Desktop Linux:

Where Art Thou?

**Catching up,
meeting new
challenges,
moving ahead**

BART DECREM, OPEN SOURCE APPLICATIONS FOUNDATION

Linux on the desktop has come a long way—and it's been a roller-coaster ride. At the height of the dot-com boom, around the time of Red Hat's initial public offering, people expected Linux to take off on the desktop in short order. A few years later, after the stock market crash and the failure of a couple of high-profile Linux companies, pundits were quick to proclaim the stillborn death of Linux on the desktop.

Then a funny thing happened: the GNOME and KDE desktop environments kept getting better, Mozilla and OpenOffice.org reached their 1.0 milestones, and a new batch of Linux desktop companies sprang up (Lindows, Xandros, Lycoris). Rumors of the death of Linux on the desktop had been greatly exaggerated. Over the past year, a steady stream of new developments and announcements has added to the desktop Linux drumbeat. Perhaps most important, big IT companies, including Novell and Sun Microsystems, have placed bets on desktop Linux.

So will 2004 be "the year for Linux on the desktop"? This article takes stock of where the Linux desktop stands and looks at remaining obstacles to broader adoption of Linux on the desktop, how Linux can meet the new

Desktop Linux: Where Art Thou?

challenges created by Microsoft's plans for Longhorn, and opportunities to get ahead of the curve around collaboration-centric computing.

LINUX'S LONG MARCH TO THE DESKTOP

The epic Long March in China lasted about a year, from 1934 to 1935, during which time Mao and his band of communist fighters traveled almost 3,000 miles through mountainous terrain and established their power structure. Here's the bad part: of the 200,000 fighters who started the march, only 30,000 or so made it to the end, and it would take them another 15 years to gain power.

We're now six years into the March of Linux on the desktop and there have been some setbacks along the way, the biggest one being the fading away of Netscape. While Netscape was not a Linux desktop company, Microsoft's control over 98 percent of the browser market continues to be one of the biggest threats, not just to the future of Linux on the desktop, but to the future of the Internet. Linux on the desktop, however, has been surprisingly resilient. The failures of companies such as Eazel proved to be mere speed bumps. Out of the ashes of Netscape rose a lizard whose roar is becoming louder by the day, with Mozilla's new Firefox browser getting rave reviews before even reaching its 1.0 release.

TAKING STOCK

Linux on the desktop is now good enough for certain classes of users, but much work remains to be done. Two large, dynamic open source projects, KDE and GNOME, offer complete desktop environments that include file managers, desktop administration tools, and a broad range of applications—from simple games to complete office productivity suites. Both of these projects are reasonably mature, offering fairly polished and stable desktops. They may not be as glitzy as Apple's Panther operating system, but they compare reasonably well with Microsoft Windows. The main problem with the Linux desktop is that there are multiple projects, making it hard for users and developers to pick one to focus on and diluting the energies of the open source developers.

Through their tools and development libraries, Micro-

soft and especially Apple provide a more or less unified look and feel and a dominant development environment. Not so for Linux, where KDE, GNOME, Mozilla, OpenOffice, and the Wine project all offer their own development environments, resulting in an often-inconsistent end-user environment.

A consistent end-user environment, however, may be less critical than a strong selection of end-user applications. After all, many of the leading Windows applications don't adhere to Microsoft's user interface guidelines. In most of the core application areas, there is at least one strong Linux offering. Multiple strong Web browser options are available; OpenOffice is a complete, mature office suite; and there are numerous strong Internet applications ranging from IM (instant messaging) to most of the popular multimedia plug-ins.

But significant gaps remain. While there are thousands of end-user applications, most are unpolished and incomplete. There is no end-user Linux software industry to speak of. There's no strong personal finance application for Linux. If you need pixel-for-pixel compatibility

The Long Linux March to the Desktop

1991	Unix X11 GUI framework ported to Linux
1998–1999	KDE 1.0 and GNOME 1.0, GUI environments for Linux
2001	Ximian Evolution 1.0, an Outlook look-alike that can be turned into a Microsoft Exchange client
2002	StarOffice 6.0/OpenOffice 1.0, real alternatives to Microsoft Office
2002	Mozilla 1.0, a competitive browser
2002	KDE 3.0 and GNOME 2.0. Linux GUIs become more polished.
2003	Munich switches to Linux. The first big desktop success stories emerge.
2003	Sun and Novell join the party. IT leaders throw their weight behind Linux on the desktop.

with PowerPoint, or need to tap into an older Exchange server, Linux may not be right for you. You won't find educational games for Linux lining the shelves of your local computer superstore.

Similarly, the Linux ecosystem has its strengths and weaknesses. There's a wide range of Linux distributions to suit anyone's needs, and a large, thriving community of open source developers each scratching their own itch. But while hundreds of white-box vendors are shipping Linux workstations, it's still all but impossible to purchase a Linux desktop PC in mainstream retail channels.

Three major obstacles mean that broad consumer adoption for Linux is years away, if it will happen at all:

1. The lack of a critical mass of consumer applications.

Linux lacks the types of applications that fill the aisles at your local Best Buy—everything ranging from personal finance software to games to polished photo album software. While the open source community fills some of these gaps, and the Wine project—an open source implementation of the Windows APIs on top of X and Unix—offers a partial solution, it's unlikely that independent software vendors will focus on the Linux market until demand is established, creating a chicken-or-egg trap. The gradual shift of the software industry to Web-based models may offer the best hope for overcoming this challenge—as long as these rich Web applications aren't inextricably tied to the Windows operating system.

2. The lack of major OEM support and consumer distribution channel access. While major OEMs such as Dell and software distributors have the most to gain if the Microsoft monopoly breaks down, they also have powerful financial incentives—in the form of joint marketing agreements with Microsoft—that make it almost impossible for these low-margin, commodity vendors to ship systems with alternative operating systems. Still, with the operating system and other Microsoft applications now making up the most expensive component in cheaper PCs, manufacturers have a major stake in advancing a lower-cost alternative.

3. Resistance to change. Computer users invest tremendous energy learning to operate their computers and configuring them to meet their needs. They build up piles of data that make them reluctant to upgrade to a new computer, much less to an unfamiliar operating system. Of course, the successive waves of computer viruses, worms, and Trojan horses have driven many users to despair and made them ready to explore safer alternatives.

While the consumer and knowledge-worker mar-

Desktop Linux: Making the Grade?

The Product

The desktop	B
Desktop developer platform	C-
Computer	B-
Peripheral devices	D
Applications	C+
Windows connectivity	A

The Ecosystem

Linux distributions	A-
Desktop platforms	A-
Developers	B
ISVs	D
OEMs	C
Distribution channels	D
Consortia	A
News and support	C

Linux on the desktop is good enough for certain classes of users, but much work remains to be done. While there's a thriving community of open source developers, there is still a dearth of commercial third-party applications, and distribution channels remain inaccessible.

kets, especially in the United States, remain elusive, many other markets offer significant opportunities for Linux on the desktop, including "transactional workers," price-sensitive markets outside of the United States, and cheap, limited-functionality PCs. These markets have the potential to provide desktop Linux with double-digit market share, making a serious dent in the current desktop monoculture. What's needed to seize the opportunities?

NEEDED: STRENGTHENED DESKTOP FOUNDATIONS

There are at least five Linux desktop platforms: KDE, GNOME, OpenOffice, Mozilla, and Wine. Each of these successful open source projects includes a vibrant community of developers, is an umbrella to a significant number of projects, and is built on a different toolkit. OpenOffice and Mozilla are cross-platform projects whose users mostly run the Windows operating system.

Much of the technical challenge ahead can be summed up, in the words of Red Hat's Havoc Pennington, as work that falls through the cracks between the existing open source projects or work that spans multiple projects.

Desktop Linux: Where Art Thou?

The latter problems are particularly intractable in the open source community, which is prone to endless heated technical arguments. Since code rules in the open source world and there is no one dictator or single business imperative to help resolve issues, integration issues often remain unresolved. As a result, perhaps the single most critical challenge to the success of Linux on the desktop may be the need to create stronger communication links among the various projects and, ultimately, a willingness for these projects to make painful concessions to improve interoperability and jointly deliver a more competitive desktop environment.

Let's take a look at some of the specific foundation tasks that need to be tackled to ensure that the Linux desktop becomes more competitive. (Thanks to Havoc Pennington for his input in this section.)

Improved desktop "plumbing." The X Windows system is the right foundation for the Linux desktop, but innovation is needed to increase performance and usability, offer better multimedia support in the face of closed hardware specifications and patents, and generally design and implement new X extensions that allow the Linux desktop to offer the kind of graphical pizzazz that Mac OS X and Microsoft's Longhorn provide.

Robustness and performance. You shouldn't have to know how to use the command line to maintain your Linux desktop system. Errors should be transparently handled or conveyed back to the UI (user interface) with help provided for the user. Performance on the Linux desktop needs to be consistently tracked as it is for the server, monitoring metrics such as startup time, opening a menu, or launching key applications.

Better support for peripheral devices. Linux needs a hardware abstraction library that makes peripheral devices available to applications and end users. Though the Linux kernel offers good support for USB and other peripheral device standards, that doesn't mean that you can just plug your digital camera into a Linux-powered desktop and edit pictures in your favorite image-editing application. Linux doesn't have a universal hardware abstraction layer that makes these peripheral devices available to all applications. Last July, Pennington wrote

a paper on this topic, entitled "Making Hardware Just Work."¹ Since then, a community volunteer has taken the lead on this and launched the HAL (Hardware Abstraction Layer) project.²

A universal MIME system. No single repository of associations between file types and helper applications exists on Linux. Each of the desktop platforms has its own MIME (multipurpose Internet mail extensions) system, which means that things quickly get confusing if you use Evolution (a GNOME-based PIM, or personal information management system) in KDE and are trying to figure out how to define, let's say, which Web browser should handle links in e-mail messages. The Shared MIME Database project³ is making headway in tackling this challenge, with the GNOME project starting to adopt its specification approach.

A universal configuration/manageability framework. Each Linux desktop platform has its own configuration and management framework. Worse, lower pieces of the operating system stack (such as the graphing layer, a.k.a. the X windowing system, or the operating system proper) also have configuration and management frameworks. If you're lucky, you'll be able to change your monitor configuration using the GNOME control panel. More likely, you'll have to log out and log back in. It may also involve a trip to the command line. This problem is complicated by the fact that projects such as the GNOME desktop run not just on Linux but also on FreeBSD, various flavors of Unix, and even Windows.

A common end-user document filing system. The various desktop projects and Linux distributions don't always share the same filing-system conventions for deciding where to place files that go on the desktop or in a user's home directory. The GNOME project has adopted KDE's convention for the user documents directory, but other projects and various Linux desktop distributions have their own conventions; so it's quite likely that your Web browser will save documents to the desktop by default, while your word processor may save them to your home directory, and a third application may save files to a My Documents directory.

A universal virtual file system. The list of available

file systems may differ from one application to another. On Linux, just because you are able to mount WebDAV (Web-based distributed authoring and versioning) or FTP volumes on the desktop doesn't mean that you'll be able to access those volumes from your word processor. Piece-meal progress is being made in this area—for example, Mozilla recently included support for the GNOME project's GNOME-VFS (virtual file system).

A process for interaction and UI design. The GNOME project has invested the most effort in creating human interface guidelines and encouraging the maintainers of GNOME applications to follow these guidelines. But what about user interface consistency among, for example, GNOME, KDE, OpenOffice, and Mozilla?

THE LAST MILE

Tackling these and other remaining challenges to offering an integrated Linux desktop that is truly competitive with existing offerings in the marketplace won't be easy, and any successful approach will include an effort that brings together people from the disparate desktop-related projects. The good news is that the open source community is moving in exactly this direction, and two projects have recently broadened their charter to tackle these challenges: the X Consortium has morphed into the X.org Foundation, and the Freedesktop.org project now hosts a growing number of desktop foundation-related projects, including several mentioned here.

In the productivity suite arena, there is reason to be hopeful as well. The OpenOffice document filters are already quite good and continue to get better (witness the improvements made in OpenOffice 1.1 and StarOffice 7). Developers at OpenOffice, KWord, and other open source productivity applications are already collaborating informally to solve remaining problems. Improved, collaboratively maintained documentation by the open source community of the Microsoft Office DTDs (document type definitions), similar to the OpenOffice project's documentation of the Excel file format⁴ could further improve this situation.

The availability of a rich, open and XML-based file format standard for productivity applications will also be a significant boost for Linux desktop adoption. An effort is under way at the OASIS standards consortium to develop a formal standard,

largely based on the OpenOffice file format. Development, adoption, and implementation of the standard by OpenOffice and other Linux productivity applications will offer librarians, data archivists, policy makers, and other interested parties an open alternative that they can adopt as a standard for document exchange in government, educational institutions, and elsewhere. However, there is more to do:

More commercial end-user applications for Linux. As already indicated, the lack of a critical mass of commercial end-user applications is a major barrier to broader consumer adoption of Linux. This chicken-or-egg problem is unlikely to be resolved within the next few years, but the Wine project may provide a partial answer. Thanks to the efforts of the Wine community and CodeWeavers, many Windows applications, including Microsoft Office, Internet Explorer, Windows Media Player, Quicktime, and Adobe Photoshop, already run unmodified under Linux. TransGaming, the other company that builds on the Wine libraries, boasts that it allows 250 of the world's hottest games, including EverQuest, Battlefield 1942, and SimCity, to run unmodified under Linux. Additionally, the Wine libraries offer an easy way for ISVs (independent software vendors) to port applications to Linux. For instance, a few years ago MusicMatch was ported to Linux using Wine. In the not-so-distant future, further improvements to Wine may allow end-user Linux distributions to credibly claim that Linux runs Windows applications.

User-friendly documentation and support options. Most Linux documentation and the most popular mailing lists and support forums are hard to parse for nontechnical users, making it a daunting task to solve the inevitable problems that arise.

Absence of a standard Linux. The multitude of Linux distributions and hardware platforms supported by Linux makes it hard for ISVs to package applications for Linux. Efforts such as the Linux Standards Base and easy-to-use software installers such as Linux's Click-N-Run Warehouse help to address this challenge.

Brian Proffitt, managing editor of *Linux Today*, gives a great example of the end-user cost of this lack of a standard definition of Linux.⁵ The GNOME desktop boasts beautifully anti-aliased fonts, so Proffitt was horrified when he installed the latest version of the Mozilla browser on his

The lack of a critical mass of commercial end-user applications is a major barrier to broader consumer adoption of Linux.

Desktop Linux: Where Art Thou?

Red Hat system and ended up with jagged fonts. It turns out that anti-aliased fonts in GNOME require two libraries that don't come pre-installed on all the major Linux distributions, which is why XFT is not pre-compiled in every release of Mozilla.

Better support for developers. Defining a standard Linux specification would not only benefit users, but would also be a boon for developers who wish to build Linux desktop applications. But more is needed. The Linux desktop projects are still best suited for programmers who code in C or C++ and who think of Emacs as an IDE (integrated development environment), making the Linux desktop hard to approach for the millions of Java and Visual Basic programmers who program in Visual Studio. Offering primarily C and C++ strongly limits the pool of developers. Open source implementations of Java, C#, Python, and the other open source “p” languages offer great promise, but offering developers the easy-to-use tools that Microsoft provides continues to be a challenge for the Linux community.

LONGHORN: THE NEXT CHALLENGE

This technical agenda for Linux, ambitious as it may be, is really a catch-up plan that will allow the Linux desktop to make headway in areas where the current state of the desktop is *good enough* and decision makers focus mostly on cost issues, such as transactional workers, cost-sensitive markets outside of the United States, and consumers looking for limited-functionality Internet PCs. But Microsoft is planning to raise the bar considerably with its upcoming Longhorn operating system.

Perhaps most significantly for the prospects of Linux on the desktop, Longhorn will come with an XML mark-up language called XAML that allows developers to build rich Internet applications that are seamlessly integrated into the desktop, potentially relegating the Web browser to a legacy system and inextricably integrating the open APIs of the Web with the Longhorn APIs. If XAML takes off, users may be able to interact with Internet content in rich new ways, but could become more locked in to their Windows desktops than ever before.

But XAML didn't pop up out of nowhere. The lan-

guage bears some striking similarities to Mozilla's XUL (XML User Interface Language). XUL is a UI mark-up language that can be used to build Gecko-based applications (Gecko is the Mozilla rendering engine). The Mozilla 1.x suite, Firefox, and Thunderbird are all XUL applications. More intriguingly, there are a number of stand-alone XUL applications, such as the Mozilla Amazon Browser,⁶ which runs as a client application that lets you locally manipulate Amazon data without the latencies of Web page updating—just like the XAML Amazon demo shown to great fanfare at a recent Microsoft developer conference. The difference, of course, is that XUL is available now and will run on any platform.

The Mozilla people believe that the combination of XUL, the Mozilla Gecko rendering engine, SVG (Scalable Vector Graphics) support, and Web services capabilities are the keys to delivering rich Internet applications to the desktop, especially because XUL is relatively easy to learn for the millions of HTML coders. Mozilla developers are building prototype applications that combine the benefits of Web-based application deployment and management with the responsiveness and usability of desktop applications.

In sum, XUL has the potential to be a cross-platform, open answer to XAML. If the Mozilla developers work closely with Linux desktop efforts such as GTK (the toolkit used by the GNOME project) to combine the cross-platform strengths of XUL with native widget support on Linux, Linux could offer rich Internet applications that are every bit as compelling as XAML applications and offer the benefit of running on any platform.

MOVING AHEAD

The open source community has a plan for catching up on the desktop and may be able to best Microsoft in its plans for bringing rich Internet content to the desktop. But how can the open source community deliver on the original promise of the PC as “bicycles for the mind,” as Steve Jobs of Apple famously said, or the promise of the Internet as a creative enabler? Can the open source community articulate its own vision for tapping into the creative power enabled by combining a rich client envi-

ronment with the network effects of the Internet? Are there areas of opportunity that the dominant provider isn't pursuing because they don't fit into its strategy for maximizing shareholder return on investment?

Let's take a look at three projects that aim to tackle this larger challenge.

CHANDLER

Today's PIMs force users to adapt to technology. Information is locked into silos such as e-mail, calendar entries, address book items, and to-dos.

Frustrated by this inability of existing PIMs to adapt to his way of organizing his work, Mitch Kapor, the creator of Lotus 123, turned to the open source development process. He invested millions of his own money and created the Open Source Applications Foundation,⁷ where a team of two dozen developers is creating Chandler, a next-generation PIM. Slated to be released sometime next year, Chandler breaks down the walls between the various PIM applications, as well as the wall between what's mine and what's yours. All data is organized into one central repository (an idea Microsoft is picking up on in its plans for WinFX in Longhorn), and users can organize their information however they want. The calendar becomes just another way to view appointment data—you could just as easily view your data according to which project it relates to. The other major emphasis in Chandler is collaboration: the ability to share information using flexible policies. User and presence management and notifications are handled not through central servers but by using the peer-to-peer Jabber IM platform.

"Both the PC and the Internet revolution started off as decentralizing forces, putting power with the end user," notes Kapor, "but were then embraced by enterprises, which gained control over them. Software investment in general, including on Linux, is largely driven by the needs of large enterprises, who are mainly interested in managing centralized computing resources, enterprise-wide application development, limiting the ability of the user to modify their desktop environment. The end-user computing experience matters less to them."⁸

But the command-and-control approach taken by many large enterprises, and around which the business models of the large software vendors are built, is at odds with the increasing interconnection enabled by the Internet. People are working together on projects in ad hoc teams that transcend organizational boundaries. Microsoft Exchange is designed to work within the confines of a large enterprise, but what if you're a small business owner who works with a fluid group of people dispersed

around the world on a variety of different projects? Even within large organizations, bottom-up and ad hoc collaboration within the enterprise and across the corporate firewall is becoming increasingly important. That's where Chandler comes in: "A big part of Chandler's mission is to help small groups of people collaborate across organizational boundaries," says Kapor.⁹

THE DASHBOARD

Nat Friedman, founder of Ximian (recently acquired by Novell), also believes that the future of the Linux desktop lies in collaboration, and he's putting his money *and* his fingers where his mouth is. Recently, he launched Desktop Integration Bounty Hunt¹⁰ for the GNOME project, offering bounties for making more than 50 improvements to GNOME aimed at improving the experience of collaboration in the desktop environment.

Friedman also coordinates the GNOME Dashboard, a project that aims to bring relevant information to users, rather than requiring people to dig around for information. If you're chatting with a friend using IM, and your friend asks you about your recent trip to Rio, the Dashboard will helpfully figure out that you were assembling your pictures from that trip last night, dig up your shots, and allow you to drag and drop them into your chat software.

**People are working
together on projects
in ad hoc teams that transcend
organizational boundaries.**

"While you read e-mail, browse the Web, write a document, or talk to your friends on IM, the Dashboard does its best to proactively find objects that are relevant to your current activity, and to display them in a friendly way, saving you from digging around through your stuff like a disorganized filing clerk,"¹¹ as Friedman puts it.

The Dashboard is not just a single piece of software you can install. It requires some modifications to front-end applications such as the Web browser, PIM software, chat software, etc., to allow them to send "clue packets" to the dashboard. So volunteers in the GNOME project are now retrofitting GNOME applications to support the Dashboard concept. For Longhorn, Microsoft is working on a similar concept, which it calls "Implicit Query."¹²

Desktop Linux: Where Art Thou?

MOZILLA

Written off by many just a few years ago, the newly established Mozilla Foundation is not just thinking about the future of XUL, the organization is also getting ready to roll out its much-anticipated next-generation Web browser and e-mail client (Firefox and Thunderbird, respectively), slated to reach 1.0 this summer.

Looking further ahead, leaders in the Mozilla project are exploring ways to give users easier access to all the information they care about—whether it's a Web page they have visited before, a bookmark, an e-mail message, or an attachment they have filed away—using a personal data server that offers Google-like search of both local and remote files. The Mozilla Foundation is also interested in tapping into friend-of-a-friend small-world networks to enable sharing of bookmarks and Web page annotations, for example. An overview of some of the ideas the Mozilla Foundation is exploring can be found on its Web site.¹³

What do Chandler, the Dashboard, and Mozilla have in common? They share a focus on integrating the Internet into a rich client environment in innovative ways. They are breaking information out of its traditional silos. They are working to build applications that adapt to the user's needs, rather than the other way around. They bring information closer to the user and peer-to-peer communication and collaboration. In sum, these projects tap into the power of the desktop while also understanding that a network application framework, which opens up new possibilities for users and offers important cost advantages for developing, deploying, and maintaining software, is a key to success.

Most importantly, they move beyond trying to copy what Microsoft did a few years ago. Each of these projects is breaking new ground and is seeing some of its ideas copied by Microsoft, putting to shame the claim by Microsoft's operating system chief Jim Allchin that open source is a threat to innovation.¹⁴

What's next? These ideas feeding off each other and linking up with Web innovations such as wikis, RSS (rich site summary) and blogs, and social networks. Stay tuned. The open source community is bringing innovation back to the desktop. Q

REFERENCES

1. Pennington, H. Making Hardware Just Work (2003); <http://www.ometer.com/hardware.html>.
2. HAL: <http://pdx.freedesktop.org/Software/hal>.
3. Shared MIME Database project: <http://www.freedesktop.org/Standards/shared-mime-info-spec>.
4. For more on the OpenOffice project's documentation of the Excel file format, see: <http://sc.openoffice.org/excelfileformat.pdf>.
5. Proffitt, B. Moving Toward a Clearer Desktop Goal, Linux Today (Jan. 30, 2004); <http://linuxtoday.com/mailprint.php3?action=pv&tsn=2004-01-30-023-26-OP-DT-SW>.
6. Mozilla Amazon Browser: <http://mab.mozdev.org>.
7. Open Source Applications Foundation: see <http://www.osafoundation.org/>.
8. Kapor, M. In discussion with the author, March 4, 2004.
9. See reference 8.
10. Desktop Integration Bounty Hunt: <http://www.gnome.org/bounties/>.
11. Friedman, N. The Dashboard; <http://www.nat.org/dashboard/>.
12. Kanellos, M. Microsoft Aims for Search on Its Own Terms. CNET News (Nov. 24, 2003); http://news.com.com/2100-1008_3-5110910.html.
13. Mozilla Foundation: <http://www.mozilla.org/events/dev-day-feb-2004/mozilla-futures/>.
14. Allchin, J. Open-source Advocates Answer Microsoft. CNET News (May 15, 2001); <http://216.239.57.104/search?q=cache:0H4pTuAthAUJ:news.cnet.com/news/0-1003>

LOVE IT, HATE IT? LET US KNOW

feedback@acmqueue.com or www.acmqueue.com/forums

BART DECREM (<http://www.decrem.com/bart/>) works for the Open Source Applications Foundation and currently spends most of his time helping the Mozilla Foundation with marketing and business affairs.

© 2004 ACM 1542-7730/04/0500 \$5.00