UNICORE 6
A European Grid Technology

http://www.unicore.eu

Achim Streit
a.streit@fz-juelich.de

Jülich Supercomputing Centre (JSC)
History Lesson

- **UNiform Interface to COmputing Resources**
  - seamless, secure, and intuitive

- Initial development started in two German projects funded by the German ministry of education and research (BMBF)
  - 08/1997 – 12/1999: UNICORE project
    - Results: well defined security architecture with X.509 certificates, intuitive GUI, central job supervisor based on Codine from Genias
    - Results: implementation enhancements (e.g. replacement of Codine by custom NJS), extended job control (workflows), application specific interfaces (plugins)

- Continuous development since 2002 in several EU projects
- Open Source community development since Summer 2004

http://www.unicore.eu
Guiding Principles, Implementation Strategies

- **Open source** under BSD license with software hosted on SourceForge
- **Standards-based**: OGSA-conform, WS-RF 1.2 compliant
- Open, extensible **Service-Oriented Architecture** (SOA)
- **Interoperable** with other Grid technologies
- Seamless, secure and intuitive following a vertical end-to-end approach
- Mature **Security**: X.509, proxy and VO support
- **Workflow** support tightly integrated while being extensible for different workflow languages and engines for domain-specific usage
- **Application integration** mechanisms on the client, services and resource level
- Variety of **clients**: graphical, command-line, API, portal, etc.
- Quick and **simple installation** and configuration
- Support for many operating systems (Windows, MacOS, Linux, UNIX) and batch systems (LoadLeveler, Torque, SLURM, LSF, OpenCCS)
- Implemented in **Java** to achieve platform-independence

http://www.unicore.eu
Gateway

Gateway – Site 1

- UNICORE Atomic Services
- OGSA-*
- XNJS – Site 1
  - XACML entity
  - XUUDB

Gateway – Site 2

- UNICORE Atomic Services
- OGSA-*
- XNJS – Site 2
  - XACML entity
  - XUUDB

Target System Interface – Site 1

- Local RMS (e.g. Torque, LL, LSF, etc.)
- External Storage
- USpace

Target System Interface – Site 2

- Local RMS (e.g. Torque, LL, LSF, etc.)
- External Storage
- USpace

Web service stack

- X.509, Proxies, SOAP, WS-RF, WS-I, JSDL
- OGSA-RUS, UR, GLUE 2.0
- OGSA-ByteIO, OGSA-BES, JSDL, HPC-P, OGSA-RUS, UR

Authentication

- X.509, XACML, SAML, Proxies
- DRMAA
- GridFTP, Proxies

Data transfer to external storages

Scientific clients and applications

- scientific clients and applications
- web service stack
- central services running in WS-RF hosting environments
- OGSA-RUS, UR
- GLUE 2.0
- OGSA-ByteIO, OGSA-BES, JSDL
- HPC-P
- OGSA-RUS, UR

Deployment environments

- UNICORE
- OGSA-*
- XNJS
- IDB
- XUUEDB

Grid services hosting

- job incarnation
- X.509, XACML, SAML, Proxies

Authorization

- DRMAA
- GridFTP, Proxies

http://www.unicore.eu
Standards in **UNICORE 6**

- Web-Services (**WS-RF 1.2, SOAP, WS-I**) stack
- Security & authorization
  - Full **X.509** certificates as base line, **XACML** based access control
  - Support for **SAML**-based VOMS & **X.509** proxies
- Job management
  - **OGSA-BES, HPC-P**: creation, monitoring and control of jobs
  - Job definition compliant with **JSDL** (+ JSDL HPC ext.)
  - **DRMAA** communication to local resource manager for job scheduling
- Information system
  - **GLUE 2.0** information service (interaction with the GLUE WG)
- Accounting
  - **OGSA-RUS** for accounting (incl. **UR** for storing)
- Access to data storage/management and transfer
  - Fully **OGSA-ByteIO** compliant for site-to-site transfers
  - **GridFTP** support as third-party transfer under development
  - **SRM** support under development

[Link to UNICORE website](http://www.unicore.eu)
Workflows in **UNICORE 6**

- Two layer architecture for scalability
- Workflow engine
  - Based on Shark open-source XPDL engine
  - Pluggable, domain-specific workflow languages
- Service orchestrator
  - Job execution and monitoring
  - Callback to workflow engine
  - Brokering based on pluggable strategies
- Clients
  - GUI client based on **Eclipse**
  - Commandline submission of workflows is also possible

http://www.unicore.eu
Eclipse-based Rich Client (URC)

http://www.unicore.eu
UCC – Commandline Client

> ucc -h
UCC version 1.2-SNAPSHOT
Usage: ucc <command> [OPTIONS] <args>
The following commands are available:

Data management:
- ls         - list a storage
- copy-file-status - check status of a copy-file
- get-file   - get remote files
- find       - find files on storages
- resolve    - resolve remote location
- copy-file - copy remote files
- csm-get-file - get remote files
- put-file   - puts a local file to a remote server

General:
- connect   - connect to UNICORE
- list-applications - lists applications on target systems
- list-jobs  - list your jobs
- list-sites - list remote sites
- csm-system-info - Checks the availability of services.

Job execution:
- run        - run a job through UNICORE &
- get-status - get job status
- abort-job  - abort a job
- batch      - run ucc on a set of files
- get-output - get output files

Other:
- shell      - Starts an interactive UCC session
- loadtest   - load tests services
- issue-delegation - Allows to issue a trust delegation assertion
- warf       - perform a WSRF operation
- run-groovy - run a Groovy script

Workflow:
- csm-submit - submit a workflow to Chemomentum
- csm-trace  - trace info on a workflow in Chemomentum
- csm-control - control a workflow in Chemomentum
- csm-workflow-info - lists info on workflows in Chemomentum

Enter 'ucc <command> -h' for help on a particular command.
HiLA – High Level API for Grid Applications

- Simple development of clients
  - Few lines for otherwise complex functionality
- Single interface, multiple implementations
  - UNICORE 5 & 6, OGSA-BES
- URI Scheme naming resources of the Grid
  - `unicore6:/sites/FZJ_JUGGLE/storages/home`
  - `ogsa:/sites/GROW/tasks/910c9b56-d497-46f8-960f-eaee43e1af37`
  - Object navigation based on ‘container/item’ model
- Security, sites, registries, etc. via configuration

```java
Location l = new Location("unicore6:/sites/GROW/tasks/910c9b56-d497-46f8-960f-eaee43e1af37");
Task t = HiLAFactory.getInstance().locate(l);
assertTrue(TaskStatus.RUNNING, t.status());
List<File> fl = t.getOutcomeFiles();
```
Application Integration in UNICORE 6

- GridBean concept allows application-specific “enhancements” of the Eclipse-based UNICORE Rich Client (URC)
  - Domain-/application-centric specification of input parameters
  - Output of application can be visualized, e.g. as simple text or picture as well as complex 3d-rotatable molecule
  - Common for all GridBeans: file stage-in/-out, resource requirements, variables
- Easily implementable through a powerful library and good documentation
- Available GridBeans
  - General: Generic, Script (+ Povray/Blender)
  - Life science: AMBER, MDA, MDC, Molgeo, MOPAC, PAP, GAMESS

http://www.unicore.eu
Generic GridBean

- For applications with command line parameters only, e.g.
  - Perl
  - Python
  - Shells
  - Java
  - Any application

http://www.unicore.eu
Script GridBean

- With text editor to directly type scripts
  - Perl
  - Python
  - Shells

http://www.unicore.eu
AMBER GridBean

http://www.unicore.eu
Life Science Workflow
UNICORE in use – some examples

- Supercomputing
  - DEISA
  - Clinical Supercomputing (→ more details at 14:35)
  - SKIF-GRID
  - CEA (→ more details at 16:55)

- National Grids
  - D-Grid
  - AeroGrid
  - BIS-Grid

- Commercial
  - T-Systems SfR (→ more details at 17:30)
  - 52° North

http://www.unicore.eu
Supercomputing

Distributed European Infrastructure for Supercomputing Applications

- Consortium of leading national HPC centers
- Deploy and operate a persistent, pan-European, heterogeneous HPC environment
- UNICORE as Grid Middleware
  - On top of DEISA’s core services:
    - Dedicated network
    - Shared file system
    - Common production environment at all sites
  - Used e.g. for workflow applications

SKIF-GRID federation

- Joint Russian-Belarus project
- Federation of 8 HPC centers
  - UNICORE middleware
  - 3 computers in the current Jun’08 Top 500
  - ~100 TFlops peak
  - Research program in HPC services

http://www.unicore.eu
National Grids

**UNICORE** Usage in D-Grid

- Core D-Grid sites committing parts of their existing resources
  - Approx. 700 CPU
  - Approx. 1 PByte
  - UNICORE is installed
- Additional Sites receiving money from the BMWF and installing compute clusters and tools
  - Approx. 2000 CPUs
  - Approx. 2 PBytes
  - UNICORE (as well as Eureka and gLIFE) is installed as systems are adopted

**Collaboration within AeroGrid**

**BIS-Grid: Grid-based Information Systems Integration**

- **Workflow Engine for Grid Service Orchestration**
  - Executes workflows that orchestrate Web services as well as WSRF Grid Services
  - Provides workflows as WSRF Grid Services
  - Supports Grid and IS Security (RBAC, Certificates, Access Policies)
  - Provides simple monitoring functions

  - Realized as UNICORE 6 Services
  - Uses WS-BPEL and an arbitrary WS-BPEL engine for orchestration
  - WS-BPEL Engine is exchangeable
  - No proprietary WS-BPEL
    - language extensions or modifications
    - engine extensions or modifications
  - Open Source
  - More Information and outcomes on [www.bisgrid.de](http://www.bisgrid.de)

*Slide courtesy of André Höing (TU Berlin)*

[http://www.unicore.eu](http://www.unicore.eu)
Commercial
software, source code, documentation, tutorials, mailing lists, community links, and more:

http://www.unicore.eu