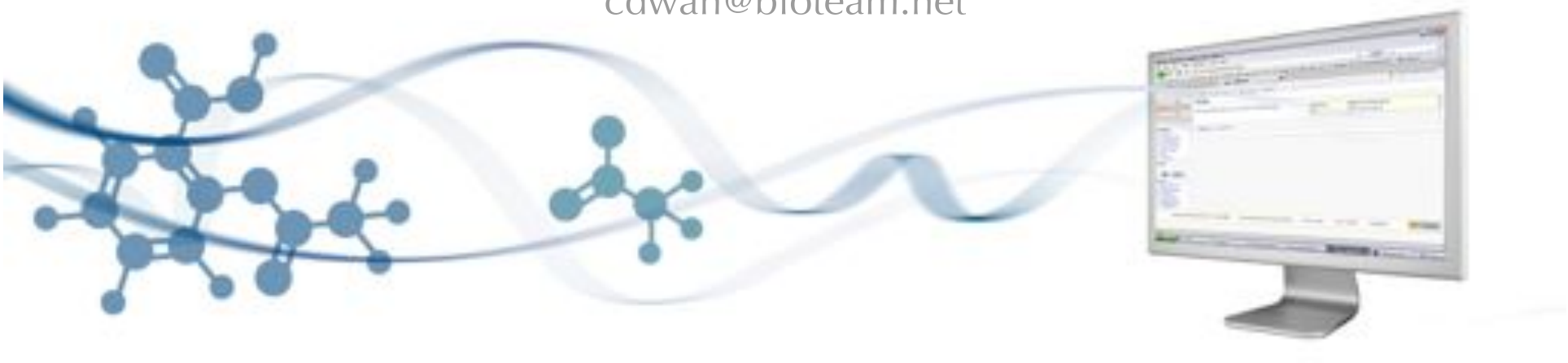
BIOTEAM
Enabling Science

BIOTEAM
Enabling Science

# Trends and Observations, October 2009

Christopher Dwan
cdwan@bioteam.net



# Bioteam

## Independent consulting group

- Technology / Vendor neutral
- Scientists forced to learn HPC to get our jobs done

## Our Specialty

- "Bridging the gap" between science and high performance computing

# Basis of my opinions

**Ongoing engagements**

- NASA Langley Science Directorate
- Navy Medical Research Lab, Bio-defense directorate
- Centers for Disease Control and Prevention
- MIT Dept. of Environmental Engineering

**Short term engagements**

- Joslin Diabetes Research Center
- Hopkins Center for Inherited Disease Research
- Several large Pharmaceuticals
- Several small biotechs

**Community presence**

- Supercomputing
- Bio IT world
- CHI / other small meetings
- Open source community

# Major Themes

**The more things change**

- Massively multicore servers
- Graphics processors
- Amazon compute cloud
- Virtualization
- Noticeably better local expertise vs. 2004

**The more they stay the same**

- Facilities issues
- Data lifecycle management
- Workflow flexibility vs. Enterprise deployment

# Massively Multicore Linux Systems

**Commodity servers:**

- 256 core cluster with 4TB RAM and 128TB disk fits in 16 rack units.
  - 4 chips x 4 cores = 16 cores per server
  - 32 DIMM slots x 8GB DIMMs = 256GB RAM per server
  - 4 x 2TB = 8 TB internal disk

# Small Clusters of "Fat" nodes

**Cost curve supports packing cores into a small number of nodes**

- Incremental cost of chassis is higher than that of nodes.

**Each node is a higher percentage of the cluster capacity**

- Worthwhile to invest in redundant power supplies, etc. on the compute nodes
- Capricious reboots are more costly.

**Non uniform memory / network topology**

- Scheduling multi threaded tasks onto a single node becomes important
- Questions of low latency networking can be pushed onto the motherboard for jobs that don't parallelize well beyond 16 or 32 cores.
- Gb/sec networking still appears adequate for most teams

**Heat / Density / Power**

# Graphics Processors

**Think of a GPU as a 100,000 core chip with only a little bit of RAM**

- Very close to having a "—target" flag for NVDIA from gcc
- Ordinary linux server with extra capability invoked in the binary
- Fits nicely in a heterogeneous compute cluster
- Also fits quietly in a lab instrument

**Field Programmable Gate Arrays (FPGAs) were supposed to fill this role 5 years ago.**

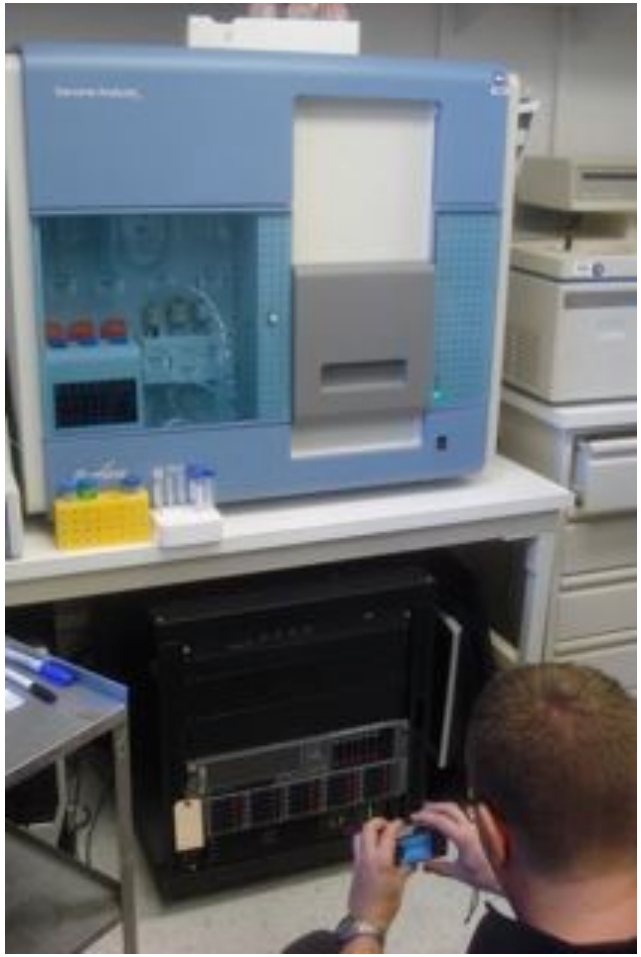
- To my knowledge, they still do not.

**Hybrid clusters including GPU servers are becoming common**

- All GPU, all the time is still somewhat academic (Harvard)
- Three customers this quarter: "stand it up, give it a dedicated queue, we'll figure it out later"

# GPU processing is a commodity

# Clusters are invisible



**Instrument vendors routinely ship outboard compute clusters.**

- Horrifying from a systems perspective

**Customers buying extra clusters**

- Rather than attempting integration with existing

**40TB direct attached storage is also a commodity.**

- Broad Inst. uses this to buy breathing room

# Moore's Law Has Changed Character

**Code used to get "better" simply by riding the annual increase in clock speed brought on by shrinking electronics.**

- Most code has gotten much *less* efficient over time as programmers knew that increased clock speeds and memory sizes would cover their slop.

**Very few examples of code that parallelizes out to 1,000s of CPUs.**

- That code is already moving to GPUs

**Important to revisit architectural assumptions regularly**

- Moore's curse: What I used to be proud of doing is now either trivial or actually a bad idea.

# Amazon Compute Cloud

**Amazon's EC2 and S3 are "The Grid" as promised in the 90's.**

- To my knowledge, all others are academic or vaporware
- This goes double for "private clouds"
- Semantic quibble: "Cloud" != "grid" != "cluster"

Instrument vendors, pharmaceutical companies, and government agencies are all outsourcing *intermittent*, *bursty* computation to the cloud.

**Hurdles Cleared:**

- Data motion (Fedex-net is up and running as of 2009)
- Security (B2B VPN on a case by case basis)
- Scaling / build out teething issues

**Remaining Challenges:**

- Social acceptance: Accurate blame for leaks and failures
- Utility / industrial chargeback model (Amex doesn't cut it)
- Quality of service. (Two nines is not sufficient for all use cases)

# Cloud Scaling

**Storage**

- Bioteam was contracted to move 40TB of imaging data out of the cloud in early 2009
- Observed high-water-mark of 200TB from a single organization *into* the cloud in late 2009
- Jury is still out on the cloud as long term destination for "permanent" storage
- Still need enterprise level assurance for long term data retention and protection

**Clusters**

- Bioteam has moved several tools into the cloud for a variety of clients.
- "Burstable" clusters of 100s of nodes are reliable and easy
- 1000s of nodes are more challenging, but possible

**Bandwidth remains a challenge**

- Fedex remains a not-unreasonable solution

# Virtualization (as distinct from 'cloud')

**Virtualized software delivery is real**

- Remote hosting (software as a service)
- VM image (software with no installer)

**Server lifecycle management is real**

- Old servers don't die, they become virtual
- Decouple hardware purchases from specific software tools.

**Effect on compute clusters**

- Have seen virtualized interactive nodes, schedulers, etc.
- Have not seen many completely virtual compute nodes
  - Caveat: May be real, I just haven't seen them in operation yet.

# Local Expertise and Open Source Tools

**Human resources for scientific computing in biology**

- An order of magnitude better in 2009 than they were in 2004.
- Caveat:  I may just be catching up
- Monster.com for "bioinformatics" yields lots of hits for real, talented people with the right experience.
- Undergraduate biology curricula are starting to include computer programming


**Open source tools**

- In many cases better supported and technically superior to their commercial counterparts

# Open Source

**Even the automatic gambling machines in bars run Linux**

# The more things remain the same

# Facilities Issues

**Modern servers are power hogs and highly efficient heaters**

- **2004:** 1A per rack unit was a reasonable estimate for steady state power draw of a loaded system
- **2009:** Vendors list 3 – 4A per rack unit, with "max draw" of up to 8A
- "Workgroup cluster" that can be plugged into a 20A wall outlet is down to 4 servers!
- Your data storage vendor should be able to hit 0.5PB / 42U rack for scientific data storage
- I personally saw two major facilities disasters just in October 2009

# Data Lifecycle Management

**Terabytes are still heavy.**

- 2009: 40TB data extraction from the cloud
- Rate limiting step: Puny memory for the RAID processor in a commodity NAS box. We could manage 1Gb/sec from the internet, and 200MB/sec to local disk, but only 8MB/sec to the NAS.

**Consider data's end destination when writing it the first time.**

- Long term storage: Think "bytes per base pair"
- How long is "forever?" (~5 years)
- Would replicating the experiment from stored samples satisfy your requirements? (-80 freezer as a backup solution)
- The curse of Moore's Law (simply migrate to the new rather than designing for the long time.

**The days of the "full backup" for the data vault are over.**

- No clear winner on an archival data solution
- MAID: Massive Array of Idle Disks

# Data Tsunami

**2nd generation of high throughput DNA sequencers will produce substantially lower data volumes**

- Vendors appear to have learned their lesson about creating compute and data storage problems

**The data "bubble" has permanently deformed research pipelines.**

- Researchers are used to using petabyte scale scratch space
- Once space is given, you never get it back.
- Multi-domain analyses, all vs. all

**Some groups will always buy $250k of whatever is available**

# The bigger they are, the harder they fall

**Data storage devices are expected to last forever**

- Therefore, I engineer for *when* components *will* fail rather than playing the probabilities on *if* they *might* fail.
- Reliability engineering has much to teach us

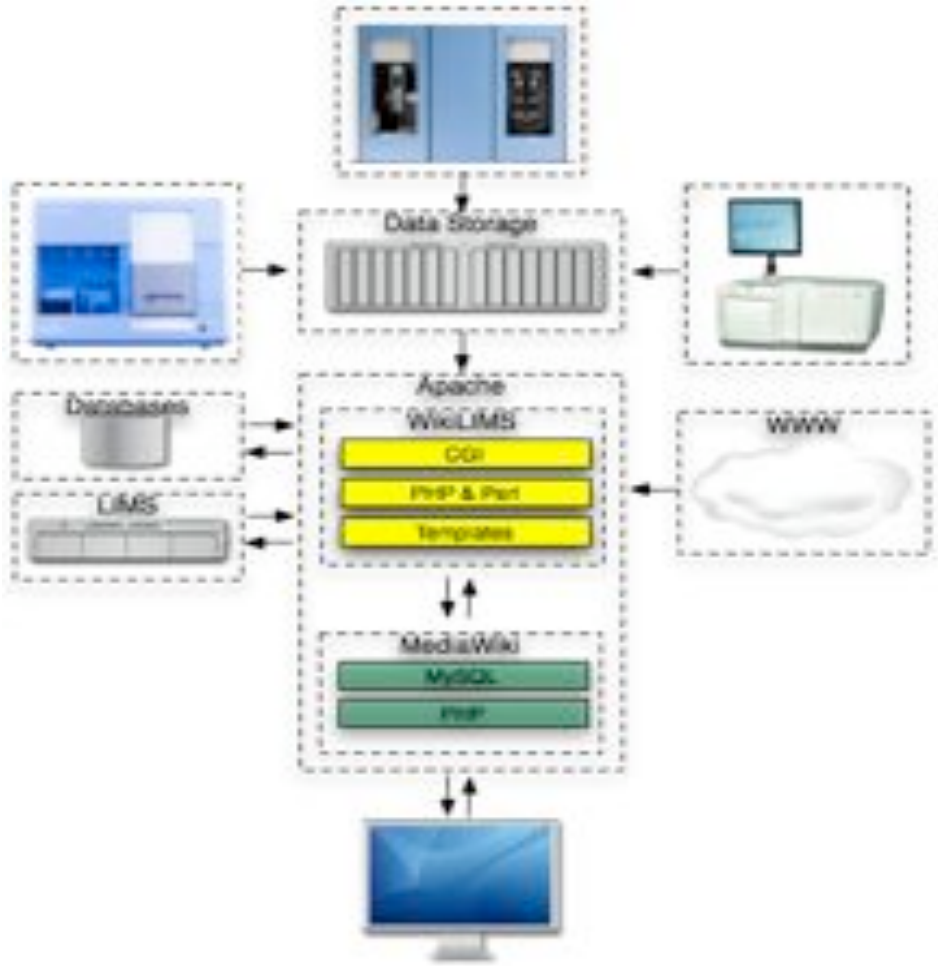
# Flexibility vs. Enterprise scaling

**Ongoing pressure between research and production**

- Local research groups can't predict usage patterns or tool needs
- Tools developed for a research group get deployed globally, far too soon.

**Wikis, particularly semantic technologies, are powerful and flexible beyond my expectations**

- Scaling far beyond my expectations
- Talk to Stan Gloss for more details



# When my phone rings

**Still several jobs per quarter:**

- Extract data from Excel into a wiki or a database
- Interview and advise on computing and storage needs for new instruments
- Install / parallelize existing code
- Process management / social pathology
- Purchasing support / deployment on large clusters / storage

**Fewer, I think:**

- Bright physicists re-inventing computational biology
- Computer vendors re-inventing computational biology
- Software startups accelerating BLAST

**New and vaguely scary:**

- Mature vendors from other industries (finance, video, etc) wanting “in”
- Highly secure (military) / federally regulated clients.

# Major Themes

**The more things change**

- Massively multicore servers
- Graphics processors
- Amazon Compute Cloud
- Virtualization
- Noticeably better local expertise vs. 2004

**The more they stay the same**

- Facilities issues
- Data Lifecycle Management
- Workflow flexibility vs. Enterprise deployment
- Basic Consulting