Future Grid Tutorial

Presented at CCGrid2011

by

Gregor von Laszewski,
Andrew Younge, Paul Marshall

Contact: laszewski@gmail.com

Help: help@futuregrid.org

An up to date version of this tutorial is available at

http://futuregrid.svn.sourceforge.net/viewvc/futuregrid/presentations/tutorial-half-day/ccgrid2011-tutorial.pdf

Acknowledgment: People

- Many people have worked on FuturGrid and we will not be able to list all them here.
- We will attempt to keep a list available on the portal Web site.
- Many others have contributed to this tutorial!!
    - Thanks!!

Acknowledgement

- The FutureGrid project is funded by the National Science Foundation (NSF) and is led by Indiana University with University of Chicago, University of Florida, San Diego Supercomputing Center, Texas Advanced Computing Center, University of Virginia, University of Tennessee, University of Southern California, Dresden, Purdue University, and Grid 5000 as partner sites.

Reuse of slides

- If you reuse the slides please indicate that they are copied from this tutorial. Include a link to https://portal.futuregrid.org
- We discourage the printing of the tutorial material due to two reasons:
  - We like to make sure the impact on the environment due to use of paper and ink is minimal
  - We intend to keep the tutorials up to date on the Web site at https://portal.futuregrid.org

Technology Previews

- Some material presented here is not available to the general user community and is potentially still under development. We show however some technology previews in order to provide you with some exciting new features that we are currently working on. Slides referring to the reviews are marked with the following icon: Technology Preview

Outline

- **Getting Access**
- **Overview of FutureGrid**
- **Future Grid Services**
  - **HPC/MPI on FutureGrid**
  - **Eucalyptus on FutureGrid**
  - **Nimbus on FutureGrid**
  - *Appliances on FutureGrid*
  - *Unicore*
  - *Genesis II*
- **Rain on FutureGrid**
  - **Image Generation**
  - **Image Deployment**
- *In Future*
  - *Pegasus*
  - *Hadoop*
  - *OpenStack*
  - *OpenNebula*

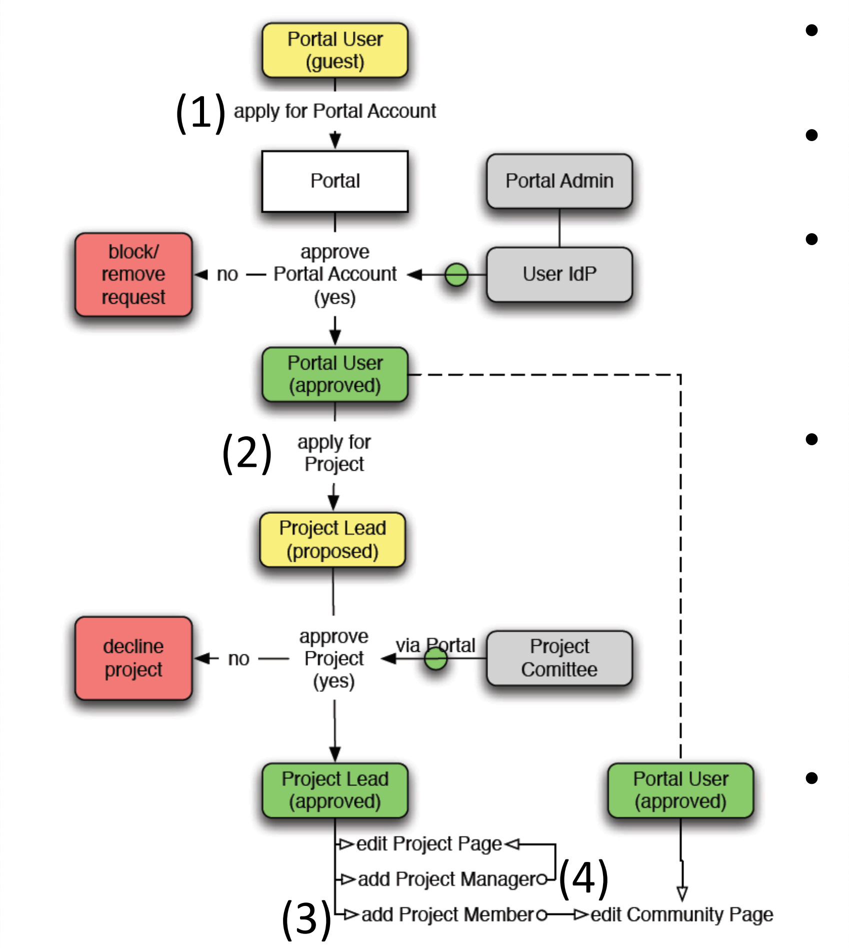
Getting Access to FutureGrid

Gregor von Laszewski

Portal Account, Projects, and System Accounts

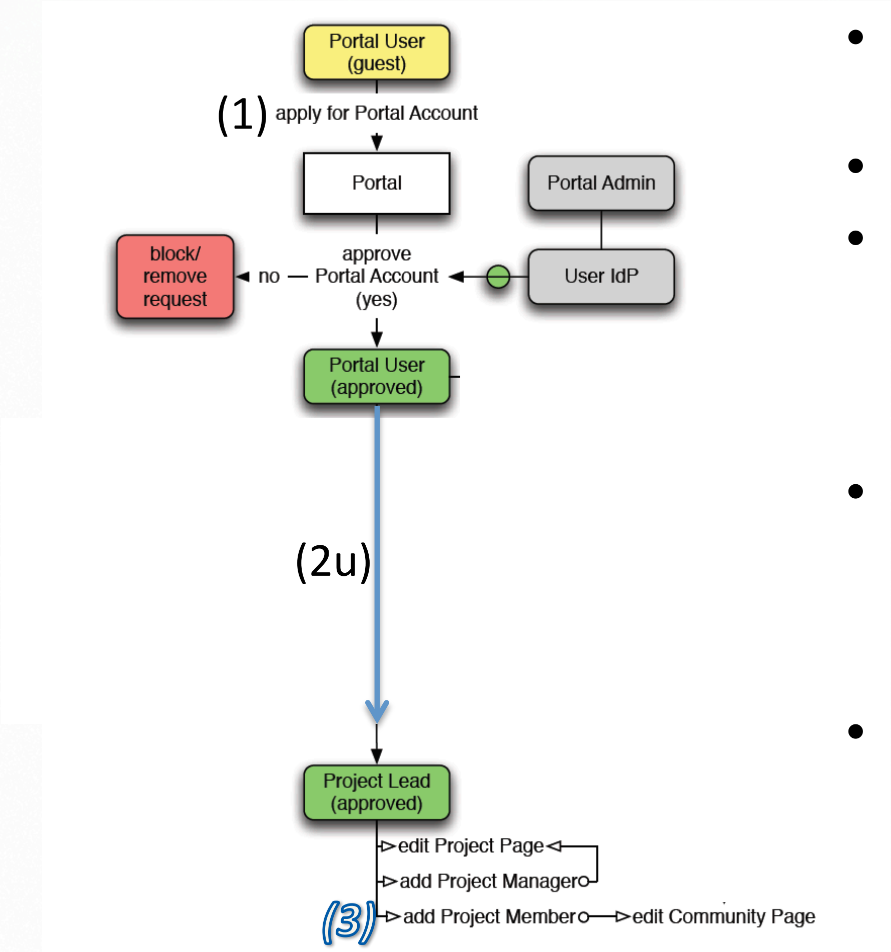
- The main entry point to get access to the systems and services is the FutureGrid Portal.
- We distinguish the portal account from system and service accounts.
  - You may have multiple system accounts and may have to apply for them separately, e.g. Eucalyptus, Nimbus
  - Why several accounts:
    - Some services may not be important for you, so you will not need an account for all of them.
      - In future we may change this and have only one application step for all system services.
    - Some services may not be easily integratable in a general authentication framework

The Process: A new Project



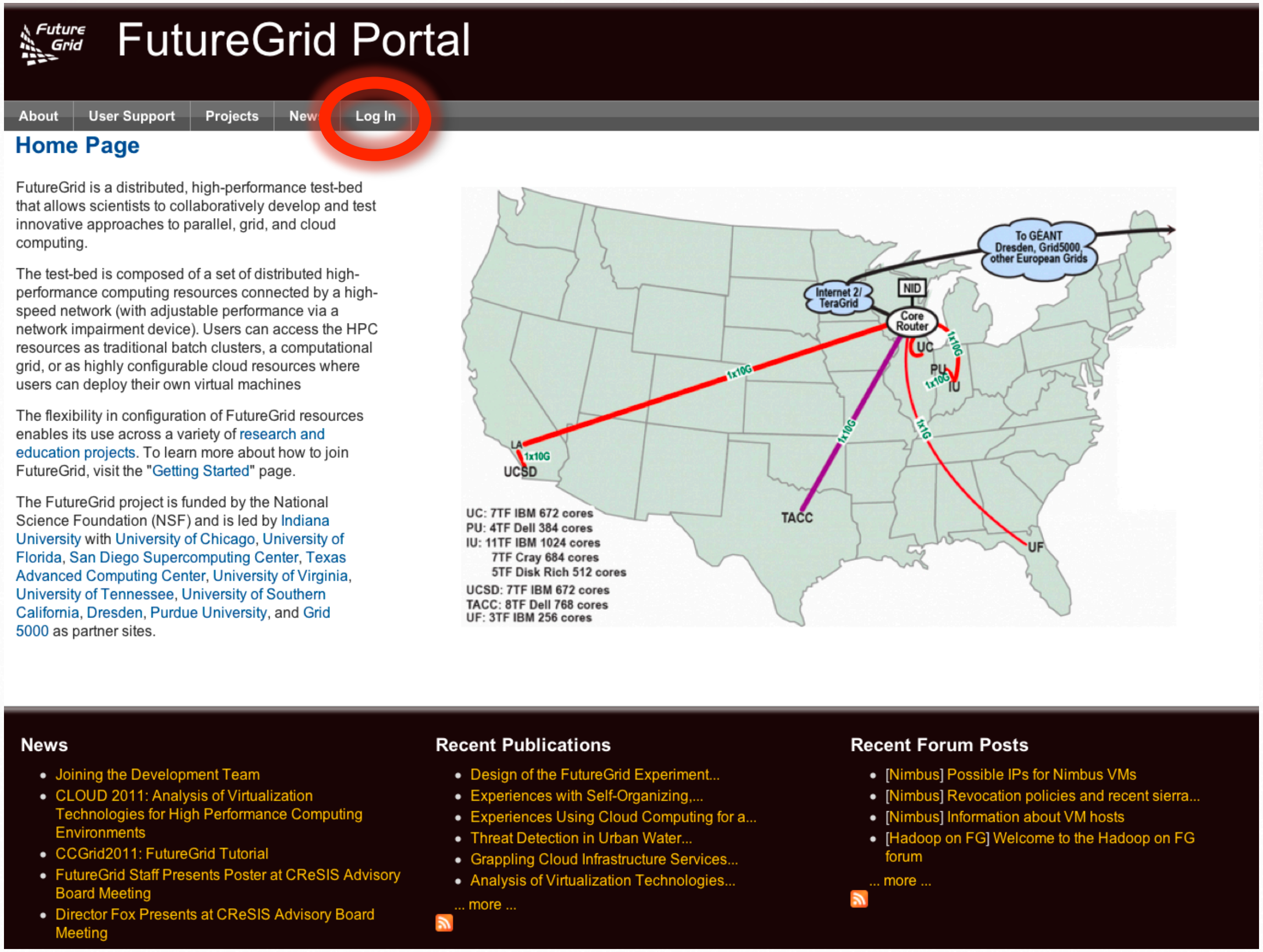
- **(1) get a portal account**
  - *portal account is approved*
- **(2) propose a project**
  - *project is approved*
- **(3) ask your partners for their portal account names and add them to your projects as members**
  - *No further approval needed*
- **(4) if you need an additional person being able to add members designate him as project manager (currently there can only be one).**
  - *No further approval needed*
- **You are in charge who is added or not!**
  - Similar model as in Web 2.0 Cloud services, e.g. sourceforge

The Process: Join A Project



- **(1) get a portal account**
  - *portal account is approved*
- **Skip steps (2) – (4)**
- **(2u) Communicate with your project lead which project to join and give him your portal account name**
- *Next step done by project lead*
  - *(3) The project lead will add you to the project*
- **You are responsible to make sure the project lead adds you!**
  - Similar model as in Web 2.0 Cloud services, e.g. sourceforge

Apply for a Portal Account



FutureGrid Portal

Home Page

FutureGrid is a distributed, high-performance test-bed that allows scientists to collaboratively develop and test innovative approaches to parallel, grid, and cloud computing.

The test-bed is composed of a set of distributed high-performance computing resources connected by a high-speed network (with adjustable performance via a network impairment device). Users can access the HPC resources as traditional batch clusters, a computational grid, or as highly configurable cloud resources where users can deploy their own virtual machines

The flexibility in configuration of FutureGrid resources enables its use across a variety of research and education projects. To learn more about how to join FutureGrid, visit the "Getting Started" page.

The FutureGrid project is funded by the National Science Foundation (NSF) and is led by Indiana University with University of Chicago, University of Florida, San Diego Supercomputing Center, Texas Advanced Computing Center, University of Virginia, University of Tennessee, University of Southern California, Dresden, Purdue University, and Grid 5000 as partner sites.

News

- Joining the Development Team
- CLOUD 2011: Analysis of Virtualization Technologies for High Performance Computing Environments
- CCGrid2011: FutureGrid Tutorial
- FutureGrid Staff Presents Poster at CReSIS Advisory Board Meeting
- Director Fox Presents at CReSIS Advisory Board Meeting

Recent Publications

- Design of the FutureGrid Experiment...
- Experiences with Self-Organizing,...
- Experiences Using Cloud Computing for a...
- Threat Detection in Urban Water...
- Grappling Cloud Infrastructure Services...
- Analysis of Virtualization Technologies...

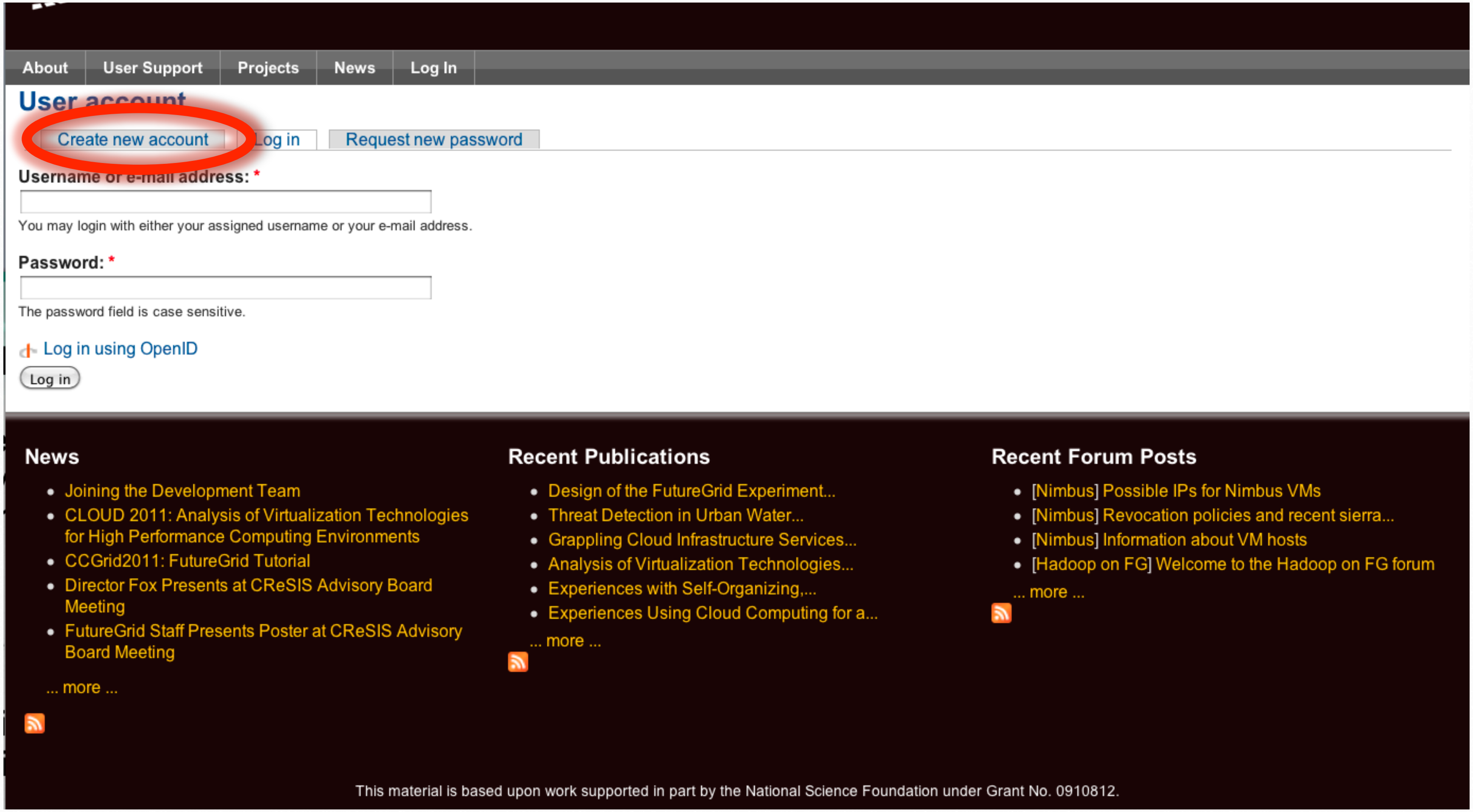
... more ...

Recent Forum Posts

- [Nimbus] Possible IPs for Nimbus VMs
- [Nimbus] Revocation policies and recent sierra...
- [Nimbus] Information about VM hosts
- [Hadoop on FG] Welcome to the Hadoop on FG forum

... more ...

Apply for a Portal Account



User account

Username or e-mail address: *

You may login with either your assigned username or your e-mail address.

Password: *

The password field is case sensitive.

Log in using OpenID

News

- Joining the Development Team
- CLOUD 2011: Analysis of Virtualization Technologies for High Performance Computing Environments
- CCGrid2011: FutureGrid Tutorial
- Director Fox Presents at CReSIS Advisory Board Meeting
- FutureGrid Staff Presents Poster at CReSIS Advisory Board Meeting

... more ...

Recent Publications

- Design of the FutureGrid Experiment...
- Threat Detection in Urban Water...
- Grappling Cloud Infrastructure Services...
- Analysis of Virtualization Technologies...
- Experiences with Self-Organizing,...
- Experiences Using Cloud Computing for a...

... more ...

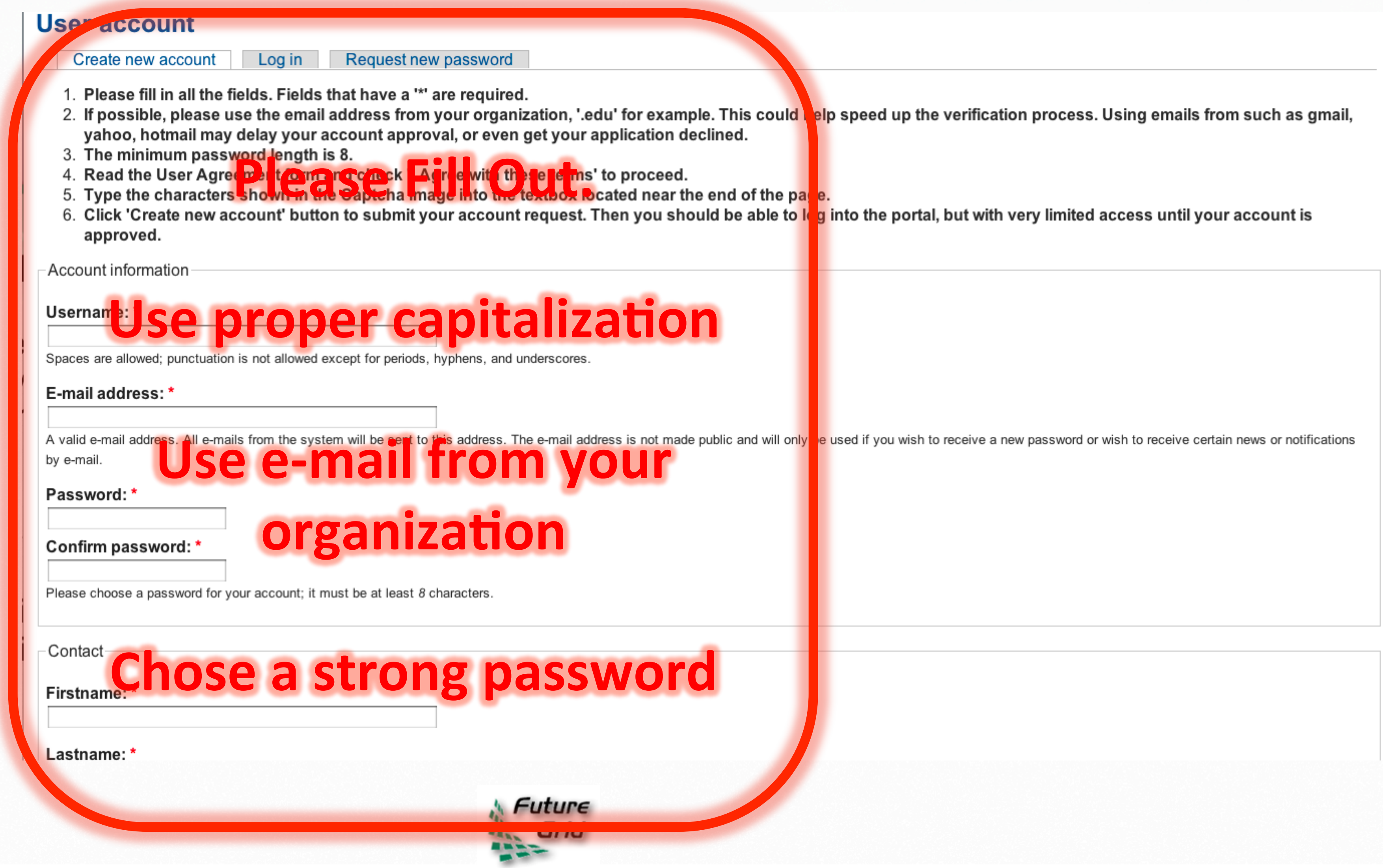
Recent Forum Posts

- [Nimbus] Possible IPs for Nimbus VMs
- [Nimbus] Revocation policies and recent sierra...
- [Nimbus] Information about VM hosts
- [Hadoop on FG] Welcome to the Hadoop on FG forum

... more ...

This material is based upon work supported in part by the National Science Foundation under Grant No. 0910812.

Apply for a Portal Account



User account

1. Please fill in all the fields. Fields that have a '*' are required.
2. If possible, please use the email address from your organization, '.edu' for example. This could help speed up the verification process. Using emails from such as gmail, yahoo, hotmail may delay your account approval, or even get your application declined.
3. The minimum password length is 8.
4. Read the User Agreement form and check 'Agree with these terms' to proceed.
5. Type the characters shown in the Captcha image into the textbox located near the end of the page.
6. Click 'Create new account' button to submit your account request. Then you should be able to log into the portal, but with very limited access until your account is approved.

Account information

Username: *

Spaces are allowed; punctuation is not allowed except for periods, hyphens, and underscores.

E-mail address: *

A valid e-mail address. All e-mails from the system will be sent to this address. The e-mail address is not made public and will only be used if you wish to receive a new password or wish to receive certain news or notifications by e-mail.

Password: *

Confirm password: *

Please choose a password for your account; it must be at least *8* characters.

Contact

Firstname: *

Lastname: *

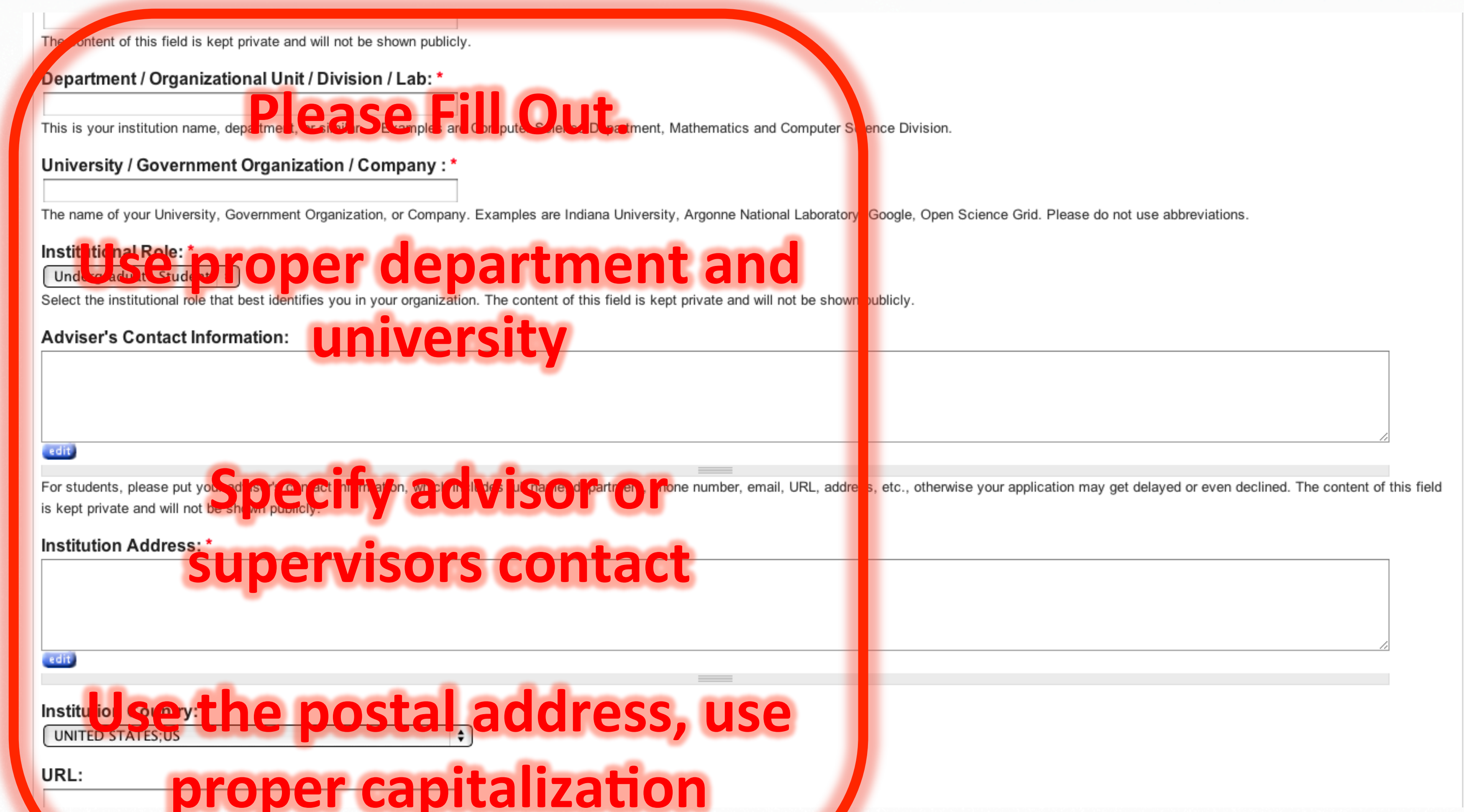
Please Fill Out.

Use proper capitalization

Use e-mail from your organization

Chose a strong password

Apply for a Portal Account



The content of this field is kept private and will not be shown publicly.

Department / Organizational Unit / Division / Lab: *

This is your institution name, department, or similar. Examples are Computer Science Department, Mathematics and Computer Science Division.

University / Government Organization / Company : *

The name of your University, Government Organization, or Company. Examples are Indiana University, Argonne National Laboratory, Google, Open Science Grid. Please do not use abbreviations.

Institutional Role: *

Undergraduate Student

Select the institutional role that best identifies you in your organization. The content of this field is kept private and will not be shown publicly.

Adviser's Contact Information:

For students, please put your adviser's contact information, which includes name, department, phone number, email, URL, address, etc., otherwise your application may get delayed or even declined. The content of this field is kept private and will not be shown publicly.

Institution Address: *

Institution Country:

UNITED STATES;US

URL:

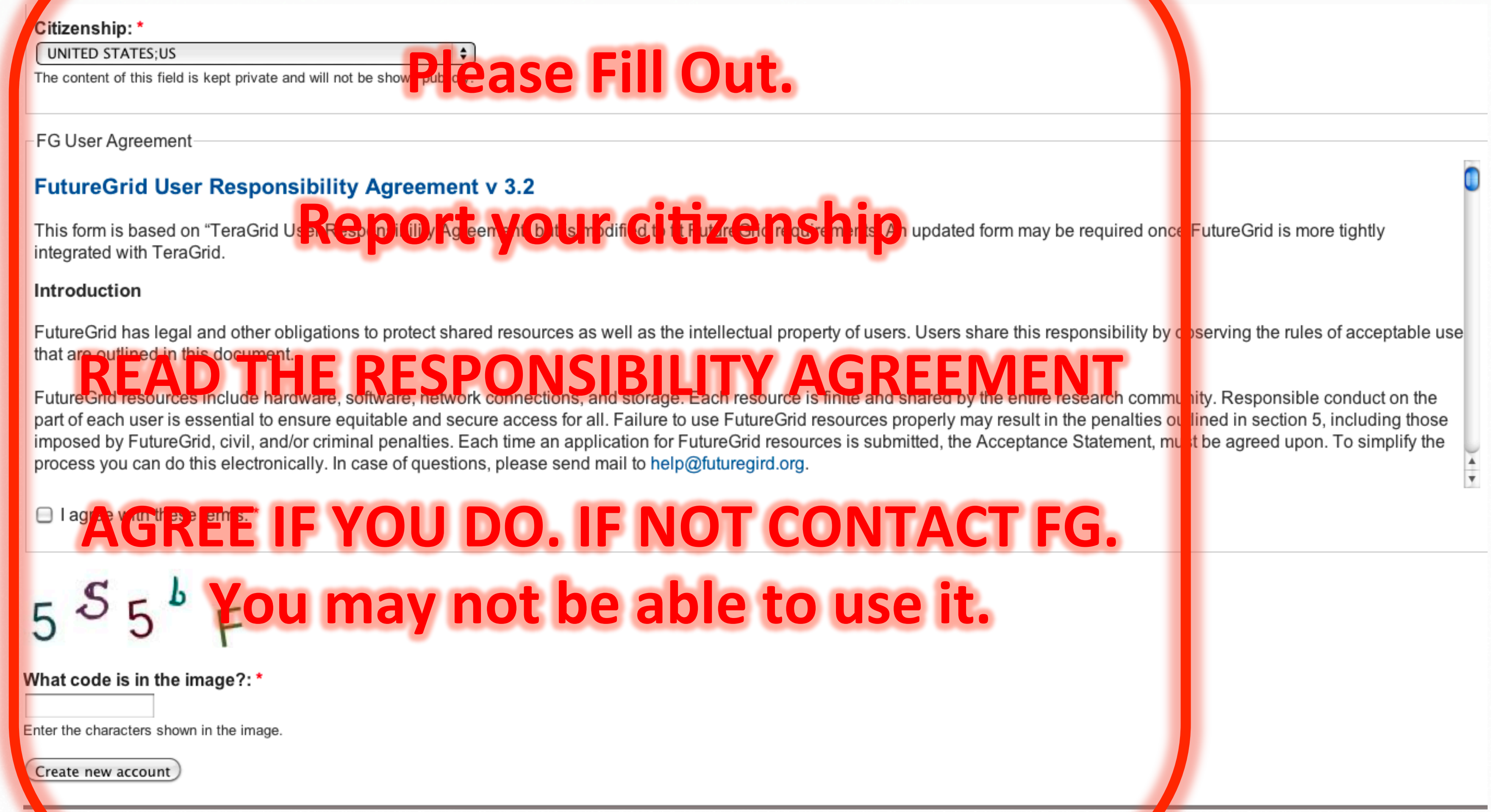
Please Fill Out.

Use proper department and university

Specify advisor or supervisors contact

Use the postal address, use proper capitalization

Apply for a Portal Account



Citizenship: *

UNITED STATES;US

The content of this field is kept private and will not be shown publicly.

Please Fill Out.

FG User Agreement

FutureGrid User Responsibility Agreement v 3.2

Report your citizenship

This form is based on “TeraGrid User Responsibility Agreement” but is modified to fit FutureGrid requirements. An updated form may be required once FutureGrid is more tightly integrated with TeraGrid.

Introduction

FutureGrid has legal and other obligations to protect shared resources as well as the intellectual property of users. Users share this responsibility by observing the rules of acceptable use that are outlined in this document.

READ THE RESPONSIBILITY AGREEMENT

FutureGrid resources include hardware, software, network connections, and storage. Each resource is finite and shared by the entire research community. Responsible conduct on the part of each user is essential to ensure equitable and secure access for all. Failure to use FutureGrid resources properly may result in the penalties outlined in section 5, including those imposed by FutureGrid, civil, and/or criminal penalties. Each time an application for FutureGrid resources is submitted, the Acceptance Statement, must be agreed upon. To simplify the process you can do this electronically. In case of questions, please send mail to help@futuregird.org.

□ I agree with these terms. *

AGREE IF YOU DO. IF NOT CONTACT FG.

You may not be able to use it.

What code is in the image?: *

Enter the characters shown in the image.

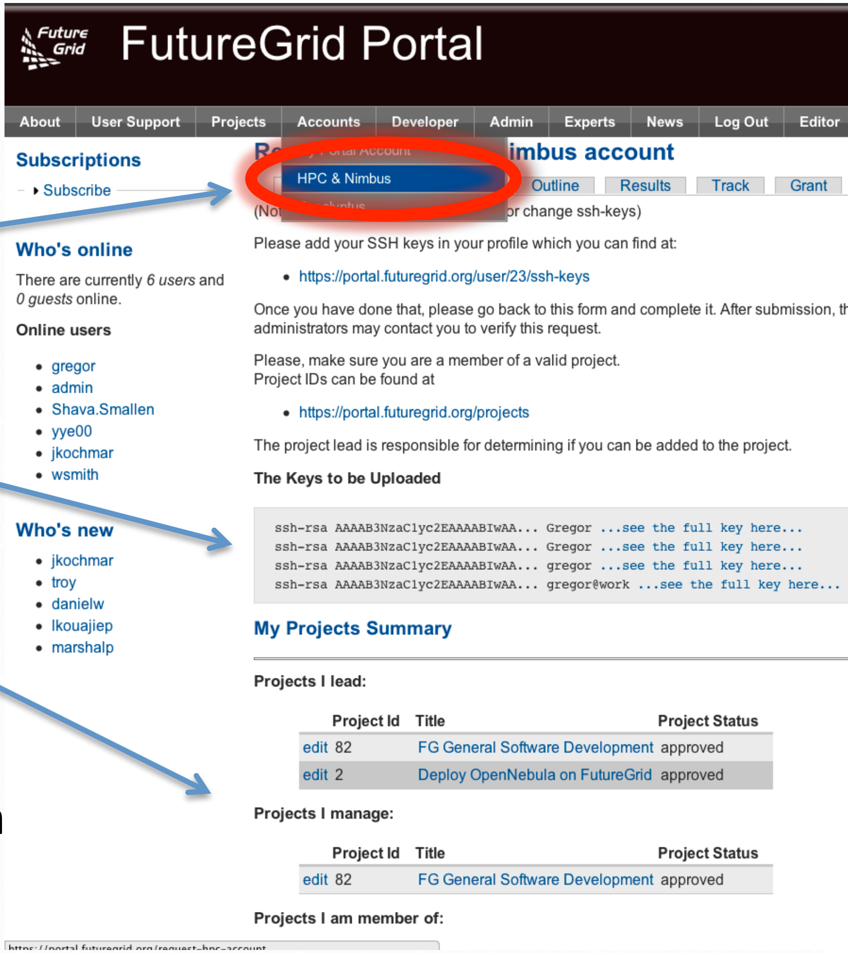
Wait

- Wait till you get notified that you have a portal account.

- Now you have a portal account (cont.)

Apply for an HPC and Nimbus account

- Login into the portal
- Simple go to
  - Accounts-> HPC&Nimbus
- (1) add you ssh keys
- (3) make sure you are in a valid project
- (2) wait for 24 business hours
  - (for tutorial users we accelerate)
  - No accounts will be granted between Friday 5pm EST – Monday 9 am EST



Re imbus account

(No or change ssh-keys)

Please add your SSH keys in your profile which you can find at:

- https://portal.futuregrid.org/user/23/ssh-keys

Once you have done that, please go back to this form and complete it. After submission, th administrators may contact you to verify this request.

Please, make sure you are a member of a valid project.
Project IDs can be found at

- https://portal.futuregrid.org/projects

The project lead is responsible for determining if you can be added to the project.

The Keys to be Uploaded

```
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAA... Gregor ...see the full key here...
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAA... Gregor ...see the full key here...
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAA... gregor ...see the full key here...
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAA... gregor@work ...see the full key here...
```

Who's online

There are currently *6 users* and *0 guests* online.

Online users

- gregor
- admin
- Shava.Smallen
- yye00
- jkochmar
- wsmith

Who's new

- jkochmar
- troy
- danielw
- lkouajiep
- marshalp

My Projects Summary

Projects I lead:

	Project Id	Title	Project Status
edit	82	FG General Software Development	approved
edit	2	Deploy OpenNebula on FutureGrid	approved

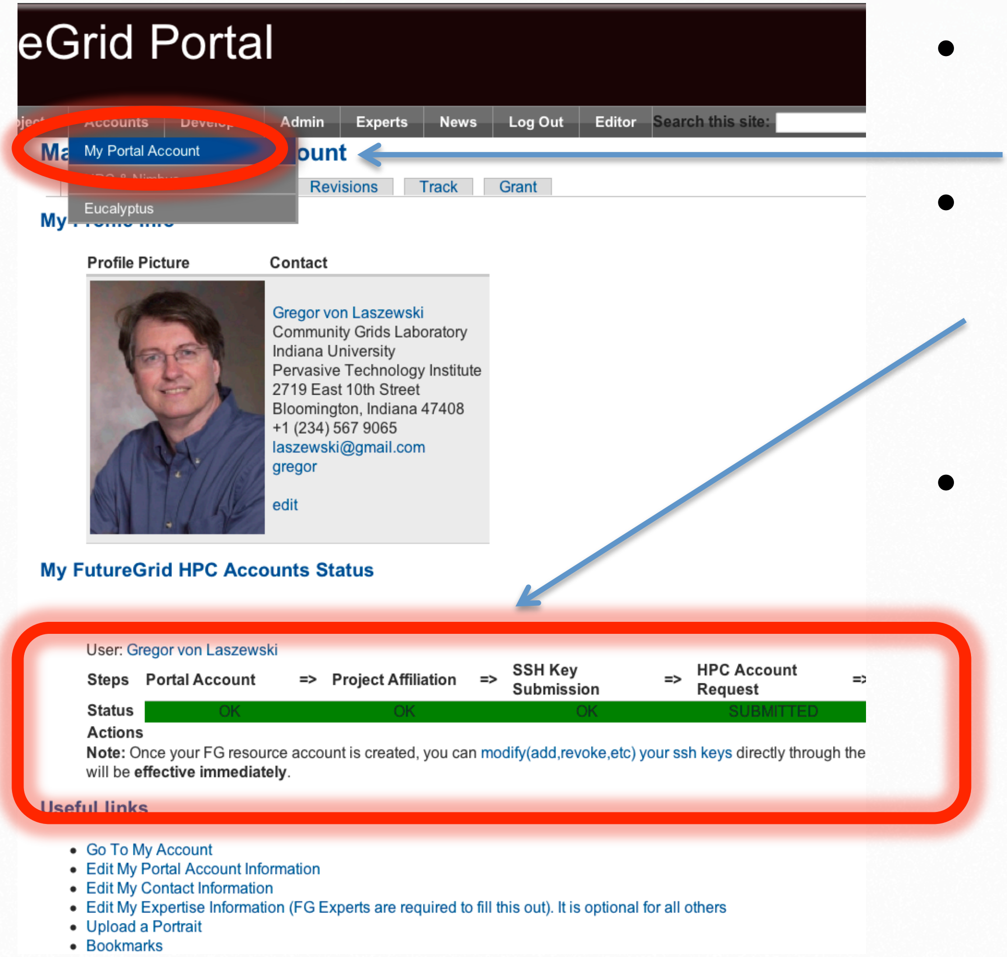
Projects I manage:

	Project Id	Title	Project Status
edit	82	FG General Software Development	approved

Projects I am member of:

https://portal.futuregrid.org/request-hpc-account

Check your Account Status



My Profile Info

Profile Picture

Contact

Gregor von Laszewski
Community Grids Laboratory
Indiana University
Pervasive Technology Institute
2719 East 10th Street
Bloomington, Indiana 47408
+1 (234) 567 9065
laszewski@gmail.com
gregor

edit

My FutureGrid HPC Accounts Status

User: Gregor von Laszewski

Steps	Portal Account	=>	Project Affiliation	=>	SSH Key Submission	=>	HPC Account Request	=>
Status	OK		OK		OK		SUBMITTED	
Actions								

Note: Once your FG resource account is created, you can modify(add,revoke,etc) your ssh keys directly through the will be **effective immediately**.

Useful links

- Go To My Account
- Edit My Portal Account Information
- Edit My Contact Information
- Edit My Expertise Information (FG Experts are required to fill this out). It is optional for all others
- Upload a Portrait
- Bookmarks

- Goto:
  - Accounts-My Portal Account
- Check if the account status bar is green
  - Errors wil indicate an issue or a task that requires waiting
- Since you are already here:
  - Upload a portrait
  - Check if you have other things that need updating
  - Add ssh keys if needed

Wait

- Once you have everything green, you have an HPC and a Nimbus account.
- **PROPAGATION OF THE ACCOUNTS TO NIMBUS CURRENTLY REQUIRES AN ADDITIONAL 30 – 60 minutes**
- **For the impatient please check your Portal account page**

HPC Account creation	Wait time
India	24 hours
Sierra	24 hours + x min
Xray	24 hours + x min
Alamo	24 hours + x min
Hotel	24 hours + x min
Foxtrot	24 hours + x min
Bravo	24 hours + x min

Service creation	Wait time
Eucalyptus	24 hours
Nimbus	HPC account creation + x min

Hours = Business hours!!!!!!!!!

Eucalyptus Account Creation

- Use the Eucalyptus Web Interfaces at

  https://eucalyptus.india.futuregrid.org:8443/

- On the Login page click on Apply for account.
- On the next page that pops up fill out ALL the Mandatory AND optional fields of the form.
- Once complete click on signup and the Eucalyptus administrator will be notified of the account request.
- You will get an email once the account has been approved.
- Click on the link provided in the email to confirm and complete the account creation process.

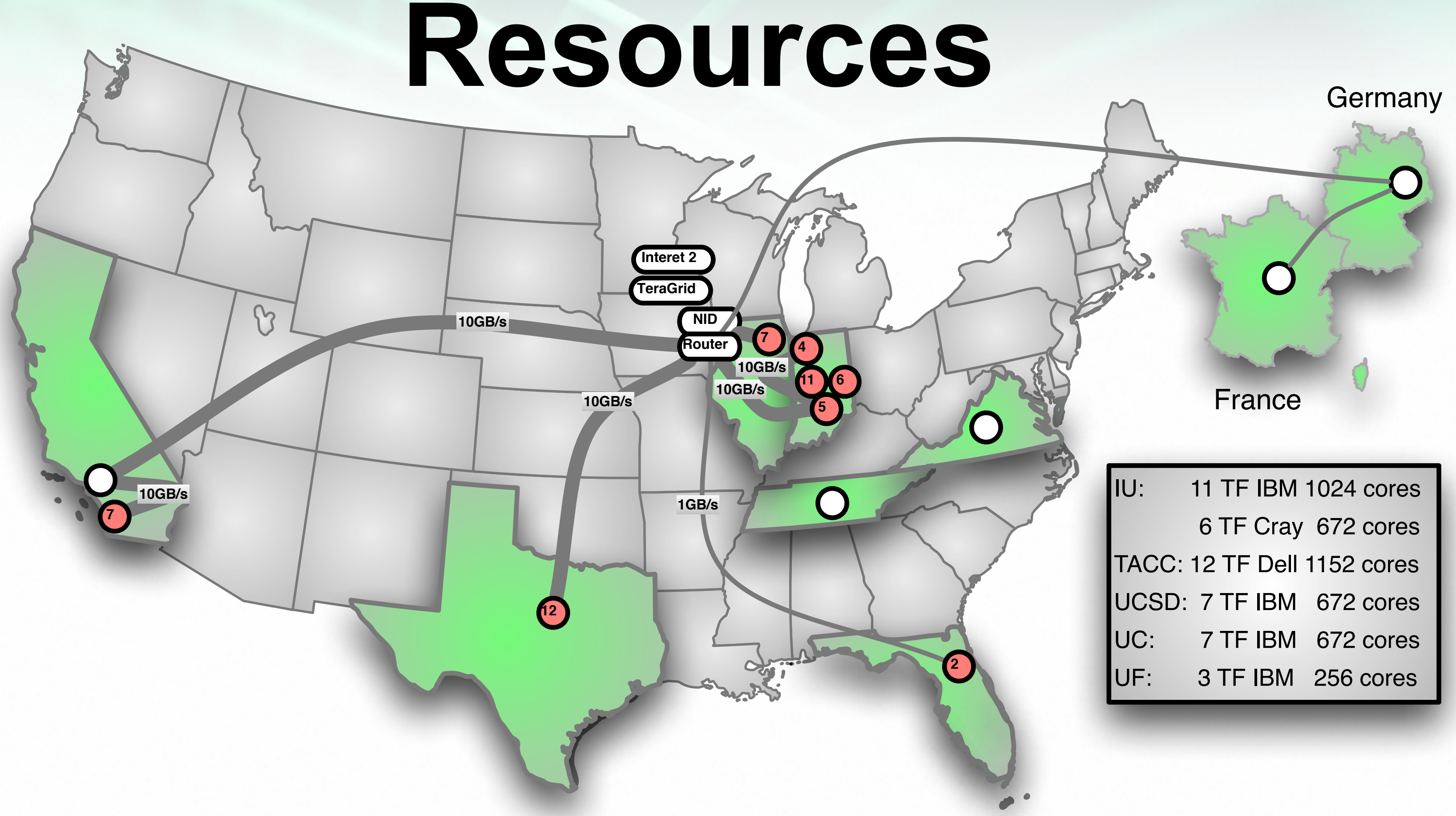
OVERVIEW OF FG

Presented by
Gregor von Laszewski

FutureGrid key Issues

- FutureGrid will provide an experimental testbed with a wide variety of computing services to its users.
- The testbed provides to its users:
    - A rich development and testing platform for middleware and application users allowing comparisons in functionality and performance.
    - A variety of environments, many be instantiated dynamically, on demand. Available resources include, VMs, cloud, grid systems …
    - The ability to reproduce experiments at a later time (an experiment is the basic unit of work on the FutureGrid).
    - A rich education an teaching platform for advanced cyberinfrastructure
    - The ability to collaborate with the US industry on research projects.
- Web Page: www.futuregrid.org
- E-mail: help@futuregrid.org.

FutureGrid Partners and Resources

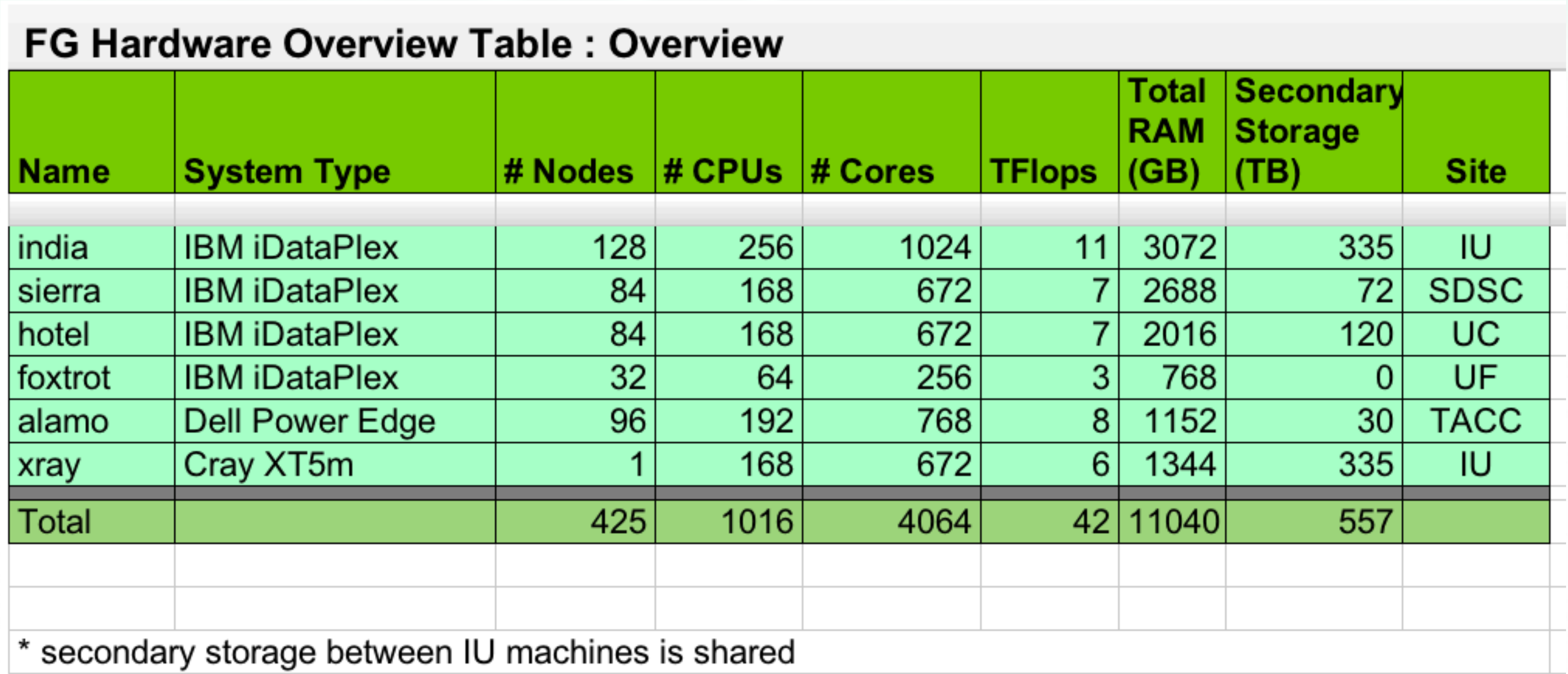


IU:	11 TF	IBM	1024 cores
	6 TF	Cray	672 cores
TACC:	12 TF	Dell	1152 cores
UCSD:	7 TF	IBM	672 cores
UC:	7 TF	IBM	672 cores
UF:	3 TF	IBM	256 cores

- **HW Resources at**: Indiana University, SDSC, UC/ANL, TACC, University of Florida, Purdue,
- **Software Partners:** USC ISI, University of Tennessee Knoxville, University of Virginia, Technische Universtität Dresden
- However, users of FG do not have to be from these partner organizations. Furthermore, we hope that new organizations in academia and industry can partner with the project in the future.

http://futuregrid.org

Current HW Overview



FG Hardware Overview Table : Overview

Name	System Type	# Nodes	# CPUs	# Cores	TFlops	Total RAM (GB)	Secondary Storage (TB)	Site
india	IBM iDataPlex	128	256	1024	11	3072	335	IU
sierra	IBM iDataPlex	84	168	672	7	2688	72	SDSC
hotel	IBM iDataPlex	84	168	672	7	2016	120	UC
foxtrot	IBM iDataPlex	32	64	256	3	768	0	UF
alamo	Dell Power Edge	96	192	768	8	1152	30	TACC
xray	Cray XT5m	1	168	672	6	1344	335	IU
Total		425	1016	4064	42	11040	557	

* secondary storage between IU machines is shared

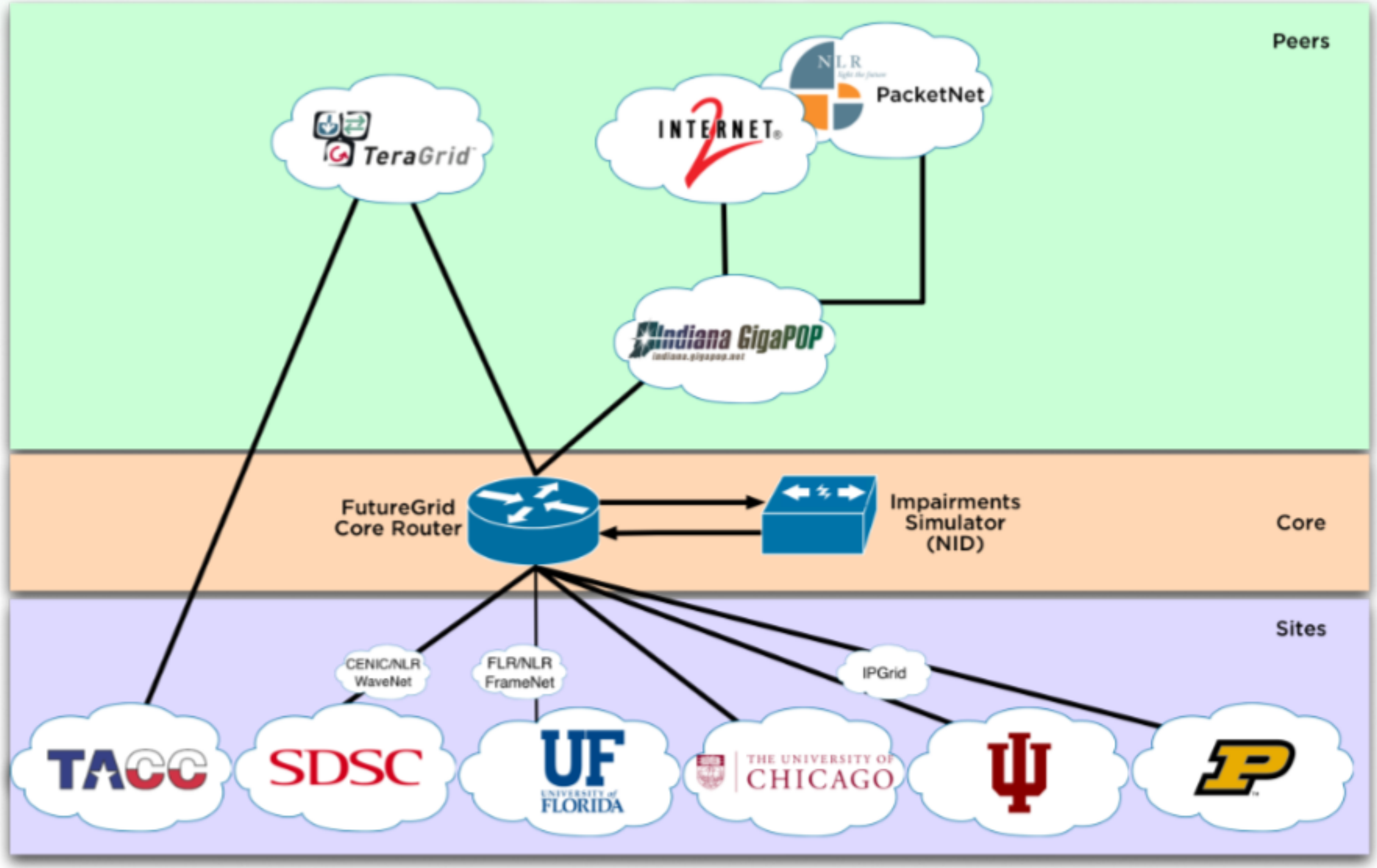
- Additional partner machines will run FutureGrid software and be supported (but allocated in specialized ways)
- (*) IU machines share same storage; (**) Shared memory and GPU Cluster in year 2

File Systems

System Type	Capacity (TB)	File System	Site	Status
DDN 9550 (Data Capacitor)	339	Lustre	IU	Existing System
DDN 6620	120	GPFS	UC	New System
SunFire x4170	72	Lustre/PVFS	SDSC	New System
Dell MD3000	30	NFS	TACC	New System

Machine	Name	Internal Network
IU Cray	xray	Cray 2D Torus SeaStar
IU iDataPlex	india	DDR IB, QLogic switch with Mellanox ConnectX adapters Blade Network Technologies & Force10 Ethernet switches
SDSC iDataPlex	sierra	DDR IB, Cisco switch with Mellanox ConnectX adapters Juniper Ethernet switches
UC iDataPlex	hotel	DDR IB, QLogic switch with Mellanox ConnectX adapters Blade Network Technologies & Juniper switches
UF iDataPlex	foxtrot	Gigabit Ethernet only (Blade Network Technologies; Force10 switches)
TACC Dell	alamo	QDR IB, Mellanox switches and adapters Dell Ethernet switches

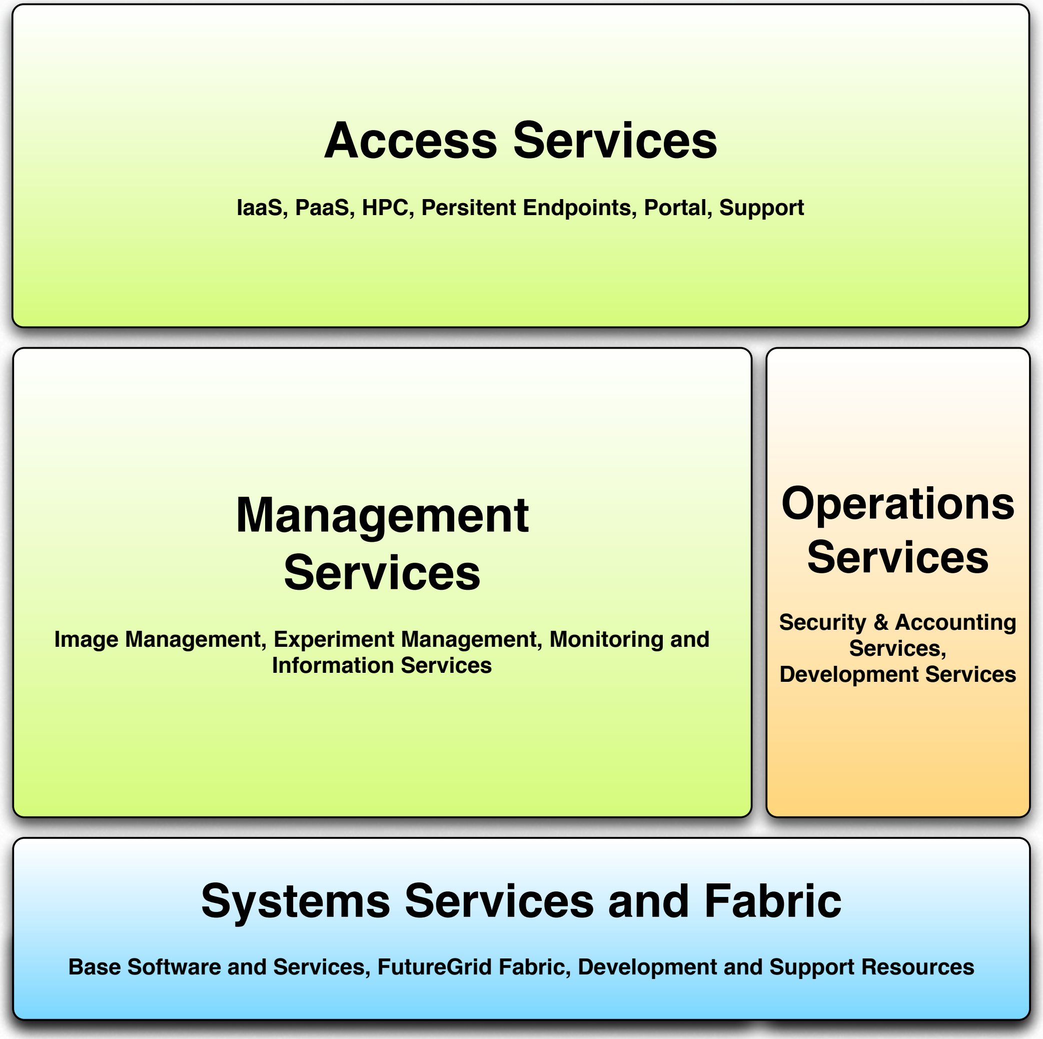
Logical Diagram



Network Impairment Device

- Spirent XGEM Network Impairments Simulator for jitter, errors, delay, etc
- Full Bidirectional 10G w/64 byte packets
- up to 15 seconds introduced delay (in 16ns increments)
- 0-100% introduced packet loss in .0001% increments
- Packet manipulation in first 2000 bytes
- up to 16k frame size
- TCL for scripting, HTML for manual configuration

Software Architecture



Access Services

IaaS, PaaS, HPC, Persitent Endpoints, Portal, Support

Management Services

Image Management, Experiment Management, Monitoring and Information Services

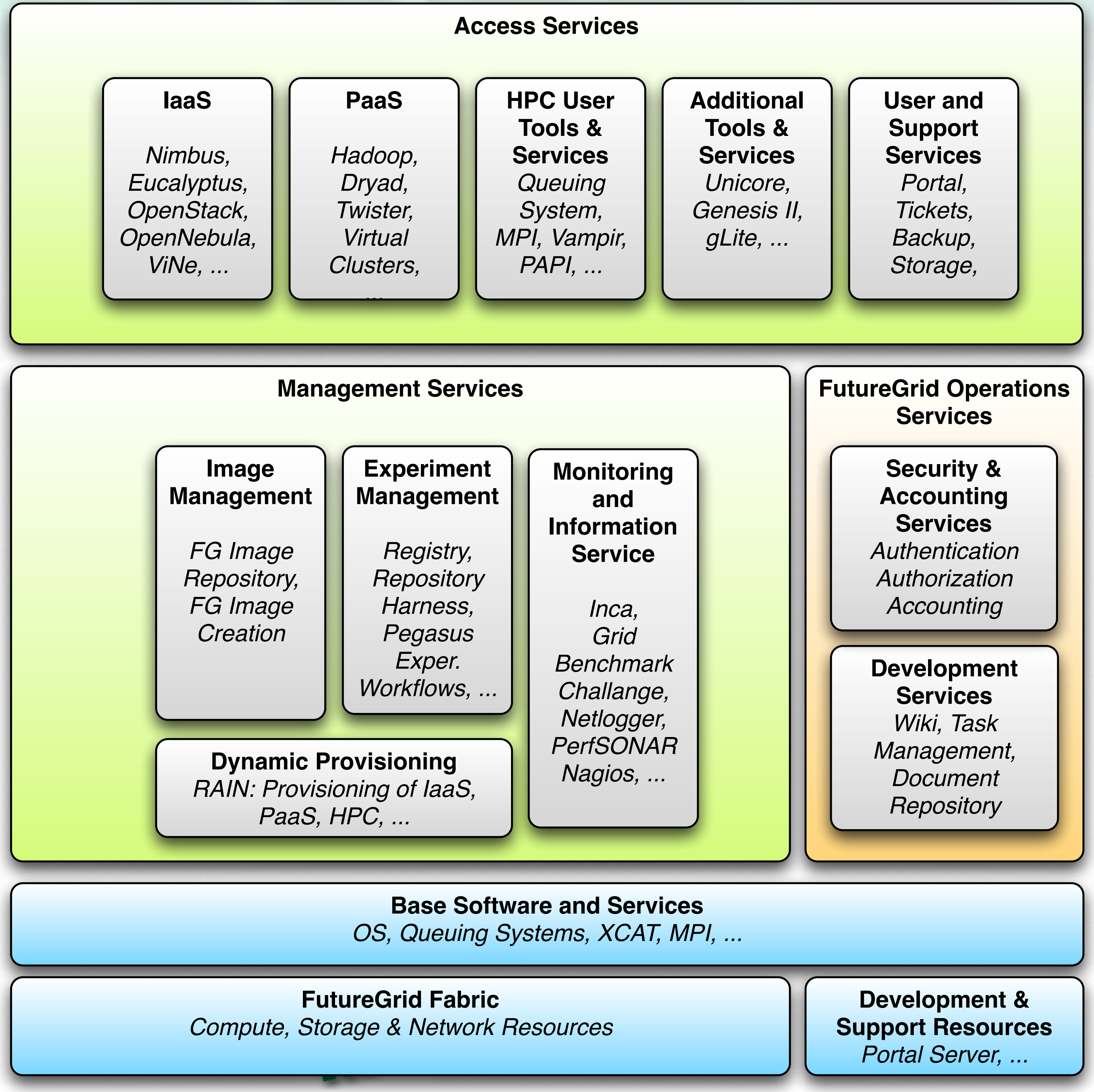
Operations Services

Security & Accounting Services, Development Services

Systems Services and Fabric

Base Software and Services, FutureGrid Fabric, Development and Support Resources

Software Architecture



Access Services

IaaS
Nimbus, Eucalyptus, OpenStack, OpenNebula, ViNe, ...

PaaS
Hadoop, Dryad, Twister, Virtual Clusters, ...

HPC User Tools & Services
Queuing System, MPI, Vampir, PAPI, ...

Additional Tools & Services
Unicore, Genesis II, gLite, ...

User and Support Services
Portal, Tickets, Backup, Storage,

Management Services

Image Management
FG Image Repository, FG Image Creation

Experiment Management
Registry, Repository Harness, Pegasus Exper. Workflows, ...

Dynamic Provisioning
RAIN: Provisioning of IaaS, PaaS, HPC, ...

Monitoring and Information Service
Inca, Grid Benchmark Challange, Netlogger, PerfSONAR Nagios, ...

FutureGrid Operations Services

Security & Accounting Services
Authentication Authorization Accounting

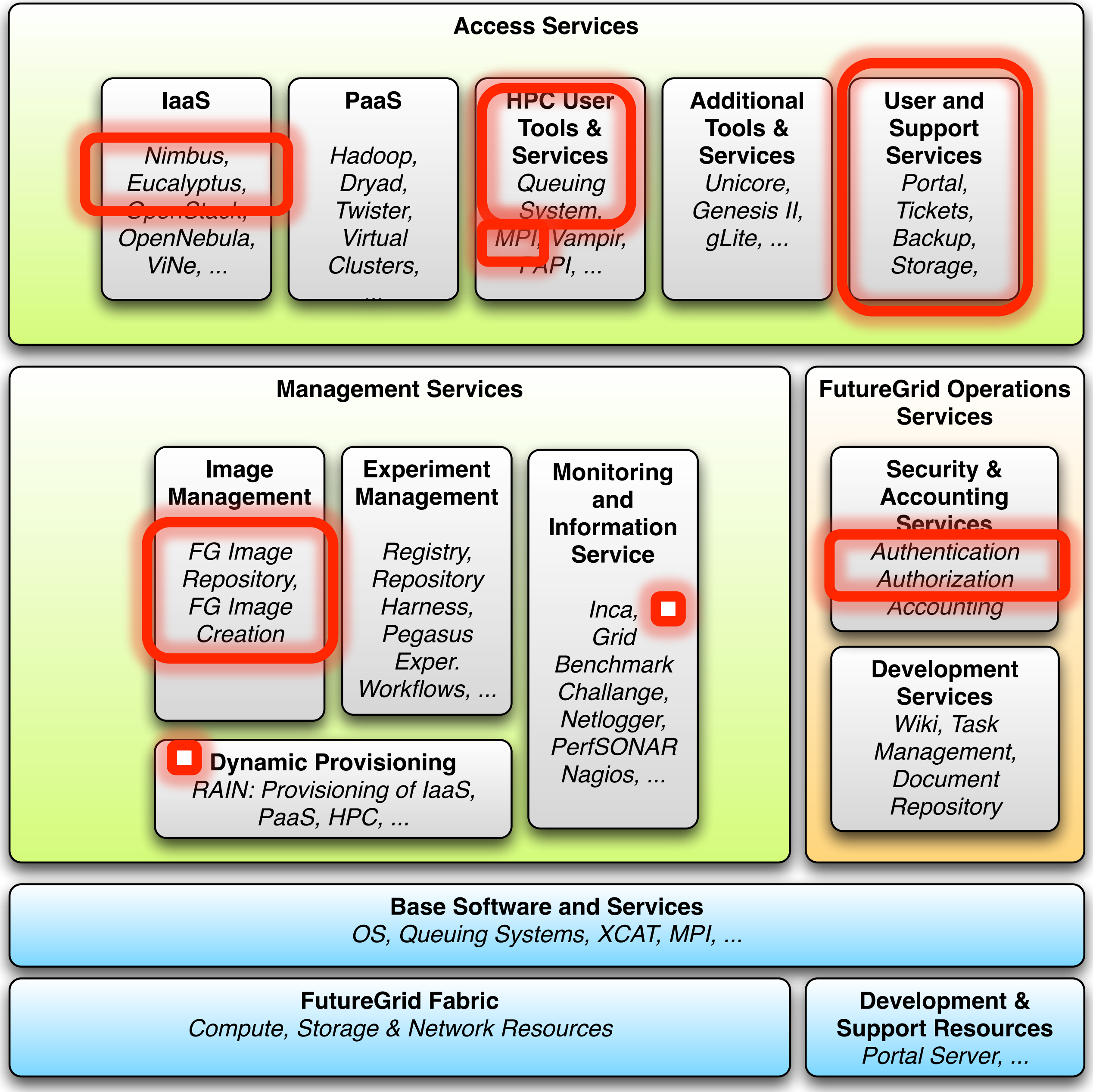
Development Services
Wiki, Task Management, Document Repository

Base Software and Services
OS, Queuing Systems, XCAT, MPI, ...

FutureGrid Fabric
Compute, Storage & Network Resources

Development & Support Resources
Portal Server, ...

We will Cover:



Access Services

IaaS
Nimbus, Eucalyptus, OpenStack, OpenNebula, ViNe, ...

PaaS
Hadoop, Dryad, Twister, Virtual Clusters, ...

HPC User Tools & Services
Queuing System, MPI, Vampir, PAPI, ...

Additional Tools & Services
Unicore, Genesis II, gLite, ...

User and Support Services
Portal, Tickets, Backup, Storage,

Management Services

Image Management
FG Image Repository, FG Image Creation

Experiment Management
Registry, Repository Harness, Pegasus Exper. Workflows, ...

Monitoring and Information Service
Inca, Grid Benchmark Challange, Netlogger, PerfSONAR Nagios, ...

Dynamic Provisioning
RAIN: Provisioning of IaaS, PaaS, HPC, ...

FutureGrid Operations Services

Security & Accounting Services
Authentication Authorization Accounting

Development Services
Wiki, Task Management, Document Repository

Base Software and Services
OS, Queuing Systems, XCAT, MPI, ...

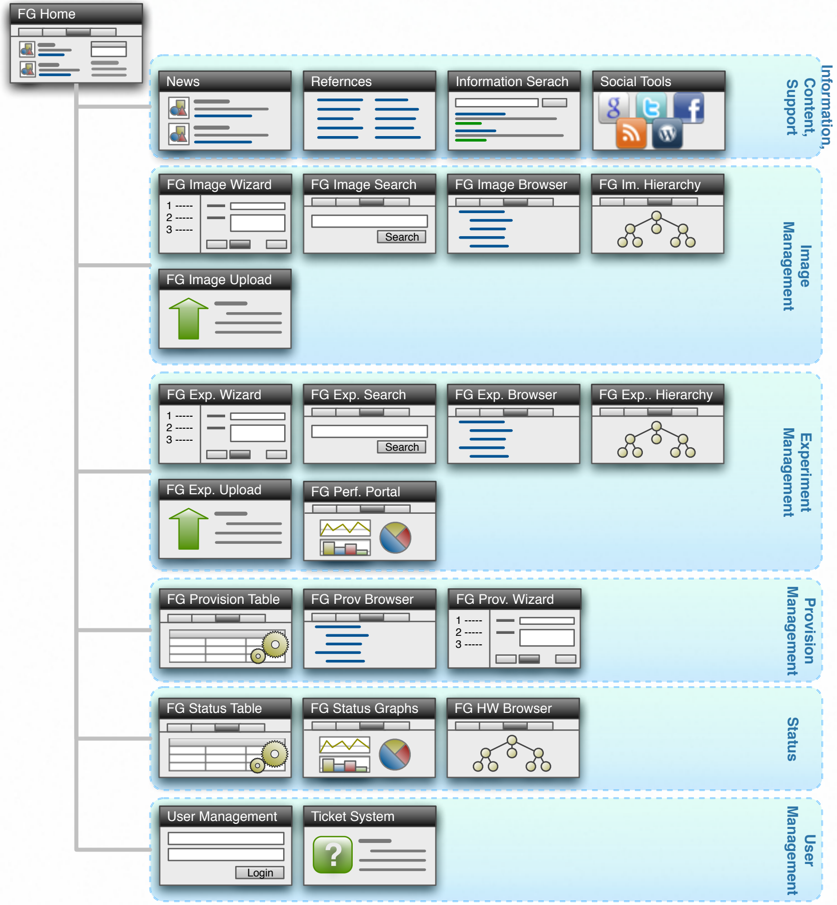
FutureGrid Fabric
Compute, Storage & Network Resources

Development & Support Resources
Portal Server, ...

Portal

Gregor von Laszewski

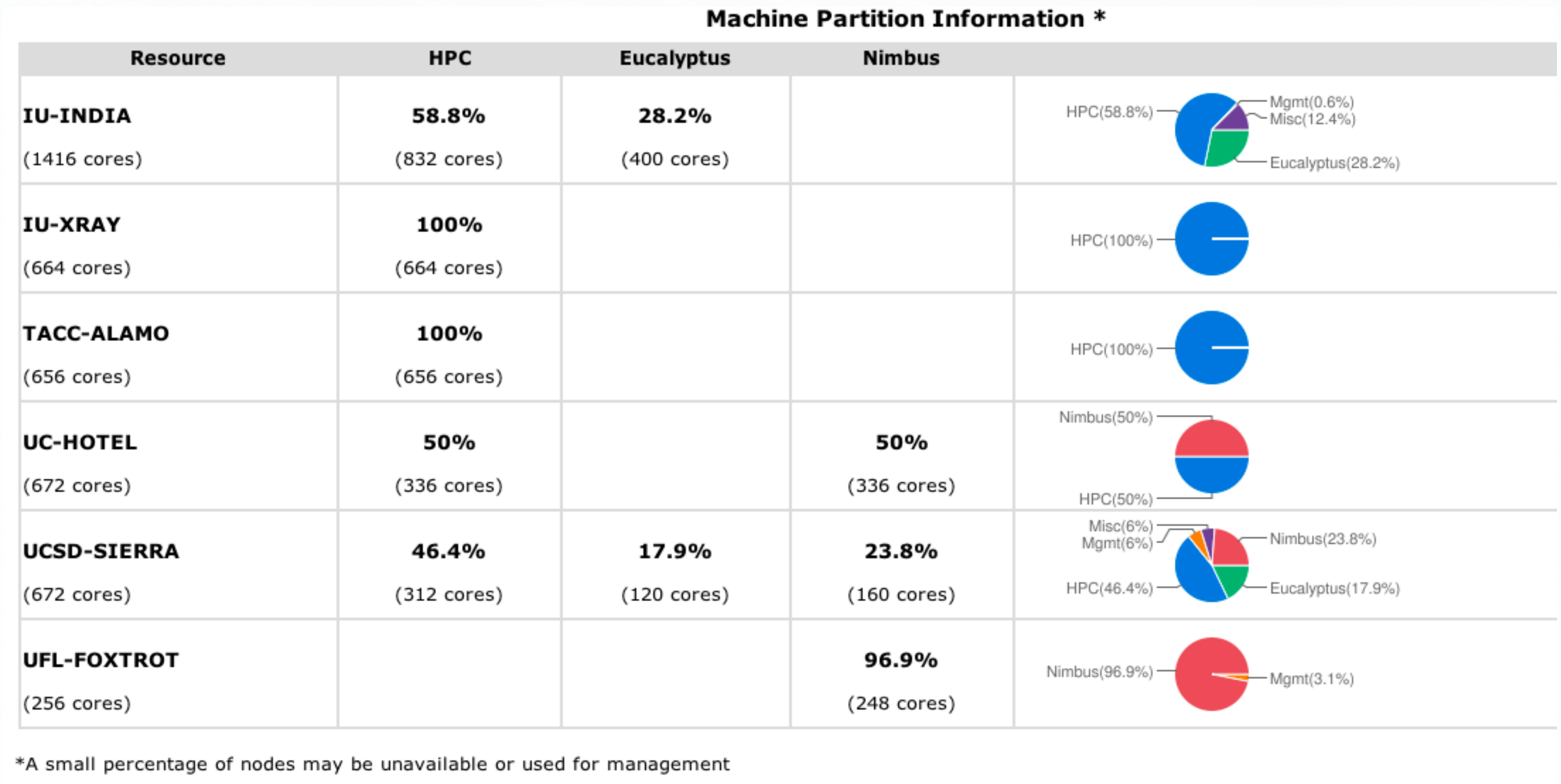
Portal Subsystem



Information Services

- What is happening on the system?
  - System administrator
  - User
  - Project Management & Funding agency
- Remember FG is not just an HPC queue!
  - Which software is used?
  - Which images are used?
  - Which FG services are used (Nimbus, Eucalyptus, …?)
  - Is the performance we expect reached?
  - What happens on the network

Simple Overview



Machine Partition Information *

Resource	HPC	Eucalyptus	Nimbus	

IU-INDIA
(1416 cores)
	58.8%
	(832 cores)
		28.2%
		(400 cores)
				HPC(58.8%)
				Mgmt(0.6%)
				Misc(12.4%)
				Eucalyptus(28.2%)

IU-XRAY (664 cores)	**100%** (664 cores)			HPC(100%)
TACC-ALAMO (656 cores)	**100%** (656 cores)			HPC(100%)

UC-HOTEL
(672 cores)
	50%
	(336 cores)
			50%
			(336 cores)
				Nimbus(50%)
				HPC(50%)

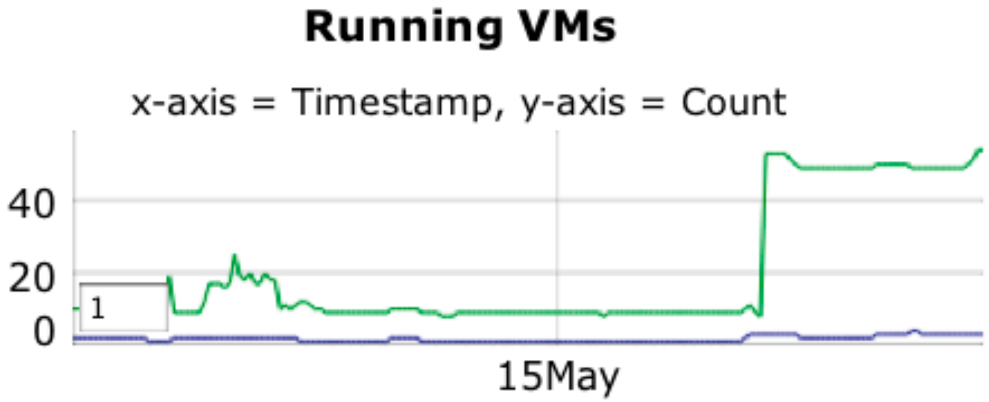
UCSD-SIERRA
(672 cores)
	46.4%
	(312 cores)
		17.9%
		(120 cores)
			23.8%
			(160 cores)
				Misc(6%)
				Mgmt(6%)
				HPC(46.4%)
				Nimbus(23.8%)
				Eucalyptus(17.9%)

UFL-FOXTROT
(256 cores)
			96.9%
			(248 cores)
				Nimbus(96.9%)
				Mgmt(3.1%)

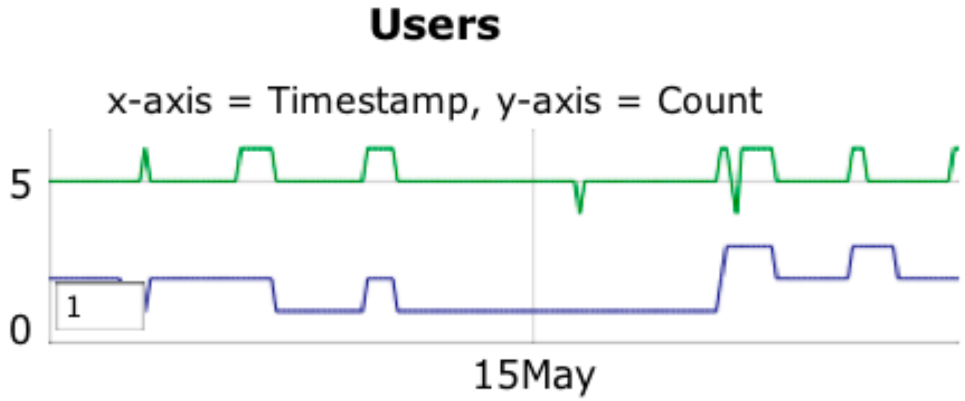
*A small percentage of nodes may be unavailable or used for management

Eucalyptus

This graph shows the number of currently running VMs within the Eucalyptus deployment on each machine.

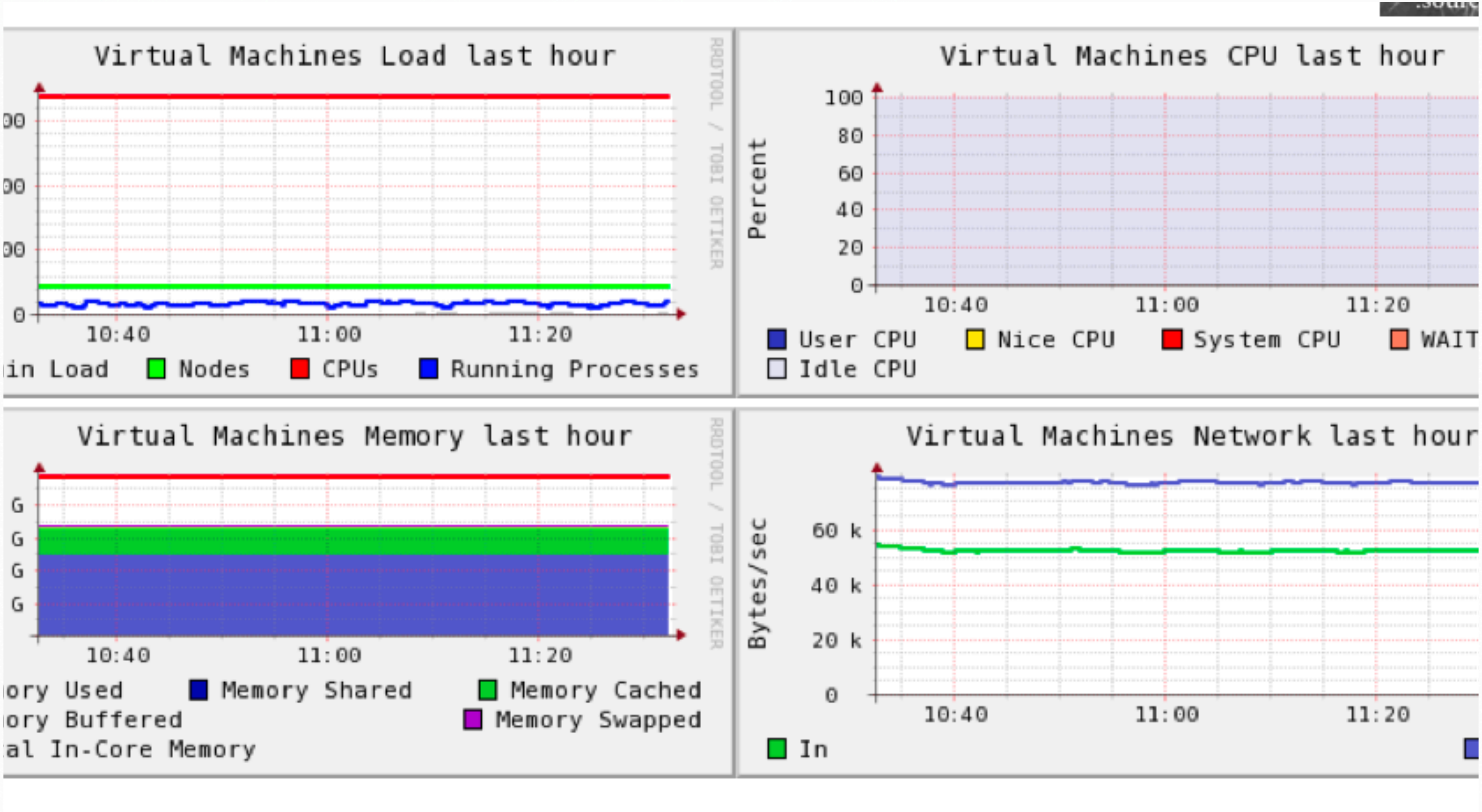


This graph shows the number of users currently running VMs within the Eucalyptus deployment on each machine.



Ganglia

On India



Using HPC Systems on FutureGrid

Andrew J. Younge
Gregory G. Pike

Indiana University

Future Grid http://futuregrid.org

A brief overview

- FutureGrid is a testbed
  - Varied resources with varied capabilities
  - Support for grid, cloud, HPC
  - Continually evolving
  - Sometimes breaks in strange and unusual ways
- FutureGrid as an experiment
  - We’ re learning as well
  - Adapting the environment to meet user needs

Getting Started

- Getting an account
- Logging in
- Setting up your environment
- Writing a job script
- Looking at the job queue
- Why won’t my job run?
- Getting your job to run sooner

http://portal.futuregrid.org/manual
http://portal.futuregrid.org/tutorials

Getting an account

- Upload your ssh key to the portal, if you have not done that when you created the portal account
    - Account -> Portal Account
        - edit the ssh key
        - or
            - Include the public portion of your SSH key!
            - use a passphrase when generating the key!!!!!

- Submit your ssh key through the portal
    - Account -> HPC

- This process may take up to 3 days.
    - If it’s been longer than a week, send email
    - We do not do any account management over weekends!

Generating an SSH key pair

- For Mac or Linux users
  - `ssh-keygen –t rsa`
  - Copy ~/.ssh/id_rsa.pub to the web form

- For Windows users, this is more difficult
  - Download putty.exe and puttygen.exe
  - Puttygen is used to generate an SSH key pair
    - Run puttygen and click “Generate”
  - The public portion of your key is in the box labeled “SSH key for pasting into OpenSSH authorized_keys file”

Logging in

- You must be logging in from a machine that has your SSH key
- Use the following command (on Linux/ OSX):
    - ssh username@india.futuregrid.org
- Substitute your FutureGrid account for username

Now you are logged in.
What is next?

Setting up your environment

- Modules is used to manage your $PATH and other environment variables
- A few common module commands
  - `module avail` – lists all available modules
  - `module list` – lists all loaded modules
  - `module load` – adds a module to your environment
  - `module unload` – removes a module from your environment
  - `module clear` –removes all modules from your environment

Writing a job script

- A job script has PBS directives followed by the commands to run your job

```
•  #!/bin/bash
•  #PBS -N testjob
•  #PBS -l nodes=1:ppn=8
•  #PBS -q batch
•  #PBS -M username@example.com
•  ##PBS -o testjob.out
•  #PBS -j oe
•  #
•  sleep 60
•  hostname
•  echo $PBS_NODEFILE
•  cat $PBS_NODEFILE
•  sleep 60
```

Writing a job script

- Use the qsub command to submit your job
  - qsub testjob.pbs
- Use the qstat command to check your job

```
> qsub testjob.pbs
25265.i136


> qstat
Job id     Name          User  Time Use S Queue
---------- ------------- ----- -------- - ------
25264.i136 sub27988.sub  inca  00:00:00 C batch
25265.i136 testjob       gpike 0        R batch
```

Looking at the job queue

- Both *qstat* and *showq* can be used to show what’s running on the system
- The *showq* command gives nicer output
- The *pbsnodes* command will list all nodes and details about each node
- The *checknode* command will give extensive details about a particular node

Run `module load moab` to add commands to path

Why won’t my job run?

- Two common reasons:
  - The cluster is full and your job is waiting for other jobs to finish
  - You asked for something that doesn’t exist
    - More CPUs or nodes than exist
  - The job manager is optimistic!
    - If you ask for more resources than we have, the job manager will sometimes hold your job until we buy more hardware

Why won’t my job run?

- Use the checkjob command to see why your job will not run

```
> checkjob 319285


job 319285

Name: testjob
State: Idle
Creds: user:gpike group:users class:batch qos:od
WallTime: 00:00:00 of 4:00:00
SubmitTime: Wed Dec 1 20:01:42
(Time Queued Total: 00:03:47 Eligible: 00:03:26)

Total Requested Tasks: 320

Req[0] TaskCount: 320 Partition: ALL

Partition List: ALL,s82,SHARED,msm
Flags: RESTARTABLE
Attr: checkpoint
StartPriority: 3
NOTE: job cannot run (insufficient available procs: 312 available)
```

Why won’t my job run?

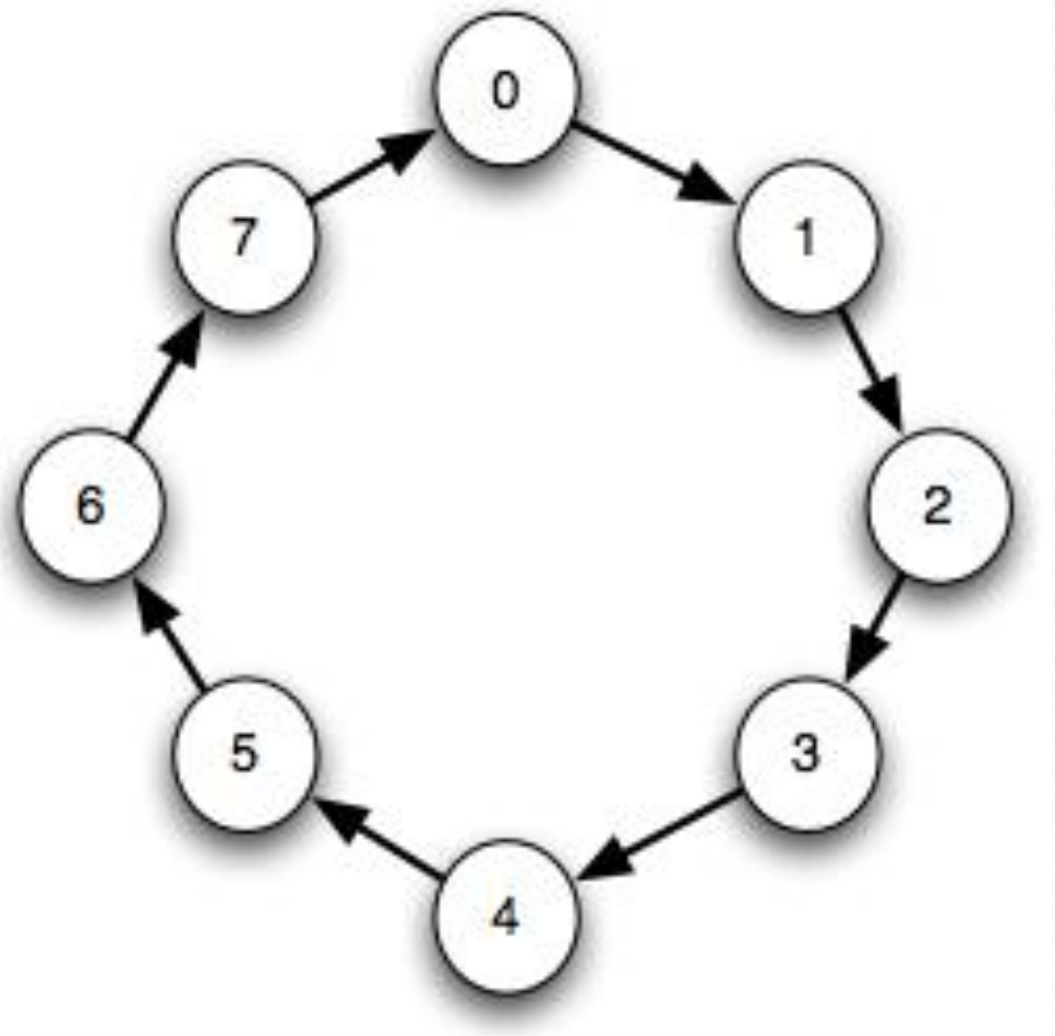
- If you submitted a job that cannot run, use qdel to delete the job, fix your script, and resubmit the job
  - `qdel 319285`
- If you think your job should run, leave it in the queue and send email
- It’s also possible that maintenance is coming up soon

Making your job run sooner

- In general, specify the minimal set of resources you need
  - Use minimum number of nodes
  - Use the job queue with the shortest max walltime
    - `qstat –Q –f`
  - Specify the minimum amount of time you need for the job
    - `qsub –l walltime=hh:mm:ss`

Example with MPI

- Run through a simple example of an MPI job
  - Ring algorithm passes messages along to each process as a chain or string
  - Use Intel compiler and mpi to compile & run
  - Hands on experience with PBS scripts



```
#PBS -N hello-mvapich-intel
#PBS -l nodes=4:ppn=8
#PBS -l walltime=00:02:00
#PBS -k oe
#PBS -j oe

EXE=$HOME/mpiring/mpiring

echo "Started on `/bin/hostname`"
echo
echo "PATH is [$PATH]"
echo
echo "Nodes chosen are:"
cat $PBS_NODEFILE
echo
module load intel intelmpi
mpdboot -n 4 -f $PBS_NODEFILE -v --remcons

mpiexec -n 32 $EXE

mpdallexit
```

Lets Run

```
> cp /share/project/mpiexample/mpiring.tar.gz .
> tar xfz mpiring.tar.gz
> cd mpiring
> module load intel intelmpi moab

Intel compiler suite version 11.1/072 loaded
Intel MPI version 4.0.0.028 loaded
moab version 5.4.0 loaded

> mpicc -o mpiring ./mpiring.c
> qsub mpiring.pbs
100506.i136

> cat ~/hello-mvapich-intel.o100506
...
```

Eucalyptus on FutureGrid

Andrew J. Younge

Indiana University

Before you can use Eucalyptus

- Please make sure you have a portal account
  - https://portal.futuregrid.org
- Please make sure you are part of a valid FG project
  - You can either create a new one or
  - You can join an existing one with permission of the Lead
- Please make sure the project you have is approved and valid.
- Do not apply for an account before you have joined the project, your Eucalyptus account request will not be granted!

Eucalyptus Systems

Eucalyptus

- Elastic Utility Computing Architecture Linking Your Programs To Useful Systems
  - Eucalyptus is an open-source software platform that implements IaaS-style cloud computing using the existing Linux-based infrastructure
  - IaaS Cloud Services providing atomic allocation for
    - Set of VMs
    - Set of Storage resources
    - Networking

Open Source Eucalyptus

- **Eucalyptus Features**
    - Amazon AWS Interface Compatibility
    - Web-based interface for cloud configuration and credential management.
    - Flexible Clustering and Availability Zones.
    - Network Management, Security Groups, Traffic Isolation
        - Elastic IPs, Group based firewalls etc.
    - Cloud Semantics and Self-Service Capability
        - Image registration and image attribute manipulation
    - Bucket-Based Storage Abstraction (S3-Compatible)
    - Block-Based Storage Abstraction (EBS-Compatible)
    - Xen and KVM Hypervisor Support

Source: http://www.eucalyptus.com

Eucalyptus Testbed

- Eucalyptus is available to FutureGrid Users on the India and Sierra clusters.
- Users can make use of a maximum of 50 nodes on India. Each node supports up to 8 small VMs. Different Availability zones provide VMs with different compute and memory capacities.

```
AVAILABILITYZONE     india   149.165.146.135
AVAILABILITYZONE     |- vm types     free / max   cpu   ram  disk
AVAILABILITYZONE     |- m1.small     0400 / 0400   1    512     5
AVAILABILITYZONE     |- c1.medium    0400 / 0400   1   1024     7
AVAILABILITYZONE     |- m1.large     0200 / 0200   2   6000    10
AVAILABILITYZONE     |- m1.xlarge    0100 / 0100   2  12000    10
AVAILABILITYZONE     |- c1.xlarge    0050 / 0050   8  20000    10
```

Eucalyptus Account Creation

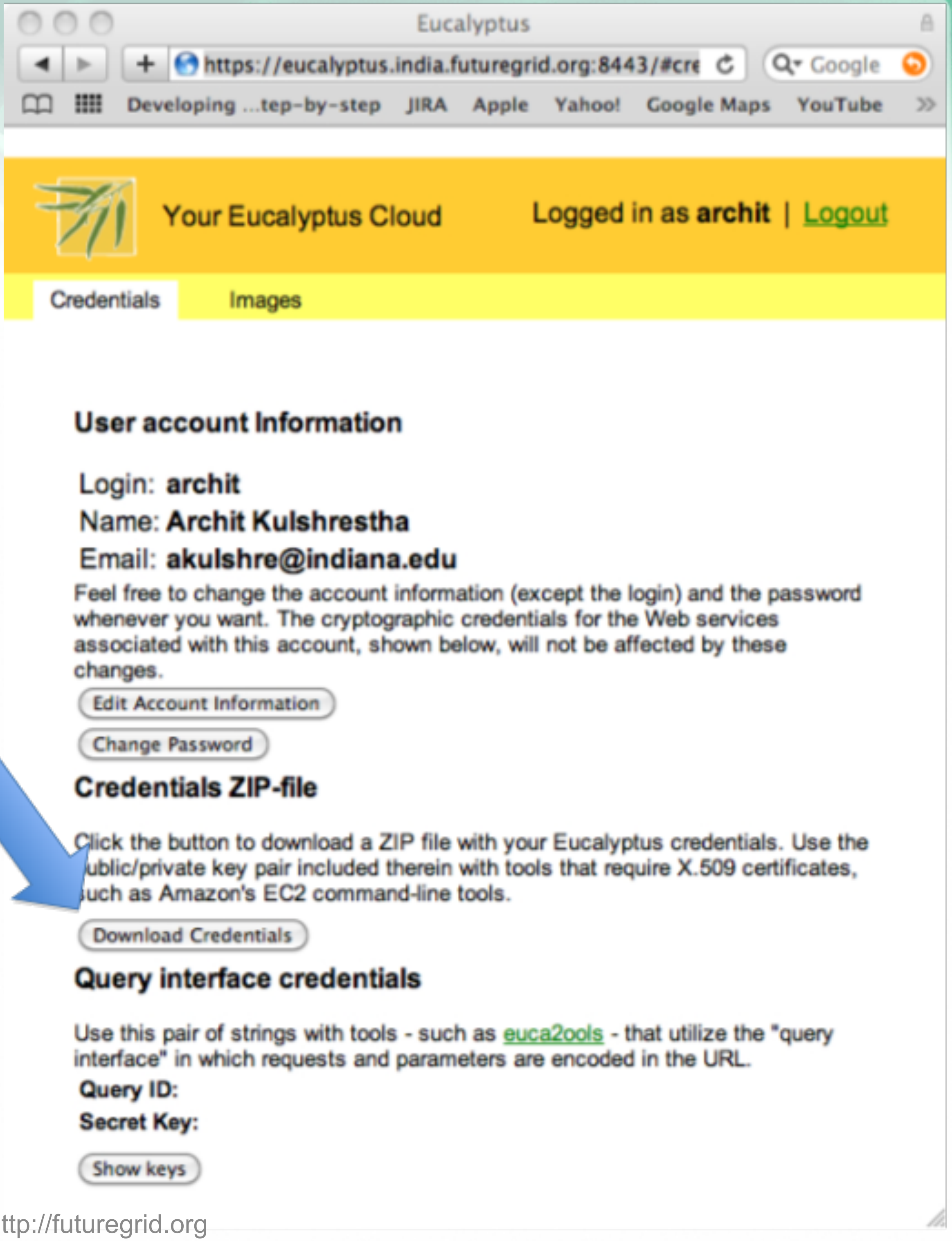
- Use the Eucalyptus Web Interfaces at

  https://eucalyptus.india.futuregrid.org:8443/

- On the Login page click on Apply for account.
- On the next page that pops up fill out ALL the Mandatory AND optional fields of the form.
- Once complete click on signup and the Eucalyptus administrator will be notified of the account request.
- You will get an email once the account has been approved.
- Click on the link provided in the email to confirm and complete the account creation process.

Obtaining Credentials

- Download your credentials as a zip file from the web interface for use with euca2ools.
- Save this file and extract it for local use or copy it to India/Sierra.
- On the command prompt change to the euca2-{username}-x509 folder which was just created.
    - cd euca2-username-x509
- Source the eucarc file using the command source eucarc.
    - source ./eucarc



Eucalyptus

https://eucalyptus.india.futuregrid.org:8443/#cre

Your Eucalyptus Cloud

Logged in as **archit** | Logout

User account Information

Login: **archit**

Name: **Archit Kulshrestha**

Email: **akulshre@indiana.edu**

Feel free to change the account information (except the login) and the password whenever you want. The cryptographic credentials for the Web services associated with this account, shown below, will not be affected by these changes.

Credentials ZIP-file

Click the button to download a ZIP file with your Eucalyptus credentials. Use the public/private key pair included therein with tools that require X.509 certificates, such as Amazon's EC2 command-line tools.

Query interface credentials

Use this pair of strings with tools - such as euca2ools - that utilize the "query interface" in which requests and parameters are encoded in the URL.

Query ID:

Secret Key:

Install/Load Euca2ools

- Euca2ools are the command line clients used to interact with Eucalyptus.
- If using your own platform Install euca2ools bundle from http://open.eucalyptus.com/downloads
    - Instructions for various Linux platforms are available on the download page.
- On FutureGrid log on to India/Sierra and load the Euca2ools module.

```
$ module load euca2ools
euca2ools version 1.2 loaded
```

Euca2ools

- Testing your setup
    - Use euca-describe-availability-zones to test the setup.
- List the existing images using euca-describe-images

```
euca-describe-availability-zones
AVAILABILITYZONE india 149.165.146.135
```

```
$ euca-describe-images
IMAGE emi-0B951139 centos53/centos.5-3.x86-64.img.manifest.xml admin
available public x86_64 machine
IMAGE emi-409D0D73 rhel55/rhel55.img.manifest.xml admin available public
x86_64 machine
…
```

Key management

- Create a keypair and add the public key to eucalyptus.

```
$ euca-add-keypair userkey > userkey.pem
```

- Fix the permissions on the generated private key.

```
$ chmod 0600 userkey.pem
```

```
$ euca-describe-keypairs
KEYPAIR userkey 0d:d8:7c:2c:bd:85:af:7e:ad:8d:
   09:b8:ff:b0:54:d5:8c:66:86:5d
```

Image Deployment

- Now we are ready to start a VM using one of the pre-existing images.
- We need the emi-id of the image that we wish to start. This was listed in the output of euca-describe-images command that we saw earlier.
  - We use the euca-run-instances command to start the VM.

```
$ euca-run-instances -k userkey -n 1 emi-0B951139 -t c1.medium
RESERVATION r-4E730969 archit archit-default
INSTANCE i-4FC40839 emi-0B951139 0.0.0.0 0.0.0.0 pending userkey
2010-07-20T20:35:47.015Z eki-78EF12D2 eri-5BB61255
```

Monitoring

- euca-describe-instances shows the status of the VMs.

```
$ euca-describe-instances
RESERVATION r-4E730969 archit default
INSTANCE i-4FC40839 emi-0B951139 149.165.146.153 10.0.2.194 pending
userkey 0 m1.small 2010-07-20T20:35:47.015Z india eki-78EF12D2
eri-5BB61255
```

- Shortly after…

```
$ euca-describe-instances
RESERVATION r-4E730969 archit default
INSTANCE i-4FC40839 emi-0B951139 149.165.146.153 10.0.2.194 running
userkey 0 m1.small 2010-07-20T20:35:47.015Z india eki-78EF12D2
eri-5BB61255
```

VM Access

- First we must create rules to allow access to the VM over ssh.

```
euca-authorize -P tcp -p 22 -s 0.0.0.0/0 default
```

- The ssh private key that was generated earlier can now be used to login to the VM.

```
ssh -i userkey.pem root@149.165.146.153
```

Image Deployment (1/3)

- We will use the example Fedora 10 image to test uploading images.
  - Download the gzipped tar ball

```
wget http://open.eucalyptus.com/sites/all/modules/pubdlcnt/pubdlcnt.php?
file=http://www.eucalyptussoftware.com/downloads/eucalyptus-images/euca-
fedora-10-x86_64.tar.gz&amp;nid=1210
```

- Uncompress and Untar the archive

```
tar zxf euca-fedora-10-x86_64.tar.gz
```

Image Deployment (2/3)

- Next we bundle the image with a kernel and a ramdisk using the euca-bundle-image command.
  - We will use the xen kernel already registered.
    - euca-describe-images returns the kernel and ramdisk IDs that we need.

```
$ euca-bundle-image -i euca-fedora-10-x86_64/fedora.10.x86-64.img --
   kernel eki-78EF12D2 --ramdisk eri-5BB61255
```

- Use the generated manifest file to upload the image to Walrus

```
$ euca-upload-bundle -b fedora-image-bucket -m /tmp/fedora.
   10.x86-64.img.manifest.xml
```

Image Deployment (3/3)

- Register the image with Eucalyptus

```
euca-register fedora-image-bucket/fedora.10.x86-64.img.manifest.xml
```

- This returns the image ID which can also be seen using euca-describe-images

```
$ euca-describe-images
IMAGE emi-FFC3154F fedora-image-bucket/fedora.
    10.x86-64.img.manifest.xml archit available public x86_64 machine
    eri-5BB61255 eki-78EF12D2
IMAGE emi-0B951139 centos53/centos.5-3.x86-64.img.manifest.xml
    admin available public x86_64 machine ...
```

Technology Preview

Dynamic Provisioning & RAIN on FutureGrid

Gregor von Laszewski

Future Grid http://futuregrid.org

Technology Preview

Classical Dynamic Provisioning

- Dynamically partition a set of resources
- Dynamically allocate the resources to users
- Dynamically define the environment that the resource use
- Dynamically assign them based on user request
- Deallocate the resources so they can be dynamically allocated again

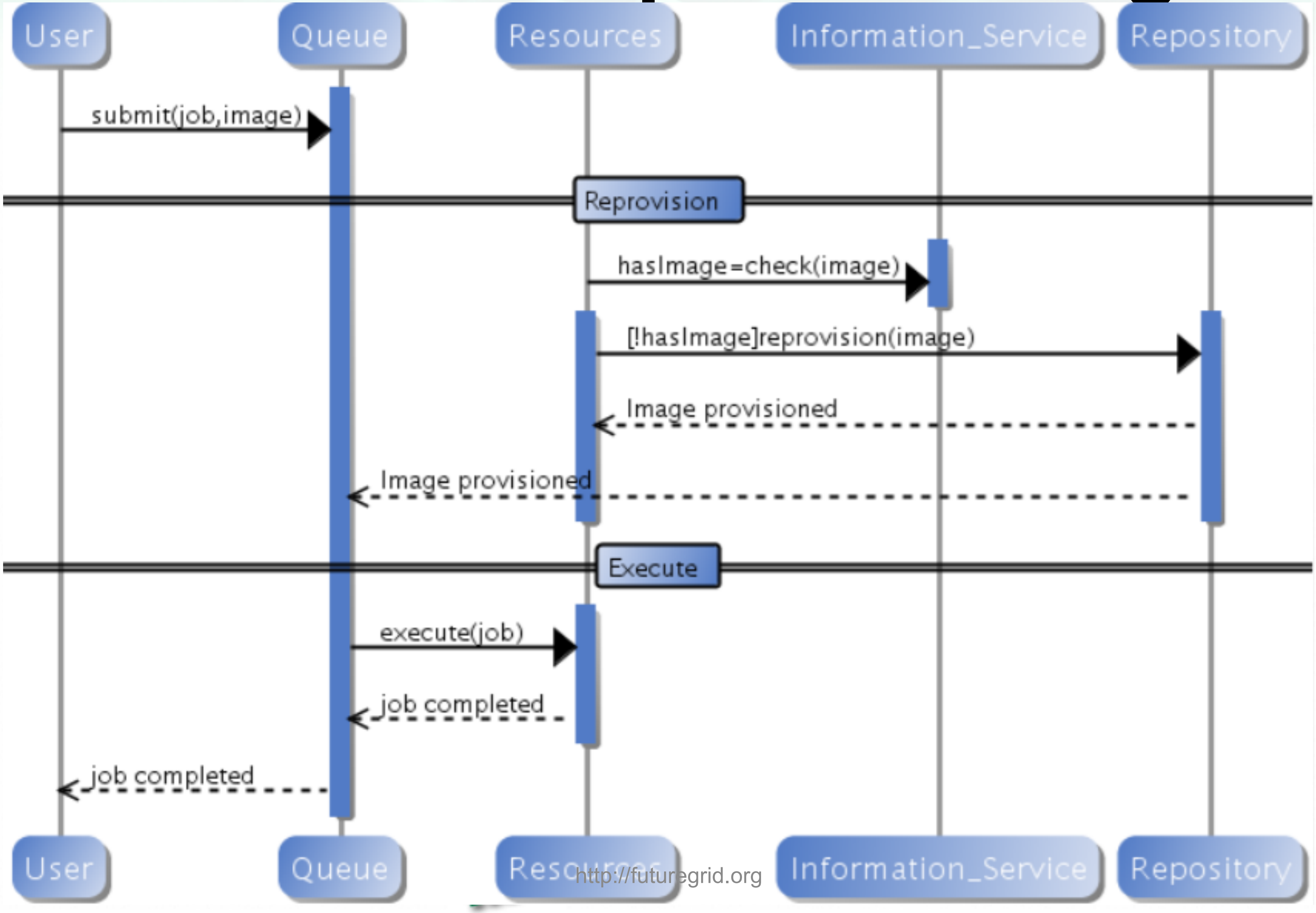
Use Cases of Dynamic Provisioning

Technology Preview

- Static provisioning:
  - Resources in a cluster may be statically reassigned based on the anticipated user requirements, part of an HPC or cloud service. It is still dynamic, but control is with the administrator. (Note some call this also dynamic provisioning.)
- Automatic Dynamic provisioning:
  - Replace the administrator with intelligent scheduler.
- Queue-based dynamic provisioning:
  - provisioning of images is time consuming, group jobs using a similar environment and reuse the image. User just sees queue.
- Deployment:
  - dynamic provisioning features are provided by a combination of using XCAT and Moab

Technology Preview

Generic Reprovisioning



Dynamic Provisioning Examples

Technology Preview

- Give me a virtual cluster with 30 nodes based on Xen
- Give me 15 KVM nodes each in Chicago and Texas linked to Azure and Grid5000
- Give me a Eucalyptus environment with 10 nodes
- Give 32 MPI nodes running on first Linux and then Windows
- Give me a Hadoop environment with 160 nodes
- Give me a 1000 BLAST instances linked to Grid5000

- Run my application on Hadoop, Dryad, Amazon and Azure … and compare the performance

From Dynamic Provisioning to "RAIN"

Technology Preview

- In FG dynamic provisioning goes beyond the services offered by common scheduling tools that provide such features.
  - Dynamic provisioning in FutureGrid means more than just providing an image
  - adapts the image at runtime and provides besides IaaS, PaaS, also SaaS
  - We call this "raining" an environment
- Rain = Runtime Adaptable INsertion Configurator
  - Users want to ``rain'' an HPC, a Cloud environment, or a virtual network onto our resources with little effort.
  - Command line tools supporting this task.
  - Integrated into Portal
- Example ``rain'' a Hadoop environment defined by an user on a cluster.
  - fg-hadoop -n 8 -app myHadoopApp.jar …
  - Users and administrators do not have to set up the Hadoop environment as it is being done for them

Technology Preview

FG RAIN Commands

- fg-rain –h hostfile –iaas nimbus –image img
- fg-rain –h hostfile –paas hadoop …
- fg-rain –h hostfile –paas dryad …
- fg-rain –h hostfile –gaas gLite …

- fg-rain –h hostfile –image img

- Additional Authorization is required to use fg-rain without virtualization.

Technology Preview

Rain in FutureGrid

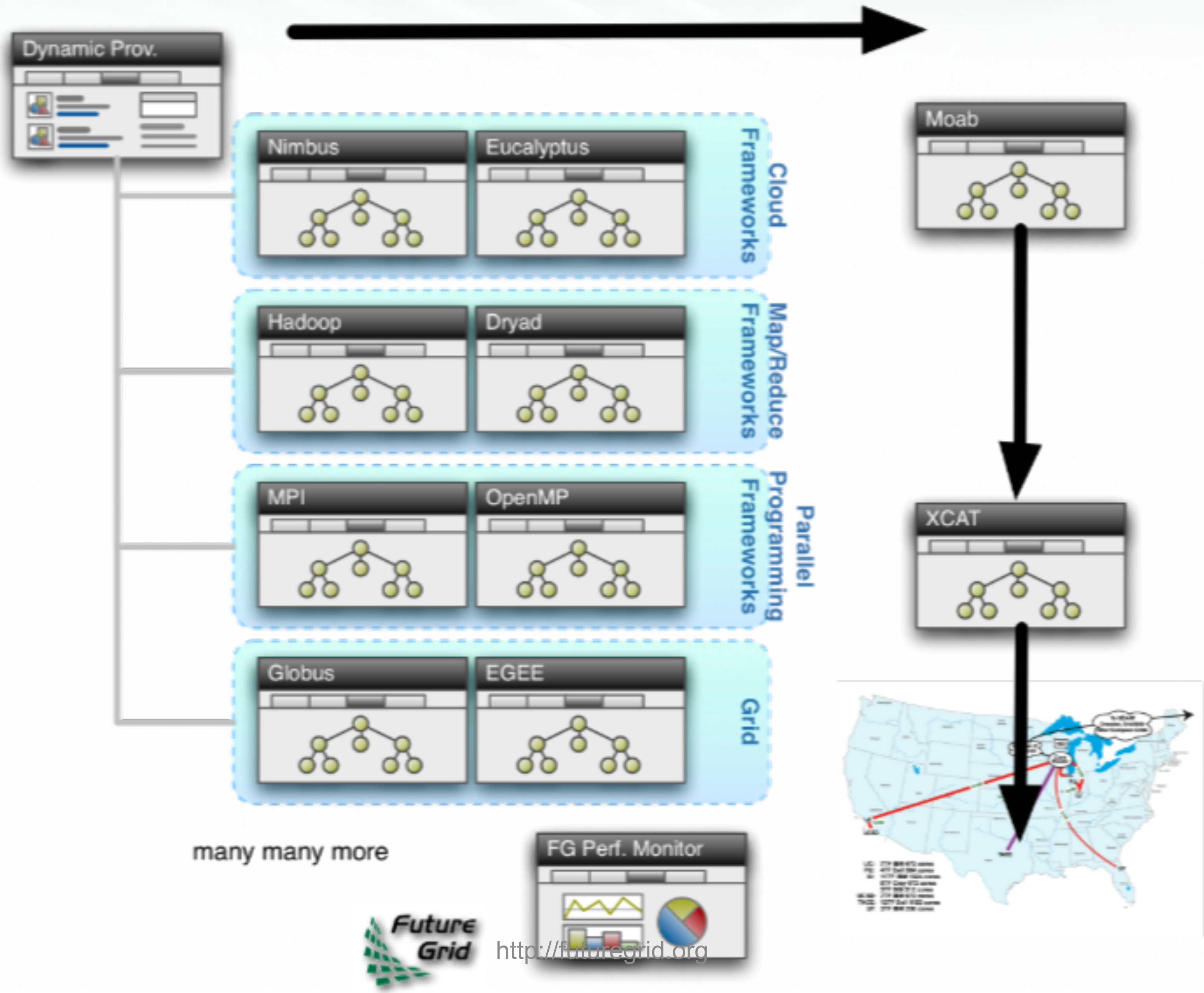


Image Generation and Management on FutureGrid

Gregor von Laszewski

Motivation

- The goal is to create and maintain platforms in custom FG VMs that can be retrieved, deployed, and provisioned on demand.
- Imagine the following scenario for FutureGrid:
    - fg-image-generate –o ubuntu –v lucid -s openmpi-bin,openmpi-dev,gcc,fftw2,emacs – n ubuntu-mpi-dev
    - fg-image-store –i ajyounge-338373292.manifest.xml –n ubuntu-mpi-dev
    - fg-image-deploy –e india.futuregrid.org –i /tmp/ajyounge-338373292.manifest.xml
    - fg-rain –provision -n 32 ubuntu-mpi-dev

Image Management

- A unified Image Management system to create and maintain VM and bare-metal images.
- Integrate images through a repository to instantiate services on demand with RAIN.
- Essentially enables the rapid development and deployment of Platform services on FutureGrid infrastructure.

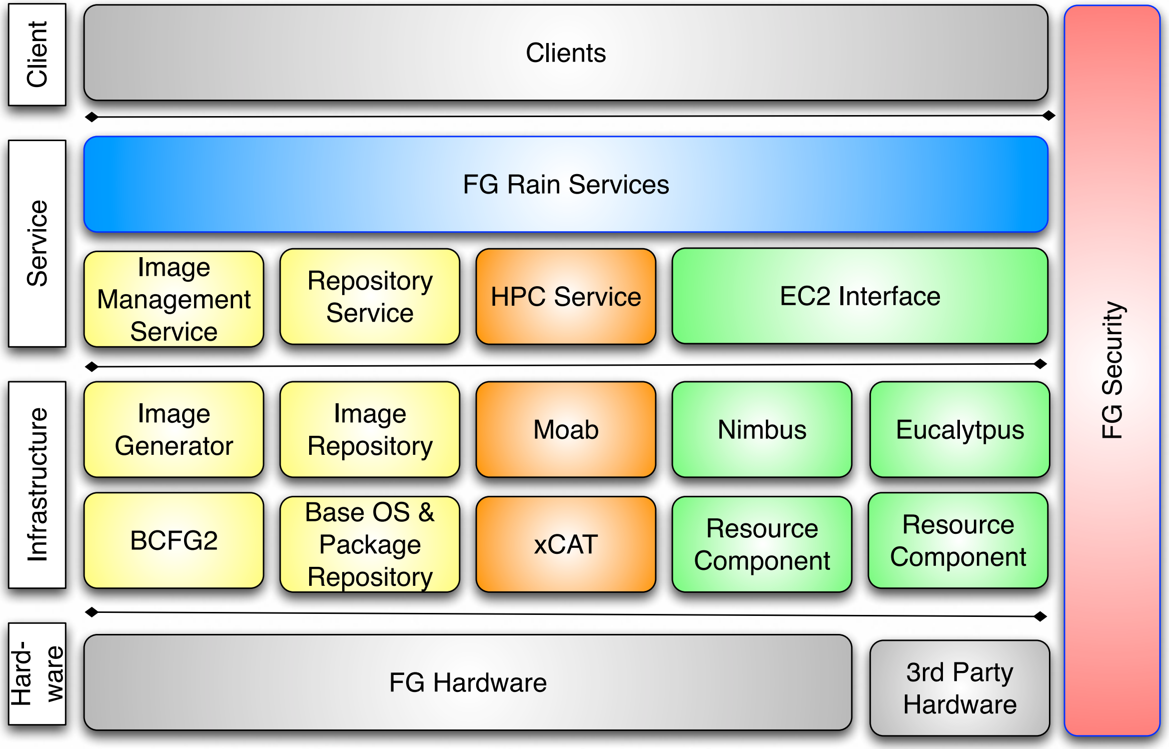
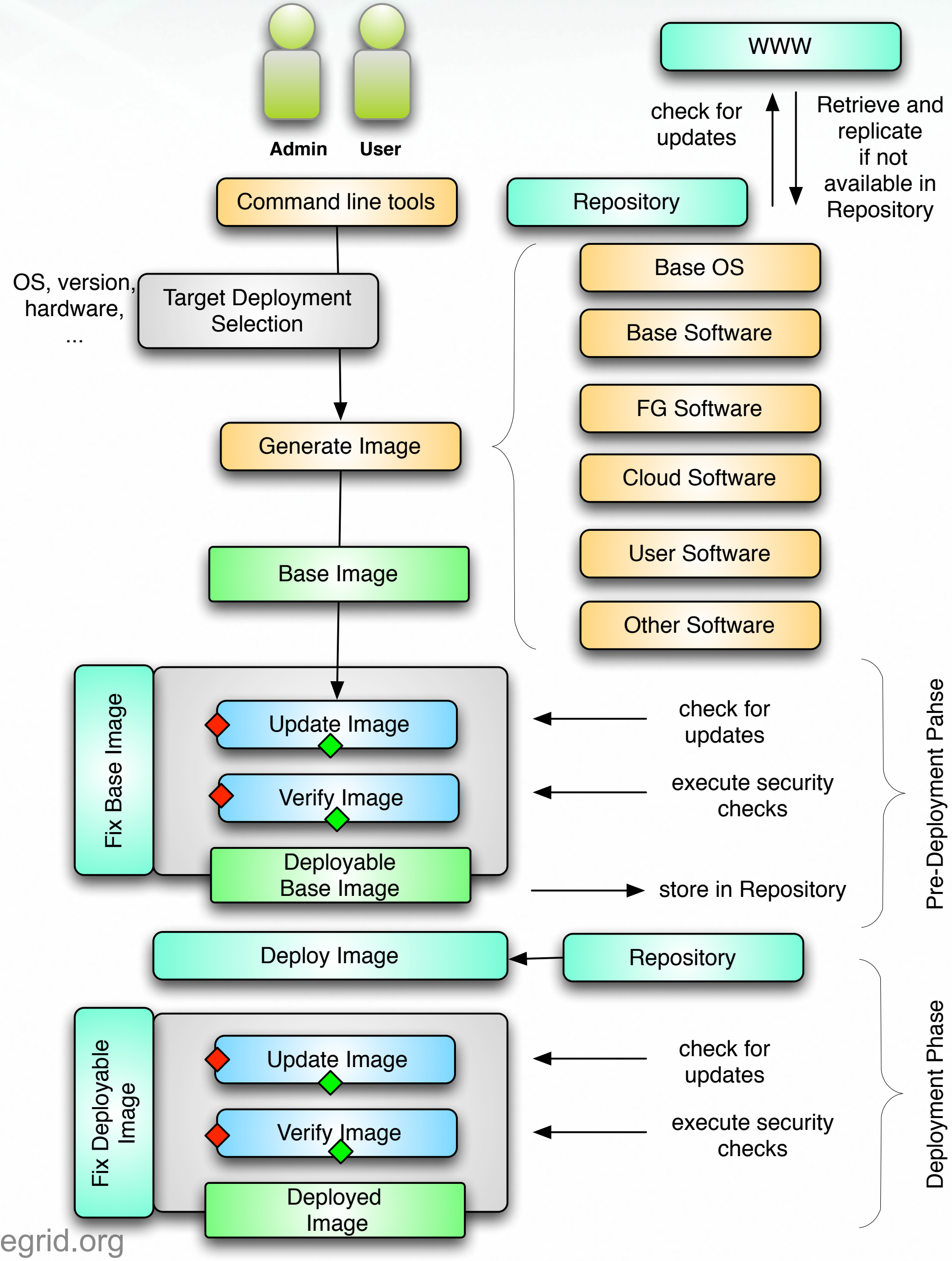


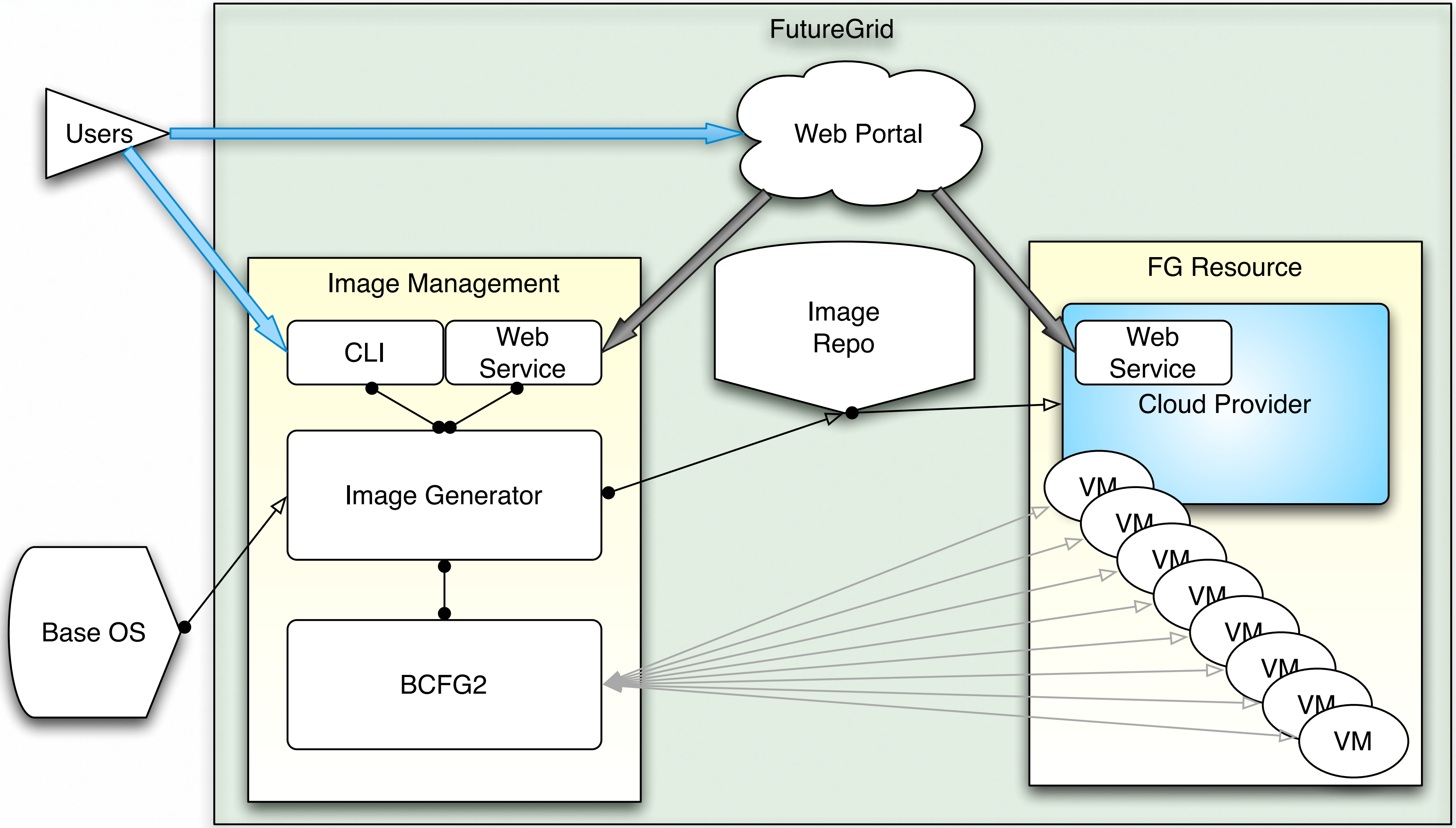
Image Generation

- Users who want to create a new FG image specify the following:
  - OS type
  - OS version
  - Architecture
  - Kernel
  - Software Packages
- Image is generated, then deployed to specified target.
- Deployed image gets continuously scanned, verified, and updated.
- Images are now available for use on the target deployed system.



FutureGrid http://futuregrid.org

Deployment View



Implementation

- Image Generator
  - Still in development, but alpha available now.
  - Built in Python.
  - Debootstrap for debian & ubuntu, YUM for RHEL5, CentOS, & Fedora.
  - Simple CLI now, but later incorporate a web service to support the FG Portal.
  - Deployment to Eucalyptus & Bare metal now, Nimbus and others soon.

- Image Management
  - Currently operating an experimental BCFG2 server.
  - Image Generator auto-creates new user groups for software stacks.
  - Supporting RedHat and Ubuntu repo mirrors.
  - Scalability experiments of BCFG2 to be tested, but previous work shows scalability to thousands of VMs without problems.

Image Repository on FutureGrid

Gregor

Image Repository

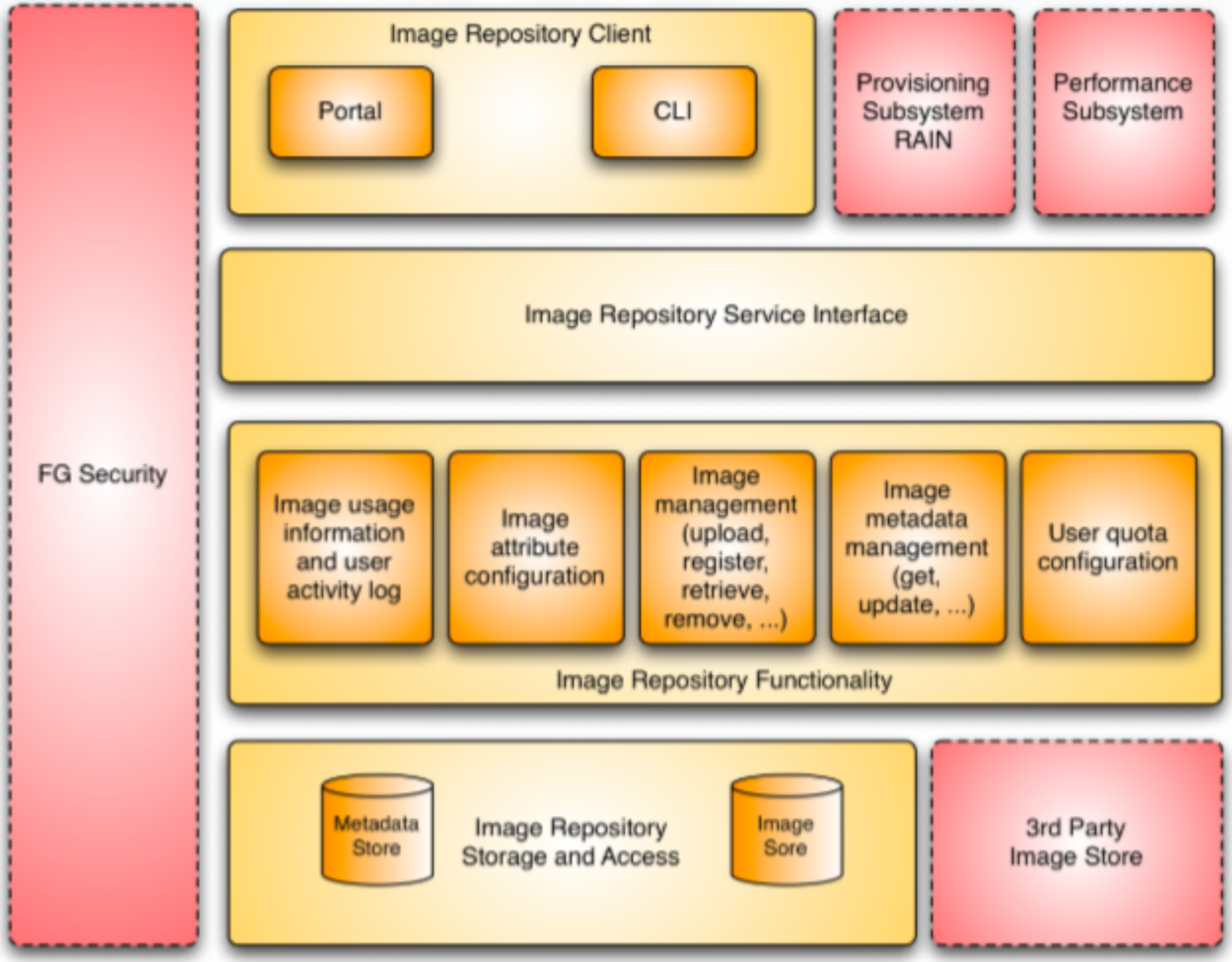


Image Repository Client
Portal
CLI
Provisioning Subsystem RAIN
Performance Subsystem
FG Security
Image Repository Service Interface
Image usage information and user activity log
Image attribute configuration
Image management (upload, register, retrieve, remove, ...)
Image metadata management (get, update, ...)
User quota configuration
Image Repository Functionality
Metadata Store
Image Repository Storage and Access
Image Sore
3rd Party Image Store

Technology Preview

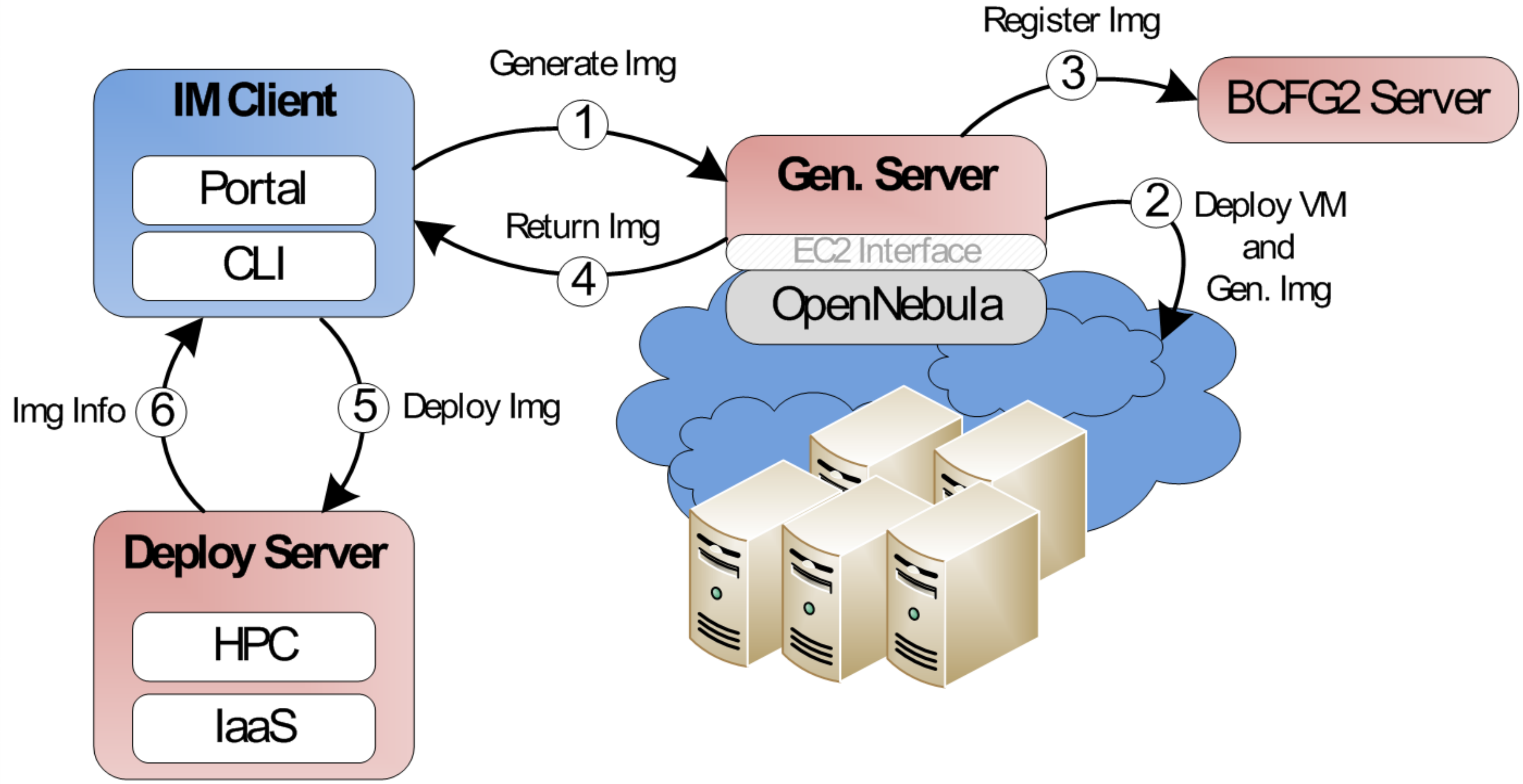
Image Generation and Management on FutureGrid: A practical Example

Presented by

Gregor von Laszewski

Technology Preview

Deployed Infrastructure



Future Grid

Technology Preview

Generating an Image (I)

- Generate an Centos image with several packages
  - **fg-image-generate-client.py –o centos –v 5.6 –a x86_64 –s emacs, openmpi –u javi**
- The output is a tgz file that contains
  - Image file
  - Manifest file
    - Username, image name, os, architecture, package list

Technology Preview

Generate Image (II)

- Log Client side:

```
2011-05-17 20:32:42,727 - root - INFO - Image generator client...
2011-05-17 20:32:42,728 - root - INFO - ssh fg-image-gen-server '/srv/cloud/one/fg-
management/fg-image-generate-server.py -a x86_64 -o centos -v 5.6 -u javi –s emacs,
openmpi ' > /tmp/1305678762.733906384900
2011-05-17 20:39:35,171 - root - INFO - Status: /srv/scratch/javi-3058834494.tgz
2011-05-17 20:39:35,171 - root - INFO - Retrieving the image
2011-05-17 20:39:35,171 - root - INFO - scp fg-image-gen-server:/srv/scratch/
javi-3058834494.tgz .
2011-05-17 20:46:21,864 - root - INFO – The image is placed in /home/javi/
javi-3058834494.tgz
2011-05-17 20:46:21,864 - root - INFO - Post processing
2011-05-17 20:46:21,864 - root - INFO - ssh fg-image-gen-server rm -f /srv/scratch/
javi-3058834494.tgz > /tmp/1305679581.862429881758
```

Technology Preview

Generate Image (III)

- Log Server side:

```
2011-05-13 14:50:36,229 - root - INFO - Image generator server...
2011-05-13 14:50:36,229 - root - INFO - The VM deployed is in 192.168.1.24
2011-05-13 14:50:36,229 - root - INFO - Mount scratch directory in the VM
2011-05-13 14:50:36,229 - root - INFO - ssh -q root@192.168.1.24 mount -t nfs
192.168.1.6:/srv/scratch/ /media/
2011-05-13 14:50:36,535 - root - INFO - Sending fg-image-generate.py to the VM
2011-05-13 14:50:36,535 - root - INFO - scp -q /srv/cloud/one/fg-management/fg-image-
generate.py  root@192.168.1.23:/root/
2011-05-13 14:50:36,877 - root - INFO - ssh root@192.168.1.24 -q '/root/fg-image-
generate.py -a x86_64 -o centos -v 5.6 -u javi –s emacs, openmpi -t /media/ ' > /tmp/
1305312636.882202999127
2011-05-13 14:55:49,158 - root - INFO - Umount scratch directory in the VM
```

Technology Preview

Generate Image (IV)

- Log VM side:

```
2011-05-17 21:52:55,065 - root - INFO - Starting image generator...
2011-05-17 21:52:55,065 - root - INFO - Building Centos 5.6 image
2011-05-17 21:52:55,065 - centos - INFO - Generation Image: centos-5.6-x86_64-base.img
2011-05-17 21:52:55,065 - centos - INFO - Creating Disk for the image
2011-05-17 21:52:58,752 - centos - INFO - Mounting new image
2011-05-17 21:52:58,800 - centos - INFO - Getting appropiate release package
2011-05-17 21:52:58,801 - exec - DEBUG - wget http://mirror.centos.org/centos/5.6/os/
x86_64/CentOS/centos-release-5-6.el5.centos.1.x86_64.rpm -O /media/centos-release.rpm
2011-05-17 21:53:00,414 - exec - DEBUG - rpm -ihv --nodeps --root /media/
javi-3058834494 /media/centos-release.rpm
2011-05-17 21:53:00,645 - exec - DEBUG - yum --installroot=/media/javi-3058834494 -y
groupinstall Core
```

Technology
Preview

Generate Image (V)

- Log VM side (cont.):

```
2011-05-17 21:57:24,786 - centos - INFO - Installing LDAP packages
2011-05-17 21:57:48,159 - centos - INFO - Configuring LDAP access
2011-05-17 21:57:48,390 - centos - INFO - Injected networking configuration
2011-05-17 21:57:48,391 - centos - INFO - Installing BCFG2 client
2011-05-17 21:57:48,391 - centos – INFO - Configured BCFG2 client settings
2011-05-17 21:57:48,812 - centos - INFO - Installing user-defined packages
2011-05-17 21:57:52,689 - centos - INFO - Genereated centos image javi-3058834494
successfully!
2011-05-17 21:57:55,344 - manifest - INFO - Genereated manifest file: /media/
javi-3058834494 .manifest.xml
```

Technology Preview

Image Deployment

- Deploy the VM for HPC (xCAT)
  -
    ```
    ./fg-image-deploy.py -x tm1r -s th1r -t /media/disk/scratch -i javi-3058834494.tgz -u jdiaz
    ```
- Output
  - The image is deployed and register in xCAT
  - The image is available for dynamic provisioning
    - qsub –l os=imagename job.sh

Technology Preview

Image Deployment (II)

- Log Client side:

```
2011-05-16 12:31:01,196 - root - INFO - Starting image deployer...
2011-05-16 12:31:01,197 - root - INFO - untar file with image and manifest
2011-05-16 12:31:16,028 - root - INFO - Using image: javi-3058834494.img
2011-05-16 12:31:16,029 - root - INFO - Mounting image...
2011-05-16 12:31:16,029 - exec - DEBUG - mkdir -p /tmp/javi-3058834494//rootimg
2011-05-16 12:31:16,032 - exec - DEBUG - sudo mount -o loop javi-3058834494.img /tmp/
javi-3058834494//rootimg/
2011-05-16 12:31:16,042 - root - INFO - Installing torque
2011-05-16 12:31:18,070 - root - INFO - Injected kernel 2.6.18-164.el5
2011-05-16 12:31:18,076 - root - INFO - Injected fstab
2011-05-16 12:31:18,076 - root - INFO - Compressing image
2011-05-16 12:32:52,039 - exec - INFO - Umounting image...
2011-05-16 12:32:52,282 - root - INFO - Uploading image.
2011-05-16 12:32:52,283 - exec - DEBUG - scp /tmp/javi-3058834494/rootimg.gz
jdiaz@th1r:/media/disk/scratch/javi-3058834494.gz
```

Technology
Preview

Image Deployment (III)

- Log Server side:

```
2011-05-16 17:08:14,525 - root - INFO - Accepted new connection
2011-05-16 17:08:14,525 - exec - DEBUG - mkdir -p /install/netboot/centos.javi.
3058834494/x86_64/compute/
2011-05-16 17:08:14,527 - exec - DEBUG - mv /media/disk/scratch/javi-3058834494.gz /
install/netboot/centos.javi. 3058834494/x86_64/compute/rootimg.gz
2011-05-16 17:08:14,563 - exec - DEBUG - mkdir -p /install/netboot/centos.javi.
3058834494/x86_64/compute/rootimg
2011-05-16 17:08:19,615 - exec – INFO – Get Kernel and Initrd
2011-05-16 17:08:19,808 - exec - DEBUG - packimage -o centos.javi. 3058834494 -p
compute -a x86_64
2011-05-16 17:08:46,279 - exec - DEBUG - stdout: Packing contents of /install/netboot/
centos.javi. 3058834494/x86_64/compute/rootimg
2011-05-16 17:08:48,279 - exec – INFO – Register image in Moab (/opt/moab/tools/msm/
images.txt)
2011-05-16 17:08:48,387 - exec - DEBUG - mschedctl -R
```

Boot Image using xCAT via Nodeset

- nodeset tc1 netboot=centos.javi.3058834494-x86_64-compute
- rpower tc1 boot
- Output
  - The image is booted in tc1 machine
- Check status
  - nodestat tc1
  - rcons tc1

Boot Image using Moab/xCAT

- qsub -l os=centos.javi.3058834494-x86_64-compute testjob.sh
- Output
  - The image is booted in a machine
- Check status
  - showq, checkjob <jobid>

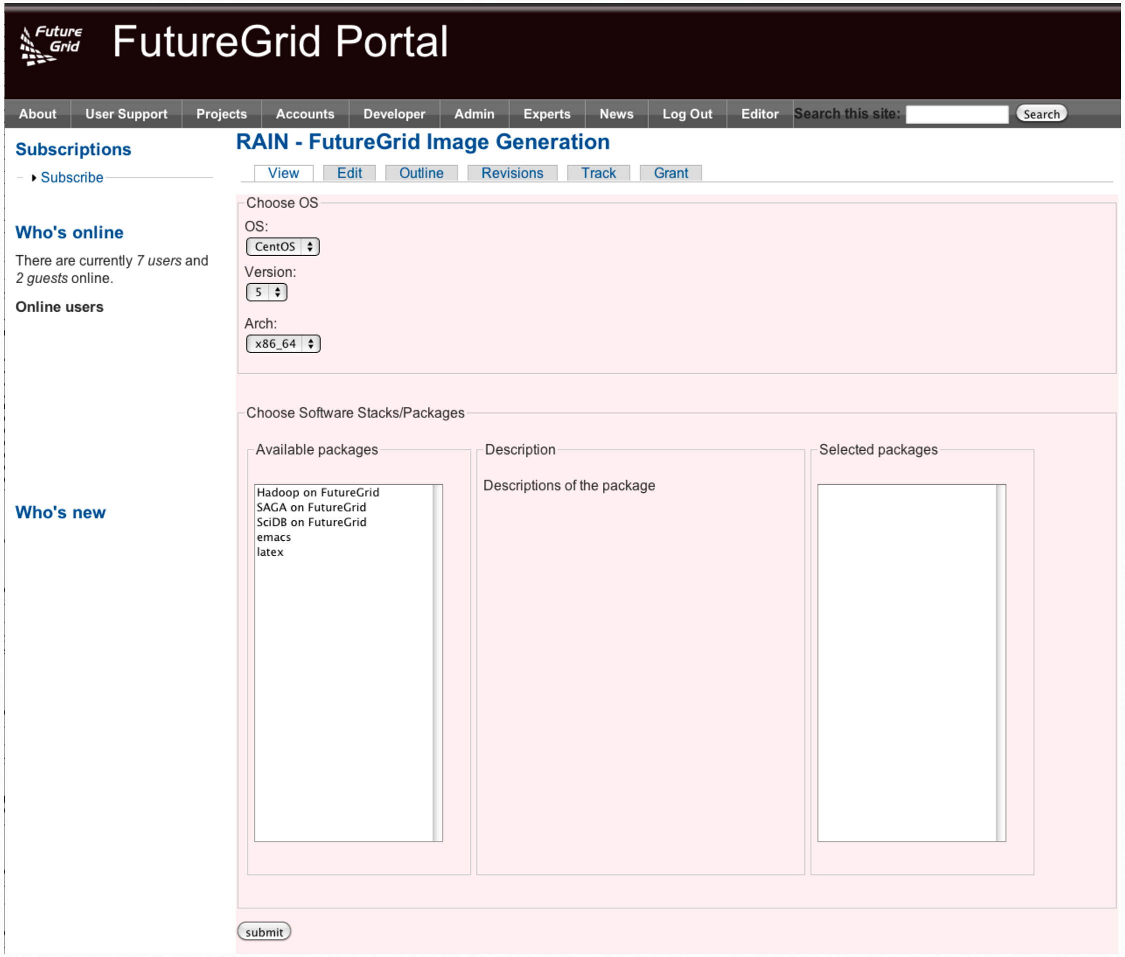
Technology Preview

Image Generation with the Portal

Future Grid

Technology Preview

Image Generation with the Portal



Subscriptions

- Subscribe

Who's online

There are currently *7 users* and *2 guests* online.

Online users

Who's new

RAIN - FutureGrid Image Generation

Choose OS

OS: CentOS

Version: 5

Arch: x86_64

Choose Software Stacks/Packages

Available packages:

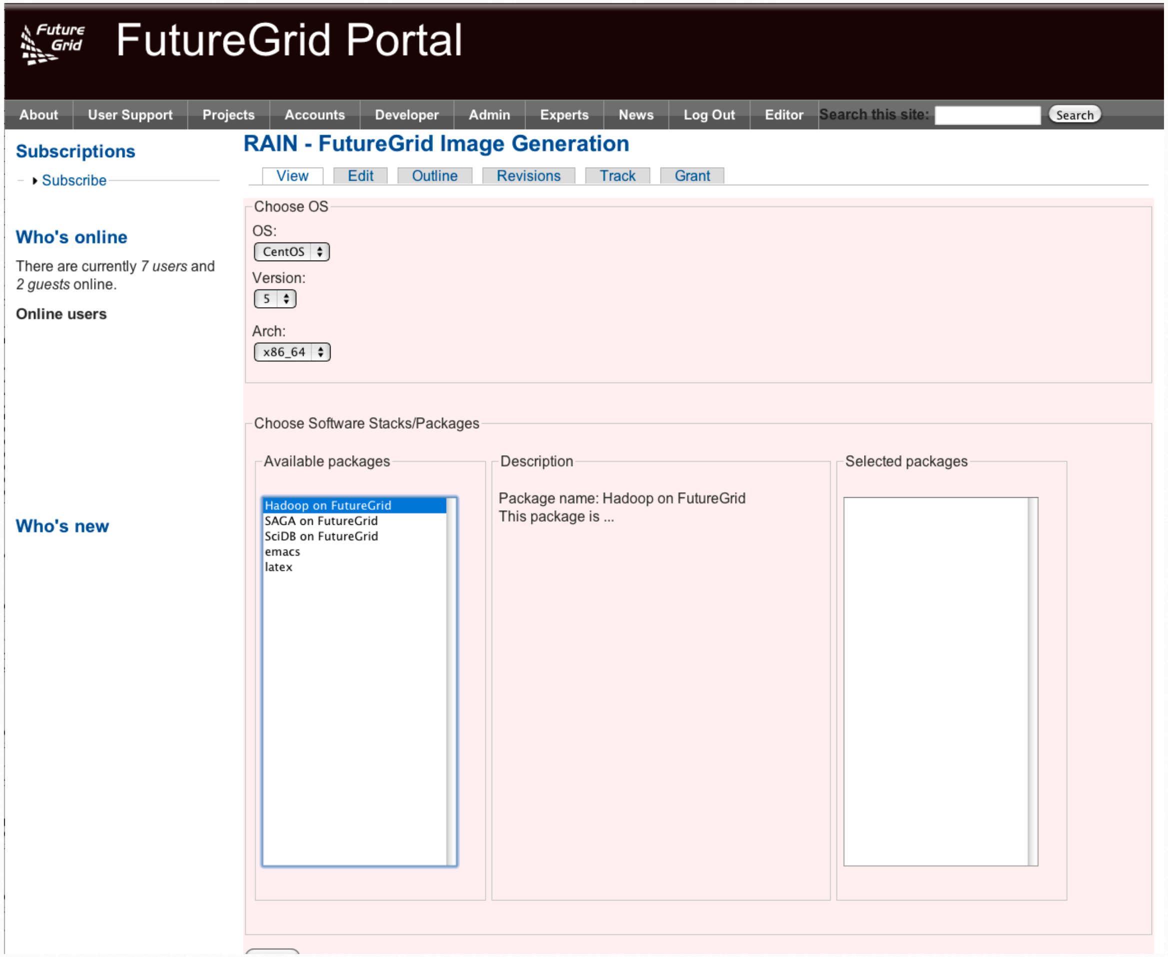
- Hadoop on FutureGrid
- SAGA on FutureGrid
- SciDB on FutureGrid
- emacs
- latex

Description: Descriptions of the package

Selected packages

Technology Preview

Image Generation with the Portal



FutureGrid Portal

Subscriptions

- Subscribe

Who's online

There are currently *7 users* and *2 guests* online.

Online users

Who's new

RAIN - FutureGrid Image Generation

Choose OS

OS: CentOS

Version: 5

Arch: x86_64

Choose Software Stacks/Packages

Available packages

- Hadoop on FutureGrid
- SAGA on FutureGrid
- SciDB on FutureGrid
- emacs
- latex

Description

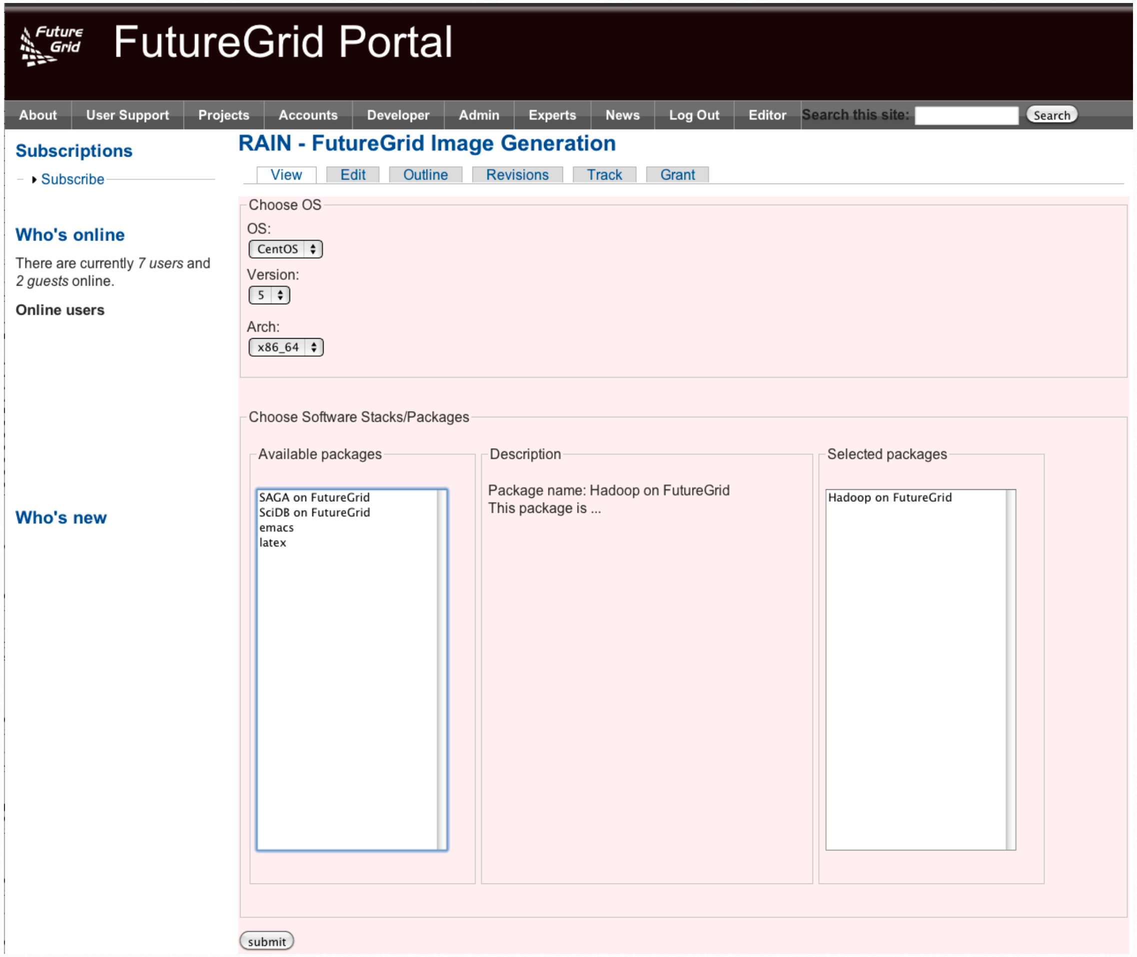
Package name: Hadoop on FutureGrid
This package is ...

Selected packages

Future Grid

Technology Preview

Image Generation with the Portal



RAIN - FutureGrid Image Generation

Who's online

There are currently *7 users* and *2 guests* online.

Online users

Who's new

Choose OS

OS: CentOS

Version: 5

Arch: x86_64

Choose Software Stacks/Packages

Available packages:
SAGA on FutureGrid
SciDB on FutureGrid
emacs
latex

Description:
Package name: Hadoop on FutureGrid
This package is ...

Selected packages:
Hadoop on FutureGrid

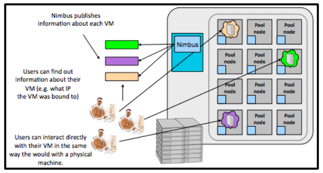
CCGrid 2011 The 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, Newport Beach, CA, USA.

May 23th -26th, 2011.

FutureGrid Tutorial: An Introduction to Nimbus

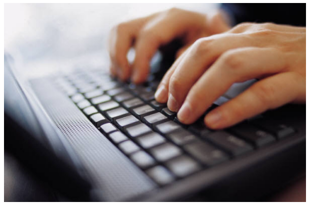
Kate Keahey, David LaBissoniere,
John Bresnahan, Tim Freeman,
Patrick Armstrong, Paul Marshall
Argonne National Laboratory
Computation Institute, University of Chicago

An Introduction to Nimbus


Nimbus publishes information about each VM
Users can find out information about their VM (e.g. what IP the VM was bound to)
Users can interact directly with their VM in the same way the would with a physical machine.
Nimbus
Pool node

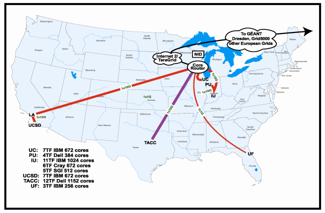
Overview of the Nimbus Project

- How Nimbus works
- Software, Features, and Community



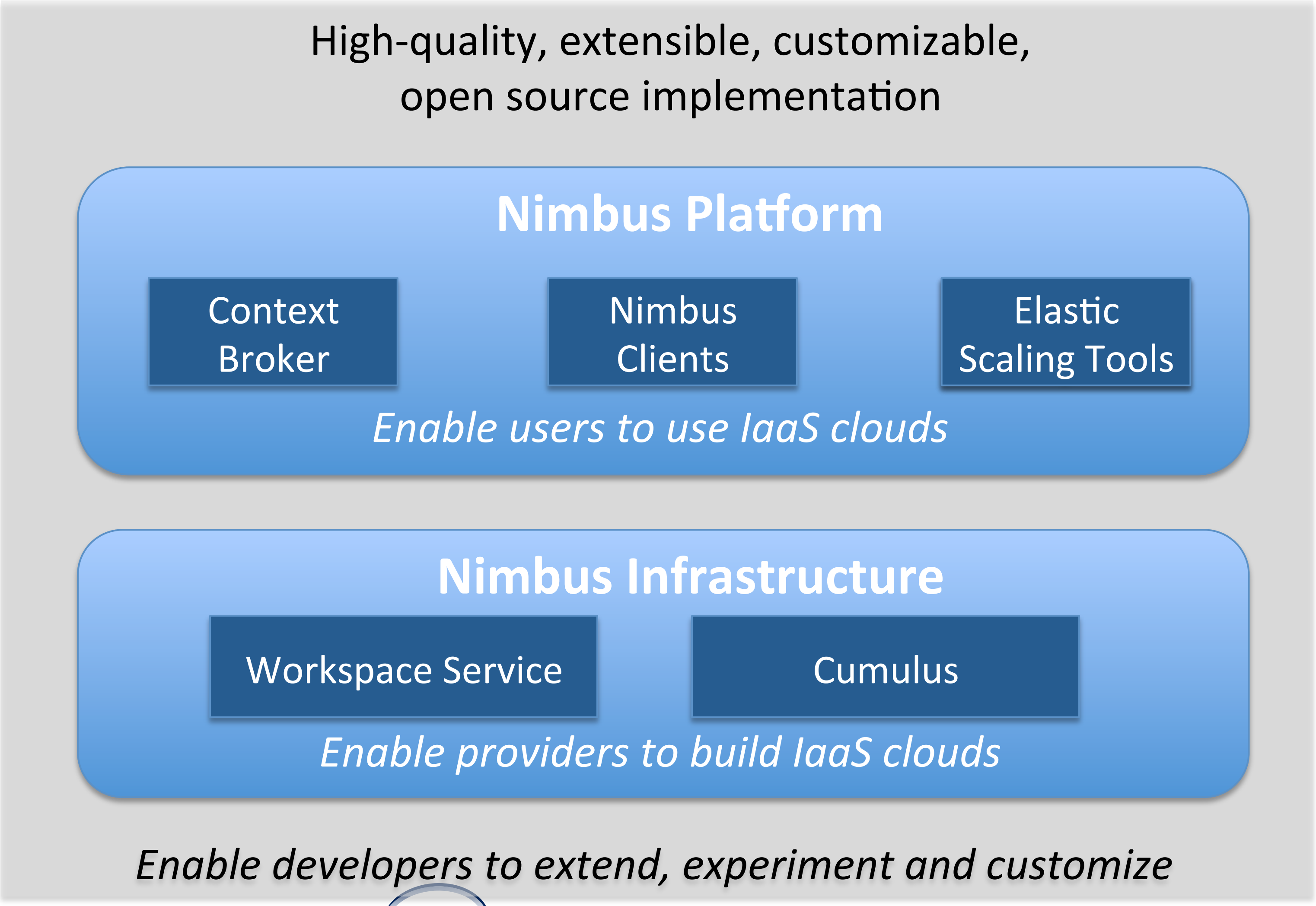
Hands-on Tutorial Exercises

- Download Nimbus cloud client
- Connect to Nimbus on FutureGrid
- Launch VMs!



FutureGrid Nimbus Use Case

Nimbus Components



High-quality, extensible, customizable,
open source implementation

Nimbus Platform

Context Broker

Nimbus Clients

Elastic Scaling Tools

Enable users to use IaaS clouds

Nimbus Infrastructure

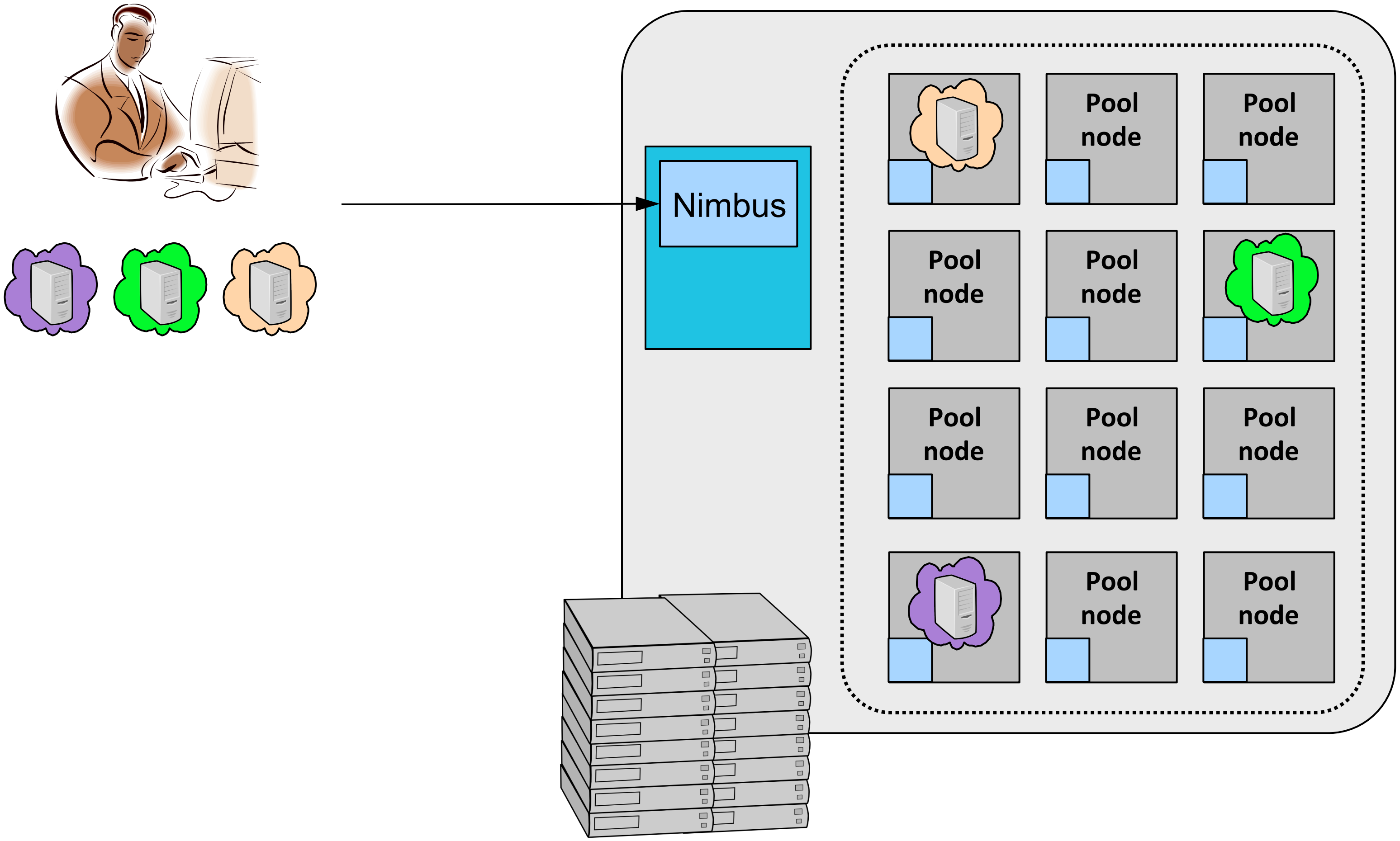
Workspace Service

Cumulus

Enable providers to build IaaS clouds

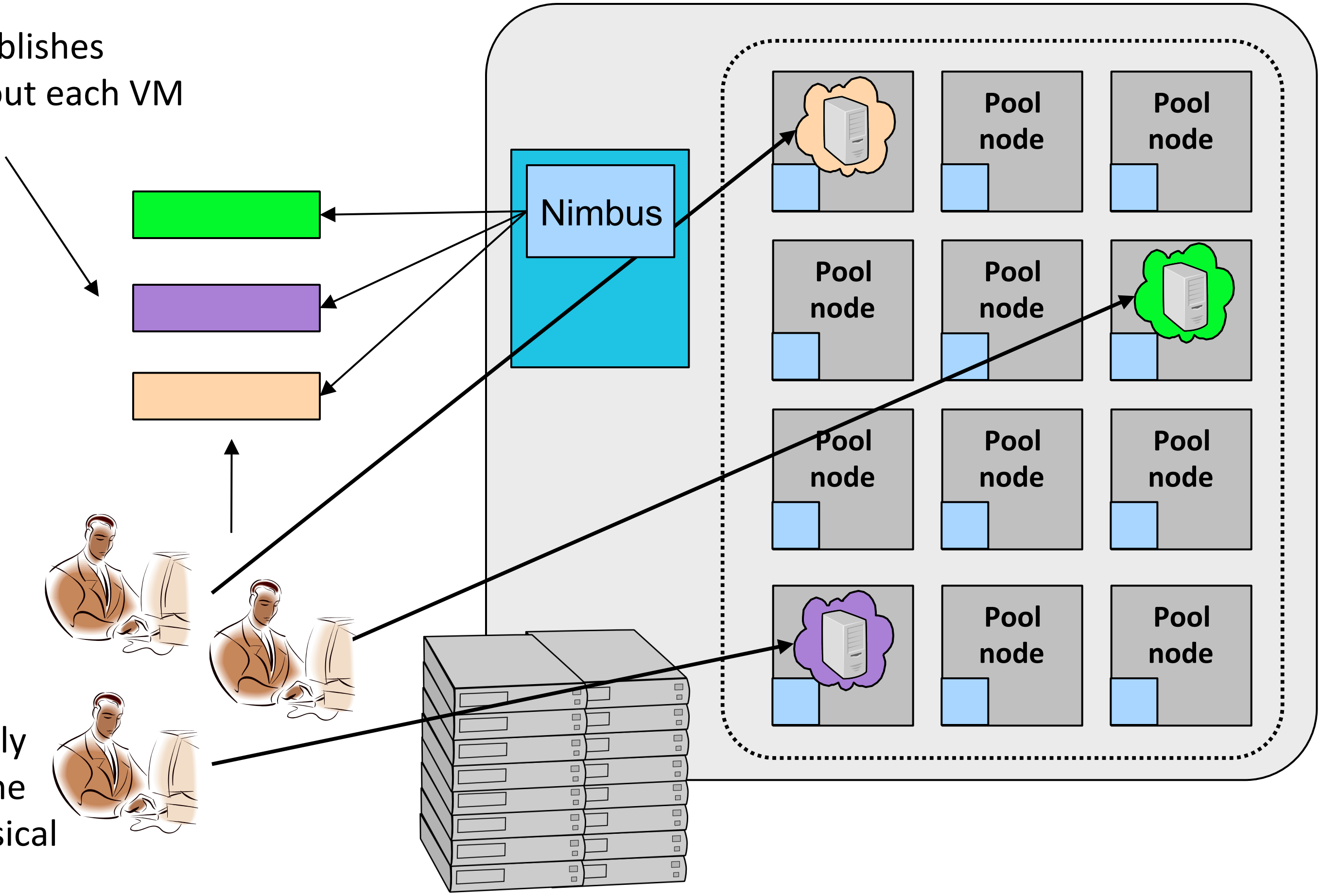
Enable developers to extend, experiment and customize

Nimbus IaaS: How it Works



Nimbus IaaS: How it Works

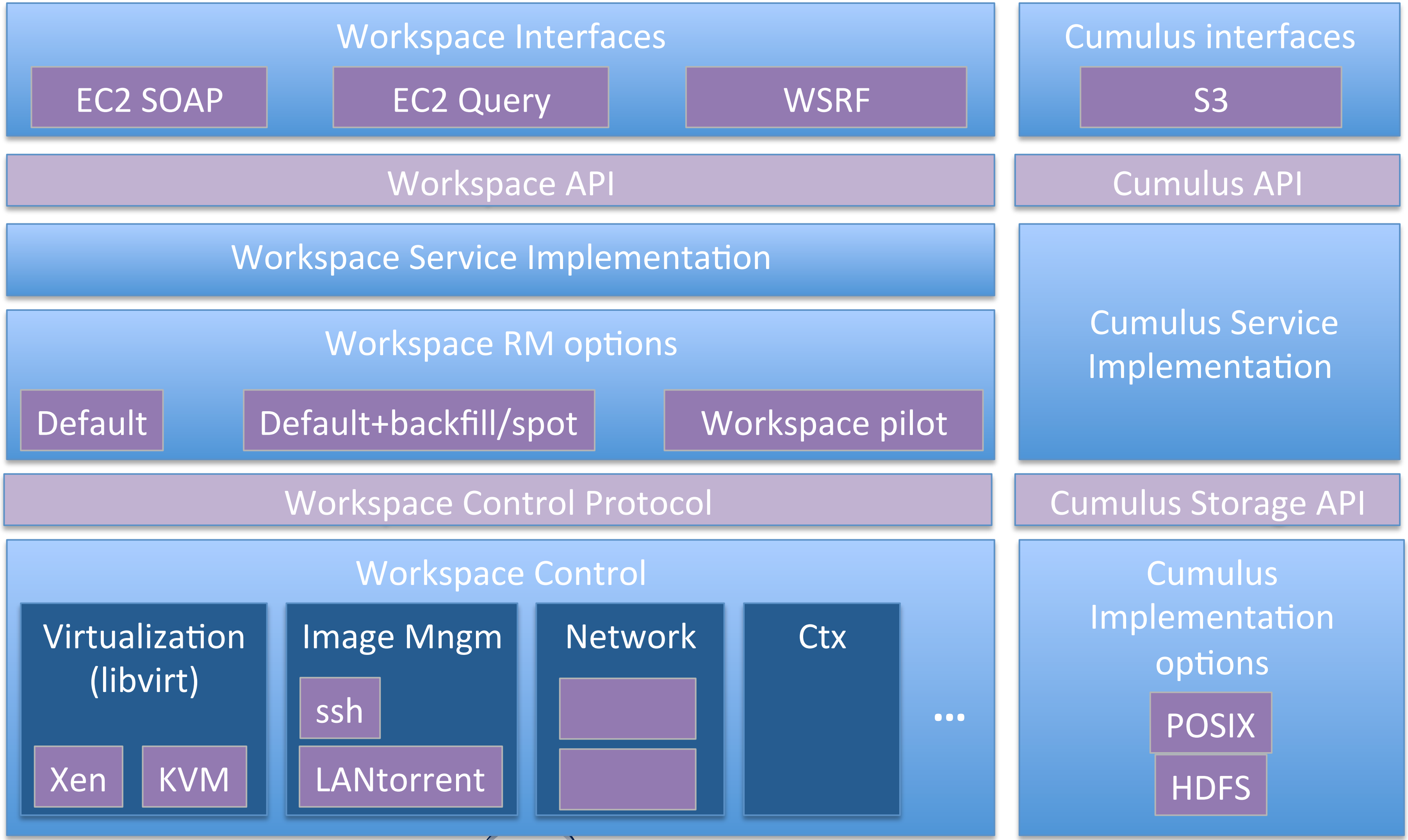
Nimbus publishes information about each VM



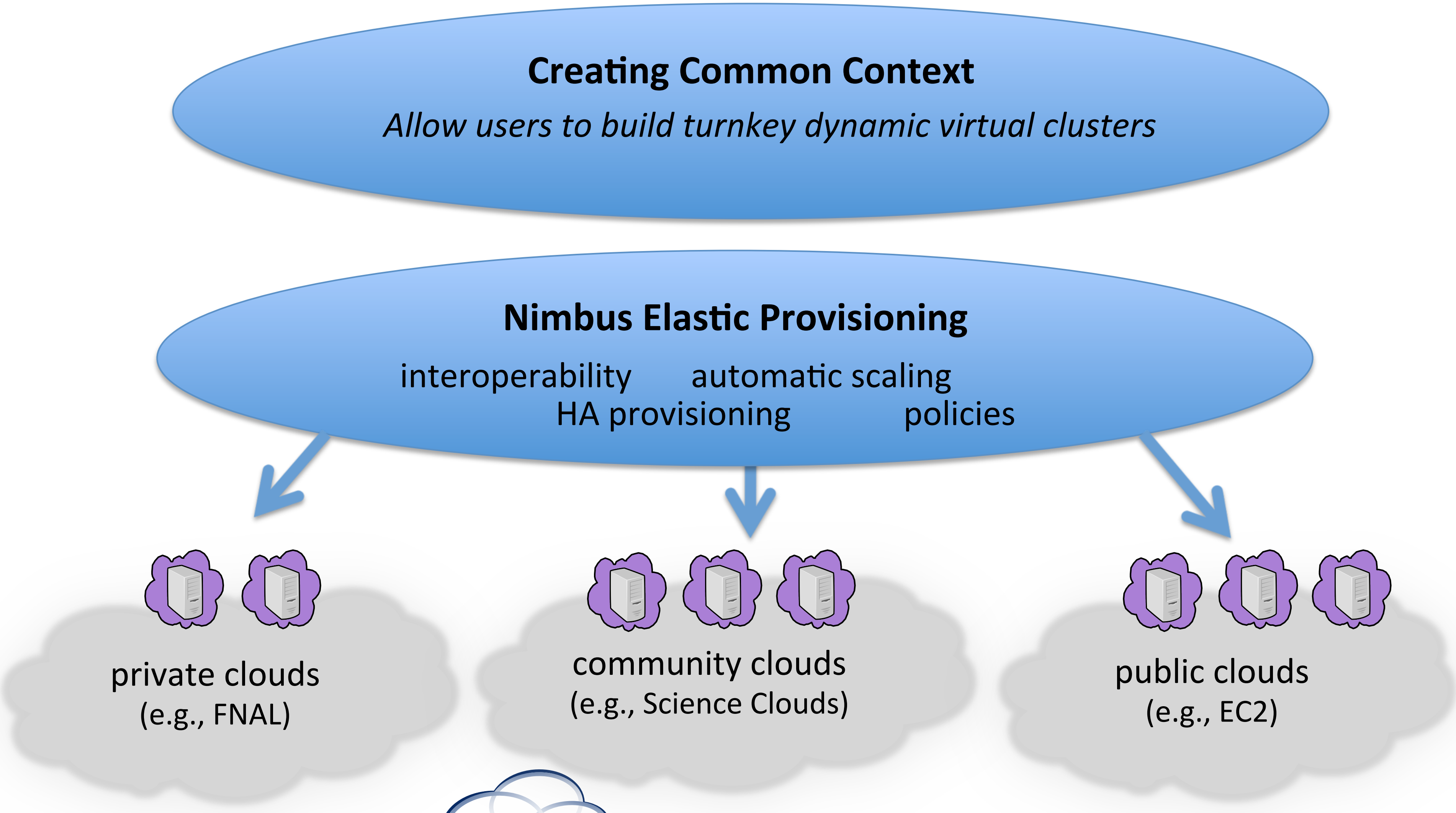
Users can find out information about their VM (e.g. what IP the VM was bound to)

Users can interact directly with their VM in the same way the would with a physical machine.

Nimbus Infrastructure: a Highly-Configurable IaaS Architecture



Nimbus Platform: Working with Hybrid Clouds



Creating Common Context

Allow users to build turnkey dynamic virtual clusters

Nimbus Elastic Provisioning

interoperability automatic scaling HA provisioning policies

private clouds (e.g., FNAL)

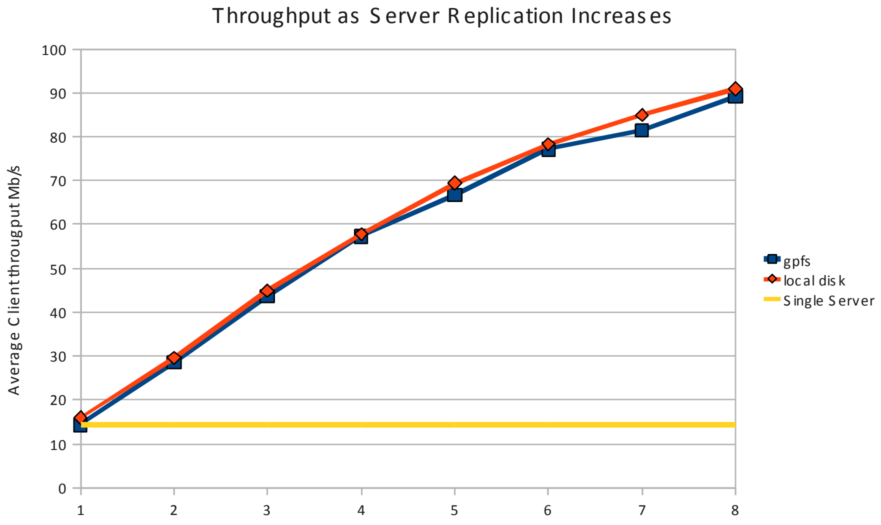
community clouds (e.g., Science Clouds)

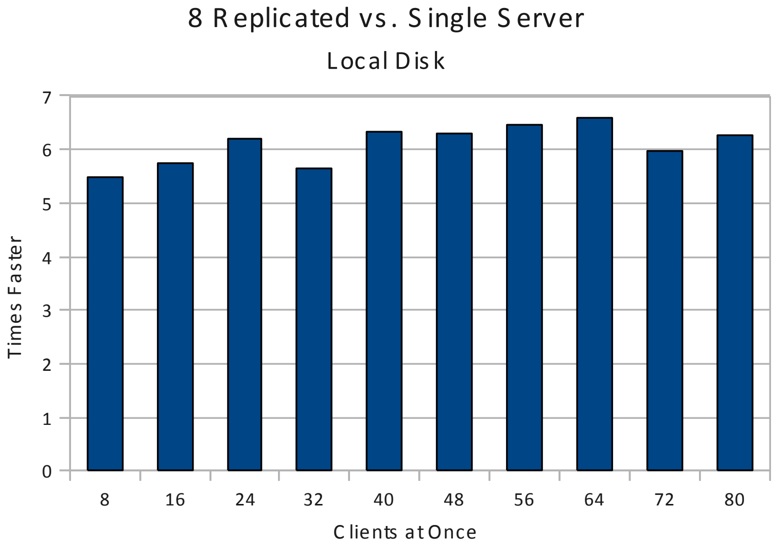
public clouds (e.g., EC2)

Nimbus Infrastructure Highlights

Cumulus: a Scalable Storage Cloud

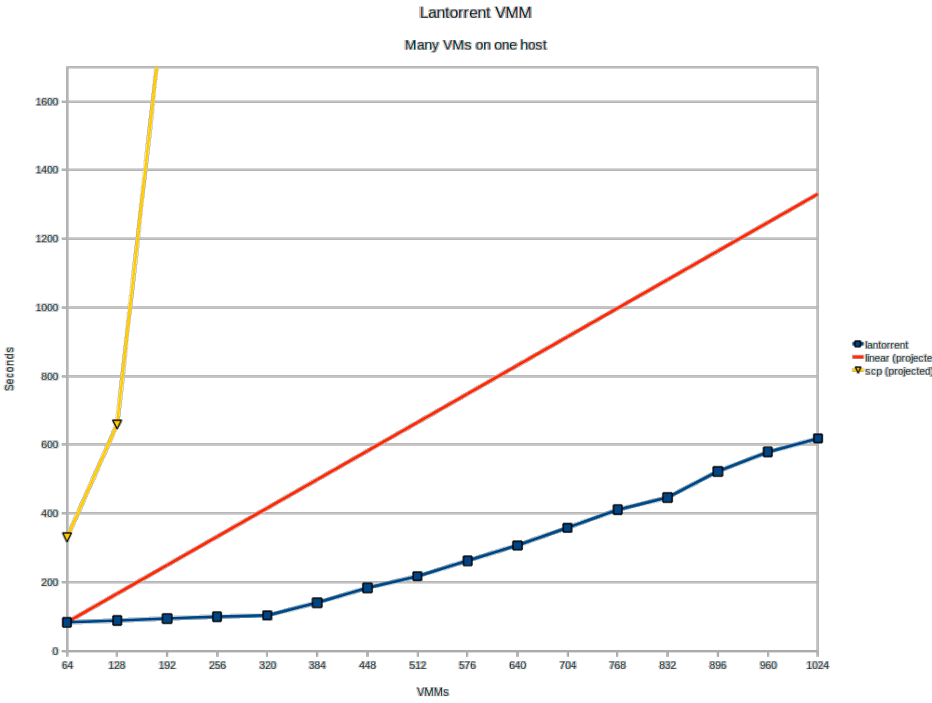
- **Challenge:** a scalable storage cloud
- S3-compatible open source implementation
- Quota support for scientific users
- Pluggable back-end to various technologies such as POSIX, HDFS, Sector, BlobSeer
- Configurable to take advantage of multiple servers
- John Bresnahan will present a paper at ScienceCloud '11 (HPDC)





LANTorrent: Fast Image Deployment

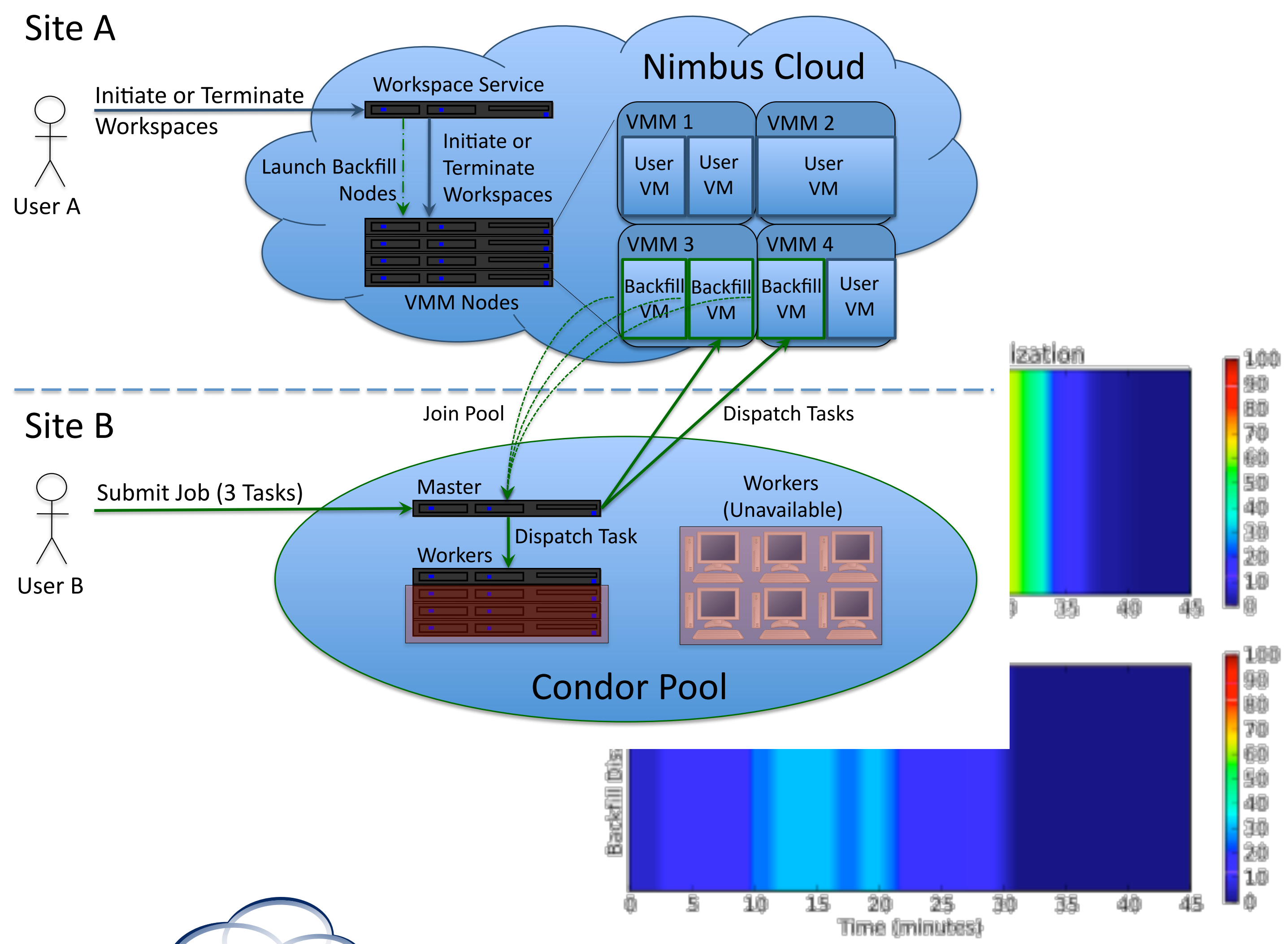
- **Challenge:** make image deployment faster
- Moving images is the main component of VM deployment
- LANTorrent: the BitTorrent principle on a LAN
- Streaming
- Minimizes congestion at the switch
- Detecting and eliminating duplicate transfers
- **Bottom line:** a thousand VMs in 10 minutes
- Nimbus release 2.6



Preliminary data using the Magellan resource
At Argonne National Laboratory

Backfill: Lower the Cost of Your Cloud

- **Challenge:** utilization
  catch-
  compu
- Solutic
  – Bacl
  – Spot
- **Bottoı**
  100%
- Nimbu



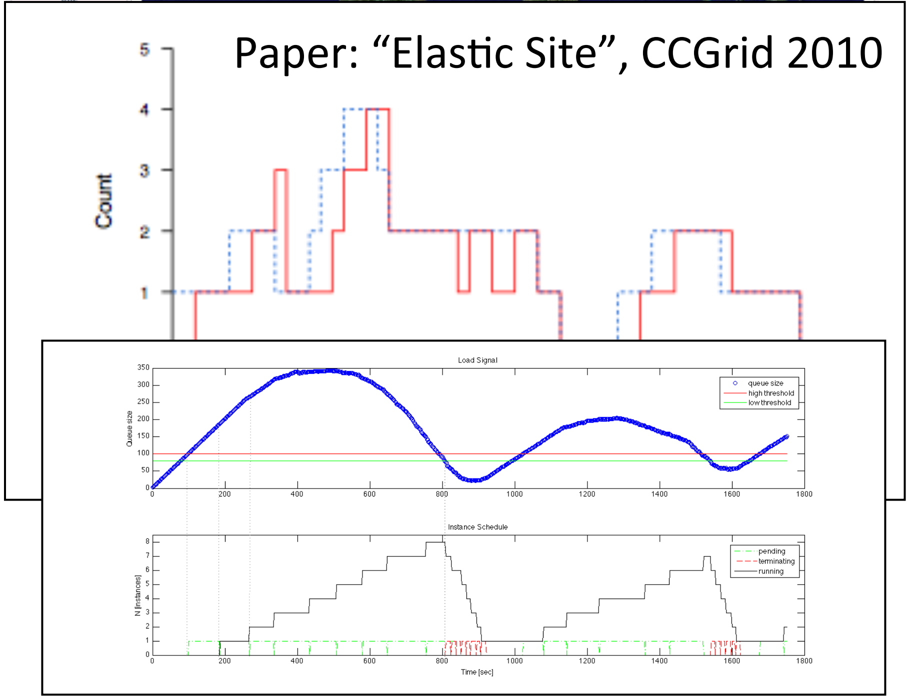
Nimbus Platform Highlights: Coming Down the Assembly Line

Elasticity, Reliability and Failure

Elasticity and reliability are different sides of the same coin.

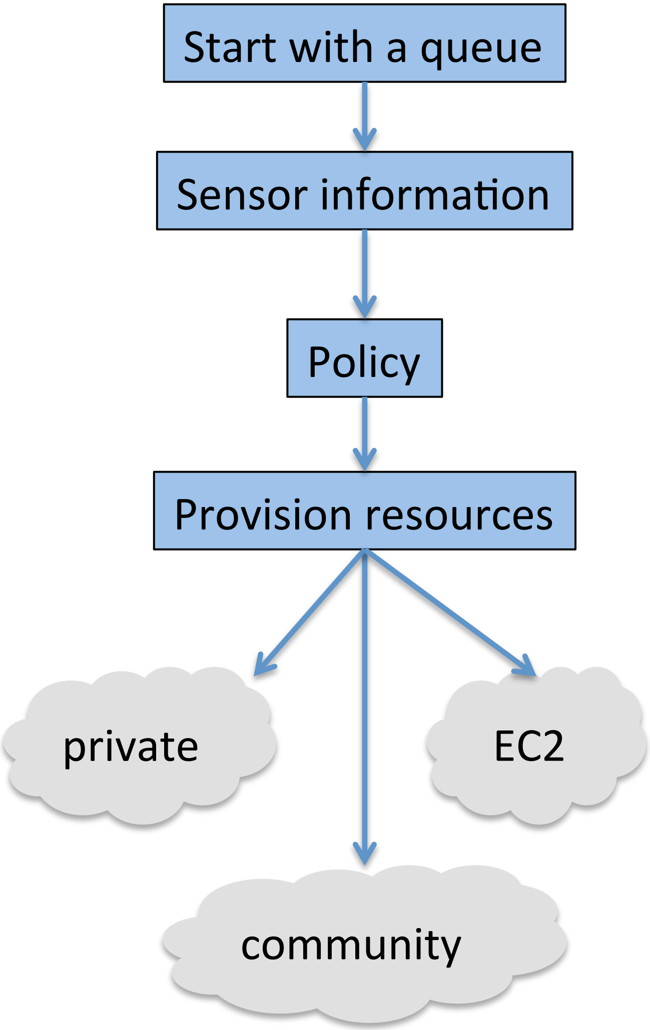
- 2008: The ALICE proof-of-concept
- 2009: ElasticSite prototype
- 2009: OOI pilot

Need for generic, HA, elastic service model


Paper: “Elastic Site”, CCGrid 2010

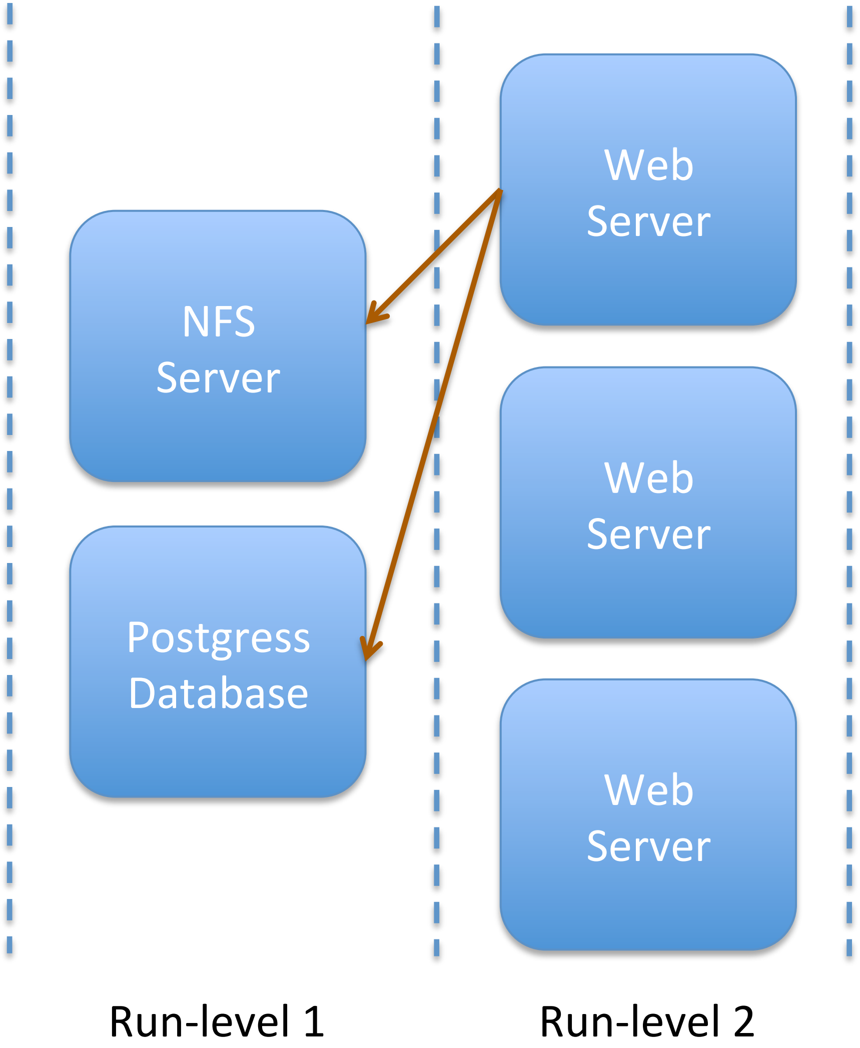
Elasticity, Reliability and Failure

- Assumption: a workload queue
  - ALiEn, PBS, AMQP,…
- React to sensor information
  - Queue properties a sensor
- Scale to demand
  - Across different cloud providers
  - Use contextualization to integrate machines across hybrid clouds
  - Highly Available
  - Scalable: latest tests scale to 100s of nodes on EC2, target is thousands
- ***Coming in Nimbus 3***



Cloudinit.d

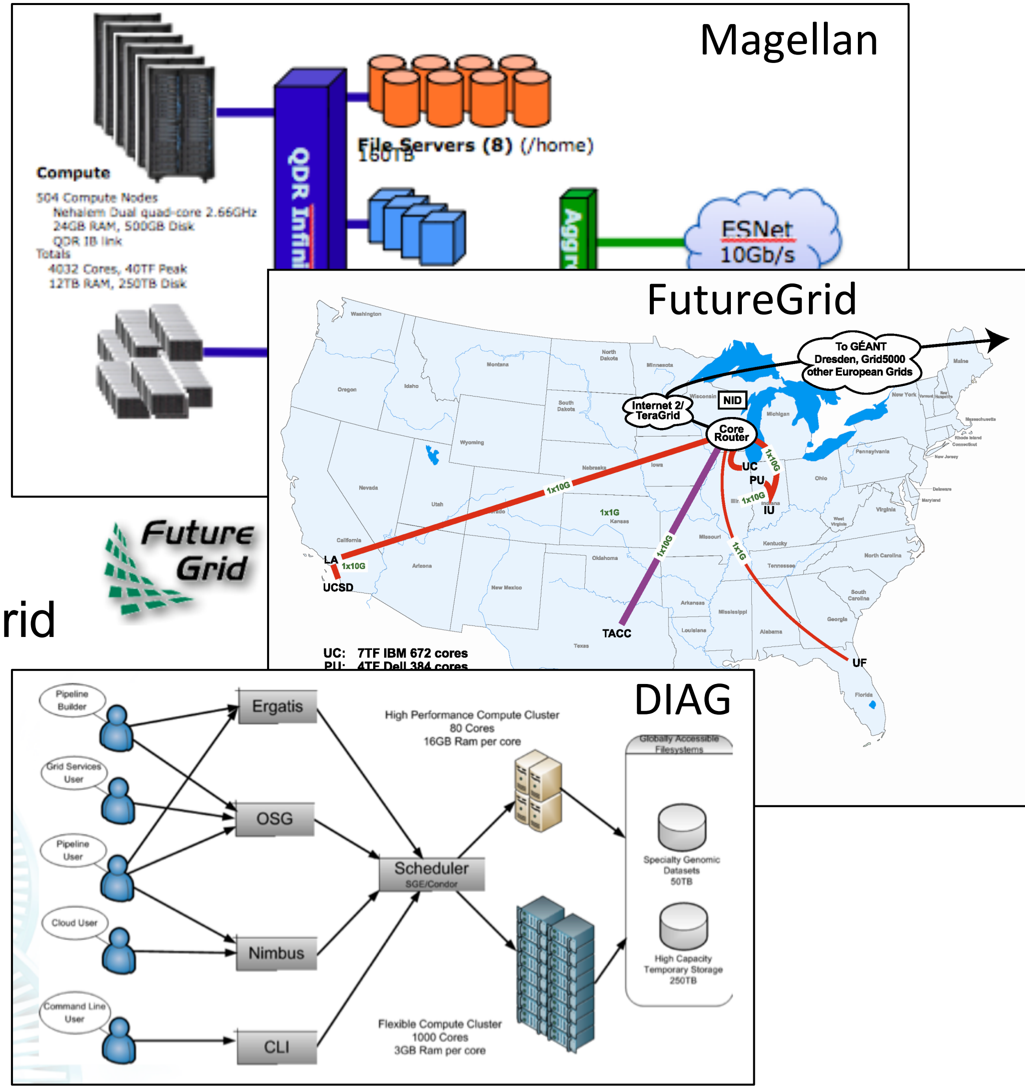
- Repeatable deployment of sets of VMs
- Coordinates launches via attributes
- Works with multiple IaaS providers
- User-defined launch tests (assertions)
- Test-based monitoring
- Policy-driven repair of a launch
- ***Coming in Nimbus 3***



Resources, Applications and Ecosystem

Scientific Cloud Resources

- Science Clouds
  - UC, UFL, Wispy@Purdue
  - ~300 cores
- Magellan
  - DOE cloud @ ANL&LBNL
  - ~4000 cores@ANL
- FutureGrid
  - ~6000 cores
- DIAG =
  - Data Intensive Academic Grid
  - U of Maryland School of Medicine in Baltimore
  - ~1200-1500 cores
- Outside of US:
  - WestGrid, Grid5000

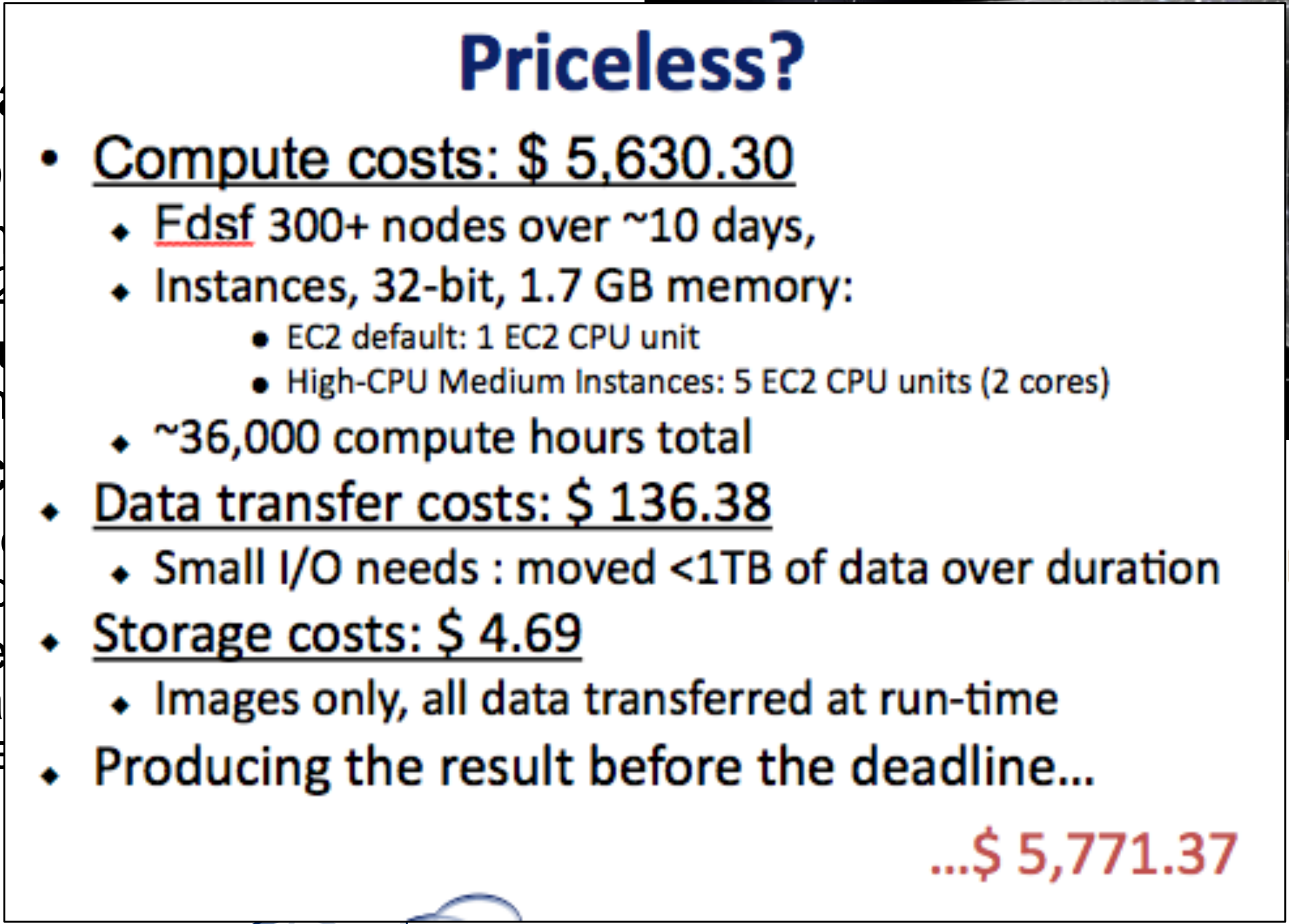


Work by Jerome Lauret (BNL) et al.

- STAR: a nuclear physics experiment at Brook Labor
- Appro
  - Nim EC2
  - Virt Nim
- Impac
  - Pro sinc
  - The dea time

Afterglow Light Pattern 400,000 yrs.
Dark Ages
Development of Galaxies, Planets, etc.
Dark Energy Accelerated Expansion

Made Easy



Priceless?

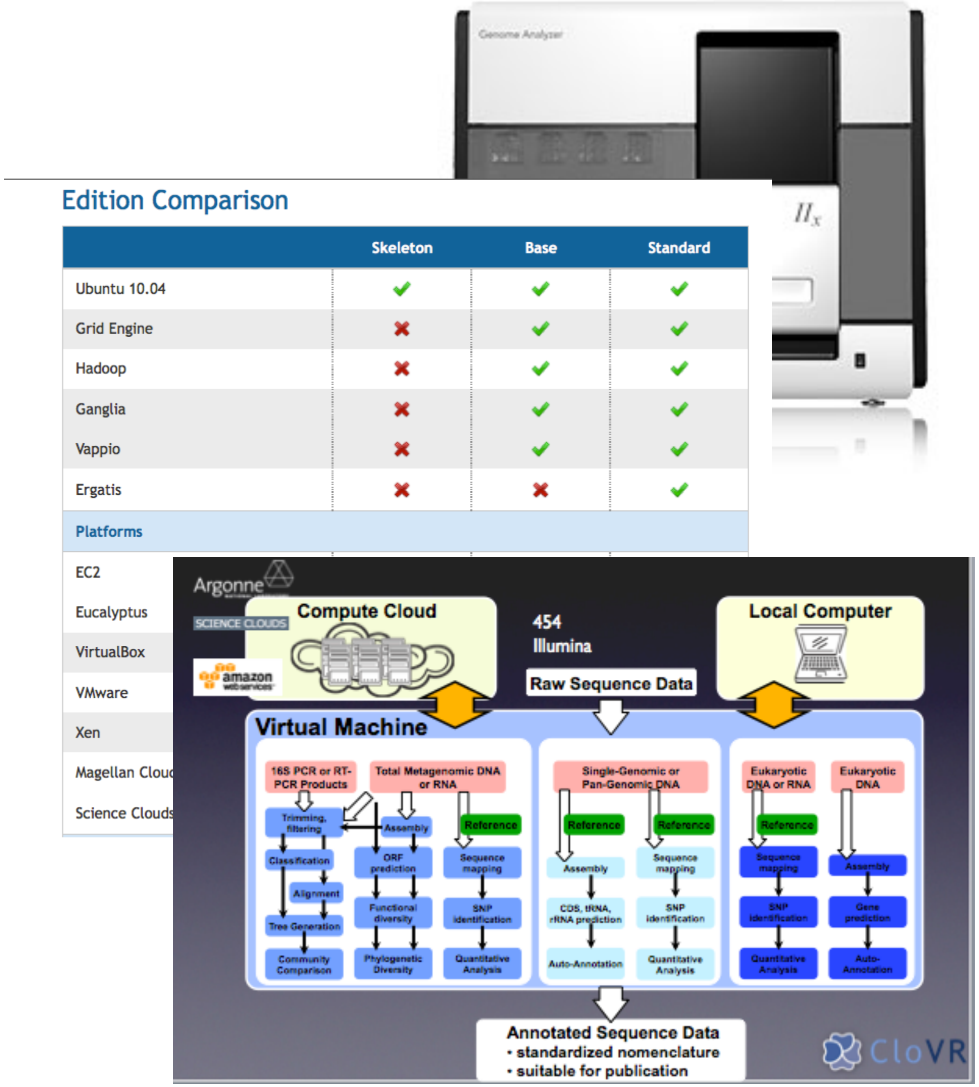
- Compute costs: $ 5,630.30
  - Fdsf 300+ nodes over ~10 days,
  - Instances, 32-bit, 1.7 GB memory:
    - EC2 default: 1 EC2 CPU unit
    - High-CPU Medium Instances: 5 EC2 CPU units (2 cores)
  - ~36,000 compute hours total
- Data transfer costs: $ 136.38
  - Small I/O needs : moved <1TB of data over duration
- Storage costs: $ 4.69
  - Images only, all data transferred at run-time
- Producing the result before the deadline…

…$ 5,771.37

CloVR

Sam Angiuoli
Institute for Genome Sciences
University of Maryland School of Medicine

- The emergent need for processing
- A virtual appliance for automated and portable sequence analysis
- Approach:
  - Running on Nimbus Science Clouds, Magellan and EC2
  - A platform for building appliances representing push-button pipelines
- Impact
  - From desktop to cloud
  - http://clovr.org



Edition Comparison

	Skeleton	Base	Standard
Ubuntu 10.04	✓	✓	✓
Grid Engine	✗	✓	✓
Hadoop	✗	✓	✓
Ganglia	✗	✓	✓
Vappio	✗	✓	✓
Ergatis	✗	✗	✓
Platforms			
EC2			
Eucalyptus			
VirtualBox			
VMware			
Xen			
Magellan Clou			
Science Clouds			

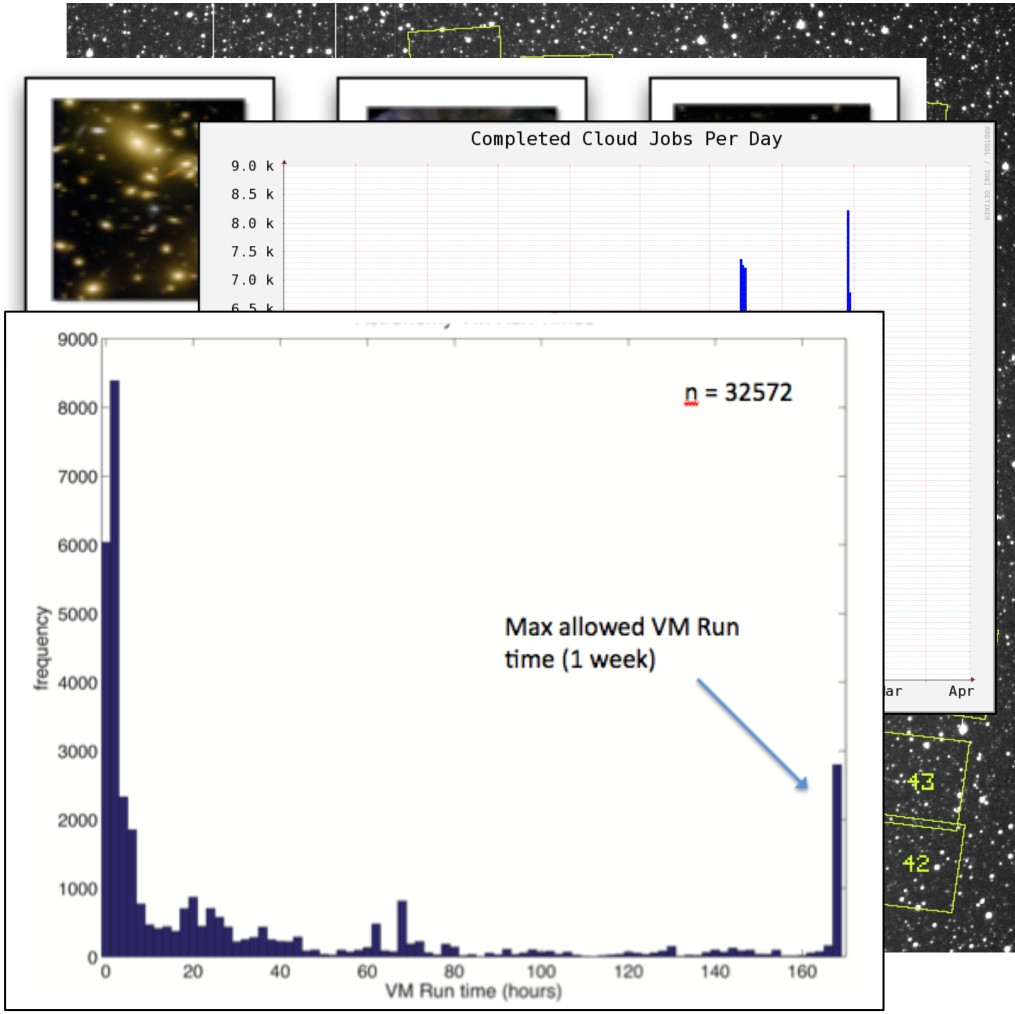
CANFAR

Canadian Advanced Network for Astronomical Research

University of Victoria NRC·CNRC

Work by the UVIC team

- Detailed analysis of data from the MACHO experiment Dark Matter search
- Provide infrastructure for six observational astronomy survey projects
- Approach:
  - Running on a Nimbus cloud on WestGrid
  - Appliance creation and management
  - Dynamic Condor pool for astronomy
- Status:
  - In production operation since July 2010

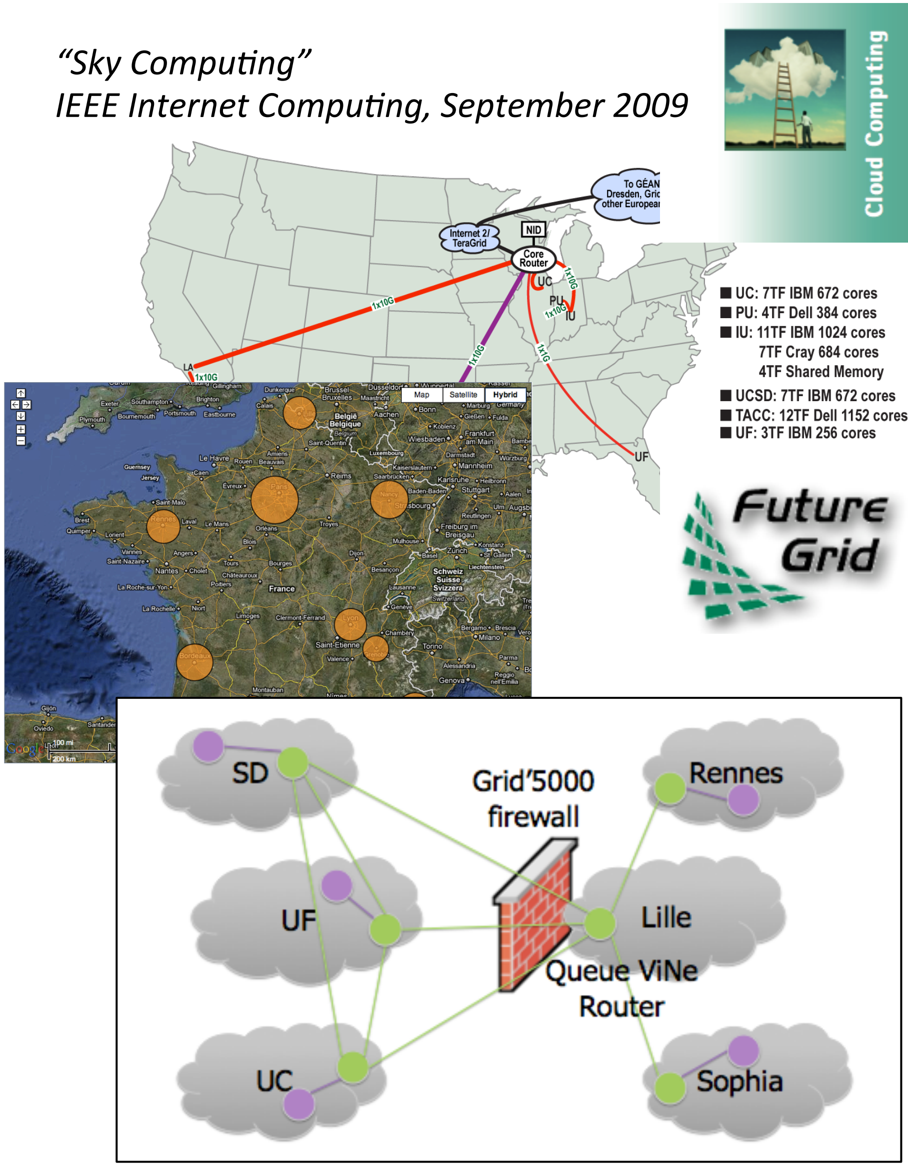


Sky Computing

Work by Pierre Riteau et al,
University of Rennes 1

- Sky Computing = a Federation of Clouds
- Approach:
  - Combine resources obtained in multiple Nimbus clouds in FutureGrid and Grid' 5000
  - Combine Context Broker, ViNe, fast image deployment
  - Deployed a virtual cluster of over 1000 cores on Grid5000 and FutureGrid – largest ever of this type
- Grid'5000 Large Scale Deployment Challenge award
- Demonstrated at OGF 29 06/10
- TeraGrid '10 poster
- More at: *www.isgtw.org/?pid=1002832*

"Sky Computing"
IEEE Internet Computing, September 2009



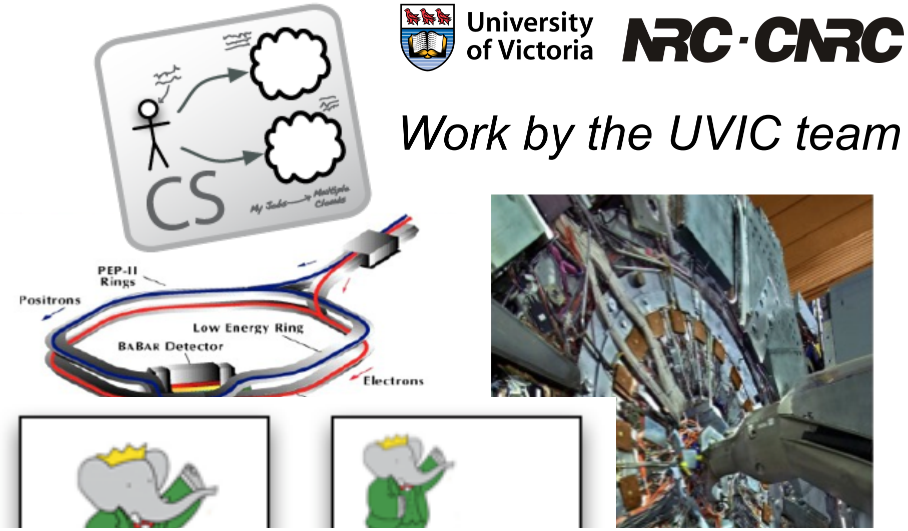
BABAR

™ and © Nelvana, All Rights Reserved

Canadian Efforts

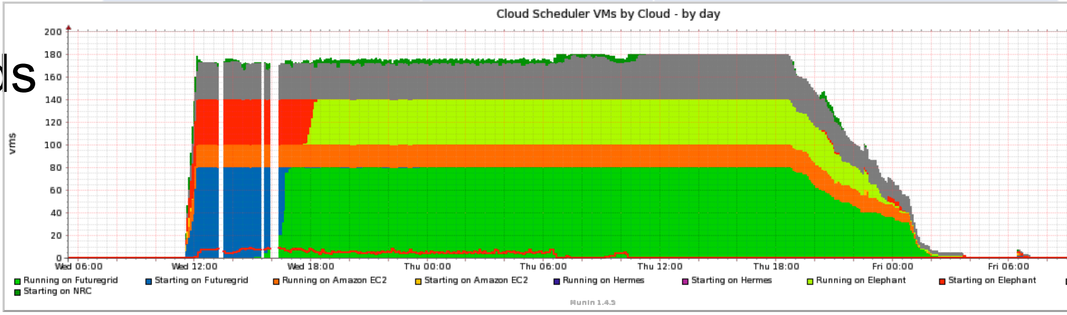
University of Victoria NRC·CNRC

Work by the UVIC team



- BarBar Experiment at SLAC in Stanford, CA
- Using clouds to simulate electron-positron collisions in their detector
- Exploring virtualization as a vehicle for data preservation
- Approach:
  - Appliance preparation and management
  - Distributed Nimbus clouds
  - Cloud Scheduler
- Running production BaBar workloads

Resource	Cores	Notes
FutureGrid @Argonne Lab	100 Cores Allocated	Resources allocation to support BaBar
Elephant Cluster @UVic	88 Cores	Experimental cloud cluster hosts (xrootd for cloud)
NRC Cloud in Ottawa	68 Cores	Hosts VM image repository (repoman)
Amazon EC2	Proportional to $	Grant funding from Amazon
Hermes Cluster @Uvic	Variable (280 max)	Occasional Backfill access



Science Program
25 to 30 Year
Construction Program
5 Year
Science Operations
Multi-scale Sustained Ocean Presence
Integrated Observatory Network
Marine Operations
Cyber Operations
Real-time Bidirectional Instrument Network
Coastal/Global
Regional
Implementing Organizations
Cyber Infrastructure
Implements

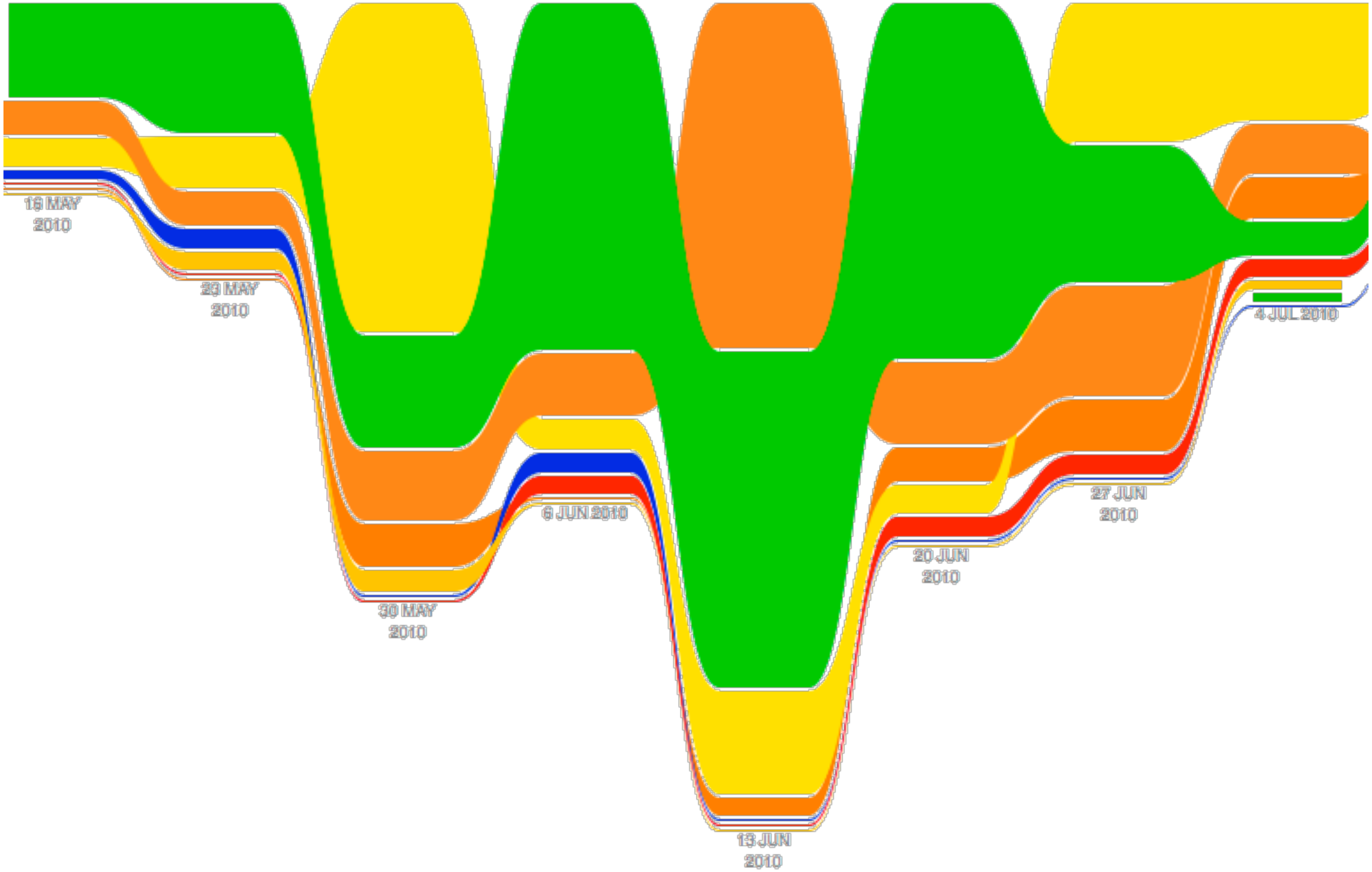
OOI
OCEAN OBSERVATORIES INITIATIVE

- Large NSF-funded observatory with requirements for tive, reliable, elastic c
- Approach:
  - Private I cloud clouds
  - Hig A) services th sources on m ased on need
  - S t OOI CI infrastructure in data and sensor management based on this model
- Status:
  - Scalability and reliability tests on 100s of EC2, FutureGrid and Magellan resources
  - HA elastic services release in Spring 2011

Trail-blazing project

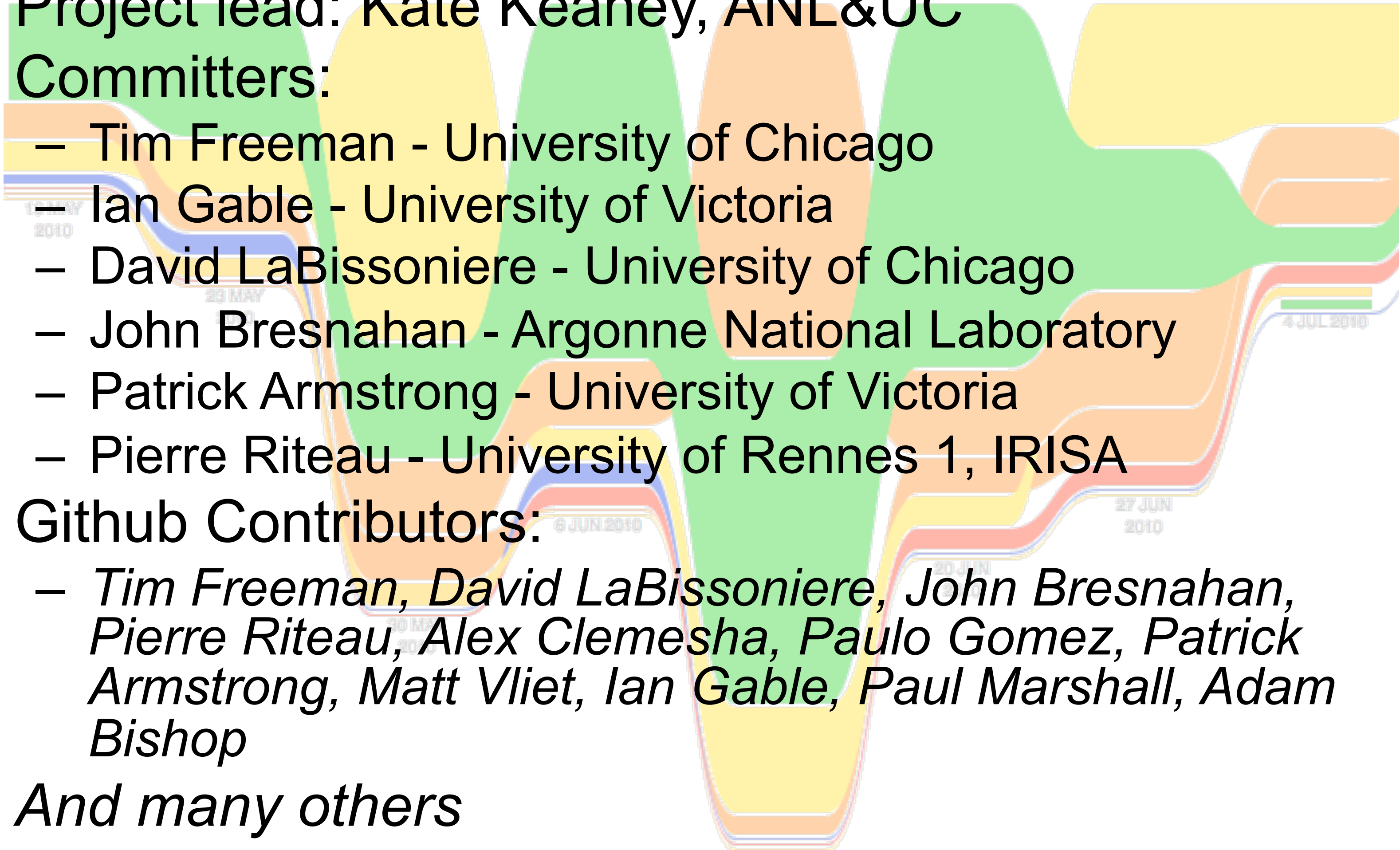
Nimbus Team

The Nimbus Team



The Nimbus Team

- Project lead: Kate Keahey, ANL&UC
- Committers:
  - Tim Freeman - University of Chicago
  - Ian Gable - University of Victoria
  - David LaBissoniere - University of Chicago
  - John Bresnahan - Argonne National Laboratory
  - Patrick Armstrong - University of Victoria
  - Pierre Riteau - University of Rennes 1, IRISA
- Github Contributors:
  - *Tim Freeman, David LaBissoniere, John Bresnahan, Pierre Riteau, Alex Clemesha, Paulo Gomez, Patrick Armstrong, Matt Vliet, Ian Gable, Paul Marshall, Adam Bishop*
- *And many others*
  - See *http://www.nimbusproject.org/about/people/*



Parting Thoughts

- Cloud Computing Challenge: Outsourcing
  - – Benefits
    - Economy of scale, access to different resources, no operation overhead, more flexible use
  - – Criteria
    - Does it provide the right offering? Is it scalable? Easy to use? Easy to outsource? Cost-effective?
  - – Not all or nothing – but close

www.nimbusproject.org

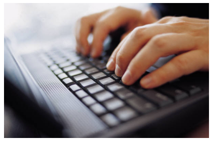
www.scienceclouds.org/blog

NIMBUS

www.nimbusproject.com

Let’s make cloud computing for science happen.

Hands-on: Get on the Cloud



Tutorial Exercises

- Download Nimbus cloud client
- Connect to *hotel* on FutureGrid
- Download your credentials
- Launch VMs!

https://portal.futuregrid.org/tutorials/nimbus

http://www.nimbusproject.org/docs/2.7/clouds/cloudquickstart.html

CCGrid
2011
May 23th -26th, 2011.
The 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, Newport Beach, CA, USA.

FutureGrid Nimbus Case Study: Extending Nimbus to Support Backfill VMs

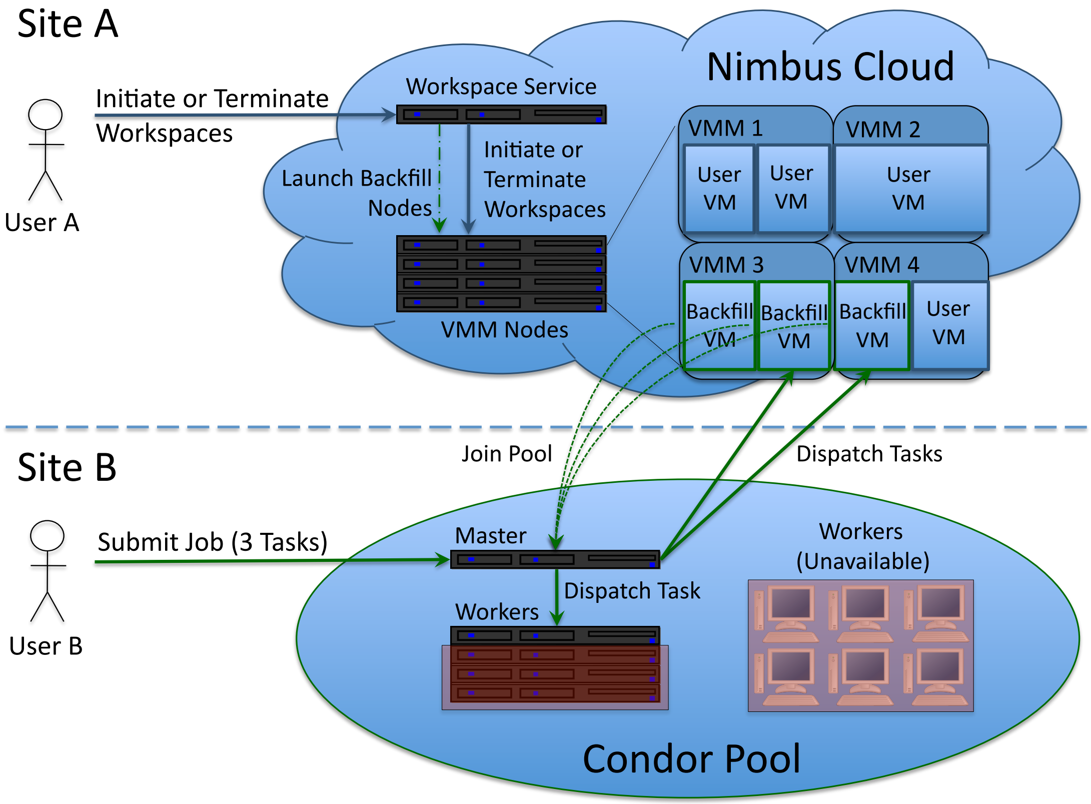
Paul Marshall

University of Colorado at Boulder

Addressing Cloud Utilization

- **Challenge:** utilization, catch-22 of on-demand computing

- Solutions:
  - Backfill
  - Spot pricing



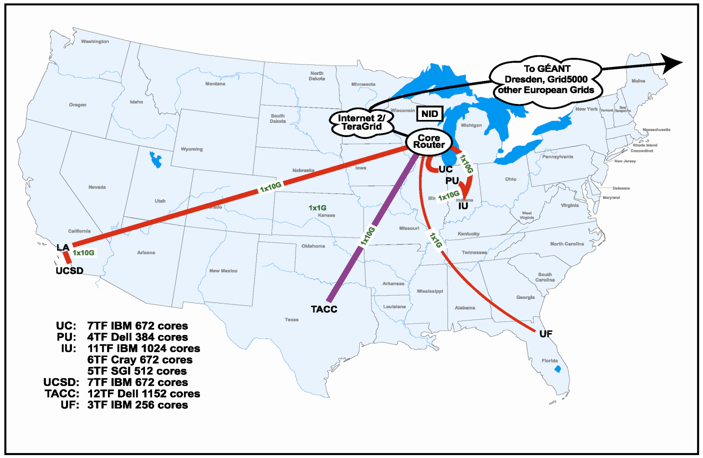
Extending Nimbus for Backfill

- Modify the Nimbus workspace service
  - Deploy backfill VMs on idle VMM nodes

- Requirements
  - Deploy and test a custom Nimbus service on a cloud frontend node
  - Integrate our custom Nimbus service with dedicated backend Nimbus VMM nodes
  - Evaluate our modified version of Nimbus in a real cloud environment

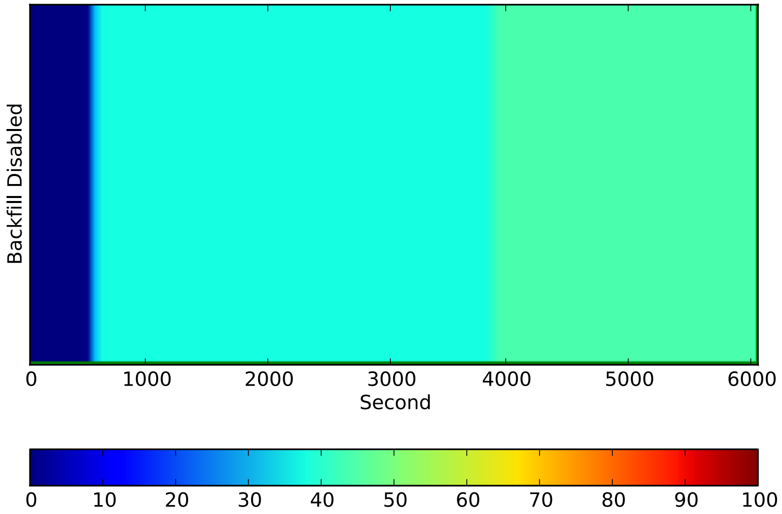
FutureGrid

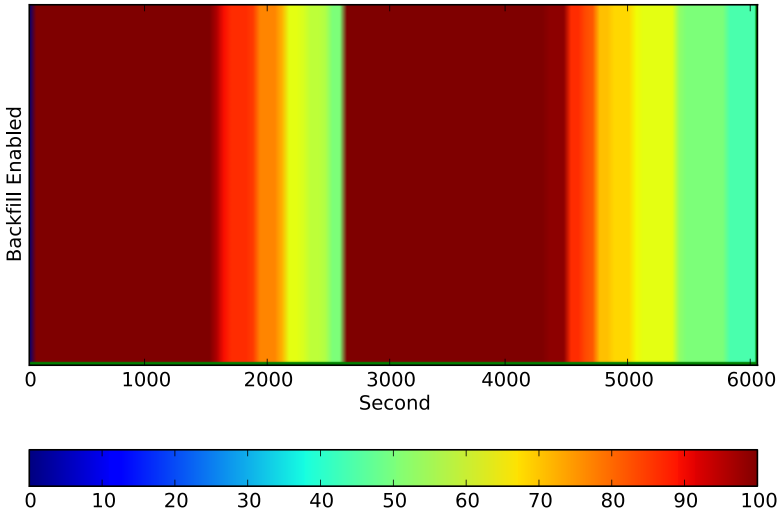
- Used the *hotel* resource on FutureGrid to deploy a custom version of the Nimbus service
- Obtained a dedicated set of Nimbus VMM nodes (16 8-core nodes) for a limited amount of time to integrate with our modified Nimbus service
- Evaluated our modified version of Nimbus in a real cloud environment



100% Utilization

- Overlaid an on-demand Nimbus workload with Condor jobs running in backfill VMs
  - Demonstrated an increase in utilization from 37.5% to 100%





For more of the details…

Paper:

Improving Utilization of Infrastructure Clouds

Authors: Paul Marshall, Kate Keahey, Tim Freeman

Presentation:

Wednesday, May 25th
Track 1: 11:00am – 12:30pm

What is next?

Try out other things

- Unicore
- Genesis

- Contribute

Feedback

- For suggestions on how to improve the tutorial, please send mail to
  - laszewski@gmail.com
- For technical questions, please send e-mail to
  - help@futuregrid.org

Virtual Appliances

What is an appliance?

- Hardware/software appliances
  - TV receiver + computer + hard disk + Linux + user interface

  - Computer + network interfaces + FreeBSD + user interface

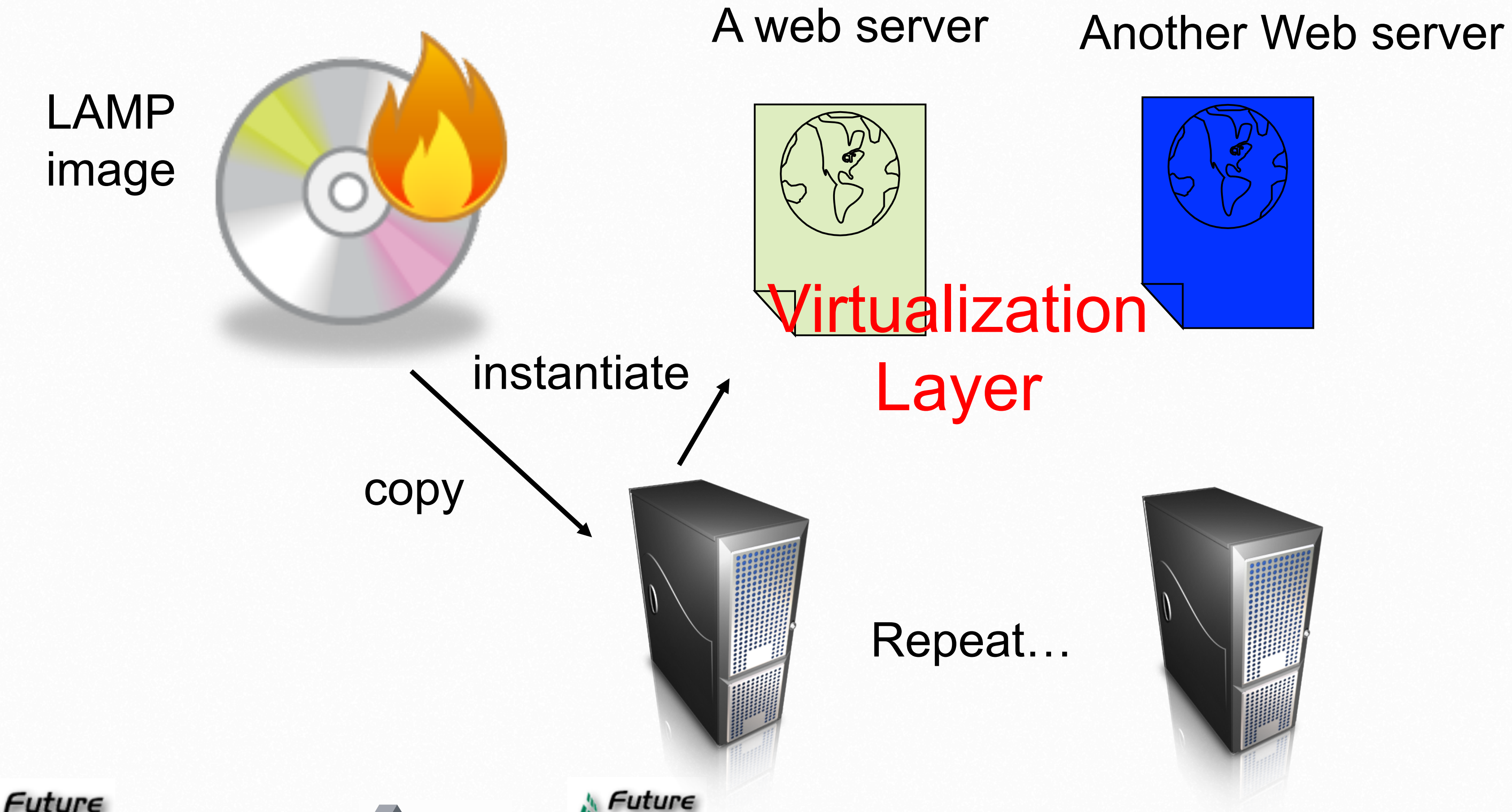
Future Grid ACIS Future Grid UF UNIVERSITY of FLORIDA

What is a virtual appliance?

- An appliance that packages software and configuration needed for a particular purpose into a virtual machine "image"
- The virtual appliance has no hardware – just software and configuration
- The image is a (big) file
- It can be *instantiated* on hardware

Virtual appliance example

- Linux + Apache + MySQL + PHP



A web server

Another Web server

LAMP
image

Virtualization
Layer

instantiate

copy

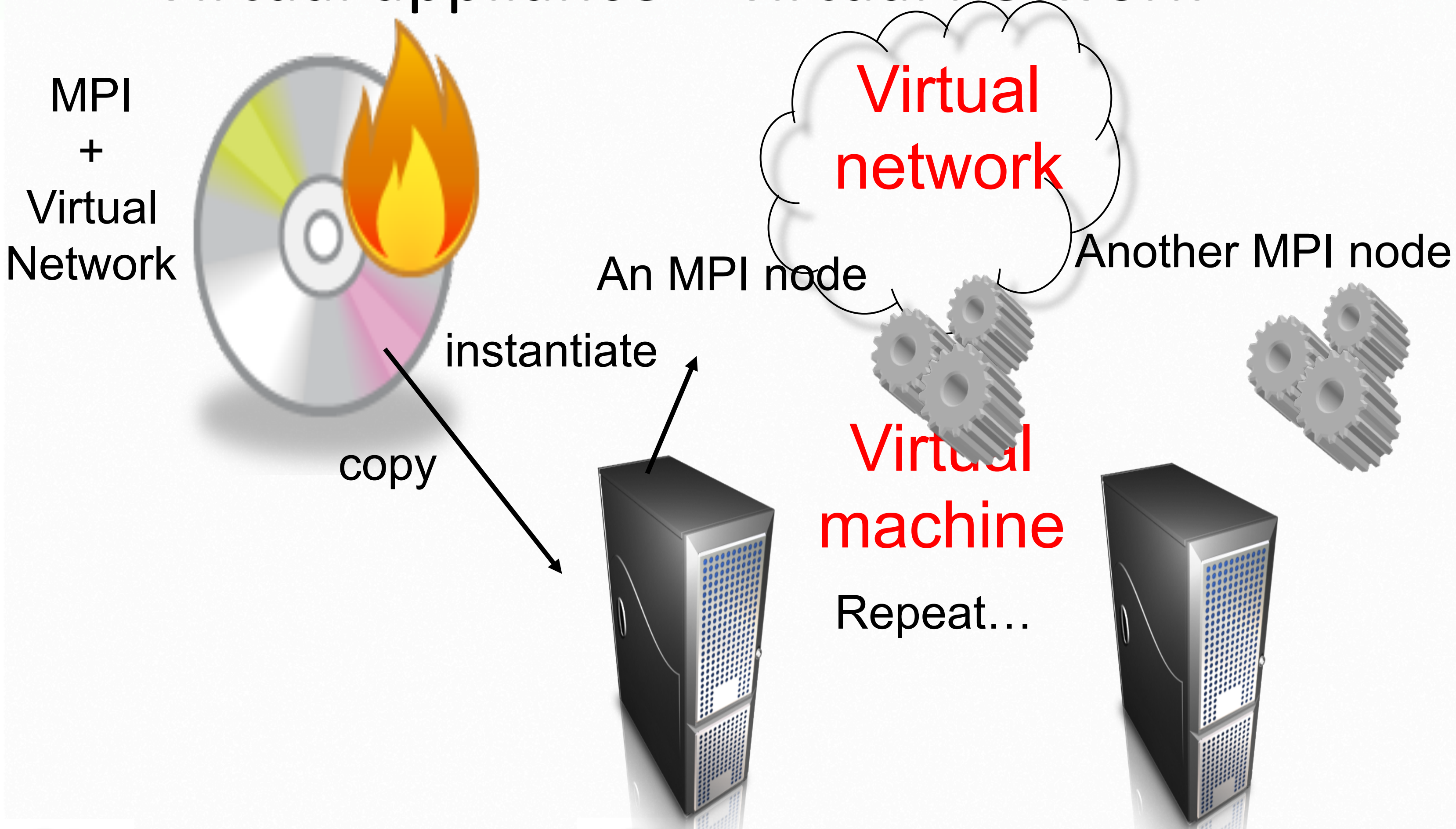
Repeat…

What about the network?

- Multiple Web servers might be completely independent from each other
- Parallel processing: workers are not
  - Need to communicate and coordinate with each other
  - Each worker needs an IP address, uses TCP/IP sockets
- Cluster middleware stacks assume a collection of machines, typically on a LAN (Local Area Network)

Virtual cluster appliances

- Virtual appliance + virtual network



MPI
+
Virtual
Network

copy

instantiate

An MPI node

Virtual network

Another MPI node

Virtual machine

Repeat…

Background

- Virtual appliances
  - Encapsulate software environment in image
    - Virtual disk file(s) and virtual hardware configuration
- The Grid appliance
  - Encapsulates *cluster* software environments
    - Current examples: Condor, MPI, Hadoop
  - Homogeneous images at each node
  - *Virtual LAN* connecting nodes to form a cluster
  - Deploy within or across domains

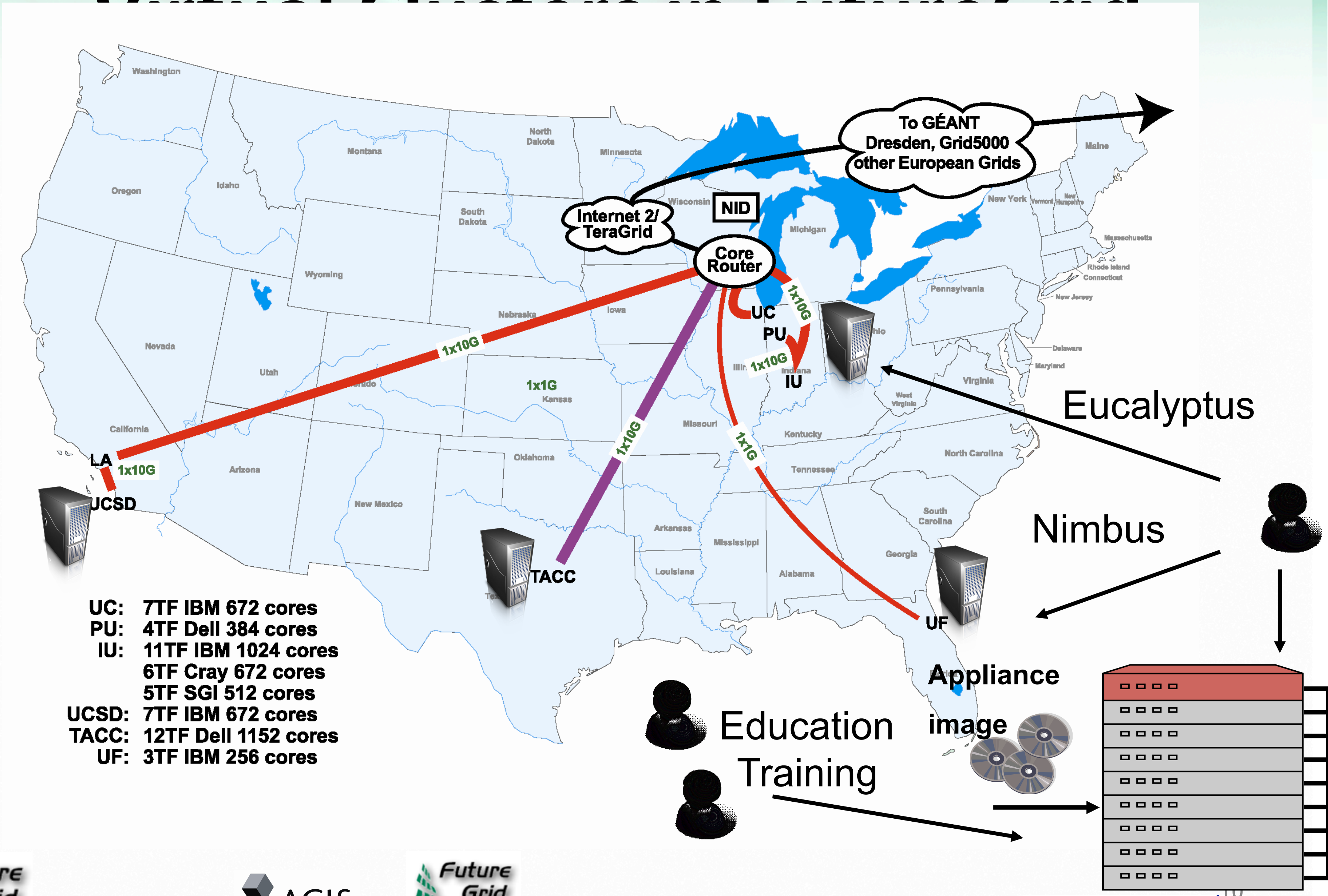
Grid appliance in a nutshell

- Plug-and-play clusters with a pre-configured software environment
  - Linux + (Hadoop, Condor, MPI, …)
  - Scripts for zero-configuration
  - "Virtual machine" appliance; open-source software runs on Linux, Windows, Mac
- Hands-on examples, bootstrap infrastructure, and zero-configuration software – *you're off to a quick start*

Grid appliance in a nutshell

- Creating an equivalent Grid on your own resources, or on cloud providers, is also easy
- Deploy image on FutureGrid, Amazon EC2
- Copy the same appliance to clusters, PC labs
- Simple deployment and management of ad-hoc clusters
  - Opportunistic computing
  - Testing, evaluation
  - Education, training

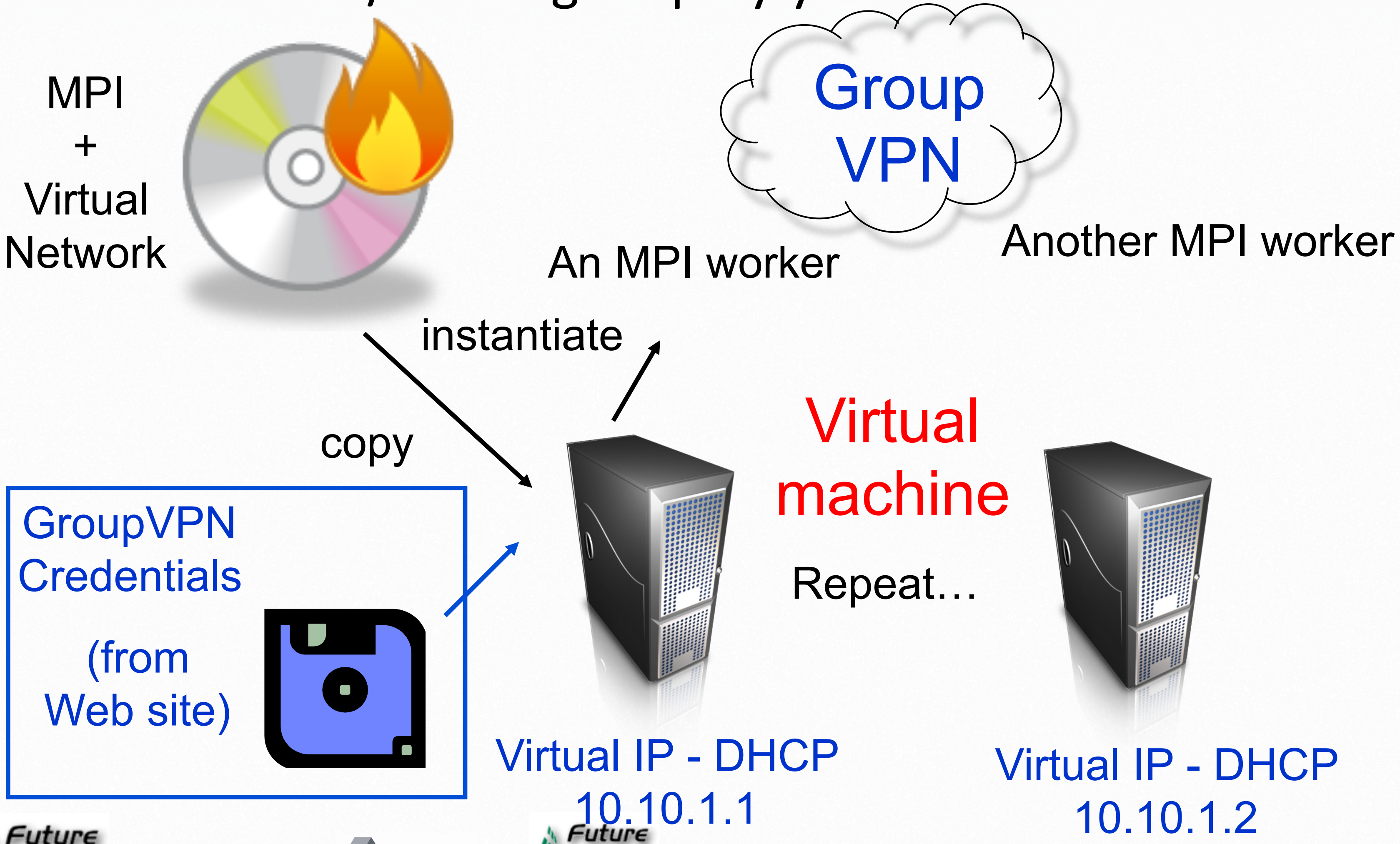
Virtual Clusters in FutureGrid



UC:	7TF IBM 672 cores
PU:	4TF Dell 384 cores
IU:	11TF IBM 1024 cores
	6TF Cray 672 cores
	5TF SGI 512 cores
UCSD:	7TF IBM 672 cores
TACC:	12TF Dell 1152 cores
UF:	3TF IBM 256 cores

Social virtual private networks

- Education/training: deploy your own cluster!



Demonstration

- Based on tutorial MP1
  - https://portal.futuregrid.org/tutorials/mp1
- Deploying a virtual appliance on FutureGrid through Nimbus
- Getting a virtual IP address and connecting to a small ‘playground’ pool of Condor nodes
- Installing MPI middleware
- Deploying MPI nodes dynamically through Condor
- Running a simple MPI task

Demonstration

- Deploying a virtual appliance on FutureGrid through Nimbus
  - Use Nimbus cloud client and baseline Grid appliance image available on alamo (TACC)

```
cloud-client.sh --conf alamo.conf --run --name grid-appliance-mpi-2.04.28.gz --hours 24
```

Demonstration

- Getting a virtual IP address and connecting to a small ‘playground’ pool of Condor nodes
  - Once instance is running:

```
ssh root@(IP address of instance)
/sbin/ifconfig tapipop
```

  - Virtual cluster’ s IP address – GroupVPN

```
condor_status
```

  - List of other nodes connected to public pool
  - You can create your own private VPN as well

Demonstration

- Installing MPI middleware

```
su griduser
cd ~/examples/mpi
./setup.sh –m32
```

  - In this example, we' re building MPI from scratch
  - If you customize an appliance with software/ middleware, you can also generate your own custom image, and deploy multiple instances from there

Demonstration

- Deploying MPI nodes dynamically through Condor and running a simple MPI task

```
/mnt/local/mpich2/bin/mpicc -m32 -o HelloWorld HelloWorld.c
```

  - Compile MPI binary

```
./mpi_submit.py -n 2 HelloWorld
```

  - Submit a Condor job that creates a 2-node MPI pool and submits the HelloWorld library

Future Grid ACIS Future Grid UF UNIVERSITY of FLORIDA

Where to go from here?

- You can download Grid appliances and run on your own resources
- You can create private virtual clusters and manage groups of users
- You can customize appliances with other middleware, create images, and share with other users
- More tutorials available at FutureGrid.org
- More information on Grid appliances also available at Grid-appliance.org
- Contact Renato Figueiredo renato@acis.ufl.edu for more information about appliances