

# Databases in the Grid

Grid-Data Source Engine and QE  
A New Data Source Oriented CE for GRID

*Claudio Vuerli – INAF-OATs*

*Giuliano Taffoni – INAF-OATs*

*EGEE '07 – Budapest – 3 October 2007*

- **The Grid limit: it is able to execute binary code or shell scripts and to store files.**
- **We didn't find any suitable solution that allowed us to have databases as an integrated resource in Grid.**
- **We started to tackle this issue in 2004 (GRID.IT project).**

**Databases includes metadata and/or data**

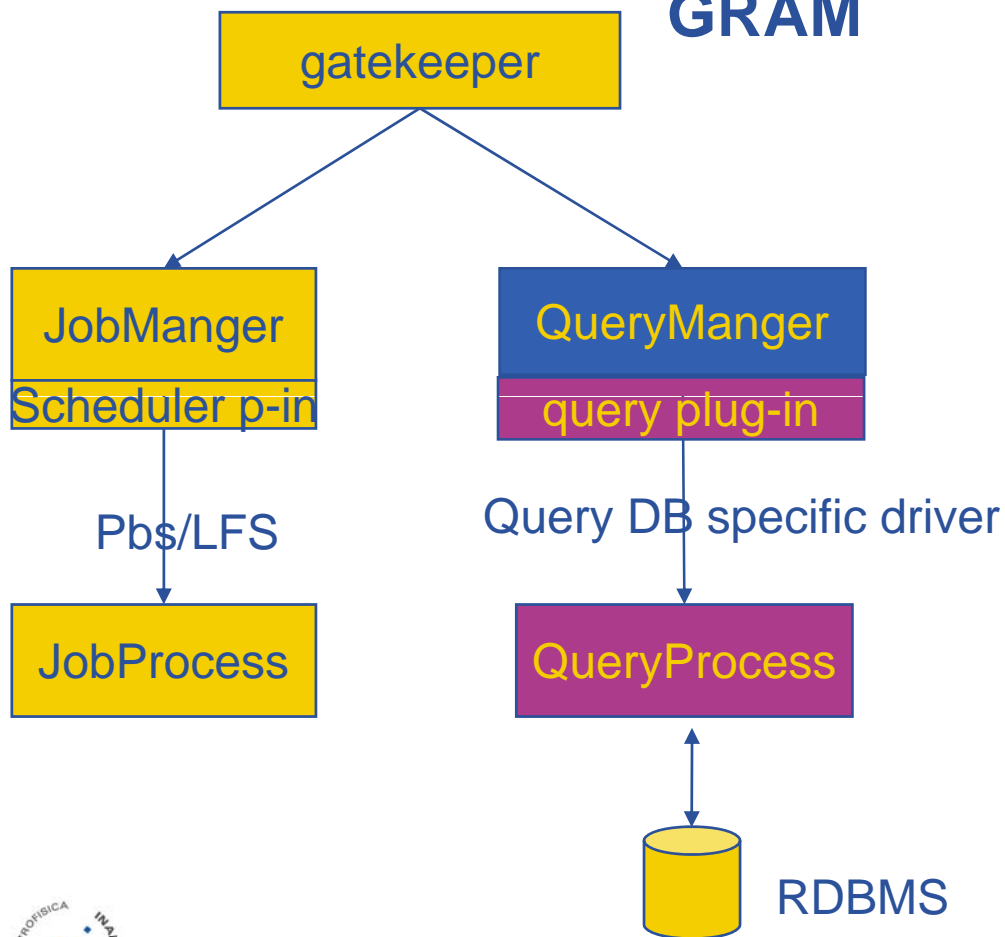
- I have a DB and I want to USE it on GRID.**
- I have a number of DBs and I need using all of them for my application.**
- I want to submit to the Grid complex jobs (workflows) where data retrieval/storage and data analysis are intermixed**

- **Sometimes I need to move the execution to the data and not data to the code.**
- **The integration/interoperability between the Grid technology and for example the Virtual Observatory is of crucial importance for my everyday work**
- **I need to stay fully compliant with the evolution of the Grid M/W**

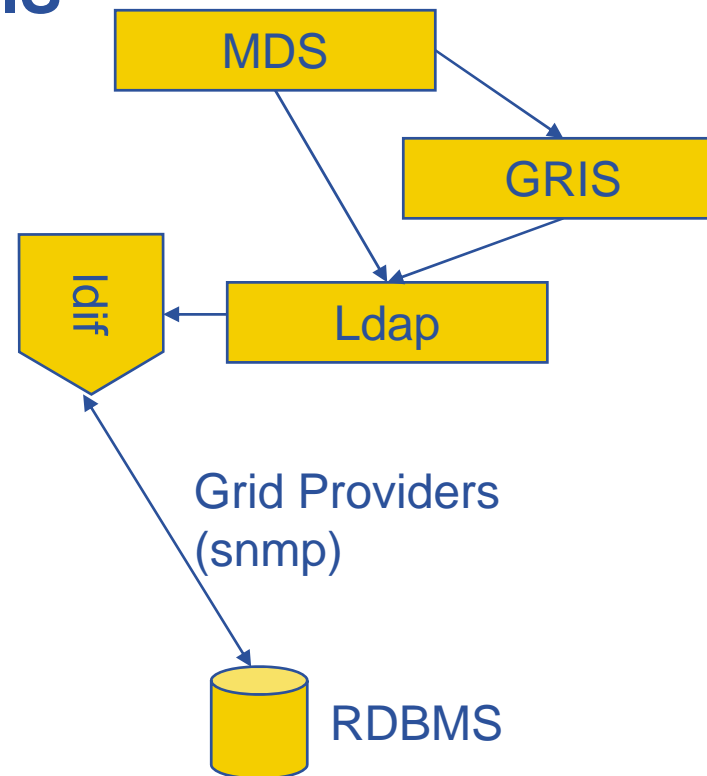
- **With G-DSE+QE, Databases become embedded resources in Grid as any other resource (CPU, Data Storage).**
- **Database machines (the new Grid nodes where the new DB resources reside) can be used to both access/retrieve data and make computations on them.**

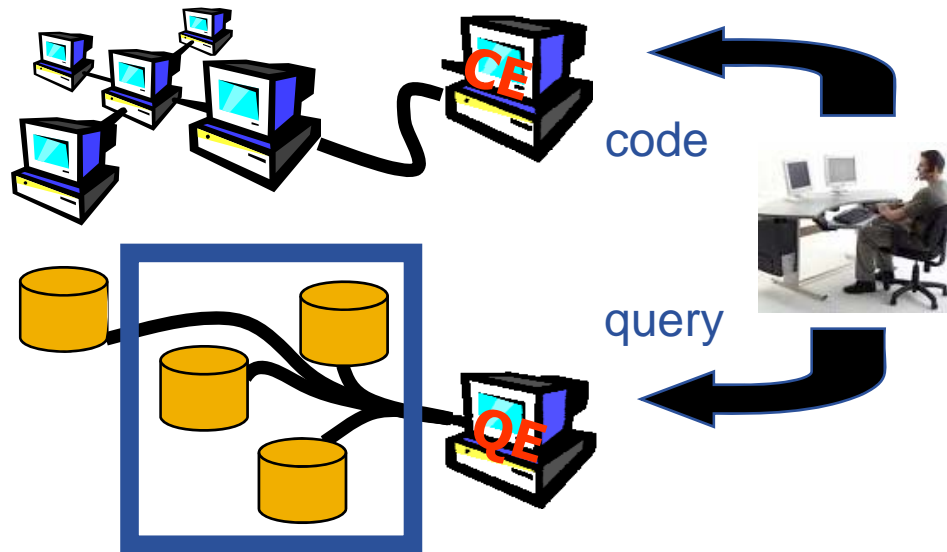
- **The Grid Resource Framework Layer, Information System and Data Model** is extended so that a software virtual machine as a Data Source Engine becomes a valid instance for a Grid computing model.
- A **new Grid component (G- DSE)** that enables the access to a Data Source Engine and Data Source, totally integrated with the Grid Monitoring and Discovery System and Resource Broker is defined
- A new **Grid Element, the Query Element**, can be built on top of the G-DSE component.

## GRAM



## GIS





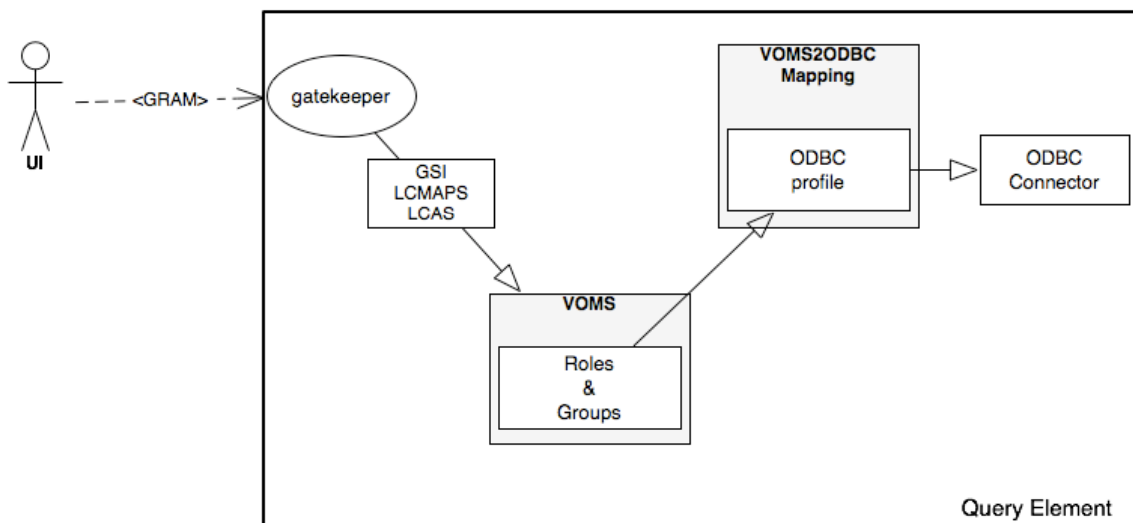
- Runs on any linux/unix flavor: **GT $\geq$ 2.4.3**
- Backends: any DB vendor + flat files

## QE

- Two protocols: GRAM or WS
- API: C, C++, python, Java, perl
- If it works with Globus it works with G-DSE



- **Access control using GSI and VOMS**
  - The certificate + roles identify the user permissions on DB
    - **Super user: create, modify, admin, grant and revoke users.... ANYTHING!!!**
    - **Standard user: select+insert**
    - **Simple user: select**



VOMS roles and groups mapping with db user:

Attribute: /vo/dbuser/  
 ROLE=astrouser/  
 CAPABILITY=select

- G-DSE supports Data Source (DS) and DSE indexing, monitoring, management and recovery through a rich set of Meta-Data bound to standard GIS.
- DS have their core engine into G-DSE, that provides a framework for activity and task management.
- A RSL/JDL Transaction/Query permits a number of tasks to be specified, together with their parameters, inputs, outputs and control flow.
- The response to a request is generated by the G-DSE within a JobQueryManager *Session*. The G-DSE analyses incoming Task and conducts authentication and authorisation
- The standard Grid WorkLoad Manager constructs an optimised execution graph.
- GIS will monitor a DS's and DSE's status digest produced by its internal monitor.
- The G-DSE has been designed to support dynamic configuration, sessions, transactions, recovery and concurrency.

*End of Presentation*  
**Thank you for your attention**