IBM z Systems

Managing z/VM and Linux Performance Best Practices

Mike Sine, IBM

August 2015

IBM

Special Notices and Trademarks

Special Notices

This presentation reflects the IBM Advanced Technical Skills organizations' understanding of the technical topic. It was produced and reviewed by the members of the IBM Advanced Technical Skills organization. This document is presented "As-Is" and IBM does not assume responsibility for the statements expressed herein. It reflects the opinions of the IBM Advanced Technical Skills organization. These opinions are based on the author's experiences. If you have questions about the contents of this document, please contact the author at sine@us.ibm.com .

Trademarks

The following are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

IBM, the IBM logo, Candle, DB2, developerWorks, iSeries, Passport Advantage, pSeries, Redbooks, Tivoli Enterprise Console, WebSphere, z/OS, xSeries, zSeries, System z, z/VM.

A full list of U.S. trademarks owned by IBM may be found at http://www.ibm.com/legal/copytrade.shtml.

NetView, Tivoli and TME are registered trademarks and TME Enterprise is a trademark of Tivoli Systems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, Internet Explorer, and the Windows logo are registered trademarks of Microsoft Corporation in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

Intel and Pentium are registered trademarks and MMX, Pentium II Xeon and Pentium III Xeon are trademarks of Intel Corporation in the United States and/or other countries.

Other company, product and service names may be trademarks or service marks of others.

AGENDA

- Introduction
- Monitoring Requirements
  - Virtual Linux and z/VM performance considerations
  - Don’t forget the hardware
  - Integration from hardware – systems – applications
    Persistent historical views
- Enterprise Management

Introduction

Products

- Tivoli OMEGAMON XE on z/VM and Linux
- Operations Manager for z/VM
- Backup and Restore Manager for z/VM
- Tape Manager for z/VM
- Archive Manager for z/VM
- IBM Wave
- Tivoli Storage Manager (TSM)

IBM Infrastructure Suite

Linux on System z

OMEGAMON XE on z/VM and Linux

Performance monitoring of z/VM and Linux guest

Tivoli Storage Manager

File Level backup and recovery for Linux Virtual Machines

z/VM

Wave for z/VM

Simple, intuitive, graphical z/VM management tool

Operations Manager for z/VM

Facilitate automated operations, take action based on events

Backup and Restore Manager for z/VM

Backup and restore full z/VM environment

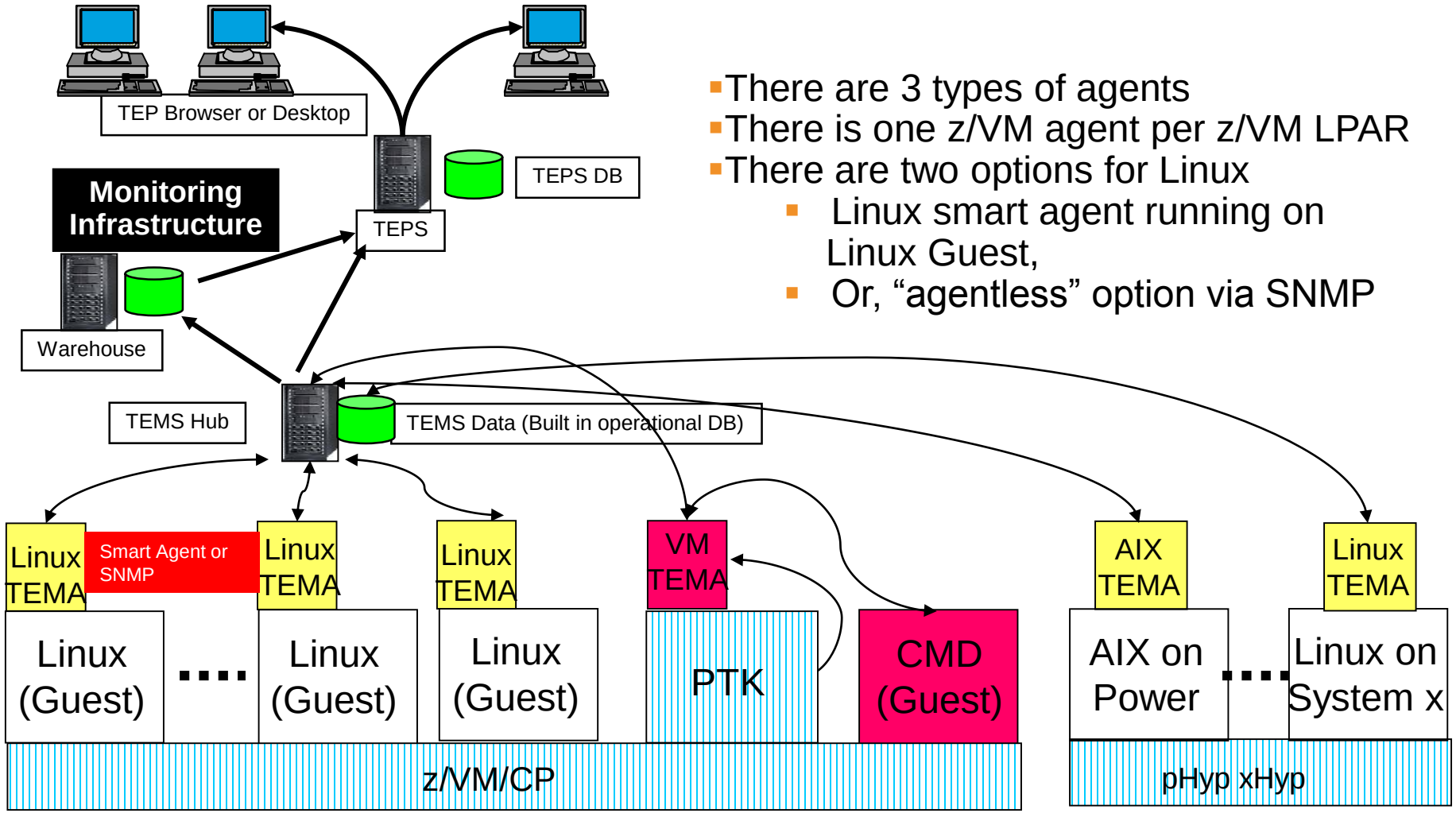
Virtual Linux servers have unique challenges versus running on physical machines

- z/VM System Programmers and Linux Administrators may not be in the same organization.
- We find that it is easy to over allocate resources; therefore, our monitoring examines resource usage of hardware, hypervisor, as well as the virtual machine. Real-time and historical metrics demonstrate peaks periods as well as average runtimes.
- Cross-platform virtualization increases these challenges

AGENDA

- Introduction
- Monitoring Requirements
  - Virtual Linux and z/VM performance considerations
  - Don’t forget the hardware
  - Integration from hardware – systems – applications Persistent historical views
- Enterprise Management

OMEGAMON XE on z/VM and Linux agents with ITM example



- There are 3 types of agents
- There is one z/VM agent per z/VM LPAR
- There are two options for Linux
  - Linux smart agent running on Linux Guest,
  - Or, “agentless” option via SNMP

Highly scalable across the entire Enterprise

An Integrated Monitoring Approach

- Provides performance monitoring for z/VM and Linux guests
- Executes automated actions in response to defined events or situations (monitoring without automation is overhead)
- Integrates and Scales well across Enterprise for central control and trending:
  - Specifically focused on z/VM and Linux guests
  - Able to integrate z/VM and Linux into Enterprise Solution
  - Data warehousing for trend analysis (interactive and batch)

Workspaces to Manage z/VM and Linux

z/VM

- **Processors**
- **SYSTEM Utilization, spinlocks**
- **Workload**
  - Linux Appldata
  - Scaled & total CPU values
- **LPAR Utilization**
- **PAGING and SPOOLING Utilization**
- **DASD**
- **Minidisk Cache**
- **Virtual Disks**
- **Channels**
- **CCW Translation**
- **REAL STORAGE Utilization**
- **NETWORK Utilization (Hiper Socket and Virtual Switch)**
- **TCPIP Utilization – Server**
- **TCPIP Utilization – Users**
- **Resource Constraint (Wait states)**
- **System Health**

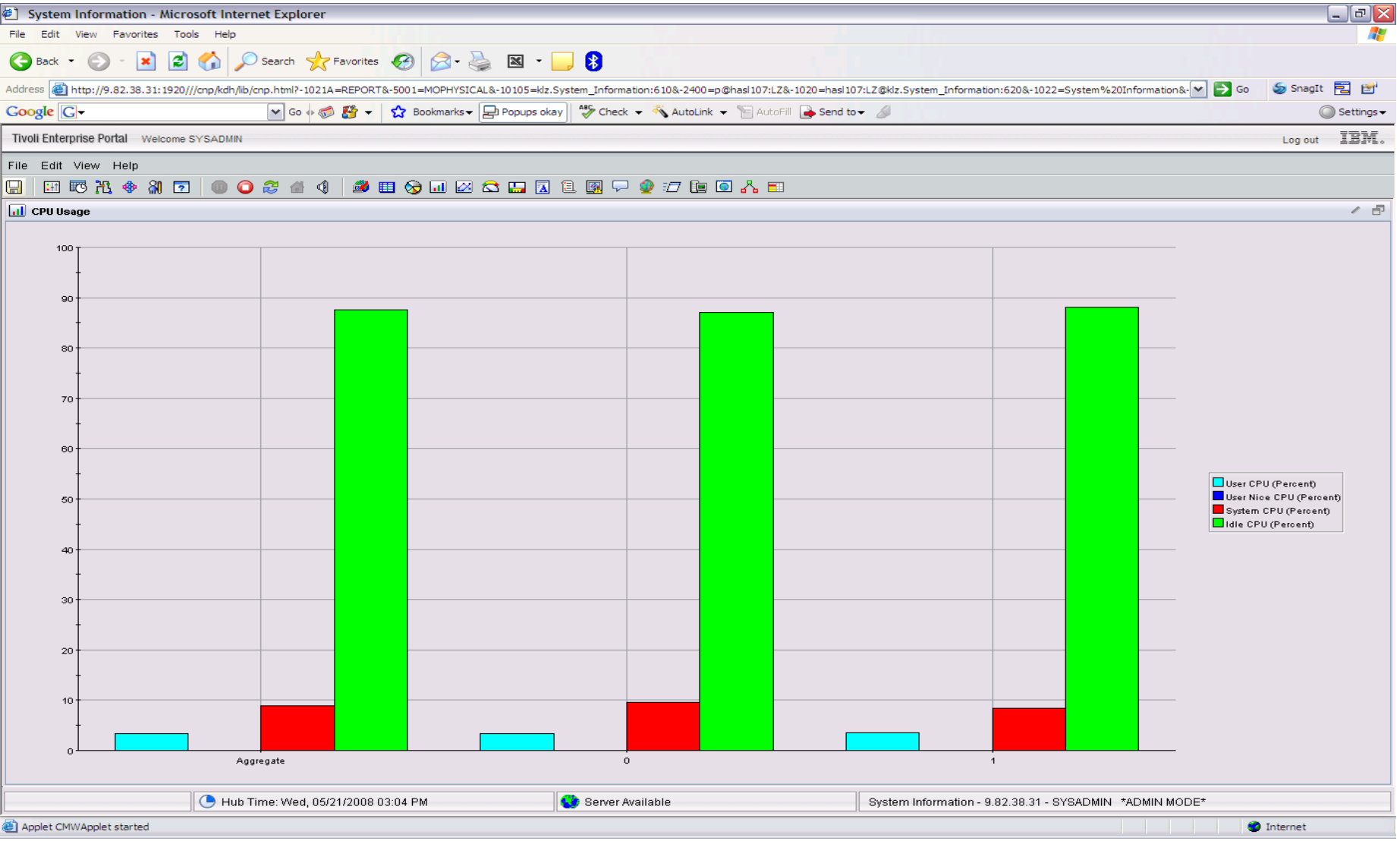
Linux

- **Linux OS**
- **System Information**
  - CPU aggregation
  - Virtual Memory Statistics
- **Process**
- **Users**
- **Disk Usage**
- **File Information**
- **Network**

Have I allocated enough Virtual CPUs to my guest?

- Do not define more virtual CPUs for a Linux guest than are needed.
  - The use of more than one processor requires software locks so that data or control blocks are not updated by more than one processor at a time.
- Linux makes use of a global lock, and when that lock is held, if another processor requires that lock, it spins.
  - Set the number of virtual processors based on need and not simply match the number of real that are available.
  - Careful when cloning as some Linux guests require more Virtual CPUs (ex: Running Websphere, Oracle) than others.

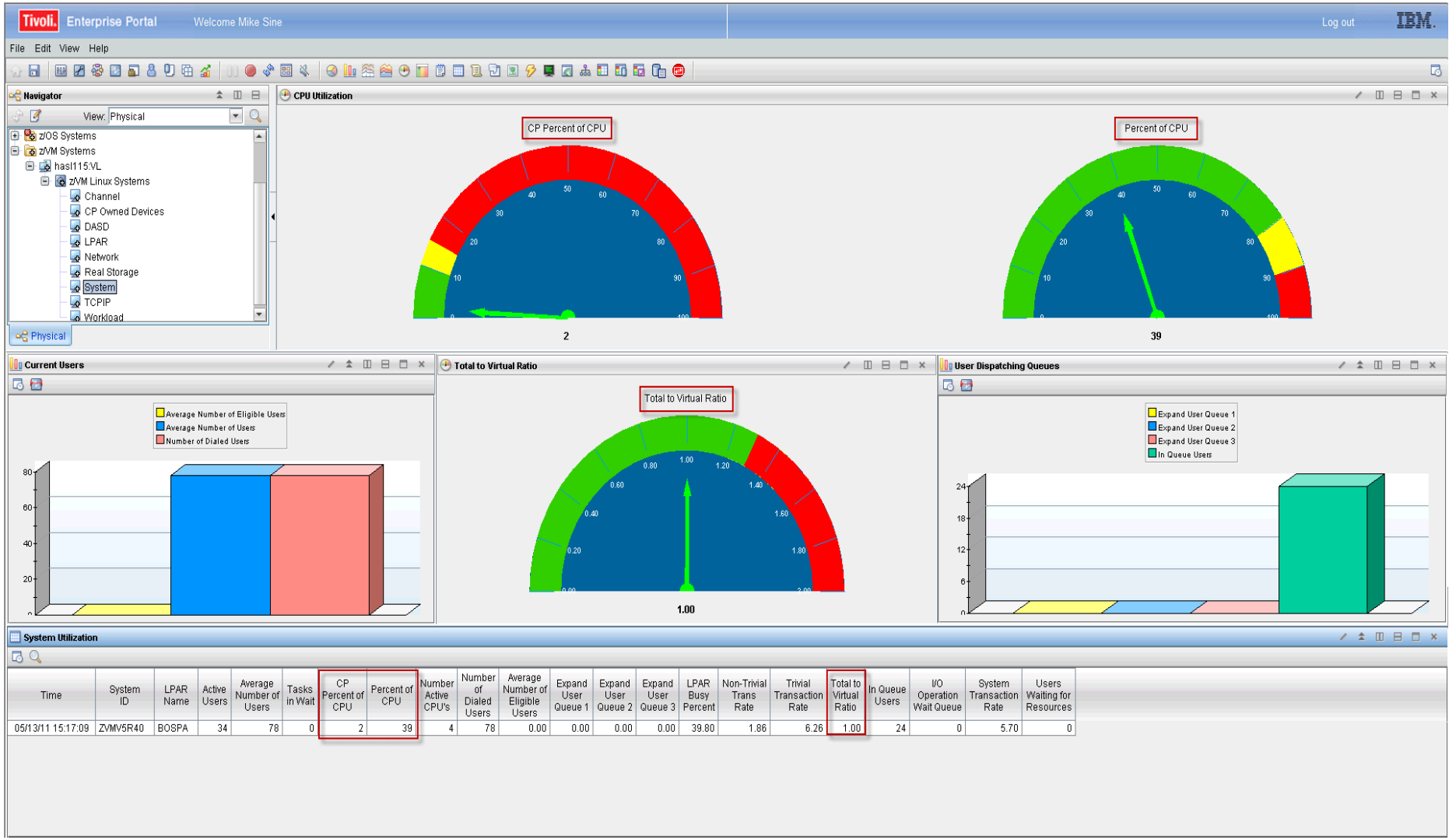
Aggregate monitoring of Virtual CPUs



z/VM Processor Utilization

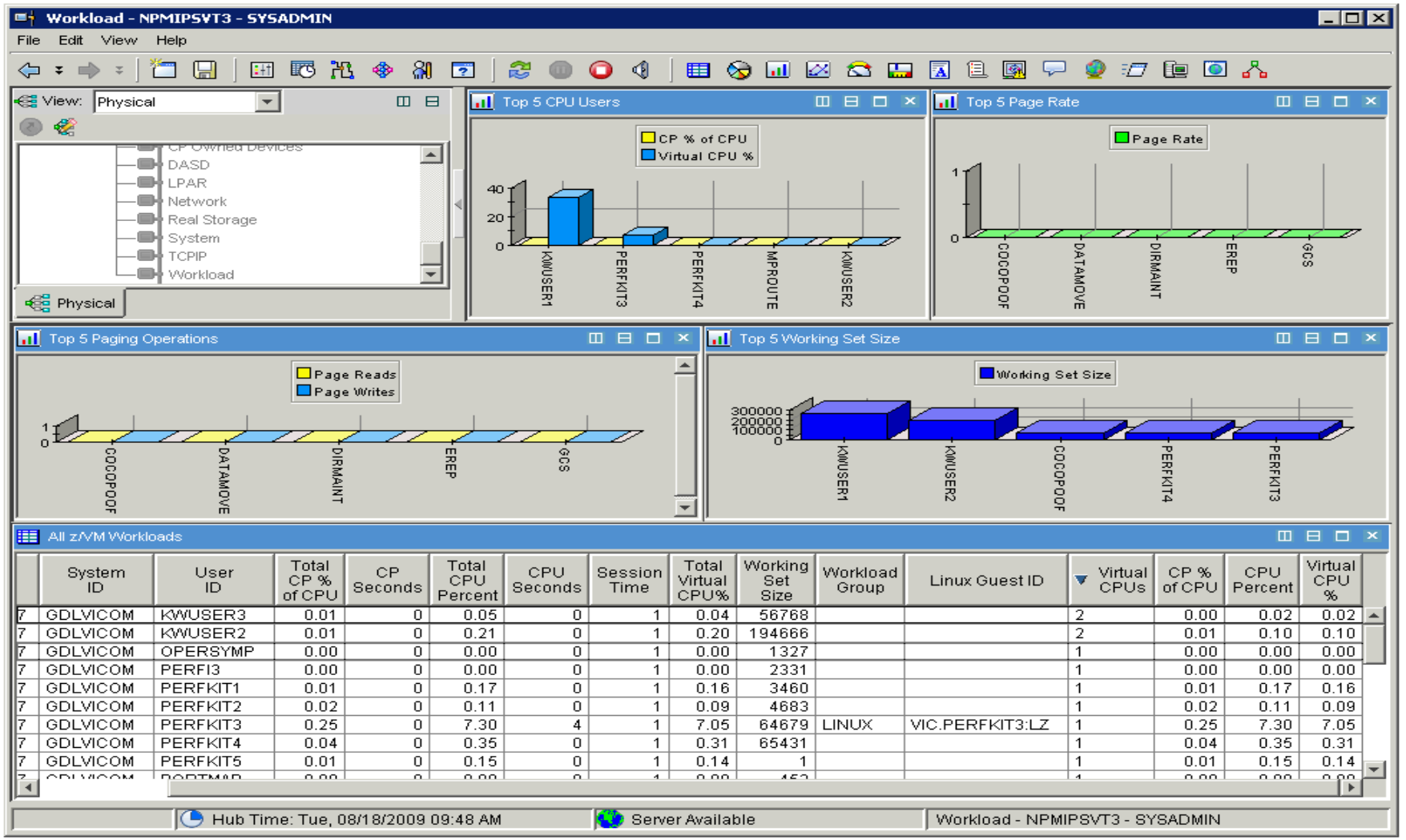
- **Total Processor Utilization** This is the processor utilization from the VM perspective and includes CP, VM System, and Virtual CPU time.
- **System Time:** This is the processor time used by the VM control program for system functions that are not directly related to any one virtual machine. This should be less than 10% of the total.
- **CP Processor Time:** This is the processor time used by the VM control program in support of individual virtual machines.
- **Virtual Processor Time: (Emulation Time):** This is processor time consumed by the virtual machine and the applications within it.
- **Total to Virtual Ratio** The ratio of total processor time to virtual processor time is often used as an indicator of z/VM efficiency or overhead. The closer to 1.0, the better the z/VM efficiency. RoT: Should explore causes of a ratio over 1.30.

System Processor Utilization Workspace



Time	System ID	LPAR Name	Active Users	Average Number of Users	Tasks in Wait	CP Percent of CPU	Percent of CPU	Number Active CPU's	Number of Dialed Users	Average Number of Eligible Users	Expand User Queue 1	Expand User Queue 2	Expand User Queue 3	LPAR Busy Percent	Non-Trivial Trans Rate	Trivial Transaction Rate	Total to Virtual Ratio	In Queue Users	I/O Operation Wait Queue	System Transaction Rate	Users Waiting for Resources
05/13/11 15:17:09	ZVMV5R40	BOSPA	34	78	0	2	39	4	78	0.00	0.00	0.00	0.00	39.80	1.86	6.26	1.00	24	0	5.70	0

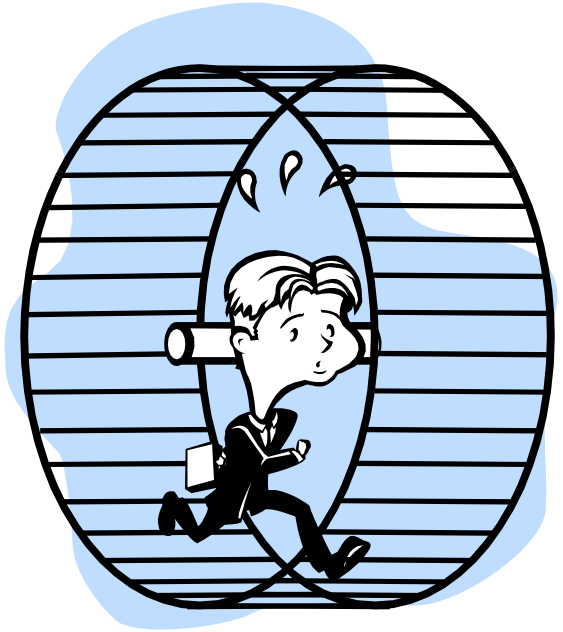
z/VM Workload Workspace



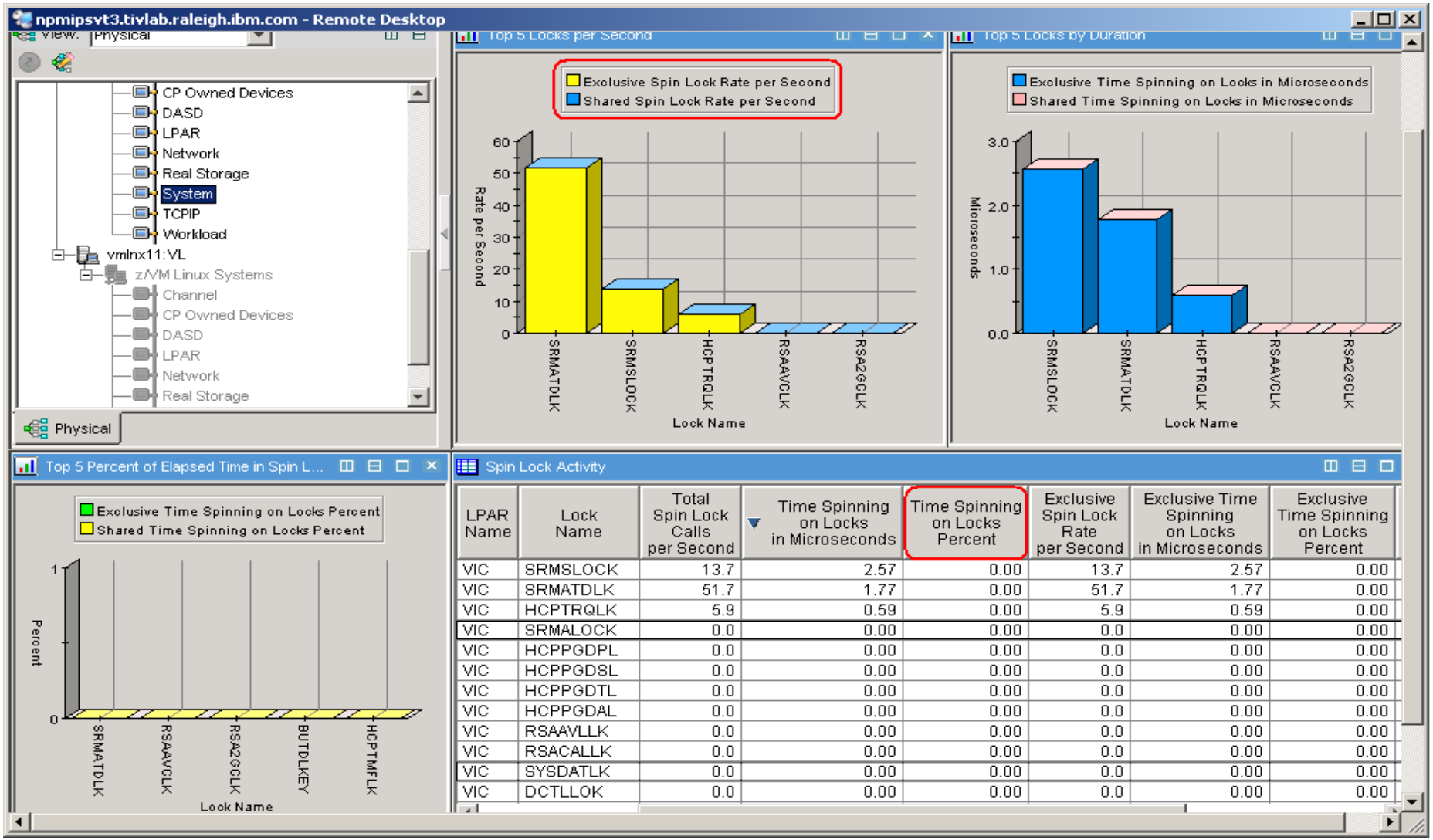
System ID	User ID	Total CP % of CPU	CP Seconds	Total CPU Percent	CPU Seconds	Session Time	Total Virtual CPU%	Working Set Size	Workload Group	Linux Guest ID	Virtual CPUs	CP % of CPU	CPU Percent	Virtual CPU %
7	GDLVICOM	KWUSER3	0.01	0	0.05	0	1	0.04	56768			2	0.00	0.02	0.02
7	GDLVICOM	KWUSER2	0.01	0	0.21	0	1	0.20	194666			2	0.01	0.10	0.10
7	GDLVICOM	OPERSYMP	0.00	0	0.00	0	1	0.00	1327			1	0.00	0.00	0.00
7	GDLVICOM	PERFI3	0.00	0	0.00	0	1	0.00	2331			1	0.00	0.00	0.00
7	GDLVICOM	PERFKIT1	0.01	0	0.17	0	1	0.16	3460			1	0.01	0.17	0.16
7	GDLVICOM	PERFKIT2	0.02	0	0.11	0	1	0.09	4683			1	0.02	0.11	0.09
7	GDLVICOM	PERFKIT3	0.25	0	7.30	4	1	7.05	64679	LINUX	VIC.PERFKIT3:LZ	1	0.25	7.30	7.05
7	GDLVICOM	PERFKIT4	0.04	0	0.35	0	1	0.31	65431			1	0.04	0.35	0.31
7	GDLVICOM	PERFKIT5	0.01	0	0.15	0	1	0.14	1			1	0.01	0.15	0.14

Spin Lock Wait

- **Time Spinning on Locks Percent:**
  - The percentage of time processors spend spinning on formal spin locks. RoT: Should be less than 10%.
  - Increases as number of logical processors increases.



Spinlock Workspace

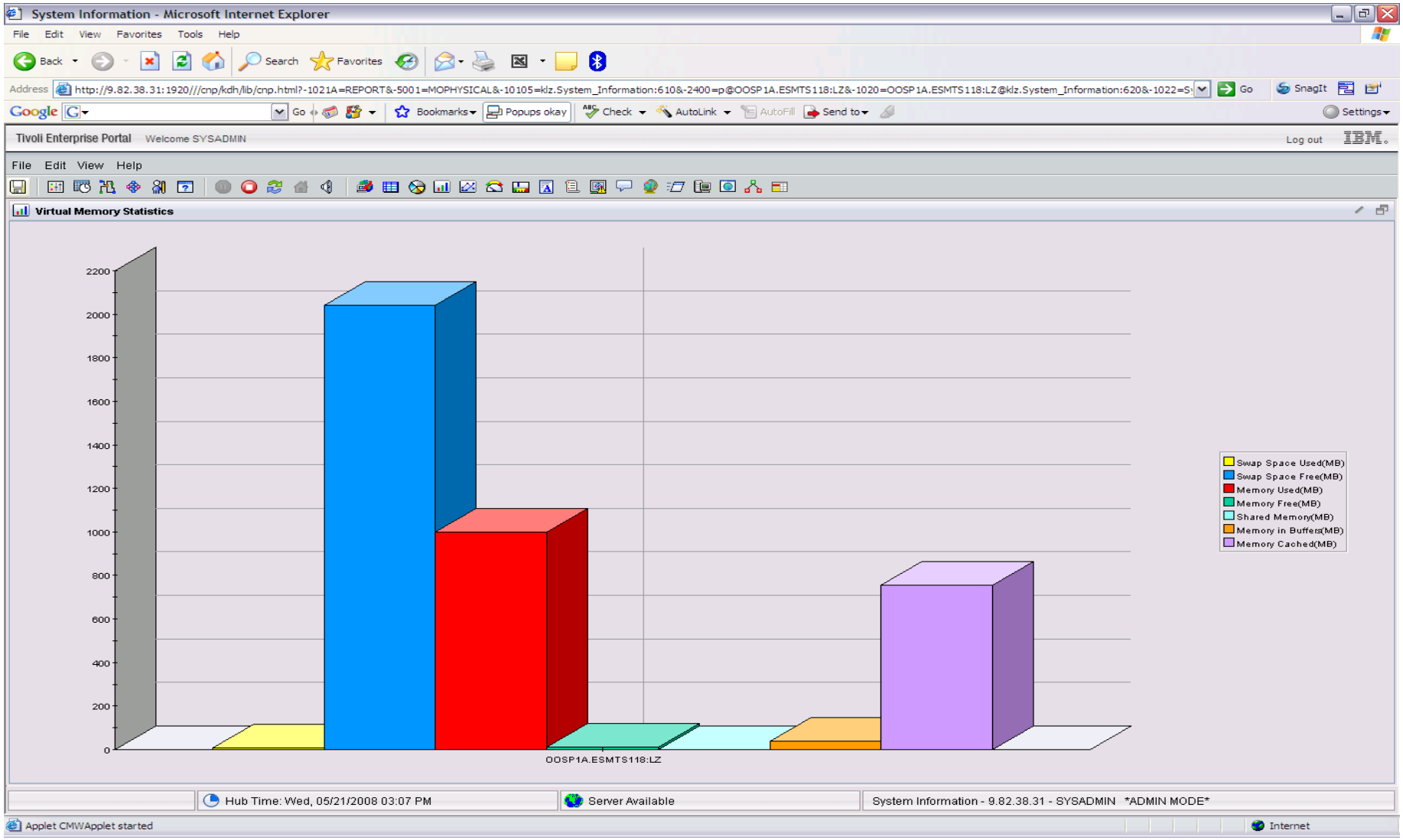


LPAR Name	Lock Name	Total Spin Lock Calls per Second	Time Spinning on Locks in Microseconds	Time Spinning on Locks Percent	Exclusive Spin Lock Rate per Second	Exclusive Time Spinning on Locks in Microseconds	Exclusive Time Spinning on Locks Percent
VIC	SRMSLOCK	13.7	2.57	0.00	13.7	2.57	0.00
VIC	SRMATDLK	51.7	1.77	0.00	51.7	1.77	0.00
VIC	HCPTRQLK	5.9	0.59	0.00	5.9	0.59	0.00
VIC	SRMALOCK	0.0	0.00	0.00	0.0	0.00	0.00
VIC	HCPPGDPL	0.0	0.00	0.00	0.0	0.00	0.00
VIC	HCPPGDSL	0.0	0.00	0.00	0.0	0.00	0.00
VIC	HCPPGDTL	0.0	0.00	0.00	0.0	0.00	0.00
VIC	HCPPGDAL	0.0	0.00	0.00	0.0	0.00	0.00
VIC	RSAAVLLK	0.0	0.00	0.00	0.0	0.00	0.00
VIC	RSACALLK	0.0	0.00	0.00	0.0	0.00	0.00
VIC	SYSDATLK	0.0	0.00	0.00	0.0	0.00	0.00
VIC	DCTLLOK	0.0	0.00	0.00	0.0	0.00	0.00

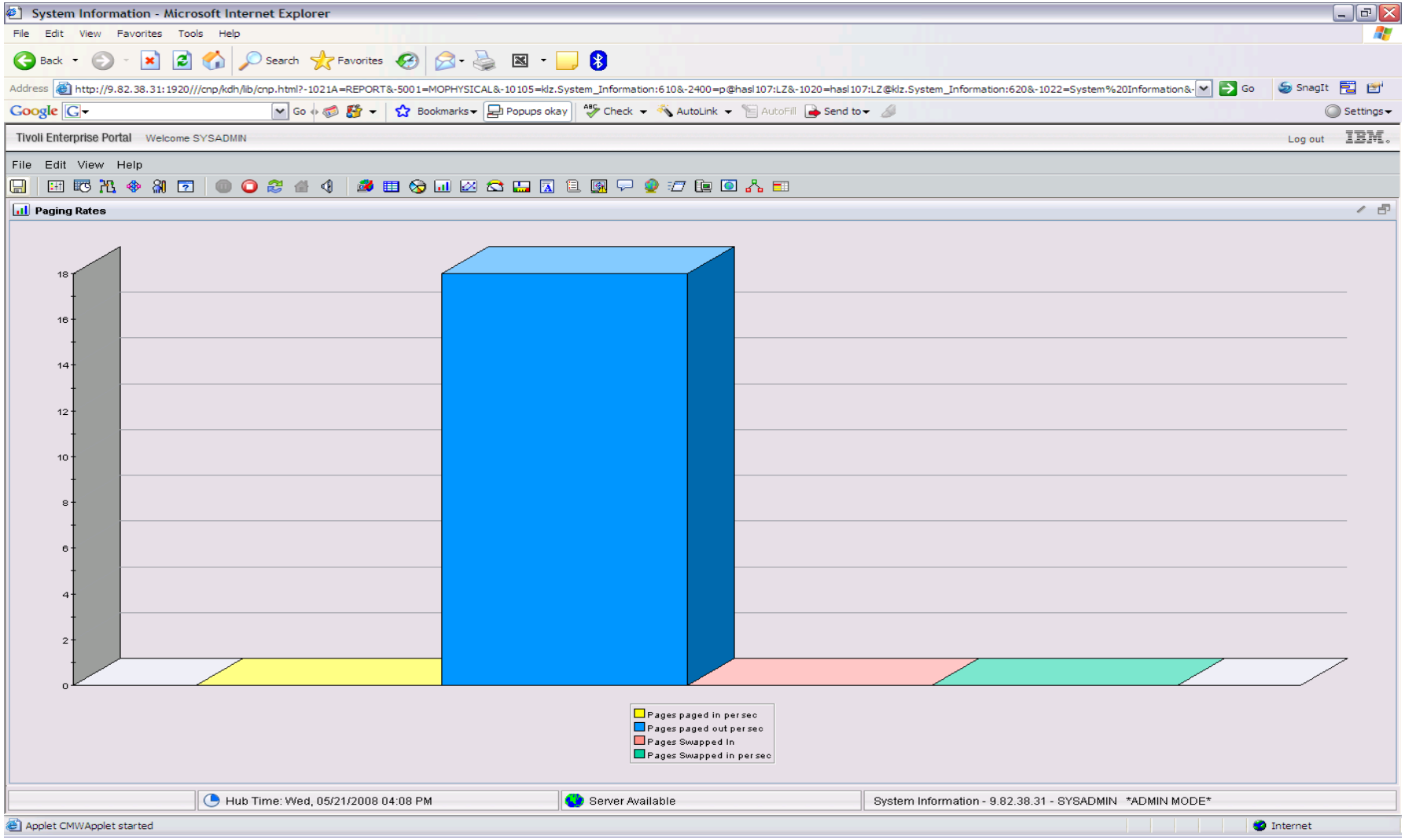
Is my Linux guest sized correctly?

- In general, do not define the Linux virtual machine larger than you need.
  - Excessive virtual machine sizes negatively impact performance.
  - Linux uses any extra storage for caching of data. For shared resources, this is an impact.
  - Reduce the size of the Linux guest until it starts to swap (use VDISK for swap).
  - Monitor swap usage.

Need breakdown of memory use



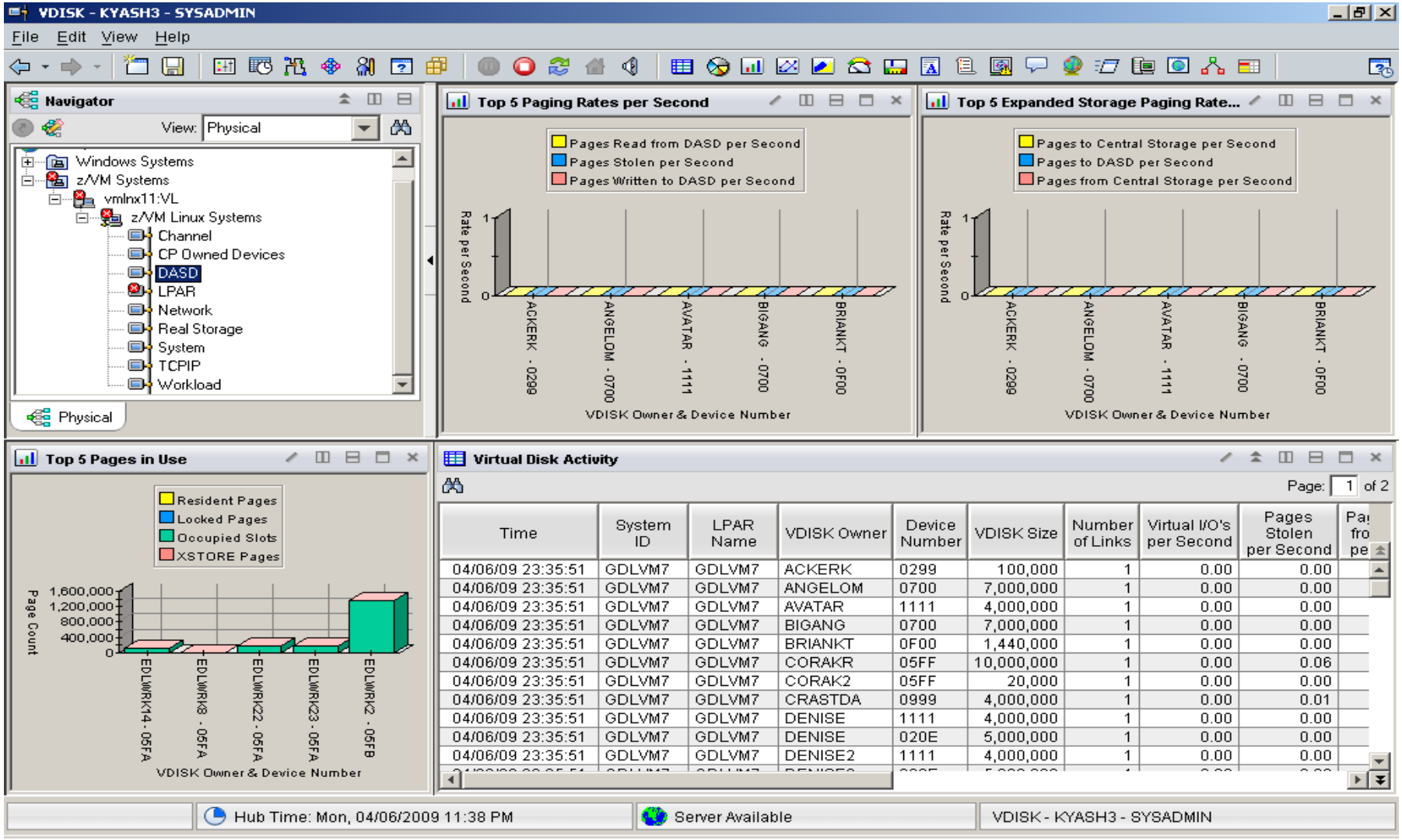
Page/Swap Attributes



VDISK

- What is it?
  - FBA (Fixed Block Architecture disk) device emulated in-memory
    - Translation: Very fast “device”.
  - High performance paging device for Linux on z.
  - Memory is allocated by CP from the Dynamic Paging Area
  - Allocated only when referenced
    - Allocating a 10 MB device does NOT instantly consume 10 MB of pages.
    - Pages are allocated when needed.
  - Not recommended in a storage-constrained z/VM system.

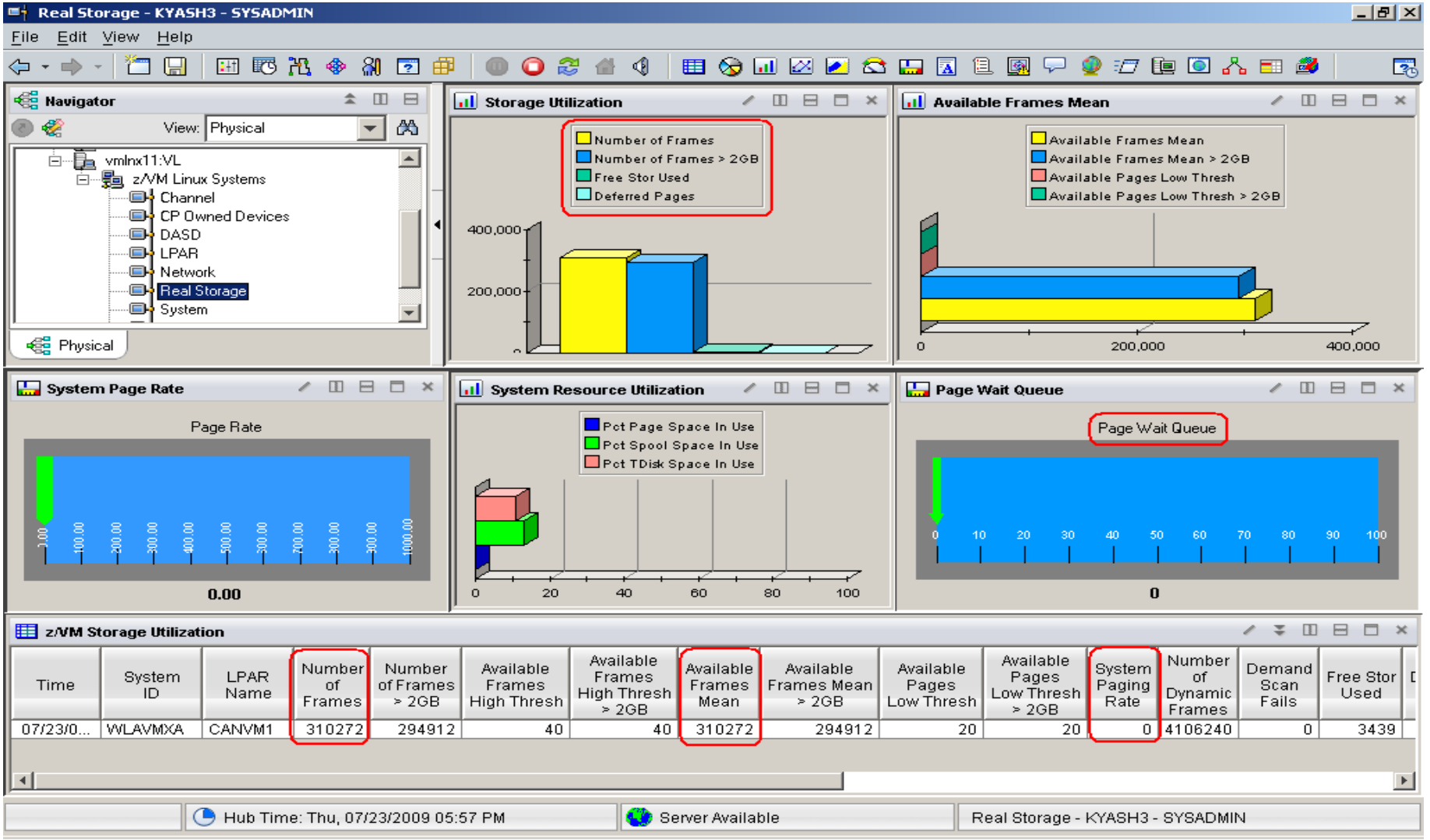
VDISK Workspace



Memory Configuration

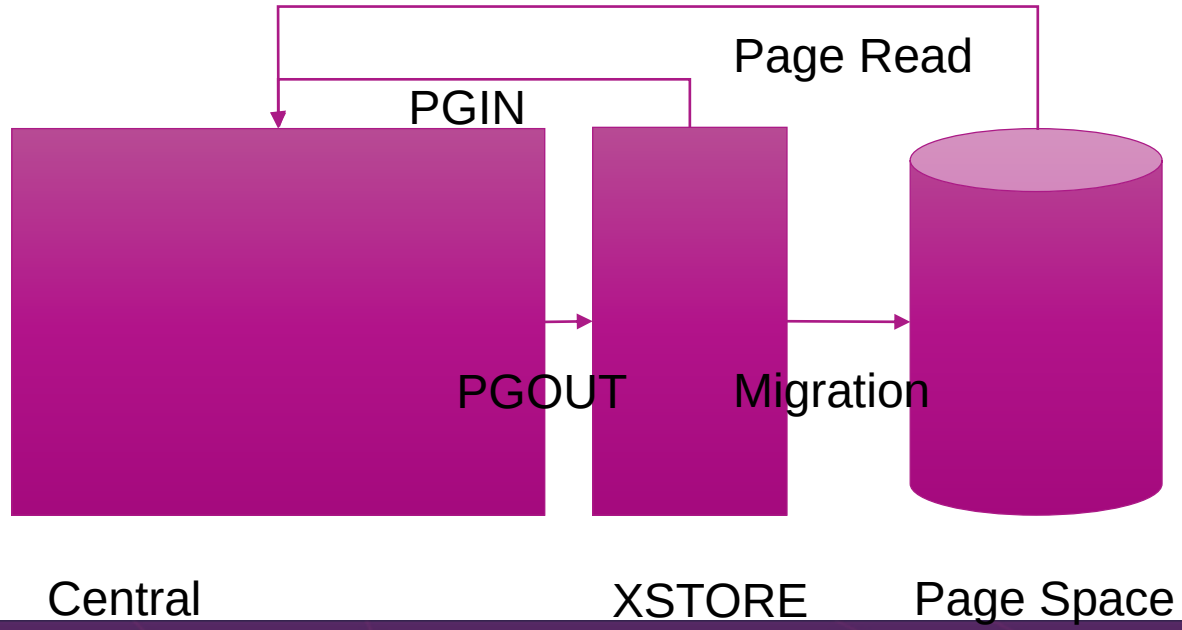
- Plan on a virtual to real (V:R) memory ratio in the range of 1.5:1 to 3:1.
- z/VM's 6.2 and earlier architecture still benefits from expanded storage:
  - Serves as high speed cache.
  - Increases consistency of response time.
  - See http://www.vm.ibm.com/perf/tips/storconf.html for the gory details.
- Rule of Thumb - start with 20-25% of memory configured as expanded:
  - The lower the paging rate, the lower the amount of expanded storage required.
  - The greater the number of page frames available in central storage above 2GB, the higher the amount of expanded storage required.
  - Some workloads 2–4GB of expanded storage is sufficient, 1GB minimum. However, more and more Linux systems are running heavy workloads and the 20-25% rule still applies.
  - XSTORE obsolete for z/VM 6.3

OMEGAMON Memory Configuration



z/VM's virtual storage prior to z/VM 6.3 consists of central storage, expanded storage (XSTORE), and page space.

- When central storage becomes constrained, determining what pages are candidates for removal is known as demand scan. Demand scan's algorithm selects pages to be moved to XSTORE until the constraint issue is reconciled. Moving pages from central to XSTORE is known as a PGOUT operation.
- When XSTORE is constrained the migration operation selects pages to go to DASD (page space). When a page is needed in central storage and it is found in XSTORE a PGIN operation brings the page from XSTORE to central storage. When the needed page is already on DASD, the operation to bring the page in from DASD to central storage is called a page read operation.



Block Paging pre z/VM 6.3

- To limit the number of I/O operations, z/VM 6.2 and earlier writes multiple pages to DASD per I/O operation.
  - This is called block paging.
- z/VM looks for continuous space to perform its block paging.
- For efficiency, recommend keeping z/VM page space less than 50% full
  - Increases likelihood that z/VM will find continuous space to perform its block paging operation.

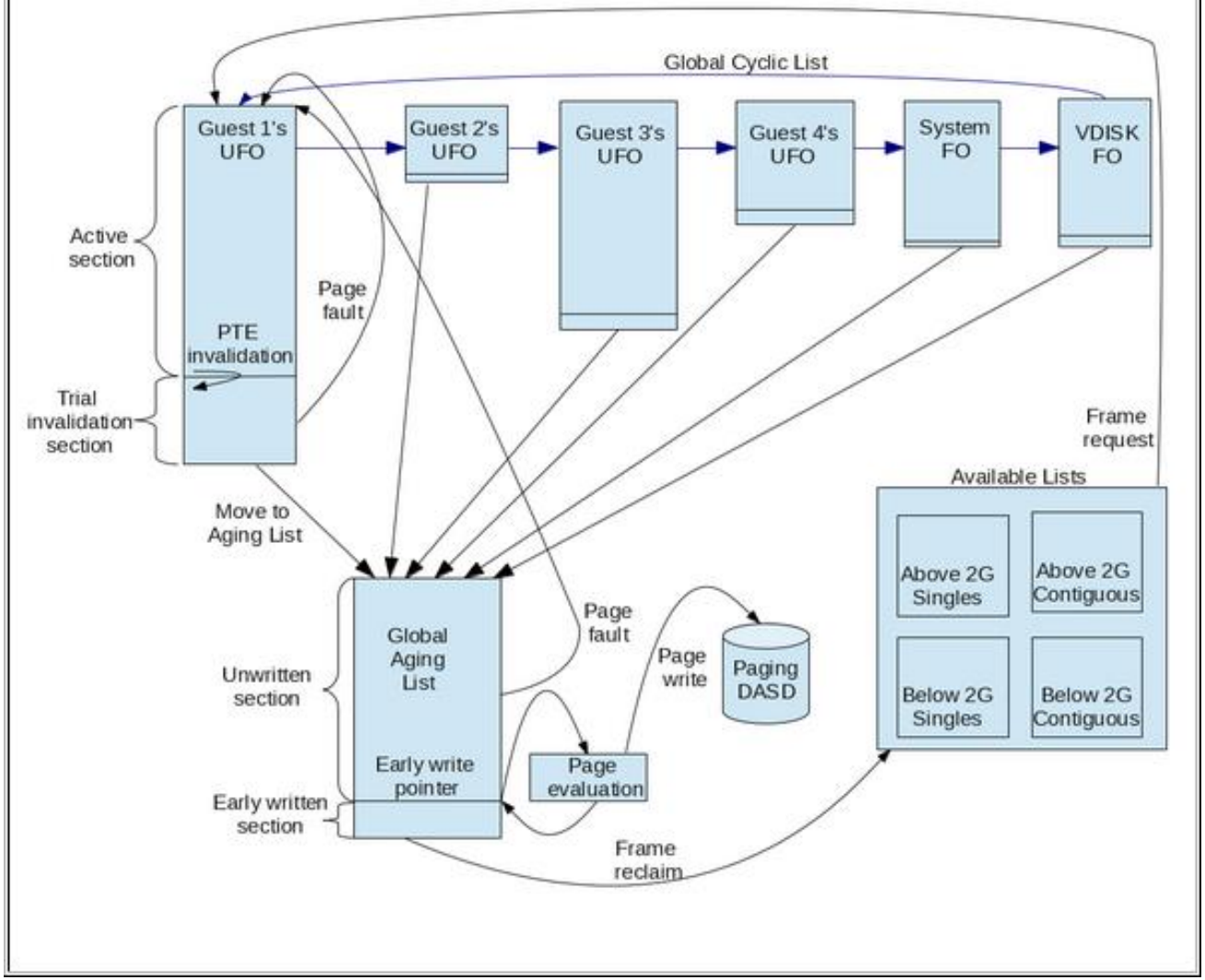
Page Space

Planning for z/VM 6.3 Large Memory

- **Normal best practices for migrating from an earlier release still apply.**
- **Change your paging XSTORE into central**
  - XSTORE provided an aging function. It helped catch reclaim selection "mistakes".
  - The new ***IBR*** concept and ***global aging list*** provide the same function but do so more efficiently in central.
- **Plan enough DASD paging space**
  - The system now pre-writes pages to DASD.
  - See z/VM manuals for detail recommendations.
- **Plan a robust paging DASD configuration**
  - Use plenty of paging volumes
  - Make the volumes all the same size
  - Put only paging space on the volumes you use for paging
  - Spread the paging volumes through your logical control units
  - Avoid logical control units that you know are hot on application I/O
  - Use plenty of CHPIDS
  - Do not use ESCON CHIPIDS (only carry one I/O at a time)
  - Do not mix ECKD paging and SCSI paging
  - Leave reserved slots in the CP-owned list

Gory details: http://www.vm.ibm.com/perf/reports/zvm/html/630mem.html

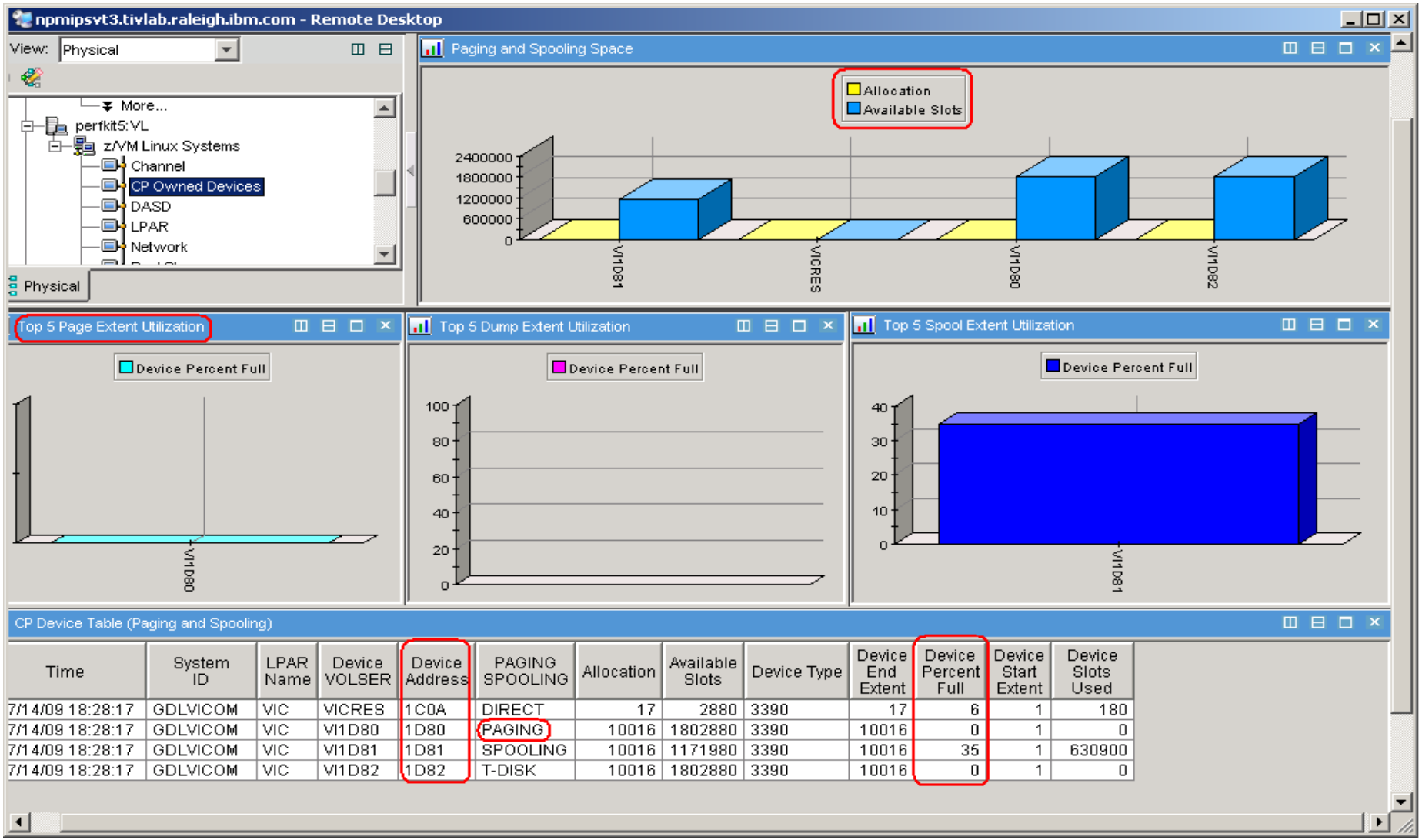
Figure 1. Memory Management Algorithm Visualization



Paging Subsystem

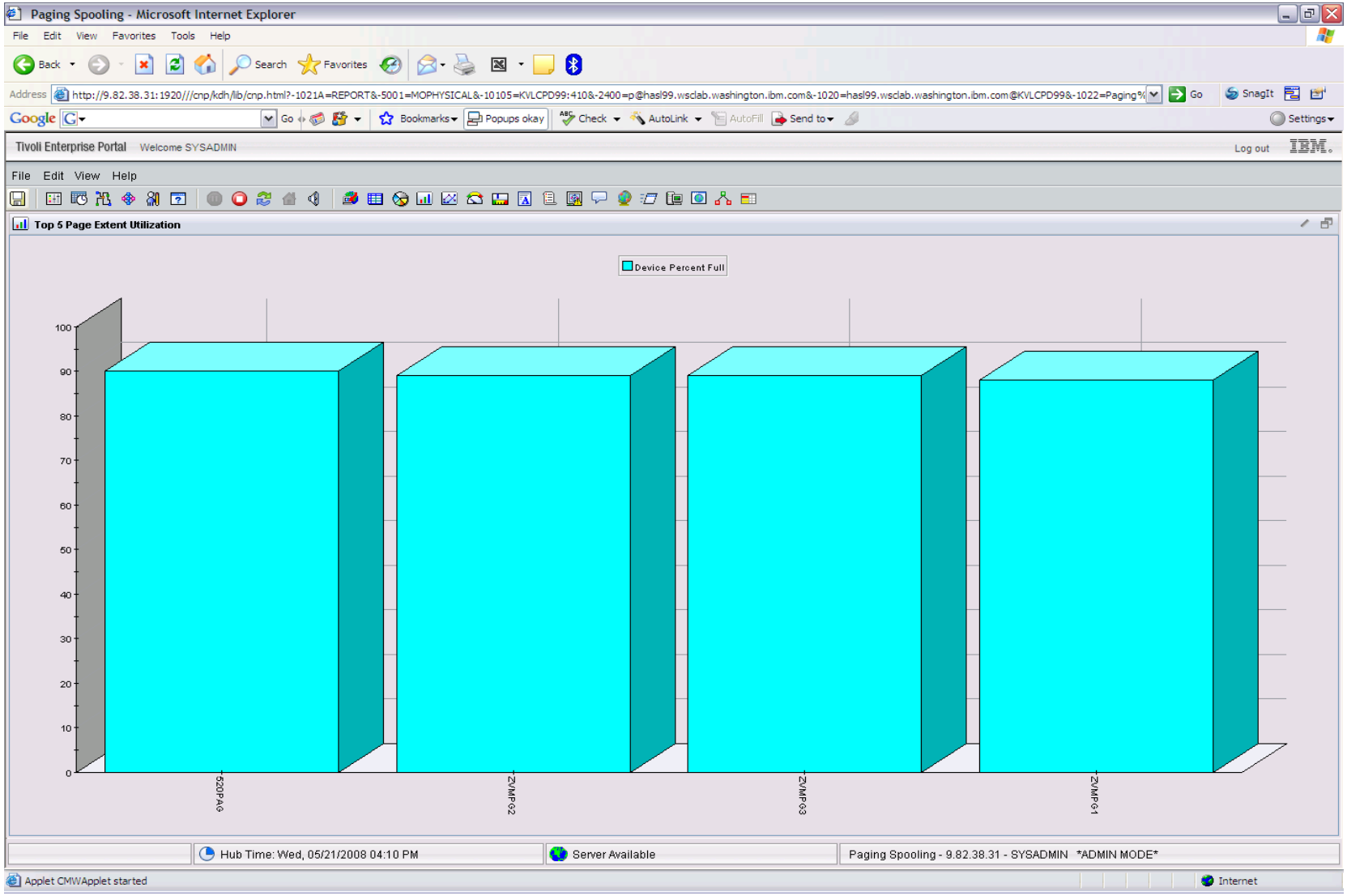
- z/VM 6.3
  - z/VM 6.3 block paging changes eliminates the benefits of 50% page space, making monitoring even more important.
    - No loss of efficiency above 50% page space utilization
    - Contiguous storage no longer needed for block paging
  - Monitoring focused on availability versus performance (avoid ABEND)
  - Early writing's goal is to keep the bottom 10% of the global aging list prewritten.
  - Whether written-on-demand or pre-written, page space is still being used. From a monitoring perspective, this is all that matters.
  - Monitor for rapid growth in page space as well as overall size thresholds.
  - The closer your monitoring threshold is to 100%, the more automation is necessary to avoid an ABEND (how quickly page space can be added).
- Calculation guidelines are located in the CP Planning and Administration Manual.

OMEGAMON CP Owned Devices – Paging Subsystem



Time	System ID	LPAR Name	Device VOLSER	Device Address	PAGING SPOOLING	Allocation	Available Slots	Device Type	Device End Extent	Device Percent Full	Device Start Extent	Device Slots Used
7/14/09 18:28:17	GDLVICOM	VIC	VICRES	1C0A	DIRECT	17	2880	3390	17	6	1	180
7/14/09 18:28:17	GDLVICOM	VIC	VI1D80	1D80	PAGING	10016	1802880	3390	10016	0	1	0
7/14/09 18:28:17	GDLVICOM	VIC	VI1D81	1D81	SPOOLING	10016	1171980	3390	10016	35	1	630900
7/14/09 18:28:17	GDLVICOM	VIC	VI1D82	1D82	T-DISK	10016	1802880	3390	10016	0	1	0

z/VM Page Attributes



Changed Behavior: Eligible List

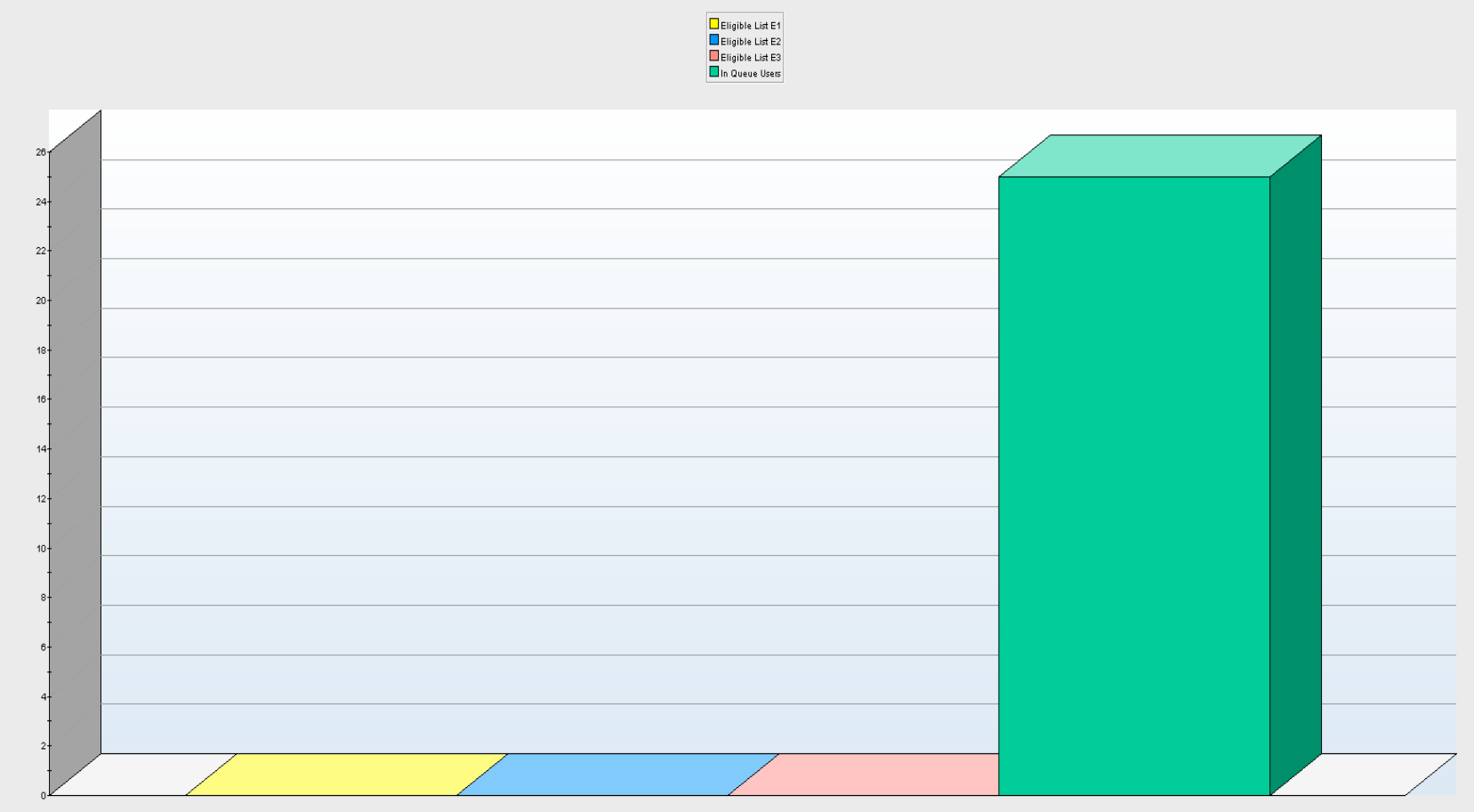
If the z/VM scheduler is holding back users because of storage constraints, we will see nonzero lengths for the eligible lists.

- One of the factors to the creation of an **eligible list** is the concept of “loading users”
  - Governed by **SET SRM LDUBUF**
  - A virtual machine is characterized as a “loading user” if its count of page faults in a dispatch slice exceeds a threshold
  - **SET SRM LDUBUF** attempts to keep the system from over-committing paging devices to the point of thrashing
- Changes in z/VM 6.3 paging algorithms can affect the number of virtual machines that are marked as “loading” users and therefore cause **eligible lists** to be formed where they had not formed prior to z/VM 6.3
  - Definition of page fault slightly different
  - Rate at which system can page fault has increased
- Recommend monitoring for eligible lists and adjusting the following as appropriate
  - **SET QUICKDSP**
  - **SET SRM LDUBUF**
- IBM is investigating improvements to avoid the unnecessary eligible list formation.

Monitor for formation of eligible lists

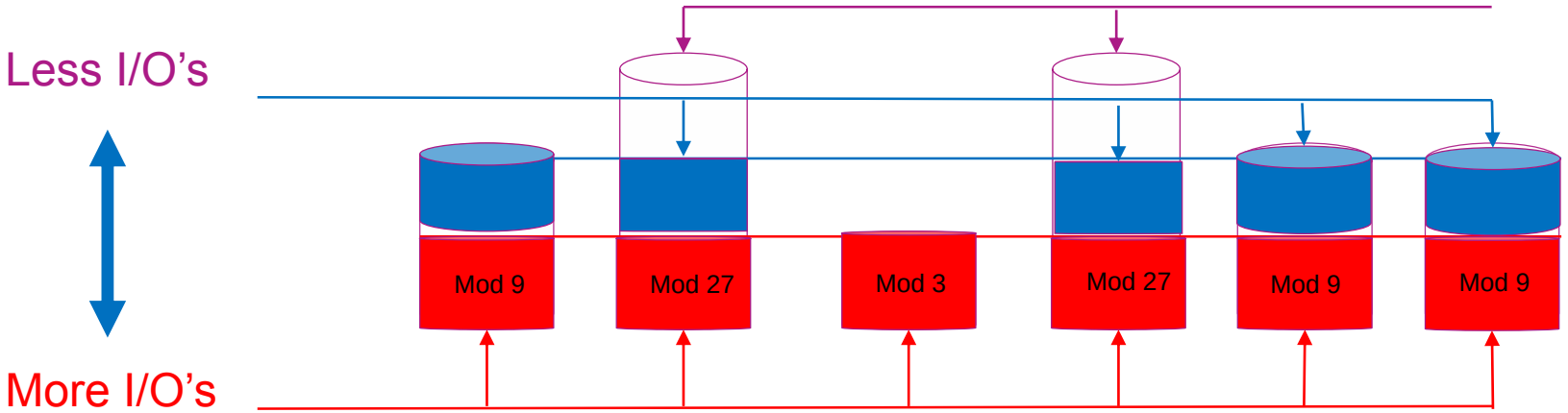
Q0 List Percent	Q1 List Percent	Q2 List Percent	Q3 List Percent	Eligible List Percent	Loading Percent
12	22	7	36	0	0
100	0	0	0	0	0
100	0	0	0	0	0
0	0	0	100	0	0
0	88	12	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

Graphical representation of eligible lists



All Versions Paging I/O

- Paging to DASD will perform better when more than one paging I/O can occur simultaneously, preventing contention between paging operations.
- Using multiple paging volumes allows for multiple paging I/O's to occur simultaneously.
- Use the same size volumes, otherwise, smaller volumes will fill up and no longer be candidates for paging I/O.
- As the number of smaller paging volumes fill, it is less likely that multiple page I/O's can occur simultaneously.



Plan a robust paging DASD configuration

- Use plenty of paging volumes
- Make the volumes all the same size
- Put only paging space on the volumes you use for paging
- Spread the paging volumes through your logical control units
- Avoid logical control units that you know are hot on application I/O
- Use plenty of CHPIDS
- Do not use ESCON CHIPIDS (only carry one I/O at a time)
- Do not mix ECKD paging and SCSI paging
- Leave reserved slots in the CP-owned list

Multiple Sizes

Mixed Utilization

09/03/14 11:21:17	ZVMV6R30	ROSPA	ZVMPG1	6B05	PAGING	3338	84240	3390-9	3338	86	1	516600
09/03/14 11:21:17	ZVMV6R30	ROSPA	ZVMPG2	6B06	PAGING	3338	84240	3390-9	3338	86	1	516600
09/03/14 11:21:17	ZVMV6R30	ROSPA	ZVMPG3	6B07	PAGING	3338	84240	3390-9	3338	86	1	516600
09/03/14 11:21:17	ZVMV6R30	ROSPA	PG6B0A	6B0A	PAGING	10017	1009800	3390-9	10016	44	0	793260
09/03/14 11:21:17	ZVMV6R30	ROSPA	ZVMPG5	6B0B	PAGING	3339	84240	3390-9	3338	86	0	516780
09/03/14 11:21:17	ZVMV6R30	ROSPA	ZVMPG6	6B0C	PAGING	3339	84240	3390-9	3338	86	0	516780
09/03/14 11:21:17	ZVMV6R30	ROSPA	ZVMPG4	6B10	PAGING	3339	84240	3390-9	3338	86	0	516780
09/03/14 11:21:17	ZVMV6R30	ROSPA	ZVMPG7	6C02	PAGING	10017	1785060	3390-9	10016	1	0	18000
09/03/14 11:21:17	ZVMV6R30	ROSPA	M01P01	6D0A	PAGING	10016	1009620	3390-9	10016	44	1	793260

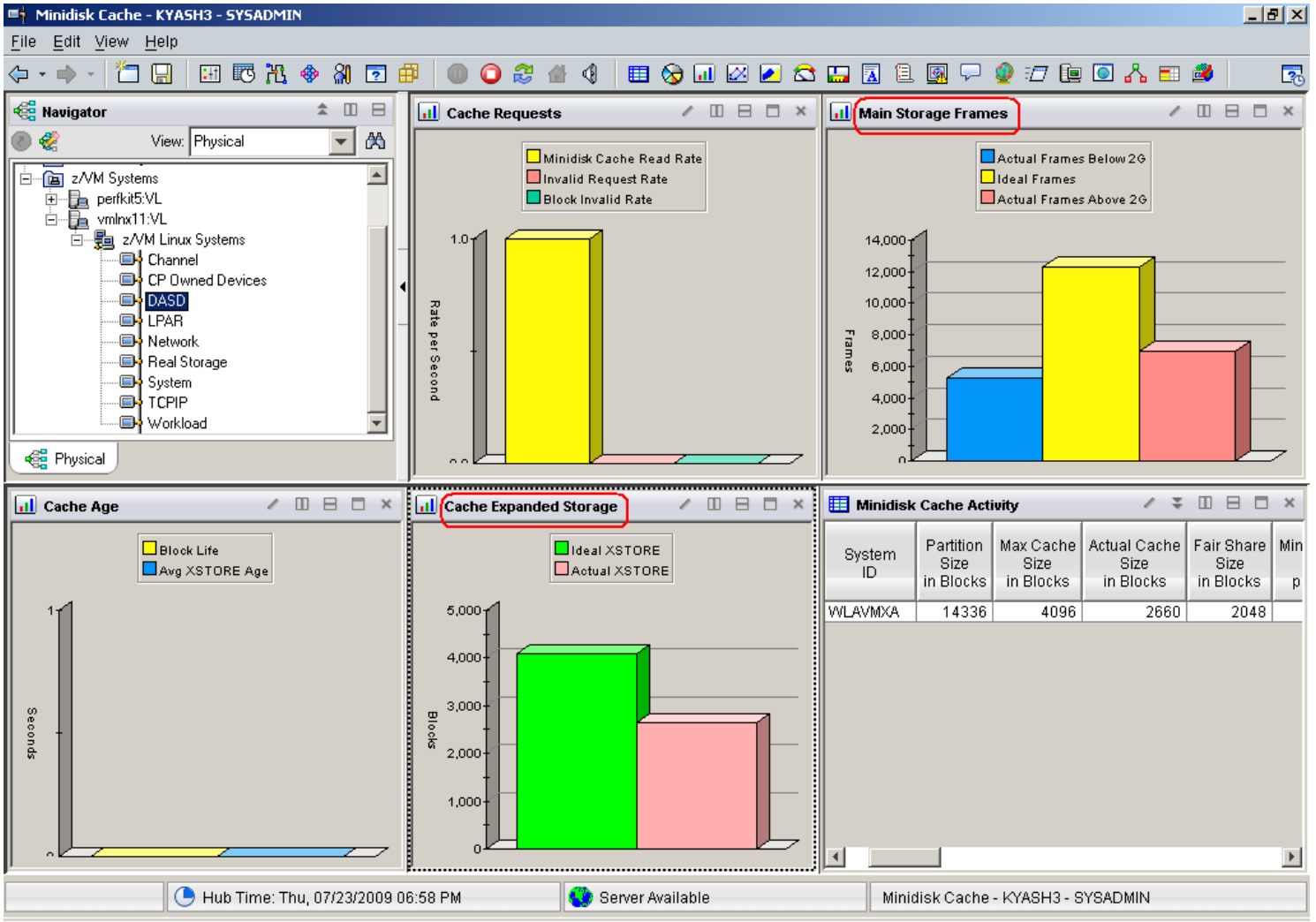
Minidisk Cache

- z/VM minidisk cache is a write-through cache:
  - Improves read I/O performance.
  - But it's not free.
- Not recommended for:
  - Memory constrained systems.
  - Linux swap file disks.
  - Flashcopy targets (see next chart)
- Default system settings are less than optimal.
- Recommended settings:
  - Eliminate MDC in expanded storage.
    - **SET MDC XSTORE 0M 0M**
  - Limit MDC in central storage – 10% is a good starting point.
    - **SET MDC STORE 0M 256M**
  - Monitor with product like OMEGAMON XE and/or the Q MDC command.

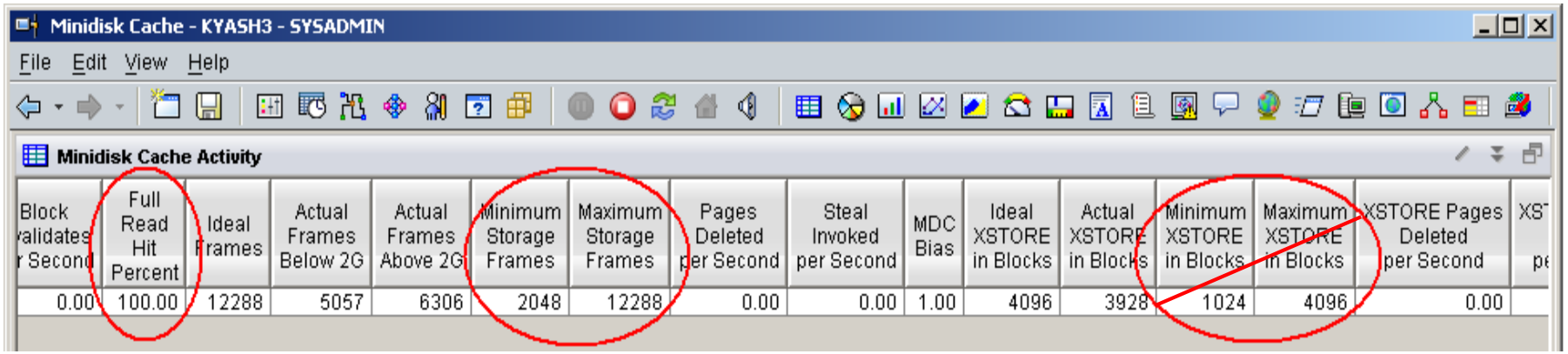
MDC and FlashCopy Interaction

- FlashCopy requests require z/VM to flush MDC for the entire minidisk.
- MDC Flush processing is very expensive even when there is no data in MDC to flush
  - System Time becomes very high.
- z/OS DFSMS and other utilities can make extensive use of FlashCopy for functions such as defragmentation
- Mitigations
  - Turn off MDC for minidisks that are FlashCopy targets

OMEGAMON MDISK Cache Allocations



OMEGAMON MDISK Cache Allocations – p. 2

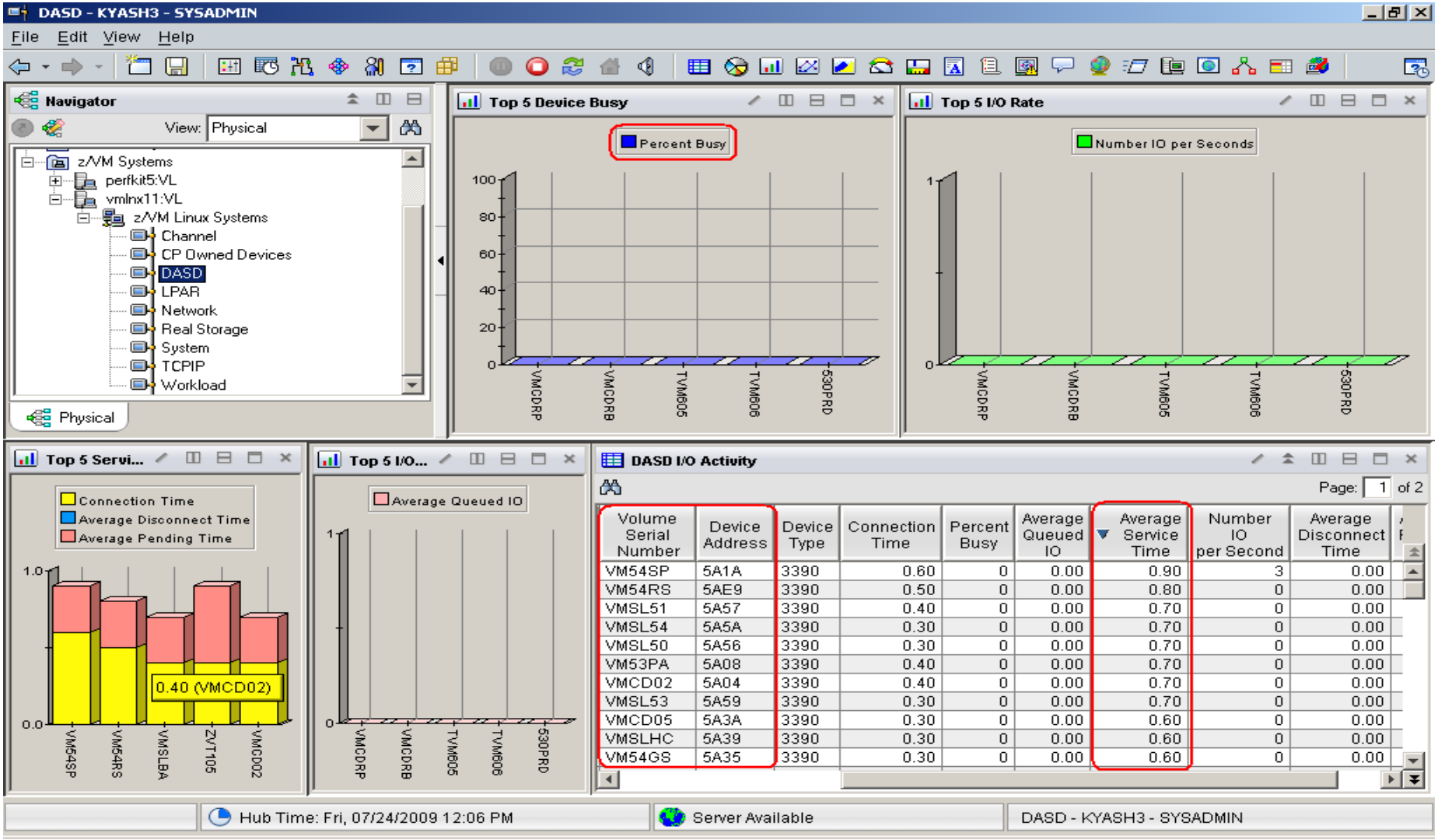


Block validates r Second	Full Read Hit Percent	Ideal Frames	Actual Frames Below 2G	Actual Frames Above 2G	Minimum Storage Frames	Maximum Storage Frames	Pages Deleted per Second	Steal Invoked per Second	MDC Bias	Ideal XSTORE in Blocks	Actual XSTORE in Blocks	Minimum XSTORE in Blocks	Maximum XSTORE in Blocks	XSTORE Pages Deleted per Second	XST pe
0.00	100.00	12288	5057	6306	2048	12288	0.00	0.00	1.00	4096	3928	1024	4096	0.00	

Direct Access Storage Devices (DASD)

- **Avg Pending Time for DASD**
  - Average pending time for real DASD I/Os. RoT: Should be less than 1 millisecond.
- Items worth keeping an eye on:
  - **Number of I/O's per Second, Percent Busy**
  - **Avg Service Time** Average service time for real DASD devices (sum of the pending, connect, and disconnect times).
  - **DASD I/O Rate** Rate of traditional real I/Os per second to real DASD devices. Worth monitoring.

DASD I/O Workspace



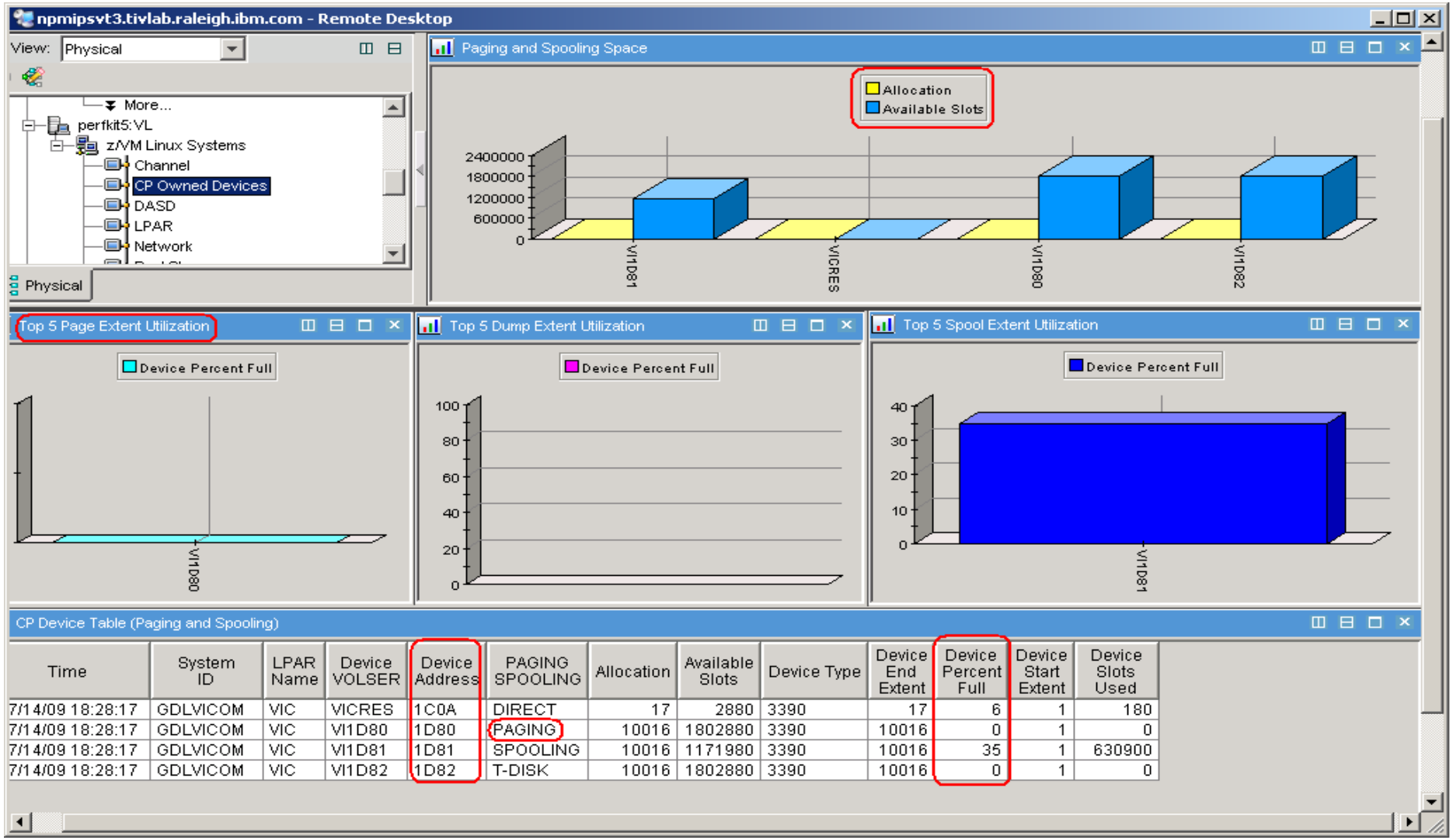
System Dump & Spool Space

- Dump Space
  - Ensure there is sufficient dump space defined to the system.
  - Dump space requirements vary according to memory usage.
    - Q DUMP – identifies allocated dump space.
    - Calculation guidelines are located in CP Planning and Administration Manual.
- Spool Space
  - Various uses:
    - User printer, punch, reader files (console logs)
    - DCSS, NSS
    - System files
    - Page space overflow
  - Spool Management:
    - Monitor with OMEGAMON, Operations Manager, Q ALLOC SPOOL cmd
    - SFPURGER utility:
      - Rule based tool to clean up spool space.
      - Included in the no charge CMS Utilities Feature (CUF).

VMDUMP Processing Concern

- VMDUMP is a very helpful command for problem determination.
- Some weaknesses:
  - Does not scale well, can take up to 40 minutes per GB.
  - It is not interruptible
- Linux provides a disk dump utility which is much faster relative to VMDUMP.
  - It is disruptive
  - Does not include segments outside the normal virtual machine.
- See **http://www.vm.ibm.com/perf/tips/vmdump.html**
- If you increase central, make sure you also increase dump space
  - More guidance available on www.vm.ibm.com/techinfo/
  - Download updated *"Allocating Space for CP Hard ABEND Dumps"*

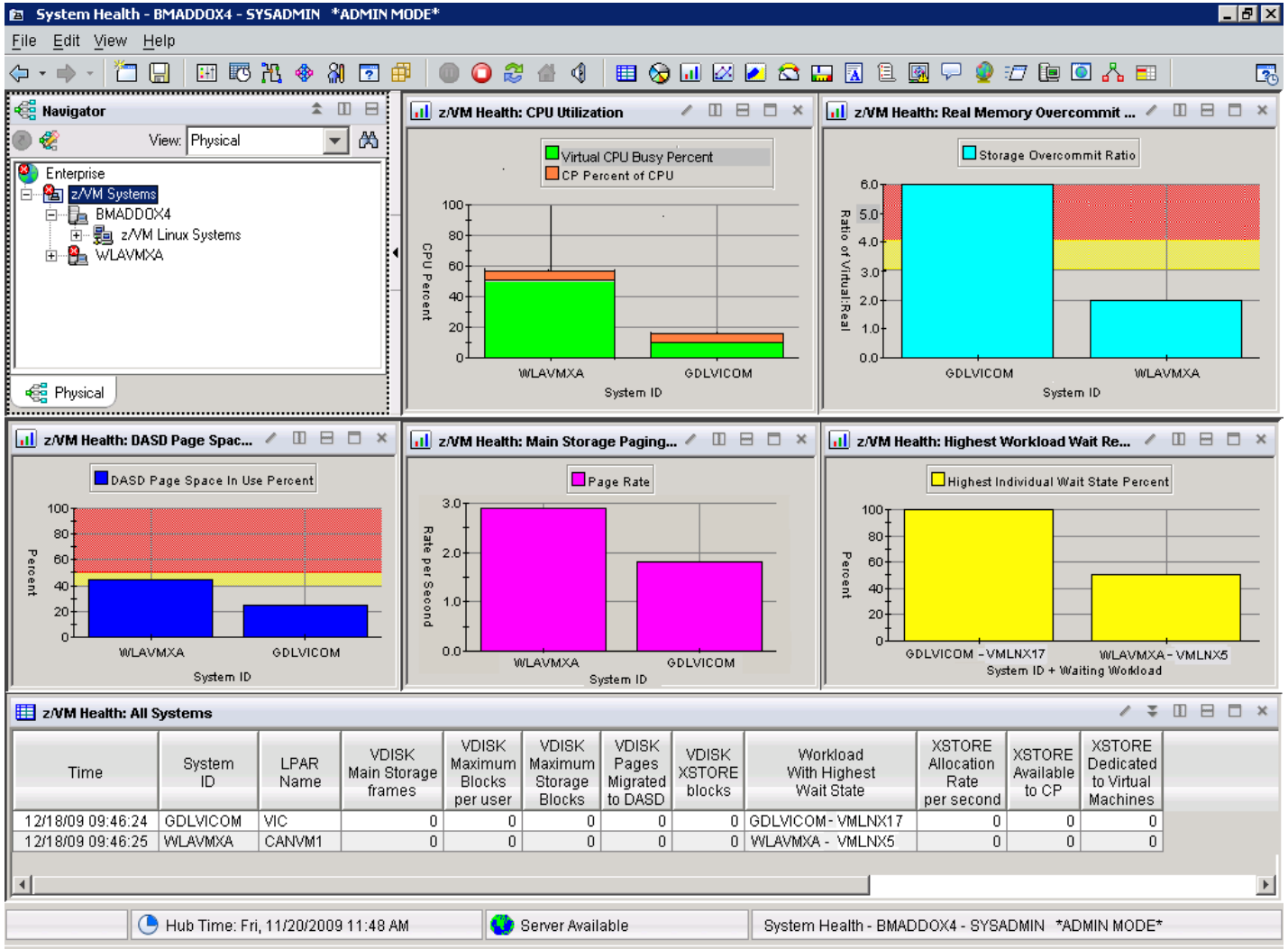
System Dump & Spool Space



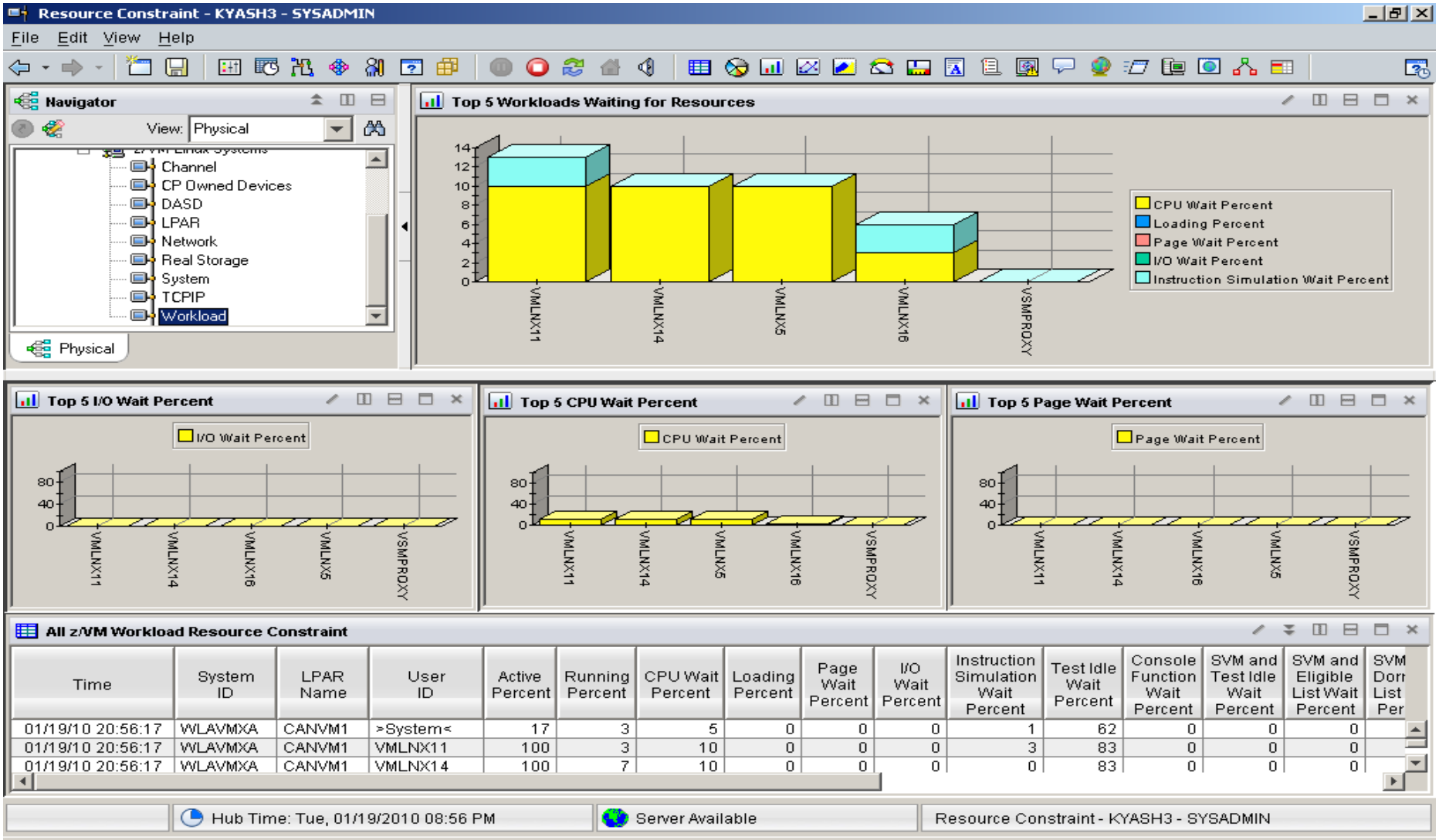
Time	System ID	LPAR Name	Device VOLSER	Device Address	PAGING SPOOLING	Allocation	Available Slots	Device Type	Device End Extent	Device Percent Full	Device Start Extent	Device Slots Used
7/14/09 18:28:17	GDLVICOM	VIC	VICRES	1C0A	DIRECT	17	2880	3390	17	6	1	180
7/14/09 18:28:17	GDLVICOM	VIC	VI1D80	1D80	PAGING	10016	1802880	3390	10016	0	1	0
7/14/09 18:28:17	GDLVICOM	VIC	VI1D81	1D81	SPOOLING	10016	1171980	3390	10016	35	1	630900
7/14/09 18:28:17	GDLVICOM	VIC	VI1D82	1D82	T-DISK	10016	1802880	3390	10016	0	1	0

Tips—Overall Health of Your System

At a quick glance you can see the %CPU usage, what your overcommit ratio is, the number of users in a wait state, and paging rates of all your z/VM systems



Resource Constraint Analysis (Waits)



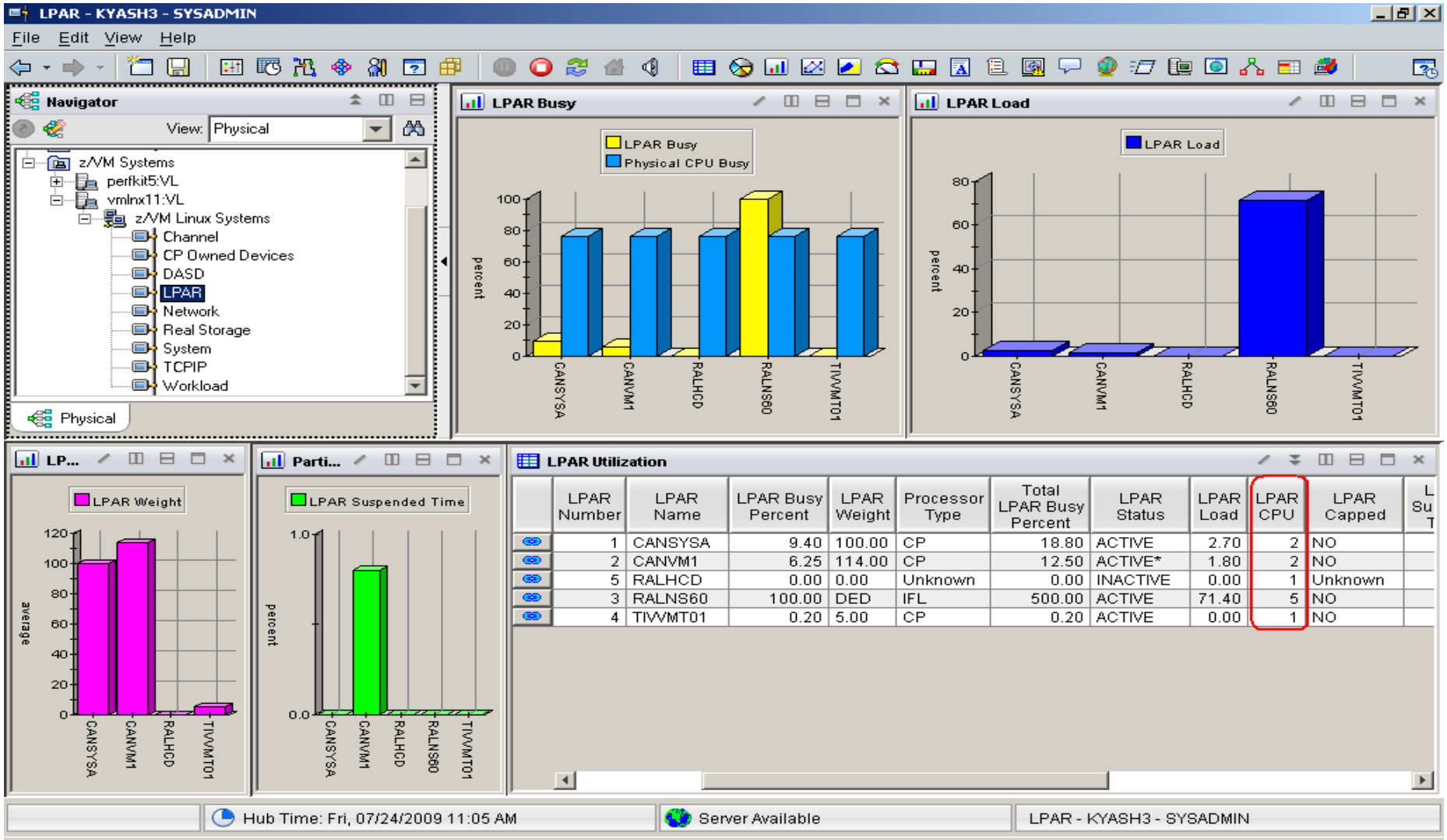
Do not ignore the hardware!

- Just because Linux resources are virtual, do not ignore the hardware!
  - Hardware is another potential layer of shared resources.
  - LPAR weight, CPU sharing, LPAR load, and other attributes need to be monitored for overall system performance.
  - The measurement should include the entire CEC and not just the LPAR hosting z/VM.

Processors

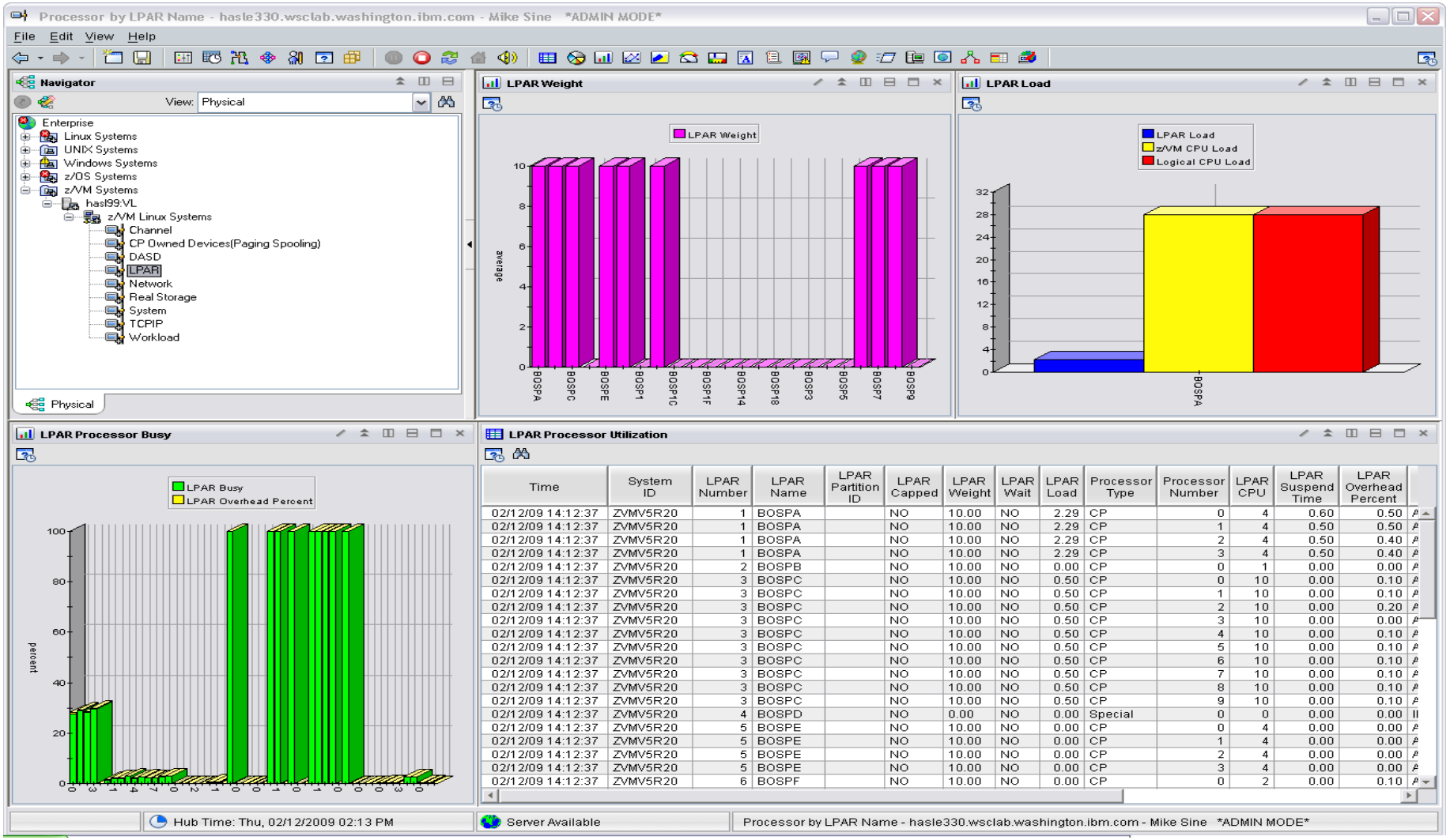
- Logical Processors
  - LPAR recommendation – no greater than a 4:1 logical to real ratio.
  - z/VM 5.1 - z/VM 5.2 support up to 24 processors.
  - z/VM 5.3 - z/VM 6.x support up to 32 processors.

LPAR Utilization Workspace

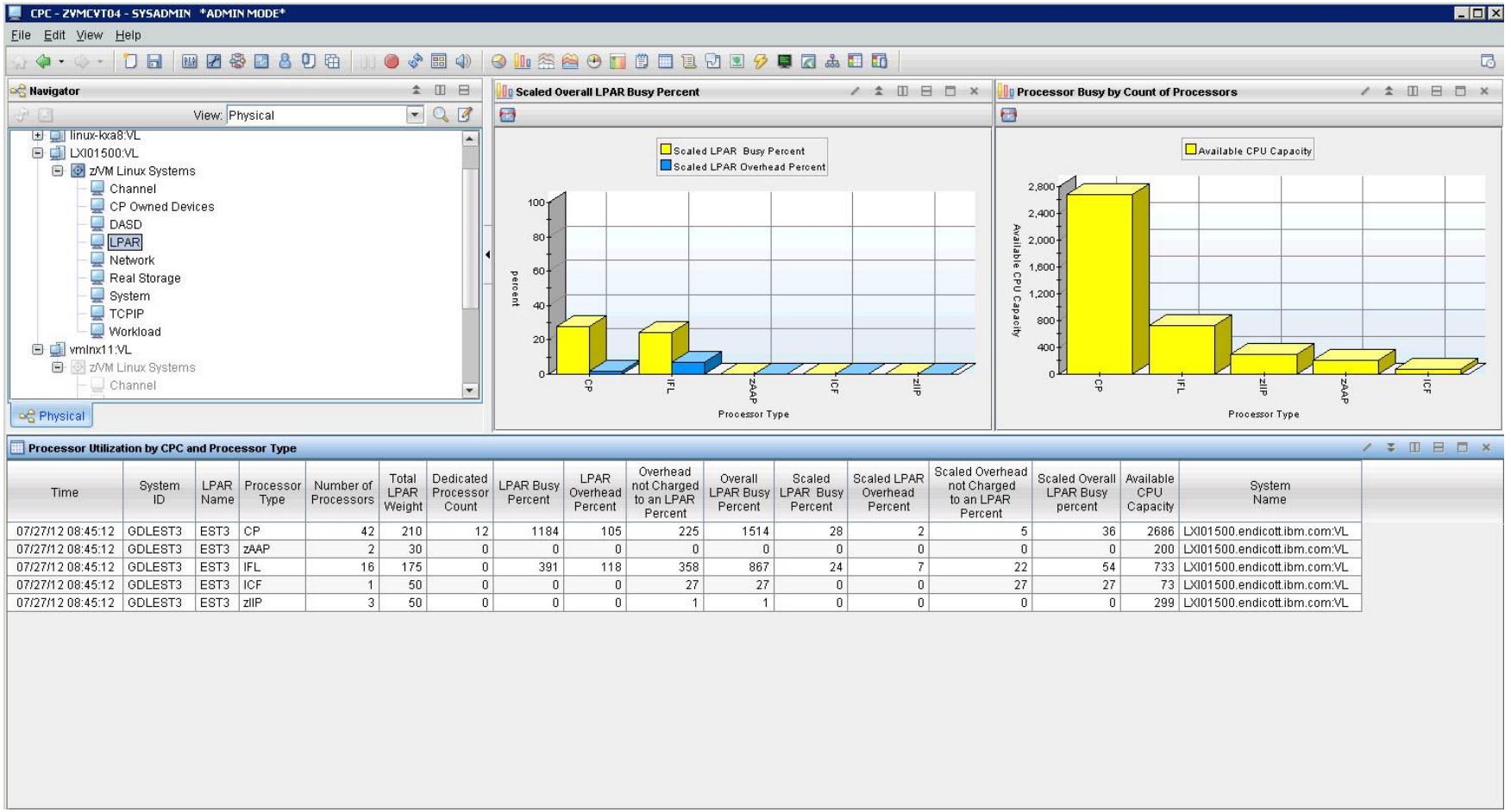


LPAR Number	LPAR Name	LPAR Busy Percent	LPAR Weight	Processor Type	Total LPAR Busy Percent	LPAR Status	LPAR Load	LPAR CPU	LPAR Capped
1	CANSYSA	9.40	100.00	CP	18.80	ACTIVE	2.70	2	NO
2	CANVM1	6.25	114.00	CP	12.50	ACTIVE*	1.80	2	NO
5	RALHCD	0.00	0.00	Unknown	0.00	INACTIVE	0.00	1	Unknown
3	RALNS60	100.00	DED	IFL	500.00	ACTIVE	71.40	5	NO
4	TIVVMT01	0.20	5.00	CP	0.20	ACTIVE	0.00	1	NO

Processor by LPAR name workspace



CPC workspace

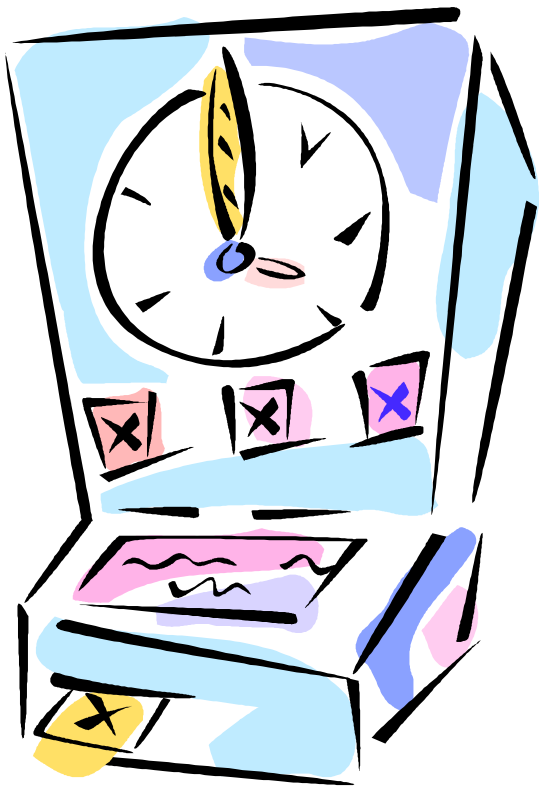


Time	System ID	LPAR Name	Processor Type	Number of Processors	Total LPAR Weight	Dedicated Processor Count	LPAR Busy Percent	LPAR Overhead Percent	Overhead not Charged to an LPAR Percent	Overall LPAR Busy Percent	Scaled LPAR Busy Percent	Scaled LPAR Overhead Percent	Scaled Overhead not Charged to an LPAR Percent	Scaled Overall LPAR Busy percent	Available CPU Capacity	System Name
07/27/12 08:45:12	GDLEST3	EST3	CP	42	210	12	1184	105	225	1514	28	2	5	36	2686	LXI01500.endicott.ibm.com:VL
07/27/12 08:45:12	GDLEST3	EST3	zAAP	2	30	0	0	0	0	0	0	0	0	0	200	LXI01500.endicott.ibm.com:VL
07/27/12 08:45:12	GDLEST3	EST3	IFL	16	175	0	391	118	358	867	24	7	22	54	733	LXI01500.endicott.ibm.com:VL
07/27/12 08:45:12	GDLEST3	EST3	ICF	1	50	0	0	0	27	27	0	0	27	27	73	LXI01500.endicott.ibm.com:VL
07/27/12 08:45:12	GDLEST3	EST3	zIIP	3	50	0	0	0	1	1	0	0	0	0	299	LXI01500.endicott.ibm.com:VL

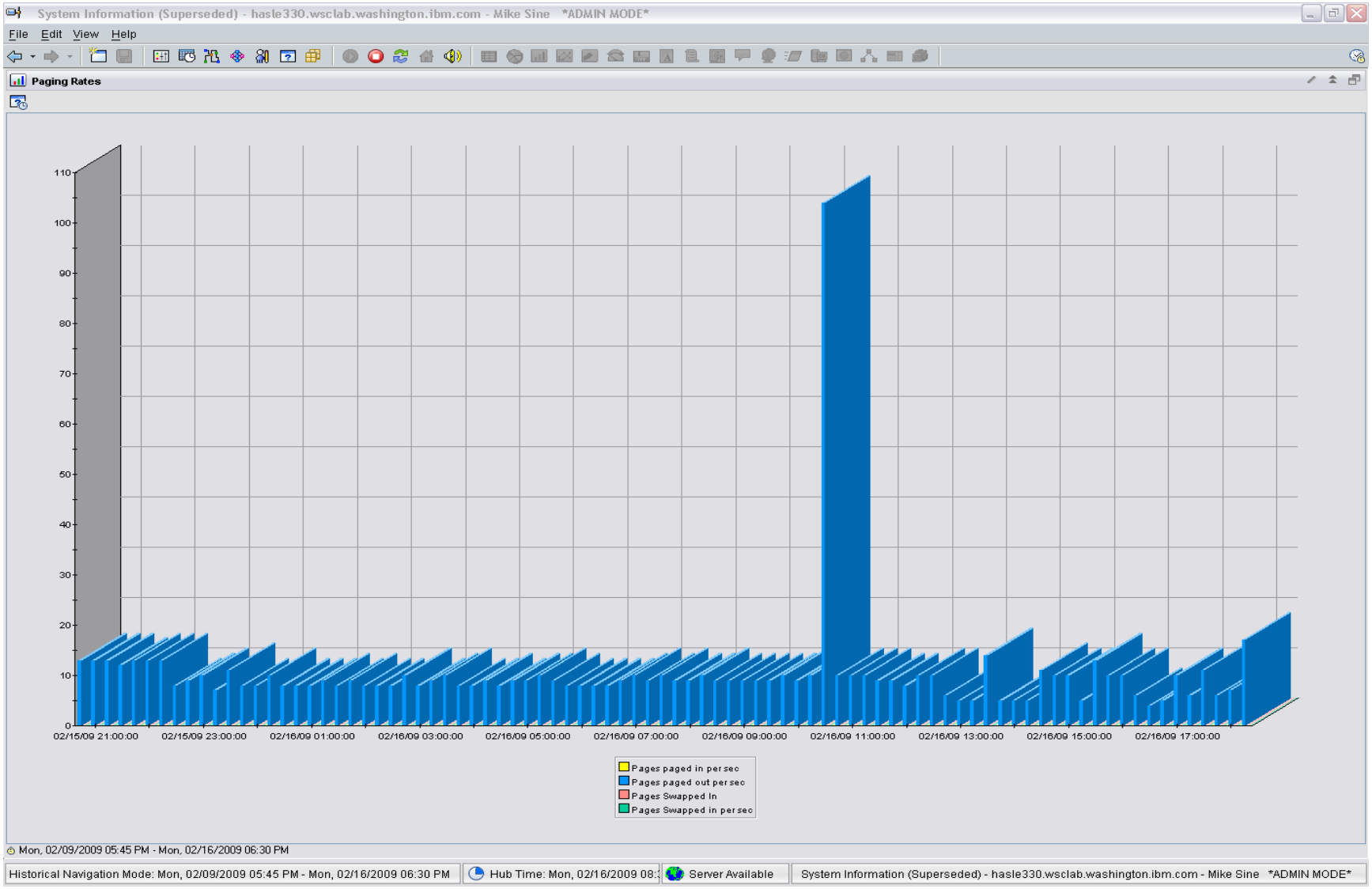
History On-Demand with Persistent Historical Views

This makes it easier to see anomalies, or match spikes. Capturing performance data as a base line is a must:

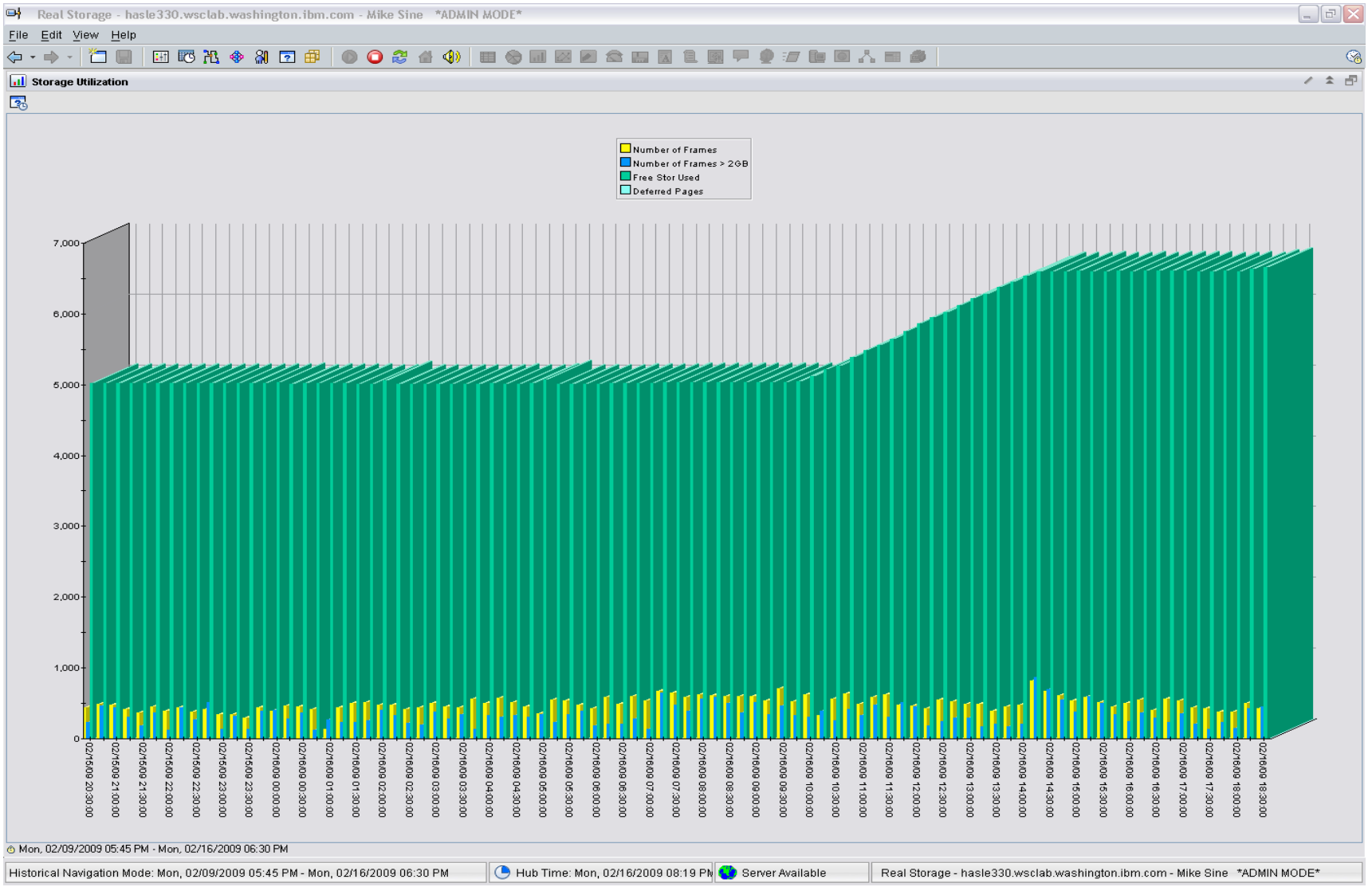
- General history data – business as usual.
- Detailed raw monitor data prior to and following any major changes.
- Ability to review attributes of a past incident through the enterprise view!
- On-Demand through the Portal or Batch



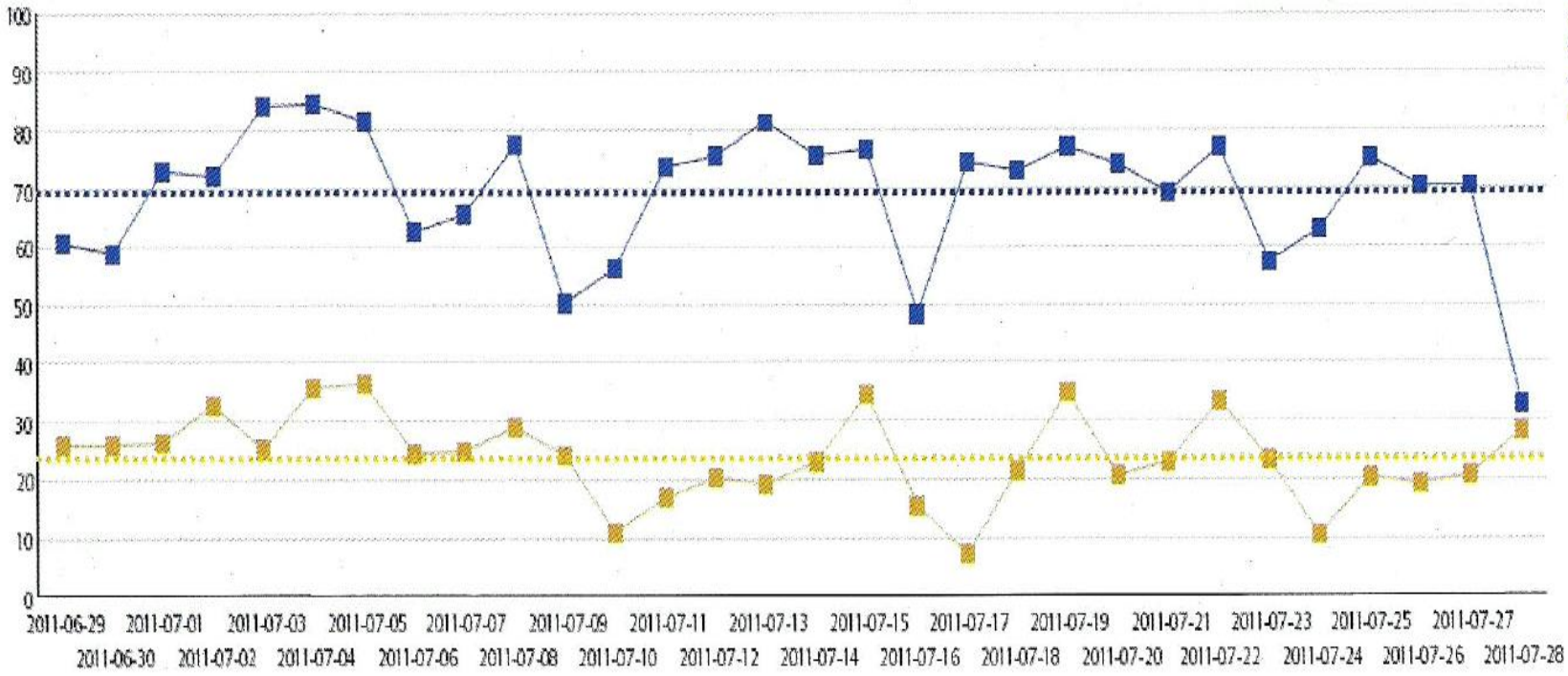
On-Demand: Persistent Historical Views



On-Demand: Persistent Historical Views



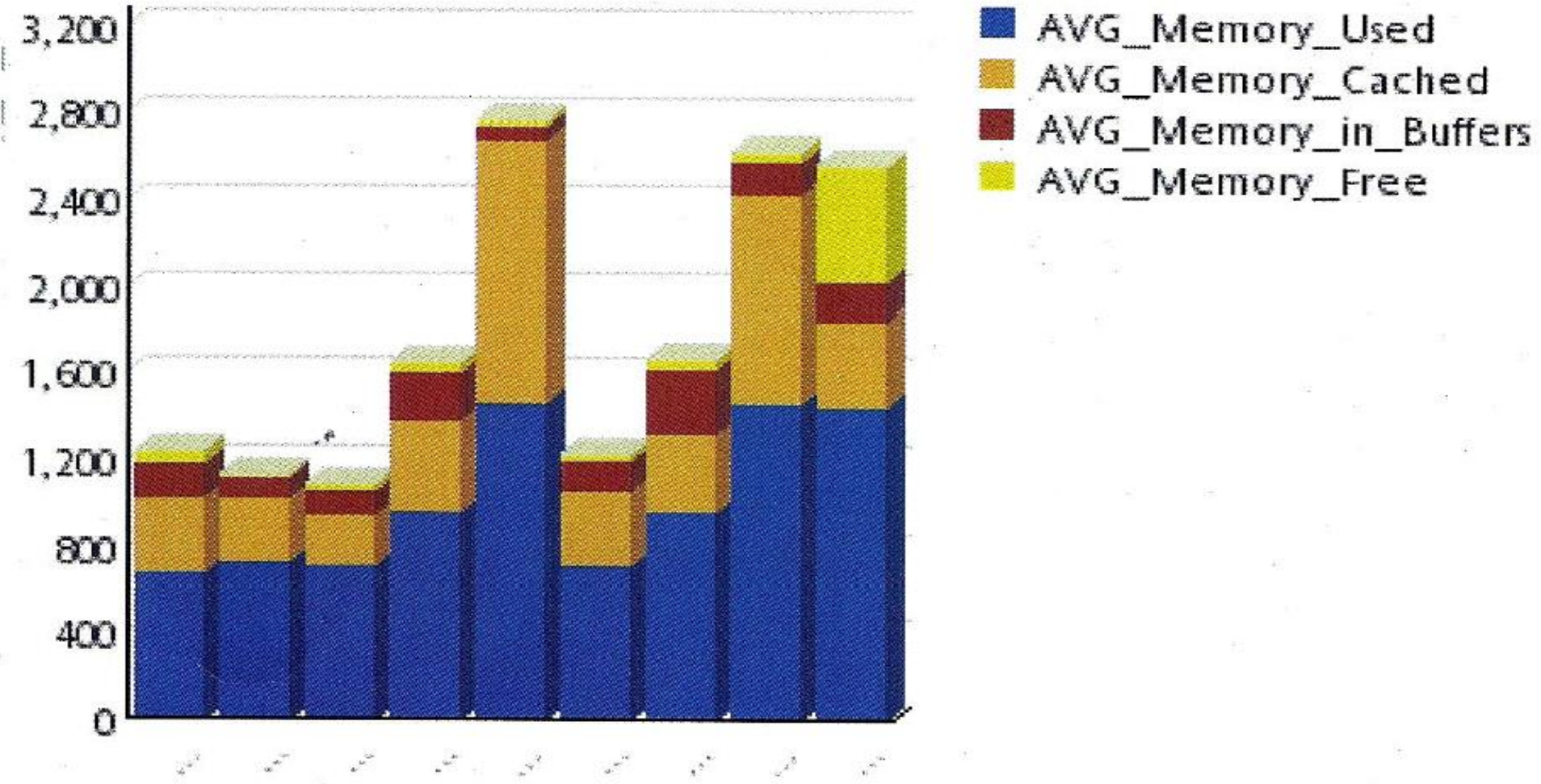
Max and Avg CPU example:



Legend:

Max_CPU_Percent:	Maximum CPU for the day as a percent of the number of virtual CPUs
Avg_CPU_Percent:	Average CPU for the day as a percent of virtual CPUs
Mean of the Maximum:	30 day average for Maximum CPU percentages
Mean of the Averages:	30 day average for the average CPU percentages
AVG_Main_Memory_Util:	Average main memory utilization for the day as a percent
AVG_Cache_Used:	Average size of memory used to cache buffers in megabytes
AVG_Page_Alloc_Rate:	Average number of pages obtained from available list in 4 kilobyte pages per second
AVG_Swap_Used:	The percent of swap space used.

Avg Linux Memory breakdown example:



Tivoli Common Reporting (TCR)

- TCR reports available on the OPAL website
  - http://www-18.lotus.com/wps/portal/topal
- What is TCR?
  - Tivoli Common Reporting.
  - Consistent approach to viewing and administering reports.
  - Cognos based.
  - Flexible development environment (Eclipse based) for creating report definitions.
  - Five templates provided for download.
  - Taking suggestions for more

Sample Reports Available

- z/VM VM System CPU Utilization
- z/VM VM System Paging Utilization
- z/VM Linux System CPU Utilization
- z/VM VM System CP-Owned Device Utilization
- z/VM VM System TCP Server Statistics

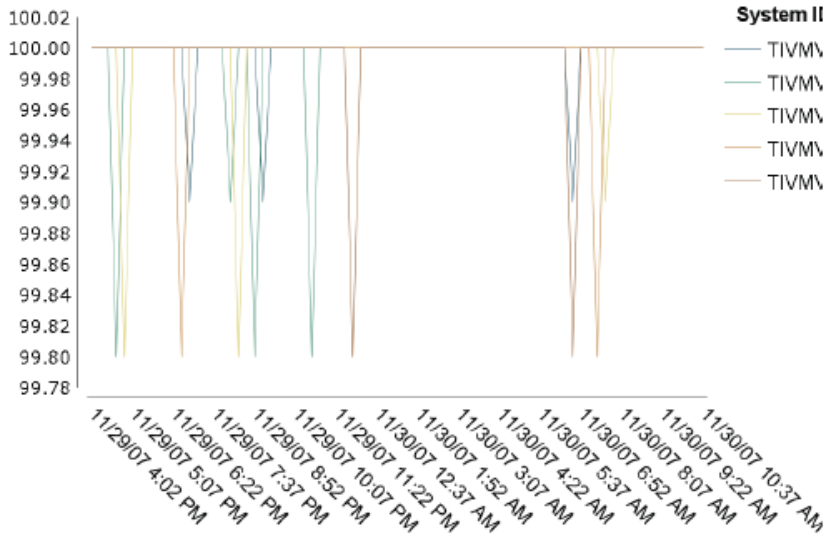
Tivoli

IBM

z/VM System CPU Utilization

Report Period	**All**	**Significant Resources Selected**	**5**
Start Date	**Dec 31, 1969 12:00 AM**	**End Date**	**Nov 30, 2007 11:59 PM**
System ID	**All**	**LPAR Name**	**All**

LPAR Busy



Available Summarization Time Periods:

Hourly
Daily
Weekly
Monthly
Not Summarized Data

System = TIVMVS6

LPAR Name	LPAR Busy	LPAR Load	LPAR Suspend Time	LPAR Overhead Time	Date/Time

November 30, 2007 2:26:24 PM EST 1 / 18

Tivoli

IBM

System = TIVMVS6

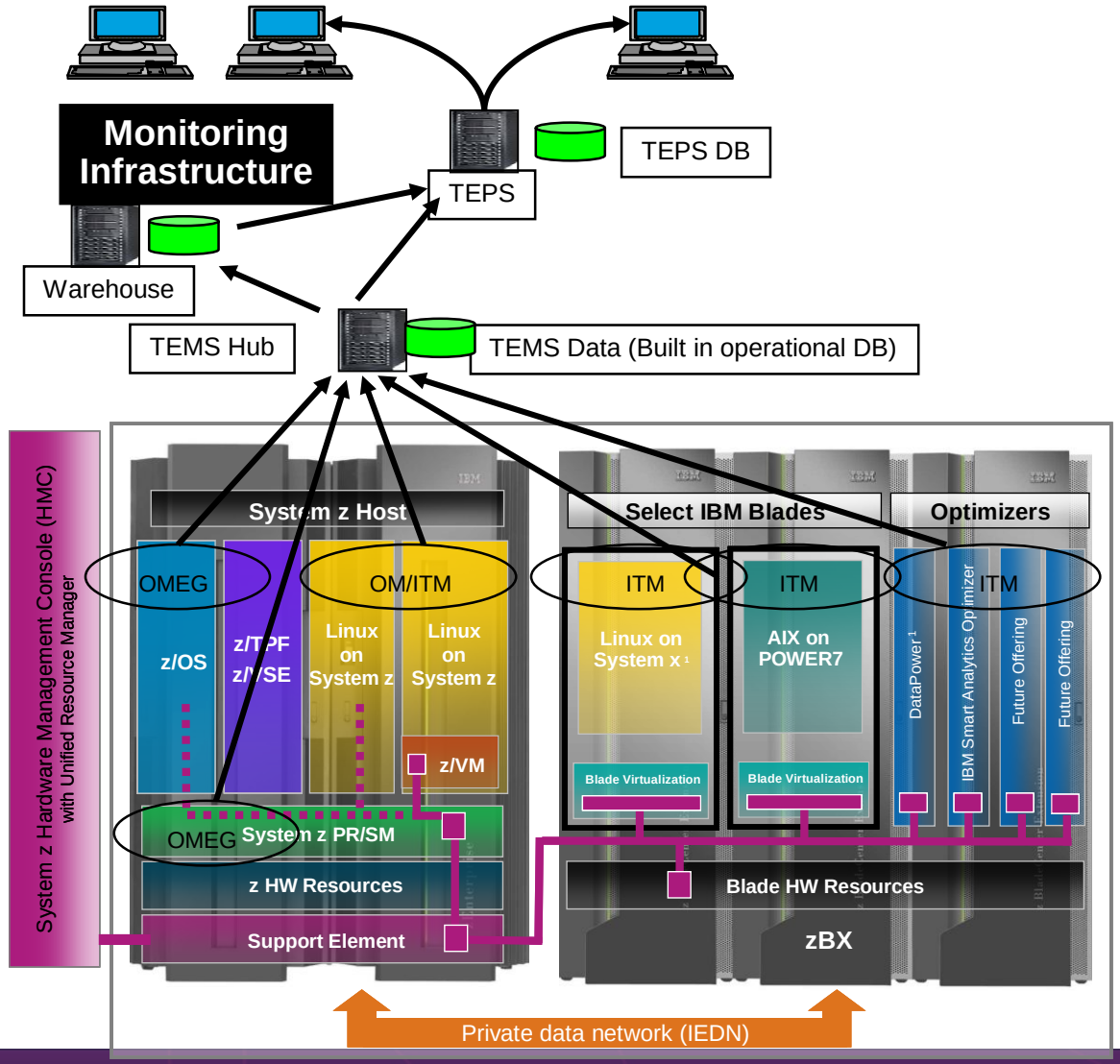
LPAR Name	LPAR Busy	LPAR Load	LPAR Suspend Time	LPAR Overhead Time	Date/Time
RALNS31	100	4.2	0	.6	Nov 29, 2007 4:02 PM
RALNS32	100	4.2	0	.6	Nov 29, 2007 4:02 PM
RALNS61	100	4.2	0	.6	Nov 29, 2007 4:02 PM
TIVMVS1	100	2.09	0	.6	Nov 29, 2007 4:02 PM
TIVMVS10	100	2.09	0	.6	Nov 29, 2007 4:02 PM
RALNS31	100	4.2	0	.6	Nov 29, 2007 4:08 PM
RALNS32	100	4.2	0	.6	Nov 29, 2007 4:08 PM
RALNS61	100	4.2	0	.6	Nov 29, 2007 4:08 PM
TIVMVS1	100	2.09	0	.6	Nov 29, 2007 4:08 PM
TIVMVS10	100	2.09	0	.6	Nov 29, 2007 4:08 PM
RALNS31	100	4.2	0	.6	Nov 29, 2007 4:22 PM
RALNS32	100	4.2	0	.6	Nov 29, 2007 4:22 PM
RALNS61	100	4.2	0	.6	Nov 29, 2007 4:22 PM
TIVMVS1	100	2.09	0	.6	Nov 29, 2007 4:22 PM
TIVMVS10	100	2.09	0	.6	Nov 29, 2007 4:22 PM
RALNS31	100	4.2	0	.6	Nov 29, 2007 4:37 PM
RALNS61	100	4.2	0	.6	Nov 29, 2007 4:37 PM
TIVMVS1	100	2.09	0	.6	Nov 29, 2007 4:37 PM
TIVMVS10	100	2.09	0	.6	Nov 29, 2007 4:37 PM
RALNS32	99.8	4.2	0	.6	Nov 29, 2007 4:37 PM
RALNS31	100	4.2	0	.6	Nov 29, 2007 4:52 PM
RALNS32	100	4.2	0	.6	Nov 29, 2007 4:52 PM
TIVMVS1	100	2.09	0	.6	Nov 29, 2007 4:52 PM
TIVMVS10	100	2.09	0	.6	Nov 29, 2007 4:52

November 30, 2007 2:26:51 PM EST 2 / 18

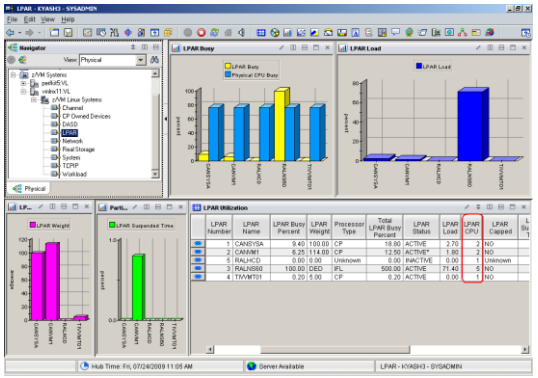
AGENDA

- Introduction
- Monitoring Requirements
  - Virtual Linux and z/VM performance considerations
  - Don’t forget the hardware
  - Integration from hardware – systems – applications Persistent historical views
- Enterprise Management

Business views across the zEnterprise



- ITM Infrastructure is shown separate to highlight components, however, each of these ITM components can reside on the zEnterprise.
- OMEGAMON agents can monitor z/OS system and subsystems, z/VM system and LPAR components, and Linux on z.
- ITM agents can monitor Linux on System z, Linux on System x, and AIX on Power7, and supported applications and databases.
- ITM monitoring of Optimizers



Common Interface across z Systems

धन्यवाद

Hindi

多謝

Traditional Chinese

감사합니다

Korean

Спасибо

Russian

Gracias

Spanish

Thank You

English

شكرا

Arabic

Obrigado

Brazilian Portuguese

Danke

German

Grazie

Italian

多谢

Simplified Chinese

Merci

French

நன்றி

Tamil

ありがとうございました

Japanese

ขอบคุณ

Thai