Boundaryless Information Flow

www.opengroup.org

Open Standards - Open Source
The Business, Legal, and Technical Challenges Ahead

June 24-25, 2003, Minneapolis, MN, USA

In association with   UNIVERSITY of ST. THOMAS

The Open Source Middleware Opportunity
Differentiating IBM Middleware from Open Source Alternatives

June 24, 2003

Andras Szakal – Chief Architect Federal SWG
aszakal@us.ibm.com

The Open Source Middleware Opportunity

- Agenda

  - ✓ Open Source Adoption

  - ✓ Why is OSS popular?

  - ✓ Why is OSS important to IBM?

  - ✓ What is IBM's Commitment and Involvement

  - ✓ Open Source Trends

  - ✓ Positioning OSS and IBM Middleware

PC's

e-business

2. What is Open Source Software (OSS)?

- OSS = Software freely available in source-code form
  - ▸ "**while (...) do {...};**" vs. **100101011**.
  - ▸ Not owned or controlled by any single entity.
  - ▸ Community develops, debugs, maintains source code base.

- Often mistaken for "Public Domain" (frequent misconception)
  - ▸ Public Domain: author has given up copyright, no restrictions on use of code.
  - ▸ OSS authors retain copyright (with right to decide how code may and may not be used).

- Generally high quality software
  - ▸ Good ideas, good developers, allows innovation to come from anywhere.
  - ▸ Constant public scrutiny of proposed changes (all contributors are essentially users).

- Generally high security
  - ▸ Example: Turnaround time for Linux community to issue fixes for identified security bugs the same as commercial Operating Systems vendors.

OSS = Software freely available in source-code form

3. Why is OSS popular?

Open Source Advocates say...

- Developers feel comfortable working with OSS because licensing terms allow:
  - ► free access to the source code,
  - ► ability to modify the software to meet their needs, and
  - ► ability to distribute copies of both the original software and any derivative works.

- Participation in OSS communities offer potential for strong personal rewards:
  - ► opportunities to solve unique, difficult, or complex problems;
  - ► ability to build status and recognition within the developer community;
  - ► possibility to gain valuable programming experience useful for their career.

Developers are attracted to OSS projects by free access to
source and potential for strong personal rewards

3. Why is OSS popular? ...continued

- Advocates also say software users are attracted to OSS because...
  - ► No licensing fees or royalties to acquire OSS packages
  - ► Levels the playing field:
    - − Users, vendors, and developers all have access to the same source code base

- OSS gives users flexibility to...
  - − evaluate the source code to determine internal function
  - − often obtain supported versions of the software from more than one vendor;
  - − locate and fix bugs in the software if vendors are unable or unwilling to do so;
  - − tailor the software to meet the needs of their organization;
  - − port the software to new operating systems and/or hardware not supported by vendors;
  - − create customized versions of the software to improve or extend the original functionality;
  - − reuse parts of the OSS in other applications.

Users and vendors are attracted to OSS because of perceived potential for lower TCO and flexibilty...

4. Why does IBM consider OSS important?

- Customers are asking for it at the lower-levels of the stack (primarily Linux)
  - ► Example: want Linux to be part of an overall, vendor-supplied and supported total solution.

- OSS is a good approach to developing Open Standards
  - ► Popular OSS projects can become Open Standards:
    - – wide distribution and use can make it a *de-facto* industry standard;
    - – the source code takes the place of the traditional written technology description.

- OSS can be a source of innovation
  - ► innovation can happen any where at any time
  - ► developing through an "open community" leads to potentially broad ideas and creativity

> **OSS has the potential to drive both Open Standards and industry innovation.**

4. IBM Strategic Goals for Open Source

- Drive rapid adoption of open standards
  - ► Ready access to high-quality, Open Source implementations of open standards can speed industry adoption (examples: Web Services, Eclipse, Linux, and Apache)
  - ► Cooperate on "Standards" -   Compete on "Implementation"

- Use OSS as a business tool to:
  - ► Block competitors from creating "lock-in" and proprietary control points ("keep it open")
  - ► Take advantage of new bus. opportunities (e.g., leverage community, share R&D expenses)
    - – Examples: Eclipse, XML

- Extend IBM mindshare
  - ► Create a preference for IBM brands by linking brands to OSS projects
    - – Examples: Apache HTTP server   -->   WebSphere App Server
                       Eclipse   -->   Websphere Development Tools
  - ► Build relationships with a broad spectrum of developers
    - – Eclipse, Apache, Linux, Li18nux

4. IBM Open Standards Commitment

★ to support heterogeneous infrastructures

★ **to enable faster integration (i.e. improving time to value & increasing responsiveness to customers)**
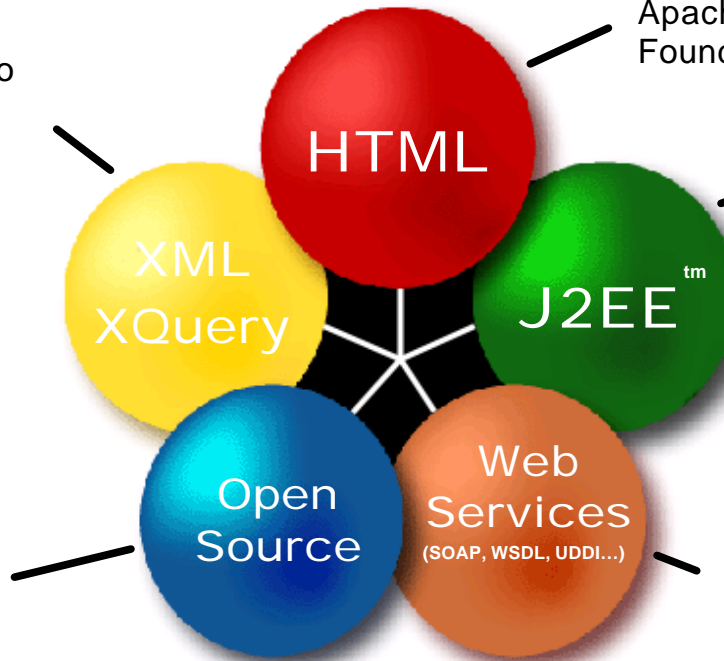
IBM helped form Apache SW Foundation

IBM provides free XML technology to developers via alphaWorks and developerWorks

IBM contributed to 80% of J2EE

HTML

XML XQuery

J2EE ™

Open Source

Web Services (SOAP, WSDL, UDDI...)

IBM contributed $40M of initial Eclipse IDE technology

IBM led or co-led the creation of Web Services foundation technologies (SOAP, WSDL, UDDI, ...)

Linux   Windows AIX   Solaris HP-UX   OS/400   OS/390 z/OS

IBM is #1 commercial investor in Linux

5. Open Source Trends - Technical Directions

- Many OSS projects - but few are "really popular" to-date

- Successful OSS:
  - ► Software development tools: GCC, Emacs, GDB, CVS, RPM, Eclipse.
  - ► Operating Systems: Linux (and to some extent FreeBSD, OpenBSD, and NetBSD).
  - ► Scripting Languages: Perl, PHP, and Python.
  - ► Internet "infrastructure" components: Apache, Tomcat, Xerces, Xalan, Sendmail, BIND, Grid.
  - ► Implementations of network protocol standards: OpenLDAP, SMTP, POP, and IMAP.

- Clear preference for lower levels of software stack.
  - ► Low barrier of entry for inexperienced technical participants.
  - ► Tendency to utilize, extend, or duplicate existing technologies.
  - ► Suitable for contributions of time and effort on personal time.

- Low probability of OSS displacing higher levels of software stack
  - ► These are complex projects requiring long-term participation of highly-skilled individuals.
  - ► Such individuals tend to be already employed by commercial software vendors.

5. Open Source Trends - Middleware

■ Is OSS moving up the software stack?

▶ Rate of evolution?.......................................

▶ Why so little progress?...............................

▶ Significant deployments?............................

▶ Inhibitors to customer uptake?....................

▶ Example: mySQL (OSS database)...............

OSS middleware needs active participation of companies with domain expertise to succeed

■ Not any time soon.....

▶ Extremely slow. OSS Middleware does not have the same market characteristics as Linux

▶ Middleware seems to hold little interest for average OSS contributors, <u>requires active participation of companies with domain expertise for significant progress</u>.

▶ None. All deployments are non-mission-critical.

▶ Lack of enterprise-class characteristics (Scalability, Reliability, Performance,...) Also: lack of experience & critical skills

▶ Maintained by mySQL AB (a company)
  − poor "support" characteristics
  − no mission-critical deployments
  − no bug updates since November 2001
  − only 4 contributors outside mySQL AB
  − project activity limited to downloads
  − OSS community concentrating on single-user, small footprint DBs

6. Does OSS work well with Commercial Software?

- YES! Open Source and Commercial Software play very well together.
  - ► Most OSS licenses allow combination and distribution of OSS and Commercial source code under a commercial license.
  - ► Some commonly encountered OSS Licenses (BSD, MIT, X11, Apache) don't require modifications to original OSS to be published upon redistribution.
- Even GPL and LGPL don't always require publication of source code
  - ► Case in point: **LINUX** - the operating system licensed under the GPL
    - – GPL <u>allows</u> commercial applications to be built on top of Linux
    - – Applications written for Linux can be distributed under a commercial license.
    - – No need to dislose source code of applications written for Linux.
- Other examples of Open Source and Commercial software working together:
  - ► Apache and WebSphere Application Server
  - ► Eclipse and WebSphere Studio Tooling
- Decision to use OSS is just another business decision
  - ► License terms need to be understood before beginning to work with OSS.

Open Source and Commercial Software work well together.

7. Major Open Source projects - Key IBM Focus Areas

- **Linux** - momentum building for the most popular OSS project

- **Eclipse** - a growing new community

- **Apache** - accelerating Open Standards adoption

- **Grid Computing** - the next frontier

7. Positioning OSS and IBM software

- **Product Support**
  - ► Greater support from OSS commercial vendors
- **Platform Support**
- **Long Term Viability**
  - ► Sometimes suspect unless supported by viable commercial vendor
- **Scalability**
  - ► Not usually part of the OSS focus
- **Reliability**
  - ► Support for QoS necessary for enterprise computing
- **Integrated Tooling**
- **Support for Open Industry Standards and Government Standards**
- **Dependence on Individuals**
- **Security**
- **Perceived TCO**
- **Partnership!**
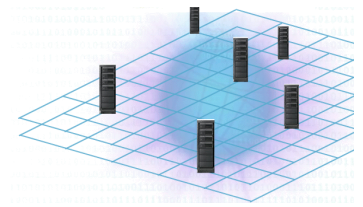
IBM Commitment to Continuous Innovation

.... while leveraging standards as an enabler
to create on demand businesses

Linux

Web Services

Grid Computing

Pervasive Computing

Autonomic Computing