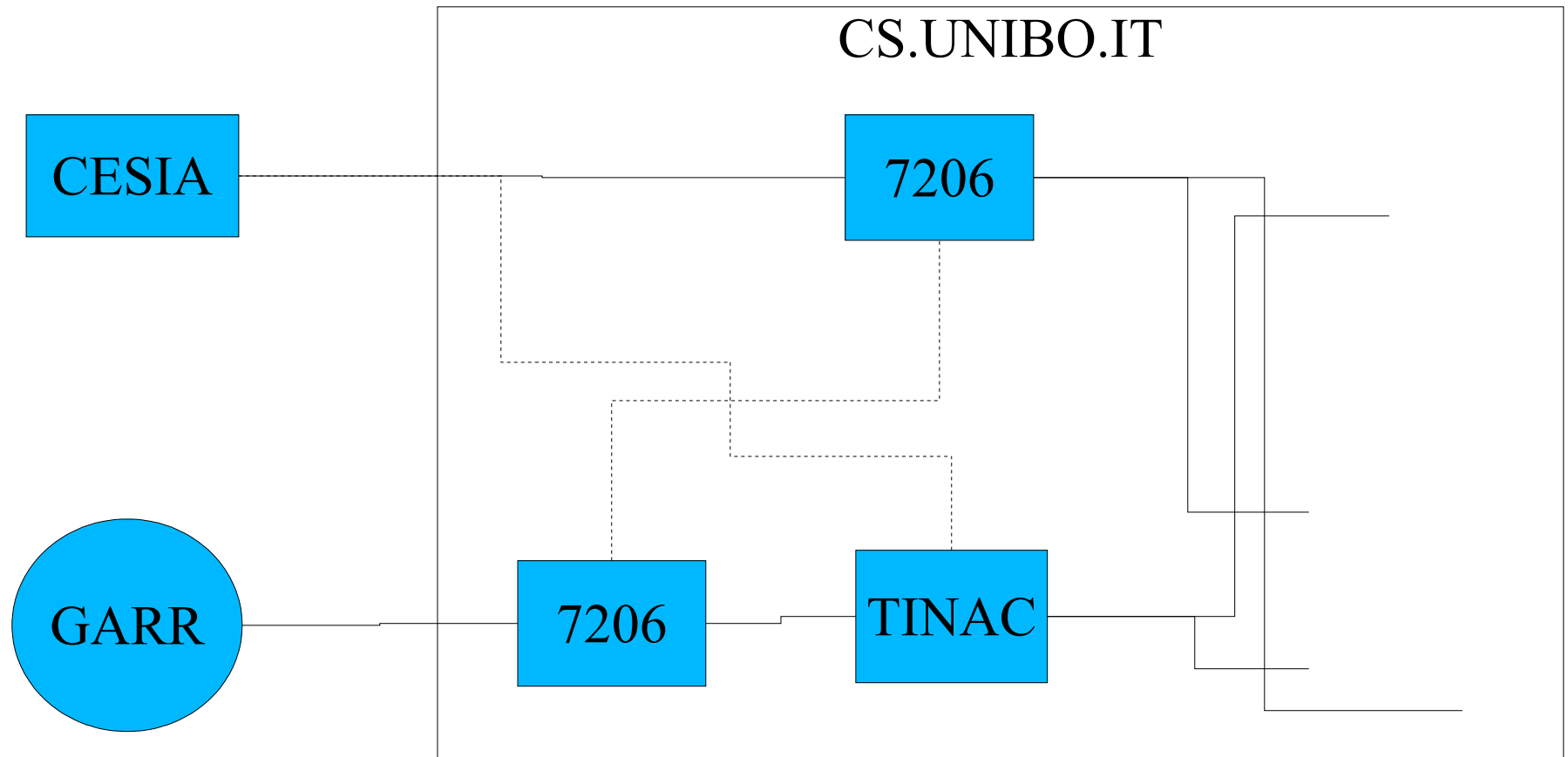
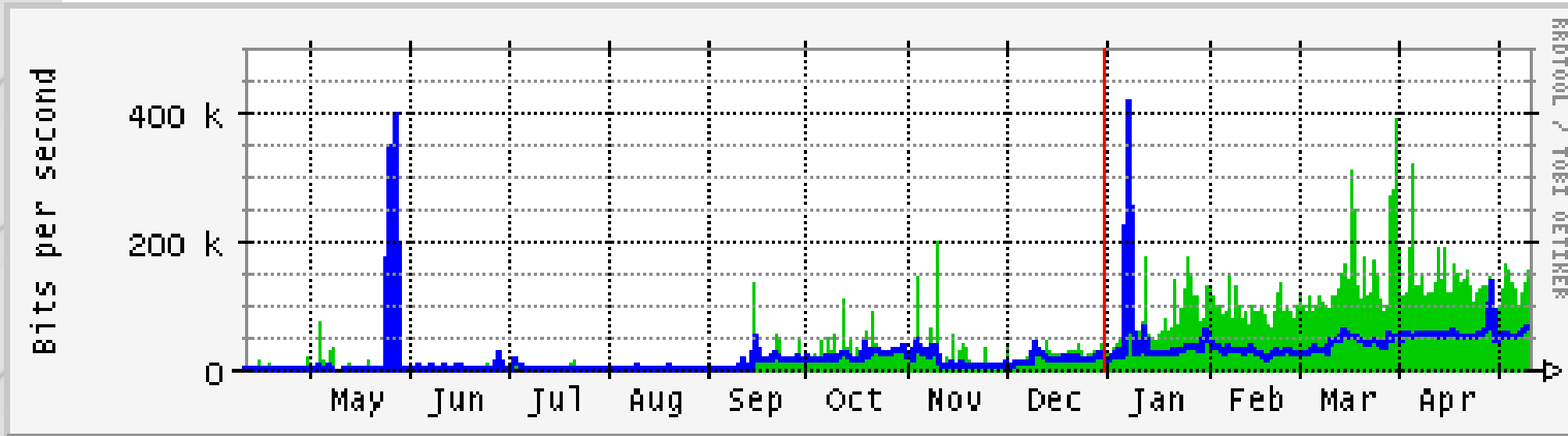


An Ipv6 Lab (inside an Ipv6 Dept.)
Renzo Davoli
Computer Science Department
University of Bologna

Ipv4 to Ipv6

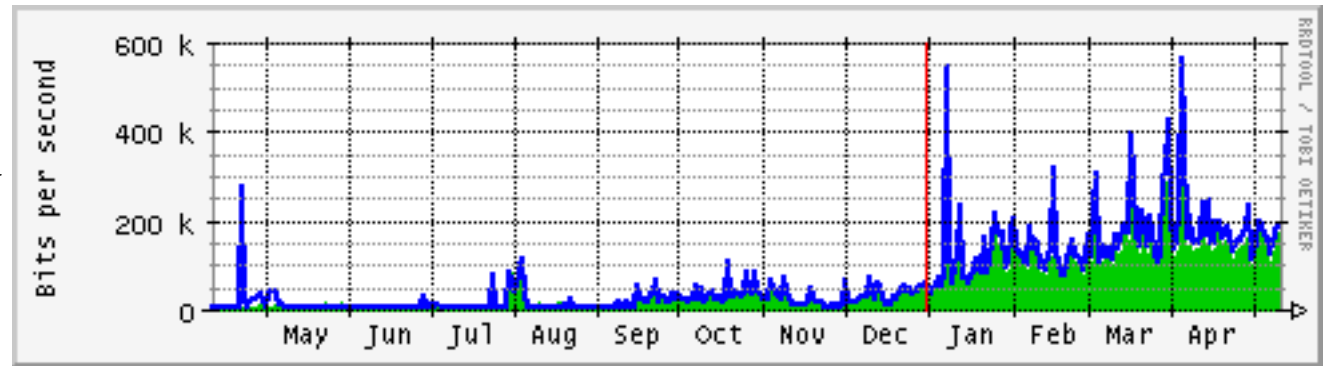


Traffic



BOLOGNA

**6net-Italy
border**



Students: the natural testbed ;-)

- Real workload
- Synergy:
 - it is educative
 - they generate real day by day traffic
 - there is a positive competition to have services
- Best feature: we keep out the “ECDL generation” students from our networks.

Interesting Tools: 1 Prometeo

- daemons/inetd = proxies/prometeo
- shared cache
- dns cache
- dynamically uploadable modules
- and obviously: Ipv4/IPV6 compatibility and service conversion
- Prometeo-proxy is GPL software on Sourceforge

Prometeo

- Is a modular proxy that provides all the shared services needed by proxies (e.g. DNS caching)
- Services can be added/deleted/started/stopped/reloaded/reconfigured at RUN time. No need to restart prometeo.
- Proxy modules are dynamic libraries loaded.
- In other words each proxy service is a plug-in for prometeo.

Interesting Tools: 2 vish

- Programmable firewall.
- Used for authenticating users (on wireless/wired networks)
- Used @cs.unibo.it to provide public access services to students' laptops inside labs, or wireless access in classrooms, corridors, garden etc.
- GPL software on Sourceforge

Interesting Tools 3: dbind

- Dynamic DNS
- laptop DNS registering (Ipv4/IPv6)
- automagical DNS tables fill-in for large clusters
- GPL software on Sourceforge

Interesting Tools 4: VDE

- Virtual Distributed Ethernet
- Overlay level 2 network
- Ipv4, Ipv6 and Ipv8 compliant (provided it will runs on an Ethernet ;-)
- GPL software on Sourceforge, it should be included soon in Debian SID

VDE

- VDE is Ethernet compliant
- VDE is general,
 - it is a virtual infrastructure that gives connectivity to several kinds of software components: emulators/virtual machines, real operating systems and other connectivity tools.
- VDE is distributed.
- VDE does not need specifically administration privileges to run.



- Square here is a mathematical notion, means virtual squared i.e. virtual elevated to the second power.
- Something is virtual square (or squared) if it is virtual in the virtual world, and this concept can then be iterated (or applied in a recursive manner) to any order of virtuality. The basic step is from virtual to virtual square.
- A virtual square world is created by virtual computers and virtual networks running on real computer and real networks and appear to the users as they were standard computer and networks.
- Nowadays it is clear that a virtual world is also a meeting place, a mean to exchange information, to communicate.

Structure of VDE

- `vde_switch`
 - it is the virtual counterpart of a physical Ethernet switch (or hub).
- `vde_cable`
 - it is the virtual counterpart of a crossed cable.
 - it is composed by two *vde_plugs* and a “wire”

VDE: applications

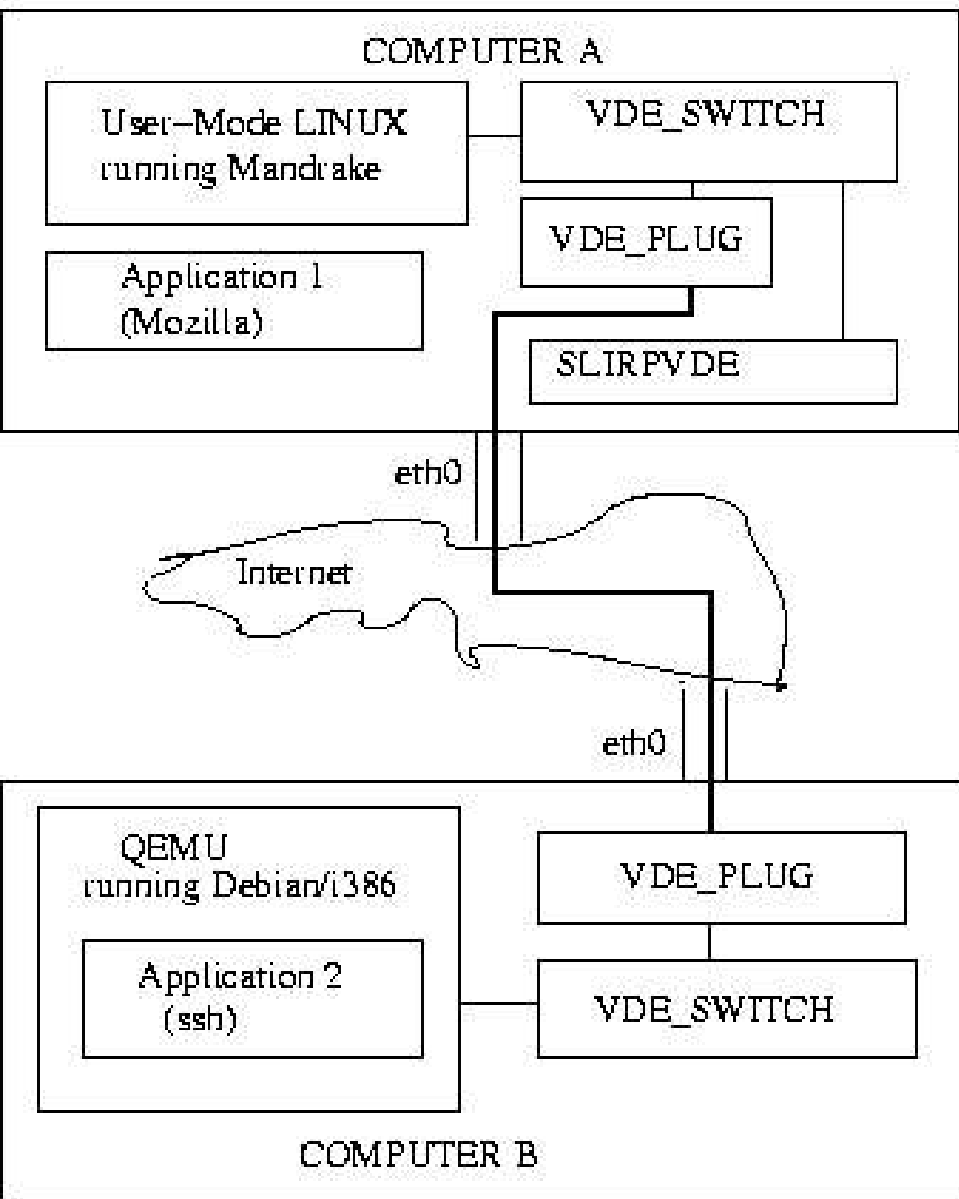
- In Education
 - networking for VM
 - virtual networking (any topology)
 - virtual parallel/distributed systems
- For Security/Privacy
 - VPN/Untraceability
- For Mobility
 - user mobility/service mobility/process mobility

VDE clients

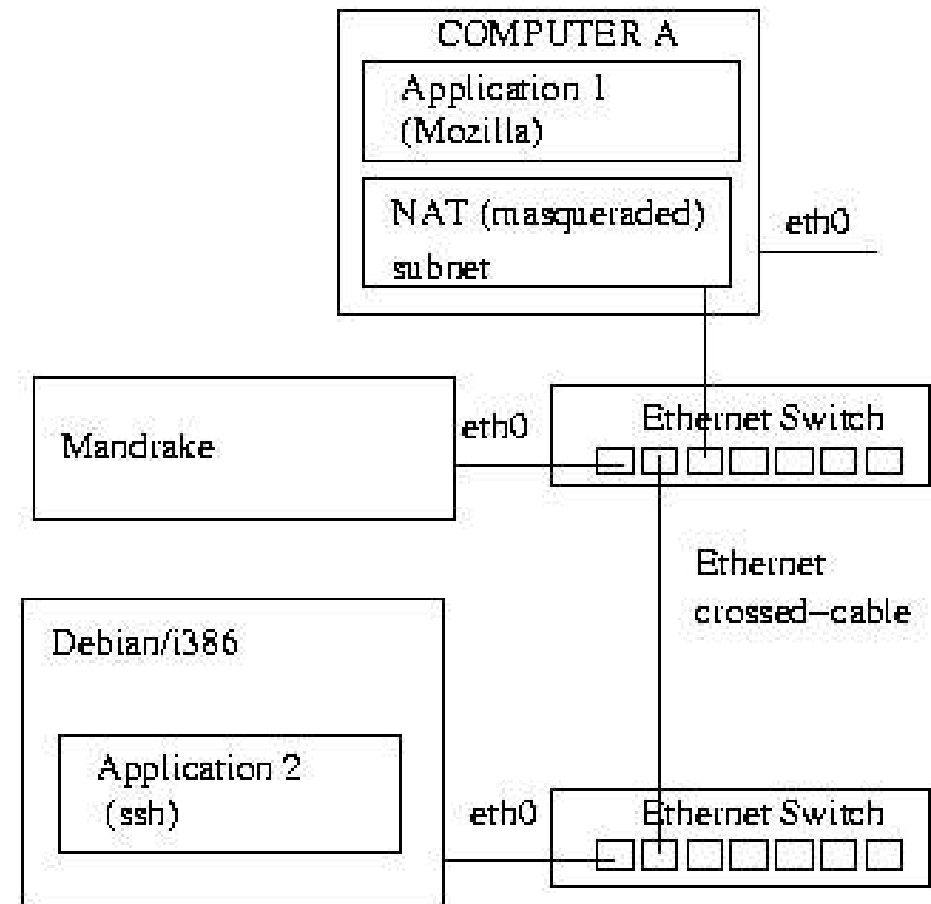
- Virtual Machines
 - qemu, bochs, uMPS, user-mode linux
- any application requiring tuntap (through an emulation layer)
- Slirp
- Ale4net
- UM-VIEWOS lwipV6 module
- A tuntap interface of the hosting operating system
 - (root access is required for this latter feature)

Example

VIRTUAL ONCE WORLD



VIRTUAL SQUARE WORLD



What's next?

- Virtual LABS (freeoszoo.org)
- VIEWOS: a process with a view
 - Partial virtual machines: the semantics of each system call can be changed independently and in several different way.
 - UM-VIEWOS is “user mode”, no root access required
 - Networking can be at process level or even at a finer degree of granularity.
 - UM-VIEWOS is GPL sw on Savannah. (first cvs release very very soon).

Thank you...

- This work is based on Free and Open Source Software. It would have been very hard, if not impossible, to do something similar based on proprietary solutions: requiring a host of non disclosure agreements, guarantees regarding access to interesting source code, possibly a promise to avoid integrating code owned by competitors, and the asking of permission to publish the research results.
- I feel that today Free and Open Source software is the key for authentic, open minded, long term research in computer science. Therefore I want to use the opportunity to thank the whole community that develops, debugs and broadcasts Free and Open Source Software and ideas.

