

Information Models for Grid Resources

International Summer
School on Grid Computing
21 July 2003, Vico Equense (NA)

Sergio Andreozzi
INFN-CNAF
Bologna (Italy)

sergio.andreozzi@cnafe.infn.it



OUTLINE

- I. Introduction to Information Modeling
- II. GLUE Schema
 - The information model
 - Computing Service model
 - Storage Manager Service model
 - Network Service model
 - Mapping to actual data models
- III. Common Information Model (CIM)
 - Introduction
 - Grid related activity

Information Model: definition

- ✎ Abstraction of real world into constructs that can be represented in computer systems (e.g., objects, properties, behavior, and relationships)
- ✎ Not tied to any particular implementation
- ✎ Used to exchange information among different domains

Information Model: why it is important

- ☞ it allows multiple experts to contribute to the problem description (e.g., scheduling experts and networking experts to work on models for service selection based on network conditions)
- ☞ it serves as a communication mean between domain experts (e.g., among replica data service developers and storage manager service developers)

Information Model: how can be represented

- Typically, graphical languages are preferred
- Several solutions are available
- We have selected the Unified Modeling Language (UML)
 - ✓ It is a widely accepted international standard (Object Management Group, OMG)
 - ✓ It is often used for information and conceptual modeling
 - ✓ It has become well established in many communities with extensive tool support from both commercial and open source vendors

Unified Modeling Language (UML)

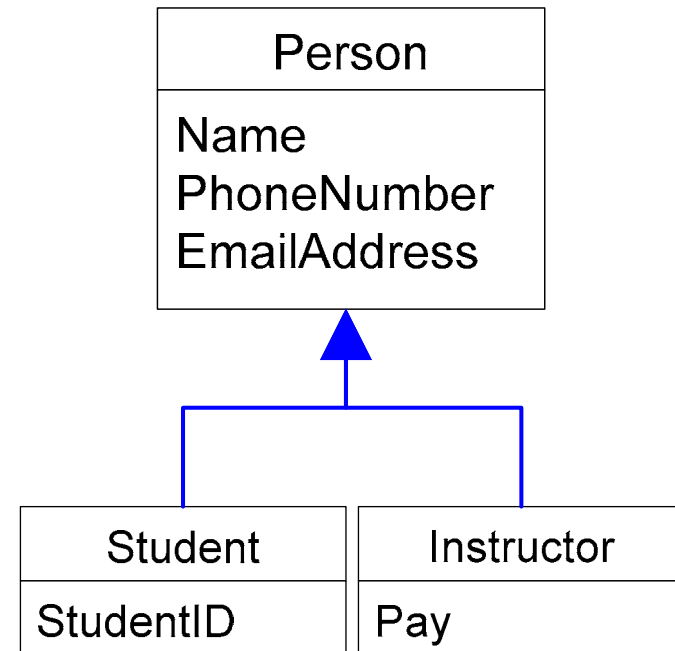
- ✓ The Unified Modeling Language (UML) is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system.
- ✓ The UML offers a standard way to write a system's blueprints, including conceptual things such as business processes and system functions as well as concrete things such as programming language statements, database schemas, and reusable software components.

Unified Modeling Language

- First Specification in 1997
- Current Specification version 1.5 (12 different diagrams)
- Finalizing Specification version 2.0 (13 different diagrams)
- Each diagram type has:
 - Semantics: what does the diagram type do?
 - Notation: what graphical symbols can the diagram type contain?
- Diagram groups:
 - Structural: model the static aspects of a system
 - Behavioral: model the behavior of a system (dynamic model)
- We use **Class diagrams**: they show the static structure of the model, in particular, the things that exist (such as classes and types), their internal structure, and their relationships to other things

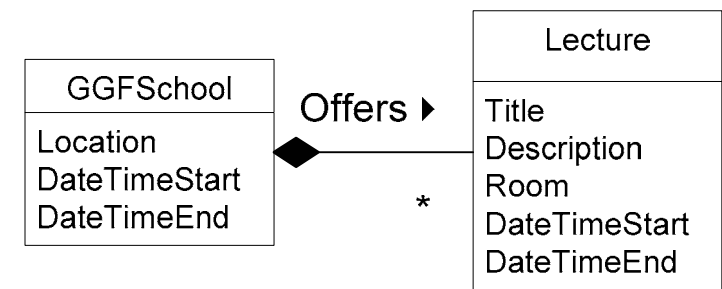
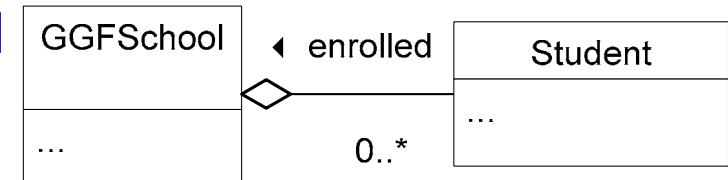
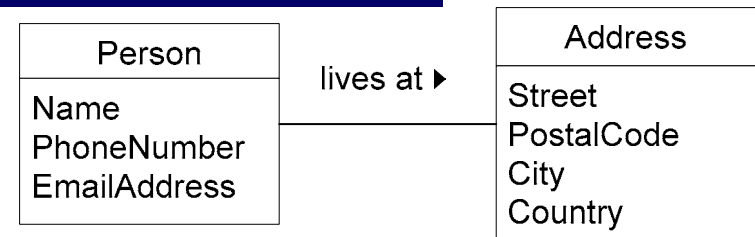
UML Class Diagram elements

- **Class** represents a concept within the system being modeled. It has data structure, behavior and relationships to other elements
- **Generalization:** taxonomic relationship between a more general element (the parent) and a more specific element (the child) that is fully consistent with the first element and that adds additional information. It is used for classes, packages, use cases, and other elements



UML Class Diagram elements

- **Binary association:** an association among exactly two classes (maybe also from a class symbol to itself)
- **Aggregation:** it denotes weak ownership (i.e., the part may be included in several aggregates) and its owner may also change over time. Deleting the aggregate referencing does not imply deletion of the parts
- **Composition:** strong form of aggregation; a part instance may be included in at most one composite at a time; the composite object has sole responsibility for the disposition of its parts



PART II

The GLUE Schema

GLUE:

WHAT

- GLUE: **G**rid **L**aboratory **U**niform **E**nvironment
- 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

GLUE: WHO and WHEN

- Promoted by DataTAG (EU) and iVDGL (US)
- Contributions from DataGrid, Globus, PPDG and GriPhyn
- GLUE Schema activity is started in April 2002
- ✓ A common information model for Grid resources is one of the main tasks

GLUE Schema overview

- Focus on modelling Grid resources
- In particular, we concentrate on all those resources that participate in the Grid system and that are requested to be discoverable and monitored
- Final goal: **produce schema for available Grid Information Services (GIS)**
- If concepts and relationships are properly modelled, the same information can be retrieved from different GISs relying on different technology (e.g. R-GMA, MDS 2)

Grid Information Services

	MDS 2.x	R-GMA	MDS 3.x
DATA Model	LDAP	Relational	XML
Communication	LDAP	HTTP	Independent
Storage	in-memory	In-memory or DBMS	in-memory or xIndice (XML DB)
Query language	LDAP	SQL	XPath (later XQuery, XSLT)
Resource Information Service	GRIS	Producer	service
Aggregation Service	GIIS	Archiver	Index Service
Directory Service	GIIS	Registry	Index Service

GLUE Schema: modeling guidelines

- ☞ Clear separation between system and service entities
 - System: a set of connected items or devices which operate together as a functional whole
 - Service: actions that form a coherent whole from the point of view of service providers and service requesters
- ☞ Generalization
 - capture common aspects for different entities providing the same functionality (e.g. uniform view over different batch services)
- ☞ Deal with both monitoring needs and discovery needs
 - Monitoring: concerns those attributes that are meaningful to describe the status of resources (e.g., useful to detect fault situation)
 - Discovery: concerns those attributes that are meaningful for locate resources on the base of a set of preferences/constraints (e.g., useful during matchmaking process)

GLUE Computing resources: warm up

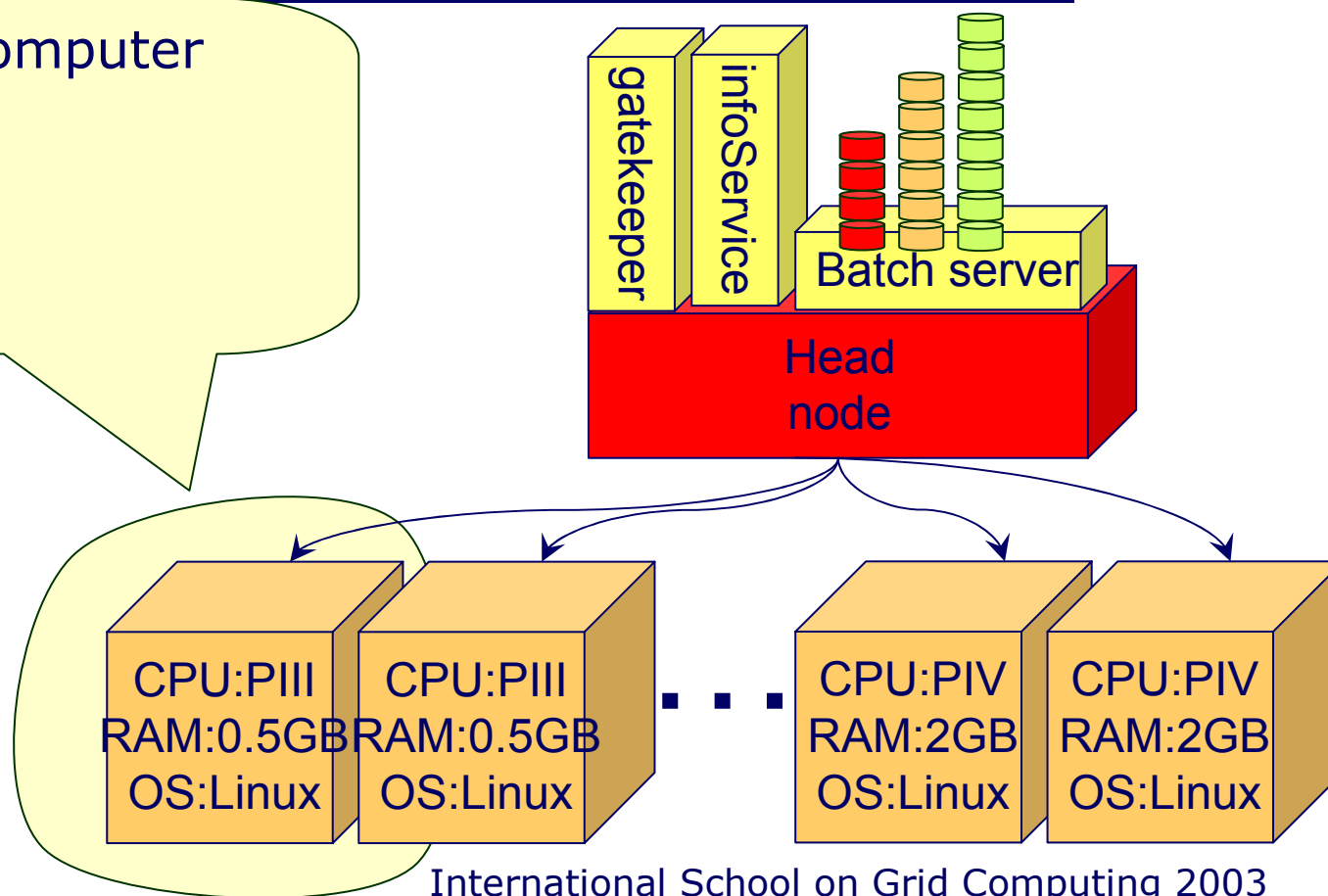
- ☞ What is the core offered functionality?
 - ☞ Computing power
- ☞ What I need to know in order to use it?
 - ☞ Offered execution environment (e.g., OS type, available software libraries)
 - ☞ Offered Quality of Service (e.g., estimated response time)
 - ☞ Status (e.g., number of running jobs)
 - ☞ 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

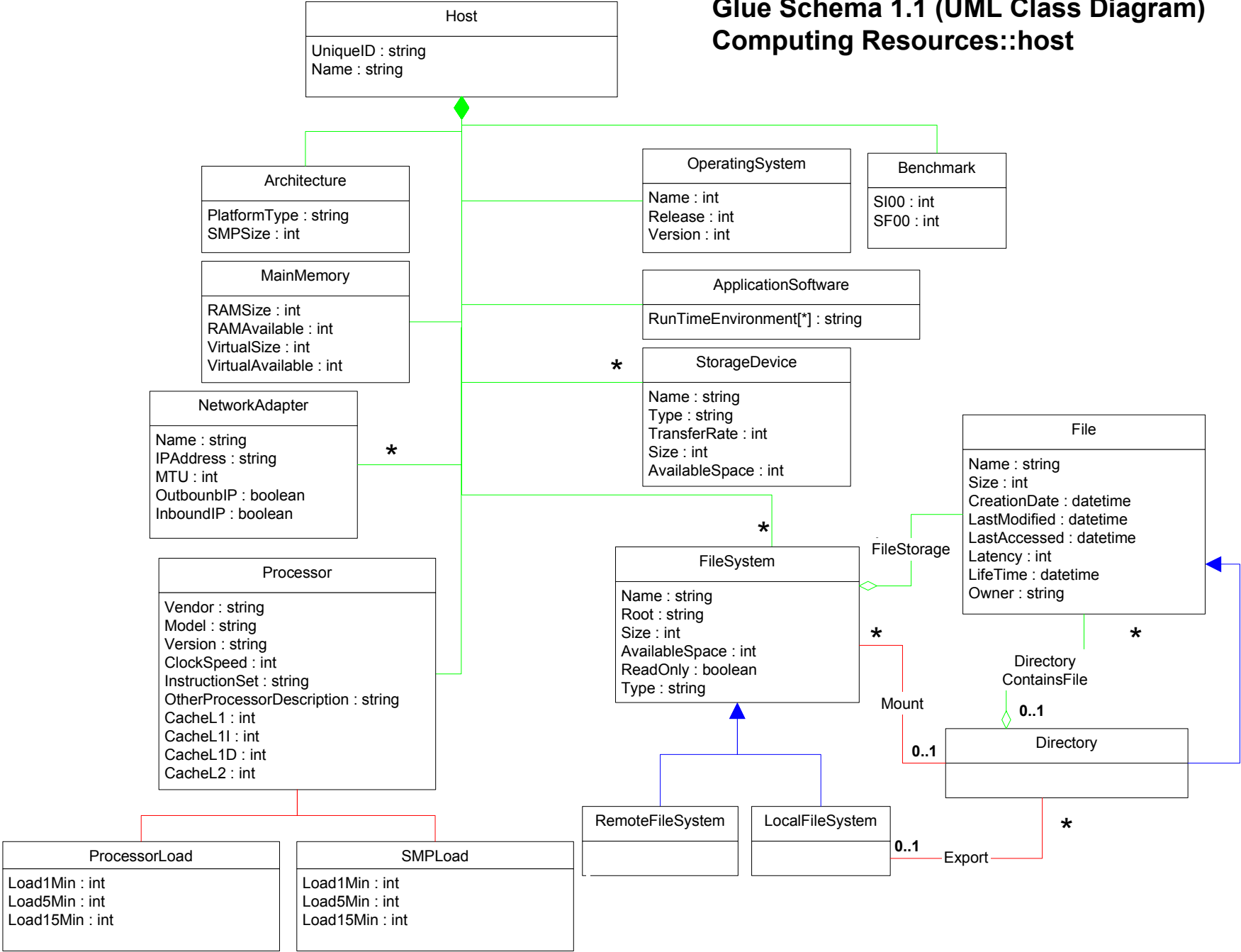
GLUE Computing resources: Host (the system)

Host: a single computer system



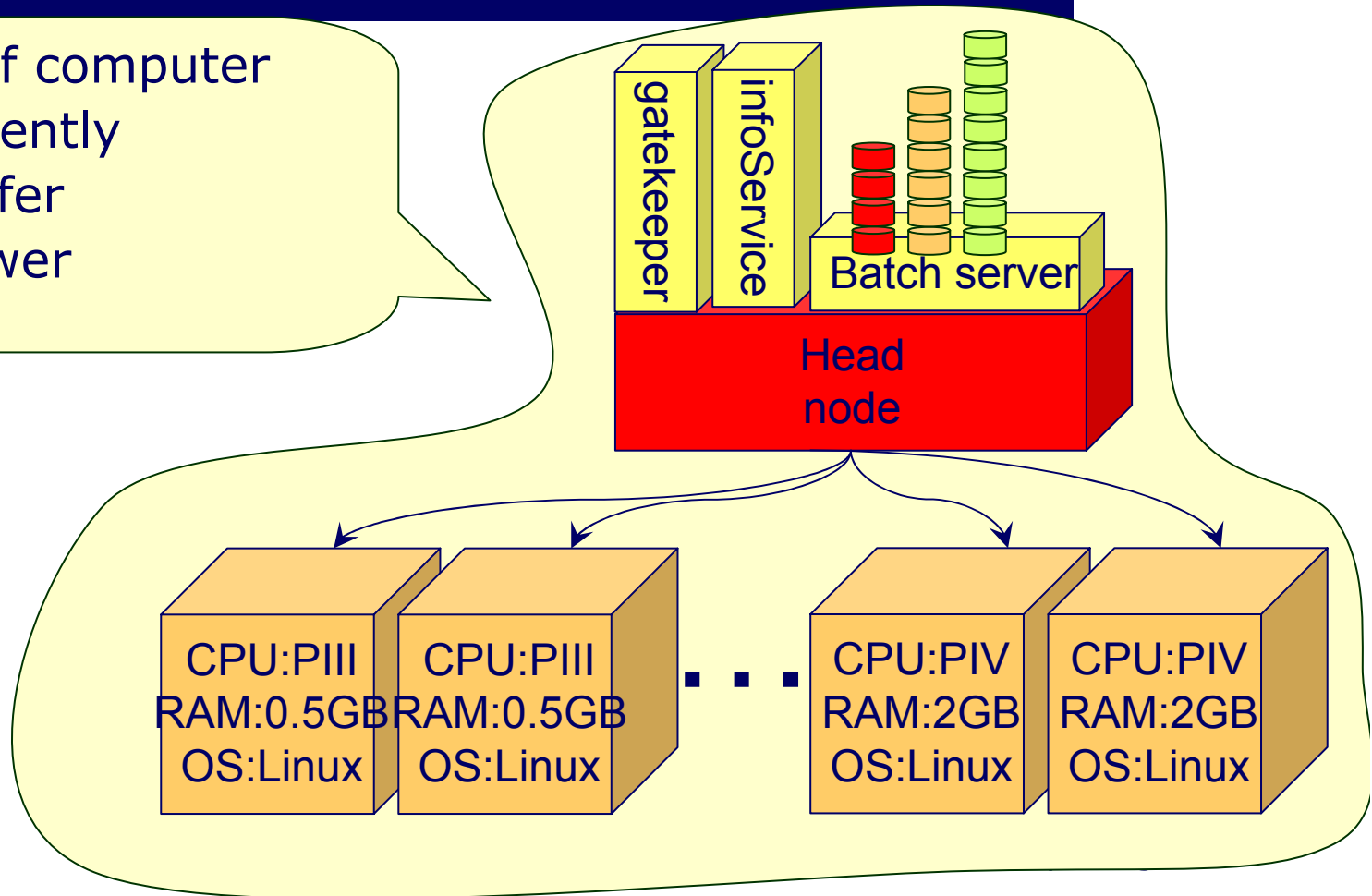
Glue Schema 1.1 (UML Class Diagram)

Computing Resources::host



GLUE Computing resources: Cluster (the system)

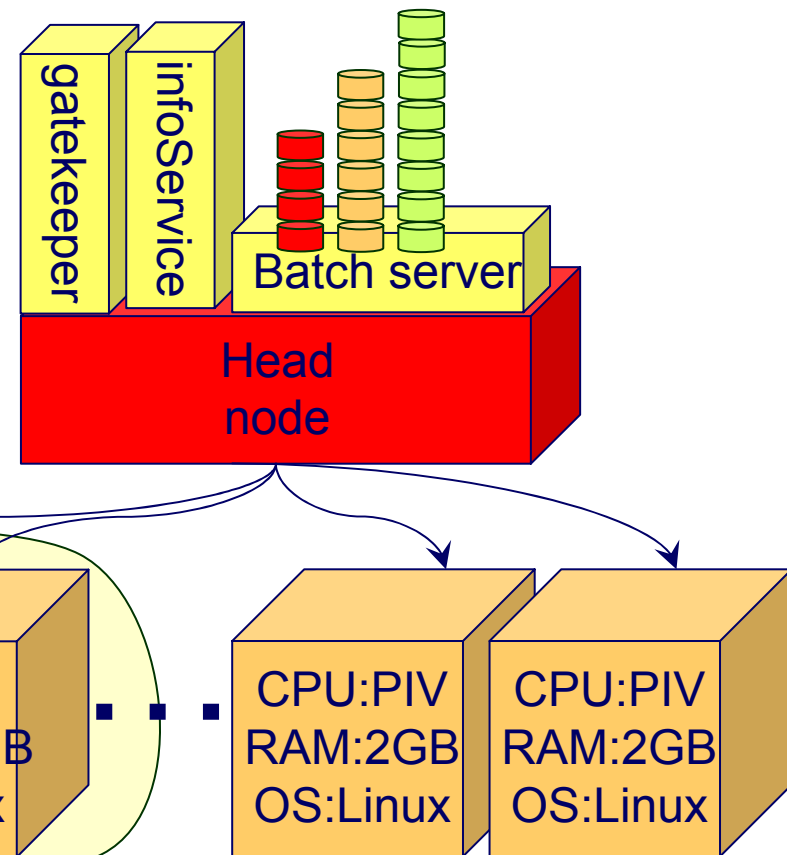
Cluster: set of computer systems coherently
Managed to offer
computing power



GLUE Computing resources: SubCluster (aggregate information)

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

- Hosts are homogeneous if they have the same values for the given set of attributes
- e.g. CPUType, RAMSize, OSType



2
1

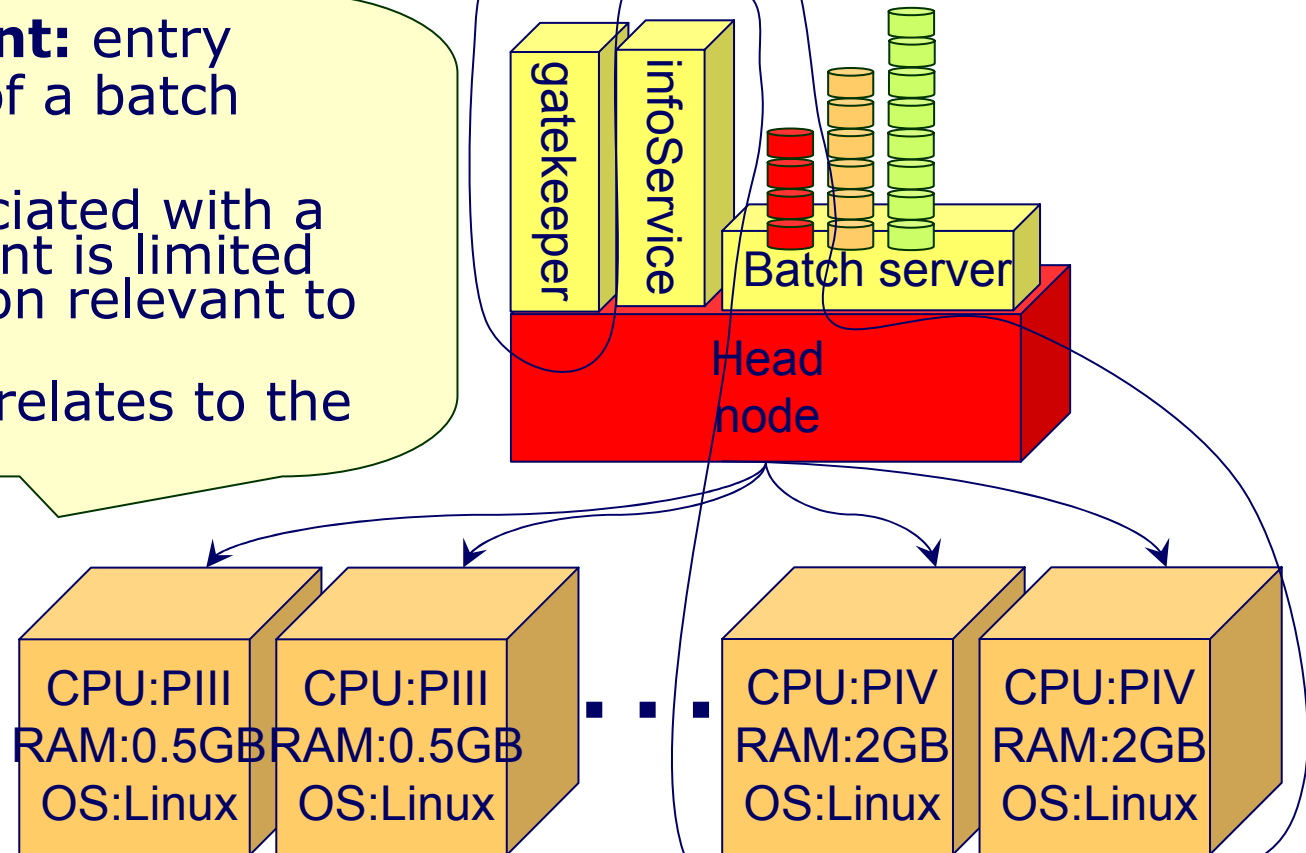


Number of nodes
maybe $O(1000)$

GLUE Computing resources: Computing Element (the Service)

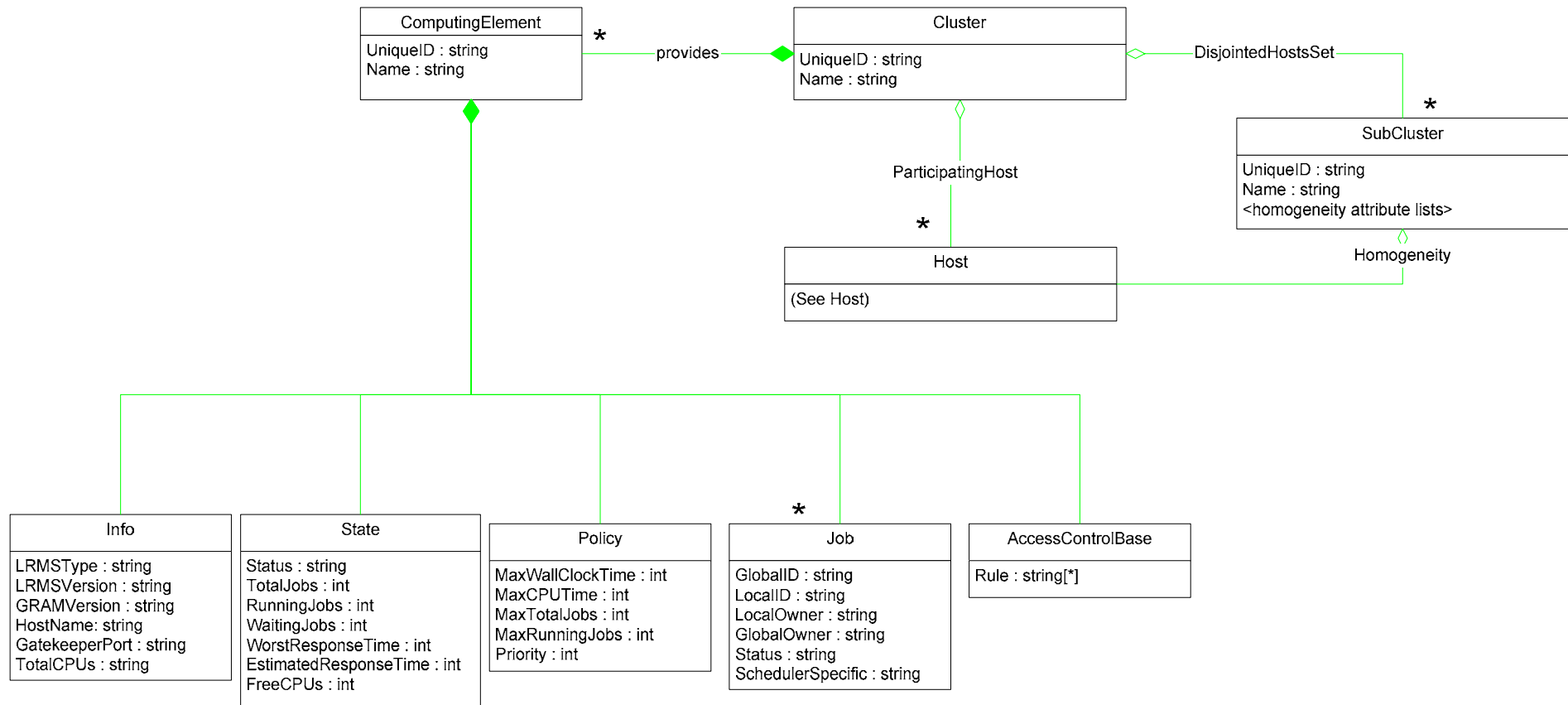
Computing Element: entry point into a queue of a batch system

- 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



GLUE Computing resources

open issue: CE viewpoint of the cluster

- ☞ The SubCluster concept relates only to host and cluster concept, does not take into account resources assigned to the queue
- ☞ Given a set of hosts assigned to a queue, the decision of the actual one that will execute a job is up to the local batch system
- ☞ From the service requester viewpoint, only a specific computing element can be selected (any of the available hosts can be a valid candidate for a job)
- ☞ **We need an aggregate description of the resources assigned to each computing element**
- ☞ Current practice within DataGrid: homogenous clusters!!!
- ☞ Needs for refinement in the near future

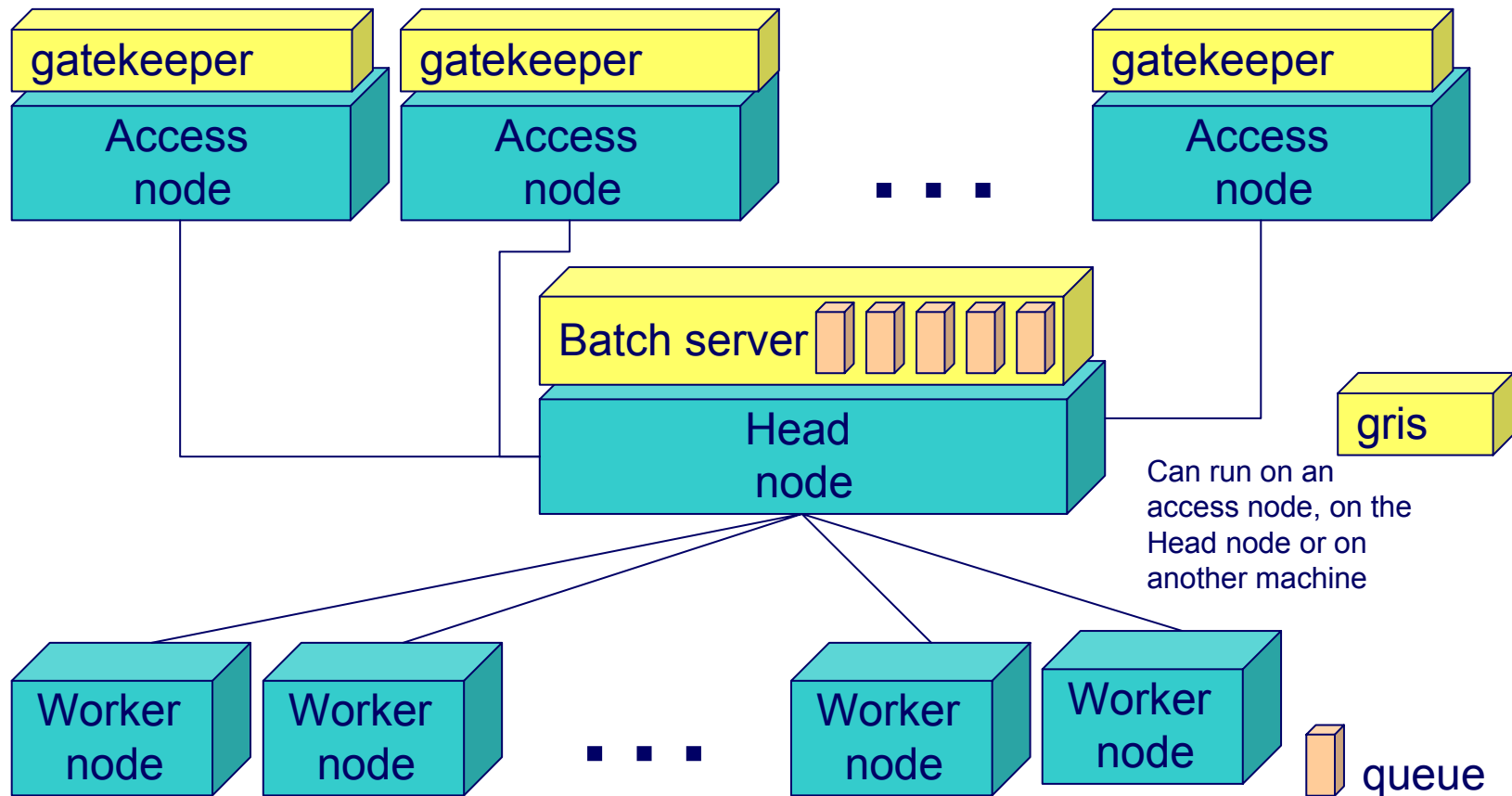
GLUE Computing resources

open issue: multiple entry points to a queue

- ☞ Computing Element relates to **an entry point to a queue**
- ☞ For scalability issues, big clusters can have more several gatekeepers in front of them
- ☞ Need for refine computing element concept (association to a queue besides its entry point; the set of entry points is a property)

GLUE Computing resources

open issue: multiple entry points to a queue



GLUE Storage resources: warm up

- ☞ 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)

GLUE

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 (e.g., in SRMv2, volatile, permanent and durable)
- at present is a generalization for different storage service types

GLUE

Storage Service/Space/Library

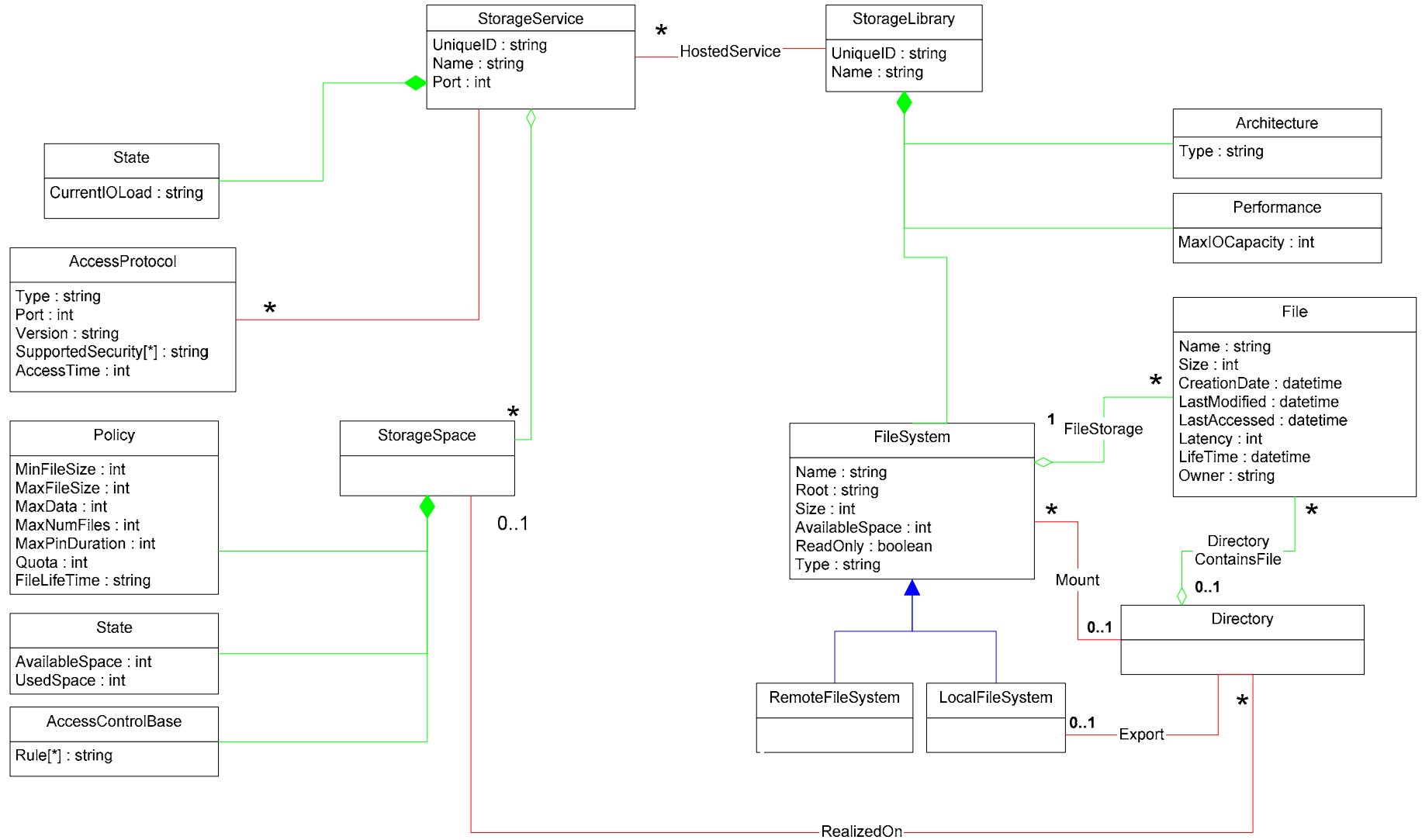
- **Storage Space:** portion of a logical storage extent that:
 - is **assigned to a Virtual Organization**
 - is **associated to a directory** of the underlying file system (e.g. /permanent/CMS)
 - has a **set of policies** (MaxFileSize, MinFileSize, MaxData, MaxNumFiles, MaxPinDuration, Quota)
 - has 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)

GLUE

Storage Service/Space/Library

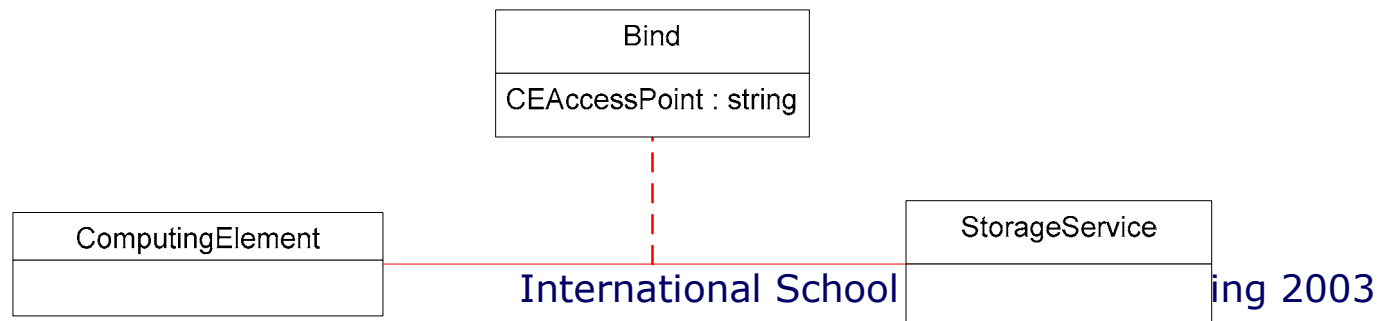
- **Storage Library:** the machine providing for both storage space and storage service
 - ☞ The system entity for storage resources is not yet well modeled since clear requirements are missing;
 - ☞ A storage system can vary from a simple disk server to complex hierarchical storage systems;
 - ☞ Two possible evolutions for the storage library concept are:
 1. It will be the edge machine of the storage system (e.g., to monitor the execution environment of the storage service)
 2. It will model the whole storage system complexity

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
- SiteAdmins may want to specify preferences on which Storage Spaces should be used by jobs running on certain computing services
- The possibility of expressing such preference is modelled by (GLUE CE-SE Bind concept)
- CE Access point refer to an eventual NFS mountpoint



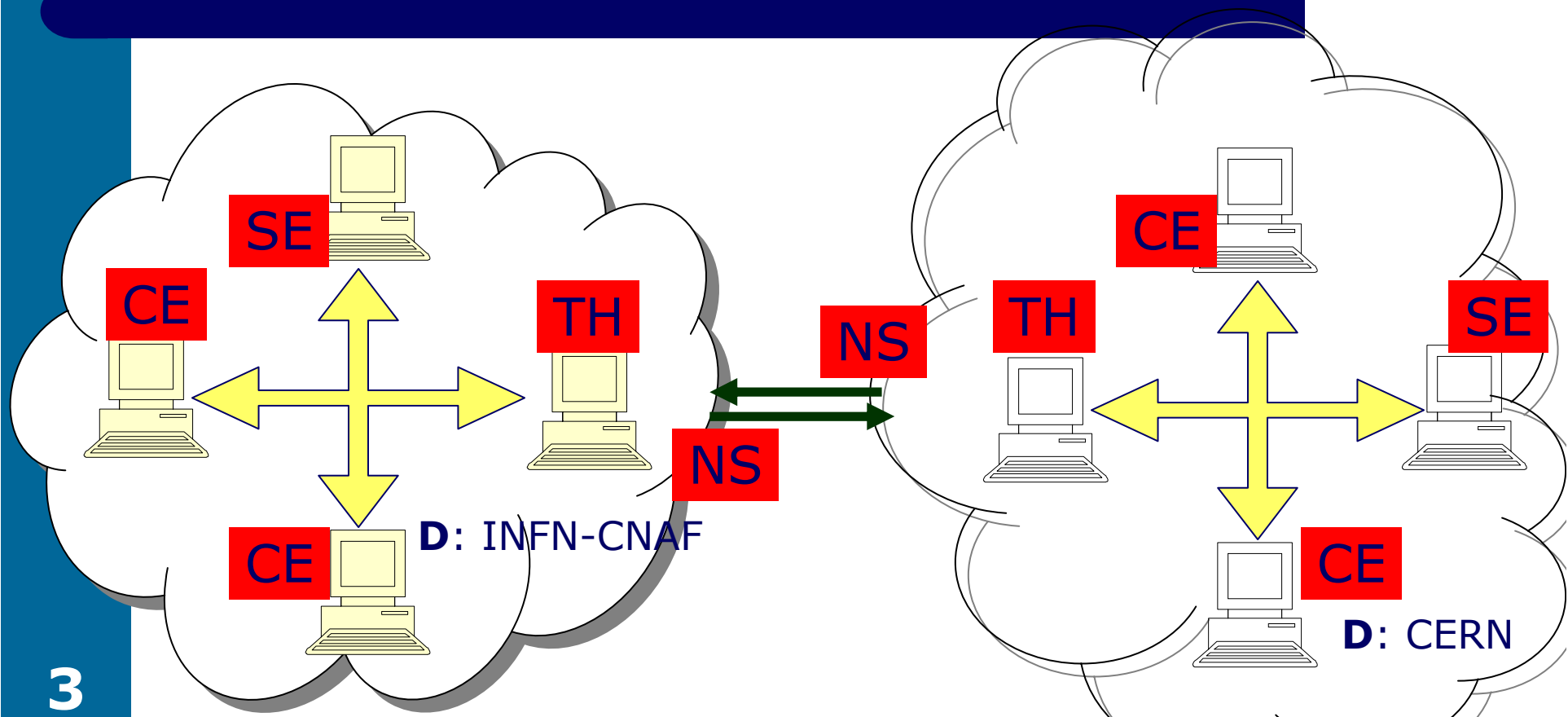
GLUE Network Resources

- Work in Progress:
 - 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.

Partitioning the Grid into Domains

- A **Domain** is a set of elements identified by URI's (referred in the model as **edge services**)
- **Connectivity** is a metric that reflects the quality of communication through a link between two Edge Services
- A **Domain** communicates with other domains using Network Services
- A **Network Service** offers a unidirectional communication service between two Domains
- Each domain has a **Theodolite Service** that gather network service related metrics towards others domains

GLUE Network Service example scenario



GLUE: Conceptual model status

- version 1.0
 - ☑ Finalized in Oct '02
 - ☑ Model of computing resources
 - ☑ Model of storage resources
 - ☑ Model relationships among them
- Version 1.1
 - ☑ Finalized in Mar '03
 - ☑ Some fix
 - ☑ Model of network resources

GLUE: Implementation Status

- **Implementation status - version 1.1**

- For Globus MDS 2.x (part of GT 2.x):
 - ☑ LDAP Schema (DataTAG WP4)
 - ☑ Info providers both computing (EDG WP4, valid for PBS, LSF and Condor) and storage resources (EDG WP5, valid for trivial file system and edg-se)
- For EDG R-GMA:
 - ☑ Relational schema (EDG WP3)
 - ☑ Info providers for computing and storage resources translate output of LDAP info provider in a suitable format to be stored in the relational model (EDG WP3)
 - ☑ Info providers for network resources (EDG WP7+DataTAG WP4)
- For Globus MDS 3.x (part of GT 3):
 - ☑ XML Schema for computing resources (Globus)
 - ☑ Info provider (Globus)

GLUE: Deployment Status

- **Included in:**
 - **DataGrid 2.0**
 - with mixed R-GMA/MDS2 scenario
 - **VDT 1.1.6 and later (MDS2)**
 - **LCG0 (MDS2)**
 - **LCG1 (MDS2 for the moment, will move to R-GMA)**
 - **Globus Toolkit 2.x**
 - as optional, only for computing resources
 - **Globus Toolkit 3**
 - as optional, only for computing resources

Future Work

- Computing:
 - Model the service viewpoint of the cluster to enable more flexibility in cluster configuration
 - Refine computing element definition to meet multiple entry points scenario
- Storage:
 - Several fix needed to the service part
 - Understand what we really need from the system part
- Network:
 - Experience, experience, experience
- High Level Grid Services modeling

PART III

Common Information Model (CIM) and the GRID

Common Information Model

- **CIM: Common Information Model**

- Conceptual view of the managed environment for IT resources that attempts to unify and extend the existing instrumentation and management standards
- Targeted at **management of resources**, where management is defined as the active process of monitoring, modifying, and making decisions about a resource
- Maintained by Distributed Management Task Force (DMTF), a worldwide industry organization
- It uses UML Class Diagram as a modeling language

CIM related activities at GGF

- CIM Grid Schema WG (CGS WG)
 - Started at GGF 5
 - Goal: define CIM extensions for the Job Submission Service Model, i.e.:
'managed objects and their relationships for managing the execution and monitoring of batch jobs in a grid environment'
 - Defined extensions will be submitted to DMTF for inclusion in the official CIM standard
- Common Resource Model WG (CRM WG)
 - BOF at GGF 7
 - Goal: define CIM extensions to describe manageable resources as OGSA services

REFERENCES

- GLUE Schema Official documents
<http://www.cnaf.infn.it/~sergio/datatag/glue>
- S. Andreatto, M. Sgaravatto, C. Vistoli, "Sharing a conceptual model of grid resources and services", In Proceedings of CHEP 2003
<http://www.cnaf.infn.it/~sergio/publications/CHEP2003.pdf>
- S. Andreatto, "GLUE Schema implementation for the LDAP model", Technical report, first draft, 29/05/03
<http://www.cnaf.infn.it/~sergio/publications/Glue4LDAP.pdf>
- GGF CIM Grid Schema WG
<http://www.isi.edu/~flon/cgs-wg/>
- GGF Common Resource Model WG (BOF)
<http://www.gridforum.org/Meetings/ggf7/BOFS/CRM%20Working%20Group%20Home%20for%20BOF1.htm>