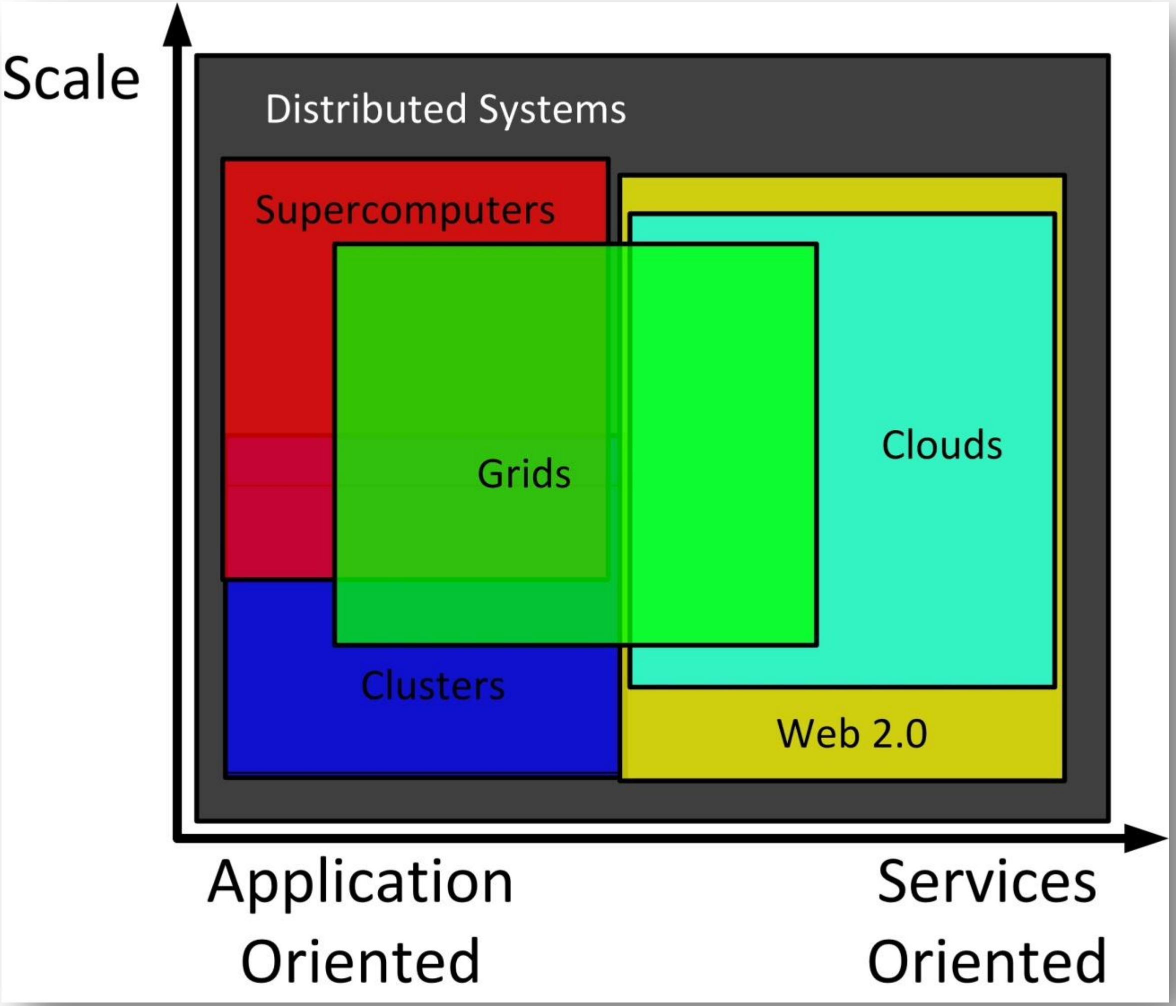
Cloud Computing and Grid Computing 360-Degree Compared

Ioan Raicu
Computer Science Department, Illinois Institute of Technology
Math and Computer Science Division, Argonne National Laboratory

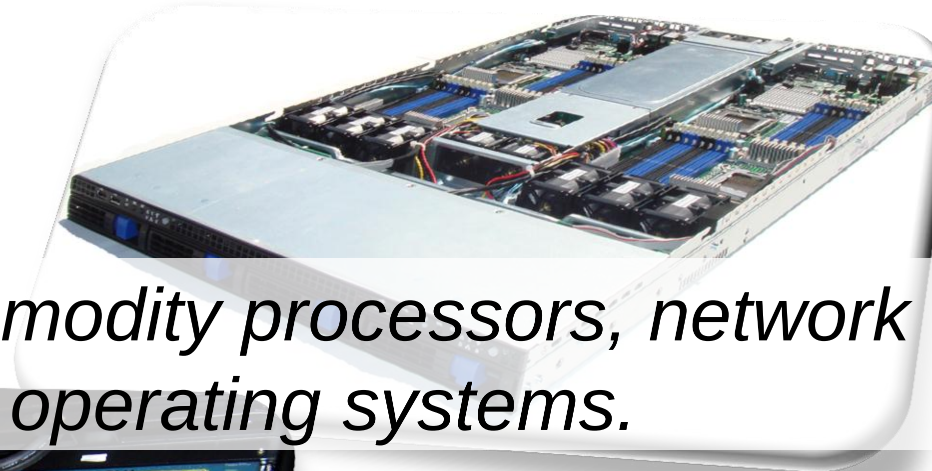
Loyola University
March 17th, 2011

Clusters, Grids, Clouds, and Supercomputers



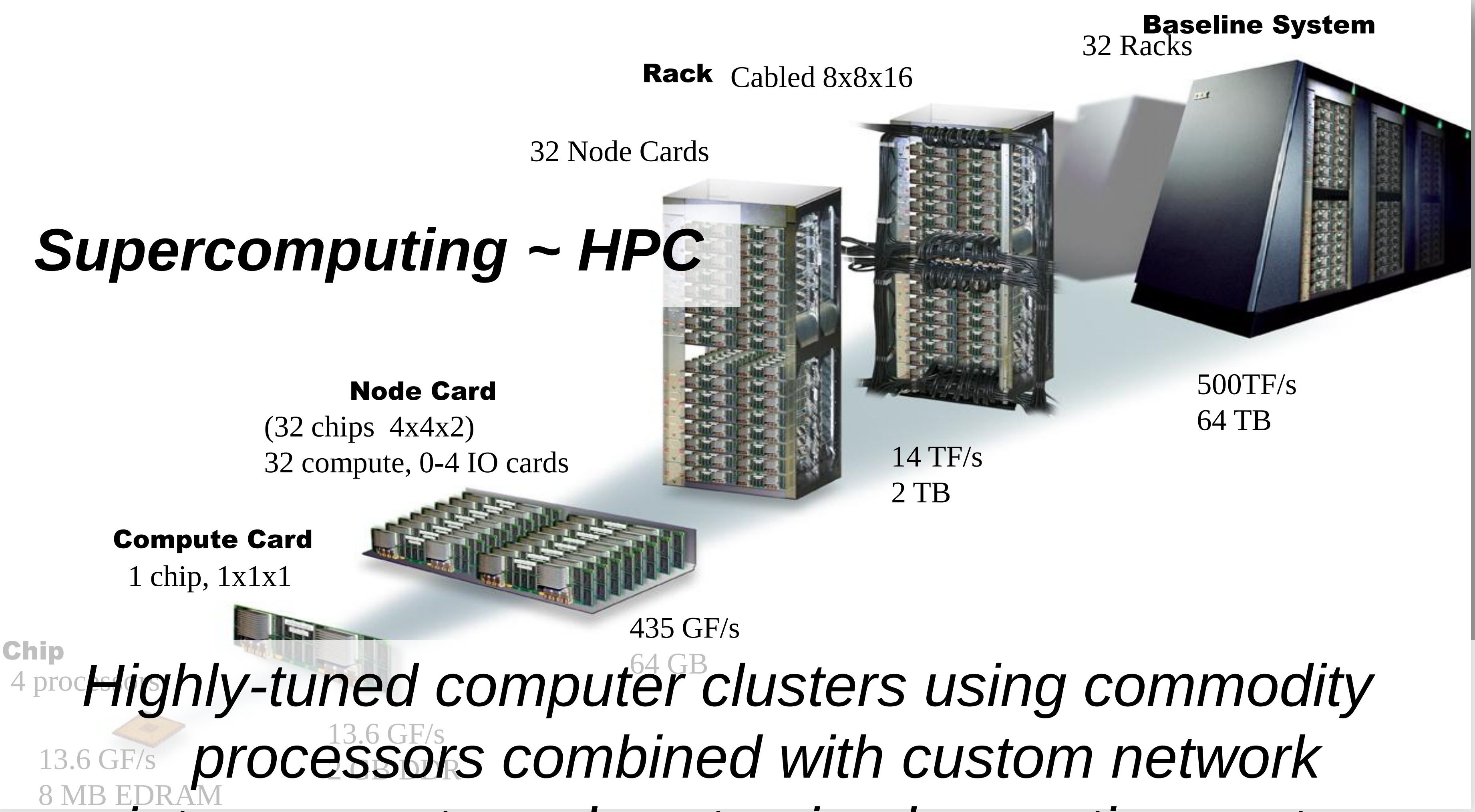
[GCE08] "Cloud Computing and Grid Computing 360-Degree Compared"

Cluster Computing



Computer clusters using commodity processors, network interconnects, and operating systems.

Supercomputing



Supercomputing ~ HPC

Highly-tuned computer clusters using commodity processors combined with custom network interconnects and customized operating system

Top 10 Supercomputers from Top500

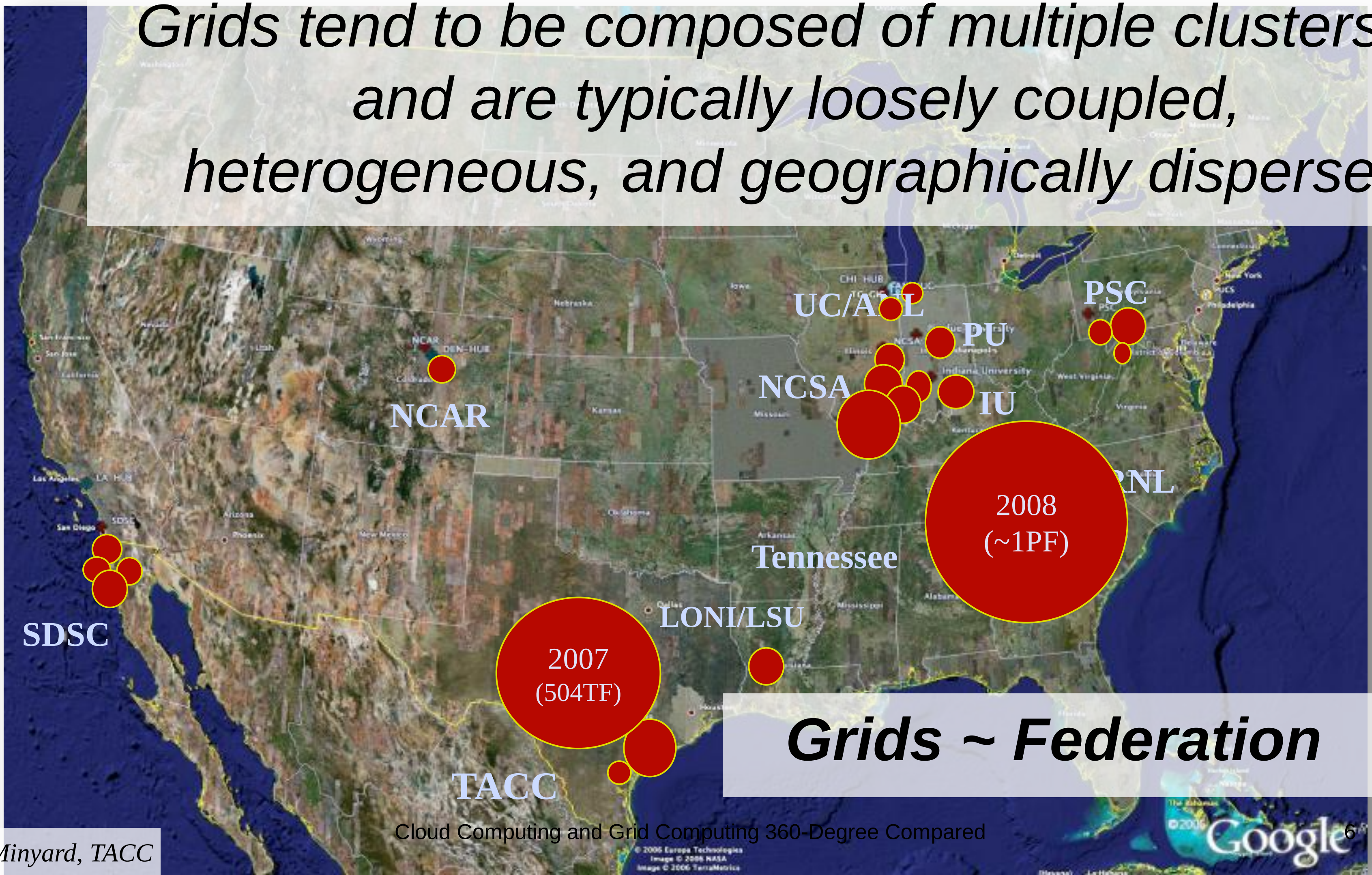
- Cray XT4 & XT5
  - Jaguar #1
  - Kraken #3
- IBM BladeCenter Hybrid
  - Roadrunner #2
- IBM BlueGene/L & BlueGene/P
  - Jugene #4
  - Intrepid #8
  - BG/L #7
- NUDT (GPU based)
  - Tianhe-1 #5
- SGI Altix ICE
  - Plaiedas #6
- Sun Constellation
  - Ranger #9
  - Red Sky #10





Grid Computing

Grids tend to be composed of multiple clusters, and are typically loosely coupled, heterogeneous, and geographically dispersed



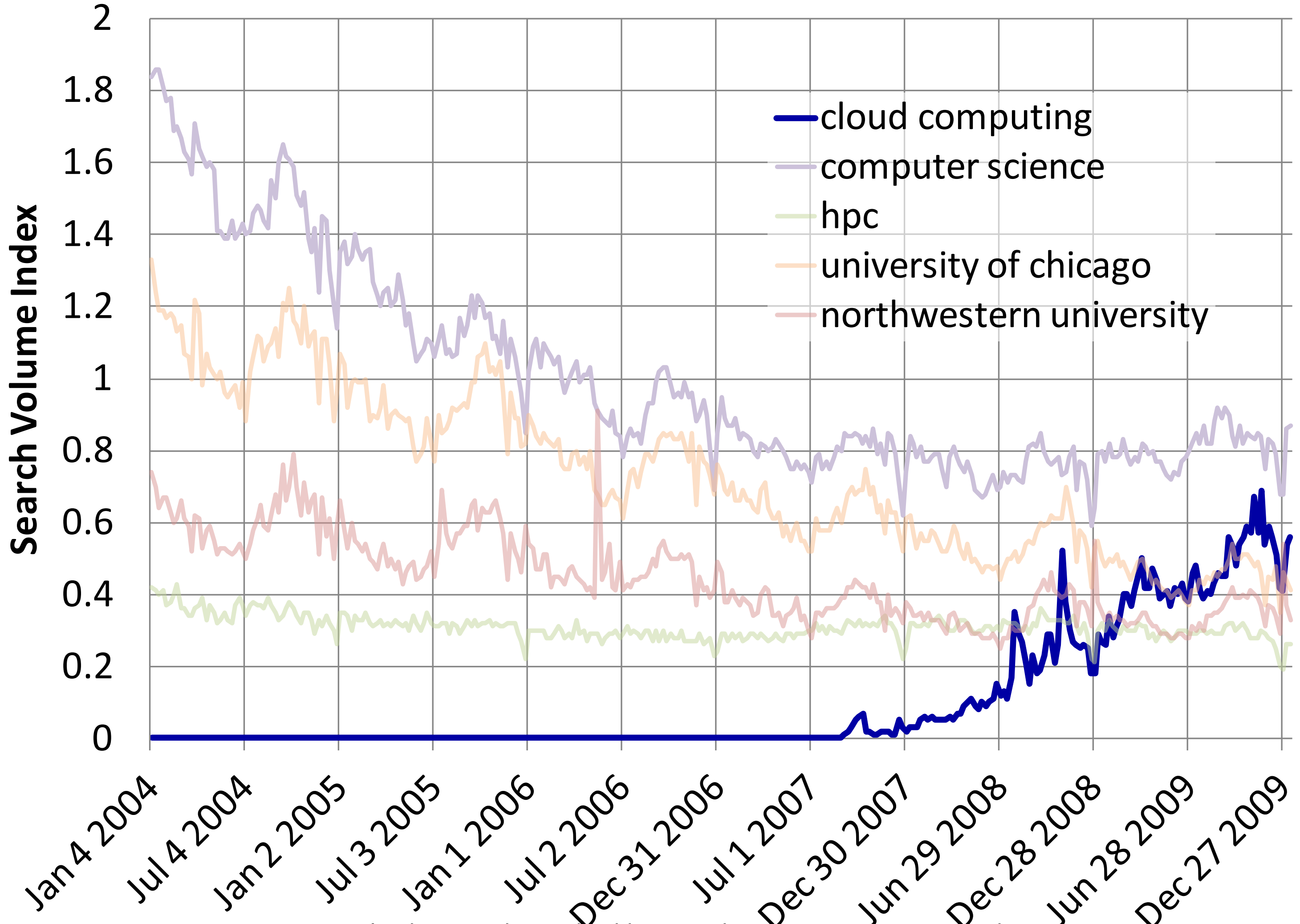
Grids ~ Federation

Tommy Minyard, TACC

Major Grids

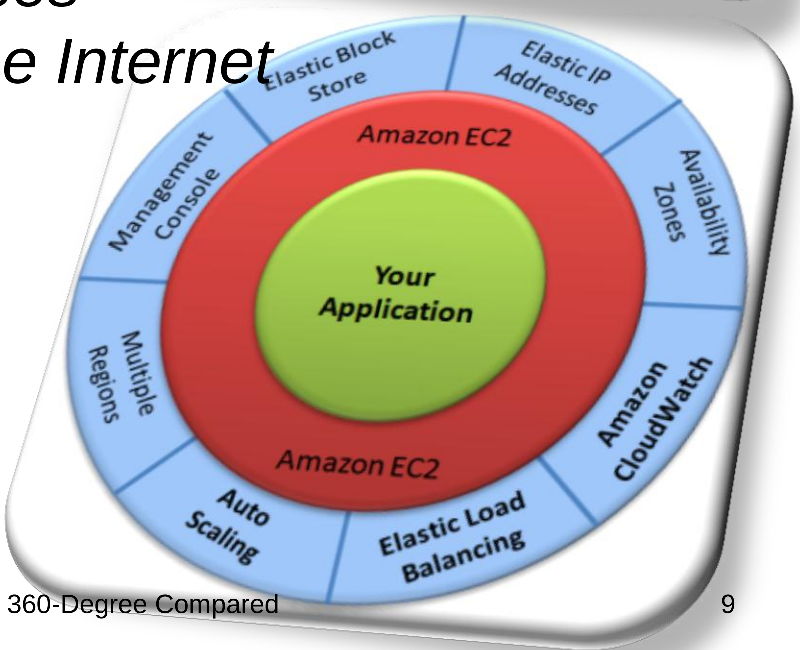
- TeraGrid (TG)
  - 200K-cores across 11 institutions and 22 systems over the US
- Open Science Grid (OSG)
  - 43K-cores across 80 institutions over the US
- Enabling Grids for E-sciencE (EGEE)
- LHC Computing Grid from CERN
- Middleware
  - Globus Toolkit
  - Unicore

Cloud Computing: An Emerging Paradigm



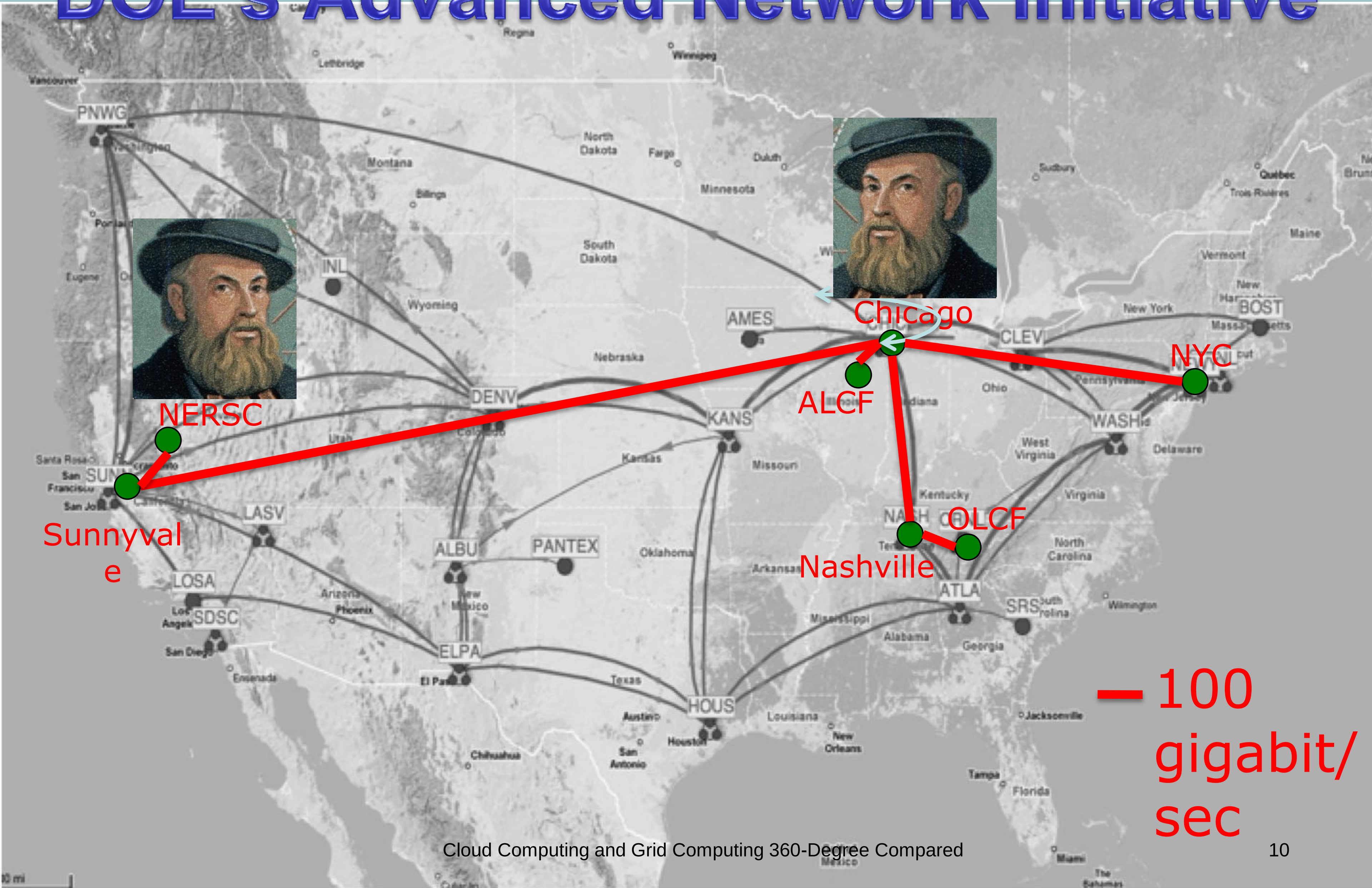
Cloud Computing

- *A large-scale distributed computing paradigm driven by:*
  1. *economies of scale*
  2. *virtualization*
  3. *dynamically-scalable resources*
  4. *delivered on demand over the Internet*



Clouds ~ hosting

Magellan + DOE's Advanced Network Initiative



Major Clouds

- Industry
  - Google App Engine
  - Amazon
  - Windows Azure
  - Salesforce
- Academia/Government
  - Magellan
  - FutureGrid
- Opensource middleware
  - Nimbus
  - Eucalyptus
  - OpenNebula

So is "Cloud Computing" just a new name for Grid?

- IT reinvents itself every five years
- The answer is complicated…

- **YES**: the vision is the same
  - to reduce the cost of computing
  - increase reliability
  - increase flexibility by transitioning from self operation to third party

So is "Cloud Computing" just a new name for Grid?

- **NO**: things are different than they were 10 years ago
  - New needs to analyze massive data, increased demand for computing
  - Commodity clusters are expensive to operate
  - We have low-cost virtualization
  - Billions of dollars being spent by Amazon, Google, and Microsoft to create real commercial large-scale systems with hundreds of thousands of computers
  - The prospect of needing only a credit card to get on-demand access to *infinite computers is exciting; *infinite<O(1000)

So is “Cloud Computing” just a new name for Grid?

- **YES:** the problems are mostly the same
  - How to manage large facilities
  - Define methods to discover, request, and use resources
  - How to implement and execute parallel computations
  - Details differ, but issues are similar

Outline

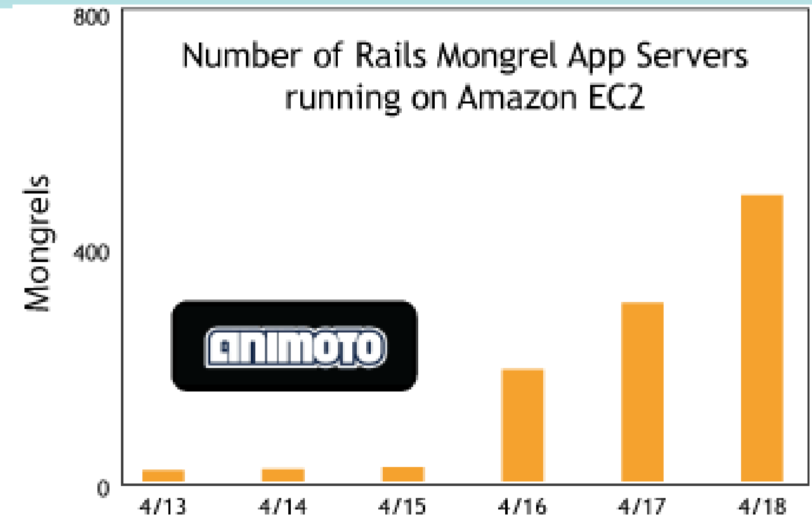
- Business model
- Architecture
- Resource management
- Programming model
- Application model
- Security model

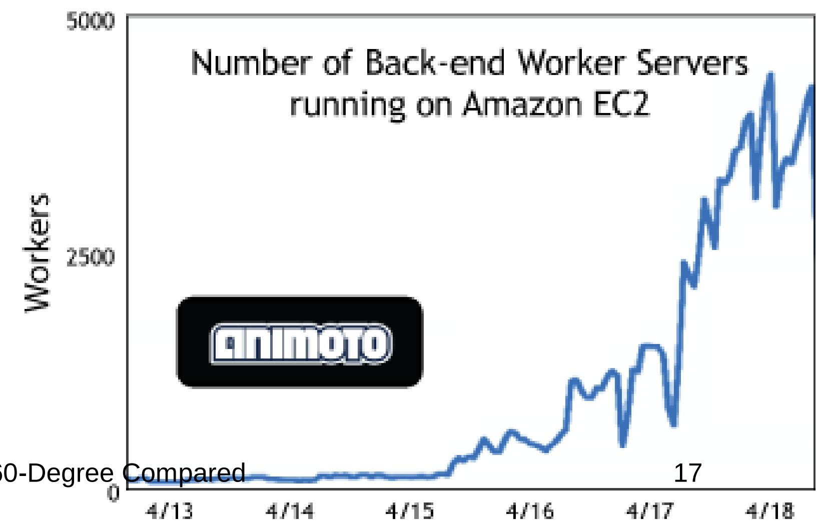
Business Model

- Grids:
  - Largest Grids funded by government
  - Largest user-base in academia and government labs to drive scientific computing
  - Project-oriented: service units
- Clouds:
  - Industry (i.e. Amazon) funded the initial Clouds
  - Large user base in common people, small businesses, large businesses, and a bit of openn science research
  - Utility computing: real money

Business Model
Why is it a big deal?

- Why is this a big deal?
  - No owned infrastructure
  - All resources rented on demand
- Critical for startups with risky business plans
- Not possible without Cloud Computing and a credit card
  - Launched in 2007/2008 timeframe



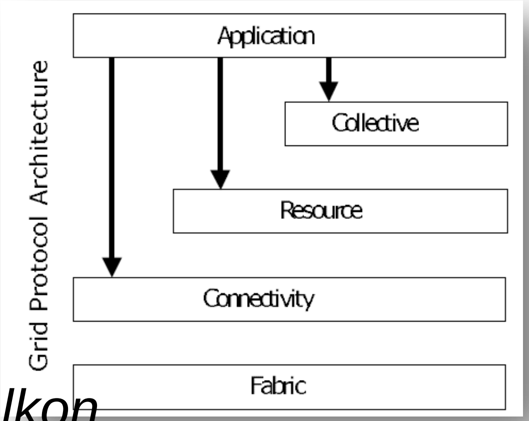


An Example of an Application in the Cloud

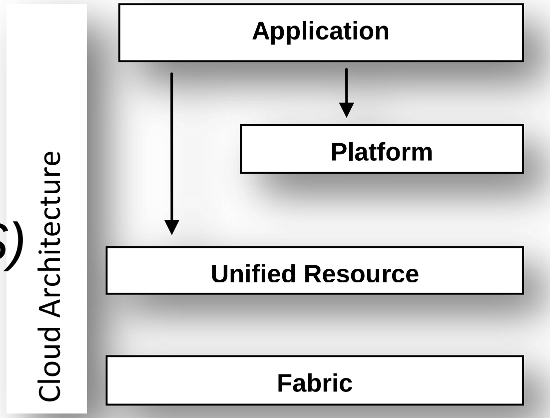
- [Animoto]
  - Makes it r... ate videos with t...

Architecture

- Grids:
  - Application: *Swift, Grid portals (NVO)*
  - Collective layer: *MDS, Condor-G, Nimrod-G*
  - Resource layer: *GRAM, Falkon, GridFTP*
  - Connectivity layer: *Grid Security Infrastructure*
  - Fabric layer: *GRAM, PBS, SGE, LSF, Condor, Falkon*



- Clouds:
  - Application Layer: *Software as a Service (SaaS)*
  - Platform Layer: *Platform as a Service (PaaS)*
  - Unified Resource: *Infrastructure as a Service (IaaS)*
  - Fabric: *IaaS*



Resource Management

- Compute Model
  - batch-scheduled vs. time-shared
- Data Model
  - Data Locality
  - Combining compute and data management
- Virtualization
  - Slow adoption vs. central component
- Monitoring
- Provenance

Programming and Application Model

- Grids:
  - Tightly coupled
    - High Performance Computing (MPI-based)
  - Loosely Coupled
    - High Throughput Computing
    - Workflows
  - Data Intensive
    - Map/Reduce
- Clouds:
  - Loosely Coupled, transactional oriented

Programming Model Issues

- **Multicore** processors
- Massive **task parallelism**
- Massive **data parallelism**
- Integrating **black box applications**
- Complex **task dependencies** (task graphs)
- **Failure**, and other execution management issues
- **Dynamic task graphs**
- Documenting **provenance** of data products
- **Data management**: input, intermediate, output
- **Dynamic data access** involving large amounts of data

Gateways

- Aimed to simplify usage of complex resources
- Grids
  - Front-ends to many different applications
  - Emerging technologies for Grids
- Clouds
  - Standard interface to Clouds

An Example of an Application in the Grid

Security Model

- Grids
  - Grid Security Infrastructure (GSI)
  - Stronger, but steeper learning curve and wait time
    - Personal verification: phone, manager, etc
- Clouds
  - Weaker, can use credit card to gain access, can reset password over plain text email, etc

Conclusion

- Move towards a mix of micro-production and large utilities, with load being distributed among them dynamically
  - Increasing numbers of small-scale producers (local clusters and embedded processors—in shoes and walls)
  - Large-scale regional producers
- Need to define protocols
  - Allow users and service providers to discover, monitor and manage their reservations and payments
  - Interoperability

Conclusion (cont)

- Need to combine the centralized scale of today's Cloud utilities, and the distribution and interoperability of today's Grid facilities
- Need support for on-demand provisioning
- Need tools for managing both the underlying resources and the resulting distributed computations
- Security and trust will be a major obstacle for commercial Clouds by large companies that have in-house IT resources to host their own data centers

More Information

- More information:
  - http://www.cs.iit.edu/~iraicu/
  - iraicu@cs.iit.edu
- Questions?