



**2nd Symposium of the
HyperTransport Center of
Excellence**

February 11, 2009

Michael Widenius
MySQL founder
monty@openoceancapital.com

Eben Moglen, Software Freedom Law Center



"The license is the constitution for the community"

Key questions addressed

- What is an Open Source License?
- Why go Open Source?
- How to choose a License?
- What are the main popular Licenses in use, and what are their main differences?
- How to develop high quality open source products
- What kind of community creates high quality products?

What is an Open Source License?

...Is a **Copyright License** for Computer Software
...that makes the **source code available**
...for **modification** and **redistribution**
...**free of charge**

However:

- It does **NOT** mean you can do what you want with the code
- But you can use it freely **as long as** you follow the ***Terms of the License***

Why go Open Source?

- Spread the product more quickly (more users)
- Get some part of the development done elsewhere (lower cost)
- Get things more tested and get more bug reports (higher quality)
- Possible to get development done in 'not business critical' directions (more useful product)

The above means more market recognition, feedback, leads, business, partners and sell opportunities and a strong trademark.

In general open source projects gets more feedback and better bug reports than closed source project.

Reasons for users to trust a open source vendor

- Open source vendors are more trustworthy as they depend on trust to survive
- No vendor lock in. Your investment in using the product is safe even if:
 - > Vendor goes out of business
 - > Vendor would surprisingly change business terms radically
 - > Vendor would stop supporting the version of the product you are using

If above would happen to a popular product, someone would go away with the code and start maintain it themselves

- Little risk for hidden trap doors as one can examine the product code

Benefits for developers using open source vendor

- Easy to get access, look at and use the code
- Freedom to examine and change any part of the code to satisfy your business reasons, fix bugs or port to other systems.
- Freedom to find anyone to do the above
- Freedom to use (read, build and change) the code and redistribute the code in an open source environment.

Why go Open Source?

IF you can create a sustainable business model around Open Source, proprietary vendors will have hard to compete with you.

It's nowadays increasingly harder to sell closed source solutions in a lot of business sectors.

Nor surprisingly, it's hard to choose the right license for your software as it significantly impacts your strategy, i.e.

- How you can co-operate with your user community
- How you can build your business

How to choose an OS license?

- Key Questions:
- What is your business idea around the open source product?
- What rights do you keep to your code when used, modified and potentially redistributed?
- What kind of community do you want around the product?

How to choose an OS license?

Do you want to allow others to:

- do anything with your code?
- use your code freely in any manner?
- change your code without giving you access to the changes?
- include and redistribute your code together with commercial, closed source code?
- use your code together with other open source code?
- release their modifications to your code under a different license?

What are the main popular Licenses in use, and what are their main differences?

License	Author	Link from code with a different license	Release changes under a different license
APL (Apache License)	Apache Software Foundation	Yes	Yes
BSD license	Regents of the Univ. of California	Yes	Yes
GPL (GNU General Public License)	Free Software Foundation	No	No
LGPL (GNU Lesser General Public License)	Free Software Foundation	Yes	No
MPL (Mozilla Public License)	Mozilla Foundation	Yes	Limited
Public domain		Yes	Yes
SPL (Sun Public License)	Sun Microsystems	Yes	No

Full list under http://en.wikipedia.org/wiki/Comparison_of_free_software_licences

The main open source licenses you need to consider (Very simple view)

- **Public domain**
 - > Gives user freedom to do anything, including changing your copyright and claim they wrote it.
- **BSD/Apache**
 - > Gives user freedom to full use, but needs to keep copyright in the source code.
- **LGPL**
 - > Gives user freedom to use freely, but if they distribute it changed, they need to publish the changes under LGP
- **GPL**
 - > Gives user freedom to use it freely, but if they distribute it, they need to publish changes **and** their code under GPL
- **AGPL (Optional addition to GPL V3)**
 - > Free usage, but users needs to publish the code and the code connected to it.

Business models to use with open source

- Open-Core Model - have an open-source core and sell closed-source features on top of it (e.g. SugarCRM)
- Dual Licensing Model- one product/project that gets licensed with a viral, GPL-style license and a commercial closed-source license (e.g. MySQL)
- Services Models - where you get to download a productized version of an open-source project and pay a fee for the support you get on it. You can normally also pay for training, features etc

Open Core

- Probably most popular way nowadays for business trying to do open source first time
- Is not an open source business model, because it uses closed components and most of the benefits open source developers expect from the product is gone:
 - > You can't change, modify, port or redistribute the code
 - > You are locked to one vendor
- You may be able to create a small developer community around the product but mainly by people that doesn't need the extensions.
- For community developers, the “worst” possible offer is open core or closed source that used under subscription and it stop working when subscription runs out.

So, How to decide on licensing?

You need to provide the right balance of freedom:

- Being too restrictive will harm Community growth
 - > Here you compete with companies like Microsoft
- Being too permissive will harm Business growth
 - > Here you compete with projects like Wikipedia

Going “All-In” on either, can be tough in the competitive market of today if you are not unique

Try to create a business model, which enables you to grow a business, but is as free as possible to help you create a large Community

How to develop high quality open source products

To create quality open source program, you need a community with many of the following characteristics:

- A designated active leader or active leaders that have the respect from the community
- A group of good open Open Source Citizens
- Active and passionate user and developer community around your product.
- Having developers that is using the product daily in production
- Having developers that need to extend the product for their own needs
- An open development model (all email list, discussions architecture, plans, decisions should be open)

How to develop high quality open source products

Needed community characteristics, part 2:

- Clear guidelines for how things are done and will be done.
- Clear license and business model
- Good review process of patches that enforces quality
- People with good "old time" coding style standards that actively teach others how to produce high quality code
- Extensive documentation
- Good open toolset for development (Sourceforge, Launchpad or own)

Generally the quality of the product is defined by the technical leaders

How to develop high quality open source products

You do this by:

- Having a 'enlighten dictator' (Linus) or a small group of enlightened people that works good together (PostgreSQL)
- Release early - release often, binaries + source
- Aim that each release, including alpha, should be bug free enough to be usable in production by those that try it.
- Be transparent with your plans and let users influence them
- Be open about your bugs, and when they will be addressed

How to develop high quality open source products

You do this by:

- Listen to what your users want and either do it or help them do it themselves.
- Understanding that people who write the code 'owns' the code
- Don't expect the community to produce any code!
- **Keep your promises!**
- Recognize that the path for turning a user to a customer can be long, and don't let short term monetary interest distract your long term winning strategy

How to develop high quality open source products

- Ensure that that all the technical drivers are very active with the community.
- Be open about your use of external code - Share the fame!
- Provide quick feedback to providers (bug reports and fixes, code enhancements, ideas)
- Help others in forums (around your code, and around related code you know)
- Keep up a high ethical standard on the email lists/forums
- Being a good Open Source Citizen will build trust in you and your product, and will pay off over time.
- Building a supportive Community later in the game is extremely hard - You should start day 1

For potential entrepreneurs

Open Ocean - The first major investor in MySQL AB, is launching an initiative to invest into new technology startups and help them become successful

We can provide entrepreneurs Open Source technology insight, International business know-how and Funds

For further information, please contact:
monty@openoceancapital.com



The end