



# Extending the PlanetLab Usage Model for Distributed Heterogeneous Research Testbeds

**Roberto Canonico\*, Salvatore D'Antonio and Giorgio Ventre**

*University of Napoli "Federico II" – Italy*

*&*

*CINI Consorzio Interuniversitario Nazionale per l'Informatica*

`roberto.canonico@unina.it`



# Outline

- Kinds of experiments running on a distributed network testbed
- Controllability vs. realism
- The PlanetLab usage model
- The Emulab usage model
- Heterogeneity of the OneLab testbed
- How to take diversity into account
- Current trends in the evolution of large-scale distributed testbeds: GENI, VINI, ...
- How to extend the PlanetLab usage model ...
- Discussion



# Kinds of experiments running on a distributed network testbed

- *Continuously-running services*
  - inherently robust to variations of resource availability
  - benefit from the availability of a **realistic** execution environment (e.g. the *real* Internet)
  - usually implemented as overlay testbeds on top of Internet, since it's difficult to involve in the experiment intermediate nodes (routers) of the real Internet
- *Short-term experiments*
  - aimed at producing replicable results in **controlled** conditions
  - ability to inject exogenous events
  - admission control mechanisms to ensure sufficient capacity is allocated for the duration of a run and that the experiment may run in the desired conditions
  - implemented as emulation testbeds or as experimental networks reproducing real nets

# Resource sharing in a distributed testbed

- To maximize the benefit/cost ratio, the infrastructure must be able to support multiple concurrent experiments at the same time
- As always in the case of concurrency, resource sharing must be properly handled
- Resources to be shared:
  - CPU
  - Memory
  - Communication resources (IP addresses, port numbers, link bandwidth, radio channels, ...)
  - Storage
  - Other specialized hardware
  - ...

# Approaches to resource sharing

- In traditional OSES, resources are shared in time-sharing by means of proper schedulers
  - Usually, these schedulers aim at fairness
  - In real-time OSES, special schedulers combined with proper admission control tests, are able to provide guaranteed amount of a shared resource per task
- In distributed testbeds, a portion of the testbed resources, called a *slice*, is assigned to each experiment
- Slicing is usually implemented by means of virtualization techniques

# An example of overlay testbed: PlanetLab

- In PlanetLab slice creation and resource allocation are decoupled
- When a slice is first created, a best effort service is associated with it
  - Slices are not bound to sets of guaranteed resources
  - Resources are acquired and released by the slices during their entire lifetime
- PlanetLab has not been designed for controlled repeatable experiments, but to test services in a real world scenario
- The testbed cannot be isolated from the Internet's varying and unpredictable conditions
- SIRIUS has been introduced later, and manages only a few of the shared resources...

# An example of emulation testbed: Emulab

- Emulab provides users with full control over the environment of their experiments by enabling them to introduce exogenous events
  - link failures, changes in traffic volume, congestions
- While allocating resources, Emulab aims at solving a combinatorial optimization/constraint satisfaction problem
  - To give users resources that match their requests based on the knowledge of the current network state
- Repeatability of experiments is guaranteed, but no realistic element belonging to an operational network is included
- Emulab has evolved in Flexlab, which aims at overcoming the limits of pure emulation testbeds by introducing the possibility of integrating real network and traffic models captured from the Internet



- The OneLab research infrastructure is being built with the following goals:
  - Extend PlanetLab into new environments, beyond the traditional wired Internet
    - In particular, integrating wireless access technologies: UMTS, WiMax, WMNs, ...
    - ... and support for multihoming scenarios
  - Enrich PlanetLab with monitoring and topology discovery capabilities
  - Provide experimenters with special purpose boxes that can be used, for instance, to emulate the behavior of any kind of access network
- Project funded by the EC (IST-2006-034819)
  - [www.one-lab.org](http://www.one-lab.org)





# Heterogeneity of the OneLab testbed

- OneLab introduces diversity in a large-scale distributed testbed
- Not all the nodes are same
  - Nodes differ in terms of equipments, processing power, link capacity, communication technologies, and so on
  - Not all resources may be easily shared in time-sharing
- Problem #1: how to take diversity into account at slice creation time
  - E.g. provide users with info on what is available where and when
- Problem #2: how to take diversity into account at resource allocation time
  - E.g. by assuring that experiments do not collide or interfere with each other when using special kinds of resources (e.g. a wireless link)

# How to take diversity into account at slice creation time

- Extend the centralized database of nodes with node-specific information
  - E.g. list of NICs, type of networking technology, sharing model
- Supporting complex queries to select nodes that satisfy some specific requirements
  - E.g. list all nodes located in Italy with a UMTS connection provided by 3G operator X
  - E.g. select 10 nodes that have a connection to the same Wireless Mesh Network in San Diego, CA
  - E.g. list all nodes with link emulation capabilities provided by a dummynet box

# How to take diversity into account at resource allocation time

- Unless specific resource sharing techniques are developed for special kinds of resources, ...
  - E.g. wireless virtualization in GENI
- ... these resources should only be used through a time-based advance resource reservation system
- Extending the SIRIUS system is a solution
- Other mechanisms must be implemented in the sliver implementation to enforce that only authorized slices are allowed to use a given resource

# Current trends in the evolution of large-scale distributed testbeds: GENI, VINI, ...

- The GENI initiative aims at building a large-scale realistic experimental facility for evaluating new network architectures
  - Support for both kinds of experiments
  - Meant to last 15-20 years, with a construction phase of 5-7 yrs.
  - Includes a specific effort aimed at studying virtualization of wireless links
- VINI is a virtual network testbed which supports experiments requiring some level of both control and realism
  - VINI is a pragmatic step in the direction of GENI
  - Mainly aimed at testing routing protocols on overlay networks
  - Virtual links implemented as tunnels between PlanetLab-based nodes

# The role of virtualization in slicing

- Virtualization used to create separate non-interfering execution environments
  - A tool to achieve multiplexing of shared resources
- PlanetLab design choice: use of a lightweight virtualization technique
- The IP protocol stack is not really duplicated among execution environments (contexts)
  - Good for performance
  - Mainly suitable for service testing, less for lower level experiments
  - Leads to the necessity of tricks (e.g. tunneling) if multiple virtual networks (overlays) need to coexist in the same node
- Virtualization also as a long-term solution for the future Internet
  - Project CABO @ Princeton, (Jennifer Rexford)
- Other virtualization techniques are being investigated in PlanetLab
- Question:
  - can we decouple the concept of slice from its implementation and let different virtualization techniques coexist in the same (federated) infrastructure ?

# Extending the PlanetLab usage model

- ... for OneLab and beyond ...
- Other projects, like GENI and VINI, are doing the same for their own purposes
- Better integration of SIRIUS with PLC seems a step
- Time-dependent resource reservation for managing scarce resources or any kind of resource that should not be shared in order not to jeopardize an experiment
- Is VServer the right virtualization technique for any kind of node?
- Would it be possible to adopt different slicing techniques on different nodes ?

## Extending the PlanetLab usage model (2)

- Not all the resources should be made available to everybody
- Some management actions on resources (e.g. configuration of interfaces, ...) not to be assigned directly to the experimenter but mediated by the PLC, or another centralized entity in charge of creating the environment for the experiment on the nodes
- Maybe some facilities to create the virtual execution environments needed by the experiment on the nodes
  - Possibility of choosing kernel, tools, and so on
  - A la “Frisbee” in Emulab ...
  - ... or Virtual Workspaces in GLobus GRIDS

# Conclusions

- OneLab's usage model was initially designed after PlanetLab's
- New generation testbeds are investigating slightly different usage models to better support controllable experiments on infrastructures partly overlaid on top of Internet
- In some cases a time-based advance resource reservation may be necessary
  - In particular, for controllable and replicable experiments
- SIRIUS is a starting point
- Maybe we should not stick on VServer/VNet only
  - Andy Bavier's today presentation confirms this ...
- To extend federability even further, maybe the concept of slicing should be decoupled from its implementation



# Question time

