UNICORE -
a technical introduction
Overview

- What is UNICORE?
- User Interfaces
- Job Management
- Data Management
- Administrator View
- Support
- Community

www.unicore.eu
What is UNICORE?

• **UNiform Interface to COmputing Resources**
  - PCs, server
  - Clusters
  - Parallel computers
  - Data storages

• **Goals**
  - Seamless
  - Secure
  - Intuitive
What can you do with UNICORE?

• Submit jobs to the Grid
  – Single scripts, applications …
  – Workflows of interdependent applications
  – Large numbers of applications - High Throughput Computing
  – Parallel programs - High Performance Computing

• Manage data in the Grid
  – Store and access files
  – Create, store and access metadata
  – Transfer data between sites and between jobs
Simple Example

- User wants to run a script (anywhere).
- How to get access?
  - UNICORE Clients
    1. UNICORE Rich Client **URC** (Graphical Client)
    2. UNICORE Commandline Client **UCC**
    3. High Level API **HiLA**
    4. Portal Client (first release autumn 2013)
UNICORE Rich Client (URC)

- Graphical User Interface
- Building, submitting and monitoring jobs
- Workflow editor and tracer
- Integrated data and storage management
  - Drag and drop from/to desktop environment
  - Copy and paste
  - Remote file editing

www.unicore.eu
UNICORE Rich Client (URC)

- Pluggable credentials
- “Simple view” for novice users
- Based on Eclipse framework
- Extensibility through plug-ins
  - Examples: Shibboleth, CIS view, Admin Interfaces
    (user and virtual organization management, dynamically deploy/undeploy services)
- Installation/update mechanism for plug-ins and GridBeans
- For Windows, Linux/Unix and MacOS/X

www.unicore.eu
URC – Application Integration

• Application-specific interfaces with GridBeans
  – 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 API
  – Developer's Guide:

• Available GridBeans
  – General: Generic, Script (+ Povray/Blender)
  – Scientific applications: AMBER, MDA, MDC, Molgeo, MOPAC, PAP, GAMESS, NAMD, Clustal, BLAST and more life science Gridbeans at http://unicore-life.sourceforge.net/
URC – Application Integration

- Molecular Orbital PACkage
  - a semi-empirical quantum chemistry program
UNICORE Comandline Client (UCC)

- The “Swiss Army Knife” of UNICORE
- Job and workflow handling
- Data management
- Find operation
- Shell/Batch mode for high throughput job processing
- Generation and usage of SAML trust delegation assertions
- Supports multiple registries
- Remote administration of UNICORE services
- Scriptable, easily extensible
High Level API (HiLA)

• Integrate UNICORE into your application or working environment
• Simple development of clients
• Few lines for otherwise complex functionality
• URI Scheme naming resources of the Grid
  – unicore6:/sites/FZJ_JUROPA/storages/home
  – ogsa:/sites/GROW/tasks/910c9b56-d497-46f8-960f-eaee43e1af37
  – Object navigation based on ‘container/item’ model
• Security, sites, registries, etc. via configuration file
• Single interface, multiple implementations
  – UNICORE, OGSA-BES
• Used by third parties to integrate UNICORE with their tools and programming environments
  – JavaGAT
  – EMI-ES Java Client
Portal Client

- Will be released in autumn 2013
- Secure web based interface to access Grid resources
- Easy launching and monitoring of jobs and workflow including data management
Back to the Simple Example

Simple click to submit the job

Create script
What happens? - A view at the architecture

Clients

Single Entry Point, Authentication

Standard interfaces for job management

Grid services hosting

Job incarnation & authorisation

(perallel) jobs of multiple end-users on target systems

Gateway

UNICORE Atomic Services
OGSA-*

XNJS

XACML entity

XUUDB

Target System Interface

Local RMS (e.g. Torque, LL, LSF, etc.)

UNICORE WS-RF hosting environment

Portal e.g. GridSphere

UCC command-line client

URC Eclipse-based Rich client

HILA Programming API

www.unicore.eu
Target System Interface (TSI)

- Interoperation with OS, local resource management system, and file system
- Perl daemon running on the frontend of the target resource
- Translates abstracted commands from the Grid to system-specific commands (job and data management)
- Available for a variety of commonly used resource management systems
  - Torque, LoadLeveler, LSF, SLURM, OpenCCS …
  - For OGF-DRMAA standard
Security Architecture and Guiding Principles

• Guiding principles:
  – Use open industry standards:
    X.509 public key infrastructure, TLS, SAML, XACML
  – Control flow encryption
  – User permissions only for services
• Several user authentication options:
  – X.509 certificate
  – Username/password (autumn 2013)
  – LDAP, Kerberos, DFN-AAI (autumn 2013)
• Many authorization options:
  – User attribute sources
    ▪ XUUDB – mapping of user certificate to local identity
    ▪ Mapfile
    ▪ UVOS/UNITY – UNICORE VO Service, Grid user management
  – Authorization decision through local XACML policies
More complex example - workflows

• User wants to do a parameter study
  – Single executable with different input data files
  – For-Each Loop

• User has interdependent programs to be executed
  – A program produces an output file which is input to two other executables
  – Depending on the results of a program (e.g. a value or an output file or the exit status) other programs are executed taking the output as input
  – If-Then-Else

• … and much more
Workflows – A view at the architecture

- Portal e.g. GridSphere
- UCC command-line client
- URC Eclipse-based Rich client
- HiLA Programming API

- UNICORE Atomic Services
- OGSA-
- XNJS – Site 1
- UNICORE WS-RF hosting environment
- XACML entity
- Gateway – Site 1
- Service Registry
- CIS Info Service
- Workflow Engine
- Service Orchestrator
- XUUDB
- Gateway
- UNICORE Atomic Services
- OGSA-
- XNJS – Site 2
- UNICORE WS-RF hosting environment
- XACML entity
- Gateway – Site 2

- Target System Interface – Site 1
  - Local RMS (e.g. Torque, LL, LSF, etc.)
- External Storage
- USpace
- Storage
- Target System Interface – Site 2
  - Local RMS (e.g. Torque, LL, LSF, etc.)
- USpace
- Storage

- Scientific clients and applications
- Authentication
- Standard interfaces
- Grid services hosting
- Job incarnation & authorisation

(parallel) jobs of multiple end-users on target systems
Workflows

• Workflow engine
  – Pluggable, domain-specific workflow languages

• Service orchestrator
  – Resource brokering based on pluggable strategies
  – Low-level Grid job execution and monitoring
  – Multiple instances can be deployed for scalability

• Tracing Service
  – Logs workflow events with timestamps
  – Useful for performance evaluations, monitoring and reporting

• Control Constructs:
  ForEach-Loop, While-Loop, If-Statement, Grouping …

• Conditions:
  exit code, file existence, file size, workflow variables
Iterate over files or variables.

Data files can be local or remote.

Create complex workflows graphically, e.g. for-each-loop:

- Iterate over file names or variable values.
- Data files can be local or remote.
Data Management

• Storage Management Service (SMS) provides an abstract file system-like view on a storage resource
  – Common operations (e.g. mkdir, delete, listDirectory)
  – Various back ends available (e.g. file system, Apache Hadoop, iRODS, DataFinder, SRM, LFC)
  – designed to be easily extensible
Data Management

• Distributed Storage Management dSMS
  – Unites distributed SMS to a single system
  – File catalogue

• Metadata
  – Per SMS
  – Key-value-pairs
  – Automatic metadata creation and indexing
    (Apache Tika and Lucene)
File Transfer

• Import/export files from/to clients/servers/jobs
• Built-in
  – UFTP – secure parallel file transfer, only single open port needed, high performance
  – BFT transfer - based on HTTPs, single open port needed, simple interface (bulk write, read supports byte ranges), several MB/sec
  – OGSA ByteIo - uses SOAP messages, single open port needed, rich interface (POSIX-like, block read/write, etc), ~400kB/sec
• Plus: alternative mechanisms can be plugged in
  – UDT, GridFTP, parallel HTTP ...
Standards are Used Where Ever Possible

- Web-Services stack: WS-RF 1.2, SOAP, WS-I
- Security
  - Full X.509 certificates as base line, support for X.509 proxies
  - SAML authorization (delegation), XACML based access control
- Job management
  - Job definition compliant with JSDL (+ JSDL HPC ext.)
  - OGSA-BES, HPC-BP: creation, monitoring and control of jobs
- Data transfers
  - HTTPS and OGSA-ByteIO
  - GridFTP
Administrators View on UNICORE

• (Almost) all components Java based → portable
• tar/rpm/deb packages available
• Part of EMI and EGI’s UMD middleware distribution
• Administration Service (access controlled!)
  – Service deployment/undeployment
  – Publish operational metrics (Java VM, service performance)
Administrators View on UNICORE

- Check functionality of full UNICORE installation by sending a single job (simply testing the availability is not enough)
- Reports/errors are published by email, text message (SMS), etc.
- Connection (“Reporters”) to INCA and Nagios available, other are possible

Current Network Status
Last Updated: Tue May 4 16:07:21 CEST 2009
Updated every 60 seconds
Logfile: /var/log/nagios.log
Logged in: no administrators

View History For All Hosts
View Nagios For All Hosts
View Host Status For All Hosts

Service Status Details For All Hosts

<table>
<thead>
<tr>
<th>Service</th>
<th>Status</th>
<th>Last Check</th>
<th>Duration</th>
<th>Attempt</th>
<th>Status Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>BookList</td>
<td>OK</td>
<td>05-06-2009 16:55:37</td>
<td>5h 49m</td>
<td>1/1</td>
<td>OK - load average: 0.86, 0.48, 0.32</td>
</tr>
<tr>
<td>Current Load</td>
<td>OK</td>
<td>05-06-2009 16:55:37</td>
<td>5h 49m</td>
<td>1/1</td>
<td>OK - load average: 0.86, 0.48, 0.32</td>
</tr>
<tr>
<td>Current Users</td>
<td>OK</td>
<td>05-06-2009 16:55:37</td>
<td>5h 49m</td>
<td>1/1</td>
<td>USERS OK - 1 users currently logged in</td>
</tr>
<tr>
<td>HTTP</td>
<td>WARNING</td>
<td>05-06-2009 16:55:37</td>
<td>5h 49m</td>
<td>1/1</td>
<td>HTTP WARNING HTTP/1.1 403 Forbidden</td>
</tr>
<tr>
<td>DNS</td>
<td>OK</td>
<td>05-06-2009 16:55:37</td>
<td>5h 49m</td>
<td>1/1</td>
<td>DNS OK - Packet loss = 0%, RTT = 0.03 ms</td>
</tr>
<tr>
<td>Swap Usage</td>
<td>OK</td>
<td>05-06-2009 16:55:37</td>
<td>5h 49m</td>
<td>1/1</td>
<td>SWAP OK - 41% free (627 MB out of 2055 MB)</td>
</tr>
<tr>
<td>Total Processes</td>
<td>OK</td>
<td>05-06-2009 16:55:37</td>
<td>5h 49m</td>
<td>1/1</td>
<td>PROCES OK - 33 processes with STATE = RSZDT</td>
</tr>
<tr>
<td><a href="http://www.unicore.eu">www.unicore.eu</a></td>
<td>9 Matching Service Entries Displayed</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
UNICORE Support

• Web Resources
  – www.unicore.eu
  – Documentation as pdf and HTML
  – Wiki

• Mailing list
  – Unicore-support@lists.sourceforge.net
UNICORE Community

• UNICORE is developed by an international Open Source community
  – Main actors
    ▪ Forschungszentrum Jülich
    ▪ ICM – Uniwersytet Warszawski
    ▪ Technische Universität Dresden – ZIH
    ▪ CINECA – Consorzio Interuniversitario
  – http://sourceforge.net/projects/unicore/
UNICORE Community

- Supported by the UNICORE Forum e.V., http://www.unicore.eu/forum
  - Founded by developers, leading European HPC centers, universities and supporting hardware vendors as a non-profit association
  - To foster and support the distribution and use, coordinate the development