CS 425: Distributed Systems

Lecture 27

“The Grid”

Klara Nahrstedt

Acknowledgement

- The slides during this semester are based on ideas and material from the following sources:
  - Slides prepared by Professors M. Harandi, J. Hou, I. Gupta, N. Vaidya, Y-Ch. Hu, S. Mitra.
  - Slides from Professor S. Gosh's course at University o Iowa.

Administrative

- **MP3 posted**
  - Deadline **December 7** (Monday) – pre-competition
    - Top five groups will be selected for final demonstration on Tuesday, **December 8**
  - Demonstration Signup Sheets for Monday, 12/7, will be made available this week (Thursday, 12/3 lecture)
  - Main Demonstration in front of the Qualcomm Representative will be on Tuesday, December 8 afternoon - details will be announced on Thursday and also on the website and newsgroup

Administrative – MP3

- Don't forget **versioning of your messages** in your protocols between client and server (Google phones are getting quickly obsolete so it will be important to know what version of client software/hardware you are running and synchronize the overall application as we upgrade)
  - **Readme file must include:**
    - Boot-straping routine – how one install your system – developers manuscript
    - How one use your system – usage prescription for users
    - Known bugs, what are the issues with your system/application
  - **Tar or zip your source code and upload it to agora wiki**
    - URL Information will be provided on the web/in class/on newsgroup
  - **Fill out project template as specified**

Administrative

- **MP3 instructions**
  - Here's the template page for cs425 students to copy and fill out.

https://agora.cs.illinois.edu/display/mlc/cs425-TemplateProject

- Website only cs425 students and instructors can access to post the template page and also upload attachments

https://agora.cs.illinois.edu/display/mlc/cs425-fa09-projects

Plan for Today

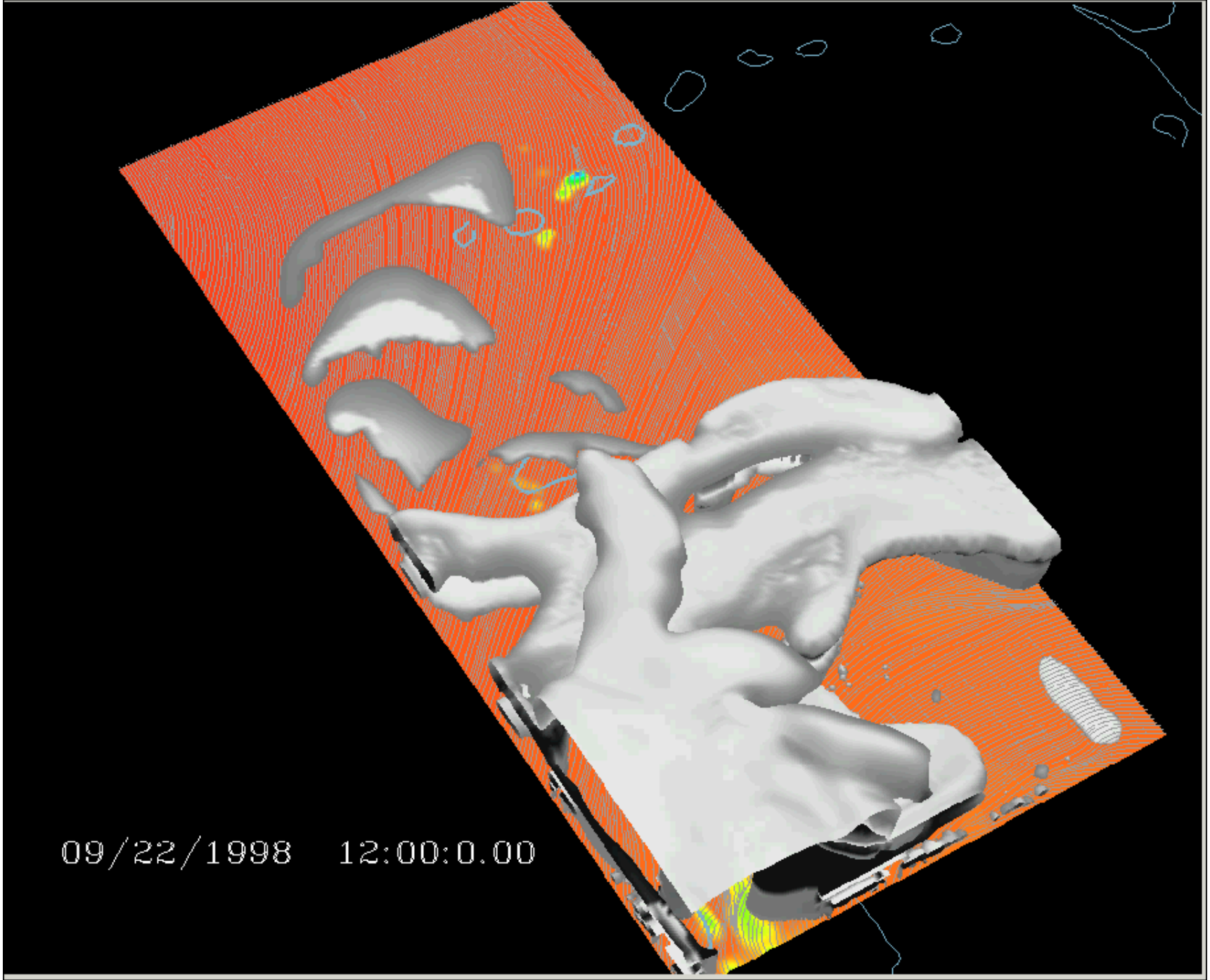
- Discussion what is “Grid” distributed computing paradigm
- Some basic capabilities of Grid and tools/protocols/services that drive Grid
- Comparison between Grid and P2P

Sample Grid Applications

- Astronomers: SETI@Home
- Physicists: data from particle colliders
- Meteorologists: weather prediction
- Bio-informaticians
- ….

Example: Rapid Atmospheric Modeling System, Colorado State University

- Weather Prediction is inaccurate
- Hurricane Georges, 17 days in Sept 1998



cloud0001.avi

- Hurricane Georges, 17 days in Sept 1998
  - "RAMS modeled the mesoscale convective complex that dropped so much rain, in good agreement with recorded data"
  - Used 5 km spacing instead of the usual 10 km
  - Ran on 256+ processors

Recently: Large Hadron Collider

- http://lcg.web.cern.ch/lcg/
- LHC@home

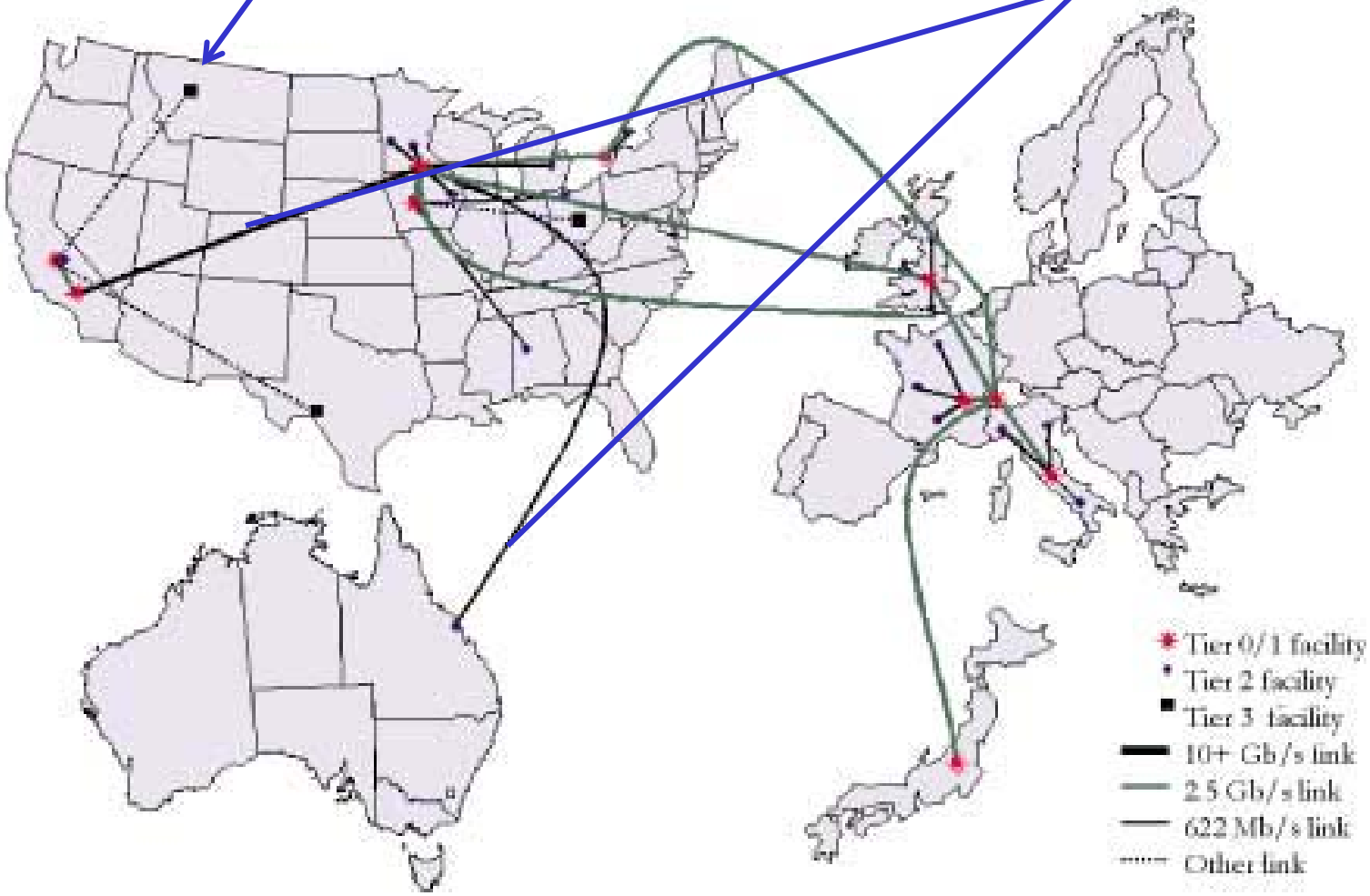
"LHC collisions will produce 10 to 15 petabytes of data a year"

http://www.techworld.com/mobility/features/index.cfm?featureid=4074&pn=2

The Grid

Each location is a cluster

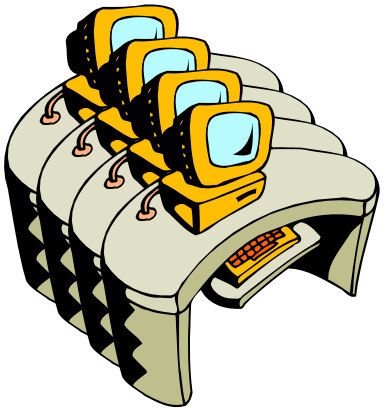
Some are 40Gbps links!
(The TeraGrid links)



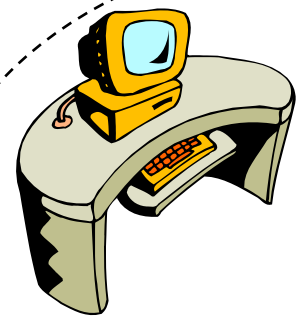
“A parallel Internet”

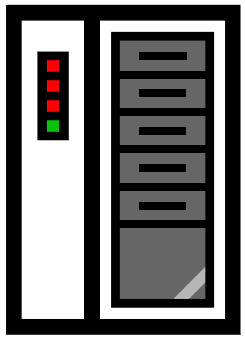
Distributed Computing Resources in Grid

Wisconsin

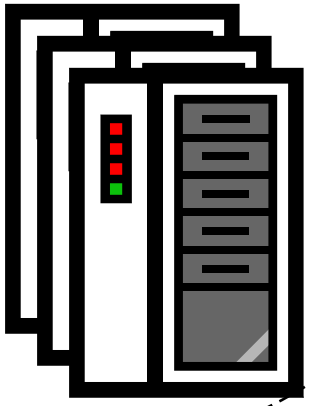


MIT

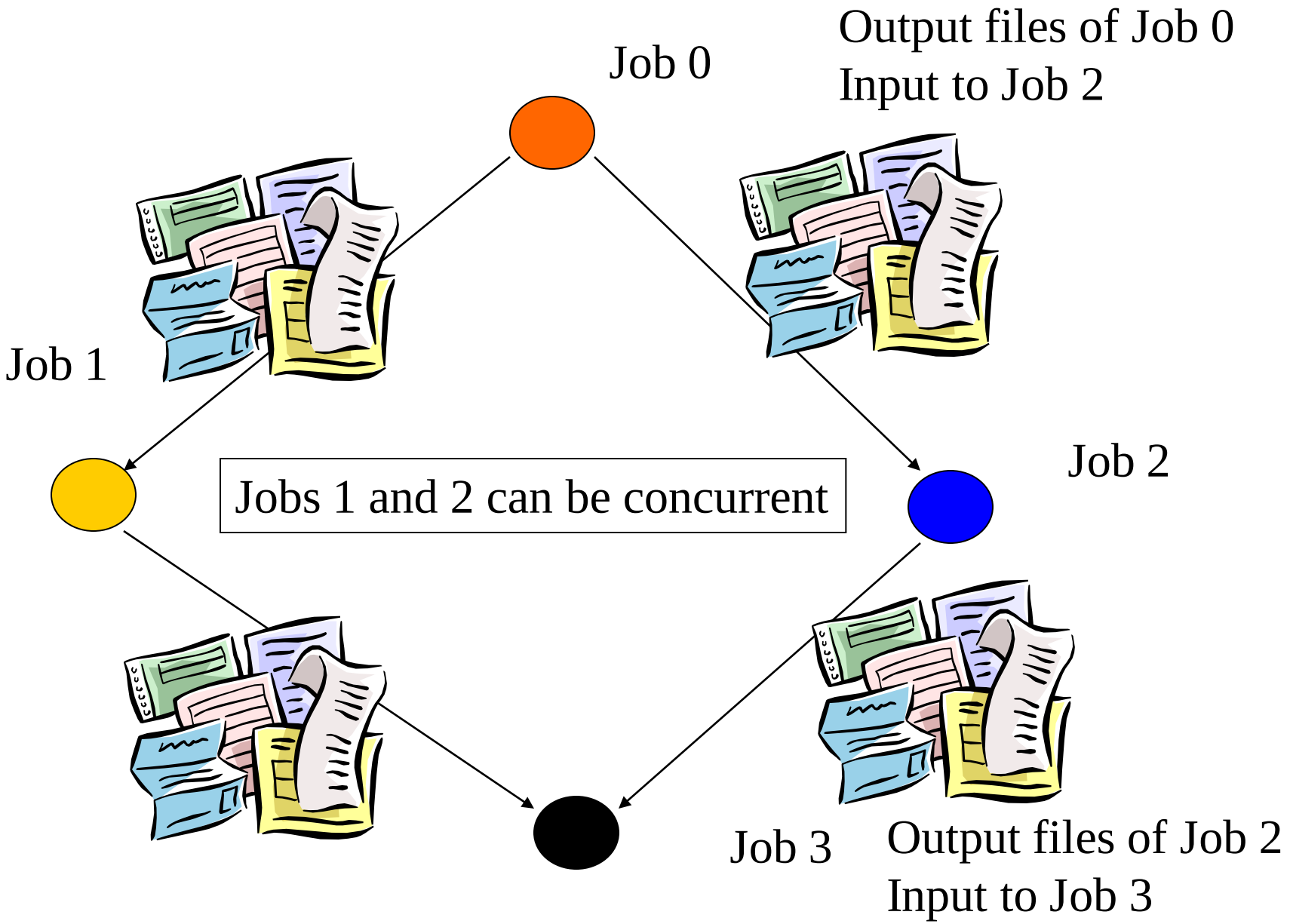




NCSA/UIUC



Application Coded by a Meteorologist



Application Coded by a Meteorologist

Output files of Job 0
Input to Job 2

Several GBs

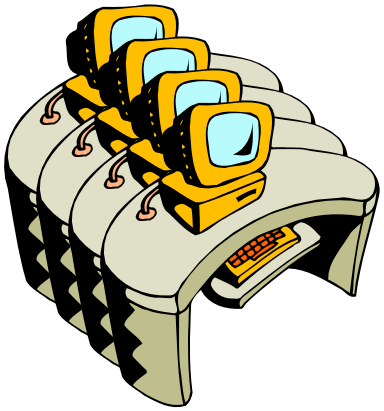
May take several hours/days
4 stages of a job
- Init
- Stage in
- Execute
- Stage out
- Publish

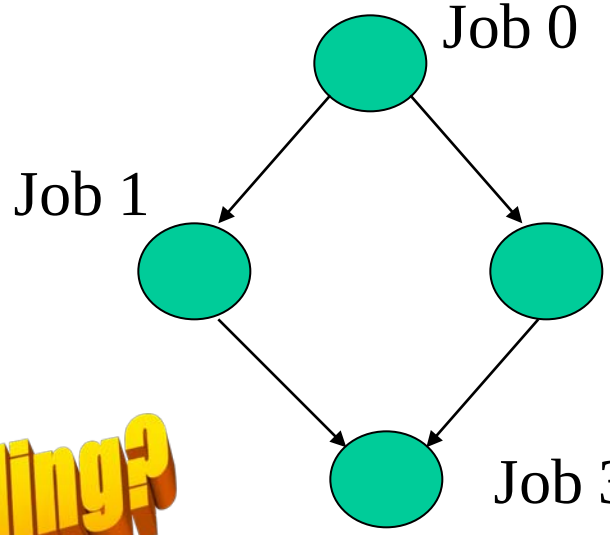
Computation Intensive,
so Massively Parallel

Job 2

Output files of Job 2
Input to Job 3

Wisconsin



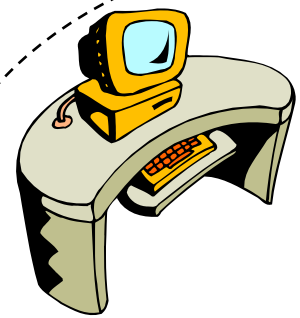


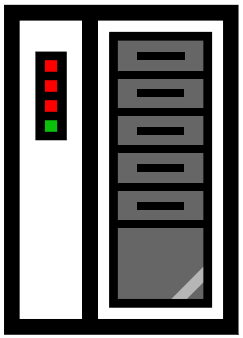
Allocation?

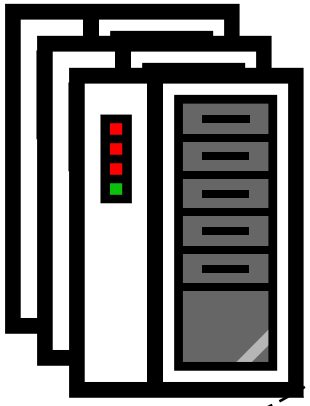
Scheduling?

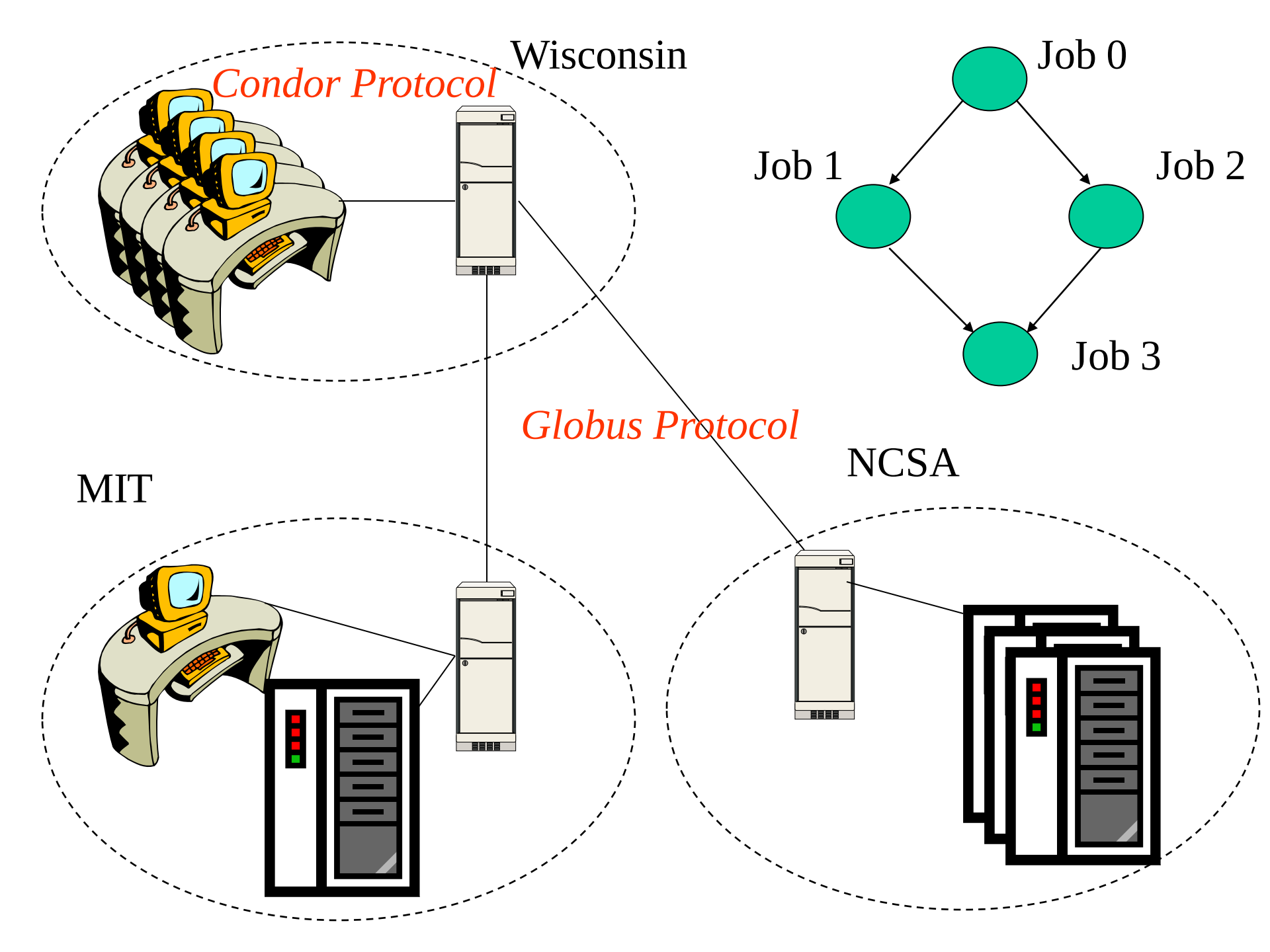
MIT

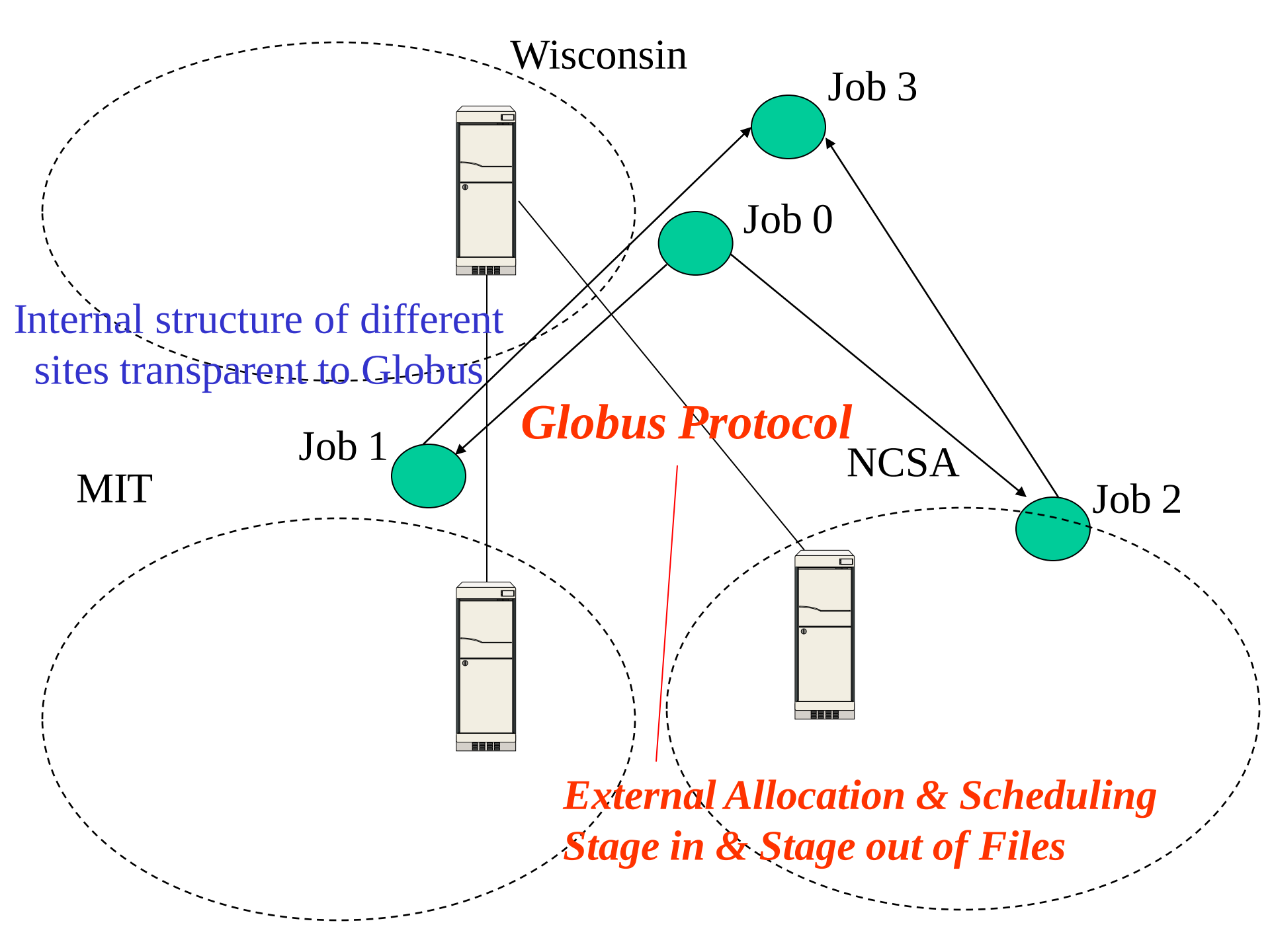
NCSA

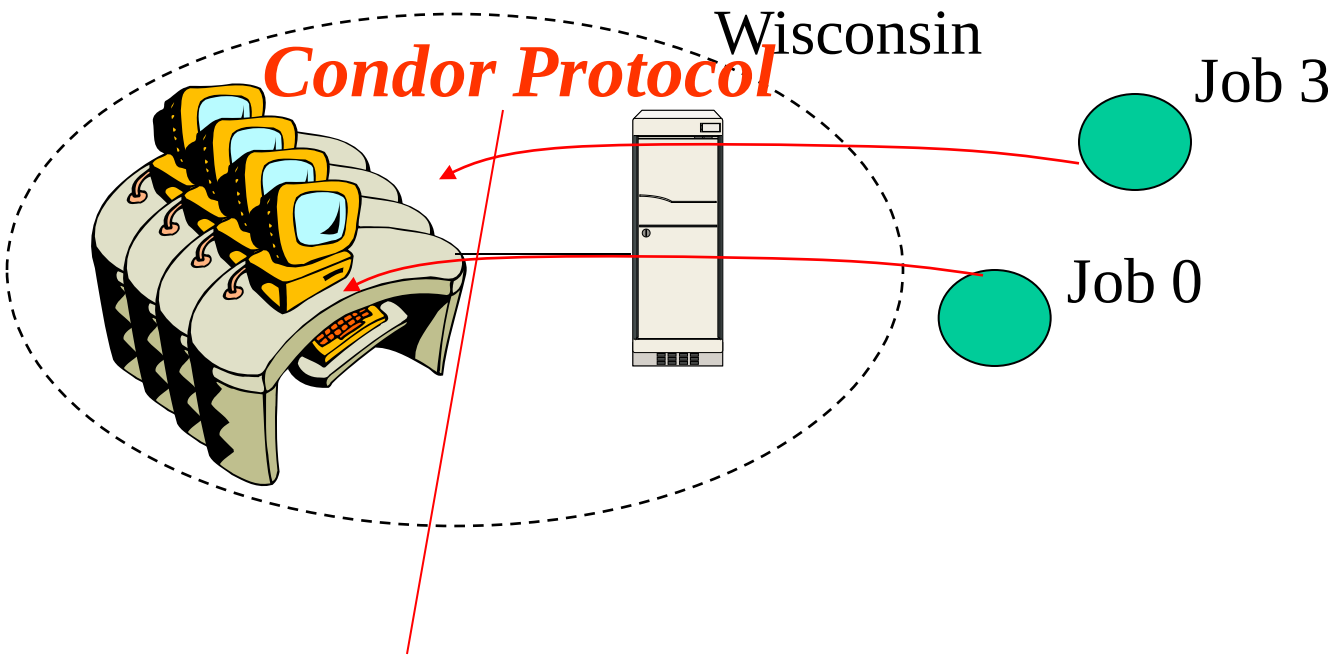






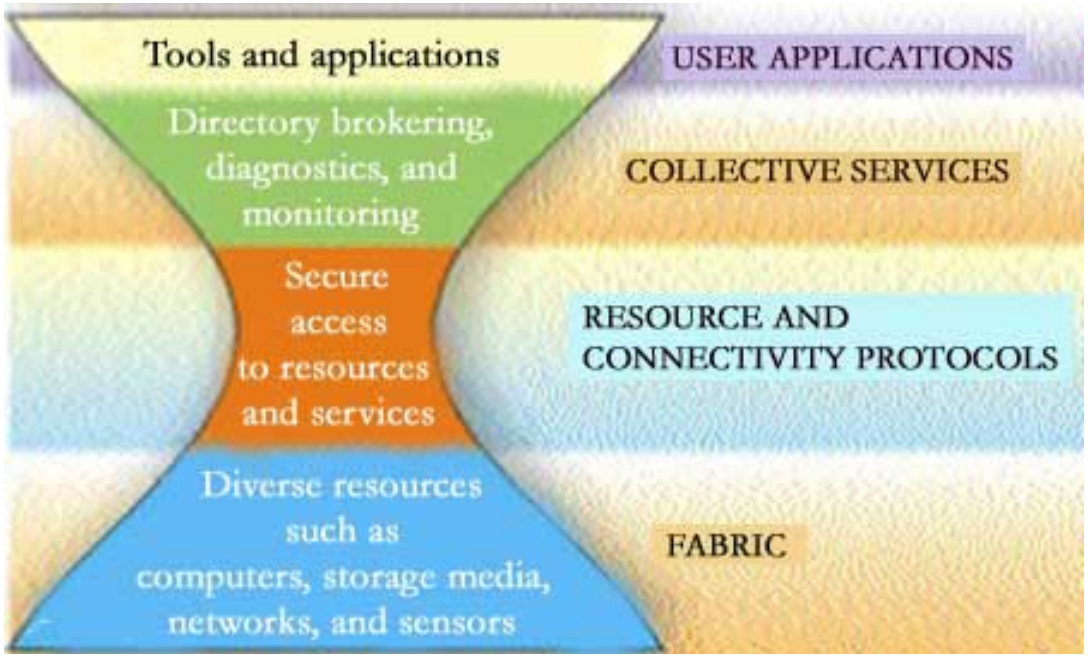

Wisconsin
Condor Protocol
Job 0
Job 1
Job 2
Job 3
Globus Protocol
MIT
NCSA


Wisconsin
Job 3
Job 0
Internal structure of different sites transparent to Globus
Globus Protocol
Job 1
MIT
NCSA
Job 2
External Allocation & Scheduling
Stage in & Stage out of Files



Internal Allocation & Scheduling
Monitoring
Distribution and Publishing of Files
Resource Matchmaking 'ClassAd' concept

Tiered Architecture (OSI 7 layer-like)



High energy Physics apps

Globus

e.g., Condor

Workstations, LANs

Trends: Technology

- **Doubling Periods –** storage: 12 mos, bandwidth: 9 mos, and (what law is this?) cpu speed/capacity: 18 mos
- Then and Now

  Bandwidth

  – 1985: mostly 56Kbps links nationwide
  – 2003: 155 Mbps links widespread
  – 2009: 1 Gbps links wide spread

  Disk capacity

  – Today's PCs have 100GBs, and clusters – terabytes/petabytes, same as a 1990 supercomputer

Trends: Users

- Then and Now

  Biologists:

  – 1990: were running small single-molecule simulations

  – 2003: want to calculate structures of complex macromolecules, want to screen thousands of drug candidates

  Physicists

  – 2006: CERN's Large Hadron Collider produced about 10^15 B during the year

- Trends in Technology and User Requirements: Independent or Symbiotic?

Globus Alliance

- Alliance involves U. Illinois Chicago, Argonne National Laboratory, USC-ISI, U. Edinburgh, Swedish Center for Parallel Computers, NCSA
- Activities : research, testbeds, software tools, applications
- Globus Toolkit (latest ver – GT4)

  "The Globus Toolkit includes software services and libraries for resource monitoring, discovery, and management, plus security and file management. Its latest version, GT3, is the first full-scale implementation of new Open Grid Services Architecture (OGSA)."

More

- Entire community, with multiple conferences, get-togethers (GGF), and projects
- Grid Projects:

  http://www-fp.mcs.anl.gov/~foster/grid-projects

- Grid Users:
  - Today: Core is the physics community (since the Grid originates from the GriPhyN project)
  - Tomorrow: biologists, large-scale computations (nug30 already)?

Prophecies

In 1965, MIT's Fernando Corbató and the other designers of the **Multics operating system** envisioned a computer facility operating “like a power company or water company”.

Plug your thin client into the computing Utility

and Play your favorite Intensive Compute & Communicate Application

- [Will this be a reality with the Grid?]

Recap: Grid vs. …

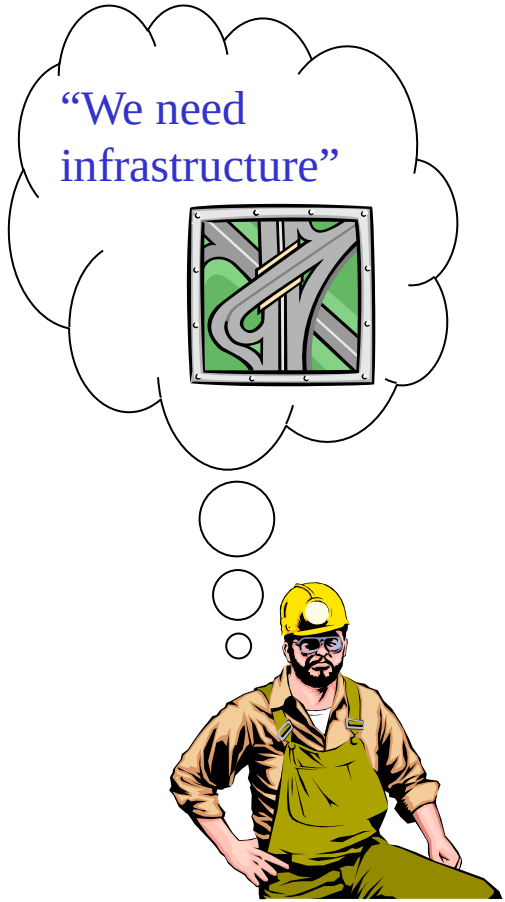
- LANs?
- Supercomputers?
- Clusters?
- Cloud?

What separates these? The same technologies?

…P2P???

P2P

Grid



Definitions

Grid

- “Infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities” (1998)

P2P

- “Applications that takes advantage of resources at the edges of the Internet” (2000)

Definitions

Grid

- "Infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities" (1998)
- "A system that coordinates resources not subject to centralized control, using open, general-purpose protocols to deliver nontrivial QoS" (2002)

P2P

- "Applications that takes advantage of resources at the edges of the Internet" (2000)
- "Decentralized, self-organizing distributed systems, in which all or most communication is symmetric" (2002)

Definitions

Grid

- "Infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities" (1998)
- "A system that coordinates resources not subject to centralized control, using open, general-purpose protocols to deliver nontrivial QoS" (2002)

(good legal applications without intellectual fodder)

P2P

- "Applications that takes advantage of resources at the edges of the Internet" (2000)
- "Decentralized, self-organizing distributed systems, in which all or most communication is symmetric" (2002)

(clever designs without good, legal applications)

Grid versus P2P
- Pick your favorite

Applications

Grid

- Often complex & involving various combinations of
  - Data manipulation
  - Computation
  - Tele-instrumentation
- Wide range of computational models, e.g.
  - Embarrassingly ||
  - Tightly coupled
  - Workflow
- Consequence
  - Complexity often inherent in the application itself

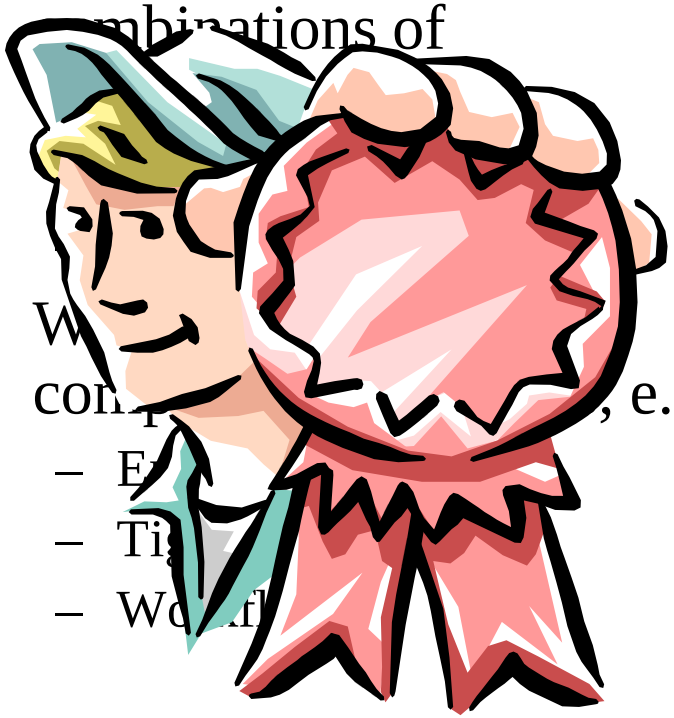
P2P

- Some
  - File sharing
  - Number crunching
  - Content distribution
  - Measurements
- Legal Applications?
- Consequence
  - Low Complexity

Applications

Grid

- Often complex & involving various combinations of



- W
  com , e.g.
  - E
  - Ti
  - W
- Consequence
  - Complexity often inherent in the application itself

P2P

- Some
  - File sharing
  - Number crunching
  - Content distribution
  - Measurements
- Legal Applications?
- Consequence
  - Low Complexity

Scale and Failure

Grid

- Moderate number of entities
  - 10s institutions, 1000s users
- Large amounts of activity
  - 4.5 TB/day (D0 experiment)
- Approaches to failure reflect assumptions
  - e.g., centralized components

P2P

- V. large numbers of entities

FastTrackC	4,277,745
iMesh	1,398,532
eDonkey	500,289
DirectConnect	111,454
Blubster	100,266
FileNavigator	14,400
Ares	7,731

(www.slyck.com, 2/19/'03)

- Moderate activity
  - E.g., 1-2 TB in Gnutella ('01)
- Diverse approaches to failure
  - Centralized (SETI)
  - Decentralized and Self-Stabilizing

Scale and Failure

Grid

- Moderate number of entities
  - 10s institutions, 1000s users
- Large amounts of activity
  - 4.5 TB/day (D0 experiment)
- Approaches to failure reflect assumptions
  - E.g., centralized components

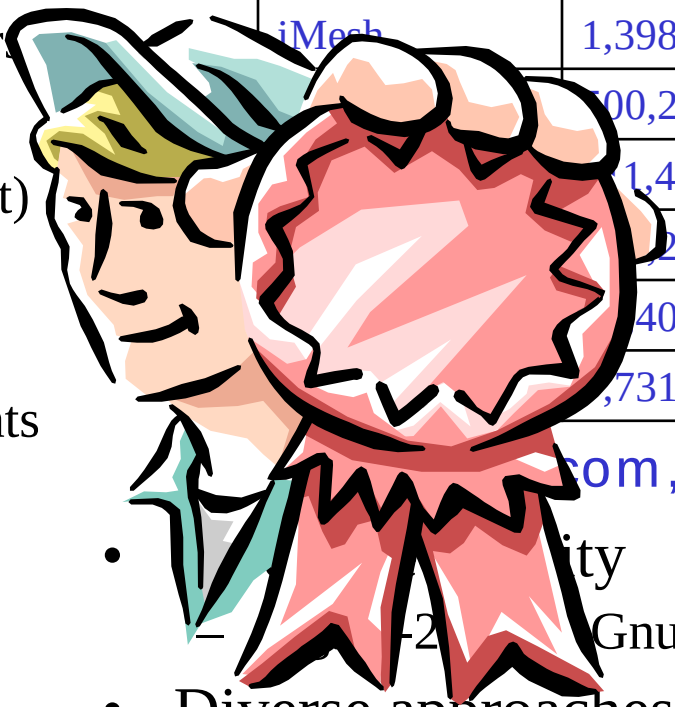
P2P

- V. large numbers of entities

FastTrackC	4,277,745
iMesh	1,398,532
	00,289
	1,454
	266
	400
	,731

com, 2/19/03)

- ity
  - -2 Gnutella (' 01)
- Diverse approaches to failure
  - Centralized (SETI)
  - Decentralized and Self-Stabilizing




Some Things Grid Researchers Consider Important

- Single sign-on: collective job set should require once-only user authentication
- Mapping to local security mechanisms: some sites use Kerberos, others using Unix
- Delegation: credentials to access resources inherited by subcomputations, e.g., job 0 to job 1
- Community authorization: e.g., third-party authentication

Services and Infrastructure

Grid

- Standard protocols (Global Grid Forum, etc.)
- De facto standard software (open source Globus Toolkit)
- Shared infrastructure (authentication, discovery, resource access, etc.)

Consequences

- Reusable services
- Large developer & user communities
- Interoperability & code reuse

P2P

- Each application defines & deploys completely independent "infrastructure"
- JXTA, BOINC, XtremWeb?
- Efforts started to define common APIs, albeit with limited scope to date

Consequences

- New (albeit simple) install per application
- Interoperability & code reuse not achieved

Services and Infrastructure

Grid

- Standard protocols (Global Grid Forum, etc.)
- [illegible]ftware
- S[illegible] (au[illegible]very, reso[illegible])

Conseq[illegible]

- Reusabl[illegible]ic
- Large developer & user communities
- Interoperability & code reuse

P2P

- Each application defines & deploys completely independent "infrastructure"
- JXTA, BOINC, XtremWeb?
- Efforts started to define common APIs, albeit with limited scope to date

Consequences

- New (albeit simple) install per application
- Interoperability & code reuse not achieved

Summary: Grid and P2P

1) Both are concerned with the same general problem
   - Resource sharing within virtual communities
2) Both take the same general approach
   - Creation of overlays that need not correspond in structure to underlying organizational structures
3) Each has made genuine technical advances, but in complementary directions
   - “Grid addresses infrastructure but not yet failure”
   - “P2P addresses failure but not yet infrastructure”
4) Complementary strengths and weaknesses => room for collaboration (Ian Foster)

EXTRA

Grid History – 1990's

- **CASA** network: linked 4 labs in California and New Mexico
  - Paul Messina: Massively parallel and vector supercomputers for computational chemistry, climate modeling, etc.
- **Blanca**: linked sites in the Midwest
  - Charlie Catlett, NCSA: multimedia digital libraries and remote visualization
- More testbeds in Germany & Europe than in the US
- **I-way** experiment: linked 11 experimental networks
  - Tom DeFanti, U. Illinois at Chicago and Rick Stevens, ANL:, for a week in Nov 1995, a national high-speed network infrastructure. 60 application demonstrations, from distributed computing to virtual reality collaboration.
- **I-Soft**: secure sign-on, etc.