

Automating CFML App Build, Test, Deploy

Joseph Lamoree
@jlamoree

Who is this guy?

- CFML developer since '96
- Open Source contributor
- Southern California native
- Principal Engineer at eCivis in Pasadena

Where you at?

- Blog: <http://lamoree.com>
- Twitter: @jlamoree
- Email: joseph@lamoree.com

Questions

- During
- After

Problem Description

- Developer preferences
- Project lifecycle
- Manual deployment sucks

Objectives

- What's possible in automation
- How components fit together
- Team workflow practices
- Avoid common problems

Working Locally

- Use your favorite IDE
- Deploy to a local CFML engine
- Commit
- Demo

Demo Config

Test and Deploy Server

Sasha



Local development
Jenkins server
Selenium Grid Hub



Development Environment

Mercedes



NGINX



Functional Testing

Nikki



Selenium Grid Node



Staging/Prod Environment

Adolf



NGINX



Lili



Git server (gitolite)



Porsche



NGINX



Isis



Selenium Grid Node



Application

- CFML application code
- Dependencies: frameworks, Java libraries
- Documentation
- Tests
- Build: properties for everyone

Apache Ant

- Choosing a deploy method
- Properties: first one wins
- Token replacement
- Creating artifacts

Ant Targets

- Break project into logical pieces
- Run groups of targets
- Set properties to modify behavior
- Example

Jenkins

- CI, and more
- Really, just a JAR file
- Organize your environments
- Monitor progress and keep history

Jenkins Plugins

- Git: enable repository interaction
- JobConfigHistory: track changes
- Selenium: create a Selenium Grid

Jenkins Jobs

- Job/Project definitions
- Nodes: masters and slaves
- Platform considerations
- Jobs tied to node or capability

Job Triggers

- *As you want*
- Scheduled
- After a commit

Cascading Jobs

- Jobs run after another
- Stopping the cascade on failure
- Adding quiet time

Job Actions

- Run multiple Ant targets
 - Target sets for each environment
 - Properties to change build
- Execute shell commands
 - Cycle services, clean caches

Testing

- Unit tests with MX Unit
- Functional tests with Selenium
- Distributing tests

Test Results

- Job history and trends
- Jobs build JUnit reports
- Subscribe to RSS for results

Git Branches

- Master is Production
- Staging is for QA
- Development is for CI
- Personal/Feature is for work

Workflow Demo

- Merge to dev
- Check results
- Merge to stage
- Verify automatic deployment

Questions