Offering containers as a service: an ALMA use case and prototype

Brian Major, Sebastien Fabbro, JJ Kavelaars

IVOA, May 2018, Victoria

National Research Council Canada
Conseil national de recherches Canada

Use Case: Running CASA on ALMA datasets

ALMA Calibration and Image Processing with CASA:

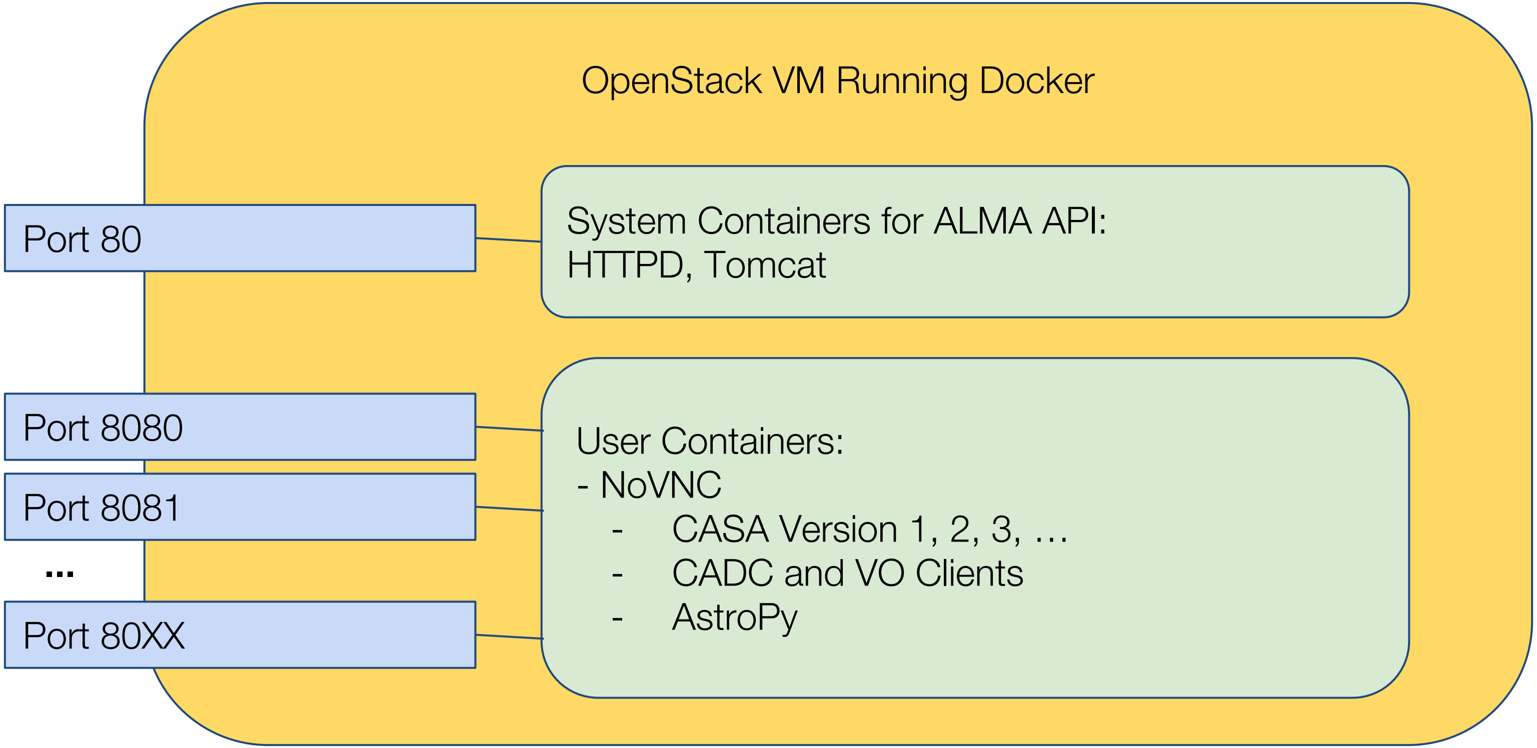
- High barrier to entry for ALMA users
- Needs a lot of disk / scratch space to run
- Each ALMA dataset needs a specific version of CASA (about 7 of them)
    - These versions are dependent on different OS distributions
- CASA can run in a shell or with a visual GUI
- VOSpace client
- CADC Archive clients
- AstroPy
- Other python libraries?
- Anything else?

Design Goals

1. Offer containers at the *USER*, not *SYSTEM* level

2. Keep containers small, single purpose
   a. Use composition, not inheritance?

Prototype Architecture



OpenStack VM Running Docker

Port 80

Port 8080

Port 8081

...

Port 80XX

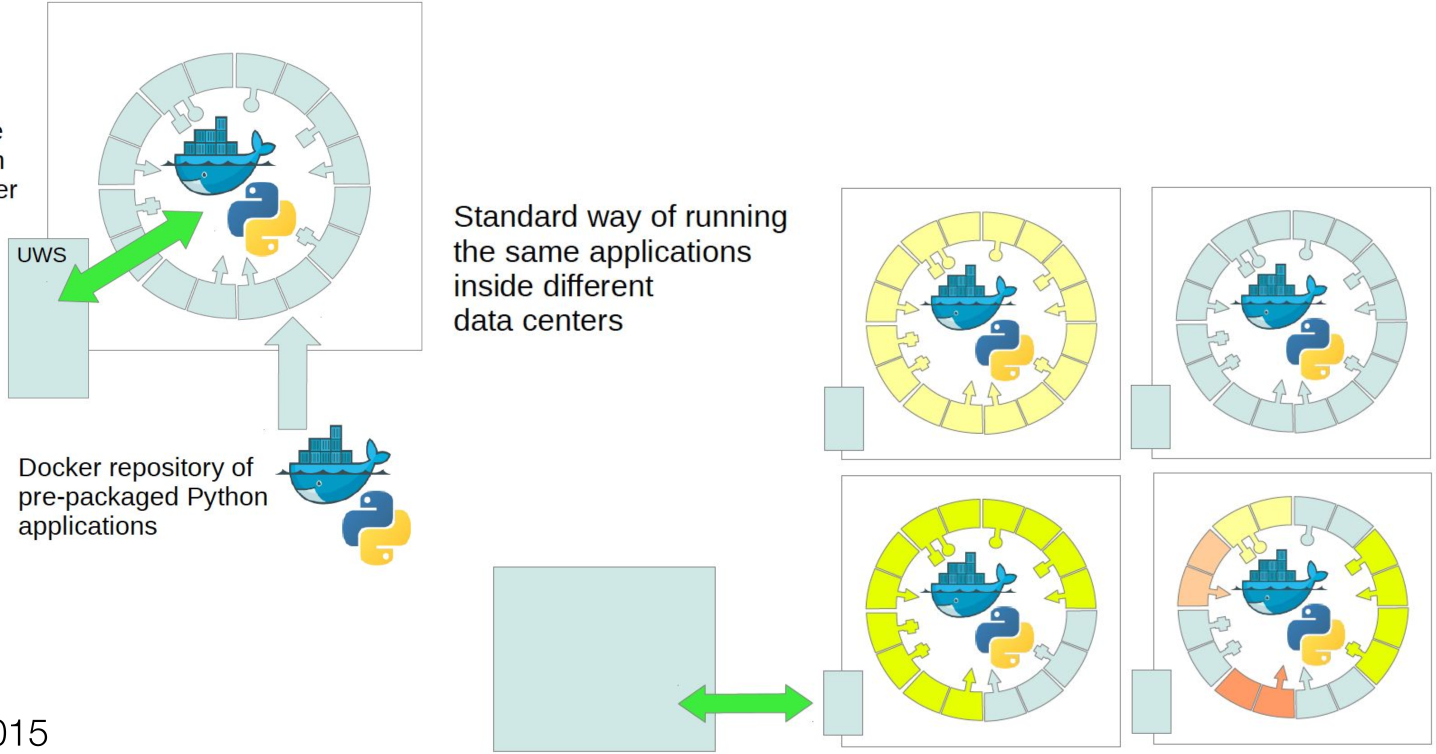
System Containers for ALMA API:
HTTPD, Tomcat

User Containers:
- NoVNC
    - CASA Version 1, 2, 3, …
    - CADC and VO Clients
    - AstroPy

Demo

Standard Platform Service Discovery & Execution

UWS service capable of launching a Python application in a Docker container inside the datacenter



Docker repository of pre-packaged Python applications

Standard way of running the same applications inside different data centers

D.Morris
Institute for Astronomy,
Edinburgh University
June 2015

Dave Morris, Sesto 2015

Conclusion

- Issues:
  - Security: Container processes running as root. Singularity?
  - Disk I/O: mounted distributed storage too slow
- Future:
  - Scaling, Kubernetes
  - Moving to Jupyter?
    - Jupyter-CASA Repository - Asterics 2020 project *
  - Using a mounted VOSpace -- Cavern
  - Apache AirFlow - workflow in containers

* https://www.asterics2020.eu/dokuwiki/doku.php?id=open:wp3:jupyter-casa_repository