



# SZDG, eCom4Com technology, EDGeS-EDGI in large

**P. Kacsuk**
**MTA SZTAKI**



The EDGI/EDGeS projects receive(d) Community research funding

# Outline of the talk

- SZTAKI Desktop Grid (SZDG)
- SZDG technology: eCom4Com
- EDGeS
- EDGI

# SZTAKI Desktop Grid global version

- **Main objectives:**
  - **Demonstrate the power of the Desktop Grid concept**
  - **Support Hungarian scientific applications**
  - **Introducing DG technology in Hungary**
- **Three steps for everybody to try and use the technology:**
  1. Donate one PC to test the client site
  2. Port application to the DG server and register PCs for that application
  3. Set up a DG server for the community (univ., company, city, etc.)
- SZTAKI helps in steps 2 and 3 as explained in detail at **http://www.desktopgrid.hu/**

MTA SZTAKI

SZTAKI DESKTOP GRID

BOINC

Number of users: **39066** Number of hosts: **86706**
Estimated performance of last 48 hours: **1738.814 GFlop/s**
Peak performance: **3.4 TFlop/s**

## About SZTAKI Desktop Grid

SZTAKI Desktop Grid is operated by the Laboratory of Parallel and Distributed Systems at the Hungarian Academy of Sciences, Budapest, Hungary.

The SZTAKI Desktop Grid and its applications are partly supported by the DEGISCO project. The work leading to these results has received funding from the **European Union Seventh Framework Programme** (FP7/2007-2013) under grant agreement n° 261556. The experts of the **International Desktop Grid Federation** provide further support for the SZTAKI Desktop Grid infrastructure, its applications, and its integration into the DEGISCO infrastructure.

## Join SZTAKI Desktop Grid

- Read our rules and policies
- This project uses BOINC. If you're already running BOINC, select Attach to Project. If not, download BOINC.
- When prompted, enter **http://szdg.lpds.sztaki.hu/szdg/**
- If you're running a command-line or pre-5.0 version of BOINC, create an account first.
- If you have any problems, get help here.

## Applications being run by SZTAKI Desktop Grid

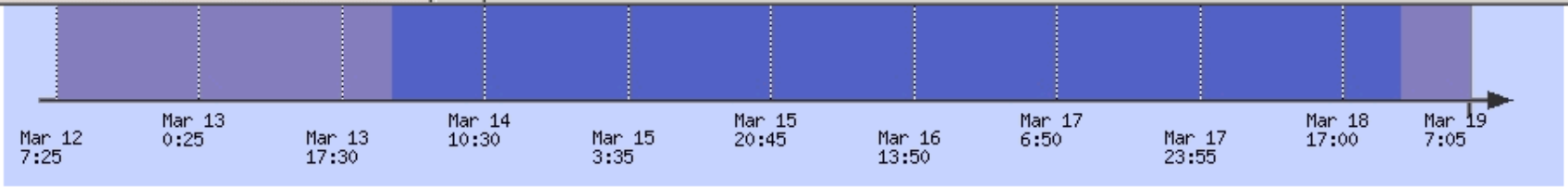
| Application | Field | | Launch |
|---|---|---|---|
| BinSys | Mathematics | Description Results | Jun, 2005 |
| UC-Explorer | Physics | Description Results N/A | Feb, 2009 |

## User of the day

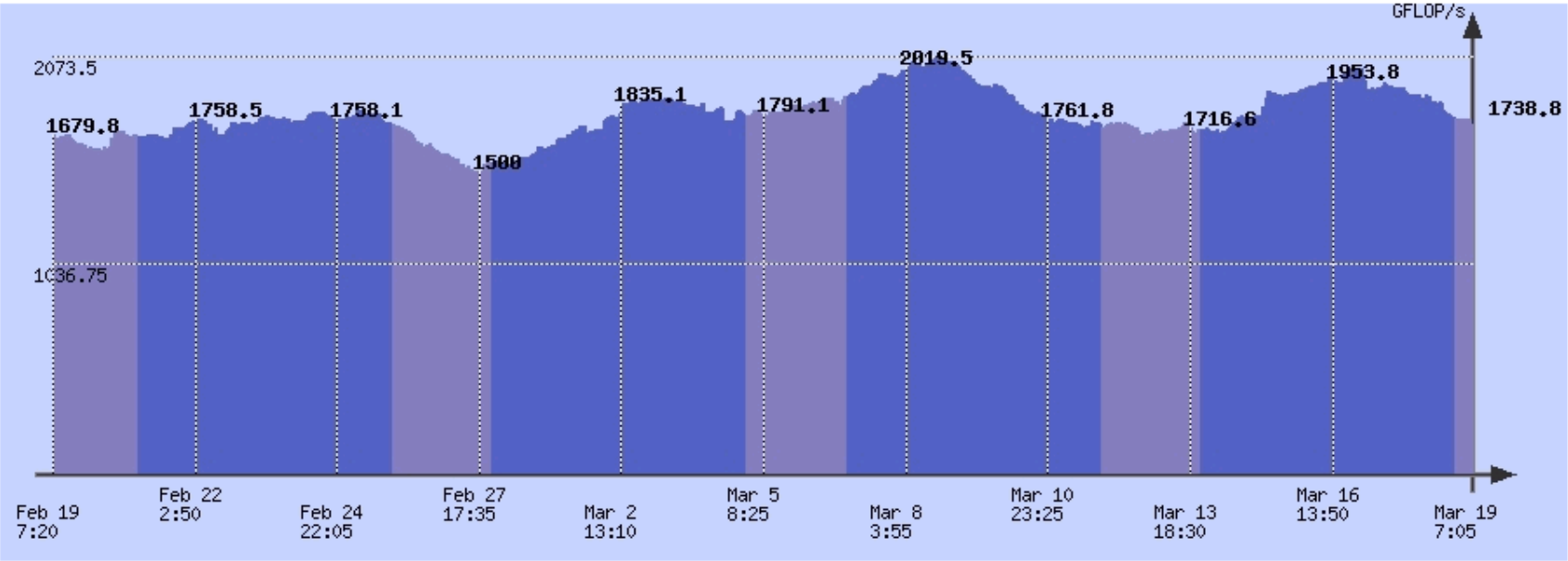
Blurf
Tech Support Agent for a Dialup/High-Speed ISP in Upstate NY.

Married. Have 2 King Charles Cavalier Spaniels (Isabella and Abigail)....

## News




Last 168 hours performance




Last 4 weeks performance

Home | My Account | Message Boards

# Lessons learnt from SZDG project

- BOINC is excellent to create volunteer DG projects
  - Reliable, stable, robust and scalable technology
- Drawback
  - Difficult to port applications to BOINC
  - There is no user interface through which end-users could easily launch applications with their parameters

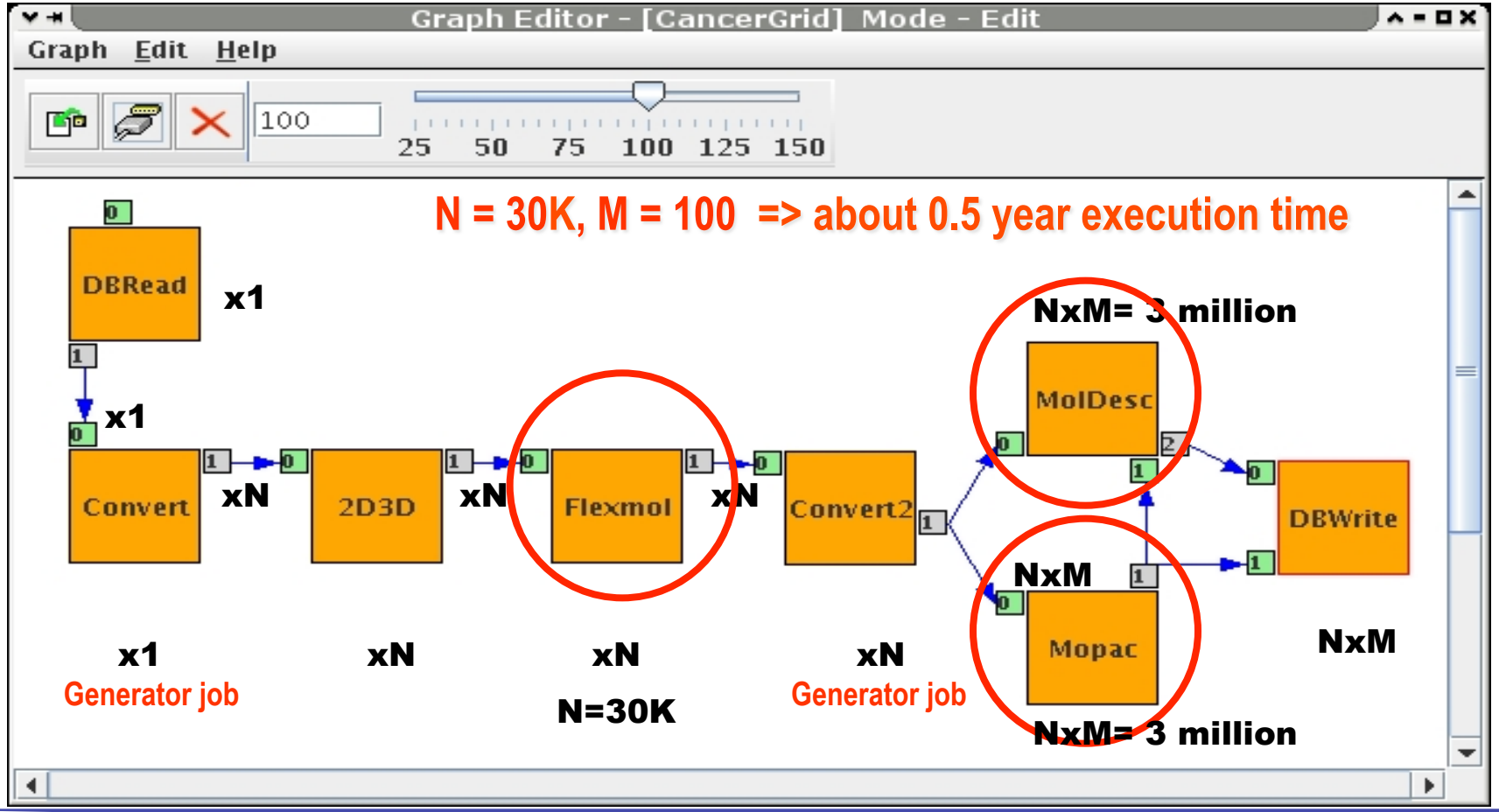
# Question: Can we use BOINC to build institutional DGs?

- Yes, but in this case we have to solve the following problems:
  - Abandon the credit system
  - Enable
    - Fast deployment
    - Fast application porting
    - Easy usage by end-users
    - Possible extension with
      - service grids
      - desktop grids
      - clouds

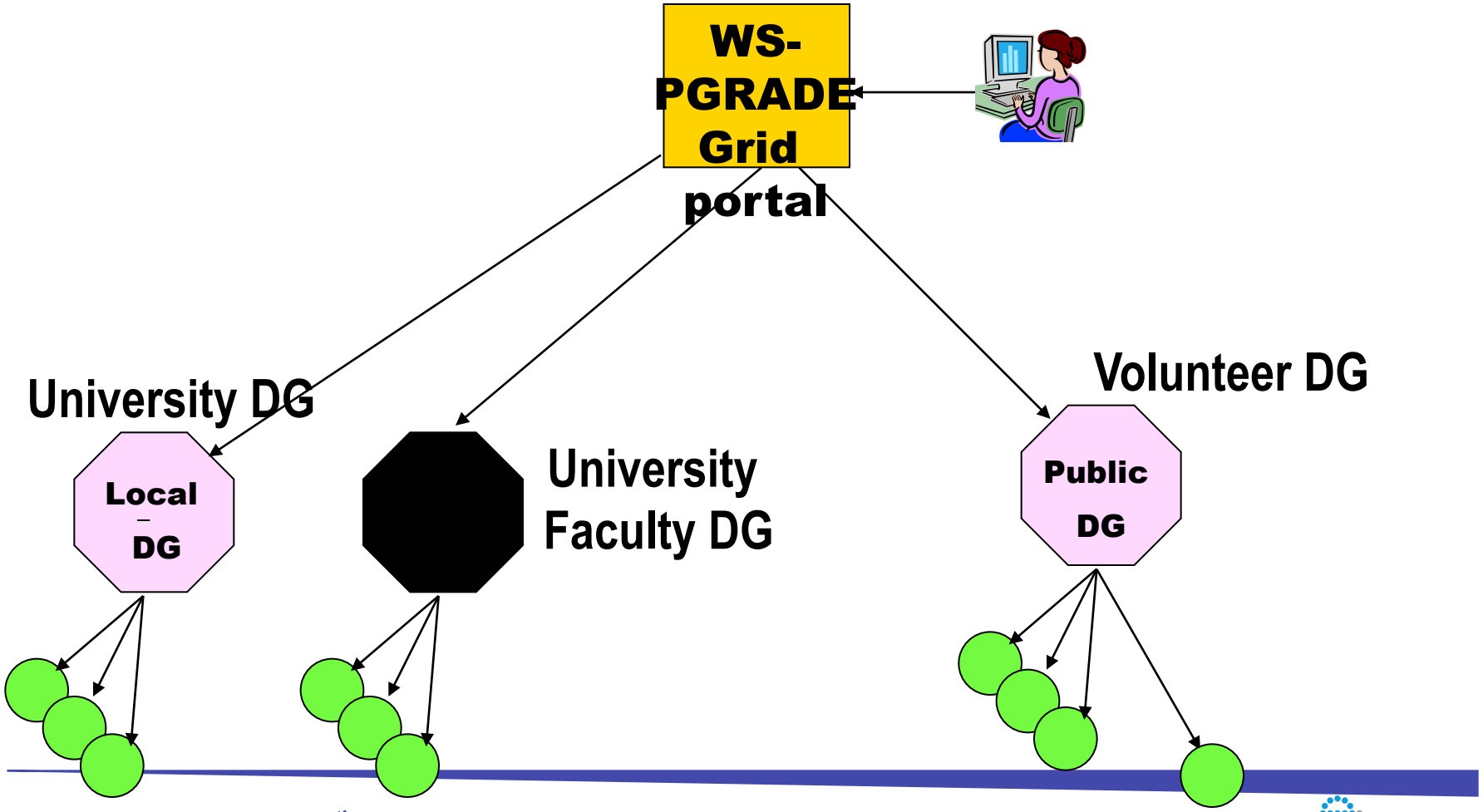
# eCom4Com

- SZTAKI developed a new technology based on BOINC and SZTAKI Desktop Grid experiences.
- The new technology is called eCom4Com (e-Computing for Communities)
- The goal of the eCom4Com technology is to quickly build and easily use BOINC-based DG systems
- Easily run PS-nodes of workflows in such DG systems.

# A real example: CancerGrid workflow



# Accessing Desktop Grids by a Grid portal



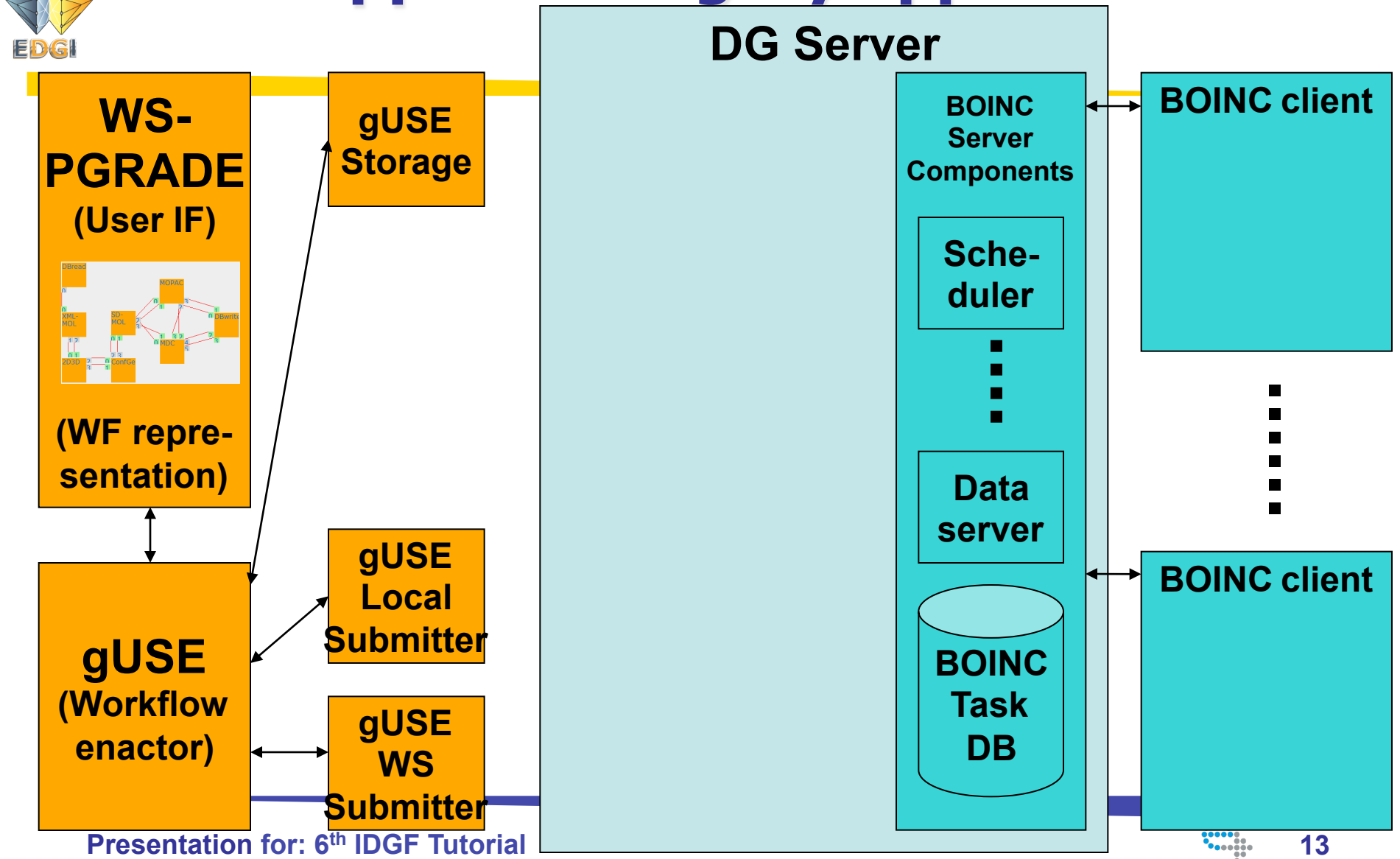
# eCom4Com software

- eCom4Com technology provides the following components on top of BOINC:
  - Easy setup: Debian packages for the server
    - For Volunteer Computing: Global
    - For Private/Local Desktop Grids: Local
  - Simplify writing applications: DC-API
  - Support porting legacy applications: GenWrapper
  - WS-PGRADE workflow portal for easily run applications
  - Allow setups peculiar to DGs:
    - clusters as resources,
    - 3G Bridge enables hierarchy, bridging to/from other systems (service grids, clouds)
  - Security enhancements: certificates, virtualisation
  - VMs to easily deploy DG systems with bridges

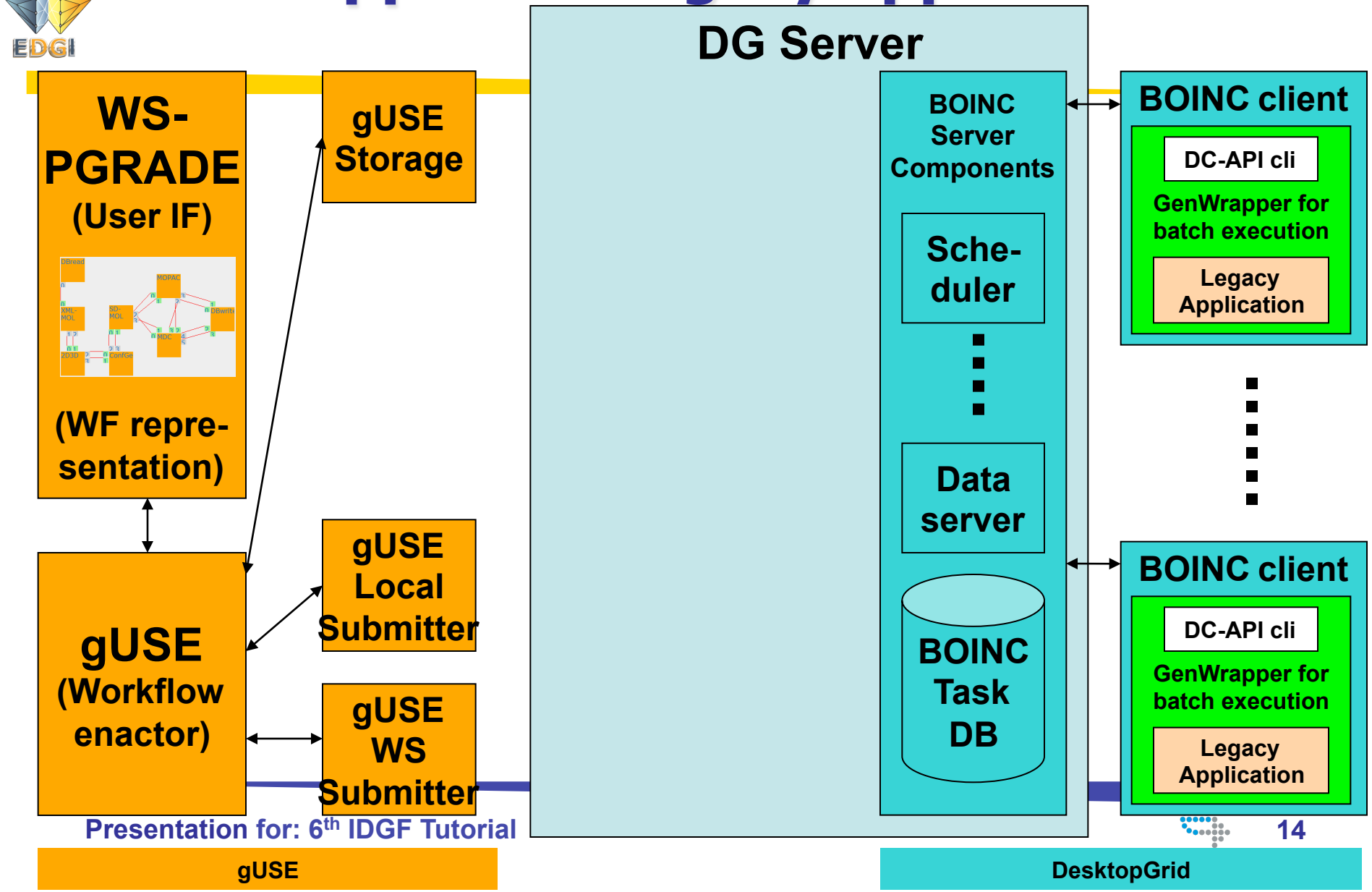
# Automatic generation of master and client code

- SZTAKI developed the DC-API (Distributed Computing API) that can
  - **automatically generate WUs** for PS jobs at the server of the DG system
- SZTAKI developed a generic wrapper that can
  - eliminate the boincification of the client code
  - **automatically generates the client code** without modification of the original code

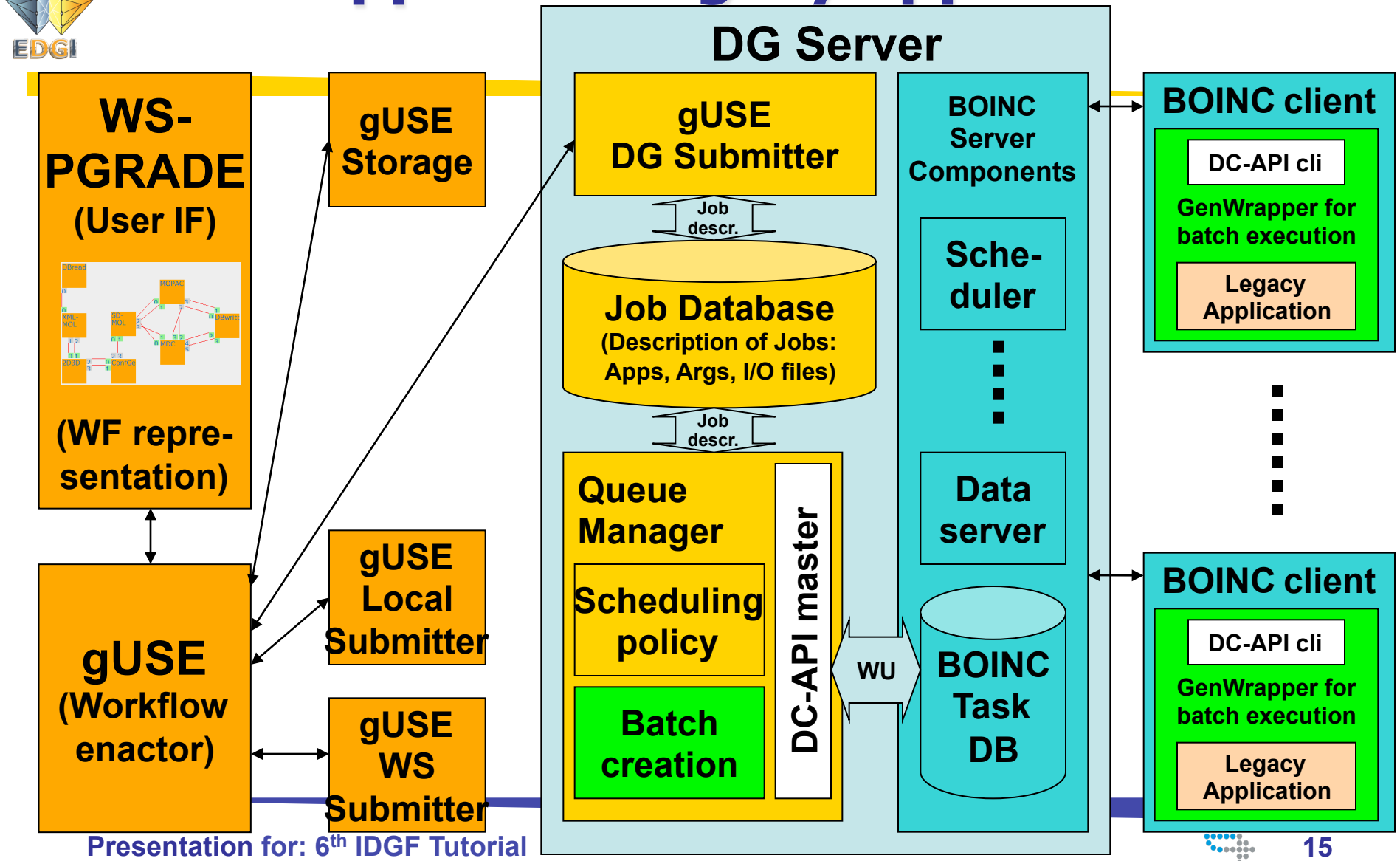
# The eCom4Com architecture to support PS legacy applications



# The eCom4Com architecture to support PS legacy applications



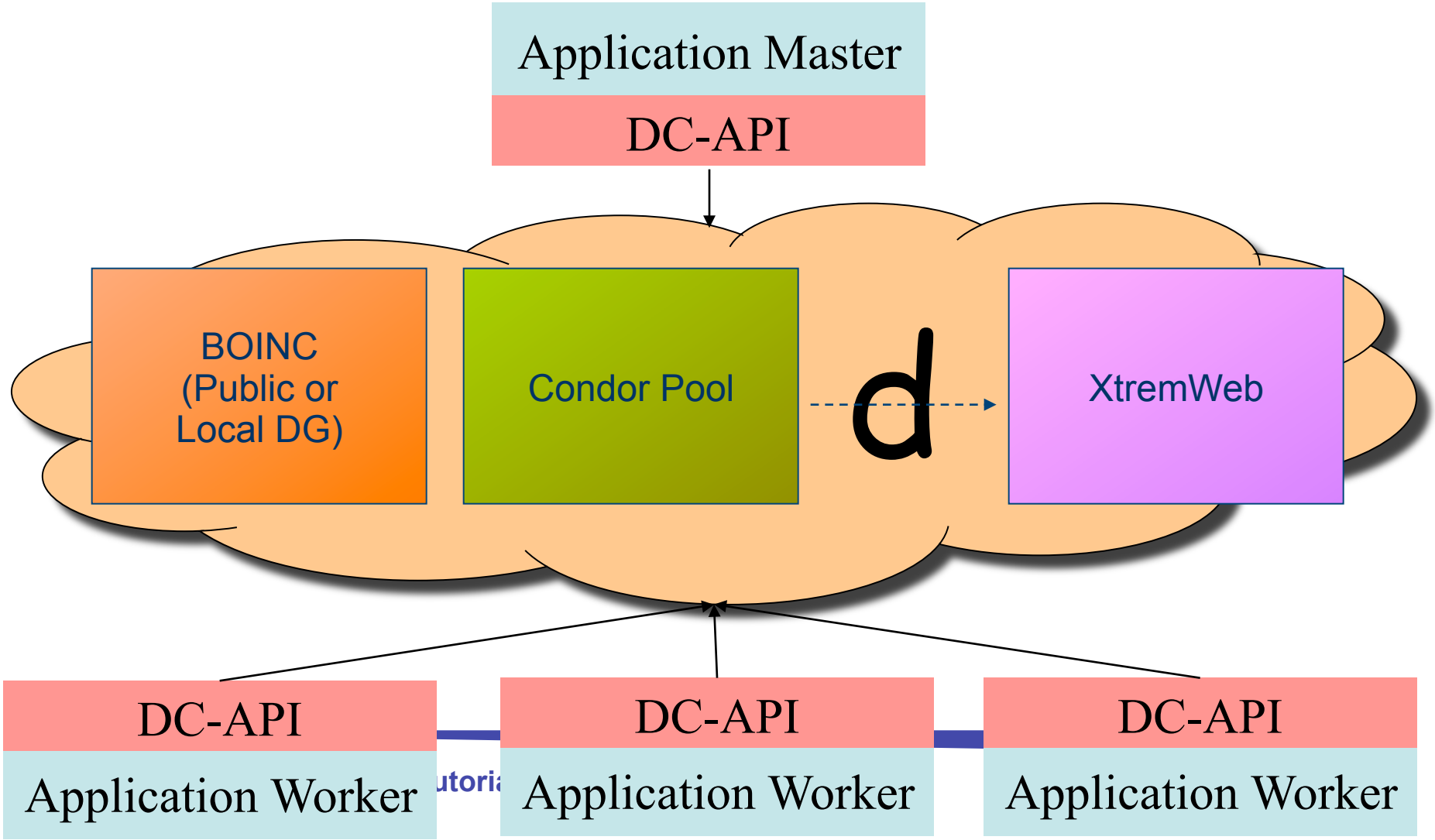
# The eCom4Com architecture to support PS legacy applications



# DC-API: Writing an application

- Simple API to hide the grid infrastructure from application developers
- Usable with minimal set of functions but has additional features that can be used when needed
- Allows application deployment on different grid infrastructures without source code modification via different backends:
  - Standalone (local) for testing
  - BOINC
  - Condor
  - XtremWeb

# DC-API application

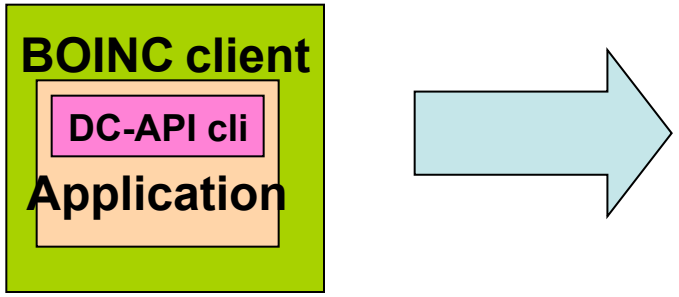


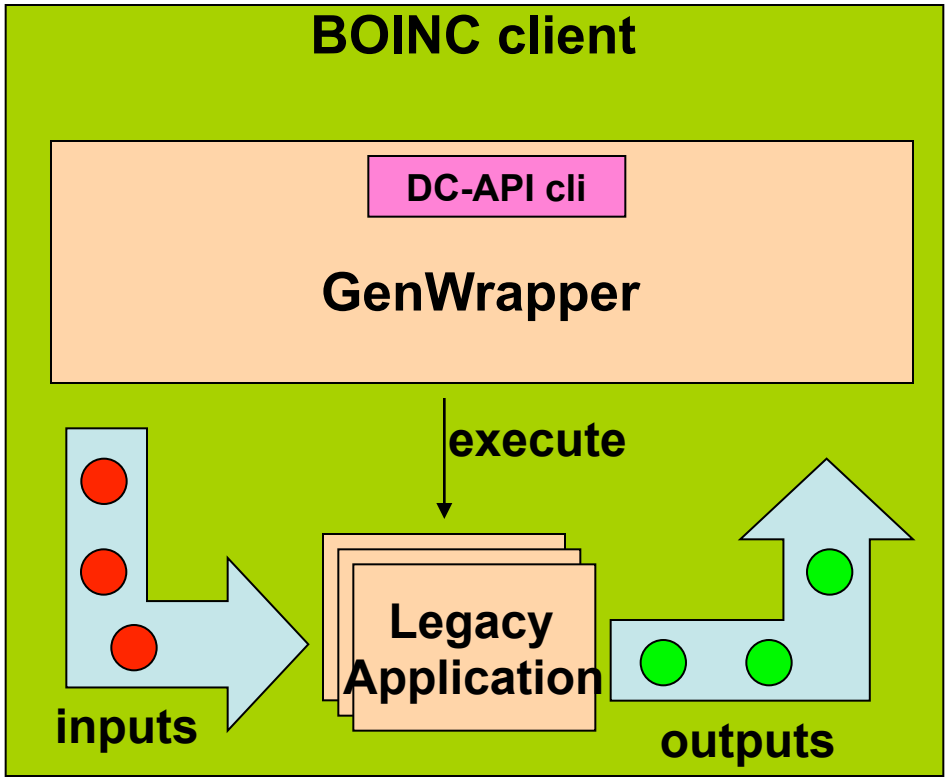
# Generic Wrapper (GenWrapper)

- **Objective of GenWrapper**
  - The features of **BOINC wrapper is not enough** (e.g. patching config files on client machines, generating extra messages, independent jobs in a WU, etc.)
  - Wanted to **be prepared for unknown requirements** might be raised by future applications (e.g. Cancergrid)
  - We did not want to extend the BOINC wrapper to make it an XML-based programming language, we choose to **BOINCify an existing language** -> Bourne shell

# Wrapping on the DG client

- GenWrapper
  - Interfaces DC-API on behalf of the application
  - prepares environment for the application
  - unpacks application binaries
  - executes application
  - handles multiple ins/outs

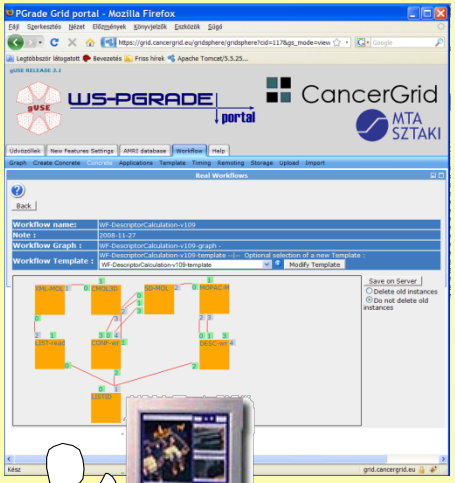


# SZTAKI Desktop Grid local version

- **Main objective:**
  - **Enable the creation of local DGs for any community**
  - **Demonstrate how to create such a system**
- Building production Grids requires huge effort and represents a privilege for those organizations where high Grid expertise is available
- **SZTAKI Desktop Grid local version is built on the eCom4Com technology**
- **Using the local SZDG package**
  - **Any organization** can build a local DG in a day with minimal cost (a strong PC is enough as a server machine)
  - The applications of the local community will be executed by the **spare PC cycles** of the local community
  - There is no limitation for the applied PCs, all the PCs of the organization can be exploited (**heterogeneous Grid**)
  - Users of the local SZDG can access the local DG system via WS-PGRADE portal

# The CancerGrid System (CSG)




executing workflows

WS-PGRADE Portal

DG jobs | Local jobs

3G Bridge

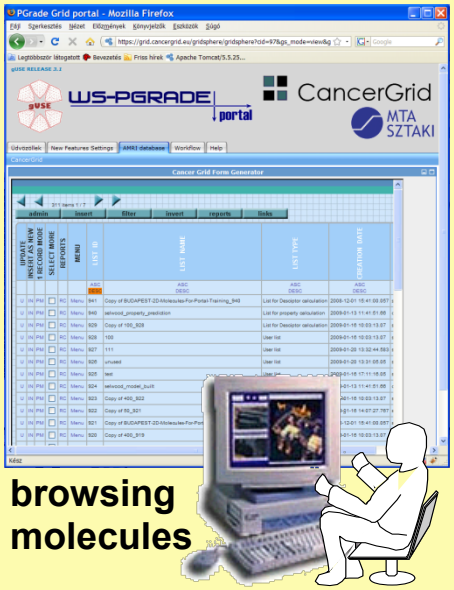
Job 1
Job 2
Job N

Local Resource

BOINC server

WU 1
WU 2
WU N

Portal Storage




browsing molecules

BOINC client

GenWrapper for batch execution

WU X
WU Y

Legacy Application

DG clients from all partners

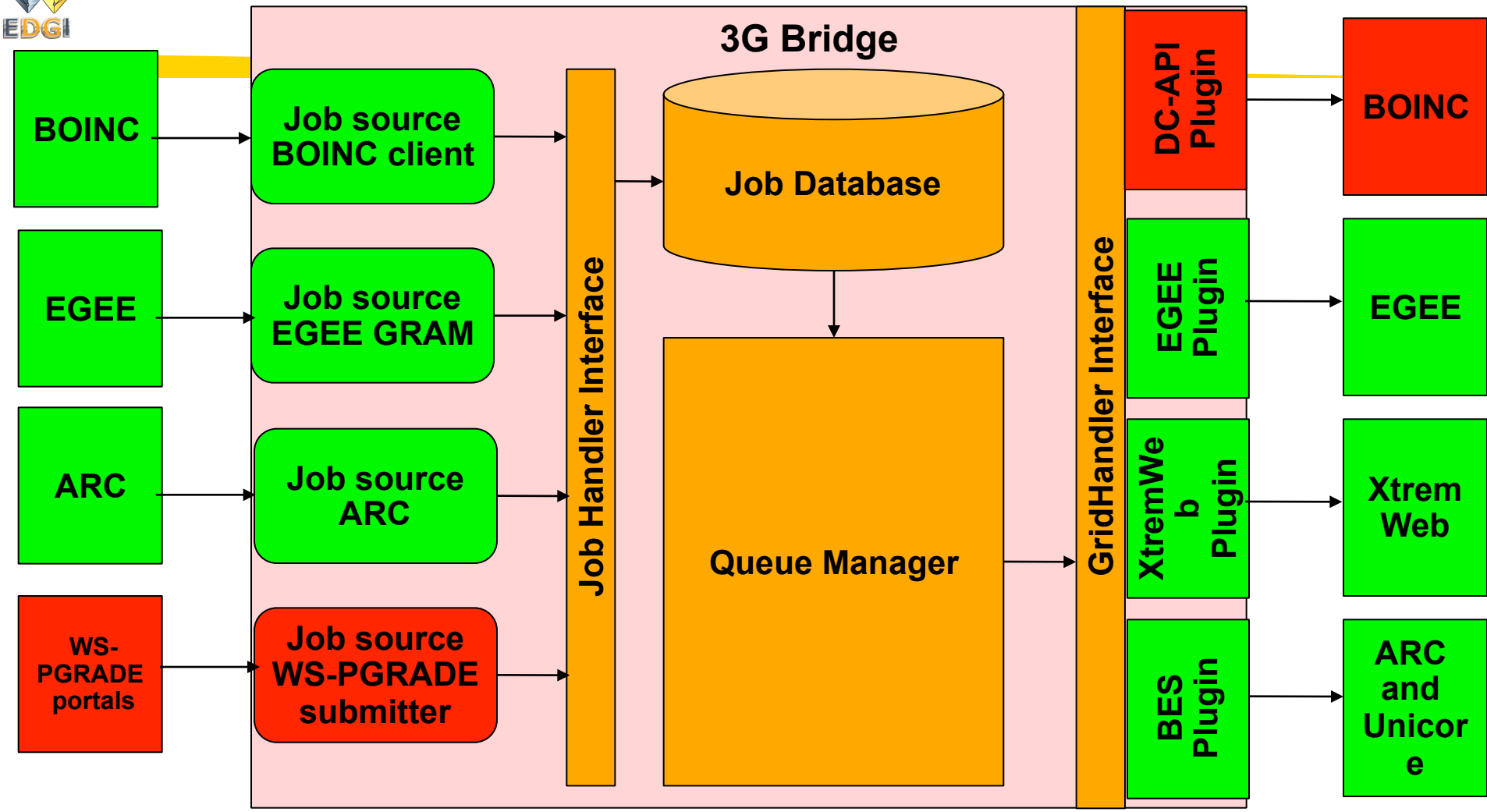
Portal and Desktop Grid server

(4 selectable workflow applications)

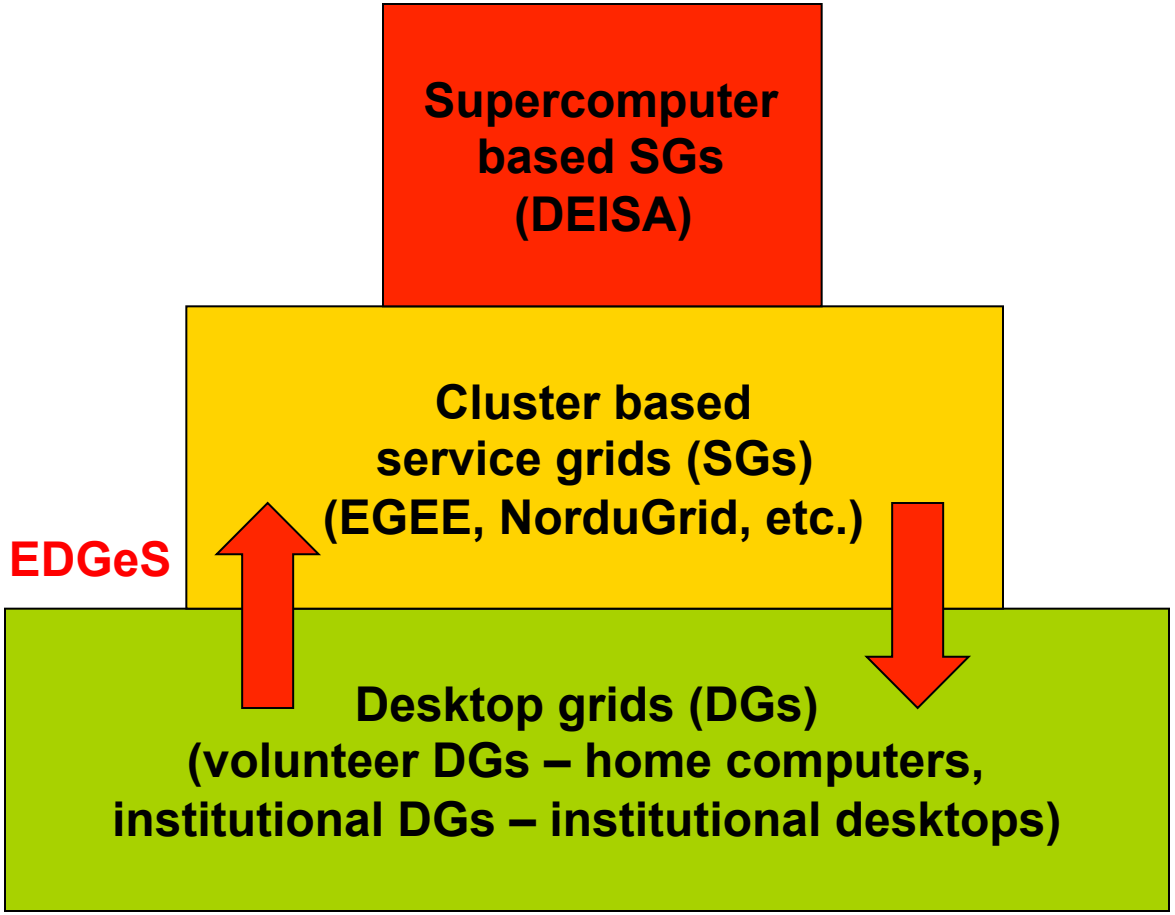
molecule database

Molecule database server

# Generic Grid-Grid (3G) Bridge

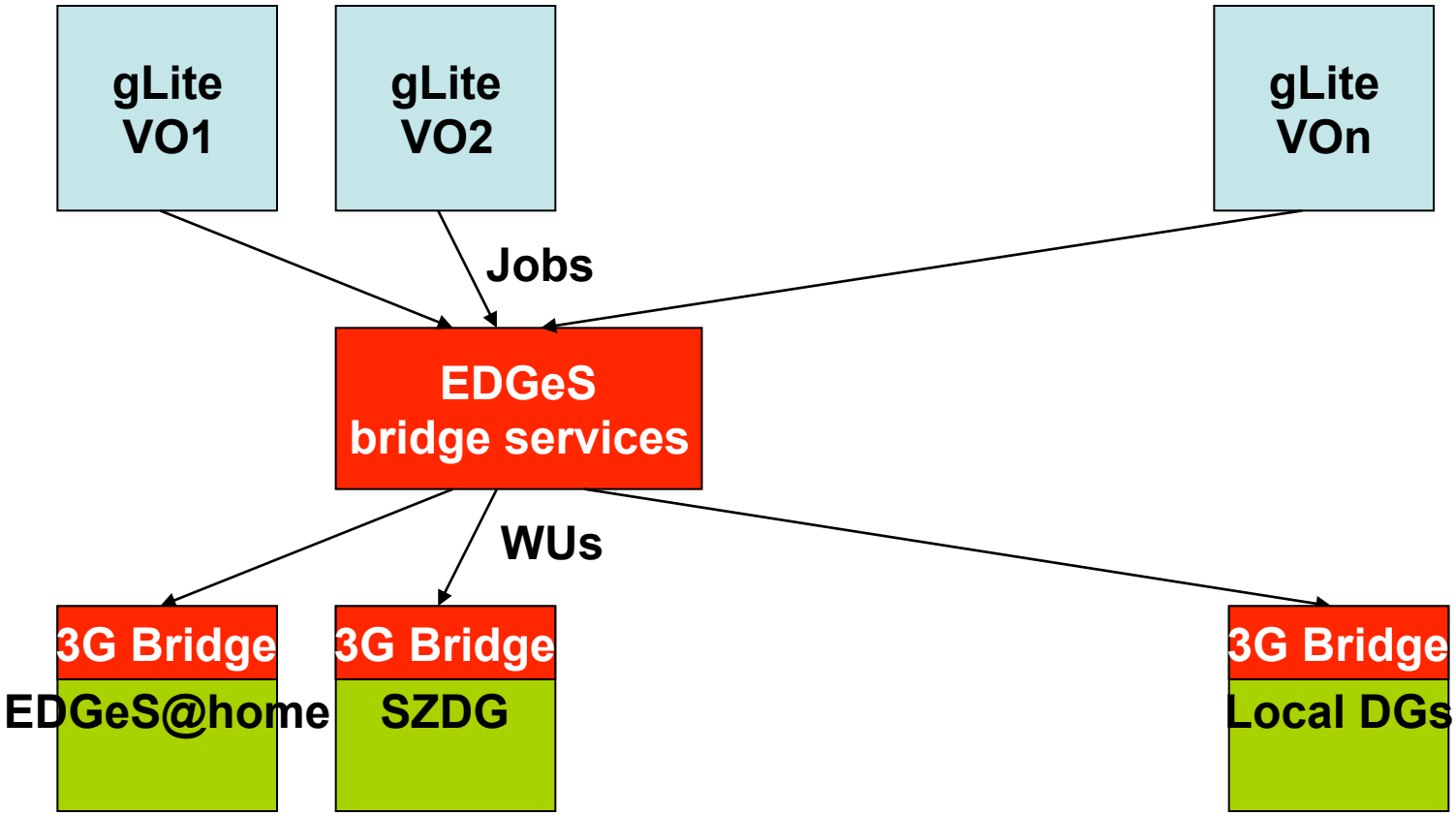


# The Grid Ecosystem

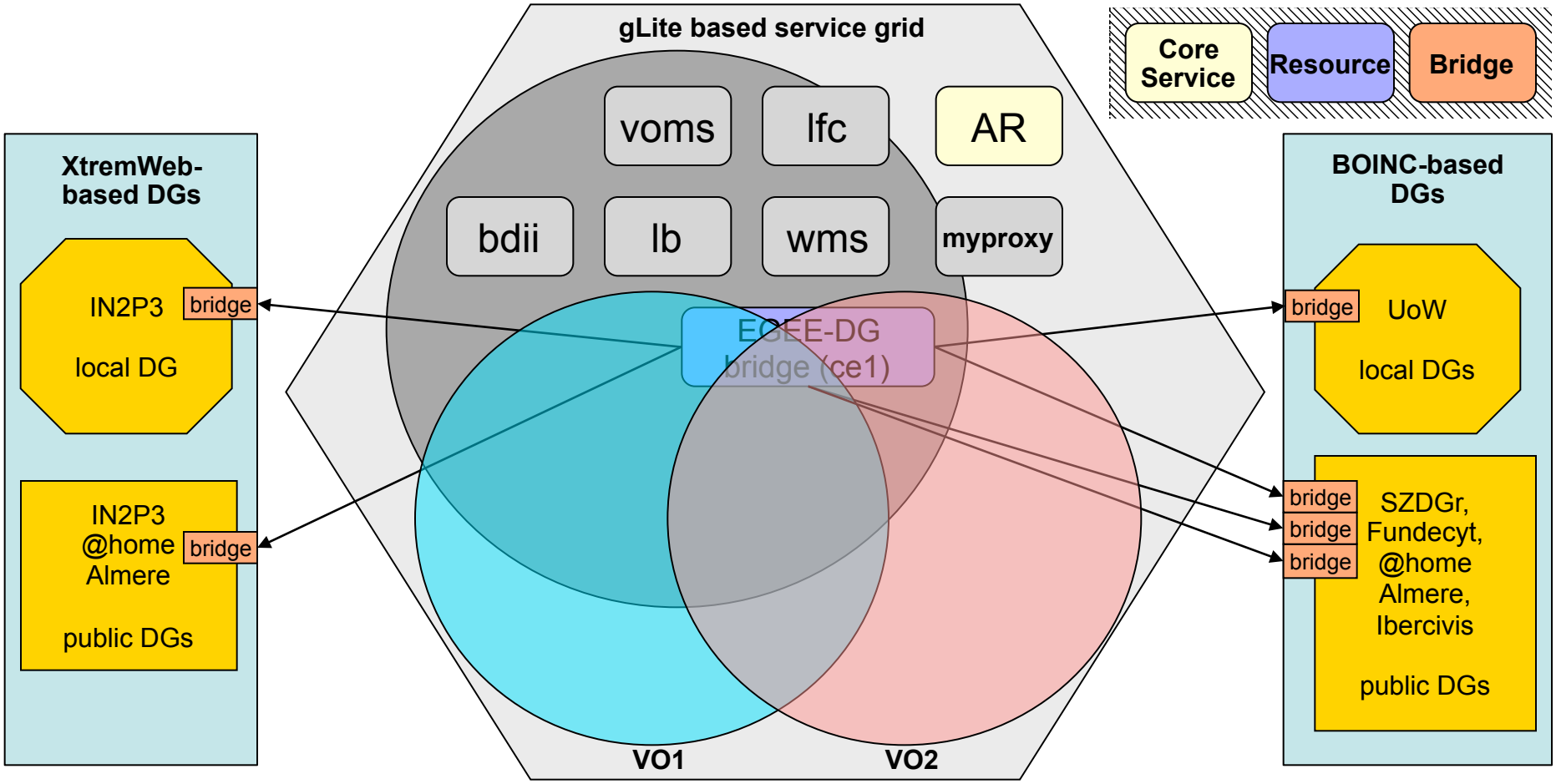



- **Very expensive,**
- small number of sites and very large number of cores
- **MPI appls**
- **Moderately expensive,**
- **moderate number of sites and CPUs**
- **any appls**
- **Inexpensive,**
- **very large number of CPUs (~10K – 1M)**
- **Bag of task appls**

# gLite->DG infrastructure



# Production gLite⇒DG Infrastructure of EDGeS



# Desktop Grid types to support SG VOs

- Volunteer desktop grids:
  - Edges@home
  - City grid: Almeregrid
- School Grid:
  - Extremadura School Grid
- Local University Grid:
  - University of Westminster


# EDGeS@Home

**Project Performance:**

**Number of users:** 8215
**Number of hosts:** 10810

**Estimated performance of last 48 hours:** 2011.187 GFlop/s

## About EDGeS@Home

The aim of the EDGeS@Home project is to support the execution of selected and validated scientific applications developed by the EGEE and EDGeS community.

The project currently hosts the ISDEP - Integrator of Stochastic Differential Equations in Plasmas - application at production level.

It also hosts several other applications at beta (experimental) level.

More information about the EDGeS project can be found here.

## Join EDGeS@Home

- Read our rules and policies
- This project uses BOINC. If you're already running BOINC, select Attach to Project. If not, download BOINC.
- When prompted, enter **http://home.edges-grid.eu/home/**
- If you're running a command-line or pre-5.0 version of BOINC, create an account first.
- If you have any problems, get help here.

## Returning participants

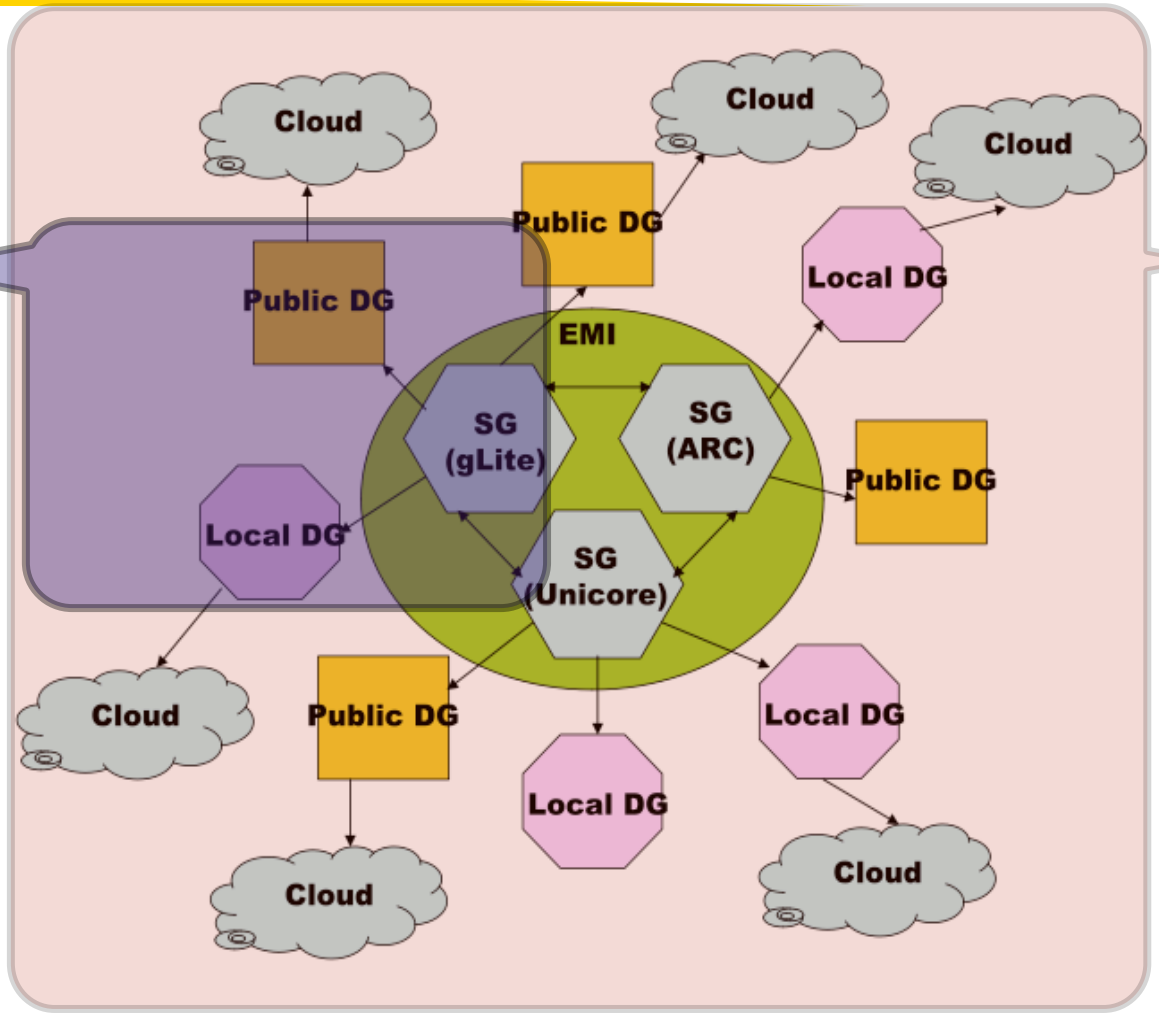
- Your account - view stats, modify preferences

## Community

- Profiles

# Plans for EDGI

EDGeS scope only for compute intensive applications for EGEE (gLite)



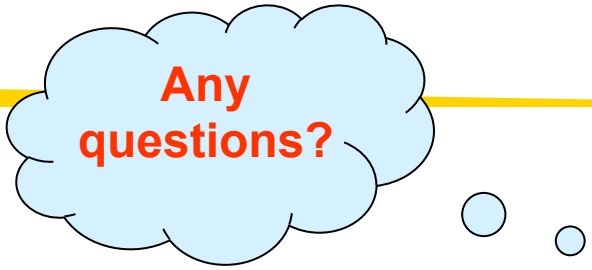

EDGI scope for both compute and data intensive applications for EMI/EGI (gLite, ARC, Unicore)

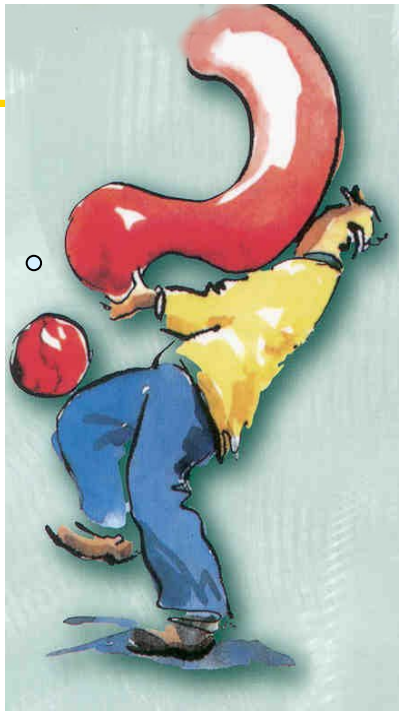
Extend Desktop Grids with Clouds for QoS

# Summary

- eCom4Com technology enables to
  - easily build and program local DG systems
  - Easily extend them with other DGs, grids and clouds
- EDGeS created production infrastructure to extend
  - DG systems with grids
  - Grids with DGs
- EDGeS creates production infrastructure to extend
  - Grids with DGs
  - DG systems with clouds
- Based on these new scenarios the DG concept will be much more popular than before (see the case in Europe)
- We invite you to join us in IDGF and let's collaborate

# Thank you for your attention …



**For more information please visit the EDGeS and EDGI Websites:**

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

http://edgi-project.eu

**and/or send e-mail to me:**

**kacsuk@sztaki.hu**