

Simplified Concepts in Web Animations Naming



By Sam Thorogood

Evangelises Chrome and the mobile web in the Developer Relations team at Google.

Native support for Web Animations first shipped [in Chrome 36](#), and was updated with playback control [in Chrome 39](#). The `Element.animate()` method can be used to trigger imperative animations directly from JavaScript, and its returned object can be leveraged to control the playback of these animations. These methods are detailed in the current draft of the [Web Animations W3C spec](#).

There's a [shipped polyfill](#) under active development that tracks all Web Animations features implemented natively, and which is supported in all modern browsers. These core methods are ready for use right now, and deserve to be part of your toolbox for building rich experiences that benefit from animations (such as for the [Google I/O 2015 web app](#) [↗](#)).

Constructor and groups changes

The Web Animations spec also describes *groups and sequences*, and constructors for animations and players. These have been available in the [web-animations-next](#) polyfill, which has been designed to showcase features still undergoing discussion and yet to be implemented natively. In response to developer feedback, the team developing Web Animations are renaming these features to be more self-explanatory.

The [EXTE](#) [↗](#) recently met in Sydney, Australia, and [discussed naming](#), as a number of developers raised valid points about some of the naming being confusing. As a result, the following naming changes were agreed on:

- **Animation** becomes **KeyframeEffect**
- **AnimationSequence** becomes **SequenceEffect**
- **AnimationGroup** becomes **GroupEffect**
- **AnimationPlayer** becomes **Animation**

Remember that while animations and their players are available natively in Chrome and as part of the polyfill, they are currently created directly via the `Element.animate()` method.

Existing code that uses the `Element.animate()` method will require no changes.

The new names more accurately represent the behavior provided by each object. `KeyframeEffect`, for instance, describes keyframe-based effects that can target HTML elements. In contrast, the new `Animation` object represents an animation in one of many states (such as playing, paused etc).

SourceCodeEffect

If you're using parts of the draft spec via the web-animations-next polyfill, you'll have to update your code within the deprecation period to reflect these new names. As per the [polyfill changes policy](#), we aim to support an old version for three months and provide console warning statements if your site use deprecated features or names.

If you're keen to try out these features, then watch out for the [v2 release](#) of the polyfill to take advantage of these new names. Finally, be sure to subscribe to the [web-animations-changes](#) group to hear about any other changes.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated July 2, 2018.