Grid Engine & Amazon EC2

2009 Sun HPC Workshop

Chris Dagdigian

BioTeam Inc.

BIOTEAM
Enabling Science

Utility Computing For Cynics

- **Doubt and cynicism are totally appropriate**
- **Personally burned by 90's era "OMG!! GRID Computing" Hype**

In 2009 trying hard never to use the word "*cloud*"
in any serious technical conversation. Vocabulary matters.

BIOTEAM
Enabling Science

Utility Computing For Cynics

- **Understand My Bias:**
  - Speaking of “utility computing” as it resonates with *infrastructure* people
  - My building blocks are systems, pipelines & workflows, not APIs, software stacks or services
  - Goal: Replicate, duplicate, improve or relocate complex systems

BIOTEAM
Enabling Science

BioTeam Inc.

- **Independent Consulting Shop:**
  Vendor/technology agnostic

  Distributed entity - no physical office
- **Staffed by:**
  - Scientists forced to learn
    High Performance IT to conduct research
  - Many years of industry & academic experience
- **Our specialty:**
  Bridging the gap between Science & IT

BIOTEAM
Enabling Science

Our Use Of AWS Today

- **Running Our Business**
  Development, Prototyping & CDN
  - Effective resource for tech-centric firms
- **Grid Training Practice**
  - Self-organizing Grid Engine clusters in EC2
  - Students get root on their own cluster
- **Proof Of Concept Projects**
  - For UnivaUD - UniCluster on EC2 experiment
  - For Sun - SDM ‘spare pool’ servers from EC2
- **Directed Efforts on AWS**
  ISV software ports, Pharma clients, etc.

BIOTEAM
Enabling Science

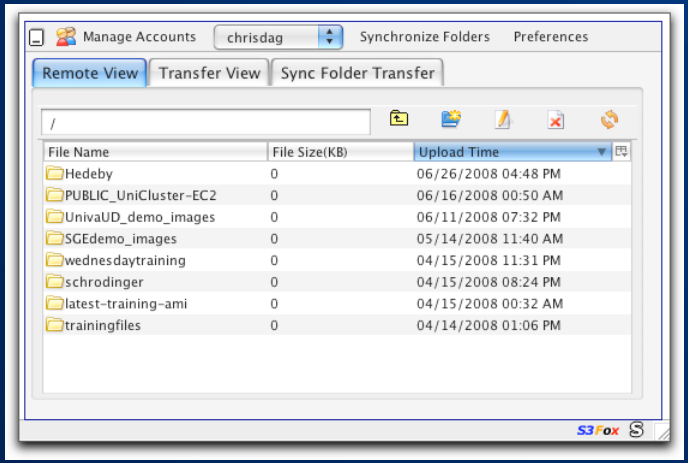
Favorite AWS Project This Year

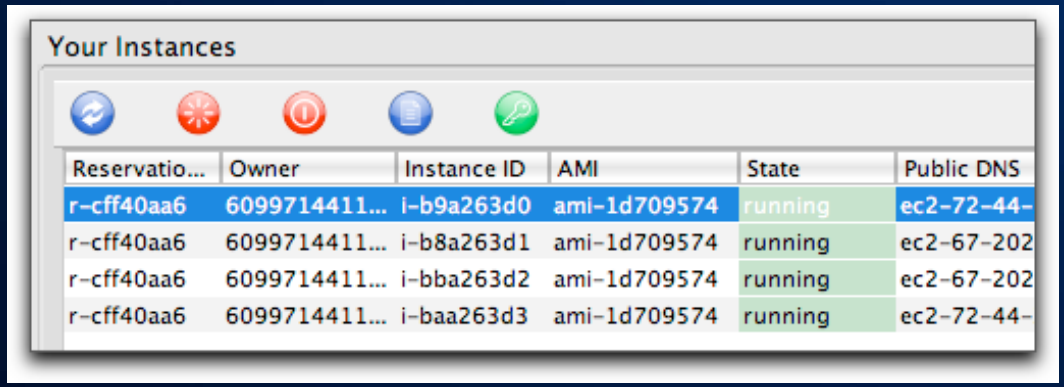
- **Special project:**
  20 TB bulk data export project from Amazon S3
  - Our client generated the data on EC2
  - Data had to be delivered to secretive recipient on physical 5TB NAS devices
  - Amazon S3 Export had not launched yet
  - We eventually moved 2TB/day from S3 (!)

BIOTEAM
Enabling Science

Lets Be Honest

- **Not rocket science**
- **Fast becoming accepted and mainstream**
- **Easy to understand the pros & cons**





BIOTEAM
Enabling Science

While we are being honest …

- **Amazon *is* the cloud**
  - Lead time measured in year(s)
- **Everyone else playing catch-up**
  - Replicating things AWS has done for years
  - … while AWS pushes out new product and continues to define the market
- **These are not viable EC2 alternatives:**
  - Flashy vendor press announcements
  - Your university thesis project

BIOTEAM
Enabling Science

The real EC2 action …

- **AWS is just infrastructure**
  - The cool stuff comes from people doing clever things with the base foundation technologies
- **Companies I watch:**
  - UnivaUD
  - Rightscale
  - CycleComputing

BIOTEAM
Enabling Science

HPC & AWS: Whole new world

- **For cluster people some radical changes**
  Years spent tuning systems for shared access
  - Utility model offers *dedicated* resources
  - EC2 not architected for *our* needs
  - Best practices & reference architectures will change
- **Current State: Transition Period**
  - Still hard to achieve seamless integration with local clusters & remote utility clouds
  - Most people are moving entire workflows into the cloud rather than linking grids
  - Data movement & network speeds are the issue

BIOTEAM
Enabling Science

The Obvious Question

Why use Grid Engine at all?

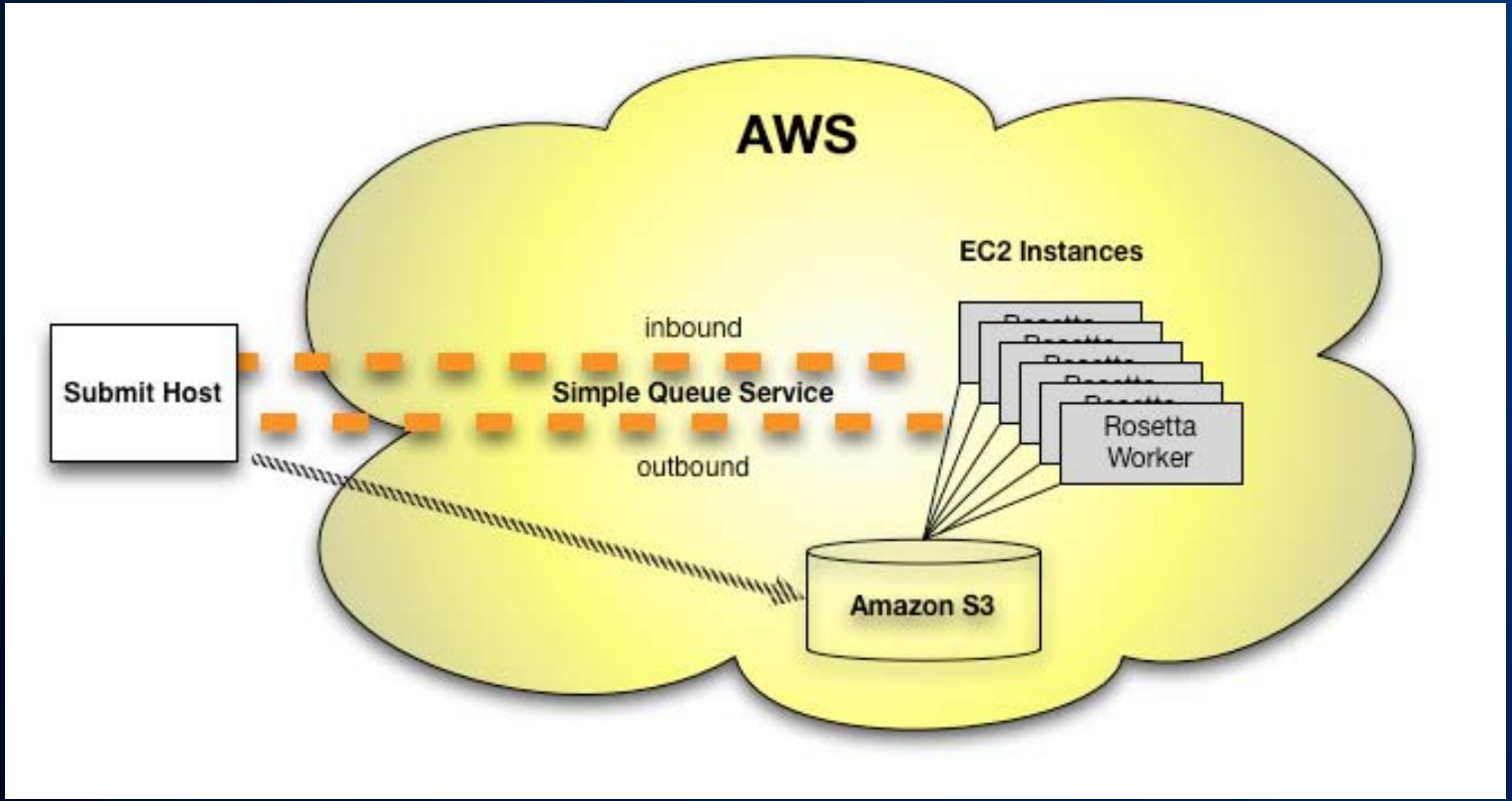
BIOTEAM
Enabling Science

Why use Grid Engine at all?

- Methods and best practices for workflow and workloads already exist within AWS
  - Why do something different?
- Answer:
  - For the non-trivial use cases
  - When re-architecting does not make sense
  - For hybrid clusters
  - For ‘cloud bursting’ (*ugh..)*

BIOTEAM
Enabling Science

Protein Engineering the Amazon way



BIOTEAM
Enabling Science

Protein Engineering the Amazon way

- Inbound/Outbound SQS queues
- Job specification in JSON format
- Data for each work unit in S3 buckets
- Custom EC2 AMI
- Workers pull from S3, push back when finished
- Job provenance/metadata stored in SimpleDB
- Independent work units allow dynamic allocation of Worker instances

BIOTEAM
Enabling Science

So you still want SGE in “the cloud?

My $.02

BIOTEAM
Enabling Science

Deployment Options

- 100% self-contained cluster-in-the-cloud
- Hybrid system (“cloud bursting”)
- Virtual Private Cloud

BIOTEAM
Enabling Science

Self Contained SGE on AWS

- In a nutshell:
  - Amazon instance AMI's capable of booting and self-organizing into fully operational Grid Engine systems
- My use case for this …

BIOTEAM
Enabling Science

Uses SGE; produces 1TB/day

Ilumina "next-generation" DNA Sequencing instrument

BIOTEAM
Enabling Science

If you don't solve the cold storage problem …

180+ TB stored on lab bench

The life science "data tsunami" is no joke.

BIOTEAM
Enabling Science

Use Case

- Lab instrument produces 1TB/day
- SGE integral in raw data analysis
- 1TB distilled down to “few GB” in final form
- Storing the raw data is problematic
  - Many people decide to toss it
  - Cheaper to repeat the experiment if needed
  - Storage is cheap, but …
  - Big, long-term “safe” storage is not cheap

BIOTEAM
Enabling Science

The idea …

- Initial analysis done onsite with local cluster
- Smaller “derived” data kept local
- Raw data copied to 1TB SATA disks
- FedExed to Amazon for S3 Ingest
- Key point: 1-way data movement
- Amazon provides geo-redundant bulk store
- If we ever need to reanalyze the data …
- Bring data back inhouse? Hell no!
  - Fire up our self-organizing SGE cluster in the cloud and (re)manipulate the data in-situ

BIOTEAM
Enabling Science

Hybrid methods

aka “cloud bursting”

BIOTEAM
Enabling Science

Hybrid SGE/Cloud Systems

- Central idea
  - Persistent local qmaster and other local infrastructure elements
  - Probably local compute resources as well
  - Fire up SGE compute nodes within EC2 as needed
  - Bind them to your local cluster
  - Move “appropriate” work into EC2 nodes

BIOTEAM
Enabling Science

Hybrid SGE/Cloud Systems

- Univa UniCluster does this today
- Univa UniCloud does this today

- Sun HPC stack seems able to do this using SDM and the EC2 adapter
  - For Linux & OpenSolaris

- Rolling your own
  - Not that hard to do
  - Grid Engine can do this with SDM & EC2 adaptor

BIOTEAM
Enabling Science

Virtual Private Clouds

aka “grab your BS shovel …”

BIOTEAM
Enabling Science

Virtual Private Clouds

- Cloud infrastructure in your own datacenter(s)
  - May or may not ‘bridge’ out to other utility computing service providers
- Claim: “*All of the benefits of cloud computing right in your datacenter! Free unicorns & kittens too!*”
- *Everybody* is trying to play in this space
  - Just like the 90’s “Grid Computing” hype
  - Similar potential for utter failure to deliver on promises made

BIOTEAM
Enabling Science

Virtual Private Clouds

- My only recommendation is to perform your due diligence carefully
- If you drill down into some of these offerings you may find:
  - Wholesale replacement of legacy IT gear required
  - Showcase demo systems that only really do one thing
  - Reference sites that receive significant financial, implementation and engineering assistance that “regular” customers will not receive

BIOTEAM
Enabling Science

Virtual Private Clouds

- Calming down now …
- Fact is
  - Lots of data on where and when utility computing can make sense as a point solution for particular problems
  - Very little data on suitability as a full-on internal IT platform
- Univa's UniCloud EDA presentation yesterday was interesting and compelling
  - Use of Xen VM migration was pretty cool
  - Kudos for presenting benchmark data

BIOTEAM
Enabling Science

Amazon VPC Announcement

- New product in private beta: Amazon VPC
- Using “Virtual Private Cloud” term as well
- What Amazon VPC does
  - Direct VPN tunnel to AWS
  - You control subnets and IP addressing
  - Your own firewalls/security gear can be used

BIOTEAM
Enabling Science

Amazon VPC Announcement

- Why this matters:
  - EC2 addressing, NAT and internal vs. external hostnames are a massive pain if you have to link to your own systems or infrastructure
  - Most people already know this and already use software VPNs (Sun SDM, Univa, etc.)

BIOTEAM
Enabling Science

And that leads me to …

An attempt at some practical advice

BIOTEAM
Enabling Science

SGE & AWS: Pain Points

- EC2 (compute), S3 & EBS (storage) are not that hard to deal with. Really.
- The awkward bits usually involve networking:
  - Nodes do not know their public IP address
  - DNS and hostnames resolve only within AWS network
  - No guarantee your EC2 nodes will even be on the same subnet
    - Tough for MPI …
- This is why so many people use VPN technologies when building cloud-resident systems

BIOTEAM
Enabling Science

SGE autodeploy in EC2

- Single AMI server image (CentOS 5)
- On boot
  - Must be able to learn all hostnames associated with instance reservation
    - Perl Net::Amazon::EC2, etc.
  - Must be able to elect who is ‘master’ and who is ‘worker’
    - Query reservation hostnames; Sort alphabetically
    - 1st on list is ‘master’, all others are ‘workers’

BIOTEAM
Enabling Science

SGE autodeploy in EC2

- If “master”
  - Configure /etc/exports & start NFS
  - NFS for easy data sharing, not Grid Engine
  - Configure ‘dsh’ for passwordless SSH to nodes
- If “worker”
  - Query reservation, learn master hostname
  - Populate $SGE_ROOT/$SGE_CELL/act_qmaster
  - Create /etc/fstab entry for NFS mount of master
- Then
  - Wait …

BIOTEAM
Enabling Science

SGE autodeploy in EC2

- Why wait?
  - No guarantee that EC2 systems will be provisioned and booted in any particular order
  - Can't assume our "master" will be up before our "workers"
- So
  - All systems create necessary files and then wait
  - First root login to 'master' node kicks off a finishing script
    - NFS client & server start
    - Template-driven SGE auto-installation
- If that is not acceptable
  - Easy enough to solve with a script that runs on the elected master
  - When all hosts in reservation instance are 'alive', trigger final cluster assembly steps

BIOTEAM
Enabling Science

SGE with Sun SDM on EC2

- Specifically seen in my early SDM/SGE experiments:
  - Java SDM code really wants to bind to the public IP
  - SDM will break on EC2 when nodes don't know their publicly reachable IP
  - Solution:
    - Query reservation, learn public IP
    - Via 'ifconfig alias' or similar, add this to your EC2 hosts and let SDM bind here

BIOTEAM
Enabling Science

And finally

BIOTEAM
Enabling Science

Cloud Sobriety

McKinsey presentation "Clearing the Air on Cloud Computing" is a must-read

- Tries to deflate the hype a bit
- James Hamilton has a nice reaction:
  - http://perspectives.mvdirona.com/

Both conclude:

- IT staff needs to understand "the cloud"
- Critical to quantify your own internal costs
- Do your own due diligence

BIOTEAM
Enabling Science

End;

- Thanks!
- Comments/feedback:
  - chris@bioteam.net

BIOTEAM
Enabling Science