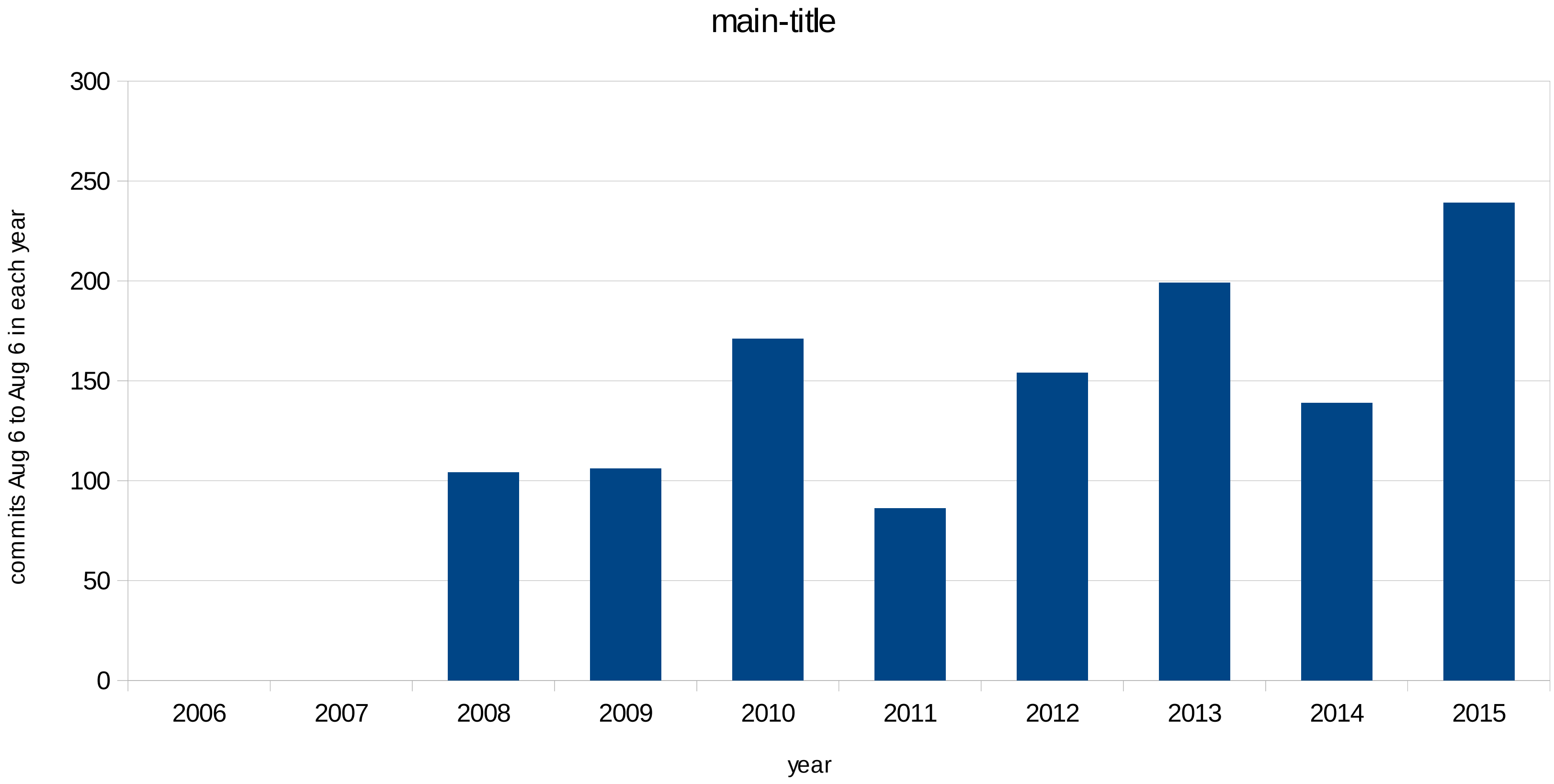
Virtio 1 - why do it?
And - are we there yet?

2015

Michael S. Tsirkin
Red Hat

Lots of work ...


main-title
commits Aug 6 to Aug 6 in each year
300
250
200
150
100
50
0
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
year

Virtio 1: update

- Documented assumptions
- More Robust
- More Extendable

Conformance statements

Virtio 0.9

- DRIVER_OK status bit is set.
- The device can now be used.

```
drv→probe(dev);
        netif_carrier_on(dev)
```

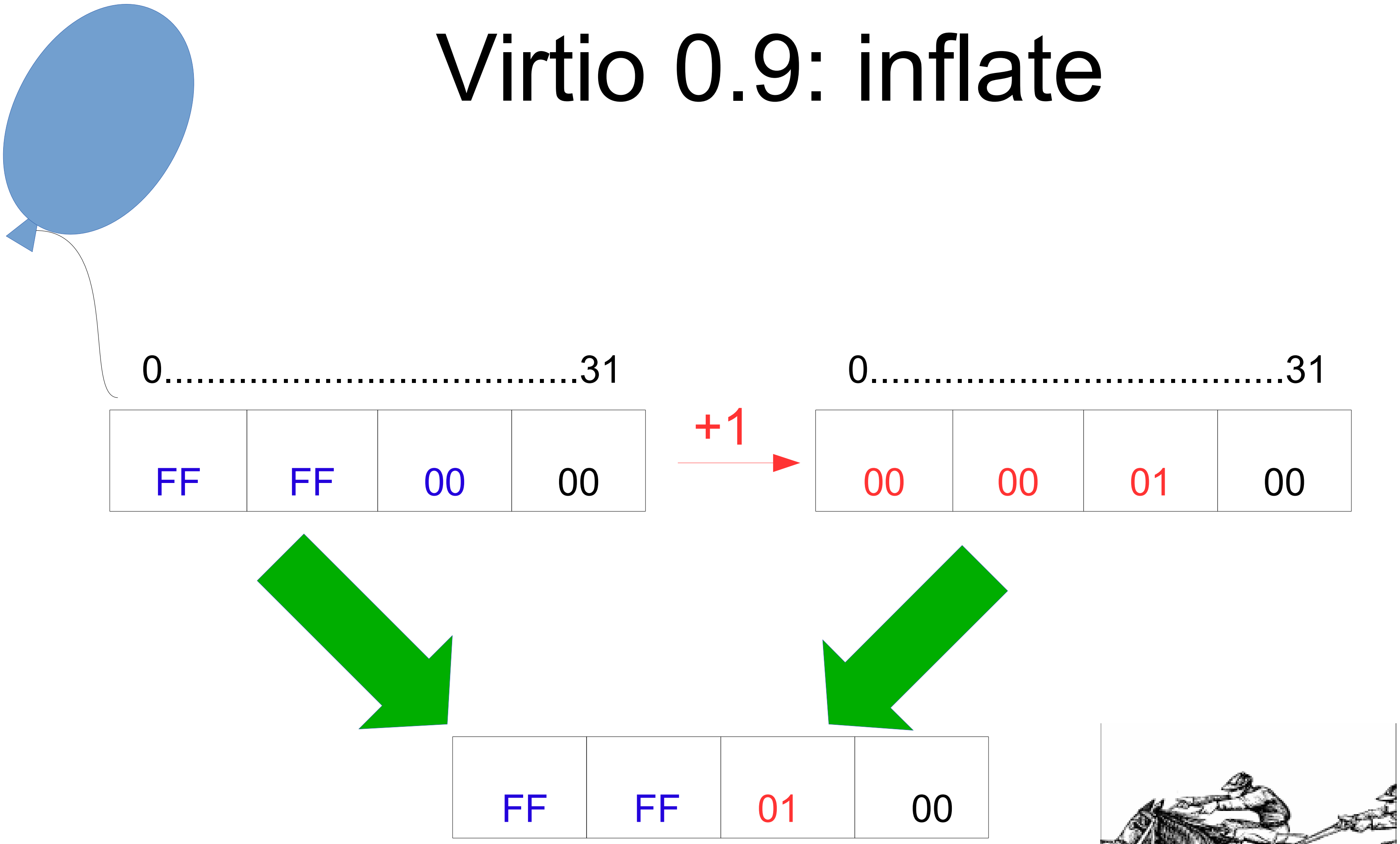
```
add_status(dev, DRIVER_OK);
```

Virtio 1.0

The driver **MUST NOT** notify the device before setting DRIVER_OK.

```
drv→probe(dev);
      add_status(dev, DRIVER_OK);
      netif_carrier_on(dev)
```

Virtio 0.9: inflate



0.....................................31

FF	FF	00	00

+1

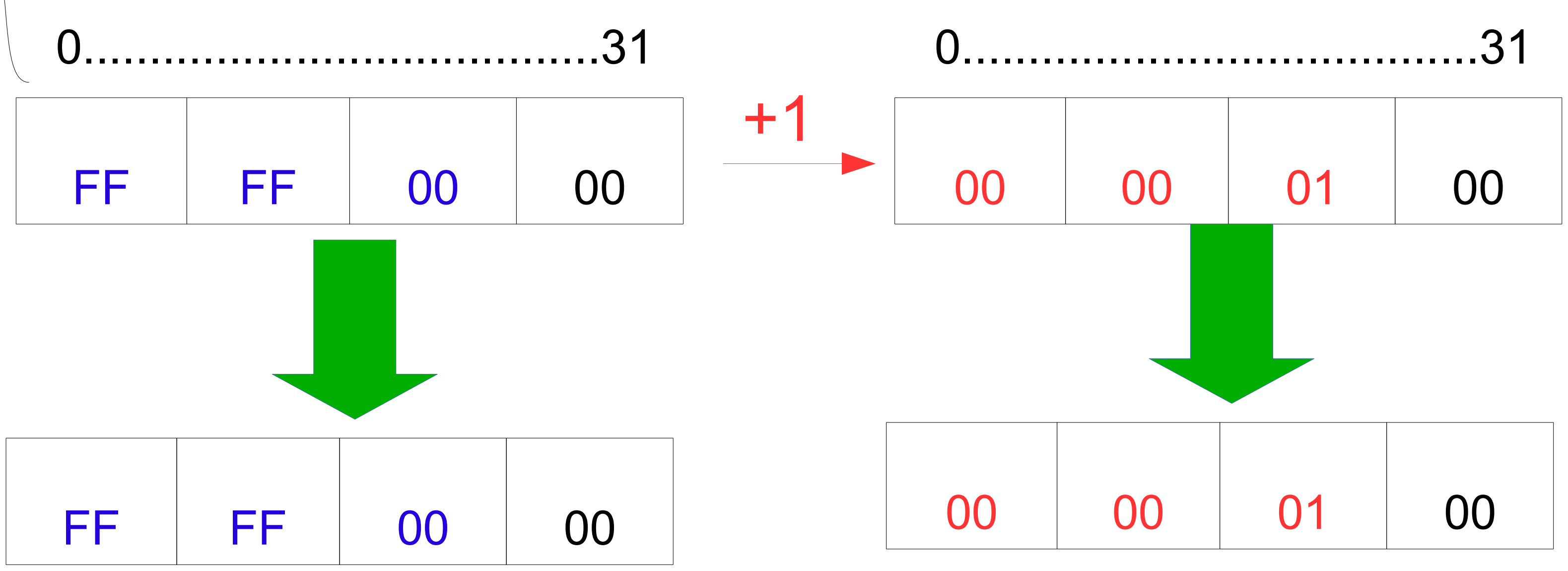
0.....................................31

00	00	01	00

FF	FF	01	00

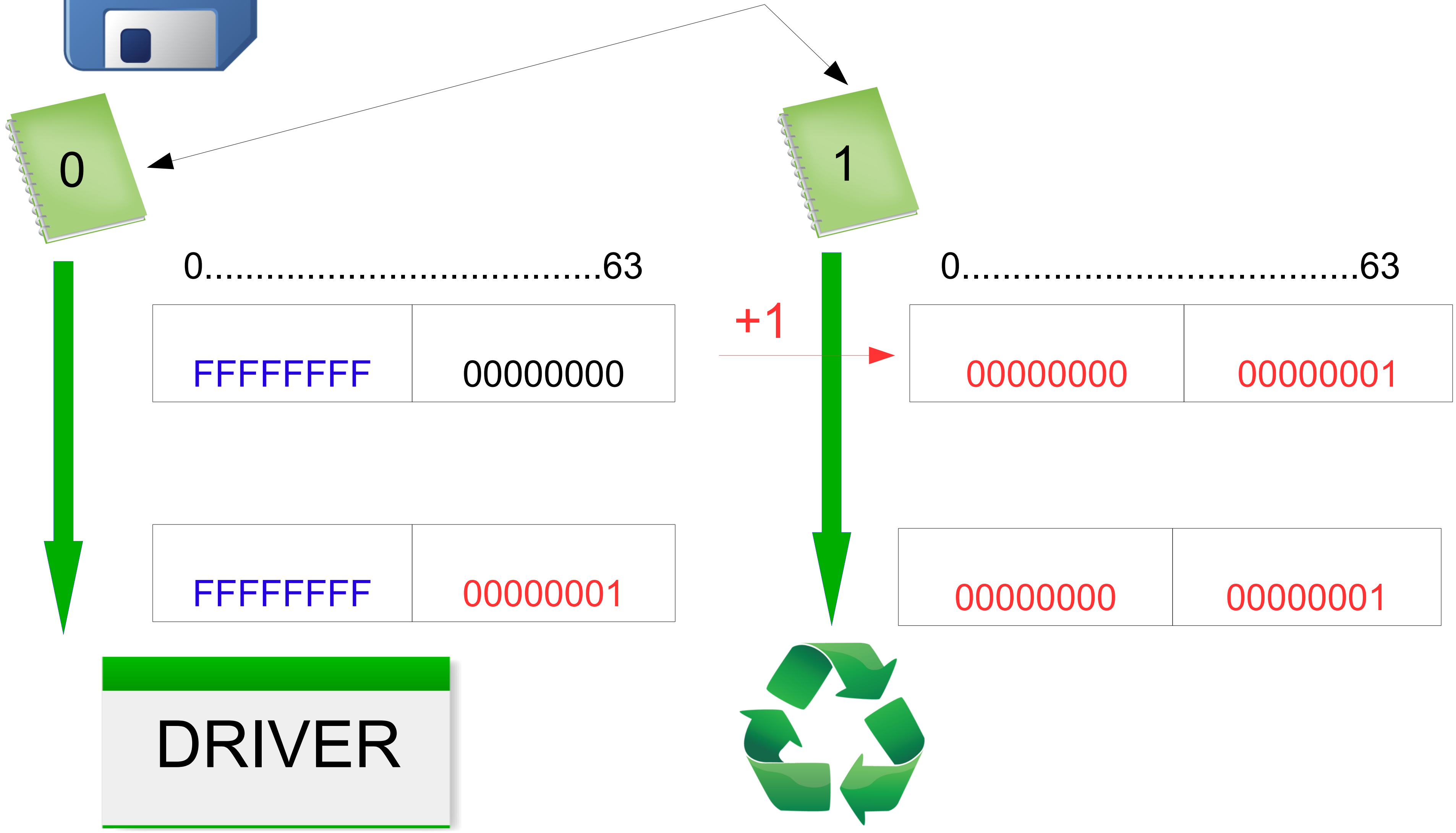
DRIVER

Virtio 1.0: inflate



DRIVER

Generation counter



0

1

0.....................................63

FFFFFFFF	00000000
FFFFFFFF	00000001

+1

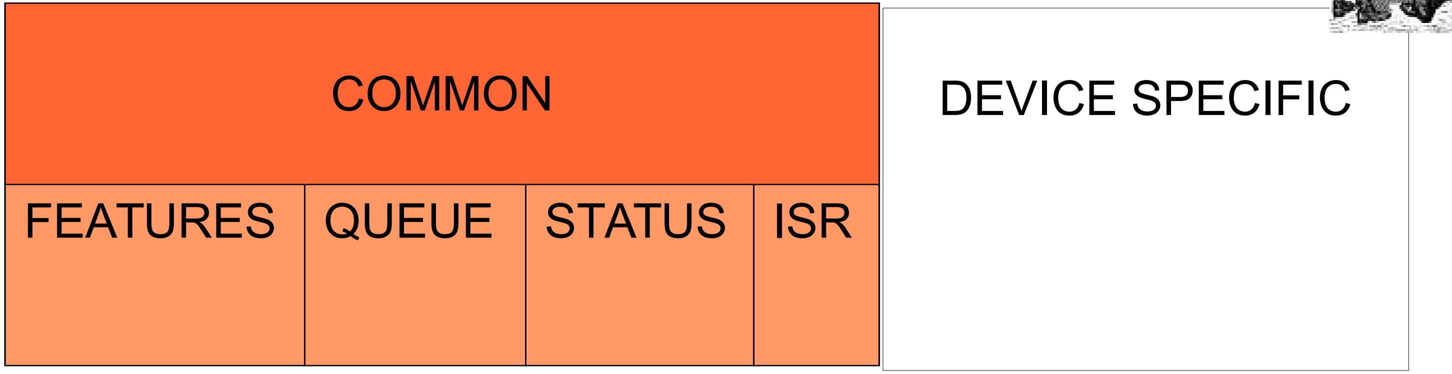
0.....................................63

00000000	00000001
00000000	00000001

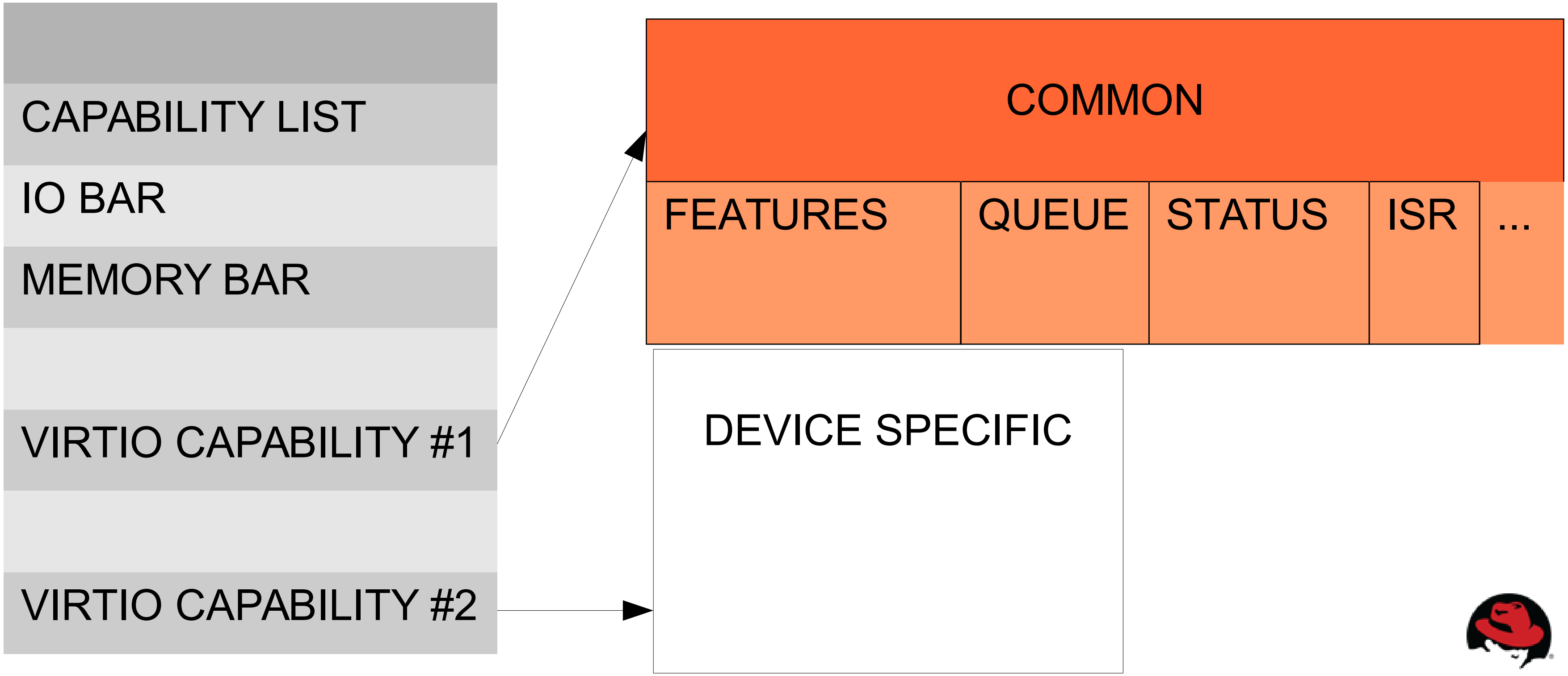
DRIVER

Memory map

0.9



1.0

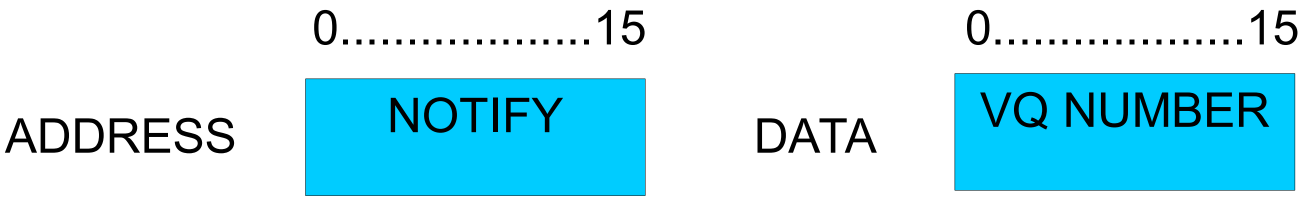


Virtio 0.9: Port IO vs Memory

	Port IO	MM IO
x86 decode: address	✓	✓
x86 decode: data	✓	✗
Fast on x86	✓	✗
32/64 bit	✗	✓
Page tables	✗	✓
Required by PCI Express	✗	✓

Fast MMIO
avoid need to decode data

0.9



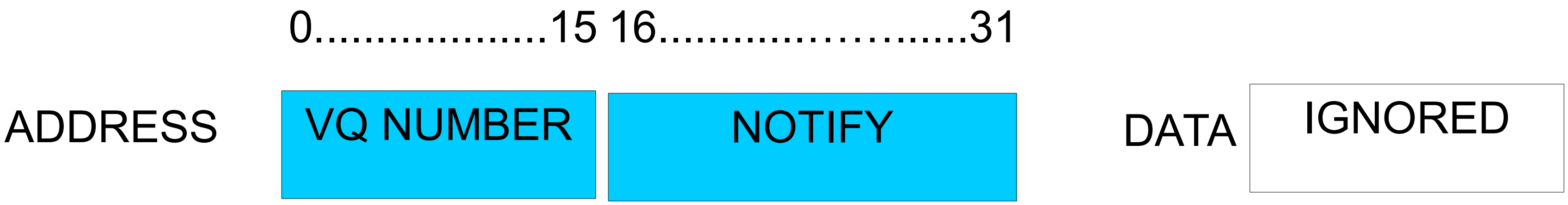
0...................15

ADDRESS NOTIFY

0...................15

DATA VQ NUMBER

1.0

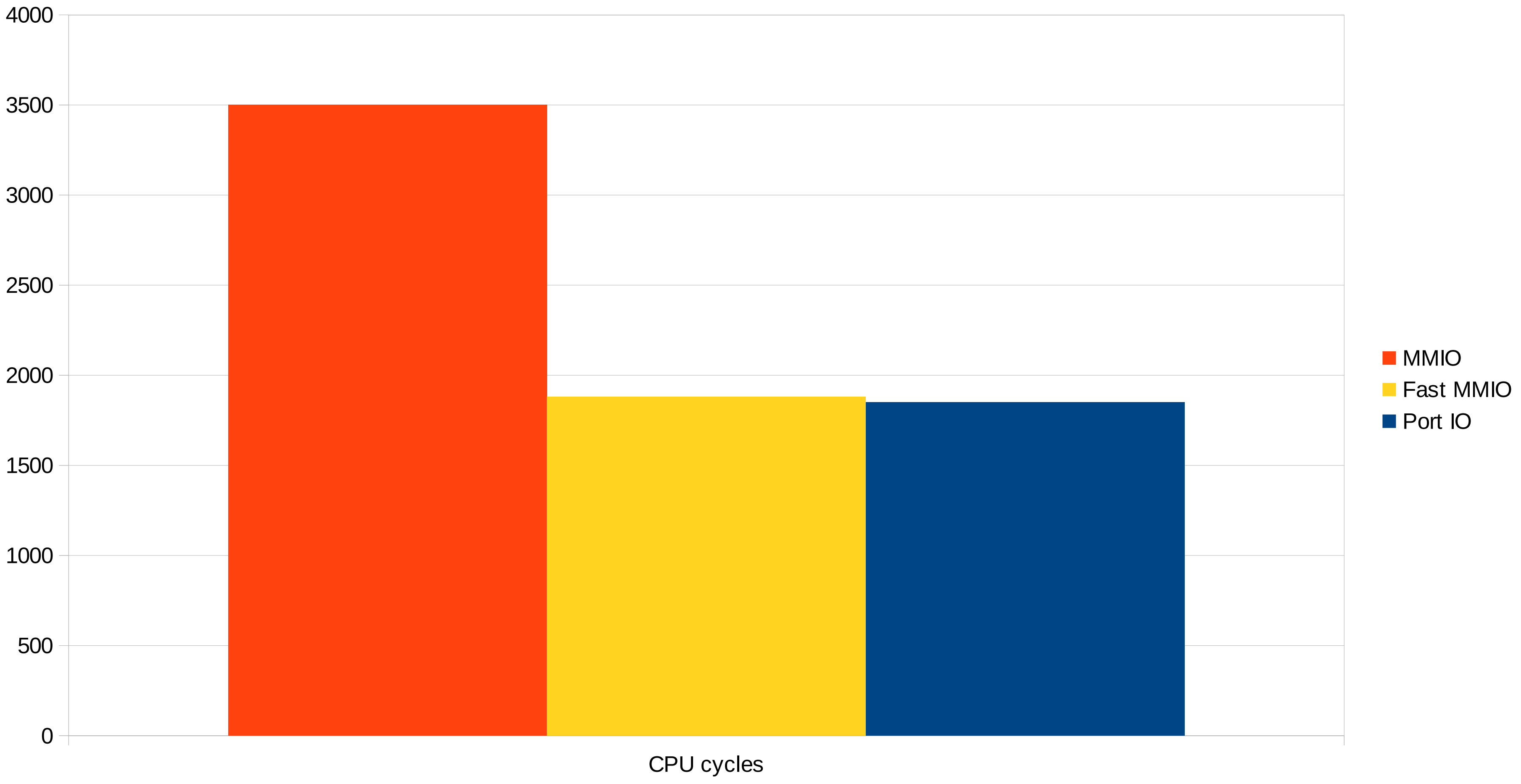


0...................15 16.........……......31

ADDRESS VQ NUMBER NOTIFY

DATA IGNORED

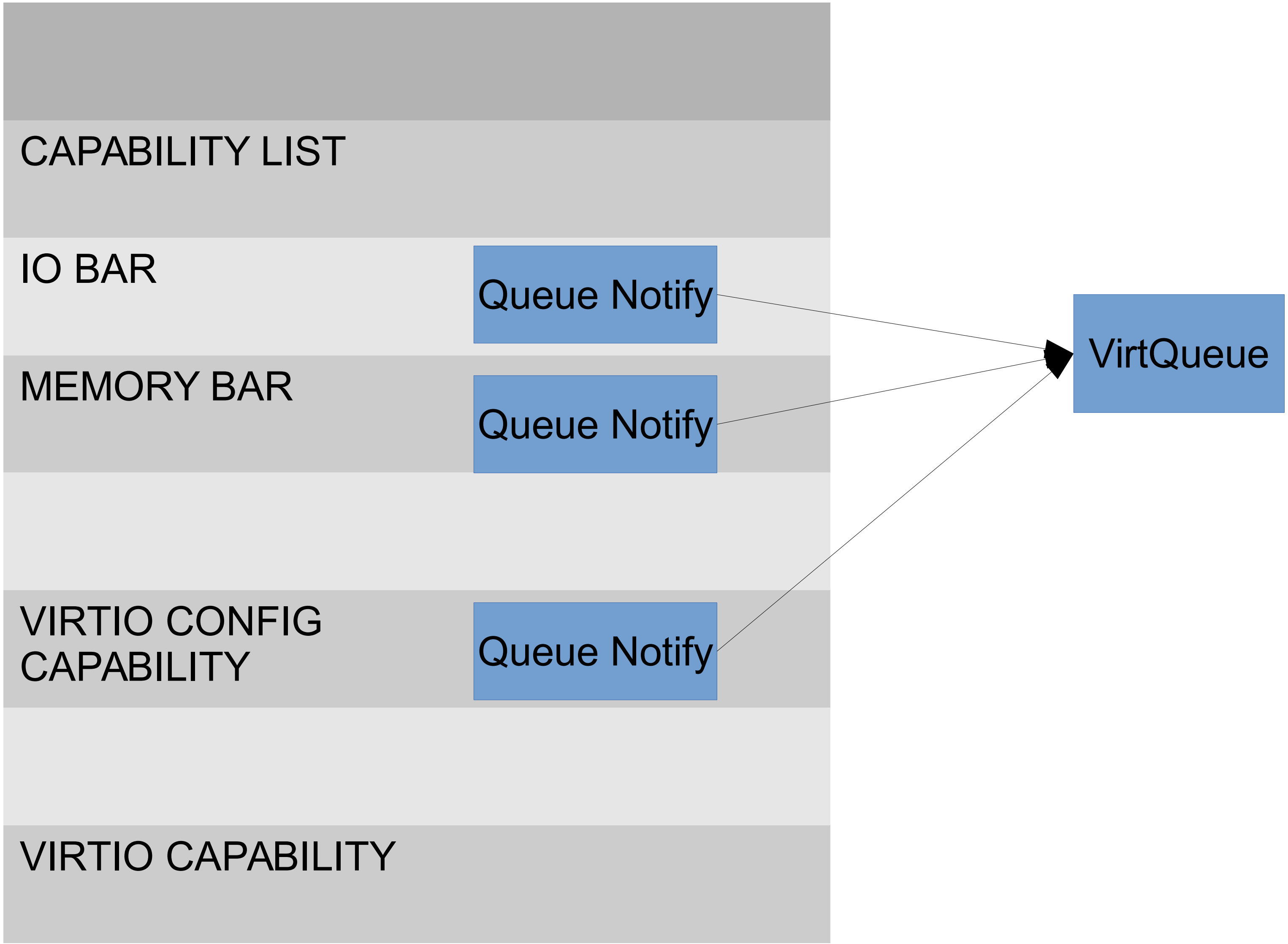
Virtio 1: Access times on KVM x86: Cycles per access (lower is better)



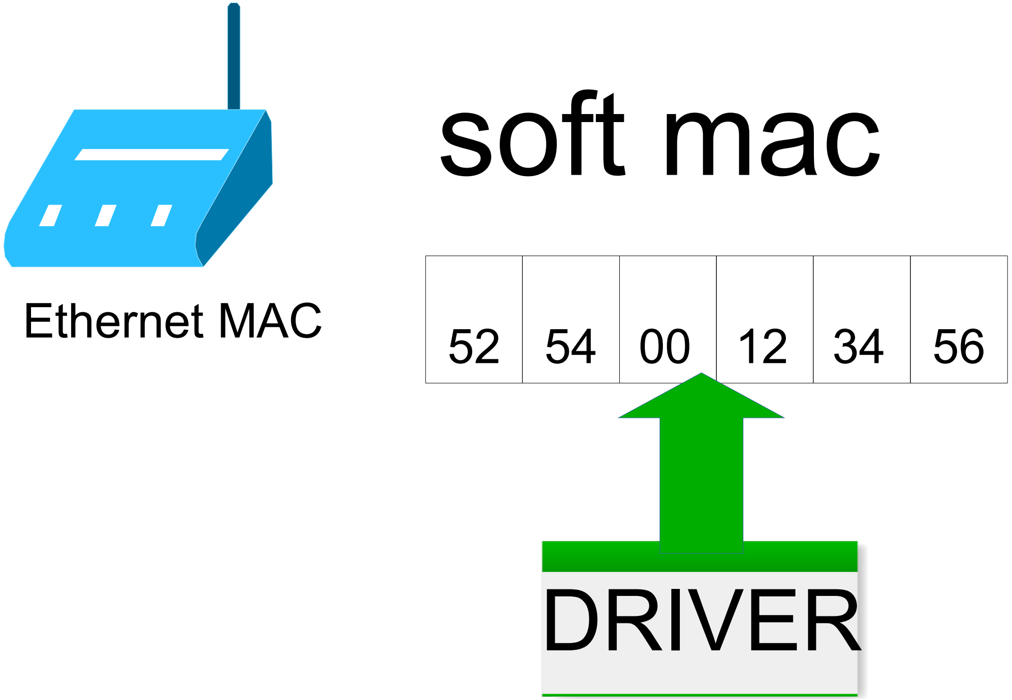
Virtio 1: Port IO vs Memory

	Port IO	MM IO
x86 decode: address	✓	✓
Fast on x86	✓	✓
32/64 bit	✗	✓
Page tables	✗	✓
Required by PCI Express	✗	✓

Memory Region Aliases



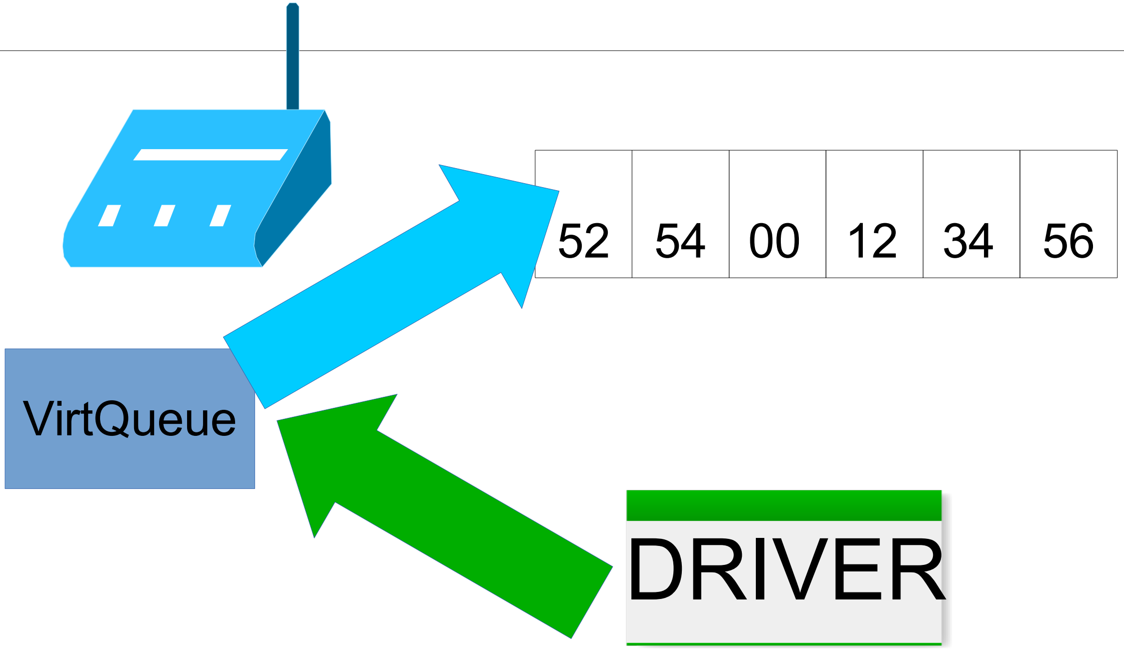
soft mac



52	54	00	12	34	56

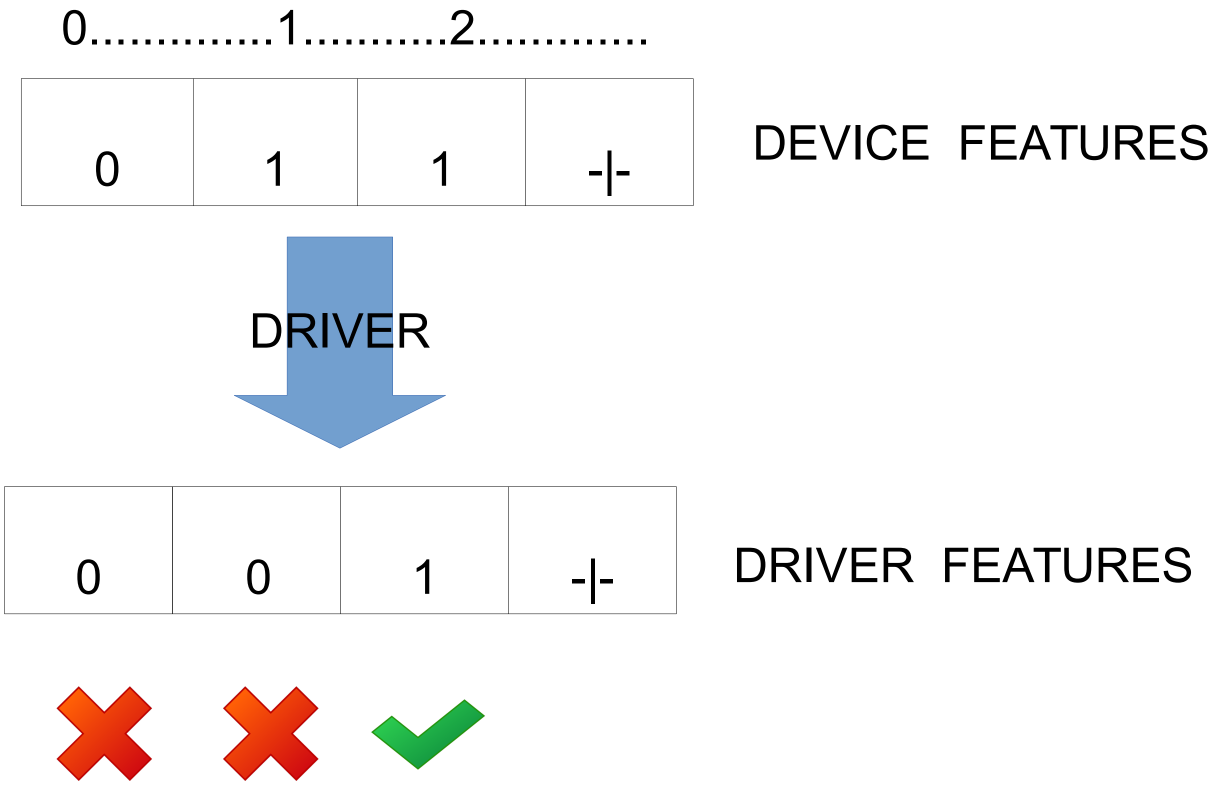
0.9

1.0



52	54	00	12	34	56

Virtio feature negotiation



0.............1..........2............

0	1	1	-|-	DEVICE FEATURES

DRIVER

0	0	1	-|-	DRIVER FEATURES
✗	✗	✓		

Defaults must be maintained forever!

Virtio 1: Error handling

- DRIVER: set features
- DRIVER: set FEATURES_OK bit

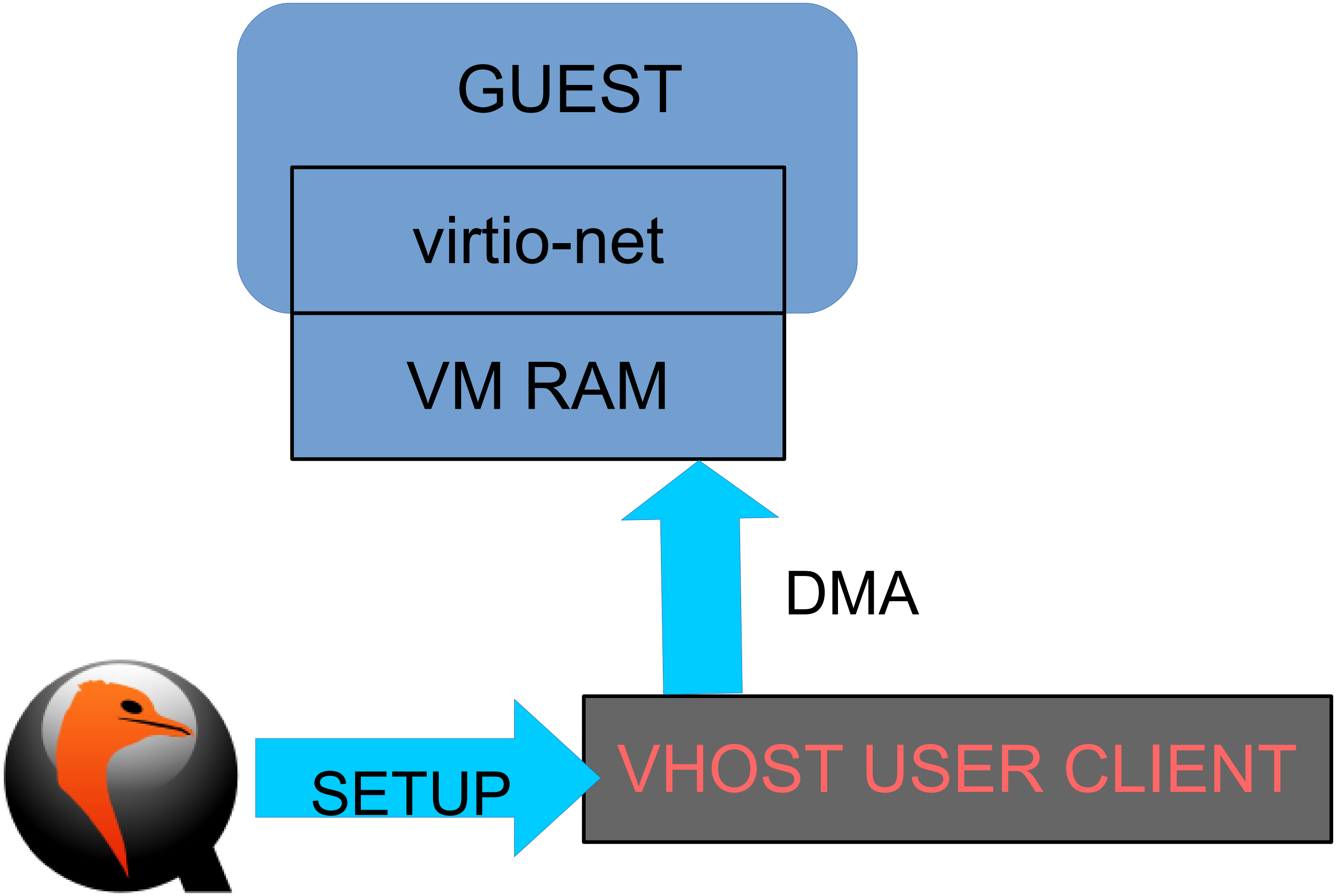
- DEVICE: check features
- DEVICE: clear FEATURES_OK on error

- DRIVER: check FEATURES_OK bit
- DRIVER: fail gracefully if not set !

Error handling: Virtio 0.9

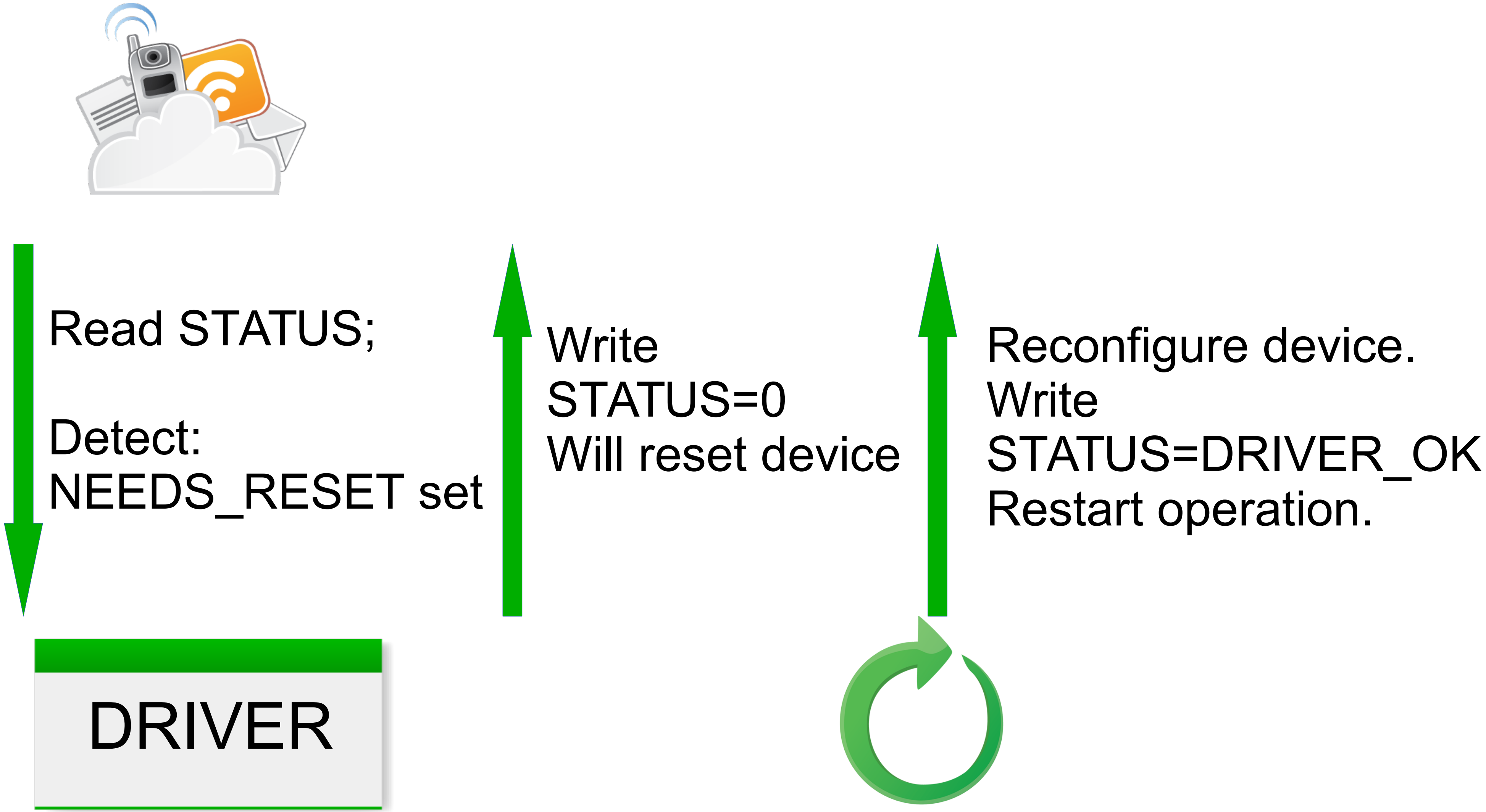
- Can't recover from device errors
- Not very useful?
- Just stop guest.

Vhost-user



Client crash or restart need not cause guest crash!

DEVICE_NEEDS_RESET



Read STATUS;

Detect:
NEEDS_RESET set

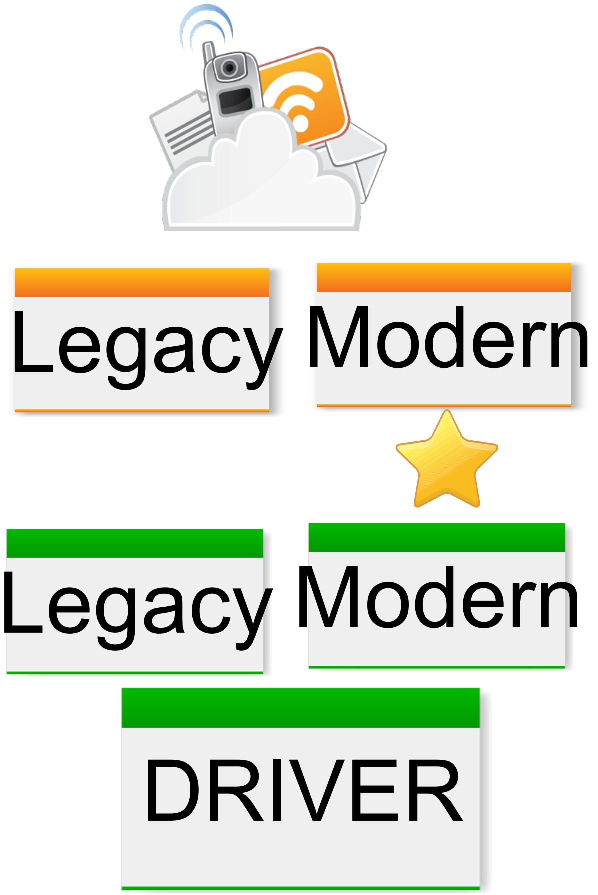
Write
STATUS=0
Will reset device

Reconfigure device.
Write
STATUS=DRIVER_OK
Restart operation.

DRIVER

Compatibility

Transitional Device & Driver

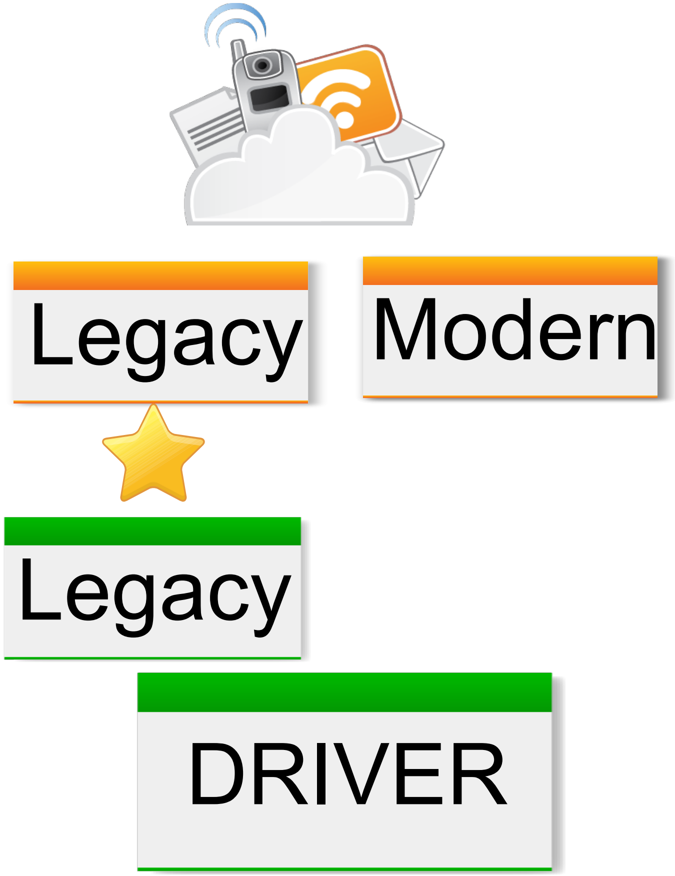


Legacy Modern

Legacy Modern

DRIVER

Legacy Driver

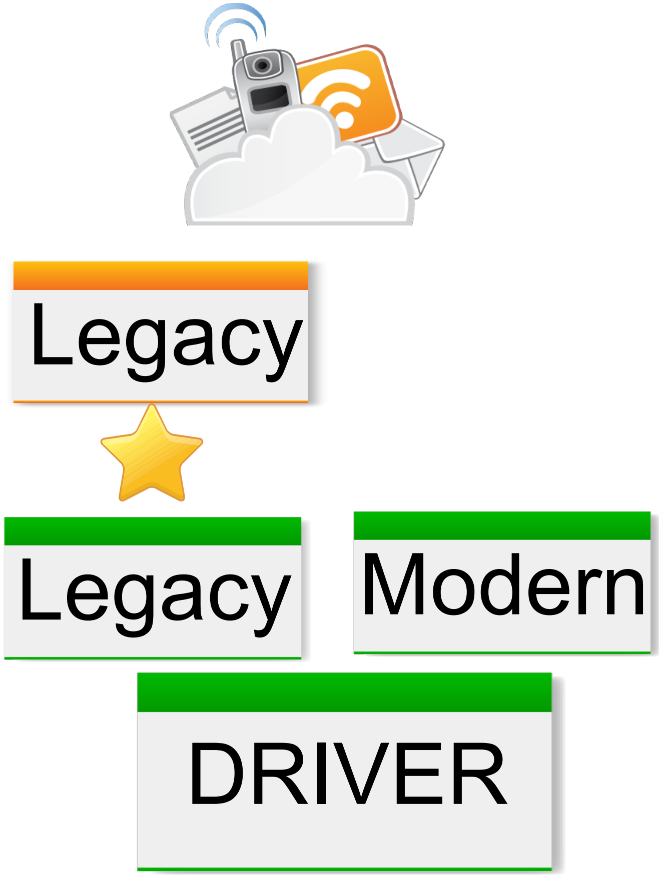


Legacy Modern

Legacy

DRIVER

Legacy Device

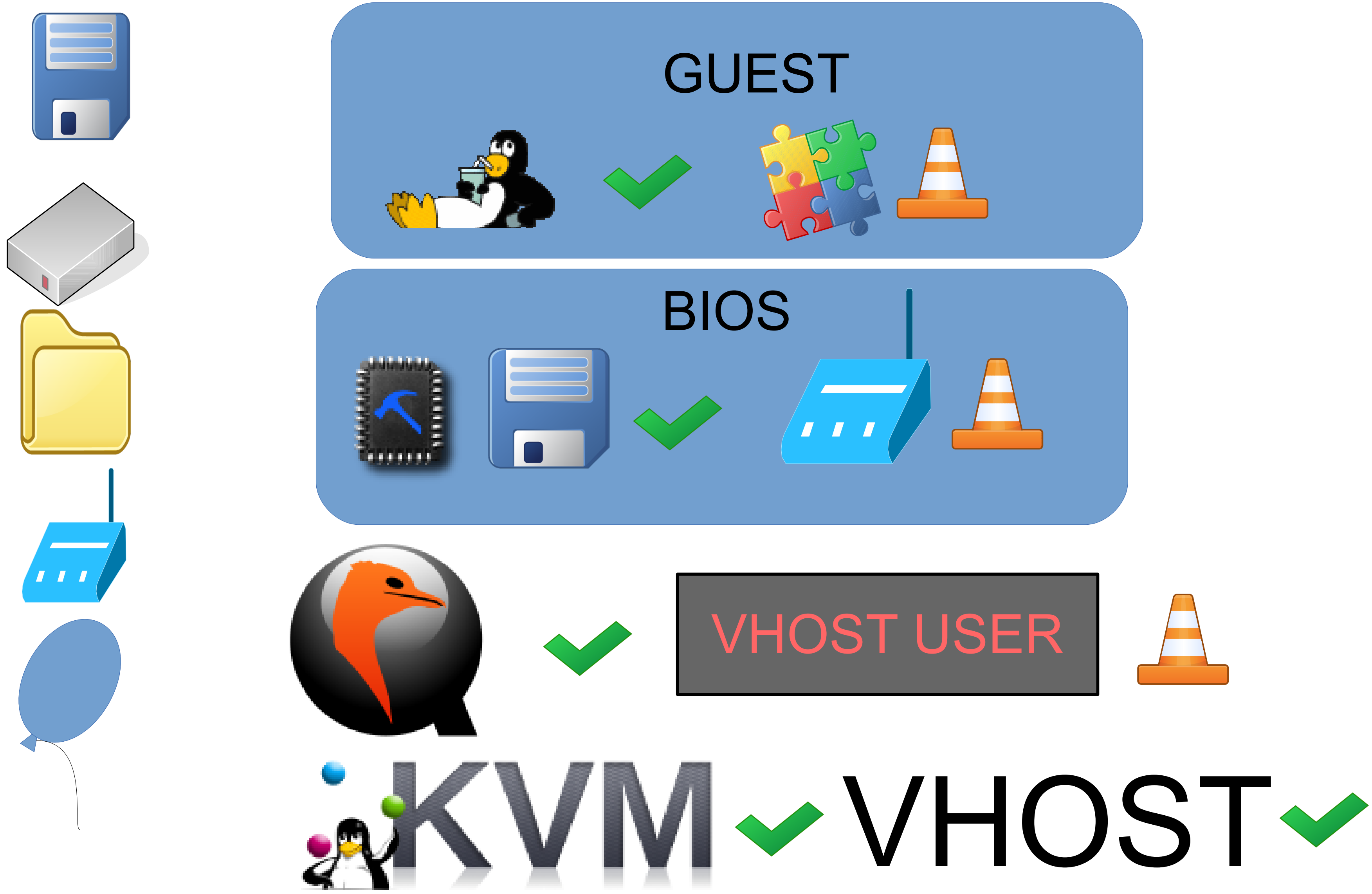


Legacy

Legacy Modern

DRIVER

Are we there yet?



GUEST ✓

BIOS ✓

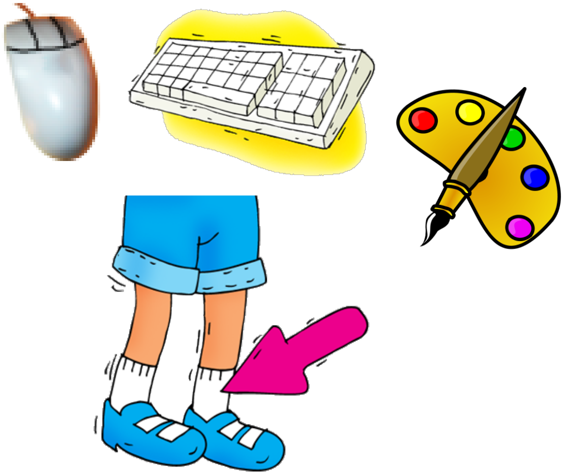
✓ VHOST USER

KVM ✓

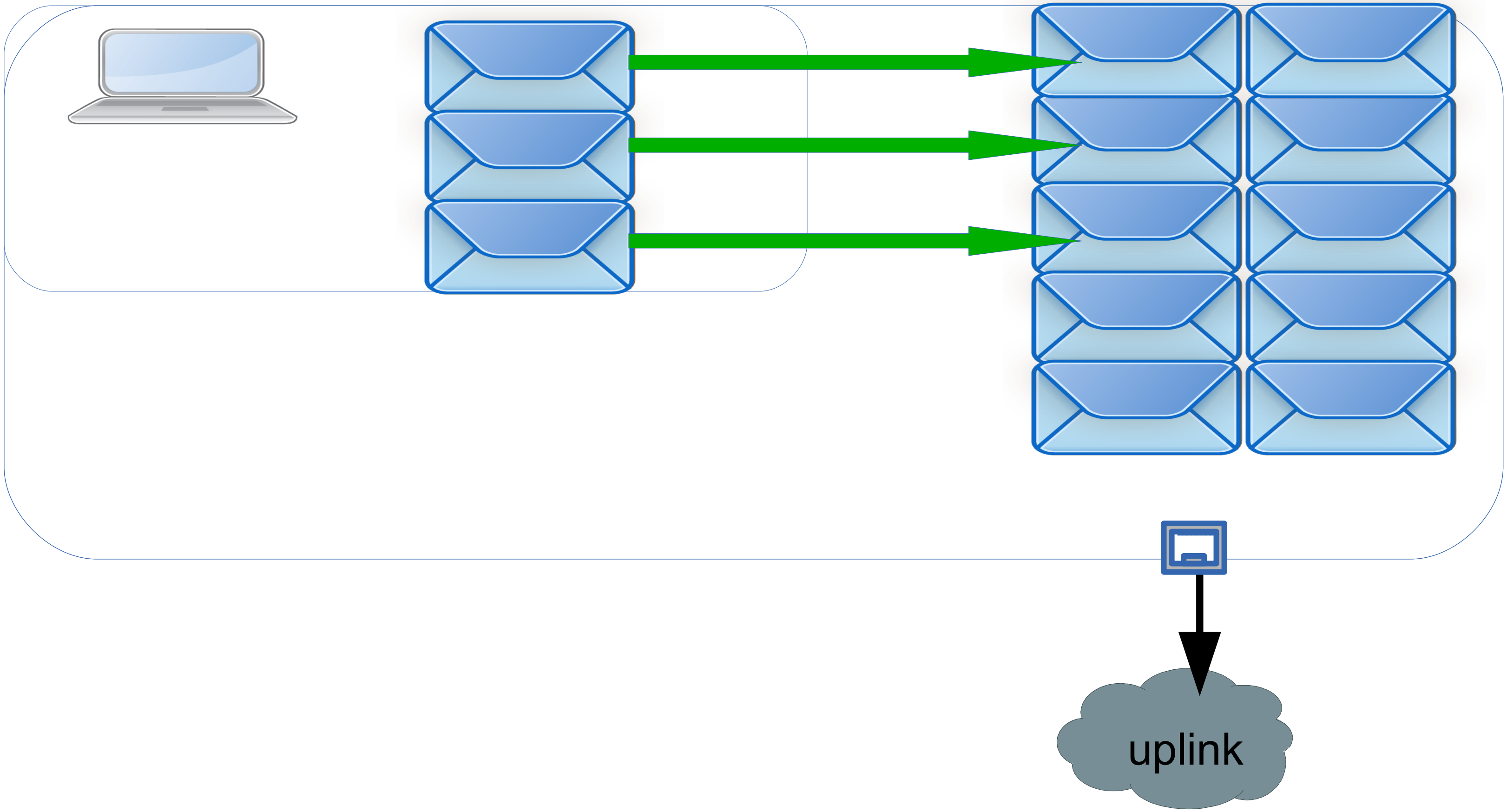
VHOST ✓

What to expect?

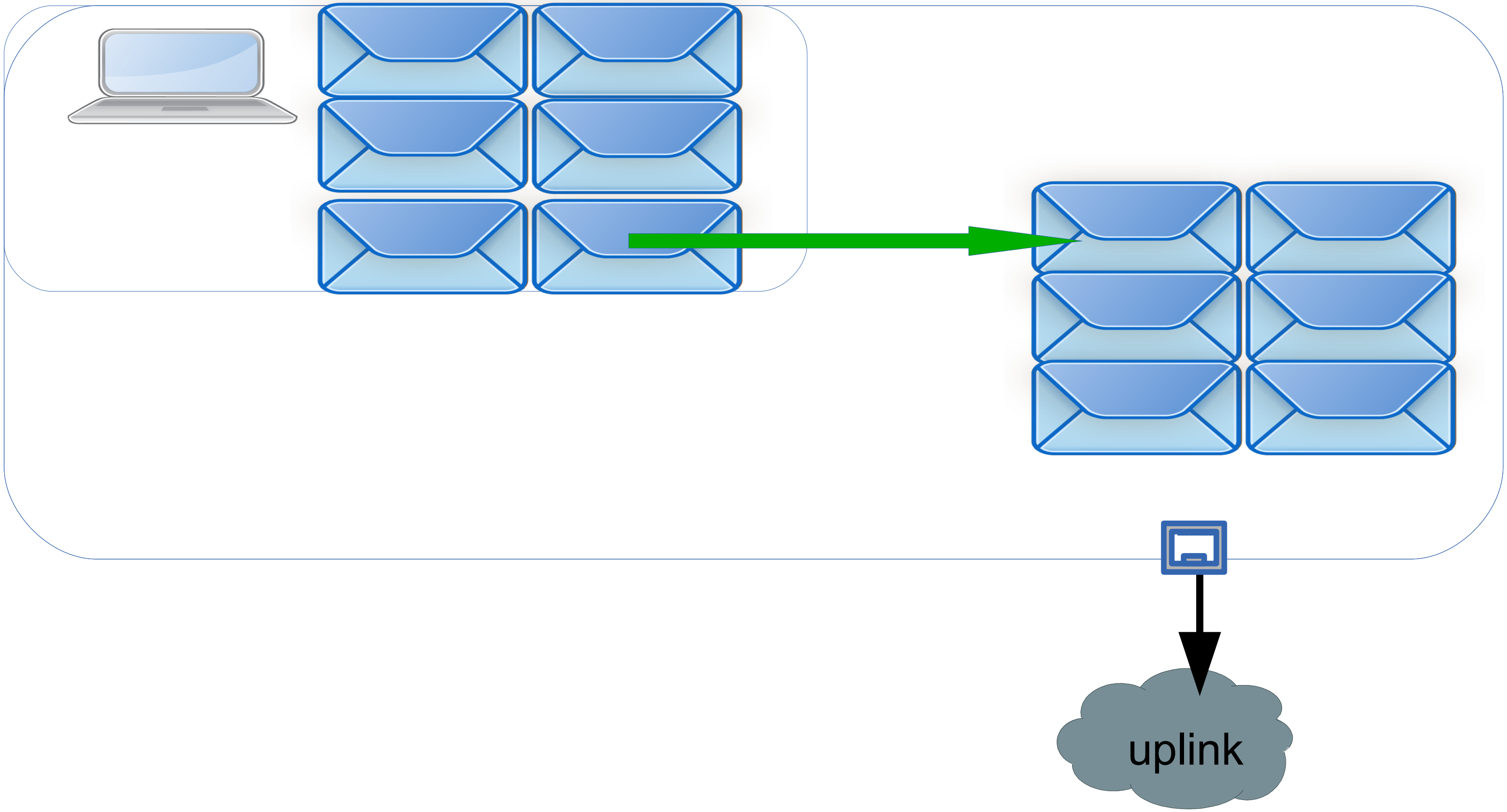
- Current: Virtio-v1.0-cs03
- Next bugfix: Virtio-v1.0-cs04
  - Virtio-blk: writeback / writethrough control
  - More update guidance
- Next feature: Virtio-v1.1-cs01
  - Virtio-input
  - Virtio-gpu
  - Virtio-vsock



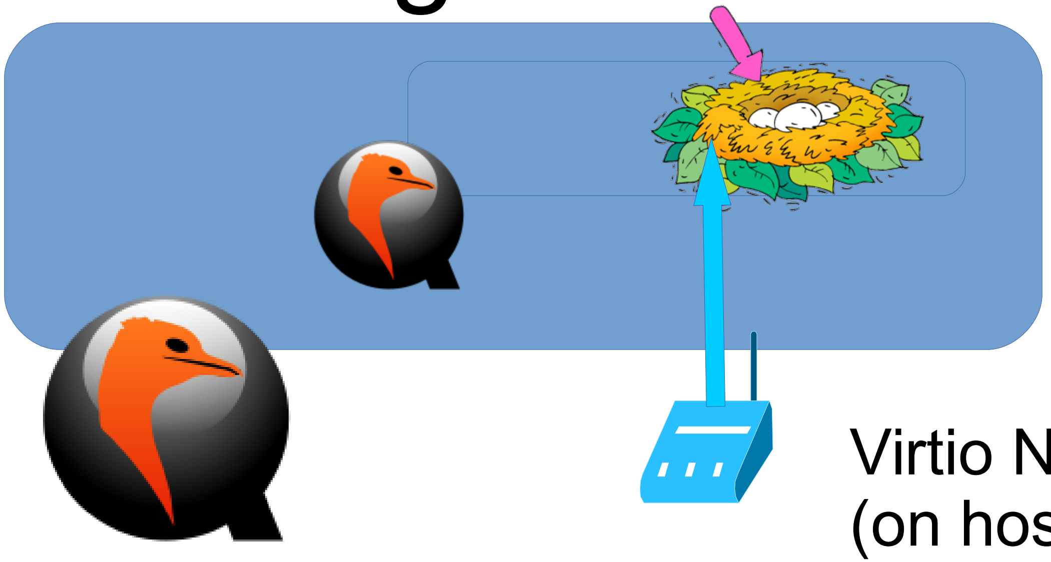
TX: Interrupt avoidance



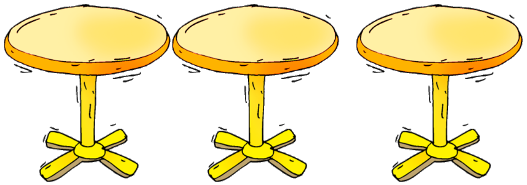
TX: Interrupt coalescing



Pass-through for nested virt



- Memory mapped: use page tables



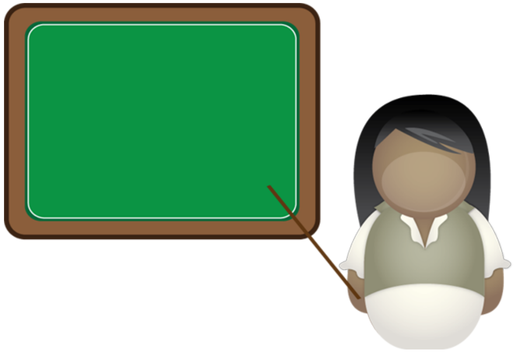
- IOMMU: translate and protect guest memory

Virtio as PCI Express device

- Uses memory mapped IO support
- Multi-root for NUMA
- Native hotplug
- Advanced Error Reporting

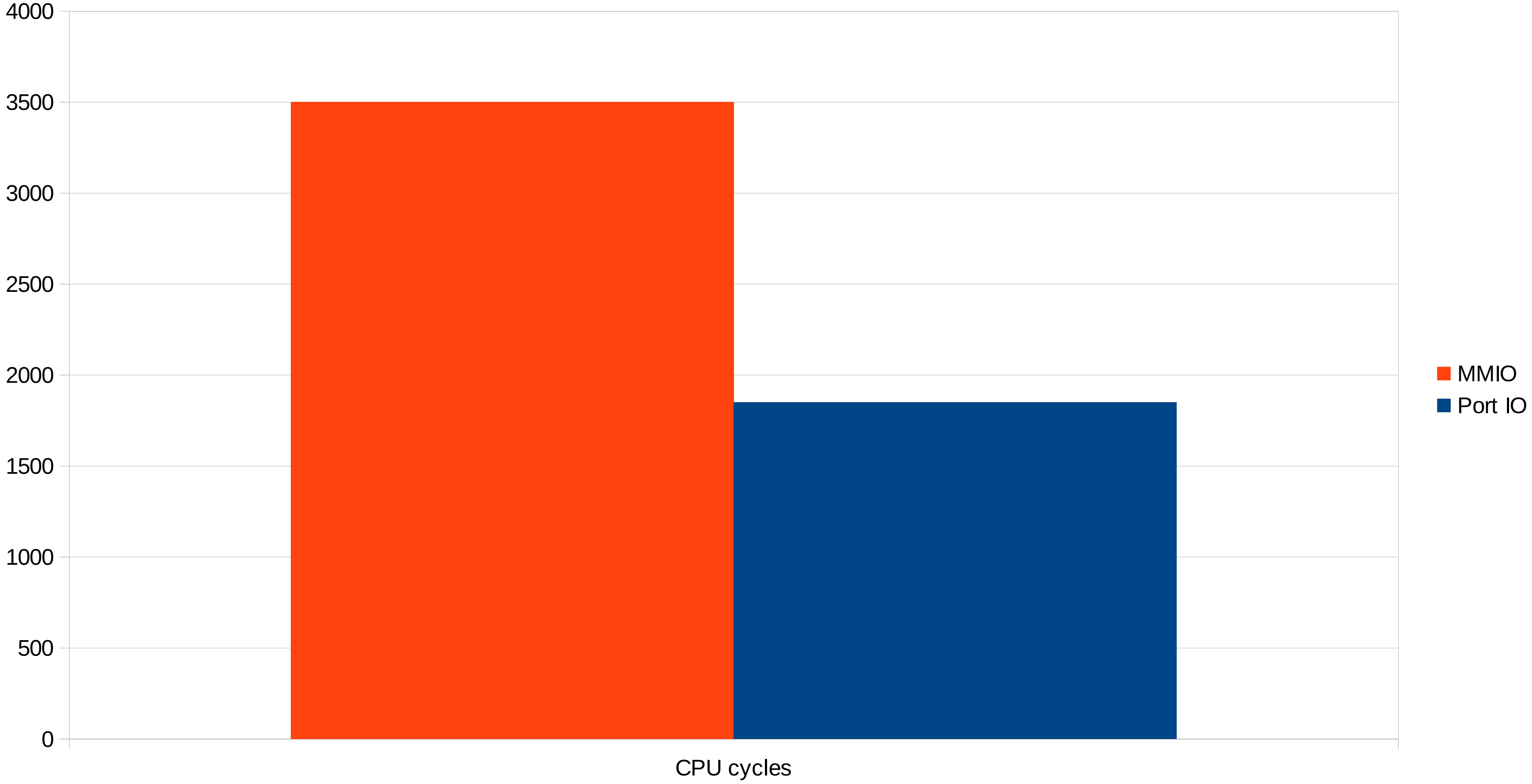
Summary

- Why do it?
  - Improved robustness for virtual devices
- Are we there yet?
  - Yes!
  - And there's more to come.

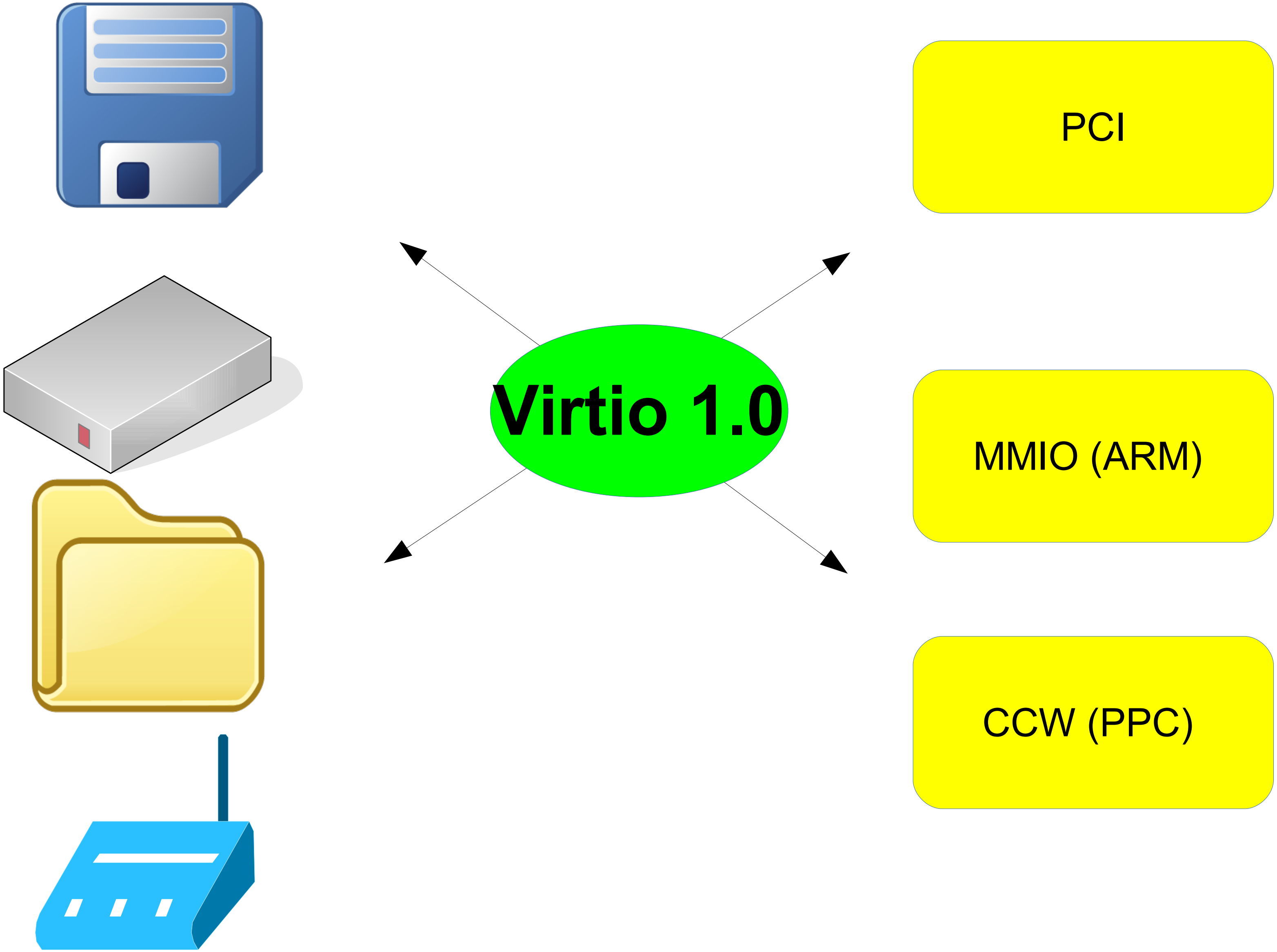


Thank you!

Virtio 0.9: Port IO versus memory on KVM x86: cycles per access (lower is better)



OASIS Virtio TC



Virtio 1.0

PCI

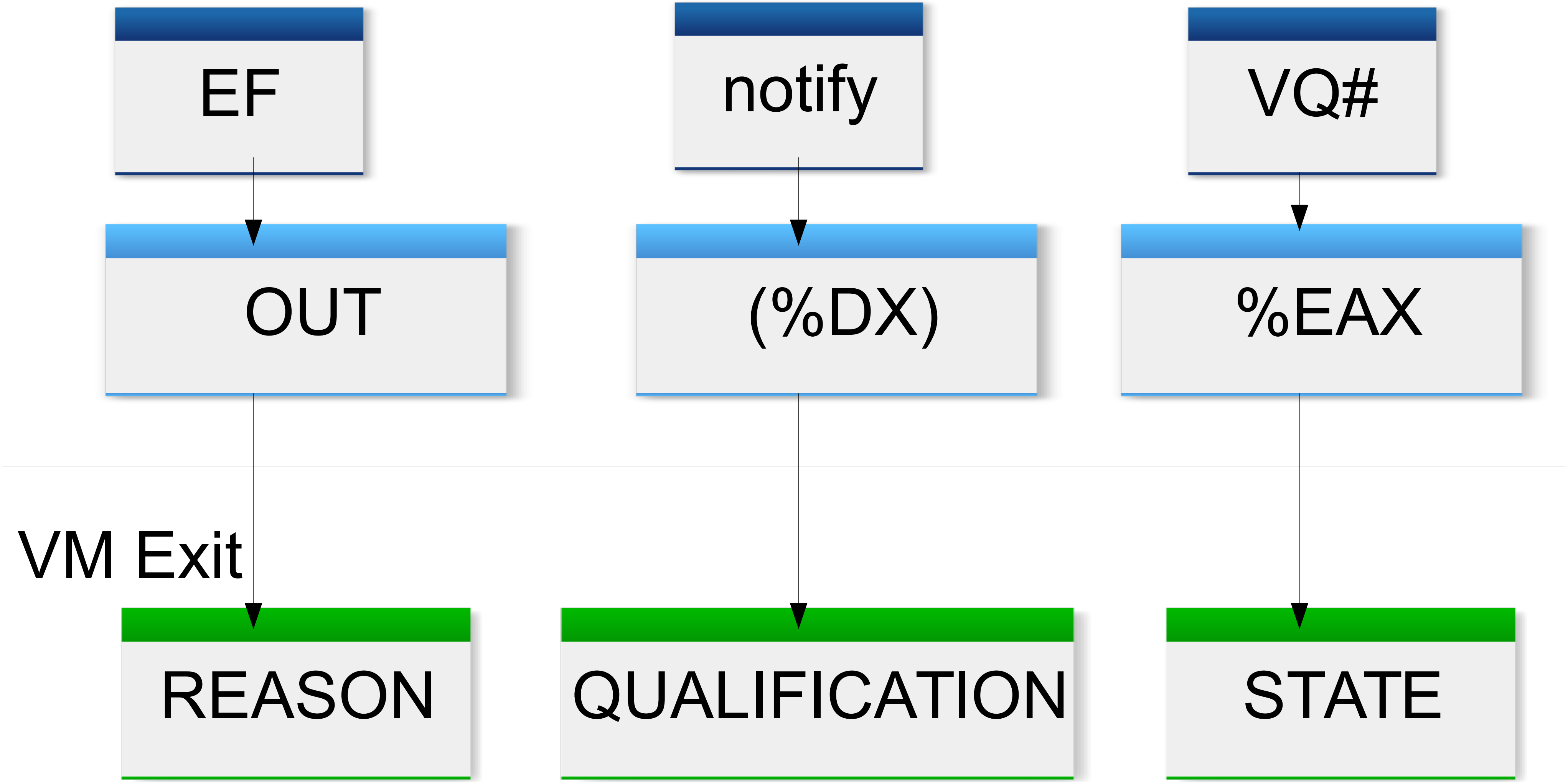
MMIO (ARM)

CCW (PPC)

Virtio 1.0

- Virtio PCI:
  - Replace Port IO with Memory mapped IO
  - PCI Express (hotplug, AER, multi-root, SRIOV)
  - Infinite features
- Reduced memory requirements
- Fixed endianness
- Compatibility

Port IO: outl


EF

notify

VQ#

OUT

(%DX)

%EAX

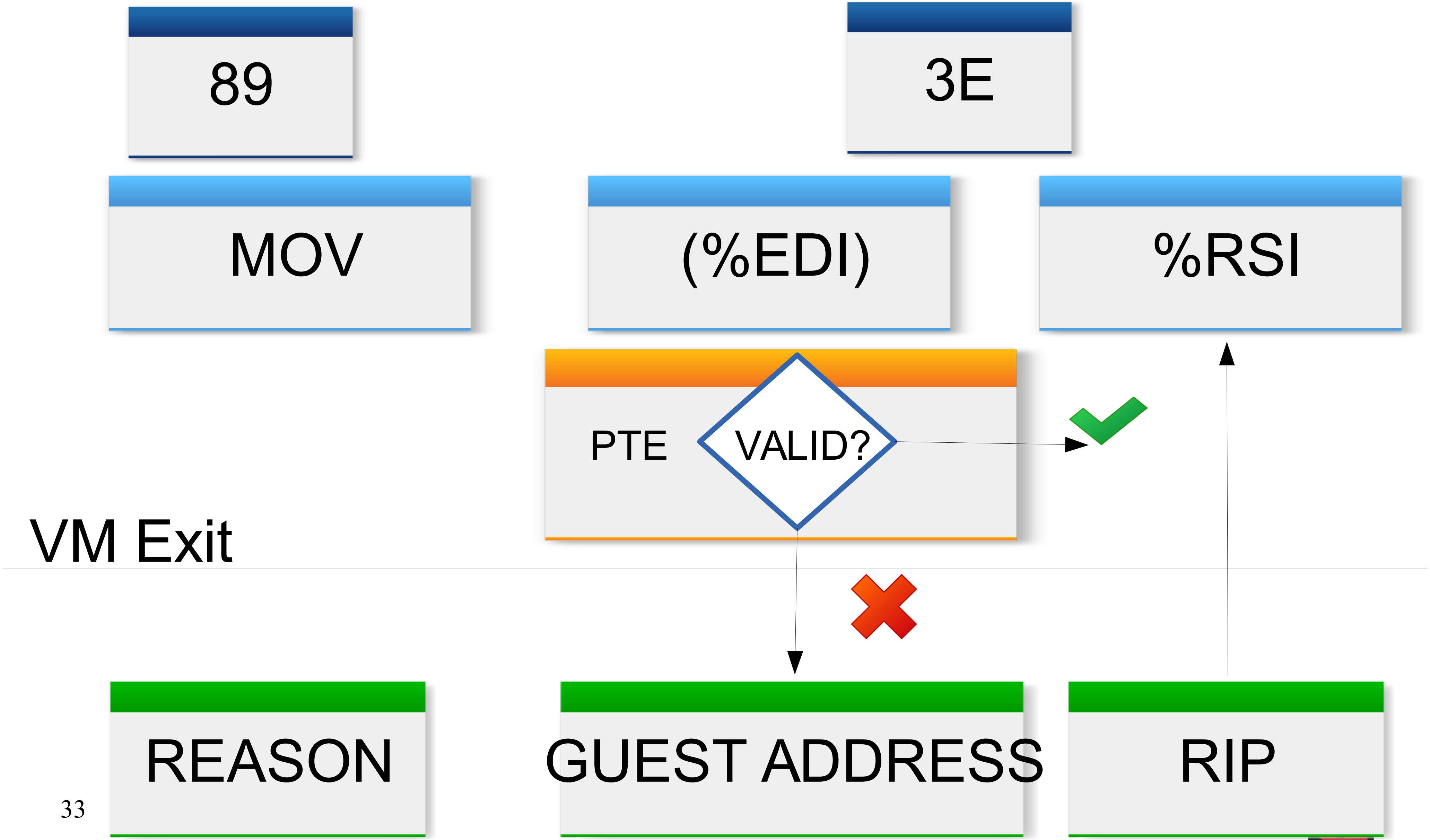
VM Exit

REASON

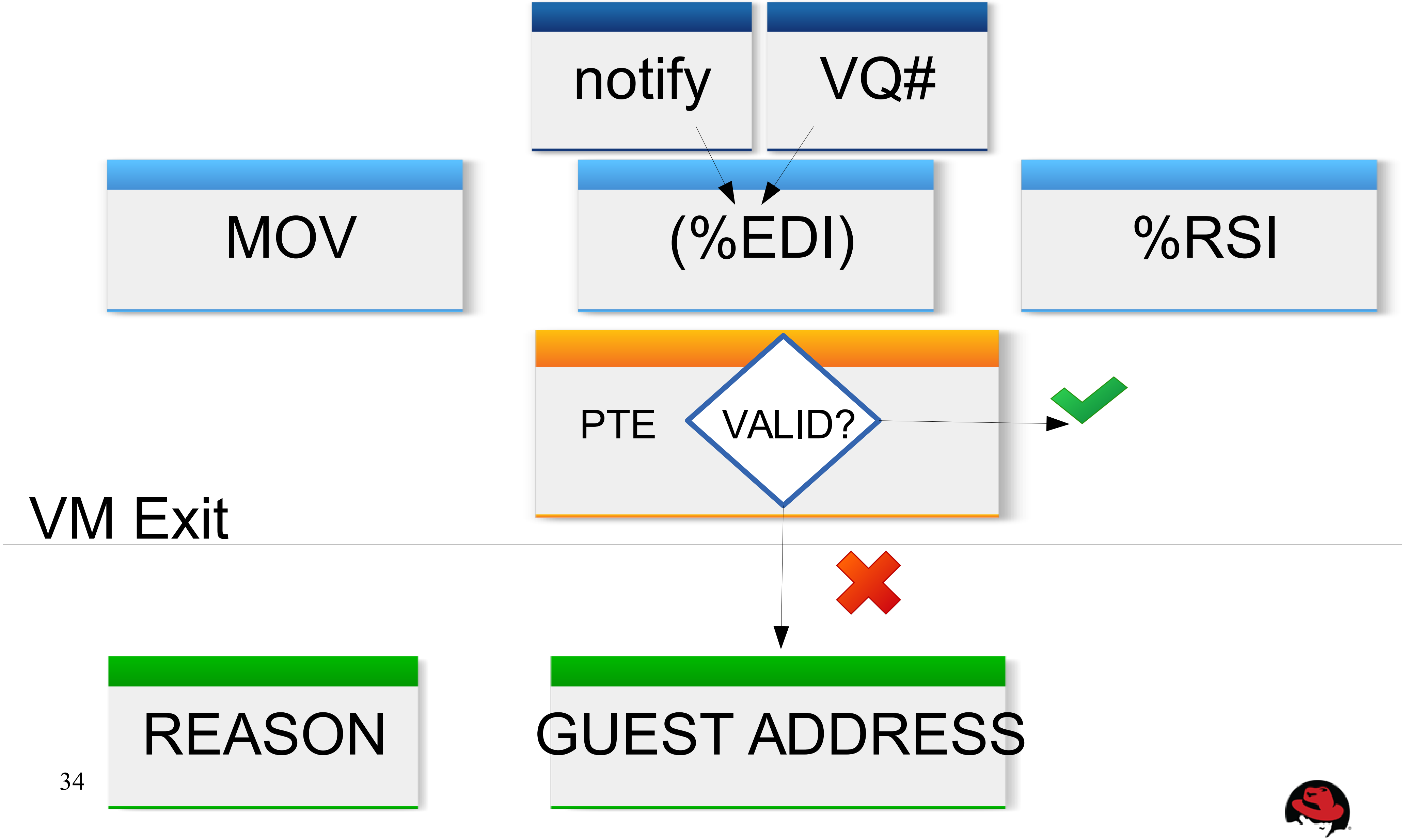
QUALIFICATION

STATE

Memory mapped IO: writel



Fast MMIO


notify

VQ#

MOV

(%EDI)

%RSI

PTE

VALID?

✓

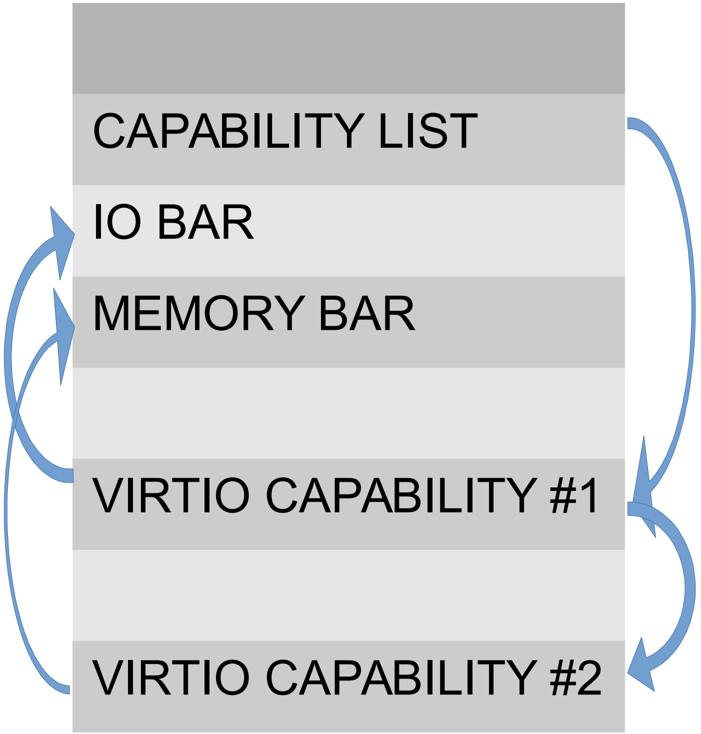
VM Exit

✗

REASON

GUEST ADDRESS

Multiple interfaces



CAPABILITY LIST

IO BAR

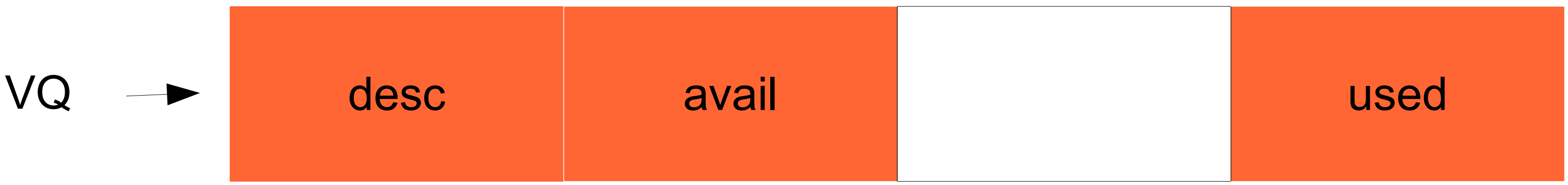
MEMORY BAR

VIRTIO CAPABILITY #1

VIRTIO CAPABILITY #2

Memory requirements

0.9



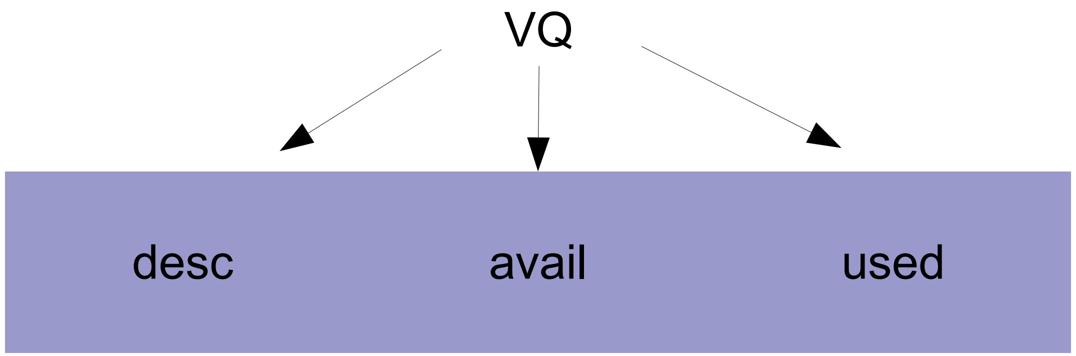
VQ

desc

avail

used

1.0



VQ

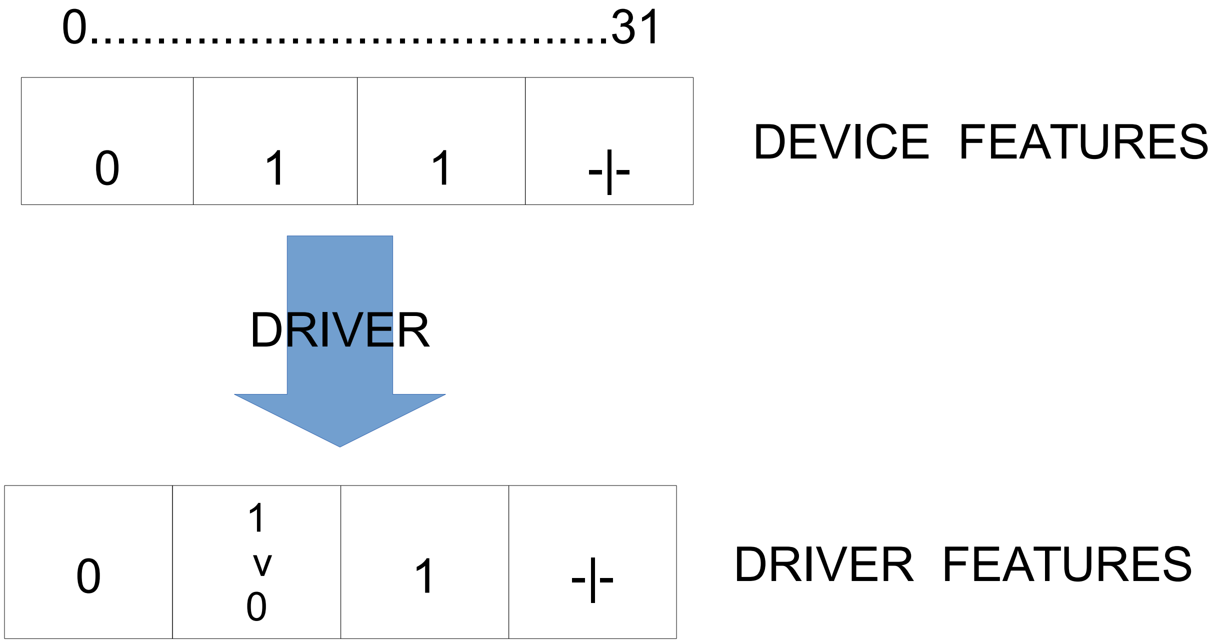
desc

avail

used

features

0.9



0.....................................31

0 | 1 | 1 | -|-

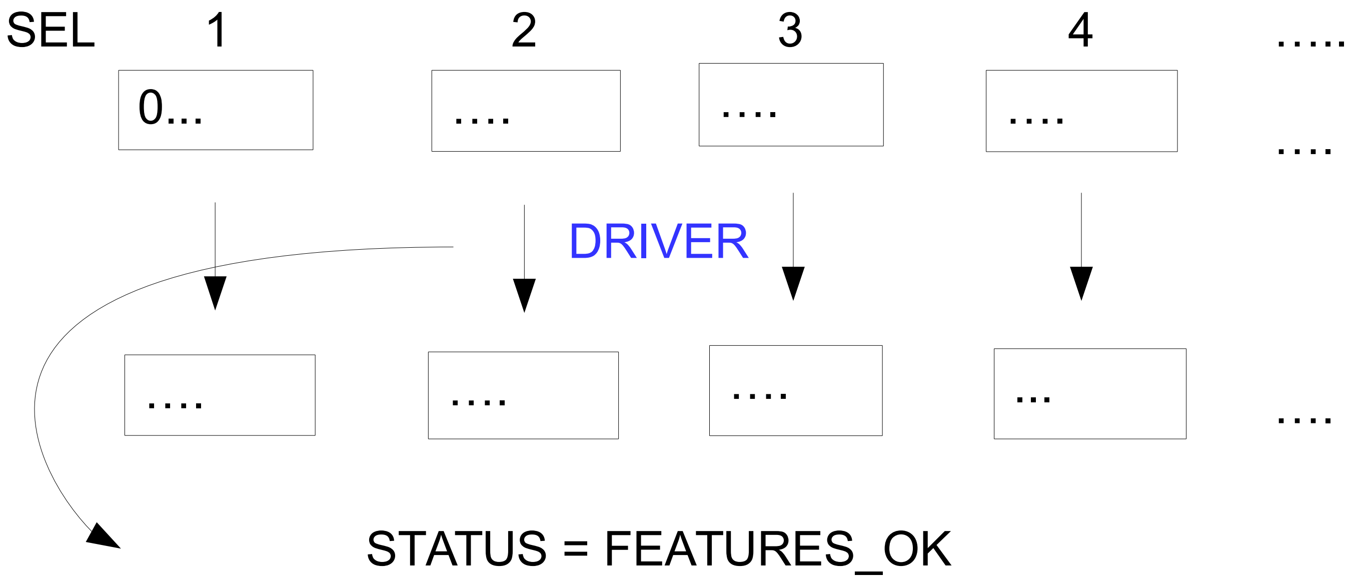
DEVICE FEATURES

DRIVER

0 | 1 v 0 | 1 | -|-

DRIVER FEATURES

1.0



SEL 1 2 3 4 .....

0... | .... | .... | .... | ....

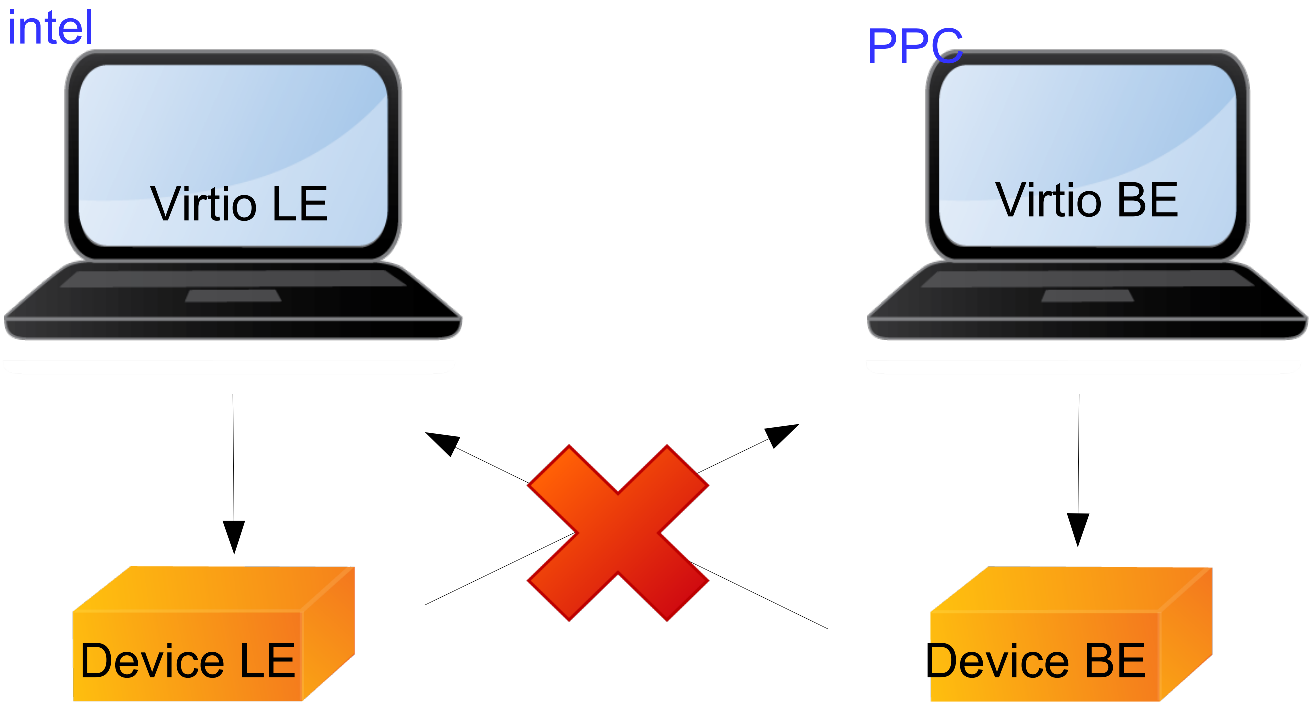
DRIVER

.... | .... | .... | ... | ....

STATUS = FEATURES_OK

Endianness

Virtio 0.9



intel

Virtio LE

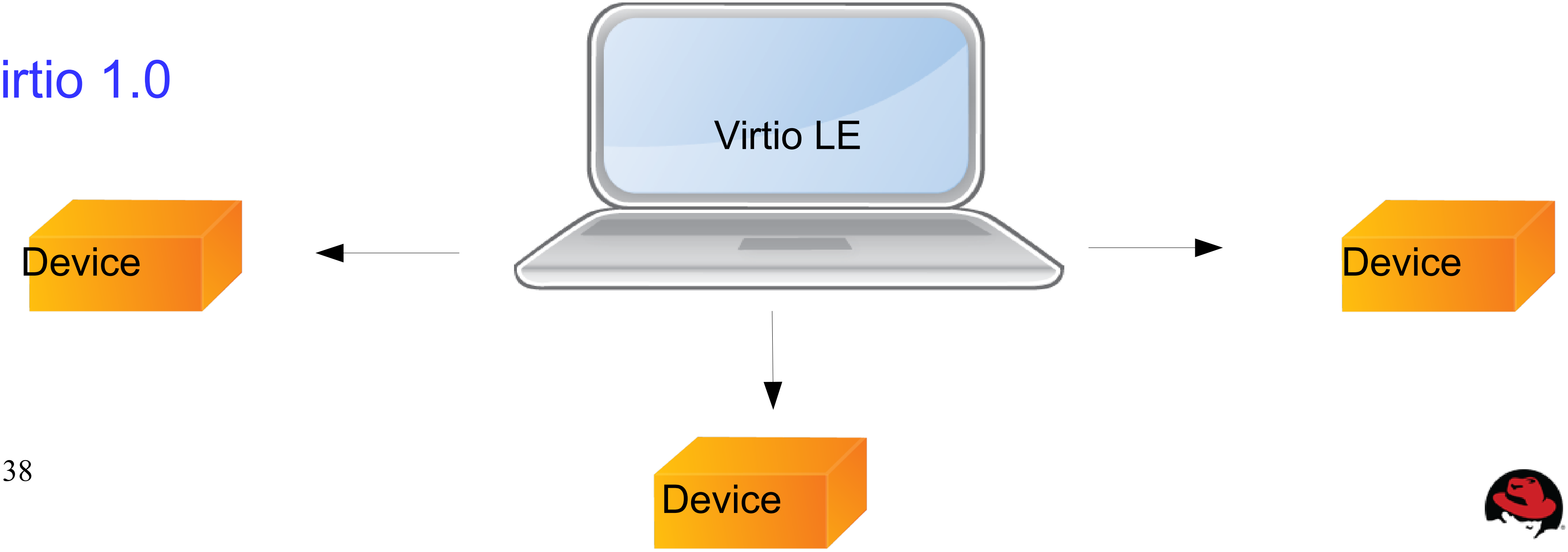
Device LE

PPC

Virtio BE

Device BE

Virtio 1.0



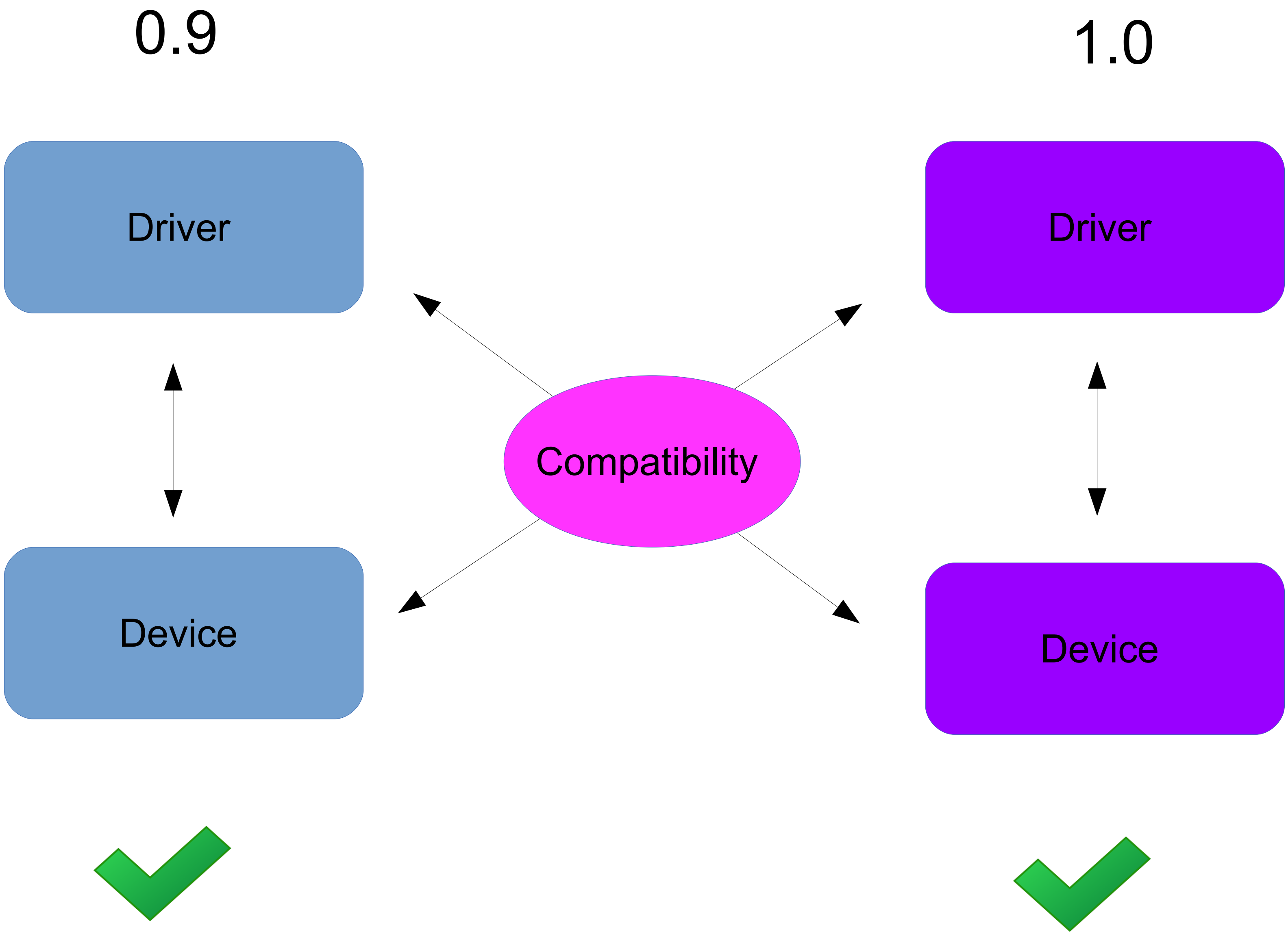
Virtio LE

Device

Device

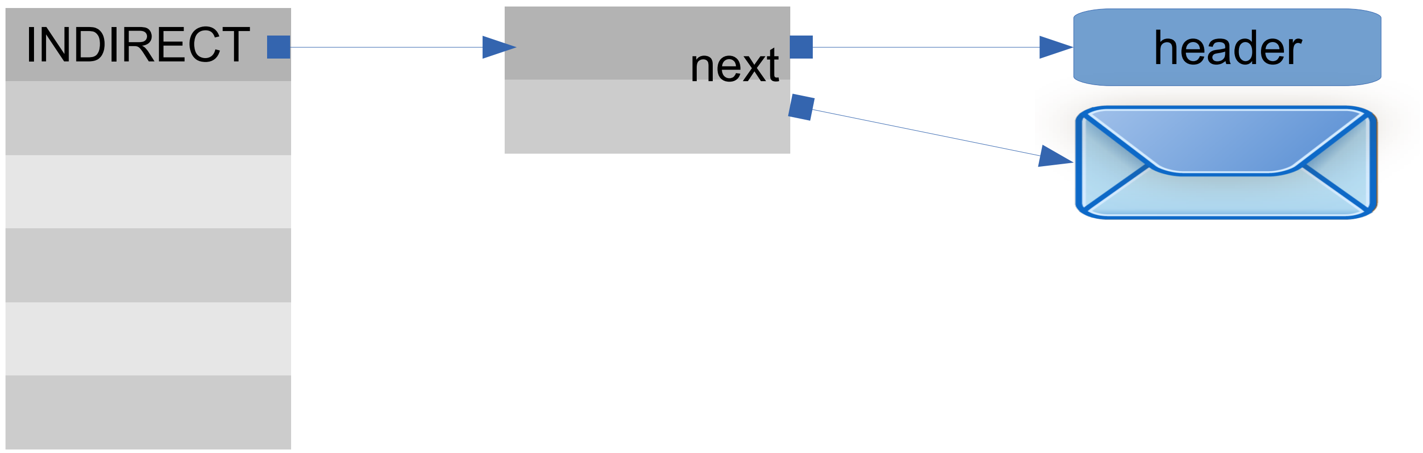
Device

compatibility

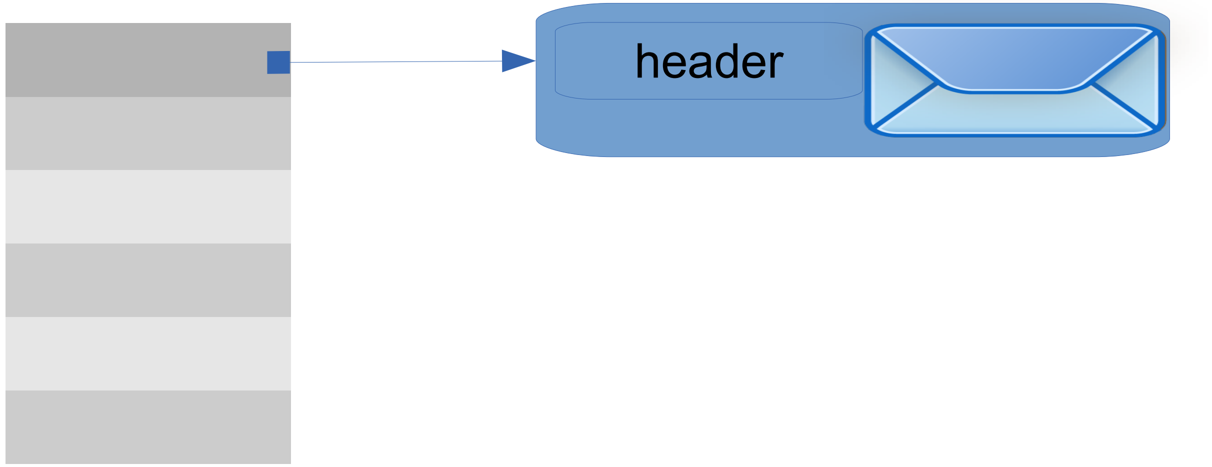


Packet layout

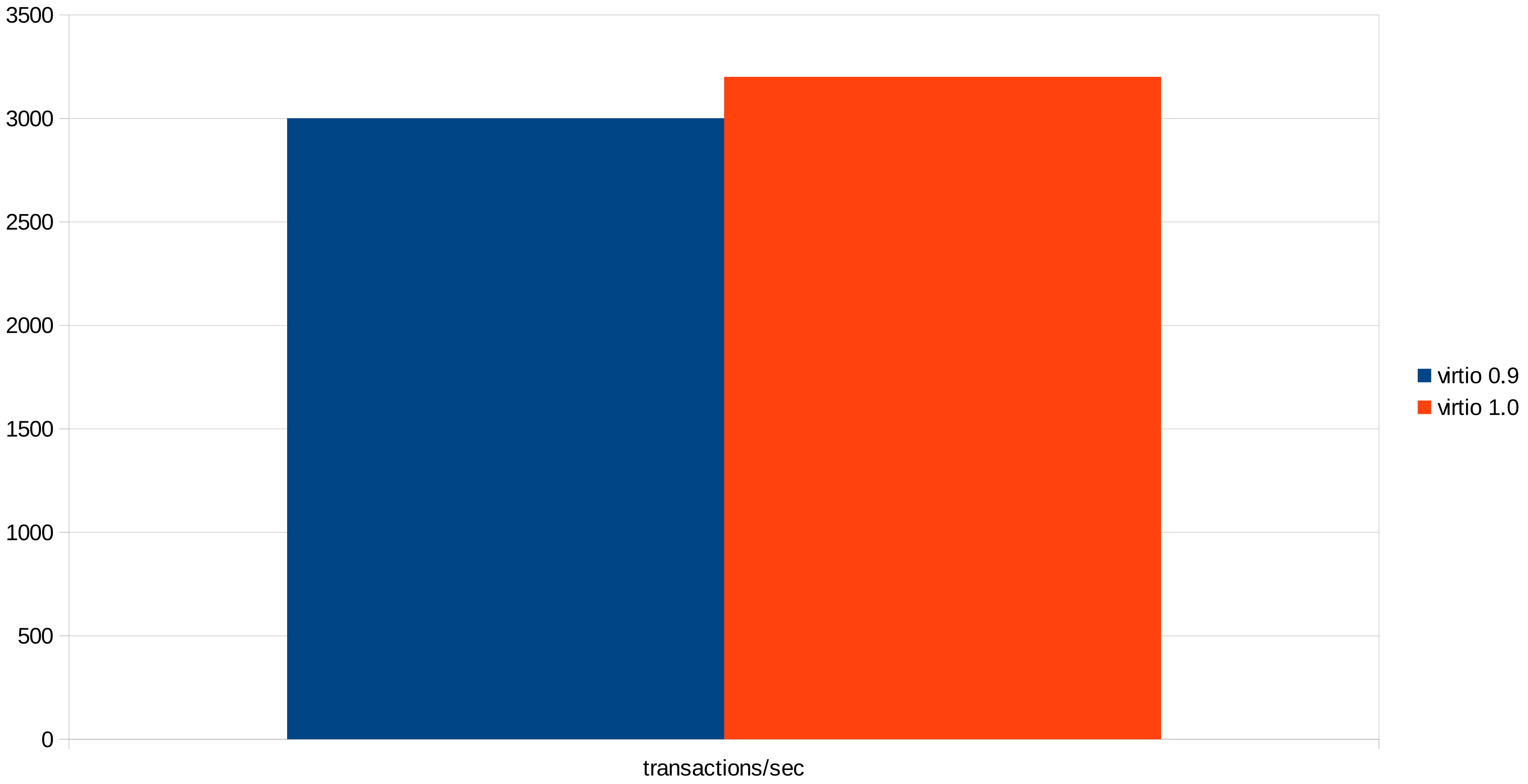
Virtio 0.9



Virtio 1.0



Packet layout: transactions per sec (higher is better)



More: virtio 1.0 versus 0.9.5

- Virtio 9p
- Virtio blk: WCE
- Virtio-net Multiqueue
- Virtio-net dynamic offloads
- Already upstream (based on spec draft)

vhost updates

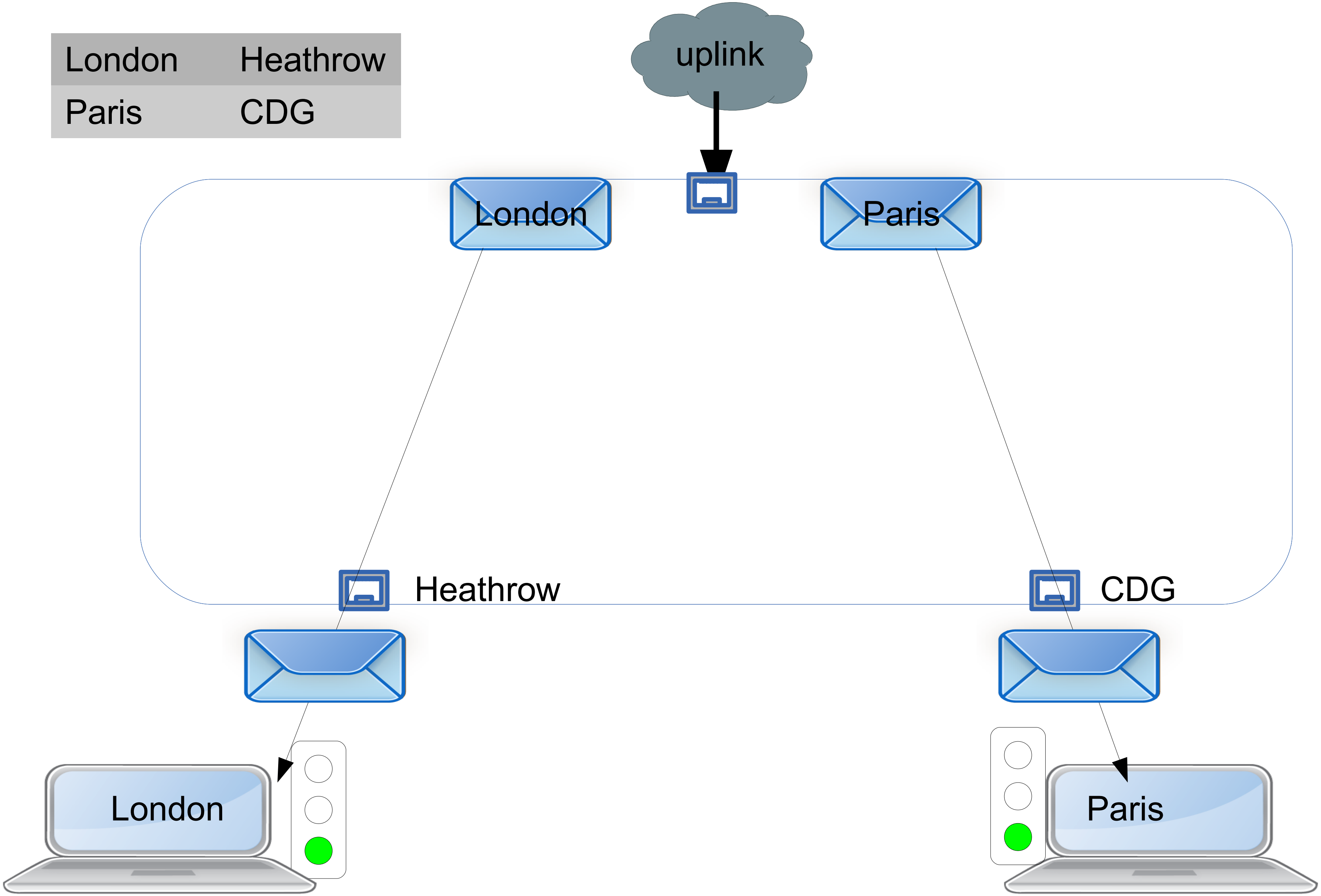
- Vhost scsi
- Vhost-net zero copy transmit
- No need for driver changes

Kvm networking

- Openvswitch – if time allows
- Ethernet bridge

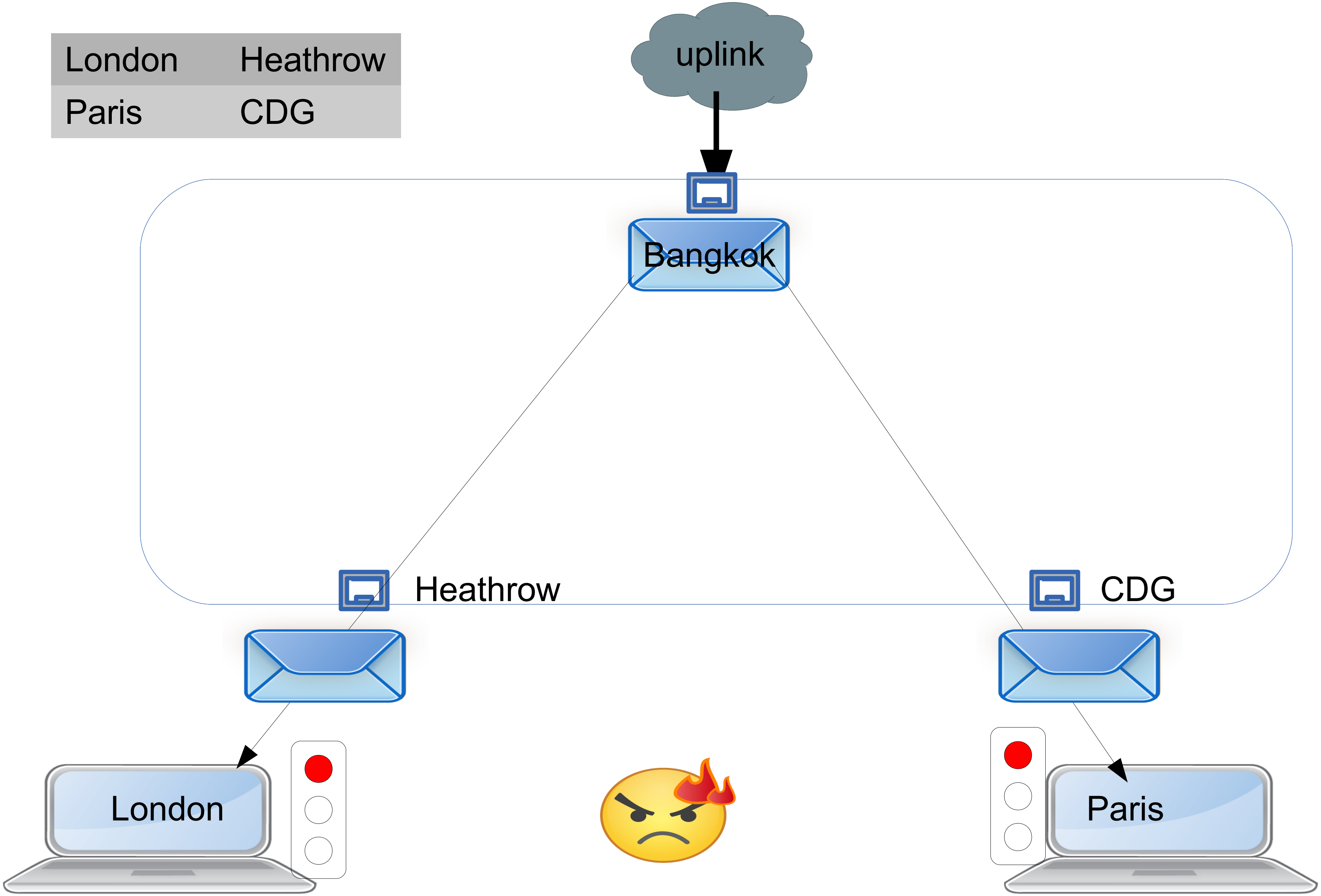
Bridge FDB

London	Heathrow
Paris	CDG



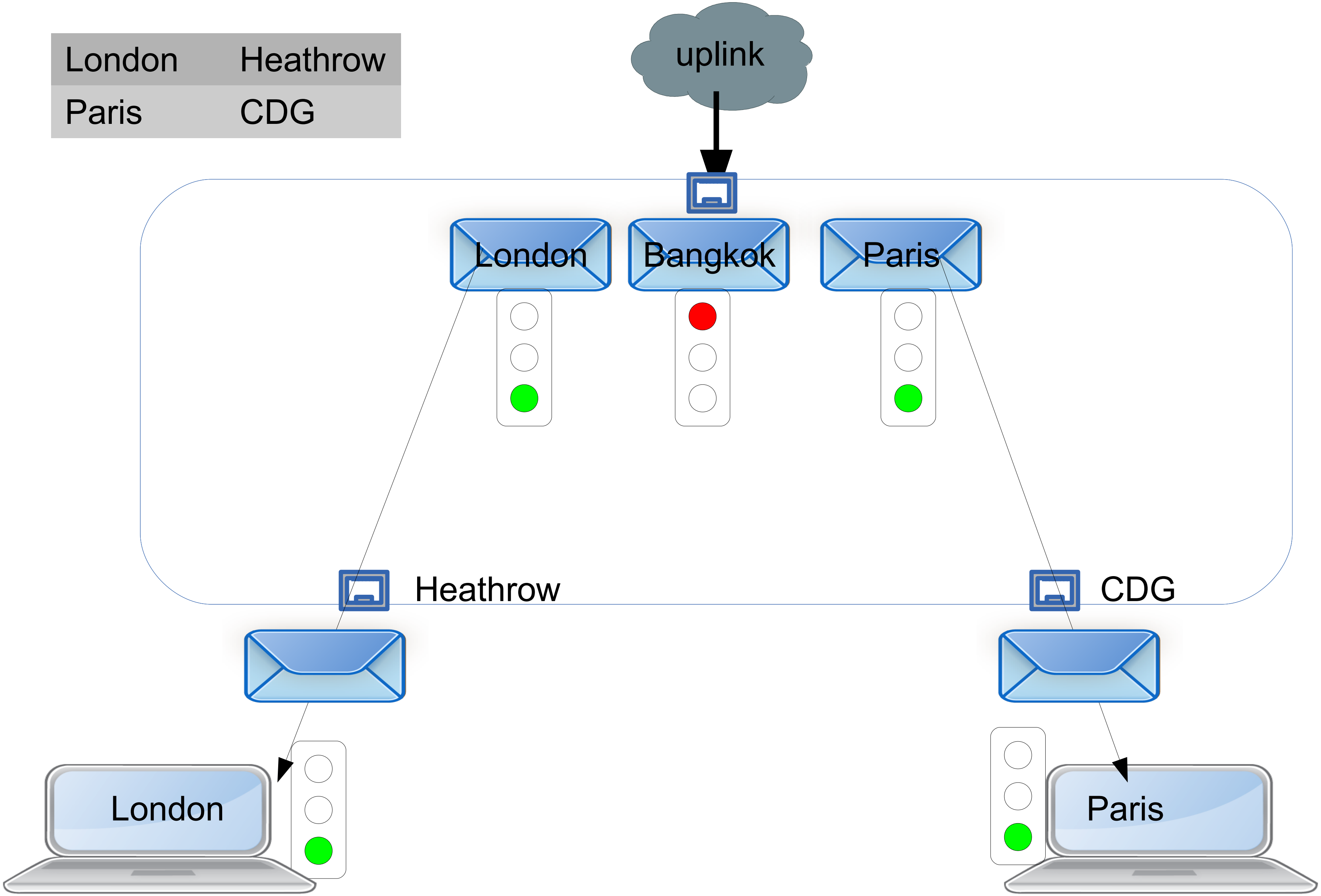
Flood: DOS potential

London	Heathrow
Paris	CDG



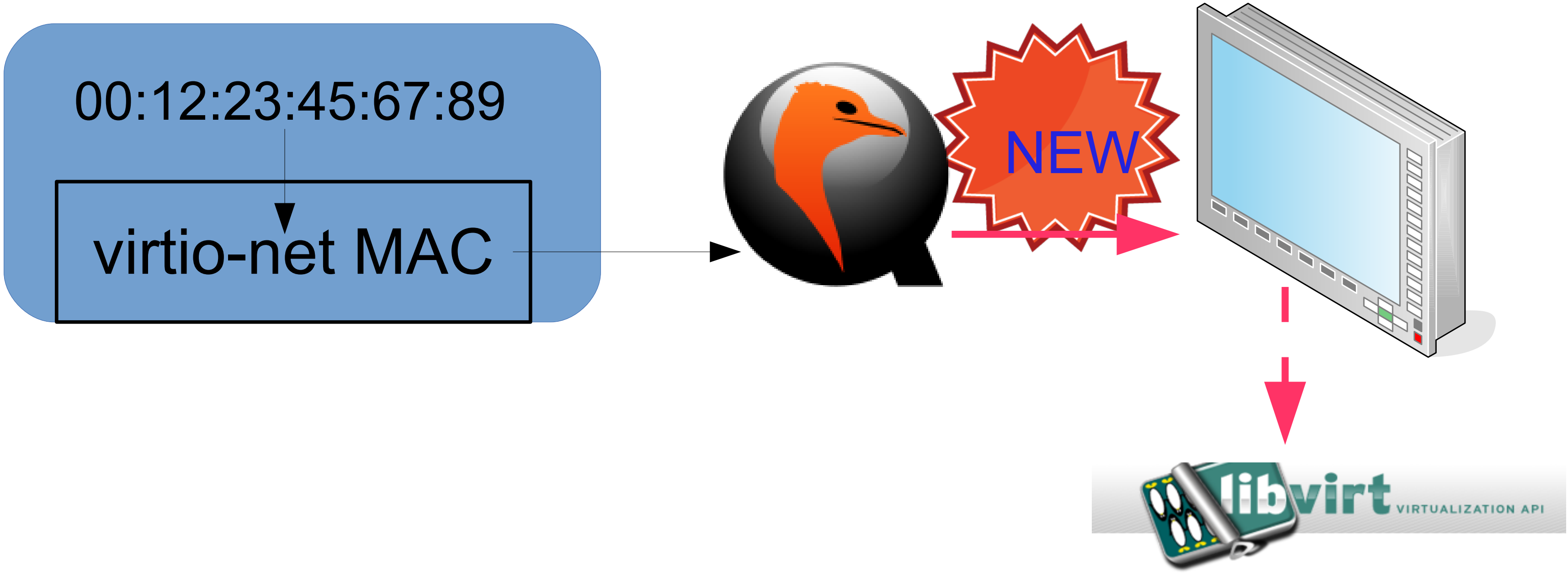
Disable flood

London	Heathrow
Paris	CDG

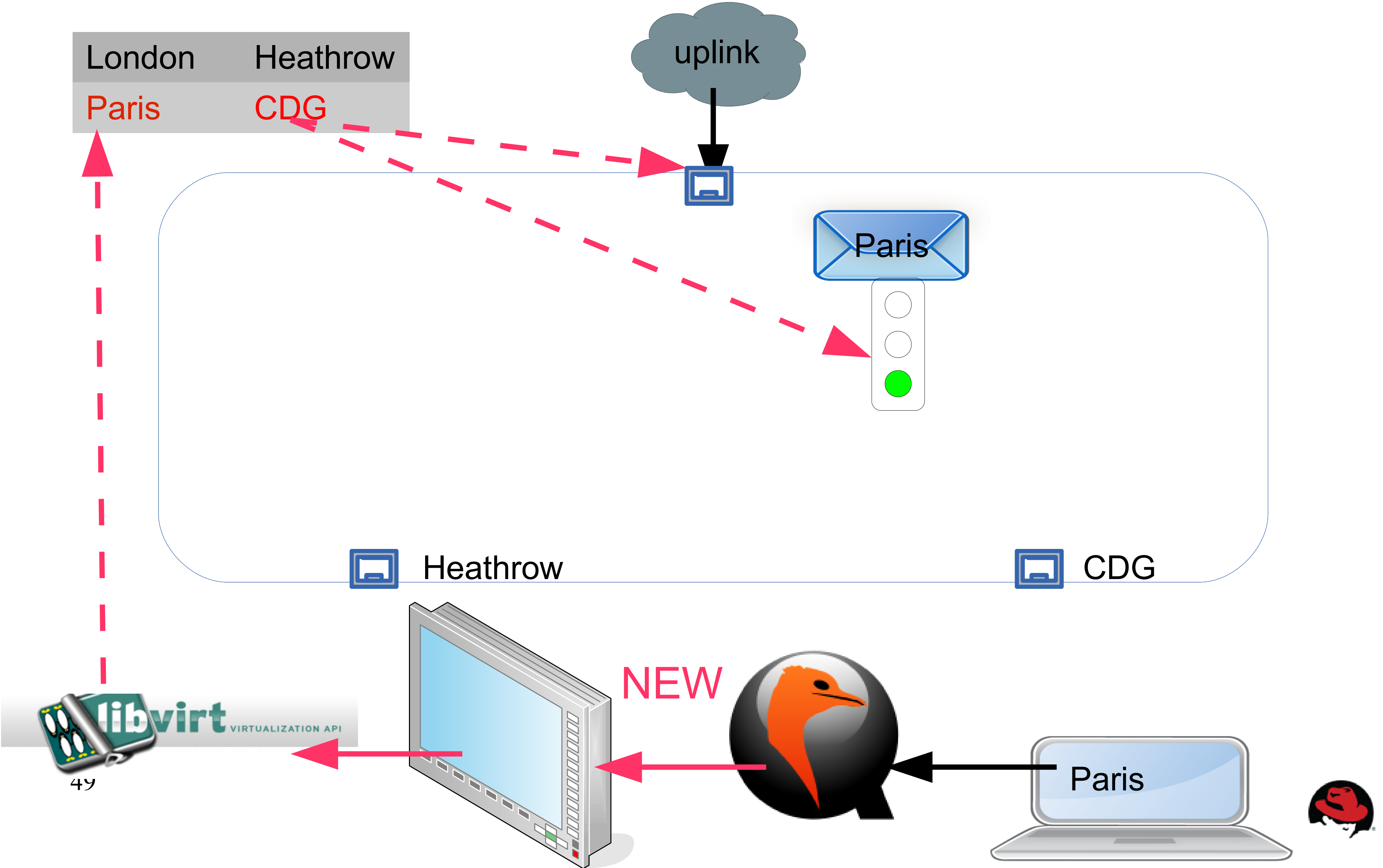


softmac

- Ifconfig eth0 hw ether 00:12:23:45:67:89



Using softmac/non promiscuous

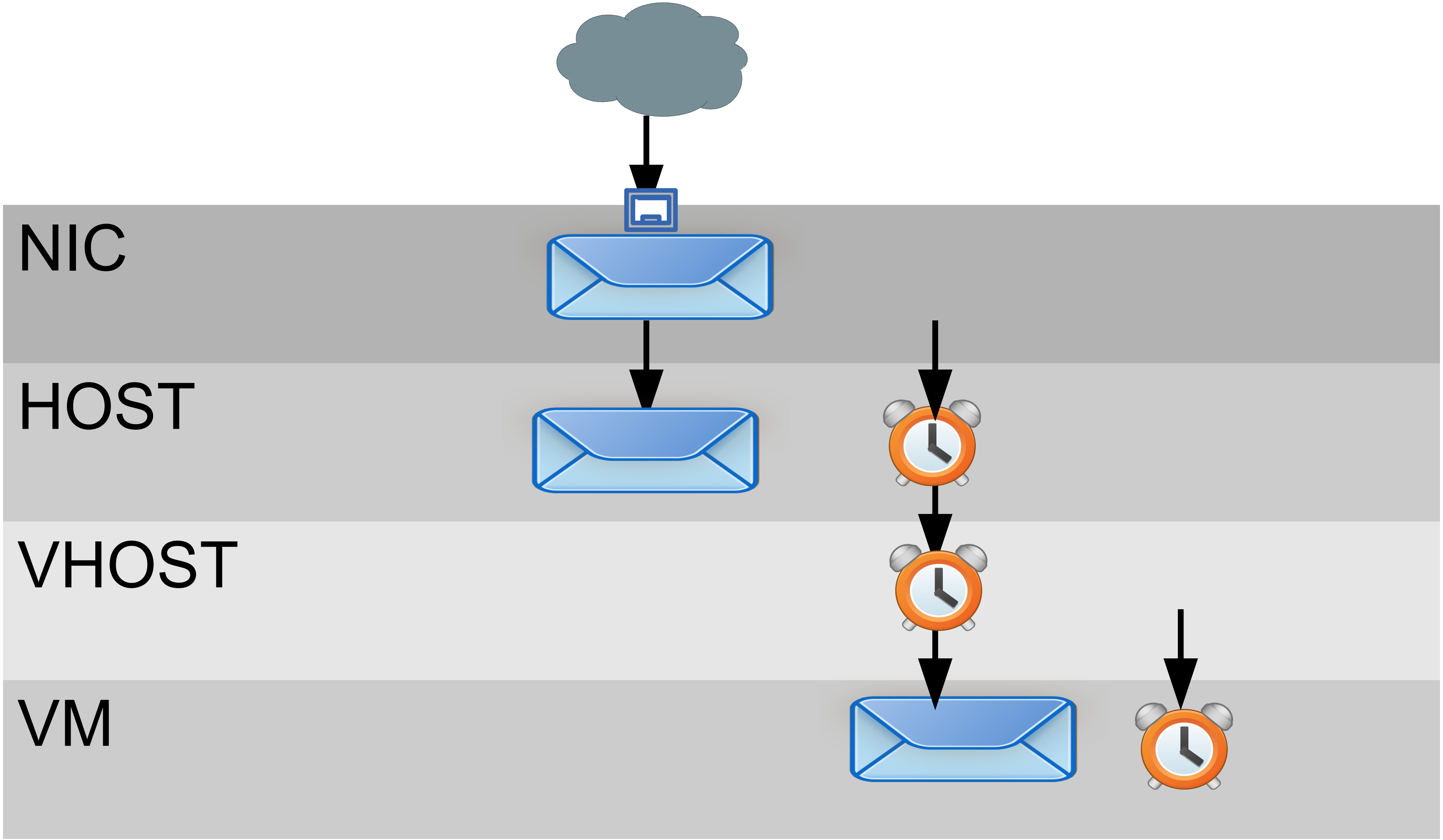


London	Heathrow
Paris	CDG

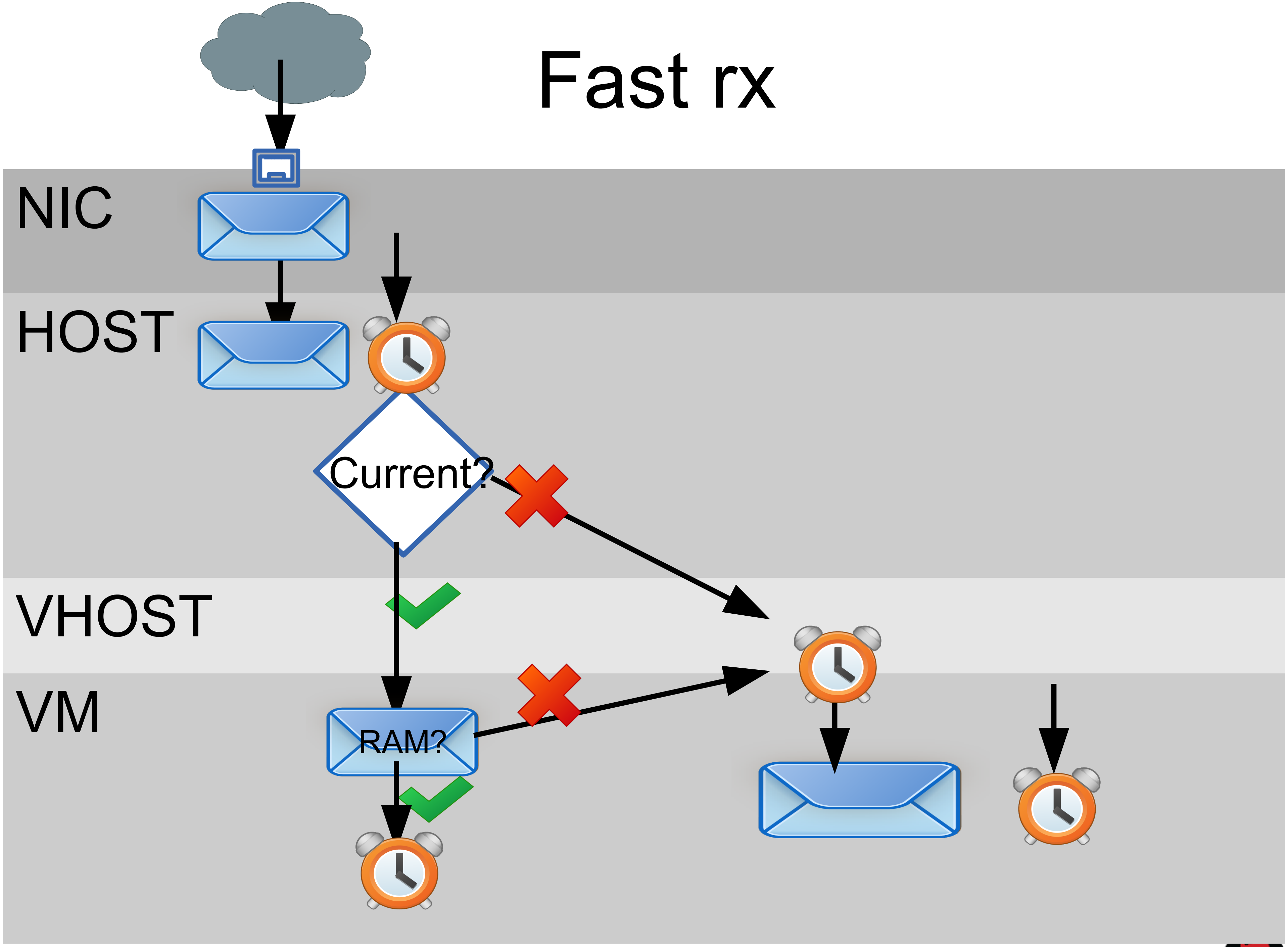
Work in progress

- ELVIS (vhost blk/vhost net)
- Virgl
- Vhost-net performance

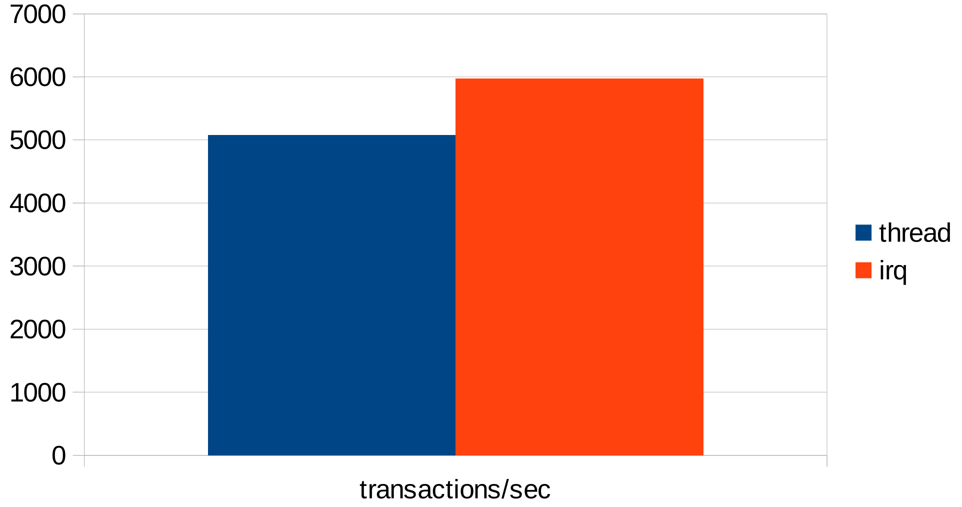
RX latency



Fast rx

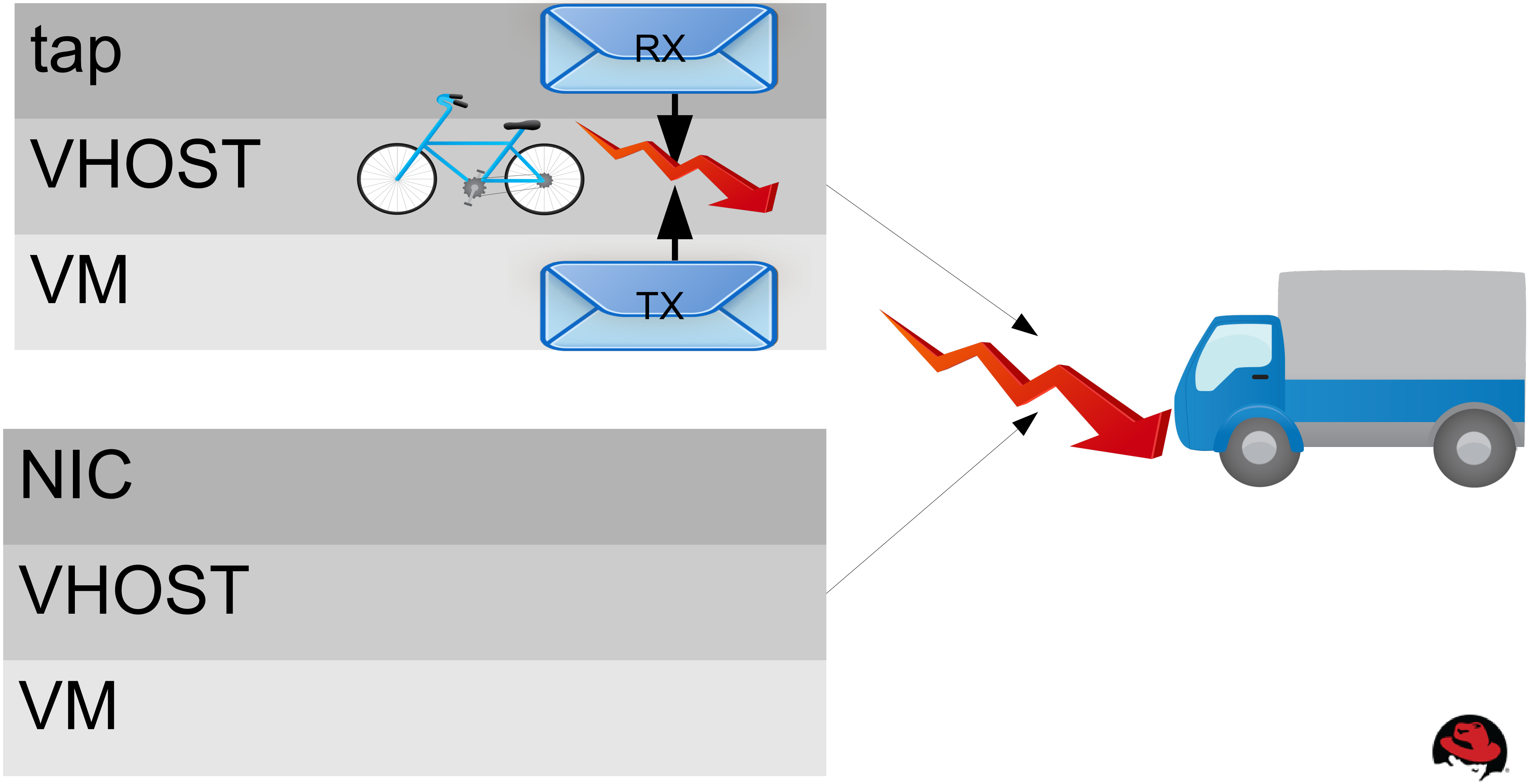


Fast rx: transactions per sec (higher is better)



Hit	331668
Miss	79

Vhost-net threading



tap

RX

VHOST

VM

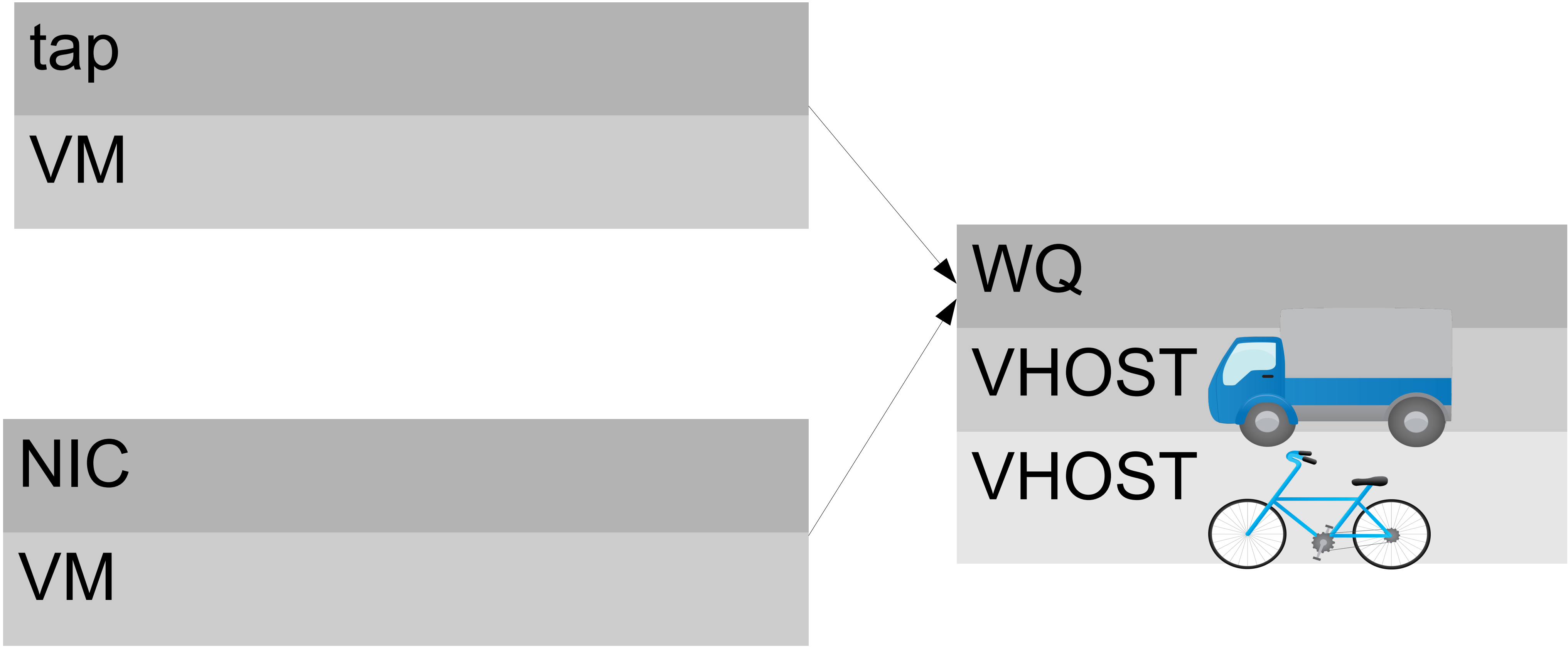
TX

NIC

VHOST

VM

Vhost-net thread pool



tap

VM

NIC

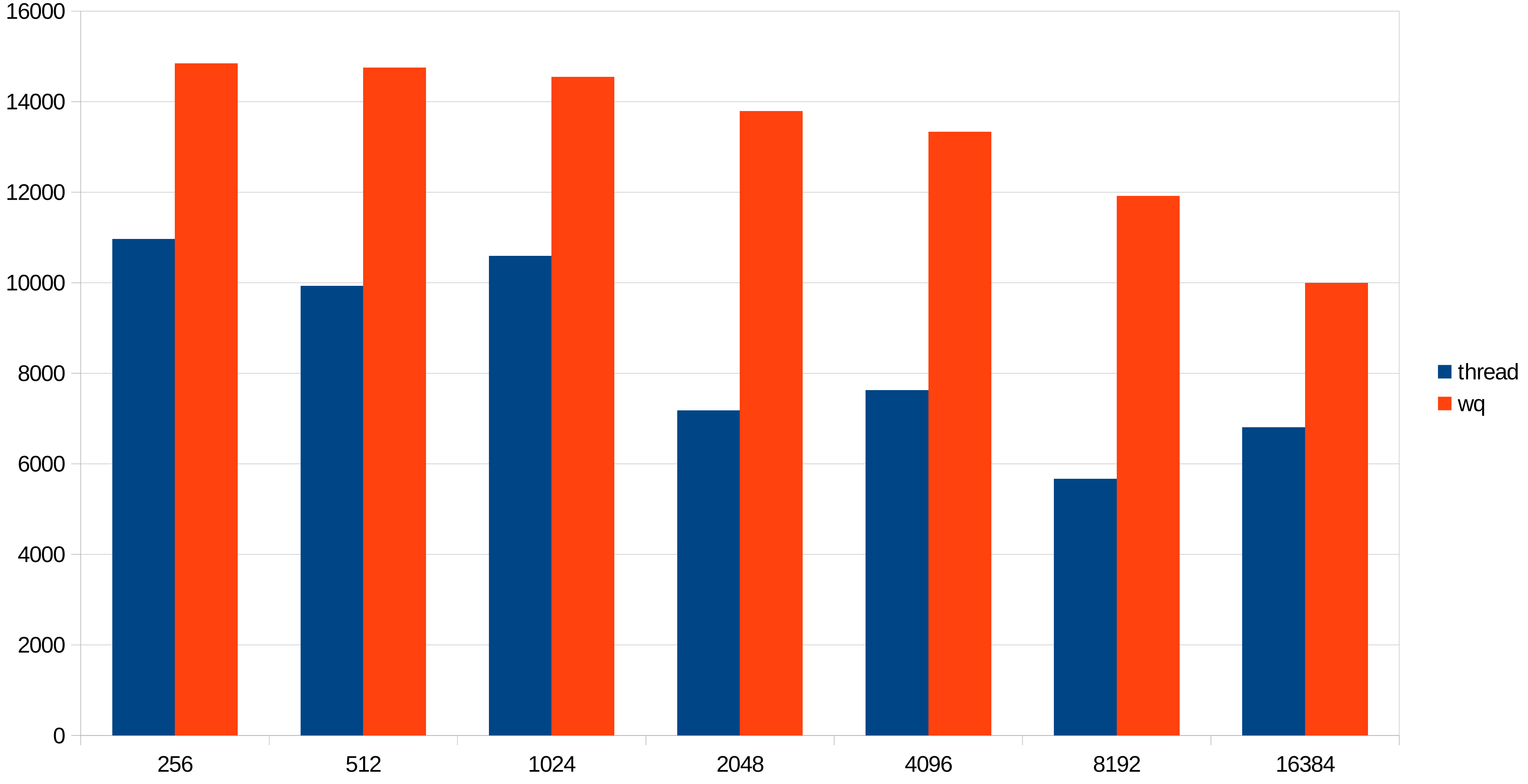
VM

WQ

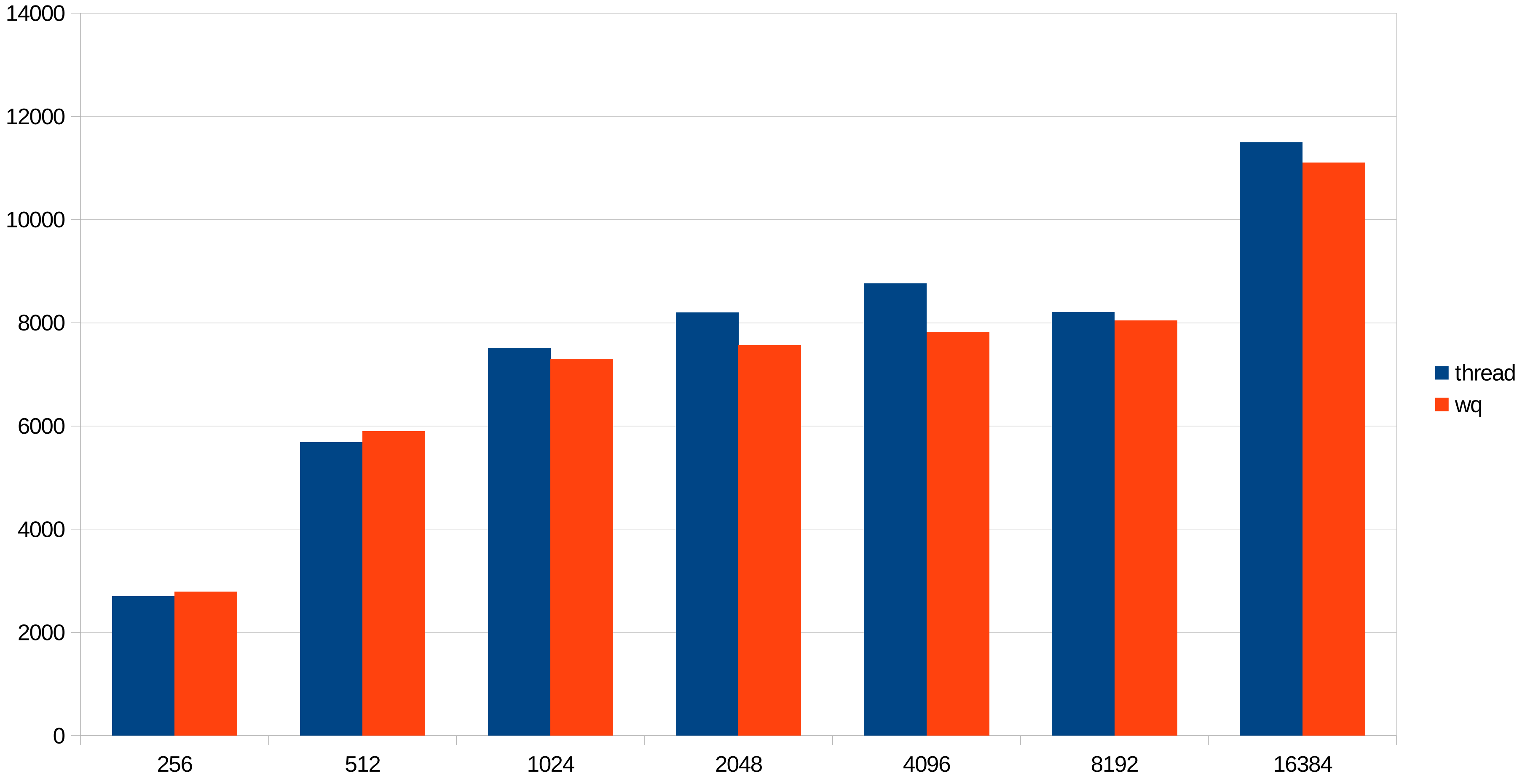
VHOST

VHOST

threading: UDP RR transactions/sec (higher is better)



threading: TCP STREAM transactions/sec (higher is better)

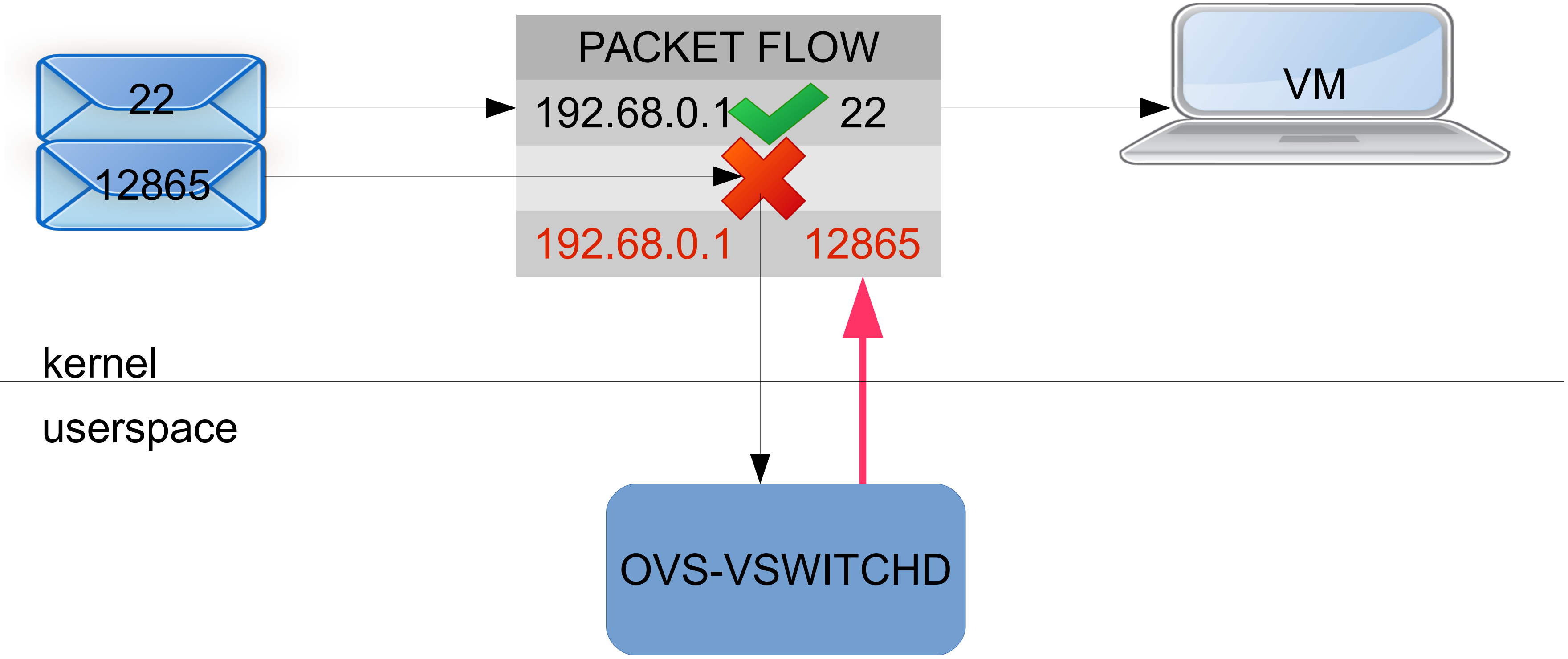


summary

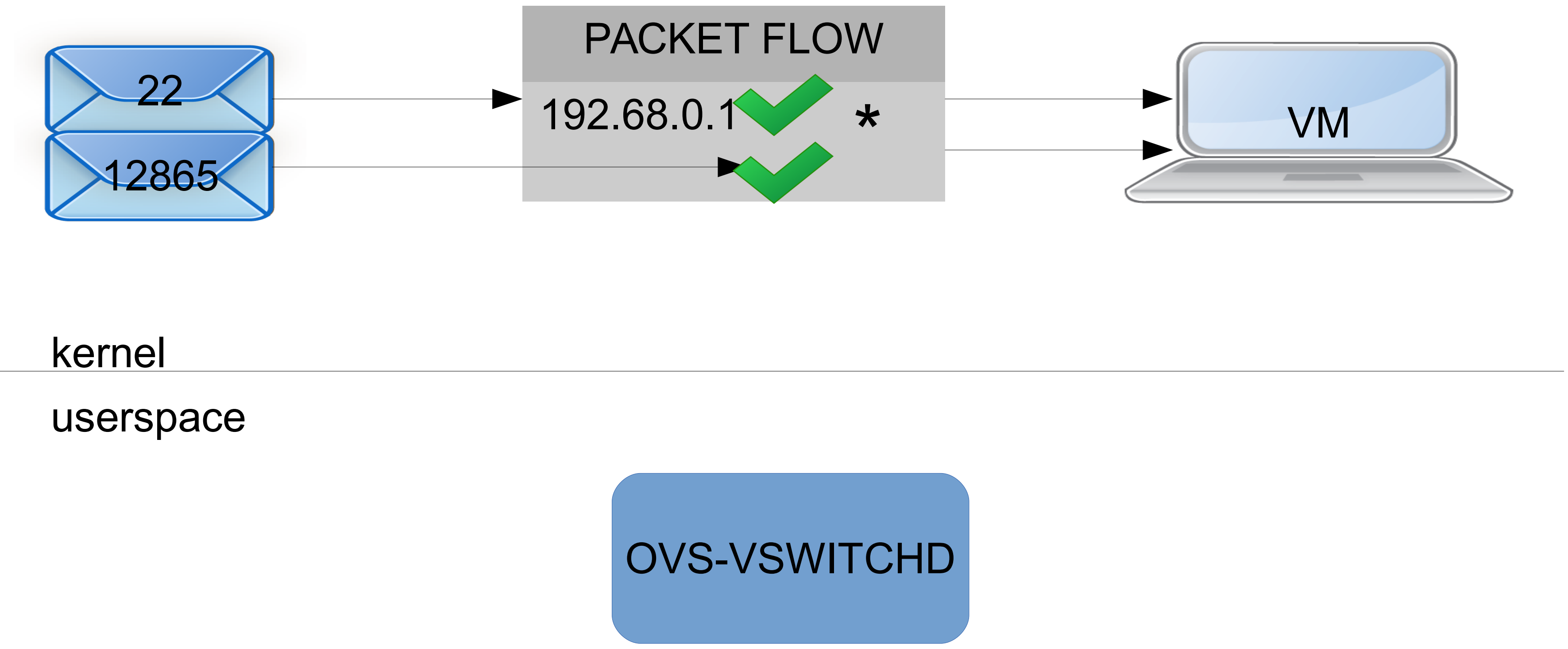
- Performance
- Manageability
- Security

Questions?

OVS: flow match



OVS: wildcard match



22

12865

PACKET FLOW

192.68.0.1 ✓ *

✓

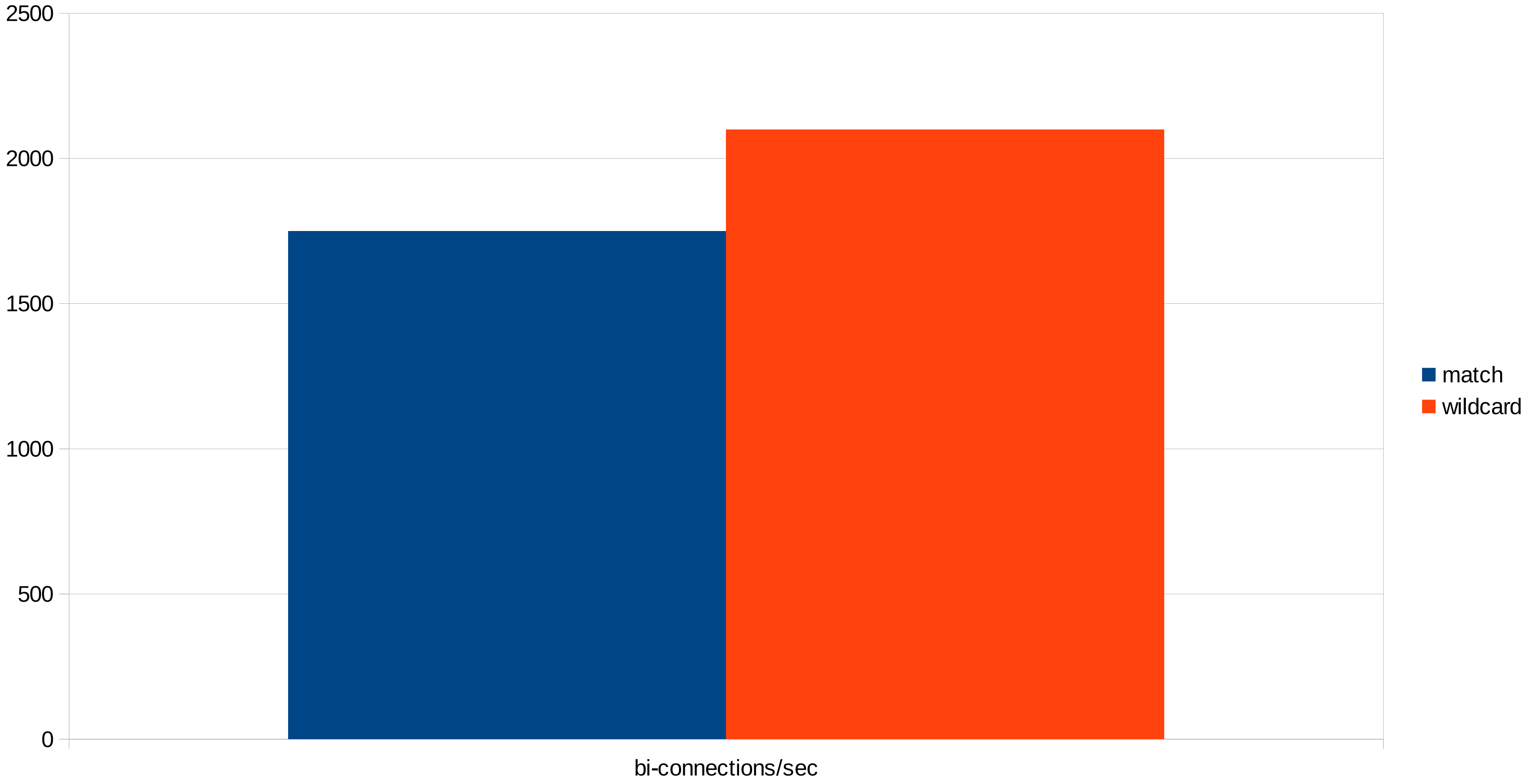
VM

kernel

userspace

OVS-VSWITCHD

Wilcard: netperf CRR (higher is better)



libvirt
VIRTUALIZATION API