

Extracted from:

# Designed for Use

---

## Usable Interfaces for Applications and the Web

This PDF file contains pages extracted from Designed for Use, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

**Note:** This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printer versions; the content is otherwise identical.

Copyright © 2011 The Pragmatic Programmers, LLC.

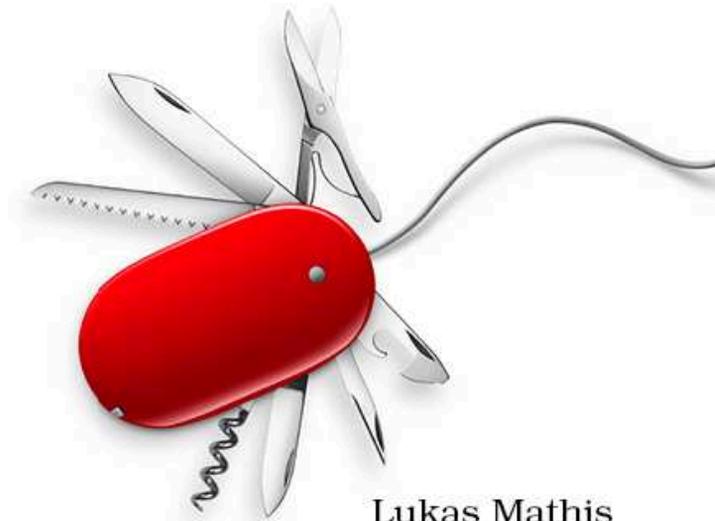
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The  
Pragmatic  
Programmers

# Designed for Use

*Create Usable Interfaces  
for Applications and the Web*



Lukas Mathis  
*edited by Jill Steinberg*



## Chapter 15

# Animations

---

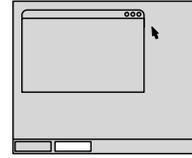
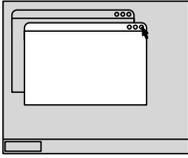
Working with a modern computer is an immensely visual experience. Almost everything we do results in visible, graphical changes on our screens. When we write a letter, images and characters appear. When we look for something on the Web, the content of our browser window changes to represent the web page we're currently visiting. When we read email, messages open and close.

Modern user interfaces make use of both very small visual changes (a single dot next to a message disappearing or reappearing because we read a message or chose to mark it as unread) and very large visual changes (everything on our screen changing because we opened a different website). It's often easy either to miss changes or to be confused by the magnitude of a change.

Animations, when used properly, can help users understand what is happening on their computers. Animations can help explain the causal relationships between two different visual states of a screen, and they can draw attention to changes that might otherwise go unnoticed. Animations can even help people form a correct mental model of how a product works.

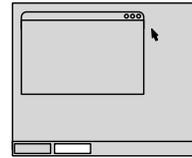
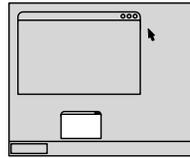
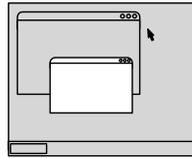
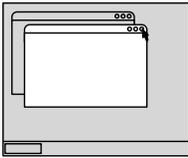
### 15.1 Explaining State Changes

Visual state changes can easily disorient people. If you're running a usability test or just teaching somebody how to use a product, often it doesn't take long until you hear someone ask, "What just happened?"



*I'M JUST CLICKING ON THAT LITTLE BUTTON HERE...  
HUH, WHERE DID THE WINDOW DISAPPEAR TO?*

If you hear somebody say “What did I just do?” or “Where did this go?” or “How do I get back?” it’s possible that you’ve found a problem that could be solved with an animation.



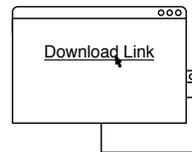
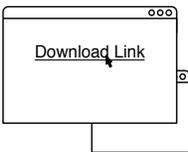
*ADDING A SIMPLE ANIMATION MAKES IT OBVIOUS  
WHERE THE WINDOW WENT.*

Animating the change between the two states makes the state change and the relationship between the two states obvious.

## 15.2 Directing User Attention

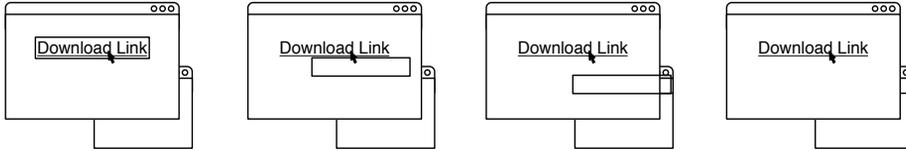
When people get confused by large state changes, they usually react by asking themselves what just happened. When they miss small state changes, their reaction is typically along the lines of “Did I just do something, or not? Did it take?”

Here’s something you’ve probably experienced. You click a link to download a file, and nothing happens. So, you click again. Still nothing. It isn’t until you’ve clicked a few more times that you notice that you’ve downloaded the same file half a dozen times; the computer just didn’t tell you about it.



*WHEN I CLICKED ON THAT DOWNLOAD LINK,  
DID ANYTHING ACTUALLY HAPPEN?*

You can solve problems of this type by animating the state change, with the goal being to exaggerate the change. When somebody initiates a download, it's a good idea to show an exaggerated animation that guides the user's eyes from the link to the download window.



*ADDING A SIMPLE ANIMATION MAKES IT OBVIOUS THAT CLICKING ON THE LINK HAS STARTED A DOWNLOAD.*

Make it impossible for the user to miss what's going on. Google Chrome, for example, shows a big animated blue arrow when the user clicks a download link to indicate that a new download has started and where to look for it.



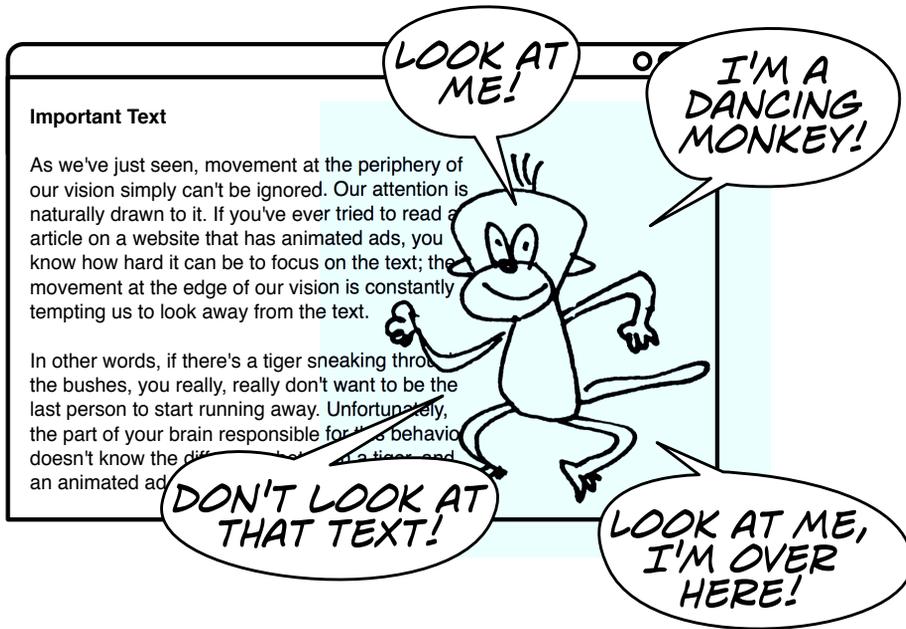
In the book *Information Visualization* [War04], Colin Ware points out that it may not be motion itself that attracts attention but the appearance of new objects. He writes:

When early man was outside a cave, intently chipping a lump of flint into a hand axe, or when early woman was gathering roots out on the grassland, awareness of emerging objects in the periphery of vision would have had clear survival value. Such a movement might have signaled an imminent attack. Of course, the evolutionary advantage goes back much further than this. Monitoring the periphery of vision for moving predators or prey would provide a survival advantage for most animals. Thus, the most effective reminder might be an object that moves into view, disappears, and then reappears every so often.

This behavior is so deeply ingrained in our evolutionary history, it's impossible to turn off.

## 15.3 Avoid Unimportant Animations

As we've just seen, movement at the periphery of our vision simply can't be ignored. Our attention is naturally drawn to it. If there's a tiger sneaking through the bushes, you really, really don't want to be the last person to start running away. Unfortunately, the part of your brain responsible for this behavior doesn't know the difference between a tiger and an animated ad.

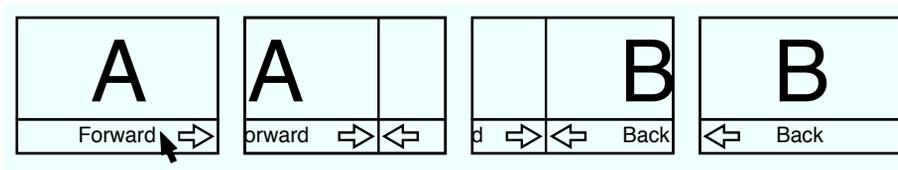


This kind of animation can drive people mad and make it impossible for them to focus on the actual content.

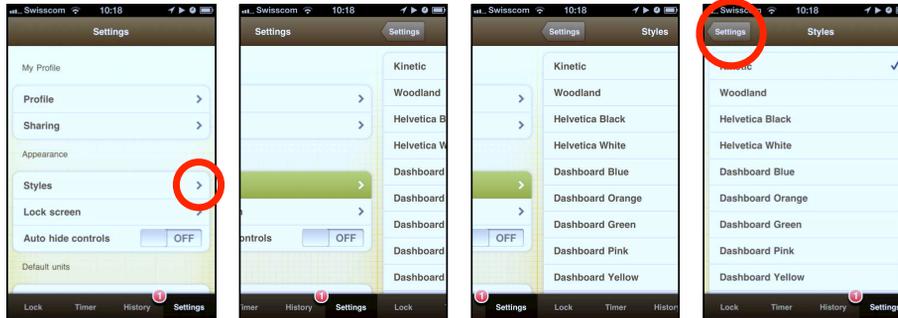
Use animation only if you truly want to interrupt your users and force them to look at something and if the animation is used to convey relevant information. Otherwise, avoid it.

## 15.4 Help Users Form Suitable Mental Models

Animations can be used to influence the user's mental model of your product. The information that animations convey should be consistent with the rest of the user interface. For example, if an animation replaces the screen content by sliding a new screen in from the right, then the back button should have an icon of a left-pointing arrow. Otherwise, the icon is at odds with the animation; the two imply conflicting models.



Here's an example from the iPhone app Kinetic:<sup>1</sup>



The icons are consistent with the animation. Arrows that point to the right move the user to the right, and vice versa.

My Android phone's home button is one example where the animation and the actual behavior are inconsistent. If I'm in an application and hit the home button, the animation implies that I'm moving to the right, even though the home button's icon—a house—doesn't imply any spatial relationship.



Indeed, if I then hit the back button, whose arrow icon implies that it should move me back to the left, nothing happens.

1. It's an app that helps you keep track of your running. You can learn more at <http://wearemotherhip.com/kinetic>.



*HOME BUTTON  
MOVES TO THE  
RIGHT...*



*...BUT LEFT ARROW  
DOESN'T MOVE TO  
THE LEFT*

In newer versions of the OS, Google has replaced the home button's animation. This simple fix makes it less likely that people will think that the back button works for jumping back into an application after leaving it via the home button.

Think about the user's mental model. Use animations wisely to reinforce correct mental models and dissuade incorrect ones.

## 15.5 Learning from Cartoons

Cartoons have a hundred years of development behind them. Cartoon animators have come up with a complex visual language that even small children can easily understand. We can follow cartoons even if they display a large number of objects or characters at the same time, each exhibiting intricate behaviors. In their paper "Animation: From Cartoons to the User Interface,"<sup>2</sup> Bay-Wei Chang and David Ungar point out a number of lessons we can learn from cartoons:

### Solidity

In the real world, objects are solid: they have mass, inertia, weight, and balance. These are attributes that people understand intuitively. By replicating these attributes in your animations, it's easier for users to figure out what's going on.

Let users interact with objects instead of showing outlines or other placeholders during animations. For example, when people drag and drop emails in some applications, these apps show only a single icon below the mouse cursor (instead of the dragged messages). This breaks the impression that emails are solid "things" similar to a paper letter.

2. You can read the paper at [http://research.sun.com/techrep/1995/smlr\\_tr-95-33.pdf](http://research.sun.com/techrep/1995/smlr_tr-95-33.pdf).



*IF YOU DRAG FILES IN WINDOWS 7, YOU CAN SEE THE ACTUAL FILES YOU'RE DRAGGING*



*MAIL ON THE MAC, ON THE OTHER HAND, ONLY SHOWS AN ABSTRACT REPRESENTATION OF DRAGGED MESSAGES*

To make objects appear to be solid, there should be no perceptible lag in your animations, and movements should seem smooth. In his paper “Principles of traditional animation applied to 3D computer animation,” Pixar’s chief creative officer John Lasseter notes:

As the speed of the action increases, so does the distance between positions. When the distance becomes far enough that the object does not overlap from frame to frame, the eye then begins to perceive separate images.

As a rule of thumb, if an object moves more than half its size between frames, stretching the object in the direction of motion or adding motion blur to the animation allows users to see the object as solid, even though it actually jumps from place to place.<sup>3</sup>



*TO HELP PEOPLE PERCEIVE A MOVING OBJECT AS SOLID...*



*...EITHER STRETCH IT IN THE DIRECTION OF THE MOTION...*



*...OR BLUR IT IN THE SAME DIRECTION.*

In the real world, solid objects never just pop into existence out of nowhere. Chang and Ungar recommend that to maintain the illusion of solidity, objects should fly in from offscreen, grow from a point, or fade in (and exit the screen the same way).

3. As Keith Lang, COO and co-designer of Skitch points out, motion blur can also make animations more aesthetically pleasing: <http://www.uiandus.com/blog/2009/7/2/blur-the-new-black.html>.

## Exaggeration

You can take liberty with what's possible in reality to make what's happening on the screen more obvious. Exaggerating the important parts of an animation can sometimes make the animation feel more realistic.

Similar to taking a step back before taking a leap, it can be a good idea to start an animation with a small movement that is contrary to the main movement of the animation. This is a specific form of exaggeration called *anticipatory movement*.

John Lasseter notes “without anticipation many actions are abrupt, stiff and unnatural.”

## Acceleration and Deceleration

Real objects don't start out at a specific speed; they have to accelerate to that speed. So, it makes sense to start animations slowly, speed them up, and then slow them down again toward the end (sometimes called *ease-in* and *ease-out*). This logic also applies to growing and shrinking animations.

Similarly, real-world objects rarely travel in straight lines. When you throw a ball, it flies in an arc. For some kinds of animations, it makes sense to mimic that behavior. For example, when you buy a Mac application in Apple's App Store, it flies into the Dock in an arc.

Finally, real-world objects such as springs sometimes overshoot their target and bounce back. Replicating this kind of behavior can also reinforce the physicality of an object.

After introducing the first iPhone at the Macworld Conference & Expo 2007 keynote and showing the device's smooth, bouncy scrolling, Steve Jobs told the following little anecdote:<sup>4</sup>

I was giving a demo to somebody a little while ago, who had never seen this before, inside Apple. I finished the demo, and he told me this. He said, you had me at scrolling.

Paying attention to these kinds of details turns a merely acceptable, workable device into a delightful experience.

---

4. You can watch it here: <http://www.apple.com/quicktime/qtv/mwsf07/>.

## User Interfaces Are Not Cartoons

It's important to remember that the goal is not to turn the user interface into a cartoon but to make it easier for people to understand how it works by using the same kinds of tools that cartoons use.

Don't use these tools, and animations in general, just because they look cute or flashy. Android, for example, provides animated home-screen wallpapers. Although it would be possible to use this feature in meaningful ways, in practice most of these wallpapers just serve as a distraction and do not show any useful information or provide any useful functionality.

It's sometimes OK to use animations to make a product more fun to use, even though the animations provide no tangible benefit. However, keep in mind that animations are distracting. Attempting to make a product more fun at the expense of its usability typically doesn't work (see Section 26.4, *Fun vs. Usability*, on page 233 for more thoughts on that).

## Takeaway Points

- Use animations to draw attention to small or otherwise unnoticeable changes.
- Use animations to make large state changes understandable.
- Animations, especially at the edge of vision, are attention magnets. Don't abuse them; it'll drive people crazy.
- You can use animations to reinforce the user's mental model of your product.
- Cartoons have a visual language for animation that we can use: objects should be solid and accelerate and decelerate realistically, and movement should sometimes be exaggerated.

## Further Reading

*Information Visualization* [War04] by Colin Ware has some information on animations. Bay-Wei Chang and David Ungar's paper "Animation: From Cartoons to the User Interface"<sup>5</sup> is also a great resource.

---

5. It can be found at [http://research.sun.com/techrep/1995/smlr\\_tr-95-33.pdf](http://research.sun.com/techrep/1995/smlr_tr-95-33.pdf).

Markus Weber talks about using animation in user interface design in his blog.<sup>6</sup> Keith Lang has written about using animation in user interfaces.<sup>7</sup> Max Steenbergen also tackles this topic in his blog.<sup>8</sup>

And finally, John Lasseter's paper "Principles of traditional animation applied to 3D computer animation" contains invaluable information on how to use animation.<sup>9</sup>

---

6. At <http://www.centigrade.de/en/blog/article/animations-in-user-interface-design-essential-nutrient-instead-of-eye-candy/>.

7. At <http://www.uiandus.com/blog/2009/2/1/interfaces-and-animation.html>. He also points out that people may be blind to certain kinds of changes in his essay "The Art of Expectations" at <http://www.uiandus.com/blog/2008/8/25/the-art-of-expectations.html>.

8. At <http://facevalue.virb.com/blog/text/12641234>.

9. You can find it at <http://portal.acm.org/citation.cfm?id=37407>.

# The Pragmatic Bookshelf

---

The Pragmatic Bookshelf features books written by developers for developers. The titles continue the well-known Pragmatic Programmer style and continue to garner awards and rave reviews. As development gets more and more difficult, the Pragmatic Programmers will be there with more titles and products to help you stay on top of your game.

## Visit Us Online

---

### Homepage for Designed for Use

<http://pragprog.com/titles/lmuse>

Source code from this book, errata, and other resources. Come give us feedback, too!

### Register for Updates

<http://pragprog.com/updates>

Be notified when updates and new books become available.

### Join the Community

<http://pragprog.com/community>

Read our weblogs, join our online discussions, participate in our mailing list, interact with our wiki, and benefit from the experience of other Pragmatic Programmers.

### New and Noteworthy

<http://pragprog.com/news>

Check out the latest pragmatic developments, new titles and other offerings.

## Buy the Book

---

If you liked this eBook, perhaps you'd like to have a paper copy of the book. It's available for purchase at our store: [pragprog.com/titles/lmuse](http://pragprog.com/titles/lmuse).

## Contact Us

---

Online Orders:	<a href="http://www.pragprog.com/catalog">www.pragprog.com/catalog</a>
Customer Service:	<a href="mailto:support@pragprog.com">support@pragprog.com</a>
Non-English Versions:	<a href="mailto:translations@pragprog.com">translations@pragprog.com</a>
Pragmatic Teaching:	<a href="mailto:academic@pragprog.com">academic@pragprog.com</a>
Author Proposals:	<a href="mailto:proposals@pragprog.com">proposals@pragprog.com</a>
Contact us:	1-800-699-PROG (+1 919 847 3884)