R-GMA: Architecture and use

Giuseppe La Rocca
INFN Catania
Retreat between GILDA and ESR VO on gLite
Bratislava, 27-30.06.2005

Outline

- Introduction to R-GMA.
- Producers.
  - Producers Type.
    - Primary Producer
    - Secondary Producer
- Consumer.
- Registry.
- Security.
- History or Latest Query
- Continuos query
- Using R-GMA.
  - The R-GMA Browser
  - The R-GMA client CLI
- R-GMA APIs
- Links

Introduction to R-GMA

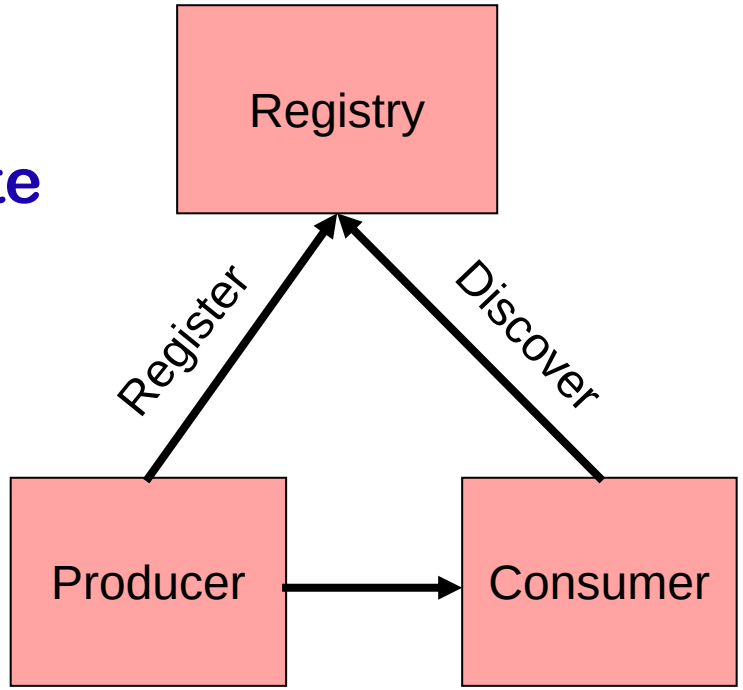
- (R-GMA) Relational Grid Monitoring Architecture is an implementation of the Grid Monitoring Architecture (GMA), which models the informations of a Grid as a set of
  - *Producers* (who provide informations);
  - *Consumers* (who request informations);
  - *Registry* (which regulate the comunication between Consumers and Producers);

- Data is viewed as a table.
- Each entry is a row (tuple).
- Uses a query languages based on a subset of SQL.

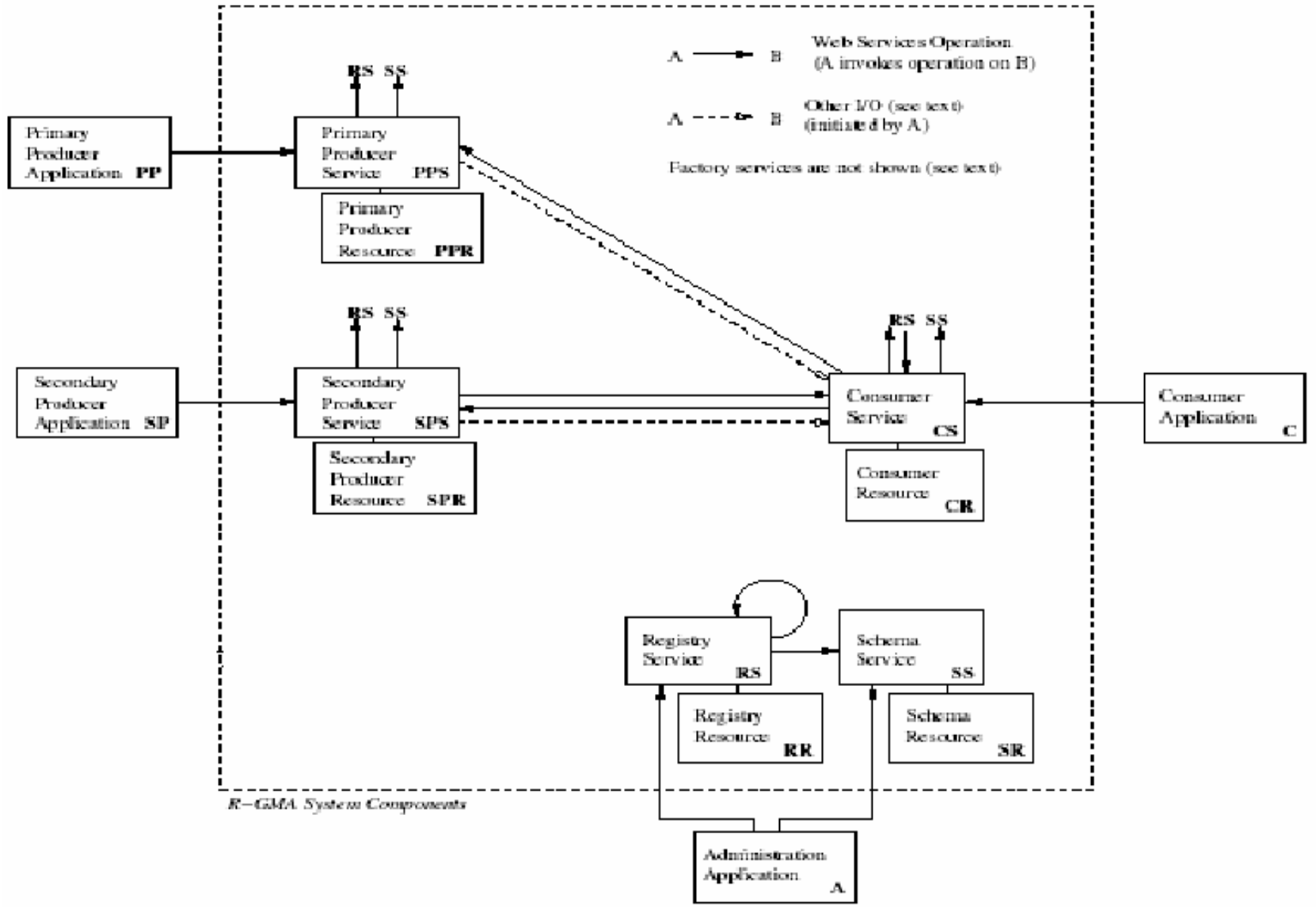
Name	ID	birth	Group

GMA

- The Producer stores its information in the Registry.
- The Consumer can query the Registry to find out what type of information is available and locate the best producers that provide such information (*mediation*).
- Once a Consumer has this information it can contact the Producer to get all the data.
- The Registry contains, for each table, a list of producers who have offered to publish rows for it.

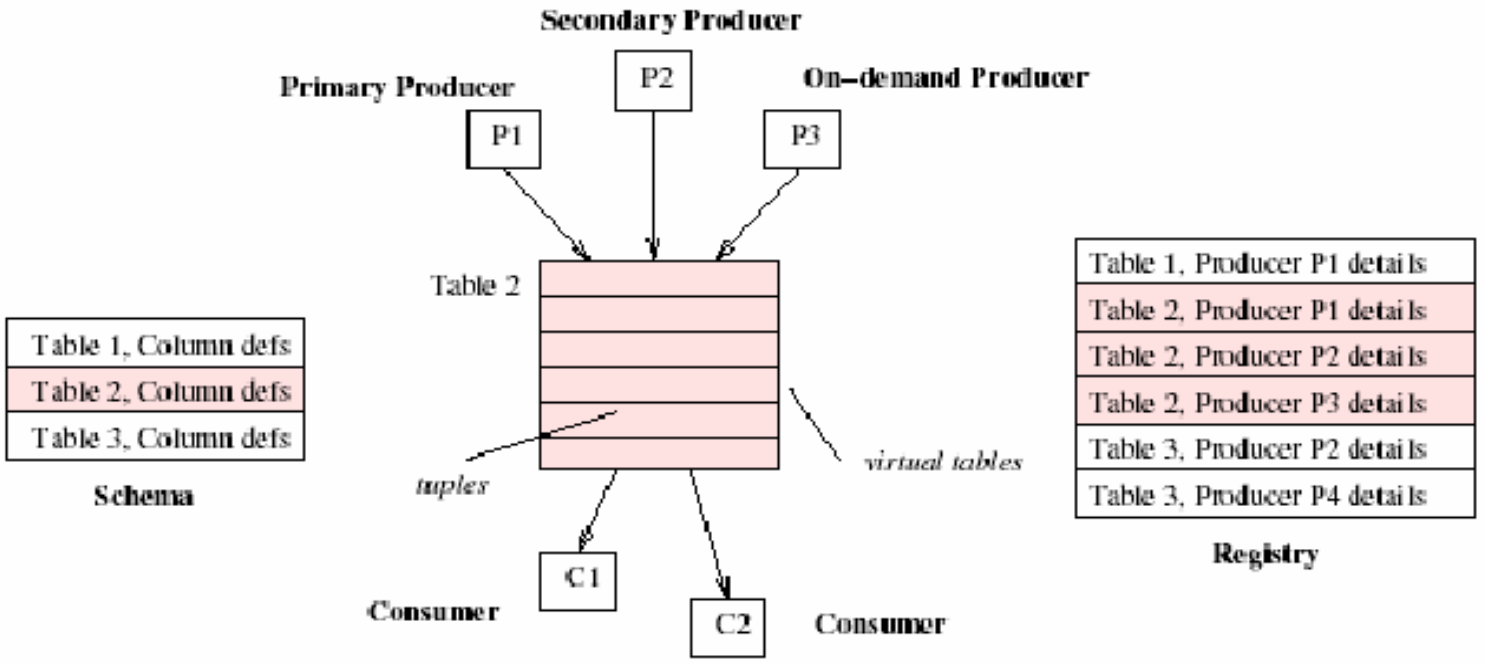


Overview


A → B
Web Services Operation
(A invokes operation on B)
A - - -> B
Other I/O (see text)
(initiated by A)
Factory services are not shown (see text)
Primary Producer Application PP
Primary Producer Service PPS
Primary Producer Resource PPR
Secondary Producer Application SP
Secondary Producer Service SPS
Secondary Producer Resource SPR
Consumer Service CS
Consumer Resource CR
Consumer Application C
Registry Service RS
Schema Service SS
Registry Resource RR
Schema Resource SR
Administration Application A
R-GMA System Components

R-GMA Virtual Database

- R-GMA collect the informations resources of a VO in a *virtual databases* containing a set of tables.
- The Schema contains the name and the definition of all the virtual tables for each VO.
- It is implemented as a database.



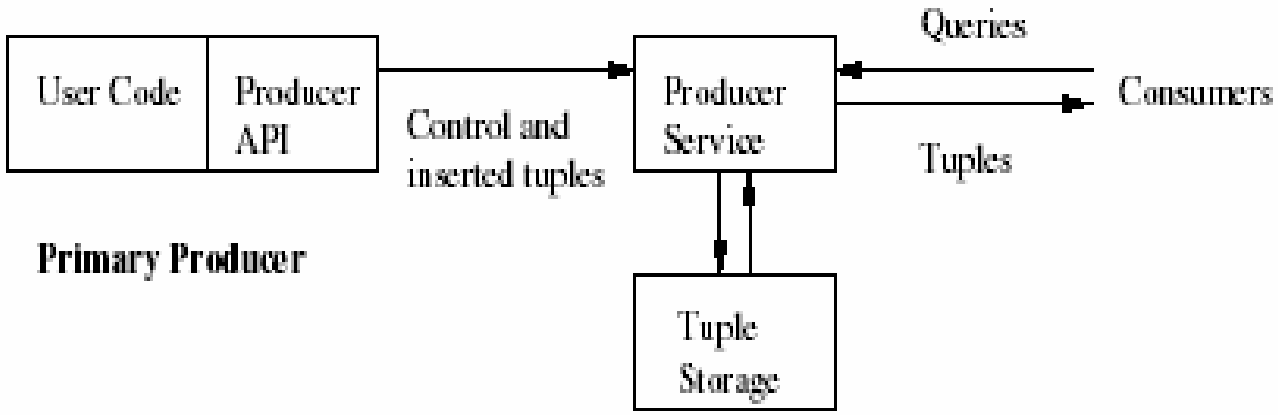
R-GMA In-depth

Producer Types.

- There are two class of producers:
  - Primary & Secondary Producers.
- ..and three different types of query:
  - Continuos.
    - All the new tuples that matching the query are automatically sent to the Consumer as soon as they are available.
  - Latest.
    - Only the latest tuple which representing the “current state” are returned.
  - History.
    - All the tuples which matching the query are returned.

Primary Producer (1)

- In a Primary Producer, the user code periodically inserts tuples into Storage maintained by the Primary Producer itself.
- The Producer Service answers Consumer queries from this Storage.
- The Producer Service is a process running on a server on behalf of the user code.

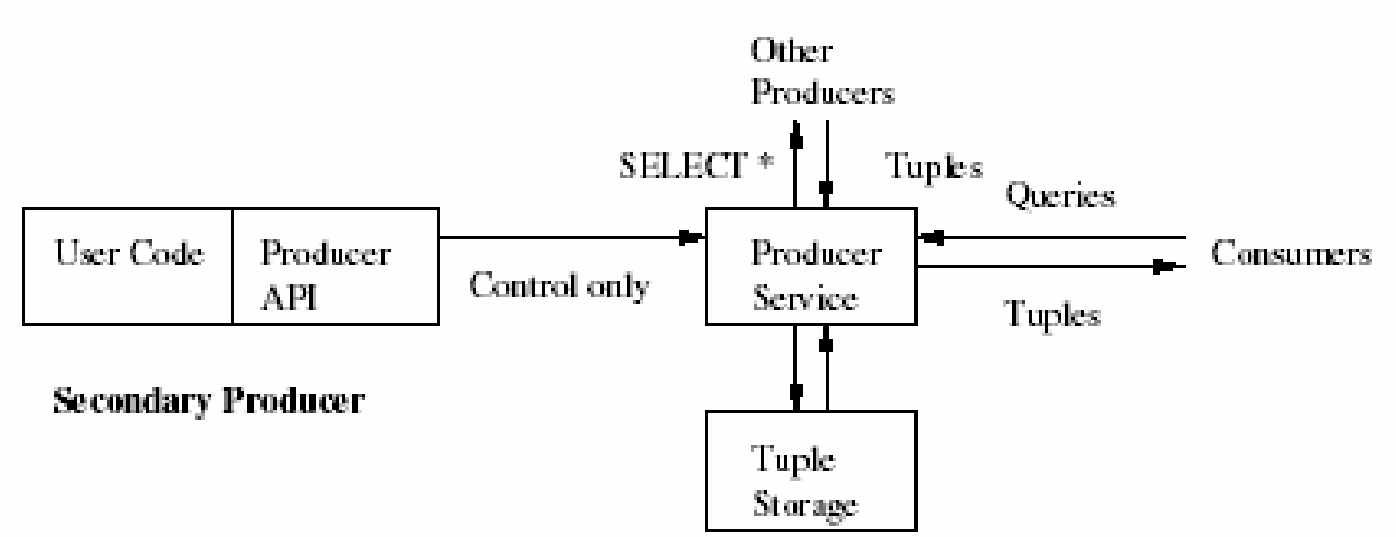


Primary Producer (2)

- A Primary Producer Resource (PPR) is created when a user calls the method createPrimaryProducer of Primary Producer Factory Service.
- Each PPR has a Termination Interval that is a time interval within the user must contact with the Producer Service in order to keep the resource alive and maintain its entry in the Registry.
- The Termination Interval is set by user when the resource is created and can be subsequently changed by calling the method setTerminationInterval.
- The resource is destroyed after the user sends a close or destroy request or when has expired its Termination Interval without any contact from the user.

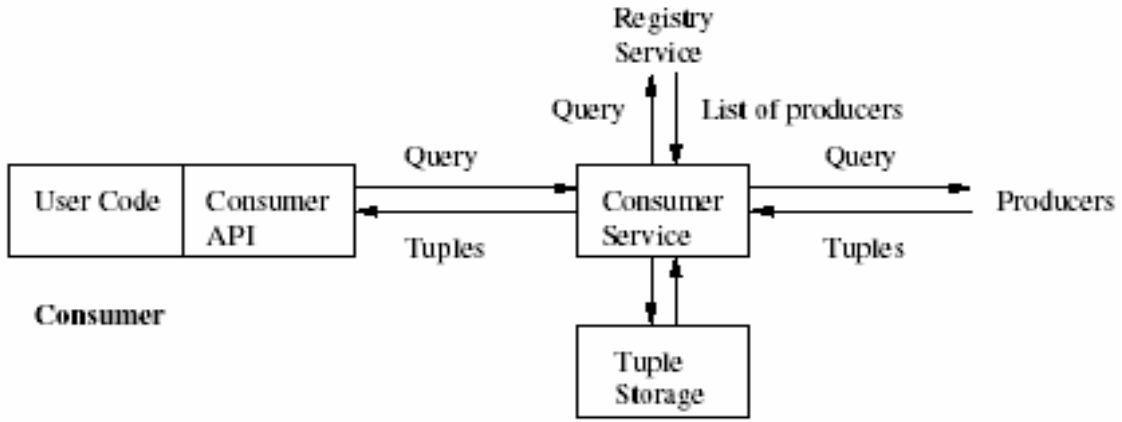
Secondary Producer

- The Secondary Producer populates its Storage with the tuples come from other producers running its own queries.
- The Secondary Producer answers Consumer queries as well as the Primary Producer does.



Consumers (1)

- In R-GMA each Consumers represents a single SQL SELECT statement request.
- The request is initiated by the user code but the Consumer Service carried out all the work on its behalf.
- The query is first passed to the Registry to identify which producers must be contacted to get an answer, then it is passed to each relevant producer to obtain the tuples.



Consumers (2)

- A new Consumer Resource is created when a user calls the Consumer Factory Service's createConsumer methos.
- Each Consumer Resource has a Termination Interval that is a time interval within the user must contact the Consumer Service in order to keep the resource alive and maintain its entry in the Registry.
- The Termination Interval is set by user when the resource is created and can be subsequently changed by calling the method setTerminationInterval.
- The resource is destroyed after the user sends a close or destroy request, or has expired the Termination Interval without any contact from the user.

Registry

- The Registry provides the resource discovery mechanism for R-GMA
  - It allows producers to announce their ability to publish rows to a virtual table;
  - It allows consumers to find producers which can answer their queries.
- The Registry is essentially a database which contains, for each VO, a list of publisher who are available to publish tuples.
- Each Producer entry in the Registry has a termination time after which the entry will be unregistered.

Security

- **Security is available in R-GMA**
  - **Uses https instead of http.**
  - **Authentication via Grid Certificates.**
  - **Authorization will be coming soon.**

Using R-GMA.

The R-GMA Browser

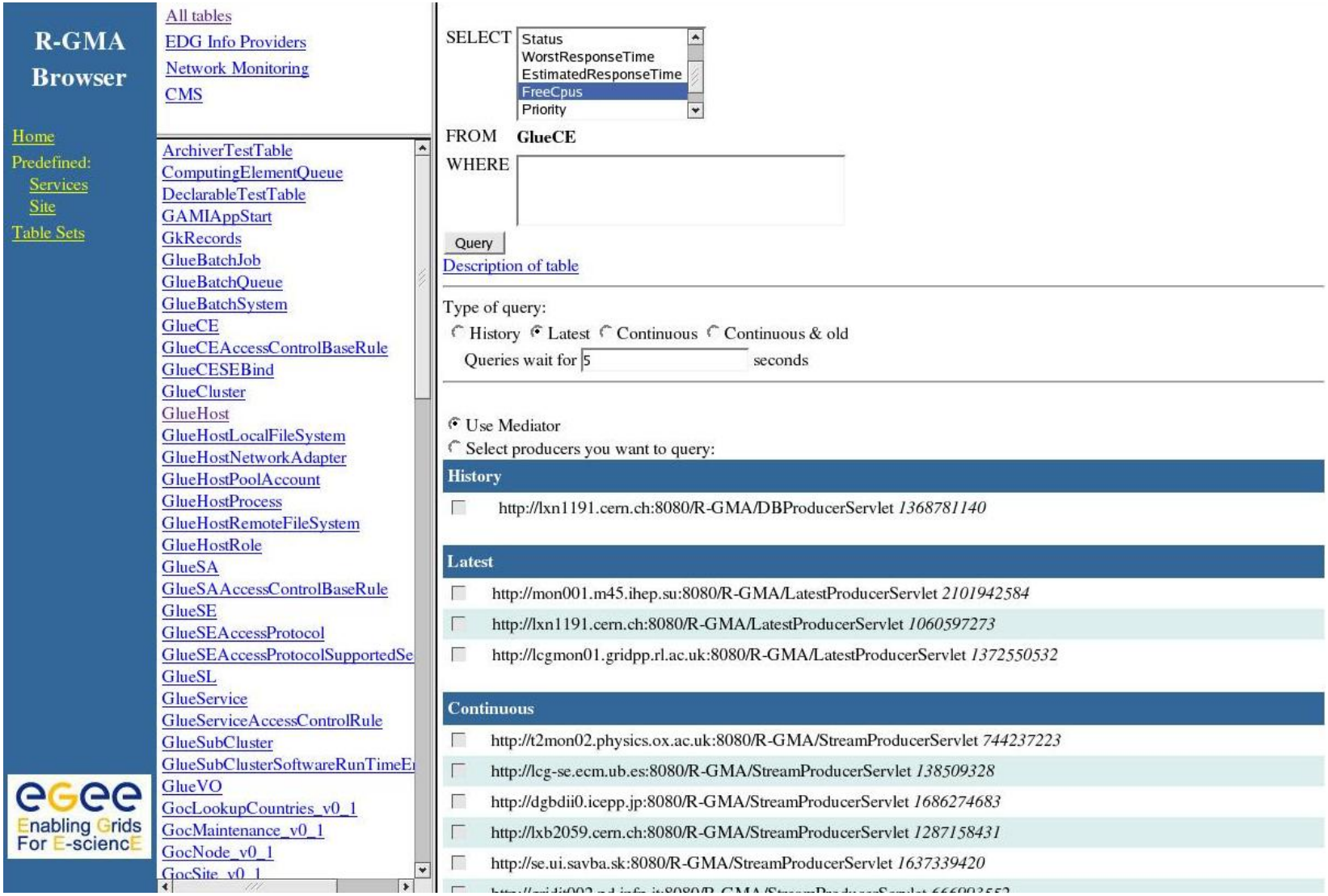
- **The easiest way to try out R-GMA.**
  - It is installed on the machine running the Registry and Schema:

    lcgic01.gridpp.rl.ac.uk:8080/R-GMA/index.html

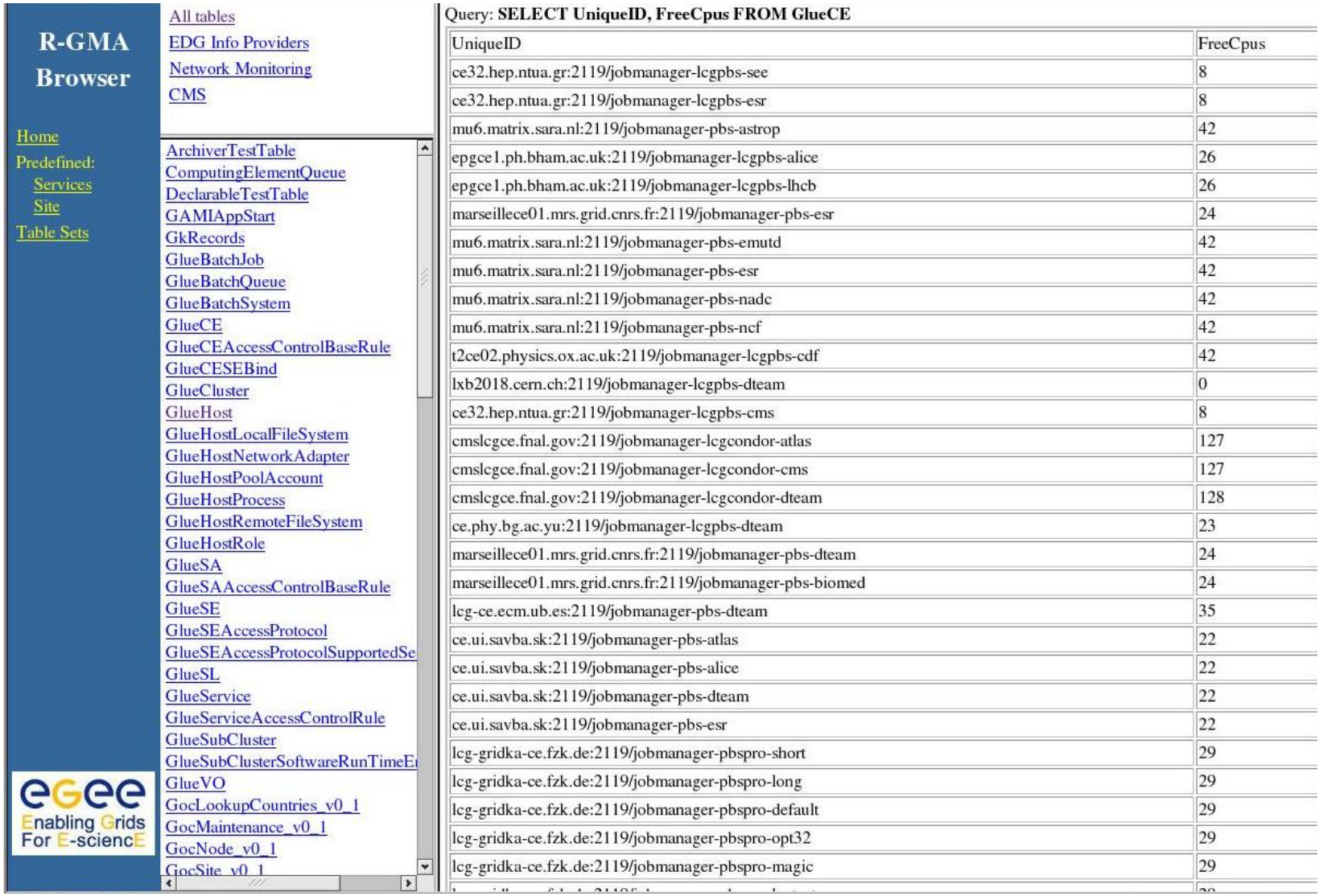
    https://rgmasrv.ct.infn.it:8443/R-GMA/

- **Using the Browser you can do the following.**
  - Browse the tables in the schema.
  - Look at the table definitions.
  - See all the available producers for a table.
  - Query a table.
  - Query only selected producers.

The R-GMA Browser (II)



The R-GMA Browser (III)



Query: **SELECT UniqueID, FreeCpus FROM GlueCE**

UniqueID	FreeCpus
ce32.hep.ntua.gr:2119/jobmanager-lcgpbs-see	8
ce32.hep.ntua.gr:2119/jobmanager-lcgpbs-esr	8
mu6.matrix.sara.nl:2119/jobmanager-pbs-astrop	42
epgce1.ph.bham.ac.uk:2119/jobmanager-lcgpbs-alice	26
epgce1.ph.bham.ac.uk:2119/jobmanager-lcgpbs-lhcb	26
marseillece01.mrs.grid.cnrs.fr:2119/jobmanager-pbs-esr	24
mu6.matrix.sara.nl:2119/jobmanager-pbs-emutd	42
mu6.matrix.sara.nl:2119/jobmanager-pbs-esr	42
mu6.matrix.sara.nl:2119/jobmanager-pbs-nadc	42
mu6.matrix.sara.nl:2119/jobmanager-pbs-ncf	42
t2ce02.physics.ox.ac.uk:2119/jobmanager-lcgpbs-cdf	42
lxb2018.cern.ch:2119/jobmanager-lcgpbs-dteam	0
ce32.hep.ntua.gr:2119/jobmanager-lcgpbs-cms	8
cmslcgce.fnal.gov:2119/jobmanager-lcgcondor-atlas	127
cmslcgce.fnal.gov:2119/jobmanager-lcgcondor-cms	127
cmslcgce.fnal.gov:2119/jobmanager-lcgcondor-dteam	128
ce.phy.bg.ac.yu:2119/jobmanager-lcgpbs-dteam	23
marseillece01.mrs.grid.cnrs.fr:2119/jobmanager-pbs-dteam	24
marseillece01.mrs.grid.cnrs.fr:2119/jobmanager-pbs-biomed	24
lcg-ce.ecm.ub.es:2119/jobmanager-pbs-dteam	35
ce.ui.savba.sk:2119/jobmanager-pbs-atlas	22
ce.ui.savba.sk:2119/jobmanager-pbs-alice	22
ce.ui.savba.sk:2119/jobmanager-pbs-dteam	22
ce.ui.savba.sk:2119/jobmanager-pbs-esr	22
lcg-gridka-ce.fzk.de:2119/jobmanager-pbspro-short	29
lcg-gridka-ce.fzk.de:2119/jobmanager-pbspro-long	29
lcg-gridka-ce.fzk.de:2119/jobmanager-pbspro-default	29
lcg-gridka-ce.fzk.de:2119/jobmanager-pbspro-opt32	29
lcg-gridka-ce.fzk.de:2119/jobmanager-pbspro-magic	29

The R-GMA client CLI

- **R-GMA has a command line interface.**
  - This interface has a similar look an feel to the MySQL DB.
- **To start the R-GMA command line tool, run the following command: $RGMA_HOME/bin/rgma**
- **On startup you should receive the following message:**

```
Welcome to the R-GMA virtual database for Virtual Organisations.
You are connected to the R-GMA registry service at

http://<registry-host>:8080/R-GMA/RegistryServlet

Type "help" for a list of commands.
rgma>
```

The R-GMA client CLI (II)

- **Summary of commands:**

help [<command>]	Information (general or about command)
exit / quit	Exit the R-GMA command line
show [tables | producers of <table> …]	Show the tables in the schema, the producers of a given table…
describe <table>	Show column names and types for <table>
SQL select	Query R-GMA
set query latest | continuous | historical	Set type of query
SQL insert	Insert tuple into the primary producer
Secondaryproducer <table>	Declare table to be consumed and republished by secondary producer
set [secondary]producer latest | continuous | historical	Set supported type for the producer or the secondary producer
set [timeout | maxage] <timeout> [<units>]	Set timeout for queries or maximum age of tuples to return

R-GMA APIs

- **APIs exist in Java, C, C++, Python.**
  - For clients (servlets contacted behind the scenes)
- **They include methods for…**
  - Creating consumers
  - Creating primary and secondary producers
  - Setting type of queries, type of produces, time outs…
  - Retrieving tuples, inserting data
- **You can create your own Producer or Consumer.**
- **Documentation exists for all APIs.**
  - Read the documentation!!
  - Example code is in the documentation.

Links

- **R-GMA overview page.**
  - http://www.r-gma.org/
- **R-GMA in EGEE**
  - http://hepunx.rl.ac.uk/egee/jra1-uk/
- **R-GMA Documenation**
  - http://hepunx.rl.ac.uk/egee/jra1-uk/LCG/doc/