



Network virtualisation using Crossbow Technology

Uroš Nedić, MSc
OpenSolaris Contributor

- Crossbow Technical Background
- Network Machines
- Network in a Box

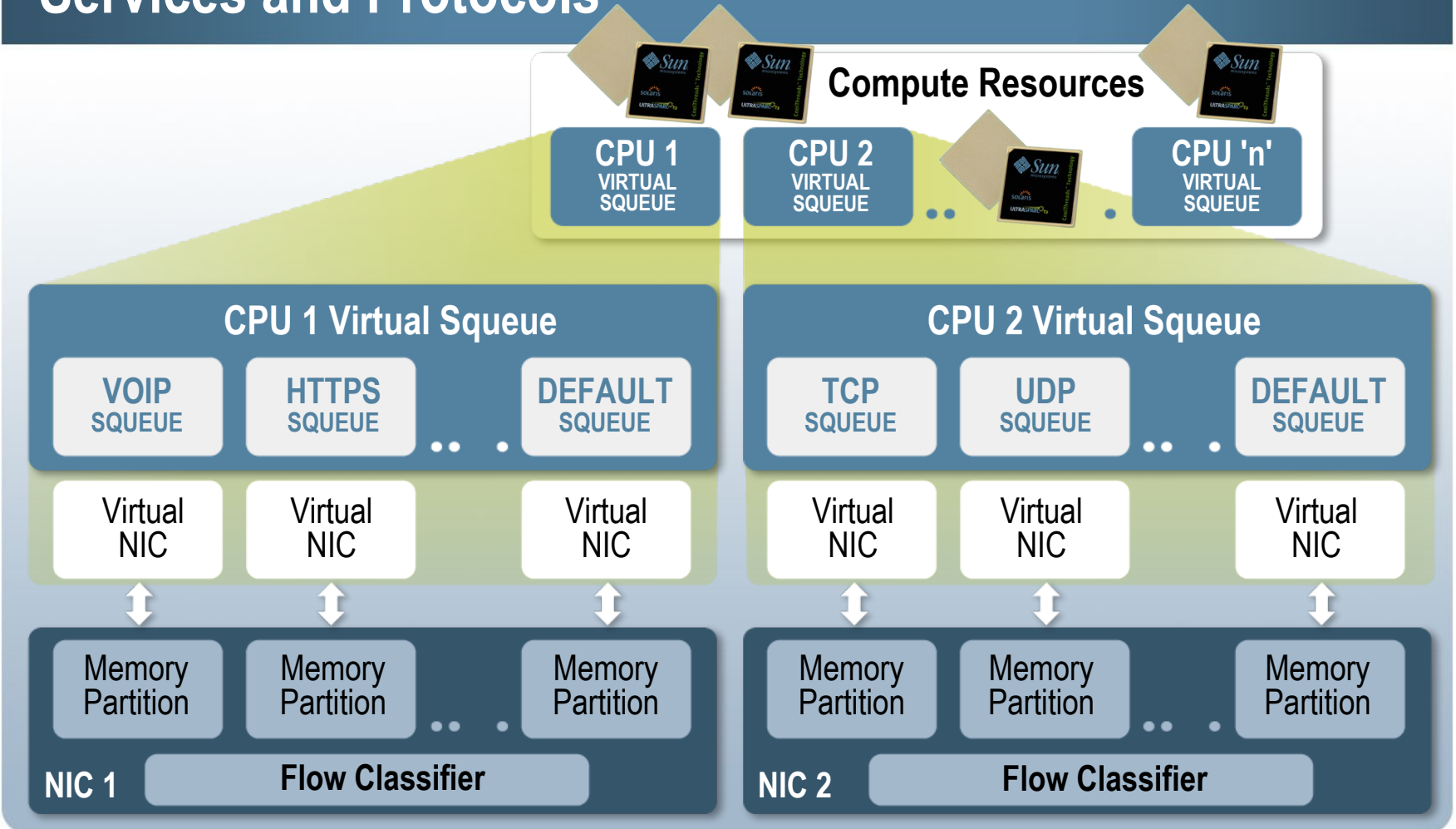
Crossbow Features

- Virtualization
 - > IP Instances
 - > NIC Virtualization - VNICs
 - > Service Virtualization - Flows
- Resource partitioning
 - > QoS/Diffserv (without performance penalties)
 - > SLA on a per connection basis
 - > Class of Service Support
- Performance & Observability
 - > Dynamic polling
 - > H/W and S/W fanout to multiple cores
 - > Real Time Usage and History

Better Defense against DDOS attacks

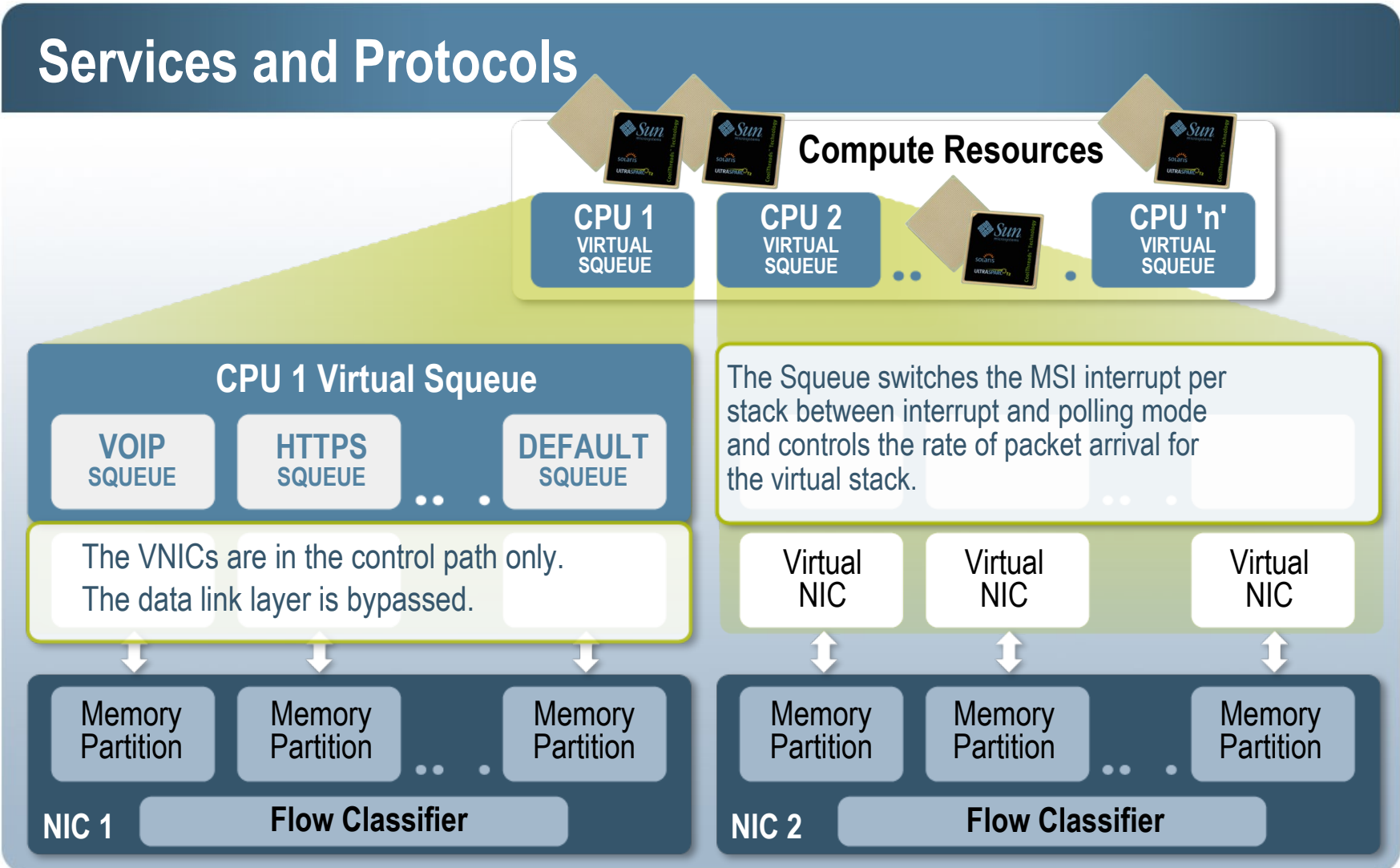
Virtual Stacks

Services and Protocols



Virtual Stacks

Services and Protocols



Crossbow Architecture

- Partition the NIC Hardware, kernel queues, and CPU and allow creation of Virtual NICs
- Use dynamic polling on Virtual NICs to schedule rate of packet arrival per VNIC
- Effect of dynamic polling

Mpstat (older driver)

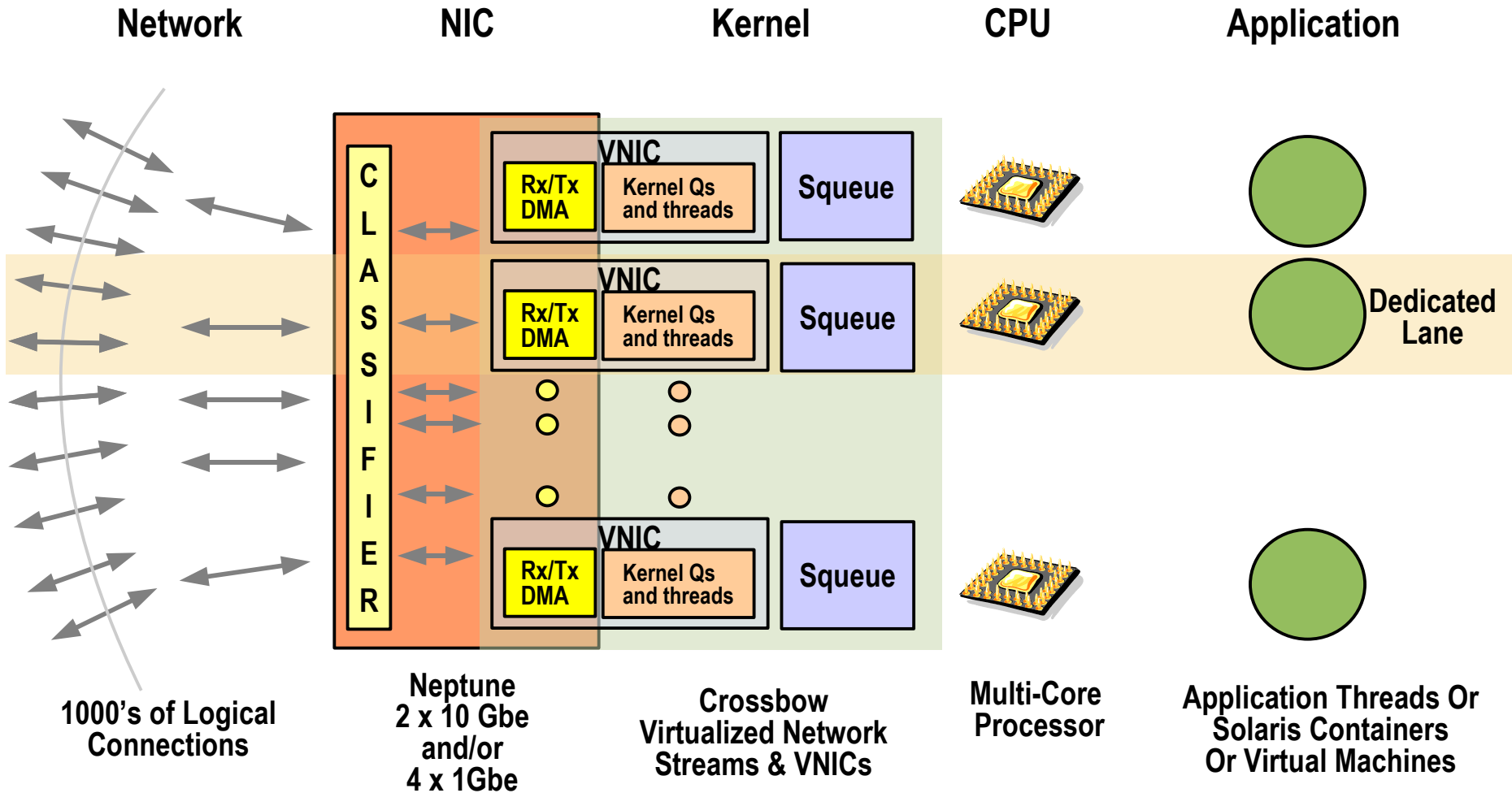
intr	ithr	csw	icsw	migr	smtx	srw	syscl	usr	sys	wt	idl
10818	8607	4558	1547	161	1797	289	19112	17	69	0	12

Mpstat (Crossbow based driver)

intr	ithr	csw	icsw	migr	smtx	srw	syscl	usr	sys	wt	idl
2823	1489	875	151	93	261	1	19825	15	57	0	27

- Use Dynamic polling for B/W partitioning and isolation without any support from switches and routers

Parallelized Stack: Made for Cores/Threads



Virtual Network Containers

Virtualization

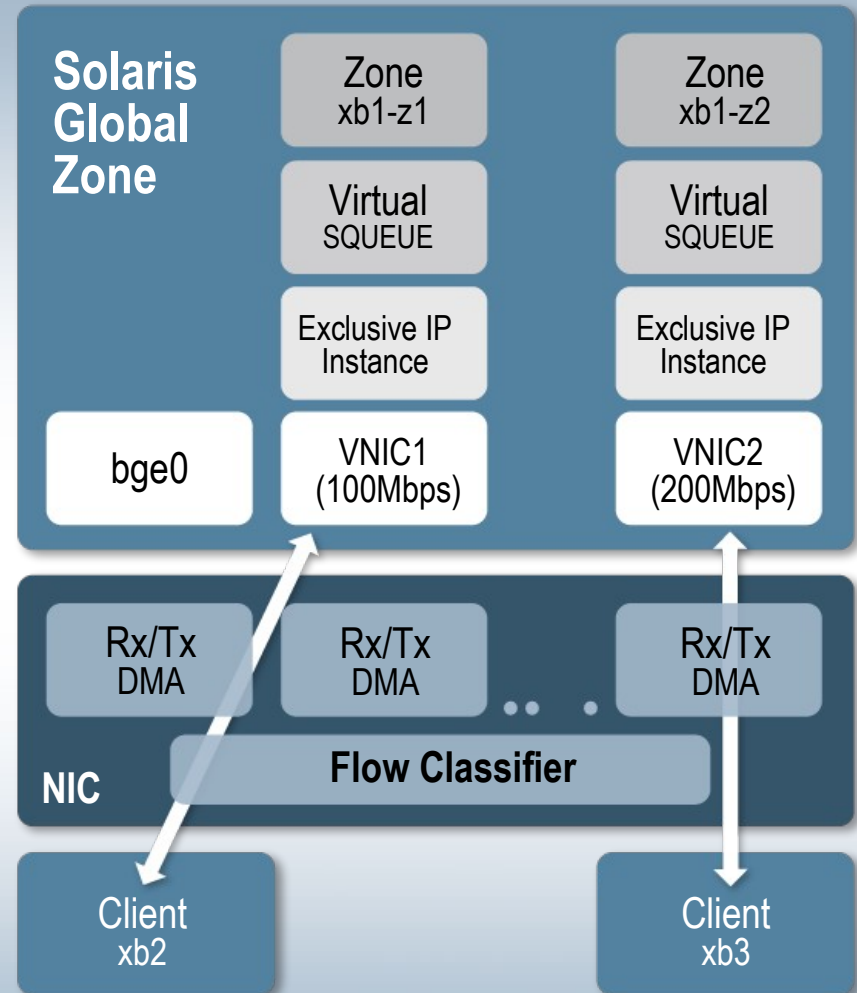
- Exclusive IP Stack
- Virtual NICs
- Virtual Switches

Resource Control

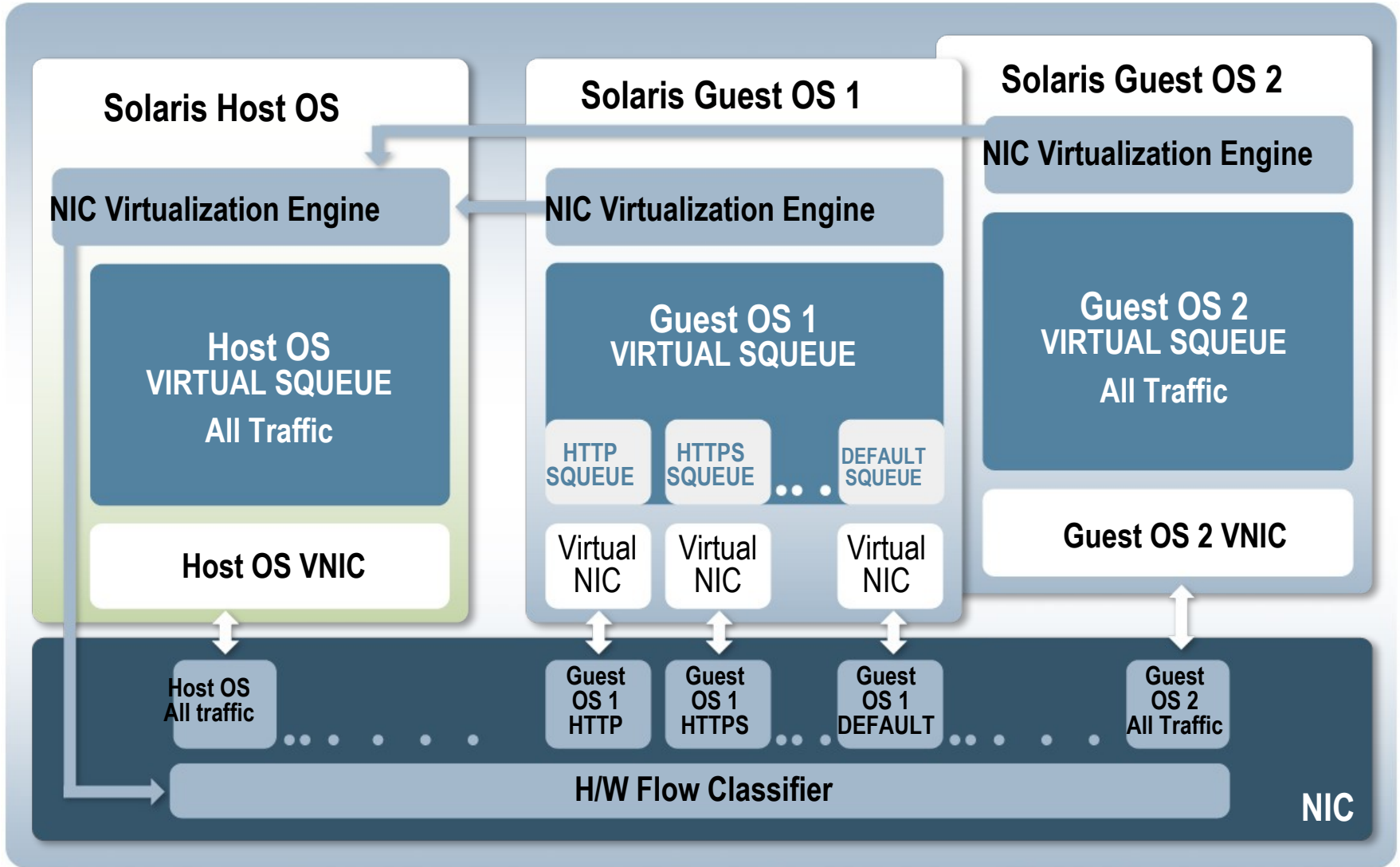
- Bandwidth Limits & Guarantees
- Priority
- Dedicated CPUs

Observability

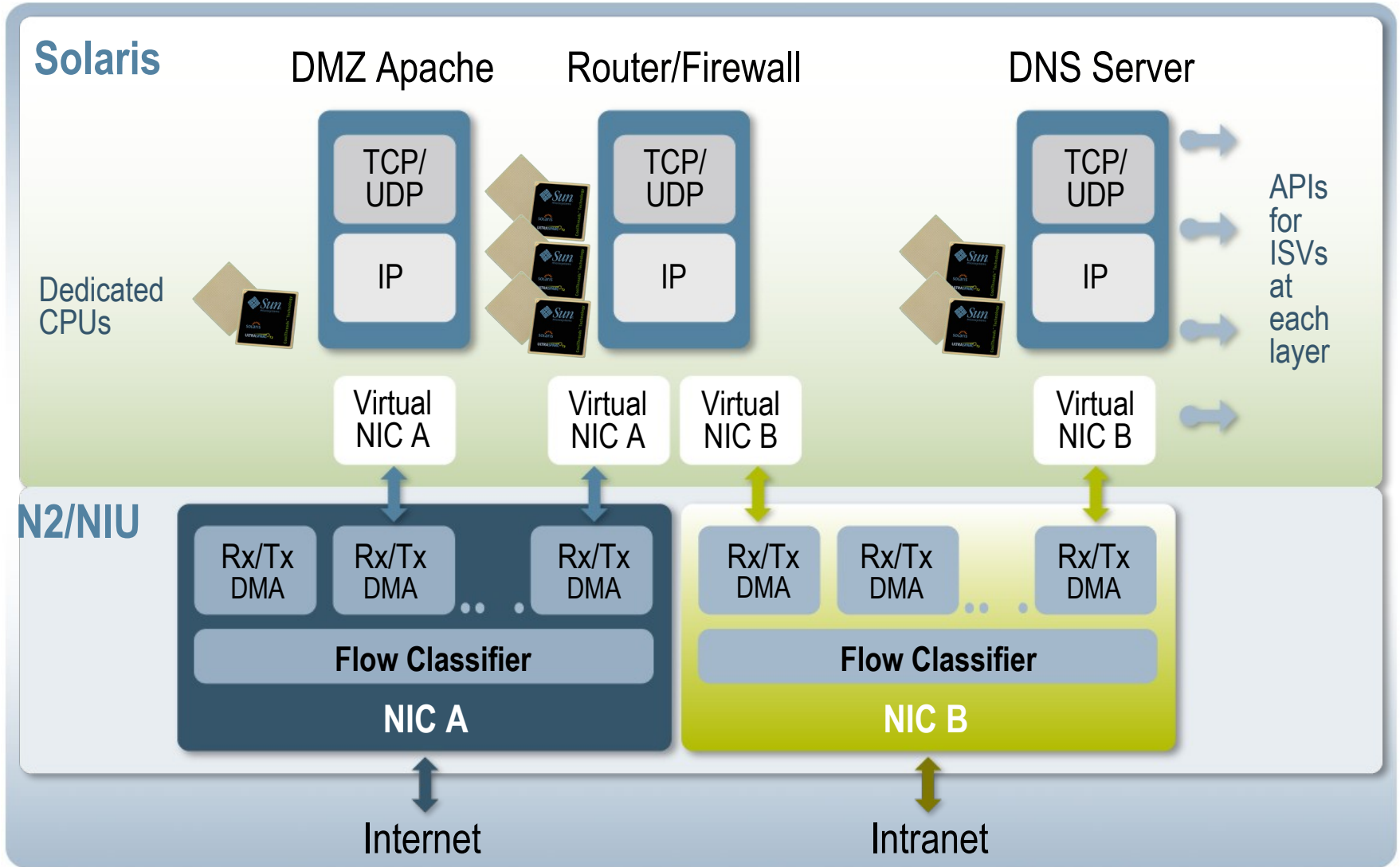
- Watch real time usage for each VNIC
- Snmp and Kstat per VNIC
- History at no cost



Virtual Machines

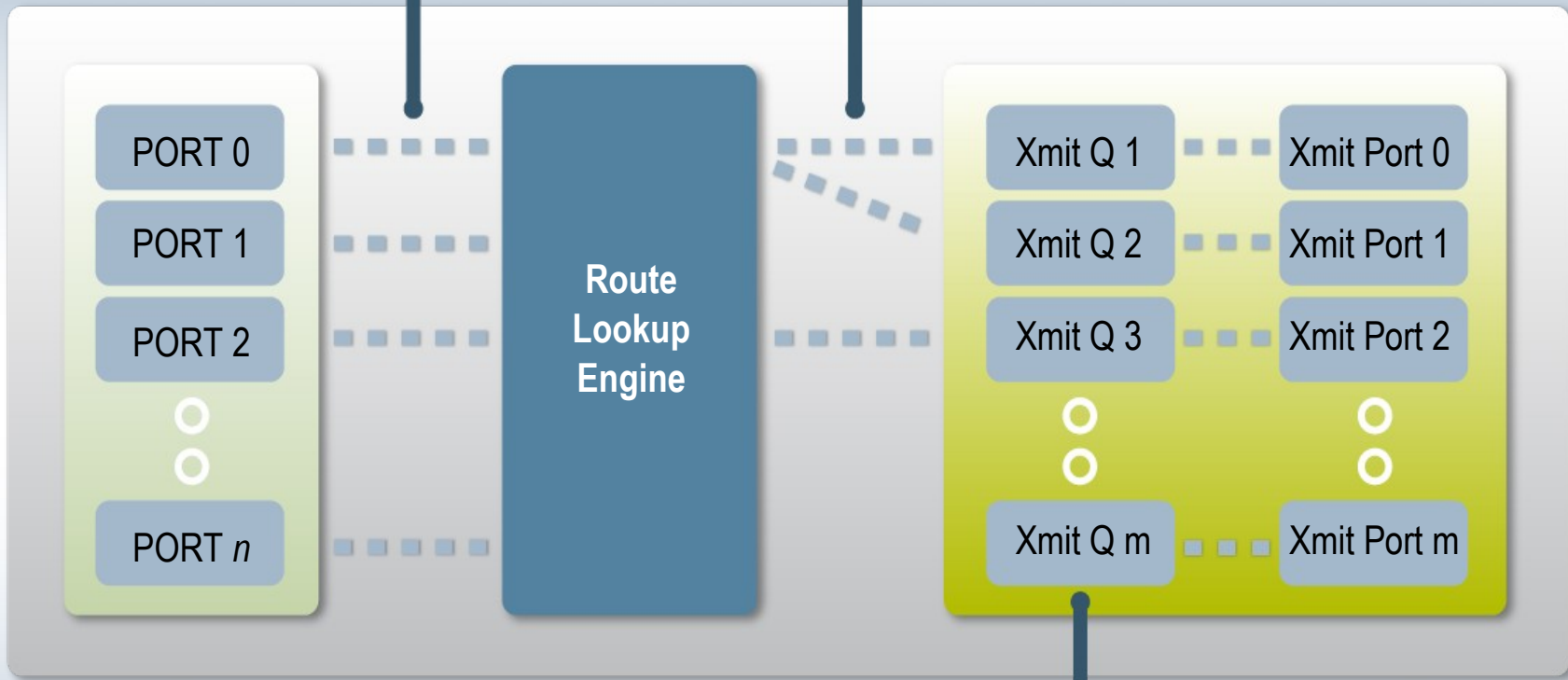


Network Machines



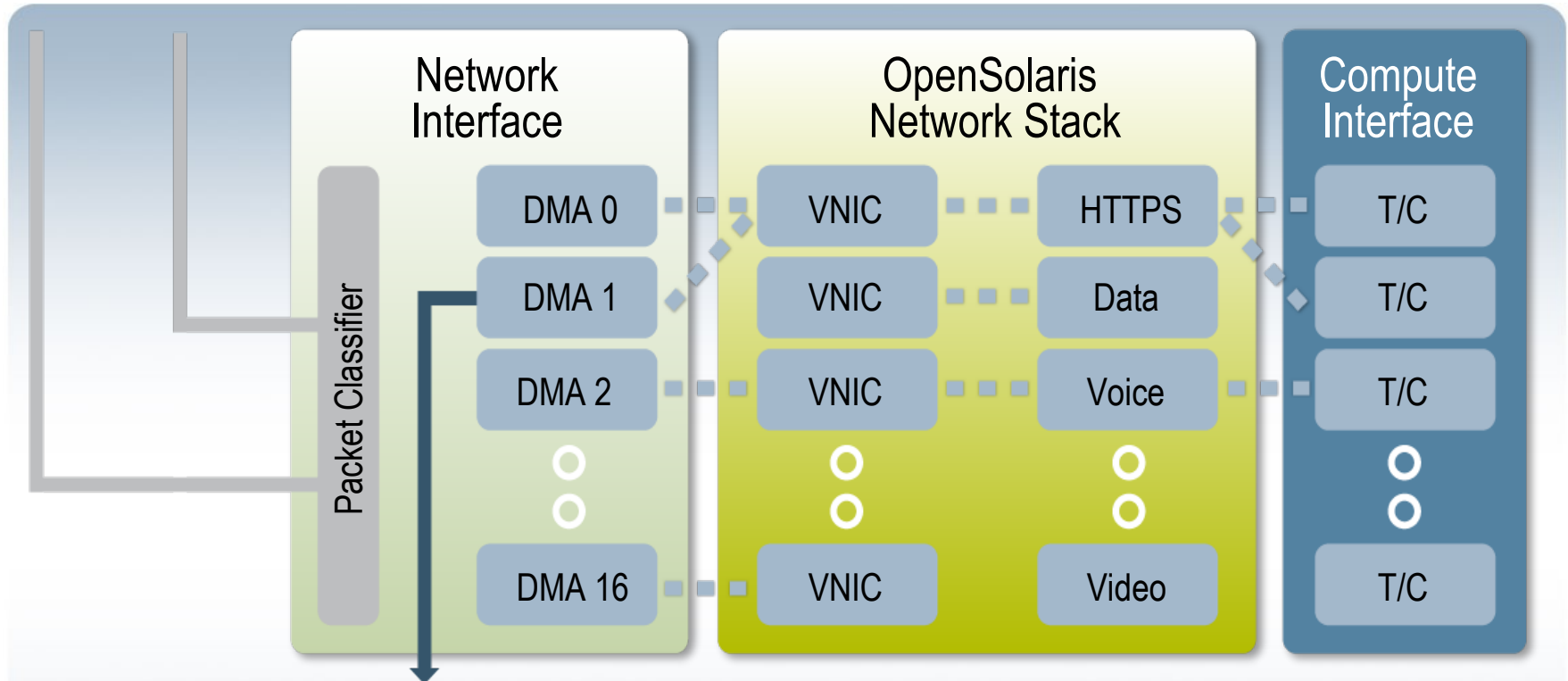
Anatomy of a Typical Router

Major cost paid on bringing the packet in and route lookup etc.



Packets are dropped on Xmit Queues

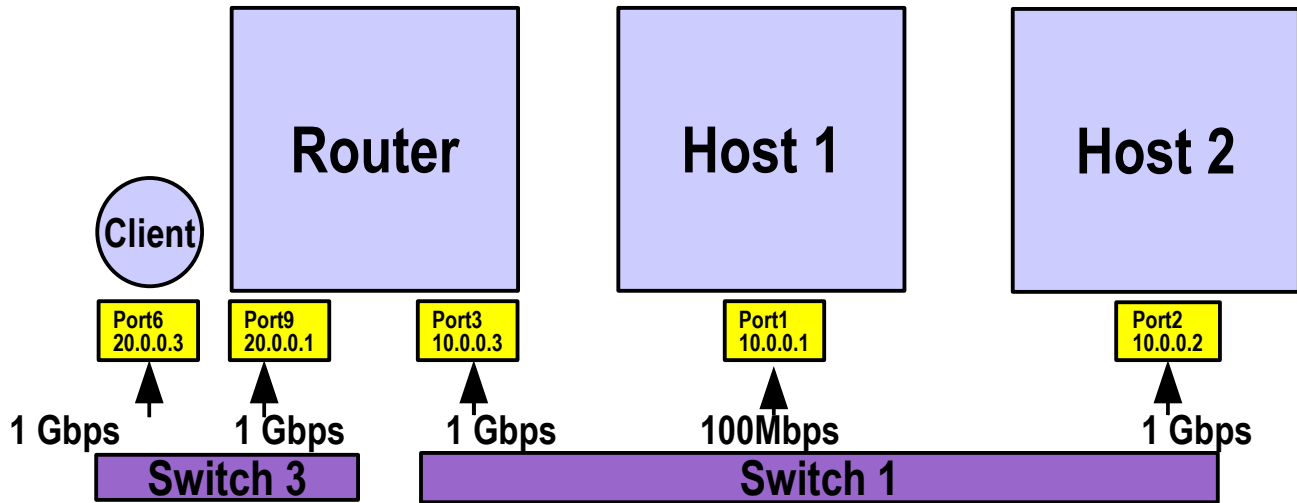
Anatomy of Crossbow Router



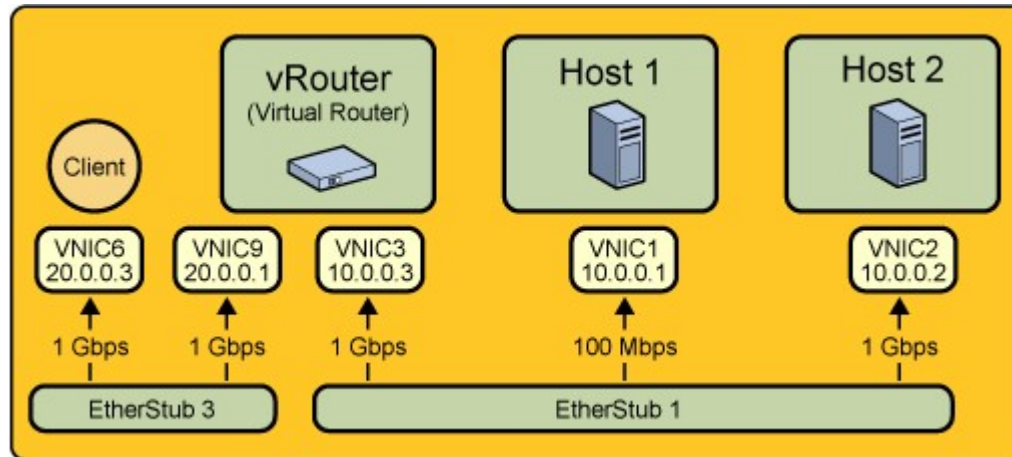
Policy-based packet drop on ingress—zero cost drop.

- Explicit mapping of Compute Threads or Cores to Network Threads creating virtual “express lanes”
- Dedicated BW, Priority, and CPU/Threads assigned to each lane
- High performance packet classification

Physical Network



Crossbow Network in a Box



Example VNIC Usage

- Done using `dladm(1M)`, as with other data-link interface administration

```
Terminal
```

```
# dladm create-vnic -d bge1 vnic1
# dladm create-vnic -d bge1 -m random -p maxbw=100M -p cpus=4,5,6 vnic2
# dladm create-etherstub vswitch1
# dladm show-etherstub
LINK
vswitch1
# dladm create-vnic -d vswitch1 -p maxbw=1000M vnic3
# dladm show-vnic
```

LINK	OVER	MACTYPE	MACVALUE	BANDWIDTH	CPUS
vnic1	bge1	factory	0:1:2:3:4:5	-	-
vnic2	bge1	random	2:5:6:7:8:9	max=100M	4,5,6
vnic3	vswitch1	random	4:3:4:7:0:1	max=1000M	-

Join Us. ...

- Our communities and projects are open on OpenSolaris.org:
 - > CrossBow: <http://opensolaris.org/os/project/crossbow>
 - > VNM: <http://opensolaris.org/os/project/vnm>
 - > Networking:
<http://opensolaris.org/os/community/networking>
- Where you will find:
 - > Lively discussions, design docs, FAQs, source code drops, binary releases, etc...