Open Source, Storage Innovation and Business Models

James E.J. Bottomley
Hansen Partnership, Inc
What is Open Source

Open Source is any type of software made available under a freely redistributed Licence

Linux is just one example of an Open Source Project

FreeBSD is another

Essential element is redistributability
What is Linux?

Not organised: functions as a set of autonomous collectives (called subsystems)

Sole method of communication is email (exchanging code as patches)

Maintainer is an arbiter, but can be overridden by others in the hierarchy (like Linus)

Technical merit is sole qualification for patch inclusion
Linux and Open Source

Linux has a give back requiring licence (GPLv2).

Give Back requires publication of code for all modifications to pieces of the kernel

Acts to keep upstream head from forking.

Also makes life very difficult for vendors who wish to keep their source code proprietary.

Need good business reasons to avoid upstream.
Influencing Open Source

There is no central Architectural Council for guiding open source projects.

In Linux, guidance provided by a loose federation of Maintainers.

The development of projects is guided purely by ideas which are **backed with code**

Thus, to influence, you must field code back by engineers to argue for its inclusion.
Standards

Standards can be classified into:

- Leading Edge
- Trailing Edge

Most Linux standards (Posix, LSB) are Trailing

Storage standards mostly Leading
Business Models

The Business Model is usually to use a leading edge standard to drive development.

Ultimate object is getting some product or service to the end user via the OS.

Using a standard instead of direct OS development keeps delivery OS neutral.

Strategy takes a long time to mature!
Open Source Business Models

Traditional business value is commoditisation
- Takes pieces of stack that cost money to maintain and moves them into commodity

Rising value is participatory development
- All competitors work on same base
- Can franchise customers and ISVs in shared development

Loss of control empowers customers/partners
Standards, Products and Operating Systems

The Path to the End User

OS Development

Adoption

Publication

Creation and Refinement

implementation

Delivery
Old World Methods

In the old world, operating systems were binary.

No ability to see or change code (except with OS development licence).

All changes go through Architecture review boards.

Approval and justification is required
New World Methods

Open Source facilitates the New World

Can see and participate in development process
(No approvals required)

No need to use Standards to drive development

Can also use OS as proving ground for Implementations.
Standards In the New World

The Path to the End User

OS Development

Adoption

Publication

Creation and Refinement

Implementation

Delivery

copyright ©2008 James Bottomley

www.storage-developer.org
What does this Mean?

The OS becomes both the delivery vehicle and the implementation test bed.

To leverage the OS resources, need to seed development with early access devices.

Traditional standards path still followed for push to non Open Source OSs.

Early adoption in Open Source will drive faster adoption in other OSs.
How is this Done?

Need to have OS coders working on Standard.

Coders need to interact with the upstream community.

Object is early adoption of driver code for feature.

Development becomes a partnership with Open Source.
What does Open Source need from Standards?

Participation, but in the form of the ability to read evolving standard

i.e. Open Access

Don't necessarily need ability to attend meetings (hard to engage developer interest)

Do need contact to complain to if things look to be going in the wrong direction
Conclusions

Open Source development can help drive device/feature adoption faster than Old World Models.

Using this strategy requires changes in the Business Model.

Standards development and implementation becomes much more of a partnership.
Questions?

Can also use email:

James.Bottomley@HansenPartnership.com

Would prefer the mailing lists;
Allows Q&A to be archived

linux-scsi@vger.kernel.org