IBM Systems & Technology Group

z/VM Memory Management

MVMUA
October 2009

IBM z/VM Performance Evaluation
Bill Bitner bitnerb@us.ibm.com

Trademarks

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml: AS/400, DBE, e-business logo, ESCO, eServer, FICON, IBM, IBM Logo, iSeries, MVS, OS/390, pSeries, RS/6000, S/30, VM/ESA, VSE/ESA, Websphere, xSeries, z/OS, zSeries, z/VM

The following are trademarks or registered trademarks of other companies

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation
Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
LINUX is a registered trademark of Linus Torvalds
UNIX is a registered trademark of The Open Group in the United States and other countries.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
Intel is a registered trademark of Intel Corporation
* All other products may be trademarks or registered trademarks of their respective companies.

NOTES:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

Any proposed use of claims in this presentation outside of the United States must be reviewed by local IBM country counsel prior to such use.

The information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

Agenda

- **Describe and discuss limits associated with Memory Management**
- **Look at recent changes**
- **Case Studies**

Memory

- **Central storage**
  - Supported central storage: 256 GB
  - Unsupported central storage:
    - 512 GB minus your HSA (z9 EC)
    - 1 TB (z10 EC)
  - The largest we ever managed was 440 GB
- **Expanded storage (architected): 16TB**
  - z/VM Limit: 128GB
  - See http://www.vm.ibm.com/perf/tips/storconf.html
- **Virtual machine size (hardware):**
  - Supported/Tested 1 TB (2^{40})
  - Hardware limits:
    - z10: 8TB
    - z9: 1TB
    - z900 and z990: 256GB

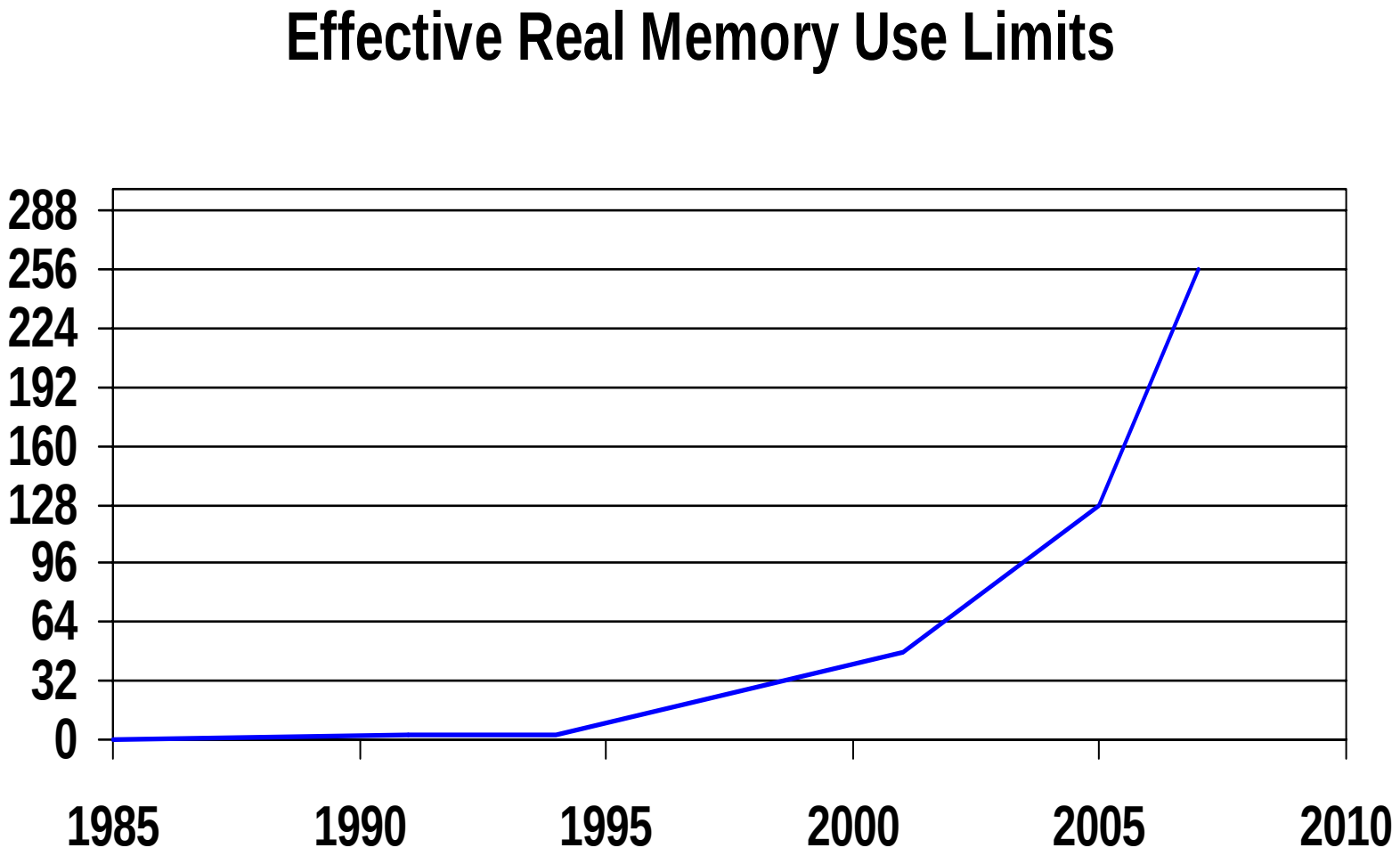
Memory

- **Active, or instantiated, guest real limit imposed by PTRM space limits (architected): 8 TB**
  - 16 4-GB PTRM spaces; each PTRM space can map 512 GB of guest real
- **Virtual to real ratio (practical): about 3:1**
  - Unless you really, really do your homework on your paging subsystem
  - Many factors come into play here:
    - Active:Idle Virtual machines
    - Workload/SLA sensitivity to delays
    - Exploitation of shared memory
    - Other

Memory

- **Paging space (architected) (optimal when <= 50% allocated):**
  - 11.2 TB for ECKD
  - 15.9 TB for Emulated FBA on FCP SCSI
- **Concurrent paging I/Os per paging volume: 1 for ECKD, >1 for EDEV (Have observed 1.6)**
- **System Execution Space (SXS) (architected): 2 GB**
  - For practical purposes it is 2GB, but there are structures in the space placed above 2GB
- **DCSS aggregate size (architected):**
  - Individual Segments up to 2047 MB
  - Segments can reside above 2GB, starting in z/VM 5.4.0
- **Minidisk Cache (architected): 8GB**
  - Practical 2GB

Memory Scaling



Page Slots: FCX146 AUXLOG

```
FCX146  Run 2007/09/06 14:00:28          AUXLOG
                                         Auxiliary Storage Utilization, by Time
From 2007/09/04 09:07:00
To   2007/09/04 10:00:00
For   3180 Secs 00:53:00
```

	<Page Slots>		<Spool Slots>		<Dump Slots>		<----- Spool Files ----->				<Average MLOAD>	
Interval	Total	Used	Total	Used	Total	Used	<-Created-->		<--Purged-->		Paging	Spooling
End Time	Slots	%	Slots	%	Slots	%	Total	/s	Total	/s	msec	msec
>>Mean>>	87146k	44	5409096	52	0	..	54	.02	54	.02	2.8	.8
09:08:00	87146k	44	5409096	52	0	..	1	.02	1	.02	2.3	.8
09:09:00	87146k	44	5409096	52	0	..	1	.02	1	.02	3.9	.8
09:10:00	87146k	44	5409096	52	0	..	1	.02	1	.02	3.6	.8
09:11:00	87146k	44	5409096	52	0	..	1	.02	1	.02	2.8	.8
09:12:00	87146k	44	5409096	52	0	..	1	.02	1	.02	2.9	.8

1. This system is using 44% of its page slots.

DASD I/O: FCX109 DEVICE CPOWNED

```
FCX109  Run 2007/09/06 14:00:28        DEVICE   CPOWNED                                          Page  152
                                       Load and Performance of CP Owned Disks
From 2007/09/04 09:07:00
To   2007/09/04 10:00:00                                                      CPU 2094-700
For   3180 Secs 00:53:00                                                       z/VM   V.5.3.0 SLU 0701
```

```
 Page / SPOOL Allocation Summary
 PAGE slots available       87146k          SPOOL slots available      5409096
 PAGE slot utilization          44%         SPOOL slot utilization          52%
 T-Disk cylinders avail.   .......          DUMP slots available             0
 T-Disk space utilization      ...%         DUMP slot utilization           ..%
```

```
____ .            .                    .     .     .     .     .     .     .     .     .     .     .     .
< Device Descr. ->                      <------------- Rate/s ------------->  User        Serv MLOAD Block
   %Used
            Volume Area  Area      Used  <--Page---> <--Spool-->         SSCH Inter Queue  Time  Resp  Page   for
Addr Devtyp Serial Type  Extent       %  P-Rds P-Wrt S-Rds S-Wrt Total  +RSCH feres Lngth /Page  Time  Size Alloc
F08B 3390   VS2P49 PAGE     0-3338   45    2.6   1.7   ...   ...   4.4    1.6     1   .02   2.4   2.4     7    89
F090 3390   VS2P69 PAGE     0-3338   45    2.7   1.6   ...   ...   4.3    1.6     1     0   2.7   2.7     7    84
```

V:R Ratio and Segment Tables: FCX113 UPAGE

```
FCX113  Run 2007/09/06 14:00:28        UPAGE                                                    Page  173
                                       User Paging Activity and Storage Utilization
From 2007/09/04 09:07:00                                                          VS2
To   2007/09/04 10:00:00                                                          CPU 2094-700   SN 2BFBD
For   3180 Secs 00:53:00                                                          z/VM   V.5.3.0 SLU 0701
```

	Data	<--- Paging Activity/s --->			<--- Number of Pages --->												
	Spaces	<Page Rate>		Page	<--Page Migration-->						<-Resident->		<--Locked-->				Stor	Nr of
Userid	Owned	Reads	Write	Steals	>2GB>	X>MS	MS>X	X>DS	WSS	Resrvd	R<2GB	R>2GB	L<2GB	L>2GB	XSTOR	DASD	Size	Users
>System<	.0	1.7	1.1	4.1	.0	2.4	3.7	1.4	122050	0	2347	106962	6	24	12240	179131	1310M	212
DATAMOVF	.0	.0	.0	.0	.0	.0	.1	.0	13	0	0	0	0	0	483	254	32M	
DATAMOVA	.0	.0	.0	.0	.0	.5	.5	.0	147	0	0	0	0	0	220	368	32M	
DATAMOVB	.0	.0	.0	.0	.0	.6	.6	.0	192	0	0	0	0	0	220	366	32M	
DATAMOVC	.0	.0	.0	.0	.0	.6	.6	.0	191	0	0	0	0	0	220	369	32M	
DATAMOVD	.0	.0	.0	.0	.0	.6	.6	.0	189	0	0	0	0	0	220	362	32M	

1. Resident Guest Pages = (2347 + 106962) * 212 = 88.3 GB
2. V:R = (1310 MB * 212) / 91 GB = 2.98
3. Segment Table Pages: hard to say. Worst case (all 8 GB guests):
   212 guests * (4 ST/guest * 4 pg/ST) = 13 MB

PTRM Space: FCX134 DSPACESH

```
FCX134  Run 2007/09/06 14:00:28          DSPACESH
                                 Shared Data Spaces Paging Activity

From 2007/09/04 09:07:00
To   2007/09/04 10:00:00                                                    CPU 2094-700
For   3180 Secs 00:53:00                                                    z/VM   V.5.3.0 SLU
   0701
```

```
   _
```

```
 _____                          .     .     .     .     .     .     .     .     .     .     .     .     .     .     .
                                <--------- Rate per Sec. ---------> <------------------Number of Pages------------------>
 Owning                                                                    <--Resid--> <-Locked--> <-Aliases->
 Userid     Data Space Name      Pgstl Pgrds Pgwrt X-rds X-wrt X-mig Total Resid R<2GB  Lock L<2GB Count Lockd XSTOR  DASD
 >System<   --------              .026  .016  .001  .015  .026  .000  103k  1208    51     0     0     0     0    34  4981
 SYSTEM     FULL$TRACK$CACHE$1    .000  .000  .000  .000  .000  .000  524k     0     0     0     0     0     0     0     0
 SYSTEM     ISFCDATASPACE         .000  .000  .000  .000  .000  .000  524k   113     8     8     8   113   100     0    27
 SYSTEM     PTRM0000             4.257  .492  .442 3.957 4.036  .000 1049k  386k 15885     0     0     0     0  5195  683k
 SYSTEM     REAL                  .000  .000  .000  .000  .000  .000   24M     0     0     0     0     0     0     0     0
 SYSTEM     SYSTEM                .080  .001  .034  .079  .080  .000  524k    45    10     0     0    44     0    47  510k
```

1. PTRM space = (386,000 + 15885) = 401,885 = 1.53 GB
   (NB: this is z/VM 5.3)

Real Memory: FCX254 AVAILLOG

```
FCX254  Run 2007/09/06 14:00:28        AVAILLOG                                                    Page  190
                                       Available List Management, by Time
From 2007/09/04 09:07:00
To   2007/09/04 10:00:00                                                          CPU 2094-700
For   3180 Secs 00:53:00                                                          z/VM   V.5.3.0 SLU 0701
```

	<---------- Available List Management ---------->															
	<---- Thresholds ----->		<---------- Page Frames ---------->						<-Times->		<----------- Replenishment ------------>					Perct
Interval	<---Low--->	<--High--->	<Available>		<Obtains/s>		<Returns/s>		<-Empty->		<--Scan1-->		<--Scan2-->		<-Em-Scan->		Scan	Emerg
End Time	<2GB	>2GB	<2GB	>2GB	<2GB	>2GB	<2GB	>2GB	<2GB	>2GB	Compl	Pages	Compl	Pages	Compl	Pages	Fail	Scan
>>Mean>>	20	7588	5820	13388	5130	7678	323.3	857.4	311.5	844.8	0	0	27	1381k	63	1380k	58	84490	82	88
09:08:00	20	7680	5820	13480	6665	15122	353.3	838.5	353.2	1007	0	0	0	43091	3	26491	0	0	3	100
09:09:00	20	7680	5820	13480	3986	5496	163.1	640.2	108.9	442.7	0	0	1	14528	0	0	0	0	0	0
09:10:00	20	7681	5820	13481	6622	9542	222.4	556.1	257.0	598.3	0	0	0	30103	2	8868	0	0	1	100
09:11:00	20	7681	5820	13481	4982	6710	292.1	615.2	248.8	533.6	0	0	0	21246	0	8547	1	3989	1	100
09:12:00	20	7681	5820	13481	4769	1560	284.9	946.9	254.4	830.0	0	0	0	18253	0	22438	2	656	1	100

1. Pct ES = 88% generally this system is tight on storage
2. Scan fail >0 generally this system is tight on storage
3. Times Empty = 0 this indicates it isn't critical yet

SXS Space: FCX261 SXSAVAIL

```
FCX261  Run 2007/09/06 14:00:28          SXSAVAIL                                                    Page  261
                                         System Execution Space Page Queues Management
From 2007/09/04 09:07:00
To   2007/09/04 10:00:00                                                              CPU 2094-700
For   3180 Secs 00:53:00                                                              z/VM   V.5.3.0 SLU 0701
```

	<-- Backed <2GB Page Queue -->					<-- Backed >2GB Page Queue -->					<---------- Unbacked Page Queue ----------								
Interval	Avail	<-Pages/s-->		<Preferred>		Avail	<-Pages/s-->		<Preferred>		Avail	<-Pages/s-->		<Preferred>		<---- Replenishment ---->			
End Time	Pages	Taken	Return	Used	Empty	Pages	Taken	Return	Used	Empty	Pages	Taken	Return	Used	Empty	Thres	Att/s	Stolen	MinPgs
>>Mean>>	26	.513	.509	.513	.000	3	1.798	1.804	1.798	4.114	466946	130.3	130.1	126.2	.000	128	.000	128	...
09:08:00	26	.483	.383	.483	.000	0	1.650	1.650	1.650	3.667	467829	128.2	127.3	124.5	.000	128	.000	128	...
09:09:00	26	.500	.500	.500	.000	0	.583	.583	.583	3.067	465679	120.8	84.98	117.8	.000	128	.000	128	...
09:10:00	27	.517	.533	.517	.000	0	1.183	1.183	1.183	4.000	467657	109.1	142.1	105.1	.000	128	.000	128	...
09:11:00	27	.517	.517	.517	.000	0	1.633	1.633	1.633	2.917	467632	137.2	136.8	134.3	.000	128	.000	128	...
09:12:00	29	.450	.483	.450	.000	0	2.000	2.000	2.000	3.383	467654	129.9	130.2	126.5	.000	128	.000	128	...
09:13:00	27	.517	.483	.517	.000	0	2.483	2.483	2.483	3.550	467698	139.3	140.0	135.7	.000	128	.000	128	...
09:14:00	25	.550	.517	.550	.000	0	2.000	2.000	2.000	2.750	465651	119.0	84.92	116.3	.000	128	.000	128	...

1. How we touch guest pages: (1) 64-bit; (2) AR mode; (3) SXS.
2. There are 524,288 pages in the SXS.
3. This system has 466,000 SXS pages available on average.

```
FCX178  Run 2008/04/15 10:00:22          MDCSTOR            Page   76
             Minidisk Cache Storage Usage, by Time
From 2008/04/15 09:47:11
To   2008/04/15 10:00:11                        CPU 2084-320   SN 17F2A
For    780 Secs 00:13:00                        z/VM   V.5.3.0 SLU 0000

<---------------- Main Storage Frames ----------------->
 Interval          <--Actual--->    Min     Max  Page    Steal
 End Time    Ideal    <2GB   >2GB    Set     Set Del/s Invokd/s   Bias
 >>Mean>>    5839k   82738  1354k      0   7864k     0     .000   1.00
 09:57:41    5838k  119813  1932k      0   7864k     0     .000   1.00
 09:58:11    5838k  119813  1932k      0   7864k     0     .000   1.00
 09:58:41    5838k  119825  1932k      0   7864k     0     .000   1.00
 09:59:11    5838k  119825  1932k      0   7864k     0     .000   1.00
 09:59:41    5838k  119825  1932k      0   7864k     0     .000   1.00
 10:00:11    5838k  119837  1932k      0   7864k     0     .000   1.00
```

- Xstore not used for this configuration so edited out from report.
- Add up the pages in Main Storage and you get ~8GB

```
FCX134  Run 2008/04/15 10:00:22          DSPACESH
                                  Shared Data Spaces Paging Activity

From 2008/04/15 09:47:11
To   2008/04/15 10:00:11
For    780 Secs 00:13:00                 This is a performance report for system XYZ
```

Owning Userid	Data Space Name	Users Permt	Number of Pages Total	<--Resid--> Resid	R<2GB	<-Locked--> Lock	L<2GB	<-Aliases-> Count	Lockd	XSTOR	DASD
>System<	--------	0	1507k	5665	101	0	0	100	0	0	0
SYSTEM	FULL$TRACK$CACHE$1	0	524k	0	0	0	0	0	0	0	0
SYSTEM	FULL$TRACK$CACHE$2	0	524k	0	0	0	0	0	0	0	0
SYSTEM	FULL$TRACK$CACHE$3	0	524k	0	0	0	0	0	0	0	0
SYSTEM	FULL$TRACK$CACHE$4	0	524k	0	0	0	0	0	0	0	0
SYSTEM	ISFCDATASPACE	0	524k	0	0	0	0	0	0	0	0
SYSTEM	PTRM0000	0	1049k	44489	0	0	0	0	0	0	0
SYSTEM	REAL	0	7864k	0	0	0	0	0	0	0	0
SYSTEM	SYSTEM	0	524k	805	787	0	0	800	0	0	0
SYSTEM	VIRTUAL$FREE$STORAGE	0	524k	23	23	0	0	0	0	0	0

- You'll see the address spaces used for MDC (track cache)
- Values here are zero for page counts, ignore.

Reorder Processing - Background

- ***Page reorder* is the process in z/VM of managing user frame owned lists as input to demand scan processing.**
  - It includes resetting the HW reference bit.
  - Serializes the virtual machine (all virtual processors).
  - In all releases of z/VM
- **It is done periodically on a virtual machine basis.**
- **The cost of reorder is proportional to the number of resident frames for the virtual machine.**
  - Roughly 130 ms/GB resident
  - Delays of ~1 second for guest having 8 GB resident
  - This can vary for different reasons +/- 40%

Reorder Processing - Diagnosing

- **Performance Toolkit**
  - Check resident page fields (“R<2GB” & “R>2GB”) on FCX113 UPAGE report
    - Remember, Reorder works against the resident pages, not total virtual machine size.
  - Check Console Function Mode Wait (“%CFW”) on FCX114 USTAT report
    - A virtual machine may be brought through console function mode to serialize Reorder. There are other ways to serialize for Reorder and there are other reasons that for CFW, so this is not conclusive.
- **REORDMON**
  - Available from Bill Bitner now and the VM Download Page http://www.vm.ibm.com/download/packages/
  - Works against raw MONWRITE data for all monitored virtual machines
  - Works in real time for a specific virtual machine
  - Provides how often Reorder processing occurs in each monitor interval

REORDMON Example

Userid	Num. of Reorders	Average Rsdnt(MB)	Average Ref'd(MB)	Reorder Times	
LINUX002	2	18352	13356	13:29:05	14:15:05
LINUX001	1	22444	6966	13:44:05	
LINUX005	1	14275	5374	13:56:05	
LINUX003	2	21408	13660	13:43:05	14:10:05
LINUX007	1	12238	5961	13:51:05	
LINUX006	1	9686	4359	13:31:05	
LINUX004	1	21410	11886	14:18:05	

Reorder Processing - Mitigations

- **Try to keep the virtual machine as small as possible.**
- **Virtual machines with multiple applications may need to be split into multiple virtual machines with fewer applications.**
- **Known requirement at IBM to bring relief in this area.**

CMM Futures

- **CMM 2 (aka CMMA, MEMASSIST)**
  - Linux support limited to SLES 10
  - Off by default at the Linux Level
  - Check "cmma=on" option with "cat /proc/cmdline" to see if in use.
- **CMM 2 Lite**
  - Form of CMMA that only uses the "Stable" and "Unused" states (isolated to architecture specific code).
  - Direction of future distributions
- **CMM 1**
  - Can be used via VMRM support
  - Originally thought to be more of a tactical solution with CMM 2 being the strategic solution
  - Service to improve: VM64439
  - Expect more investigation in this area in future.
- **For more performance information, see:**
  - http://www.vm.ibm.com/perf/reports/zvm/html/530cmm.html

Virtual Machines Not Going Dormant - Background

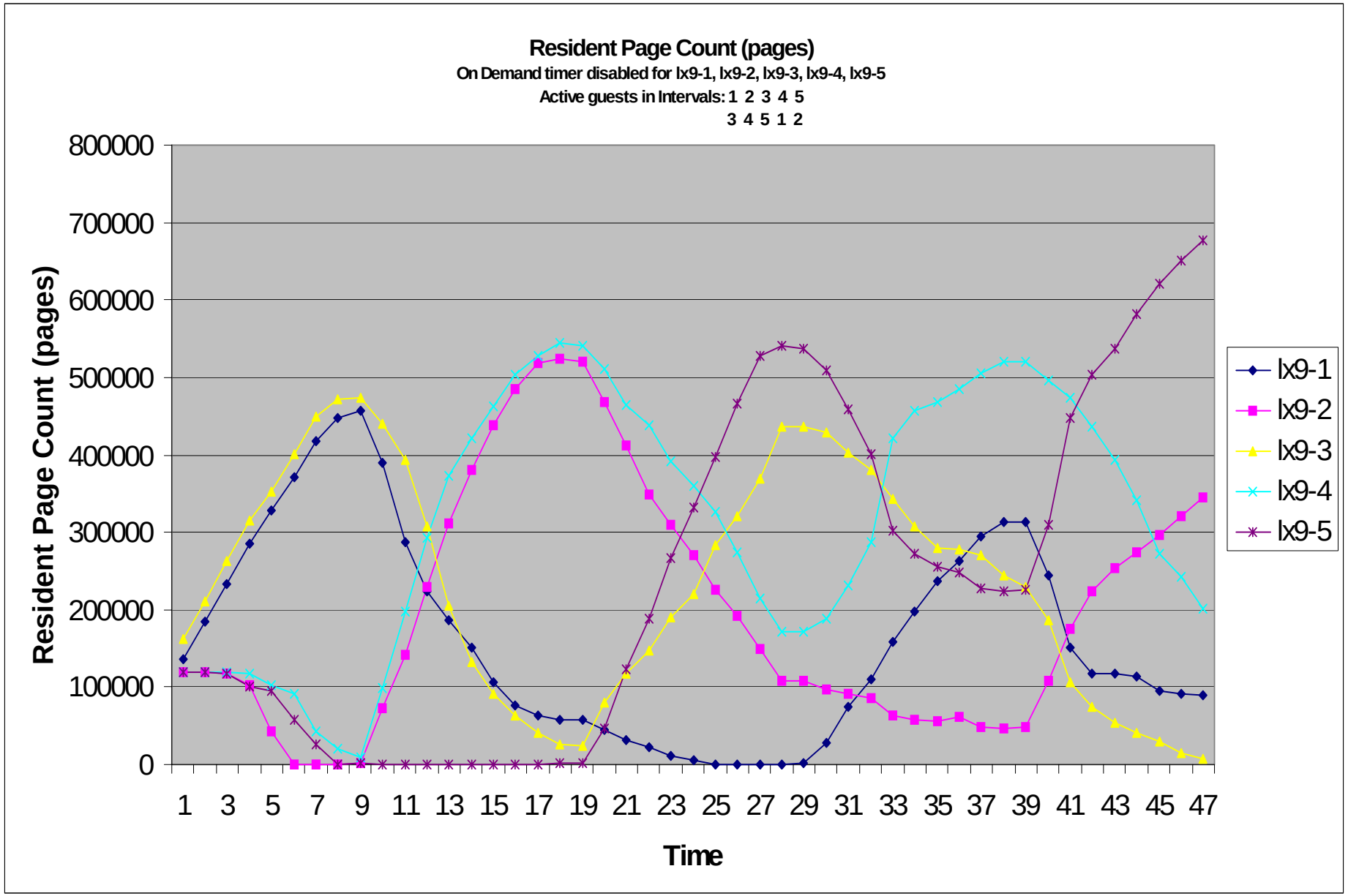
- **z/VM has concept of going dormant, dropping from dispatch list.**
- **Has a “test idle” grace period of 300ms**
- **Activity, such as timer requests, occurring within the 300ms prohibit the virtual machine from going true dormant and dropping from dispatch list**
- **Memory Management Demand Scan processing uses information on active and idle (idle from VM perspective) in algorithms to decide which pages to steal.**
- **Demand Scan goes through a series of three passes (1, 2, Emergency)**
  - Selects pages of different characteristics in each pass
  - Pages from active virtual machines are in later passes

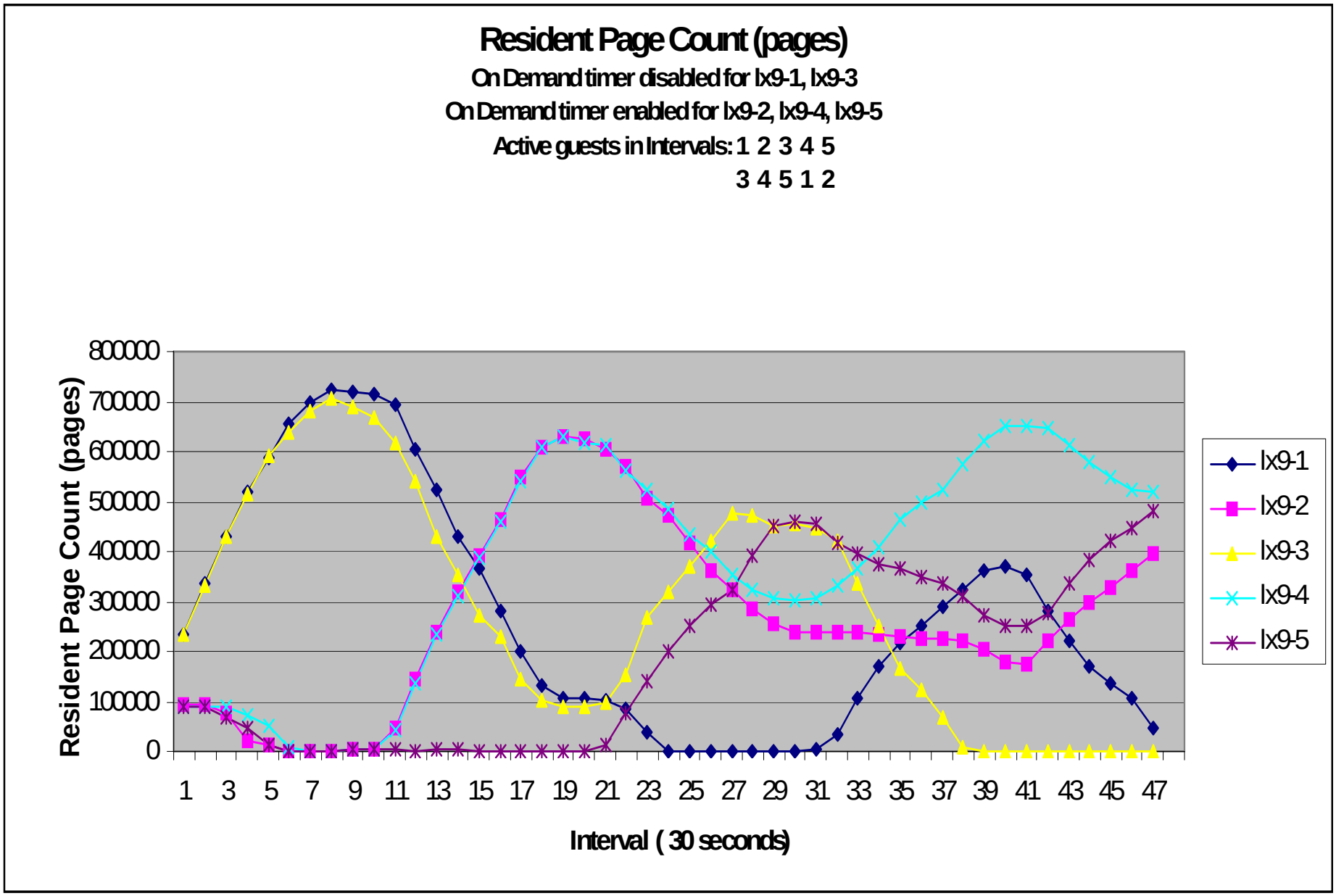
Virtual Machines Not Going Dormant - Problem

- **Some environments and configurations may result in Linux virtual machine never going "dormant" even when it is "not active" from a customer perspective.**
- **This is a problem that IBM is exploring.**
  - Some software has been corrected over time
- **FUD: A virtual machine that never goes dormant prevents z/VM from taking pages from it.**
- **Truth: Pages can be stolen, but the memory management is just not as intelligent about it.**

Example Measurements

- **Series of Linux virtual machines running Apache with simple web serving workload**
- **Two virtual machines at a time are active**
- **Rotate through which two are active**
- **On Demand Timer setting manipulated**
  - Disabled = Off
    - Only wakes up when needed
    - Therefore drops from dispatch list going true dormant





Case Study: Why Doesn't My System Page Faster

Question from Customer

- **“z/VM pages extremely slowly”**
- **Inactive Linux guest is paged in at only about 1000 pages per second**
- **12 3390-9 paging packs, 2 LCUs, with 6 FICON chpids**
- **During busy periods of running 30 guests, he sees 6000 pages per second**
- **Customer thinks this single guest should page in much faster**
- **He devised a 300 MB thrasher that reproduced the behavior**
- **He sent us lots of charts and graphs**
- **We asked for MONWRITE data**

Customer Sent MONWRITE Data

- **User LIN102 is running the 300 MB thrasher**
- **It touched 64,000 pages in 61 seconds (1049/sec)**
- **The interesting time period is 15:12:30 to 15:13:20**
- **He used MONITOR SAMPLE 10 SEC (brilliant!)**
- **Ran his data through PERFKIT BATCH**
- **Looked at some interesting reports for that period**

User Configuration

FCX226 UCONF – user configuration report

					<-------- Share -------->						No	Stor	
		Virt	Mach	Stor			%	Max.	Max.	Max.	QUICK	MDC	Size	Reserved
Userid	SVM	CPUs	Mode	Mode	Relative	Absolute	Value/%	Share	Limit	DSP	Fair	(MB)	Pages
LIN102	No	1	EME	V=V	100	...	...	..	..	No	No	768M	0

Virtual uniprocessor with one process (thread) running the memory initializer.

Implications:

1. Memory initializer will touch pages serially.
2. Page faults will happen serially.

Activity on Paging DASD

FCX108 INTERIM DEVICE 15:12:40 to 15:12:51

<-- Device Descr. -->			Mdisk	Pa-	<-Rate/s->	<------- Time (msec) ------->					Req.	<Percent>		SEEK
Addr	Type	Label/ID	Links	ths	I/O	Avoid	Pend	Disc	Conn	Serv	Resp	CUWt	Qued	Busy	READ	Cyls
9F11	3390	VSPPG8 CP	0	6	25.5	.0	.2	.0	3.9	4.1	4.1	.0	.0	10	0	131
A062	3390	VSPPG5 CP	0	6	25.0	.0	.2	.0	3.3	3.5	3.5	.0	.0	9	100	2580
A02D	3390	VSPPG3 CP	0	6	27.4	.0	.2	.1	3.1	3.4	3.4	.0	.0	9	100	505
9F41	3390	VSPPGB CP	0	6	29.8	.0	.2	.0	3.0	3.2	3.2	.0	.0	10	100	753
A03D	3390	VSPPG2 CP	0	6	35.4	.0	.2	.0	2.9	3.1	3.1	.0	.0	11	100	832
9F01	3390	VSPPG7 CP	0	6	38.0	.0	.2	.0	2.8	3.0	3.0	.0	.0	11	0	1174
9F5A	3390	VSAPAG CP	0	6	40.9	.0	.2	.0	2.7	2.9	2.9	.0	.0	12	100	33
A05D	3390	VSPPG6 CP	0	6	38.9	.0	.2	.0	2.7	2.9	2.9	.0	.0	11	100	1446
A01B	3390	VSPPG4 CP	0	6	32.3	.0	.2	.0	2.5	2.7	2.7	.0	.0	9	100	2670
9F21	3390	VSPPG9 CP	0	6	45.6	.0	.2	.0	2.2	2.4	2.4	.0	.0	11	0	0
9F51	3390	VSPPGC CP	0	6	48.5	.0	.2	.0	2.2	2.4	2.4	.0	.0	12	100	2971
		TOTAL			387.3									115		

Eleven paging devices:

1. Each in the neighborhood of 10% busy, all reads
2. Each showing response time of about 3.1 msec

Who Else is Doing Paging Activity?

FCX113 UPAGE

```
           Data  <--------- Paging Activity/s ---------->
         Spaces  <Page Rate>   Page  <--Page Migration-->  Nr of
Userid    Owned  Reads Write Steals  >2GB> X>MS MS>X X>DS  Users
>System<     .0    2.3   1.6    7.2     .0  4.6  6.3  1.7     44

User Data:
LIN102       .0   75.8    .0     .0     .0 35.2  4.5   .0
```

44 * 2.3 = 101 pages read/sec altogether.

LIN102 accounts for 76% of this, 76 pages read/sec.

What We Know So Far

- **Each paging I/O takes about 3.1 msec**
- **One single-threaded application in one guest is responsible for most of the paging I/Os**
- **This means we should see about (1000/3.1) = 323 SSCH ops for paging per second**
- **We actually saw 387/sec, but remember other guests are paging slightly**
- **Because one single-threaded guest is responsible for most of the paging I/O, the paging device utilizations should add to about 100%**
- **They actually add to 115%, but remember other guests are paging slightly**

What Did We Tell The Customer?

- **LIN102's page reading speed is limited by its single-threaded nature and the speed of the paging DASD.**
- **Your system pages at higher rates when 30 guests are running because with multiple guests you can generate concurrent page reads. You have multiple paging exposures too and so you can parallelize paging I/O.**
- **Your 11 paging exposures look like they could support (1100%/115%) = 9.5 such thrashers concurrently.**
- **But from FCX109 DEVICE CPOWNED, we see your page space is about 15% full so I wouldn't try more than four of them at once.**

Something Interesting About LIN102

```
FCX163  Run 2008/05/19 12:18:57          UPAGELOG LIN102
                                         User Paging Activit

From 2008/05/15 15:10:10
To   2008/05/15 15:15:50
For    340 Secs 00:05:40
```

Page Data Log for User LIN102

	Data	<--------- Paging Activity/s ---------->						
Interval	Spaces	<Page Rate>		Page	<--Page Migration-->			
End Time	Owned	Reads	Write	Steals	>2GB>	X>MS	MS>X	X>DS
15:12:40	0	437	.0	.0	.0	116	4.2	.0
15:12:50	0	534	.0	.0	.0	167	.6	.0
15:13:00	0	440	.0	.0	.0	342	37.7	.0
15:13:10	0	313	.0	.0	.0	288	.2	.0
15:13:20	0	473	.0	.0	.0	246	3.4	.0
Avg		439				232		

Thrasher touched 1049/sec altogether.

1. 439/sec read from disk
2. 232/sec read from XSTORE
3. 378/sec resident

A Note on User States

```
FCX164  Run 2008/05/19 12:18:57          USTATLOG LIN102
                                         User Wait States

From 2008/05/15 15:10:10
To   2008/05/15 15:15:50
For    340 Secs 00:05:40
```

Wait State Data Log for User LIN102

Interval End Time	%ACT	%RUN	%CPU	%LDG	%PGW	%IOW	%SIM	%TIW	%CF
15:12:30	100	0	0	0	100	0	0	0	
15:12:40	100	0	0	0	100	0	0	0	
15:12:50	100	0	0	0	100	0	0	0	
15:13:00	100	0	0	0	100	0	0	0	
15:13:10	100	0	0	0	100	0	0	0	
15:13:20	100	0	0	0	100	0	0	0	

Customer said this means LIN102 “is in page wait 100% of the time”.
This is not correct.
It means 100% of the times we looked, LIN102 was in a page wait.
We looked only once every two seconds (FCX149 MONSET).
After all, LIN102 was also *touching* pages.

Case Study Summary

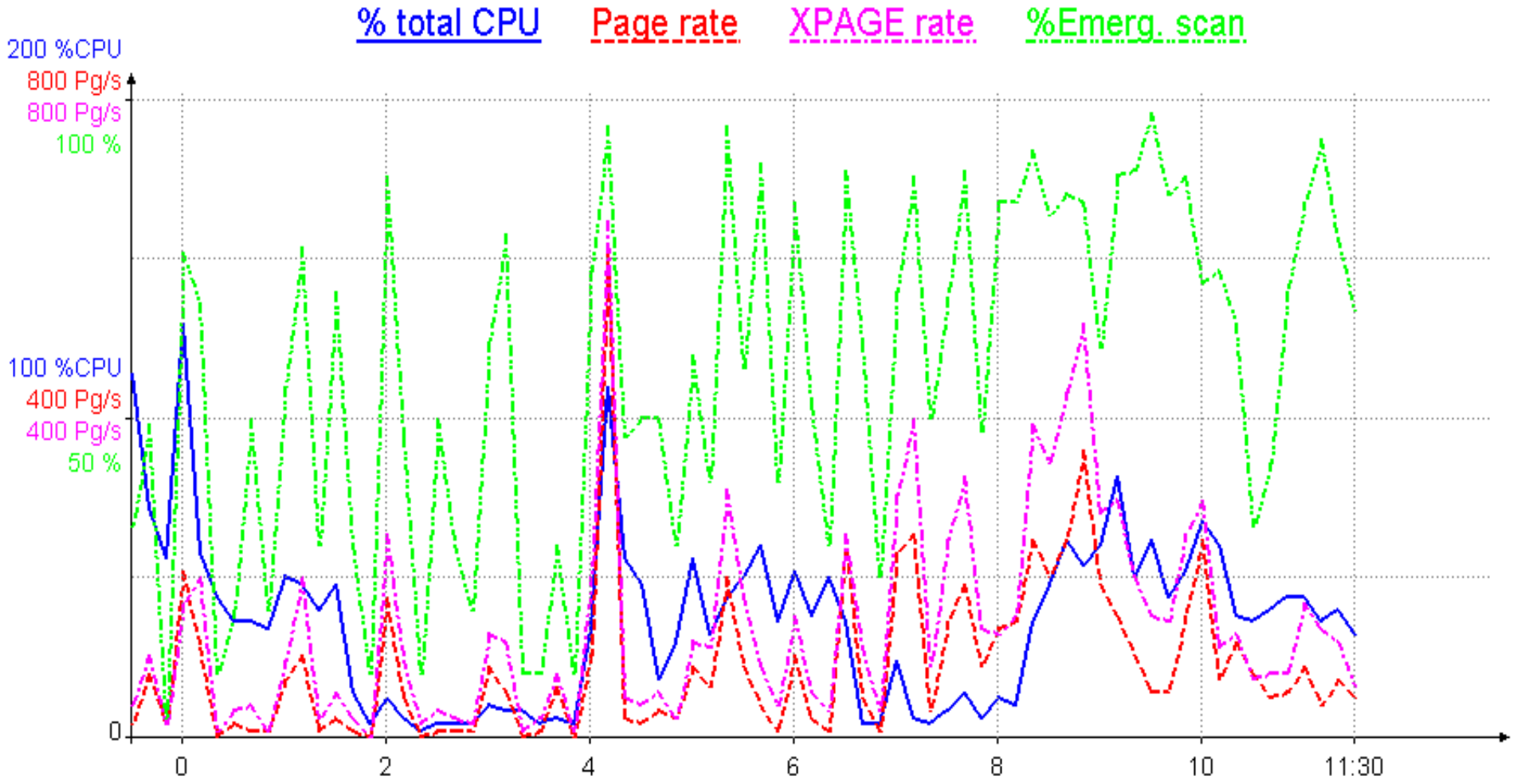
- **Customer became absorbed with z/VM measurements but forgot what his workload does**
- **Knowledge of the workload's behavior is crucial in understanding why the system performs the way it does**
- **Customer was very good at collecting raw monitor data appropriate for the diagnosis task at hand**
- **Fun question that was not too difficult to answer**

Case Study: Emergency Scan

Question from Customer

- **My system seems to have a high percentage of emergency scan**
- **Application performance doesn’t seem bothered**
- **Should I be worried?**

Graph from Customer



Source data: Storage

Finding a Memory Frame

- **Pass 1: tries to be friendly to dispatched users**
  - Unreferenced shared address space pages
  - Long-term-dormant users
  - Eligible-list users
  - Dispatch-list users' unreferenced pages down to WSS
- **Pass 2: a little more aggressive… like pass 1 except:**
  - Avoids shared address spaces
  - Will take from dispatch-list users down to their SET RESERVE
- **Emergency scan: anything we can find**
- **Bit of a misnomer**
- **Want to know more? Read the prologue of HCPALD**

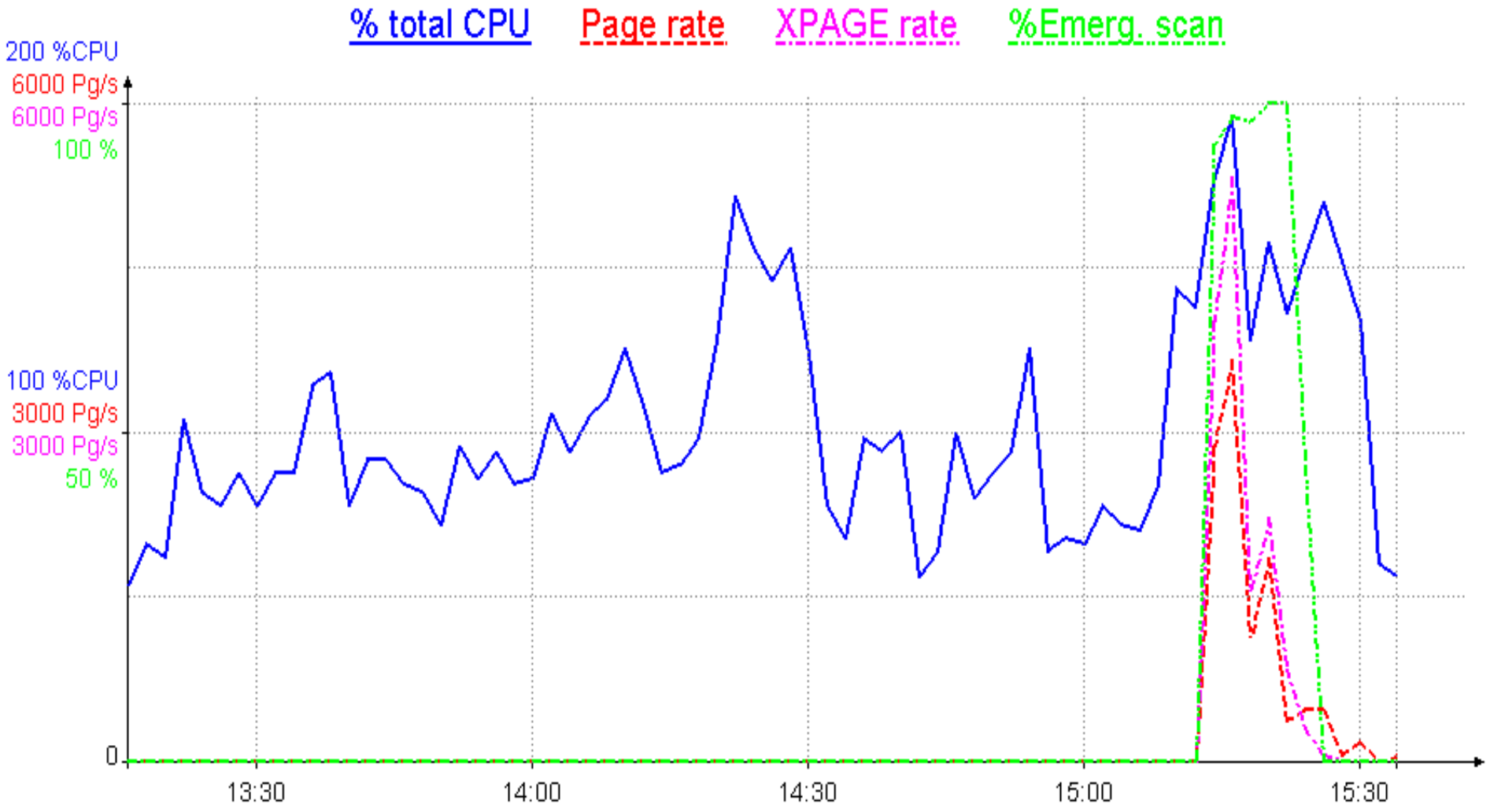
Is Emergency Scan A Sign of Duress?

- **Not alone, no.**
- **Evaluate some other things too.**
  - Are free frame lists routinely zero? (FCX254 AVAILLOG)
  - Is system T/V high? (FCX225 SYSSUMLG)
  - Are we spinning significantly on any locks? (FCX265 LOCKLOG)
  - Does USTAT show users in page wait? (FCX114 USTAT)
  - Is an eligible list forming? (FCX100 CPU)
  - Are MDC hits satisfactory? (FCX103 STORAGE, FCX108 DEVICE)
  - Do you have plenty of SXS space? (FCX264 SXSUTIL)
  - Is DASD page rate > XSTORE page rate? (FCX143 PAGELOG)
  - Are there queues at paging DASD? (FCX109 DEVICE CPOWNED)
  - Is paging MLOAD OK? (FCX109 DEVICE CPOWNED)
  - Is paging blocking factor OK? (FCX103 STORAGE)
  - Is paging space too full? (FCX109 DEVICE CPOWNED)
  - Does application performance seem OK? (you tell me)

Storage Management and VDISKs

- **Referenced VDISK pages are avoided in Pass 1**
- **This customer realized he had a lot of VDISK for Linux swap space**
- **If those VDISK pages are used often, they will tend to stick and be ejectable by only emergency scan**
- **Hmm, customer tried an experiment…**

Customer Removed His VDISKs



Source data: Storage

Case Study Summary

- **Try to look at system as a whole**
- **Whether applications seem debilitated is the best indicator of whether the system is suffering**

Summary

- **It's important to know the limits of a system from all perspectives and to track where you are in comparison to those limits.**
- **It's important to know how the technology changes over time.**
- **Knowing the above, makes it easier to understand and manage your systems.**