Running With The Bulls: "Master"ing Production

Presented by
Benjamin Shum
& Jason Stephenson

Trigger Warning!

Extreme sarcasm and technobabble ahead.

(some possibly graphic pictures of people being chased by livestock may occur)

Big Picture

This talk contains the “why” and “how” of using master to make your life with Evergreen more interesting. And “easier”. We hope.

“Works for me.” **OKAY THEN!**

Why do we use master?

1) New features when you want them.
2) Bug fixes come direct to you.
3) Database upgrades are simple.
4) Maintaining customizations through upgrades is a breeze.
5) You help the community.
6) Live testing is fun.

New Features

New is always better!

…except when it isn’t. :)

New Features

Scenario: Your library pays to develop a new feature. The community says we can add that for the next release (3.awesome). You don’t want to wait, so you can have it now.

Bug Fixes

Why wait for the fix to be “released”?

“Git” your fix now!!

Database upgrades are simple

Having trouble with your version upgrades?

We wouldn’t know about that. But we’ll help you anyways.

Customizations

Ever upgrade and lose your customizations?

Neither have we.

You help the Community

- You are a permanent “beta site”.
- You see new features before anyone else.
- You see problems before anyone else.
- Aren’t you lucky? /sarcasm

Bonus: extra karma points in IRC for you!

Live testing is fun!

How do we use master?

Toro, toro, technobabble!

This is *not* a Git presentation

The internet contains multiple guides on how to use Git. And you can ask your friendly local Evergreen developer for more help. Failing that, there’s always #git on freenode.

Git - making your life easier

Where: http://git.evergreen-ils.org/

Git Repositories - Community

- Main Evergreen (canonical master): http://git.evergreen-ils.org/?p=Evergreen.git;a=summary
- Working branches (that cool new thing): http://git.evergreen-ils.org/?p=working/Evergreen.git;a=summary

Git Repositories - Yours

- Your repo (you have the power too):
    - git clone git://git.evergreen-ils.org/Evergreen.git
- Add working to get new features and bug fixes:
    - git remote add working git://git.evergreen-ils.org/working/Evergreen.git

Making your branch

```
git checkout -b mystuff origin/master
```

hack, hack, hack

```
git commit -as
```

organizing your repo

Branch(es) with customizations:

- mvlc_ttopac
- mvlc_symbols
- mvlc_configuration

Maintaining your customizations

```
git fetch --all
git checkout -b test1 origin/master
git merge mvlc_ttopac
git merge mvlc_symbols
git merge mvlc_configuration
```

“That usually works, except when it doesn’t.”

Abort, abort, abort!!!

```
git merge --abort
```

git rebase

```
git checkout mvlc_ttopac
git rebase -i origin/master

# fix conflicts #  “you’re on your own kid”

git commit ; git rebase --continue
```

Bug fixes and features

- git merge working/user/bshum/bug_fix
- git merge working/user/dyrcona/some_new_thing

Bonus: Works for OpenSRF too

And SIPServer, and NCIPServer, and…..

When to upgrade (avoiding the horns)

1) After alpha, before beta:
    - After alpha, there are some new features that are cool.
    - Before beta, the last minute rush before new feature cutoff.
2) Leading up to the X.Y.1
3) Whenever you want that new thing.

And now it’s your turn

Reuters photo

Questions? (of course not!)

Photo Credits

- Slide 4 - http://blogs.sacbee.com/photos/2009/07/running-of-the-bulls.html
- Slide 11 - http://www.strangezoo.com/images/content/141757.jpg
- Slide 13 - http://www.boston.com/bigpicture/2013/07/san_fermin_festival_2013_runni.html

- Slide 16 - http://en.wikipedia.org/wiki/San_Ferm%C3%ADn#/media/File:Chupinazo8.jpg
- Slide 22 - http://www.dailyrecord.co.uk/news/uk-world-news/three-people-gored-bull-run-2046918
- Slide 25 - http://www.turismo.navarra.es/eng/propuestas/san-fermines/desarrollo/fiesta.htm
- Slide 27 - https://thelastarena.wordpress.com/2015/02/25/the-chicago-tribune-running-with-the-bulls/