Information System

Valeria Ardizzone
INFN

EGEE NA4 Generic Applications Meeting
Catania, 09-11 January 2006

Outline

- Introduction to R-GMA and Grid Monitoring Architecture (GMA).
- R-GMA within Testbed
- R-GMA in depth:
  - Schema, Registry, Producer and Consumer
  - Query and Storage Types
  - R-GMA Browser
- A use case of R-GMA for user application.

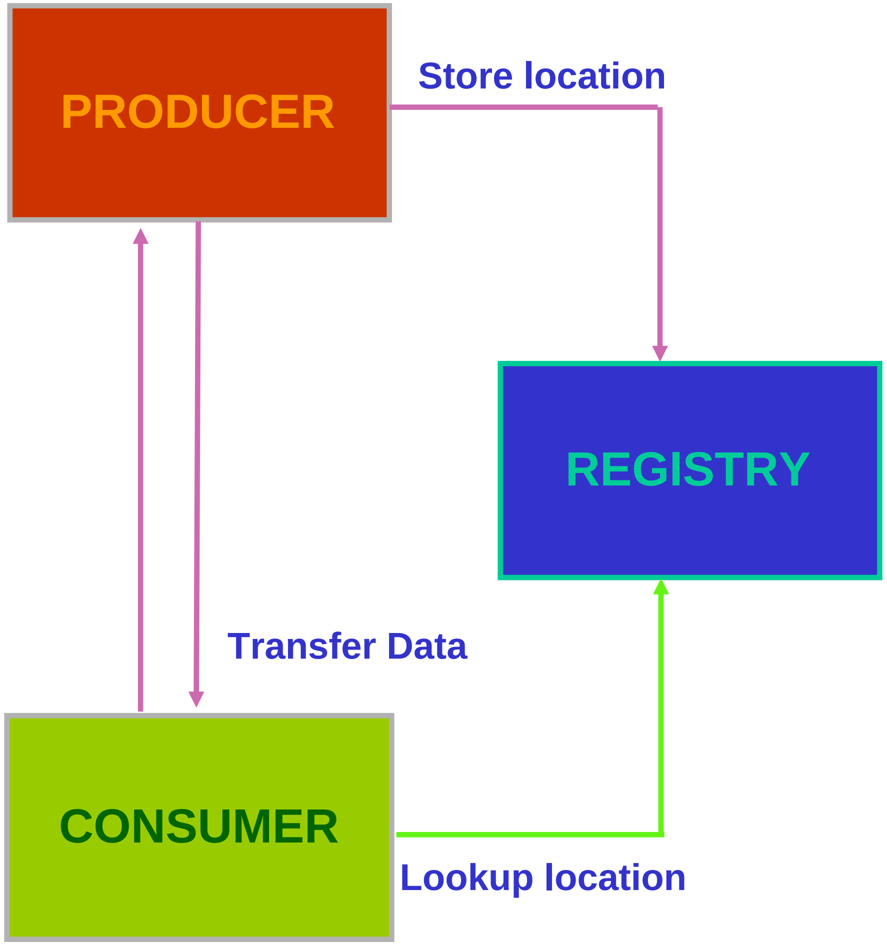
Introduction to R-GMA

- **Relational Grid Monitoring Architecture (R-GMA)**
  - Developed as part of the EuropeanDataGrid Project (EDG)
  - Now as part of the EGEE project.
  - Based the Grid Monitoring Architecture (GMA) from the Global Grid Forum (GGF).

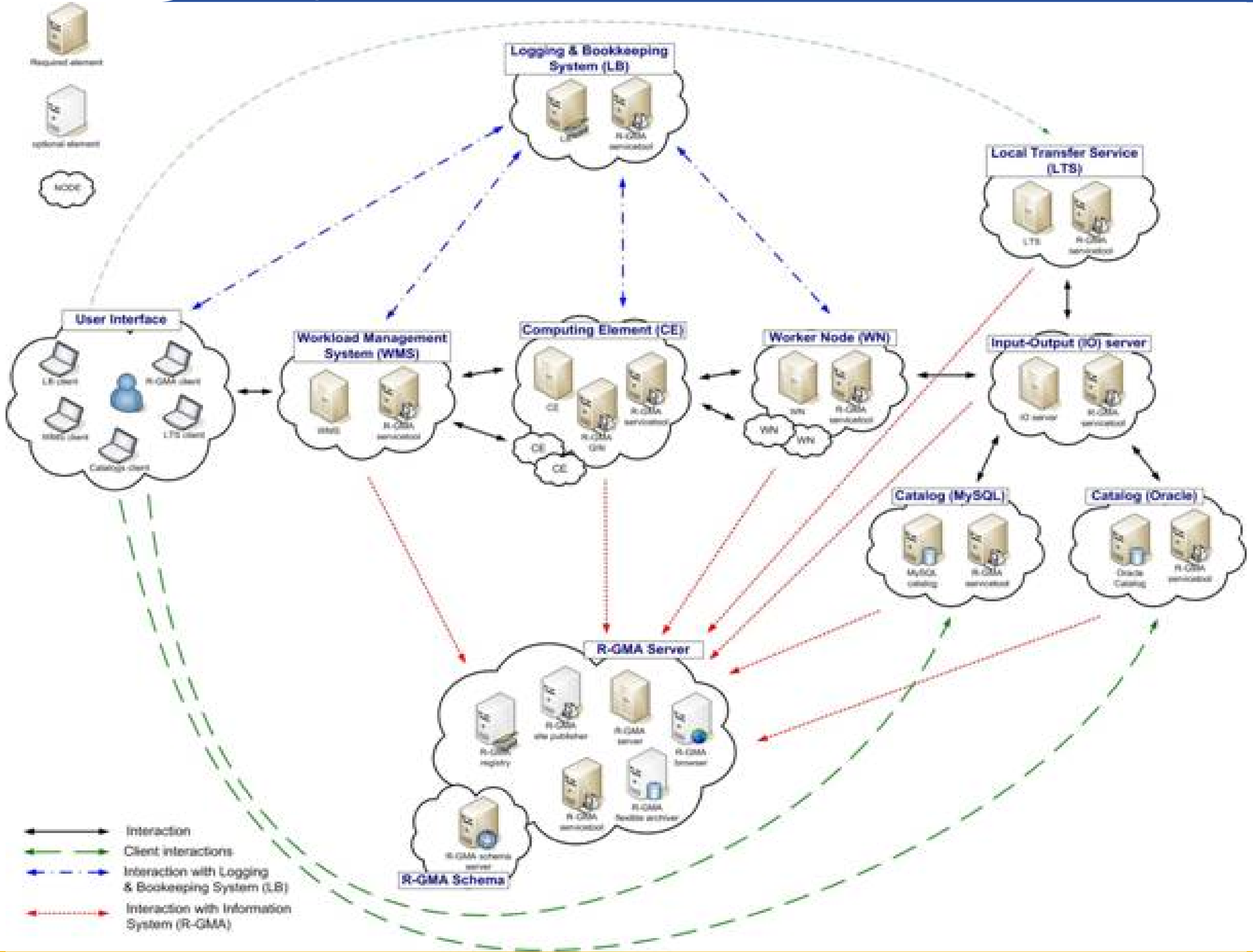
- **Uses a relational data model.**
  - Data are viewed as tables.
  - Data structure defined by the columns.
  - Each entry is a row (tuple).
  - Queried using Structured Query Language (SQL).

Grid Monitoring Architecture(GMA)

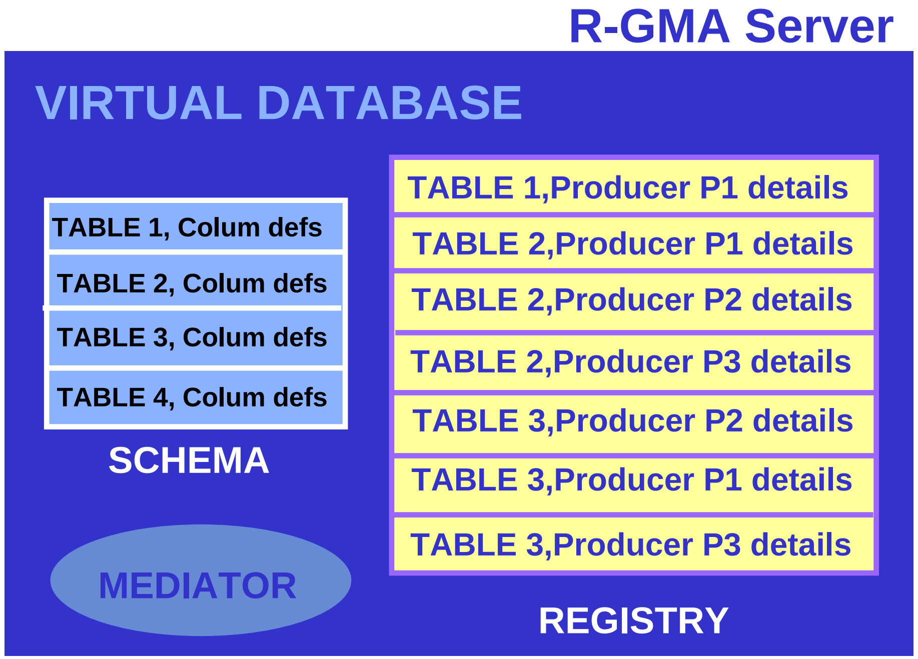
- The Producer stores its location (URL) in the Registry.
- The Consumer looks up producer URLs in the Registry.
- The Consumer contacts the Producer to get all the data or the Consumer can listen to the Producer for new data.



R-GMA within Testbed



R-GMA: Schema-Registry-Mediator


R-GMA Server
VIRTUAL DATABASE
TABLE 1, Colum defs
TABLE 2, Colum defs
TABLE 3, Colum defs
TABLE 4, Colum defs
SCHEMA
TABLE 1,Producer P1 details
TABLE 2,Producer P1 details
TABLE 2,Producer P2 details
TABLE 2,Producer P3 details
TABLE 3,Producer P2 details
TABLE 3,Producer P1 details
TABLE 3,Producer P3 details
REGISTRY
MEDIATOR

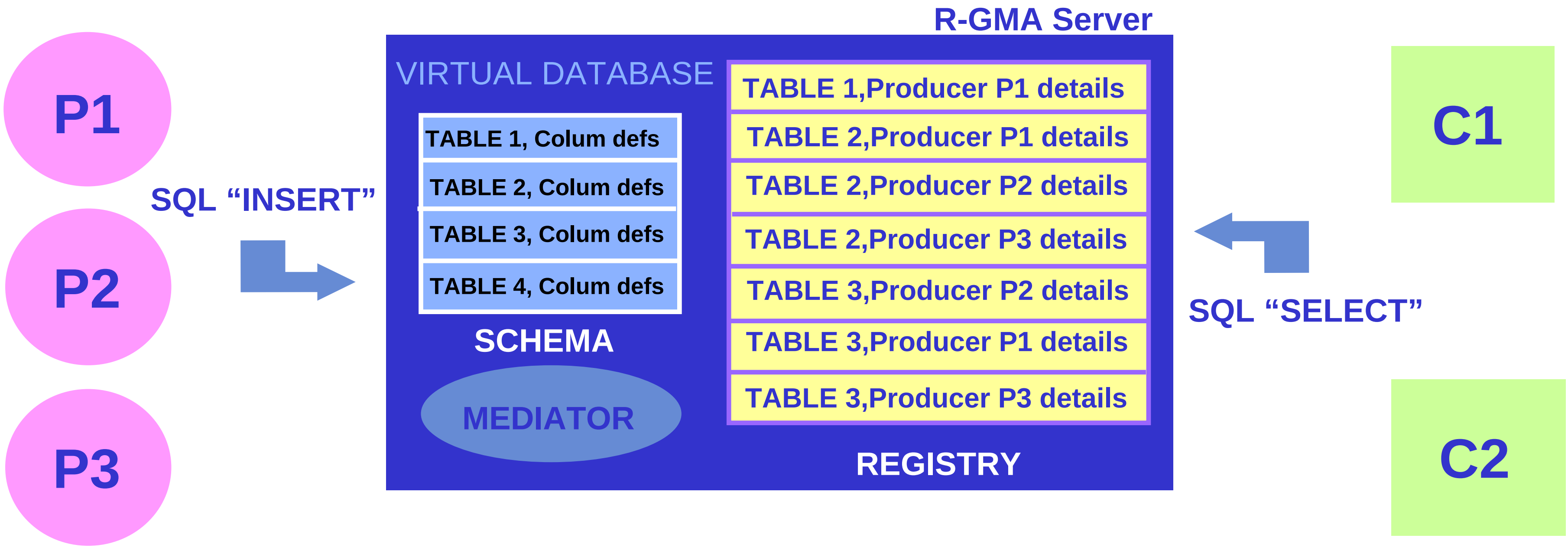
SCHEMA : it holds the names and definitions of all of the tables in the virtual database, and their authorization rules.

REGISTRY: It holds the details of all producers that are publishing to tables in the virtual database and it also holds the details of "continuous" consumers.

MEDIATOR: a set of rules for deciding which data providers to contact for any given query.

R-GMA: Producer-Consumer

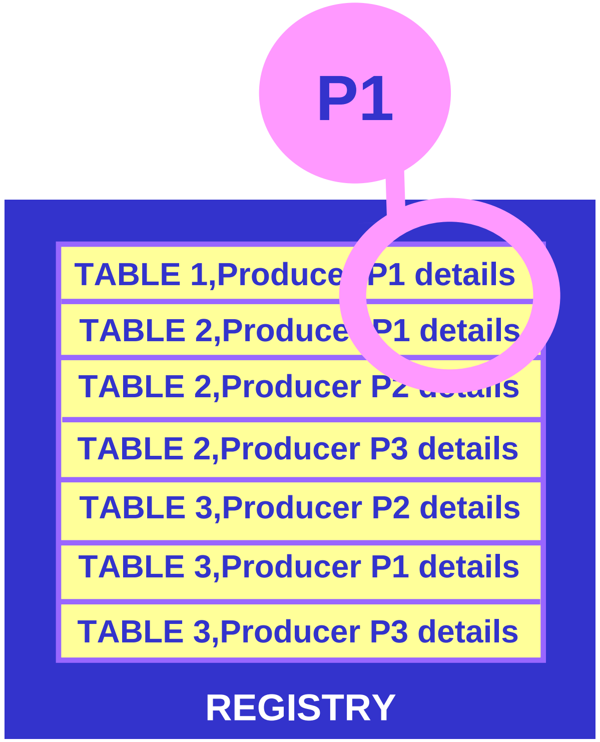
Producers: are the data providers for the virtual database. Writing data into the virtual database is known as publishing, and data is always published in complete rows, known as tuples. There are three types of producer: Primary, Secondary and On-demand.


R-GMA Server
P1
P2
P3
SQL “INSERT”
VIRTUAL DATABASE
TABLE 1, Colum defs
TABLE 2, Colum defs
TABLE 3, Colum defs
TABLE 4, Colum defs
SCHEMA
MEDIATOR
TABLE 1,Producer P1 details
TABLE 2,Producer P1 details
TABLE 2,Producer P2 details
TABLE 2,Producer P3 details
TABLE 3,Producer P2 details
TABLE 3,Producer P1 details
TABLE 3,Producer P3 details
REGISTRY
SQL “SELECT”
C1
C2

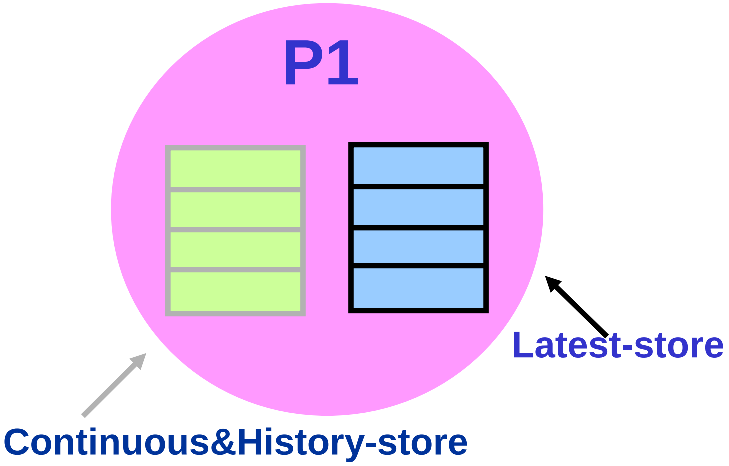
Consumer: represents a single SQL SELECT query on the virtual database. The query is matched against the list of available producers in the Registry. The consumer service then selects the best set of producers to contact and sends the query directly to each of them, to obtain the answer tuples.

Query and Storage Types

- **Continuous:** as soon as new data becomes available it is broadcast to all interested parties.
- **Latest:** correspond to intuitive idea of current information.
- **History:** return time sequenced data.


P1
TABLE 1,Producer P1 details
TABLE 2,Producer P1 details
TABLE 2,Producer P2 details
TABLE 2,Producer P3 details
TABLE 3,Producer P2 details
TABLE 3,Producer P1 details
TABLE 3,Producer P3 details
REGISTRY

Tuple-store can be in Memory or Database



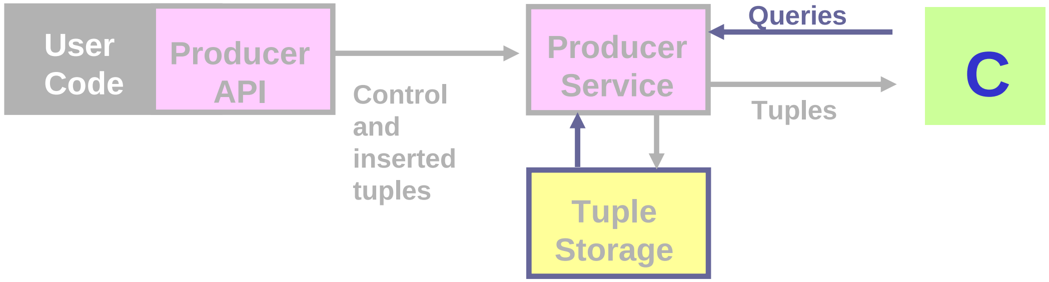
LATEST RETENTION PERIOD (LRP) and

HISTORY RETENTION PERIOD (RTP)

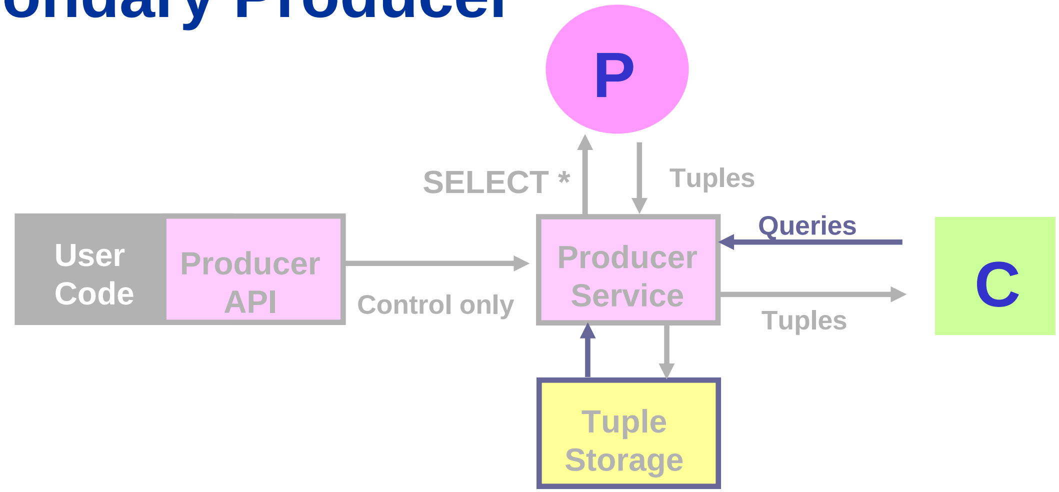
allow producers to periodically purge old tuples, and to give a precise meaning to the “current state”.

Producer Types

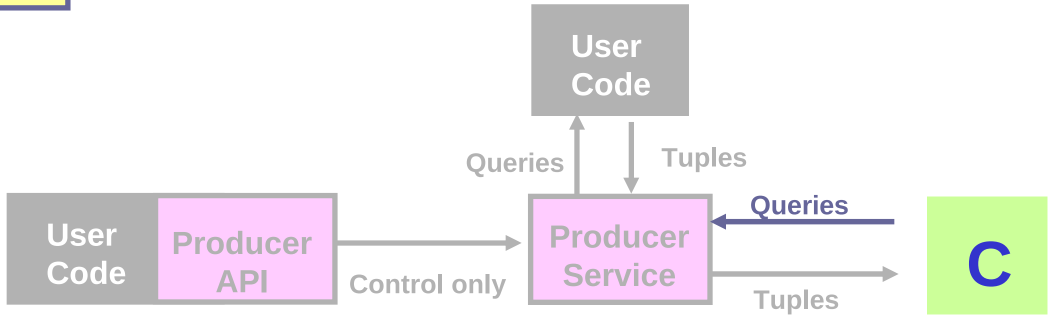
- Primary Producer



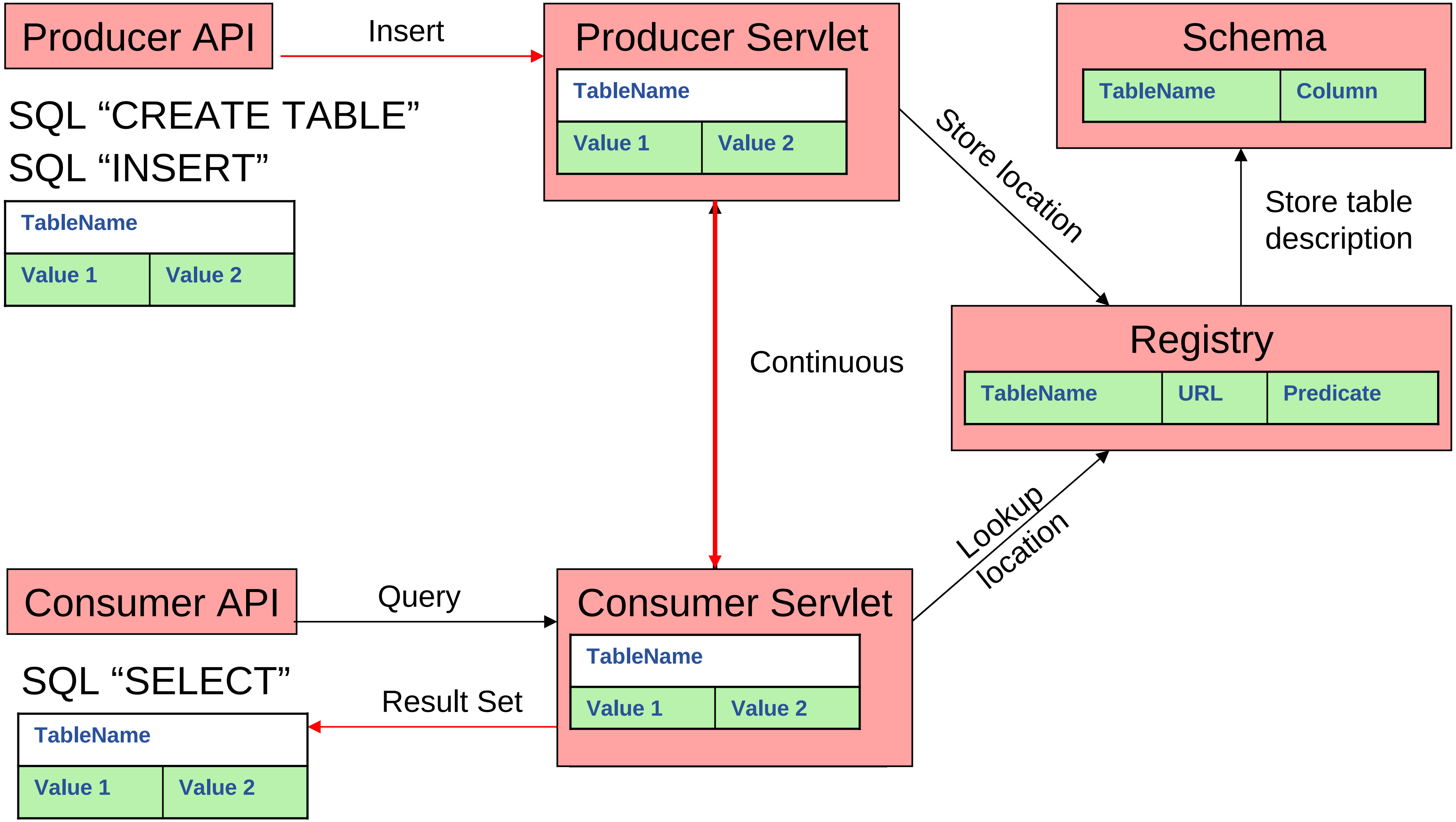
- Secondary Producer



- On-Demand Producer



Continuous



Producer API

Insert

Producer Servlet

TableName	
Value 1	Value 2

Schema

TableName	Column

SQL “CREATE TABLE”

SQL “INSERT”

TableName	
Value 1	Value 2

Store location

Store table description

Continuous

Registry

TableName	URL	Predicate

Lookup location

Consumer API

Query

Consumer Servlet

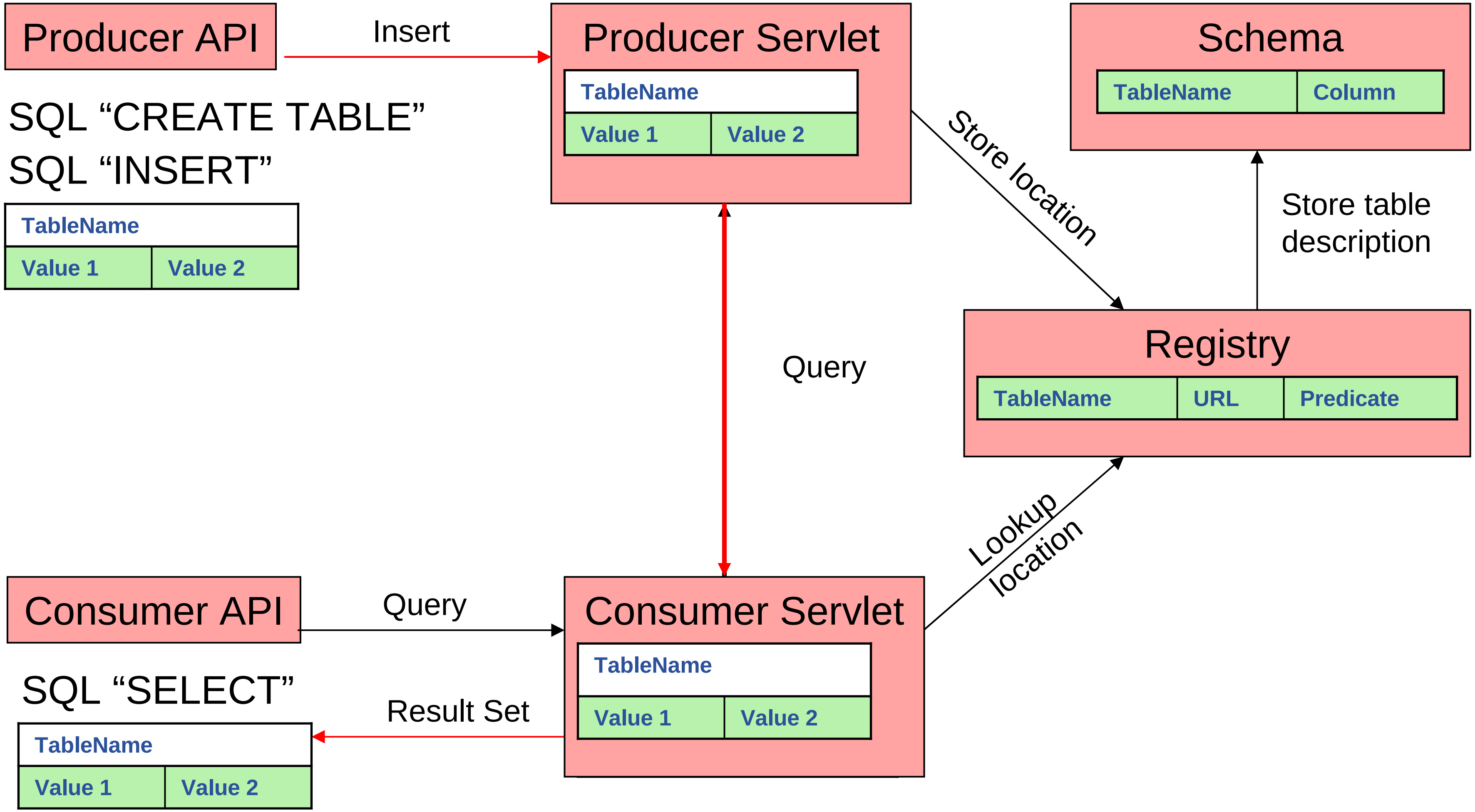
TableName	
Value 1	Value 2

SQL “SELECT”

Result Set

TableName	
Value 1	Value 2

History or Latest



Producer API

Insert

SQL “CREATE TABLE”

SQL “INSERT”

TableName	
Value 1	Value 2

Producer Servlet

TableName	
Value 1	Value 2

Store location

Schema

TableName	Column

Store table description

Query

Registry

TableName	URL	Predicate

Lookup location

Consumer API

Query

Consumer Servlet

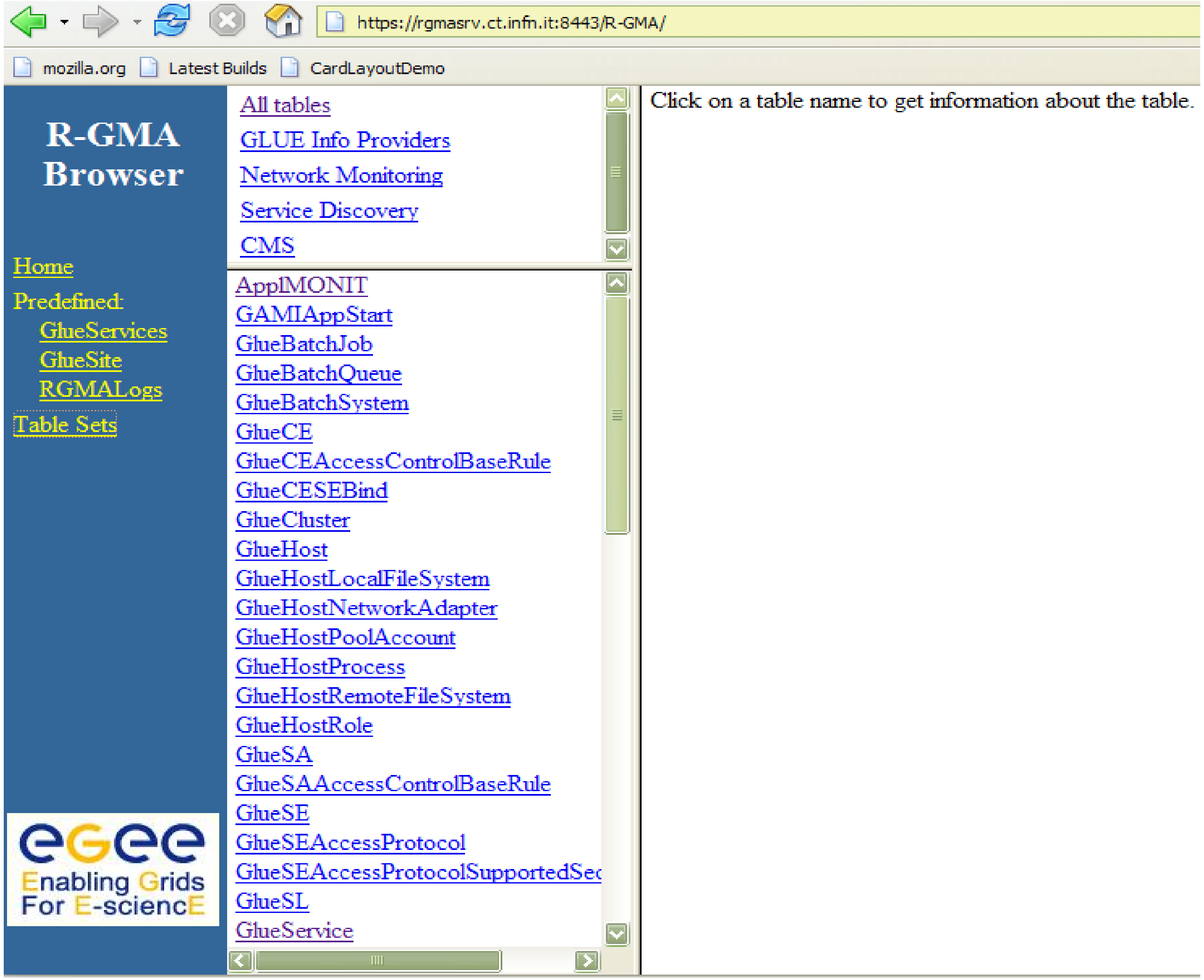
TableName	
Value 1	Value 2

SQL “SELECT”

Result Set

TableName	
Value 1	Value 2

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



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

R-GMA Browser

- Home
- Predefined:
  - GlueServices
  - GlueSite
  - RGMALogs
- Table Sets

All tables

GLUE Info Providers

Network Monitoring

Service Discovery

CMS

ApplMONIT

GAMIAppStart

GlueBatchJob

GlueBatchQueue

GlueBatchSystem

GlueCE

GlueCEAccessControlBaseRule

GlueCESEBind

GlueCluster

GlueHost

GlueHostLocalFileSystem

GlueHostNetworkAdapter

GlueHostPoolAccount

GlueHostProcess

GlueHostRemoteFileSystem

GlueHostRole

GlueSA

GlueSAAccessControlBaseRule

GlueSE

GlueSEAccessProtocol

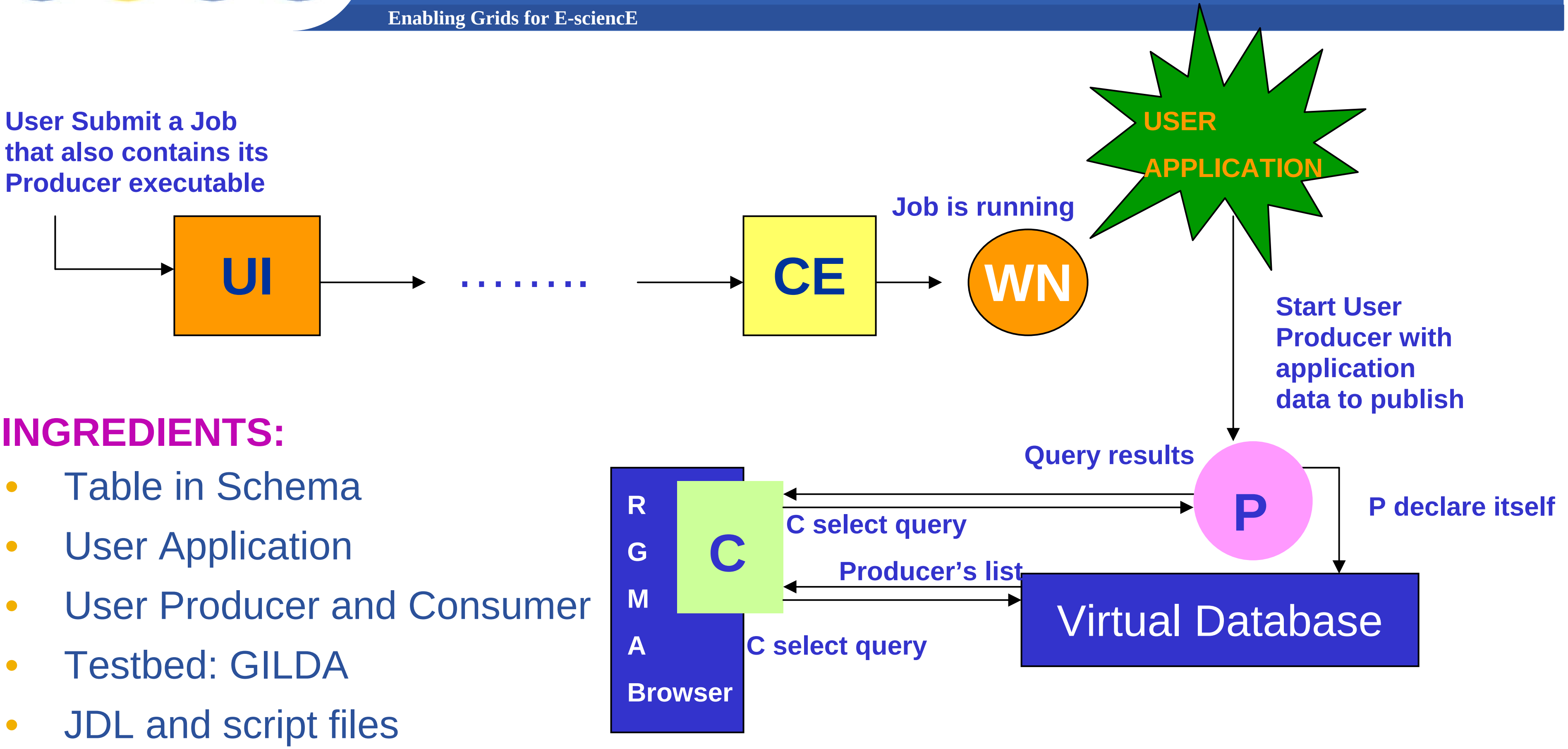
GlueSEAccessProtocolSupportedSec

GlueSL

GlueService

Click on a table name to get information about the table.

R-GMA use case for User Application

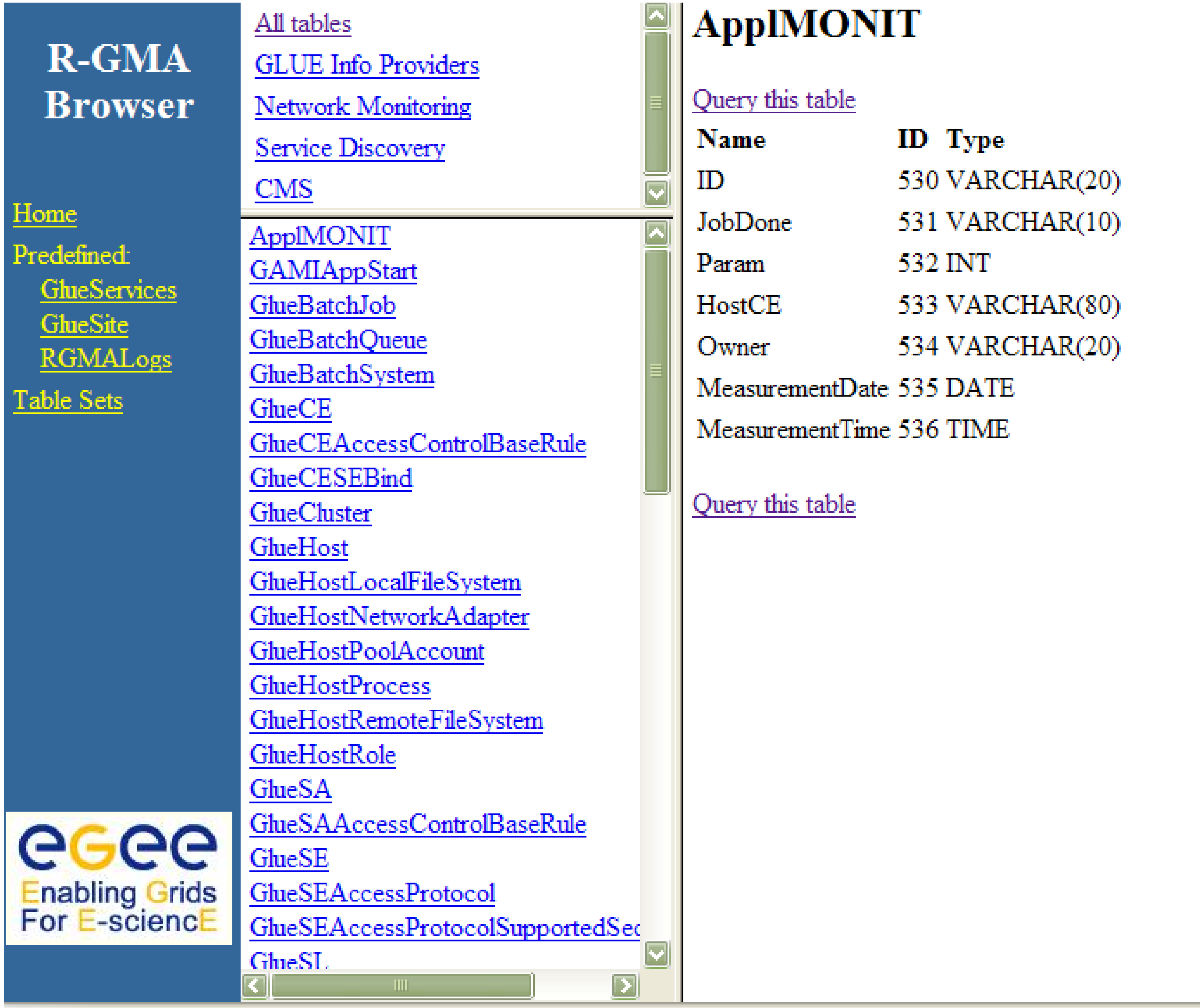


INGREDIENTS:

- Table in Schema
- User Application
- User Producer and Consumer
- Testbed: GILDA
- JDL and script files

USE CASE TIMELINE: To submit the JDL file from the GENIUS portal and monitoring its status. In the meantime, from RGMA Browser, monitoring the table and if there is any producers that are publishing tuples. If there is one, to send a query with a predicate to obtain the answer tuples.

Create a table in Schema



R-GMA Browser

Home

Predefined:

GlueServices

GlueSite

RGMALogs

Table Sets

All tables

GLUE Info Providers

Network Monitoring

Service Discovery

CMS

ApplMONIT
GAMIAppStart
GlueBatchJob
GlueBatchQueue
GlueBatchSystem
GlueCE
GlueCEAccessControlBaseRule
GlueCESEBind
GlueCluster
GlueHost
GlueHostLocalFileSystem
GlueHostNetworkAdapter
GlueHostPoolAccount
GlueHostProcess
GlueHostRemoteFileSystem
GlueHostRole
GlueSA
GlueSAAccessControlBaseRule
GlueSE
GlueSEAccessProtocol
GlueSEAccessProtocolSupportedSec
GlueSL

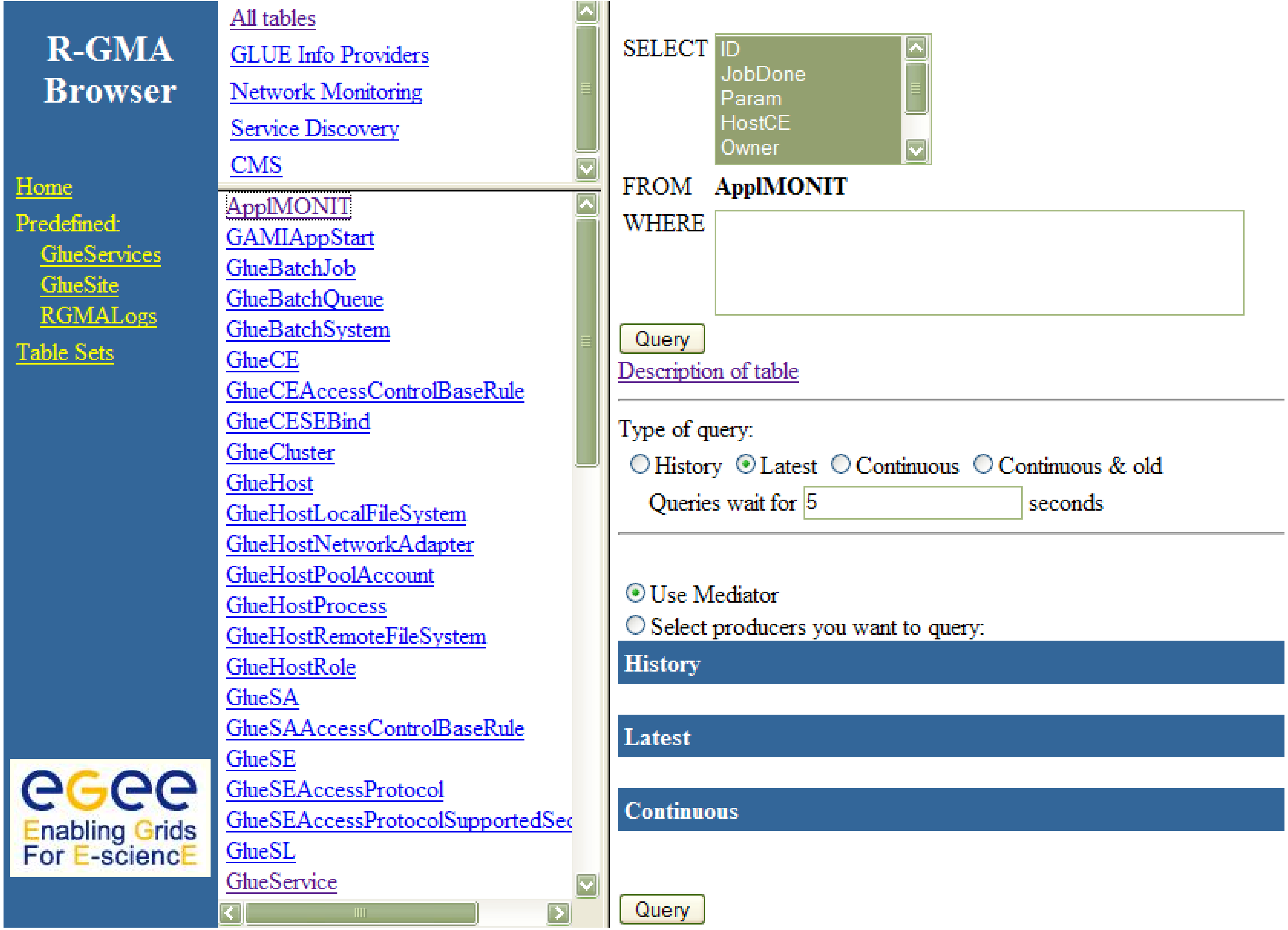
ApplMONIT

Query this table

Name	ID	Type
ID	530	VARCHAR(20)
JobDone	531	VARCHAR(10)
Param	532	INT
HostCE	533	VARCHAR(80)
Owner	534	VARCHAR(20)
MeasurementDate	535	DATE
MeasurementTime	536	TIME

Query this table

R-GMA Browser as Consumer



R-GMA Browser

- Home
- Predefined:
  - GlueServices
  - GlueSite
  - RGMALogs
- Table Sets

Enabling Grids For E-sciencE

All tables
GLUE Info Providers
Network Monitoring
Service Discovery
CMS

ApplMONIT
GAMIAppStart
GlueBatchJob
GlueBatchQueue
GlueBatchSystem
GlueCE
GlueCEAccessControlBaseRule
GlueCESEBind
GlueCluster
GlueHost
GlueHostLocalFileSystem
GlueHostNetworkAdapter
GlueHostPoolAccount
GlueHostProcess
GlueHostRemoteFileSystem
GlueHostRole
GlueSA
GlueSAAccessControlBaseRule
GlueSE
GlueSEAccessProtocol
GlueSEAccessProtocolSupportedSec
GlueSL
GlueService

SELECT ID
JobDone
Param
HostCE
Owner

FROM **ApplMONIT**

WHERE

Description of table

Type of query:

○ History ● Latest ○ Continuous ○ Continuous & old

Queries wait for 5 seconds

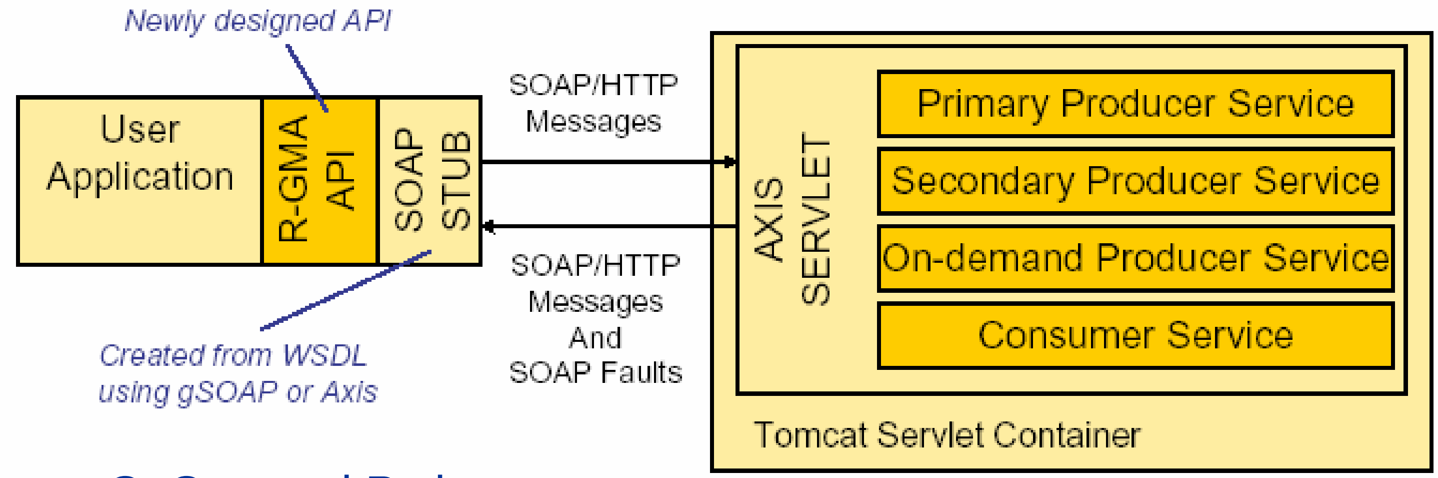
● Use Mediator
○ Select producers you want to query:

History

Latest

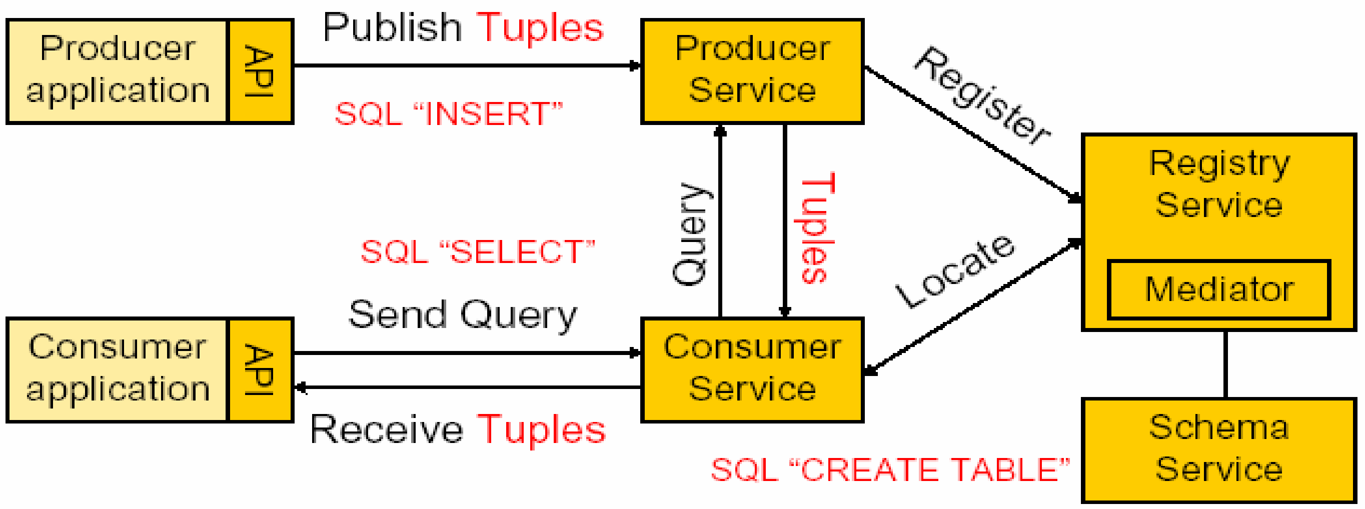
Continuous

User Producer and Consumer



API available for Java, C, C++ and Python

Users may by-pass API if they wish, but API is the easiest way to use R-GMA services



Producer Application (in Java)(1)

Producer Properties

Type: **Primary**

Storage type: **Database**

Termination Interval: 300 (seconds)

Predicate: **Where …**

Query type: **HISTORY**

Latest Retention Period: 300 (seconds)

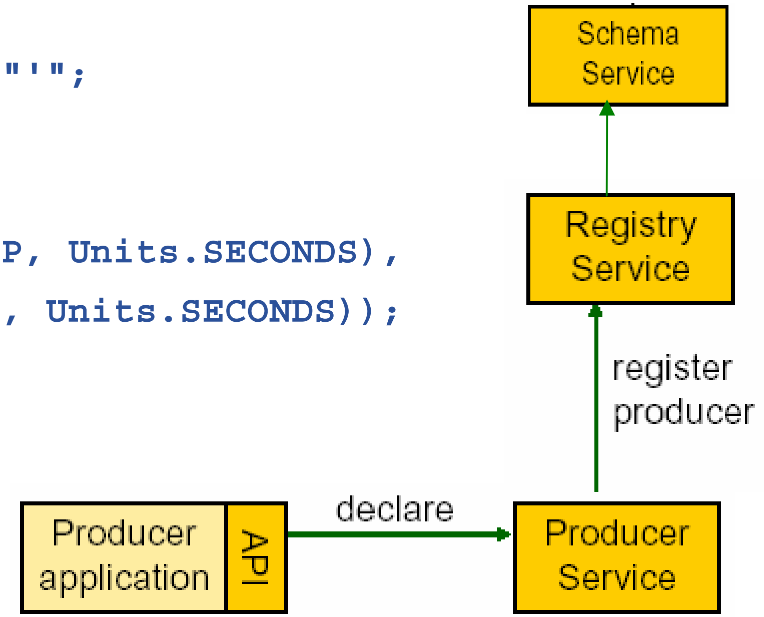
History Retention Period: 300 (seconds)

Producer application

```
. . . . . . . . . .
ProducerProperties producerProps = null;
if (producerType.equals("CONTINUOUS"))
{   producerProps = new ProducerProperties(Storage.MEMORY, 0);   }
else if (producerType.equals("LATEST"))
{   producerProps = new ProducerProperties(Storage.DATABASE, ProducerProperties.LATEST);   }
else if (producerType.equals("HISTORY"))
{    producerProps = new ProducerProperties(Storage.DATABASE, ProducerProperties.HISTORY); }
else
{    System.err.println("Invalid producer type (" + producerType + ").");
     System.exit(1);
}
```

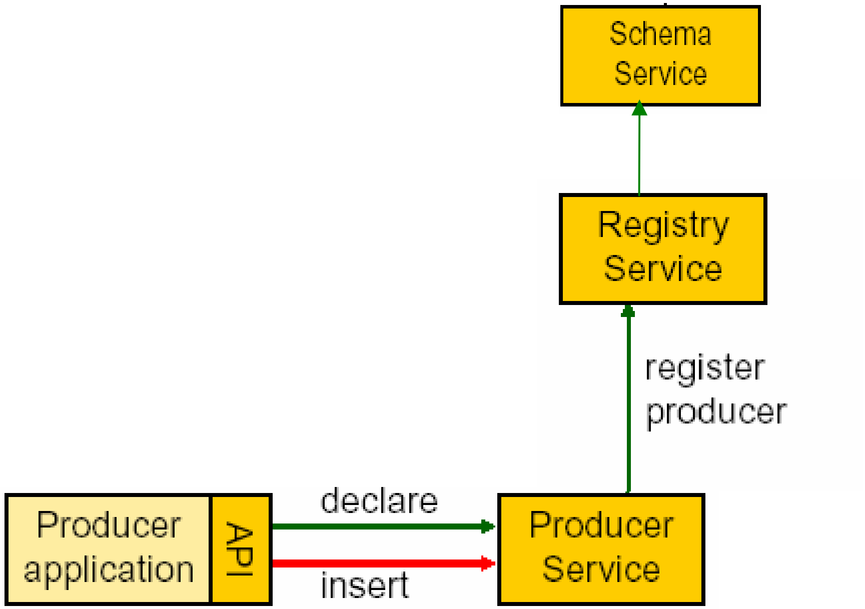
Producer Application (in Java)(2)

```
. . . . . . . . . .
PrimaryProducer pp = null;
ResourceEndpoint endpoint = null;
Try
{  ProducerFactory pf = new ProducerFactoryStub();
   TimeInterval ti = new TimeInterval(terminationInterval, Units.SECONDS);
   pp = pf.createPrimaryProducer(ti, producerProps, null);
   endpoint = pp.getResourceEndpoint();
   String predicate = "WHERE ID = '" + Id + "'";
   pp.declareTable(tableName,
                   predicate,
                   new TimeInterval(historyRP, Units.SECONDS),
                   new TimeInterval(latestRP, Units.SECONDS));
```



Producer Application (in Java)(3)

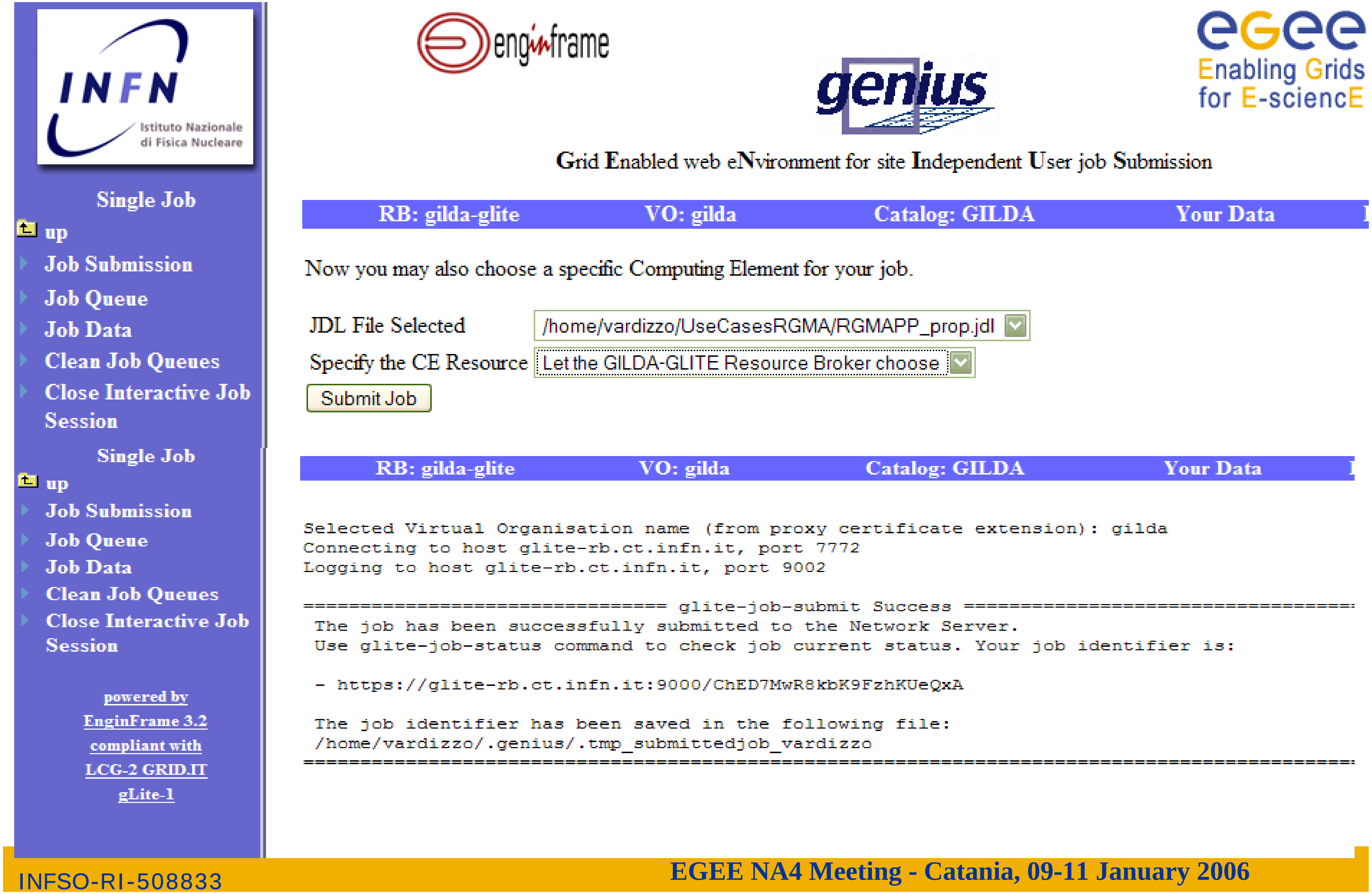
```
. . . . . . . . . .

String insert = "INSERT INTO "+ tableName +
     " (ID, JobDone, Param, HostCE, Owner) VALUES ('"
     + Id + "','" + per + "','" + i + "','" + hostce + "','" + owner + "')";

pp.insert(insert);
. . . . . . . . . .
pp.close();
```



JDL with User Producer Application

```
[
Type = "Job";
JobType = "Normal";
Executable="startPP.sh";
Arguments = "100 HISTORY Valeria_Ardizzone";
StdOutput="stdout.log";
StdError="stderr.log";
InputSandbox={"startPP.sh","pp.class"};
OutputSandbox={"stdout.log","stderr.log"};
…….
]
```

JDL Submission from GENIUS



Grid Enabled web eNvironment for site Independent User job Submission

Now you may also choose a specific Computing Element for your job.

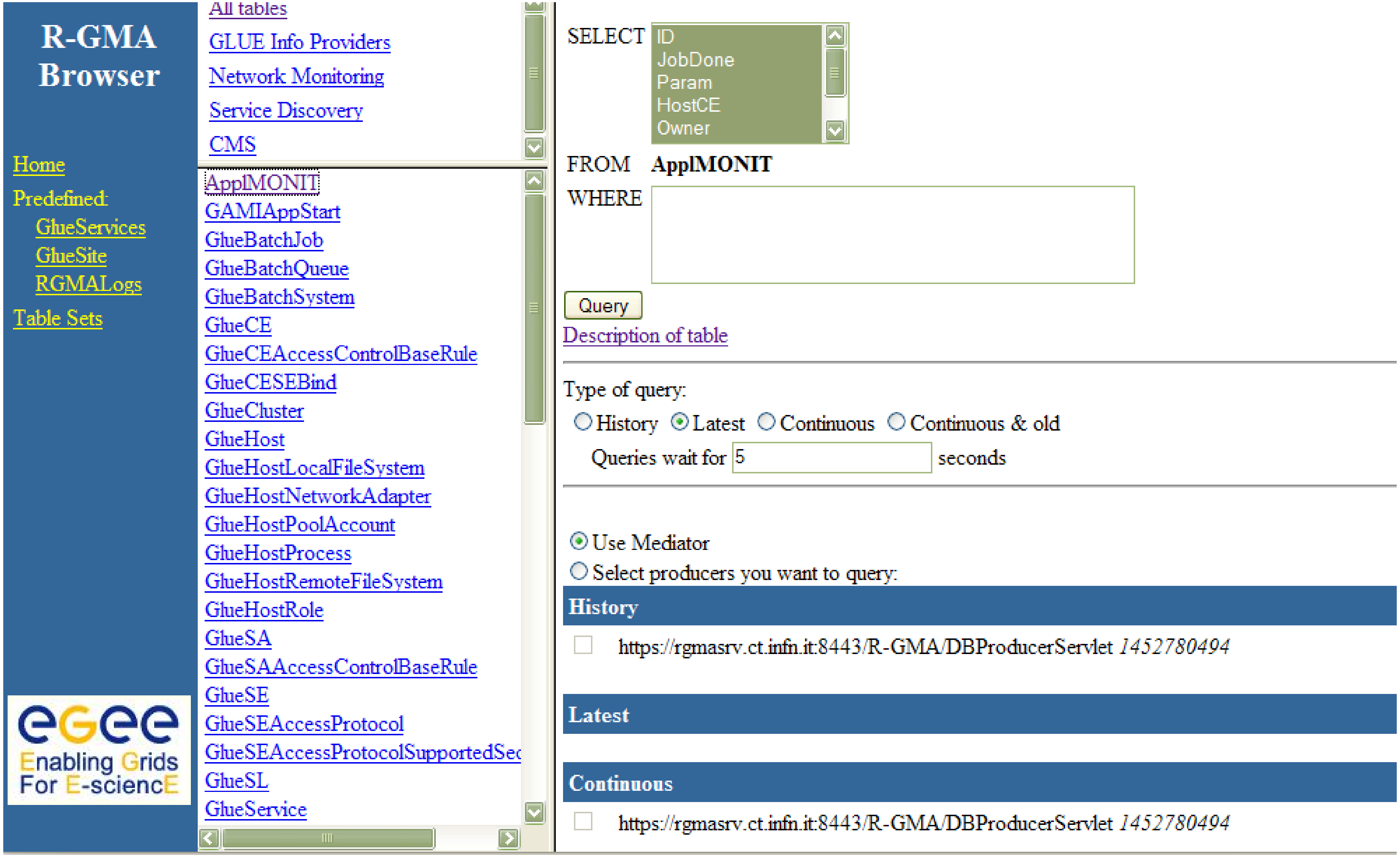
JDL File Selected: /home/vardizzo/UseCasesRGMA/RGMAPP_prop.jdl

Specify the CE Resource: Let the GILDA-GLITE Resource Broker choose

```
Selected Virtual Organisation name (from proxy certificate extension): gilda
Connecting to host glite-rb.ct.infn.it, port 7772
Logging to host glite-rb.ct.infn.it, port 9002

================================ glite-job-submit Success ==================================
 The job has been successfully submitted to the Network Server.
 Use glite-job-status command to check job current status. Your job identifier is:

 - https://glite-rb.ct.infn.it:9000/ChED7MwR8kbK9FzhKUeQxA

 The job identifier has been saved in the following file:
 /home/vardizzo/.genius/.tmp_submittedjob_vardizzo
=============================================================================================
```

User Producer in R-GMA Browser

Enabling Grids for E-sciencE



R-GMA Browser

Home

Predefined:

GlueServices

GlueSite

RGMALogs

Table Sets

All tables

GLUE Info Providers

Network Monitoring

Service Discovery

CMS

ApplMONIT

GAMIAppStart

GlueBatchJob

GlueBatchQueue

GlueBatchSystem

GlueCE

GlueCEAccessControlBaseRule

GlueCESEBind

GlueCluster

GlueHost

GlueHostLocalFileSystem

GlueHostNetworkAdapter

GlueHostPoolAccount

GlueHostProcess

GlueHostRemoteFileSystem

GlueHostRole

GlueSA

GlueSAAccessControlBaseRule

GlueSE

GlueSEAccessProtocol

GlueSEAccessProtocolSupportedSec

GlueSL

GlueService

SELECT ID, JobDone, Param, HostCE, Owner

FROM **ApplMONIT**

WHERE

Query

Description of table

Type of query:

○ History ● Latest ○ Continuous ○ Continuous & old

Queries wait for 5 seconds

● Use Mediator

○ Select producers you want to query:

History

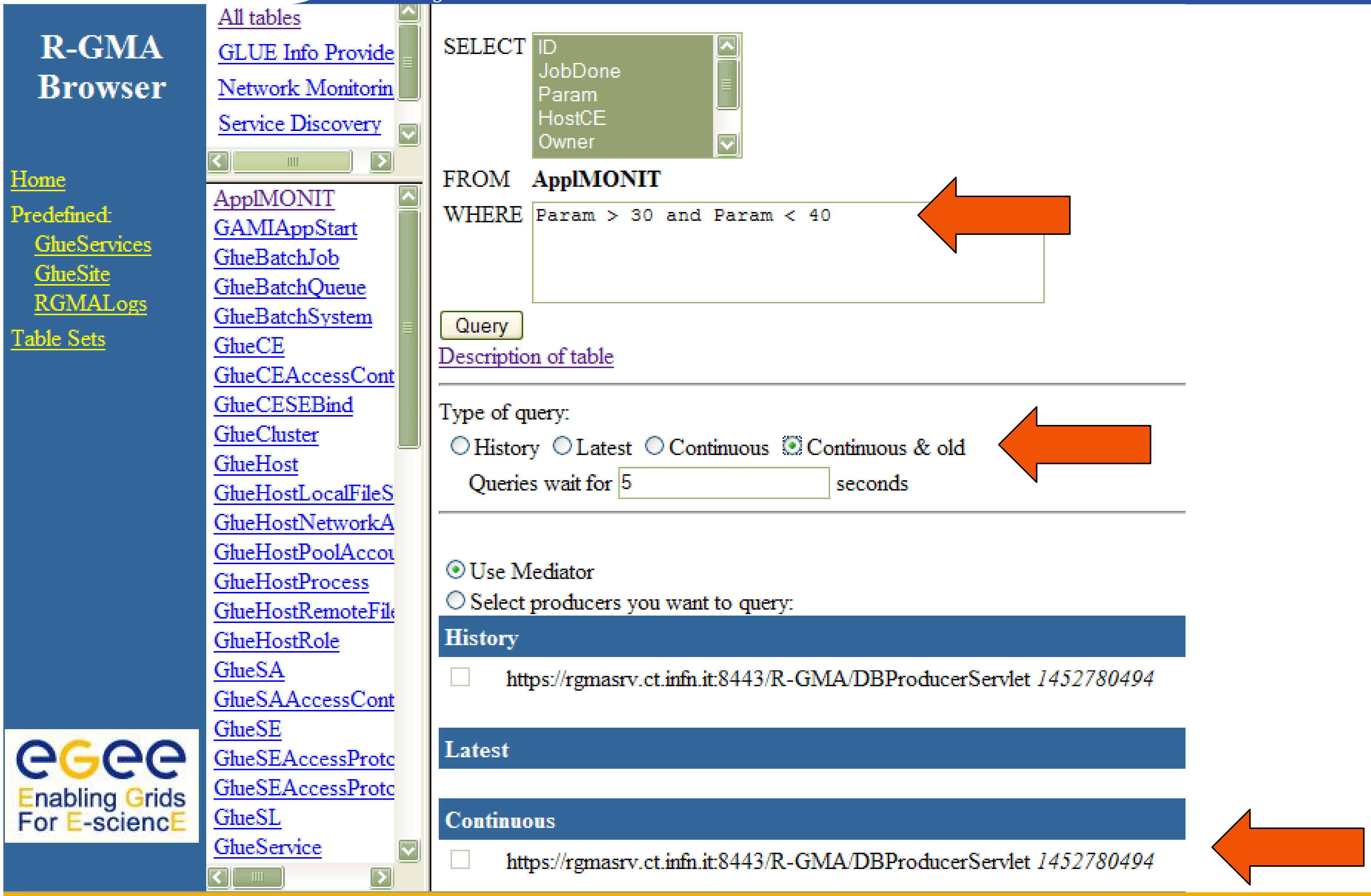
□ https://rgmasrv.ct.infn.it:8443/R-GMA/DBProducerServlet *1452780494*

Latest

Continuous

□ https://rgmasrv.ct.infn.it:8443/R-GMA/DBProducerServlet *1452780494*

Query from R-GMA Browser



R-GMA Browser

Home

Predefined:

- GlueServices
- GlueSite
- RGMALogs

Table Sets

SELECT ID JobDone Param HostCE Owner

FROM **ApplMONIT**

WHERE `Param > 30 and Param < 40`

Query

Description of table

Type of query:

○ History ○ Latest ○ Continuous ● Continuous & old

Queries wait for 5 seconds

● Use Mediator

○ Select producers you want to query:

History

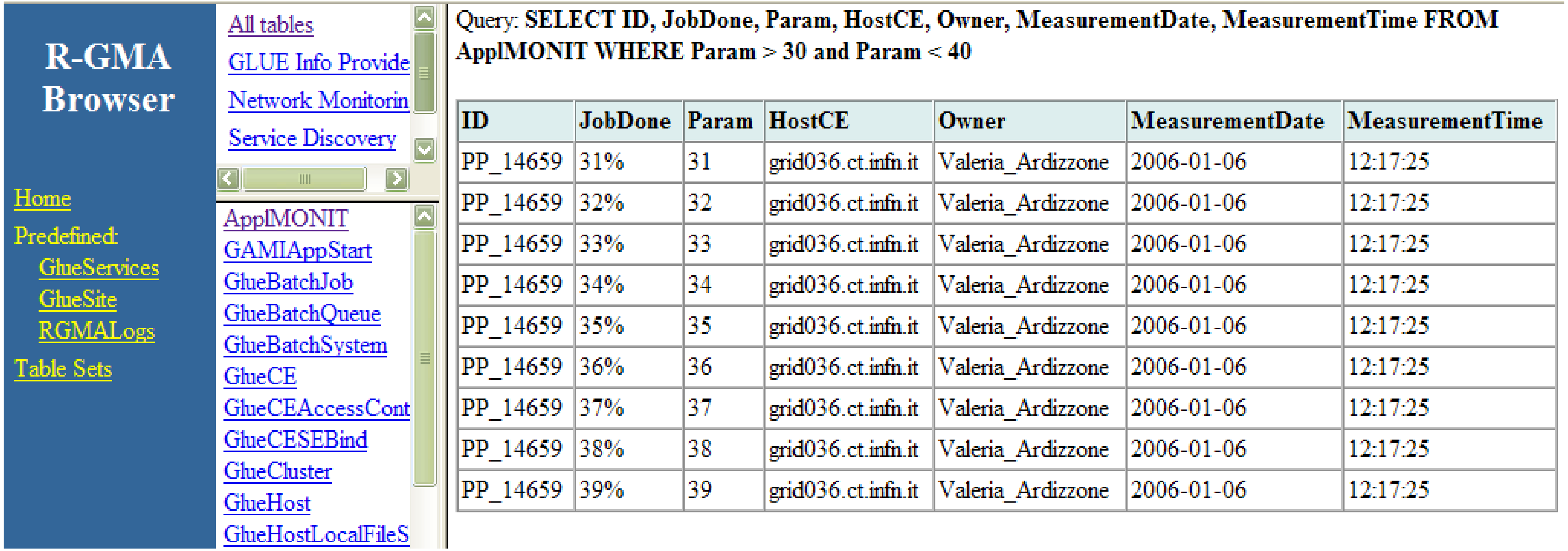
□ https://rgmasrv.ct.infn.it:8443/R-GMA/DBProducerServlet *1452780494*

Latest

Continuous

□ https://rgmasrv.ct.infn.it:8443/R-GMA/DBProducerServlet *1452780494*

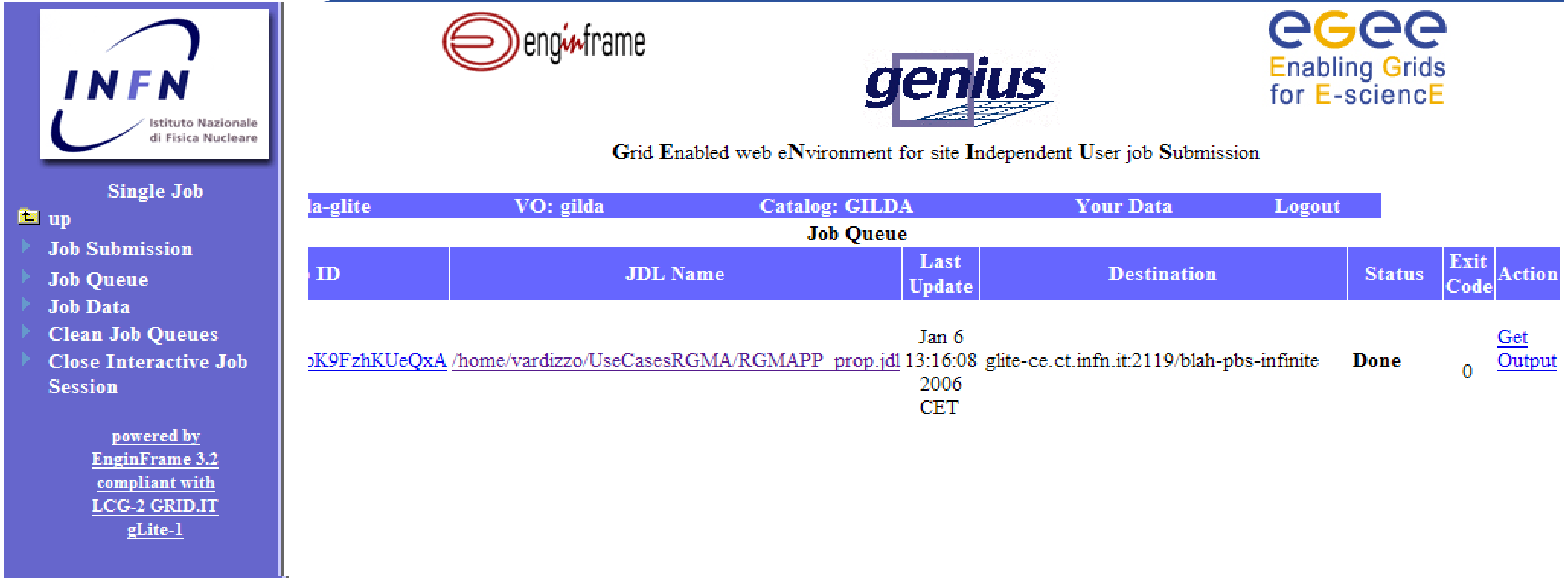
Query Results



Query: **SELECT ID, JobDone, Param, HostCE, Owner, MeasurementDate, MeasurementTime FROM ApplMONIT WHERE Param > 30 and Param < 40**

ID	JobDone	Param	HostCE	Owner	MeasurementDate	MeasurementTime
PP_14659	31%	31	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	32%	32	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	33%	33	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	34%	34	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	35%	35	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	36%	36	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	37%	37	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	38%	38	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25
PP_14659	39%	39	grid036.ct.infn.it	Valeria_Ardizzone	2006-01-06	12:17:25

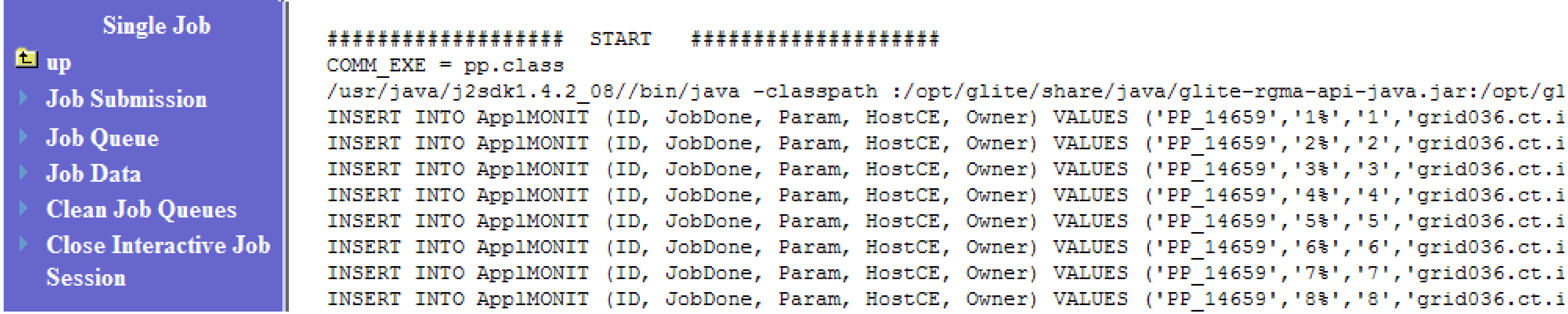
Job Output in GENIUS



Grid Enabled web eNvironment for site Independent User job Submission

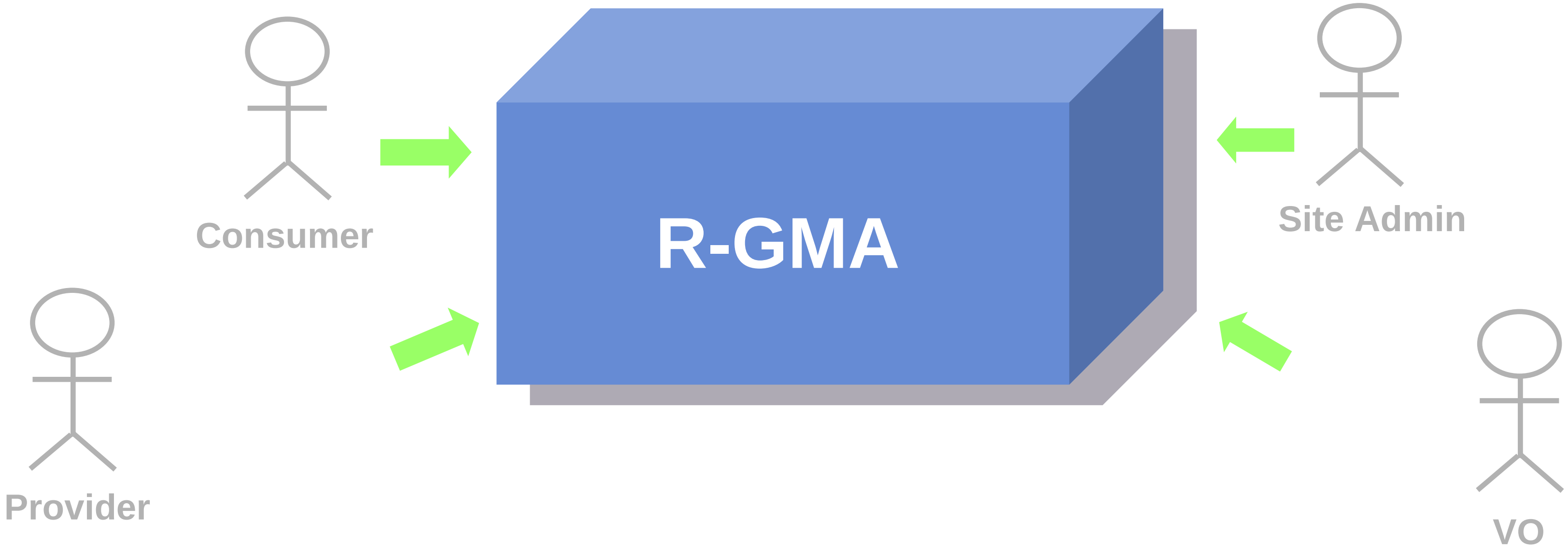
Job Queue

ID	JDL Name	Last Update	Destination	Status	Exit Code	Action
›K9FzhKUeQxA	/home/vardizzo/UseCasesRGMA/RGMAPP_prop.jdl	Jan 6 13:16:08 2006 CET	glite-ce.ct.infn.it:2119/blah-pbs-infinite	Done	0	Get Output



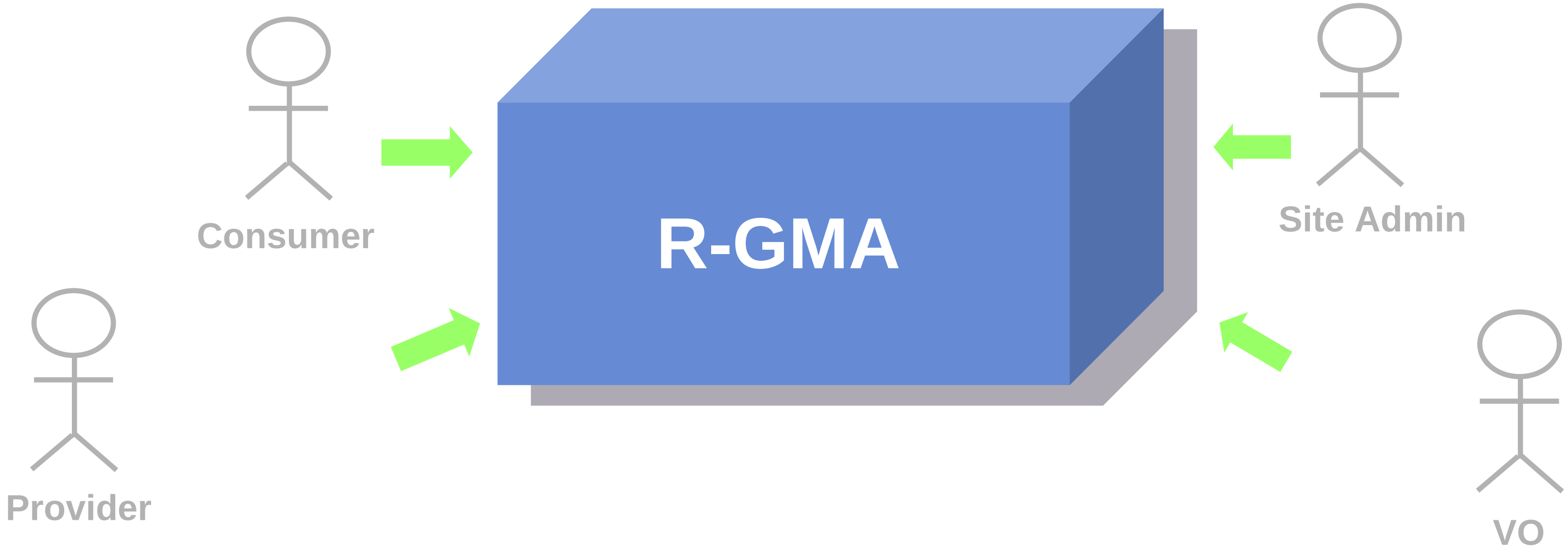
```
###################  START   ####################
COMM_EXE = pp.class
/usr/java/j2sdk1.4.2_08//bin/java -classpath :/opt/glite/share/java/glite-rgma-api-java.jar:/opt/gl
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','1%','1','grid036.ct.i
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','2%','2','grid036.ct.i
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','3%','3','grid036.ct.i
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','4%','4','grid036.ct.i
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','5%','5','grid036.ct.i
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','6%','6','grid036.ct.i
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','7%','7','grid036.ct.i
INSERT INTO ApplMONIT (ID, JobDone, Param, HostCE, Owner) VALUES ('PP_14659','8%','8','grid036.ct.i
```

Security: Requirements



- Consumer users: who requests information.
- Producer users: who provides information.
- Site administrators: who runs R-GMA services.
- Virtual Organizations: who “owns” the schema and registry.

Security: Solution



- Mutual Autentication: guaranteeing who is at each end of an exchange of messages.
- Encryption: using an encrypted transport protocol (HTTPS).
- Authorization: implicit or explicit.

More information

- R-GMA overview page.
  - http://www.r-gma.org/
- R-GMA documentation in EGEE
  - http://hepunx.rl.ac.uk/egee/jra1-uk/
- R-GMA in GILDA
  - http://hepunx.rl.ac.uk/egee/jra1-uk/