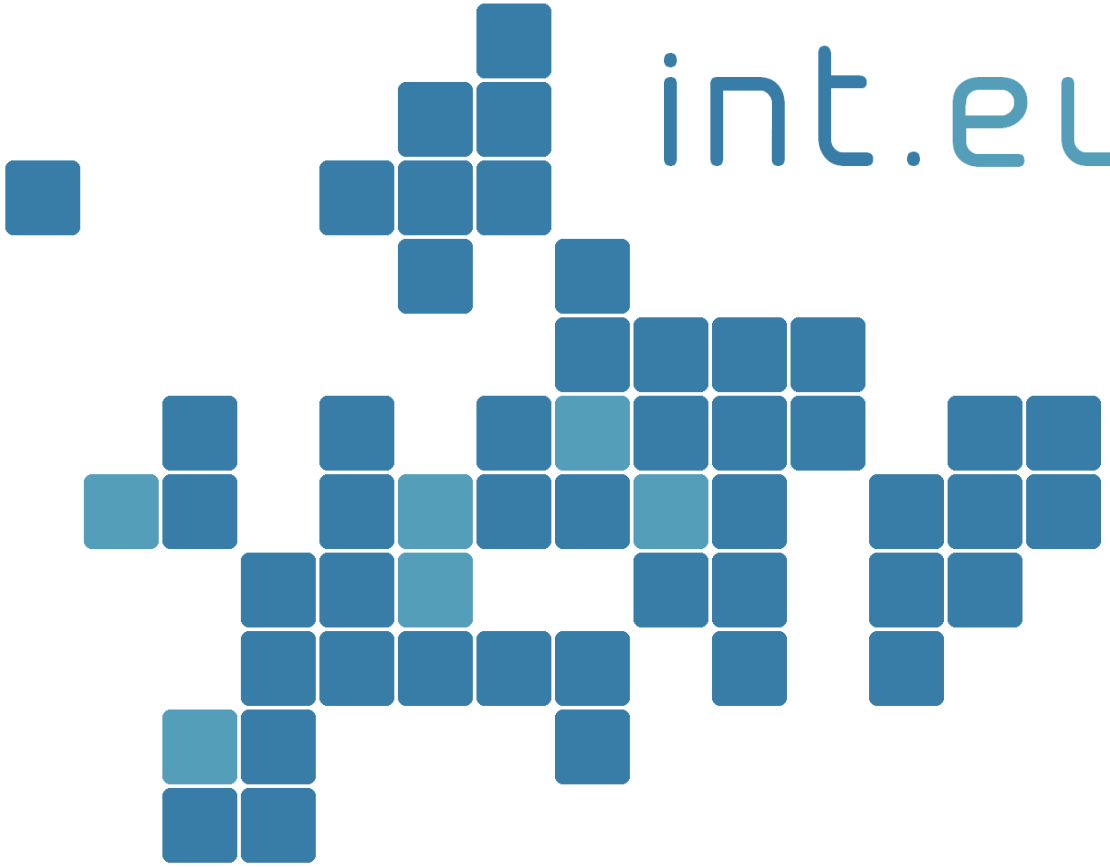


int.eu.grid

http://www.interactive-grid.eu

Where am I from?

- Karlsruhe Institute of Technology (KIT)
  - Merger between University Karlsruhe (UKA)
  - Forschungszentrum Karlsruhe (FZK)
  - 7800 Employees
  - 50 Institutes
- Steinbuch Centre for Computing (SCC)
  - 200 Employees
  - Grid
  - Cloud
  - HPC-Applications
  - EU-projects

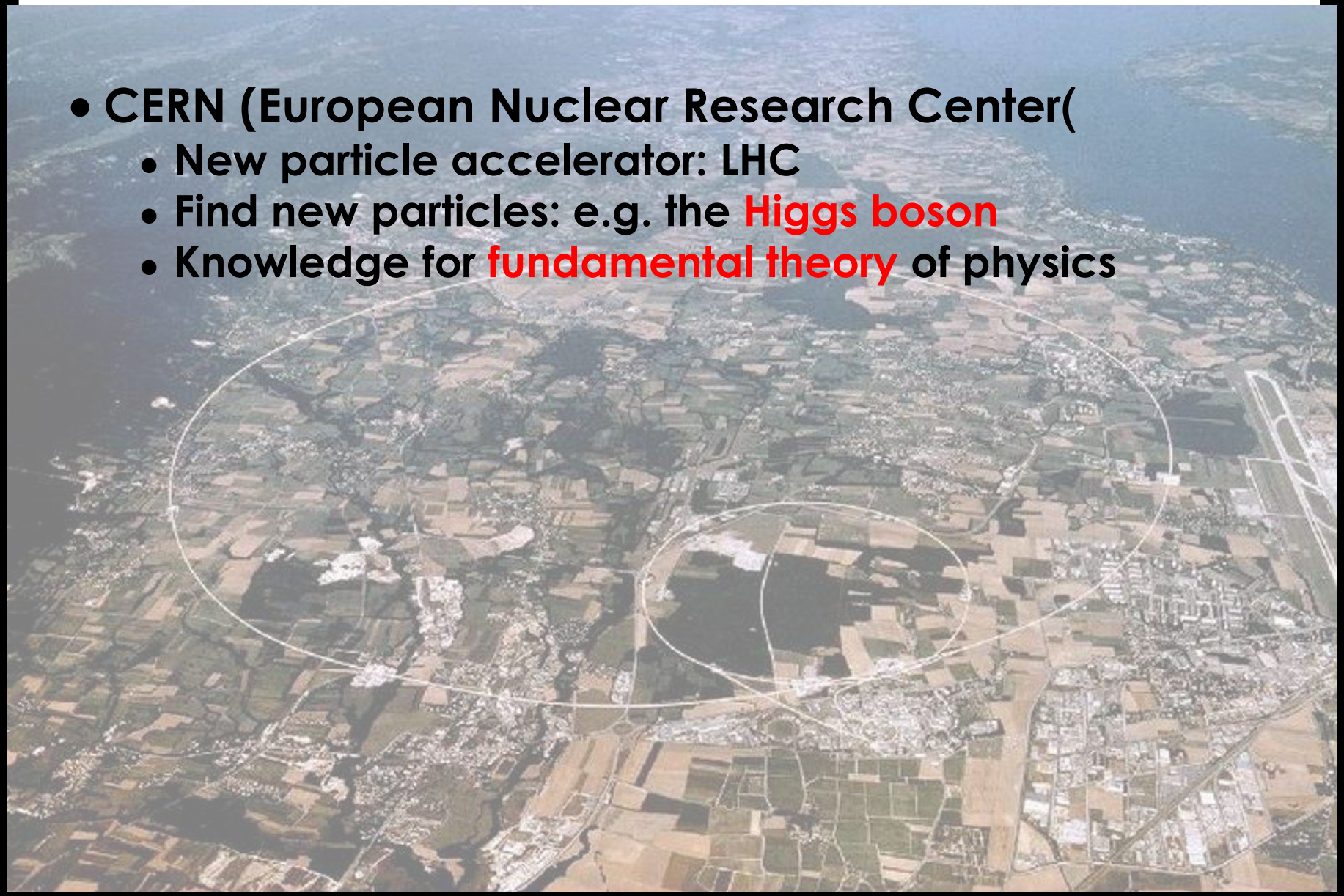
The Grid

Marcus Hardt
SCC / KIT

Outline

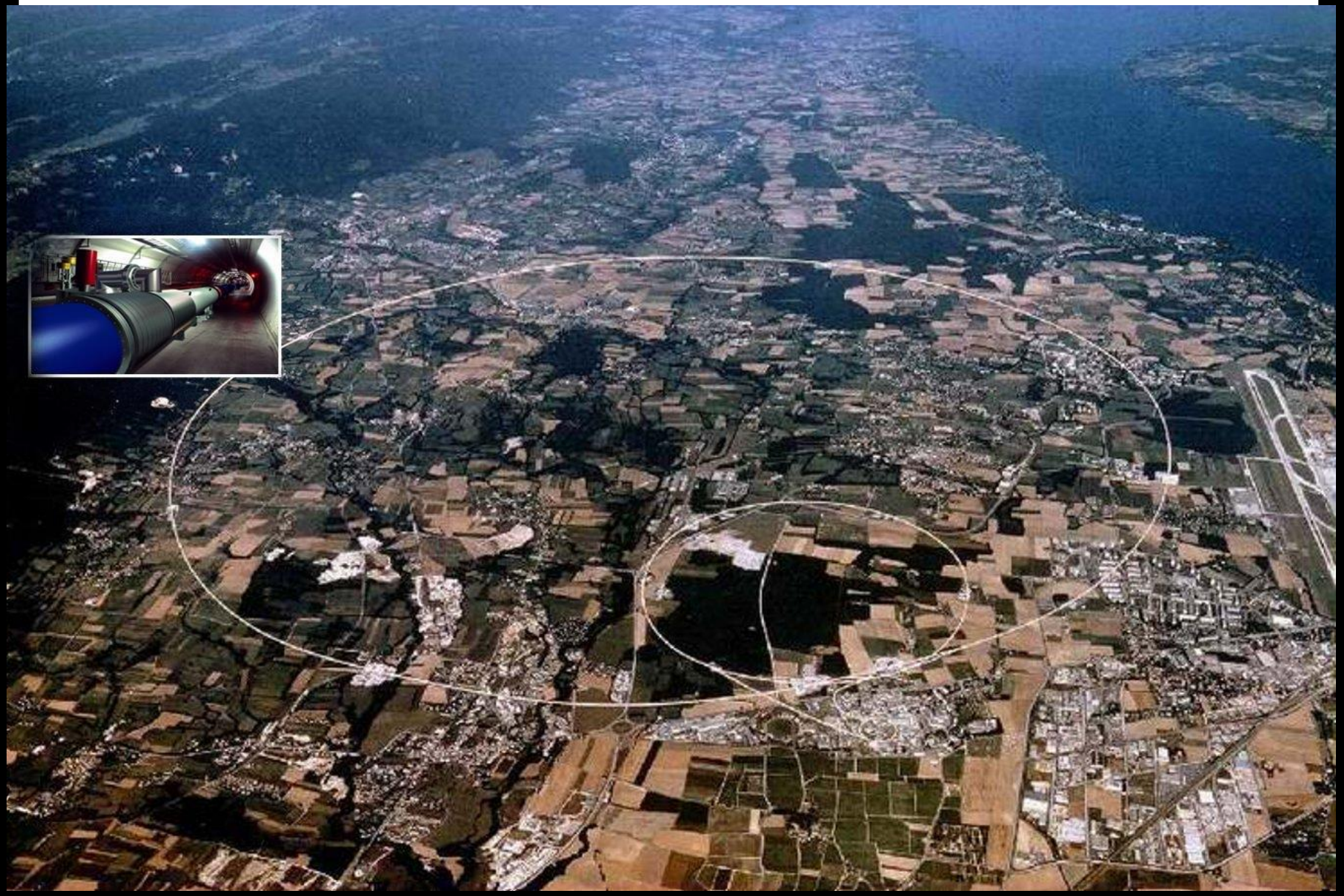
- Why Grid?
- The hardware
- The software
- Current uses on the grid
- Improvements to the grid
- Q&A

2001: Why should we build a grid

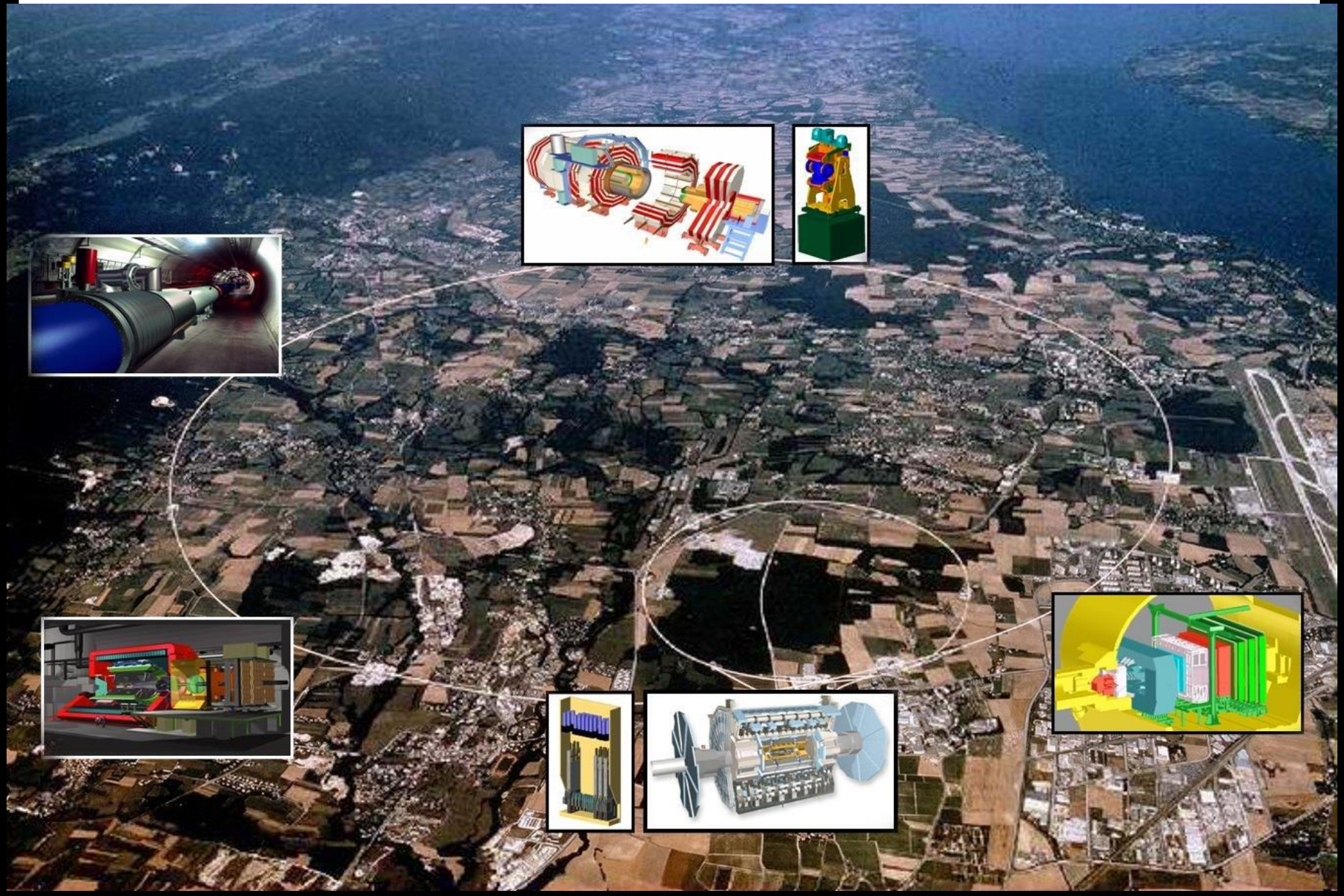


- CERN (European Nuclear Research Center(
  - New particle accelerator: LHC
  - Find new particles: e.g. the Higgs boson
  - Knowledge for fundamental theory of physics

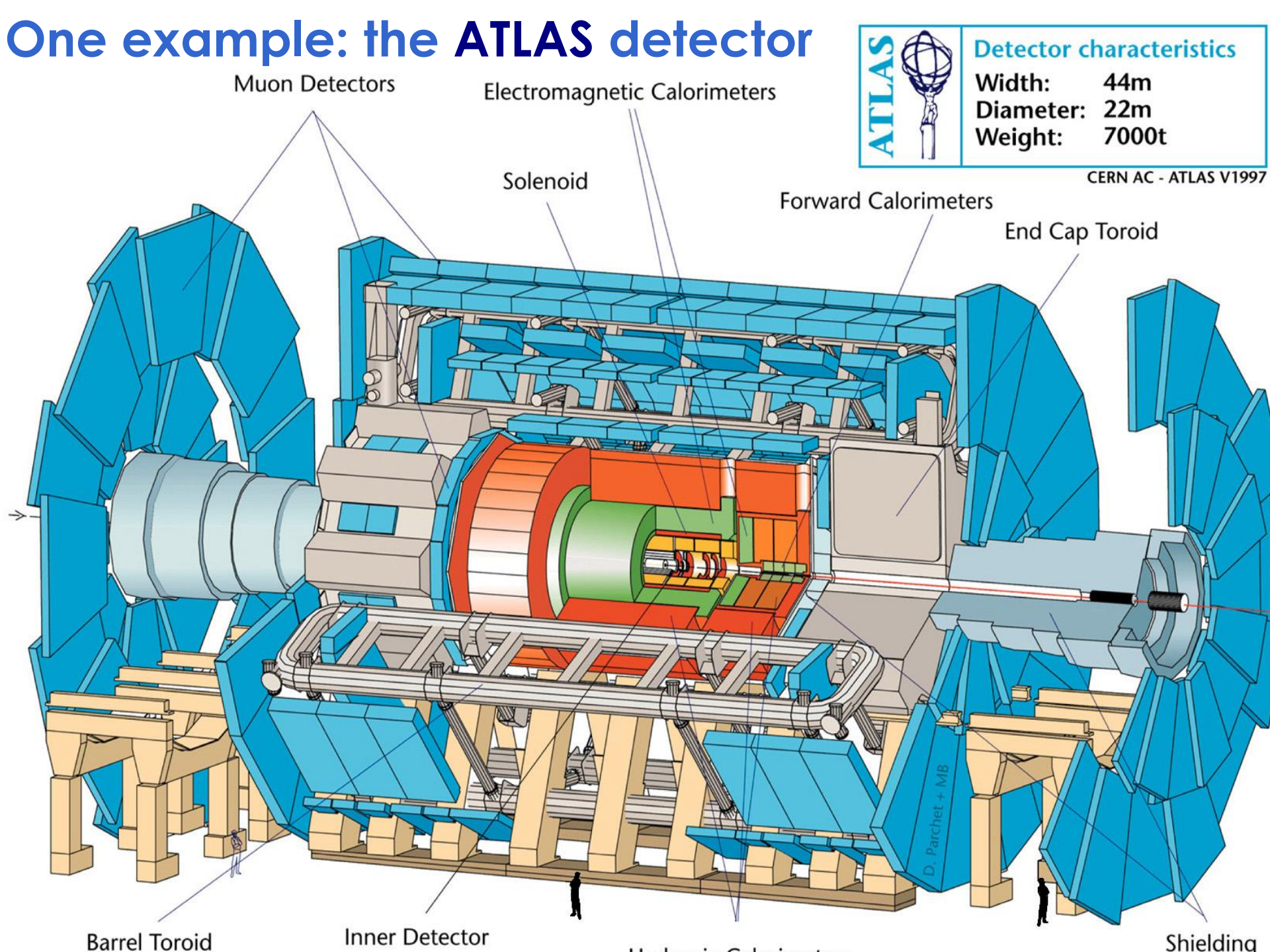
2001: Why should we build a grid

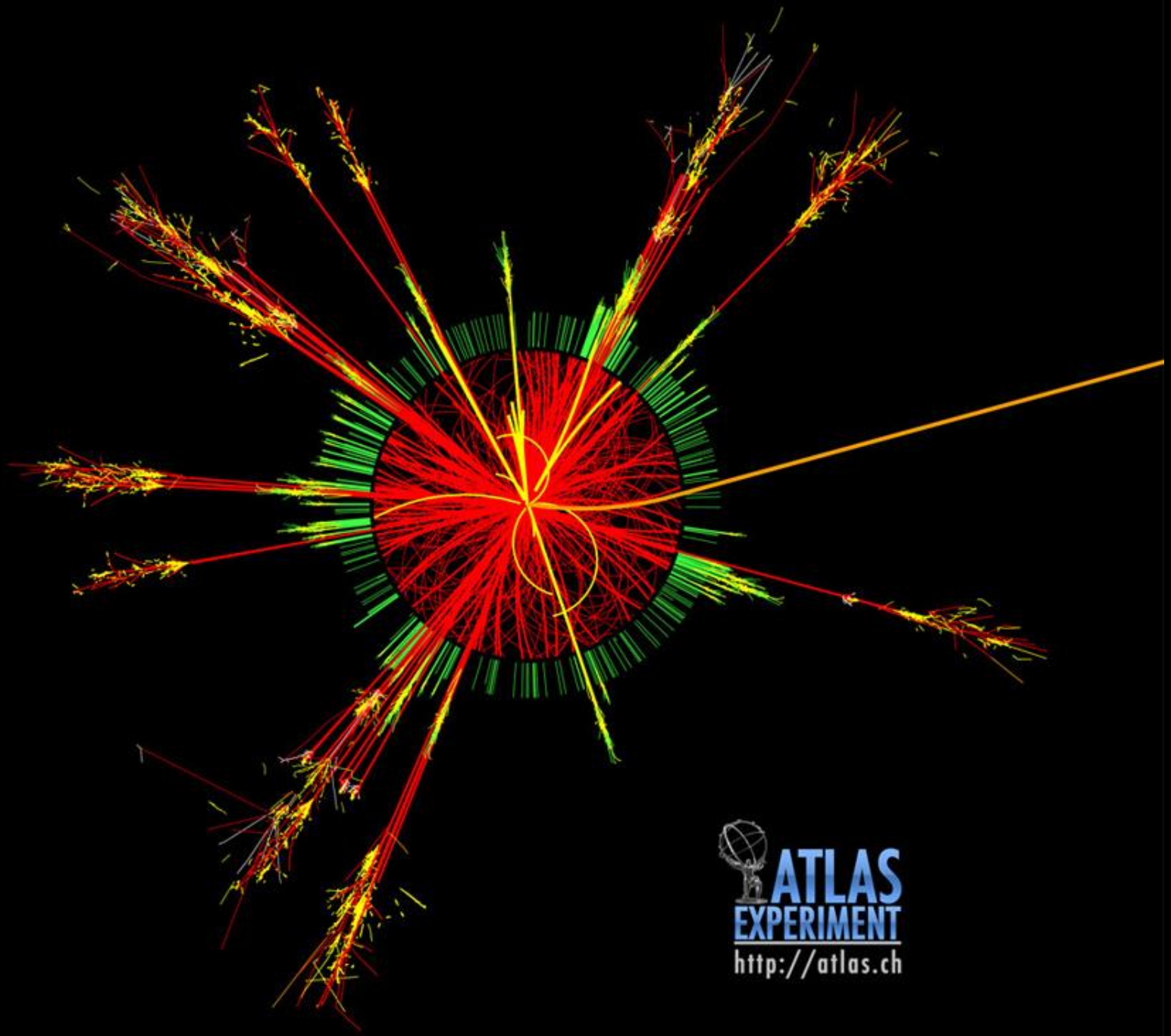


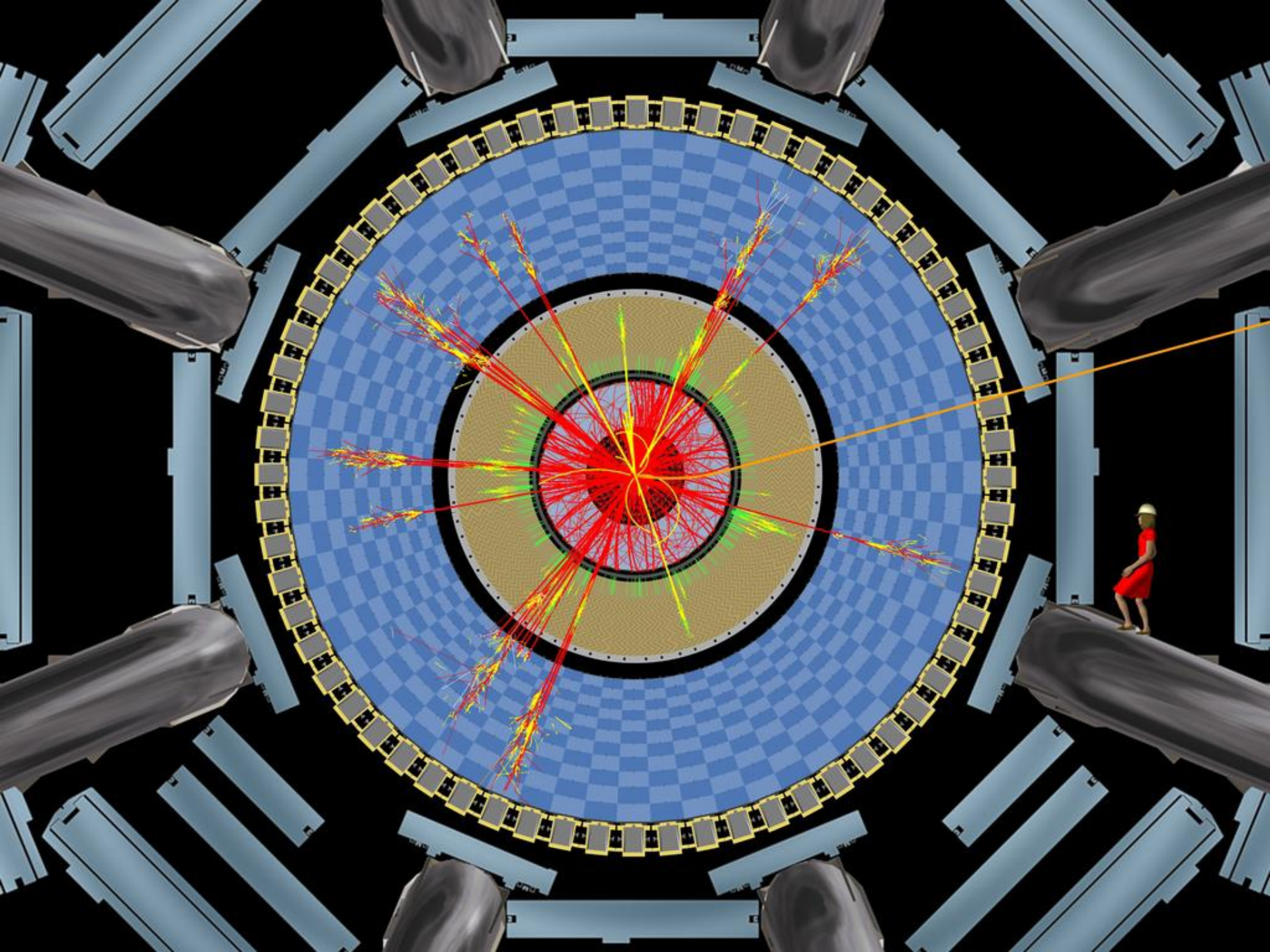
2001: Why should we build a grid



One example: the ATLAS detector



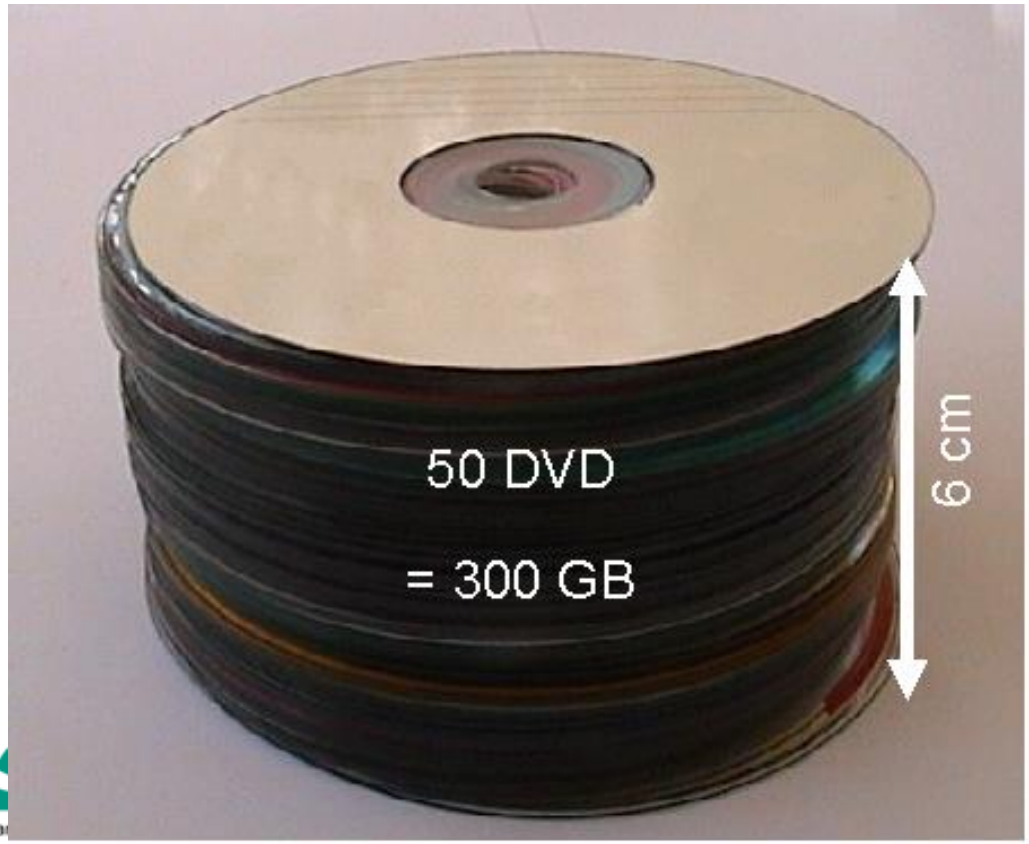

ATLAS
EXPERIMENT
http://atlas.ch

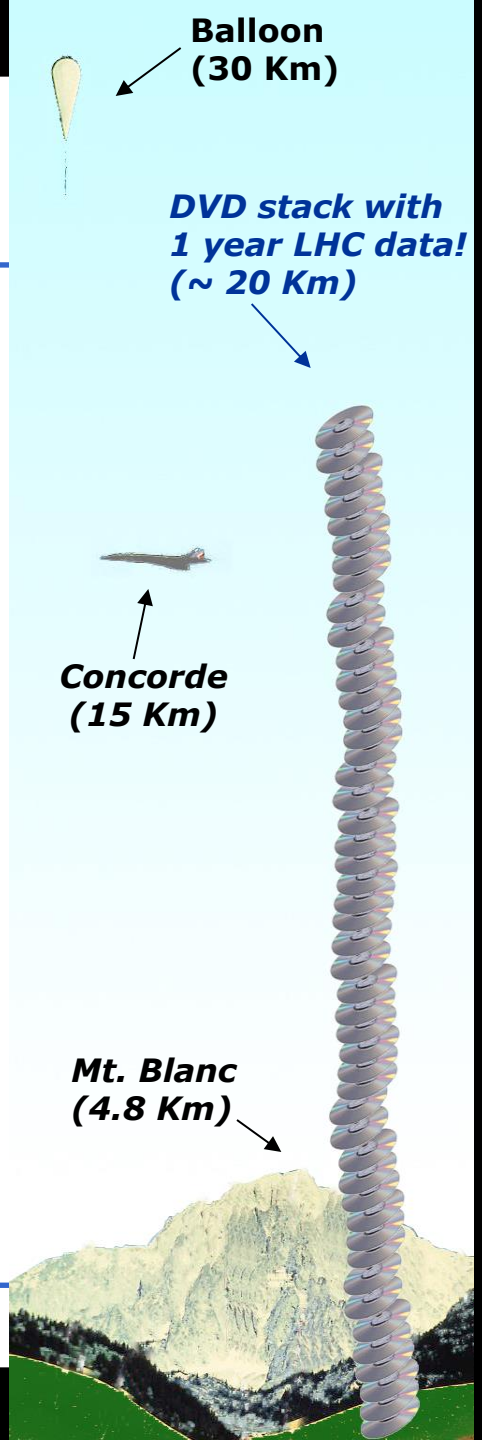


Data Volume

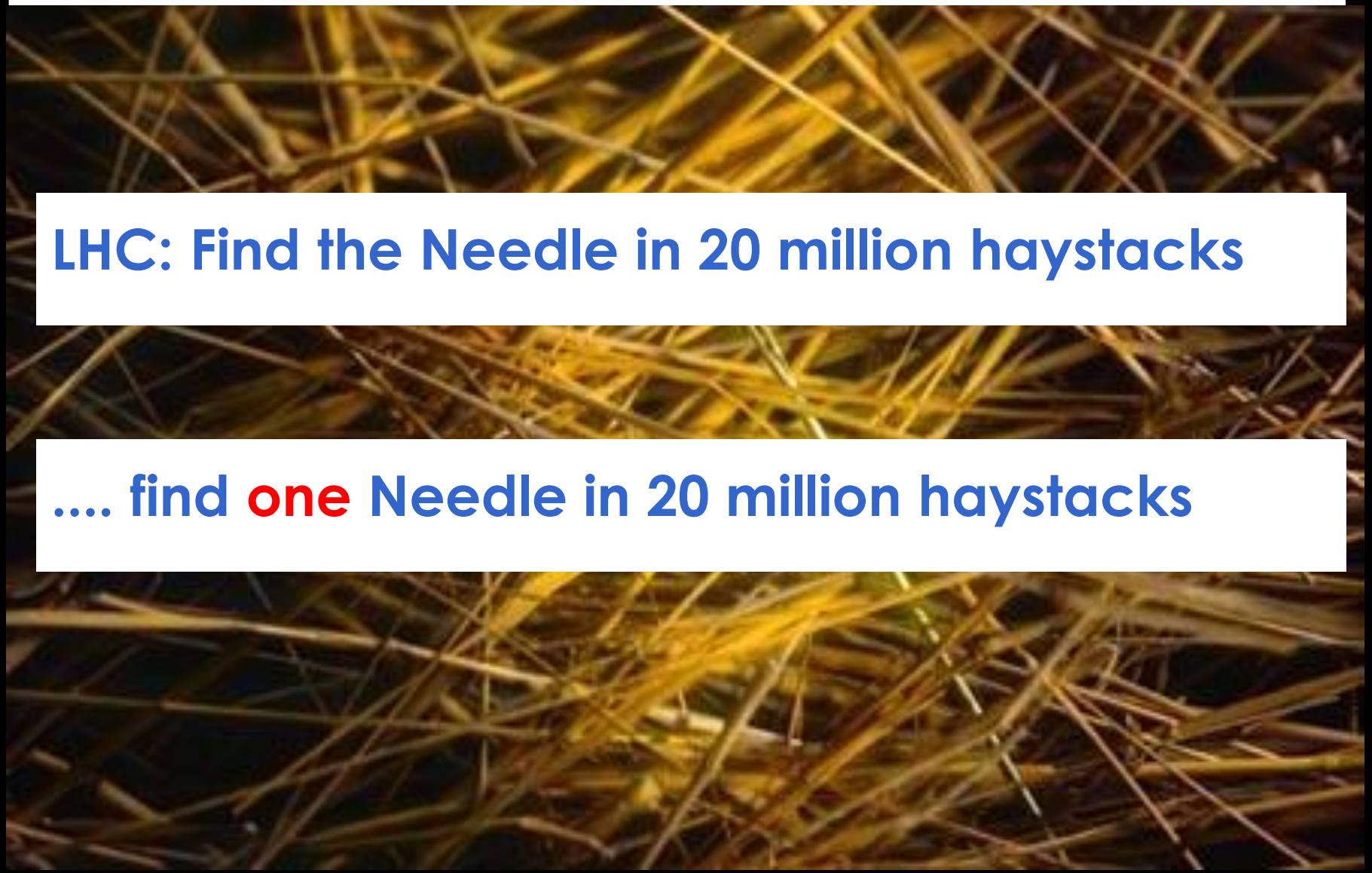
Annual data storage:

10 PetaBytes/year





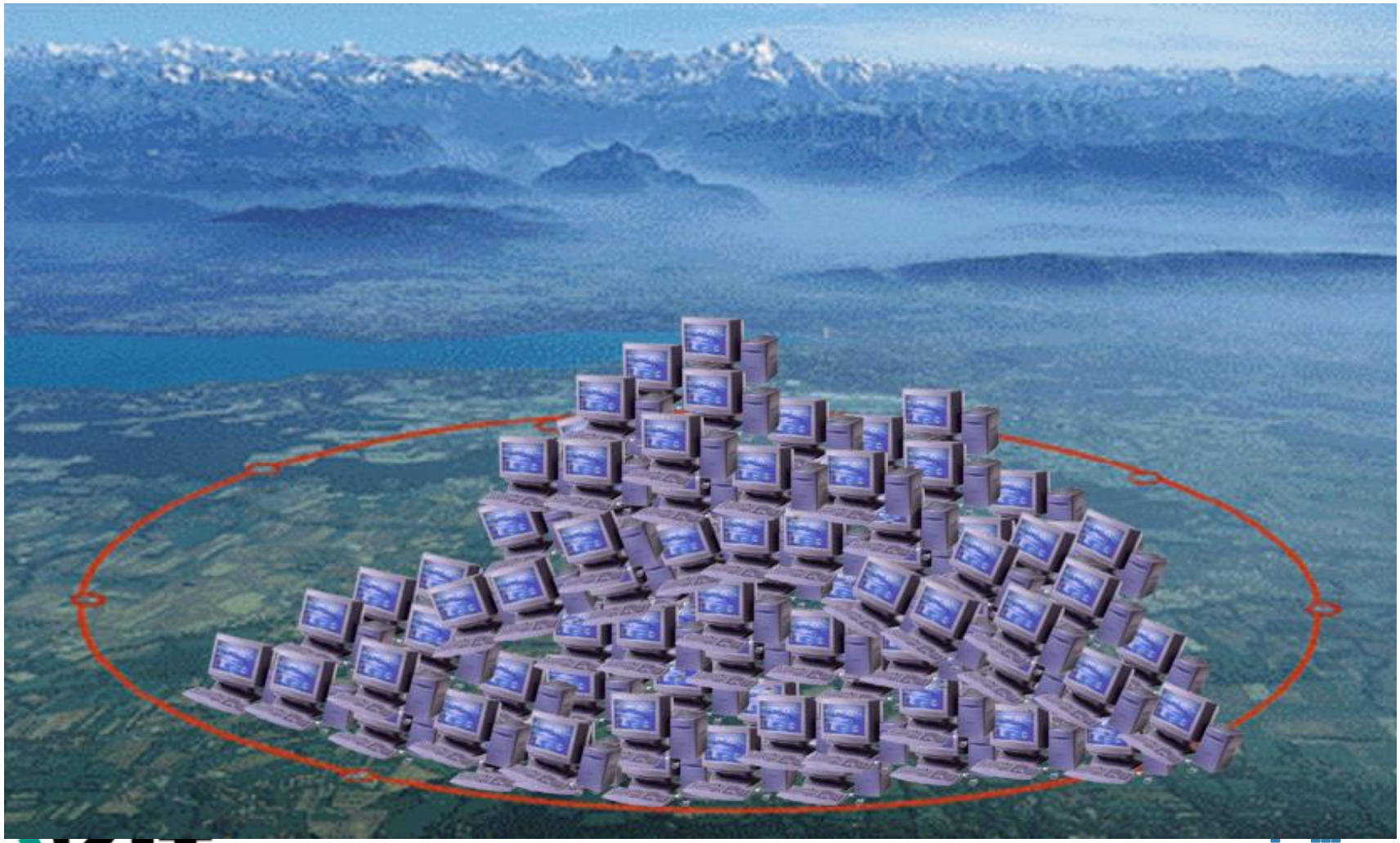
Easy: Find the Needle in a haystack



LHC: Find the Needle in 20 million haystacks

.... find one Needle in 20 million haystacks

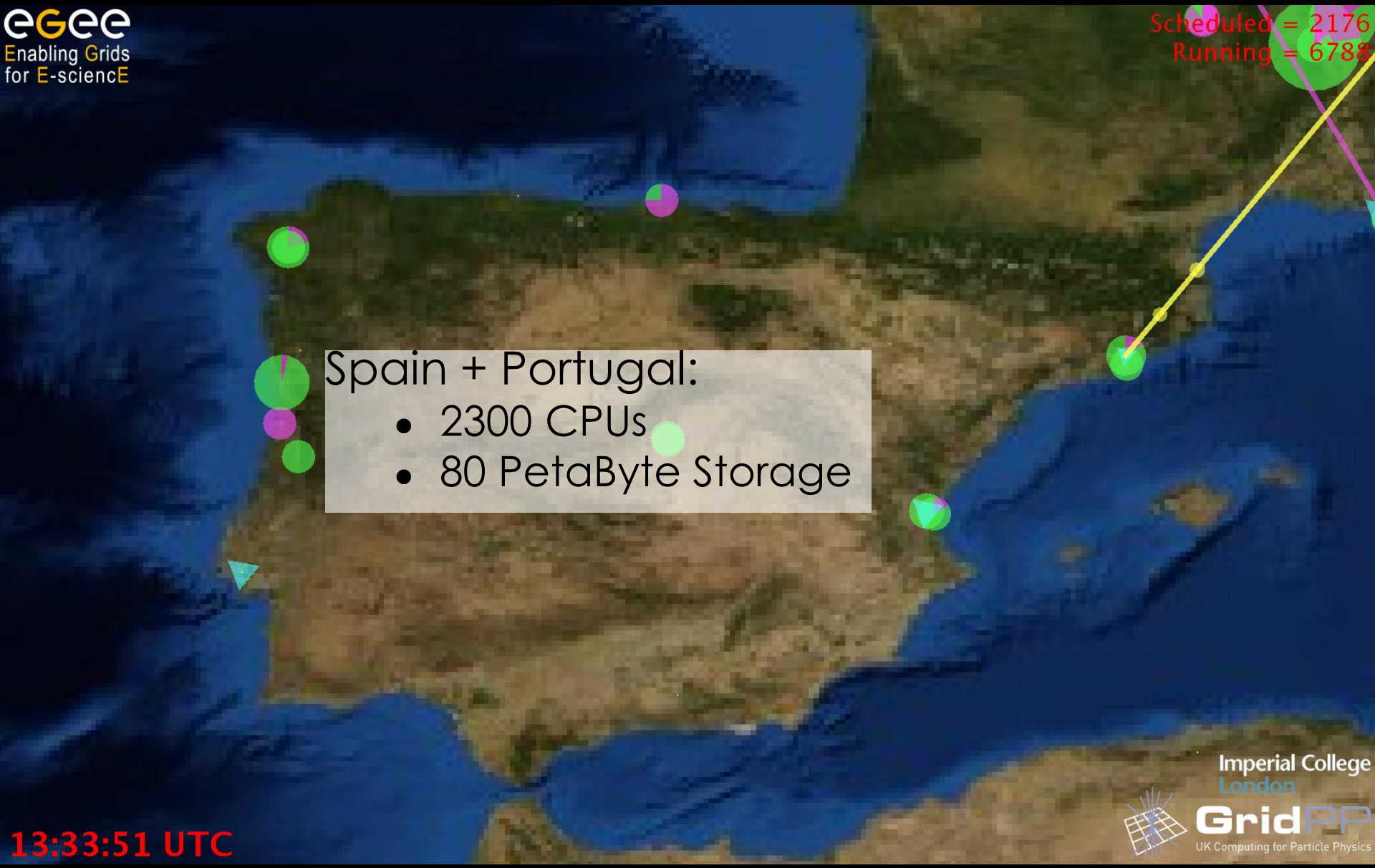
How to provide the required hardware?



The infrastructure

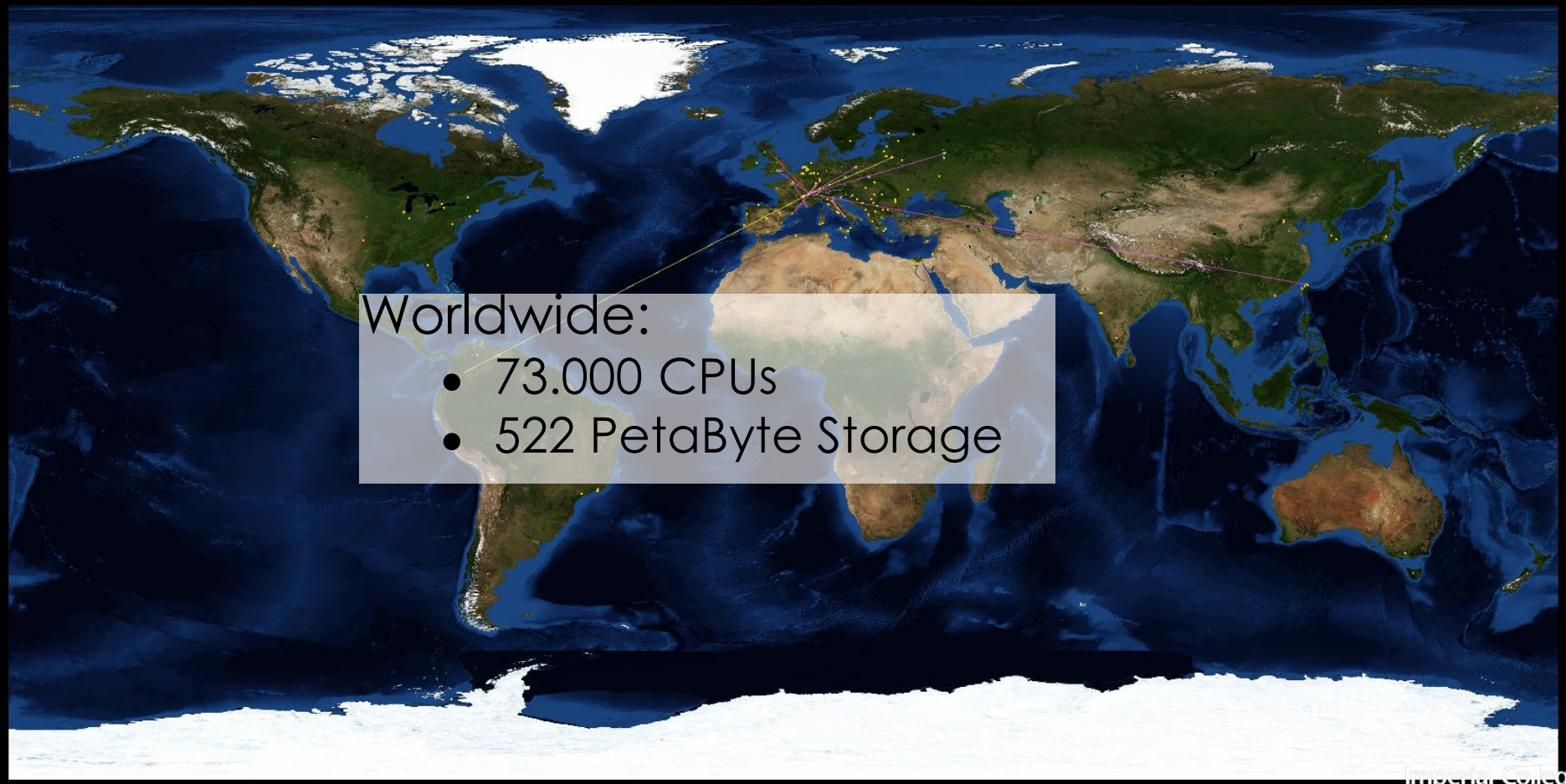
- Hierarchical Organisation:
  - Tier0: CERN
  - Tier1: 12 Big Computer Centres (Spain: PIC, Barcelona)
  - Tier2: Universities
- Organised by regions
  - Southwest Europe (= Spain + Portugal)
  - Southeast Europe
  - Germany/Switzerland
  - France...
  - ....


eGee
Enabling Grids
for E-sciencE
Scheduled = 2176
Running = 6788
13:33:51 UTC
Imperial College
London
GridPP
UK Computing for Particle Physics


eGee
Enabling Grids
for E-sciencE
Scheduled = 2176
Running = 6788
Spain + Portugal:
• 2300 CPUs
• 80 PetaByte Storage
Imperial College
London
GridPP
UK Computing for Particle Physics
13:33:51 UTC

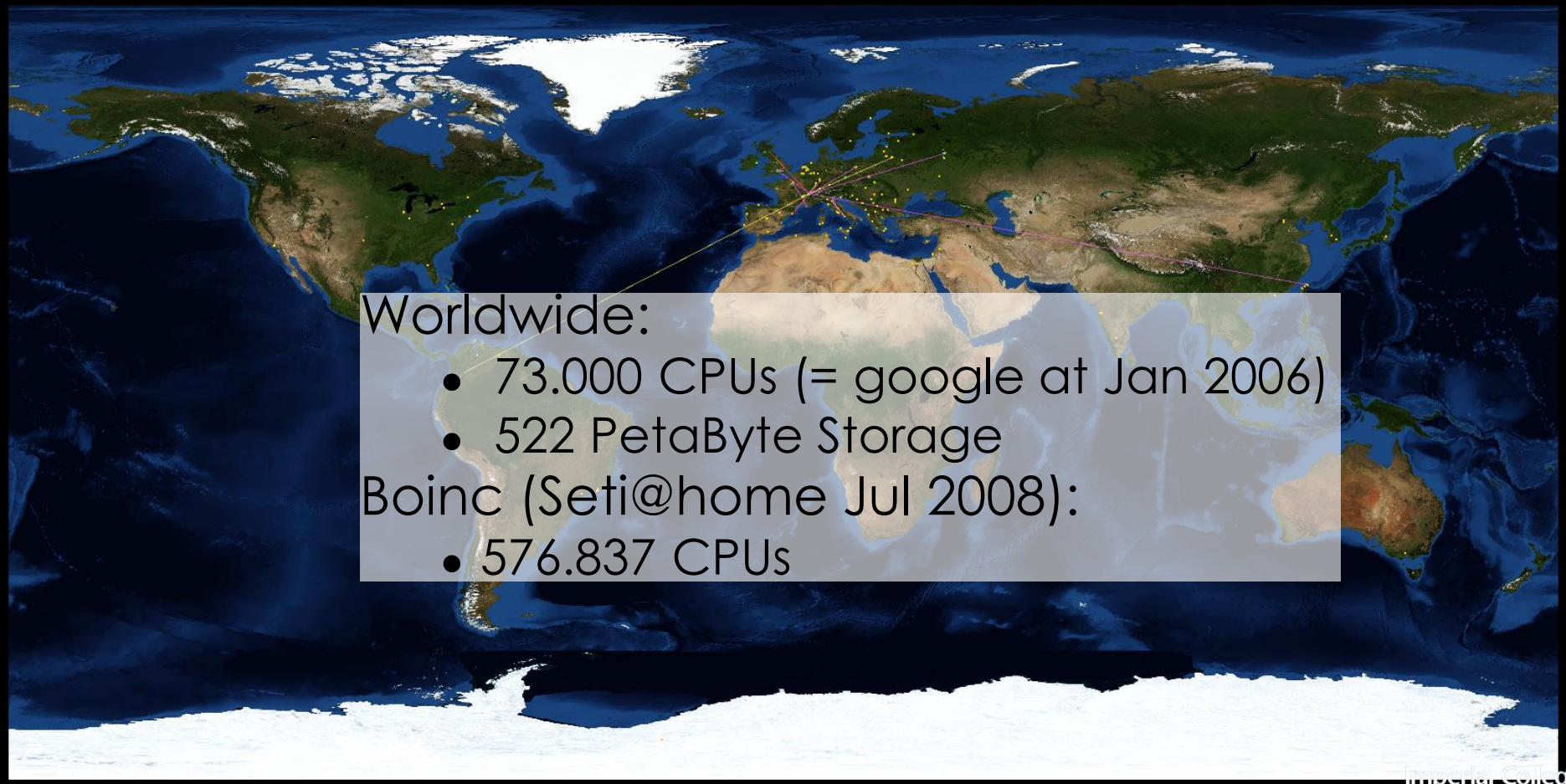

IFCA
Instituto de Física de Cantabria
CSIC
IFIC
INSTITUT DE FÍSICA CORPUSCULAR

eGee
Enabling Grids
for E-sciencE



13:24:29 UTC

Imperial College
London
GridPP
UK Computing for Particle Physics

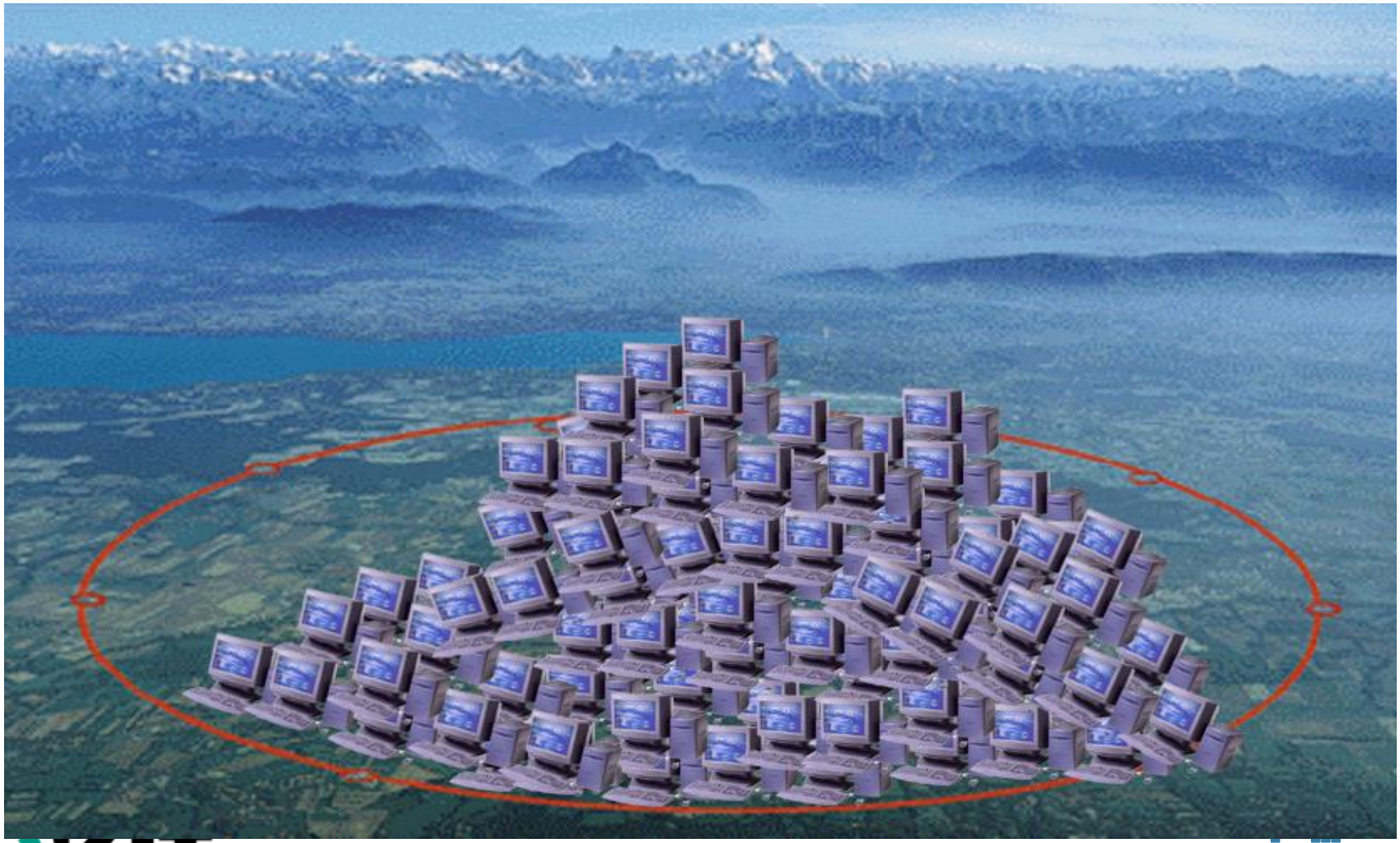


eGee
Enabling Grids
for E-sciencE

Imperial College
London
GridPP
UK Computing for Particle Physics

13:24:29 UTC

How to provide the required hardware?




How to organise the hardware?

Grid middleware

- Middleware
  - Software between application and operating system
- gLite
  - Grid middleware
  - Developed with EU-funding
  - High Energy Physics community (**CERN**)
- gLite architecture
  - gLite defines a set of building blocks
    - (CE, SE, UI, MON, RB, CA, VO, VOMS, BDII)

Open Source?

- Open Source?
  - Sure! What else!
- Reasons:
  1. Grid is **strategic platform** on which EU wants to provide computer infrastructure for science
  2. We don't want to develop the same thing twice
  3. We want others to participate
  4. We use taxpayers money
- License
  - Based on BSD-License
    - Lacks "mention everybody"
    - Adds "return bugfixes"

Access to resources

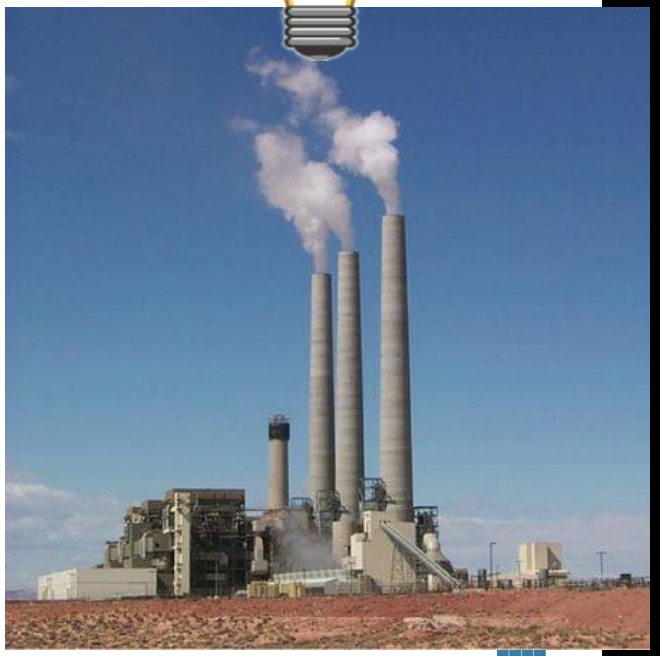
- Authentication
  - Certification Authority (CA) infrastructure
  - Who are you?
    => X.509 Certificate
- Authorisation
  - What are you working on? Which Group?
    => Virtual Organisation (VO)
  - Which role do you play?
    => User, Software manager, ...
  - Which resources allow access for you?
    => **Virtual Organisation** concept

The name GRID

- Analogy to electrical power grid
  - Plug a lamp into the wall and get light
  - Plug a computer into the network and get power

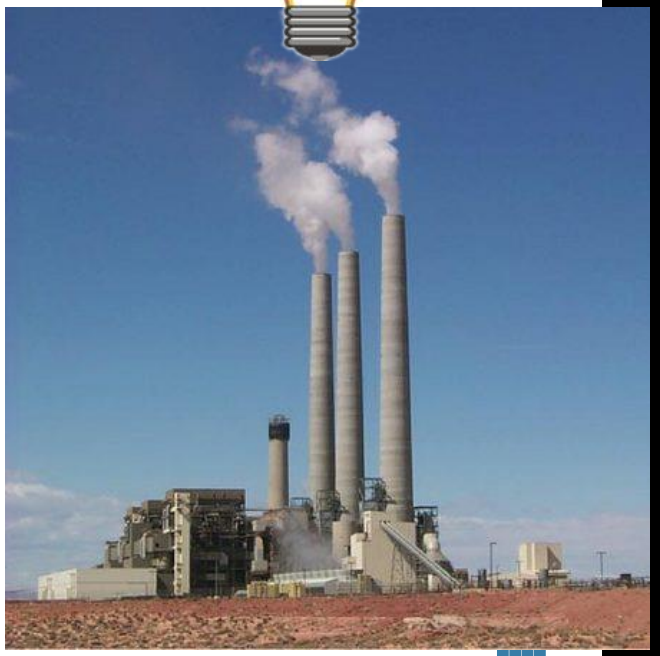
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...



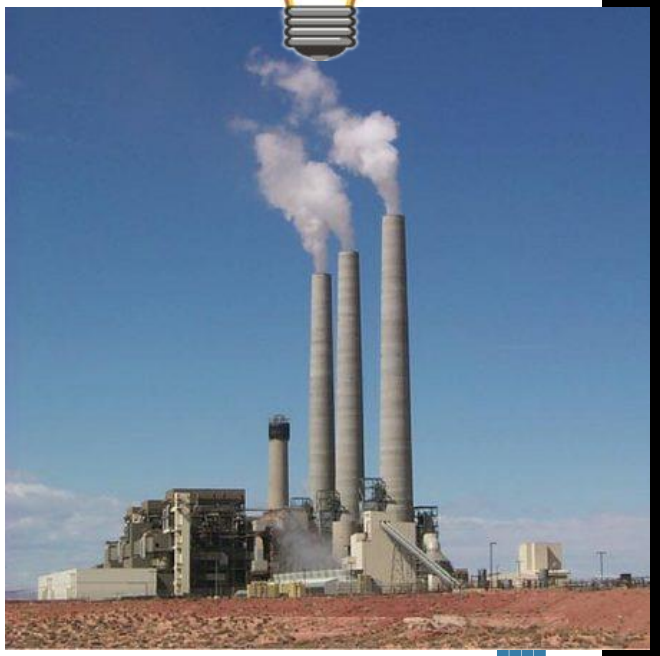
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...
- Submit request for electricity to broker



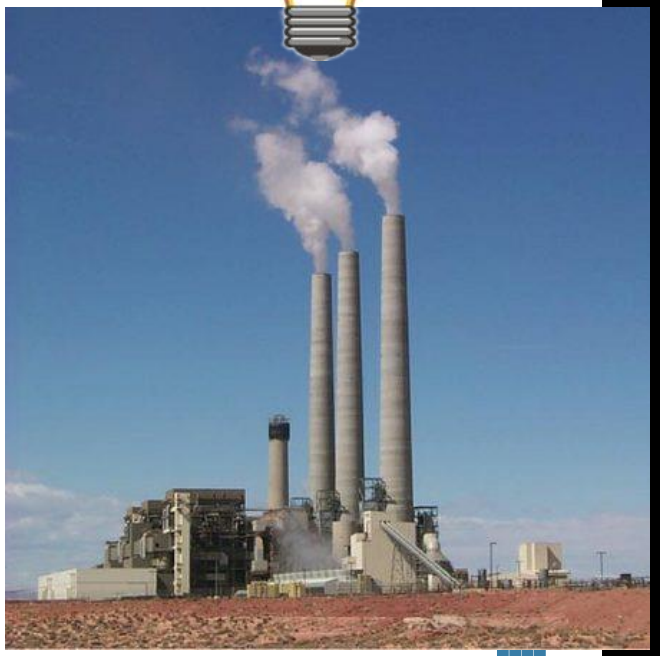
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...
- Submit request for electricity to broker
  => Powerplant chosen for you



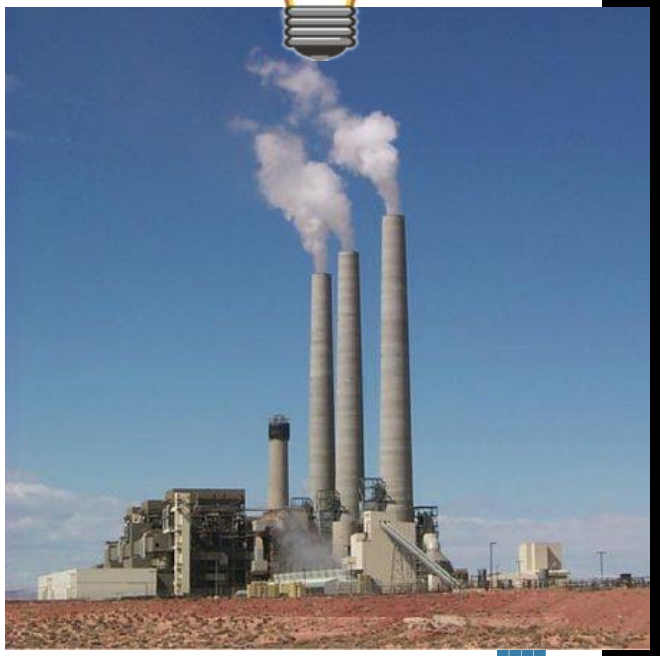
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...
- Submit request for electricity to broker
  => Powerplant chosen for you
  => Send lightbulb to powerplant



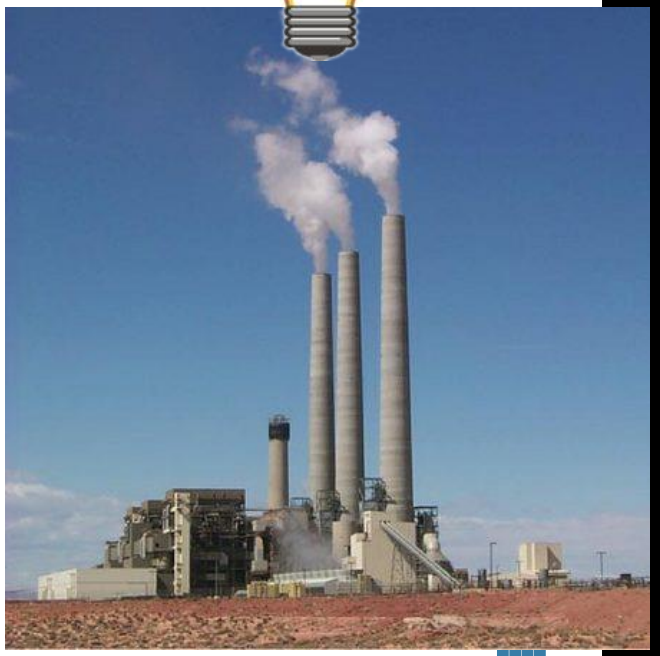
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...
- Submit request for electricity to broker
  => Powerplant chosen for you
  => Send lightbulb to powerplant
  => Wait for electricity



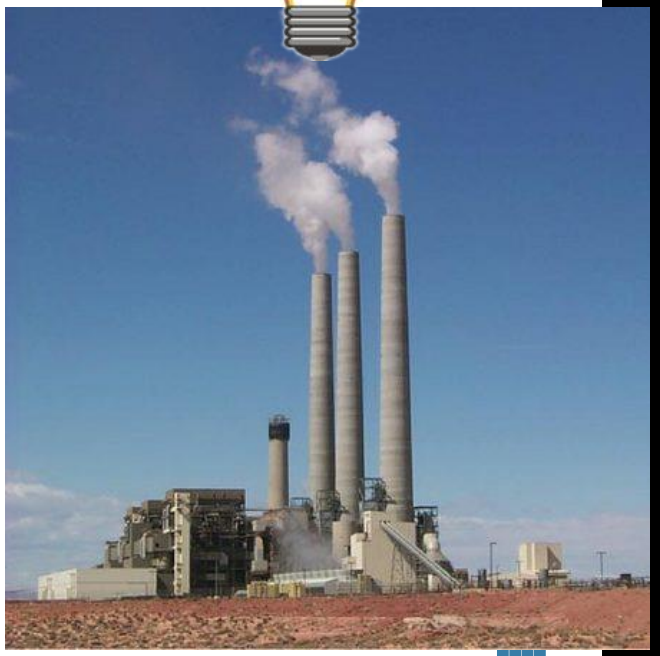
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...
- Submit request for electricity to broker
  => Powerplant chosen for you
  => Send lightbulb to powerplant
  => Wait for electricity
  => Lightbulb glows



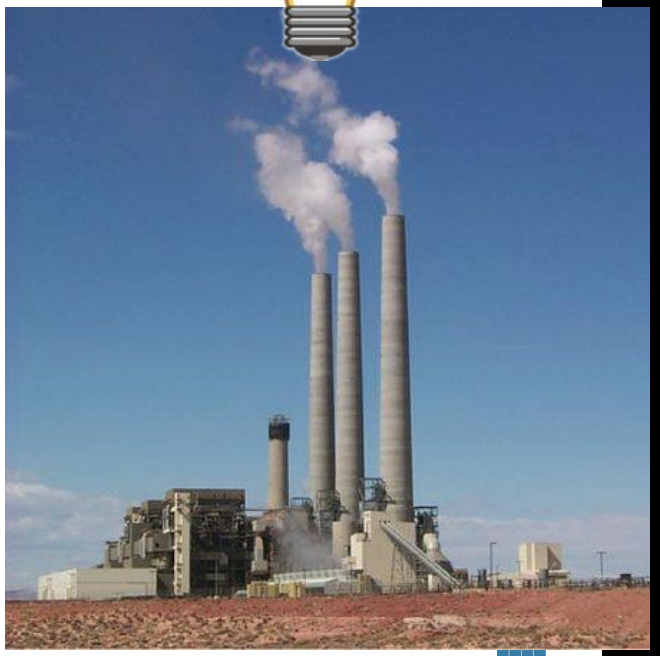
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...
- Submit request for electricity to broker
  => Powerplant chosen for you
  => Send lightbulb to powerplant
  => Wait for electricity
  => Lightbulb glows
- Results come back



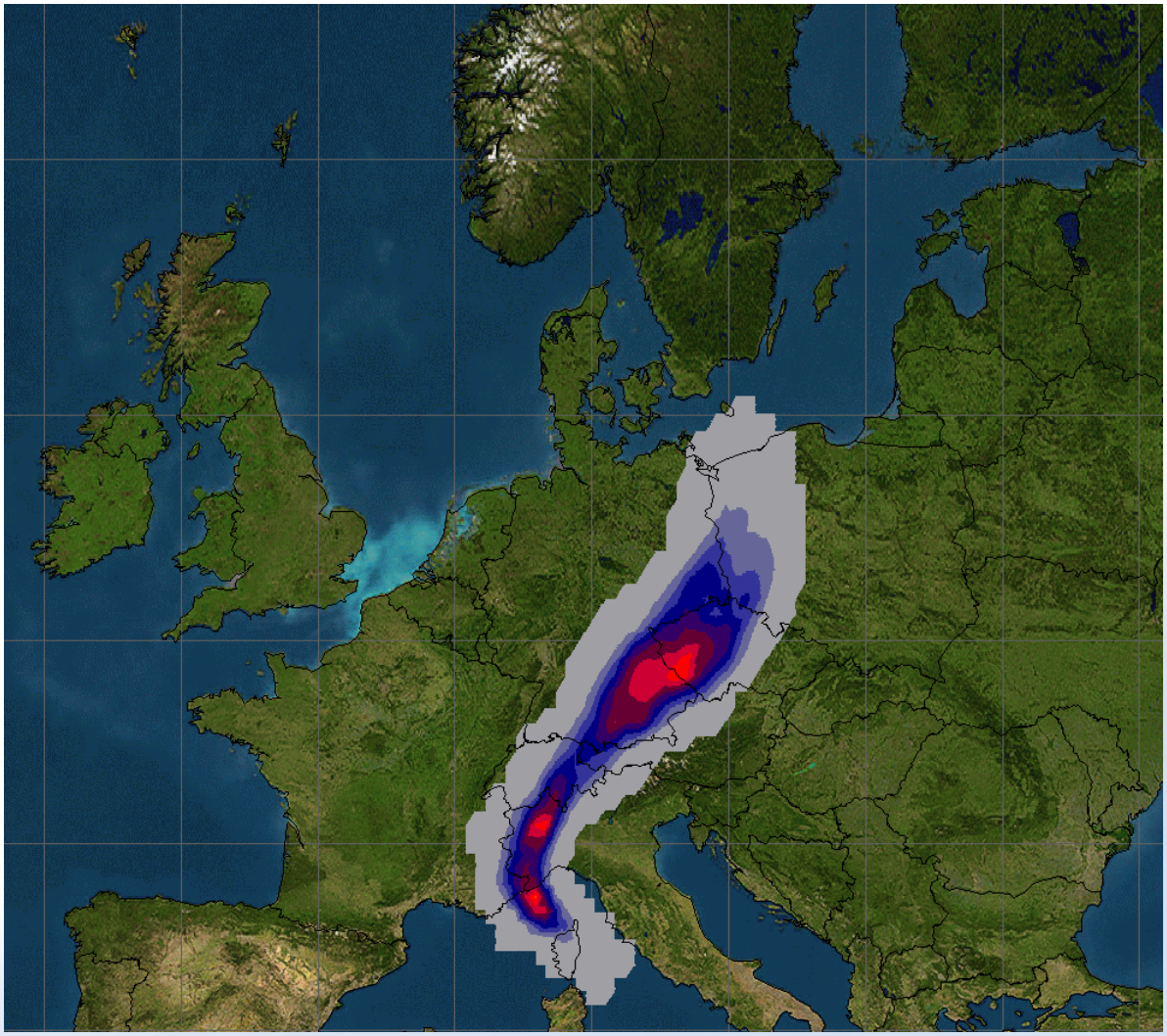
Using a lightbulb in the job based grid world

- Describe the lightbulb
  Voltage, Watts, Amount
  Lighting_time, ...
- Submit request for electricity to broker
  => Powerplant chosen for you
  => Send lightbulb to powerplant
  => Wait for electricity
  => Lightbulb glows
- Results come back
  - About 20% of the bulbs broken




Intermediate Summary

IMS Model Suite
Mass Concentration 33h after Release
Bordeaux



T + 33h

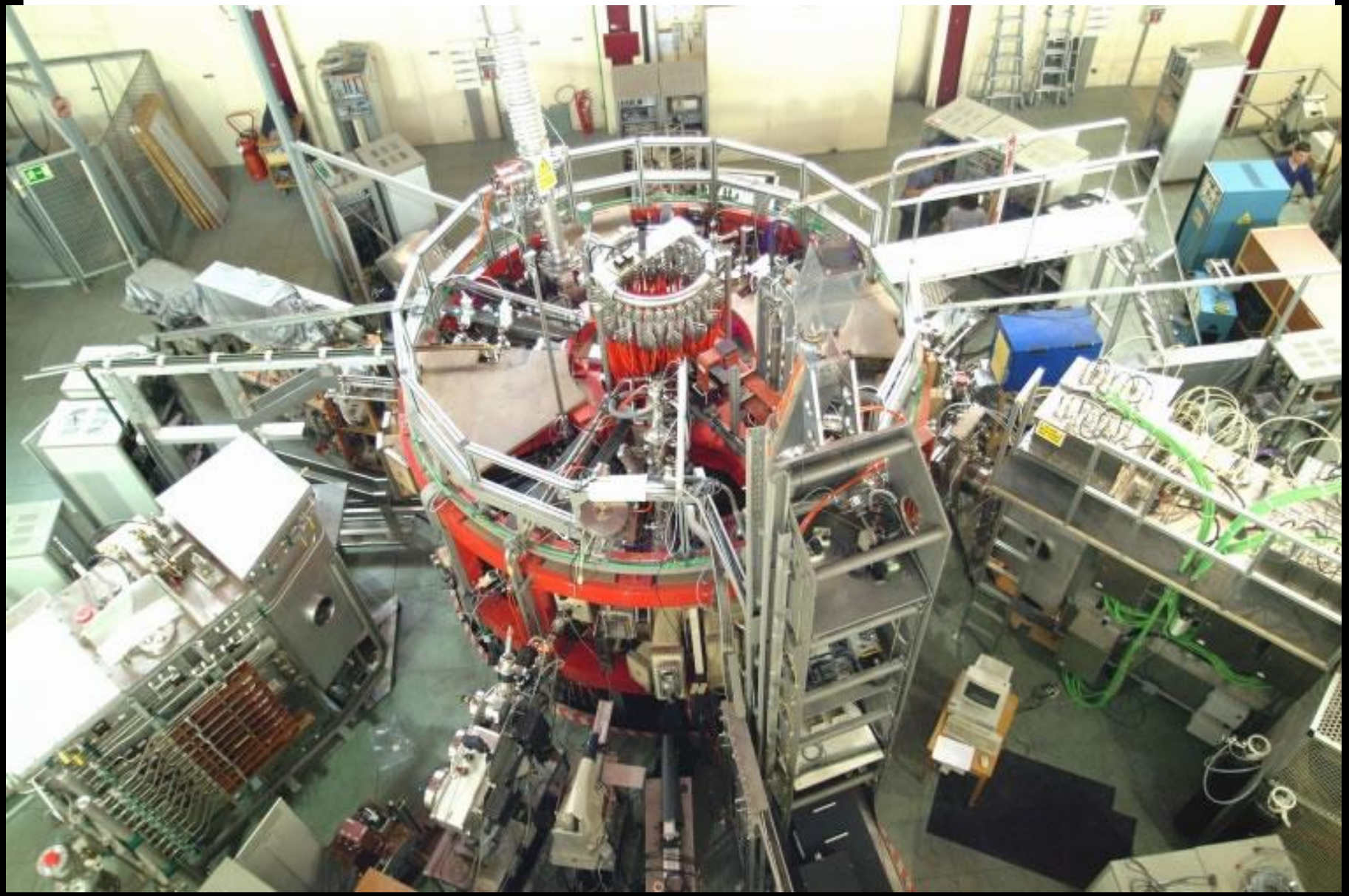
ng/m³

Volumetric concen-tration in ng/m³

Interval	Color
(-INF, 0)	
<0, 0.2)	
<0.2, 0.4)	
<0.4, 0.6)	
<0.6, 0.8)	
<0.8, 1)	
<1, 1.2)	
<1.2, 1.4)	
<1.4, 1.6)	
<1.6, 1.8)	
<1.8, 2)	
<2, +INF)	

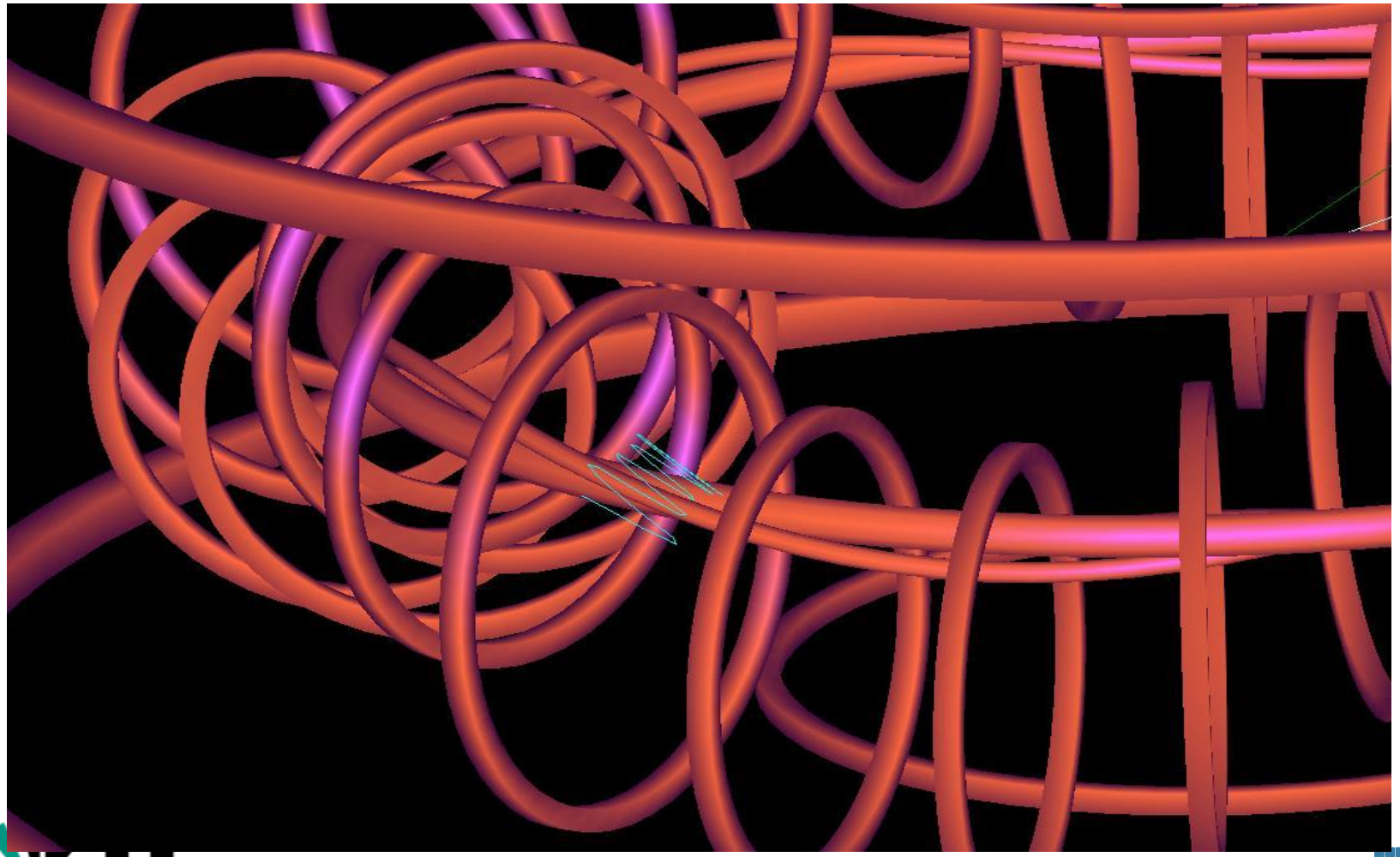
Slide courtesy Isabel Campos, IFCA

Fusion Reactor "TJ-II" in Zaragoza

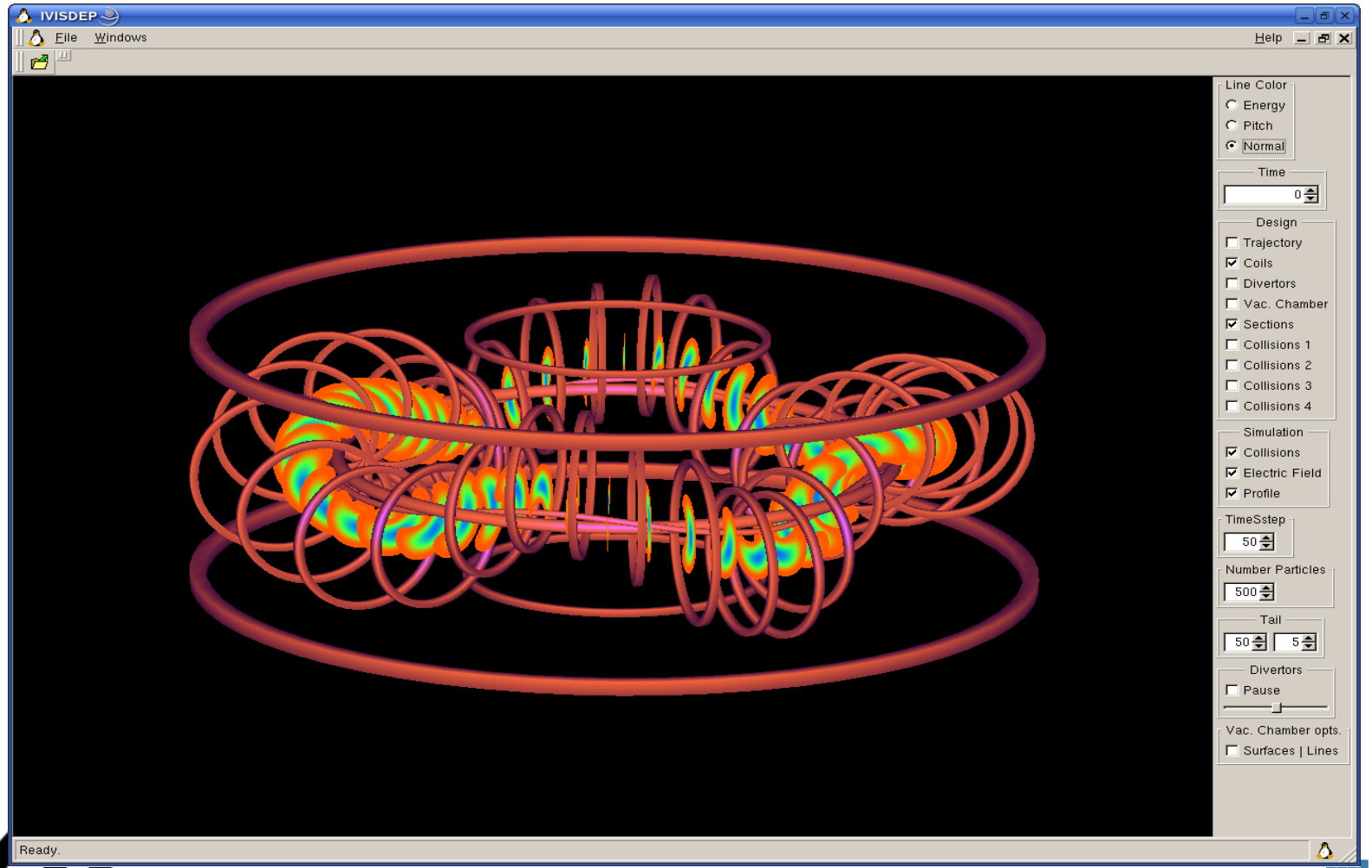


Slides courtesy Ruben Vallez, Bifi, Uni Zaragoza

Trayectorias individuales



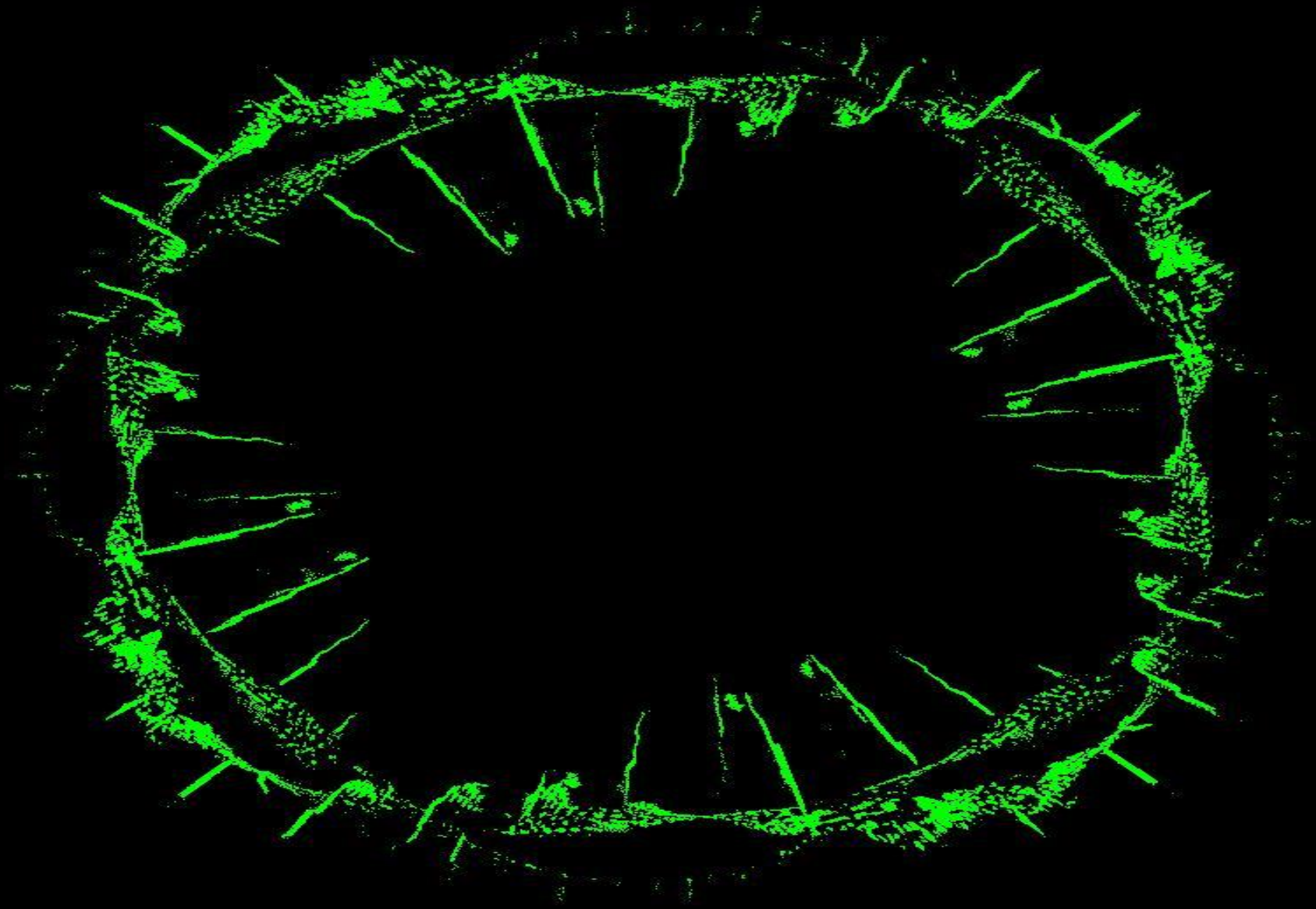
Interactive Visualizator for ISDEP



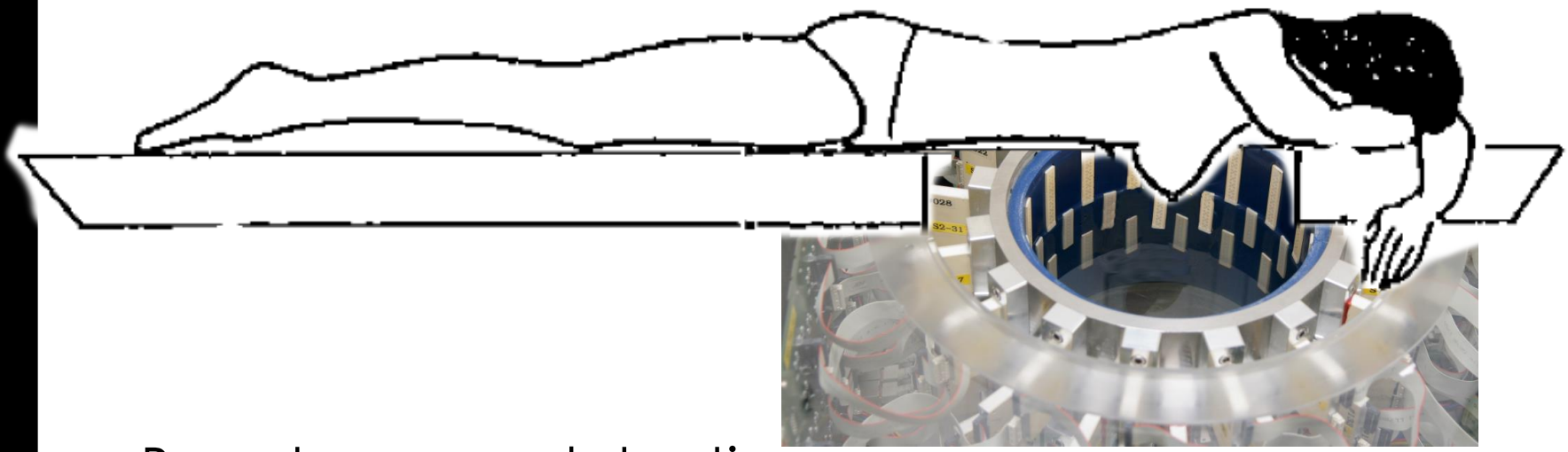
Karlsruhe Institute of Technology

int.eu.grid

Puntos de choque con la cámara de vacío



Ultrasound CT



- Breast cancer detection
- Ultrasound is difficult to trace
- Intelligent algorithms required
  - Intelligence requires CPU time

Ultrasound CT Characteristics

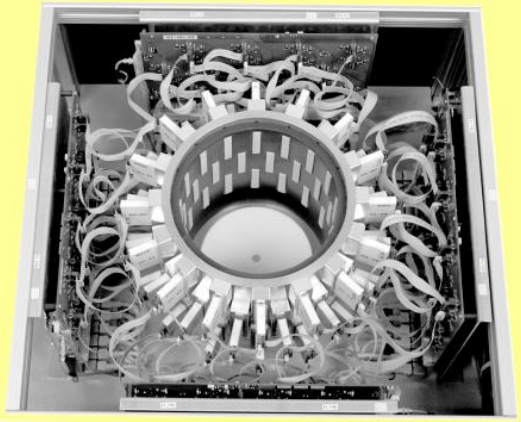
- Data: 20 GB (one full measurement)
- Computing time depends
  - On output size / resolution
  - amount of input data

- Currently possible:
  - 2 GB input
  - 100x1024^2 3D-image

10 days of computation on 1 CPU P-IV-2GHz

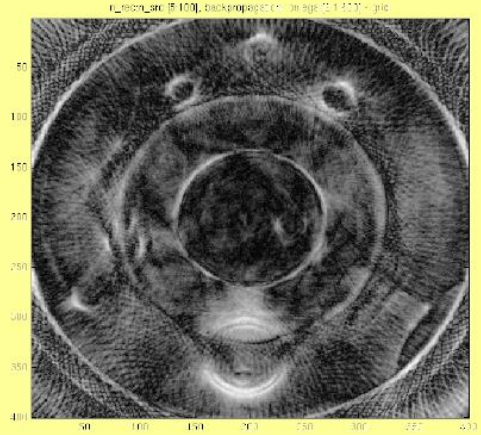
Ultrasound CT on the grid

Phantom
- This phantom is used for testing the measurement hardware and the software algorithms
- For measurement it is placed into the USCT



3D USCT demonstrator
- Measurement device, located under the treatment table on which the patient lies

Grid reconstruction in progress
- Every line of the image is processed on a different CPU of the int.eu.grid infrastructure



Final reconstruction of the phantom
- Speedup by using 40 CPUs of int.eu.grid is roughly 2h versus 60h on a laptop

Problems with the infrastructure

- There are many **excellent** componets!!!
- Just **some** things are bad:
  - Design of gLite was made by & for CERN
  - Submission of jobs is unflexible
  - No API style access
  - Startup of resources takes minutes
  - No interactivity

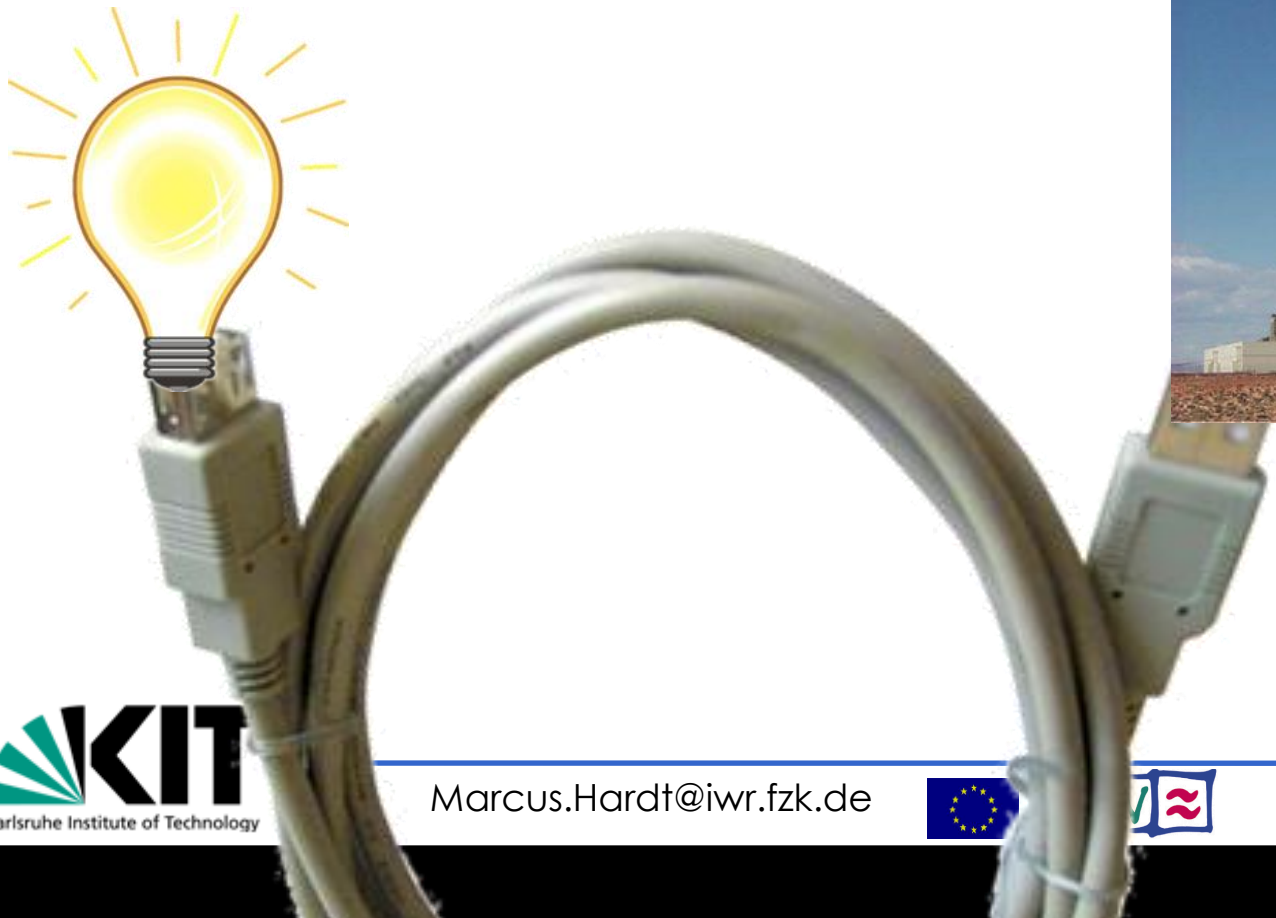
Is there a solution?

Is there a solution?

Yes:
We just need a direct connection

Is there a solution?

Yes:
We just need a direct connection



Solutions

- Several solutions exist
- Just a layer on top of the grid
- Remote Procedure Calls <=> API like access
- One tool:
  - GridSolve

GridSolve interface

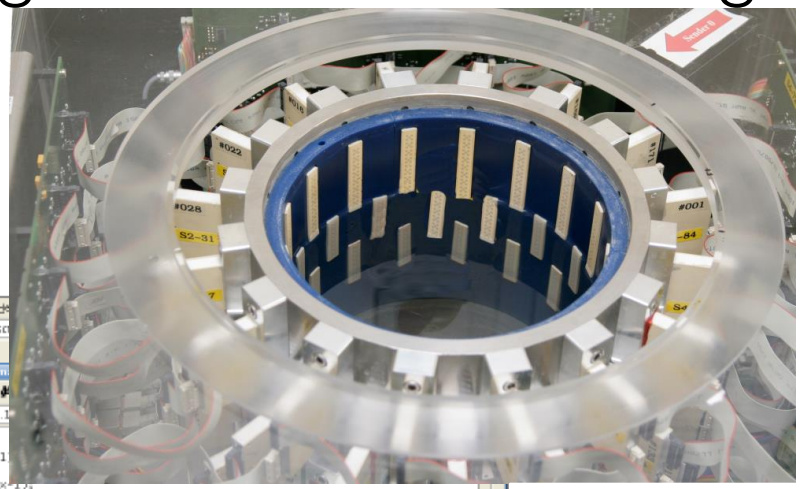
- Client interface for Java, C, C++, Fortran, Matlab, Octave, and many more
- Easy to use:

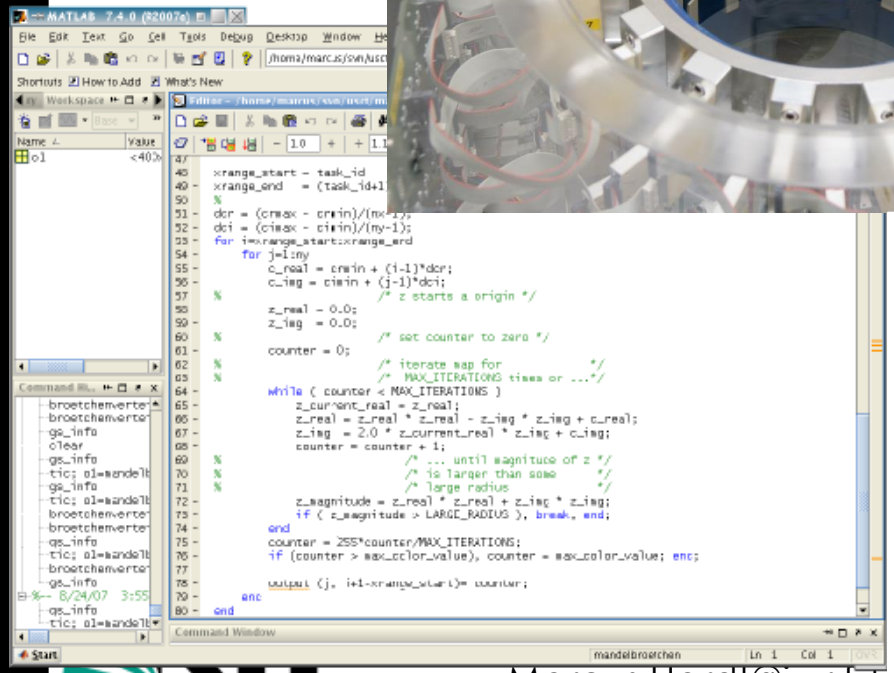
  y=problem(x) <=> y=gs_call('problem', x)

  - Transport input parameters to remote side
  - Execute “problem”
  - Transport result back

Demonstration

- Running Ultrasound CT on the grid






```
xrange_start = task_id
xrange_end   = (task_id+1
dcr = (crmax - crmin)/(mx-1);
dci = (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
```

Life-Demo

- **Life demo** on int.eu.grid

Source code

```
function f=broetchenverteiler_p (N, RESO, MAX_ITERATIONS)
for i=1:N;
        session_id(i)=gs_call_async('maendele', i-1, N, RESO, M
end
while (num_finished < N)
        for i=1:N;
                status(i)=gs_probe(session_id(i));
                if (status(i) == 0 )
                        result=gs_wait(session_id(i));
                end
        end
end
```

Muchas gracias por su atención

What's missing?

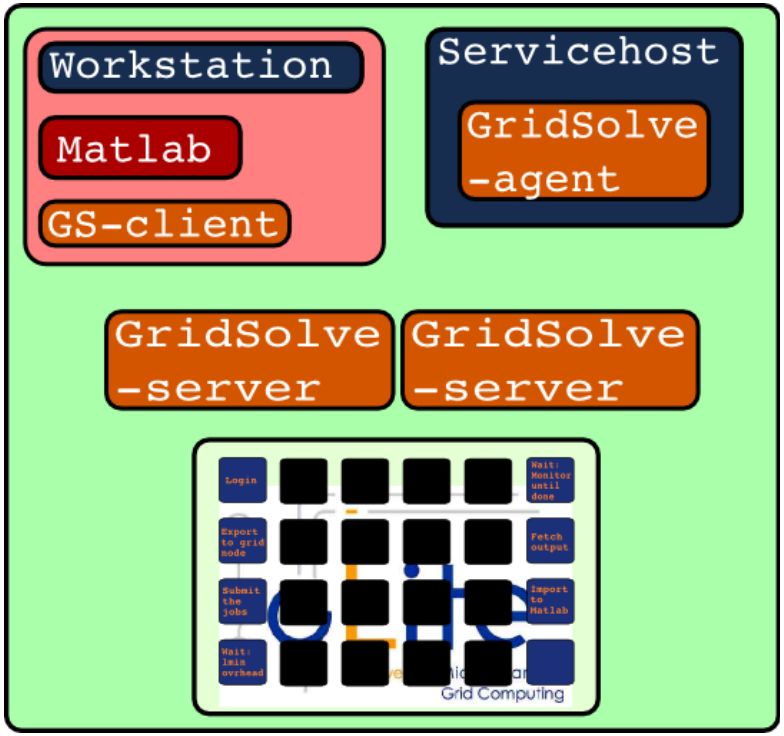
- **Goal:**
  - **Seamless**
  - **Interactive**
  - **Grid access**
  - **From matlab**

- Seamless
  - Don't compile standalone application
- Interactive
  - No overhead (< 10 s)
  - No manual data movem
- From Matlab
  - Run Matlab-functions rer

Example:
Large Excel Table
- **Excel must run locally**
- **Computation in the grid**

How to do it?

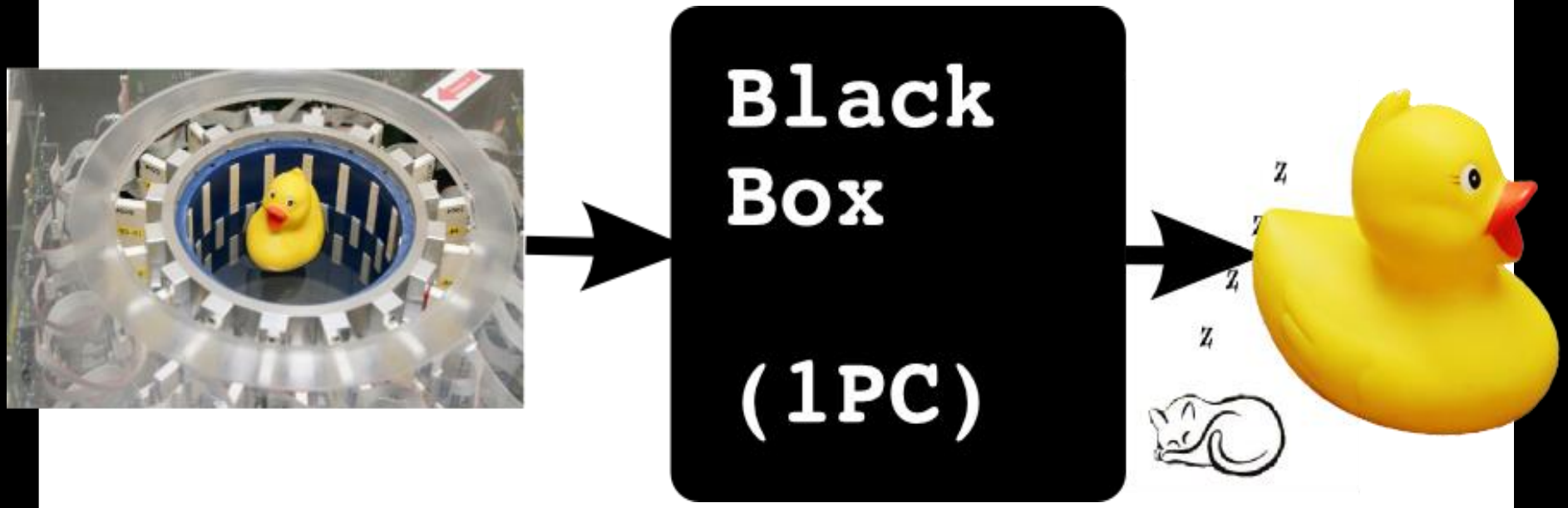
- 1. Make Matlab run on gLite
- 2. Integrate GridSolve with gLite



=> **G**rid **i**n **M**atlab using **G**ridsolv**e** & **R**PC

GIMGER

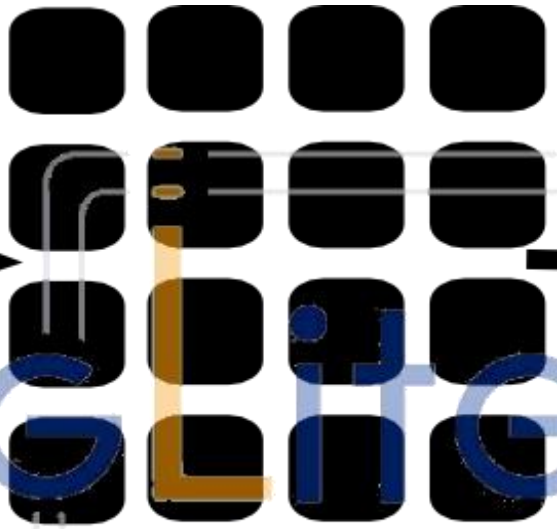
USCT reconstruction := “Black Box”

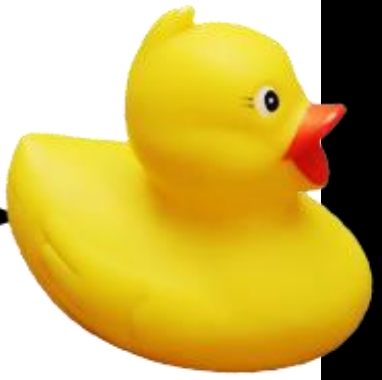


- Computation takes long (days, weeks, years)

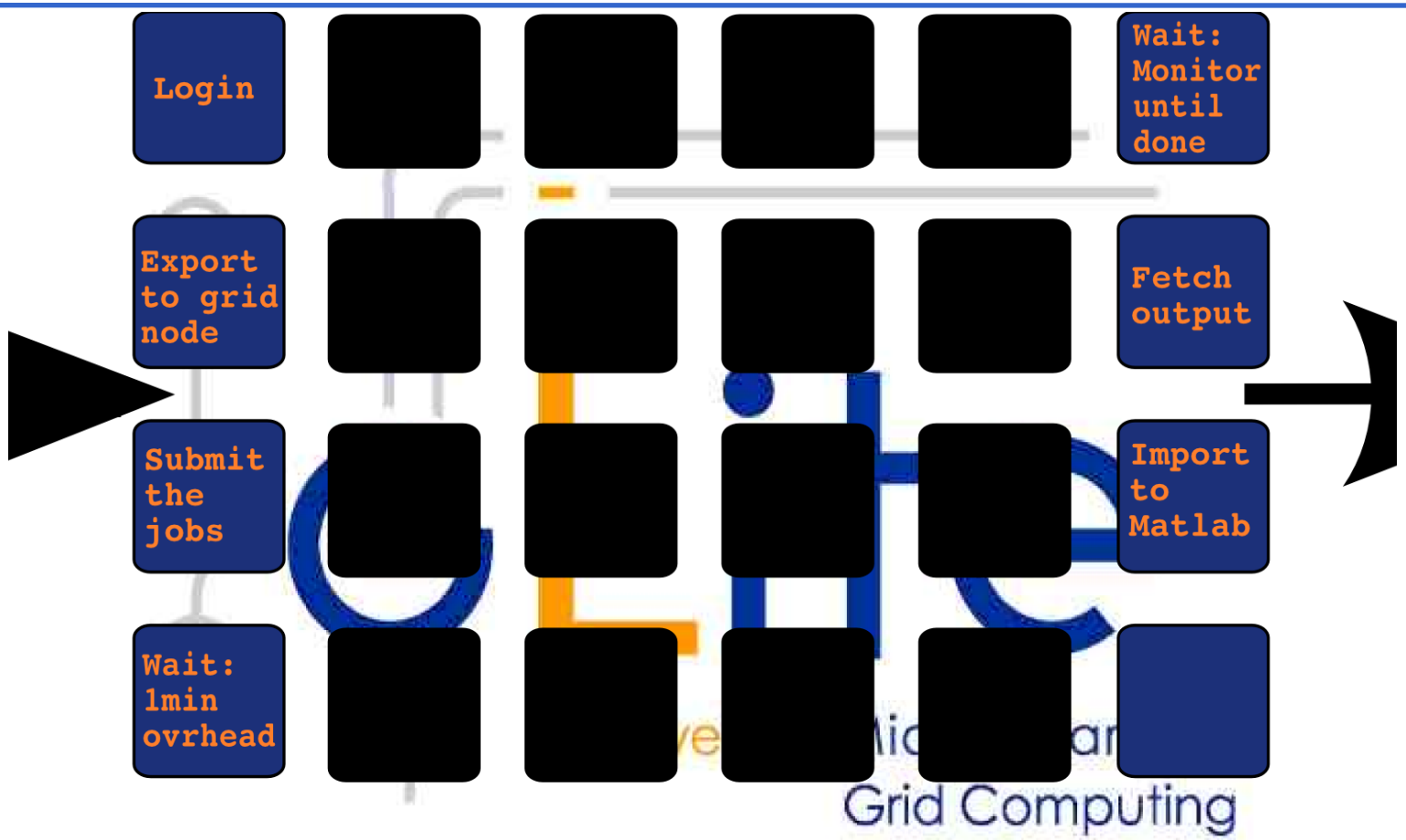
Using gLite

- Initial approach to parallel execution:
  - Partitioning of data
  - Many parallel jobs





Using gLite in practise



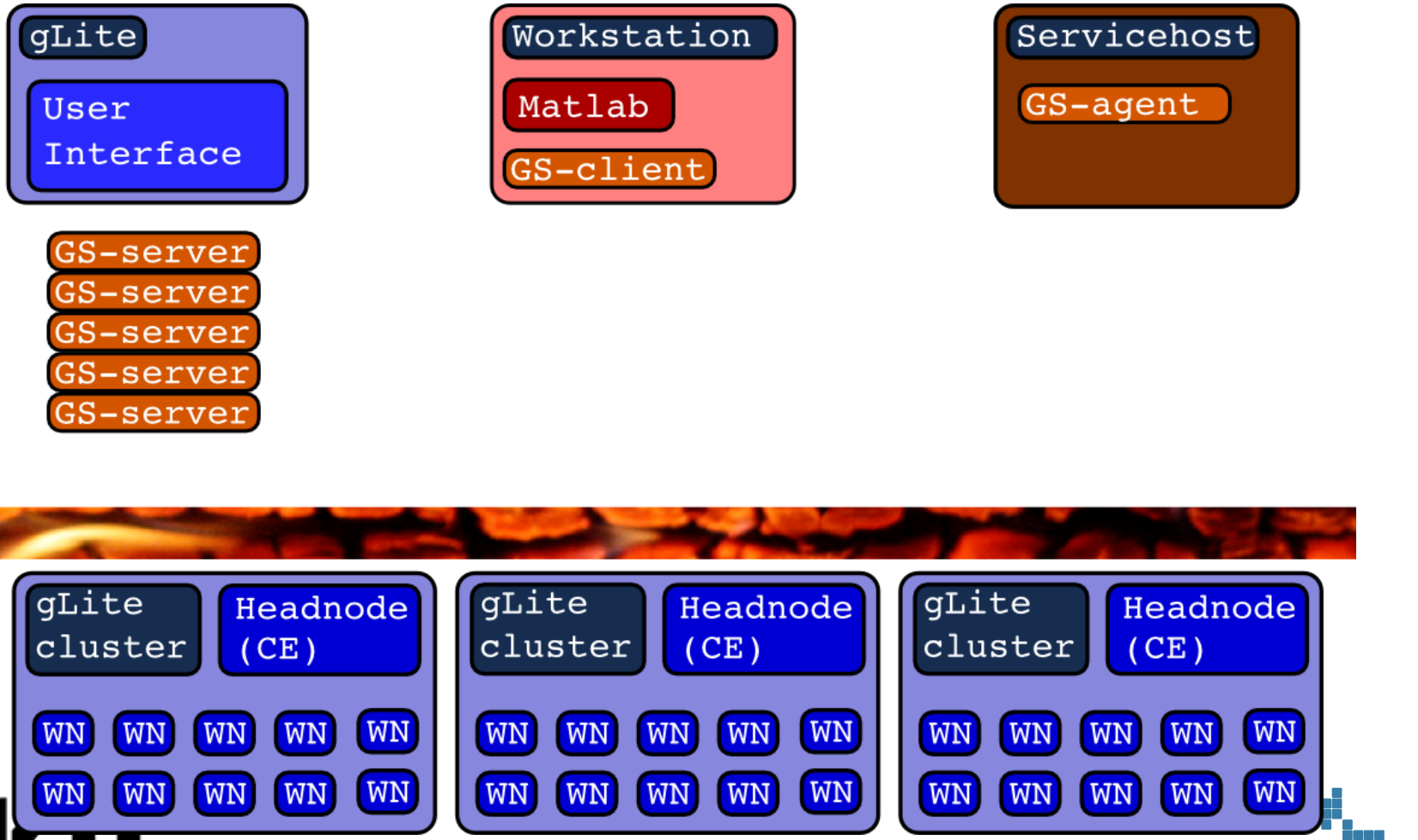
- A lot ot work is lett to the user

What's missing?

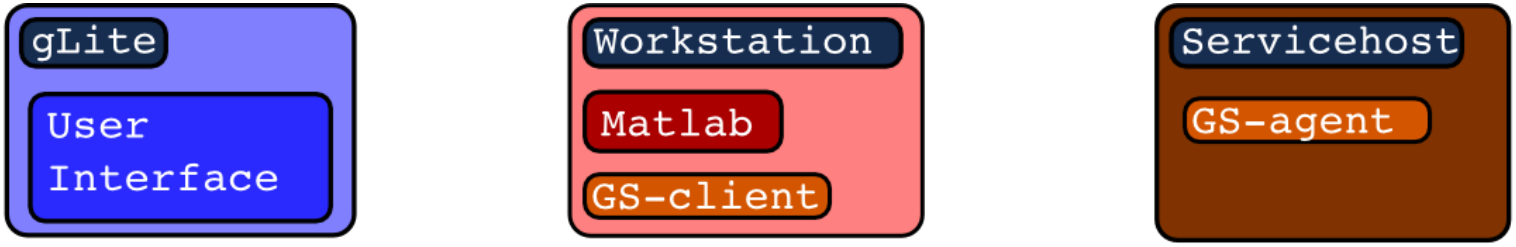
- **Goal:**
  - **Seamless**
  - **Interactive**
  - **Grid access**
  - **From matlab**

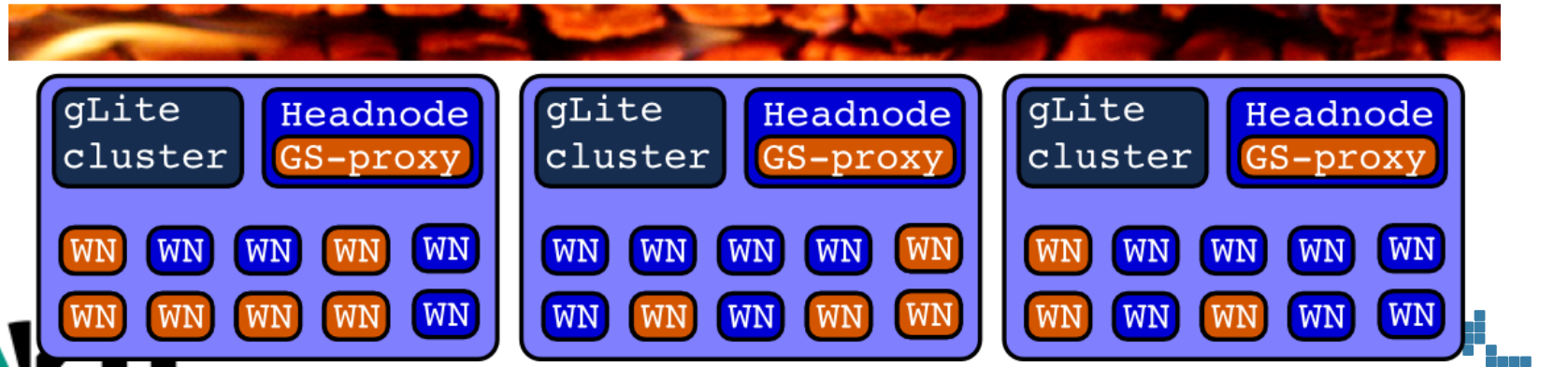
- Seamless
  - Don't compile standalone application
- Interactive
  - No overhead (< 10 s)
  - No manual data movement
- From Matlab
  - Run Matlab-functions remotely

Our cable: GridSolve

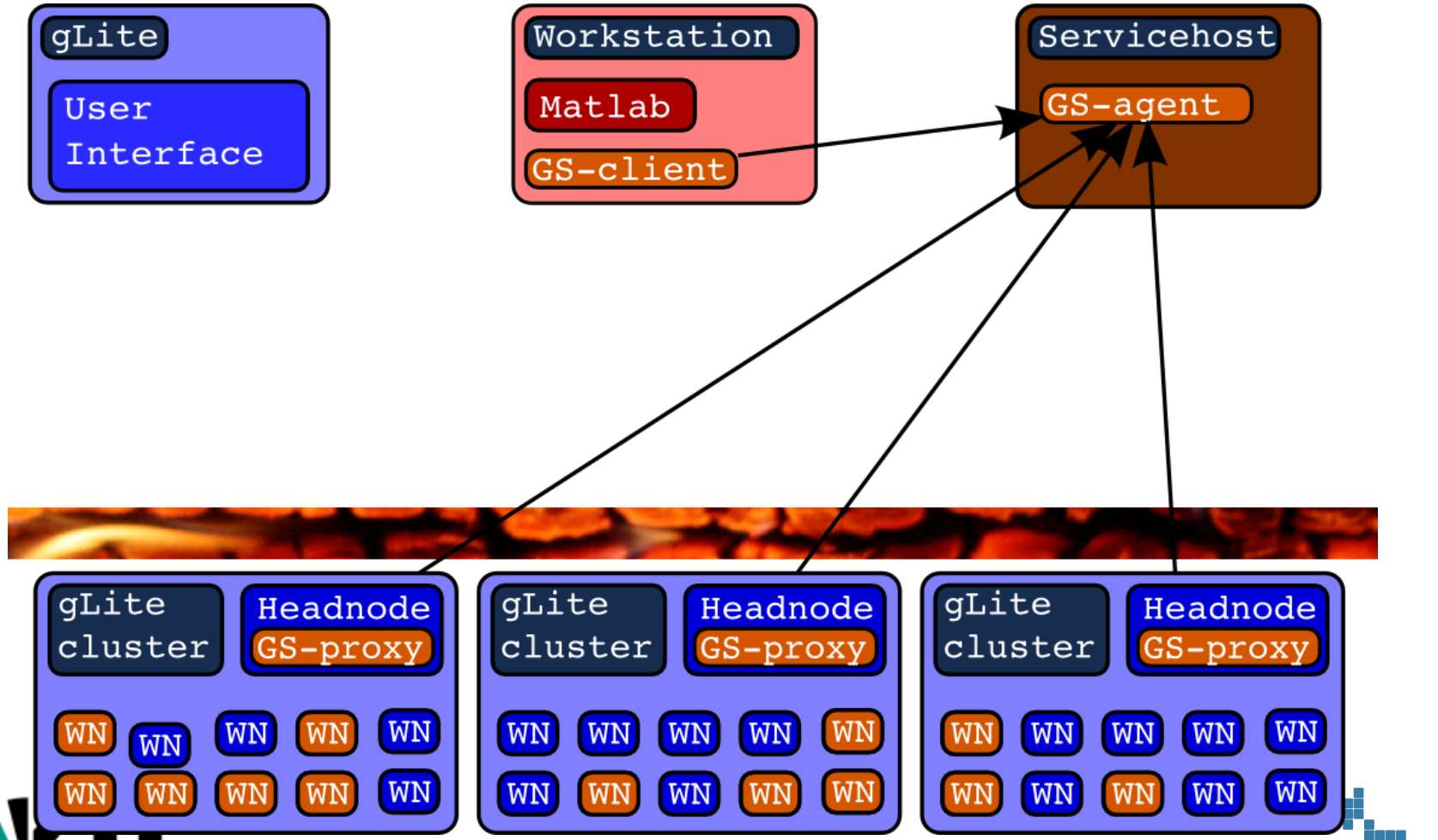


GridSolve ready for action





GridSolve in action



GridSolve in action