MANAGING A TIER 2 COMPUTER CENTRE WITH A PRIVATE CLOUD INFRASTRUCTURE

Stefano Bagnasco, Riccardo Brunetti, Stefano Lusso (INFN-Torino), Dario Berzano (CERN)
The amount of resources and the variety of applications is steadily increasing, manpower unfortunately is not.

- It is becoming almost mandatory to consolidate such resources to achieve scalability and economies-of-scale
  - Separate application management from infrastructure management
  - Our Data Centres need to become providers of computing and storage resources, not (only) of high level services
- The Cloud approach (IaaS) might help to better provision resources to the different scientific computing applications
  - Grid Sites, small or medium computing farms, single users,…
- Several cloud computing projects are starting at national and European level
  - From definition of best practices and reference configurations to deployment of large-scale distributed infrastructures
  - A local working cloud infrastructure will also allow to take immediately part in such activities
Managing a Tier-2 Computer Centre with a Private Cloud Infrastructure

Stefano Bagnasco

**Philosophy**

- Ensure **interoperability**
- Favour **manageability and flexibility** over performance
- Provide a **production service** to applications

- **Keep it simple**
- **Stay mainstream**
- **Be user-oriented**
PHILOSOPHY

- Don’t use too many tools
- Develop as few pieces as possible
- Introduce features only when needed by applications
- Use few simple images plus contextualization

Keep it simple

Stay mainstream

Be user-oriented
Choose stable and widely used tools and components:

- OpenNebula cloud stack
  - Common interfaces: OCCI, EC2, OCA
- GlusterFS filesystem
- OpenWRT for network management

Keep it simple

Stay mainstream

Be user-oriented
- Adopt an **agile development** cycle
- Give resources to users as soon as possible
- Add functionalities as they become needed
Two Clusters

Services

VMs providing **critical services**:
- in- & out-bound connectivity
- public & private IP
- live migration
- no special I/O requirements

Workers

VMs providing **computing workforce**:
- example: Grid WNs
- private IP only
- high storage I/O performance
Managing a Tier-2 Computer Centre with a Private Cloud Infrastructure

**TWO CLUSTERS**

- **Server-class hardware**
- **Shared image repository**
- **Resiliency-optimized FS for shared system disks**
- **Currently 4 hosts**

- **Working-class hardware 😊**
- **Cached image repository**
- **Access to performance-optimized FS for data needs**
- **Currently 35 hosts**
Cloud management Toolkit: OpenNebula

- Open Source stack with a wide user community
- Modular architecture
- Already provides most of the required functionalities
- Uses “standard” interfaces (EC2, OCCI, OCA)
- Easy to customize (mostly shell and ruby scripts)
- OpenStack, now widely adopted in new projects, was too embryonic when we started
- ...and arguably* OpenNebula is better suited at Data Center Nebulization
- Currently using version 3.6, will migrate to 3.8 soon (or 4.0, available since last week)
- We use templates based on few very simple images plus full contextualization via context scripts and puppet (looking into CloudInit)

* See e.g. blog.opennebula.org/?p=4042
THE INGREDIENTS

● **Backend storage: **GlusterFS
  - Easy to setup in a basic configuration
  - Flexible enough to cater to different needs with a single tool (see next slides)
  - Proven robustness and scalability

● **VM network management: **OpenWRT
  - Light-weight Linux distribution for embedded systems
  - Provides tools for network configuration and management
  - We deploy “VRouters” for virtual clusters
  - Again, OpenVSwitch was not integrated in OpenNebula when we started
MULTIPURPOSE STORAGE: GLUSTERFS

GlusterFS mimics RAID functionalities at filesystem level by aggregating “bricks” on different machines:

- distributed
- replicated
- striped (can be combined)

- Horizontal scalability:
  - no master host, all synchronizations are peer-to-peer
  - clients access data directly from the node hosting it

- Easy management:
  - On-line addition, removal, replacement of bricks
Our use cases:

- **VM image repository:**
  - one brick exported

- **System datastore for service-class hosts:**
  - replicated on two servers for redundancy.
  - Replica is synchronous, self-healing enabled.
  - Continuous r/w occurs

- **Experiment data**
  - pool of aggregated disks (currently ~50 TB).
  - Very high throughput towards many concurrent clients
Two storage servers with 10Gbps interface provide some of the LUNs through GlusterFS

- All the virtual machines run on RAW or QCOW file images
- Services System Datastore is shared to allow live migration
- Workers System Datastore is local to the hypervisors to increase I/O capacity. Images repository is locally cached on each hypervisor to reduce startup time.
  - An ad-hoc script synchronizes the local copies using a custom “torrent-like” tool (scpWave + rsync) when new versions of the images are saved
**Network Isolation (Level 2)**
- Each user has its own Virtual Network, isolated using `ebtables` rules defined on the hypervisor bridge (OpenNebula V-net driver takes care of this).

**Virtual Routers (Level 3)**
- Lightweight VM image (1 CPU, 150 MB Ram) with a Linux distribution designed for embedded systems
- DHCP Server, DNS Server, NAT
- Firewalling/Port Forwarding

This provides the user with a *dedicated fully featured class-C network* while the connectivity remains under our control (the user has no access to the V-Router)
<table>
<thead>
<tr>
<th>ID</th>
<th>Owner</th>
<th>Group</th>
<th>Name</th>
<th>Registration time</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>aguarise</td>
<td>INFN-TO</td>
<td>c5-ectics-devel-v1</td>
<td>01/22/2013</td>
</tr>
<tr>
<td>62</td>
<td>aguarise</td>
<td>INFN-TO</td>
<td>c6-devel-eclipse-vram</td>
<td>01/23/2013</td>
</tr>
<tr>
<td>63</td>
<td>cernvm</td>
<td>users</td>
<td>CernVM-Slave</td>
<td>02/07/2013</td>
</tr>
<tr>
<td>64</td>
<td>cernvm</td>
<td>users</td>
<td>CernVM-Master</td>
<td>02/07/2013</td>
</tr>
<tr>
<td>66</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>SLC5-SSO</td>
<td>03/04/2013</td>
</tr>
<tr>
<td></td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>OneMaster-3.8-V3</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>WN-EMI2-CentOS6-V2</td>
<td>04/09/2013</td>
</tr>
<tr>
<td>73</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>WN-EMI2-CentOS6-V2-small-postinstall</td>
<td>04/09/2013</td>
</tr>
<tr>
<td>75</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CE-EMI2-CentOS6-v5-postinstall</td>
<td>04/10/2013</td>
</tr>
<tr>
<td>76</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CE-EMI2-CentOS6-V5-install</td>
<td>04/15/2013</td>
</tr>
<tr>
<td>79</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>WN-EMI2-CentOS6-V2-CVMFS</td>
<td>04/17/2013</td>
</tr>
<tr>
<td>80</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>BDII-EMI2-CentOS6-V5-install</td>
<td>04/22/2013</td>
</tr>
<tr>
<td>81</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>SE-EMI2-CentOS6-V5-install</td>
<td>05/03/2013</td>
</tr>
</tbody>
</table>

**Tier-2 services and worker node templates**

**CERNVM-based templates**
Managing a Tier-2 Computer Centre with a Private Cloud Infrastructure | Stefano Bagnasco

ACAT2013 | Beijing, May 16-21, 2013 - 18/417

TIER 2 CPU EFFICIENCY

ALICE overall average over same period ~85%
VAF components: overview

- User interacts with:
  - PoD to request and book workers
  - PROOF to execute jobs

- Under the hood:
  - worker requests are scheduled by HTCondor
  - CernVM virtual machines are part of the HTCondor cluster

PROOF
PoD
HTCondor
CernVM

Services stack
**THE VIRTUAL ANALYSIS FACILITY**

**VAF components: CernVM**

- CernVM is our **reference platform**:
  - uniform development environment
  - lightweightness: software downloaded on demand with cvmfs
  - online public context repository (sort of "marketplace")

- CernVM Cloud ecosystem: **EXPERIMENTAL**
  - Entire VAF cluster instantiated with one click using CernVM Gateway

---

PROOF

PoD

HTCondor

CernVM

Services stack

Dario.Berzano@cern.ch

http://goo.gl/CFnMM
THE VIRTUAL ANALYSIS FACILITY

CernVM ecosystem: elasticity

- CernVM Agent and Gateway are experimental
- CernVM components enable automatic "elasticity"

Dario.Berzano@cern.ch
http://goo.gl/CFnMM
FUTURE DEVELOPMENTS

- Understand the opportunities given by the CernVM “ecosystem”

- Study the integration of the OpenNebula Authn/Authz system in a VO context or using federated authentication mechanisms.

- Explore the GlusterFS UFO Object Storage to provide a “DropBox-like” storage to users.

- Participate in upcoming projects aimed to develop a higher-level federated cloud infrastructure
● The infrastructure is in full production mode since more than one year

● The core software stack (OpenNebula + GlusterFS) proved itself stable and robust

● The management of the centre was actually simplified
  ■ Trivial example: rolling updates

● Lots of room for improvement and optimization
  ■ Example: there is no trivial method to optimize allocation of sets of identical machines on heterogeneous hypervisors (8, 12, 24 cores per host)

● Lots of room also for new features, extensions and integrations
Questions?

Stefano.Bagnasco@to.infn.it