Docker Takeaways

Author: William Roberts

Organization: Space Science and Engineering Center @ University of Wisconsin Madison

Purpose of this talk

Expose listeners to interesting and useful Docker concepts. My goal is to spark ideas and bring attention to as many tools and concepts as I can, without wasting time on specifics. We will NOT dive into in-depth on examples.

Examples are on this github repository:
https://github.com/wroberts4/docker_take_aways

Intended audience

This talk is aimed at people who already have a very basic understanding of Docker. However, you don’t need to be familiar with it to take away useful information!

Introduction to Docker

Docker is an open platform for creating application "images", and running said application images in "containers". Docker makes it trivial to run software between systems and environments; if you have Docker, you can run the software! Personally, Docker is my favorite and most used development tool because allows me to save lots of time and effort in both development and deployment. Similar to how virtual environments (anaconda/mamba) simplifies package management and deployment, Docker simplifies system and application management and deployment.

Create an image

Dockerfile:

```
FROM centos:7
RUN yum -y install python3
COPY hello.txt /hello.txt
```

hello.txt:

```
hello world!
```

Command line:

```
$ docker build -t my_image .
$ docker run --rm -it centos:7 pwd
/
$ docker run --rm -it my_image cat hello.txt
hello world!
$ docker run --rm -it my_image python3 –version
Python 3.6.8
```

Pre-built software

- Jupyter
- Java
- Conda/Mamba
- Nginx
- Docker in Docker

Dockerfile commands

Multi-stage builds

Docker logs

Command line flags

Docker volumes

Docker networks

Docker deploy to registry

Buildkit

Non-root images

Docker deploy to cloud