Grid Information Services

The INFNGrid Project Team

Grid Information Services

- The aim of the Information and Monitoring Service is to deliver a flexible infrastructure that provides information
- System information is critical to operation of the grid and construction of applications
  - What resources are available?
    - Resource discovery
  - What is the "state" of the grid?
    - Resource selection
  - How to optimize resource use
    - Application configuration and adaptation?
- Used by:
  - Grid Users (applications)
  - Grid middleware services
    - E.g. the Resource Broker query the IS to find the available Grid resources and check their characteristics and status

Information Service Schema

- The schema defines the information that is represented in the Information Services
- All the resources that participate in the Grid system and are requested to be discoverable and monitored should be represented
- Necessary to agree on a common schema, in order to guarantee interoperability
- → Glue schema

Glue

- GLUE: Grid Laboratory Uniform Environment
- collaboration effort focusing on interoperability between US and EU HEP Grid related projects
- Targeted at core grid services
  - Resource Discovery and Monitoring
    - GLUE Schema
  - Authorization and Authentication
  - Data movement infrastructure
  - Common software deployment procedures
- Preserving coexistence for collective services
- Promoted by DataTAG (EU) and iVDGL (US)
- Contributions from DataGrid, Globus, PPDG and GriPhyn

Glue Schema

- Three types of resources modeled in the Glue Schema:
  - Computing Resources (Computing Element: CE)
  - Storage Resources (Storage Element: SE)
  - Network Resources (Network Element: SE)
- Not HEP specific
  - Discussed at GGF

GLUE Computing resources

- What is the core offered functionality?
  - Computing power
- What I need to know in order to use it?
  - Offered execution environment (e.g., OS type, available software libraries)
  - Offered Quality of Service (e.g., estimated response time)
  - Status (e.g., number of running jobs)
  - Policy (e.g., max execution time, assigned CPUs)
  - Access rights (e.g., can I use it?)
  - Location (e.g., Uniform Resource Locator or URL)

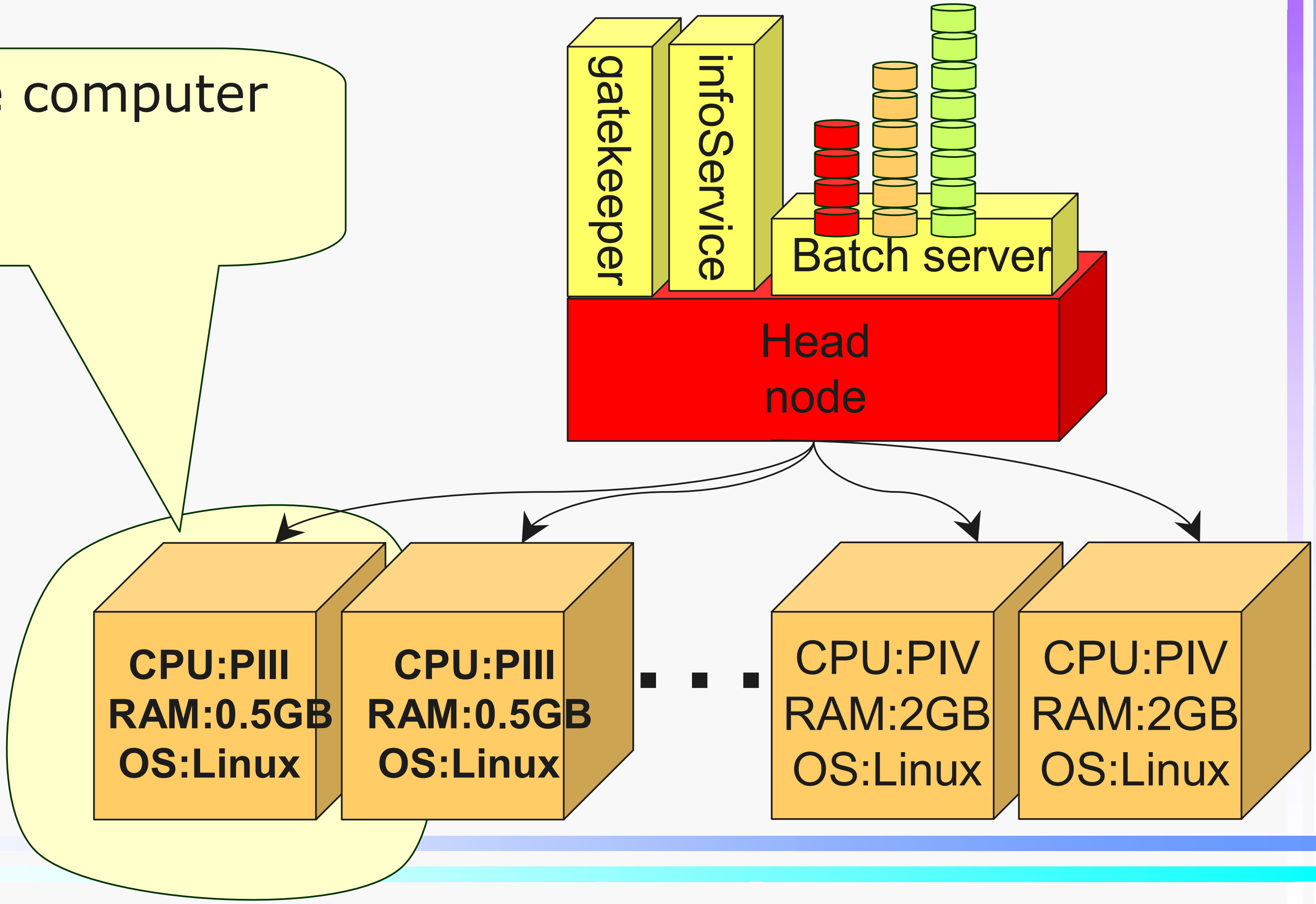
GLUE Computing resources: some more thought about the service

- The **computing power** is typically offered by cluster systems
- Requests are typically staged into **queues** for efficient system usage
- Queue policies enable **service differentiation** (e.g., dedicated CPUs vs. shared CPUs assignment, differentiated max CPU time, differentiated queue service strategy)
- A service has quality aspects

→ The **computing service** is in 1-to-1 relationship with a queue and its assigned computing resources

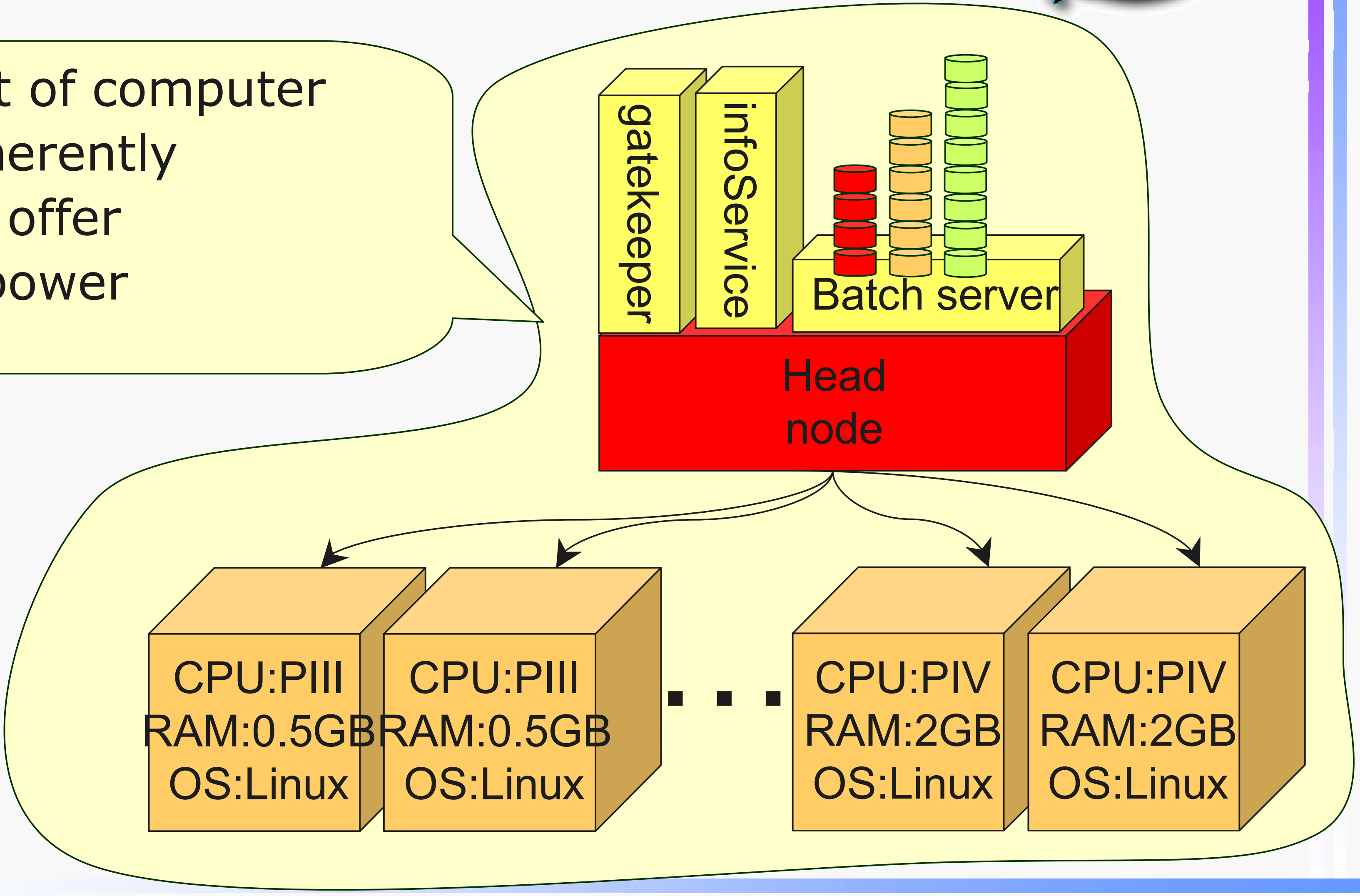
GLUE Computing resources: Host (the system)

Host: a single computer system



GLUE Computing resources: Cluster (the system)

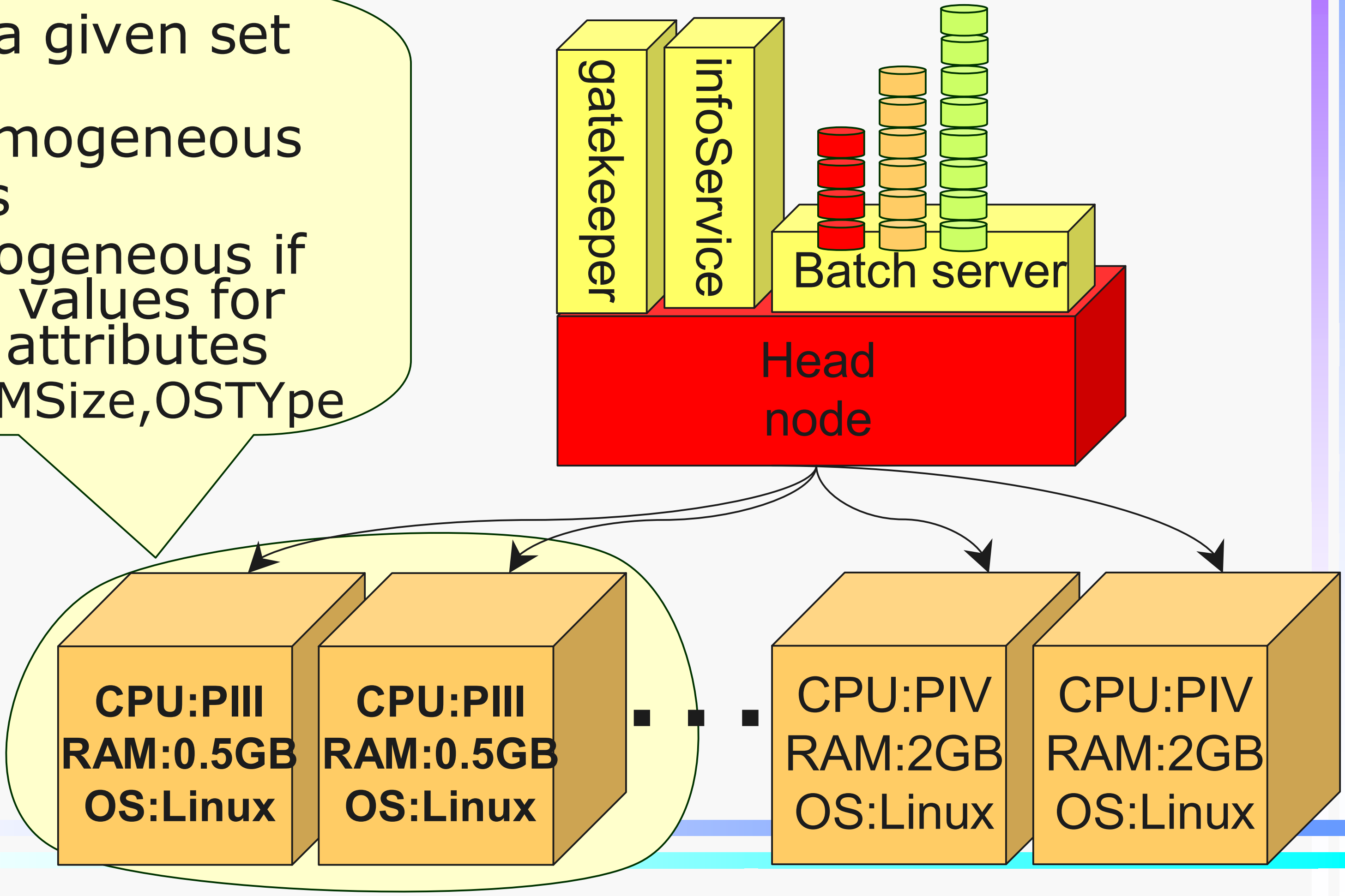
Cluster: set of computer systems coherently Managed to offer computing power



GLUE Computing resources: SubCluster (aggregate information)

SubCluster: for a given set of properties, an homogeneous collection of hosts

- Hosts are homogeneous if the have same values for the give set of attributes
- e.g.CPUType,RAMSize,OSTYpe

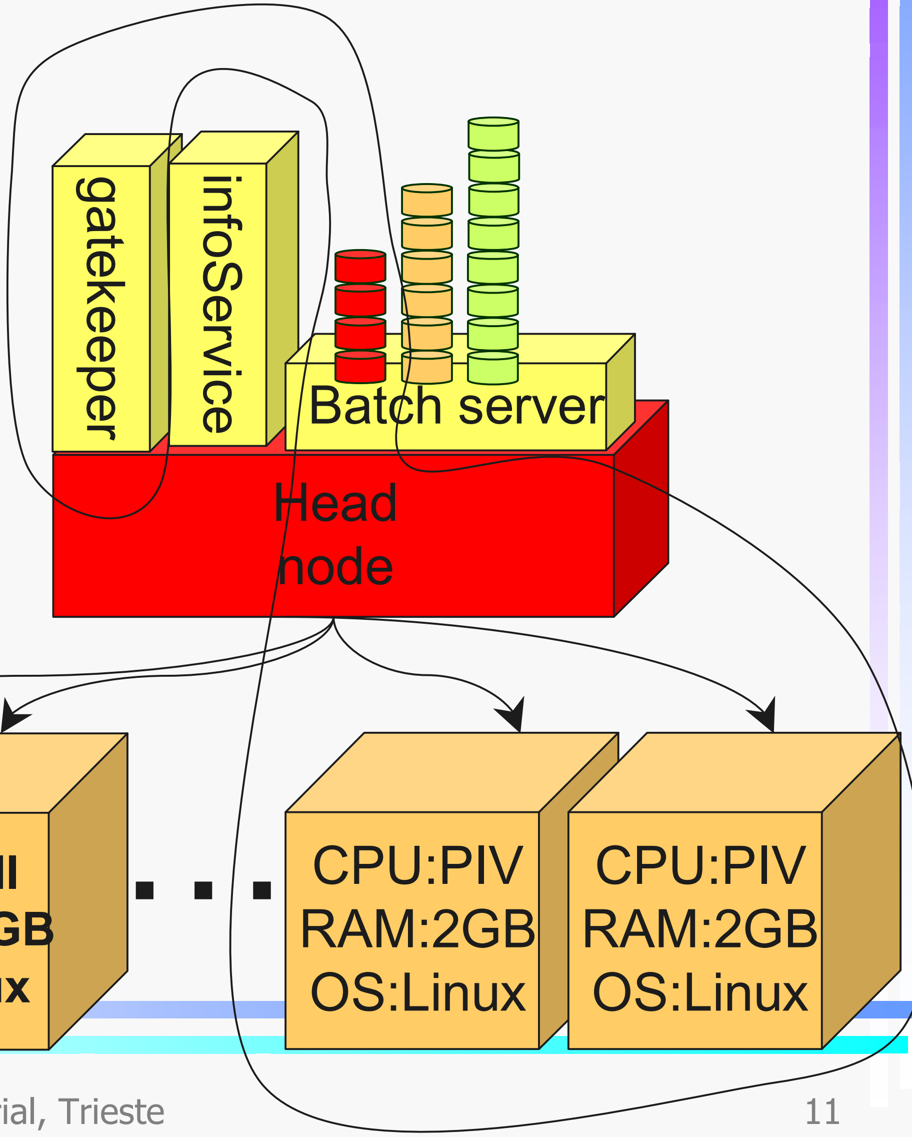


☝ Number of nodes maybe O(1000)

GLUE Computing resources: Computing Element (the Service)

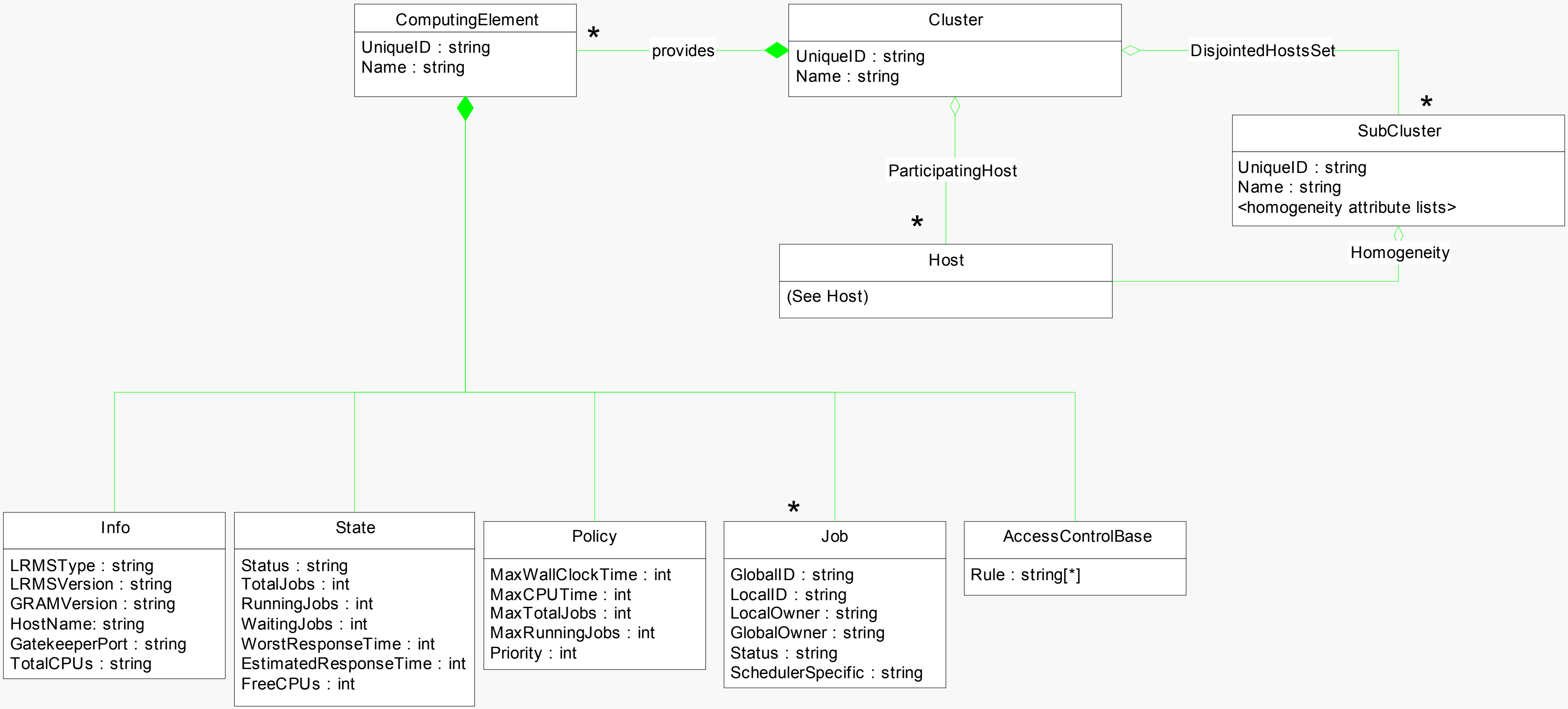
Computing Element: entry point into a queue of a batch system

- information associated with a computing element is limited only to information relevant to the queue
- Resource details relates to the system



in the example the red queue is assigned for two hosts

Glue Schema 1.1 (UML Class Diagram)
Computing Resources::cluster
Computing Resources::computing element



ComputingElement

UniqueID : string
Name : string

provides

*

Cluster

UniqueID : string
Name : string

DisjointedHostsSet

*

SubCluster

UniqueID : string
Name : string
<homogeneity attribute lists>

ParticipatingHost

*

Host

(See Host)

Homogeneity

Info

LRMSType : string
LRMSVersion : string
GRAMVersion : string
HostName: string
GatekeeperPort : string
TotalCPUs : string

State

Status : string
TotalJobs : int
RunningJobs : int
WaitingJobs : int
WorstResponseTime : int
EstimatedResponseTime : int
FreeCPUs : int

Policy

MaxWallClockTime : int
MaxCPUTime : int
MaxTotalJobs : int
MaxRunningJobs : int
Priority : int

*

Job

GlobalID : string
LocalID : string
LocalOwner : string
GlobalOwner : string
Status : string
SchedulerSpecific : string

AccessControlBase

Rule : string[*]

GLUE Storage resources

- What is the core offered functionality?
  - Storage Space usage
- What I need to know in order to use it?
  - Storage Service manager type (e.g., file system, edg-se, srmv1, srmv2)
  - Available data access protocols (e.g., gridftp, rfio)
  - Offered Quality of Service (e.g., availability, reliability)
  - State (e.g., available space)
  - Policy (e.g., file life time, MaxFileSize)
  - Access rights (e.g., can I use it?)
  - Location (e.g., Uniform Resource Locator or URL)

Storage **Service**/Space/Library

- **Storage Service:**
  - Grid service **identified by a URI** that **manages disk and tape resources in term of Storage Spaces**
  - All **hardware details** are **masked**
  - The Storage Service **performs file transfer** in or out of its Storage Spaces **using** a specified set of **data access protocols** (e.g. GridFTP, rfio, nfs)
  - **Files** are **managed** in respect of the lifetime policy specified for the Storage Space where they are kept

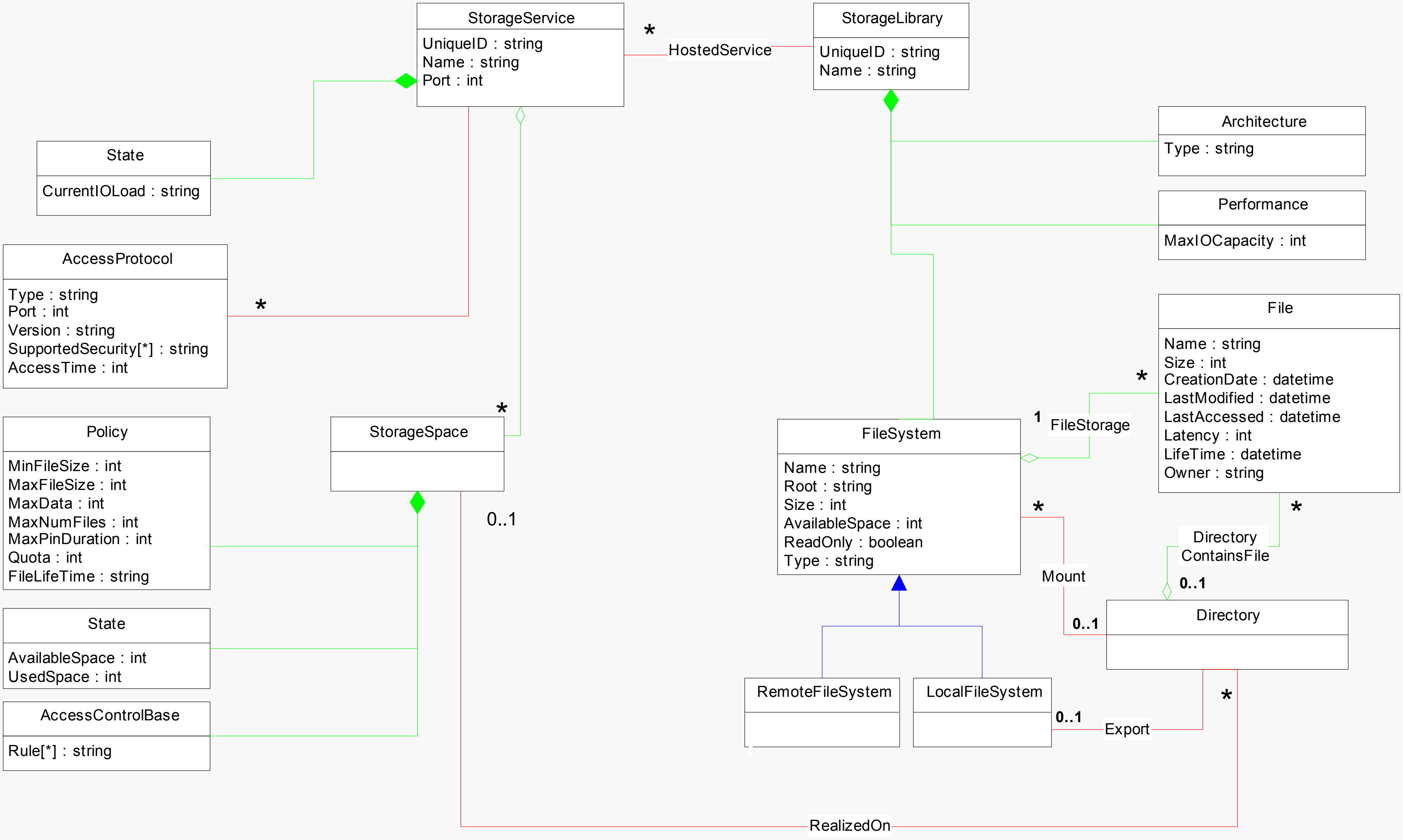
Storage Service/**Space**/Library

- **Storage Space:** portion of a logical storage extent that:
  - Is **assigned to a Virtual Organization**
  - Is **associated to a directory** of the underlying file system (e.g. /permanent/CMS)
  - Has a **set of policies** (MaxFileSize, MinFileSize, MaxData, MaxNumFiles, MaxPinDuration, Quota)
  - Has a set of **access control base rules** (to be used to publish rules to discover who can access what)
  - Has a **state** (available space, used space)

Storage Service/Space/**Library**

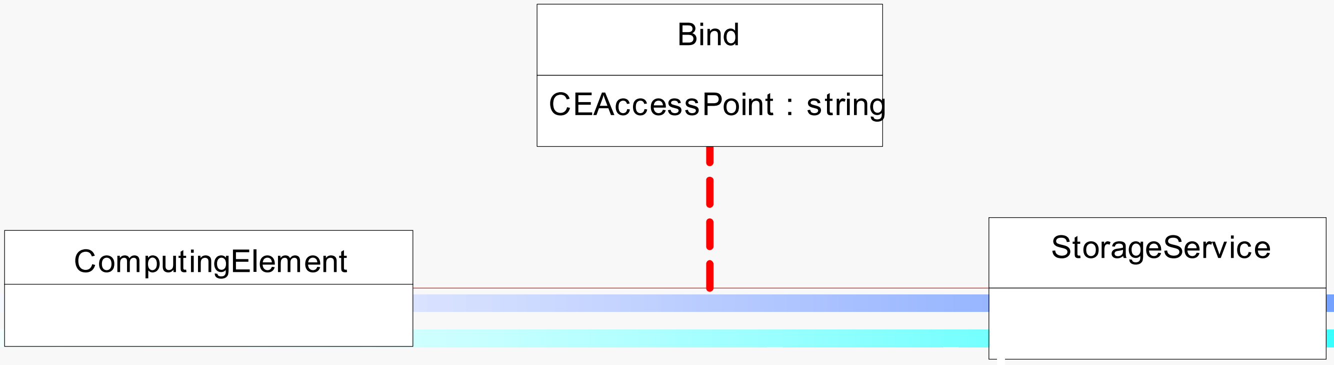
- **Storage Library**: the machine providing for both storage space and storage service
  - ☞ A storage system can vary from a simple disk server to complex hierarchical storage systems

Glue Schema 1.1 (UML Class Diagram)
Storage Resources::Storage Service
Storage Resources::Storage Space
Storage Resources::Storage Library



Expressing relationships among Computing and Storage Services

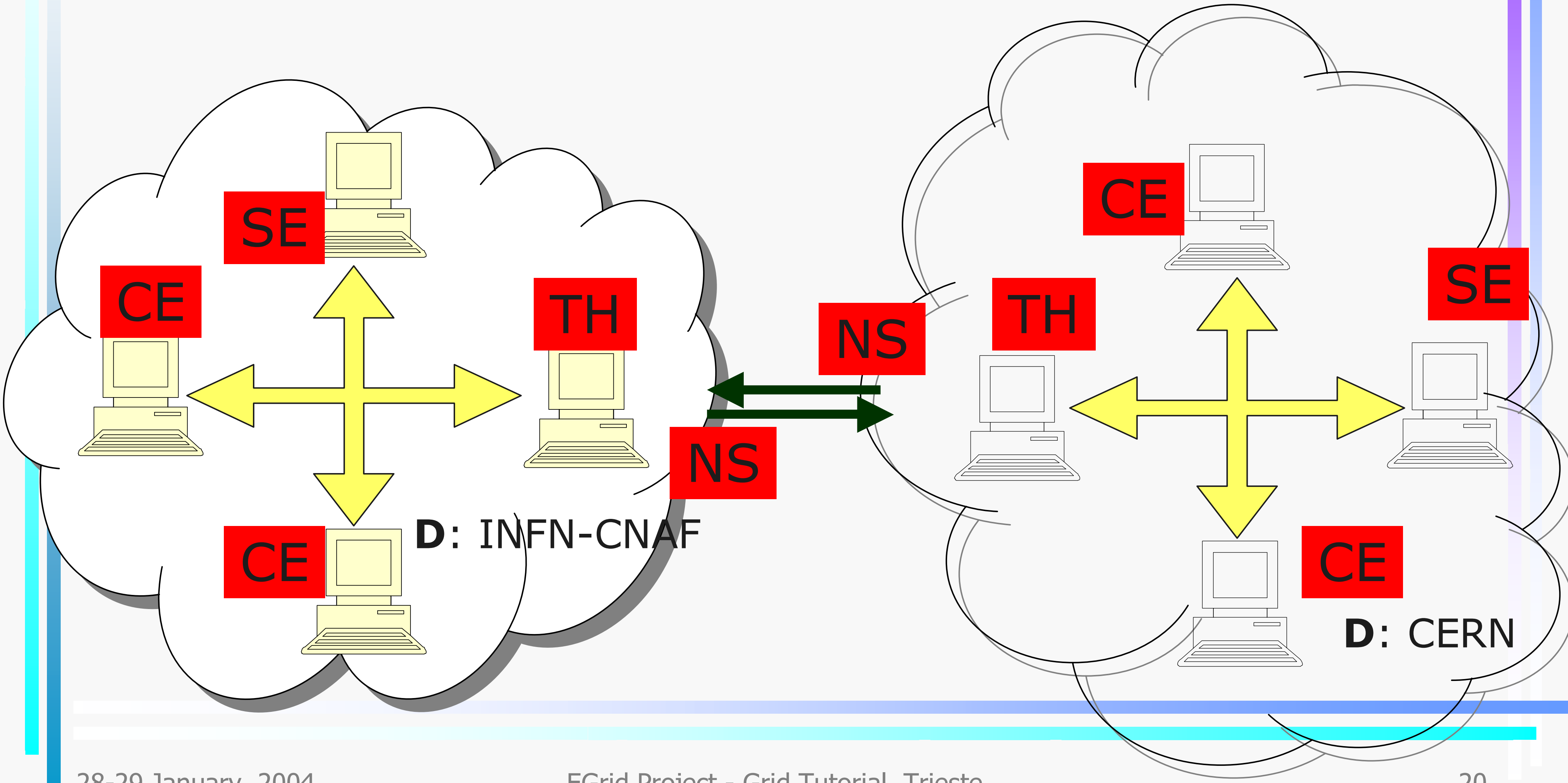
- A typical job execution request involve:
  - Certain properties for the computing service
  - Access to a storage space
- Possible to specify preferences on which Storage Services should be used by jobs running on certain computing services
  - Usually to be preferred Storage Spaces “close to” the considered computing service
- The possibility of expressing such preference is modeled by (GLUE CE-SE Bind concept)
- CE Access point refer to a possible NFS mountpoint



GLUE Network Resources

- Definition of a network model that enables an efficient and scalable way of representing the communication capabilities between grid services
- Partition the Grid into Domains, and limiting the monitoring activity to the observation of Domain-to-Domain paths
- Communication characteristics measured within the boundaries of D1 and D2 are negligible with respect to the same characteristic measured between the boundaries of D1 and D2
- Work in Progress

GLUE Network Service example scenario



Information Service implementations

- Two main Information Service implementations:
  - Globus MDS (Metacomputing Directory Service or Monitoring and Discovery Service as it is now called)
    - Used in the US Grid projects, in the LCG Grid, in the Grid.it grid, etc.
  - EDG R-GMA (Relation Grid Monitoring Architecture)
    - Used in the EDG testbed

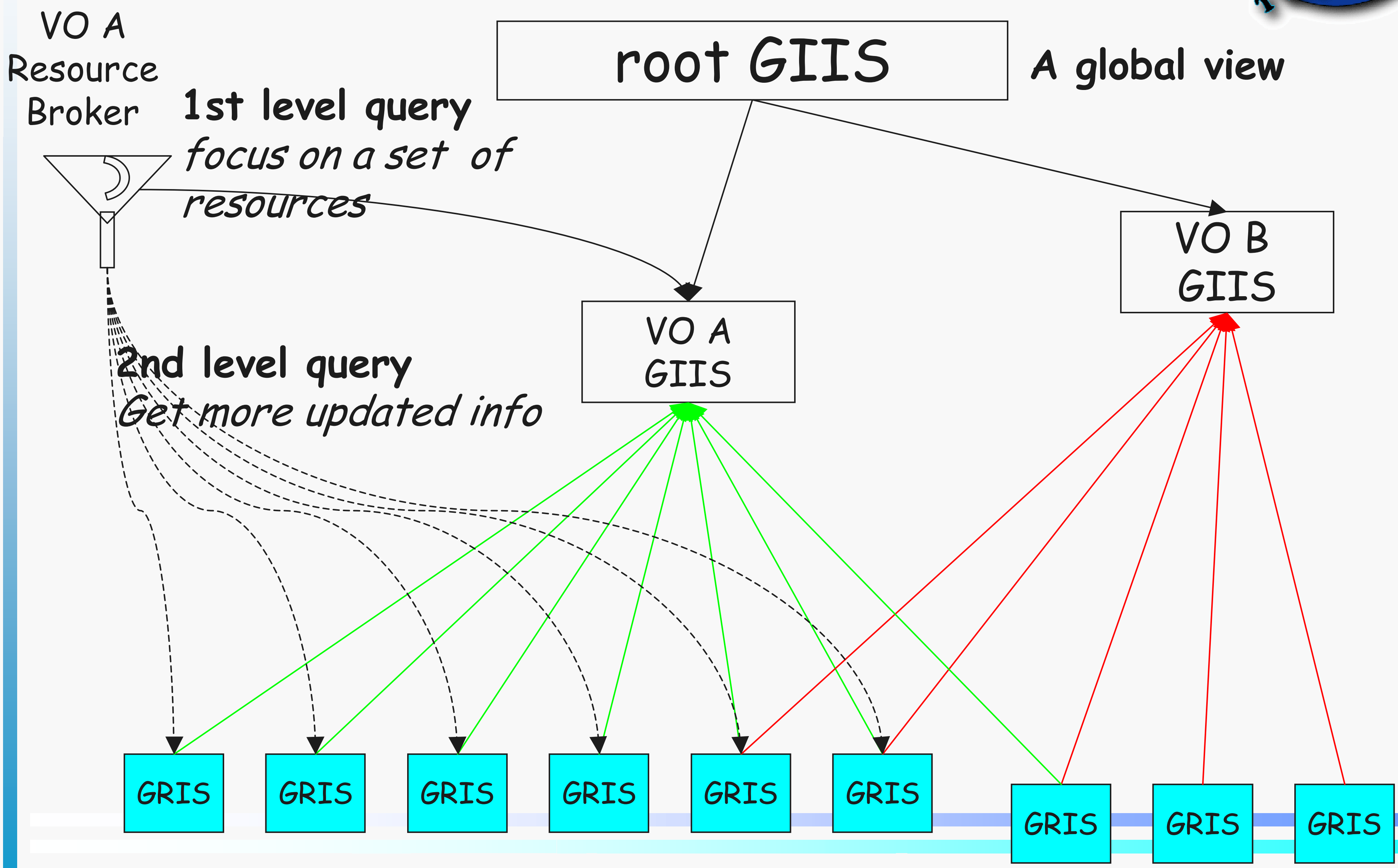
MDS

- Use LDAP
  - Standard interface and protocol
- Access information in a distributed directory
  - Directory represented by collection of LDAP servers
  - Information is cached by the server to improve performance
- Information updated by Information providers
  - Information providers for Computing Element, Storage Elements, …
- Information dynamically available to tools and applications

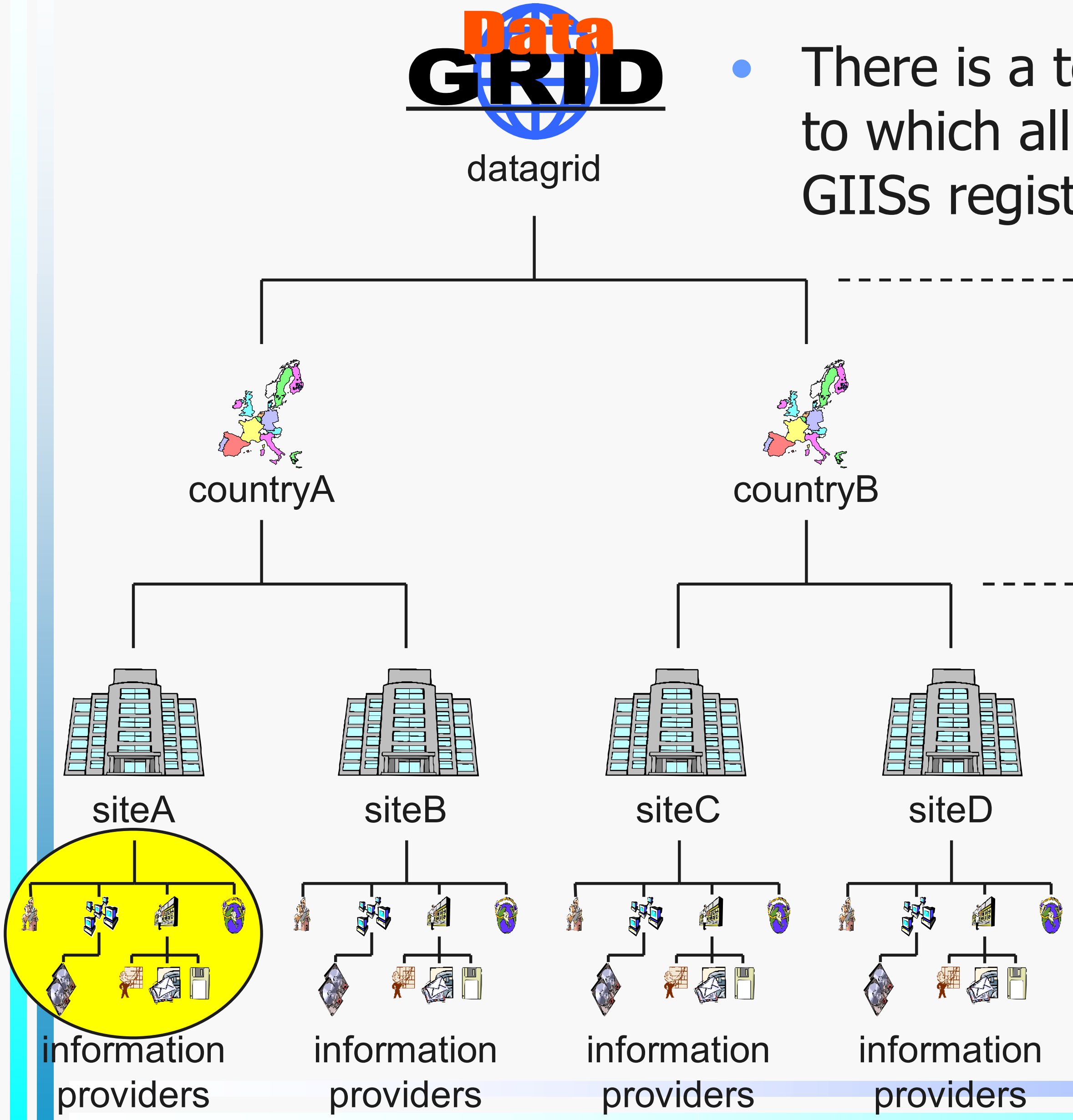
Two Classes Of MDS Servers

- Grid Resource Information Service (GRIS)
  - Supplies information about a specific resource
  - 'White page' functionality
    - E.g. look up the amount of memory, the load, etc. of a particular resource
  - The GRISs use soft state registration to register with one or more GIISs
- Grid Index Information Service (GIIS)
  - Supplies collection of information which was gathered from multiple GRIS servers
  - 'Yellow pages' functionality
    - E.g. find all the resources of a particular class or with a particular property
  - In turn a GIIS may register with another GIIS
  - A GIIS may represent a site, country, virtual organization, etc.

Scheduling/Resource discovery scenario



Example of GRIS/GIIS Hierarchy



- There is a top level EDG GIIS to which all of the country GIISs register
- Each country has a GIIS to which all of the site GIISs register
- Each Site has a Grid Information Index Server (GIIS) which acts as a single point of contact for all of the sites resources. The GRISs register with their site GIIS
- Information providers publish information to a local LDAP server known as a Grid Resource Information Server (GRIS)

Entries

- The entries are organized into tree
  - Called Directory Information Tree (DIT)
- Position in tree uniquely names entry
  - Distinguish Name (DN)
- Entries are types by their object class
  - GlueCE
  - GlueCEPolicy
  - GlueSE
  - GlueCESEBind
  - …

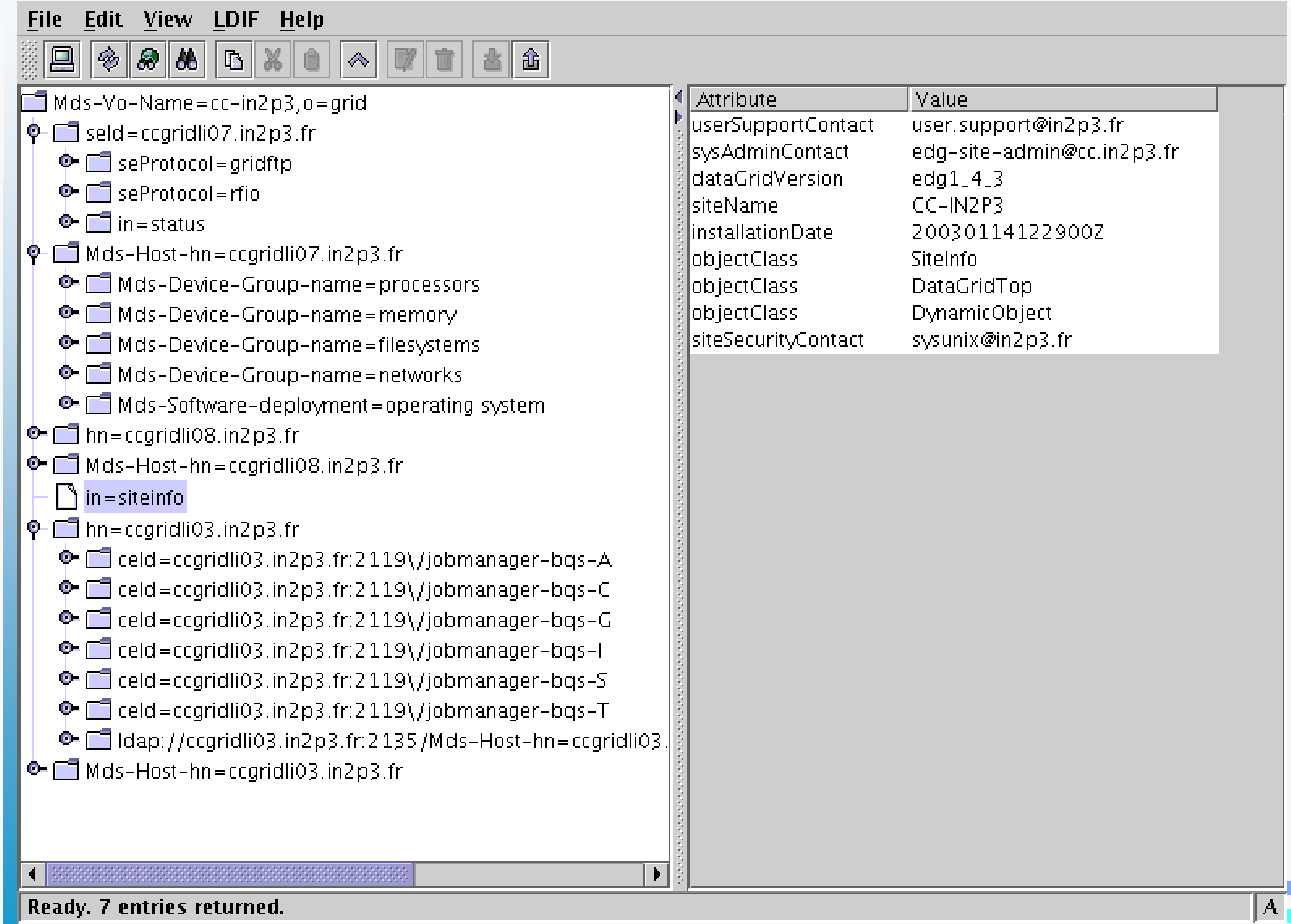
Querying the Information Service

- Queries can be posed to the Information Service using LDAP search commands:

```
$ldapsearch\
 -x\
 -H ldap://lxshare0225.cern.ch:2135\
 -b 'Mds-Vo-name=datagrid,o=grid’\
 'objectclass=GlueCE‘\
 GlueCEUniqueId GlueCEInfoLRMSType
```

- “simple” authentication
- Address of LDAP server (GRIS or GIIS)
- base distinguished name for search filter
- attributes to be returned

LDAP Browsers

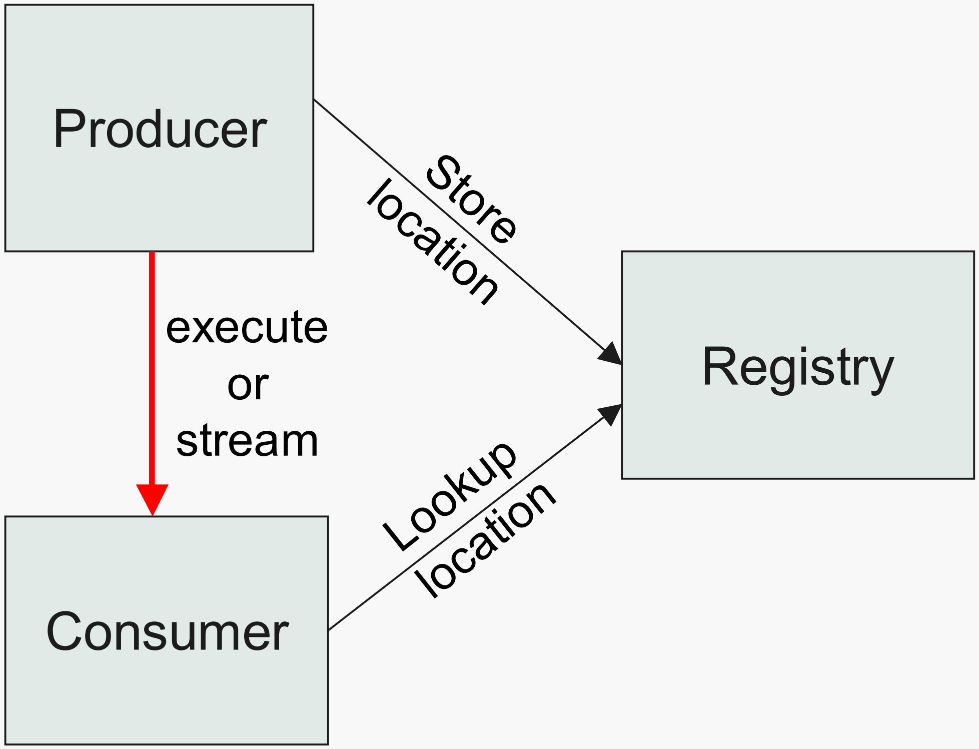


```
Mds-Vo-Name=cc-in2p3,o=grid
  seId=ccgridli07.in2p3.fr
    seProtocol=gridftp
    seProtocol=rfio
    in=status
  Mds-Host-hn=ccgridli07.in2p3.fr
    Mds-Device-Group-name=processors
    Mds-Device-Group-name=memory
    Mds-Device-Group-name=filesystems
    Mds-Device-Group-name=networks
    Mds-Software-deployment=operating system
  hn=ccgridli08.in2p3.fr
  Mds-Host-hn=ccgridli08.in2p3.fr
  in=siteinfo
  hn=ccgridli03.in2p3.fr
    ceId=ccgridli03.in2p3.fr:2119\/jobmanager-bqs-A
    ceId=ccgridli03.in2p3.fr:2119\/jobmanager-bqs-C
    ceId=ccgridli03.in2p3.fr:2119\/jobmanager-bqs-G
    ceId=ccgridli03.in2p3.fr:2119\/jobmanager-bqs-I
    ceId=ccgridli03.in2p3.fr:2119\/jobmanager-bqs-S
    ceId=ccgridli03.in2p3.fr:2119\/jobmanager-bqs-T
    ldap://ccgridli03.in2p3.fr:2135/Mds-Host-hn=ccgridli03.
  Mds-Host-hn=ccgridli03.in2p3.fr
```

Attribute	Value
userSupportContact	user.support@in2p3.fr
sysAdminContact	edg-site-admin@cc.in2p3.fr
dataGridVersion	edg1_4_3
siteName	CC-IN2P3
installationDate	20030114122900Z
objectClass	SiteInfo
objectClass	DataGridTop
objectClass	DynamicObject
siteSecurityContact	sysunix@in2p3.fr

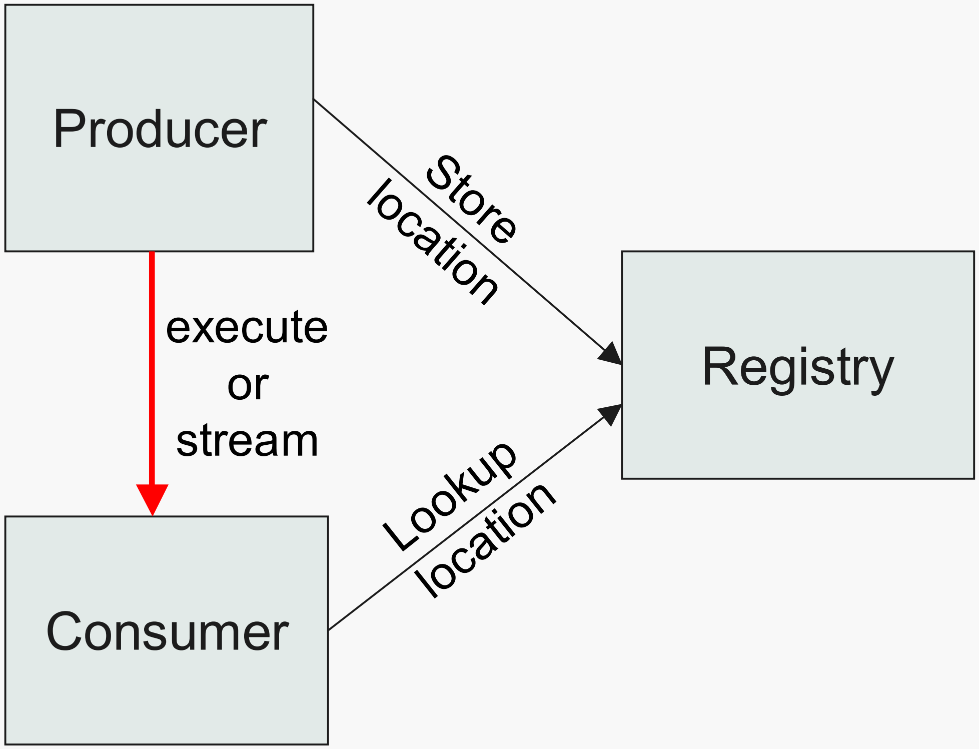
Ready. 7 entries returned.

GMA



- From GGF
- Very simple model
- Does not define:
  - Data model
  - Data transfer mechanism
  - Registry implementation

R-GMA



- Use the GMA from GGF
- A relational implementation
  - Powerful data model and query language
    - All data modelled as tables
    - SQL can express most queries in one expression
- Applied to both information and monitoring
- Creates impression that you have one RDBMS per VO

Relational Data Model in R-GMA

- Not a general distributed RDBMS system, but a way to use the relational model in a distributed environment where global consistency is not important
- Producers announce: SQL “CREATE TABLE”
  publish: SQL “INSERT”
- Consumers collect: SQL “SELECT”

Example: 2 tables

Service

URI	VARCHAR(255)	URI to contact the service
VO	VARCHAR(50)	Where info should be published – or an empty string to indicate all
type	VARCHAR(50)	Type of service
emailContact	VARCHAR(50)	The e-mail of a human being to complain to
site	VARCHAR(50)	Domain name of site hosting the service
secure	VARCHAR(1)	‘y’ or ‘n‘ - indicates whether or not this is a secure service
majorVersion	INT	Version of protocol not implementation
minorVersion	INT	Version of protocol not implementation
patchVersion	INT	Version of protocol not implementation

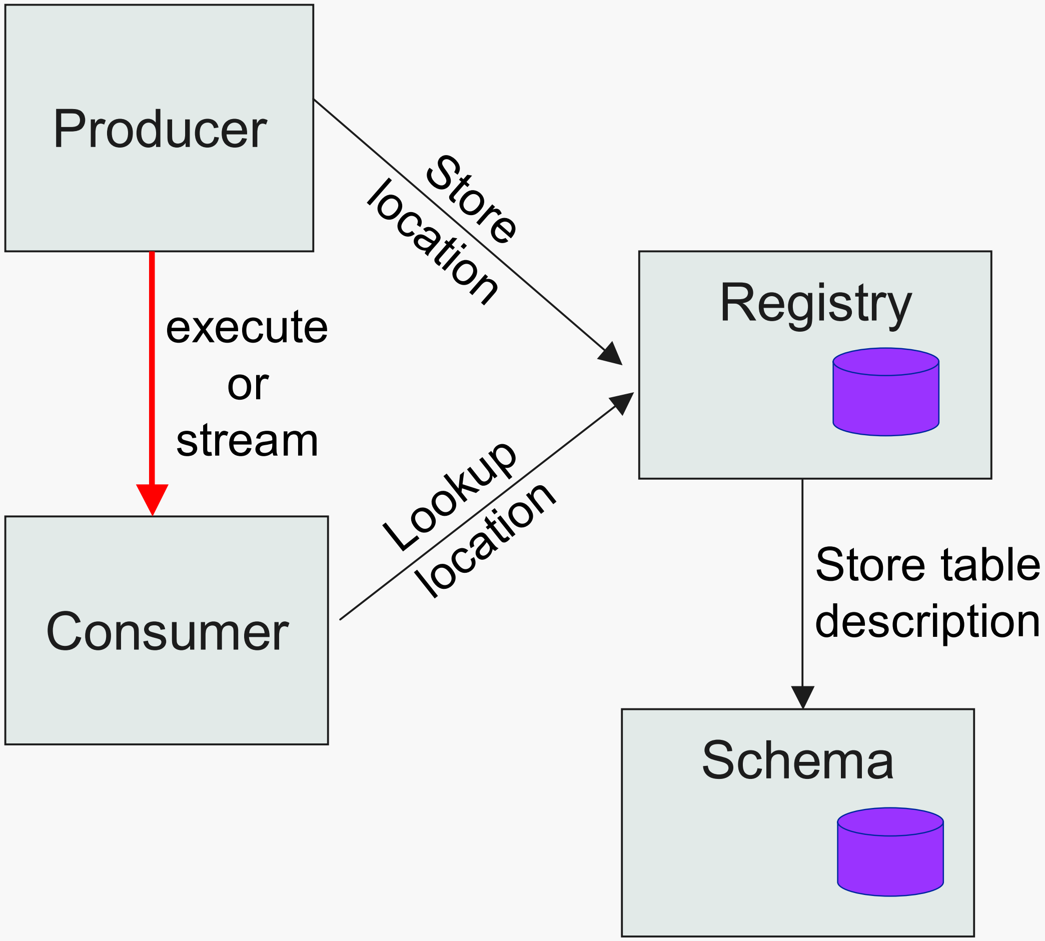
ServiceStatus

URI	VARCHAR(255)	URI to contact the service
status	INT	status code. 0 means the service is up.
message	VARCHAR(255)	Message corresponding to status code

Data Transfer: Producer ➡ Consumer

- Consumer can issue one-off queries
  - Similar to normal database query
- Consumer can also start a continuous query
  - Requests all data published which matches the query
    - As new data matching the query is produced it is streamed to the Consumer
    - Can be seen as an alert mechanism

Registry and Schema



- Registry has two main tables:
  - Producer
    - Table name
    - Predicate
    - Location
  - Consumer
    - Query
    - Location
- Schema holds description of tables
  - Column names and types of each table

Mediator

- The Mediator must:
  - find the right Producers
  - combine information from them
- Hidden component – but vital to R-GMA
- Will eventually support full distributed queries but for now will only merge information
  - from multiple producers for queries on one table
  - or over multiple tables from one producer

Archiver (Re-publisher)

- It is a combined Consumer-Producer
  - Follows the GMA concept but packaged for ease of use
- You just have to tell it what to collect and it does so on your behalf

More info

- Glue Schema
  - http://www.hicb.org/glue/glue-schema/schema.htm
  - http://www.cnaf.infn.it/~sergio/datatag/glue/
- Globus MDS
  - http://www.globus.org/mds
- EDG-RGMA
  - http://www.r-gma.org