

The OurGrid Project

Walfredo Cirne

walfredo@dsc.ufcg.edu.br

Universidade Federal de Campina Grande

- Computers are changing scientific research
 - Enabling collaboration
 - As investigation tools (simulations, data mining, etc...)
- As a result, many research labs around the world are now computation hungry
- Buying more computers is just part of answer
- Better using existing resources is the other

Solution 1: Globus

- Grids promise “plug on the wall and solve your problem”
- Globus is the closest realization of such vision
 - Deployed for dozens of sites
- But it requires highly-specialized skills and complex off-line negotiation
- Good solution for large labs that work in collaboration with other large labs
 - CERN’s LCG is a good example of state-of-art

Solution 2: Voluntary Computing

- SETI@home, FightAIDS@home, Folding@home, YouNameIt@home have been a great success, harnessing the power of millions of computers
- However, to use this solution, you must
 - have a very high visibility project
 - be in a well-known institution
 - invest a good deal of effort in “advertising”

And what about the thousands of small and middle research labs throughout the world which also need lots of compute power?

Solution 3: OurGrid

- OurGrid is a peer-to-peer grid
- Each lab correspond to a peer in the system
- OurGrid is easy to install and automatically configures itself
- Labs can freely join the system without any human intervention
- To keep it doable, we focus on Bag-of-Tasks application

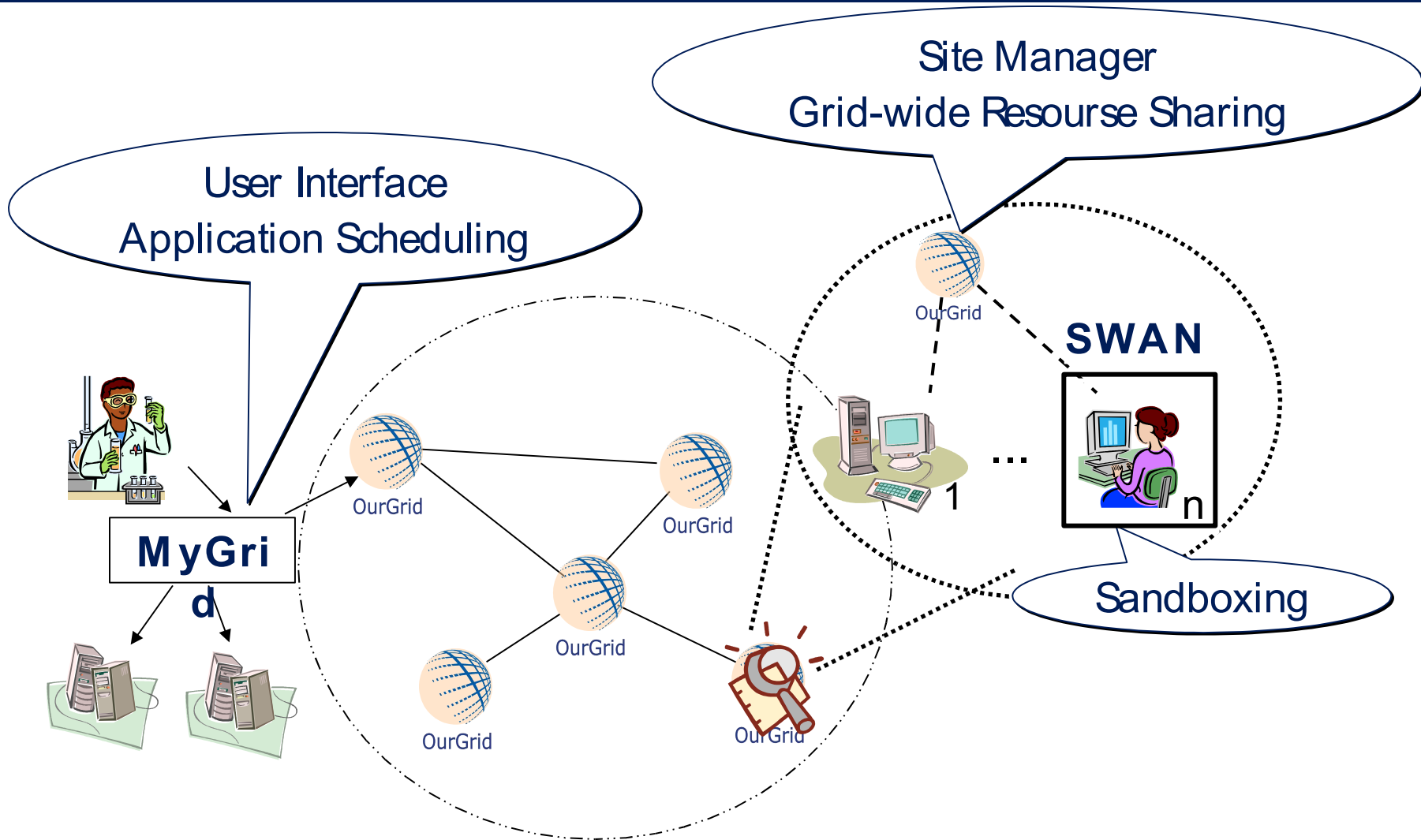
Bag-of-Tasks Applications

- Data mining
- Massive search (as search for crypto keys)
- Parameter sweeps
- Monte Carlo simulations
- Fractals (such as Mandelbrot)
- Image manipulation (such as tomography)
- And many others...

OurGrid Components

- **OurGrid**: A peer-to-peer network that performs fair resource sharing among unknown peers
- **MyGrid**: A broker that schedules BoT applications
- **SWAN**: A sandbox that makes it safe running a computation for an unknown peer

OurGrid Architecture



An Example: Factoring with MyGrid

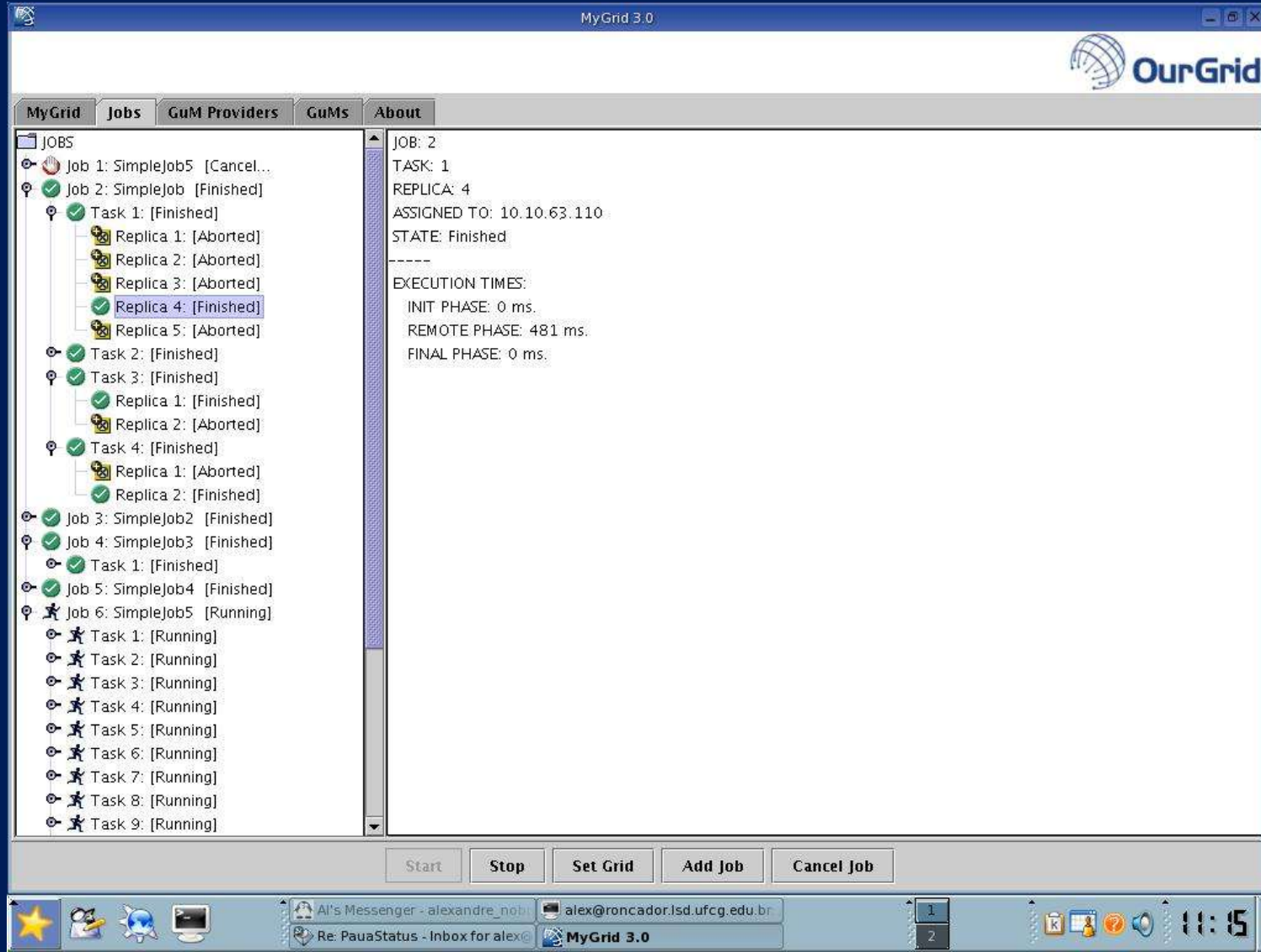
```
task:
  init: put ./Fat.class $PLAYPEN
  remote: java Fat 3 18655 34789789799 output-$TASK
  final: get $PLAYPEN/output-$TASK results

task:
  init: put ./Fat.class $PLAYPEN
  remote: java Fat 18656 37307 34789789799 output-$TASK
  final: get $PLAYPEN/output-$TASK results

task:
  init: put ./Fat.class $PLAYPEN
  remote: java Fat 37308 55968 34789789799 output-$TASK
  final: get $PLAYPEN/output-$TASK results
```

....

MyGrid GUI



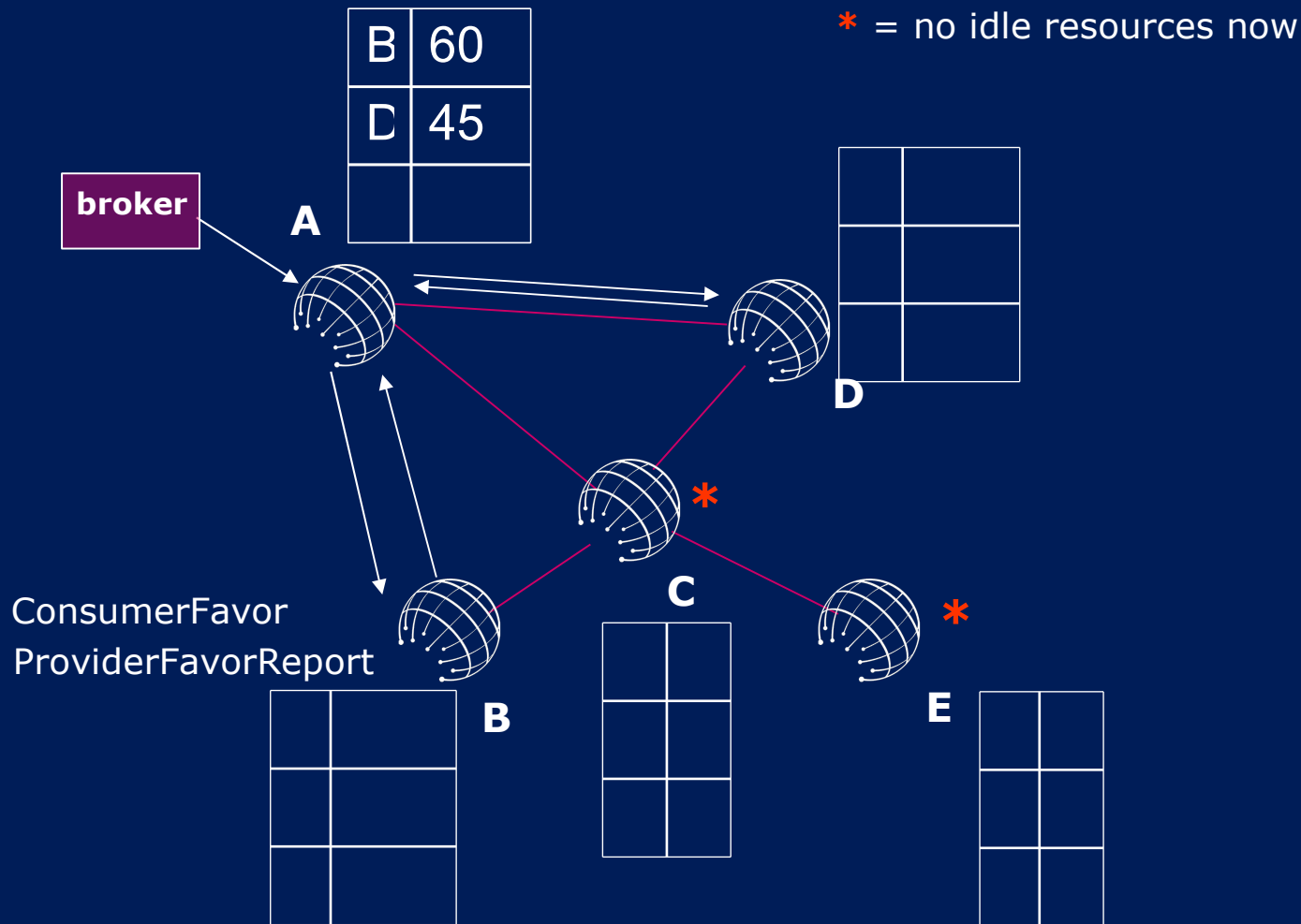
The screenshot displays the MyGrid 3.0 GUI. The window title is "MyGrid 3.0" and it features the OurGrid logo in the top right corner. The interface is divided into several sections:

- Navigation Tabs:** MyGrid, Jobs, GuM Providers, GuMs, About.
- Tree View (Left):** A hierarchical view of jobs and tasks. Job 2 is expanded, showing Task 1 (Finished) with five replicas (Replica 1-5). Replicas 1, 2, 3, and 5 are marked as [Aborted], while Replica 4 is [Finished]. Other jobs (1, 3, 4, 5, 6) are also shown with their respective task and replica statuses.
- Details View (Right):** Provides information for the selected Job 2:
 - JOB: 2
 - TASK: 1
 - REPLICA: 4
 - ASSIGNED TO: 10.10.63.110
 - STATE: Finished
 -
 - EXECUTION TIMES:
 - INIT PHASE: 0 ms.
 - REMOTE PHASE: 481 ms.
 - FINAL PHASE: 0 ms.
- Buttons:** Start, Stop, Set Grid, Add Job, Cancel Job.
- Taskbar:** Shows the Windows taskbar with various icons, including the MyGrid 3.0 application icon, and the system clock displaying 11:15.

Network of Favors

- OurGrid forms a peer-to-peer community in which **peers are free to join**
- It's important to encourage collaboration within OurGrid (i.e., resource sharing)
 - In file-sharing, most users **freeride**
- OurGrid uses the **Network of Favor**
 - All peers maintain a **local** balance for all known peers
 - Peers with greater balances have priority
 - The emergent behavior of the system is that by donating more, you get more resources
 - **No additional infrastructure is needed**

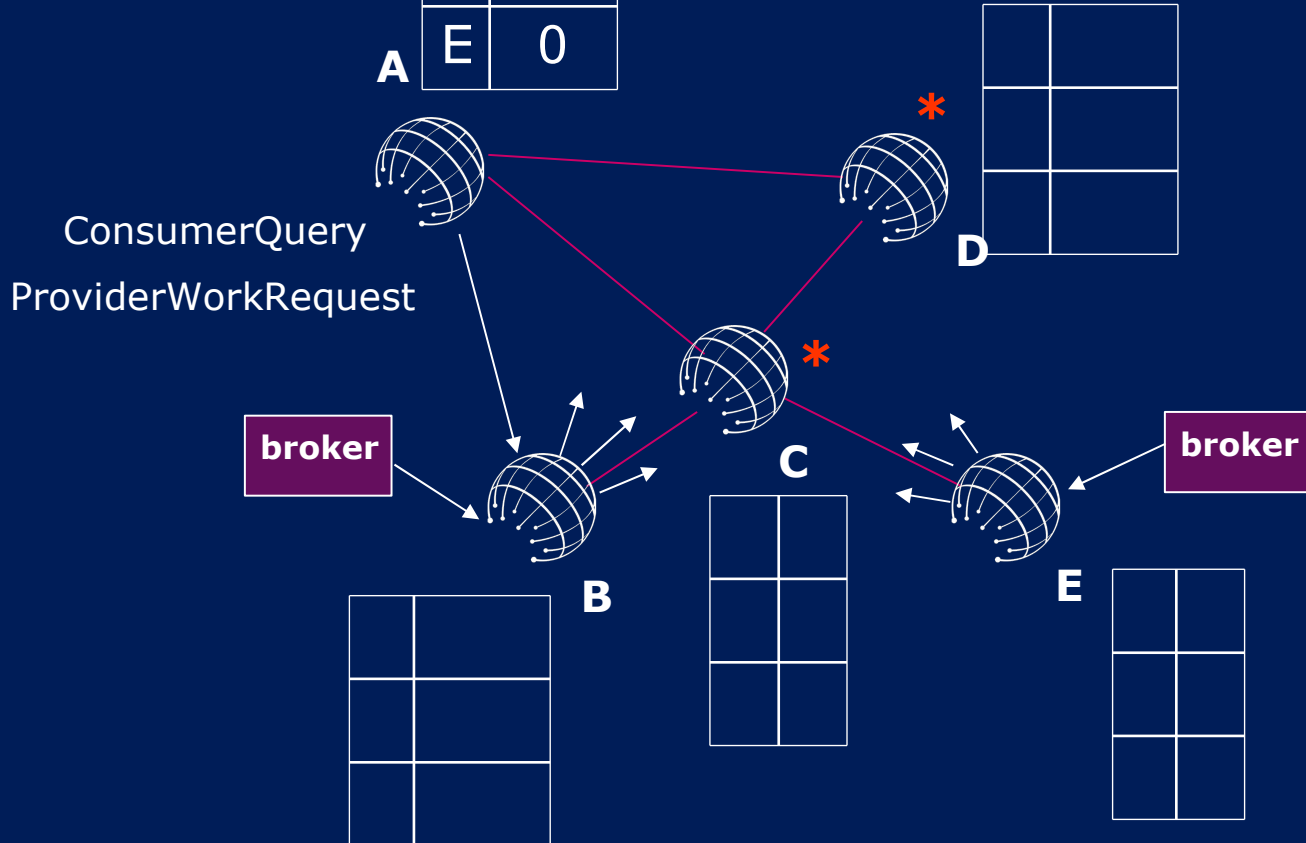
NoF at Work [1]



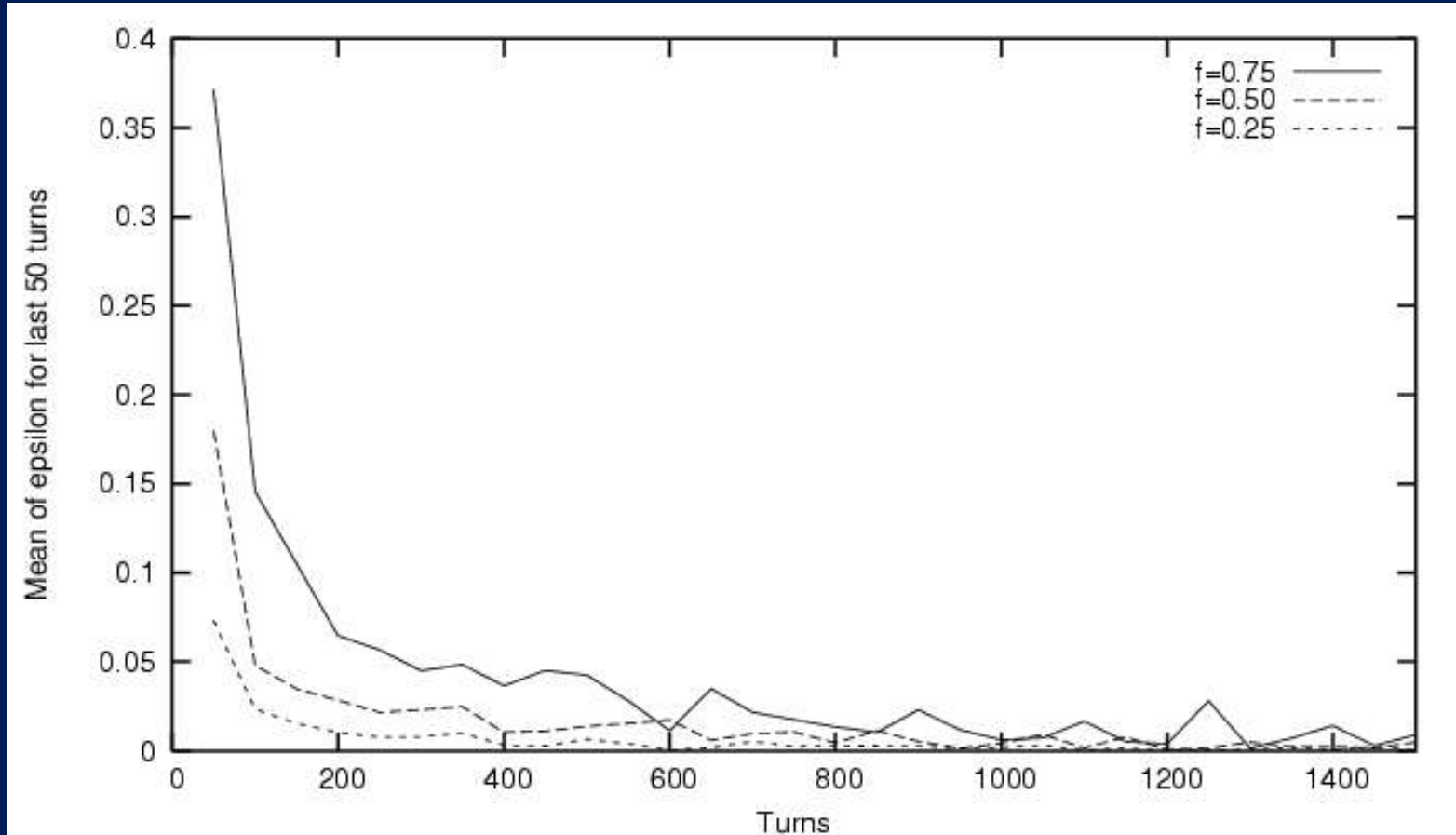
NoF at Work [2]

| | |
|---|----|
| B | 60 |
| D | 45 |
| E | 0 |

* = no idle resources now

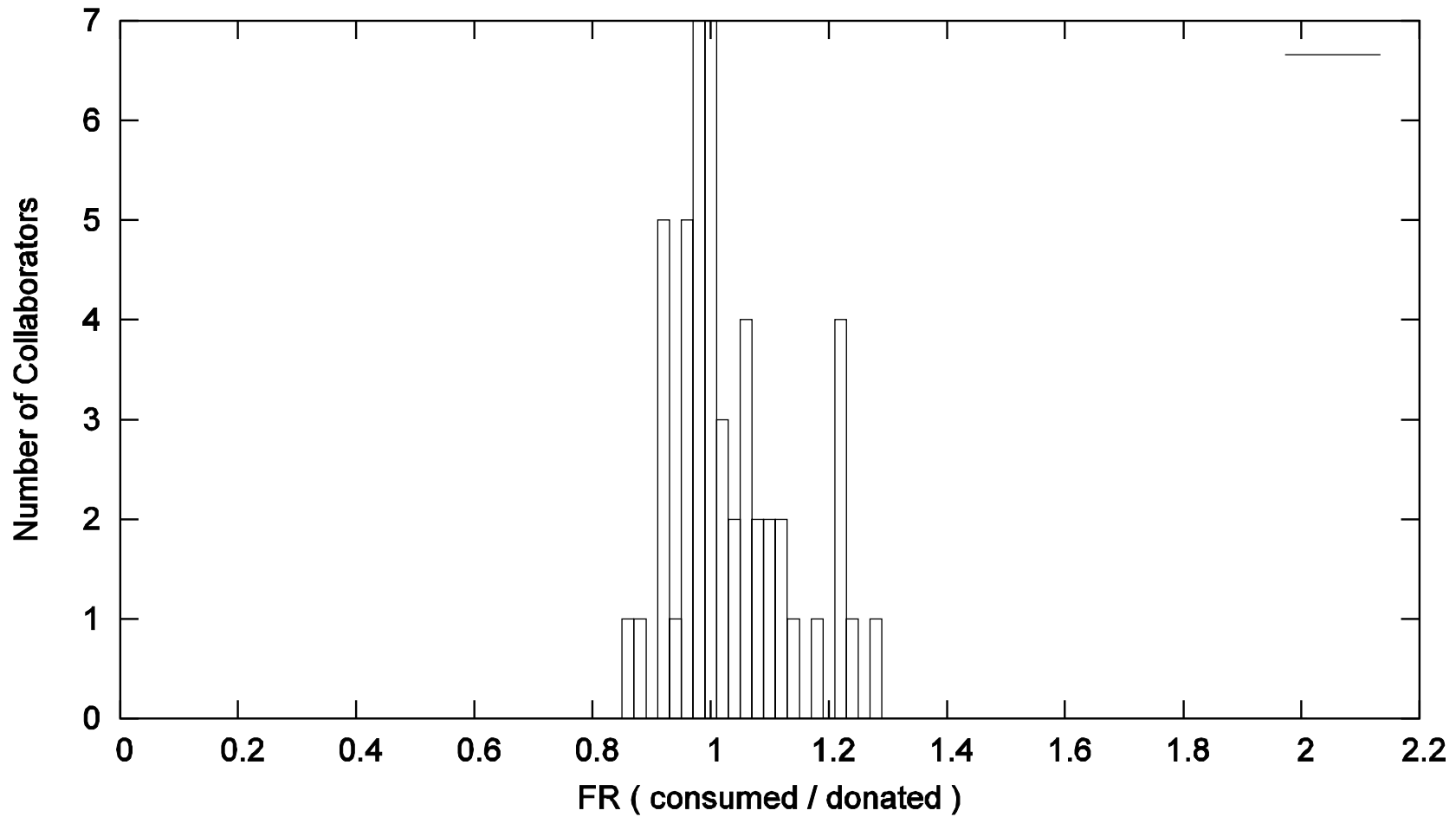


Free-rider Consumption



- Epsilon is the fraction of resources consumed by free-riders

Equity Among Collaborators



Scheduling with No Information

- Grid scheduling typically depends on information about the grid (e.g. machine speed and load) and the application (e.g. task size)
- However, getting good information is hard
- Can we schedule without information and deploy the system now?
- Work-queue with Replication
 - Tasks are sent to idle processors
 - When there are no more tasks, running tasks are replicated on idle processors
 - The first replica to finish is the official execution
 - Other replicas are cancelled

Work-queue with Replication

- 8000 experiments
- Experiments varied in
 - grid heterogeneity
 - application heterogeneity
 - application granularity
- Performance summary:

| | Sufferage | DFPLTF | Workqueue | WQR 2x | WQR 3x | WQR 4x |
|------------------|-----------|----------|-----------|----------|----------|----------|
| Average | 13530.26 | 12901.78 | 23066.99 | 12835.70 | 12123.66 | 11652.80 |
| Std. Dev. | 9556.55 | 9714.08 | 32655.85 | 10739.50 | 9434.70 | 8603.06 |

WQR Overhead

- Obviously, the drawback in WQR is cycles wasted by the cancelled replicas
- Wasted cycles:

| | WQR 2x | WQR 3x | WQR 4x |
|------------------|---------------|---------------|---------------|
| Average | 23.55% | 36.32% | 48.87% |
| Std. Dev. | 22.29% | 34.79% | 48.93% |

Data Aware Scheduling

- WQR achieves good performance for CPU-intensive BoT applications
- However, many important BoT applications are data-intensive
- These applications frequently reuse data
 - During the same execution
 - Between two successive executions
- **Storage Affinity** uses replication and just a bit of static information to achieve good scheduling for data intensive applications

Storage Affinity Results

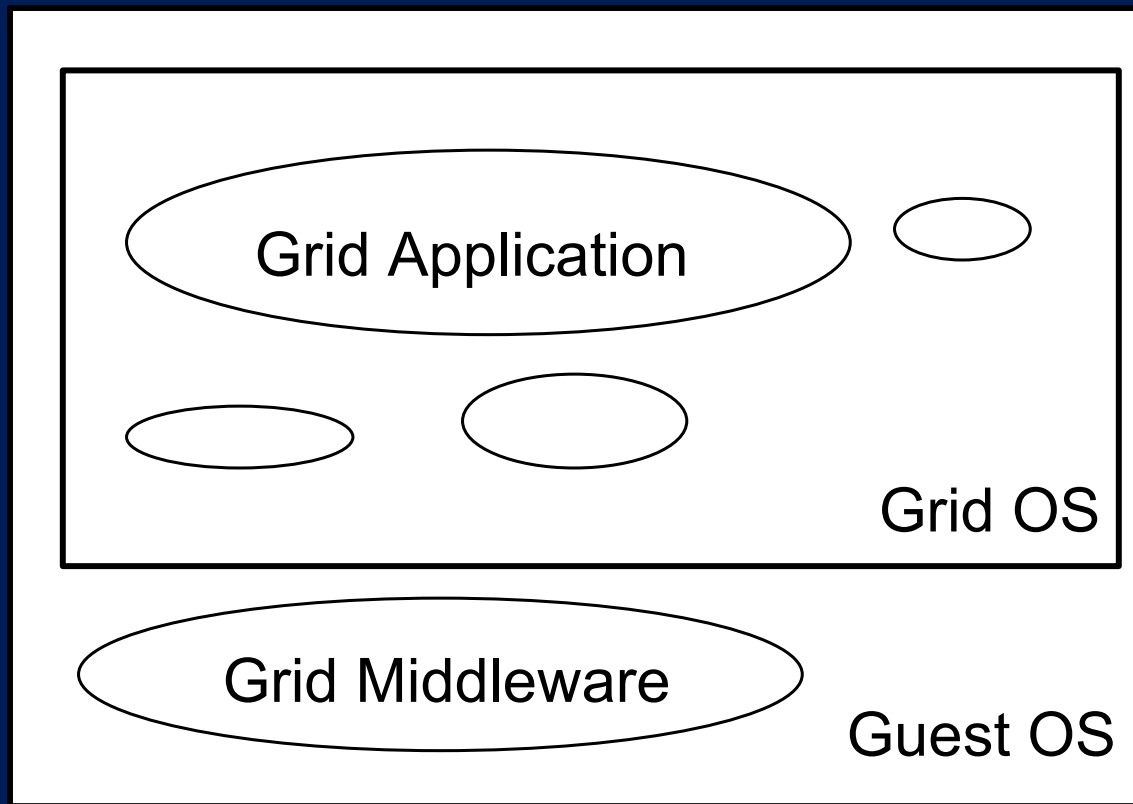
- 3000 experiments
- Experiments varied in
 - grid heterogeneity
 - application heterogeneity
 - application granularity
- Performance summary:

| | Storage Affinity | X-Suffrage | WQR |
|--------------------|------------------|------------|---------|
| Average (seconds) | 57.046 | 59.523 | 150.270 |
| Standard Deviation | 39.605 | 30.213 | 119.200 |

SWAN: OurGrid Security

- Bag-of-Tasks applications only communicate to receive input and return the output
 - This is done by OurGrid itself
- The remote task runs inside a Xen virtual machine, with no network access, and disk access only to a designated partition

SWAN Architecture



Making it Work for Real...



OurGrid Status

- OurGrid free-to-join community is in production since December 2004
- OurGrid is **open source** (GPL) and is available at **www.ourgrid.org**
 - We've had external contributions
- OurGrid latest version is 3.1
 - It contains the 10th version of MyGrid
 - The Network of Favors is available since version 3.0
 - SWAN has been made available with version 3.1
 - We've had around 180 downloads

http://status.ourgrid.org

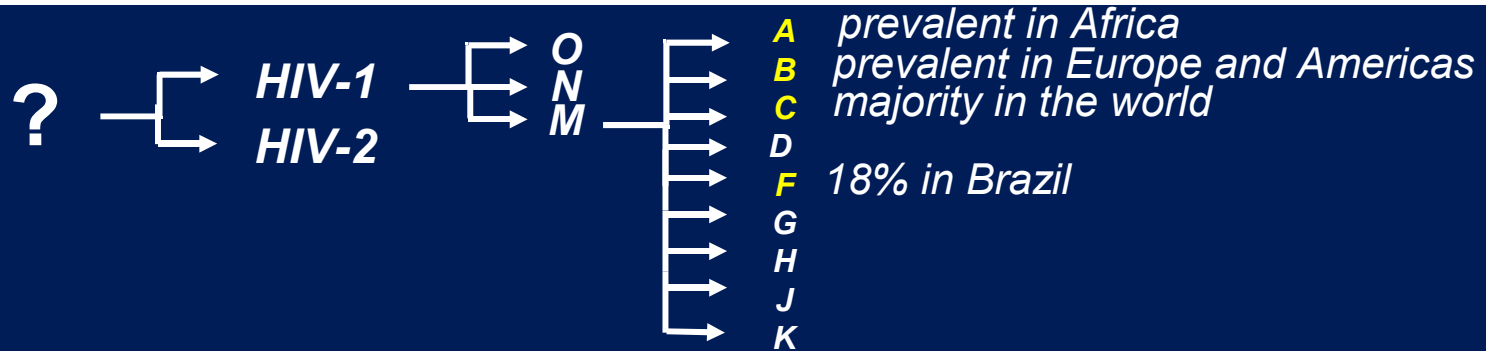


The screenshot shows a Netscape browser window titled "OurGrid Web Status - Netscape". The address bar contains "http://status.ourgrid.org/". The page content includes:

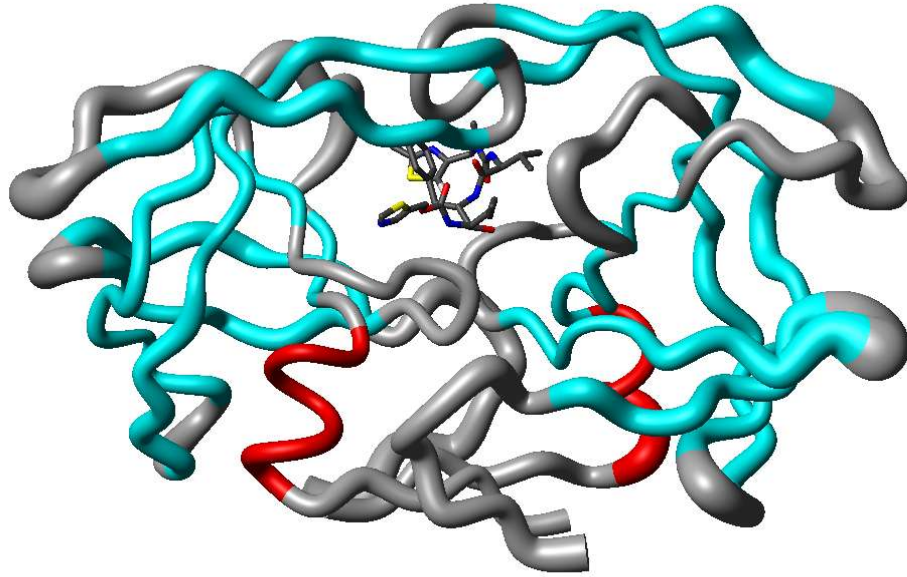
- OurGrid 3.0** logo in the left sidebar.
- OurGrid Web Status** main heading.
- Peers** section with a list of 9 links:
 1. [lncg.br](#)
 2. [dca.ufcg.edu.br](#)
 3. [public.lsd.ufcg.edu.br](#)
 4. [lcc.ufcg.edu.br](#)
 5. [copad.lsd.ufcg.edu.br](#)
 6. [cpad.pucrs.br](#)
 7. [mutuca](#)
 8. [lsd.ufcg.edu.br](#)
 9. [lmrs.ufcg.edu.br](#)
- lncg.br** section.
- Local GuMs:** ♦ 146.134.200.7
- Community GuMs:** This peer has no community GuMs!
- Network of Favors Accounting:** Peer [input] Debt [input]

The Windows taskbar at the bottom shows the "start" button, system tray with the time 21:55, and several open applications: "2005-04-SSA", "OurGrid Web Status - ...", "Microsoft PowerPoint ...", and "baleia.lsd.ufcg.edu.b...".

HIV research with OurGrid

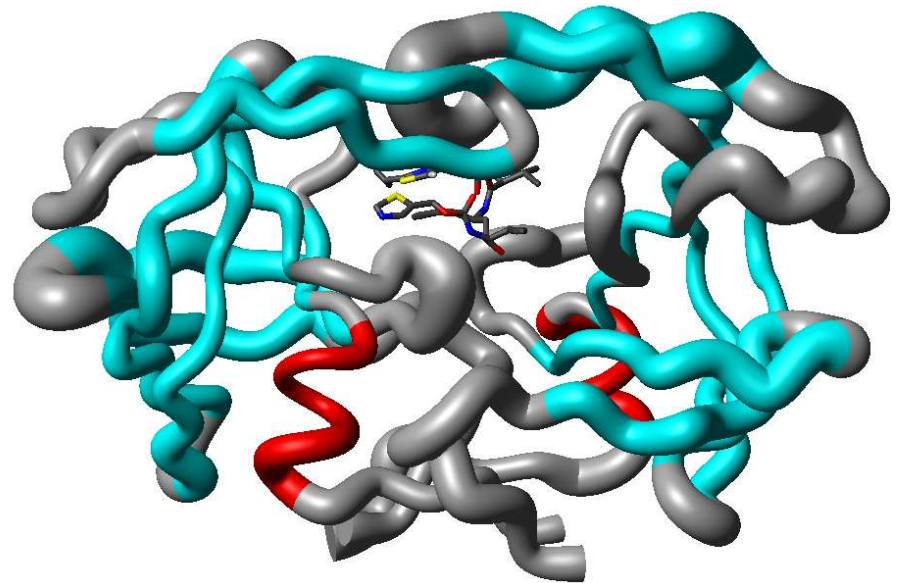


HIV protease + Ritonavir



Subtype B

Subtype F



Performance Results for the HIV Application



- 55 machines in 6 administrative domains in the US and Brazil
- Task = 3.3 MB input, 1 MB output, 4 to 33 minutes of dedicated execution
- Ran 60 tasks in 38 minutes
- **Speed-up is 29.2 for 55 machines**
 - Considering an 18.5-minute average machine

Conclusions

- We have an **free-to-join** grid solution for Bag-of-Tasks applications working **today**
- Real users provide invaluable feedback for systems research
- Delivering results to real users is really cool! :-)



OurGrid

Questions?

Thank you!

Merci!

Danke!

Grazie!

Gracias!

Obrigado!

More at www.ourgrid.org