

SUPERSEDED FEATURES

View online: <https://www.construct.net/en/make-games/manuals/construct-3/tips-and-guides/superseded-features>

Construct has been in development for many years. Over time new features are occasionally introduced that supersede older features and essentially make them redundant. Sometimes in this case the older features are phased out and ultimately removed from Construct. For this case, see [Deprecated features](#). However in some cases these features are left in Construct indefinitely for backwards compatibility reasons. For example the newer feature may not have exactly the same capabilities as the older feature and so cannot fully replace it in every possible usage scenario. Alternatively the older feature may have been around for such a long time that removing it would require updating vast amounts of documentation, including printed books, videos, third-party tutorials, lesson plans used in educational institutions, and so on, which is on the whole infeasible and so it's better to leave the older feature in.

Where older superseded features are left in, it's perfectly acceptable for existing projects, guides and so on to keep using them. However new projects should avoid using them in favor of the newer feature that supersedes them. Further, usually the older feature will not be supported to the same extent as the newer feature, such as fixing possible issues or adding new requested features, as we would instead recommend using the feature that supersedes it. In order to help Construct users know which features to prefer and avoid, this guide lists the features we consider superseded.

Pin behavior

The [Pin behavior](#) was widely used for many years, but it is almost completely superseded by the [Hierarchies](#) feature. The '[Add child](#)' [hierarchy action](#) is similar to pinning the child on to the parent, but works more reliably when chains of objects are connected. Hierarchies can also be set up in the [Layout View](#).

Fade behavior

The [Fade behavior](#) was widely used for many years, but is superseded by the [Tween behavior](#). The Fade behavior essentially runs a pre-defined series of opacity tweens, which is now better done in a more general way with the Tween behavior.

Solid behavior tags

A *Tags* property was added to the [Solid behavior](#) prior to the introduction of the instance tags feature. Once instance tags were supported, the Solid behavior's own *Tags* property became redundant. However it is still supported for backwards compatibility reasons only. The default for newly added Solid behaviors is to have *Use instance tags* checked, meaning the Solid behavior will use the instance tags instead of its own tags, which is the recommended usage.

Deprecated features

See also the section on [Deprecated features](#), which covers features that will be completely removed in future (or have already been removed).