Asymmetric animation timing

By [Paul Lewis](#)

Paul is a Design and Perf Advocate

Asymmetric animation timing improves the user experience by allowing you to express personality while at the same time respond quickly to user interactions. It also provides contrast to the feel, which makes the interface more visually appealing.

TL;DR

- Use asymmetric animation timing to add personality and contrast to your work.
- Always favor the user's interaction; use shorter durations when responding to taps or clicks, and reserve longer durations for times when you aren't.

Like most "rules" of animation, you should experiment to find out what works for your application, but when it comes to the user experience, users are notoriously impatient. The rule of thumb is to **always respond to a user interaction quickly**. That said, most of the time the user's action is asymmetric, and therefore the animation can be, too.

For example, when a user taps to display a sidebar navigation, you should display it as quickly as possible, with a duration of around 100ms. When the user dismisses the menu, however, you can afford to animate the view out a little more slowly, say, around the 300ms mark.

By contrast, when you bring on a modal view, this is normally to display an error or some other critical message. In such cases, you will want to bring on the view a little more slowly, again around the 300ms mark, but dismissal, which the user triggers, should happen very quickly.

The general rule of thumb, then, is the following:

- For UI animations triggered by a user's interaction, such as view transitions or showing an element, have a fast intro (short duration), but a slow outro (longer duration).
- For UI animations triggered by your code, such as errors or modal views, have a slower intro (longer duration), but a fast outro (short duration).