VM/ESA Performance Case Studies

Last Updated November 12, 1999

Bill Bitner
VM Performance
607-752-6022
bitner@vnet.ibm.com

This session gives you a glimpse into my life at work. While I do work with the current release under development, write a little code, consult on design alternatives, and interface with monitor vendors, the majority of my job deals with helping customers who are unhappy with the performance they are seeing or who are just looking for an explanation for some anomaly.
This session will look at 5 such cases, which span problems in a variety of areas. Different methods will be used to evaluate and understand the system performance. These are actual situations I worked on throughout 1998 and early 1999.

Legal Stuff

Disclaimer

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

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

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

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

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

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

Trademarks

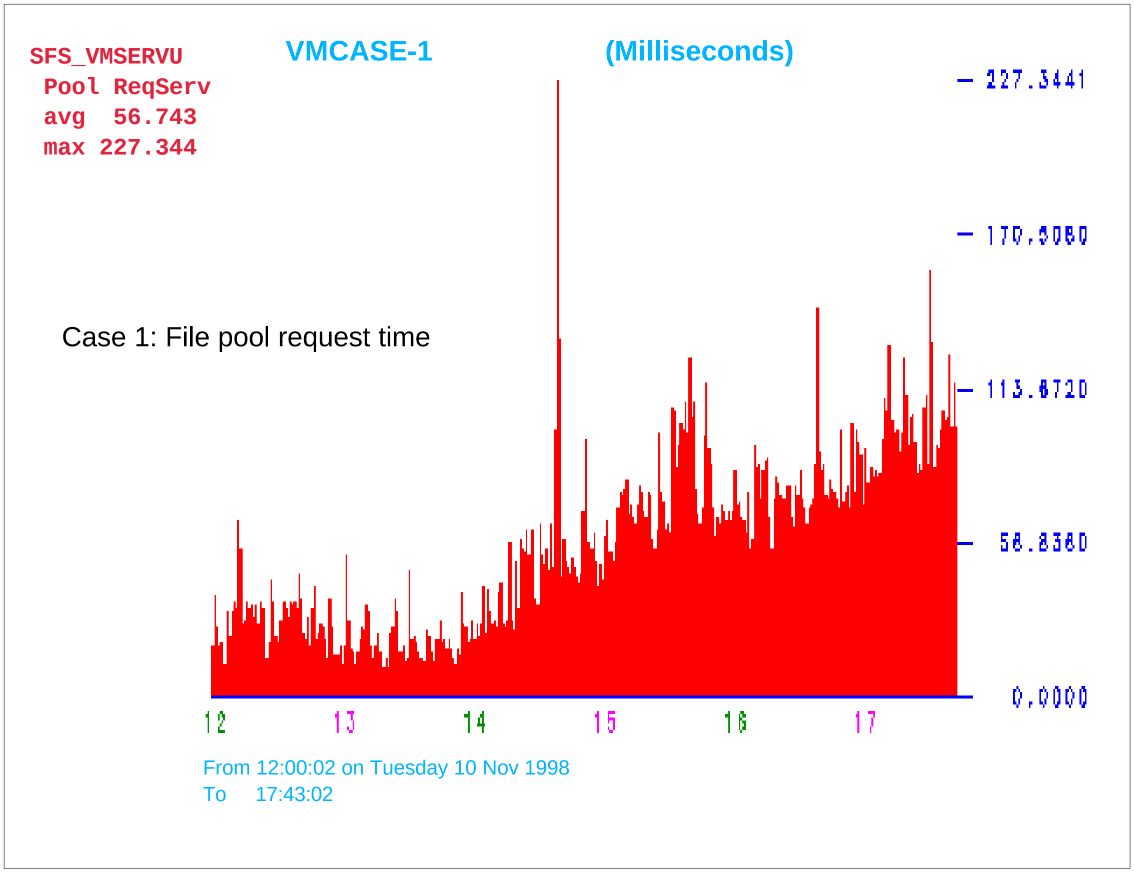
The following are trademarks of the IBM Corporation:
IBM, VM/ESA

I will also show various example of reports and data in this presentation. Many of the reports have been slightly edited to allow them to fit on the page and to highlight the important information.

Case 1: Several wrongs make a mess.

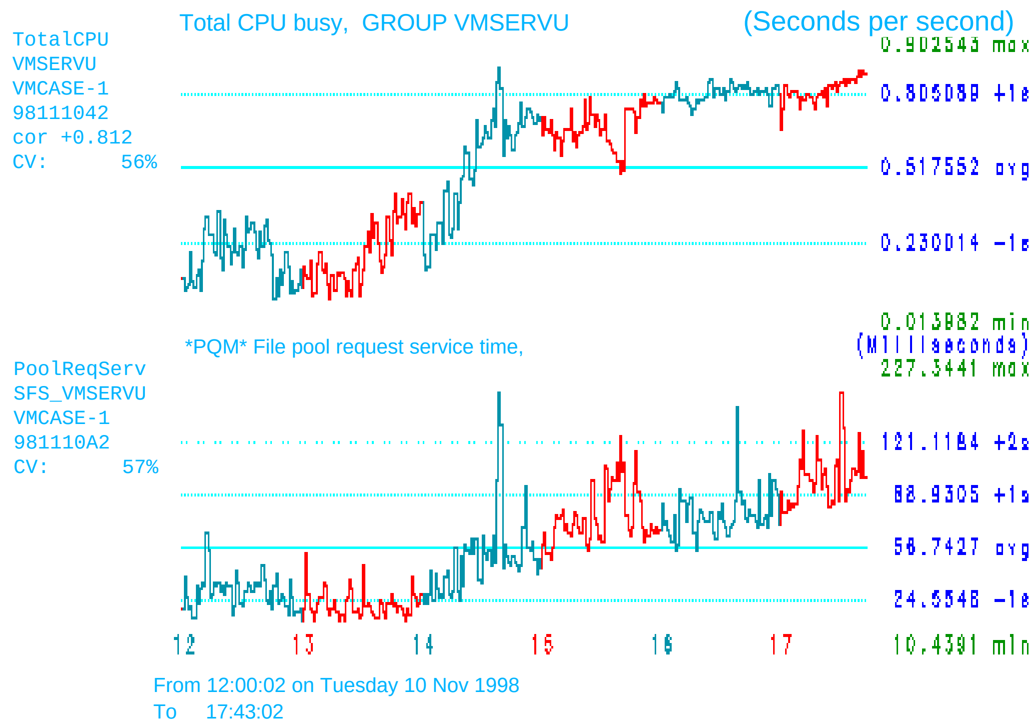
- Customer doing "long" migration between VM/ESA 1.2.1 and 2.2.0.
- VM/ESA 2.2.0 running second level
- Also moving from minidisk to SFS
- Also changing applications for year 2000

"Wrongs" here are not meant to be a slam against anyone. It is to illustrate that when several things go wrong at once, it is more difficult to sort things out.
A "long" migration is my phrase to describe a migration that spans several releases. In this case, the changes from three releases would be involved. To complicate things further: SFS was replacing standard minidisks for all of the user data, the 2.2.0 release was running as a V=R guest, and applications were in the middle of changes for year 2000 work. Fortunately, the hardware stayed the same (except for some data being moved between 3390-3s and RVA).



This is output from VMPAF (VM Performance Analysis Feature). The x-axis is time spanning about 6 hours. The y-axis is the service time for SFS file pool requests. The customer had complained about poor application response time when using SFS on the VM/ESA 2.2.0 system. The units shown are milliseconds. The average and maximum are listed in upper left-hand corner as 58.743 and 227.344. The average is much higher than I would expect to see for a typical SFS workload.

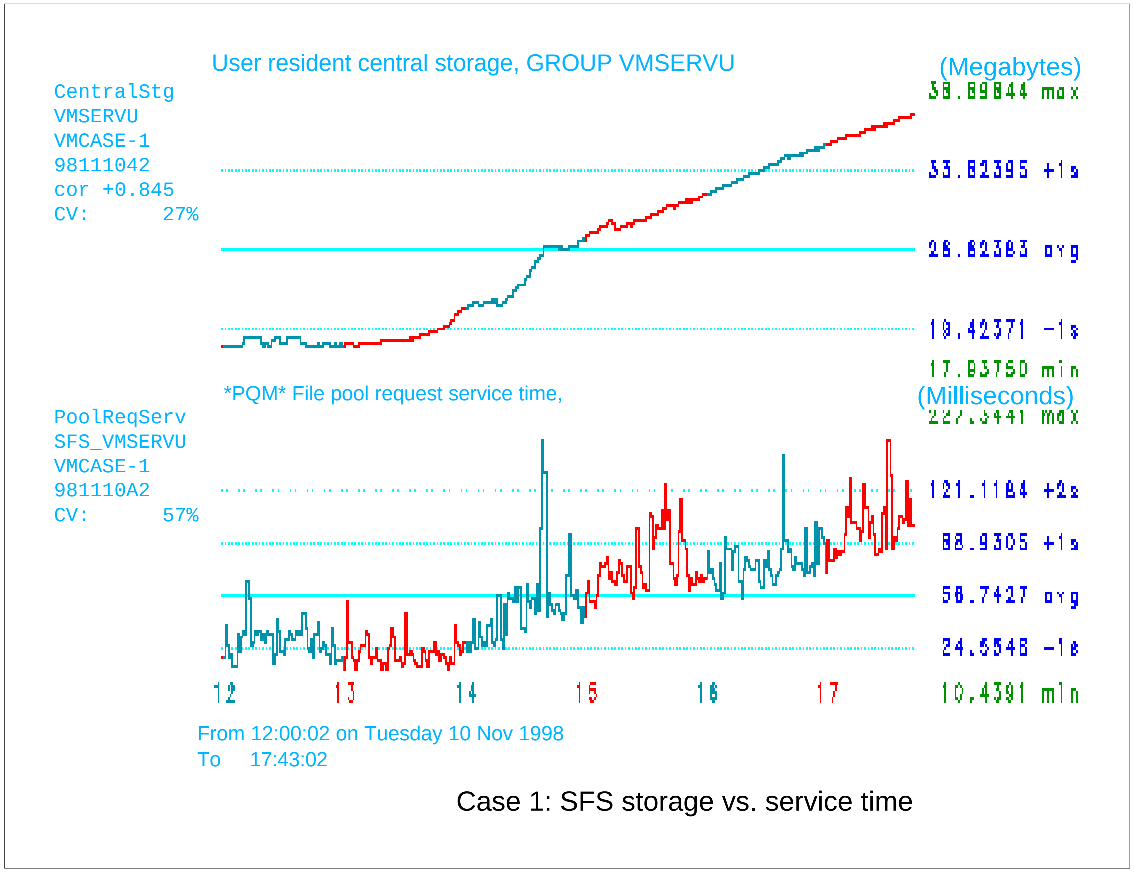
By looking at SFS request response time over this time span, we see some times are better than others.



Case 1: SFS processor usage vs. service time

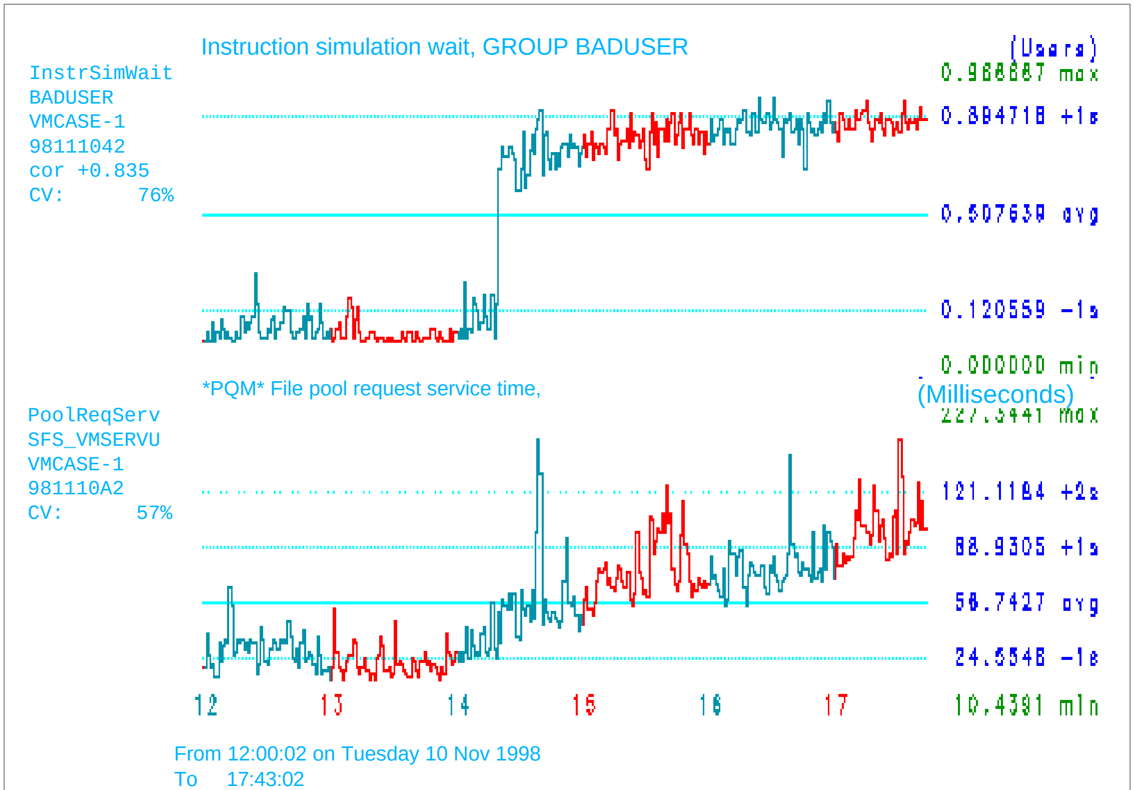
This is another VMPAF chart. In the case you see two graphs that cover the same time span. The file pool request time is at the bottom. This is noted as our PQM (performance quality measure). After choosing a PQM, we can ask VMPAF to do a correlation analysis on other variables and then look at those with the highest correlation. The top graph shows the total CPU time used by the SFS file pool server. It had a correlation value of 0.812 (the closer to 1, the better the two graphs correlate.

The two different colors for the hours are just to make the chart more readable.



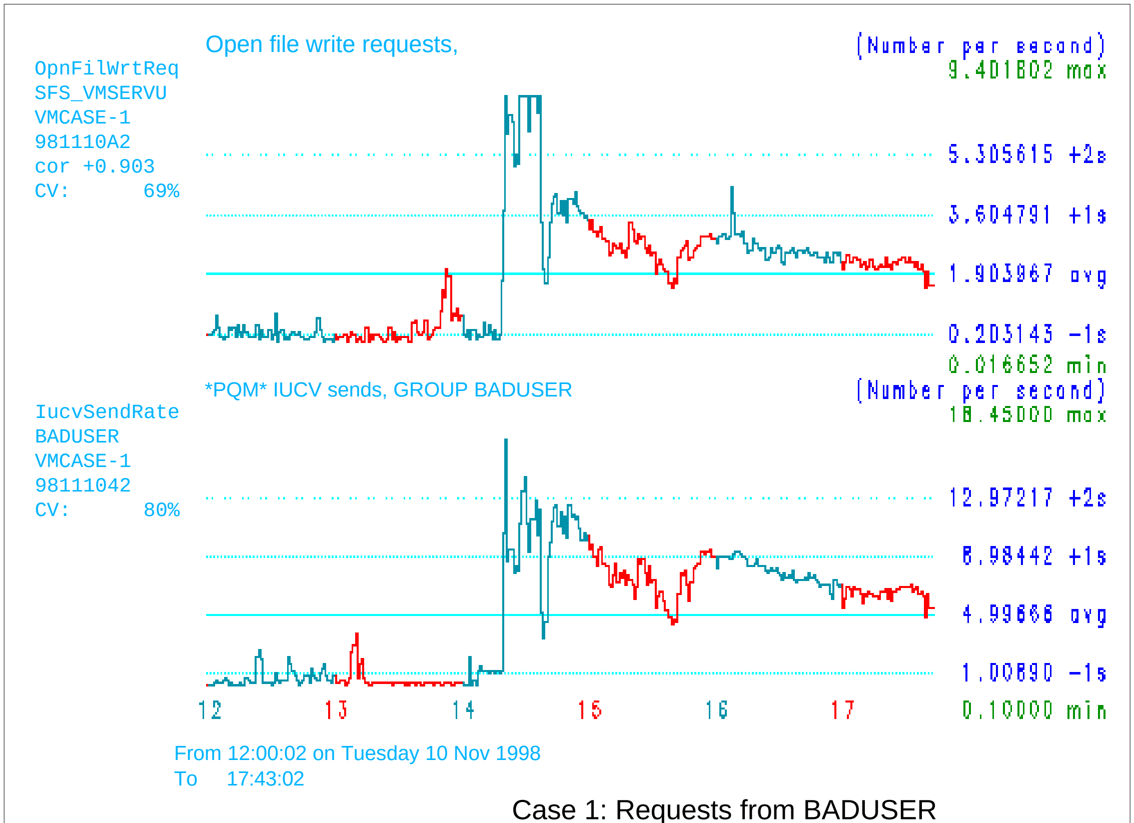
Case 1: SFS storage vs. service time

Another variable that tracked well was the resident central storage for file pool server. Note how it was relatively constant until about 14:00, at which point it continues to grow. This indicates that storage consumption for some reason was growing.



Case 1: BADUSER tracks to SFS service time

This next variable is interesting because it is from a user other than the file pool server. The top graph here is of the instruction simulation wait time for BADUSER. Looking at a dump of the SFS server would show some interesting things about BADUSER, which we can confirm in the following graphs.



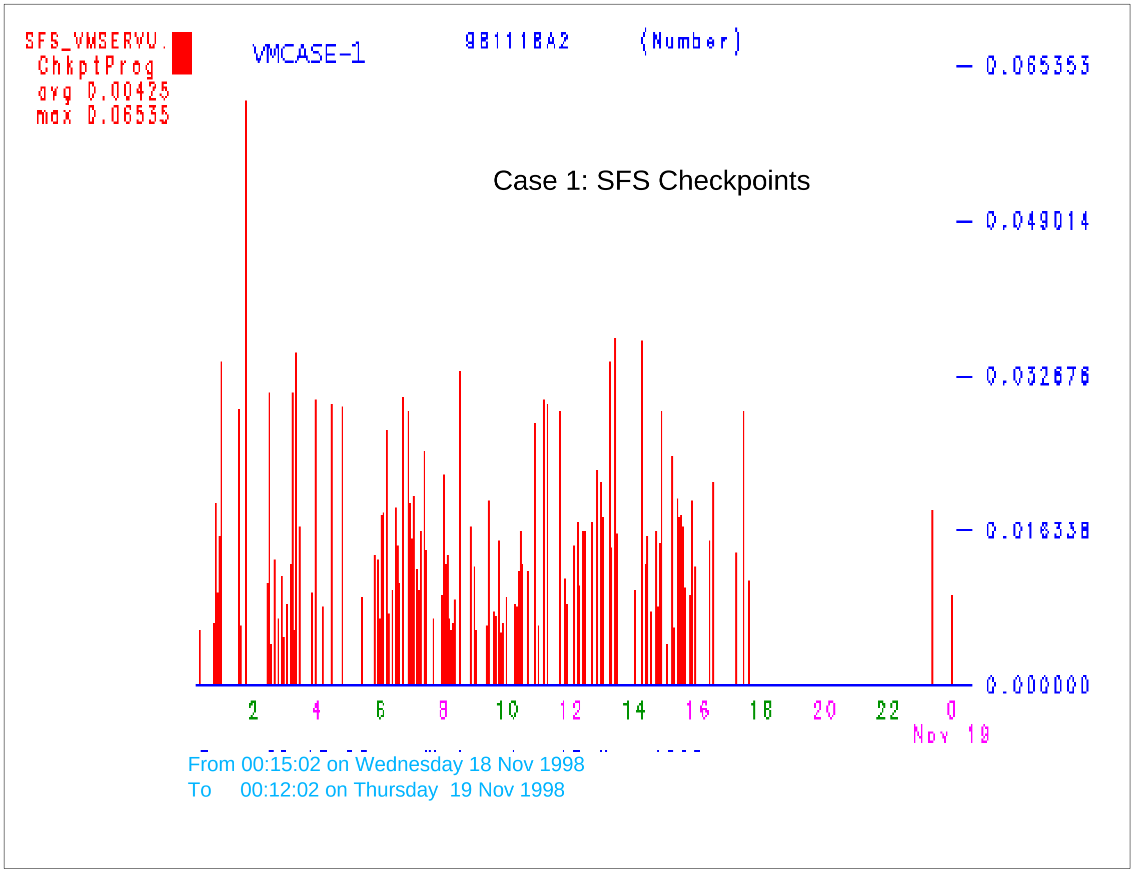
Case 1: Requests from BADUSER

We now change the PQM to IUCV send rate for BADUSER. SFS requests travel over APPC/VM which is seen in VM monitor data as the IUCV rate. Notice here how the Open File Write requests track well to the IUCV send rate. If we also looked at the Close request rate and File Write request rate, we would see those tracking very closely and being a large portion of the SFS activity.

Case 1: SFS APAR VM62086

- An APL application was repeatedly:
  - opening 5 or 6 files in another file space
  - writing a few records
  - closing without commit
- SFS server grew in costs
  - storage to support all the iterations (5000+) of each file
  - processor time to manage some of this
- During uninterruptible periods of the processing, other requests were ignored.

The dump showed BADUSER running an application that did file manipulation thousands of times without committing the work. Storage was consumed for control blocks created with the various changes that had not been committed. Processor time also grew for the longer scan times. Other users would appear "locked" out of the server during uninterruptible periods of the processing. The SFS APAR VM62086 was created to allow the file pool server to open windows during these times to let other agents run. Also, the application was corrected to write more records in each open/close cycle.



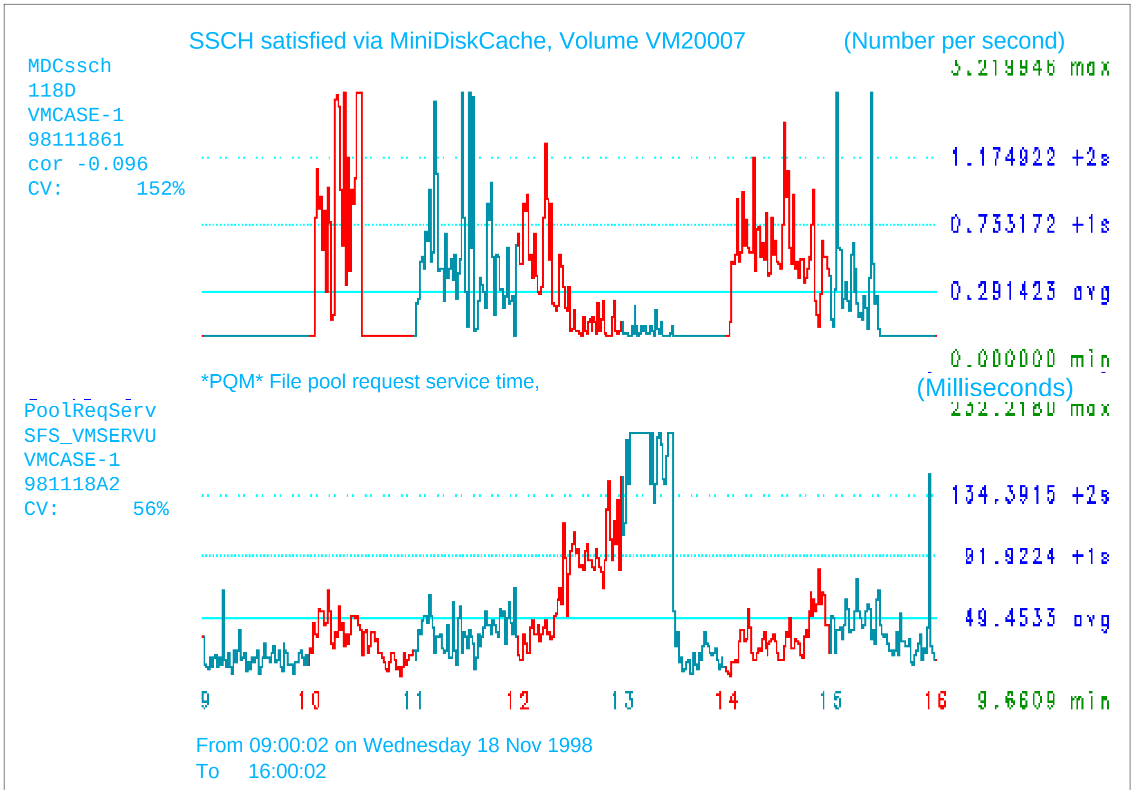
On the same system, but for a 24 hour span, we see the number of SFS checkpoints for the interval. Since SFS checkpoint rate is a factor of the amount of work going on in SFS, we see more checkpoints during the prime shifts of the day.

Checkpoint processing is required to reclaim log space. There have been improvements to it over the years. During checkpoint processing there is a point where no other processing can occur. This serializes the server very briefly, but can have some significant impact in certain special cases.

Case 1: Delete User

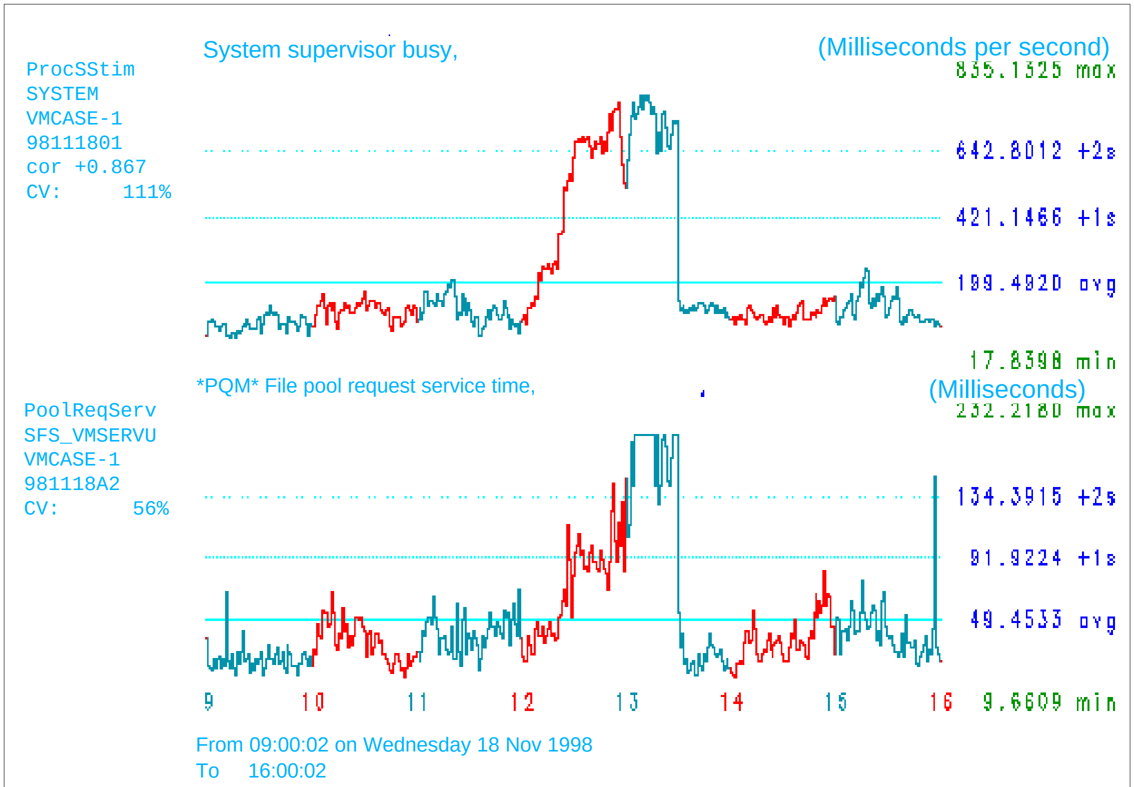
- Delete User is an expensive command
  - No index for reverse look up of authority
  - Must read all the catalog authority data to find what objects the user being deleted might have been granted authority
- Locks out other users if a checkpoint occurs during Delete User
  - Therefore schedule DELETE USER off-shift
- Check QUERY FILEPOOL REPORT for revoke user (or appropriate monitor data)
- Will also see large number of catalog reads

Delete user is an expensive command, particularly for authorization revoking processing. The file pool server must examine every catalog authority row to see if any objects described by that data have been granted authority to the user being deleted.
If checkpoint processing occurs at the same time as a Delete user is in progress, it will lock out other users. Checkpoint processing will try to serialize the server by not allowing any new work to start until current work finishes and checkpoint serialization processing runs. This is why we recommend deleting users off-shift.
The revoke user counter and catalog reads can indicate a delete user has taken place. The customer was doing some of these.



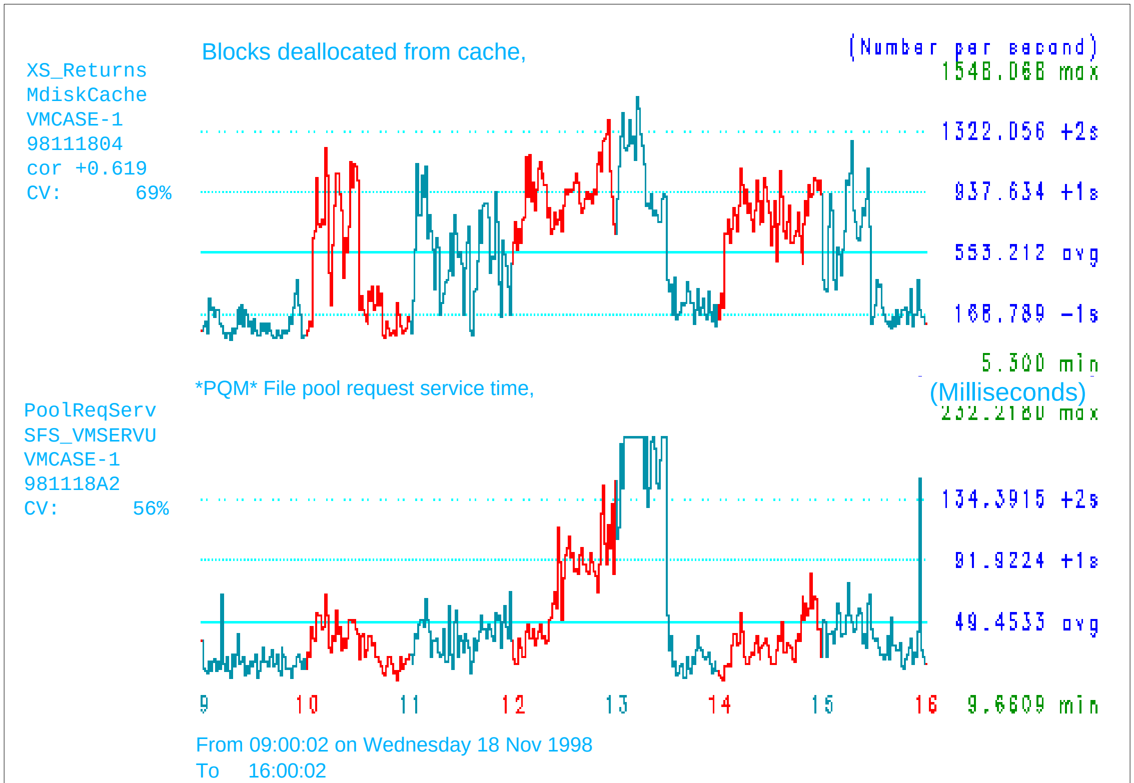
Case 1: MDC for SFS User Data

Another anomaly was that performance was worse when MDC (minidisk cache) was enabled for SFS Storage Group 2 (user data). The chart here shows the real I/Os avoided due to MDC for SG 2 on the top and the file pool request service time on the bottom. MDC was enabled from about 10:00 to 10:30, 11:00 to 13:30, and 14:00 to 15:45. You see benefit from MDC, but also large jumps in the file pool request service time whenever MDC is enable.
This had me very puzzled.



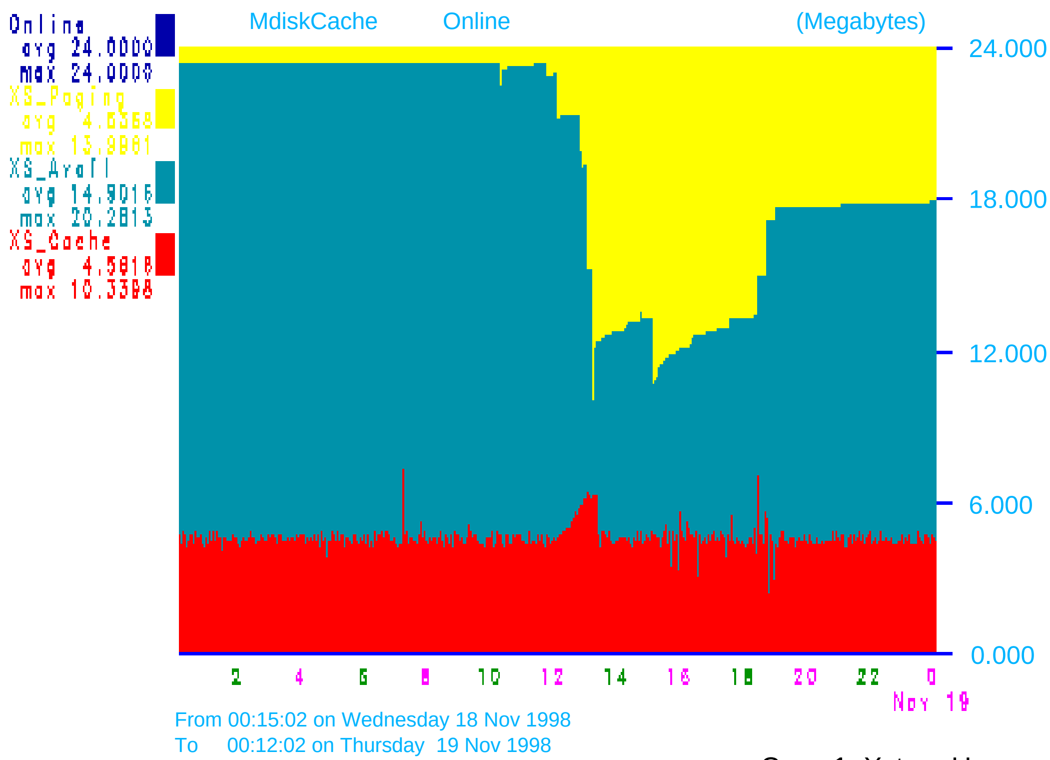
Case 1: System time vs. SFS service time

Using the file pool request service time as our PQM, one of the higher correlating values was System processor time. This is processor time that is charged to the system because it cannot be associated with anyone particular user. Typically scheduler overhead, monitor, etc. fall into this category.



Case 1: MDC Xstore turnover

Another variable with a high correlation was rate per second of blocks in expanded storage being deallocated from MDC. The top graph shows over 500 blocks per second. This seemed very odd since the machine was not that storage constrained. This could be associated with the system processor time and was worth further investigation.



Case 1: Xstore Usage

This is a VMPAF layer chart which shows how the 24 MB of expanded storage for the V=R guest was being used between MDC, available (unallocated), and paging (on top). You can see that for all of the day, we have a large amount of expanded storage not being used and MDC is being held basically to around 6 MB. That would constrain the system, but I did not understand why only 6 MB was being used.

VMPRF PRF103: MINIDISK_CACHE_USAGE_BY_TIME

From Time	Ideal	Actual	Min Set	Max Set	Rate Pages Deleted	Rate Steal Invoked	Bias
	<----------------	Xstore	--------------------	>			
11:50	1228	1131	0	2048	402	2.563	0.20
12:00	1228	1171	0	2048	653	4.177	0.20
12:10	1228	1195	0	2048	949	12.490	0.20
12:20	1228	1220	0	2048	763	28.722	0.20
12:30	1228	1264	0	2048	948	76.410	0.20
12:40	1228	1361	0	2048	896	65.013	0.20
12:50	1254	1442	0	2048	1223	67.520	0.20
13:00	1345	1544	0	2048	1190	63.847	0.20
13:10	1376	1574	0	2048	1354	67.140	0.20
13:20	1368	1573	0	2048	818	61.815	0.20
13:30	1228	1123	0	2048	298	1.873	0.20

Who changed this!

VMPRF (VM Performance Reporting Facility) added a report when MDC was enhanced. An abbreviated version is shown here for the Xstore portion of MDC. A tuning knob, called MDC Bias, that is seldom used is reported here. I had overlooked it earlier. It had been set to 0.20 which means CP would restrict MDC to only 20% of what the arbiter thought it needed in expanded storage. This could cause a thrashing environment as is shown in the large values for "Rate Pages Deleted" and "Rate Steal Invoked". Also note the maximum had been set to 8 MB! These needed some changing.

Case 1: Solution

- SFS service
  - VM61547: delete of large files
  - VM62008: son of VM61547
  - VM62086: large open/write/close loops
- Fix some applications
- Avoid deleting SFS users in prime shift
- Remove MDC bias and max settings.

Change management is a performance analyst's friend.

In summary, there were a couple key SFS APARs that the system needed. One came from this customer situation. The customer also corrected some misbehaving applications and try to educate the application folks about the differences between minidisks and SFS. Simply avoiding the deleting of SFS users during prime shift helped. The bias and maximum settings were removed from MDC. In the long run, storage configuration would be changed significantly as VM/ESA 2.2.0 became the first level system. Situations like this show the value of have change management systems that are all inclusive. Some time was lost in measuring things that were changing. It is more difficult to hit a moving target.

Case 2: Network Problem

- Customer had put in a 2216 to connect VM/ESA host to network with NT machines.
- Plan to exploit with ADSM
- They expected more than they saw from initial testing with FTP
  - 1.5 MB/Sec to/from VM
  - 3.0 MB/Sec between NT machines
- Belief there was something wrong with 2216

Customer purchased 2216 believing it would meet their networking needs for ADSM, but they were disappointed with initial testing with FTP by only getting about half the throughput to VM as they were between two NT machines. It was their belief, and I was willing to agree, that something was wrong with the 2216.

Case 2: Basic TCP/IP Reports

FCON/ESA TCP/IP Links Activity Log for Server TCPIP

```
                     <------------- Received/s ------------->
                           <----------- Packets ----------->
Interval                    Uni-    Non- Dis-        Unknown
End Time Link Name   Bytes  cast Unicast card Error Protocol

11:41:01 NET2216TR0  1594k 195.4    .000  .00  .000     .000


                     <------ Transmitted/s ------->
                           <------ Packets ------->
Interval                    Uni-    Non- Dis-
End Time Link Name   Bytes  cast Unicast card Error

11:41:01 NET2216TR0   6504 101.1    .000  .00  .000
```

We were able to look at some basic TCP/IP information from the monitor records generated by the TCP/IP stack through some tools we had in the lab. That same data is shown here as report by the FCON/ESA performance tool. We were able to see the basic throughput and confirm that packets were not being discarded or in error.

Case 2: A bit of tuning

- Multiple FTPs showed slightly better performance.
- 2216 has a couple of key tuning parameters
  - BLKTIMER- 2216 waits for data from other connections before sending what it has on to the host. Helps minimize interrupts. In single thread benchmark, this is not good. Blktimer is how long to delay before sending any way.
  - ACKLEN - To avoid delays of acknowledgments, send any requests this size or smaller immediately (bad default of 10).

Interestingly, multiple FTPs showed slightly better performance. That led us to believe that it was an attribute of single threaded host testing. The 2216, like many network boxes, will try to avoid flooding the host with interrupts because of the cost of processing. One aspect of this is to hold or delay data for a period of time before presenting to the S/390 host so that other data can arrive and be presented with fewer interrupts. This delay factor is controlled by the 2216 tuning parameter BLKTIMER. The default of 5 milliseconds could be significant in a single thread scenario. However, the 2216 does not wish to delay acknowledgments which depend on low latency times for good performance. The ACKLEN parameter is used for this. Any request the size of ACKLEN or smaller is sent immediately. However, the default of 10 bytes is too small for a VM environment (100 bytes is a better choice).

Case 2: I/O Traces showed TCP/IP Waiting ...for what?

- TRSOURCE I/O Traces that included enough data to get header information
- Showed periodic delays that weren't necessarily network...
- The processor is a 9221-421 (2-way)

Well, after changing those two parameters, the throughput did not increase significantly. So we collected some TRSOURCE I/O traces that contained data which made up the header information. From this information, we were able to piece together the flow and delays of the requests. This analysis showed delays that were not necessarily in the network. This turned us to look at something other than the network. Since the processor was an older machine, a 9221-421, that was a good place to start.

Case 2: Drat, it's not the network!

FCON/ESA User State Display

Userid	%ACT	%RUN	%CPU	%LDG	%PGW	%IOW	%SIM	%TIW	%CFW
>System<	17	19	14	0	0	1	3	5	0
ADAITM	100	0	0	0	0	0	0	0	0
DSMSERV	100	23	**52**	0	0	0	7	0	0
TCPIP	100	33	**13**	0	0	0	**20**	0	0
VSESYSV	100	83	5	0	0	0	0	0	0

FCON/ESA CPU Load Display:

PROC	%CPU	%CP	%EMU	%WT	%SYS
P00	93	24	69	7	3
P01	95	20	75	5	2

It's easy to blame the network, but not always correct.

Using FCON/ESA again, we looked at the User State display to see what users were waiting on. The DSMSERV (ADSM server) and TCP/IP were both waiting for CPU. In addition, there was simulation wait time for TCP/IP, perhaps waiting for replies from the ADSM server. From a system view, there was not a lot of processor resources available. Both processors, as seen in the CPU Load display, were over 90% busy. Additional processor resources would increase the throughput.
While it is easy to blame the network, that is not always the correct thing to do. Fortunately, we had enough data and the tools to look at that data to get to the real answer.

Case 3: Broken RAMAC ?

- Customer had acquired some RAMAC I DASD and called in with concern about performance based on DDR measurements.
- Configuration:
  - RAMAC I DASD run as 3390-3 volumes
  - 3990-3 Control Unit 64MB cache / 4MB NVS
  - DDR tests: COPY ALL from one RAMAC volume to another.

This next case also deals with expectations not being met. The customer had acquired some RAMAC I DASD and after some preliminary testing with DDR were concerned about the performance of this new DASD. Of the various RAMAC configurations, this one falls on the lower end of the scale (RAMAC Subsystem and RAMAC DASD with 3990-6 being higher). Note also, the amount of cache and NVS compared to other current offerings.

Case 3: Why high disconnect?

VMPRF DASD_BY_ACTIVITY PRF012:

Dev	SSCH Rate	Pct Busy	Pend	Disc	Conn	Serv	Resp
Input	7.9	16.5	0.2	3.8	16.9	20.9	20.9
Output	7.9	75.5	0.1	**75.2**	20.1	95.4	96.0

VMPRF DASD_BY_CONFIG_EF PRF096:

	<---Rate-->		<-------------Percent-------------						-->
	Total	Read		<---------Hits---->				Cache	Norm
	I/O	NonSq	Read	Tot	Read	Wrt	DFW	I/O	Stge
Input	7.9	7.9	100	0	0	0	0	100	100
Output	7.9	0.0	0	0	0	0	0	100	0

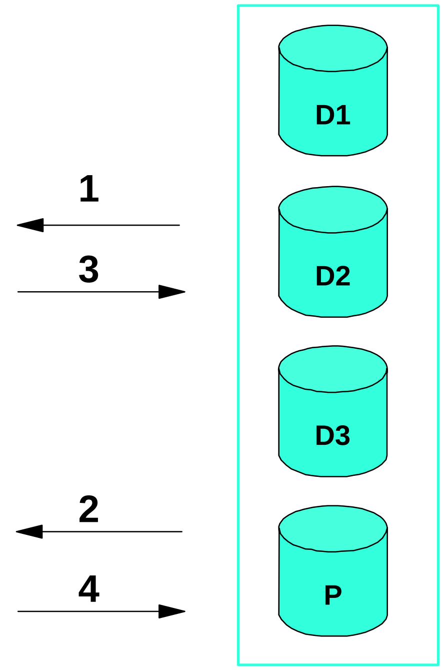
Cache is of no help in this scenario.

I had the customer collect some monitor data and send it in. The two VMPRF reports show the two volumes involved with the DDR. As you can see in the PRF012 DASD_BY_ACTIVITY report, the service time is very high for the volume being written, and most of this is in the 'disconnect' component of service time. The PRF096 Enhanced Functions report shows that cache is of no benefit in this scenario.

Case 3: Why is Write worse than Read?

Normal RAID-5 write penalty
example of updating data on D2:

1. read old data from D2
2. compare old and new data for parity computation
3. read old parity from P
4. write new data to D2
5. write new parity to P



IBM RAMAC has features to mitigate the write penalty, but not enough in this configuration and scenario.

To understand why there are significant differences between read and write performance, we need to understand something about RAID 5. In a normal RAID-5 environment, data is stripped along disks along with parity information for that data. (Actually, a single volume can contain data and parity information, but the parity information is not for the data on that disk). Updating data on the disk, could involve 4 different I/O operations as we determine the old parity and the new parity, write out the new data and new parity.

Now the IBM RAMAC has features to help mitigate the write penalty. However, this configuration is weak on cache and this is write once data which is not cacheable in this configuration.

Case 3: Solution

- DDR backup or copy on RAMAC I is a worse case scenario and will run slow, especially on 3990-3 with small cache sizes.
- Normal workloads would prove to run fine.

Do not judge a device by its DDR times!

When you think about it, DDR restore or copy on RAMAC I are worse case scenarios, especially with the less sophisticated 3990-3 control unit. The customer accepted this explanation and saw that normal workloads would show better performance.
So don't judge a device by its DDR times (unless that's all you do with the device).

Case 4: The Grinch that stole performance.

From VMPRF USER_STATES_BY_TIME PRF007 Report January 5:

<----Percent of True Non-Dormant Time Waiting on--------------->										
							<---SVM and---->			I/O
	Load-			Inst	Test	Cons	Test	Elig-	Dor-	Ac-
CPU	ing	Page	I/O	Sim	Idle	Func	Idle	ible	mant	tive
0.1	0.1	0.1	**18.8**	2.3	10.0	0.4	3.4	0	50.8	8.4
0.1	0	0.1	**16.0**	1.9	9.9	0.4	3.1	0	53.8	9.9

From VMPRF DASD_BY_ACTIVITY PRF012 Report January 5:

	SSCH	Pct	<-----------Time------------>					<--Queue-->	
Dev.	Rate	Busy	Pend	Disc	Conn	Serv	Resp	Mean	Max
1742	26.7	65.4	1.3	**18.4**	4.7	24.5	**69.0**	1.2	8.5

Went to check VMPRF DASD_BY_ACTIVITY_EF PRF095 for control unit cache stats, but it didn't exist!

It is a good thing I keep historical data, lets go back and see what's going on...

It was the first week in January with people coming back from the holidays ready to code up wonderful things. However, the system response time was horrible. Looking at the VMPRF User States report, I could see that we were waiting longer than usual for I/O. A look at the DASD_BY_ACTIVITY report showed one of the devices with poor service time and terrible response time. The high disconnect time made me think there was something wrong with the cache. However, when I looked at the Extended Functions report for DASD, the device was not there! It was time to look at some historical data I had kept for just a time as this.

Case 4: When did we last see it?

From VMPRF DASD_BY_ACTIVITY PRF012 Report from December 8:

	SSCH	Pct	<-----------Time------------>					<--Queue-->	
Dev.	**Rate**	**Busy**	**Pend**	**Disc**	**Conn**	**Serv**	**Resp**	**Mean**	**Max**
1742	41.0	10.5	0.3	0.2	2.0	2.6	2.9	0.0	0.3
Jan5:	26.7	65.4	1.3	**18.4**	4.7	24.5	**69.0**	1.2	8.5

VMPRF DASD_BY_ACTIVITY_EF PRF095 Report for 1742 on Dec 8:

<---------Rate-------->				<-----Percent----------->				
Total	Read	Read	Write		<---------Hits----->			
I/O	NonSq	Seq	FW	Read	Tot	Read	Wrt	DFW
53.0	52.3	0	0.6	99	99	99	96	96

Going back to VMPRF reports from December 8th, we saw that there was a big difference, and that cache was there and effective at one time. It was odd that cache was no longer being reported.

Case 4: Down for the 3 count

```
q dasd details 1742
1742 CUTYPE = 3990-EC, DEVTYPE = 3390-06, VOLSER= USE001
      CACHE DETAILS:  CACHE NVS CFW DFW PINNED CONCOPY
           -SUBSYSTEM   F    Y   Y   -    Y       N
           -DEVICE      Y    -   -   Y    N       N
      DEVICE DETAILS: CCA = 02, DDC = 02
      DUPLEX DETAILS: SIMPLEX
```

Pinned data! Yikes! I had never seen that before!

I did a simple QUERY DASD DETAILS and saw something I had never actually seen before: pinned data. On 3990 control units that support DASD Fast Write through NVS, pinned data occurs when the control unit cannot write the data out to the actual DASD for some reason. In that case, the data must be held in NVS until the problem can be resolved. In this case, the failure also resulted in cache being disabled for the control unit. It then looked like a non-caching control unit and therefore was not listed in some of the cache reports.

Case 4: FCON/ESA Device Report

```
FCX110     CPU 2003   GDLVM7    Interval INITIAL. - 13:08:47     Remote Data

Detailed Analysis for Device 1742 ( SYSTEM )
Device type :  3390-2      Function pend.:     .8ms      Device busy   :   27%
VOLSER      :  USE001      Disconnected  :   20.3ms      I/O contention:    0%
Nr. of LINKs:     404      Connected     :    5.4ms      Reserved      :    0%
Last SEEK   :    1726      Service time  :   26.5ms      SENSE SSCH    :  ...
SSCH rate/s :    10.5      Response time :   26.5ms      Recovery SSCH :  ...
Avoided/s   :    ....      CU queue time :     .0ms      Throttle del/s:  ...
Status: SHARABLE

Path(s) to device 1742:    0A    2A    4A
Channel path status   :    ON    ON    ON

Device            Overall CU-Cache Performance           Split
DIR ADDR VOLSER   IO/S %READ %RDHIT %WRHIT ICL/S BYP/S   IO/S %READ %RDHIT
08  1742 USE001     .0     0      0      0    .0    .0   'NORMAL' I/O only
```

The FCON/ESA device report provides the same information as some of the VMPRF reports. This common information is shown here.

Case 4: FCON/ESA Device Report

```
  MDISK Extent    Userid   Addr IO/s VSEEK Status    LINK MDIO/s
+-----------------------------------------------------------------+
|   101 -  200    EDLSFS   0310   .0     0 WR           1     .0 |
|   201 -  500    EDLSFS   0300   .0     0 WR           1     .0 |
|   501 -  600    EDLSFS   0420   .0     0 WR           1     .0 |
|   601 - 1200    EDLSFS   0486   .0     0 WR           1     .0 |
|  1206 - 1210    RAID     0199   .0       owner                 |
|                 BRIANKT  0199   .0     0 RR           5     .0 |
|  1226 - 1525    DATABASE 0465   .0       owner                 |
|                 K007641  03A0   .0     0 RR           3     .0 |
|  1526 - 1625    DATABASE 0269   .0       owner                 |
|                 BASILEMM 0124   .0     0 RR          25     .0 |
|  1626 - 1725    DATABASE 0475   .0       owner                 |
|                 SUSANF7  0475   .0     0 RR           1     .0 |
|  1726 - 2225    DATABASE 0233   .0       owner                 |
|                 ACTSMACH 0233   .0     0 RR         366   10.5 |
+-----------------------------------------------------------------+
```

This part of the FCON/ESA Device Report is different than most other performance products. It lists various active minidisks on the subject volume and provides an approximate I/O rate for each. As we see here, the DATABASE 233 disk is located on the volume that was seeing such poor performance. This is a key disk in our development library processing tools. Being a software development lab, we are somewhat dependent on this disk.

Case 4: Solution

- Use **Q PINNED** CP command to check for what data is pinned.
- Discussion with Storage Management team.
- Moved data off string until corrected.

Pinned data is very rare, but when it happens it is serious.

The Storage Management team moved the key minidisks off of that troubled control unit until the problem could be resolved. The CP QUERY PINNED command can be used to determine exactly which tracks of information are pinned. This helps in the problem management process.
As I said earlier, pinned data is very rare. However, it is also very serious. Performance suffers significantly when cache is lost.

Case 5: Best of times, worst of times

```
VMPRF RESPONSE_ALL_BY_TIME PRF006 Report:
<---Response Time---->  <--------Throughput--------->
 Triv  Non-Triv  QDisp   Triv  Non-Triv  QDisp  Total
Good time (7:20 to 7:24):
 0.026    0.224  0.456   116.71  112.21  16.37 245.28
Bad time (9:10 to 9:14):
 0.038    0.706  4.246   174.09  158.64   5.43 338.16
```

Users getting in early get good performance, while those coming in later see the worst performance.

This last case is from a customer who had a good idea what the problem was, but simply wanted confirmation that they were looking at the right things. Early in the morning, this system seemed to provide good response time. However, around 9:00 or so the response time degraded over 45%.

Case 5: Good Performance

VMPRF PROCESSORS_COMPLEX_BY_TIME PRF015 7:20 to 7:24

	<---Percent Busy------>				<--Rate-->		<--------PLDV-------->			
								<-----VMDBKs----->		
								<Ct>	<----Rate--->	
CPU	Total	User	Syst	Emul	Inst Siml	SSCH and RSCH	Pct Em-pty	Mean when Non0	Stolen	Moved to Master
0	80.9	61.4	19.6	10.6	836	444	13	4	324.6	4321.3
1	71.2	64.9	6.3	52.2	1704	129	81	1	611.9	0
2	69.9	64.0	5.9	51.6	1675	126	82	1	594.6	0
3	68.7	62.7	5.9	50.8	1614	123	87	1	572.0	0
4	66.6	60.9	5.7	49.2	1548	121	87	2	557.6	0
5	64.7	59.2	5.5	47.7	1500	117	88	1	536.8	0
6	62.8	57.3	5.5	46.0	1466	119	88	1	532.2	0
7	61.6	56.1	5.5	45.2	1401	114	86	1	512.1	0
8	60.4	55.2	5.2	44.4	1387	113	87	1	504.3	0
9	59.5	54.2	5.3	43.4	1368	110	90	1	494.9	0

They happen to be running on a 10-way processor. The VMPRF Processors Complex by Time report is shown here. You can determine the master processor by looking for the non-zero value in the "Moved to Master" column. The report from the "good times" shows the master processor slightly busier than the others, but still processing user emulation work.

Case 5: Bad Performance

```
VMPRF PROCESSORS_COMPLEX_BY_TIME PRF015 9:10 to 9:14
  <---Percent Busy------> <--Rate--> <--------PLDV-------->
                                         <-----VMDBKs----->
                                         <Ct> <----Rate--->
C                               SSCH Pct Mean         Moved
P                          Inst  and Em- when            to
U Total  User  Syst  Emul  Siml RSCH pty Non0 Stolen Master
0  93.9  69.0  24.8   0.4    40  429   0   10    3.1 5688.6
1  82.6  72.7  10.0  57.0  2044  189  61    3  638.6      0
2  81.8  72.3   9.5  56.8  2003  193  55    2  632.2      0
3  81.1  71.4   9.7  56.2  1969  188  65    3  612.6      0
4  79.9  70.3   9.7  55.4  1896  188  63    2  600.7      0
5  79.0  69.8   9.3  54.8  1898  185  64    3  586.6      0
6  78.0  68.7   9.3  54.3  1804  182  64    2  566.8      0
7  76.7  67.7   9.0  53.2  1811  184  66    3  563.1      0
8  76.4  67.5   8.9  53.1  1794  184  62    3  544.4      0
9  75.7  66.8   8.8  52.7  1781  184  63    3  537.4      0
```

Looking at the same report for the "bad times", we see all the processors more utilized and the skew between master and alternates about the same. However, the master is no longer doing any emulation work. It is spending all its time doing CP work which most likely is master-only work. The PLDV queues show that the master is never without work to do and therefore seldom has time to 'steal' work from other processors to help out as seen in the 'Stolen' column.

Case 5: Good vs. Bad

VMPRF PROCESSORS_COMPLEX_BY_TIME PRF015 9:10 to 9:14

	<---Percent Busy------>				<--Rate-->		<--------PLDV-------->			
								<-----VMDBKs----->		
								<Ct>	<----Rate--->	
CPU	Total	User	Syst	Emul	Inst Siml	SSCH and RSCH	Pct Em-pty	Mean when Non0	Stolen	Moved to Master
Good Time:										
0	80.9	61.4	19.6	10.6	836	444	13	4	324.6	4321.3
1	71.2	64.9	6.3	52.2	1704	129	81	1	611.9	0
Bad Time:										
0	93.9	69.0	24.8	**0.4**	40	429	**0**	**10**	3.1	5688.6
1	82.6	72.7	10.0	57.0	2044	189	61	3	638.6	0

This is easier to see if we look at the good times and bad times together. Note that a high utilization on the master does not necessarily mean a bottleneck on the master processor. However, in this case we see the VMDBKs queued on the master PLDVs and the lack of user emulation work. These are significant clues.

Case 5: User States

From VMPRF USER_STATES_BY_TIME PRF007 Report:

<-----Percent of True Non-Dormant Time Waiting on------>										
							<---SVM and--->			I/O
	Load-			Inst	Test	Cons	Test	Elig	Dor-	Ac-
CPU	ing	Page	I/O	Sim	Idle	Func	Idle	ible	mant	tive
Good time:										
4.9	0.7	1.0	0.7	5.2	36.0	4.2	6.7	0	9.2	29.1
Bad time:										
8.0	1.3	1.4	0.7	**15.7**	15.6	**17.6**	3.5	0	8.8	25.8

Instruction simulation and console function mode wait are two common states to be in if bottlenecked on master.

Futher clues are given in the VMPRF User States report. Both the Instruction Simulation and Console Function mode wait states increase drastically in the bad times. A user waiting on the master processor can find itself in either of these states. (note that these states can be high for other reasons also).

Case 5: Solution

- Users are wise to come in a bit earlier
- Master processor constraint in this heavy OV/VM environment
- Customer was already investigating moving to a new CMOS machine with fewer and faster engines
- Tuning and configuration:
  - On VM/ESA 2.3.0, so have VMCF improvement from 2.2.0.
  - 4 millisecond minor dispatch slice
  - Using sensible Monitor settings

The customer was correct in believing they were bottlenecking on the master processor during these heavy load times. It would be smart to come in a bit earlier if you worked on that system. The customer had seen this trend increasing, and had begun research for faster engines. They had also done some of the key tuning and configuration changes to mitigate master processor contention. Prior to VM/ESA 2.2.0, VMCF was serialized on the master processor. With VM/ESA 2.3.0, this is no longer a concern. They had slightly lowered the minor dispatch slice to prevent users from running and holding the master for long stretches at a time. Since monitor runs on the master processor, using sensible settings are important.

