From Grid to Cloud, the STAR Experience

Jérôme Lauret
S&C project Leader @ RHIC/STAR

SciDAC
Scientific Discovery through Advanced Computing

BROOKHAVEN NATIONAL LABORATORY

Outline

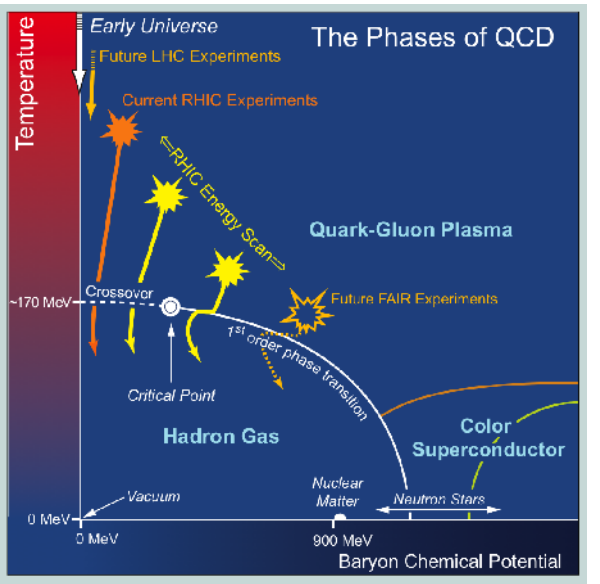
Focused on “our” usage (standard NP workflow), not a review of Clouds capabilities and services …

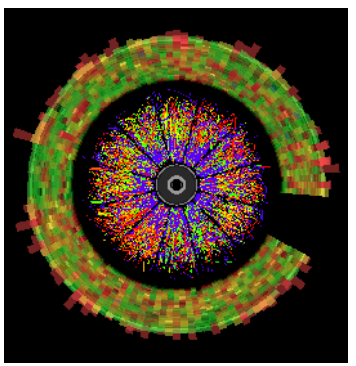
- Introduce STAR & data challenge
- Path from Grid to Cloud, problem analysis
- Cloud and virtualization, usage and tested models pro & cons
- Concluding remarks

Introduction

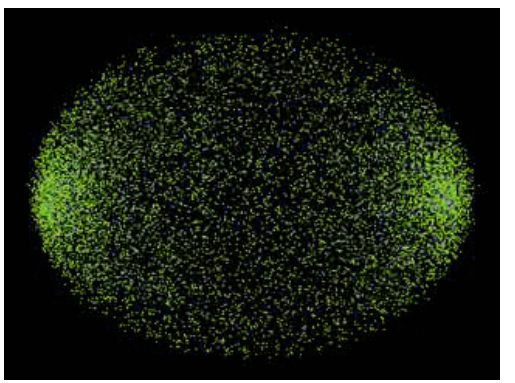
The Soleinoidal Tracker At RHIC (STAR) experiment

- **A Nuclear Physics experimental groups part of the Relativistic Heavy Ion Collider (RHIC) program located at BNL**
  - Provide unique insight into how quarks and gluons behaved collectively at the very first moment our universe was born.
  - Understand how mass and spin combine into building blocks of nature
  - Help study the fundamental principles of Physics leading to symmetry breaking, help study the nuclear equation of state



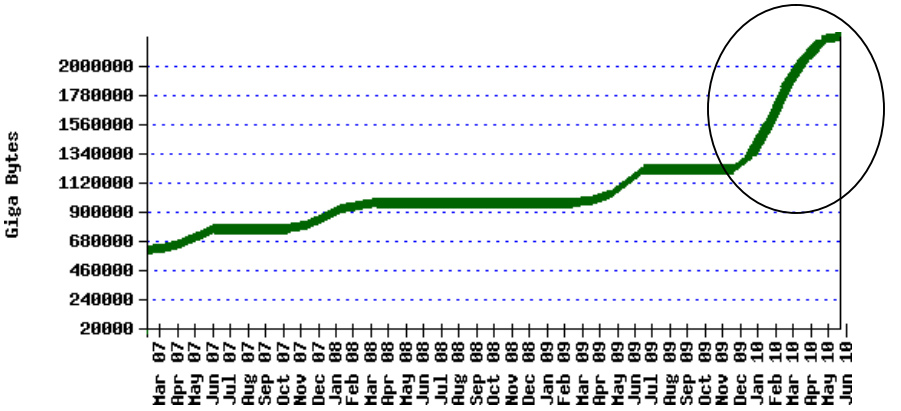


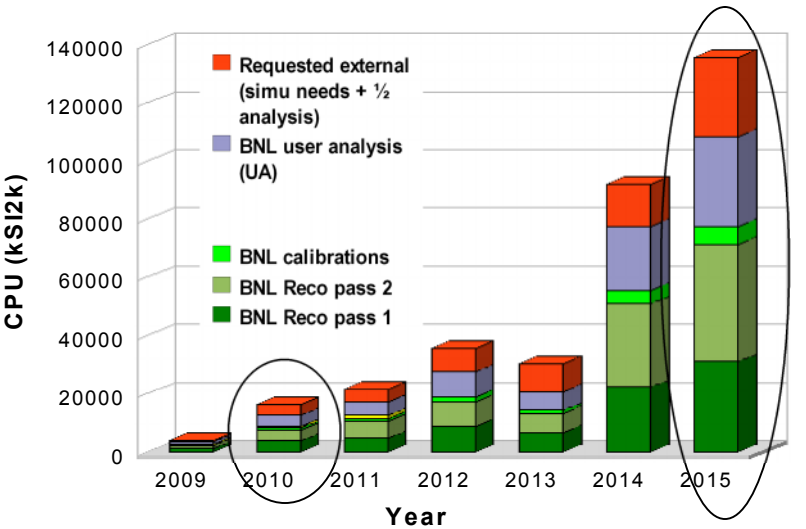
Time machine



STAR, a data challenge

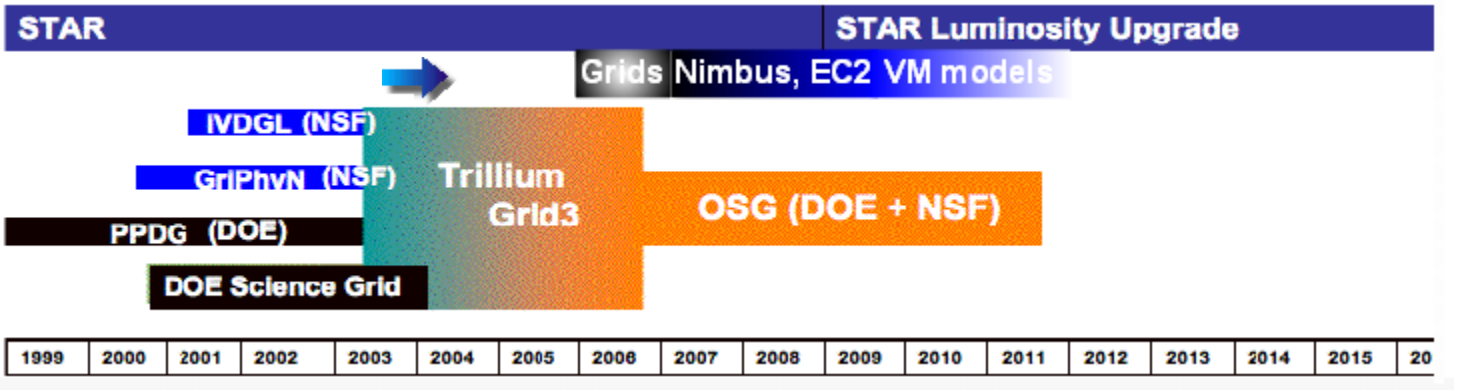
- A Peta-Scale (data) experiment with lots of computational challenges
  - Year 10 data (experiment ended a month ago) = ∑ data for all previous years
- Resource need projections
  - User analysis and real data production are the Highest resource demand
  - Constrained to ONLY 2 pass data reconstruction (anything else needs to be outsourced)
  - **Must outsource simulation + ½ of user analysis**
- Uncertainties in estimates?
  - Large data sets bring interesting challenges: **moving from a statistically challenged to a systematic driven precision regime**
  - **Quantification of uncertainties → additional simulations?**





STAR resource planning document 2009-2015, **CSN0474**

STAR, Grids and …



- Computing for the RHIC experiments **(CHEP 2009)**
- Nimbus cloud project saves brainiacs' bacon **(TechTarget - 2009)**
- Number Crunching Made Easy **(Newsweek - 2009)**
- Clouds make way for STAR to shine **(iSGTW Feature - 2009)** **"**Last minute" need fulfilled
- Nimbus … Meet STAR Production Demands **(HPCWire - 2009)**
- The new Nimbus: first steps in the clouds **(iSTGTW - 2008)** First use of cloud for MC
- Integrating X-Grid into the HENP distributed computing model **(CHEP 2007)**
- SunGrid "utility computing" **(CHEP 2007)**
- OSG SUMS Workspace Demo (**CHEP 2007**) First interest in Cloud
- SunGrid and the STAR Experiment (**Sun.com - 2006**)

BROOKHAVEN NATIONAL LABORATORY

STAR

From Grids to Cloud

Grid, success and limitations

- What are we doing Grid-wise?
  - **Data transfer** in STAR: bulk transfers using BeStMan/SRM (NERSC/PDSF) since 2002, Transfer to China in 2004 (picoDST), Routine transfer to Prague in 2008 (FDT), "Raw" Grid ftp (KISTI/Korea, …) @ 1 Gb/sec sustained, …
  - **Development and/or hardening of middleware**: SRM, BitMap indexing, distributed data model and access, Meta-Scheduler, planner, …
  - **Development of infrastructure for job submission**: Efficiency in pre 2006 ~ 65%, 2006-2007 85%, 2008 90%, today @ 97%+ (improvement due to operational support including OSG)

Open Science Grid

- Achievements
  - STAR Monte-Carlo productions moved nearly all on Grid: 2006
  - In 2010, still only MC & seldom use overall (~ 64k hours/week)
  - Nearly all on dedicated sites (software stack pre-installed)

BROOKHAVEN NATIONAL LABORATORY

Problem Analysis

- Why not user analysis or real data productions? Where are the problems for production environments? Why dedicated sites?
- Technical reasons
  - **Grids are complex** and **too heterogeneous** for science production environment
    - Troubleshooting is inadequate. Messages cryptic, plethora of OS and environment + lack of interactivity exacerbate the problem
  - **Experimental software stacks are complex + Deployment require customized environments**
    - STAR case: Developed over more than 10 years, by more than 100 scientists, comprises ~ 2.5 M lines: Rely on the right combination of compiler versions and available libraries + Dynamically load external libraries depending on the task to be performed (system or third parties: ROOT, mysql, libxml, …)
- Physics and staffing reasons
  - **Compiling "on the fly" impossible + Code validation and regression tests are essential**
    - Heterogeneous platforms → homogeneous results
    - Cannot be done on all OS flavors (**workforce considerations**)
  - Science evolves, need to re-validate past data
    - How do I go back to an old library release and run on new OS & compilers ? [not always portable] … 10 years down the road

BROOKHAVEN NATIONAL LABORATORY

Software and OS complexity way out?

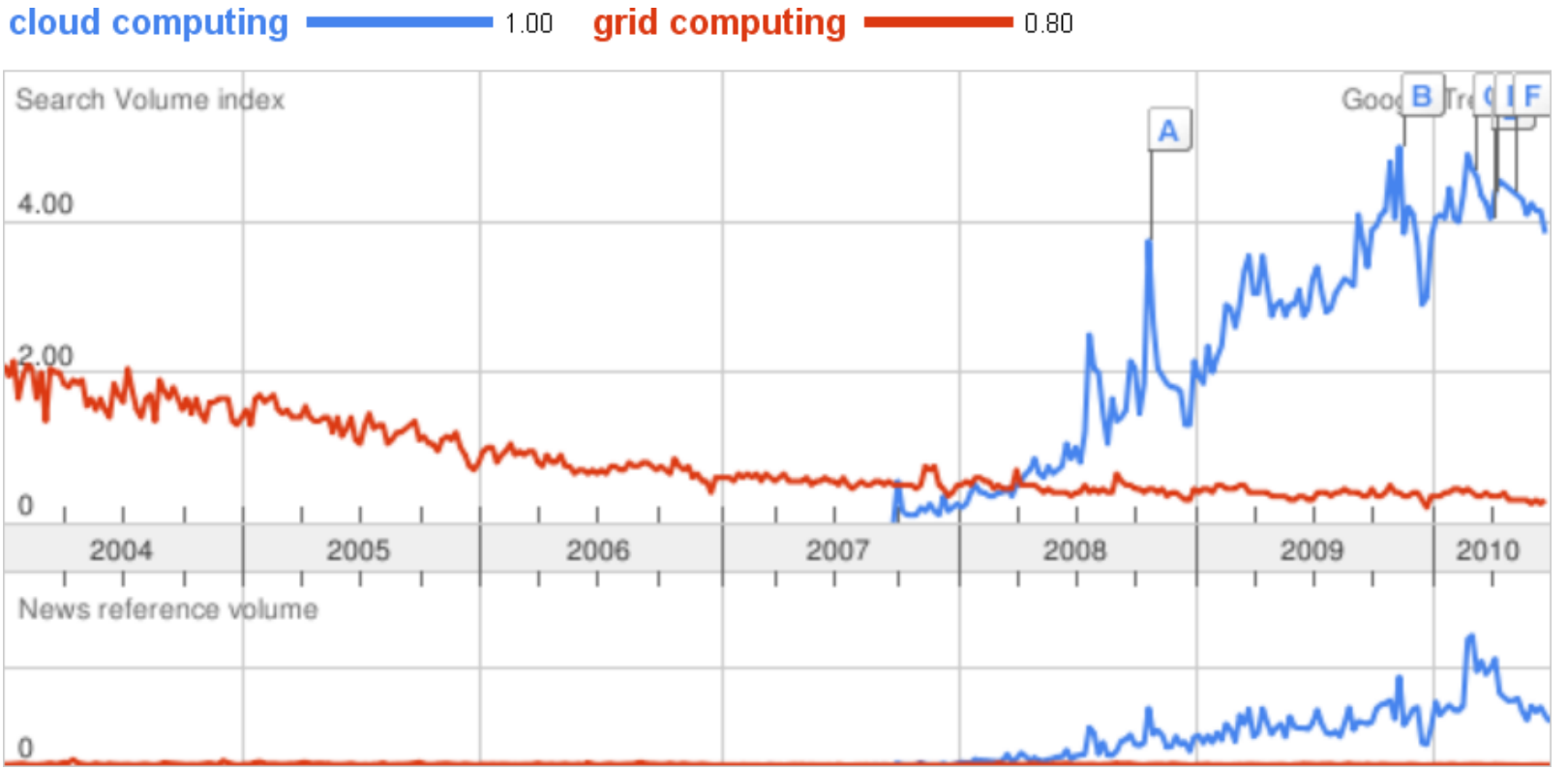
- Burst of resources difficult to acquire
  - Could Grids be dynamic and versatile? Yes if they acquire "truly opportunistic" characteristics
  - **Can virtualization help?**
    - VM is "canned": **Has all what I need to run "inside"**, could have all the services, etc …
    - … and the answer is **YES IT CAN!!** [in the opportunistic usage dimension at least]

  - **Virtualization displaces the problem**
    - VM machinery layer needs maintenance for long term support
    - AND/OR "Translator" between VM technologies are needed: Xen, KVM, VMWare, …
  - **Virtualization and/or Cloud are NOT silver bullet to operational support**
    - Troubleshooting and monitoring remain essential

Open Science Grid

Clouds have VM machinery at the core – one problem down, dozen more to go … **Lots of interest in Cloud computing …**

Quantifying interest?



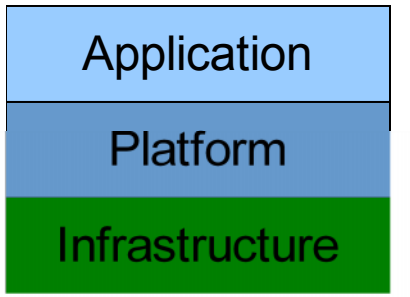
Google trends – based on community search assumed to be proportional to interest

Clouds testing & tasting ☺ …

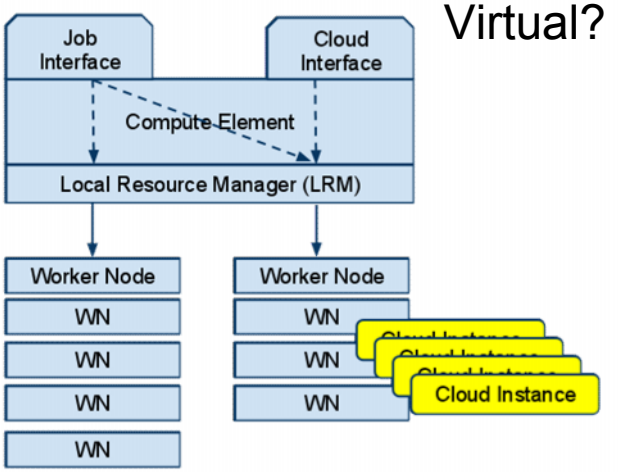
General remarks

- Does the anatomy of Clouds matter?
  - Keywords: SaaS, PaaS, IaaS …
  - Grid: request job slots – Cloud: request VM instances
  - Cloud are economic driven & de-localized
    - Pay as you go, pay for what you need, share the infrastructure, public "utility" service
    - Geographic boundaries less clear
- What do I choose?
  - Many providers, many stacks …
  - Amazon EC2, SGI Cyclone, IBM CloudBurst, …, Magellan (DOE), Azure (NSF), …
  - Many emerging technologies: Nimbus, Eucalyptus, Cloudera, …
- What did we test?
  - Amazon/EC2 native interface
  - Amazon+Nimbus or Nimbus+Grid resources
  - Clemson Virtual Organization Cluster (VOC) model
  - Condor/VM scheduling (GLOW)
  - Clemson Kestrel model

  …



Virtual?

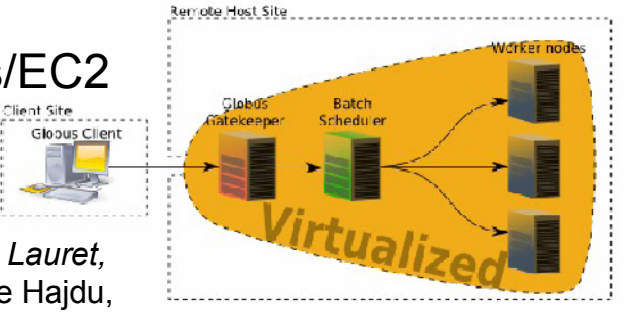


Models – virtualization boundaries

STAR @ MIT, *Adam Kocoloski*
Jan Balewski, Mathew Walker

Purely Web based + ssh login possible. WN "see" the world

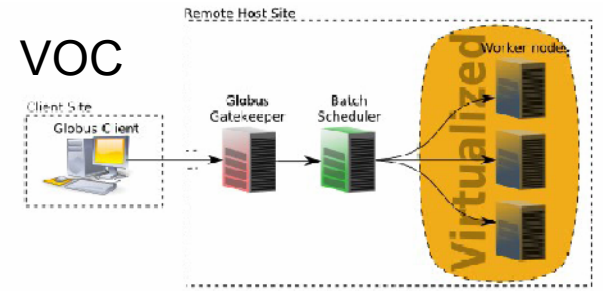
Nimbus/EC2



Kate Keahey, Jérôme Lauret,
Tim Freeman, Levente Hajdu,
Lidia Didenko

Gatekeeper + WN form a virtual cluster. WN "see" the world

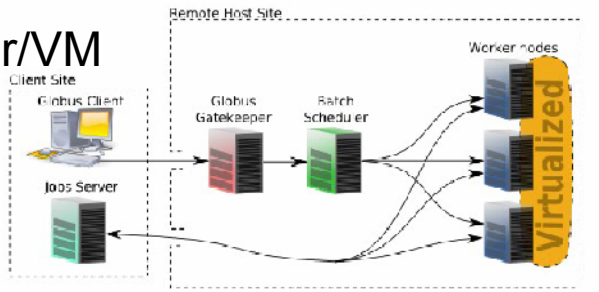
VOC



Sebastien Goasgen, Jérôme Lauret,
Michael Fenn, Levente Hajdu

"on-demand" VM subscribe to external RMS. VMs forms an additional network layer

Condor/VM



Miron Livny, Greg Thain, Jan Balewski,
Matthew Walker*, Jérôme Lauret*

Semi-standard GK used to start VMs. Private IP space, need SE + start/stop mechanism for VMs

BROOKHAVEN NATIONAL LABORATORY

Models

- Amazon/EC2 native
  - **Efficiency exceeds 99%** in O2 scale; medium instance; bound to ~ 5 MB/sec /node – price/performance not always clear – IO rather inadequate for large scale efforts
  - Good for simulations and simple workflows: little "I", not that much "O" in IO – unlikely suitable for HPC/HTC or large data mining
  - **Key points, advantages and caveats**
    - **Amazon has a concept of VM repository:** Ownership and trust
    - **Amazon AAA rudimentary (lacking?):** AA especially – used SSH keys or myproxy with image having "proper" gsi components
    - **Amazon has a simple and competitive pricing model:** $0.09 / hour in our case - A 100 jobs, week long simulation cost ~ $1,510. A year long CPU @ 100 jobs saturation ~ 79k$ - ATTENTION: S3 cost not advantageous
- Nimbus/EC2
  - **Efficiency 85% first submission ; 97%+ for one failure re-submission**
    - Drop mostly due to batch system and scalability of PBS – may be improved
  - Same target – simple simulation workflows, not much HPC/HTC
  - **Key points, advantages and caveats**
    - **OSG stack inside, GK+WN - virtual space looks like "another OSG site"**
    - **Creation of "clusters" made easier**
    - **Some contextualization to make at startup (GK not known a-priori, batch "inside" need to know topology)**

BROOKHAVEN NATIONAL LABORATORY

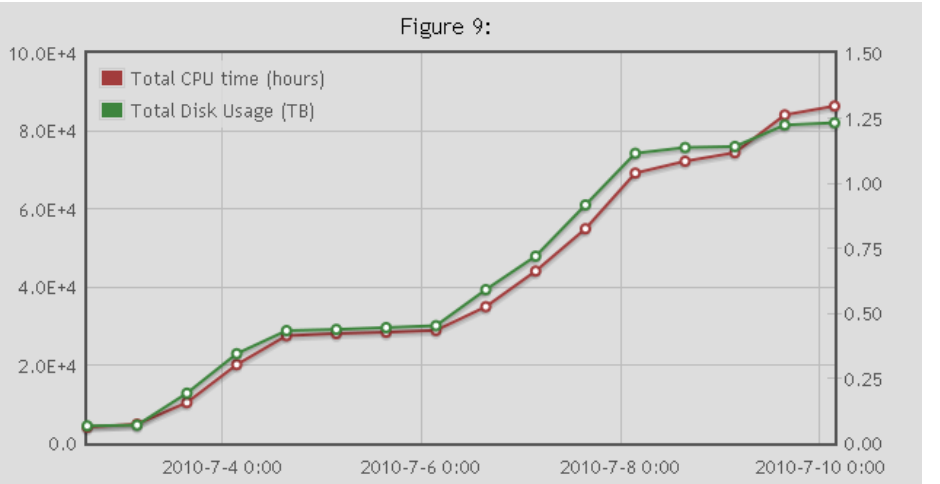
STAR

Models

- Virtual Organization Cluster (VOC) [ACAT 2010]
  - **Efficiency: 100%?** – not a single loss, extremely stable, no stress test
  - Usability: ultimate convenience and transparency, simulation (transfer is Site/Site limited)
  - **Key points, advantages and caveats**
    - **Interface is standard Grid – user is agnostic of technology**
    - **Contextualization remains a site specific overhead**
    - **VM instance appear/disappear as demand grows/decrease – transparent**
      - Cluster is shared between native/virtual "on demand" + excellent tracking of demand/provisioning
    - Lesson learn
      - **Performance dramatically improved by caching image** locally OR directing changes to local disk – not possible to control on EC2. Final overhead < 1% ; near immediate job startup.
      - **VM on top of IaaS – IP address space problem**
- Condor/VM
  - **Efficiency: unclear but ~ 80-85% top**
    - 10% of the VM never started, 15% stopped (crashed), 5% net loss for long simulation jobs (VM reboot every 24 hours). Need to be able to extend lease?
  - Usability: was very useful at 500 VMs for full simulation, external transfer mechanism to SE
  - **Key points, advantages and caveats**
    - **Interface remains grid-like – After VM is started, no real job get "inside" – need supplemental "pull model" (not self-sufficient)**
    - **As many VMs as one wants: nearly no contextualization (apart from SE) reduce overheads on local staff, condor steering**
    - **IP space is local – no connection to outside**
    - **Need to handle data transfer separately**

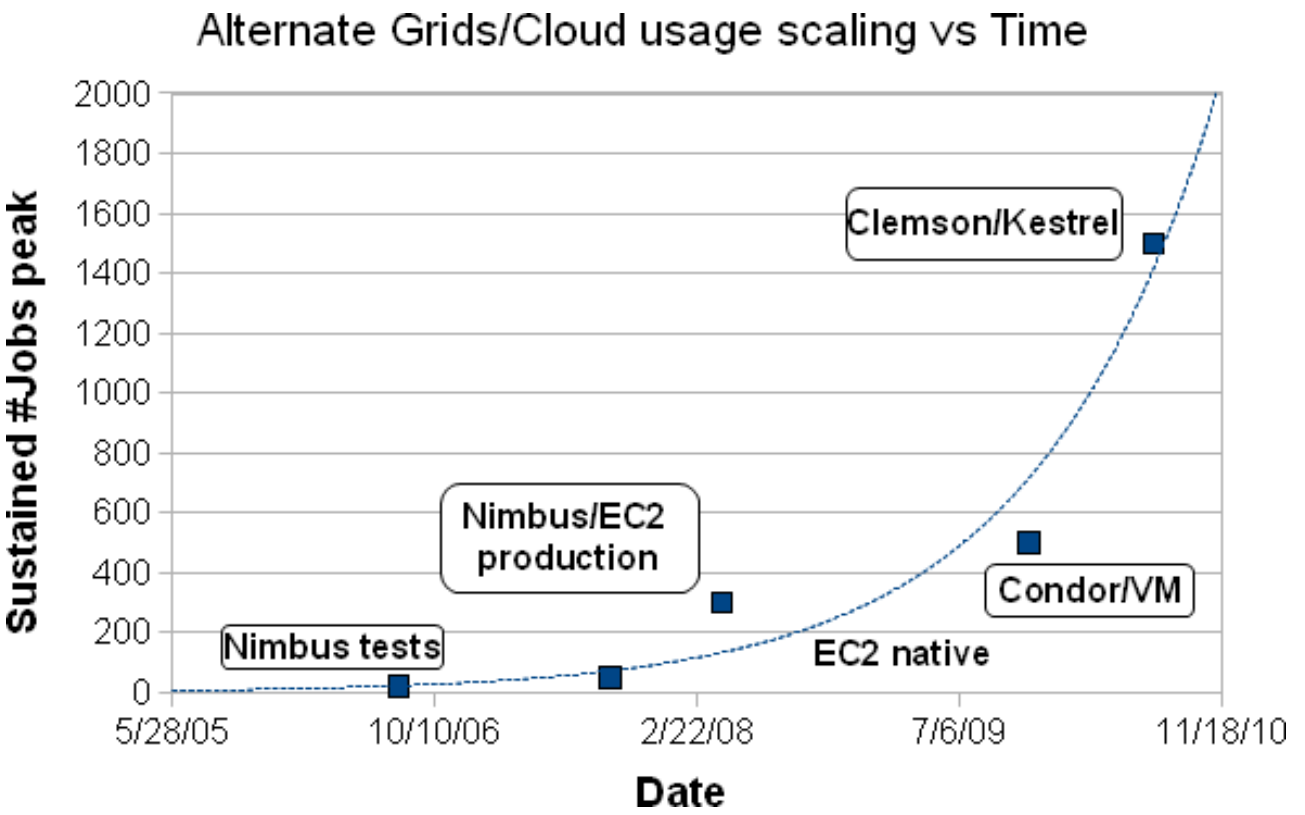
Models

- Clemson/Kestrel
  - **Efficiency: unknown**
  - Usability: model testing
    - Additional feature in this test – start a MySQL service “within” (data production requires one and so are detector response simulations)
    - Additional: 0.5 GB of local space



  - **Key points**:
    - VM may be dynamic
    - IP local BUT Kestrel **allows “IM” like communication inside** – after VM is started, there is a way to start what you want and stop / restart VMs or expand demand
    - **Contextualization can be a standard** Kestrel deployment (would be true of any standardized model)
    - **Mixing nodes from Clemson and CERN** in our test (**working example**)

Testing scale & usage growth



Today, 1,000 jobs on Cloud or Hybrid ("virtualized Grids") is possible (some challenges with stability / scalability)

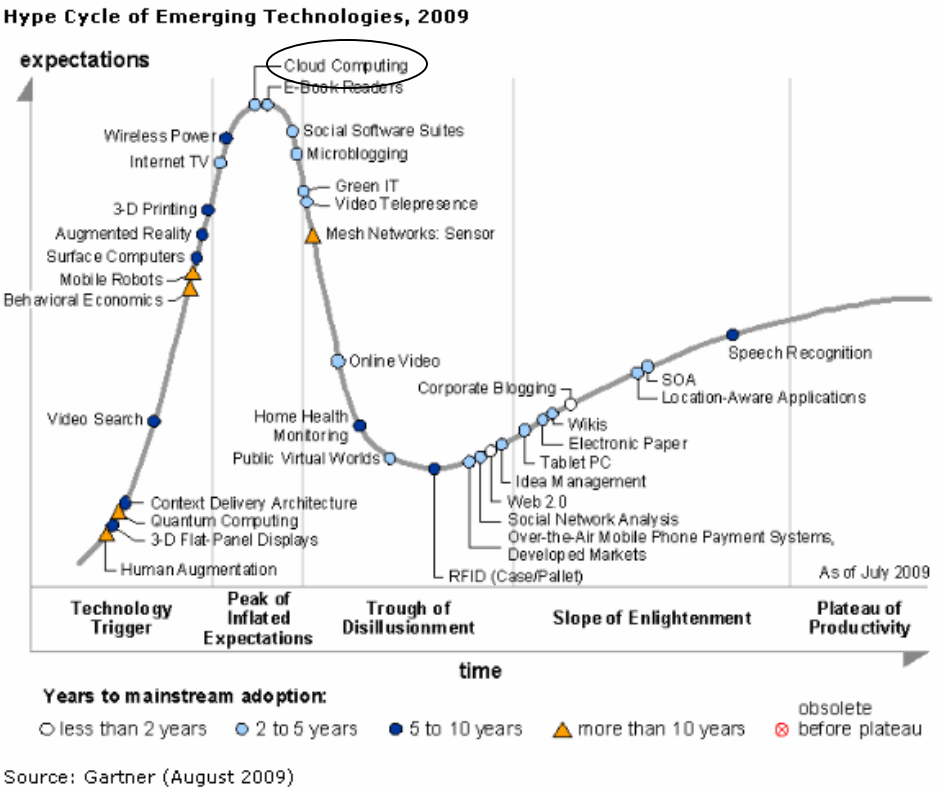
10 k to 100 k jobs needed for STAR, OSG ~ 13 M jobs at times

Promising ... long way to go …

Toward summary & conclusions

Note of caution …

- Top of hype curve
  - Will need "help" beyond faith for steering a constructive direction so (a) survive the fall through the shadow of the valley of death (i.e. disillusion) (b) be truly useful to science communities and meet expectations
- Cloud are here and likely to stay …
  - Amazon efficiency comparable to Grid efficiency (scalability not-tested)
    - Commercial and private clouds appearing like mushrooms …Prices are competitive for simple workflows
    - July 13th : EC2 "Cluster Compute" instance now available (1.6 TB store; 23 GB of mem; 1.60$ / hour; 10 Gb network), rated 146 / top 500 – price assumed to be as low as 0.56 $ / hour with discount
  - Attractive parts with VM: not a dream, "opportunistic" usage IS possible
  - May help smoothing resource gaps across national laboratories
    - You need a VM + resource providers
    - What is not used "there" can be used (modulo contextualization)



Source: Gartner (August 2009)

Points summary …

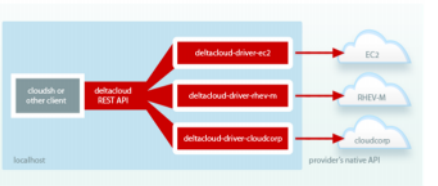
- Identified issues
  - VM repositories (trust and ownership) & caching of VMs (efficiency)
  - Authentication, Authorization, Accountability (security model), who authorized a "cluster" to join a VO resource? How to start a service and be approved to "join" a global monitoring system?
  - Payment model (economic model now possible?)
  - Format of VM and easy portability across sites (standardization?), dynamic and elasticity feature needed (not need to know), contextualization made easy, image (format) evolution
  - IO in/out of VM (SE) considering VM de-localization (a "VC" may be truly distributed)
  - Standard interface and plug-and-play approach
  - Service scalability, truly distributed services

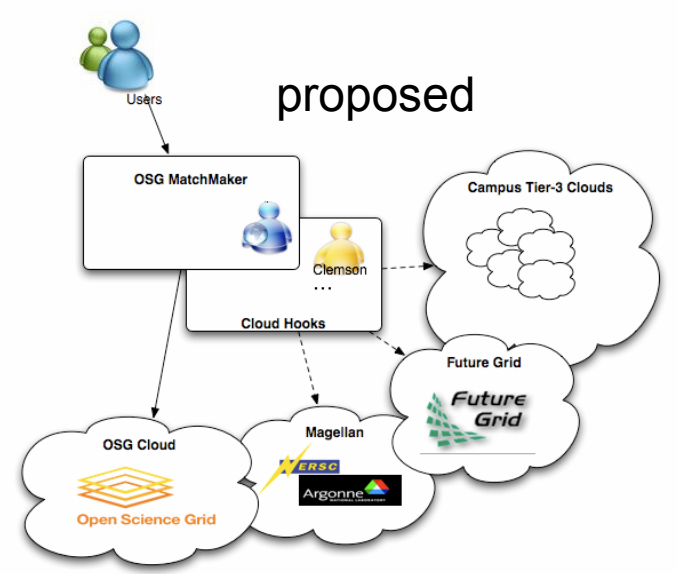
- Grid+VM or Cloud: Application and environment moves with you + (near) infrastructure independent approach
  - **Experiment "hand off" a container** → Easy software provisioning of TierX, X>0, ease of use
    - Updates of OS / software stacks will still be driven by Tier0 … problem of support reduced for facilities
  - **Facilities may "carve" a piece of their medium-size clusters**
    - Exascale? Maintaining "commodity" hardware cluster … tomorrow becoming specialized … should be an obsolete infrastructure approach.
  - **New notion of "clusters"** – can serve vast amount of communities
    - Ex: PDSF @ NERSC overlap with other clusters?
  - **Need for dedicated High Performance facilities will remain**
    - Ex: 3 Gb/sec to HPSS @ BNL, close to experimental data taking a real need (Amazon or private Clouds unlikely to ever support this) + Cost of storing PB of data

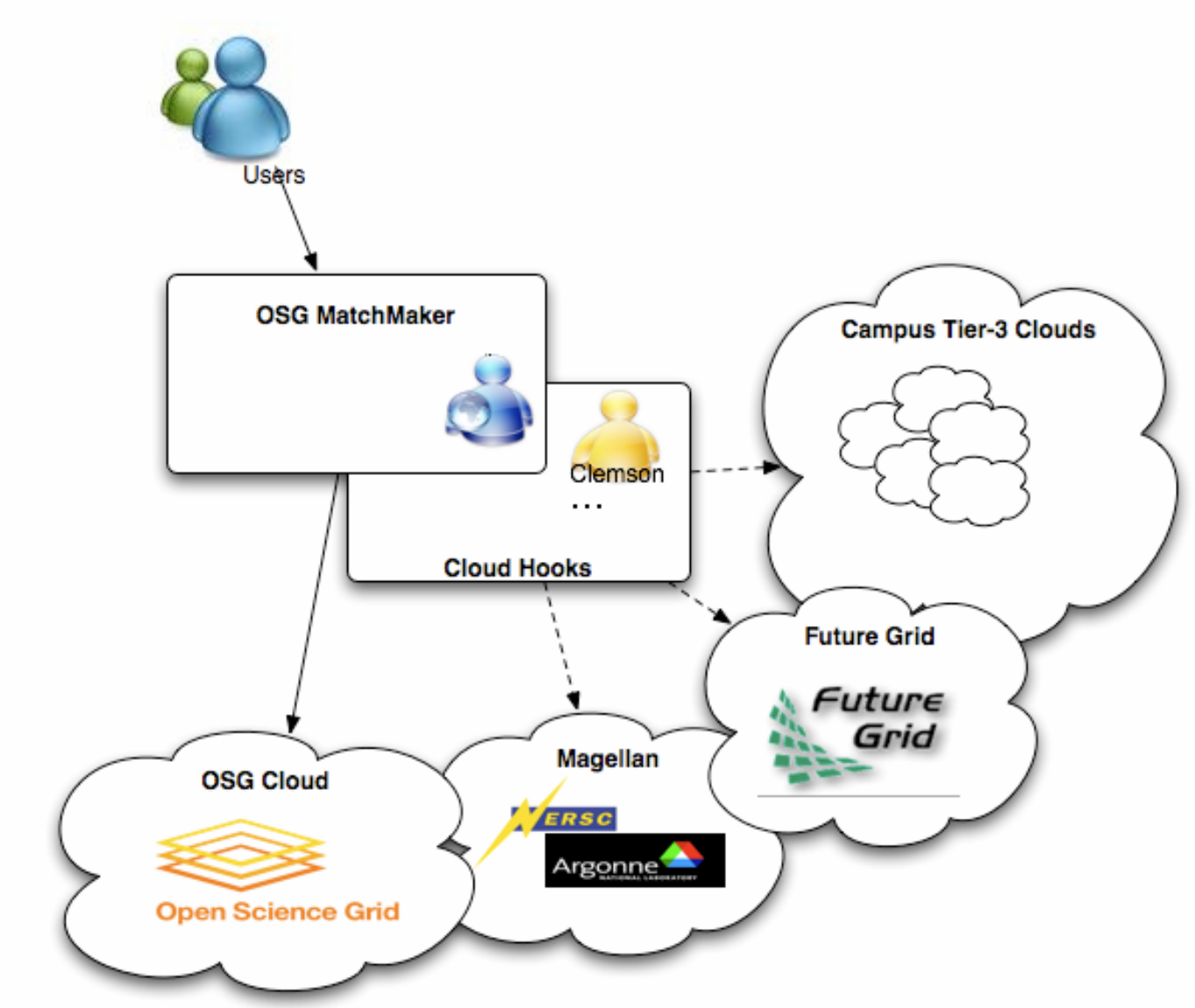
Thoughts & Conclusions?

- Any new possibilities with “Cloud” or VM?
  - Easy software access, lease and flexible licensing (concurrent licensing, not keyed to node) – worth investigating
  - Opening to bidding and economic model?
    - IaaS + VM + a standard interface → YES, can do
    - Best price / fastest delivery may be possible (more motivation for industry?)
- Models & interfaces are numerous
  - Any need for a unification? Grid of grids idea all over again?
    - StratusLab: Enhancing Grid with Cloud computing
    - DeltaCloud: support for all major Grid providers via plug-and-play
- Activities ahead
  - STAR intends to leverage Magellan resources (ANL and/or NERSC) to answer some of the “cluster” and interface questions
  - Other efforts
    - Problem/challenge that spans DOE and NSF (Joint OSG / TeraGrid): ExTENCI project will explore use of our VMs across OSG and TeraGrid sites
  - Further ahead: OSG / VM satellite effort?

StratusLab






Users
OSG MatchMaker
Clemson
...
Cloud Hooks
Campus Tier-3 Clouds
Future Grid
Future Grid
Magellan
ERSC
Argonne
OSG Cloud
Open Science Grid

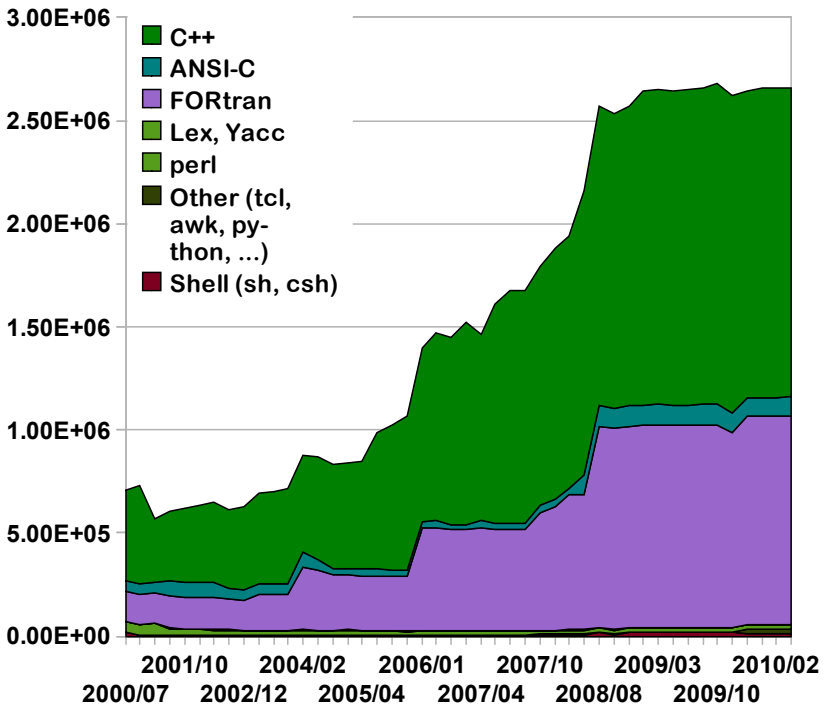
The end …

BROOKHAVEN NATIONAL LABORATORY

STAR

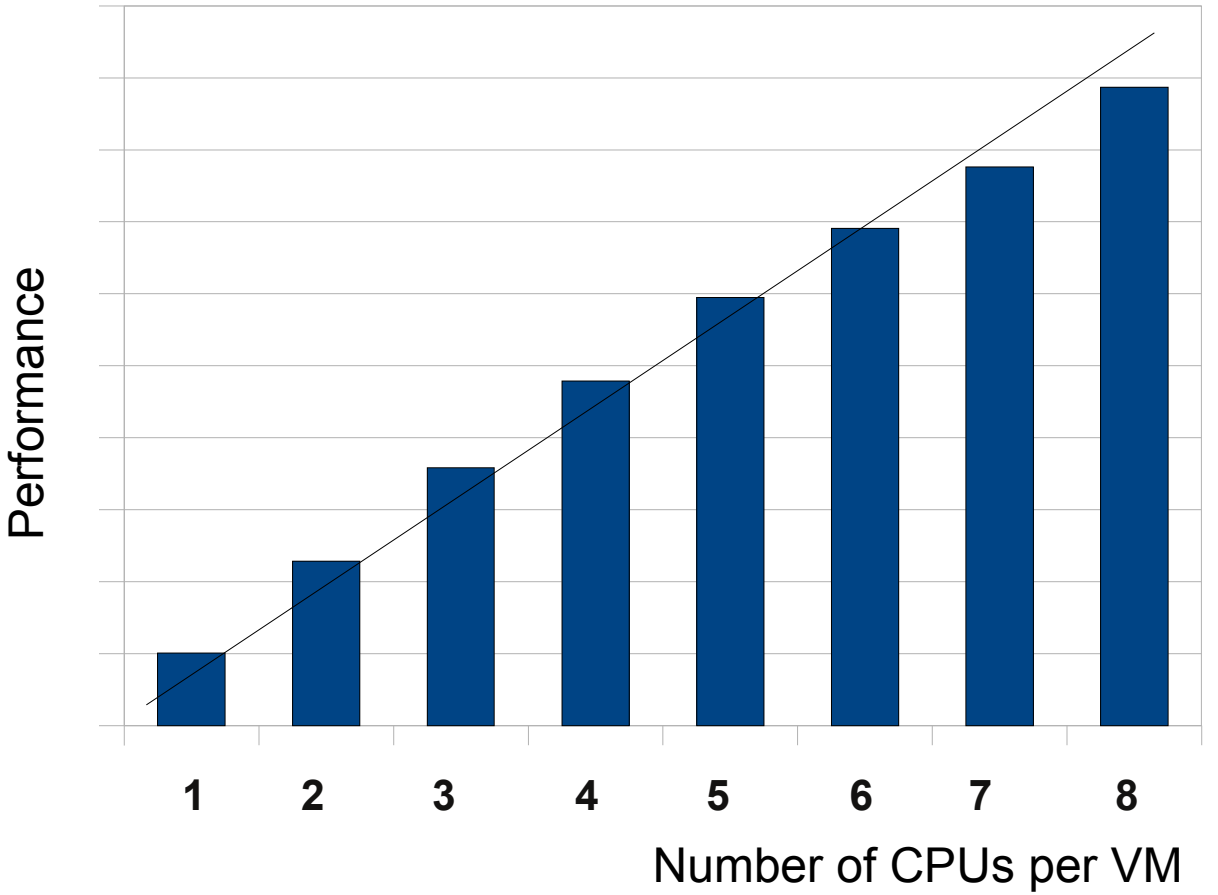
Backup slide

Evolution of #line of codes in STAR



Test case: 1 VM / hypervisor, raising the number of CPUs/hypervisors

Use case: VM for multithreaded apps - Xen



Ulrich Schwickerath (CERN), Sebastien Goasguen (Clemson/CERN)

Amazon EC2 (native)

- The people: STAR @ MIT – *Adam Kocoloski, Jan Balewski*, Mathew Walker
- Interface – standard Web access to EC2
  - General recipe – prepare VM
    - ~ 2 hours preparation to be done once
    - Contextualize (EC2 specifics)
    - Ship it to EC2 (slow? 20 mnts, also a one time job)
    - Login & check the VM exists, select STAR image, select machines, select SSH keys, firewall, …
  - Press the "launch" button … do your physics
  - Pay
- Our test
  - A 100 jobs, week long simulation cost ~ $1,510
  - A year long CPU @ 100 jobs saturation ~ 79k$
  - EC2+Nimbus
    - 300+ nodes for 10 days in 2008 (non-optimized) ~ $5,600

EC2 Prices on February 11, 2010

Standard On-Demand Instances	Linux/UNIX Usage
Small (Default)	$0.085 per hour
Large	$0.34 per hour
Extra Large	$0.68 per hour
High-Memory On-Demand Instances	**Linux/UNIX Usage**
Double Extra Large	$1.20 per hour
Quadruple Extra Large	$2.40 per hour
High-CPU On-Demand Instances	**Linux/UNIX Usage**
Medium	$0.17 per hour
Extra Large	$0.68 per hour

STAR users can use a MasterCard, VISA, Amex and run simulations on EC2 today!

… that is what they have done @ MIT
Usage context: last minute resource boost

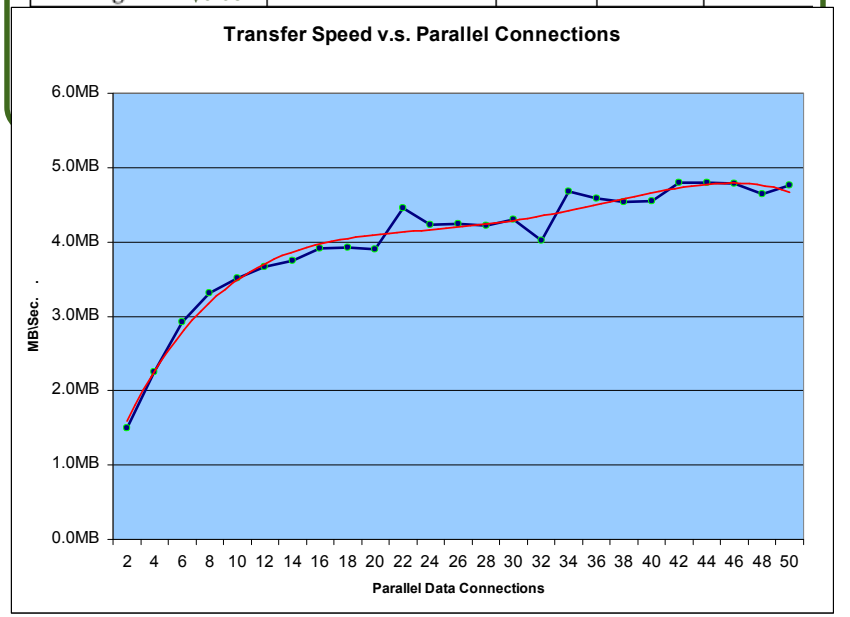
Performance?

- Instance & performance scaling tricky
  - CPU
    - 1 starsim would use 40% of the CPU on a small instance (effective price 0.21$/hour)
    - 1 starsim would take 99% of the CPU on a medium instance (effective price 0.17$/hour)
    - 2 starsim / medium instance gives 2x 95% of CPU (0.17 $/ hour) → $0.09/hour of used CPU
  - IO even more mysterious
    - But 5 MB/sec per VM is enough for STAR
- Our experience
  - Instances survived the run within O2 scale, efficiency > 99%
  - Simplistic interface – Web interface
  - IO – For simulation, enough
    - For real data transfer / 20% of our data production in 2011 requires 1.5 Gbits line for real time transfer

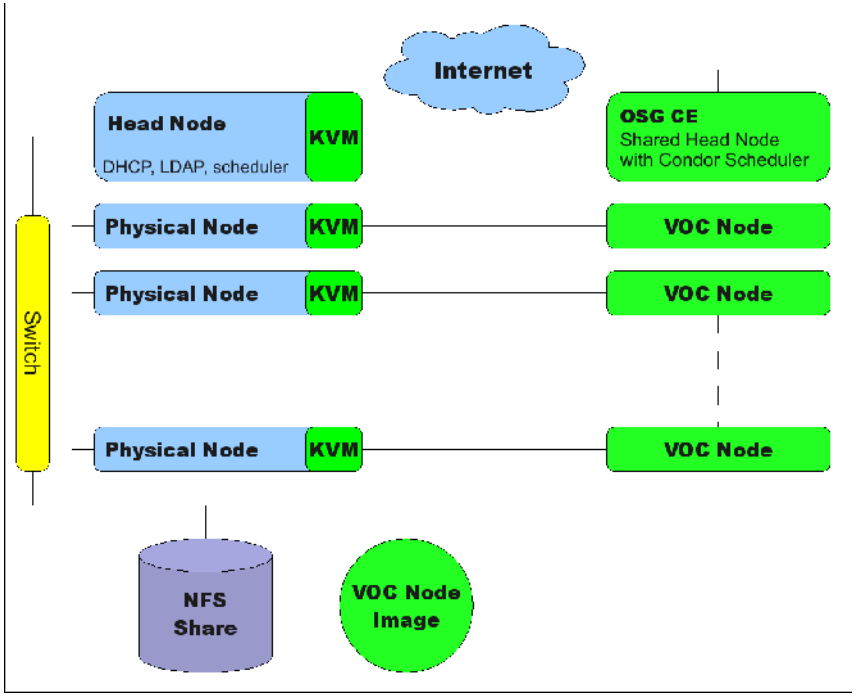
CPU speed test made by Adam Kocoloski

The CPU types are:
1 = Opteron 2218 HE @ 2.66GHz
2 = Opteron 270 @ 2.00GHz
3 = Xeon E5430 @ 2.66GHz
4 = Xeon E5345 @ 2.33GHz

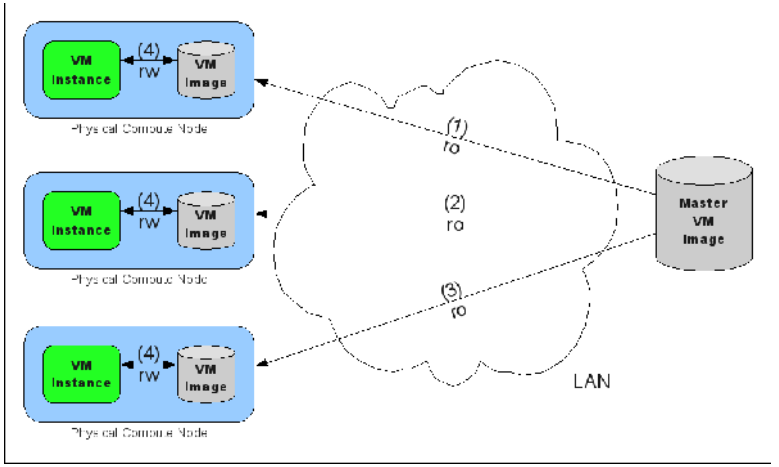
type	$/wall hour	zone	cpu	wall	cpu type
m1.small	$0.085	any	28.5	65	1
m1.large	$0.34	us-east-1a	35.5	36	2
m1.large		us-east-1b	15.3	20	3
m1.xlarge	$0.68	us-east-1a	28.3	49	1



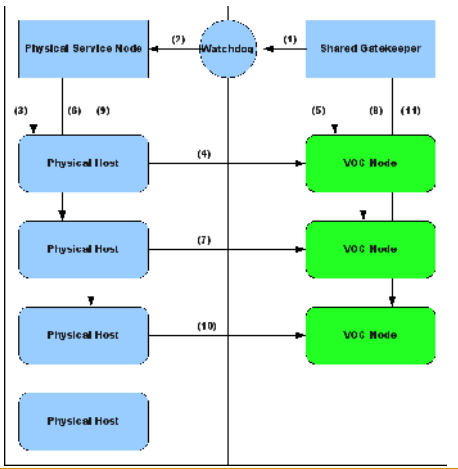
Clemson Virtual Organization Model



- KVM available on all physical nodes
- OSG CE VM running on Cluster head node
- VO-specific image available on NFS
- Physical nodes mount NFS location
- VMs are started directly from image on NFS

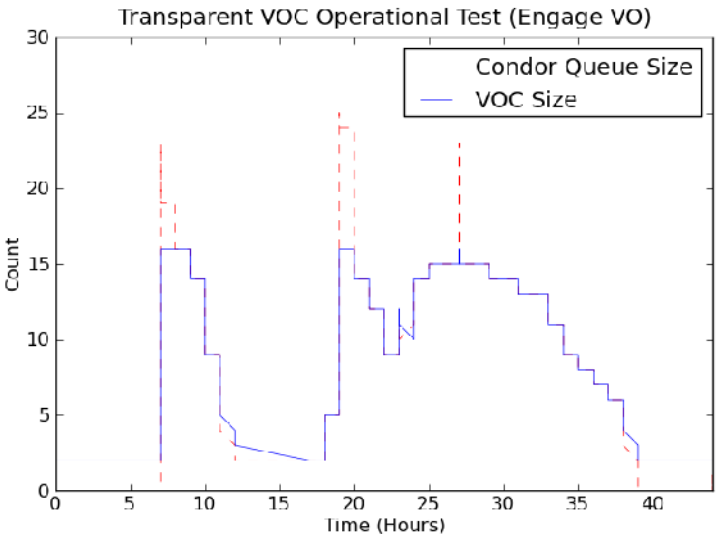


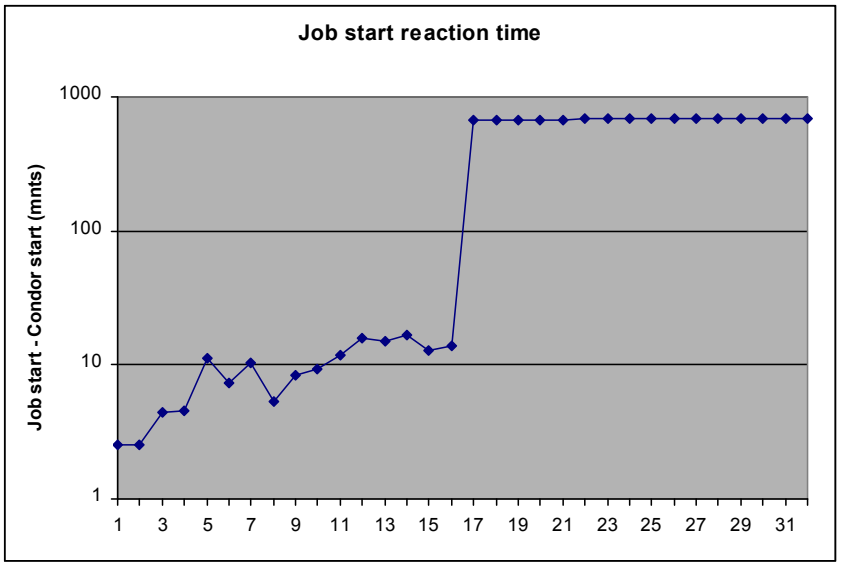
KVM -snapshot option allows 1-to-N relationship between image and instances



Watchdog process dynamically sizes virtual cluster

Behavior





Good tracking of queue demand and slots opening. Good tracking of queue demand decrease and slot closing.
Note: VM are NOT necessarily shutdown between jobs

Activation time average 7 minutes
Job length ~ 11 hours
Overhead for starting 1%