UJ Cluster workshop
Introduction

About me

- Ben Clifford
- University of Chicago Computation Institute staff
- Work on
  - Swift – programming language and environment for large scale distributed parallel applications
  - OSG Education, Outreach and Training
- Used to work on Globus Toolkit – building blocks from which to construct grids
- At UJ for a month to work on cluster and grid applications with anyone who wants to

Programme

- 1. Introduction
- 2. From PCs to Clusters to Grids
- 3. Submitting jobs to the grid with Condor
- 4. More advanced application techniques
- 5. More about the cluster
- 6. Guts of the grid
- 7. South African National Grid (Bruce Becker)
- 8. Porting your own applications

Module: PCs to Clusters to Grids

- Lots of people have experience building and running a scientific application on their PC
- Want to scale up to cluster and grid scale
- This module will give a practical example of an application starting on my laptop and growing to grid-scale.

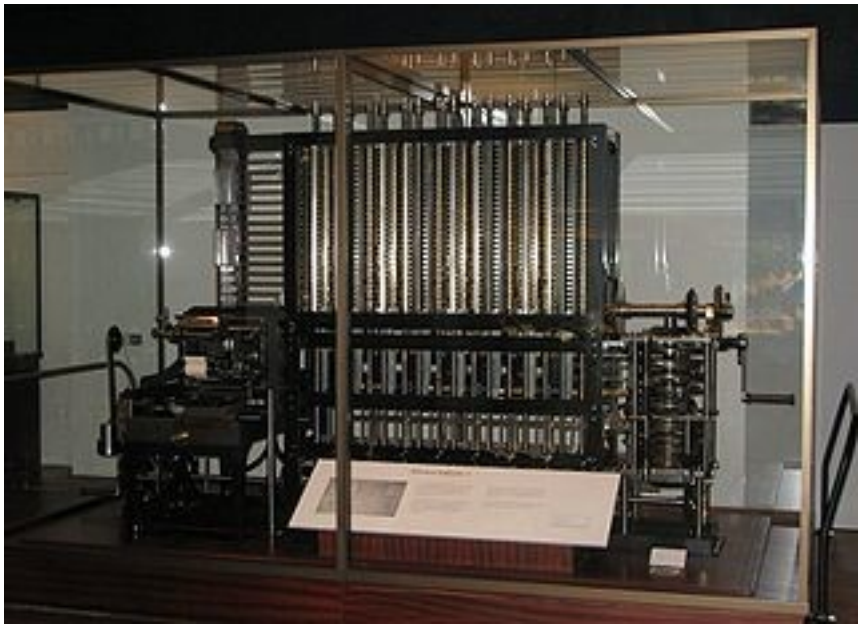
scientific computing

- doing science with computers
- (distinct from computer science – studying computers)
- lots of people doing this at the desktop scale
  - running programs on your PC
  - hopefully you have a feel for the benefits of doing that and also the limitations

Benefits of scientific computing

- Calculations that you couldn't (reasonably) do by hand
- Difference engine – designed (but not built) early 1800s to compute numerical tables for uses such as navigation and engineering

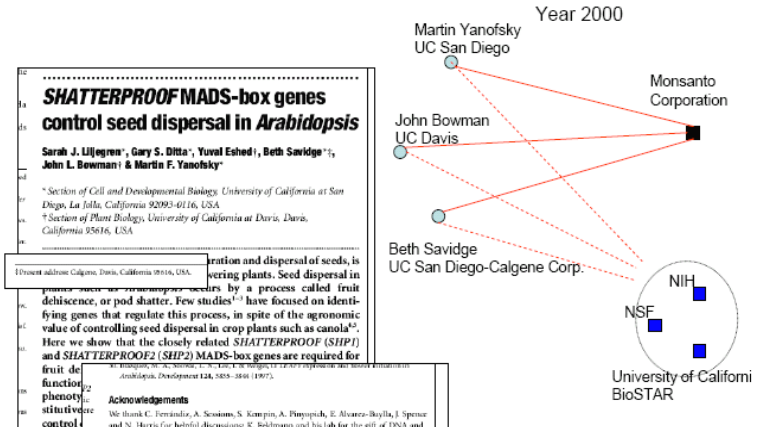
A contemporary of Babbage, Dionysius Lardner, wrote in 1834 that a random selection of forty volumes of numerical tables contained no fewer than 3,700 acknowledged errata and an unknown number of unacknowledged ones. - sciencemuseum.org.uk

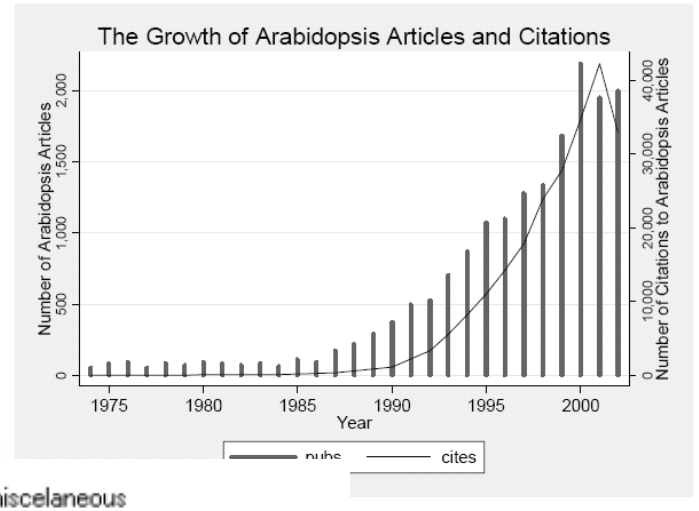


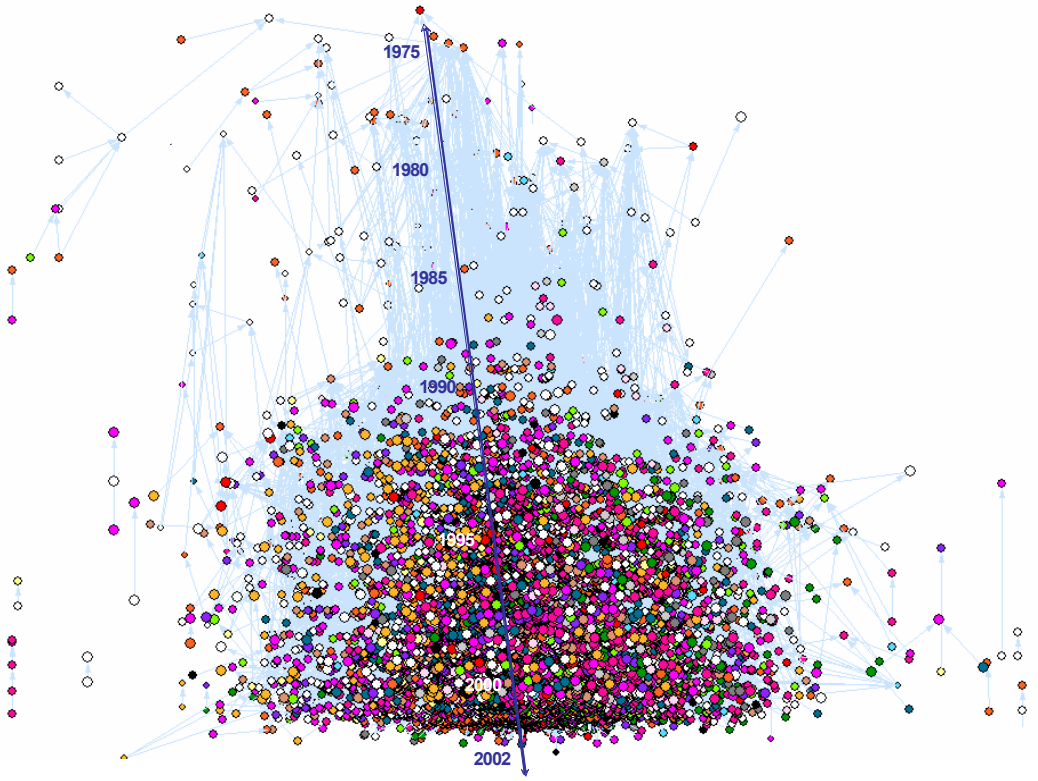
Limitations on the desktop

- You make a program
- It gives good results in a few minutes
- Hurrah!
- You start feeding in more and more data...

Scaling up Science: Citation Network Analysis in Sociology







Work of James Evans, University of Chicago, Department of Sociology

Scaling up the analysis

- Query and analysis of 25+ million citations
- Work started on desktop workstations
- Queries grew to month-long duration
- With data distributed across U of Chicago TeraPort cluster:
  - 50 (faster) CPUs gave 100 X speedup
  - Many more methods and hypotheses can be tested!
- Higher *throughput* and *capacity* enables *deeper analysis* and *broader community access*.

Time dimension: 30 minutes vs a month

- If your analysis takes 30 minutes:
  - about 10..20 runs in a working day
  - about 300 a month
  - like drinking a cup of coffee
- If your analysis takes 1 month:
  - about 1 a month
  - like paying rent
- Much more interactive

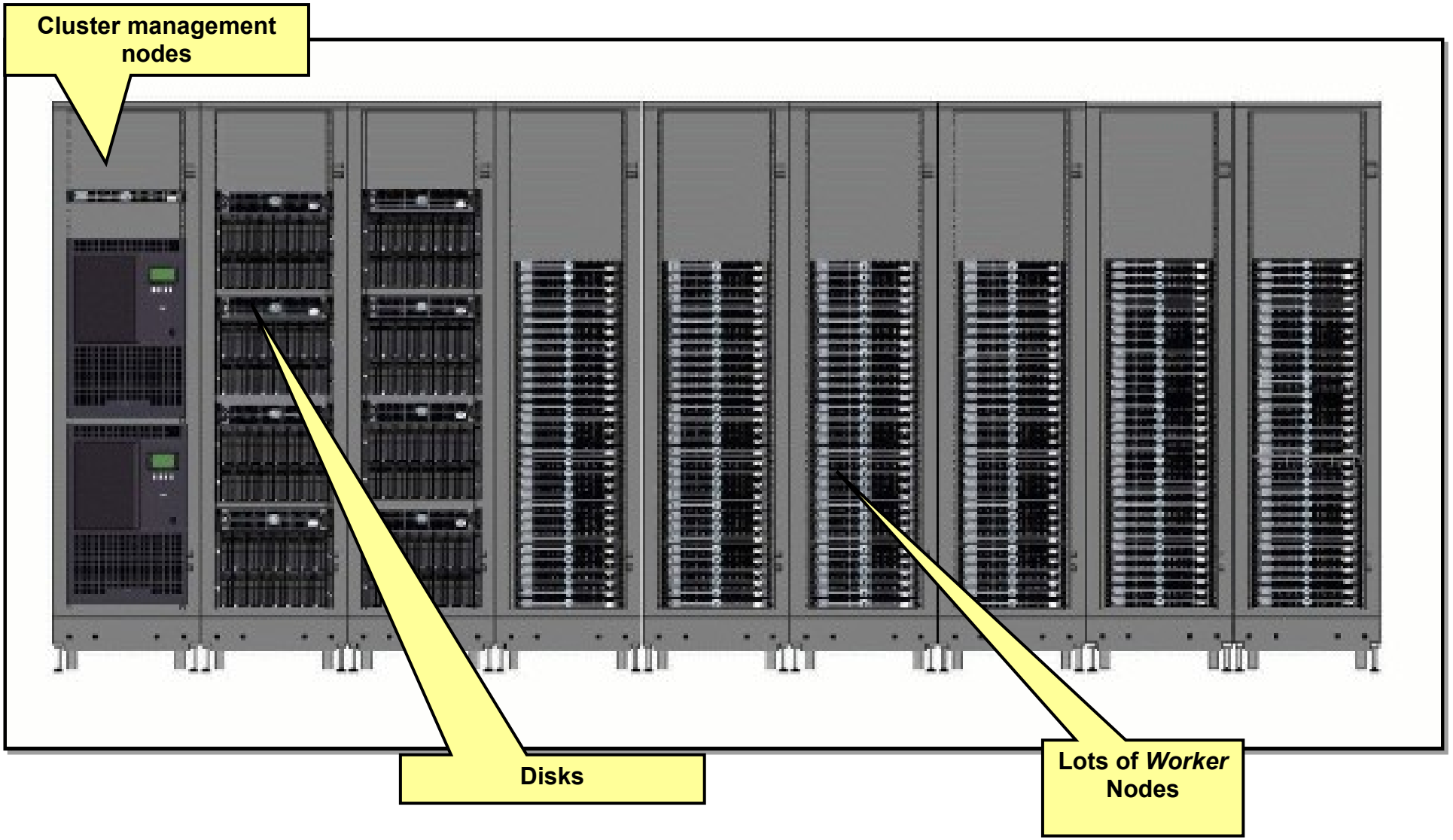
Size dimenson: 1 CPU vs 100 CPUs

- In the same time, you can do 50..100x more computation
  - more accuracy
  - cover a large parameter space
  - Shot of tequila vs 1.5l of tequila

Scale up from from your desktop to larger systems

- In this course going to talk about two large resources:
  - UJ cluster – ~100x more compute power than desktop
  - Grids – Open Science Grid (me), SA National Grid (Bruce) - ~30000x more compute power than desktop

A cluster

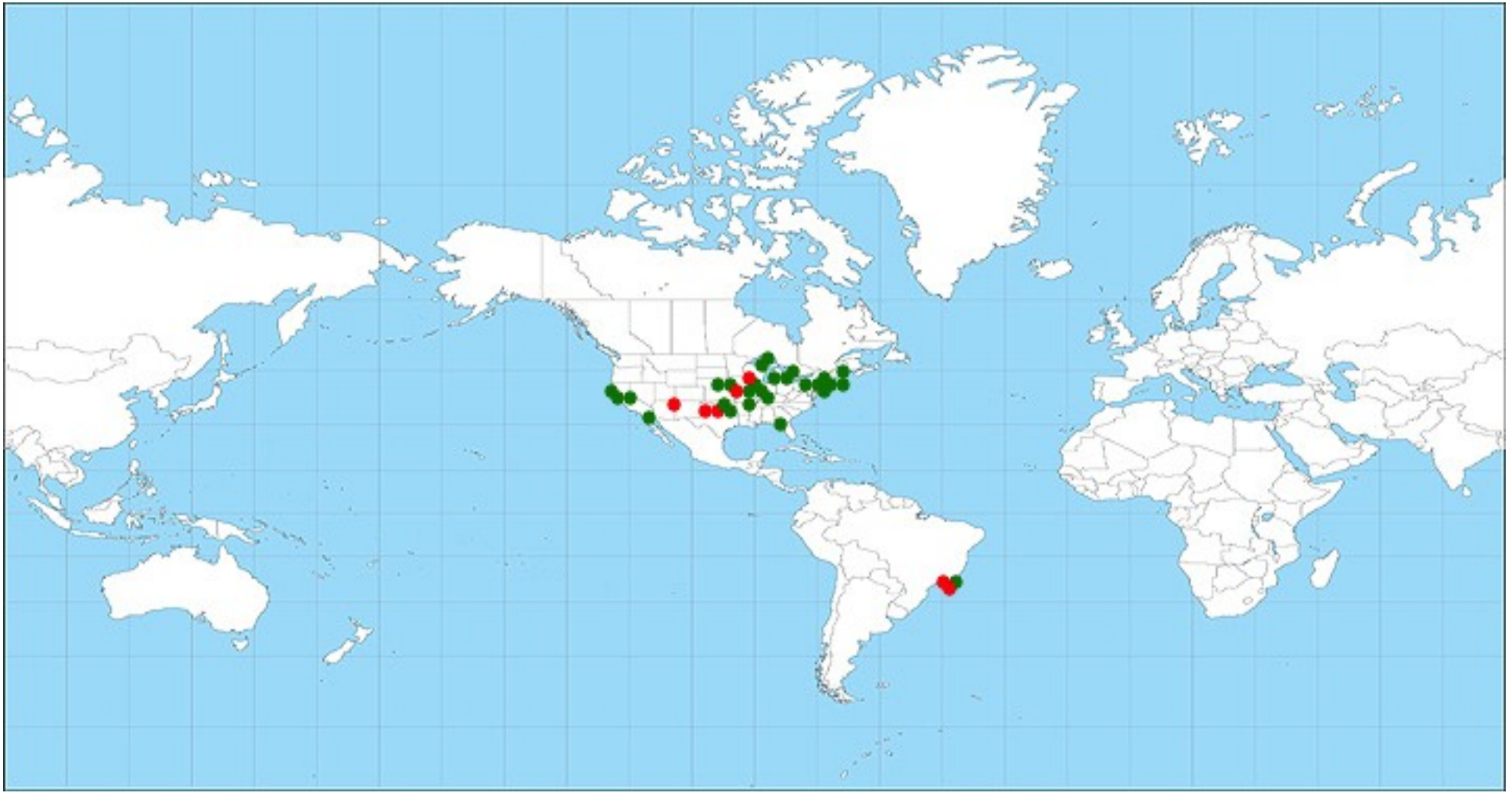


A cluster

- Worker nodes – these perform the actual computations for your application
- Other nodes
  - manage job queue, interface with users, provide shared services such as storage and monitoring

Open Science Grid

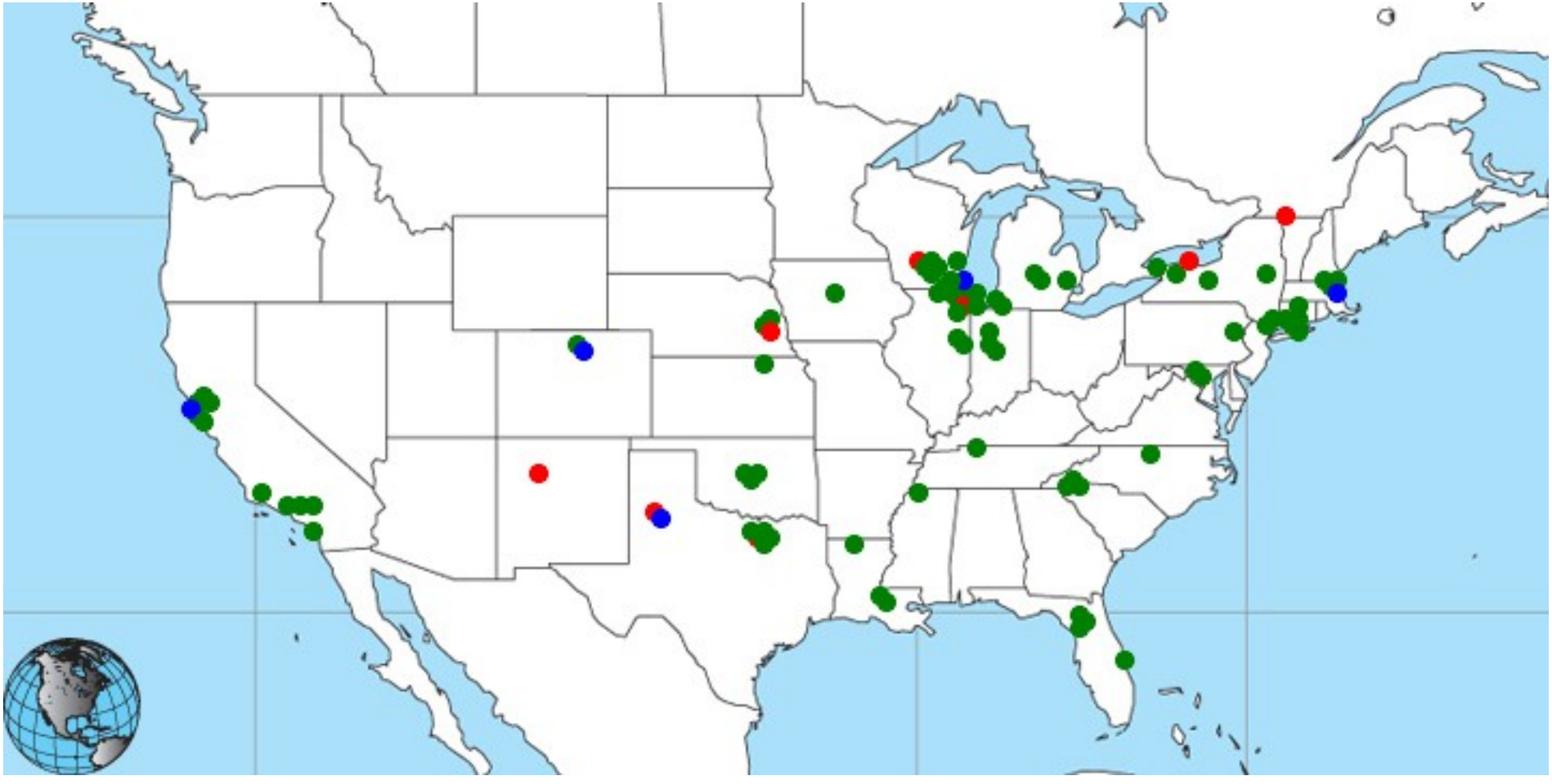
(from VORS - outdated)



- Dots are OSG sites (~= a cluster)

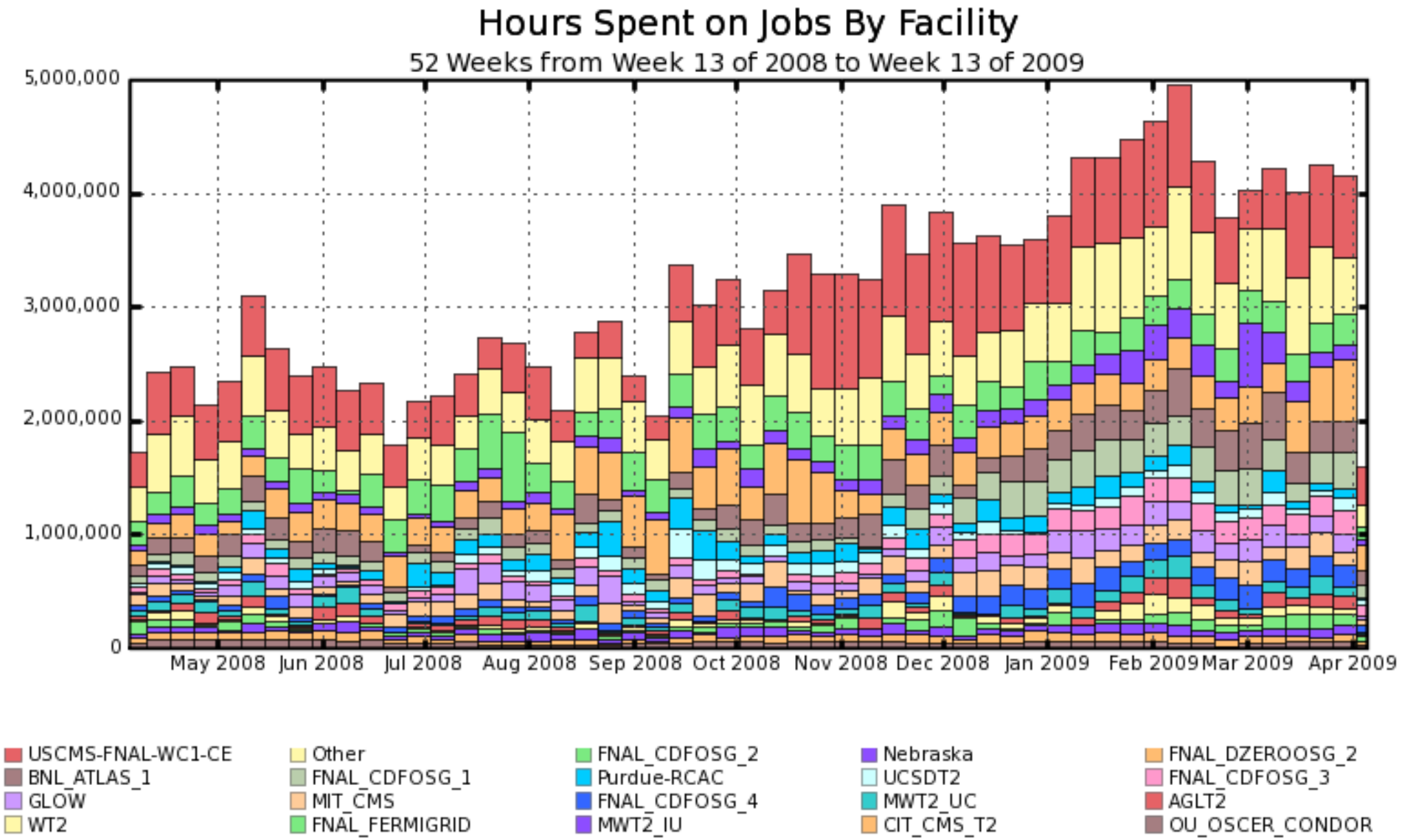
Open Science Grid

OSG US sites



]

Who is providing OSG compute power?



Initial Grid driver: High Energy Physics

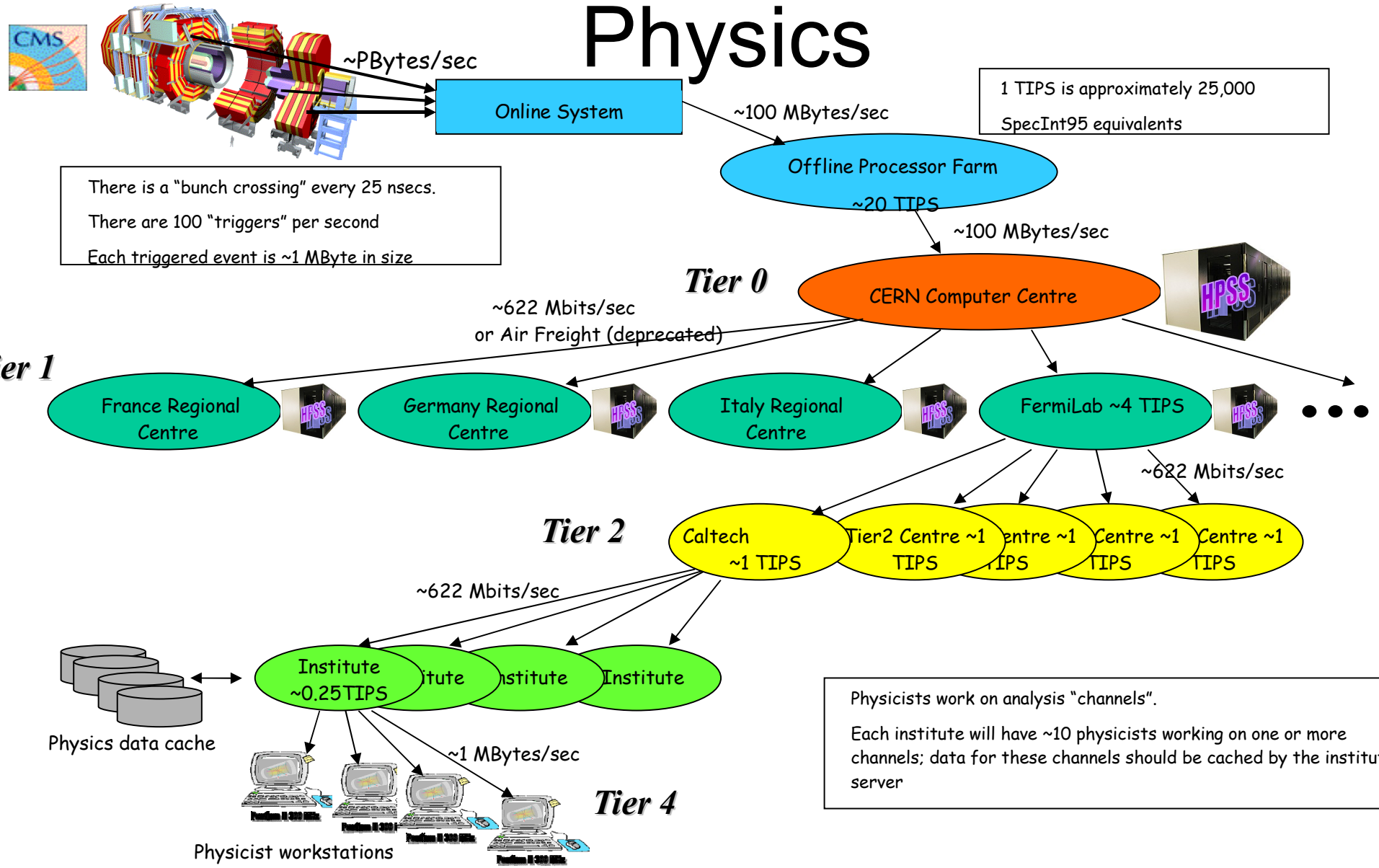


Image courtesy Harvey Newman, Caltech

High Energy Physics

- Lots of new data to process from live detector
- Lots of old data to store and reprocess
  - eg when you improve some algorithm to give better results, want to rerun things you've done before using this new algorithm
- This is science that couldn't happen without large amounts of computation and storage power.
- On Open Science Grid, HEP using equivalent of ~20000 PCs at once

How to structure your applications

- The “PCs to Clusters to Grids” module is mostly about the basic techniques needed to structure applications to take advantage of clusters and grids.
- How to make an application parallel
  - so that it can use multiple CPUs
- How to make an application distributed
  - so that it can use multiple CPUs in multiple locations
- Hands-on running on the UJ cluster

Module: Submitting jobs to the grid with Condor

- This will deal with the practical aspects of running in a grid environment in more depth.
- Introduce software package called Condor
- Practical will run an application on the Open Science Grid

Condor-G

- Condor-G (G for Grid)
- A system for sending pieces of your application to run on other sites on the grid
- Uses lower layer protocols from software called Globus Toolkit (that I used to work on) to communicate between sites
- Queues jobs, gives you job status, other useful things

DAGman

- Define dependencies between the constituent pieces of your application
- DAGman then executes those pieces (using eg. Condor-G) in an order that satisfies those dependencies
- (DAG = Directed Acyclic Graph)

Module: More advanced application techniques

- Introduce software package called Swift
- Use this to construct more complicated grid applications
- Discuss a wider range of issues that are encountered when running on grids

Swift

- Straightforwardly express common patterns in building grid applications
- SwiftScript – a language that is useful for building applications that run on clusters and grids.
- Handles many common problems
- (disclaimer: this is my project)

abstractness

more abstract

less abstract

Swift

DAGman

Condor-G

Globus
Tookit

manual
interaction
with sites

Grid-scale issues

- Where on the grid to run your jobs?
  - How can I find them?
  - How can I choose between them?
- How to gracefully deal with failures?
- How to find out what is wrong?
- How well is application working?
- How can I get my application code installed on the grid?
- How to track where data has come from

Module: More about the cluster

- Digging deeper into the structure of the cluster
- Earlier modules will talk about how to run stuff on the UJ cluster. This module will talk about what the cluster is.

Components of the cluster

- Hardware
  - whats in the rack?
- Software
  - for managing use of the cluster – ensuring fair access
  - providing services for users of the cluster – shared data space
  - monitoring what is happening on the cluster

Module: Guts of the grid

- Learn more about the Open Science Grid
- Technical and political structure of OSG
- Protocols and software used under the covers
  - job submission
  - data transfer
  - site discovery
  - security
- Running your own site

Open Science Grid

The Open Science Grid vision

Transform processing and **data** intensive science through a cross-domain self-managed **national distributed** cyber-infrastructure that **brings together** campus and community infrastructure and facilitating the needs of **Virtual Organizations** (VO) at all scales

Transform processing and **data** intensive science through a cross-domain self-managed **national distributed** cyber-infrastructure that **brings together** campus and community infrastructure and facilitating the needs of **Virtual Organizations** (VO) at all scales

Already seen some example applications: small and large

Transform processing and **data** intensive science through a cross-domain self-managed **national distributed** cyber-infrastructure that **brings together** campus and community infrastructure and facilitating the needs of **Virtual Organizations** (VO) at all scales

self-managed – the participants manage (vs having big OSG HQ running everything)

national – actually international for a few years

distributed – spread throughout the participating institutions

cyber-infrastructure that brings together campus [infrastructure] such as UJ cluster

and

community infrastructure belonging (for example) to collaborations

Transform processing and **data** intensive science through a cross-domain self-managed **national distributed** cyber-infrastructure that **brings together** campus and community infrastructure and facilitating the needs of **Virtual Organizations** (VO) at all scales

OSG is a VO Centric VO

The blue print for the OSG that was developed four years ago states that *"The OSG architecture is Virtual Organization based".* A VO is considered as party to contracts between Resource Providers & VOs which govern resource usage & policies and may consist of sub-VOs which operate under the contracts of the parent.

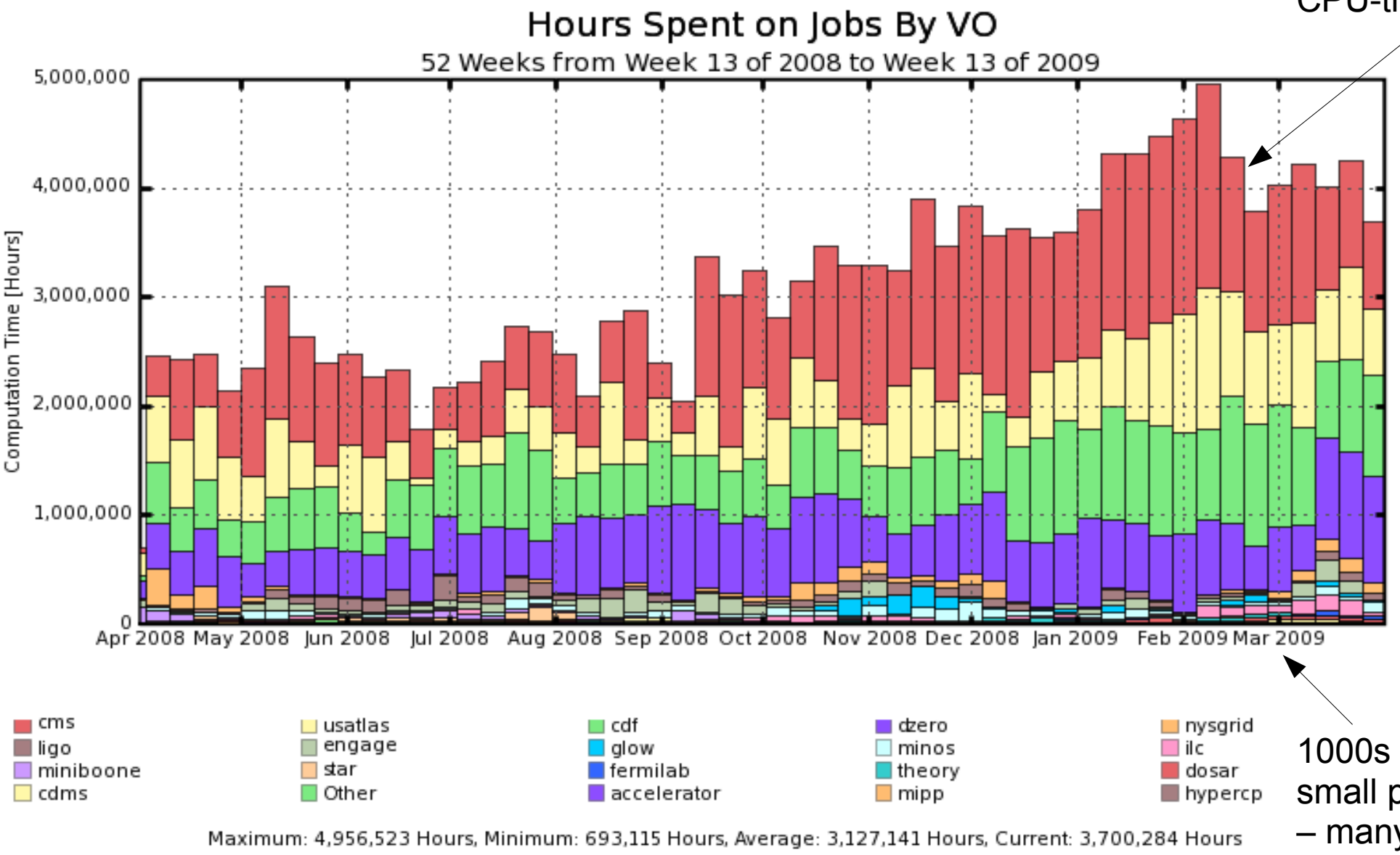
- What makes an organization a VO?
- How do we define relationships between (V)Os?
- Is a user a VO?

Virtual Organisations

- Groupings of participants who consume and provide resources for some particular common purpose
- In OSG, some are very large, some are very small

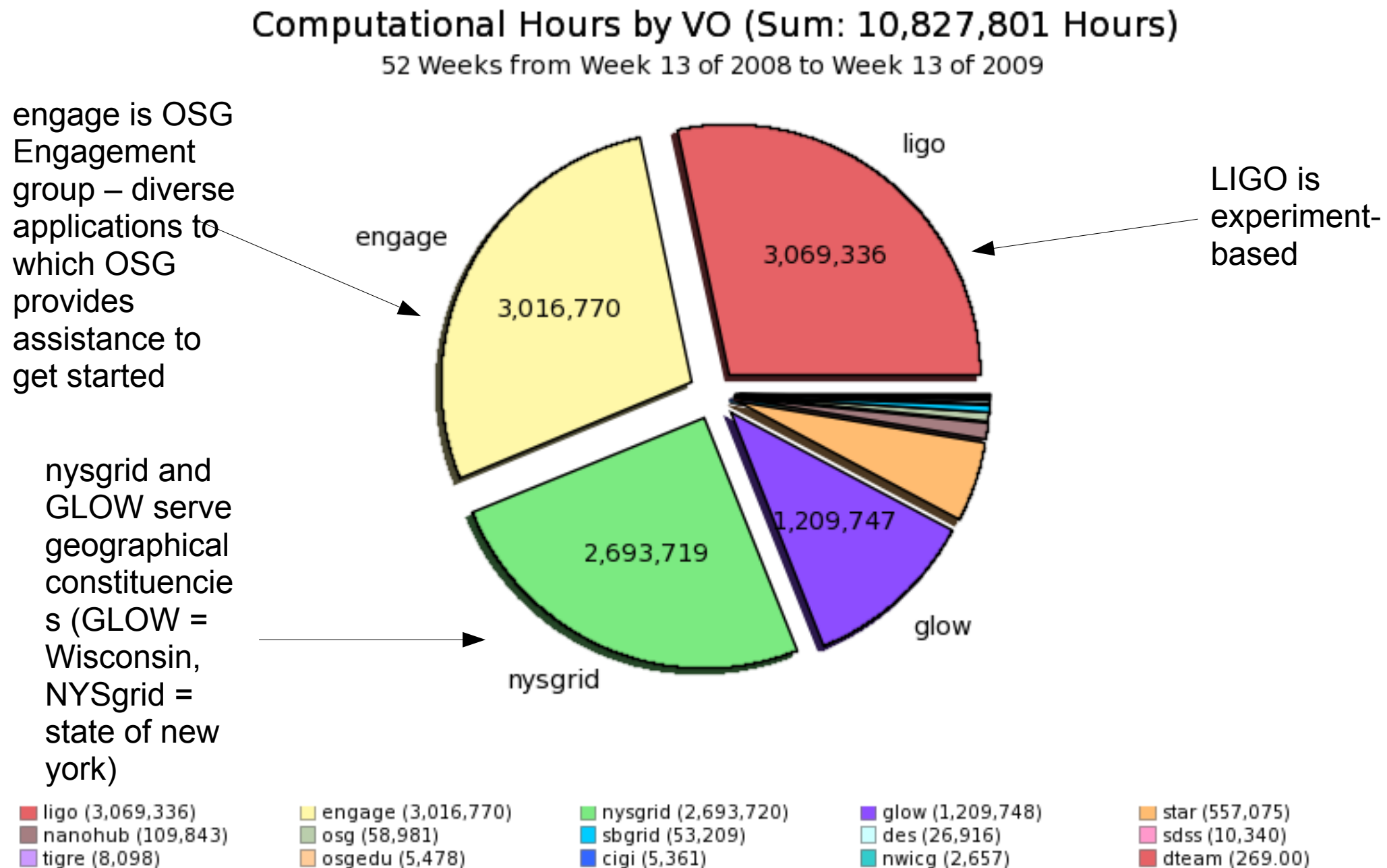
Virtual Organizations (VO) at all scales

big lhc-style experiments dominate CPU-time



1000s hours small projects – many.

other people do large computations too (just not as large)



engage is OSG Engagement group – diverse applications to which OSG provides assistance to get started

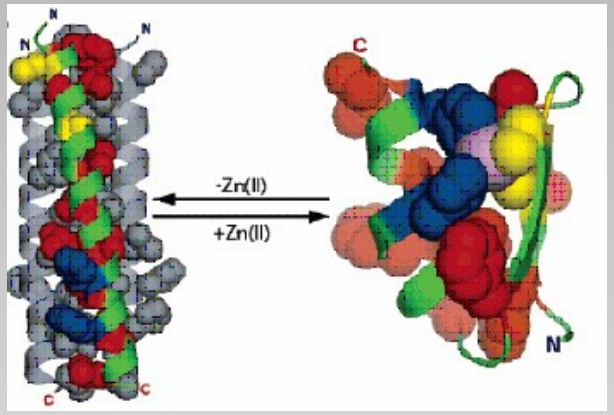
LIGO is experiment-based

nysgrid and GLOW serve geographical constituencies (GLOW = Wisconsin, NYSgrid = state of new york)

Open Science Grid

Protein folding at UNC

- Designing proteins that fold into specific structures and bind target molecules
- Millions of simulations lead to the creation of a few proteins in the wet-lab
- Assistant Professor and a lab of 5 graduate students
- For each protein designed, consume about 5000 CPU hours.
- ~250000 CPU hours consumed so far
- Still doing “wet” science – but using large-scale computing to help



One protein can fold in many ways. This computationally designed protein switches between a zinc finger structure and a coiled-coil structure, depending on its environment.

http://www.isgtw.org/?pid=1000507

]

Other grids

- SA National Grid
  - Bruce will talk on Sunday
- US: TeraGrid
- Europe: EGEE
- Others... (many national grids in various stages of deployment)

Module: Porting your own applications

- Hopefully quite interactive
- Talk about applications that UJ people have, and how people can see porting them to cluster and/or grid.
- Hands-on playing with code?
- Lead into the next 3 weeks...

After this week

- Me at UJ for 3 more weeks specifically to help people get things running on cluster and grid
- Various grid related events such as:
- International Summer School on Grid Computing in France – if you're interested specifically in Grid stuff. July. http://www.issgc.org/