How To Set Up a Gen3 Data Commons Using Helm Charts

Center for Translational Data Science (CTDS), University of Chicago
Open Commons Consortium (OCC)
Australian BioCommons
New Zealand eScience Infrastructure (NeSI)
The Agenda

- Introduction
- Helm Charts on Desktop - Center for Translational Data Science (CTDS), University of Chicago
- Helm Charts on AWS - Center for Translational Data Science (CTDS), University of Chicago
- Helm Charts on GCP - Open Commons Consortium (OCC)
- Cloud Automation on OpenStack - New Zealand eScience Infrastructure (NeSI)
- Open Discussion
Introduction
Robert Grossman, Center for Translational Data Science, University of Chicago
Welcome to the Bi-Monthly Gen3 Community Forum

Gen3 is a data platform for building data commons and data ecosystems.

The Gen3 platform consists of open-source software services that support the emergence of healthy data ecosystems by enabling the interoperation and creation of cloud-based data resources, including data commons and analysis workspaces. Gen3 aims to accelerate and democratize the process of scientific discovery by making it easy to manage, analyze, harmonize, and share large and complex datasets in the cloud.

Experience Demo Get Started
Gen3 Community Forums

- Make it easier for third parties to set up, operate commons.
- Make it easier for third parties to contribute to the open source Gen3 software base.
- Build a community of researchers using Gen3 to explore and analyze data.
Helm Charts for Desktop

Jawad Qureshi, Center for Translational Data Science, University of Chicago
Helm Charts for Desktop
Helm Charts for AWS

Jawad Qureshi, Center for Translational Data Science, University of Chicago
Helm Charts for AWS
Helm Charts for GCP
Plamen Martinov, Mikisha Patel, Urvi Sheth; Open Commons Consortium
Helm Charts for GCP Use Case

- Enable students to start Gen3 resources with $300 GCP credits and billing tight to their own account automatically while using Gen3 resources with OCC security boundary. For more information go to https://pandemicresponsecommons.org/blog/

- Enable organizations natively on GCP to use Gen3 within their own secure boundary with ease of setup.
Design Overview

CSOC & COMMONS

CSOC Admin Group GCP IAM
- Allowed to access csoc VM and gen3-cisco

Cloud SQL
- Fence, Sheeplong, Index

Google Load Balancer
- Managed Certificate

Cloud NAT
- Attached to Public IP

K8s master has public IP. Service account is the only identity allowed to connect to the cluster. Worker nodes only have private IPs.

Cloud NAT used for private worker nodes to access public internet.
Infrastructure pipeline

Developers

GitHub

Build

Cloud Build

Store

Cloud Storage

Deploy

Terraform

Helm Chart

Google Cloud environment(s)

Compute Engine

Cloud Storage

GKE
Build a Gen3 Project in GCP
Cloud Automation Deployment for OpenStack

Somesh Nistala, Eirian Perkins; New Zealand eScience Infrastructure
About Aotearoa Genomic Data Repository

Data Repository for Taonga Species -
https://data.agdr.org.nz/

This drove the need to deploy GEN3 on NeSI own Cloud system... on premise deployment

GEN3 is used for the Aotearoa Genomics Data Repository and Rakeiora project (prototype application)

- https://data.agdr.org.nz/
- https://rakeiora.data.nesi.org.nz/login (URL will change)
On-premise Deployments

1. Docker-compose deployment for submitting data
   
   https://repo.data.nesi.org.nz/
   
   ● Prototyping first in Docker-compose
   ● HPC platform

2. Kubernetes deployment for the application
   
   https://data.agdr.org.nz/
   
   ● Data storage on NeSI storage via Globus
   ● FlexiHPC platform (Openstack)
Gen3 Changes

- Creating and assigning (Minting) new Digital Object Identifiers for projects visible on the Discovery page
- Deployment NeSI own metadata-service to easily support biocultural (BC) and traditional knowledge (TK) labels and notices
  - BC/TK information must be dynamically retrieved per project
K8s Deployment

We started from gen3OnK8s.md deployment approach

- [https://github.com/uc-cdis/cloud-automation/blob/master/doc/gen3OnK8s.md](https://github.com/uc-cdis/cloud-automation/blob/master/doc/gen3OnK8s.md)

To deploy the solution on NeSi cloud:

- use Kubernetes deployment files present under cloud-automation/kube/services at master · uc-cdis/cloud-automation
- gen3 scripts cloud-automation/gen3 at master · uc-cdis/cloud-automation,
- but excluded the scripts present under cloud-automation/tf_files at master · uc-cdis/cloud-automation .

The required Kubernetes infrastructure to deploy the gen3 solution has been created using openstack core automation or using UI.
K8s Architecture

- AWS Route53
- AWS CloudFront
- data.agdr.org.nz
- internaldata.agdr.org.nz
- Magnum
- Flexi LoadBalancer
- Admin VM (Gen3 tools, pkgs)
- Ambassador Edge Stack
- Rev Proxy
- VELERO
- Elastic Cloud on K8S
- kubefwd
- dnsmasq
- PSQL Helm Chart
- Fence
- Indexd Arborist
- Sheerdox

For backups
Deployment Steps

1. Setup Ubuntu instance as Gen3 admin VM
2. Customize configurations
   - Adjust **cloud-automation** to deploy on the OpenStack e.g decouple specific AWS configs
   - Adjust **cloud-automation** repo the scripts to custom service, AGDR metadata-service, Images, adjust nginx rev proxy
   - Updated fence, user.yaml, db credentials, etc in Gen3Secrets, etlMappings, manifest.json, etc in cdis-manifest, and other configs
3. Deploy k8s cluster on Flexi Infrastructure using on OpenStack CLI
4. Whitelist Kube API server access
5. Configure DNSMASQ packages and **kubefwd** to enable access to k8s from adminVM
6. Deploy Postgresql Helm charts, configure fence_db, arborist_db, indexd_db and metadata_db database
7. Roll Gen3 deployment
8. Deploy Ambassador Edge Stack & Configure AWS route 53
9. Configure AWS CloudFront for internal access during the maintenance windows
10. Deploy ECK stack for logging
11. Deploy Velero for the k8s backup and recovery

---

**Step 3:**

```
openstack coe cluster create ProdAgdrGen3 --cluster-template AGDR_NoFlotIp --keypair xxxxxxx --node-count 2 --master-count 3 --master-flavor m3.medium --flavor m3.xlarge --fixed-network xxxxxxxxxxxxxxxxxxxxxxxx --fixed-subnet xxxxxxxxxxxxxxxxxxxxxxxxxxx --labels monitoring_enabled=false --labels floating_ip_enabled=false --labels master_lb_enabled=true --labels master_lb_floating_ip_enabled=true --labels auto_healing_enabled=false --labels auto_scaling_enabled=true --labels min_node_count=2 --labels max_node_count=5 --labels admission_control_list="NodeRestriction,NamespaceLifecycle,LimitRanger,ServiceAccount,ResourceQuota,TaintNodesByCondition,Priority,DefaultTolerationSeconds,DefaultStorageClass,StorageObjectInUseProtection,PersistentVolumeClaimResize,MutatingAdmissionWebhook,ValidatingAdmissionWebhook,RuntimeClass"
```
Monitoring System (in progress)
Documentation

- Not infrastructure independent
  - Lots of trial and error to make it work with OpenStack

- Troubleshooting to improve
Additional Resources

- Gen3 Website: [https://gen3.org/](https://gen3.org/)
- Gen3 Helm Documentation: [https://github.com/uc-cdis/gen3-helm](https://github.com/uc-cdis/gen3-helm)
- Gen3 User Forum: [https://forums.gen3.org/](https://forums.gen3.org/)
- Gen3 on Slack: [https://docs.google.com/forms/d/e/1FAIpQLSczyhhOXeCK9FdVtpQpelOHYnRj1EAq1rwwnm9q6cPAe5a7ug/viewform](https://docs.google.com/forms/d/e/1FAIpQLSczyhhOXeCK9FdVtpQpelOHYnRj1EAq1rwwnm9q6cPAe5a7ug/viewform)
- Email support: [support@datacommons.io](mailto:support@datacommons.io)
Questions?