



Open Source Software: Risks and Rewards

ECAR Symposium

Tuesday – 16 November 2004

Gary Hein
VP and Service Director
Application Platform Strategies

ghein@burtongroup.com

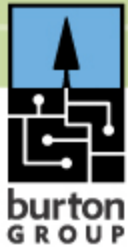
www.burtongroup.com

Burton Group Background

2



- **Headquarters in Salt Lake City since 1990**
 - Unmatched technical depth, focus
 - Only research firm focused solely on technology infrastructure
 - Foremost research firm on directories, networks, identity management, web services, security, IP telephony, wireless LANs, SOA, application platforms
 - 100s of Fortune 500 consulting engagements
- **Top industry analysts**
 - **Directory and Security Strategies:** Jamie Lewis, Dan Blum, Phil Schacter, Fred Cohen, Gerry Gebel, Trent Henry, Mike Neuenschwander
 - **Network and Telecom Strategies:** Dave Passmore, Mike Disabato, Eric Siegel, Bill Terrill
 - **Application Platform Strategies:** Anne Thomas Manes, Gary Hein, Jim Kobielus, Peter O'Kelly, Richard Monson-Haefel
- **Vendor Neutral**
 - No paid research, no product/vendor endorsements



Open Source Software

3

Overview

- Open source software (OSS) is not a fad; it's here to stay
- OSS will provide additional choices and may reduce costs in software development and IT deployments, but it is not free
- OSS is changing your vendors; be aware of who will win and lose, and what your future vendor relationships will look like



Agenda

- What is open source software?
- What are the benefits and risks?
- Market assessment and vendor revenue models
- Recommendations



Open Source Software

5

Agenda

- *What is open source software?*
- What are the benefits and risks?
- Market assessment and revenue models
- Recommendations



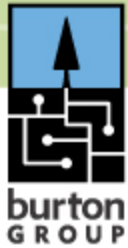
What is Open Source Software?

Open source software is more than just “free”

- Open source is:
 - Freedom for anyone to acquire source code, inspect, modify, use, and create derived works
 - Integrity of author’s source code, derived works, and license
 - No discrimination against fields of endeavor, persons or groups, technology, or other software
- But it’s more than just source code...
 - A powerful community of altruistic individuals
 - A set of software licenses
 - A software development model
 - A catalyst for new businesses and new business models
 - A force that is accelerating software commoditization

What is Open Source Software?

7



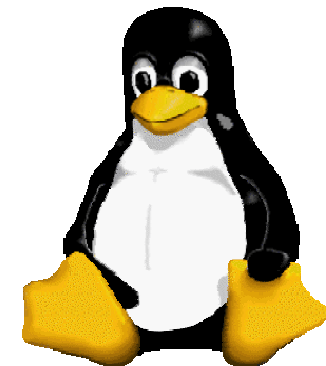
Shibboleth®



SOURCEFORGE®

Over 90,000 open source projects with 950,000 registered users

www.sourceforge.net



The **Apache Software Foundation**

<http://www.apache.org/>



What is Open Source Software?

Three basic open source software licenses

- GPL – GNU Public License
 - Original work and any derived works must remain free
 - Great for developers who wish to prevent inclusion in a proprietary or commercial product
- LGPL – Lesser (Library) General Public License
 - Internal code is GPL, but software that interacts through published interfaces is not considered a derived work
 - Permits use of LGPL libraries in proprietary or commercial products
- MIT/BSD/Apache
 - Enables full access with few, if any, restrictions on derived works
- Most other licenses are derived from these three basic licenses
 - IBM, Intel, Mozilla, Apple, Nokia, Sun, Lucent, and others have license derivatives that provide additional patent, encryption, or other protections



Open Source Software

9

Agenda

- What is open source software?
- *What are the benefits and risks?*
- Market assessment and revenue models
- Recommendations



Open Source Software Benefits

10

Freedom from vendor initiatives

- Software cost – free*
 - Anyone can build their own binaries from source code
- Not bound by vendor initiatives
 - Desktop churn, application churn, upgrade requirement to get new features or hardware support
- No usage restrictions
 - Maximum CPUs, network connections or type, data size
 - Internal or external users
 - Length of software license
- Enables best-of-breed solutions
 - Pull from a multitude of projects and customize as necessary



Open Source Software Benefits

11

Freedom to control destiny

- Freedom to inspect and verify
 - Intentional security issues like backdoors
 - Unintentional security issues like buffer overflows or weak encryption
 - Can verify that source code = binary code
- Freedom to modify, customize, enhance, or repair
- Open projects allow anyone to influence, drive, or lead
 - "Meritocracy" – ability and code contributions drive influence
- And if you don't like it, fork it
 - Often considered a last result, but it is an inherent open source freedom



Open Source Software Risks

12

Not quite as free as you might believe...

- "Free as in Free Puppy"
 - Software is typically <5% of the total cost of ownership of most projects
 - Still requires training, service, and support, which may cost more than the commercial product it's replacing
 - New business models charge for patches, updates, and upgrades
- Frequent updates
 - Projects may be updated frequently – as often as every day
 - Rapid updates impose implementation, interoperability, and support challenges
 - Some progress with enterprise-friendly release cycles and early code access

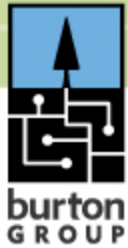


Open Source Software Risks

13

Not quite as free as you might believe...

- Less accountability for backwards compatibility
 - Mitigated by access to prior source code, but that may impose additional in-house development and support requirements
 - Can seek out commercial alternatives to buffer enterprise world from open source communities
- Support
 - Perceived as the biggest issue with open source software deployments
 - Internal? External? Accountability? Service Level Agreements?
 - Requires greater involvement and different skill set from operations staff
 - Good news – major vendors now providing strong certification and support
 - Red Hat and Novell – Linux and popular open source packages
 - IBM – all 5 product lines now support Linux



Open Source Software Risks

14

Business and legal risks

- Immaturity leads to incompatibility
 - Linux Standard Base (LSB) and Internalization (L18N)
 - Initiatives such as OSDL's Carrier Grade Linux
- Some uncertainty in every project
 - Community disputes may impact project viability and future
 - Example: JBoss developers split from JBoss.org, XFree86 and X.org
 - Sun's mixed support of open source (Solaris, Java, and Apache)
 - Open source projects may not stay open in future releases (SourceForge)
- Potential legal risks
 - Threats from SCO regarding Linux copyright infringement
 - Open source licenses do not provide indemnification
 - Vendors are addressing this issue, but read the fine print
 - Open source licenses remain untested in the world's courts
 - Can open source software deployments pass an audit?



Open Source Software Risks

15

GPL-specific development concerns

- Software that's derived from an existing GPL project, in whole or in part, must be released under the same license
- Software that's an aggregation of GPL and non-GPL software might qualify if:
 - Static linking of a GPL library
 - A GPL Perl module or Java class
 - Dynamic linking is a grey area
 - Programs that make function calls to each other and share data structures
 - A proprietary program that depends on GPL software
 - May use separate EULA, download, and install if not required for product to function
- Developer contamination
 - Developer reviews an existing open source project, sees an elegant solution, then implement a similar function in closed source project
 - How do you de-contaminate a developer?
 - May introduce HR and/or software development procedural changes
- Read <http://www.gnu.org/licenses/gpl-faq.html>



Agenda

- What is open source software?
- What are the benefits and risks?
- *Market assessment and revenue models*
- Recommendations

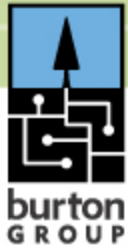


Market Analysis

Open Source software is not a fad...

- 54% of IT executives said open source would be their dominant server platform in 5 years (CIO Magazine)
- Apache has 67% of Web Server market (Netcraft 10/04 survey)
- Linux-based servers account for 26% of servers, a 39.5 percent grow in 2003 (IDC)
- Open source software is already in products from commercial software vendors
 - Red Hat, IBM, Macromedia, Borland, Novell, Microsoft, and many others

Market Analysis

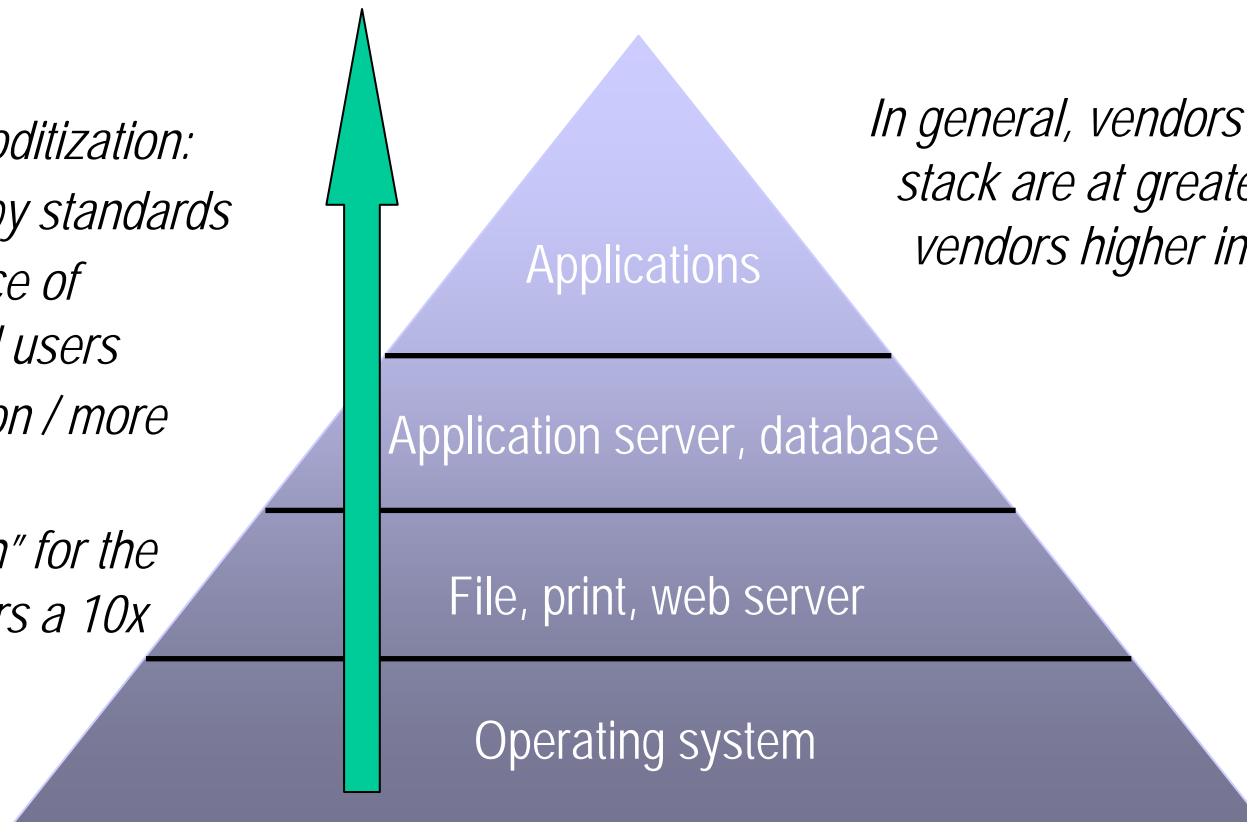


Open source is commoditizing the stack, from the bottom up

Commoditization starts at the bottom of the stack and moves up

Ripe for commoditization:

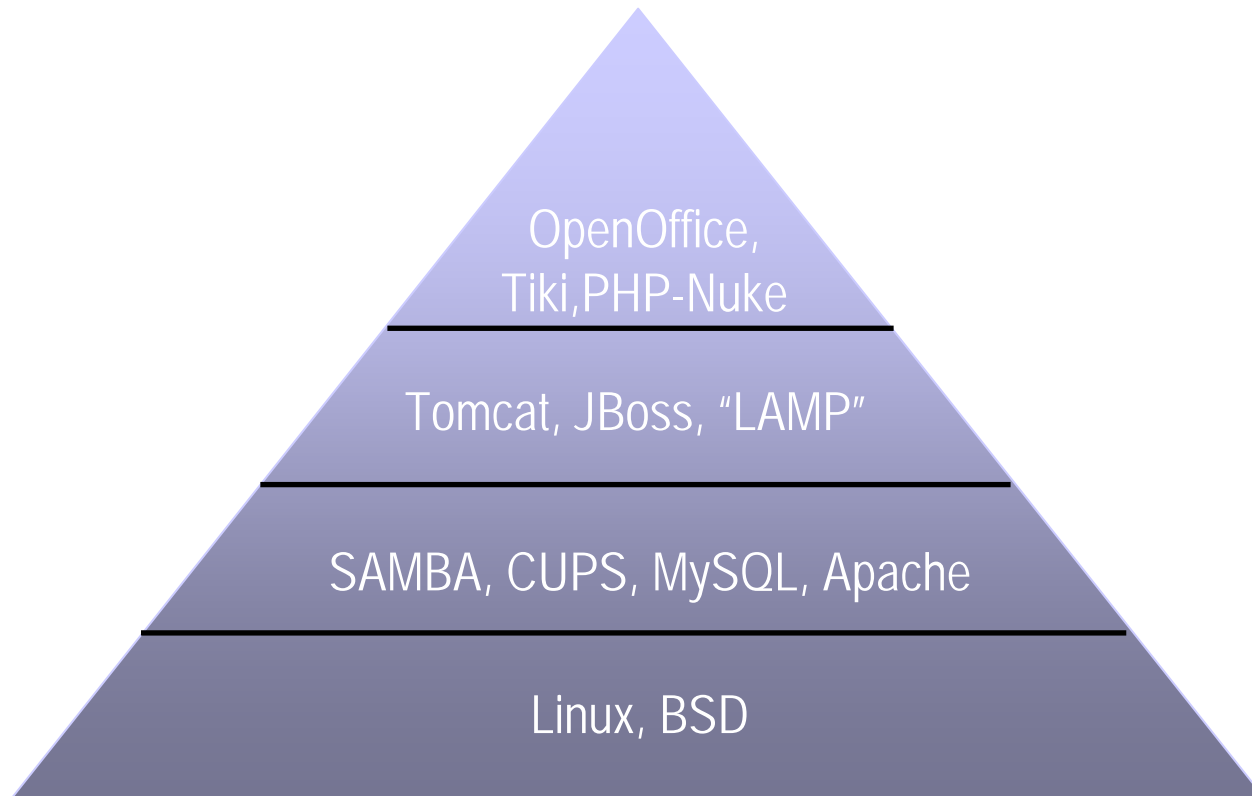
- *Well defined by standards*
- *Large audience of developers and users*
- *Less innovation / more adaptation*
- *"Good enough" for the task and delivers a 10x benefit*



In general, vendors lower in the stack are at greater risk than vendors higher in the stack



Commoditization in action

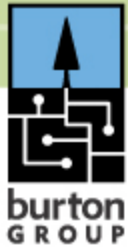




Market Analysis

How will vendors compete with free?

- Hardware, Service, and Support
 - Hardware vendors like Linux because they can replace proprietary Unix hardware with commodity Intel technology
 - Great for embedded device and task-specific appliances
 - Red Hat selling tested/integrated platform and service contracts
- Management and Integration
 - Open source is great at point technologies, but weak at integration, especially across multiple technologies and packages
 - No single vendor dominates management and integration
 - Opportunities exist for Red Hat, Novell, CA, others
- Dual license strategies
 - Product is released under the GPL and a commercial license
 - Vendor must own 100% of the copyright
 - Customers take commercial license for use that would normally violate the GPL
 - Embedded or commercial application or deployment
 - Examples: MySQL, SleepyCat Software, Trolltech AS



Market Analysis

How will vendors compete with free?

- Application and data lock-in
 - Oracle data and applications
 - Win32 APIs and Microsoft Office document format
 - NTFS and NetWare volumes
 - Not a long-term viable solution; vendors must move up the stack
- Software is the vehicle to deliver intellectual property
 - Most open source successes are lower in the solution stack
 - Higher-level solutions are IP delivered via software, thus harder to deliver via open source
- Or just sue?
 - SCO is testing the waters by threatening Linux vendors and customers
 - Others may follow suit if and when revenue is threatened



Open Source Software

22

Agenda

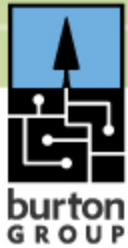
- What is open source software?
- What are the benefits and risks?
- Market assessment and revenue models
- *Recommendations*



Recommendations

Start planning for open source now

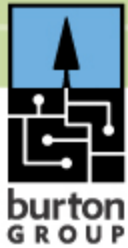
- Determine your open source commitment
 - Embrace, avoid, or wait-and-see? Or some combination thereof?
 - Determine cost/benefit vs. alternatives
 - Look for the 10x rule – 10x faster, cheaper, or better
 - Be realistic about software cost savings
- Identify key areas or low-hanging fruit
 - Infrastructure and edge servers: DNS, Web servers, app servers
 - Look for combination hardware/software devices (i.e., NAS)
 - Use open source drive to down infrastructure costs and proprietary software to compete
- Get involved
 - Learn the development process and community
 - Code check-in, applying patches, module dependencies, interoperability testing, community, support resources, mail lists, etc.
 - Start by contributing resources or useful projects



Recommendations

Start planning for open source now

- Decide on in-house or external support
 - Will you build these skills internally or depend on external vendors? Who? At what cost? How fast?
 - Can you transition your existing skills to open source projects?
 - Will you contribute or just consume?
- Understand the open source licenses
 - Treat as you would any other software license
 - Be careful: can your developers make legal and policy decisions?
 - Pay special attention to indemnification, patent issues, and derived works
- Leverage in vendor negotiations
 - Microsoft: "Under NO circumstances lose against Linux"
 - Not just Microsoft: all vendors are exposed to some degree



Conclusion

Open source software is here to stay

- Evaluation and adoption is important, but be clear of your motives and expectations
- Cost / benefit analysis is essential; don't assume that open source will be less expensive
- It's almost inevitable; start planning today
- Understand the risks and take precautions to mitigate

Additional Resources

26



- www.fsf.org
- www.opensource.org
- www.sourceforge.net
- www.freestandards.org
- www.osdl.org