



Enabling Grids for E-scienceE

The FiReMan Catalog

Emidio Giorgio

INFN

First gLite tutorial on GILDA, Catania, 13-15.06.2005

www.eu-egee.org



- **FiReMan overview**
- **Guiding principles**
- **Features**
- **Installation & Configuration**



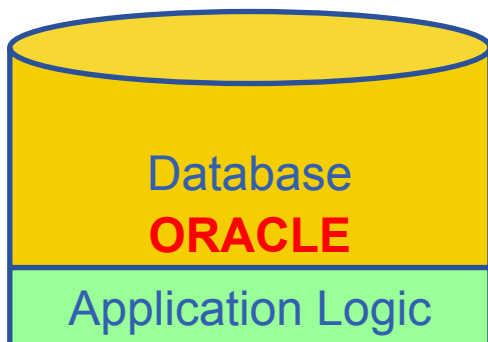
- **Catalogs built based on requirements from HEP experiments and the Biomedical EGEE community**
- **Started design from AliEn File Catalog**
 - Logical namespace management
 - Virtual Filesystem view (DataSets via directory hierarchy)
 - Support Metadata attached to files
 - Bulk Operations
 - Strong security: basic unix permissions and fine-grained ACLs (i.e. not just directory but file-granularity)
 - Support flexible deployment models
 - Single central catalog model
 - Site local catalogs connected to a single central catalog model
 - Site local catalogs without single central catalog model
 - Scalable to many clients and to a large number of entries; address performance issues seen with EDG RLS

- **Oracle / MySql implementation**
- **Hierarchy: Filesystem view**
- **Full Security support**
 - VOMS integration
 - Fine grained ACL control
 - Minimal performance penalties
- **Bulk operations**
 - Necessary to meet performance requirements
- **WS-tuning**
 - Web services are not as bad as they first seem
- **Useful and intuitive interfaces**
 - Logical interface decomposition for well-defined feature-sets
- **Metadata support**

- 2 independent implementations exist

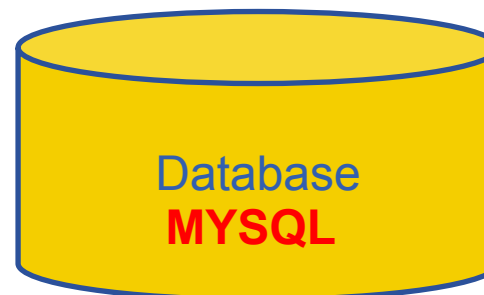
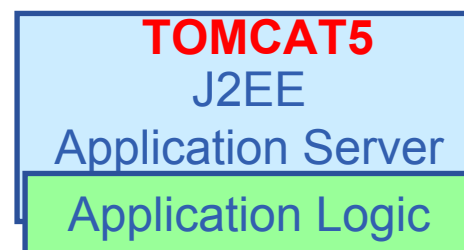
Oracle Implementation

- Catalog Logic lives inside Oracle as Stored Procedures
- Tomcat parses credential only, passes operations through to DB



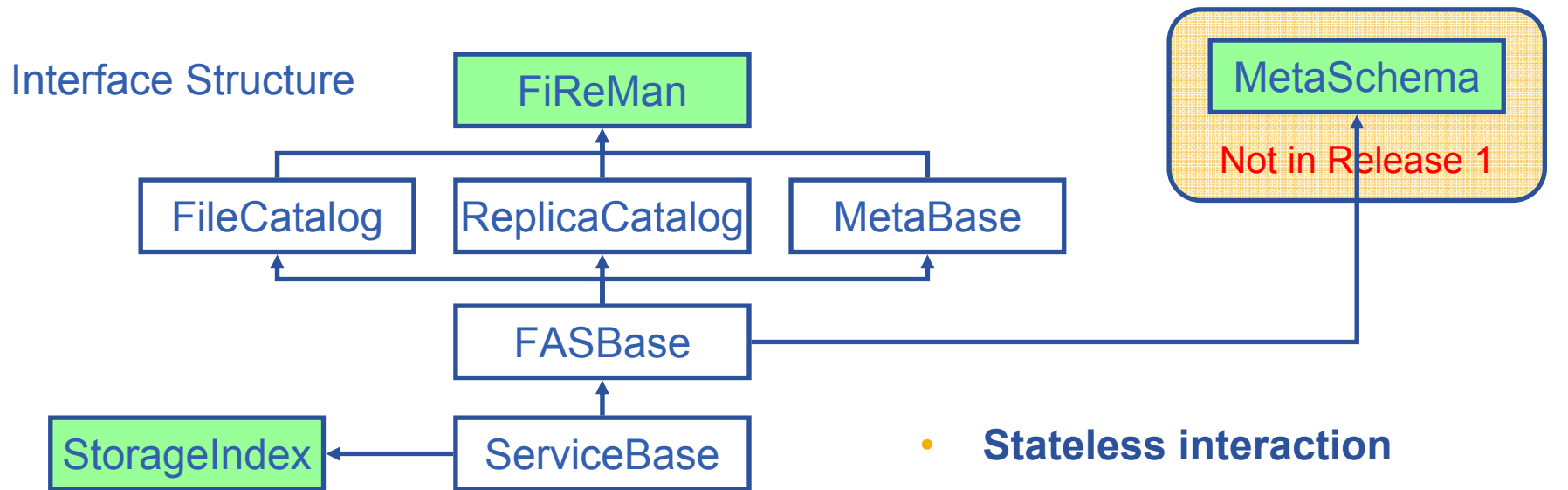
MySQL Implementation

- Simple Table Structure using InnoDB tables
- Credential parsing and all of the logic is in Tomcat



- Logical File Namespace management
- Replica locations
- File-based metadata
- Metadata Management
- Authentication and Authorization information (ACLs)
- Service Metadata
- WMS interaction and global file location

FileCatalog
ReplicaCatalog
MetaBase
MetaSchema
FASBase
ServiceBase
ServiceIndex



- Stateless interaction
- No transactions outside Bulk

- **Web-services interface:** Guarantees client support on many platforms and many languages.
- **Standardization effort ongoing.** It is being managed through the EGEE PTF. Are provided:
 - **Linux Command Line tools**
 - **C/C++ API**
 - **Java API**
 - **Perl modules**
 - **JavaScript (for web clients)**
 - **gLite integrated bash (glitesh) – prototype**
- **Security:** Fine-grained ACL support with minimal performance penalty.
 - DNs own the files
 - VOMS group support
 - Basic Unix security (ugo rwx)
 - Additional ACLs for setPermission, list, remove, setMetadata, getMetadata

- **FiReMan Catalog**
 - Release 1: Single Central deployment model only
 - Release 2: Distributed catalog according to design using Java Messaging Services to propagate updates between catalog instances
- **Storage Index**
 - Already in Release 1
 - Main interaction point with Workload Management
- **Metadata Catalog**
 - Release 1: Base Implemented by FiReMan
 - Also a standalone service, single central instance
 - Release 2: distribution using a messaging infrastructure

- **Start from a fresh install of SLC 3.0.x (x >= 4)**
- **Download the script installer from glite web site**
http://glite.web.cern.ch/glite/packages/R1.1/R20050430/installers/glite-data-single-catalog_installer.sh (MySQL)
- **Launch the script installer**

```
sh glite-data-single-catalog_installer.sh
```
- **That's All ! The script installer will download and install all the needed packages, except for Java that has to be downloaded and installed separately**

- After the installation, under `/opt/glite/etc/config/templates` find the xml templates for configuration files.
- Copy them under `/opt/glite/etc/config`, and start editing
- Fireman specific conf. file :
 - `glite-data-single-catalog.cfg.xml`
- Other xml files are needed for the general environment setting (glite-global), GSI (glite-security), R-GMA (glite-rgma-*)
- For these configuration is basic : attribute description is self-explaining
- Check that value for `JAVA_HOME` attribute in `glite-global.cfg.xml` is correct !

- **Essential parameters**

- VO Name served by the instance (more VO can be served)
- DB Name, user and password for database (configurator script will create it if not existing)
- Secure / Unsecure mode, switched through the `allow.unsecure.port` attribute.

If working in secure mode, the endpoint will be reachable only through https protocol at the port 8443, and communication will pass through GSI. Otherwise http and the port 8080 can be used

- **Once configured, launch the configurator script**

```
cd /opt/glite/etc/config/scripts/
python glite-data-single-catalog.config.py
```

- **The endpoint of the catalog (needed by IO server / client)**

can be retrieved from the value of the attribute path in

```
/etc/tomcat5/Catalina/localhost/glite-data-catalog-service-fr-
mysql.gilda-single-catalog.xml
```

- **The full endpoint is**

```
http[s]://<FQDN>:[8443|8080]/<path>
```

- If your IO server is configured properly, and your UI can “see” the Catalog (appropriate entry must be in `/opt/glite/etc/services`) you can execute **functionality tests**
- From the UI, try to copy and register a file
`glite-put <yourFile> lfn:///lfn_name`
`glite-catalog-ls /` should show the file just copied and registered
- Many more exercises in the hands-on session !

cretino chi legge

- **JRA1 Data Management homepage**
<http://cern.ch/egee-jra1-dm>
- **gLite FiReMan user guide**
 - Overview
<https://edms.cern.ch/file/570643/1/EGEE-TECH-570643-v1.0.pdf>
 - Command Line tools
<https://edms.cern.ch/file/570780/1/EGEE-TECH-570780-v1.0.pdf>
 - C/C++ API
<https://edms.cern.ch/file/570780/1/EGEE-TECH-570780-C-CPP-API-v1.0.pdf>
 - Java API
<https://edms.cern.ch/file/570780/1/EGEE-TECH-570780-JAVA-API-v1.0.pdf>
- **gLite Release 1**
 - <http://glite.web.cern.ch/glite/packages/R1.0/R20050331>
 - <http://glite.web.cern.ch/glite/documentation>

