Linux on zSeries Performance Update

Session 9390

Steffen Thoss
thoss@de.ibm.com

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.
Enterprise Storage Server
ESCON*
FICON
FICON Express
HiperSockets
IBM*
IBM logo*
IBM eServer
Netfinity*
S/390*
VM/ESA*
WebSphere*
z/VM
zSeries
* Registered trademarks of IBM Corporation
The following are trademarks or registered trademarks of other companies.
Intel is a trademark of the Intel Corporation in the United States and other countries.
Java and all Java-related trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc., in the United States and other countries.
Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation.
Linux is a registered trademark of Linus Torvalds.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
Penguin (Tux) compliments of Larry Ewing.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
UNIX is a registered trademark of The Open Group in the United States and other countries.
* All other products may be trademarks or registered trademarks of their respective companies.

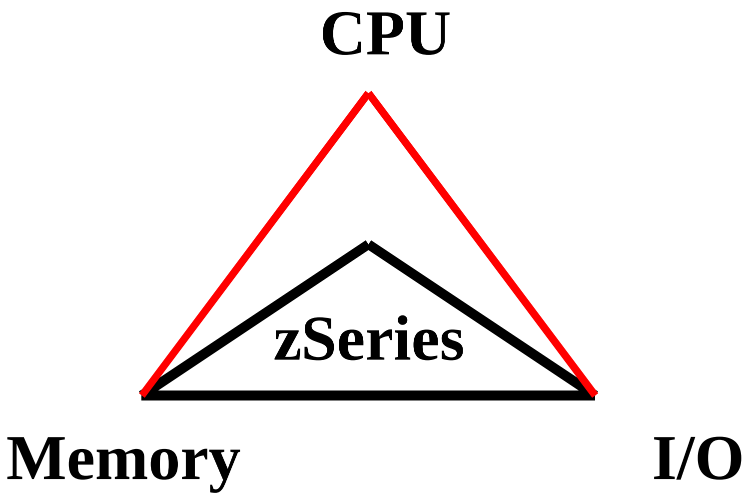
I

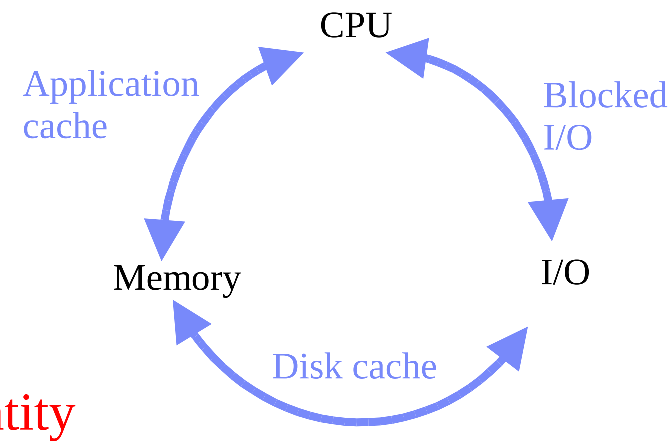
Agenda

- Relative System Capacity
- zSeries Hardware
- Scalability
- Networking
- Context Switches
- Disk I/O
  - Parallel Access Volume (PAV)
  - ESS

Relative System Capacity

- A system provides different types of resources
- Capacity for each resource type may be different
- The ideal machine provides enough capacity of each type
- Don't forget additional Resources (Network, Skilled staff, Money, availability of software, reliability, time ...)

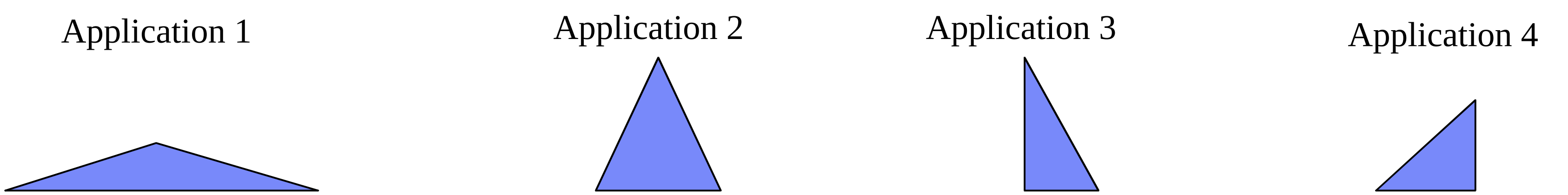




The ideal platform requires a mix of resources in right quantity

Resource Profiles

- Each application has its specific requirements
  - CPU intensive
  - I/O intensive
  - Memory
- Applications can often be tuned to change the resource profile
  - Exchange one resource for the other
  - Requires knowledge about available resources
- Some platforms can be extended better than others
  - Not every platform runs every application well
  - It's not easy to determine the resource profile of an appl.

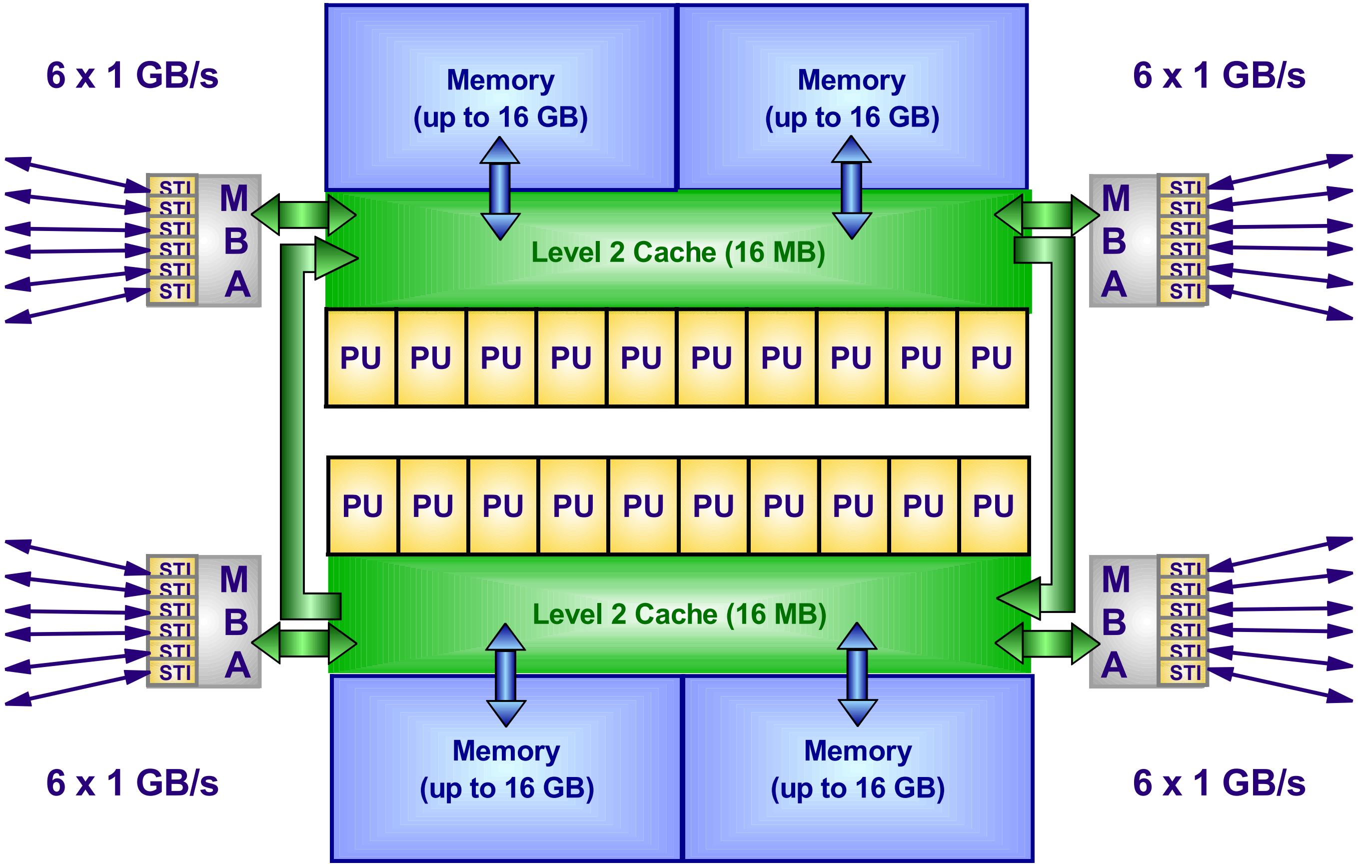


zSeries Hardware

z800/z900

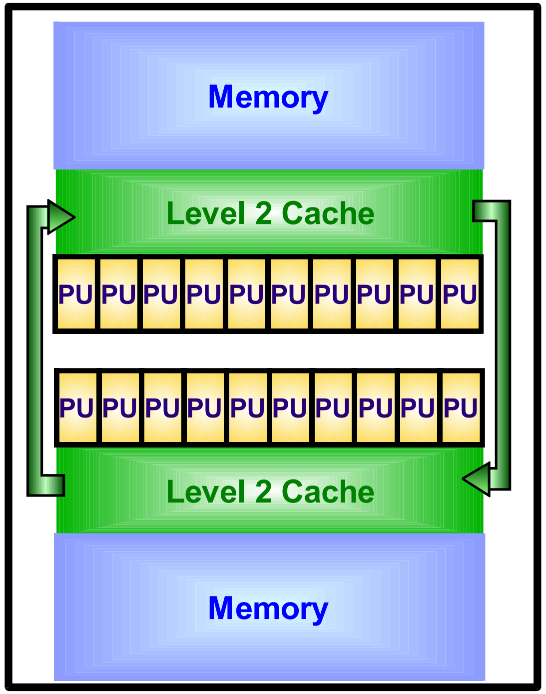
z990

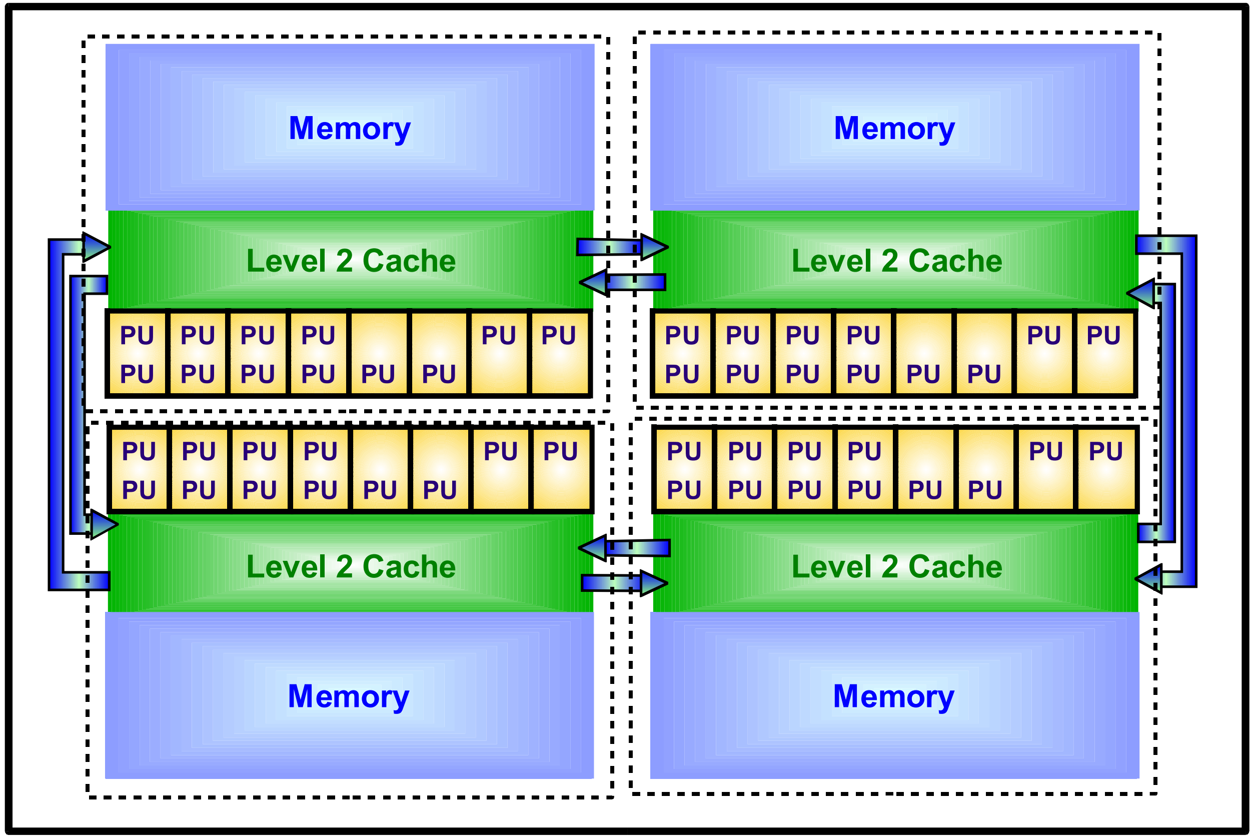
z900 System structure: Optimized for maximum external bandwidth



- 20 PU Chips @ 1.3 / 1.09 ns
- 3 SAP's, 1 spare
- up to 16 CP's
- up to 8 ICF's/IFL's

zSeries 2003: Extended Multi-Node(Book)-Structures:



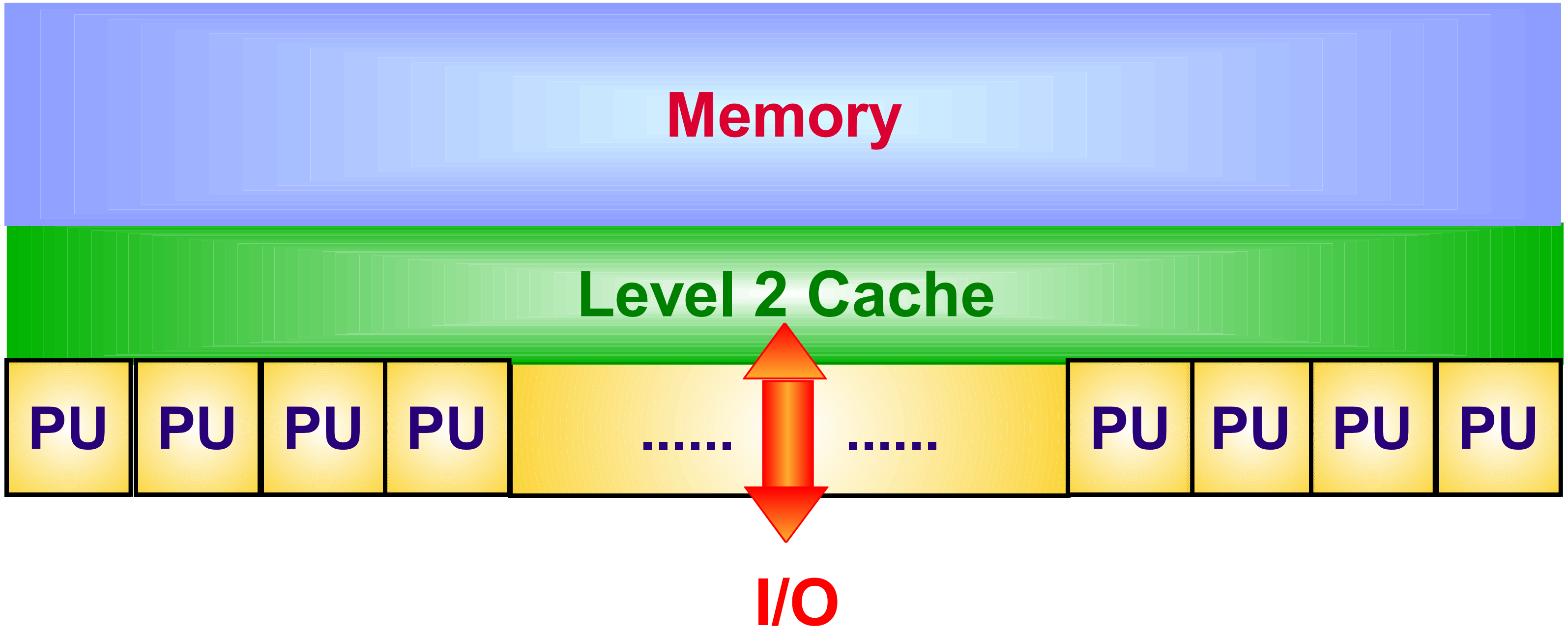


From z900 ...

... to next generation modular zSeries systems with x-fold capacity

- 0.8x nsec CPU-Cycle
- Superscalar Design
- ..50..60.. % more UP-Performance vs 2C1

zSeries 2003: Multi-Book(Node)-Structures (logical view)



- A single pool of physical resources (CPU's, memory, I/O) in modular implementation (n=1/2/3/4 nodes/'books')
- Multiple Channel Subsystems (n x 256 CHPIDs)
- Exploitation through virtual servers: 15, 30, 60 (SOD) LPARs ...100+... (VM)

z990 – 4 New Models

- Machine Type: 2084
- 4 Models:
  - A08 & B16 (GA1 = 6/16/2003) maximum 2 Books
  - C24 & D32 (GA2 = 4Q2003) maximum 4 Books
- Each Processor-Book has:
  - 12 PUs
  - 8 PUs available for characterization as CPs, IFLs, ICFs or additional SAPs
  - 2 PUs standard as SAPs
  - 2 PUs standard as spares
- Memory:
  - Up to 64 GB per Book
  - System minimum of 8 GB
  - 8 GB increments
- I/O:
  - Each Book has up to 12 STIs @ 2 GB/s
  - Up to 96 GB System bandwidth
  - Up to 512 channels/CEC - dependent on Channel types

Our Hardware for Measurements

2064-216 (z900)

1.09ns (917MHz)
2 * 16 MB L2 Cache (shared)
64 GB
FICON
HiperSockets
OSA Express GbE

2105-F20 (Shark)

384 MB NVS
16 GB Cache
128 * 36 GB disks
10.000 RPM
FCP (2 Gbps)
FICON (1 Gbps)

2084-B16 (z990)

2 Books each with 8 PU
64 GB
OSA Express GbE
FICON
Hipersocket
z/VM 4.4.

8687-3RX (8-way X440)

8-way Intel Pentium 3 Xeon
1.6 GHz
8 * 512K L2 Cache (private)
hyperthreading
summit chipset

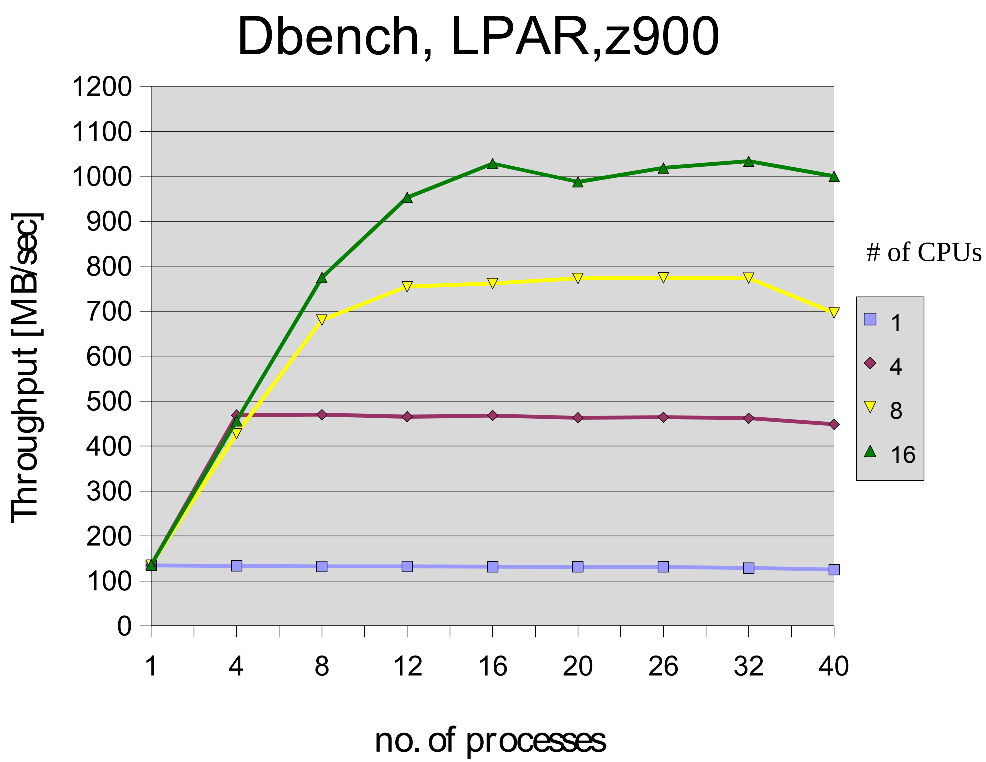
SuSE SLES7 versus SuSE SLES8

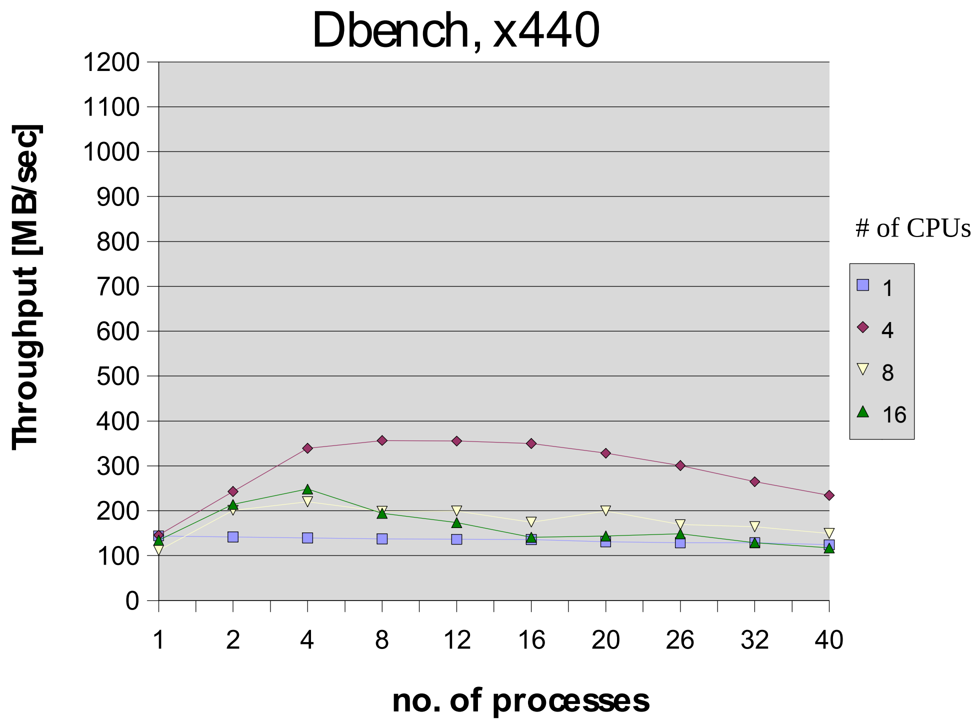
- From Kernel version 2.4.7 / 2.4.17 to version 2.4.19
- From glibc version 2.2.4-31 to version 2.2.5-84
- From gcc version 2.95.3 to version 3.2-31
- Huge number of United Linux patches
- 1.3 MLOC (including x,p,i changes)
- New Linux scheduler
- Async I/O
- SLES8 SP2 available

Scalability – Dbench File I/O

- Emulation of Netbench benchmark, rates windows fileservers
- Mixed file operations workload for each process: create, write, read, append, delete
  - Scaling for Linux with 1, 4, 8, 16 PUs
  - Scaling for 1, 4, 8, 12, 16, 20, 26, 32, 40 clients (processes) simultaneously
- Runs completely in main memory/buffercache in our scenario (2 GB main memory)

Scalability - z900 vs Intel, ext2, 31/32Bit

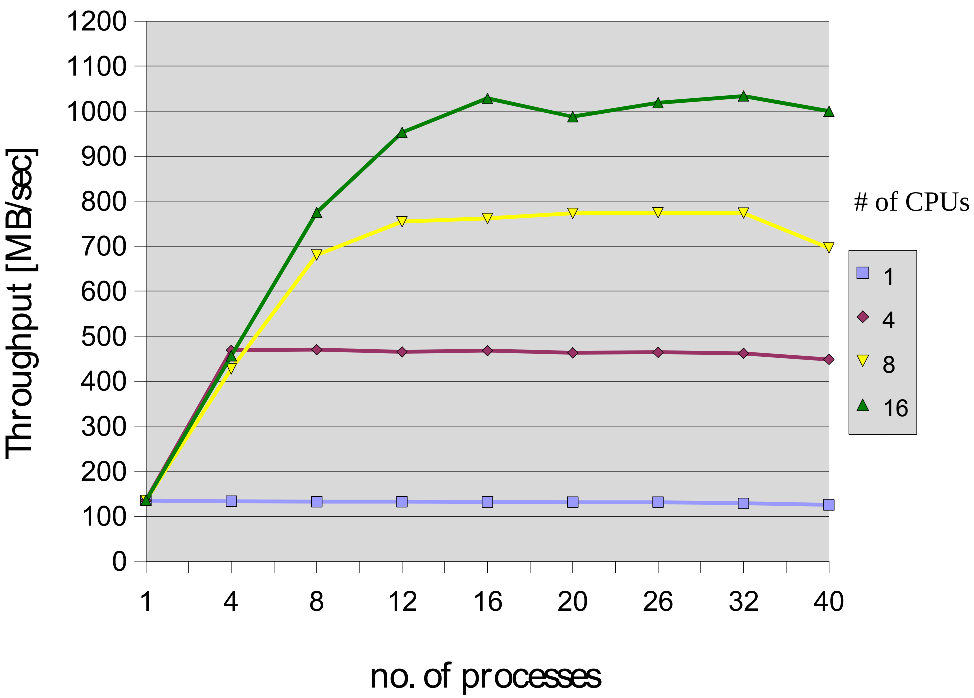




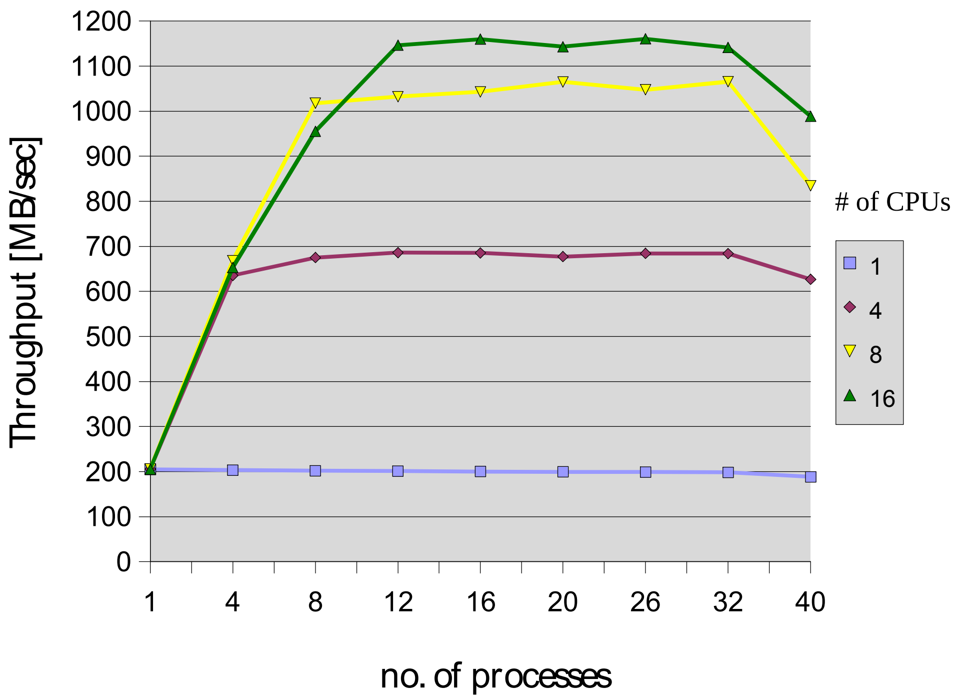
- z900 shows good scaling behavior
- x440 shows best throughput with 4 CPU, strong throughput degradation with more than 4 CPUs

Scalability - z900 vs z990, ext2, 31 Bit

Dbench, LPAR, z900



Dbench, LPAR, z990

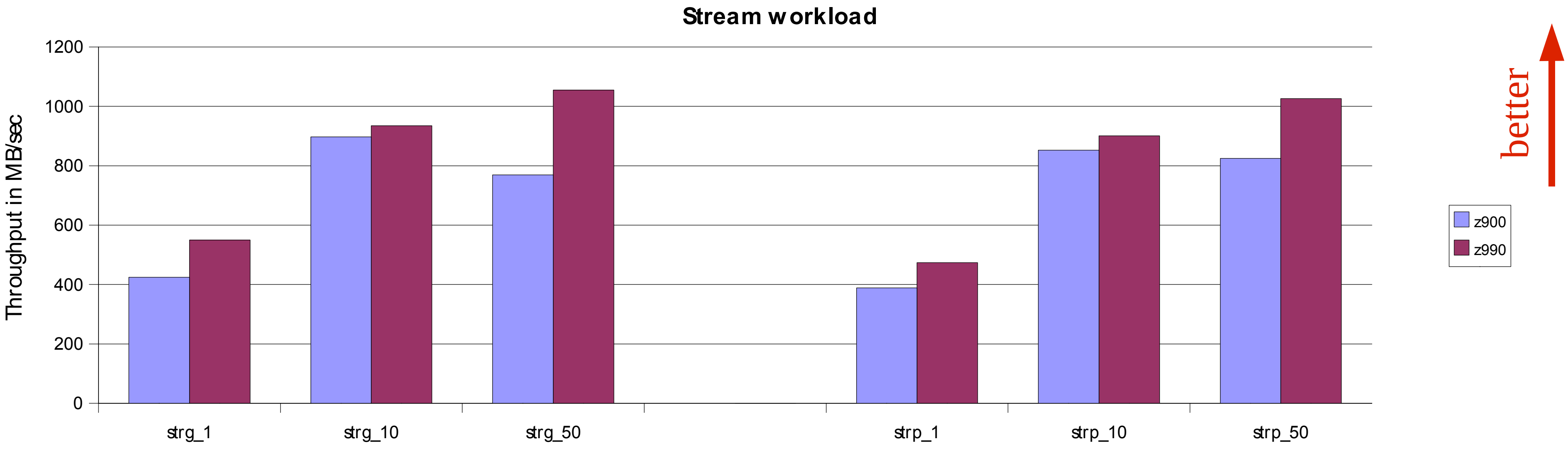


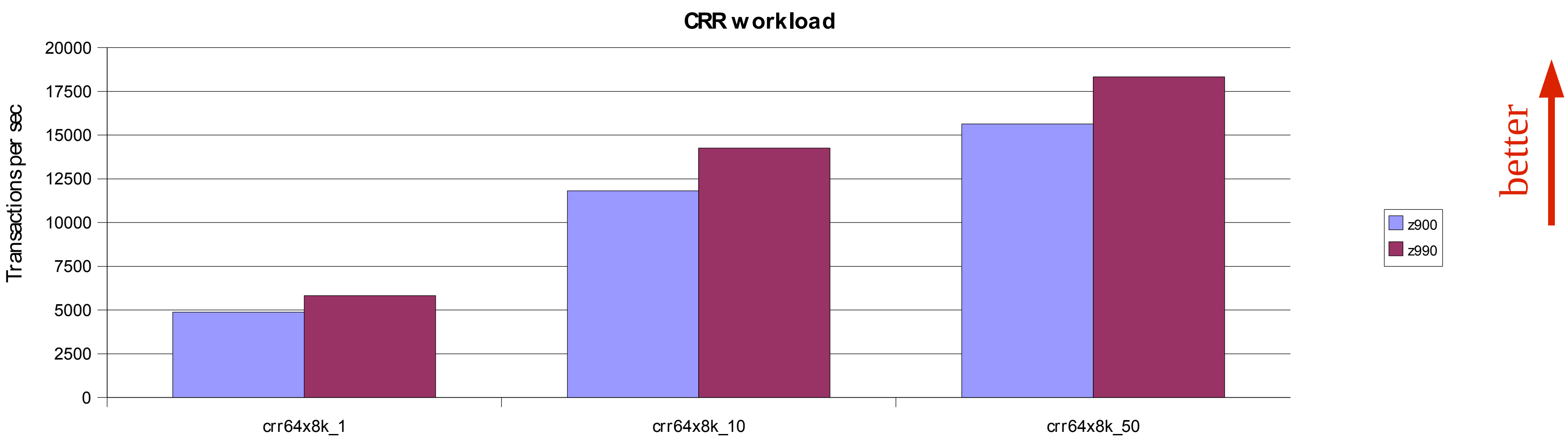
- z990 takes advantage of higher memory bandwidth
- small improvement with 16 PUs because PUs are spread over 2 books

Networking

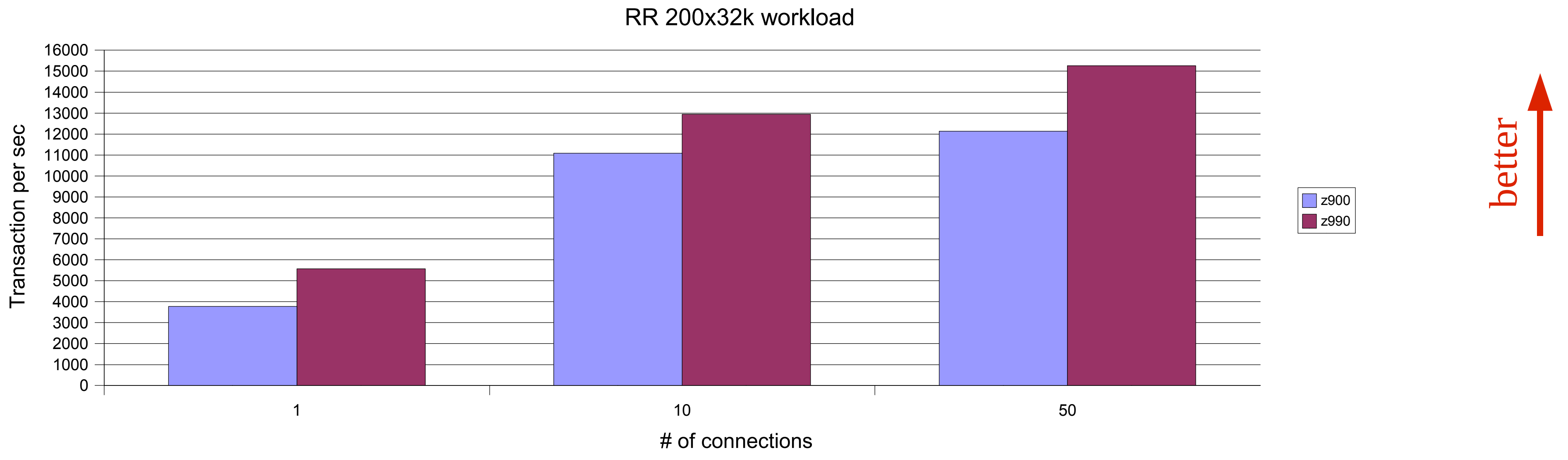
- IBM internal benchmark Netmark 2
- Based on netperf
- Simulates network traffic
- Adjustable parameters
  - runtime
  - packet size
  - number of connections
  - ...
- Huge results file with much statistical information
- Numbers measured on z900 and z990

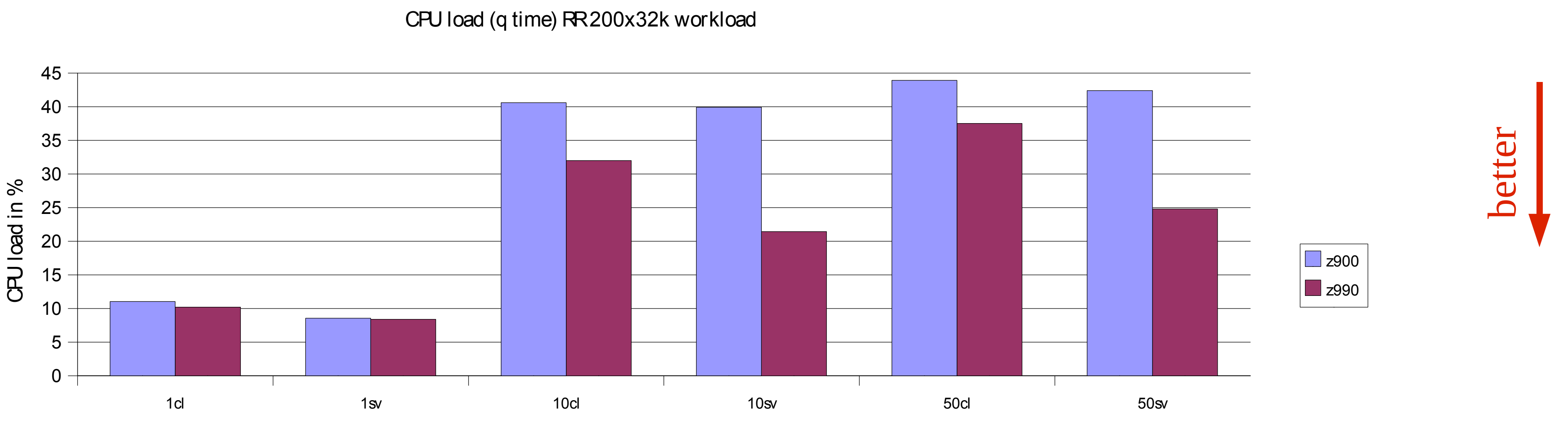
Hipersocket MTU 32k – LPAR





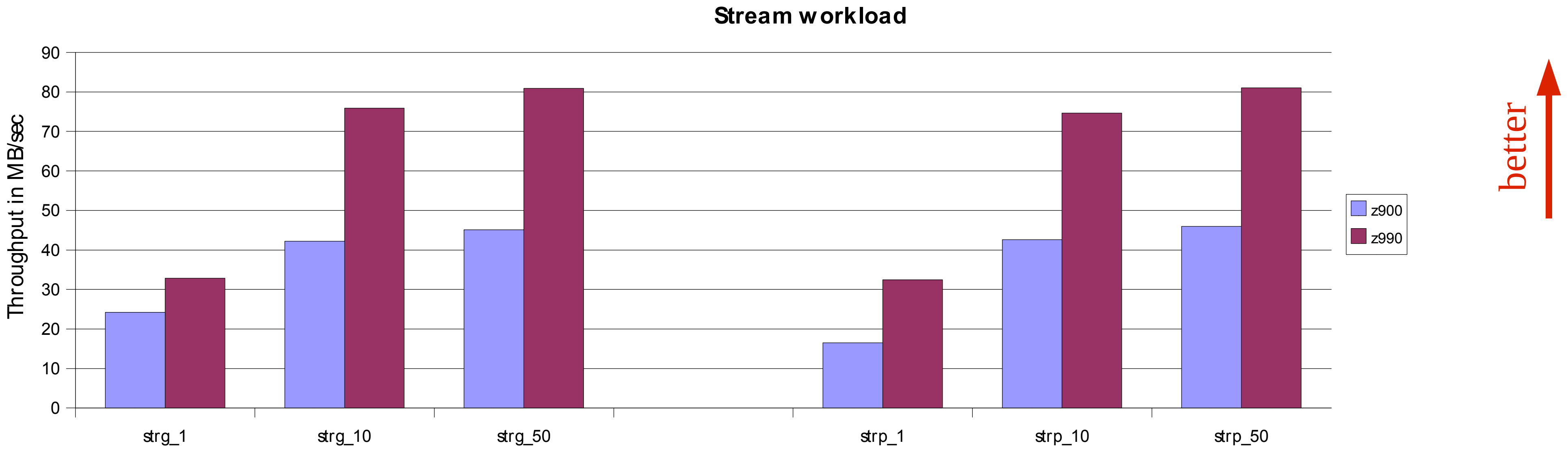
GuestLAN type Hipersocket MTU 32k – z/VM

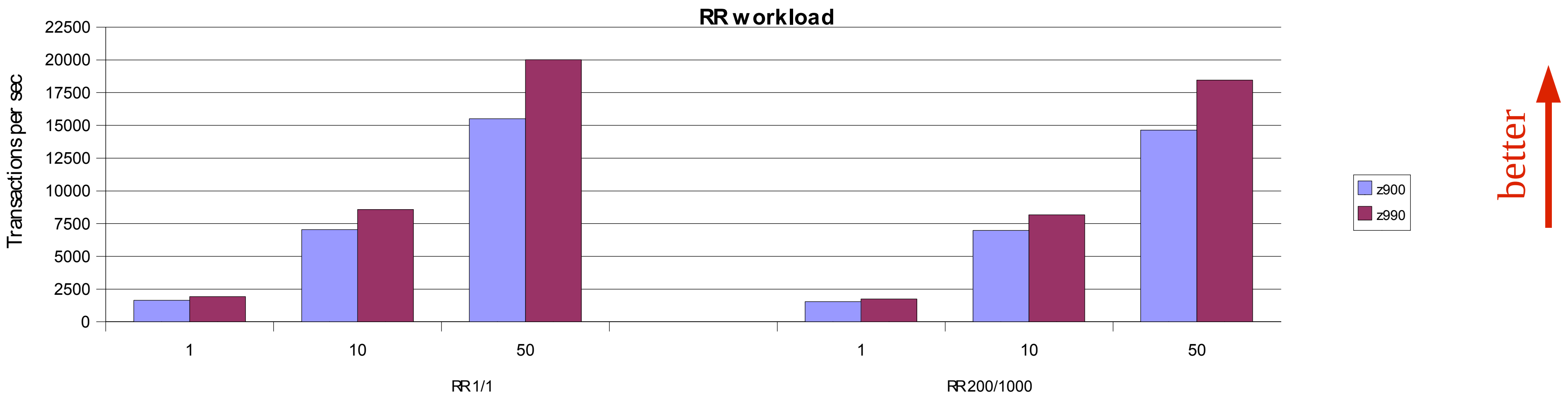




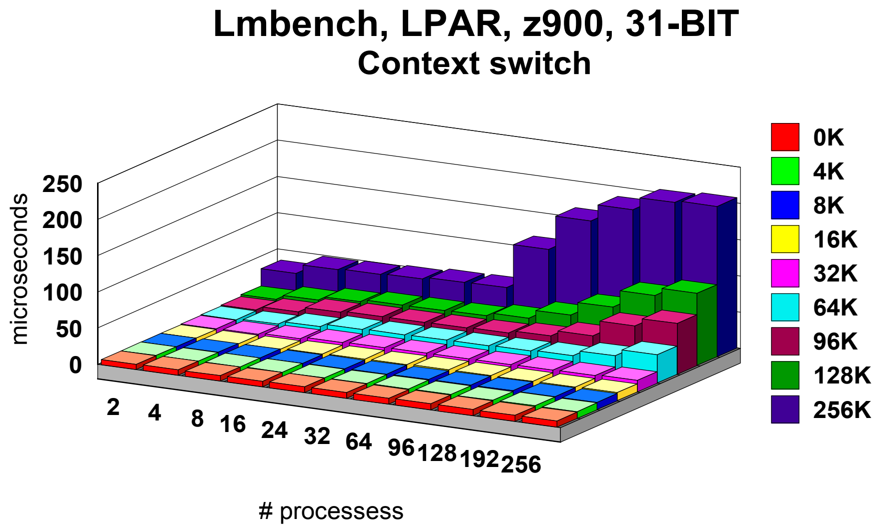
1cl = 1 connection client side (sv=server)

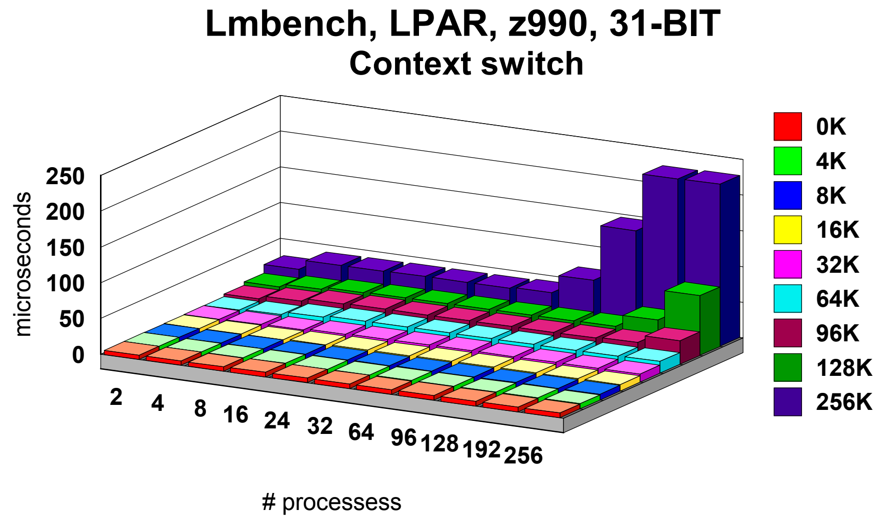
Gigabit Ethernet MTU 1500 – z/VM

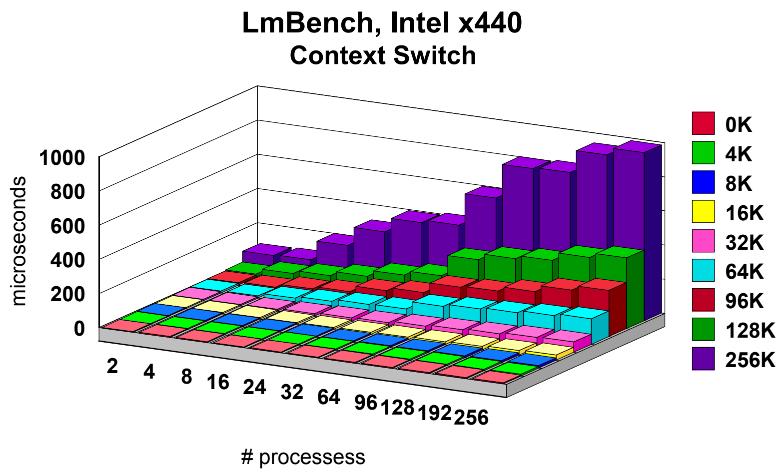




Kernel – Context Switches

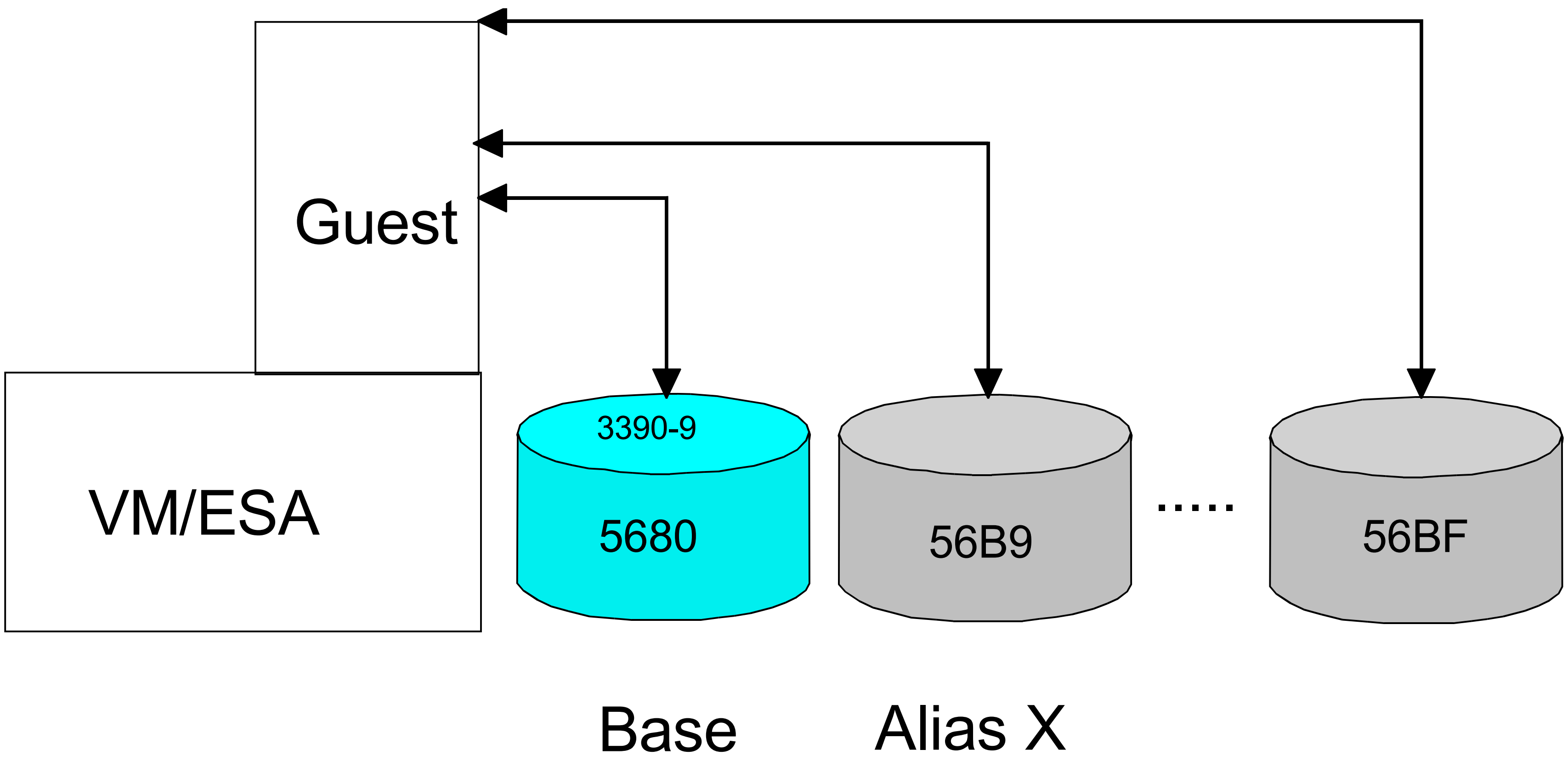






- Context Switches much faster on zSeries because of large shared caches
- Be aware of different Y-axis scale

Parallel Access Volume (PAV)
A Lab experiment



Linux cannot enable PAV on the ESS but can use it under VM

Base and Aliases (PAV Cont.)

- IOCDS changes

```
IODEVICE ADDRESS=(5680,024),UNITADD=00,CUNUMBR=(5680),   *
      STADET=Y,UNIT=3390B
IODEVICE ADDRESS=(5698,040),UNITADD=18,CUNUMBR=(5680),   *
      STADET=Y,UNIT=3390A
```

- ATTACH Base and Aliases to the guest
- QUERY PAV shows base and alias addresses

cat /proc/dasd/devices

```
5794(ECKD) at ( 94:  0) is dasda   : active at blocksize: 4096, 1803060 blocks, 7043 MB
5593(ECKD) at ( 94:  4) is dasdb   : active at blocksize: 4096, 601020 blocks, 2347 MB
5680(ECKD) at ( 94:  8) is dasdc   : active at blocksize: 4096, 1803060 blocks, 7043 MB
56bf(ECKD) at ( 94: 12) is dasdd   : active at blocksize: 4096, 1803060 blocks, 7043 MB
```

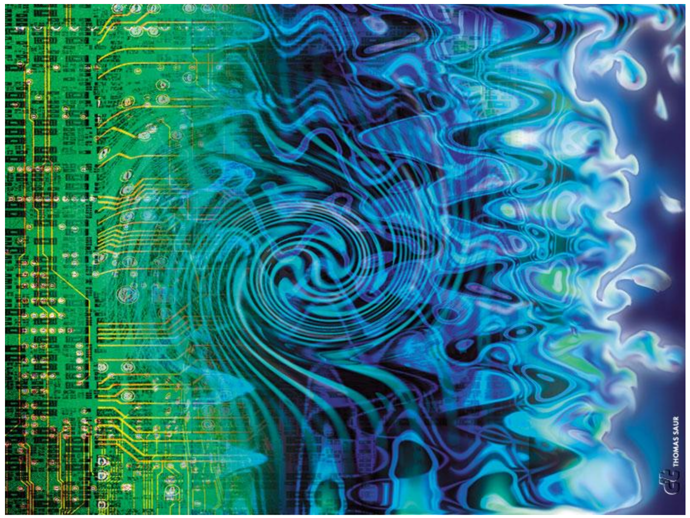
cat /proc/subchannels | egrep "5680|56BF"

```
5680   0030  3390/0C   3990/E9  yes    FC  FC  FF  C6C7C8CA CBC90000
56BF   0031  3390/0C   3990/E9  yes    FC  FC  FF  C6C7C8CA CBC90000
```

This works only with z/VM

LVM commands (PAV Cont.)

- vgscan: create configuration data
- pvcreate /dev/dasdc1
- vgcreate vg_kb /dev/dasdc1
- vgdisplay



vgdisplay

vgdisplay -v vg_kb

```
--- Volume group ---
VG Name               vg_kb
VG Access             read/write
VG Status             available/resizable
VG #                  0
MAX LV                256
Cur LV                0
Open LV               0
MAX LV Size           255.99 GB
Max PV                256
Cur PV                1
Act PV                1
VG Size               6.87 GB
PE Size               4 MB
Total PE              1759
Alloc PE / Size       0 / 0
Free  PE / Size       1759 / 6.87 GB
VG UUID               3nwJYn-SxW1-gKym-OvZs-TYIf-CrHP-inO5Yp

--- No logical volumes defined in "vg_kb" ---
```

More LVM commands

lvcreate --name lv_kb --extents 1759 vg_kb

cat /proc/lvm/global

```
LVM module LVM version 1.0.5(mp-v6)(15/07/2002)
Total:  1 VG  1 PV  1 LV (0 Lvs open)
Global: 32300 bytes malloced   IOP version: 10   3:18:35 active
VG:  vg_kb  [1 PV, 1 LV/0 open]  PE Size: 4096 KB
  Usage [KB/PE]: 7204864 /1759 total  7204864 /1759 used  0 /0 free
  PV:  [AA] dasdc1                 7204864 /1759     7204864 /1759
         0 /0
        +-- dasdd1
    LV:  [AWDL   ] lv_kb                      7204864 /1759     close
```

lvscan

```
lvscan -- ACTIVE            "/dev/vg_kb/lv_kb" [6.87 GB]
lvscan -- 1 logical volumes with 6.87 GB total in 1 volume  group
lvscan -- 1 active logical volumes
```

Enable Paths

pvpath -qa

```
Physical volume /dev/dasdc1 of vg_kb has 2 paths:
       Device  Weight Failed Pending State
 #  0:  94:9        0      0       0 enabled
 #  1:  94:13       0      0       0 disabled
```

The second path can be enabled:

pvpath -p1 -ey /dev/dasdc1

```
vg_kb: setting state of path #1 of PV#1 to enabled
```

pvpath -qa

```
Physical volume /dev/dasdc1 of vg_kb has 2 paths:
       Device  Weight Failed Pending State
 #  0:  94:9        0      0       0 enabled
 #  1:  94:13       0      0       0 enabled
```

Now LVM is ready to use both paths to the volume

Results

iozone sequential write/read 1 disk

Paths	Write (MB/s)	Read (MB/s)
1	14.9	27.0
2	18.7	46.4
3	22.4	65.9
4	23.4	81.4
5	23.2	96.9
6	22.6	106.7
7	21.2	106.7
8	21.1	119.0

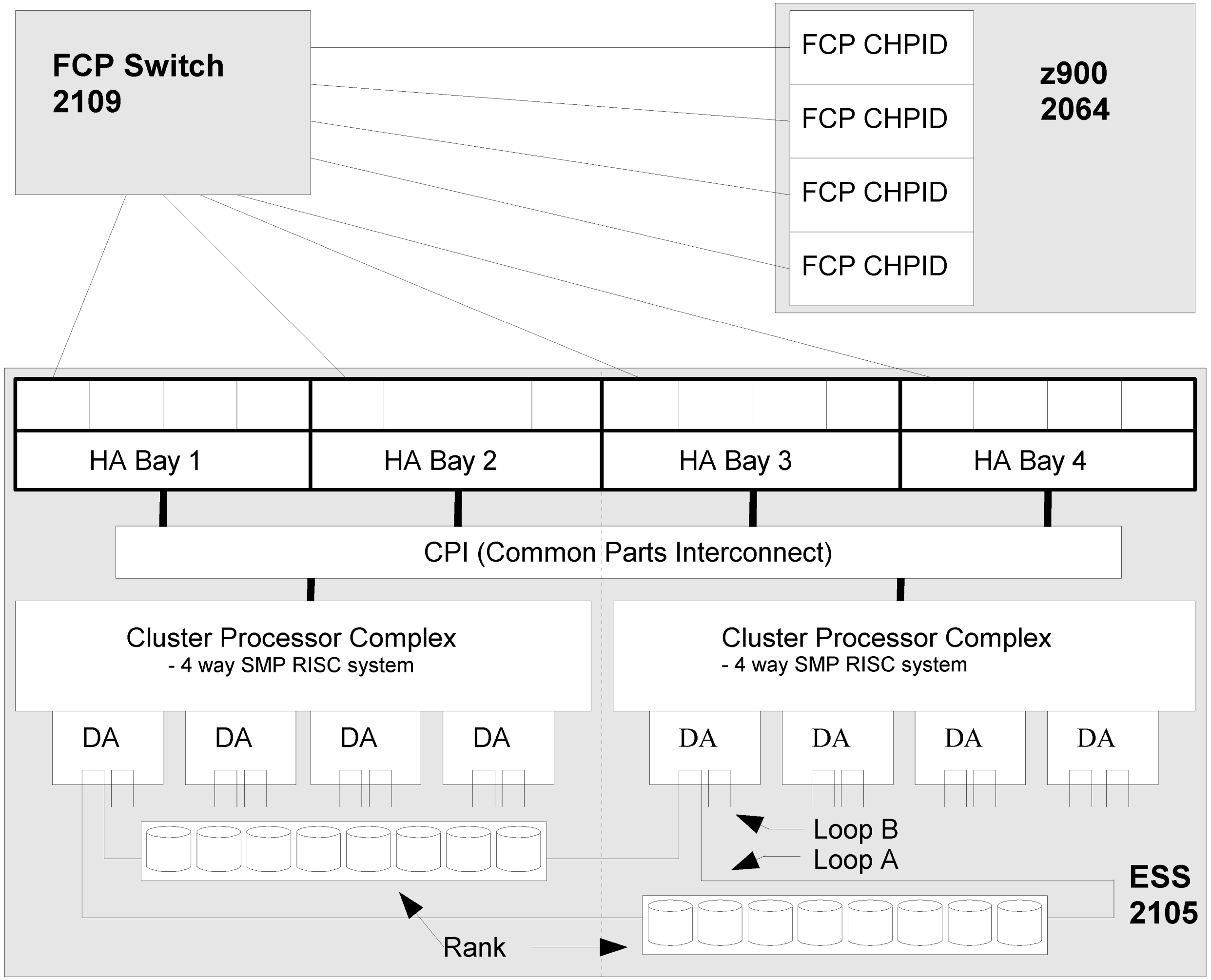
These are preliminary results in a controlled environment.
PAV is not yet officially supported with Linux on zSeries!

ESS – Disk I/O

- Don't treat ESS as a black box, understand its structure
- The default is close to worst case:
- You ask for 16 disks and your SysAdmin gives you
- addresses 5100-510F
- What's wrong with that?

ESS Architecture

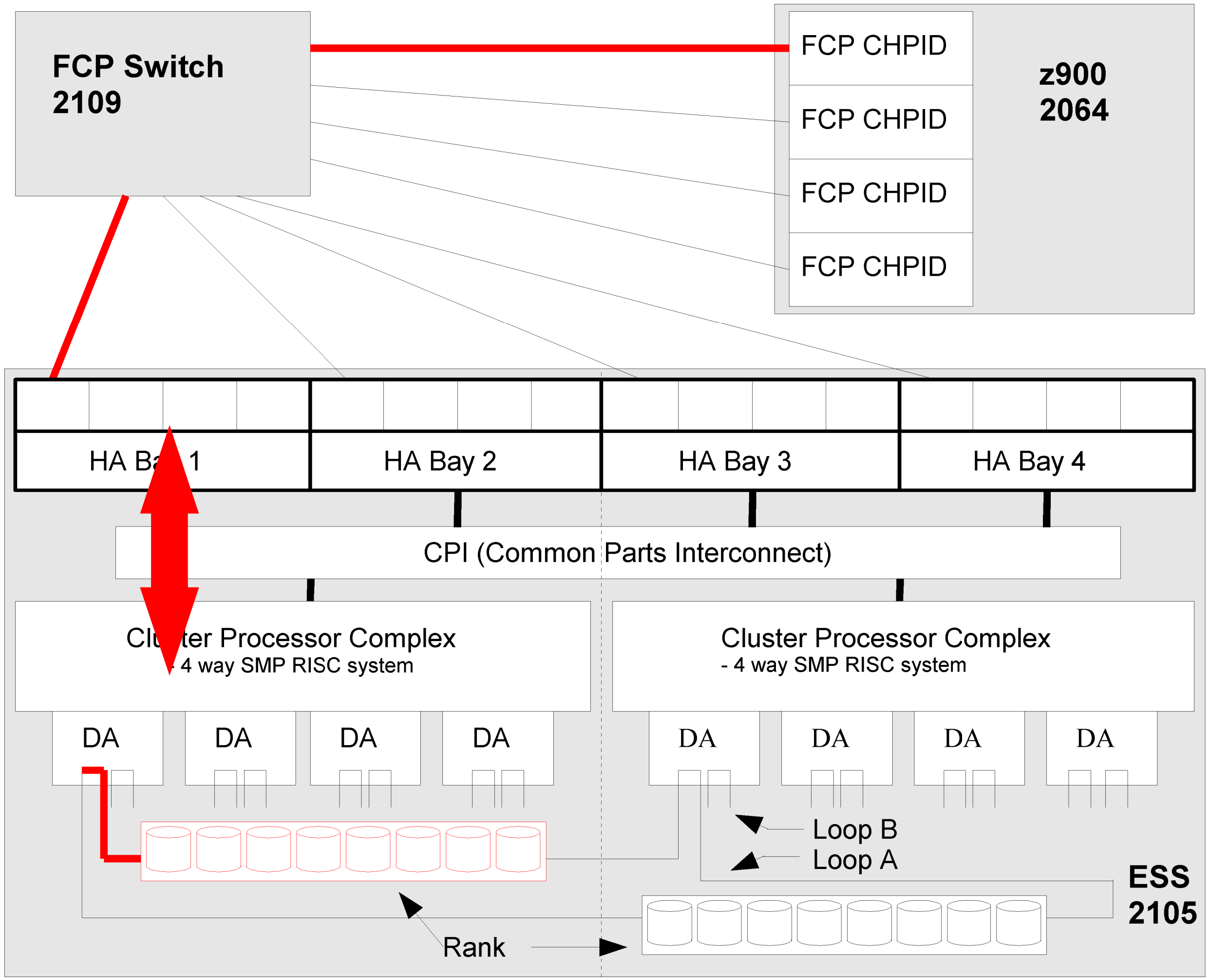
Let's have a deeper look to the elements of the scenario:



- **CHPIDs**
- **Host Adapter (HA) supporting FCP (FCP port)**
  -16 Host Adapters, organized in 4 bays, 4 ports each
- **Device Adapter Pairs (DA)**
  - each one supports two loops
- **Disks are organized in ranks**
  - each rank (8 physical disks) implements one RAID 5 array (with logical disks)

ESS Architecture

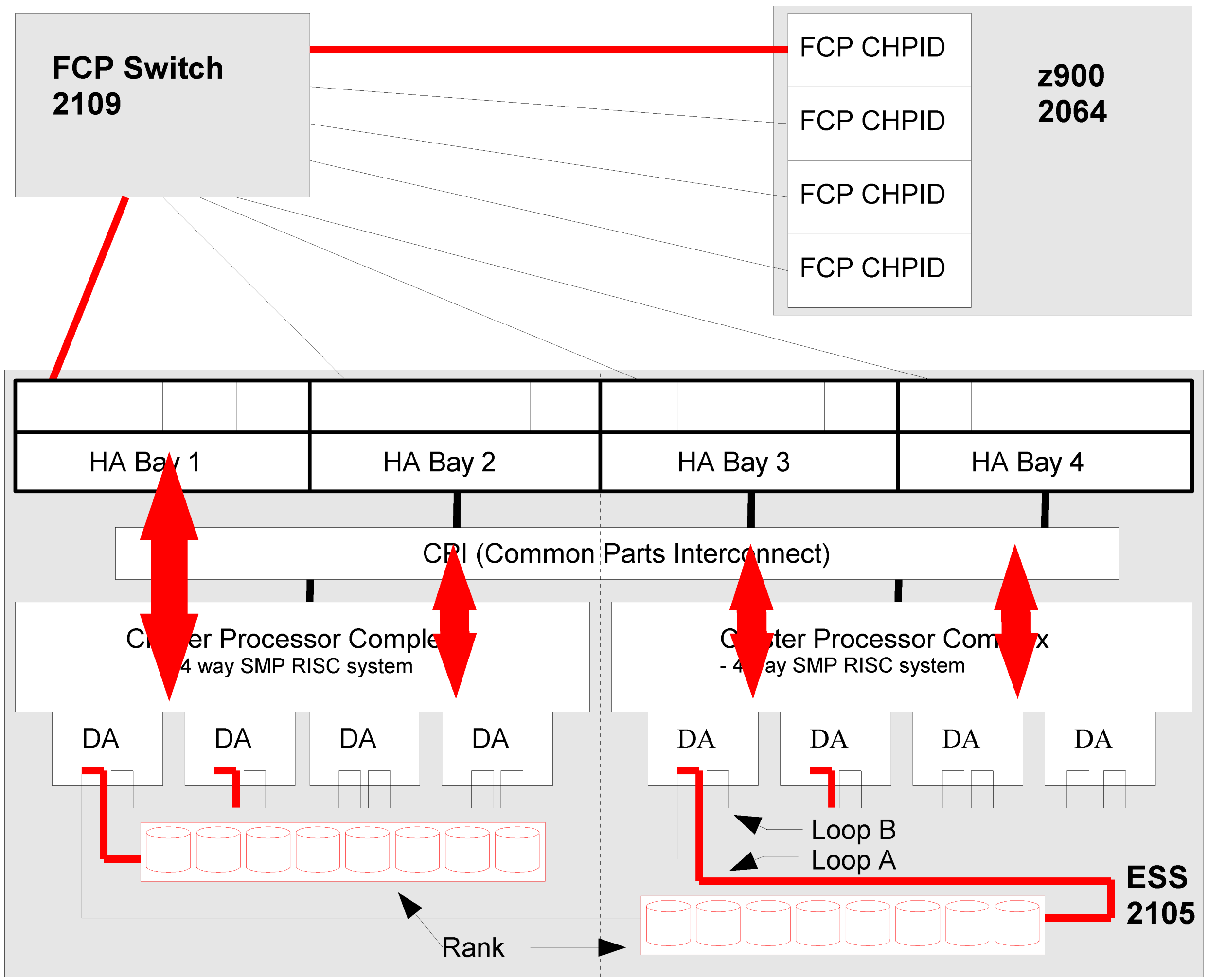
Scenarios: single disk, single rank



- **CHPIDs**
- **Host Adapter (HA) supporting FCP (FCP port)**
  -16 Host Adapters, organized in 4 bays, 4 ports each
- **Device Adapter Pairs (DA)**
  - each one supports two loops
- **Disks are organized in ranks**
  - each rank (8 physical disks) implements one RAID 5 array (with logical disks)

ESS Architecture

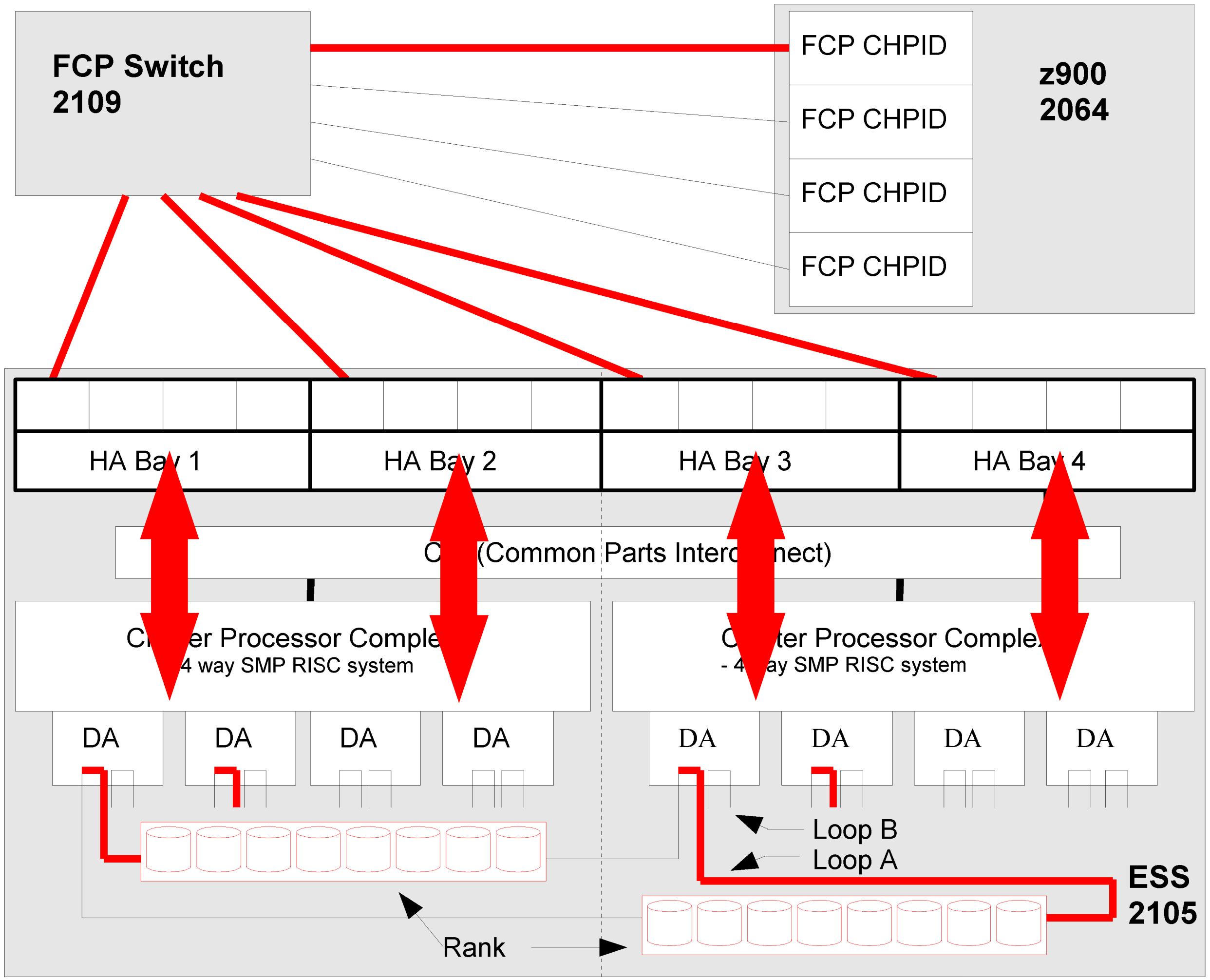
Scenario: single host adapter



- **CHPIDs**
- **Host Adapter (HA) supporting FCP (FCP port)**
  - 16 Host Adapters, organized in 4 bays, 4 ports each
- **Device Adapter Pairs (DA)**
  - each one supports two loops
- **Disks are organized in ranks**
  - each rank (8 physical disks) implements one RAID 5 array (with logical disks)

ESS Architecture

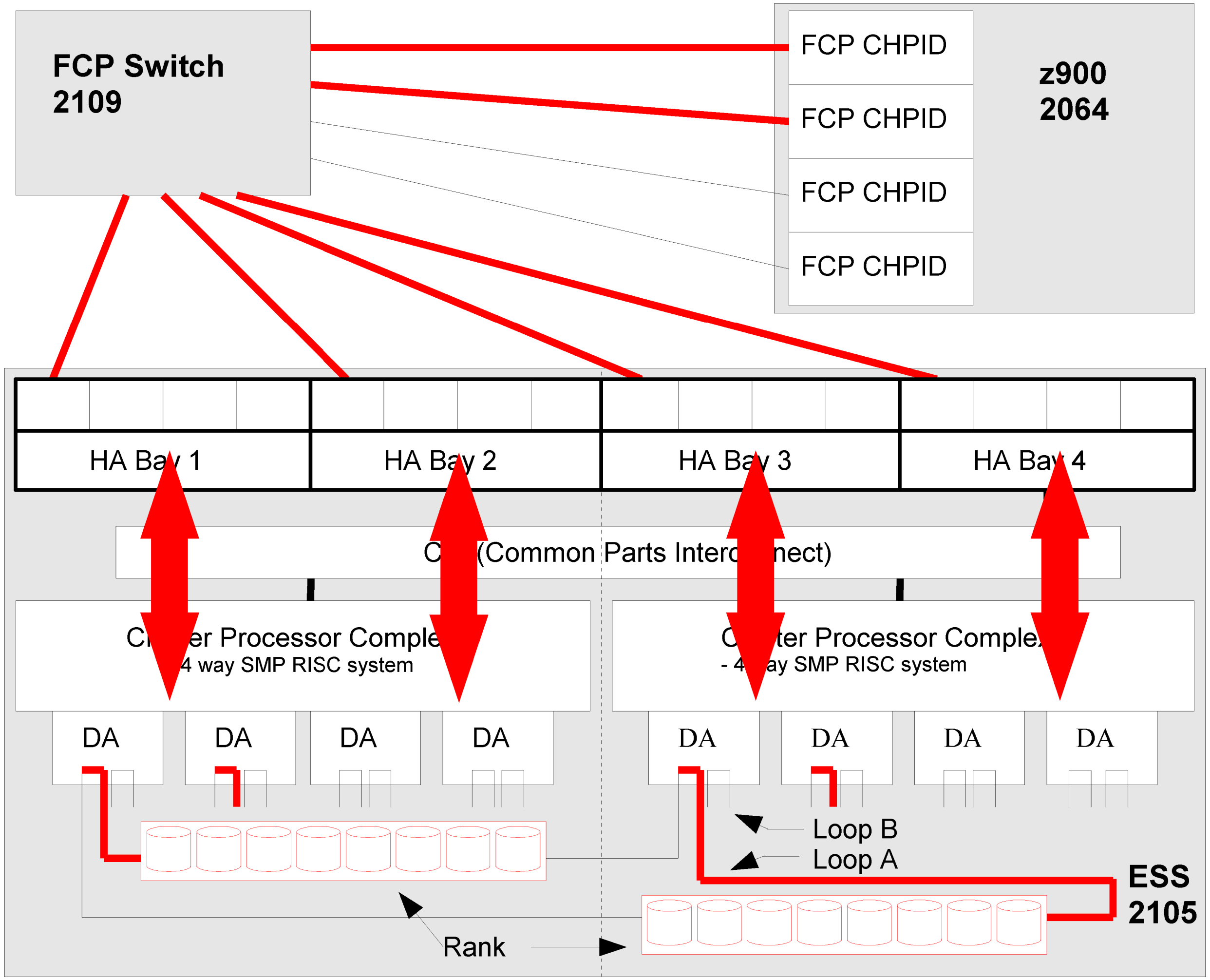
Scenario: single CHPID



- **CHPIDs**
- **Host Adapter (HA) supporting FCP (FCP port)**
  -16 Host Adapters, organized in 4 bays, 4 ports each
- **Device Adapter Pairs (DA)**
  - each one supports two loops
- **Disks are organized in ranks**
  - each rank (8 physical disks) implements one RAID 5 array (with logical disks)

ESS Architecture

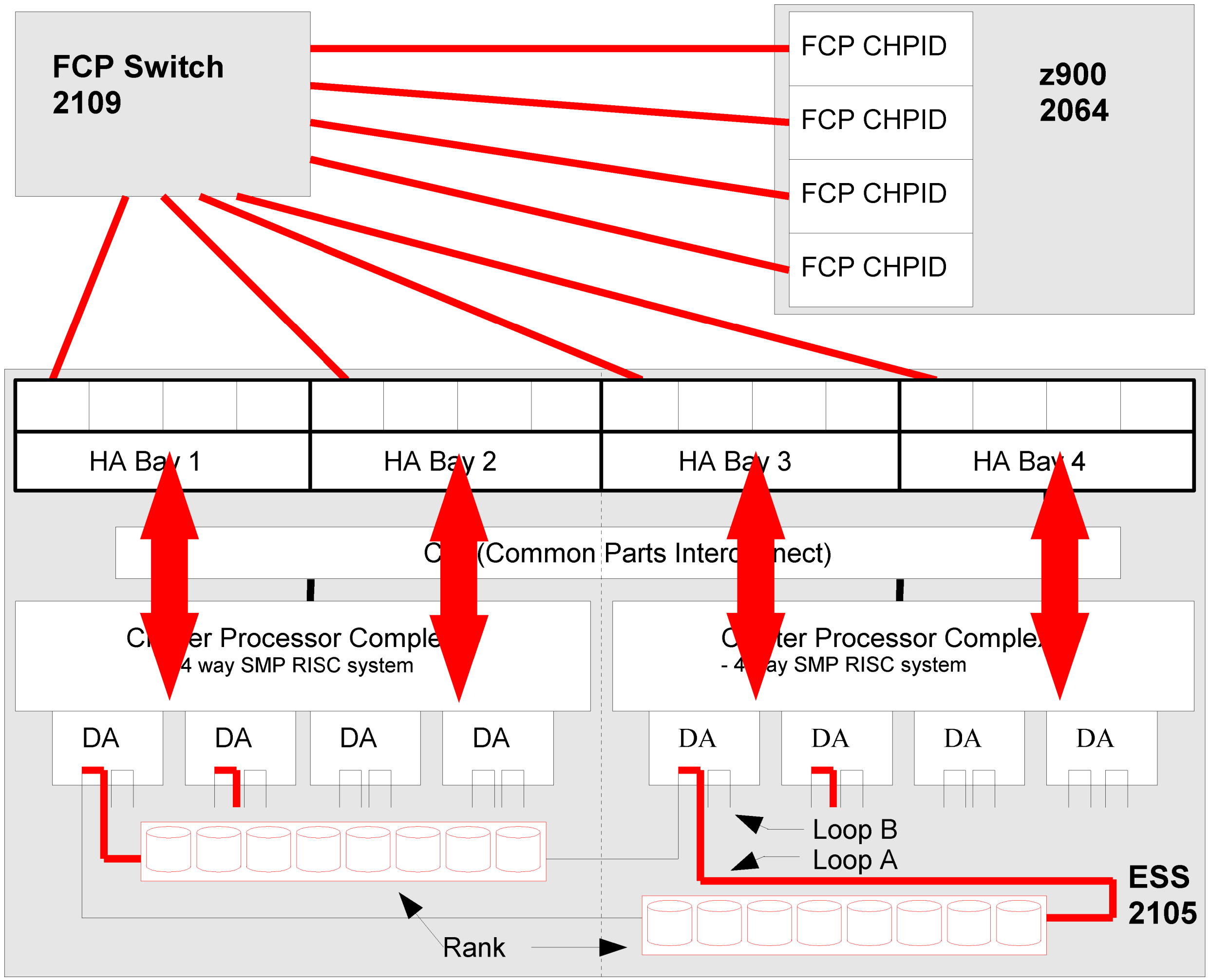
Scenario: two CHPIDs



- **CHPIDs**
- **Host Adapter (HA) supporting FCP (FCP port)**
  -16 Host Adapters, organized in 4 bays, 4 ports each
- **Device Adapter Pairs (DA)**
  - each one supports two loops
- **Disks are organized in ranks**
  - each rank (8 physical disks) implements one RAID 5 array (with logical disks)

ESS Architecture

Scenario: four CHPIDs (4C4H4R ESS 2105)



- **CHPIDs**
- **Host Adapter (HA) supporting FCP (FCP port)**
  -16 Host Adapters, organized in 4 bays, 4 ports each
- **Device Adapter Pairs (DA)**
  - each one supports two loops
- **Disks are organized in ranks**
  - each rank (8 physical disks) implements one RAID 5 array (with logical disks)

FCP Measurement

- Summary of the Scenarios:

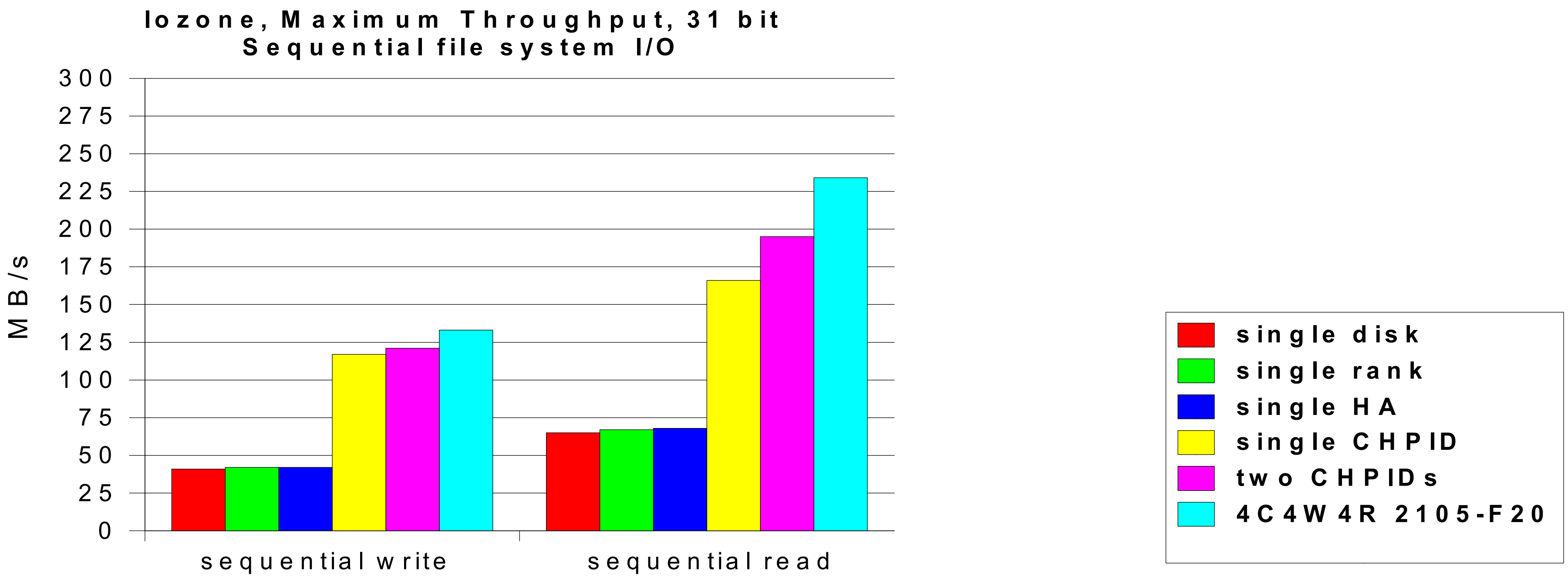
Scenario	used resources				limiting resource
	CHPIDs	HA	Ranks	Disks	
single Disk	1	1	1	1	1 host adapter
single Rank	1	1	1	8	1 host adapter
single Host Adapter	1	1	4	8	1 host adapter
single CHPID	1	4	4	16	1 CHPID
two CHPIDs	2	4	4	16	2 CHPIDs
maximum available = 4C4H4R ESS 2105	4	4	4	16	4 host adapters

- Benchmark used for measuring: **Iozone** (http://www.iozone.org)
  - multi process sequential file system I/O
  - each process writes and reads a 350 MB file on a separate disk
  - System: LPAR, 4 CPUs, 128 MB main memory, Linux 2.4.17 with hz timer off
- scaling was: 1, 2, 4, 8, 16 processes
  the maximum throughput values were taken as result

Hardware Setup

- 2064-216, 917 Mhz, 256MB LPAR
- 4 FICON Express channels used for FCP (IOCDS: type FCP)
- 6 FICON Express Channels used for FICON (IOCDS: type FC)
- 2109-F16 FCP switch
- ESS 2105-F20:
  - 16GB cache, 4 FCP host adapters, 6 FICON host adapters
  - 4 device adapter pairs
  - only A-loops contain disks (36.4 GB, 10,000 RPM):
    - - 4 ranks for FB (fixed block) disks used for FCP
    - - 8 ranks ECKD disks used for FICON measurements

Results – Maximum Throughput



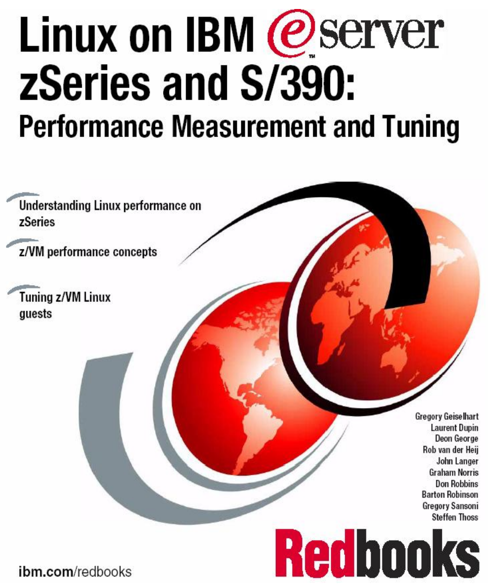
- 1 HA limits to 40MB/s write and 65 MB/s read, regardless of the number of ranks
- 4 HA are limiting to 125 MB/s write and 240 MB/s read,but 4 CHPIDs are required to make use of it
- 31 bit and 64 bit difference is small
- it is expected that the values further increase using more ranks, HA, CHPIDs

General Rules

- this makes it **slow**:
  - when all disks are from one rank and accessed via the same path
- this makes it **fast**:
  - use many host adapters
  - spread the host adapters used across all host adapter bays
  - use as much CHPIDs as possible and
    access each disk through all CHPIDs, if possible (FICON, LVM1-mp)
  - don't use more than two HAs per CHPID
  - spread the disks used over all ranks equally
  - avoid using the same resources (CHPID, host adapter, rank) as long as possible
- this applies to FCP and FICON

Visit us !

- Linux for zSeries Performance Website:
  - http://www10.software.ibm.com/developerworks/opensource/linux390/whatsnew.shtml
- Linux-VM Performance Website:
  - http://www.vm.ibm.com/perf/tips/linuxper.html
- Performance Redbook:
  - SG24-6926-00



Questions