Kubernetes Overview

Matias Carrasco Kind
Senior Research Scientist, NCSA
Assistant Research Professor, Astronomy
Data Release Scientist, Dark Energy Survey
University of Illinois at Urbana-Champaign

Linea Webinar, Oct 10th, 2019
What is Data Access?

Several meanings around a central data archive, a.k.a “data lake”, “data gravity” repository with common components:

- Storage
- Security
- Retrieving
- Interacting
- Modifying
- Understanding
What is Data Access?

Several meanings around a central data archive, a.k.a “data lake”, “data gravity” repository with common components:

- Storage
- Security
- Retrieving
- Interacting
- Modifying
- Understanding
Containerization to the rescue

- It’s been around for over 10 years, but popular since 2014 thanks to Docker
- Many other alternatives (rkt, kata, shifter, singularity, etc…)
- Lightweight, stand-alone, executable package of a piece of software that includes everything to run it
- Not just applications
- Software designed storage
- Software designed network
Container organization and orchestration

- We can create a container with an application inside, now what?
- Need to consider:
  - Resource needs
  - Fault tolerant
  - Load balancing
  - Storage management
  - Lifecycle
  - Service Discovery
  - Scalability
The Kubernetes Factor

- It solves all previous issues and more (not the only one but most popular)
- Open source container management and orchestration platform
- Developed by Google, made open sourced
- One of top 5 most commented open source repositories and #2 in number of pull request
- Standard within all cloud platforms
- Flexible and extensible, customize schedulers
- Is changing the cloud computing paradigm
Kubernetes Overview

- Cloud democratization
- Easy deployment
- Controls most of the aspects
- Adopted at NCSA, CERN, LSST, NASA
- Edge Computing
- Scalability
- Federation
- Resource Manager
Kubernetes Key Concepts

- **Pod** - A group of Containers
- **Labels** - Labels for identifying pods
- **Kubelet** - Container Agent
- **Proxy** - A load balancer for Pods
- **etcd** - A metadata service
- **cAdvisor** - Container Advisor provides resource usage/performance statistics
- **Replication Controller** - Manages replication of pods
- **Scheduler** - Schedules pods in worker nodes
- **API Server** - Kubernetes API server

![Diagram of Kubernetes Cluster]

- **Namespace**
- **Ingress**
- **Service**
- **Pod**
  - **Container**
  - **Container**
  - **Container**

**Diagram Details:***
- Provides separation between environments
- Exposes a service outside the Kubernetes cluster
- Exposes a container port to the rest of the Kubernetes cluster
- A collection of containers. Deployments are responsible for managing pods
- Individual containers, each should be responsible for one process
The Kubernetes Architecture

Kubernetes Architecture

UI
User Interface

CLI
Command Line Interface

API

Kubernetes Master

Image Registry

Node 1

Node 2

Node 3

Node n

Source: Janakiram MSV
The Kubernetes Architecture

Master

Kubernetes Master

- API Server
- Scheduler
- Controller
- etcd

 Nodes

Kubernetes Node

- Image Registry
- Node 1, 2, 3, n
- Optional Add-ons: DNS, UI, etc.
- Pod
- Supervisord
- Docker
- kubelet
- kube-proxy
- Fluentd

Source: Janakiraman MSV
Namespaces/Labels

- Namespaces; can partition cluster in resources, users, etc. Different namespaces for different environments (prod vs devel)
- Labels: Used to select resources within the cluster or namespace, to select pods, nodes, deployments,

Users/Groups/Service Accounts

- User and Groups refer to humans running and using resources. Permissions/Roles are applied at these levels. Cluster scoped
- Service Accounts are for processes, permissions/Roles can be applied to allow a running pod to schedule another pod, etc... Namespace scoped
The Kubernetes Architecture

Roles/Cluster Roles

- Roles are namespaced scoped
  - CRUD resources namespace
  - Pods, deployments, PVC, service
  - Roles are bind to users/groups/sa

- Cluster Roles are cluster scoped
  - CRUD resources at cluster level
  - Nodes, namespaces, secrets, policy

Policies

- Pod Policy: at Cluster level to control how the pods/containers are run
  - Disabled running as root or a particular group, allow certain volumes to be mounted, limit access to port in host machine

- Network policy: how groups of pods are allowed to communicate with each other and other network endpoints. Namespaced scope, traffic control, use labels
The Kubernetes Architecture

Resource Quotas

- Namespace scoped
- Limit cpu, memory, storage, etc
- Limit count of pods, deployments, claims, services, pods, etc.
- Doesn’t work on nodes
- Can be updated dynamically

Node Selectors/Admission Control

- By labeling nodes, nodes can be tainted, reserved or specifically selected for scheduling
- Can enforce a set of dedicated nodes for a namespace using Admission Control
- Need to restart api server (not cluster)
Daemons sets can monitor nodes health, volumes
Etcd @ master monitor use of resources, status of resources
Controller manager monitors the status of the deployments and other resources
Dashboard
Applications
DES Labs: Collection of containerized tools for DES access

- Launched in March 2015
- Used by the Collaboration
- Running using Kubernetes at NCSA (hybrid model) since 2016
- Used for DR1 infrastructure
- Customizable
easyaccess: DES command line tool

- DES DB in Oracle
- Specifically designed for DES (internal and public)
- Enhanced SQL command line interpreter in Python
- Astronomer friendly
- Python API, web interface
- There are many other CLI and GUI clients.
- Needed a simple tool, easy to use and install
- Autocompletion
- Load/Save to hdf5, fits, csv
easyaccess: DES command line tool

- DES DB in Oracle
- Specifically designed for DES (internal and public)
- Enhanced SQL command line interpreter in Python
- Astronomer friendly
- Python API, web interface
- There are many other CLI and GUI clients.
- Needed a simple tool, easy to use and install
- Autocompletion
- Load/Save to hdf5, fits, csv
Services options for user (independent containers)
DB access: SQL Interface, quick and job submission

Insert your query in the box below. Data results for "Quick" Jobs (30 sec.) will be displayed at the bottom.

```
-- Example Query ...
-- This query selects stars around the center of globular cluster M2
SELECT 5
  CCDC, OBJECT_ID, RA, DEC,
  MAG_AUTO, G, G
  MAG, MAG_PS_F G, PSF,
  MAG, MAG, PSF R, R, PSF
FROM 5
WHERE RA between 323.36 -0.12 and 323.36+0.12 and
  DEC between -0.82-0.12 and -0.82+0.12 and
  WAVG_SPREAD_MODEL I > 1 and
  IMAFLAGS ISO G = 0 and
  IMAFLAGS ISO R = 0 and
  FLAGS R = 4 and
  Flags R < 4
```

Output file (.csv, .fits or .fits). Enable in order to submit.

Options:
- Compressed files (csv and h5 files). Slightly longer jobs but smaller files
- Send email after completion

Email

Matias Carrasco Kind
Cutouts Service: Get image cutouts from given positions

Bulk Cutouts Form

- Upload csv file (with header).
- Or enter values manually (with header).
- Generate color png or just FITS cutouts required
- Xsize (in arcminutes): 1.0
- Ysize (in arcminutes): 1.0
- Release Tag: Y6A1
- Job Name
- Email Options
- Return Options

- Stiff_RGB
- Lupton_RGB
- Just FITS

- Send email on completion
- Include CSV of tiles matched to objects
Cutouts Service: Get image cutouts from given positions
### Asynchronous Jobs: Job status, jobs name

#### My Jobs

<table>
<thead>
<tr>
<th>#</th>
<th>Status</th>
<th>Job Name</th>
<th>Job Type</th>
<th>Execution Time</th>
<th>Cancel Job</th>
<th>Queries</th>
<th>Results</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td>Job 1257/578/590</td>
<td>query</td>
<td>0</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>Job 1234/567/890</td>
<td>cutout</td>
<td>1</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Job 2345/678/901</td>
<td>cutout</td>
<td>3</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>Job 3456/789/012</td>
<td>cutout</td>
<td>3</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Job 4567/890/123</td>
<td>cutout</td>
<td>1</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Job 5678/901/234</td>
<td>cutout</td>
<td>1</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>Job 6789/012/345</td>
<td>cutout</td>
<td>-1</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>Job 7890/123/456</td>
<td>cutout</td>
<td>577</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>Job 8901/234/567</td>
<td>query</td>
<td>1042</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>Job 9012/345/678</td>
<td>query</td>
<td>-1</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>Job 1234/567/890</td>
<td>query</td>
<td>9</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>Job 2345/678/901</td>
<td>query</td>
<td>9</td>
<td></td>
<td>Query</td>
<td>Cutouts</td>
<td>Files</td>
</tr>
</tbody>
</table>
Footprint and Jupyter Labs: Exploring the data
Labs with access to jobs, easyaccess, api
Services can be customized (limited user view)
Services can be customized (default user view)
Special options for Jupyter Labs: Deploy on GPU Nodes

This feature is still under development. Please use with caution.
You can launch, access and delete your Jupyter Lab from here.
This Lab will run with 1 CPU and 2GB of RAM.

Lab status
- No Lab running
Status: Not Running

Lab status
- Ready
Status: Running

Deploy Lab
Delete Lab

Deploy Lab
Delete Lab

Go To Lab
See Token

Go To Lab
See Token

Help
NCSA DESaccess: Technology Overview

- **GPFS Filesystem**
  - Storage (2-ways)
  - Backups

- **DB**
  - oracle
  - easyaccess

- **VM Node**
  - Job Manager
  - Broker
  - Cutout/query

- **Deployment**
  - docker
  - kubernetes
  - jupyter

- **Code base**
  - HTML
  - JS

- **Front-end**
  -聚合器
  - polymer

- **Results/display**
  - Tornado

- **Jobs status**
  - MySQL
  - redis

Matias Carrasco Kind
NCSA DESacces: Deployment (Hybrid)
1. Push to branch

2. Trigger build image

3. Create Manifest from template

4. Deploy Manifest

5. Get configuration and secrets from Vault

6. Expose Services and test

7. Save Manifest in Github (version)

NCSA DESaccess: CI/CD
NCSA DESaccess: Monitoring

Web Servers
- deslabs
- desnca
- descut
- destest
- easyweb
- des SN
- eups
- svn

Databases
- dessci
- desdr
- desoper
- desoper1

File Servers
- desdr
- desar2
- prodbeta

DESDM Services

DE Release

Matias Carrasco Kind
Other Services

- Other Services being integrated, easy to configure and to expand
- Can expand to the cloud when needed
- Each service is an independent application
- Working on: Automatic registration and service discovery
- Image exploration tool to visualize and classify thousands of images
- Similarity and anomaly searches
- Others...
Galaxy selection and similarity search

Compress images from 200x200 to 50 or less, for fast search

Bowen, Carrasco-Kind, et al. in prep.
Galaxy selection and similarity search

Bowen, Carrasco-Kind, et al. in prep.
Galaxy Image Exploration and Classification

- Image Exploration
- Resize is done dynamically
- Quick Classification/Label
- Works fine with 10,000 images
- Individual classifications are saved and aggregated
- Keyboard control

https://github.com/mgckind/cutouts-explorer
Simple app with MySQL Back end
- Service
- 2 Deployments
- Ingress Rule
- PVC and PV
- Secrets

https://github.com/mgckind/container_demo
Thank you!

Questions?

Matias Carrasco Kind -- NCSA
mcarras2@illinois.edu
github.com/mgkind
matias-ck.com