Practical Data Visualization
Infrastructure, Tools, and Project Management

Jonathan Page
Overview

- Infrastructure
- Development and Monitoring Tools
- Project Management
Infrastructure

Virtual Machines and Server Applications
User perspective

Production

Database

Worker

Staging

File

Deployment
User perspective

Production

Database

Worker

Staging

File

Deployment
3rd Party Perspective

Production File

Database Worker

New Data Files

Staging Deployment
Admin Perspective

Production

Database

Worker

File

Staging

Deployment

New Data Source
Developer Perspective

New Feature

Production
Worker
Staging
Deployment

Database

File

Dev Test
GitHub Pull Request
Travis CI
Approval
Production Server

- UHERO Data Manager (Ruby on Rails)
- REST API (Go)
- NGINX – serves static files
  - UHERO Data Portal (Angular, Highcharts)
  - DBEDT Data Warehouse (Angular)
Database Server

- MySQL
- Redis
Worker Server

- Sidekiq Workers (Ruby on Rails)
File Server

- SMB File Server
- Stores raw data files for future processing and backup
Staging Server

- Sandbox for testing code/DB changes before move to production
- All applications on Production Server
- Nightly copy of DBs from Database Server
- Testing subset of File Server contents
- Sidekiq workers from Worker Server
Open Source Software

- github.com/UHERO/udaman
- github.com/UHERO/rest-api
- github.com/UHERO/data-portal
- github.com/UHERO/dbedt-data-portal
Lessons Learned

▪ AWS (the cloud in general) is more expensive than hosting with University ITS VMs
▪ Prefer a small group of well chosen programming languages
▪ Building is more fun than supporting, so build carefully
Tools
Development and Monitoring
Development Tools

▪ Development Environments
  ▪ IDEs: RubyMine, WebStorm, SublimeText, Atom, Vim
  ▪ Vagrant, [Future] Docker

▪ Version Control
  ▪ Git, GitHub.com

▪ Code Review
  ▪ GitHub Pull requests
Development Tools (continued)

- Testing
  - Travis.CI
- Deployment
  - Currently manual runbook, [Future] Bamboo (Atlassian)
- Documentation
  - Confluence (Atlassian)
- Issue Management
  - JIRA (Atlassian)
Lessons Learned

▪ Stick to industry standard tools as much as possible (community help is invaluable)

▪ There is always a newer, shinier tool; “if it aint broke, don’t fix it”

▪ Some tools allow you to build more reliable and more maintainable software

▪ Tool switching is not costless
Project Management

Agile
Agile

- Kanban for service teams
- Scrum for development teams
- JIRA (Atlassian)
Scrum Ceremonies

- Daily Scrum (Standup)
- Sprint Planning (Story Grooming, Planning Poker)
- Showcase (Sprint Review)
- Sprint Retrospective
Lessons Learned

- Scrum ceremony adherence should depend on development team size
- If the structure of Scrum is intimidating/overbearing, try Kanban
- If Kanban feels to unstructured, try Scrum
- It’s not uncommon for development teams in the tech industry to switch between flavors of Agile
- See if your University ITS has a license for JIRA
UHERO
THE ECONOMIC RESEARCH ORGANIZATION
AT THE UNIVERSITY OF HAWAIʻI