An overview of the DII-HEP OpenStack based CMS data analysis

An overview of the DII-HEP OpenStack based CMS data analysis

L Osmani[1], S Tarkoma[1], P Eerola[2], M Komu[3], M J Kortelainen[4], O Kraemer[4], T Lindén[4], S Toor[4] and J White[4]

[1] Department of Computer Science, P.O.B. 68, FI-00014 University of Helsinki, Finland,
[2] Department of Physics, P.O.B. 64, FI-00014 University of Helsinki, Finland, [3] Ericsson Research, 02420 Jorvas, Finland [4] Helsinki Institute of Physics, P.O.B. 64, FI-00014 University of Helsinki, Finland

E-mail: `tlinden@cc.helsinki.fi`

Abstract. An OpenStack based private cloud with the Gluster File System has been built and used with both CMS analysis and Monte Carlo simulation jobs in the Datacenter Indirection Infrastructure for Secure High Energy Physics (DII-HEP) project. On the cloud we run the ARC middleware that allows running CMS applications without changes on the job submission side. Our test results indicate that the adopted approach provides a scalable and resilient solution for managing resources without compromising on performance and high availability.

To manage the virtual machines (VM) dynamically in an elastic fasion, we are testing the EMI authorization service (Argus) and the Execution Environment Service (Argus-EES). An OpenStack plugin has been developed for Argus-EES.

The Host Identity Protocol (HIP) has been designed for mobile networks and it provides a secure method for IP multihoming. HIP separates the end-point identifier and locator role for IP address which increases the network availability for the applications. Our solution leverages HIP for traffic management.

This presentation gives an update on the status of the work and our lessons learned in creating an OpenStack based cloud for HEP.

1. Introduction
The vast amounts of data that needs to be recorded, processed and analyzed by the experiments at the Large Hadron Collider (LHC) has required the creation of a worldwide distributed computing system. The Worldwide Large Hadron Collider Computing Grid (WLCG) has been very successful as it was an necessary and important part enabling the discovery of the Higgs boson in 2012. The future LHC runs will pose even larger computing challenges in terms of data complexity and the amounts of data needed to be handled. Cloud computing can provide a more flexible framework for building distributed computing systems using virtual machines, virtual networks and virtualized storage. Adapting public or private clouds for High Energy Physics (HEP) simulation and data analysis requires some work to provide a seamless and smooth transition from grid technology to cloud technology. The creation of secure distributed computing system poses an important challenge in the cloud environment.

The Datacenter Indirection Infrastructure for Secure HEP Data Analysis (DII-HEP) project is a collaboration between the Helsinki Institute of Physics and the Department of Computer Science at the University of Helsinki funded by the Academy of Finland for the years 2012 –
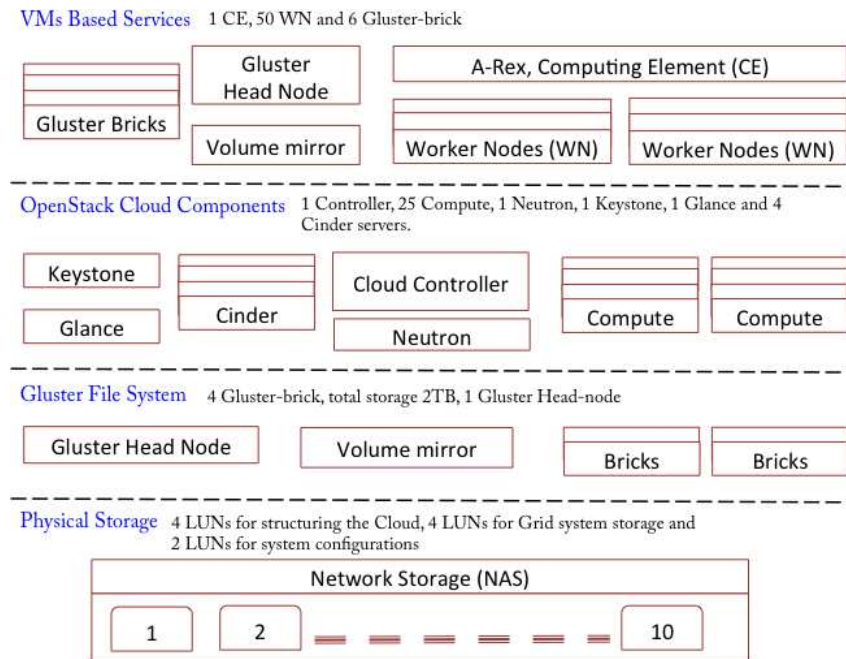
Figure 1. Overall system architecture based on OpenStack and GlusterFS.

2014. The goals of the project are to explore the latest software stacks for distributed computing infrastructures and construct a secure and scalable setup for scientific applications using the CMS analysis and production framework as a test case.

2. Initial setup

Helsinki Institute of Physics runs a *distributed* CMS Tier-2 site T2_FI_HIP [1] based on the Advanced Resource Connector (ARC) middleware [2]. The main CPU resource is the 768 core Linux cluster and ARC Compute Element (CE) Jade shared between CMS and ALICE. The Madhatter Storage Element is using dCache and is located next to Jade. On the Kumpula campus some 10 km from Jade there are two Linux clusters Korundi (400 cores) and Alcyone (892 cores) both with two ARC CEs. Korundi is shared between the Departments of Physics, Chemistry and the Helsinki Institute of Physics. Alcyone is shared between the Departments of Physics, Chemistry, Computer Science and the Helsinki Institute of Physics.

In the DII-HEP project the requirement was to *add and integrate cloud resources to the existing grid production environment* so that all *jobs could be submitted unchanged*. The use of the grid (ARC) interface also on the cloud resource allows all applications to run unchanged. The setup is **cloud based** and **grid enabled**. The cloud resource is OpenStack [3] based and uses the Gluster File System as shown in Fig. 1. It runs on a part of the Ukko cluster (1920 cores) also on the Kumpula campus. ARC is used to control the execution of jobs and it runs on a virtual ARC CE nodeslab-0002. HTCondor [4] is used as the local batch system and CERN VM File System (CVMFS) [5] is used for application software distribution. CMSSW jobs read data from the local SE using the xrootd protocol [6]. Monitoring is done with Graphite, Site Availability Monitoring- (SAM) and Nagios jobs.

OpenStack is an Open Source infrastructure for public and private clouds. It consists of several different components, *Nova* (Compute), *Quantum* / Neutron (Network), *Swift* (Object based storage), *Cinder* (Block storage), *Keystone* (Identity management) and *Glance* (Image management). The DII-HEP project started with the **Folsom** release and the current setup is
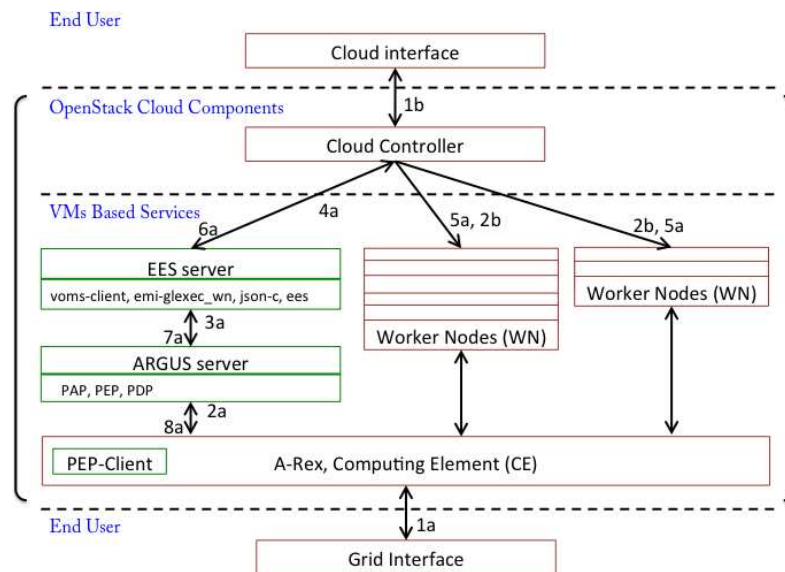
Figure 2. How EES is used with Argus for elastic VM-provisioning.

running on **Havana**, testing has been done also with the **IceHouse** release. GlusterFS provides a shared filesystem for VM-instance booting and provides the shared area for the data of the running jobs, see Fig. 1. OpenStack has been deployed on Ubuntu 12.04 LTS and the VMs are based on Scientific Linux CERN 6.4.

So far the system has run a maximum of 200 concurrent jobs. The average CPU efficiency of all CMS jobs over the previous year is 83 % from the Swedish Grid Accounting System (SGAS). There has been 147290 jobs run for 9163 walltime days for the period August 2013 to August 2014. The CPU efficiency loss due to virtualization was measured to be 4 % by running the HEPSPEC 2006 benchmark on bare metal and on VMs. This setup provides efficient resource management, but semi-statically as the *VMs are manually added or removed*. This initial *semi-static* setup has been presented at the CHEP 2013 conference [7].

3. Elastic setup status
To achieve *elastic VM-provisioning* and improve on the initial semi-static setup we have used the Argus Execution Environment Service (EES) [8] which is part of the European Middleware Iniative (EMI) release. EES is a service that may co-exist with Argus [9] to start VMs or send jobs to a cloud, see Fig 2. EES has been tested with Argus Policy Execution Point demon (PEPd) and it has a plugin written for OpenNebula. A prototype EES OpenStack plugin has been developed in the DII-HEP project to implement elastic VM-provisioning.

The EES OpenStack plugin controls the start/stop of VMs in the OpenStack cloud and communicates with the OpenStack controller using the JSON API. It is written in C and uses the standard JSON-C library to parse the OpenStack JSON responses. The C Curl library is used to send and receive JSON objects to/from OpenStack. XACML (eXtensible Access Control Markup Language) obligations are handled by the EMI saml2-xacml2 library. For more details on the EES OpenStack plugin see this article [10].

4. Host Identity Protocol
The Host Identity Protocol (HIP) [11] has been designed for mobile networks and standardized in the following Request for Comments (RFC) documents 4423, 5201–5207, 6092. It provides
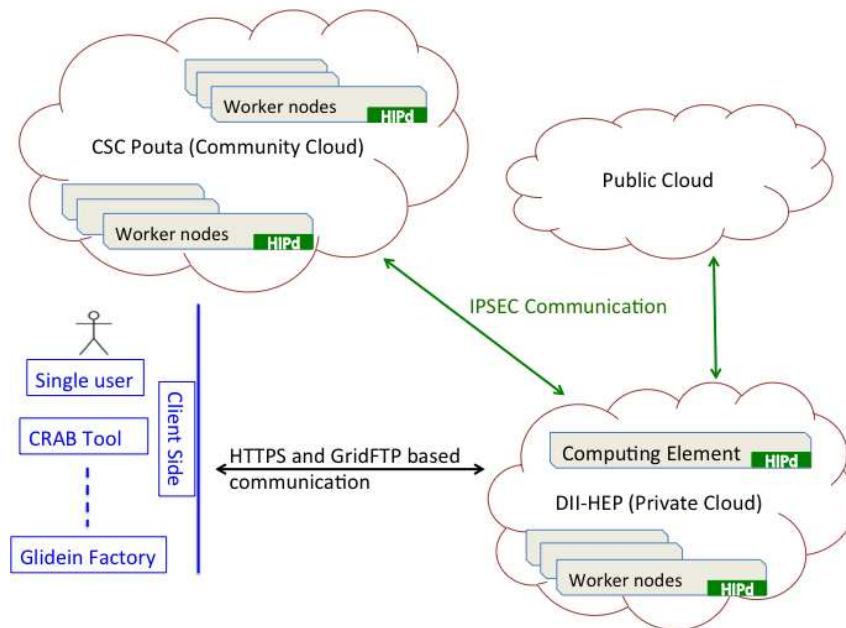
Figure 3. Hybrid cloud environment based on DII-HEP and cPouta clouds.

persistent cryptographic identifiers and supports both IPv4 and IPv6 addressing. The namespace is secure in the sense that HIP identifiers cannot be spoofed. The Host Identifiers (HI) are not routable, so they are translated into routable addresses (locators) between network and transport layer. The mapping from HI to network layer locators is dynamic, which enables end-host mobility and multihoming. *This can be useful for creating a hybrid cloud where the IP-address spaces can be different in different clouds.* The dynamic HI mapping can also be useful for VM live migration and site renumbering. The HIP connections are typically protected with IPSec. *Application software can be used unchanged with the HIP protocol.* HIP can in a way be viewed as a Virtual Private Network (VPN), but without a gateway.

A study of the impact of using the HIP protocol was made by comparing the performance for CMS CPU intensive CRAB (CMS Remote Analysis Builder) jobs in three different cases: no specific security mechanism, HIP protocol and SSL/TLS based security. The jobs run for about 170 minutes and each test was running for a about a week on 200 concurrent job slots. The summary of these tests is that the HIP protocol creates negligble CPU and network overhead on the CE. For more details on the HIP protocol use in DII-HEP see this paper [12].

5. Secure Hybrid Cloud setup
The HIP protocol has been used in DII-HEP to simplify the network configuration in a hybrid cloud setup. An additional benefit of the HIP protocol is that the communication is encrypted which is useful inside a multitenant cloud as well as on WAN connections.

CSC, The Finnish IT Center for Science has a new data center in Kajaani, more than 500 km north of Helsinki. The supercluster Taito (initially 9 216 and currently 19 048 cores) is located in Kajaani. The part of Taito dedicated for cloud computing is called cPouta, which is an Infrastructure as a Service resource. OpenStack Grizzly is in use on cPouta before the upgrade to the IceHouse release. The DII-HEP setup in Kumpula, Helsinki currently runs 40 cores on OpenStack. HTCondor can join together several HTCondor pools by *flocking* them together. On cPouta there are several VMs as worker nodes and one special HTCondor node that is used for the pool flocking connecting the cPouta worker nodes to the Kumpula cloud

resources. Fig. 3 shows how the the Host Identity Protocol is used for *secure cloud bursting* from the Kumpula DII-HEP cloud to the Kajaani Pouta cloud. Tests with 44 cores running on cPouta and connected to the OpenStack in Helsinki are currently in progress.

6. Future work
Work on making the EES OpenStack plugin production quality is continuing as well as work on scaling up the intra-cloud setup based on HIP. The Jade cluster is planned to be replaced by a cloud setup as presented here on cPouta. The presented cloud setup could also be used by other scientific applications, like in the Finnish Grid and Cloud Infrastructure consortium.

7. Summary
A scalable infrastructure for CMS jobs has been created which is **cloud based** and **grid enabled**. The ARC grid interface has been used to provide the interface for CMS applications, so that all CMS jobs can run unchanged. The cloud setup itself does not rely on grid tools. The same setup could be used for other applications as well. The cloud provides flexibility in managing the infrastructure. Up to 200 simultaneous jobs have run so far. An EES OpenStack plugin prototype has been written. Initial results show that the HIP protocol can be very helpful in resource mobility and security. Cloud bursting has been demonstrated using the HIP protocol and HTCondor flocking for 44 cores so far over a distance of more than 500 km.

References
[1] Edelmann E, Happonen K, Koivumäki J, Lindén T and Välimaa J 2011 *J. Phys.: Conf. Ser.* **331** 062037
[2] Ellert M, Grønager M, Konstantinov A, Kónya B, Lindemann J, Livenson I, Nielsen J L, Niinimäki M, Smirnova O and Wäänänen A 2007 *Future Generation Comput. Syst.* **23** 219–240
[3] OpenStack Cloud Software `http://www.openstack.org`
[4] Thain D, Tannenbaum T and Livny M 2005 *Concurrency - Practice and Experience* **17** 323–356
[5] Skipsey S, Doherty T and Turner A G D 2012 *J. Phys.: Conf. Ser.* **396** 042054
[6] Dorigo A, Elmer P, Furano F and Hanushevsky A 2005 *WSEAS Transactions on Computers* **4** 348–353
[7] Toor S, Osmani L, Eerola P, Kraemer O, Lindén T, Tarkoma S and White J 2014 *Journal of Physics: Conference Series* **513** 062047
[8] EMI Execution Environment Service `https://twiki.cern.ch/twiki/bin/view/EMI/ARGUS_EESv0_0_10Details`
[9] EMI Argus `http://www.eu-emi.eu/kebnekaise-products/-/asset_publisher/4BKc/content/argus`
[10] White J, Toor S, Eerola P, Lindén T, Kraemer O, Osmani L and Tarkoma S 2014 *Proceedings of Science* **ISGC2014** 019
[11] Moskowitz R and Nikander P 2006 Host identity protocol (hip) architecture, rfc 4423 (informational) `http://www.ietf.org/rfc/rfc4423.txt`
[12] Osmani L, Toor S, Komu M, Kortelainen M J, Lindén T, White J, Khan R, Eerola P and Tarkoma S 2015 Secure cloud connectivity for scientific applications submitted