



AonixADT for Eclipse

Release/Installation Notes

Version 3.2.3, Build 144 (Windows)

Aonix World Headquarters

5930 Cornerstone Court West, Suite 250

San Diego, CA 92121

Tel: (858) 457-2700

Fax: (858) 824-0212

For the addresses and phone numbers of Aonix distributors and support organizations worldwide, please refer to the Aonix homepage, www.aonix.com.

Copyright © 2009 Aonix®. All rights reserved. No part of this document may be reproduced in any form or by any means without permission in writing from Aonix. Aonix reserves the right to make changes in specifications and other information contained in this publication without prior notice. Consult Aonix to determine whether such changes have been made.

Aonix®, ObjectAda®, AdaWorld®, TeleUSE®, and PERC® are registered trademarks of Aonix. AonixADT™ and RAVEN™ are trademarks of Aonix. All other company and product names are the trademarks of their respective companies.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the rights in Technical Data and Computer Software clause at DFAR 252.227-7013, or FAR 52.227-14, ALT III and/or FAR 52.227-19 as set forth in the applicable Government Contract.

Aonix, 5930 Cornerstone Court West, Suite 250, San Diego, CA 92121 (Contractor)

Table of Contents

Product description	5
New features	6
Important changes	8
Software compatibility	9
Data compatibility	9
Documentation	9
Installation	10
Licensing	17
Useful information	17
Product limitations	19
Fixed problems.....	21
Improvements	27

Release/Installation Notes

This document applies to the following AonixADT versions:

- Version 3.2.3, Build 144 (Wind River Workbench/Windows)

Note: Build 144 is intended to be installed on Wind River Workbench 3.0. When you use ADT Build 144 within Wind River Workbench, you have the option of using either the Wind River debugger or the ADT debugger for debugging. For more information, see the “Running and debugging projects: VxWorks” topic in the online documentation (**Help > Help Contents > Ada Development User Guide > Tasks**).

- Version 3.2.3, Build 143 (Windows) [This build is compatible with Wind River Workbench 2.6]
- Version 3.2.2, Build 141 (Solaris)
- Version 3.2.2, Build 140 (Windows)
- Version 3.2.2, Build 137 (Solaris)
- Version 3.2.1, Build 134 (Solaris)
- Version 3.2.1, Build 133 (Windows)
- Version 3.1.1.1, Build 128 (Linux)
- Version 3.1.1.1, Build 129 (Solaris)

These release notes describe the features, enhancements, dependencies, limitations, and installation requirements of the latest release. Unless otherwise noted, when a correction or enhancement appears in a certain version of AonixADT, all later versions of AonixADT will include that change regardless of the platform type where the change first appeared (Windows, Linux, or Solaris).

Note: The name of the product will be shortened to “ADT” in the remainder of this document.

Product description

This product release consists of the components listed below.

-
- Syntactically aware Ada editor that allows for:
 - Semantic code completion
 - Semantic browsing
 - Syntax coloring
 - Automatic block completion
 - Matching Ada delimiter displaying
 - Automated comment control
 - Ada development perspective
 - Ada toolchain management
 - Ada project navigator view
 - Ada build configurations view
 - Integrated program debugger supporting attaching to running processes
 - Customizable Ada source reformatters
 - Support for standard Eclipse features such as configuration control, software updates, multilanguage development, error navigation, text searching, tasks, bookmarks, etc.
 - Support for the ObjectAda and GNAT Ada compilers and debuggers

New features

The following new features have been implemented since the previous release of the product:

Release 323b144 Windows

- Support for Wind River Workbench 3.0 and ObjectAda Real-Time x PowerPC/VxWorks 6.6.

Release 323b143 Windows

- Support for ObjectAda Real-Time x PowerPC/VxWorks 653 (Raven).
- Addition of the *Ada Development User Guide* to the Workbench interface. The document, a set of linked HTML files, is accessed from the Workbench by selecting **Help > Help Contents**.

Release 322b140 Windows

- None

Release 322b137 Solaris

- None

Release 321b134 Solaris

- Support for ObjectAda Real-Time x PowerPC/Raven (Solaris)

Release 321b133 Windows

- Support for ObjectAda Real-Time x PowerPC/Raven (Windows)

Release 311b129 Solaris

- First Integrated Solaris version (GNAT toolchain support only)

Release 311b128 Linux

- First Integrated Linux version

Release 311b127 Windows

- Added **Ada Debug > Disassembly View**.

The **Step Into Instruction**, **Step Over Instruction**, and **Run to Cursor** actions are accessible via the icons located in the upper right corner of the view.

Note: Patch 1102V82-U4 is required in order for the disassembly view to function with ObjectAda 8.2.

- The variables view now shows package level variables.
- Added a "log variable" action to the editor context menu for capturing the value of variables to the debug console.
- Ada editor text hover displays the value of a variable being debugged.
- Added a text hover to display the parameter list when using code assist.

Release 311b122 Windows

- Integration of the GNAT toolchain
- User-configurable template wizards for creating Ada procedures, functions, packages, and files
- Ada toolchain management functionality
- Ada Configurations view
- Ability to attach the debugger to a running application
- Ability to set a breakpoint on an Ada exception (ObjectAda only)
- Support for breakpoint actions (variable logging)
- Support for remote updating via the Update Manager

-
- Improved support for Eclipse tasks and bookmarks
 - Support for hyperlink navigation to Ada declarations
 - Ability to open Ada Navigator files from the editor

Important changes

The following important changes (other than features noted in the section above) have occurred since the first release.

Release 322b140 Windows

- The restrictions on GNAT projects (<35 units and filesize < 2000 lines) have been removed. The GNAT Professional Add-On is no longer required for full functionality with GNAT.

Release 322b137 Solaris

- The restrictions on GNAT projects (<35 units and filesize < 2000 lines) have been removed. The GNAT Professional Add-On is no longer required for full functionality with GNAT.

Release 321b133 Windows

- Parameters can now be passed when building GNAT standard projects.
- Users can now choose between invoking `gnatmake` and `gprmake` when building GNAT standard projects.

Release 311b127 Windows

- Reference searches (also known as "Where Used") make use of ASIS functionality called from an executable program named "adaquery". Since the format of ASIS information is compiler dependent, and ADT is intended to be used with a variety of different compilers, `adaquery` should be rebuilt and installed as part of installing ADT. See the installation instructions for more information. note- This applies only to GNAT. Compatible versions of `adaquery` are delivered with ObjectAda compilers.

Release 311b122

- ADT includes support for GNAT and ObjectAda toolchains in the same product. For ADT for Linux, projects created with the GNAT toolchain are limited in size to 35 units. Files created with the GNAT toolchain are limited in size to 2000 lines of text. The GNAT Professional Add-On for AonixADT can be obtained from Aonix to remove these limitations. For Solaris and Windows the GNAT restriction no longer exists.
- Code Assist performance has been substantially improved.

- Reference searches have been temporarily disabled (GNAT only)

Note: Also refer to [“Fixed problems” on page 21](#) and [“Improvements” on page 27](#) for more information on changes occurring in this release.

Software compatibility

- Compatiibilities relating to the host operating system: Refer to the host operating system compatibility requirements of your version of Eclipse.
- Eclipse SDK 3.1 (or greater) or Wind River Workbench 2.6 or greater
- ObjectAda Version 8.2 or greater
- Java 1.5 or greater
- GNATPRO 5.03a or greater or equivalent open source builds.

A version of `gdb` that supports the `gdb` MI protocol is needed to use the ADT debugger with GNAT. To determine if your `gdb` supports MI, run the following command:

```
gdb -interpreter=mi
```

If the `gdb` command prompt appears, your version of `gdb` supports MI.

Some ADT functionality may operate only when the appropriate GNATPRO tool (such as `gnatpp`) is installed.

Data compatibility

Projects created with Version 2 of ADT will work with ADT Version 3.

Documentation

PDF documentation provided with this product is listed below.

- The release notes you are reading now
- A “quick start guide” for users of ADT

In addition, a set of HTML files constituting the *Ada Development User Guide* is accessible from the Eclipse Workbench (**Help > Help Contents**).

Installation

To use the ADT Eclipse IDE interface, you need to have Eclipse and your Ada development environment installed and properly configured. To accomplish this, install the components below as required for running ObjectAda or GNAT under the Eclipse IDE.

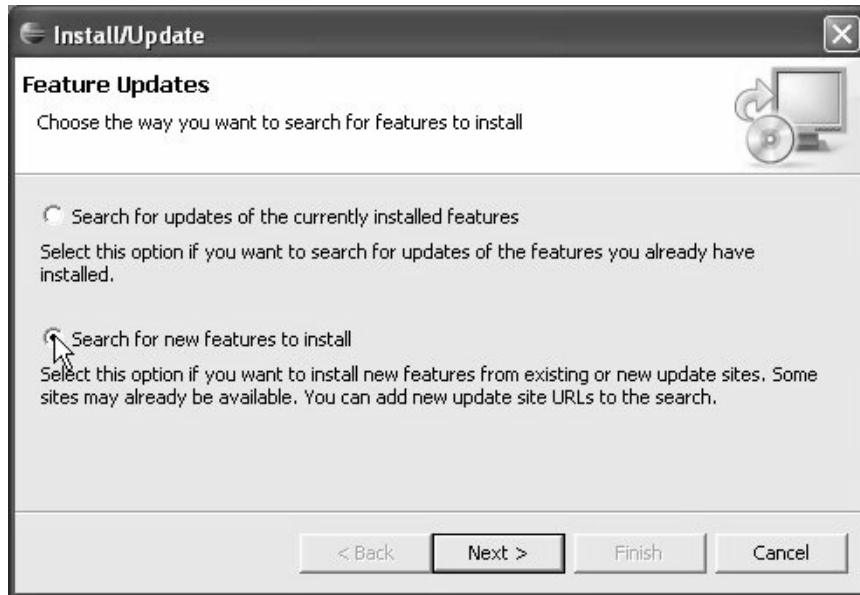
Note about installing on Windows Vista: You may encounter problems installing ADT on Windows Vista that will require you to modify Eclipse to run in Windows XP Service Pack 2. Please refer to the “Installing AonixADT components” in your ObjectAda for Windows *Release/Installation Notes* for more information.

- Eclipse SDK 3.1 or greater for the appropriate platform. It is available from the following URL:
<http://www.eclipse.org/downloads>
Install the Eclipse SDK to a directory of your choice.
- Java Runtime Environment (JRE, version 1.5 or greater) for the appropriate platform. JRE may be downloaded from the following URL:
<http://java.sun.com/>
- At least one Ada development environment:
 - ObjectAda v8.2 or greater
 - GNATPRO 5.03a or greater, or an equivalent open source GNAT build
- The ADT plug-in. To install the ADT plug-in, use the Eclipse site update functionality as described below.

Important note: You will be installing ADT into the `eclipse` directory. In order to do this, you must have permission to write into that directory. If write permission is not available, and you try to install, you will not get an error saying that you need write permission. Instead you may get another, non-intuitive error.

To install the ADT plug-in:

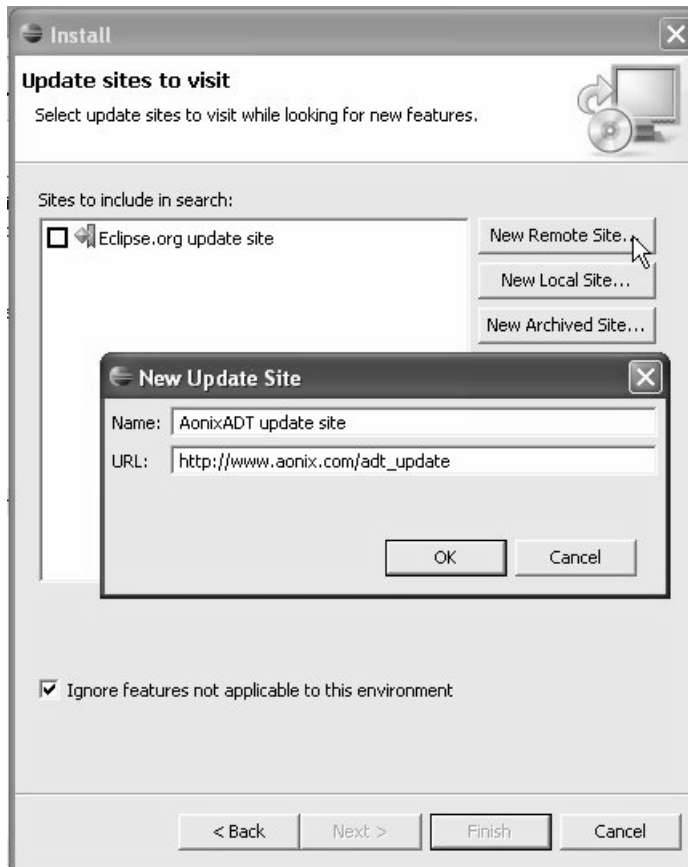
1. Launch Eclipse. The Eclipse executable is located at
`<Eclipse_Installation_Dir>/eclipse/eclipse[.exe]`
2. If this is your first time starting Eclipse, you will be prompted for the name of a workspace. Choose one or use the default name provided.
3. Choose **Help > Software Updates > Find and Install**.



4. Select **Search for new features to install** and press **Next**.
5. Then proceed as for a remote installation or a local installation.
 - a. *For a remote installation:*

In order to use the remote installation functionality of the Eclipse Update Manager, you must have a direct `http` connection to the internet. If you don't have access to the internet, or wish to install ADT from a local source, skip this step and proceed to [“For a local installation:”](#) on the next page (Step 5b).

- Select **New Remote Site**.
- Enter a name (e.g., “AonixADT update site”) and the correct URL (`www.aonix.com/adt_update`) as shown below and press **OK**.



- The update site should now be included in the search list of the site displayed in the installation window. Proceed to Step 6 on [page 14](#).

Note: The Eclipse Update Manager will automatically install the latest available version of ADT.

b. For a local installation:

ADT may be installed either directly from the Aonix website (remote updating) or from a local source (CD-ROM, hard drive, USB key, etc.). You may install ADT directly from the AonixADT CD-ROM. If desired, you may copy the contents of the CD-ROM to another file system (hard drive, USB key, etc.) and install from there. In addition, ADT is available as a compressed file on the Aonix website.

This is useful for installing on machines that may not have direct access to the internet. When using the compressed file, uncompress the file to a file system (`untar` it for UNIX versions of ADT) before following the instructions below.

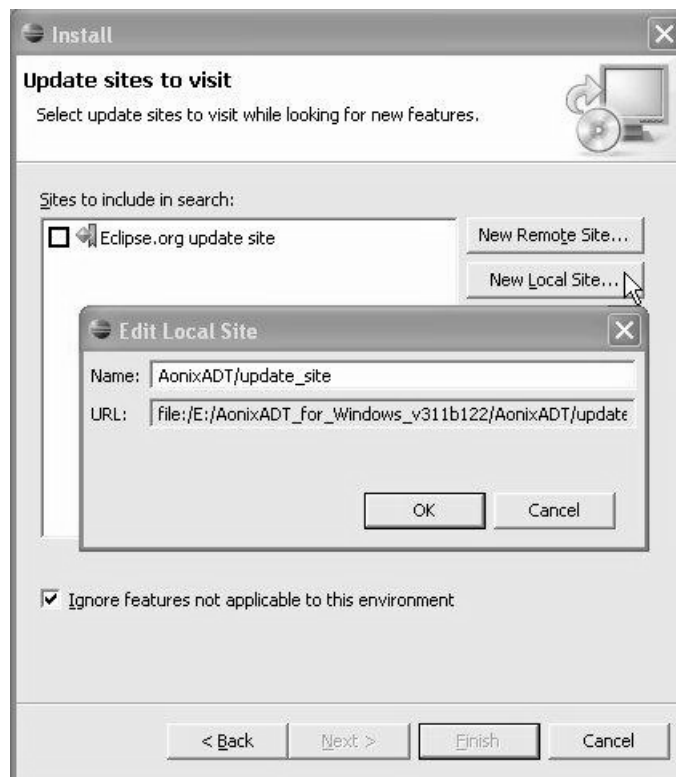
- Select **New Local Site**.

In the file system dialog that appears, navigate to the `update_site` subdirectory of your ADT installation. If you are installing directly from the ADT CD, the correct directory will look something like this:

```
E:/AonixADT_for_Windows_<version>/AonixADT/update_site
```

Make sure to choose the directory that corresponds to your operating system

- Press **OK** and select the above update site when prompted.

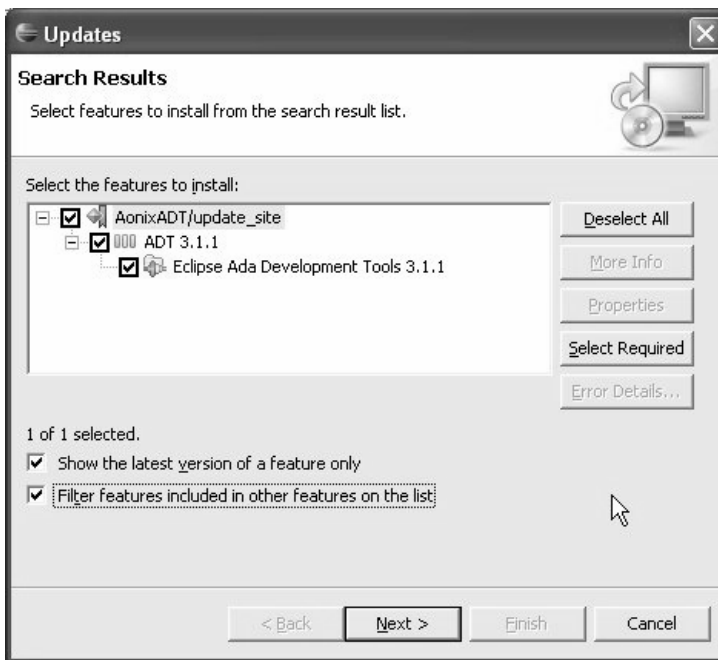


The update site should now be included in the search list of the site displayed in the installation window.

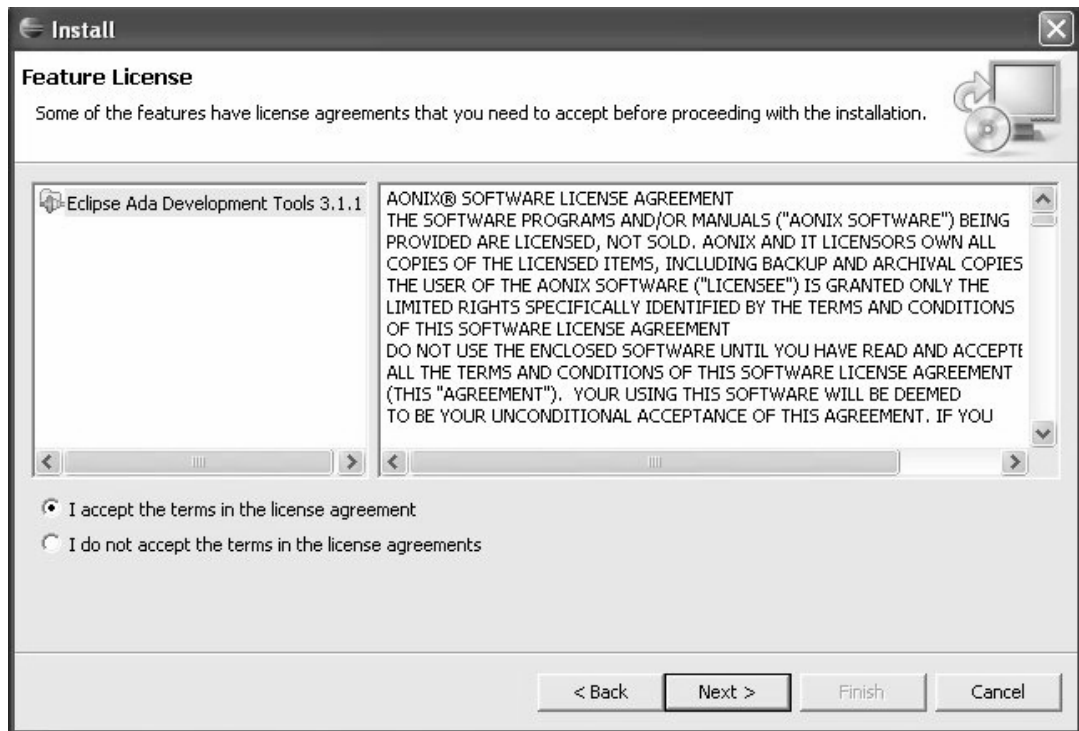
6. Press **Finish**. You will now be prompted with an Update Window dialog.

Note: In the following steps, the images will show the update site as “AonixADT/update_site”. (If you installed remotely, your update site may be indicated as “AonixADT update site”.)

7. Select the update site you have just created from the list. If this is the first update site you have created, the list will contain only the ADT update site. Should you need to install a previous version of ADT (remote update only), uncheck **Show the latest version of a feature only**. Verify that the checkbox next to the ADT update site is selected. You may expand the tree under the update site to be sure that all of the individual components of this site are selected. Press **Next**.



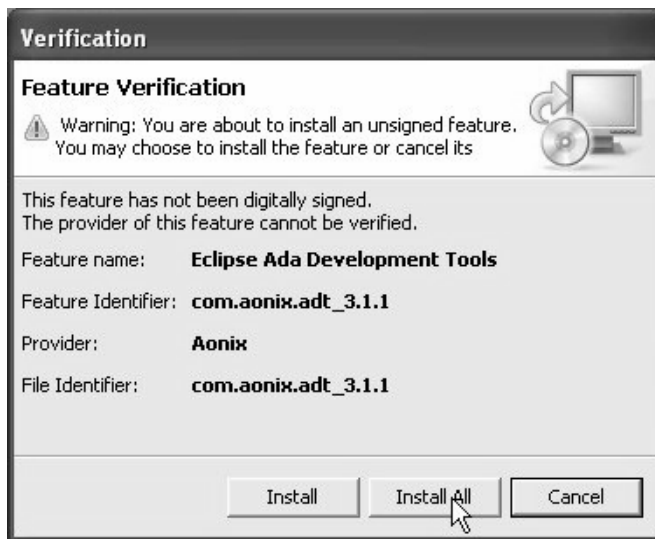
8. Read and accept the terms of the license agreement presented and press **Next**.



9. Choose the location for installing the ADT feature. This install location should be the directory where you have installed Eclipse (e.g., `D:\Eclipse3.1.1\eclipse`). Use the change location button to select the path to your Eclipse installation if necessary. Press **Finish**.



10. Press **Install All** and the ADT plug-in will be installed.



11. Restart the workbench when prompted.

12. (GNAT users only) Reference searches (also known as "Where Used") make use of ASIS functionality called from an executable program named "adaquery". Since the format of ASIS information is compiler dependent, and ADT is intended to be used with a variety of different compilers, `adaquery` must be rebuilt and installed as part of installing ADT.

Follow these steps to rebuild and install `adaquery`:

- a. Ensure that your GNAT environment is properly configured and the GNAT ASIS library is properly installed. The `adaquery` sources are located at
`<eclipse plugin directory>\com.aonix.adt.gnat.<platform_version>\AdaQuery\src\`

b. From the command line, create the `adaquery` executable using the command
`gnatmake`
`adaquery -aI<eclipse plugin directory> \com.aonix.adt.gnat.<platform_version>\AdaQuery`
`\src -I<GNAT ASIS installation directory> -largS -lasis`

which is typed on a single line. Here is an example of the command used to build `adaquery.exe` on Windows:

```
gnatmake
adaquery -aID:\Eclipse312\eclipse\plugins\com.aonix.adt.gnat.win32_3.1.1.1\AdaQuery
\src -Id:\gnatpro5.03a\asis -largS -lasis
```

- c. Copy the `adaquery` executable built in step b above to the directory
`<eclipse plugin directory>\com.aonix.adt.gnat.<platform_version>\`
- d. Make sure that the file permissions allow execution of `adaquery`.

Licensing

AonixADT does not use a license protection mechanism. The Ada development environment you choose may require its own licensing, however.

Useful information

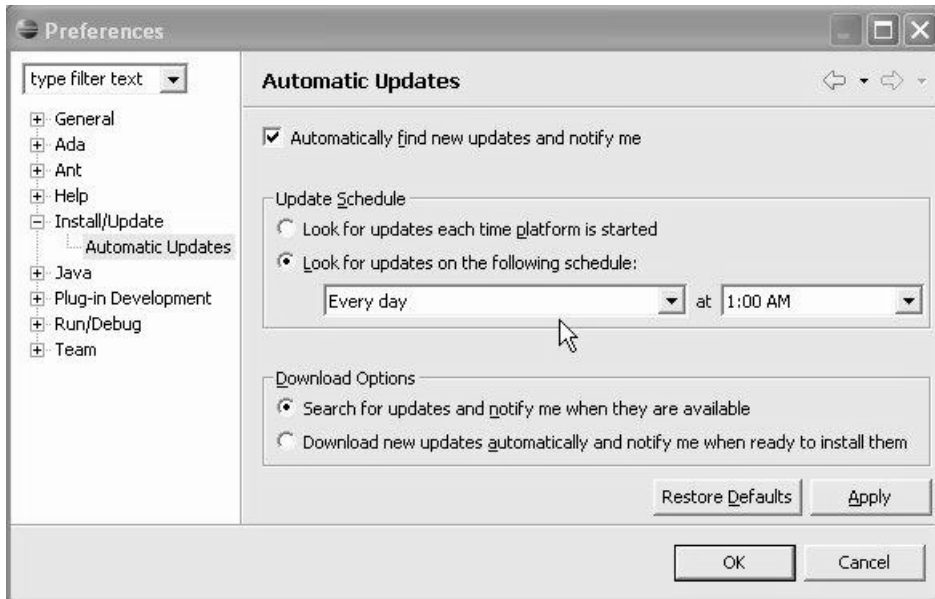
This section contains information that may not be covered elsewhere in the documentation.

Updating ADT

Once ADT has been successfully installed, you can automatically search for and install any available updates by selecting **Help > Software Updates > Find and Install > Search for updates of the currently installed features**. Even if you initially

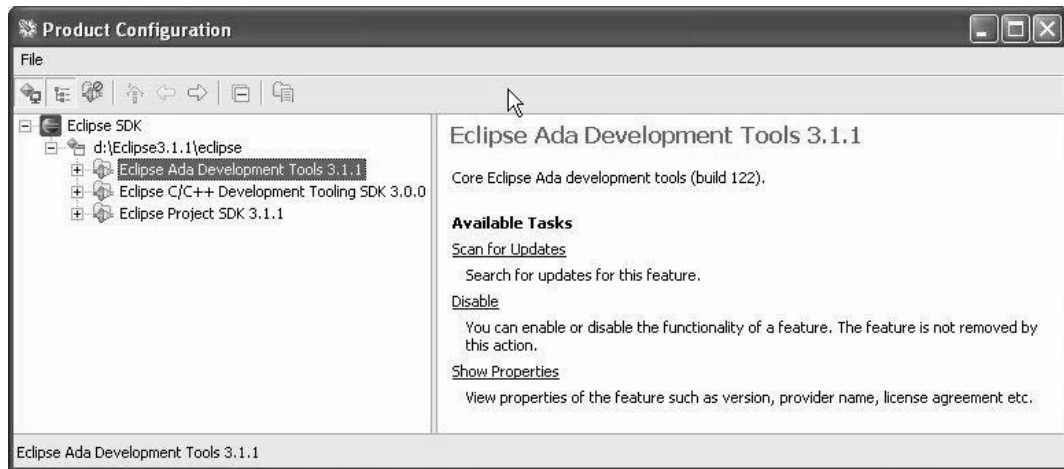
performed a local installation, Eclipse will know to look on the Aonix site for more recent versions of ADT. As mentioned above, you must have a direct http connection to the internet for this functionality to work.

Eclipse may be configured to automatically update AonixADT and other installed plugins via **Window > Preferences > Install/Update > Automatic Updates** as shown below.



You can also manually check the Aonix website for compressed versions of the latest ADT releases for download.

- ADT has been developed for ObjectAda version 8.2 and greater. If it is used with previous versions of ObjectAda, some or all of the ADT functionality may not operate properly.
- For ADT for Linux, projects created with the GNAT toolchain are limited in size to 35 units. Files created with the GNAT toolchain are limited in size to 2000 lines of text. The GNAT Professional Add-On for AonixADT can be obtained from Aonix to remove these limitations.
- To determine the precise version of ADT you have installed, open the Product Configuration window (**Help > Software Updates > Manage Configuration**). Select the Eclipse Ada Development Tools from the tree. The ADT version (e.g., Eclipse Ada Development Tools 3.1.1) and the build number (e.g., build 122) are indicated on the panel on the right (see screen below).



- The Configuration Management (CM) tool can also be used to show the ADT version installation history as well as to uninstall (remove) ADT or to revert to a previous version of ADT. See the Eclipse help for more information on this and other configuration actions.

Product limitations

- If you select *Specify Path* instead of *Use Installed Toolchain* when specifying a new configuration, code navigation and reference searches for Ada constructs in the ObjectAda libraries will fail. The workaround is to manually add the `rts` directory to the Search links in the Ada Build property page.
- During debugging, a “selection cannot be launched” message may appear when you attempt to run the debugger (**Run > Debug**) on a configuration more than once. The message indicates that Eclipse cannot determine how to run the current selection in the Eclipse window. The workaround is to use **Run > Debug History > <configuration>** or **Run > Open Debug Dialog**.
- ECR 12814
Display LRM functionality doesn't currently work on Solaris because of an Eclipse limitation
- ECR 12450
Source files are sometimes being unnecessarily reregistered (ObjectAda toolchains).

-
- ECR 12184
Enhancement request: Individual files should be selectable in the **Project > Properties > Adabuild > Source Folders** tab of an Ada Configuration.
 - ECR 12185
Enhancement request: Deleting a configuration does not delete the output directory.
 - ECR 12290
Executing a program that has an unhandled exception does not give a symbolic traceback under ADT (ObjectAda for Windows).
 - ECR 12357
Enhancement request: Add decoration to configuration icons to show which toolchain is being used.
 - ECR 12553
Linking GNAT projects beyond restriction limits can cause a `Java.lang.NullPointerException`
 - ECR 12563
Debugging GNAT projects can display an error message: "an internal error occurred during: fetching children from debug target".
 - ECR 12566
Some color options have no effect in the Ada editor preference page.
 - ECR 12567
There are usability problems with the **Libraries** tab in the **Properties > Ada Build** wizard page (GNAT toolchain).
 - ECR 12568
When the user changes the displayed tab width (Ada editor preferences), the value of the indent size (**Properties > Ada Pretty Printer**) should change too.
 - ECR 12573
Enhancement request: Add support in the editor for Ada0X.
 - ECR 12585
Creating an Ada Configuration may delete any existing directories with the same name.
 - ECR 12599
If "Show Type Names" is selected for the Variables display window, no type names are shown for the top level variables.
 - ECR 12602
Executing a step in debug view can cause an error if the focus is in Disassembly view.

- ECR 12622
Reference searches occasionally don't work.
Workaround: Manually save one of the source files in order to force generation of ASIS information.
- ECR 12638
The debugger may not run under Eclipse unless its location is added to the `PATH` environment variable.
- ECR 12639
Errors may occur when opening a run/debug configuration if a configuration exists for a closed project.
- ECR 12641
The process selection dialog for Attach to Process may show no processes.
- ECR 12642
The OK button for the process selection dialog may not enable properly.
- ECR 12665
ADT mishandles the `-o` argument to `gcc` (GNAT toolchain).
- ECR 12672
Add Expression (Ada) does not work on local variables (GNAT only)
Workaround: Make sure that the variable names entered are unqualified.
- ECR 12674
Enhancement request: Add the ability to indicate source directory search order.

Fixed problems

Version 3.2.3b144 (Windows)

- ECR 12149
Online help is now provided in HTML format with ADT.

Version 3.2.2b140 (Windows)

- ECR 12824
Fixed ordering of BSP components prevents the use of board-specific `start.o` or `ppcdefault.x` files.
- ECR 12824
Files with non-standard file extensions cannot be compiled.

Version 3.2.2b137 (Solaris)

-
- ECR 12810
Fixed missing labels in Ada build settings dialog found in version 3.2.1b134.

Version 3.2.1b133 (Windows)

- ECR 12569
Fixed problem where actions of the Search view did not work with Ada search.
- ECR 12657
NullPointerException no longer causes a popup error when starting Eclipse.
- ECR 12658
The main units section for GNAT project files are now correctly generated when multiple main units are selected.
- ECR 0012667
Selecting a `.gpr` file for a standard project will no longer erase the project if it already exists.

Version 3.1.1b127 (Windows)

- ECR 12343
The Declarations menu function does not work.
- ECR 12572
Enhancement request: Add support for disassembly of code.
- ECR 12621
Problems debugging GNAT programs which have command line arguments

Version 3.1.1b122 (Windows)

- Tasks and bookmarks are now supported in the Ada editor.
- Fixed Declaration search (was broken by unit management enhancement).
- Fixed problem with Reference and Declaration searches when invoked from an Ada editor displaying an external file.
- Fixed issue where the `UNIT.MAP` file for a configuration was incorrectly updated when the order of source links is changed. (ObjectAda only)
- Fixed potential hang when retrieving system processes while launching an "Attach to Process" debug session.

Version 3.1.0.9

- Fixed a problem with unit registration where the workspace could hang. (ObjectAda only)

- Fixed a problem using installed toolchains for build configurations where the build executables could not be found.

Version 3.1.0.8

- Fixed issue where custom environment variables were not used when debugging.

Version 3.1.0.6

- Fixed the problem with registration of Ada source files introduced in version 3.1.0.5. (ObjectAda only)

Version 3.1.0.5

- Fixed problem with modifying the project settings for a project under source control.

Version 3.1.0.4

- Active configurations are now persistently saved.

Version 3.1.0.3

- Fixed issue where the workbench would sometimes lock up when compiling tree files.

Version 3.1.0.2

- Fixed issue where the source file for a breakpoint hit would not be found if the source file path contained folders with capitalized names AND is a linked resource.

Version 3.1.0.1

- `gnaamp` pretty printer now correctly includes all source directories. (GNAAMP only)
- Modified the build output to be more readable.
- Fixed issue where all Ada files in a configuration would be re-registered if a file marker was deleted. (ObjectAda only)
- Fixed issue where the UI would be locked up while `adareg` was running. (ObjectAda only)
- Improved update performance of Ada indexes.
- Fixed issue where the source file for a breakpoint hit would not be found if the source file path contained folders with capitalized names.

Version ADT_P2_020

- [ECR 12342]
Improved performance and functionality of code completion and navigation.

Version ADT_P2_019

-
- Fixed a bug where pressing the delete key in the Ada Navigator did not always delete the selected file/folder/project.
 - Fixed a bug where the `gnat` pretty printer would not work when the project had a library dependency. (GNAT only)
 - The text for the **Link Main Units** action in the Ada Navigator is visible again.

Version ADT_P2_018:

- Fixed problem where `gnatpp` didn't work with custom naming schemes. `gnaampp` still does not work with custom naming schemes. (GNAT only)
- [ECR 12278]
Fixed a problem with debugging a program that requires arguments.
- Fixed a problem where the build output was not displayed in the console window until the build was complete when invoking the compile or link actions.

Version ADT_P2_017:

- Invoking the compile or link actions now by default prompts the user whether or not to save dirty files. The user can choose to always save dirty editors when building or never save dirty editors based on an Ada preference.
- Fixed an issue where the project directory could not be deselected as a source folder.
- Fixed double click functionality in the Ada editor.
- [ECR 12186]
A new build type "library" has been added for ObjectAda configurations to build linked libraries. (ObjectAda only)
- The **Source > Format** option in the main menu is now enabled when a valid element is selected in the Ada Navigator.
- Default options for ObjectAda configurations have been corrected. By default, a configuration is set up for debugging. (ObjectAda only)
- Procedure subunits are no longer considered eligible main units.
- [ECR 12203] Folders may no longer be dropped onto themselves in the Ada Navigator.

Version ADT_P2_016

- Debugging now works even if **Link Main Units** is performed without explicitly compiling all source files.
- Fixed issue where an error dialog is occasionally displayed while debugging.

- Initial Ada Application when creating a launch configuration now points to a valid executable path.
- [ECR 12207]
Capitalization error of executables is now fixed.

Version ADT_P2_015

- Fixed issue with changing colors for syntax highlighting in the Ada editor.

Version ADT_P2_014

- Added use of the variable `$(Project_Root)` to generically refer to the user-specific location of a project within project properties.
- [ECR 12158]
Made improvements to the smart indentation algorithm.
- The cursor now remains on the same line number following an invocation of the pretty printer.
- Block completion has been improved to more intelligently determine when an end statement should be added.
- Undo of a block completion now only requires a single undo.

Version ADT_P2_013

- Eclipse no longer runs out of file handles.
- Fixed problem related to invoking some versions `gnatpp`. (GNAT only)

Version ADT_P2_012

- When creating a configuration, the toolchain and build type combo boxes are initially set to the first item in the list.
- Default dot replacement value is now a dash ("-").
- User is now prompted before the `.adaproject` file is checked out by the CM tool.

Version ADT_P2_011

- Removed dependence on the `.adaspec` directory, which was causing numerous version of the `.project` file to be created in ClearCase.
- Fixed a potential conflict with versions of GNAT other than 5.03 existing in the system path, causing code completion and code navigation to fail. (GNAT only)
- Fixed execution of the GNAT pretty printer to display exactly the text produced by `gnatpp`. (GNAT only)
- Fixed the issue regarding typing a '\$' character in the Ada editor.

-
- [ECR 12155] [ECR 12206]
Corrected the description for the Ada Managed Project wizard.
 - Corrected the name of the Ada problems marker in the filter for the Problems view.

Version ADT_P2_010

- Removed the `.adaspec` linked directory from Ada projects due to problems interacting with ClearCase. As a result, code completion for core Ada language libraries will not be available until an alternative is found.

Version ADT_P2_009

- Corrected a problem handling tab characters according to user preferences.
- Added a preference to turn on/off automatic dot completion.

Version ADT_P2_008

- Corrected a problem with the ObjectAda toolchain where the Ada libraries for GNAT were being included in ObjectAda builds. (ObjectAda Only)
- Corrected a problem where CM tools were not being consulted before modifying files.
- Fixed a bug in copying build settings from existing configurations where the `gpr` file is misnamed.
- Corrected a problem where a toolchain installed under directories containing spaces was not being properly called.

Version ADT_P2_007

- There are now no requirements on the system path for code completion and Ada searches or navigation.
- Improved performance when code completion or Ada search first requested after Eclipse started or was saved.

Version ADT_P2_005

- Fixed a problem with applying completions for child packages. The entire name for the completion was being inserted so that if the user typed "Ada." and selected "Text_IO" from the completion list, the result was "Ada.Ada.Text_IO".

Version ADT_P2_004

- Fixed an issue with the initial creation of tree files for a new project.
- Fixed a problem where the build path for some projects included the `.adaspec` linked directory, causing Eclipse to incorrectly include the Ada core specification files in the build.

Version ADT_P2_003

- Fixed an issue with the format of toolchain path parameters to `gnatmake` (and `gnaampmake`) when the toolchain path contains spaces.

Version ADT_P2_001

- Fixed incorrect syntax in the `gnat` project file for source directories. (GNAT only)

Improvements

Version 3.2.1b133 (Windows)

- ECR 12668
Enhancement: Allow specification of the `.gpr` file when creating GNAT standard projects.
- ECR 12669
Enhancement: Allow users to choose `gprmake` instead of `gnatmake`.
- ECR 12670
Enhancement: Allow parameters to `gnatmake` for standard projects.
- ECR 12671
Enhancement: Add accelerators to the Ada Navigator.

Version 3.1.1.1b127 (Windows)

- Added Disassembly functionality in the debugger.
- Ada editor text hover displays the value of a variable being debugged.

Version 3.1.1b122 (Windows)

- Internal restructuring

Version 3.1.0.9

- Added a registration mechanism to GNAT toolchain. Functionality is limited until toolchain is registered. (GNAT only)
- Added actions to the Ada Navigator view to change the source directories for the active configuration.
- Added New Ada unit wizards for packages, procedures, and function units.
- Added the ability to specify templates for new Ada files created using the new unit wizards.
- Added the ability to import/export the list of installed toolchains.

-
- Added an **Actions** menu item to the Breakpoints view's context menu. This can be used to specify variables whose values should be logged to the Debug Console when the breakpoint is hit. The user may also specify that the target be automatically resumed when the actions are complete.
 - Added expand/collapse actions to the Variables and Expressions views' toolbar and context menus.
 - Open Spec Declaration can now be invoked in the Ada Editor by CTRL+Click selecting the Ada element.
 - Open Body Declaration can now be invoked in the Ada Editor by CTRL+Shift+Click selecting the Ada element.
 - Added **Show In Ada Navigator** to the Ada editor context menu. Added support for the **Navigate > Show In** main menu item.
 - Selecting **Run > Add Expression (Ada)** from the main menu now by default uses the fully qualified name of the element under the Ada editor cursor.
 - [ERC 12204] [ECR 12205]
Improved file renaming in the Ada Navigator

Version 3.1.0.8

- Added a column to the process-selection dialog to display the window title for the process. Also added a properties table beneath the process table displaying additional properties of the selected process.

Version 3.1.0.7

- Added the ability to break on exceptions.
- Added a dialog to prompt the user to select a process to attach to when launching an "Attach To Process" debug session.

Version 3.1.0.1

- Added a **Toolchains** button to the `gnat` pretty printer settings dialog to allow the path to `gnatpp` to be set by selecting an installed toolchain. (GNAT only)

Version ADT_P2_020

- The installed toolchain wizard has a new page to set the location of the source files for a toolchain. All source files in these directories will be used in code completion and navigation searches.
- Build output for all builds is now more descriptive.
- [ECR 12187] Improved performance and functionality of code completion and navigation.

Version ADT_P2_019

- Performance of compiling a group of source files has been improved. (ObjectAda only)

Version ADT_P2_018

- Added a launch configuration type for attaching to a running Ada process.

Version ADT_P2_017

- Added a “clean” button to the main toolbar and **Clean Project** actions to the **Project** menu and to the Ada Navigator and Ada editor context menus.

Version ADT_P2_015

- Merged development branches for ObjectAda and Gnat. (Internal)
- [ECR 12171]
Updated the main unit browser to display the list of eligible main units.
- [ECR 12153]
The **New Ada File** wizard should open the new file after creation is complete.

Version ADT_P2_014

- [ECR 12162]
Added the ability to register the location of toolchain (termed "Installing" a toolchain) executables with ADT. This allows the project properties to refer to a named toolchain rather than a user-specific location.
- Added an option in the Ada editor preference pages to cause trailing spaces on lines to be removed when the file is saved.
- Added a **Fix Indentation** menu item and hot key (Ctrl+i) to fix the indentation of the selected lines in the open editor.
- Added an option in the Ada editor preference pages to cause full line comments to continue on the following line.
- Added a hot key for invoking the pretty printer (Ctrl+Shift+F).

Version ADT_P2_012:

- [ECR 12183]
Implemented the Ada Configurations view to display active configurations for all open projects and provide easier access to configuration properties.
- Created main toolbar buttons and hot keys for compiling the current selection and linking the selected project.

-
- When you press **OK** in the configuration management window, the selected configuration is set as the active configuration. Creating a configuration in the Ada Configurations view prompts for activation.

Version ADT_P2_011

- [ECR 12182]
Added support for the Eclipse Update Manager.

Version ADT_P2_010

- Added menu items to the Ada Navigator context menu and the main **Project** menu to trigger compilation of one or more Ada files and to trigger linking of the main files selected for a project.

Version ADT_P2_008

- Added a menu action to the Ada Navigator to allow pretty printing of selected files or all files in a directory or project.
- Added capability to select more than one item at a time from workspace resource trees.

Version ADT_P2_007:

- The user may now choose to use either the default pretty printer or the GNAT pretty printer (*gnatpp*). The user must already have GNAT installed in order to use the GNAT pretty printer. (GNAT only)
- The initial settings for a new configuration may now be copied from an existing configuration.

Version ADT_P2_006:

- Control+Space completion is now supported.
- Added menu items to the Eclipse menu and the Ada editor context menu to navigate to the spec declaration or body declaration of an Ada element.

Version ADT_P2_005

- The available completions for code completion updates as the user types. A completion may now be applied by selecting the appropriate completion and typing any of the following characters: space, period, semicolon, open parenthesis, or tab.

Version ADT_P2_004

- Parameters to subprograms are now displayed in the code completion dialog
- Elements in the code completion dialog now include an icon indicating the type of Ada component.

Version ADT_P2_003

- Added code assist for completion of dot-qualified identifiers.
- Searching for declarations and references for the element selected in the editor is now available in the editor context menu.

Version ADT_P2_002

- Added search for Ada construct declarations via the Eclipse **Search** dialog.
- Added search for references to Ada constructs via the Eclipse **Search** dialog.

Version ADT_P2_001

- Re-added file-specific linker options.

Version ADT_P2_000

- The Ada Navigator is accessible through the Ada perspective or via **Window > Show View > Ada > Ada Navigator**.
- The outline view for Ada files has been updated to display additional Ada constructs.
- The pretty printer is now accessible through the context menu for Ada source files via **Source > Format** or through the **Source** menu when an Ada source file is active.
- Properties for pretty printing are set in the project properties in the **Ada Pretty Printer** section.
- Pretty printer properties include the following:
 - (1) Casing for identifiers and keywords:
 - All upper case
 - All lower case
 - Mixed case, first letter is capitalized, others are forced to lower case
 - Mixed case, first letter is capitalized, others are ignored
 - (2) Indentation size for new blocks:
 - Toggle advanced formatting on/off. This includes restructuring context clauses and parameter lists.
 - (3) Toggle reformatting of type declarations. This includes the structuring of records:
 - Source directories for configurations may now be specified through the project properties dialog in the **Source Directories** tab.
- The current configuration may be changed through a menu item in the context menu for a project in the Ada navigator
