V95

z/VM Guest Performance

Brian K. Wade, Ph.D. - bkw@us.ibm.com

IBM
SYSTEM z9 AND zSERIES EXPO
October 9 - 13, 2006

Orlando, FL

RETURN TO INDEX

Legal Stuff

The information contained in this document has not been submitted to any formal IBM test and is distributed on an "as is" basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

In this document, any references made to an IBM licensed program are not intended to state or imply that only IBM's licensed program may be used; any functionally equivalent program may be used instead.

Any performance data contained in this document was determined in a controlled environment and, therefore, the results which may be obtained in other operating environments may vary significantly.

Users of this document should verify the applicable data for their specific environments.

It is possible that this material may contain references to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country or not yet announced by IBM. Such references or information should not be construed to mean that IBM intends to announce such IBM products, programming, or services.

Should the speaker start getting too silly, IBM will deny any knowledge of his association with the corporation.

The following are **Trademarks** of the IBM Corporation:

VM/ESA, e-business logo*, HiperSockets, IBM*, IBM logo*,
IBM eServer, RAMAC*, TotalStorage, z/OS, z/VM, zSeries

LINUX is a registered trademark of Linus Torvalds

Penguin (Tux) compliments of Larry Ewing

Overview

- General management of resources
- Processor
- I/O
- Storage and paging
- Linux® guidelines
- Networking
- The cost of VM

What Do You Mean by "Performance"?

- **ETR** (External Throughput Rate): work per wall clock second
- **ITR** (Internal Throughput Rate): work per CPU second
- **CPU utilization:** how busy processor is; tied to ITR
- **Response time**: how long jobs take; reciprocal of ETR
- **Consistency:** is today's behavior like yesterday's?
- How many phone calls you get

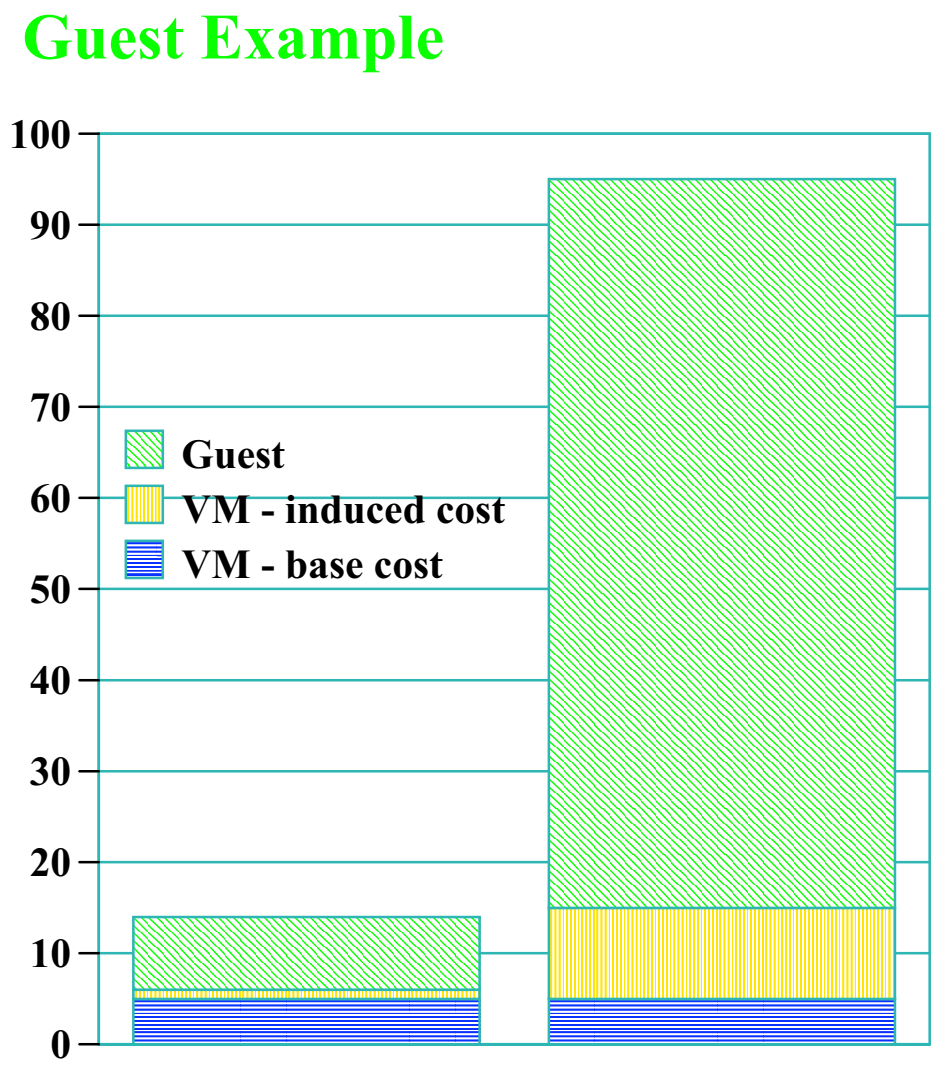
Processor

Processor Resources

- Configuration
  - Virtual 1- to 64-way, defined in user directory or via CP command
  - A real processor can be dedicated to a virtual machine
- Control and Limits
  - "Share" setting
  - Absolute or relative
  - Target minimum and maximum values
  - Maximum values (limit shares) either hard or soft
  - Virtual machine share is divided among its virtual processors
- Rules of thumb
  - For each guest, $N_v <= N_l$
  - Define only as many virtual processors as the workload needs
    - Share dilution; Diag x'44' overhead
  - Do not mix shared and dedicated processors

Processor Usage by VM

- Base costs and background work
  - Scheduling
  - Dispatching
  - Accounting
  - Monitor
- Costs proportional to guest requests or requirements of VM
  - Paging
  - Virtualizing I/O



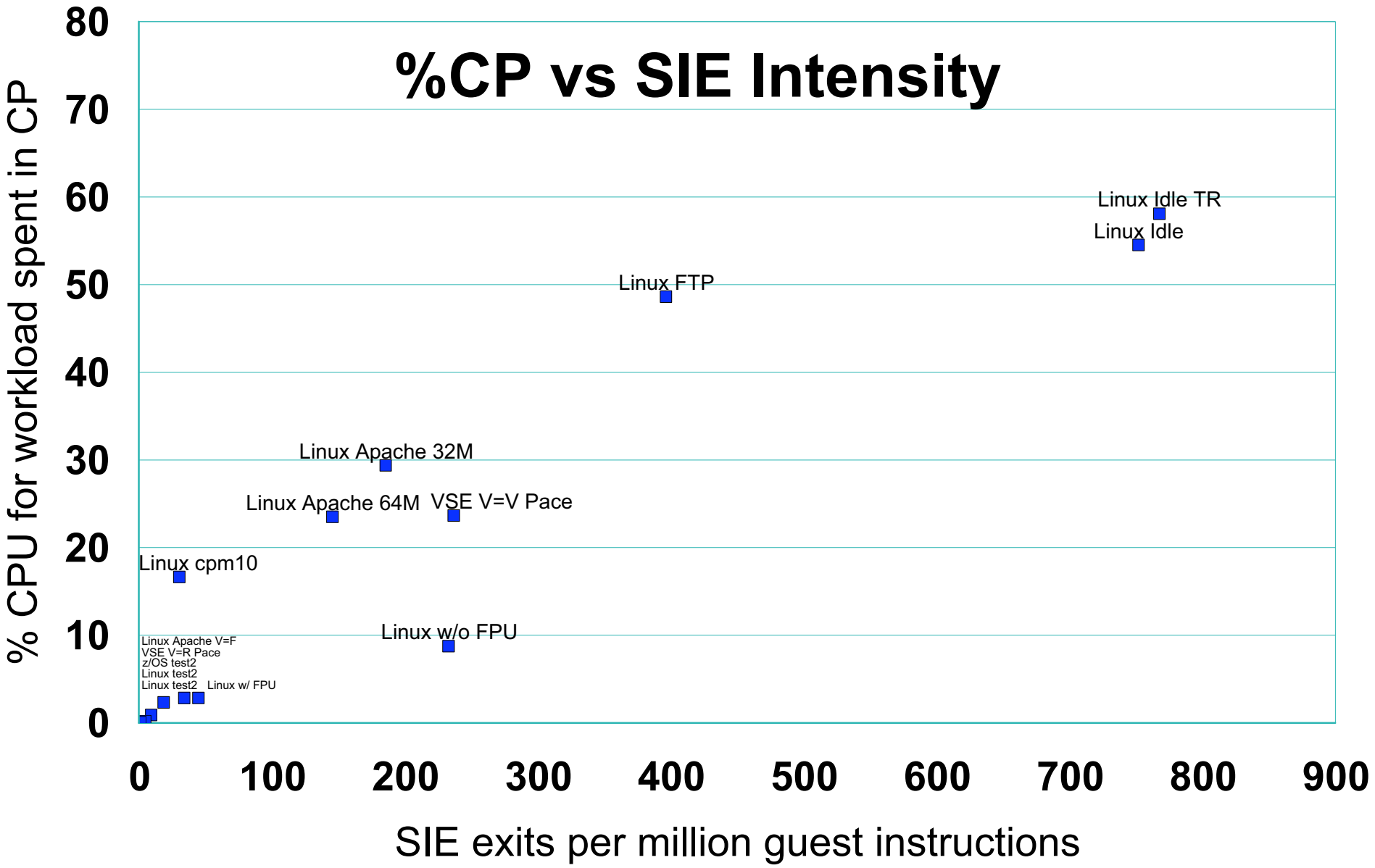
Processor: SIE Exits

- SIE = Start Interpretive Execution
- Used by z/VM™ to run a guest
- Exits from SIE indicate work for VM
  - I/O processing
  - Page fault resolution
  - Instruction simulation
  - Minor time slice expires
  - Loaded wait state
- Each reason for exiting SIE has a different cost (CPU time spent in CP)
- Rate of SIE executions available from most performance monitor products (for example, Performance Toolkit FCX239 report)

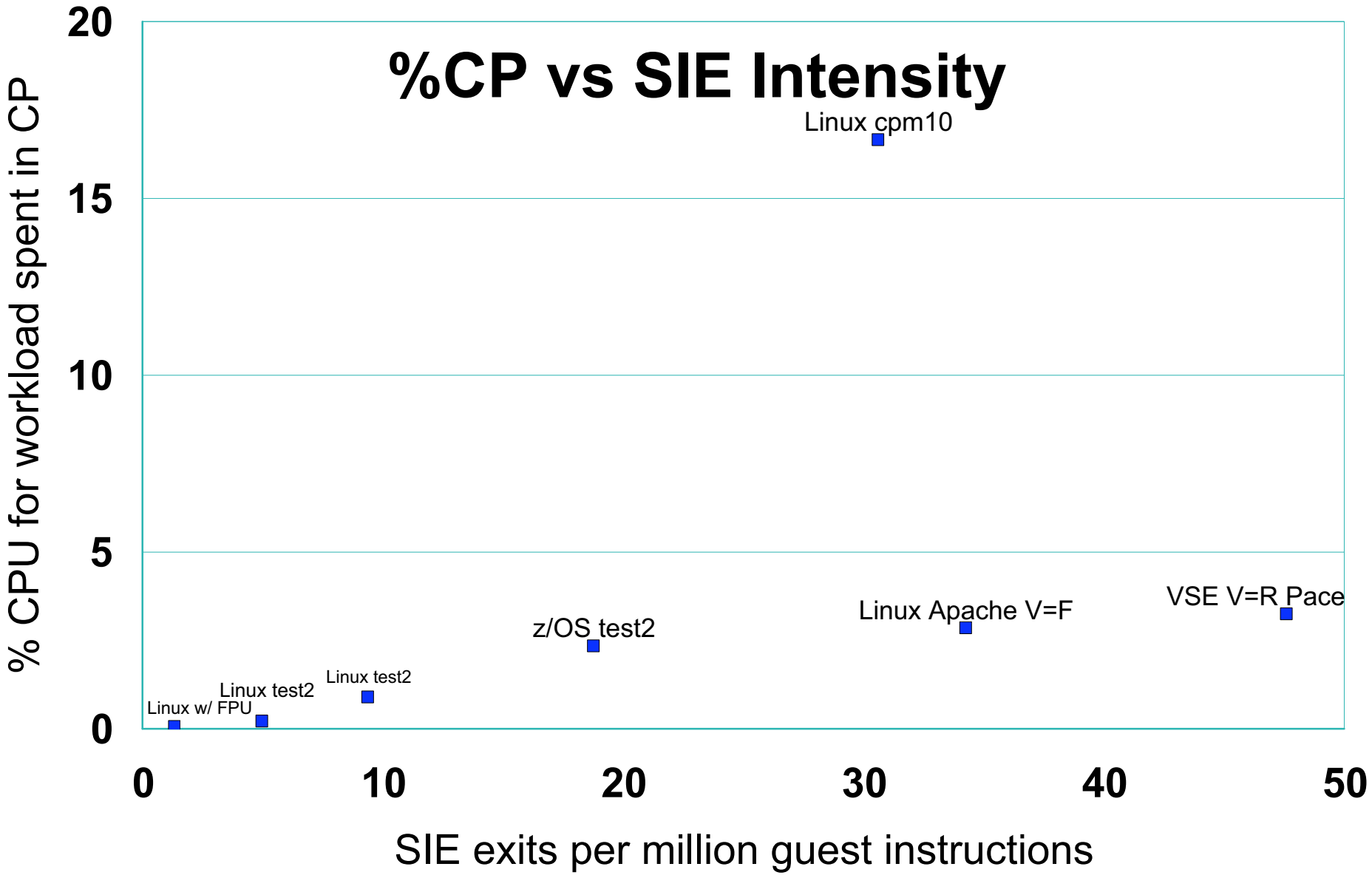
Avoiding Exits from SIE

- Guest data-in-memory techniques avoid guest I/O. Examples:
  - Linux XIP file system
  - CMS programs in segments
  - Shared File System directories in data spaces
- QDIO assists - we've done lots of work here
  - QDIO Assist Part 1 (aka Adapter Interrupt Passthrough (AIP))
    - SIGA assist -- Diag X'98' guest remains in SIE when doing QDIO operations
    - AI assist -- running guest remains in SIE for QDIO interrupt delivery
  - QDIO part 2 (aka QEBSM)
    - Guest SIGA does not require a SIE exit
  - IBM recognizes the pressure for more of these kinds of assists
- Avoid paging:
  - Reserved pages for important guests
  - Sufficient storage for the workload
- Minor time slice: SET SRM DSPSLICE (pros and cons)
- Dedicated processor gets 500 msec minor slice and wait state assist
- Get SIE rate from Performance Toolkit (e.g., FCX239 report)

VM Overhead Cloud Chart



VM Overhead Cloud Chart



Processor: Virtual MP

- Define additional processors dynamically
  - Directory include MACHINE ESA 2
  - CP DEFINE CPU vcpu_addr
- Or put everything in the directory
  - CPU 00 NODEDICATE
  - CPU 01 NODEDICATE
- Detaching a virtual processor resets virtual machine
- Usually, not more virtual processors than real ones
- Do not define virtual processors unnecessarily
  - Dilutes share
  - Produces excessive Diag x'44' overhead

Processor: Virtual MP

- CP commands of interest
  - QUERY VIRTUAL CPUS
  - CPU vcpu_addr cmd_line
  - DEDICATE and UNDEDICATE
- Share setting is for virtual machine, divided among all virtual processors
- Mixing dedicated and shared processors is not recommended
- Dedicated processor appears 100% busy on various VM performance reports

I/O

I/O Resources

- Configuration
  - Dedicated devices (tape drives, DASD, network devices)
  - Subdivided devices (minidisks)
  - Virtualized devices (minidisks, crypto)
  - Simulated devices (guest LAN, virtual CTCs, VDISKs)
  - Define or attach dynamically
- Control and Limits
  - Indirect control through "share" setting
  - Real devices can be throttled at device level
  - Priority can be set for virtual machine
    - CP uses to affect queue placement for DASD devices
    - HW uses to affect priority in channel usage
  - Minidisk Cache fair share limits can be turned off for virtual machine

I/O Considerations

- Dedicated I/O is not eligible for Minidisk Cache (MDC)
- MDC read performance is as good as VDISK performance
- Both VDISKs and MDC require sufficient storage
- Watch for excessive below-2-GB page movement (z/VM 5.2 fixes this!)
  - ">2GB>" column in Performance Toolkit's UPAGE report
  - >1 engine's worth of CP time
  - High %IO wait values in Performance Toolkit's user states report
  - High <2G page residency counts
- SCSI vs. ECKD
  - SCSI: higher I/O rates, higher CPU cost per I/O
  - ECKD: a little slower, but a lot cheaper in CPU/tx

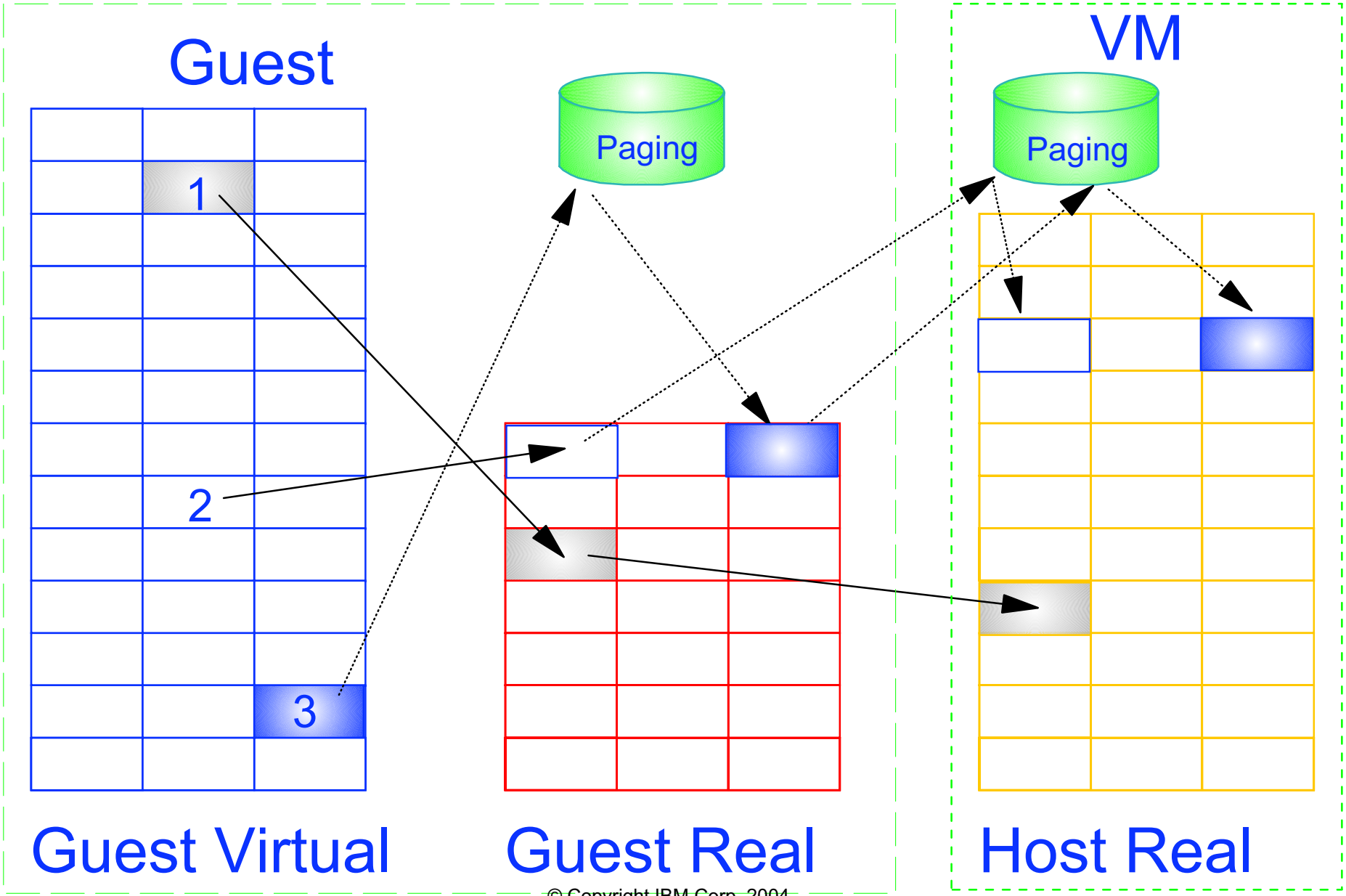
Memory

Storage Resources

- Configuration
  - Defined in user directory or via CP command
  - Can define storage with gaps (useful for testing)
  - Can attach expanded storage to virtual machine

- Control and Limits
  - Scheduler helps control overcommitting storage and paging resources
  - Virtual machines that do not "fit" criteria are placed in eligible list
  - Virtual machine can be made exempt from eligible list via QUICKDSP
  - Can "reserve" or "lock" pages for important guests
    - Reserve a number of pages to influence storage management page steal algorithms (recommended approach)
    - Lock specific pages (less flexible, requires clairvoyance)

Paging Considerations



Paging Considerations

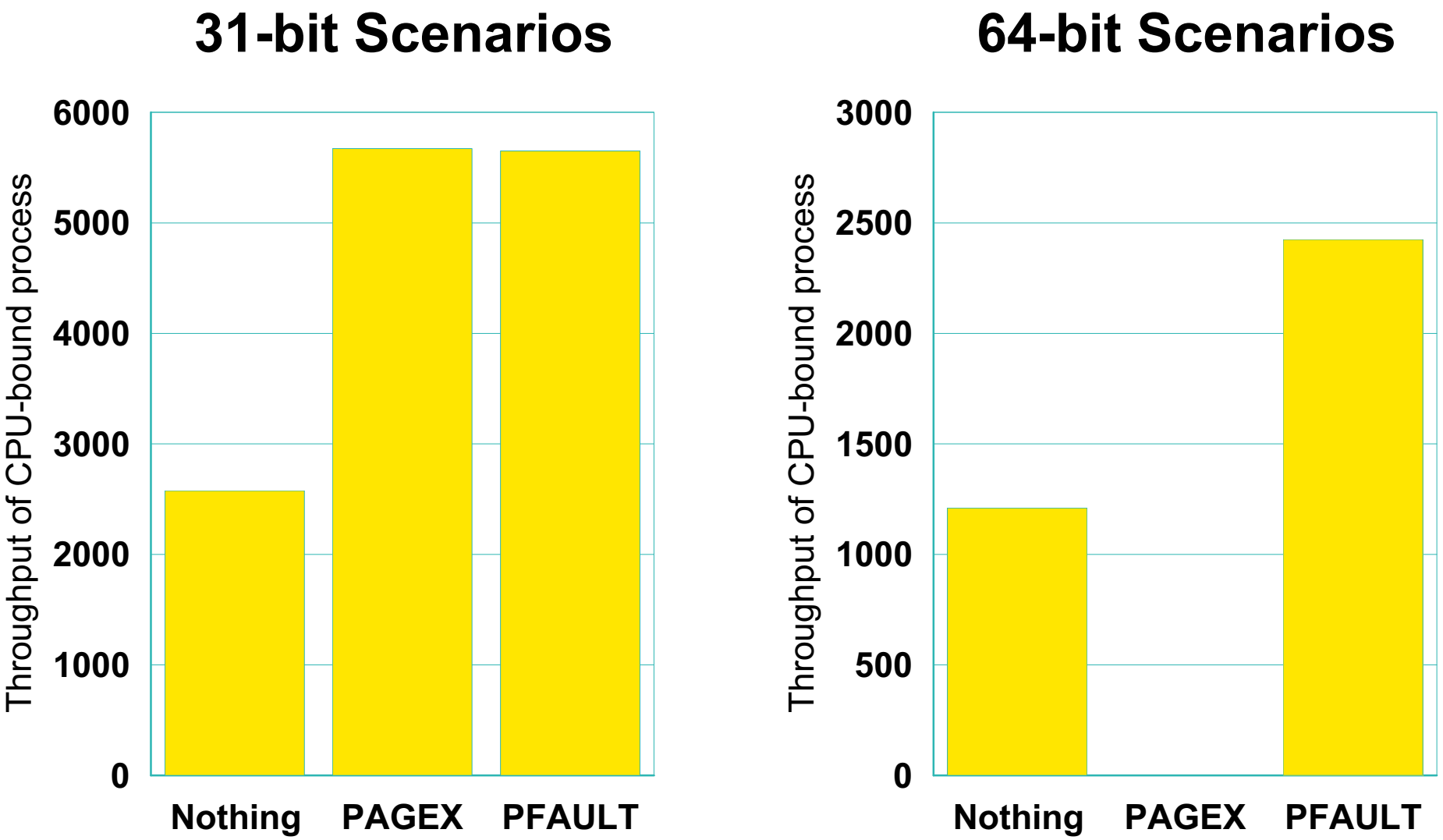
- Guest paging:
  - For DAT-on guests the potential exists for "double paging"
    - Guest virtual not in guest real: guest handles a fault
    - Guest real not in host real: host handles a fault
  - Right-sizing guest storage reduces guest paging (aka swapping)
    - Maybe you need to run 64-bit Linux
    - However, oversizing Linux guests has negative effects
  - Use PAGEX and Asynchronous Page Fault where appropriate

- Host paging:
  - Configure z/VM's paging resources appropriately
    - Plenty of paging space (no more than 50% full)
    - Plenty of paging extents (not just one big paging pack)
    - Don't mix paging extents with other extent types
    - Spread load across chpids and DASD subsystems
  - VM uses expanded storage for high speed paging device
    - Define 25% of partition's storage as expanded, up to 2 GB
    - Paging hierarchy helps reduce the cost of wrong page-out choices
    - See http://www.vm.ibm.com/perf/tips/storconf.html for help

Asynchronous Page Fault Facility

- Ordinarily, page faults serialize the virtual machine. This can be a throughput and response time problem for guest systems.
- Enhancements designed for Linux
- z/VM 4.2.0 and Linux 2.4 required (old hat nowadays)
- PFAULT macro
  - Accepts 64-bit inputs
  - Provides 64-bit PSW masks
- Diagnose x'258'
- Older PAGEX interface limited to 31-bit

Page Fault Measurements



VM Data-in-Memory Techniques

- VM Virtual Disk in Storage (VDISK)
  - Volatile FBA minidisk
  - Private or shareable
  - Can be used for the Linux swap file (be careful if system is storage-constrained)
- Minidisk cache
  - Undedicated 3380, 3390, 9345, RAMAC®, and IBM ESS boxes
  - SSCH and Diagnose I/O
  - Read-once data generally does not benefit
  - NOMDCFS lets servers overconsume MDC
  - Define some central storage for MDC

Contention for Memory below 2 GB (z/VM 5.1)

- z/VM lets a guest use 64-bit, but itself still uses 31-bit (mostly)
  - Guest pages have to be pulled below the 2 GB bar for CP to manipulate them
- Permanent structures below the bar:
  - V=R/F areas (pre-z/VM-5)
  - CP nucleus
  - Frame table
  - CP control blocks (RDEVs, etc.)
  - Segment tables
  - Data structures for dedicated QDIO (FCP) devices
  - Locked guest pages
- Temporary structures below the bar
  - Guest pages containing channel programs or I/O buffers
  - Guest pages associated with Guest LAN or VSWITCH
  - Page management blocks (VDISK ones are permanent)
  - Others
- Result: memory below 2 GB is precious

Am I Experiencing Below-2-GB Contention?

- Page pull-downs per second per guest (PerfKit FCX113 >2GB>)
- Size of available list (PerfKit FCX143 Avail)
- Pass 2 and emergency scan rates (PerfKit FCX102)
- >1 engine's worth of system time (PerfKit FCX225)
- Elevated %IO wait in PerfKit user states report

Reducing Below-2-GB Contention

- Minimize use of dedicated OSA, FCP, and HiperSockets devices
  - ►Use a Linux or z/VM TCP/IP router, or use VSWITCH (z/VM 4.4 or later)
  - ►If you must dedicate an OSA, consider tuning QDIO buffers down
- Minimize size and number of VDISKs
- Minimize size of Linux guests (fewer pages = fewer I/O buffers)
- Run the Linux "fixed I/O buffer" patch (SLES 9 SP1 and RHEL 4)
- Use MDC
- Run multiple LPARs instead of one giant LPAR
- Tune applications (for example, Oracle direct I/O patch)
- **Migrate to z/VM 5.2.0**
- See www.vm.ibm.com/perf/tips/2gstorag.html for details

Linux Guests

Linux Guest Guidelines

- **Why does my idle Linux consume processor resources?**
  - Timer pops (z/VM 4.4.0 scheduler lock relief helps with this)
  - Easy to turn off timer pops in SLES 8 or later
- **Is the number and size of guests important?**
  - Yes! It is virtual storage, but it isn't magic. It has to reside somewhere when Linux guest is running.
- **How big should my Linux guest be?**
  - Not bigger than you need
  - compare /proc/dasd/statistics to VM monitor data
- **Where should Linux swap?**
  - Multiple choices: XPRAM, minidisk, dedicated disk, T-disk, VDISK, DCSS
  - Be careful when using VDISK in storage-constrained environments
- **Should I set QUICKDSP ON for my Linux guest?**
  - Production vs. test vs. development machines
- **How many virtual processors should I define?**
  - Not more than there are in the partition, and the right number for the workload
- See the following URL for other information: www.vm.ibm.com/perf/tips/linuxper.html
- See APAR VM63282 for better dispatch list management (applies mostly to storage-constrained environments)

Swapping Configuration

- The trade-off
  - Defining virtual machine too large may cause excess memory to be used inefficiently for file and buffer cache.
  - Defining virtual machine too small may cause swapping which is expensive in processor time and impacts response time.

- Configure so that swap rate is zero or very low.
  - Maybe you need to be using a 64-bit Linux

- Virtual disk in storage (VDISK) can be used to mitigate cost of swapping.
  - Pros:
    - Very easy from administration view
    - Virtual disk blocks not created unless referenced
    - Performance
  - Cons:
    - DAT structures required below 2 GB and are not pageable
    - Steal algorithms tend not to take VDISK pages
    - Disk blocks reside below 2 GB prior to z/VM 4.4.0
    - Linux workloads tend to be storage-constrained anyway
  - See www.vm.ibm.com/perf/tips/lxswpvdk.html for guidance

- Do not define virtual disks in storage larger than necessary

Networking

Networking Choices

- Lots of variations for connecting:
  - Guests to other guests
  - Guests to another LPAR
  - Guests to external network
- Continued improvement in both Linux and VM stacks
  - z/VM 4.4.0 TCP/IP and VSWITCH are notable
- Workload-dependent
  - MTU impact
  - Below-2-GB storage impact (z/VM 5.1 or earlier)
  - Performance may improve as load increases
    - Data rate and number of connections

Guest to Guest

- Guest LAN
  - Simulated HiperSockets
  - Slightly lower pathlength than simulated GbE
  - Use with the virtual switch (z/VM 4.4.0 or later)
  - No persistent locking of guest pages
  - No architected limits on number of LANs
- HiperSockets
  - Requires locking real memory below 2 GB (z/VM 5.1 or earlier)
  - Configuration limitations
    - Numbers of HiperSockets chpids and IP addresses
  - Better performance for large data transfers

Guest to Another LPAR

- HiperSockets
  - Best solution
  - Pay attention to MFS (MTU)
- Shared OSA GbE
  - Additional overhead and latency even when shared card

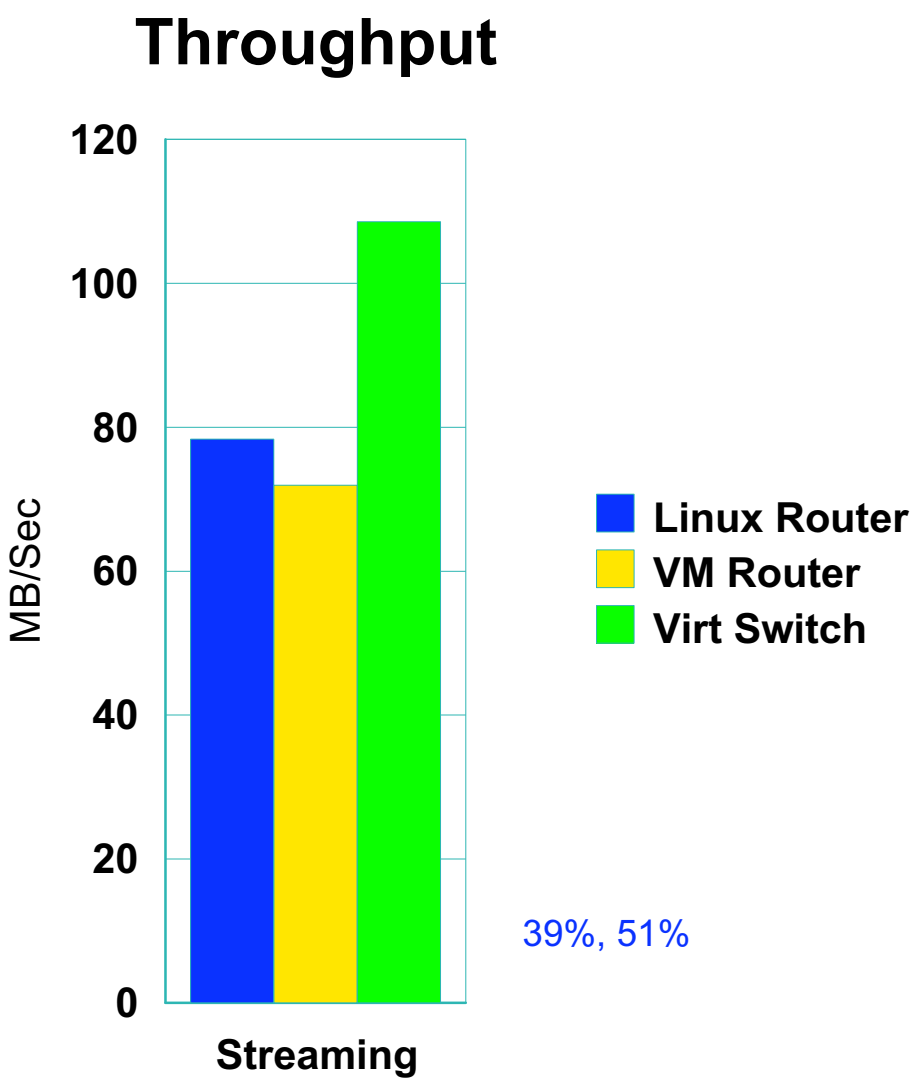
Guests to External Network

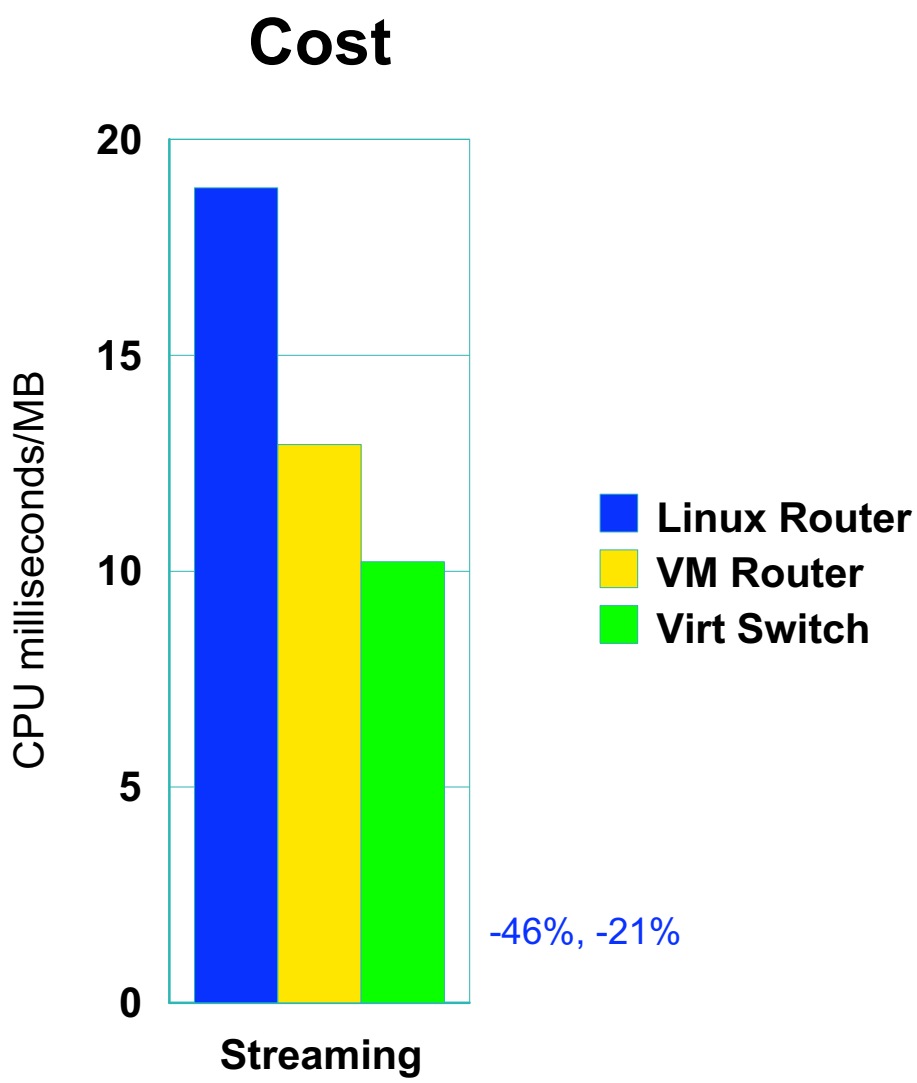
- Guests direct connect to OSA
  - Lowest pathlength, especially with z/VM 4.4.0 and hardware that supports Queued I/O Assist
  - Requires locking real memory below 2 GB
- Virtual switch
  - Requires z/VM 4.4.0 or higher
  - Very good performance characteristics
- Virtual machine router
  - Extra pathlength for moving and processing data

Virtual Switch - New in z/VM 4.4.0

- Layer 3 (IP packet) switch
  - Switches IP packets between QDIO guest LAN and OSA Express physical network
  - Eliminates need for IP (layer 3) router
  - Switching function performed entirely by CP
  - z/VM TCP/IP stack used for setup and control functions
- Layer 2 support in z/VM 5.1.0
  - With the PTF for APAR VM63538 and PQ98202
- Reduces processor resource requirements for some environments

Virtual Switch - Streaming (MTU 8992)





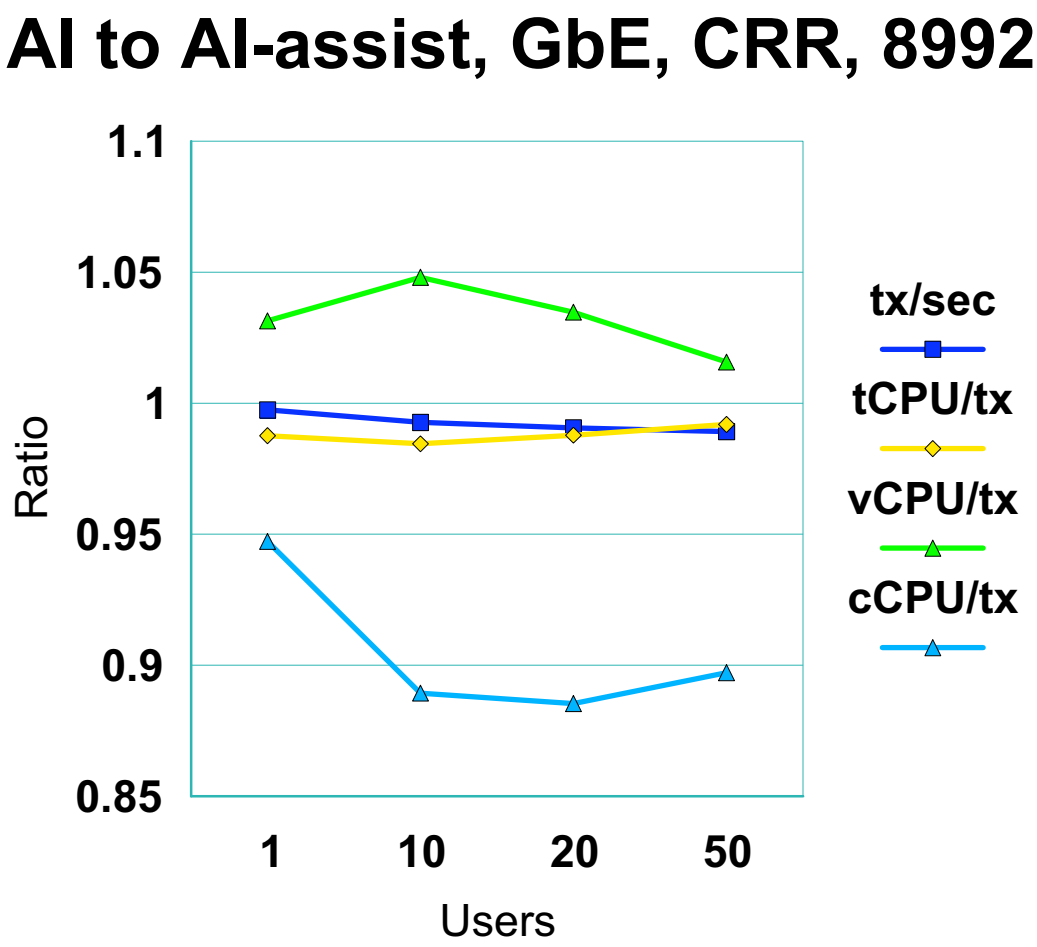
Queued I/O Assists

- QDIO devices (FCP, OSA Express, HiperSockets) induce overhead due to high interruption rates
  - ►z/VM Control Program has to mediate between hardware interruptions and guests
  - ►As interruption rates go up, this overhead increases
- QDIO Assist Part 1 (z/VM 4.4.0, z990 or z890 or z9) helps somewhat
  - ►Lets hardware present QDIO interrupts to guest, if guest happens to be in SIE
  - ►If target guest is idling, hardware rearranges CP control blocks and delivers "thin" signal
  - ►Works for HiperSockets, QDIO, and FCP
- QDIO Assist Part 2 (z/VM 5.2.0, z990 or z890 or z9) helps even more
  - ►Guest stays in SIE when it issues SIGA
  - ►Almost all of the Control Program mediation is now gone
- Changes in z/VM and Linux to take advantage of these assists
  - ►QUERY/SET QIOASSIST
- See www.vm.ibm.com/perf/aip.html and .../qebsm.html for more information.
- There are commands to disable these assists if needed (problem determination)
- You will want VM63685

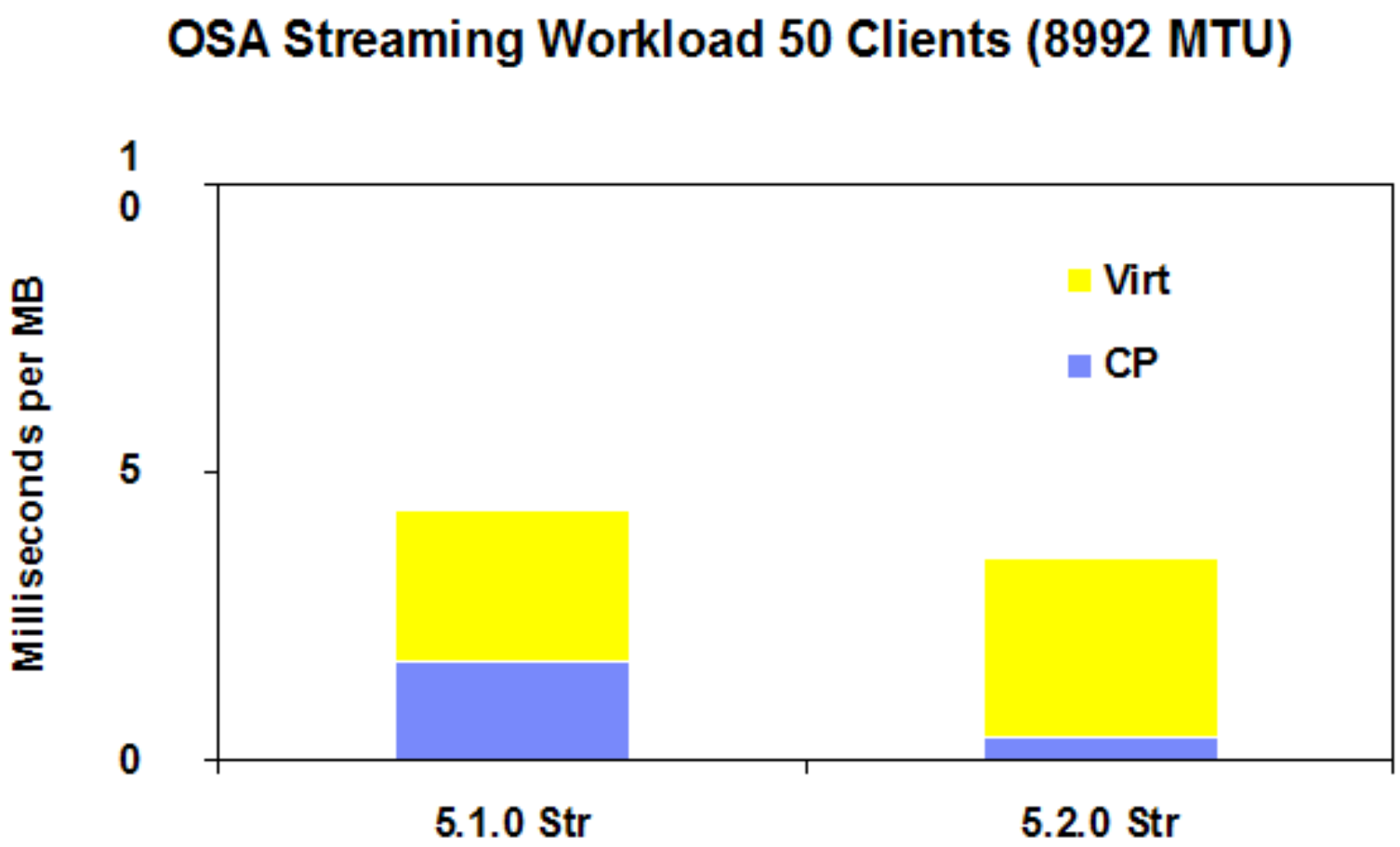
Al to Al-Assist, Linux, GbE

Generally, we see this:

- Tx/sec flat
- Small rise in virtual/tx
- Good drop in CP/tx



Effect of QDIO Assist Part 2

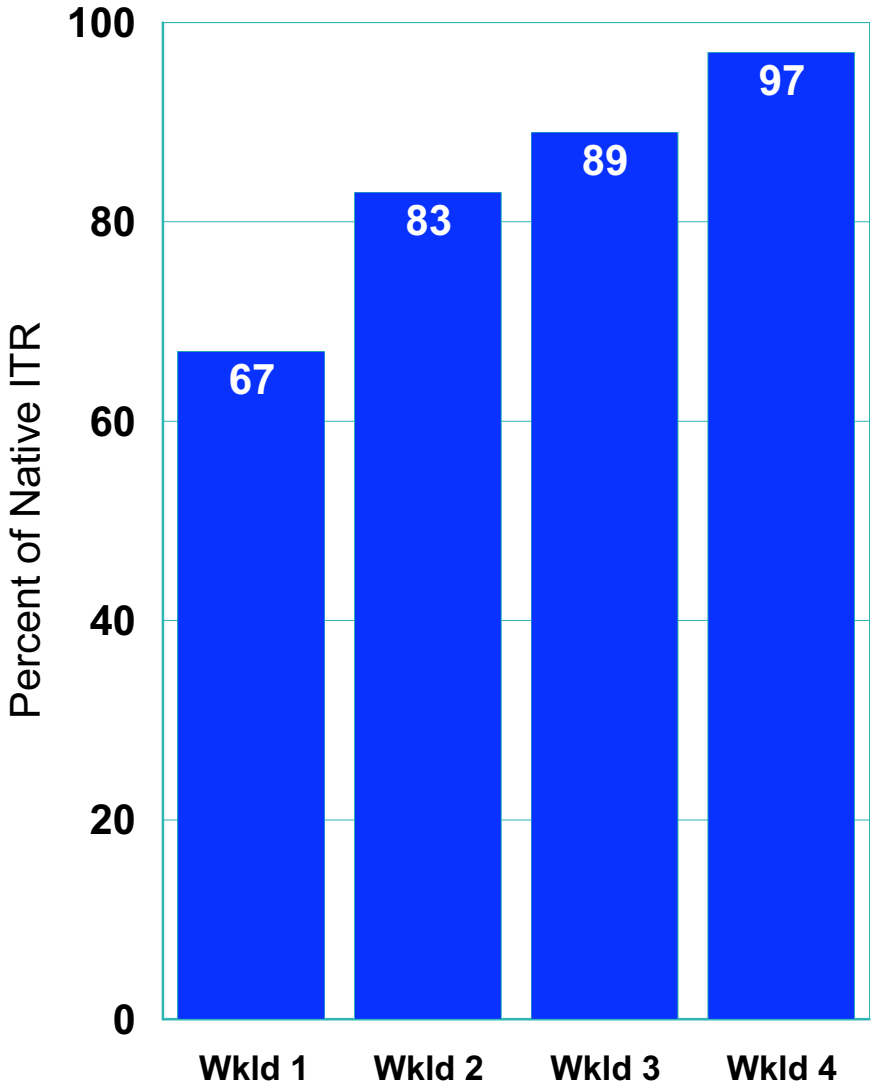


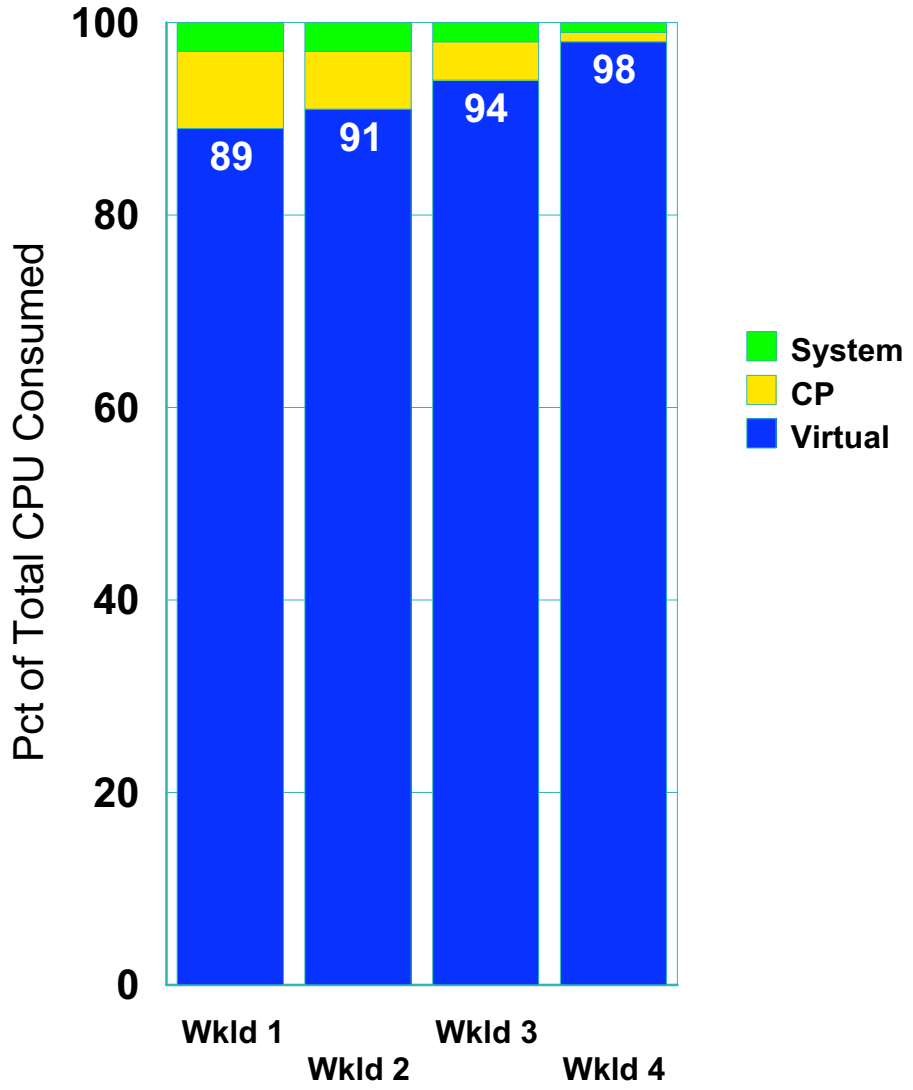
The Cost of VM

Two Views of VM Costs

- Direct comparison measuring VM guest compared to Linux in an LPAR
  - Usually assessed using ITR (Internal Throughput Rate... tx/CPU-sec)
  - Often described as "percent of native ITR"
  - Arguably meaningless (management or configuration considerations)
  - **Good news:** z/VM 5.2 helps a lot on this front
- How much processor time is consumed by VM code
  - Measured through products reducing VM monitor data
  - Three components
    - Virtual (aka emulation): time spent in guest code
    - CP: time spent in VM code directly in support of a particular guest
    - System: time spent for general VM system management, not connected directly to any guest
  - "Overhead" sometimes equated to time spent in CP or in CP+System
- Other views do exist:
  - Price to leverage unused resources and manage more effectively
  - Response time impacts
  - Throughput impacts

Two Views of VM Costs
Single VM Guest vs. LPAR





VM Costs - Summary

- "It depends"
  - Range is much greater than shown in example
  - Consult the IBM and vendor resources for sizing
  - Depends on SIE interaction intensity (SIEs per million guest instructions)
  - Depends on HW and SW configuration
- Be sure you know what is meant by "overhead" or "cost"
- Differences between the two views include:
  - Fraction-spent-in-VM does not capture loss of MIPS due to sharing processor cache, different instruction mix, etc.
  - Percent-of-native-ITR does not capture value of VM

Summary

Summary

- Stay in SIE. Every exit means the guest is not running.
- Many features to be exploited
- The answer is, "It depends. With Linux, it depends even more."
- Optimum configuration will depend on:
  - What you mean by the term "performance"
  - What you mean by the term "good"
  - What resources you have available
- See VM web site for additional information:
    - www.vm.ibm.com
    - www.vm.ibm.com/perf/
    - www.vm.ibm.com/perf/tips/

RETURN TO INDEX