GLUE Schema:
Conceptual model and implementation

Sergio Andreozzi
INFN-CNAF
Bologna (Italy)

sergio.andreozzi@cnaf.infn.it

OUTLINE

- Short introduction to the GLUE activity
- GLUE Schema overview
  - The conceptual model
  - The implementation status
  - Deployment roadmap
- EDT-LCG Monitoring effort
- Open issues
- EDG WP2 needs vs. Glue schema

GLUE: WHAT

- GLUE: Grid Laboratory Uniform Environment
- collaboration effort focusing on interoperability between US and EU HEP Grid middlewares
- Targeted at core grid services
  - Resource Discovery and Monitoring
  - Authorization and Authentication
  - Data movement infrastructure
  - Common software deployment procedures
- Preserving coexistence for collective services

GLUE: WHO and WHEN

- Promoted by DataTAG and iVDGL
- Contributions from DataGRID, Globus, PPDG and GriPhyn

- Activity started in April 2002

GLUE Schema overview 1/3

- **Conceptual model** of **grid resources** to be used as a base schema of the Grid Information Service for **discovery** and **monitoring** purposes
- Based on the experience of DataGRID and Globus schema proposals
- Attempt to gain from CIM effort
  (and hopefully to contribute in the GGF CGS WG)

GLUE Schema overview 2/3

- Conceptual model – version 1.0
  - ☑ Finalized in Oct '02
  - ☑ Model of computing resources (Ref. CE)
  - ☑ Model of storage resources (Ref. SE)
  - ☑ Model of relationships among them (Ref. Close CE/SE)
- Currently working on version 1.1
  - ☑ Adjustements coming from experience
  - ☑ Extensions (e.g. EDG WP2 needs :-)
  - ☑ Model of network resources

GLUE Schema overview 3/3

- **Implementation status – version 1.0**
  - For Globus MDS:
    - ☑ LDAP Schema (DataTAG WP 4.1)
    - ☑ Info providers both computing and storage resources
    - ☑ Ongoing work for monitorin extensions
  - For EDG R-GMA:
    - ☑ Relational model
  - For Globus OGSA:
    - ☑ XML Schema

Computing Resources

- **Globus schema**: representing canonical entities such as host and its component parts (e.g. file system, operating system, CPU, disk)
  - ✓ Host detailed info (good for monitoring)
  - × No concept of cluster, batch system, queue viewpoint of cluster
- **EDG schema**: Computing Element (CE) as abstraction for any computer fabric.
  - ✓ Driven from Resource Broker needs, it takes into consideration concepts such us:
    - ✓ batch computing systems, batch queues, cluster from the queue viewpoint
    - ✓ service relationships for discovery purposes (close CE/SE)
  - × Too wide concept to model both services and device that implement it
    - × Often, I heard asking: “What do you mean for CE, the batch queue or the cluster head node?”
  - × Lack in monitoring (no detailed info for hosts)
  - × *Close* relationship implementation (static and not really *close*)

GLUE Computing resources: assumptions and requirements

- In HEP area, clusters are usually composed by same kind of computers
- Separation between services and resources that implement it
- Needs for both detailed host info (monitoring issue) and aggregate view (discovery issue)

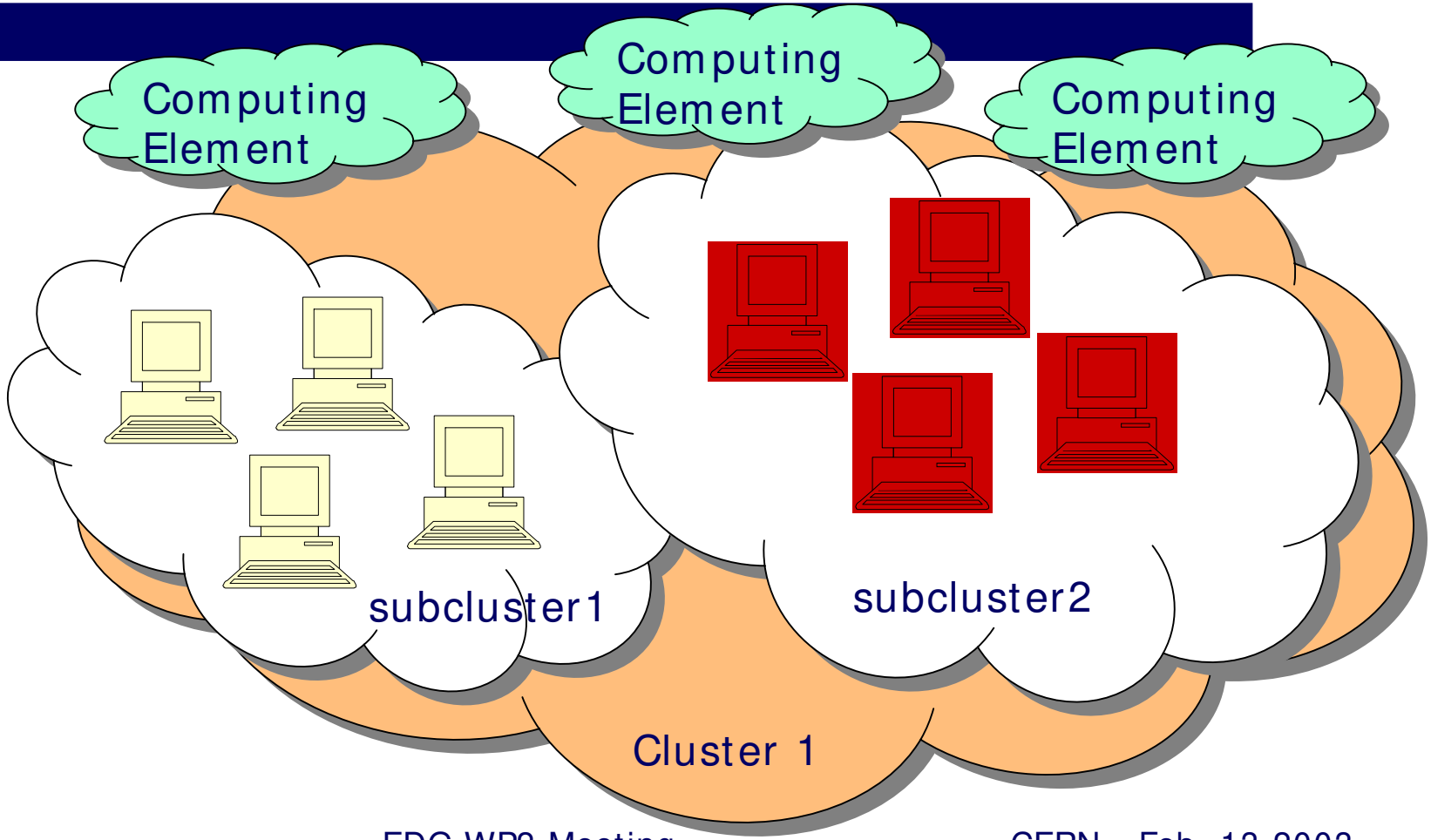
GLUE
Computing Element

- **Computing Element**: entry point into a queuing system
  - There is one computing element per queue
  - Queuing systems with multiple queues are represented by creating one computing element per queue
  - The information associated with a computing element is limited only to information relevant to the queue
  - All information about the physical resources access by a queue are represented by the Cluster information element

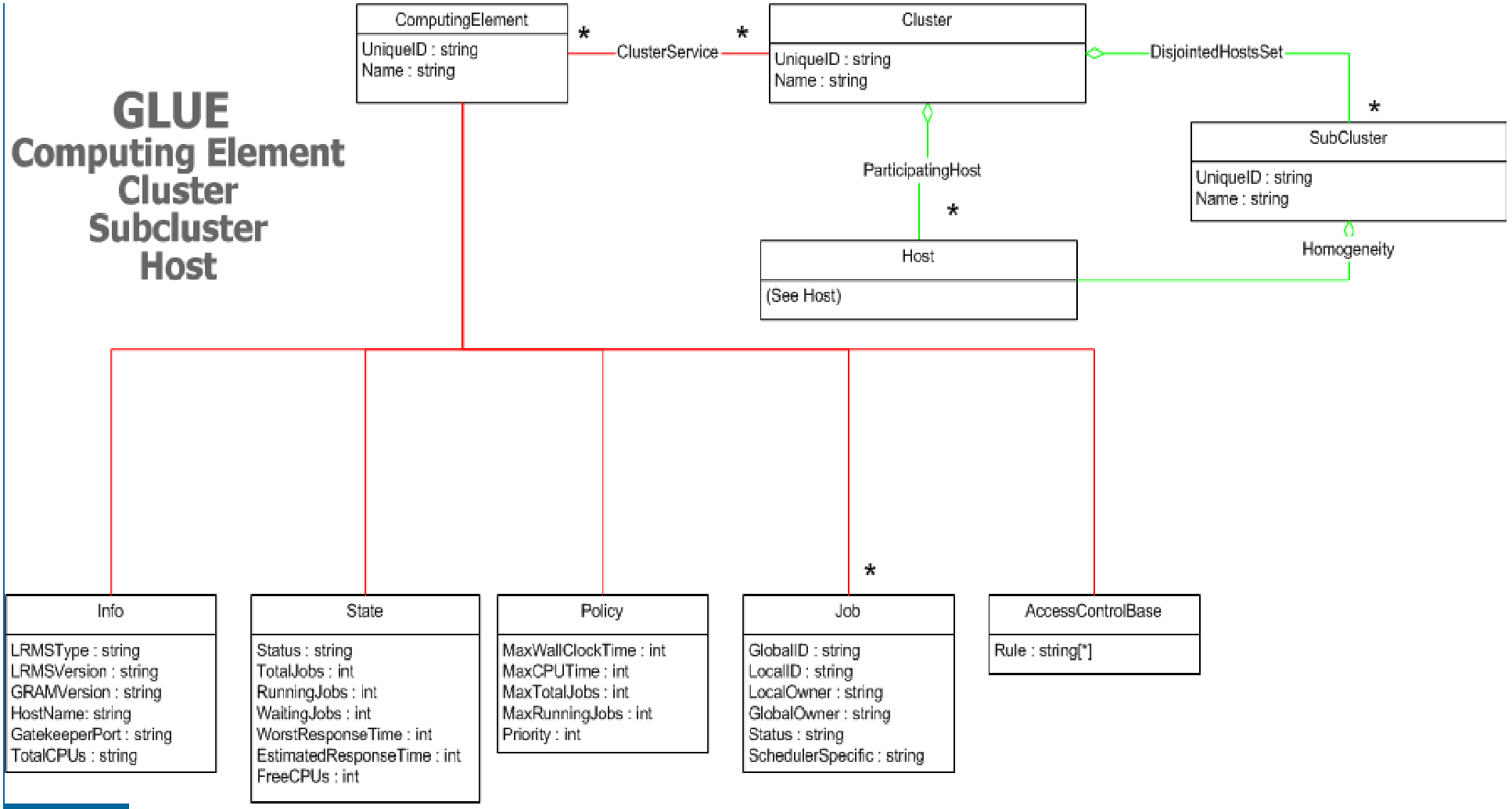
GLUE
Cluster/ Subcluster/ Host

- **Cluster**: container that groups together subclusters or nodes. A cluster may be referenced by more then one computing element.
- **Subcluster**: "homogeneous" collection of nodes, where the homogeneity is defined by a collection whose required node attributes all have the same value. A subcluster captures a node count and the set of attributes for which homogeneous values are being asserted.
- **Host**: characterizes the configuration of a computing node (e.g. processor, main memory, software)

Computing Resources in GLUE

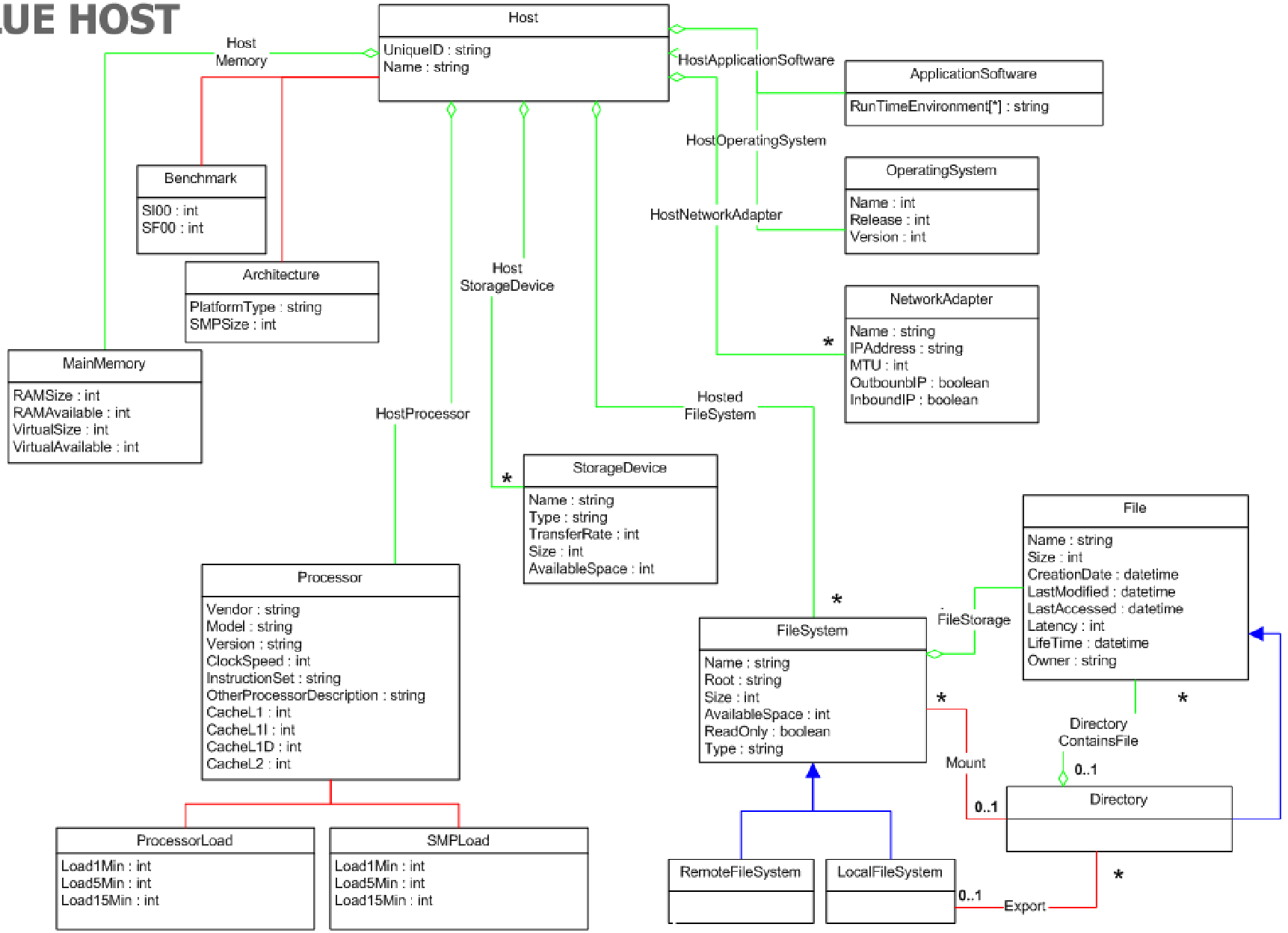


GLUE
Computing Element
Cluster
Subcluster
Host



* UML Class diagram slightly different than agreed in Glue Schema 1.0

GLUE HOST


Host
UniqueID : string
Name : string
Host Memory
HostApplicationSoftware
ApplicationSoftware
RunTimeEnvironment[*] : string
HostOperatingSystem
OperatingSystem
Name : int
Release : int
Version : int
HostNetworkAdapter
Benchmark
SI00 : int
SF00 : int
Architecture
PlatformType : string
SMPSize : int
Host StorageDevice
NetworkAdapter
Name : string
IPAddress : string
MTU : int
OutbounbIP : boolean
InboundIP : boolean
MainMemory
RAMSize : int
RAMAvailable : int
VirtualSize : int
VirtualAvailable : int
HostProcessor
Hosted FileSystem
StorageDevice
Name : string
Type : string
TransferRate : int
Size : int
AvailableSpace : int
File
Name : string
Size : int
CreationDate : datetime
LastModified : datetime
LastAccessed : datetime
Latency : int
LifeTime : datetime
Owner : string
Processor
Vendor : string
Model : string
Version : string
ClockSpeed : int
InstructionSet : string
OtherProcessorDescription : string
CacheL1 : int
CacheL1I : int
CacheL1D : int
CacheL2 : int
FileSystem
Name : string
Root : string
Size : int
AvailableSpace : int
ReadOnly : boolean
Type : string
FileStorage
Directory ContainsFile
Mount
Directory
ProcessorLoad
Load1Min : int
Load5Min : int
Load15Min : int
SMPLoad
Load1Min : int
Load5Min : int
Load15Min : int
RemoteFileSystem
LocalFileSystem
Export

Computing Resources in GLUE: comments

- Does this model fulfills EDG WP1 requirements?
  - Yes, but not in a clean way (...in my opinion)
- Why?
  - Subclusters describe “homogeneous” subset of hosts, independently from the queue
  - For discovery purpose, the broker needs an aggregate view of the resources from the queue viewpoint
  - Even though I have a homogeneous knowledge of the cluster, I cannot force the job to run on a certain subcluster (if the queue can submit to all nodes)
- Current practice/constraint:
  - only one subcluster per cluster

Storage Resources

- **EDG Schema:**
  - **Storage Element** (SE) as abstraction for any storage system (e.g. a mass storage system or a disk pool).
  - It provides Grid users with storage capacity.
  - The amount of storage capacity available for Grid jobs varies over time depending on local storage management policies that are enforced on top of the Grid policies.

GLUE
Storage Service/Space/Library

- **Storage Service:**
  - grid service **identified by a URI** that **manages disk and tape resources in term of Storage Spaces**
  - each Storage Space is **associated to a Virtual Organization** and a set of **VO-specific policies** (syntax and semantic of these to be defined)
  - all **hardware details** are **masked**
  - the Storage Service **performs file transfer** in or out of its Storage Spaces **using** a specified set of third part **data movement services** (e.g. GridFTP)
  - **files** are **managed** in respect of the lifetime policy specified for the Storage Space where they are kept; a specific date and time lifetime policy can be specified for each file and this is applied against a compatibility rules table

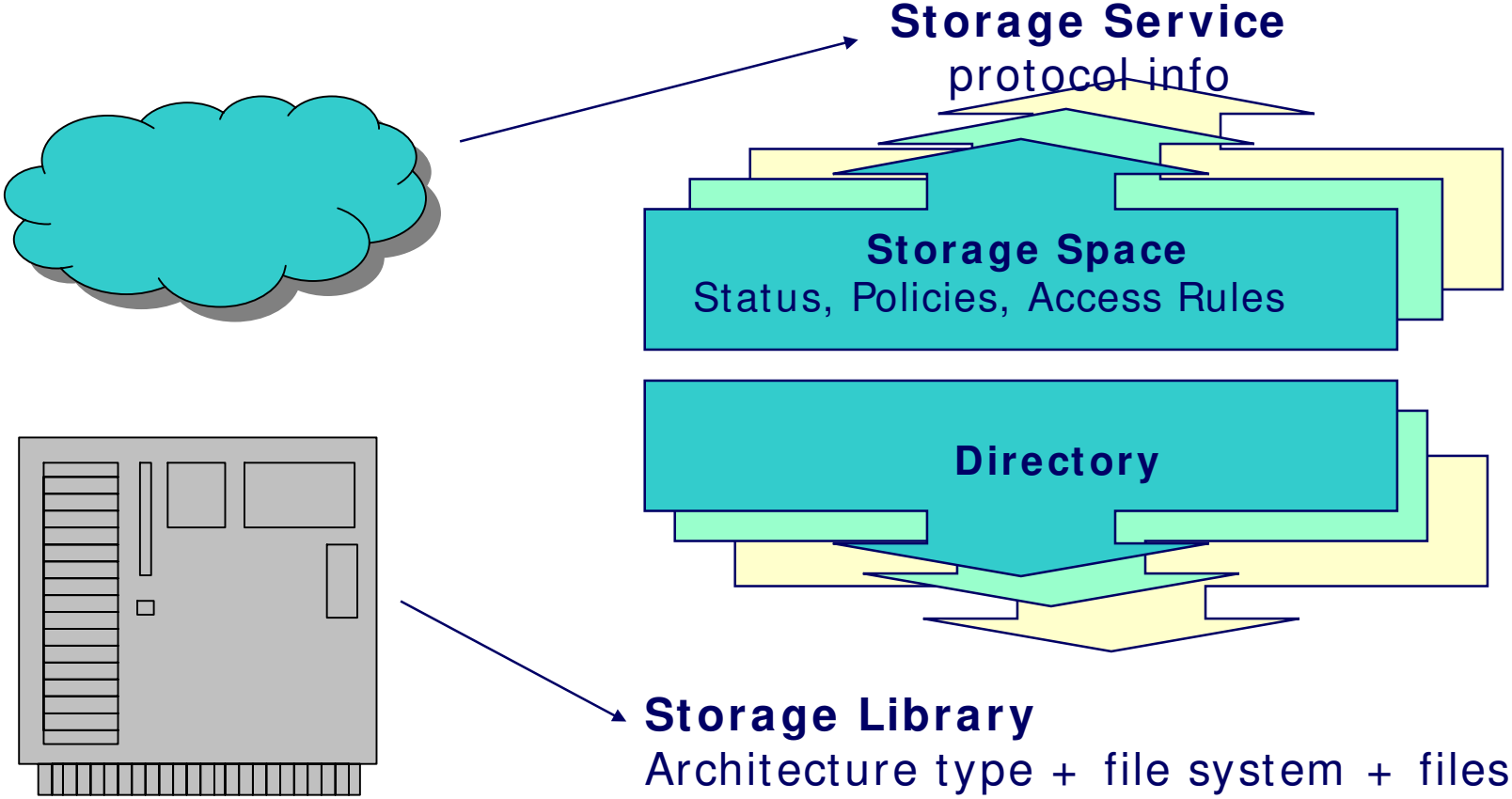
GLUE
Storage Service/**Space**/Library

- **Storage Space:** portion of a logical storage extent identified by:
  - an **association to a directory** of the underlying file system (e.g. /permanent/CMS)
  - a **set of policies** (MaxFileSize, MinFileSize, MaxData, MaxNumFiles, MaxPinDuration, Quota)
  - an association to **access control base rules** (to be used to publish rules to discover who can access what, syntax to be defined)

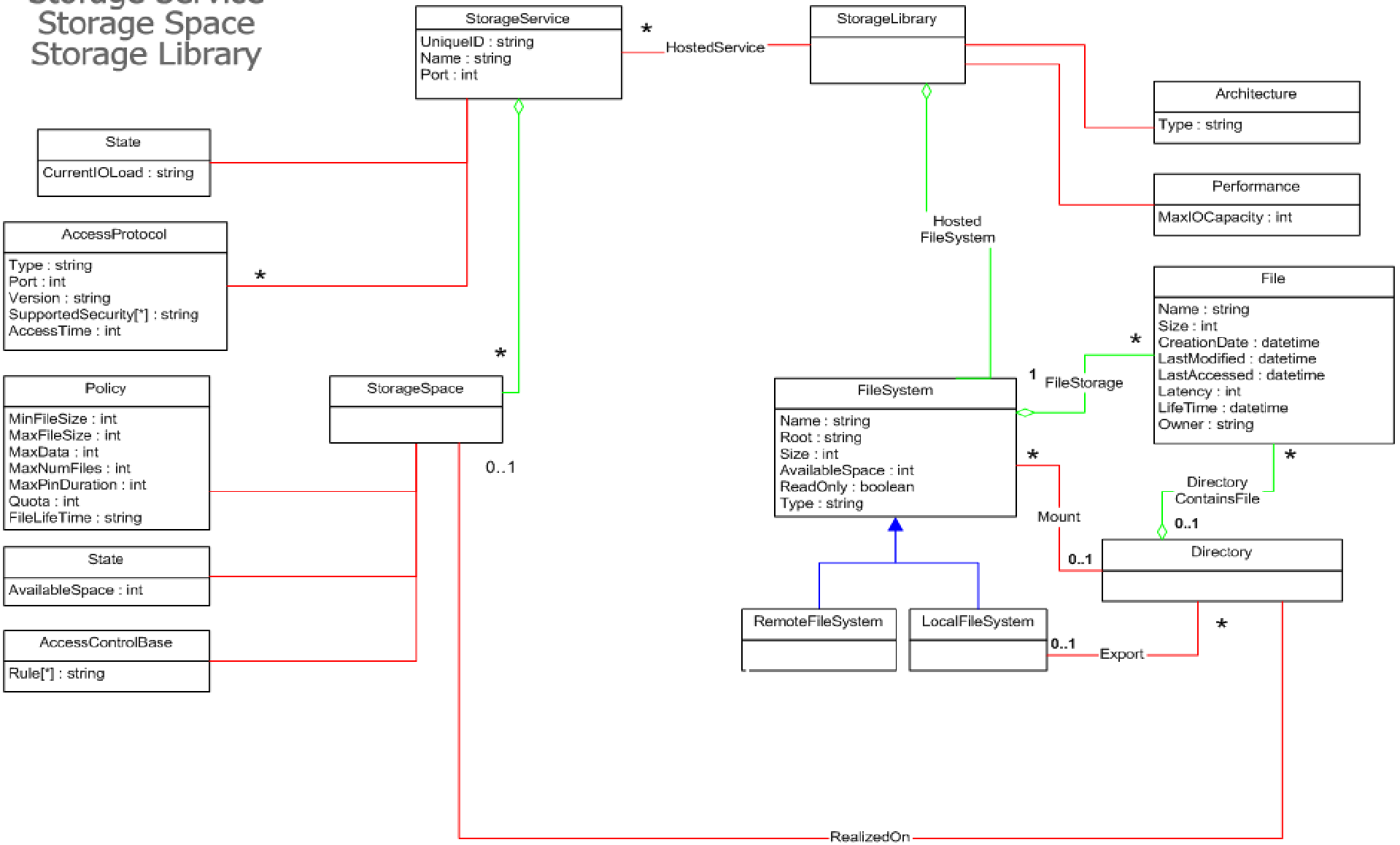
GLUE
Storage Service/Space/**Library**

- Storage Library: the machine providing for both storage space and storage service

GLUE
Storage Service/ Space/ Library



GLUE
Storage Service
Storage Space
Storage Library



CE-SE relationship

- The problem:
  - Job executed by Computing Elements
  - Job may require files stored in Storage Space
  - Several replicas can be spread over the grid
  - The best replica is CE-dependent
  - Which strategy to assign the job to a CE and select the best replica for it?
- Ideal world:
  - Among all CE's accessible by the job owner and that match the job requirements, I select the best one that can access the best replica
- Possible idea of best replica for a given CE:
  - Minimum network load along the path CE-SE
  - Maximum IO capacity for the SE
  - Minimum file latency for the replica

CE-SE relationships

- Real world
  - Missing network statistics
  - Missing max IO capacity (coming in GLUE schema)
  - Missing file latency (in Glue schema, but no i/p)
- We have defined a specific association class (CESEBind) that aims:
  - To represent CE-SE association (chosen by SiteAdmin's)
  - To add parameters that can enforce discovery capabilites
  - For each CE:
    - Group level: list of bound SE
    - Single level: SE-specific info to support the broker decision
    - At the moment: mount point if file locally accessible

Network Resources

- Current activity:
  - Definition of a network model that enables an efficient and scalable way of representing the communication capabilities between grid services for brokering activity
- Idea: partition Grid resources in domains so that resource brokerage does not need to know neither internal details of partitions (such as service location etc.), nor the implementation of the communication infrastructure between partitions

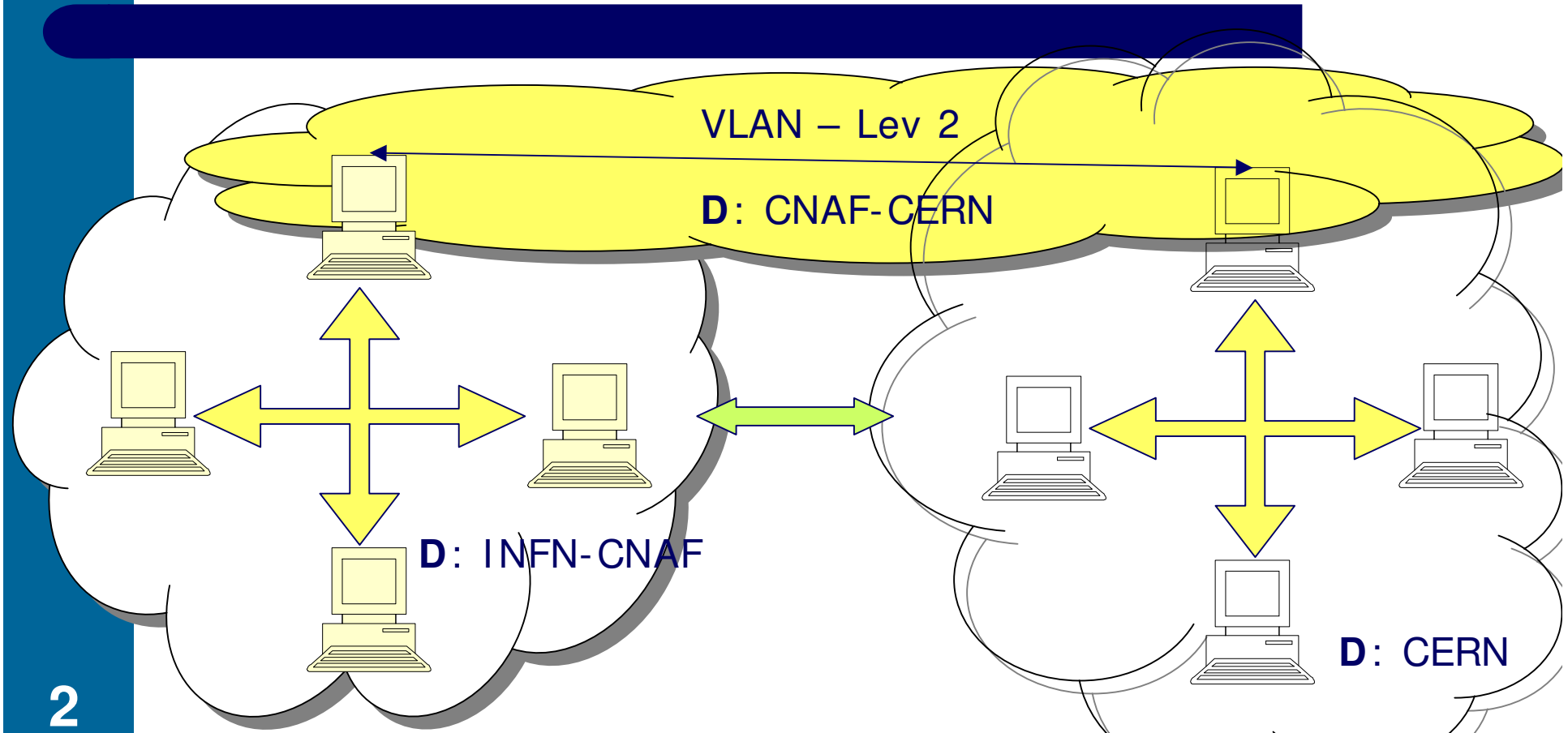
Partitioning the Grid into Domains

- A **Domain** is a set of elements identified by URI's (referred in the model as **edge elements**)
- **Connectivity** is a metric that reflects the quality of communication through a link between two Edge Elements
  - Connectivity between Edge Elements inside a domain is (far) better than the Connectivity with Edge Elements in other Domains
  - In this context, domains are not related to the organization (not an administrative concept)

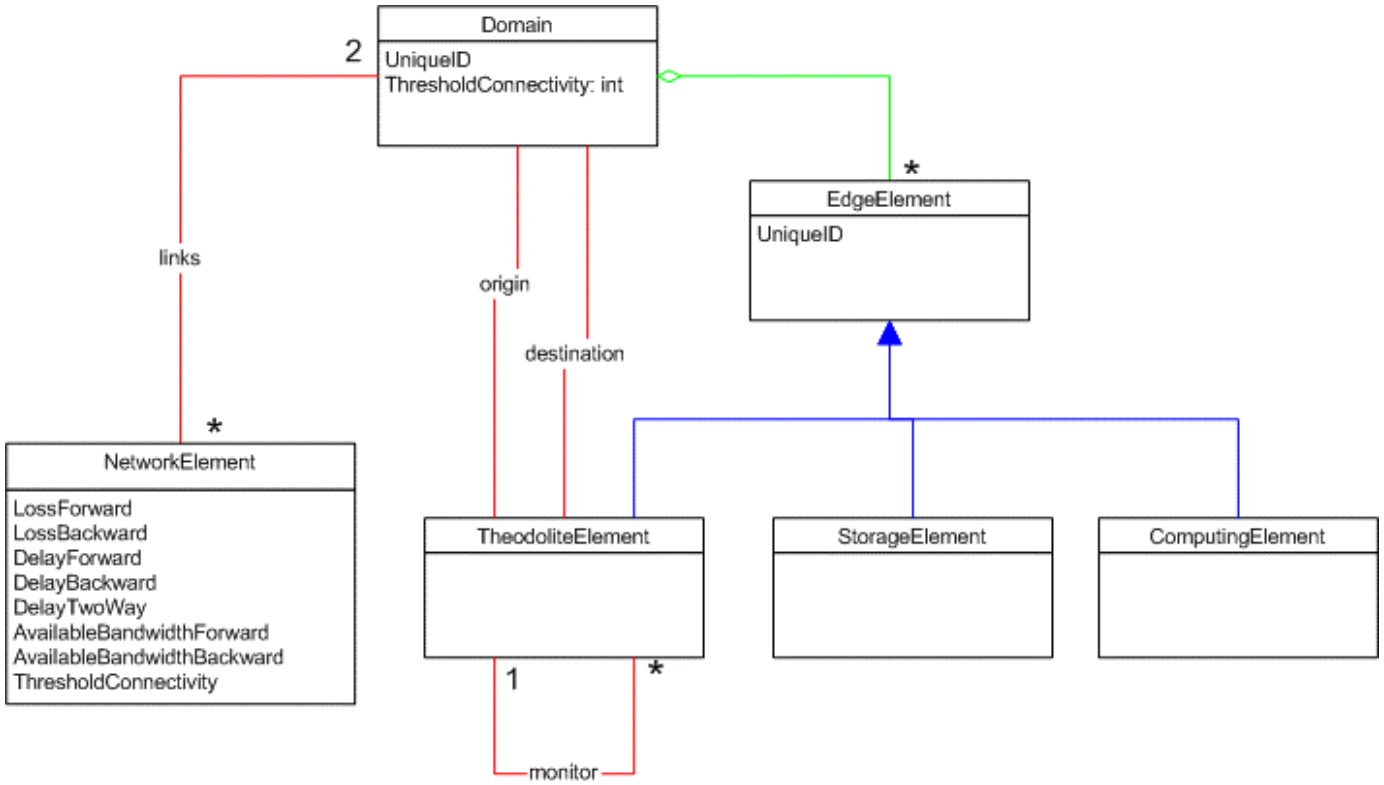
The Network Element

- A **Domain** communicates with other domains using Network Elements
- A **Network Element** offers a communication service (bi-directional) between two Domains; the offered connectivity must not be better than the internal connectivity of the two adjacent Domains
- Each domain has a **Theodolite Element** that gather network element related metrics towards others domains

GLUE
Network Element



A tentative UML Class diagram



Implementation status

- GLUE SCHEMA 1.0 for MDS 2.x
  - LDAP Schema (DataTAG WP4.1)
  - CE Info providers
    - EDG WP4: CE, Cluster, Subcluster
    - DataTAG: host detailed info + monitoring extension
  - SE Info providers
    - EDG WP5 (waiting...I'm doing something by myself at the moment)

Deployment roadmap

- Experimental testbed already working (DataTAG-GLUE):
  - Based on EDG software, rel. 1.4.3
  - Added schema, info providers, GLUE Broker
  - Currently nodes in Bologna, Milano, Napoli and Padova
  - Plans to extend to
    - CERN (LCG)
    - US
      - Wisconsy, VDT on Condor-based cluster
      - FNAL
- LCG software 1.0
- EDG software 2.0

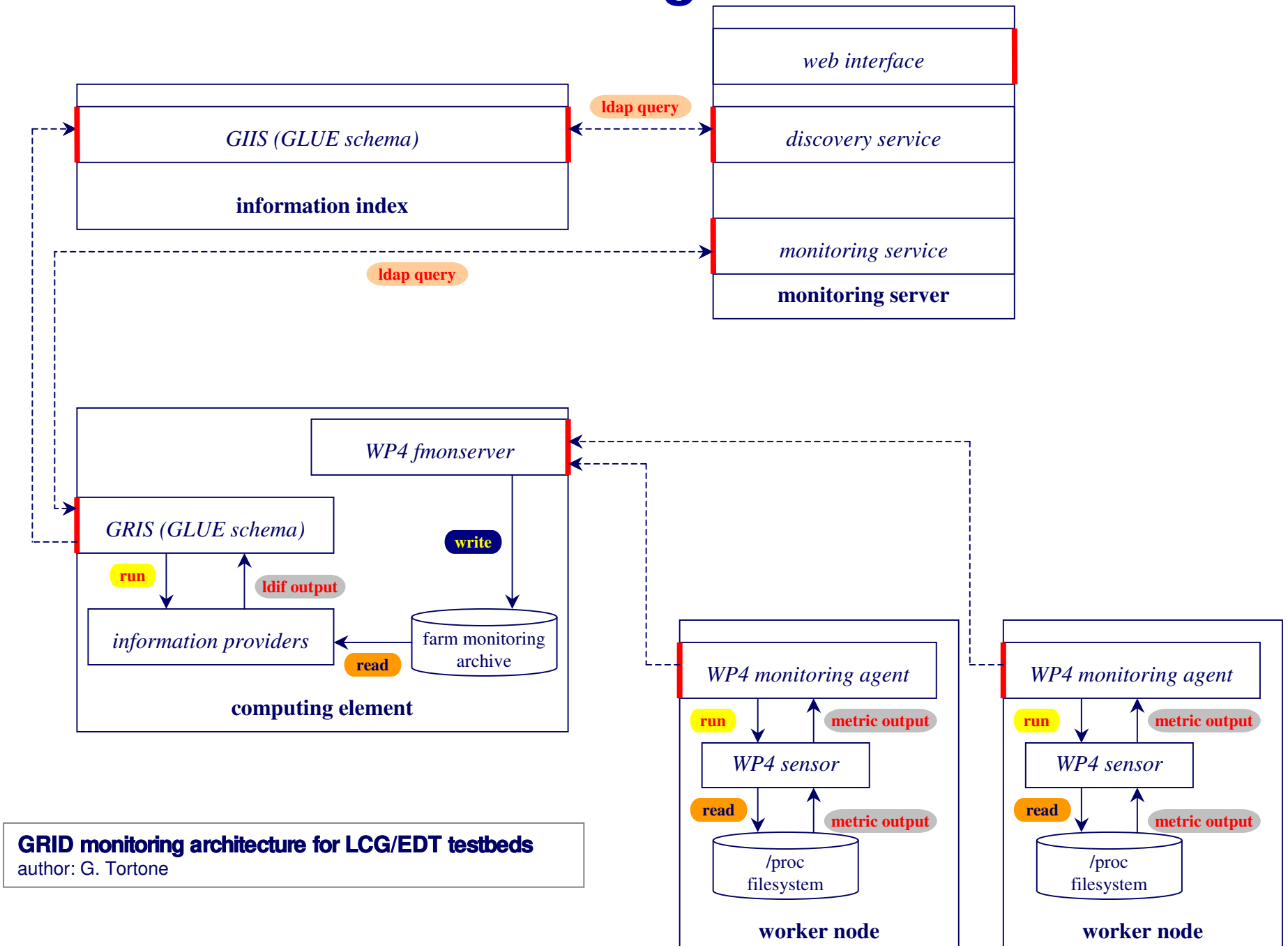
EDT-LCG Monitoring collaboration

Goal

development of a **Grid monitoring tool** in order to monitor the overall functioning of the Grid.

The software should enable the grid administrators to **quickly identify problems** and **take appropriate action**

EDT-LCG Monitoring collaboration



GRID monitoring architecture for LCG/EDT testbeds
author: G. Tortone

EDG WP2 needs

- Queries to the current MDS, how do they change with Glue Schema?
  - Which VO's can access a certain SE and their root directory
  - Which data access protocol are supported by an SE
- New needs: publishing associations between RLS, RMC, ROS and VO's that can invoke them

Moving to Glue schema - queries

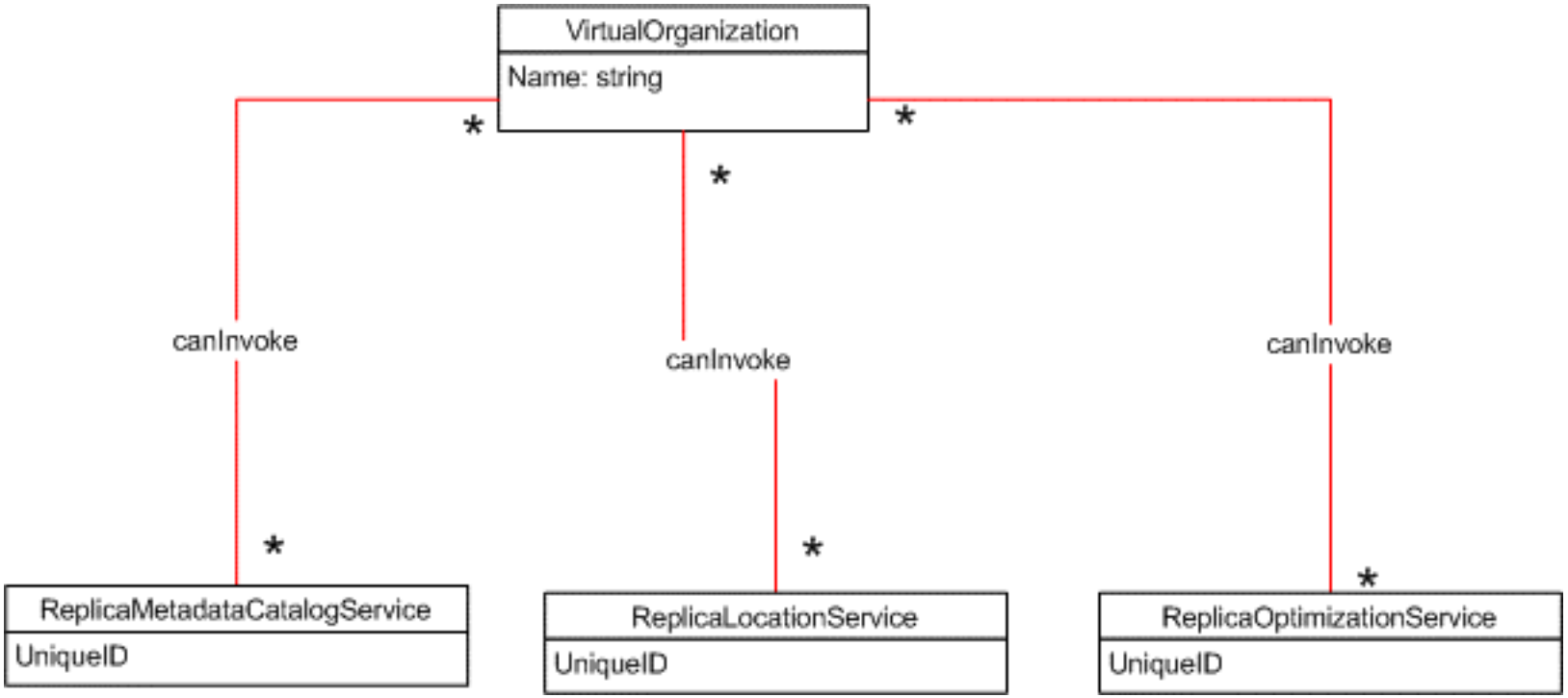
- Supported VO's by a certain SE:

ldapsearch -h *hostname* -p *port* -x
-b "mds-vo-name=*vo-name*,o=grid"
"(&(objectclass=GlueSA)
(GlueChunkKey=GlueSEUniqueID=edt004.cnaf.infn.it))"
GlueSAAccessControlBaseRule GlueSARoot

- Supported protocols by a certain SE:

ldapsearch -h *hostname* -p *port* -x
-b "mds-vo-name=*vo-name*,o=grid"
"(&(objectclass=GlueSEAccessProtocol)
(GlueChunkKey=GlueSEUniqueID=edt004.cnaf.infn.it))"
GlueSEAccessProtocolType

Association between Replica Services and VO's



WHO WILL PUBLISH THESE ASSOCIATIONS?

Main open issues

- Need for a Glue Schema core model
  - coherent naming
  - harmonic evolution
- Computing:
  - aggregated view of a cluster
- Storage:
  - Understand what we really need
  - Tune the schema against SRM people feedback
- Network:
  - Multi-homed hosts
  - Dealing with different class of services
- High Level Grid Services

REFERENCE

- Grid Laboratory Uniform Environment (GLUE)
  DataTAG WP4 and iVDGL Interoperability Group
  version 0.1.2
- GLUE Schema documents
  http://www.cnaf.infn.it/~sergio/datatag/glue
- EDT-LCG Monitoring
  http://gridmon.na.infn.it/lcg-edt
- GGF CIM Grid Schema WG
  http://www.isi.edu/~flon/cgs-wg/