AMGA metadata catalog with use cases

Tony Calanducci
INFN Catania
gLite tutorial at the EGEE User Forum
Geneva, 27-28 February 2006

Contents

- Background and Motivation for AMGA
- Interface, Architecture and Implementation
- Metadata Replication on AMGA
- Deployment Examples
- GILDA Use cases

Metadata on the GRID

- **Metadata is data about data**
- **On the Grid: information about files**
  - Describe files
  - Locate files based on their contents
- **But also simplified DB access on the Grid**
  - Many Grid applications need structured data
  - Many applications require only simple schemas
    - Can be modelled as metadata
  - Main advantage: better integration with the Grid environment
    - Metadata Service is a Grid component
    - **Grid security**
    - Hide DB heterogeneity

ARDA/gLite Metadata Interface

- **2004 - ARDA evaluated existing Metadata Services from HEP experiments**
  - AMI (ATLAS), RefDB (CMS), Alien Metadata Catalogue (ALICE)
  - Similar goals, similar concepts
  - Each designed for a particular application domain
    - Reuse outside intended domain difficult
  - Several technical limitations: large answers, scalability, speed, lack of flexibility
- **ARDA proposed an interface for Metadata access on the GRID**
  - Based on requirements of LHC experiments
  - But generic - not bound to a particular application domain
  - Designed jointly with the gLite/EGEE team
  - Incorporates feedback from GridPP
- **Adopted as the official EGEE Metadata Interface**
  - Endorsed by PTF (Project Technical Forum of EGEE)

AMGA Implementation

- **ARDA developed an implementation of PTF interface**
  - AMGA – ARDA Metadata Grid Application
- **Began as prototype to evaluate the Metadata Interface**
  - Evaluated by community since the beginning:
    - LHCb and Ganga were early testers (more on this later)
  - Matured quickly thanks to users feedback
- **Now part of gLite middleware**
  - Official Metadata Service for EGEE
  - First release with gLite 1.5
  - Also available as standalone component
- **Expanding user community**
  - HEP, Biomed, UNOSAT…

Metadata Concepts

- **Some Concepts**
  - Metadata - List of attributes associated with entries
  - Attribute – key/value pair with type information
    - Type – The type (int, float, string,…)
    - Name/Key – The name of the attribute
    - Value - Value of an entry's attribute
  - Schema – A set of attributes
  - Collection – A set of entries associated with a schema
  - Think of schemas as tables, attributes as columns, entries as rows

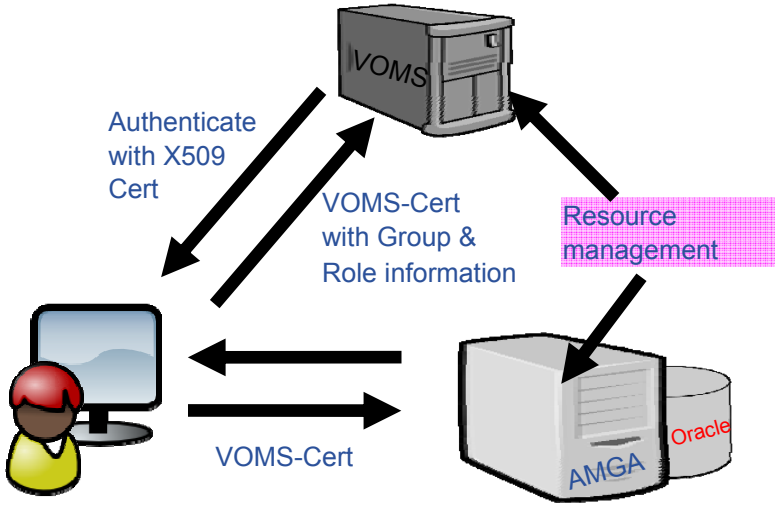
AMGA Features

- **Dynamic Schemas**
  - Schemas can be modified at runtime by client
    - Create, delete schemas
    - Add, remove attributes
- **Metadata organised as an hierarchy**
  - Collections can contain sub-collections
  - Analogy to file system:
    - Collection ⇔ Directory; Entry ⇔ File
- **Flexible Queries**
  - SQL-like query language
  - Joins between schemas
  - Example

```
selectattr /DLibrary:FileName /DLAudio:Author /DLAudio:Album
          '/DLibrary:FILE=/DLAudio:FILE and like(/DLibrary:FileName, "%.mp3")'
```

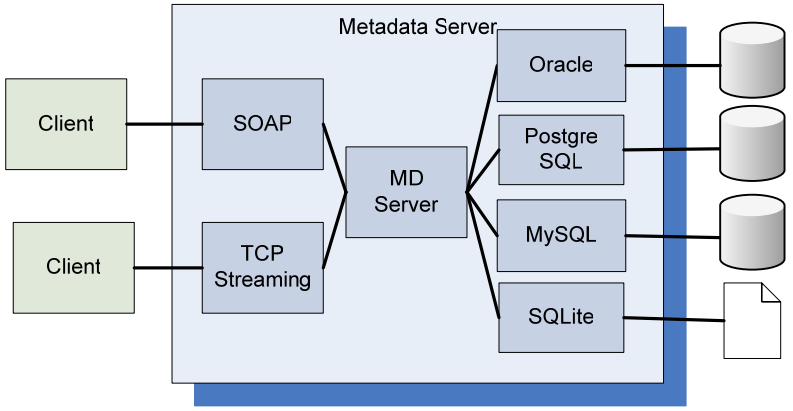
Security

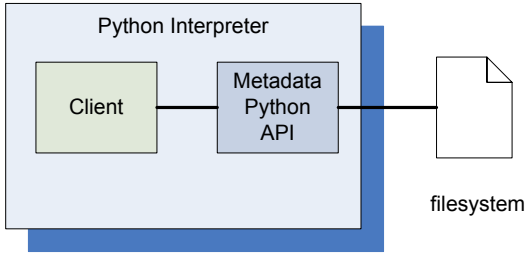
- **Unix style permissions**
- **ACLs – Per-collection or per-entry.**
- **Secure connections – SSL**
- **Client Authentication based on**
  - Username/password
  - General X509 certificates
  - Grid-proxy certificates
- **Access control via a Virtual Organization Management System (VOMS):**



AMGA Implementation

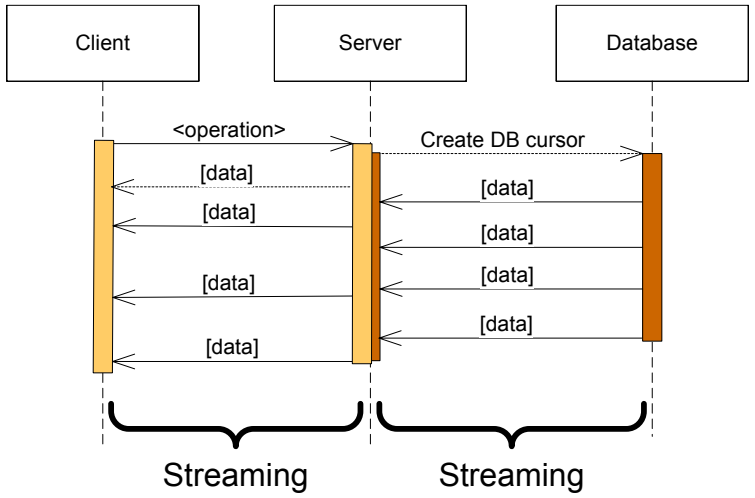
- **C++ multiprocess server**
  - Runs on any Linux flavour
- **Backends**
  - Oracle, MySQL, PostgreSQL, SQLite
- **Two frontends**
  - TCP Streaming
    - High performance
    - Client API for C++, Java, Python, Perl, Ruby
  - SOAP
    - Interoperability
- **Also implemented as standalone Python library**
  - Data stored on filesystem





Architecture
TCP-Streaming frontend

- **Designed for scalability**
  - Asynchronous operation
    - Reading from DB and sending data to client
  - Response sent to client in chunks
    - No limit on the maximum response size
- **Example: TCP Streaming**
  - Text based protocol (like SMTP, POP3,…)
  - Response streamed to client



Client: `listattr entry`

Server:
```
0
entry
value1
value2
…
<EOT>
```

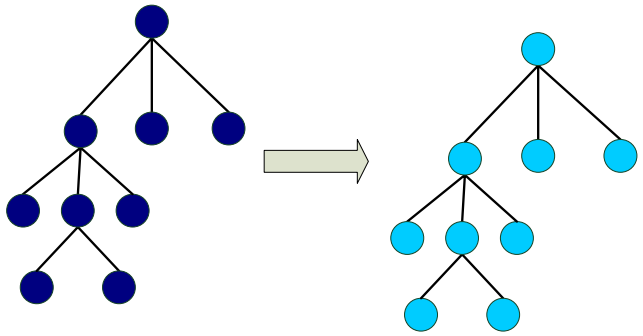
Metadata Replication

- **Currently working on replication/federation mechanisms for AMGA**
- **Motivation**
  - Scalability – Support hundreds/thousands of concurrent users
  - Geographical distribution – Hide network latency
  - Reliability – No single point of failure
  - DB Independent replication – Heterogeneous DB systems
  - Disconnected computing – Off-line access (laptops)
- **Architecture**
  - Asynchronous replication
  - Master-slave – Writes only allowed on the master
  - Replication at the application level
    - Replicate Metadata commands, not SQL → DB independence
  - Partial replication – supports replication of only sub-trees of the metadata hierarchy

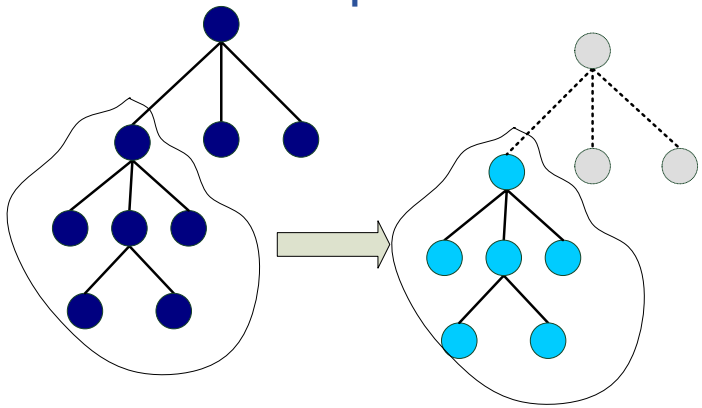
Metadata Replication

Some use cases

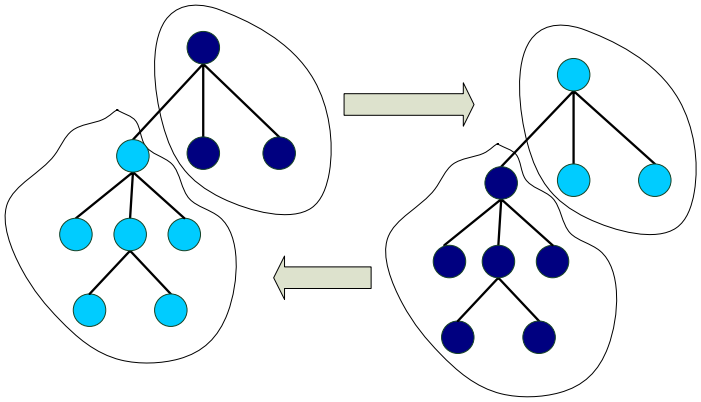
Full replication



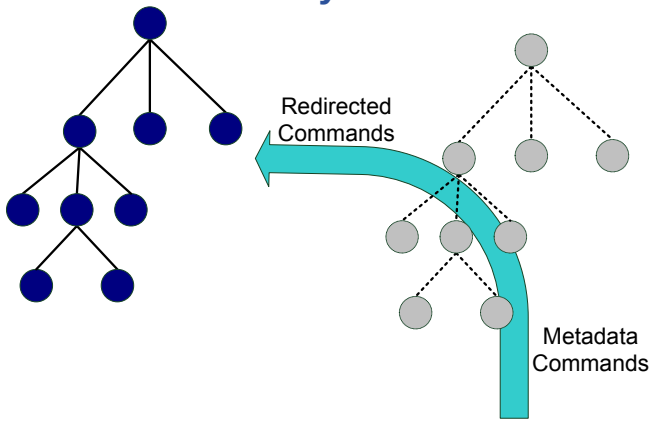
Partial replication



Federation



Proxy



Metadata Replication

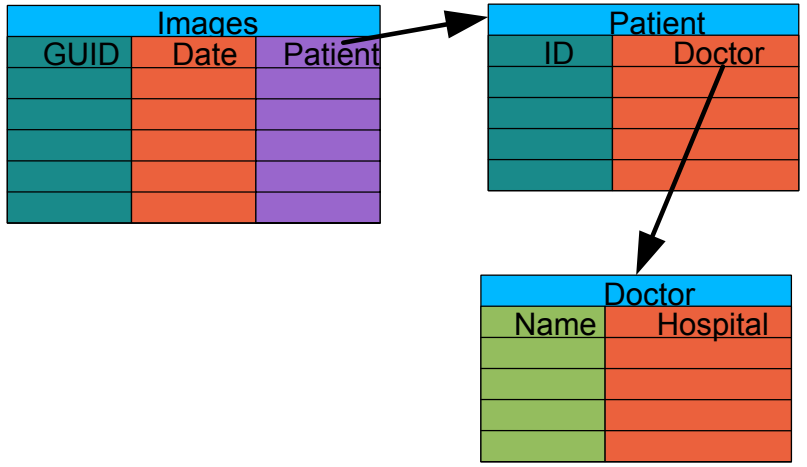
- **Current Status**
  - Implementation under way
  - Integrated on AMGA, no external software needed
  - Early prototype. Basic functionality working in a single slave configuration
    - Initial slave synchronization
    - Update propagation
- **Future Plans**
  - More development and testing needed
  - Working prototype expected soon.

Early adopters of AMGA

- **LHCb-bookkeeping**
  - Migrated bookkeeping metadata to ARDA prototype
    - 20M entries, 15 GB
    - Large amount of static metadata
  - Feedback valuable in improving interface and fixing bugs
  - AMGA showing good scalability
- **Ganga**
  - Job management system
    - Developed jointly by Atlas and LHCb
  - Uses AMGA for storing information about job status
    - Small amount of highly dynamic metadata

Biomed

- **Medical Data Manager – MDM**
  - Store and access medical images and associated metadata on the Grid
  - Built on top of gLite 1.5 data management system
  - Demonstrated at last EGEE conference (October 05, Pisa)
- **Strong security requirements**
  - Patient data is sensitive
  - Data must be encrypted
  - Metadata access must be restricted to authorized users
- **AMGA used as metadata server**
  - Demonstrates authentication and encrypted access
  - Used as a simplified DB



- **More details at**
  - https://uimon.cern.ch/twiki/bin/view/EGEE/DMEncryptedStorage

Conclusion

- **AMGA – Metadata Service of gLite**
  - Part of gLite 1.5
  - Useful for simplified DB access
  - Integrated on the Grid environment (Security)
- **Replication/Federation under development**
- **Tests show good performance/scalability**
- **Already deployed by several Grid Applications**
  - LHCb, ATLAS, Biomed, …
  - DLibrary (next presentation)

- **AMGA Web Site**

  `http://project-arda-dev.web.cern.ch/project-arda-dev/metadata/`

GILDA Use cases

- DLibrary
- AMGA for GIS datatypes metadata
- gMOD

DLibrary goals

- **DLibrary goal is to create a Multimedia Contents Management System on the Grid**
- **Examples of Multimedia Contents handled by DLibrary:**
  - Images
  - Videos
  - Music
  - Office Documents (PPT, Word, Excel)
  - E-Mails
  - PDF
  - ….
- **DLibrary is a repository of well structured and uniform metadata for files present on Storage Elements**
- **Users and jobs can easily submit new contents and look later for them in an easy, fast and secure way through the Grid**
- **We choose AMGA to archive metadata for files stored in SEs and to answer users' and jobs' queries**

DLibrary use cases

- **Example 1:**
  - An user needs to find all the PowerPoint Presentation about Data Management System in 2006 run by Uncle Sam (fantasy name)
- **Example 2 (Google for Storage):**
  - a job behaves as a “storage crawler”: it scans all the files stored in Storage Elements and publishes their related specific information into DLibrary for later searches through their attributes.
- **Example 3:**
  - A user wants to look for all the movies in which Jennifer Aniston performed, produced in 2004; or find all the songs of Led Zeppelin that last for more than 6 minutes.
- **Example 4:**
  - A doctor wants to retrieve all the articles and presentations about lung cancer and download some lung X-ray images to be printed in his article for a scientific magazine

DLibrary prototype implementation

- **Definitions:**
  - An entry represents a document archived in a SE.
  - Attributes are the entries' related information (ex. FileName, Type, SubmissionDate, Keywords, Comment, …)
  - Collections are a set of entries sharing the same attributes (called also schema). In a file system analogy, they are directories.
- **DLibrary is built using the following collections:**
  - /DLibrary contains generic information per each entry
  - /DLTypes contains associations between document types and "special features" collection path
  - /DLAudio, /DLImage, /DLVideo, /DLPPT, /EGEEPPT, /DLDoc are some examples of "special features" collections
  - /DLKeys is used to store Decryption Keys

Entry Example

Collection	/DLibrary

Entry Names	Attributes			
	FileName	PathName	Type	Submitter
4ffaffc8-26e7-4826-b460-3d5bf08081a4	DedicatoAte.mp3	/grid/gilda/calanducci	Audio	Tony Calanducci
00454dca-a269-4b93-8a45-c4012af05600	ardizzonelarocca_is_231005.ppt.gpg	/grid/gilda/calanducci/ EGEE	EGEEDOC	Tony Calanducci

/DLibrary (continuum)

Attributes

SubmissionDate	DecryptKeyDir	Description	Keywords	CreationDate
2006-01-05 00:00:00		Canzone delle vibrazioni che ha ricevuto un enorme successo tra i teenagers nel 2003	Vibrazioni	2004-02-05 00:00:00
2005-01-05 16:44:22	/DLKeys/gildateam	gLite Information System	R-GMA, RGMA, BDII, IS	2005-10-05 23:40

Entry Example (II)

Collection	/DLTypes
Entry names	Attributes
	Path *(refers to a collection)*
Audio	/DLAudio
Image	/DLImage
Video	/DLVideo
Documents	/DLDOC
PowerPoint	/DLPPT
EGEEDOC	/EGEEPPT

Collection	/DLKeys/gildateam
Entry names	Attributes
	Passphrase
00454dca-a269-4b93-8a45-c4012af05600	ardizzo

Collection	/EGEEPPT							
Entry names	Attributes							
	Title	Runtime	Author	Type	Date	Event	Speaker	Topic
00454dca-a269-4b93-8a45-c4012af05600	Information Systems	00:30:00	Valeria Ardizzione, Giuseppe La Rocca	Theorical	2005-10-23	4th EGEE Conference	Giuseppe La Rocca, Valeria Ardizzone	R-GMA, BDII

Collection	/DLAudio					
Entry names	Attributes					
	SongTitle	Duration	Album	Genre	Singer	Format
4ffaffc8-26e7-4826-b460-3d5bf08081a4	Dedicato A Te	00:03:27	Dedicato A Te	Pop	Le Vibrazioni	MP3

DLibrary security features

- **3 kinds of users:**
  - a LibraryManager can create new DLibrary Document Types
  - a LibrarySubmitter can add new entries
  - authorized VO members can browse and search the library.

  LibraryManagers can also grant VO users submission rights
- **DLibrary provides an initial level of cryptography for data in SEs: a symmetric passphrase can be stored besides regular attributes per entry in DLibrary collections.**
  - LibrarySubmitters can authorize VO users to access decryption keys
- **All this was easily implemented thanks to the powerful and fine-grained user authorization and ACL features of AMGA**

DLibrary Authorization

```
Query> acl_show /DLibrary/
>> LibraryManager rwx
>> LibraryManager:libsubmitters rwx
>> gilda:users rx
Query> acl_show /DLAudio
>> LibraryManager rwx
>> LibraryManager:libsubmitters rwx
>> gilda:users rx
```

```
Query> acl_show DLTypes
>> LibraryManager rwx
>> LibraryManager:libsubmitters rx
>> gilda:users rx
```

```
Query> whoami
>> LibrarySubmitter
Query> acl_show /DLKeys/gildateam
>> LibrarySubmitter rwx
>> LibrarySubmitter:gildateam rx
Query> grp_show gildateam
>> tony
>> valeria
>> giuseppe
>> emidio
Query> user_listcred tony
>>
>> 'C = IT, O = GILDA, OU = Personal
Certificate, L = INFN Catania, CN =
Tony Calanducci, emailAddress =
tony.calanducci@ct.infn.it'
```

Example: EGEE PPT submission

Log into your UI and locate the file you want to upload to a SE.

The use GPG (as an example) to encrypt it using a symmetric passphrase:

```
$ gpg -c VOMS_Server_Installation.ppt
Enter Passphrase: *******
$ ls VOMS_Server_Installation.ppt*
VOMS_Server_Installation.ppt    VOMS_Server_Installation.ppt.gpg
```

Upload the encrypted file to a SE

```
$ lcg-cr -v --vo gilda -d grid009.ct.infn.it
         -l lfn:/grid/gilda/calanducci/EGEE/VOMS_server_Installation.ppt.gpg
         file://$PWD/VOMS_Server_Installation.ppt.gpg
Using grid catalog type: lfc
Source URL: file:/home/tcaland/DLibrary Docs/EGEEPPT/VOMS_Server_Installation.ppt.gpg
File size: 657042
VO name: gilda
Destination specified: grid009.ct.infn.it
Destination URL for copy: gsiftp://grid009.ct.infn.it/flatfiles/SE00/gilda/generated/2006-
01-07/file24151089-8bf2-499a-9ca0-1b0120f1ed69
Alias registered in Catalog:
lfn:/grid/gilda/calanducci/EGEE/VOMS_server_Installation.ppt.gpg
            0 bytes      0.00 KB/sec avg      0.00 KB/sec inst
Transfer took 4130 ms
Destination URL registered in Catalog:
sfn://grid009.ct.infn.it/flatfiles/SE00/gilda/generated/2006-01-07/file24151089-8bf2-499a-
9ca0-1b0120f1ed69
guid:1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
```

Example: EGEE PPT submission (II)

Initialize your VOMS proxy asking for LibrarySubmitter Role

```
$ voms-proxy-init --voms gilda:/gilda/Role=LibrarySubmitter
$ voms-proxy-info -fqan | grep LibrarySubmitter
/gilda/Role=LibrarySubmitter/Capability=NULL
```

Edit your .mdclient.config setting Login=gilda:/gilda/Role^=LibrarySubmitter
Log into AMGA using mdclient command line tool

```
$ mdclient
Connecting to wn1-test.ct.infn.it:8899...
ARDA Metadata Server 1.0.0
Query> whoami
>> LibrarySubmitter
```

Example: EGEE PPT submission (III)

Create a new entry inside /DLibrary collection and one with the related "special features" into /EGEEPPT using the GUID obtained by lcg-cr as entry name

```
Query> addentry /DLibrary/1f6e9ac6-5c86-4599-b03b-560e0e7ea38a FileName
VOMS_server_Installation.ppt.gpg PathName /grid/gilda/calanducci/EGEE Type
EGEEDOC Submitter 'Tony Calanducci' SubmissionDate '2006-01-07 18:44'
DecryptKeyDir '/DLKeys/gildateam' Description 'VOMS Server installation tutorial
done in Venezuela' Keywords 'VOMS Server' CreationDate '2005-10-08 18:28'
Query> addentry /EGEEPPT/1f6e9ac6-5c86-4599-b03b-560e0e7ea38a Title 'Installing
a gLite VOMS Server' Runtime '01:30:00' Author 'ziggy, Giorgio' Type 'Admin,
Installation' Date '2005-11-21 11:00' Event 'First Latin American Workshop for
Grid Administrators' Speaker 'Emidio Giorgio' Topic 'VOMS'
```

Add the passphrase to the collection with ACLs listing people you want to give access to

```
Query> addentry /DLKeys/gildateam/1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
Passphrase 'EGEE2006'
Query> acl_show /DLKeys/gildateam
>> LibrarySubmitter rwx
>> LibrarySubmitter:gildateam rx
Query> grp_show gildateam
>> tony
>> valeria
>> giuseppe
>> emidio
```

Example: DLibrary queries

Initialize your VOMS proxy asking to be member of the gilda VO

Edit your .mdclient.config setting Login=NULL (user will be retrieved from your proxy extensions)

Log into AMGA

```
$ voms-proxy-init --voms gilda
$ voms-proxy-info -fqan
/gilda/Role=NULL/Capability=NULL
$ grep Login .mdclient.config
Login = NULL
```

Suppose we want to look for all contents about VOMS

```
Query> whoami
>> gilda
Query> selectattr /DLibrary:FILE /DLibrary:FileName /DLibrary:Type
'like(/DLibrary:Keywords, "%VOMS%")'
>> 1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
>> VOMS_server_Installation.ppt.gpg
>> EGEEDOC
```

Now let's find out in which collection EGEEDOC attributes are stored

```
Query> getattr /DLTypes/EGEEDOC Path
>> EGEEDOC
>> /EGEEPPT
```

Example: DLibrary queries (II)

Now we can make a JOIN between the 2 tables to extract all the information we like

```
Query> selectattr /DLibrary:FILE /DLibrary:FileName /DLibrary:Description
/EGEEPPT:Author /EGEEPPT:Title /EGEEPPT:Event '/DLibrary:FILE=/EGEEPPT:FILE and
like(/DLibrary:Keywords, "%VOMS%")'
>> 1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
>> VOMS_server_Installation.ppt.gpg
>> VOMS Server installation tutorial done in Venezuela
>> ziggy, Giorgio
>> Installing a gLite VOMS Server
>> First Latin American Workshop for Grid Administrators
```

Let's see where the passphrase to decrypt the file is stored

```
Query> selectattr /DLibrary:FILE DecryptKeyDir 'FILE="1f6e9ac6-5c86-4599-b03b-
560e0e7ea38a"'
>> 1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
>> /DLKeys/gildateam
```

But …

```
Query> getattr /DLKeys/gildateam/1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
Passphrase
Error 4: Permission denied
```

Because gilda is not a member of the gildateam group

Example: DLibrary queries (III)

Set Login = tony into .mdclient.config. I will be authenticated correctly because my DN is mapped onto the AMGA user tony

```
Query> whoami
>> tony
Query> grp_member
>> gilda:users
>> LibrarySubmitter:gildateam
Query>  getattr /DLKeys/gildateam/1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
Passphrase
>> 1f6e9ac6-5c86-4599-b03b-560e0e7ea38a
>> EGEE2006
```

Yet another example of a query to look for Audio contents:

```
Query> selectattr /DLibrary:FileName SubmissionDate Submitter
/DLAudio:SongTitle Singer Duration Genre '/DLibrary:FILE=/DLAudio:FILE'
>> DedicatoAte.mp3
>> 2006-01-05 00:00:00
>> Tony Calanducci
>> Dedicato A Te
>> Le Vibrazioni
>> 00:03:27
>> Pop
```

Accessing AMGA

- **TCP Streaming Front-end**
  - mdcli & mdclient and C++ API (md_cli.h, MD_Client.h)
  - Java Client API and command line mdjavaclient.sh & mdjavacli.sh (also under Windows !!)
  - Python Client API
- **SOAP Frontend (WSDL)**
  - C++ gSOAP
  - AXIS (Java)
  - ZSI (Python)

AMGA for GIS Datatype Metadata

- **AMGA Datatypes**

	PostgreSQL	MySQL	Oracle	SQLite	Python
int	integer	int	number(38)	int	int
float	double precision	double precision	float	float	float
varchar(n)	character varying(n)	character varying(n)	varchar2(n)	varchar(n)	string
timestamp	timestamp w/o TZ	datetime	timestamp(6)	unsupported	time (unsupp.)
text	text	text	long	text	string
numeric(p,s)	numeric(p,s)	numeric(p,s)	numeric(p,s)	numeric(p,s)	float

  - Using the above datatypes you are sure that your metadata can be easily moved to all supported back-ends
  - If you do not care about DB portability, you can use, in principle, as entry attribute type ALL the datatypes supported by the back-end, even the more esoteric ones (PostgreSQL Network Address type or Geometric ones)
- **We played a little bit with GIS Datatype offered by MySQL 5**

Example with ESR data

We created a /ESR/opera_nno collection asking AMGA to use the MyISAM table engine

```
Query> listattr /ESR/opera_nno
>> Dataset
>> varchar(30)
>> File_Name
>> varchar(50)
>> Footprint
>> multipolygon
>> Lat
>> numeric(8,2)
>> Level
>> varchar(5)
```

```
>> Lon
>> numeric(8,2)
>> Orbit
>> int(5)
>> Proc_centre
>> varchar(50)
>> Proc_date
>> timestamp
>> Start_Date
>> timestamp
>> Stop_Date
>> timestamp
...
```

We used *insert* command that evaluates all inserted values:

```
insert sameEntryName Dataset "GOME" Level 2 Version "v1.1" Orbit 25421
File_Name "/grid/esr/gome/utv/2000/03/00301000.utv" Start_Date '"2000-02-29
00:01:00.0"' Stop_Date '"2000-02-29 00:58:00.0"' Footprint
'MPolyFromText("MULTIPOLYGON(((82.96 -59.12,75.95 -89.07,75.95 -89.07,76.46 -
94.77,76.84 -100.85,77.07 -107.21,77.13 -115.34,77.00 -121.80,76.72 -
128.08,76.30 -134.03,75.74 -139.59,75.07 -144.70,74.30 -149.36,80.26 -
179.07,80.26 -179.07,81.52 -174.78,82.71 -169.12,83.81 -161.42,84.76 -
150.74,85.47 -136.17,85.80 -117.93,85.57 -94.31,84.94 -78.84,84.03 -
67.39,82.96 -59.12)))")' Proc_centre "EGEE" Proc_date '"2005-10-14
13:20:00.0"' File_input "00301000.lv1" Proc_description '"Algorithm: utv"'
```

Sample queries

Let's check if the entry was properly inserted (we need to use AsText() to decode a MultiPolygon):

```
Query> selectattr /ESR/opera_nno:File_Name AsText(/ESR/opera_nno:Footprint) ''
>> /grid/esr/gome/utv/2000/03/00301000.utv
>> MULTIPOLYGON(((82.96 -59.12,75.95 -89.07,75.95 -89.07,76.46 -94.77,76.84 -
100.85,77.07 -107.21,77.13 -115.34,77 -121.8,76.72 -128.08,76.3 -134.03,75.74 -
139.59,75.07 -144.7,74.3 -149.36,80.26 -179.07,80.26 -179.07,81.52 -174.78,82.71
-169.12,83.81 -161.42,84.76 -150.74,85.47 -136.17,85.8 -117.93,85.57 -
94.31,84.94 -78.84,84.03 -67.39,82.96 -59.12)))
```

We want to look for a Polygon that cointains a given point:

```
Query> selectattr /ESR/opera_nno:File_Name /ESR/opera_nno:Start_Date
/ESR/opera_nno:Stop_Date 'Contains(/ESR/opera_nno:Footprint,
GeomFromText("POINT(82.96 -59.12)"))'
>> /grid/esr/gome/utv/2000/03/00301000.utv
>> 2000-02-29 00:01:00
>> 2000-02-29 00:58:00
```

- **As a summary, the following functions work: GeomFromText(), MPolyFromText(), Contains(), AsText()**
- **In principle PostgreSQL+PostGIS would also work but this is not tested.**

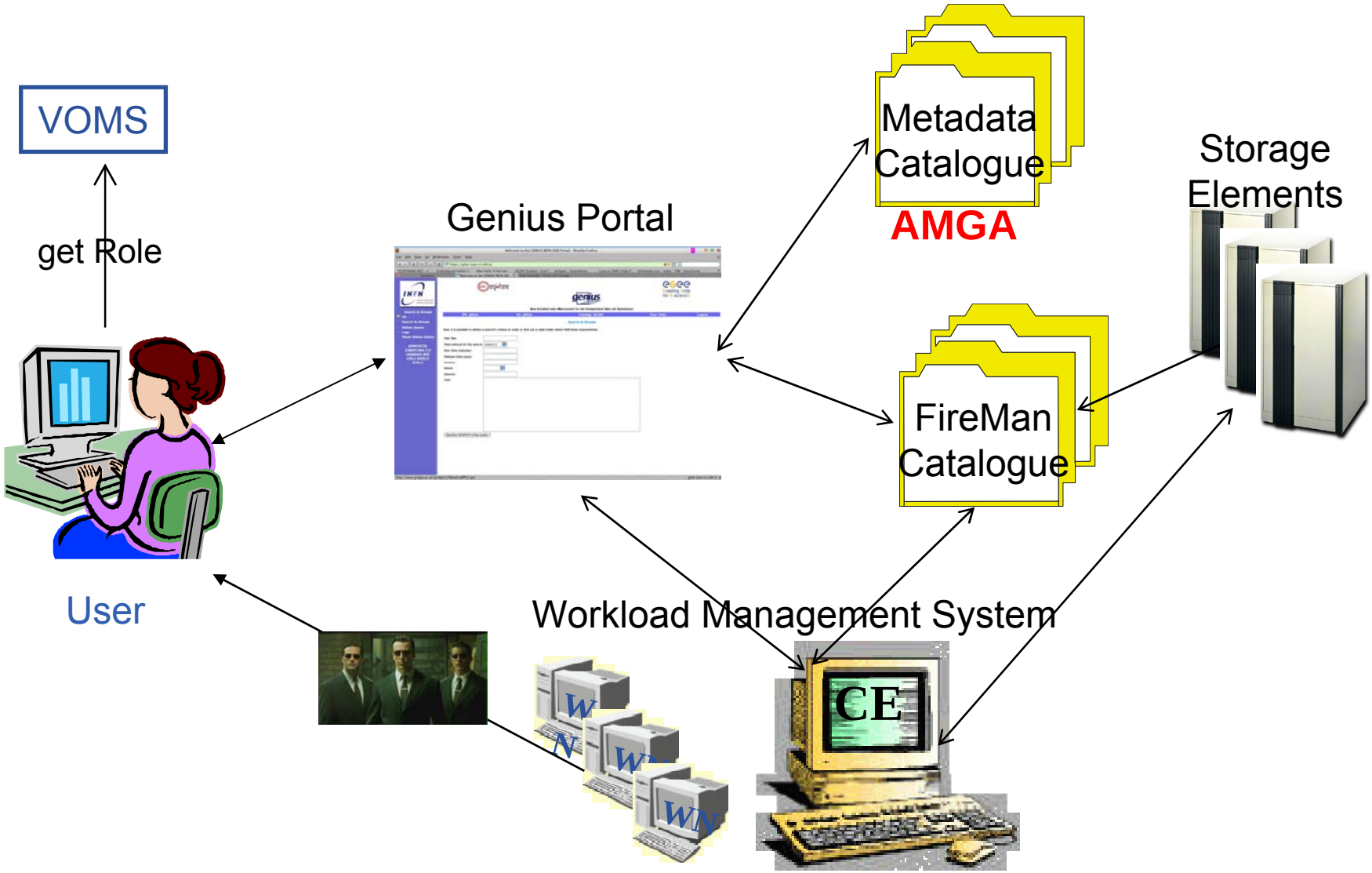
gMOD: grid Movie On Demand

- gMOD provides a Video-On-Demand service
- User chooses among a list of video and the chosen one is streamed in real time to the video client of the user's workstation
- For each movie a lot of details (Title, Runtime, Country, Release Date, Genre, Director, Case, Plot Outline) are stored and users can search a particular movie querying on one or more attributes
- Two kind of users can interact with gMOD: **TrailersManagers** that can administer the db of movies (uploading new ones and attaching metadata to them); **GILDA VO users (guest)** can browse, search and choose a movie to be streamed.

gMOD under the hood

- Built on top of gLite services:
- Storage Elements, sited in different place, physically contain the movie files
- FireMan, the File Catalogue, keeps track in which Storage Element a particular movie is located
- AMGA is the repository of the detailed information for each movie, and makes possible queries on them
- The Virtual Organization Membership Service (VOMS) is used to assign the right role to the different users
- The Workload Management System (WMS) is responsible to retrieve the chosen movie from the right Storage Element and stream it over the network down to the user's desktop or laptop

gMOD interactions



gMOD screenshot

gMOD is accesible through the Genius Portal (https://glite-tutor.ct.infn.it

Any questions?

Thanks for the attention