

Making Commercial Use of Open Source

Björn Hajek, LL.M. (Infineon Technologies AG)
Nick Holland (Field Fisher Waterhouse LLP)

31 May 2011, Session 111

Schedule

- § Definition Free Software / Open Source Software
- § Making Commercial Use of Open Source (Overview)
- § The Copyleft Effect
- § GPLv2 to GPLv3
- § Open Source and Software Patents
- § Open Source Related Case Law
- § Risk Mitigation Strategies
- § Risks of Non-Compliance with Open Source Licenses

Definition: Free / Open Source Software

§ Free Software

- Free software is a matter of the user's freedom to run, copy, distribute, study, change and improve the software and requires four essential freedoms:
 - The freedom to run the program for any purpose;
 - The freedom to study and modify the program;
 - The freedom to redistribute copies of the program;
 - The freedom to distribute copies of modified versions so that the whole community can benefit from the changes.
- Free software is “free speech” not “free beer.”

§ Open Source Software

- More or less identical concept but different philosophy
- Free software focuses on the philosophical freedom it gives to users while open source focuses on the strengths of a peer-to-peer development model

Making Commercial Use of Open Source

Benefits

- § Costs
- § Reliability
- § Stability
- § Auditability
- § Flexibility
- § Support

Risks

- § Infringement risk
- § Non-compliance liability
- § Dependencies
- § Dilution of proprietary software model
- § Devaluation of own IP
- § Viral Contamination
- § Warranty/Indemnity Gap
- § Risks on corporate disposal / acquisition

Challenges

- § Proper software choice
- § Supplier management
- § Monitoring internal use
- § Contract drafting
- § Compliant distribution

The Copyleft Effect



Copyleft means that works derived from a copyleft work must themselves be copyleft when distributed, i.e.:

- § The rights granted under a copyleft license must be preserved in modified / derived versions of the work
- § Scope of copyleft effect depends on the wording of the individual Open Source license

The Copyleft Effect: License Examples

§ Permissive licenses (no copyleft effect)

- BSD
- “Freeware”

§ Licenses with limited copyleft effect

- LGPL
- Mozilla Public License
- Apache License
- Eclipse

§ Restrictive licenses (strong copyleft effect)

- GPLv2
- GPLv3

GPLv2 to GPLv3

§ GPLv3 published June 2007

- GPLv2 remains a valid license.
- v2-covered and v3-covered programs can be used side by side in an operating system.

§ Why upgrade to v3?

“Tivoisation”

- Appliance containing GPL-covered software cannot be modified because appliance shuts down if it detects modified software.
- v3 provides mechanism to deal with such software that sits on hardware where DRM technologies restrict/report on use of copyright works.
- S.6 of v3 requires a licensee distributing software with a consumer product to accompany source code with ‘installation information’ required to install and execute modified versions.

GPLv2 to GPLv3 (cont.)

§ Patent deals

- Software patent landscape has changed considerably since publication of GPLv2.
- GPLv2 users rely on implicit patent license to make sure that a company which provided them a copy will not sue them, or the people they redistribute copies to, for patent infringement.
- GPLv3 – express patent clause in S.11.
 - Whenever someone conveys software covered by GPLv3 that they have written or modified, they must provide **every** recipient with any patent licenses necessary to exercise the rights that the GPL gives them.
 - Additionally, if any licensee tries to use a patent suit to stop another user from exercising those rights, their license will be terminated.

GPLv2 to GPLv3 (cont.)

§ New compatibility with other licenses

- e.g. Apache License 2.0 and Affero General Public License
- The change is hoped to foster more cooperation and sharing within the free software community.

§ Smoother path to compliance

- GPLv2 terminated automatically upon failure to comply with its terms and continued use of the program was copyright infringement.
 - It did not address how to reinstate the rights under the license after coming back into compliance.
 - Troublesome for inadvertent breaches.
- GPLv3 directly addresses this issue in S.8 which continues to provide for automatic termination but now also includes a procedure for reinstatement.

Software Patents

§ Growing threat to FOSS community

- By 2007, US Patent and Trademark Office had granted 200,000 software patents and continues to issue such patents at a rate of approximately 200,000 a year.
- One study found that Linux potentially infringes 283 software patents.
- End Software Patents coalition estimates that \$11.4 billion is wasted every year on litigation over software patents, with 55 software patent lawsuits filed every week.
- Companies becoming more nervous to enter markets such as mobile technology/smartphones due to uncertainty being created: “Patent infringement is easy to do accidentally, especially when it comes to software” (Andrew Katz).

Software Patents (cont.)

§ US Supreme Court decision in case of Re Bilski (2008)

- Decision confirmed that it is possible to patent a business method in the US.
- Although the patent application here was rejected on the grounds of unpatentability of 'abstract ideas', majority of justices stopped far short of a broader ruling that OS advocates hoped would have curbed so-called business method and software patents.
- Appeals court had said a patent on a non-tangible process can be granted only if it passes so-called "machine-or-transformation (MOT) test" but Supreme Court rejected this, saying MOT test is not the sole means of determining whether a process is patentable.
- Confusion and uncertainty behind ruling.

European Patent Office (EPO) and UK Intellectual Property Office (IPO) coming closer to position of US – software patents now easier to obtain.

Open Source Related Case Law

§ Oracle v Google

- In August 2010, Oracle filed a lawsuit against Google for "developing Android".
- The suit claims wilful infringement of 7 software patents related to the Java programming language distributed on Google's Android software.
- Oracle bought the Java programming language through its acquisition of Sun Microsystems in January 2010.
- Google denies charges and characterises suit as an attack on OS. Google views the inventions in Oracle's patents as "efficiency improvements" to Java and "very incremental inventions" rather than central to Java.
- OS advocates have come out fighting on side of Google – FSF has encouraged Google to take a principled stand against all software patents by fighting Oracle rather than reaching a licensing agreement.
- The trial, which will be highly technical in nature, is expected to be held before November 2011, if no settlement is reached.

Open Source Related Case Law (cont.)

§ SCO-Linux controversies

- A series of legal and public disputes between the software company SCO Group (SCO) and various Linux vendors and users.
- By mid 2004, five major lawsuits had been filed:
 - SCO v. IBM
 - SCO v. Novell
 - Red Hat v. SCO (was stayed pending outcome of IBM)
 - SCO v. DaimlerChrysler (dismissed)
 - SCO v. AutoZone (was stayed pending outcome of IBM)

Open Source Related Case Law (cont.)

§ SCO-Linux controversies (cont.)

- SCO sued IBM in 2003, claiming that it had violated SCO's rights by contributing trade secret Unix code to Linux.
- In 2004 SCO filed slander of title suit against Novell.
- In 2005 Novell counterclaims
- In 2007, court case concludes that "Novell is the owner of the Unix and UnixWare copyrights". SCO did not own Unix and UnixWare copyrights because they ignored a provision in the deal which specifically excluded certain assets (including copyright) from being transferred.
- In 2009, Tenth Circuit appeals court partially reverses summary judgment in SCO v. Novell but, in 2010, Novell, not SCO, is declared the owner of UNIX licenses.
- SCO decides to continue the lawsuit against IBM for causing a decline in SCO revenues.
- Novell now alleges that its ownership of the Unix copyrights meant SCO did not have authority to sell its own Unix-related licenses and that any such licenses really belong to Novell.

Open Source Related Case Law (cont.)

§ Bedrock Computer Technologies v Google (April 2011)

- Google found guilty in April 2011 of infringing a patent related to the Linux kernel and fined \$5m (£3.2m).
 - Bedrock sued Google in 2006, along with Softlayer Technologies, CitiWare Technology Solutions, Google, Yahoo!, MySpace, Amazon.com, PayPal, Match.com, AOL and the CME Group, alleging infringement of a Linux-related patent filed in January 1997.
 - Patent describes *"a method and apparatus for performing storage and retrieval"* - Bedrock said the companies had infringed this patent by using versions of the Linux operating system kernel on their servers.
 - Google uses Linux on employee desktops (the so-called Goobuntu flavor), its back-end servers and, since the suit was filed, has also used Linux as the basis for its Android and Chrome OS operating systems.
- Bedrock has asked for an injunction preventing Google from infringing on its patent - court has yet to rule on this.
- Red Hat also intervened in this case (several of its Red Hat Enterprise Linux customers were sued) and filed a declaratory judgment suit with the same court, attempting to have the patent declared invalid - yet to be settled.

Open Source Related Case Law (cont.)

§ Implications of Bedrock verdict?

- Remains to be seen what ripples judgment will have exactly but, as the kernel is at the core of the open-source operating system, it is thought that the verdict could be far-reaching.
- Bedrock likely to be quite successful working out license deals with Linux users and distributors, who will be required to pay royalties.
- This could be a major impediment to the growth of Linux and makes companies, including Google, that rely on open source code particularly vulnerable to patent threats.
- Companies that are profiting from open source software are evidently not immune behind the protection of open source licenses.

Open Source Related Case Law (cont.)

- § It is estimated that in last **14 months** there have been **41 patent infringement** suits levied at Google's Android platform and its rapidly growing developer ecosystem.
- Just last month, Apple declared it was suing Samsung Electronics for allegedly copying the design of its iPad and iPhone. Samsung's Galaxy products use Google's Android operating system. Samsung has in turn countersued Apple for violating its patent rights.
 - In March, Microsoft lodged a suit against Barnes & Noble, focusing on the Nook e-reader and Nook Colour tablet which run the Android OS.
 - **Elsewhere**
 - Microsoft has sued Motorola for violations of patents with smartphones that use Google's Android.
 - Microsoft has forced HTC, one of the first adopters of the Android, to reach a licensing agreement.
 - Nokia is suing Apple, Sharp, LG and Samsung.
 - RIM is suing Motorola and Sharp.
 - There are a number of other suits involving Sony-Ericsson, Qualcomm, ELAN and Kodak.
- § All these cases revolve around both copyright and patent infringements but show that the Open Source world is shifting from issues of copyright to questions about patenting.

Risk Mitigation Strategies

- § Open source policy
- § Education, awareness and internal communication
- § Maintaining software audit trail
- § Sourcing open source from commercial distributors
- § Open source carve outs in the licensing chain
- § Analysis of licenses and tracking license obligations

Risk Mitigation Strategies (cont.)

- § The act of distribution generally triggers the “additional” license obligations.
- § Common distribution requirements for Open Source software:
 - Making available of Source Code
 - Download link is not sufficient in general (with exceptions)
 - Offer to provide source code upon request requires software repository
 - Strict source code distribution helps to improve overall compliance
 - Making available of Licensing Terms
 - Download link is not sufficient in general
 - Strict source code distribution helps to improve overall compliance (e.g. with respect to unidentified components or modified license terms)

Risks of Non-Compliance with Open Source Licenses

- § The license rights granted under an Open Source license will automatically cease in case of violation of the licensing terms
 - either explicitly (e.g. GPL2/3, LGPL): „Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License.”
 - or implicitly (e.g. BSD): “Redistribution and use in source and binary forms are permitted provided that ...”
- § Liability risk (increasing awareness within FOSS community)
- § Image problem (inside and outside FOSS community)
- § Obligation to disclose (copyleft) source code?
- § Re-licensing likely to be possible

Risks of Non-Compliance with Open Source Licenses

§ Organisations procuring compliance with licenses

- The Software Freedom Law Centre in the US
- E.g. suit against Best Buy, Samsung, JVC and eleven other companies in 2009 for violation of GPL: non-compliance with source code release obligations.

§ GPLviolations.org in Germany

- E.g. prevailed in court litigation against D-Link Germany GmbH regarding D-Link's alleged inappropriate and copyright infringing use of parts of the Linux Kernel.
- The judgement finally provided the on-record, legal precedent that the GPL is valid and that it will stand up in German courts.

Open Source

Open Questions?