Progress Indicators

Cocoa > User Experience

2002-11-12

Apple Inc.
© 2002 Apple Computer, Inc.
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, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, Cocoa, and Objective-C are trademarks of Apple Inc., registered in the United States and other countries.

Java and all Java-based trademarks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

Introduction to Progress Indicators 5

Organization of This Document 5

About Progress Indicators 7

Using Determinate Progress Indicators 9

Using Indeterminate Progress Indicators 11

Document Revision History 13

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

Introduction to Progress Indicators

A progress indicator shows that a lengthy task is under way. It can either be determinate and display how much of the task is done, or it can be indeterminate and display only that the application is busy.

Organization of This Document

This topic describes how to use a progress indicator. "About Progress Indicators" (page 7) gives basic information on progress indicators. "Using Indeterminate Progress Indicators" (page 11) and "Using Determinate Progress Indicators" (page 9) describe how to use the different types of indicators.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

About Progress Indicators

A progress indicator shows that a lengthy task is under way. Some progress indicators do nothing more than spin to show that the application is busy, while others show the percentage of the task that has been completed. NSProgressIndicator provides both types of display:

- An indeterminate progress indicator (the "barber pole") that spins until the task is complete, like this:

- A determinate progress indicator that draws a three-dimensional progress bar from left to right in the view as the task progresses, like this:

NSProgressIndicator is a subclass of NSView. To display a progress indicator, your application creates a window and adds the progress indicator as a subview of the window's content view or any subview. You can create a progress indicator programmatically either with its Java constructor or initialize it with the Objective-C `initWithFrame:` method. However, you normally use Interface Builder to create and initialize a progress indicator and to install it in an application view.

You can display progress indicators of different sizes by varying the frame size. However, the default size is designed to provide the best results.

By default, a progress indicator is drawn with a bezeled frame, but you can use the `setBezeled:` method to modify the bezeled-frame setting.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

Using Determinate Progress Indicators

By default, a progress indicator is indeterminate. Your can specify a determinate progress indicator when you set up the view with Interface Builder, or you can use code like the following example to change the default value programmatically:

```
[myProgressIndicatorView setIndeterminate:FALSE];
```

For a determinate progress indicator, invoke the `incrementBy:` method to advance the progress bar. By default, a determinate progress indicator goes from 0.0 to 100.0. You can increment by any amount, but if you vary the increment too widely, progress may appear uneven or jerky. You typically choose an increment value that evenly divides 100.0. For example, you might invoke `incrementBy:` 50 times, incrementing by 2.0 each time, to draw the complete progress bar. To set the value directly, use `setDoubleValue:`. To modify the default range of 0.0 to 100.0, you can invoke `setMinValue:` to modify the minimum value and `setMaxValue:` to modify the maximum value.

After each invocation of `incrementBy:`, you can invoke the `displayIfNeeded` method to ensure immediate redrawing.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

Using Indeterminate Progress Indicators

For an indeterminate progress indicator, invoke `startAnimation:` to start the animation (the spinning of the barber pole) and `stopAnimation:` when the task is complete. By default, the delay between animation steps is one twelfth of a second (5/60). You can change the animation delay by invoking `setAnimationDelay:`. Setting the delay to a double value larger than the default value will slow the animation, while setting the delay to a smaller value will speed it up.

Instead of invoking `startAnimation:` and `stopAnimation:`, you can control an indeterminate progress indicator directly by sending the `animate:` message. Each time you invoke `animate:`, the animation advances by one step. You can speed up or slow down the animation by varying how often you invoke `animate:`. Like other views, a progress indicator redisplays itself on each pass through the event loop, if needed. To ensure immediate redrawing, however, you can invoke the `displayIfNeeded` method (inherited from NSView) each time you invoke `animate:`.

If your application is threaded, you may want the progress indicator's animation to happen in a separate thread by calling `setUsesThreadedAnimation:`. If your application doesn't already use threads, creating a thread just for the progress indicator can actually slow down your application.

By default, a progress indicator is indeterminate.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

Document Revision History

This table describes the changes to *Progress Indicators*.

Date	Notes
2002-11-12	Revision history was added to existing topic. It will be used to record changes to the content of the topic.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.

2002-11-12 | © 2002 Apple Computer, Inc. All Rights Reserved.