Managing z/VM and Linux Performance Best Practices

Jim Newell
IBM

August 15, 2013
Session Number 13498

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
- Best Practices Monitoring Requirements
  - Virtual Linux and z/VM performance considerations
  - Don’t forget the hardware
  - Integration from hardware – systems – applications Persistent historical views
- Integrated Monitoring Approach
- Linux on z Health Checker

Abstract

- In today's virtualized environments it's important that we adhere to a set of best practices when it comes to managing the environment. Even though our applications may all run within the same physical environment many of the challenges faced managing an application stack spread across multiple servers still exist.

- Furthermore, there are unique challenges associated with z/VM and Linux environments for less experienced users.

- This presentation highlights the Performance and Availability management best practices for z/VM and Linux on System z while showing how OMEGAMON XE on z/VM and Linux can be used to measure for deviations from those best practices. Regardless of the systems management tools that you are using in your installation, the information in this presentation should apply to those tools.

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
- Best Practices Monitoring Requirements
  - Virtual Linux and z/VM performance considerations
  - Don’t forget the hardware
  - Integration from hardware – systems – applications Persistent historical views
- Integrated Monitoring Approach
- Linux on z Health Checker

“Best Practices”

- z/VM
  - System Scope items
    - *Maintenance*
    - *Memory*
    - *Paging*
    - *DASD*
    - *VDISK*
    - *Processors/LPAR*
    - *System Utilization*
    - *DASD I/O*
    - *Spool*
    - *Workloads: Virtual Processors, Paging*

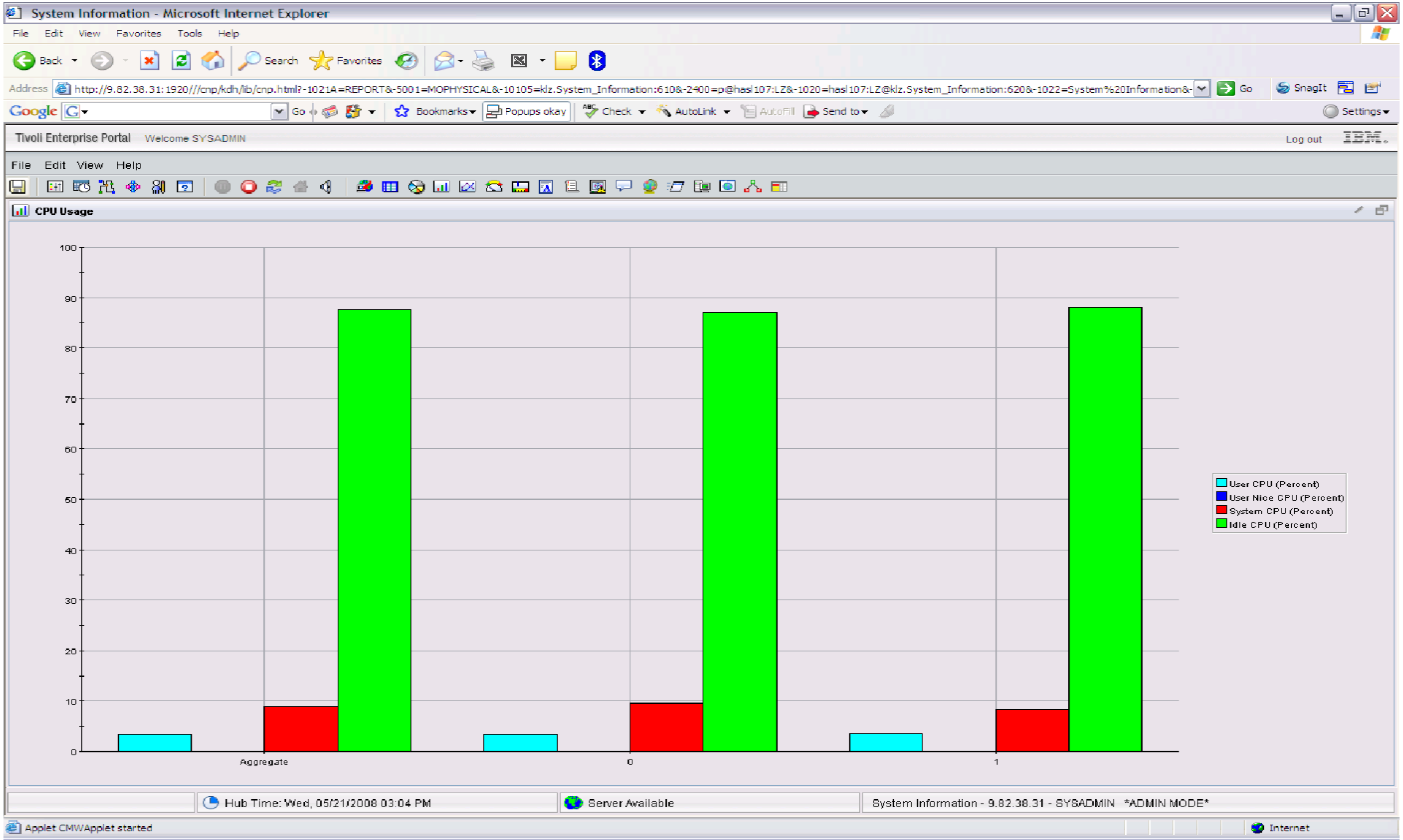
Maintenance Levels

- Recommend maintaining current service levels.
- Apply latest Recommended Service Upgrade (RSU):
  - z/VM Family
    - Released every 3-6 months
    - Contains cumulative service including all pre- and co-requisites in a pre-built format.
  - Includes service for all integrated components and the following pre-installed program products:
    - DirMaint, VM/RACF, Performance ToolKit
      - *Available on tape, DVD, or electronically.*

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.

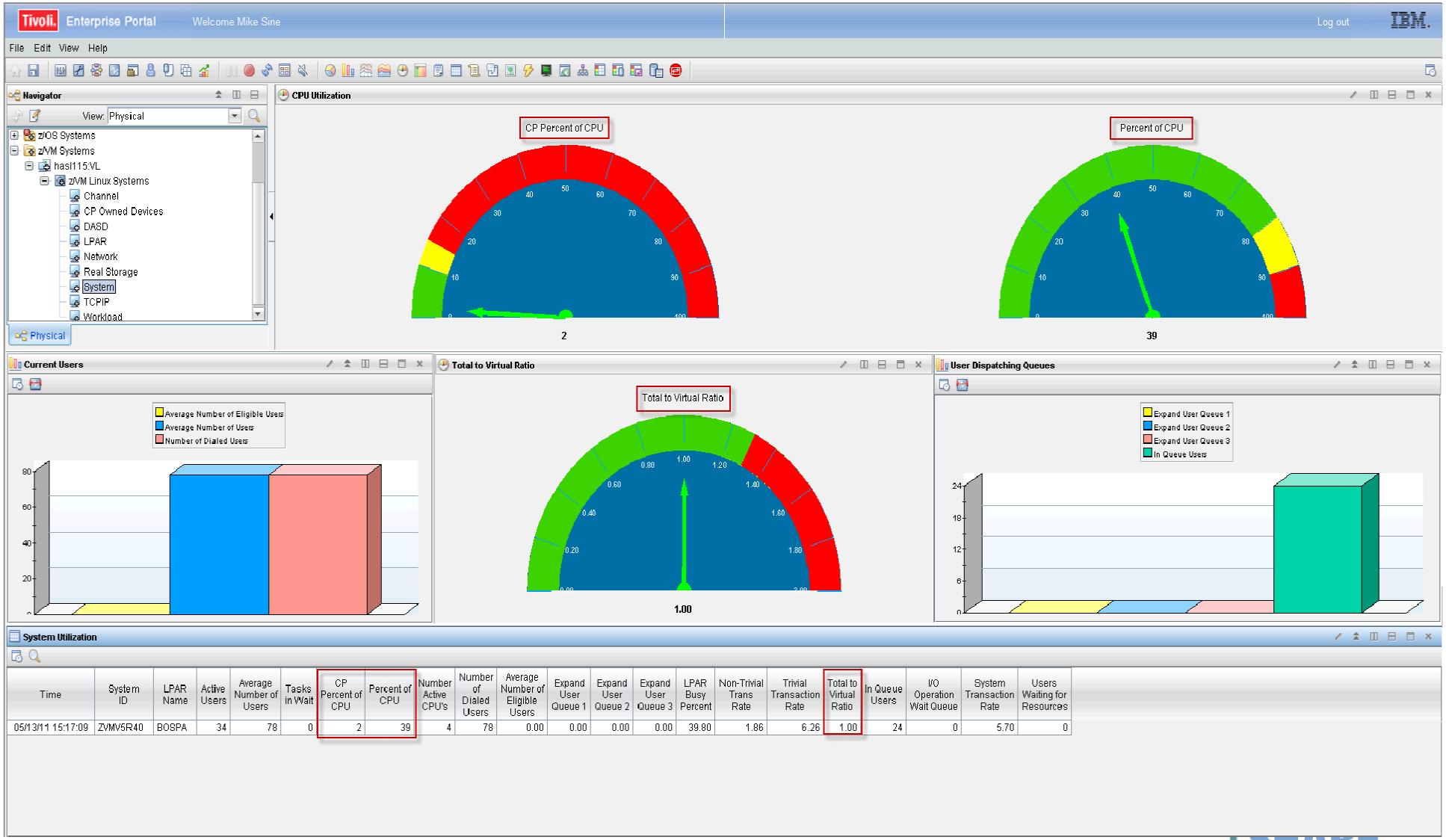
Aggregate monitoring of Virtual CPUs



z/VM Processor Utilization

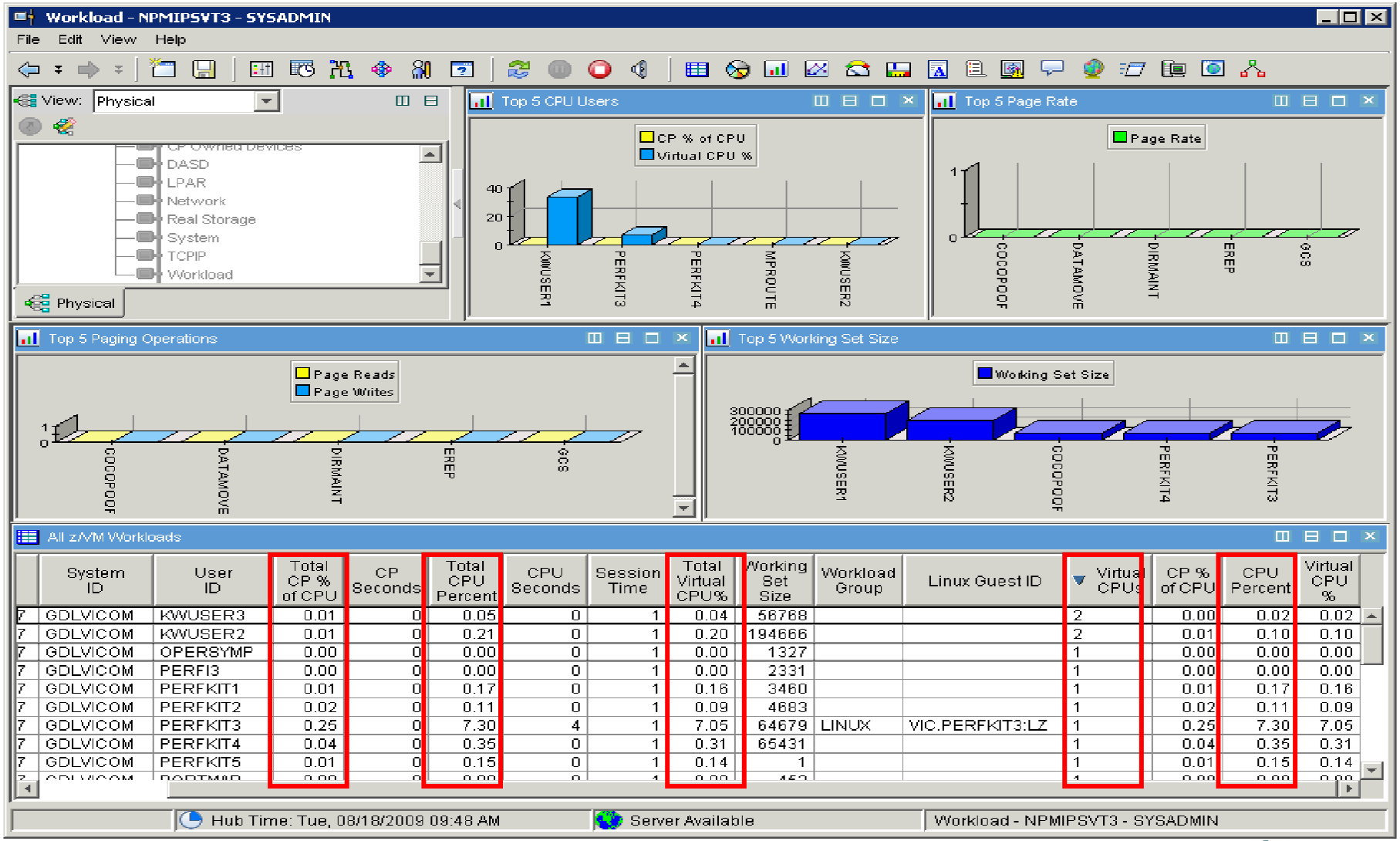
- **Total Processor Utilization** This is the processor utilization from the VM perspective and includes CP, VM System, and Virtual CPU time.
- **System Time:** This is the processor time used by the VM control program for system functions that are not directly related to any one virtual machine. This 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	ZVMVSR40	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

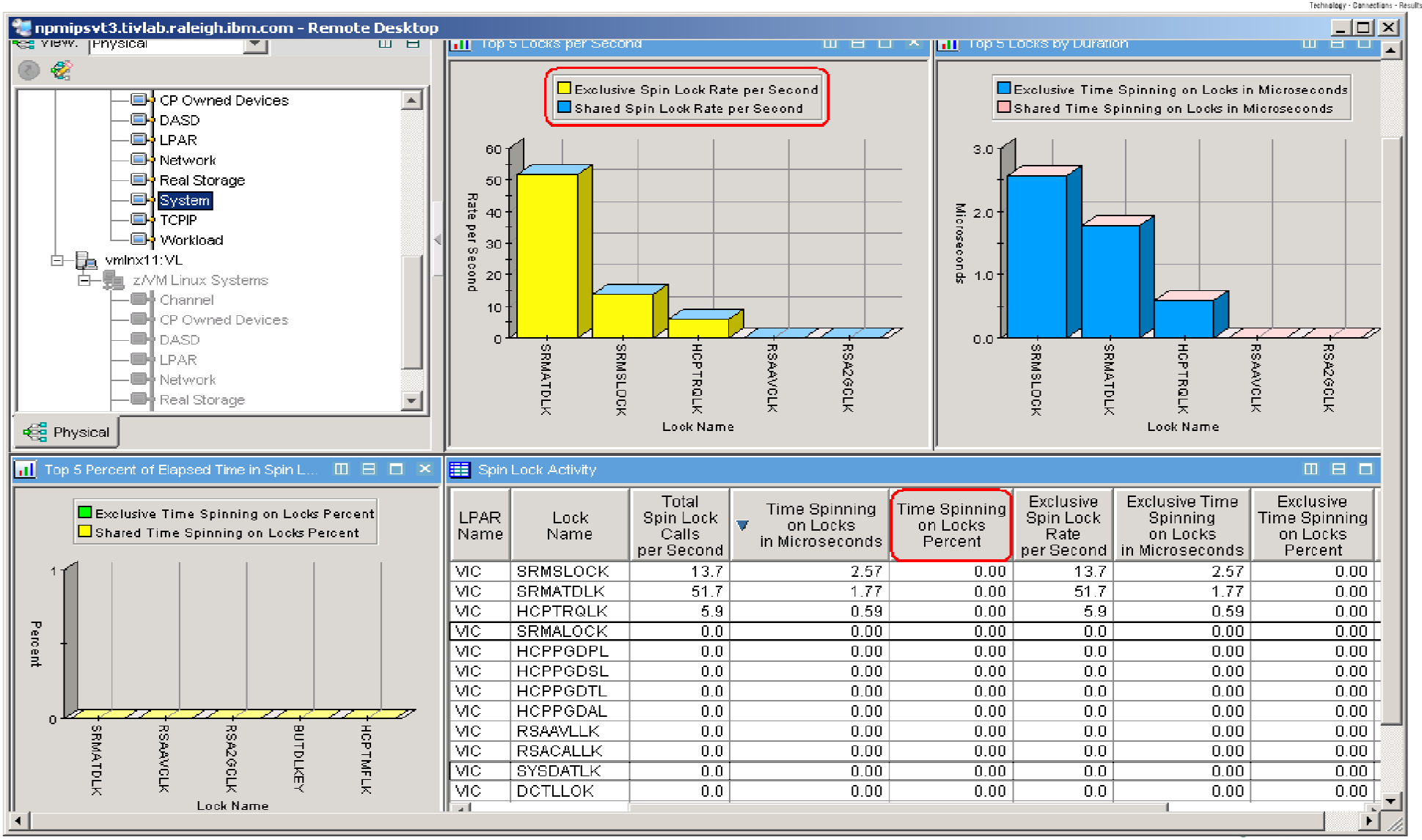


	System ID	User ID	Total CP % of CPU	CP Seconds	Total CPU Percent	CPU Seconds	Session Time	Total Virtual CPU%	Working Set Size	Workload Group	Linux Guest ID	Virtual CPUs	CP % of CPU	CPU Percent	Virtual CPU %
7	GDLVICOM	KWUSER3	0.01	0	0.05	0	1	0.04	56768			2	0.00	0.02	0.02
7	GDLVICOM	KWUSER2	0.01	0	0.21	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%. If larger should be investigated with z/VM Support.**
  - Increases as number of logical processors increases.

Spinlock Information



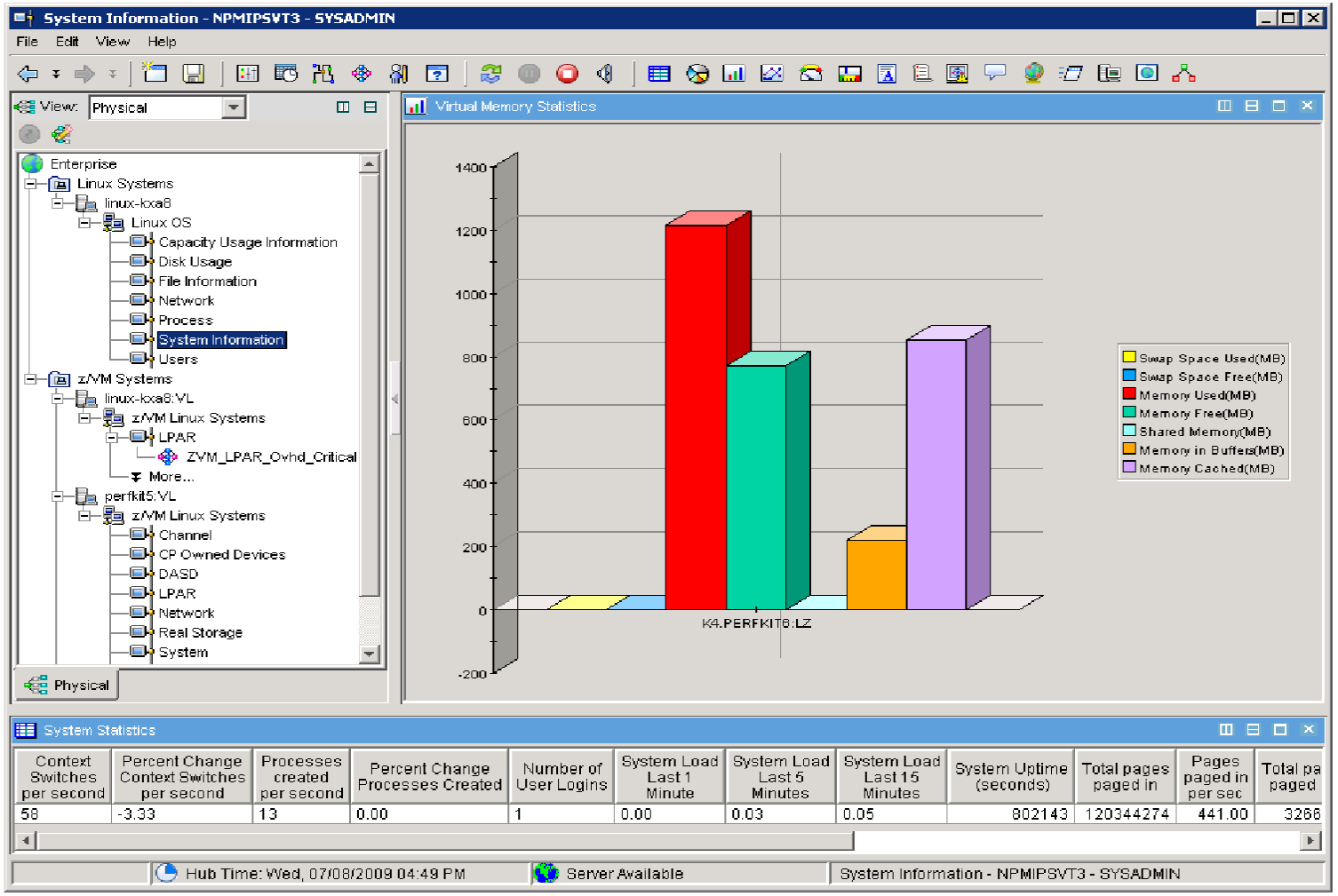
LPAR Name	Lock Name	Total Spin Lock Calls per Second	Time Spinning on Locks in Microseconds	Time Spinning on Locks Percent	Exclusive Spin Lock Rate per Second	Exclusive Time Spinning on Locks in Microseconds	Exclusive Time Spinning on Locks Percent
VIC	SRMSLOCK	13.7	2.57	0.00	13.7	2.57	0.00
VIC	SRMATDLK	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.
  - More memory is not always better.
  - Excessive virtual machine sizes negatively impact performance.
  - Linux uses any extra storage for caching of data. For shared resources, this is an impact.
    - Larger Linux guests means that z/VM has to page out larger virtual machines when running other guests
    - Note the example on the next page. This may be a good candidate for resizing.
  - Reduce the size of the Linux guest until it starts to swap (use VDISK for swap).
  - A good exercise is to compare Linux memory usage to z/VM working set size for the guest.

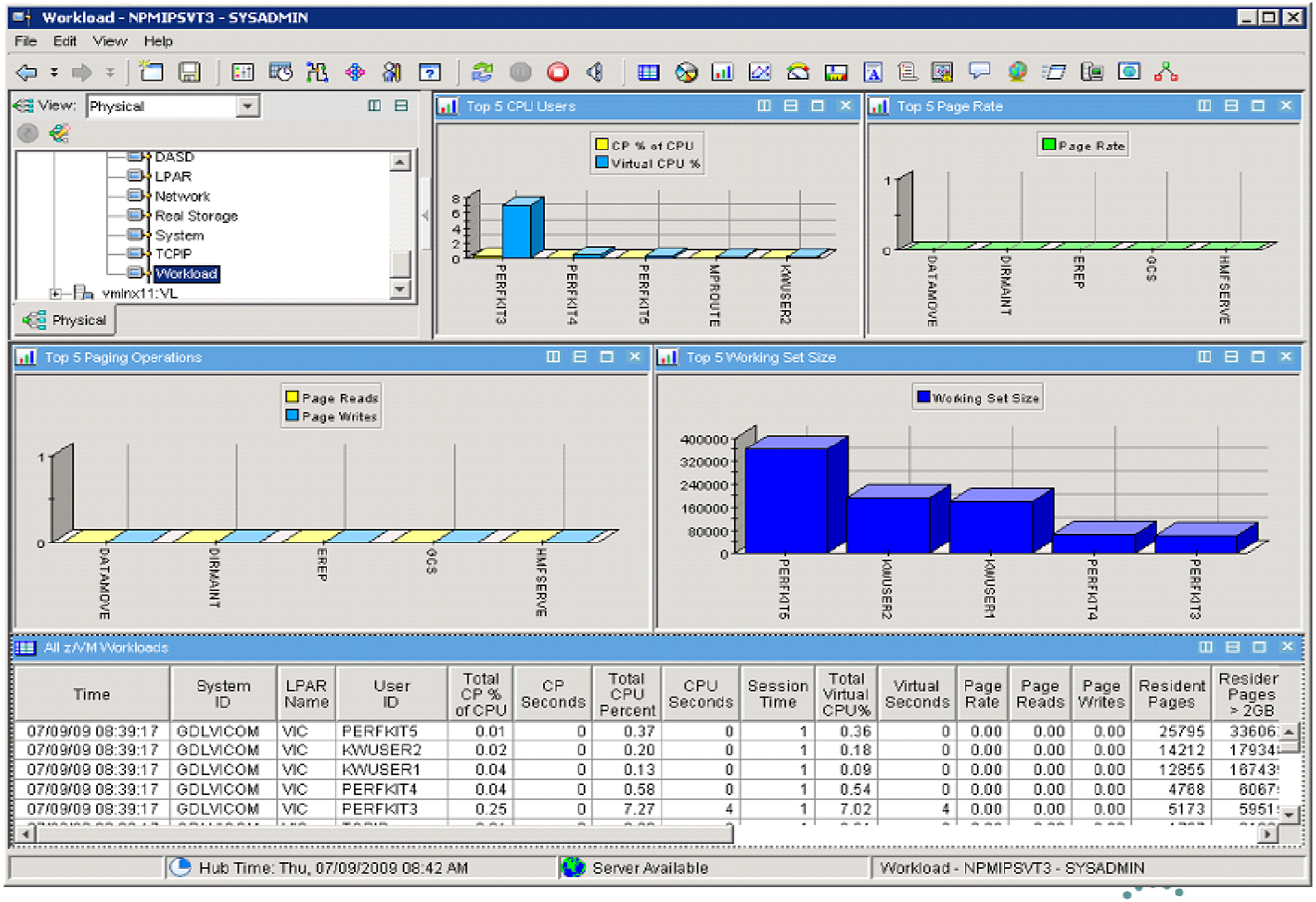
Sizing Linux Guests

Memory usage of a particular Linux virtual machine

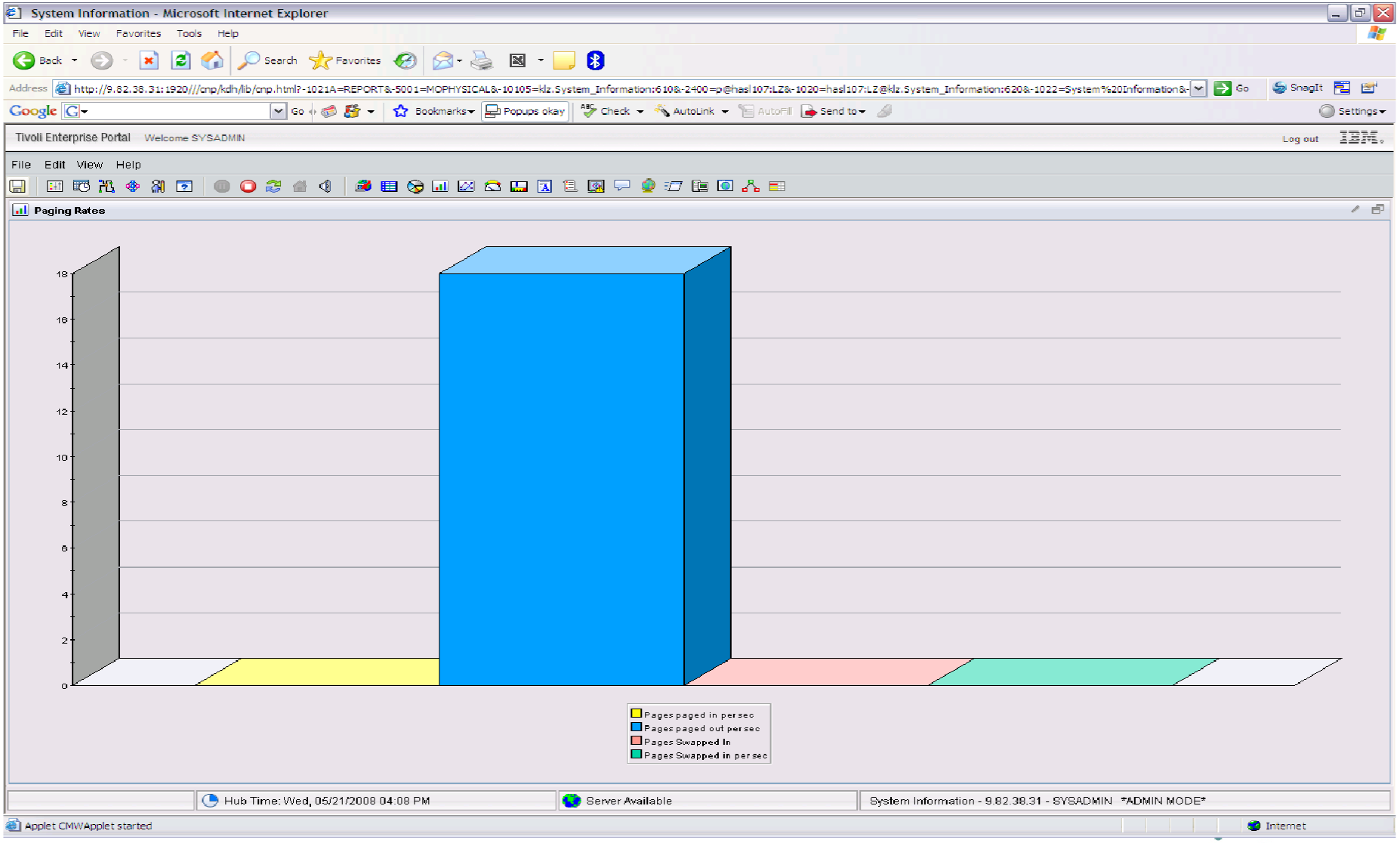


Sizing Linux Guests

Working Set Size can be found on the Workload workspace of the z/VM agent



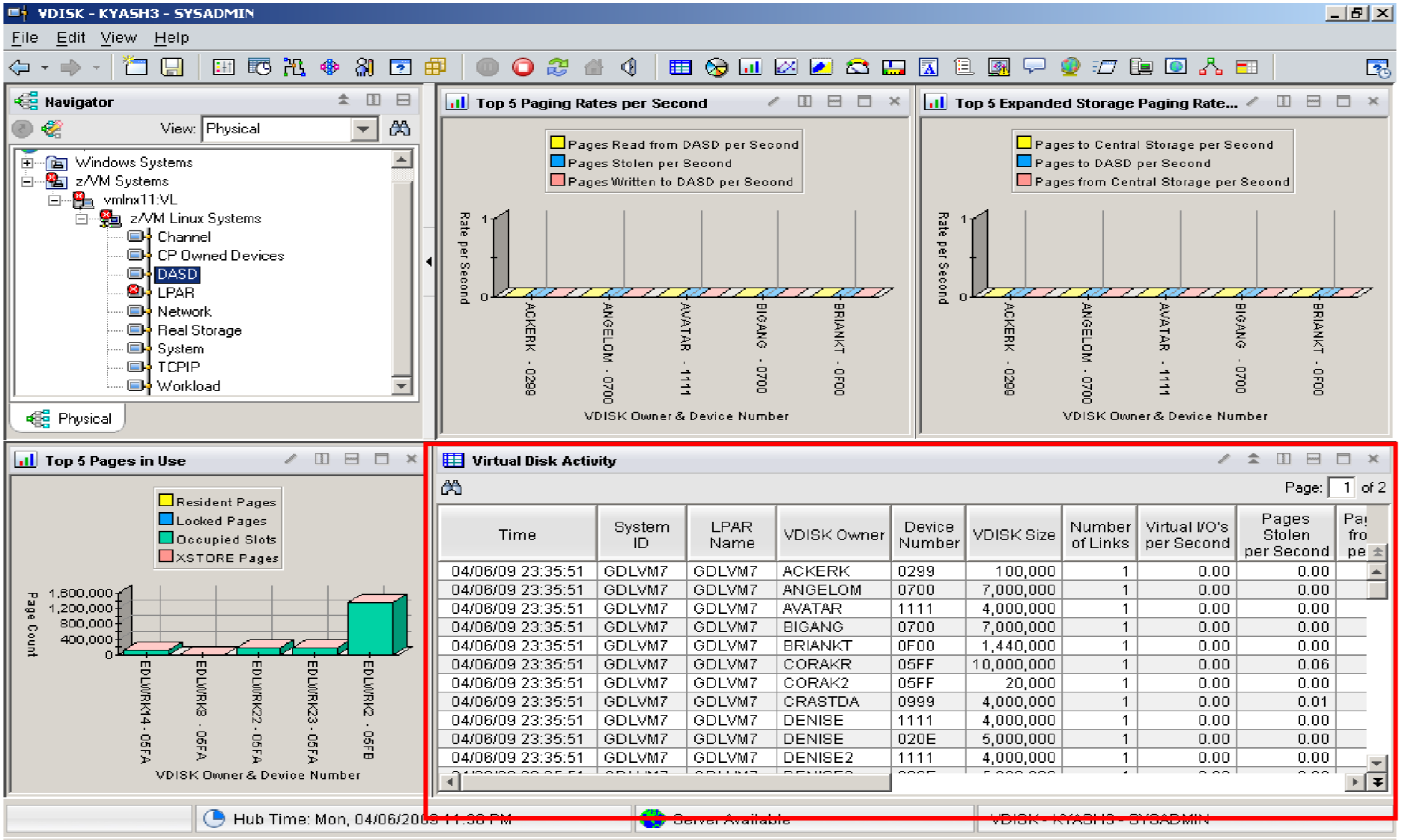
Page/Swap Attributes



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.
  - Need to factor VDISK in the overall memory planning for systems.
  - Not recommended in a storage-constrained z/VM system.

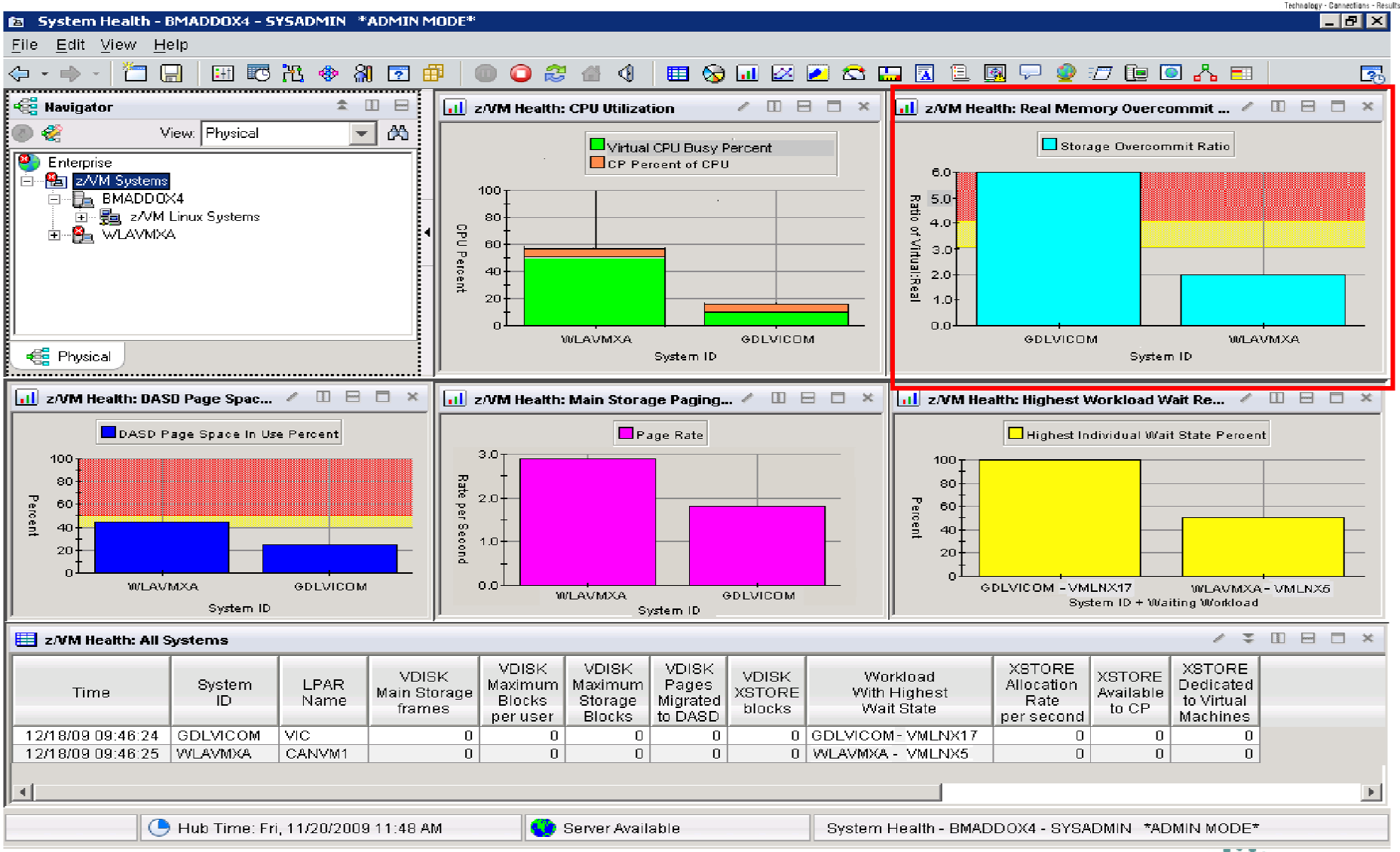
VDISK Information



Memory Configuration

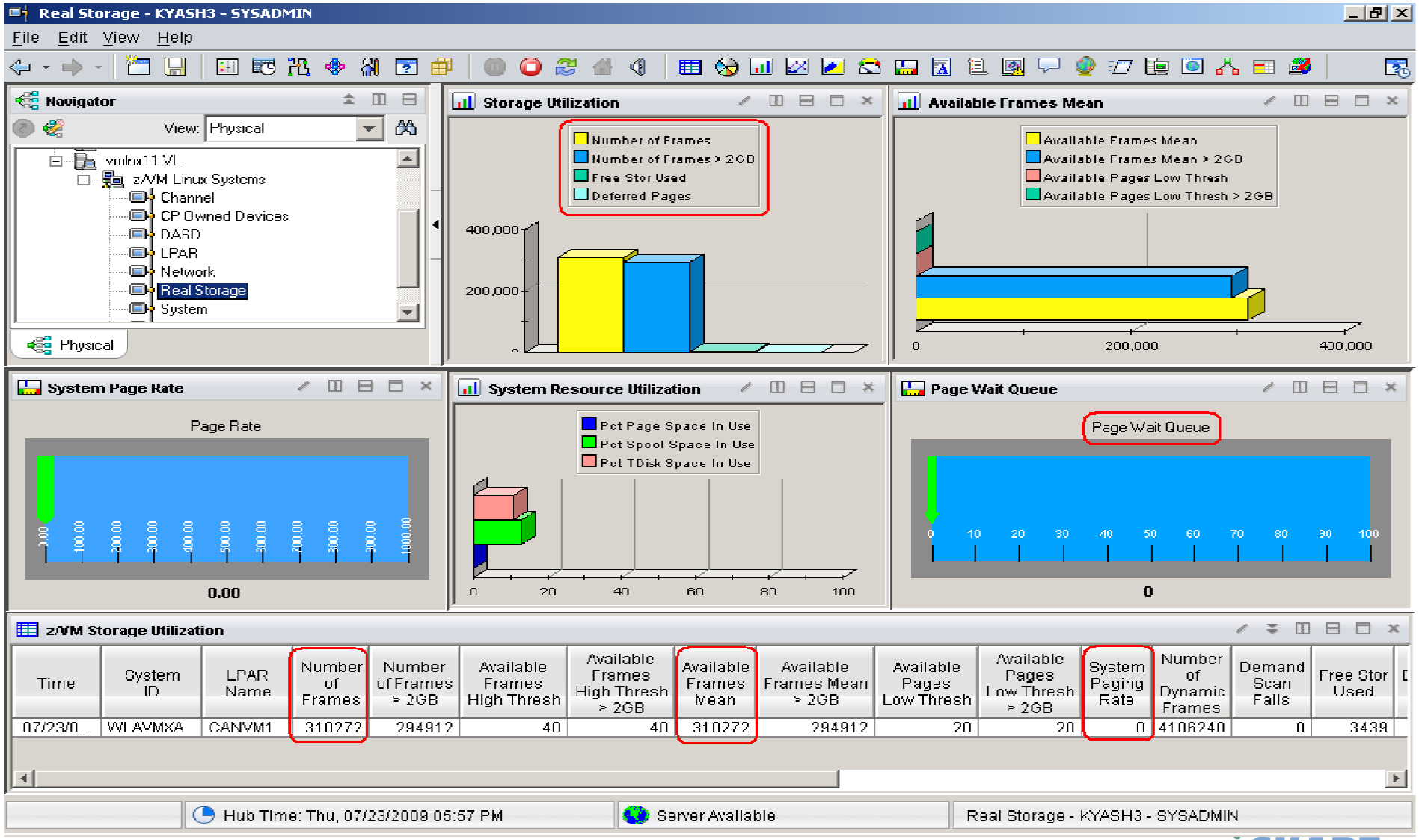
- Plan on a virtual to real (V:R) memory ratio in the range of 1.5:1 to 3:1.
- z/VM's architecture still benefits from expanded storage:
  - Serves as high speed cache.
  - Increases consistency of response time.
  - See http://www.vm.ibm.com/perf/tips/storconf.html for the gory details.
- Rule of Thumb - 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.

Memory Configuration



Time	System ID	LPAR Name	VDISK Main Storage frames	VDISK Maximum Blocks per user	VDISK Maximum Storage Blocks	VDISK Pages Migrated to DASD	VDISK XSTORE blocks	Workload With Highest Wait State	XSTORE Allocation Rate per second	XSTORE Available to CP	XSTORE Dedicated to Virtual Machines
12/18/09 09:46:24	GDLVICOM	VIC	0	0	0	0	0	GDLVICOM - VMLNX17	0	0	0
12/18/09 09:46:25	WLAVMXA	CANVM1	0	0	0	0	0	WLAVMXA - VMLNX5	0	0	0

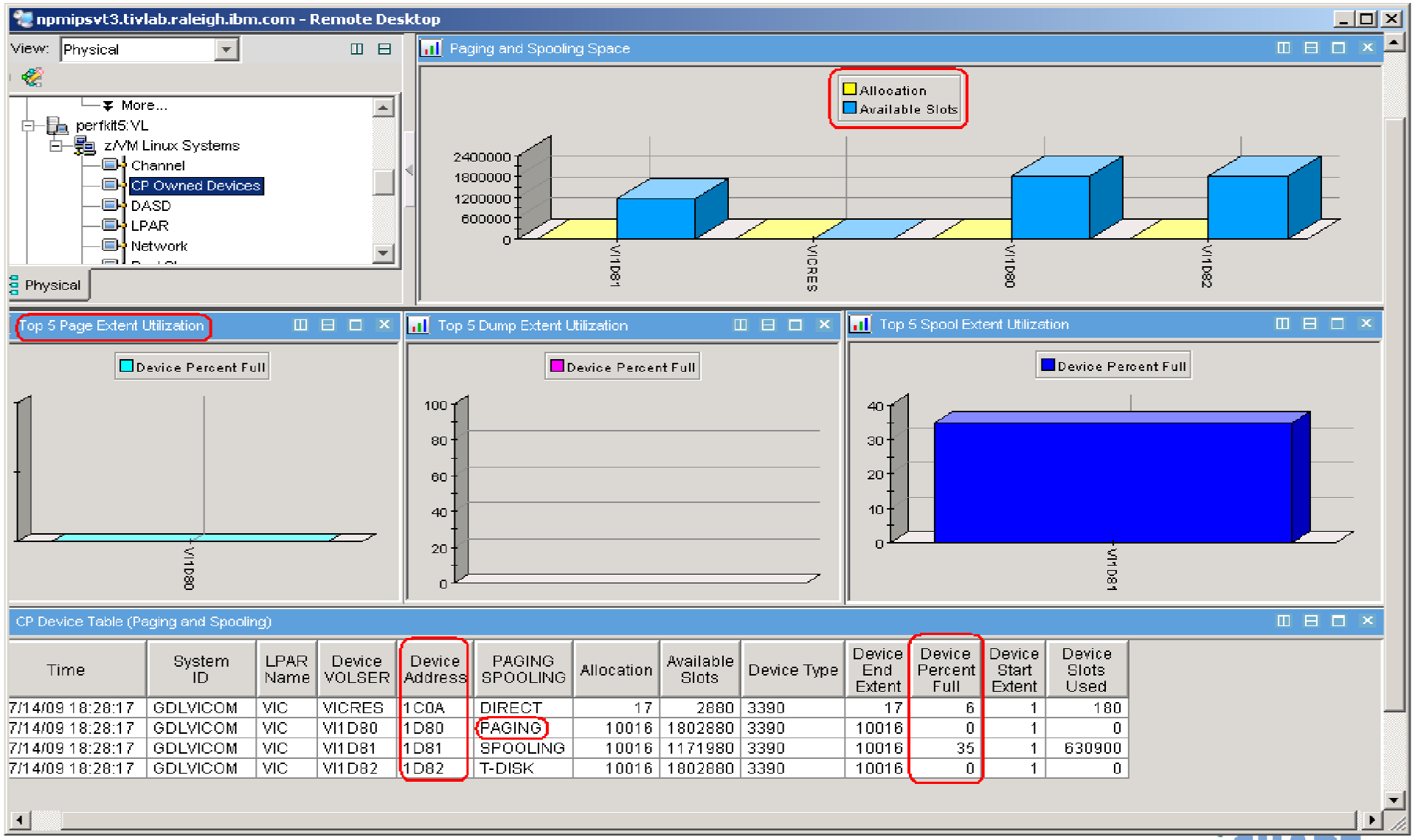
Memory Configuration



Paging Subsystem

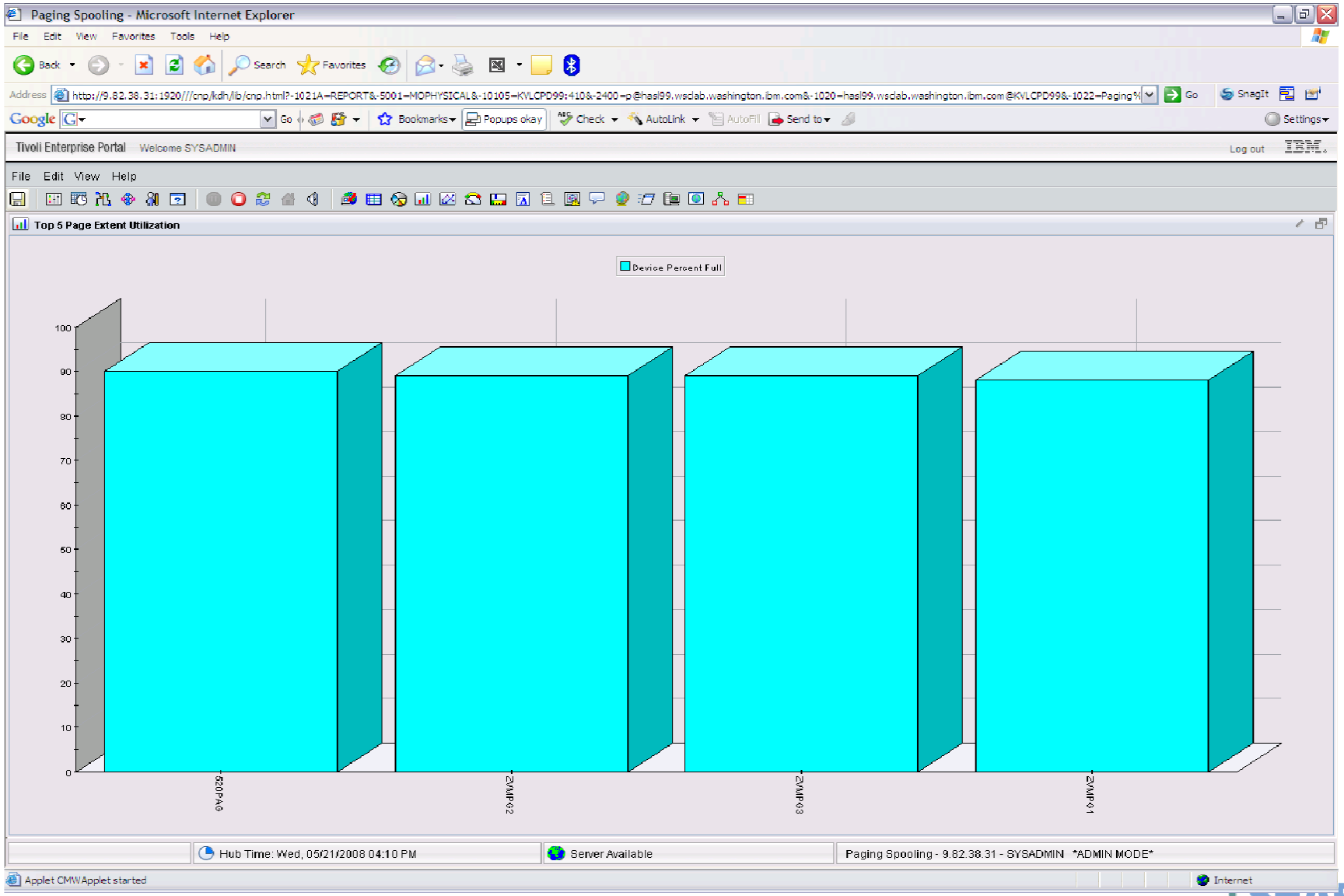
- Plan for DASD page space utilization < 50%:
  - Page space tends to get fragmented over time.
  - Large contiguous free space allows for greater paging efficiency.
  - Monitor usage with OMEGAMON XE or Q ALLOC PAGE command.
- Do not mix page space with any other space on a volume.
- Recommend using devices of the same size/geometry.
- When Paging fills up, it spills over to the Spool. When Spool fills up, z/VM abends. z/VM issues a warning message at 90%, by then it is typically too late.
- Calculation guidelines are located in the CP Planning and Administration Manual.
- Changes in z/VM 6.3. Need for maintaining <50% for efficiency removed. However, managing for availability still an issue.

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



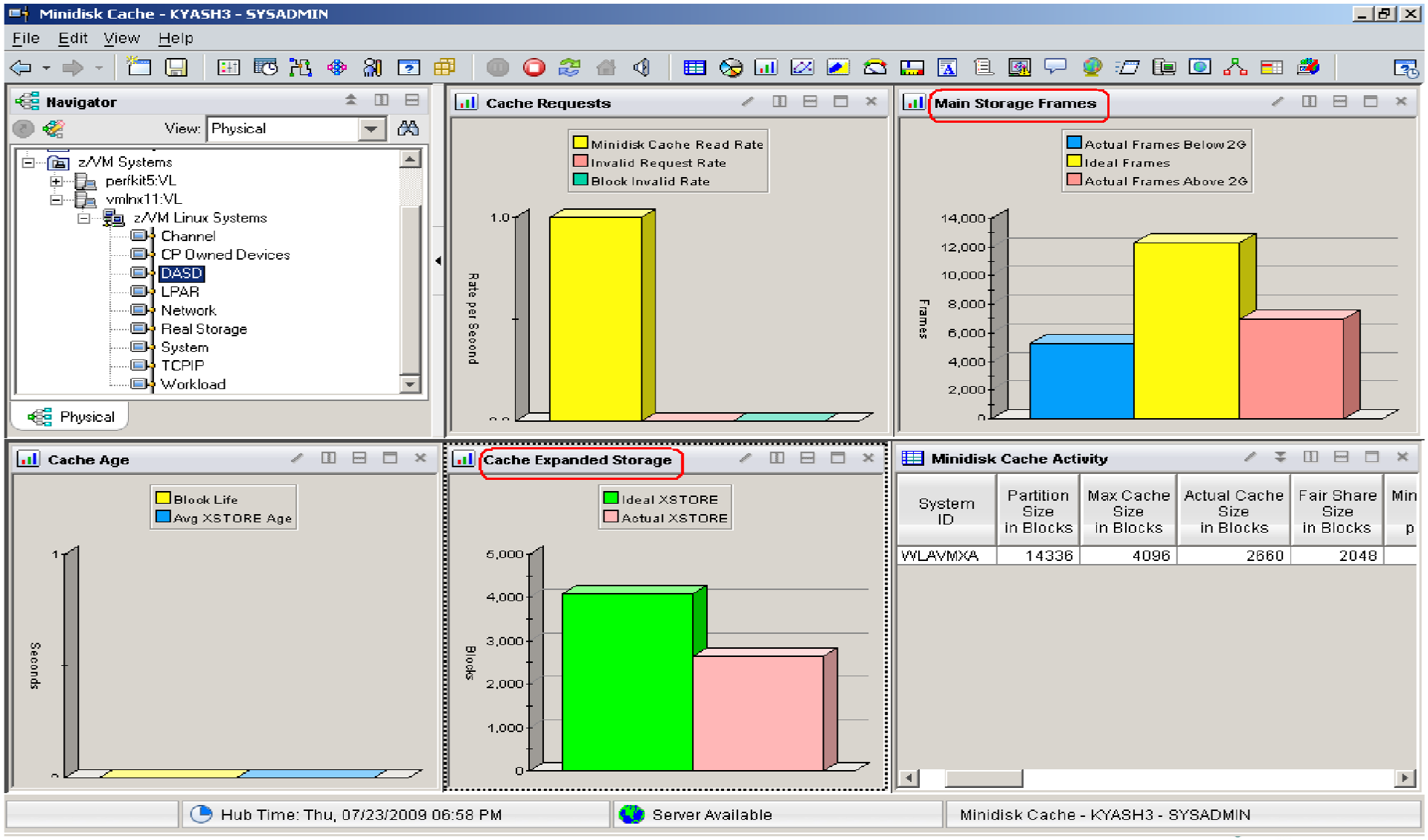
Minidisk Cache

- 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.

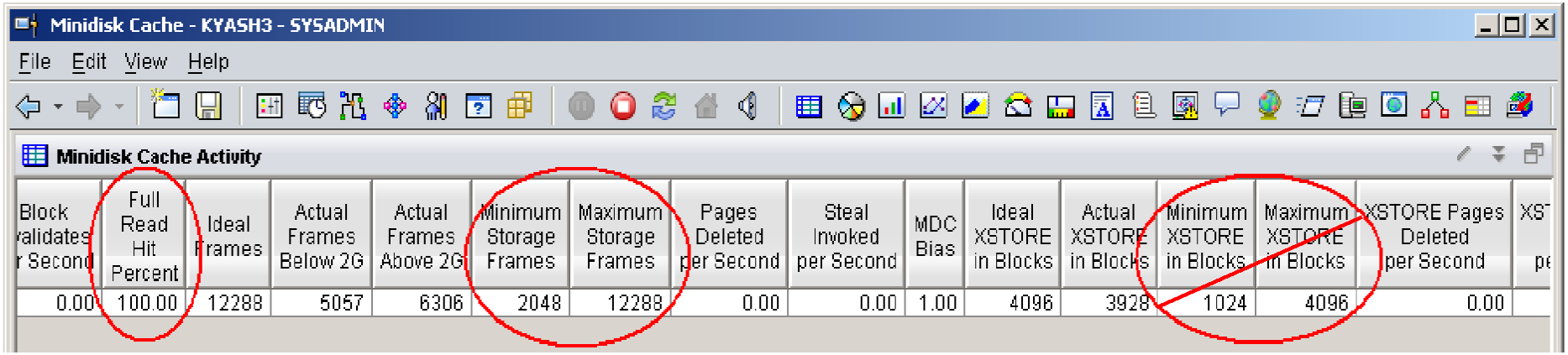
MDC and FlashCopy Interaction

- 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

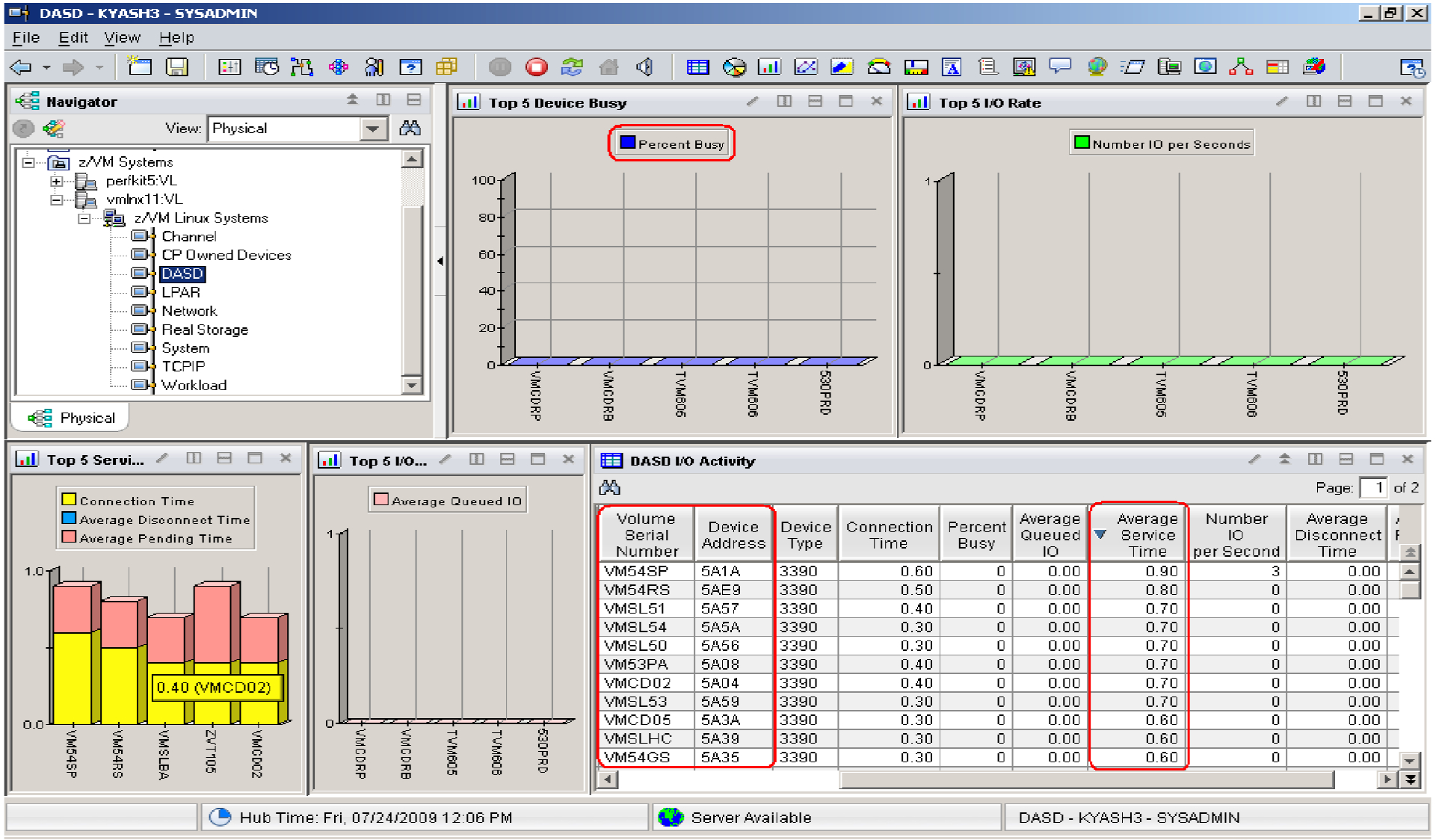


Block validates r Second	Full Read Hit Percent	Ideal Frames	Actual Frames Below 2G	Actual Frames Above 2G	Minimum Storage Frames	Maximum Storage Frames	Pages Deleted per Second	Steal Invoked per Second	MDC Bias	Ideal XSTORE in Blocks	Actual XSTORE in Blocks	Minimum XSTORE in Blocks	Maximum XSTORE in 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



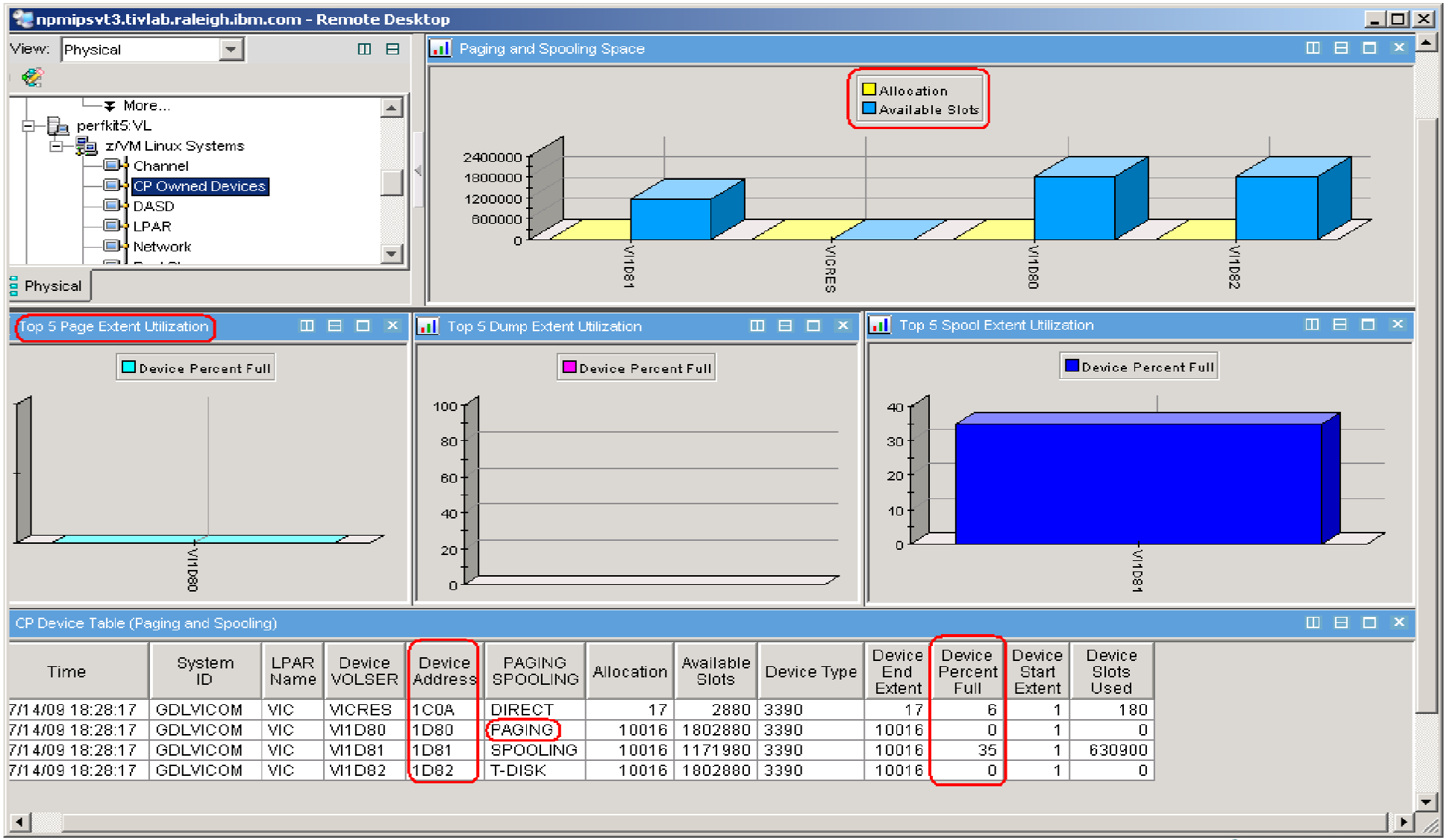
System Dump & Spool Space

- 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
    - APAR VM64548 is open to address this.
- 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

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

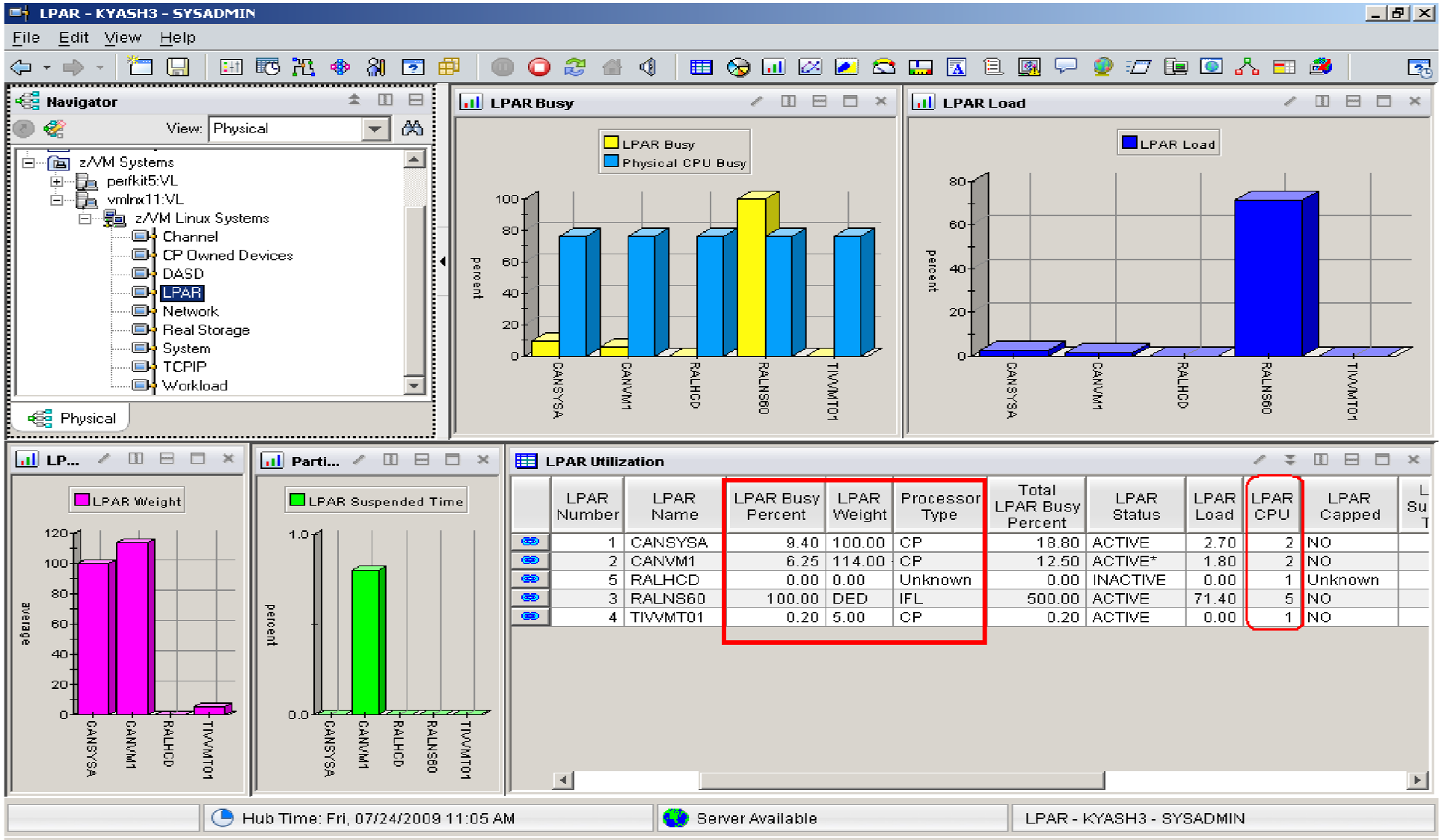
Do not ignore the hardware!

- 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

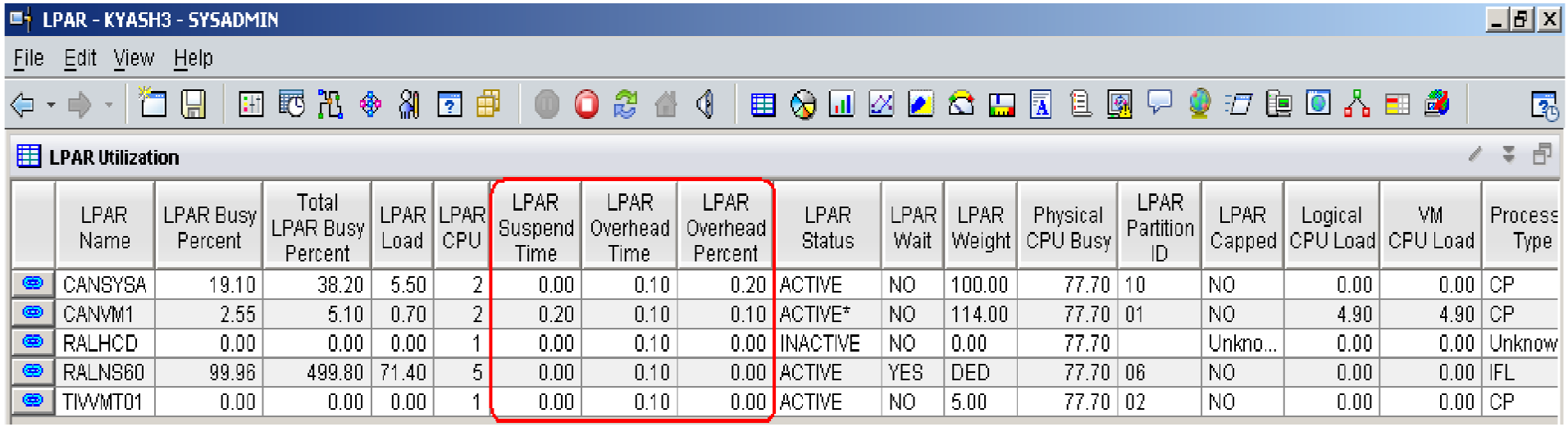
- Logical Processors
  - LPAR recommendation – no greater than a 4:1 logical to real ratio.
  - z/VM 5.1 - z/VM 5.2 support up to 24 processors.
  - z/VM 5.3 - z/VM 6.3 support up to 32 processors.

LPAR Utilization



	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

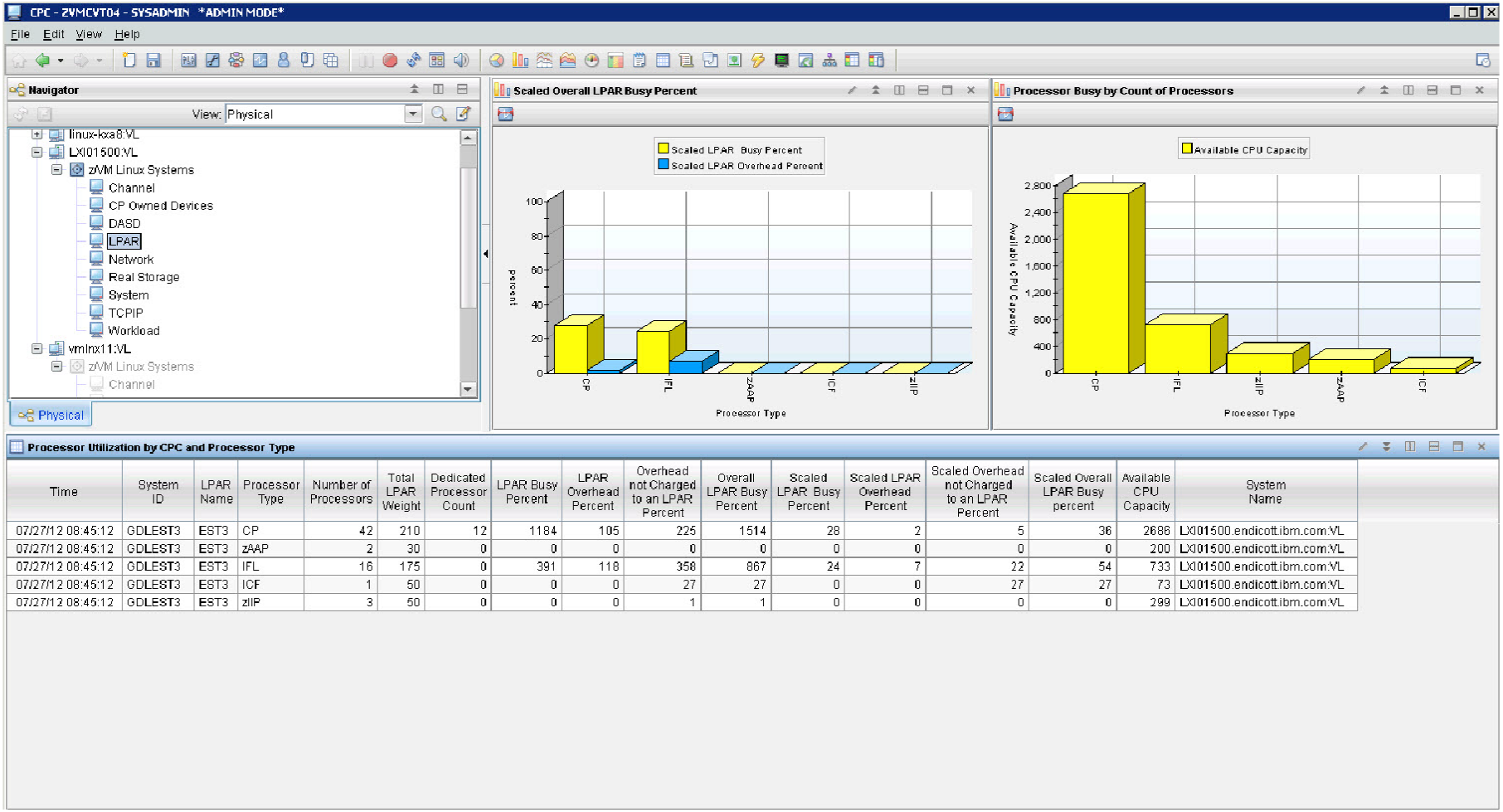
LPAR Utilization



	LPAR Name	LPAR Busy Percent	Total LPAR Busy Percent	LPAR Load	LPAR CPU	LPAR Suspend Time	LPAR Overhead Time	LPAR Overhead Percent	LPAR Status	LPAR Wait	LPAR Weight	Physical CPU Busy	LPAR Partition ID	LPAR Capped	Logical CPU Load	VM CPU Load	Process Type
	CANSYSA	19.10	38.20	5.50	2	0.00	0.10	0.20	ACTIVE	NO	100.00	77.70	10	NO	0.00	0.00	CP
	CANVM1	2.55	5.10	0.70	2	0.20	0.10	0.10	ACTIVE*	NO	114.00	77.70	01	NO	4.90	4.90	CP
	RALHCD	0.00	0.00	0.00	1	0.00	0.10	0.00	INACTIVE	NO	0.00	77.70		Unkno...	0.00	0.00	Unknow
	RALNS60	99.96	499.80	71.40	5	0.00	0.10	0.00	ACTIVE	YES	DED	77.70	06	NO	0.00	0.00	IFL
	TIVVMT01	0.00	0.00	0.00	1	0.00	0.10	0.00	ACTIVE	NO	5.00	77.70	02	NO	0.00	0.00	CP

- **LPAR Suspend Time: RoT: 5% Suspend time is yellow line, 10% is red line for concern.**
- **LPAR Overhead: This should generally be less than 5% of the Physical IFLs (CEC in an all-IFL configuration) for general LPAR management overhead, and then less than 5% of the z/VM partition IFLs.**

CPC workspace



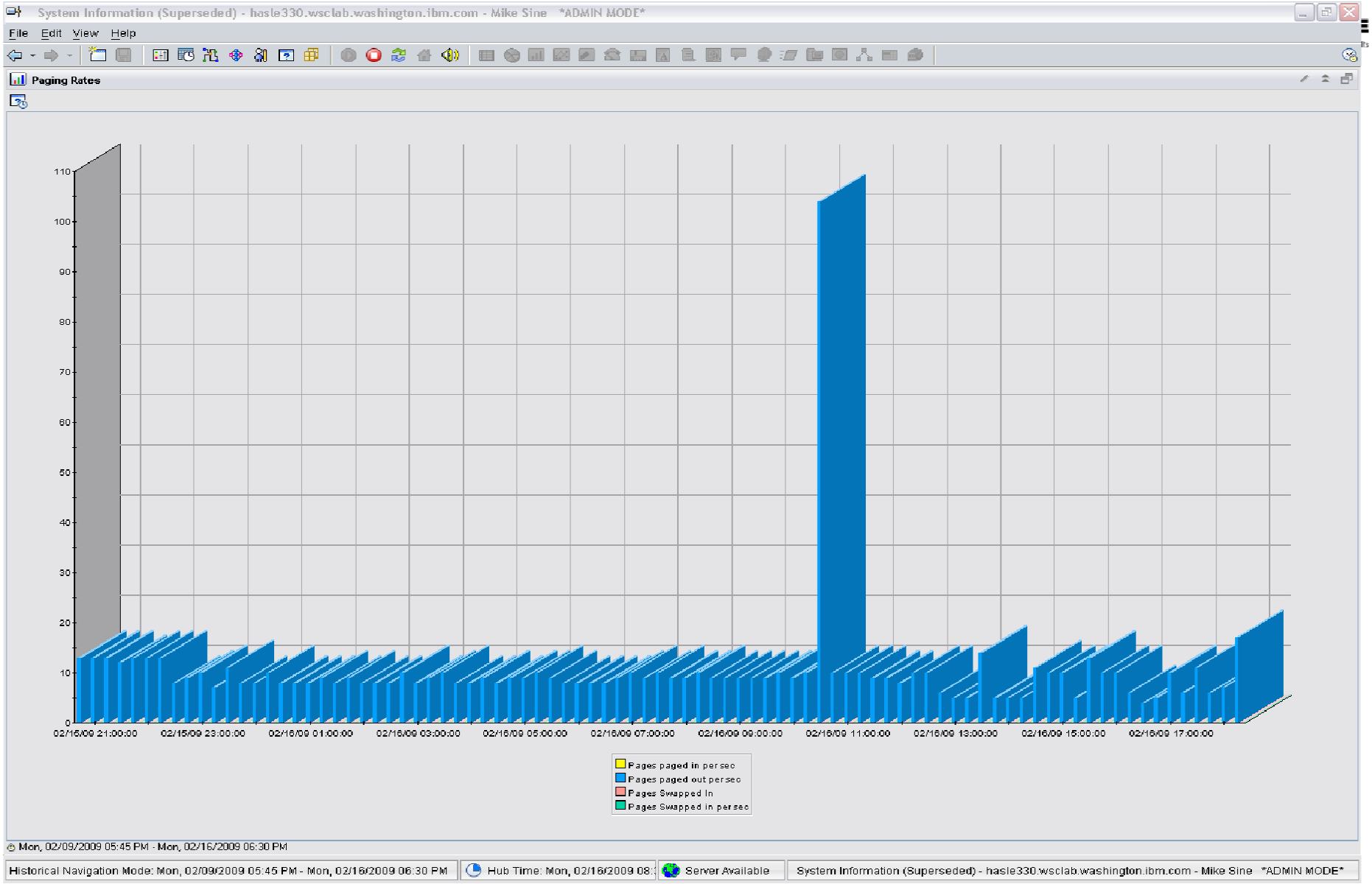
Time	System ID	LPAR Name	Processor Type	Number of Processors	Total LPAR Weight	Dedicated Processor Count	LPAR Busy Percent	LPAR Overhead Percent	Overhead not Charged to an LPAR Percent	Overall LPAR Busy Percent	Scaled LPAR Busy Percent	Scaled LPAR Overhead Percent	Scaled Overhead not Charged to an LPAR 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	2688	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

Persistent Historical Views

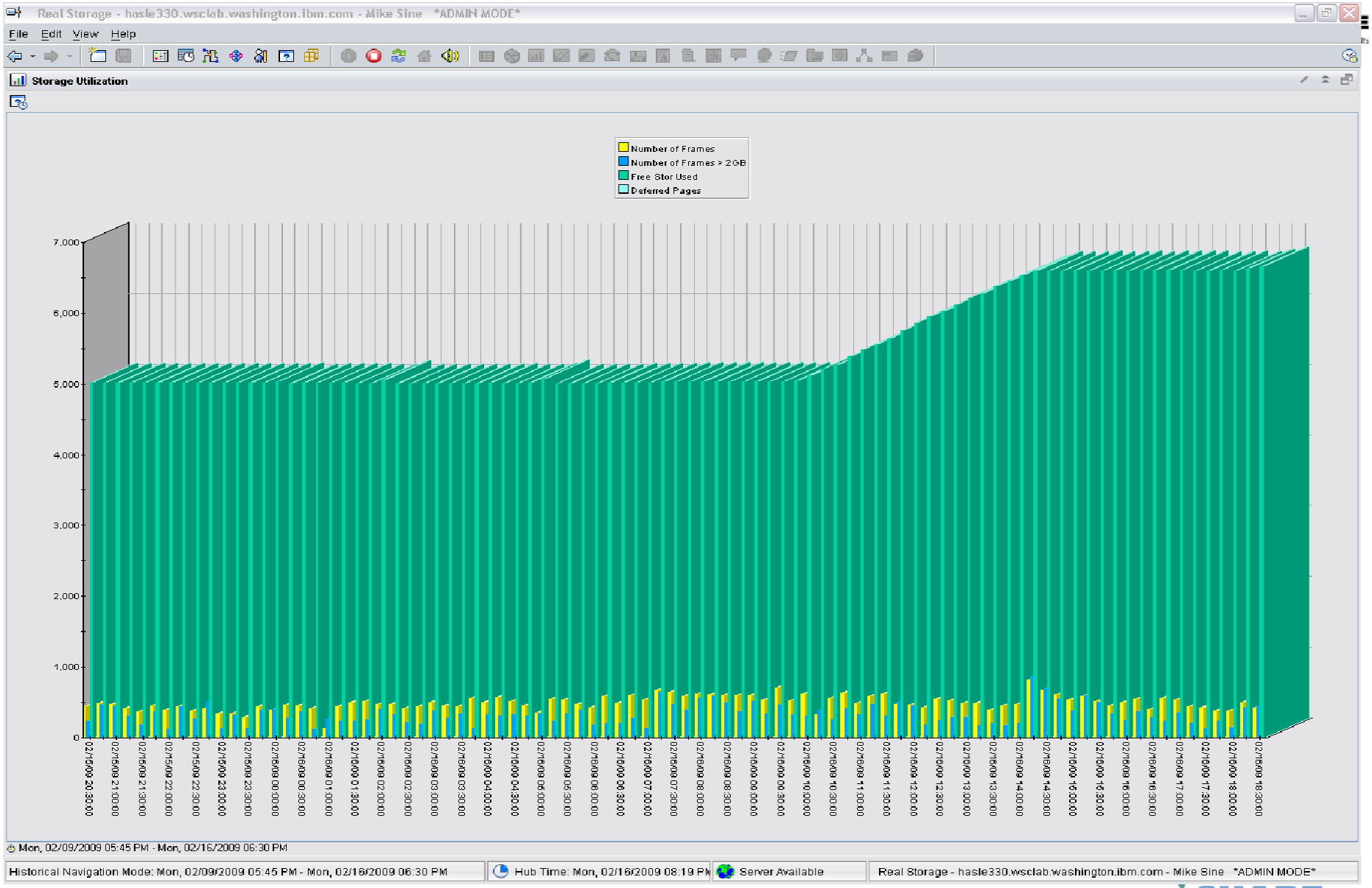
This makes it easier to see anomalies, or match spikes.
Capturing performance data as a base line is a must:

- 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.

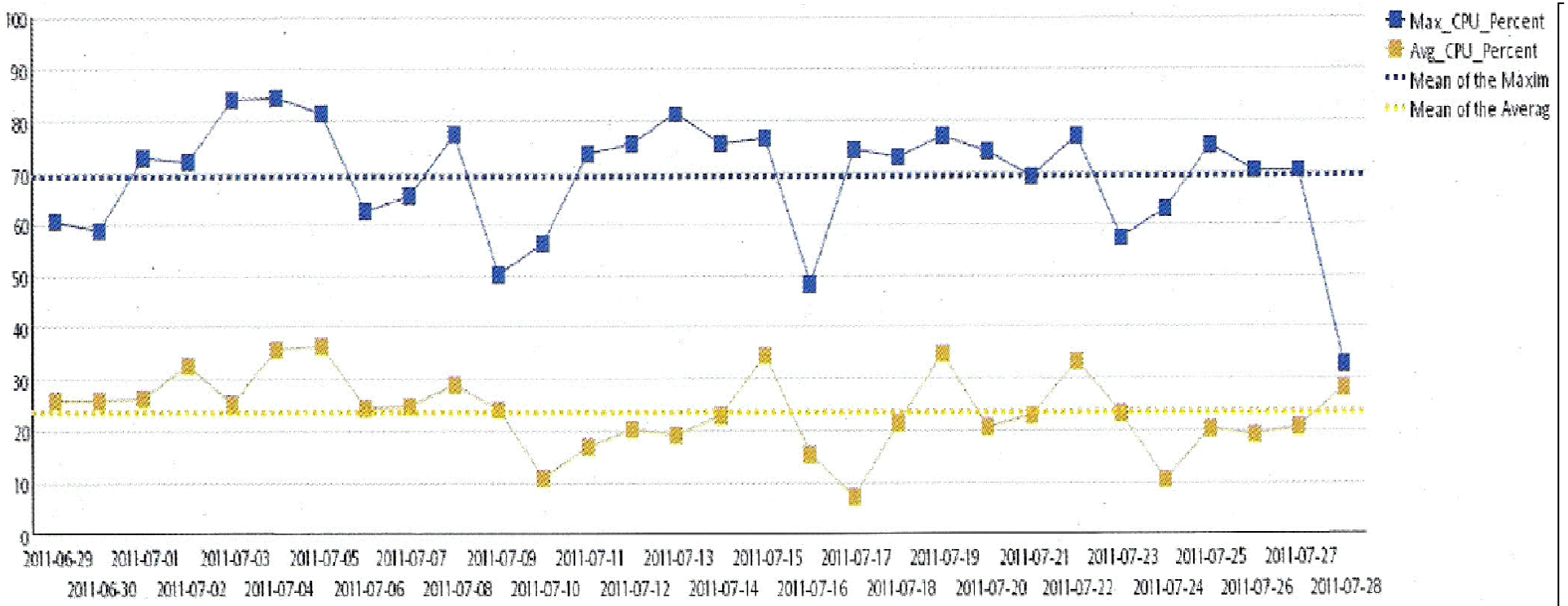
Persistent Historical Views



Persistent Historical Views



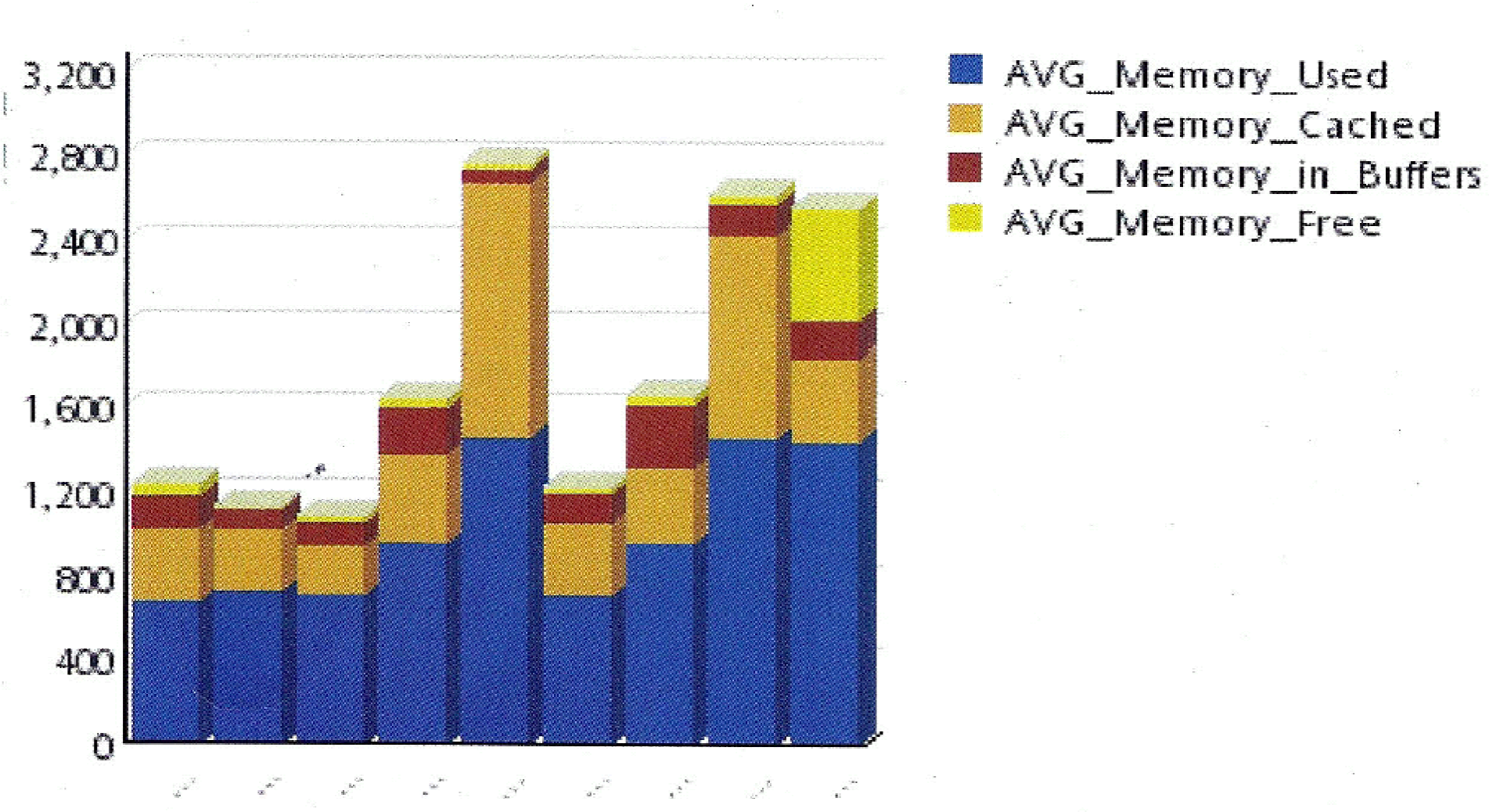
Max and Avg CPU example:



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:



AGENDA

- Introduction
- Best Practices Monitoring Requirements
  - Virtual Linux and z/VM performance considerations
  - Don’t forget the hardware
  - Integration from hardware – systems – applications Persistent historical views
- Integrated Monitoring Approach
- Linux on z Health Checker

An Integrated Monitoring Approach

- Provides performance monitoring for z/VM and Linux guests
- Executes automated actions in response to defined events
- Integrates 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
- OMEGAMON XE for z/VM and Linux
  - Linux agents gather performance data from Linux guests
    - Native Agent or agentless via SNMP
  - z/VM agent gathers performance data from z/VM
    - Including z/VM view of guests
    - Uses IBM Performance Toolkit for VM as its data source
    - Linux provides APPLDATA to CP monitor (another form of agentless monitoring)

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

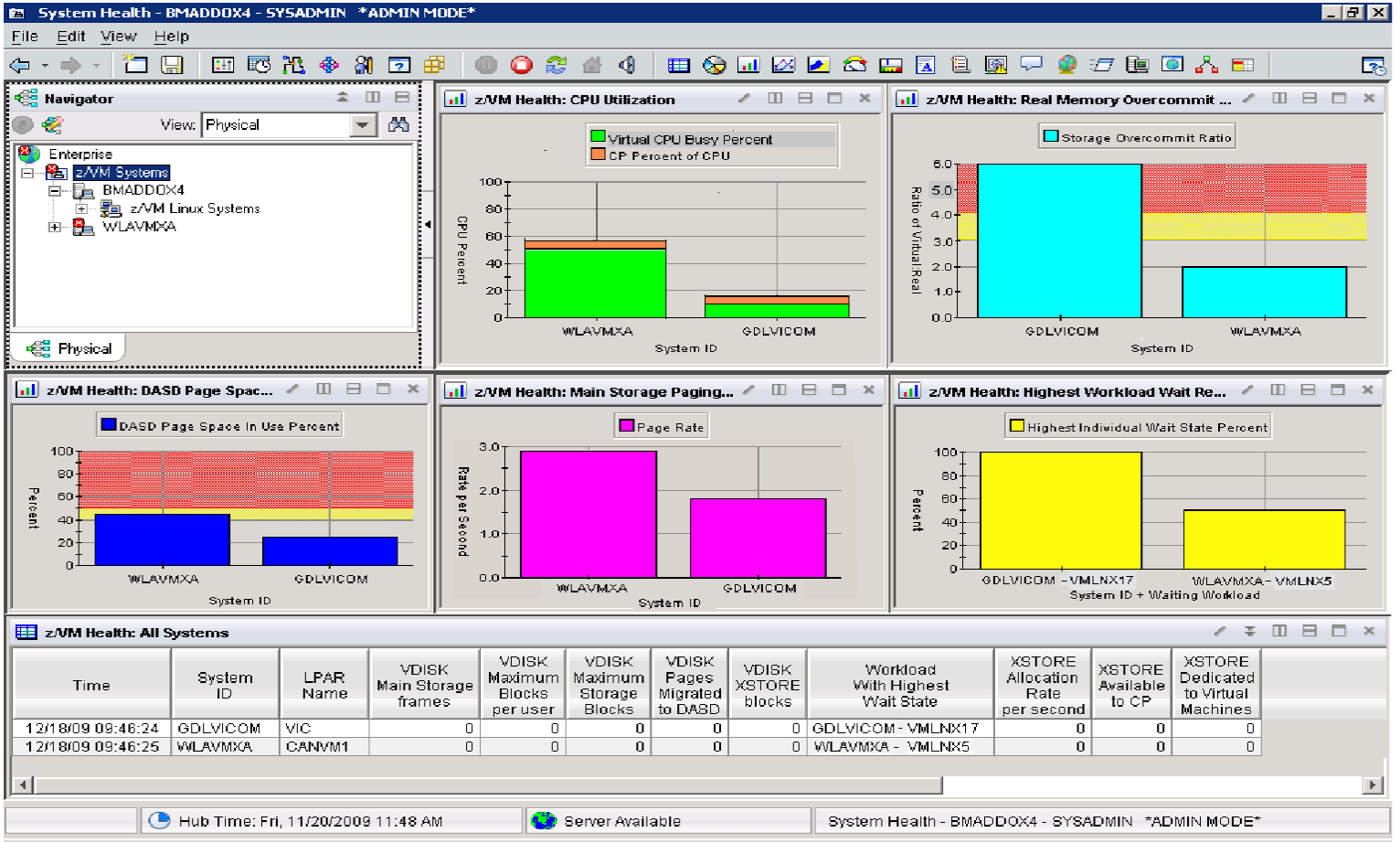
- Linux OS
- System Information
  - CPU aggregation
  - Virtual Memory Statistics
- Process
- Users
- Disk Usage
- File Information
- Network

Use Case Scenarios

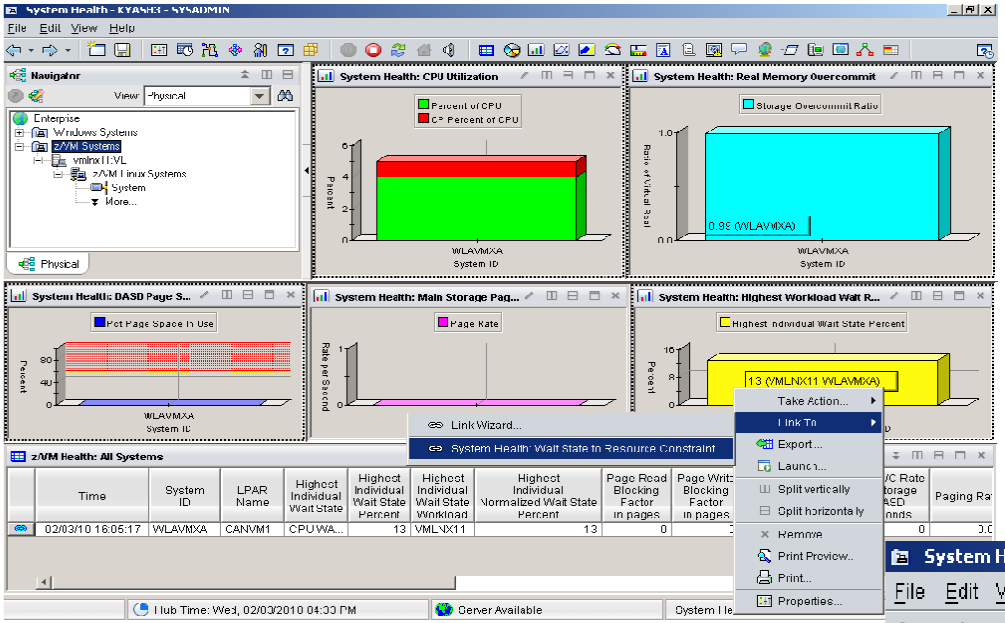
- **Overall health of your z/VM systems**
- **Adding Additional Linux Servers**
- **System running slowly**

Scenario 1— Overall Health of Your System

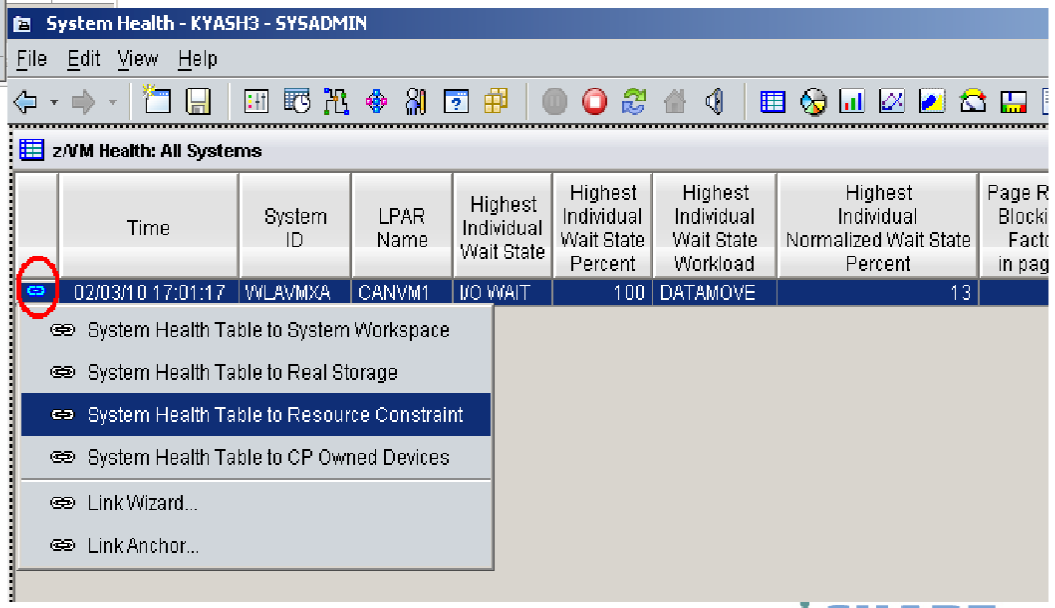
At a quick glance you can see the %CPU usage, ratio of real to virtual memory ratio, paging space, paging rates highest wait state, and VDISK usage for all your z/VM systems



Scenario 1— Overall Health of Your System

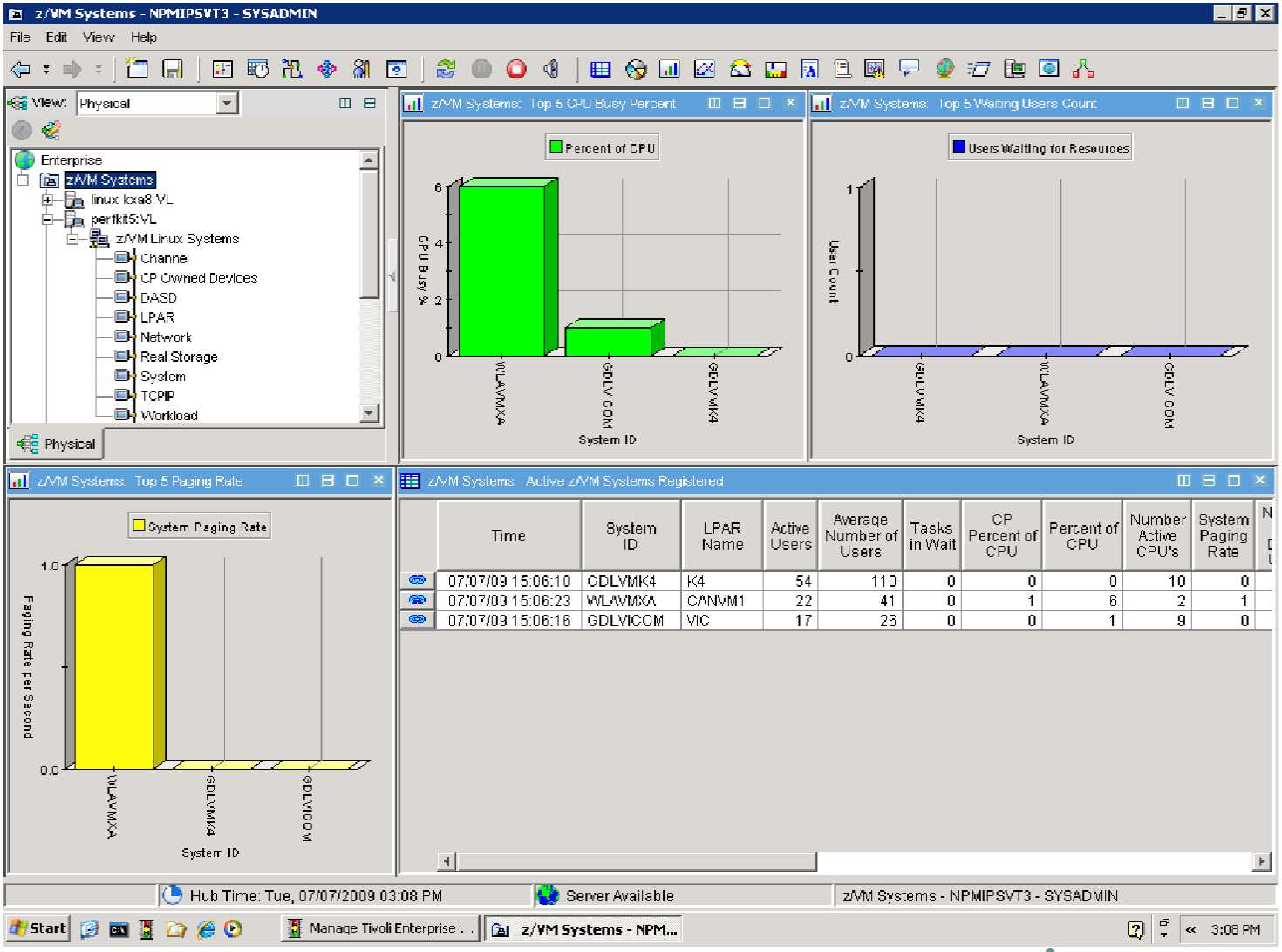


Links are available to drill down for further detailed data



Scenario 1 — Overall Health of Your System

By following the link to the System workspace, you can see at a quick glance the %CPU usage, number of users in a wait state, and paging rates of all your z/VM systems



Scenario 1— Overall Health of Your System

- **Things to look for**
  - CPU usage
    - Is any one system using more CPU than expected
    - Is any one system using less CPU than expected—you may have an underutilized processor and be wasting capacity
    - Remember, a DEDICATED processor will show 100%
  - Users waiting for resources
    - Number of users at the end of the monitoring interval who are either in:
      - Eligible list—waiting to enter the dispatch list
      - Nondispatchable
        - Waiting for paging
        - Waiting for I/O completion
      - Dispatchable
        - Waiting for a processor

Scenario 1— Overall Health of Your System

- **Things to look for**
  - System paging rate
    - Number of page reads per second
    - Not a complete indicator of your paging effectiveness, but a good first glance
      - If the rate is low, and you don't have many users waiting or paging to complete (dispatch list), then you don't have a problem
      - If rate is low and you DO have many users in dispatch list, it may be an indication of a paging problem.
      - High dispatch list number could be for other reasons such as I/O contention. You need to check.
    - If the rate is high, then you may need to tune your paging subsystem.

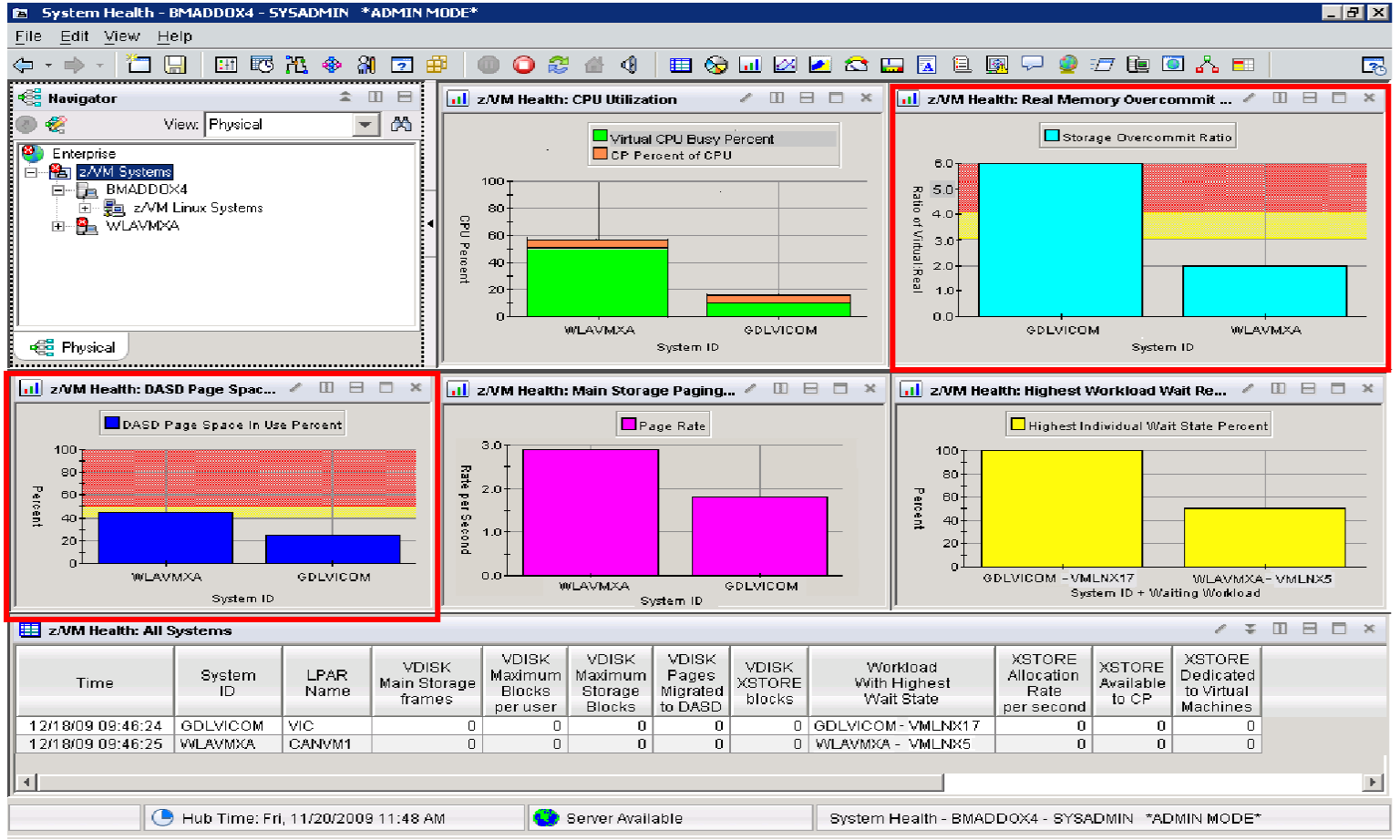
Scenario 2— Adding Additional Linux Servers

Again by using the System Health Workspace at a quick glance you can see ratio of real to virtual memory ratio.

As a rule of thumb you do not want to overcomitt memory greater than 3:1

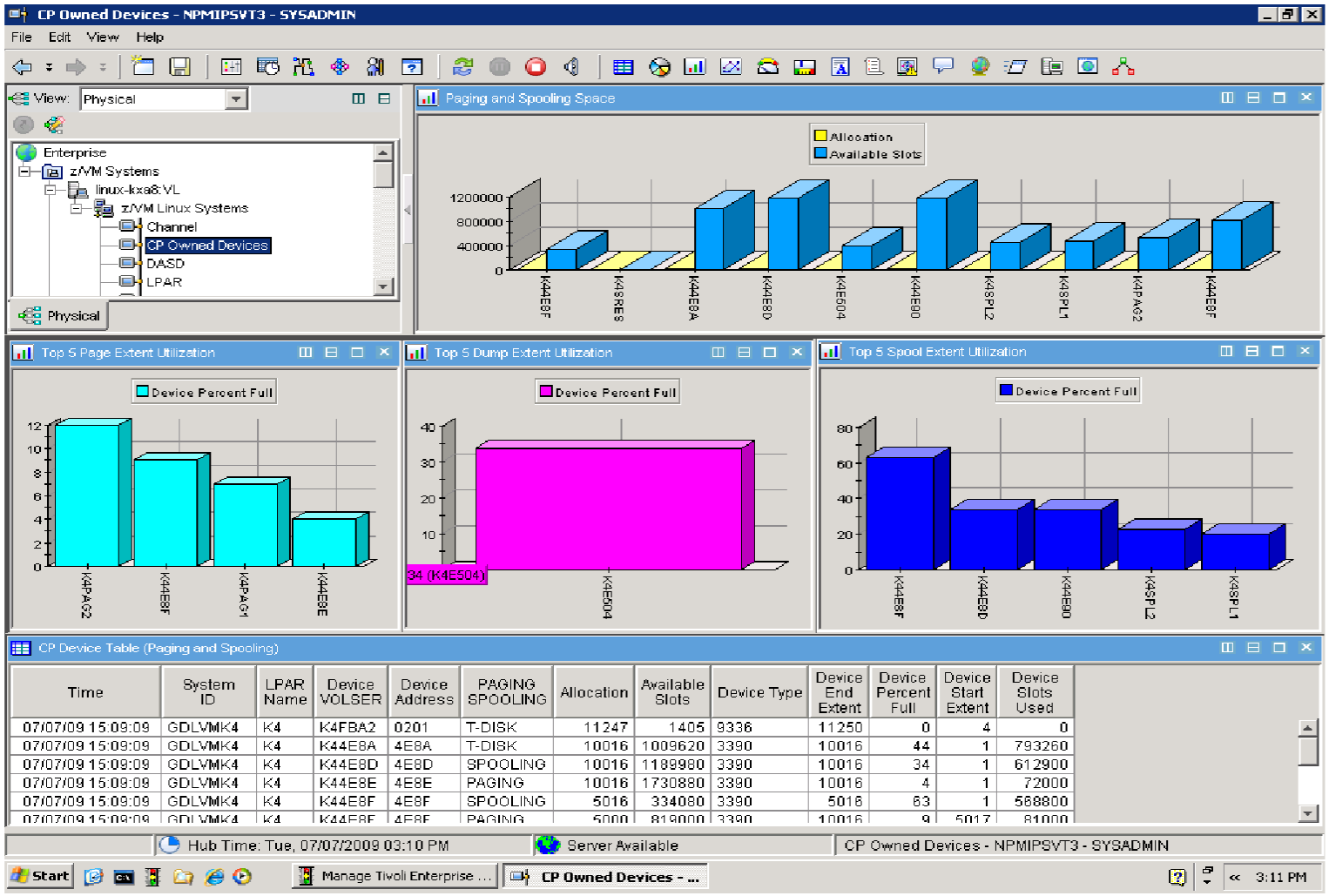
Additional page space is also needed to be added before more workload is added

To better understand the overall Paging Utilization Data, follow the link from the DASD Page Space Utilization view to get additional details on the paging configuration



Scenario 2 — Adding Additional Linux Servers

Using the information in the CP Owned Volumes workspace, one can determine available paging slots, the allocation of existing free space and whether the paging subsystem can handle additional large guests

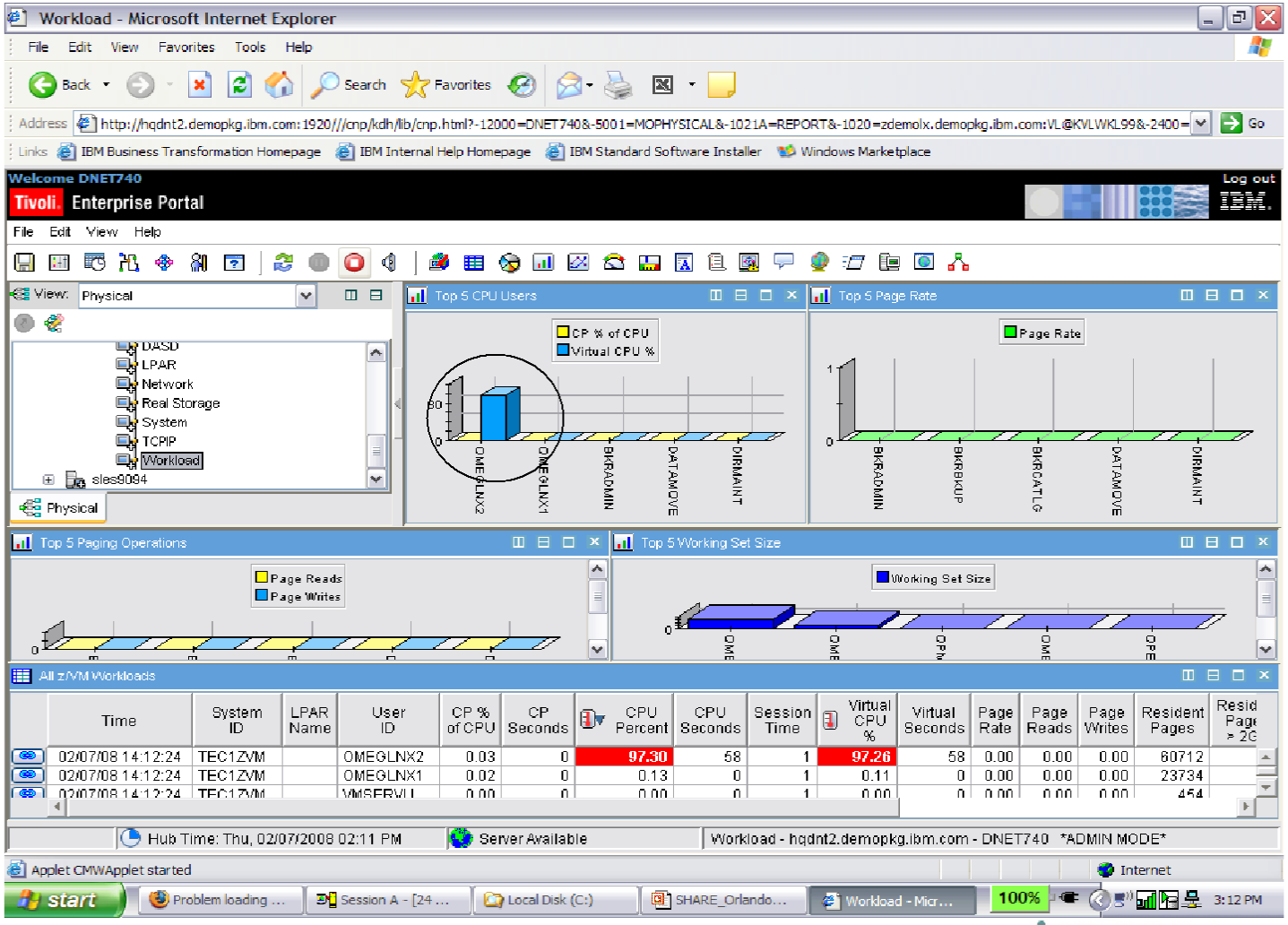


Scenario 2 — Adding Additional Linux Servers

- **General tips**
  - Page space utilization should always be < 50%
  - Never put Paging and Spool space on the same volume
  - Allocate Spool and Page volumes to try and reduce I/O contention by separating them as much as possible (control unit, channel, etc)
  - Dedicated paging devices reduce contention for paging
  - Try to avoid putting highly used files on the same volume as paging and spool space, such as the CMS system disk
  - Use your fastest devices for Paging
  - Multiple Paging devices allow more overlap of paging operations
  - Expanded storage can be used for paging
  - Directory space is not heavily used, can be placed anywhere

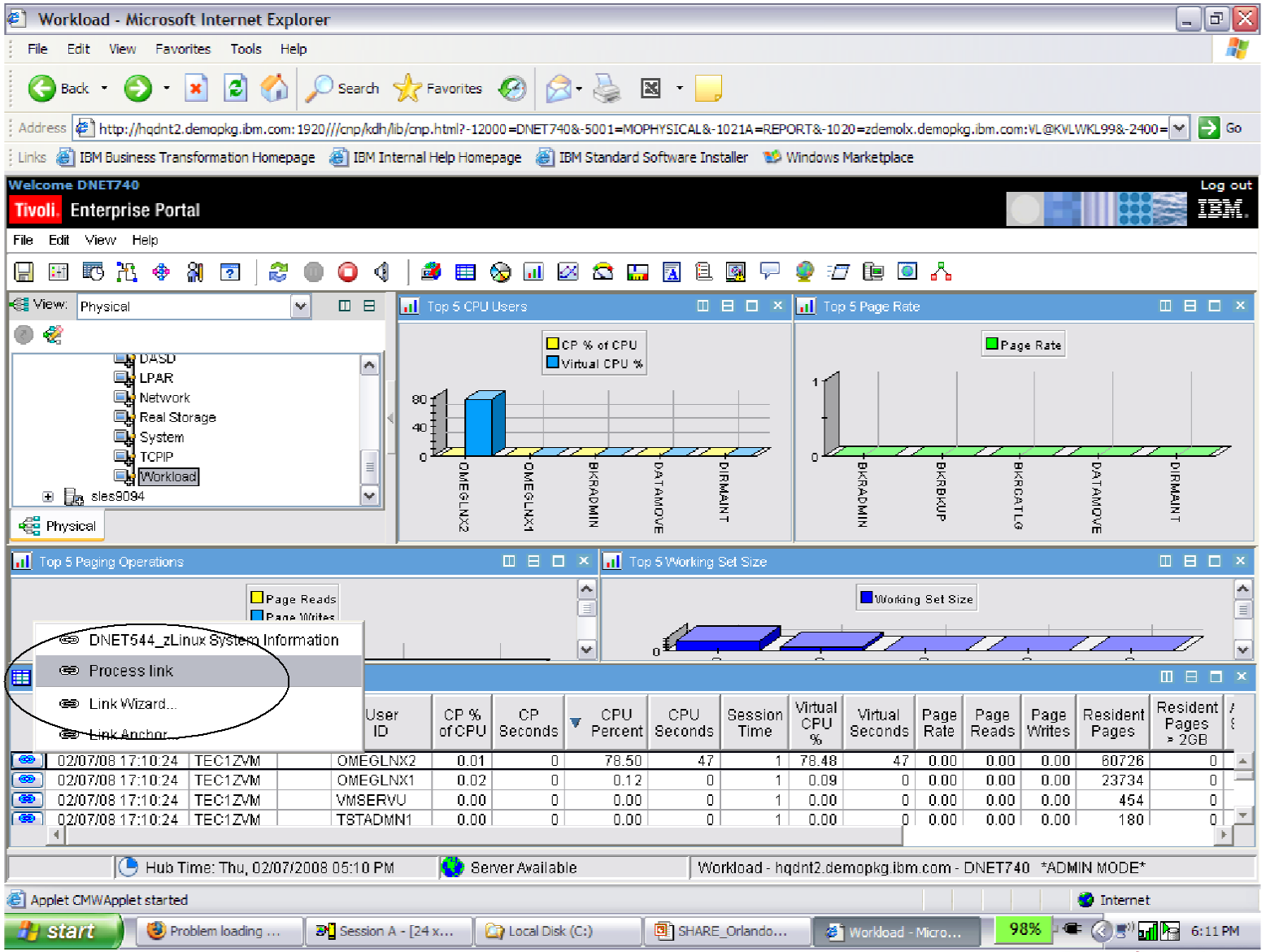
Scenario 3 — System Running Slowly

System is running slowly. Check Workload workspace to see if any particular user is hogging the CPU.



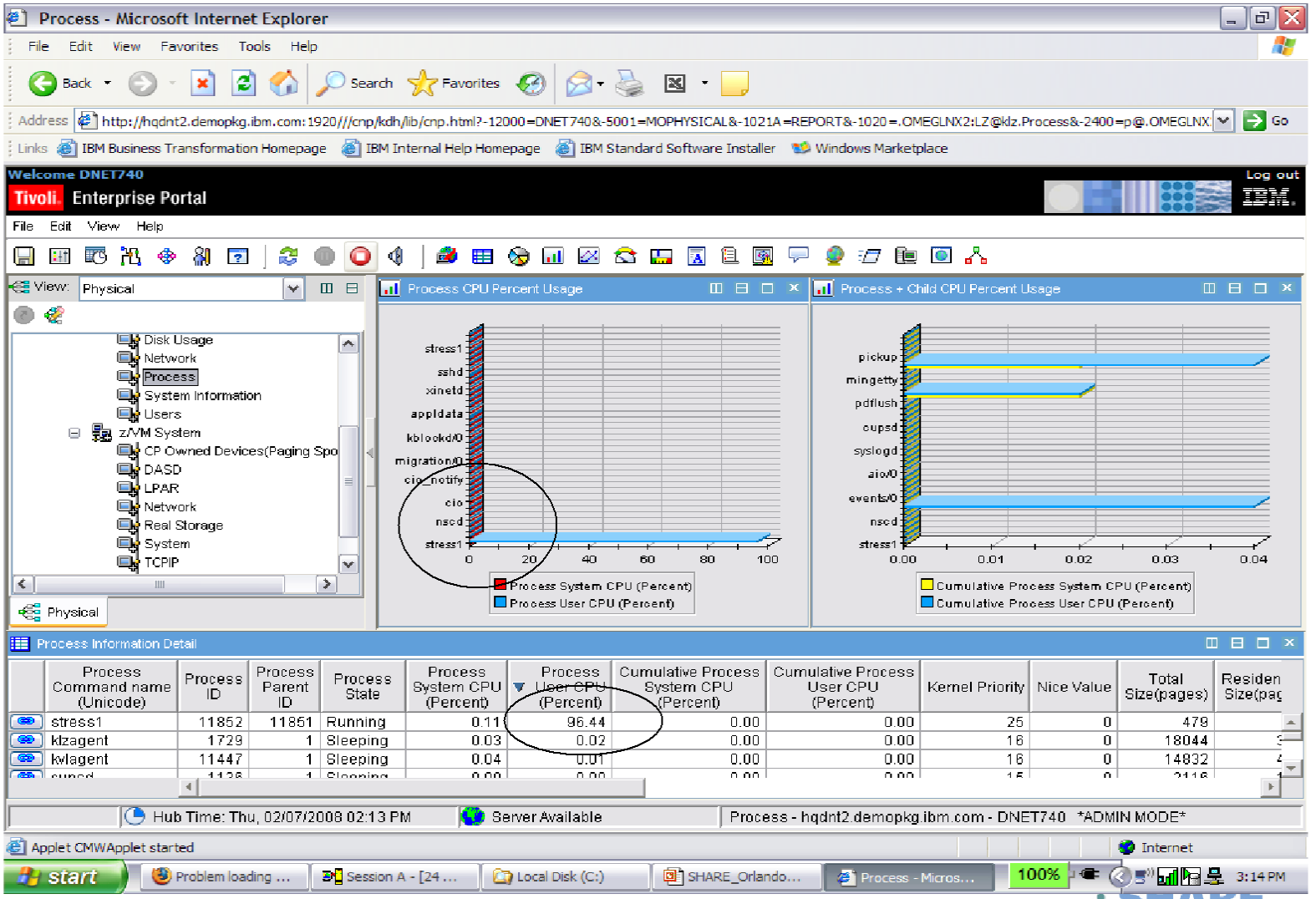
Scenario 3 — System Running Slowly (cont)

Predefined Link to take You directly To the Process workspace



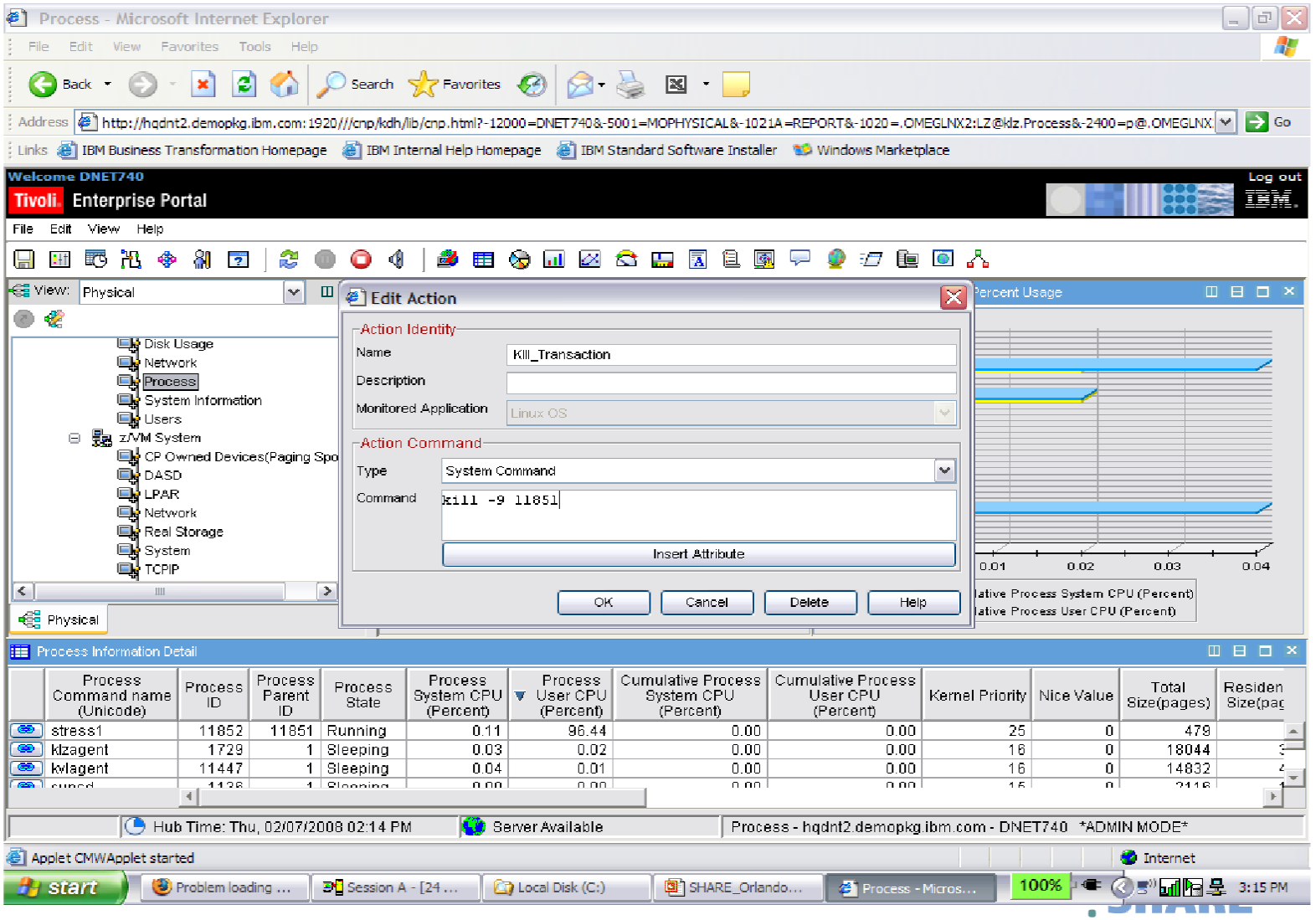
Scenario 3 — System Running Slowly (cont)

See if there is a process which is using too much CPU



Scenario 3 — System Running Slowly (cont)

You can issue a Take Action command to stop the offending process



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

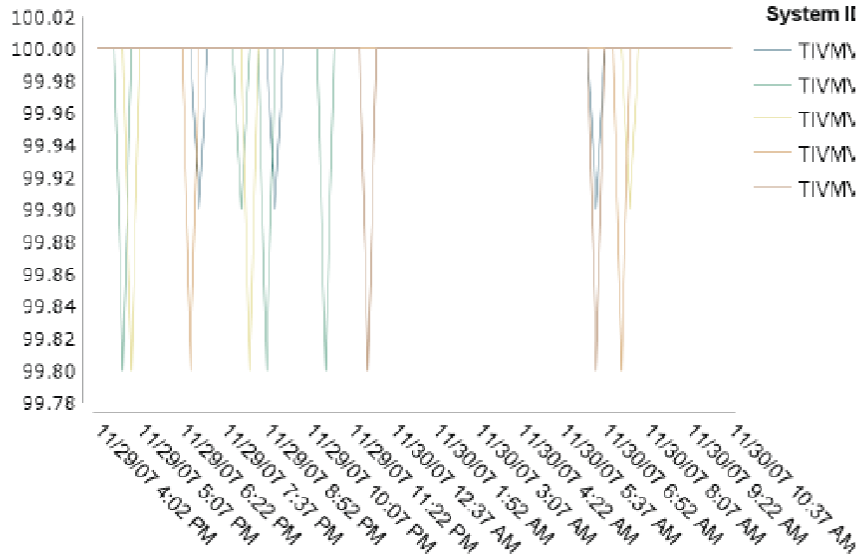
IBM

SHARE
Technology · Connections · Results

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

Tivoli

IBM

SHARE
Technology · Connections · Results

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

AGENDA

- Introduction
- Best Practices Monitoring Requirements
  - Virtual Linux and z/VM performance considerations
  - Don’t forget the hardware
  - Integration from hardware – systems – applications Persistent historical views
- Integrated Monitoring Approach
- Linux on z Health Checker

What is a health check?

- A program that...
  - Checks system configuration and status against best practices
  - Finds potential problems *before* they cause an outage or affect performance
  - Identifies settings that can be optimized
  - Reports findings through *exception messages*
  - *Health checks help you to maintain and increase health of your Linux instances*
  - Health checks provide you with expert knowledge
- It is not.....
  - A monitoring tool
  - Monitoring and health checking both gather data from a system and report problems. They both aim to ensure that important functions of a system are available and perform well. However, monitoring and health checking differ in some important aspects.

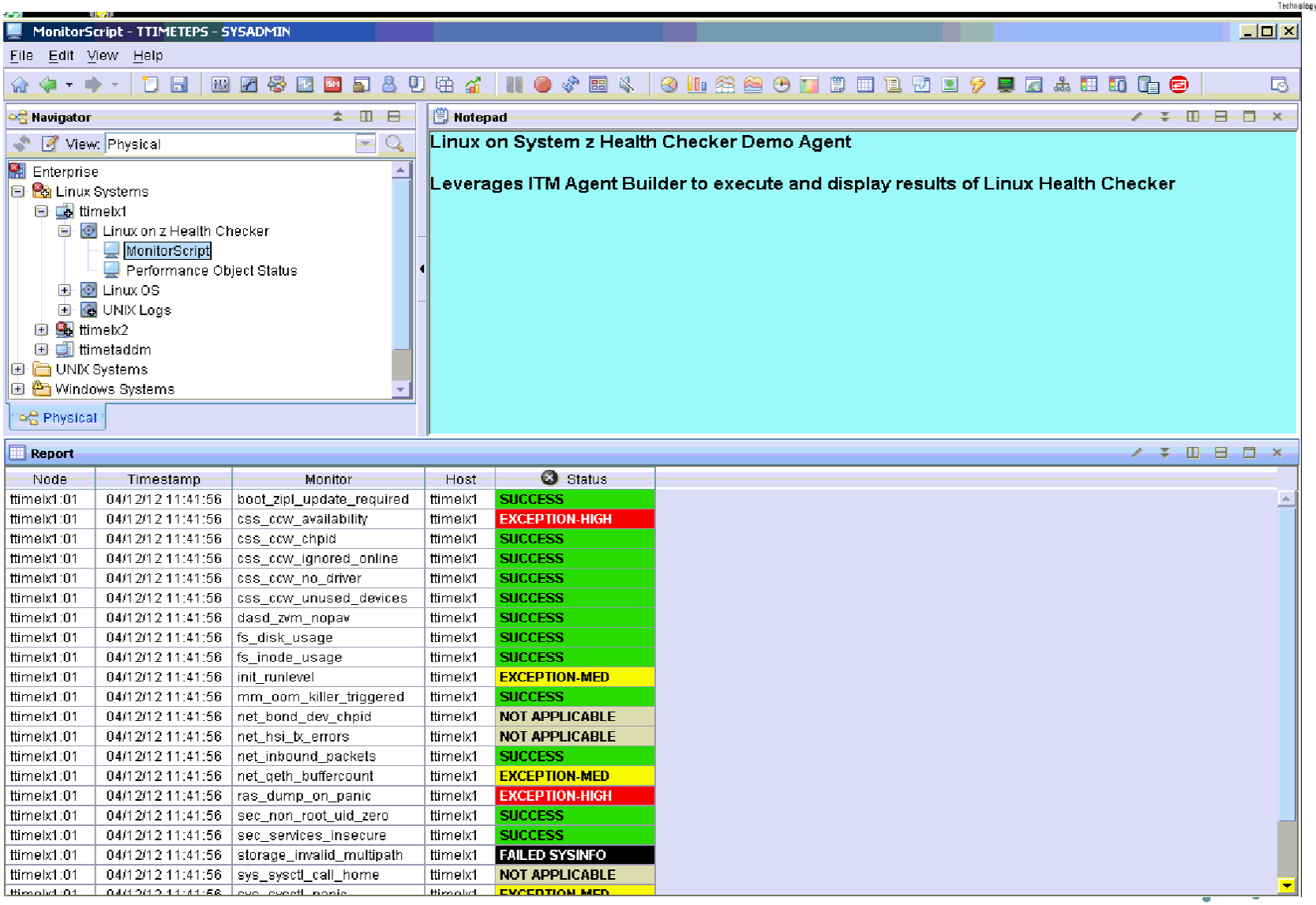
What is a health check?

- Examples of what health checks should find
  - Configuration errors
  - Deviations from best-practices
  - Hardware running in degraded mode
  - Unused accelerator hardware
  - Single point-of-failures
- The Linux Health Checker requires
  - Perl, version 5.8 or later
  - Additional perl modules that are part of standard Linux distributions
  - Each health check might have additional software requirements
- Ease of use
  - Simple setup: Install and run
  - Primary tasks easily accessible through command line interface

What's the difference between health checking and monitoring?

- Health checking is like a medical check-up
  - Analyzes current configuration and status
  - Identifies weaknesses
  - Presents you with actions to take *before* problems might occur
- Monitoring is like a long-term ECG
  - Observes selected data points in your system over time
  - Discovers trends and otherwise interpret the results
- Use health checking and monitoring in combination

Extending and Incorporating into your existing Systems Management Toolset



Node	Timestamp	Monitor	Host	Status
ttimelx1:01	04/12/12 11:41:56	boot_zipl_update_required	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	css_ccw_availability	ttimelx1	EXCEPTION-HIGH
ttimelx1:01	04/12/12 11:41:56	css_ccw_chpid	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	css_ccw_ignored_online	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	css_ccw_no_driver	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	css_ccw_unused_devices	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	dasd_zwn_nopav	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	fs_disk_usage	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	fs_inode_usage	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	init_runlevel	ttimelx1	EXCEPTION-MED
ttimelx1:01	04/12/12 11:41:56	mm_oom_killer_triggered	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	net_bond_dev_chpid	ttimelx1	NOT APPLICABLE
ttimelx1:01	04/12/12 11:41:56	net_hsi_tx_errors	ttimelx1	NOT APPLICABLE
ttimelx1:01	04/12/12 11:41:56	net_inbound_packets	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	net_qeth_buffercount	ttimelx1	EXCEPTION-MED
ttimelx1:01	04/12/12 11:41:56	ras_dump_on_panic	ttimelx1	EXCEPTION-HIGH
ttimelx1:01	04/12/12 11:41:56	sec_non_root_uid_zero	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	sec_services_insecure	ttimelx1	SUCCESS
ttimelx1:01	04/12/12 11:41:56	storage_invalid_multipath	ttimelx1	FAILED SYSINFO
ttimelx1:01	04/12/12 11:41:56	sys_sysctl_call_home	ttimelx1	NOT APPLICABLE

Health checks in version 1.2

61 checks in total (v1.0 had 25):

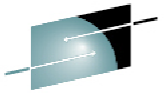
Check whether the recommended runlevel is used and set as default
Check whether the CPUs run with reduced capacity
Verify System z cryptographic hw support through CCA
Confirm that CPACF is used
Verify System z cryptographic hw support for PKCS#11 clear key [...]
Verify System z cryptographic hw support for PKCS#11 clear key [...]
Verify System z cryptographic hw support for PKCS#11 secure key [...]
Verify System z cryptographic hw support for PKCS#11 secure key [...]
Check whether the path to the OpenSSL library is configured correctly
Verify System z cryptographic hw support through an OpenSSL stack
Verify System z cryptographic hw support through an OpenSSL stack [...]
Identify I/O devices that are in use although they are on the exclusion list
Check for CHPIDs that are not available
Identify unusable I/O devices
Check for an excessive number of unused I/O devices
Identify I/O devices that are not associated with a device driver
Verify that the bootmap file is up-to-date
Identify standard DASD device nodes in the fstab file
Check if filesystems are skipped by filesystem check (fsck)
Check file systems for an adequate number of free inodes
Check for read-only filesystems
Verify that temporary files are deleted at regular intervals.
Check file systems for adequate free space
Confirm that automatic problem reporting is activated
Check if control program identification displays meaningful Linux names
Verify that syslog files are rotated
Check if swap space is available
Ensure memory usage is within the threshold
Identify bonding interfaces configured with single network interfaces
Identify bonding interfaces aggregating qeth interfaces with same CHPID
Ensure nameserver is listed with correct address
Check for an excessive error ratio for outbound HiperSockets traffic
Check the inbound network traffic for an excessive error or drop ratio
Identify qeth interfaces that do not have an optimal number of buffers
Identify network services that are known to be insecure
Ensure processes do not hog cpu time
Ensure the system is running with optimal load

Check the kernel message log for out-of-memory (OOM) occurrences
Ensure processes do not hog memory
Ensure that privilege dump is switched off
Ensure kdump is configured and running
Confirm that the dump-on-panic function is enabled
Ensure that panic-on-oops is switched on
Confirm that root logins are enabled for but restricted to secure terminals
Screen users with superuser privileges
Identify CDL-formatted DASD where metadata area used for storing data
Confirm 4K block size on ECKD DASD devices
Check Linux on z/VM for the "nopav" DASD parameter
Identify active DASD alias devices without active base device
Identify multipath setups that consist of a single path only
Identify multipath devices with too few available or many failed paths
Spot getty programs on the /dev/console device
Check for current console_loglevel
Detect terminals with multiple device nodes
Confirm that all available z/VM IUCV HVC terminals are enabled for logins
Identify idle terminals
Identify idle users
Identify unused terminals (TTY)
Check privilege classes of z/VM guest VMs on which Linux instances run

Checks by Component

Conclusion

- This presentation has highlighted the best practices for performance and availability management in managing z/VM and Linux on System z. To maximize the benefits of your shared environment, you must also consider the following factors:
  - Security (IBM RACF®, IBM Tivoli zSecure for RACF z/VM)
  - Directory Maintenance (DIRMAINT)
  - Backup and Recovery (IBM Backup and Restore Manager, Tivoli Storage Manager)
  - Automation (z/VM Operations Manager, System Automation for Multiplatform, System Automation Application Manager)
  - Accounting and Chargeback (Tivoli Usage and Accounting Manager)
  - Real resource management (Tape Manager, OSA/SF)
  - Virtual machine provisioning and management (z/VM 6.3 – Open Stack , IBM Tivoli Provisioning Manager, IBM Tivoli Service Automation Manager, CSL-Wave, SmartCloud Provisioning/Orchestration)



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

Managing z/VM and Linux Performance Best Practices

Jim Newell
IBM

August 15, 2013
Session Number 13498