



Enabling Grids for E-science

Licensed Software in the EGEE infrastructure

Vangelis Floros, GRNET
EGEE NA4 Deputy Activity Leader

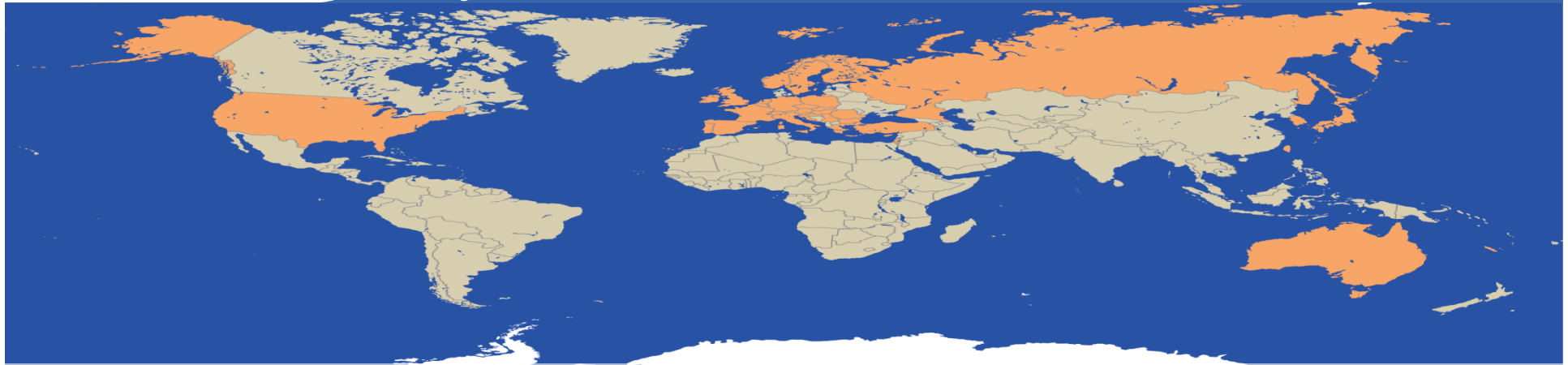
Grid License Management BoF
4th EGEE User Forum/OGF25
6 March 2009, Catania, ITALY



www.eu-egee.org



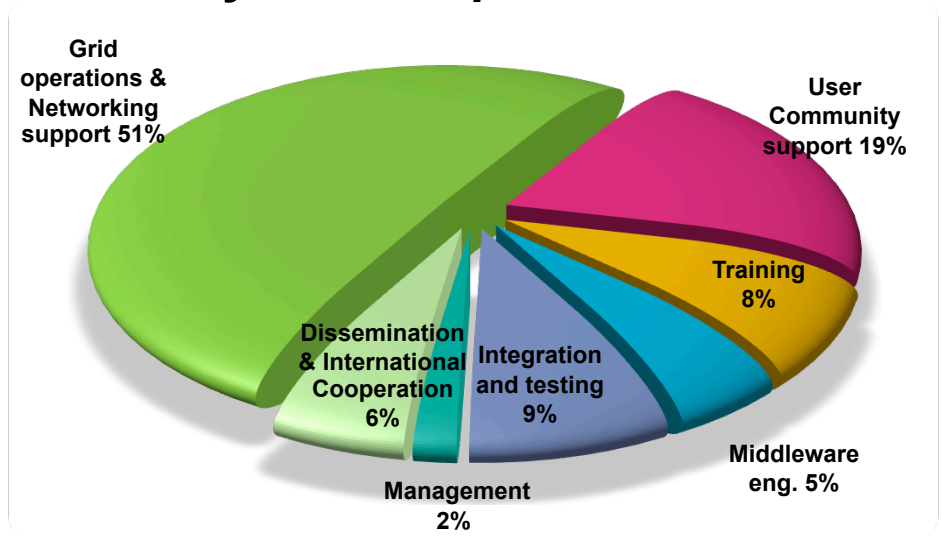
- **Introduction to EGEE's infrastructure and scientific communities**
- **Classification of licensed software used**
- **Approaches for license management**
- **Problems and Open Issues**
- **Conclusions**



Flagship Grid infrastructure project co-funded by the European Commission

Main Objectives

- Expand/optimize existing EGEE infrastructure, include more resources and user communities
- Prepare migration from a project-based model to a sustainable federated infrastructure based on National Grid Initiatives



Duration: 2 years

Consortium: ~140 organisations across 33 countries

EC co-funding: 32Million €

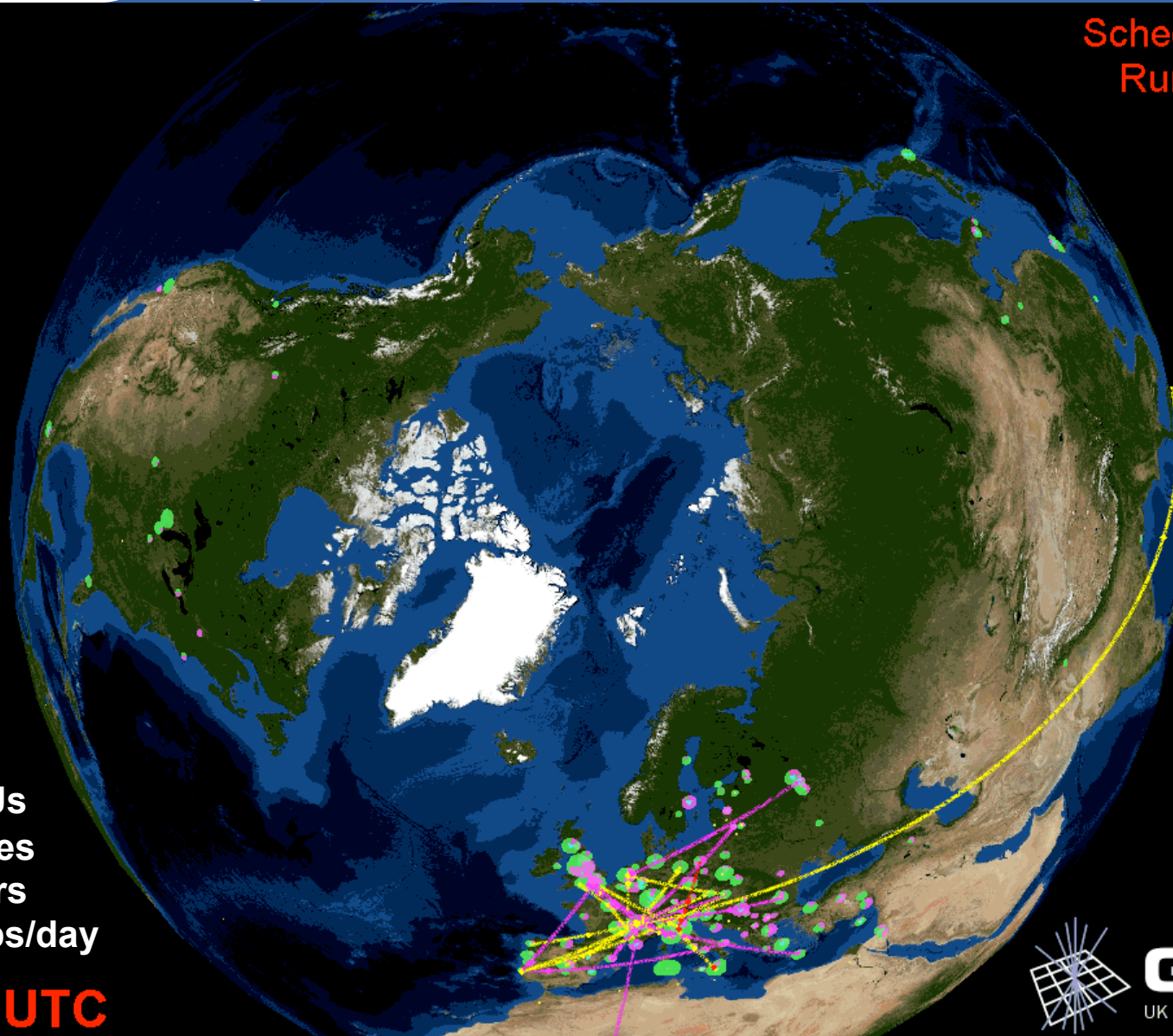


Enabling Grids for E-science

EGEE Grid Infrastructure

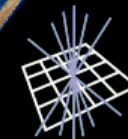
EGEE
Enabling Grids
for E-science

Scheduled = 21539
Running = 25374



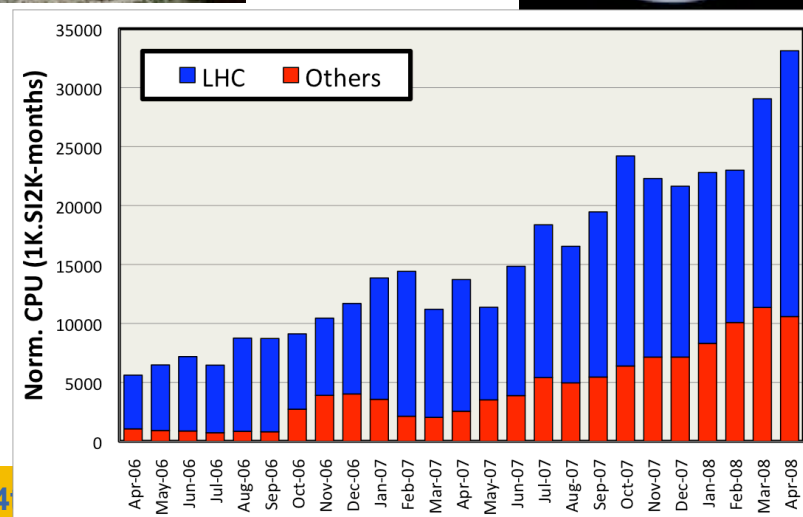
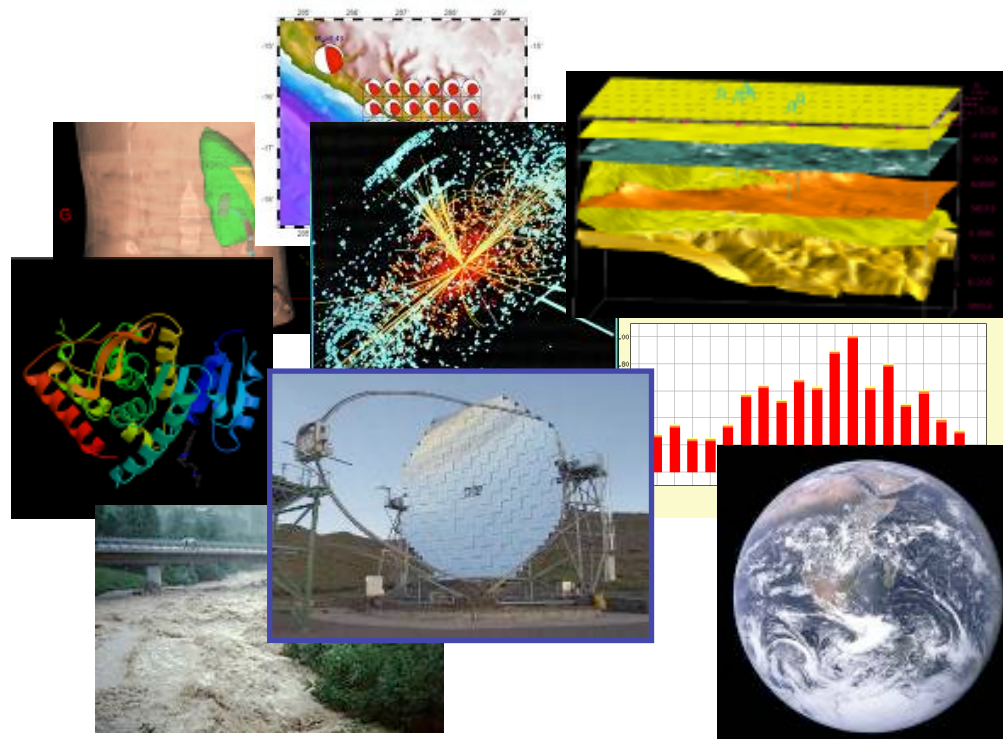
~280 sites
45 countries
>80,000 CPUs
>20 PetaBytes
>14,000 users
>250,000 jobs/day

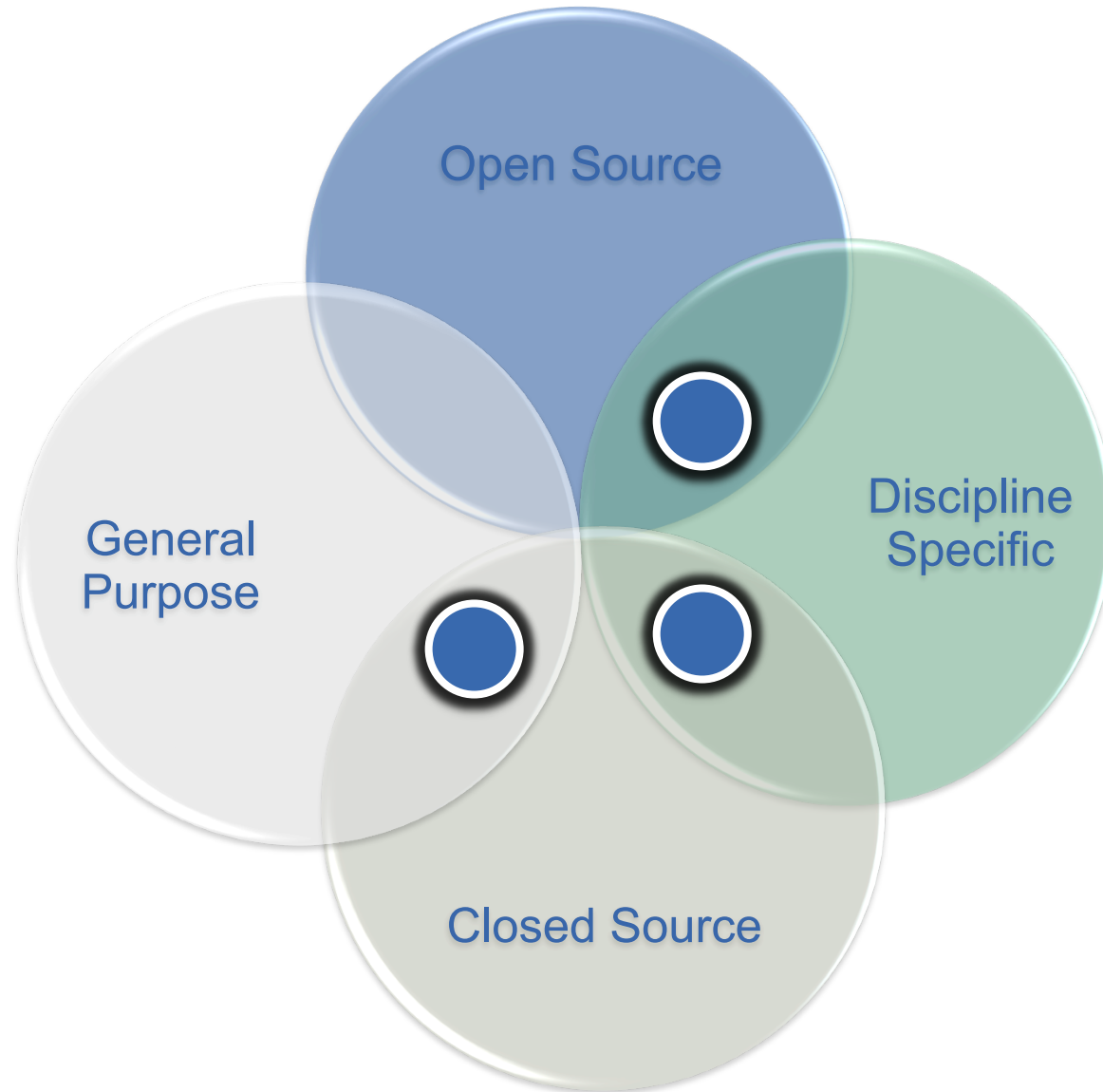
21:13:50 UTC



GridPP
UK Computing for Particle Physics

- **>260 VOs from several scientific domains**
 - Astronomy & Astrophysics
 - Civil Protection
 - Computational Chemistry
 - Comp. Fluid Dynamics
 - Computer Science/Tools
 - Condensed Matter Physics
 - Earth Sciences
 - Fusion
 - High Energy Physics
 - Life Sciences
 -
- **Much of the required software is developed within the communities themselves**
- **Also significant requirements for licensed software**





Discipline specific

- Computational Chemistry
- Life Sciences
- ✓ **Molecular and fluid dynamics**
 - AMBER
 - CHARM
 - NWCHEM
- ✓ **Molecular Docking**
 - FlexX

General Purpose

- Broad scope of disciplines
 - Finance
 - Mechanical Engineering
 - Optics
- ✓ **Compilers**
 - PGI
 - Intel
- ✓ **Development tools & Environments**
 - MATLAB

Computational Chemistry

- Gaussian
- Turbomole
- Gamess
- NAMD
- NWCHEM

Life Sciences

- FlexX
- Gold
- Amber 9

There is no “one-size fits all” approach

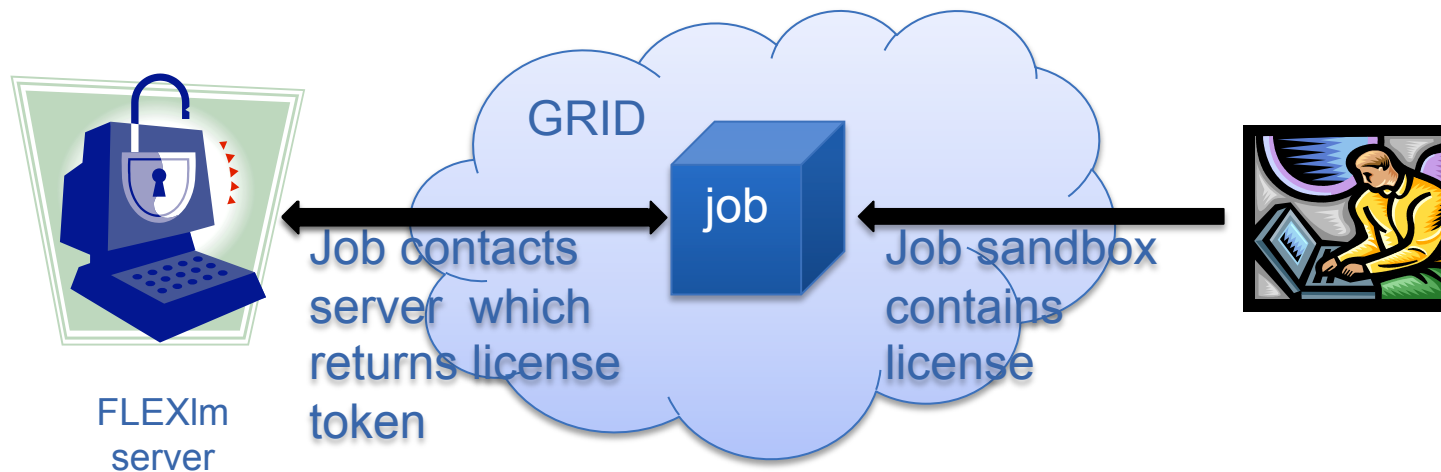
- Variable requirements per application
- Different community dynamics and potentials
- Broadly speaking we can identify two approaches:
VO-based and **VO-independent**

- **Dedicated VO for accessing a specific software**
 - Software is managed by the local admin. Installed in storage software area dedicated for the VO
 - User needs to enroll to the VO in order to access the software
 - Typically, one License per site

- **Examples: GAUSSIAN, NWCHEM**
- **Need to enroll to the VO to use the software**
 - Pros:
 - Simple to setup.
 - National and International scope
 - Cons:
 - No flexibility. Heavyweight approach. If more than to software needs to be accessed they have to be included in the same VO (no support for cross-VO interactions in gLite)

- **Server only**
 - The service provider buys the license and installs them on the host
 - CompChem example: GAUSSIAN, Turbomole
- **Client only**
 - User applies for individual license and can use the programs anywhere
 - CompChem example:
- **Server and Client**
 - Both server and clients have to own a license
- **Pros:**
 - Out-of-the box support for distributed environments
 - License per service provider or/and user
- **Cons**
 - No security. Cannot enforce grid-based authorization and access control mechanisms

- **FLEXIm used for FlexX and Gold**
 - Grid-friendly approach
 - Thousands of floating licenses
 - Server installed outside VO-scope
 - Need to setup multiple license servers for load-balancing
 - Licenses granted for specific drug-discovery challenges



- **PGI compilers**

- needed for meteorological models compilation MM5, BOLAM etc.
- Compiler installed in UI. Software statically compiled in order to run independently on the grid sites
 - Works but you loose potential optimizations for specific architectures

- **The case for INTEL compilers**

- Investigated possibility to install in HellasGrid infrastructure (6 clusters in Greece)
- License model inappropriate for Grid usage
 - License per CPU-> 800 cores in all six sites. Forbidding final cost

- **EGEE and The Mathworks have joined forces to integrate MATLAB Parallel Computing Products with gLite**
- **MDCS installed in two gLite sites in France and Greece**
 - One license per server
- **User submits jobs to gLite sites through regular MATLAB GUI Environment (e.g. from a Windows workstations)**
 - One PCT license per user.
 - License travels together with the job to the server (CE).
 - User responsible for her/his own license
- **Currently in trial phase**
 - Various participating research teams from different disciplines

- “So far the model where each user has to attach license file to every job seems to be the best solution for the grid.”
- **Advantages:**
 - No need to create VO for specific software
 - Unlimited number of software packages can coexist in single VO. Site admin applies the same permissions setup for them
 - Several packages can be used during job run
 - Easy code protection – user without license won’t be allowed to use the software
 - Easy to maintain the software – you have to just set common interface on each site
- **Disadvantages**
 - User has to remember to attach the license file or several files if he/she plan to use more than one package
 - Necessity to have license for certain software on each site supporting it.
- **Issues, which need to be solved in a near future:**
 - Where to store the software (grid or sites)
 - How to protect binaries against unauthorized access in case of grid storage

- **EGEE infrastructure serves a large number of scientific communities**
 - Need for both discipline specific and general purpose licensed software
 - Demand for licensed software at the moment comes mainly from CompChem and LS communities
- **Technically the VO-based approach is easy to apply but inflexible**
- **Software vendors should consider grid-oriented licensing schemes**
- **Close collaboration with Mathworks to provide MATLAB environment on the Grid**
 - Negotiating a grid-friendly LM approach

- **Special thanks to:**
 - Mariusz Sterzel,
 - Jean Salzemann
 - Horst Swichtenberg
 - Vincent Breton