



Application Porting for EDGeS

Tamas Kiss – University of Westminster

Leader of the EDGeS Application Support Service

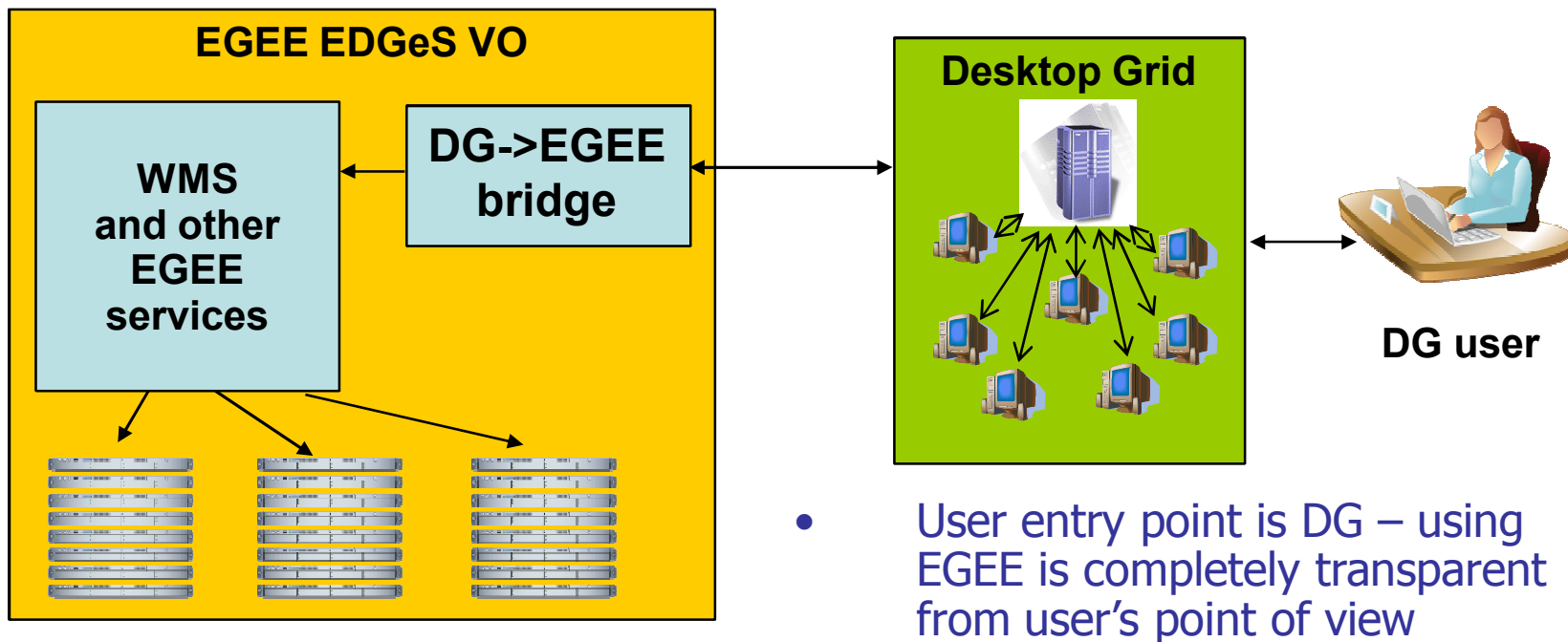
E-mail: kist@wmin.ac.uk

What is an EDGeS application?

EDGeS application

- An EDGeS application is capable using both **Service (EGEE)** and **Desktop Grid** resources
- EDGeS application scenarios:
 1. Application runs on DG and uses EGEE resources via the DG to EGEE bridge
 2. Application submitted to EGEE and uses DG resources via the EGEE to DG bridge
 3. Application uses both SG and DG resources via an external scheduling and job submission system (not through the EDGeS bridges)

Scenario 1 – DG to EGEE via bridge



Scenario 1.1: application already runs on DG

Scenario 1.2: application needs to be ported to DG

Scenario 1.1 Application already runs on DG

Porting to EDGeS:

1. Develop **Linux version** of your client application
2. Test **client application** on EGEE
 - Use the EDGeS Development VO
 - Use other EGEE VO you have access to
3. Test your application via the **DG->EGEE bridge**
 - Use the EDGeS development infrastructure
4. Write **EDGeS Test report**
 - Template downloadable from EDGeS website
5. Deploy your application on **EDGeS production infrastructure**
 - Send test report to EDGeS VO Admin for approval
 - Deploy your application in production
 - On a DG already connected to EDGeS VO
 - Connect your DG to EDGeS VO (details yesterday)

Which DG applications are suitable for EDGeS?

- **If an application is capable to run on a heterogeneous DG then it is typically capable to run in EGEE too.**
 - **Public DG** applications are almost certain to run on EGEE resources too.
 - **Local DG** applications may require more thorough testing and some potential fine-tuning (may suppose less heterogeneous infrastructure).
 - **E.g.:** may require large amount of memory available on local machines but not on every EGEE site.

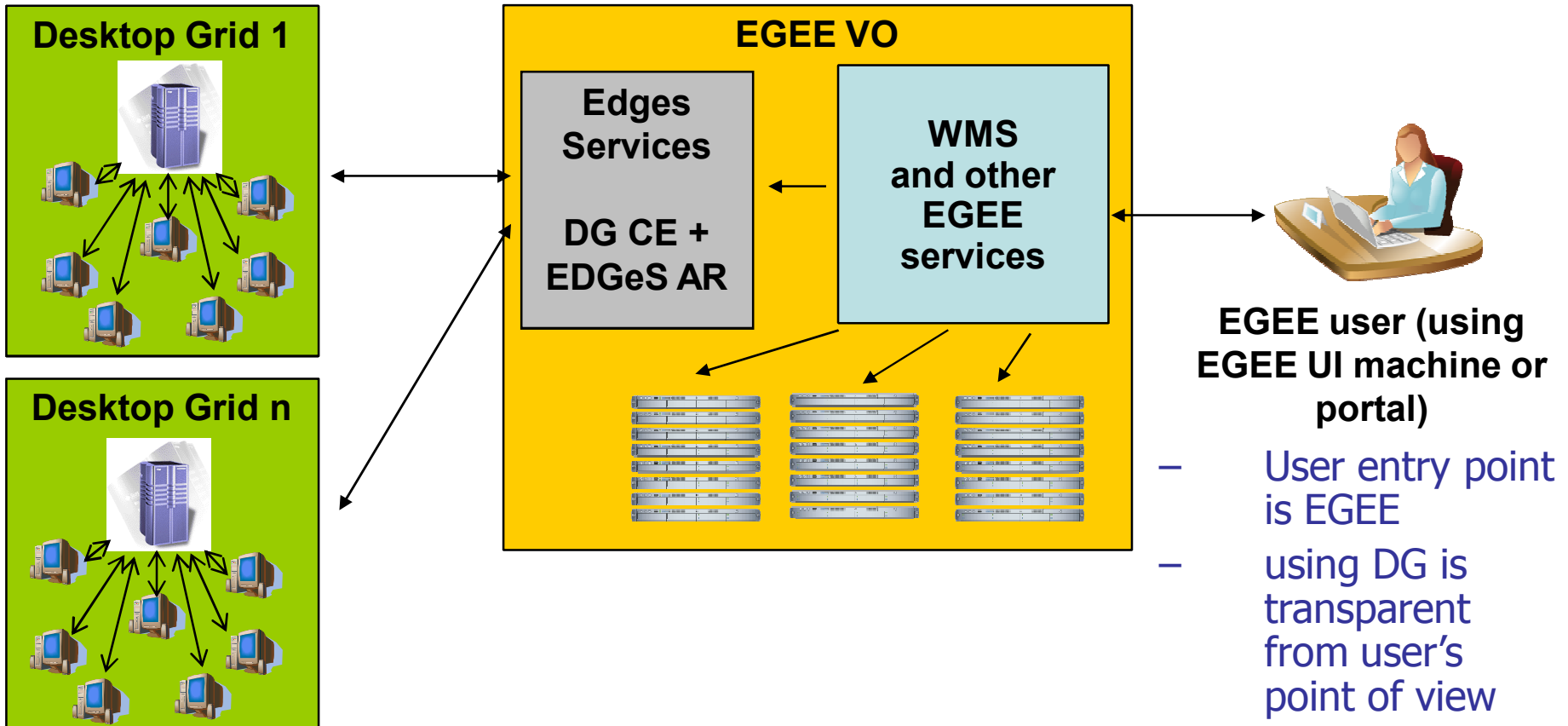
Porting to EDGeS:

1. Develop and test DG version of application
 - Use EDGeS Development DG infrastructure
 - Use your own development infrastructure
2. Deploy application on target DG system
3. Follow every step from scenario 1.1

EDGeS application examples:

- **Scenario 1.1:**
 - **ISDEP fusion application** – previously run on Ibercivis DG
 - **Cellular automata based laser dynamics** – DG version already used by University of Seville
- **Scenario 1.2:**
 - **Video stream analysis (VISAGE)** – DG version developed within EDGes by Correlation Systems
 - **Extraction of x-ray diffraction profiles** – algorithm parallelized and DG version developed by University of Westminster and University of Extremadura

Scenario 2 – EGEE to DG via bridge



Scenario 2.1: application already runs on EGEE

Scenario 2.2: application needs to be ported to EGEE

Porting to EDGeS:

1. Develop **DG version** of application
2. Test **DG version** and write **test report**
 - Use the EDGeS Development DG
 - Use your own development DG
 - Test report template downloadable from EDGeS website
3. Have your application **validated**
 - Validation is done by EDGeS Validation Team
 - Outcome: validation document
 - Validation includes tests via the EDGeS EGEE->DG test bridge
4. Publish your application in **EDGeS Application Repository**
 - EDGeS Services (AR and EDGeS CE) should be deployed by your EGEE VO as pre-requisite
 - DG Admins can download your application from EDGeS AR if they are ready to support it

Which EGEE applications are suitable for EDGeS?

- **EDGeS applications should run on both EGEE and DGs**
 - EGEE supports a much wider scale of applications than DGs
 - We should assure that the application runs on DGs
- **Requirements towards an EGEE application to be executable on DGs:**
 - **Parallelization:**
 - Only **master/worker** or **parameter sweep** parallelisation
 - **No MPI** or internal communication between worker nodes
 - Nodes can only use the results of other nodes **through the server**
 - **Data handling:**
 - small or medium-sized (max. 100 MB per worker) inputs and outputs
 - No shared data storage
 - No confidential data (sent down to potentially un-trusted worker)

Which EGEE applications are suitable for EDGeS?

- To achieve good performance
 - the execution time of individual jobs should be
 - **Minimum:**
 - over 10 minutes (otherwise the overhead caused by the DG will reduce the performance)
 - **Maximum:**
 - less than 30-60 minutes (if more, application level checkpointing is required to avoid loss of computation caused by user interventions)
 - the execution of individual jobs should take around the same amount of time (better scheduling, less load on the server)

Which EGEE applications are suitable for EDGeS?

Other limitations

- **Programming languages**

- APIs available primarily in C and C++ (there is a binding for FORTRAN)
- GenWrapper or BOINC wrapper technologies are supporting any other legacy applications

- **Operating systems**

- Depends on the DGs where the application will run
 - windows version may be required to utilise larger number of resources

Applications suitable for EDGeS

- Parameter sweep
 - The size of inputs and outputs should be under 100 MB per slave
 - If they are over 100 MB, multiple data servers and load balancing is necessary
- Master/worker
 - The same size limits apply
- Multi-threaded applications (worker applications that start multiple threads to utilise multi-core CPUs)
- Applications using GPUs

Porting to EDGeS:

1. Develop and test EGEE version of application
 - Use EDGeS development infrastructure (EDGeS Development VO)
 - Use any other EGEE VO
2. Follow every step from scenario 2.1

EDGeS application examples:

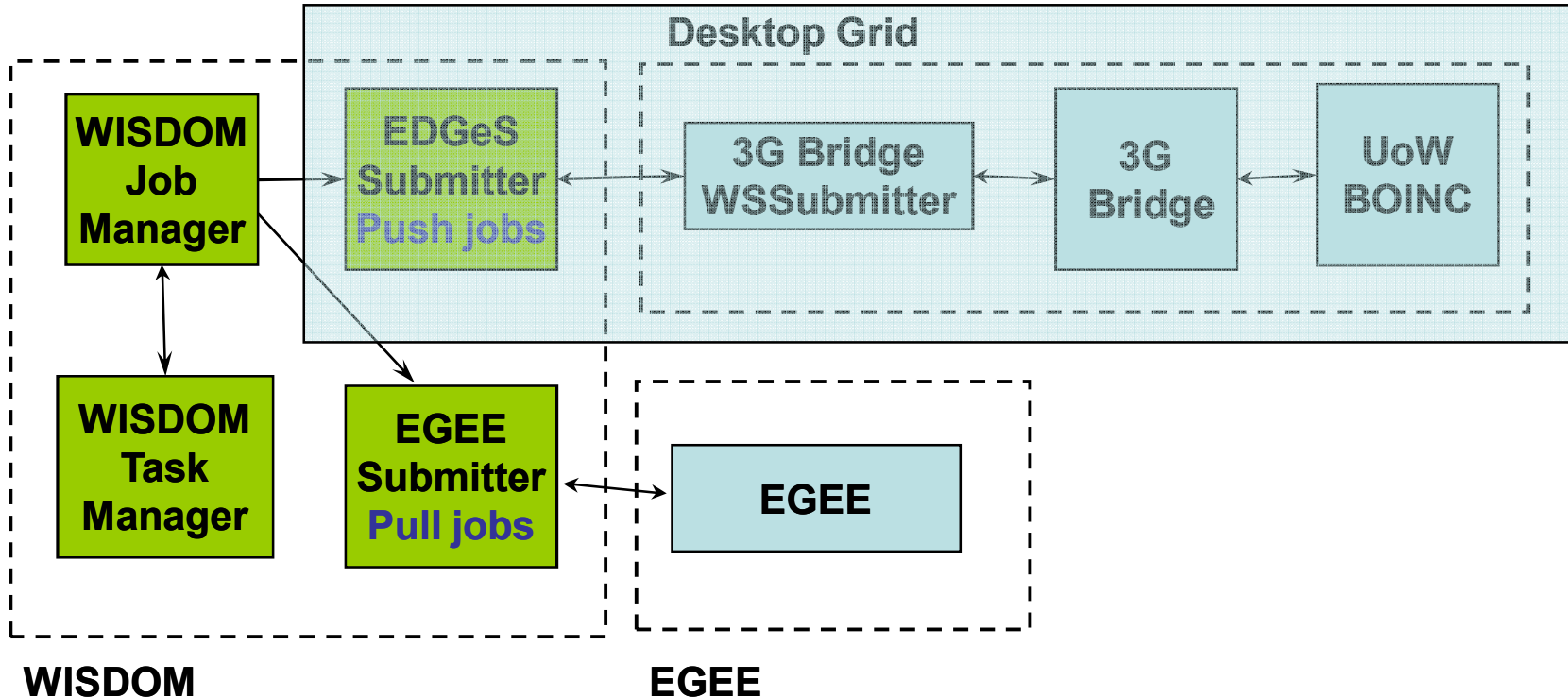
- **Scenario 2.1:**
 - **EMMIL E-marketplace application** – run on SEE-Grid already
 - **VisIVO - Visualisation Interface to the Virtual Observatory** – EGEE application by Astrophysical Observatory of Catania, currently ported to DG by UoW
- **Scenario 2.2:**
 - **3D video rendering** – application currently ported to EGEE by UoW (however, DG version already exists)

Scenario 3 – SG/DG resources but not through EDGeS bridges

- **Using external job submission and scheduling system to submit jobs to both EGEE and DG resources**
 - **G-USE portal supports this scenario by default**
 - E.g.:
 - CancerGrid application uses EGEE and DG resources via the g-USE portal (but not through the bridges - see presentation on Friday afternoon)
 - **Could be justified with specific user requirements**
 - E.g.:
 - WISDOM project uses only pull jobs on EGEE that are unsuitable to be bridged to DG
 - **Both solutions use EDGeS technology: 3GBridge**

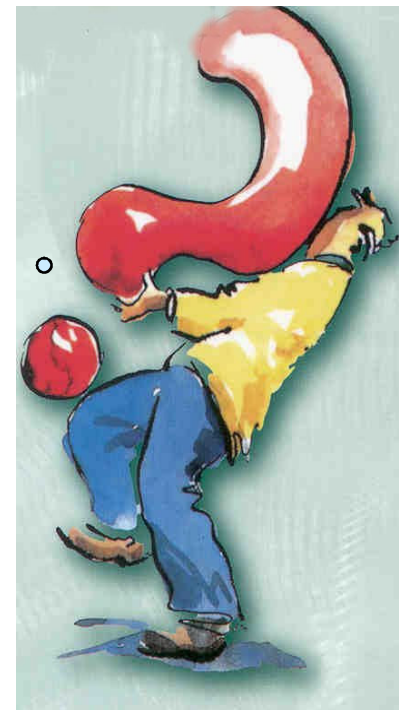
Scenario 3 – e.g. DG submitter for WISDOM

- **WISDOM:** Meta middleware to submit pull (pilot) jobs to EGEE
- **The DG submitter:**
 - Submit push (direct) jobs to the DG when EGEE resources are overloaded



Thank you for your attention ...

Any
questions?



For more information please visit the EDGeS Website:

Join the EDGeS User & Industry Forum here!

<http://www.edges-grid.eu/>

Please contact us if you need support in porting your application to EDGeS!

Email: kisst@wmin.ac.uk