

Linux Plugin

This topic explains how to install the C++test plugin into a working copy of Eclipse or Linux system.

- For instructions on installing the plugin into Wind River Workbench, see [Overview of Wind River Workbench - C++test Integration](#).
- For instructions on installing the plugin into ARM RealView Development Suite, see [Overview of ARM RVDS - C++test Integration](#).

The section includes:

- [About Multi-user Installations](#)
- [Prerequisites](#)
- [Installation](#)
- [Startup](#)
- [Licensing](#)

About Multi-user Installations

Each instance of Eclipse creates and writes to a configuration folder. If multiple instances of Eclipse/C++test are running in parallel, data corruption may occur. Eclipse provides a number of strategies for initializing the Eclipse configuration area in order to support multi-user installations. If you are implementing a multi-user installation of C++test, please choose the appropriate installation scenario:

Shared Configuration

In this scenario, users share not only an install area but also a master configuration area. By default, users must still have their own private writable configuration areas. A user's private configuration area is cascaded to the master configuration and will not contain any interesting data if the master configuration has been fully initialized and no changes to the set of plug-ins to be installed has occurred.

In this scenario, the system administrator initializes the master configuration (typically under the install location), and ensures the whole install and configuration areas are read-only to users. When users run the Eclipse-based product from the shared install location, since they do not have write access privileges to the configuration area under the install area, a local configuration area will be automatically computed and initialized.

Shared Installation

In this scenario, a single install area is shared by many users. The 'configuration' directory under the install area is home only to the config.ini as shipped with the product (it is not initialized). Every user has their own local standalone configuration location. This scenario requires making the install area read-only for regular users. When users start Eclipse, this causes the configuration area to automatically default to a directory under the user home dir.

Prerequisites

System Requirements

- One of the following platforms:
 - Linux with glibc 2.12 or higher and an x86-compatible processor.
 - Linux with glibc 2.12 or higher and an x86_64-compatible processor.
- 4 GB RAM (8 GB is recommended).
- A supported compiler or cross-compiler.
 - See [Supported Environments](#) for a list of supported compilers.

Proper Compiler Configuration is Critical

In most cases, C++test needs to invoke the compiler and linker in order to perform static analysis and runtime testing tasks, which commonly involve preprocessing, compiling, and linking programs.

To access C++test's full functionality, the machine where C++test is run must have the complete development environment and compiler tool chain.

Choose the Proper Plug-In Architecture

- 32-bit IDEs require 32-bit C++test.
- 64-bit IDEs require 64-bit C++test.

IDE Requirements

- Eclipse IDE for C/C++ Developers 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 4.2, 4.3, 4.4, 4.5, 4.6, or 4.7 (32-bit) and a Java Runtime Environment (JRE) supported by Eclipse.
- Eclipse IDE for C/C++ Developers 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 4.2, 4.3, 4.4, 4.5, 4.6, or 4.7 (64-bit linux) and a Java Runtime Environment (JRE) supported by Eclipse.
- Wind River Workbench 2.6, 3.0-3.3.
- ARM Workbench IDE for RVDS 3.0-3.2.
- ARM DS-5 IDE

Other Requirements

- All users must be able to write to the Eclipse configuration directory. If all users cannot write to the current Eclipse configuration directory, then the location of that directory must be changed. To change the directory location, open the `<Eclipse_Installation_dir>/configuration/config.ini` file, then add a line of the format `osgi.configuration.area=@user.home/EclipseConfigData` (Be sure to enter the appropriate location.) This configures Eclipse to keep all its configuration data in the `$(HOME)/EclipseConfigData` directory. You must have full access rights to that location
- If earlier versions of C++test are installed on your system, the latest version of C++test must be installed in a fresh directory. Please install C++test in a different directory than the previous version installation, and then modify the PATH to point to the new installation.
- During installation, you need to have permission to write to the Eclipse installation directory.
- Additional disk space is required for C++test project data.
- The recommended Japanese language encoding is Shift_JIS (ja_JP.PCK locale on Unix). Other encodings might cause font problems or prevent C++test from reading test results.
- Installation scripts require that the working directory has write permissions.



Warning - Critical workaround for installations where some users have restricted write privileges

Known issues with the location of Eclipse configuration/cached data (https://bugs.eclipse.org/bugs/show_bug.cgi?id=54919) could prevent Eclipse from starting properly once C++test is installed.

To prevent this problem:

1. Open the `<Eclipse_Installation_Dir>/configuration/config.ini` file for editing.
2. Add the following line (the actual directory name can be changed): `osgi.configuration.area=@user.home/EclipseConfigData`

As a result, Eclipse should keep all its config data in `$(HOME)/EclipseConfigData` directory (you must have full access rights to that location).

Installation

Installing C++test involves installation of Parasoft Test components that are common across a number of Parasoft products. Multiple Parasoft Test version 9.x products can coexist on a single Eclipse installation. If you run the Parasoft C++test installation as described below, C++test will be added to the same Eclipse installation as your existing Parasoft products — just be sure to specify that you want to install C++test into the Eclipse installation directory that the other Parasoft Test products use.

If Parasoft Test is already installed, it will be upgraded to the latest version (if needed) during C++test installation.

If you want C++test to be installed along with pre 9.x Parasoft Test products, either install C++test into another Eclipse installation directory or uninstall the other products.

To install the C++test plugin for Eclipse on Linux:

1. Copy the distribution file to the target location.
 - Because Linux does not have any restrictions on having multiple versions of the same software installed on the same machine, the best way to install C++test is to provide a separate directory for each version.
 - For example, if your software is installed in `/opt/app/`, then the recommended installation directory configuration should be something like `/opt/app/parasoft/cpptest-extension/<version>`. Version 9.0 could be installed in `/opt/app/parasoft/cpptest-extension/9.0/` and version 9.1 could be installed in `/opt/app/parasoft/cpptest-extension/9.1/`. This provides the safest, most intuitive, and most flexible installation structure. During the installation, you will be prompted to enter a destination location for C++test Extension and Parasoft Test Extension files; defaults will be "parasoft/cpptest-extension/<version>" and "parasoft/test-extension/<version>". In this case, the target location for the distribution file should be `/opt/app`.
2. Unpack the installation script using the appropriate command(s):
 - `gunzip cpptest_<version>_<arch>.tar.gz`
 - `tar -xf cpptest_<version>_<arch>.tar`
3. Execute the following command to run the installation script:
 - `.cpptest_<version>_<arch>.sh`
4. Choose the appropriate Eclipse/CDT target platform type.
5. Enter your Eclipse (or Development Suite) installation directory, then click **OK**.
 - Choose the directory that contains `.eclipseproduct`.

If you later want to uninstall C++test:

1. Go to the C++test installation directory.
2. Execute `extuninstall` C++test will then be "unregistered" from Eclipse.
3. (Optional) Remove C++test files from the hard drive.

Alternative installation methods

For details on performing a silent installation see [Preconfiguration C++test During Installation](#).

Startup

Before launching C++test

To allow C++test to autodetect compiler and makefile settings, ensure that the necessary executables (compiler/linker, makefile, etc.) are correctly configured. "Correctly configured" means different things on different compilers, but it typically involves ensuring that the executable is on the PATH.

To launch the plugin:

- Launch Eclipse as normal.

Eclipse will automatically find the C++test plugin.

After Eclipse is launched, you should see a **Parasoft** menu added to the Eclipse menu bar. If you do not see this menu, choose **Window> Open Perspective> Other**, select `C++test`, then click **OK**.

If you suspect that C++test is not properly installed, see [Troubleshooting and FAQs](#) for help resolving some common installation problems.

Licensing

The license is configured through the centralized licensing framework (**Parasoft> Preferences> Parasoft> Licenses**). For details, see [Licensing](#).