ConflictManager® Reference

Legal Notice

The software described in this document is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

ConflictManager® (Wise Package Studio 7.0 SP3)

Copyright © 2001-2008 Symantec Corporation. All Rights Reserved.

Symantec, the Symantec Logo, Altiris, and any Altiris and Symantec trademarks used in the product are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Symantec Corporation and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, "Rights in Commercial Computer Software or Commercial Computer Software Documentation", as applicable, and any successor regulations. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

ConflictManager® is protected by U.S. Patent No. 7,028,019.

Symantec Corporation
20330 Stevens Creek Blvd.
Cupertino, CA 95014
http://www.symantec.com

Altiris, Inc.
47911 Halyard Dr.
Plymouth, MI 48170
http://www.altiris.com

Contents

Preface

This chapter includes the following topics:

- *Product Documentation* on page 5
- *Technical Support Resources* on page 6

Product Documentation

This documentation assumes that you are proficient in the use of the Windows operating system. If you need help using the operating system, consult its user documentation.

Use the following sources of information to learn this product.

Online Help

The online help contains detailed technical information and step-by-step instructions for performing common tasks.

Access help in the following ways:

- To display context-sensitive help for the active window or dialog box, press F1.
- To select a help topic from a table of contents, index, or search, select Help menu > Help Topics.

Reference Manual

All the material in the online help is also available in a .PDF-format reference manual, which you can access by selecting Help menu > Reference Manual.

Getting Started Guide

The *Getting Started Guide* contains system requirements, installation instructions, and a tutorial. You can access a .PDF version of the *Getting Started Guide* from the Windows Start menu.

The installation and repository management sections of the *Getting Started Guide* are also available as online help. In the Wise Repository Manager, select Help menu > Help Topics, or click the Help button on any of the Wise Package Studio installation dialog boxes.

Release Notes

The product release notes cover new features, enhancements, bug fixes, and known issues for the current version of this product. Access the release notes in the following ways:

- Browse the product CD.
- Select Release Notes from the Altiris program group on the Windows Start menu.

Technical Support Resources

If you need help beyond what the product documentation provides, visit the Altiris Service Center, which is your complete online resource for Altiris product support. Access the Altiris Service Center from the Support section of the Altiris Web site.

Use the Altiris Service Center to access the following Altiris support tools and services.

Knowledgebase	Provides a central repository for technical information at Altiris. Articles are reviewed and refined by Altiris personnel and provide information about past problems and their resolutions.
Support forums	Lets Altiris users collaborate and share information. The support forums are monitored by experienced customers, Altiris partners and Altiris personnel.
License Management Portal	Manages and provides access to Altiris product licenses.
Altiris Support Helpdesk	Lets Altiris premium and enterprise support customers use a Web-based tool to log new support incidents, update existing incidents and communicate with Altiris support personnel.
User groups	Provide a place for Altiris users to discuss IT management projects, learn best practices, discover the latest product features, and network with other users.

Before you contact technical support, obtain the following information:

- Serial number and product version, which you can find by selecting Help menu > About.

- Operating system version and service pack version if applicable.

- A description of what you do before the problem occurs.

- The text of any error messages that appear.

- Your name, company name, and how to contact you.

- Contract number or payment information, if applicable.

Chapter 1
Introduction to ConflictManager

This chapter includes the following topics:

- About ConflictManager on page 7
- Starting ConflictManager on page 8
- About the ConflictManager Window on page 9
- About Conflicts on page 12

About ConflictManager

ConflictManager is a tool in the Professional Edition of Wise Package Studio.

Use ConflictManager to detect and resolve conflicts between packages in the Software Manager database. After conflicts are resolved, you export and recompile resolved installations, which are then ready for deployment.

Conflicts between software packages are a time-consuming and costly issue in the Windows workstation environment. Software packages that run independently without incident often fail when installed together due to file and other conflicts. Conflicts can occur for many reasons, but they most commonly occur when two installed packages use a DLL with the same name. Example: Suppose Package 1 installs version 1.0.0.4 of a file named report.dll. Package 2 installs a newer version of the same file, version 2.0.0.1. When the two packages are installed on the same computer, Package 1 might conflict with the newer version of report.dll that is installed by Package 2. This conflict could cause a General Protection Fault, a hung application, or some other problem.

ConflictManager helps you avoid software conflicts. It detects conflicts that would be virtually impossible to find efficiently using any other method and helps you achieve conflict-free computers throughout your organization.

Major features of ConflictManager include:

- A means to detect and view conflicts between packages.
- Conflict resolution rules that let you resolve conflicts automatically, saving time, reducing errors, and providing consistency in conflict resolution. You can create and edit the resolution rules.
- A Resolve wizard that steps you through the process of resolving file conflicts.
- Reports that help you review and analyze conflicts detected by ConflictManager.
- The ability to export and recompile resolved installations individually or in a batch.

Note
(Enterprise Management Server only.) ConflictManager menu commands that you do not have permission to use are unavailable. Functional security for ConflictManager is set in the Wise Package Studio Security Setup.

See *Setting Software Manager and ConflictManager Security* in the Wise Package Studio Help.

Starting ConflictManager

To start ConflictManager

1. In Wise Package Studio, do one of the following:

    - On the Projects tab, click the Run link to the right of the task or tool associated with ConflictManager.

    - On the Tools tab, double-click ConflictManager.

2. If the database contains packages but conflicts have not been detected yet, the Detect Conflicts dialog box appears. If you previously were working in multiple databases, a message appears listing the databases requiring conflict detection.

    - To detect conflicts, select applications and click OK.
      See Detecting Conflicts on page 28.

    - To start ConflictManager without detecting conflicts, click Cancel.

Which Database is Opened?

The first time you start ConflictManager, it opens the Software Manager database that you connected to during installation. Thereafter, ConflictManager opens the database most recently used by either Software Manager or ConflictManager. With Enterprise Management Server, multiple databases might open.

When you start ConflictManager from a task that specifies the database in its command-line options, the specified database is opened.

If ConflictManager cannot find the default Software Manager database, the Database Setup wizard starts. Do the following:

- If the Applications in Use dialog box appears, close the applications listed, close ConflictManager, and click Next.

- On the Share Point Directory page, specify the share point you are using and click Next. Then set up the database as described in *Setting Up Database Connections and Authentication* in the *Getting Started Guide*.

Opening and Closing Software Manager Databases

➢ *Enterprise Management Server only.*

You can open multiple Software Manager databases simultaneously. This lets you browse multiple databases in Software Manager and ConflictManager, and work in a second database without closing the first one.

One database must always be open in ConflictManager.

To open an additional database

1. If the database does not exist, create a new database and then create an ODBC DSN entry. See *Creating Software Manager Databases* in the *Getting Started Guide*.

2. Select File menu > Open Database.

    The Select Data Source dialog box appears. This is a standard Windows ODBC connection wizard, which lets you connect to a database through an ODBC data source.

3. Connect to the database. If you need help, ask your database administrator.

   The database's contents are listed in the Applications/Packages pane.

4. To make a database active, click anywhere in its tree.

   All ConflictManager activities are performed within the active database.

To close a database

1. Select File menu > Close Database.

   The Close Database dialog box appears.

2. Mark the check box next to the database or databases to close and click OK.

See *About the Software Manager Database* and *Multiple Software Manager Databases* in the Software Manager Help.

About the ConflictManager Window

When you start ConflictManager, you see the main ConflictManager window. Some of the panes might be empty if no packages are in the Software Manager database.

Panes of the ConflictManager window
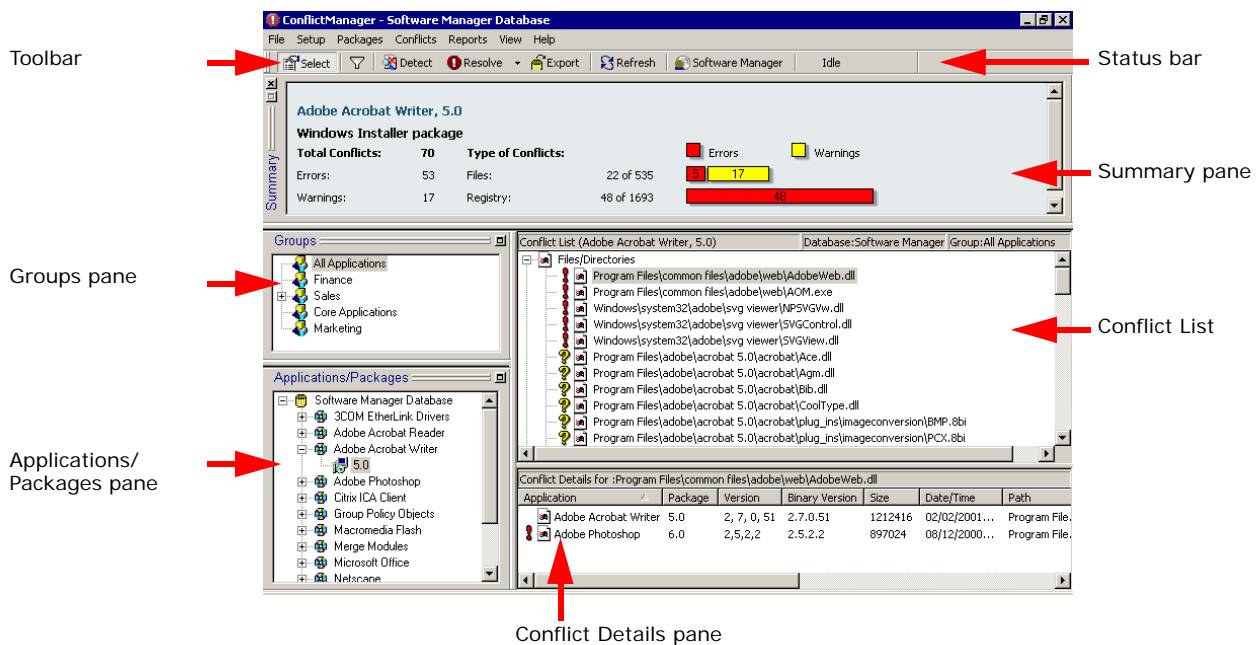
Toolbar

Contains buttons for quick access to many ConflictManager commands and to Software Manager. While conflict detection is running, its status appears at the right end of the toolbar.

Summary Pane

Displays different information depending on what is selected in the Applications/ Packages pane:

- When you select a database, the Summary pane displays the current filter criteria.

- When you select an application, the Summary pane lists the packages contained in that application.

- When you select a package, the Summary pane displays a summary of the types of conflicts, if any, in that package. If the package is read-only, Read Only appears in parentheses at the end of the second line.

The Summary pane initially appears immediately below the toolbar, but you can move or hide it.

Groups Pane

Displays the package group defined for each database. A package group consists of a subset of the packages in the database. When a group is selected, the Applications/ Packages pane lists only the packages assigned to that group. This lets you reduce the number of conflicts that appear.

See *Package Groups* in the Software Manager Help.

The Groups pane initially appears on the left side of the window below the Summary pane, but you can move or hide it.

Applications/Packages Pane

Displays the applications and packages in the Software Manager database. Expand an application to display its packages. Select a package to make it the active package.

(Enterprise Management Server only.) When multiple databases are open, select a database in the Applications/Packages pane to make it the active database. All ConflictManager activities are performed within the active database.

Icons appear in this pane to help you quickly identify items in the database.

See *Icons in Software Manager* in the Software Manager Help.

Conflict List

Displays conflicts between the package selected in the Applications/Packages pane and all other packages in the database. If a package group is selected in the Groups pane, only the conflicts between the selected package and the packages in the group are displayed.

Icons appear in this pane to help you quickly identify the types of resources and conflicts that are displayed.

See Resource Conflicts on page 15 and Types of Conflicts on page 15.

Conflict Details Pane

Displays resources in other packages that conflict with the item selected in the Conflict List. You can sort the items in this pane by clicking a column header, and you can re-order the columns by dragging the column headers. Double-click an item in this pane to display the Properties dialog box.

See Viewing Conflicts on page 29.

Customizing the ConflictManager Workspace

- Hide or show the Packages or Groups panes by clicking the Select tool on the toolbar. Because you often don't need these panes after you use them to make your initial selections, this tool can quickly make your working area less cluttered.

- Hide or show the toolbar or the Summary pane by selecting its name from the View menu.

- Move, dock, and undock the panes as you would any Windows pane, by dragging their title area.

The Conflict List and Conflict Details panes cannot be hidden or moved.

Refreshing the ConflictManager Display

- Select File menu > Refresh.

This clears the Conflict List and Conflict Details panes, rereads the package data, and redisplays the conflict information. This can be useful if packages have been added to the Software Manager database since you started this ConflictManager session.

Viewing Package Attributes

In ConflictManager, you can view package attributes but, except for meta data, you cannot edit them. To edit package attributes, use Software Manager.

See *Viewing and Editing Package Attributes* in the Software Manager Help.

To view package attributes

1. In the Applications/Packages pane, select a package.

   Note
   Be sure to select a package; you cannot display attributes for an application.

2. Select Packages menu > Attributes.

   The Package Attributes dialog box appears.

3. Click the General tab to view general information about the package, or to edit package meta data.

   The **Package Meta Data** section contains both predefined and custom meta data fields. To edit custom meta data, click in the Value column and enter or select a value. If the field requires a URL, include http:// or other tag (mailto:, ftp:, file:) so that the field will become an active hyperlink on the Package pane.

   The predefined meta data is read-only. To edit predefined meta data, use the Product Details page in Windows Installer Editor.

   With Enterprise Management Server, Security Setup determines whether you can edit meta data in Software Manager and ConflictManager.

4. Click the Source tab to view the location of the package installation file and any transforms and patches that were applied to the package.

5. Click OK.

   See also:

About Package Meta Data in the Software Manager Help

About Conflicts

ConflictManager finds and lets you resolve conflicts caused when different packages install the same resources. To save time during conflict detection, ConflictManager only searches for the following situations:

- Duplicate files, of any type, in the Common Files directory and any subdirectory of Windows, including the System directory. Example: If two packages install readme.txt in the Common Files directory, ConflictManager marks them as conflicts. If you don't want ConflictManager to find conflicting readme.txt files, you can exclude that file, or any other files you don't care about, from the conflict detection.

- Duplicate executable files only (.exe, .dll, and .ocx) in all other directories. Example: If two packages install sslib.dll in the Program Files directory, it is considered a conflict.

- When multiple packages install the same system resource, such as registry keys, Autoexec.bat, Config.sys, ODBC, NT services, devices, .ini files, shortcuts, and the PATH variable. Experienced users can resolve registry key and .ini file conflicts in ConflictManager. The other types of conflicts cannot be resolved in ConflictManager and are displayed for informational purposes only.

What Kind of Information Does ConflictManager Compare?

During conflict detection, ConflictManager searches the following information for each package and marks items that conflict:

File hash value
File name
Version
Binary version
Size
Date/time
Path
Company
Product
16 bit (Yes or No)
Registry key value

ConflictManager searches the following information for Windows Installer installations only:

Component GUID
KeyPath
Resource count
Shared DLL

View this information:

- In the Conflict Details pane when you select a conflict in the Conflict List.

- In the Resolve wizard, which you start by selecting Conflicts menu > Resolve.

- On the Properties dialog box, which you open by double-clicking a package in the Conflict Details pane.

Hash Value Comparisons in Conflict Detection

The Import Wizard in Software Manager scans every source file in a package and, if it can find the file, it generates a hash value for that file. If the file cannot be found (example: if its source path is broken), a hash value is not generated.

A hash value is a number that is generated from every byte in the file, in such a way that it is extremely unlikely that some other file will produce the same hash value. ConflictManager can use the file hash value to find conflicts.

Example: Two packages install files with identical names at the same location.

- If both files have a hash value, ConflictManager compares them. Matching hash values mean that the files are the same and a conflict will not occur; non-matching hash values mean that the files are not the same and there is a conflict. Further conflict checks are not necessary.

- If only one file has a hash value, or neither file has a hash value, then ConflictManager compares other file information (version, size, date/time, and so on).

  See About Conflicts on page 12.

Generating the hash values can slow the package import. However, using hash value comparisons results in faster conflict detection and analysis because, once it compares the hash values, ConflictManager does not have to compare anything else. Hash value comparison also results in 100% accurate file conflict detection because it eliminates false positives (example: files that are the same at the byte level but have different date/time stamps).

The hash value is displayed in the Conflict Details pane in ConflictManager, and on the conflict resolution dialogs.

To use hash value comparison, mark the **Use file hash values to determine conflicts** check box in Conflict Settings.

See Defining Types of Conflicts to Detect on page 17.

Finding and Viewing the Right Conflicts

ConflictManager provides flexibility in the types of conflicts you can detect, and lets you filter out items that might otherwise be identified as conflicts. This lets you see only the types of conflicts that are important in your environment.

Define conflict settings.

Conflict settings determine the type of conflicts that are detected and the files and registry keys that are excluded from conflict detection.

See About Conflict Settings on page 17.

Filter the conflict display.

The conflict display filter determines which packages and conflicts appear in the Applications/Packages, Conflict List, and Conflict Details panes.

See Filtering the Conflict Display on page 31.

Ignore conflicts when components are permanent.

When two Windows Installer applications install components with identical resources, but different component GUIDs, ConflictManager detects a conflict. This happens

because Windows Installer installs and uninstalls resources at the component level. Therefore, uninstalling one of the applications could result in the removal of one or more of the resources required by the other application. However, when the components for all such applications are marked to remain permanently installed, this conflict does not occur. The permanent component remains installed even if all of the applications associated with that component are uninstalled. ConflictManager does not consider such cases to be conflicts.

Hide conflicts.

Conflicts that are identified by ConflictManager are often found to be harmless upon further testing. To filter these harmless conflicts from the conflict display, you can designate certain conflicts between specific packages to be ignored, or hidden.

To designate a conflict as hidden, (or to remove the "hidden" designation), click the Application Name column and then right-click the conflict in the Conflict Details pane.

The Conflict Settings dialog box contains a check box, **Display Conflicts Marked as Hidden**, that lets you show hidden conflicts.

See Hiding Conflicts on page 32.

Resolve Formatted Registry Values for Conflict Detection

Registry entries in Windows Installer and WiseScript packages can use formatted text strings that, when evaluated literally, might lead to false positives during conflict detection. To avoid this problem, Software Manager resolves the formatted text strings during import of the following types of packages: .WSI, .MSI, .MSM, .MSP, .MST, .WSE.

Example: Application 1 creates a registry key under HKLM\Software\InstallDirKey with a value of [$comp1]. Application 2 creates a registry key under HKLM\Software\InstallDirKey with a value of [$comp4]. During installation, both [$comp1] and [$comp4] evaluate to the same value. If these values were not resolved, ConflictManager would identify this as a conflict. However, because Software Manager resolves the values during import, and stores them in the Software Manager database, a conflict is not detected, which is the correct behavior.

SoftwareManager resolves all properties in a Windows Installer installation. In a WiseScript installation (.WSE), SoftwareManager resolves only the following predefined variables:

WiseScript variable	Resolves to
%Common%	Program Files\Common
%fonts%	Windows\Fonts
%MainDir%	Program Files*default maindir* or, if *default maindir* is empty: Program Files
%Program_Files%	Program Files
%Sys%	Windows\System
%Sys32%	Windows\System32
%Win%	Windows

When the package resource appears in Software Manager and ConflictManager, the resolved value is displayed instead of the formatted text string.

Resource Conflicts

You use ConflictManager primarily to find file conflicts, but it also can find conflicts between other resources. The resource type is indicated by an icon that appears to the left of the conflict item in the Conflict List and Conflict Details panes.

An icon to the left of the resource icon indicates the conflict type.

See Types of Conflicts on page 15.

Icon	Resource type	Conflict caused by
	Files and components	• Files with identical names and different file hashes that are installed at the same location by multiple packages • Windows Installer components that have matching key files, but one or both contain additional non-advertising resources • Windows Installer components that have additional resources that conflict with a file in a WiseScript installation
	Registry	Registry keys and named values that are set differently by multiple packages
	Autoexec.bat	Changes to Autoexec.bat that are made by multiple packages. These conflicts do not occur in Windows Installer installations.
	Config.sys	Changes to Config.sys that are made by multiple packages. These conflicts do not occur in Windows Installer installations.
	ODBC	ODBC data source and driver conflicts that occur when the registry entries of multiple packages configure the same driver differently
	Service	Windows NT services of the same name that are installed by multiple packages
	Device	Device drivers that are installed in Win.ini by multiple packages. These conflicts do not occur in Windows Installer installations.
	.INI file	.INI files and entries that are changed by multiple packages
	Shortcut	Shortcuts of the same name that are installed by multiple packages
	Path	Changes to the PATH variable made by multiple packages. These conflicts do not occur in Windows Installer installations.

Types of Conflicts

Conflicts are classified as either warnings, errors, or information. After conflict detection, conflicts are displayed in the Conflict List and Conflict Details panes. An icon to the left of the resource icon indicates the conflict type.

See Resource Conflicts on page 15.

An icon whose color is dimmed indicates that the conflict has been designated as hidden.

See Hiding Conflicts on page 32.

Icon	Conflict type	Represents
	Warning	Conflicts that are not critical but might require some attention. A warning occurs when multiple packages install a non-16-bit file with the same file name but different hash values or different file information (file version number or date/time) to different directories.
	Error	More serious conflicts that deserve careful inspection. Errors can occur in the following situations: ● When multiple packages install files of the same name and different hash values to the same directory. ● When multiple packages install files of the same name and same hash value to the same directory, but the component GUIDs do not match. ● When multiple packages install a non-16-bit file of the same name with different file information to the same directory. Because 16-bit DLLs are shared by all packages, any 16-bit DLLs installed to different directories with the same file name but different file information are marked as errors. ● When multiple packages set the same registry key to different values. Other types of errors are possible. See Guidelines for Resolving File Conflicts on page 33.
	Information	● When multiple packages install a non-16-bit file with identical file information or the same hash value to the same directory or to different directories. ● When multiple packages set the same registry key to the same value. Because these are not true conflicts, they do not appear in ConflictManager. However, they appear in the Package Details pane in Software Manager.

Chapter 2
Setting Up ConflictManager

This chapter includes the following topics:

- *About Conflict Settings* on page 17
- *Conflict Resolution Rules* on page 22

About Conflict Settings

Conflict settings determine the type of conflicts that are detected and the files and registry keys that are excluded from conflict detection. This helps eliminate conflicts that don't have a negative impact on any installation. Example: readme.txt is installed by many packages, but generally is not a conflict that you need to resolve. Therefore, you could specify that file to be excluded from conflict detection.

These settings are stored in the Software Manager database and affect any user who is connected to that database.

You define the conflict settings on the Conflict Settings dialog box, which you access by selecting Setup menu > Conflict Settings. This dialog box contains the following tabs:

Types	Define the kinds of conflicts to detect.
	See *Defining Types of Conflicts to Detect*.
Files / Directories	Specify the directories and files to ignore during conflict detection.
	See *Excluding Files and Directories From Conflict Detection* on page 20.
Registry Keys	Specify the registry subtrees and individual keys to ignore during conflict detection.
	See *Excluding Registry Keys From Conflict Detection* on page 21.

Note
If you change conflict settings after you run conflict detection, you must rerun conflict detection to apply the changes.

Defining Types of Conflicts to Detect

These settings are stored in the Software Manager database and affect any user who is connected to that database.

To define types of conflicts to detect

1. Select Setup menu > Conflict Settings > Types tab.

2. From **File Conflicts**, select an option.

■ **Normal file conflicts (warnings and errors only)**
Find files with the same name but different file information that are installed to the same directory or to different directories by multiple packages.

■ **Normal conflicts plus informational**
Find files with identical file information that are installed to the same directory or to different directories by multiple packages. Finding information conflicts slows conflict detection.

■ **Do not detect file conflicts**
Run conflict detection without looking for file conflicts. You might select this option to check for a specific type of conflict other than file conflicts.

3. **Use file hash values to determine conflicts**

The Import Wizard in Software Manager scans every source file in a package and, if it can find the file, it generates a hash value for that file. Mark this to use the hash value to find conflicts. Using hash value comparisons results in faster conflict detection and analysis because, once it compares the hash values, ConflictManager does not have to compare anything else.

See *Hash Value Comparisons in Conflict Detection* on page 13.

4. **If file has version information, ignore date/time for conflict detection**

Mark this to ignore date and time differences between files that have the same version and size. Clear this to have the conflict detection find date and time differences when all other file information is the same.

5. **Ignore Component GUID mismatches if both files have the shared DLL counter set**

Normally, installing two files with the same name to the same location when their component GUIDs do not match causes a conflict, even if the shared DLL counter is set for both files. When the shared DLL counter is set, problems are not likely to occur. Mark this check box to ignore such conflicts.

6. From **Registry Conflicts**, select an option.

■ **Normal Registry conflicts (warnings and errors only)**
Find registry keys that are set to different values by multiple packages.

■ **Normal Registry plus informational conflicts**
Find registry keys that are set to the same value by multiple packages. Finding information conflicts slows conflict detection.

■ **Do not run registry conflicts**
Run conflict detection without looking for registry conflicts. You might select this option to check for file conflicts only, or for a specific type of conflict other than registry conflicts.

7. From **Component Conflicts**, select an option to determine whether to detect Windows Installer component conflicts.

■ **Component conflicts involving files only**
Find component conflicts involving files (example: components that install the same file but have different component GUIDs). Although these conflicts are legitimate, some organizations choose to ignore them because they believe that the Windows Installer self-healing functionality will resolve most of the problems they might cause.

- **Component conflicts involving registry keys only**
Find component conflicts involving registry keys (example: components that update or create the same registry key but have different component GUIDs). Because these conflicts cannot be resolved from within ConflictManager, you might not want to see them.

- **Component conflicts involving files or registry keys**
Find both types of component conflicts.

- **Do not display any component conflicts**
Ignore component conflicts.

8. Mark any of the following check boxes to find additional conflicts.

Note

These check boxes are not often used because these types of conflicts are rarely found and cannot be resolved. Leaving these check boxes cleared speeds conflict detection.

- **Autoexec.bat**
Find changes to Autoexec.bat that are made by multiple packages.

- **Config.sys**
Find changes to Config.sys that are made by multiple packages.

- **ODBC**
Find Open Data Base Connectivity (ODBC) data sources that are configured differently by multiple packages.

- **NT Services**
Find Windows NT services that are installed by multiple packages.

- **Devices**
Find device drivers that are installed in Win.ini by multiple packages.

- **INI Files**
Find changes to .ini files that are made by multiple packages.

- **Shortcuts**
Find shortcuts that are installed by multiple packages.

- **Path Additions**
Find changes to the PATH variable that are made by multiple packages.

9. To run conflict detection when a package is imported into the Software Manager database, mark **Detect conflicts during package import**. This is marked by default.

You can override this setting during import by marking or clearing the **Detect conflicts during package import** check box on the Import Type page of the Import Wizard.

When conflict detection runs during import, it checks for conflicts between the imported package and all other packages; it does not re-detect conflicts between all packages.

10. To show conflicts that are designated as hidden, mark **Display Conflicts Marked as Hidden**.

You can designate certain conflicts between specific packages to be ignored during conflict detection. This lets you filter out conflicts that you determine to be

unimportant. To designate a conflict to be ignored, (or to remove the "ignore" designation), right-click the conflict in the Conflict Details pane.

11. To define additional settings, see:

- *Excluding Files and Directories From Conflict Detection* on page 20

- *Excluding Registry Keys From Conflict Detection* on page 21

12. Click OK on the Conflict Settings dialog box.

See also:

Finding and Viewing the Right Conflicts on page 13
Types of Conflicts on page 15

Excluding Files and Directories From Conflict Detection

On the Files/Directories tab of the Conflict Settings dialog box, specify files and directories to ignore during conflict detection. You can use wildcards to ignore all instances of a particular type of file within a directory.

Examples:

- Readme.txt

- MSCREATE.DIR, entered as a wildcard under All Directories

- *.txt

As you use ConflictManager, you might find additional files and directories that can be ignored. You can add them to the conflict settings at any time.

These settings are stored in the Software Manager database and affect any user who is connected to that database.

The upper-left list box contains a composite of all directories created on the destination computer by all packages in the Software Manager database. Some installed files or directories might be located in the Windows directory or the Program Files directory. When you select a directory in the upper-left list box, the upper-right list box contains the files in that directory.

Note
The upper-left list box contains information for all packages, even if a package group is selected in the Groups pane.

The lower list boxes contain the directory trees and files to ignore during conflict detection.

To exclude files from conflict detection

1. Select Setup menu > Conflict Settings.

   The Conflict Settings dialog box appears.

2. Click the Files/Directories tab.

3. In the upper-left list box, navigate to a directory.

4. In the upper-right list box, select a file name and click Add File.

   The file is added to the lower-right list box, and the corresponding directory structure appears in the lower-left list box.

5. To define additional settings, see:

- *Defining Types of Conflicts to Detect* on page 17
- *Excluding Registry Keys From Conflict Detection* on page 21

6. Click OK on the Conflict Settings dialog box.

Note

If you delete a file from the lower-right list box and it empties the directory, the directory structure is still displayed temporarily but disappears after you close the Conflict Settings dialog box.

To exclude directories or wildcards from conflict detection

1. Select Setup menu > Conflict Settings.

   The Conflict Settings dialog box appears.

2. Click the Files/Directories tab.

3. In the upper-upper-left list box, navigate to a directory and click Add Tree.

   The Add Wildcard dialog box appears.

4. On the Add Wildcard dialog box:

   - To ignore all files in the directory, leave the **Wildcard** field blank.
   - To ignore only certain files in the directory, enter a wildcard in the **Wildcard** field. (Example: *.txt)
   - To apply your choice to all subdirectories of the directory also, mark **Include Sub-directories**.

5. Click OK on the Add Wildcard dialog box.

   The Conflict Settings dialog box reappears with the directory added to the lower-left list box. The lower-right list box contains files in that directory that should be ignored, and indicates whether they are applied to subdirectories. If you entered a wildcard, it appears in the Filename column. If you left the **Wildcard** field blank, a single asterisk (*) appears in the Filename column.

6. To define additional settings, see:

   - *Defining Types of Conflicts to Detect* on page 17
   - *Excluding Registry Keys From Conflict Detection* on page 21

7. Click OK on the Conflict Settings dialog box.

Excluding Registry Keys From Conflict Detection

On the Registry Keys tab of the Conflict Settings dialog box, specify registry key subtrees and registry values to ignore during conflict detection. As you use ConflictManager, you might find additional registry keys that can be ignored. You can add them to the conflict settings at any time.

These settings are stored in the Software Manager database and affect any user who is connected to that database.

The upper list boxes contain registry keys and values for all packages in the Software Manager database. The lower list boxes contain the keys and values to be ignored

during conflict detection. The left list boxes contain key structure, and the right list boxes contain values.

Note

The upper-left list box contains information for all packages, even if a package group is selected in the Groups pane.

To exclude registry keys from conflict detection

1. Select Setup menu > Conflict Settings.

    The Conflict Settings dialog box appears.

2. Click the Registry Keys tab.

3. In the upper-left list box, navigate to a registry key or value.

    ■   To ignore an entire tree, select the key in the upper-left list box and click Add Tree.

    ■   To ignore a value, select the value in the upper-right list box and click Add Value.

    The registry key tree or value is added to the corresponding lower list box.

4. To define additional settings, see:

    ■   *Defining Types of Conflicts to Detect* on page 17

    ■   *Excluding Files and Directories From Conflict Detection* on page 20

5. Click OK on the Conflict Settings dialog box.

Note

If you delete a value from the lower-right list box and it makes the key empty, the corresponding registry key subtree is still displayed temporarily but disappears after you close the Conflict Settings dialog box.

Conflict Resolution Rules

The ConflictManager resolution rules let you resolve file conflicts automatically. Using resolution rules saves time, reduces errors, and provides consistency in conflict resolution.

After you detect conflicts, you resolve the conflicts according to a rule set you select. The majority of conflicts can be resolved this way. If conflicts remain after resolving with rules, use the Resolve wizard.

See:

Resolving File Conflicts With Rules on page 37
Resolving File Conflicts Individually on page 38

About Rule Sets

A rule set is a collection of rules. A rule consists of actions, conditions, and sort options.

●   An action states how a conflict will be resolved.

●   A condition determines the criteria that a conflict must meet in order for the action to be performed. Example: If you select the condition Version differs, the Resolve

with Rules process performs the action only on conflicts caused by files having different version numbers.

- A sort option determines how conflicting files are sorted and thus which one is used to resolve the conflict. Example: If you sort by version number, then the file with the highest version number is used.

ConflictManager contains predefined rule sets. You can edit the predefined rule sets or create new rule sets to reflect your corporate standards.

See *Creating a New Rule Set* on page 24.

Before you create your own rule sets, experiment with the predefined rules. You might find that the predefined rule sets resolve your conflicts satisfactorily and you don't have to create your own.

Rule sets are stored with the Software Manager database, therefore, if you use multiple databases, you must set up rule sets separately in each one. The predefined rule sets are added to each new database you create. If you open a database created in a previous version of ConflictManager, it is updated and the predefined rule sets are added.

See also *Predefined Rule Sets* on page 23.

How Conflict Resolution Rules Are Applied

Conflict resolution rules are applied in the order they appear from top to bottom in the list of rules on the Conflict Resolution Rules dialog box.

When a rule contains multiple conditions, only conflicts that meet all of the conditions have the rule applied to them.

Predefined Rule Sets

ConflictManager contains the following predefined rule sets:

Conservative	This rule set changes as little as possible to get the package to work. It replaces files in the active package only with the corresponding files in the conflicting packages. An active package file is changed only if its version and date are lower than that of the conflicting file, and only if the manufacturer is the same as in the conflicting file.
	This rule set ignores conflicts that cannot be resolved in a conservative way. This can result in more conflicts for you to resolve with the Resolve wizard.

Aggressive (using application paths) Aggressive (using isolated components)	These rule sets change as many files as possible in the active package and can change files in other conflicting packages. They replace all conflicting files with the most recent version; this means that files in the other conflicting packages might be replaced. Do this only when you have access to the other packages' source files and can recompile the other packages. The aggressive rule sets also try to isolate any files that are shared by multiple packages. The aggressive rule sets can resolve more conflicts than the conservative rules. The disadvantage is that you must export, recompile, and retest any other packages that are changed.
Restore original conflicts	This empty rule set provides an easy way to undo a package's conflict resolutions and revert to its original file versions and locations. When you resolve with rules, the first thing any rule set does is revert all previous conflict resolutions for the selected package. Because this rule set contains no rules, that's all it does.

Creating a New Rule Set

If the predefined rule sets do not meet your needs, and you don't want to edit a predefined rule set, you can create a new rule that reflects your corporate standards.

To create a new rule set

1. Select Setup menu > Conflict Resolution Rules.

   The Conflict Resolution Rules dialog box appears. The upper list box contains the rules in the selected rule set. Select a rule to display its details in the lower list box (the **Rule description** list).

2. From **Rule Set Name**, select **<new>**.

   The Enter Rule Set Name dialog box opens.

3. In **Rule Set Name**, enter a unique name to identify this rule set and click OK.

4. On the Conflict Resolution Rules dialog box, click Add to add the first rule for this rule set.

   The Rules wizard starts and opens the Name page.

5. Enter a name for this rule and click Next.

   The Actions page appears.

6. In **Which action(s) do you want to perform?**, mark the check box next to each action to perform, in the order they should be performed.

   The actions you mark appear in the **Rule description** list.

   Note
   When you mark an action, any other actions that are incompatible with it are unavailable.

7. If an action contains underlined text, click the underlined text to open a Rule Detail Settings dialog box, where you can specify a value for the underlined text.

    Example: If you marked the action **Isolate the file via AppPaths into the directory Shared**, you would click the word <u>AppPaths</u> and select the isolation method, and then click the word <u>Shared</u> and enter a directory name.

8. When you finish adding actions, click Next on the Actions page.

    The Conditions page appears.

9. In **Which condition(s) do you want to check?**, mark the check box next to each condition to check, in the order they should be checked.

    As you mark check boxes, the conditions appear in the **Rule description** list.

10. If a condition contains underlined text, click the underlined text to open a Rule Detail Settings dialog box, where you can select a value for the underlined text.

    Example: If you marked the condition **File is executable**, you would click the word <u>is</u> and select either **is** or **is not**.

11. When you finish adding conditions that comprise the rule, click Next on the Conditions page.

    The Sort Order page appears.

12. In **Which sort(s) do you want to perform?**, mark the check box next to each sort option to perform, in the order they should be performed. You can rearrange the sort order later.

    As you mark check boxes, the sort options appear in the **Rule description** list.

13. To rearrange the sort order, select a sort option in the **Rule description** list and click Move Up or Move Down.

14. When you finish adding and arranging sort options, click Finish on the Sort Order page.

    The Conflict Resolution Rules dialog box reappears. The new rule is displayed in the upper list box and its details are displayed in the **Rule description** list.

15. To edit a rule's details, delete a rule, or rearrange the rule order, use the buttons on the Conflict Resolution Rules dialog box.

16. To change the value in an action or condition, click its underlined text and enter new text on the Rule Detail Settings dialog box.

17. You can continue to add and edit rules. When you finish, click OK on the Conflict Resolution Rules dialog box.

See also:

Guidelines for Resolving File Conflicts on page 33

Chapter 3
Conflict Detection and Resolution

This chapter includes the following topics:

About Conflict Detection and Resolution

After you import packages into the Software Manager database, use ConflictManager to:

- Detect conflicts.
- Display conflicts.
- Resolve conflicts.
- Export the changes to the original installation file and recompile it.
- Generate ConflictManager reports.

You also can detect and resolve conflicts in Windows Installer Editor. Do this to compare a package that is still in the authoring phase with those that have already been deployed, without importing the package into the Software Manager database. This lets you:

- Streamline repackaging by identifying conflicts earlier in the process.
- Maintain your Software Manager database as a pristine image of packages that have already been deployed in the production environment.

See *Resolving File Conflicts Within Windows Installer Editor* in the Windows Installer Editor Help.

Prerequisites for Conflict Detection and Resolution

Before you use ConflictManager to detect and resolve conflicts:

- Define conflict settings, which determine the type of conflicts that are detected and the files and registry keys that are excluded from conflict detection.
  See About Conflict Settings on page 17.
- Decide whether to use conflict resolution rules and, if so, decide which predefined rule sets to use. You can edit the predefined rule sets or create new rule sets.

See Conflict Resolution Rules on page 22.

- Import packages into the Software Manager database, using Software Manager. See *Package Import* in the Software Manager Help.

Note
Conflict detection requires package resources to be in the Software Manager database. If a package's meta data is in the database but not its resources, you must perform an import in Software Manager.

Process for Resolving Conflicts

Typically, you will follow the steps below in the order listed. In some cases, however, you might not follow all these steps. Example: During the analysis phase of a repackaging project, you might want to check for conflicts but not resolve them. In that case, you would perform the conflict detection step only.

1. **Detect conflicts.**

    ConflictManager compares the resources each package installs. When it finds resources that conflict, it populates the Software Manager database with conflict information.

    View conflicts in the Applications/Packages pane and the Conflict List, or use the ConflictManager reports to display and print conflict information.

    See Detecting Conflicts on page 28 and Viewing Conflicts on page 29.

2. **Resolve conflicts.**

    Resolving a conflict involves looking at each file that is installed by multiple packages and selecting the version to install on the destination computer. You also can change the location of conflicting files so that each package can use its version of the file.

    You can use conflict resolution rules to resolve conflicts automatically, or run the Resolve wizard, which lets you review and resolve file conflicts one at a time.

    See Conflict Resolution on page 33.

3. **Export resolved packages.**

    After you resolve conflicts, export the changes to the original Windows Installer or WiseScript installation to produce an installation that does not conflict with other packages.

    See Package Export After Conflict Resolution on page 43.

4. **Test the resolved installation.**

    After you save and recompile a resolved package, install it on a clean machine and test it.

    Note
    It is important to test the package on a clean machine. A machine that is not clean might contain newer versions of some of the files you resolved. Because these newer files are not overwritten during the installation, you do not get a true representation of the resolved installation.

    ---

5. **Redo conflict resolution if necessary.**

If the package does not work, return to ConflictManager and redo the conflict resolution in any of the following ways.

To revert changes for just one or a few files	• Select Conflicts menu > Resolve.
	• In the Resolve wizard, select one or more files and click Revert to undo your previous conflict resolutions for those files.
	• Try other ways to resolve the conflicts.
To redo the conflict resolutions with rules	• Select Conflicts menu > Resolve with Rules.
	• Select a different rule set. This reverts all the previous conflict resolutions for that package and then resolves them using the different rule set.
To revert all previous resolutions without applying a new rule set	• Select Conflicts menu > Resolve with Rules.
	• Select the Restore original conflicts rule set. This reverts all previous conflict resolutions but does not resolve the conflicts.
	• Resolve conflicts with a different rule set or with the Resolve wizard.

After you redo the conflict resolution, repeat the steps to export and test the package.

Detecting Conflicts

ConflictManager compares the resources each package installs. When it finds resources that conflict, it populates the Software Manager database with conflict information. The database contains the combined results of all the detection processes you have performed. Repeating the detection process for a package overwrites previous conflict information for that package only.

The settings in Setup menu > Conflict Settings > Types determine which conflicts are detected.

If **Detect conflicts during package import** is marked in Conflict Settings, then conflict detection runs automatically when a new package is imported into the Software Manager database. When conflict detection runs during import, it checks for conflicts between the imported package and all other packages; it does not re-detect conflicts between all packages.

To detect conflicts

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. Select Conflicts menu > Detect.

   The Detect Conflicts dialog box appears.

3. Mark one of the following:

- **Detect conflicts between all packages**
  Find conflicts between every package and all other packages in the database.

- **Detect conflicts for selected packages**
  Find conflicts between the package you select and all other packages in the database.

  Marking this option enables the list of packages on the dialog box. In the list, select one or more packages to find conflicts for.

  Example: Suppose your database contains Microsoft Word, Lotus Notes, and Adobe Acrobat. When you select Word, ConflictManager finds resources in Word that conflict with Lotus or Acrobat. It also finds resources in Lotus and Acrobat that conflict with Word. It does not find resources in Lotus that conflict with Acrobat.

4. Click OK.

Depending on the number of packages in the Software Manager database and the speed of your computer, conflict detection can take several minutes. While conflict detection is running:

- Its status appears at the right end of the toolbar.

- You can continue to use ConflictManager.

- You cannot close ConflictManager.

When conflict detection is complete, the right end of the toolbar displays Idle and you can review conflicts.

See Viewing Conflicts on page 29.

Stopping Conflict Detection

Stop conflict detection if you need to close ConflictManager.

Stopping conflict detection before it finishes can leave partial records in the database. If you must stop conflict detection, be sure to rerun it later. You cannot revert the database to its previous state unless you have a current backup copy.

- Select Conflicts menu > Stop.
  The right end of the toolbar displays Idle.

Viewing Conflicts

The Conflict List displays conflicts between the selected package and all other packages in the database. You can view conflicts for only one package at a time.

You can view additional information about a conflicting resource by displaying its Properties dialog box. You also can view, but not edit, properties in the Package Details pane in Software Manager.

You can use package groups to reduce the number of conflicts that appear. A package group consists of a subset of the packages in the database.

See *Package Groups* in the Software Manager Help.

To view conflicts

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. If you are using package groups, select a group name in the Groups pane.

   The Applications/Packages pane displays only packages that are assigned to that group.

3. In the Applications/Packages pane, select a package.

4. The package's conflicts appear in the Conflict List and are grouped by type.

   See Resource Conflicts on page 15.

   If a package group is selected in the Groups pane, only the conflicts between the selected package and the packages in the group are displayed.

5. To change the types of conflicts that appear, select Conflicts menu > Filter and set the filter criteria.

   See Filtering the Conflict Display on page 31.

   Note
   The settings in Setup menu > Conflict Settings > Types determine which conflicts are detected. Example: If your conflict settings are set to detect file conflicts only, then registry conflicts do not appear even when the conflict filter is set to display registry conflicts.

6. In the Conflict List, select a conflict.

   The conflicting resources in other packages appear in the Conflict Details pane.

7. To view a conflicting resource's properties, double-click it in the Conflict Details pane.

   The Properties dialog box appears.

   Note
   Because they are rare, the Properties dialog box is not available for Autoexec.bat, Config.sys, service, device, shortcut, and path conflicts.

Warning
Changing the information on the Properties dialog box is not recommended except for registry keys and .ini files, and then only if you are the developer of the package and you have extensive knowledge of how Windows operates. Changing property information incorrectly can damage the package.

See Resolving Registry Conflicts on page 40 and Resolving INI Conflicts on page 41.

When Conflict Detection Yields Unexpected Results

If conflict detection results in conflicts that you did not expect, or ignores conflicts that you did expect, here's what to do:

- Review how ConflictManager detects conflicts. See:
  About Conflicts on page 12
  Hash Value Comparisons in Conflict Detection on page 13
  Finding and Viewing the Right Conflicts on page 13

- Check your conflict settings to verify that ConflictManager is detecting the types of conflicts that are important to you.
  See About Conflict Settings on page 17

- Verify that ConflictManager is set to display the types of conflicts that you want to see. See:
  Filtering the Conflict Display
  Hiding Conflicts on page 32

Filtering the Conflict Display

You can specify filter criteria to determine which packages and conflicts appear in the Applications/Packages, Conflict List, and Conflict Details panes. You can filter the display by conflict attributes and package attributes. The filter criteria you specify remain the same for subsequent uses of ConflictManager.

To filter the conflict display

1. Select Conflicts menu > Filter.

   The Filters dialog box appears.

2. To filter by conflict attributes, click the Conflicts tab and mark any combination of check boxes.

   - **Errors**
     Include conflicts that will cause problems.

   - **Warnings**
     Include conflicts that are not critical but might require some attention.

   - **File Conflicts**

   - **Registry Conflicts**

   - **Other Conflicts**
     Include ODBC and .ini file conflicts.

   - **Only display packages with conflicts**

   - **Display resolved conflicts**
     Include resolved conflicts in addition to those that haven't been resolved.

3. To specify filter criteria to determine which packages appear in the Applications/Packages pane, click the Packages tab.

   - To filter by package status, mark any combination of check boxes in the **Package Status to Display** section.

   - (Enterprise Management Server only.) If multiple databases are open, select from **Database** a database containing the meta data field to filter by.

   - To filter by whether the package is defined, mark check boxes in the **Filter Based on Package Definition** section.

     ♦ **Display Defined Packages** displays packages that were defined in Workbench and Software Manager.

     ♦ **Display Undefined Packages** displays all undefined packages.

- To filter the display by a meta data value, mark **Filter Based on Meta Data**. Then specify the following filter criteria:

    ♦ **Name**
      Select a meta data field name to filter packages by. The drop-down list contains meta data fields that are defined in the selected database.

    ♦ **Operation**
      Select the operation to use when comparing the meta data value to the value you enter below. The operations that are available depend on the data type of the meta data field.

    ♦ **Value**
      Specify the meta data value to match.

- To filter by package type, mark any combination of check boxes in the **Package Type to Display** section.

    To clear or mark a group of package types, select the group of package types and click one of their check boxes.

4. Click OK.

Note

The settings in Setup menu > Conflict Settings > Types determine which conflicts are detected. Example: If your conflict settings are set to detect file conflicts only, then registry conflicts do not appear even when the conflict filter is set to display registry conflicts.

Hiding Conflicts

Conflicts that are identified by ConflictManager are often found to be harmless upon further testing. Normally, ConflictManager will continue to display these conflicts. To filter these harmless conflicts from the conflict display, you can designate certain conflicts between specific packages to be ignored, or hidden. Example: Designate that any conflicts involving abc.dll between Package A and Package B should be ignored

When you hide a conflict, it does not appear in:

- The Conflict Details pane.

- The Resolve wizard.

- ConflictManager reports.

To designate a conflict as hidden

In the Conflict Details pane, right-click the conflict and select Hide Conflict. The icon to the left of the conflict is dimmed.

To show or hide hidden conflicts

1. Select Setup menu > Conflict Settings and click the Types tab.

2. Mark or clear **Display Conflicts Marked as Hidden** to determine how ConflictManager filters hidden conflicts.

Conflict Resolution

Resolving a conflict involves looking at each file that is installed by more than one package and selecting the version to install on the destination computer. You also can change the location of conflicting files so that each package can use its version of the file.

See *Guidelines for Resolving File Conflicts.*

Your options for resolving conflicts are as follows:

Resolve conflicts automatically	Use conflict resolution rules to resolve file conflicts automatically.
	See Resolving File Conflicts With Rules on page 37.
Resolve individual conflicts manually	Run the Resolve wizard, which lets you review and resolve file conflicts one at a time.
	See Resolving File Conflicts Individually on page 38.

The Resolve with Rules and Resolve commands resolve file conflicts only. To resolve registry key value and .ini file conflicts, you must use the Properties dialog box.

See Resolving Registry Conflicts on page 40 and Resolving INI Conflicts on page 41.

You cannot resolve conflicts in:

- Packages that are marked as read-only on the Package Attributes dialog box in Software Manager.
- Certain types of packages.
  See Restrictions on Resolving and Exporting Package Types on page 42.

Guidelines for Resolving File Conflicts

Before you resolve a conflict, analyze the conflicting file's version, date/time, size, and other information to determine the cause of the conflict. You can view the file information in either the Conflict Details pane or the Resolve wizard pages. If you use conflict resolution rules, the rules perform the conflict analysis and resolution.

The following table provides guidelines for resolving file conflicts. The Recommended Resolution column describes in general terms how to resolve the conflict, specifies the button to click to resolve the problem in the Resolve wizard, and specifies the rule action to use when resolving with conflict resolution rules.

Conflict	Problem	Recommended resolution
Two files with the same name have different hash values, or a different version, date/time, or size.	If the installation does not use versioning rules (WiseScript replacement options or the Windows Installer REINSTALLMODE property), the older version can install over the newer version, causing a problem when each package requires a specific version of the file. Also, the new file might not be compatible with the old one.	Copy the file information (source path included) from one package to another, or isolate the file. See Copying File Information on page 35 and Isolating Files on page 36.
Two files with the same name and the same hash value are installed to the same directory, but component GUIDs do not match, and none of the components are marked to be permanently installed.	After both packages are installed, uninstalling one of them will remove the file, breaking the remaining package.	Match GUIDs for identical files across packages. **Resolve wizard:** Copy Up or Copy Down button **Rule action:** Replace the MSI component GUID with the matching conflicting file's GUID Or Use the MSI component GUID in the active package to replace the MSI component GUID in the conflicting packages
File is KeyPath to a component in one package, but not in another.	After both packages are installed, uninstalling one of them will remove the file, breaking the remaining package.	Make both files KeyPaths to their respective components. **Resolve wizard:** Fix Comp button **Rule action:** Use the MSI component KeyPath in the active package to replace the MSI component KeyPaths in the conflicting packages Or Replace the MSI component KeyPath with the matching conflicting file's KeyPath

Conflict	Problem	Recommended resolution
The shared DLL counter is not set for a file shared by a WiseScript package and a Windows Installer package.	In a mixed WiseScript/Windows Installer environment, shared DLL counters must be set so the WiseScript's shared DLL reference counts are up to date. Otherwise, uninstalling either package will break the other.	Set the Shared DLL flag on one or both files. **Resolve wizard:** Fix Comp button **Rule action:** Set shared DLL counters if required

Copying File Information

When you resolve a conflict by copying file information, you either replace the active package's file with the matching conflicting file or replace the conflicting file with the active package's file.

When to copy file information

Copy file information when the conflicting files are two versions of the same file, that is, all information is the same except version, date/time, or size.

Do not copy file information when:

● The older file is substantially larger than the newer file. The older file might contain functions that the new version does not. In this case, you might have to experiment to find which file works best.

● Two files have the same name but different values in the **Company** field, meaning they were created by different manufacturers. These errors are difficult to resolve, because the two files might be entirely different. Example: If they are DLL files, they might have different APIs and different functions. In this case, it is best to isolate the files.

See Isolating Files on page 36.

How to copy file information

You can copy files using a conservative method, which does not change other packages, or an aggressive method, which changes other packages. You must export, recompile, and test any other packages that you change.

The following table describes how to copy file information using a conservative or aggressive method.

Method	How to copy with the Resolve wizard	How to copy with rules
Conservative	Use the Copy Up or Latest button	● Use a conflict resolution rule containing this action: Replace the active application's file with the matching conflicting file ● Add a condition to compare the package file version to the conflicting files, and sort by version.

Method	How to copy with the Resolve wizard	How to copy with rules
Aggressive	Use the Copy Down button	• Use a conflict resolution rule containing this action: Replace the conflicting files with the active application's file • Add a condition to compare the package file version to the conflicting files.

Isolating Files

Isolating files to resolve conflicts is generally safer than copying them.

When to isolate files

- When different packages install the same executable file (.dll, .ocx, or .exe) to the System directory or to any common directory.

- When different packages install files having the same name but different values in the **Company** field, meaning they were created by different manufacturers.

Isolation works by moving one or both of the files to their respective application directories and telling the operating system to look for the file in the application directory before looking in the System directory.

Note
When you isolate a file, the file is still copied to the System directory in addition to the application directory. If the System directory already contains an earlier version of the file, the earlier version is replaced. This could cause a problem in any package that uses the earlier version of the file. Therefore, it is best to isolate both versions of the file in their respective packages.

File isolation methods

Isolation method	How it works	When to use
Isolated components	Isolate files using Windows Installer isolated components. The isolation is managed by the operating system.	• For .MSI and .WSI packages only • Whether or not conflicting files are listed in the registry
Application paths (AppPaths)	Moves files out of the System directory and into a private directory, typically the application directory.	• For Windows Installer and WiseScript packages • For installations intended for any Windows operating system • When conflicting files are not listed in the registry (No programs will try to find it by looking in the registry.)

When the conflicting files are listed in the registry, using an application path to move them to a private application directory can cause problems. ConflictManager updates the

registry when a file is moved to a private application directory. This can cause other packages to follow the moved file and result in a further conflict. Experiment to determine which packages find the file in the registry and which look in the path. Then move the file for the packages that look in the registry, and leave a copy in the System directory for packages that look in the path.

How to isolate a file with the Resolve wizard

In the Welcome page of the Resolve wizard, select one of the following options from the **Isolation Method** drop-down list:

- **Isolated Components**

- **Application Paths**
  Selecting this option enables the **Application Path Settings** field. Enter the path where Windows should look for the files. This path is created under the main application directory.

Continue using the Resolve wizard.

See

How to isolate a file with rules

Use a rule containing the following action:

Isolate the file via [AppPaths OR Isolated Components]

Both of the predefined aggressive rule sets contain this action.

If you create your own rules, add this action and set its variable to AppPaths or Isolated Components. When you select the AppPaths variable, a second line is added to the action:

into directory Shared

If you leave this line as is, the application path is set to a directory named Shared under the main application directory. To use a different application path, click the word Shared and type a new directory name.

See

Resolving File Conflicts With Rules

Using conflict resolution rules is the fastest way to resolve conflicts. The rules do the conflict analysis and resolve the conflicts automatically. This saves time, reduces errors, and provides consistency in conflict resolution.

Resolving with rules always deletes all previous conflict resolutions for the selected package.

To resolve file conflicts with rules

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. Select Conflicts menu > Resolve with Rules.

   The Resolve with Rules dialog box appears.

3. Specify the application and package to resolve conflicts for.

4. From **Rule Set Name**, select the rule set to use.

5. Click OK.

    The Resolving Conflicts dialog box appears, and conflicts are resolved. If there are few conflicts to resolve, the dialog box appears briefly.

6. When conflict resolution is finished, view conflicts for that package. If conflicts appear, they could not be resolved with rules. Resolve the remaining conflicts manually. See:

    - Resolving File Conflicts Individually on page 38
    - Resolving Registry Conflicts on page 40
    - Resolving INI Conflicts on page 41

    See also:

    Conflict Resolution Rules

Resolving File Conflicts Individually

Use the Resolve wizard to resolve file conflicts without using resolution rules, or to resolve conflicts that cannot be resolved automatically with rules.

To resolve individual file conflicts

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. Select Conflicts menu > Resolve.

    The Welcome page appears.

3. Specify the application and package to resolve conflicts for.

4. From **Isolation Method**, select an option. Selecting an isolation method activates the Move button on the File Conflicts and File Conflicts in Registry dialog boxes.

    See Isolating Files on page 36.

    - **Do not move files**
      Do not allow file isolation.

    - **Isolated Components**
      Isolate files using Windows Installer isolated components. The isolation is managed by the operating system. This method works with .MSI and .WSI packages only.

    - **Application Paths**
      Moves files out of the System directory and into a private directory. This method works for Windows Installer and WiseScript packages, and is supported by all Windows operating systems.

      By default, the files are placed in the application directory. To place files in a different directory, in **Application Path Settings**, specify a subdirectory of the application directory or the full path to a different directory. You can use variables in the path.

5. Click Next.

    The File Conflicts page appears. The upper list box contains files in the selected package that have conflicts and are not listed in the registry. You usually can move

such files without causing problems. The lower list box contains the conflicting files in other packages.

6. In the upper list box, select one or more files and take one of the following actions.

   See Guidelines for Resolving File Conflicts on page 33.

   When you resolve a conflict, the exclamation point to the left of the file name changes from red to white.

   ■ To move the selected file to a private directory and change the file path in the package, click Move. This button is not available if you selected the **Do not move files** isolation method.

   > **Note**
   > In most cases, when you move a non-executable file, such as a help file, to an isolated or private directory, the application still uses the version in the shared directory. The advantage of using isolation is that the different versions of the file are saved in the isolated directories and will not be overwritten by other packages. You can add a shortcut to the application to point to the appropriate file location.

   ■ To use the most recent version of the file for the active package, click Latest. If one file has a newer version but another has a newer date/time, the Latest File Selection dialog box appears, where you specify whether to use the file with the newest modified date or the highest internal version number.

   ■ To change the file's component, click Fix Comp.

      ♦ If the file is a KeyPath to a component in one package but not in another, the file becomes a KeyPath to its own component.

      ♦ If the file has extra non-advertising resources, they are moved to a new component.

      ♦ If the file's shared .DLL counter is not set, it is set.

      If the component has more than one of these issues, a Fix Options dialog box appears, where you select which actions to take.

   ■ To apply a file in the upper list box to a package in the lower list box, select a file in each list and click Copy Down. The file remains in its current location but will be used to install the package in the lower list box.

   ■ To apply a file in the lower list box to the active package, select a file in each list and click Copy Up. The file remains in its current location but will be used to install the package in the upper list box.

   ■ To undo changes, click the file and click the upper or lower Revert button, depending on whether the file is in the upper or lower list box. This resets the file to the state it was in when you imported the package.

7. Click Next on the File Conflicts page.

   The File Conflicts in Registry page appears. The upper list box contains files in the selected package that have conflicts and are listed in the registry. The lower list box contains the conflicting files in other packages.

8. In the upper list box, select one or more files and take one of the following actions.

   See Guidelines for Resolving File Conflicts on page 33.

When you resolve a conflict, the exclamation point to the left of the file name changes from red to white.

- To move the selected file to a private directory and change the file path in the package, click Move. This button is not available if you selected the **Do not move files** isolation method.

  Warning

  When conflicting files are listed in the registry, moving them to a private application directory can cause problems. ConflictManager updates the registry when a file is moved to a private application directory. This can cause other packages to follow the moved file and result in a further conflict.

- To use the most recent version of the file for the active package, click Latest. If one file has a newer version but another has a newer date/time, the Latest File Selection dialog box appears, where you choose whether to use the file with the newest modified date or the highest internal version number.

- To apply a file in the upper list box to a package in the lower list box, select a file in each list and click Copy Down. The file remains in its current location but will be used to install the package in the lower list box.

- To apply a file in the lower list box to the active package, select a file in each list and click Copy Up. The file remains in its current location but will be used to install the package in the upper list box.

- To open a separate dialog box listing the registry keys that contain the path to the selected file, click Reg Details.

- To undo changes, click the file and click the upper or lower Revert button, depending on whether the file is in the upper or lower list box. This resets the file to the state it was in when you imported the package.

9. When you have resolved all file conflicts, click Finish on the File Conflicts in Registry page.

The Resolve wizard makes the changes you specified and closes.

Resolving Registry Conflicts

The Resolve with Rules and Resolve commands resolve file conflicts only. To resolve registry key conflicts, use the Properties dialog box.

Warning

Resolve registry key conflicts only if you are the developer of the package and you have extensive knowledge of how Windows operates. Changing registry information incorrectly can damage the installation. In particular, changing information in HKEY_CLASSES_ROOT can damage the operating system.

To resolve registry conflicts

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. If you are using package groups, select a group name in the Groups pane.

   The Applications/Packages pane displays only packages that are assigned to that group.

3. In the Applications/Packages pane, select a package.

    The package's conflicts appear in the Conflict List and are grouped by type.

    See Resource Conflicts on page 15.

4. If you do not see the conflicts you expected, select Conflicts menu > Filter and make sure the **Errors** and **Registry Conflicts** check boxes are marked.

    See Filtering the Conflict Display on page 31.

    Note
    The settings in Setup menu > Conflict Settings > Types determine which conflicts are detected. Example: If your conflict settings are set to detect file conflicts only, then registry conflicts do not appear even when the conflict filter is set to display registry conflicts.

    ---

5. In the Conflict List, select a registry item.

    The conflicts for the item appear in the Conflict Details pane.

6. In the Conflict Details pane, double-click the package containing the registry value to use.

    The Properties dialog box appears.

7. On the Registry Key Settings tab:

    a. Click in the **Data Value** field.

    b. Copy the value to the clipboard.

    c. Click Cancel.

8. In the Conflict Details pane, double-click the package containing the registry conflict to resolve.

    The Properties dialog box appears.

9. In the Registry Key Settings tab,

    a. Select the contents of the **Data Value** field.

    b. Paste the value you copied from the other registry entry, overwriting the existing value.

    c. Click OK.

In a few seconds, the Conflict Details pane is refreshed.

Resolving INI Conflicts

The Resolve with Rules and Resolve commands resolve file conflicts only. To resolve .INI file conflicts, use the Properties dialog box.

Warning
Resolve .INI file conflicts only if you have extensive knowledge of how Windows operates and you understand how changing an .INI file affects the installation.

To resolve .INI conflicts

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. If you are using package groups, select a group name in the Groups pane.

   The Applications/Packages pane displays only packages that are assigned to that group.

3. In the Applications/Packages pane, select a package.

   The package's conflicts appear in the Conflict List and are grouped by type.

   See Resource Conflicts on page 15.

4. If you do not see the conflicts you expected, select Conflicts menu > Filter and make sure the **Errors** and **Other Conflicts** check boxes are marked.

   See Filtering the Conflict Display on page 31.

   Note
   The settings in Setup menu > Conflict Settings > Types determine which conflicts are detected. Example: If your conflict settings are set to detect file conflicts only, then .INI conflicts do not appear even when the conflict filter is set to display .INI conflicts.

   ---

5. In the Conflict List, select an .INI item.

   The conflicts for the item appear in the Conflict Details pane.

6. In the Conflict Details pane, double-click the package containing the .INI value to use.

   The Properties dialog box appears.

7. On the Edit INI File Settings tab:

   a. In the **INI File Contents** section, click in the Entry column.

   b. Copy the value to the clipboard.

   c. Click Cancel.

8. In the Conflict Details pane, double-click the package containing the .INI conflict to resolve.

   The Properties dialog box appears.

9. In the Edit INI File Settings tab,

   a. In the **INI File Contents** section, click in the Entry column.

   b. Paste the value you copied from the other .INI file, overwriting the existing value.

   c. Click OK.

In a few seconds, the Conflict Details pane is refreshed.

Restrictions on Resolving and Exporting Package Types

You cannot resolve conflicts in or export certain types of packages.

| Device drivers | Device drivers typically are provided by a vendor and should not be changed. Instead, change the package that conflicts with a device driver. |

Group Policy Objects	Conflicts between packages and Group Policy Objects are not typical conflicts that you can resolve by changing files or registry keys, rather, they are provided to inform you of packages that might override your group policies.
Merge modules	Changing a merge module affects all packages that use it. Instead, consider adding the merge module to the package it conflicts with.
Microsoft hotfix	Hotfixes typically are signed and cannot be changed. Alternative: Create a WiseScript wrapper that installs the hotfix and then performs a post-installation operation to install a changed source file.
Transforms	If the transform was applied to a base package when it was imported, you can resolve conflicts in and export the transform package. If the transform was applied to a blank Windows Installer database, you cannot resolve conflicts in or export the transform package.
Patches	Patch files contain binary differences instead of full files. Alternatives:
	• Apply the patch during import of the base package. Then the patch's resources are added to the resources of the base package and you can resolve conflicts for the base package.
	• Create a WiseScript wrapper that installs the patch and then performs a post-installation operation to install a changed source file.
SOE Snapshots	SOE Snapshot files are created solely to let you import a standard operating environment into the Software Manager database. You cannot compile an SOE snapshot file into an executable installation. Instead, change the package that conflicts with an SOE Snapshot.
Subscribed packages that you copied from another database by package subscription	You cannot change the source package files. Instead, change the packages in your database.
InstallShield® Developer installations that you imported from an InstallShield executable	Neither the original .EXE nor its embedded .MSIs can be edited.
Non-Windows Installer or non-WiseScript installations that you imported with the Universal Import option	The original installation file cannot be edited.

Package Export After Conflict Resolution

After you resolve conflicts, you must export the changes to the original Windows Installer or WiseScript installation to produce an installation that does not conflict with other packages on your organization's computers.

You have the following options for exporting packages:

Export	Export changes for a single package. After you export the package you must compile it in Windows Installer Editor or WiseScript Editor.
	See Exporting a Single Package on page 44.

| Export and recompile | Export changes for all packages whose conflicts have been resolved. You also can have the installations recompiled automatically after exporting. |
| | See Exporting and Recompiling Packages on page 45. |

In order for you to export a package:

● The original package installation must exist. The export process cannot create a new installation.

● The source files in the original installation and any files you added during conflict resolution must have valid source paths.

● The **Prevent Package Export** check box on the Package Attributes dialog box must not be marked.

You cannot export:

● Packages that are marked as read-only on the Package Attributes dialog box in Software Manager.

● Certain types of packages.
  See Restrictions on Resolving and Exporting Package Types on page 42.

Exporting a Single Package

After you resolve conflicts for a package, you can export the changes to the original Windows Installer or WiseScript installation, which you then recompile in Windows Installer Editor or WiseScript Editor.

To export a single package

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. Select Packages menu > Export.

   The Export Package dialog box appears.

3. Complete the dialog box:

   ■ **Application Name**
     Select the application to export.

   ■ **Package Name**
     Select the package to export.

   ■ **Installation File**
     Specify the path and file name of the Windows Installer installation (.WSI or .MSI) or WiseScript (.WSE) to export to. Normally, you should accept the default, which is the file that was originally imported into Software Manager. If you are exporting a WiseScript, you can export to a different file. You must export to the same type of file that was originally imported.

   The remaining options are different depending on the type of file you are exporting.

4. If you are exporting a WiseScript, select one of the following:

- **Overwrite existing installation**
  Replaces the entire package installation with information in the Software Manager database.

- **Merge changes with existing installation**
  Adds only the conflict resolution changes to the existing installation.

5. If you are exporting a Windows Installer package that you previously distributed to end users, select an option in **Upgrade Type** to change the installation so that it can function as an update.

    - **None (Do not modify)**
      Select this option if you have never distributed this package to end users. It adds only the conflict resolution changes to the existing installation.

    - **Small (Change PackageCode)**
      Select this option to export the package as a patch or reinstall. This option creates an installation containing conflict resolution changes plus a new package code, which will reinstall over the previous version to update it. The version of the installed package will not be updated. To create a patch between the package you are exporting and the previous version, use the Patch Creation tool.

    - **Minor (Change Version and PackageCode)**
      Select this option to ship this version of the package as a patch or reinstall. This option creates a package installation containing conflict resolution changes plus a new package code and version, which will reinstall over the previous version to update it. To create a patch between the package you are exporting and the previous version, use the Patch Creation tool.

    - **Major (Change ProductCode, Version and PackageCode)**
      Select this option to ship this version of the package as an upgrade. This option creates a package installation containing an upgrade entry, which will upgrade previous versions.

6. If you are exporting an .MSI and you do not want to change the original package installation, specify a transform file (.MST) in **Transform File**. The conflict resolution changes are saved in the transform file, which you can apply to the original package installation.

7. Click OK.

    The changes are exported to the original package installation.

8. If you exported a .WSI or .WSE, open the package in Windows Installer Editor or WiseScript Editor and compile it. You do not have to compile .MSI files.

Exporting and Recompiling Packages

After you resolve conflicts for multiple packages, you can export the changes to the original Windows Installer or WiseScript installations. You also can have the installations recompiled automatically after exporting. This is useful when you resolve conflicts using the Resolve with Rules command, because you don't have to keep track of which packages have changed, and you don't have to recompile them individually.

This process always overwrites the original installation files.

To export and recompile a package

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. Select Packages menu > Export and Recompile.

   The Export Changed Applications and Recompile dialog box appears. This lists all packages with conflict resolution changes that have not been exported. All check boxes are marked by default.

3. Clear the check boxes for packages you do not want to export.

4. To recompile the packages automatically after exporting them, mark **Compile the packages after export**.

5. Click OK.

The selected packages are exported to their original installation files. If you marked **Compile the packages after export**, they are recompiled.

ConflictManager Reports

ConflictManager lets you generate the following reports that provide information about the packages in the Software Manager database.

Note
When you install Wise Package Studio, triggers are added to the Software Manager database. Because the default SQL Server backup process does not include triggers, you should change your backup configuration to include them.

File Conflicts by Package	Lists all file conflicts between selected packages and the packages of all other applications. The conflicts are grouped by application and package name and arranged in each group by file name.
File Conflicts By Group	Lists all file conflicts for a group of packages. The conflicts are grouped by application and package name and arranged in each group by file name.
File Conflicts By Group	Lists all packages that have conflicts with the file you specified. The conflicts are grouped by file name and arranged in each group by conflict level and application and package name.
Registry Conflicts By Package	Lists all registry conflicts between selected packages the packages of all other applications. The conflicts are grouped by application and package name and arranged in each group by registry key.
Registry Conflicts By Group	Lists all registry conflicts for a group of packages. The conflicts are grouped by application and package name and arranged in each group by registry key.

| Registry Conflicts By Key | Lists all registry conflicts by registry key. The conflicts are grouped by registry key and arranged in each group by conflict level and application and package name. |

Generating a ConflictManager Report

You can generate a ConflictManager report that displays in a report viewer window, from which you can save or print it. You can also generate a report directly to a file.

See Saving a ConflictManager Report Directly to a File on page 48.

To generate a ConflictManager report

1. (Enterprise Management Server only.) If multiple databases are open, select a database in the Applications/Packages pane.

2. From the Reports menu, select a report.

    The Report Parameters dialog box appears. The fields that appear on this dialog box vary depending on the type of report that was selected.

3. For predefined reports, complete the Report Parameters dialog box:

    ■ **Entries Per Page**
    Enter the number of conflicts to display per page. (Example: If you enter 1, each page of the report displays a single entry, therefore, a report with 10 entries results in 10 pages.) The lower the number, the faster the report generates. The default value is 6000.

    > **Note**
    > If you save a report as an HTML file, an HTML file is created for each page of the report.

    ■ **Conflict Types to Display**
    Errors are conflicts that deserve attention. Warnings are conflicts that probably will not cause a problem.

    ■ **Application and Package Name**
    (Conflicts by Package only.) Specify the applications and packages on which to run reports. SQL wildcards are accepted. If you select an application from **Application Name**, only its packages appear in **Package Name**.

    ■ **Group Name**
    (Conflicts by Group only.)

    ■ **File Name**
    (File Conflicts by File only.) SQL wildcards are accepted.

    ■ **Key Path and Value Name**
    (Registry Conflicts by Key only.) SQL wildcards are accepted.

4. Click OK.

    The report opens in the report viewer window.

    See Using the Report Viewer on page 48.

Saving a ConflictManager Report Directly to a File

You can generate and save a ConflictManager report to a file without opening it in the report viewer. You might do this if you want to generate a large report overnight. This involves creating an .INI file that contains the instructions for generating the report and then using a command line to generate the report.

To create the .INI file

1. Open the ReportConfig.ini file located in the share point Reports directory.

2. Read the comments that precede each variable and determine what values you need to enter to generate your report.

   Each line of comments begins with a semicolon.

3. Enter the values for the appropriate variables.

   Some variables apply only to specific reports.

4. Save the .INI file with a name that clearly identifies it.

To generate the report from a command line

Use the following command line:

"*Wise Package Studio installation directory*\Workbench\IMReport.exe" /infofile="*path to ReportConfig.ini file*"

When you run the command line, the report is saved in the directory you specify in the .INI file. A report.log is also created in the same directory. If you generate multiple reports, save them in different directories to prevent the report.log from being overwritten.

Using the Report Viewer

After you generate a report, it is displayed in a report viewer window.

- To save a report, click Save As in the lower right of the report viewer window. You can save a report in HTML, XML, or CSV format. When you save a report as HTML, an HTML file is created for each page of the report, and each file has the same name with its page number appended.

- To print a report, click Print in the lower right of the report viewer window.

- The date and time when the report was generated appears in the lower left corner of the report.

- To perform a text search, click on the report and press Ctrl+F.

- The first column in each table displays one or more letters that represent the conflict level. To access an explanation of the conflict levels, click the conflict letter. This displays a **Conflict Levels** legend in the lower right of the report.

- If there are no conflicts, the report viewer opens and **No Conflicts** appears in the middle of the report.

- At the top of the report, you can see the number of **Packages in the Software Manager database** and the number of **Packages on which conflicts were run**.

Interpreting the Conflict Level

When you view a ConflictManager report in the report viewer, you can easily access a legend that explains the conflict level.

See

If you save a report as a .CSV file, the conflict level appears as a number at the beginning of each conflict. The conflict level can have one of the following values.

1	Informational, not a true conflict and not likely to cause a problem
2	Warning, a possible conflict
4	File error, deserves attention
8	Component error, deserves attention

A file can have a combination of conflicts with different applications. In that case, the conflict level has one of the following values.

3	informational + warning (1+2)
5	file error + informational (4 + 1)
6	file error + warning (4 + 2)
9	informational + component error (1 + 8)
10	component error + warning (8 + 2)
12	component error + file error (8 + 4)

Index