



Enabling Grids for E-science

EGEE Middleware



General presentation
Last update May 2007

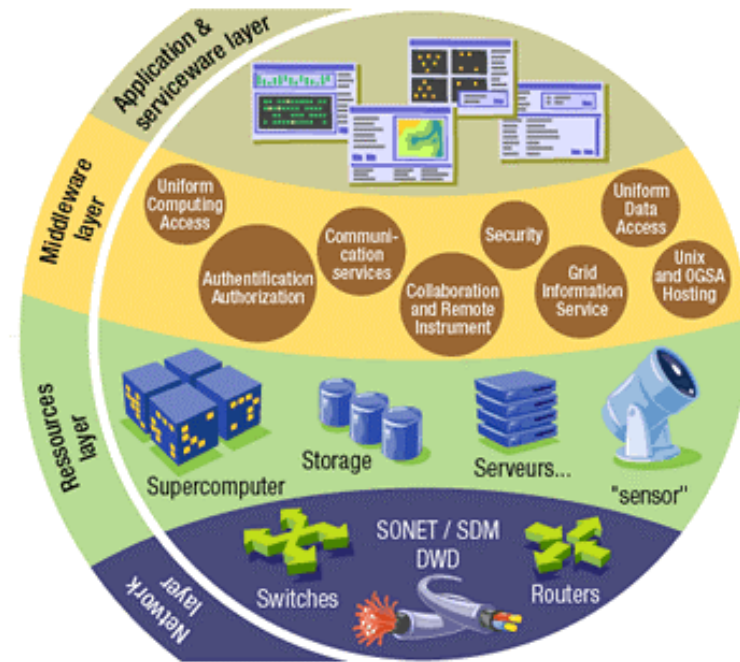
www.eu-egee.org



- The Grid relies on advanced software, called **middleware**, which interfaces between resources and the applications

- **The Grid middleware:**

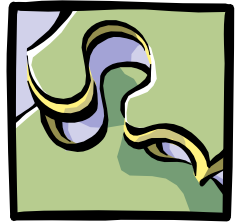
- Basic services
 - Secure and effective access to resources
- High level services
 - Optimal use of resources
 - Authentication to the different sites that are used
 - Job execution & monitoring of progress
 - Problem recovery
 - Transfer of results back to the user



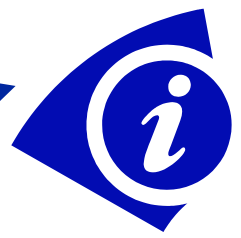
User Interface



Resource Broker



Information System

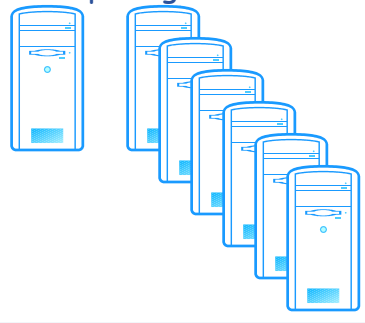


File and Replica Catalogs

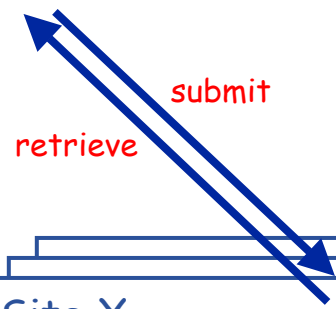
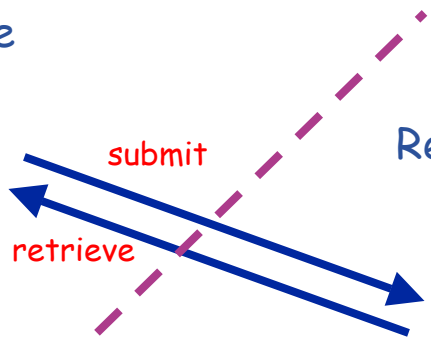
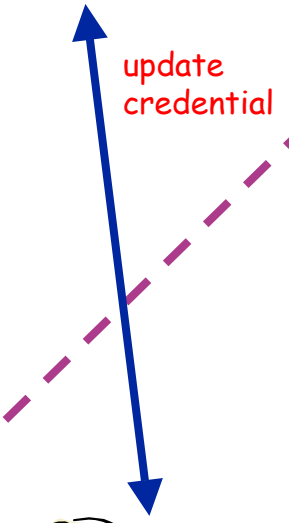
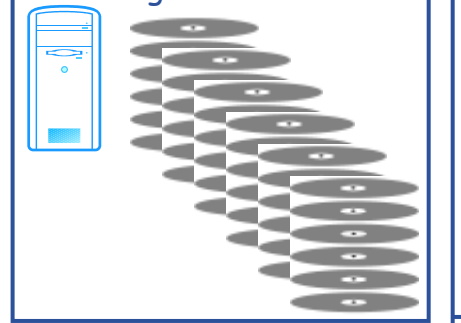


Site X

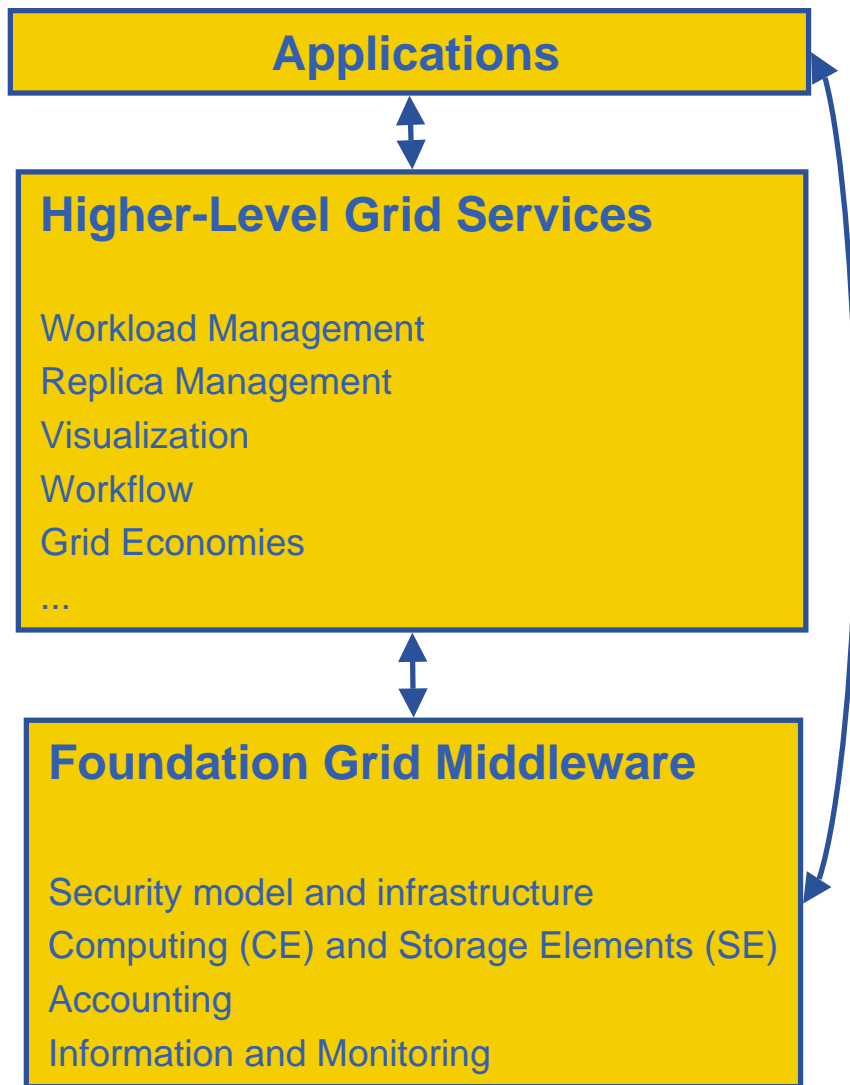
Computing Element



Storage Element



Authorization Service



- **Access for applications to:**
 - Higher-level Grid Services
 - Foundation Grid Middleware
- **Higher-Level Grid Services are supposed to help the users building their computing infrastructure but should not be mandatory**
- **Foundation Grid Middleware will be deployed on the EGEE infrastructure**
 - Must be complete and robust
 - Should allow interoperation with other major grid infrastructures
 - Should not assume the use of Higher-Level Grid Services

Key success factors for production quality software:

- **Strict software process**
 - Use industry standard software engineering methods
 - Software configuration management, version control, defect tracking, automatic build system, ...
- **Conservative in what software to use**
 - Be careful about “cutting-edge” software
 - Deployment on some 200 sites cannot assume a homogenous environment
 - middleware needs to work with many underlying software flavors
 - Be aware of evolving standards
 - Evolving standards change quickly (and sometime significantly cf. OGSI vs. WSRF) – impossible to keep pace on almost 200 sites

You will not develop and deploy your PhD project on a production Grid infrastructure

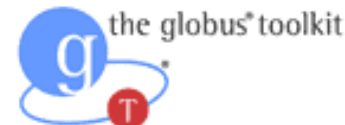
There is a long (and tedious) path from prototypes to production

- **gLite 3.0**

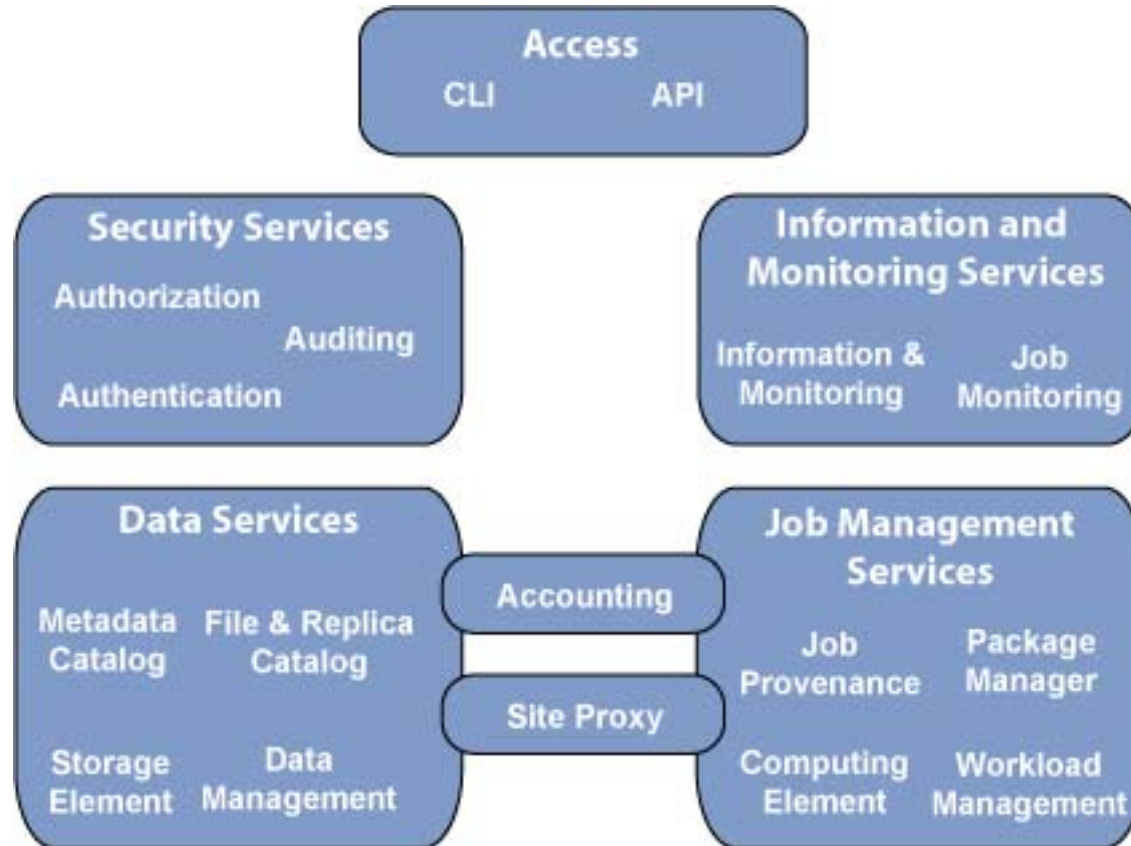
⇒ Merger of LCG 2.7 and GLite 1.5



- Exploit **experience and existing components** from VDT (Condor, Globus), EDG/LCG, and others
- Develop a **lightweight stack of generic middleware** useful to EGEE applications (HEP and Biomedics are pilot applications).
 - Should eventually deploy dynamically (e.g. as a globus job)
 - Pluggable components – cater for different implementations
- Focus is on providing a stable and usable infrastructure



gLite offers a range of services

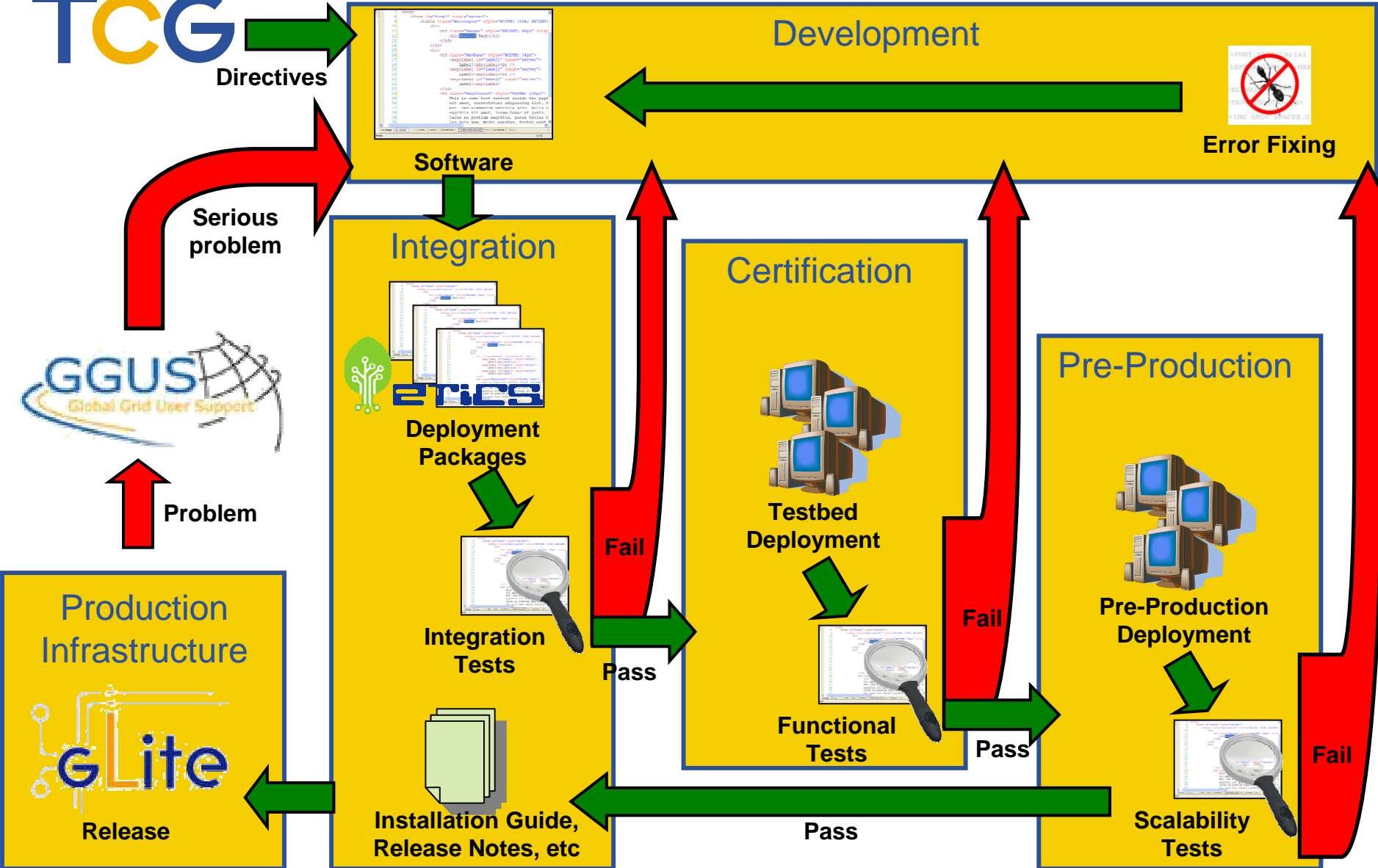


- **Centered around VOs**
 - It's ultimately the VO who gets resources allocated and need to decide how to best use them (share them among the VO users)

- **Distinguish between *infrastructure* and VO services**

- **Infrastructure services**
 - Operated and trusted by the resource administrator
 - Implement **site policies**
 - Including what share of the resources are allocated to a VO
 - Provide the required security, auditing, and accounting
 - Grid and standard services
 - E.g. batch system, gatekeeper, gridFTP, ...

- **Security:**
 - Different administrative domains interconnected on the Grid and the need to establish mutual trust
- **Data management:**
 - Need to store and transfer data on different storage systems using different access technologies
- **Workload:**
 - Computational tasks of thousands of users need to be managed and monitored on the available Grid resources



- gLite 3.0 merges **LCG 2.7** and **gLite 1.5**
- EGEE provides a **complete middleware stack**
 - ⇒ Security infrastructure, information system and monitoring, workload management, data management
- Developed according to a **well defined process**
 - ⇒ Controlled by the EGEE Technical Coordination Group
- Development is continuing to provide **increased robustness, usability and functionality**
- **Responsive** to user needs
- **Deployed on EGEE production service**

Lightweight Middleware for Grid Computing