Configuring the NetBeans IDE for C/C++/Fortran

April 2011 [Revision number: V7.0-1]

This document provides information about downloading, installing, and configuring C/C++ support in the NetBeans IDE. Before you can develop in C/C++, the NetBeans IDE requires the C/C++ plugin module, and third party C/C++ compilers, make utilities, and debuggers.

Contents

To follow this tutorial, you need the following software and resources.

Software or Resource	Version Required
NetBeans IDE	version 7.0 with NetBeans C/C++ plugin
Java Developer Kit (JDK)	version 6

Enabling C/C++/Fortran in the IDE

The NetBeans IDE is a dynamic modular IDE, which means you can change it by adding and removing modules of the program. You can add functionality by installing plugin modules, or remove functionality by uninstalling plugin modules. See the NetBeans IDE online help for more information about plugins.

If you want to work with C, C++, or Fortran programs in the NetBeans IDE, your NetBeans installation must include the C/C++ plugin.

> Although the plugin is named C/C++, it also includes support for Fortran programming.

If you have not yet installed NetBeans IDE 7.0

If you have not yet installed the NetBeans IDE, download either the **C/C++** bundle or the **All** bundle from NetBeans IDE 7.0 Download Page. If you will not be using other languages such as Java and Ruby, you should download the C/C++ bundle.

If you have NetBeans IDE 7.0 but do not know if you have the C/C++ plugin

If you already have NetBeans IDE, determine if your NetBeans IDE includes the C/C++ plugin already, by selecting File > New Project. If C/C++ is listed as one of the Categories, you have the C/C++ plugin module. You should skip to the section Installing and Setting Up the Compilers and Tools.

If you have NetBeans IDE 7.0 without the C/C++ plugin

If your NetBeans IDE does not show a C/C++ project category when you select File > New Project, complete the following steps to add the C/C++ plugin module to the IDE.

1. If your network uses a proxy, choose Tools > Options > General in the IDE, select Manual Proxy Settings, enter the HTTP Proxy and Port for your proxy, and click OK.
2. Choose Tools > Plugins.
3. In the Plugins dialog box, click the Available Plugins tab, and scroll to the C/C++ category.
4. Select the C/C++ checkbox and click Install to start the NetBeans IDE Installer.
5. In the NetBeans IDE Installer, click Next.
6. Read the license agreement, select the checkbox to accept the terms of the license agreement, and click Next.
7. Click Install.
8. After the installation completes, select either Restart IDE Now or Restart IDE Later and click Finish.

Installing and Setting Up the Compilers and Tools

The Netbeans C/C++ module requires a C compiler, C++ compiler, make utility, and gdb debugger. See the following

instructions for the platform of your development system.

- Microsoft Windows
- Oracle Solaris OS
- Linux
- Macintosh OS X

Microsoft Windows

The NetBeans C/C++ module has been tested with compilers from Cygwin and MinGW. If you install both Cygwin and MinGW, be careful to keep their installation locations completely separate and do not mix tools from Cygwin and MinGW in one tool collection in the IDE.

> If you want to use Qt with Windows, you must use MinGW, which is included in the Qt SDK. You should not install MinGW separately.

- Cygwin Compilers and Tools
- MinGW Compilers and Tools

Cygwin Compilers and Tools

The NetBeans C/C++ module has been tested with the following compilers and tools from Cygwin.com.

Software or Resource	Version Tested	Description
`cygwin1.dll`	1.7.7	Cygwin Linux-like environment for Windows
`gcc`	4.3.4	Cygwin C compiler
`g++`	4.3.4	Cygwin C++ compiler
`gdb`	6.8	Cygwin GNU debugger
`make`	3.81	Cygwin make utility

If you already have the Cygwin gcc and `g++` compilers, GNU `make`, and `gdb` debugger installed on your Windows system and your path is set up correctly to find them, make sure that you have the correct versions.

To check the versions of your Cygwin compilers and tools:

1. Check the version of Cygwin environment by typing the following commands at a Windows command prompt:

```
C:\> cygcheck -c cygwin
```

2. Check the versions of the Cygwin gcc and `g++` compilers, `make`, and `gdb` by typing the following commands at a Windows command prompt:

```
C:\> gcc --version
C:\> g++ --version
C:\> make --version
C:\> gdb --version
```

If you have the correct versions, then no further setup is necessary. See Verifying the Installation to verify that the tools are installed correctly for the NetBeans IDE.

To install the GNU gcc and `g++` compilers, `make`, and `gdb` debugger from cygwin.com**:**

1. Refer to the Cygwin User's Guide for complete information about installing and using Cygwin.
2. Download the Cygwin `setup.exe` program by clicking Install Cygwin in the left navigation bar, or by clicking this direct `setup.exe` link.
3. Run the `setup.exe` program. Accept the defaults until you reach the Select Your Internet Connection page. Select the option on this page that is best for you. Click Next.
4. On the Choose Download Site page, choose a download site you think might be relatively close to you. Click Next.
5. On the Select Packages page you select the packages to download. Click the + next to Devel to expand the development tools category. You may want to resize the window so you can see more of it at one time.
6. Select each package you want to download by clicking the Skip label next to it, which reveals the version number of the

package to download. At a minimum, select
- gcc-core: C compiler
- gcc-g++: C++ compiler
- gdb: The GNU Debugger
- make: the GNU version of the 'make' utility

Packages that are required by the packages you select are automatically selected as well.

7. Click Next to connect to the download site and download the packages you selected, and click Finish when the installation is complete.
8. Now add the Cygwin compiler directory to your path to enable NetBeans IDE to find the tools collection:

    a. Open the Control Panel:
    - On Windows XP select Start > Settings > Control Panel) and double-click System.
    - On Windows 7, type **var** in the Start menu's search box to quickly find a link to Edit the system environment variables.

    b. Select the Advanced tab and click Environment Variables.

    c. In the System Variables panel of the Environment Variables dialog, select the `Path` variable and click Edit.

    d. Add the path to the *`cygwin-directory`*`\bin` directory to the `Path` variable, and click OK. By default, *`cygwin-directory`* is `C:\cygwin`. Directory names must be separated with a semicolon. Your edited path should look something like `%SystemRoot%\system32;%SystemRoot%;C:\Program Files\QuickTime\QTSystem;C:\cygwin\bin`

    e. Click OK in the Environment Variables dialog and the System Properties dialog.

    f. See Verifying the Installation to verify that the tools were installed correctly for the NetBeans IDE.

MinGW Compilers and Tools

NetBeans IDE 7.0 was tested with Minimalist GNU for Windows (MinGW) and the Minimal System (MSYS) Unix-like environment. Versions tested and installation instructions are shown below.

> If you want to use Qt with Windows, you should use the version of MinGW that is included in the Qt SDK instead of installing MinGW separately.

Software or Resource	Version Tested	Description
`gcc`	3.4.5	MinGW C compiler
`g++`	3.4.5	MinGW C++ compiler
`gdb`	7.0	MinGW GNU debugger
`make`	3.81	MSYS `make` utility Note that MinGW make is not supported

To install the GNU compilers, `make`, and `gdb` debugger from mingw.org**:**

1. Log in to Windows using an account with computer administrator privileges.
2. Download the self-extracting installer from How To Install the MinGW (GCC) Compiler Suite on mingw.org or for convenience you can download from this direct link.
3. Run the MinGW installer in Windows.
4. In the MinGW installer, accept the defaults until you get to the Choose Components page.
5. On the Choose Components page, select the `g++` compiler. The gcc compiler is automatically included in the installation so it is not a selectable component.
6. Do not select MinGW make, as you need to use the make from MSYS instead, which is downloaded separately. Click Next.
7. Use the default `C:\MinGW` as the destination folder if possible to minimize any potential difficulty with using the compilers from another location.

8. Click Install to install the MinGW tools, then click Next, then click Finish.
9. Add the `C:\MinGW\bin` directory to your path, as described in Environment Settings in the MinGW wiki.
10. See the installation instructions for MSYS at the MinGW MSYS wiki. You only need to install the MSYS 1.0 files. You do not need to install the DTK or the core files mentioned in the wiki page.
11. For convenience, you can download from this direct link to MSYS-1.0.10.exe
12. Make sure you are using a Windows account with "computer administrator" privileges when you install MSYS. The NetBeans IDE might have issues when you try to build and run projects later if MSYS is installed under a limited user account.
13. Run the MSYS-1.0.10.exe installer and accept the defaults.
14. To install the gdb debugger, download from this direct link to gdb-7.0-2-mingw32-bin.tar.gz. You must have a Windows zip utility such as gzip, 7-zip, or WinZip to extract this gdb archive.
15. Unzip the gdb-7.0-2-mingw32-bin.tar.gz to your `C:\MinGW` directory so that the gdb executable is installed into your `C:\MinGW\bin` directory.
16. See Verifying the Installation to verify that the tools were installed correctly for the NetBeans IDE.

Oracle Solaris OS

On the Oracle Solaris OS, you can use GNU tools or Oracle Solaris Studio tools. The GNU tools are included in Oracle Solaris 10 OS in the `/usr/sfw/bin` directory by default. The Oracle Solaris Studio software is a suite of developer tools that are free to download. The Oracle Solaris Studio compilers are optimized for Sun hardware, and make it easier to produce performance tuned Oracle Solaris binaries.

NetBeans IDE 7.0 has been tested with the following compilers and tools.

Software or Resource	Version Tested	Description
`cc`	5.9, 5.10, 5.11	Sun Studio 12, Sun Studio 12 Update 1, and Oracle Solaris Studio 12.2 C compilers
`CC`	5.9, 5.10, 5.11	Sun Studio 12, Sun Studio 12 Update 1, and Oracle Solaris Studio 12.2 C++ compilers
`gcc`	3.4.3, 3.4.6	GNU C compilers
`g++`	3.4.3, 3.4.6	GNU C++ compilers
`gdb`	6.8	GNU debugger
`gmake`	3.81	GNU make
`make`	3.81	Solaris make
`dmake`	7.8, 7.9, 8.0	Sun Studio 12, Sun Studio 12 Update 1, and Oracle Solaris Studio 12.2 distributed make utility

Oracle Solaris Studio 12.2 Compilers

If you want to use the Oracle Solaris Studio 12.2 compilers on Solaris 10 OS:

- If you have Oracle Solaris Studio 12.2 software installed, ensure that */installation directory/*`solstudio12.2/bin` is in your path before you start the NetBeans IDE. The default location on Solaris 10 OS is `/opt/solstudio12.2/bin`
- If you do not have Oracle Solaris Studio 12.2 software installed, you can download it free at `http://www.oracle.com/technetwork/server-storage/solarisstudio/downloads/index-jsp-141149.html`.

To download and install the Oracle Solaris Studio 12.2 compilers:

1. Create a directory for the downloaded file. You must have write permission for this directory.
2. Download the package installer file for your platform into the download directory.
3. Go to the download directory, and uncompress and untar the downloaded file.

```
bzcat filename | /bin/tar xvf -
```

4. Follow the instructions in Chapter 2 of the *Oracle Solaris Studio 12.2 Installation Guide* to install the C compiler, C++ compiler, and required Solaris patches.
5. Edit your PATH to add the path to the Oracle Solaris Studio 12.2 software before starting the NetBeans IDE.
6. See Verifying the Installation to verify that the tools were installed correctly for the NetBeans IDE.

GNU Compilers and GNU make

If you want to use the GNU compilers and GNU `make`:

- If you have a standard installation of the Oracle Solaris 10 OS, the compilers and `gmake` are installed in `/usr/sfw/bin`. Make sure that this location is in your PATH before starting the NetBeans IDE.
- If the compilers and `gmake` are not installed on your system, you can download them from http://www.sunfreeware.com.

To download and install the GNU compilers and make utility

1. Download `gcc` and `make`.
2. If the download zip files are not automatically extracted during download, unzip them with `gunzip`.
3. Install the packages with the `pkgadd` command.
4. Make sure to include the GNU compiler directory and the GNU make directory in your path before starting the NetBeans IDE.
5. See Verifying the Installation to verify that the tools were installed correctly for the NetBeans IDE.

gdb Debugger

Whether you use the Oracle Solaris Studio compilers and Solaris `make` or the GNU compilers and GNU `make`, you must have the `gdb` debugger to debug applications in NetBeans IDE. You can download `gdb` from http://www.sunfreeware.com.

To download and install gdb:

1. Download `gdb` 6.6 or 6.8.
2. If the download zip file is not automatically extracted during download, unzip it with `gunzip`.
3. Install the package with the `pkgadd` command.
4. Make sure to include the path to `gdb` in your PATH before starting the NetBeans IDE.
5. See Verifying the Installation to verify that the tools were installed correctly for the NetBeans IDE.

Linux

On Linux platforms, you can use GNU tools or Oracle Solaris Studio tools.

NetBeans IDE has been tested with the following compilers and tools:

Software or Resource	Version Tested	Description
`cc`	5.9, 5.10, 5.11	Sun Studio 12, Sun Studio 12 Update 1, and Oracle Solaris Studio 12.2 C compilers
`CC`	5.9, 5.10, 5.11	Sun Studio 12, Sun Studio 12 Update 1, and Oracle Solaris Studio 12.2 C++ compilers
`gcc`	4.3.3, 4.4.5	GNU C compiler in Red Hat Enterprise Linux 5 and Ubuntu 8.04, GNU C compiler in Oracle Linux 6 and Ubuntu 10.10
`g++`	4.3.3, 4.4.5	GNU C++ compiler in Red Hat Enterprise Linux 5 and Ubuntu 8.04, GNU C++ compiler in Oracle Linux 6 and Ubuntu 10.10
`gdb`	6.8, 7.2	GNU debugger in Red Hat Enterprise Linux 5 and Ubuntu 8.04, GNU debugger in Oracle Linux 6 and Ubuntu 10.10
`gmake`	3.81	GNU make in Red Hat Enterprise Linux 5 and Ubuntu 8.04

Oracle Solaris Studio 12.2 Compilers on Linux

If you want to use the Oracle Solaris Studio 12.2 compilers in a Linux OS:

- If you have Oracle Solaris Studio 12.2 software installed, ensure that `/installation directory/solstudio12.2/bin` is in your path before you start the NetBeans IDE. The default location is `/opt/oracle/solstudio12.2/bin` when installing with Linux packages.
- If you do not have Oracle Solaris Studio 12.2 software installed, you can download it free at `http://www.oracle.com/technetwork/server-storage/solarisstudio/downloads/index-jsp-141149.html`.

To download and install the Oracle Solaris Studio 12.2 compilers:

1. Create a directory for the downloaded file. You must have write permission for this directory.
2. If your browser is set to download to a particular location such as your Desktop or a Downloads directory without prompting, set the browser preferences to download to the directory you created. For Firefox, the download directory is set in Edit > Preferences > Main.
3. Download the file for your platform into the download directory you created. If you are using Ubuntu, download the tarfile installation into the directory where you want to install it because the packages are for Oracle Linux, Red Hat Enterprise Linux, and SuSE Linux Enterprise System. Note also that Ubuntu is not an officially supported platform for Oracle Solaris Studio IDE, but the compilers have been tested for use in NetBeans IDE and been found to work.
4. Go to the download directory, and uncompress and untar the downloaded file.

```
bzcat filename | tar xvf -
```

5. If you downloaded the SUSE or RPM packages, follow the instructions in Chapter 2 of the *Oracle Solaris Studio 12.2 Installation Guide* to install the C compiler, C++ compiler, and Fortran compiler.
6. Edit your PATH to add the path to the Oracle Solaris Studio software before starting the NetBeans IDE.
7. See Verifying the Installation to verify that the tools were installed correctly for the NetBeans IDE.

To download and install the GNU debugger in Oracle Linux or Red Hat Enterprise Linux, type:

```
yum install gdb
```

To download and install the GNU debugger in Ubuntu, type:

```
apt-get install gdb
```

Macintosh OS X

NetBeans IDE has been tested with the following compilers and tools:

Software or Resource	Version Tested	Description
gcc	4.2.1	GNU C compiler in Mac OS X 10.6
g++	4.2.1	GNU C++ compiler in Mac OS X 10.6
gdb	6.3.5	GNU debugger in Mac OS X 10.6
make	3.81	GNU make in Mac OS X 10.6

Install the following packages that are provided with your Macintosh OS X:

- Xcode
- X11

These packages can also be downloaded from Apple Developer Connection with the free ADC membership.

Verifying the Installation

To verify that the installation is correct, start the NetBeans IDE, build a sample project, and run it in the gdb debugger.

Start the NetBeans IDE

To start the IDE, do one of the following:

- Double-click the NetBeans IDE icon on the desktop.
- On Linux or Solaris platforms, navigate to the `bin` subdirectory of your NetBeans installation and type `./netbeans`.
- Launch the NetBeans IDE through the desktop menu.
  This method is available on Windows platforms, but might not be available on the Solaris 10 OS and some Linux platforms.

Build a Sample Project to Test the Tools

On all platforms, you can build a sample project to test the compiler installation, and run it in the debugger to test the gdb installation.

To build a sample project and run it in the debugger:

1. Open the New Project wizard by choosing File > New Project.
2. In the Categories panel on the Choose Project page of the wizard, expand the Samples category and select the C/C++ subcategory.
3. In the Projects panel, select the `Welcome` project. Click Next.
4. On the Project Name and Location page, click Finish.
5. In the Projects window, right-click the `Welcome_1` project and choose Build. If your compilers and `make` utility are installed correctly and the path to them is set, build output is displayed in the Output window and the project builds successfully.
6. Expand the Source Files node of the project and double-click the `welcome.cc` file to open it in the Source Editor.
7. Set a breakpoint by clicking in the left margin of the Source Editor window next to any line.
8. Right-click the project and choose Debug. If the `gdb` debugger is installed correctly and the path to it is set, `gdb` starts up, the Debugger tabs are displayed, and the Welcome application runs and stops at the breakpoint.
9. Choose Debug > Continue to run the application to completion.
10. If the project doesn't build or debugger doesn't work, see the next section.

Troubleshooting Tool Issues

1. Select Tools > Options and click C/C++ in the top panel of the Options dialog box.
2. In the Build Tools tab, look at the Tool Collection list to see if your tool collection is listed.
3. Select the tool collection if it is listed, and check the paths to the tools. If the Base Directory path is incorrect, you may have made a mistake when setting your path environment variable. Refer back to the instructions for setting the path in the section for your platform in this document, and correct the path if necessary.
4. If the tool collection is not listed, click Restore Defaults. This will cause the IDE to rescan your environment to look for tools and the tool collection should be found if the path environment variable is correct.
5. If you are still having an issue, try adding a new tool collection and specifying the path to the tools, as follows:
   a. Click Add below the Tool Collection list.
   b. Browse to the directory that contains the executables for the compilers, debugger, and make utility. This is usually the `bin` directory.
   c. Select the directory and click Open. The IDE should be able to determine the family of the tools and update the other fields in the dialog box appropriately. If not, an error is displayed.
   d. If the tools are identified correctly, click OK in the Add Tool Collection dialog box. The new collection should be displayed in the list.

e. Select the new tool collection and click Default to make sure this tool collection is used automatically for new projects.

If you cannot solve the problem, ask for help at the NetBeans C/C++ User Forum.

Next Steps

To learn quickly how to develop a C or C++ application with the NetBeans IDE, see the C/C++ Projects Quick Start Tutorial

Send Us Your Feedback