Open Source Routing
KTH CSD Kick-Off Workshop

Robert Olsson Uppsala University

2008-09-02

Why Open Source?

- Reclaim research and development to universities etc
- To be a part in the development loop
- Open for wide collaboration
  - No national boundaries
  - No organizational boundaries
- Easy experimentation to prototype new ideas
  Next-Generation Internet take-off
  Other ideas we can't even think of right now

Why Open Source?

- Possibilities for superior quality
  Work can be reviewed by many people
- Very fast development can be achieved
- Process can be independent from business or politics
- Non-discriminating
- Economical possibilities
- Idea started in computer science

Relation to Open Source

- Your are getting other people's work for “free”
  Respect

- Open Source does not work without contributions
  Compare a relay race. Reuse and recycle work.

- Open Source has strong momentum
  Business models are developed etc

Open Source Networking Now

- Interesting suitable hardware

  Technological breakthrough

  Multi-Core CPU, other silicons

  Fiber Optics

  Fast buses PCI-Express

- There are interesting applications

- Open source OS has come a long way

Open Source Competitors

- MIT click modular router
- Berkeley, CA

    XORP

- Vyatta

Over 10 years in production

- Three major installations

- UU core routers towards SUNET
- UU Student Network 30.000 students
- ftp.sunet.se

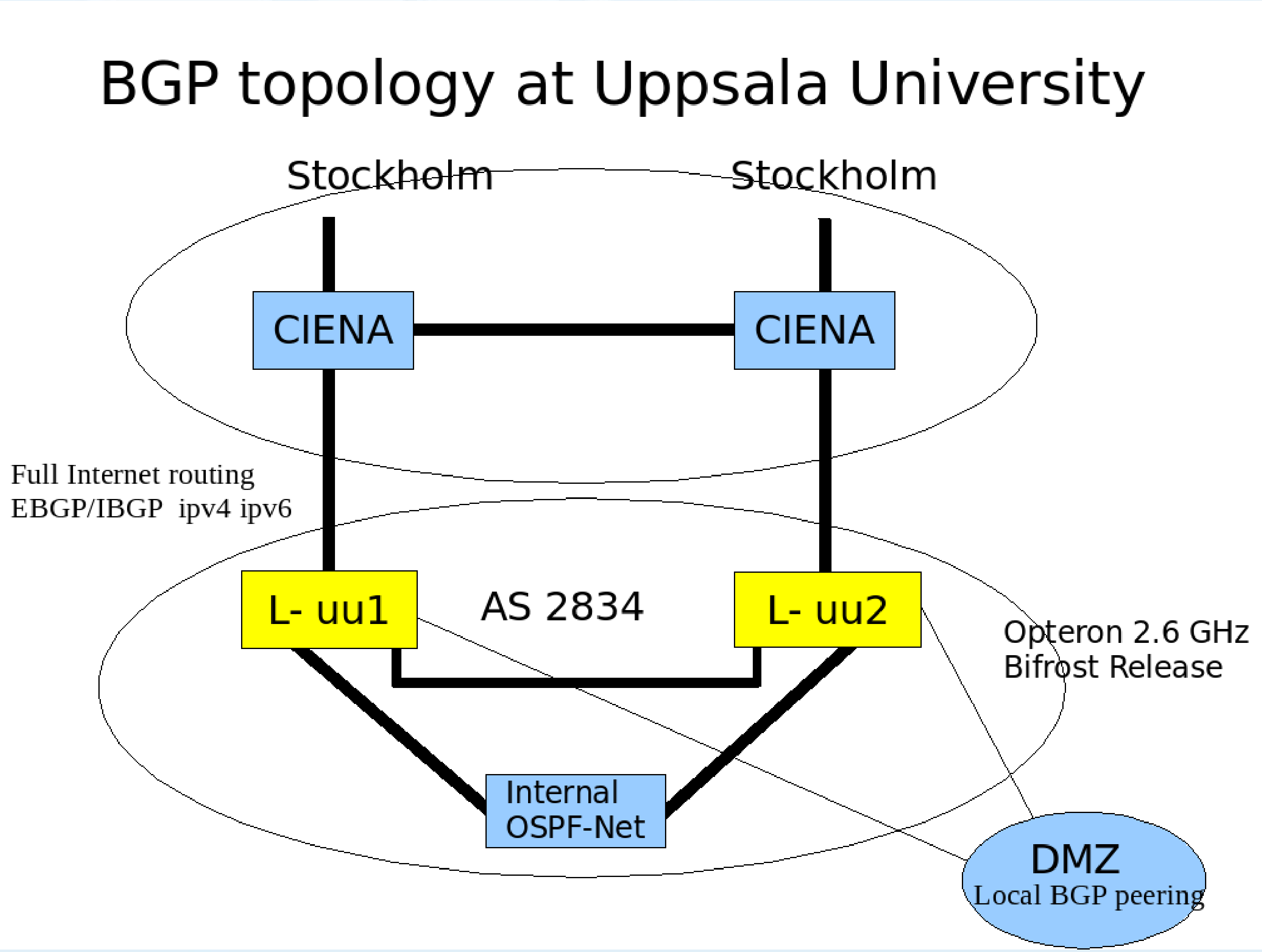
Over 10 years in production

UU facts

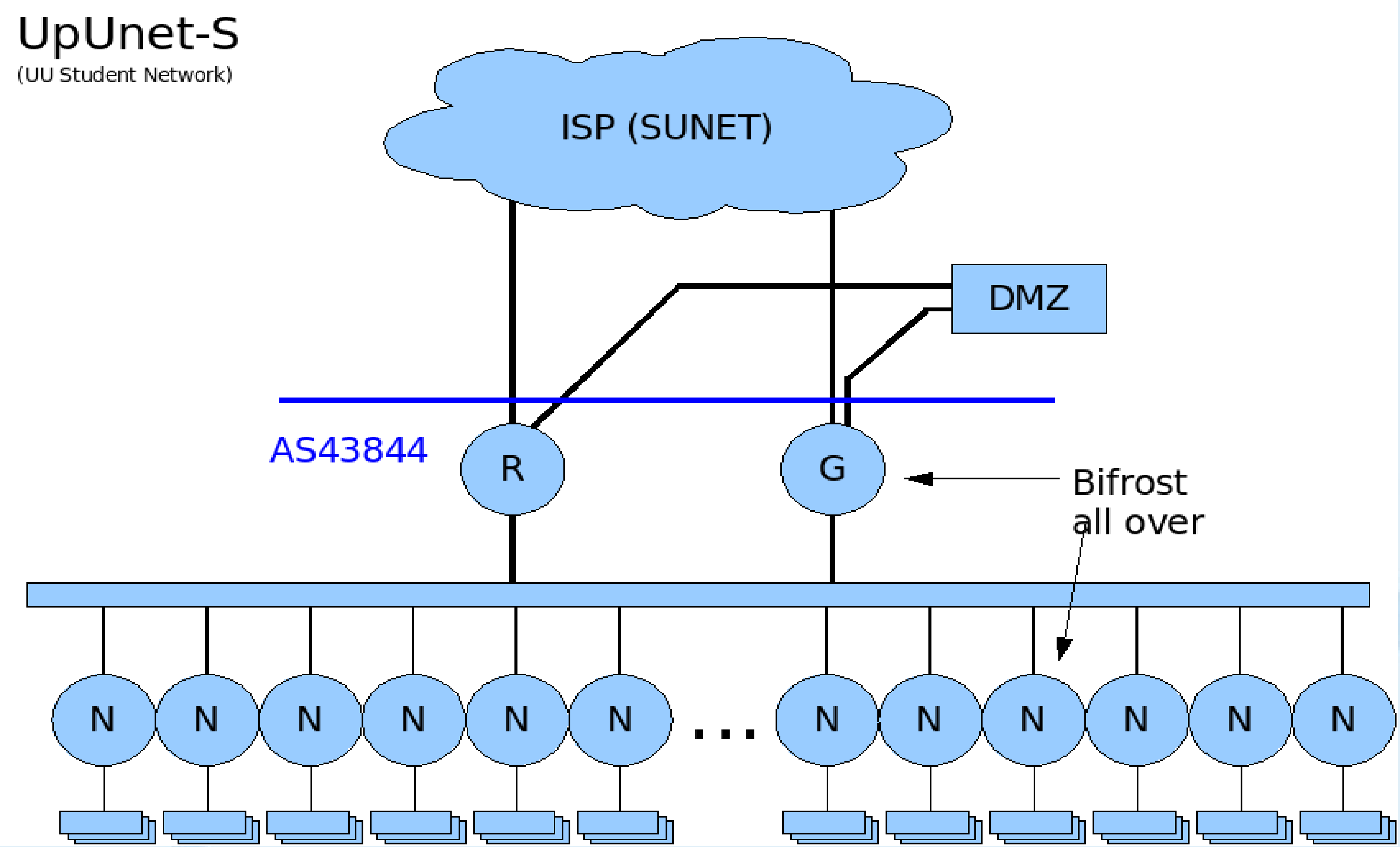
Over 25.000 registered hosts
Dual ISP BGP connect GIGE
Local BGP peering GIGE
Ipv4/Ipv6
OSPFv2/OSPFv3
600 netfilter rules
10 Cisco 6500 OSPF-routers
Redundant Power

10g planned

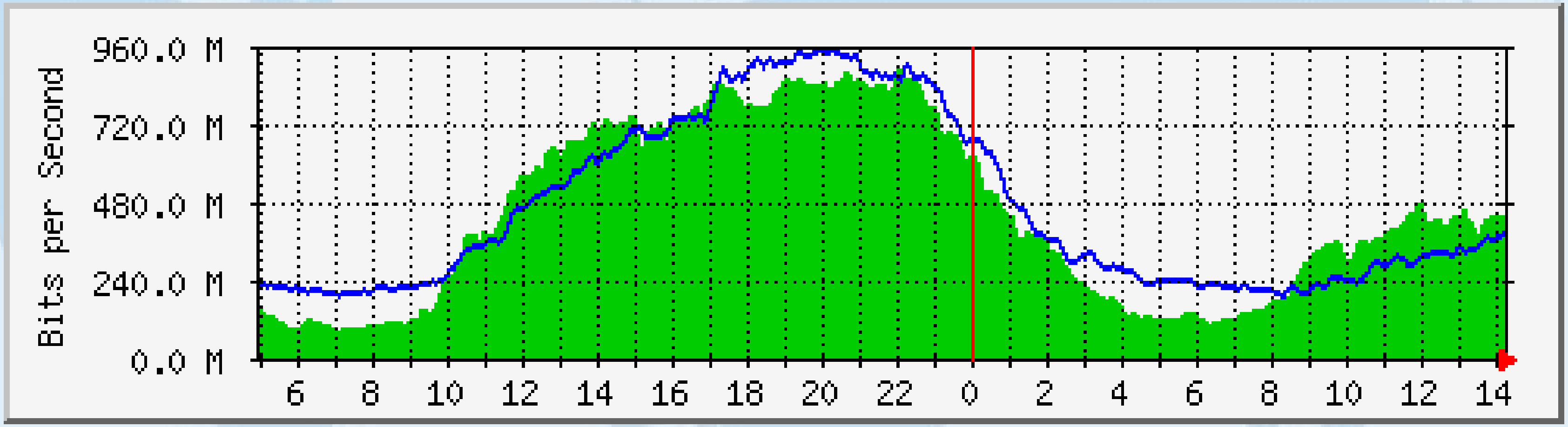
Over 10 years in production

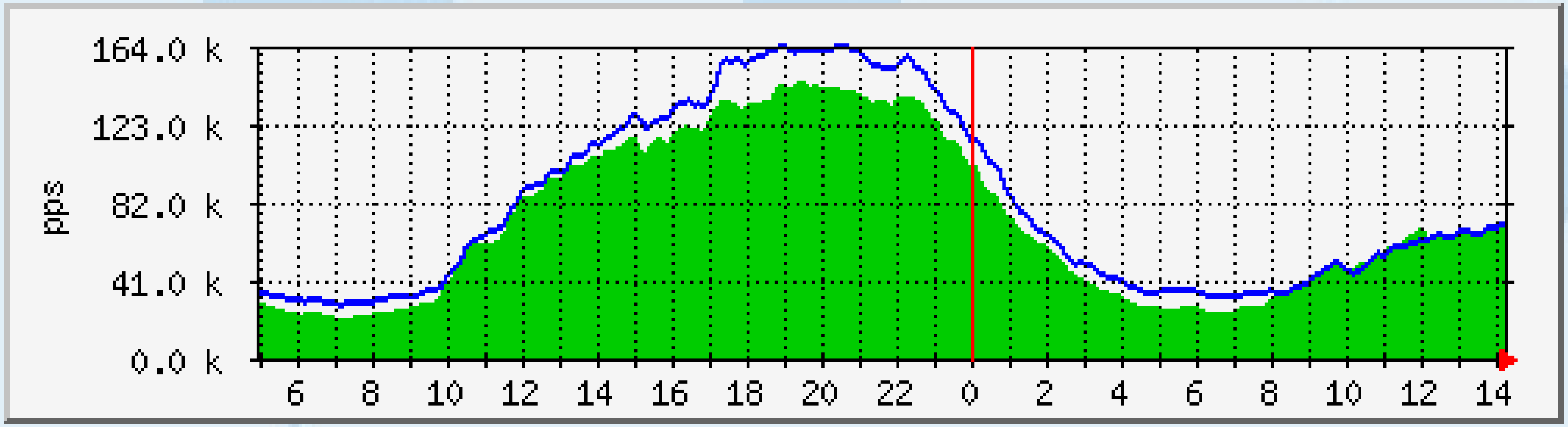


Over 10 years in production



Over 10 years in production
Student Network Core Router





Over 10 years in production

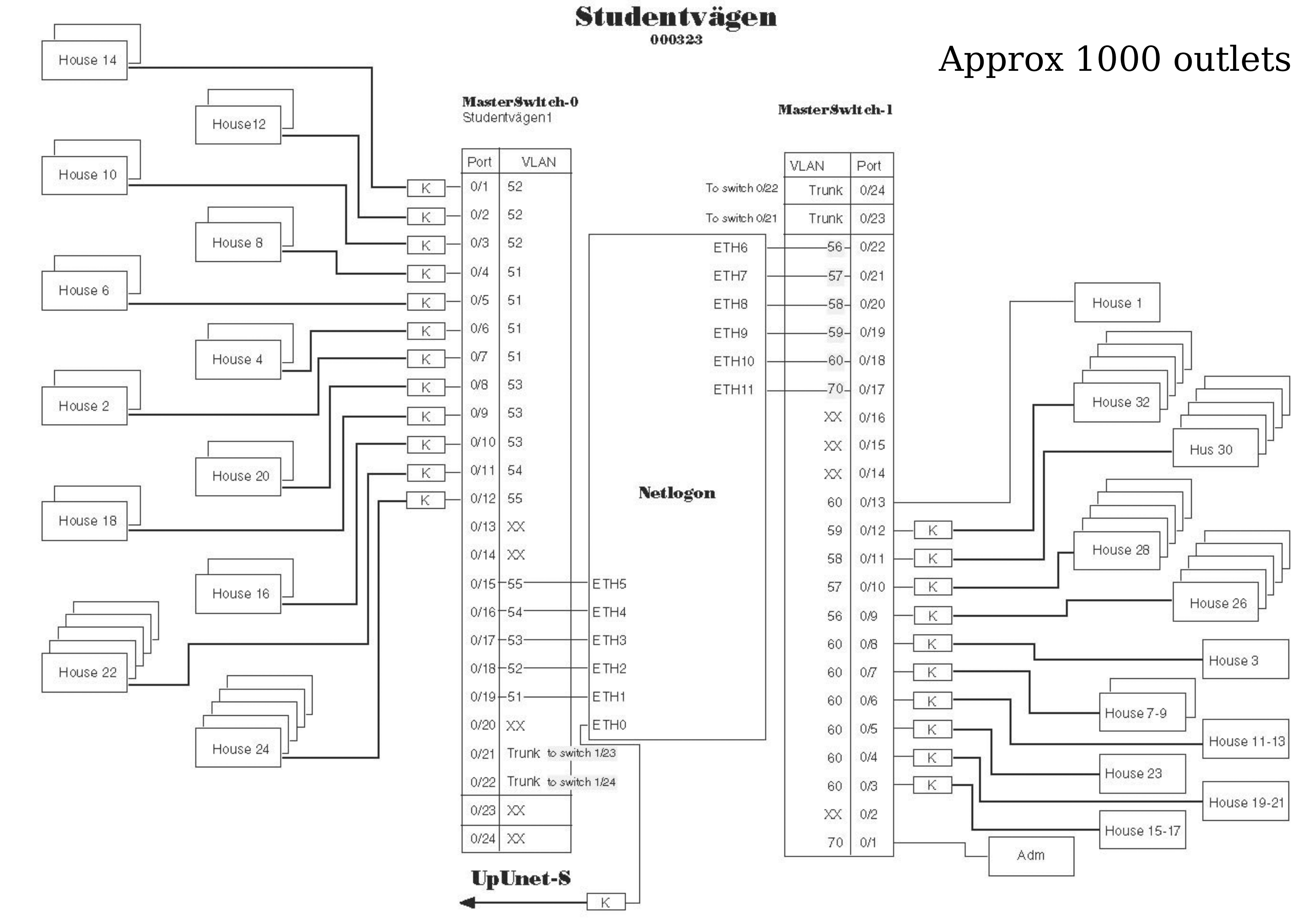
Student Network facts

Dual ISP BGP connect GIGE
Local BGP peering GIGE
Ipv4
IRDP (ICMP)
About 30 netfilter rules
19 netlogin-service boxes for premises

10g planned

IP-login installation

at Uppsala University



Approx 1000 outlets

MasterSwitch-0
Studentvägen1

Port	VLAN
0/1	52
0/2	52
0/3	52
0/4	51
0/5	51
0/6	51
0/7	51
0/8	53
0/9	53
0/10	53
0/11	54
0/12	55
0/13	XX
0/14	XX
0/15	55
0/16	54
0/17	53
0/18	52
0/19	51
0/20	XX
0/21	Trunk to switch 1/23
0/22	Trunk to switch 1/24
0/23	XX
0/24	XX

MasterSwitch-1

	VLAN	Port
To switch 0/22	Trunk	0/24
To switch 0/21	Trunk	0/23
	56	0/22
	57	0/21
	58	0/20
	59	0/19
	60	0/18
	70	0/17
	XX	0/16
	XX	0/15
	XX	0/14
	60	0/13
	59	0/12
	58	0/11
	57	0/10
	56	0/9
	60	0/8
	60	0/7
	60	0/6
	60	0/5
	60	0/4
	60	0/3
	XX	0/2
	70	0/1

Testing, Verification
Development & Research

- Started out as simple testing.
- Curiosity, Open Source, Collaboration

- Relatively freedom, the idea to use in own infrastructure. No need for external funding.

- OS was intended for desktops.

Testing, Verification Development & Research

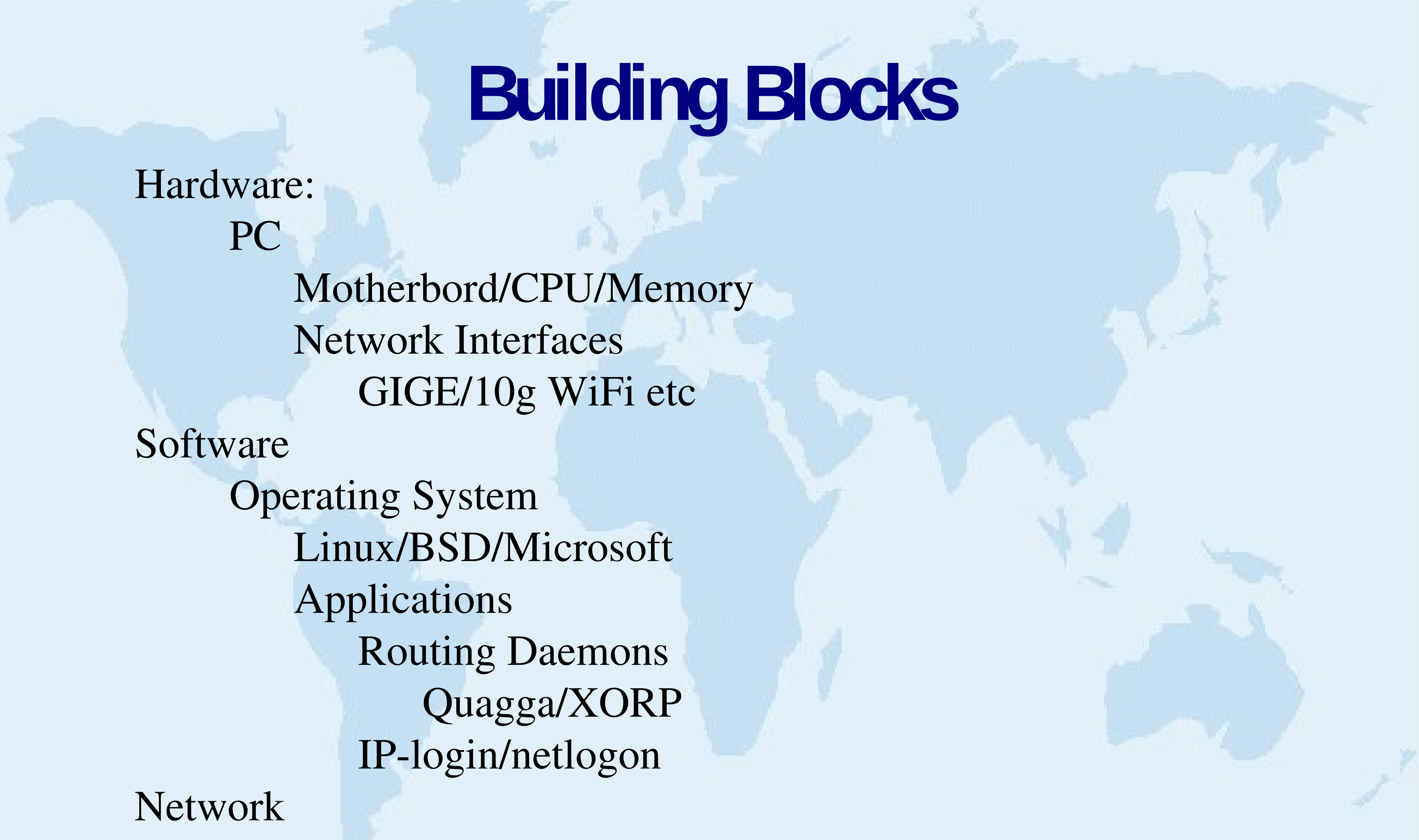
No need for test network. We could test in own infrastructure. (Or SLU)

Problem oriented vs Project oriented

We could work on complicated issues

- NAPI 3years
- pktgen 2years
- fib_trie 1year
- TRASH 1year

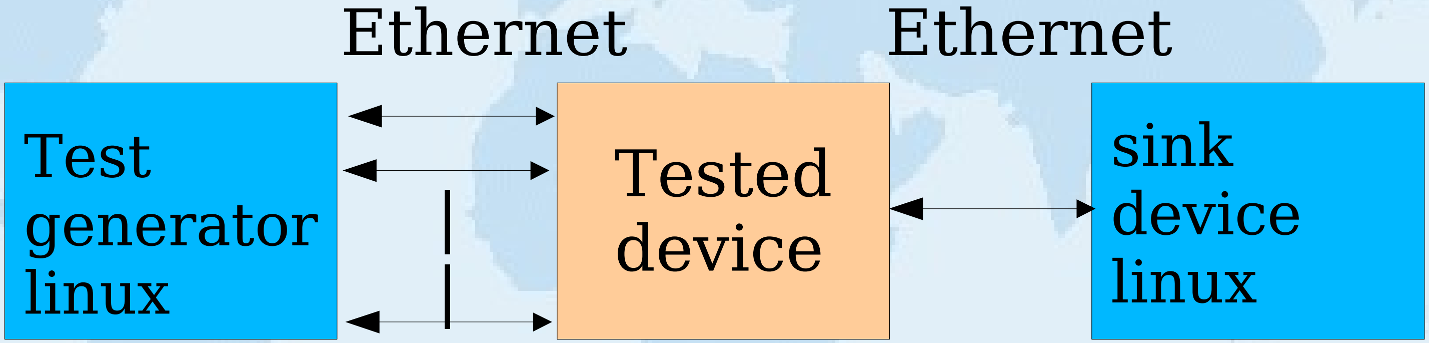
Building Blocks



- Hardware:
  - PC
    - Motherbord/CPU/Memory
    - Network Interfaces
      - GIGE/10g WiFi etc
- Software
  - Operating System
    - Linux/BSD/Microsoft
    - Applications
      - Routing Daemons
        - Quagga/XORP
      - IP-login/netlogon
- Network
  - Cable, Fiber, Copper
  - Equipment, Switches

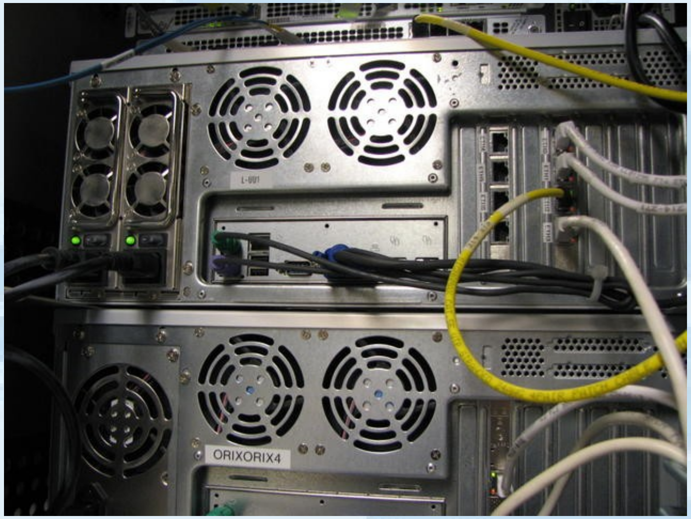
Flexible netlab at Uppsala University

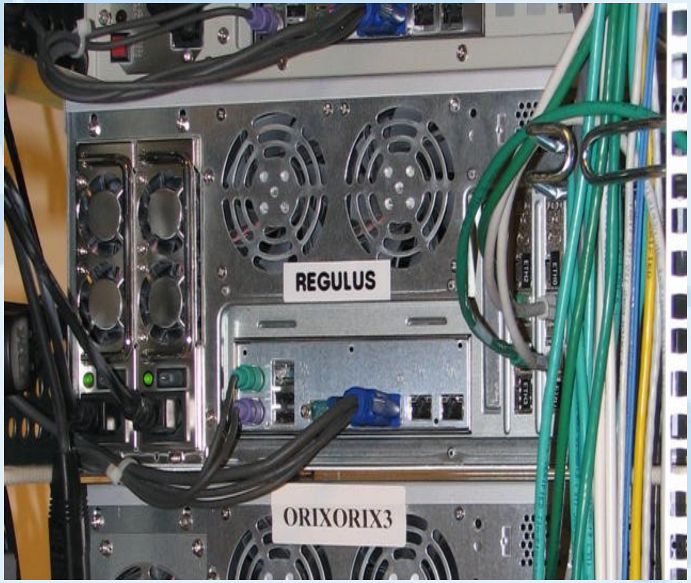
El cheapo-- High customable -- We write code :-)



* Raw packet performance
* TCP
* Timing
* Variants

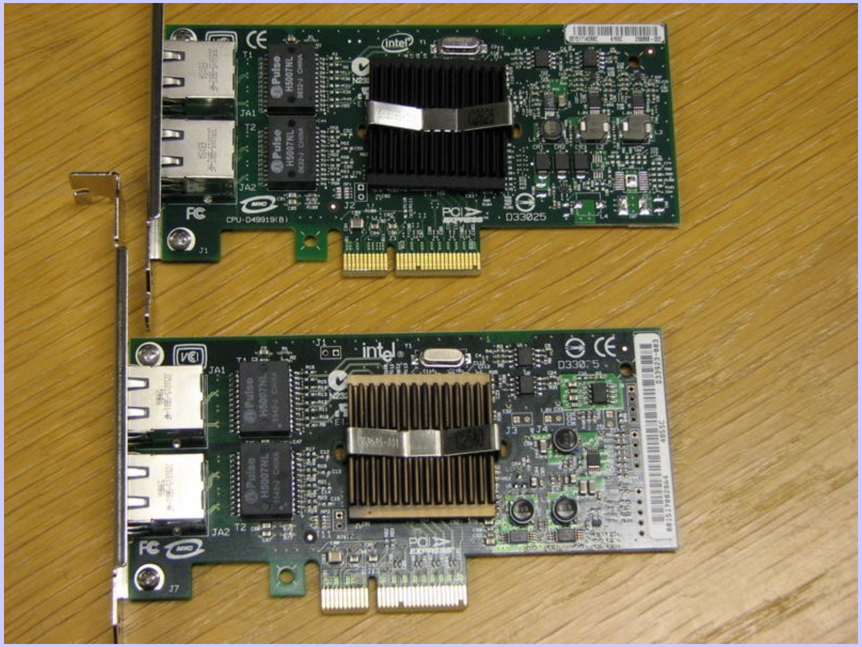
Lab at UU

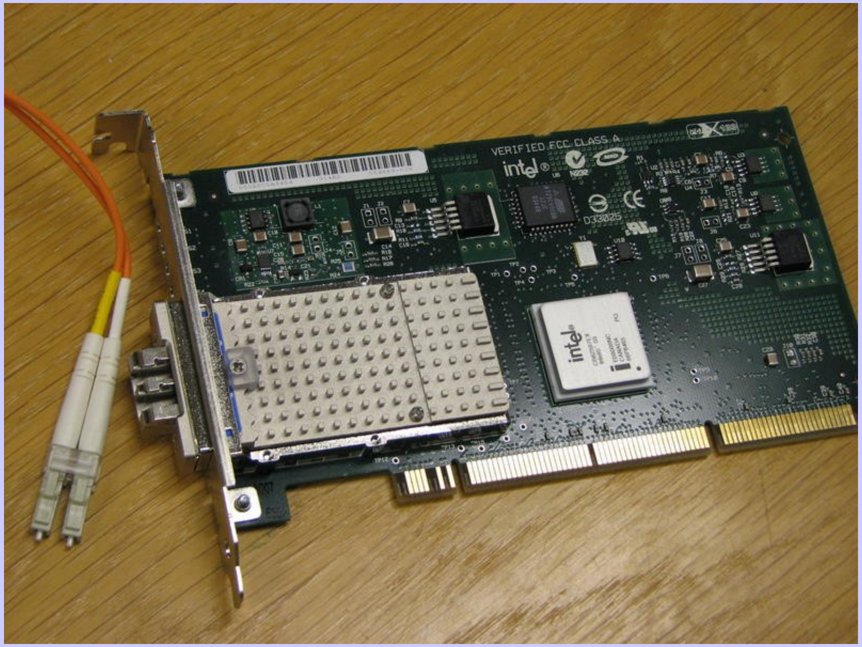

ORIXORIX4

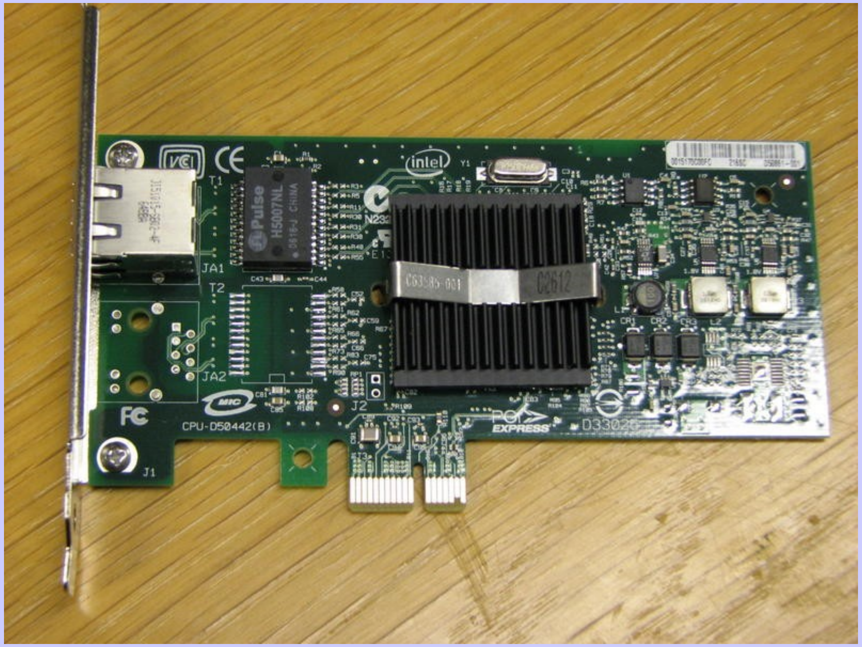

REGULUS
ORIXORIX3

IBM
IBM
ARCHIVE-1
ARCHIVE-2

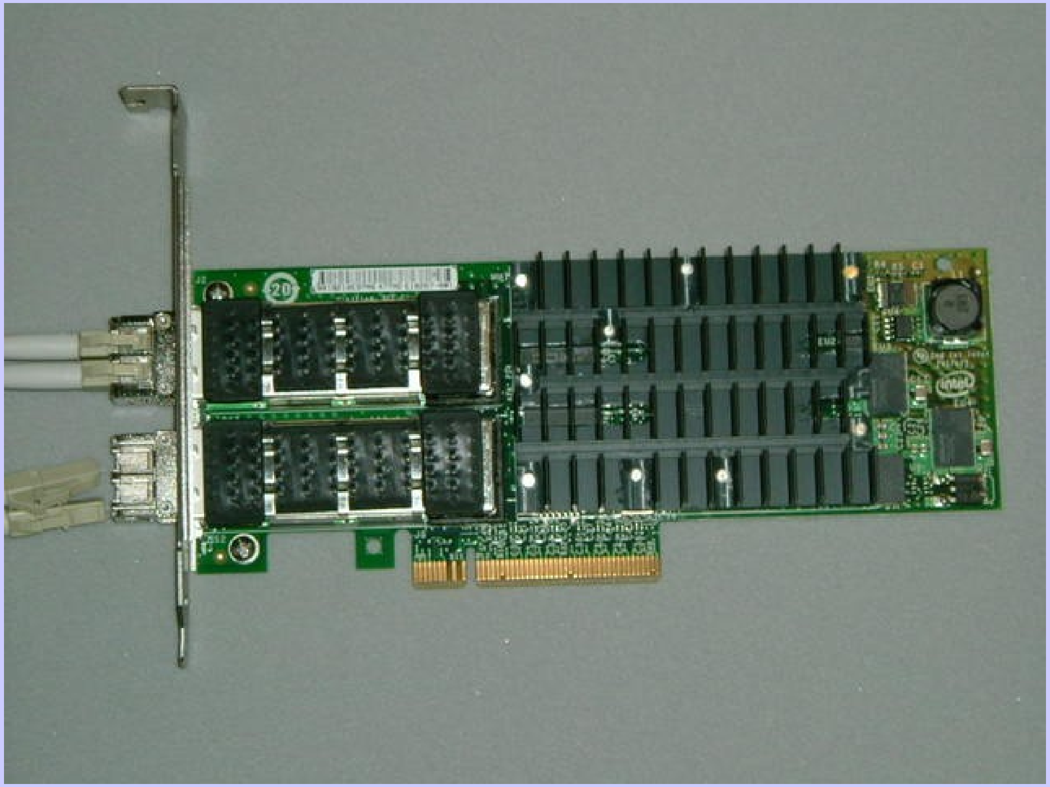
Intel NIC's


intel
CPU-D49919(B)
D33025
Pulse H5007NL
PCI EXPRESS


VERIFIED FCC CLASS A
intel
D33025


intel
Pulse H5007NL
CPU-D50442(B)
PCI EXPRESS
D33025

Latest & Greatest Hardware



Intel 10g board Chipset 82598

Latest & Greatest Hardware

2U Hi-End Opteron box

Not all were blessed...

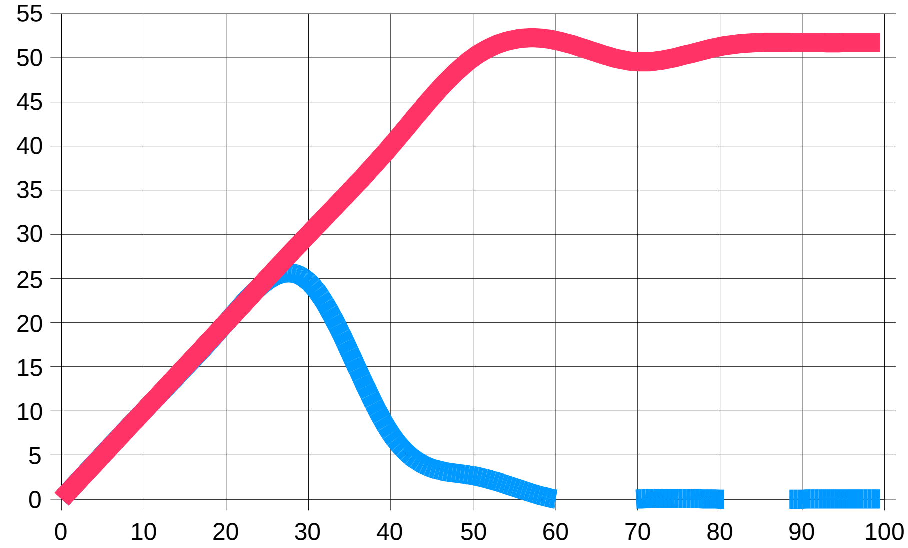
Bifrost concept

- Linux kernel collaboration
- Performance testing, development of tools and testing techniques
- Hardware validation, support from big vendors
- Detect and cure problems in lab not in the network infrastructure.
- Test deploy (Often in own network)

Overall Effect

- Inelegant handling of heavy net loads
  - System collapse
- Scalabiity affected
  - System and number of NICS
    - A single hogger netdev can bring the system to its knees and deny service to others

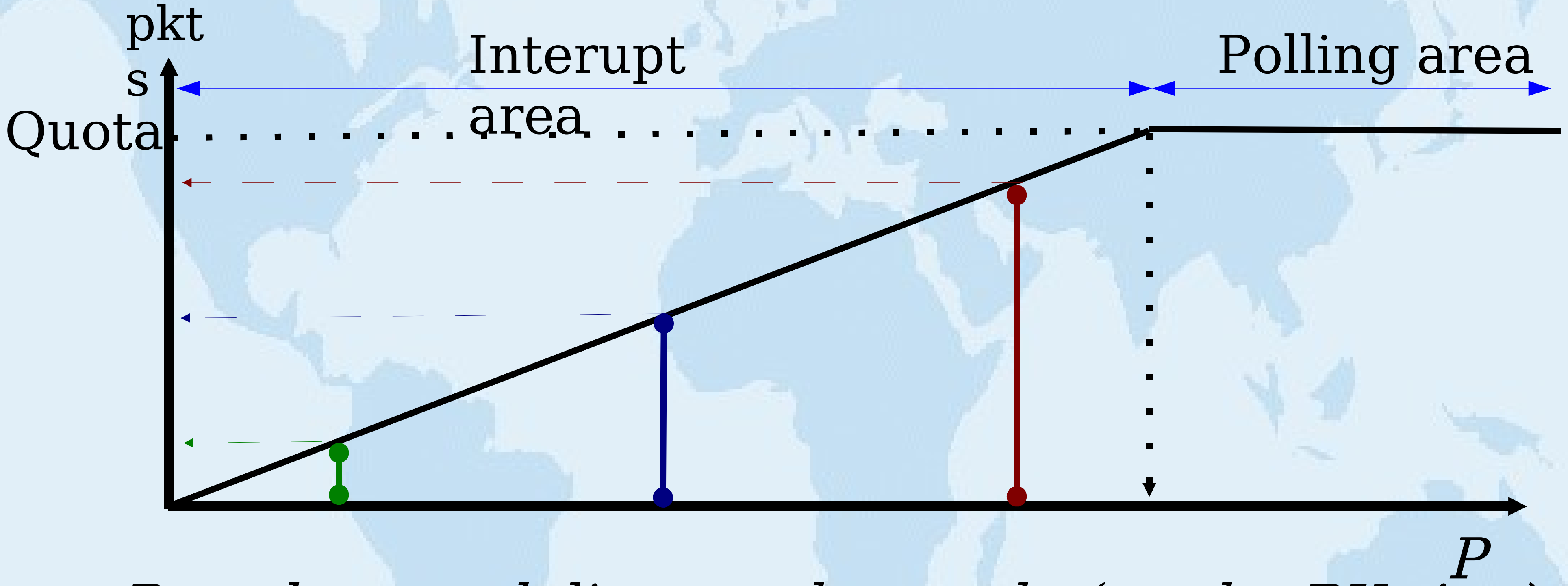
Summary 2.4 vs feedback



March 15 report on lkml

Thread: "How to optimize routing perfomance"
reported by
Marten.Wikstron@framsfab.se
- Linux 2.4 peaks at 27Kpps
- Pentium Pro 200, 64MB RAM

A high level view of new system



- ➔*P packets to deliver to the stack (on the RX ring)*
- ➔Horizontal line shows different netdevs with different i
- ➔Area under curve shows how many packets before next
- ➔*Quota* enforces fair share

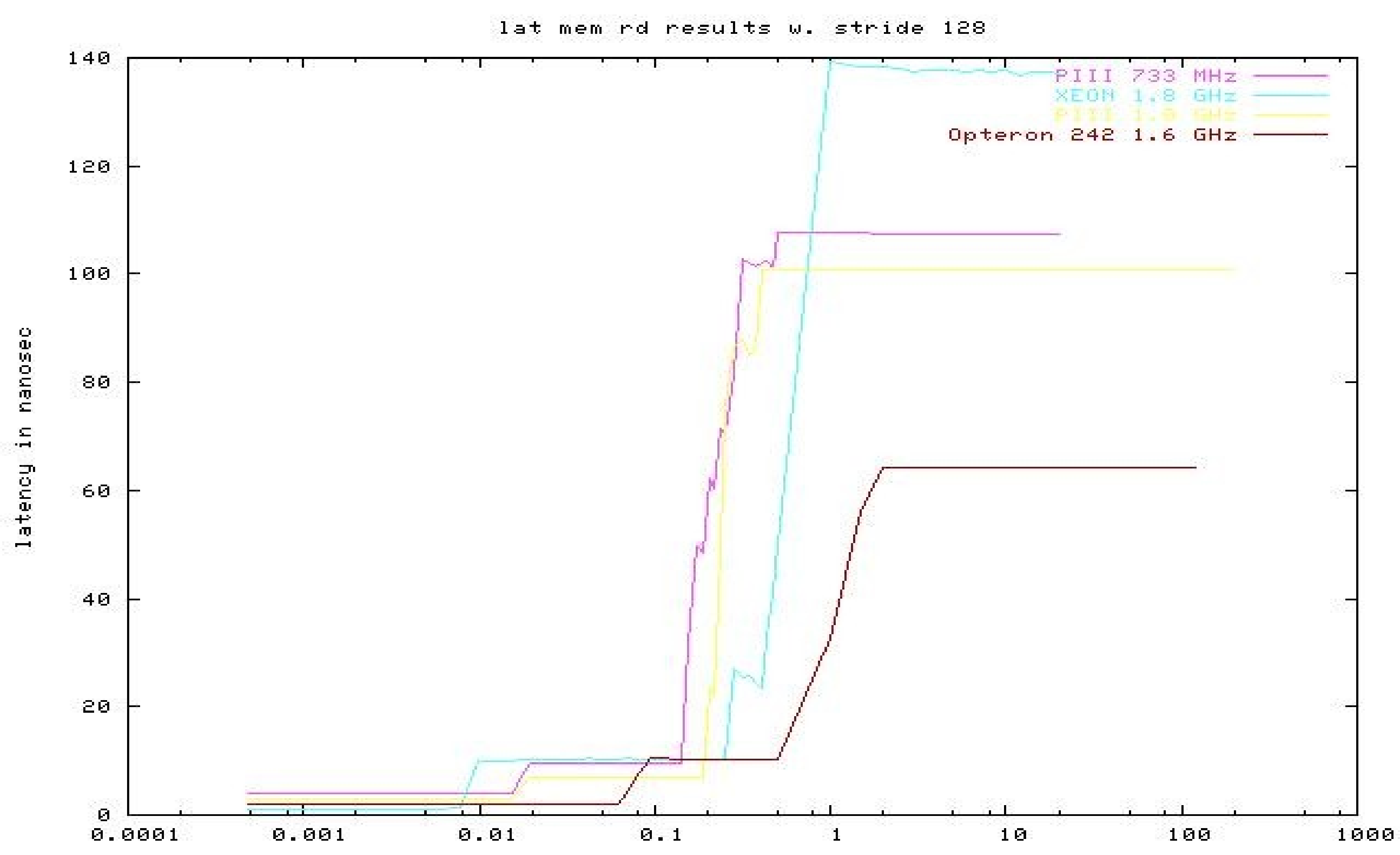
Kernel support

NAPI kernel part was included in:
2.5.7 and back ported to 2.4.20

Current driver support:

e1000 Intel GIGE NIC's
tg3   BroadCom GIGE NIC's
dl2k   D-Link GIGE NIC's
tulip (pending) 100 Mbs

Cache effect/Performance



Cache effect/Performance

Relative speed (Very approximative)

L1/L2 cache	1
Memory	1000
IO	10000

Mordern programming takes this into account.

Cache effect/Performance

Cache line 32 – 128 bytes

Optimize struct for cache and multiprocessors usage

PIO even worse then cache miss

PIO READ stalls CPU

PIO WRITE can be posted

DMA copies of data into RAM

Does prefetch solve problems?

A new network symbol has been seen…

The Penguin Has Landed