FermiCloud

Infrastructure as a Service (IaaS) Cloud Computing In Support of the Fermilab Scientific Program

OSG All Hands Meeting 2012

Steven Timm timm@fnal.gov

Fermilab Grid & Cloud Computing Dept.

For FermiCloud team: K. Chadwick, D. Yocum, F. Lowe, G. Garzoglio, T. Levshina, P. Mhashilkar, H. Kim

Work supported by the U.S. Department of Energy under contract No. DE-AC02-07CH11359

FermiCloud Introduction

- As part of the FY2010 activities, the (then) Grid Facilities Department established a project to implement an initial "FermiCloud" capability.
- **GOAL: Deliver production-capable Infrastucture-as-a-service to support Fermilab Scientific Program**
- **Reuse what we learned from Grid**
- **High Availability, Authentication/Authorization, Virtualization**

- FermiCloud Phase I—Completed Nov. 2010:
  - Specify, acquire and deploy the FermiCloud hardware,
  - Establish initial FermiCloud requirements and select the "best" open source cloud computing framework that best met these requirements (OpenNebula).
  - Deploy capabilities to meet the needs of the stakeholders (JDEM analysis development, Grid Developers and Integration test stands, Storage/dCache Developers, LQCD testbed).
  - Replaced six old racks of integration/test nodes with one rack.

FermiCloud – Current Activities

- FermiCloud Phase II:
  - Implement x509 based authentication (patches contributed back to OpenNebula project and are generally available in OpenNebula V3.2), perform secure contexualization of virtual machines at launch.
  - Implement monitoring and accounting,
  - Target "small" low-cpu-load servers such as Grid gatekeepers, forwarding nodes, small databases, monitoring, etc.
  - Begin the hardware deployment of a distributed SAN,
- FermiCloud Phase III:
  - Select and deploy a true multi-user filesystem on top of a distributed & replicated SAN,
  - Deploy 24x7 production services
  - Live migration becomes important for this phase.

FermiCloud – Hardware Specifications

Currently 23 systems split across FCC-3 and GCC-B:

- 2 x 2.67 GHz Intel “Westmere” 4 core CPU
  - Total 8 physical cores, potentially 16 cores with Hyper Threading (HT),
- 24 GBytes of memory (we are considering an upgrade to 48),
- 2 x 1GBit Ethernet interface (1 public, 1 private),
- 8 port Raid Controller,
- 2 x 300 GBytes of high speed local disk (15K RPM SAS),
- 6 x 2 TBytes = 12 TB raw of RAID SATA disk = ~10 TB formatted,
- InfiniBand SysConnect II DDR HBA,
- Brocade FibreChannel HBA (added in Fall 2011),
- 2U SuperMicro chassis with redundant power supplies

FermiCloud—Software Stack

- Current production
  - Scientific Linux 5.7 host, SLF5 and SLF6 guest
  - KVM hypervisor (Xen available on request).
  - OpenNebula 2.0 with command-line launch
  - Virtual machines distributed via SCP
- Coming soon
  - Scientific Linux 6.1, SLF5 and SLF6 guests
  - KVM hypervisor
  - OpenNebula 3.2 with X.509 authentication
    - Command line, SunStone Web UI, EC2 emulation, OCCI interface, Condor-G
  - Persistent virtual machines stored on SAN (GFS).
- **All Open Source**

FermiCloud Typical VM Specifications

- Unit:
  - 1 Virtual CPU [2.67 GHz “core” with Hyper Threading (HT)],
  - 2 GBytes of memory,
  - 10-20 GBytes of of SAN based “VM Image” storage,
  - Additional ~20-50 GBytes of “transient” local storage.

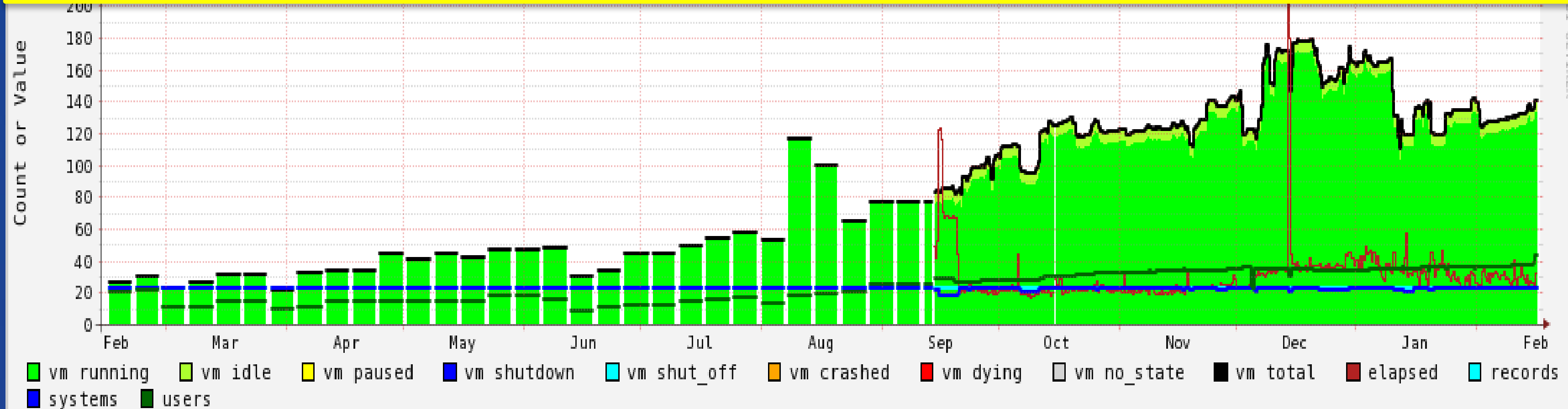
- Additional CPU “cores”, memory and storage are available for “purchase”:
  - Based on the (Draft) FermiCloud Economic Model,
  - Raw VM costs are competitive with Amazon EC2,
  - FermiCloud VMs can be custom configured per “client”,
  - Access to Fermilab science datasets is much better than Amazon EC2.

FermiCloud – Monitoring

- Temporary FermiCloud Usage Monitor:
  - http://www-fermicloud.fnal.gov/fermicloud-usage-data.html
  - Data collection dynamically "ping-pongs" across systems deployed in FCC and GCC to offer redundancy,
  - See plot on next page.

- FermiCloud Redundant Ganglia Servers:
  - http://fcl301k1.fnal.gov/ganglia/
  - http://fcl002k1.fnal.gov/ganglia/
- ***Preliminary*** RSV based monitoring pilot:
  - http://fermicloudrsv.fnal.gov/rsv

Note – ***FermiGrid*** Production Services are operated at 100% to 200% “oversubscription”



VM states as reported by “virsh list”

	Maximum	Average	Minimum	LastVal
records	23	23	23	23
systems	23	23	18	23
vm total	179	86	22	140
vm running	172	83	22	133
vm idle	7	7	6	7
vm paused	1	0	0	0
vm shutdown	0	0	0	0
vm shut off	0	0	0	0
vm crashed	0	0	0	0
vm dying	0	0	0	0
vm no state	0	0	0	0
users	43	24	9	43
elapsed	305	31	17	32

Note - vm states as reported by virsh list
Data for ferm… b 2012
Plot generate… id0.fnal.gov

FermiCloud Target

FermiCloud Capacity	# of Units
Nominal (1 physical core = 1 VM)	184
50% over subscription	276
100% over subscription (1 HT core = 1 VM)	368
200% over subscription	552

Description of Virtual Machine States Reported by "virsh list" Command

State	Description
running	The domain is currently running on a CPU. Note – KVM based VMs show up in this state even when they are "idle".
idle	The domain is idle, and not running or runnable. This can be caused because the domain is waiting on I/O (a traditional wait state) or has gone to sleep because there was nothing else for it to do. Note – Xen based VMs typically show up in this state even when they are "running".
paused	The domain has been paused, usually occurring through the administrator running virsh suspend. When in a paused state the domain will still consume allocated resources like memory, but will not be eligible for scheduling by the hypervisor.
shutdown	The domain is in the process of shutting down, i.e. the guest operating system has been notified and should be in the process of stopping its operations gracefully.
shut off	The domain has been shut down. When in a shut off state the domain does not consume resources.
crashed	The domain has crashed. Usually this state can only occur if the domain has been configured not to restart on crash.
dying	The domain is in process of dying, but hasn't completely shutdown or crashed.

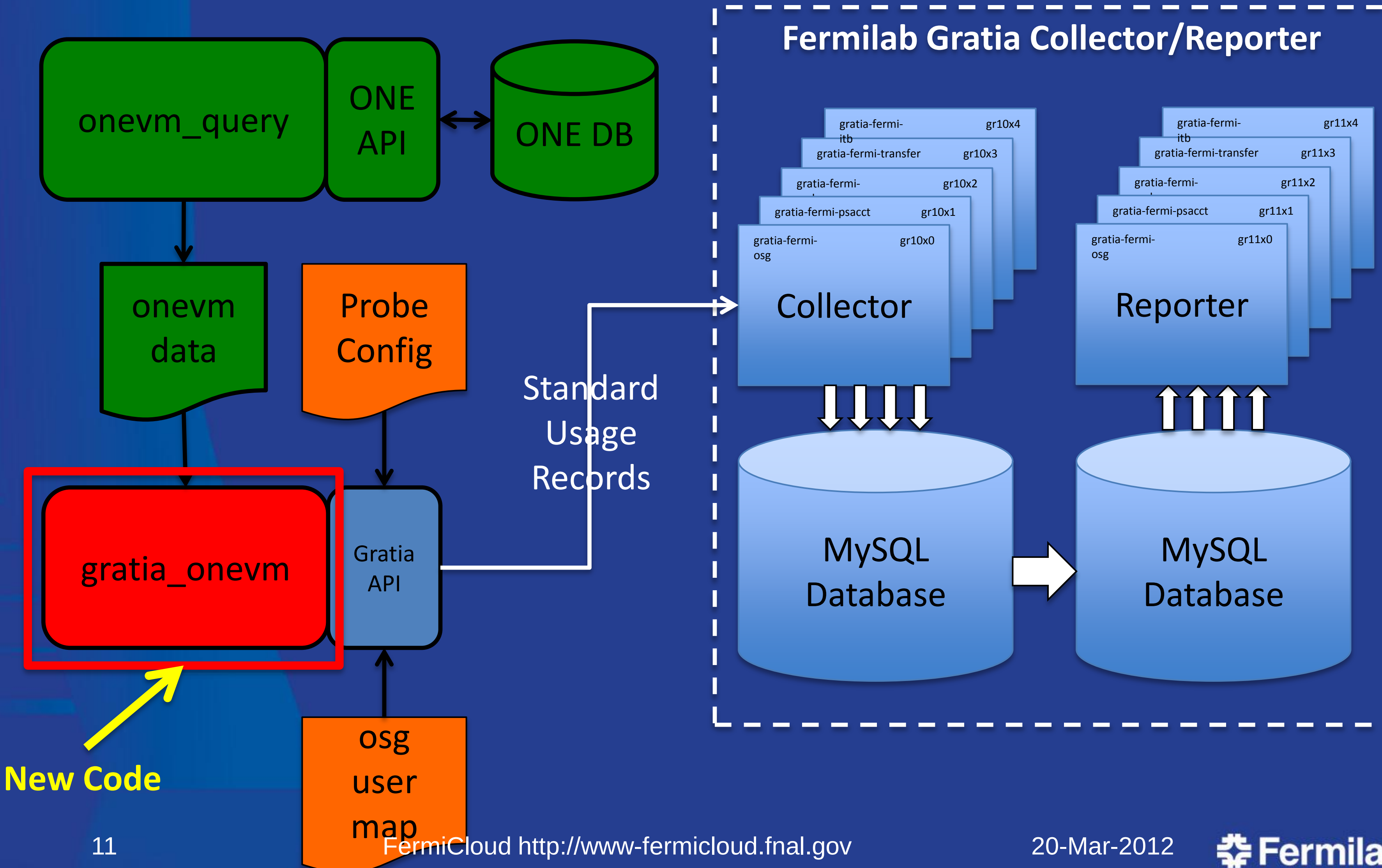
FermiCloud – Monitoring Requirements & Goals

- Need to monitor to assure that:
  - All hardware is available (both in FCC3 and GCC-B),
  - All necessary and required OpenNebula services are running,
  - All Virtual Machine hosts are healthy
  - All “24x7” & “9x5” virtual machines (VMs) are running,
  - 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.
- We plan to replace the temporary monitoring with an infrastructure based on either Nagios or Zabbix during CY2012.
  - Possibly utilizing the OSG Resource Service Validation (RSV) scripts.
  - This work will likely be performed in collaboration with KISTI.
- Goal is to identify really idle virtual machines and suspend if necessary.
  - Can’t trust hypervisor VM state output on this—Need rule-based definition
  - In times of resource need, we want the ability to suspend or “shelve” the really idle VMs in order to free up resources for higher priority usage.
  - Shelving of “9x5” and “opportunistic” VMs will allow us to use FermiCloud resources for Grid worker node VMs during nights and weekends (this is part of the draft economic model).

FermiCloud - Accounting

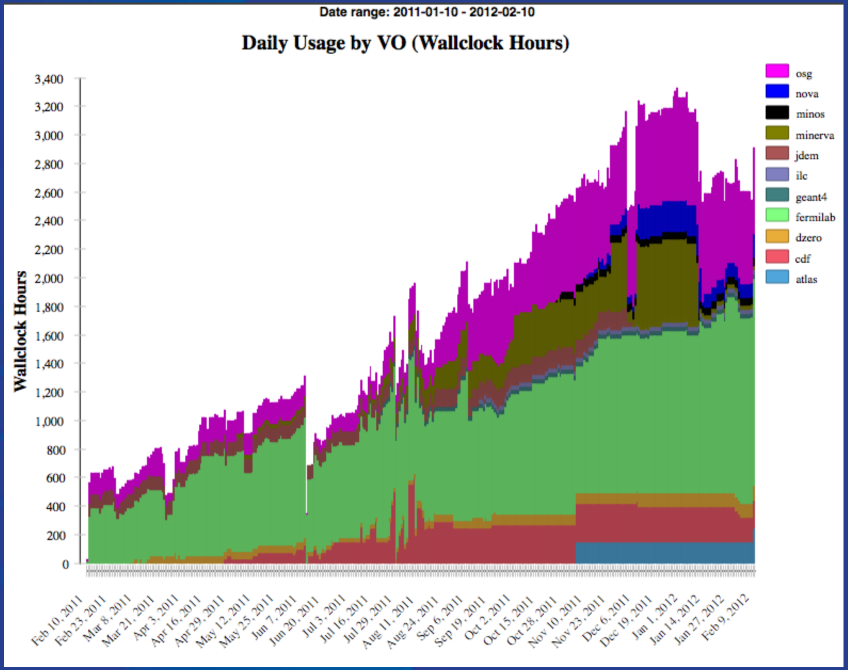
- Currently have two "probes" based on the Gratia accounting framework used by Fermilab and the Open Science Grid
- Standard Process Accounting ("psacct") Probe:
  - Installed and runs within the virtual machine image,
  - Reports to standard gratia-fermi-psacct.fnal.gov.
- Open Nebula Gratia Accounting Probe:
  - Runs on the OpenNebula management node and collects data from ONE logs, emits standard Gratia usage records,
  - Reports to the "virtualization" Gratia collector,
  - The "virtualization" Gratia collector runs existing standard Gratia collector software (no development was required),
  - The development of the Open Nebula Gratia accounting probe was performed by Tanya Levshina and Parag Mhashilkar.
- Additional Gratia accounting probes could be developed:
  - Commercial – OracleVM, VMware, ---
  - Open Source – Nimbus, Eucalyptus, OpenStack, …

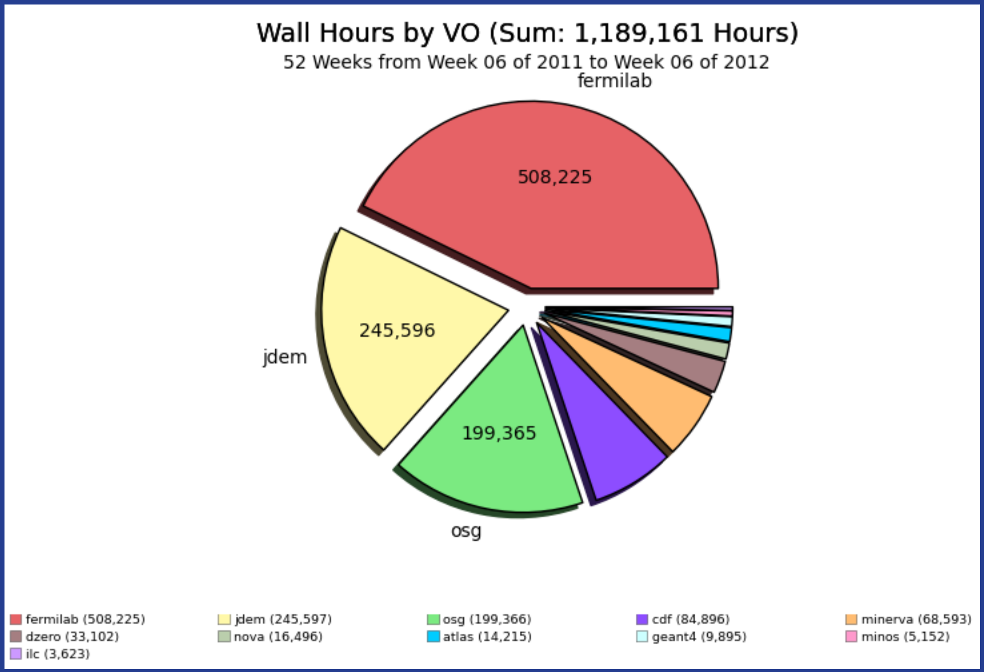
Open Nebula Gratia Accounting Probe



FermiCloud – Gratia Accounting Reports

Here are the ***preliminary*** results of “replaying” the previous year of the OpenNebula “OneVM” data into the new accounting probe:





Fermilab

vCluster

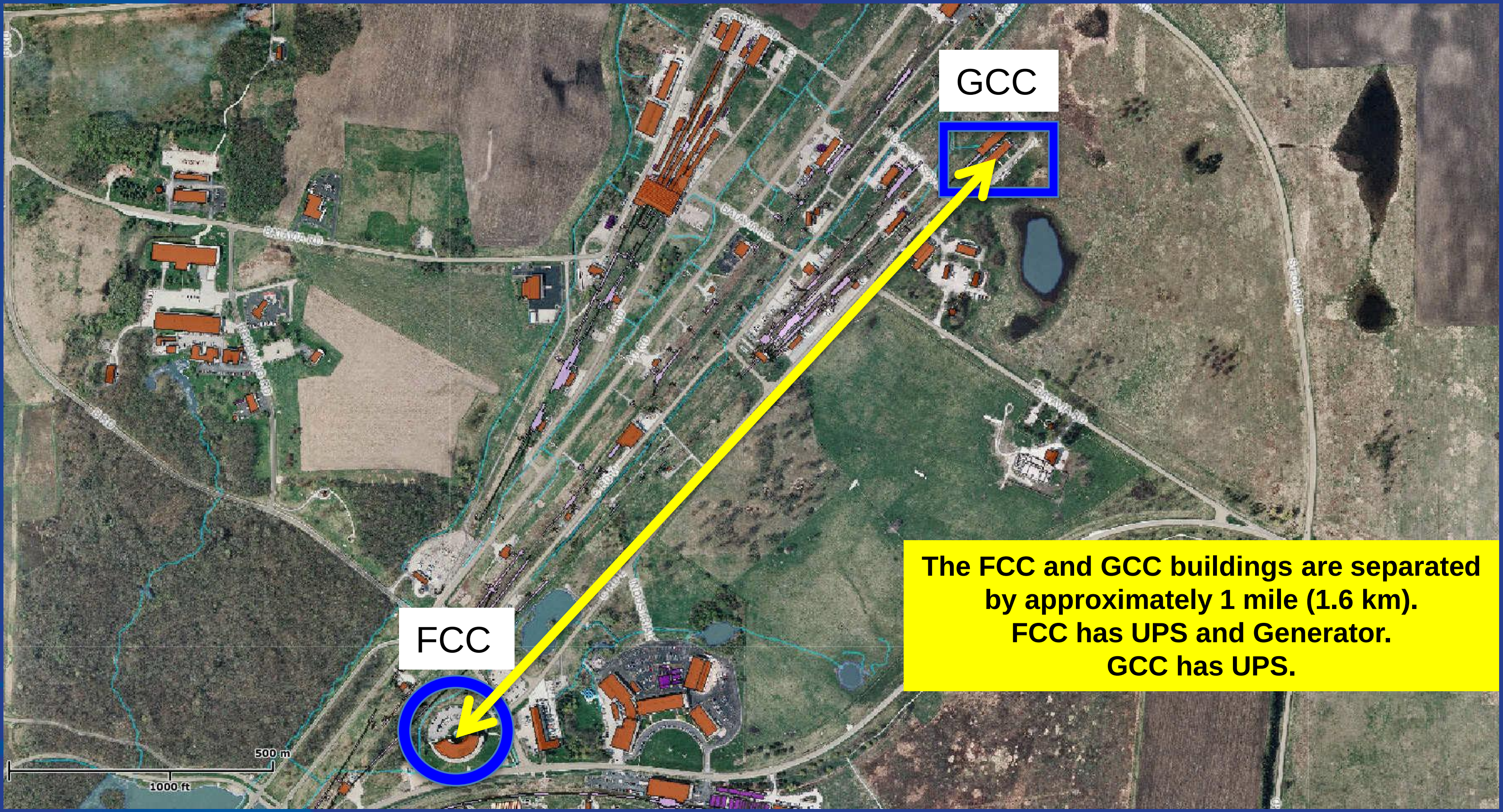
- Deployable on demand virtual cluster using hybrid cloud computing resources.
  - Head nodes launched on virtual machines within the FermiCloud private cloud.
  - Worker nodes launched on virtual machines within the Amazon EC2 public cloud and on FermiCloud
  - Work done by Dr. Seo-Young Noh, KISTI-GSDC visitor at Fermilab, summer 2011
  - Look ahead at Condor queue to see what kind of virtual machines are needed and submit as needed.

MPI on FermiCloud (Note 1)

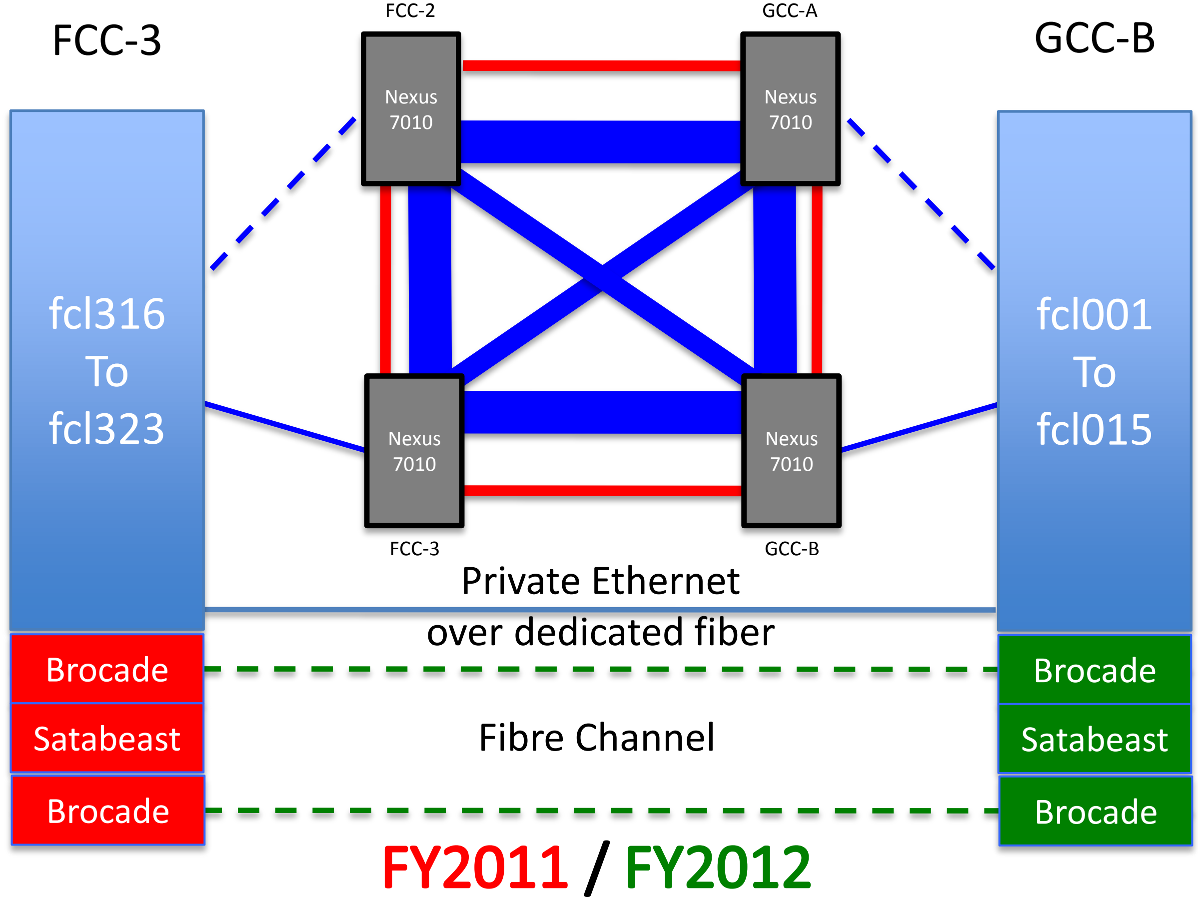
Configuration	#Host Systems	#VM/host	#CPU	Total Physical CPU	HPL Benchmark (Gflops)
Bare Metal without pinning	2	--	8	16	13.9
Bare Metal with pinning (Note 2)	2	--	8	16	24.5
VM without pinning (Notes 2,3)	2	8	1 vCPU	16	8.2
VM with pinning (Notes 2,3)	2	8	1 vCPU	16	17.5
VM+SRIOV with pinning (Notes 2,4)	2	7	2 vCPU	14	23.6

Notes:
(1) Work performed by Dr. Hyunwoo Kim of KISTI in collaboration with Dr. Steven Timm of Fermilab.
(2) Process/Virtual Machine "pinned" to CPU and associated NUMA memory via use of numactl.
(3) Software Bridged Virtual Network using IP over IB (seen by Virtual Machine as a virtual Ethernet).
(4) SRIOV driver presents native InfiniBand to virtual machine(s), 2nd virtual CPU is required to start SRIOV, but is only a virtual CPU, not an actual physical CPU.

FermiCloud Facilities

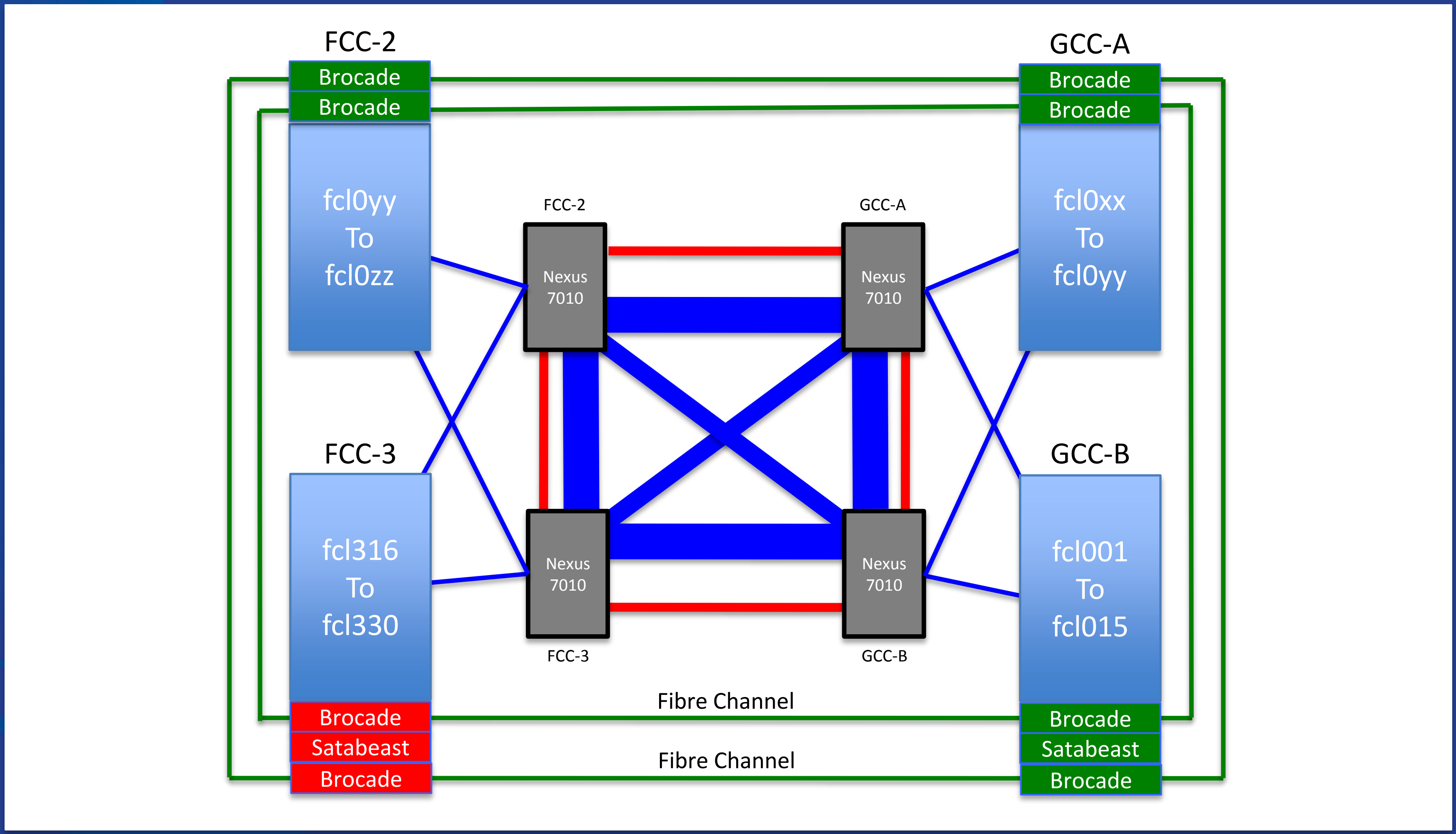


FermiCloud – Network & SAN "Today"



Note – Dashed Lines = Waiting for Equipment or Fiber

FermiCloud – Network & SAN (Possible Future – FY2013/2014)



FermiCloud – Support for Science (direct and indirect)

- Grid middleware developers and integrators as well as CDF, D0 and the Cosmic and Intensity Frontiers.
- Detailed analysis of filesystem performance for Grid applications
  - (Lustre vs. Hadoop vs. OrangeFS vs. Bluearc).
- Tuning and stress tests of GUMS-XACML and SAZ-XACML
- dCache NFS v4.1 interface
- Fabric testbeds—IPv6 for Grid and Cloud applications
- OSG Software Team integration and testing of OSG 3.0 RPMS
- Scientific control and communication system tests for JDEM and NOVA
- Production services have been deployed on FermiCloud, including:
  - Specialized GridFTP virtual machines for the Intensity Frontier experiments,
  - NFS Administration virtual machines to allow Intensity Frontier experiments to perform file ownership management of their BlueArc volumes.
  - SAMGrid forwarding nodes to send D0 jobs to WLCG sites via GlideinWMS

Recent FermiCloud Uses

Application/Project	Stakeholder
JDEM Demo Data Processing System	LSST
OSG Software Team build and test	OSG
GEANT4 validation server	CET
GlobusOnline test work (CEDPS)	OSG & CEDPS
CVMFS testing	IF
Test GUMS and SAZ servers	FermiGrid
Extenci Lustre-over-WAN	OSG
Vcluster demo (grid cluster in the cloud)	GCC & KISTI
Expt-specific gridftp servers (MINOS, NOVA)	IF
SAMGrid forwarding nodes	D0
dCache NFS4.1 testing	REX & DMS
Scientist survey	Directorate
MCAS	GCC
OSG-XSEDE interoperability	OSG, XSEDE

FermiCloud Conclusions

- The existing (temporary) FermiCloud usage monitoring shows that the peak FermiCloud usage is ~100% of the nominal capacity and ~50% of the expected oversubscription capacity.
- The FermiCloud collaboration with KISTI has leveraged the resources and expertise of both institutions to achieve significant benefits.
- FermiCloud has plans to implement both monitoring and accounting by extension of existing tools in CY2012.
- Using SRIOV drivers on FermiCloud virtual machines, MPI performance has been demonstrated to be >96% of the native “bare metal” performance.
  - Note that this HPL benchmark performance measurement was accomplished using **2 fewer** physical CPUs than the corresponding “bare metal” performance measurement!
- FermiCloud personnel are working to implement a SAN storage deployment that will offer a true multi-user filesystem on top of a distributed & replicated SAN.
- Science is directly and indirectly benefiting from FermiCloud.

Summary

- FermiCloud operates at the forefront of delivering cloud computing capabilities to support physics 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 an Infrastructure as a Service cloud computing capability that supports science at Fermilab.
  - The Open Science Grid software team is using FermiCloud resources to support their RPM “refactoring”.
- Thanks to:
  - Open Source collaborators—OpenNebula and Scientific Linux
  - Fermilab Networking and Facilities team
  - Collaborators from KISTI-GSDC
- We welcome collaborators and visitors!
- Keep an eye on Fermilab’s employment website for opening (hopefully) coming soon.

Extra Slides

FermiCloud vs. Magellan

FermiCloud	Magellan
Bottom up requirements and design.	Top down requirements and mission.
Funded out of existing budget ($230K+$128K).	Funded via ARRA ($32M).
Multi phase project, with each phase building on knowledge gained during previous phases.	Fixed term project without ongoing funding.
Evaluated available open source cloud computing frameworks (Eucalyptus, Nimbus, OpenNebula) against requirements and selected OpenNebula. We plan to “circle back” and evaluate OpenStack this year.	Spent a lot of time trying to get the open source version of Eucalyptus to work at scale, eventually switched to a combination of Nimbus and OpenStack late in the project.
Approached cloud computing from a Grid and high throughput computing (HTC) perspective.	Approached cloud computing from a high performance computing (HPC) perspective.
Significant prior experience delivering production Grid services via open source virtualization (Xen and KVM).	Unclear.
Have SRIOV drivers for InfiniBand.	Did not have SRIOV drivers for InfiniBand before the end of the project.
Actively sought collaboration (OpenNebula, KISTI).	Project was sited at NERSC and Argonne.