FermiCloud update - enabling scientific workflows in the cloud

Gerard Bernabeu (gerard1@fnal.gov)
Grid & Cloud Services Department
Fermilab

Work supported by the U.S. Department of Energy under contract No. DE-AC02-07CH11359
And by joint CRADA FRA 2013-0001/ KISTI C13013 between KISTI and Fermilab

Outline

- The FermiCloud Project
- Current infrastructure
- Use cases
- OpenNebula
- CRADA with KISTI
- VMs as jobs
- Idle VM detection
- What's next?

FermiCloud – Initial Project Specifications

FermiCloud Project was established in 2009 with the goal of developing and establishing Scientific Cloud capabilities for the Fermilab Scientific Program,

- Building on the very successful FermiGrid program that supports the full Fermilab user community and makes significant contributions as members of the Open Science Grid Consortium.
- Reuse High Availabilty, AuthZ/AuthN, Virtualization from Grid

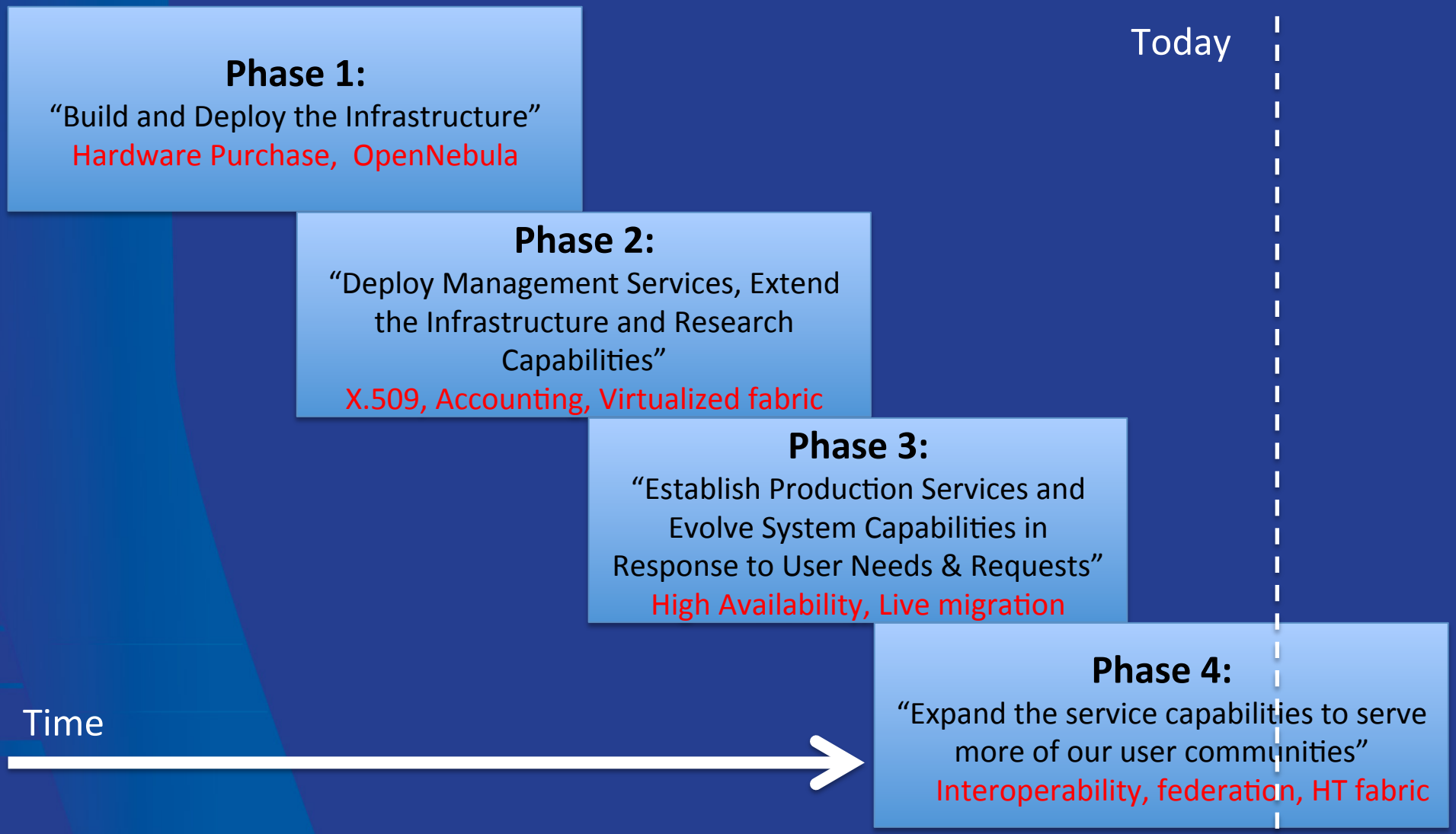
The FermiCloud project is a program of work that is split over several overlapping phases.

- Each phase builds on the capabilities delivered as part of the previous phases.

In a broad brush, the mission of the FermiCloud project is:

- To deploy a production quality Infrastructure as a Service (IaaS) Cloud Computing capability in support of the Fermilab Scientific Program.
- To develop additional IaaS, PaaS and SaaS Cloud Computing capabilities based on the FermiCloud infrastructure in support of on-demand services for our scientific stakeholders

Overlapping Phases



2013 FermiCloud Resources

The current FermiCloud hardware resources include:

- 23 servers with 16 HT Intel E5640 cores & 48GB of ram each (prod)
  - HW support extended for FY14
- Public network access via the high performance Fermilab network,
  - This is a distributed, redundant network.
- Private 1 Gb/sec network,
  - This network is bridged across FCC and GCC on private fiber,
- High performance Infiniband network,
  - Currently split into two segments (10+13 servers),
  - InfiniBand available in VMs via SRIOV
- Access to a high performance FibreChannel based SAN,
  - This SAN spans both buildings. Distributed Shared FileSystem on top of it.
- Access to the high performance BlueArc based filesystems,
  - The BlueArc is located on FCC-2,
- Access to the Fermilab dCache and enStore services,
  - These services are split across FCC and GCC,
- Access to 100 Gbit Ethernet test bed in LCC (Integration nodes),
  - Intel 10 Gbit Ethernet converged network adapter X540-T1.
- Documentation at http://fclweb.fnal.gov/

Typical Use Cases

Public net virtual machine:

- On Fermilab Network open to Internet
- Can access dCache and Bluearc Mass Storage
- Common home directory between multiple VM's

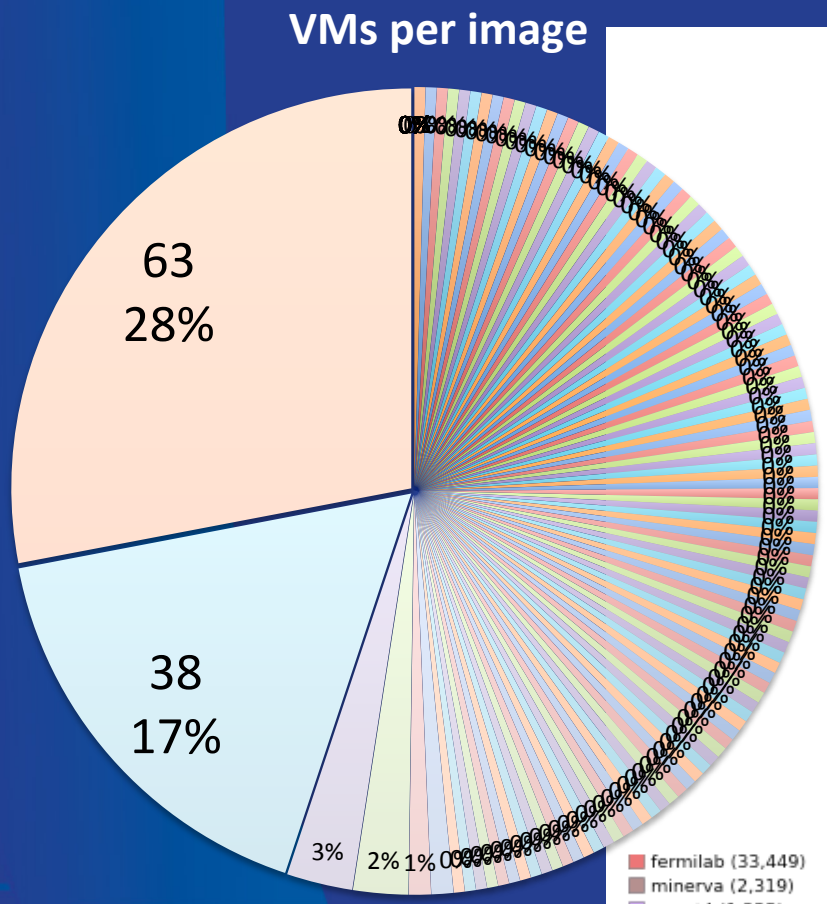
Public/Private Cluster:

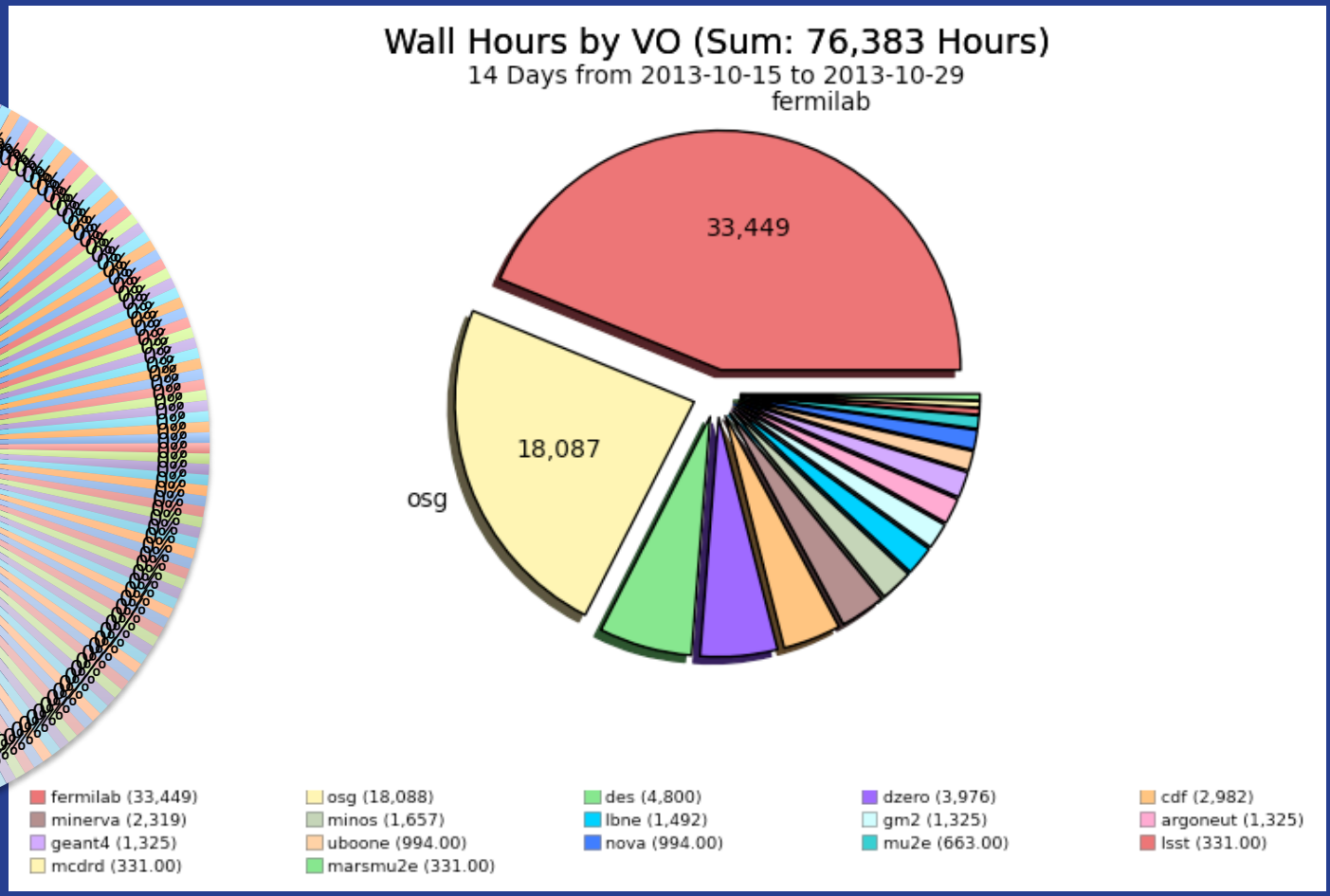
- One gateway VM on public/private net
- Cluster of many VM's on private net
- Data acquisition simulation

Storage VM:

- VM with large non-persistent storage
- Use for large MySQL or Postgres databases, Lustre/Hadoop/ Bestman/xRootd/dCache/OrangeFS/IRODS servers.

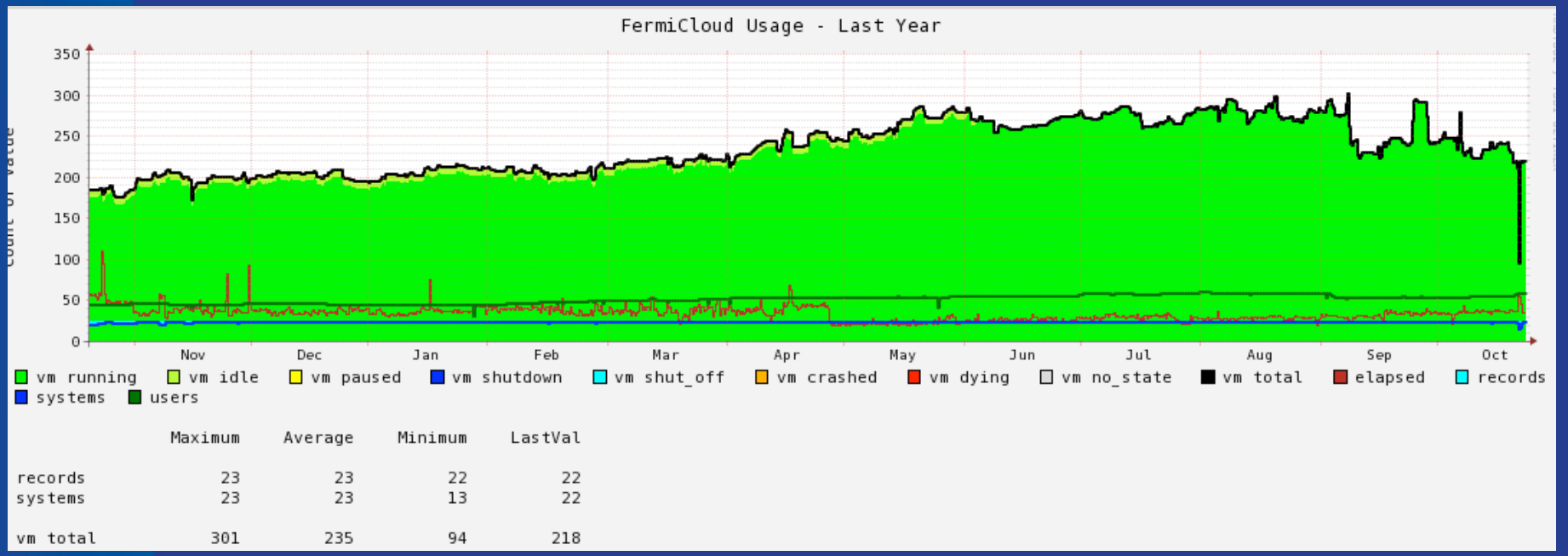
Typical Use Cases (Oct 2013)





115 active VM images (out of 150)
~45% are ephemeral SL5/6 instances

FermiCloud usage



	Maximum	Average	Minimum	LastVal
records	23	23	22	22
systems	23	23	13	22
vm total	301	235	94	218

- September peak corresponds to grid bursting NOvA jobs on FermiCloud dynamically provisioned WorkerNodes.
- ~60 fixed IP persistent VMs
- Current scalability limits are
  - Public IPs (~400 usable IPv4 for VMs).
    - This is ~7 VMs per active user (58/177 users)
  - Memory (RAM)

OpenNebula

- OpenNebula was picked as result of evaluation of Open source cloud management software.
- OpenNebula 2.0 pilot system in GCC available to users since November 2010.
  - Began with 5 nodes, gradually expanded to 13 nodes.
  - 4500 Virtual Machines run on pilot system in 3+ years.
- OpenNebula 3.2 production-quality system installed in FCC in June 2012 in advance of GCC total power outage—now comprises 23 servers.

OpenNebula 4.x

- Transition of VMs from ONe 2.0 pilot system to production system (ONe3.2) complete.
- In the meantime OpenNebula has done five more releases.
  - We have updated the evaluation of Open source cloud management software
  - currently working on the upgrade to ONe 4.x

FermiCloud – Fault Tolerance

As we have learned from FermiGrid, having a distributed fault tolerant infrastructure is highly desirable for production operations.

We are actively working on deploying the FermiCloud hardware resources in a fault tolerant infrastructure:

- The physical systems are split across two buildings,
- Puppet (&cobbler) automated server provisioning to ensure homogeneity
- There is a fault tolerant network infrastructure in place that interconnects the two buildings,
- We have deployed SAN hardware in both buildings,
- We have a dual head-node configuration with clustered (rgmanager) failover
- We have a GFS2 + CLVM for our multi-user filesystem and distributed SAN.
- SAN replicated between buildings using CLVM mirroring.
  - This is a complex solution with moderate scalability.

GOAL:

- If a building is "lost", then automatically relaunch "24x7" VMs on surviving infrastructure, then relaunch "9x5" VMs if there is sufficient remaining capacity,
- Perform notification (via Service-Now) when exceptions are detected.

Cooperative R+D Agreement with KISTI

Partners:

- Grid and Cloud Services Dept. @FNAL
- Global Science Experimental Data hub Center @KISTI (Korea Institute of Science and Technology Information)

Project Title:

- Integration and Commissioning of a Prototype Federated Cloud for Scientific Workflows

Status:

- Three major work items:
  1. Virtual Infrastructure Automation and Provisioning,
  2. Interoperability and Federation of Cloud Resources,
  3. High-Throughput Fabric Virtualization.

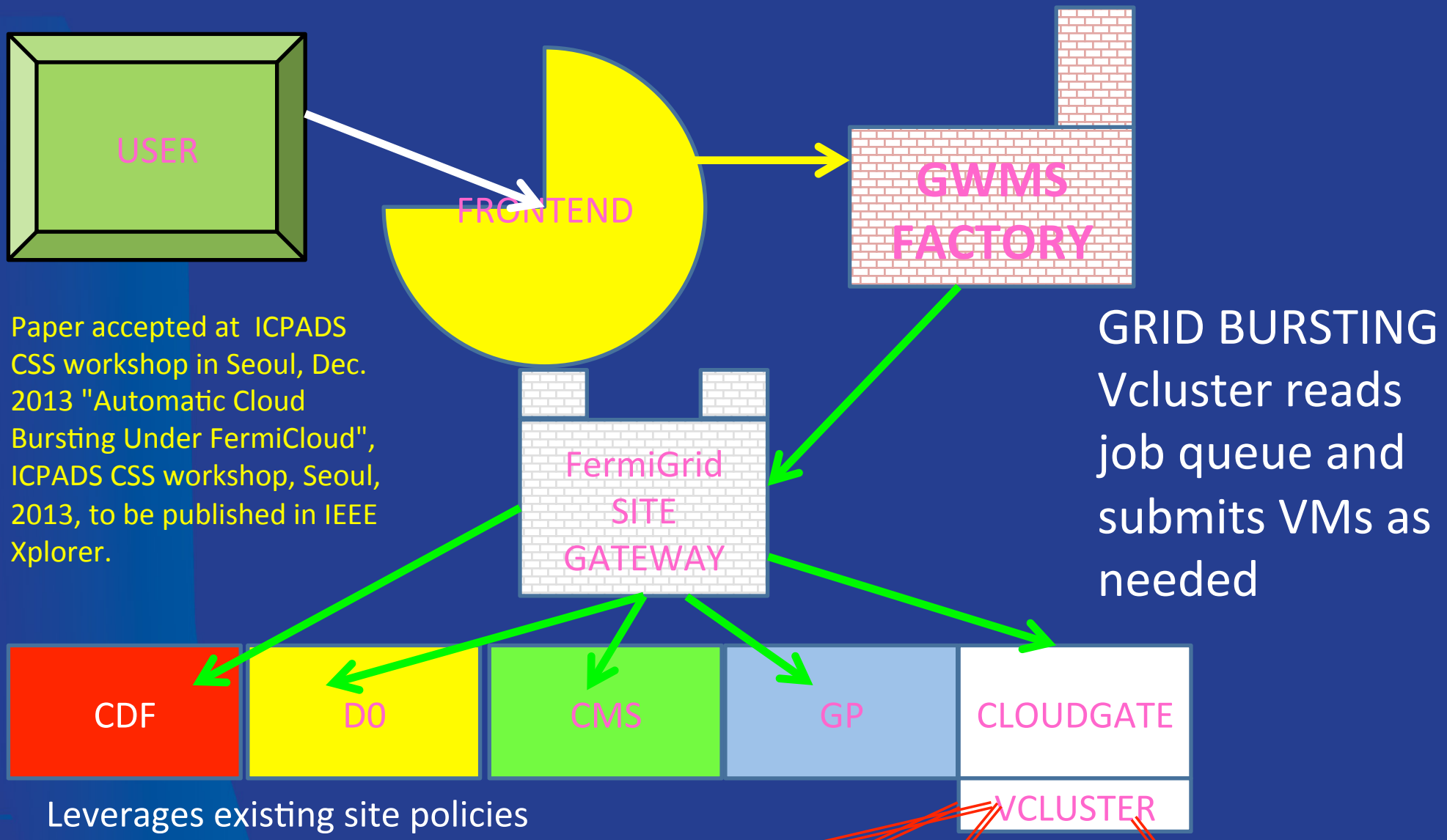
Virtual Machines as Jobs

OpenNebula (like most open-source IaaS stacks) provide an emulation of Amazon EC2.

HTCondor developers added code to their “Amazon EC2” universe to support the X.509-authenticated protocol.

Goal to submit NOvA workflow to OpenNebula @ FermiCloud, OpenStack @ Notre Dame, and Amazon EC2.

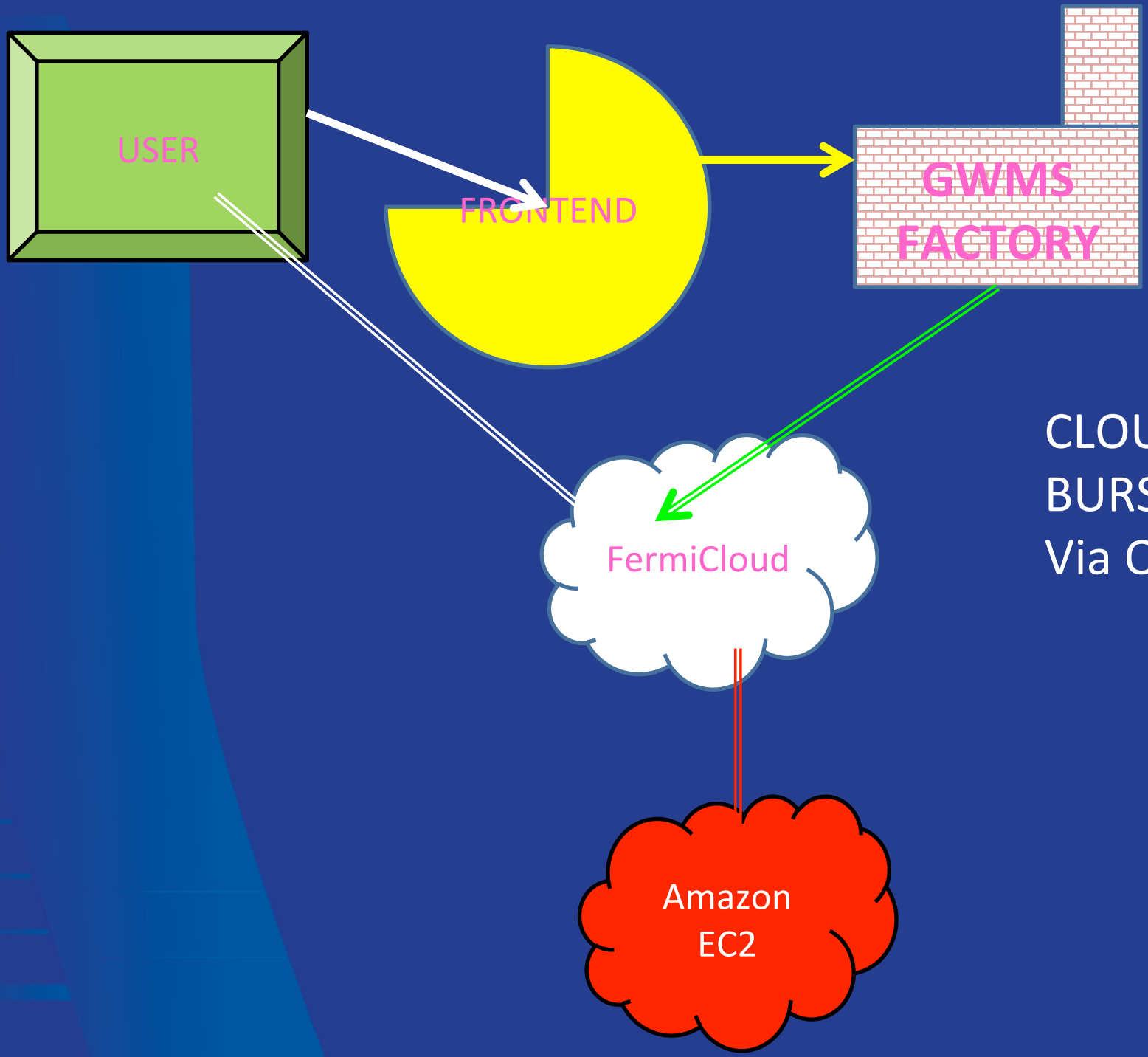
Smooth submission of many thousands of VM’s is key step to making the full infrastructure of a site into a science cloud.



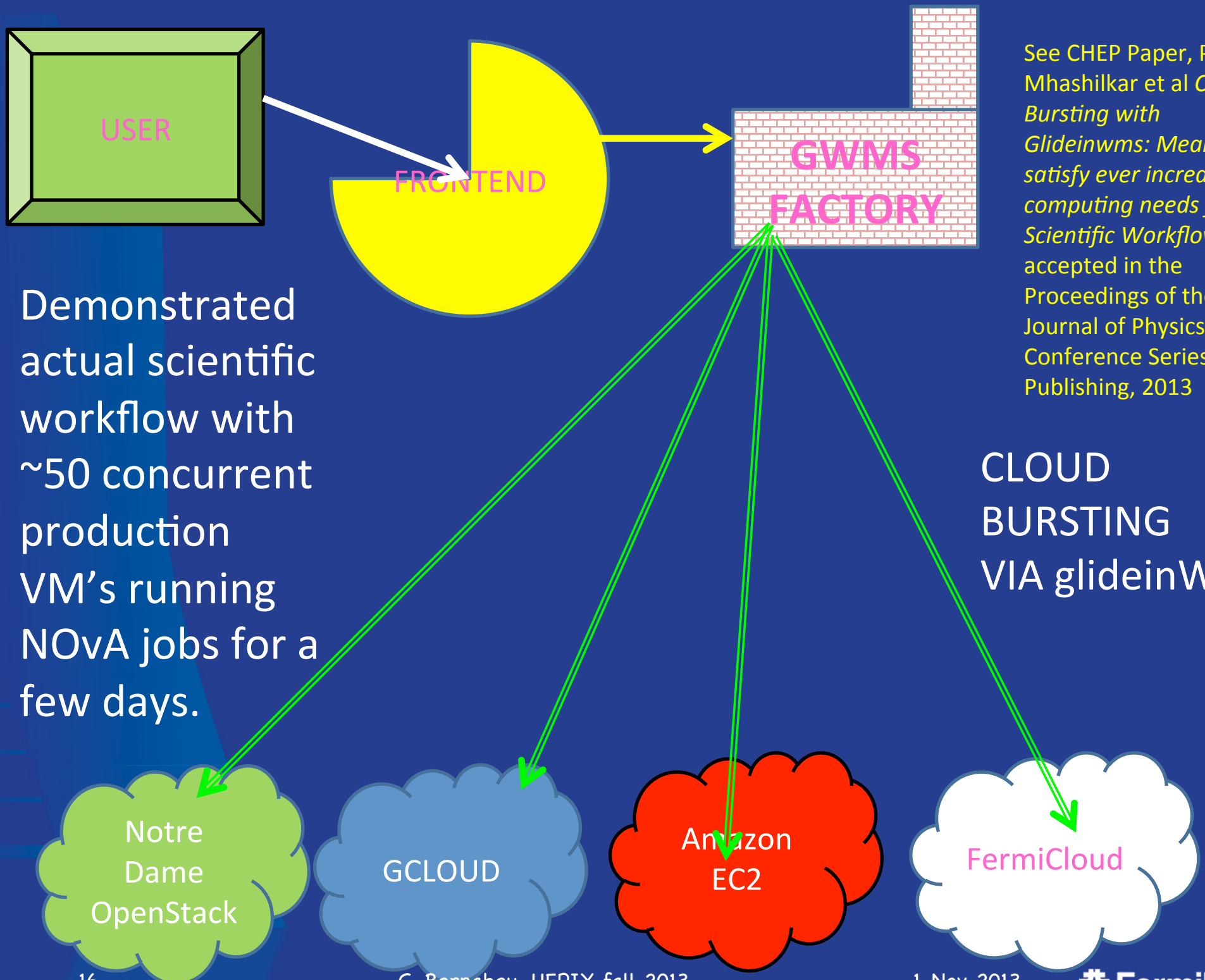
Paper accepted at ICPADS CSS workshop in Seoul, Dec. 2013 "Automatic Cloud Bursting Under FermiCloud", ICPADS CSS workshop, Seoul, 2013, to be published in IEEE Xplorer.

GRID BURSTING
Vcluster reads job queue and submits VMs as needed

Leverages existing site policies



CLOUD
BURSTING
Via OpenNebula



Demonstrated actual scientific workflow with ~50 concurrent production VM's running NOvA jobs for a few days.

See CHEP Paper, P. Mhashilkar et al *Cloud Bursting with Glideinwms: Means to satisfy ever increasing computing needs for Scientific Workflows*, accepted in the Proceedings of the Journal of Physics: Conference Series by IOP Publishing, 2013

CLOUD BURSTING VIA glideinWMS

True Idle VM Detection

In times of resource need, we want the ability to suspend or "shelve" idle VMs in order to free up resources for higher priority usage..

Shelving of "9x5" and "opportunistic" VMs allows us to use FermiCloud resources for Grid worker node VMs during nights and weekends

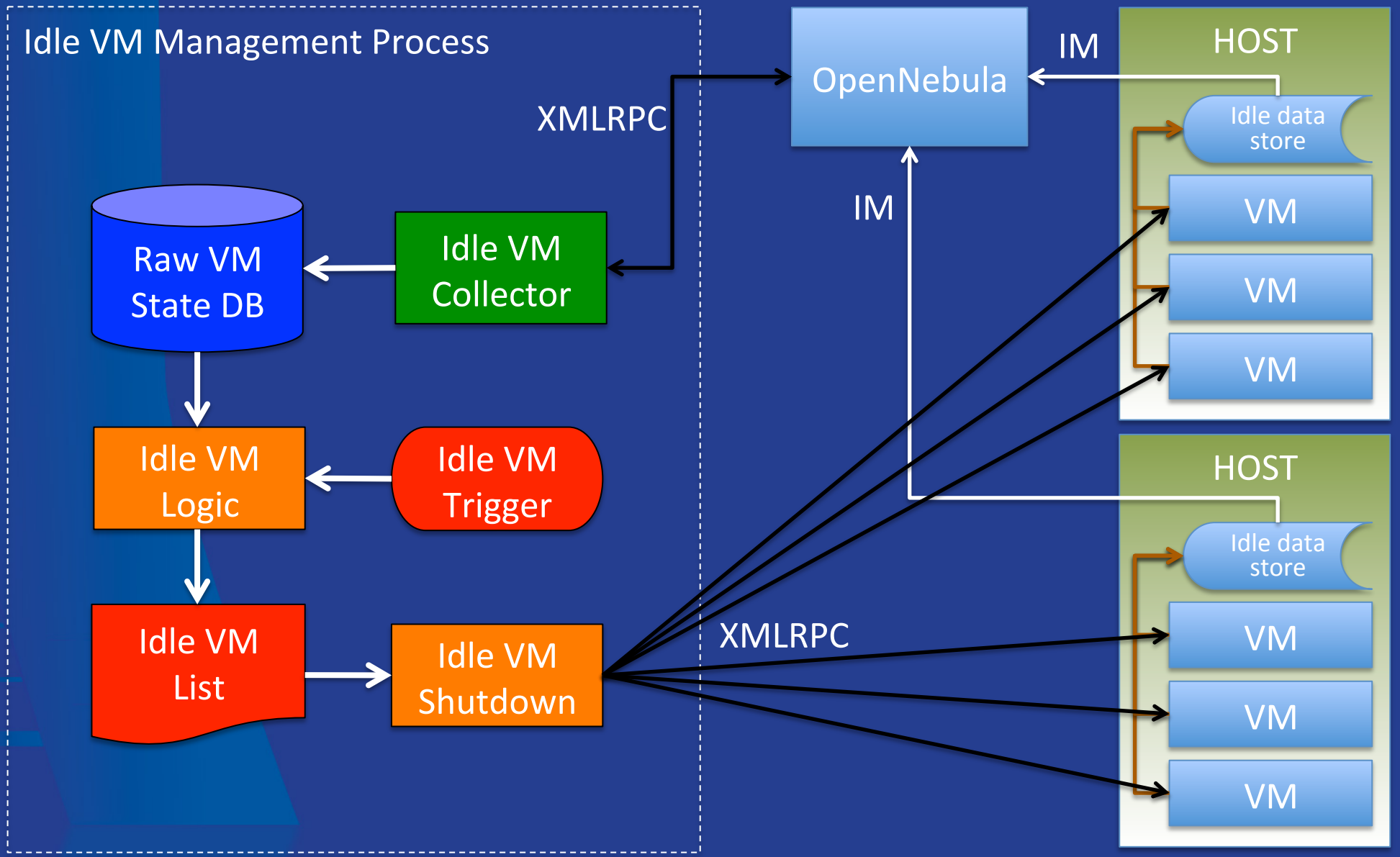
Components:

- Probe software that runs inside Virtual Machine to measure that can be used to detect idle virtual machines based on CPU, disk I/O and network I/O.
- Extensions to OpenNebula Virtual Machine Manager to make port for information
- Extensions to OpenNebula Information Manager to bring the info into OpenNebula information system
- Web GUI to show state of all virtual machines Idle/Busy and enter rules.
- Action engine to shutdown/resume virtual machines using rule-based system.

Software now running in our pre-production cloud

Making plans to deploy in production cloud next month.

Idle VM Information Flow



What's next? – short term

- 6 new servers with 20 HT Intel E5-v2 cores and 80 GB RAM each.
- Private IPv4 for grid-bursted WorkerNodes (IPv6?)
- Upgrade to OpenNebula 4.x
- Building on top of IaaS to deliver on-demand PaaS and SaaS in FermiCloud
  - Leveraging puppet
  - data transfer servers, private clusters

What's next? – mid term

- Review available distributed shared storage solutions
  - Highly-Available VMs (currently GFS2)
  - Image repository (currently GFS2)
  - VMs with large persistent storage
- FermiCloud Phase 5
  - More on API interoperability
  - Automate image interoperability

FermiCloud Project Summary - 1

Science is directly and indirectly benefiting from FermiCloud:

- CDF, D0, Intensity Frontier, Cosmic Frontier, CMS, ATLAS, Open Science Grid,…

FermiCloud operates at the forefront of delivering cloud computing capabilities to support scientific research:

- By starting small, developing a list of requirements, building on existing Grid knowledge and infrastructure to address those requirements, FermiCloud has managed to deliver a production class Infrastructure as a Service cloud computing capability that supports science at Fermilab.
- FermiCloud has provided FermiGrid with an infrastructure that has allowed us to test Grid middleware at production scale prior to deployment.
- The Open Science Grid software team used FermiCloud resources to support their RPM “refactoring” and is currently using it to support their ongoing middleware development/integration.

Acknowledgements

None of this work could have been accomplished without:

- The excellent support from other departments of the Fermilab Computing Sector – including Computing Facilities, Site Networking, and Logistics.
- The excellent collaboration with the open source communities – especially Scientific Linux and OpenNebula,
- As well as the excellent collaboration and contributions from KISTI.
- And talented summer students from Illinois Institute of Technology and INFN

QUESTIONS?

fermicloud-help@fnal.gov

Interoperability and Federation

Driver:

- Global scientific collaborations such as LHC experiments will have to interoperate across facilities with heterogeneous cloud infrastructure.

European efforts:

- EGI Cloud Federation Task Force (HelixNebula) – several institutional and commercial clouds (OpenNebula, OpenStack, StratusLab).

Our goals:

- Show proof of principle—Federation including FermiCloud + KISTI "G Cloud" + commercial cloud providers (Amazon EC2) + other research institution community clouds if possible. DONE!
- Participate in existing federations if possible.

Core Competency:

- FermiCloud project can contribute to these cloud federations given our expertise in X.509 Authentication and Authorization, and our long experience in grid federation

Virtual Image Formats

Different clouds have different virtual machine image formats:

- File system ++, Partition table, LVM volumes, Kernel?

We have identified the differences and written a comprehensive step by step user manual, soon to be public.

Interoperability/Compatibility of API's

Amazon EC2 API is not open source, it is a moving target that changes frequently.

Open-source emulations have various feature levels and accuracy of implementation:

- Compare and contrast OpenNebula, OpenStack, and commercial clouds,
- Identify lowest common denominator(s) that work on all.

VM Image Distribution

Open Source Amazon S3 emulations:

- Installed OpenStack “Swift” storage module
- Functional but didn’t stress test it yet.

Secure repository for incoming virtual machines

- Machine boots in private network
- Run security scans similar to Fermilab site scans to detect unauthorized services and vulnerabilities
- Work in progress, will eventually include gridftp “door” as well.

High-Throughput Fabric Virtualization

Followed up earlier virtualized MPI work:

- Results similar to before, 96% of “Bare Metal” performance with SR-IOV drivers and CPU pinning
- Now can use stock Red Hat kernel and stock OFED drivers
- Automated provisioning of MPI-based virtual machines in OpenNebula
- Use it in real scientific workflows
  - DAQ system simulation
  - Large multicast activity
  - Small scale MPI programs

Paper in progress to submit to CCGRID.

Experiments done with virtualized 10GBe on 100GBit WAN testbed.