



Enabling Grids for E-science

gUSE: grid User Support Environment

Peter Kacsuk, Krisztian Karoczkai, Andras Schnautigel, Istvan Marton, Gabor Herman
MTA SZTAKI

www.lpds.sztaki.hu

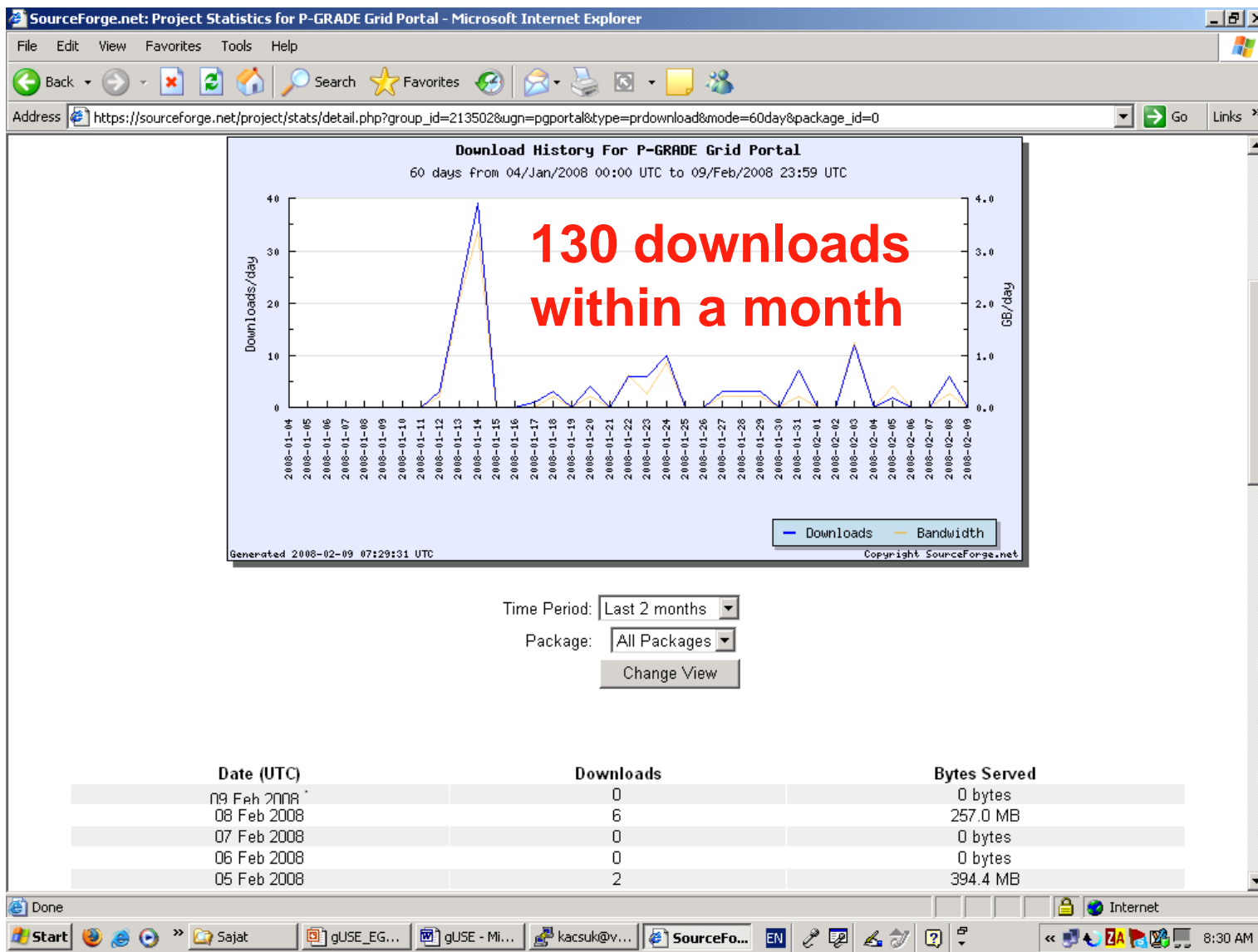
www.eu-egee.org



- **Motivations**
 - Lessons learnt from P-GRADE portal
 - Lessons learnt from accessing production Grid infrastructures
 - Lessons learnt from providing multi-grid service
- **The service-oriented architecture of gUSE**
- **Services in gUSE**
- **Workflow concept of gUSE**
- **Parameter sweep support of gUSE**
 - CancerGrid
- **Usage of gUSE**
 - EDGeS
- **Conclusions**

- **Popular because it provides**
 - Easy-to-use but powerful workflow system (graphical editor, wf manager, etc.)
 - Easy-to-use **parameter sweep** concept support
 - Easy-to-use MPI program execution support
 - Multi-grid/multi-VO access mechanism (job submission grid interoperability at workflow level) for LCG-2, gLite and GT2
- **Its extension with GEMLCA enables**
 - The usage of **legacy codes** as grid-enabled services
 - The usage of service/job repository
 - Access to SRB and OGSA-DAI
 - Multi-grid/multi-VO access mechanism for LCG-2, gLite, GT2 and **GT4**
 - Data management level of grid interoperability

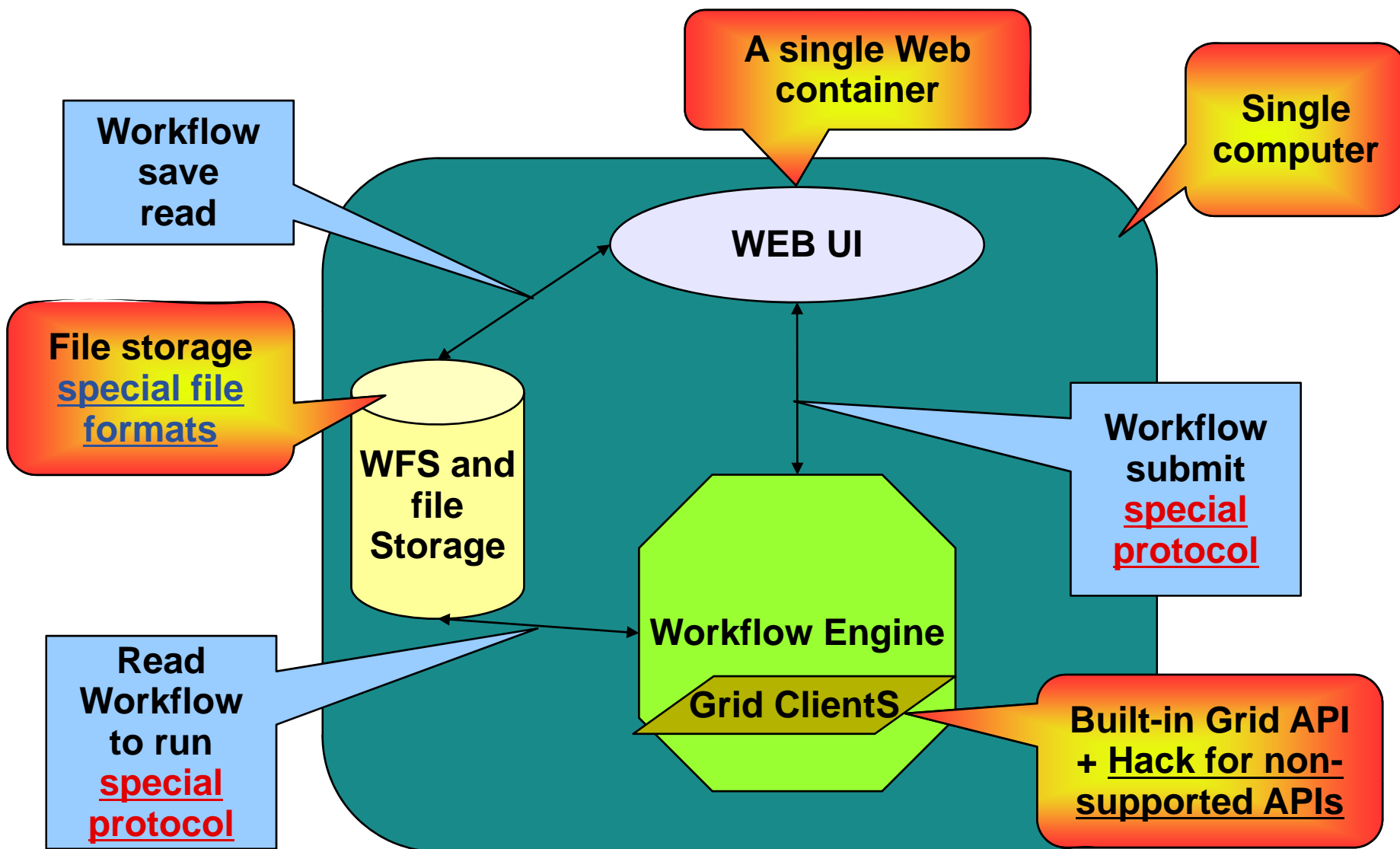
- **It has been used in many EGEE and EGEE-related VOs:**
 - GILDA, VOCE, SEE-GRID, BalticGrid, BioInfoGrid, EGRID, etc.
- **It has been used in many national grids:**
 - UK NGS (a GT2-based grid), Grid-Ireland, Turkish Grid, Croatian Grid, Ukrainian Grid, etc.
- **It has been used as the GIN VO Resource Testing Portal**
- **It became OSS in the beginning of Januar 2008:**
<https://sourceforge.net/projects/pgportal/>



- **Restricted workflow capabilities**
 - No cycle construct, no if-then-else, no embedding
- **Static parameter sweep capabilities**
 - PS can not be used inside a workflow
- **Single user view**
 - Too simple for IT people
 - Too complicated for end-users
- **Lack of collaborative tools supporting user communities**
- **Monolithic architecture and as a result problems with**
 - Scalability:
 - simultaneous number of jobs in the range of 100s
 - simultaneous number of users in the range of 30-50
 - Adaptivity: difficult to adapt to new grid services

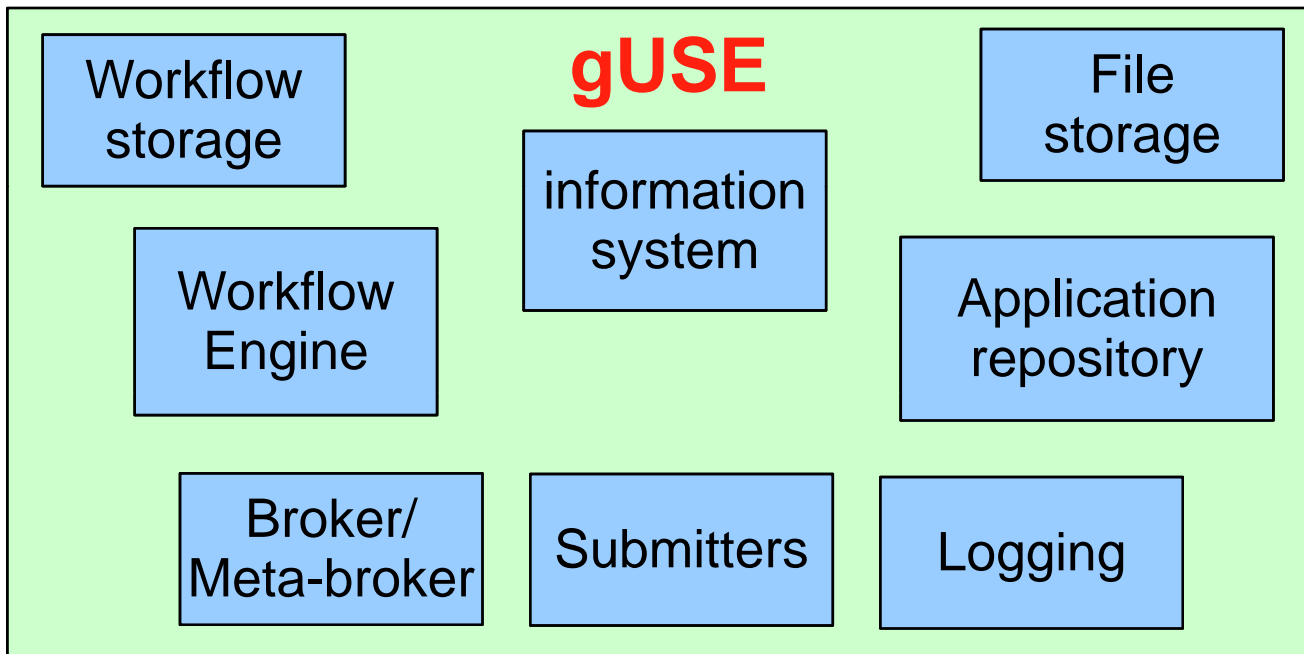
- **Production Grids do not enable you to modify anything, just use their services (no matter they are good or bad)**
- **Usually they provide basic grid services**
- **The user should construct higher level services**
- **However, if you do not want to be locked with one particular grid the user-written service should be interoperable with many basic grid services provided by different grids**

- **We wanted**
 - to overcome the limitations of the current P-GRADE portal
 - To create a set of high-level grid services that can be used with many different grids
- **Therefore we have defined a new service-oriented grid layer that can be deployed**
 - on a single machine
 - on a cluster
 - on different grid sites as Web Services
- **Performance comparison**
 - P-GRADE portal monolithic architecture: 100-200 jobs
 - WS-PGRADE/gUSE SOA architecture: **10.000 jobs**



Graphical User Interface: **WS-PGRADE**

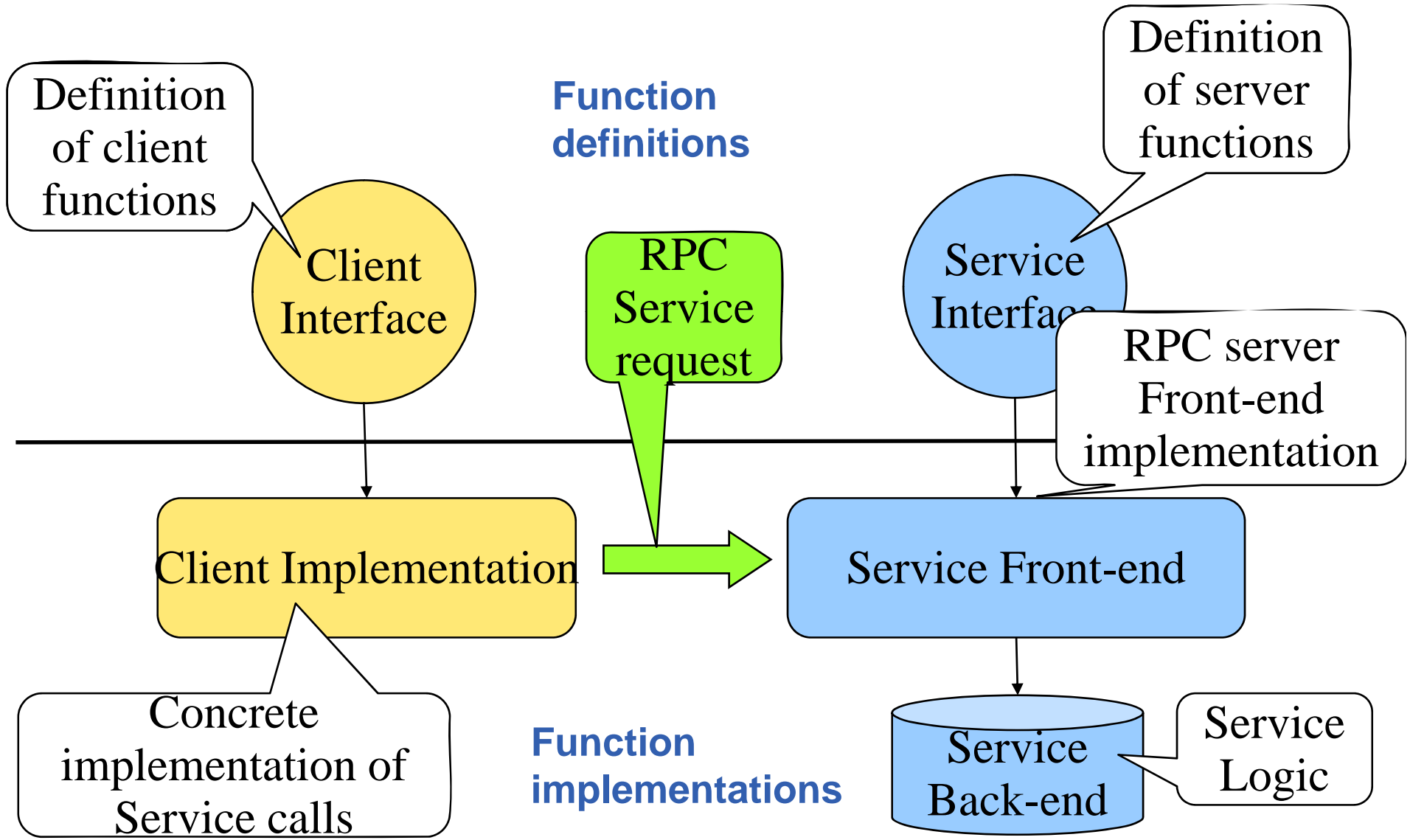
*Gridsphere
portlets*

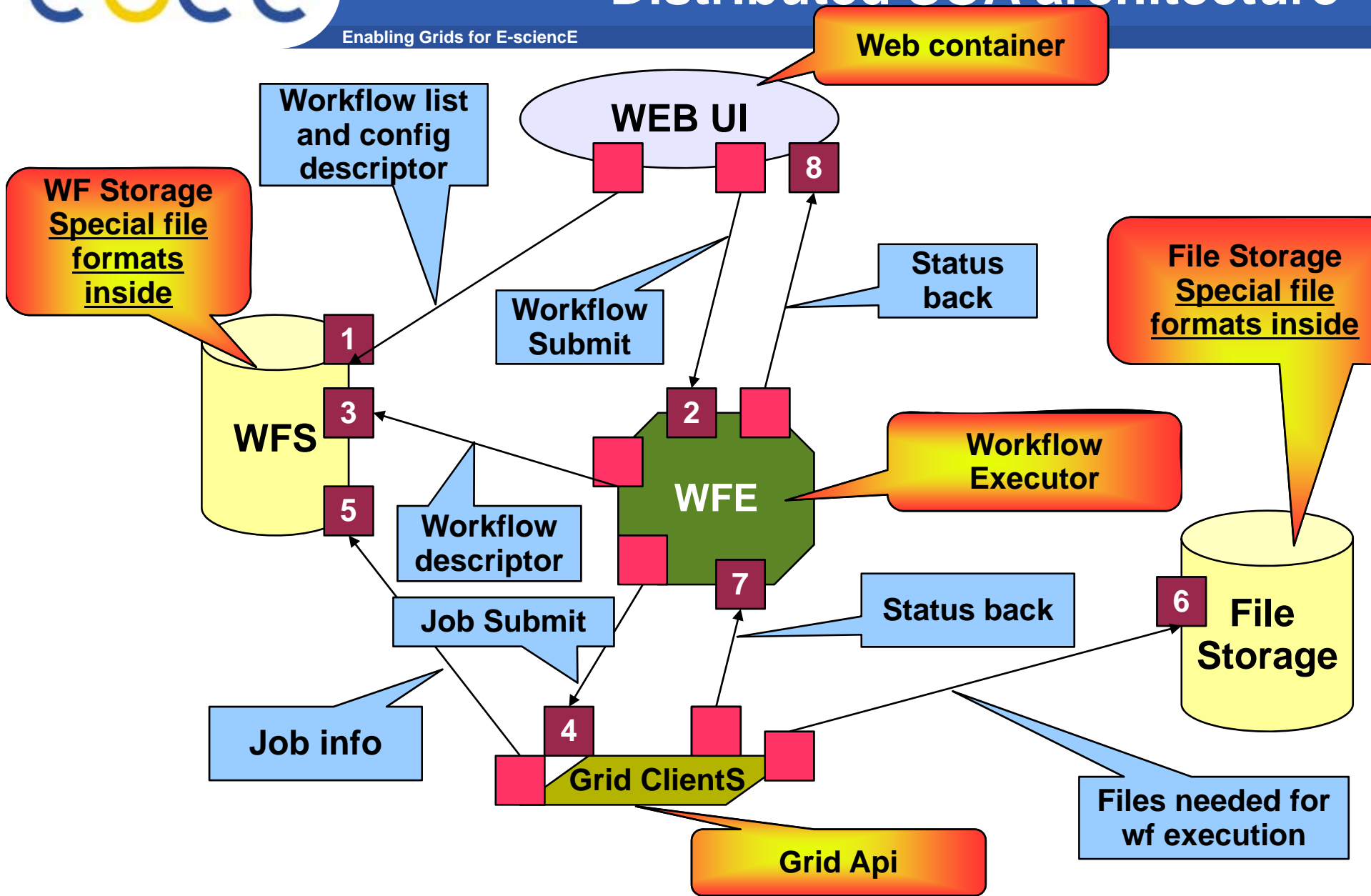


*Autonomous
Services:
high level
middleware
service layer*

gLite resources, Globus resources and Web services

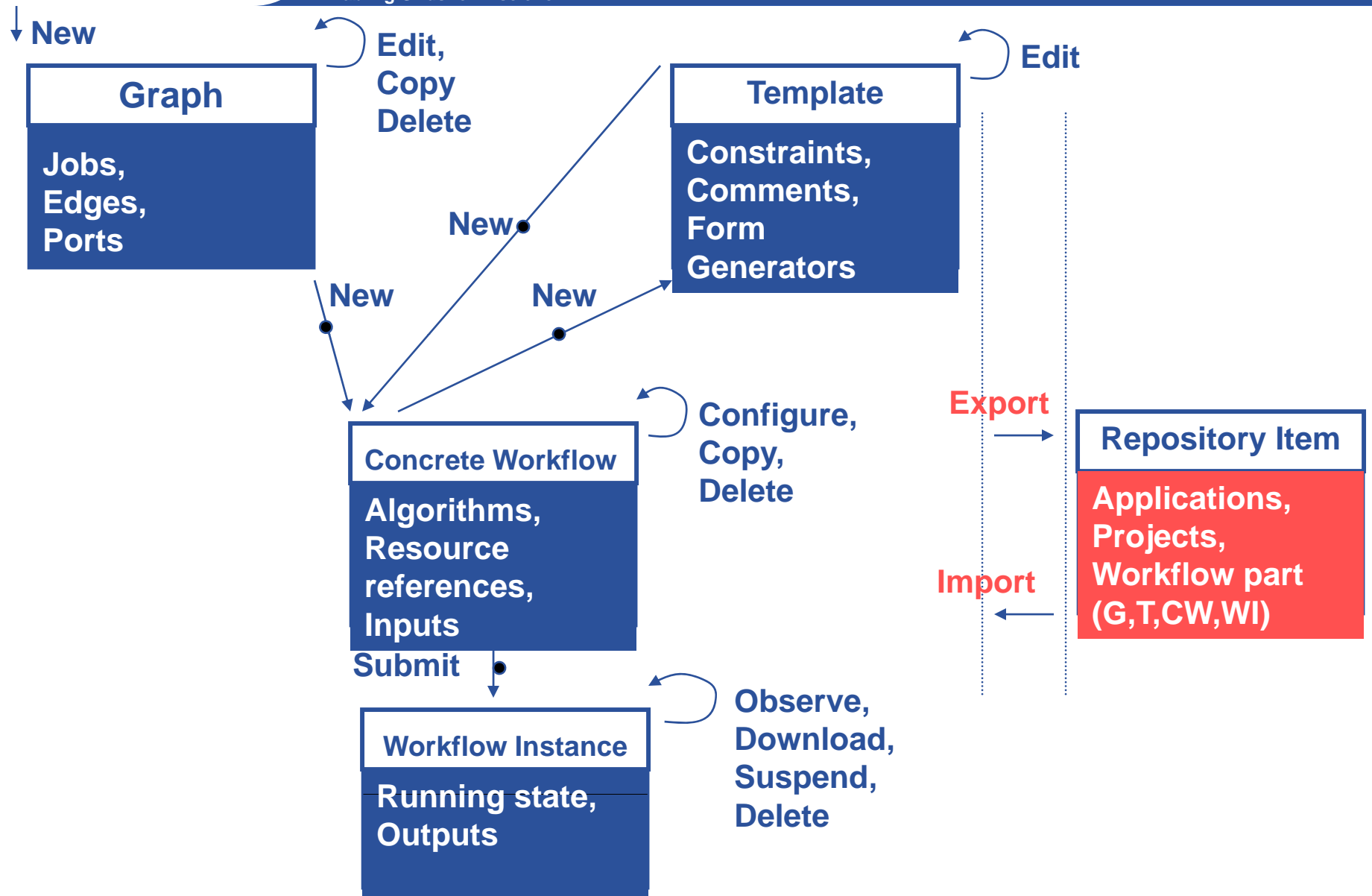
*gLite or Globus
or Web service:
low level
middleware
service layer*





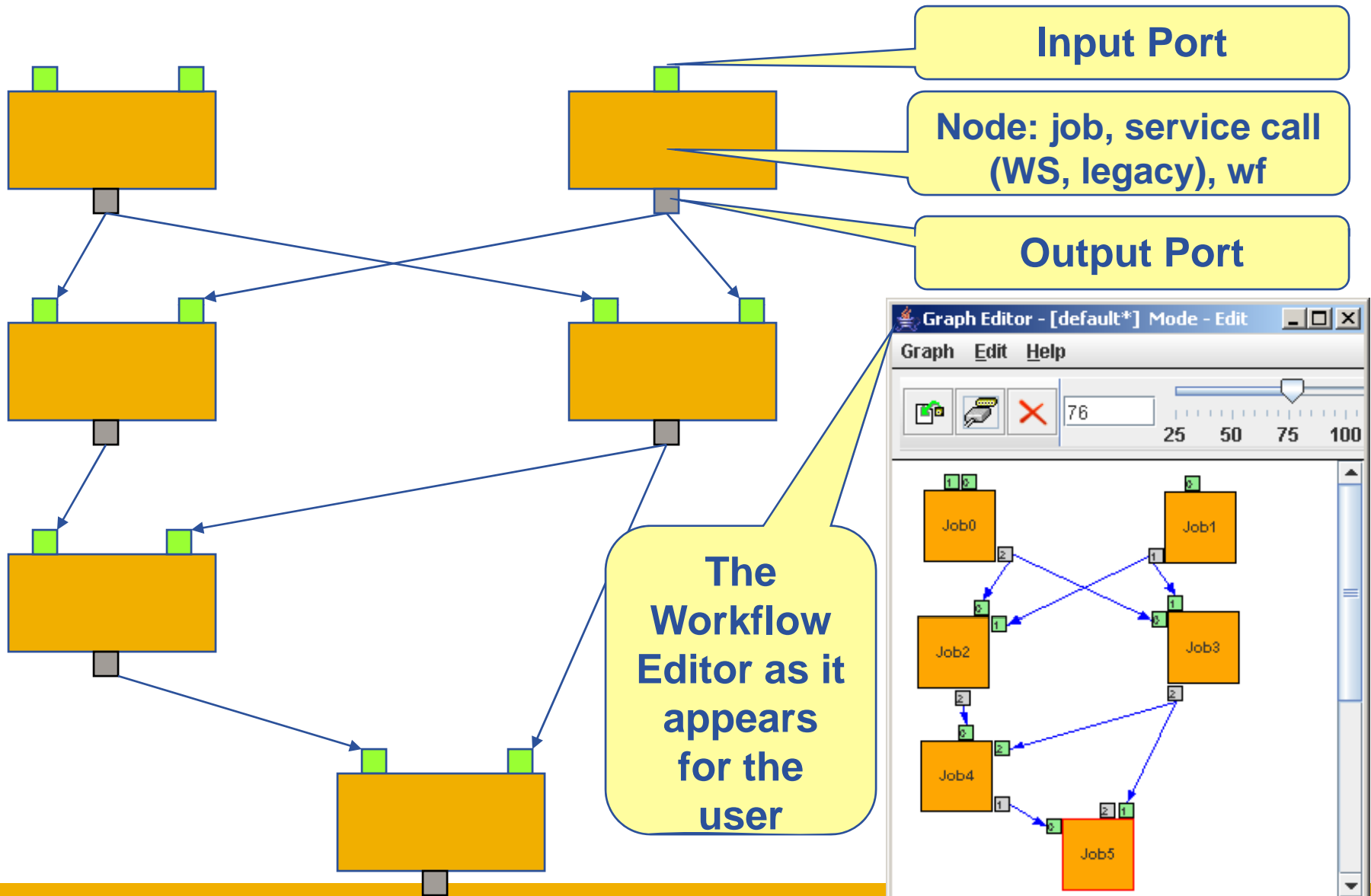
- **Users of gUSE can be either**
 - grid application developers
 - or end-users.
- **Application developers can develop sophisticated workflow applications where workflows can be**
 - embedded into each other at any depth
 - recursive workflows are allowed
 - gUSE supports the following workflow types
 - graphs (abstract workflows)
 - workflow templates
 - concrete workflows
 - workflow instances
- **Parametric sweep nodes and normal nodes can be used in a mixed way.**

- **Application developers can**
 - publish
 - incomplete wf applications (projects), wf parts (templates, graphs, concrete wf, wf instances) into a **workflow repository** for the use of other developers
 - ready-to-run wf applications for end-users
 - import workflows from the repository and can continue the work on them even if they were published by other developers
- **End-users can**
 - import ready-to-run wf applications from the repository
 - execute ready-to-run wf applications imported from the repository based on a simplified portal interface hiding grid details
- **Grid is exposed only for application developers.**

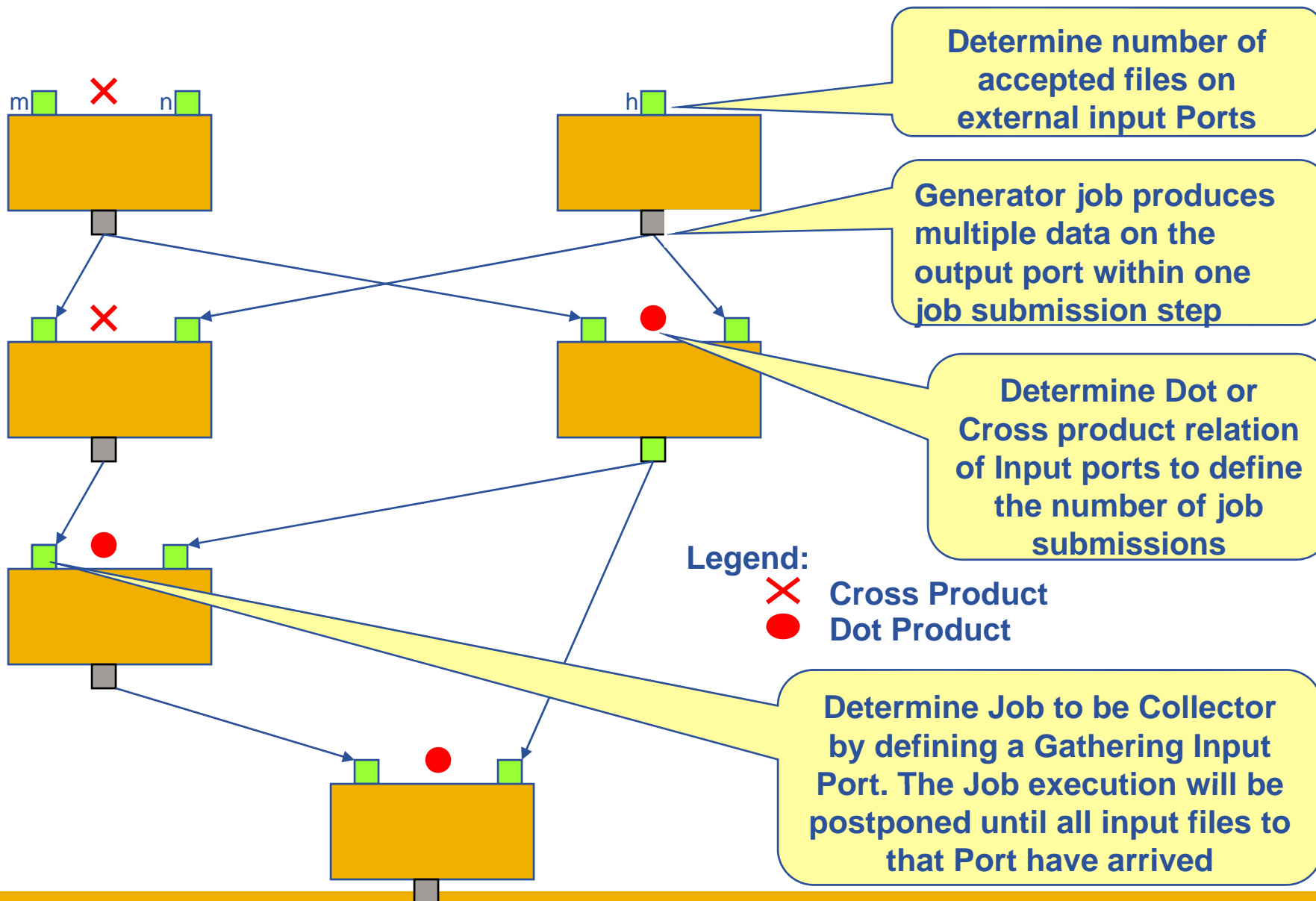


- **The workflow concept of gUSE is much more flexible than P-GRADE portal and many other workflow systems**
- **Its DAG topology is extended with**
 - embedded WFs
 - recursive embedded WFs
 - parameter sweep nodes
 - conditional control mechanism
 - special workflow starting control mechanisms based on
 - external events or
 - periodic timing

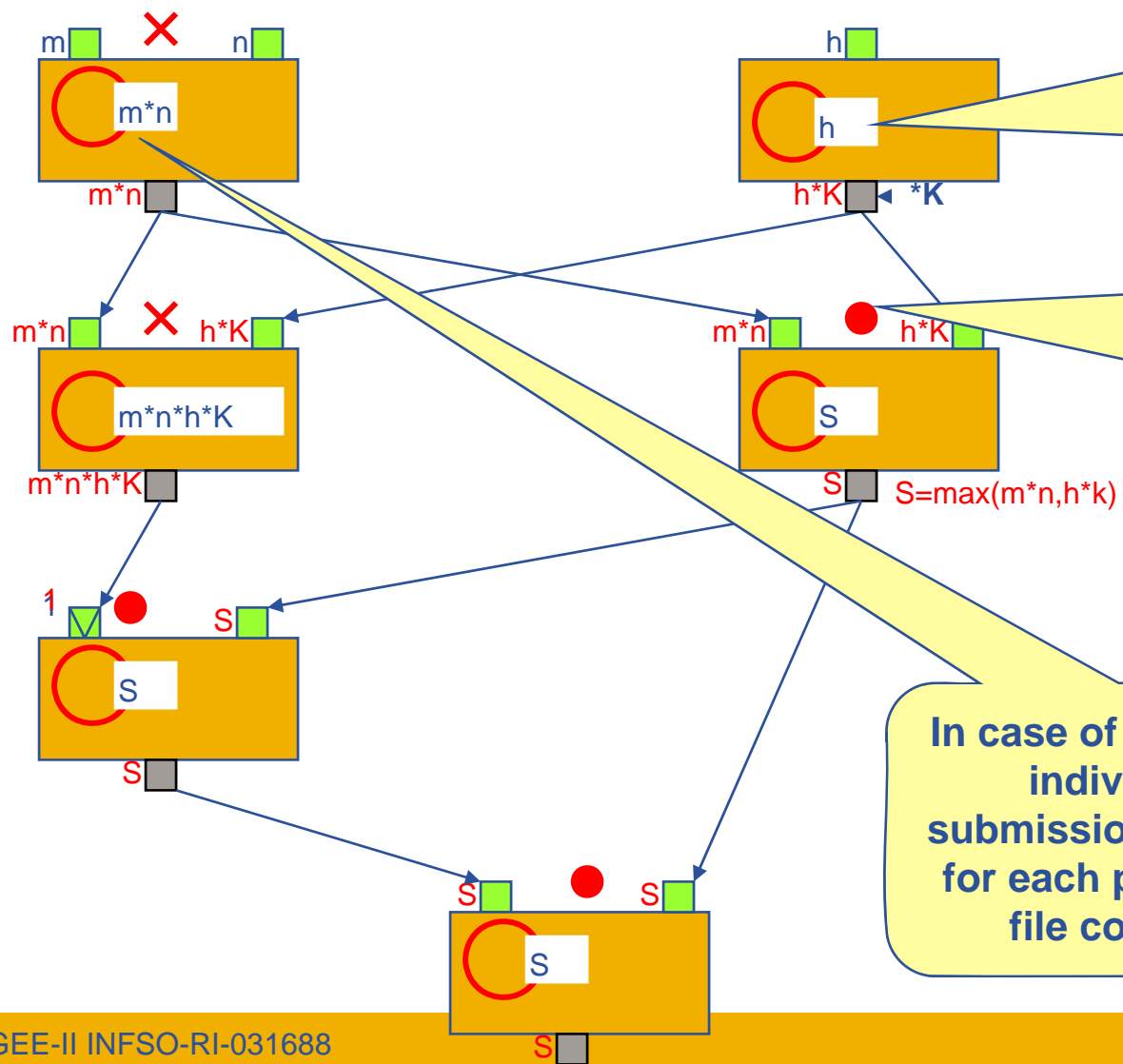
Workflow Graph: Overview



Configuring the Workflow: Overview



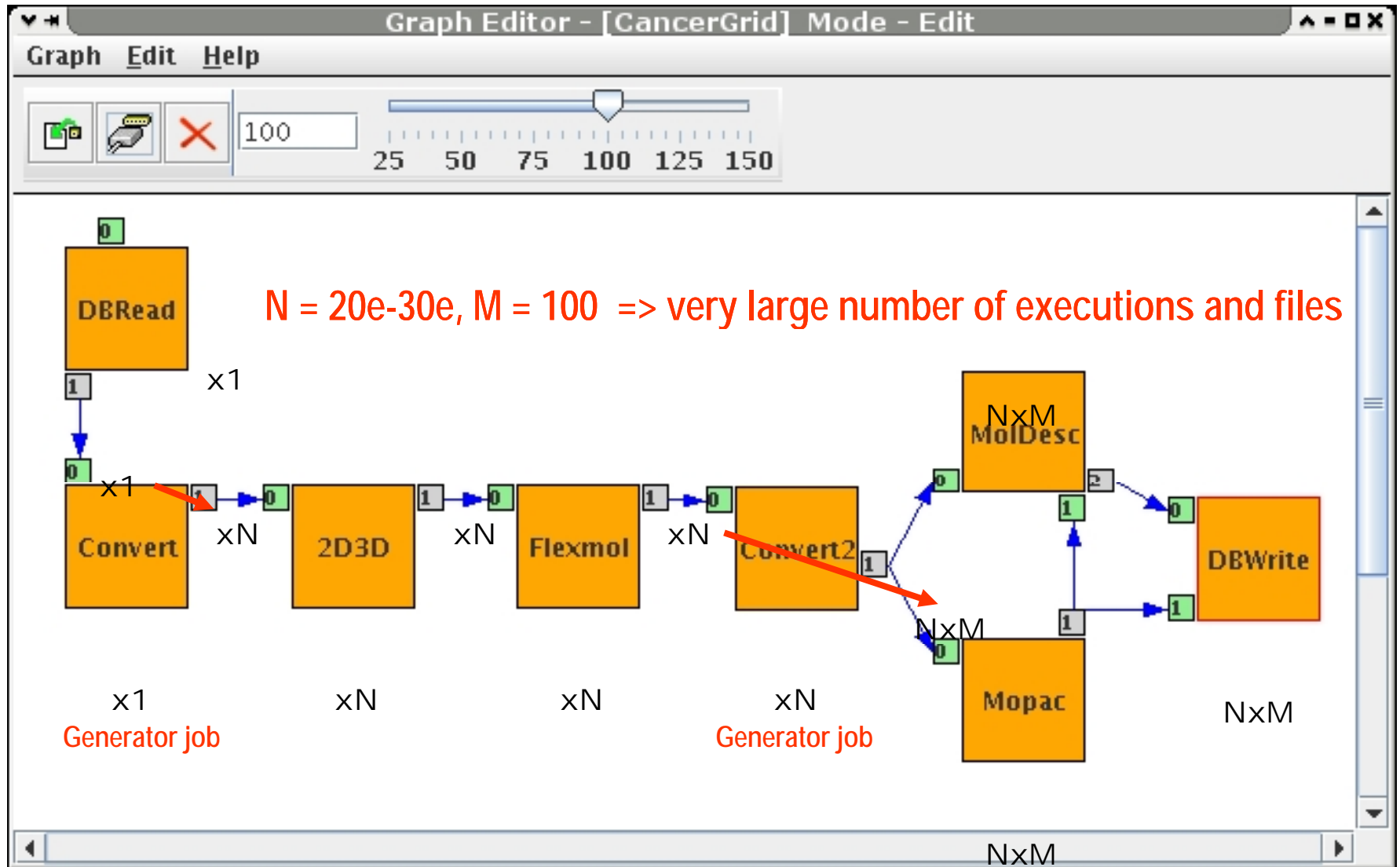
Animation the number of generated output files



In case of Generator job the number of job submissions may differ from the number of files on Output Ports

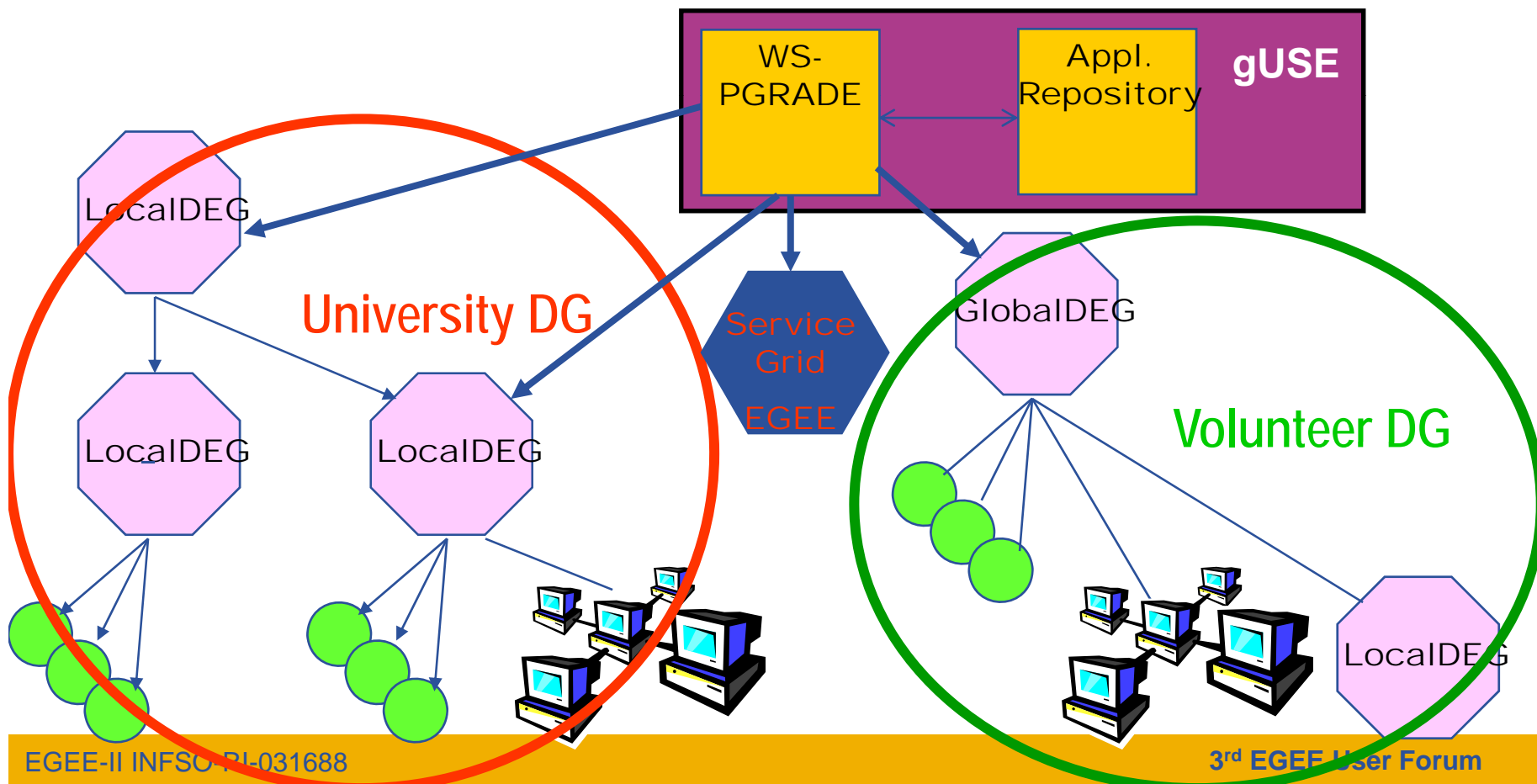
In case of dot product the Job is submitted with input files having a common index number in each input Ports

In case of cross product individual Job submission is generated for each possible input file combination



- **gUSE supports:**
 - grid interoperability
 - workflow interoperability
- **gUSE can easily be connected to any known grid middleware. It is already connected to GT2, GT4, LCG-2, gLite and WS based grid systems**
- **gUSE can also be connected to local systems like clusters or supercomputers**
- **It contains a built-in grid broker that can automatically distribute the jobs of a workflow into any of the connected grids**
- **It can use other grid brokers like the gLite broker or GridWay**

- EDGeS: Enabling Desktop Grids for e-Science
- To integrate EGEE with Desktop Grids
- gUSE can provide the transparent access of EGEE and DGs





- **P-GRADE portal** and **gUSE/WS-PGRADE** represent a family of user support products
- They support the whole range of user types:
 - **Novice application developers**: 1st generation P-GRADE portals
 - **Advances application developers**: 2nd generation WS-PGRADE portal developer view
 - **End-users without grid knowledge**: 2nd generation WS-PGRADE portal end-user view

- **P-GRADE**
 - Parallelizing applications for clusters and grids
- **P-GRADE portal**
 - Creating simple workflow and parameter sweep applications for grids
- **P-GRADE/GEMMLCA portal**
 - Creating workflow applications using legacy codes and community codes from repository
- **gUSE/WS-PGRADE**
 - Creating complex workflow and parameter sweep applications for clusters, service grids and desktop grids
 - Creating workflow applications using embedded workflows, legacy codes and community workflows from workflow repository

- **gUSE solves all the limitation problems of P-GRADE portal:**
 - Implementation of gUSE is highly scalable, can be distributed on a cluster or even on different grid sites.
 - Stress tests show that it can simultaneously serve thousands of jobs
 - Its workflow concept is much more expressive than in P-GRADE portal (recursive wf, generic PS support, etc.)
 - Its user interface called as WS-PGRADE provides a graphical workflow editor that is much faster than the one in P-GRADE portal
 - gUSE provides a workflow repository and its use by end-users and application developers
 - gUSE solves grid interoperability at workflow level
 - among service grids
 - between service grids and desktop grids (see EDGeS project)

- First version was demonstrated at **SC'07**
- First version will be released in **March 2008** with full support for EGEE, GT2 and GT4
- Second version will be released in **July 2008** with full support for desktop grids
- Third version solving interoperability between EGEE and desktop grids will be released by **SC'08**