Agenda

CONTINO

3 Musketeers why & who
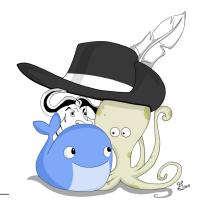
- The drive for the 3 Musketeer the approach is to have a consistent patterns for managing applications.
- Implementation of 3 Musketeers can vary based on the needs of the team.
- The goal is consistency in the way software is delivered and run.

3 Musketeers Who & Why

Who		
Docker	**Compose**	**Make**
- Builds, tests, and deployments can happen independent of the host system. - Jenkins VS local execution.	- Orchestrating multiple docker containers. - Simplifies makefile docker commands.	- Multi-platform build automation tool. - Uses Makefiles and associated targets.

3 Musketeers Who & Why

Why		
Consistency	**Control**	**Confidence**
Run the same commands no matter where you are. • Linux • CI/CD tools	Take control of languages, versions, and tools you need, and version source control your pipelines with your preferred SCM.	Test your code and pipelines locally before your CI/CD tool runs it. Feel confident that if it works locally, it will work in your CI/CD server.

CONTINO

Make

Make

- Make
    - A powerful cross platform build automation tool.
    - Make uses 'Makefiles' and associated targets to run build steps.
    - Originally created Stuart Feldman at Bell Labs in 1976!
- Common make targets
    - `Make Clean:`
    - `Make Build:`
    - `Make Run:`
    - `Make Help:`

Docker

Docker

What is docker and how are we using it?

- A standard unit in which an application resides. Containerization uses the host operating systems Kernel while implementing its own file system.
  - Verification images
    - Images used in the Hello-World sample app pipeline
    - Verify-sonarqube
  - Utility scanner images
    - Utility-scanner-sonarqube
- Docker containers are NOT the same as VMs, but they can be described this way for new learners to understand the concept.

Docker

- Single-use
  - Each docker image has a specific use. Verifier or utility.

- Additional Highlights
  - Code packaged with dependencies so applications can run across environments
  - Dockerfiles -> Docker images.
  - Docker images -> Containers at runtime.
  - Standalone environment.

Docker Compose

Docker Compose

- Compose
  - A tool for defining and running multi-container Docker applications.
  - Creates a network of running/stopped containers.
  - Can help with cleaning up after yourself.
- Utilise compose to help clean up cluttered makefiles.
  - When docker run commands become too long an complex, consider implementing compose.
  - Makefile without compose.
    - ```
      docker run -v ${WORKSPACE}/app -it ${IMAGE_NAME} bash
      ```
  - Makefile with compose.
    - ```
      docker-compose run local
      ```

My Opinion

CONTINO

Assessment: Pros

- Can really help unify a consistency in the management of applications across a number of teams

- Simplifies the commands needed to manage applications.

- Code repo comes with everything you need to run and manage the application.

- Components are generally available for any platform. Making app stack more portable

- Relatively low bar of entry

- Simple to implement, but can support an incredible range of complexity and customization

- Easy to establish contracts for running/managing applications across disparate teams

Assessment: Challenges

- Tool and version Sprawl

- Can get messy managing targets across multiple disparate environments

- Can add considerable overhead of deploying/managing a given application

- Hard to do well without these skills represented on the team. (read: non cross functional teams)

- Can quickly get too big to run in local/dev environment.

Assessment: Best Practices

- Near facist level enforcement of standards. Establish Contracts

- **KEEP MAKEFILES AS CLEAN AS POSSIBLE**. Defer complex logic to services/components outside the Makefile.

- Make sure you account for a complete CRUD capability for every 3M process

Q & A

Links

- Demo Code - https://github.com/gpatmore/cncb-3musketeers
- 3M Site - https://3musketeers.io
- Make Docs - https://www.gnu.org/software/make/manual/make.html
- Docker Docs - https://docs.docker.com
- Docker-Compose Docs - https://docs.docker.com/compose/

CONTINO

Thank You

London

london@contino.io

New York

newyork@contino.io

Melbourne

melbourne@contino.io

Sydney

sydney@contino.io

Atlanta

atlanta@contino.io

www **contino.io** 🐦 **continohq** in **contino**