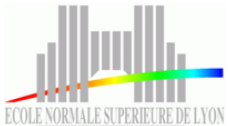
June 14, 2007

DIET Dashboard & GRUDU: software for DIET on Grid'5000

David Loureiro

INRIA/LIP/GRAAL

June 14, 2007




Outline

Outline

Outline

Outline

Outline

Introduction

Introduction

Context

Introduction

Context

- The GRAAL team from the LIP develops DIET (Distributed Interactive Engineering Toolbox), a set of elements that can be used to build applications using the GridRPC paradigm.

Introduction

Context

- The GRAAL team from the LIP develops DIET (Distributed Interactive Engineering Toolbox), a set of elements that can be used to build applications using the GridRPC paradigm.
- Grid'5000 is a research effort developping a large scale nation wide infrastructure for Grid research.

Introduction

Context

- The GRAAL team from the LIP develops DIET (Distributed Interactive Engineering Toolbox), a set of elements that can be used to build applications using the GridRPC paradigm.
- Grid'5000 is a research effort developping a large scale nation wide infrastructure for Grid research.

My work

Introduction

Context

- The GRAAL team from the LIP develops DIET (Distributed Interactive Engineering Toolbox), a set of elements that can be used to build applications using the GridRPC paradigm.
- Grid'5000 is a research effort developping a large scale nation wide infrastructure for Grid research.

My work

- Develop tools to ease the use of DIET on Grid'5000: DIET Dashboard & GRUDU

Introduction

Context

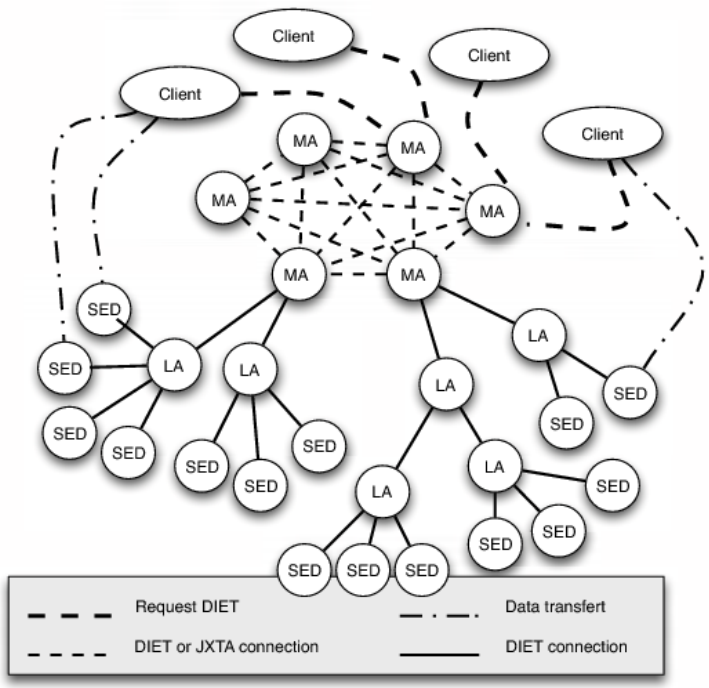
- The GRAAL team from the LIP develops DIET (Distributed Interactive Engineering Toolbox), a set of elements that can be used to build applications using the GridRPC paradigm.
- Grid'5000 is a research effort developping a large scale nation wide infrastructure for Grid research.

My work

- Develop tools to ease the use of DIET on Grid'5000: DIET Dashboard & GRUDU
- Validate DIET, DIET Dashboard and GRUDU on Grid'5000 through numerous large experiments

DIET short description

- Aim : build computational servers
- Huge problems can be computed thanks to Grid Computing Environments
- Able to find an appropriate server according to the information given in the client's request, etc



DIET Dashboard presentation

- In Grid Environments, users need several and complex tools for the management of resources and client/server applications.

DIET Dashboard presentation

- In Grid Environments, users need several and complex tools for the management of resources and client/server applications.
- Drawback: Most grid software use command line interfaces and do not provide GUI or only partially.

DIET Dashboard presentation

- In Grid Environments, users need several and complex tools for the management of resources and client/server applications.
- Drawback: Most grid software use command line interfaces and do not provide GUI or only partially.

DIET Dashboard presentation

- In Grid Environments, users need several and complex tools for the management of resources and client/server applications.
- Drawback: Most grid software use command line interfaces and do not provide GUI or only partially.

DIET Dashboard

DIET Dashboard provides such an interface.

DIET Dashboard presentation

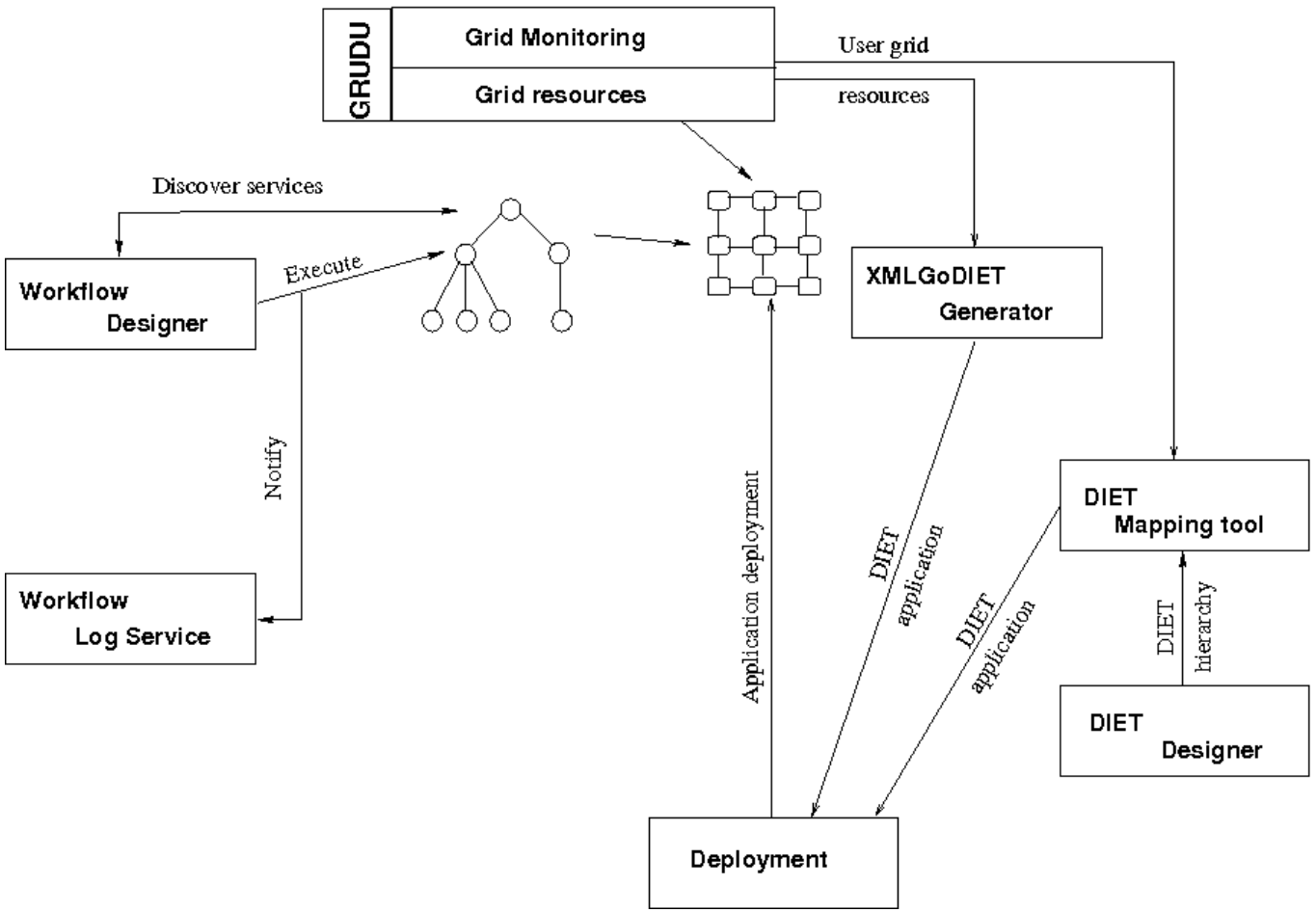
- In Grid Environments, users need several and complex tools for the management of resources and client/server applications.
- Drawback: Most grid software use command line interfaces and do not provide GUI or only partially.

DIET Dashboard

DIET Dashboard provides such an interface.

It is a set of tools written in Java, that provides to the DIET end-user, friendly-user interfaces to design, deploy, monitor the execution of client/server applications and for the allocation of resources on Grid'5000.

DIET Dashboard architecture



The DIET tools of the deployment in DIET Dashboard

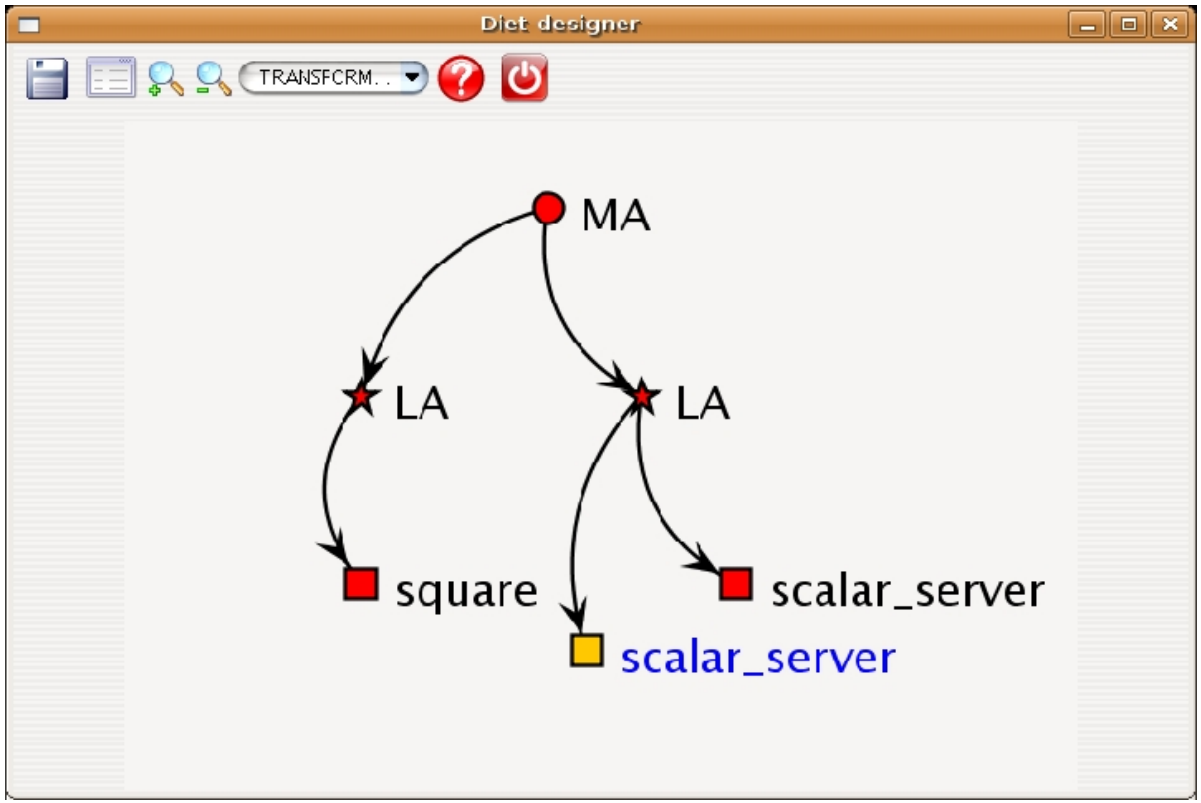
The DIET tools of the deployment in DIET Dashboard

For the deployment of a DIET hierarchy :

The DIET tools of the deployment in DIET Dashboard

For the deployment of a DIET hierarchy :

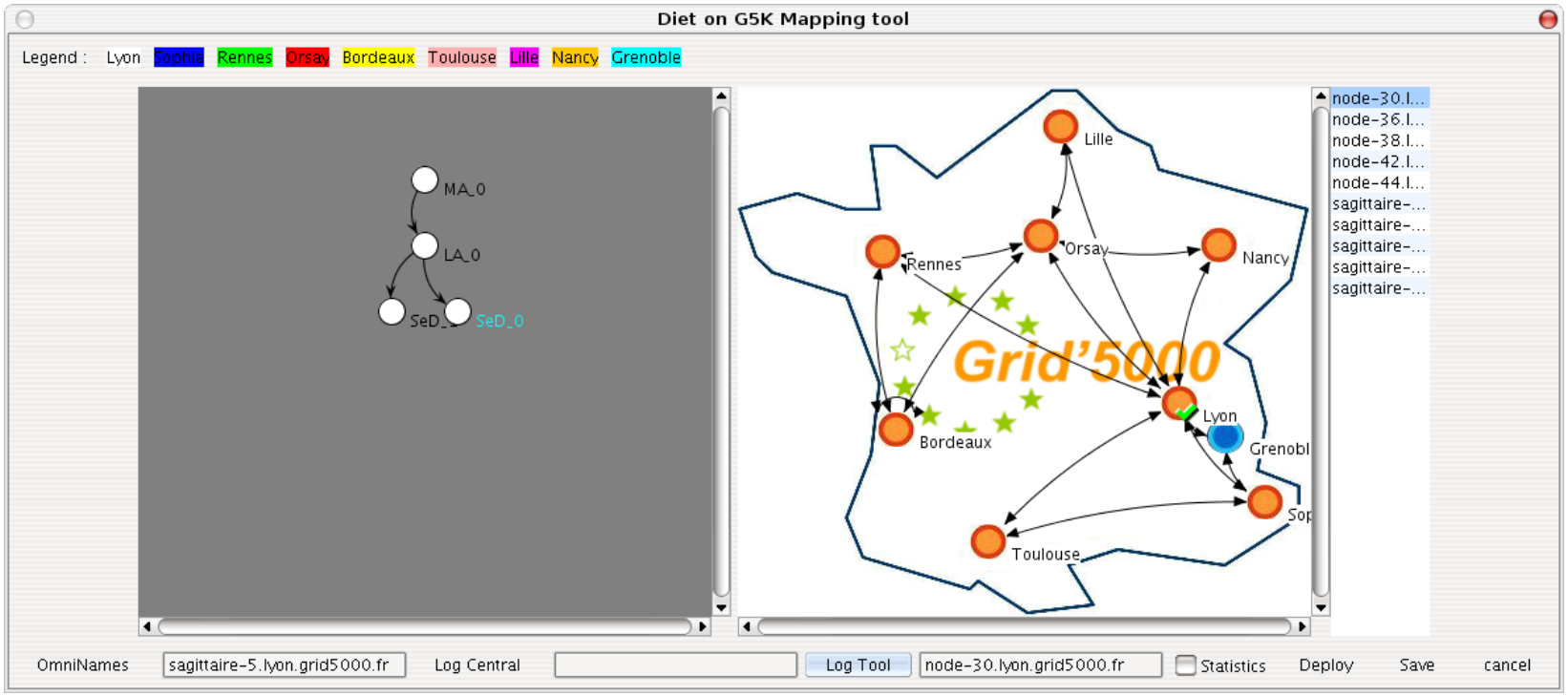
- DIET Designer → DIET Mapping Tool → DIET Deploy Tool (GoDIET)



The DIET tools of the deployment in DIET Dashboard

For the deployment of a DIET hierarchy :

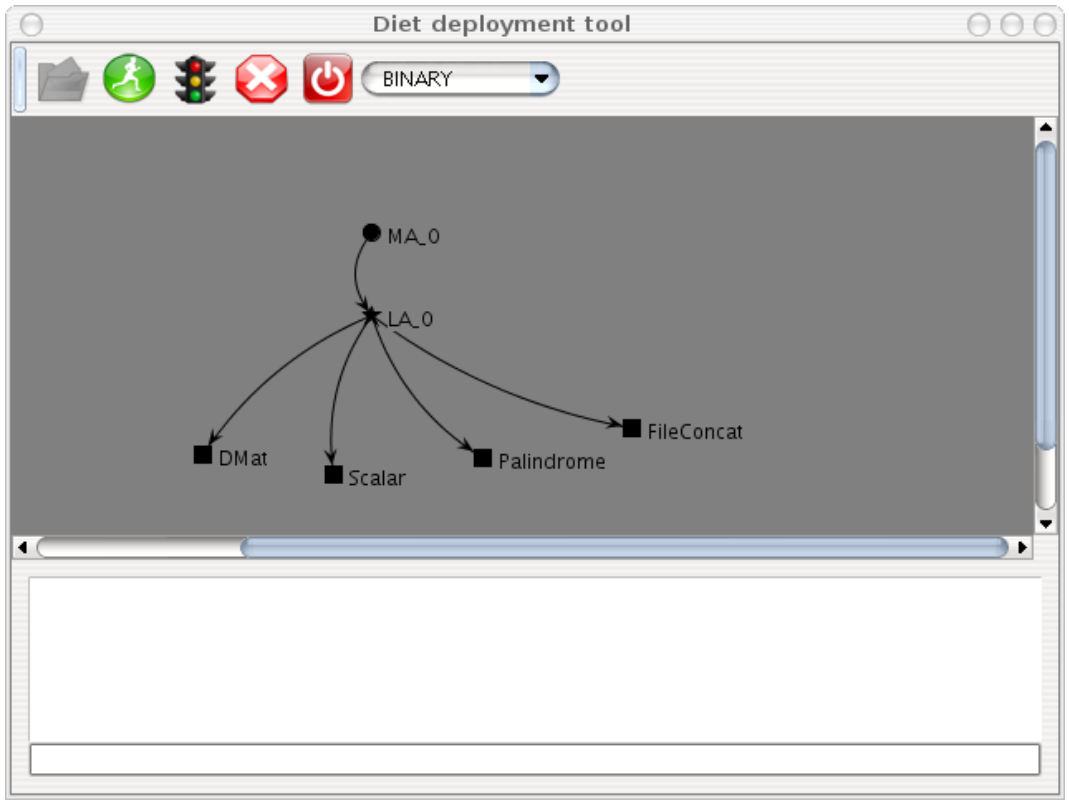
- DIET Designer → DIET Mapping Tool → DIET Deploy Tool (GoDIET)



The DIET tools of the deployment in DIET Dashboard

For the deployment of a DIET hierarchy :

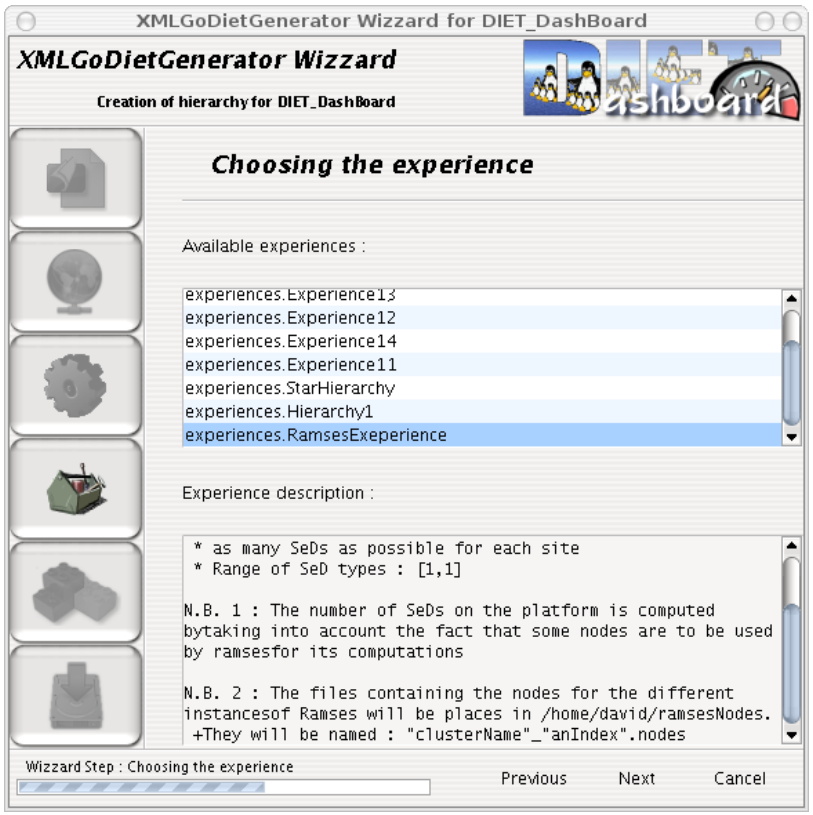
- DIET Designer → DIET Mapping Tool → DIET Deploy Tool (GoDIET)



The DIET tools of the deployment in DIET Dashboard

For the deployment of a DIET hierarchy :

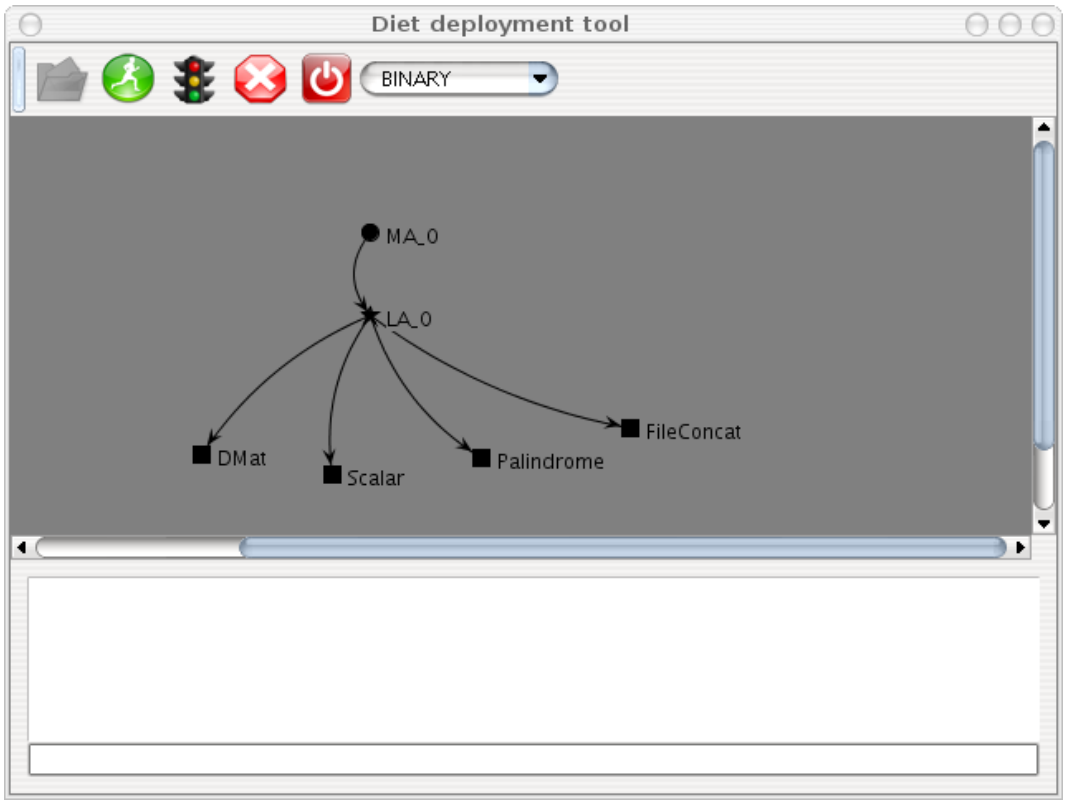
- DIET Designer → DIET Mapping Tool → DIET Deploy Tool (GoDIET)
- XMLGoDIETGenerator → DIET Deploy Tool (GoDIET)



The DIET tools of the deployment in DIET Dashboard

For the deployment of a DIET hierarchy :

- DIET Designer → DIET Mapping Tool → DIET Deploy Tool (GoDIET)
- XMLGoDIETGenerator → DIET Deploy Tool (GoDIET)



Deployment with GoDIET

- GoDIET deploys and runs the different elements of the hierarchy on remote machines

Deployment with GoDIET

- GoDIET deploys and runs the different elements of the hierarchy on remote machines
- Based on a XML file describing

Deployment with GoDIET

- GoDIET deploys and runs the different elements of the hierarchy on remote machines
- Based on a XML file describing
  - Resources: clusters, nodes

Deployment with GoDIET

- GoDIET deploys and runs the different elements of the hierarchy on remote machines
- Based on a XML file describing
  - Resources: clusters, nodes
  - Services: naming service, Log Tool, etc ...

Deployment with GoDIET

- GoDIET deploys and runs the different elements of the hierarchy on remote machines
- Based on a XML file describing
  - Resources: clusters, nodes
  - Services: naming service, Log Tool, etc ...
  - Hierarchy: Master Agent, Local Agent(s), SeD(s) and their config

Deployment with GoDIET

- GoDIET deploys and runs the different elements of the hierarchy on remote machines
- Based on a XML file describing
  - Resources: clusters, nodes
  - Services: naming service, Log Tool, etc ...
  - Hierarchy: Master Agent, Local Agent(s), SeD(s) and their config
- Integrated in DIET Dashboard

XMLGoDIETGenerator

- Time-consuming to write large GoDIET files for large hierarchies

XMLGoDIETGenerator

- Time-consuming to write large GoDIET files for large hierarchies
- GoDIET files are resources-dependent

XMLGoDIETGenerator

- Time-consuming to write large GoDIET files for large hierarchies
- GoDIET files are resources-dependent
- **XMLGoDIETGenerator** is resources-driven

XMLGoDIETGenerator

- Time-consuming to write large GoDIET files for large hierarchies
- GoDIET files are resources-dependent
- **XMLGoDIETGenerator** is resources-driven
- Based on frameworks of experiments representing usual patterns (Star hierarchy, One-level hierarchy)

XMLGoDIETGenerator

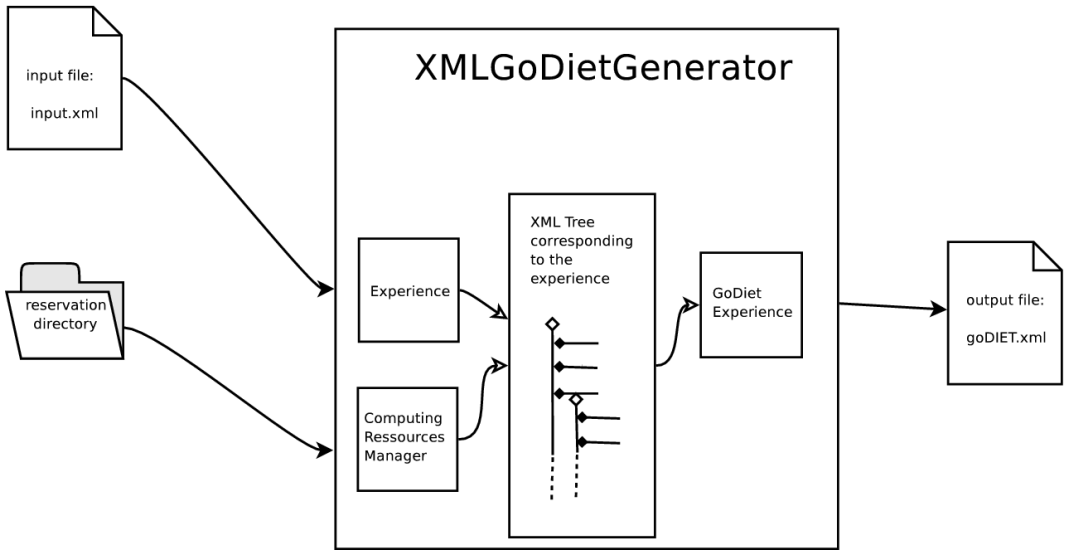
- Time-consuming to write large GoDIET files for large hierarchies
- GoDIET files are resources-dependent
- **XMLGoDIETGenerator** is resources-driven
- Based on frameworks of experiments representing usual patterns (Star hierarchy, One-level hierarchy)
- Written in Java: the users can supply their own framework classes at execution time

XMLGoDIETGenerator

- Time-consuming to write large GoDIET files for large hierarchies
- GoDIET files are resources-dependent
- **XMLGoDIETGenerator** is resources-driven
- Based on frameworks of experiments representing usual patterns (Star hierarchy, One-level hierarchy)
- Written in Java: the users can supply their own framework classes at execution time

XMLGoDIETGenerator

- Time-consuming to write large GoDIET files for large hierarchies
- GoDIET files are resources-dependent
- **XMLGoDIETGenerator** is resources-driven
- Based on frameworks of experiments representing usual patterns (Star hierarchy, One-level hierarchy)
- Written in Java: the users can supply their own framework classes at execution time



Presentation

- Tool designed to manage user grid resources

Presentation

- Tool designed to manage user grid resources
- Currently dedicated to Grid'5000 and OAR

Presentation

- Tool designed to manage user grid resources
- Currently dedicated to Grid'5000 and OAR
- Displays the status of Grid'5000 with different levels of granularity: grid, cluster, node, job

Presentation

- Tool designed to manage user grid resources
- Currently dedicated to Grid'5000 and OAR
- Displays the status of Grid'5000 with different levels of granularity: grid, cluster, node, job
- Realizes the reservation of resources

Presentation

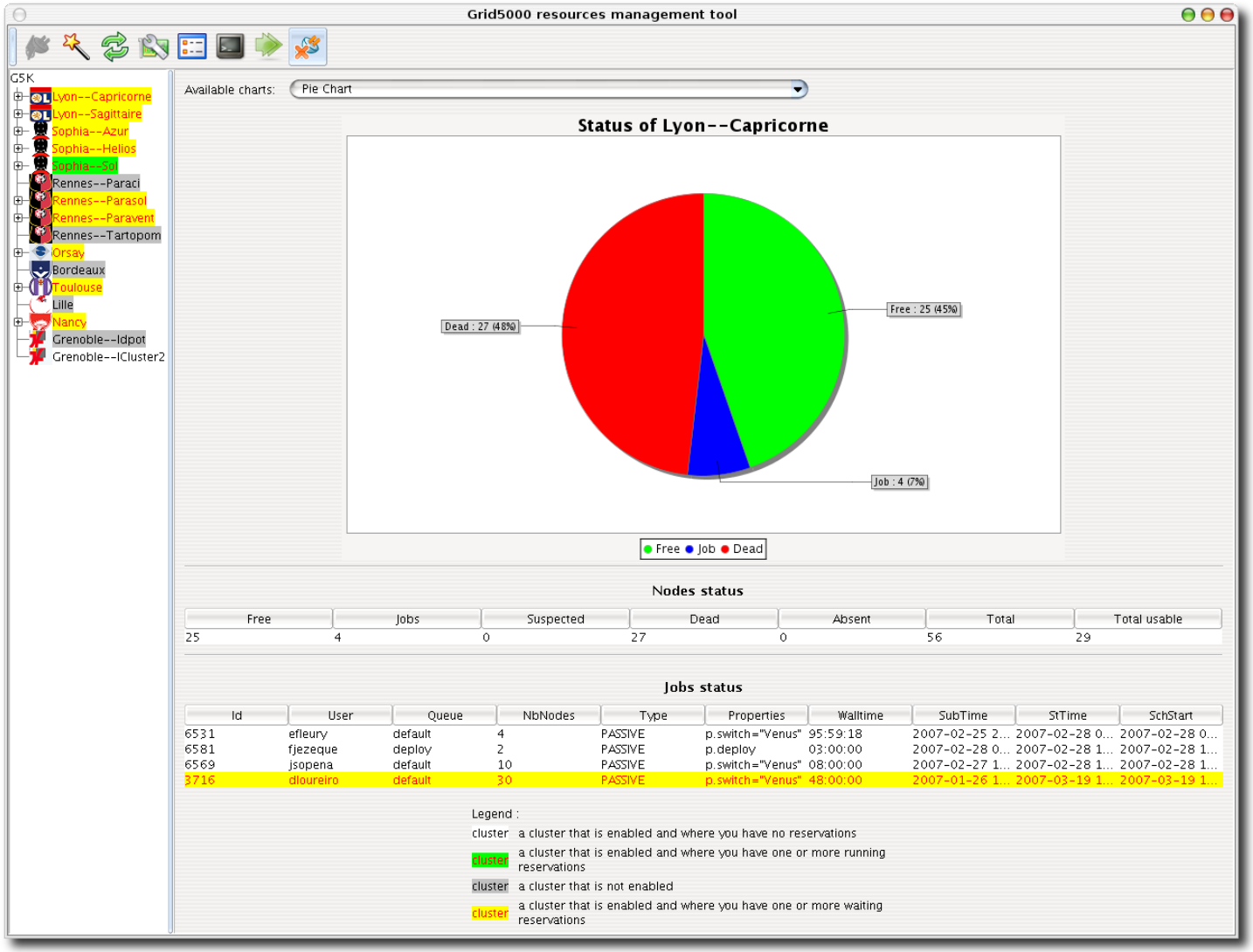
- Tool designed to manage user grid resources
- Currently dedicated to Grid'5000 and OAR
- Displays the status of Grid'5000 with different levels of granularity: grid, cluster, node, job
- Realizes the reservation of resources
- Allows the deployment of images through KaDeploy

Presentation

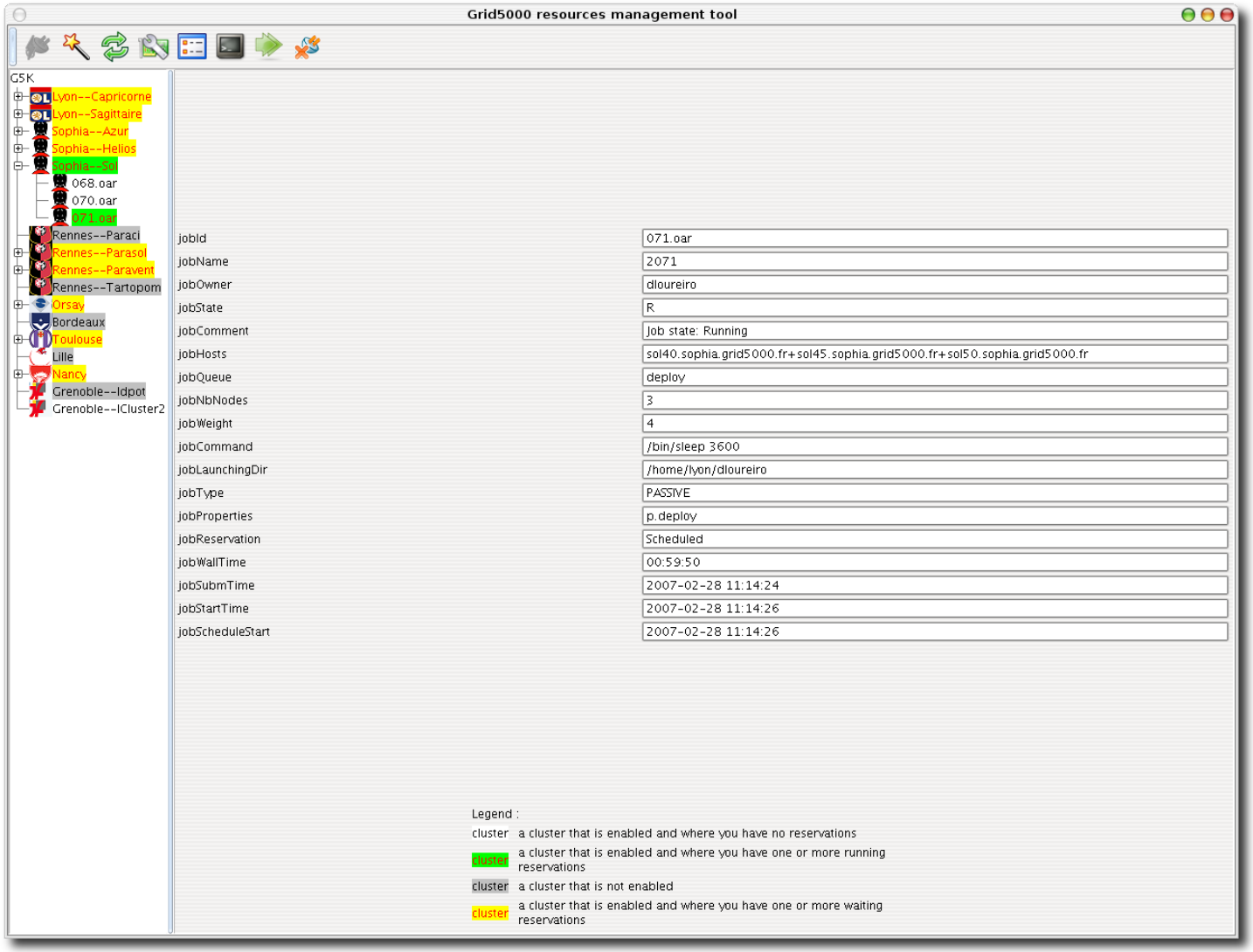
- Tool designed to manage user grid resources
- Currently dedicated to Grid'5000 and OAR
- Displays the status of Grid'5000 with different levels of granularity: grid, cluster, node, job
- Realizes the reservation of resources
- Allows the deployment of images through KaDeploy
- Offers terminal access on clusters, reserved nodes

Grid'5000 status

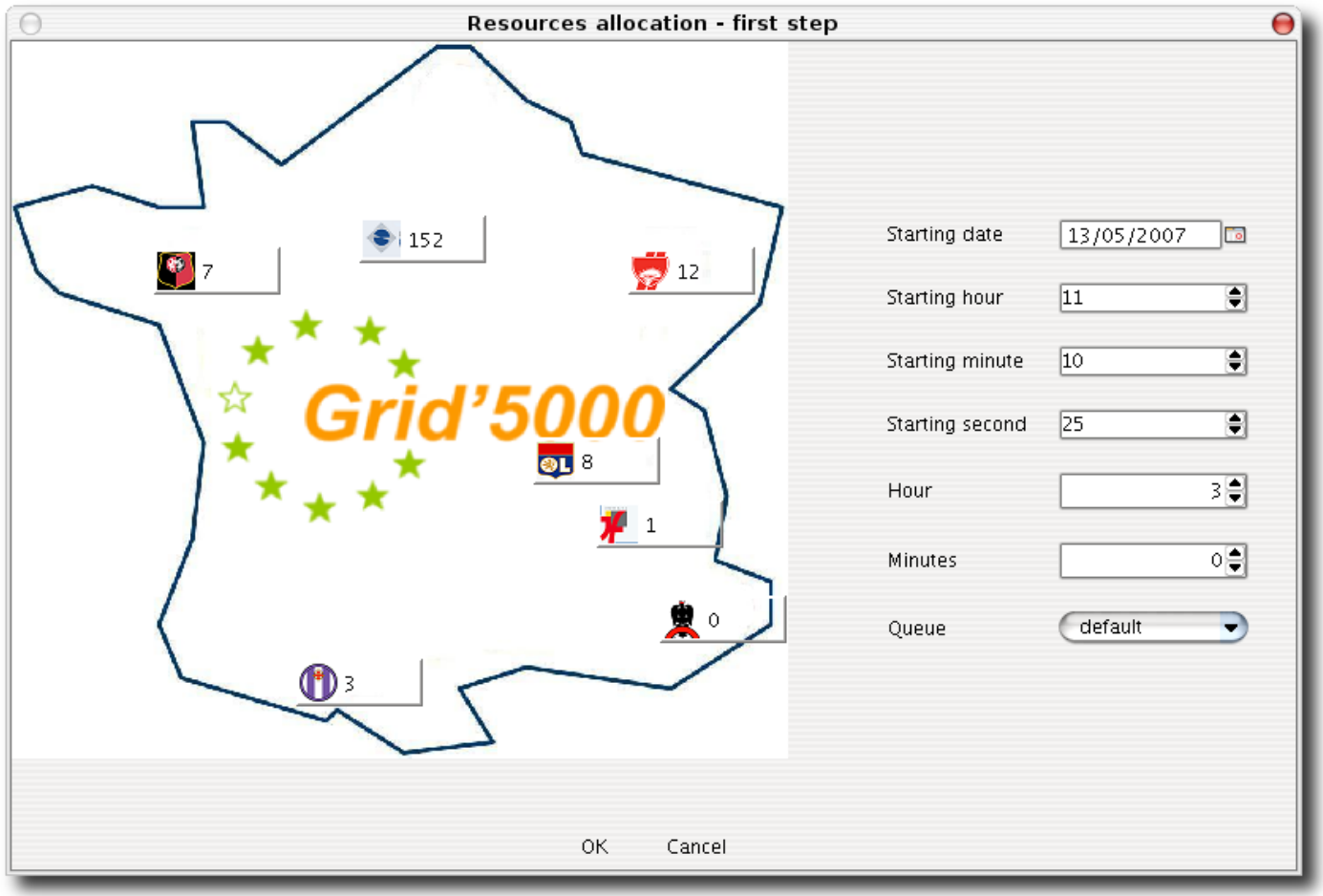
Grid'5000 status



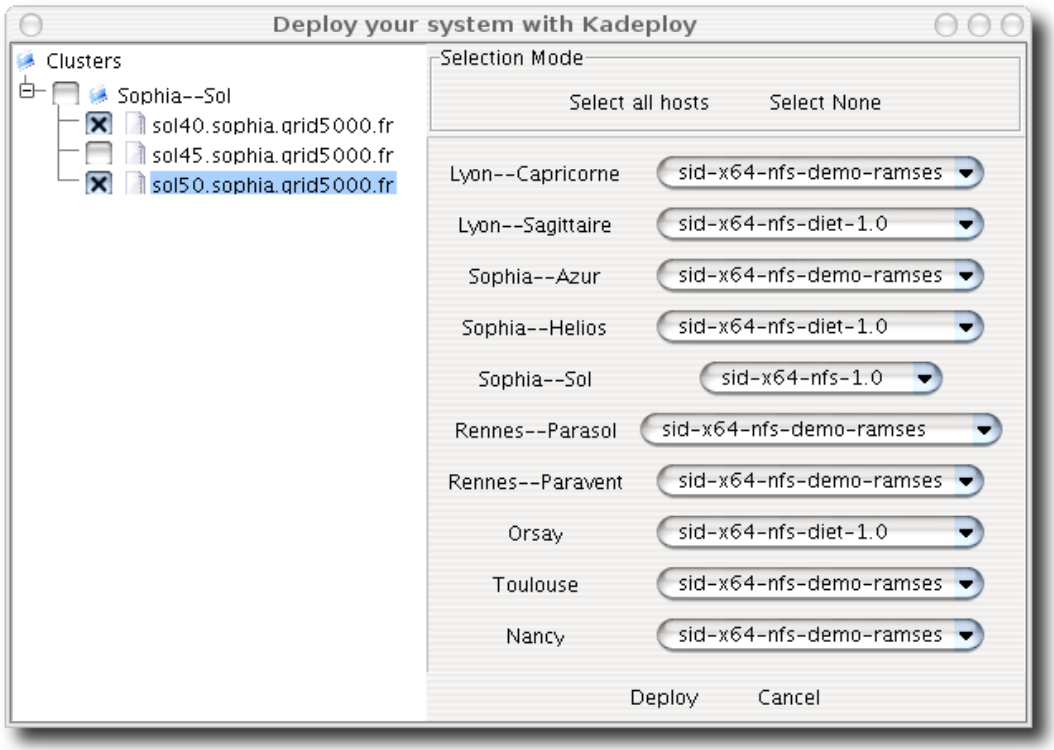
Grid'5000 status



Resources allocation and images deployment



Resources allocation and images deployment



Experiments on Grid'5000 with DIET Dashboard & GRUDU

Experiments on Grid'5000 with DIET Dashboard & GRUDU

Goal

Test the capabilities of DIET, DIET Dashboard and GRUDU for a large number of machines in real life (Cosmological computations).

Experiments on Grid'5000 with DIET Dashboard & GRUDU

Goal

Test the capabilities of DIET, DIET Dashboard and GRUDU for a large number of machines in real life (Cosmological computations).

- The largest reservation on KaDeploy queue

Experiments on Grid'5000 with DIET Dashboard & GRUDU

Goal

Test the capabilities of DIET, DIET Dashboard and GRUDU for a large number of machines in real life (Cosmological computations).

- The largest reservation on KaDeploy queue
- The largest DIET hierarchy for the maximum number of cosmological application jobs

Experiments on Grid'5000 with DIET Dashboard & GRUDU

Goal

Test the capabilities of DIET, DIET Dashboard and GRUDU for a large number of machines in real life (Cosmological computations).

- The largest reservation on KaDeploy queue
- The largest DIET hierarchy for the maximum number of cosmological application jobs
- The code executed on each server is RAMSES: a grid-based hydro solver for the study of large scale structures and galaxies formation developed in Saclay (DAPNI/CEA)

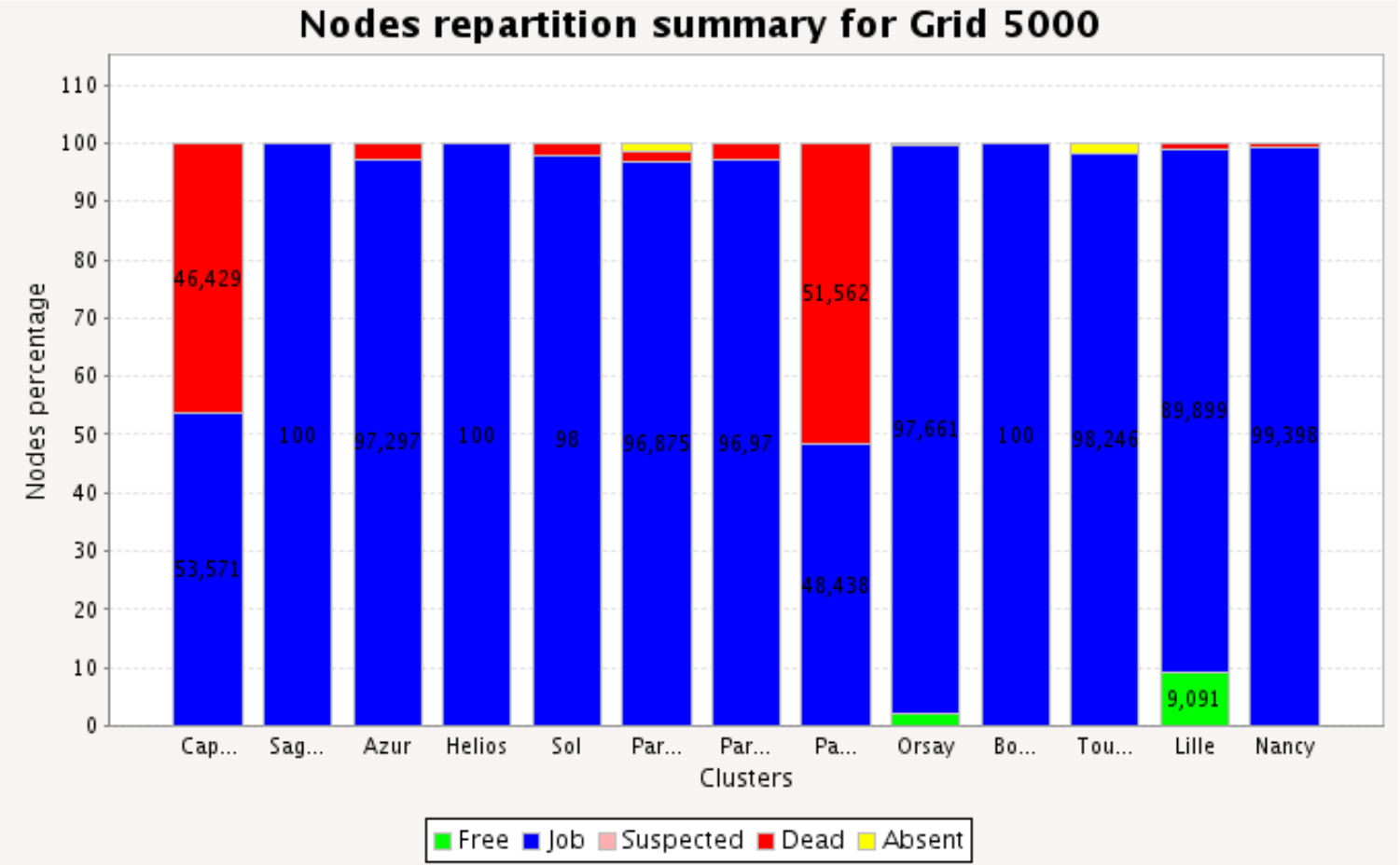
Resources for the experiment 1/3

979 machines reserved with a deployed environment using Kadeploy during 48 hours.

Details: 12 clusters on 7 sites during 48 hours

- Rennes : 189 machines
- Orsay : 303 machines
- Nancy : 46 machines
- Bordeaux : 99 machines
- Lyon : 99 machines
- Toulouse : 56 machines
- Sophia : 187 machines

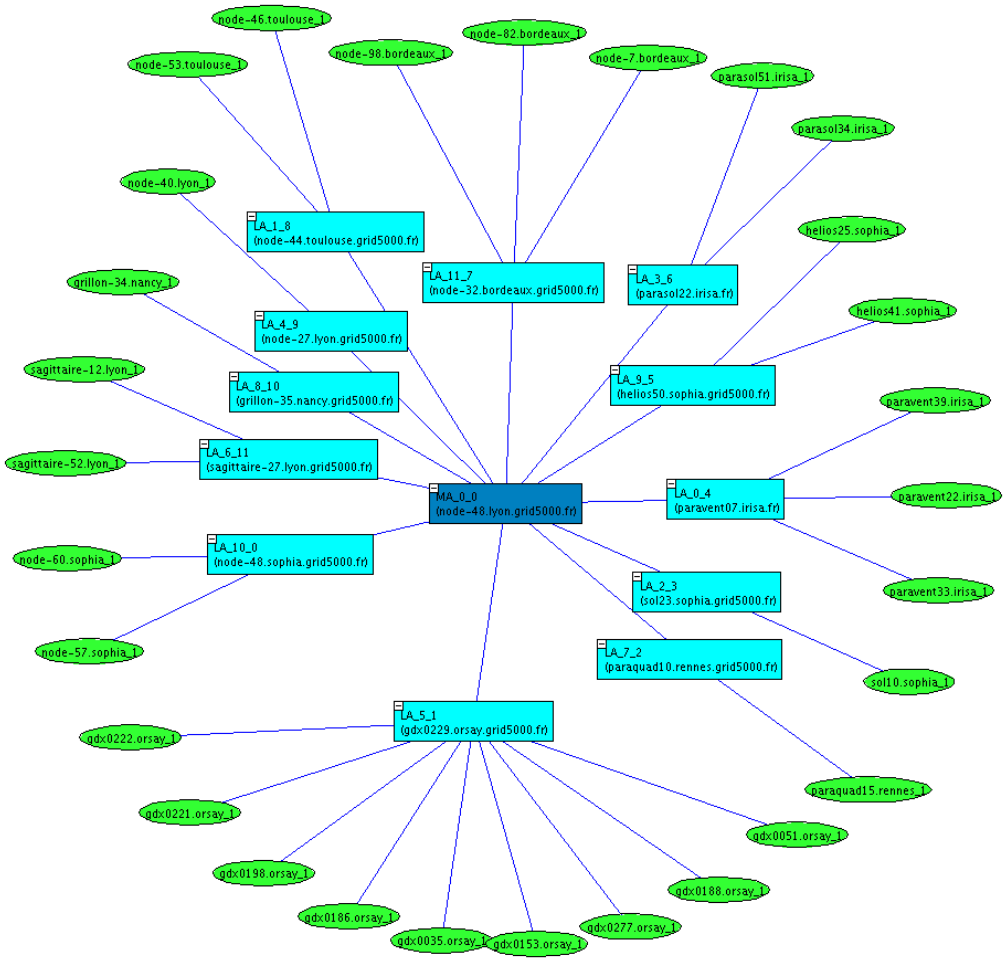
Resources for the experiment 2/3



Resources for the experiment 3/3

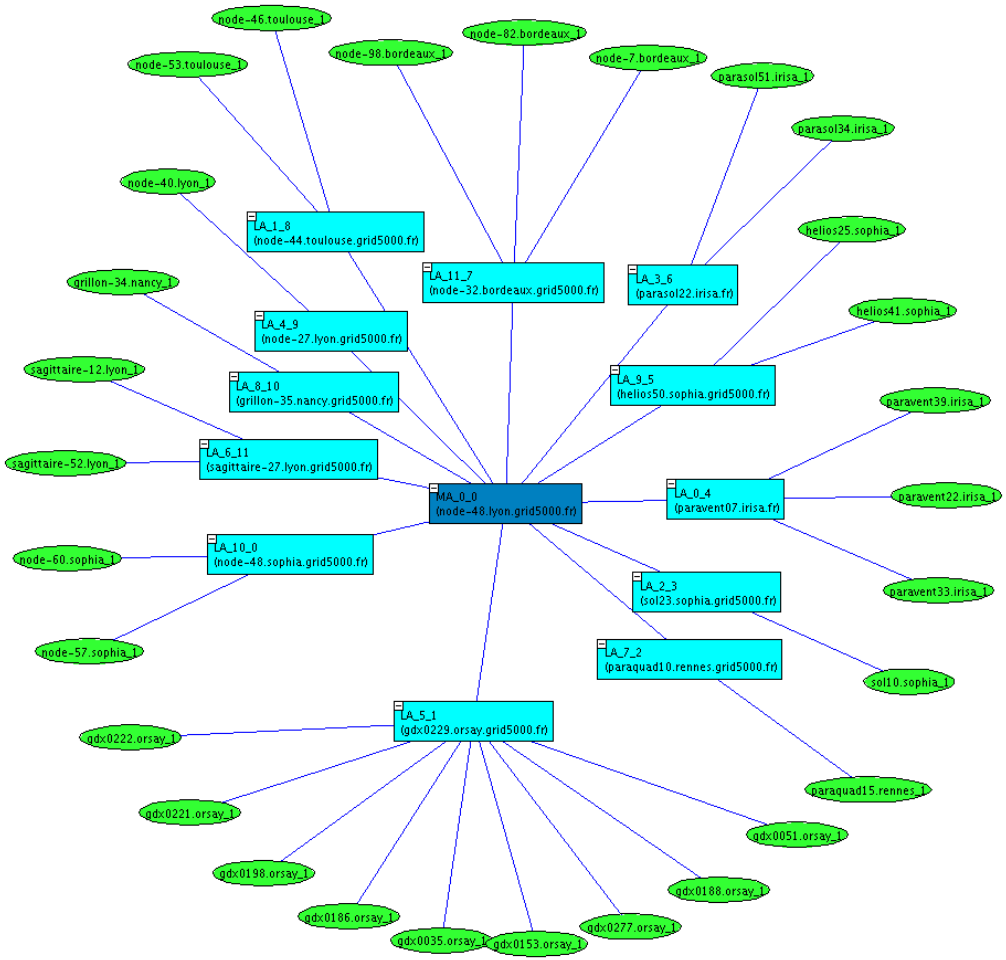
DIET hierarchy

- 1 Master Agent



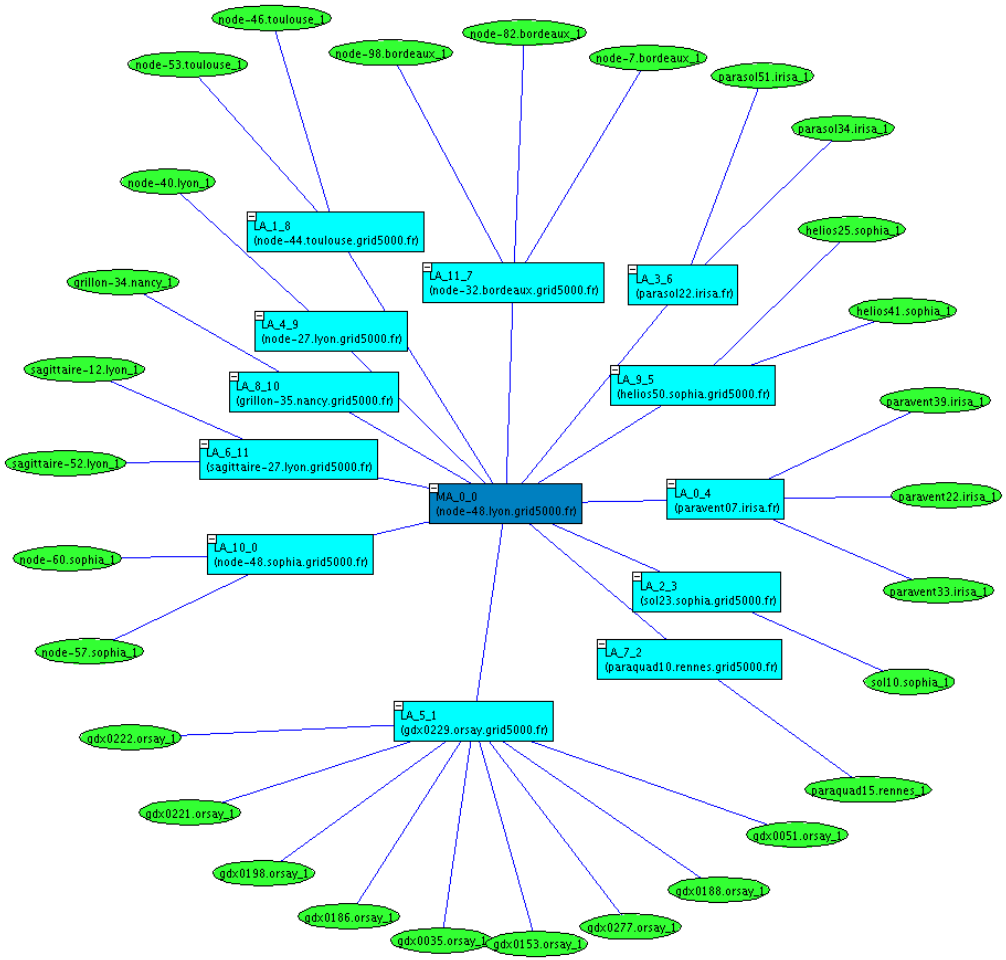
DIET hierarchy

- 1 Master Agent
- 12 Local Agent



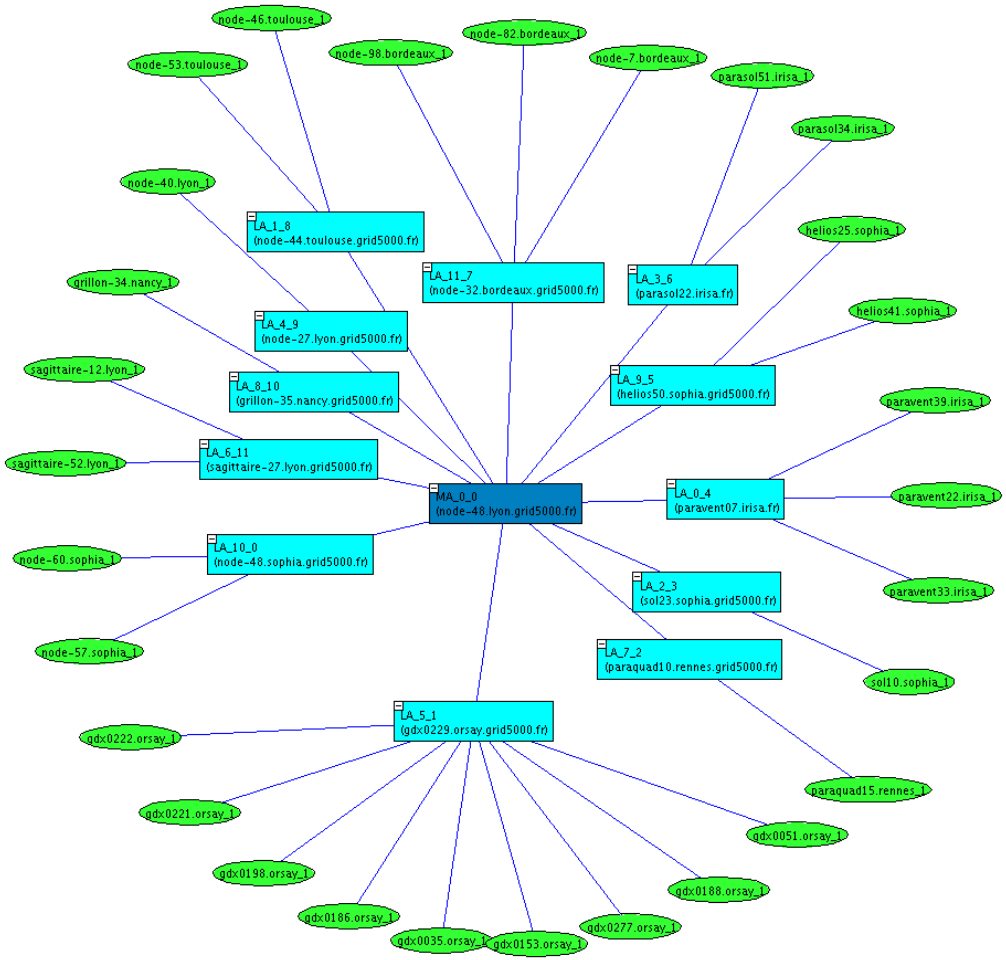
DIET hierarchy

- 1 Master Agent
- 12 Local Agent
- 29 Server Deamons



DIET hierarchy

- 1 Master Agent
- 12 Local Agent
- 29 Server Deamons
- **816 nodes dedicated to Ramses**



Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites
- NFS disk usage caused application failures

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites
- NFS disk usage caused application failures

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites
- NFS disk usage caused application failures

The experiment led to 59 simulations with 33 fully computed and 26 partial realized

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites
- NFS disk usage caused application failures

The experiment led to 59 simulations with 33 fully computed and 26 partial realized

	DAPNIA/CEA	Grid'5000
Simulation time (h)	4464 (6 months)	25

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites
- NFS disk usage caused application failures

The experiment led to 59 simulations with 33 fully computed and 26 partial realized

	DAPNIA/CEA	Grid'5000
Simulation time (h)	4464 (6 months)	25
Processors	32	1824

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites
- NFS disk usage caused application failures

The experiment led to 59 simulations with 33 fully computed and 26 partial realized

	DAPNIA/CEA	Grid'5000
Simulation time (h)	4464 (6 months)	25
Processors	32	1824
Complete simulations	50	33

Results

Different problems appear during the experiment

- Some clusters unavailable for KaDeploy
- Wrong or different KaDeploy configurations on some sites
- NFS disk usage caused application failures

The experiment led to 59 simulations with 33 fully computed and 26 partial realized

	DAPNIA/CEA	Grid'5000
Simulation time (h)	4464 (6 months)	25
Processors	32	1824
Complete simulations	50	33
Simulation speed (simu/h)	0.0112	1.18

What has been done

- Deploy DIET hierarchies on Grid'5000

What has been done

- Deploy DIET hierarchies on Grid'5000
- New functionalities in DIET Dashboard and forking GRUDU

What has been done

- Deploy DIET hierarchies on Grid'5000
- New functionalities in DIET Dashboard and forking GRUDU
- User's support on DIET

What has been done

- Deploy DIET hierarchies on Grid'5000
- New functionalities in DIET Dashboard and forking GRUDU
- User's support on DIET
- Numerous large and real life experiments

What has been done

- Deploy DIET hierarchies on Grid'5000
- New functionalities in DIET Dashboard and forking GRUDU
- User's support on DIET
- Numerous large and real life experiments

What has been done

- Deploy DIET hierarchies on Grid'5000
- New functionalities in DIET Dashboard and forking GRUDU
- User's support on DIET
- Numerous large and real life experiments

Links :

DIET `http://graal.ens-lyon.fr/DIET`

What has been done

- Deploy DIET hierarchies on Grid'5000
- New functionalities in DIET Dashboard and forking GRUDU
- User's support on DIET
- Numerous large and real life experiments

Links :

DIET `http://graal.ens-lyon.fr/DIET`

Grid'5000 `https://www.grid5000.fr`

Future work

- Improvements of DIET Dashboard and GRUDU

Future work

- Improvements of DIET Dashboard and GRUDU
- Managing mailing lists, bug reports, feature-requests for GRUDU

Future work

- Improvements of DIET Dashboard and GRUDU
- Managing mailing lists, bug reports, feature-requests for GRUDU
- Developments in DIET for its deployment on the Decrypthon grid

Future work

- Improvements of DIET Dashboard and GRUDU
- Managing mailing lists, bug reports, feature-requests for GRUDU
- Developments in DIET for its deployment on the Decrypthon grid
- Realizing new tests for the benchmarking of DIET