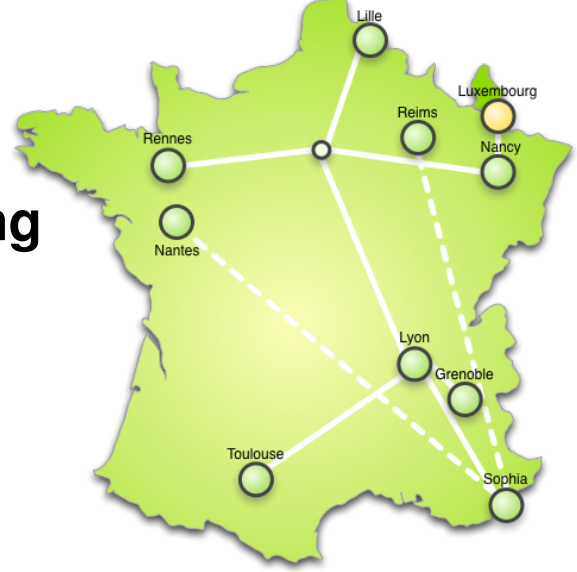
Virtualization on Grid'5000

Lucas Nussbaum

with the Grid'5000 architects committee
and the Grid'5000 technical team

The Grid'5000 testbed

- **World-leading testbed for distributed computing**
  - 9 sites, 30 clusters, 859 nodes, 8456 cores
  - Dedicated 10-Gbps backbone network
  - 550 users and 100 publications per year



The Grid'5000 testbed

- **World-leading testbed for distributed computing**
  - 9 sites, 30 clusters, 859 nodes, 8456 cores
  - Dedicated 10-Gbps backbone network
  - 550 users and 100 publications per year
- Not a typical grid / cluster / Cloud, more a meta-grid, meta-cloud:
  - Used by CS researchers in HPC / Clouds / Big Data / Networking to perform experiments
  - **Design goals:**
    - ★ **Large-scale, shared infrastructure**
    - ★ **Support high-quality, reproducible research**
  - Litmus test: *are you interested in the result of your computation, or in how it performed?*

Some virtualization & Cloud experiments

- Virtual machines management
  - Study of the migration process $\rightsquigarrow$ SimGrid model[1]
  - Improving performance of VM migration[2]
  - Evaluation of VM placement strategies[3]
- Energy efficiency of cloud infrastructures
- Design / Improvement of cloud middlewares
  - Autonomic IaaS Cloud: Snooze[4]
  - Fog computing, Distributed OpenStack (DISCOVERY project, Inria/Orange joint lab)[5]

[1] Laurent Pouilloux et al. "SimGrid VM: Virtual Machine Support for a Simulation Framework of Distributed Systems". In: *IEEE Transactions on Cloud Computing* (Sept. 2015).

[2] Pierre Riteau. "Dynamic Execution Platforms over Federated Clouds". Theses. Université Rennes 1, Dec. 2011.

[3] Adrien Lebre et al. "VMPlaceS: A Generic Tool to Investigate and Compare VM Placement Algorithms". In: *Europar 2015*. Vienne, Austria, Aug. 2015.

[4] Eugen Feller. "Autonomic and Energy-Efficient Management of Large-Scale Virtualized Data Centers". Theses. Université Rennes 1, Dec. 2012.

[5] Frédéric Desprez et al. "Energy-Aware Massively Distributed Cloud Facilities: The DISCOVERY Initiative". In: *IEEE International Conference on Green Computing and Communications (GreenCom)*. Sydney, Australia, Dec. 2015, pages 476–477.

Reconfiguring the testbed

- Typical needs:
  - How can I install $SOFTWARE on my nodes?
  - How can I add $PATCH to the kernel running on my nodes?
  - Can I run a custom MPI to test my fault tolerance work?
  - How can I experiment with that Cloud/Grid middleware?
  - Can I get a stable (over time) software environment for my experiment?

Reconfiguring the testbed

- Typical needs:
  - How can I install $SOFTWARE on my nodes?
  - How can I add $PATCH to the kernel running on my nodes?
  - Can I run a custom MPI to test my fault tolerance work?
  - How can I experiment with that Cloud/Grid middleware?
  - Can I get a stable (over time) software environment for my experiment?
- Likely answer on any production facility: **you can't**
- Or:
  - Install in $HOME, modules, etc. $\rightsquigarrow$ no root access, need to handle custom paths
  - Use virtual machines $\rightsquigarrow$ experimental bias (performance), limitations
  - Containers: kernel is shared $\rightsquigarrow$ various limitations, security?

Reconfiguring the testbed

- Operating System reconfiguration with Kadeploy:
  - Provides a *Hardware-as-a-Service* Cloud infrastructure
  - Enable users to deploy their own software stack & get *root* access
  - **Scalable, efficient, reliable and flexible**:
    200 nodes deployed in ~5 minutes (120s with Kexec)

Creating and sharing Kadeploy images

- Avoid manual customization:
  - Easy to forget some changes
  - Difficult to describe
  - The full image must be provided
  - Cannot really serve as a basis for future experiments (similar to binary vs source code)
- Kameleon: Reproducible generation of software appliances
  - Using *recipes* (high-level description)
  - Persistent cache to allow re-generation without external resources (Linux distribution mirror) $\rightsquigarrow$ self-contained archive
  - Supports Kadeploy images, LXC, Docker, VirtualBox, qemu, etc.

http://kameleon.imag.fr/

Other Virtualization & Cloud XP requirements

- Efficient provisioning of hypervisors
  - ✓ Kadeploy (support for Xen & KVM)
- Storage (VM images, etc.)
  - ✓ Storage5k (reserved NFS storage), Ceph clusters (block)
- Easy Cloud stacks deployment
  - ✓ Tool to automate OpenStack installation inside a job
- Networking support

IP range reservation: G5K-subnets

- Grid'5000 enables different users to run experiments concurrently
  - Need a mechanism to provide IP ranges for virtual machines
- G5K-subnets adds IP ranges reservation to OAR

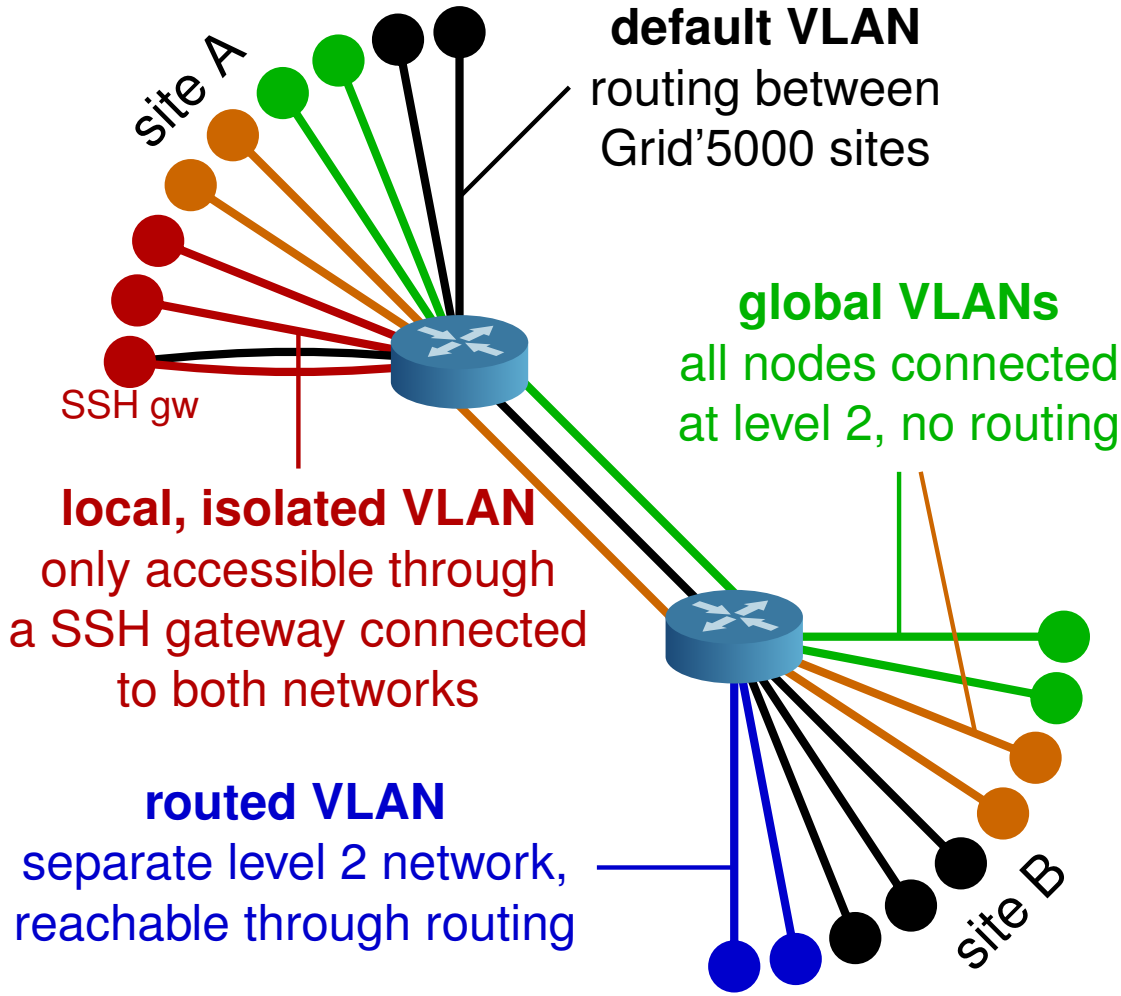
```
oarsub -l slash_22=2+nodes=8 -I
```

- Those IP ranges are routed inside Grid'5000
- But no isolation: one can *steal* IP addresses

Network isolation with KaVLAN

- Reconfigures switches for the duration of a user experiment to achieve **complete level 2 isolation**:
  - Avoid network pollution (broadcast, unsolicited connections)
  - Enable users to start their own DHCP servers
  - Experiment on ethernet-based protocols
  - Interconnect nodes with another testbed without compromising the security of Grid'5000
- Some nodes with several (up to 4) network interfaces
- Relies on **802.1q (VLANs)**
- Compatible with many network equipments
  - Can use SNMP, SSH or telnet to connect to switches
  - Supports Cisco, HP, 3Com, Extreme Networks and Brocade
- Controlled with a command-line client or a REST API

KaVLAN - different VLAN types



Conclusions

- Bare metal deployment, virtual machines, containers, modules all have pros and cons
  - Bare-metal is slow and a heavy solution for some needs
  - On Grid'5000, we also provide sudo-g5k (root access on the *standard* (default) environment)
- Other problems must be addressed:
  - Images management (home-made, or Vagrant, Docker, etc.?)
  - Images storage
  - Networking support
    - Allocation and reservation of IP addresses
    - Isolation? ($\rightsquigarrow$ VLANs? VXLAN?)
  - Orchestration: shell scripts might not be sufficient
- Note: Grid'5000 has an Open Access program. Feel free to try it!

Bibliography

- Resources management: Resources Description, Selection, Reservation and Verification on a Large-scale Testbed. http://hal.inria.fr/hal-00965708
- Kadeploy: Kadeploy3: Efficient and Scalable Operating System Provisioning for Clusters. http://hal.inria.fr/hal-00909111
- KaVLAN, Virtualization, Clouds deployment:
  - Adding Virtualization Capabilities to the Grid'5000 testbed. http://hal.inria.fr/hal-00946971
  - Enabling Large-Scale Testing of IaaS Cloud Platforms on the Grid'5000 Testbed. http://hal.inria.fr/hal-00907888
- Kameleon: Reproducible Software Appliances for Experimentation. https://hal.inria.fr/hal-01064825
- Distem: Design and Evaluation of a Virtual Experimental Environment for Distributed Systems. https://hal.inria.fr/hal-00724308
- XP management tools:
  - A survey of general-purpose experiment management tools for distributed systems. https://hal.inria.fr/hal-01087519
  - XPFlow: A workflow-inspired, modular and robust approach to experiments in distributed systems. https://hal.inria.fr/hal-00909347
  - Using the EXECO toolbox to perform automatic and reproducible cloud experiments. https://hal.inria.fr/hal-00861886
  - Expo: Managing Large Scale Experiments in Distributed Testbeds. https://hal.inria.fr/hal-00953123
- Kwapi: A Unified Monitoring Framework for Energy Consumption and Network Traffic. https://hal.inria.fr/hal-01167915
- Realis'2014: Reproductibilité expérimentale pour l'informatique en parallélisme, architecture et système. https://hal.inria.fr/hal-01011401