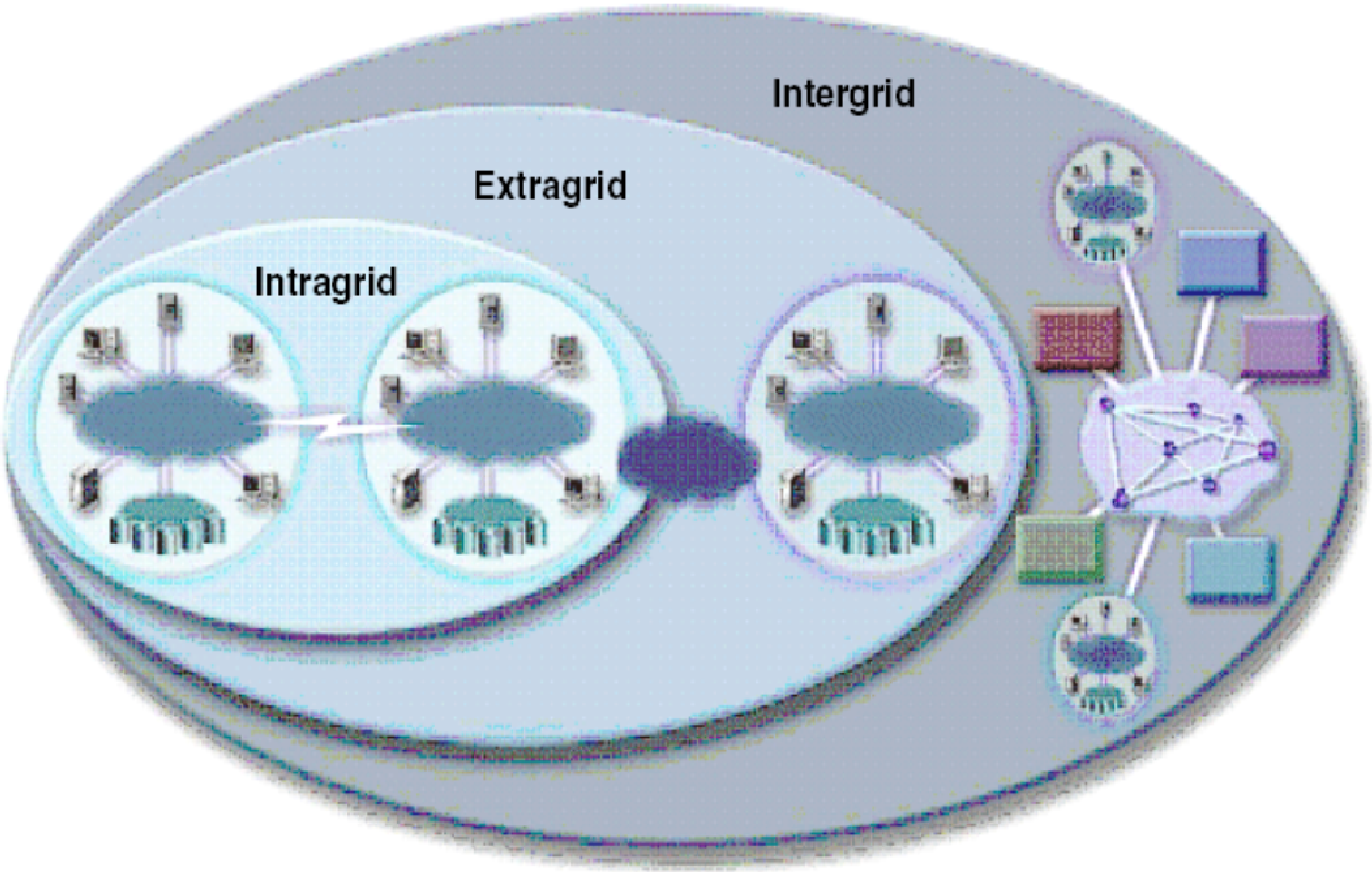
Grid Architectural Models

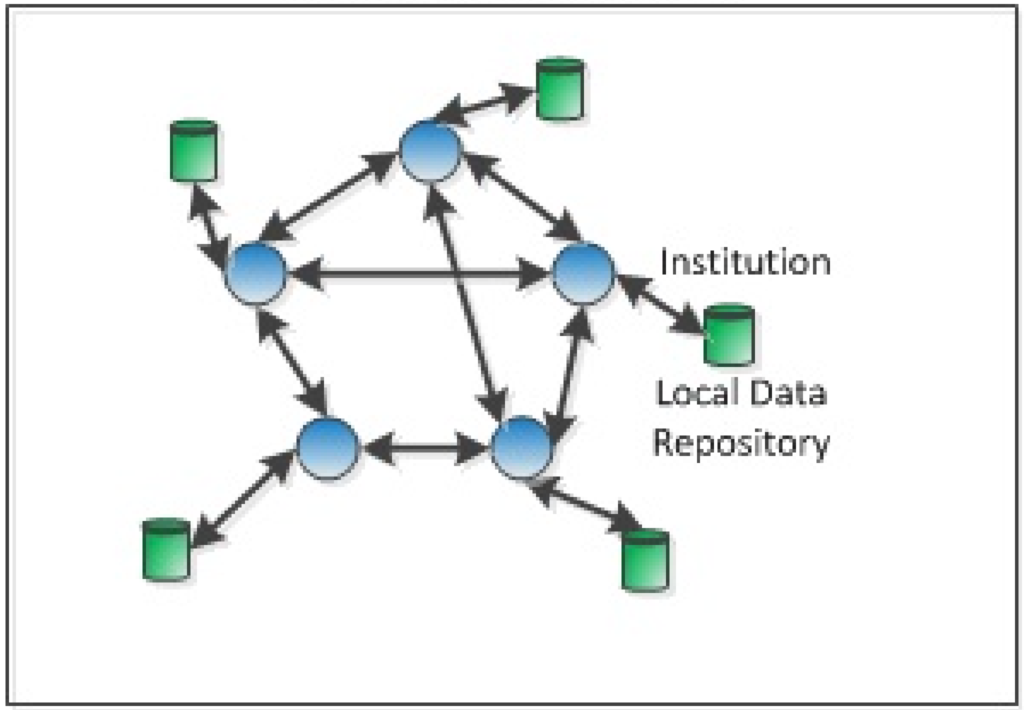
- **Computational Grids**
  - A computational Grid aggregates the processing power from a distributed collection of systems
  - This type of Grid is primarily composed of low powered computers with minimal application logic awareness and minimal storage capacity
  - The primary benefits of computational Grids are a reduced Total Cost of Ownership (TCO), and shorter deployment life cycles
  - Example: SETI@Home
- **DataGrids**
  - Data Grids focus on providing secure access to distributed, heterogeneous pools of data
  - Through collaboration, data Grids can also include a new concept such as a federated database
  - Data Grids also harness data, storage, and network resources located in distinct administrative domains, respect local and global policies governing how data can be used, schedule resources efficiently, again subject to local and global constraints, and provide high speed and reliable access to data
  - Example: CERN's LHC Data Grid

Computational Grid topologies

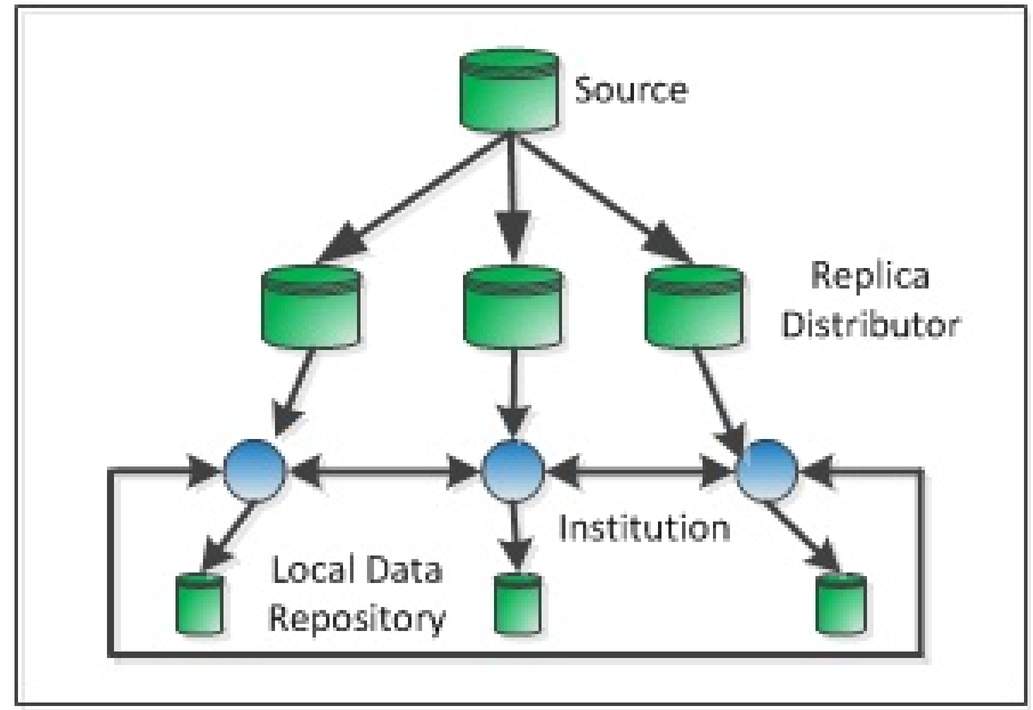


Data Grid topologies

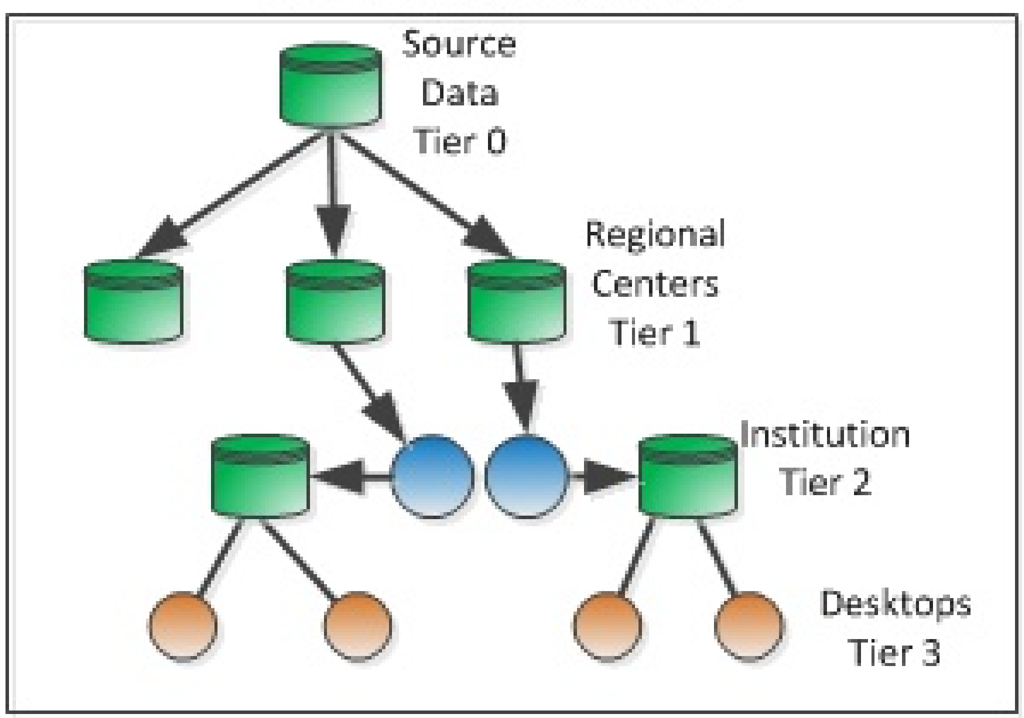
Federation Topology



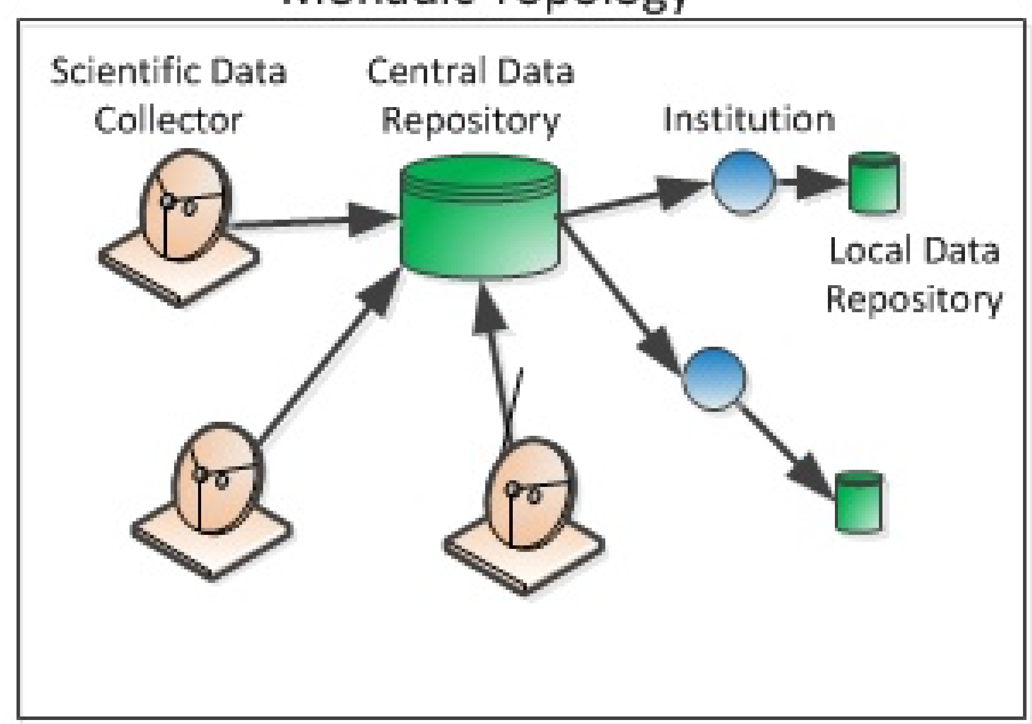
Hybrid Topology



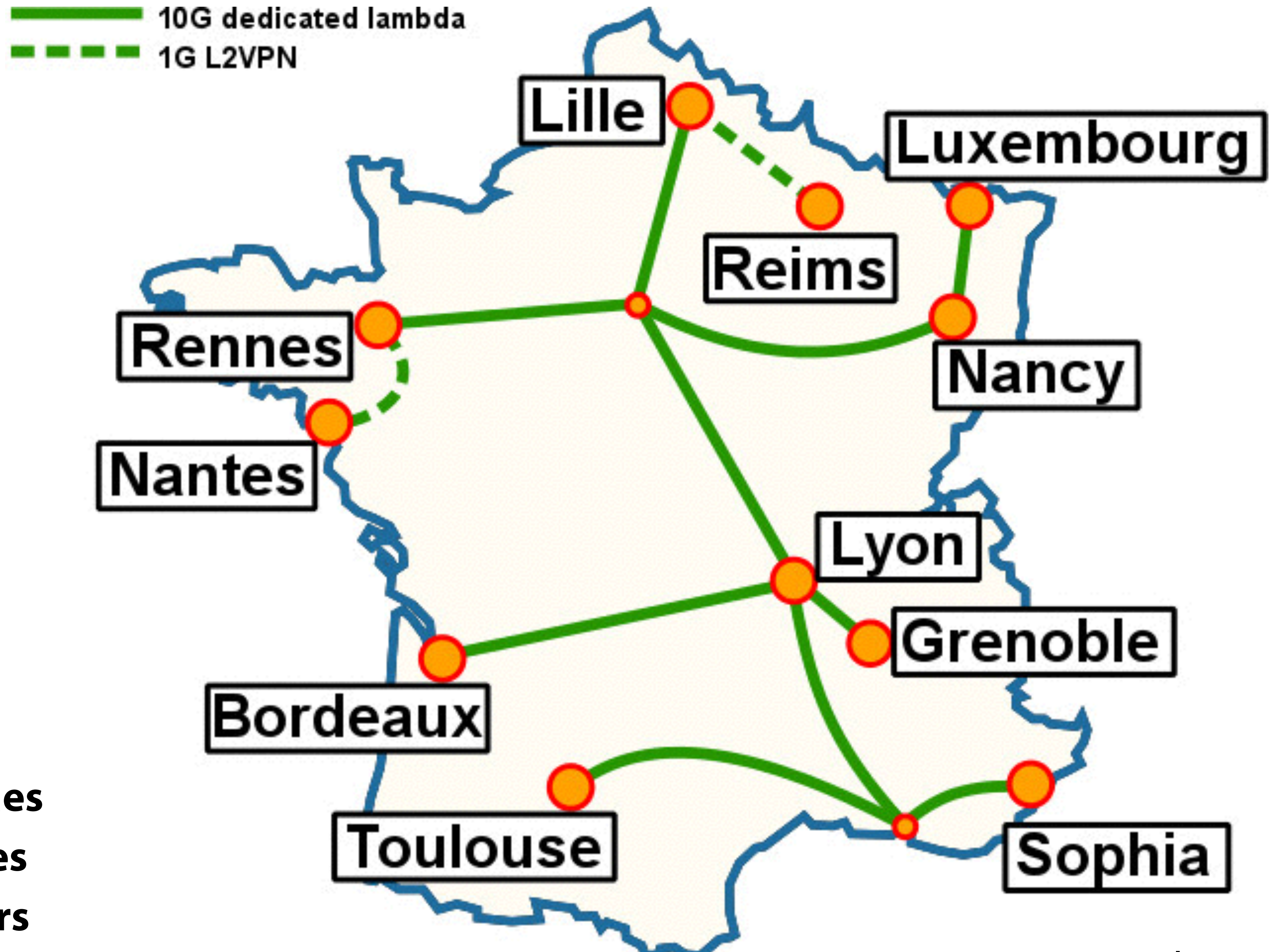
Hierarchical Topology



Monadic Topology



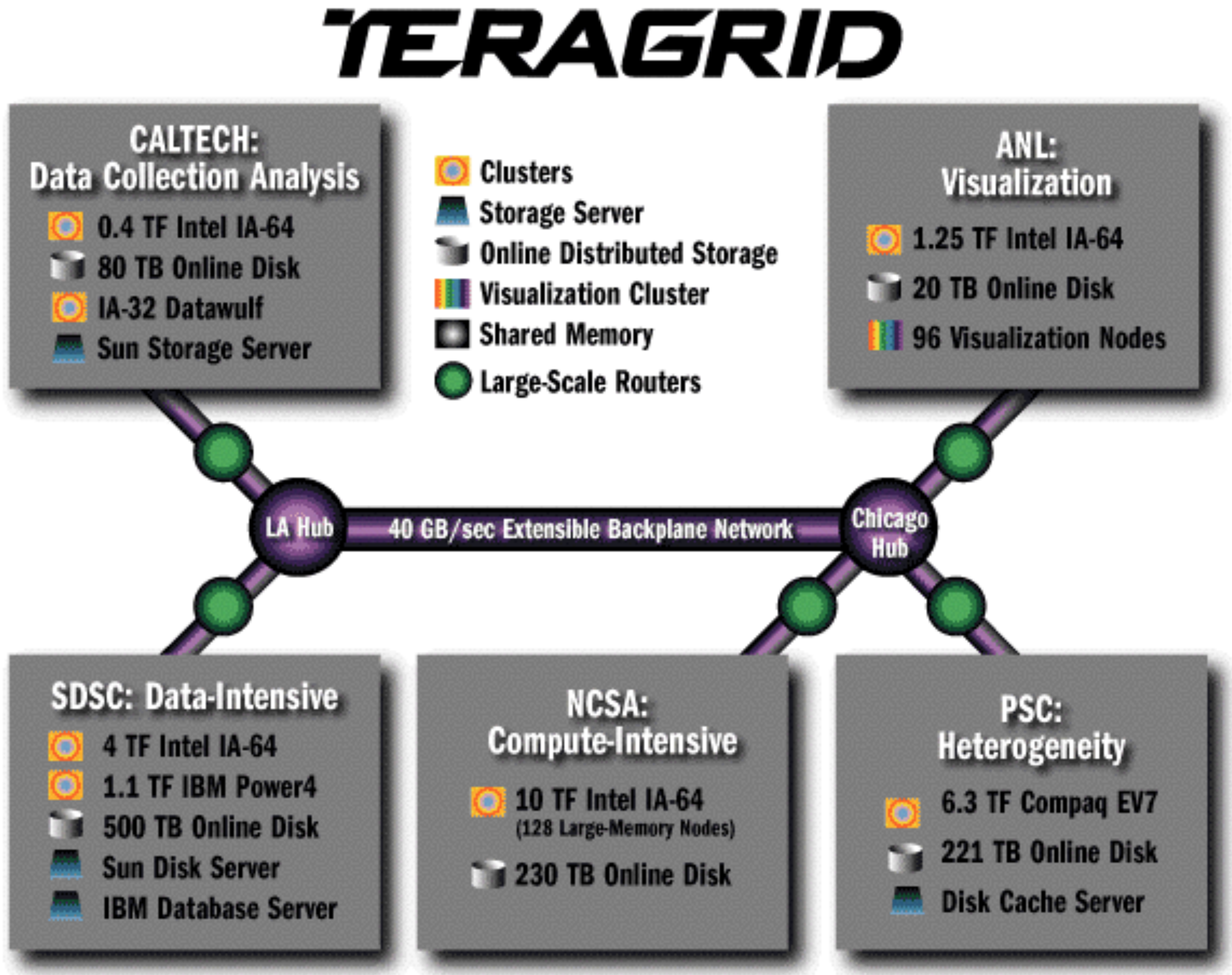
Grid 5000 Example



- **1000 nodes**
- **8000 cores**
- **500+ users**

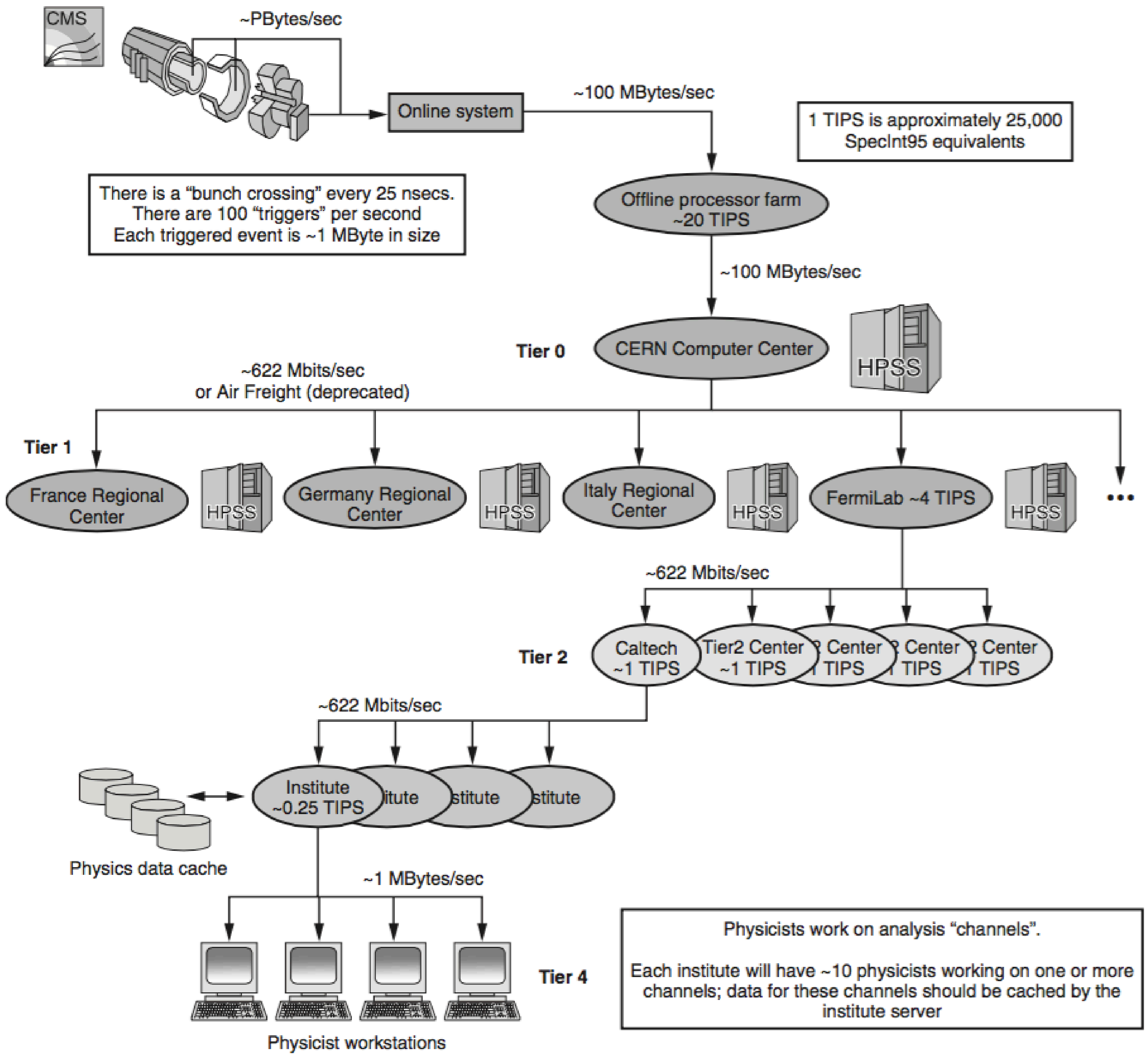
http://grid5000.fr

TeraGrid Example



xsede.org

European Data Grid Example



http://eu-datagrid.web.cern.ch/eu-datagrid/

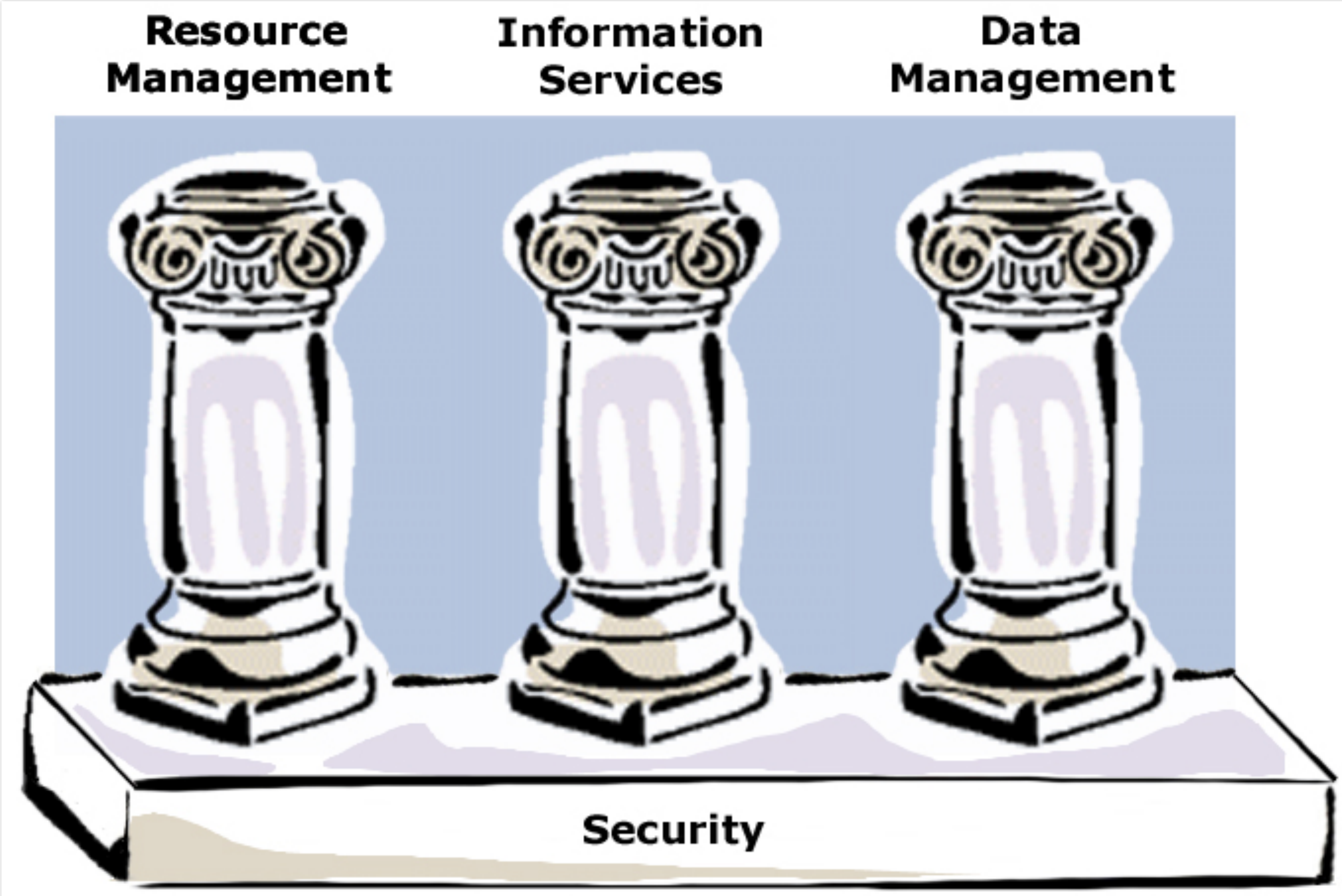
Globus Toolkit

- An example Grid middleware

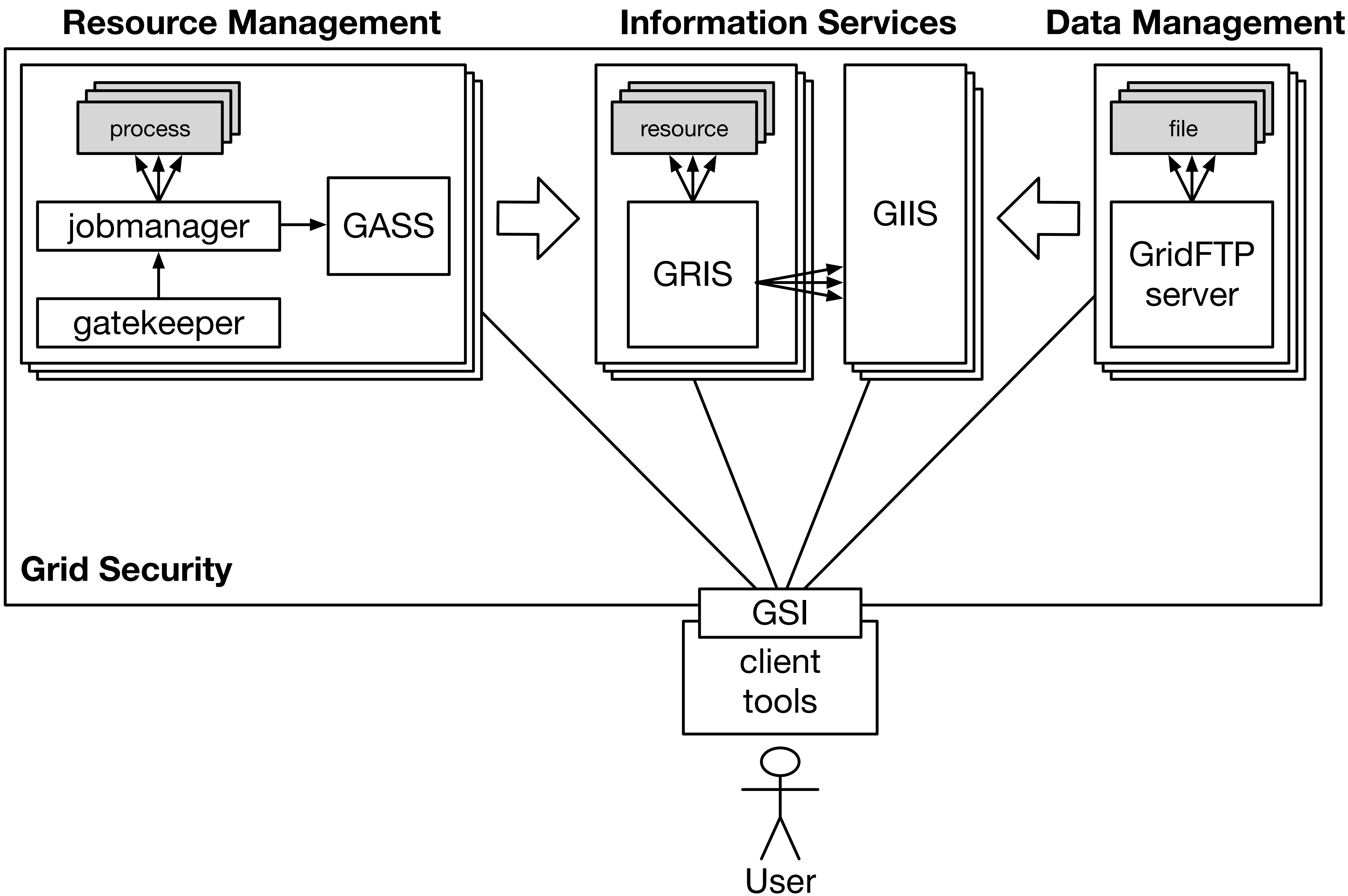
http://www.globus.org/toolkit/

- A software toolkit addressing key technical problems in the development of Grid enabled tools, services, and applications
  - Offer a modular "bag of technologies"
  - Enable incremental development of Grid-enabled tools and applications
  - Implement standard Grid protocols and APIs (the "core" of the hourglass)
  - Is available under liberal open source license
- Now is evolving to Cloud middleware

Key Protocols



GT2 Protocols and Services

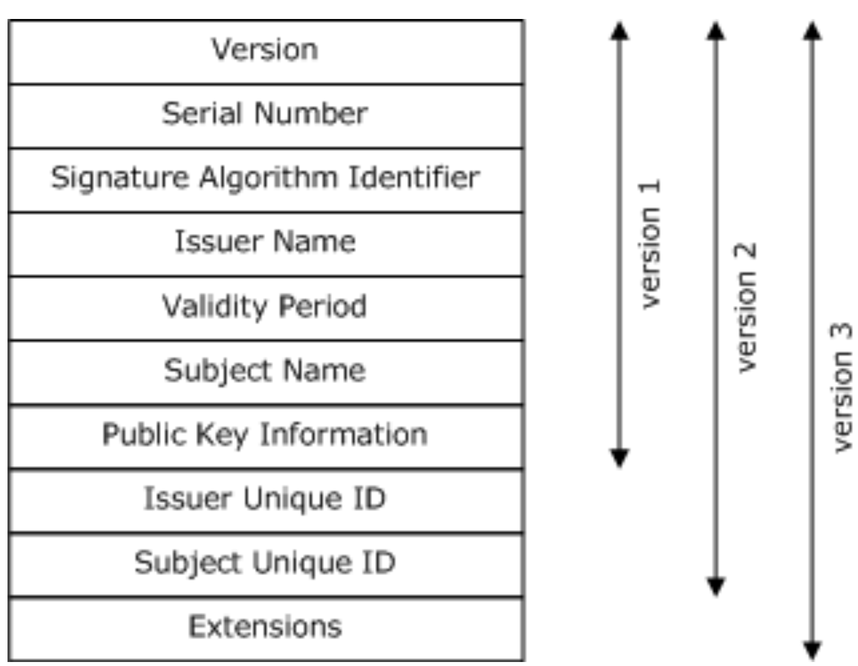


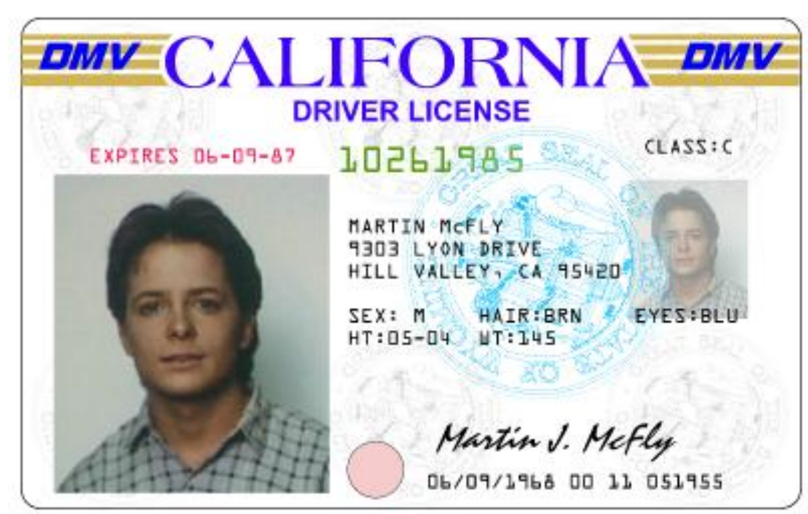
Grid Security

- Resources being used may be valuable & the problems being solved sensitive
- Resources are often located in distinct administrative domains
  - Each resource has own policies & procedures
- Set of resources used by a single computation may be large, dynamic, and unpredictable
  - Not just client/server, requires delegation
- It must be broadly available & applicable
- Standard, well-tested, well-understood protocols; integrated with wide variety of tools

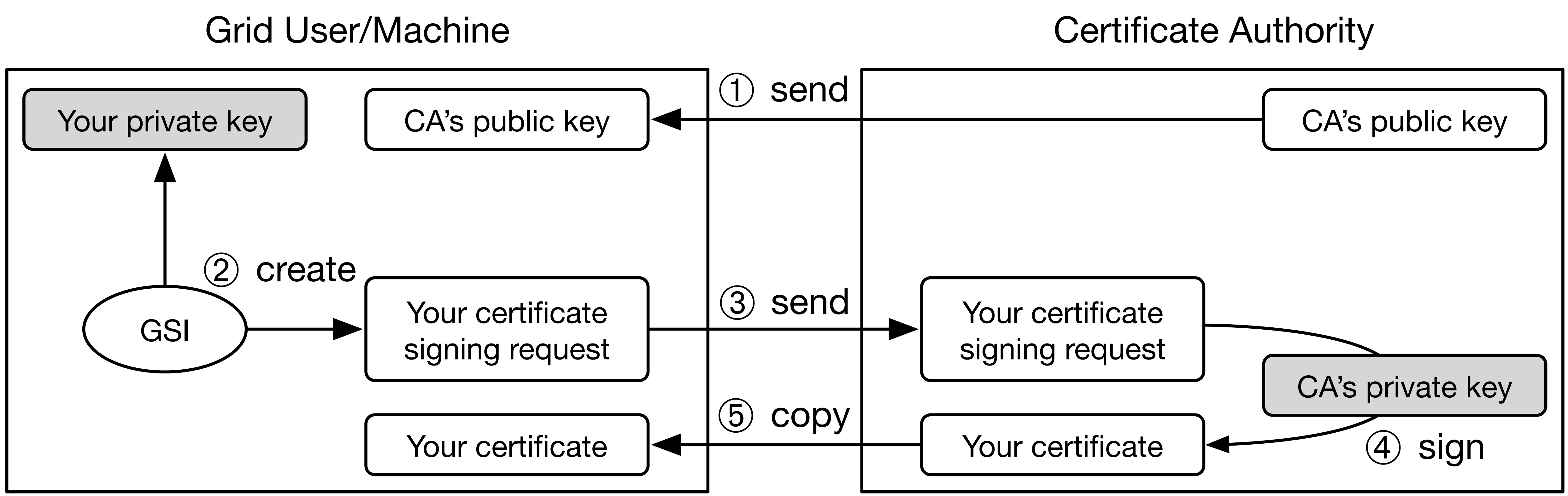
Public Key Infrastructure

- PKI allows you to know that a given public key belongs to a given user
- PKI builds upon asymmetric encryption:
  - Each entity has two keys: public and private
  - Data encrypted with one key can only be decrypted with the other
  - The private key is known only to the owner
- The public key is given to the world encapsulated in a X.509 certificate
  - Similar to passport or driver's license

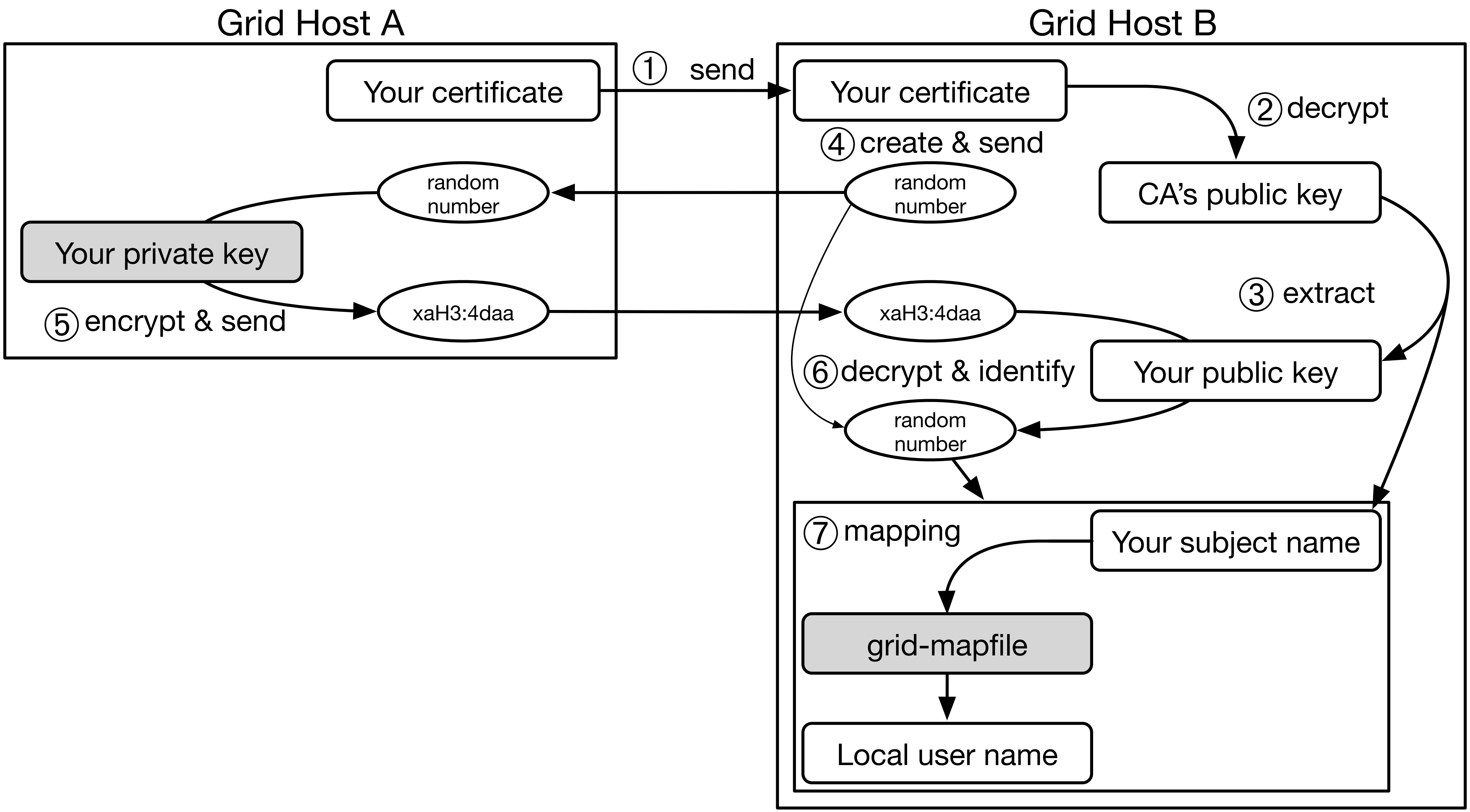




Certificate Authority



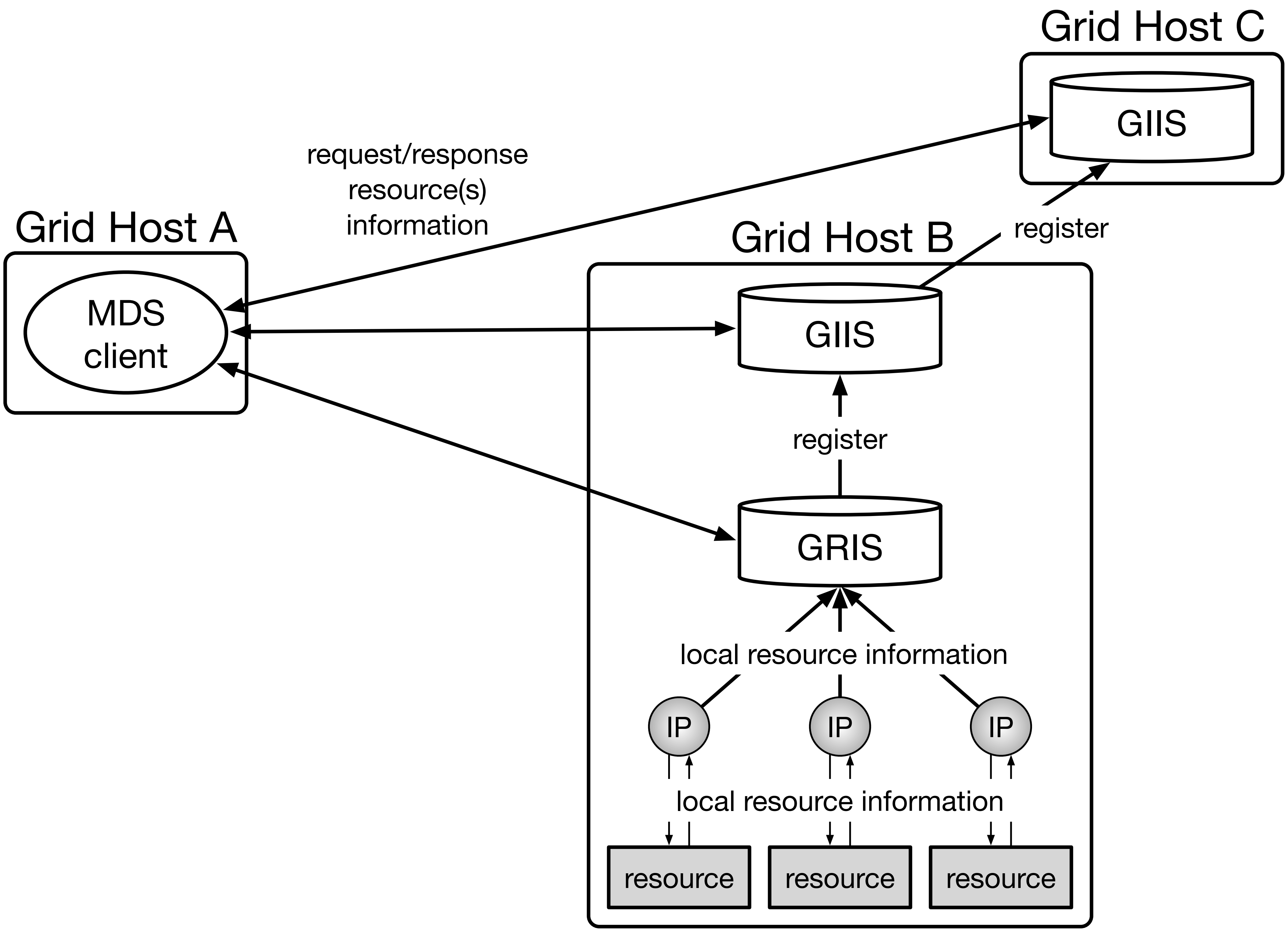
Authentication & Authorization



Grid Information Service

- Provide access to static and dynamic information regarding system components
- A basis for configuration and adaptation in heterogeneous, dynamic environments
- Resource Description Services
  - Supplies information about a specific resource
- Aggregate Directory Services
  - Supplies collection of information which was gathered from multiple resource description services
  - Customized naming and indexing

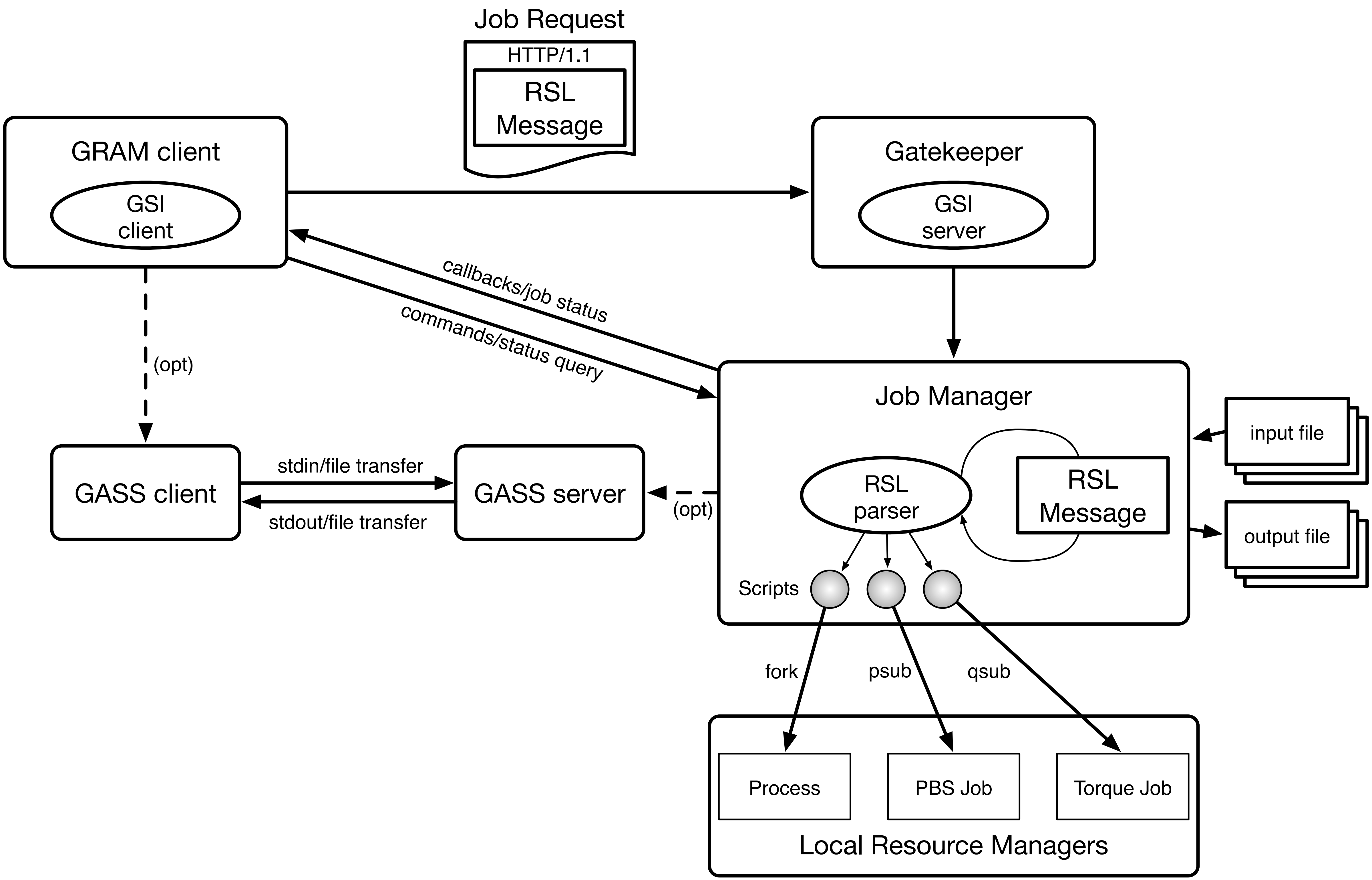
MDS Protocols and Services



Grid Resource Management

- Grid Resource Management System consists of :
  - Local resource management system (Resource Layer)
    - Basic resource management unit
    - Provide a standard interface for using remote resources
    - Grid Resource Allocation Manager (GRAM)
  - Global resource management system (Collective Layer)
    - Coordinate all Local resource management system within multiple or distributed Virtual Organizations (VOs)
    - Provide high-level functionalities to efficiently use all of resources
      - Job Submission
      - Resource Discovery and Selection
      - Scheduling
      - Co-allocation
      - Job Monitoring, etc.
    - e.g. Meta-scheduler, Resource Broker, etc.

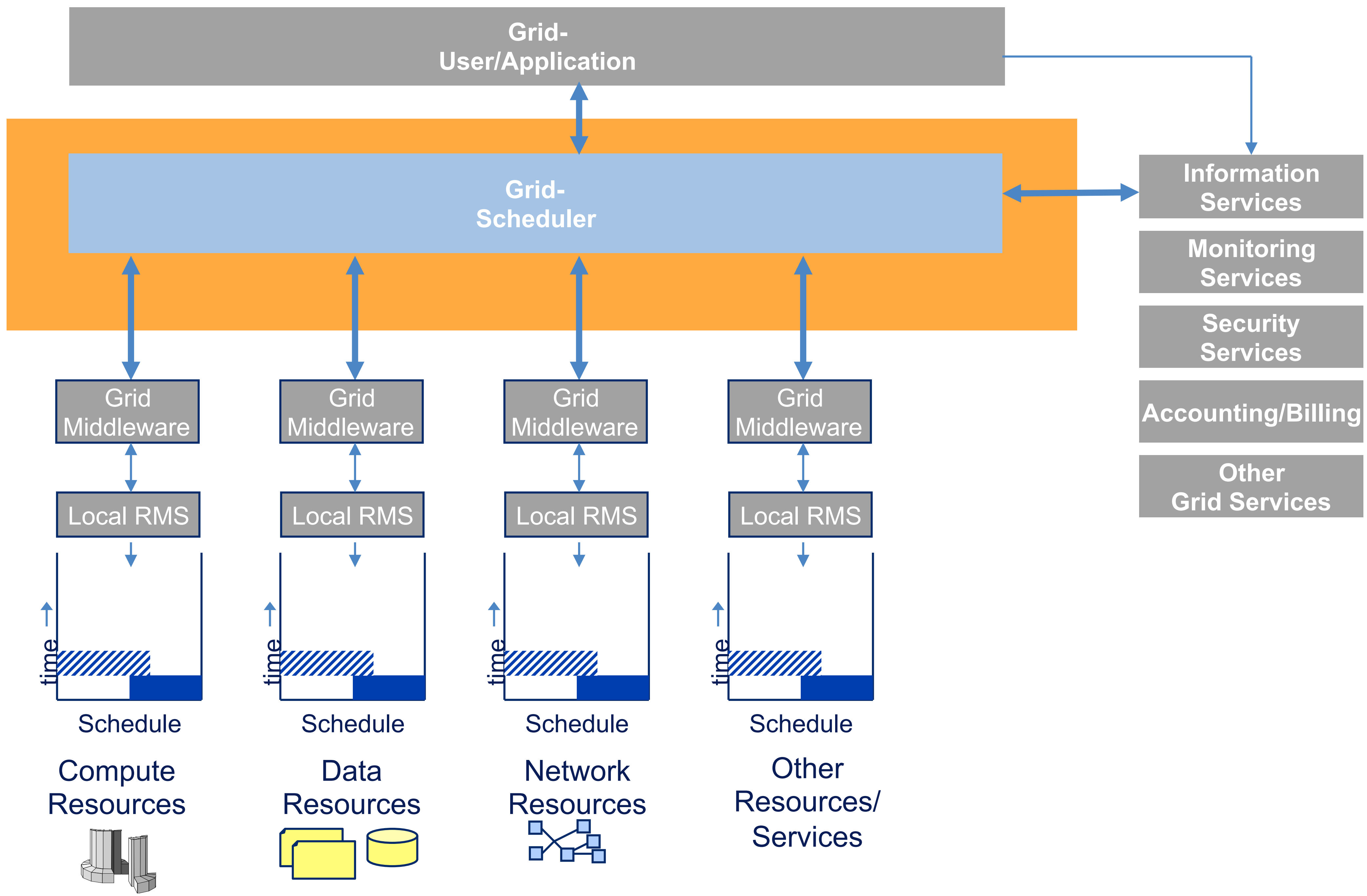
GRAM Protocols and Services



Grid Scheduling Levels

- Resource-level scheduler
  - low-level scheduler, local scheduler, local resource manager
  - scheduler close to the resource, controlling a supercomputer, cluster, or network of workstations, on the same local area network
  - Examples: Open PBS, PBS Pro, LSF, SGE
- Enterprise-level scheduler
  - Scheduling across multiple local schedulers belonging to the same organization
  - Examples: PBS Pro peer scheduling, LSF Multicluster
- Grid-level scheduler
  - Also known as super-scheduler, broker, community scheduler
  - Discovers resources that can meet a job's requirements
  - Schedules across lower level schedulers

Grid Scheduler



A Grid scheduler allows the user to specify the required resources and environment of the job without having to indicate the exact location of the resources

Grid Scheduler Activities

Phase One-Resource Discovery

1. Authorization Filtering
2. Application Definition
3. Min. Requirement Filtering

Phase Two - System Selection

4. Information Gathering
5. System Selection

Phase Three- Job Execution

6. Advance Reservation
7. Job Submission
8. Preparation Tasks
9. Monitoring Progress
10 Job Completion
11. Clean-up Tasks

Issues

- Resources may dynamically join and leave the Grid
- Not all currently unused resources are available to grid jobs
- Resource owner local policies can restrict maximum number of grid jobs allowed
- Hard to predict how long jobs will wait in a queue
- User information not accurate as mentioned before
- New jobs arrive that may surpass current queue entries due to higher priority
- Local jobs have typically higher priority than Grid jobs
- Limited information about the local schedulers is available (privacy)
- Data Management
- Network Management

Example of Grid Scheduling Process

Scheduling Service:

1. receives job description
2. queries Information Service for static resource information
3. prioritizes and pre-selects resources
4. queries for dynamic information about resource availability
5. queries Data and Network Management Services
6. generates schedule for job
7. reserves allocation if possible
   otherwise selects another allocation
8. delegates job monitoring to Job Supervisor

Job Supervisor/Network and Data Management:
service, monitor and initiate allocation

Example:

40 resources of requested type are found.

12 resources are selected.

8 resources are available.

Network and data dependencies are detected.

Utility function is evaluated.

6th tried allocation is confirmed.

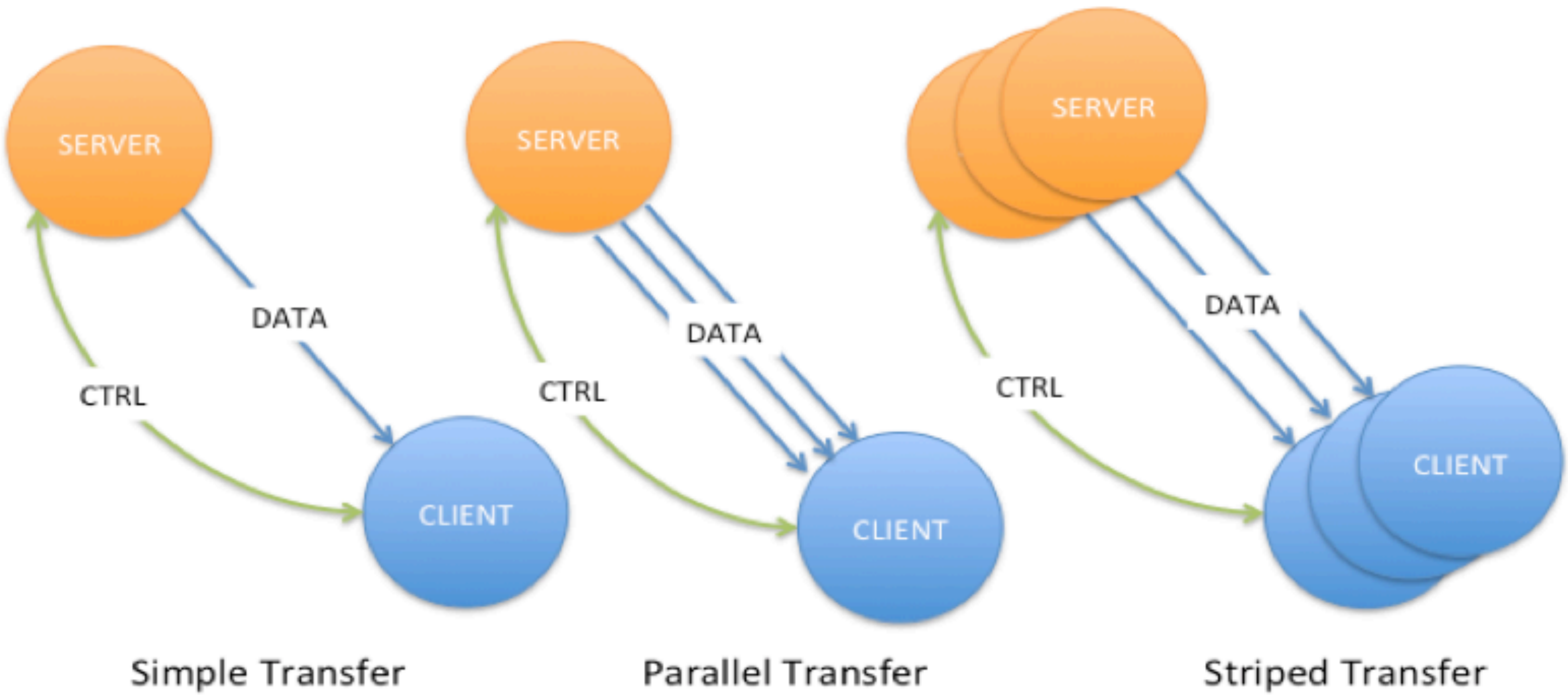
Data/network provided and

job is started

Grid Data Management

- Data access and transfer
  - **GASS**: Simple multi-protocol tool to transfer 'normal' files; integrated in GRAM
  - **GridFTP**: Reliable and high-performance file transfer protocol for 'big' files in computer networks
- Replica Management
  - **Replica Catalog**: Service to keep updated information on sets of replicated data
  - **Replica Management**: Service to create and manage sets of replicated data

GridFTP


SERVER
DATA
CTRL
CLIENT
SERVER
DATA
CTRL
CLIENT
SERVER
DATA
CTRL
CLIENT
Simple Transfer
Parallel Transfer
Striped Transfer