Distributed Cloud Computing in High Energy Physics

Randall Sobie
Institute of Particle Physics
University of Victoria
High Energy Physics
(Particle Physics)

The area of physics that studies the fundamental particles of nature and their interactions.
Is the Higgs boson the source of mass of our fundamental particles?
Why is the universe made of matter and not equal amounts of matter/antimatter?
What is origin of Dark Matter and Dark Energy?

We do not know the composition of 95% of the universe.

Temperature of the universe
WMAP satellite
Understanding our World
Large Hadron Collider at CERN in Geneva

- 27 km in circumference
- 100 m underground
- Probes distances of $10^{-20}$ m
Blue tubes contain the two beam pipes and magnets at 1.8 degrees Kelvin
ATLAS Detector

44 m wide
25 m diameter
ATLAS detector during construction in 2005
Physicists Find Elusive Particle Seen as Key to Universe

Scientists in Geneva on Wednesday applauded the discovery of a subatomic particle that looks like the Higgs boson.
40 million collisions per second

100,000 collisions selected

200 events per second

WLCG Computing Grid
CERN Tier0
10 Tier1 sites
60+ Tier2 sites
150 PB data
Long-term preservation of software and data of HEP experiments

Utilize special computing resources attached to the detectors

Simplify the management of heterogeneous in-house resources

Distributed cloud computing using HEP and non-HEP clouds

Use commercial clouds for exceptional computing demands
Distributed cloud computing

*Grid of Clouds*

Use multiple, heterogeneous *IaaS* clouds for *batch* workloads

**Design Goals**

- Independent of the *IaaS* cloud type or hypervisor
- No application software on any cloud
  (No HEP specific software)
- Use dedicated and opportunistic resources
- Leverage existing cloud or batch computing software

*Sky Computing*
*K. Keahey et al*
Distributed Batch Cloud Computing

- Job
  - Submit user script
- HTCondor
- CloudScheduler
- Compute Cloud
  - VM
  - VM Image Repository
  - Starts job
  - Start VMs

Randall Sobie  IPP/UVictoria
**HTCondor**

Designed as a cycle scavenger

Ideal as a job scheduler for a dynamic environment

Job description file
- VM image
- RAM
- Number cores
- Storage space
**Cloud Scheduler**  
Custom component

Reviews the requirements of jobs in the HTCondor queue

Requests the boot of user images

Monitors the VMs

We run multi-core VMs
VM remains active if jobs in queue
Work flow

Cloud Scheduler

HTCondor JobQueue

Discover User-Job

UserJob

User VM

IaaS Clouds
HTCondor
JobQueue
UserJob

Cloud Scheduler

Boot User-VM on a cloud

User VM

IaaS Clouds

Randall Sobie  IPP/UVictoria
Work flow

VM registers with HTCondor

HTCondor
JobQueue

UserJob

Cloud Scheduler

User VM

IaaS Clouds
Work flow

Cloud Scheduler

HTCondor dispatches User-Job to VM

User Job

User VM

IaaS Clouds
Cloud resources

- OpenStack
- Nimbus
- EC2/GCE
There are four HTCondor/CloudScheduler instances in production
Three HEP (ATLAS-Europe, ATLAS-North America and BelleII), and astronomy

Each has its own set of clouds
Running VMs

Number of 8-core VMs (15 weeks)
Each colour is a different cloud
Each HTCondor/CloudScheduler instance has different clouds
Cloud evaporation and condensation

Number of 8-core VMs
Operate three HEP HTCondor/CS Instances

ATLAS
North America (2.5 years)
Europe (8 months)

BelleII
North America (8 months)

Over 1.6M Jobs Completed

Over 140,000 Jobs Completed

Average work load 1500 cores
Peak work 2500 cores
Evolution of the distributed cloud

Managing VM images over multiple clouds

Micro VM images
Evolution of the distributed cloud

Remote access to application software

Site-specific contextualization
Evolution of the distributed cloud

Application data federation
Accessing distributed data repositories
We need to manually upload our images to each site
Each site has different credentials
VM management in a distributed cloud

We need to manually upload our images to each site. Each site has different credentials.
Glint – a VM distribution service

Glint provides the user the ability to manage their images on multiple OpenStack clouds

Integrated in the OpenStack Glance GUI

Horizon dashboard
Glint VM distribution service

Keystone authentication and authorization

Exploring how to make it work with other clouds such as EC2 and GCE
Application software distribution

- **OpenStack**
  - **Glance**
    - **VM**

  - **HTTP network read-only file system**
    - Also delivers operating system software

  - **Highly scalable**
  - **Multiple levels of caching**

- **CVMFS Server**
  - **CERN VM file system**

- Reduces the size of the VM to few MB
- Micro-VM images

Randall Sobie  IPP/UVictoria
Rather than query the CVMFS master repository we use Squid HTTP caches

Approximately 100 Squid caches distributed around the world

Each VM uses a statically assigned Squid cache
Application software distribution III

Shoal – a group of Squids

- Shoal keeps a list of active Squid caches
- Discovers new Squids and removes inactive Squids
- VM queries Shoal for the location of the nearest Squid

[github.com/hep-gc/shoal](http://github.com/hep-gc/shoal)
Figure 1. A $\mu$CernVM based virtual machine is twofold. The $\mu$CernVM image contains a Linux kernel, the AUFS union file system, and a CernVM-FS client. The CernVM-FS client connects to a special repository containing the actual operating system. The two CernVM-FS repositories contain the operating system and the experiment software.

VM (12 MB) with “just enough operating system” and a CVMFS client
Used in production operation starting July 2014
Approximately 100 computing sites
Each HEP site has CPU and storage
10-100G network

We want to use the data at any SE on all clouds
Storage Element (SE)
Data Federation

Generic Data Federator
Developed at CERN

- Presents a unified view of the data file tree to the application
- Dynamic discovery
- Directs the VM to the nearest Storage Element with the input data
- GeoIP information to pick the closest site

We use WebDAV for file access
Open-source HTTP protocol standard
Single file tree of the data
Two copies of this file
Networks are critical to distributed clouds

Data federations are only becoming feasible with 100G research networks
http://supercomputing.uvic.ca

Scientists break world record for data transfer speeds

High-Energy Physicists Smash Records for Network Data Transfer

Canadian physicists achieve 100 gigabit/second transatlantic transmission, enabled by CANARIE and its global partners
Cloud bursting

Cloud Bursting
Using commercial clouds when our existing facilities are fully utilized

Amazon EC2
1000 cores
Spot price tracker

Google Compute Engine
Under test

Each commercial cloud has its own configuration and complexity
Not all sites are connected to the research network
We are often the first user of a new cloud

Clouds are not configured in the same way
(meta-data is often inconsistent)

Cloud software is rapidly changing
(OpenStack did not exist a few years ago)

ID management and authentication

HEP applications stress the non-HEP clouds
(many clouds oversubscribe their resources)
Future

Distributed services
- VM images
- Software caches
- Data federations

Cloud management
- Automate integration and testing of new clouds
- Standardized cloud configuration
- Common authentication and authorization
- Improved monitoring

Commercial clouds
- Viable solution
- Cost trend
Summary

We have designed and constructed a distributed cloud computing system

Currently used for HEP and astronomy applications

Uses IaaS clouds distributed around the world

We are adding new IaaS clouds to the system

We are enhancing its dynamic capabilities (context-aware cloud computing)

Exploring the use of commercial resources for cloud-bursting
Acknowledgements

Contact information: rsobie@uvic.ca
Web sites: http://rjs.phys.uvic.ca/
http://heprc.phys.uvic.ca/home/
**VM contextualization process:**

CloudScheduler boots the user-VM and includes the user-data (specified in the Job Description file)

During the VM boot sequence: cloud-init interprets the user-data and configures the Puppet-agent on the VM

The Puppet-agent on the VM connects to our Puppet master to retrieve the configuration for the VM