
What's New in WebObjects 5.2.2

(Legacy)

[Internet & Web](#) > [WebObjects](#)



2003-10-16



Apple Inc.
© 2003 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, Carbon, Cocoa, Mac, Mac OS, WebObjects, and Xcode 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

Chapter 1 **What's New in WebObjects 5.2.2** 5

Support for Mac OS X v 10.3 and Xcode 5

Support for Java 1.4.1 6

Support for JBoss Application Server 6

Resolved Problems 6

Summary of API Changes 6

What's New in WebObjects 5.2.2

Important: The information in this document is obsolete and should not be used for new development.

This document describes the major new features of WebObjects 5.2.2 and explains the most significant changes made to the product.

This release includes support for Mac OS X v 10.3 and Xcode, the new development tools for Mac OS X. This release also fully supports Java 1.4.1 and deployment of WebObjects applications using the JBoss application server. Some problems were fixed in this release but mostly the API was changed in support of servlet/J2EE containers. You can now optionally build your WebObjects applications and frameworks as true JAR files. In support of this, all String path-based API was deprecated and replaced with URL-based API.

Refer to these sections for more details:

- [“Support for Mac OS X v 10.3 and Xcode”](#) (page 5)
- [“Support for Java 1.4.1”](#) (page 6)
- [“Support for JBoss Application Server”](#) (page 6)
- [“Resolved Problems”](#) (page 6)
- [“Summary of API Changes”](#) (page 6)

Support for Mac OS X v 10.3 and Xcode

WebObjects applications can now be developed using Xcode on Mac OS X v 10.3. Xcode is Apple's new integrated development environment (IDE) for Mac OS X. It is designed to fully support all of the major platform initiatives of Mac OS X, such as the Carbon and Cocoa frameworks, Java, and new application packaging mechanisms. For more information on Xcode see [A Quick Tour of Xcode](#).

Important: WebObjects 5.2.2 requires that you have Mac OS X v 10.3, Java 1.4.1, and Xcode Developer Tools installed. WebObjects 5.2.2 no longer supports developing WebObjects applications using the Project Builder and JDK 1.3.1. Currently, some Xcode features are not available to WebObjects and Java developers.

Support for Java 1.4.1

WebObjects 5.2.2 now fully supports Java 1.4.1 (earlier versions of Java are not supported). Both the Java 1.4.1 Update 1 and the Mac OS X v 10.3 release for Java 1.4.1 provide bug fixes and enhancements, especially for graphics functionality. These updates and changes are all discussed in detail in the release notes for Java 1.4.1, the Java 1.4.1 Update 1, and for Mac OS X v 10.3 at <http://developer.apple.com/releasesnotes/Java/index.html>.

Support for JBoss Application Server

WebObjects applications can now be hosted by any servlet/J2EE container including the JBoss application server available on Mac OS X v 10.3. JBoss is configured to use Tomcat (using the AJP connector) as its web server and servlet container. This differs from the standard configuration, which uses Jetty. In addition, HTTP and HTTPS (through port 8443) are enabled by default. For more information on JBoss and Mac OS X, see Mac OS X Java Application Server Administration and other related documents at www.apple.com/server/documentation.

Important: JBoss is installed by default on Mac OS X v 10.3 Server but is an optional package on the Mac OS X v 10.3 Xcode Developer Tools CD.

Resolved Problems

Several fixes were made to the Enterprise Object Frameworks runtime and miscellaneous classes such as WOFileUpload. See the WebObjects 5.2.2 release notes for details on the problems fixed in this release, and remaining problems that will be fixed in future releases.

Summary of API Changes

There were many miscellaneous changes to the WebObjects API across all frameworks. In general, the API changes were made in support of hosting WebObjects applications on any servlet/J2EE container including the JBoss application server. For example, all String path-based API was replaced with URL-based API.

Now when you create a servlet/J2EE container application you can choose the True WAR option. To build a True WAR servlet, all of your frameworks must be in the NSJarBundle format, and located in the `/Library/WebObjects/lib/` directory. When you choose the True WAR option, all your application and

framework code is bundled together into a single file—the WAR archive. Deploying a WAR archive does not require that the WebObjects runtime be installed on the deployment host. WAR archives are intended for use in Servlet Containers such as JBoss.

Typically, WebObjects applications and frameworks exist in a directory format. Applications can now optionally be built as correct WAR files, instead of stubbed WAR or SSDD, and frameworks can be built as JARs implemented by the `NSJarBundle` class. To build an `NSJarBundle` from a framework, set the `BUNDLE_AS_JAR` build property to `YES`.

However, to use this option, you have to avoid using file and directory paths in your code. In general, it is recommended that WebObjects developers use URLs instead of String paths. Therefore, in WebObjects 5.2.2, all API that takes a String path as an argument has been deprecated, and replaced with API that takes a URL as an argument. Although all the WebObjects frameworks have been modified to use URLs, third party frameworks might still use file paths and therefore, may not work in the `NSJarBundle` format.

Also, in support of servlet management tools, `getProperty()` was added to `NSProperties`. Use this method instead of `System.getProperty()` to get system properties. This new method will nicely fall back to JNDI lookups if there are no relevant properties, which makes the properties easy to modify or inspect from servlet management tools.

See </Developer/Documentation/WebObjects/Reference/changes.html> for details on the API changes made in WebObjects 5.2.2.

