Using OpenStack to reduce HPC service complexity

... no, that is not an oxymoron!

John Garbutt, Principal Engineer, StackHPC
5th February 2022
Why build a Supercomputer with OpenStack?
<table>
<thead>
<tr>
<th>Rank</th>
<th>TOP500 Rank</th>
<th>System</th>
<th>Cores</th>
<th>Rmax (TFlop/s)</th>
<th>Power (kW)</th>
<th>Power Efficiency (GFlops/watt)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>301</td>
<td>MN-3 - MN-Core Server, Xeon Platinum 8260M 24C 2.40Hz, Preferred Networks MN-Core, MN-Core DirectConnect, Preferred Networks Preferred Networks Japan</td>
<td>1,664</td>
<td>2,181.2</td>
<td>55</td>
<td>39.379</td>
</tr>
<tr>
<td>2</td>
<td>291</td>
<td>SSC-21 Scalable Module - Apollo 6500 Gen10 plus, AMD EPYC 7543 32C 2.86Hz, NVIDIA A100 80GB, Infiniband HDR200, HPE Samsung Electronics South Korea</td>
<td>16,704</td>
<td>2,274.1</td>
<td>103</td>
<td>33.983</td>
</tr>
<tr>
<td>3</td>
<td>295</td>
<td>Tethys - NVIDIA DGX A100 Liquid Cooled Prototype, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100 80GB, Infiniband HDR, Nvidia NVIDIA Corporation United States</td>
<td>19,840</td>
<td>2,255.0</td>
<td>72</td>
<td>31.538</td>
</tr>
<tr>
<td>4</td>
<td>280</td>
<td>Wilkes-3 - PowerEdge XE8545, AMD EPYC 7763 64C 2.45GHz, NVIDIA A100 80GB, Infiniband HDR200 dual rail, DELL EMC University of Cambridge United Kingdom</td>
<td>26,880</td>
<td>2,287.0</td>
<td>74</td>
<td>30.797</td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>HIPERGator AI - NVIDIA DGX A100, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100, Infiniband HDR, Nvidia University of Florida United States</td>
<td>136,880</td>
<td>17,200.0</td>
<td>583</td>
<td>29.521</td>
</tr>
</tbody>
</table>
Traditional HPC
Traditional HPC and TREs
Traditional HPC and HPDA

- HPC
  - HPC Specific Orchestration
  - HPC Hardware

- Big Data
  - Big Data Specific Orchestration
  - Big Data Hardware

- AI and Deep Learning
  - AI Specific Orchestration
  - AI and Deep Learning Hardware
Evolving User Requirements

Evolution away from a single cluster:

- Trusted Research Environments (TREs)
- GPU enabled Kubernetes Cluster?
- Partitions with specialist hardware types
- Hadoop based Big Data processing
- Dedicated AI platforms
- Large data sets
- “Bring your own” interactive data platforms
- …
Key Challenges

- Managing Complexity
- Sharing Knowledge
- Performance vs Flexibility
- Maintaining High Utilization
- Enable new use cases, while keeping the lights on
What does Success look like?
Adopting DevOps & ResOps in HPC

Four key measures of Software Delivery Performance:

- **Lead Time:** from customer request to being satisfied
- **Mean Time to Restore (MTTR):** failure will happen, get good recovery
- **Change Fail Percentage:** a proxy for quality throughout the process
- **Deployment Frequency:** a proxy for small batch size
Traditional HPC and HPDA

- HPC
  - HPC Specific Orchestration
  - HPC Hardware

- Big Data
  - Big Data Specific Orchestration
  - Big Data Hardware

- AI and Deep Learning
  - AI Specific Orchestration
  - AI and Deep Learning Hardware
Supercomputing Clouds

APIs: AAAI, Storage Services, Compute Services

Shared Configurable Hardware
HPC 2.0

- **HPC**
- **Big Data**
- **AI and Deep Learning**

*APIs: AAAl, Storage Services, Compute Services*

*Shared Configurable Hardware*
Supercomputing Cloud

DevOps Toolkit

Science Workflows

Science Platforms

Local Cloud APIs

Shared Configurable Hardware

External Cloud APIs
Supercomputing Cloud

- DevOps Toolkit
- AAAI
- Storage Platforms
- Compute Platforms
- Shared Configurable Hardware
- Science Workflows
- Science Platforms
- External Clouds
Supercomputing Cloud

Science Workflows

Prometheus
elastic
GitLab
ANSIBLE
Terraform
NEUTRON
IRONIC
AWS
Microsoft Azure
openstack
Microsoft Azure
openHPC
RUCIO
PANGEA
binder
jupyterhub
ceph
Nexus
kubernetes
openstack
Ironic
ovHcloud
In the Beginning (c. 2015)

- Capture use-cases for Software Defined HPC
- Help establish a Research Computing SIG within OpenStack Foundation
- Establish Cambridge as thought-leader in Scientific OpenStack
- Leverage SKA use-case and develop OpenStack Kayobe
OpenStack in Cambridge (c. 2020)

- Clinical Cloud (2016+)
  - Virtualised Cloud for Brain Imaging, etc
- OpenCB (2016+)
  - Genomics Platform
- AlaSKA (2016+)
  - Bare metal platform for SKA-SDP prototyping
- STFC-IRIS (2018+)
  - Virtual and bare metal cloud for medium scale HPC
  - Virtualised resource for medical informatics
- Arcus (2020+)
  - Unified OpenStack, including Large-scale HPC
<table>
<thead>
<tr>
<th>Rank</th>
<th>TOP500 Rank</th>
<th>System</th>
<th>Cores</th>
<th>Rmax (TFlop/s)</th>
<th>Power (kW)</th>
<th>Power Efficiency (GFlops/watt)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>301</td>
<td>MN-3 - MN-Core Server, Xeon Platinum 8260M 24C 2.40GHz, Preferred Networks MN-Core, MN-Core DirectConnect, Preferred Networks Preferred Networks Japan</td>
<td>1,664</td>
<td>2,181.2</td>
<td>55</td>
<td>39.379</td>
</tr>
<tr>
<td>2</td>
<td>291</td>
<td>SSC-21 Scalable Module - Apollo 6500 Gen10 plus, AMD EPYC 7543 32C 2.86GHz, NVIDIA A100 80GB, Infiniband HDR200, HPE Samsung Electronics South Korea</td>
<td>16,704</td>
<td>2,274.1</td>
<td>103</td>
<td>33.983</td>
</tr>
<tr>
<td>3</td>
<td>295</td>
<td>Tethys - NVIDIA DGX A100 Liquid Cooled Prototype, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100 80GB, Infiniband HDR, Nvidia NVIDIA Corporation United States</td>
<td>19,840</td>
<td>2,255.0</td>
<td>72</td>
<td>31.538</td>
</tr>
<tr>
<td>4</td>
<td>280</td>
<td>Wilkes-3 - PowerEdge XE8545, AMD EPYC 7763 64C 2.45GHz, NVIDIA A100 80GB, Infiniband HDR200 dual rail, DELL EMC University of Cambridge United Kingdom</td>
<td>26,880</td>
<td>2,287.0</td>
<td>74</td>
<td>30.797</td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>HIPerGator AI - NVIDIA DGX A100, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100, Infiniband HDR, Nvidia University of Florida United States</td>
<td>136,800</td>
<td>17,200.0</td>
<td>583</td>
<td>29.521</td>
</tr>
</tbody>
</table>
Shared Configurable Hardware
Supercomputing Cloud

- Science Workflows
- Science Platforms
- DevOps Toolkit
  - AAAI
  - Storage Platforms
  - Compute Platforms
- External Clouds
- Shared Configurable Hardware
OpenStack Kayobe Architecture
OpenStack Kayobe Deployment
OpenStack Kayobe Deployment
Baremetal Node <-> Hypervisor

- Kayobe and Kolla-Ansible, OpenStack
  - Network switches configured via ansible in Kayobe
- Ironic to deploy Controllers and Hypervisors
  - Controllers via Bifrost Ironic
  - Hypervisors are a baremetal workload
  - Science Platforms can use baremetal and/or VM servers
- Storage
  - Lustre and Ceph are applications running on baremetal
- Dynamic Networking
  - OpenStack Neutron used to configure physical switches
  - networking-generic-switch can change the access VLAN
  - Dedicated networks for inspecting, provisioning and cleaning
Bootstrapping Physical Infrastructure

- Barcode scan servers in racks
- iDRAC DHCP
  - MAC from scanning, Neutron for DHCP
- Ansible driven Ironic Enrollment
  - Ansible enables IPMI, updates Firmware
  - State machine stored in Ironic
- Ironic Inspector
  - First on 1GbE, and update ConnectX-6
  - Re-inspection on 50GbE, LLDP to get switch info
- Testing: Burn-in with HPL, MPI ping tests
- [https://github.com/stackhpc/arcus-terraform-idrac](https://github.com/stackhpc/arcus-terraform-idrac)
Tracking Bootstrapping
Tracking Bootstrapping
Tracking Bootstrapping
Science Platform:
OpenHPC Slurm
Supercomputing Cloud

- Science Workflows
- Science Platforms
- AAAI
- Storage Platforms
- Compute Platforms
- Shared Configurable Hardware
- External Clouds

DevOps Toolkit
OpenHPC Slurm on OpenStack

- Ansible install and configure OpenHPC packages
  - Terraform can manage infrastructure
  - Open OnDemand web interface
  - Monitoring using Grafana and Prometheus
  - Self-tests: HPL, MPI ping pong
- (Optional) Image build pipeline, using Packer and Ansible
- Non-impacting upgrades
  - Slurm rebuild scripts
  - OpenStack rebuild
- Autoscaling (coming soon)
- [https://github.com/stackhpc/ansible-slurm-appliance](https://github.com/stackhpc/ansible-slurm-appliance)
Tuning Ironic for Scale

- **Target:** rebuild all hosts to apply new kernel
- **networking-generic-switch**
  - Added cumulus driver to networking-generic-switch
  - Added experimental “batching” of config
- **Ironic-conductor configuration**
  - IMPI driver, avoids reboot in iDRAC driver
  - Direct deploy, not iSCSI
    - force_raw_images = false
    - Avoid neutron router bottleneck
  - Server Delete: increased rpc_timeout
- **connect_timeout** in HAproxy and MariaDB
Science Platforms:
On Demand
Supercomputing Cloud

- DevOps Toolkit
- Science Workflows
- Science Platforms
  - AAAI
  - Storage Platforms
  - Compute Platforms
- Shared Configurable Hardware
- External Clouds
Azimuth Cloud Portal

- Reduce time to science & reduce operational effort of onboarding
- Easier to **find** and **reuse** common lego bricks, between science communities
- Builds on the STFC funded work done by JASMIN
- Target use cases
  - Works with OpenID Connect
  - Get me a:
    - Bigger laptop
    - Slurm Clusters, Kubernetes clusters
    - JupyterHub, Kubeflow, DaskHub, …

https://github.com/stackhpc/azimuth
Science Platforms:
Get me a bigger Laptop!
Welcome to IRIS IAM

Sign in with your IRIS IAM credentials

- Username
- Password

Sign in

Forgot your password?

Or sign in with

Your organisation via eduGAIN

Not a member?

Apply for an account

About Us, Contact Information and Privacy Policy
Dashboard

Available tenancies

- Iris-alaska-prod
- Iris-euclid
- rcp-cloud-portal-demo
- rcp-cloud-portal-dev
- rcp-cloud-portal-prod
- rcp-rbs-lpstr-dev
- rcp-scientific-opensstack-ci-cd
Create a new machine

Machine name

*test-web-console*

Must contain alphanumeric characters, dot (), and dash (-) only.

Image

Ubuntu-Focal-20.04-20210624

Size

vm.r1.tiny 1 cpu, 1GB RAM, 10GB disk

Enable web console?

Install Apache Guacamole to provide access to the machine via a web browser.

Enable remote desktop for web console?

WARNING The remote desktop can take a long time to install and configure.

Create machine
## rcp-cloud-portal-demo

<table>
<thead>
<tr>
<th>Name</th>
<th>Image</th>
<th>Size</th>
<th>Status</th>
<th>Power State</th>
<th>Task</th>
<th>Internal IP</th>
<th>External IP</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>test-web-console</td>
<td>Ubuntu-Focal-20.04-20210624</td>
<td>3.34 GB</td>
<td>ACTIVE</td>
<td>Running</td>
<td>-</td>
<td>192.168.3.161</td>
<td>-</td>
<td>a minute ago</td>
</tr>
<tr>
<td>johng-dactest</td>
<td>CentOS8.4-OFED-5.4-1.0.3.0</td>
<td>1.11 GB</td>
<td>ACTIVE</td>
<td>Running</td>
<td>-</td>
<td>192.168.3.176</td>
<td>-</td>
<td>14 mins</td>
</tr>
</tbody>
</table>

2 machines

### Actions
- Access web console
- Attach external IP
- Detach external IP
- Firewall rules
- Start machine
- Stop machine
- Restart machine
- View machine logs
- Delete machine
Welcome to ubuntu.

The programs included with the Ubuntu system are free software; 
the exact distribution terms for each program are described in the 
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by 
applicable law.

To run a command as administrator (user "root"), use "sudo <command>". 
See "man sudo_root" for details.

ubuntu@test-web-console:~$
How did the VM access work?

- There is nothing hiding the OpenStack API here
  - Similar to Exposhere
- Login to OIDC integrated OpenStack Keystone
  - AARC Blueprint, IRIS IAM
- Create OpenStack server via API
  - Cloud-init configures guacamole and proxy
- Zenith Proxy
  - Authenticate users via OpenStack
  - Authorize based on OpenStack project membership
  - Don’t always burn a public IP, via ingress controller style proxy
Science Platforms:
Get me a Slurm Cluster!
Create a new cluster

Pick a cluster type

Slurm

Batch cluster running the Slurm workload manager.

Select

Set cluster options

Next
**Cluster name**
- test-slurm
  - Must contain lower-case alphanumeric characters and dash (-) only.

**External IP**
- 128.232.222.183
  - The external IP to use for the login node.

**Compute node count**
- 2
  - The number of compute nodes in the cluster.

**Login node size**
- vm.alaska.cpu.general.small
  - 4 cpus, 4GB RAM, 40GB disk
  - The size to use for the login node.

**Control node size**
- vm.alaska.cpu.general.small
  - 4 cpus, 4GB RAM, 40GB disk
  - The size to use for the control node.

**Compute node size**
- vm.iris.cpu.dac-quarter
  - 14 cpus, 38GB RAM, 2100GB disk
  - The size to use for the compute node.

**Cluster monitoring**
- **Enable cluster monitoring?**
  - If selected, a monitoring stack will be deployed allowing you to track and visualise the state of the cluster.
  - WARNING: This can take a significant amount of time to deploy and configure.

**Post-configuration validation**
- **Run post-configuration validation?**
  - If selected, post-configuration jobs will be executed to validate the core functionality of the cluster when it is re-configured.
rcp-cloud-portal-demo

Overview  Machines  Volumes  Kubernetes  Clusters

Name | Cluster Type | Status     | Task                | Created          | Updated | Patched
-----|-------------|------------|---------------------|------------------|---------|--------
Test-slurm  Slurm  CONFIGURING  Waiting to be scheduled  a few seconds ago  -     -

1 cluster
rcp-cloud-portal-demo

<table>
<thead>
<tr>
<th>Name</th>
<th>Cluster Type</th>
<th>Status</th>
<th>Task</th>
<th>Created</th>
<th>Updated</th>
<th>Patched</th>
</tr>
</thead>
<tbody>
<tr>
<td>test-slurm</td>
<td>Slurm</td>
<td>READY</td>
<td>-</td>
<td>19 minutes ago</td>
<td>4 minutes ago</td>
<td>4 minutes ago</td>
</tr>
</tbody>
</table>

1 cluster
How did you create the cluster?

- Generic, not specific to Slurm
- AWX runs Ansible playbooks
  - [https://github.com/stackhpc/caas-slurm-appliance](https://github.com/stackhpc/caas-slurm-appliance)
- Terraform creates infrastructure
  - State stored in Consul
  - Inventory updated from Terraform output
- SSH in to configure hosts
- Optionally using image build via Packer pipeline
  - Referencing Pulp repo mirrors
- Auto-detect if project has RDMA provider network
Science Platforms:
Get me a Kubernetes Cluster
Create a new Kubernetes cluster

Cluster name
Cluster name
Must contain lower-case alphanumeric characters and dash (-) only.

Cluster template
Select a Kubernetes cluster template...
The template determines the Kubernetes version for the cluster.

Control Plane Size
Select a size...
The size to use for the Kubernetes control plane node(s).

Enable auto-healing?
If enabled, the cluster will try to remediate unhealthy nodes automatically.

Node Groups

<table>
<thead>
<tr>
<th>Name</th>
<th>Node Size</th>
<th>Node Count</th>
</tr>
</thead>
</table>

No node groups configured yet.

Cluster Addons

- Enable cert-manager?
- Enable Kubernetes Ingress?
- Enable cluster monitoring?

Create cluster
## rcp-cloud-portal-demo

<table>
<thead>
<tr>
<th>Name</th>
<th>Status</th>
<th>Template</th>
<th>Control Plane</th>
<th>Workers</th>
<th>Addons</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>matt-demo</td>
<td>Ready</td>
<td>v1.22.6</td>
<td>Ready</td>
<td>3 (3 ready)</td>
<td>5 (5 ready)</td>
<td>13 minutes ago</td>
</tr>
</tbody>
</table>

1 cluster
### Kubernetes

<table>
<thead>
<tr>
<th>Name</th>
<th>Status</th>
<th>Template</th>
<th>Control Plane</th>
<th>Workers</th>
<th>Addons</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>matt-demo</td>
<td>Deleting</td>
<td>v1.22.6</td>
<td>Ready</td>
<td>3 (3 ready)</td>
<td>1 (0 ready)</td>
<td>15 minutes ago</td>
</tr>
</tbody>
</table>

1 cluster
Kubeconfig for matt-demo

Use this configuration file with the `kubectl` command-line tool to access your cluster.

```yaml
apiVersion: v1
clusters:
- cluster: 
  certificate-authority-data: L3B0Li5lSlQrUy04Zi5yQ0FURS15Sbi1kY35uQz5lRQV3Mz5lMj5lMH5lMj5lMj5lMj
  server: https://198.232.232.52:8443
  name: matt-demo
contexts:
- context: 
  cluster: matt-demo
  user: matt-demo-admin
  name: matt-demo-admin@matt-demo
  current-context: matt-demo-admin@matt-demo
kind: Config
preferences: {}
uers:
- name: matt-demo-admin
- name: matt-demo-admin

Close
```
How did you create K8s Clusters?

● **Kubernetes Cluster API**
  ○ OpenStack Cluster API Provider
  ○ Cinder CSI and Cloud Provider OpenStack
  ○ General good stuff: Rolling upgrades, Self-healing, Auto-scaling
  ○ Add additional RDMA network when available

● **Opinions wrapped with Helm charts**
  ○ Add and remove “add-ons” (e.g. OFED and CUDA)
  ○ Wrapped up options, driven by an operator
  ○ [https://github.com/stackhpc/capi-helm-charts](https://github.com/stackhpc/capi-helm-charts)

● **Previously used OpenStack Magnum**
  ○ … looking at adding a Cluster API driver into Magnum
Science Platforms:
Get me a JupyterHub
What are you planning?

- Pre-install KubeApps
  - With a curated Helm repository
- Expose from the Portal
  - Kubernetes Dashboard
  - Grafana
  - KubeApps
  - … and JupyterHub created by KubeApps
- Zenith Proxy for Kubernetes Service
  - Why: Share auth with the Portal
  - How: Zenith Client as a sidecar
Science Platforms:
Can I get some resources?
OpenStack Quota

- Resource Allocation converted into OpenStack Quota
  - Defines the high watermark of your usage
  - … very hard to “catch up” if you fall behind in your usage

- Dedicated Quota
  - Underutilization a problem
  - Sometimes implemented using Tenant isolation filters

- Overcommit Quota
  - Might not be able to use your Quota

… but what if you want need half your allocation in June, the rest in September?
The Coral Reef Cloud
OpenStack Blazar Reservations

Compute Resources

Time
OpenStack Blazar Preemptibles

Compute Resources

Time
Tracking Efficiency / Utilization
What can I do?
OpenStack based HPC

- **OpenStack Ironic** to manage baremetal servers
- Slurm non-impacting updates by **reboot** triggering a **rebuild**
- **Kubernetes Cluster API** to create and upgrade K8s on OpenStack
- **Azimuth** Cloud Portal for on-demand Science Platforms
- **OpenStack Blazar** for resource reservations and preemption
Would you like to get involved?

- Scientific OpenStack SIG
  - https://www.openstack.org/use-cases/science/
- OpenStack Ironic
  - https://www.openstack.org/use-cases/bare-metal/
- OpenStack Kolla, Kolla-Ansible and Kayobe
  - https://docs.openstack.org/kayobe/latest/getting-started.html
- OpenHPC Slurm on OpenStack
  - https://github.com/stackhpc/ansible-slurm-appliance
- Azimuth Cloud Portal
  - https://github.com/stackhpc/azimuth
Questions?