Scenario (1)

- Increasing of computing and data infrastructures in the last decade
- A number of major projects have been established within Europe to share the growing amount of resources
- The project **EGI-InSPIRE** has been created to coordinate and maintain a sustainable European infrastructure to support European research communities and their collaborators
Scenario (2)

- **EGI** continues from the previous **EGEE (Enabling Grid for E-sciencE)** started on May 2010 for four years.
- EGI will guarantee the long-term availability of a **generic e-infrastructure** also collaborating with software providers in Europe to provide innovative solutions to deliver capability required by user communities.

Scenario (3)

- The **European Middleware Initiative (EMI)** is a collaboration of the four major software tools: **ARC, gLite, UNICORE** and **dCache**.
- The EMI project will take in charge to **integrate** these tools into a **common software layer**.
- EGI and EMI will work **together** providing European scientists and international organizations with well-designed services.
EMI Execution Services

• What is needed is a common access point to Computing Elements and Compute Nodes
• EMI Execution Services have been conceived as a set of standard web services for job submission and execution
• They’re designed to reach interoperability and to offer multiple capabilities for different users in the scientific community

EMI ES Specification (1)

• The EMI Execution Services Specification provides the interface description, data and state models, activity and resource specification for each single execution service
• The targets are the so-called Computing Elements (CEs), services providing access to computing resources usually localized at a site
EMI ES Specification (2)

This specification covers many different aspects:

- Common Interfaces to create and manage activities
- EMI Activity Description Language
- Data staging capabilities
- Activity related information
- Resource related information
- Delegation

EMI ES Specification (3)

![Diagram showing information, activity description, compute, data staging, and trust delegation relationships]
UNICORE EMI ES

The UNICORE EMI ES are composed by four main web services, associated to the corresponding interface elements:

- The DelegationService
- The CreateActivity
- The ActivityManagement
- The ResourceInfo

CreateActivity

- The CreateActivity interface method gets an Activity Description in XML format (ADL), then creates an activity java object to submit a single job
- The Activity Description is validated, the activity is created and the UNICORE Client gets back the url path to import input files into the StageIn directory
- When input files have been copied inside the directory (globus_url_copy) the Client notifies the Server with the notify method
DelegationService

- The **DelegationService** forwards a **CSR (Certificate Signing Request)**, the Client creates and signs a **Proxy certificate** with its Private Key and sends it to the Server.
- The Server stores the Proxy with its proper **delegation id** into an **hashmap**, more than a Proxy can be associated to different delegation ids for a single user (example with different VOs).
ActivityManagement

- **ActivityManagement** interface offers methods to perform `getStatus`, `remove`, `pause`, `resume`, `getInfo`
- When the `getStatus` method is invoked the ActivityManagement service retrieves the job status numeric value and maps it to the corresponding `ActivityStatus` object
UNICORE Services

- **UNICORE XNJS libraries** have been included as dependencies and a new code layer has been added above.
- **xnjs-module-core** has been included into the project configuration file, in order to import base XNJS functionalities.
- The **XNJS class** provides methods to start services and support job submission.

Core Services

![Diagram showing UNICORE client requests, EMI UNICORE ES, xnjs-module-core, XNJS Core Services, and job running status.]
Next Steps

- Common Authentication Libraries
- Extended Trust Delegation and SAML
- Storage Data Management

Thank you!

EMI is partially funded by the European Commission under Grant Agreement RI-261611