

CS 425 / ECE 428  
Distributed Systems  
Fall 2026

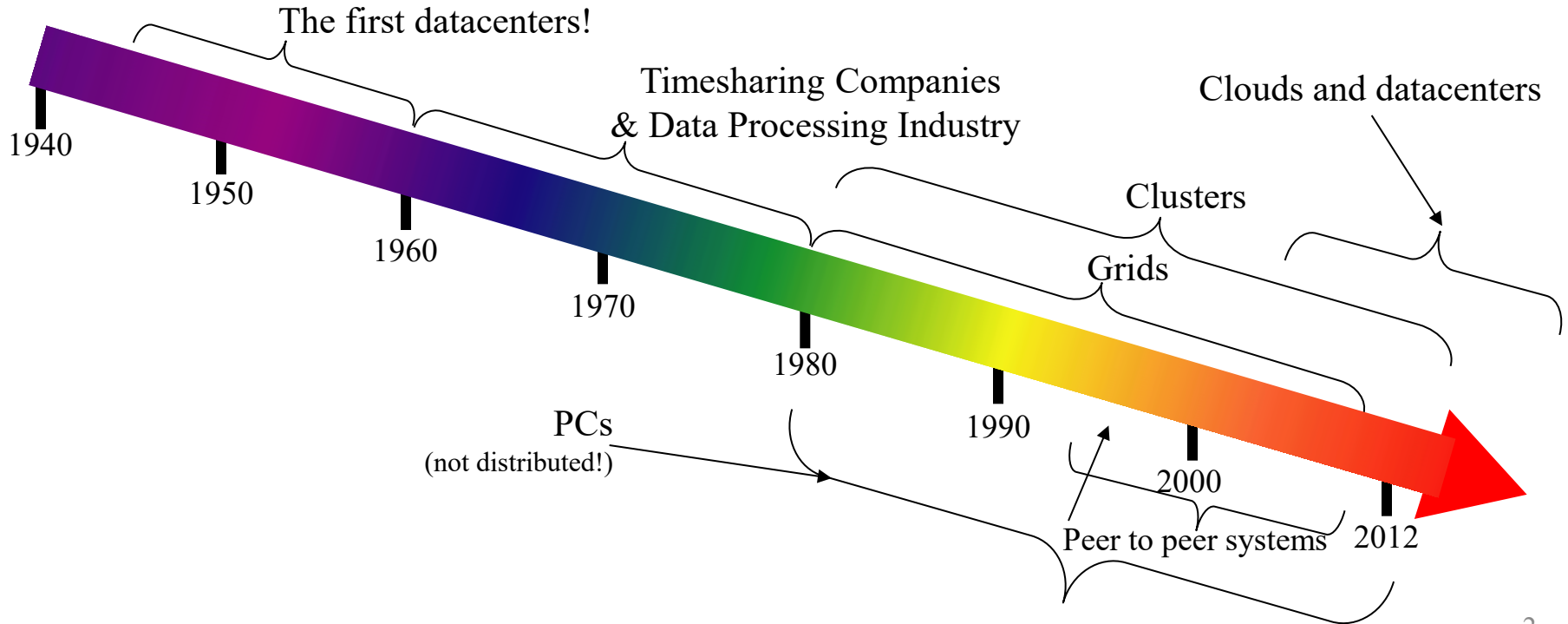
Aishwarya Ganesan  
w/ Ram Kesavan

Ack: Indranil Gupta (Indy)

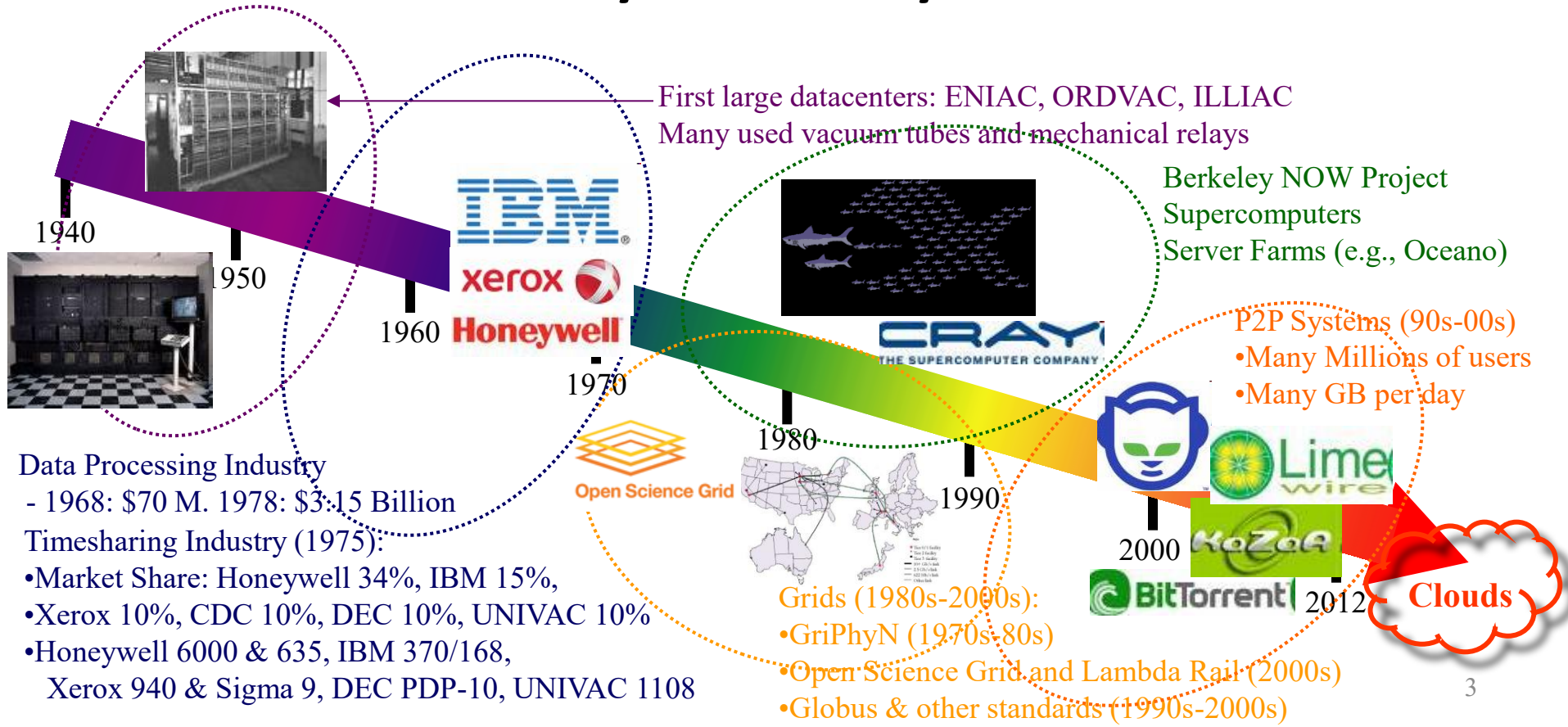
*Lecture: Grids*

All slides © IG

# “A Cloudy History of Time”



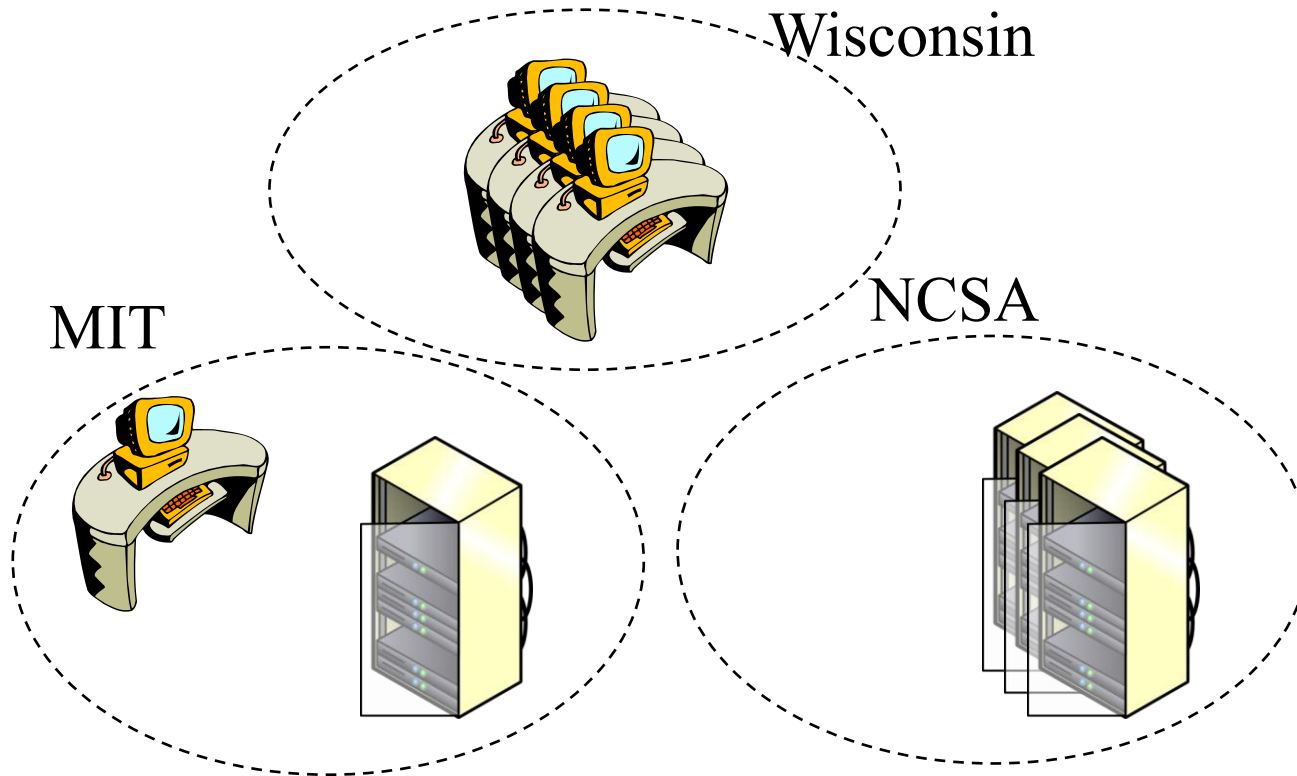
# “A Cloudy History of Time”



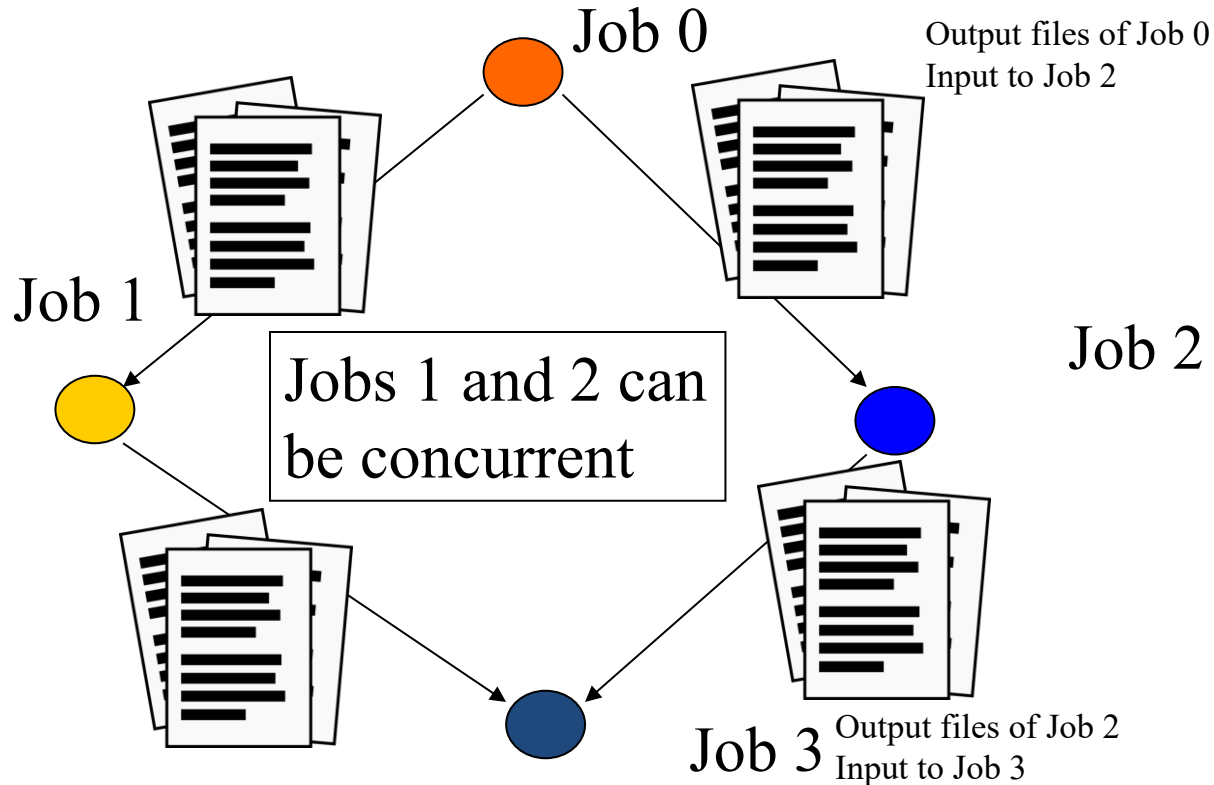
# Example: Rapid Atmospheric Modeling System, ColoState U

- 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
- Computation-intensive computing (or HPC = high performance computing)
- *Can one run such a program without access to a supercomputer?*

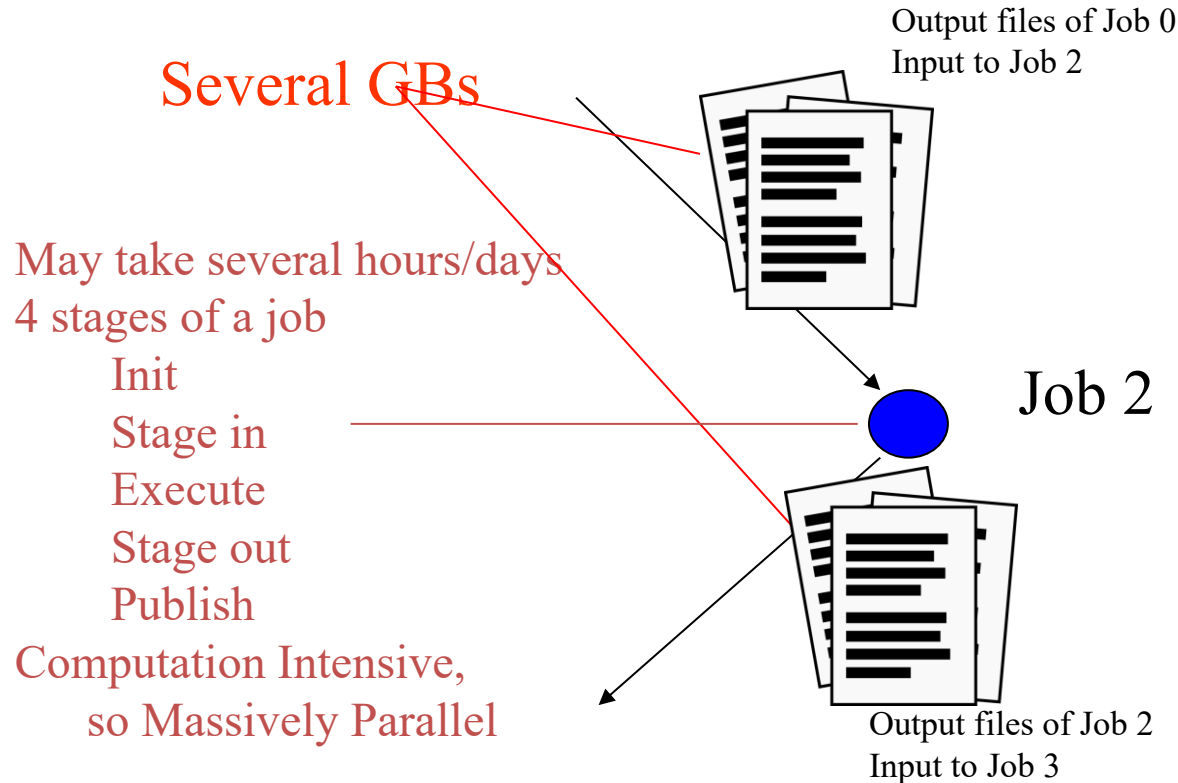
# Distributed Computing Resources



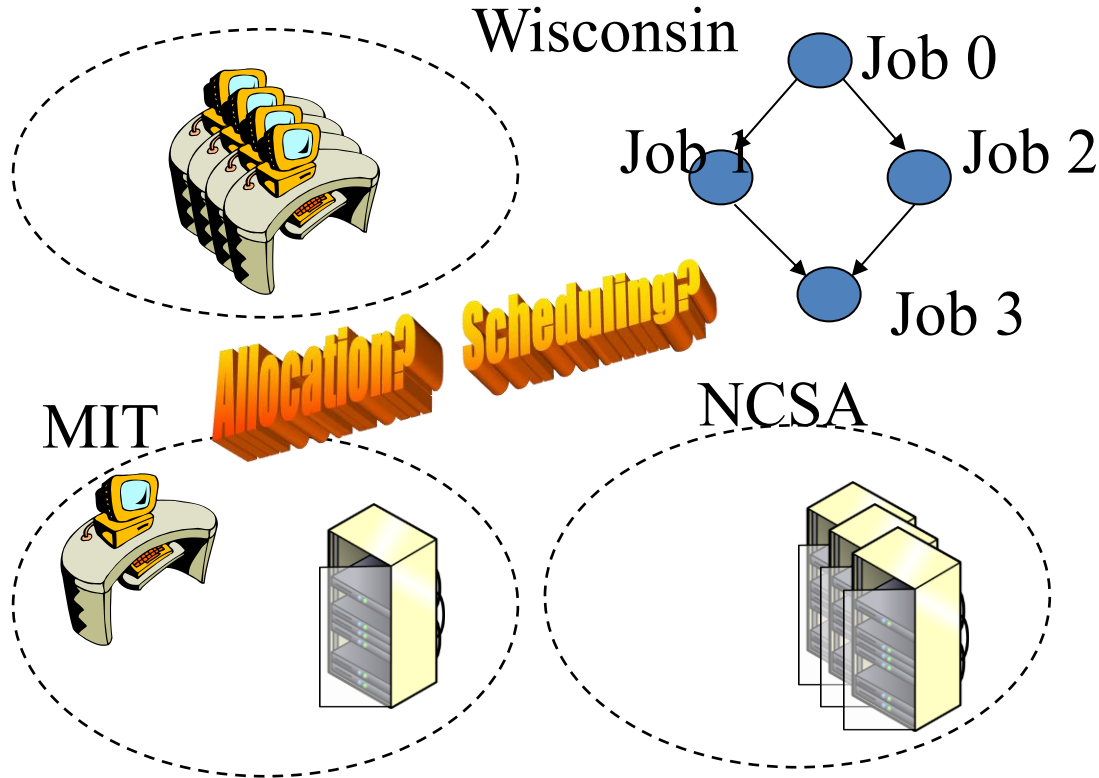
# An Application Coded by a Physicist



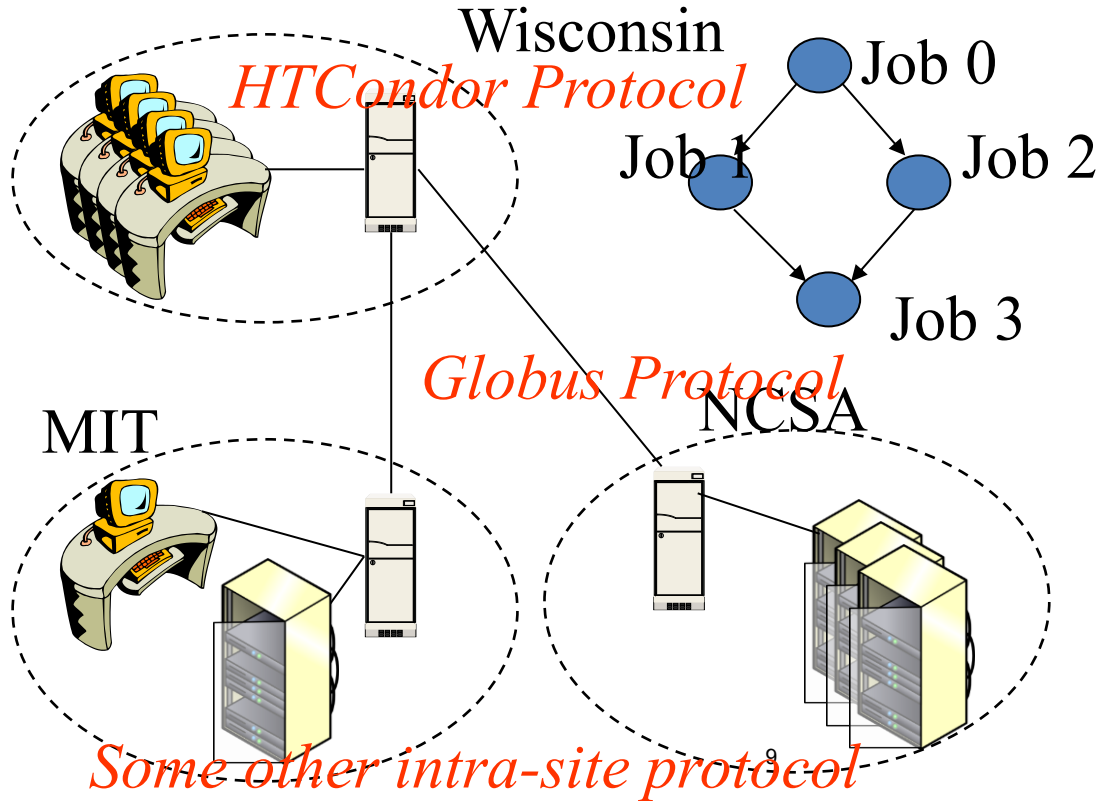
# An Application Coded by a Physicist



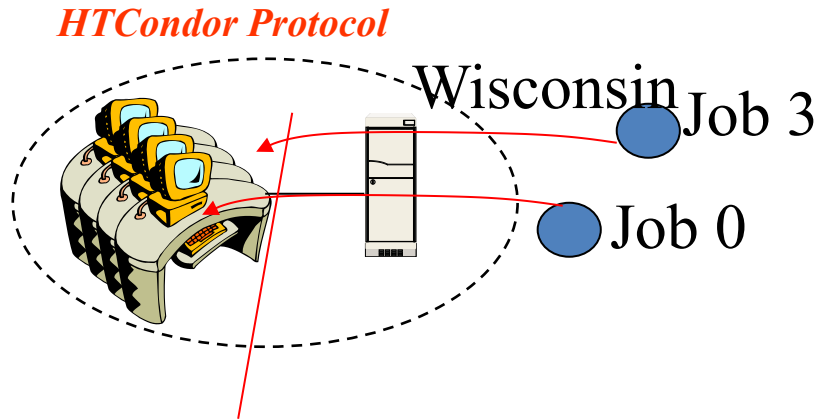
# Scheduling Problem



# 2-level Scheduling Infrastructure



# Intra-site Protocol



*Internal Allocation & Scheduling*  
*Monitoring*  
*Distribution and Publishing of Files*

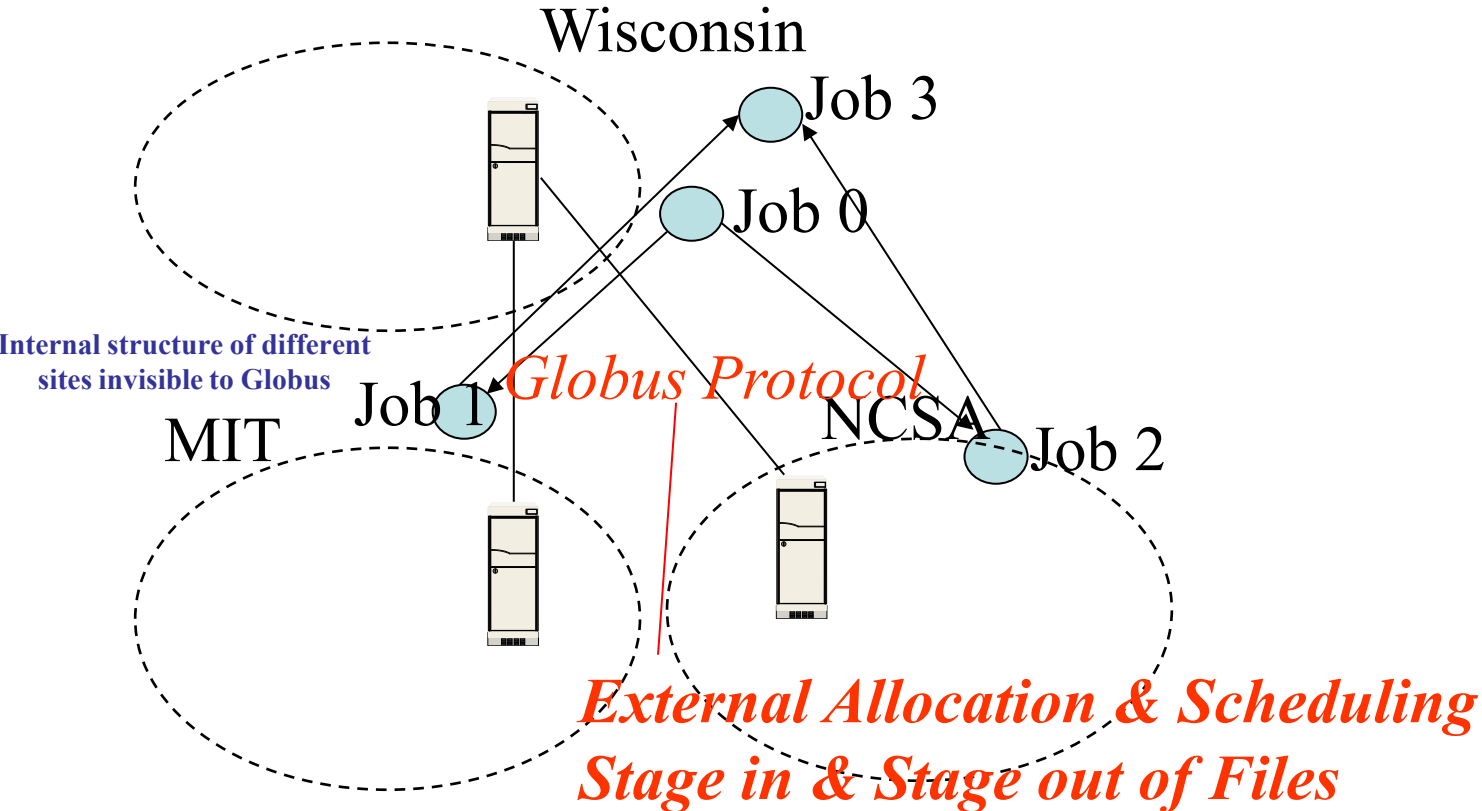
# Condor (now HTCondor)

- High-throughput computing system from U. Wisconsin Madison
- Belongs to a class of “Cycle-scavenging” systems
  - SETI@Home and Folding@Home are other systems in this category

Such systems

- Run on a lot of workstations
- When workstation is free, ask site’s central server (or Globus) for tasks
- If user hits a keystroke or mouse click, stop task
  - Either kill task or ask server to reschedule task
- Can also run on dedicated machines

# Inter-site Protocol



# Globus

- Globus Alliance involves universities, national US research labs, and some companies
- Standardized several things, especially software tools
- Separately, but related: Open Grid Forum
- Globus Alliance has developed the Globus Toolkit

<http://toolkit.globus.org/toolkit/>

# Globus Toolkit

- Open-source
- Consists of several components
  - [GridFTP](#): Wide-area transfer of bulk data
  - [GRAM5](#) (Grid Resource Allocation Manager): submit, locate, cancel, and manage jobs
    - Not a scheduler
    - Globus communicates with the schedulers in intra-site protocols like HTCondor or Portable Batch System (PBS)
  - [RLS](#) (Replica Location Service): Naming service that translates from a file/dir name to a target location (or another file/dir name)
  - Libraries like [XIO](#) to provide a standard API for all Grid IO functionalities
  - Grid Security Infrastructure ([GSI](#))

# Security Issues

- Important in Grids because they are *federated*, i.e., no single entity controls the entire infrastructure
- **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
- These are also important in clouds, but less so because clouds are typically run under a central control
- In clouds the focus is on failures, scale, on-demand nature

# Summary

- Grid computing focuses on computation-intensive computing (HPC)
- Though often federated, architecture and key concepts have a lot in common with that of clouds
- Are Grids/HPC converging towards clouds?
  - E.g., Compare OpenStack and Globus