Journey Towards Containers

Marisa Strong: Applications Program Manager
Eric Lopatin: Product Manager, Digital Preservation
Terry Brady: Software Developer/Technical Lead

UC3

University of California
CDL
California Digital Library

Land Acknowledgement

Merritt

We acknowledge that the cities of Oakland and Berkeley sit in the territory of the Huichin, part of the traditional land of the **Chochenyo Ohlone**, the successors of the historic and sovereign Verona Band of Alameda County.

We acknowledge that Seattle sits on the traditional land of the **Coast Salish** people, including the Duwamish People past and present. We honor with gratitude the land itself and the Duwamish Tribe.

This acknowledgement does not take the place of authentic relationships with indigenous communities, but serves as a first step in honoring the land we are on.

UC3/Merritt Overview

Merritt

Merritt is a **low-cost**, **bit-level** preservation repository service from the University of California Curation Center (UC3) that enables the **UC community** to manage, archive, and **share** its digital content.

- 24 Servers, 9 high availability services
  207 TB, 60M files (3 copies)
- https://merritt.cdlib.org/

Three Viewpoints

PM

PGM

DEV

- From a Product Manager perspective
- From an Application Program Manager perspective
- From a Developer perspective

Why are we interested in containers?

PM

PGM

DEV

- Robust, readily available, containerized environments enable experimentation and boost productivity.
- Common environments promote collaboration and tribal knowledge across the team.
- A containerized stack provides a platform for end-to-end testing during development.
  - The stack can be easily reset to a known state.
  - The same tests are run after Stage and Production deploys

Why are we interested in containers?

PM

PGM

DEV

- Reducing our footprint which results in **reduced costs**.
  - Servers and DB Instances
  - DevOps resources to support servers
- **Stability**: providing a much needed regression test environment that can be run by any team member
- **Iterative**: allows for iterative development for building out a test / dev environment
- DevOps benefits: configuration of containers is stored in the deployment (12 Factor App Methodology: https://12factor.net/config)

UC3

Why are we interested in containers?

PM

PGM

DEV

- Containers provide a common way to access running services
  - Supported on a Mac, Windows or Linux
  - Supported on a desktop and cloud server
- Containers integrate with a variety of development environments
  - Without prescribing a development environment
- Container Disposability is particularly suited for development and test environments
- Well “containerized” applications (vs monolithic applications) may be easier to auto-scale

UC3

PM

PGM

DEV

What prevents us from going all-in with containers?

- Analogs of a production services running in containers do not exist at CDL, though we are keen to discuss these with external teams.
- Need to determine what we think would be impacts to users and systems in the event of microservice outages.
- Team bandwidth – we have a lot going on! The addition of a migration to containers would delay many projects.

UC3

PM

PGM

DEV

What prevents us from going all-in with containers?

- Scalability with large file objects
- Support dependency from infrastructure team; troubleshooting containers
- Maintainability of images; security implications

UC3

PM

PGM

DEV

What prevents us from going all-in with containers?

- Question:
  - From a system administration standpoint, is container orchestration a better/cheaper/more flexible way to operate a set of services?
- It is EASY to introduce individual containers in a development workflow
- It may be COMPLEX to introduce container orchestration into a production environment

UC3

What we have done with containers

- The UC3 Development Team ran through a Docker Tutorial together
    - https://cdluc3.github.io/docker-tutorial/
- Created a DEV stack to test a dozen microservices
    - This replaced a dedicated set of DEV servers
    - https://github.com/cdluc3/merritt-docker
- Created an automated test suite for our system
    - https://github.com/CDLUC3/mrt-integ-tests
- Deployed Lambda applications (serverless apps) packaged as containers
    - https://github.com/CDLUC3/mrt-admin-lambda

Container Lessons learned so far

PM

PGM

DEV

- The process of prototyping is much improved through use of containerized development environment.
- The occasional technical "mysteries" that arise from learning to use containers are worth riding out, in terms of overall team benefit.
- Temptation to move forward with containers in production may be tempered with resource constraints and dependencies.

Container Lessons learned so far

PM

PGM

DEV

- Provided opportunity for team to learn together which support shift in culture
- Change can look differently to everyone; how each developer uses container may differ

UC3

Container Lessons learned so far

PM

PGM

DEV

- The modifications required to make our services testable in a containerized environment have been an undisputed improvement to our services
  - Deploy all configuration with the code
  - Use AWS system parameters
  - Make minimal assumptions about the application file system
- Our docker-compose files have been a great mechanism to convey a high level overview of a system

Thank You!

Project Links

- https://uc3.cdlib.org/
- https://merritt.cdlib.org/
- https://github.com/cdluc3/mrt-doc
- https://github.com/cdluc3/merritt-docker
- https://cdluc3.github.io/docker-tutorial/

Our GitHub Profiles

- https://github.com/terrywbrady
- https://github.com/marisastrong
- https://github.com/elopatin-uc3