



Enabling Grids for E-sciencE

Installing & configuring



Lightweight Middleware for Grid Computing

*Joachim Flammer
Integration Team, CERN*

*EMBRACE Tutorial,
Clermont-Ferrand
July 2005*

www.eu-egee.org



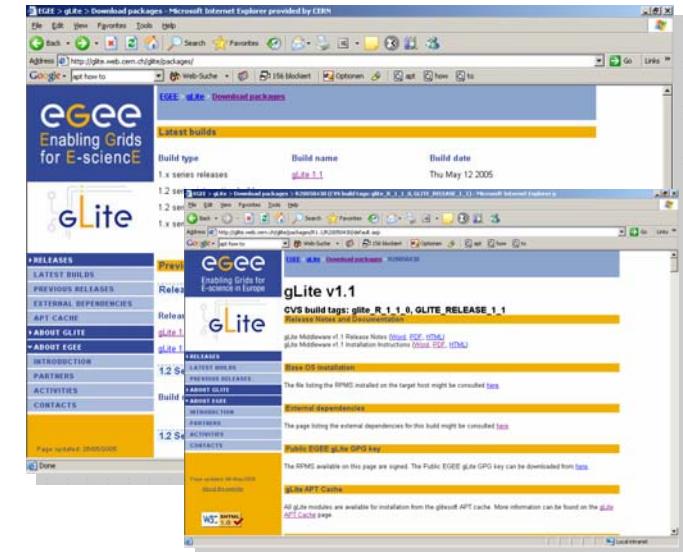
- **Deployment concepts**
- **Installation**
 - Installation scripts
 - APT
- **Configuration**
 - Concepts
 - Configuration files
 - Configuring a service
 - Configuring a site
- **Getting more information**

- **Modular software**
 - **Middleware is composed out of several high-level components/services**
 - Components can be installed separately
 - Each service is one deployment component
- **Complete distribution packages**
 - **Middleware is distributed in form of RPMs (and tarballs)**
 - Deployment module contains all necessary scripts and configuration files
 - All required external dependencies are available in RPM format
 - Downloaded together with RPMS
 - *Exception: non-freely distributable software like Java, ORACLE,...*
- **Decoupling of installation and configuration**
 - **Installation and configuration are two separate steps**
 - Allow different methods of installation
 - Allow different methods of configuration
 - *Individual services vs entire sites*
 - *possibility to incorporate gLite configuration into other configuration/system management systems (e.g. quattor)*
- **Homogenous installation and configuration**
 - **All gLite components are installed and configured in the same way**

Installing gLite ...

- **Install gLite via:**

- Download individual RPMs
- Use Installer scripts
 - Make scripts executable and run them
 - Downloads and installs all required RPMs including external dependencies
- APT repository (see next slide)
- Source/binary tarballs:
 - Available from gLite web site and EGEE web server:
jra1mw.cvs.cern.ch:/cvs/jra1mw



<http://glite.web.cern.ch/glite/packages>

for gLite components

<http://glite.web.cern.ch/glite/packages/externals/bin/rhel30/RPMS>

for the external dependencies rpms

1. Check if APT is already installed on your node:

```
rpm -qa | grep apt
```

2. Install APT as a root user (using the Scientific Linux repository):

```
rpm -ivh http://linuxsoft.cern.ch/cern/slc30X/i386/SL/RPMS/apt-0.5.15cnc6-8.SL.cern.i386.rpm
```

3. Put one of the following lines in a file inside the /etc/apt/sources.list.d directory (e.g. you can create a file named 'glite-rhel30.list'). Make sure the file has a ending of .list

```
rpm http://glitesoft.cern.ch/EGEE/gLite/APT/R1.2 rhel30 externals Release1.2
```

4. Update your apt repository

```
apt-get update
```

5. Upgrade the apt and all not-up-to-date packages installed on your system

```
apt-get upgrade
```

6. Get the gLite module of your choice:

```
apt-get install glite-xxxx-config
```

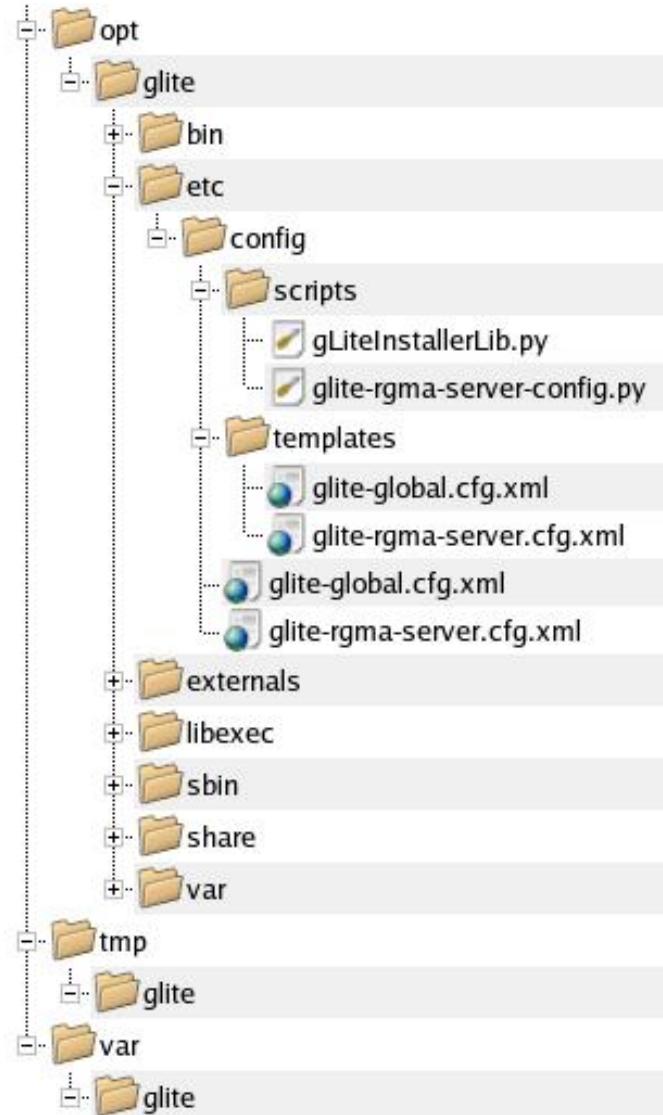
where xxx the name of the component.

APT downloads automatically all required dependencies.

More information about APT can be found at

<http://www.debian.org/doc/manuals/apt-howto/index.en.html>

- All the gLite software
 - is installed by default into /opt/gLite
 - Follows standard unix structure
- Installation area root can be adapted if necessary
- User interface (UI) and Worker Node (WN) can be installed in user area by non-root user



Configuring gLite ...

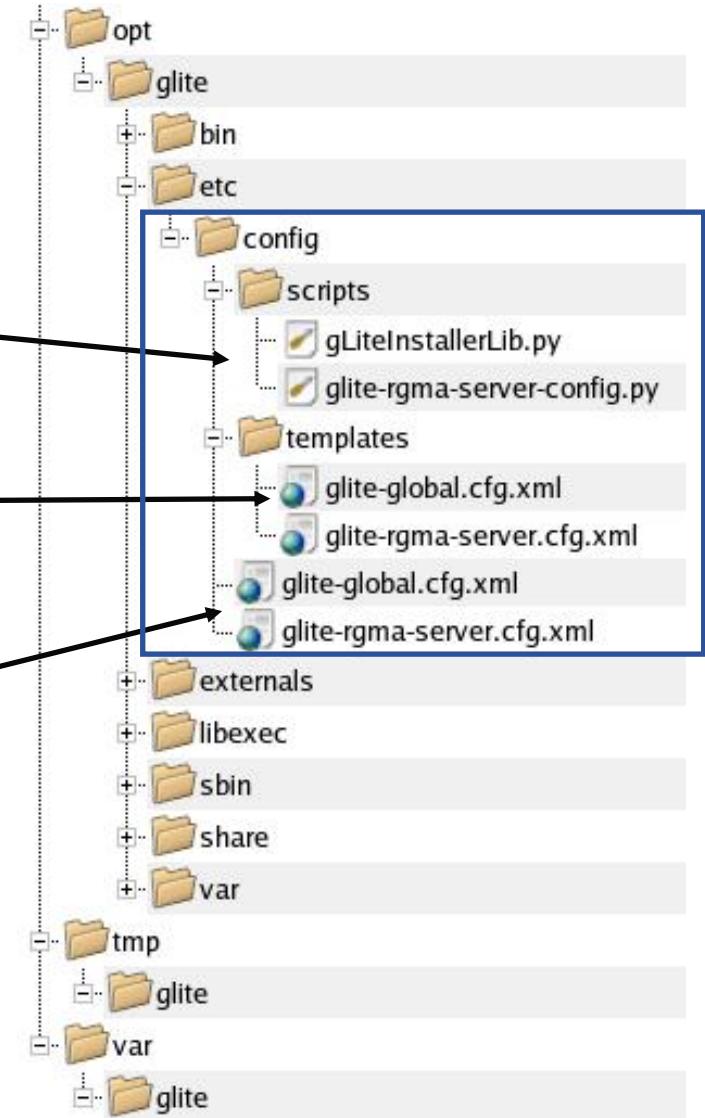
- **Each gLite deployment module contains**
 - Set of RPMS for the necessary internal & external dependencies
 - One or more configuration RPMs that provide the configuration files & scripts:
 - **glite-config-x.y.z-r.noarch.rpm**
 - *Contains global configuration files & scripts required by all gLite modules*
 - **glite-<service>-config-x.y.z-r.noarch.rpm**
 - *Contains configuration files & scripts required by a particular service*
 - ... other configuration rpms for some modules
- **Configuration RPM contains**
 - **Configuration scripts**
 - written in python
 - naming convention: **glite-<service>-config.py**
 - Additional configuration helper scripts that are used by service configuration scripts to configure various applications (e.g. mysql.py, tomcat.py)
 - **Configuration files**
 - xml encoded files
 - distributed as pre-configured templates
 - naming convention: **glite-<service>.cfg.xml**

- Configuration installed into:
/opt/glite/etc/config

Configuration scripts:
/opt/glite/etc/config/scripts

Configuration files templates:
/opt/glite/etc/config/templates

Configuration files:
/opt/glite/etc/config/



- An XML document is text not binary data – most famous example: HTML
- XML uses tags to structure its content:
 - Building blocks are elements:
 - content of an element is between start tag <element> and end tag </element>
 - Case sensitive <Person> ≠ <PERSON>
 - Element can have attributes
 - All attribute values are quoted by a double " quotes.
 - Tags may be nested but may not overlap.
 - Comments <!-- This is a comment -->

```
<!--one example entry of a person -->
<Person id="123" department="IT">
    <LastName>Blog</LastName>
    <FirstName>John</FirstName>
    <Projects>
        <Project>EGEE</Project>
        <Project>gLite</Project>
    </Projects>
</Person>
```

A good way to check for correct XML
is to just load it into a web browser as a file

- Configuration file in xml format
- One parameter entry contains
 - name of the parameter
 - description of the parameters (incl. Example values, Units etc.)
 - value of the parameter

```
<glite.user.name description="Name of the user account used to run the  
gLite services on this WMS node.  
[Example: testUser]."  
    value="changeme"/>
```

- Configuration supports variable substitution using the \${xxx} syntax:

```
<wms.config.file description="Location of the wms configuration file"  
    value="${GLITE_LOCATION}/etc/glite_wms.conf"/>
```

- The value of all the parameters with ‘changeme’ must be changed !!!

- **Arrays**

```
<rgma.registry.hostname description="Host name of the R-GMA registry service.  
[Example: lxb2029.cern.ch] [Type:'string']">  
    <value>host1</value>  
    <value>host2</value>  
</rgma.registry.hostname>
```

- **Instances: multiple occurrences of a set of parameters (e.g. for each service instance)**

```
<instance name="changeme" type="catalog">  
    <parameters>  
        <data.metadata-catalog.VO description="Name of the Virtual Organization which is  
            served by the catalog instance."  
            value="changeme"/>  
        <data.metadata-catalog.DBNAME description="Name of DB used for the catalog."  
            value="changeme"/>  
        <data.metadata-catalog.DBUSER description="DB user name to access the catalog."  
            value="changeme"/>  
    </parameters>  
</instance>
```

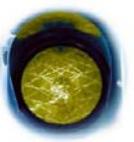
- In order to guide the user in configuring the services, all parameters of gLite are divided into 3 categories:

– User-defined parameters



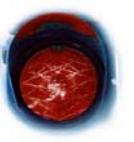
- have a default value of 'changeme'
- have to be replaced with values before running the configuration scripts
- in some cases user-defined parameters are alternative depending on the deployment scenario
 - Remove unused parameter or leave parameter empty (remove 'changeme' parameter)*

– Advanced parameters



- have always valid default values
- Can be changed by a user or system administrators to customize an installation e.g. depending on site policies

– System parameters



- have always valid default values
- System administrator can change them to modify the behavior of the system for special usages

```
<global>
  <parameters>
    <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->
    <!-- User-defined parameters - Please change them -->
    <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->

    <!-- Site configuration URL-->
    <site.config.url description="The URL of the Site Configuration file for this node. "
      value="changeme"/>

    <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->
    <!-- Advanced parameters - Change them if you know what you're doing -->
    <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->

    <!-- gLite configuration -->
    <GLITE_LOCATION value="/opt/glite"/>
    <GLITE_LOCATION_VAR value="/var/glite"/>
    <GLITE_LOCATION_LOG value="/var/log/glite"/>
    <GLITE_LOCATION_TMP value="/tmp/glite"/>

    <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->
    <!-- System parameters - You should leave these alone      -->
    <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->

    <installer.export.filename description="Full path of the script containing environment definitions."
      value="/etc/glite/profile.d/glite_setenv.sh"/>
  </parameters>
</global>
```

- **One global configuration file for all services:**
 - `glite-global.cfg.xml`
 - contains parameters with gLite wide scope
 - applicable to all services
 - parameters loaded first
 - can be overwritten by individual service configuration
- **Each service has (set of) individual configuration file(s)**
 - `glite-servicename.cfg.xml`
 - Specific to each service
- **Order of loading the configuration files is**
 1. Local `glite-global.cfg.xml` file for a given node
 2. Local service specific configuration files
 - New defined parameter overwrites previous definition

- All the configuration scripts
 - are located in `/opt/glite/etc/config/scripts`
 - naming convention: `glite-<service>-config.py`
 - Support a set of standard options:

```
[root@koogeki scripts]# ./glite-rgma-server-config.py --help
glite-rgma-server-config v. 5.0.0
Usage: python glite-rgma-server-config [OPTION...]
      -c, --checkconf      print the service configuration
      -v, --version        print the version of the configuration script
      -h, --help            print this usage information
      --configure          configure R-GMA server
      --start              start gLite R-GMA Server
      --stop               stop gLite R-GMA Server
      --status             print service status (0 = running, 1 = idle)
      --siteconfig=URL    URL of the site configuration file
```

Remark: gLite v1.1

no option	configures and starts service
--configure	option does not exist

- 1. Copy the configuration file templates from the templates directory to /opt/glite/etc/config/**
- 2. Adapt the configuration values in all affected files**
 - Change all the 'changeme' values – or remove them if not applicable
 - Adapt the configuration settings to your needs
 - Set the file permissions of the configuration files
- 3. Run configuration scripts to configure & start service**
 - `glite-<service>-config.sh --configure`
 - `glite-<service>-config.sh --start`
- 4. Verify that everything went ok**
 - Check the output of the script
 - `glite-<service>-config.sh --status`

- **Configure all services from a central point**
 - All gLite implement a method to load the configuration information from a remote URL
 - URL of configuration file can be specified via
 - site.config.url parameter in glite-global.cfg.xml file (normal operation)
 - as a command line parameter

```
glite-ce-config.py --siteconfig=http://server.domain.com/sitename/siteconfig.xml
```
- The site configuration file
 - contains a global section
 - Identified by <parameters> tag
 - can contain one section for each node to be remotely configured
 - Identified by <node name="xxxx"> tag
 - Name must be the value of the \$HOSTNAME environment variable
 - can contain combination of parameters from different services (e.g. if a node contains WMS and CE both parameter lists can be put to the same node section)

- **glite-global.cfg.xml** must be present on each node
 - **individual service config files not**
- Order of loading the configuration files is
 1. **Local glite-global.cfg.xml file for a given node**
 2. **Remote site config file**
 3. **Local service specific configuration files (if they exist)**
- New defined parameter overwrites previous definition
- Configuration information can be distributed over several files
 - **uses xml inclusion mechanism**
 - **Point gLite site configuration to the master file and reference other files inside master file**
 1. **In master file replace ‘normal’ <siteconfig> tag by**

```
<siteconfig xmlns:xi="http://www.w3.org/2001/Xinclude">
```

2. **Include files via**

```
<xi:include href="glite-xxx.cfg.xml"/>
```

Site configuration file example

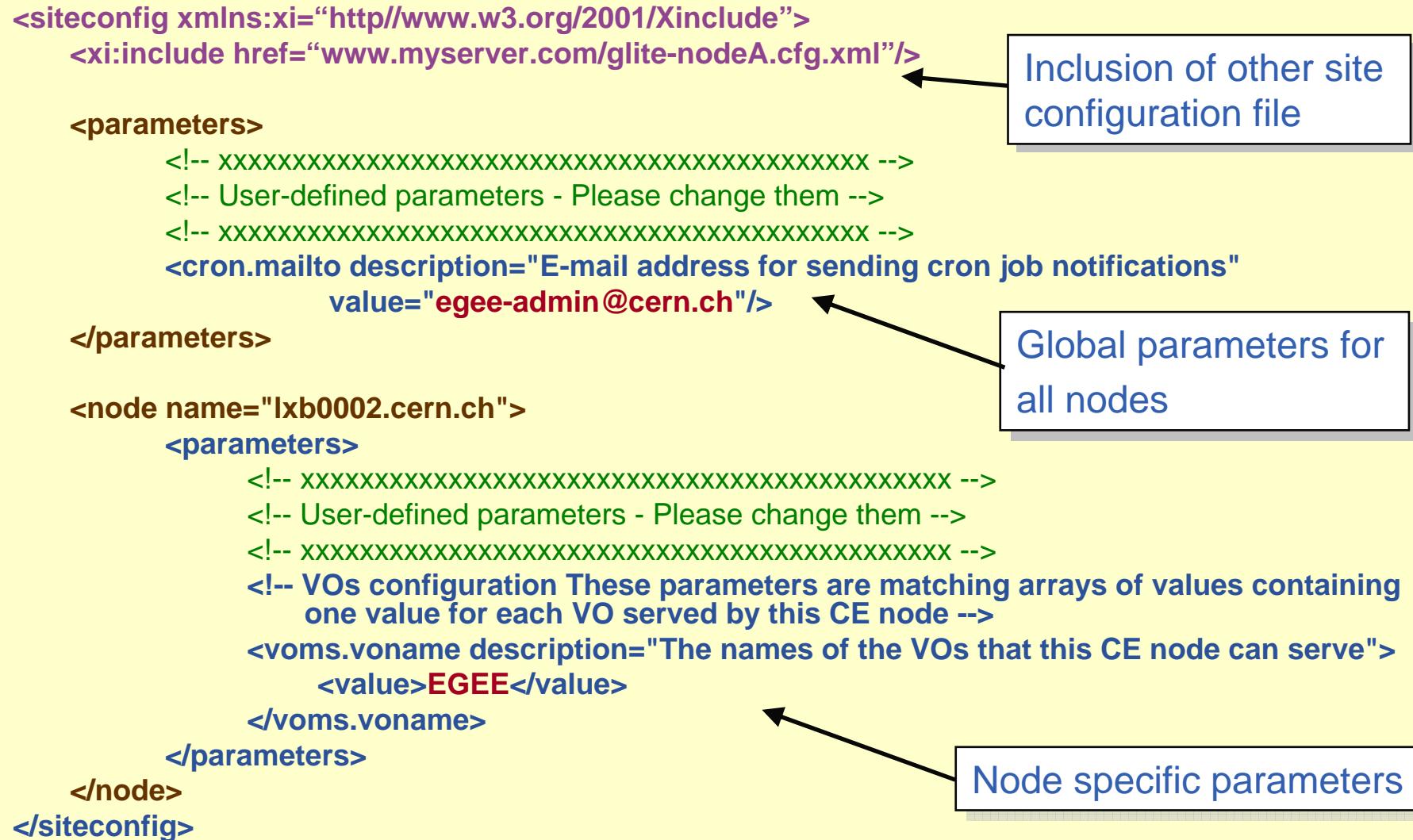
```

<siteconfig xmlns:xi="http://www.w3.org/2001/Xinclude">
    <xi:include href="www.myserver.com/glite-nodeA.cfg.xml"/>

    <parameters>
        <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->
        <!-- User-defined parameters - Please change them -->
        <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->
        <cron.mailto description="E-mail address for sending cron job notifications"
                      value="egee-admin@cern.ch"/>
    </parameters>

    <node name="lxr0002.cern.ch">
        <parameters>
            <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->
            <!-- User-defined parameters - Please change them -->
            <!-- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -->
            <!-- VO configuration These parameters are matching arrays of values containing
                one value for each VO served by this CE node -->
            <voms.voname description="The names of the VOs that this CE node can serve">
                <value>EGEE</value>
            </voms.voname>
        </parameters>
    </node>
</siteconfig>

```



Inclusion of other site configuration file

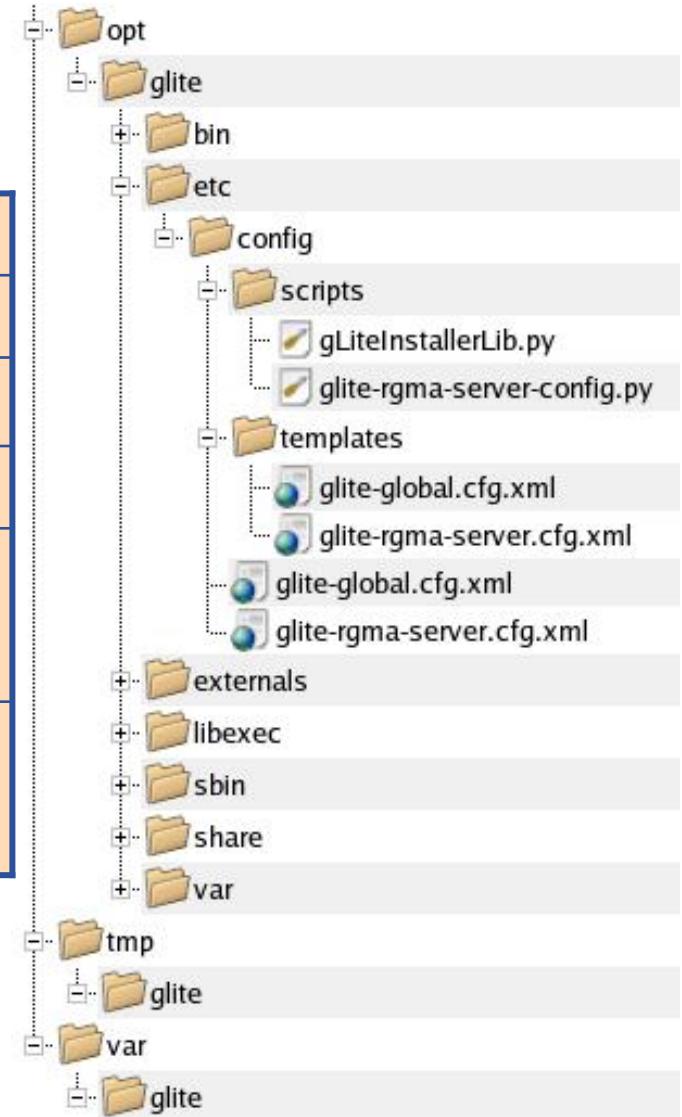
Global parameters for all nodes

Node specific parameters

- **Users under control of the middleware**
 - Proper environment is created automatically for all different kind of logins
 - gLite configuration scripts creates
 - `/etc/glite/profile.d/glite_setenv.sh`
 - `/etc/glite/profile.d/glite_setenv.csh`
 - contains all environment definitions to run the gLite services
 - file is automatically added to `.bashrc` for all user under direct control of the middleware (e.g. service accounts, pool accounts ...)
 - `.bash_profile` of the accounts is modified to `source .bashrc` and to set `BASH_ENV=.bashrc`
 - **Users not under control of middleware**
 - `source /etc/glite/profile.d/glite_setenv.sh` manually
 - **Override configuration**
 - Values in gLite configuration files can be overwritten by setting the appropriate key/value pairs in
 - `/etc/glite/glite.conf` system wide scope
 - `~/.glite/glite.conf` user-scope

Default environment variables

GLITE_LOCATION	/opt/glite
GLITE_LOCATION_VAR	/var/glite
GLITE_LOCATION_LOG	/tmp/glite
GLITE_LOCATION_TMP	/tmp/glite
PATH	/opt/glite/bin: /opt/glite/externals/bin: \$PATH
LD_LIBRARY_PATH	/opt/glite/lib: /opt/glite/externals/lib: \$LD_LIBRARY_PATH



- **gLite web page**
<http://www.glide.org>

- Documentation
 - Installation & configuration
 - gLite parts
- Download
- APT cache

The image contains two side-by-side screenshots of Microsoft Internet Explorer 6.0. The left screenshot shows the gLite website at <http://www.glide.org>. The main content area displays the "gLite Lightweight Middleware for Grid Computing" banner and a brief introduction. On the left sidebar, there's a menu titled "GLITE SUBSYSTEMS" with options like COMPUTING ELEMENT, DATA MANAGEMENT, ACCOUNTING, LOGGING AND REPORTING, EXECUTIVE & MONITORING, SECURITY, and WORKLOAD MANAGEMENT. The right sidebar has sections for "Want to know more about gLite?", "gLite News", and "How gLite web site evolved". The right screenshot shows the JRA1: Middleware website at <http://egee-jra1.web.cern.ch/egee-jra1/>. It features a navigation bar with links like Home, Tasks, Information, Publications, Infrastructure, Data Management, Search, and Measurement. The main content area includes sections for DOCUMENTS (with links to various reports and documents), PRESENTATIONS (with links to presentations from various events), and NEWS (with a list of recent news items).

- **gLite subsystems**
<http://egee-jra1.web.cern.ch/egee-jra1/>

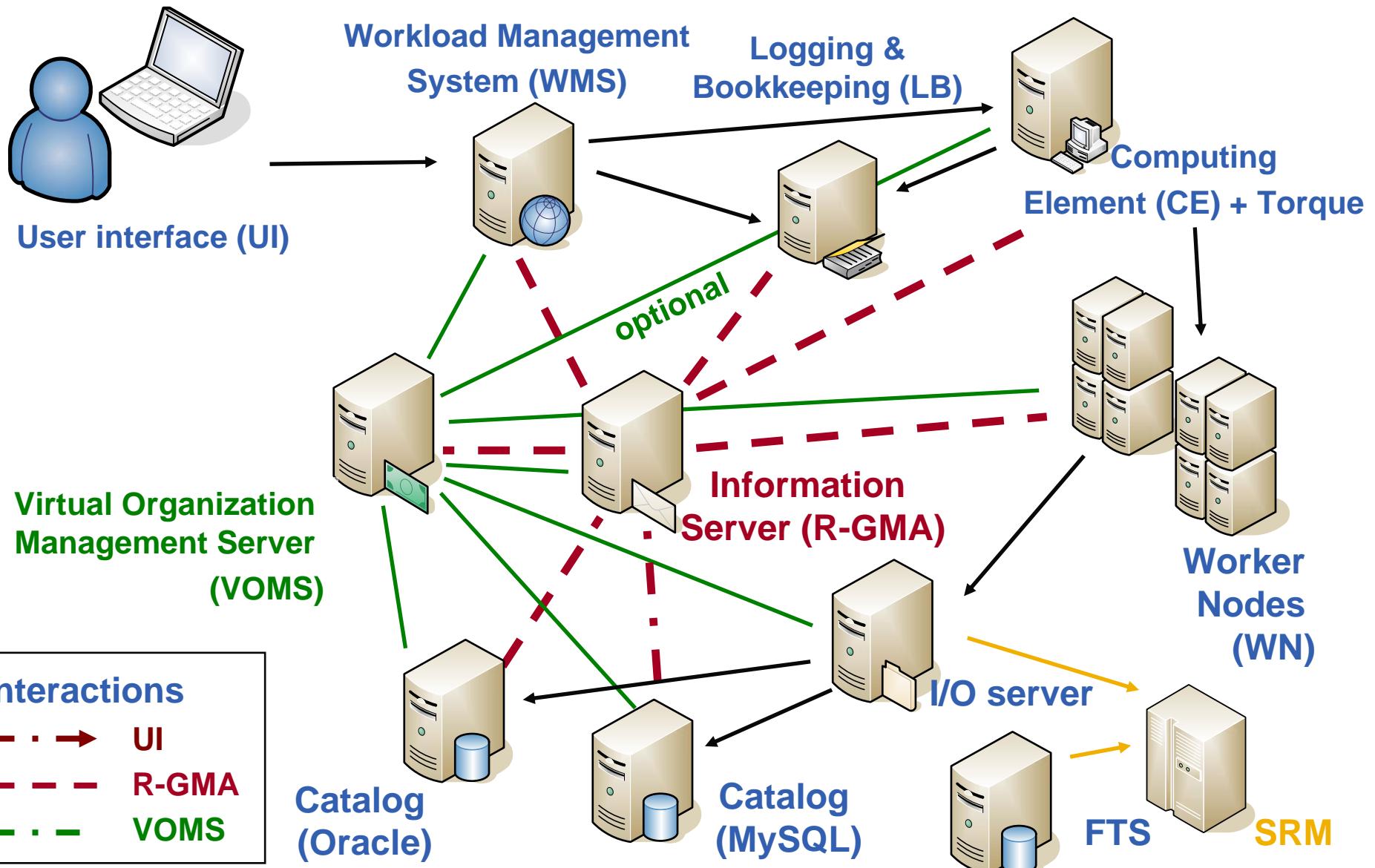
- **Contact:**
gite-discuss@cern.ch



- **EGEE homepage**
<http://egee-intranet.web.cern.ch/egee-intranet/gateway.html>
- **gLite homepage**
<http://glite.web.cern.ch/glite/>
- **gLite download**
<http://glite.web.cern.ch/glite/packages/>
<http://glite.web.cern.ch/glite/packages/R1.1/R20050430/default.asp> (Release 1.1.)
- **Documentation
(Installation guide, user guides, release notes)**
<http://glite.web.cern.ch/glite/documentation/>
- **Apt instructions**
<http://glite.web.cern.ch/glite/packages/APT.asp>
- **External dependencies**
<http://glite.web.cern.ch/glite/packages/externals/external-dependencies.htm>
- **JRA1 home**
<http://egee-jra1.web.cern.ch/egee-jra1/>

Thank you very much for your attention!





Overview – User Interface

