Grid Activities in CMS

Asad Samar (Caltech)

PPDG meeting, Argonne
July 13-14, 2000

Introduction

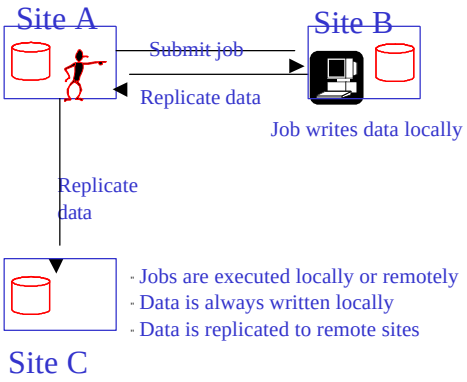
- Need for CMS Grid activities
- Requirement Specification
- Distributed Data Management
  - Functionality required
  - Architecture
- Distributed Computation
  - Functionality required
  - Architecture
- Common Components
- Deliverables, mile stones and status
- Meat for other projects
- Conclusions

Need for CMS Grid Activities

- Other related Grid projects
  - EU-Grid (DataGrid)
  - PPDG
  - GriPhyN
- Time scale for CMS Grid services
  - Fall 2000 (usable prototype with a minimum replication functionality)

Requirement Specification

- Distributed Job Execution and Data Handling
  - Transparency
  - Performance
  - Security
  - Fault Tolerance
  - Automation

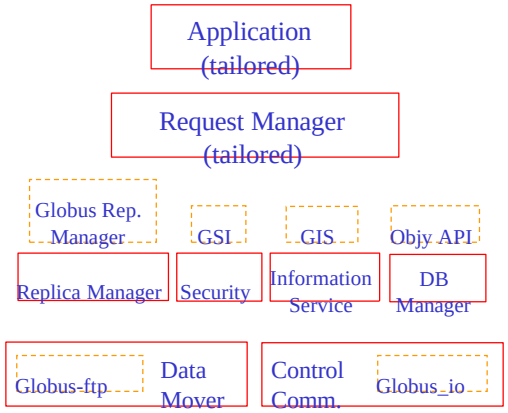


Distributed Data Management

- Functionality required
  - Store and manage the data produced locally.
  - Replication
    - Replicate both the data and meta-data produced to sites on an on-demand basis or streaming for special scenarios.
    - Handle the replica catalogues.
    - Perform replica selection and synchronization tasks.
  - Provide secure access to remote users.

...Distributed Data Management

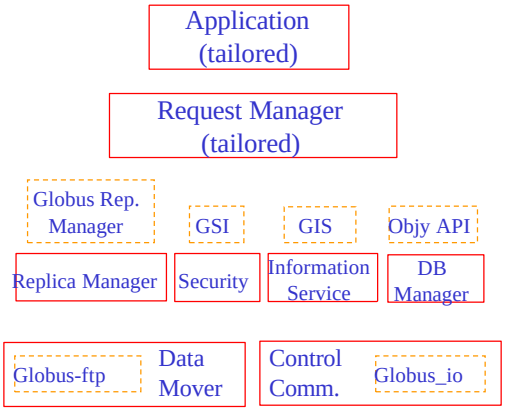
- Architecture
  - Communication
    - Control messages and file transfers.
  - Data Mover
    - Logging incoming and outgoing files.
    - Error checks
    - Dealing with partial transfers etc
  - Security
    - Authentication and authorization on organization level.



Layered Architecture for Distributed Data Management

...Distributed Data Management

- ...Architecture
  - Replica Manager
    - Handling replica catalogues
    - Replica selection and synchronization
  - Information Service
    - Publish information about the data and network resources of every site.



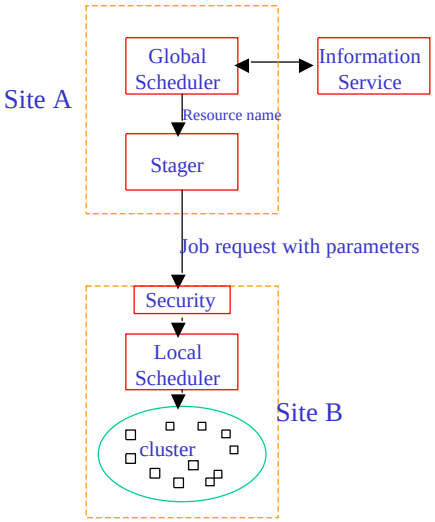
Layered Architecture for Distributed Data Management

Distributed Computation

- Functionality required
  - Select a remote machine/cluster to run the job, depending on
    - Hardware and software specs.
    - CPU Load and jobs already in queue.
    - Available disk space.
    - Network status
  - Stage the right parameters and data cards for the executable on the remote machine/cluster.
  - Schedule the job locally based on the loads on different nodes in the cluster.

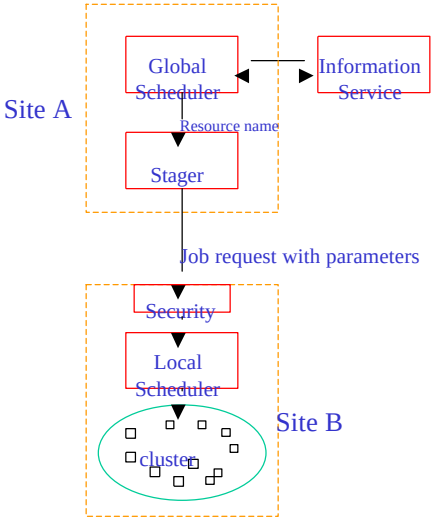
...Distributed Computation

- Architecture
  - Global Scheduler
    - Use the information infrastructure and built-in algorithms to find out where to submit jobs.
  - Stager
    - Stage the right parameters and data cards to the selected remote machine.



...Distributed Computation

- ...Architecture
  - Security
    - Authenticate and authorize the remote user for job submission.
  - Local Scheduler
    - Dividing work between local nodes in the cluster depending on their loads.



Common Components

- Security
  - Implemented using GSI
- Information Service
  - In Progress using GIS
- Network Monitoring
  - To be done using NWS

Deliverables and Milestones

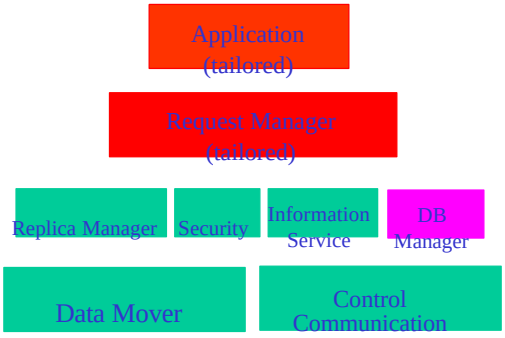
- First Prototype
  - August 2000
  - Basic infrastructure
    - Security — Done
    - Control Messaging — Done
    - File transfers — Done
    - Request Manager — Done
    - Replica Manager
      - Replica synchronisation — In Progress (OBJY. Catalog)
    - Database Manager — In Progress
    - Multithreaded server — Done

...Deliverables and Milestones

- Second Prototype
  - Jan 2001
    - Network monitoring (using NWS)
    - Replica Manager (Using Globus replica catalogue)
    - Information Services (GIIS)
    - Basic services for distributed computing.
- Final Prototype
  - Fall 2001
  - Distributed Data Management and Distributed Computing in place.
- Testing Phase

Meat for other Projects

- Can be used as a complete tool for secure file replication
- Different sub-systems can be used independently
- Benefit from the strategies evaluated and stress testing tools like Globus and NWS etc



- Can be used independently in any system
- Can be used in some specific systems
- Dependencies on other modules

Conclusions

- Design is flexible enough to incorporate extensions or modifications
- The experiences gained, strategies evaluated, test results and deliverables can be used in other Grid projects
- Globus
  - Fulfills most of the middle-ware requirements
  - Accepted middle-ware for Grid services at CMS