# Importing your C/C++ source files into Eclipse

Using the CVS Repository Exploring perspective, you can check out modules or directories into existing projects, or create new projects. A common scenario that you may encounter when starting to use the CDT, is determining how to bring existing C/C++ source files into Eclipse. There are a number of ways to do this. The scenarios described below are recommended approaches.

## Creating a project from source files in CVS

If your existing source tree is managed in CVS, you can use the CVS Repository perspective to "Checkout As..." any folder in the repository. The first time you "Checkout As...", the New Project wizard launches and you need to create a C or C++ project for the folder. For more information, see [Creating a project](http://docs.google.com/tasks/cdt_t_proj_new.htm), [Creating a project via a project template](http://docs.google.com/tasks/cdt_t_proj_new_with_template.htm) and [Working with C/C++ project files](http://docs.google.com/tasks/cdt_o_proj_files.htm).

After you perform a checkout operation, a CVS checkout of the project occurs into the project's location. It is recommended that you eventually add and commit the CDT project files back into CVS. These files include .project, .cdtproject, and .cdtbuild (for Managed Build projects) and are located at the root folder of each CDT project.

Bringing code into the IDE from CVS differs slightly, depending on what you're importing:

* an existing C/C++ project
* existing C/C++ code that isn't part of a project
* existing C/C++ code that needs to be added to an existing project

### Importing a C/C++ project from CVS

To check out an existing C/C++ project from the CVS repository into your workspace, right-click the project in the CVS Repositories view, and select **Check Out** from the menu. A project with the same name as the CVS module is checked out in to your workspace.

### Importing C/C++ code from CVS

To check out existing C/C++ code that is not part of a project:

1. Right-click the module or directory in the CVS Repositories view and choose **Check Out As…** from the menu.  
   The Check Out As wizard dispalys.
2. Choose how to check out this project:
   * as a project configured using the New Project wizard  
     or:
   * as a new project in the workspace  
     or:
   * as a Standard Make C/C++ Project – if you need to create your own Makefile to integrate with an existing build process

Choose the workspace location for this project, then the CVS tag to check out.

1. Click **Finish** to exit the **Check Out As** dialog.
2. Click **Next** to continue.

### Importing C/C++ code into an existing project

To import a directory full of C/C++ code into an existing project:

1. Right-click the module or directory in the CVS Repositories view and choose **Check Out As** from the menu.  
   The IDE displays the Check Out As dialog.
2. Choose **Check out into an existing project**, and then click **Next**. The IDE displays the Check Out Into dialog:
3. Select an existing project from the list, and then click **Finish** to add the code from CVS to the selected project.

## Creating new projects from existing source roots

If your resource code is not managed in CVS but is available from the file system, then you need to perform two steps:

1. Identify a "root folder" of your source code tree.
2. Create a new C/C++ project using the New Project Wizard, and specify the "root folder" as a non-default location of the new project.

Typically, existing projects will have their own makefiles, so you should create a new Make C/C++ project. For more information see [Creating a project](http://docs.google.com/tasks/cdt_t_proj_new.htm) and [Working with C/C++ project files](http://docs.google.com/tasks/cdt_o_proj_files.htm).

To help you to identify a root folder for your project, consider the following guidelines:

* all source code for the project is available on or beneath the root folder
* the build results are also produced in or beneath the root folder
* there is often a makefile in the root folder. In complex projects, the makefile in the root folder calls other makefiles in other directories to produce the build results.
* external header files and library files do not need to be in or beneath the root folder.

The resources for the project are maintained in the remote location specified, not in the workspace folder for Eclipse. However, your existing folder structure is displayed in the C/C++ Projects view. Meta data for the project, such as the index for the project and the link to the existing source, is stored in the metadata directory in the workspace folder. For more information on the workspace folder, see **Workbench User Guide > Tasks > Upgrading Eclipse**.

After you create a CDT project, you cannot easily move it or redefine its root folders. If you need to, you can delete the CDT project (without deleting its contents) and then recreate it specifying a different non-default location.

## Import your C/C++ source file system

Another approach would be to create a C/C++ Project and then import your existing file system. For more information about importing, see **Workbench User Guide > Tasks > Importing > Importing resources from the file system**.

This approach copies the files from your file system to an Eclipse Workbench project or folder. Your original source files remain unchanged and it is the copies of the files that will be edited, built and debugged using the CDT.  When you have successfully imported your existing file system, the folder structure is displayed in the C/C++ Projects view. Again, you should identify an appropriate "root folder" to import from.

**Tip:**

* Importing your existing file system can consume significant disk space depending on the size of your files.
* Your files may become detached from an existing source control system that previously referenced the original file location such as a ClearCase view.



[Overview of the CDT](http://docs.google.com/concepts/cdt_c_over_cdt.htm)

[CDT Projects](http://docs.google.com/concepts/cdt_c_projects.htm)



[Working with C/C++ project files](http://docs.google.com/tasks/cdt_o_proj_files.htm)



[Project properties](http://docs.google.com/reference/cdt_o_proj_prop_pages.htm)

