Recent Advances in the UNICORE 6 Middleware

The UNICORE Grid system provides a seamless, secure and intuitive access to distributed computational and data resources such as supercomputers, clusters, and large server farms. UNICORE serves as a solid basis in many European and international research projects that use existing UNICORE components to implement advanced features, higher-level services, and support for scientific and business applications from a growing range of domains. Since its initial release in August 2006, the current version UNICORE 6 has been significantly enhanced with new components, features and standards, offering a wide range of functionality to its users from science and industry. After giving a brief overview of UNICORE 6, this article introduces some of these new features and components.

Overview of UNICORE 6

UNICORE 6 is built on industry-standard technologies, using open, XML-based solutions for secure inter-component communication. UNICORE 6 is designed according to the principles of Service Oriented Architectures (SOAs). It uses SOAP-based stateful Web services, as standardized by the standards consortia W3C, OASIS and the Open Grid Forum (OGF). The security architecture is based on X.509 certificates, the Security Assertion Markup Language (SAML) and the eXtensible Access Control Markup Language (XACML). UNICORE 6 is implemented in the platform-neutral languages Java and Perl, and UNICORE 6 services successfully run on Linux/Unix, Windows and MacOS, requiring only a Java 5 (or later) runtime environment.

UNICORE 6 Architecture

A bird's eye view of the UNICORE 6 system is given in Figure 1. The system can be decomposed into three layers: the client, services and system layers. On the client layer, a variety of client tools are available to the users, ranging from the graphical UNICORE Rich client (URC), the UNICORE command-line client (UCC) to an easy-to-use high-level programming API (HiLA). Custom clients such as portal solutions can be easily implemented.

The middle layer comprises the UNICORE 6 services. Figure 1 shows three sets of services, the left and right one containing services at a single site while the middle shows the central services, such as the central registry, the work-flow services and the information service, which serve all sites and users in a UNICORE Grid. The Gateway component acts as the entry point to a UNICORE site and performs the authentication of all incoming requests. It acts as a HTTP and Web services router, forwarding client requests to the target service, realizing a secure firewall tunnel. Therefore, in a typical UNICORE installation, only a single open firewall port is necessary.

The central UNICORE server component is based on UNICORE 6's Web services hosting environment. The XNJS component is the job management and execution engine of UNICORE 6. It performs the job incarnation, namely the mapping of the abstract job description to the concrete job description for a specific compute resource according to the rules stored in the so-called IDB (Incarnation Database). The functionality of the XNJS is accessible via two sets of Web service interfaces. The first set of interfaces is called UAS (UNICORE Atomic Services) and offers the full functionality to higher level services, clients and users. In addition to the UAS, a second set of interfaces based on open standards defined by the Open Grid Forum is available (depicted as "OGSA-*"). Some of the OGSA-* services are still under development or await official ratification, and the UNICORE community actively contributes to these discussions and provides reference implementations for emerging standards. The service layer includes a flexible and extensible security infrastructure, that allows interfacing to and integrating a variety of security components. Instead of the default UNICORE user database, Virtual Organization (VO) management systems can be accessed for getting user attributes such as role, project membership or local login. To enable interoperability scenarios, proxy certificates that are common in other Grid software are optionally supported.

On the lowest layer the TSI (Target System Interface) component is the interface between UNICORE and the resource management system and operating system of a Grid resource. In the TSI component the abstracted commands from the Grid layer are translated to system-specific commands, e.g. in the case of job submission, the specific commands like *llsubmit* or *qsub* of the resource management system are called. The TSI component is perform-
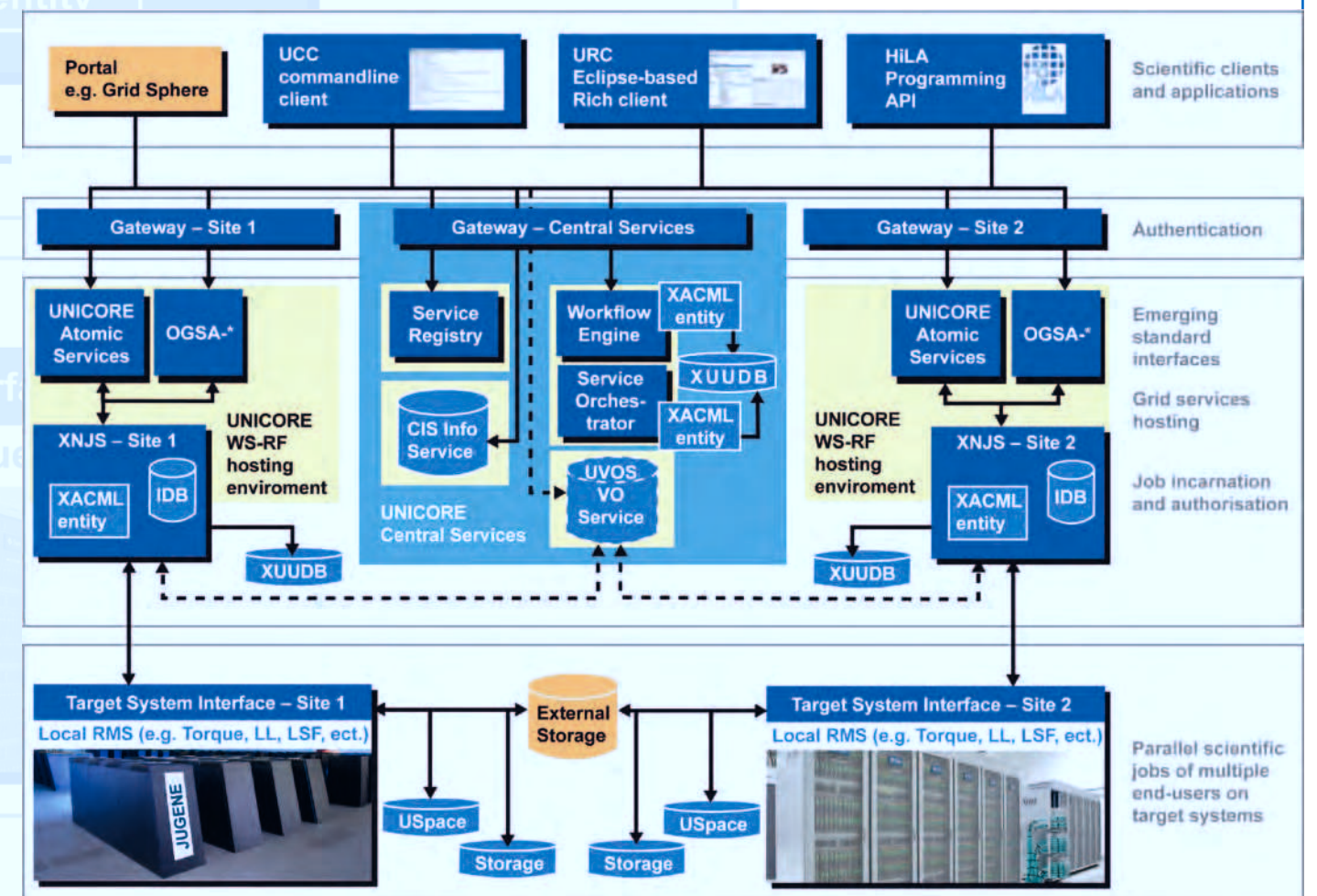

Figure 1: Overview of the UNICORE 6 system

Projects

ing tasks under the users UID and is the only UNICORE component that needs to be executed with root privileges. The TSI is available for a variety of commonly used resource management systems such as Torque, LoadLeveler, LSF, SLURM, and Sun GridEngine. In addition to job submission and job management services, UNICORE 6 provides unified access to storages via its Storage-Management and FileTransfer service interfaces. File systems are accessed directly via the TSI, while other storages can be integrated using custom adaptors. For example, an adaptor to the Apache Hadoop cluster storage system has been realized.

Recent Development Highlights

Workflow System
The UNICORE 6 workflow system offers powerful workflow features such as sequences, branches, loops and conditions, while integrating seamlessly into UNICORE Grids and into the UNICORE clients. The workflow system supports automated resource selection and brokering, and is scalable due to its two-layered architecture. Recently a for-each loop construct has been added that allows to process large file sets or perform parameter studies in a scalable and very user-friendly fashion. Figure 2 shows a screen shot of the URC, with the workflow editor shown on the right-hand side.

Improved Data and Storage Management
UNICORE's data and storage management capabilities have been integrated into the UNICORE Rich Client in an easy to use fashion. The URC allows common operations like editing, deleting or renaming, as well as drag and drop of files between the remote sites and the user's desktop. A new StorageFactory service allows to create and manage remote storage resources.

Execution Environments: Improved Support for Parallel Applications
Users running parallel applications are usually required to know about the invocation details of the parallel environment (such as OpenMP or MPI). A recently added UNICORE feature allows administrators to easily parametrize the available parallel environments in the UNICORE IDB, so that users can easily choose the required environment and set options and arguments, without being required to know the system specifics.

Remote Administration
It is often convenient to check site health, get up-to-date performance metrics and even deploy and un-deploy services remotely using the standard communication channels (i.e. Web services), without having to rely on other tools such as Nagios, or needing to actually log in to the UNICORE 6 server. For this reason, an Admin service has been developed that offers various remote administration and monitoring features. It can be used through a graphical URC plugin and through the commandline client. Using the UNICORE XACML based access control, access to this service is of course limited to privileged users.

Shibboleth Integration in the UNICORE Rich Client
Shibboleth is a standards-based identity management system aiming to provide single sign-on across organisational boundaries. It is considered a prime candidate for simplifying end user experience by hiding the specifics of the X.509 certificate based security. Therefore a plugin for the URC was developed that allows to access Shibboleth-enabled attribute authorities, which create short-lived certificates that enable access to the Grid resources.

Interactive Access to Grid Sites
Using the single sign-on feature of the UNICORE Rich Client, it is desirable to provide an interactive terminal access to the remote site. A URC plugin has been developed that can be used to open an interactive connection using a variety of means such as standard SSH using username and password, GSI – SSH and a custom X.509 based solution. The client-side plugin offers VT100 emulation and port forwarding.

Open Source Model
UNICORE 6 is available as open source under BSD license from SourceForge. At **http://www.unicore.eu** more information can be obtained and lightweight UNICORE 6 installation packages can be downloaded. A small UNICORE 6 testgrid is available for immediate testing. UNICORE 6 is continually evolved, with regular releases approximately every three months. The main contributors in the international UNICORE open-source community are: ICM Warsaw, CINECA, CEA, Technische Universität Dresden, and Forschungszentrum Jülich.
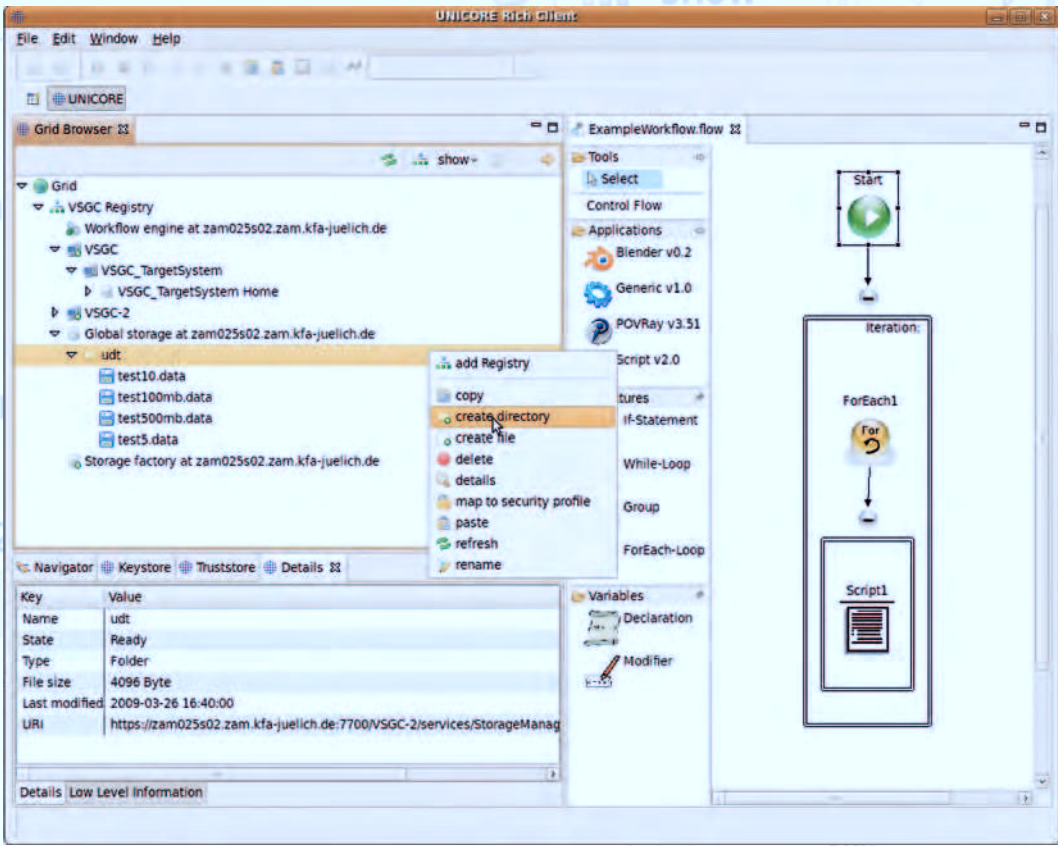

Figure 2: Screen shot of the UNICORE Rich Client

• Bernd Schuller
• Morris Riedel
• Achim Streit

Jülich
Supercomputing
Centre,
Forschungs-
zentrum Jülich