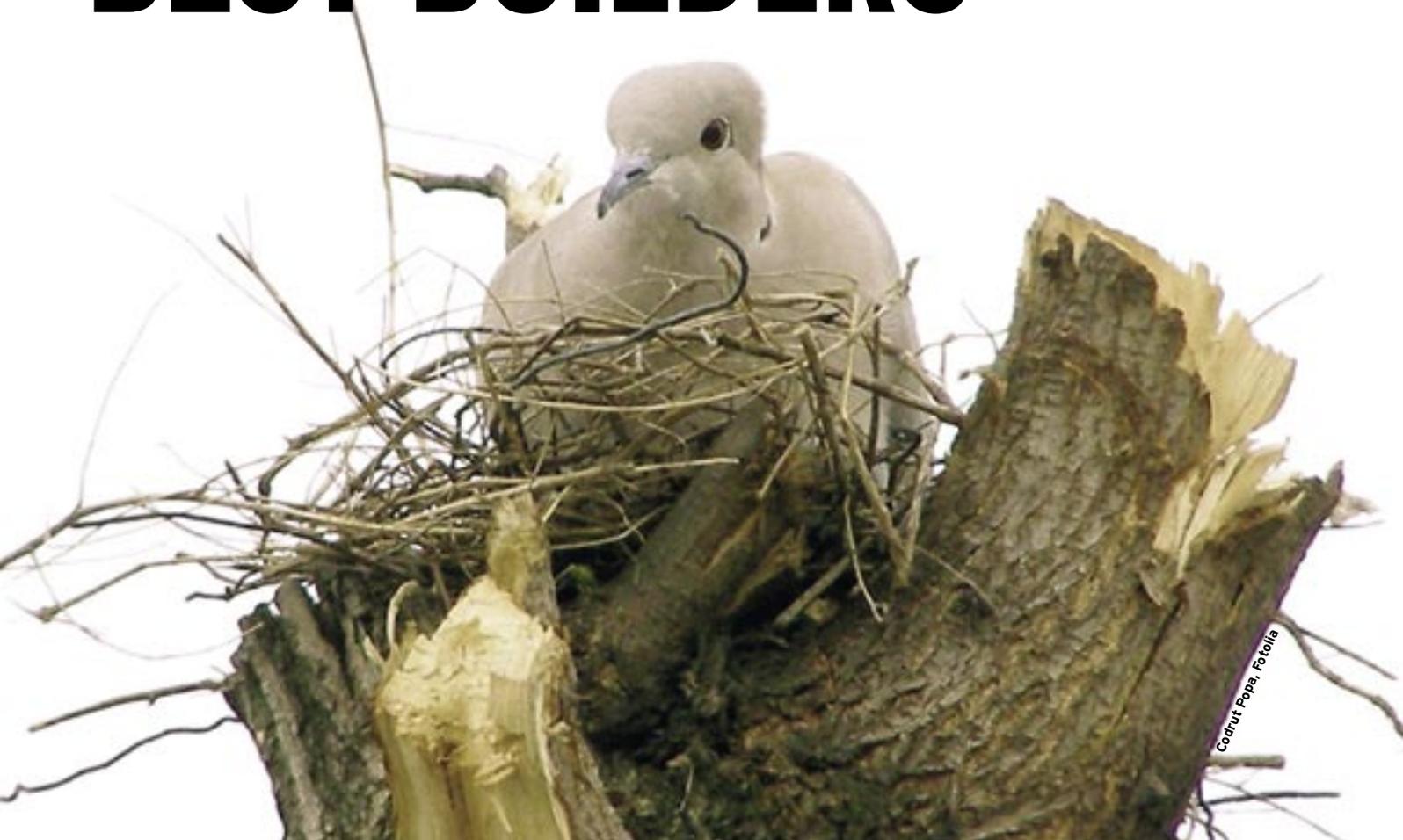


Comparing Python IDEs

BEST BUILDERS



A number of useful development environments are available for Python in Linux. We'll show you around some popular Python IDEs that are more than just text editors. **BY CARSTEN SCHNOBER**

What can you do with an Integrated Development Environment (IDE) that you can't do with a text editor? Vi, Emacs, Gedit, and Kate all offer features such as syntax highlighting and code folding for all major programming languages.

A good IDE, on the other hand, adds features such as debugging and project management.

The Contenders

This article examines some of the leading Python IDEs. I will focus on applications intended specifically for the Python environment. Other options include tools, such as Komodo and Eclipse, that

support Python along with a number of other programming languages.

► Wing IDE

Wing IDE [1] (Figure 1) is a commercial developer environment that focuses entirely on Python. Two variants of Wing are available: Wing IDE Professional costs around US\$ 180, whereas the Personal edition is US\$ 35. Despite its price, Wing does lack a couple of critical features (see Table 1).

If you are interested, you can download Wing and use your credit card to purchase a license. Up to three evaluation licenses per person are available for a trial period of up to 10 days.

Without a license, the program runs for only 10 minutes. For just US\$ 30 more, the developer will send you a CD, but only within the United States.

What I like about Wing IDE is its clear overview of the programming environment. The Source Browser in the Professional version lists functions, classes, and methods. Down to the level of individual variables, you can expand entries and then click to add them to your code.

The Project Management feature supports multiple-file software development, giving the programmer an overview of the files within the project scope and defining specific properties, such as an alternative Python interpreter or an extended Python path.

The Professional version of Wing IDE, in particular, facilitates collaboration on projects. It can handle project synchronization via CVS, Subversion, or Perforce automatically, and Wing will even handle details such as which control charac-

ter to use for line breaks – a typical headache in cross-platform cooperation.

The Personal version lacks the ability to define keyboard macros and many debugging functions compared with the Professional version. Additionally, only the editor in the more expensive Wing IDE version supports code folding.

Because the Personal version lacks so many useful features, it is more like a trial version, and although the price of US\$ 35 is far less than the asking price for Wing IDE Professional version, the Personal version hardly has anything going for it compared with free developer environments.

► Boa Constructor

The GPL-licensed Boa Constructor [2] developer environment (Figure 2) relies doubly on the Wxwidgets [3] GUI toolkit. For one thing, the Boa's own user interface is programmed with Wxwidgets, and for another, Boa Constructor uses this GUI framework to support users in their graphical development work.

Unfortunately, Boa Constructor just goes to prove some of the rumors about the Wxwidgets toolkit: The cluttered and old-fashioned interface typically leaves the user wondering where to go next. Large numbers of unlabeled, unintuitively designed, and all-too-similar icons are hidden away in innumerable tabs, forcing the user to mouse over to find out what they do.

The interface follows what is known as Multiple Top-Level Windows Interface Design (or MTI for short): In other words, there isn't a main window, but

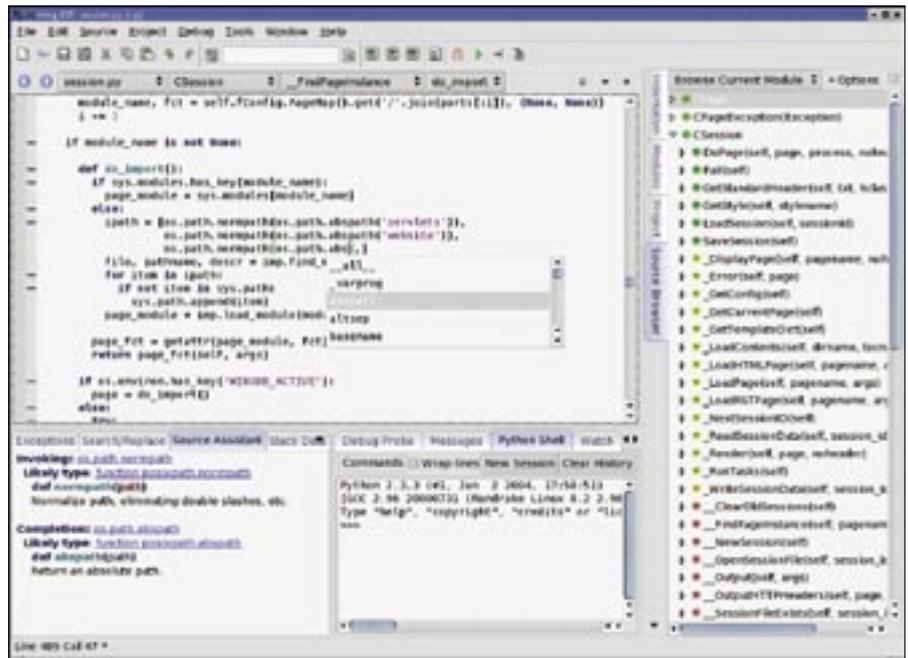


Figure 1: The Wing development environment comes in both Professional and Personal editions, but lacks some critical features.

the program components request space on the desktop directly from the window manager, without coordinating their requests. Again, this behavior is very much contrary to today's usability standards because it forces users to search for the right window time and again on crowded desktops.

Once you have become accustomed to the usability drawbacks, Boa Constructor rewards you with a robust IDE. The editor organizes your Python source code and only lacks code folding. Code completion is available at the press of a button (Ctrl + Space), and this includes custom-designed functions and classes.

If you really do want to develop Wxwidgets applications with Python, the Boa Constructor Frame Designer will be a big help. Although it will take you a while to find your way around the numerous icons, a nicely structured tutorial does shorten the learning curve.

The debugger has all the basic features a programmer needs for bug hunting – but again, with the restriction that searching for the right icons to control the debugger can be a nerve-wracking experience. By clicking with the mouse, you can set breakpoints in the editor; this lets you monitor variable values in debug mode.

Table 1: Python-IDEs

	Wing IDE Personal 2.1.3	Wing IDE Professional 2.1.3	Boa Constructor 0.4.4	Drpython 161	Eric 3.9.1	Idle 1.1.4
Supported programming languages	Python	Python	Python	Python, C++, HTML	Python, Ruby	Python
GUI development	–	–	Wxwidgets	–	Qt	–
Macros	–	✓	–	–	✓	–
Code folding	–	✓	–	As a plugin	✓	–
Code completion	✓	✓	✓	As a plugin	✓	–
CVS	–	✓	✓	–	✓	–
Subversion	–	✓	–	–	✓	–
License/price	Proprietary/ US\$ 35	Proprietary /US\$ 180	GPL	GPL	GPL	GPL
Languages	English, German	English, German	English	English	English, German, French, Russian	English
Other operating systems	Windows, Mac OS X	Windows, Mac OS X	Windows	Windows	Windows Mac OS X	Windows Mac OS X

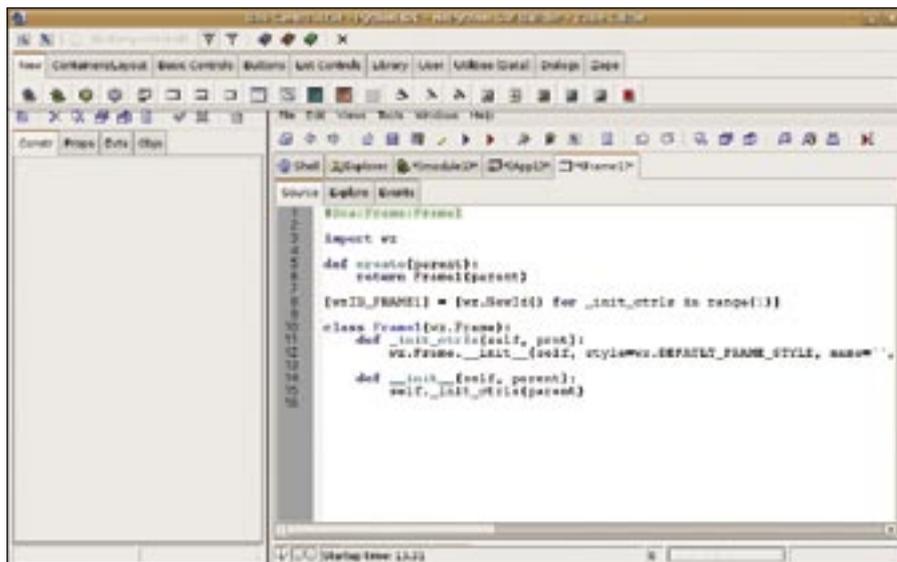


Figure 2: The cluttered Boa Constructor interface needs some getting used to. Good documentation provides an acceptable learning curve despite the complexity.

I did not experience any stability issues, contrary to my expectations from recent reports on the web, although abstruse error messages on launching and quitting the program made me wonder.

All told, Boa Constructor is a good choice for developing Wxpython applications because of its frame designer. Apart from this, Boa gives you an honest developer environment but without any show-stopping features that would warrant enduring the shortcomings of the graphical user interface.

► Drpython

Drpython [4] (Figure 3) relies on simplicity rather than feature scope. The program provides a simple editor with Python syntax highlighting and displays the modules, classes, and methods in the source browser. Like Boa Constructor, Drpython was written in Wxpython and thus provides cross-platform support. Drpython does not have any special tools for GUI development.

Bug-hunting help is limited to a simple syntax check in Drpython. The program lacks automatic code completion; this said, you can install a plugin to fill this gap. When launched, Drpython gives you a choice between beginner and expert mode; however, the differences are restricted to the options in the preferences dialog.

The program walks the line between editor and developer environment. If you already have a favorite editor for Python programming, moving to Drpython

just because of its source browser is hardly worth your while.

► Eric

Development work on Eric [5] (Figure 4) is currently continuing in two different branches: Eric3 relies on version 3 of the Qt interface, and Eric4 uses its successor. Apart from this, both versions have the same functionality. In a perfect example of the principles of free software, Eric combines existing programs to create a comprehensive developer environment. For example, the integrated editor uses

the Scintilla framework [6], and Eric simply launches Qt Designer [7] to design graphical interfaces.

Eric helps Python developers with a project management module. The Eric IDE also lets you define macros and run Pyunit for unit tests. Qt Designer helps programmers quickly create complex interfaces; however, the .ui files this approach generates are geared for use with C++ . Integrating them with PyQt takes some manual attention, and Eric does not give you any help with this.

If you decide to do without the assistance of Qt Designer, you will not find much in the way of support for interface programming. Eric uses wizards to automate a couple of standard dialogs in Qt.

Eric has the largest feature scope of the free Python developer environments, and it has a fairly self-explanatory user interface at the same time, which more or less makes up for the sparse documentation. This said, Eric is not really a big help for those of you interested in GUI programming. Eric does not make it easier for you to integrate Qt Designer-styled .ui files with your Python programs. As with any other IDE, if you want to program a Qt interface, you'll need to have some knowledge of Qt.

► Idle

Idle's [8] claim to fame is that it is the brainchild of Python's inventor, Guido

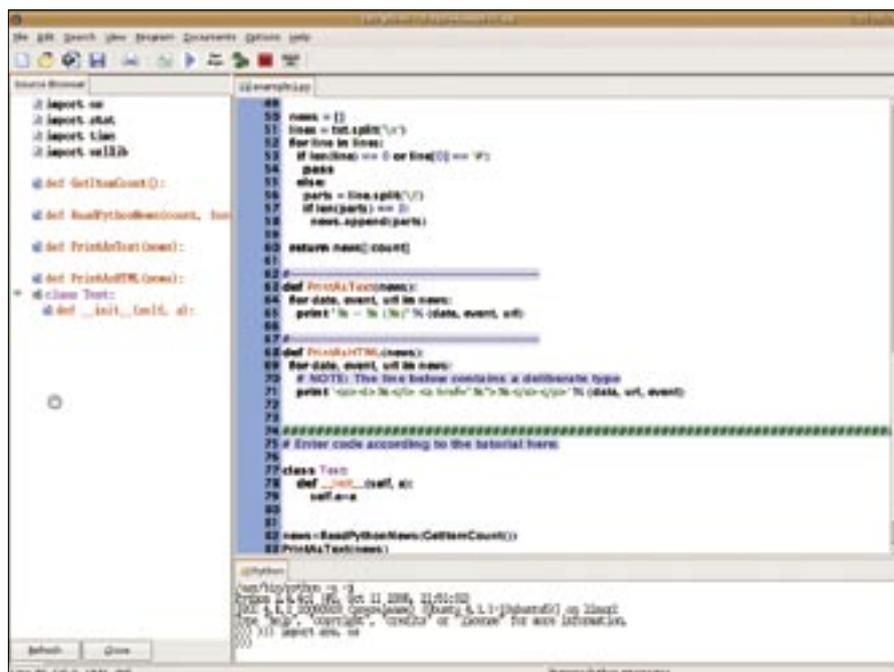


Figure 3: Drpython gives developers a clear view of their Python code, but it lacks a debugger and other tools.

van Rossum, in cooperation with various other Python gurus (Figure 5). Idle is part of the official Python distribution. Just like Drpython, it mainly provides a text editor with Python-specific syntax highlighting. The class browser (Figure 6) provides an overview of the functions and classes in the script along with their methods. The Path Browser is a useful feature that gives users the ability to browse packages and modules discovered in the Python path.

Idle does not have automated code completion, although the editor does display the signature of a known function once you have entered it.

The debugger is still under development, and it is restricted to basic functionality after launching a program in a Python shell. The Idle debugger currently doesn't let you monitor variable values. The Windows version still doesn't let users set breakpoints, and this makes the debugger more or less useless on the Windows operating system. Fortunately, this problem does not exist on Linux, and troubleshooting with Idle lived up to our expectations for the most part.

► Conclusions

It is probably because of the typical Linux user's love of spartan text editors that most free developer environments



Figure 5: Despite its status as the official Python IDE, Idle lacks some of the features found in alternative tools.

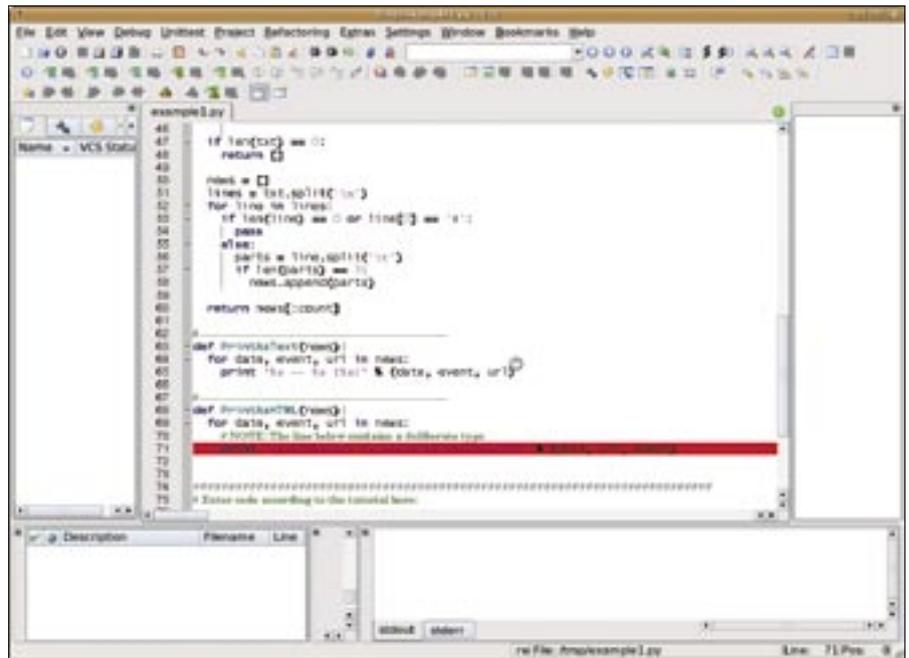


Figure 4: For interface design, Eric simply launches Qt Designer, which was mainly developed with C++ in mind.

have no more than basic functionality. The advantage of known commercial IDEs, especially from the world of Windows, compared with simple text editors, lies mainly in their ability to remove many manual steps from GUI development.

All the Python IDEs investigated here performed fairly badly in this respect, whether they are free or proprietary. Boa

Constructor is a tool to be taken seriously by Wxpython developers, assuming they are able to work around the usability issues. If you prefer to avoid the manual coding of GUI toolkits, you might have no alternative to tools such as Glade [9], Gazpacho [10] for GTK, or Qt Designer.

Wing IDE made the most positive impression of the field. Its controls are intuitive, and the Professional version supports cross-platform collaborative programming. The asking price, however, is way above what an average Linux home user would be willing to invest, al-

though it is quite normal for enterprise use. Thanks to the evaluation license, anybody interested in Wing IDE can form their own opinion before opting for either of the commercial variants. ■



Figure 6: Idle's class browser provides a neat overview of Python code.

INFO

- [1] Wing IDE: <http://wingware.com>
- [2] Boa Constructor: <http://boa-constructor.sourceforge.net>
- [3] Wxwidgets: <http://www.wxwidgets.org>
- [4] Drpython: <http://drpython.sourceforge.net>
- [5] Eric: <http://www.die-offenbachs.de/eric/index.html>
- [6] Scintilla: <http://www.scintilla.org>
- [7] Qt Designer: <http://www.trolltech.com/products/qt/features/designer>
- [8] Idle: <http://www.python.org/idle>
- [9] Glade: <http://glade.gnome.org>
- [10] Gazpacho: <http://gaspacho.sicem.biz>