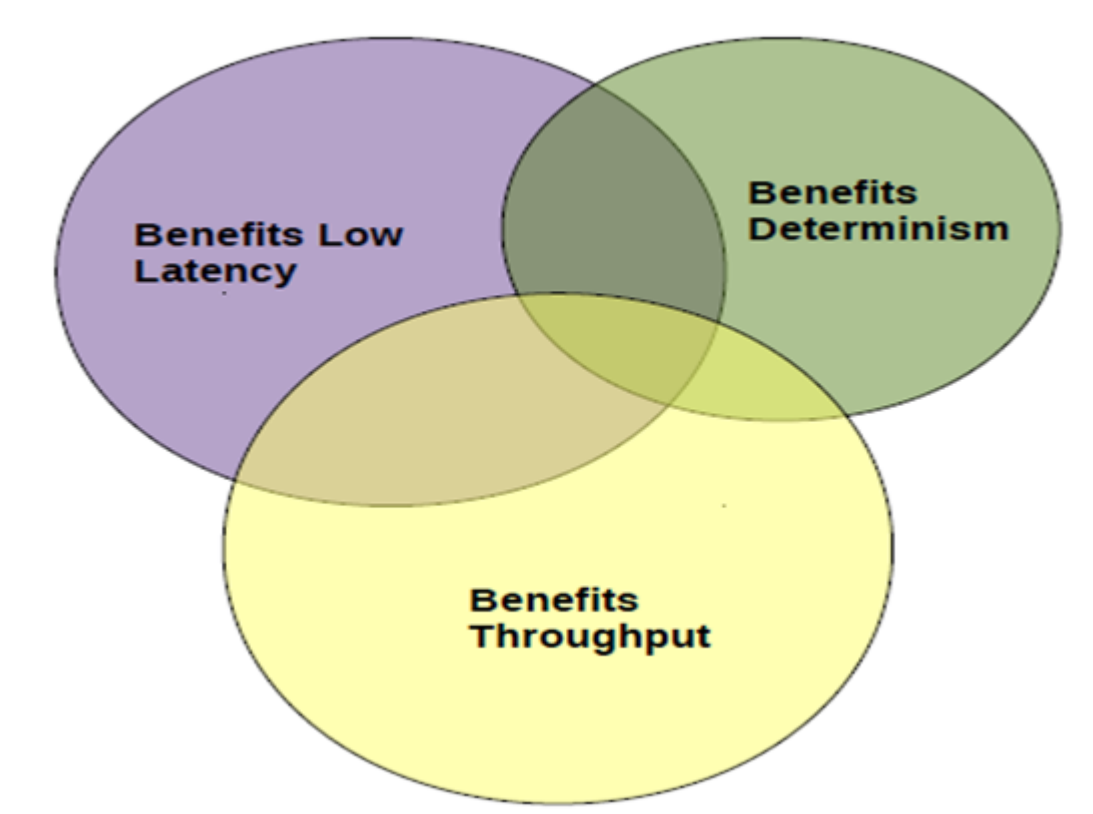
SACC
2012中国系统架构师大会
SYSTEM ARCHITECT CONFERENCE CHINA 2012
架构设计 · 自动化运维 · 云计算

打造Linux下的高性能网络

北京酷锐达信息技术有限公司
技术总监 史应生
shiys@solutionware.com.cn

BY DEFAULT, LINUX NETWORKING NOT TUNED FOR MAX PERFORMANCE, MORE FOR RELIABILITY

Trade-off :Low Latency, throughput, determinism



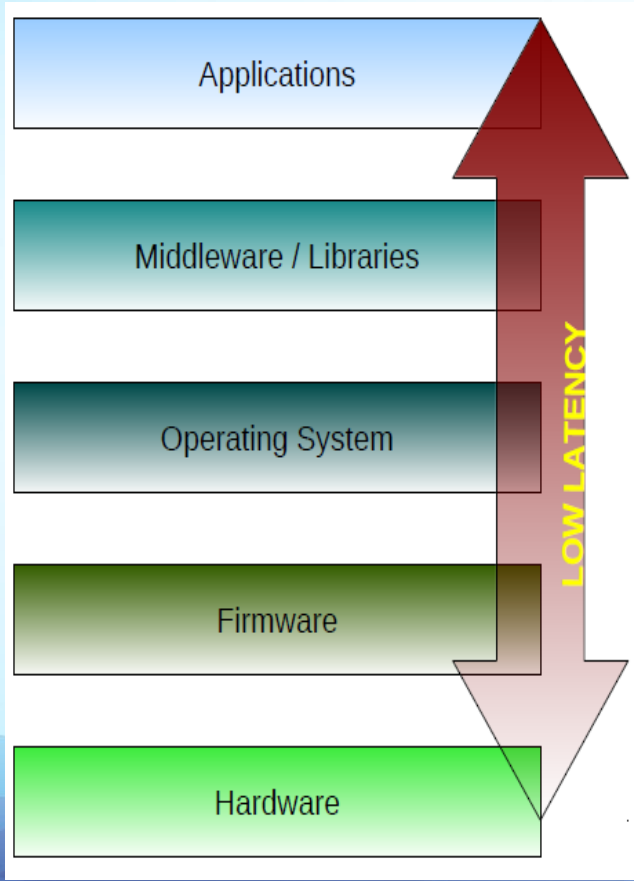
Performance Goals

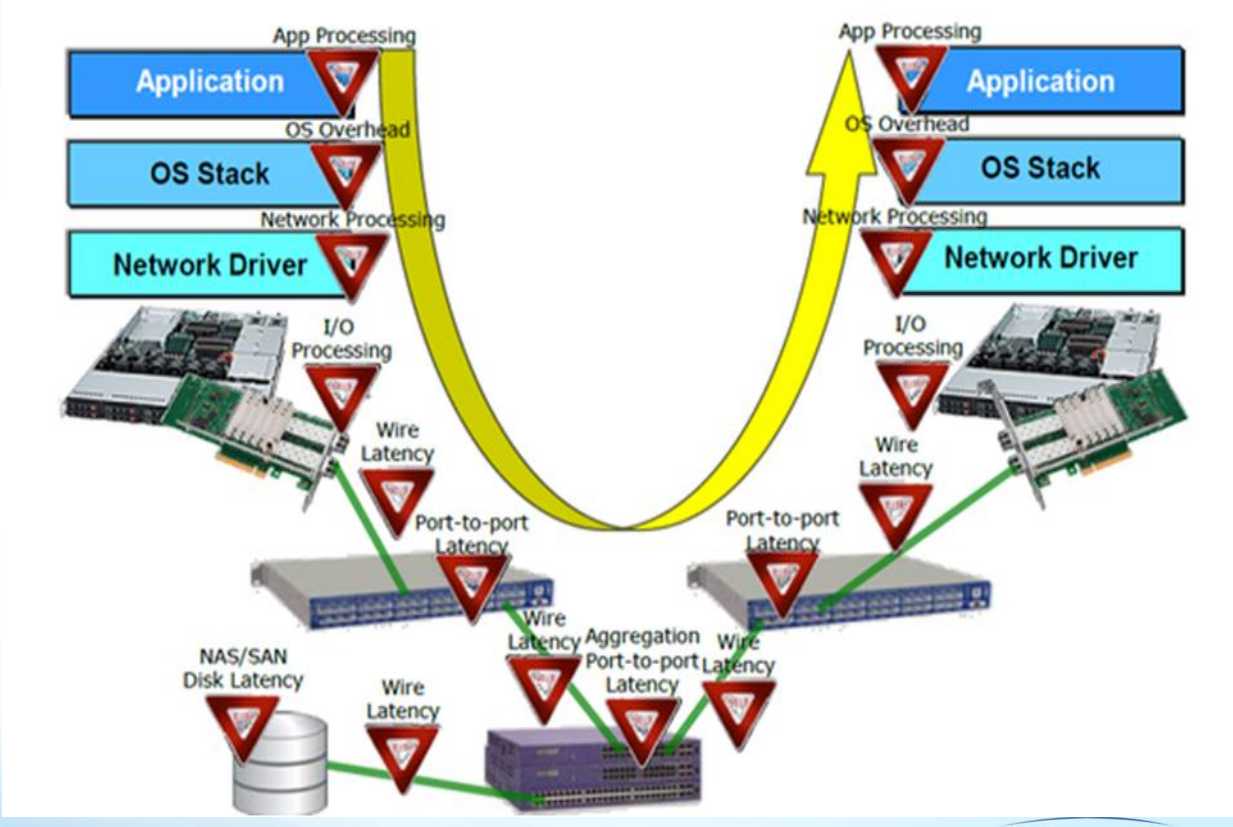
- **Throughput**
  - Optimize for **best average**
  - Default design criteria for most operating systems
  - "how much can you do at a time? "
- **Low Latency**
  - Optimize for **best minimum**
  - Minimize execution times for certain paths
  - "what's the fastest we can push a packet out? "
- **Determinism**
  - Optimize for **best (lowest) maximum**
  - Fewest/lowest outliers
  - "what's the maximum time it will take?"

State of the Art NIC characteristics

- 56 Gigabits per second (Unix network stack was designed for 10Mbits)
- 4-7 Gigabytes per second (Unix: 1 MB/s)
- >8 million packets per second (Unix: ~1000 packets per second).
- Less than a microsecond per packet for processing.

Latency Factors





BIOS Settings for Low Latency

System Setup Screen	Setting	Default	Recommended Alternative for Low-Latency Environments
Processor Settings	Logical Processor	Enabled	Disabled
Processor Settings	Turbo Mode	Enabled	Disabled[3]
Processor Settings	C-States	Enabled	Disabled
Processor Settings	C1E	Enabled	Disabled
Power Management	Power Management	Active Power Controller	Maximum Performance

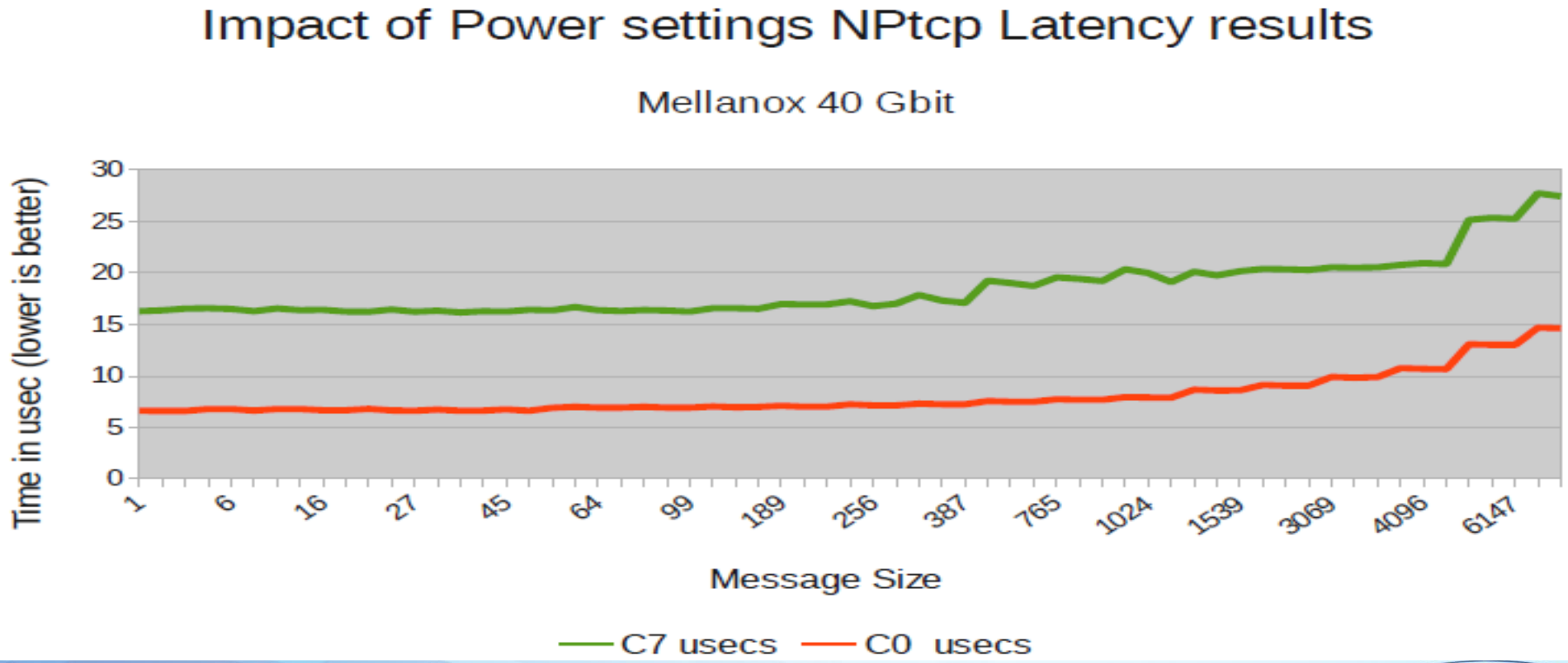
CSTATE default – C7 on this config

pk	cor	CPU	%c0	GHz	TSC	%c1	%c3	%c6	%c7	%pc2	%pc3	%pc6	%pc7	SMIs
			0.04	1.43	2.19	0.08	0.00	0.00	99.89	4.46	0.00	93.94	0.00	0
0	0	0	0.01	1.28	2.19	0.93	0.01	0.00	98.66	3.13	0.01	93.91	0.00	0
0	1	1	0.04	1.66	2.19	0.06	0.00	0.00	99.91	3.13	0.01	93.91	0.00	0
0	2	2	0.01	1.73	2.19	0.01	0.00	0.00	99.98	3.13	0.01	93.92	0.00	0
0	3	3	0.01	1.72	2.19	0.02	0.01	0.00	99.96	3.13	0.01	93.92	0.00	0
0	4	4	0.01	1.85	2.19	0.01	0.00	0.00	99.98	3.13	0.01	93.92	0.00	0
0	5	5	0.01	1.94	2.19	0.01	0.00	0.00	99.98	3.13	0.01	93.91	0.00	0
0	6	6	0.01	1.92	2.19	0.02	0.00	0.00	99.98	3.13	0.01	93.91	0.00	0
0	7	7	0.01	1.76	2.19	0.01	0.00	0.00	99.98	3.13	0.01	93.91	0.00	0
1	0	8	0.01	1.71	2.19	0.02	0.01	0.00	99.96	5.80	0.00	93.96	0.00	0
1	1	9	0.01	1.69	2.19	0.02	0.01	0.00	99.97	5.80	0.00	93.96	0.00	0
1	2	10	0.01	1.75	2.19	0.02	0.00	0.00	99.97	5.80	0.00	93.96	0.00	0
1	3	11	0.01	1.83	2.19	0.02	0.00	0.00	99.97	5.80	0.00	93.96	0.00	0
1	4	12	0.01	1.84	2.19	0.02	0.00	0.00	99.97	5.80	0.00	93.96	0.00	0
1	5	13	0.01	1.91	2.19	0.02	0.00	0.00	99.98	5.80	0.00	93.96	0.00	0
1	6	14	0.01	1.96	2.19	0.02	0.00	0.00	99.98	5.80	0.00	93.96	0.00	0
1	7	15	0.01	2.38	2.19	0.03	0.00	0.00	99.96	5.80	0.00	93.96	0.00	0

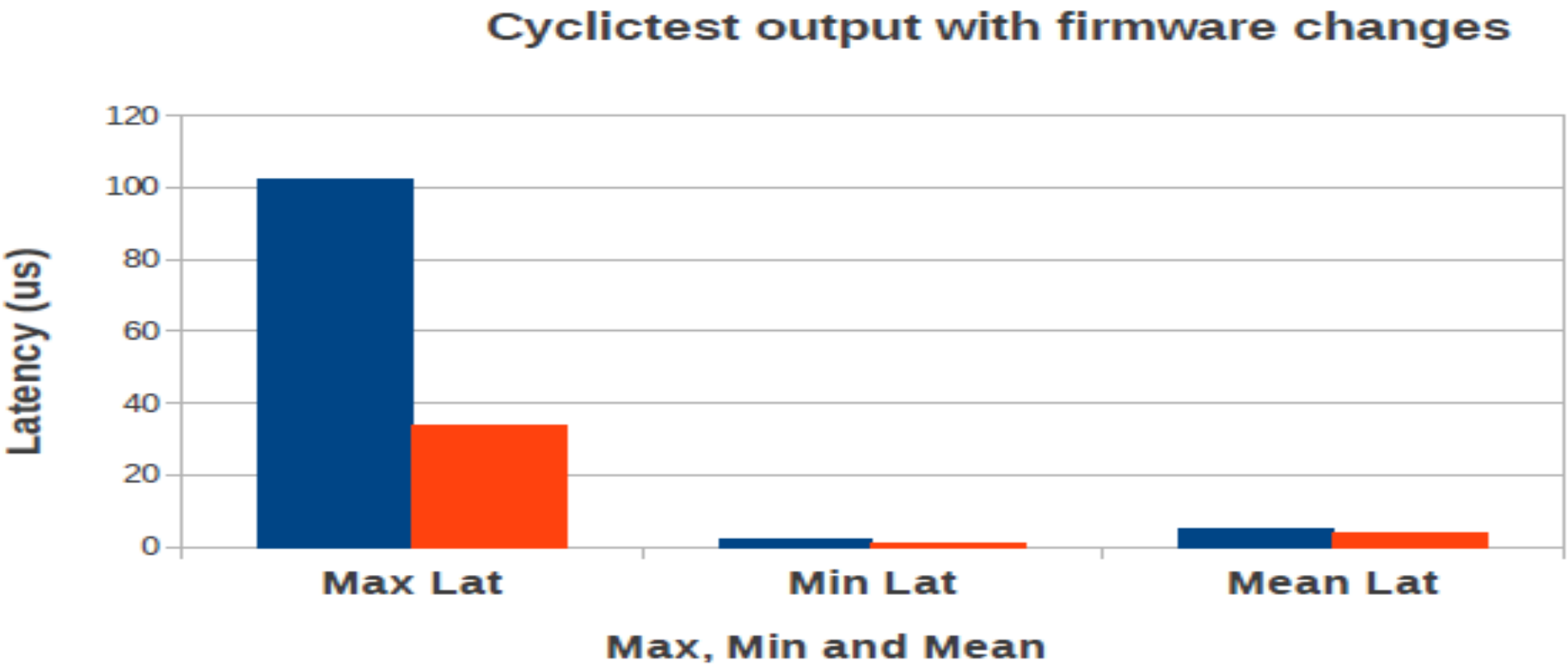
CSTATE disabled – Note speed

pk	cor	CPU	%c0	GHz	TSC	%c1	%c3	%c6	%c7	%pc2	%pc3	%pc6	%pc7	SMIs
			100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	0	0	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	1	1	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	2	2	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	3	3	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	4	4	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	5	5	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	6	6	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
0	7	7	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	0	8	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	1	9	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	2	10	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	3	11	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	4	12	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	5	13	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	6	14	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
1	7	15	100.00	2.69	2.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0

NPtcp latency vs cstates – c7 vs c0



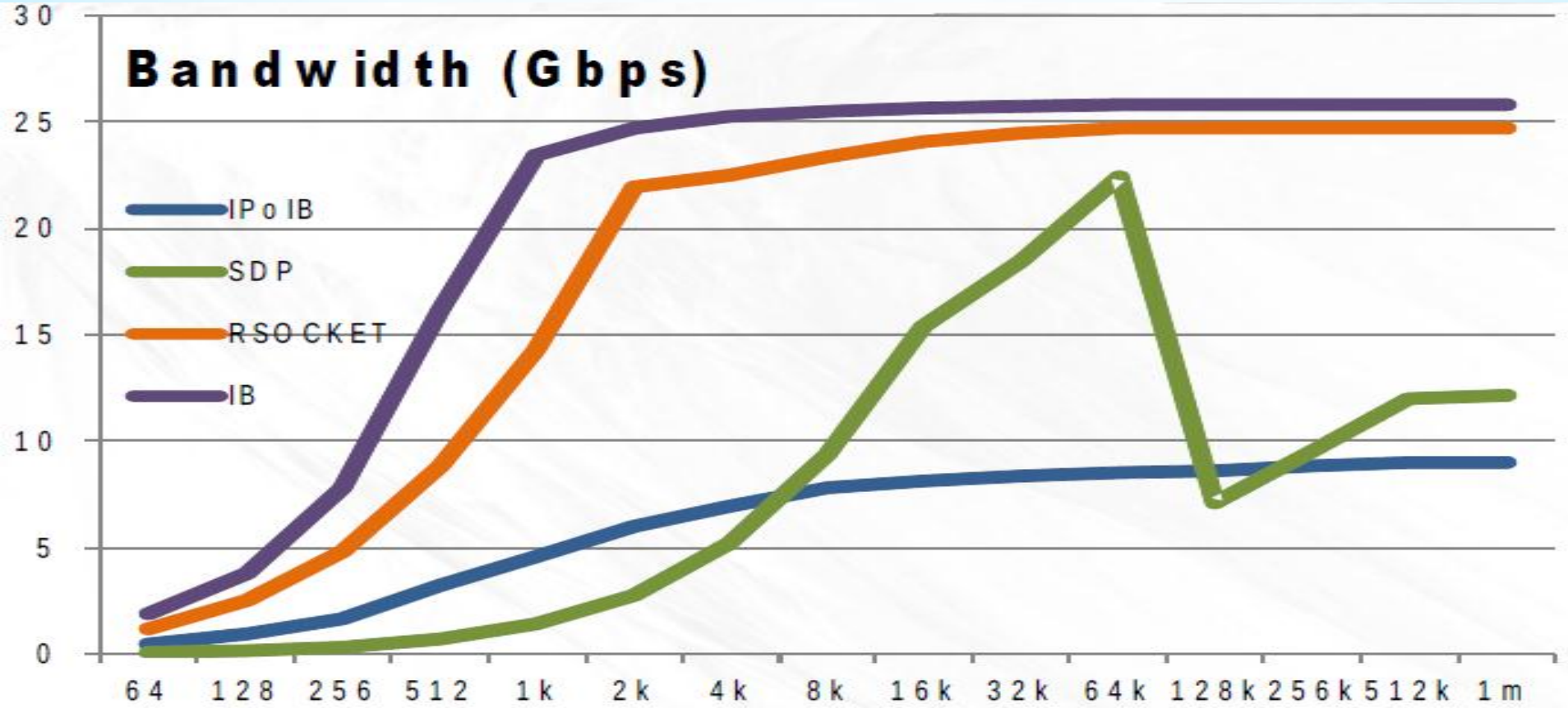
Firmware tuning impact – a drastic picture



Different Technology To Max Performance

- IPOIB = Kernel Sockets layer using IP emulation on Infiniband.
- SDP = Kernel Sockets layer using Infiniband native connection.
- IB = Native Infiniband connection. User space → User Space
- Rsockets = Socket Emulation layer in user space
- Performance comparison shows that kernel processing is detrimental to performance. Bypass is essential.

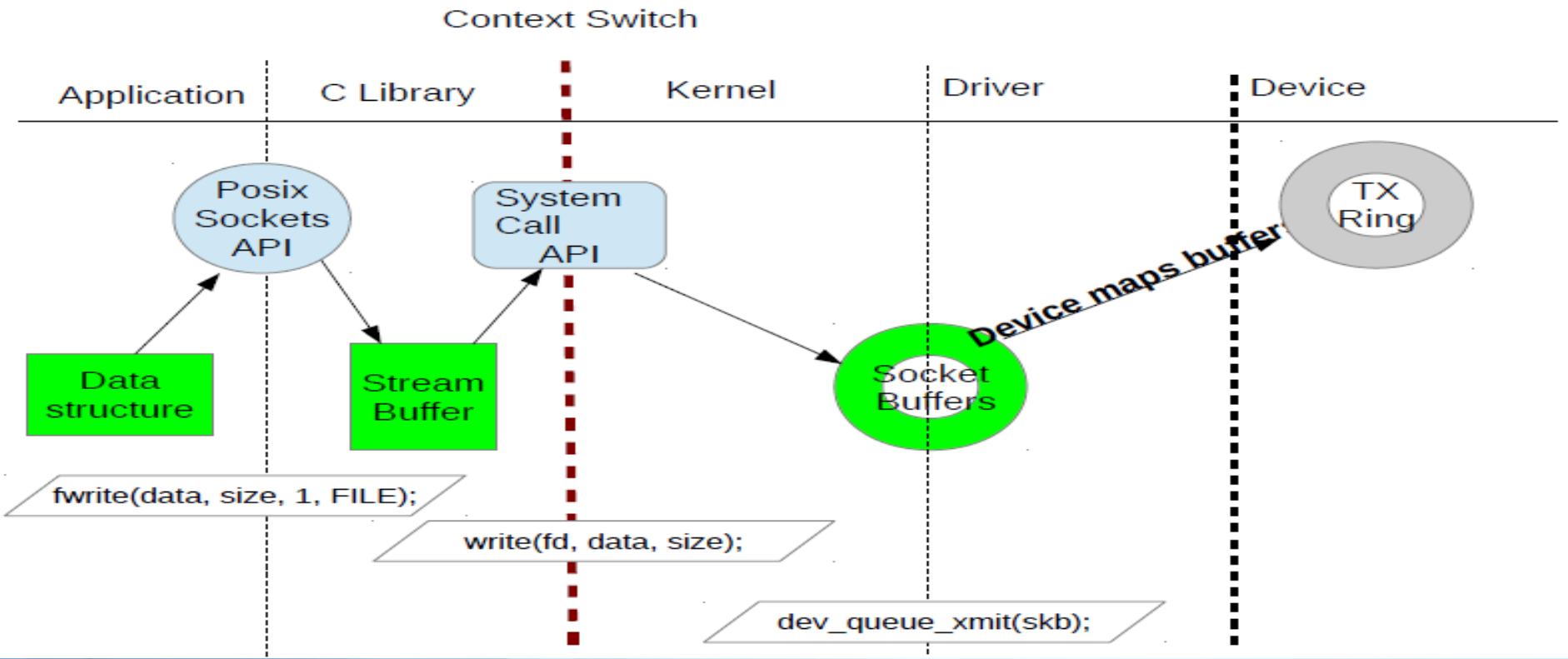
Different Technology To Max Performance



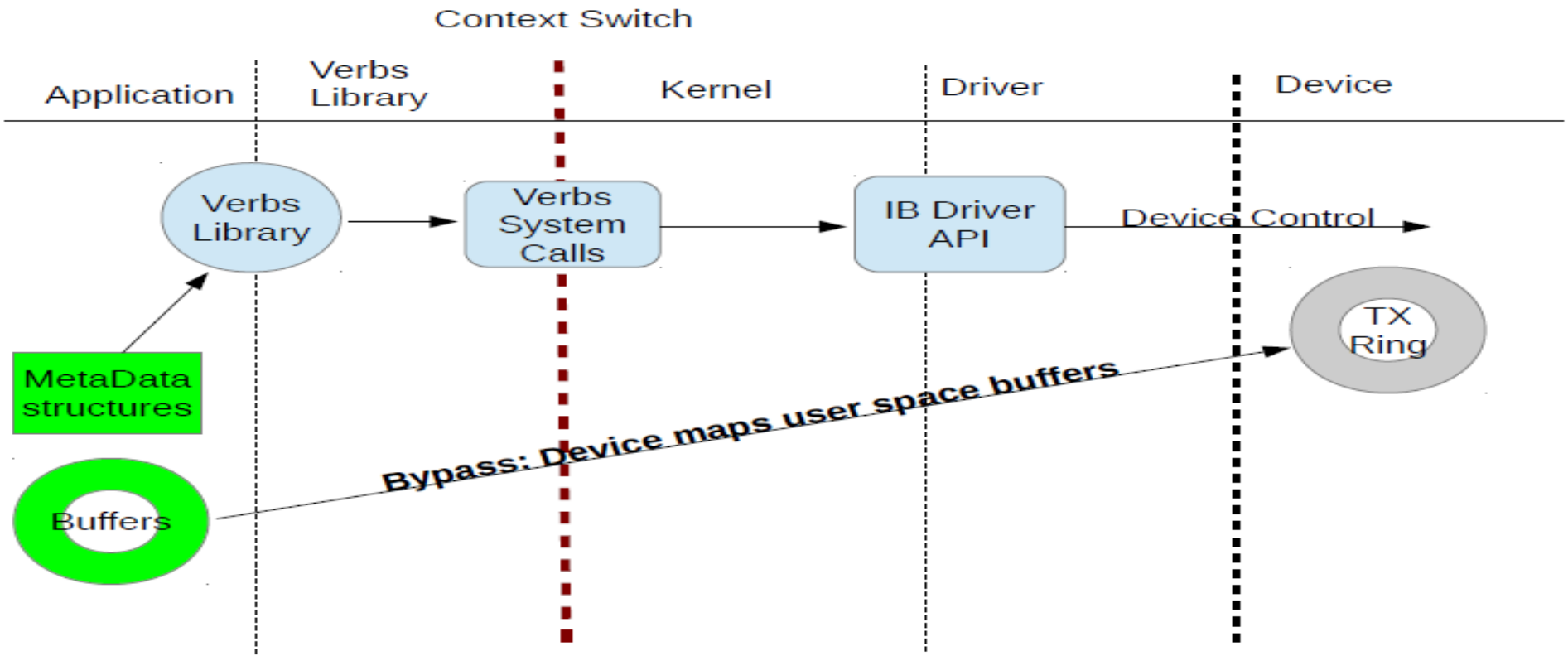
Why bypass the kernel?

- Kernel is too slow and inefficient at high packet rates. Problems already begin at 10G.
- Contemporary devices can map user space memory and perform transfer to user space.
- Kernel must copy data between kernel buffers and userspace.
- Kernel is continually regressing in terms of the overhead of basic system calls and operations. Only new hardware compensates.

Sending a message via the sockets API

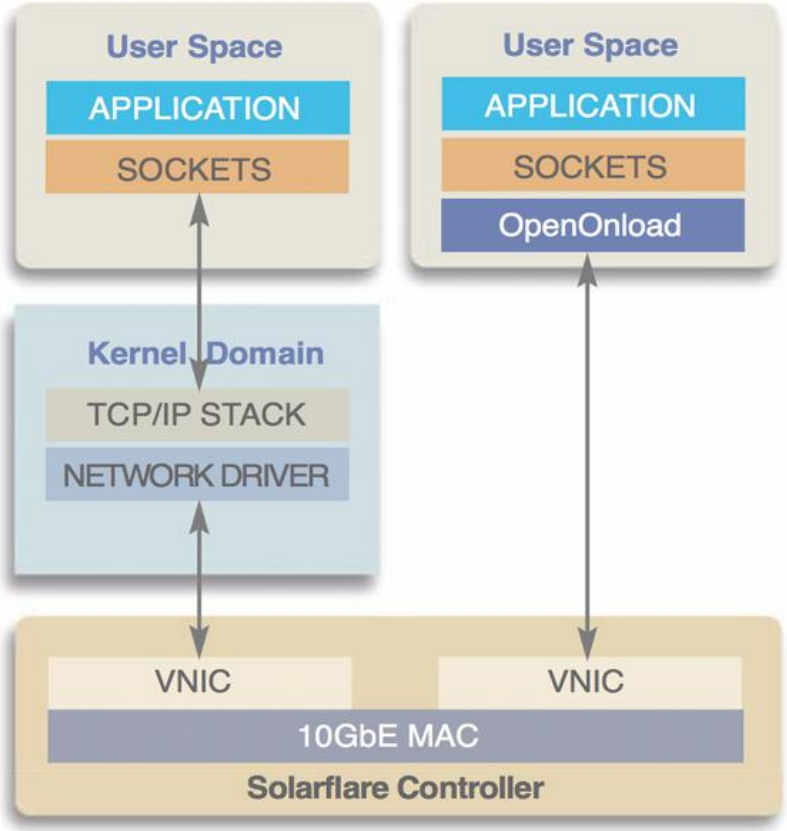


Kernel Bypass



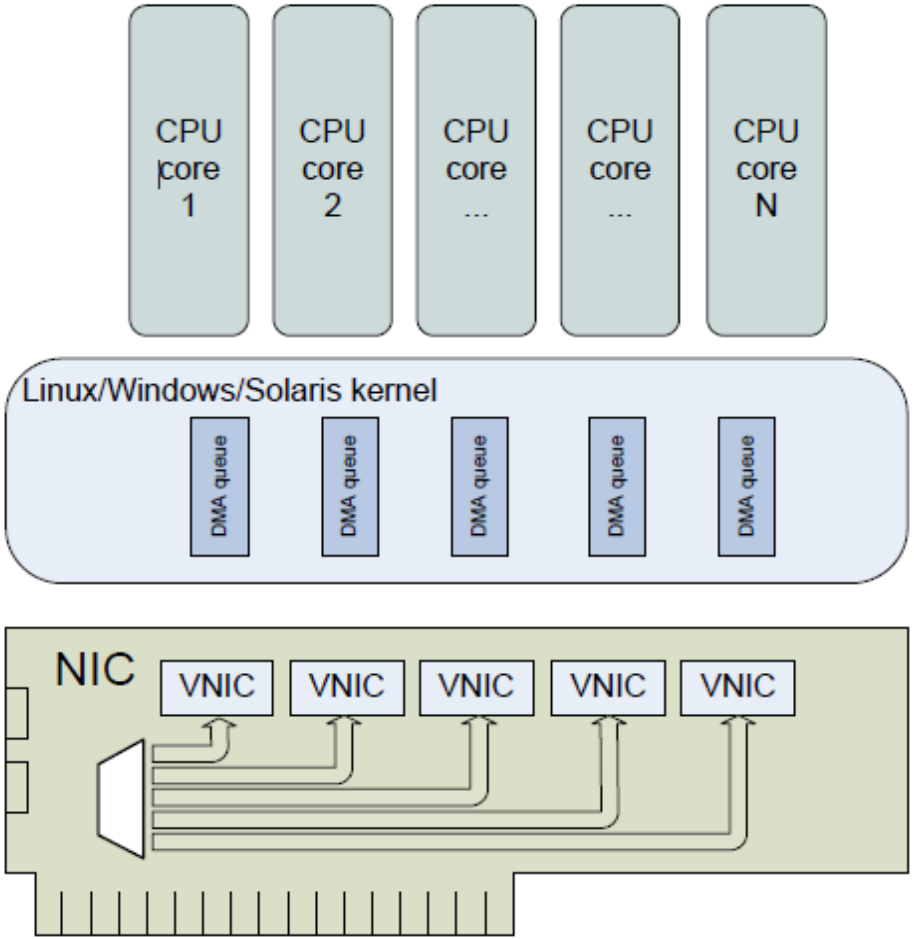
Kernel Bypass

- TCP and UDP Acceleration
  - Kernel bypass
    - App gets direct access to hardware
    - Fewer context switches, copies
  - Benchmarks
    - Reduces latency by 50%
    - Increases message rates 2x to 3x
  - "Real" applications even more benefit
- Compatibility
  - No recompile/application mods
  - Regular Ethernet/IP network
  - Unicast and multicast
  - "Just works"

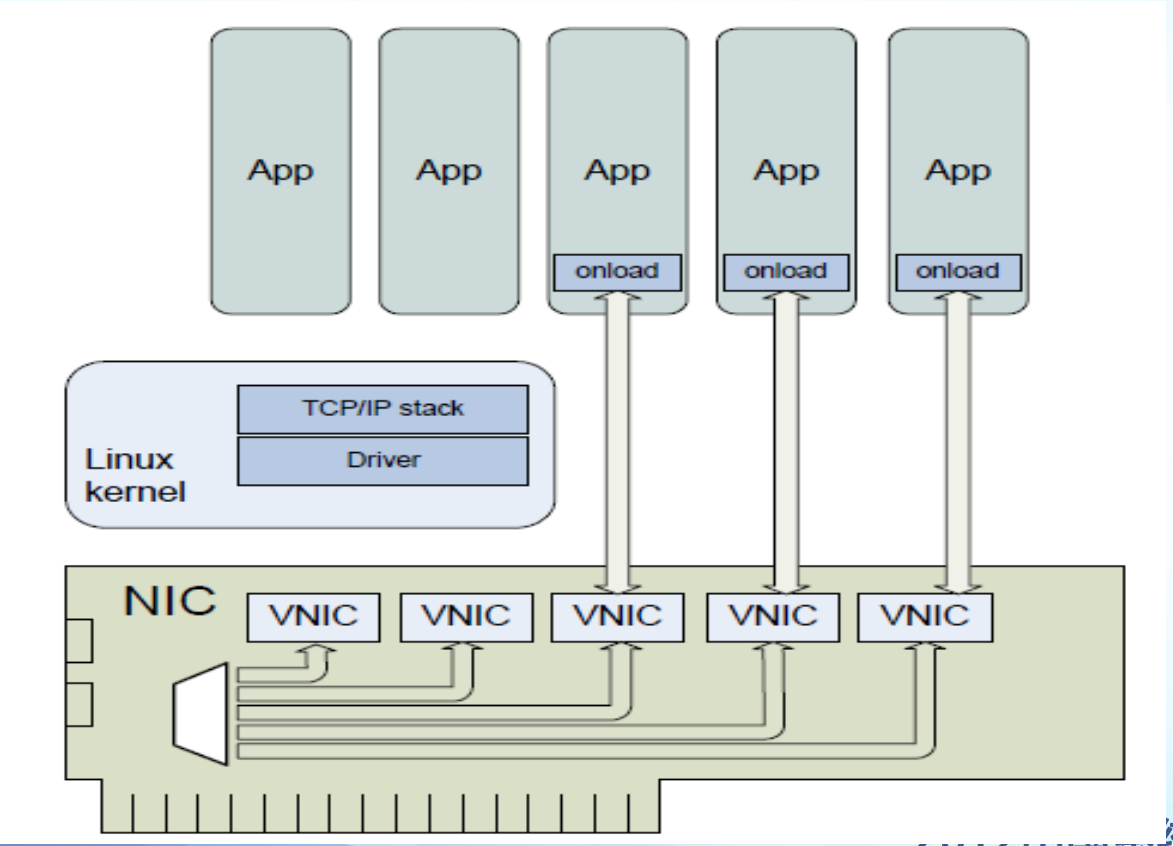


VNIC per CPU core (RSS)

- RX queue per CPU core
- TX queue per CPU core
- Complete CPU core separation
- Performance scales across CPUs

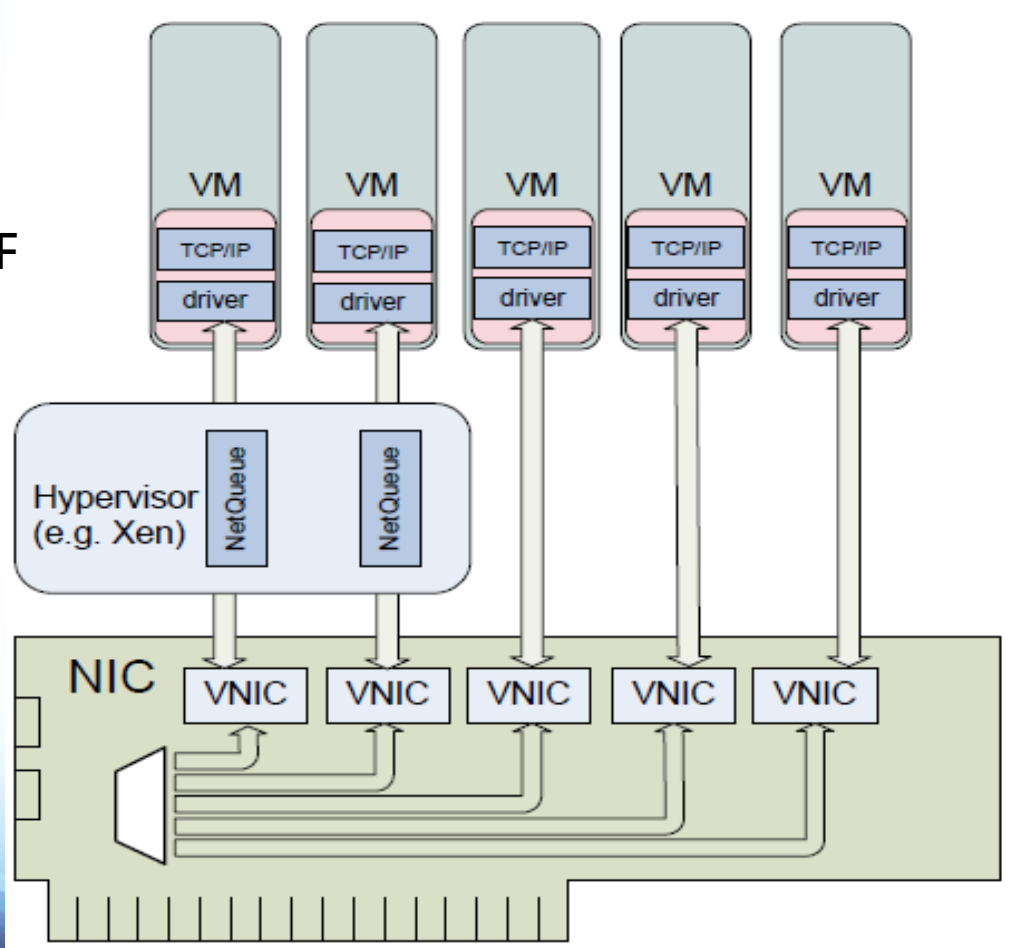


Virtual NICs for application acceleration



Virtual NICs for VM

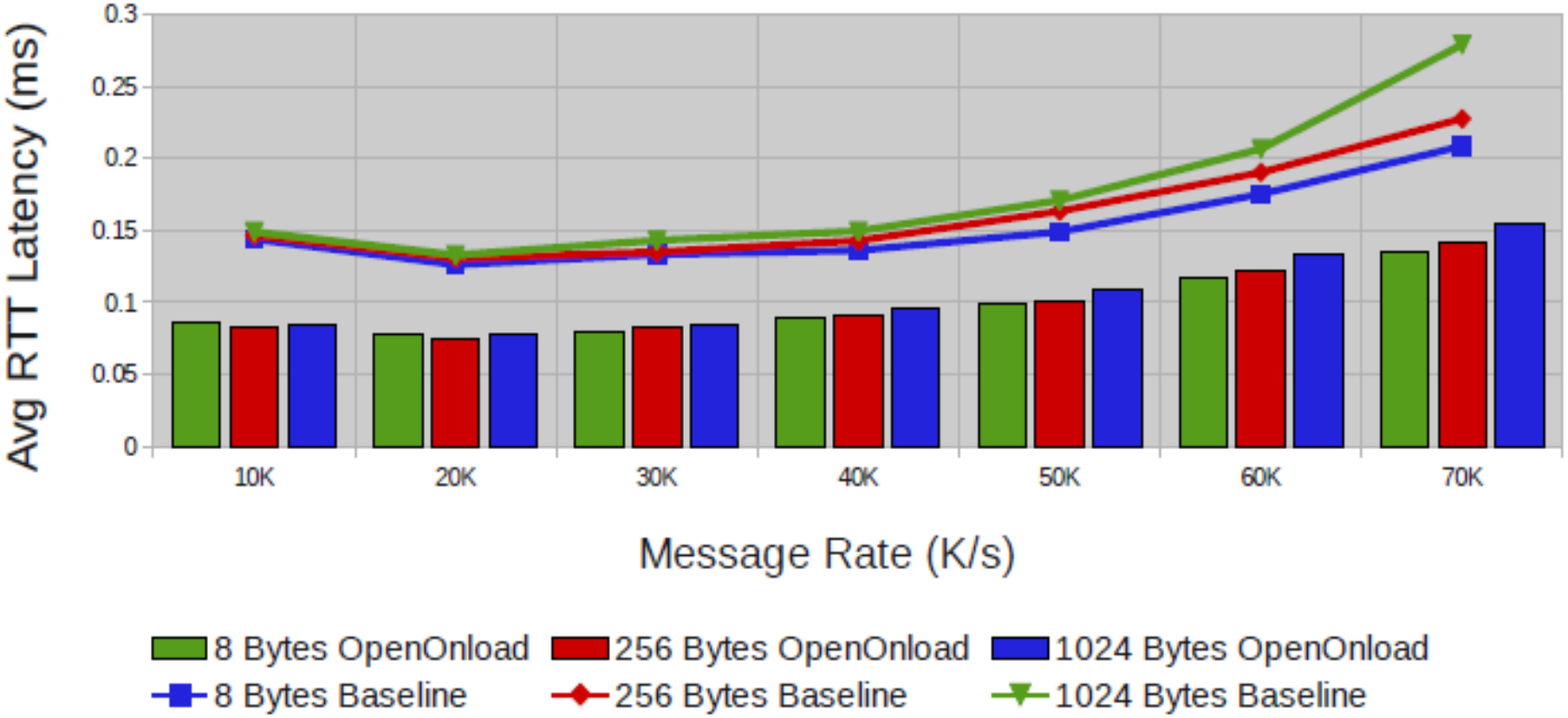
- Same model used for SR-IOV
- In this case VM has direct access to VNIC(s) via SR-IOV VF



Acceleration Middleware

- Just a library and a kernel module
  - No application changes
  - No recompile
  - No kernel patches
  - No protocol changes
- Picks up existing Linux network configuration
  - IP addresses and route table
  - Bonding (aka teaming)/ VLANs
  - Multicast (IGMP)
  - Kernel settings, e.g. socket buffer sizes

Offload – Solarflare OpenOnload



Ethtool – View and change Ethernet card settings

- Works mostly at the HW level
  - ethtool -S – provides HW level stats

    Counters since boot time, create scripts to calculate diffs
  - ethtool -c - Interrupt coalescing
  - ethtool -g - provides ring buffer information
  - ethtool -k - provides hw assist information
  - ethtool -i - provides the driver information

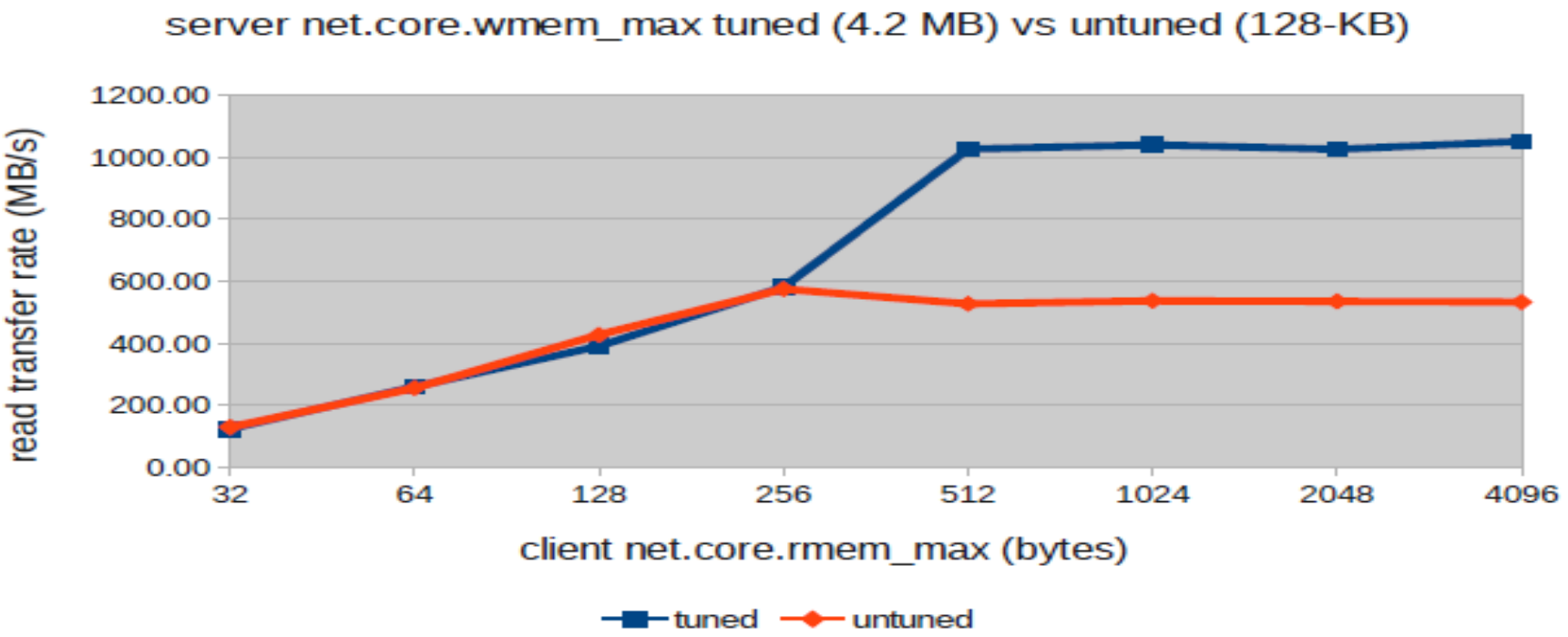
sysctl – popular settings

- These settings are often mentioned in tuning guides
  - net.ipv4.tcp_window_scaling
    - toggles window scaling
  - net.ipv4.tcp_timestamps
    - toggles TCP timestamp support
  - net.ipv4.tcp_sack
    - toggles SACK (Selective ACK) support

sysctl – ” core” memory settings

- CORE memory settings
  - net.core.(r/w)mem_max
    - max size of (r/w)x socket buffer
  - net.core.(r/w)mem_default
    - default (r/w)x size of socket buffer
  - net.core.optmem_max
    - maximum amount of option memory buffers
  - net.core.netdev_max_backlog
    - how many unprocessed rx packets before kernel starts to drop
- These settings also impact UDP !

Effect of net.core.rmem_max on read throughput



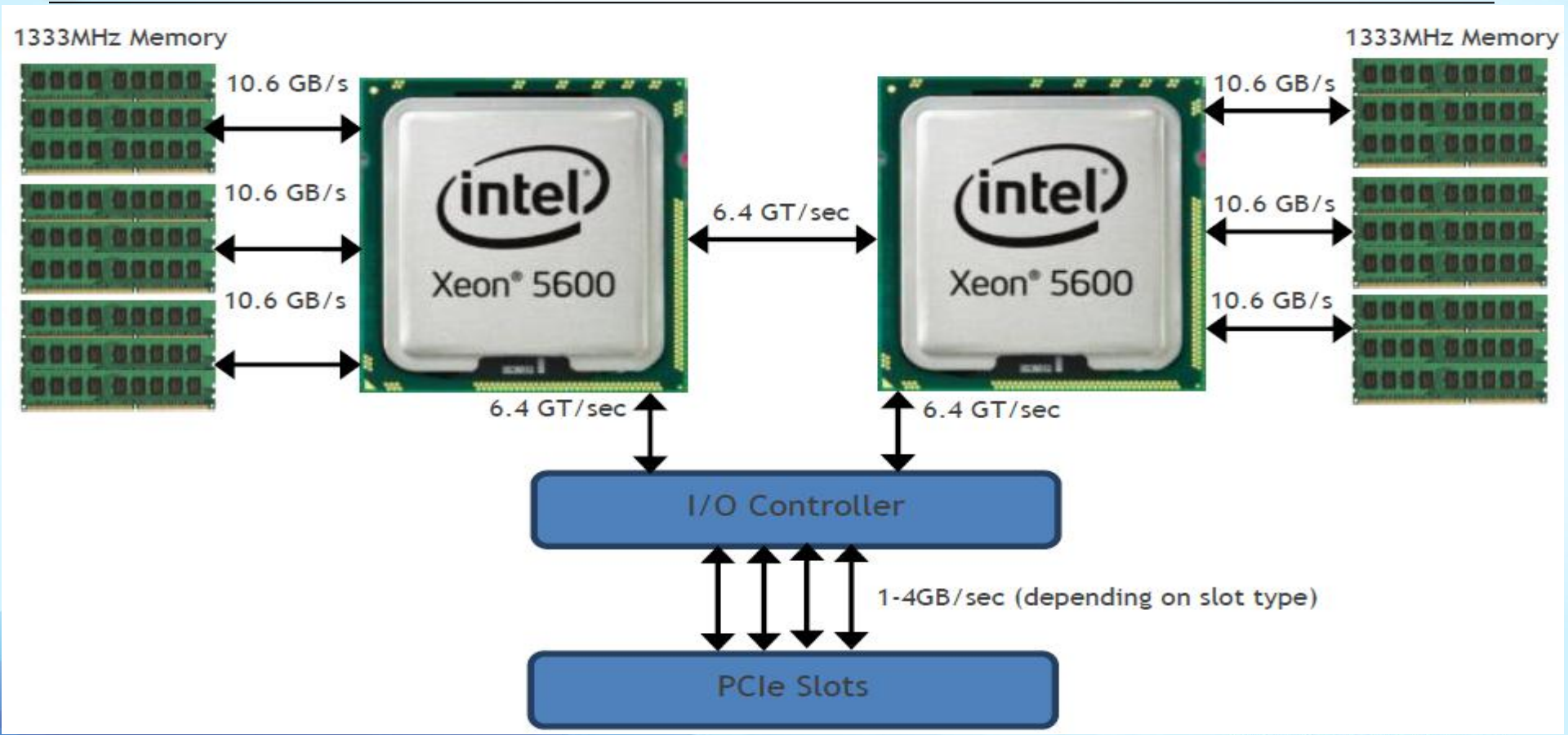
Offload is

- Replacement of what could be done in software with dedicated hardware.
- Overlaps with Bypass because direct device interactions replaces software action in the kernel through the actions of a hardware device.
- Typical case of hardware offload: DMA engines, GPUs, Rendering screens, cryptography, TCP (TOE), FPGAs

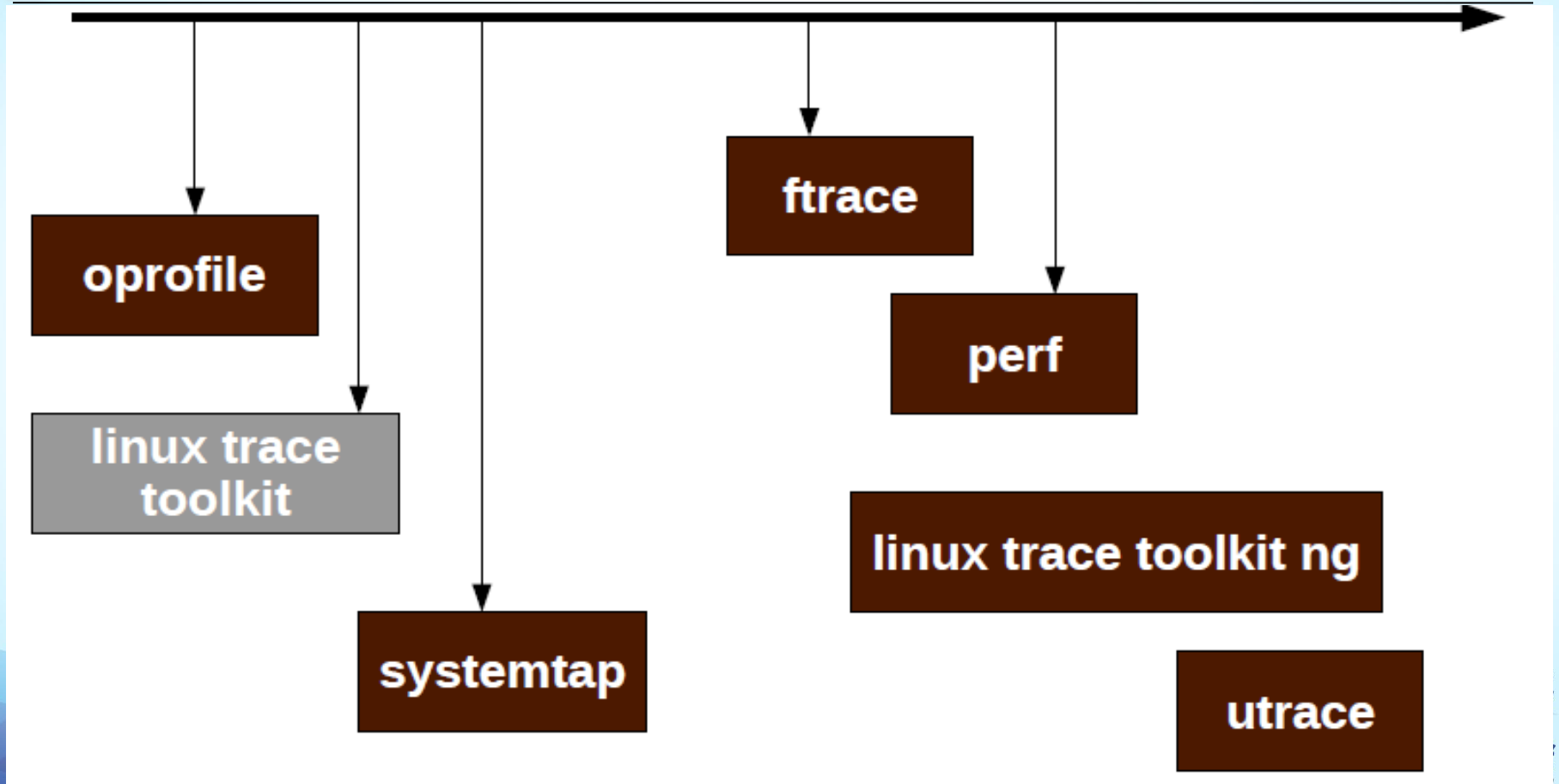
Network Card Hardware Tuning

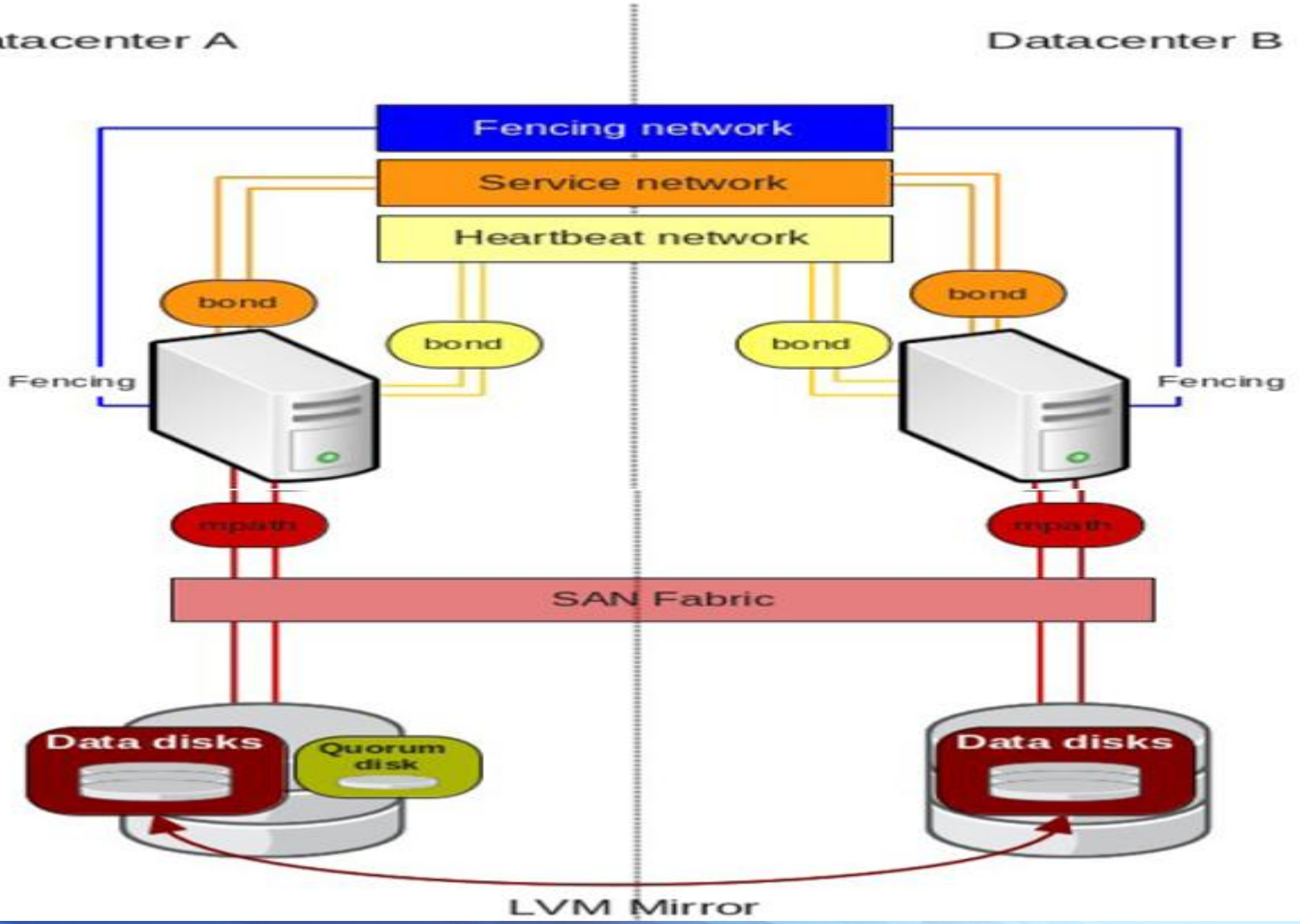
- Jumbo Frames
- Transmission queue
- Multi streams
- interrupt moderation
- RX, TX checksum offload
- TCP Segmentation Offload
- TCP Large Receive Offload (LRO)

Numa In Network Transfer



Performance diagnostic tools




Datacenter A
Datacenter B
Fencing network
Service network
Heartbeat network
bond
bond
bond
bond
Fencing
Fencing
mpath
mpath
SAN Fabric
Data disks
Quorum disk
Data disks
LVM Mirror

Coming in 2013

- 100 GB/sec networking
- >100 GB/sec SSD / Flash devices
- More cores in Intel processors.
- GPUs already support thousands of hardware threads. Newer models will offer more.

Who Are We?

中国领先的Linux全面解决方案提供商

服务形式	现场服务		远程服务(电话和邮件)	
知识传递	专题/定制	通用	基于项目	技术讨论会
	培训		知识传授	
咨询	常规	高级	标准化	
迁移移植	迁移计划	应用移植	应用迁移	
全面解决方案	双机热备高可用集群	系统备份和恢复	统一身份认证和管理	
	自动化定制 安装光盘	系统安全加固	升级/补丁 生命周期管理	系统监控和报警
操作系统	RHEL(红帽企业版Linux操作系统)			
虚拟化云计算	服务器虚拟化解决方案		桌面虚拟化解决方案	
	RHEV (红帽企业版虚拟化)			

Join Us . . . . . .

hr@solutionware.com.cn

酷锐达 SolutionWare

Q&A