
What's New in Xcode

Tools & Languages: IDEs



2010-07-02



Apple Inc.
© 2010 Apple 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

.Mac is a registered service mark of Apple Inc.

Apple, the Apple logo, Cocoa, iPhone, Leopard, Mac, Mac OS, Safari, Snow Leopard, Tiger, and Xcode are trademarks of Apple Inc., registered in the United States and other countries.

IOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license.

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 5

Organization of This Document 5

Xcode 3.2 Feature Overview 7

General 7
Project Management 7
Workspace 8
Documentation Access 9
Debugging 9
Compilers 9
SCM 10
Platforms and Architectures 10

Xcode 3.1 Feature Overview 11

iPhone SDK 11
Project Package Format 11
Toolbar Items 11
Text Editor 12
Property List Editor 12
Framework and Library Use 12
Build System 12
New Project and New Target Assistants 13
Open Quickly 13
FileMerge Application 13
SCM Workflow 14
Compilers 14
Xcode Persistent Cache 14

Document Revision History 15

Index 17

Introduction

This document provides an overview of the new and improved features in Xcode releases.

This document is intended for people interested in the changes Xcode releases bring about. You can use this document to learn about features that enhance your workflow, improve your efficiency, and increase your productivity.

Organization of This Document

This document includes the following articles:

- ["Xcode 3.2 Feature Overview"](#) (page 7)
- ["Xcode 3.1 Feature Overview"](#) (page 11)

Xcode 3.2 Feature Overview

This article describes the new features and improved functionality in Xcode 3.2.

General

Xcode is included in the Mac OS X v10.6 Snow Leopard DVD. You can also download it from <http://developer.apple.com/mac/>.

Software requirements: Xcode 3.2 runs on Mac OS X 10.6 Snow Leopard and later.

Project Management

Xcode's project-management capabilities have been enhanced to provide you with a better user experience and improved flexibility.

- **Static analysis.** Static analysis is a method for finding potential problems in source code without running the corresponding executable. Using static analysis you can find bugs in your code before your product goes to testing or it released to customers. It also helps you ensure your code follows Cocoa programming guidelines, such as memory-management conventions. See "Analyzing Code" for details.
- **Redesigned Build Results window.** The Build Results window now saves the results of multiple builds so that you can track your progress as you address build problems. You can also display build messages by step or by issue, letting you easily address similar build problems. The Error & Warnings group in the Groups & Files list has been removed. See "Viewing Build Results" in *Xcode Project Management Guide* for details.
- **Searching in a file.** Searching within a file has been improved with a single-window experience. Instead of a search window in which to specify your search criteria, you define your search in a search bar on the text editor or property-list editor window you're working on. See "Searching in a File" in *Xcode Project Management Guide* for details.
- **Improved build and run/debug workflow.** Straightforward command for building, building and running, and build and debugging have been added for a more predictable yet flexible workflow. See "Building a Target" in *Xcode Project Management Guide* for details.
- **Redesigned Welcome to Xcode window.** The Welcome to Xcode window now lets you create a project, open a recently-opened project, or jump to Xcode's getting-started resources or the Apple Dev Centers.
- **Organization Name setting.** You can now set your organization in the Project Info window. This allows you to define your company name in one place in a project for all build configurations. For more information, see "General Project Attributes" in *Xcode Project Management Guide*.

- **Renaming a project.** Xcode can now rename a project. Choosing Project > Rename lets you specify a new name for a project. Xcode shows you the places in which the rename operation takes place, which includes the project package, targets, and info-plist files. See “Snapshots” for details.
- **Improved New Project dialog.** The New Project dialog contains streamlined lists of project templates and provides options for specifying the features to include in a new project, such as Core Data–based storage.
- **Multiple project roots.** Instead of a single project root (the directory containing the project package), a project can now contain multiple roots, each associated with its own SCM system. Multiple roots allow you to work on a large project made up of several subprojects that may use different SCM repositories. They also improve the performance of SCM and snapshot operations on projects with subprojects. For more information, see “General Project Attributes” in *Xcode Project Management Guide*.
- **Automatic offline Subversion operations.** Xcode now supports offline SCM operations on Subversion repositories. Operations you perform while disconnected from the repository are synchronized when the connection is re-established.
- **Added support for SSE 4.1 and 4.2 extensions.** To turn on SSE 4.1 and SSE 4.2 extension support in your code, use the Enable SSE4.1 Extensions and Enable SSE4.2 Extensions build settings, respectively. See “Compiler Build Settings” in *Xcode Build Setting Reference* for details.
- **Added build settings for the xib-file compiler.** Build settings to specify how to process xib files, including whether to flatten them, have been added to the Xcode build system. To learn about the new build settings, search for “IBC_” in the Build pane of the Project Info window.
- **Xcode 3.2 project format.** Xcode 3.2 introduces the Xcode 3.2 project format. Features introduced in Xcode 3.2 may not be available in projects that use project formats earlier than the Xcode 3.2 project format. For more information about project formats, see “Choosing the Project Format” in *Xcode Project Management Guide*.

Workspace

The Xcode workspace has been improved to enhance your development experience.

- **Source-code navigation.** The Go Back and Go Forward buttons of the text editor toolbar—in addition to previously viewed files—take you to places within the current file at which you’ve made changes. To jump between files, Option-click the Go Back or Go Forward buttons. In-file navigation works only with C-based source files. See “Navigating Code” in *Xcode Workspace Guide* for details.
- **Single-file search.** Single-file search is now similar to searching a webpage in Safari. When you choose Edit > Find > Find (Command-F), the search bar appears. This search bar allows you to set the type of search to perform (search or search & replace), enter the search term (in the search field), and specify search options (through the pop-up menu in the search field). For details, see “Searching in a File” in *Xcode Project Management Guide*.
- **Code-completion improvements.** Code completions offers more useful and accurate completions; for example, inaccessible symbols are not included in completions, and block signatures are included in completions with block parameters. Placeholders can be converted to plain text. See “Completing Code” in *Xcode Workspace Guide* for more information.
- **Code-navigation improvements.** The Go Back and Go Forward commands, in addition to moving between opened files, let you navigate through interesting locations within a source file (mostly, locations in which you’ve made changes). See “Navigating Code” in *Xcode Workspace Guide* for details.

- **General preferences changes.** Xcode can reopen the projects that were open when it last quit. See “General Preferences” in *Xcode Workspace Guide* for details.
- **Viewing HTML files in the text editor.** You cannot view HTML files as web pages in the text editor, but you can edit them as HTML source. See “Specifying How Files Are Opened” in *Xcode Workspace Guide* for more information.

Documentation Access

Xcode provides you with fast access to documentation through an improved and streamlined Documentation window and Quick Help (a revamped implementation of the Research Assistant).

- **Xcode Quick Start.** Xcode Quick Start is a resource especially designed for people new to Xcode. It provides access to basic information about Xcode development, videos that show how to perform essential tasks, and other Apple Developer Connection resources. Xcode Quick Start appears only in the Documentation window. See “Using the Documentation Window” in *Xcode Workspace Guide* for more information.
- **Quick Help.** Quick Help is a lightweight window that provides essential information about the symbol selected in the text editor without taking away the focus from the editor. From Quick Help you can jump to the header file that declares the symbol or to the symbol’s API reference. For details, see “Using the Research Assistant” in *Xcode Workspace Guide*.
- **Xcode Help search.** Using the search field in the Help menu, you can perform quick searches through the Xcode guides. Xcode presents the search results in the Documentation window.
- **Improved documentation search.** Searching documentation is now simpler and faster. As you type your search term, Xcode performs three kinds of search simultaneously: API, Title, and Full Text. For details, see “Searching Documentation” in *Xcode Workspace Guide*.
- **Sample-code download improvements.** When downloading a sample-code project, Xcode unarchives and opens the project. See “Using the Documentation Window” in *Xcode Workspace Guide* for more information.

Debugging

Xcode debugging improvements support debugging of programs running on iOS-based devices.

- **Attach on launch/push notification.** The Debugging pane of the Executable Info window lets you indicate whether you want to attach to a process when it launches. This feature is especially useful when debugging iOS applications that use push notifications.

Compilers

These are compilers supported in Xcode:

- GCC 4.0

- GCC 4.2 (default)
- LLVM-GCC 4.2
- Clang 1.0

SCM

Xcode SCM capabilities have been enhanced with automatic synchronization of offline Subversion operations. Operations you perform while disconnected from the repository are synchronized when the connection is re-established.

Platforms and Architectures

Xcode can build products that run in the following platforms and architectures:

- **Mac OS X v10.4 Tiger and v10.5 Leopard (10.5.6 or later).** Binaries targeted at Mac OS X v10.4 and v10.5 can run on these architectures: PPC, PPC64, i386, and x86_64.
- **Mac OS X v10.6 Snow Leopard.** Binaries targeted at Mac OS X v10.6 can run on these architectures: PPC, i386, and x86_64.

Xcode 3.1 Feature Overview

Xcode 3.1 is a major revision to the Xcode IDE. The main goal of this release is to support iOS application development. However, this release also includes new features to enhance your workflow.

This article describes the new features and improved functionality in Xcode 3.1.

iPhone SDK

Using Xcode 3.1 and the iPhone SDK you can develop applications for iOS. The iPhone SDK includes iOS frameworks and developer tools with which you can build iOS applications. It also includes iPhone Simulator, an environment that allows you to run iOS applications on your Mac to perform initial testing. You can also run your application on actual devices, after becoming a member of the iPhone Dev Center.

For more information about iOS development, see *iOS Development Guide*.

Project Package Format

Xcode 3.1 introduces the 3.1 project-package format.

Xcode 3.1 reads and builds project packages v2.1–3.0 and automatically upgrades project packages v1.5–2.0 to v3.1.

In general, project packages v3.1 can be opened and built in Xcode 3.0 and 2.5. Xcode tells you when a project package uses a feature it doesn't support.

Toolbar Items

Xcode 3.1 introduces two toolbar items:

- **Active SDK.** Lets you specify the SDK to use to build the active target. You can choose between iPhone OS SDKs and Mac OS X SDKs.
- **Overview.** Groups the “active” project settings into one, easily accessible control. With this toolbar item you can set a project's active target, SDK, build configuration, architecture and executable.

See "Setting Build Factors" in *Xcode Project Management Guide* for more information.

Text Editor

In Xcode 3.1 the text editor gains two new features:

- **Edit All in Scope.** The Edit All in Scope command allows you to edit several symbolname occurrences at the same time in a more convenient way than using Find & Replace. For example, you can change the name of a method argument, and Xcode replicates the change within the method's body at the same time. See "Editing Symbol Names" in *Xcode Workspace Guide* for details.
- **Symbol assist menu.** The symbol assist menu is an automatic shortcut menu that appears when you select a symbolname in a source code file. This menu allows you to perform one of several symbol-related commands on the selected symbols, including Edit All in Scope and Jump to Definition.

Note: The symbol assist menu appears only when Edit All in Scope is active. See "Code Sense Preferences" in *Xcode Workspace Guide* for details.

Property List Editor

The user interface of the Property List Editor application has been revamped to make it easier to edit property list files. Among the improvements are:

- Reorder/move elements
- Cut/Copy/Paste element
- Property list types

You can create files based on particular property-list schemas (such as the iPhone Settings schema used in iOS applications)

Xcode now provides property list editor that uses the same user interface the Property List Editor application uses.

Framework and Library Use

Xcode 3.1 provides a new, straightforward way for adding frameworks and libraries to a target. See "Managing Files and Folders in a Project" in *Xcode Project Management Guide* for details.

Build System

The Xcode 3.1 build system received several improvements:

- Conditional build settings

In Xcode 3.0 you can define build setting specifications for particular architectures and build variants. Xcode 3.1 adds another condition: The SDK to use to build the active target.

See "Conditional Build Settings" in *Xcode Build System Guide* for details.

- Support for weak/required frameworks (new)

Similar to symbol weak-linking, framework weak-linking allows you to build a product that is able to use a framework that may not be present at runtime. See "Managing Libraries and Frameworks" in *Xcode Project Management Guide* for details.

- Compiler Version build setting group (new)

The C/C++ Compiler Version build setting is more visible in the settings editor; it's now in the Compiler Version build setting group.

- New `xcodebuild` options

The `xcodebuild` tool supports these new options:

- `-sdk <sdk_name>` Specifies the SDK to use.
- `-fine <tool_name>` Provides the filepath to `tool_name`.

For details, see the man page for `xcodebuild`.

- Strings files output encoding option (new)

The new `STRINGS_FILE_OUTPUT_ENCODING` build setting lets you specify the output encoding of strings files.

New Project and New Target Assistants

Xcode 3.1 updates the New Project and New Target assistants with new project and target types. For details, see "Creating Projects" and "Creating Targets" in *Xcode Build System Guide*.

Open Quickly

Xcode 3.1 simplifies the Open Quickly dialog and makes it more effective. See "Opening Files by Filename or Symbolname" in *Xcode Workspace Guide* for details.

FileMerge Application

Xcode 3.1 improves the FileMerge application by:

- Performing faster comparisons
- Handling files with different line encodings properly
- Ignoring whitespace on lines

SCM Workflow

In Xcode 3.1 you can use drag-and-drop to import and check-out directories. And, when you check out a project directory, Xcode configures SCM in the project packages the directory contains so that you can perform SCM operations without having to configure SCM first (as in previous Xcode releases). For details, see "Browsing and Modifying Repositories" in *Xcode Source Management Guide*.

Compilers

Xcode 3.1 adds support for GCC 4.2 and LLVM-GCC 4.2 (see *LLVM-GCC Release Notes*).

Xcode Persistent Cache

To improve the security of Xcode caches, Xcode 3.1 places them in a more secure location. If you don't use earlier versions of Xcode, you should delete `/Library/Caches/com.apple.Xcode` and `/Library/Caches/Xcode`.

In Xcode 3.1 you can also delete Xcode caches using the **Empty Caches** command. For more information, see "Xcode Persistent Cache" in *Xcode Project Management Guide*.

Document Revision History

This table describes the changes to *What's New in Xcode*.

Date	Notes
2010-07-02	Made minor corrections.
2009-10-19	Added more information about Xcode 3.2.
2009-09-09	Added information about Xcode 3.2.
2009-08-28	Added information about Xcode 3.2.
2009-03-05	Added index.
2009-01-06	Made minor corrections.
2008-05-22	Updated for Xcode 3.1.
2006-08-01	New document that provides an overview of new and improved features in the Xcode application.

Index

E

edit all in scope [12](#)

P

property list editor [12](#)

S

symbol assist menu [12](#)