UVA HPC & BIG DATA COURSE

Grid Computing

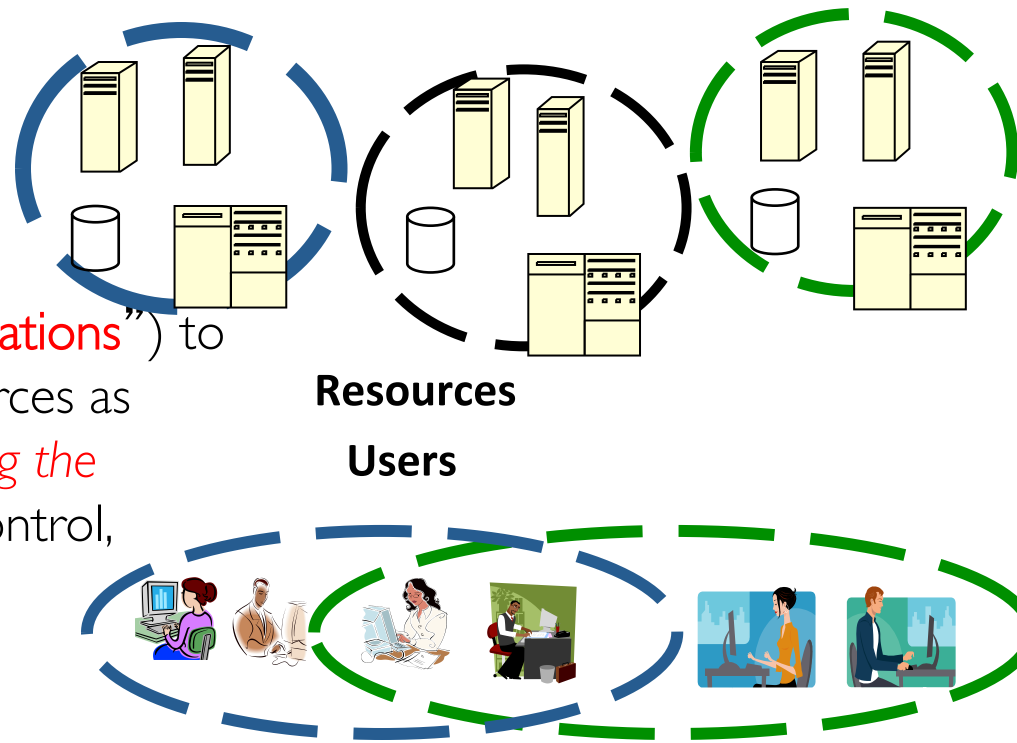
Adam Belloum

outline

- Grid computing: Approach and vision
- Potential Application for Grid computing:
- Grid-middleware
- Example of Services provided by Grid Systems

Grand Vision of Grid systems

- Flexible, secure, coordinated resource sharing among dynamic collections of individuals, institutions, and resources



- Enable communities ("Virtual Organizations") to share geographically distributed resources as they pursue common goals -- *assuming the absence* of : central location, central control, existing trust relationships.

From the Grid tutorials available at : http:.//www.globus.org

Grand Vision of Grid Computing

- Grid computing is the use of hundreds, or thousands of **geographically** and **organizationally** disperse and diverse resources to solve:

  ➔ problems that require more computing power than is available from a single machine or from a local area distributed system

outline

- Grid computing: Approach and vision
- Potential Application for Grid computing:
- Grid-middleware
- Example of Services provided by Grid Systems

Potential Grid Applications

- An application which requires the grid solution is likely distributed (Distributed Computing) and fit in one of the following paradigms:
  - High throughput Computing
  - High performance Computing

Grid computing will be mainly needed for large-scale, high-performance computing.

Distributed Computing

- Distributed computing is a programming model in which processing occurs in many geographically distributed places.
  - Processing can occur wherever it makes the most sense, whether that is on a server, Web site, personal computer, etc.

- Distributed computing and grid computing either
  - overlap or distributed computing is a subset of grid computing

From "The Anatomy of the Grid: Enabling Scalable Virtual Organizations" Foster et al

High Throughput Computing

- HTC employs large amounts of computing power for very lengthy periods
  - HTC is needed for doing sensitivity analyses, parametric studies or simulations to establish statistical confidence.
- The features of HTC are
  - Availability of computing power for a long period of time
  - Efficient fault tolerance mechanism
- The key to HTC in grids
  - Efficiently harness the use of all available resources across organizations.

High Performance Computing

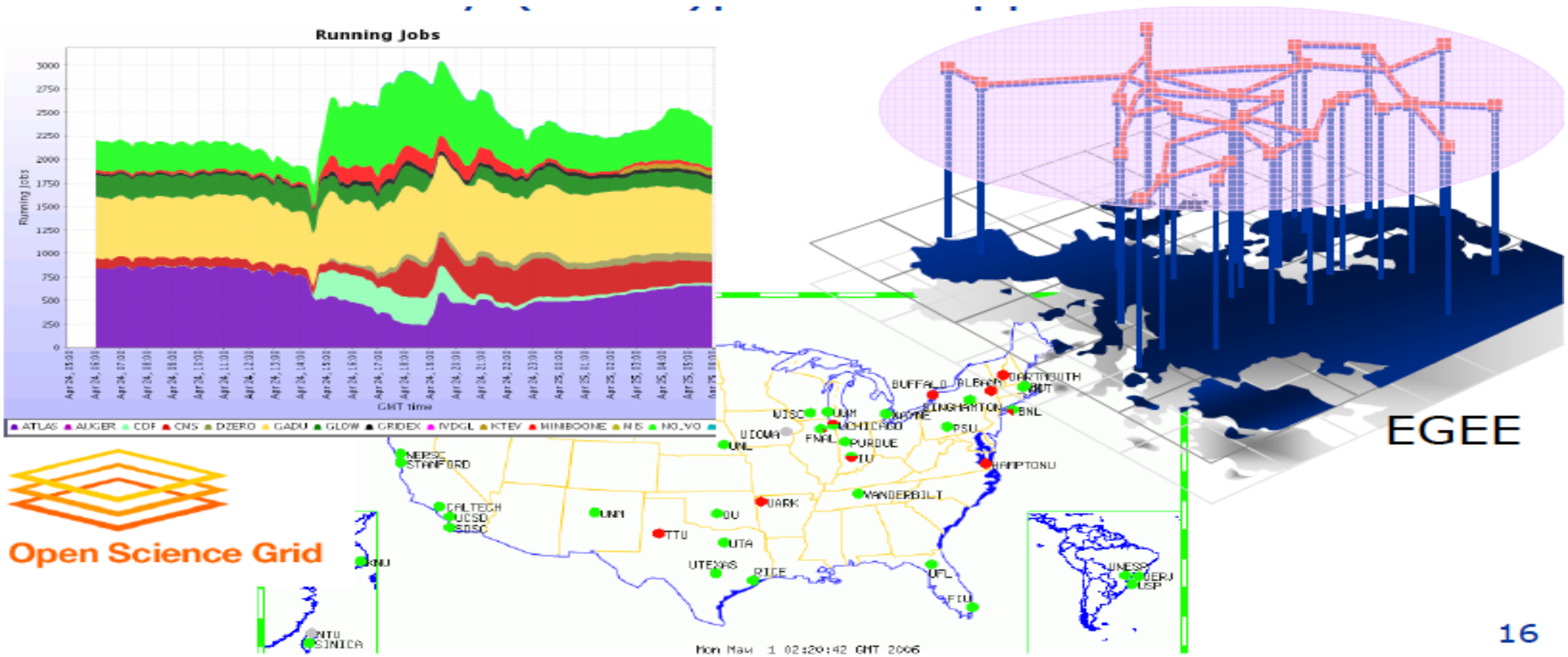
- HPC brings enormous amounts of computing power to bear over relatively short periods of time.
  - HPC is needed for decision-support or applications under sharp time-constraint, such as weather modeling
- HPC applications are:
  - Large in scale and complex in structure.
  - Real time requirements.
  - Ultimately must run on more than one type of HPC system.

HPC/HTC requirements

- HPC/HTC requires a balance of computation and communication among all resources involved.
  - Managing computation,
  - communication,
  - data locality

First Generation of Grids: Batch computing

Focus on aggregation of many resources for massively (data-)parallel applications



Second Generation Grids: Service-Oriented Science

- Empower many more users by enabling
  - On-demand access to **services**
- Grids become an enabling technology for **service oriented science** (or business)
  - Grid infrastructures host **services**
  - Grid technologies used to build **services**

"Service-Oriented Science", *Science*, 2005

Second Generation Grids: Service-Oriented Science (Transient Service Instances)

- Web services address discovery & invocation of persistent services
  - Interface to persistent state of enterprise
- In Grids, must also support transient service instances, created/destroyed dynamically
  - Significant implications for how services are managed, named, discovered, and used

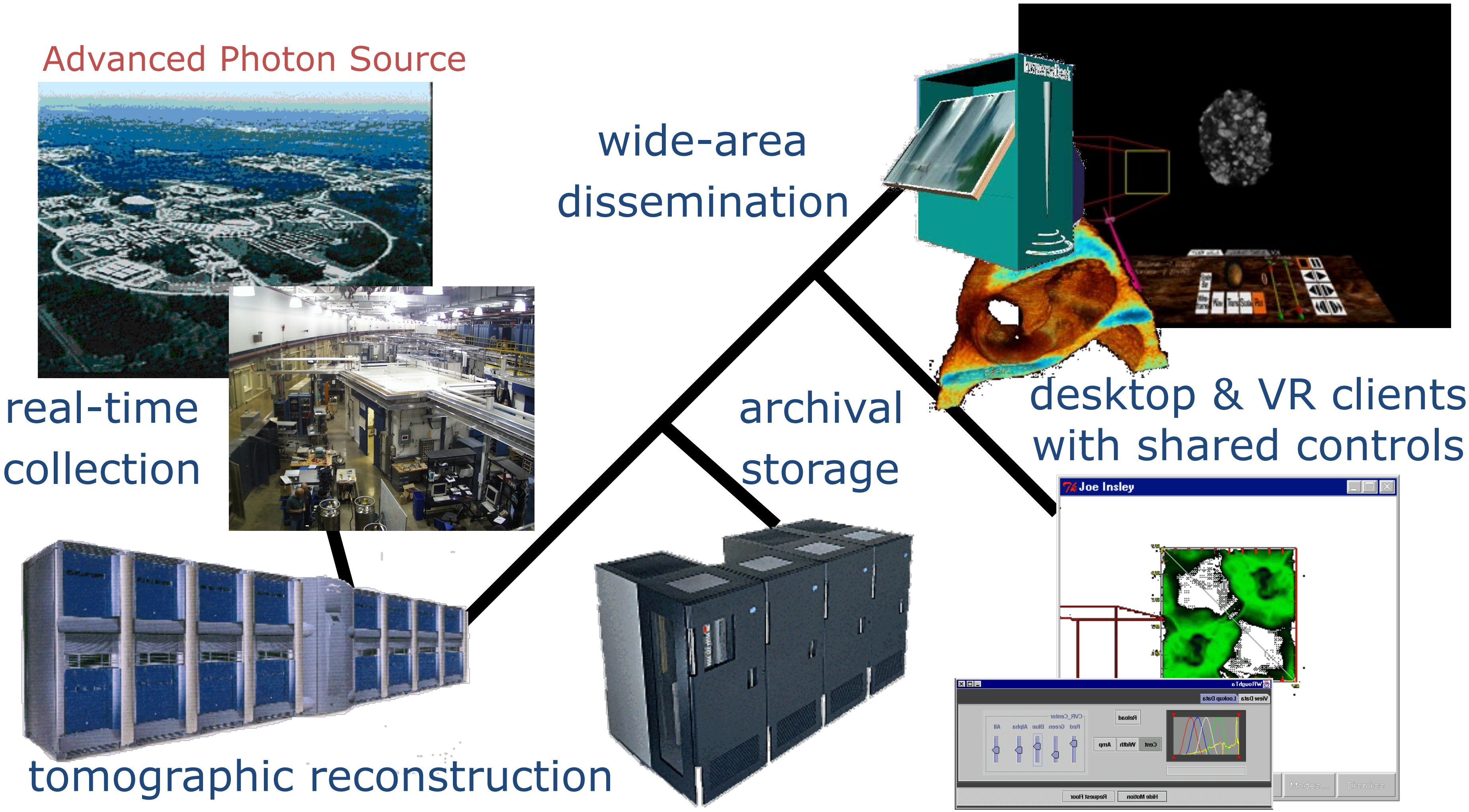
"Globus Toolkit Futures: An Open Grid Services Architecture" Ian Foster et al. Globus Tutorial, Argonne National Laboratory, January 29, 2002

eScience: Applications that needs the Grid

- "eScience promotes innovation in collaborative, computationally or data intensive research across all disciplines, throughout the research lifecycle''

- Nowadays Scientific Applications are
  - CPU intensive
  - Produce/process Huge sets of Data
  - Requires access to *geographically distributed* and *expensive* instruments

Online Access to Scientific Instruments

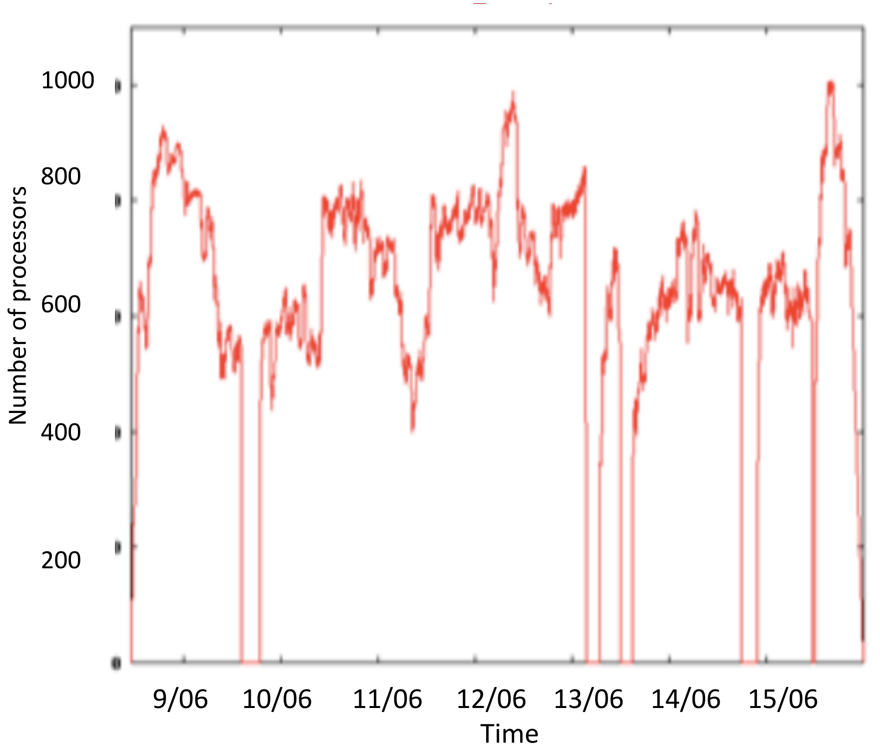


DOE X-ray grand challenge: ANL, USC/ISI, NIST, U.Chicago

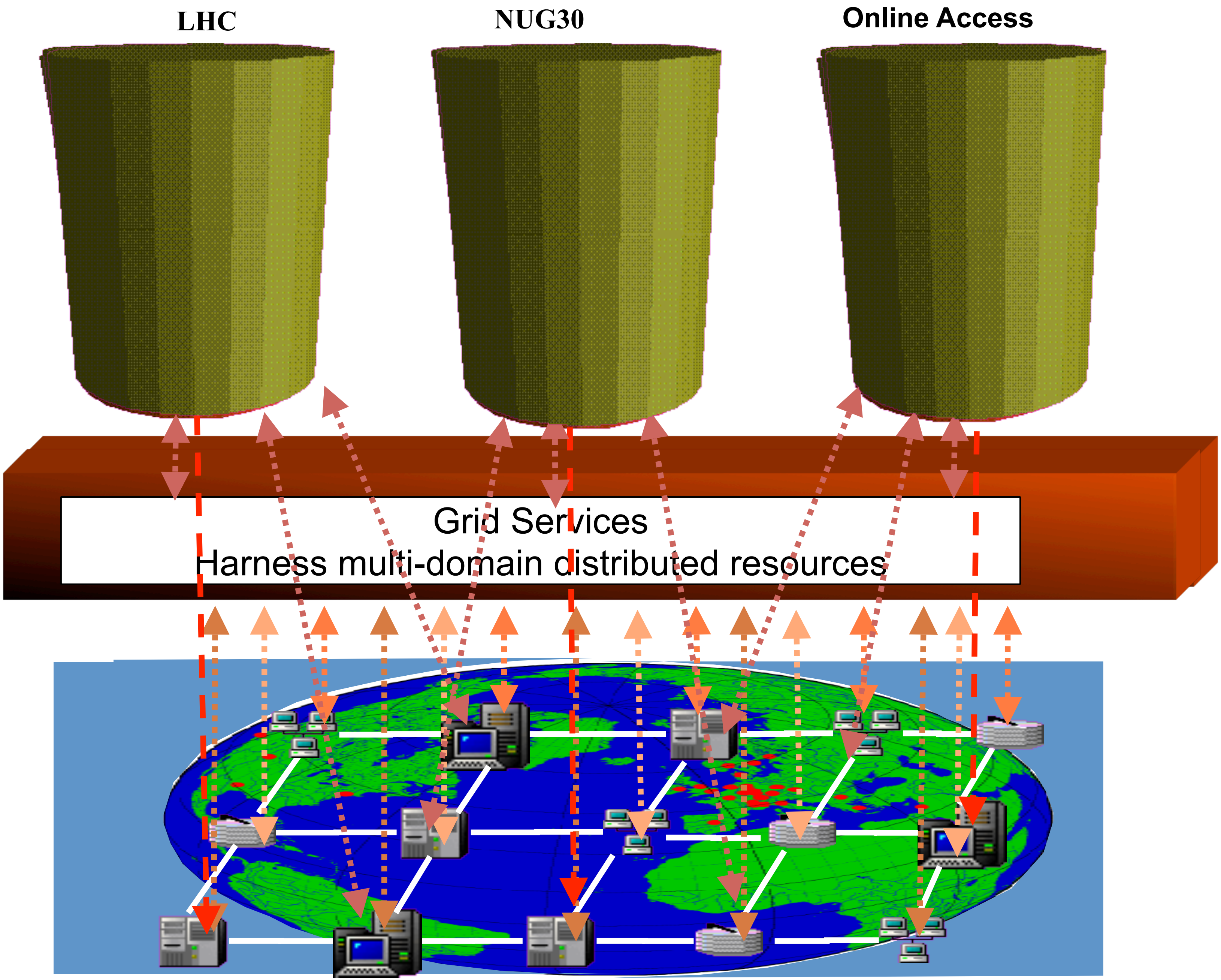
From the Grid tutorials available at : http:.//www.globus.org

CPU intensive Science: Optimization problem NUG30

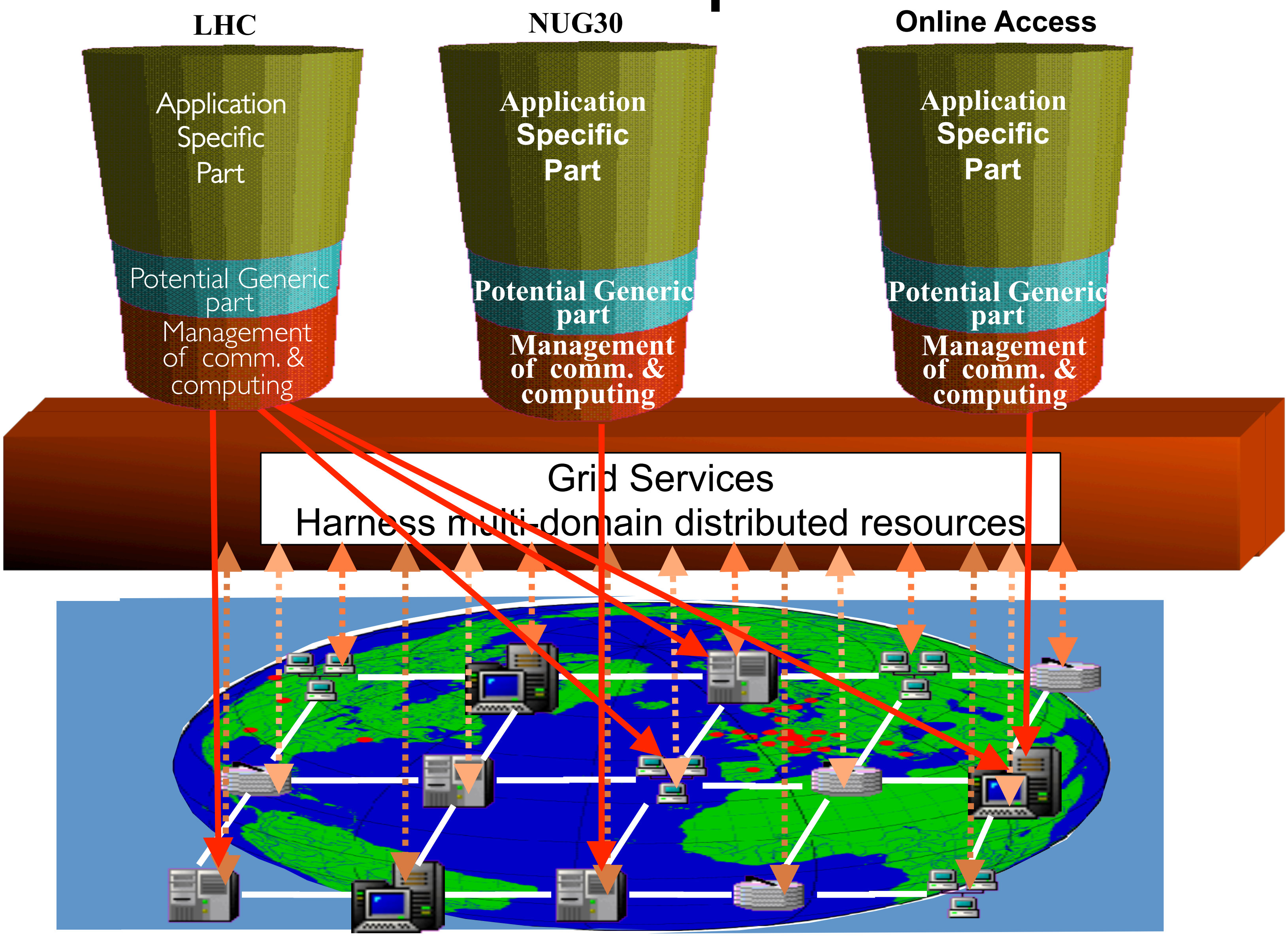
- The problem, a quadratic assignment problem (QAP) known as NUG30
  - given a set of n locations and n facilities, the goal is to assign each facility to a location.
  - There are **n!** possible assignments
- NUG30 proposed in 1968 as a test of computer capabilities, but remained unsolved because of its great complexity.



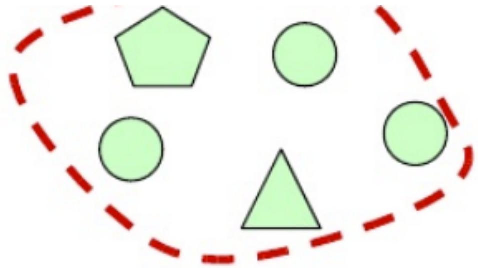
Nug30 Quadratic Assignment Problem Solved by 1,000 https://scout.wisc.edu/archives/r7125


LHC
NUG30
Online Access
Grid Services
Harness multi-domain distributed resources

To solve these problems?



The Grid paradigm

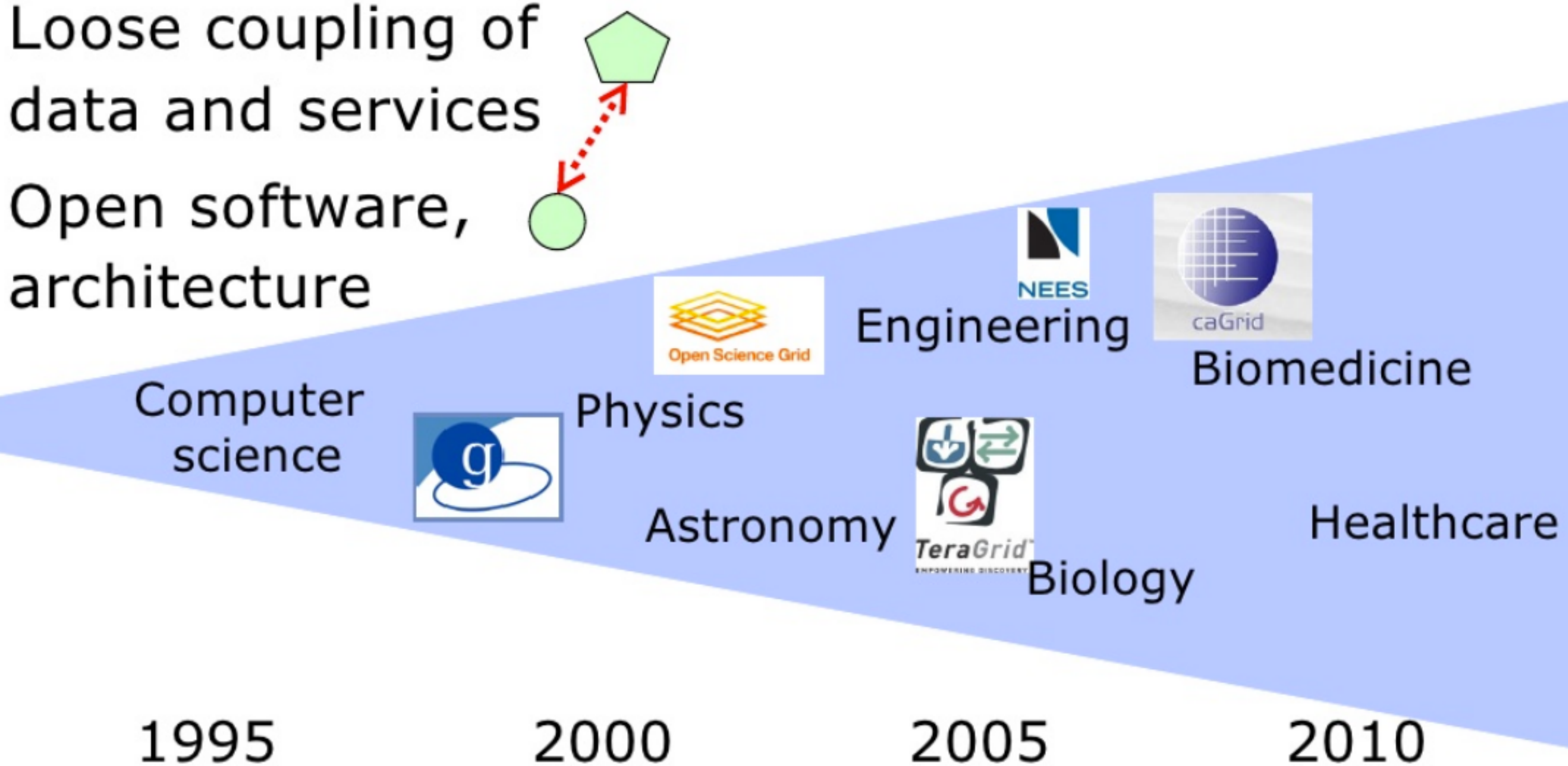


Principles and mechanisms for dynamic VOs

Leverage service oriented architecture (SOA)

Loose coupling of data and services

Open software, architecture



outline

- Grid computing: Approach and vision
- Potential Application for Grid computing:
- Grid-middleware
- Example of Services provided by Grid Systems

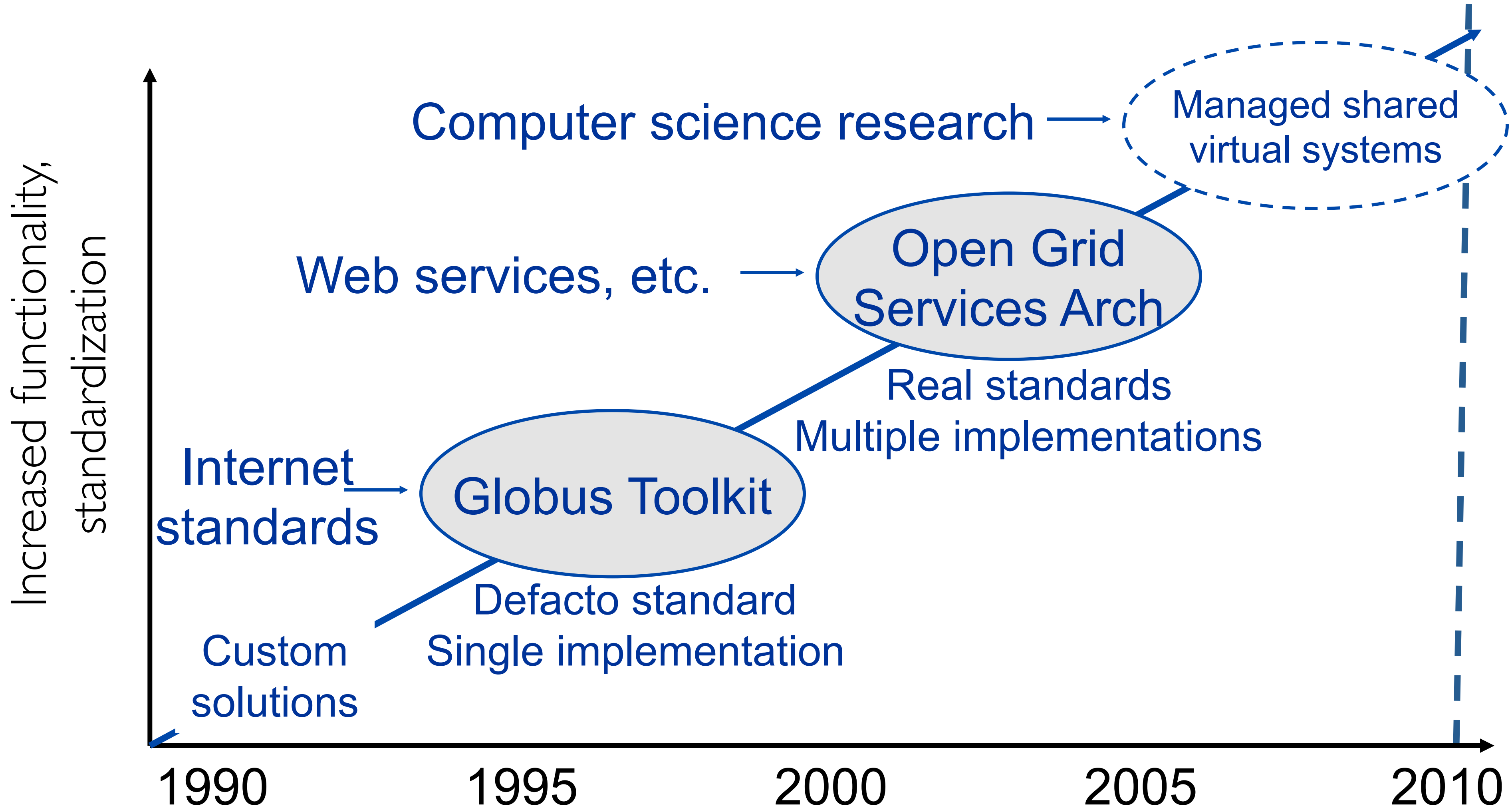
Grid Middleware Definition

- Software toolkit addressing key technical areas
  - Offer a modular "bag of technologies"
  - Enable incremental development of grid-enabled tools and applications
  - Define and standardize grid protocols and APIs
- Focus is on inter-domain issues, not clustering
  - Collaborative resource use spanning multiple organizations
  - Integrates cleanly with intra-domain services
  - Creates a collective service layer

Grid Middleware Definition

- Architecture identifies the fundamental system components, specifies purpose and function of these components, and indicates how these components interact with each other.
- Grid architecture is a protocol architecture, with protocols defining the basic mechanisms by which VO users and resources negotiate, establish, manage and exploit sharing relationships.
- Grid architecture is also a service standard-based open architecture that facilitates extensibility, interoperability, portability and code sharing.

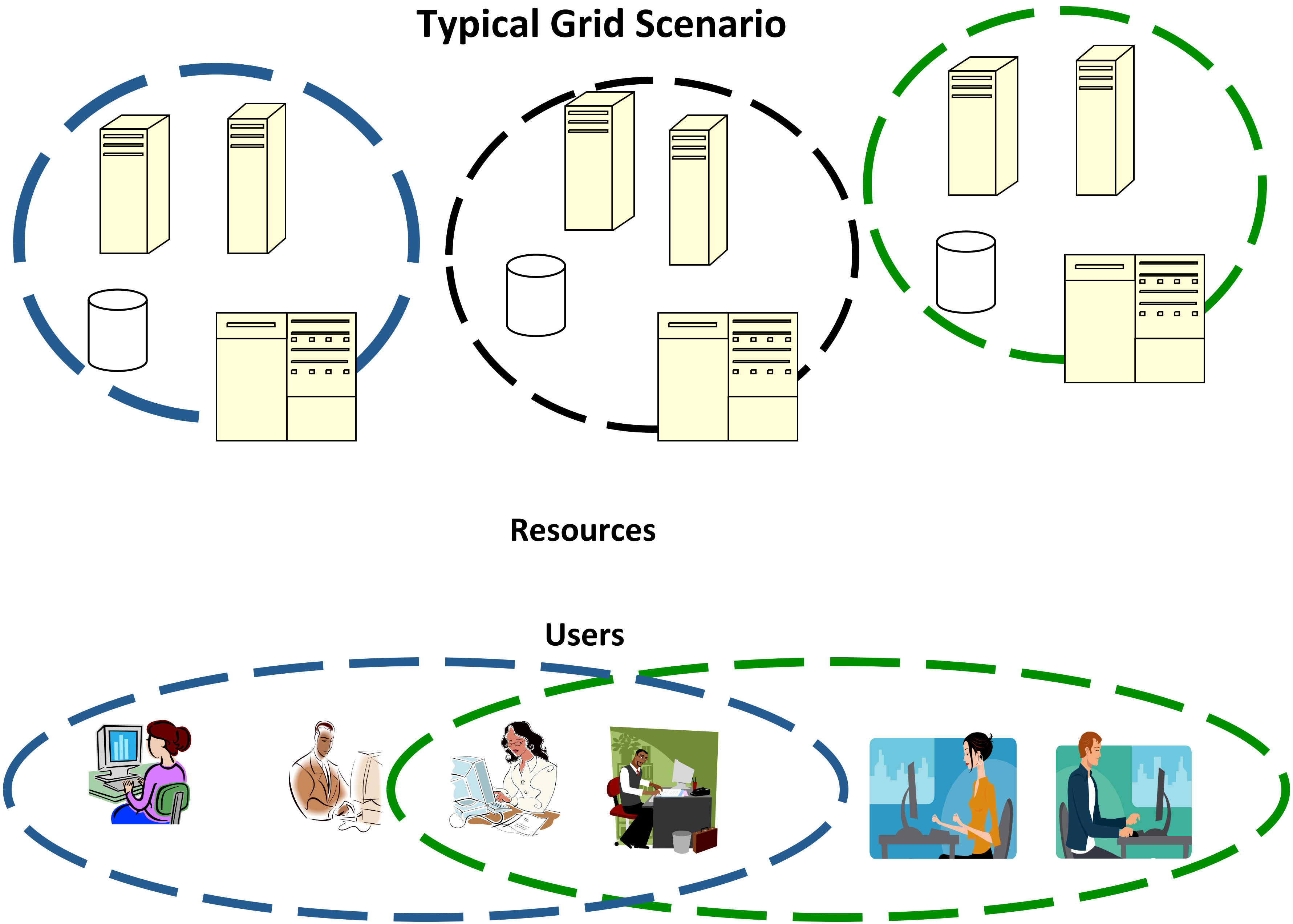
Emergence of Open Grid Standards

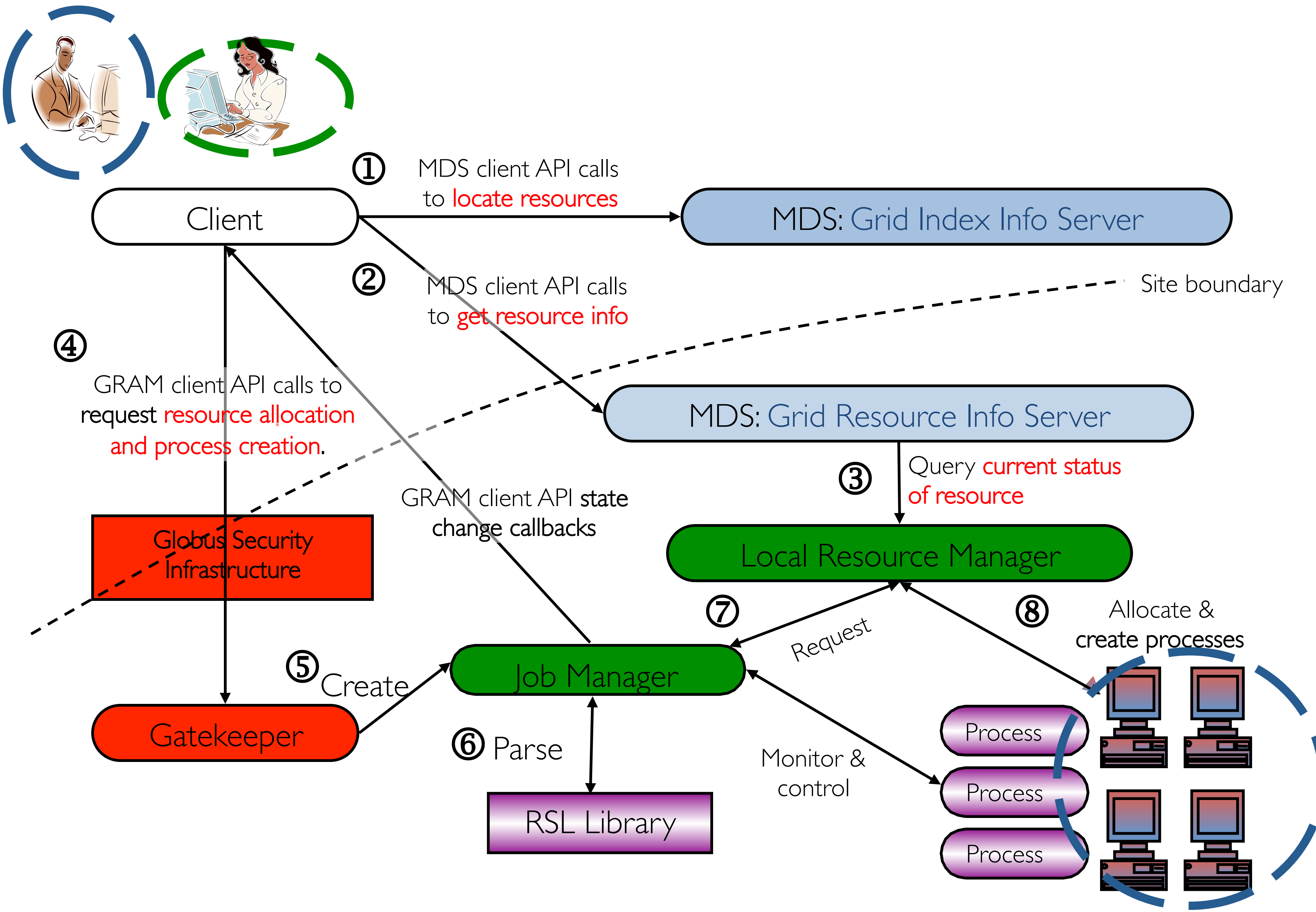


"Grid Computing and Scaling Up the Internet" I. Foster, IPv6 Forum, an Diego, 2003

outline

- Grid computing: Approach and vision
- Potential Application for Grid computing:
- Grid-middleware
- Example of Services provided by Grid Systems:
  - Authentication, Authorization, Data management, computing resource management

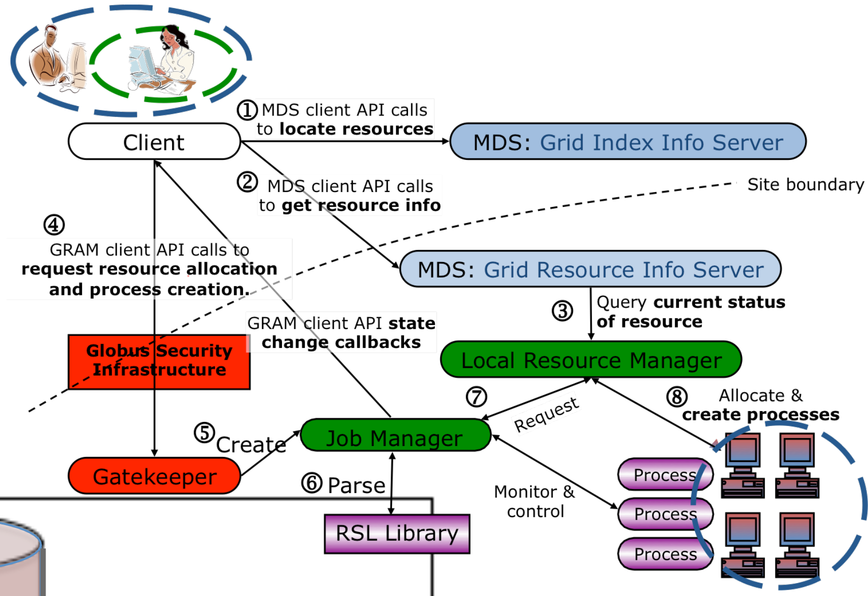

Typical Grid Scenario
Resources
Users



Globus Toolkit 2.0 Components

The Four components of a Grid infrastructure

- Resource Management
- Information services
- Data Management
- Security



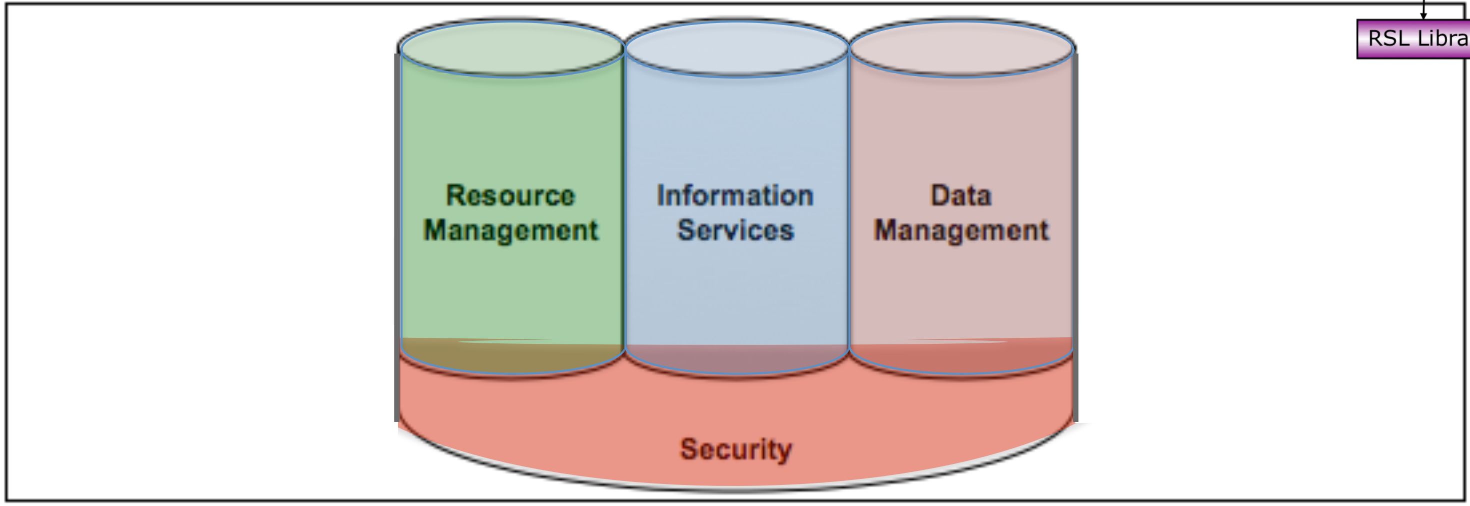
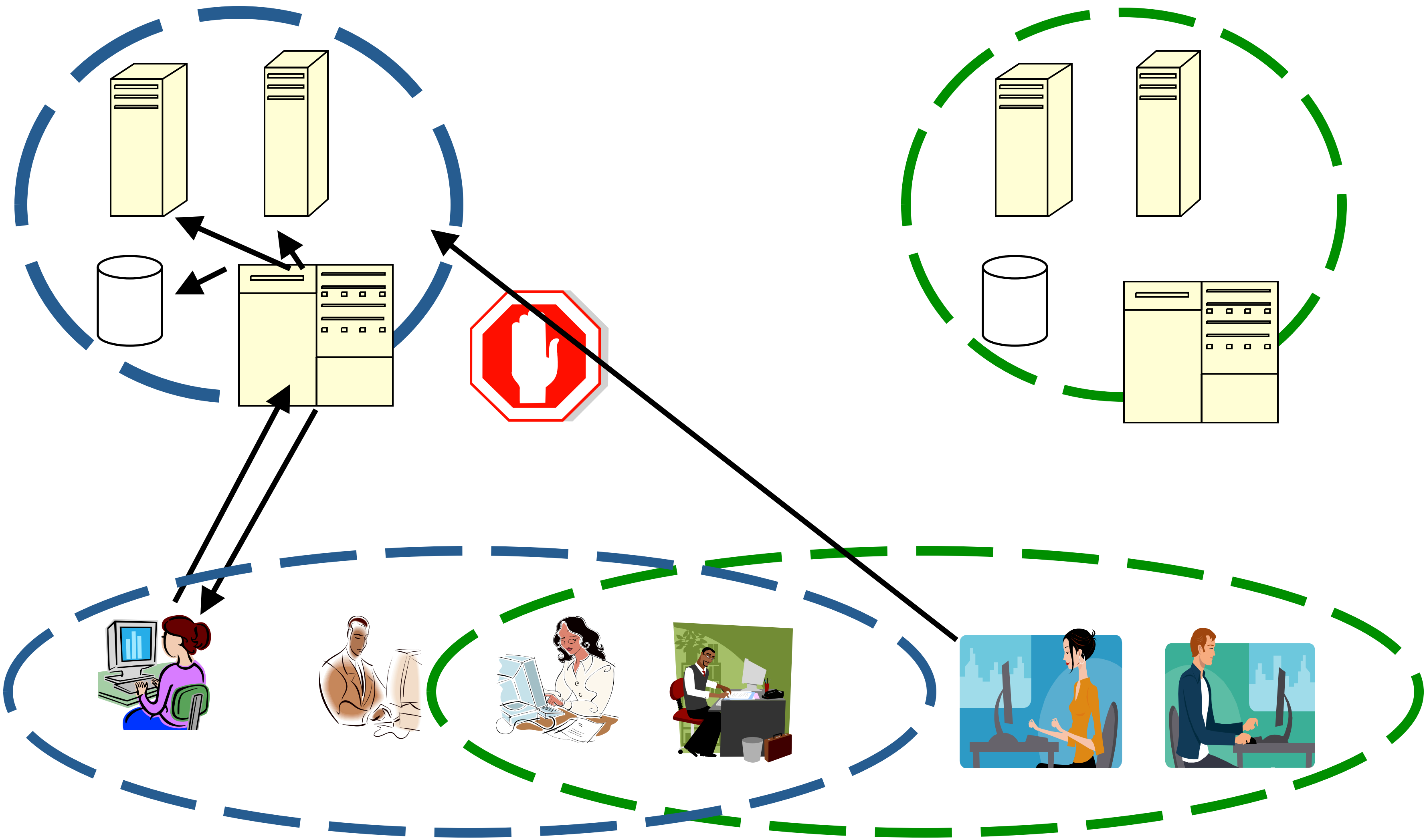


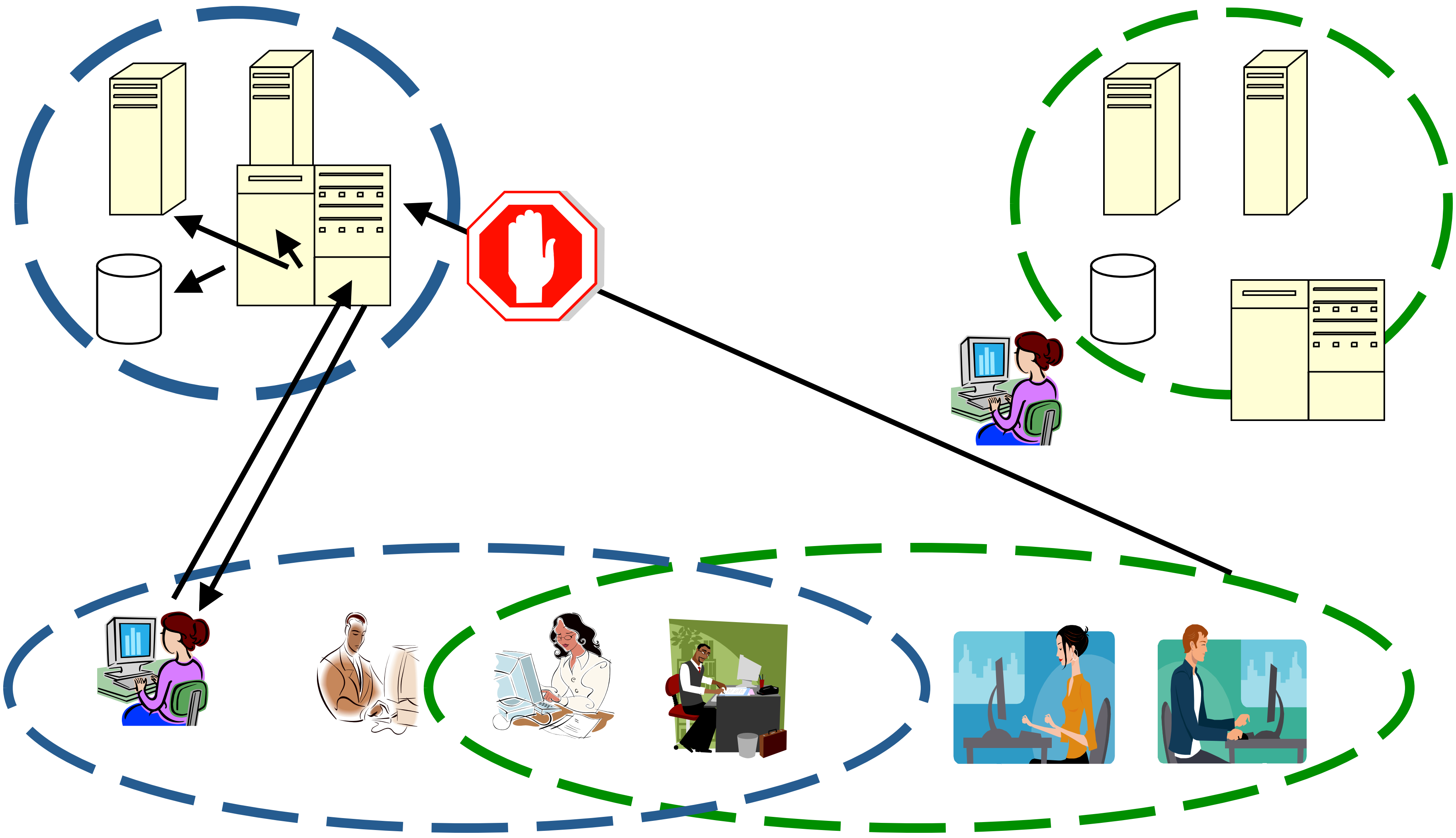
Figure 1-1 Grid computing key areas

Grid Security: Identity



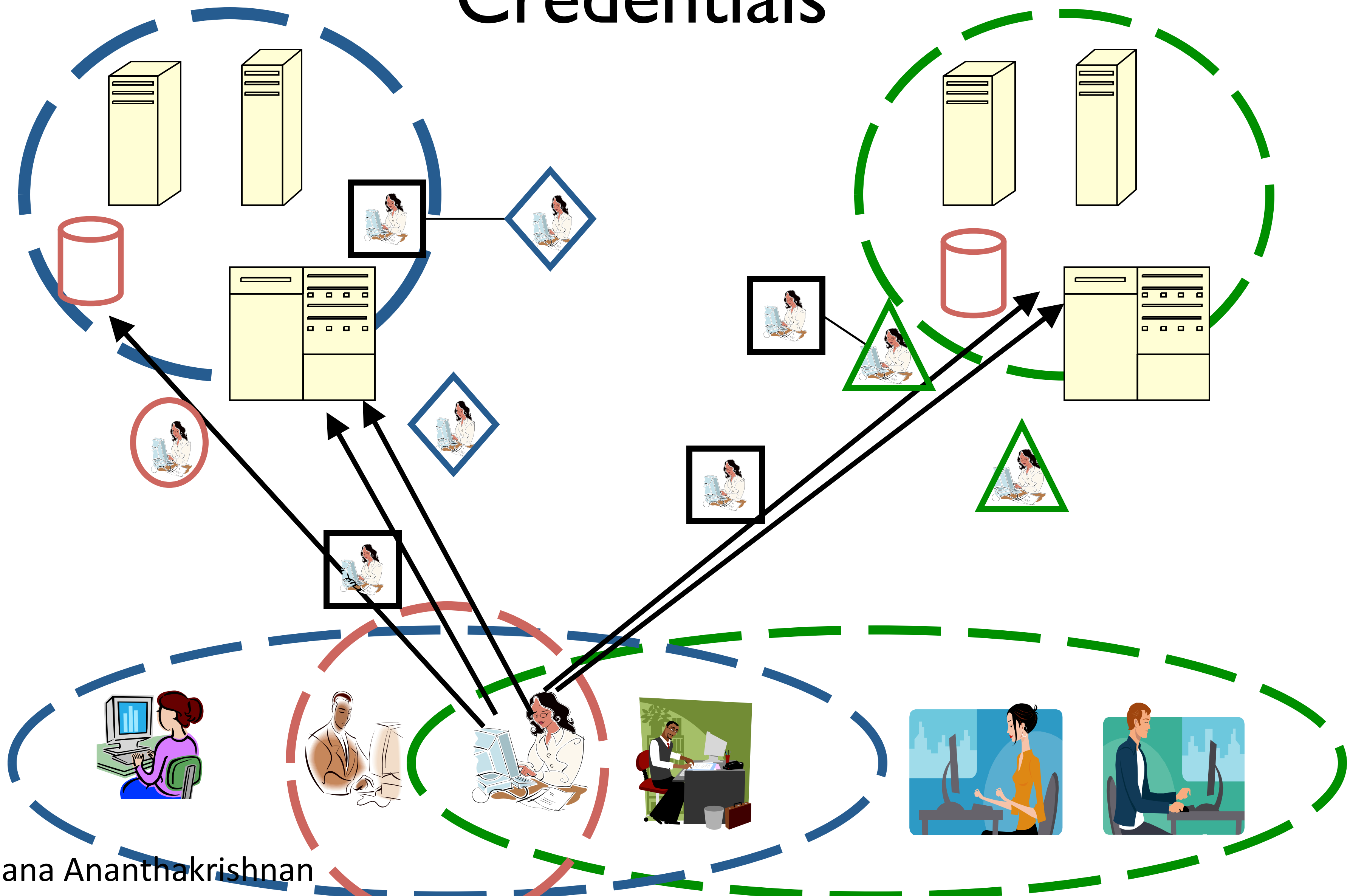
Rachana Ananthakrishnan

Grid Security: Authentication



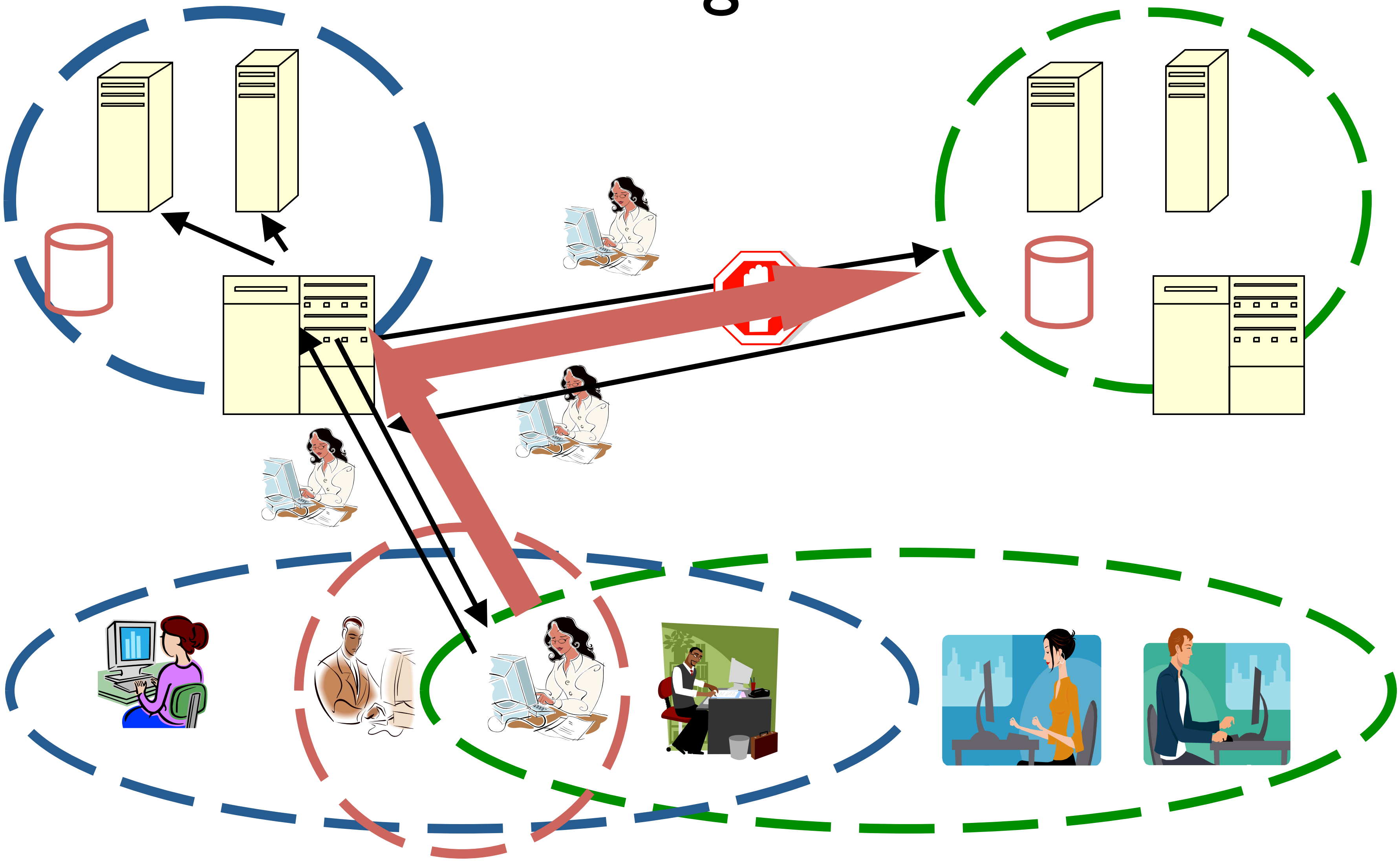
Rachana Ananthakrishnan

Grid Security: Single Sign On Varied Credentials



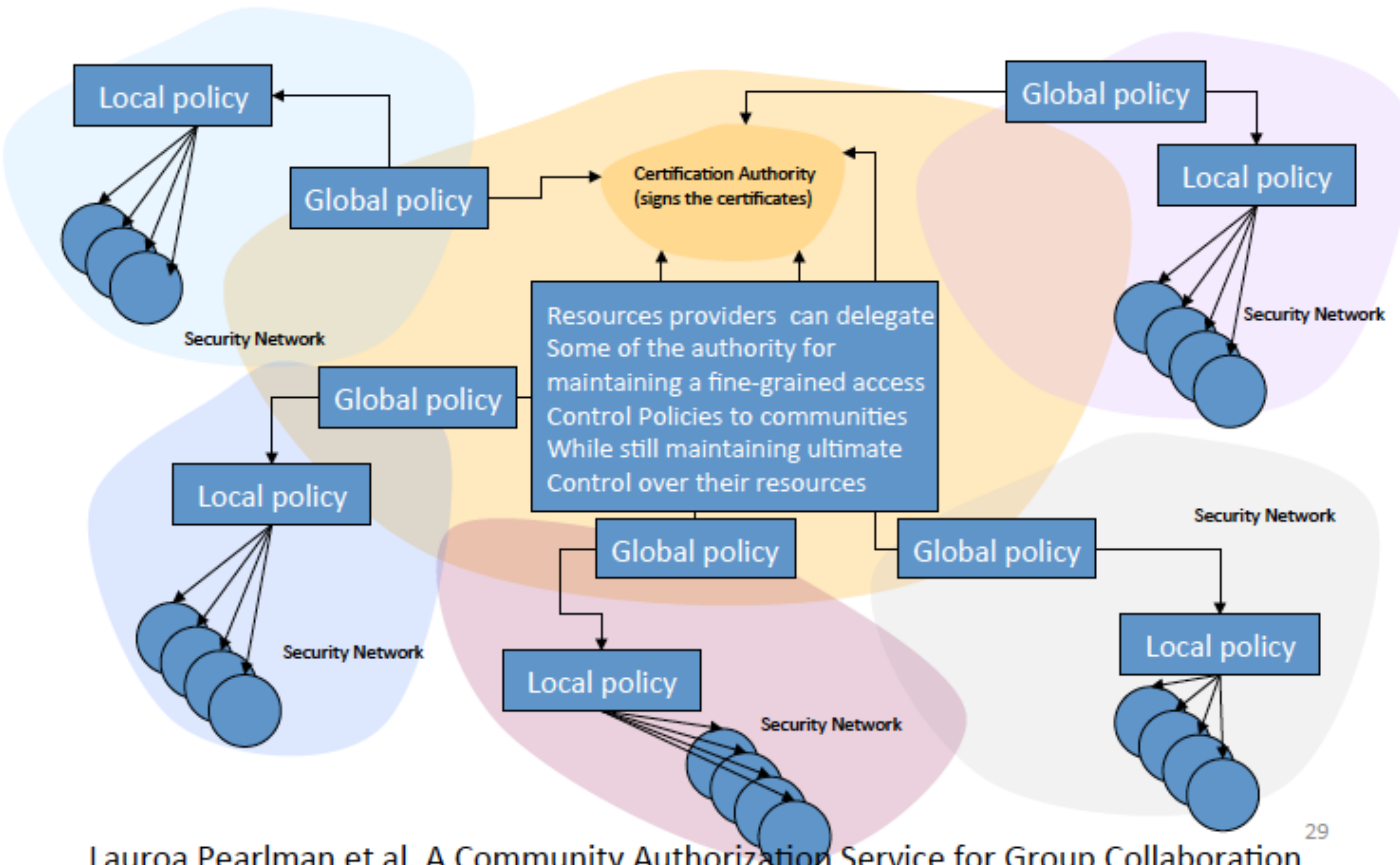
Rachana Ananthakrishnan

Grid Security: Single Sign On Delegation



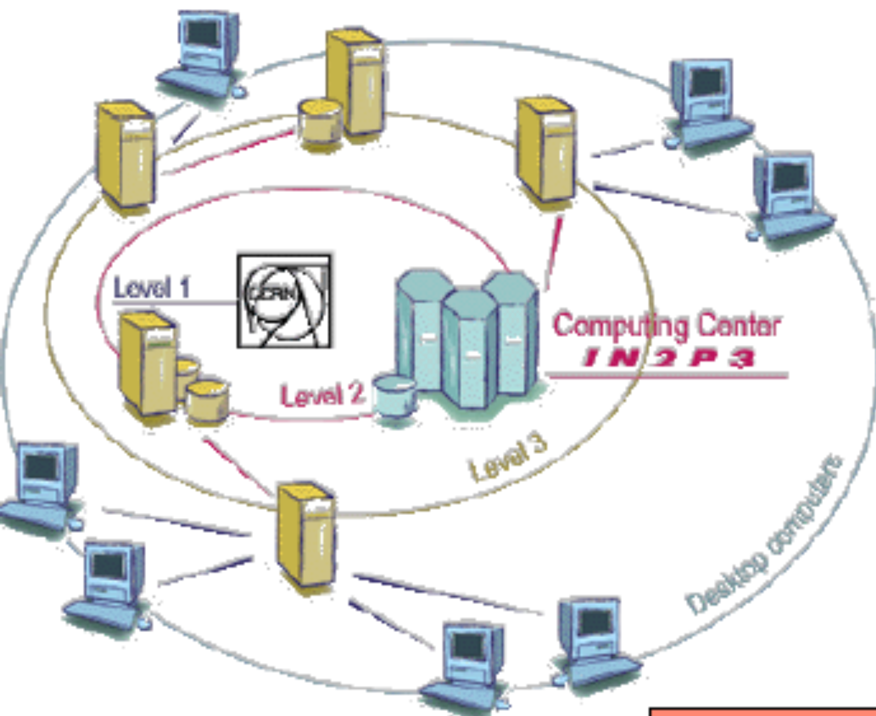
Rachana Ananthakrishnan

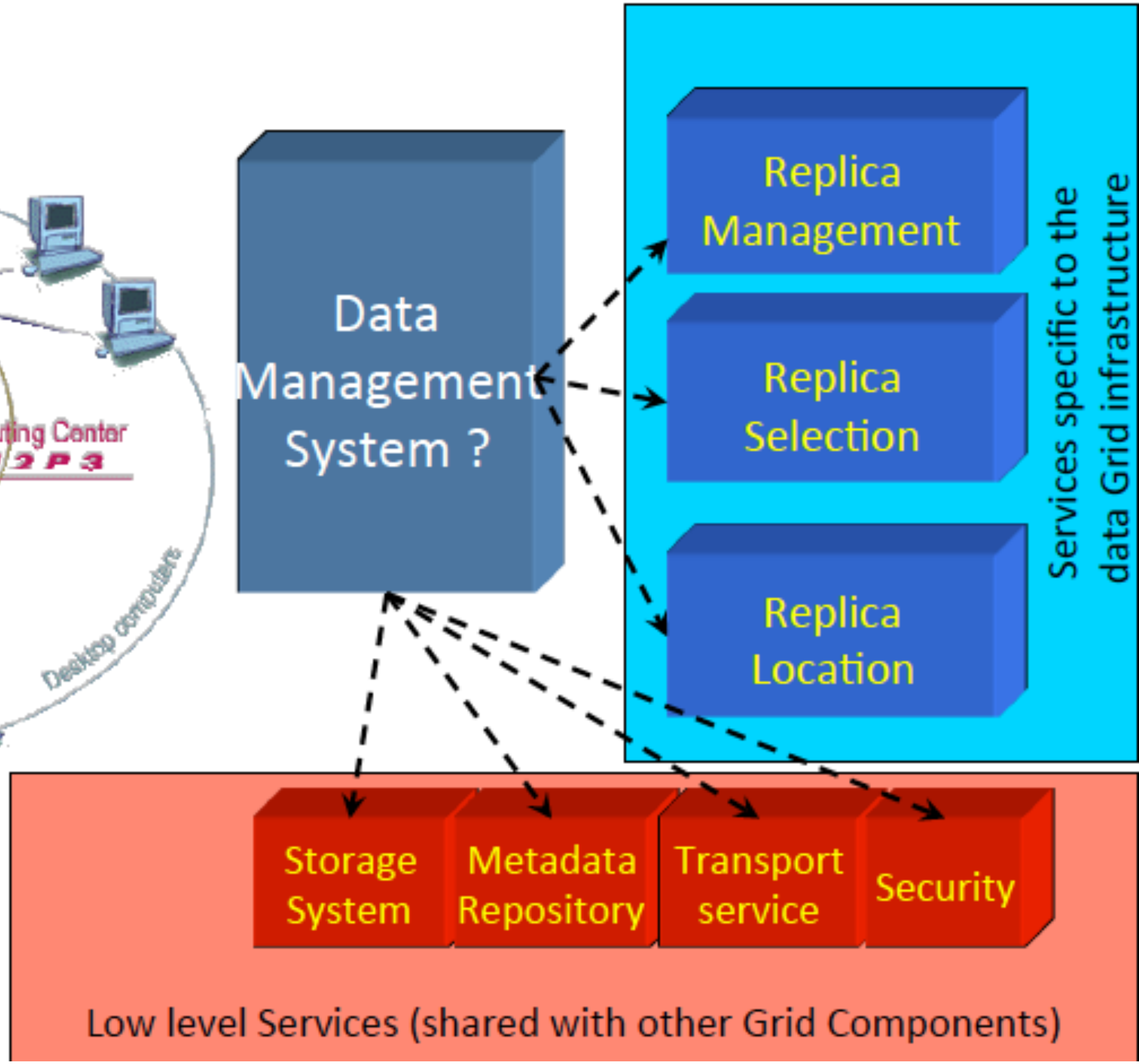
Security cross Grid (V.O.)



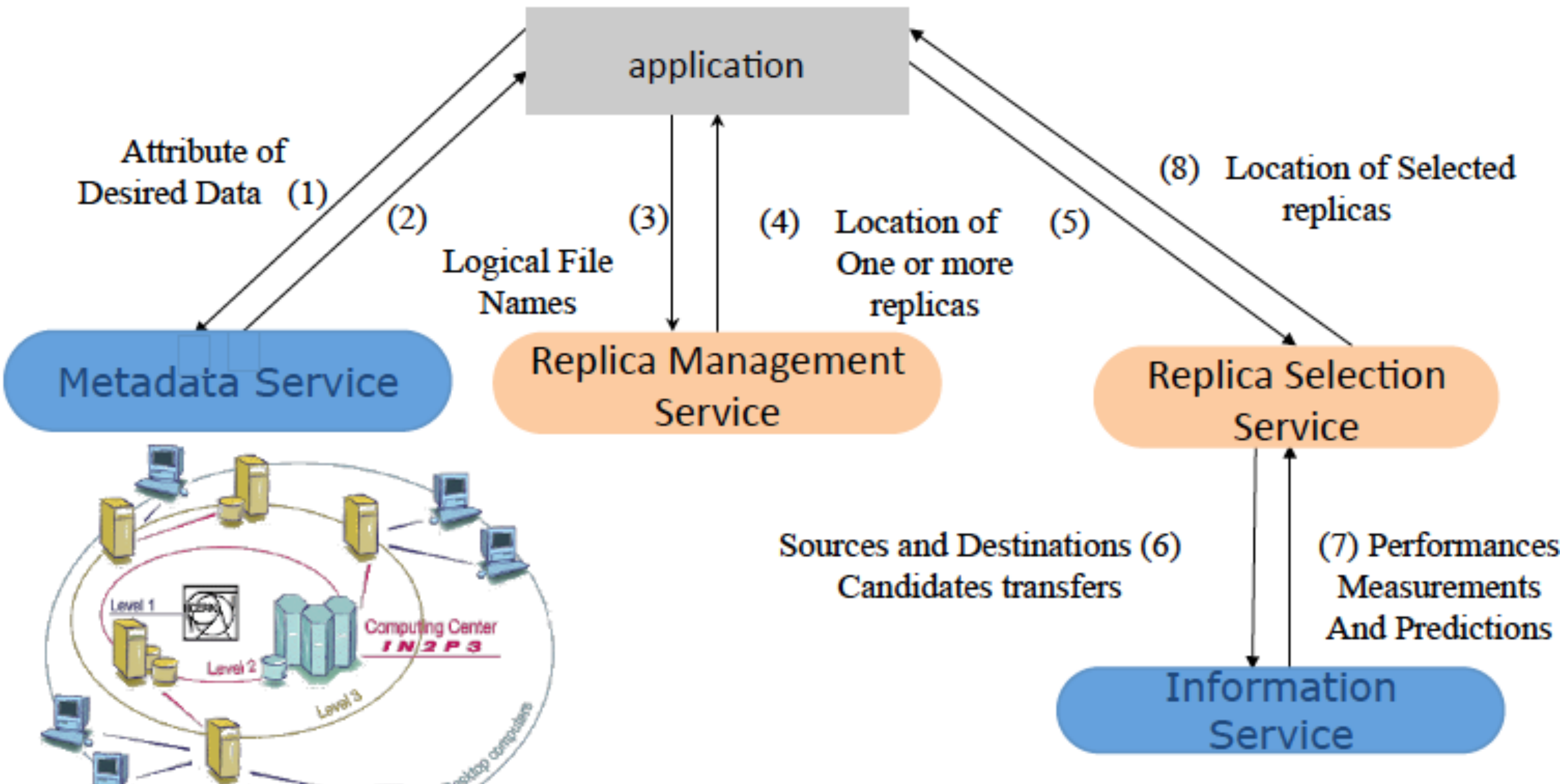
Lauroa Pearlman et al. A Community Authorization Service for Group Collaboration

Grid data management



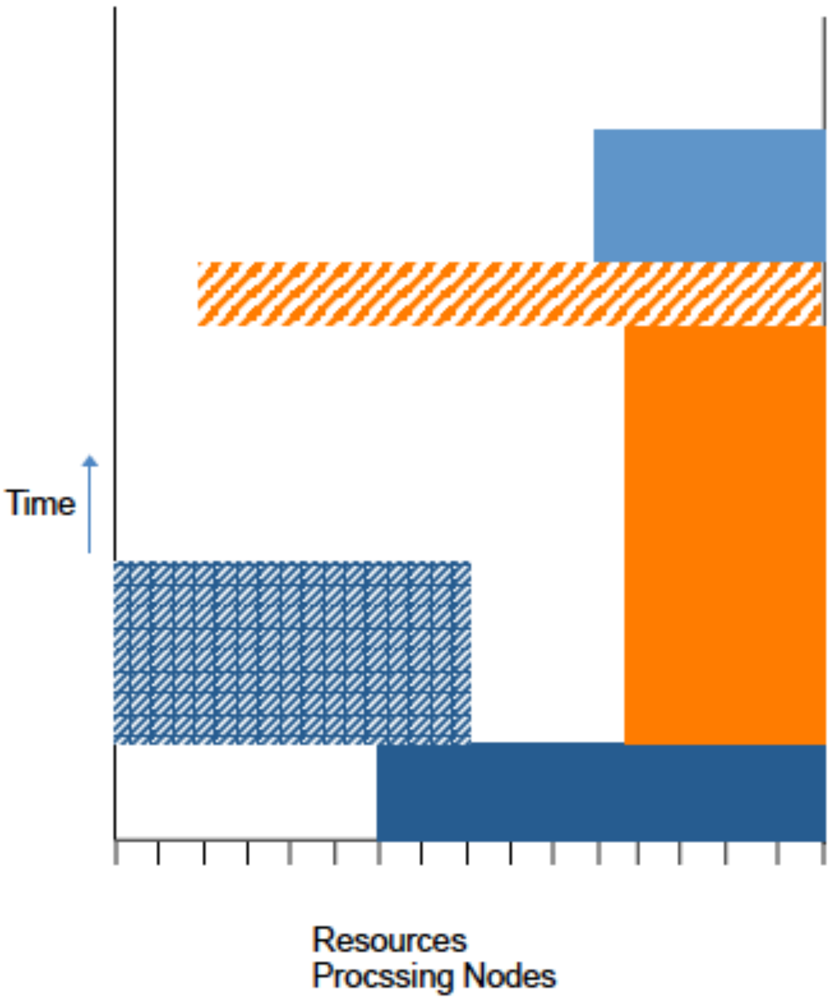


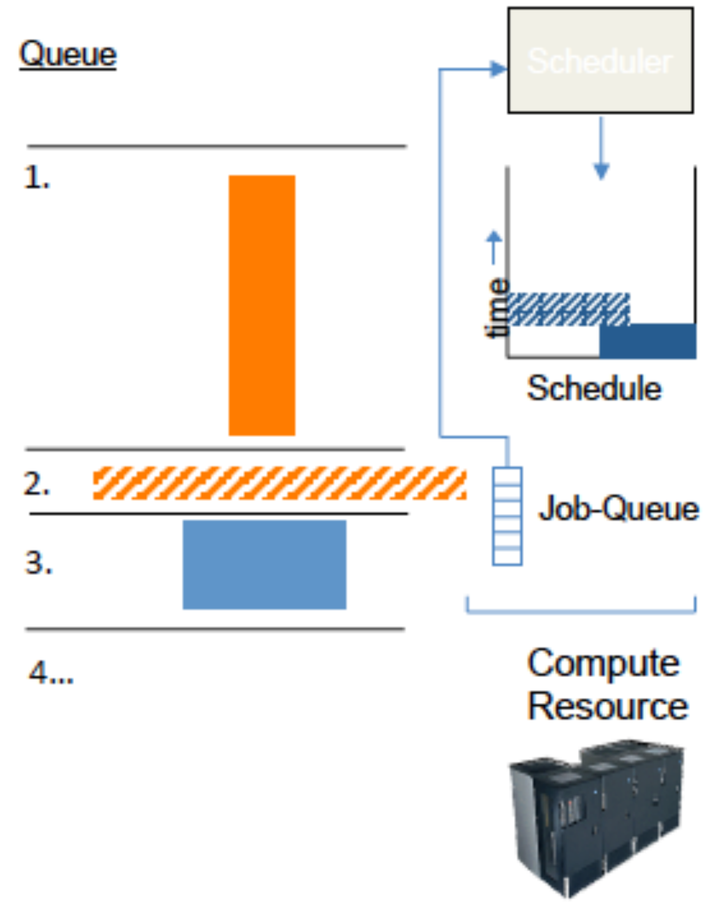
A Data selection scenario



Sudharshan Vazhkudai “Replica Selection in the Globus Data Grid”

Grid Job Scheduling





Grid Programming Issues

- Portability, Interoperability, and Adaptability
- Discovery
- Performance
- Fault Tolerance
- Security

Programming models

- Shared-state models
- Message passing models
- RPC and RMI models
- Hybrid Models
- Peer to Peer Models
- Web Service Models
- …