GENERAL INTRODUCTION TO UNICORE 6

Michele Carpené – CINECA SCAI
m.carpen@cineca.it
Liverpool 03/09/2012

Agenda

Outline:

- Introduction
- Architecture and Services
- Clients and Application Integration
- Conclusions
- References

Introduction

UNiform **I**nterface to **CO**mputing Resources

- Job submission service
- seamless, secure, and intuitive
- Initial development started in two German projects funded by the German ministry of education and research (BMBF)
- Results: well defined security architecture with X.509 certificates, intuitive GUI, central job supervisor
- Results: implementation enhancements, extended job control (workflows)
- application specific interfaces (plugins)
- Continuous development since 2002 in several EU projects
- Open Source community development since Summer 2004

Job Submission

UNICORE provides a seamless interface for preparing and submitting jobs to the computing resource. It has a three-tier design:

- a *client GUI*, used for the preparation, submission, monitoring, and administration of jobs,
- a *gateway*, a point of contact for all UNICORE connections, checking the user's certificate and providing the access to the service
- a *Target System Interface (TSI),* a server that runs on the target machine interfacing to the specific batch scheduler.

Data Management

- Storage Management Service (SMS) provides an abstract “file system” view on a storage resource
- Common operations (e.g. mkdir, delete, listDirectory, etc.)
- Designed for extensibility
- File transfer management

Workflows

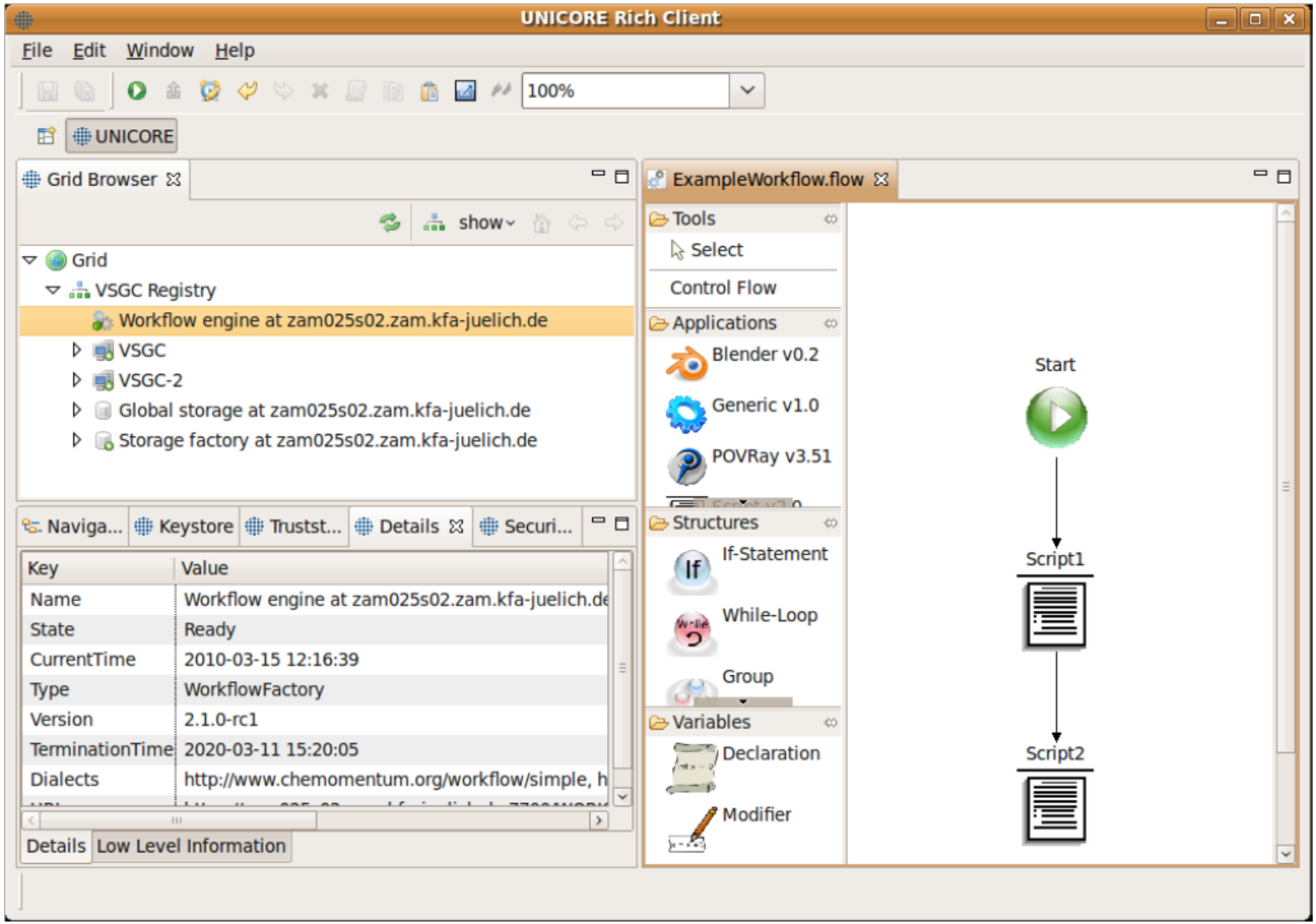
- Workflow engine
- Multiple instances can be deployed for scalability
- Trace utility
- Useful for performance evaluations, monitoring and reporting
- Control Constructs: ForEach-Loop, While-Loop, If-Statement, Grouping
- Conditions: exit code, file existence, file size, workflow variables
- Clients
- URC and UCC

URC – UNICORE Rich Client

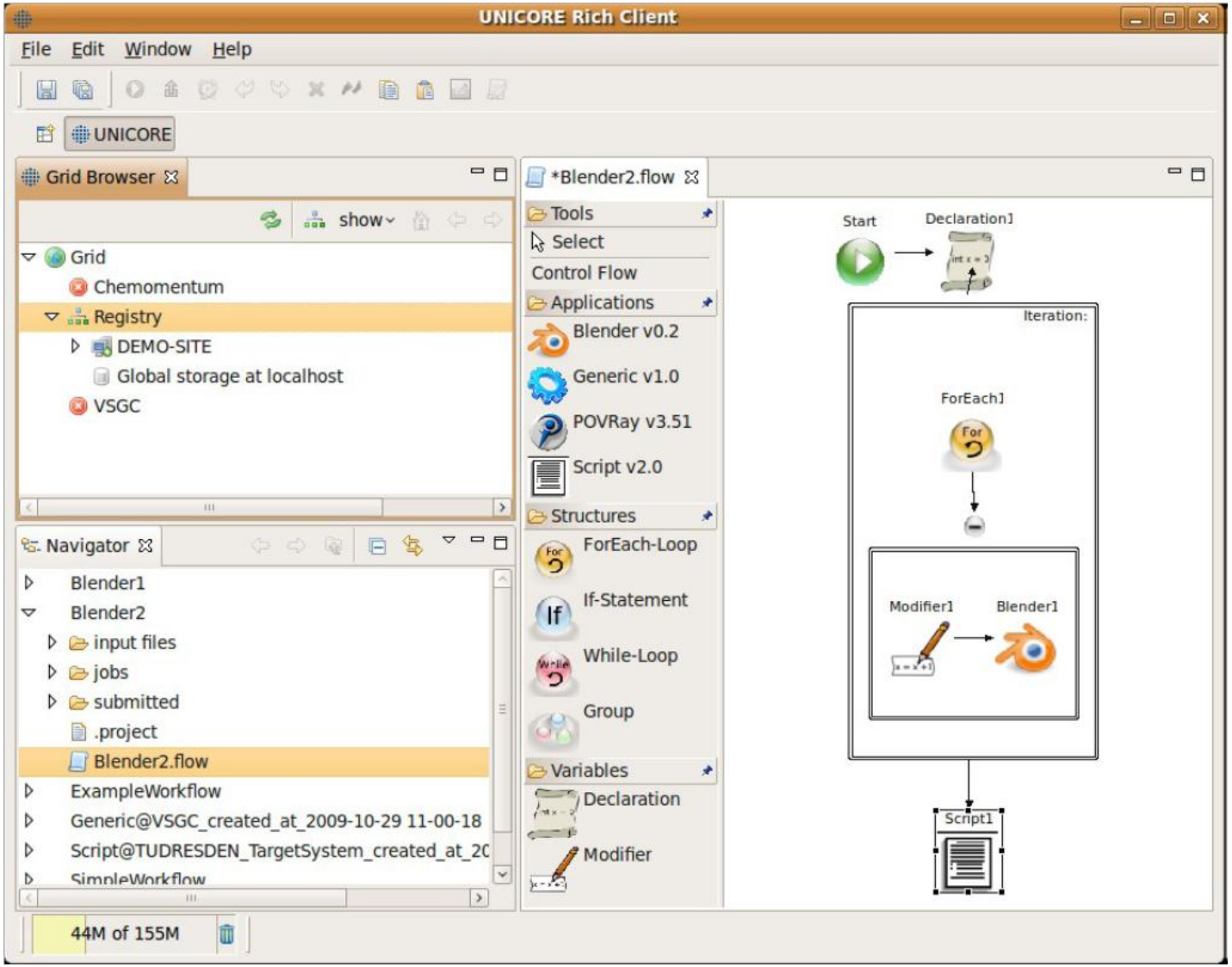
Graphical client for UNICORE 6

- Building, submitting and monitoring jobs
- Workflow editor
- Integrated data and storage management
- Pluggable credentials
- “Simple view” for novice users
- Based on Eclipse framework
- Extensibility through GridBeans
- For Windows, Linux/Unix and MacOS/X (with Java 5 and Eclipse 3.5)

URC – UNICORE Rich Client



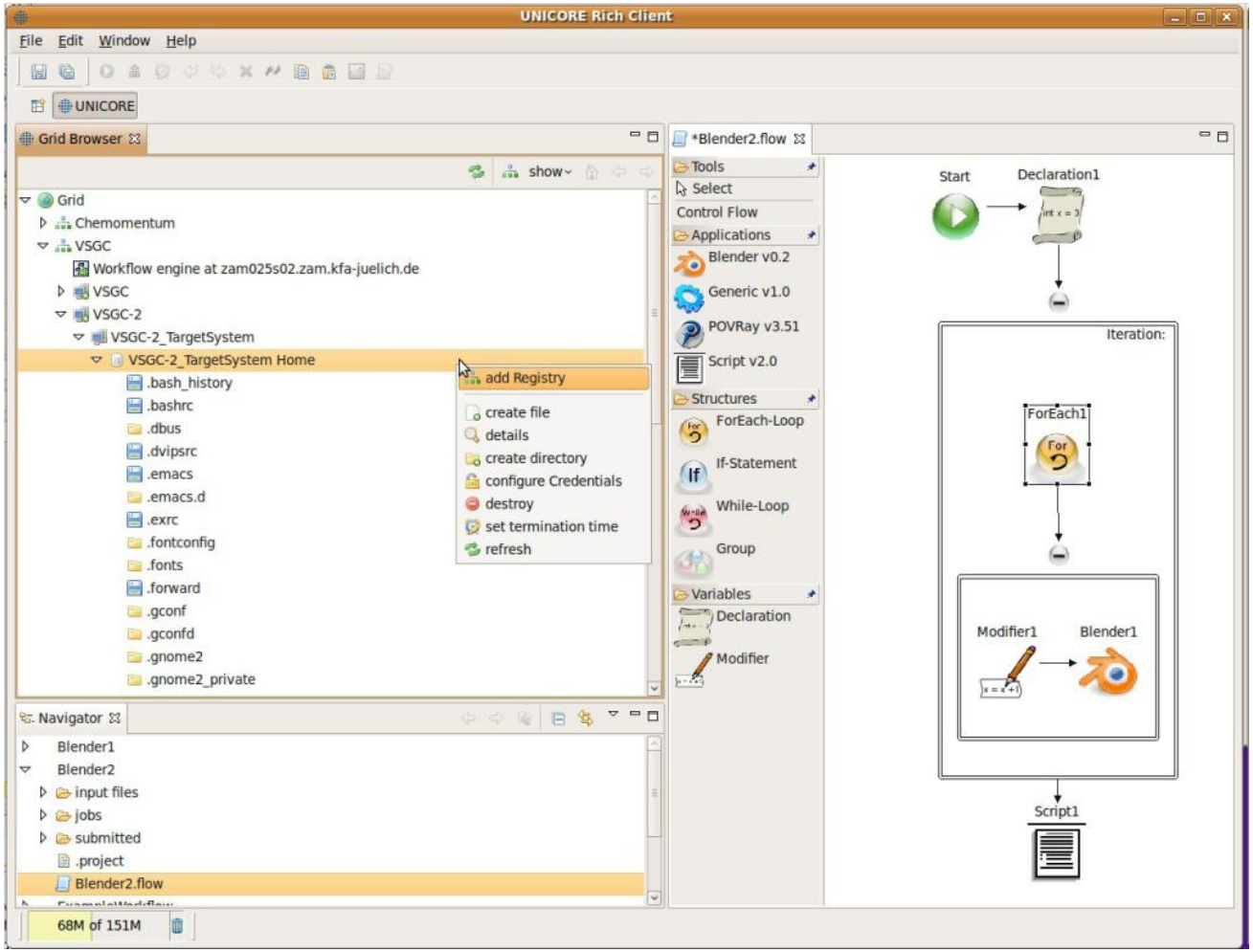
URC - Complex Workflows



URC – UNICORE Rich Client

Integrated Storage Management

- Create files
- Drag and drop from/to desktop
- Copy and paste
- Remote file editing



Application Integration

- Grid Bean as additional plugins
- GridBean allows application-specific interfaces for the UNICORE Rich Client (URC)
- Input parameters available
- Output of application can be visualized
- Common for all GridBeans: file stage-in/-out, resource requirements, variables
- Easily implementable through a powerful API
- Available existing GridBeans for scientific applications: MOPAC, **NAMD**, Blender, Povray, and more life science Gridbeans at **http://unicore-life.sourceforge.net/**

Conclusions

11 Reasons to use UNICORE

- Mature Grid technology from Europe
- Made for Supercomputing / HPC Grids, Grids of Clusters and single PCs
- User support directly from the developers
- Easy to install and configure
- Graphical user interface
- Command line client
- Application integration
- Workflow support
- Interoperable
- Extensible
- General Purpose

References

Main references:

- UNICORE site: **http://www.unicore.eu/**
- Download Page: **http://www.unicore.eu/download/unicore6/**
- SourceForge: **UNICORE Rich Client for Linux/Windows/MacOs**
- Manuals: **http://www.unicore.eu/documentation/manuals/unicore6/**
- Plugin resources: **http://unicore-life.sourceforge.net/**
- Plugin Developer's Guide:
http://www.unicore.eu/documentation/manuals/GridbeanDevelopersGuide.pdf

What you need:

- Rich Client (last version 6.3.0)
- INFN CA certificate
- Your personal certificate
- Keystore, downloadable:
  http://winnetou.sara.nl/deisa/certs/keystore.jks
- Configure CINECA-Registry with proper url:
  https://grid.cineca.it:13303/CINECA-REGISTRY/services/Registry?res=default_registry