Science Clouds and CFD

NIA CFD Conference: *Future Directions in CFD Research, A Modeling and Simulation Conference* August 6-8, 2012

Embassy Suites Hampton Roads -

August 6 2012

Geoffrey Fox

gcf@indiana.edu

Informatics, Computing and Physics

Pervasive Technology Institute

Indiana University Bloomington

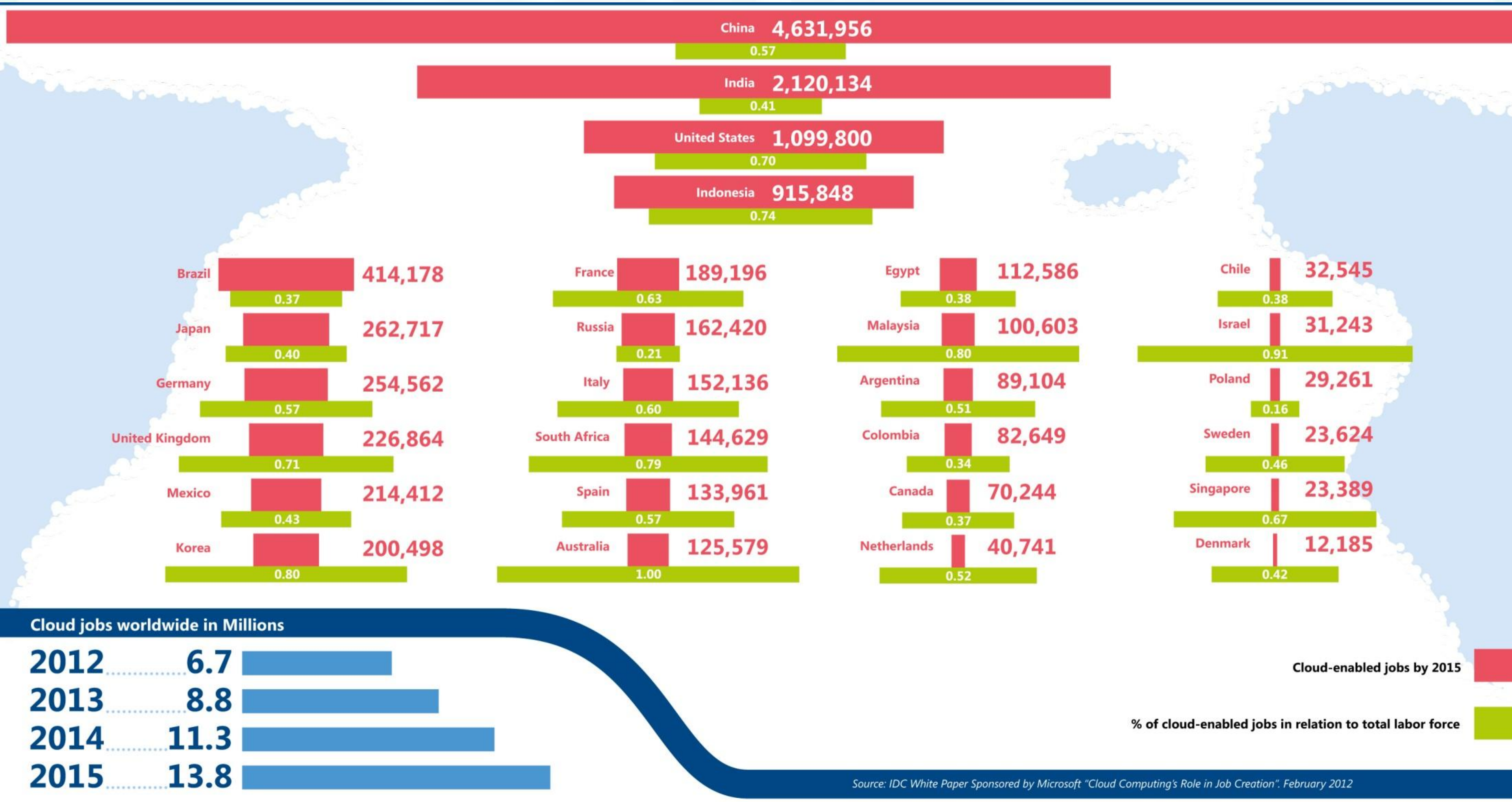
https://portal.futuregrid.org

Broad Overview: Clouds

Clouds Offer From different points of view

- **Features from NIST:**
  - On-demand service (elastic);
  - Broad network access;
  - Resource pooling (sharing);
  - Flexible resource allocation;
  - Measured service
- **Economies of scale** in performance and electrical power **(Green IT)**
- **Ease of Use** can be better for clouds
- **Clouds** have lots of **Jobs** and capture attention of **students**
- Powerful new **software models**
  - **Platform as a Service** is not an alternative to **Infrastructure as a Service –** it is instead an incredible valued added
- **Clouds** are likely to **drive commercial node** architecture, power, storage, programming **technologies** and so be **enabler of Exascale**

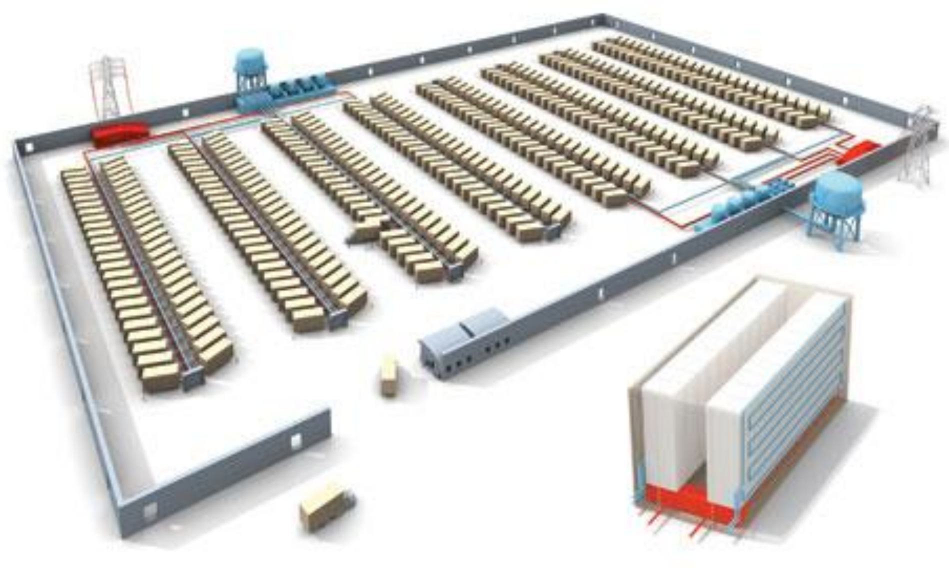
Cloud Jobs v. Countries



Clouds as Cost Effective Data Centers

- Clouds can be considered as just the best biggest data centers
- **Right** is 2 Google warehouses of computers on the banks of the Columbia River, in The Dalles, Oregon
- **Left** is shipping container (each with 200-1000 servers) model used in Microsoft Chicago data center holding 150-220

Data Center Part	Cost in small-sized Data Center	Cost in Large Data Center	Ratio
Network	$95 per Mbps/ month	$13 per Mbps/ month	7.1
Storage	$2.20 per GB/ month	$0.40 per GB/ month	5.7
Administ ration	~140 servers/ Administ rator	>1000 Servers/ Administr ator	7.1



Some Sizes in 2010

- http://www.mediafire.com/file/zzqna34282frr2f/koomeydatacenterelectuse2011finalversion.pdf
- 30 million servers worldwide
- Google had 900,000 servers (3% total world wide)
- Google total power ~200 Megawatts
  - < 1% of total power used in data centers (Google more efficient than average – **Clouds are Green**!)
  - ~ 0.01% of total power used on anything world wide
- Maybe total clouds are 20% total world server count (a growing fraction)

Some Sizes Cloud v HPC

- **Top Supercomputer** Sequoia Blue Gene Q at LLNL
  - 16.32 Petaflop/s on the Linpack benchmark using 98,304 CPU compute chips with 1.6 million processor cores and 1.6 Petabyte of memory in 96 racks covering an area of about 3,000 square feet
  - 7.9 Megawatts power
- **Largest (cloud)** computing data centers
  - 100,000 servers at ~200 watts per chip (two chips per server)
  - Up to 30 Megawatts power
- So **largest supercomputer** is a bit smaller than **largest major cloud computing centers;** it is **~ 1%** of **total** major cloud systems
  - Sum of all machines in Top500 ~ 10x top machine
  - Total “supercomputers” ~20x top machine

Clouds Grids and HPC

2 Aspects of Cloud Computing: Infrastructure and Runtimes

- Cloud infrastructure: outsourcing of servers, computing, data, file space, utility computing, etc..
- Cloud runtimes or Platform: tools to do data-parallel (and other) computations. Valid on Clouds and traditional clusters
  - Apache Hadoop, Google **MapReduce**, Microsoft Dryad, Bigtable, Chubby and others
  - MapReduce designed for information retrieval but is excellent for a wide range of science data analysis applications
  - Can also do much traditional parallel computing for data-mining if extended to support iterative operations
  - **Data Parallel File system** as in HDFS and Bigtable
- **Service Oriented Architectures portals** and **workflow** appear to work similarly in both grids and clouds

Science Computing Environments

- **Large Scale Supercomputers** – Multicore nodes linked by high performance low latency network
  - Increasingly with GPU enhancement
  - Suitable for highly parallel simulations
- **High Throughput Systems** such as European Grid Initiative EGI or Open Science Grid OSG typically aimed at pleasingly parallel jobs
  - Can use “cycle stealing”
  - Classic example is **LHC data analysis**
- **Grids** federate resources as in EGI/OSG or enable convenient access to multiple backend systems including supercomputers
  - **Portals** make access convenient and
  - **Workflow** integrates multiple processes into a single job
- Specialized **visualization**, **shared memory parallelization** etc. machines

Clouds HPC and Grids

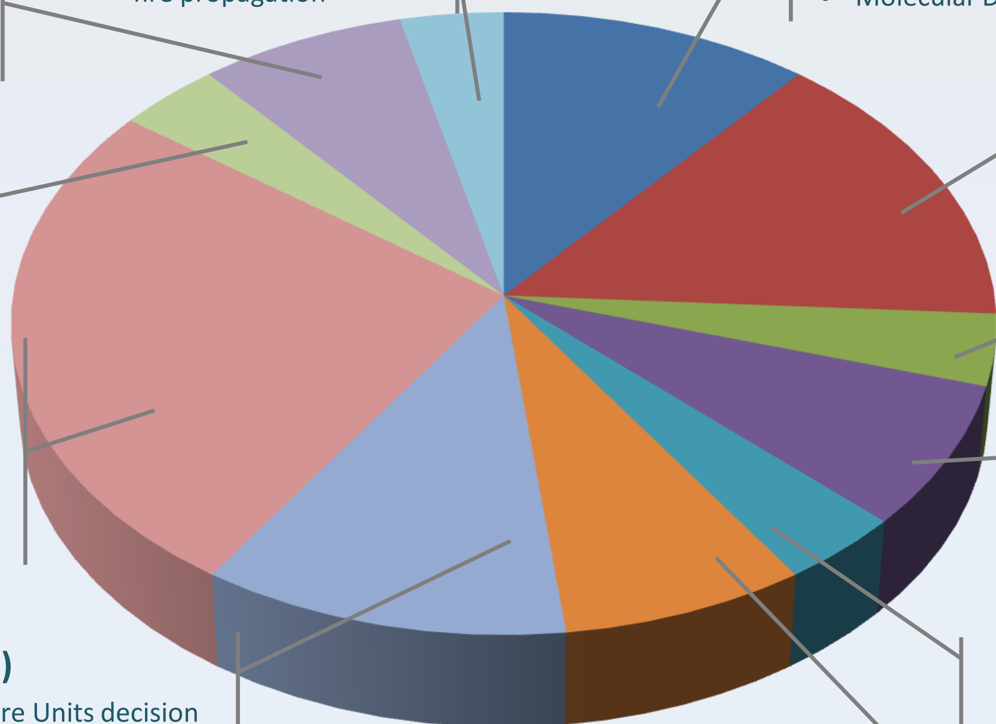
- Synchronization/communication Performance
  Grids > Clouds > Classic HPC Systems
- **Clouds** naturally execute effectively **Grid** workloads but are not good for closely coupled HPC applications on large clusters
  - **GPU's** being added efficiently to **Cloud** Infrastructure (OpenStack, Amazon)
- Note nodes are easy virtualization unit and so **node sized** (moving to modest # nodes) **problems natural for clouds**
- **Classic HPC machines** as MPI engines offer highest possible performance on closely coupled problems
- May be for immediate future, science supported by a mixture of
  - **Clouds** – some practical differences between private and public clouds – size and software
  - **High Throughput Systems** (moving to clouds as convenient)
  - **Grids** for distributed data and access
  - **Supercomputers** ("MPI Engines") going to exascale

What Applications work in Clouds

- **Pleasingly (moving to modestly) parallel** applications of all sorts with roughly independent data or spawning independent simulations
  - **Long tail** of science and integration of distributed sensors
- **Commercial and Science Data analytics** that can use MapReduce **(**some of such apps) or its **iterative** variants (most other data analytics apps)
- **Which science applications are using clouds**?
  - **Venus-C** (Azure in Europe): 27 applications **not using** Scheduler, Workflow or MapReduce (except roll your own)
  - 50% of applications on **FutureGrid** are from Life Science
  - Locally **Lilly** corporation is commercial cloud user (for drug discovery)
  - Nimbus applications in bioinformatics, high energy physics, nuclear physics, astronomy and ocean sciences

27 Venus-C Azure Applications

Red related to CFD



Chemistry (3)
- Lead Optimization in Drug Discovery
- Molecular Docking

Civil Protection (1)
- Fire Risk estimation and fire propagation

Biodiversity & Biology (2)
- Biodiversity maps in marine species
- Gait simulation

Civil Eng. and Arch. (4)
- Structural Analysis
- Building information Management
- Energy Efficiency in Buildings
- Soil structure simulation

Physics (1)
- Simulation of Galaxies configuration

Earth Sciences (1)
- Seismic propagation

Mol, Cell. & Gen. Bio. (7)
- Genomic sequence analysis
- RNA prediction and analysis
- System Biology
- Loci Mapping
- Micro-arrays quality.

ICT (2)
- Logistics and vehicle routing
- Social networks analysis

Medicine (3)
- Intensive Care Units decision support.
- IM Radiotherapy planning.
- Brain Imaging

Mathematics (1)
- Computational Algebra

Mech, Naval & Aero. Eng. (2)
- Vessels monitoring
- Bevel gear manufacturing simulation

Parallelism over Users and Usages

- "**Long tail of science**" can be an important usage mode of clouds.
- In some areas like particle physics and astronomy, i.e. "**big science**", there are just a few major instruments generating now petascale data driving discovery in a coordinated fashion.
- In other areas such as genomics and environmental science, there are many **"individual" researchers** with distributed collection and analysis of data whose total data and processing needs can match the size of big science.
- Similarly **"parameter searches"** with myriad of jobs exploring parameter space
- Can be **map only** use of MapReduce if different usages naturally linked e.g. exploring docking of multiple chemicals or alignment of multiple DNA sequences
  - Collecting together or summarizing multiple "maps" is a **simple Reduction**

Internet of Things and the Cloud

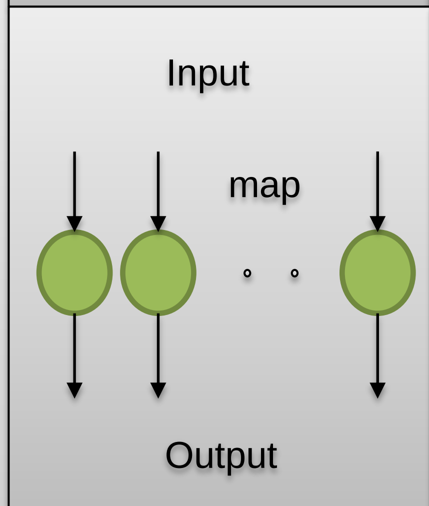
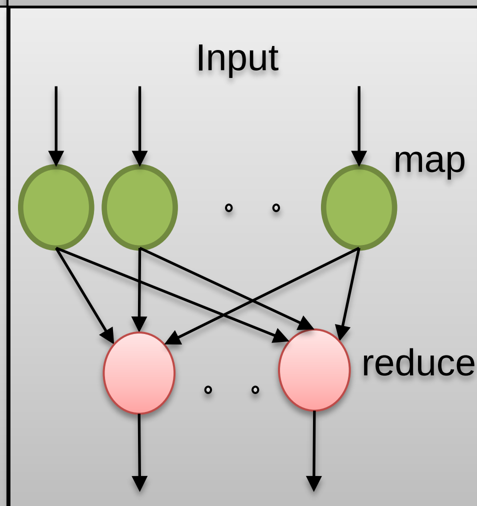
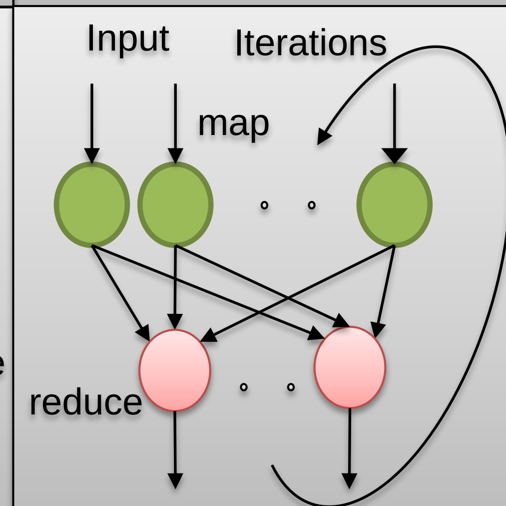
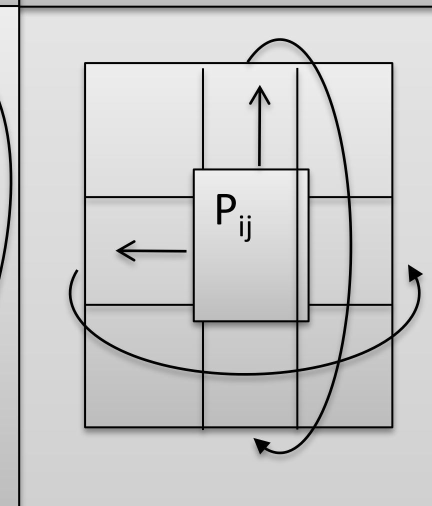
- It is projected that there will be **24 billion devices** on the Internet by 2020. Most will be small sensors that send streams of information into the cloud where it will be processed and integrated with other streams and turned into knowledge that will help our lives in a multitude of small and big ways.
- The **cloud** will become increasing important as a controller of and **resource provider for the Internet of Things.**
- As well as today's use for smart phone and gaming console support, "smart homes" and "ubiquitous cities" build on this vision and we could expect a growth in cloud supported/controlled **robotics**.
- Some of these "things" will be supporting science e.g. instruments monitoring and recording aircraft performance
- Natural parallelism over "things"
- "Things" are distributed and so form a Grid

Parallel Computing on Clouds and HPC

Classic Parallel Computing

- **HPC:** Typically SPMD (Single Program Multiple Data) "maps" typically processing particles or mesh points interspersed with multitude of low latency messages supported by specialized networks such as Infiniband and technologies like MPI
  - Often run large capability jobs with 100K (going to 1.5M) cores on same job
  - National DoE/NSF/NASA facilities run 100% utilization
  - Fault fragile and cannot tolerate "outlier maps" taking longer than others
- **Clouds:** MapReduce is dominant commercial messaging system with as dynamic asynchronous maps (computations). Final reduce phase integrates results from different maps
  - Fault tolerant and does not require map synchronization
  - **Map only** useful special case
- **HPC + Clouds:** Iterative MapReduce caches results between "MapReduce" steps and supports SPMD **parallel computing with large messages** as seen in parallel kernels (linear algebra) in clustering and other data mining

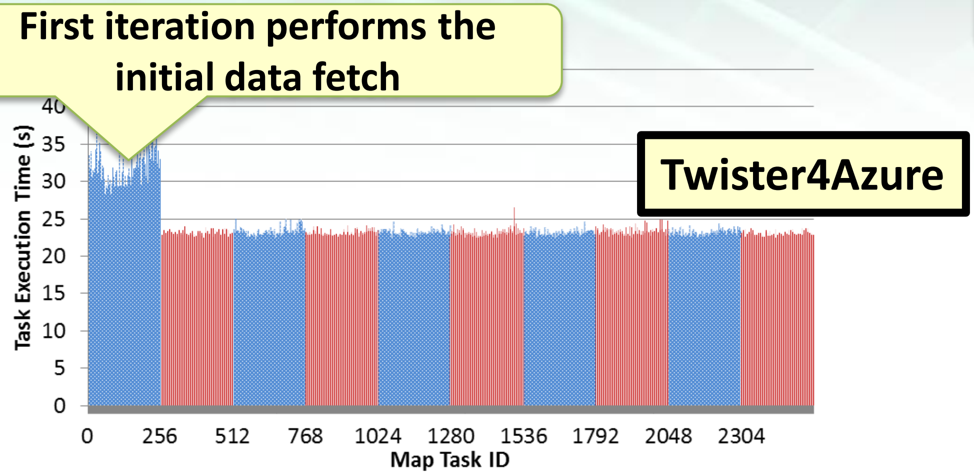
4 Forms of MapReduce

<table>
<tr><th>(a) Map Only</th><th>(b) Classic MapReduce</th><th>(c) Iterative MapReduce</th><th>(d) Loosely Synchronous</th></tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLAST Analysis
Parametric sweep
Pleasingly Parallel</td>
<td>High Energy Physics (HEP) Histograms
Distributed search</td>
<td>Expectation maximization
Clustering e.g. Kmeans
Linear Algebra, Page Rank</td>
<td>Classic MPI
PDE Solvers and particle dynamics</td>
</tr>
<tr>
<td colspan="3">Domain of MapReduce and Iterative Extensions
Science Clouds</td>
<td>MPI
HPC Clusters</td>
</tr>
</table>

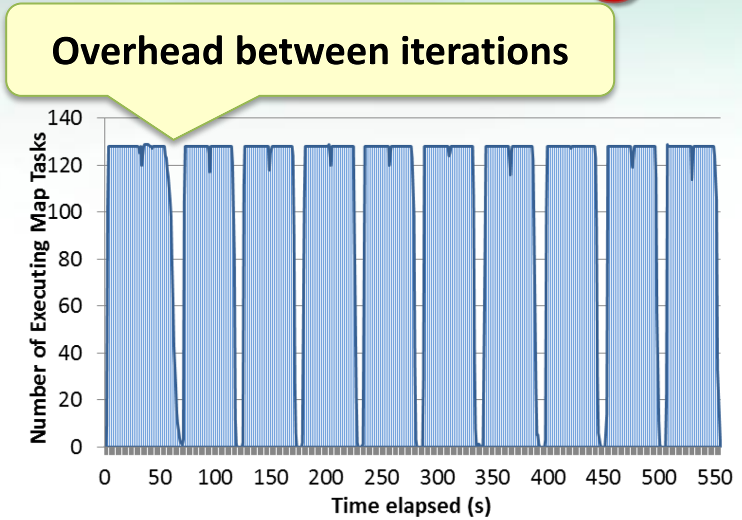
Clouds and Exascale

- **Clouds** are application **driving multicore** as natural parallelism exploiting cores
  - Clients becoming smaller; can't exploit cores
- **Commodity Server** node technology aimed at clouds
  - **Blue Gene** good example of HPC node but GPU + Commodity is common between Exascale and clouds
- **Clouds pioneering fault tolerance** in large scale systems
  - **Exascale** harder as applications are **more closely coupled**
  - **MapReduce** has fault tolerance and load balancing irregular loads
- Clouds point to **Green IT** and data center approaches
- Clouds have **more I/O** than traditional HPC systems
- **Node programming model** comes from commodity applications
  - note data parallel "analytics" (Pig) more successful than "simulations" (HPF)
- **Commercial Exascale** will **build** on **cloud technology**
- Exascale **Network technology** likely to be special

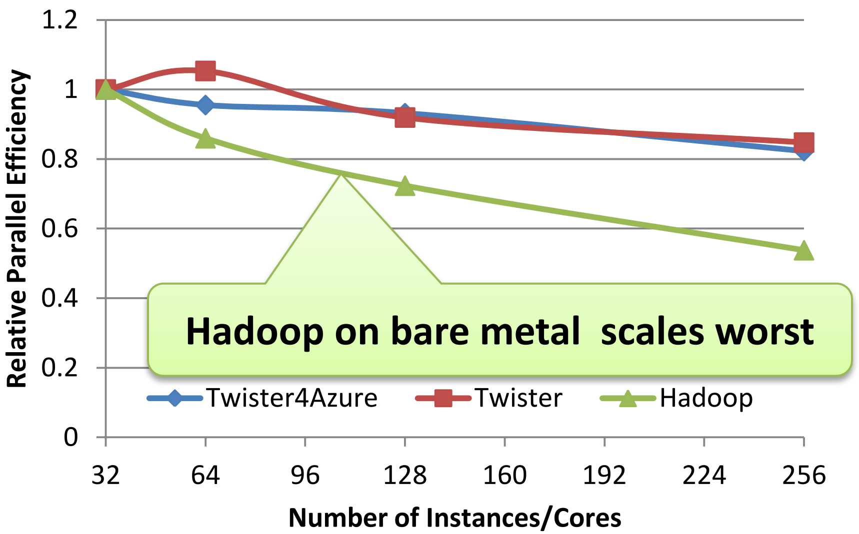
Performance – Kmeans Clustering



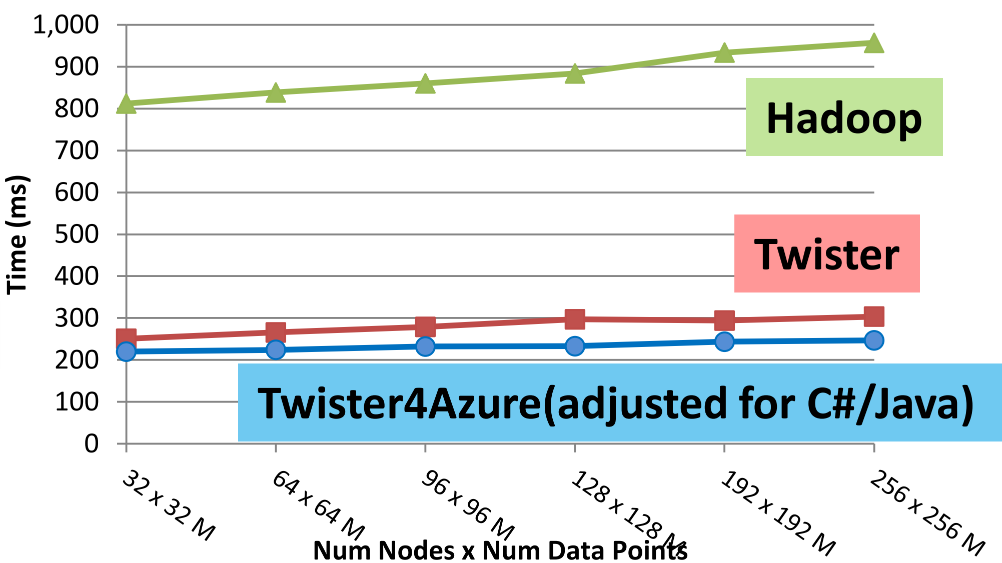
Task Execution Time Histogram



Number of Executing Map Task Histogram



Strong Scaling with 128M Data Points



Weak Scaling

Qiu, Gunarathne

Future Grid

https://portal.futuregrid.org

Infrastructure as a Service
Platforms as a Service
Software as a Service

Infrastructure, Platforms, Software as a Service

SaaS
- **System e.g. SQL, GlobusOnline**
- **Applications e.g. Nastran, Fluent**

PaaS
- **Cloud e.g. MapReduce**
- **HPC e.g. PETSc, SAGA**
- **Computer Science e.g. Languages, Sensor nets**

IaaS
- **Hypervisor**
- **Bare Metal**
- **Operating System**
- **Virtual Clusters, Networks**

- Software Services are building blocks of applications
- The middleware or computing environment

Includes virtual clusters, virtual networks, management systems Nimbus, Eucalyptus, OpenStack

aaS and Roles/Appliances I

- Putting capabilities into Images (software for capability plus O/S) is key idea in clouds
  - Can do in two different ways: **aaS** and **Appliances**
- If you package a capability X as a service **XaaS**, it runs on a separate VM and you interact with messages
  - **SQLaaS** offers databases via messages similar to old JDBC model
- If you build a **role** or **appliance** with X, then X built into VM and you just need to add your own code and run
  - **i.e. base images can be customized**
  - **Generic worker role in Venus-C (Azure)** builds in I/O and scheduling
  - What do we need for a CFD Appliance or a set of MDO appliances?

aaS and Roles/Appliances II

- I expect a growing number of **carefully designed images** and services
  - Supports **ease of use** of both existing code and developing new codes with appliances that have useful features loaded
  - Supports **reproducible** science&**engineering** as appliances + virtual clusters can be specified and rerun on demand
- **Multidisciplinary Optimization** well supported by SaaS and Appliances as needs several interacting services that we can ready to go on cloud
- Can specify **appliances abstractly** so can instantiate on **Amazon, Azure, Eucalyptus, Nimbus, OpenNebula, OpenStack** or directly on a bare metal **HPC** node

What to use in Clouds: Cloud PaaS

- Job Management
  - **Queues** to manage multiple tasks
  - **Tables** to track job information
  - **Workflow** to link multiple services (functions)
- Programming Model
  - **MapReduce** and **Iterative MapReduce** to support parallelism
- Data Management
  - HDFS style **file system** to collocate data and computing
  - **Data Parallel Languages** like Pig; more successful than HPF?
- Interaction Management
  - **Services** for everything
  - **Portals** as User Interface
  - **Scripting** for fast prototyping
  - **Appliances** and **Roles** as customized images
- New Generation Software tools
  - like **Google App Engine, memcached**

What to use in Grids and Supercomputers? HPC (including Grid) PaaS

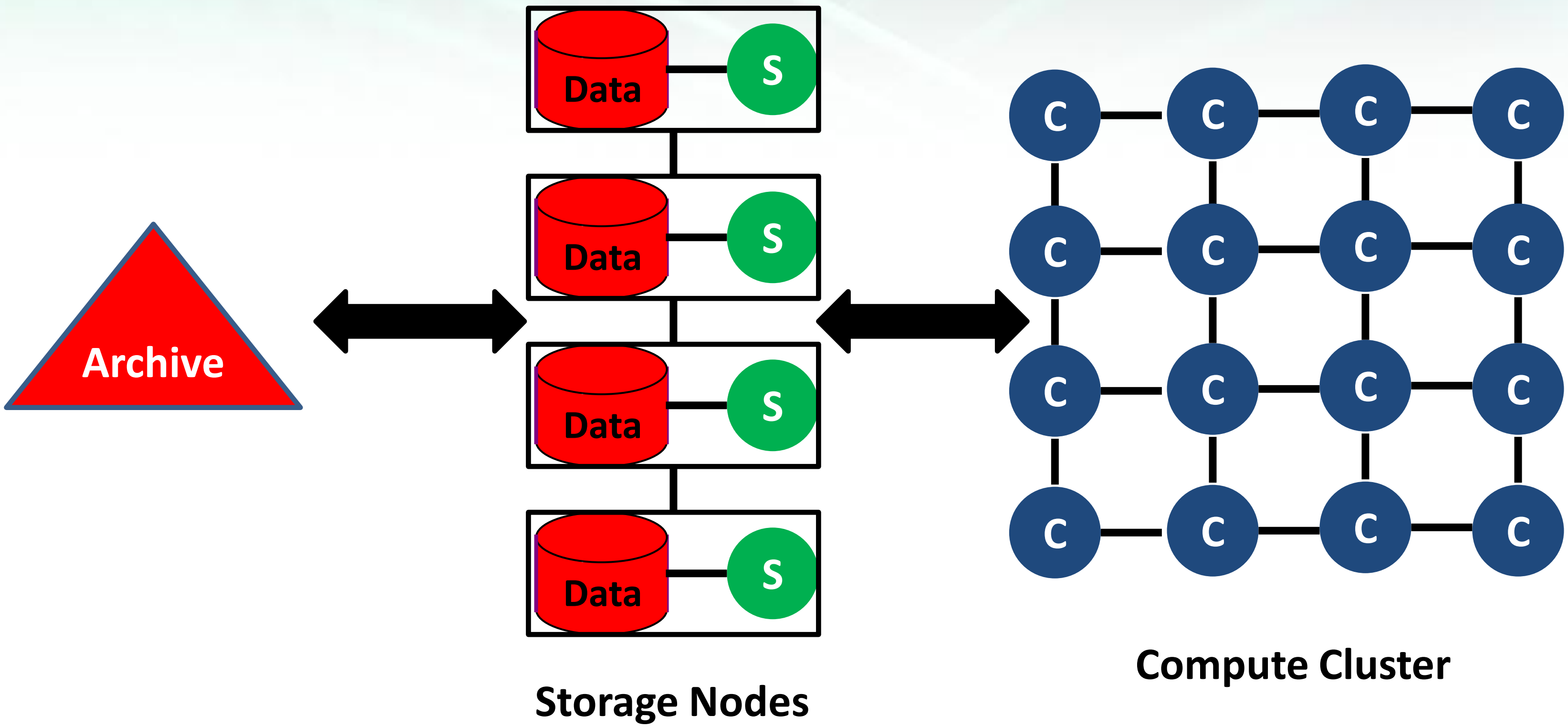
- **Job Management**
  - **Queues, Services Portals** and **Workflow** as in clouds
- **Programming Model**
  - **MPI** and **GPU/multicore threaded** parallelism
  - **Wonderful libraries** supporting parallel linear algebra, particle evolution, partial differential equation solution
- **Data Management**
  - **GridFTP** and high speed networking
  - **Parallel I/O** for high performance in an application
  - **Wide area File System** (e.g. Lustre) supporting file sharing
- **Interaction Management and Tools**
  - **Globus, Condor, SAGA, Unicore, Genesis** for Grids
  - Scientific **Visualization**
- **Let's unify Cloud and HPC PaaS and add Computer Science PaaS**?

Computer Science PaaS

- Tools to support Compiler Development
- Performance tools at several levels
- Components of Software Stacks
- Experimental language Support
- Messaging Middleware (Pub-Sub)
- Semantic Web and Database tools
- Simulators
- System Development Environments
- Open Source Software from Linux to Apache

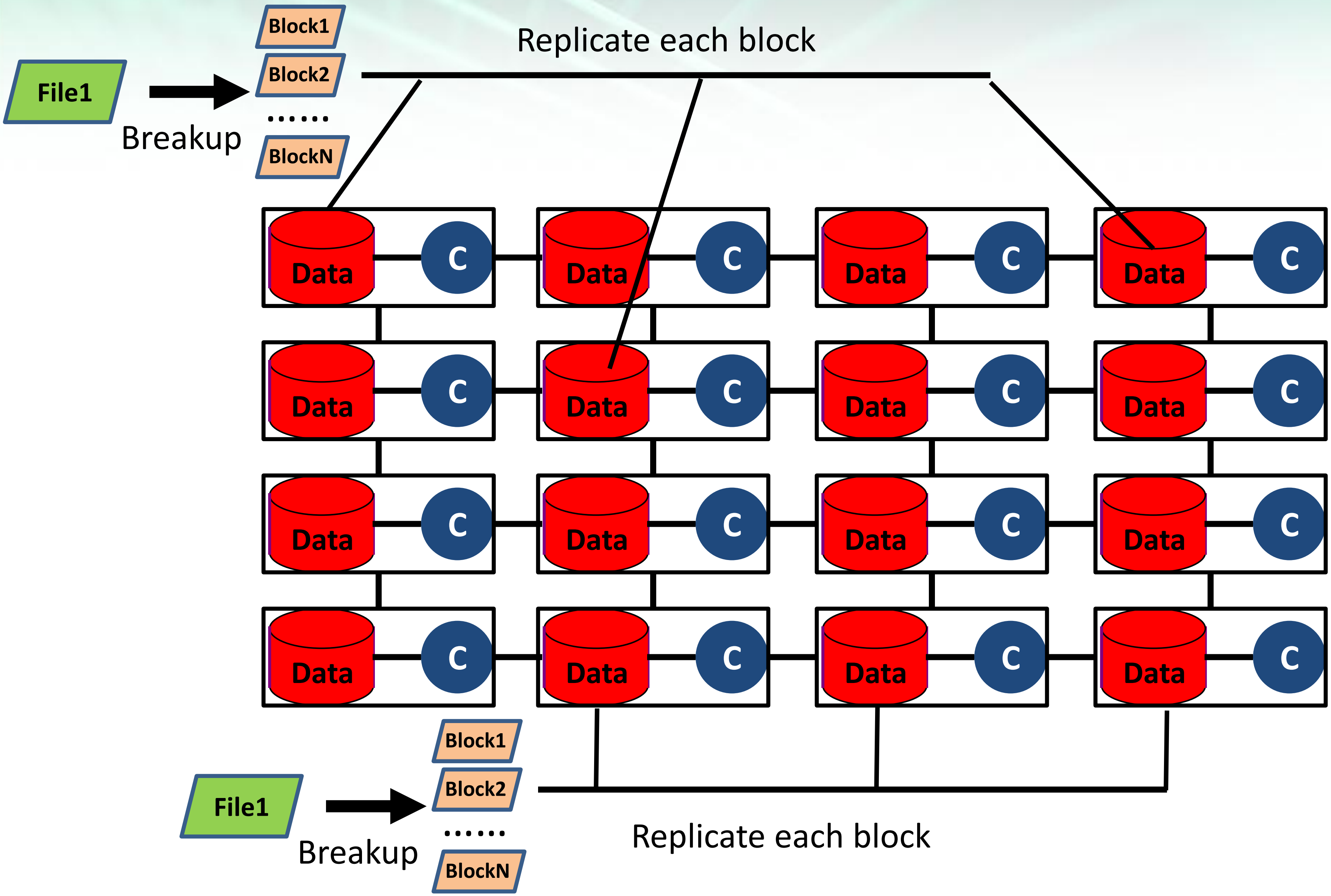
Authentication and Authorization: Provide single sign in to All system architectures
Workflow: Support workflows that link job components between Grids and Clouds.
Provenance: Continues to be critical to record all processing and data sources
Data Transport: Transport data between job components on Grids and Commercial Clouds respecting custom storage patterns like Lustre v HDFS
Program Library: Store Images and other Program material
Blob: Basic storage concept similar to Azure Blob or Amazon S3
DPFS Data Parallel File System: Support of file systems like Google (MapReduce), HDFS (Hadoop) or Cosmos (dryad) with compute-data affinity optimized for data processing
Table: Support of Table Data structures modeled on Apache Hbase/CouchDB or Amazon SimpleDB/Azure Table. There is “Big” and “Little” tables – generally NOSQL
SQL: Relational Database
Queues: Publish Subscribe based queuing system
Worker Role: This concept is implicitly used in both Amazon and TeraGrid but was (first) introduced as a high level construct by Azure. Naturally support **Elastic Utility Computing**
MapReduce: Support MapReduce Programming model including Hadoop on Linux, Dryad on Windows HPCS and Twister on Windows and Linux. Need Iteration for Datamining
Software as a Service: This concept is shared between Clouds and Grids
Web Role: This is used in Azure to describe user interface and can be supported by portals in Grid or HPC systems

Traditional File System?



- Typically a shared file system (Lustre, NFS …) used to support high performance computing
- Big advantages in flexible computing on shared data but doesn’t “**bring computing to data**”
- Object stores similar structure (separate data and compute) to this

Data Parallel File System?



- No archival storage and computing brought to data

FutureGrid

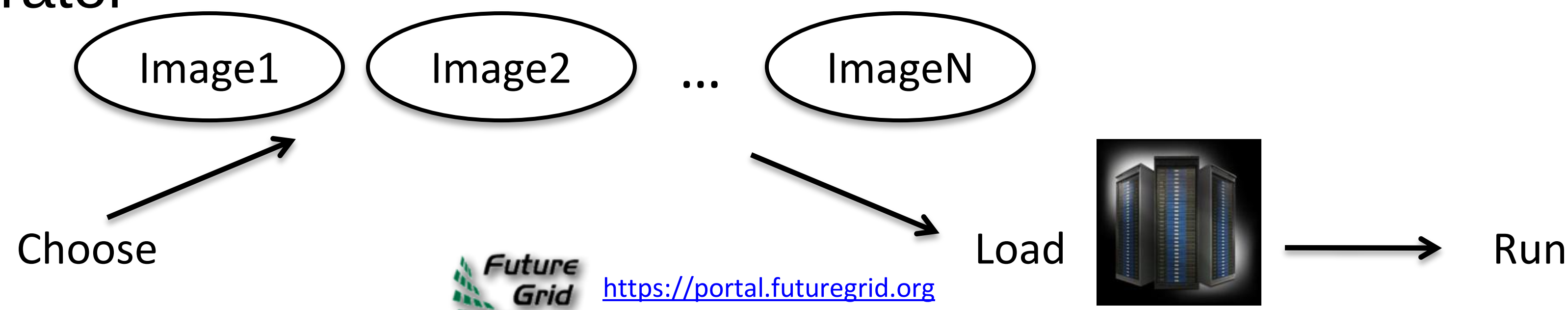
FutureGrid key Concepts I

- FutureGrid is an international testbed modeled on Grid5000
  - July 15 2012: **223 Projects, ~968 users**
- Supporting international Computer Science and Computational Science research in cloud, grid and parallel computing (HPC)
- The FutureGrid testbed provides to its users:
  - A flexible development and testing platform for middleware and application users looking at interoperability, functionality, performance or evaluation
  - FutureGrid is user-customizable, accessed interactively and supports Grid, Cloud and HPC software with and without VM's
  - A rich education and teaching platform for classes
- See G. Fox, G. von Laszewski, J. Diaz, K. Keahey, J. Fortes, R. Figueiredo, S. Smallen, W. Smith, A. Grimshaw, **FutureGrid - a reconfigurable testbed for Cloud, HPC and Grid Computing**, Bookchapter – draft

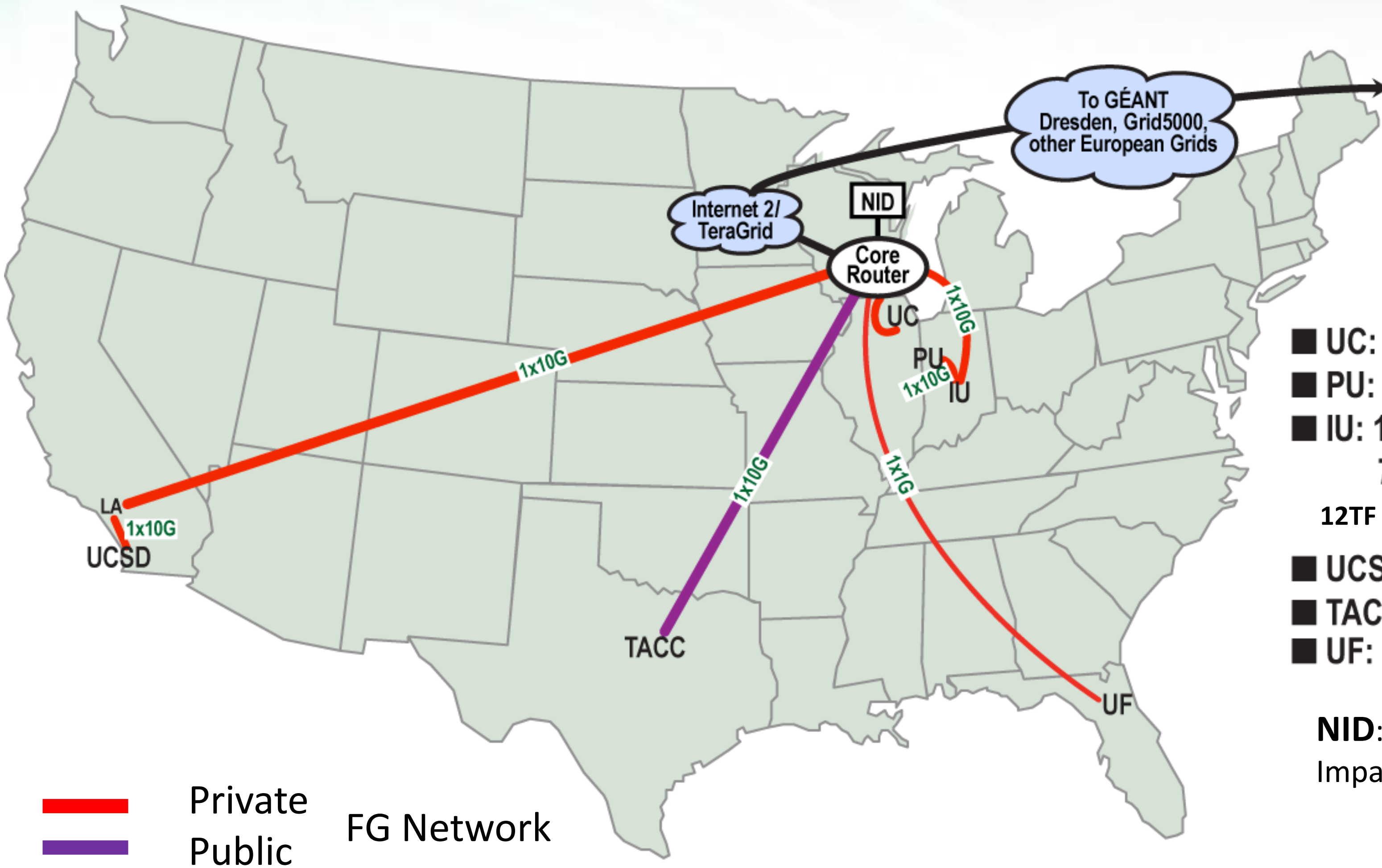
Future Grid https://portal.futuregrid.org

FutureGrid key Concepts II

- Rather than loading images onto VM's, FutureGrid supports Cloud, Grid and Parallel computing environments by provisioning software as needed onto "bare-metal" using Moab/xCAT (need to generalize)
  - Image library for MPI, OpenMP, MapReduce (Hadoop, (Dryad), Twister), gLite, Unicore, Globus, Xen, ScaleMP (distributed Shared Memory), Nimbus, Eucalyptus, OpenNebula, KVM, Windows …..
  - Either statically or dynamically
- Growth comes from users depositing novel images in library
- FutureGrid has ~4400 distributed cores with a **dedicated network** and a Spirent XGEM network fault and delay generator



FutureGrid: a Grid/Cloud/HPC Testbed



- UC: 7TF IBM 672 cores
- PU: 4TF Dell 384 cores
- IU: 11TF IBM 1024 cores
  7TF Cray 684 cores
  12TF Disk rich + GPU 512 cores
- UCSD: 7TF IBM 672 cores
- TACC: 12TF Dell 1152 cores
- UF: 3TF IBM 256 cores

NID: Network Impairment Device

4 Use Types for FutureGrid TestbedaaS

- **223** approved projects (968 users) July 14 2012
  - USA, China, India, Pakistan, lots of European countries
  - Industry, Government, Academia
- **Training Education and Outreach (10%)**
  - Semester and short events; interesting outreach to small universities
- **Computer science and Middleware (59%)**
  - Core CS and Cyberinfrastructure; Interoperability (2%) for Grids and Clouds; Open Grid Forum OGF Standards
- **Computer Systems Evaluation (29%)**
  - XSEDE (TIS, TAS), OSG, EGI; Campuses
- **New Domain Science applications (26%)**
  - Life science highlighted (14%), Non Life Science (12%)
  - Generalize to building **Research Computing-aaS**

Fractions are as of July 15 2012 add to > 100%

FutureGrid offers Computing Testbed as a Service

Research Computing aaS
- Custom Images
- Courses
- Consulting
- Portals
- Archival Storage

SaaS
- System e.g. SQL, GlobusOnline
- Applications e.g. Nastran, Fluent

PaaS
- Cloud e.g. MapReduce
- HPC e.g. PETSc, SAGA
- Computer Science e.g. Languages, Sensor nets

IaaS
- Hypervisor
- Bare Metal
- Operating System
- Virtual Clusters, Networks

FutureGrid Uses Testbed-aaS Tools
- Provisioning
- Image Management
- IaaS Interoperability
- IaaS tools
- Expt management
- Dynamic Network
- Devops

FutureGrid Usages
- **Computer Science**
- **Applications** and understanding **Science Clouds**
- **Technology Evaluation** including XSEDE testing
- **Education and Training**

https://portal.futuregrid.org

Research Computing as a Service

- Traditional Computer Center has a variety of capabilities supporting (scientific computing/scholarly research) users.
  - Could also call this **Computational Science as a Service**
- **IaaS, PaaS** and **SaaS** are lower level parts of these capabilities but commercial clouds do not include
  1) **Developing roles/appliances for particular users**
  2) Supplying **custom SaaS** aimed at user communities
  3) **Community Portals**
  4) Integration across disparate resources for data and compute (i.e. **grids**)
  5) **Data transfer** and network link services
  6) **Archival storage, preservation, visualization**
  7) Consulting on use of particular appliances and SaaS i.e. on particular software components
  8) Debugging and other problem solving
  9) Administrative issues such as (local) accounting
- This allows us to develop a new model of a computer center where **commercial companies operate base hardware/software**
- A combination of XSEDE, Internet2 and computer center supply 1) to 9)?

Summary

Using Science Clouds in a Nutshell

- **High Throughput Computing**; pleasingly parallel; grid applications
  - Includes CFD Parameter Exploration
- **Multiple users** (long tail of science) and **usages** (parameter searches)
- **Internet of Things** (Sensor nets) as in cloud support of smart phones
- (**Iterative**) **MapReduce** supports HPC and Clouds
- Exploiting **elasticity** and **platforms** (HDFS, Object Stores, Queues ..)
  - Combine HPC and Clouds in storage and programming
- **Exascale** likely to **leverage** many **Cloud** technologies
- Use **worker roles**, **services**, **portals** (gateways) and **workflow**
  - **Design new CFD Appliances**
  - **Reproducible science with appliances and virtual clusters**
- Can do **experiments** on **FutureGrid**
- Commercial clouds could **change role** of **computer support organizations**