Cross-Grids Simulation Scenarios with the GridSFEA Framework

Dr. Ing. Ioan Lucian Muntean
Advanced Computing Application Lab
Department of Computer Science
UTC-N, Cluj-Napoca

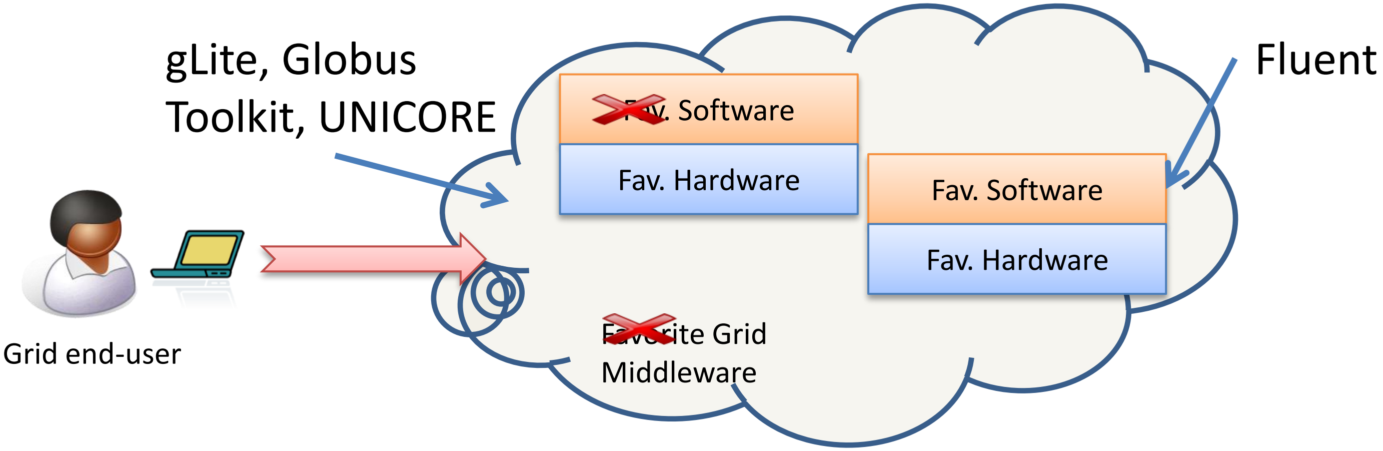
EGI User Forum
Vilnius, April 14, 2011

Outline

- Motivation and Challenges
- GridSFEA
  - Overview
  - Technical insights
  - Job submission, migration of scenarios
- Simulation Scenarios
- Conclusions

Motivation

How to work with more grids?



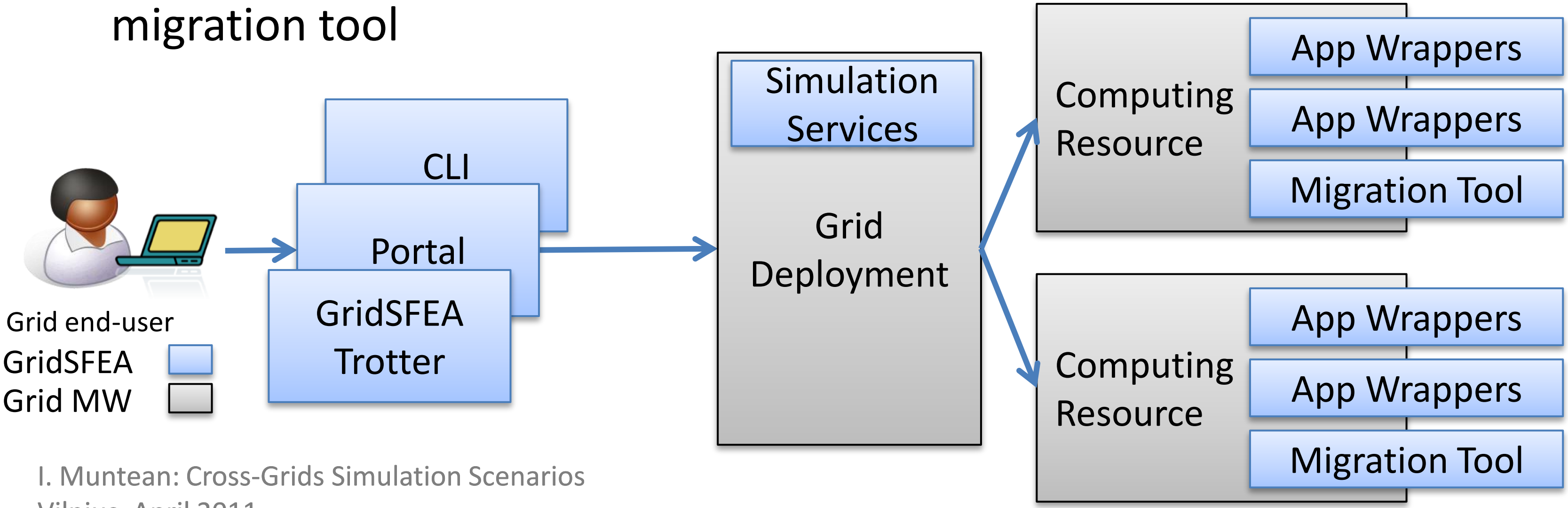
- Potential approaches
  - Unify the grid middleware stacks – UMD of EGI (large European efforts!)
  - Develop client tools capable of working with multiple middleware
- An end-user oriented tool for working with different compute grids (not only with your favorite one!)
  - Increase the user’s flexibility of working with grids
  - Globus Toolkit, gLite, Unicore etc
- Immediate benefits for the end user communities
  - Start right away with the using the needed middleware
  - On the long run – capable of operating with the upcoming UMD

Challenges

- …of supporting Application Level Interoperability
- Although public open standards are available …
  - Different job submission languages (RSL, RSL (XML), JDL etc)
  - Different protocols for file transfer and access
  - Differences in authentication/authorization mechanisms (e.g. no VOMS support in GT so far)
- Integrate client libraries/tools from different middleware
- Provide an additional layer to the middleware specifics

GridSFEA

- Grid Simulation Framework for Engineering Applications →*a tool for grid end-users*
- Simulation of large systems, Slow physical processes, Optimization tasks
- Developed initially at TUM, now mainly at UTC-N
- Bring together: simulation and application specialists, grid administrators, grid developers (TUM, UTC-N, Univ. of Stuttgart, LRZ, HLRS)
- Main components: user tools, simulation services, application wrappers, migration tool



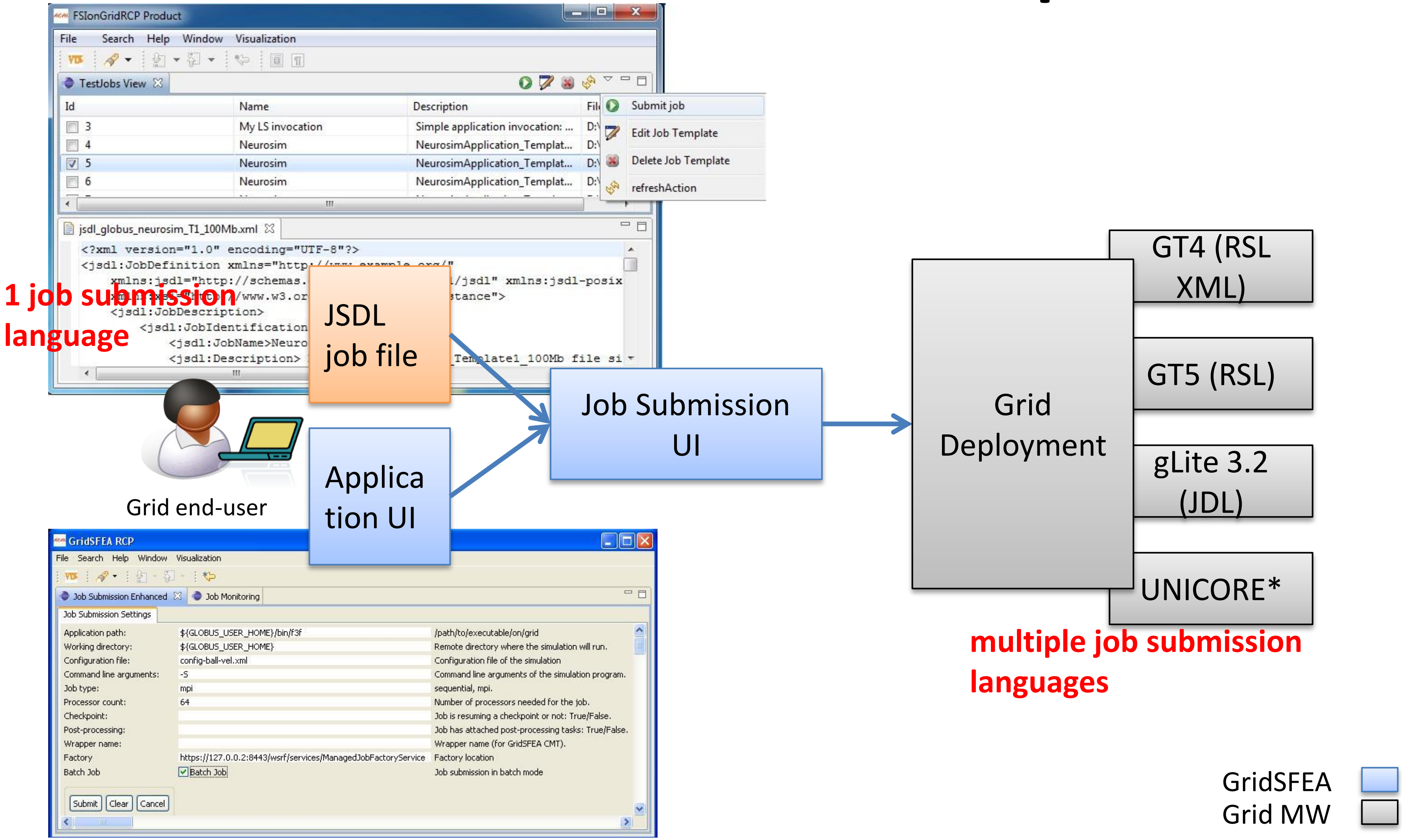
GridSFEA Plugins

- Goals:
  - Foster export of functionalities to other Grid community tools (e.g. g-Eclipse, WS-VLAM)
  - Provide a frame for proper integration of further features in GridSFEA tools
  - Straight forward creation of customized grid applications with GridSFEA features
- User-side application: based on Eclipse
- GridSFEA user tools →Eclipse plugins
- Features available as plugins
  - Job submission, job monitoring, scenario/file transfer
  - Simulation explorer, parameter study

GridSFEA features

•Migration of scenarios
•Simulation preview
•Post processing
•High-level operations on simulation data
•Parameter studies
•Long simulation runs
•Deployment on systems from D-Grid, DEISA Grid, RO-NGI
•Lightweight realization
•Wrapper-based gridification

Submission of Jobs – User's Perspective



FSIonGridRCP Product

File Search Help Window Visualization

TestJobs View

Id	Name	Description	Fil
□ 3	My LS invocation	Simple application invocation: ...	D:\
□ 4	Neurosim	NeurosimApplication_Templat...	D:\
☑ 5	Neurosim	NeurosimApplication_Templat...	D:\
□ 6	Neurosim	NeurosimApplication_Templat...	D:\

jsdl_globus_neurosim_T1_100Mb.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<jsdl:JobDefinition xmlns="http://www.example.org/"
    xmlns:jsdl="http://schemas.                    /jsdl" xmlns:jsdl-posix
    xmlns:xsi="http://www.w3.or                    stance">
    <jsdl:JobDescription>
        <jsdl:JobIdentification
            <jsdl:JobName>Neuro
            <jsdl:Description>                     Template1_100Mb file si
```

1 job submission language

GridSFEA RCP

File Search Help Window Visualization

Job Submission Enhanced | Job Monitoring

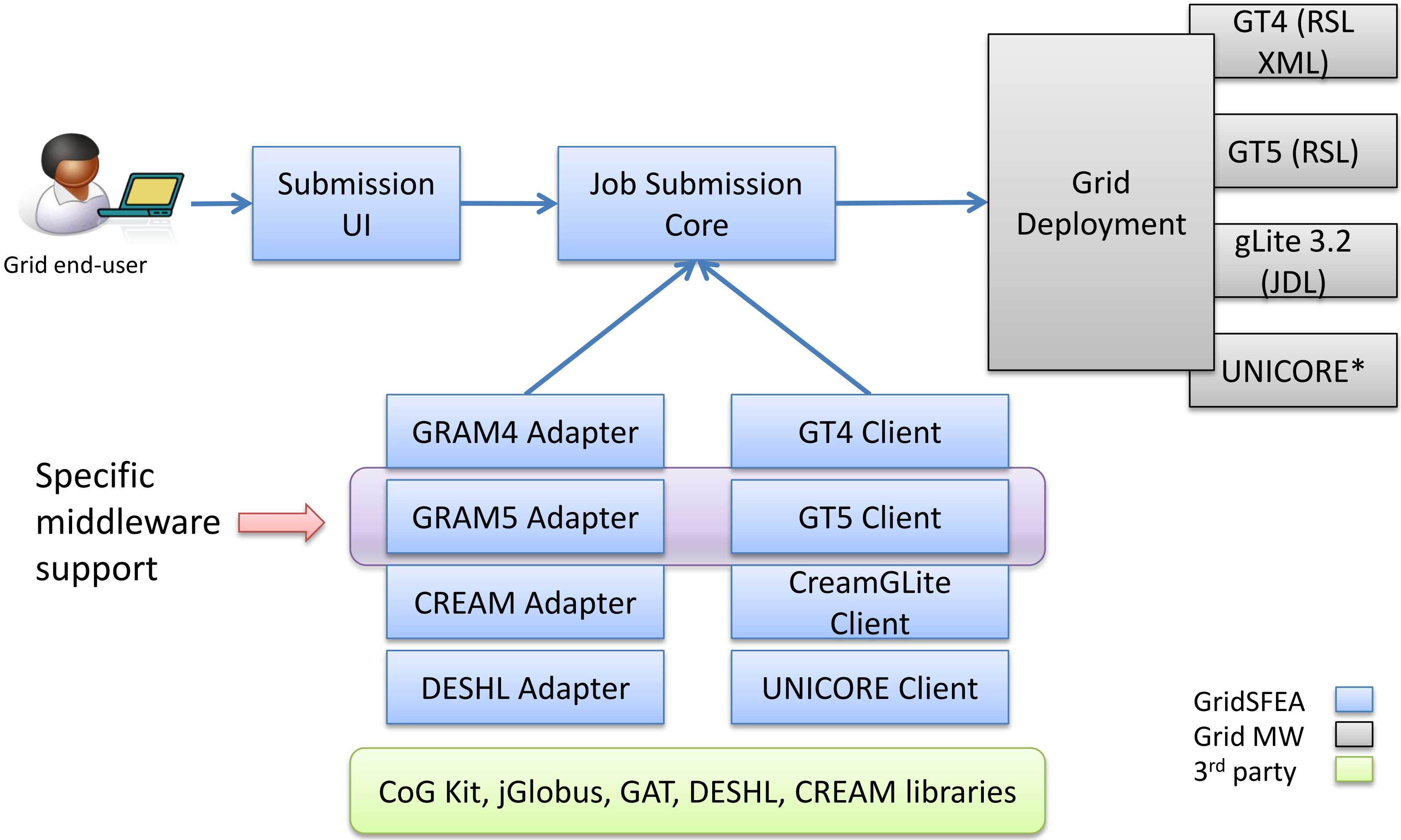
Job Submission Settings

Application path:	${GLOBUS_USER_HOME}/bin/f3f	/path/to/executable/on/grid
Working directory:	${GLOBUS_USER_HOME}	Remote directory where the simulation will run.
Configuration file:	config-ball-vel.xml	Configuration file of the simulation
Command line arguments:	-S	Command line arguments of the simulation program.
Job type:	mpi	sequential, mpi.
Processor count:	64	Number of processors needed for the job.
Checkpoint:		Job is resuming a checkpoint or not: True/False.
Post-processing:		Job has attached post-processing tasks: True/False.
Wrapper name:		Wrapper name (for GridSFEA CMT).
Factory	https://127.0.0.2:8443/wsrf/services/ManagedJobFactoryService	Factory location
Batch Job	☑ Batch Job	Job submission in batch mode

multiple job submission languages

GridSFEA ▭
Grid MW ▭

Submission of Jobs – Dev's Perspective



Grid end-user

Submission UI

Job Submission Core

Grid Deployment

GT4 (RSL XML)

GT5 (RSL)

gLite 3.2 (JDL)

UNICORE*

GRAM4 Adapter

GT4 Client

GRAM5 Adapter

GT5 Client

CREAM Adapter

CreamGLite Client

DESHL Adapter

UNICORE Client

Specific middleware support

CoG Kit, jGlobus, GAT, DESHL, CREAM libraries

GridSFEA 🟦
Grid MW ⬜
3rd party 🟩

Job Submission

- Based on the JSDL standard and its extensions JSDL-POSIX, JSDL-SPMD Application
  - Partial mappings to RSL, RSL-XML, CREAM JDL
  - UNICORE submission handled with DESHL directly (only in DEISA grid)
- Advantages of our approach
  - Use only one specification language for jobs (though also the other are supported)
  - Exploit features available in each client of grid middleware (e.g. delegation, file staging)
  - Proper encapsulation of bunches of third-party libraries (limit propagation of incompatibilities and code conflicts)
  - Easy integration of further middleware adaption as plugins
- Limitations
  - Initial development effort quite high
  - Still have to cope with some middleware specifics (e.g. VOs)
  - Selection of middleware type based on external properties file

File Transfers

- Use supported transfers provided by community libraries
  - GAT, DESHL, …
  - Based on GridFTP
- Advantages
  - Wide deployment of GridFTP in production grids
  - Present in both GT- and gLite-based environments
  - Support of X.509 certificates
  - Rather stable API

Migration of Simulation Scenarios with GridSFEA

- Handle the execution of long simulation scenarios on heterogeneous Grids
  - Application-generated checkpoints
  - Use of meta data
  - Application wrappers → minimal invasion of user program/simulation software
  - Submission of *continuation jobs* in behalf of the user (delegation)
- Application-oriented approach to migration
  - Nearly compliant with OGF recommendation for system-oriented GridCPR
- Enable migration on *production* Grids

Annotation of simulation scenarios

- Bunch of input files
- Configuration files
- Simulation output (results)
- Simulation software
- Checkpoints

Migration of Simulation Scenarios (2)

- Automated (autonomous) migration
  - The framework submits redundant jobs to the sites specified by the user
  - Jobs join a competition at startup. Only the winner may compute. The others terminate.
  - May lead to high resource consumption (cpuh!)
- Basic (assisted) migration
  - The user submits manually continuation jobs, without updating anything in the job files
  - More suitable when accounting systems are in place ☺
- Continuation jobs can resume from any checkpoint (by default from the last one registered with the framework)

Migration with GridSFEA

- 1 x simple configuration file

```
APP_PATH = "simProgram"
CHECKPOINT = True
POSTPROCESS = True
JOB_TYPE = “mpi"
WRAPPER_NAME = “appwrapper"
APP_ARGLIST = ["-arg"]
```

- 1 x job script
- job submission

Migration of Simulation Scenarios (3)

GridSFEA

- services, application wrapper, migration tool

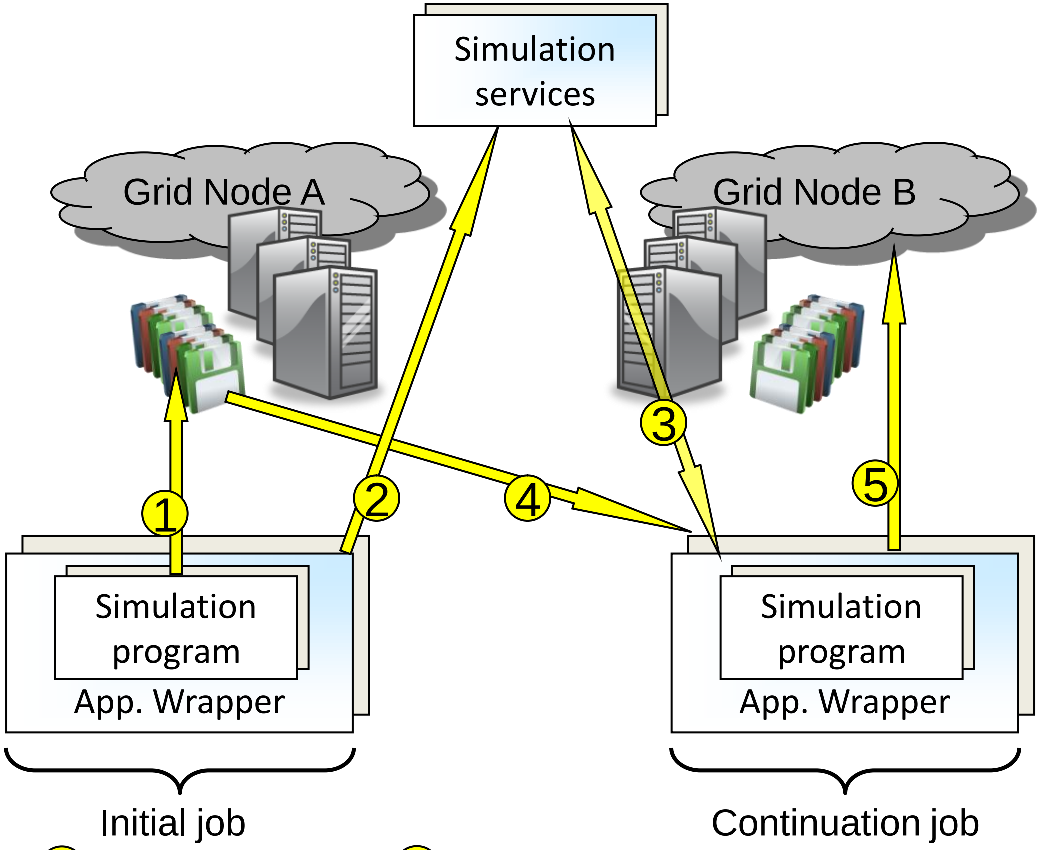
Technologies

- Web Service, WSDL, GSI

Prerequisites for the CSE program

- non-interactive use
- checkpointing
- simple application wrapper (configure handling of in/out/checkpoint data)
- pre-installation on Grid

Application level checkpoint-based migration



1 write checkpoint

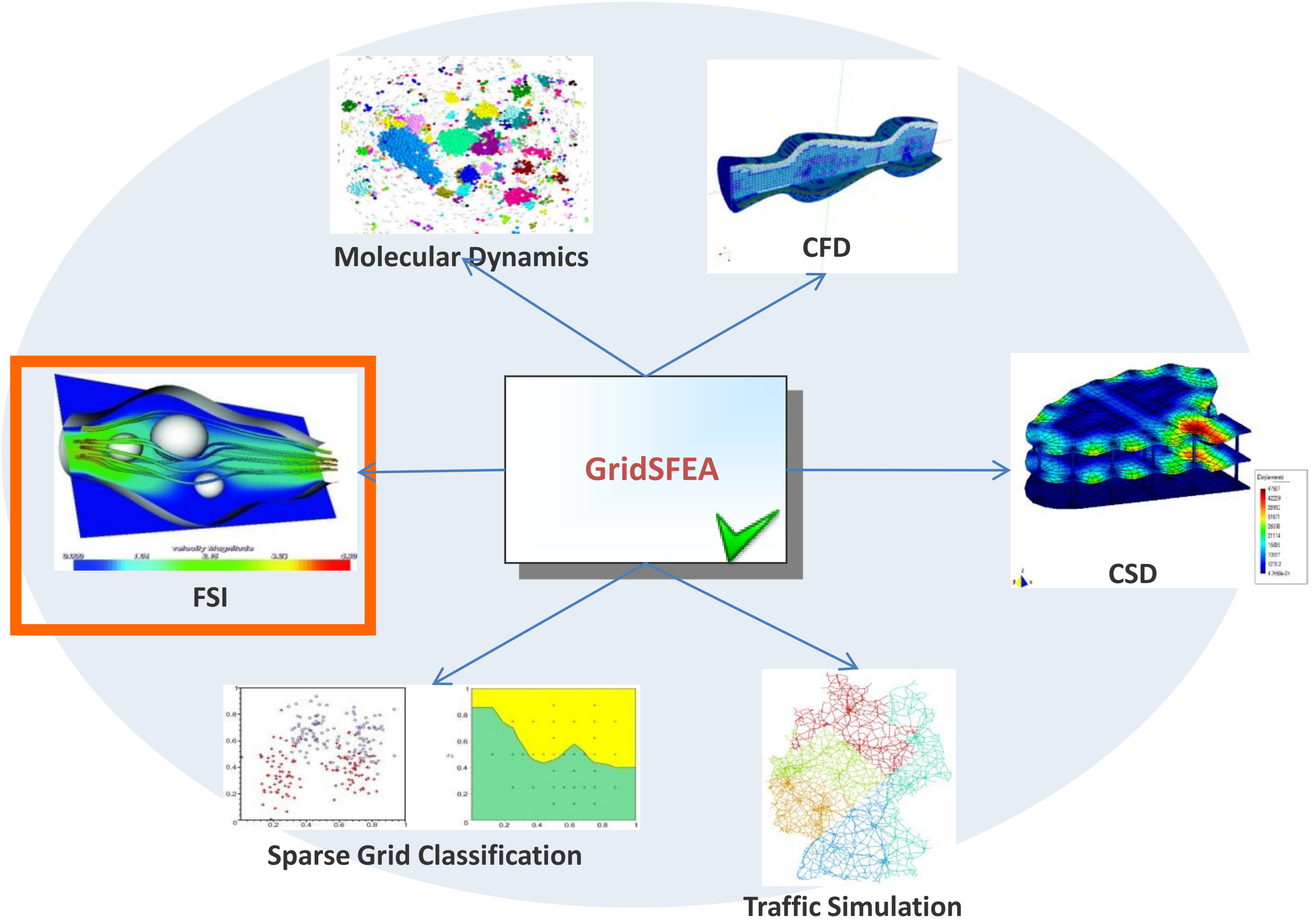
2 register meta data

3 request/get meta data

4 retrieve checkpoint

5 submit job

Selected Application Scenarios



GridSFEA

Molecular Dynamics

CFD

FSI

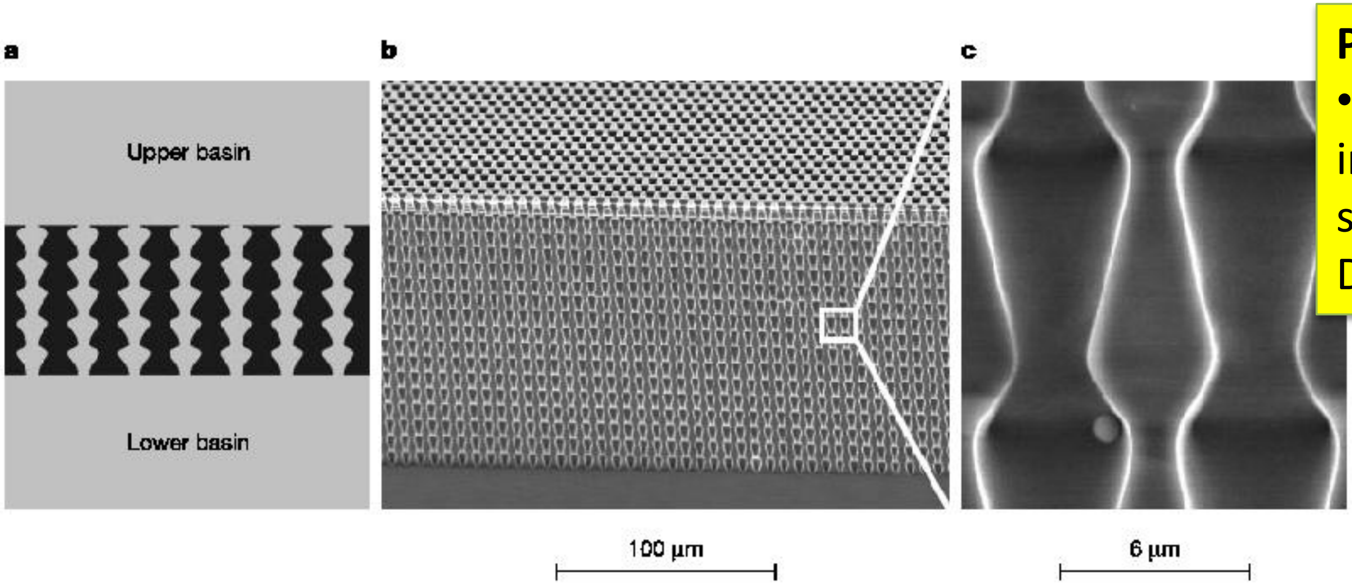
CSD

Sparse Grid Classification

Traffic Simulation

Drift Ratchet Simulations

- Suite of distributed particle simulation scenarios – *The Drift Ratchet Problem*
  - Silicon wafers filled with water, pressure pumps with oscilating direction → particle sorting devices
  - Goal of the research: understand the physics behind, development of simplified yet sufficiently relevant models



Potential application
•Medicine and drug industry: sorting particles such as proteines and DNA fragments

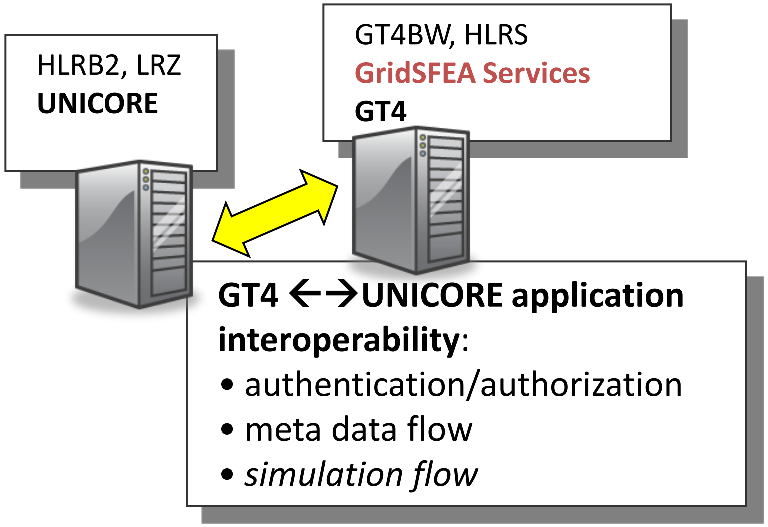
Challenges of the Drift Ratchet

- Model challenges
  - Phenomena in different domains (fluid flow – particle movement)
  - Type of models (deterministic fluid flow – stochastic Brownian motion)
  - Time scales (fast oscillations of pressure in the pressure pumps and of the particles – long-term averaged direction of particle movements)
  - Particle movements over the entire computational domain
- Computational challenges
  - 3D flows, regular meshes, rather high resolution, low Reynolds number (Re<0,1), tiny time-steps

Very long simulation times!!!

Migration and Preview on D-Grid and DEISA

- Basic (assisted) migration!
- Job submission, checkpoint registration and transfer, computing job submission



HLRB2, LRZ
UNICORE

GT4BW, HLRS
GridSFEA Services
GT4

GT4 ←→UNICORE application interoperability:
- authentication/authorization
- meta data flow
- *simulation flow*

Migration times (D-Grid)

	Setup	File Transfer	Computation
initial phase	43 s	1.5 s	12 h
cont. phases	47 s	2.5 s	12h,72 h

GridSFEA overhead < 40 time steps/migration!

one simulation ~ 1mio time steps

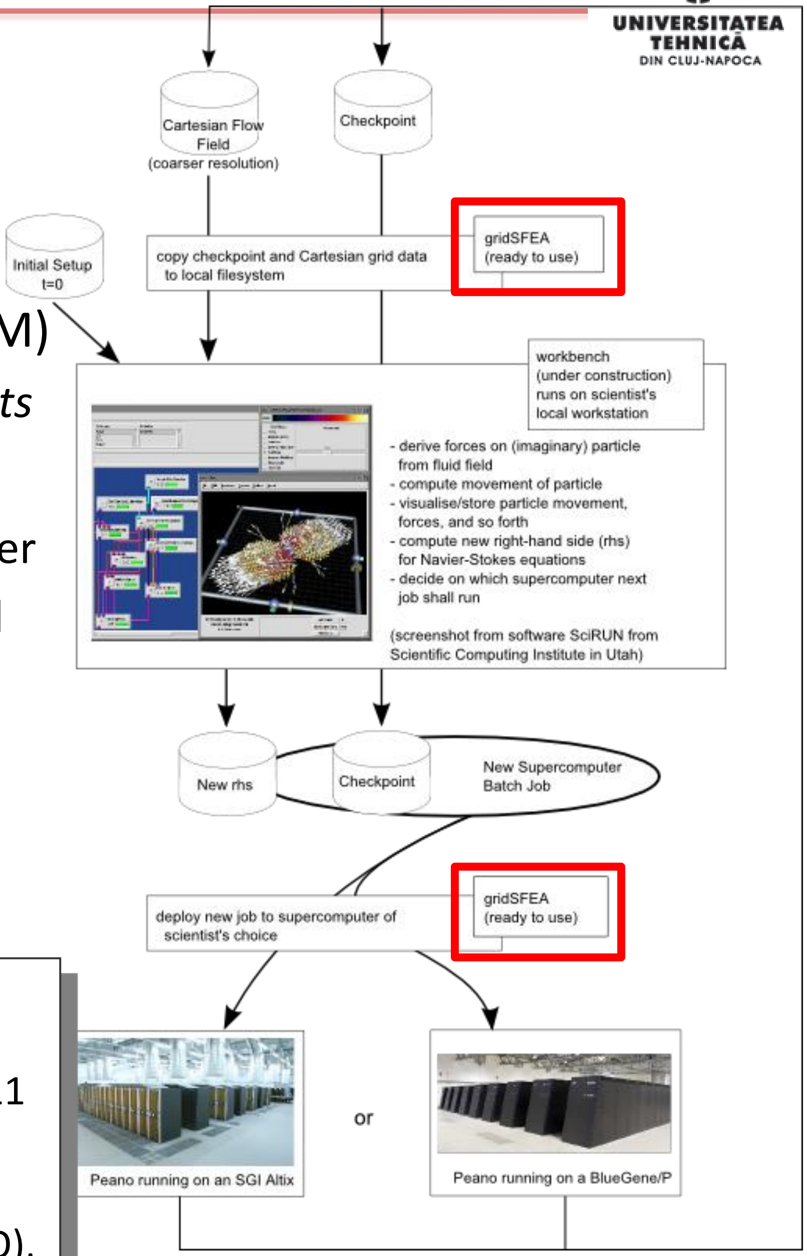
Joint work with SCCS@TUM

GridSFEA & Other Tools

- Workbench for numerical simulation (@TUM)
  - *Computational Steering* based on *checkpoints*
- GridSFEA framework
  - Acts as application-oriented middleware layer
  - Packs simulation tasks/scenarios as jobs and computes them on grid (in DEISA)
  - Handle grid-related interactions (job submission & monitoring, checkpoint registration and transfer etc)
  - Plugin-based integration of the two frameworks

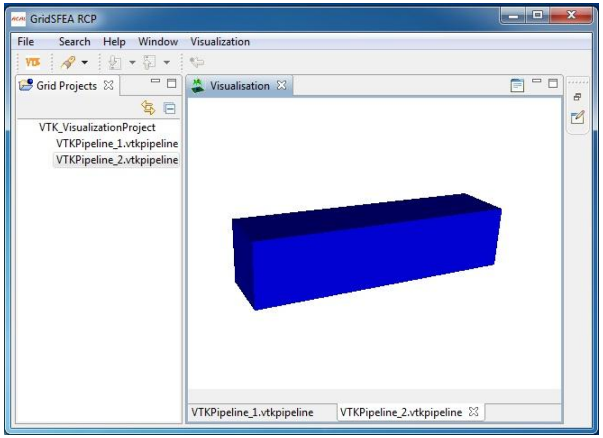
The DiParTs Project

- DEISA – DECI project
  - October 2010 – April 2011
  - About 500k cpuh x 3 (DEISA+LRZ+KAUST)
- Partners: TUM (DE), UTC-N (RO), KAUST (SA)

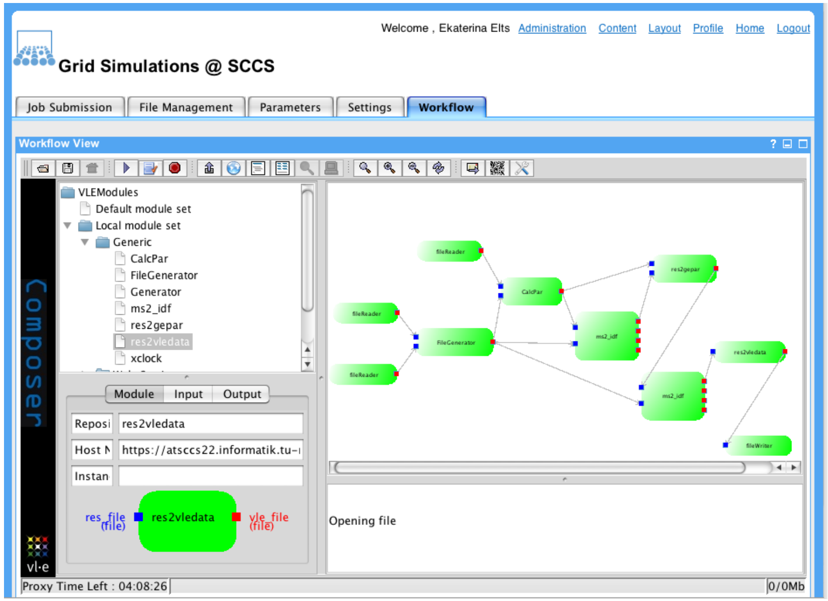

Cartesian Flow Field (coarser resolution)
Checkpoint
Initial Setup t=0
copy checkpoint and Cartesian grid data to local filesystem
gridSFEA (ready to use)
workbench (under construction) runs on scientist's local workstation
- derive forces on (imaginary) particle from fluid field
- compute movement of particle
- visualise/store particle movement, forces, and so forth
- compute new right-hand side (rhs) for Navier-Stokes equations
- decide on which supercomputer next job shall run
(screenshot from software SciRUN from Scientific Computing Institute in Utah)
New rhs
Checkpoint
New Supercomputer Batch Job
deploy new job to supercomputer of scientist's choice
gridSFEA (ready to use)
Peano running on an SGI Altix
or
Peano running on a BlueGene/P

Joint work with T. Weinzierl, TUM

GridSFEA & Other Tools


GridSFEA RCP
File Search Help Window Visualization
Grid Projects
Visualisation
VTK_VisualizationProject
VTKPipeline_1.vtkpipeline
VTKPipeline_2.vtkpipeline
VTKPipeline_1.vtkpipeline
VTKPipeline_2.vtkpipeline

Import of the VTK Visualization plugin from gEclipse



Reposi: res2vledata

Host N: https://atsccs22.informatik.tu-

Instan:

Opening file

Proxy Time Left : 04:08:26

0/0Mb

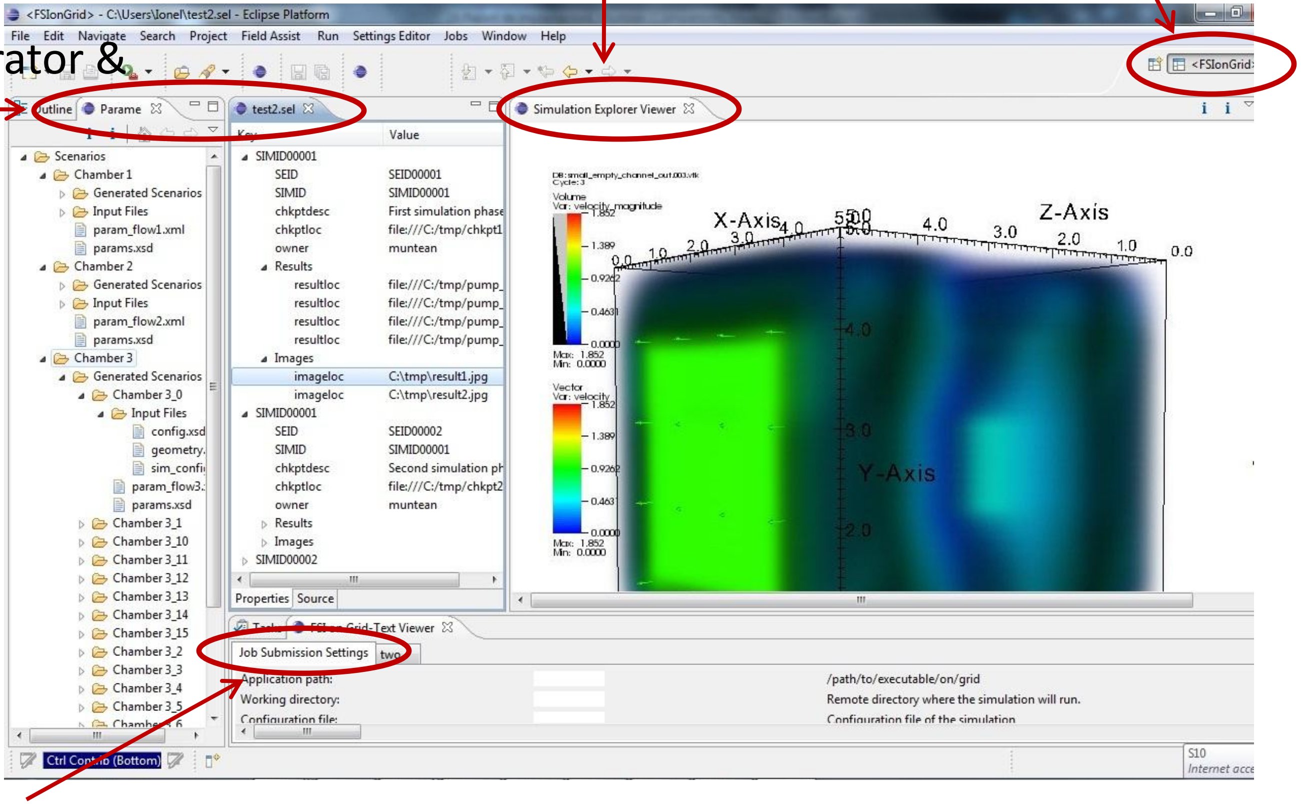
Export of the GridSFEA's parameter generator module to WS-VLAM workflows – work of Dr. E. Elts, TUM

A Customized Application: FSI on Grid

Simulation Explorer View

FSI on Grid perspective

Parameter generator & explorer



Key	Value
SIMID00001	
SEID	SEID00001
SIMID	SIMID00001
chkptdesc	First simulation phase
chkptloc	file:///C:/tmp/chkpt1
owner	muntean
Results	
resultloc	file:///C:/tmp/pump_
resultloc	file:///C:/tmp/pump_
resultloc	file:///C:/tmp/pump_
resultloc	file:///C:/tmp/pump_
Images	
imageloc	C:\tmp\result1.jpg
imageloc	C:\tmp\result2.jpg
SIMID00001	
SEID	SEID00002
SIMID	SIMID00001
chkptdesc	Second simulation ph
chkptloc	file:///C:/tmp/chkpt2
owner	muntean
Results	
Images	
SIMID00002	

Application path: /path/to/executable/on/grid

Working directory: Remote directory where the simulation will run.

Configuration file: Configuration file of the simulation

Job submission settings

Joint work with L. Munteanu,R. Cuc,A. Badiu (UTC-N)

Partly supported by the CGUTCN project

Conclusions & Outlook

- Development of user-oriented grid tools is important in order to make users benefit from the grid (esp. new communities)
- Differences between grids can be smoothed from user's perspective by placing an intermediate layer of applications
  - Increase user's independence of the grid MW
  - Comfortable interactions with the grid
- High development effort and a lot of grid know-how is demanded for application developers
- GridSFEA encapsulates/isolates MW specifics and offers a set of high-level operations for the end-user
- Short term – have GridSFEA user tools publicly released
- On the mid term – have client applications capable of operating with UMD services/products from different MWs

THANK YOU!

QUESTIONS?

How to Get GridSFEA

- Only the plugin-based stuff!
- Application plugins (for Eclipse IDE)
  - Public release (1.0) scheduled for May 31st, 2011
  - Check *http://acal.utcluj.ro*→*Research*→*Software* for updates OR http://acal.utcluj.ro/index.php?option=com_content&view=article&id=19&Itemid=19
- FSI on Grid
  - Release 1.0 pending (scheduled for May 31st 2011)
  - Licensing model – under discussion
- Upcoming features
  - GridWay client support
  - Handling of proxies (VOMS)