Glint - Image Distribution in a Multi-Cloud Environment

Colin Leavett-Brown, University of Victoria

Randall Sobie, Frank Berghaus, Andre Charbonneau, Ron Desmarais, Ian Gable, Micheal Paterson, Ryan Taylor, Alex Lam, Colson Driemel
Glint - Image Distribution in a Multi-Cloud Environment

• Overview:
  - Who we are and what we do
  - Why did we develop Glint?
  - All about Glint
  - Summary
Glint - Who we are...

- High Energy Physics (HEP) research team at the University of Victoria.
- Developing computing and networking solutions for HEP.
- Support other scientific disciplines, particularly those which leverage High Throughput Computing in a batch environment.
- Actively working on virtualization and cloud computing since 2006.
High Energy Physics
(Particle Physics)

The area of physics that studies the fundamental particles of nature and their interactions.

• Primary interests are in the ATLAS and Belle-II experiments
• ATLAS discovering the source of mass; the Higgs Boson
• Belle-II probing why our universe is not made equally of matter and anti-matter
Glint – LHC, Geneva and ATLAS Detector Under Construction
Glint – ATLAS production jobs on distributed cloud in 2014

- Monte Carlo simulation jobs
- Approximately 12 hours duration each
- 95%+ efficient
- 1.2 million jobs this year
- Three continents
Glint – Belle-II production jobs, worldwide

- 251,000 jobs in 30 days.
- UVic third highest producer at 8.6% of the total jobs run
- Currently running 12,000 jobs per day
- 10+ clouds, including 3 commercial
Glint – Using clouds on three continents

- Nimbus
- OpenStack
- EC2 & GCE
Glint – High Throughput Computing work flow
Glint – High Throughput Computing work flow
Glint – High Throughput Computing work flow
Glint – HTC on distributed clouds
Glint – HTC on distributed clouds
Glint – HTC on distributed clouds

- HTCondor
- CloudScheduler
Glint – HTC on distributed clouds

Job

CloudScheduler

HTCondor

Job

Colin Leavett-Brown, University of Victoria
Glint – HTC on distributed clouds

CloudScheduler

HTCondor

Jobs

Colin Leavett-Brown, University of Victoria
Glint – HTC on distributed clouds
Glint – HTC on distributed clouds

HTCondor

VM

CloudScheduler
Glint – HTC on distributed clouds

HTCondor

CloudScheduler

VM
Glint – HTC on distributed clouds
Glint – HTC on distributed clouds

HTCondor

CloudScheduler

VM

VM

VM

VM

VM

VM
Glint – HTC on distributed clouds

CloudScheduler

HTCondor

VM

VM

VM

VM
Glint – HTC on distributed clouds
Glint – HTC on distributed clouds

CloudScheduler

HTCondor

Software Distribution
- CernVM-FS
- Squid
- Shoal

Data Distribution
- Storage Element
- Unified
- Generic
- Redirector
- HTTP
Glint – HTC on distributed clouds

ATLAS / PanDA

Belle II

DIRAC

HTCondor

CloudScheduler

Software Distribution

CernVM-FS + Squid + Shoal

Data Distribution

Storage Element + Unified Generic Redirector + HTTP

Colin Leavett-Brown, University of Victoria
Glint – HTC on distributed clouds

What about VM image distribution?
Glint – HTC on distributed clouds

- Nimbus
- CloudScheduler
- HTCondor
- Data Distribution
- Software Distribution
- ATLAS / PanDA
- Belle II
- DIRAC
Glint – HTC on distributed clouds

ATLAS / PanDA

HTCondor

Repoman

CloudScheduler

HTTP/HTTPS

Nimbus

Data Distribution

Software Distribution
Glint – HTC on distributed clouds

- ATLAS / PanDA
- HTCondor
- CloudScheduler
- Repoman
- Data Distribution
- Software Distribution
- Glance
- Glance
- Glance
- Glance
Glint – HTC on distributed clouds

ATLAS / PanDA

HTCondor

CloudScheduler

Repoman

Data Distribution

Software Distribution

Belle II

DIRAC

Glance
Automated cloud computing with manual image distribution

Manual image distribution is manageable with a few clouds and images, but...

- Nimbus
- OpenStack
Glint – Manual image distribution no longer practical

- Manual Distribution:
  - Time Consuming
  - Error prone:
    - UUIDs, AMIs, Names

- OpenStack
- Nimbus
- EC2 & GCE
Glint - Why did we develop Glint?

- We needed something to transfer images to many sites
Glint - Why did we develop Glint?

- We needed something to transfer images to many sites
- Blueprints

OpenStack Image Registry and Delivery Service (Glint)

Image Transfer Service
Glance » Blueprints » Image Transfer Service
Registered by John Bresnahan on 2013-02-25

OpenStack could benefit from a service designed specifically for large data transfers. This service would be on both the source and the destination of an image transfer and be in a position to negotiate the best protocol (http, bittorrent, etc.) and prevent the transfer from over using I/O resources (both disk and network). A beginning of these ideas can be found here: http://tropicaldevel.wordpress.com/2013/01/11/an-image-transfers-service-for-openstack/

Blueprint information
- Status: 
- Approver: John Bresnahan
- Started:
- Priority: 
- Drafters:

Related branches
Related bugs
Scripts

https://blueprints.launchpad.net/glance/+spec/image-transfer-service
Glint - Why did we develop Glint?

- We needed something to transfer images to many sites
- Blueprints – abandoned 2014-02-07
- Looked at Staccato (John Bresnahan)

OpenStack Image Registry and Delivery Service (Glint)

Image Transfer Service

OpenStack could benefit from a service designed specifically for large data transfers. This service would be on both the source and the destination of an image transfer and be in a position to negotiate the best protocol (http, bittorrent, etc) and prevent the transfer from over using I/O resources (both disk and network). A beginning of these ideas can be found here: [http://tropicaldevel.wordpress.com/2013/01/11/an-image-transfers-service-for-openstack/](http://tropicaldevel.wordpress.com/2013/01/11/an-image-transfers-service-for-openstack/)

Related branches
Related bugs
Scripts

https://blueprints.launchpad.net/glance/+spec/image-transfer-service
Glint - Design objectives

- Reuse existing functionality, only develop missing pieces.
  - Keystone for authentication.
  - Glance data movement
  - Local Glance repository as “home” repository.
- Distribution/instantiation by image name.
- Pluggable architecture to allow support for other cloud types.
- Threaded to perform data movement in parallel.
- Security and data integrity.
- Use standard OpenStack development architecture.
Glint – Four components

- glint – image distribution service
- glint-horizon – horizon dashboard user interface
- glint-service – installation scripts
- glint_backup – stand-alone backup utility
Glint – Uses Glance (services, images and meta-data)
Glint – Uses Glance (services, images and meta-data)

Meta-data DB

Images

image_ID, owner=<tenant>

Backend Storage
Glint – Uses Horizon, Keystone and Glance
Glint – Uses Horizon, Keystone and Glance

Local Cloud

Keystone

Horizon

Glance

token
Glint – Uses Horizon, Keystone and Glance

Local Cloud

Keystone

Glance

Horizon

image_data

token
Glint – Has its own meta-data (and a cache)

<table>
<thead>
<tr>
<th>Repositories</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Cloud, URL, Type, Description</em></td>
</tr>
<tr>
<td>Repositories</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>Cloud, URL, Type, Description</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Credentials</td>
</tr>
<tr>
<td>Local_tenant, Tenant, User, Password</td>
</tr>
</tbody>
</table>

Glint – Has its own meta-data (and a cache)
Glint – Has its own meta-data (and a cache)

Repositories

Cloud, URL, Type, Description

Credentials

Local_tenant, Tenant, User, Password

Users (dynamic session status)

Local_tenant, Cloud, Token
Glint – Distributing images

Local Cloud
- Keystone
- Glance

Horizon
- Glint Cache
- Glint (horizon)

Remote Clouds
- Keystone
- Glance

Glint (service)
Glint – Distributing images

Local Cloud
- Keystone
- Glance

Remote Clouds
- Keystone
- Glance

Copy images

Glint Cache

Glint (service)
Glint – Distributing images

Local Cloud
- Keystone
- Glance

Remote Clouds
- Keystone
- Glance

Glint Cache
- Glint (service)
- Glint (horizon)

copy_images
Glint – Distributing images

Local Cloud
- Keystone
- Glance

Remote Clouds
- Keystone
- Glance
- Glint (service)
- Glint Cache

Copy images flow:
- From Local Cloud to Remote Clouds
- Glint (service) to Glint Cache

Glint – Distributing images

- Local Cloud
  - Keystone
  - Glance
- Horizon
  - Glint (horizon)
- Glint Cache
  - Glint (service)
- Remote Clouds
  - Keystone
  - Glance

copy_images
Glint – Distributing images

Local Cloud

- Keystone
- Glance

Remote Clouds

- Keystone
- Glance

Glint Cache

Copy_images
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface

How to use Glint to distribute images.

Workflow:
1. Select the local tenant/project
2. Add remote repositories.
3. Add remote credentials
4. Distribute images.
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface

![Horizon dashboard interface](image-url)

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Rat01 (ATLAS)</th>
<th>Alto (atlas)</th>
<th>Mouse (HEP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fedora-image</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>belle-2-dem</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>atlas-demo-2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>atlas-demo-1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ucernvm-prod.1.18-4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ucernvm-devel.1.18-4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fedora</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Horizon dashboard interface
Glint – Backup utility

- Incremental backup of local Glance repository to local directory.
- Saves both images and metadata.
- Stand-alone utility run by cron under non-privileged user (requires keystone admin account).
- Simple JSON configuration:

```json
{
    "glint_backup_admin_pw_key": "hiera_key_for_password_value",
    "glint_backup_admin_user": "admin_user_name",
    "glint_backup_admin_tenant": "admin_tenant_name",
    "glint_backup_auth_URL": "http://192.168.1.1:5000/v2.0",
    "glint_backup_dir": "/local/backup/glance",
    "glint_backup_hiera_config": "/usr/local/etc/glint/hieradata/hiera.yaml",
    "glint_backup_logfile": "/var/log/glint/glint_backup.logfile",
    "glint_backup_testing": "False"
}
```
Glint – Useful links

- https://github.com/hep-gc/glint
  - Source code repository for glint service and backup utility.
- https://github.com/hep-gc/glint-horizon
  - Source code repository for modified Horizon dashboard.
- https://github.com/hep-gc/glint-service
  - Repository for installation scripts.
- http://heprc.phys.uvic.ca
  - Team website: publications, reports, and presentations.
Glint – Command line interface

• New Feature (https://github.com/hep-gc/glint/issues/30):

Suggested utility/syntax:

Usage:

```
glint <subcommand> <parameters>
```

(Note: Use an openrc.sh to establish authentication to a tenant on the local cloud prior to issuing the glint command.)

Subcommands:

- `repository-add --name <*repository_name> --url <*authorization_url> ---format <*cloud_type | OpenStack> –description <*description>`
- `repository-delete --name <*repository_name>`
- `repository-edit --name <*repository_name> --url <authorization_url> --format <cloud_type | OpenStack> –description <description>`
- `repository-list`
- `credentials-add --name <*repository_name> --tenant <*tenant_name> --username <*>user_name> --password <*>password>`
Glint – How to try it

1) Get a copy of Glint's installation scripts (this is all you need) git clone https://github.com/hep-gc/glint-service.git.

2) “cd” into the "glint-service" directory.

3) Double check that ports 8080, 8483, and 9494 are open on your glint test machine.

4) Change the configuration files to match your installation (i.e., point glint installation to your current openstack deployment). These are the glint_services.yaml and glint_setup.yaml files.

5) Start Download, Install and Setup of glint using "sudo python glint_setup.py install" (this takes about 10-15 minutes).
Glint - Image Distribution in a Multi-Cloud Environment

• Summary:
  - We are continuing to develop and enhance it.
  - We believe it has applicability and general utility for a wider audience.
  - We would like to see Glint incorporated into OpenStack.
  - Acknowledgements: UVic, IPP, CANARIE, and NSERC.
Glint - Image Distribution in a Multi-Cloud Environment

• Summary:
  - We are continuing to develop and enhance it.
  - We believe it has applicability and general utility for a wider audience.
  - We would like to see Glint incorporated into OpenStack.
  - Acknowledgements: UVic, IPP, CANARIE, and NSERC.

- Colin Leavett-Brown <crlb@uvic.ca>