

Open Source development

J.H.M. Dassen (Ray) <jdassen@debian.org>

February 17th 1999

What do these organisations have in common?

Adobe, Agfa, AMD, Amdahl, AT&T, Baan, Bellcore, Boeing, Bosch, British Telecom, Caltech, Canon, CERN, CMU, Compaq, Cygnus, Daimler-Benz, Dasa, DEC, DLR, Ericsson, FAA, Ford, GMD, Harvard, Honeywell, HP, IBM, Inria, Intel, J.P. Morgan, Kodak, Lockheed, Lucent, MCI, MIT, Motorola, NASA, Netscape, Nokia, Oracle, Philips, Raytheon, SAIC, SAP, SGI, Shell, Siemens, Stanford, Star Division, SUN, Thomson, TI, US Air Force, US Navy, VocalTec, Xerox

People from all these organisations have sent in bug reports about **DDD**, an Open Source GUI interface for command line debuggers.

Open Source software:

- Free software defined precisely through the **Open Source Definition**.
- A label for free software that emphasises its openness rather than its philosophical/moral/ethical/political aspects.

Free software

- Think free speech, not free beer.
- Free software is software whose licence(s) encourage widespread distribution, sharing and modification (\Rightarrow source available).

Proprietary software

- Proprietary software is software whose license(s) try to limit user's freedom as much as possible (restrict copying, no access to source, restrict use (e.g. forbid reverse engineering, only one copy running at a time, yearly license fees)).

Open Source definition

The **Open Source Definition** addresses:

- Free redistribution
- Availability of source code
- Derived works
- Integrity of the author's source code
- No discrimination against persons or groups
- No discrimination against fields of endeavour
- Distribution of license
- License must not be specific to a product
- License must not contaminate other software

Example Open Source licenses: **GNU GPL**, **BSD**, **X Consortium**, **Artistic License**, **MPL**, **QPL**.

Free software, Open Source software: origins and history

Prehistory Academia, hackers. Before there was a software industry, software was free.

Late 1960's Computers become more widespread. Distinction between users and programmers. Software increasingly complex (first complex OSes: Multics, OS/360). Discipline of Software Engineering born.

Late 1960's – early 1970's Couple of frustrated former Multics developers at AT&T Bell Labs hack together Unix. With hardware developments, Unix is ported to new machines after rewrite in “portable assembly”: C. Unix source is made freely available to academia (anti-trust!). Elsewhere: birth of Arpanet.

Origins and history (cont'd)

Late 1970's – early 1980's Academic hackers (esp. at Berkeley) dig Unix and extend it. Unix gains networking and X. (Both X and BSD code are true free software).

Mid 1980's **AT&T** split into baby Bells. Unix goes commercial. Legal battle over BSD Unix. Richard Stallman starts the **Free Software Foundation** (FSF) and the GNU (GNU is Not Unix) project. GPL and LGPL licenses created.

Late 1980's – early 1990's Last tainted code in BSD replaced. BSD becomes free; still aimed at big systems.

Origins and history (cont'd)

Early 1990's Helsinki Univ. student Linus Torvalds buys a 386 to learn protected mode programming. Project mutates into from scratch implementation of Unix kernel ideas.

1991–1992 First Linux distributions emerge, starting Linux' accessibility for non-hardcore hackers. Elsewhere, the web is born.

1993 SLS 1.03 distribution is extremely buggy. Ian Murdock starts project for an openly developed Linux distribution: Debian.

1993–1994 Debian pioneers package concept.

1994–1997 Steady Linux growth. Steady Debian growth; growing pains in organisation.

Origins and history (cont'd)

April 1997 Debian system **controls** plant growth experiment on board of **space shuttle** Columbia.

Mid 1997 Debian adopts its **social contract**, writes down the **Debian Free Software Guidelines** (DFSG). Eric Raymond considers it the best definition of free software so far.

Mid-late 1997 **Netscape Inc.** foresees losing the browser war; rethinks its business strategy. Linux desktop development projects become visible.

Origins and history (cont'd)

Jan 1998 **Netscape** announces intention to release browser source under a DFSG-free license. Free software community realises it may indeed change the world.*

Feb 1998 Think tank comes up with "Open Source" to market free software.

Apr 1998 **Mozilla** project born. Linux starts to get mainstream press coverage.

May 1998 Oracle and Informix reportedly have no plans to support Linux. Linux support for Ingres and O_2 announced.

Jun 1998 Silicon Valley Linux Users Group launches Windows 98... on a rocket.

Jul 1998 Mainstream press starts to have doubt over Windows 2000. Oracle and Informix announce Linux support. Linus on Forbes cover.

*1998 material based on Linux Weekly News, <http://www.lwn.net/>

Origins and history (cont'd)

Sep 1998 IBM announces DB2 for Linux support. IBM contributes updated “Haifa” instruction scheduler and other improvements to **EGCS**.

Oct 1998 **Intel** and **Netscape** invest in **Red Hat**. **Intel** joins **Linux International**. **Microsoft** cites Linux as evidence it isn't a monopoly. **Corel** announces WP8 for Linux will be “free for personal use”. 144,000 Mexican schools will run Linux with **GNOME**.

Nov 1998 **Halloween documents**: **Microsoft** internal memos about Linux and Open Source development leak.

Origins and history (cont'd)

Dec 1998 **SGI** joins **Linux International**. **Sun** announces Linux on UltraSparc support. Linux sales growth over 1998 estimated at more than 200%; unmatched market share growth. IBM releases **AFS** and **DB2** beta for Linux. IBM releases **Jikes** and **Secure Mailer** (aka **PostFix**) under license intended to be Open Source. “Linux” and “Open Source” are listed as IT trends of the year everywhere.

Jan 1999 **Lotus** announces Notes and Domino servers for Linux.

Feb 1999 **IBM Software** joins **Linux International**.

Open Source development

More fluid; does not suffer many of the problems of proprietary software development:

- More people have access to the source code. No “security through obscurity”.
- Distinction between users and developers blurs; natural evolution user → power user → contributor → maintainer, developer.
- “Debugging is parallelisable”; “release early, release often” (CVS servers: mozilla, EGCS, GNOME, ...)

Open Source development (c'd)

- No fixed amount of manpower available.
- Not deadline driven.
- Not marketing driven; less chance for creeping featurism.
- Driven by technical insight and appreciation of hack value.
- More code reuse possible.
- Utilises solid free development tools (`gcc`, `gdb`, `gmake`, `autoconf`, `automake`, `libtool`, `CVS`, `DDD` etc).
- Global development using the net (mail, web, CVS, usenet, IRC).
- Development may branch and remerge.

Recommended reading

- Eric S. Raymond, *The Cathedral and the Bazaar*
- Eric S. Raymond, *Homesteading the Noosphere*
- Ian Jackson, *Why is software freedom useful, and what does it mean?*, SANE'98,
- **Halloween I**: *Open Source software: A (New?) Development Methodology*
- **Halloween II**: *Linux OS Competitive Analysis: The Next Java VM?*,

Example: Debian GNU/Linux

Distribution: selected, integrated, documented, consistent collection of software packages.

Intermediate between end users and coders.

Distribution maintainers:

- Build packages, integrate (update menu, register MIME viewer...), test, provide conveniences (eximconfig).
- Determine Debian **policy**.
- Provide feedback to upstream developers.
- Lobby: licenses (**mozilla**, **ncftp**, **KDE/Qt** ...)
- Help users.
- Provide first-line **bug tracking**.
- Provide infrastructure for the Debian community.

Debian community

- **Mailing lists** (debian-user, debian-announce, debian-news, debian-devel, debian-policy, debian-legal ...)
- **Website**
- Mail/web based **bugtracking system**
- Web-based **FAQ-o-matic**.
- **Internet Relay Chat** (irc.debian.org # debian).
- Usenet (comp.os.linux.*)