CNRS - INP - UT3 - UT1 - UT2J

Institut de Recherche en Informatique de Toulouse

slices FR slices RI

FRANCE 2030 PROGRAMME DE RECHERCHE NUMÉRIQUE POUR L'EXASCALE

Grid5000@IRIT

Overview and demo of Grid5000 platform: from CPU/GPU/ARM/… computing to low-level experimentation

Georges Da Costa

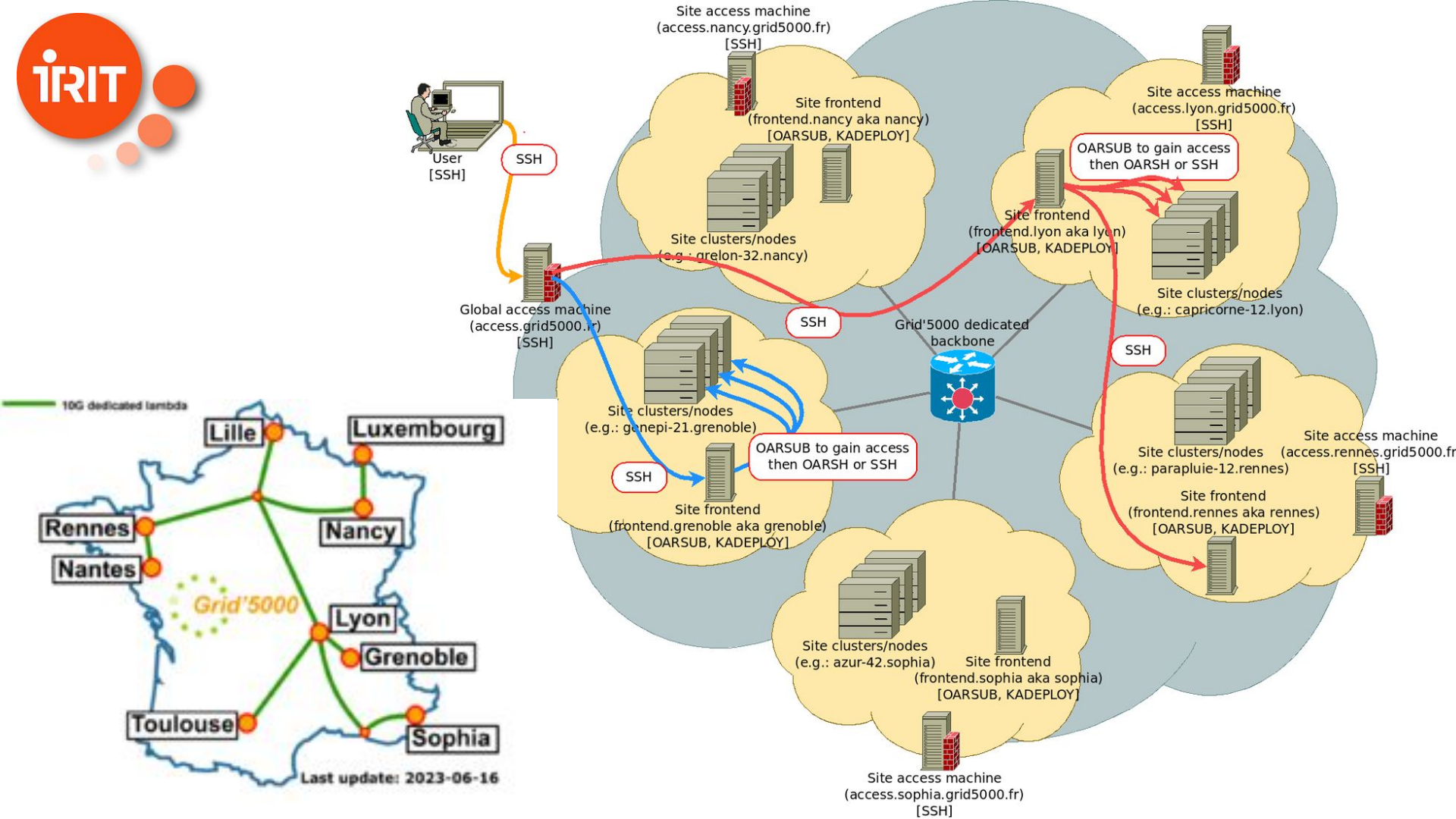
What is Grid5000?

Academic research infrastructure for computer science, not for production

- Large amount of resources: 15000 cores, 800 compute-nodes
  - PMEM, GPU, SSD, NVMe, 10G and 25G Ethernet, Infiniband, Omni-Path
  - https://www.grid5000.fr/w/Hardware
- Highly reconfigurable and controllable
  - Direct utilization down to reboot on your own OS
  - Network reconfiguration
- Monitoring (from system to network and wattmeters)

Cost:

- Acknowledgement in publications


Site access machine
(access.nancy.grid5000.fr)
[SSH]
Site frontend
(frontend.nancy aka nancy)
[OARSUB, KADEPLOY]
Site clusters/nodes
(e.g.: grelon-32.nancy)
User
[SSH]
SSH
Global access machine
(access.grid5000.fr)
[SSH]
SSH
Site access machine
(access.lyon.grid5000.fr)
[SSH]
OARSUB to gain access
then OARSH or SSH
Site frontend
(frontend.lyon aka lyon)
[OARSUB, KADEPLOY]
Site clusters/nodes
(e.g.: capricorne-12.lyon)
Grid'5000 dedicated
backbone
SSH
Site clusters/nodes
(e.g.: genepi-21.grenoble)
OARSUB to gain access
then OARSH or SSH
SSH
Site frontend
(frontend.grenoble aka grenoble)
[OARSUB, KADEPLOY]
Site clusters/nodes
(e.g.: parapluie-12.rennes)
Site access machine
(access.rennes.grid5000.fr)
[SSH]
Site frontend
(frontend.rennes aka rennes)
[OARSUB, KADEPLOY]
Site clusters/nodes
(e.g.: azur-42.sophia)
Site frontend
(frontend.sophia aka sophia)
[OARSUB, KADEPLOY]
Site access machine
(access.sophia.grid5000.fr)
[SSH]
10G dedicated lambda
Lille
Luxembourg
Rennes
Nancy
Nantes
Grid'5000
Lyon
Grenoble
Toulouse
Sophia
Last update: 2023-06-16

Grid5000 is part of SLICES-FR

Super Infrastructure for Large-scale Experimental Computer Science

- FIT, IoT experimentation platform
  - 9 sites in France
  - Test platforms on fixed and mobile networks
  - Experiments from objects to data processing
- Grid'5000
  - 10 sites in France, 8000 cores, including Toulouse since 2004
  - Experiments on Cloud, HPC, BigData, AI, etc., up to bare-metal
  - High diversity of hardware (Intel, AMD, GPUs, ARM, ...)
- SLICES-FR
  - Merger of the two platforms
  - Mutualization of hardware and software
  - Experiments from IoT to large-scale data processing

Accessing Grid5000

Fully open to all IRIT researchers (masters students, PhD, permanent staff)

- Request an account: https://www.grid5000.fr/w/Grid5000:Get_an_account
  - “Group Granting Access”: IRIT
  - Provide precise information in “motivation” and “Intended usage”
- Abide by the usage policy
  - Mostly tests/development in ‘small’ chunk during work day
  - Large experiments during night and WE
  - Add acknowledgement in publications using Grid5000
    - Add grid5000 tag on HAL

Accessing Grid5000

A ‘Site’ oriented architecture

- From outside, only `access.grid5000.fr` is available for ssh / scp
- Then access to one of the 8 sites
  - `ssh access.grid5000.fr`
  - `ssh lyon`
- Each site has its own homes (without backups)
  - On a site all computers have access to user homes
  - From outside: `scp local_file access.grid5000.fr:nancy/`

Starting to work the interactive way

No work allowed on frontend of each site (ex. `fnancy` for nancy frontend)

https://www.grid5000.fr/w/Getting_Started

1. Check availability of resources: https://www.grid5000.fr/w/Status
2. Select your site and go there (ssh)
3. Request resources
   a. `oarsub -I`
   b. Wait for the session to start. By default it lasts 1h
4. Use remotely (you have access to your home) (can use `sudo-g5k`)
5. When you exit or at the end of allocated time, the computer is wiped (files in the home are not impacted)

Starting to work the script way

Usually the scripts are tested interactively first and run directly afterward

1. Select your site and go there (ssh)
2. Check that your script is there and executable (chmod +x)
3. Submit the script
   a. `oarsub ./script.sh`
4. To check the end:
   a. https://www.grid5000.fr/w/Status or `oarstat -u` *`Login`* on the site frontend
5. Stdout and Stderr are stored in `OAR.*.[stdout|stderr]` files

When dealing with multiple servers

Similar in scripts and interactive modes

`$OAR_NODE_FILE` : file containing list of reserved cores

Multiple solutions

- Directly use the file for `ssh` / `mpirun` / … and run applications
- Use high-level tools
  - python-grid5000
  - execo
  - expetator
  - https://www.grid5000.fr/w/Grid5000:Software

Selection of resources

Everything can be chosen

- Number of servers
- Type of processors
  - Speed, Intel/AMD/ARM
- Network topology
  - Force same router
  - Force different routers
- Type of network card
  - Ethernet
  - Infiniband
  - Omni-path
- Type of storage
  - SSD / HDD / NVME / Multiple disks
- Accelerators
  - GPUs, Xeon Phi
- RAM
  - Size, PMEM
- O/S
  - Can reinstall the O/S

Jupyter Demo

Direct access to a normal or GPU node

https://www.grid5000.fr/w/Notebooks

Directly on a normal node

Checking the resources

1. https://www.grid5000.fr/w/Status
   a. Gives all type of information: occupation, network, energy
   b. In Drawgantt the gant shows the current occupation
2. Exemple in Nancy
   a. https://intranet.grid5000.fr/oar/Nancy/drawgantt-svg/
3. https://intranet.grid5000.fr/notebooks/hub/home
   a. Start My Server
   b. Select a site : here **Nancy**
4. Start and wait

Directly on a normal node

A Jupyter hub now runs on a new server

- All files of this site (here **Nancy**) are available (through NFS)
- Can directly load an existing `.ipynb` file or create a new one
  - Bash (example on Nancy site)
    - `demo_nancy_node_bash.ipynb`
  - Python (example on Nancy site)
    - `demo_nancy_node_python.ipynb`
- Libs must be installed once per site
  - Python / binaries / …
- Temporary files can be put in `/tmp/` directory (for speed concern)

Directly on a normal node

Eternity is really long, especially near the end

Jupyter hub stays until it is terminated or reservation is finished

- Can be seen in the Drawgantt
- Can be seen directly on the frontal
  - `oarstat -u gdacosta`
- To go back: https://intranet.grid5000.fr/notebooks/hub/home
- As long as it is not closed the computer is blocked !

Close your hub when not in use: Stop My Server

Directly on a GPU node

Checking the resources

Not all sites have GPU nodes

- List is available here: https://www.grid5000.fr/w/Hardware

Otherwise similar

- Replace `/host=1` by `/gpu=1` on a site where GPUs are available (exemple Lille)
- Example (on Lille site): `demo_lille_gpu_torch.ipynb`

Directly on a node with wattmeter

Adding expetator (https://gitlab.irit.fr/sepia-pub/expetator)

Several types of wattmeters

- BMC level: time resolution of 1s
- Wattmeters: time resolution can go down to 1/20s
  - Example at Nancy: `{cpu >= 1681 AND cpu <= 1711}/host=2`
- https://www.grid5000.fr/w/Energy_consumption_monitoring_tutorial
- Example at Nancy site: `demo_nancy_node_expetator.ipynb`

[spam] Expetator

https://gitlab.irit.fr/sepia-pub/expetator

Tool for testing experimental campaign

- Multiple benchmarks (NPB, gromacs, gpu, mem, net)
- Multiple leverages (dvfs, powercap, GPU dvfs)
- Multiple monitoring (RAPL, Performance counters, system load, network, server power, GPU power)

Watermarking for aggregating data from multiple sources

Still in development, contact Georges Da Costa for informations

On a frontend

For coordination of experiments

Classical rules apply on frontends

- No heavy workload
- No high usage of memory

Mostly used to start/monitor complex or large experiments

- Usage of grid5000-python (example on Nancy frontend)
  - `demo_nancy_frontal.ipynb`
- Usage of execo (example on Nancy frontend)
  - `demo_nancy_frontal-execo.ipynb`

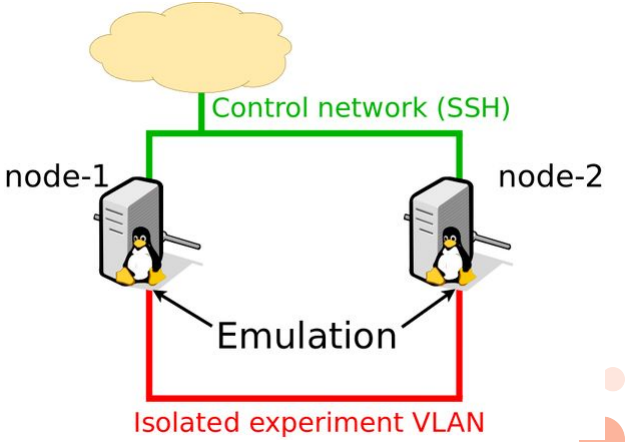
Network emulation

https://www.grid5000.fr/w/Network_emulation

Experiment of complex networks

Direct usage or emulation

1. Experiments can be done between sites using Grid5000 network
2. Bandwidth, latency, and packet loss can be changed on particular links
   a. Netem
   b. Virtual LAN (reservation of LAN similar to reservation of servers)
   c. Distem is a tool that can emulate a distributed system on a homogeneous Cluster
   d. EnOSlib, Distrinet



Virtualization

Virtualization: Cloud/SDN/Storage

https://www.grid5000.fr/w/Virtualization_in_Grid%275000

Multiple directions

- Virtual machines: tutorial on KVM / XEN
- SR-IOV and Virtual Function PCI passthrough
- SDN: using Virtual Lan and sub-net reservation
- Higher level virtualization
  - Singularity / Kubernetes / libvirt / docker / OpenStack
- Storage
  - Shared storage (NFS, Ceph, …) but also local disk reservation

Bare-metal access

Lowest level of access

Root access on servers

- Command `sudo-g5k`

Possibility to reboot servers on any image

- Command `kadeploy`
- With `kavlan` provide a completely tunable environment

Conclusion

Two main usages

- Access to particular type of hardware/configuration/scale
- Access to large amount of raw computing power

Next steps

- More interconnection with IoT systems
- @IRIT: renewal of the infrastructure with Jetson Xavier NX cluster
  - Supercomputer for Embedded and Edge Systems from NVIDIA
- Project-focused support
  - Contact georges.da-costa@irit.fr to have your own personnel training