

By Jack Wallen

Linux is a great development environment. But without sound development tools, that environment won't do you any good. Fortunately, plenty of Linux and/or open source development tools are available. If you're a new user you might not know which tools are there, but worry not. Here are 10 outstanding tools that will help you take your development to another level.

1: Bluefish

[Bluefish](#) is one of the most popular IDEs for Web development available. It can handle programming and markup languages, but it focuses on creating dynamic and interactive Web sites. Like many Linux applications, Bluefish is lightweight (using about 30% to 40% of the resources that similar applications use) and fast. Bluefish can open multiple documents at once (up to 3,500 documents, if needed). It includes project support, remote file support, search and replace (including regular expressions), unlimited undo/redo, customizable syntax highlighting for many languages, anti-aliased text in windows, and multiple encodings support, among other features.

One of Bluefish's nice features is the Quickba, a user-defined toolbar that allows you to add buttons by right-clicking and choosing Add To Quickbar. You can add any HTML toolbar buttons to the Quickbar. Bluefish has a number of simple tools to help you add various elements to your code. Need a DHTML auto-submit select box? Easy. Choose Auto-submit Select Box from the DHTML drop-down and fill out the necessary items to add the element to your code. Bluefish has wizards for C, Apache, DHTML, DocBook, HTML, PHP+HTML, and SQL. If you develop your sites by hand, you should certainly be using Bluefish.

2: Anjuta

[Anjuta](#) is a free, open source IDE for the C and C++ languages. It's easy to install (*urpmi anjuta* on Mandriva, for example) and offers such features as project management, application wizards, an interactive debugger, and a powerful source code editor (with source browsing, code completion, and syntax highlighting). The Anjuta team developed this powerful IDE to be easy to use and still meet all of your C and C++ programming needs.

Anjuta has a flexible and powerful user interface that allows you to drag and drop the tools in the layout to arrange the GUI nearly any way you like. And each user-configured layout is persistent for the project (so you can have different layouts for every project you have going). Anjuta also enjoys a powerful plug-in system that allows you to decide which plug-ins are active and which are not for each project. And like all open source projects, you can develop your own plug-ins for Anjuta. One of the most powerful tools in the Anjuta application is the project manager. This tool can open nearly any automake/autoconf-based project. This project manager doesn't add any Anjuta-based information to the project, so your project can be maintained and developed outside of Anjuta as well.

3: Glade

[Glade](#) is a RAD (rapid application development) tool used to create GTK+ toolkit and for the GNOME desktop. Its interface is similar to that of The GIMP and can be customized and even embedded into Anjuta. Glade includes a number of interface building blocks, such as text boxes, dialog labels, numeric entries, check boxes, and menus, to make the development of interfaces quicker. Interface designs are stored in XML format, which allows these designs to be easily interfaced with external tools. Installing Glade is simple. For instance, when in Fedora, you can issue the command *yum install glade3*. Glade does not have as powerful a project manager as Anjuta, but you can create, edit, and save projects with Glade.

4: Gcc

[Gcc](#) is a GNU compiler that works for C, C++, Objective C, FORTRAN, Java, and Ada. It's a command-line tool but is very powerful. Many IDEs have tools that are merely front ends for gcc. Gcc is actually a set of tools. The most used are the compilers for C and C++ code. How does one tool compile for different languages? Simple: For C, you invoke the "gcc" command and for C++, you invoke the "g++" command. Two compilers in the same toolkit. And g++ is a compiler, not just a preprocessor. It will build object code from source code without using an intermediary to first build C code from C++ source. This creates better object code and gives you better debugging information.

5: Kdevelop

[Kdevelop](#) was created in 1998 to be an easy-to-use IDE for the KDE desktop. Kdevelop is now released under the GPL and is free to use. It's plug-in based, so you can add and remove plugs to create the exact feature set you need. Kdevelop also includes profile support so that various sets of plug-ins can be associated with specific projects. Kdevelop supports 15 programming languages, with each having language-specific features. Kdevelop also offers an included debugger, version control system (Subversion), application wizard, documentation viewer, code snippets, Doxygen integration, RAD tools, Ctags support, code reformatting, QuickOpen support, and dockable windows and toolbars. One of the best things about Kdevelop is that it takes much of the low level tasks out of the hands of the users. Having to deal with make, automake, and configure can be cumbersome. Any good developer should know these tools, and Kdevelop includes an Automake manager to simplify their use. One other nice touch is that the output window of the compiler is colorized, so it is easier to instantly see the difference between errors, warnings, and messages.

6: Gdb

[Gdb](#) isn't really a developer tool, but it is a tool that most *NIX developers (and many Windows developers) consider a must-have. Gdb is the GNU Debugger. This tool is issued from the command line and will give a developer instant feedback from within another program while it is executed. Say you create, complete, and release an application, only to hear there are problems. To help you find these problems you can start your program from the *gdb* utility which will help you discover where the problems are. With gdb you can do the following:

- Start your application specifying arguments, switches, or input that would affect its behavior.
- Cause your application to halt on specified behavior.
- Examine what has occurred when your application stops.
- Make changes to your program so you can test on the fly.

Gdb is also very handy to have when doing bug reporting.

7: Kompozer

[Kompozer](#) is an easy-to-use WYSIWYG Web authoring tool that is aimed at the nontechnical user who wants to create a professional Web site without having to know HTML. Kompozer has a number of outstanding features. One of the best features is the ability to open a Web site from a URL, edit that site, and upload the edited site. This allows for simple updating of sites without having to edit HTML. Of course, you will need permission to upload to a site for this to actually work. This feature is also good for using other sites as templates. Kompozer shouldn't be thought of as a beginner-only tool. Think of it as a free open source replacement for Microsoft FrontPage and Adobe Dreamweaver. And like its more costly alternatives, Kompozer can jump between WYSIWYG and code editing with a click of a tab.

8: Eclipse

[Eclipse](#) is a multi-language IDE, written in Java, with an extensive plug-in system to allow you to extend functionality. Downloaded over 1 million times each month, Eclipse is one of the strongest forces in software development today. Eclipse is, in fact, the de facto standard for open source development. Probably the strongest aspect of Eclipse is the plug-in feature. In languages alone, Eclipse boasts 58 plug-ins. These aren't spoken languages -- they're development languages. On top of the rich environment, Eclipse has a huge community and plenty of organizations offering training in the IDE (which even includes an Eclipse University.)

9: Make

Make is a Linux utility that can automatically determine which pieces of a larger program need to be recompiled. Once make determines what bits need to be recompiled, it issues the necessary commands to finish the action. Make is often used when installing applications from source, so developers of open source applications should have a strong understanding of the make tool and how it is used. If you plan on developing an application that can be installed from source, you will need to know how to create a makefile. This makefile describes the relationships among the various files of your application and includes the statements needed to piece them together. If you're familiar with installing applications, you know the command string `./configure; make ; make install`.

10: Quanta Plus

[Quanta Plus](#) is another HTML development tool, similar to Kompozer. Quanta Plus is capable of both WYSIWYG and hand coding and supports: HTML, XHTML, CSS, XML (and XML-based languages), and PHP. Quanta Plus features on-the-fly tag completion, project management, live preview, a PHP debugger, CVS support, and Subversion support (with a plug-in). Where Kompozer is aimed primarily at the nontechnical user, Quanta Plus is aimed at the more technical user who wants a good WYSIWYG editor.

Additional resources

- TechRepublic's [Downloads RSS Feed](#) **XML**
- Sign up for the [Downloads at TechRepublic](#) newsletter
- Sign up for our [Linux NetNote](#)
- Check out all of TechRepublic's [free newsletters](#)
- [10 questions to ask before migrating to Linux](#)
- [10 obscure Linux applications you need to try](#)
- [10 quick tips to make Linux networking easier](#)

Version history

Version: 1.0

Published: March 16, 2009