
public static
if(securityP
return se
public static final String
he target system factory
"<service name=\""+TSF+"\"
"<interface class=\"" + Targ
"<implementation class=\""
"/>"
"</service>"
ies;
he target system service
"<service name=\""+TSS+"\"
terface class=\"" + Target
plementation class=\""
/>"
ervice>"
job management servi
ervice name=\""+JM
nterface class=\""
mplementation cl
"/>"
/service>"
e){
he storage manage
"<service name=\""
"<interface class=
"<implementation
"/>"

UNICORE

UNICORE

# Table of Contents

UNICORE

# 10 reasons to use UNICORE

### Manager

1. Mature, well-proven Grid technology from Europe; widely used in European and national Grid projects, initiatives, and infrastructures

2. Made for Supercomputing / HPC Grids as well as Grids of Clusters and single PCs

3. Fast and competent support directly from the developers

### User & Resource Provider

4. Easy to install and configure clients and services

5. Intuitive graphical user interface and powerful command line client for seamless access to resources

6. Application integration mechanisms on the client, services and resource level

7. Mature workflow support tightly integrated but extensible for different, domain-specific workflow languages and engines

8. Interoperability through common open standards

### Developer

9. Source code (all Java) is available and easy to modify

10. Easily extensible with own developments with the option of integration in official releases

UNICORE FORUM





# UNICORE Forum e.V.

## Overview

The UNICORE Forum e.V. was founded in December 1999 by developers, leading European HPC centres, and supporting hardware vendors as a non-profit association to promote the development and distribution of the UNICORE Grid technology.

Membership is open to users and developers of Grid software as well as to hardware vendors. The UNICORE Forum e.V. has 25 members.

Details can be found at **www.unicore.eu/forum/members**.

The statutes can be found at **www.unicore.eu/forum/statute**.

## Objectives

- foster the distribution and use of UNICORE
- organize workshops and booths at major conferences
- support presentations at conferences
- publish and maintain the specifications
- coordinate further development
- certify implementations and extensions

## Membership

According to §9 of the statute membership is open to all organisations (private or public) which may contribute to the goals of the UNICORE Forum.

More details can be found at **www.unicore.eu/forum/membership**.

## Technical Advisory Board

### Objectives

- develop the future strategy and roadmap of UNICORE
- drive and monitor the open source development of UNICORE
- prepare technical proposals regarding UNICORE
- evaluate technical proposals regarding UNICORE
- advise the Board of the UNICORE Forum on request

### Members

Krzystof Benedyczak (ICM – Warsaw University)
Xavier Delaruelle (CEA/DAM Ile de France)
Bastian Demuth (Forschungszentrum Jülich)
Donal K. Fellows (University of Manchester)
Roger Menday (Fujitsu Laboratories of Europe)
Gert Ohme (T-Systems SfR)
Bernd Schuller (Forschungszentrum Jülich)

## UNICORE @ ISC 2011


## Contact

UNICORE Forum e.V.
Chairman of the Board
c/o Karlsruhe Institute of Technology (KIT)
Steinbuch Centre for Computing
Prof. Dr. Achim Streit
Hermann-von-Helmholtz-Platz 1
76344 Eggenstein-Leopoldshafen
Germany

Email: unicore-forum-board@unicore.eu

UNICORE
OPEN SOURCE

# Community

UNICORE is developed within an open, international developer community. **www.unicore.eu** serves as the main entry and contact point for the UNICORE community. It contains documentation, information about UNICORE-related events, highlights of developments made in the community, downloads, links to the software repository as well as communication channels.

### Documentation

Any information about UNICORE and its components can be found at the main web site **www.unicore.eu/documentation**. The FAQ may already answer the most urgent questions. More in depth manuals are available for the core components. A comprehensive overview of the UNICORE 6 architecture with a clickable map is also available.

The tutorials section provides introductory documents and installation guides, as well as video tutorials showing how to use UNICORE in practice. Presentations that were given at various occasions are also available.

UNICORE has been presented in papers at many scientific conferences, workshops and other occasions. Select research papers are available in the documentation area.

In addition, snippets of documentation for components under development can be found in the Wiki at **sourceforge.net/apps/mediawiki/unicore/**

### Events

UNICORE participates in a number of regular events throughout the year. First and foremost, there is the UNICORE Summit, an annual event that invites researchers and practitioners working with UNICORE in the areas of Grid and distributed computing, to exchange and share their experiences, new ideas, and latest research results on all aspects of UNICORE.
**www.unicore.eu/summit**

Secondly, members of the UNICORE community organize and participate in specifically targeted training events such as the International Summer/Winter School on Grid Computing. These events provide deep insight and hands-on sessions to UNICORE.
**www.unicore.eu/events**

# SourceForge.net

In 2004, UNICORE has been made available as open source in a SourceForge project. The source code and other downloads are reachable from **www.unicore.eu/download** or directly at SourceForge.net under **sourceforge.net/projects/unicore/files**.

### Mailing Lists

The primary contact point with the UNICORE community and developers is the UNICORE Support mailing list **unicore-support@lists.sourceforge.net**. It is open to the public and should be used for any technical question about UNICORE.

Notification about updates around UNICORE, e.g. new releases of the server, clients, or other, will be sent via the UNICORE Announce list **unicore-announce@lists.sourceforge.net**.

Other mailing lists are available for more close contact to the community. One can subscribe for notifications about new or updated tracker items or updates of the source code. Developer questions can be posted to a specific developer list.

Subscription to any of the above mentioned lists can be done at **sourceforge.net/mail/?group_id=102081**. Mailing list archives are available from the same location.

### Trackers

Bug reports and feature requests are tracked in the respective trackers at **sourceforge.net/tracker/?group_id=102081**.

### Code Repository

The Subversion software repository at SourceForge.net is the main UNICORE source code repository. It can be found at **unicore.svn.sourceforge.net/svnroot/unicore**.

### Test Grid

In order to quickly get started evaluating and using UNICORE 6, one can look at the UNICORE Test Grid and sign up for a temporary certificate to use the Test Grid for a limited amount of time. Users of the Test Grid will get the same impression as users of a full UNICORE Grid infrastructure.

The Test Grid is available at **www.unicore.eu/testgrid**.

Both the UNICORE commandline client (UCC) and UNICORE Rich Client (URC) provide a simplified registration procedure to the Test Grid.

UNICORE

# Grid Infrastructures

## XSEDE – Extreme Science and Engineering Discovery Environment

Scientists, engineers, social scientists, and humanities experts around the world—many of them at colleges and universities– use advanced digital resources and services every day. Things like supercomputers, collections of data, and new tools are critical to the success of those researchers, who use them to make us all healthier, safer, and better off.

XSEDE integrates these resources and services, makes them easier to use, and helps more people use them. XSEDE supports 16 supercomputers and high-end visualization and data analysis resources across the US. Job submission to high performance computing resources is performed via OGSA-BES compliant UNICORE 6 services, whereas interoperable and lightweight Genesis II services handle job submission on smaller computers.

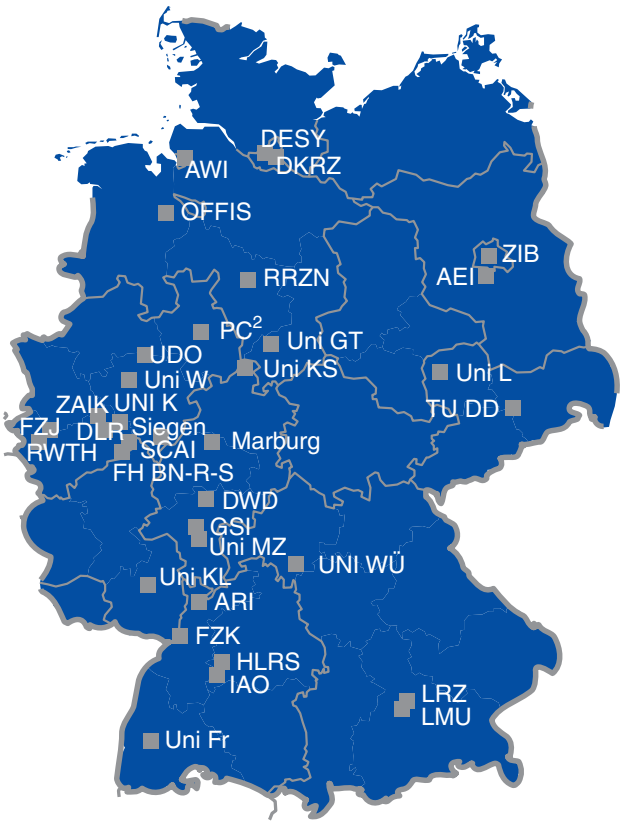

## PRACE – Partnership for Advanced Computing in Europe

PRACE (**www.prace-ri.eu**) is a unique persistent pan-European Research Infrastructure providing access to high-end supercomputing resources (Tier-0 systems). UNICORE is a core service for the access to PRACE HPC systems.


## SKIF-GRID federation –

SKIF-GRID (skif-grid.botik.ru) is a Program of the Belarusian-Russian Union State offering supercomputing resources to support research. It provides access to its resources through UNICORE 6.

The figure shows the partners in SKIF-GRID.




The figure shows the resource providers in D-Grid.

## D-Grid – The German Grid Initiative

The German national e-Science infrastructure (**www.d-grid.de**) offers access to its resources through all of the middlewares UNICORE, Globus and gLite to allow each scientific community (virtual organisation) to use the middleware best suited for their applications.

D-Grid offers Grid services for scientists, industry, and business, including applications in the construction industry, finance, aerospace and automotive, enterprise information and resource planning systems, geographical data, and general IT services.

D-Grid is migrating towards NGI-DE (**www.ngi-de.eu**) the National Grid Initiative for Germany, which is part of the European Grid Infrastructure EGI (**www.egi.eu**). EGI´s software infrastructure consists of components drawn from open source or commercial providers by means of the Unified Middleware Distribution (UMD). UNICORE is part of UMD.

UNICORE

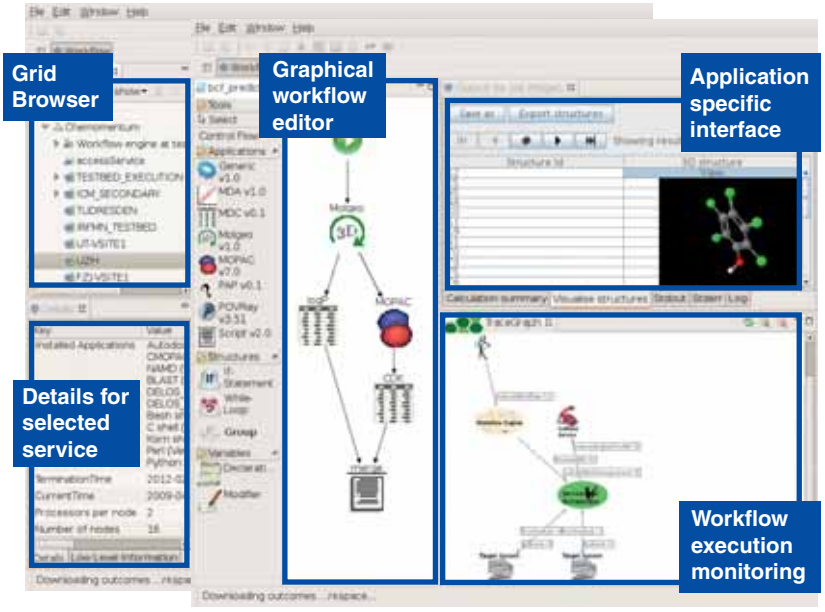
# Client Software

## The UNICORE Rich Client (URC)

The URC is an extensible Grid client framework based on Eclipse.

### Features

The user interface allows for accessing and browsing any UNICORE 6 based Grid. Computational jobs and workflows can be created with graphical editors and can be submitted to the appropriate execution services. Application specific graphical user interfaces can be easily integrated in order to obtain a high level view on the input and output data of scientific applications. Monitoring job and workflow execution is supported, as well as transferring remote files and administering Grid services.




Grid browsing, workflow editing and execution monitoring with the UNICORE Rich Client.

### Extension Points

Predefined extension points can be used for augmenting the framework in different ways. Thus, new Grid services and additional storage types can be introduced without changing any existing code. The workflow editor can easily be extended with new types of workflow structures and activities.

## The UNICORE Commandline Client (UCC)

The UCC is a versatile commandline tool that enables users to access all features of the UNICORE 6 service layer from a shell or scripting environment. Like the UNICORE Rich Client, it allows to run jobs and workflows, monitor their status and retrieve generated output.

Additional features include a powerful batch mode for submitting multiple jobs in an automated fashion, listing and searching remote storages, transferring files. There are administrative tools for checking site properties, or to perform clean up operations.

An important quality of the UCC is its extensibility. New commands can easily be added, and the “run-groovy” command allows the execution of scripts written in the Groovy programming language. A dedicated UCC mode for the popular Emacs editor is also available.

## The High Level Grid API (HiLA)

HiLA is an API for accessing Grid resources through different middleware in a consistent manner. Although forged for UNICORE 5, no interface changes were necessary for supporting UNICORE 6, due to its high level of abstraction. Thus, a smooth transition was ensured. There is also an implementation of HiLA that uses the standards-based job submission with OGSA-BES.

HiLA is employed in various Grid client libraries, e.g. in the DEISA Services for Heterogeneous management Layer (DESHL). It can be used as a convenient toolkit for building web interfaces or command line environments for Grids.

HiLA 2 has been designed with changes and additions based on lessons learnt and feedback from developers, who used the library in their projects. The latest release of HiLA is 2.2 and has been published in February 2011.

## Further Reading

- URC User Manual
  www.unicore.eu/documentation/manuals/unicore6/files/RichClient.pdf

- Video tutorial for the URC
  www.unicore.eu/documentation/tutorials/unicore6

- Manual for the UCC
  www.unicore.eu/documentation/manuals/unicore6/files/ucc

- Developer’s guide and programming reference for HiLA
  www.unicore.eu/community/development/hila-reference.pdf

UNICORE

# Application GridBeans

The graphical UNICORE Rich Client (URC) provides a comfortable and easy access to UNICORE 6 managed resources. Among other functionalities, it supports the creation and configuration of various types of application Grid jobs via feature rich interfaces, so called GridBeans. Scientists can create an application Grid job with less effort, by using the GridBeans graphical user interface that represents the application's behavior. It avoids that users need knowledge about the format of application start parameters or have to be aware of technical details of submitting and executing jobs in the Grid.

The GridBean concept is a plugin technology for graphical UNICORE clients that abstracts from scientific applications. Fundamentally, it offers methods for creating standard job descriptions and submitting jobs via standardized Web services. Furthermore, interfaces for file input/output, resource management and environment variable are provided for the users interactions.

Application specific GridBeans offer additional specific graphical user interfaces, which provide user changeable representations of application parameters and input/output data. In addition, other extensions are possible within the Eclipse environment, e.g. launching of external tools for visualisation. GridBeans can be used for single jobs as well as in workflows.

The left and middle screenshot show the GridBean for the application MDA. On the right, the GridBean for the molecular dynamic application AMBER is shown.

## GridBean Community

### Standard GridBeans

A set of default GridBeans is included in the URC distribution. It includes a Script GridBean, which provides an interface for editing and submitting any kind of scripts and a Generic GridBean, which offers generic panel elements for individually configuring and submitting any application installed on the target system.

To install additional GridBeans, they only need to be copied in the appropriate folder of the URC installation.

### Life Science

Several GridBeans were developed for applications from the life science domain. They can be found at the UNICORE Life website. Some to mention are MDA, Molgeo and Mopac, which were developed in the EU project Chemomentum.

### Others

Scientific GridBeans have been developed for many applications in various domains including mathematics, chemistry, biology, physics. Among others, a NAMD and AMBER GridBean, which is used in the field of molecular dynamics, a Gaussian GridBean used for molecule structure simulations, and a POVRay GridBean, which is used in visualisation, have been already developed by different communities.

## GridBean Development

UNICORE 6 users can develop their own application GridBeans. Required is a plugin, consisting of one application model and a variable number of panels, thus the developing effort is low. The model represents the application parameters, and the panels represent the graphical user interfaces for the model, input/output files and other information, like an application guideline. Typically, panels consist of easy GUI elements, but they may also be very complex, as for the visualization of output files.

## Further Reading

- UNICORE Life website
  http://sourceforge.net/projects/unicore-life/support

- Software download
  www.unicore.eu/download

- Developments from the community
  www.unicore.eu/community/development

UNICORE

# Services and Architecture

## Overview

UNICORE is a Grid middleware technology for scientific and commercial domains providing seamless access to computational and data resources.

### Portability

Being Java-based, all UNICORE 6 needs to run is Java SE 1.5 or later.

### Interoperability

UNICORE 6 uses HTTPS-based Web services as well as several common open Grid standards.

### Security

Access to UNICORE 6 is governed by authentication through a HTTPS gateway and authorisation via a rule-based access control engine.

### Service-orientation

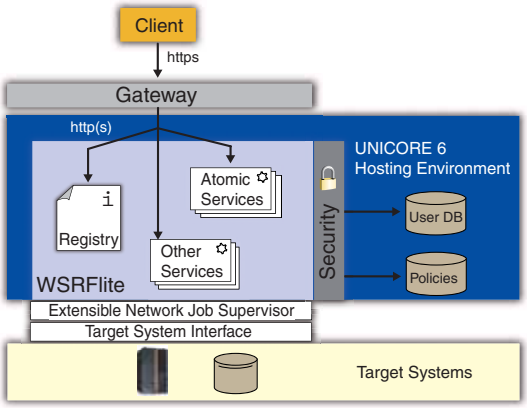
All services accessible to users are implemented as Web services and cover, among others, use cases such as target system access and job management.

### Extensibility

The modular architecture and open-source character of UNICORE 6 provide for ease of extensibility.

### Scalability

Multiple UNICORE 6 installations can be combined to form a distributed, multi-organisational Grid allowing for thousands of jobs.




The figure on the left shows the major building blocks of the UNICORE 6 architecture.

## Features

The UNICORE 6 architecture consists of the following major building blocks.

### Security

In order to establish a secure communications channel, a user and UNICORE 6 mutually authenticate each other using X.509 certificates. UNICORE 6 uses a user database for authenticating registered users and authorises users against a rule-based security policy. The security policy maps users to Grid resources and to the user actions permitted on these.

### Services

The service layer of UNICORE 6 is implemented using the WSRFlite hosting environment conforming to the Web Service Resource Framework standard as specified by OASIS (Organization for the Advancement of Structured Information Standards). This standard essentially specifies stateful Web services which are ideally suited for representing distributed computing resources via a standardised interface. Out of the box, UNICORE 6 provides a service registry, the UNICORE 6 Atomic Services (file stage-in and stage-out, job submission, job management), services for workflow support and service orchestration, as well as implementations of emerging Grid standards. Furthermore, WSRFlite offers tools for implementing custom UNICORE 6 services.

### Execution Management

The XNJS (Extensible Network Job Supervisor) is a processing engine tailored to the needs of UNICORE 6. Apart from managing job execution, it handles the mapping of abstract job definitions to real executables on the used target systems.

### Target System Interface

The TSI (Target System Interface) allows for calling executables as well as accessing and modifying files on target systems. A typical UNICORE 6 installation comes with a set of different TSI implementations catering to different target system types, such as Torque, LoadLeveller and SGE. Adding a custom TSI is straightforward.

## Further Reading

– Detailed UNICORE 6 Architecture Description
  www.unicore.eu/unicore/architecture.php

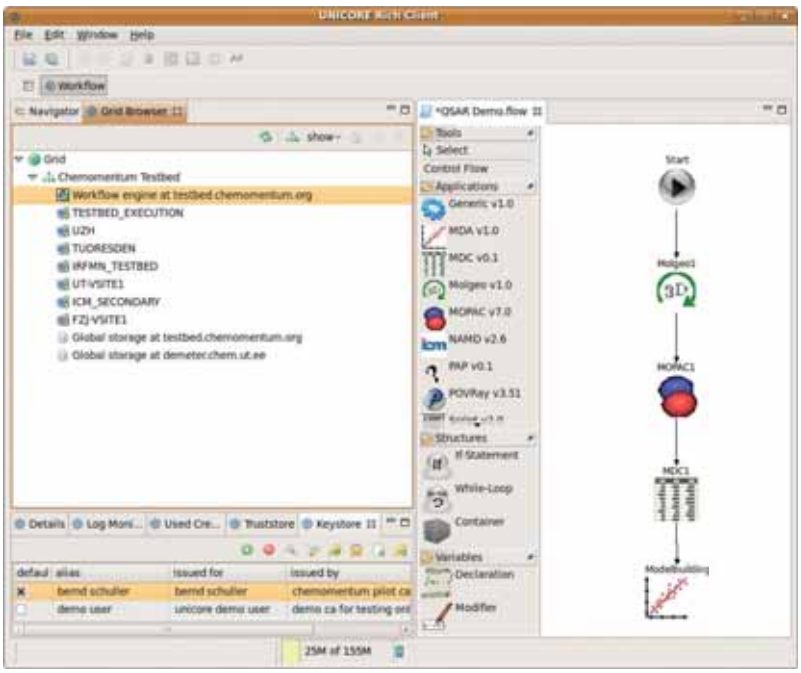
UNICORE

# Workflows

## Overview

Integrated workflow support is a major strength of UNICORE. In order to accomodate the wide range of requirements in scientific workflows, the UNICORE 6 workflow system has been designed from the ground up for flexibility, scalability, extensibility and ease of use.

The workflow features of UNICORE 6 can be used from both the graphical UNICORE Rich Client (URC) as well as from the command line client UCC.




Workflow editing in the UNICORE Rich Client.

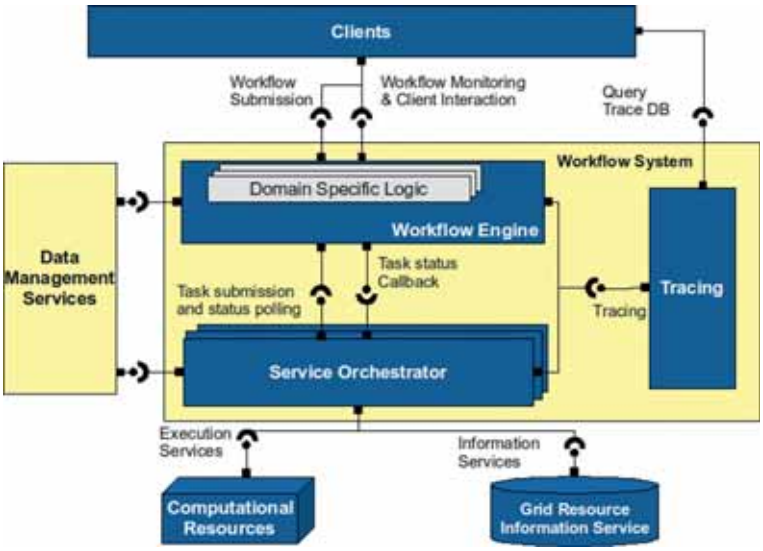
### Supported Workflow Features

The workflow is composed of individual application invocations and control structures. It supports common features such as parallel and sequential execution of tasks, loops, conditional transitions (e.g. if-else constructs) and workflow variables.

The workflow system is ideally suited for scientific applications and parameter studies.

## Architecture of the Workflow System

In the following the major parts of the workflow system are described.




Component view of the UNICORE 6 Workflow system.

### Workflow Engine

The workflow engine offers a Web service frontend that clients use to submit workflows. It is responsible for processing the workflow and submitting tasks to the service orchestrator layer. The system allows to use domain-specific workflow languages by plugging a custom converter into the workflow engine. The service-oriented architecture even allows replacing the complete workflow engine by another one.

### Service Orchestrator

The service orchestrator layer is responsible for executing the individual tasks in a workflow, handling job execution and monitoring on the Grid. Different brokering strategies are implemented to find the most suitable resource for each workflow step. Other brokering strategies can easily be plugged-in. To increase performance and scalability of the service orchestrator layer, multiple instances can be installed.

### Data Management

The workflow system can interface to data management solutions. A location mapping service is used to deal with file locations in an abstract manner.

### Tracing

An optional tracing Web service is available for collecting the timestamps of all workflow operations. For example, the trace can be queried and used to generate performance data.

## Further Reading

– Detailed UNICORE 6 Architecture Description
www.unicore.eu/unicore/architecture.php

UNICORE

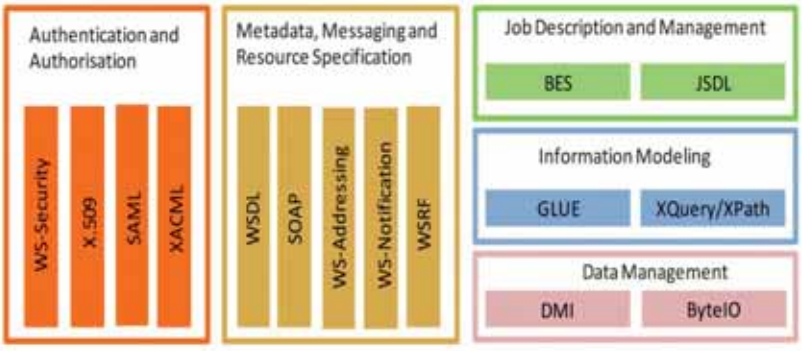
# Standards Adoption

## Overview

The standards compliance on every tier of UNICORE 6 attains seamless deployment and usage in production infrastructures despite the increasing complexity and variety of other Grid technologies and systems. The compliance of UNICORE 6 with open standards is an important means of fostering interoperability with other Grid middlewares, and thus it widens the horizon of users and application developers.

## UNICORE and Standards

**Authentication and Authorisation** are essential elements and critical for any Grid technology. Achieving interoperability in this context plays a significant role by adopting off-the-shelf components. UNICORE's security is based on Public Key Infrastructure (PKI) based model with authentication through X.509 certificates, while Security Assertion Markup Language (SAML) and eXtensible Access Control Markup Language (XACML) are used for role and attribute based resource access. Moreover, UNICORE 6 leverages WS-Security to protect its services.




**Metadata, Messaging and Resource Specification:** Since UNICORE 6 is a Web services based middleware, the underlying message exchange between the service and client is done via Simple Object Access Protocol (SOAP), and Web Services Description Language (WSDL) is used to describe Web service interfaces. UNICORE comes with a complete implementation of a Web Services Resource Framework (WSRF) hosting environment, by which Grid resources

(jobs, target systems, and storages) are implemented as stateful Web services. Resources are identified by employing the WS-Addressing specification in order to address Web service endpoints, and to secure end to end service identification. For notification mechanisms, WS-Notification is implemented to achieve the Web services based Publish/Subscribe model.

**Job Description and Management** is one of the imperative functions of any Grid technology. UNICORE 6 provides an implementation of the OGF's Basic Execution Service (BES) and Job Submission Description Language (JSDL) in order to provide interoperable job management and submission functions.

**Information Modeling:** UNICORE 6 offers an implementation of OGF's Grid Laboratory Uniform Environment (GLUE) schema which serves as information model for managing and monitoring backend Grid resources. UNICORE 6's Common Information Service (CIS) consumes and publishes resource information using the GLUE model to enable monitoring, discovery and advertisement functions. CIS clients can easily search for Grid resources by using XQuery and XPath languages.

**Data Management** functions in Grids generally includes access, management and transport of data. For interoperability UNICORE 6 realises data management by using the W3C's HTTP and OGF's ByteIO implementation.

## Standards Bodies

- Open Grid Forum (OGF), www.ogf.org
- Organization for the Advancement of the Structured Information Standards (OASIS), www.oasis-open.org
- Web Services Interoperability Organization (WS-I), www.ws-i.org
- World Wide Web Consortium (W3C), www.w3.org
- Internet Engineering Task Force (IETF), www.ietf.org

UNICORE is an organizational member of the Open Grid Forum.

## Further Reading

- Detailed UNICORE 6 Architecture Description
  www.unicore.eu/unicore/architecture.php

UNICORE

# Interoperability Benefits for e-Science Applications

## Research Using e-Science Infrastructures

The term e-Science stands for research that focuses on global collaboration in key areas of science by jointly using interoperable next generation computing infrastructures.

These infrastructures are known as Grids, e-Science infrastructures, or e-infrastructures.

The power of e-Science infrastructures is accessible to end-users using a wide variety of functionality available in Grid middleware distributions and other Grid software providers (e.g. data technologies).

Unfortunately, most software providers developed proprietary Grid middleware distributions over the years leading to a variety of non-interoperable solutions.

## Using Open Standards to Enable Interoperability

One of the key features of UNICORE 6 is its adoption of common open standards from standard bodies such as OGF, OASIS, W3C, WS-I, and IETF.

These adoptions are used with e-Science applications in order to drive the standardization process thus taking important application experience into account.

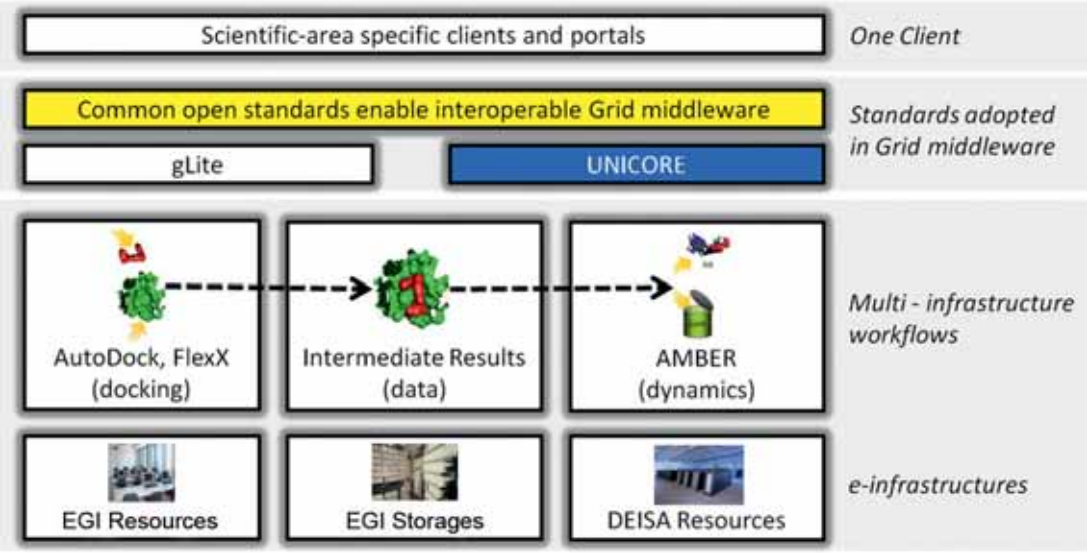
The advantage of adopting open standards in UNICORE 6 allows for the seamless use of UNICORE 6 components by other technologies. Furthermore, UNICORE 6 can easily use interoperable components provided by other software vendors thus enabling the access to a broader variety of computational resources.

This, for instance, allows for jointly using High Throughput Computing (HTC) as well as High Performance Computing (HPC) in complex application workflows.

## Fostering e-Science Applications

The increasing complexity of e-Science applications that embrace multiple physical models (i.e. multi-physics) and consider a larger range of scales (i.e. multi-scale) creates a steadily growing demand for interoperable infrastructures.

Interoperating infrastructures enable new innovative types of e-Science, for instance, by jointly using HPC and HTC resources of different infrastructures in one scientific workflow.




Apart from the life science community, UNICORE is also used in other application communities that use interoperable e-Science infrastructures such as DEISA, TeraGrid, EGI or OSG.

Some examples are specific e-Science communities that work towards the Virtual Physiological Human (VPH) or future electricity-producing fusion power plants.

## Contact

Email: unicore-interop@unicore.eu

UNICORE

# Recent Developments

## High-performance file transfer

Up to now a convenient, fully integrated, high-performance file transfer tool was not available in UNICORE. All available solutions have certain drawbacks, either in terms of performance, or in terms of ease of use and ease of deployment. To overcome these issues, a new tool called UFTP is being integrated into UNICORE. Similar to GridFTP, UFTP uses multiple TCP connections for a more efficient data transfer. However, in contrast to GridFTP, the UFTP solution requires neither open port ranges on the firewall, nor does it rely on proxy certificates for security. Being implemented purely in Java, UFTP is suitable for both server use and seamless integration into the UNICORE client.

## Metadata management

In order to make user data more accessible, an urgent need of services for organizing, indexing, and searching scientific data has to be satisfied. UNICORE is being extended with a metadata service, which allows to index, search, create and modify metadata. This service is de-centralized, and stores metadata in a schema-free way as key-value pairs. It is using Apache Lucene as underlying indexing and search engine. The usage of the extensible metadata extractor framework Apache Tika which is integrated into the UNICORE metadata crawler allows for quick and easy processing of the scientific data.

## Distributed storage service

In order to cope with the ever-increasing amount of scientific data, first and foremost a scalable way of storing data must be provided. For this purpose a new distributed storage facility, called Distributed Storage Management Service (dSMS) is being introduced into the UNICORE architecture. The subsystem allows end-users to have a single access point for the whole Grid storage, without the necessity


UNICORE

www.unicore.eu