FermiCloud Update
EGI Technical Forum 2012

Keith Chadwick
Grid & Cloud Computing Department
Fermilab

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

Fermilab

Outline

Personnel
Mission, Strategy, Project, Phases
Economic Model
Authentication and Contextualization
Fault Tolerance and High Availability (HA)
Monitoring, Accounting
Grid and Cloud "Bursting"
Virtualized MPI
Stakeholders
Summary

Fermilab Grid & Cloud Computing Department

Keith Chadwick (Department Head)
Gabriele Garzoglio (Associate Head)

Distributed Offline Computing Services
Gabriele Garzoglio (Leader)
David Dykstra
Giovanni Franzini (Summer Student)
~~Ted Hesselroth (departed Sep '11)~~
Hyunwoo Kim
Tanya Levshina
Parag Mhashilkar
Marko Slyz
Douglas Strain

FermiGrid Services
Steven C. Timm (Leader)
Hyunwoo Kim (Visitor - KISTI)
Kevin Hill (OSG Security)
~~Faaroog Lowe (departed 30-Mar)~~
~~Seo-Young Noh (Visitor – KISTI)~~
Neha Sharma
Karen Shepelak
~~Daniel R. Yocum (departed 21-Jun)~~
New Hire (starting 8 Oct)
Posted Opening (soon)

FermiCloud – Mission, Strategy, Goals & History

As part of the FY2010 activities, the (then) Grid Facilities Department established a project to implement an initial "FermiCloud" capability with the goal of developing and establishing Scientific Cloud capabilities for the Fermilab Scientific Program,

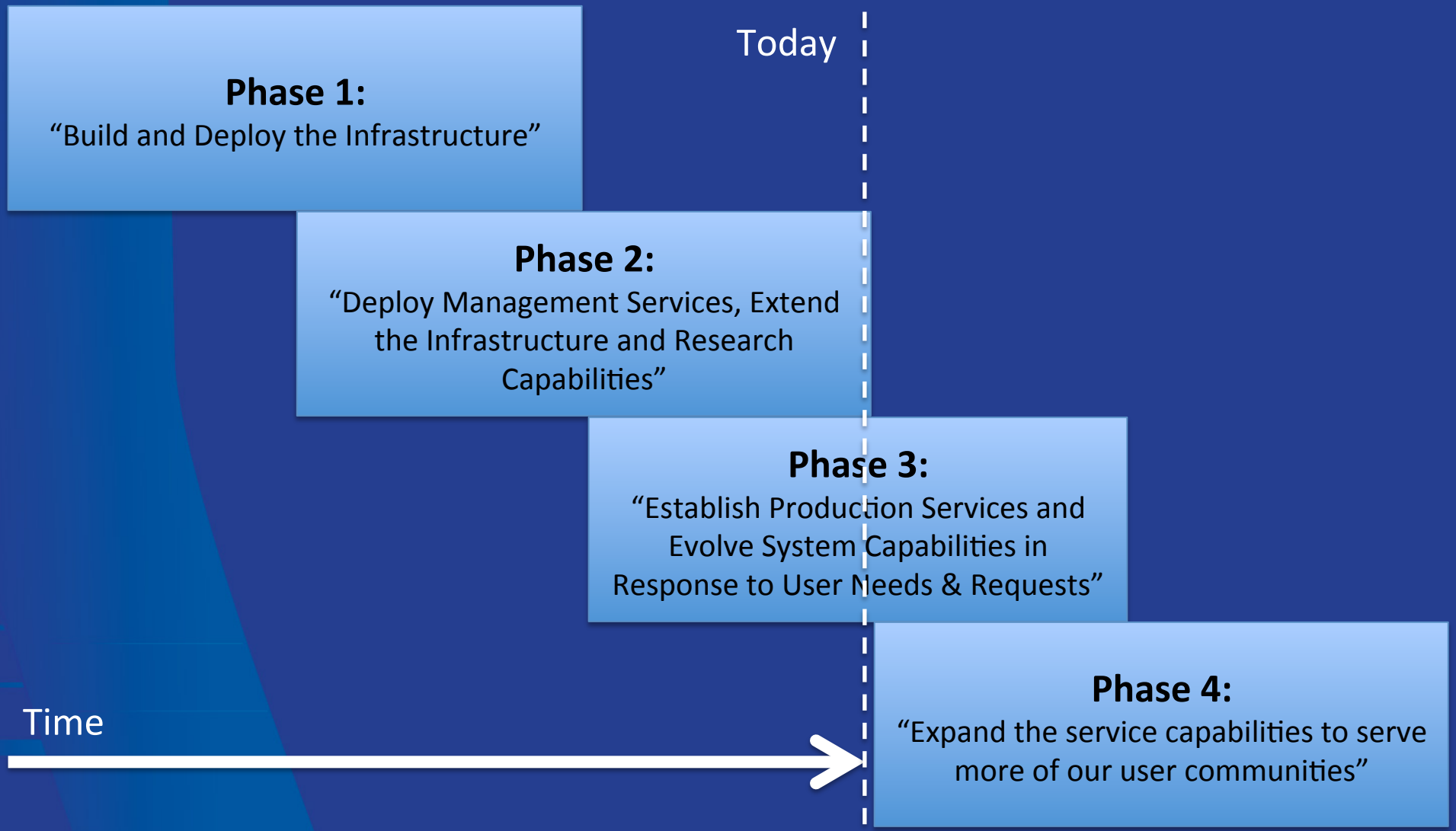
- Building on the very successful FermiGrid program that supports the full Fermilab user community and makes significant contributions as members of the Open Science Grid Consortium.

In a (very) broad brush, the mission of FermiCloud is:

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

This project is split over several overlapping phases.

Overlapping Phases



FermiCloud Phase 1: "Build and Deploy the Infrastructure"

- 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).

Completed

FermiCloud Phase 2: "Deploy Management Services, Extend the Infrastructure and Research Capabilities"

- Implement x509 based authentication (patches contributed back to OpenNebula project and are generally available in OpenNebula V3.2), perform secure contextualization of virtual machines at launch.
- Perform virtualized filesystem I/O measurements,
- Develop (draft) economic model,
- Implement monitoring and accounting,
- Collaborate with KISTI personnel to demonstrate Grid and Cloud Bursting capabilities and perform initial benchmarks of Virtualized MPI,
- Target "small" low-cpu-load servers such as Grid gatekeepers, forwarding nodes, small databases, monitoring, etc.,
- Begin the hardware deployment of a distributed SAN,
- Investigate automated provisioning mechanisms (puppet & cobbler).

Almost Complete

FermiCloud Phase 3:
"Establish Production Services and Evolve System Capabilities in Response to User Needs & Requests"

- Deploy highly available 24x7 production services,
  - Both infrastructure and user services.
- Deploy puppet & cobbler in production,
- Develop and deploy redundant machine detection,
- Research possibilities for a true multi-user filesystem on top of a distributed & replicated SAN,
- Live migration becomes important for this phase.

Underway

FermiCloud Phase 4:
"Expand the service capabilities to serve more of our user communities"

- Deploy a true multi-user filesystem on top of a distributed & replicated SAN,
- Demonstrate interoperability as well accepting VM's as batch jobs, interoperating with KISTI cloud, Nimbus, etc.,
- Participate in Fermilab 100 Gb/s network testbed,
- Perform more "virtualized MPI" benchmarks and run some real world MPI codes,
- OpenStack evaluation.

Specifications Under Development

FermiCloud Economic Model

Calculate rack cost:

- Rack, public Ethernet switch, private Ethernet switch, Infiniband switch,
- $11,000 USD (one time).

Calculate system cost:

- Based on 4 year lifecycle,
- $6,500 USD / 16 processors / 4 years = $250 USD / year

Calculate storage cost:

- 4 x FibreChannel switch, 2 x SATAbeast, 5 year lifecycle,
- $130K USD / 60 Tbytes / 5 years = $430 USD / TB-year

Calculate fully burdened system administrator cost:

- Current estimate is 400 systems per administrator,
- $250K USD / year / 400 systems = $1,250 USD / system-year

Service Level Agreements

24x7:
Virtual machine will be deployed on the FermiCloud infrastructure 24x7. Typical use case – production services.

9x5:
Virtual machine will be deployed on the FermiCloud infrastructure 8-5, M-F, may be "suspended or shelved" at other times. Typical use case – software/middleware developer.

Opportunistic:
Virtual machine may be deployed on the FermiCloud infrastructure providing that sufficient unallocated virtual machine "slots" are available, may be "suspended or shelved" at any time. Typical use case – short term computing needs.

HyperThreading / No HyperThreading:
Virtual machine will be deployed on FermiCloud infrastructure that [has / does not have] HyperThreading enabled.

Nights and Weekends:
Make FermiCloud resources (other than 24x7 SLA) available for "Grid Bursting".

FermiCloud Draft Economic Model Results (USD)

SLA	24x7		9x5		Opportunistic
	No HT	HT	No HT	HT	--
"Unit" (CPU + 2 GB)	$250	$125	$90	$45	$24
Add'l memory per GB	$30	$30	$30	$30	$30
Add'l local disk per TB	$40	$40	$40	$40	$40
SAN disk per TB	$450	$450	$450	$450	$450
BlueArc per TB	$430	$430	$430	$430	$430
System Administrator	$1,250	$1,250	$1,250	$1,250	$1,250

FermiCloud / Amazon Cost Comparison (USD)

SLA (CPU Only)	FermiCloud	EC2 Small	EC2 Large	EC2 High CPU Medium
24x7 No HT	$250/yr	$220.50/yr	$910.00/yr	$455.00/yr
24x7 With HT	$125/yr	n/a	n/a	n/a
9x5 No HT	$90/yr	n/a	n/a	n/a
9x5 With HT	$45/yr	n/a	n/a	n/a
Opportunistic	$25/yr $0.00285/hr	$0.02/hr	$0.34/hr	$0.17/hr

FermiCloud - Authentication

Reuse Grid x509 based authentication:

- Patches to OpenNebula to support this were developed at Fermilab and submitted back to the OpenNebula project (generally available in OpenNebula V3.2 onwards).

User authenticates to OpenNebula via graphical console via x509 authentication, EC2 query API with x509, or OCCI with x509,

Virtual Machines are launched with the users x509 proxy.

FermiCloud - Contextualization

Within FermiCloud, virtual machine contextualization is accomplished via:

- Use users x509 proxy credentials to perform secure access via openSSL to an external "secure secrets repository",
- Credentials are copied into ramdisk within the VM and symlinks are made from the standard credential locations (/etc/grid-security/certificates) to the credentials in the ramdisk.

On virtual machine shutdown, the contents of the ramdisk disappear and the original credential remains in the "secure secrets repository".

These mechanisms prevent the "misappropriation" of credentials if a VM image is copied from the FermiCloud VM library,

- No credentials are stored in VM images "at rest" in the VM library.

This is not perfect – a determined user (VM administrator) could still copy the secure credentials off of their running VM, but this does not offer any additional risk beyond that posed by the administrator of a physical system.

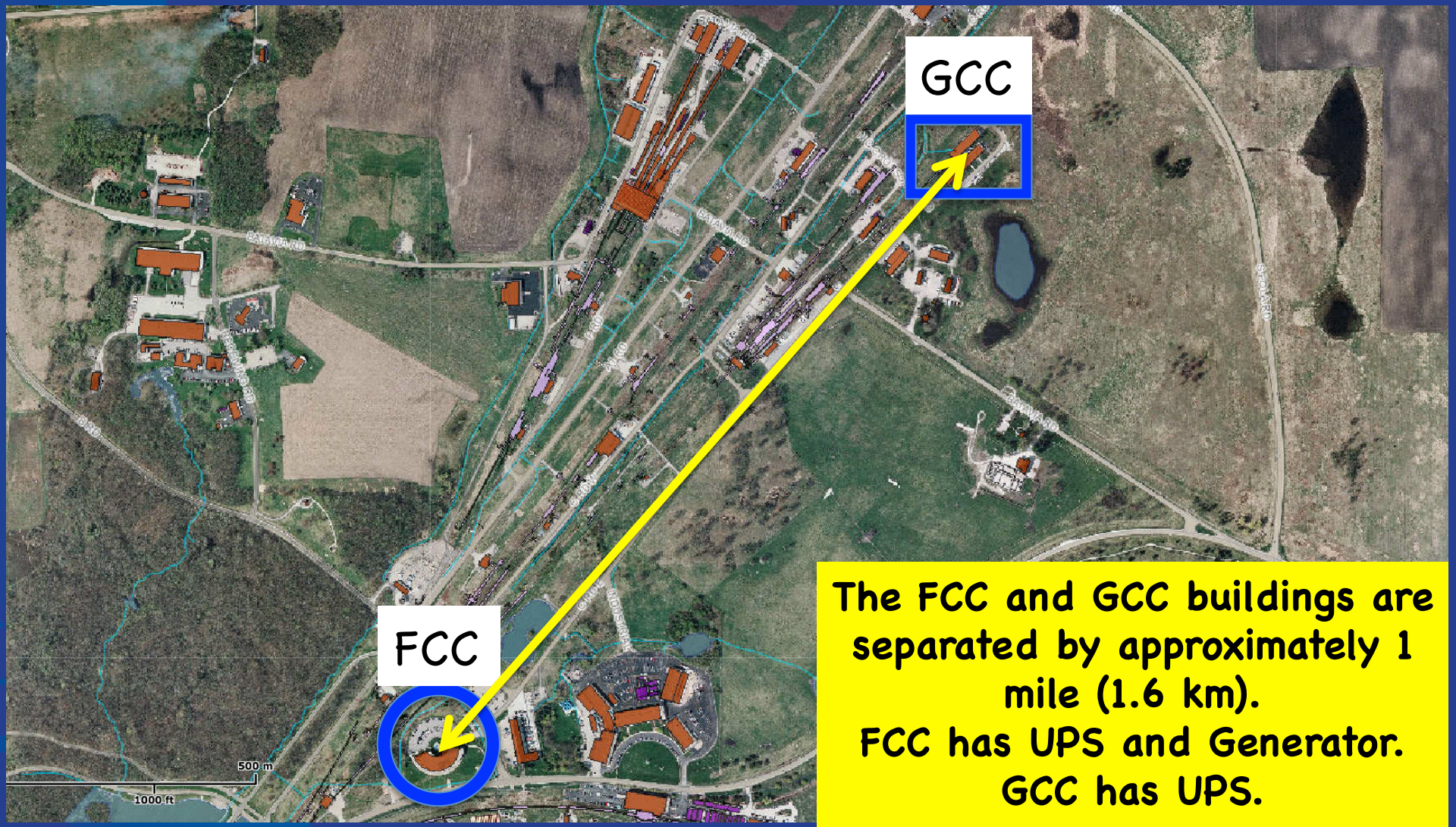
FermiCloud – Fault Tolerance

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:

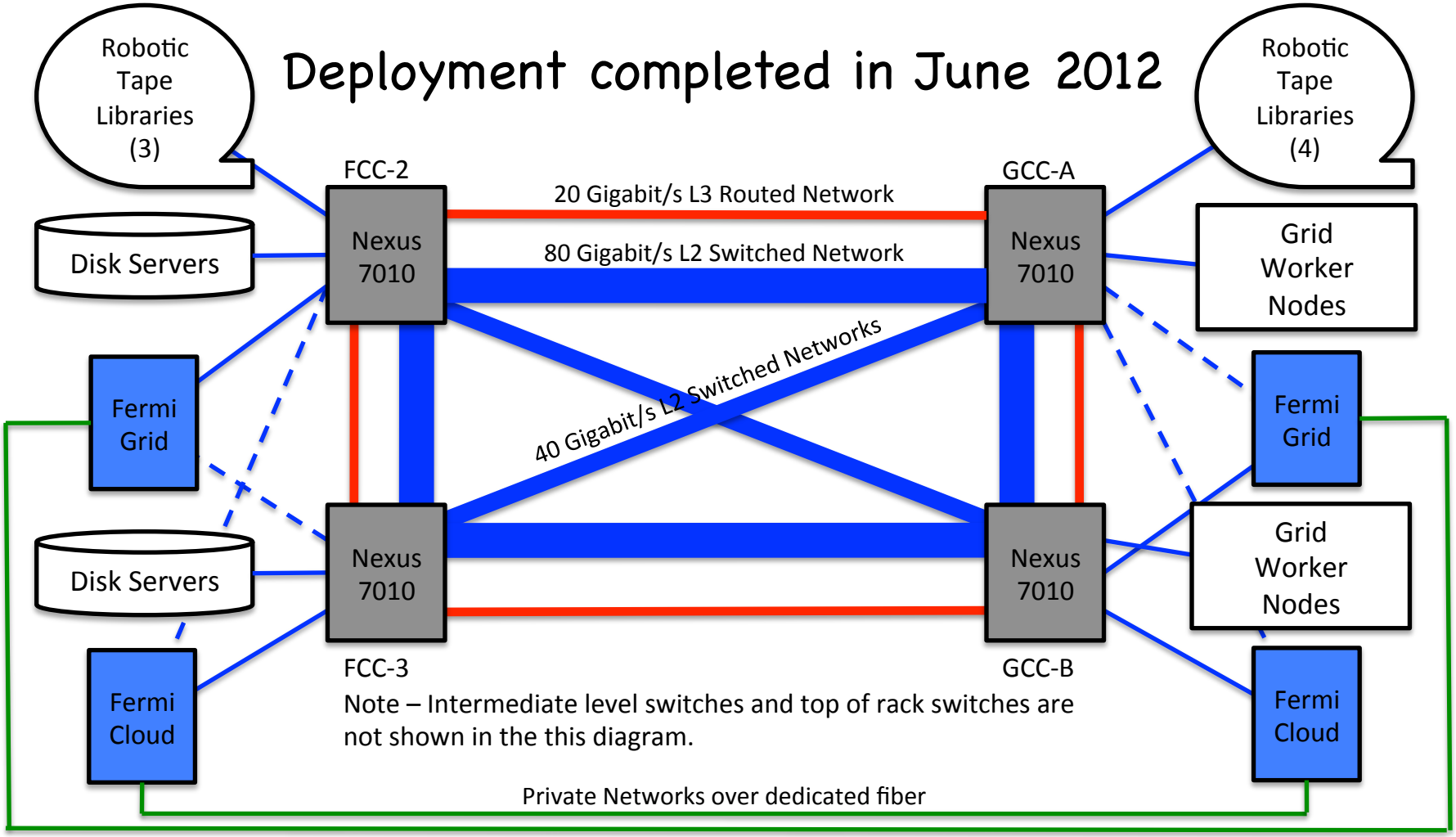
- The physical systems are split across two buildings,
- There is a fault tolerant network infrastructure in place that interconnects the two buildings,
- We have deployed SAN hardware in both buildings,
- We are finalizing the FermiCloud HA "head node" configuration,
- We are evaluating GFS for our multi-user filesystem and distributed & replicated SAN.

FCC and GCC

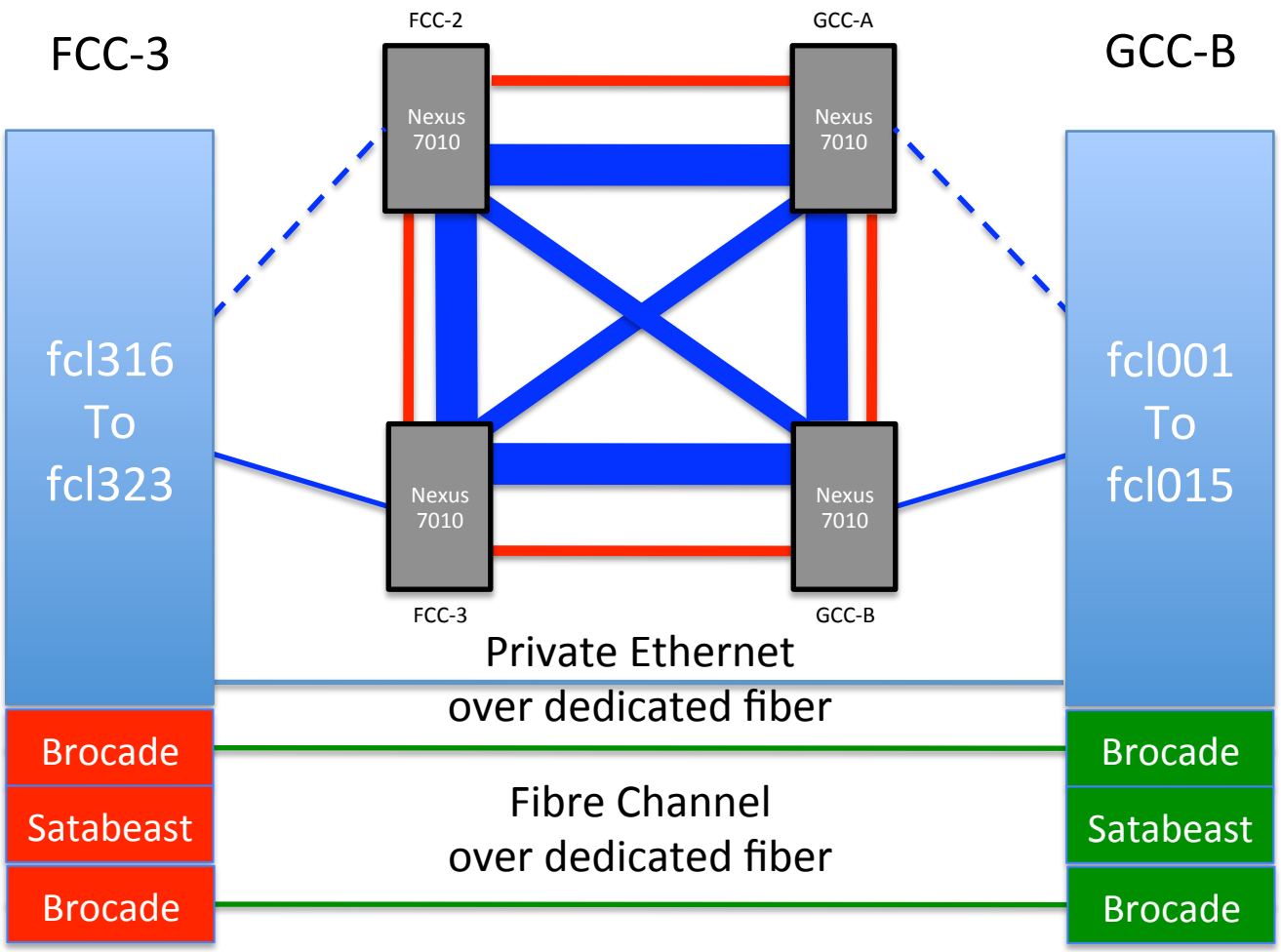


The FCC and GCC buildings are separated by approximately 1 mile (1.6 km).
FCC has UPS and Generator.
GCC has UPS.

Distributed Network Core Provides Redundant Connectivity

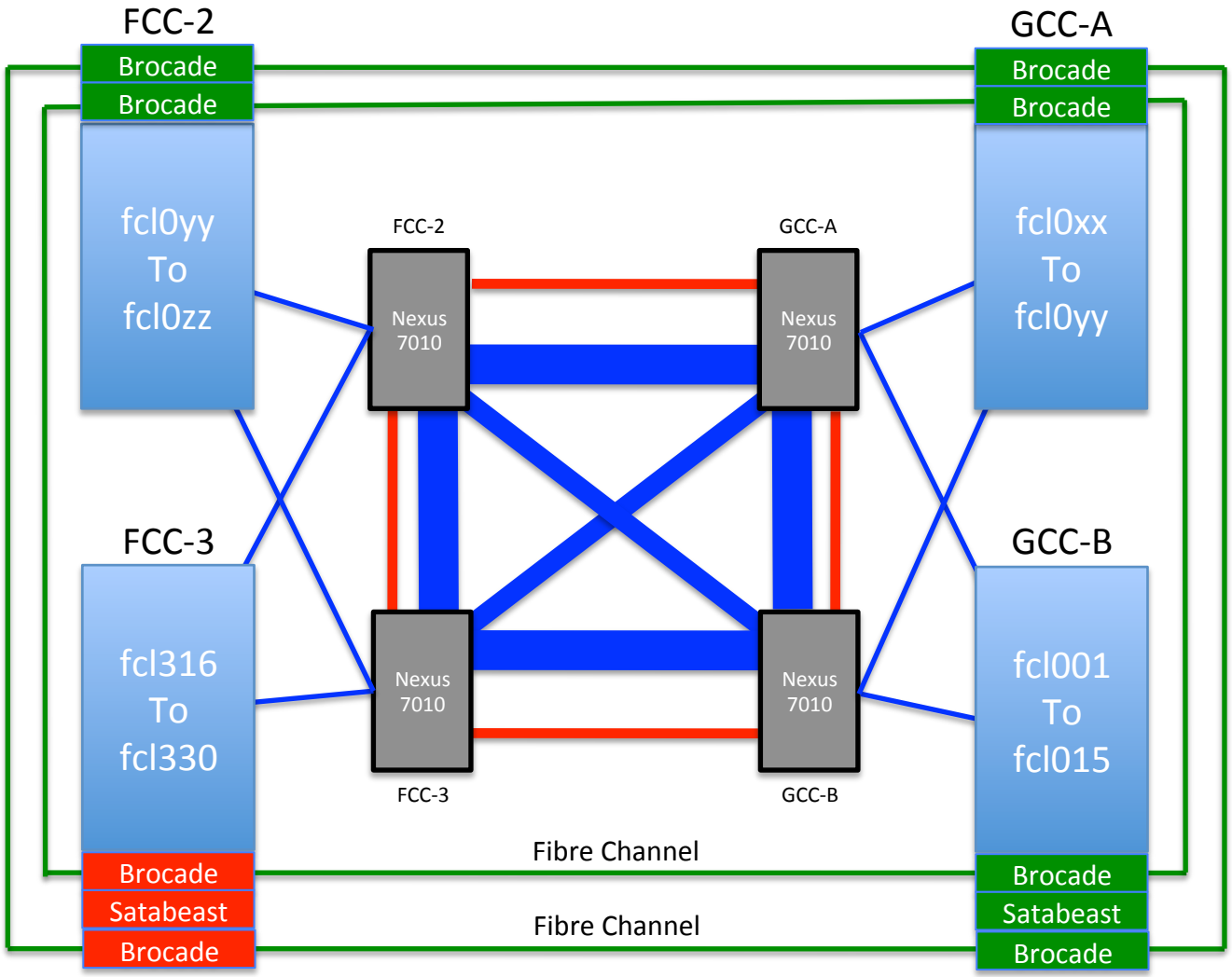


FermiCloud – Network & SAN "Today"

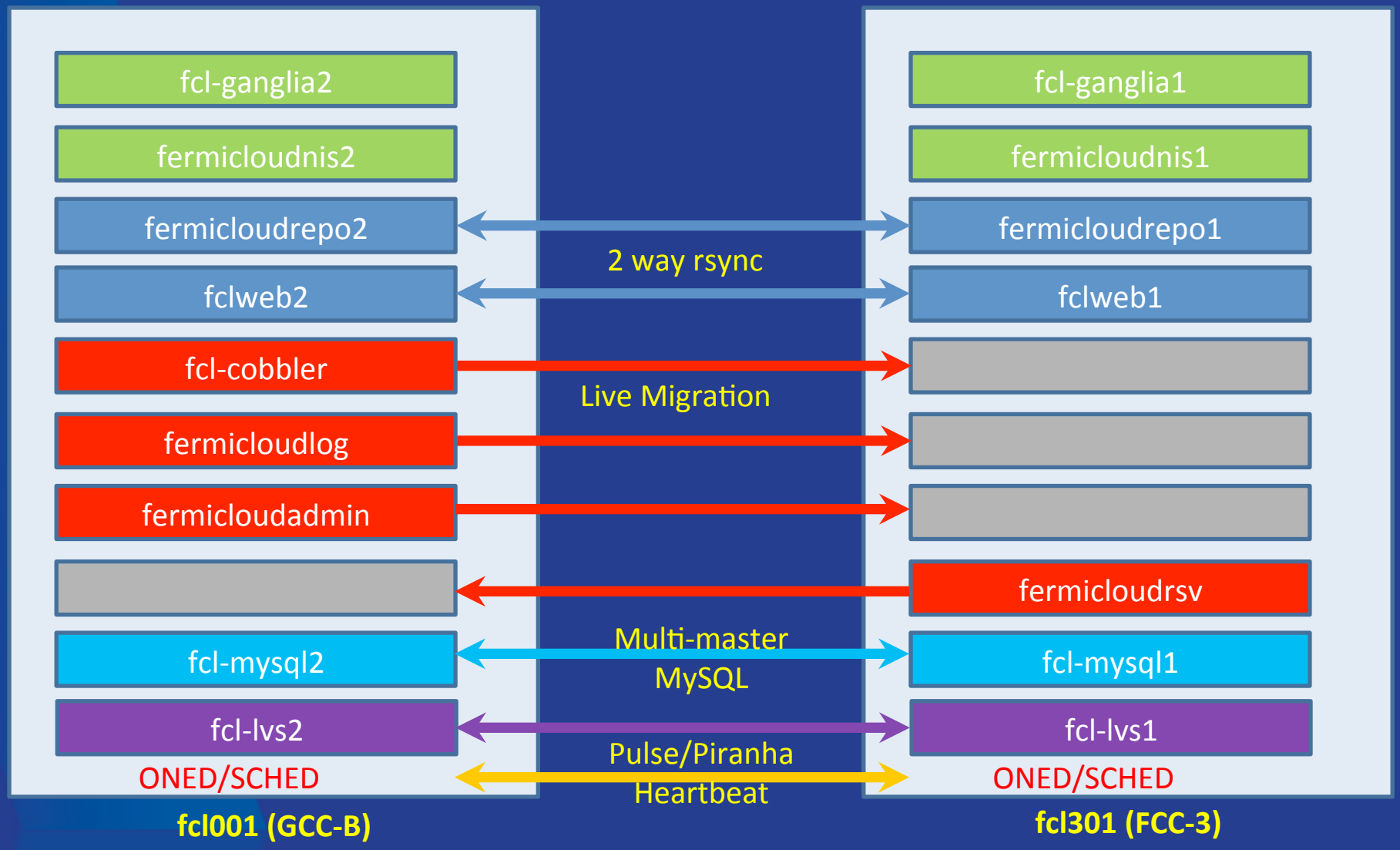


FY2011 / FY2012

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



FermiCloud-HA Head Node Configuration



FermiCloud – Monitoring Requirements & Goals

Need to monitor to assure that:

- All hardware is available (both in FCC-3 and GCC-B),
- All necessary and required OpenNebula services are running,
- 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.

FermiCloud – Monitoring

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.

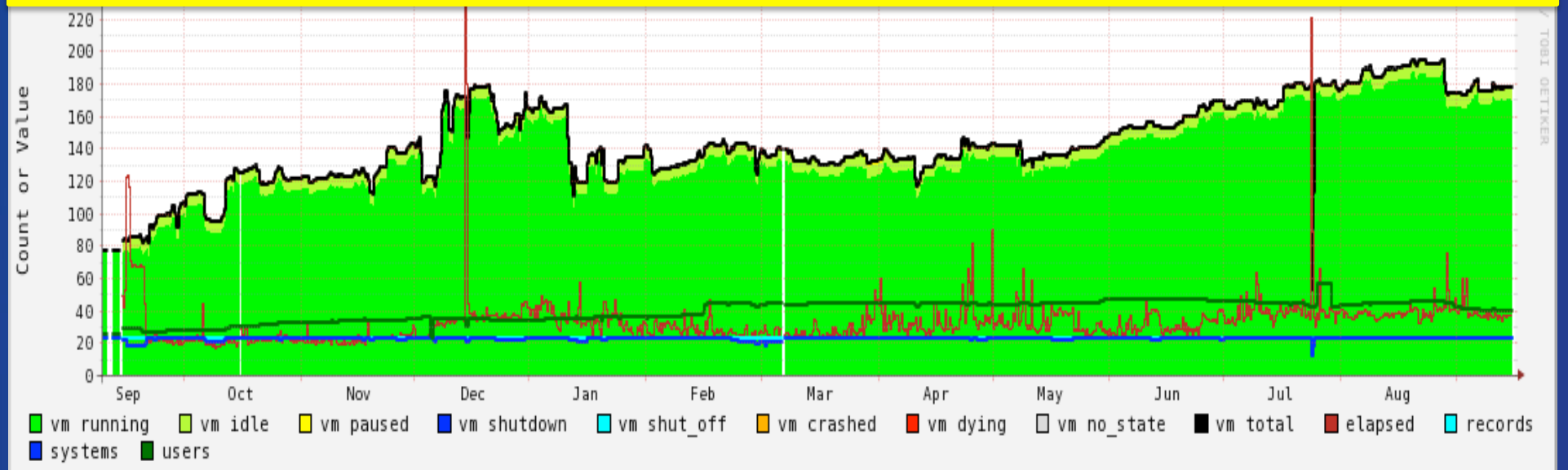
We have deployed a production monitoring infrastructure based on Nagios:

- Utilizing the OSG Resource Service Validation (RSV) scripts.

A "stretch" goal of the monitoring project was to figure out how to identify really idle virtual machines:

- Unfortunately, the "virsh list" output cannot be used, since actively running Xen based VMs are incorrectly labeled as "idle" and idle KVM based VMs are incorrectly labeled as "running".

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



VM states as reported by "virsh list"

	Maximum	Average	Minimum	Last Val
records	23	23	23	23
systems	23	23	13	23
vm total	195	144	53	178
vm running	187	137	52	170
vm idle	7	7	1	7
vm paused	2	0	0	1
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	56	40	23	40
elapsed	305	33	17	37

Note - vm states as reported by virsh list

Data for ferm… …2012

Plot generate… …d0.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

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.

- This is especially important in the event of constrained resources (e.g. during building or network failure).

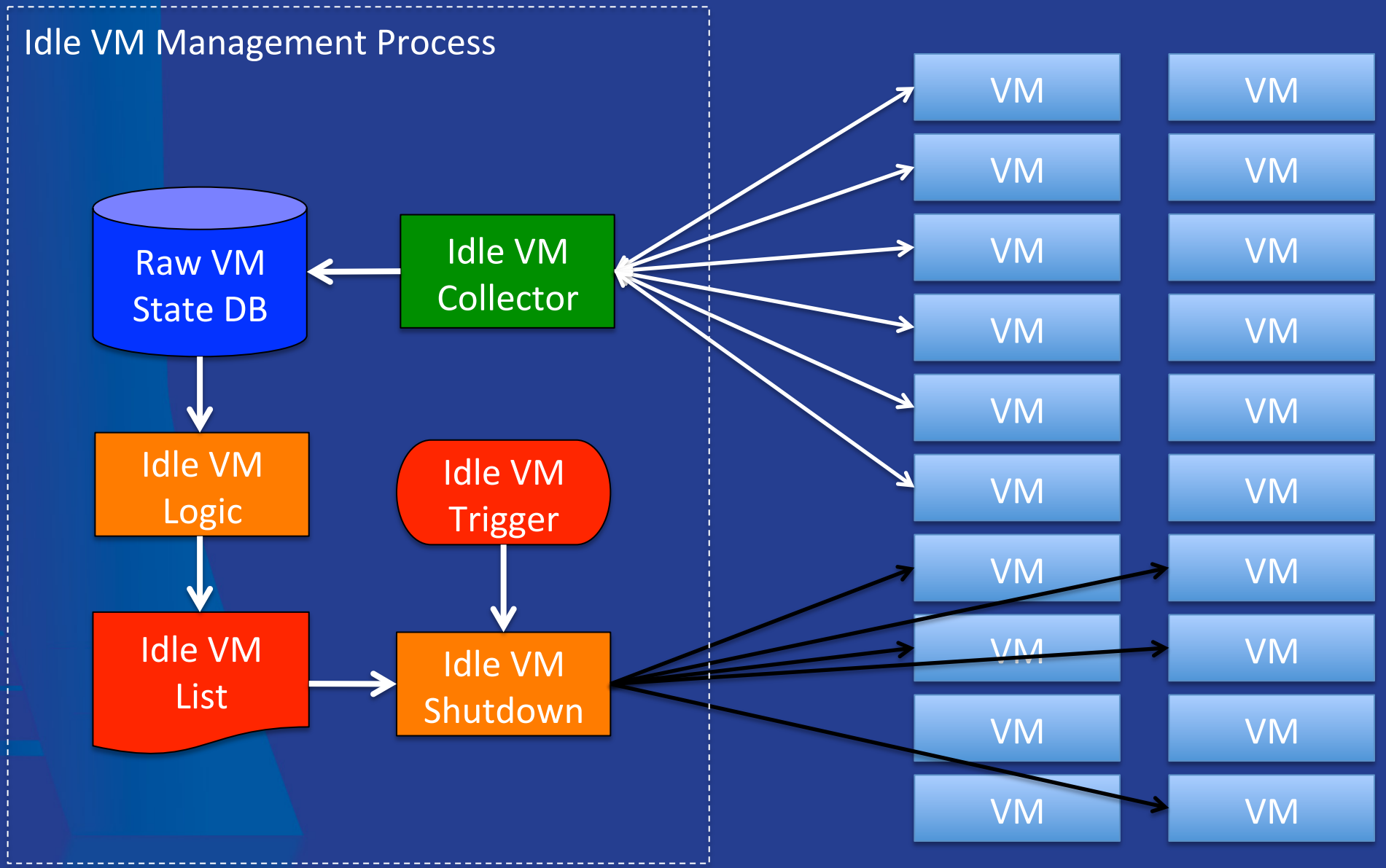
Shelving of "9x5" and "opportunistic" VMs allows us to use FermiCloud resources for Grid worker node VMs during nights and weekends

- This is part of the draft economic model.

Giovanni Franzini (an Italian co-op student) has written (extensible) code for an "Idle VM Probe" that can be used to detect idle virtual machines based on CPU, disk I/O and network I/O. The deployment will be along the lines of the rup/rusers service:

- A central "collector" will periodically contact the idle VM probe to determine the state of all the "running" virtual machines, and store the results.
- The VM state results will be processed by a separate "logic" engine that will determine candidate "idle" virtual machines,
- When necessary, the appropriate commands will be issued to shut down "idle" virtual machines (typically to make resources available for higher priority virtual machines).

Idle VM Information Flow



FermiCloud - Accounting

Currently have two "probes" based on the Gratia accounting framework used by Fermilab and the Open Science Grid:

- https://twiki.grid.iu.edu/bin/view/Accounting/WebHome

Standard Process Accounting ("psacct") Probe:

- Installed and runs within the virtual machine image,
- Reports to standard gratia-fermi-psacct.fnal.gov.

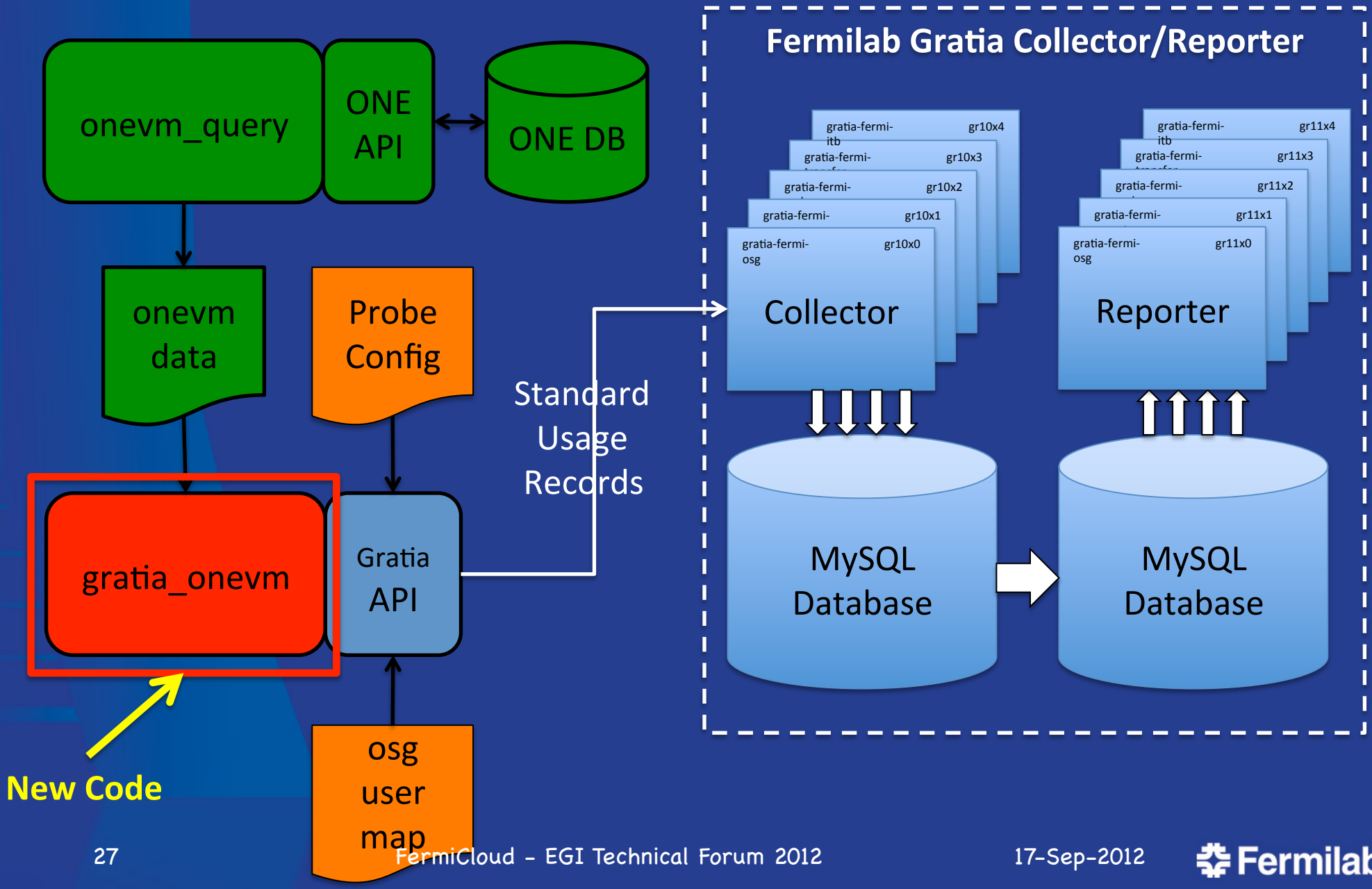
Open Nebula Gratia Accounting Probe:

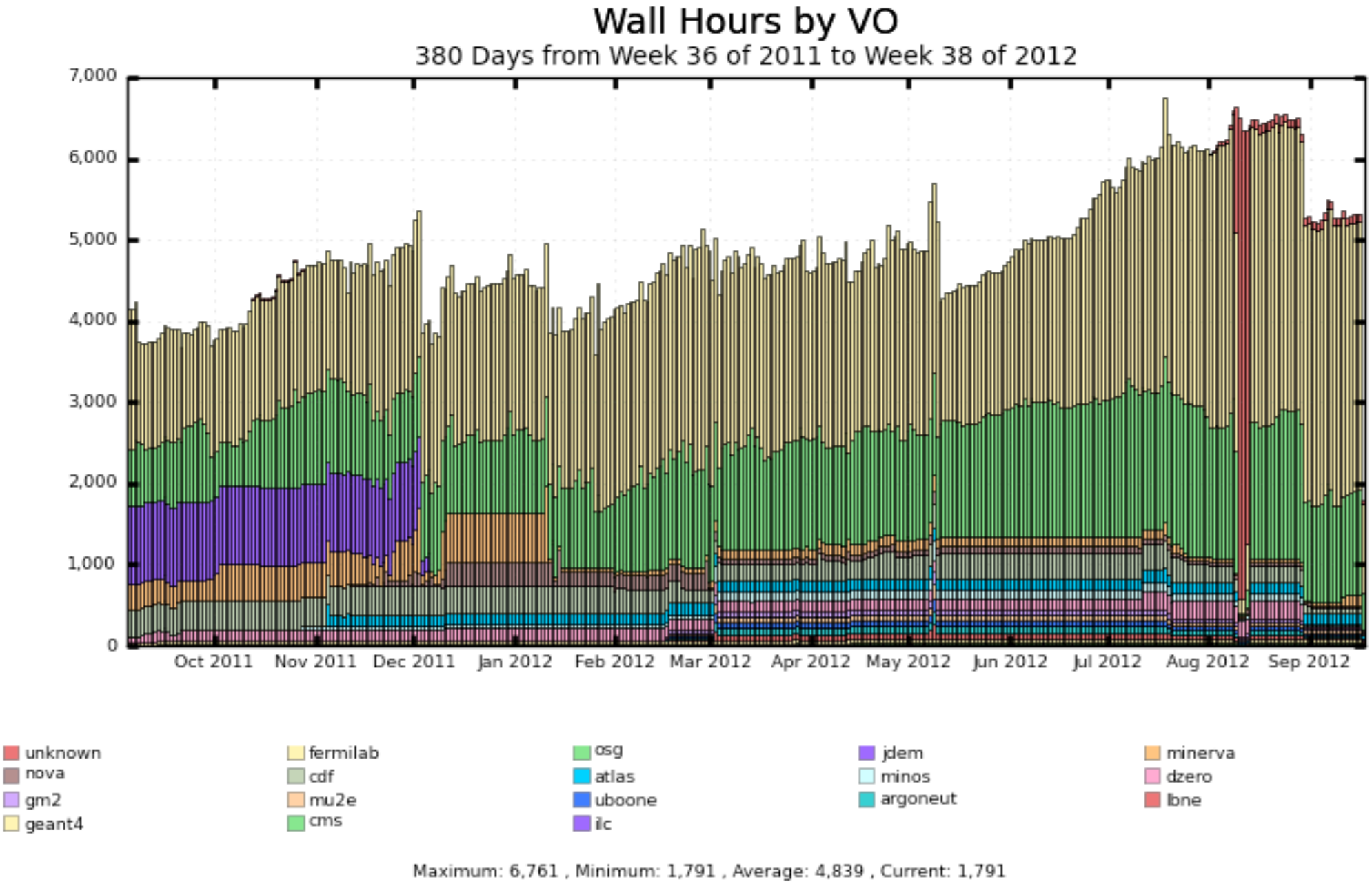
- Runs on the OpenNebula management node and collects data from ONE logs, emits standard Gratia usage records,
- Reports to the "virtualization" Gratia collector,
- The "virtualization" Gratia collector runs existing standard Gratia collector software (no development was required),
- The development of the Open Nebula Gratia accounting probe was performed by Tanya Levshina and Parag Mhashilkar.

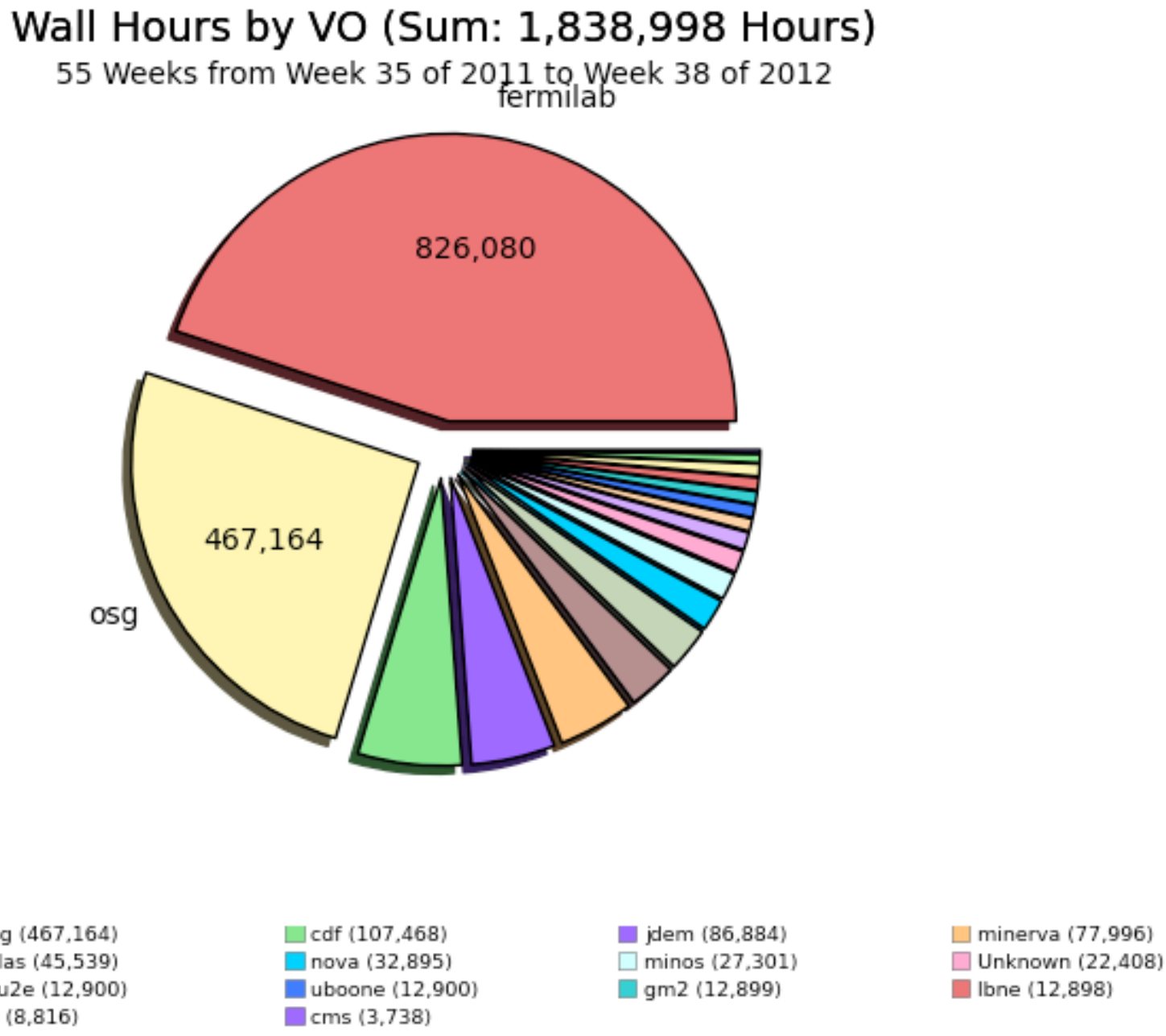
Additional Gratia accounting probes could be developed:

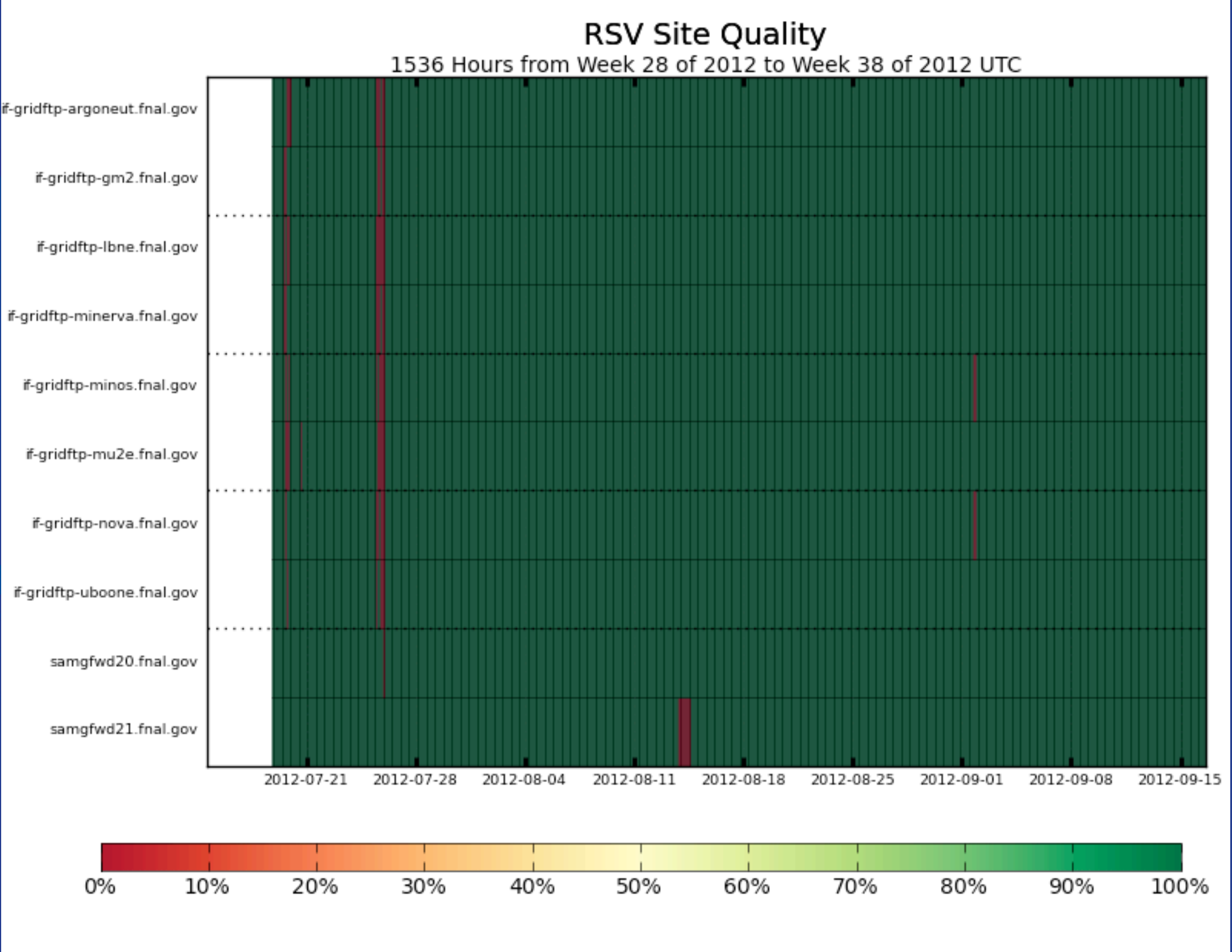
- Commercial – OracleVM, VMware, ---
- Open Source – Nimbus, Eucalyptus, OpenStack, …

Open Nebula Gratia Accounting Probe




Wall Hours by VO
380 Days from Week 36 of 2011 to Week 38 of 2012
7,000
6,000
5,000
4,000
3,000
2,000
1,000
0
Oct 2011
Nov 2011
Dec 2011
Jan 2012
Feb 2012
Mar 2012
Apr 2012
May 2012
Jun 2012
Jul 2012
Aug 2012
Sep 2012
unknown
nova
gm2
geant4
fermilab
cdf
mu2e
cms
osg
atlas
uboone
ilc
jdem
minos
argoneut
minerva
dzero
lbne
Maximum: 6,761 , Minimum: 1,791 , Average: 4,839 , Current: 1,791


Wall Hours by VO (Sum: 1,838,998 Hours)
55 Weeks from Week 35 of 2011 to Week 38 of 2012
fermilab
826,080
467,164
osg
fermilab (826,081)
osg (467,164)
cdf (107,468)
jdem (86,884)
minerva (77,996)
dzero (50,173)
atlas (45,539)
nova (32,895)
minos (27,301)
Unknown (22,408)
geant4 (18,058)
mu2e (12,900)
uboone (12,900)
gm2 (12,899)
lbne (12,898)
argoneut (12,882)
ilc (8,816)
cms (3,738)


RSV Site Quality
1536 Hours from Week 28 of 2012 to Week 38 of 2012 UTC
if-gridftp-argoneut.fnal.gov
if-gridftp-gm2.fnal.gov
if-gridftp-lbne.fnal.gov
if-gridftp-minerva.fnal.gov
if-gridftp-minos.fnal.gov
if-gridftp-mu2e.fnal.gov
if-gridftp-nova.fnal.gov
if-gridftp-uboone.fnal.gov
samgfwd20.fnal.gov
samgfwd21.fnal.gov
2012-07-21
2012-07-28
2012-08-04
2012-08-11
2012-08-18
2012-08-25
2012-09-01
2012-09-08
2012-09-15
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%

FermiCloud - Interoperability

From the beginning, one of the goals of FermiCloud has been the ability to operate as a hybrid cloud:

- Being able to join FermiCloud resources to FermiGrid resources to temporarily increase the Grid capacity or GlideinWMS with VMs (Grid Bursting),
- Being able to join public cloud resources (such as Amazon EC2) to FermiCloud (Cloud Bursting via Public Clouds).
- Participate in compatible community clouds (Cloud Bursting via other Private Clouds).

FermiCloud - Grid Bursting

Join "excess" FermiCloud capacity to FermiGrid:

- Identify "idle" VMs on FermiCloud,
- Automatically "shelve" the "idle" VMs,
- Automatically launch "worker node" VMs,
- "worker node" VMs join existing Grid cluster and contribute their resources to the Grid.
- "Shelve" the "worker node" VMs when appropriate.

AKA - The "nights and weekend" plan for increased Grid computing capacity.

FermiCloud – Cloud Bursting

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.

Work performed by Dr. Seo-Young Noh (KISTI).

- Refer to his ISGC talk on Friday 2-Mar-2012 for more details:
- http://indico3.twgrid.org/indico/contributionDisplay.py?contribId=1&confId=44

FermiCloud – Infiniband & MPI

To enable HPC stakeholders, the FermiCloud hardware specifications included the Mellanox SysConnect II Infiniband card that was claimed by Mellanox to support virtualization with the Infiniband SRIOV driver.

Unfortunately, despite promises from Mellanox prior to the purchase, we were unable to take delivery of the Infiniband SRIOV driver while working through the standard sales support channels.

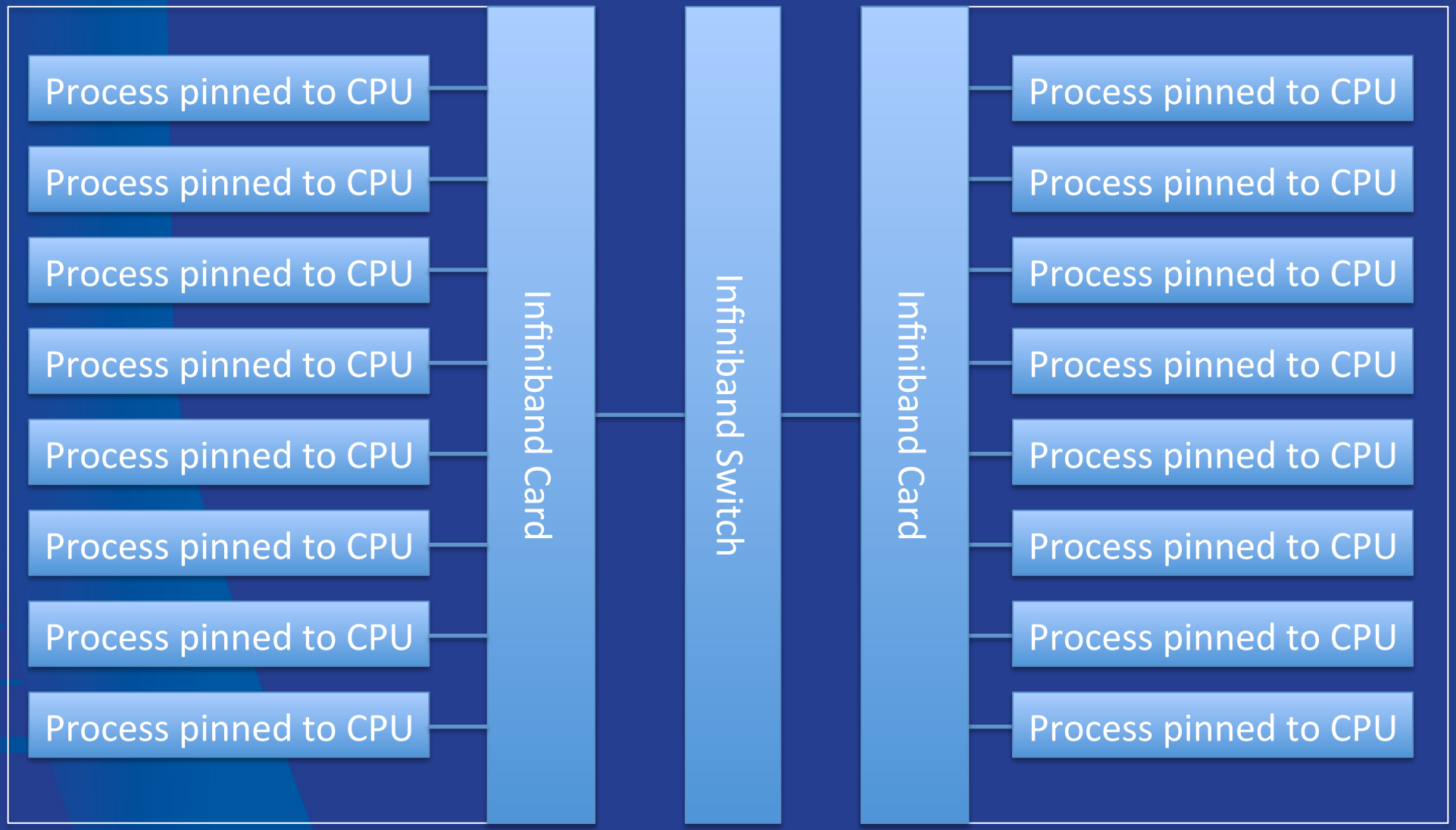
While at SuperComputing 2011 in Seattle in November 2011, Steve Timm, Amitoj Singh and I met with the Mellanox engineers and were able to make arrangements to receive the Infiniband SRIOV driver.

The Infiniband SRIOV driver was delivered to us in December 2011, and using this driver, we were able to make measurements comparing MPI on "bare metal" to MPI on KVM virtual machines on the identical hardware.

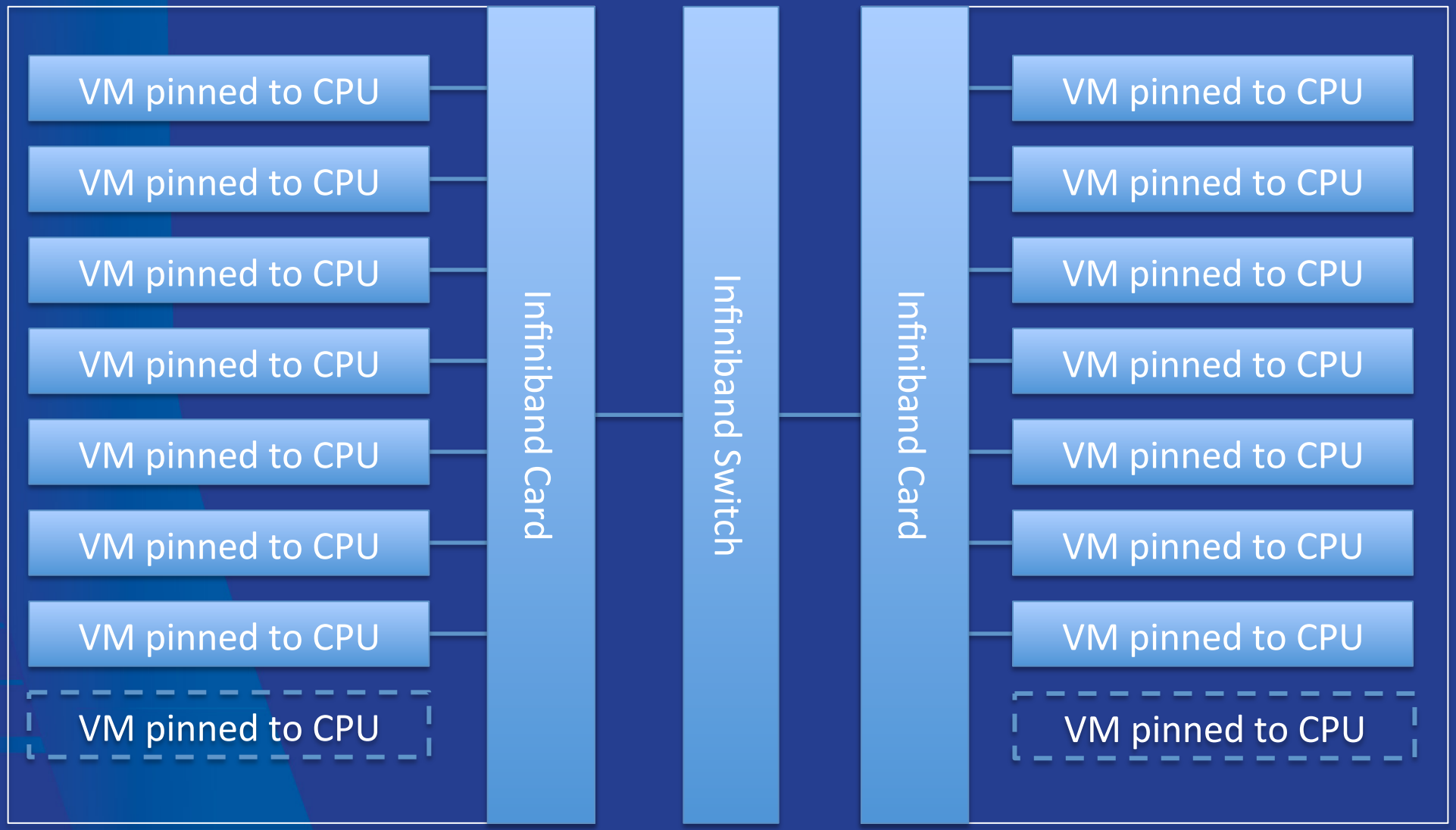
- The next three slides will detail the configurations and measurements.

This allows a direct measurement of the MPI "virtualization overhead".

FermiCloud "Bare Metal MPI"



FermiCloud "Virtual MPI"



MPI on FermiCloud (Note 1)

Configuration	#Host Systems	#VM/host	#CPU	Total Physical CPU	HPL Benchmark (Gflops)	Gflops/Core
Bare Metal without pinning	2	--	8	16	13.9	0.87
Bare Metal with pinning (Note 2)	2	--	8	16	24.5	1.53
VM without pinning (Notes 2,3)	2	8	1 vCPU	16	8.2	0.51
VM with pinning (Notes 2,3)	2	8	1 vCPU	16	17.5	1.09
VM+SRIOV with pinning (Notes 2,4)	2	7	2 vCPU	14	23.6	1.69

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.

Current Stakeholders

Grid & Cloud Computing Personnel,

Run II – CDF & D0,

Intensity Frontier Experiments,

Cosmic Frontier (JDEM/WFIRST,),

Korean Institute for Science & Technology Investigation (KISTI),

Open Science Grid (OSG) software refactoring from pacman to RPM based distribution.

FermiCloud Efforts Today?

- Production deployment of OpenNebula v3.2,
  - Will shortly follow with ONE v3.6 (we will skip ONE v3.4).
- Monitoring, including Idle VM detection and management,
- FermiCloud-HA design and deployment,
- We are using FermiCloud resources to build, test and integrate the new RPM based OSG software stack,
  - Including stress testing of the software configurations at production scale.
- Gearing up to use FermiCloud with the soon to be deployed 100 Gb/s network testbed:
  - We will start by using 100 Gb/s TX cards in our integration infrastructure, and may purchase some new FermiCloud servers.

FermiCloud Summary - 1

The 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 implemented both monitoring and accounting by extension of existing tools.

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!
- The **2 fewer** physical CPUs is a limitation of the current Infiniband SRIOV driver, if we can get this fixed then the virtualization results are likely to achieve full parity with the bare metal results.
- We will be taking additional measurements of virtualized MPI performance in the near term!

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.

FermiCloud Summary - 2

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 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".

None of this 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.

Job Opening

The Grid and Cloud Computing Department has been approved for a new personnel opening to work with the FermiGrid and FermiCloud projects:

- Cloud and Grid Administrator (Computer Services Specialist II)

The opening will be posted on the "Jobs at Fermilab" web site shortly:

- https://fermi.hodesiq.com/

Thank You!

Any Questions?

Extra Slides

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),

48 GBytes of memory (we started with 24 & upgraded 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/Spring 2012),

2U SuperMicro chassis with redundant power supplies

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 – VM Format

Virtual machine images are stored in a way that they can be exported as a device:

- The OS partition contains full contents of the / partition plus a boot sector and a partition table.
- Not compressed.

Kernel and initrd are stored internally to the image,

- Different from Amazon and Eucalyptus,

Note that it is possible to have Xen and KVM kernels loaded in the same VM image and run it under either hypervisor.

Secrets are not stored in the image,

- See slides on Authentication/Contextualization.

We are currently investigating the CERN method of launching multiple copies of same VM using LVM qcow2 (quick copy on write) but this is not our major use case at this time.

We will likely invest in LanTorrent/LVM for booting multiple "worker node" virtual machines simultaneously at a later date.

Comments on Cost Comparison

The FermiCloud "Unit" (CPU+2GB) without HyperThreading is approximately two Amazon EC2 compute units.

Amazon can change their pricing model any time.

The Amazon EC2 prices do not include the costs for data movement, FermiCloud does not charge for data movement. Since the typical HEP experiment moves substantial amounts of data, the Amazon data movement changes will be significant.

The prices for FermiCloud do not include costs for the infrastructure (building/computer room/environmental) and the costs for operation (electricity).

System administrator costs factor out of the comparison, since they apply equally to both sides of the comparison [FermiCloud / Amazon].

- Our expectation/hope is that with the puppet & cobbler deployment, the VM system administrator costs will decrease.

Amazon Data Movement Cost Range (USD)

Annual Data Movement (TB)	Data In (TB)	Data Out (TB)	Estimated Annual Cost
10	4	6	$1,331
25	10	15	$3,328
50	20	30	$6,656
100	40	60	$13,312
250	100	150	$24,064
500	200	300	$48,128
1,000	400	600	$96,256
2,000	800	1,200	$180,224
5,000	2,000	3,000	$450,560

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://fcl001k1.fnal.gov/ganglia/
- http://fcl002k1.fnal.gov/ganglia/

Preliminary RSV based monitoring pilot:

- http://fermicloudrsv.fnal.gov/rsv

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.

Virtualized Storage Service Investigation

Motivation:

General purpose systems from various vendors being used as file servers,

Systems can have many more cores than needed to perform the file service,

- Cores go unused => Inefficient power, space and cooling usage,
- Custom configurations => Complicates sparing issues.

Question:

Can virtualization help here?

What (if any) is the virtualization penalty?

Virtualized Storage Server Test Procedure

Evaluation:

Use IOzone and real physics root based analysis code.

Phase 1:

Install candidate filesystem on "bare metal" server,

Evaluate performance using combination of bare metal and virtualized clients (varying the number),

Also run client processes on the "bare metal" server,

Determine "bare metal" filesystem performance.

Phase 2:

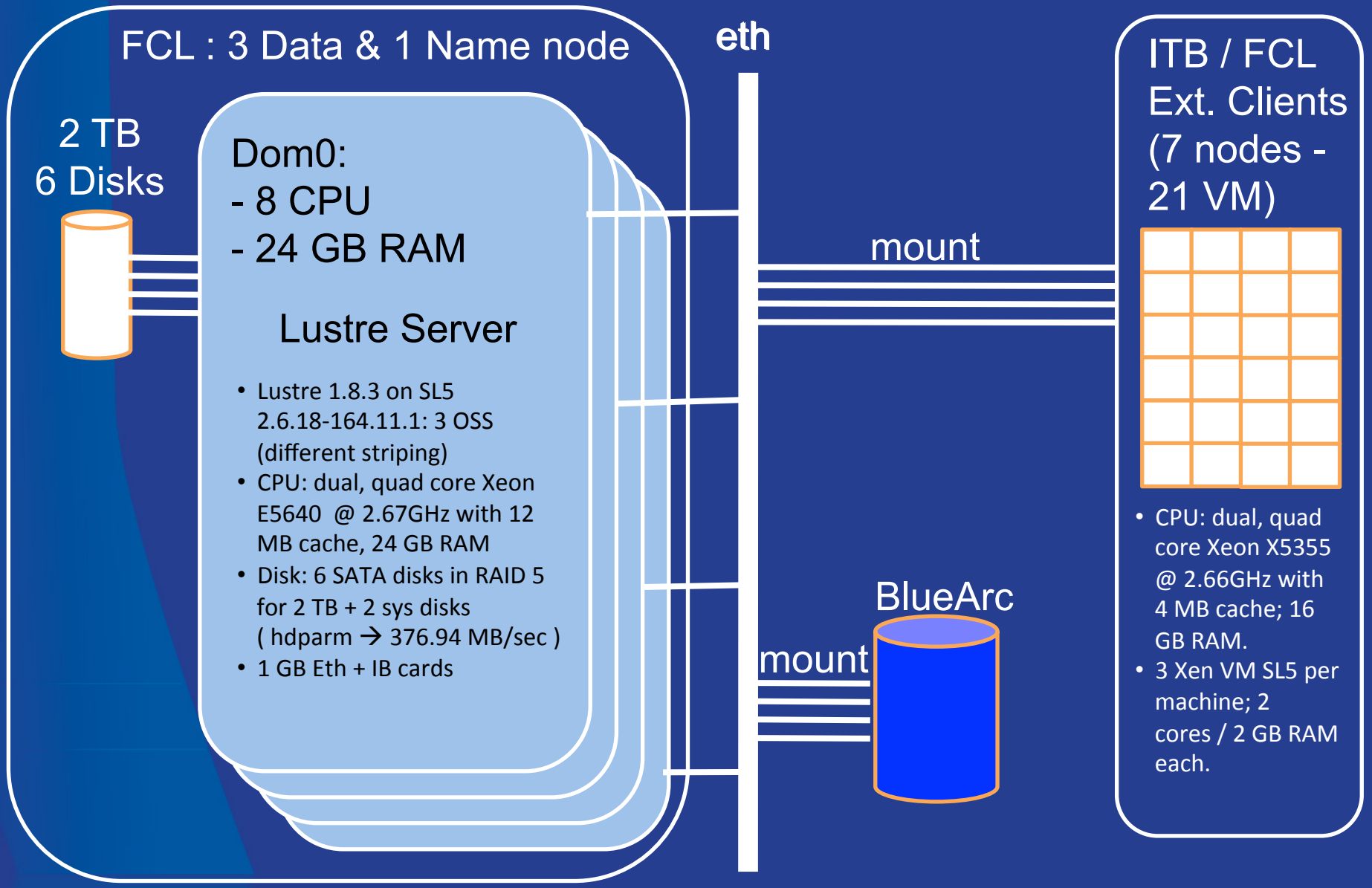
Install candidate filesystem on a virtual machine server,

Evaluate performance using combination of bare metal and virtualized clients (varying the number),

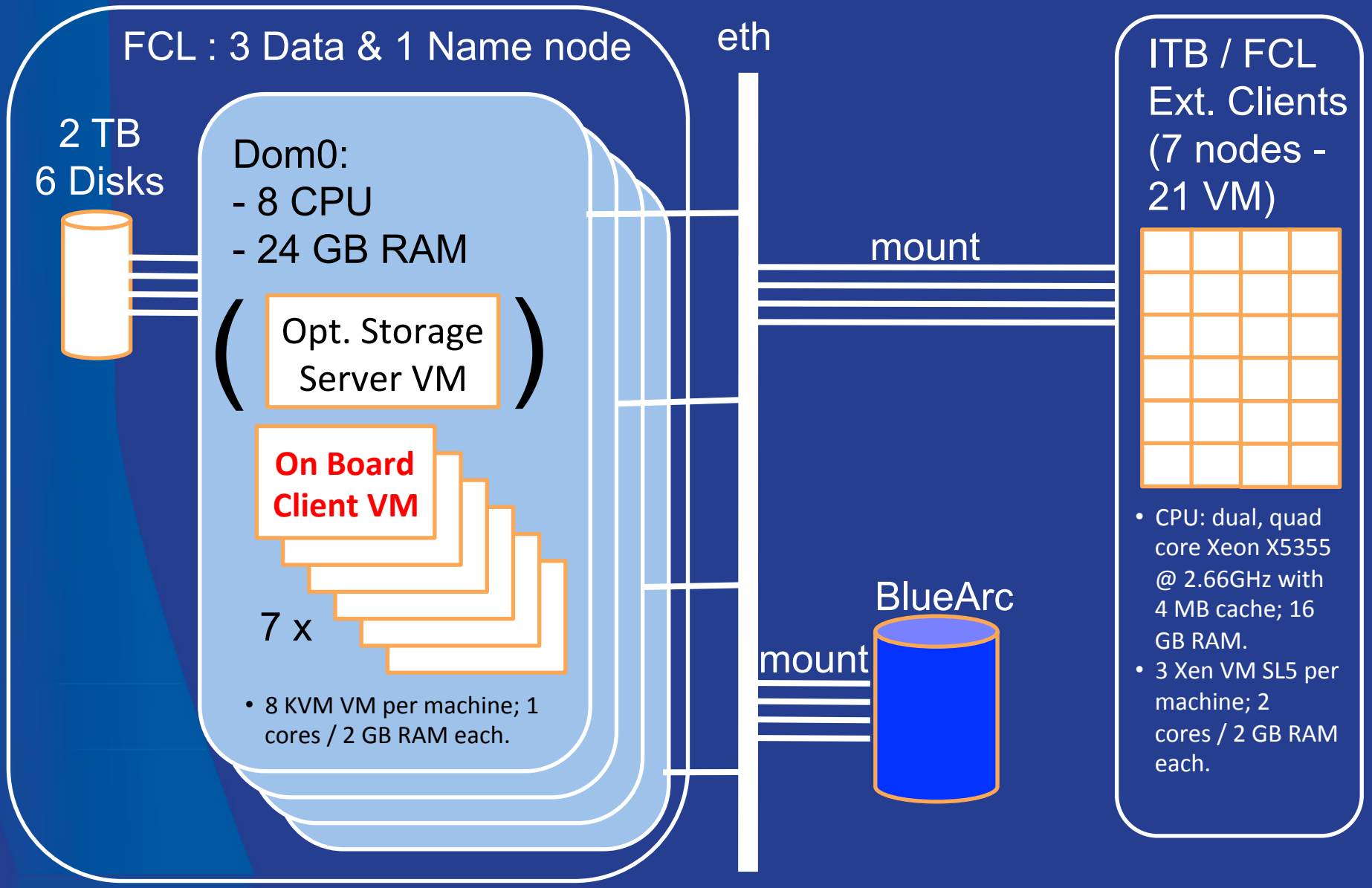
Also use client virtual machines hosted on same physical machine as the virtual machine server,

Determine virtual machine filesystem performance.

FermiCloud Test Bed - "Bare Metal" Server



FermiCloud Test Bed - Virtualized Server



Virtualized File Service Results Summary

FileSystem	Benchmark	Read (MB/s)	"Bare Metal" Write (MB/s)	VM Write (MB/s)	Notes
Lustre	IOZone	350	250	70	Significant write penalty when FS on VM
	Root-based	12.6	-	-	
Hadoop	IOZone	50 - 240	80 - 300	80 - 300	Varies on number of replicas, fuse does not export a full posix fs.
	Root-based	7.9	-	-	
OrangeFS	IOZone	150 - 330	220 - 350	220 - 350	Varies on number of name nodes
	Root-based	8.1	-	-	
BlueArc	IOZone	300	330	n/a	Varies on system conditions
	Root-based	8.4	-	-	

contribId=32&sessionId=36&resId=0&materialId=slides&confId=44

FermiCloud – Running "External" VMs

We participate in:

- The HEPiX virtualization working group led by Tony Cass (CERN),
- The "Security for Collaborating Infrastructures" (SCI) group led by Dave Kelsey (RAL).

That being said…

- It is our intention that FermiCloud (and likely FermiGrid at some future date) will support the submission of VMs for running on FermiCloud or FermiGrid resources via standard Cloud/Grid mechanisms. Such as Amazon S3, OCCI, globus-url-copy (GridFTP) or globus-job-run in addition to direct OpenNebula console access via x509 authentication.
- As part of our security infrastructure we will likely reuse the existing "network jail" to treat new untrusted VMs similarly to how Fermilab treats new untrusted laptops on the site network.
- If there is any issue with the VM, we will directly contact the people who:
  1. Transferred the VM to Fermilab (*yes, we do keep logs as well as monitor the actions of the VM on the network*).
  2. Launched the VM on FermiCloud and/or FermiGrid (*again – yes, we do keep logs as well as monitor the actions of the VM on the network*).
- If we don't get satisfactory answers from both, then both of the DNs will likely wind up on the site "blacklist" infrastructure.