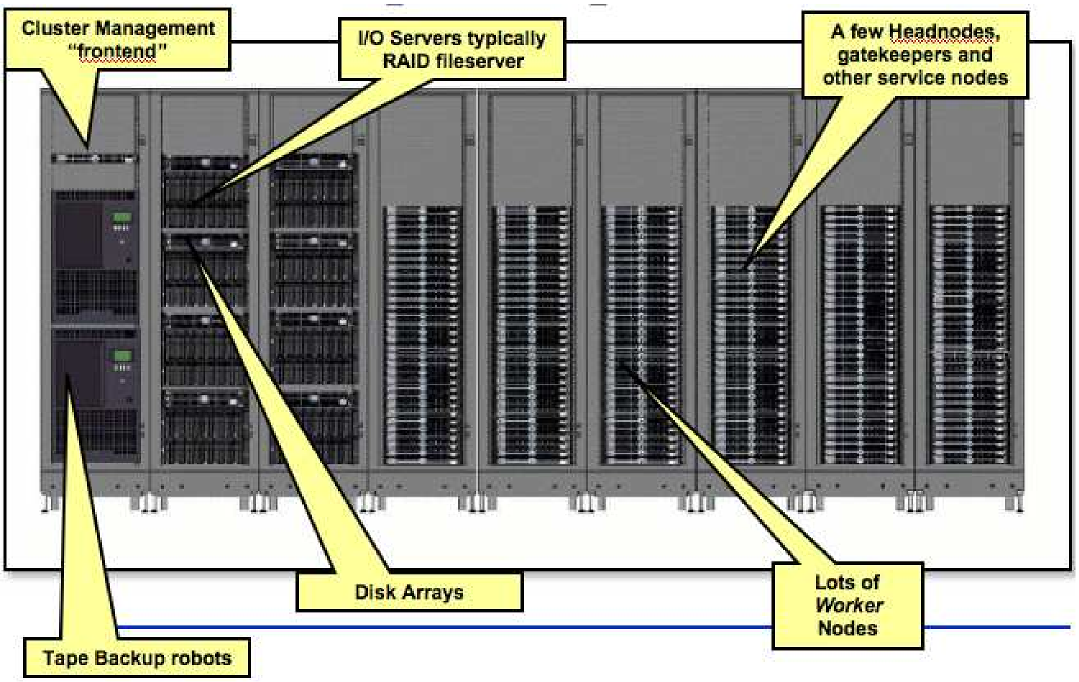
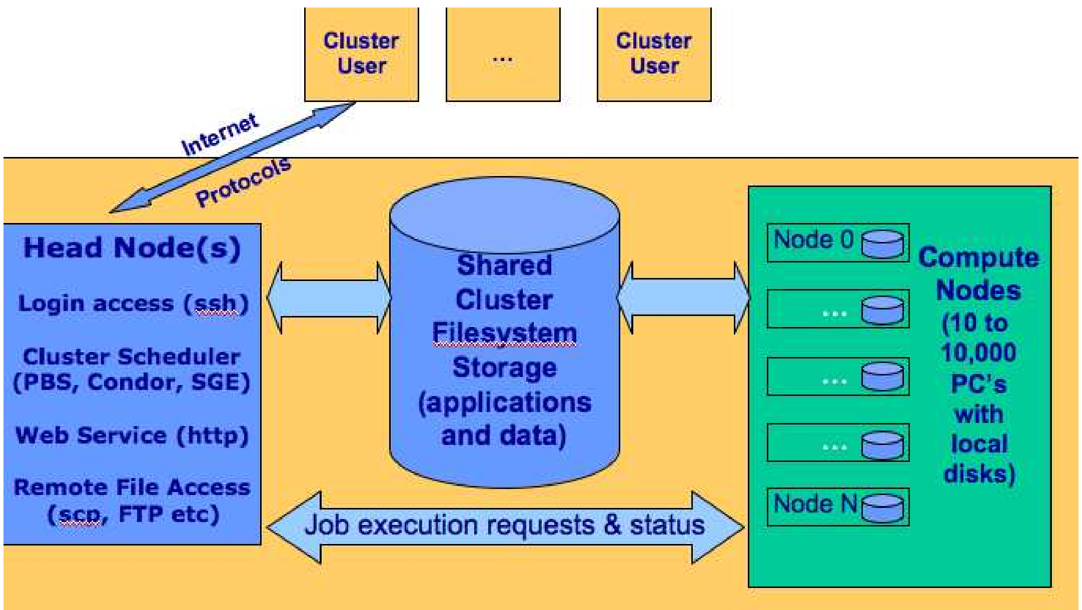
Motivation and History[1]

Introduction

- Computing clusters
  - Current trend in supercomputing



  - Cluster architecture



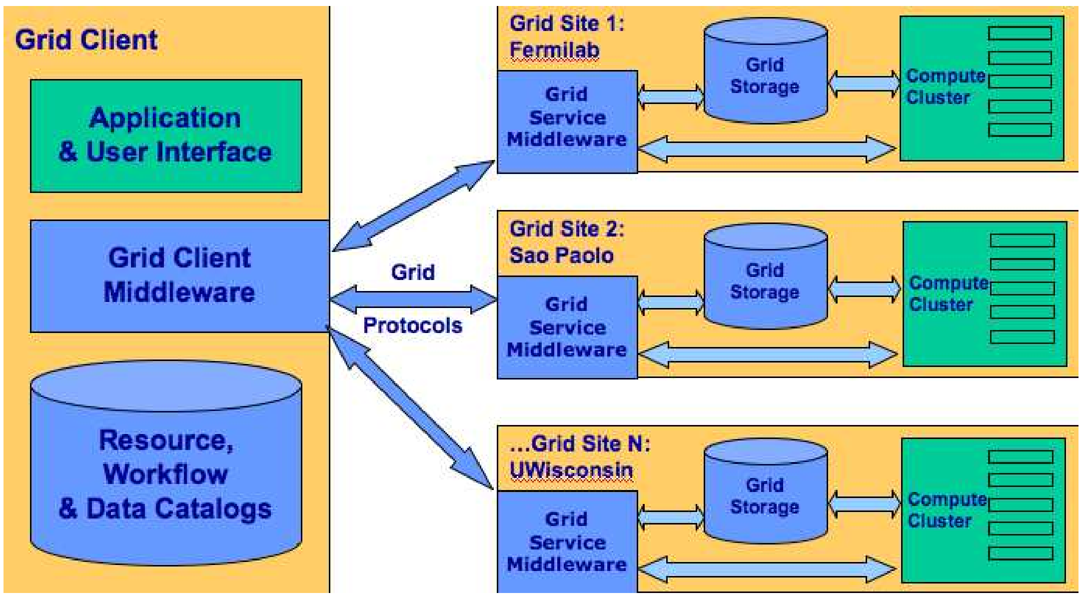
  - Scaling up analysis
    - Query and analysis of > 25 million citations

[1] Most of the material in this set of notes is from the Educational division of Open Science Grid.

- ∗ Work started on desktop workstations
- ∗ Queries grew to month-long duration
- ∗ Data distributed across U of Chicago TeraPort cluster
  - · 50 CPUs gave 100X speedup (30 days vs 1/3rd day)
  - · Many more methods and hypotheses can be tested
- ∗ Higher throughput and capacity enables deeper analysis and broader community access

Grid

- Other names for grid computing: metacomputing, scalable global computing, internet computing
- Distributed clusters
  - Clusters provide a mechanism for distributed computing



  - Grids are distributed sets of clusters
    - ∗ Distributed computing within each cluster
    - ∗ Distributed computing between clusters
- Grid computing extends scientific parallel computing on single machines to distributed systems
- Issues in grid computing
  - Security to control access and protect communication (GSI)
  - Directory to locate grid sites and services (VORS, MDS)
  - Uniform interface to computing sites (GRAM)
  - Facility to maintain and schedule queues of work (Condor-G)
  - Fast and secure data set mover (GridFTP, RFT)
  - Directory to track location for datasets (RLS)
- Processing vast datasets
  - Consider the example from astronomy and high energy physics
    - ∗ Large datasets as inputs (find datasets)

    * Processing the input datasets
    * Output datasets (store and publish)
  - Emphasis on sharing and distribution of these large datasets
  - Workflows of independent program can be parallelized

- Typical good job for grid computing
  - Large varied distributed collection of data
  - Lots of CPU cycles and storage; teraflops and terabytes
  - Share results, code, parameter files
  - Advanced visualization and steering

- Ian Foster's grid checklist
  - Coordinate resources not subject to centralized control
  - Uses standard, open, general-purpose protocols and interfaces
  - Delivers non-trivial quality of service
    * Data management
    * Resource discovery and information
    * Authentication and authorization
    * Accounting and tracking
    * Job management
    * Response time, security, throughput

- Virtual organizations
  - Groups of organizations that use the grid to share resources for specific purposes
  - Support a single community
  - Deploy compatible technology and agree on working policies
    * Security policies – difficult
  - Deploy different network accessible services
    * Grid information
    * Grid resource brokering
    * Grid monitoring
    * Grid accounting

- Grid middleware stack

Grid Application (often includes a Portal)		
Workflow system (explicit or ad-hoc)		
Job management	Data management	Grid information services
Grid security Infrastructure		
Core Globus services		
Standard network protocols and web services		

  - Job management
    * Multiple layer in itself
    * Client queuing system (Condor G)
      · Facility to maintain and schedule queues of work
    * GRAM – Grid Resource Access and Management

      - Uniform interface to computing sites
    - Interface to schedulers
  - Job-oriented models
    - Run an application program; get a result
  - Resources
    - Grid sites are physical collections of resources
    - Configuration and status
    - Directory to locate grid sites and services – VORS, MDS
  - Core Globus services
    - Globus used to deploy the most common core grid infrastructure
    - API level services to write grid middleware applications
    - Higher level services researched and built using Globus
- Quality of service
  - Data management
    - Fast and secure data set movers – GridFTP, RFT
    - Directory to track dataset location – RLS
  - Resource discovery and information
  - Authentication and authorization (access control) – GSI
  - Accounting and tracking
  - Job management
  - Response time, security (communication protection), throughput

Globus and Condor

- Globus Toolkit – base middleware
  - Client tools, usable from command line
  - APIs – scripting languages, C, C++, java – to build your own tools, or use direct from applications
  - Web service interfaces
  - Higher level tools built from basic components, for example, RFT (Reliable File Transfer)
- Condor – for client and server scheduling
  - An agent to queue, schedule, and manage work submission

Open Science Grid

- US grid computing infrastructure
- Supports scientific computing via an open collaboration of science researchers, software developers, and computing, storage, and network providers

Grid Architecture

- Evolving into a service-oriented approach

- Users compose workflows
- Workflows invoke application services
- Application services provide provisioning of resources

- Two layers
  1. Service-oriented applications
     - Wrap applications as services
     - Compose applications into workflows
  2. Service-oriented grid infrastructure
     - Provision physical resources to support application workloads

- Provisioning
  - Assemble and configure resources to meet user needs
  - Make sure resource will do what is desired, with the right quality of service
  - Tasks range from reservation to configuration to ...

- Virtualization
  - Separation of concerns between provider and consumer of “content”
  - Client and service
  - Service/resource provider
  - Need to sustain desired qualities of service despite dynamic environment (management)