Introduction to Grid'5000

Aladdin-G5K development team

INRIA

November 4, 2008

Introduction

How to study large scale parallel or distributed systems?

Different approaches:

- Formal proof
  - How to get a mathematical model of reality ?
- Simulation
  - How to make sure the simulator is realistic ?
- Emulation
  - How to emulate processors, network cards, switches and routers ?
- Experimentation
  - Where to find an full-scale experimentation testbed ?

Grid'5000 aims at providing an experimentation testbed to study large scale parallel or distributed systems

Grid'5000 is still an experimental platform as well: building such a platform is a full-fledged research topic

Producing reproducible and relevant scientific results

Simulation

- Find (develop) a good simulator and archive the version you used
- Archive the version of your application as well as input files

Emulation/Experimentation

- Prepare an environment for your experiment, trying to minimize outside interferences
- Archive the version of your application and input files
- Archive the whole environment you used:
  - Archive the software environment (OS, software, configuration information) used on the nodes
  - Archive the description of the resources used in the experience (CPU, memory, network, ...)

While obtaining relevant results when doing simulation highly depends on finding a realistic model, obtaining reproductible results when doing full-scale experiments is a real challenge.

Other testbeds for experiments

- Wide area testbed composed of about 900 nodes spread over 460 sites world-wide
- Allocation of slices: virtual machines
- Designed for experiments Internet-wide: new protocols for Internet, overlay networks (file-sharing, routing algorithm, multi-cast, ...)

- Emulab provides a platform where operating system and network can be tuned (emulated topology)
- Main installation in Univ. of Utah: about 350 nodes
- Not designed for experiments on large scale distributed systems

The other testbeds for experiments

- Netherland testbed composed of 272 nodes (about 800 CPU/cores)
- On the fly network backbone reconfiguration (optical routers with configurable wavelength)
- The software stack is not reconfigurable
- Strong links between DAS-3 and Grid'5000 communities

GLOBAL ENVIRONMENT FOR NETWORK INNOVATIONS

- An experimental infrastructure to run experimentation on the design of the Next Generation Internet
- Innovative technologies in the fields of network and virtualization
- Still in design...

Definitions

Some definitions

Parallel computing

The simultaneous execution of the same task (split up and specially adapted) on multiple processors in order to obtain results faster. The idea is based on the fact that the process of solving a problem usually can be divided into smaller tasks, which may be carried out simultaneously with some coordination.

Distributed computing

A programming paradigm focusing on designing distributed, open, scalable, transparent, fault tolerant systems. This paradigm is a natural result of the use of computers to form networks.

Cluster

Group of linked computers, working together closely so that in many aspects they form a single computer. The components of a cluster are commonly, but not always, connected to each other through fast LAN.

Some definitions

Grid

The sharing of computing resources (computers, clusters, parallel machines, ...) by a collection of people and institutions in a flexible and secured environment. The computing resources may be loosely coupled.

Site

Geographical place where a set of computing resources shares the same administration policy.

Large scale

Today, thinking large scale is thinking bigger than a big cluster on one site. The problems ALADDIN-G5K seeks to address are those of using hundreds of machines distributed on different sites.

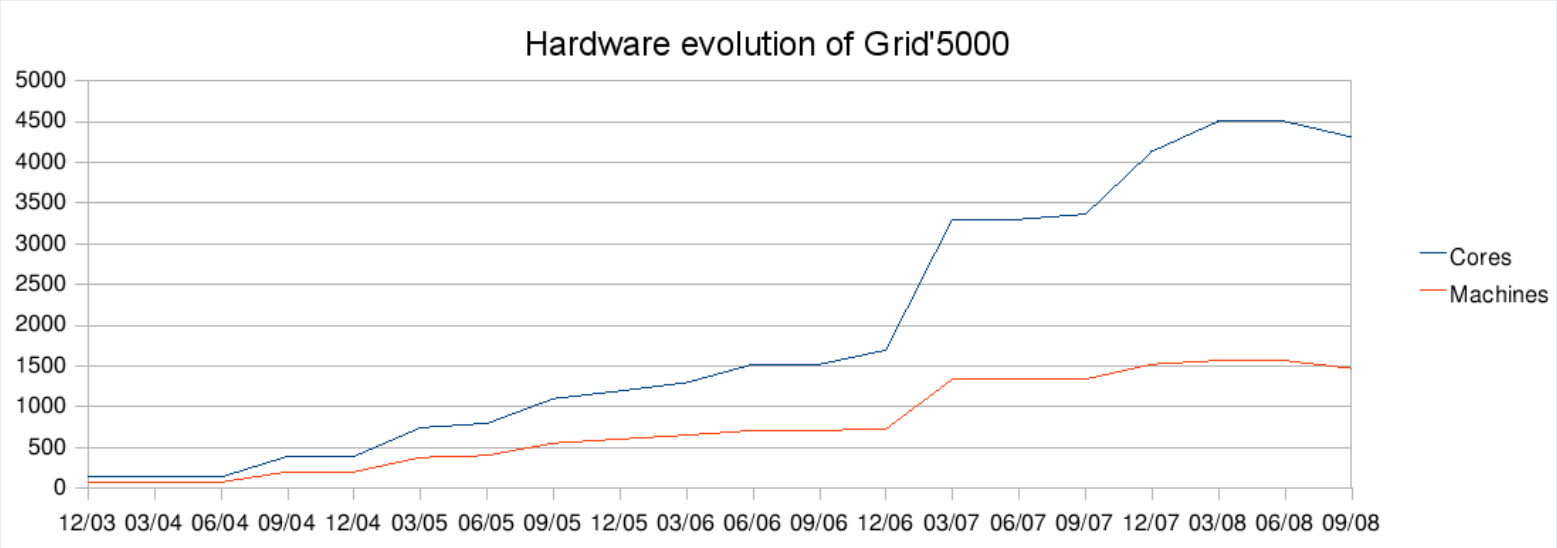
General presentation of Aladdin/Grid'5000

A bit of history

Structures

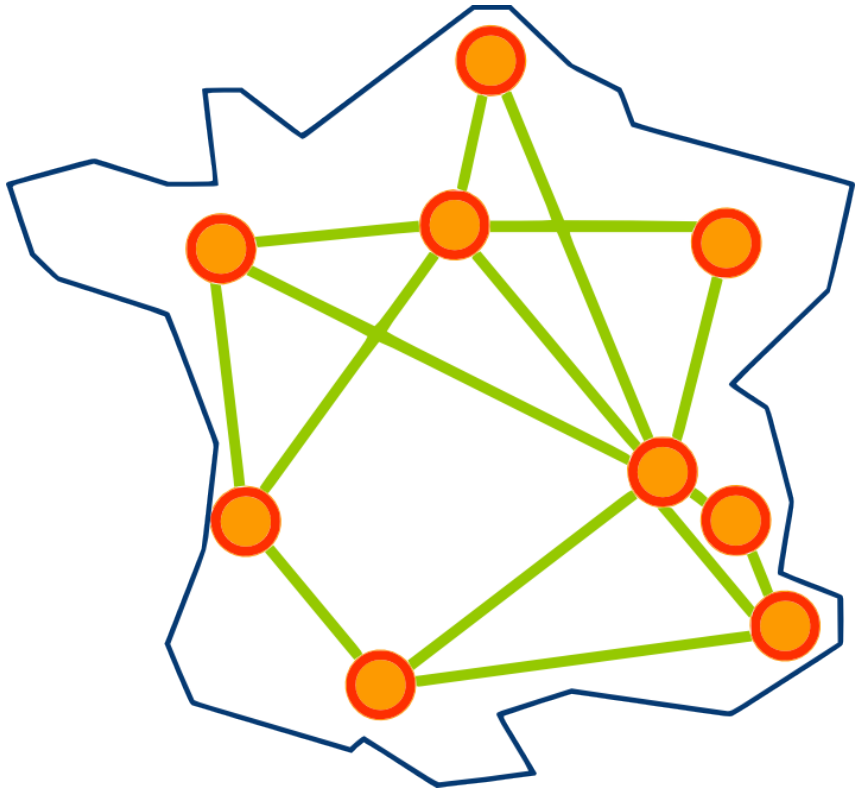
- Prototype: the Grid'5000 project of the french ACI GRID incentive is launched - 2003-2005
- First phase: the Grid'5000 platform is opened to users - 2005-2007
- Today: ALADDIN-G5K, INRIA's effort to further develop Grid'5000 - 2008-2011

Hardware



A nation-wide grid

9 sites



Sites

Bordeaux, Grenoble, Lille, Lyon, Nancy, Orsay, Rennes, Sophia, Toulouse

The hardware

CPU families

- AMD Opteron (78%), Intel Xeon EMT64 (22%)
- MonoCore (41%), DualCore (46%), QuadCore (13%)
- All machines are bi-processors
- In the past: Intel Itanium 2 and Xeon IA32, IBM PowerPC

High performance networks

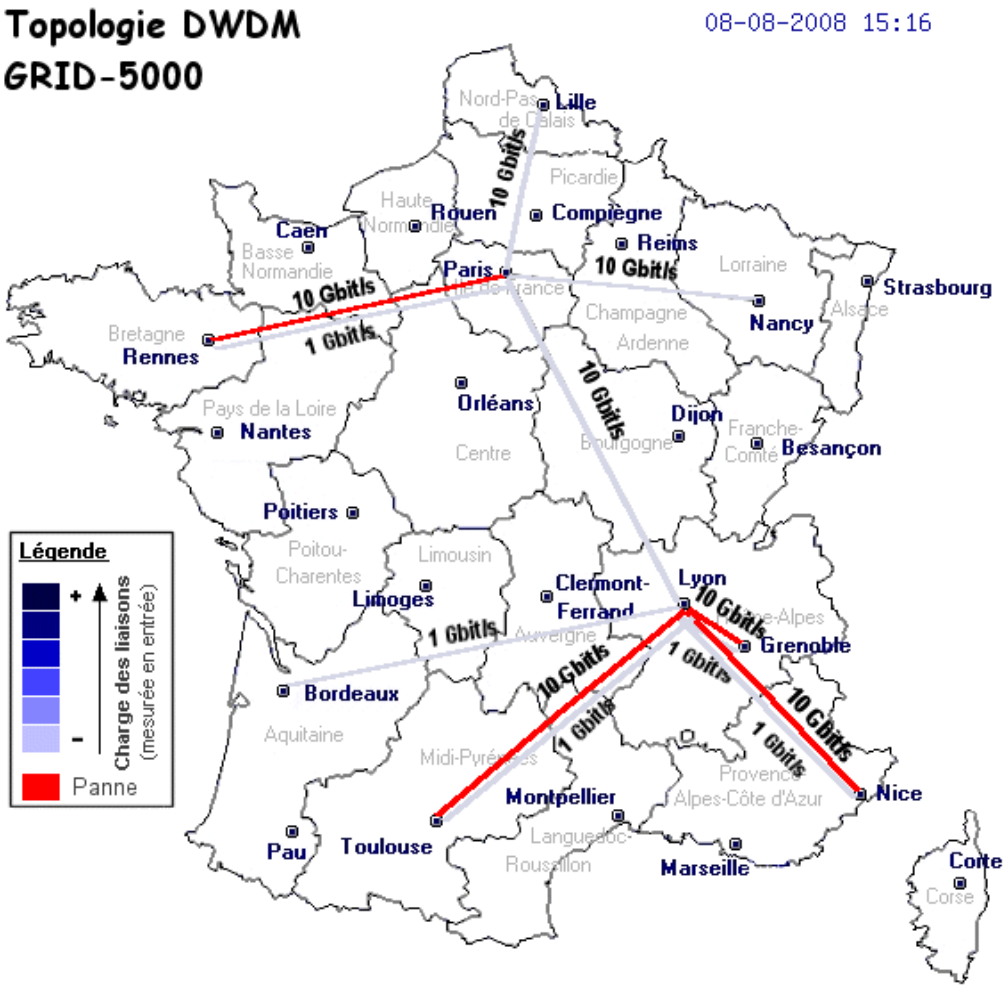
- Myrinet 2000 (222 cards)
- Myrinet 10G (423 cards)
- InfiniBand 10G (161 cards)

At a glance

- 4792 cores / 9 sites
- Gigabit Ethernet interconnect everywhere and 10Gb/s backbone
- More informations on:
  `https://www.grid5000.fr/mediawiki/index.php/Special:G5KHardware`

A 10Gb/s backbone network

Renater 4



People

Steering committee

Representative of the institutional partners involved in the project

Executive committee

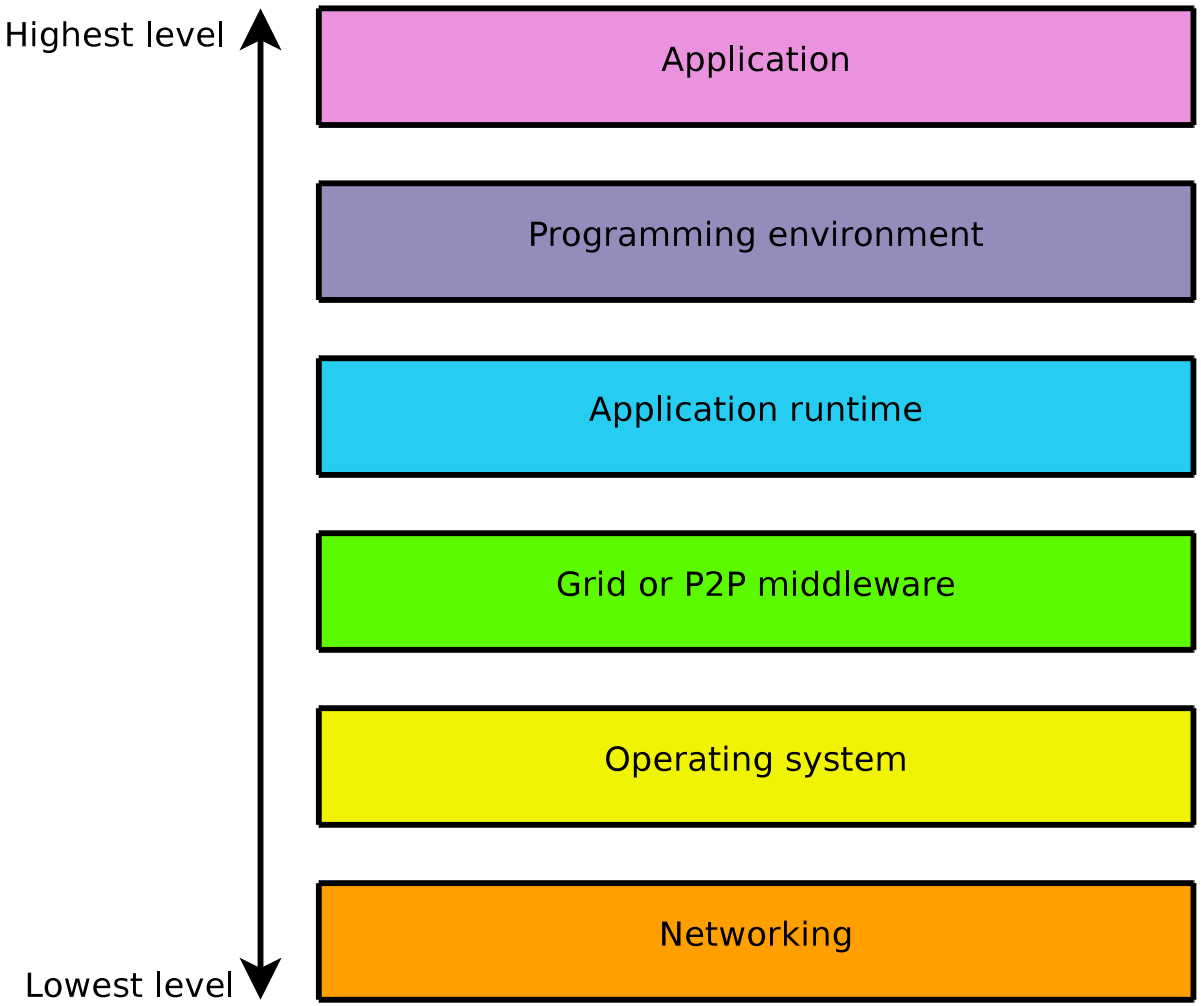
- Head director: Thierry Priol
- Scientific director: Franck Capello
- Technical director: David Margery

Technical committee

Set of engineers divided in two teams

- The support team: administration of the platform, development of administration tools, support to users
- The development team: design and development of the major tools used for the platform operation
- Contributors are welcome (developments, meetings, feedbacks, ...)

A large research applicability



Context of work

Shared instrument

Everyone should be civic-minded and should avoid the following behaviors:

- I think that my experience is the most important, so I can use all the resources for a very long time
- In order to let the user perform their experiments, the platform features a low security level. Thus I can abuse the system and disturb other users while they are performing experiments

User charter

Everyone must read and accept the user charter

`https://www.grid5000.fr/mediawiki/index.php/Grid5000:UserCharter`

Help can be found in the community

Grid'5000 is a community

Questions can be asked:

- to colleagues on your site or other Grid'5000 users you know
- to the local Grid'5000 staff if questions are about the usage of the infrastructure (BUT your local admin is not an MPI or a Globus expert)
- to the Grid'5000 users' mailing-lists

And please participate to the communitty effort by also answering questions when you can help !

Using Grid'5000 resources

Typical use case

1. Connect to the platform on a site
2. Reserve some resources
3. Configure the resources (optional)
4. Run your experiment
5. Grab the results
6. Free the resources

Provided services

Your account

With a Grid'5000 account, you'll get

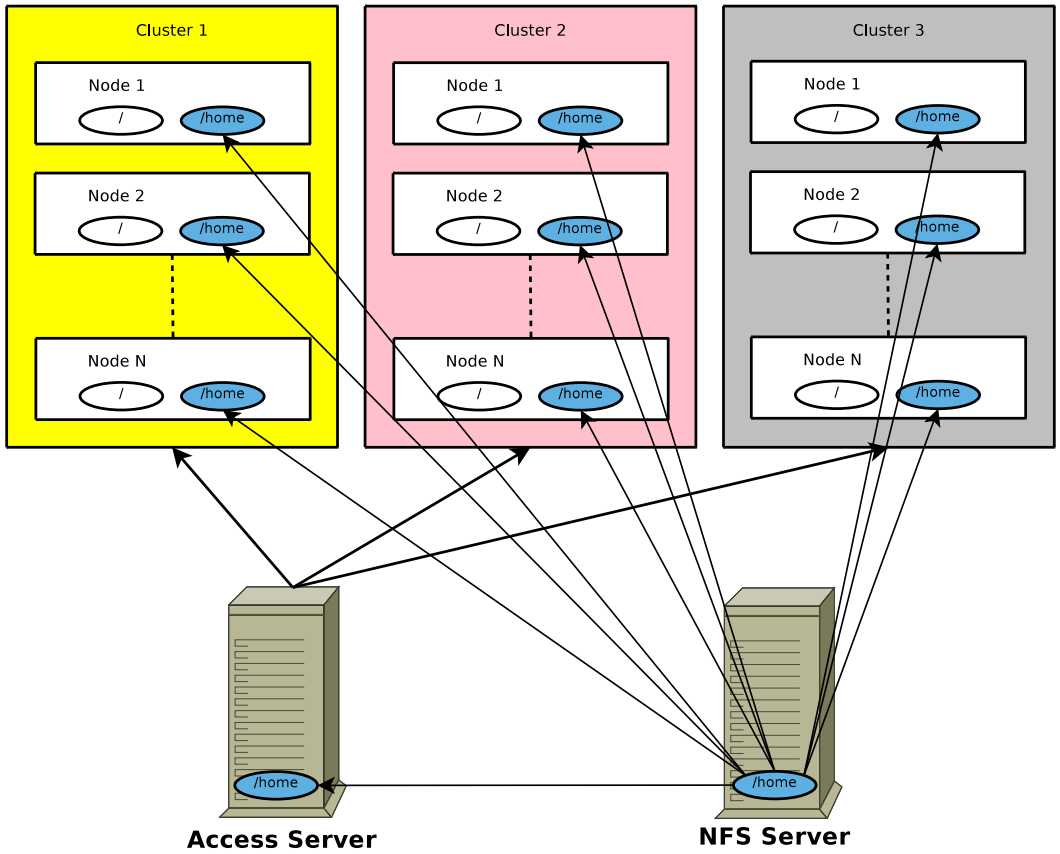
- Access to the Grid'5000 wiki
- Subscribed to {users,platform,announce}@lists.grid5000.fr
- Disk quota for your home directory on every Grid'5000 sites
- Access to Grid'5000

The key to Grid'5000 access is SSH

Warning

You shouldn't expect to be able to use Grid'5000 if you don't understand how SSH works and how it interacts with your home directory.

Shared home directory on a site



Advice

- There are as many NFS servers (and therefore different home directories) as sites
- If you need to share some files between several sites, you must perform the synchronization explicitly (with rsync for instance)

Grid'5000's software stack

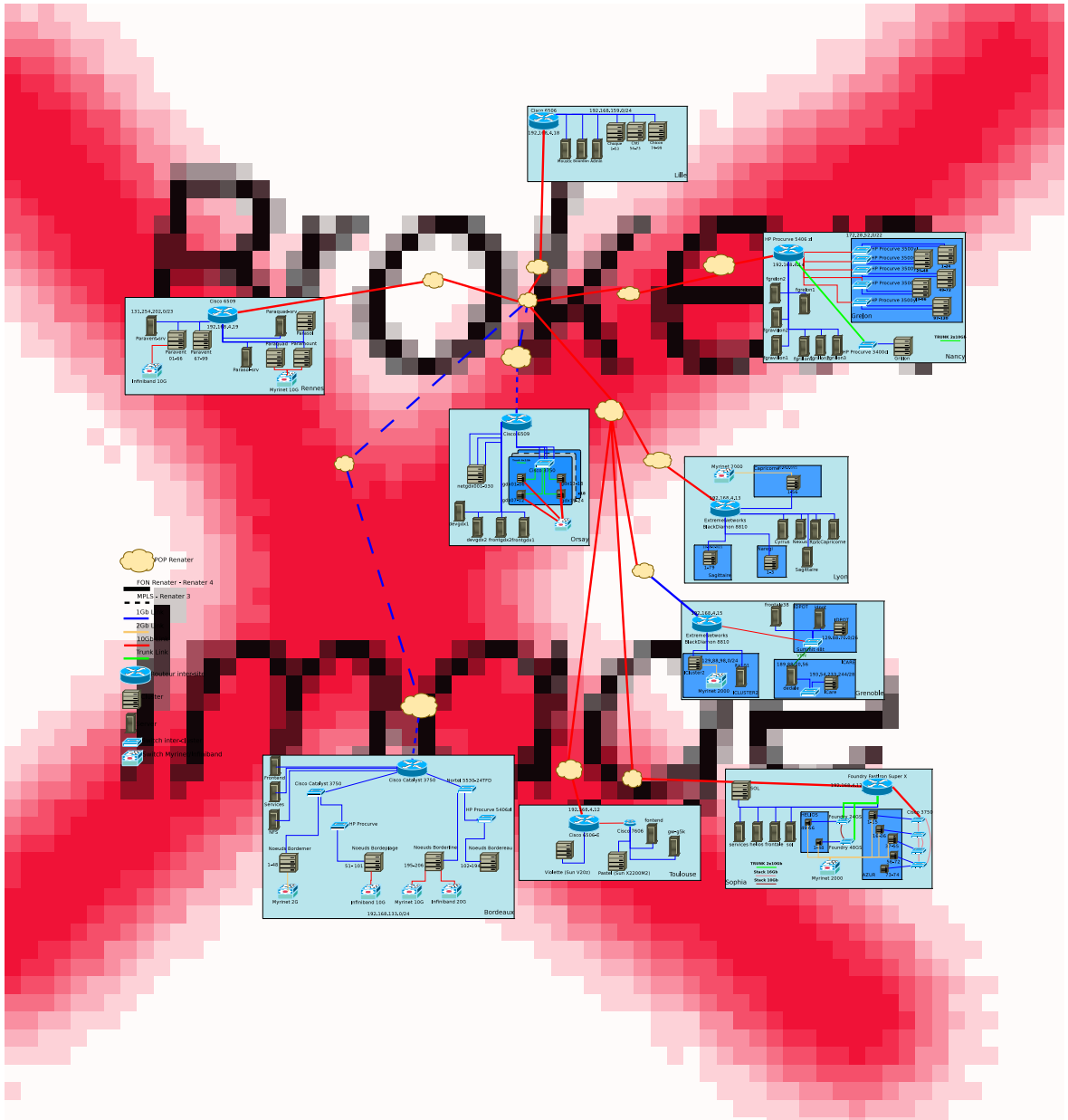
The tools you'll be using are a mixture of

- Standard tools (e.g. ssh, openldap, ganglia, squid, mediawiki, bugzilla, ...)
- Tools dedicated to Grid'5000, developed and supported
  - by teams loosely related to Grid'5000 technical staff (OAR, taktuk, GRUDU)
  - now under the maintenance of the technical staff (kadeploy)
- User contributed tools, sometimes hosted on the grid5000-code project on gforge.inria.fr (e.g. oargrid, katapult, kanon)

All credits or blames do not go the the ALADDIN-G5K development team !

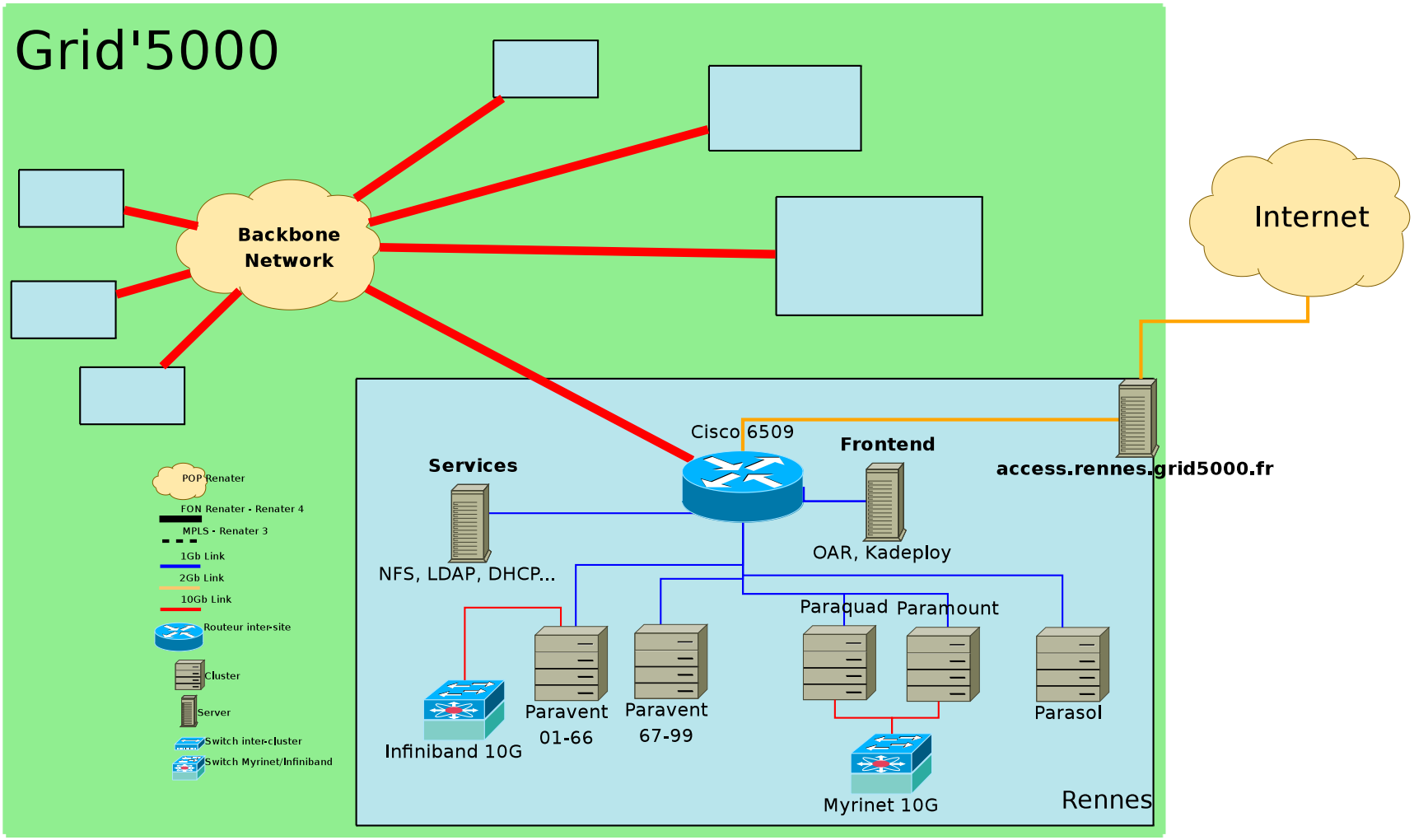
The grid topology

Global view



A site topology

Site view



Grid'5000 resource manager: OAR

Developed under the supervision of Olivier Richard (LIG / Mescal)

What is OAR ?

Definition

OAR is the resource manager used in Grid'5000 to *allocate resources to users* for their experiments. The resource manager *creates jobs* for users, which are basically an *execution time on a set of resources*. Grid'5000 features *1 OAR resource manager per site*.

OAR features includes

- *interactive jobs:* I want resources now for a bunch of time
- *advanced reservations:* I want resources at that date/time for that duration
- *batch jobs:* I want my job to run by itself with this script
- *best effort jobs:* I use many resources but accept to release them at any time
- *deploy jobs:* I want to be granted to deploy a customized OS environment and have full access to the resources
- *powerfull resource filtering/matching:* I want only quad core machines with more than 8GB of RAM located on the same network equipment

What is a resource in Grid'5000 ?

Definition

In Grid'5000 context, a resource is a node of a cluster (a network host) or a part of it: a CPU or a core. A resource is described by a set of properties.

Overview of the properties used by the resource manager to select resources

- Cpu architecture
- Cpu frequency
- Cluster name
- Switch name
- Memory size
- Disk type
- Virtualization capability
- OS reconfiguration capability

What is a batch mode job ?

Use case

When do I use a batch mode job ?

- Your experimentation does not require your intervention after it starts (non-interactive application)
- You can script all the steps involved in your job, from the starting to the result retrieval mechanisms
- You don't care about the start date (the delay will depend on the platform load)

If these requirements are fulfilled, you should use the batch reservation mode

Advice

The batch mode is the best one to optimize the resource utilization. It is preferable with regard to the community

Example of batch mode job

Utilization

Example of use of the batch mode:
I would like to execute launcher.sh on 4 nodes with 10G Myrinet NIC and my job will not last more than 1h15

```
oarsub -p "myri10g=YES" -l nodes=4,walltime=01:15:00 \
       ./launcher.sh
```

Note about the walltime

Be careful to correctly set the walltime value

- If the value is too small, your job will be terminated before it finishes
- If the value is too large, your job will prevent the scheduling to be performed optimally, which is bad with regard to the community
- But if the execution of your job finishes before the walltime, resources are freed for later jobs usage

What is an interactive mode job ?

Use case

When do I use a interactive mode job ?

- You want (a small bunch of) resources NOW for a preliminary experimentation
- Your experimentation requires your intervention once it starts (interactive application)

Advice

Getting access to the resources for an interactive job is not always as quick as one would wish, depending on the platform load and on the amount of resources one requires. As a result one may often not be able to run an interactive job as wished

Example of interactive mode job

Utilization

- You want to run a basic interactive job:

  ```
  oarsub -I
  ```

  You get access to one of the node of the site
- You want to start a job immediately for 15 minutes on nodes featuring Myrinet 10G cards:

  ```
  oarsub -I -p "myri10g=YES" -l nodes=4,walltime=01:15:00
  ```

What is a advance reservation job ?

Use case

When do I use an advance reservation mode job ?

- I need to get access to a set of resources at that date/time precisely
- I have a huge experiment that I will run during the night
- I need several jobs (on different sites) to run at the same time
- My experiment needs my intervention so I need to set the start date/time in order to be present once it start (and I can't use interactive jobs)

Notes

- Advance reservations prevent the scheduler of the resource manager to optimize the platform usage
- Advance reservations cannot give a guaranty that your resource request will be fulfilled: some resources might eventually be broken at the start date, in which case you only get the available ones.
- Advance reservations allow *both* interactive executions and scripted executions

Example of advance reservation job

Utilization

- You want to reserve resources for an interactive session at a given time

```
oarsub -p "myri10g=YES" -l nodes=4,walltime=01:15:00 \
       -r "2008-02-30 11:00:00"
```

- You want to reserve resources to run a script at a given time

```
oarsub -p "myri10g=YES" -l nodes=4,walltime=01:15:00 \
       -r "2008-02-30 11:00:00" "script.sh"
```

Note about the reservation mode

If you use this mode, you will obtain a JobID after the execution of `oarsub`. Once your reservation is started, you can *connect* to your reservation using:

```
oarsub -C JobID
```

Some other features of OAR

View the reservations

```
oarstat
oarstat -f -j JobID
```

Cancel a reservation

```
oardel JobID
```

Get information on the nodes

```
oarnodes
oarnodes -l
oarnodes -s
```

Multi-site resource allocation

OarGrid

OarGrid is a tool built on the top of OAR designed to aggregate multiple site resource allocation. See the man pages of the following commands for more details.

```
oargridsub
oargriddel
oargridstat
```

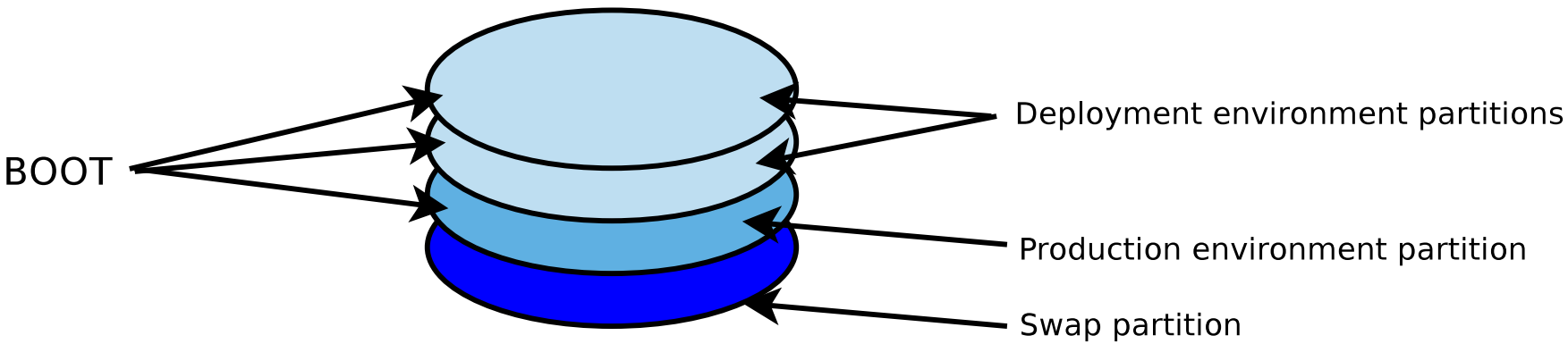
Kadeploy: The reconfiguration tool

- Initial concept and development under the supervision of Olivier Richard (LIG / Mescal)
- Now maintained and developed by Emmanuel Jeanvoine (INRIA / ALADDIN-G5K development team)

Purpose

Modify the entire software stack on the nodes

The Grid'5000 nodes are running with a given operating system based on GNU/Linux. For many reasons, you may want to use something else than the default installation, for example to change the operating system. This is the purpose of the Kadeploy tool.



Modifying the environment on a set of nodes

First step

Perform a resource reservation with OAR and specify that you want to deploy an environment on these nodes

```
oarsub -I -l nodes=4 -t deploy
```

Second step

Launch Kadeploy on the set of nodes

```
kadeploy -e environment -f $OAR_NODEFILE
```

Third step

At the end of the deployment, Kadeploy shows you the nodes that have been correctly deployed or not with your environment.

After your reservation has ended

The nodes will be automatically rebooted on the production environment.

Management of the environments

Deployable environment are recorded in a database. You can use

- An environment provided by the staff (they should be described on the wiki)
- An environment created by another user
- An environment you created yourself

How to list the environments recorded on a site?

List your own environment and the staff maintened environments

Execute `kaenvironment`

List the environment of an other user

Execute `kaenvironment -l user`

The maintained environments

The support team maintains some environments that are usable on all clusters (their kernel has support for the whole range of Grid'5000's hardware). They should be suitable as a seed for customization for the majority of users. These environments are based on the stable and unstable Debian distributions.

Three flavors for the environments

- **base**: provides a minimal software set and to avoid unnecessary services annoyances
- **nfs**: same package list as **base** plus the ability to log in with your LDAP account and access your home directory on the deployed node
- **big**: provides the same package list as **nfs** and a set of additional packages used for compilation, debugging, text edition, ...

Create your own environment

Modify an existing environment

- Deploy the existing environment and modify it
- Dump the deployed partition (`tgz-g5k` tool)
- Provide a description of your environment and record it with the `karecordenv` tool

Create your own environment from scratch

- Deploy any environment on the 1st deployment partition to become root
- Use the 2nd deployment partition as a target to install your new OS
- Use a virtual machine to install the OS from an ISO cd on the target partition or use a software like debootstrap to install a Debian based OS
- Dump the deployed partition with the `tgz-g5k` tool
- Provide a description of your environment and record it with the `karecordenv` tool

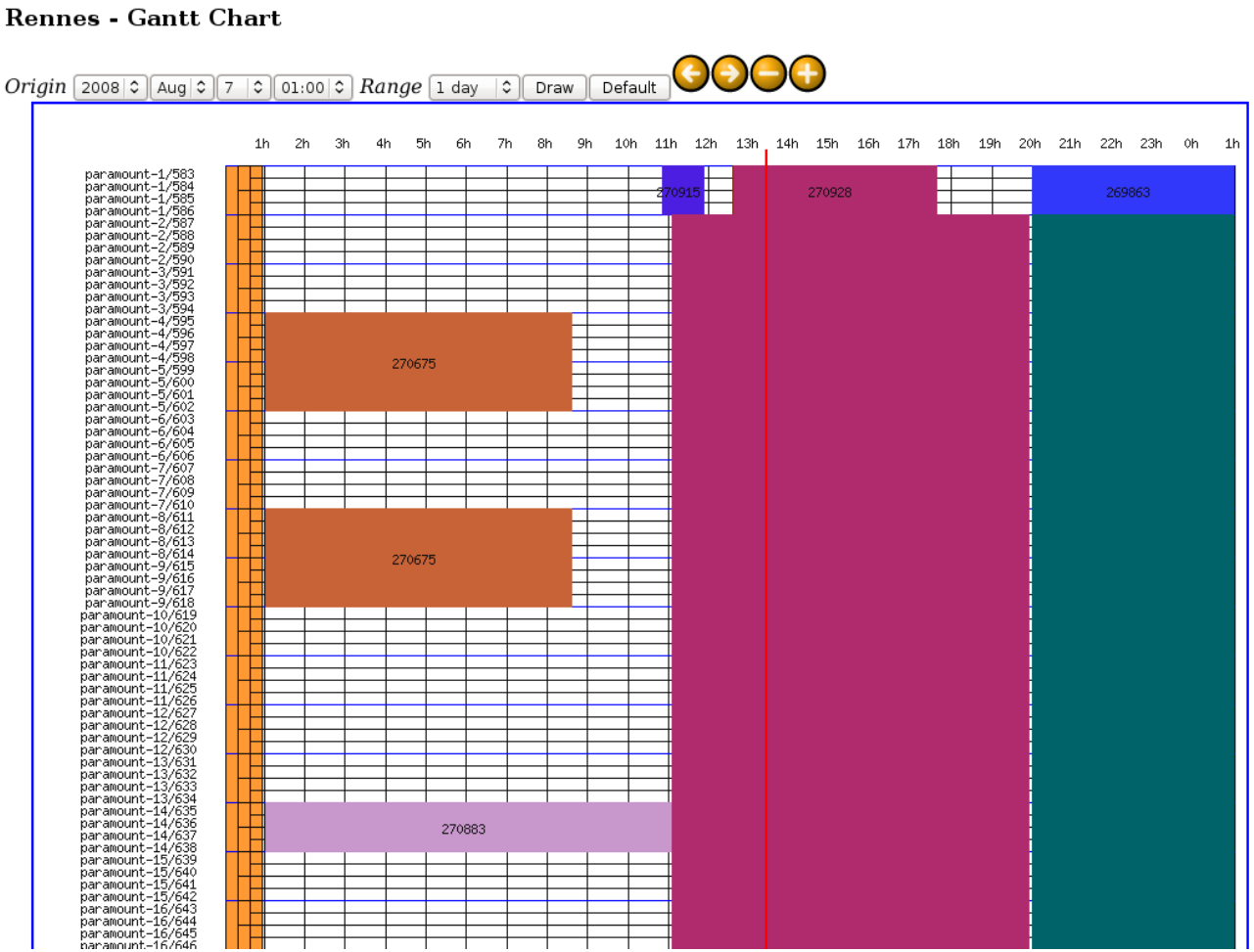
Benefits of Kadeploy

- By creating your own environment, you can have the libs you want and you can tweak the system
- By deploying your own environment, you can become root on the nodes
- By using the `-t deploy` options of OAR, you gain access to the `kareboot` and `kaconsole` commands to freshly boot the default environment.
- By using your own environment, you can reproduce your experiments without being bothered by a system update performed by the administrator

The experience steering tools

The Gantt chart

Graphical view of the job submitted on the platform



Monika

Graphical view of the job submitted of the state of the nodes

Grid5000 Lyon OAR nodes

Summary:

OAR node status	Free	Busy	Total
Nodes	52	75	135
Cores	104	150	270

Reservations:

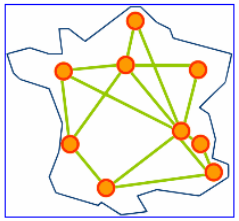
capricorne-1	148954 148954	capricorne-2	Absent	capricorne-3	Free Free	capricorne-4	148965 148965
capricorne-5	148965 148965	capricorne-6	Free Free	capricorne-7	148964 148964	capricorne-8	Free Free
capricorne-9	148964 148964	capricorne-10	148963 148963	capricorne-11	148946 148946	capricorne-12	148960 148960
capricorne-13	148953 148953	capricorne-14	148963 148963	capricorne-15	148959 148959	capricorne-16	Free Free
capricorne-17	148951 148951	capricorne-18	148963 148963	capricorne-19	Free Free	capricorne-20	148945 148945
capricorne-21	Free Free	capricorne-22	Free Free	capricorne-23	Free Free	capricorne-24	Free Free
capricorne-25	Free Free	capricorne-26	Free Free	capricorne-27	Absent	capricorne-28	148965 148965
capricorne-29	Absent	capricorne-30	Free Free	capricorne-31	Free Free	capricorne-32	Free Free
capricorne-33	Free Free	capricorne-34	148949 148949	capricorne-35	Absent	capricorne-36	148965 148965
capricorne-37	Free Free	capricorne-38	Free Free	capricorne-39	Free Free	capricorne-40	Free Free
capricorne-41	148965 148965	capricorne-42	148965 148965	capricorne-43	Free Free	capricorne-44	Free Free

Fine grain monitoring

Renater

Renater — *GRID5000 - Network statistics* — RENATER Contact

last 2 hours
last day
weekly
monthly
yearly
Or choose a period, STOP field can be empty.
Start date (YYYYMMDD)
Stop date (YYYYMMDD) ok delete

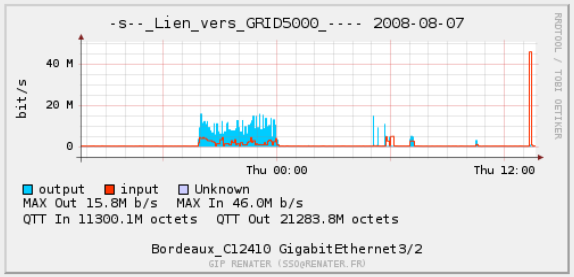


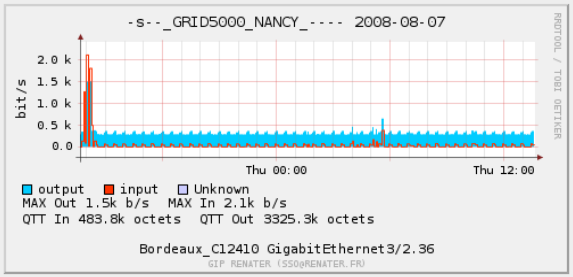
Bordeaux

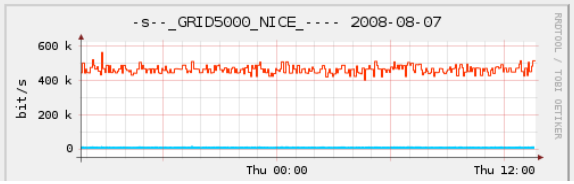
Dark fiber site interface:

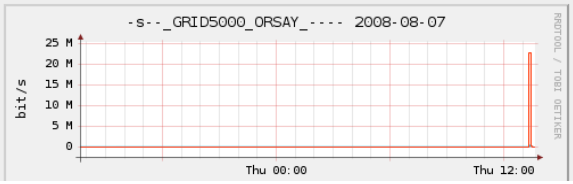
Pas d'interfaces

MPLS site interfaces:



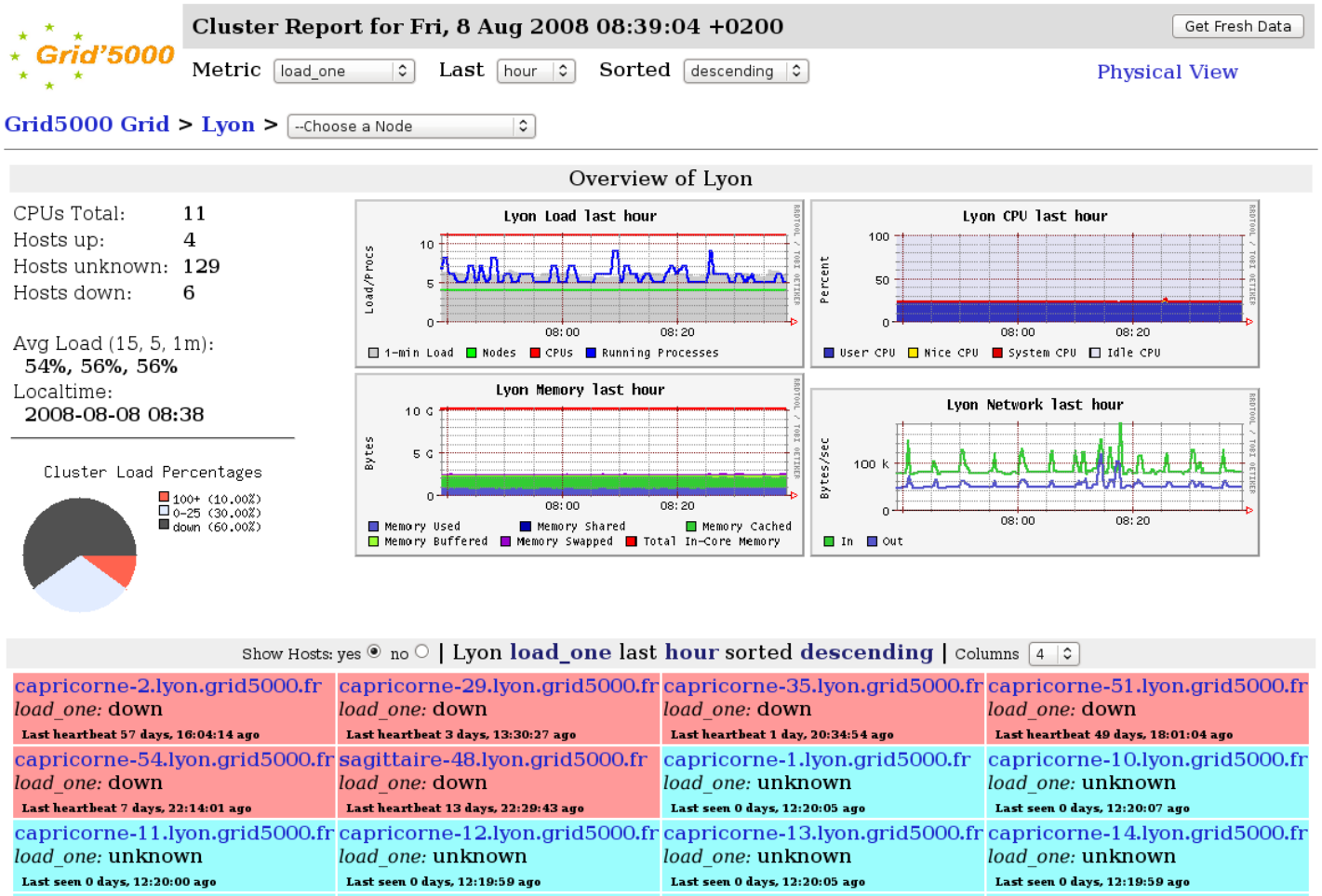






Fine grain monitoring

Ganglia

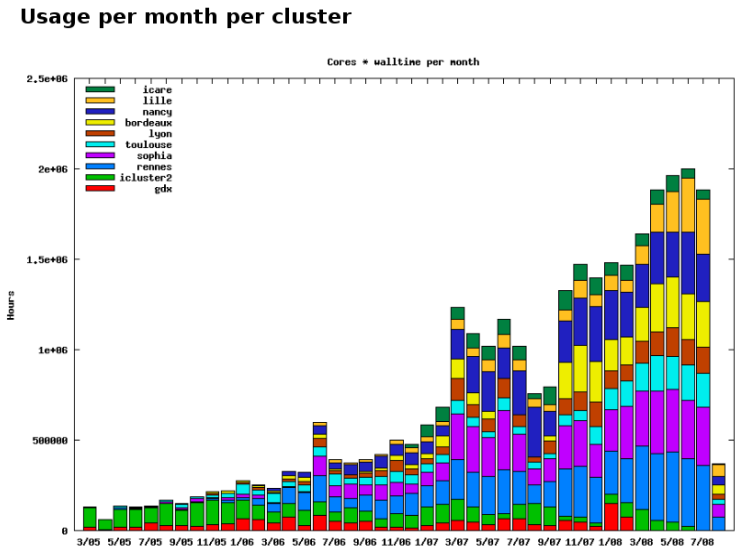


Show Hosts: yes ◉ no ○ | Lyon **load_one** last **hour** sorted **descending** | Columns 4

capricorne-2.lyon.grid5000.fr *load_one:* down **Last heartbeat 57 days, 16:04:14 ago**	capricorne-29.lyon.grid5000.fr *load_one:* down **Last heartbeat 3 days, 13:30:27 ago**	capricorne-35.lyon.grid5000.fr *load_one:* down **Last heartbeat 1 day, 20:34:54 ago**	capricorne-51.lyon.grid5000.fr *load_one:* down **Last heartbeat 49 days, 18:01:04 ago**
capricorne-54.lyon.grid5000.fr *load_one:* down **Last heartbeat 7 days, 22:14:01 ago**	sagittaire-48.lyon.grid5000.fr *load_one:* down **Last heartbeat 13 days, 22:29:43 ago**	capricorne-1.lyon.grid5000.fr *load_one:* unknown **Last seen 0 days, 12:20:05 ago**	capricorne-10.lyon.grid5000.fr *load_one:* unknown **Last seen 0 days, 12:20:07 ago**
capricorne-11.lyon.grid5000.fr *load_one:* unknown **Last seen 0 days, 12:20:00 ago**	capricorne-12.lyon.grid5000.fr *load_one:* unknown **Last seen 0 days, 12:19:59 ago**	capricorne-13.lyon.grid5000.fr *load_one:* unknown **Last seen 0 days, 12:20:05 ago**	capricorne-14.lyon.grid5000.fr *load_one:* unknown **Last seen 0 days, 12:19:59 ago**

Kaspied

Software developed by Lucas Nussbaum (LIG / Mescal)



Set of statistics about the use of the platform

- Per site
- Per kind of job (best-effort, interactive)
- Per user
- Per laboratories

User reports

Grid'5000 user reports

This report aims at providing information about you and your usage of the Grid'5000 platform.
Please give relevant information so that we can present works being done on the platform in conferences, reports and project evaluations... As it is public, the content of your report can be moderated *(for discussions or other remarks, please use the mailing list: users@lists.grid5000.fr)*. All information are stored in a database, so you will be able to edit everything again if needed.
Thanks for filling your report *(in english preferably)* and keeping it up to date...

User information | Experiments | Publications | Collaborations | Highlights | View my report | List all reports go

Bordeaux:

- **Alexandre Denis** *(Researcher (CR))*, Runtime LaBRI Bordeaux *(2008-07-28 23:45:25)*
- **Kristian Kocher** *(Master student)*, runtime inria bordeaux - sud ouest Bordeaux *(2008-06-30 11:57:37)*
- **Francois Trahay** *(PhD student)*, Runtime LaBRI Bordeaux *(2008-06-05 14:31:37)*
- **Stéphanie Moreaud** *(PhD student)*, Runtime LaBRI Bordeaux *(2007-11-29 15:35:46)*
- **Olivier Teytaud** *(Researcher (CR))*, TAO (Inria Futurs) Lri (Cnrs, Inria, Univ Paris-Sud) Orsay (France) *(2007-11-24 00:19:28)*
- **Pierre Ramet** *(Lecturer/Associate Professor (MCF))*, ScAlApplix LaBRI Bordeaux *(2007-11-16 10:00:43)*
- **Pascal Henon** *(Researcher (CR))*, ScAlApplix (INRIA) LaBRI Bordeaux *(2007-11-09 15:29:00)*
- **Adrien Goeffon** *(Post-Doc)*, MAGNOME LaBRI Bordeaux *(2007-11-06 11:55:02)*
- **David Sherman** *(Researcher (CR))*, Magnome INRIA Futurs Bordeaux *(2007-10-03 11:05:04)*
- **brice goglin** *(Engineer)*, LaBRI Bordeaux *(2007-09-05 14:41:57)*
- **Nicolas Bonichon** *(Lecturer/Associate Professor (MCF))*, Cepage LaBRI Bordaeux *(2007-03-20 18:17:27)*
- **Nathalie Furmento** *(Engineer)*, Runtime LaBRI Bordeaux *(2007-02-26 15:54:27)*
- **Olivier Aumage** *(Researcher (CR))*, Runtime LaBRI Bordeaux *(2007-02-26 13:38:36)*
- **Elisabeth Brunet** *(PhD student)*, Runtime LaBRI Bordeaux *(2007-02-26 12:07:09)*
- **Brice Goglin** *(Researcher (CR))*, Runtime LaBRI Bordeaux *(2007-02-26 11:28:02)*
- **Christophe Frezier** *(Engineer)*, Runtime LaBRI Bordeaux *(2007-02-23 17:15:20)*
- **Nicolas Richart** *(PhD student)*, ScAlApplix LaBRI Bordeaux *(2007-02-21 17:38:00)*
- **Mickael Raynaud** *(Engineer)*, Iparla LaBRI / Inria Futurs Bordeaux *(2007-02-21 17:20:29)*
- **Aurelien Esnard** *(Lecturer/Associate Professor (MCF))*, ScAlApplix LaBRI Bordeaux *(2007-02-21 17:08:13)*
- **Guilhem Caramel** *(Engineer)*, ScAlApplix LaBRI Bordeaux *(2007-02-21 17:01:40)*
- **Mathieu Souchaud** *(Engineer)*, ScAlApplix LaBRI Bordeaux *(2007-02-21 17:00:44)*
- **Frank Prat** *(Post-Doc)*, Magique3D LMA Pau *(2006-08-04 14:19:53)*
- **Samuel Thibault** *(PhD student)*, Runtime LaBRI Bordeaux *(2006-08-04 13:47:26)*
- **Mathieu Bernatet** *(Master student)*, ANR LEGO/Numasis LaBRI Bordeaux *(2006-08-03 20:07:02)*
- **Stephane Blanchard** *(Master student)*, ANR LEGO/NUMASIS LaBRI Bordeaux *(2006-08-03 20:04:32)*
- **Olivier Coulaud** *(Senior researcher (DR))*, ScAlApplix Inria Futurs Bordeaux *(2006-08-03 16:56:20)*
- **Francois Broquedis** *(Master student)*, LEGO/NUMASIS LaBRI Bordeaux *(2006-08-03 16:26:23)*
- **Jérôme Clet-Ortega** *(Master student)*, ANR LEGO LaBRI Bordeaux *(2006-08-03 16:25:12)*
- **Guillaume Anciaux** *(PhD student)*, Scalapplix LABRI Bordeaux *(2006-08-03 16:18:35)*

(29 reports, 24 experiments, 11 publications, 4 collaborations, 60 users)

Grenoble:

- **Alexander Klaser** *(PhD student)*, LEAR LJK Grenoble *(2008-07-23 19:09:20)*
- **Pierre-Francois Dutot** *(Lecturer/Associate Professor (MCF))*, MOAIS LIG Grenoble *(2008-07-21 15:55:45)*
- **Xavier Besseron** *(PhD student)*, MOAIS LIG Grenoble *(2008-07-07 16:33:56)*
- **Sami Achour** *(PhD student)*, MOAIS LIP Grenoble *(2008-07-01 16:06:46)*
- **Lucas Schnorr** *(PhD student)*, MOAIS LIG Monbonnot *(2008-06-23 10:54:20)*
- **Krzysztof Rzadca** *(PhD student)*, MOAIS LIG *(2008-06-17 16:38:27)*
- **Frederic Bouquet** *(Master student)*, LIG MESCAL *(2008-05-27 14:15:49)*

Grid'5000 user reports

This report aims at providing information about you and your usage of the Grid'5000 platform.
Please give relevant information so that we can present works being done on the platform in conferences, reports and project evaluations... As it is public, the content of your report can be moderated *(for discussions or other remarks, please use the mailing list: users@lists.grid5000.fr)*. All information are stored in a database, so you will be able to edit everything again if needed.
Thanks for filling your report *(in english preferably)* and keeping it up to date...

User information | Experiments | Publications | Collaborations | Highlights | View my report | List all reports go

User information

Thomas Ropars (PhD student)
Paris IRISA Rennes, France (Rennes)
Email address: tropars@irisa.fr

Experiments

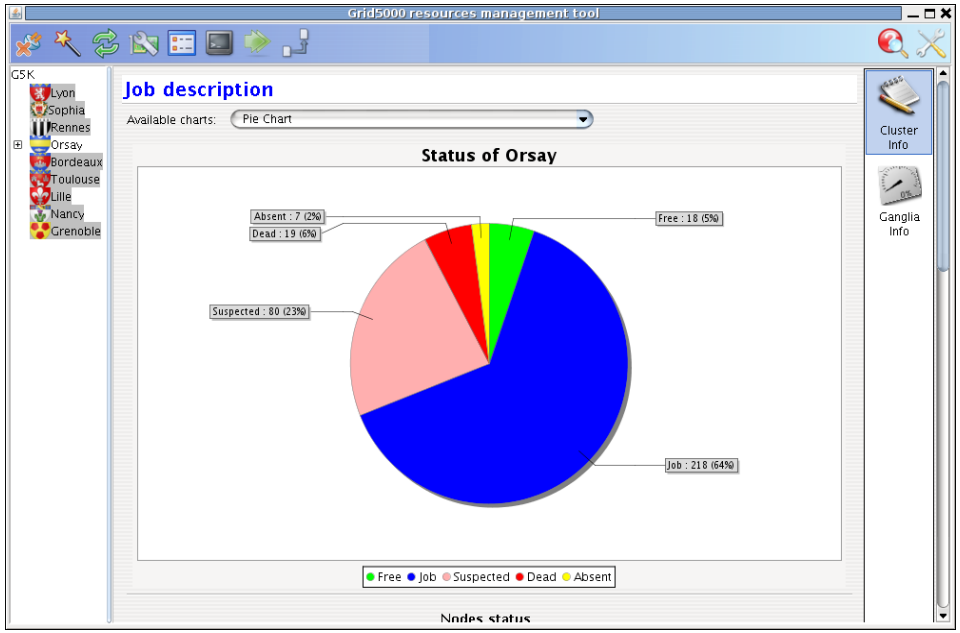
- **Application Monitoring in Vigne (Middleware) [achieved]**
  Description: Vigne is a Grid Operating System. We have tested the application monitoring system of Vigne and especially the failure detection. To do this, we randomly kill some of the processes of the applications executed by Vigne to see if the failures were detected and the failed applications re-scheduled.
  Results:
- **Monitoring cost of GAMoSe (Middleware) [achieved]**
  Description: GAMoSe is an Application Monitoring System designed for grids. It is designed to handle high availability and scalability issues. A set of monitoring mechanisms are used to effectively monitor nodes and application processes. GAMoSe has been integrated into the Vigne Grid Operating System. For this experiments, Vigne is deployed on all the nodes and applications are submitted. Failures are simulated with kill signals send to some applications. Through this experiement, we want to show that GAMoSe is able to provide dependable information with a minimal cost on Grid performances.
  Results:
- **Evaluation of O2P (Middleware) [in progress]**
  Description: O2P is an optimistic message logging protocol that aims at providing fault tolerance for message passing applications. O2P is implemented in Open MPI. We want to evaluate the cost of O2P on failure free execution using the Nas Parallel Benchmarks. We want to compare normal execution with execution using O2P regarding execution time and message size.
  Results:

Publications

- **GAMoSe: An Accurate Monitoring Service for Grid Applications [2007]** *(international)*
  EntryType: inproceedings
  Author: Ropars, Thomas and Jeanvoine, Emmanuel and Morin, Christine
  Month: July
  Booktitle: 6th International Symposium on Parallel and Distributed Computing (ISPDC 2007)
  Pages: 295--302
  Address: Hagenberg, Austria
  Keywords: GRID, MONITORING, Vigne
- **Providing QoS in a Grid Application Monitoring Service [2006]** *(international)*
  EntryType: techreport
  Author: Ropars, Thomas and Jeanvoine, Emmanuel and Morin, Christine
  Number: RR-6070
  Address: IRISA, Rennes, France
  Type: Research Report
  Institution: IRISA/Paris Research group, Université de Rennes 1, EDF R&D, INRIA
  Url: http://hal.inria.fr/inria-00121059

Grudu: Grid'5000 Reservation Utility for Deployment Usage

Software developed by David Loureiro under the supervision of Eddy Caron (ENS / Graal)



GUI

- Grid'5000 status
- Job status
- Resource allocation
- Image deployment through Kadeploy
- See: `http://grudu.gforge.inria.fr`

Katapult

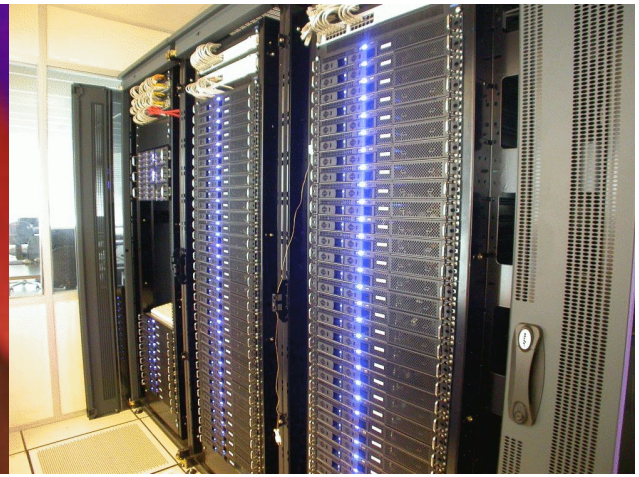
Software developed by Lucas Nussbaum (LIG / Mescal)

Automates some tasks for experiments using deployments

- Deploying the nodes
- Re-deploying the nodes if too many of them failed
- Copying the user's SSH key to the nodes
- See: `http://www-id.imag.fr/~nussbaum/katapult.php`

Planned evolutions

Hardware evolutions

- Renewal of machines/clusters (depending on fundings)
- Network backbone evolutions (Renater 5)
- New equipments (network probes / power consumption probes)

Software evolutions

The development team is actively working on several software

- Kaspied-NG: set of indicators about the Grid'5000 usage
- Kavlan: network isolation
- OAR: several improvements to fit the Grid'5000 usage
- Kadeploy: improvement of the robustness and support to virtualization

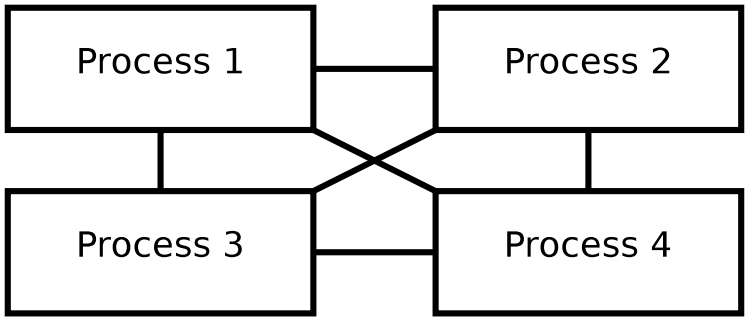
Event organization for the community

- Seminars
- Newsletter
- Promotion of the contribution to the platform (tools developed by users that can be used by other)
- Tutorial sessions
- Summer/Winter schools

Gridify your applications

Gridify your applications

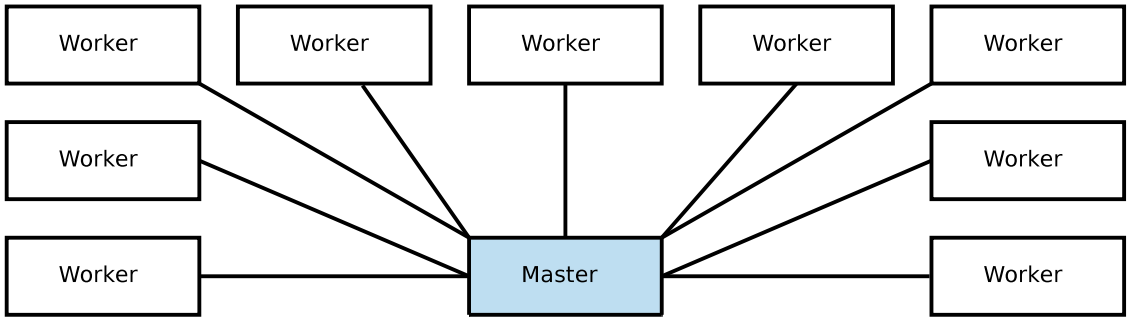
Parallel applications



- Set of parallel processes that perform a lot of communications
- Several paradigms: message passing, distributed shared memory
- Great performances when executed on a single cluster that has a fast network interconnection (Myrinet, Infiniband)
- Example of domains: linear algebra, ray tracing, ciphering

Gridify your applications

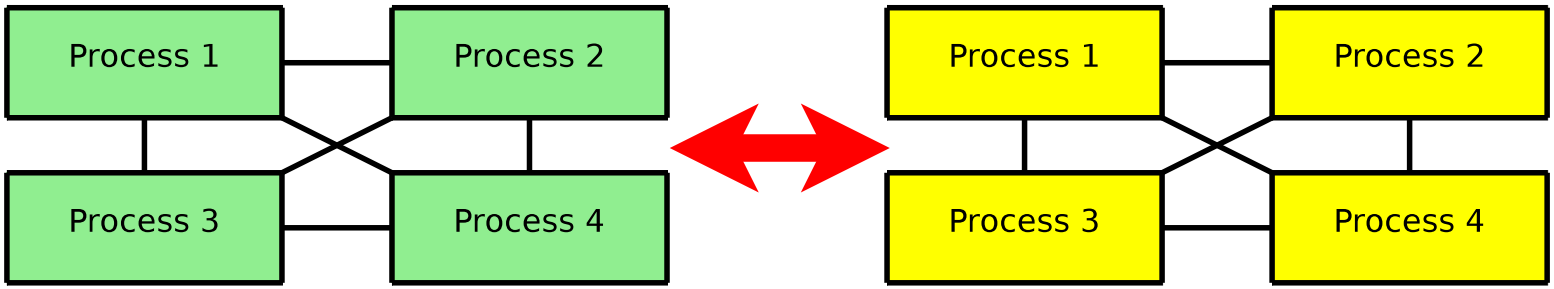
Master/worker applications



- Large set of independent processes
- A master node is responsible for the distribution of the load on the worker nodes
- Since the processes are independent, the latency is not a important criteria. Good candidate for the grid
- Example of domain: Monte-Carlo simulation

Gridify your applications

Code coupling applications



- Several coupled applications
- This kind of application is suitable for a grid if the communication between the several applications is not sensitive to the latency
- Example of domain: multi-physics simulations (for instance, one application for the ocean and another for the atmosphere)

Some links

Usefull links for novice people

The Grid'5000 wiki

- The main page:
  `https://www.grid5000.fr/mediawiki/index.php/Grid5000:Home`
- The user pages:
  `https://www.grid5000.fr/mediawiki/index.php/Category:Portal:User`
- The platform status:
  `https://www.grid5000.fr/mediawiki/index.php/Status`

The mailing lists

- At the opening of your account, your email will be automatically added to the Grid'5000 user list. You will be able to send your questions to the same list by using the following address : users@lists.grid5000.fr
- If you are interested by the development of the platform, you can subscribe to the devel mailing-list: `http://lists.grid5000.fr/wws/subrequest/devel`