



Open Source: Theory and Practice

John Ackermann
john.ackermann@ncr.com



Transforming Transactions
into Relationships

What is Open Source?

- An evolution of “Free Software”
 - “Free” refers to freedom, not price
 - The Free Software Foundation and its GNU (Gnu’s Not Unix) project are keepers of the flame
- Open Source is a parallel, but more pragmatic, philosophy
 - Free software purists see Open Source proponents as fallen brethren
 - Free Software is Open Source, but not necessarily the reverse
 - New term is “Free and Open Source Software” (“FOSS”)

The Open Source Development Model

- Development in the bazaar, not the cathedral
 - The communication and distribution channel is the Internet
 - Development group is typically open
 - Rapid release of new versions
 - Open, public and ongoing testing

Why is Open Source Important?

- Open Source was an important part of the Internet from the beginning, but the outside world didn't notice
- GNU/Linux made the world sit up and take notice
 - GNU developed most of the pieces of a free clone of the Unix operating environment
 - Linus Torvalds started Linux development as a hobby project, but it just grew...
 - The Linux kernel plus GNU tools result in a complete, free, alternative to commercial Unix

The Linux Revolution

- GNU/Linux is Free and Open Source Software licensed under the GPL (more on that later)
- Major vendor support
 - IBM, HP, Sun, HP(Compaq (Digital))
 - Oracle, Informix, SAP, SAS...
- The only thing that Micro\$oft fears

But Open Source is More Than Linux

- The Internet: web servers, mail systems, name servers, and more
- FreeBSD is an alternative to Linux
- Mac OSX is under Apple “public source” license
- OpenOffice (Open source version of Sun StarOffice)
- Mozilla (Netscape)

What is the Open Source Business Model?

- You can't make money through licensing, but you don't have to be a philanthropist
- Some companies create Linux "Distributions" that assemble all the bits and pieces
- Others provide after-market support, add-ons, and other services

How Does This Impact Traditional Developers?

- (Other than as competition...)
- Just about any Unix system contains some free software. You or your client may be:
 - Using it
 - Developing with it
 - Modifying and distributing it

Free Software Legal Summary

- Using it
 - “As Is” but otherwise no real issues
- Developing with it
 - License may impact works created with certain tools like “gcc”
- Distributing modified versions
 - License may require source code distribution of derivative works
- When I say “license,” I mean mainly the GPL

Major Free Software Licenses

- GPL -- the most common (and controversial) Free Software license
- Lesser GPL -- for libraries and other development tools
- BSD/MIT Licenses
- Artistic License
- Mozilla License
- Apple Public Source License

The General Public License

- Authored by the FSF and reflects the purist vision of free software
- Used by Linux kernel, all GNU software, and many other programs
- Basic concept: GPL'd software cannot be taken from the public through proprietary modifications

How Does the GPL Work?

- Three key points:
 - Object code distributions must provide access to source code at no charge
 - Derivative works fall under GPL
 - Subsequent licensors can't change license terms

The Heart of the GPL

- Any derivative of GPL'd code must be **“licensed as a whole at no charge to all third parties under the terms of this License.”**
- This and the requirement of source code availability guarantee that any derivative of GPL'd code will remain free software

The Lesser GPL

- Addresses mainly “libraries” which are linked with programs creating derivative works – intended to be a little less draconian than the GPL
- Exempts programs which link LGPL'd libraries from GPL terms if certain conditions are met
 - The user must have the ability (and tools) to modify the library and relink to create a new executable

The BSD and MIT Licenses

- Very straightforward, without the infectious nature of the GPL
- “Redistribution and use in source and binary forms, with or without modification” permitted subject to notice requirements (which are reduced in the latest versions)
- Standard “as is” warranty and liability disclaimers

The Artistic License

- Important because it is the license for Perl, a key Unix tool
- A simpler version of the GPL that allows alternative ways to comply; doesn't require modifications to be freeware if they avoid confusion with the original

The Mozilla and Apple Licenses

- Developed by Netscape and Apple for their Open-Source-like development projects
- Basically, these are what you get when the GPL is redrafted by corporate law departments

**There's No Business
Like SCO Business...**

A Brief Timeline

- September 1991 – Linux version 0.01 released
- April 1992 – USL sues BSDI claiming that “unencumbered” Unix derivative infringed Unix copyrights
- January 1993 – AT&T sells USL to Novell
- February 1994 – USL settles BSDI suit; details confidential but considered a win for BSDI due to only six source files being impacted
- March 1994 – Linux version 1.0 released
- December 1995 – Novell sells UnixWare to SCO

A Brief Timeline

- August 2000 – Caldera acquires SCO's Unix server division; remaining SCO business renamed Tarantella
- January 2001 – Caldera releases “Ancient Unix” under BSD-like license
Summer 2002 – Darl McBride takes over as Caldera CEO and company renamed SCO Group.
- January 2003 – SCO starts talking about its IP rights and hires David Boies
- March 2003 – SCO sues IBM, claims breach of contract and violation of confidentiality, not copyright infringement

A Brief Timeline

- May 2003 – McBride starts talking about SCO source code making its way into Linux; SCO stops selling Linux and starts sending letters
- August 2003 – Redhat files unfair competition, etc., suit against SCO
- November 2004 – Simmering Novell/SCO fight goes public; Novell claims its sale of UnixWare reserved rights; demands that SCO leaves prior licensees alone
- January 2004 – SCO sues Novell for slander of title; Novell completes purchase of Linux vendor SuSE

A Brief Timeline

- February 2004 – SCO amends its complaint against IBM; drops trade secret claims and replaces them with copyright infringement
- March 2004 – SCO sues Linux (and SCO) users AutoZone and Daimler-Chrysler

SCO's Claims

- IBM and SGI developed modifications to Unix were contributed to Linux; these were derivative works belonging to SCO
- Other Unix code (e.g., header files) is present in Linux
- The GPL is unconstitutional
- Novell doesn't have retained contractual right to trump SCO's claims

Further Information

- Free Software Foundation and GNU Project:
www.gnu.org
- The Open Source Initiative:
www.opensource.org
- The best place for SCO info: www.groklaw.net
- Copies of my chapter materials and of this presentation, as well as links to related sites, are at www.febo.com/law/opensource/