

About Me

- Jānis Meģis, MikroTik
- Jānis (Tehnnical, Trainer, NOT Sales)
 - Support & Training Engineer for almost 7 years
 - Specialization: QoS, PPP, Firewall, Routing
 - Teaching MikroTik RouterOS classes since 2005

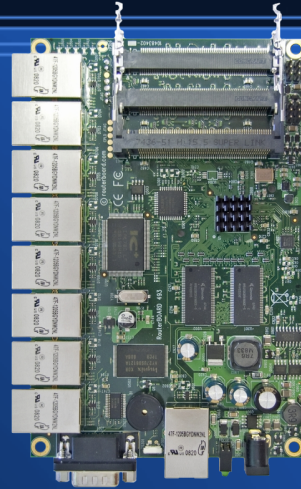
Hardware Performance and RouterOS

Phoenix, AZ
MUM USA 2010

Available Hardware Options

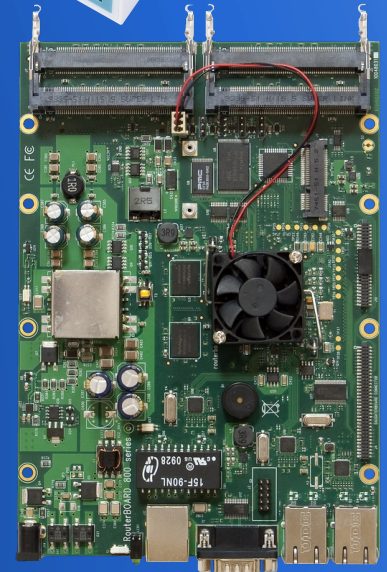
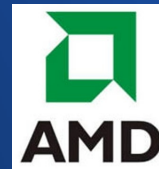
- CPE

- RB4xx series
- RB7xx series



- Core Routers

- X86 multicore solutions
- RB800/ RB1100



Router Performance

- Usually routers performance is measured in
 - Throughput - amount of traffic it is possible to process
 - Latency – delay that is created when packets travel through the router
- Both of these characteristics can be increased by:
 - Faster CPU
 - Faster memory
 - Hardware acceleration

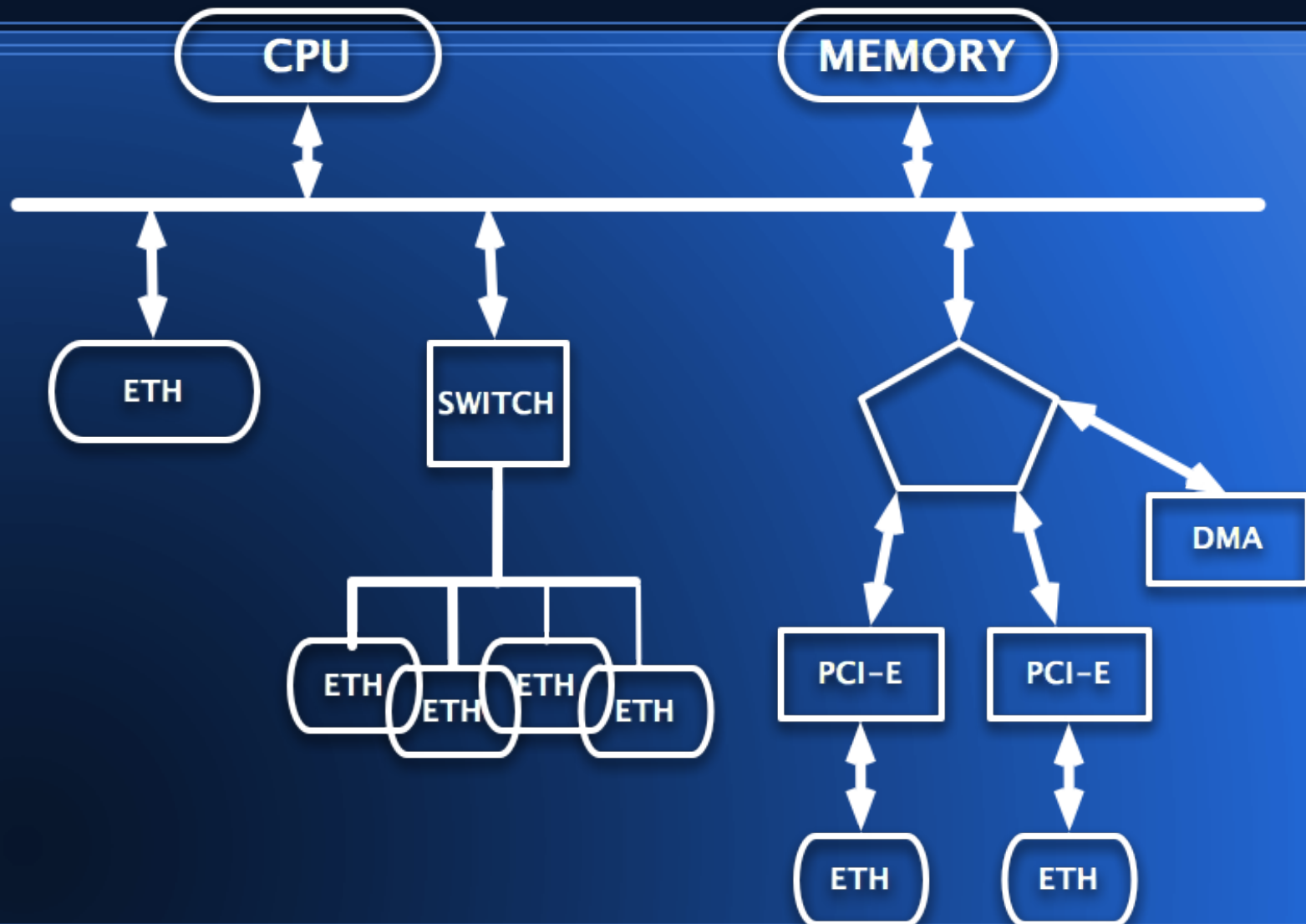
"David and Goliath"

- *“Why can a RouterBOARD with 1GHz CPU handle more traffic than my 3GHz PC???”*
- There can be several reasons:
 - RouterBOARDS have CPUs specifically designed for networking , x86 uses general purpose CPUs
 - hardware acceleration features (like IPSec acceleration or RB1000)
 - Interfaces....

Interfaces

- Directly connected to local bus
 - Have direct access to memory
 - PowerPC architecture allows to write packets directly to CPUs L2 memory
- Connected through the switch-chip
 - Allow wire speed communications between Ethernets in this switch group
- Connected via standard PCI-E
 - (standard way no shortcuts)

Ethernets



RouterBOARD Comparison

- RouterBOARD maximal Ethernet throughput comparison can be found here:
- http://www.routerboard.com/pdf/routerboard_performance_tests.pdf
- All tests was done
 - through the router
 - with minimal configuration
 - with Agilent N2X testing equipment

Guidelines for Comparison

- Results should be used to compare RouterBOARDS between each other not to production systems
 - 64 byte packet throughput reflect CPU performance
 - 1500 byte packet throughput reflects Memory performance
 - 512 byte packet reflects CPU/Memory combined performance

How to compare? (1)

- In simple bridge configuration with 64 byte packets:
 - RB7xx can handle 91000 pps
 - RB1xx can handle 14000 pps
- So RB7xx is 6-7x faster on installations with low configuration and high pps

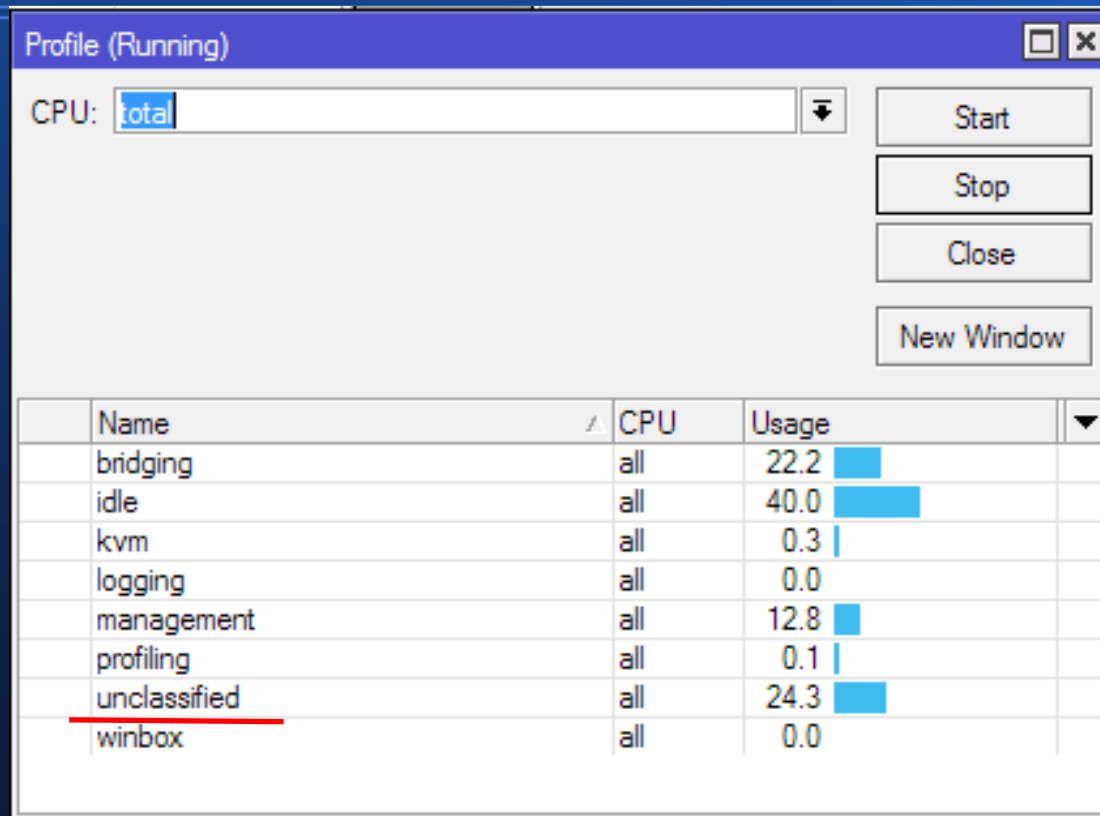
How to compare? (2)

- In bridge configuration with 64 byte packets with firewall and connection tracking enabled
 - RB7xx can handle 41600 pps
 - RB1xx can handle 4930 pps
- So as soon as we enable firewall and connection tracking (2 of most resource expensive options) RB7xx is 8-9x faster

Results of comparison

- From previous slides we can see
 - With small configuration RB7xx is 6+ times faster than RB1xx
 - As soon as configuration gets more complicated gap between board performance gets even larger
 - It is impossible to compare big packets as RB7xx meets with interface limitations (but potential can be seen in RB750G results)

CPU Load: Tools Profile(r)



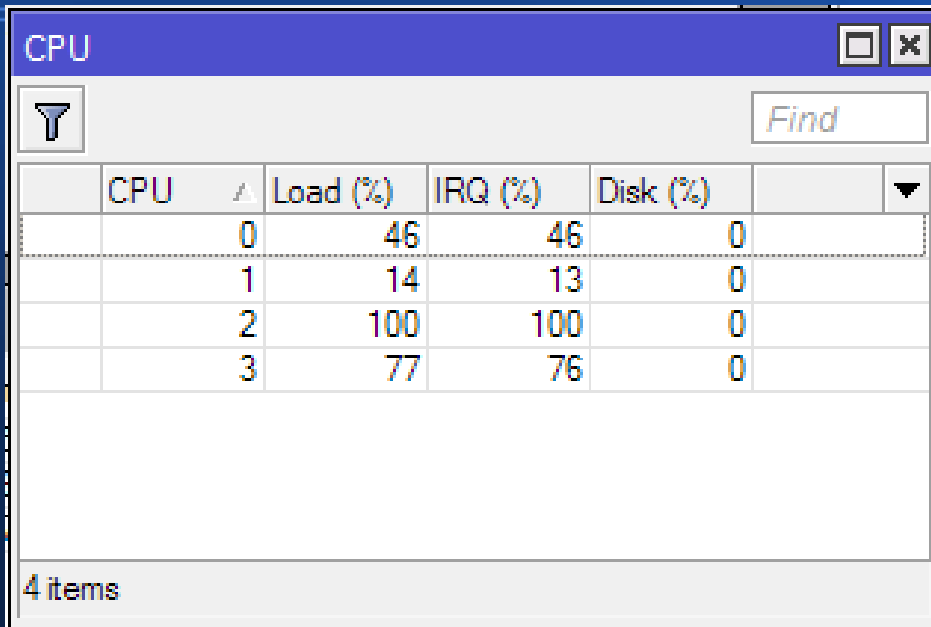
- RouterOS 5.x can report system usage by processes
- Work is still in progress

- Please, report all “unclassified” processes to support@mikrotik.com (with supout.rif)

CPU Usage

- Common **misconception** is that router works slower (bigger latency, more drops, less throughput) on 70% CPU load, then it does at 20%.
- As long as load is not 100% CPU can handle all that is thrown at it. So 1%, 17%, 50% or 98% - all the calculations are done without delays and there are no slowdowns because of CPU.
- Exceptions are multi-core systems

CPU Load per Core



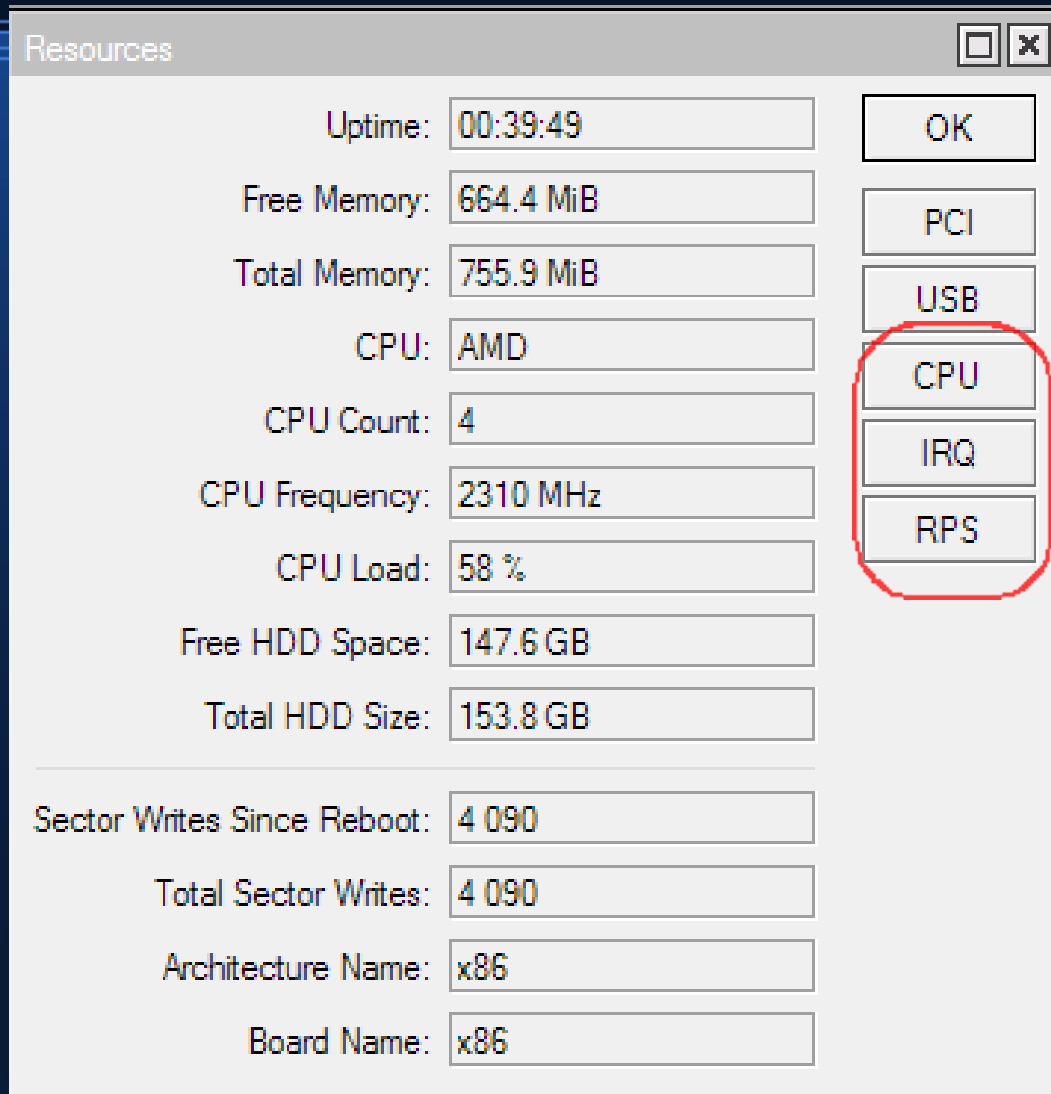
The screenshot shows a window titled 'CPU' with a filter icon and a 'Find' search box. Below is a table with columns: CPU, Load (%), IRQ (%), and Disk (%). The table contains four rows of data for CPU cores 0, 1, 2, and 3. Core 2 shows 100% load and 100% IRQ. The status bar at the bottom indicates '4 items'.

CPU	Load (%)	IRQ (%)	Disk (%)
0	46	46	0
1	14	13	0
2	100	100	0
3	77	76	0

- In RouterOS version 5.x multi-core routers can report CPU load per core

- At this point it is possible to recognize CPU usage from disk operation and interrupt request handling (most common usage)
- From CLI just use “/tool resources cpu print”

System Resources Menu



The screenshot shows the 'Resources' window in RouterOS. It displays system status information in a table-like format and a sidebar of controls. The 'CPU' control in the sidebar is highlighted with a red circle.

Label	Value
Uptime:	00:39:49
Free Memory:	664.4 MiB
Total Memory:	755.9 MiB
CPU:	AMD
CPU Count:	4
CPU Frequency:	2310 MHz
CPU Load:	58 %
Free HDD Space:	147.6 GB
Total HDD Size:	153.8 GB
<hr/>	
Sector Writes Since Reboot:	4 090
Total Sector Writes:	4 090
Architecture Name:	x86
Board Name:	x86

Controls (highlighted with a red circle):

- OK
- PCI
- USB
- CPU**
- IRQ
- RPS

- RouterOS version 5.x users will have much more control over their multi-core routers
- Controls can be found in “/system resources” menu.

IRQ Load Balancing

- It is possible to assign CPU cores to specific IRQ's
- Device distribution to IRQ's is done by hardware (not RouterOS)
- Auto mode works based on number of interrupts

The screenshot shows the 'IRQ' configuration window in RouterOS. It contains a table with the following columns: IRQ, Users, CPU, Active C..., and Count. The table lists various IRQs and their associated users and CPU assignments. A dialog box titled 'IRQ <66>' is open, showing the configuration for IRQ 66. The dialog includes fields for IRQ (66), Users (ether6), CPU (auto), Active CPU (2), and Count (40478). The 'OK' button is highlighted.

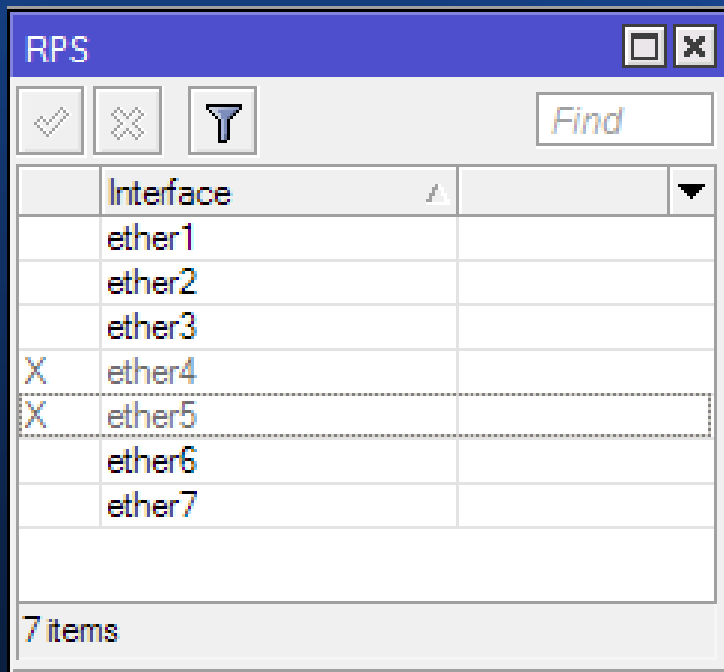
IRQ	Users	CPU	Active C...	Count
18	usb7, usb6, usb5	auto	2	0
19	usb2	a		
22	ahci	a		
48	ether7	a		
49	eth4-rx-0	a		
50	eth4-rx-1	a		
51	eth4-rx-2	a		
52	eth4-rx-3	a		
53	eth4-tx-0	a		
54	eth4-tx-1	a		
55	eth4-tx-2	a		
56	eth4-tx-3	a		
57	ether5	auto	2	34
58	eth5-rx-0	2	2	2302
59	eth5-rx-1	2	2	14033
60	eth5-rx-2	3	3	4678
61	eth5-rx-3	3	3	4676
62	eth5-tx-0	2	2	4682
63	eth5-tx-1	2	2	26117
64	eth5-tx-2	3	3	4759
65	eth5-tx-3	3	3	5365
66	ether6	auto	2	40478
67	ether1	0	0	0
68	ether2	1	1	90265
69	ether3	1	1	28420
70	ether4	0	0	82106

33 items (1 selected)

NAPI ("New API") feature

- NAPI is an optional modification to the device driver packet processing framework.
- NAPI allows drivers to run with significantly lower number of interrupts during times of high traffic.
- NAPI-compliant drivers can often cause packets to be dropped in the network adapter itself, before the Kernel sees them at all
- NAPI can force "auto" mode to use only one core

RPS: Receive Packet Steering



- NAPI can become a bottleneck under high packet load because is serialized per device queue
- RPS allows to distribute the load of received packet processing across multiple cores

Performance before optimization

- IRQ balancing = auto
- NAPI working (lots of packets)
- RPS disabled

The screenshot shows two windows from a network management application. The 'Interface List' window displays a table of network interfaces with their transmission and reception rates and packet counts. The 'CPU' window displays a table of CPU load, IRQ percentage, and disk usage for four processors.

	Name	Tx	Rx	Tx Packet (p/s)	Rx Packet (p/s)	Tx Dro
R	ether1	273.6 Mbps	273.5 Mbps	42 862	42 847	
R	ether2	273.6 Mbps	273.5 Mbps	42 860	42 854	
R	ether3	273.5 Mbps	273.5 Mbps	42 854	42 856	
R	ether4	273.5 Mbps	273.5 Mbps	42 841	42 856	
R	ether5	272.7 Mbps	272.1 Mbps	42 694	42 676	
R	ether6	41.9 kbps	5.5 kbps	6	8	
R	ether7	272.1 Mbps	272.6 Mbps	42 676	42 677	

7 items out of 14

CPU	Load (%)	IRQ (%)	Disk (%)
0	100	100	0
1	0	0	0
2	1	0	0
3	0	0	0

4 items

Performance after optimization

- IRQ balancing – static assignation (mostly)
- NAPI enabled (even more packets)
- RPS enabled only on interfaces with “RX drops”

The screenshot shows two windows from a network management tool. The 'Interface List' window displays a table of network interfaces with their transmission and reception rates and packet processing speeds. The 'CPU' window displays a table of CPU load, IRQ usage, and disk activity for four processors.

	Name	Tx	Rx	Tx Packet (p/s)	Rx Packet (p/s)	Tx D
R	ether1	500.6 Mbps	500.7 Mbps	78 134	78 163	
R	ether2	500.7 Mbps	501.1 Mbps	78 165	78 215	
R	ether3	500.0 Mbps	499.5 Mbps	78 027	77 973	
R	ether4	500.0 Mbps	500.3 Mbps	78 027	78 061	
R	ether5	497.0 Mbps	497.5 Mbps	77 814	77 745	
R	ether6	41.9 kbps	5.5 kbps	6	8	
R	ether7	497.5 Mbps	497.0 Mbps	77 746	77 815	

7 items out of 14

CPU	Load (%)	IRQ (%)	Disk (%)
0	87	88	0
1	90	88	0
2	95	95	0
3	91	91	0

4 items