

On the Importance of Upstreaming

Robin Sheat <robin@catalyst.net.nz> // 5th November 2014

catalyst 

open source technologists

About Me

About Me

- ▶ A Koha Developer

About Me

- ▶ A Koha Developer
- ▶ At Catalyst IT

About Me

- ▶ A Koha Developer
- ▶ At Catalyst IT
- ▶ Spoilt by company policy

Where'd this talk come from?

Where'd this talk come from?

- ▶ A few years ago, I ran a Koha workshop in Malaysia...



open source technologists

catalyst 

Where'd this talk come from?

- ▶ People were stuck on very old versions

Where'd this talk come from?

- ▶ People were stuck on very old versions
- ▶ They'd made changes to suit their needs

Where'd this talk come from?

- ▶ People were stuck on very old versions
- ▶ They'd made changes to suit their needs
- ▶ They no longer had an upgrade path

What is upstreaming?



What is upstreaming?

- ▶ Getting fixes/improvements into the original project

What is upstreaming?

- ▶ Getting fixes/improvements into the original project
- ▶ Not just throwing them over the wall

What is upstreaming?

- ▶ Getting fixes/improvements into the original project
- ▶ Not just throwing them over the wall
- ▶ Shepherding them through the process

Why should you upstream?

Why should you upstream?

Stop having to maintain your own patches!

Why should you upstream?

Stop having to maintain your own patches

- ▶ Maintaining your own patches can be a lot of work

Why should you upstream?

Stop having to maintain your own patches

- ▶ Maintaining your own patches can be a lot of work
- ▶ Losing patches is easy to do

Why should you upstream?

Stop having to maintain your own patches

- ▶ Maintaining your own patches can be a lot of work
- ▶ Losing patches is easy to do
- ▶ Let upstream take care of it

Why should you upstream?

Contribute to the betterness of the world!

Why should you upstream?

Contribute to the betterness of the world

- ▶ Give people something they find useful

Why should you upstream?

Contribute to the betterness of the world

- ▶ Give people something they find useful
- ▶ Maybe they'll improve on it

Why should you upstream?

Become upstream!

Why should you upstream?

Become upstream

- ▶ Help guide the future of the project

Why should you upstream?

Become upstream

- ▶ Help guide the future of the project
- ▶ Become a core developer, release maintainer, etc.

Why should you upstream?

Vanity!

Why should you upstream?

Vanity

- ▶ It's nice to see people using things you made

Why should you upstream?

Vanity

- ▶ It's nice to see people using things you made
- ▶ Get your name in the project

Why should you upstream?

Vanity

- ▶ It's nice to see people using things you made
- ▶ Get your name in the project
- ▶ Get your company's name out there

Why should you upstream?

Vanity

- ▶ It's nice to see people using things you made
- ▶ Get your name in the project
- ▶ Get your company's name out there
- ▶ Get your client's names out there

Why shouldn't you upstream?

Why shouldn't you upstream?

It is a lot of work!

Why shouldn't you upstream?

It is a lot of work

- ▶ Getting things accepted may be time consuming
 - ▶ Especially on big projects

Why shouldn't you upstream?

It is a lot of work

- ▶ Getting things accepted may be time consuming
 - ▶ Especially on big projects
- ▶ Matching coding guidelines, test coverage, make it not impact other use cases, etc., etc.

Why shouldn't you upstream?

Fear!

Why shouldn't you upstream?

Fear!

- ▶ Legal liability

Why shouldn't you upstream?

Fear!

- ▶ Legal liability
- ▶ Competitive advantage

Why shouldn't you upstream?

Fear!

- ▶ Legal liability
- ▶ Competitive advantage
- ▶ Trade secrets

Why shouldn't you upstream?

Fear!

- ▶ Are the claims valid?

Why shouldn't you upstream?

Fear!

- ▶ Are the claims valid?
- ▶ Could it be solved with education?

Why shouldn't you upstream?

Totally local customisations!

Why shouldn't you upstream?

Totally local customisations

- ▶ Can they be generalised?

Why shouldn't you upstream?

Totally local customisations

- ▶ Can they be generalised?
- ▶ Sometimes you can't

How can you make it easy for contributors?

How can you make it easy for contributors?

Small projects

How can you make it easy for contributors?

Small projects

- ▶ Use a public tracker
 - ▶ Gitorious, Github, etc.

How can you make it easy for contributors?

Small projects

- ▶ Use a public tracker
 - ▶ Gitorious, Github, etc.
- ▶ Keep an eye on issues and pull requests

How can you make it easy for contributors?

Large projects

How can you make it easy for contributors?

Large projects

- ▶ Have a well documented process

How can you make it easy for contributors?

Large projects

- ▶ Have a well documented process
- ▶ Guides for setting up developer environments

How can you make it easy for contributors?

Large projects

- ▶ Have a well documented process
- ▶ Guides for setting up developer environments
- ▶ Guides for submitting a patch

How can you make it easy for contributors?

Large projects

- ▶ Help newcomers

How can you make it easy for contributors?

Large projects

- ▶ Help newcomers
- ▶ Let them bypass parts of the process

How can you make it easy for contributors?

Large projects

- ▶ Help newcomers
- ▶ Let them bypass parts of the process
- ▶ Fix the smaller issues
 - ▶ Rather than just rejecting it

How can you make it easy for contributors?

Large projects

- ▶ Be nice

How can you make it easy for contributors?

Large projects

- ▶ Be nice
- ▶ Handhold where you can

How can you make it easy for contributors?

Large projects

- ▶ Recognise people for their contribution

How can you make it easy for contributors?

Large projects

- ▶ Recognise people for their contribution
- ▶ Entice people back by giving them a good experience

Questions?
Discussion!