ONE YEAR WITH HYDRA

WHAT WE KNOW NOW
WHAT WE WISH WE KNEW EARLIER

THOMAS SCHERZ & GLEN HORTON

UNIVERSITY OF CINCINNATI

SYSTEM/STACK DEPLOYMENT

BE CAREFUL WHAT YOU PUT IN GIT

Git history will remember passwords, keys, etc.

Use .example or .sample files instead

Use variables that will be swapped during deploy

Store real content in a safe location

USE VIRTUAL SERVER ENVIRONMENTS

Mimic a production environment locally

A reproducible, standard environment

Can be used before real servers are available

Use tools like Vagrant and Puppet

LEARN FEDORA AND SOLR

They are easy to use inside hydra-jetty

But harder when in production

Security needs to be addressed

Learn them early so it's not so hard later

SYSADMIN COLLABORATION

Many of these tools are unfamiliar to sysadmins

Fedora, Solr, Rescue, etc.

Double-check and verify backup procedures

Negotiate how much terminal access you need

DEVELOPER COLLABORATION

USE A DEVELOPMENT WORKFLOW

"Gitflow" is a good one

Use good commit messages & branch names

Practice with a test project

Adapt it to your needs

FOLLOW A SCRIPT FOR QUALITY ASSURANCE

Don’t rely on spec tests alone

Don’t just focus on changes you made

Document the steps a user would take

Run though script in QA environment

COLLABORATION AREA

RAILS APPLICATIONS

Understanding rails conventions

- **Models, Views, and Controllers**
- **Test driven development**
- **DRY - don't repeat yourself**
- **Rake tasks and Generators**

Managing Updates – Things change.

- **Locking down Gem versions**
- **Gem Dependencies**

HYDRA COMMUNITY

- Don’t reinvent the wheel
  - Look at what others have done.
  - rake task, config files, travis, …
  - Scholarsphere, DIL, Curate_ND,..
- Get Involved : Get in Touch
  - IRC, Commiters’ Call, Hydra-Tech, Hydra Camps, github
- Set up a Sandbox server
  - Try other Hydra apps and gems