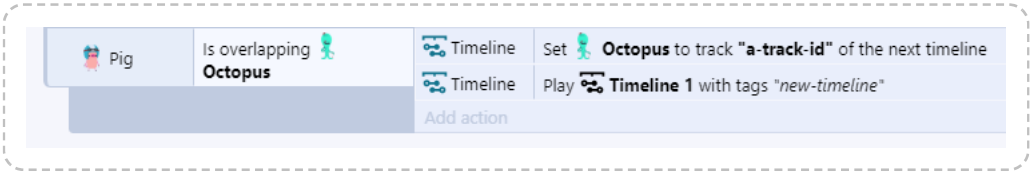
In the below example the timeline plugin **Play** action is used by itself on startup. This plays the timeline affecting the instances used in the editor to create the timeline. The timeline is tagged **"new-timeline"**.



In this example the timeline plugin **Play** action is used together with the system plugin **Create Object** action and the timeline plugin **Set Instance** action. This plays the timeline affecting the newly created instance. The new instance will be used in the track with ID **"a-track-id"** and the timeline is tagged **"new-timeline"**



This example is similar to the last one, but instead of creating a new instance from scratch, the one picked by a collision event is used.



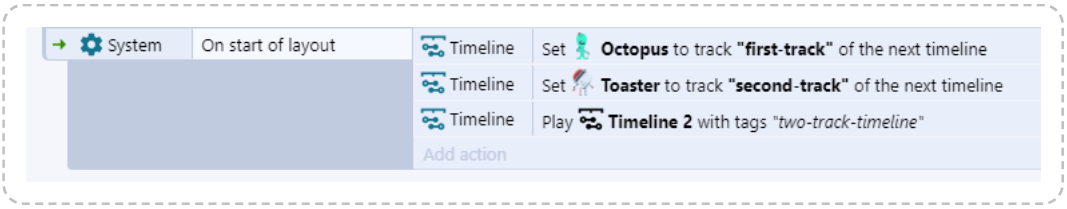
About picking

When using the **Play** action after one or more **Set Instance** actions, it is possible that one or more similar timelines will start playing. This will depend on the amount of currently picked instances for each given object type.

In the example below a timeline will be played for each group of instances.

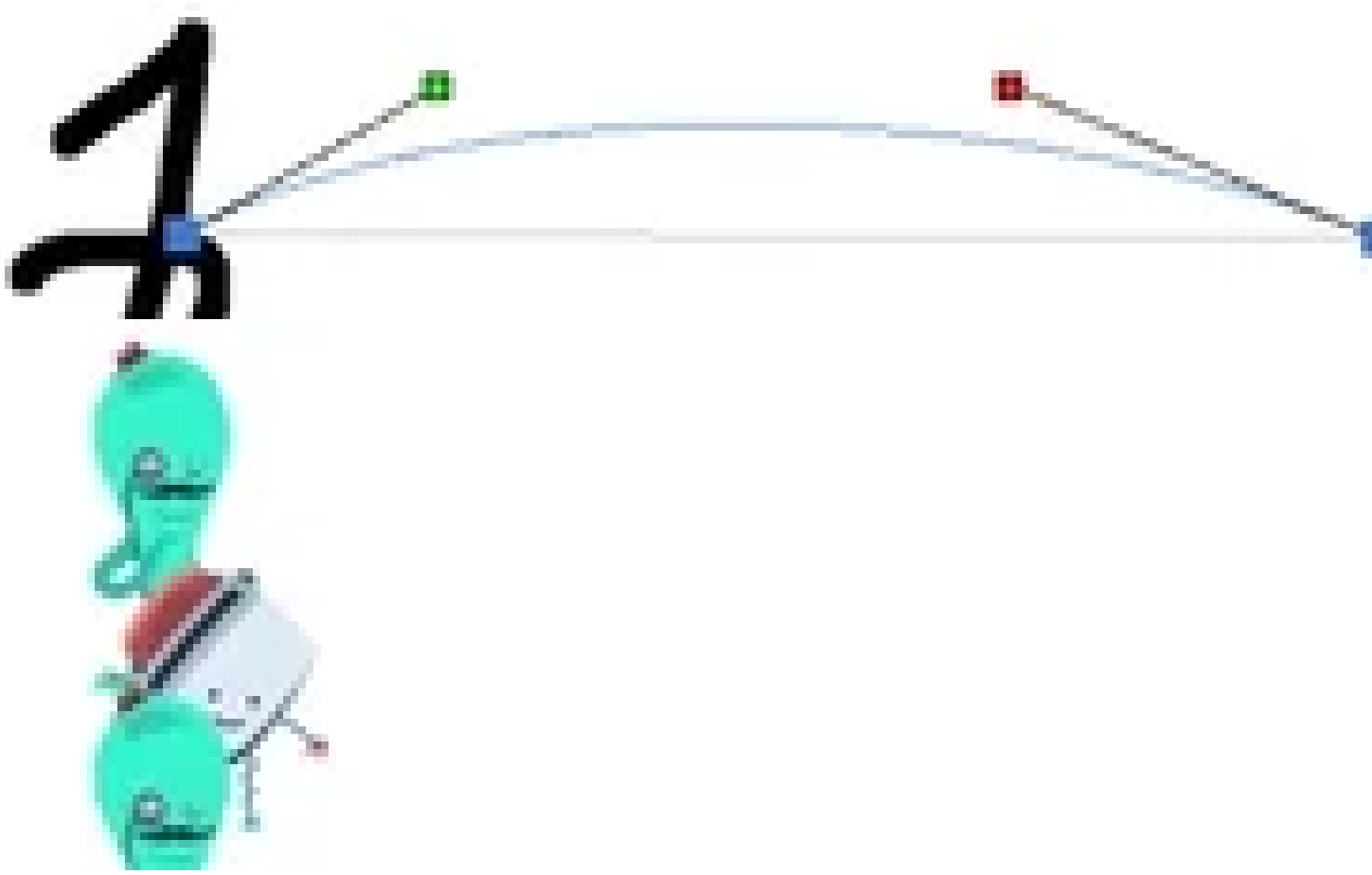
This is the preview of the timeline in the editor. It has two different tracks and placeholder instances.

Using the **Set Instance** action we specify to play a timeline for each picked group of instances at the start of the layout



Lastly the preview of the layout shows that two different timelines where created to accommodate for the four instances found at the start of the layout.

Omitting the track ID



First instance properties

Master track	
Name	Sprite - 5
Animation mode	Default
Result mode	Default
Ease	Default
Path mode	Default
Enabled	☑
Track ID	
Editor	
Visible	☑
Locked	☐
Show UI elements	☑
More information	Help

Second instance properties

Master track	
Name	Sprite2 - 6
Animation mode	Default
Result mode	Default
Ease	Default
Path mode	Default
Enabled	☑
Track ID	
Editor	
Visible	☑
Locked	☐
Show UI elements	☑
More information	Help

Events

System	On start of layout	Timeline	Set **octopus** to track "" of the next timeline
		Timeline	Set **toaster** to track "" of the next timeline
		Timeline	Play **Timeline 1** with tags *"new-timeline"*
			Add action

In this particular case the **octopus** sprite is used in the first track while the **toster** sprite is used in the second track. Because no track IDs are used, order is assumed to be the same as that defined in the editor.

Unsetting all instances

Sometimes it might be necessary to clear the state the **Timeline Controller** uses in order to use different instances in a timeline. You might find that some state was previously set but no timeline was played to use it. Such a scenario is likely to cause subsequent uses of **Set instance** and **Play** to produce unexpected results. In those cases use the **Unset instances** action to clear the **Timeline Controller** before attempting to use **Set Instance** and **Play** together again.

Timeline controller conditions

Is any playing

True if any timeline is playing.

Is playing

True if a specified timeline is playing, given by its tag.

Is any paused

True if any timeline is paused.

Is paused

True if a specified timeline is paused, given by its tag.

On started

Triggered when a timeline starts playing, given by its tag. Use the **"Type"** parameter to specify when the trigger should take place. *"Any"* will execute the trigger any time a timeline is started, *"Starting"* will execute the trigger only on the initial playback of the timeline and *"Resuming"* will execute the trigger only when the timeline is resumed after being paused.

On any started

Triggered when any timeline starts playing. Use the **"Type"** parameter to specify when the trigger should take place. *"Any"* will execute the trigger any time a timeline is started, *"Starting"* will execute the trigger only on the initial playback of the timeline and *"Resuming"* will execute the trigger only when the timeline is resumed after being paused.

On finished

Triggered when a timeline finishes playback, given by its tag.

On any finished

Triggered when any timeline finishes playback.

On keyframe reached

Triggered when a master keyframe with certain tags is reached during playback. The keyframe can be identified by whether it matches any of the given tags, or if it has all of the given tags. Separate tags with spaces.

On any keyframe reached

Triggered when any master keyframe is reached during playback. The *KeyframeTags* expression has a string of the keyframe's *Tags* property.

On time set

Triggered when the time of a timeline is set with the **Set Time** action.

Timeline controller actions

Play

Start playing a timeline, with tags to identify this playback.

Play all

Play all the timelines in the layout.

Pause

Pause a timeline by its tag. Paused timelines can subsequently be resumed.

Pause all

Pause all currently playing timelines.

Resume

Resume a paused timeline by its tag.

Resume all

Resume all paused timelines.

Stop

Stop a timeline and reset it to its initial state.

Stop all

Stop all timelines, resetting them all to their initial state.

Set time

Set the current playback time of a timeline in seconds. **Note:** you can also use a string with a keyframe tag for the *Time* parameter, in which case the time is set to the position of that keyframe. If the timeline is playing when its time is set, playback is stopped.

Set playback rate

Set the playback rate of a timeline. 1 is normal speed, 0.5 is half speed, etc. Negative numbers will play in reverse.

Set instance

Set an instance to be used for the next timeline playback. The instance can be of a different type to the one used in the editor. The instance will be set to the track with the corresponding track ID. The track ID can also be left empty in which case it uses the first track in the timeline. It can also be used repeatedly with an empty ID to keep setting the tracks in the timeline in sequence. When the timeline is played it will then affect this instance instead of the one used in the editor. Playback can be controlled by using unique tags when playing the timeline.

Unset instances

Clears all state associated with calls to **Set instance**.

Timeline controller expressions

Time(nameOrTags)

Retrieve the current time of the first matching timeline by either name or tags.

Progress(nameOrTags)

Retrieve the progress of the first matching timeline by either name or tags, returning a value in the range [0, 1].

TotalTime(nameOrTags)

The total time of the first matching timeline by either name or tags.

KeyframeTags

In a On keyframe reached or On any keyframe reached trigger, a string with the Tags property of the keyframe that was reached.

KeyframeTime

Get the time of a keyframe providing a timeline name (or tags) and keyframe tags.

TimelineName

In a trigger, a string with the name of the relevant timeline.

TimelineTags

In a trigger, a string with the tags of the relevant timeline.

Value(timelineNameOrTags, valueTrackNameOrId)

Retrieve the value of a value track by specifying a timeline name or tags and a value track name or track id. If no matching value track is found, the expression returns 0.

Ease(easeName, value)

Return the result of an ease function at a given value in the range 0-1. The ease name can be either a built-in ease, or the name of a custom ease in the project. A list of the names of built-in eases is included below.

Built-in ease names

These are the names for built-in ease functions that can be used with the TimelineController *Ease* expression.

`"linear"`		
`"in-sine"`	`"out-sine"`	`"in-out-sine"`
`"in-elastic"`	`"out-elastic"`	`"in-out-elastic"`
`"in-back"`	`"out-back"`	`"in-out-back"`
`"in-bounce"`	`"out-bounce"`	`"in-out-bounce"`
`"in-cubic"`	`"out-cubic"`	`"in-out-cubic"`
`"in-quadratic"`	`"out-quadratic"`	`"in-out-quadratic"`
`"in-quartic"`	`"out-quartic"`	`"in-out-quartic"`
`"in-quintic"`	`"out-quintic"`	`"in-out-quintic"`
`"in-circular"`	`"out-circular"`	`"in-out-circular"`
`"in-exponential"`	`"out-exponential"`	`"in-out-exponential"`