

# UNICORE

## UNICORE 6 Clients

Bastian Demuth, FZ Jülich  
[b.demuth@fz-juelich.de](mailto:b.demuth@fz-juelich.de)

UNICORE Migrationsworkshop  
29. Oktober, 2008, Langen

# Design Principles

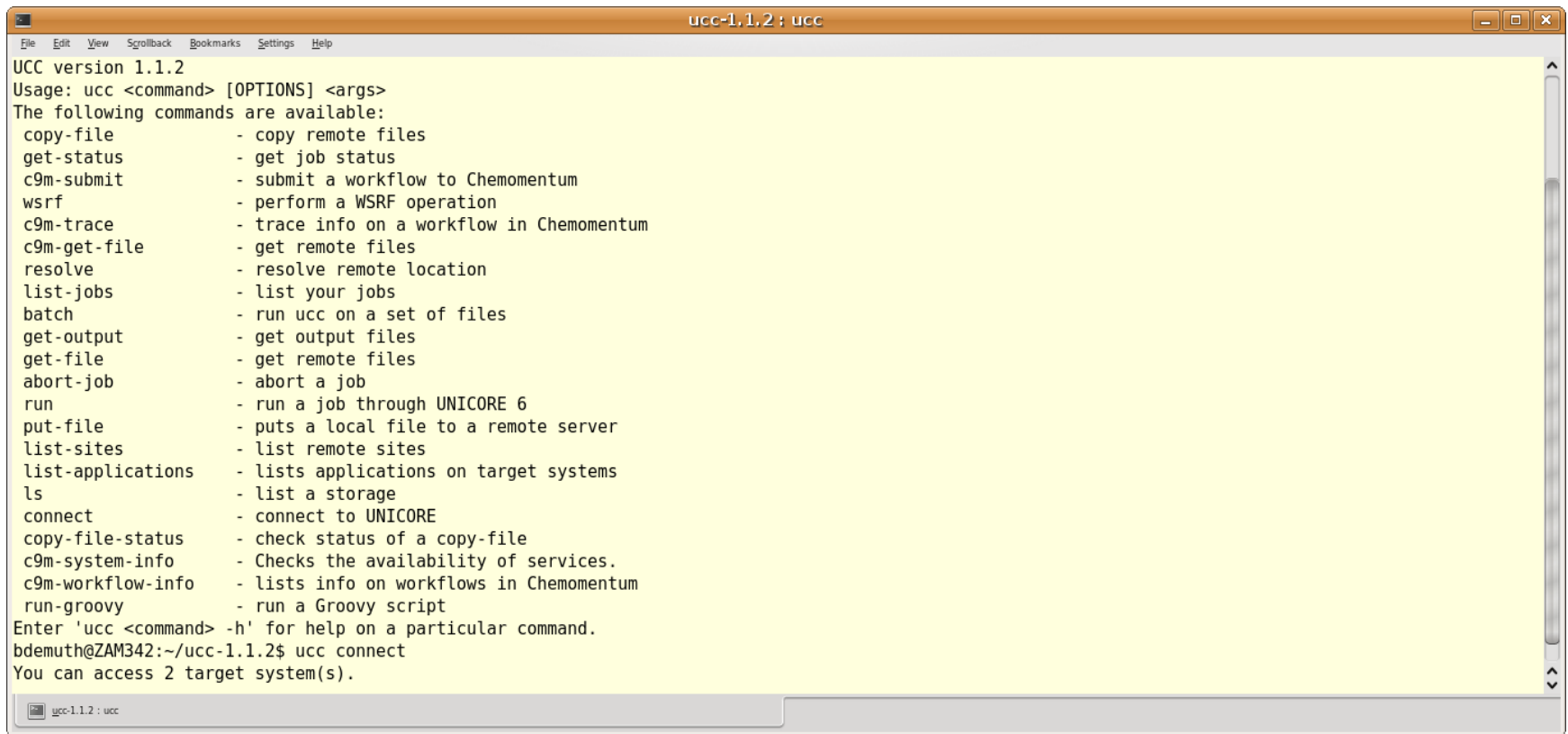
- ▶ Target: wide range of users with varying expertise
- ▶ Intuitive yet powerful
- ▶ Both graphical and commandline clients
- ▶ Extensible and maintainable
- ▶ Ease of application integration
- ▶ Many Supported Operating Systems:
  - ▶ UNIXes, Windows 2000/XP, (MacOS X)

# Different Approaches

- ▶ UNICORE commandline client (UCC)
    - ▶ Execution of jobs and workflows, file movements, administrative tasks
    - ▶ Java based, extensible through Groovy scripts
  - ▶ High Level API (HILA)
  - ▶ Application client
    - ▶ Single application run
    - ▶ Lightweight & easy to use
  - ▶ Eclipse-based rich client
    - ▶ Workflow editing & monitoring
    - ▶ File transfers via drag & drop
    - ▶ Administrative extensions
  - ▶ Web portal client
- Use Grid Programming Environment (GPE)  
=> extensible through GridBeans (application GUIs)

# UNICORE Commandline Client (UCC)

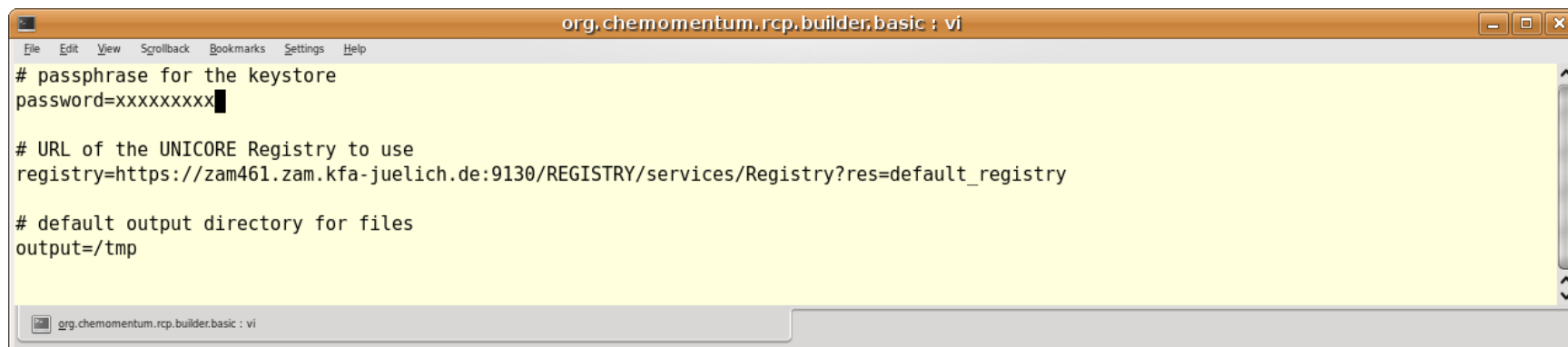
- ▶ Run jobs, get status, get output, transfer files
- ▶ Administrative use : list jobs, clean up, etc.
- ▶ Extensible: add commands



```
ucc-1.1.2 : ucc
File Edit View Scrollback Bookmarks Settings Help
UCC version 1.1.2
Usage: ucc <command> [OPTIONS] <args>
The following commands are available:
copy-file      - copy remote files
get-status     - get job status
c9m-submit     - submit a workflow to Chemomentum
wsrf           - perform a WSRF operation
c9m-trace      - trace info on a workflow in Chemomentum
c9m-get-file   - get remote files
resolve        - resolve remote location
list-jobs      - list your jobs
batch          - run ucc on a set of files
get-output     - get output files
get-file       - get remote files
abort-job      - abort a job
run            - run a job through UNICORE 6
put-file       - puts a local file to a remote server
list-sites     - list remote sites
list-applications - lists applications on target systems
ls             - list a storage
connect        - connect to UNICORE
copy-file-status - check status of a copy-file
c9m-system-info - Checks the availability of services.
c9m-workflow-info - lists info on workflows in Chemomentum
run-groovy     - run a Groovy script
Enter 'ucc <command> -h' for help on a particular command.
bdemuth@ZAM342:~/ucc-1.1.2$ ucc connect
You can access 2 target system(s).
```

# UCC: Preferences

- ▶ Set keystore and passphrase
- ▶ Enter Registry URL
- ▶ Select default output directory for files
- ▶ For convenience: options (e.g. --verbose, --recursive)



The screenshot shows a text editor window titled "org.chemomentum.rcp.builder.basic : vi". The editor contains the following configuration text:

```
# passphrase for the keystore
password=xxxxxxxx█

# URL of the UNICORE Registry to use
registry=https://zam461.zam.kfa-juelich.de:9130/REGISTRY/services/Registry?res=default_registry

# default output directory for files
output=/tmp
```

# UCC: Job definitions

```
{  
  ApplicationName: POVRay, ApplicationVersion: "3.5",  
  
  Environment: ["OUTPUT_FORMAT=PNG",  
    "SOURCE=input.pov",  
    "TARGET=output.png",  
    "WIDTH=320",  
    "HEIGHT=200" ],  
  
  Imports: [{  
    File: "scene_file",  
    To: "input.pov" }],  
  
  Exports: [{  
    File: "output.png",  
    To: "output.png"}],  
}
```

# UCC: Additional features

- ▶ Chemomentum workflow extension
  - ▶ Submission
  - ▶ Monitoring / Tracing
  - ▶ Logical filenames
  - ▶ Service check
- ▶ Batch mode
  - ▶ Submit all jobs in a designated directory
  - ▶ Follow mode
  - ▶ XMLSpaces based matchmaking
- ▶ Common Information Service (CIS) extension
  - ▶ Query
  - ▶ Add, remove & list information providers

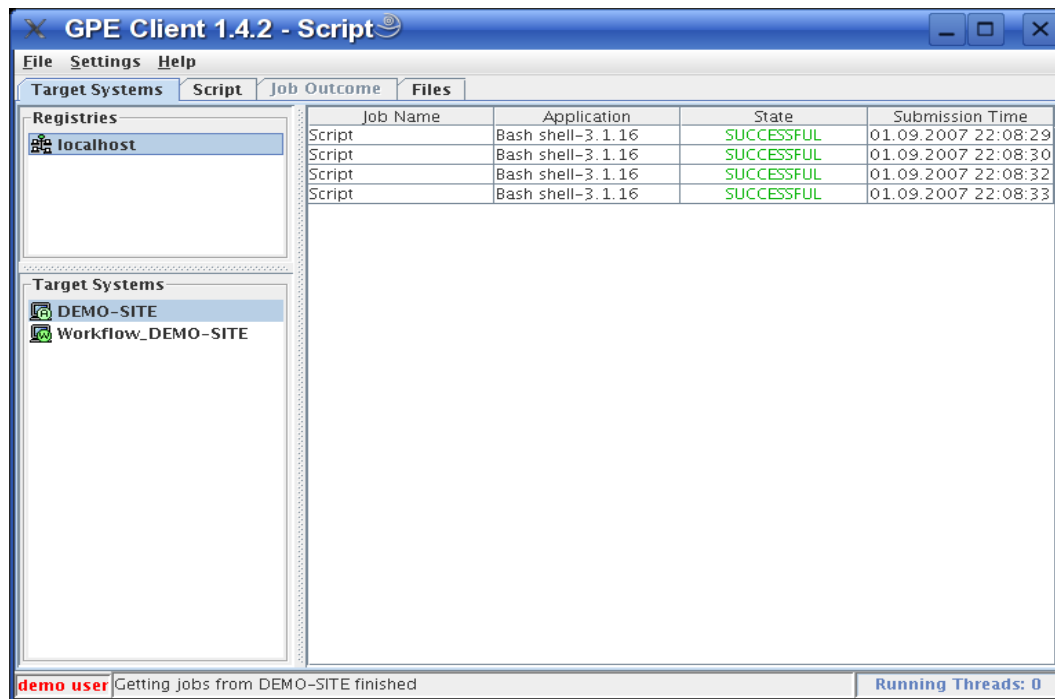
# High Level API (HiLA)

- ▶ JAVA API for programming own clients
- ▶ Very abstract
  - ▶ Easy to understand
  - ▶ Implementations for UNICORE 5 and 6
  - ▶ Access not as fine grained as with UNICORE 6 specific clients
- ▶ Model: resources interconnected by links

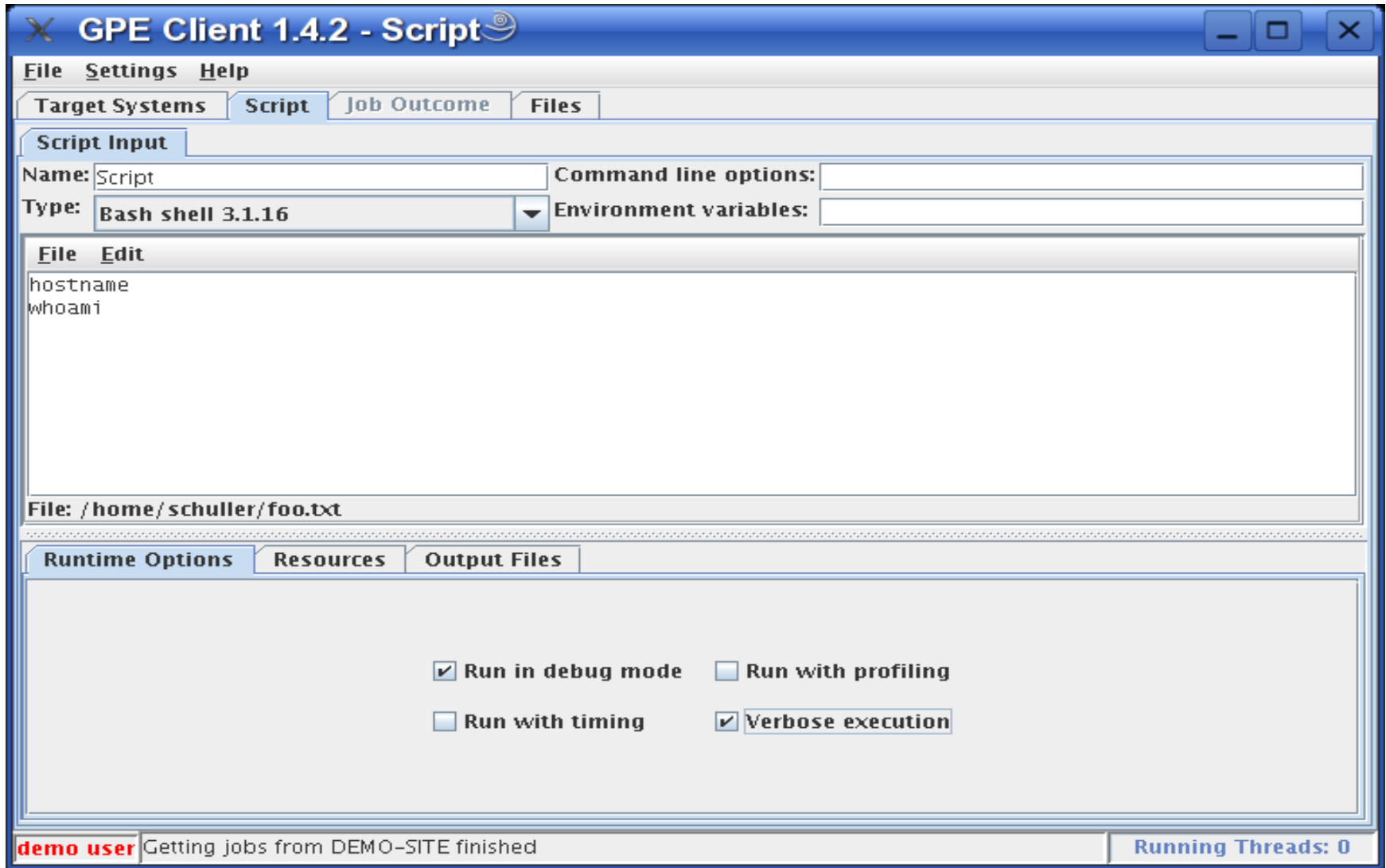
```
List<Site> sites = grid.getAllSites();  
for (Site site : sites)  
{  
    System.out.println( site + " " + site.ok() );  
}
```

# Application Client

- ▶ Supports multiple Grids/Registries at the same time
- ▶ Lists all target systems in selected Registry
- ▶ Lists all jobs at a selected target system
- ▶ Job definition and outcome visualization through GridBeans



# GridBeans: Script



# GridBeans: KoDaVis

The screenshot displays the KoDaVis interface with several components:

- Configuration Panels:** On the left, there are panels for 'Visualization Parameters' (with fields for 'Visualization nickname' and 'Isotrop'), 'Running Sessions' (with a table for session details), and 'New Session' (with fields for session name, date, start time, and duration).
- Performance Graph:** A line graph at the bottom shows 'Seconds' on the y-axis (0 to 40) and 'Timesteps' on the x-axis (10 to 30). Two lines, one red and one blue, show a sharp spike between timesteps 25 and 28, reaching approximately 30 seconds.
- Isotrop Window:** A separate window titled 'Isotrop' shows a globe visualization with a color scale from blue to red. It includes controls for 'Dataset', 'Characteristic', 'Timestep', and 'Level', along with a 'Request Data' button and a 'Disconnect from DS' button.
- Status Bar:** At the bottom, a status bar indicates 'dfn-verein id von sonja habbinga Getting jobs from Phosphorus\_FZJ finished' and 'Running Threads: 0'.

# Rich Client

The screenshot displays the UNICORE Rich Client interface. The main window is titled "UNICORE Rich Client" and features a menu bar (File, Edit, Window, Help) and a toolbar. The interface is divided into several panes:

- Workflow Engine:** A central diagram showing the workflow. It includes a "Workflow Engine" node, a "Callback Service" node, a "Service Orchestrator" node, and two "Target System" nodes. Data flows are labeled with "submittedJobForWA (5)", "finishedWA (5)", "submitWorkAssignment (5)", "finishedJob (3)", "finishedJob (2)", and "Submit (3)".
- Tools Palette:** A vertical list of tools on the right side, including "Select", "Structures", "If-Statement...", "While-Loop", "Variables", "Declaration", "Modifier", "Transitions", "Control Flow", "Applications", "POV-Ray v3.5", and "Script v2.0".
- Workflow Editor:** A large area on the right showing a detailed view of a workflow step. It starts with a "Start" node, followed by a "While" loop labeled "WhileSplit1". The loop condition is "While1\_iteration\_counter < 6". Inside the loop is a "POV-Ray1" node, which is currently active and showing a green checkmark.
- Output Panel:** At the bottom left, there is a "Filename:" field with a "Save..." button and a "Download" button. Below this is a rendered image of a blue and white striped sphere on a checkered floor. The status bar at the bottom indicates "Downloading outcome...space...".

# Rich Client: Why Eclipse?

## ▶ Pros:

- ▶ Well-known and maintained framework
- ▶ Integrates nicely with different platforms
- ▶ Proven technology
- ▶ Powerful
- ▶ Very extensible: “Everything is a plugin!”
- ▶ Our plugins can interact with other Eclipse-based tools
- ▶ Provides a sophisticated update mechanism

## ▶ Cons:

- ▶ GridBeans are based on Swing => Integration challenging
- ▶ Hard to test on all different platforms
- ▶ Big download size (~65 MB)

# Rich Client vs. UNICORE 5 Client:

## Main differences

- ▶ Based on Eclipse
  - ▶ GUI is more flexible (drag & drop of views)
  - ▶ Perspectives
  - ▶ Workspace concept (e.g. import/export workflows)
  - ▶ Less effort for developers
- ▶ More extensible
  - ▶ Grid Browser is not UNICORE specific
  - ▶ Workflow Editor supports multiple workflow engines
  - ▶ Many additional extension points (e.g. OGSA-DAI extension)
- ▶ New workflow editor
  - ▶ Shows the whole workflow graph
  - ▶ Subgraphs can be hidden
  - ▶ Allows for zooming and printing workflow Graphs

# Rich Client: Grid Browser

- ▶ Grid displayed as a tree – similar to a file browser
- ▶ Used for monitoring and performing actions on the Grid
- ▶ Predefined and configurable filters
- ▶ Details view provides information about services

The screenshot displays the Grid Browser application interface. The main window is titled "Grid Browser" and shows a tree view of the grid structure. The tree view is expanded to show the "Alpha-Site" under the "Workflow engine at omiei.zam.kfa-juelich.de". The details view for the "Alpha-Site" is shown below the tree view, displaying a table of installed applications and their versions.

Key	Value
	Povray (Version 3.7)
	JAVAC (Version 5.0)
	JAVA (Version 5.0)
	ECHO (Version 1.0)
Installed Applications	COMMAND (Version 1.0)
	Bash shell (Version 3.1.16)
	C shell (Version 6.14.00)

The filter menu is open, showing the following options:

- All Services
- Execution Services and Storages
- Workflows
- Workflow Services
- Files
- Target Systems
- Jobs
  - Userdefined Job Filter
  - All Jobs
  - Jobs with undefined status
  - Ready Jobs
  - Queued Jobs
  - Running Jobs
  - Successful Jobs
  - Failed Jobs
  - Jobs during stage-in
  - Jobs during stage-out

# Rich Client: Security Setup

- ▶ Keystore view: manage your private keys
- ▶ Truststore view: Who can you trust?
- ▶ Conclusion: fully fledged keystore editor in Eclipse
- ▶ Default credentials vs. site specific credentials

The screenshot displays the Eclipse IDE interface with several panels open. The 'Grid Browser' panel on the left shows a tree view of the Grid infrastructure, including 'UNICORE testbed', 'Internal testbed', and 'Workflow engine at zam461.zam.kfa-juelich.de'. The 'Details' panel below it shows site-specific credentials for 'bastian demuth' at various URLs. The 'Keystore' panel on the right shows a table of keys with a context menu open over the 'bastian demuth pro' entry. The 'Truststore' panel at the bottom shows a table of trusted certificates.

alias	issued for	issued by	valid to
c9m-test-servorch	c9m-test-servorch	fake ca for testing only	2008-12-31 12:00
demo user	demo user	fake ca for testing only	2008-12-31 12:00
bastian demuth pro	bastian demuth	fzj-projects-ca	
c9m fzj vsite	c9m fzj vsite	chemomentum	
xnjs	xnjs-testbed.chemc	chemomentum	
bastian demuth	bastian demuth	chemomentum	

common name	distinguished name	valid to
dfn-verein server ca grid - g0	CN=DFN-Verein Server CA Grid - G01, OU=DFI	2009-07-07 04:56
chemomentum pilot ca	OU=ICM, O=Warsaw University, CN=Chemom	2010-01-01 12:00
fzj-projects-ca	EMAILADDRESS=projects-ca@fz-juelich.de, CN=	2013-12-01 09:01

# Rich Client: Workflow Editor

- ▶ Graphical programming (drag & drop, jobs, loops, etc.)
- ▶ Workflow variables
- ▶ Zoomable, printable, automatic layouting, undo
- ▶ Highly extensible (e.g. new subgraphs)

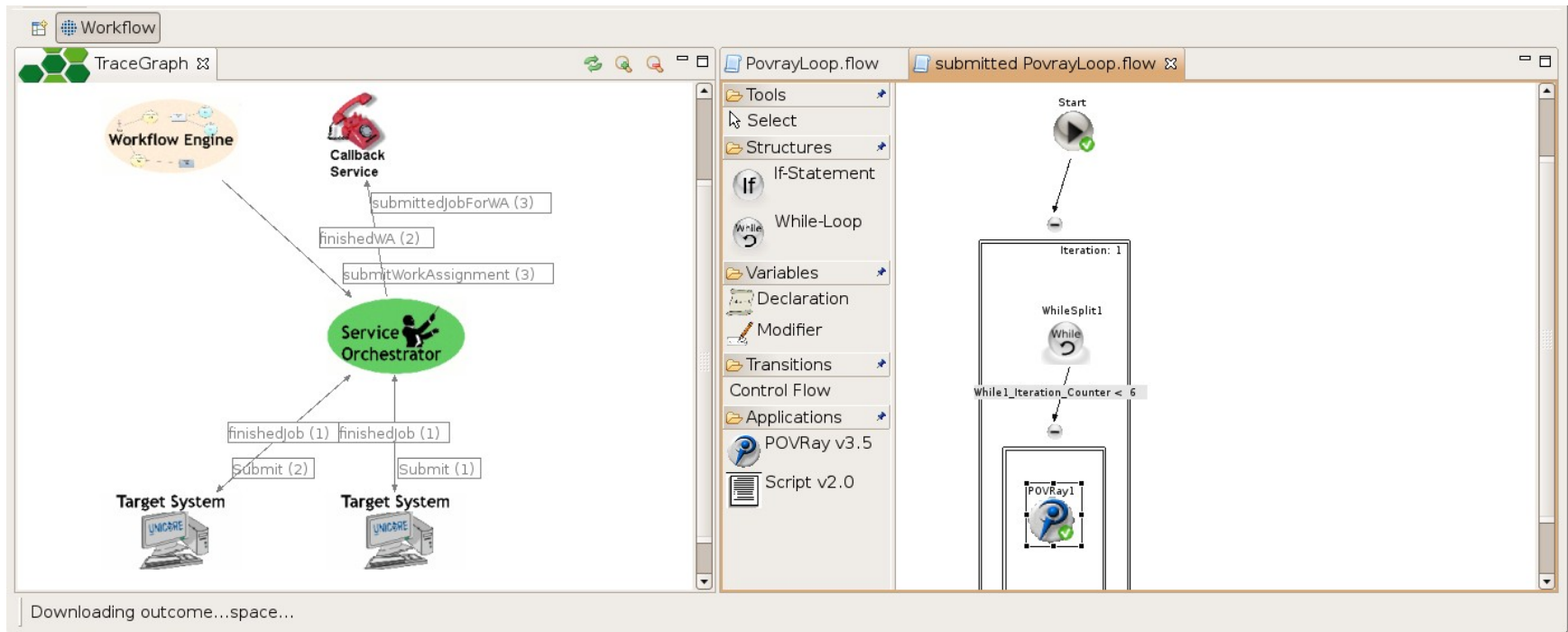
The screenshot displays the Workflow Editor interface. On the left, a 'Properties' panel for 'POVRay1' shows fields for Name, Width (320), Height (200), Anti Aliasing (0.5), Animate (checked), Initial Frame (0), Final Frame (5), Subset Start Frame (0), Subset End Frame (0), Initial Clock (0), and Final Clock (2.0). Below this is a 'Declared variables' table:

Property	Value
Declared variables	
While1_iteration_counter	
Initial value	1
Name	While1_iteration_counter

The main workspace shows a workflow diagram with a 'While' loop containing a 'WhileSplit1' node and a 'POVRay1' node. A toolbar on the right lists various actions: Team, Compare With, Replace With, Delete, Export, Layout Diagram, Open application, Print, Redo, Rename, Save, Select All, Show outcomes, Submit Workflow, Trace Workflow, Undo, Zoom In, Zoom Out, and Input Methods.

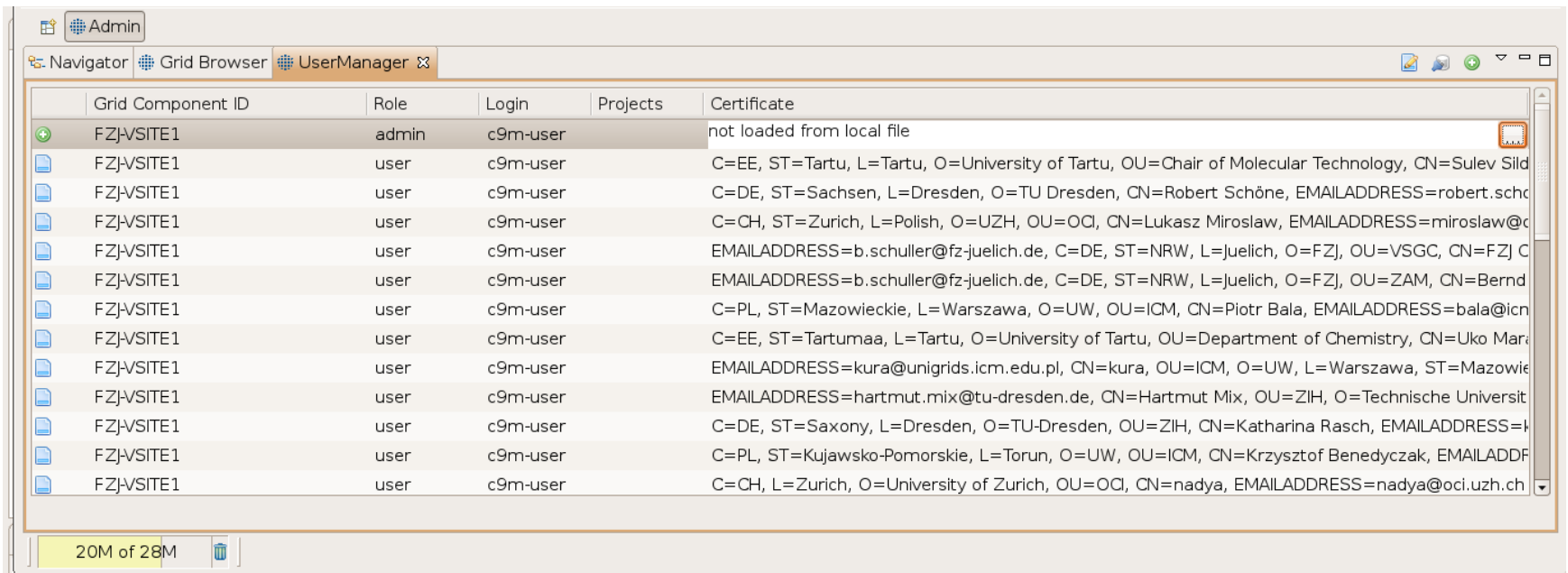
# Rich Client: Monitoring & Tracing

- ▶ Workflows monitored during execution (small status icons)
- ▶ Live tracing of messages with time stamps
  - ▶ Where did my jobs go?



# Rich Client: User Administration

- ▶ Additional Eclipse plugin
- ▶ Contact an XUADB remotely and change user data
- ▶ Add/remove users, change login, groups, certificate
- ▶ Strong security: XUADB has to trust the admin's certificate



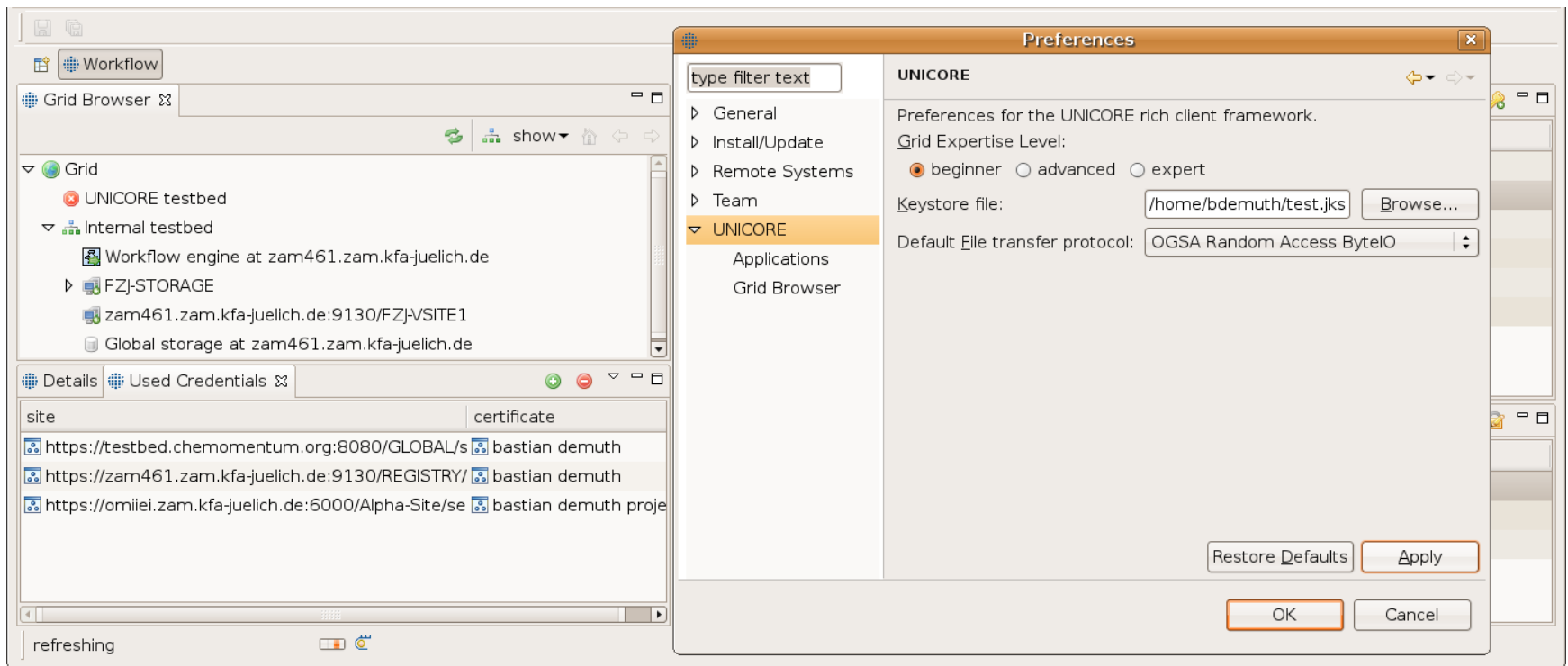
The screenshot shows the Eclipse IDE interface with the 'UserManager' plugin active. The main window displays a table of users with the following columns: Grid Component ID, Role, Login, Projects, and Certificate. The first row is selected, showing an 'admin' user with login 'c9m-user' and a certificate 'not loaded from local file'. The subsequent rows show 'user' accounts with various login names and certificates containing organizational information.

Grid Component ID	Role	Login	Projects	Certificate
FZJ-VSITE1	admin	c9m-user		not loaded from local file
FZJ-VSITE1	user	c9m-user		C=EE, ST=Tartu, L=Tartu, O=University of Tartu, OU=Chair of Molecular Technology, CN=Sulev Sild
FZJ-VSITE1	user	c9m-user		C=DE, ST=Sachsen, L=Dresden, O=TU Dresden, CN=Robert Schöne, EMAILADDRESS=robert.sch
FZJ-VSITE1	user	c9m-user		C=CH, ST=Zurich, L=Polish, O=UZH, OU=OCI, CN=Lukasz Miroslaw, EMAILADDRESS=miroslaw@c
FZJ-VSITE1	user	c9m-user		EMAILADDRESS=b.schuller@fz-juelich.de, C=DE, ST=NRW, L=Juelich, O=FZJ, OU=VSGC, CN=FZJ C
FZJ-VSITE1	user	c9m-user		EMAILADDRESS=b.schuller@fz-juelich.de, C=DE, ST=NRW, L=Juelich, O=FZJ, OU=ZAM, CN=Bernd
FZJ-VSITE1	user	c9m-user		C=PL, ST=Mazowieckie, L=Warszawa, O=UW, OU=ICM, CN=Piotr Bala, EMAILADDRESS=bala@icn
FZJ-VSITE1	user	c9m-user		C=EE, ST=Tartumaa, L=Tartu, O=University of Tartu, OU=Department of Chemistry, CN=Uko Mar
FZJ-VSITE1	user	c9m-user		EMAILADDRESS=kura@unigrads.icm.edu.pl, CN=kura, OU=ICM, O=UW, L=Warszawa, ST=Mazowie
FZJ-VSITE1	user	c9m-user		EMAILADDRESS=hartmut.mix@tu-dresden.de, CN=Hartmut Mix, OU=ZIH, O=Technische Universit
FZJ-VSITE1	user	c9m-user		C=DE, ST=Saxony, L=Dresden, O=TU-Dresden, OU=ZIH, CN=Katharina Rasch, EMAILADDRESS=K
FZJ-VSITE1	user	c9m-user		C=PL, ST=Kujawsko-Pomorskie, L=Torun, O=UW, OU=ICM, CN=Krzysztof Benedyczak, EMAILADDF
FZJ-VSITE1	user	c9m-user		C=CH, L=Zurich, O=University of Zurich, OU=OCI, CN=nadya, EMAILADDRESS=nadya@oci.uzh.ch

20M of 28M

# Rich Client: Preferences

- ▶ Re-use the preference dialog of Eclipse
- ▶ Add some UNICORE specific settings



Thank you!



Downloads, documentation, tutorials, mailing lists, community links, and more:

<http://www.unicore.eu>