

Introduction to Grid'5000

Aladdin-G5K development team

INRIA

April 27, 2011



Slides by Emmanuel Jeanvoine and David Margery

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

Conception

When preparing an experiment, using Grid'5000 or another tool, the process must be understood

- What to I want to measure ?
- What parameters will be made to change ?
- What experimental conditions need measured ?

Implementation in Grid'5000

- How will I map my experiment to the topology of Grid'5000 ?
- What are the side effects ?

Producing relevant and reproducible 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 reproducible results when doing full-scale experiments is a real challenge.

Other testbeds for experiments

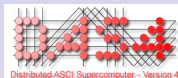


- Wide area testbed composed of about 1077 nodes spread over 570 sites world-wide
- Allocation of slices: virtual machines
- Designed for experiments Internet-wide: new protocols for Internet, overlay networks (file-sharing, routing algorithm, multi-cast, ...)
- building a federation with different plate-forms



- 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 197 nodes (about 1576 cores in 6 clusters), with accelerators
- On the fly network backbone reconfiguration (optical routers with configurable wavelength)
- The software stack is not reconfigurable
- up since December 2010.

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



- FutureGrid is a (5600 core) Science/Computer Science Cloud but it is more accurately a virtual machine based simulation environment.
- In Phase 1 (now), machines are statically partitioned between HPC, Nimbus and Eucalyptus configurations.

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.

Some definitions (Cloud computing)

Cloud computing

Paradigm where a provider gives access to resources or services as a service, without relying on software installed on the consumer's machine and without disclosing the physical location nor the implementation details. Linked a pay for what you use. In a sense, those resources/services reside in the cloud of many diagrams

IaaS

Infrastructure as a Service. In its basic form, users get access to virtual machines, storage services and network links they have to configure to their needs.

PaaS

Platform as a Service. Users get access to higher level building blocks for their applications (MapReduce)

SaaS

Software as a Service. Software is used without local installation (eg. Gmail, etc. . .)

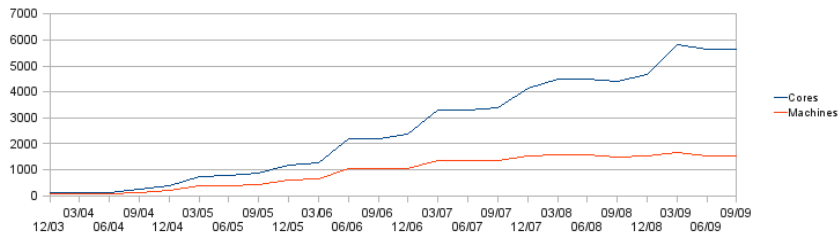
General presentation of Aladdin/Grid'5000

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 - 2008-2012

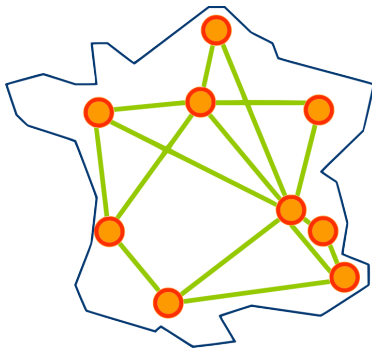
Hardware

Evolution matérielle de Grid'5000



A nation-wide grid, expanding overseas

9 sites



Sites

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

New sites

Porto-Alegre (Brazil), Reims, Luxembourg

The hardware

CPU families

- AMD Opteron (62%), Intel Xeon EMT64 (32%)
- MonoCore (38.5%), DualCore (32%), QuadCore (27%), 12-core (2.7%)
- Practically all machines are bi-processors (except 10)
- 2 GPU clusters

High performance networks

- Myrinet 2000 (222 cards)
- Myrinet 10G (509 cards)
- InfiniBand SDR (160 cards), DDR (238 cards), QDR (84 cards)

At a glance

- 7244 cores / 9 sites
- Gigabit Ethernet interconnect everywhere and 10Gb/s backbone
- More informations on: <https://www.grid5000.fr/>

A 10Gb/s backbone network

Renater 5



Board of directors

- Scientific director: Frédéric Desprez
- Technical director: David Margery

Executive committee

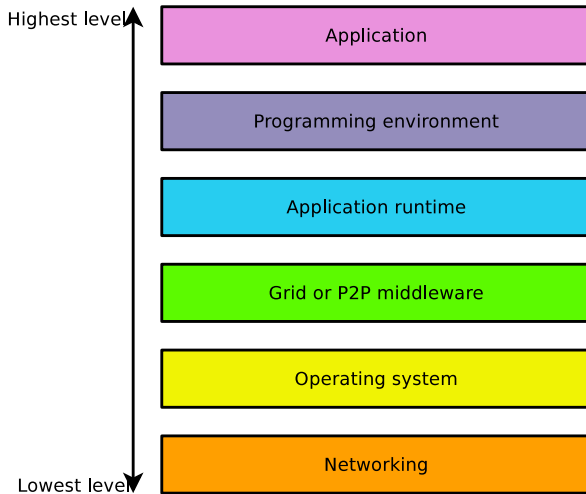
1 scientist representative from each site, 1 representative from RENATER and the board of directors.

Technical committee

Set of engineers divided in two teams

- The support staff: 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 (developments, meetings, feedbacks, ...)

A large research applicability



Context of work

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 Glite expert)
- to the Grid'5000 users' mailing-lists

And please participate to the community 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
 - ▶ with the mail known to Grid'5000
 - ▶ Manage your mail on <http://api.grid5000.fr/ui/account>
- Disk quota for your home directory on every Grid'5000 sites
- Access to Grid'5000

Please note

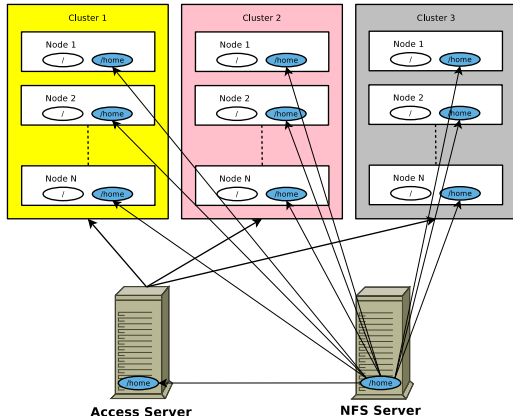
Manage your account, including expiration, on <http://api.grid5000.fr/ui/account>

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)

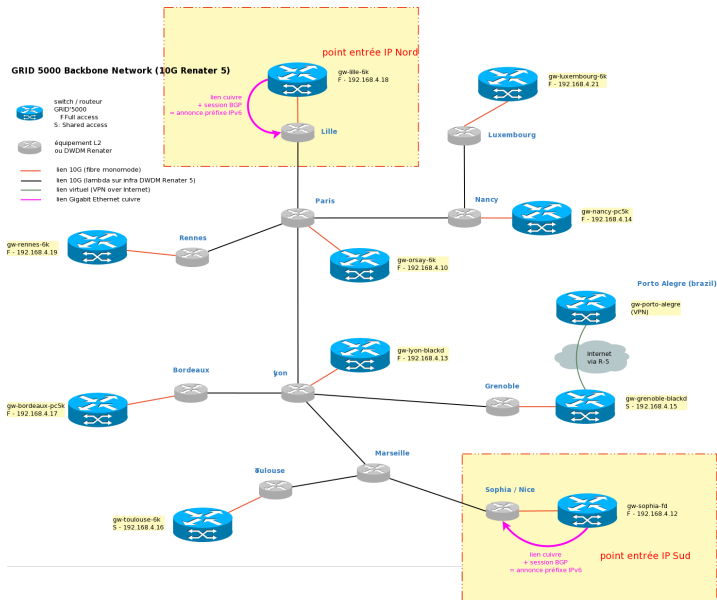
The tools you'll be using are a mixture of

- 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 (kadeploy3)
- User contributed tools, sometimes hosted on the grid5000-code project on gforge.inria.fr (e.g. oargrid, katapult, kanon)

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

The grid topology

Global view



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, by default, a node of a cluster (a network host) or a part of it: a CPU or a core. Vlans and subnets are special resources also managed by OAR. 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
- high speed networking 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
```

Example of a complex submission

You want 1 core on 2 nodes on the same cluster with 4096 GB of memory and Infiniband 10G + 1 cpu on 2 nodes on the same switch with bicore processors for a walltime of 4 hours:

```
oarsub -I -l "{memnode=4096 and ib10g='YES'}/cluster=1/nodes=2/core=1\
+{cpucore=2}/switch=1/nodes=2/cpu=1,walltime=4:0:0"
```

Some other features of OAR

View the reservations

```
oarstat
```

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

Check the status of a reservation

```
oarstat --state -j JobId
```

Cancel a reservation

```
oardel JobID
```

Get information on the nodes

```
oarnodes
```

```
oarnodes -l
```

```
oarnodes -s
```

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
```

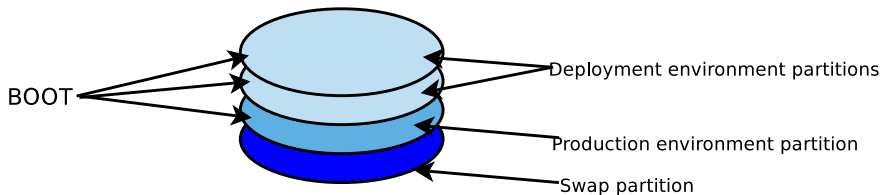
Kadeploy: The reconfiguration tool

- Initial concept and development under the supervision of Olivier Richard (LIG / Mescal)
- Now maintained and developed by Emmanuel Jeanvoine (INRIA / ALADDIN-G5K development team until summer 2009)

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

```
kadeploy3 -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.

Kadeploy tips and tricks

Install your key in the authorized_keys of root

```
kadeploy3 -e environment -f $OAR_NODEFILE -k ~/.ssh/id_rsa.pub
```

Format xda5 (/tmp or /scratch) during deployment

```
kadeploy3 -e environment -f $OAR_NODEFILE -k ~/.ssh/id_rsa.pub -r
```

Deploy from default environment (when available)

```
kadeploy3 -e environment -f $OAR_NODEFILE -k ~/.ssh/id_rsa.pub -u
```

Boot using grub (when available)

```
kadeploy3 -e environment -f $OAR_NODEFILE -k ~/.ssh/id_rsa.pub --grub
```

Deployable environment can be 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 public environments

Execute `kaenv3 -l`

List the environment of an other user

Execute `kaenv3 -l -u user`

The reference 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 Debian distributions and are called reference environments.

Three flavors for the environments

- **min**: this new flavour basically only boots with network connectivity and remote console on all clusters.
- **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, ...
- **xen**: provides the same package list as **base** and boots into a Xen Dom0

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 optionally record it with the `kaenv3` 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
- add the needed disk and network drivers in the kernel/initrd
- Dump the deployed partition with the `tgz-g5k` tool
- Provide a description of your environment and record it with the `kaenv3` 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 `kareboot3`, `kapower3` and `kaconsole3` commands.
- 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 API

Developed under the supervision of David Margery (INRIA) by Cyril Rohr (INRIA)

All API's based on a common service architecture

Principles

- Uses HTTP at the application layer
- Standard error codes (404, ...)
- Robust servers, cache technology, proxies
- Uses HTTP's verbs (GET, PUT, POST, DELETE)

Benefits

- All Grid'5000 APIs use this service architecture
- Independant from the language used, as long as an http library is available

Implemented API's in the stable version

- Reference API. A versionned description of Grid'5000
- Monitoring API. A dynamic description of the state of Grid'5000 ressources
- Metrology API. To access all metrology information given by the platform.
- Jobs API. To access current and future jobs, as well as to create new jobs.
- Deployments API. To deploy software using kadeploy3

Some technical considerations

Stability of the APIs

- The APIs are bundled in versions `https://api.grid5000.fr/VERSION`
- The current stable version is **2.0**
- The next version is available as **sid**

Authentication considerations

- because OAR and Kadeploy are made available through the API, authentication is needed
 - ▶ From outside Grid'5000, only `https://api.grid5000.fr` is available through basic authentication
 - ▶ from inside Grid'5000, `api.grid5000.fr` is only available from frontends, using ident as authentication mechanism
- You should not write scripts that keep a password or a certificate in script data
- Most examples given with `https://localhost:3443` as an established ssh tunnel

```
ssh -L 3443:api.grid5000.fr:443 access.machine
```

Reference and Monitoring API

Reference API

Get a machine readable description of the platform

- 1 from the list of sites
- 2 to detailed information about nodes

Monitoring API

Get machine readable state information for the platform

- 1 from a global status
- 2 to a detailed status about nodes

Other tools

KaVLAN

Isolate your experiment from LAN traffic on the site

Deploy alternative cluster technologies in Grid'5000

- Isolate the machines of your reservation from ethernet broadcast
- deploy your own tftp server, dhcp server, . . .

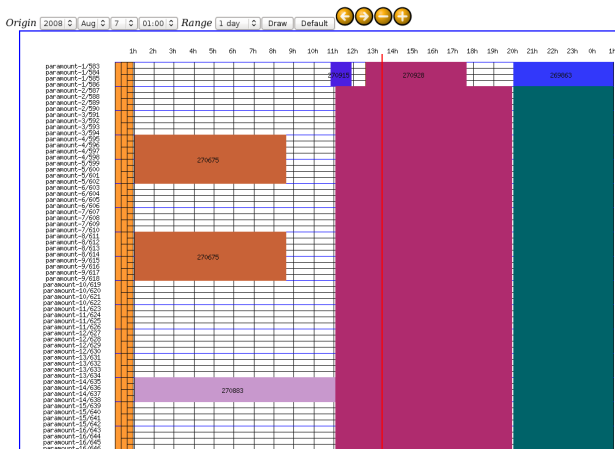
Grid'5000 in Grid'5000

Could be built using this tool.

The Gantt chart

Graphical view of the job submitted on the platform

Rennes - Gantt Chart



Grid5000 Lyon OAR nodes

Summary:

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

Reservations:

capricorne-1	148954148954	capricorne-2	Absent	capricorne-3	Free Free	capricorne-4	148965148965
capricorne-5	148965148965	capricorne-6	Free Free	capricorne-7	148964148964	capricorne-8	Free Free
capricorne-9	148964148964	capricorne-10	148963148963	capricorne-11	148946148946	capricorne-12	148960148960
capricorne-13	148953148953	capricorne-14	148963148963	capricorne-15	148959148959	capricorne-16	Free Free
capricorne-17	148951148951	capricorne-18	148963148963	capricorne-19	Free Free	capricorne-20	148945148945
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	148965148965
capricorne-29	Absent	capricorne-30	Free Free	capricorne-31	Free Free	capricorne-32	Free Free
capricorne-33	Free Free	capricorne-34	148949148949	capricorne-35	Absent	capricorne-36	148965148965
capricorne-37	Free Free	capricorne-38	Free Free	capricorne-39	Free Free	capricorne-40	Free Free
capricorne-41	148965148965	capricorne-42	148965148965	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

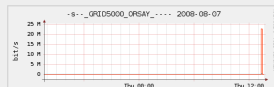
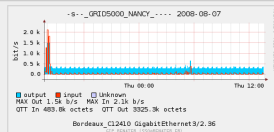
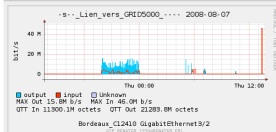


Bordeaux

Dark fiber site interface:

Pas d'interfaces

MPLS site interfaces:



Fine grain monitoring

Ganglia



Cluster Report for Fri, 8 Aug 2008 08:39:04 +0200

Get Fresh Data

Metric Last Sorted

Physical View

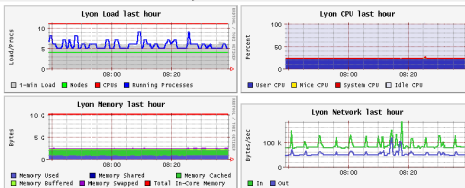
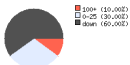
Grid5000 Grid > Lyon >

Overview of Lyon

CPUs Total: 11
Hosts up: 4
Hosts unknown: 129
Hosts down: 6

Avg Load (15, 5, 1m):
54%, 56%, 56%
Localtime:
2008-08-08 08:38

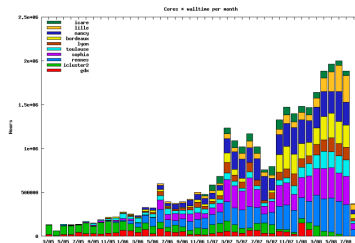
Cluster Load Percentages



Show Hosts: ☒ yes ☐ no | Lyon load_one last hour sorted descending | Columns

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, 12:30:27 ago	capricorne-35.lyon.grid5000.fr load_one: down Last heartbeat 1 day, 20:34:34 ago	capricorne-51.lyon.grid5000.fr load_one: down Last heartbeat 40 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 12 days, 22:30:43 ago	capricorne-1.lyon.grid5000.fr load_one: unknown Last seen 0 days, 12:30:05 ago	capricorne-10.lyon.grid5000.fr load_one: unknown Last seen 0 days, 12:30:07 ago
capricorne-11.lyon.grid5000.fr load_one: unknown Last seen 0 days, 12:30:00 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, 12:30:05 ago	capricorne-14.lyon.grid5000.fr load_one: unknown Last seen 0 days, 12:19:59 ago

Usage per month per cluster



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

Bordeaux:

- Alexandre Denis (Researcher (CR)), Runtime LaBRI Bordeaux (2007-07-28 22:45:35)
- Kristian Koehler (Master student), runtime inria bordeaux - sud ouest Bordeaux (2008-04-30 12:57:37)
- François Trahay (PhD student), Runtime LaBRI Bordeaux (2008-04-03 14:31:37)
- Stéphanie Moreaud (PhD student), Runtime LaBRI Bordeaux (2007-12-29 18:58:40)
- Olivier Teytaud (Researcher (CRI)), TAO (Inria Futurs) LRI (Cnrs, Inria, Univ Paris-Sud) Orsay (France) (2007-12-24 00:28:20)
- Pierre Ramet (Lecturer/Associate Professor (MCF)), SciAppix LaBRI Bordeaux (2007-12-18 10:05:43)
- Pascal Henon (Researcher (CR)), SciAppix (INRIA) LaBRI Bordeaux (2007-12-19 10:28:00)
- Adrien Goeffron (Post-Doc), MAGNOME LaBRI Bordeaux (2007-12-04 11:55:03)
- David Sherman (Researcher (CRI)), Magpome INRIA Futurs Bordeaux (2007-08-03 12:05:04)
- brice goglin (Engineer), LaBRI Bordeaux (2007-04-05 14:42:57)
- Nicolas Bonichon (Lecturer/Associate Professor (MCF)), Cepage LaBRI Bordeaux (2007-03-20 18:17:27)
- Nathalie Furmento (Engineer), Runtime LaBRI Bordeaux (2007-04-26 10:54:27)
- Olivier Aumage (Researcher (CRI)), Runtime LaBRI Bordeaux (2007-02-26 12:38:30)
- Elisabeth Brunet (PhD student), Runtime LaBRI Bordeaux (2007-04-24 12:07:01)
- Brice Goglin (Researcher (CR)), Runtime LaBRI Bordeaux (2007-03-24 11:28:01)
- Christophe Frezzer (Engineer), Runtime LaBRI Bordeaux (2007-02-22 17:15:20)
- Nicolas Richard (PhD student), SciAppix LaBRI Bordeaux (2007-04-22 17:30:00)
- Michael Raynaud (Engineer), Ipari LaBRI / Inria Futurs Bordeaux (2007-04-21 17:20:29)
- Aurelien Ennard (Lecturer/Associate Professor (MCF)), SciAppix LaBRI Bordeaux (2007-04-22 17:08:10)
- Guilhem Caramel (Engineer), SciAppix LaBRI Bordeaux (2007-04-22 17:01:40)
- Mathieu Souchaud (Engineer), SciAppix LaBRI Bordeaux (2007-04-21 17:00:44)
- Frank Prati (Post-Doc), Magpome LMA Pau (2007-04-04 14:18:50)
- Samuel Thibault (PhD student), Runtime LaBRI Bordeaux (2007-03-04 12:47:20)
- Mathieu Bernatet (Master student), ANR LEGONUMASIS LaBRI Bordeaux (2007-04-03 20:07:02)
- Stephane Blanchard (Master student), ANR LEGONUMASIS LaBRI Bordeaux (2007-04-04 20:04:30)
- Olivier Coulaud (Senior researcher (DR)), SciAppix Inria Futurs Bordeaux (2007-04-04 16:56:20)
- François Broquedis (Master student), LEGONUMASIS LaBRI Bordeaux (2007-04-03 16:26:23)
- Jérôme Clet-Ortega (Master student), ANR LEGO LaBRI Bordeaux (2007-04-03 16:32:12)
- Guillaume Anclaux (PhD student), SciAppix LaBRI Bordeaux (2007-04-04 16:18:35)

(29 reports, 24 experiments, 11 publications, 4 collaborations, 40 users)

Grenoble:

- Alexander Klaser (PhD student), LEAR UK Grenoble (2008-07-23 12:09:20)
- Pierre-François Dutot (Lecturer/Associate Professor (MCF)), MOAIS UG Grenoble (2008-07-22 10:55:45)
- Xavier Besseron (PhD student), MOAIS UG Grenoble (2008-07-17 16:33:50)
- Sami Achour (PhD student), MOAIS UG Grenoble (2008-07-16 06:40)
- Lucas Schnorr (PhD student), MOAIS UG Mombonnott (2008-06-23 10:54:20)
- Krzysztof Rzaadka (PhD student), MOAIS UG (2008-06-22 16:38:37)
- Fredéric Bouquet (Master student), UG MESCAL (2008-05-27 14:15:41)

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

User information

Thomas Ropars (PhD student)
Paris RISA Rennes, France (Rennes)
Email address: tropars@risa.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 experiment, Vigne is deployed on all the nodes and applications are submitted. Failures are simulated with kill signals sent to some applications. Through this experiment, 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 Morn, 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: technreport
Author: Ropars, Thomas and Jeanvoine, Emmanuel and Morn, Christine
Number: RR-6070
Address: RISA, Rennes, France
Type: Research Report
Institution: RISA/Paris Research group, Université de Rennes 1, EDF R&D, INRIA
Url: <http://hal.inria.fr/inria-00121059>

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>
- Now part of the Grid5000-code project on INRIA's forge.

g5k-campaign

A tentative to abstract common phases in an experiment by Cyril Rohr

Principles

- use an *engine* to declare what needs to be done before/at/after each phase
- engines can be shared and can be stacked/composed
- Will be used during the API practical session
- g5k-campaign.gforge.inria.fr

Discover ressources available at a given time

- disco is a simple resource discovery tool for oargrid to find the maximum available resources on a given time range for specified alias(s) of resources.
- Find available resources from now to now + 1 hour on paraquad cluster (located at Rennes):

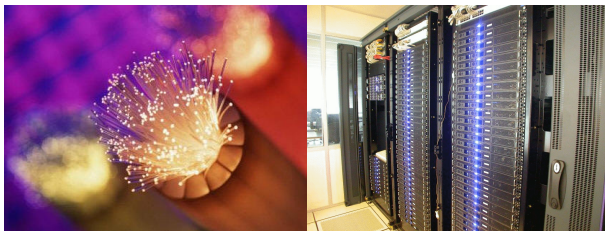
```
disco paraquad
```

- Find available resources from now to now + 1 hour on nancy lyon and rennes sites:

```
disco nancy lyon rennes
```

Planned evolutions

Hardware evolutions



- Renewal of machines/clusters (depending on fundings)
- New equipments (network probes / power consumption probes)
- new sites: Luxembourg, Reims and Porte-Alegre (Brazil)



The development team is working on several aspects

- Disk space management
- Experiment campaigns
- KVM on the production image on nodes
- User Management and integration between experiments and users
- grid5000-code: better sharing of code between users
- Network control: Monitoring/isolation and bandwidth on demand

- Seminars
- Newsletter
- Promotion of the contribution to the platform (tools developed by users that can be used by other)
- Tutorial sessions
- Spring school, now in Reims

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 tutorial pages:
<https://www.grid5000.fr/mediawiki/index.php/Category:Portal:Tutorial>
- 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>

