

Linux Standalone

This topic explains how to install the standalone version of C++test (which is built upon the Eclipse framework) on a Linux system.

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.

The shared configuration described below is the recommended installation scenario implementing a multi-user installation of C++test. Refer to the Eclipse documentation for additional information.

Shared Configuration (Recommended)

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 plugins 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.
 - If you are using a cross-compiler that is not listed as supported, see [Configuring Testing with the Cross Compiler](#) for information on how you can use it with C++test.

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.

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 <Parasoft Test Install 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.
- 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.

Installation

To install the standalone version of C++test on a Linux system:

1. Copy the distribution file to the target location.
 - Since 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/<version>`. Version 9.0 could be installed in `/opt/app/parasoft/cpptest/9.0/` and version 9.1 could be installed in `/opt/app/parasoft/cpptest/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's and Parasoft Test's files; defaults will be "parasoft/cpptest/<version>" and "parasoft/test/<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. When the script requests confirmation, confirm that C++test should be installed in the current directory.
5. Follow the additional instructions given by the installation script.

C++test will be installed in the current directory.

Following the installation, set your path to include the installation location of C++test (unless you prefer to start C++test GUI with the full path `/install/directory/path/cpptest`).

Alternative installation methods

- For details on performing a silent installation see [Preconfiguration C++test During Installation](#).
- For details on using C++test on a Headless Machine see [Working with C++test on a Headless Linux Machine](#).

Startup

Before launching C++test

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

To launch the standalone:

- Launch the `cpptest` executable. Include the appropriate location (install dir) on the \$PATH or launch it with the full path to the executable (`/install/directory/path/cpptest`).

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](#).