SISTEMI INFORMATIVI INFORMATION SYSTEMS

Grid Integrated Control System

Dr. Giuliano Taffoni

INAF

INFORMATION SYSTEMS UNIT

Plan of the lecture

- The G-ICS in practice
- An overview of GlobusToolkit
  - LDAP primer
  - MDS
  - GRAM
- The new resource: Instrument Element

Instrument crisis

- I have a IE and I want to USE it from my GRID.
- I have a number of IEs and I want to USE all of them.
- Fully compliant with (LCG) GLite
- No space for WS.

The problem

- Limits currently present in the Grid environment: the Grid is able to execute binary code or shell scripts and stores files
- Main focus on the extension of the existing Resource Manager of Globus for providing transparent access to “resources”

Instrument and sensors problems

- **May exist** a pool of Inst/Sens on which USERs' "Actions" can be processed through some control system (CS)
- **may be possible** a Mapping for "Action" on more than one Inst/Sens
- **must be possible** to Grant/Revoke "Actions" to users on Instruments/sensors;
- **must be possible** to enter "Actions" running state with specialized mechanisms.
- **may be possible** an "Action" spawning for join intra/inter G-ICSs
- **may be possible** Synchronizations and Transactions
- **must be possible** to Send "Actions" on a ICS
- **must be possible** to Receive Alarm/Signal/Data from a G-ICS
- **must be possible** to enter Termination requests for canceling "Action"
- **must be possible** to make "Action" Match-Making based on

Instrument problems

- Computing “Resources” tends to connote a tangible entity to be consumed: CPU, storage, etc.
- Simple Taxonomy and Ontology
- For Instruments %%@^@!#$^!$^#^^ !!!!!

Extending Grid Capabilities

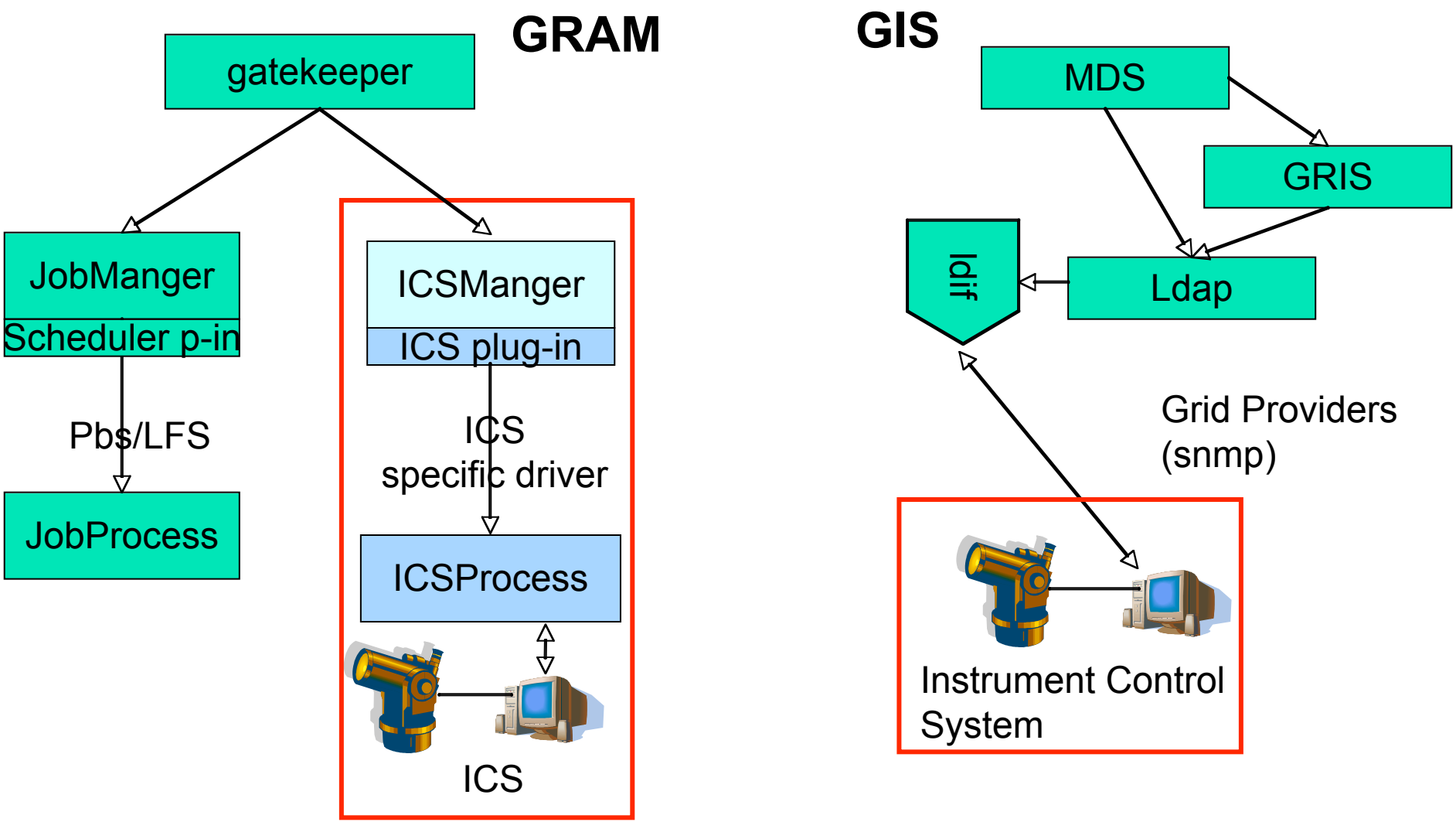
- Provide a proper extension of the Grid to care a new resource
- Security GSI: no need to extend but to use!
- First theory (Grid ASM) then…application.

"A Formal Framework for Defining Grid Systems"
Zsolt N. Nemeth & Vaidy Sunderam
2nd IEEE/ACM (CCGRID'02)

Keywords for extentions

- A resource must be find!
- A resource must be contacted!

Globus ICS integrations

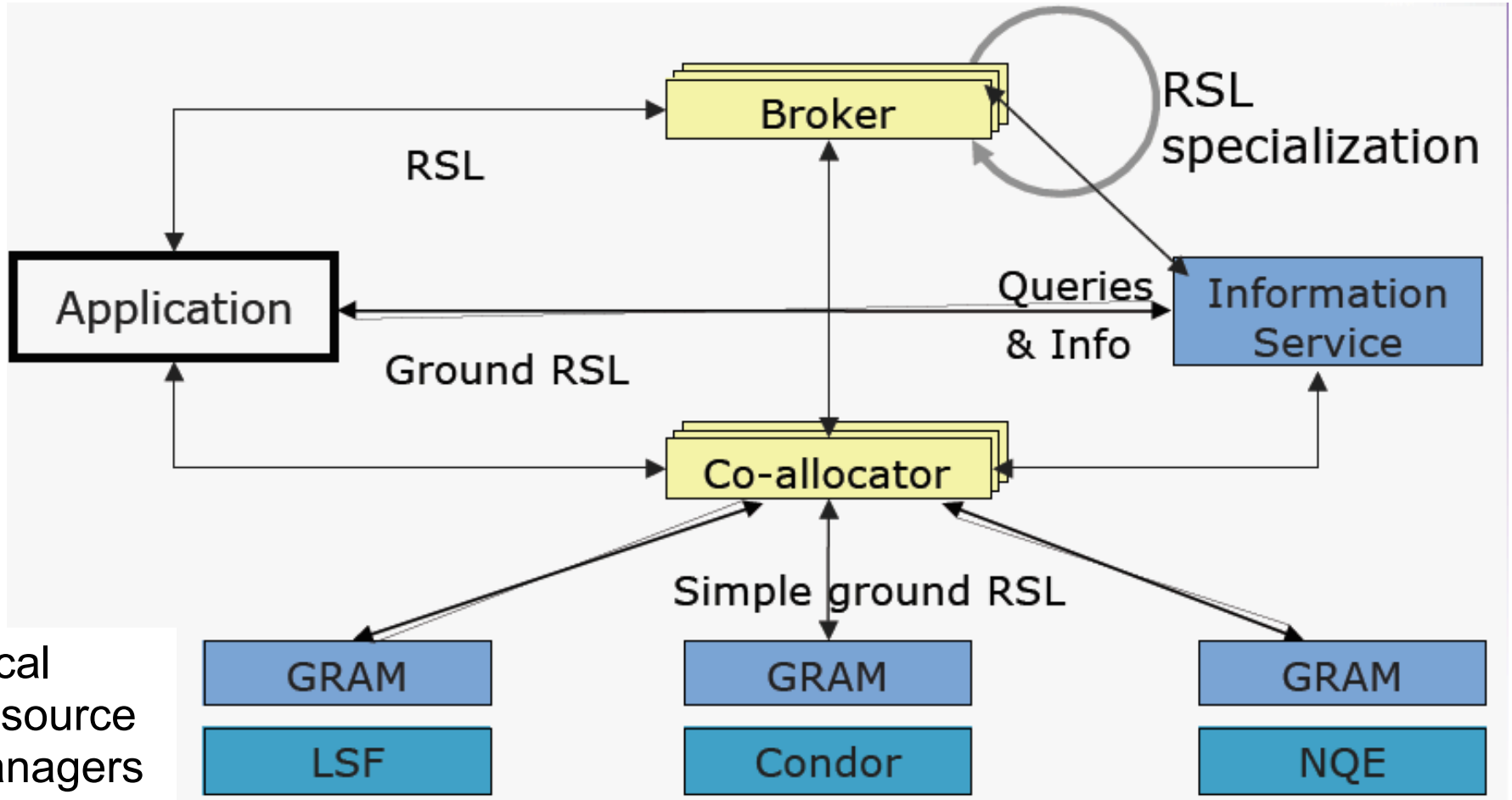


SISTEMI INFORMATIVI INFORMATION SYSTEMS

Job Management System

Resource Management

- The Grid Resource Allocation Management (GRAM) protocol and client API allows programs to be started on remote resources, despite local heterogeneity
- Resource Specification Language (RSL) is used to communicate requirements
- A layered architecture allows application specific resource brokers and co-allocators to be defined in terms of GRAM services
  - Integrated with Condor, PBS, MPICH-G2,



Local Resource Managers

Resource Specification Language

- Common notation for exchange of information between components
  - Syntax similar to MDS/LDAP filters
- RSL provides two types of information:
  - Resource requirements: Machine type, number of nodes, memory, etc.
  - Job configuration: Directory, executable, args, environment
- Globus Toolkit provides an API/SDK for manipulating RSL

RSL syntax

- Elementary form: parenthesis clause
  - (attribute op value [value …])
- Operators Supported:
  - <, <=, =, >=, > , !=
- Some supported attributes:
  - executable, arguments, environment, stdin, stdout, stderr, resourceManagerContact, resourceManagerName
- Unknown attributes are passed through
  - May be handled by subsequent tools

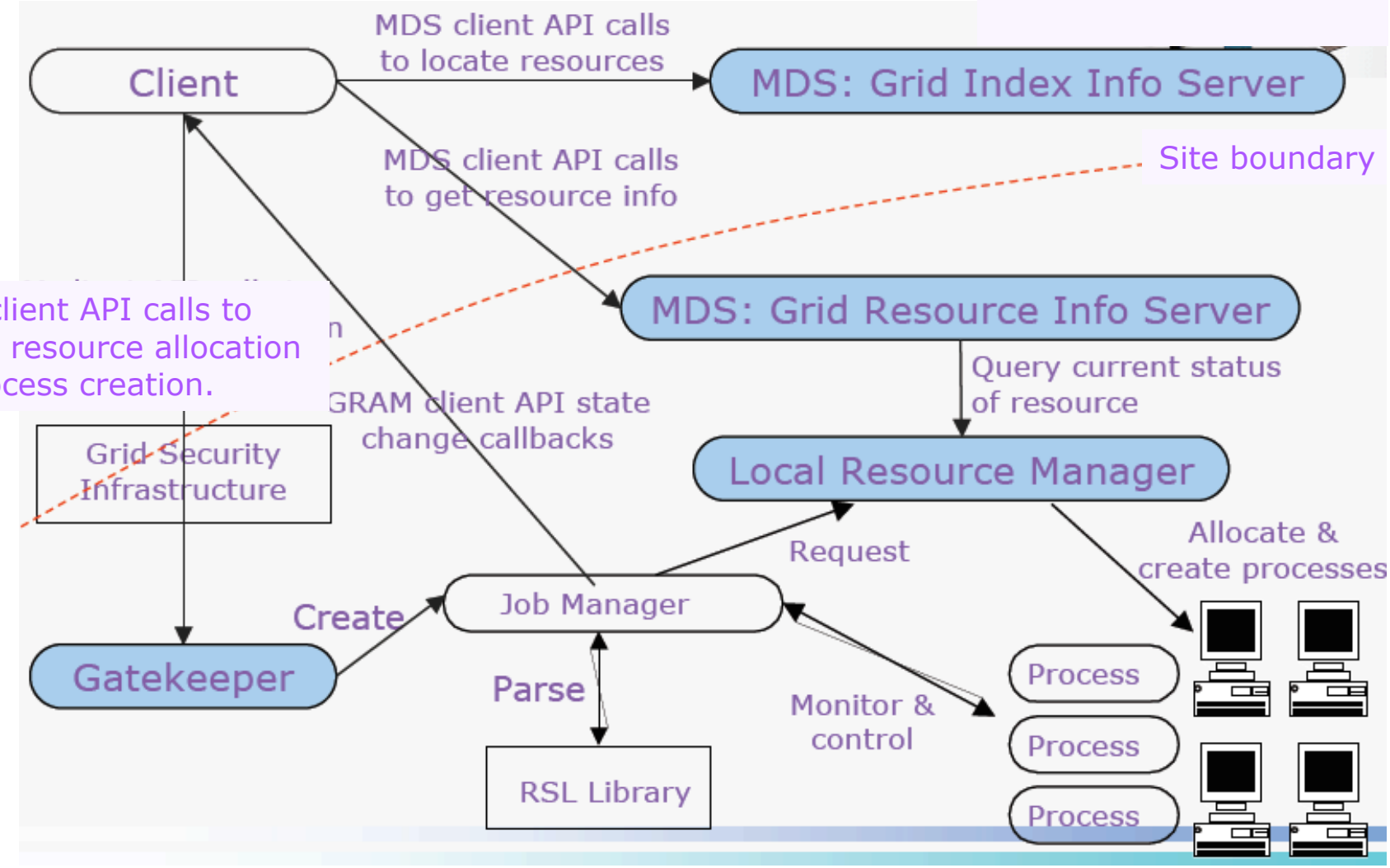
RSL example

```
(* this is a comment *)
& (executable = a.out (* <-- that is an unquoted literal *))
  (directory  = /home/nobody )
  (arguments  = arg1 "arg 2")
  (count = 1)
```

GRAM protocol

- Create job environment
- Stage files in/out the environment
- Submit a job to the local scheduler
- Monitor job state
- GRAM == protocol to communicate with job management service (JMS)
- JMS == startup + stage + monitor

Gram Components



GRAM client API calls to request resource allocation and process creation.

Job submission Interfaces

- Globus Toolkit includes several command line programs for job submission
  - globus-job-run: Interactive jobs
  - globus-job-submit: Batch/offline jobs
  - globusrun: Flexible scripting infrastructure

Some examples

```
$ globus-job-run grid-data.rl.ac.uk /bin/hostname -f grid-data.rl.ac.uk
```

```
$ globus-job-submit grid-data.rl.ac.uk/jobmanager-pbs /bin/hostname -f
https://grid-data.rl.ac.uk:64001/1415/1110129853/
$ globus-job-status https://grid-data.rl.ac.uk:64001/1415/1110129853/
DONE
$ globus-job-get-output https://grid-data.rl.ac.uk:64001/1415/1110129853/
grid-data12.rl.ac.uk
$ globus-job-clean https://grid-data.rl.ac.uk:64001/1415/1110129853/

  WARNING: Cleaning a job means:
     - Kill the job if it still running, and
     - Remove the cached output on the remote resource

  Are you sure you want to cleanup the job now (Y/N) ?
  Y

  Cleanup successful.
```

Grid Information Services

GIS concept

- 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?
- We need a general information infrastructure to answer these questions

GIS mission

- Provide access to static and dynamic information regarding system components
- A basis for configuration and adaptation in heterogeneous, dynamic environments
- Requirements and characteristics
  - Uniform, flexible access to information
  - Scalable, efficient access to dynamic data
  - Access to multiple information sources
  - Decentralized maintenance
- Information is always old
  - Time of flight, changing system state
  - Need to provide quality metrics

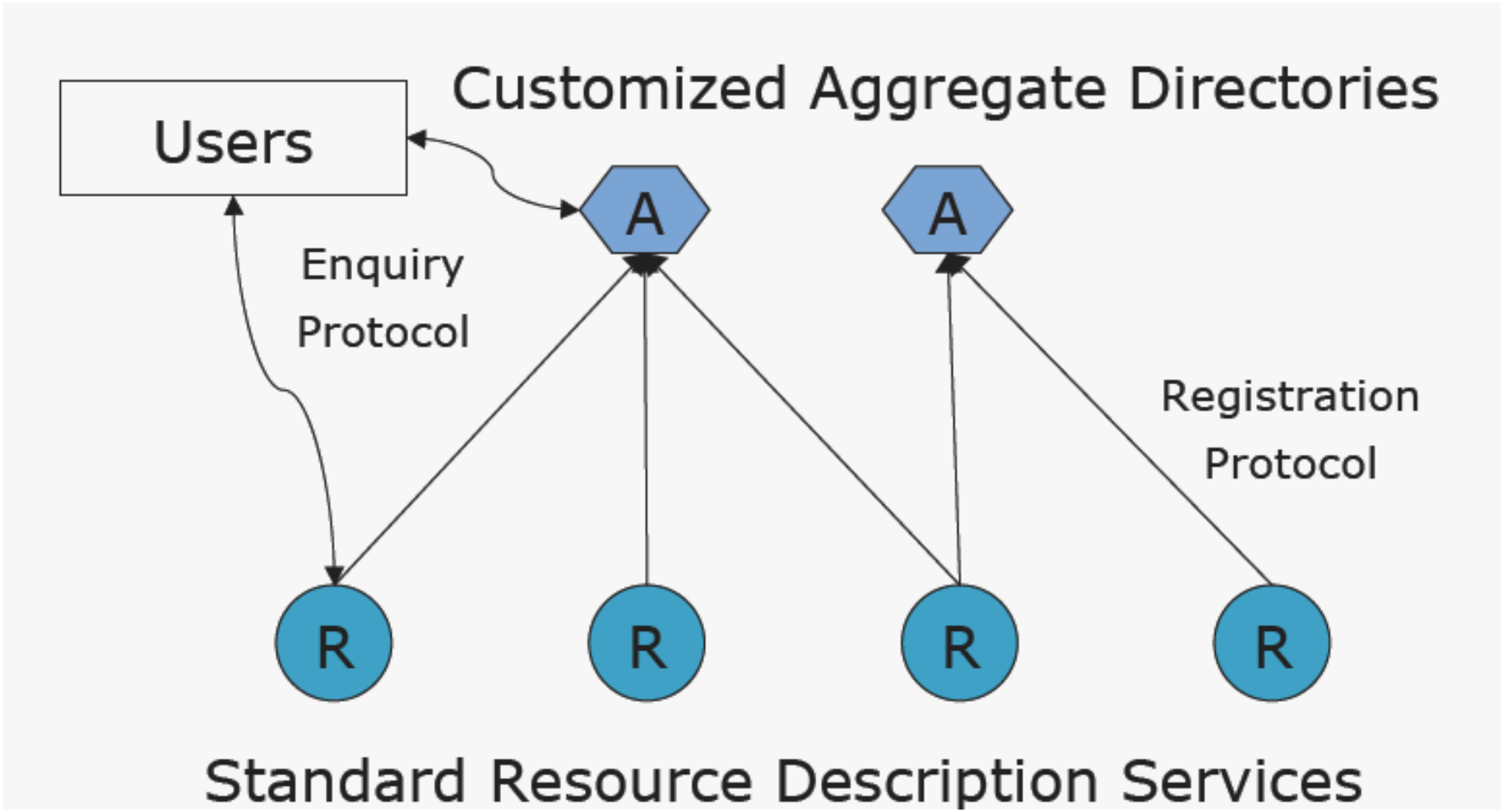
GIS architecture

- Resource Description Services
  - Supplies information about a specific resource (e.g. Globus GRIS).
- Aggregate Directory Services
  - Supplies collection of information which was gathered from multiple GRIS servers (e.g. Globus GIIS).
  - Customized naming and indexing

Information Protocols

- Grid Resource Registration Protocol
  - Support information/resource discovery
  - Designed to support machine/network failure
- Grid Resource Inquiry Protocol
  - Query resource description server for information
  - Query aggregate server for information
  - LDAP

GIS Architecture



Metacomputing Directory service

- Use LDAP as Inquiry
- Access information in a distributed directory
  - Directory represented by collection of LDAP servers
  - Each server optimized for particular function
- Directory can be updated by:
  - Information providers and tools
  - Applications (i.e., users)
  - Backend tools which generate info on demand
- Information dynamically available to tools and applications

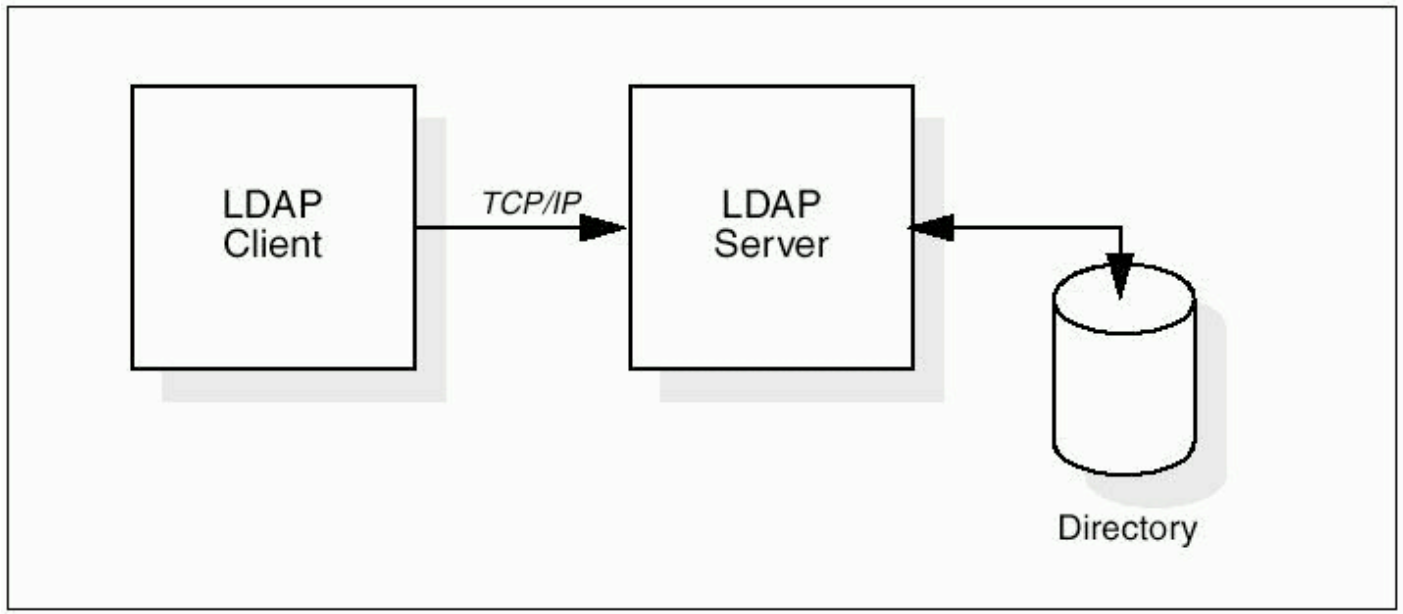
Two classes of MDS

- Grid Resource Information Service (GRIS)
  - Supplies information about a specific resource
  - Configurable to support multiple information providers
  - LDAP as inquiry protocol
- Grid Index Information Service (GIIS)
  - Supplies collection of information which was gathered from multiple GRIS servers
  - Supports efficient queries against information which is spread across multiple GRIS server
  - LDAP as inquiry protocol
- IN EGEE we use BDII

What is LDAP?

- Lightweight Directory Access Protocol
- Used to access and update information in a directory built on the X.500 model
- Specification defines the content of messages between the client and the server
- Includes operations to establish and disconnect a session from the server

LDAP Server



LDAP

- Information
  - Structure of information stored in an LDAP directory.
- Naming
  - How information is organized and identified.
- Functional / Operations
  - Describes what operations can be performed on the information stored in an LDAP directory.
- Security
  - Describes how the information can be protected from unauthorized access.

Some Basics

- Data is organised into an hierarchical tree
- Each ‘entry’ (tree node) is identified by a DN (distinguished name) e.g. uid=mclark,ou=people,dc=metaparadigm,dc=com
- Each component of a DN is called an RDN (relative DN) and represents a branch in the tree
- The RDN must be unique within the nodes at the same level of the tree (is generally equivalent to one of the attributes ie. ‘uid’ or ‘cn’ in the case of a person)
- Each node has 1 or many attribute values associated with it. Each attribute can have 1 or many values

```
dc=com
   \-----  dc=metaparadigm
             |-----  ou=people
             |         \-----  uid=mclark
             \-----  ou=groups
                       \-----  cn=users
```

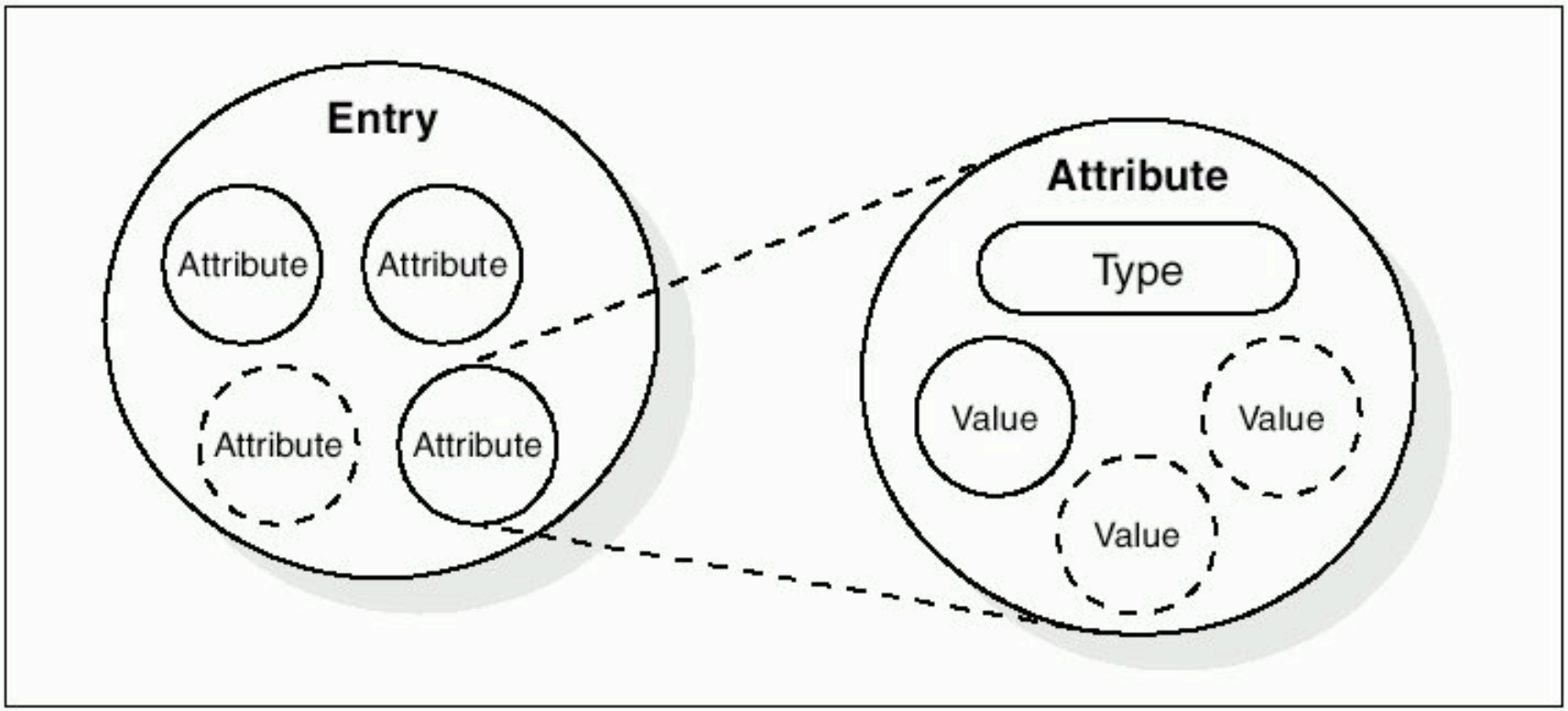
LDAP shemas

- Schema
  - Every entry must have an object class
  - Defines what object classes allowed
  - Where they are stored
  - What attributes they have (objectClass)
  - Which attributes are optional (objectClass)
  - Type/syntax of each attribute (objectClass)
  - Query server for info: zero-length DN
- LDAP schema must be readable by the client

Shemas

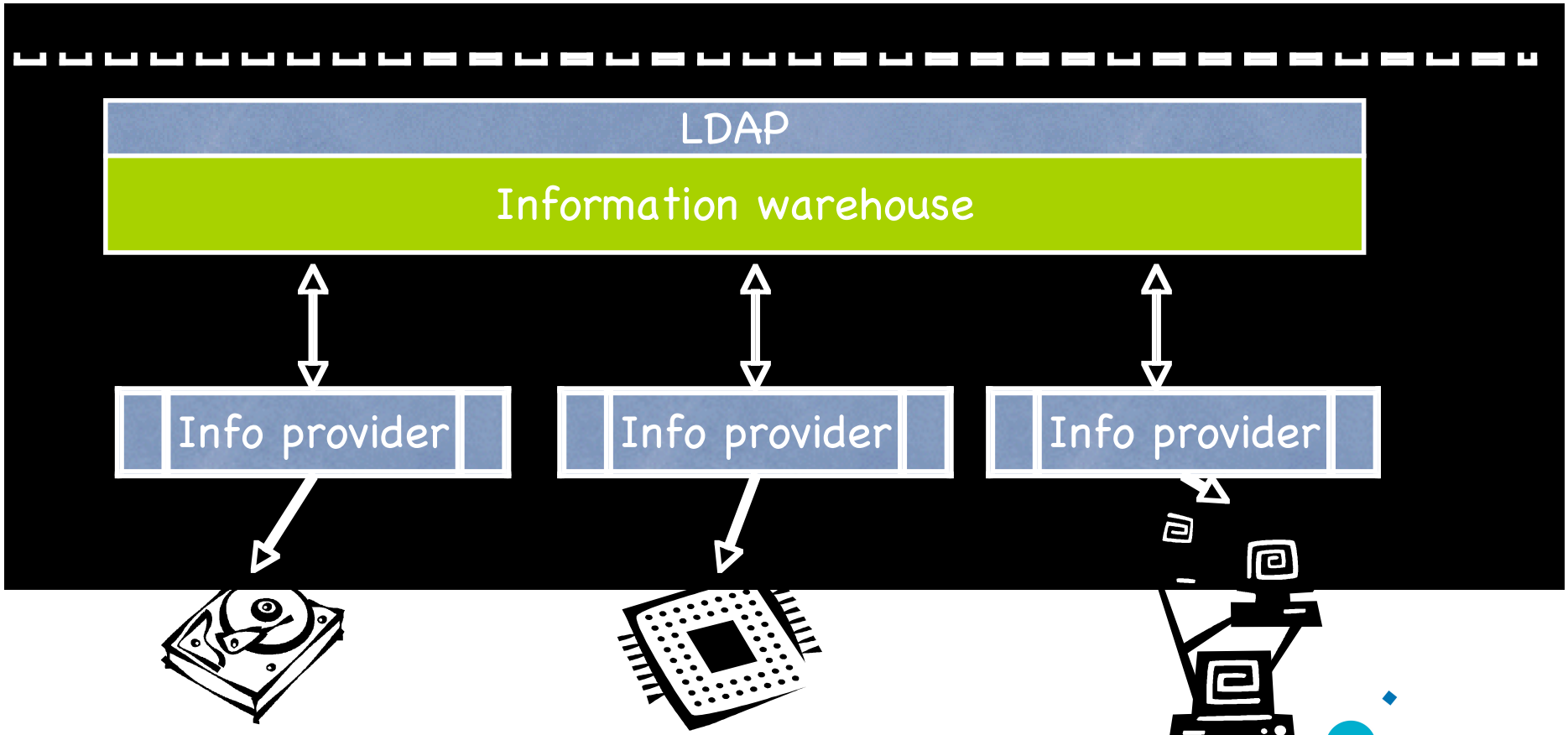
- Many standard schemas exist including:
  - People schemas - person, organisationalPerson, inetOrgPerson, posixAccount, mailLocalRecpient, strongAuthenticationUser
  - Group schemas – groupOfUniqueNames, posixGroup, organisationalRole, roleMember
  - Host / Network schemas – domain, ipHost, ipNetwork, ipProtocol, ipService,
  - ieee802Device, bootableDevice
- It exists GRID schemas (Glue + Globus)

Information Storage



How does MDS it work?

- Resources run a standard information service (Grid Resource Information system) which speaks LDAP and provides information about the resource.

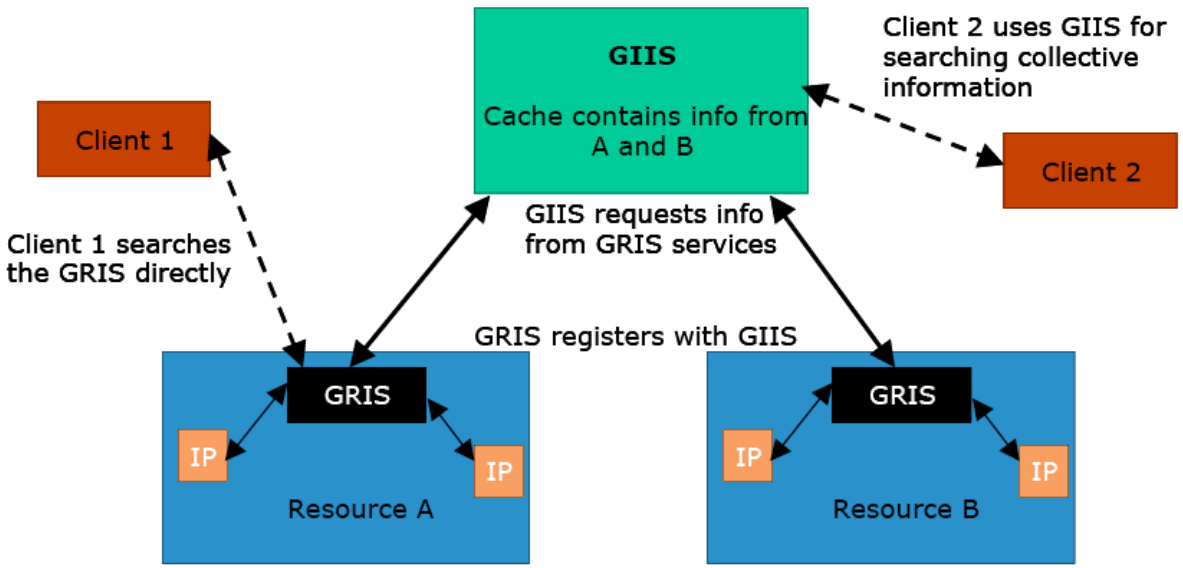


GRIS

- Server which runs on each resource
  - Given the resource DNS name, you can find the GRIS server (well known port = 2135)
- Provides resource specific information
  - Much of this information may be dynamic
    - Load, process information, storage information, etc.
    - GRIS gathers this information on demand
- "White pages" lookup of resource information
  - Ex: How much memory does machine have?
- "Yellow pages" lookup of resource options

Who Does it Works?

- GIIS (index information system) provides a “caching” service much like a web search engine.
- GIIS provides the collective-level indexing/searching function.



Some example

```
$ ldapsearch -LLL -x -h grid006.oat.ts.astro.it:2135 -b "mds-vo-name=local,o=grid"
dn: GlueCEUniqueID=grid006.oat.ts.astro.it:2119/jobmanager-lcgpbs-cert,mds-vo-
 name=local,o=grid
objectClass: GlueCETop
objectClass: GlueCE
objectClass: GlueSchemaVersion
objectClass: GlueCEAccessControlBase
objectClass: GlueCEInfo
objectClass: GlueCEPolicy
objectClass: GlueCEState
objectClass: GlueInformationService
objectClass: GlueKey
GlueCEHostingCluster: grid006.oat.ts.astro.it
GlueCEName: cert
GlueCEUniqueID: grid006.oat.ts.astro.it:2119/jobmanager-lcgpbs-cert
GlueCEInfoGatekeeperPort: 2119
GlueCEInfoHostName: grid006.oat.ts.astro.it
GlueCEInfoLRMSType: pbs
GlueCEInfoLRMSVersion: 2.1.6
GlueCEInfoTotalCPUs: 2
GlueCEInfoJobManager: lcgpbs
GlueCEInfoContactString: grid006.oat.ts.astro.it:2119/jobmanager-lcgpbs-cert
GlueCEInfoApplicationDir: /opt/exp_soft
GlueCEInfoDataDir: /flatfiles/tmp/
```

From computational resource to Instrument Element

Key concepts

- An instrument assimilated to a standard resource
- the IE inherits all the features and properties of a CE
- the IE inherits all the properties and services of SE
- IE=CE+SE

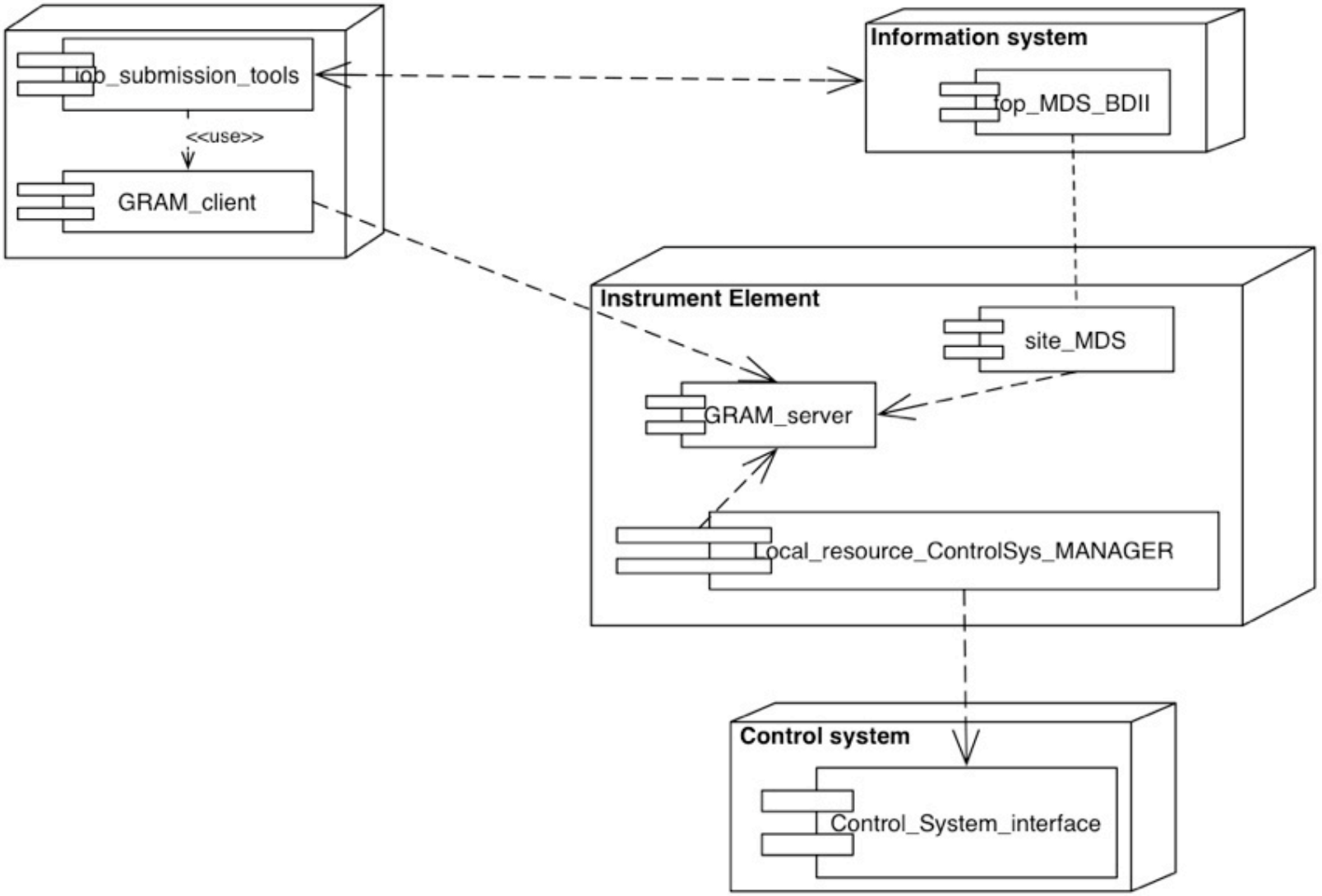
Key concepts

- We are not dealing with executables but with “Actions”
- We are not dealing with jobmanagers but with STATE managers
- We are still dealing with queue but G-ICS LRMS “queue”

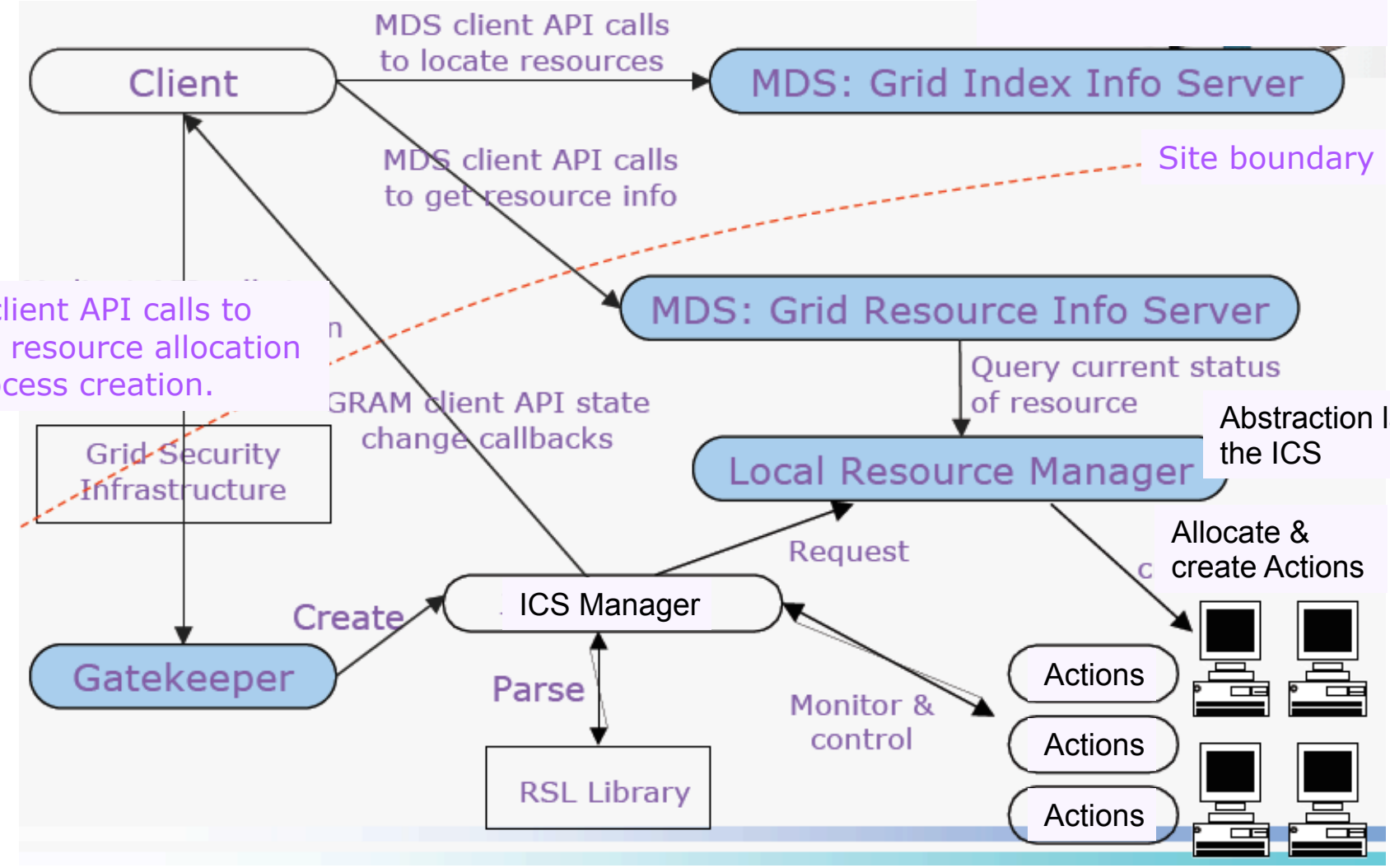
Key concepts

- Actions modify the state of an Instrument
- Standard GRIS may be used as
  - "white page": static info on the Instrument
  - "Yellow page": resource options
    - status and actions I can do on the resource
- Fully integrated in the GRID env
  - easy to use all the high level services to interact, create workflow, etc.

IE components



Gram Components



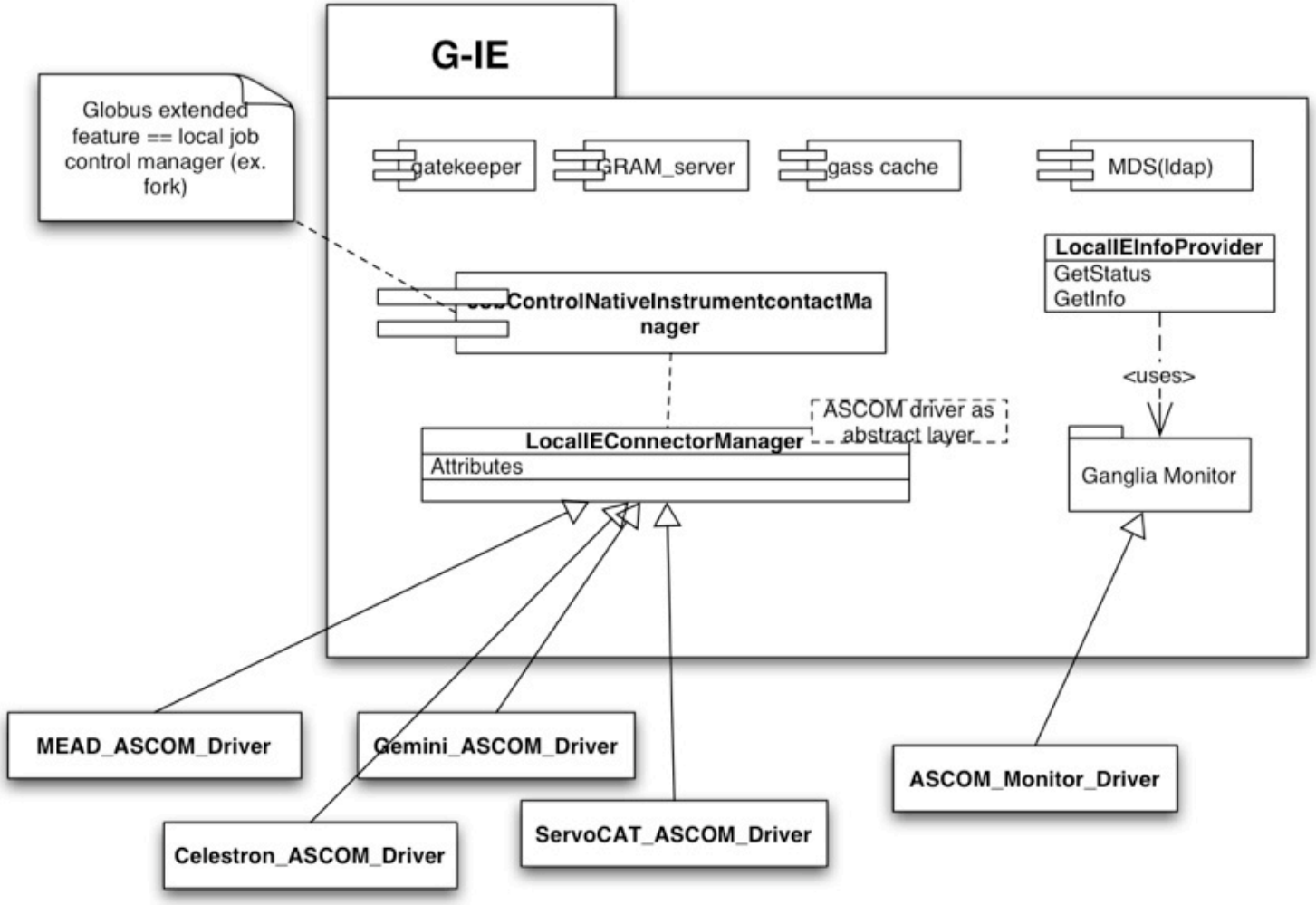
GRAM client API calls to request resource allocation and process creation.

Abstraction layer over the ICS

FROM LRM to a State RM

- Instrument States and Actions
- Queue => Control System

Telescope example



In practice

```
➢globus-job-run g.ie.host/iemanager-telescope -queue
observation pointto 123 43
DONE
```

```
➢globus-job-submit g.ie.host/iemanager-telescope -queue
observation “getbias” 300
DONE
```

```
➢globus-job-submit g.ie.host/iemanager-telescope -queue
observation “getbias” 300 -o /flatfiles/SE00/bias.fits
DONE
```

```
>glite-gridftp-ls gsiftp://g.ie.host/flatfiles/SE00/bias.fits
```

Instrument IS

- IS may be used to monitor sensors/ instrument activity
  - actions and states
- IS may be used to publish sensors/ instrument type and static informations
  - location
  - type
  - curator...

Monitoring and Descovery

- What is the status of my instrument?
- Which instrument should I need to contact?
- Which actions my I use?

How to monitor Instrument

- Define metadata related to your instrument (semantic and taxonomy).
- Create Ldap schemas on the basis of the metadata
- Create an ldif file for your instrument

GIS modifications

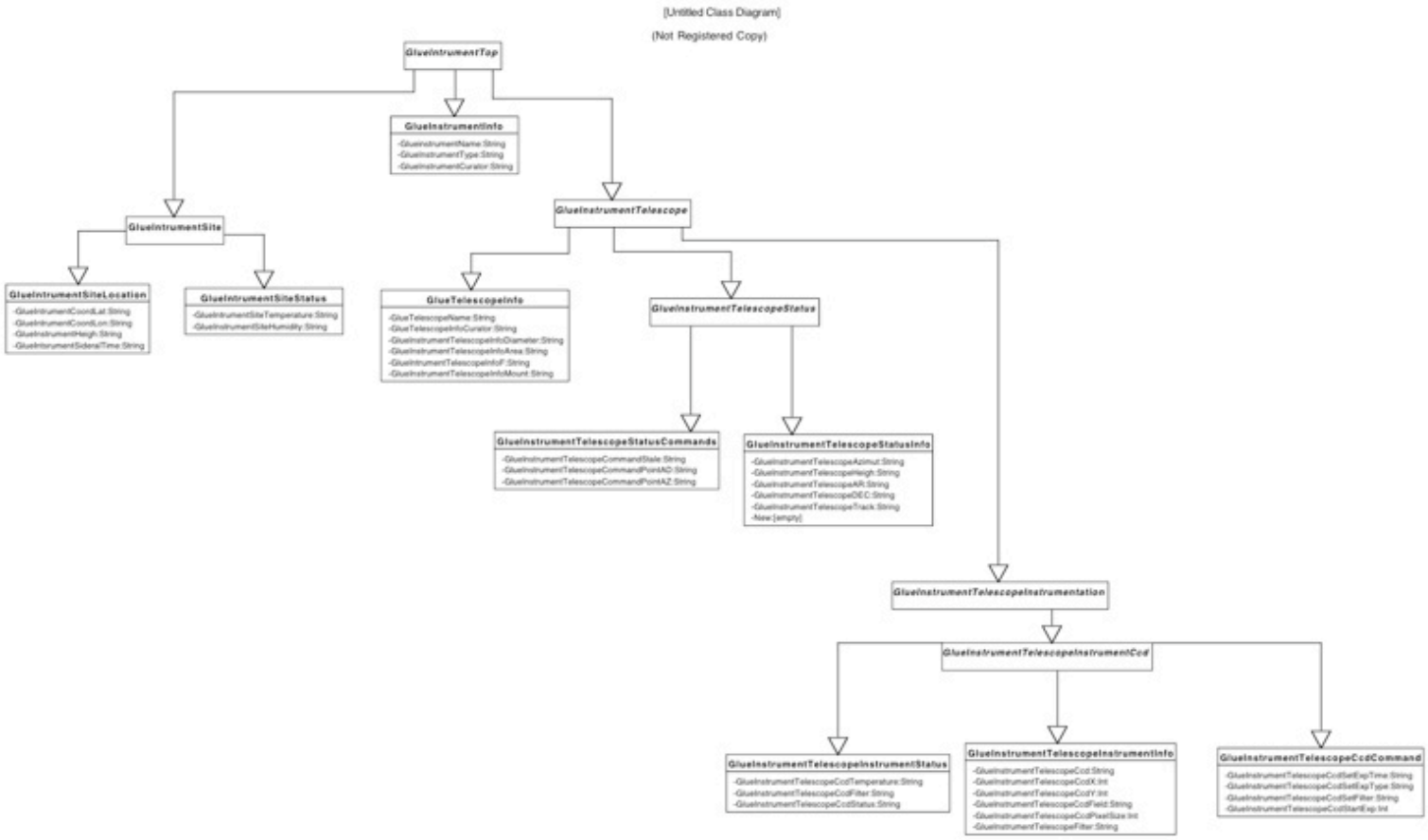
- Information providers
  - Delivers resource information to backend
  - uses ganglia of snmp to get/put informations
- new GRIS
- BDII update is for free
- APIs for accessing & updating MDS contents
  - C, Java, PERL (LDAP API, JNDI)

Telescope example

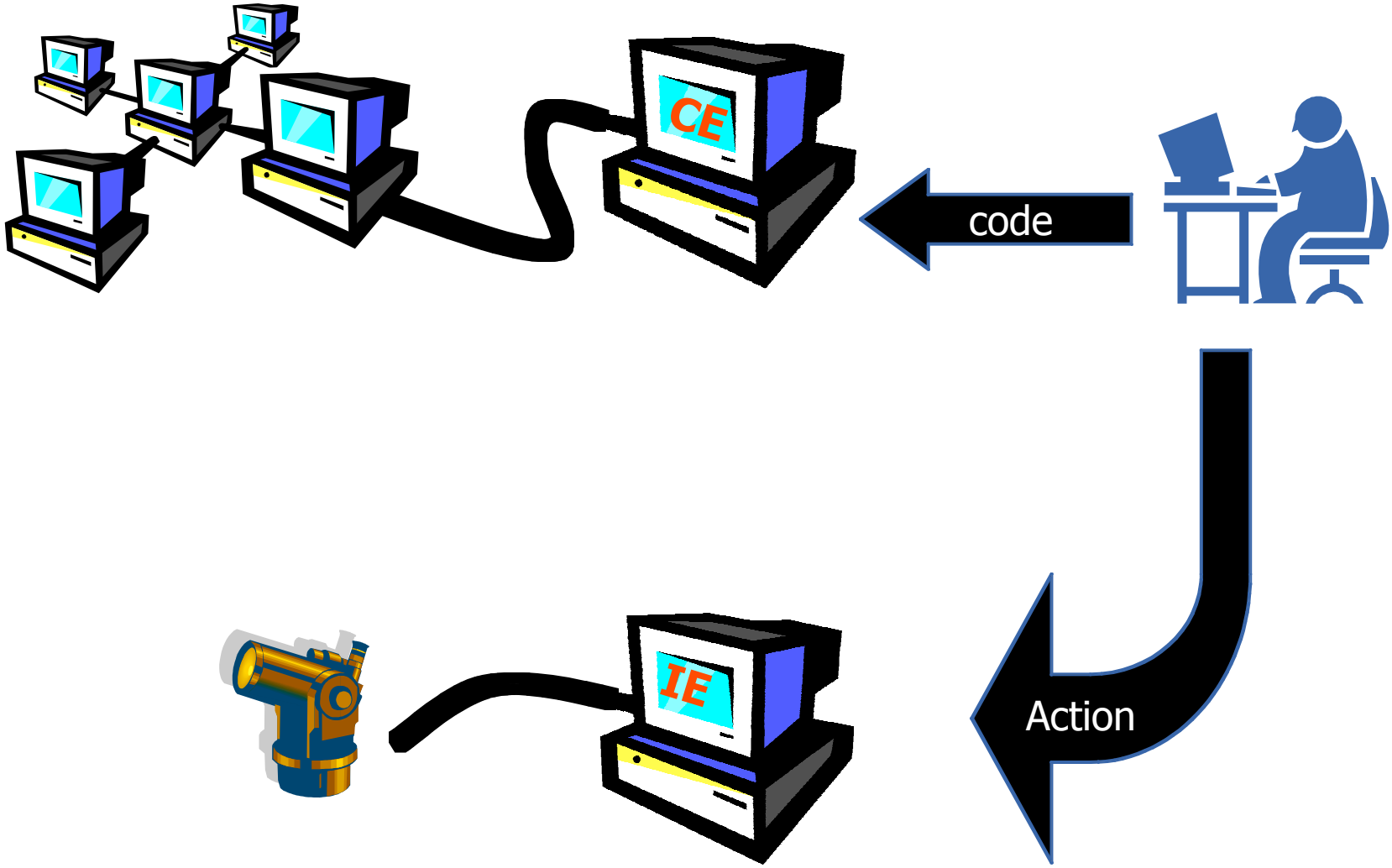
```
> ldapsearch -LLL -x -H g.ie.host -b "mds-vo-
   name=site,o=grid"

dn:GlueIEUniqueID=g.Ie.host:2119/iemanager-ODBC, mds-vo-
   name=local,o=grid
objectClass: GlueCETop
objectClass: GlueCE
objectClass: GlueIE
objectClass: GlueIETop
objectClass: GlueKey
GlueIEName: TESTDB
GlueIEStateStatus: Production
GlueIEInfoLRMSType: ARGO
GlueIEInfoLRMSVersion:7.3
```

Schema for instruments

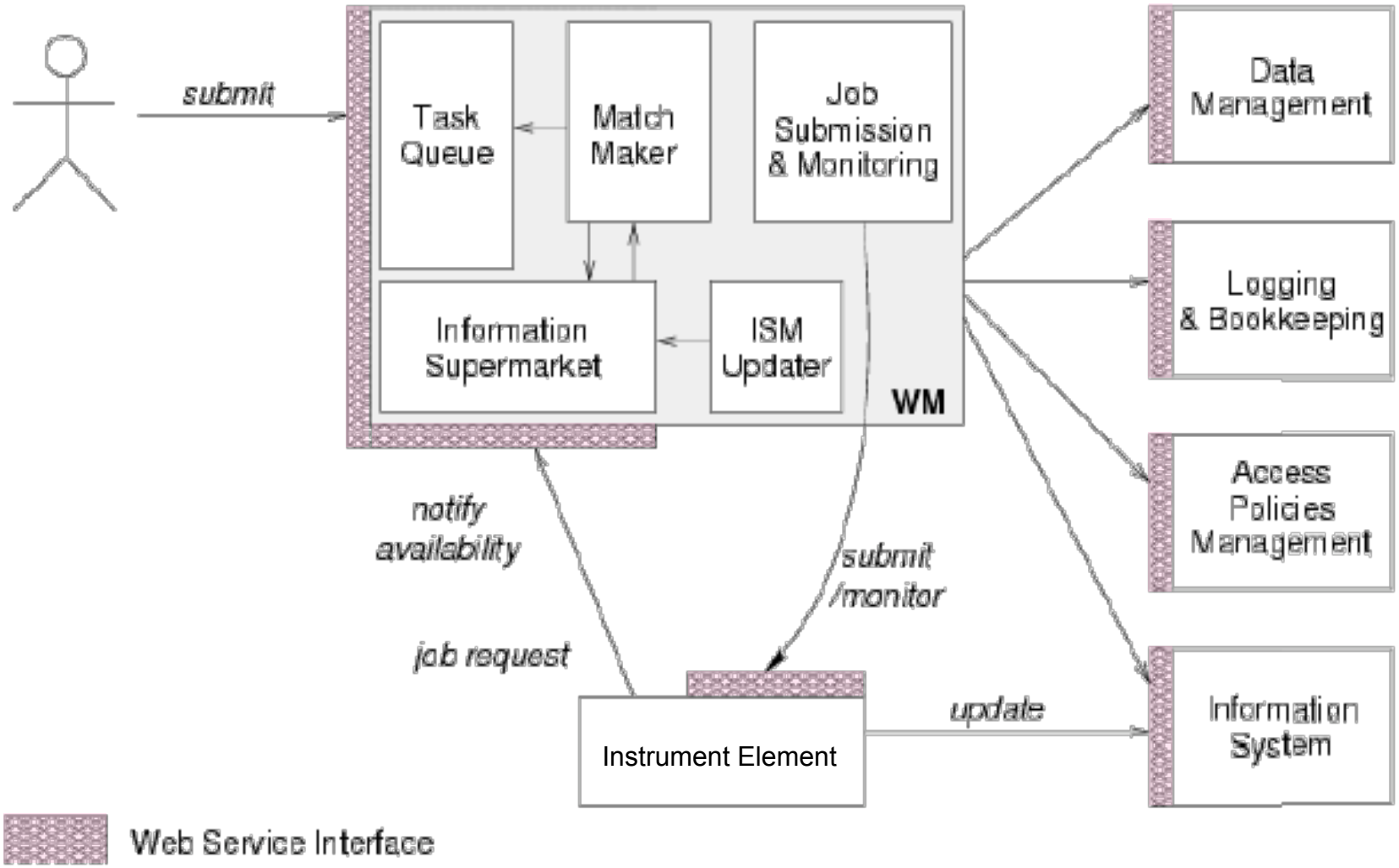


the IE picture



IE advantages

- Standard approach: fully integrated in EGEE: DMS+WMS



IE advantages