Architecting the Linux Desktop

Havoc Pennington

Red Hat, Inc. | freedesktop.org | GNOME

Me

- Started work on the Linux desktop around 1997
- Debian and GNOME developer initially
- Implemented many pieces of GTK+ and GNOME, including “gless”
- Joined Red Hat Advanced Development Labs in 1999
- Founded freedesktop.org in 2001
- Chair of GNOME Foundation board 2001, 2002 and steering committee prior to that
- RHAD Labs later became the desktop development team
- Currently lead desktop developer at Red Hat
- Hoping to escape management
- Today I'm not speaking for anyone but myself

Agenda

- Shameless collection of personal opinions
- Where does the Linux desktop stand?
- What do we have to do to succeed?
  - Where do we need to be good enough to match the competition?
  - Where do we have a chance to be better and gain a competitive edge?

Defining Success

- To provide a complete desktop solution using free software
- The more people using this solution, the more successful we are

Why the Desktop?

- Necessary for server to remain viable
- Brings open source to more people and more applications
- Fun, challenge, and profit

Why More People?

- Can't we just build a desktop for geeks?
- Marketshare is crucial to viability
  - Otherwise Microsoft dictates all the standards
  - Otherwise OEMs and ISVs don't make it go
  - Otherwise there aren't enough development resources
- It's the right thing to do

Desktop is Now

- Technology is reaching maturity milestones
- Historically commercial investment in the Linux desktop has been weak relative to the server; that is changing on several fronts
- Strong customer interest from real enterprises with real money
- Some initial large deployments in government and education are in progress and providing feedback loops
    - Brazil, Spain, China, numerous universities, and many others
- Longhorn thoroughly delayed, leaving a nice window of opportunity

When are we “Ready for the Desktop”?

- Now: many people are using it successfully
- Three years: we'll be much improved and appeal to more people
- Never: we always have room to be better and have broader appeal
- Some improved questions:
  - How will we get the next increment of desktop marketshare?
  - How will we enable the next class of people to use the Linux desktop?
  - When will we reach 2%, 3%, 5%, 10% marketshare?
  - What areas of the world are adopting Linux on the desktop?
  - How can we be not only “ready” (as good), but better?

How Long Will It Take?

- The road to here has been extremely long
  - 1984: RMS founds GNU Project
  - 1991: Linux kernel begun
  - 1996: GTK+ released, KDE launched
  - 1997: GNOME launched
  - 1999: Linux IPOs
  - 2003: Success on server seems almost assured
- Server has “primed” the desktop: Linux businesses and core OS are much more mature, customer awareness is high
- Is there a single moment where we can say we've succeeded?
- Where will we stand at Longhorn release time?
- Software development is a long-term process, not a task that ends

Some Microsoft Threats

- XAML: Windows-specific Web
- .NET vs. Java
- Patents to reserve important features for Windows only
- Lock-in strategies everywhere:
  - Microsoft Office formats
  - Windows media format
  - Exchange, Active Directory
- Billions of dollars in feature enhancements
- Aggressive business tactics
- More we can't guess

Challenges and Prerequisites

Modernize Graphics

- X Window System is the right basis
- Important for:
  - Usability – implementation of particular features
  - Credibility – we don't want to look like twm relative to state of the art
- Technical challenges:
  - Design and implement new X extensions
  - Make it go fast
- Organizational challenges:
  - X community needs work
  - Coordinate kernel on one side and desktop projects on the other
- Political challenges:
  - Closed hardware specifications
  - Patents

Robustness and Performance

- How often do you have to fix your machine via command line?
- Smart handling of issues such as disk space exhaustion
- All errors must be checked, transparently handled or conveyed back to the UI with useful help for the user. Including errors in the kernel and system scripts/daemons.
- Desktop performance metrics
  - Opening a menu
  - Launching key applications
  - Opening a file manager window with N files
  - Switching workspaces
  - Power switch to completion of login
  - Power switch to power down
- Performance needs to be consistently tracked as it is for the server

Interoperability and Migration Path

- It isn't feasible to convert an entire company to a new OS in one go
- Linux should work:
  - As a client with Microsoft servers – Exchange, Active Directory, File/print
  - As a server with Microsoft clients
  - With Microsoft file formats
  - With nearly all web pages
- Multiple solutions to run Windows applications:
  - WINE/Mono
  - Citrix
  - VMWare
  - Open source alternatives?

Hardware Support

- Requires OEM involvement; otherwise the lag is unacceptable
- Missing *infrastructure* for connecting hardware to the user interface
- Hardware Abstraction Layer:
  - http://www.freedesktop.org/Software/hal
- More important for “home users” than companies with standard platforms, though laptops are an intermediate case

ISV and OEM Support

- Software and hardware must be synced with open source desktop products
- Slow going due to fear of Microsoft and small marketshare

Multimedia

- Adequate media support still missing:
  - Primarily a patent problem: the code exists, but isn't usable, at least in many countries
  - Further legal challenges with DVDs, DRM, and so forth
  - Likely to be solved via proprietary software, at least in the short term
- Another danger area for Microsoft monopoly
- Surprising level of interest in open formats such as Ogg

Usability

- Already, several studies have shown that users train on Linux as easily as a Windows upgrade
- However, we should have aggressively *great* usability
- Process and methodology for consistent results
  - Sound principles applied by smart people
  - Empirical research and testing
  - Documentation to educate developers
  - Funding of usability labs and such will help
- Establish a meritocracy as we have for code quality; not all opinions are equal
- Not writing the software for ourselves

Reasonable Programming Languages

- Far more Java and VB programmers than C/C++ programmers
- Implementing the desktop in C/C++ primarily is a time sink
- More importantly, offering only C/C++ to developers strongly limits the pool of developers
- Interesting technologies:
  - Java (especially in open source implementations)
  - C#
  - Python and other open source P-languages
- Keeping these languages as “first class citizens” is a lot of work

Clarity and Documentation of Platform

- freedesktop.org is trying to build a base desktop platform based on specifications and agreed on by the bulk of the developer community
- GNOME/KDE split has to be adequately addressed, one way or another
- ISV interfaces aren't just library ABIs; also installing menu items, installing MIME handlers, ~/Desktop directory, and so forth
- All of this will need to be locked down and kept working
- Some elements of the platform are too immature; kioslave/gnome-vfs for example
- Platforms are a means, not an end; don't get obsessed

The Whole Network

- Desktop is not just the pixels on the client side:
  - Directory services
  - Mail server
  - Shared calendar
  - Instant messaging
  - File and printer sharing
  - Management and monitoring
  - Authentication and single-sign-on
- Linking open source efforts in all these areas will be tricky
- Both more and less commercial support than the client:
  - More because Linux on the server is already accepted
  - Less because software in these areas often competes with existing products

Opportunities to Be the Best

Open Source

- Presumably you're familiar with it
- At the root of many of the advantages I'll mention
- The one strategy Microsoft can't buy
- To preserve this advantage, all critical elements of the desktop must be open source; not only the kernel

More Opportunities for More People

- Open source software lacks vendor lock-in
- Many companies, from tiny to huge, can build a business around customization, support, development, and more
- A level playing field
- To grow relentlessly, Microsoft competes with and tries to lock out nearly everyone in the industry

Internationalization

- Many regions are simply too small for proprietary vendors to notice them
- With Linux, they can solve their own problems
  - (Large desktop examples in regions of Spain, Brazil, and China)
- Both technical and pricing/support problems can be addressed in this way
- Open source software is available for an amazing number of languages

Accessibility

- As with internationalization, a niche market can address its own problems
- Demonstrates the more scalable nature of open source development
- Linux accessibility is already better than Windows in many ways, and getting better
- An important edge in government sales

Consistent Single Platform

- Same operating system APIs and even binaries can be used:
  - Handheld to desktop to server
  - Every CPU architecture of interest
- Minimizes costs:
  - Training
  - Porting
  - Management
  - Choice of hardware vendors
- Ready to take advantage of new hardware technologies

Price

- Open source systems are cheaper
- Customer has control over when and how to spend their money
- Indirect savings by avoiding bundling:
  - Of server side with client side
  - Of software with hardware
  - Of software with protocols and document formats

Open Data Formats

- Keep your data free and future-safe:
  - Ogg media formats
  - OpenOffice.org office formats
  - SVG
  - ...
- Compelling advantages to users, if they are thinking a few years ahead

Data Access Methods

- Can we build an open standard addressing some of the same problems as WinFS?
- Search technology – Nutch.org? Medusa? Storage?
- Synchronizing data between users and machines
- Standard data stores for:
  - Address book
  - Email
  - Documents
  - Music and Movies
- What would it take to have standard APIs and protocols in these areas?
- Could we promote an open solution over WinFS?

Interoperability and Open Protocols

- Open source projects have a genuine, vs. “bullet point,” commitment to open standards
- At the same time, there's no vendor agenda to prevent interoperation with all proprietary standards
- We should not be afraid to extend the extensible – X protocol, LDAP, whatever it may be – to improve the performance of open source solutions when used together

Developers

- Open source is compelling for developers:
  - Ability to identify and correct problems
  - Ability to compensate for missing documentation
  - Ability to customize the software
  - Ability to join the community and help move forward
- Level playing field for all sorts of technologies:
  - Many application servers
  - Many programming languages
  - Many libraries
  - Domain-specific software (finance, medical, etc.)
  - Single vendor can't cover all niches

Security

- The “many eyes” principle
- Linux is a multiuser, secure system from the start
- Enhancements such as SELinux
- Can we split the desktop into multiple security contexts?
- Encrypted filesystems?
- Stronger authentication methods?
- More structured auditing?

Manageability

- Flexibility to run in many configurations:
  - Stateless thin client (VNC, SunRay)
  - X terminals
  - Diskless workstations
  - Nightly-reinstalled workstations
  - NFS/AFS home directories
  - Directory services or NIS
- Customization allows infinitely variable lockdown and control
- Many vendors can compete to offer management tools
- Command line exists for the old fashioned approach

Open Collaboration Platform

- Combine various advantages already mentioned:
  - Open data formats
  - Open protocols
  - Consistency across architectures and devices
  - Security
- Add integration and user interface for shared calendar, whiteboard, files, instant messaging, and more
- Potential for compelling features enabling people to work together efficiently
- Selfish motive: snazzy replacement for IRC

Some Final Thoughts

More Users Means More Developers

- The more users a project has, the more developers seem to work on it
- We only have 1-2% of desktop users at the moment
- Look at the current pace of Linux development
- Of course, more users also means more bugs and features...

Commercial Involvement Will Grow

- Companies have contributed relatively little to the desktop so far; perhaps a tenth of what's been contributed on the server
- This will raise organizational and political challenges for the developer community
- We have to keep projects healthy, friendly, and unfragmented
- At the same time, more developers (often) get more done

Longhorn Will Arrive Soon

- It's hugely delayed, but time flies:
  - Perhaps 3 GNOME releases until then, at 6 month intervals
  - Perhaps kernel 2.6 is it, 2.8 at most
  - Perhaps OpenOffice.org 2.0 is current when Longhorn arrives
- Fighting Windows XP is a losing strategy; need to jump directly to addressing Microsoft's latest tactics
- Cloning Windows is not always the right approach; focus on our strengths
- Upgrade to Longhorn is nontrivial and customers could take many years to do it; we can offer an alternative upgrade

Open Source is the Way

- A proprietary desktop alternative is simply BeOS or Mac OS X; cool, but not serious competition for Windows
- An open source solution with pervasive proprietary components loses the advantages
- Open source changes the rules of the game so Microsoft can't follow

Aim High

- It's not guaranteed to work
- It will be years of effort
- The potential benefits are huge
- Individuals can have a huge impact

Questions?