int.eu.grid

http://www.interactive-grid.eu

Interactive grid-access for USCT

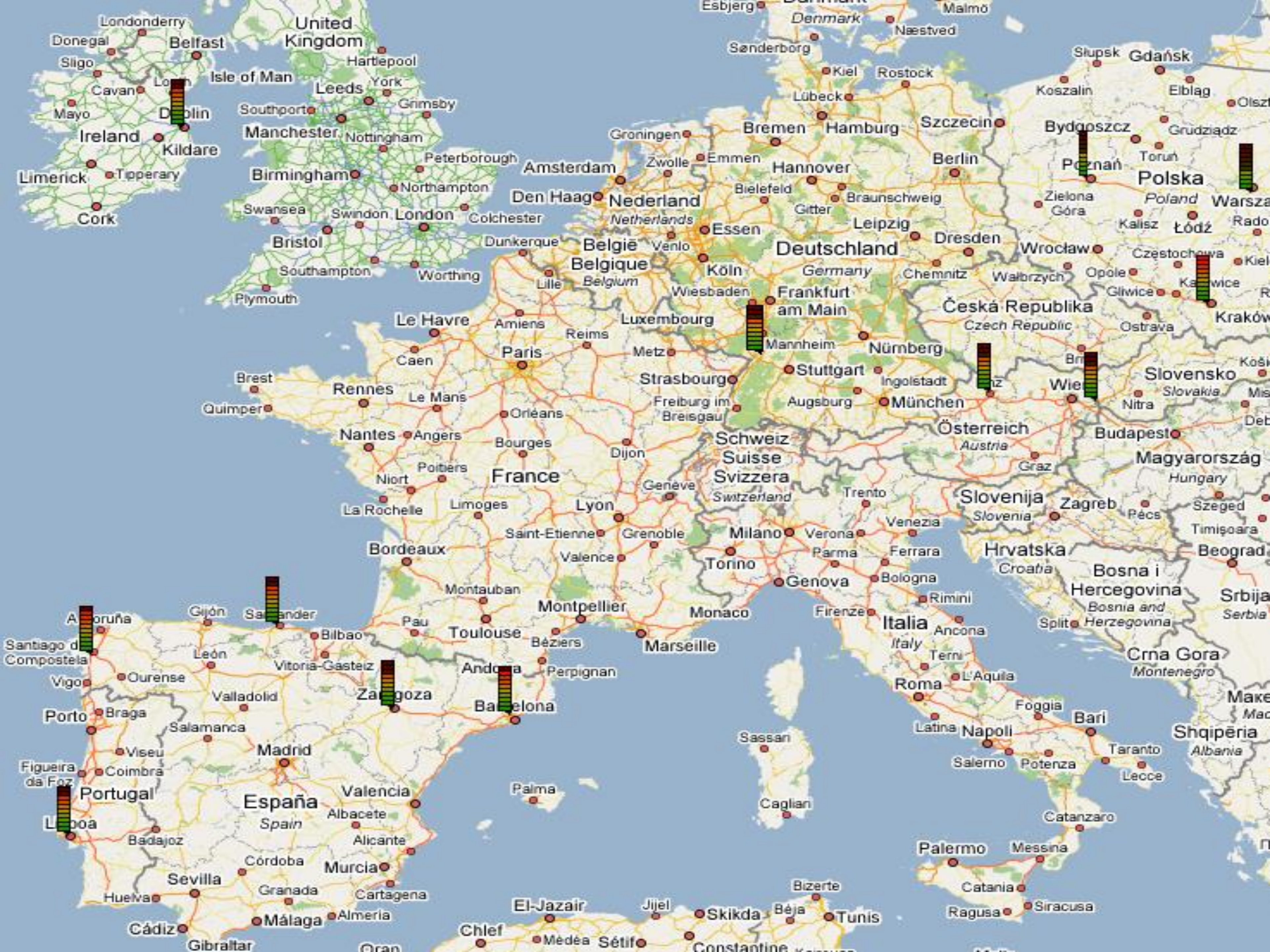
Marcus Hardt
Forschungszentrum Karlsruhe

Background

- The grid
- The application
- Interactive access

The grid

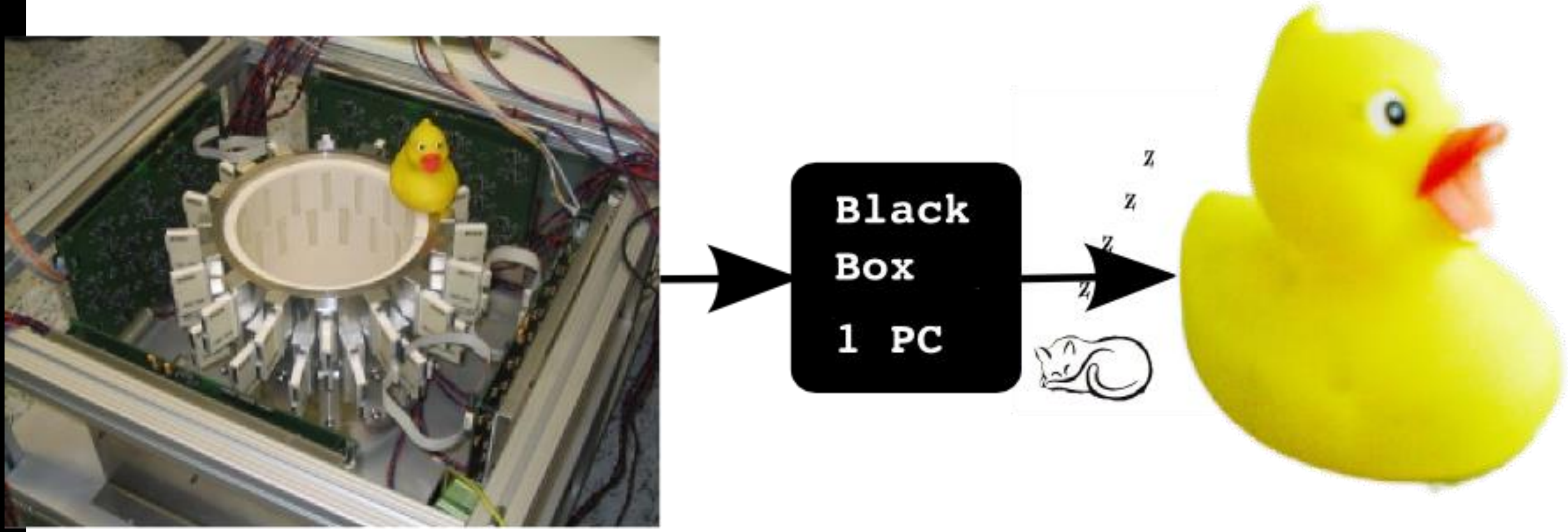
- The grid
  - **Interactive European Grid (int.eu.grid)**
  - 2 Year I3 Project (May'06 - April'08)
  - 2.5 M€ EU-funding
  - Mission
    - 100% gLite compatible
    - MPI for the grid (int**ra** and int**er** site MPI)
    - Bring grid to new user communities
    - Improved useability
  - 12 clusters in 7 countries
    ~ 400 CPUs & 25TB disk


Esbjerg
Denmark
Malmö
Næstved
Sønderborg
Kiel
Rostock
Słupsk
Gdańsk
Londonderry
Donegal
United Kingdom
Belfast
Hartlepool
Sligo
Isle of Man
York
Leeds
Lübeck
Koszalin
Elbląg
Cavan
Grimsby
Mayo
Southport
Dublin
Bremen
Hamburg
Szczecin
Bydgoszcz
Grudziądz
Ireland
Manchester
Nottingham
Groningen
Kildare
Emmen
Toruń
Zwolle
Peterborough
Hannover
Berlin
Limerick
Tipperary
Birmingham
Amsterdam
Poznań
Polska
Northampton
Bielefeld
Zielona Góra
Poland
Warszawa
Den Haag
Nederland
Braunschweig
Swansea
Cork
Swindon
London
Gitter
Netherlands
Colchester
Kalisz
Łódź
Leipzig
Essen
Bristol
Dunkerque
België
Venlo
Dresden
Wrocław
Deutschland
Częstochowa
Belgique
Southampton
Worthing
Köln
Germany
Chemnitz
Opole
Wałbrzych
Lille
Belgium
Katowice
Gliwice
Plymouth
Wiesbaden
Frankfurt am Main
Česká Republika
Kraków
Luxembourg
Czech Republic
Le Havre
Amiens
Reims
Ostrava
Mannheim
Nürnberg
Caen
Paris
Metz
Brno
Brest
Stuttgart
Slovensko
Strasbourg
Ingolstadt
Linz
Rennes
Wien
Le Mans
Slovakia
Freiburg im Breisgau
Augsburg
München
Nitra
Quimper
Orléans
Nantes
Angers
Österreich
Budapest
Schweiz
Bourges
Austria
Dijon
Suisse
Poitiers
Magyarország
Niort
Graz
France
Svizzera
Genève
Hungary
Switzerland
Trento
Slovenija
Zagreb
La Rochelle
Limoges
Lyon
Slovenia
Szeged
Pécs
Venezia
Saint-Etienne
Grenoble
Milano
Verona
Timișoara
Bordeaux
Valence
Torino
Parma
Ferrara
Hrvatska
Beograd
Croatia
Bosna i Hercegovina
Montauban
Genova
Bologna
Rimini
Srbija
Montpellier
Monaco
Bosnia and Herzegovina
Serbia
Gijón
Santander
Firenze
Italia
A Coruña
Split
Pau
Toulouse
Ancona
Bilbao
Marseille
Santiago de Compostela
Béziers
Italy
León
Crna Gora
Terni
Montenegro
Vitoria-Gasteiz
Andorra
Perpignan
L'Aquila
Ourense
Vigo
Roma
Valladolid
Zaragoza
Barcelona
Foggia
Bari
Porto
Braga
Salamanca
Sassari
Latina
Napoli
Shqipëria
Madrid
Viseu
Taranto
Albania
Figueira da Foz
Coimbra
Salerno
Potenza
Lecce
Portugal
España
Valencia
Palma
Cagliari
Lisboa
Albacete
Spain
Catanzaro
Badajoz
Alicante
Palermo
Messina
Córdoba
Murcia
Sevilla
Bizerte
Granada
Cartagena
Catania
Huelva
El-Jazair
Jijel
Skikda
Béja
Tunis
Almeria
Ragusa
Siracusa
Cádiz
Málaga
Chlef
Médéa
Sétif
Gibraltar
Oran
Constantine

USCT Reconstruction

- Current focus: development of
  - Reconstruction Algorithms
  - Improved hardware
- Matlab
  - Problem solving environment
    Maple, Mathematica, ...
  - Strategic development platform
    => 95% code in Matlab
- Data:
  - 20GB input
  - Output: 3D volume graphics (est. 8GB)

USCT Reconstruction



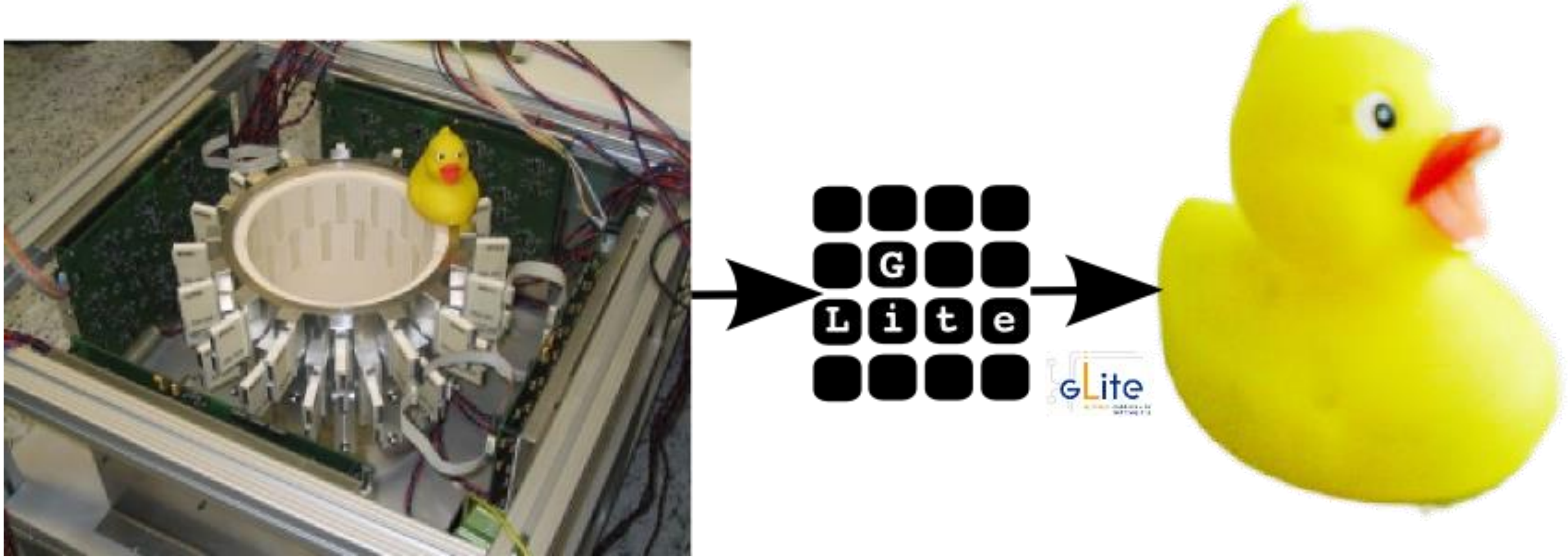
Black Box

1 PC

- Computation takes long (days, weeks, years)
- **Goal: Seemless + interactive grid access from Matlab**

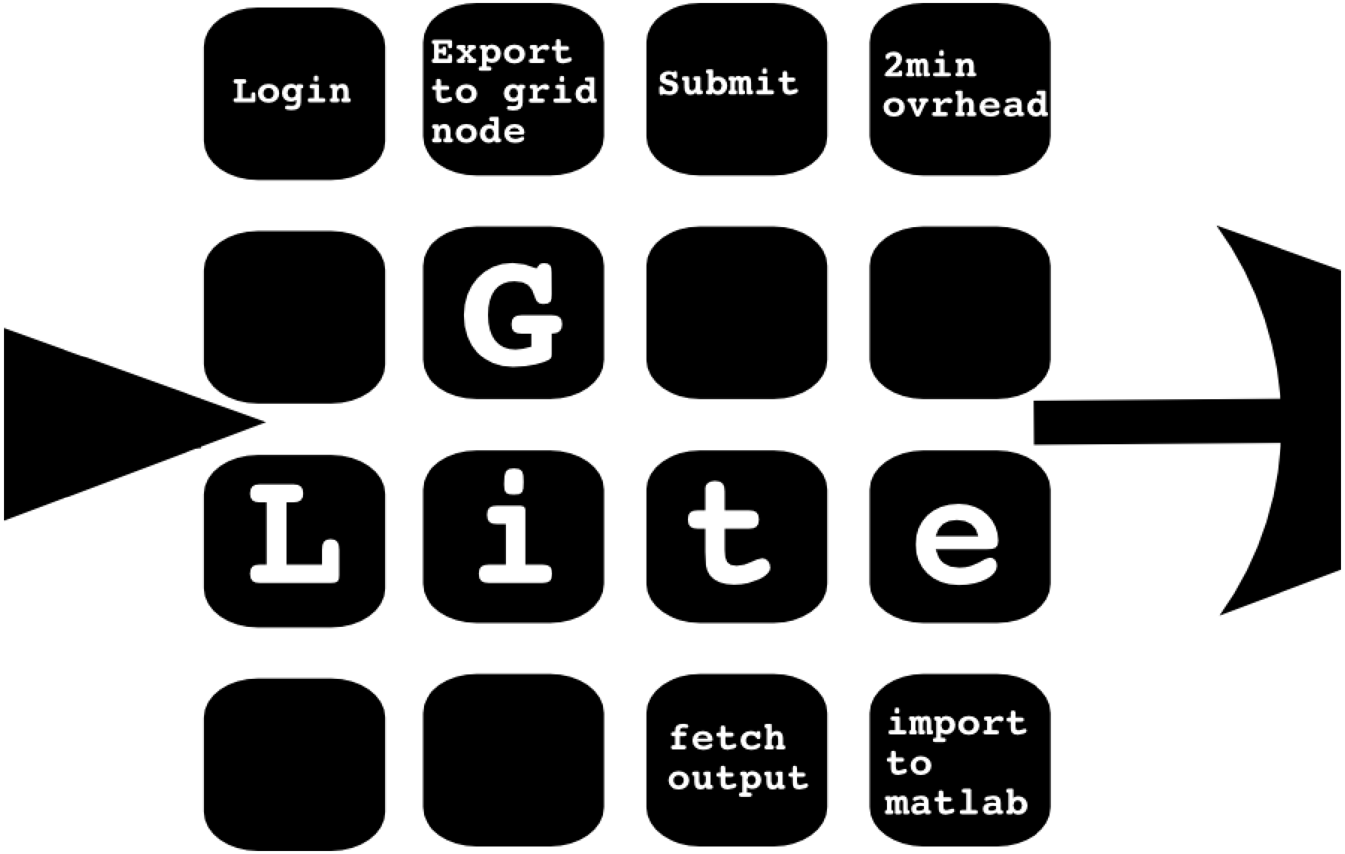
Using the grid

- 1st Approach to parallel execution:
  - Partitioning of data
  - Many parallel jobs



Using the grid

- Lets take a close look



Using the grid

- gLite
  - Good resource allocation system
- Users will run away

Improve grid access!!

- 2nd Approach
  - Grid access -- directly from Matlab
- RPC tool: **GridSolve**
  - **Developed at ICL, University Tennessee, Knoxville**
  - **Hides complexity of grid in <u>one simple call</u>**
    - **Client** interface for Java, C, Fortran, Matlab, Octave

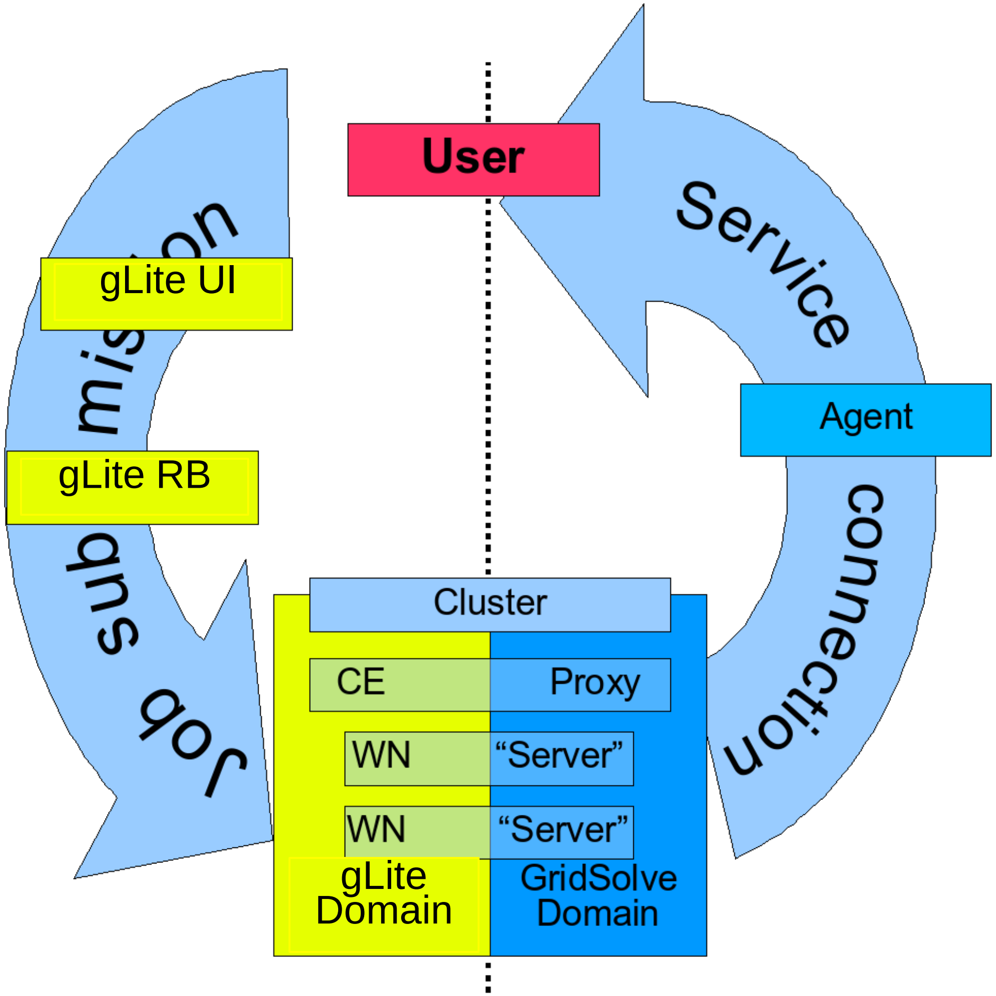
```
y=problem(x) <=> y=gs_call('problem', x)
```

    - Transport input parameters to remote side
    - Execute "problem"
    - Transport result back
  - Major side benefit
    - Parallelisation of Matlab code
      (Matlab is only single threaded)

Integrate...

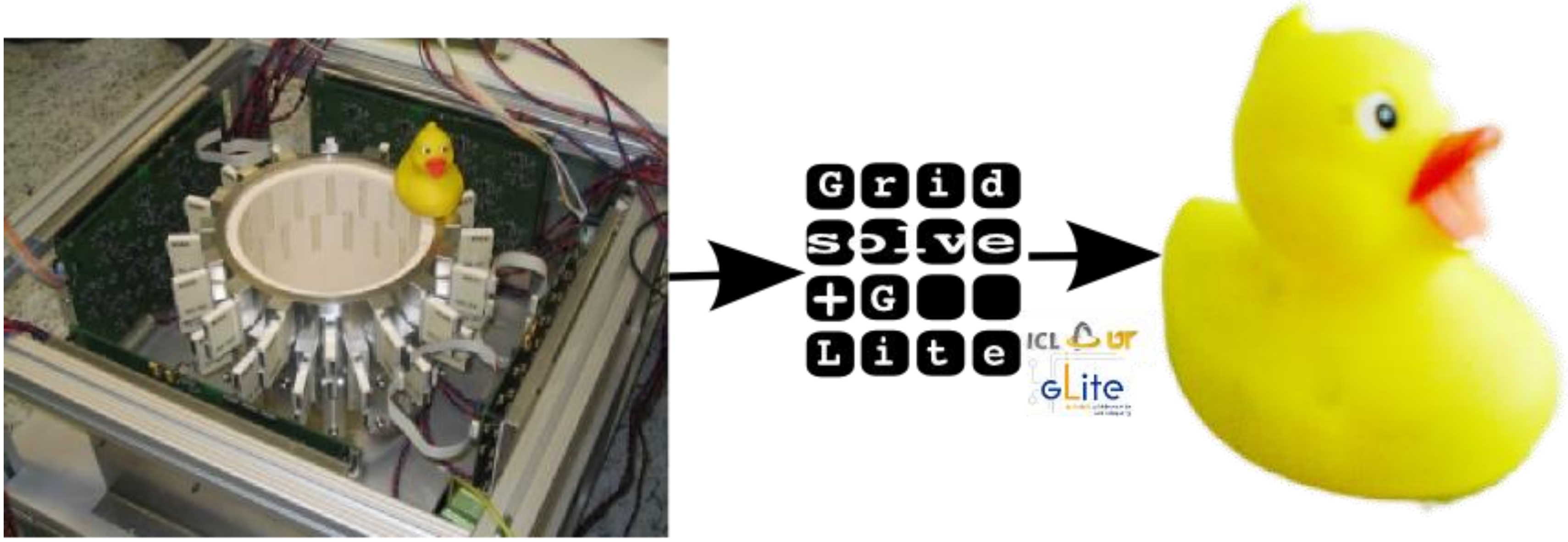
- ... GridSolve + gLite
  - Create Service hosts (GS-agent + GS-proxy)
    - Public IP address
  - Encapsulate GS-server into gLite job
    - Install GS-server on the fly
    - Deployment of problems
- ... GridSolve + Matlab
  - Point ML to the right service host
  - Support for service creation
  - Deployment of services (on the fly)
- ... Matlab + gLite
  - Matlab Compiler Runtime (MCR)
    - Install on the fly
    - Linux version incompatibility
      - Install new glibc on the fly

GridSolve on top of int.eu.grid/gLite



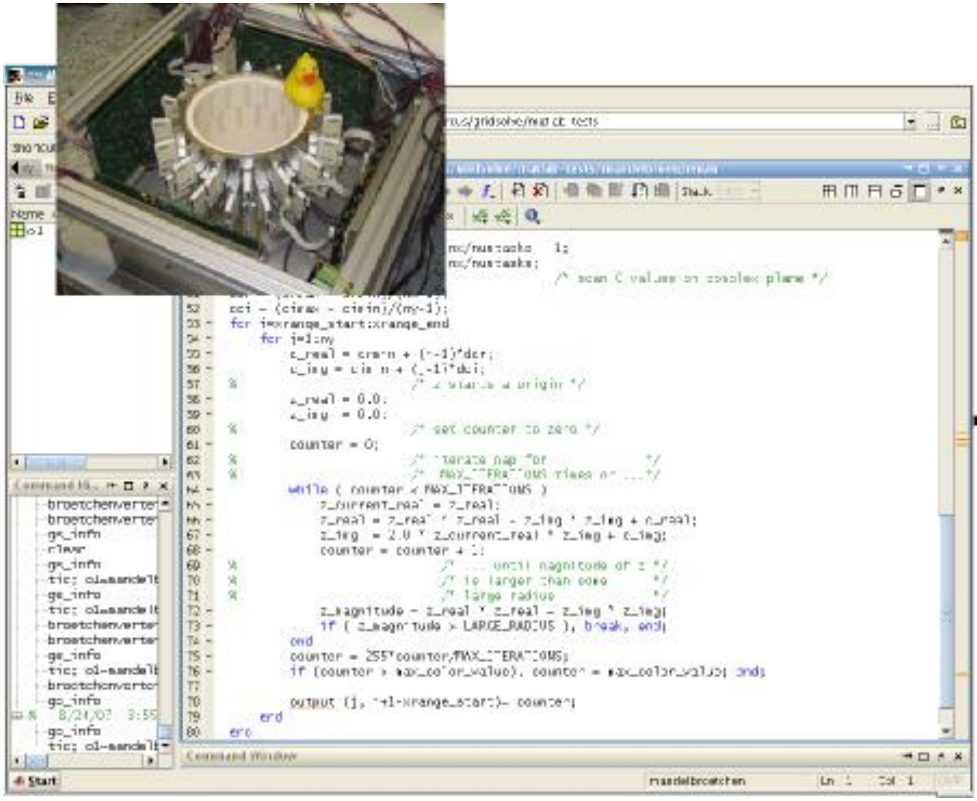
Solution

- GRPC with GridSolve on top of gLite

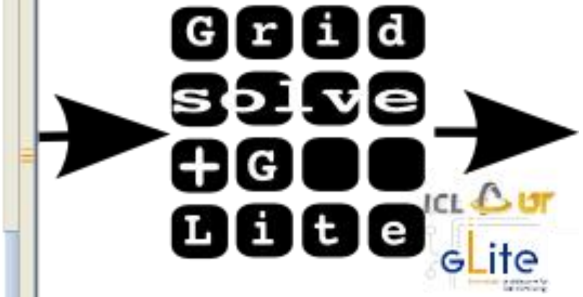


Demonstration

- USCT is a complex Application
  => Simulation for proof of concept

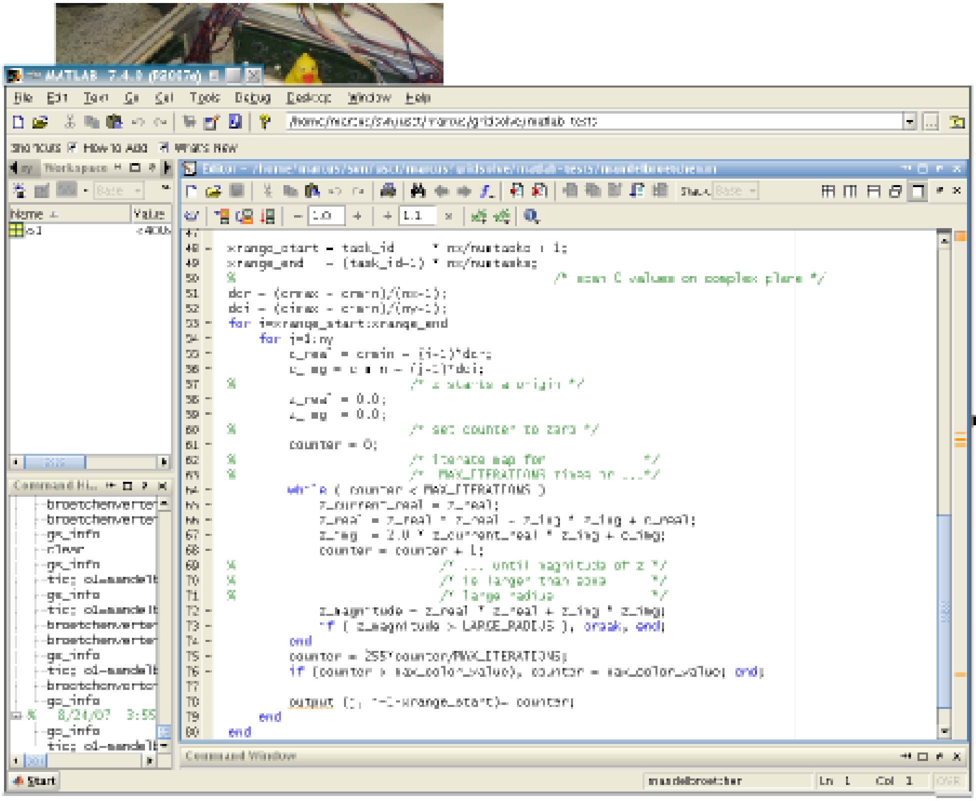


```
mx/numtasks   1;
mx/numtasks;
                        /* scan C values on complex plane */
cci = (cimax - cimin)/(my-1);
for i=xrange_start:xrange_end
    for j=1:my
        c_real = crmin + (i-1)*dcr;
        c_img = cimin + (j-1)*dci;
%                       /* z starts a origin */
        z_real = 0.0;
        z_img  = 0.0;
%                       /* set counter to zero */
        counter = 0;
%                       /* iterate map for              */
%                       /*  MAX_ITERATIONS times or ...*/
        while ( counter < MAX_ITERATIONS )
            z_current_real = z_real;
            z_real = z_real * z_real - z_img * z_img + c_real;
            z_img  = 2.0 * z_current_real * z_img + c_img;
            counter = counter + 1;
%                           /* ... until magnitude of z */
%                           /* is larger than some      */
%                           /* large radius             */
            z_magnitude = z_real * z_real - z_img * z_img;
            if ( z_magnitude > LARGE_RADIUS ), break, end;
        end
        counter = 255*counter/MAX_ITERATIONS;
        if (counter > max_color_value), counter = max_color_value; end;

        output (j, i+1-xrange_start)= counter;
    end
end
```

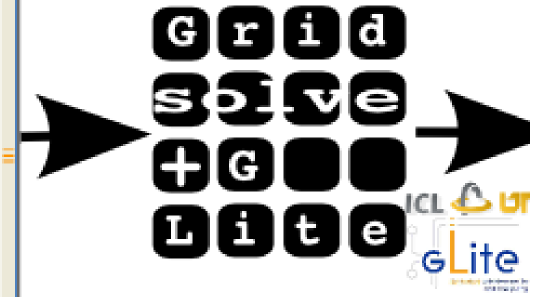


Demonstration

- Simulation: Mandelbrot fractal
  - Using the same infrastructure


MATLAB 7.4.0 (R2007a)
File Edit Text Go Cell Tools Debug Desktop Window Help
Shortcuts How to Add What's New
Workspace
Name Value
Command History
Command Window
Start
mandelbrotscher Ln 1 Col 1 OVR

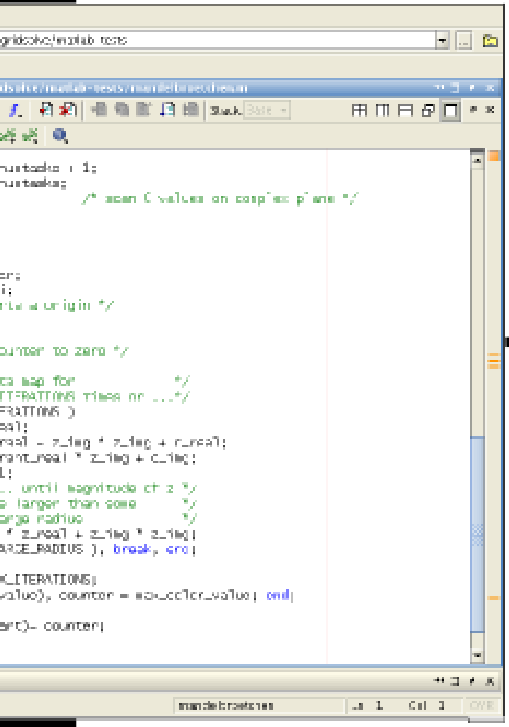
```
xrange_start = task_id     * nx/numtasks + 1;
xrange_end   = (task_id+1) * nx/numtasks;
%                                          /* scan C values on complex plane */
dcr = (crmax - crmin)/(nx-1);
dci = (cimax - cimin)/(ny-1);
for i=xrange_start:xrange_end
    for j=1:ny
        c_real = crmin + (i-1)*dcr;
        c_img = cimin + (j-1)*dci;
%                       /* z starts a origin */
        z_real = 0.0;
        z_img  = 0.0;
%                       /* set counter to zero */
        counter = 0;
%                       /* iterate map for              */
%                       /*  MAX_ITERATIONS times or ...*/
        while ( counter < MAX_ITERATIONS )
            z_current_real = z_real;
            z_real = z_real * z_real - z_img * z_img + c_real;
            z_img  = 2.0 * z_current_real * z_img + c_img;
            counter = counter + 1;
%                           /* ... until magnitude of z */
%                           /* is larger than some      */
%                           /* large radius             */
            z_magnitude = z_real * z_real + z_img * z_img;
            if ( z_magnitude > LARGE_RADIUS ), break, end;
        end
        counter = 255*counter/MAX_ITERATIONS;
        if (counter > max_color_value), counter = max_color_value; end;

        output (:, i+1-xrange_start)= counter;
    end
end
```



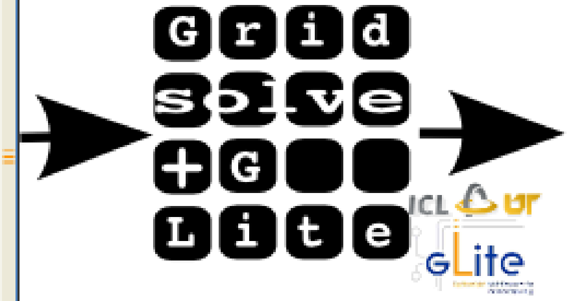
Demonstration

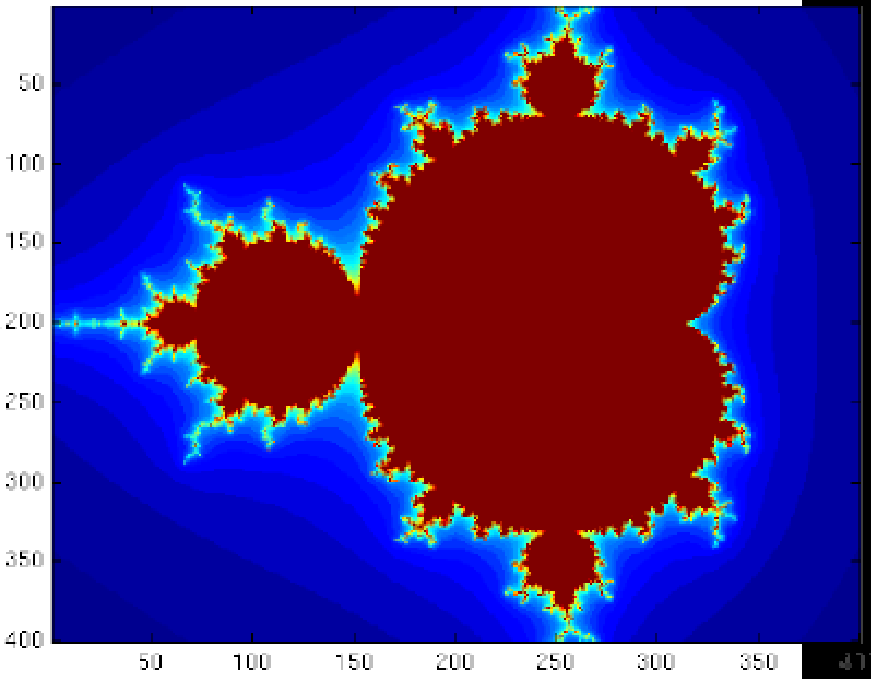
Demonstration

- Simulation works
  - Reasonable speedup



```
numtasks : 1;
numtasks;
            /* scan C values on complex plane */




er;
i;
rie a origin */


ounter to zero */

ca map for              */
ITERATIONS times or ...*/
RATIONS )
eal;
real - z_img * z_img + c_real;
rant_real * z_img + c_img;
l;
.. until magnitude of z */
s larger than some      */
arge radius             */
* z_real + z_img * z_img;
ARGE_RADIUS ), break, end;

X_ITERATIONS;
value), counter = max_color_value; end;

art)- counter;
```





Current work

- Use real code
  - Cope with the data (20 GB in, 8 GB out)
    - MPI abilities might be beneficial (depending on algorithm)
  - Identify Bottlenecks
  - Improve useability
- Data Handling (Future)
  - Distribute Input on the grid
  - Collect output efficiently
  - GFAL + gLite-DICOM

Questions?

Contact:

interactive grid