Porting Applications to IRIS OpenStack

John Garbutt

October 2019

StackHPC

Recent Industry Trends

StackHPC

High Performing Technology Organizations

StackHPC

Whitepaper: State of DevOps

DevOps: 'The Three Ways'

- Flow
- Feedback, Local Discovery -> Global Impact
- Continuous Learning and Experimentation

Accelerate

- Continuous Delivery
- Loosely coupled and empowered teams
- Lean management and Monitoring
- Product, Process and Cultural Capabilities

Site Reliability Engineering - SRE

- Availability, Monitoring, Emergency Response
- Change management, Capacity Planning
- Performance, Latency, Efficiency

Capabilities from Accelerate

StackHPC

Continuous Delivery

- Version Control
- Automate deployment
- Continuous Integration
- Trunk based development
- Test Automation (inc data)
- Security part of full lifecycle

Architecture

- Loosely coupled
- Empowered teams

Product and Process

- Customer feedback
- Make flow visible
- Work in small batches
- Foster and enable team experimentation

Lean Management and Monitoring

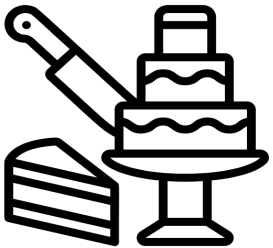
- Lightweight change approval
- Monitoring inform business decisions
- Proactive health checking
- Limit WIP
- Visualize quality and WIP

Road to Cloud

StackHPC

Server Virtualization and Before

- Pets
- Avoid Failure
- Scale up
- Bespoke



Software Defined Infrastructure

- Cattle
- Plan for Failure
- Scale out
- Automated

Road to Cloud

StackHPC

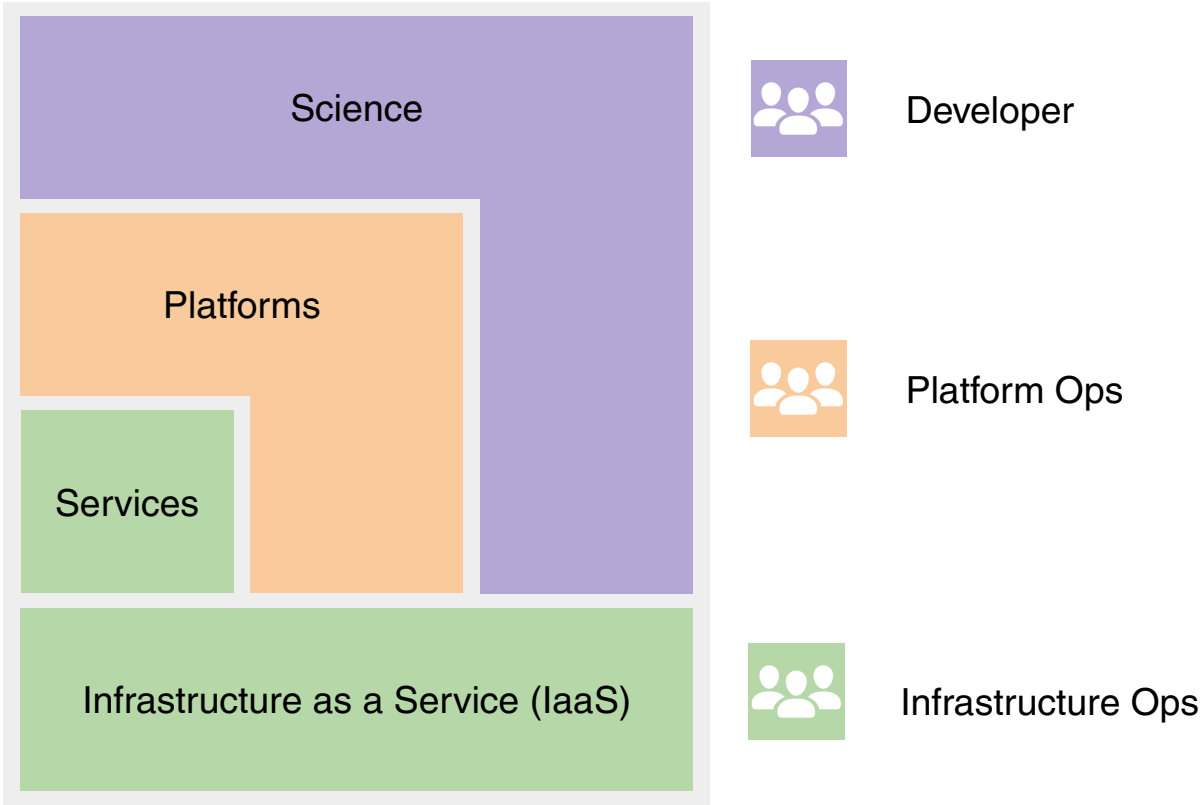
Dedicated Servers	Server Virtualization	Cloud
Isolate Failures	**Avoid Failures**	**Architect for Failure**
Service on a physical server	Service given VMs	Self-Service Infrastructure
Simple, but low utilization, long lead times	Improved utilization, higher complexity	Scale out Cattle, not Scale up Pets
Hardware redundancy to reduce Failures	Try to automatically recover from hardware failures	Container Orchestration, Event triggers

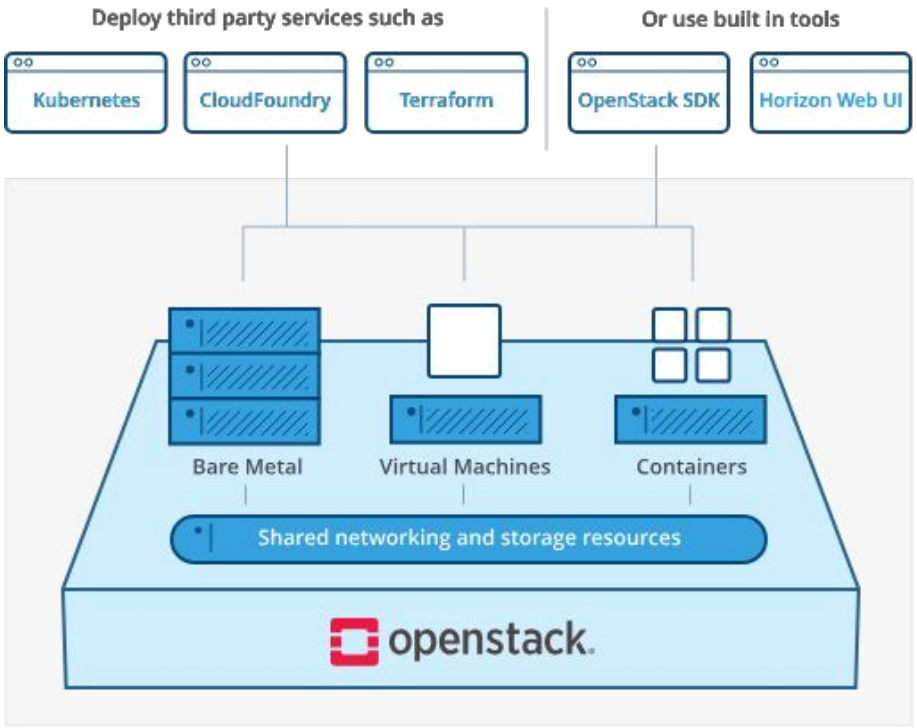
How does OpenStack help?

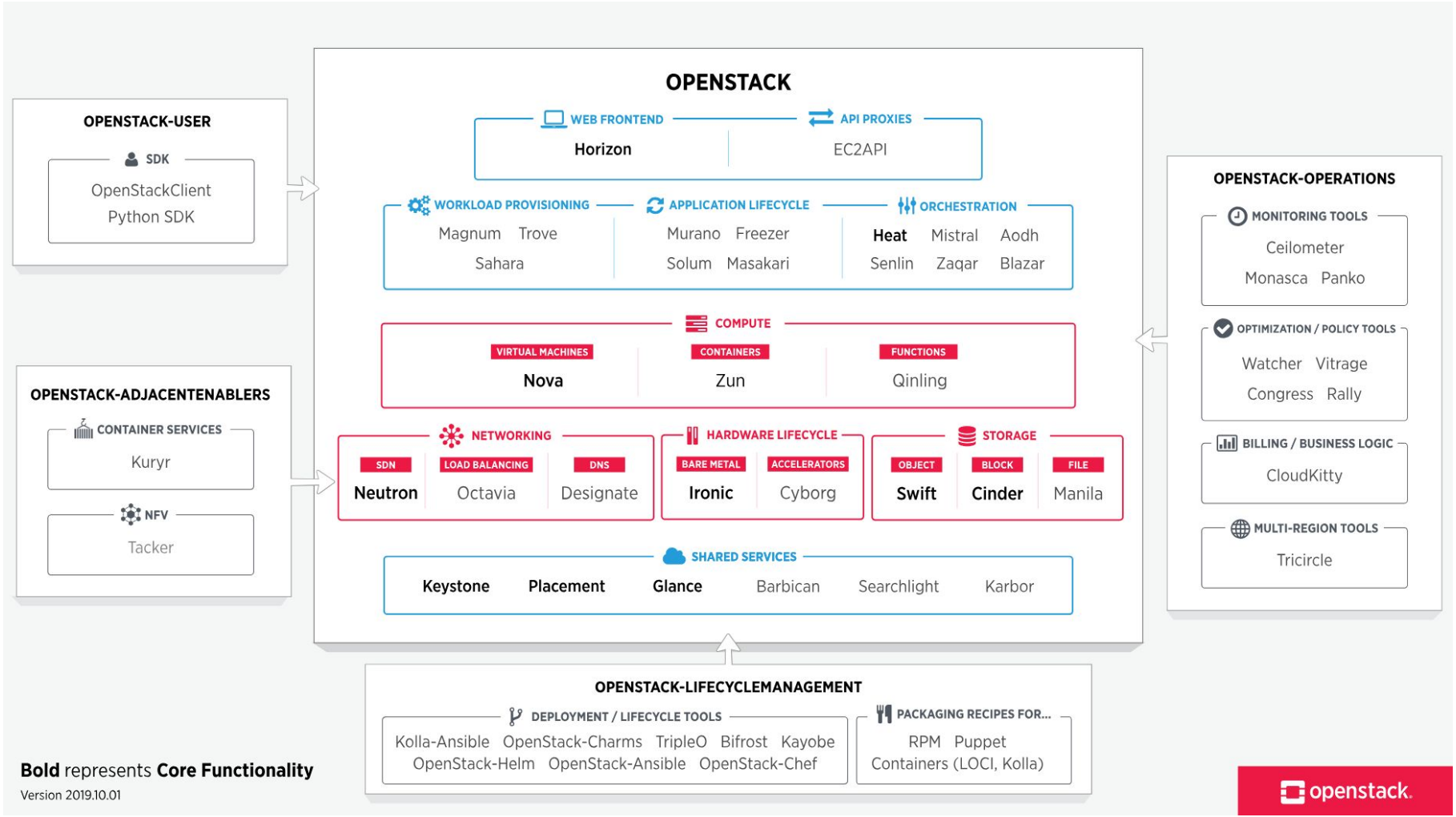
StackHPC

OpenStack Users

StackHPC




Deploy third party services such as
Or use built in tools
Kubernetes
CloudFoundry
Terraform
OpenStack SDK
Horizon Web UI
Bare Metal
Virtual Machines
Containers
Shared networking and storage resources
openstack.



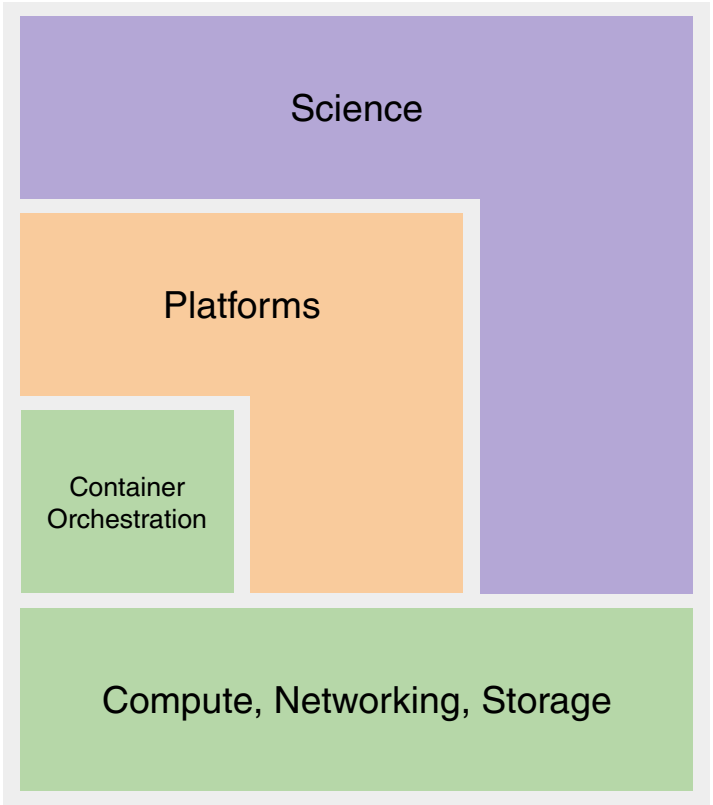
Bold represents **Core Functionality**

Version 2019.10.01

openstack.

Scientific OpenStack

StackHPC



Developer

Platform Ops

Infrastructure Ops

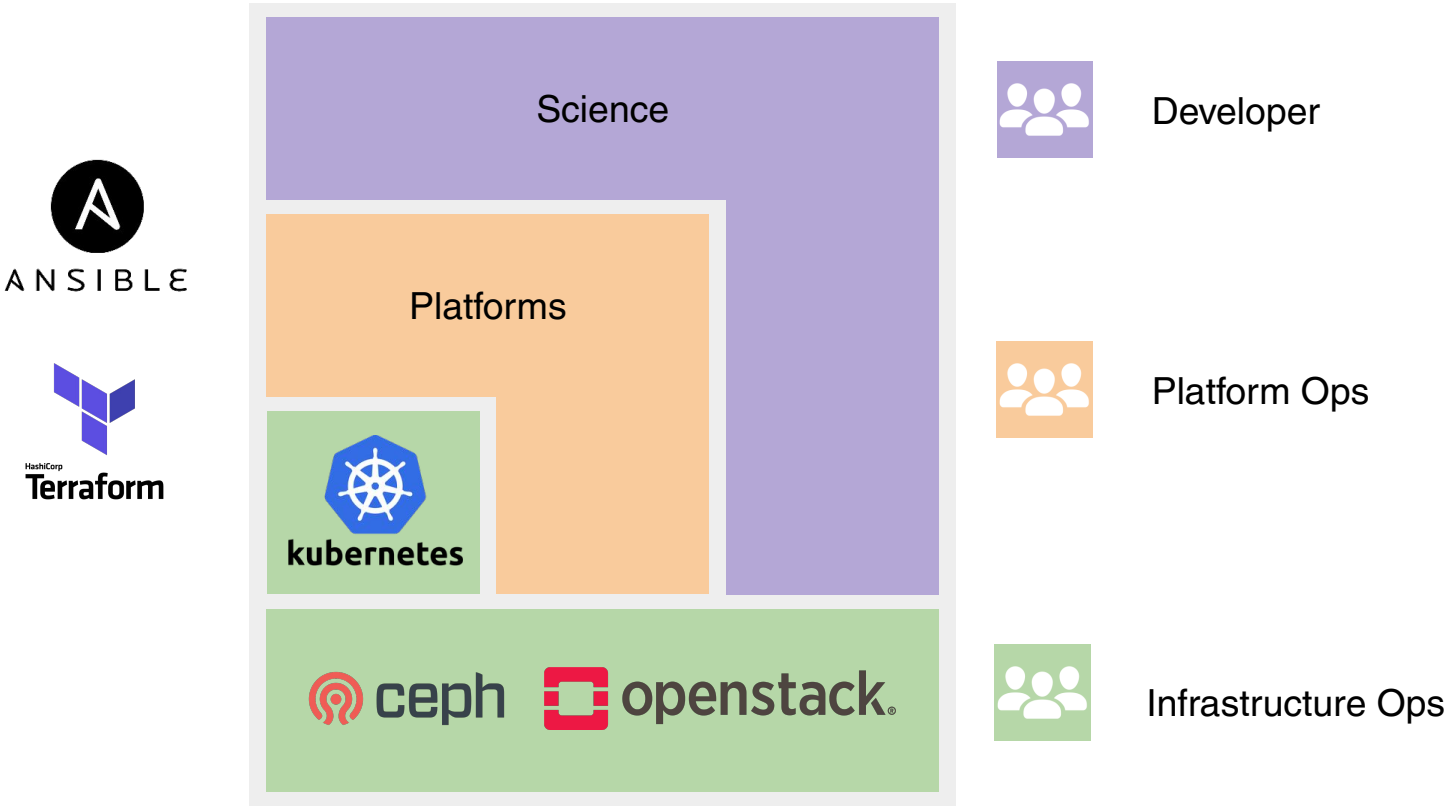
Scientific OpenStack Digital Assets

StackHPC

- Driven by Science Communities needs
- Provides Reference Platforms
- Reference OpenStack Architecture and Configuration tuned for Scientific Computing
- Tooling to help Operate OpenStack

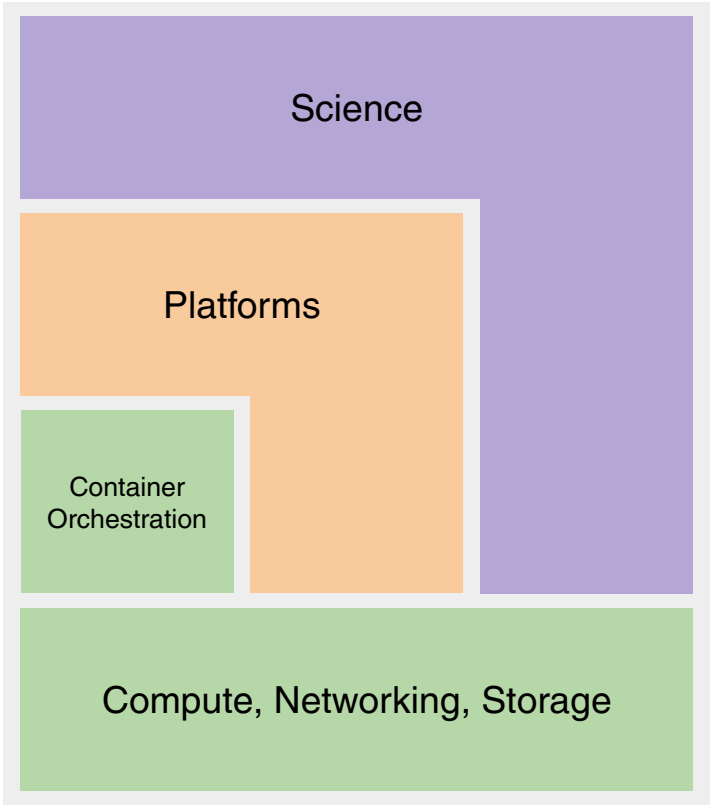
Scientific OpenStack

StackHPC



Scientific OpenStack

StackHPC



Developer

Platform Ops

Infrastructure Ops

OpenStack Compute

StackHPC

OpenStack Compute

StackHPC

Flavors

- Baremetal (Ironic) or Virtual (KVM)
- RAM, vCPU
- Root Disk and Ephemeral Disk
  - Usually Local Disk
- Maps to specific hardware pool
- Server Groups, Keypairs, Security Groups

Server attached to one or more Networks

Images

- Content of Root Disk
- Provided by upstream distro

Volumes

- Additional Block Storage
- Move between Servers
- Multiple Types possible

Example: Terraform for OpenHPC

StackHPC

```
provider "openstack" {
 cloud = "cumulus"
}

resource "openstack_compute_instance_v2""login" {
 name            = "ohpc-login"
 image_name      = "CentOS7-1907"
 flavor_name     = "general.v1.tiny"
 key_pair        = "johng"
 security_groups = ["default"]

 network {
   name = "cumulus-internal"
 }
}
```

```
resource "openstack_compute_instance_v2""comp" {
 name            = "ohpc-compute-${count.index}"
 image_name      = "CentOS7-1907"
 flavor_name     = "general.v1.medium"
 key_pair        = "johng"
 security_groups = ["default"]
 count           = 5

 network {
   name = "cumulus-internal"
 }
}
```

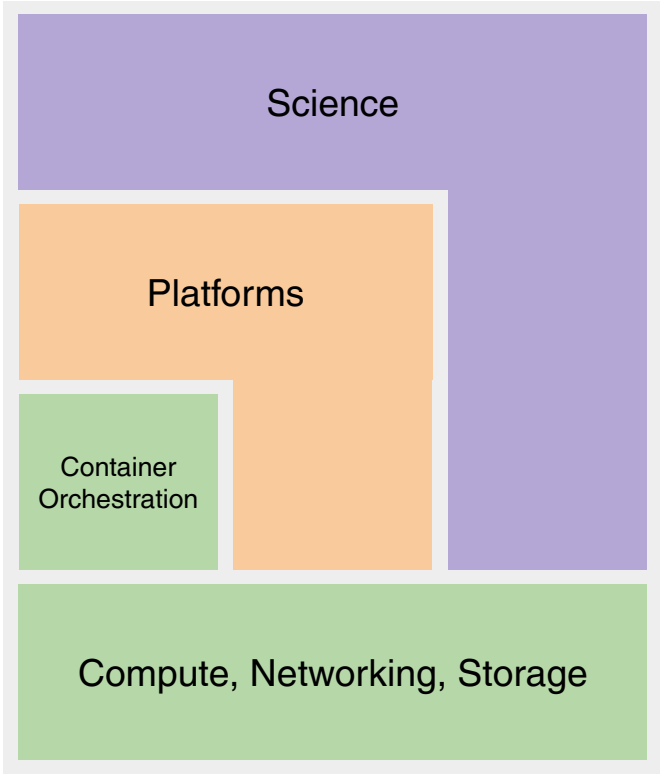
HashiCorp Terraform

OpenStack and AAI

StackHPC

Three separate steps:

- Terraform to OpenStack APIs
- Ansible to OpenStack Servers
- End User to Platform



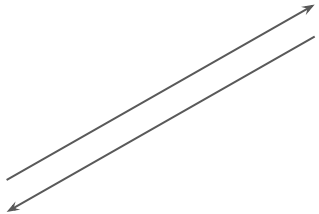
OpenStack and AAI

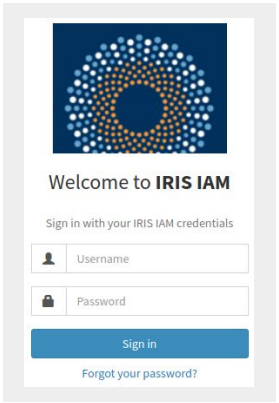
StackHPC

KEYSTONE
an OpenStack Community Project

Projects, Users, Roles

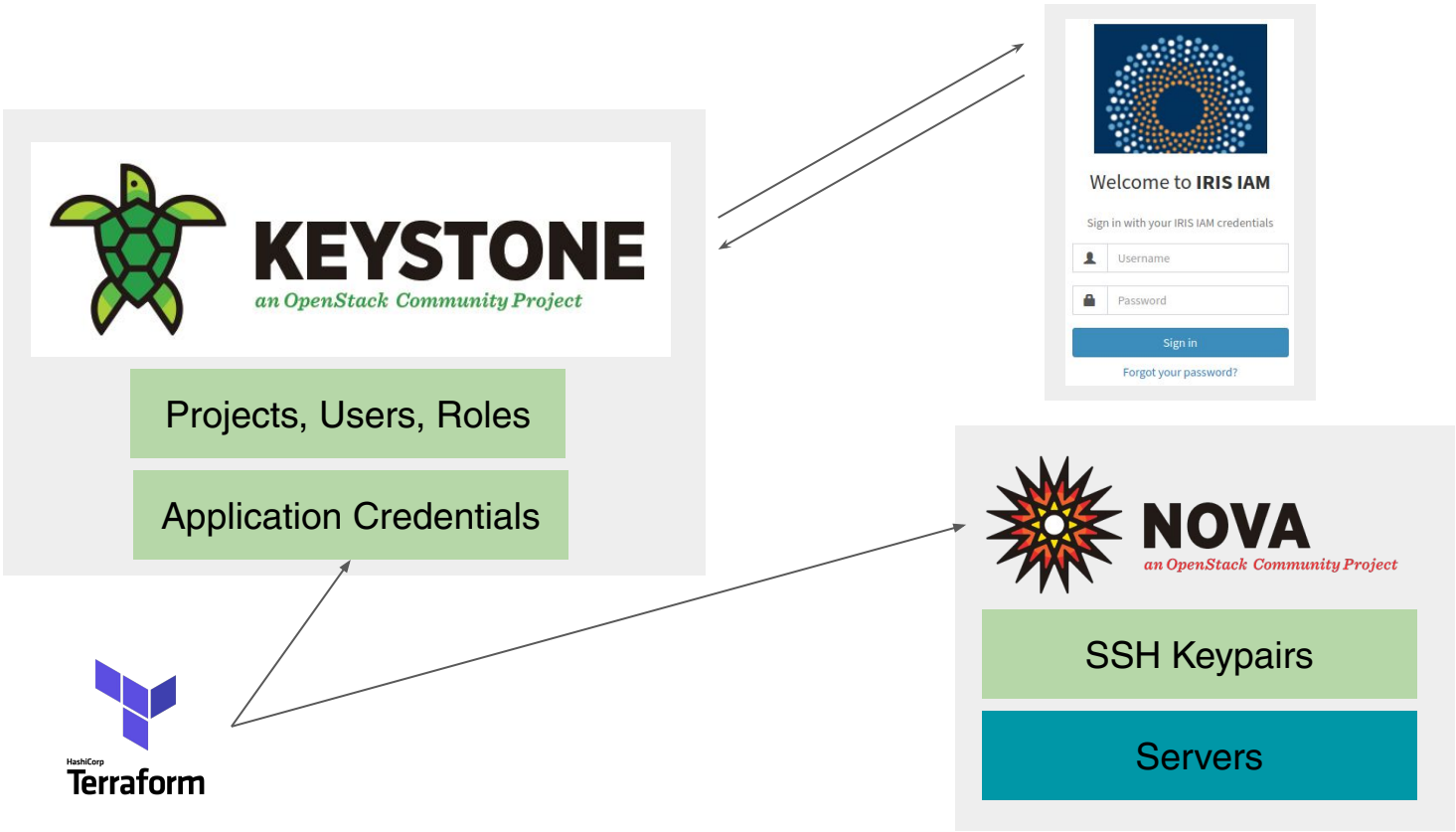
Application Credentials





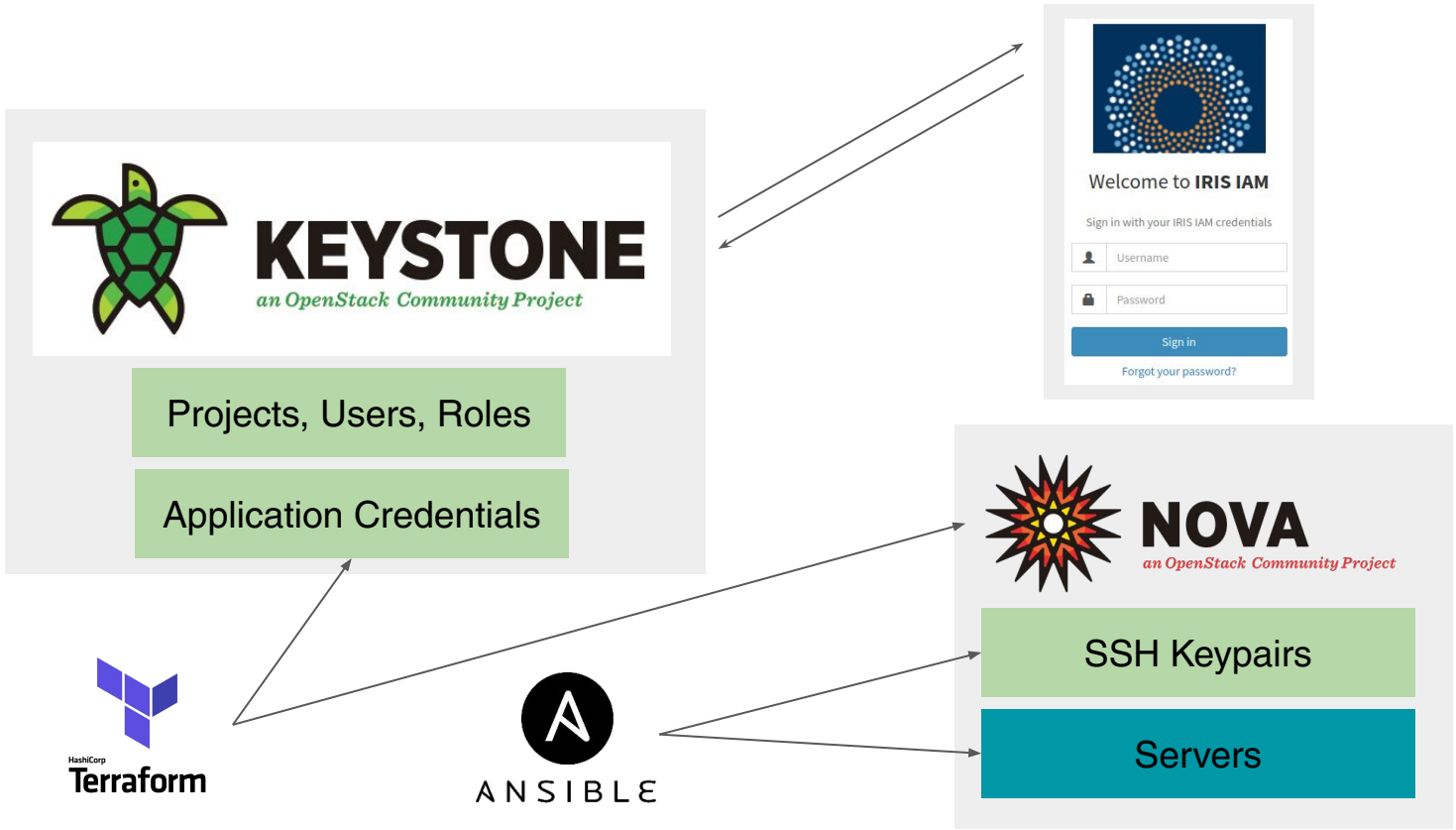
OpenStack and AAI

StackHPC



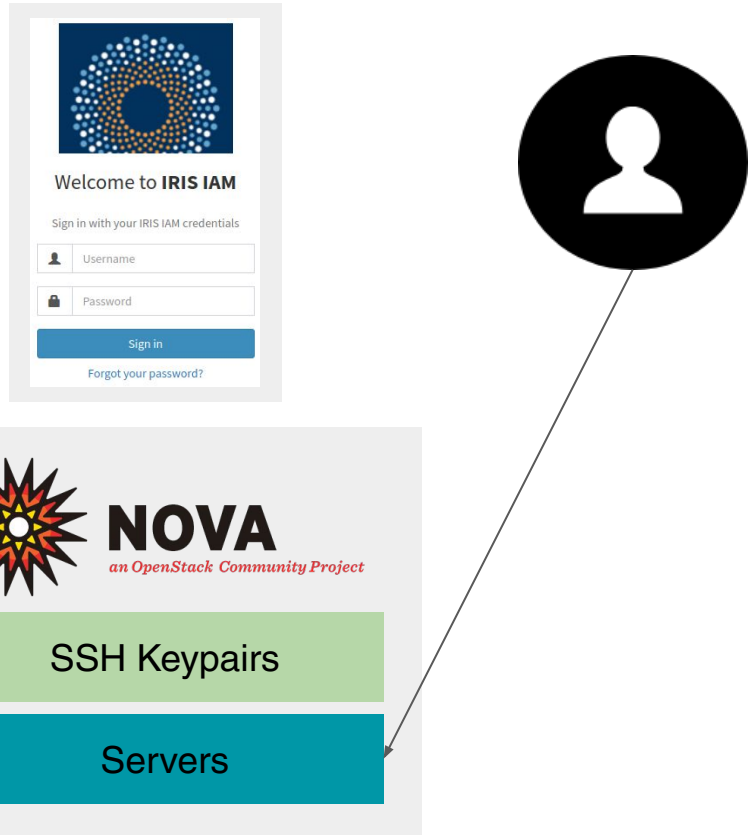
OpenStack and AAI

StackHPC



OpenStack and AAI

StackHPC

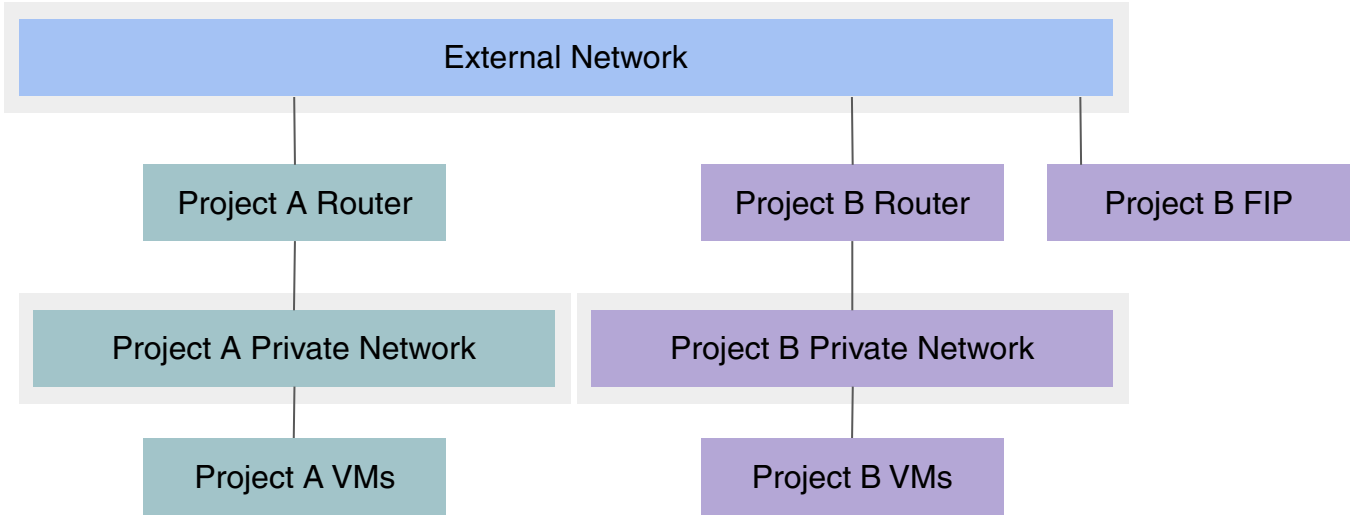


OpenStack Networking

StackHPC

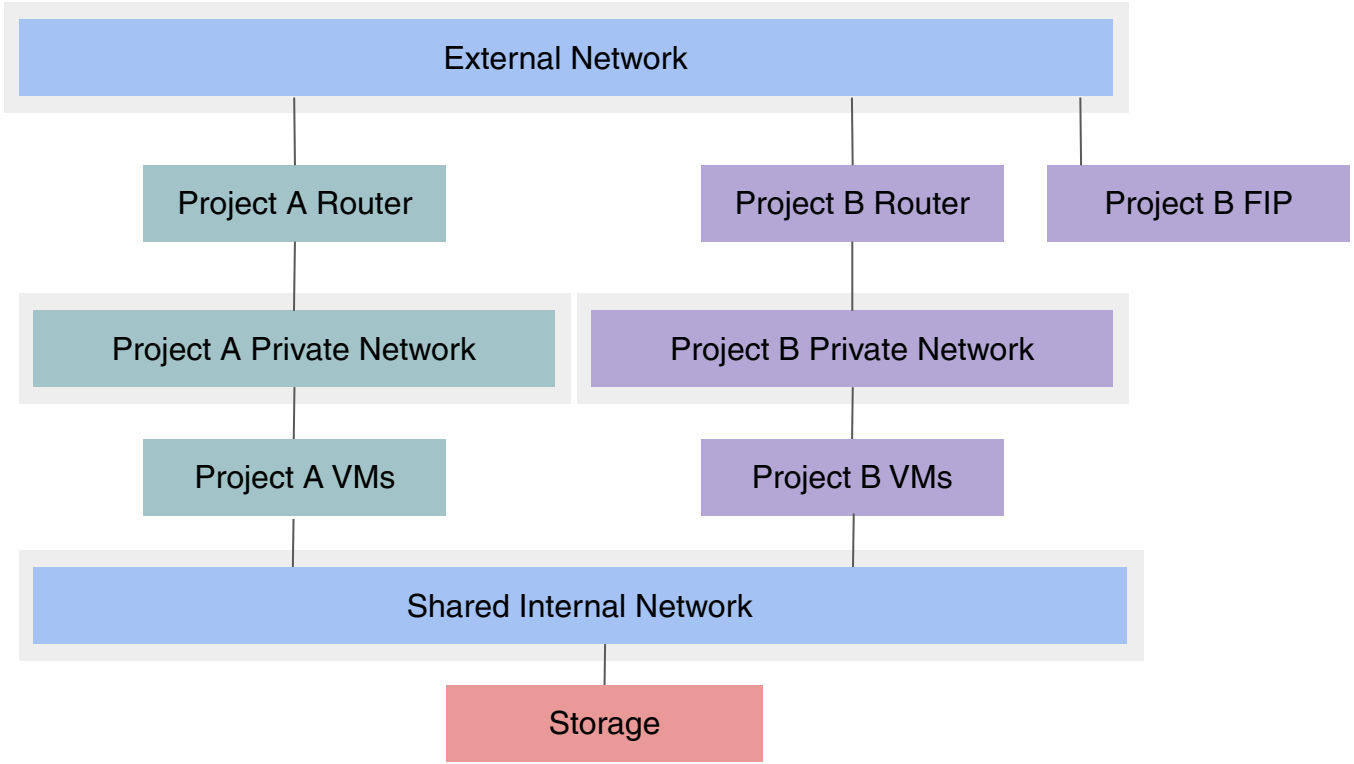
Per Project Networks

StackHPC



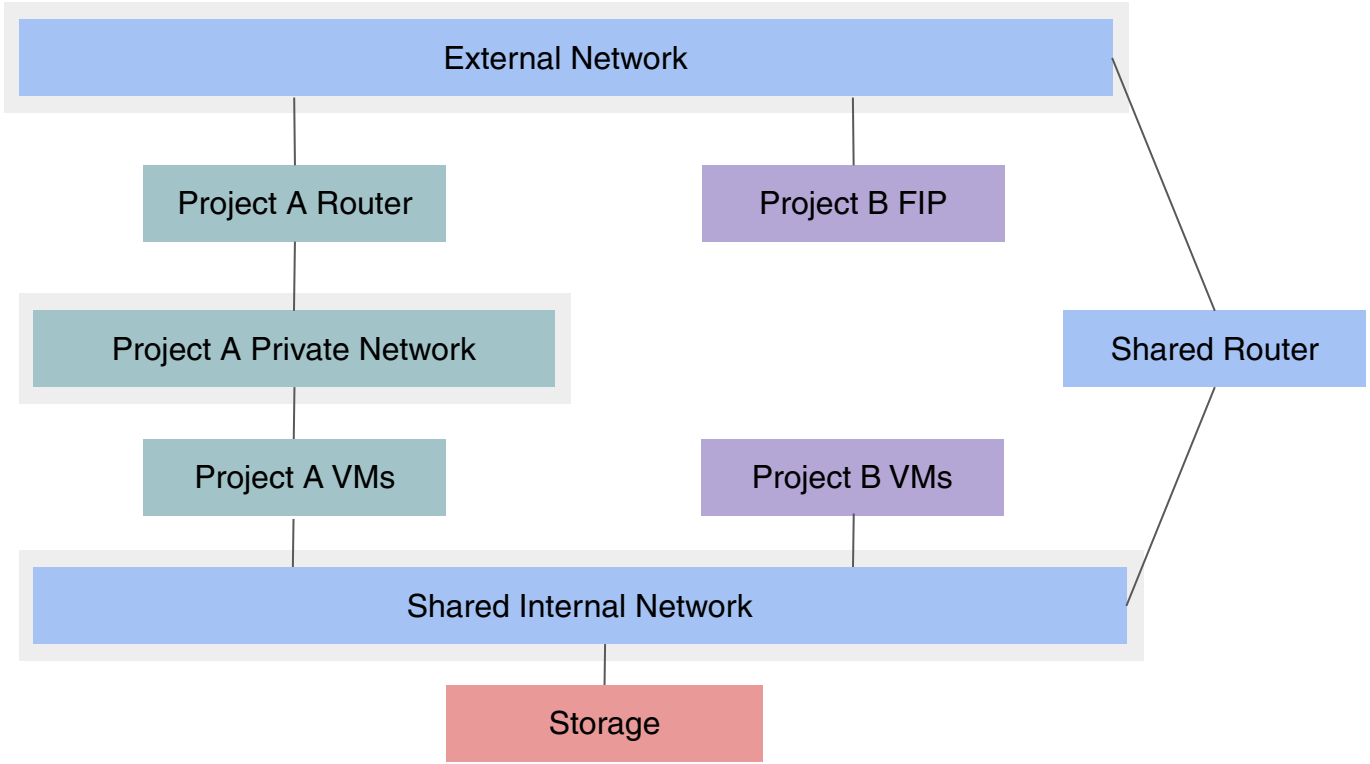
Shared Internal Network

StackHPC



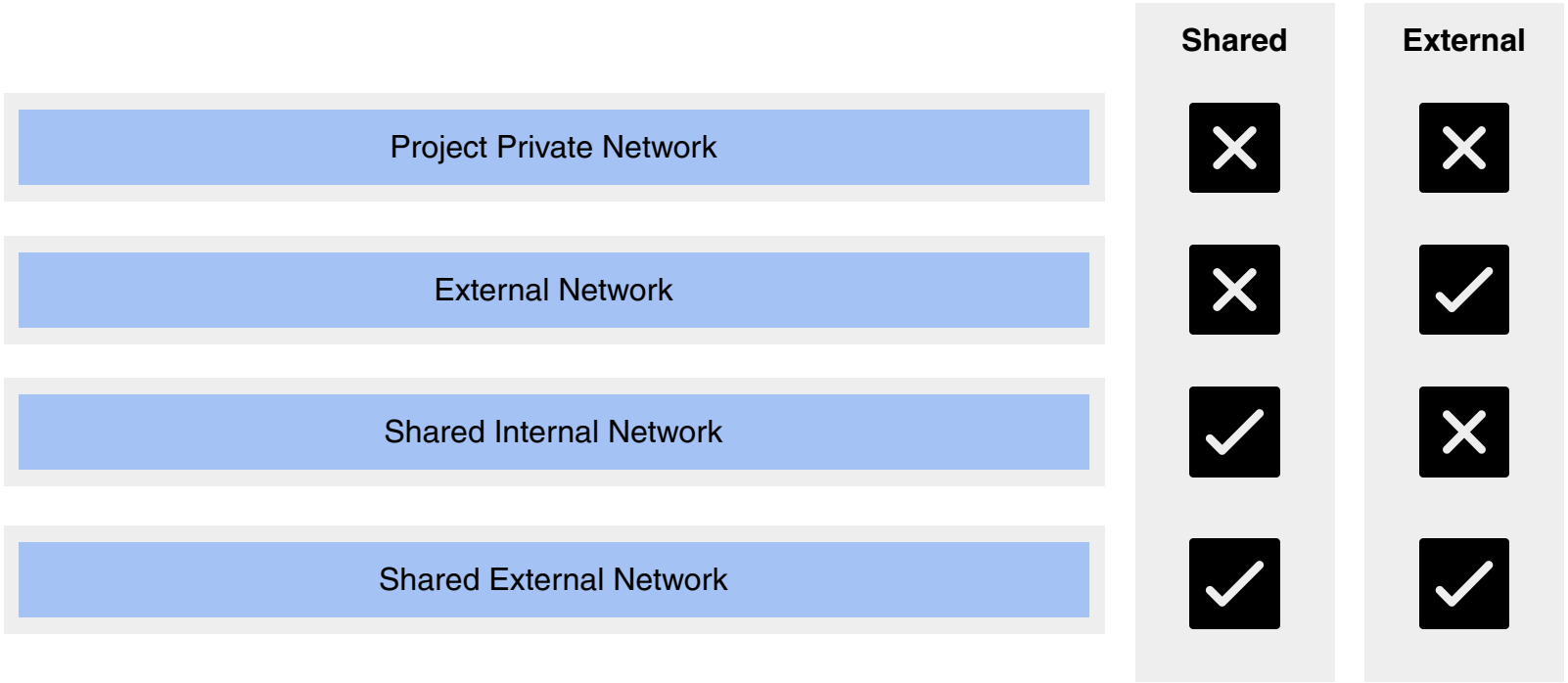
Shared Internal Network

StackHPC



Neutron Network Types

StackHPC



Icons made by
https://www.flaticon.com/authors/freepik

Security Groups

StackHPC

The Default Security Group
does not include SSH access.

OpenStack Storage
Powered by Ceph

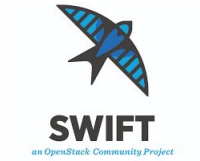
StackHPC

OpenStack Storage

StackHPC

Object Storage

- Ceph integrates with Keystone
- Support Swift and S3 APIs
- Supports Bucket Versioning and Policies
- Large S3 Ecosystem
- Globally accessible API drives adoption



Block Storage

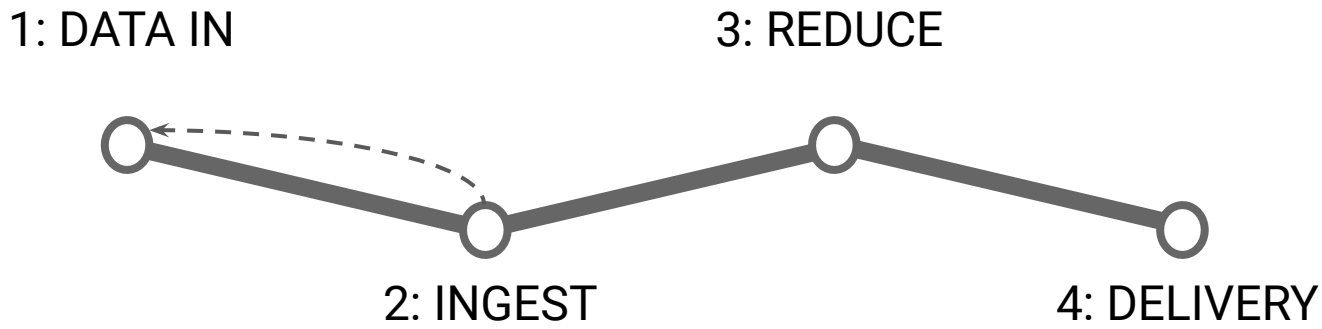
- Volumes attached to Servers
- Basic snapshot support

File Storage

- Create share, control access
- Used by K8s PVC
- Basic snapshot support
- WIP: Lustre via LNET routers

Data from Scientific Instruments

StackHPC



Storage Locality Matters

StackHPC

- Independent OpenStack and Object Storage Regions
  - IRIS has multiple Object Storage endpoints
- Data is local to Compute
  - Transparent data movement is a special case
  - Coordinated data and compute placement required
- Data Management and Data Movement is Critical
  - OpenStack gives you only basic building blocks
  - … but other tools like Rucio and iRODS build on basic services
- Backup is data movement
  - Restic seems popular for backup to S3 APIs

RUCIO
SCIENTIFIC DATA MANAGEMENT

iRODS

restic

Monitoring

StackHPC

OpenStack and Ceph

- Used by Infrastructure Operator
- Monitor system as seen by Platform Operators

- Grafana dashboards
- Prometheus Collectors and Alerts
- ELK for Logs
- WIP: Export metrics to Platforms

Platforms

- Working for Platform Operator
- Monitor system as seen by Scientists (users of platform)

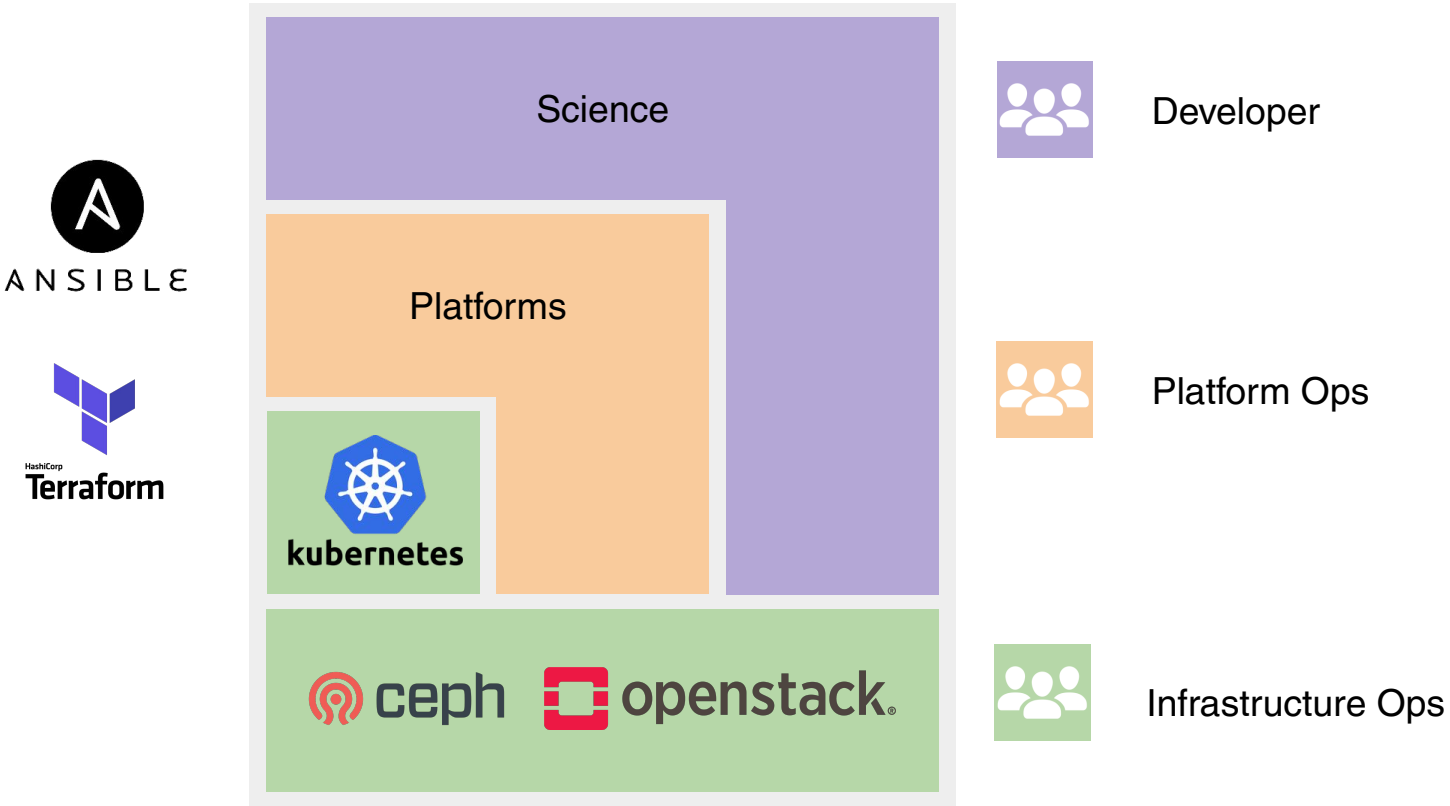
- Grafana dashboards
- Prometheus Collectors and Alerts
- WIP: Loki for Logs

Platforms

StackHPC

Scientific OpenStack

StackHPC



Example Platform

Euclid OpenHPC Slurm

StackHPC

Reference Platforms

StackHPC

Bootstrap vs Bake

StackHPC

Bootstrap

- Boot from Base OS Image
- Cloud-Init injects SSH Key and Network
- Ansible (or similar) to Bootstrap

- Bootstrap can be slow
- Can be hard to fix all binaries

Hybrid: Bootstrap using Container Images

Bake

- Boot from a custom pre-baked image
- Cloud-Init injects SSH Key and Network
- Small user-agent script to “join” cluster

- Boot from custom image can be slow
- No in place updates, must rebuild

Base Infrastructure Types

StackHPC

OpenStack Server

- Terraform creates Infrastructure
  - Use base OS image
  - No difference for Baremetal vs VMs
- Ansible modifies base OS to deploy Platform stack and Monitoring stack

Kubernetes

- Terraform creates K8s cluster
  - Manila CSI, cluster-autoscaler, Octavia Ingress, Prometheus, Grafana
- Ansible deploys apps via Helm, Kustomize

Example: Terraform for OpenHPC

StackHPC

```
provider "openstack" {
 cloud = "cumulus"
}

resource "openstack_compute_instance_v2""login" {
 name            = "ohpc-login"
 image_name      = "CentOS7-1907"
 flavor_name     = "general.v1.tiny"
 key_pair        = "johng"
 security_groups = ["default"]

 network {
   name = "cumulus-internal"
 }
}
```

```
resource "openstack_compute_instance_v2""comp" {
 name            = "ohpc-compute-${count.index}"
 image_name      = "CentOS7-1907"
 flavor_name     = "general.v1.medium"
 key_pair        = "johng"
 security_groups = ["default"]
 count           = 5

 network {
   name = "cumulus-internal"
 }
}
```

HashiCorp Terraform

Example: Terraform for K8s

StackHPC

```
provider "openstack" {
 cloud = "cumulus"
}
resource "openstack_containerinfra_clustertemplate_v1""kubernetes_template" {
 name = "kubernetes-1.15.3"
}

resource "openstack_containerinfra_cluster_v1""cluster" {
 name                = "my_test_k8s"
 cluster_template_id = "${openstack_containerinfra_clustertemplate_v1kubernetes_template.id}"
 master_count        = 2
 node_count          = 3
 keypair             = "johng"
}
```

HashiCorp Terraform

Reference Platforms

StackHPC

OpenHPC Slurm

- EUCLID single site, using Manila
- IRIS IAM via Open OnDemand
- Investigate autoscaling

Jupyter Hub

- Minimise post-Mangum steps
- Use Cluster Autoscaler and Ingress
- Considering: Spark, Dask/Pangeo

Baremetal via Ironic

StackHPC

- Maximum Performance
- Latency sensitive, e.g. MPI
  - RDMA Ethernet, RoCEv2 or iWARP
  - Dataset larger than single node's memory
  - SR-IOV is a possible alternative
- Trust issues around direct access to hardware
  - Cleaning is already supported
  - Can be avoided by providing a "Managed" service
- Optionally used by Kayobe for Server Lifecycle Management

Improving Hardware Utilization

StackHPC

- Map Infrastructure monitoring back to APEL usage
- Isolated platforms can be costly
  - Make better, dynamic, resource requests (K8s and Slurm autoscale)
  - More flexible shared platforms (podman/charliecloud in Slurm, etc)
- Building blocks
  - Make space for GridPP Backfill, Reclaim space from GridPP Backfill
  - Blazar to reserve space
  - External Reaper (CERN Preemptables, CPU hour credits and Quota)
- Digital Asset: Document Best Practice Outcomes

BLAZAR
an OpenStack Community Project

What's next?

StackHPC

Scientific OpenStack Digital Assets

StackHPC

- Documentation on OpenStack Best Practices
- Demos at IRIS Face to Face
- More feedback welcome!

@stackhpc
@johnthetubaguy
@oneswig

StackHPC