How to Port Scientific Application on the GRID ?

Stefano Cozzini

CNR/INFM Democritos and ICTP

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Outline

- a look at EU-IndiaGrid/gLite infrastructure
- Users&Application&Problems
- Our methodology to work on the GRID
- Parallel computing on the GRID
  - some concepts and our tricks
- Conclusions

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

EU-IndiaGRID infrastructure for applications (1)

- What it can offer:
  - CPU time on best effort basis if not otherwise specified/decided with Specific SLAs agreement
  - A job submission mechanism (batch system)
  - Storage and Data Management tools for specialized data
  - A specialized VO you can take part of...

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

EU-IndiaGRID infrastructure for applications (2)

- CPUs are loosely coupled (even if dual/quad core machines starts appearing..)
- Limited MPI support
  - No inter-site MPI computation (not really a problem)
  - No real way to distinguish between clusters and farms for MPI parallel jobs..

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

People on the GRID

- Users:
  - I want to use the GRID
- Providers:
  - Please use my GRID..

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Users and their requests

- Users:
  - Scientists/Researcher NOT computer scientists
  - IT skills not always present
  - IT Laziness/scepticism quite spread (even among young people)
  - The concepts of GRID computing is seen/perceived in many different ways leading to a variety of requests

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Which kind of applications?

- home made codes
  - easy: users are supposed to know everything about it
- Some well know packages used with slight modification/adaptations
  - less easy: users do not know too much
- "black box" application (a.k.a. legacy application)
  - difficult: nobody knows about it

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

which kind of "computational experiment ?"

- just a single application to run
  - parallel
    - embarrassingly /tightly coupled
  - serial
- a bunch of applications linked together (workflow)

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Our method to port applications..

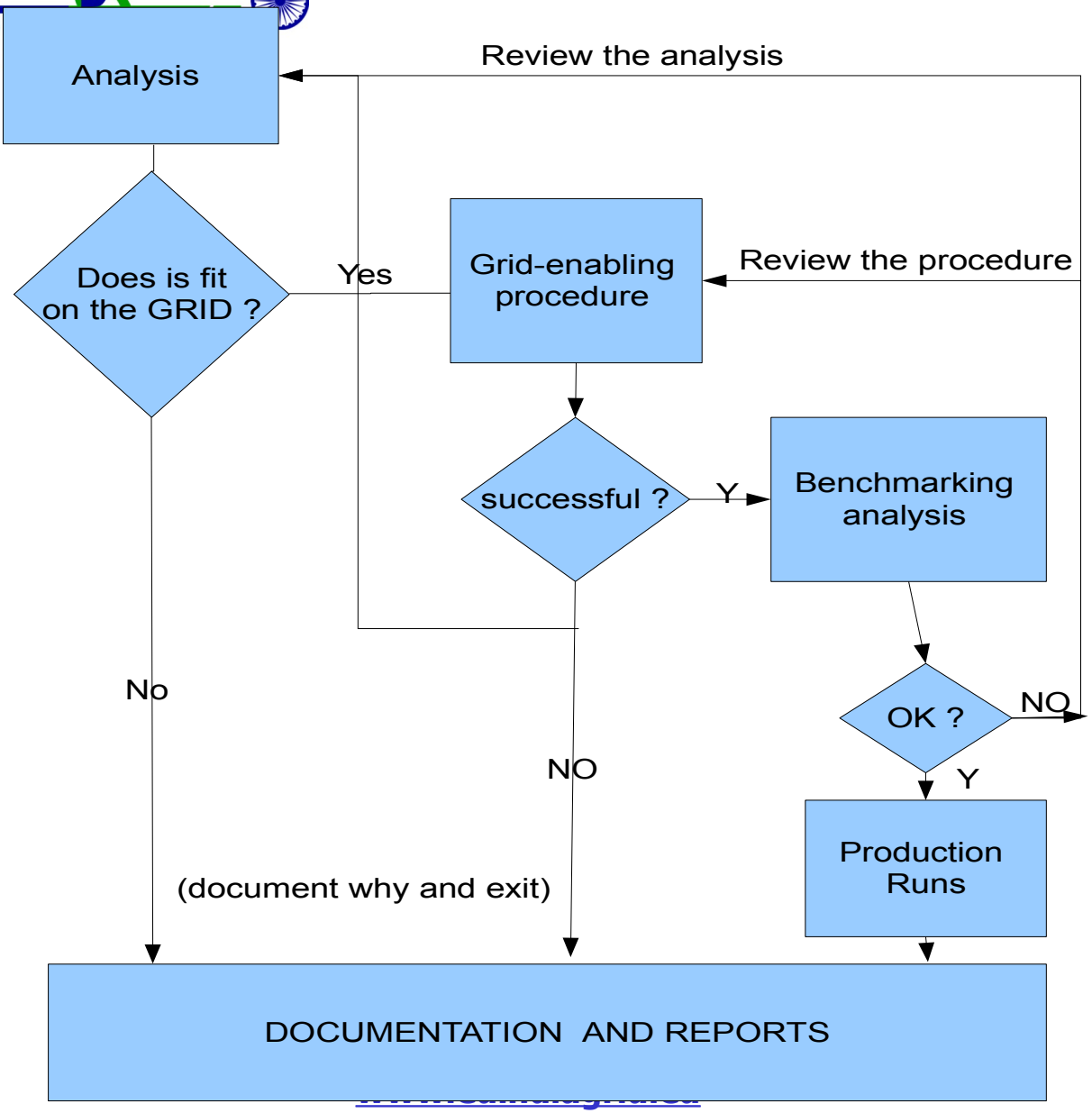
- Identify them through user's discussions and meeting:
  - Understand the computational requirements of the group
  - Understand the level of IT skills in the group
  - Understand their concept of GRID and applications
  - Understand the application requirements
  - Propose a solution (if exists)

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Steps to follow..

- Step 0: awareness of grid computing opportunities /analysis of the computational requirement
- Step 1: technical deployment on the infrastructure
- Step 2: benchmarking procedures and assessment of the efficiency
- Step 3: production runs and final evaluation
- Step 4: dissemination of the results among peers.

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme


Analysis
Review the analysis
Does is fit on the GRID ?
Yes
Grid-enabling procedure
Review the procedure
successful ?
Y
Benchmarking analysis
No
NO
OK ?
NO
Y
Production Runs
(document why and exit)
DOCUMENTATION AND REPORTS

Options for step 1: technical deployment

- Simple serial porting of an application
  - setup of a few simple tools (scripts etc.) to run the computational experiment (generally a parametric study)
- Porting of parallel applications with MPI:
  - this again is quite simple in principle but present MPI limitation on the infrastructure makes this strategy not very efficient.

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Options for step 1: technical deployment (2)

- Porting of complex codes/packages
  - requires important changes on the original way to run the application and the experiment associated.
- Client/Server approach
  - to run embarrassingly or loosely coupled applications/experiments

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

What do we need to port them on GRID ?

- TOOLS
  - Scripting languages for CLI
  - Portals
- HUMAN FACTOR
  - Man power (from both sides: providers+ users)
  - Goodwill (from users)

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Command Line interface..

- UI commands are generally complex, they are usually reiterated very frequently becoming:
  - boring/error prone/inefficient
- However this approach is really flexible..
- Hackers will love it
- Windows guys will hate it (grid is not “just one click away”.)

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

CLI flexibility...

- You can combine different commands into simple scripts
- Scripts can be easily re-used/changed... or combined together..
- This is not grid computing this is just plain linux/unix hacking..
- To do that please use your preferred scripting language.

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Which kind of applications/computational experiments fit/unfit?

- FIT:
  - Parameter sweep computational experiments
  - Embarrassingly parallel computations
  - CPU-bound applications
  - Long running computations
- UNFIT:
  - Tightly coupled parallel programs
  - I/O, Memory bounded applications

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

parallel computing on the GRID

- Pose the problem: parallel computing and the GRID
- Our partial solutions
  - accessing SMP resources on the GRID
  - use client/server approach

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

A few concepts in Parallel Computing

- Goal of parallel computing
  - Solve a problem on a parallel machine that is impractical on a serial one
  - How long does/will the problem take on P processors?
- Understanding parallel performance
  - Speedup: the effectiveness of parallelism
  - Limits to parallel performance
- Understanding serial performance
  - Parallelism in modern processors

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Speed-up

- The speedup of a parallel application is:

  Speedup(p) = Time(1)/Time(p)

- Where
  - Time(1) = execution time for a single processor and
  - Time(p) = execution time using p parallel processors
- If Speedup(p) = p we have perfect speedup (also called linear scaling)
- As defined, speedup compares an application with itself on one and on p processors, but it is more useful to compare
  - The execution time of the best serial application on 1 processor

    versus
  - The execution time of best parallel algorithm on p processors

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Amdahl's law (1)

- Although not a law, it is a very useful rule
- It states the following:

The Non-parallelizable fraction of a program limits performance and poses a upper limit on the speed-up

- $T_{solution} = T_{Non\ Parallel} + T_{Parallel}$
- NON parallel region limits speed up

$$S = 1 / (f_{np} + f_p/p) \text{ with } f_{np} + f_p = 1$$

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Amdahl's law(2)

- Which fraction of serial code is it allowed ?

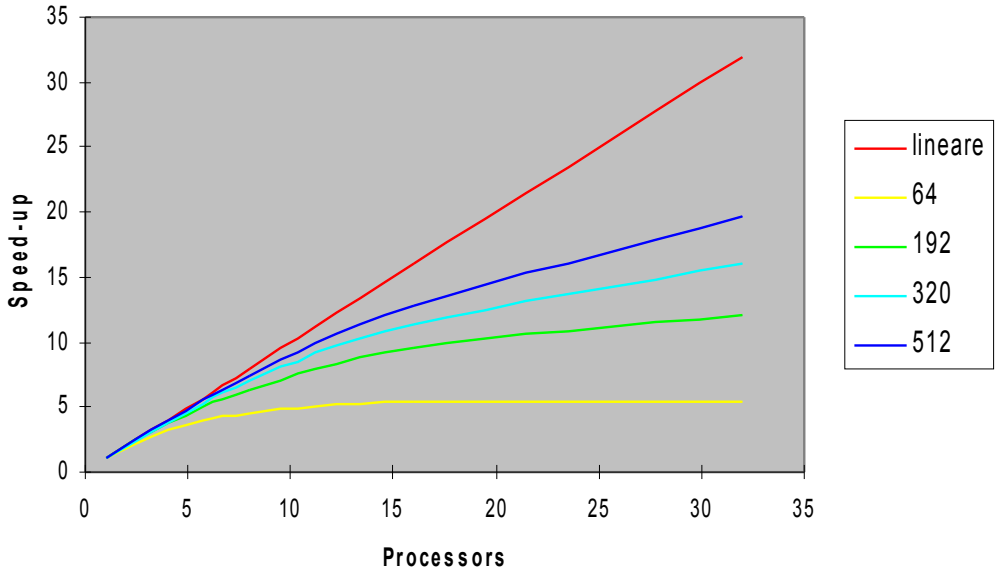
>	2	4	8	32	64	256	512	1024
5%	1.91	3.48	5.93	12.55	15.42	18.62	19.28	19.63
2%	1.94	3.67	6.61	16.58	22.15	29.60	31.35	32.31
1%	1.99	3.88	7.48	24.43	39.29	72.11	83.80	91.18

What about Scalability ???

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Problem scaling..

- Amdahl's Law is relevant only if serial fraction is indipendent of problem size, which is rarely true
- Fortunately "The proportion of the computations that are sequential (non parallel) normally decreases as the problem size increases " (a.k.a. Gustafon's Law)



www.euindiagrid.eu

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Different level of parallelism

- Within the core/cpu
  - Instruction level parallelism
- Within the node:
  - Threaded libraries/ openMP
- Within the cluster:
  - Message passing approach (i.e. MPI)
- Within the GRID
  - Message passing vs client/server approach..

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Which hardware for GRID/HPC nodes ?

- MULTIPROCESSORS machines
  - dual processor quite common/inexpensive
  - Quad processor available..

Multiple, externally visible processors on a single die where the processors have independent control-flow, separate internal state and no critical resource sharing

- MULTICORE !
  - dual core for AMD
  - quad core for Intel

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

What is offering glite middleware ?

- " The current release of the EGEE middleware (gLite 3.0.1 at time of writing) provides the MPICH job type to support MPI jobs. Users wishing to submit MPI(CH) jobs set the JobType variable to "MPICH" rather than "Normal" and set the variable "NodeNumber" to indicate how many nodes they require "

  [From MPI Working Group Report within EGEE]

- Check out news at:
  - http://www.grid.ie/mpi/wiki

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

What else ?

- GRID clusters are NOT created equal but they are considered equal from the MPI approach..
- No information about:
  - Presence of High Speed Network (infiniband/myrinet etc..)
  - Different MPI implementations...
  - etc../etc..

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

MPI-start approach

- to enable running MPI applications on different sit es with different configurations
- A user can use custom scripts to specify the MPI version/ scheduler/file distribution system etc. to b e used for a particular site

AVAILABLE JUST FROM A FEW MONTHS:
LOTS OF CONFIGURATION PROBLEMS
on EUINDIAGRID RESOURCES

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

My parallel code does not fit EGEE Grid model..

- YES if you want:
  - to run a very large tightly coupled parallel computation
  - to work alone (or in a small group) in a single institution.
  - To run/use just a single (parallel) application in your computer experiment

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

My parallel code does not fit EGEE Grid model..

- Maybe not if you want:
  - To run many different (and not so parallel) computations
  - To work in collaboration with many people/groups geographically distributed.
  - your MPI-enabled code is not so tightly coupled

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

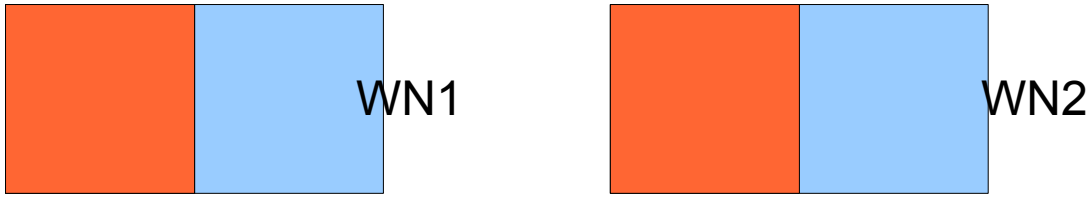
Our non MPI approach

- Goal:
  - Exploit the parallelism at node level
- Motivation
  - SMP node with 2/4 and even 8 core are available !
- Idea:
  - Select (reserve) a node for SMP parallel computation
  -

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

How to identify and reserve the SMP nodes ?

- Use the MPICH flag in the JDL
  - it could works if the LRSM is configured appropriately but
  - Your CPUs can be across two nodes...



EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Our approach..

- Develop a small tool that just submits a bunch of 1 cpu jobs and check which ones are landing on the same SMP node and then use them all together..
- ./reserve_job -i
  - How many CPU do you want on the same node ?
  - How many jobs do I need to submit ?
  - How should I wait before get rid of all ?
  - Give me please a script to load on the node..

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

An important point

- reserve_job just reserve the node and coordinates the jobs landed on the top of it
- It is up to the user (through a script) to use the node efficiently and in parallel.

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Ideas how to use it efficiently

- Compile statically an MPI program using MPI shared memory implementation and run it
- Compile a serial code against threaded libraries and get parallelism for free

- hands on session later on both examples

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Using reserve_smp_nodes..

- Run Q/Espresso simulations on small systems...
  - statically compiled using smp enabled MPI implementation
- We are now at step 2: benchmarking...
- preliminary results:
  - SMP performance comparable with HPC resources
  - overhead from Grid Submission sometime too large..

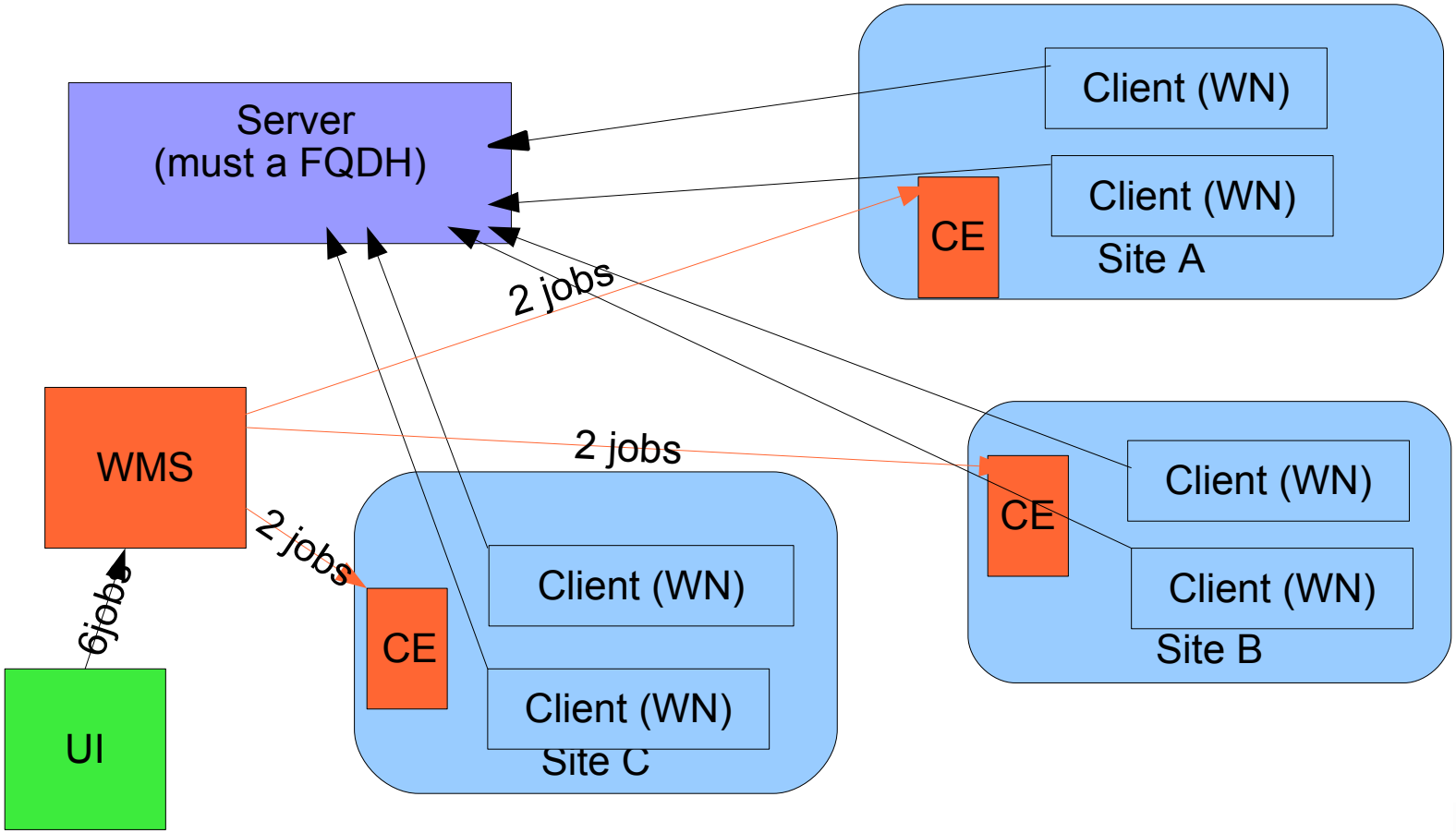
EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Client/Server architecture

- Fits perfectly for loosely coupled application:
- Idea
  - Submit N independent jobs and coordinate them through a specific client-server architecture
  - Each node performs a task
  - Exchanges among clients are coordinated by the client/server mechanism..
- Requirement:
  - outbound connectivity from Wns to Server

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Client/Server mechanisms



EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Client/Server examples

- BEMUSE:
  - Biased Exchange Metadynamics Submission Environment

  (see talk Fabio Pietrucci/Riccardo di Meo 's talk tomorrow)
- Quantum Simulation:
  - Phonon calculations with Quantum/Espresso
  - see Paolo Giannozzi's talk tomorrow

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Conclusions

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Some tips to users (1)

- State clearly the computational requirements:
  - CPU intensive
  - I/O requirements
  - memory requirements
  - scientific libraries needed (if any)

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Some tips to users (2)

- try to estimate the actual need of resource for your computational problem
  - how many times do I need to run my program ?
  - how much does it take to run ?
  - is there any room from improvement in term of performances ?

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme

Lesson learned

- Role of human interaction is fundamental and sometimes underestimated
- GRID users speak different language from GRID providers
- GRID providers need to adopt their language NOT viceversa
- Technical tools are important BUT not sufficient to win the user inertia..

EU-IndiaGrid (RI-031834) is funded by the European Commission under the Research Infrastructure Programme