HLT Cloud:
current status and timeline

Dario Berzano

ALICE Offline - CERN

ALICE Offline Week - 20.03.2015

Preface

Rationale

- New High Level Trigger: very powerful farm for realtime processing
  - Up to 7000 job slots (with hyperthreading): ~Tier-1
- Close to the detector: dedicated, isolated, only controlled software
  - Grid is much less secure: compiled code from users
- HLT might be unused sometimes (partly or entirely)
  - Major shutdowns, technical stops
- Exploit HLT for offline tasks when unused: opportunistic Grid site
- Two extremely different security models
  - Virtualization (and more) to provide isolated sandboxes

The new High Level Trigger

- 180 nodes with 2 Intel Xeon processors, 10 cores each
- Hyperthreading is on: 40 single-core jobs per node
- 128 GB RAM: 3.2 GB per job
- GPU: AMD Firepro W8000 graphics card
- Hard disks are SSD: each node has one Grid-dedicated disk
- Network: 1 Gbit/s Ethernet and InfiniBand
- Uplink to the CERN General Purpose Network: 80 Gbit/s

Components

- Virtualization infrastructure: OpenStack Icehouse
- Network setup: VLANs
- Batch jobs: HTCondor + elastiq + AliEn

Administrative domains

- High Level trigger experts have total control of HLT nodes
  - They select which nodes we can use for job submission
- The Offline manages virtualization, batch queue, submission policies
  - No user jobs there: only centrally managed

Virtualization infrastructure

OpenStack

- Popular and well supported cloud orchestrator
  - We use Icehouse (last-but-one version)
- We need a very basic setup
  - One VM network, one user, one head node
  - Ability to start and shutdown virtual machines
    - Selectively decide which physical nodes can run VMs
- Very modular: scales well, but basic setups complicated
  - Official doc is difficult to follow
  - Our colleagues from Bari made this very nice guide: https://github.com/infn-bari-school/OpenStack-Icehouse-Installation/wiki

OpenStack and cloud glossary

- Hypervisor: the physical node running virtual machines
- Nova Compute: OpenStack service running virtual machines
  - Compute nodes are OpenStack hypervisors
- Image: a “disk dump” containing a base Operating System installation
  - An image can be instantiated many times
- Flavor: a set of resources (CPU, RAM, disk) assigned to a VM
- Instance: the VM, essentially image + flavor
- Glance: OpenStack image manager
  - Stores, caches, deploys images

Neutron and Nova Network

- OpenStack Icehouse has two network modules:
  - Neutron: network is virtualized, like a software switch, does L2/L3
  - Nova Network (or "Legacy"): rely on "physical" networks and VLANs
- Legacy is deprecated but we use it anyway
- Much simpler, no performance problems, much more stable

Deprecation of Nova Network

This leaves you with an important point of decision when designing your cloud. OpenStack Networking is robust enough to use with a small number of limitations (performance issues in some scenarios, only basic high availability of layer 3 systems) and provides many more features than `nova-network`. However, if you do not have the more complex use cases that can benefit from fuller software-defined networking capabilities, or are uncomfortable with the new concepts introduced, `nova-network` may continue to be a viable option for the next 12 months.

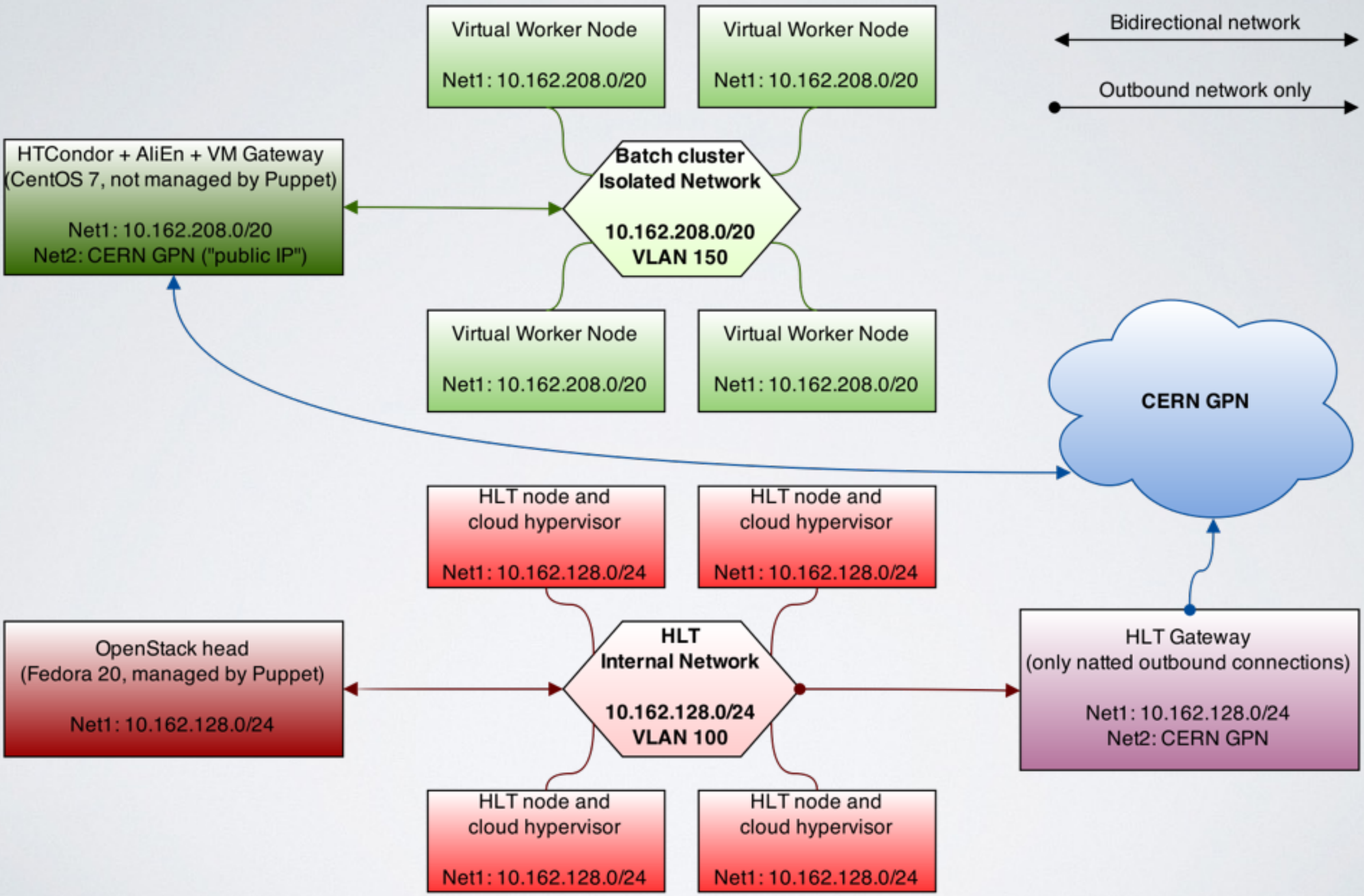
http://docs.openstack.org/openstack-ops/content/nova-network-deprecation.html

Setup and isolation: network, disks, head node

Network setup evolution

- First setup on the HLT devel cluster:
  - Subset of the HLT net: netmask and L2 isolation via ebtables
  - One server for OpenStack head and AliEn VOBox
- Current setup on devel cluster (will be ported to production):
  - Tagged VLAN for VMs
  - A physical AliEn VOBox: only sees VMs and world
  - A physical OpenStack head: only sees hypervisors
  - More reliable hardware isolation, as agreed with the HLT experts
  - Possible to do traffic shaping based on VLANs

Network setup schema



Disks and LVM

- Grid jobs might wear out HLT disks more quickly because of swap
  - New HLT nodes have a dedicated SSD for our VMs
- Running virtual machines use LVM partitions on the dedicated disk
  - Better performances than running them from files

Nodes configuration deployment

- The HLT farm uses Foreman and Git+Puppet to sync configuration
  - Push to Git, and Puppet will sync: all is versioned
  - Push config on Git branches assigned to groups of test hosts
- Configuration and OSes on Offline-managed nodes:
  - OpenStack head and Compute: Fedora 20, managed by Puppet
  - AliEn VOBox, HTCondor head: CentOS 7, manually configured
  - HTCondor VMs: CernVM with boot-time contextualization

Grid-on-HLT head node roles

- Grid-on-HLT node sees virtual machines and the external world
- Standard services: HTCondor and AliEn *(see next)*
- Additional services:
  - CVMFS proxy (Squid): used by virtual machines as a cache
  - NAT and gateway: for outbound connectivity

AliEn and the dynamic Grid-on-HLT cluster

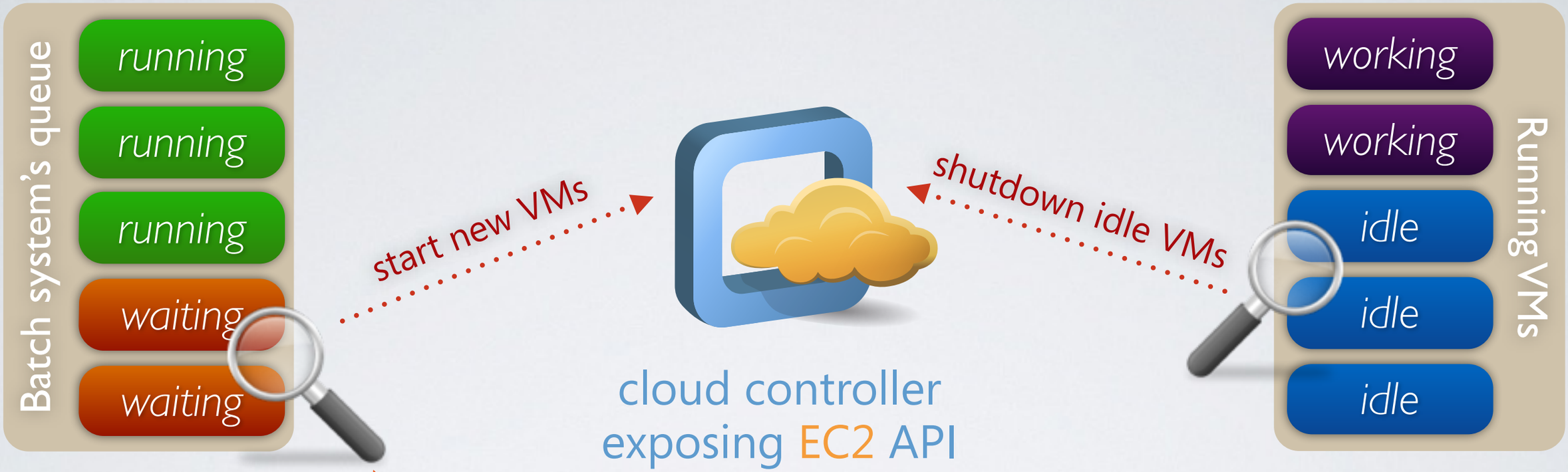
Use case

- Grid-on-HLT farm is an opportunistic site
  - HLT admins can get back resources by killing our VMs
  - We only get what they give us, when they give us
  - We try to relinquish resources first to prevent abrupt killing
- Components have to deal with:
  - Disappearing/reappearing hypervisors
  - Killed VMs/new VMs started
  - Zombie jobs

HTCondor + AliEn + elastiq

- HTCondor: the highly configurable batch system from overseas
  - Nodes self-registration, deals correctly with nodes that disappear
- AliEn, our Grid middleware: supports HTCondor
  - Transparently submit on Grid-on-HLT with a standard interface
  - Easily configurable submission policies: (*i.e.* which jobs to run there)
  - Pure AliEn site (*i.e.* no WLCG): much simpler to configure
- elastiq: mediator between HTCondor and OpenStack
  - Starts new Grid-on-HLT virtual machines when we have new jobs
  - Shut down idle virtual machines

elastiq: orchestration



Jobs waiting too long will trigger a **scale up**

Supports **minimum and maximum quota** of VMs

Mix physical/virtual nodes: elastiq will only manage the nodes it started

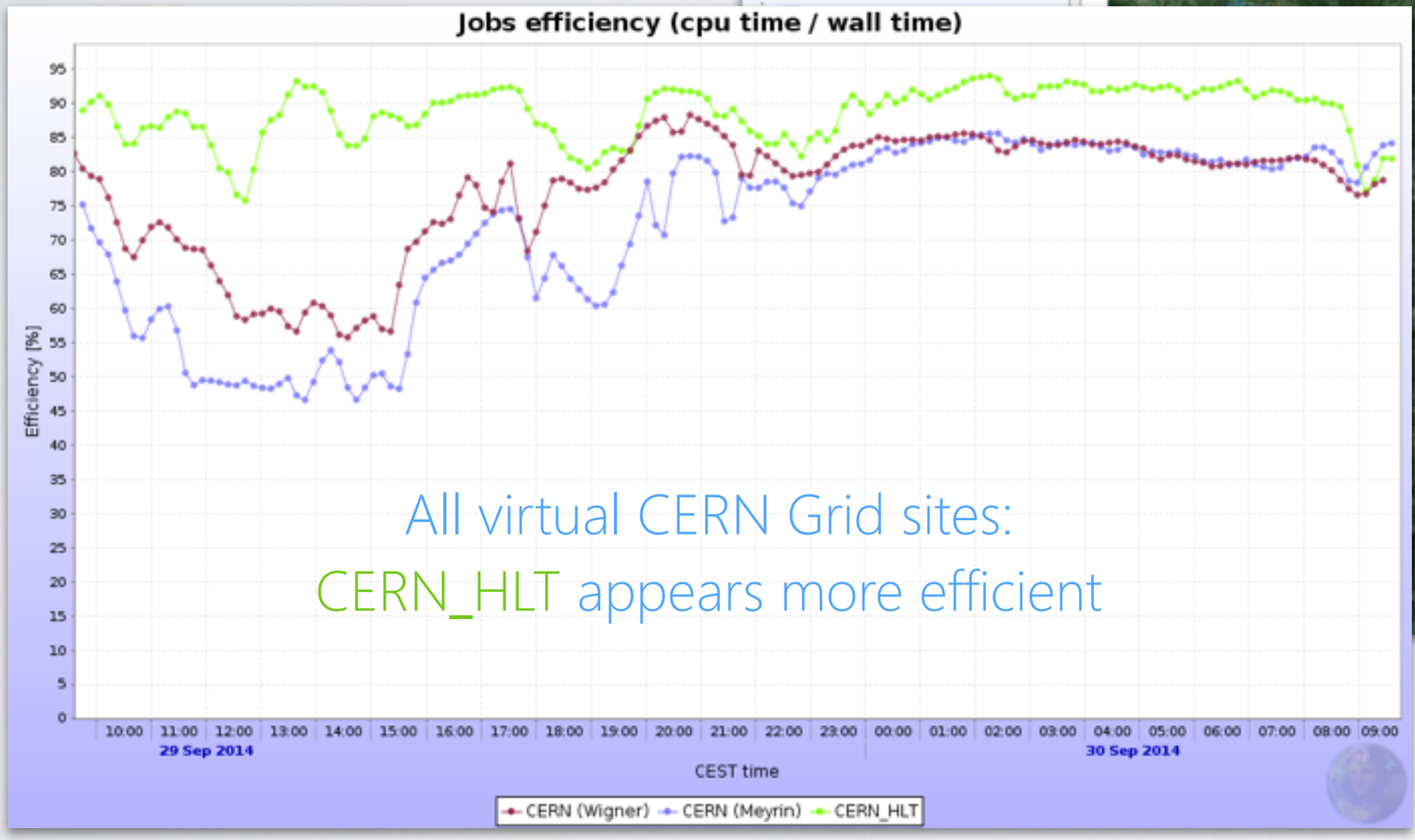
Robust: detects VM deployment and boot **errors** and **reinstantiates** them

https://github.com/dberzano/elastiq

(debs and RPMs available)

Conclusions

ALICE::CERN::CONDOR



All virtual CERN Grid sites:
CERN_HLT appears more efficient

http://alimonitor.cern.ch/

Timeline/1

- October 2014
  - First setup on the HLT devel cluster with Fedora 19
  - VMs in HLT network: insecure, but VLANs not enabled on switches
- February 2015
  - New setup on the HLT devel cluster with Fedora 20
  - Switches configured for supporting tagged VLANs

Timeline/2

- March 16-20 2015 (today!)
  - Some VLAN glitches fixed: VLANs fully operational
  - HTCondor + elastiq + VOBOX + NAT configured and tested
    - VOBOX is ready to be physically moved devel → prod
  - OpenStack Head OK: ready to be moved devel → prod as well
  - Ordering a 10 GbE interface for the head node (NAT)
  - Ordering the first batch of 180 disks needed (~50)

Timeline/3

- April/May 2015
  - 10 GbE expected: mounting and testing
  - First ~50 disks expected: mounting them
  - Finally physically moving VOBOX + OpenStack Head to prod
    - The first ~50 HLT nodes can be added to the cloud and used
- Fall 2015
  - Expecting all the remaining disks and adding all the nodes
  - Full operation and maintenance mode