BRUDEN · OSSG
On-Shore Systems Group

OpenVMS Utilities Update
October 2006

Guy Peleg
Senior Member of the Technical Staff
Director of EMEA Operations
guy.peleg@bruden.com

Veux-tu m’ententre roter?

Agenda

- ***V8.3 new features***
- Licensing changes
- Prior releases
- Latest happenings with OpenVMS BACKUP

Agenda

Walt Disney Pictures
Presents
A
PIXAR
ANIMATION STUDIOS FILM
FINDING NEMO
CATCH IT NOW
ON DVD AND VIDEO
findingnemo.com
© DISNEY/PIXAR

Customizing CTRL-T output

- The output of the CTRL-T message may be customized
- The contents of the symbol DCL$CTRLT will be appended to the traditional CTRL-T output
  - Useable from applications / DCL
  - May be used for debugging applications
  - Display the name of current procedure being executed by DCL
  - ….and much more…
- Like every symbol, DCL$CTRLT may have different values in different procedure levels

```
 IPL31> ty ctrlt_loop.com
$ inner=0
$ outer=0
$ loop:
$ loop1:
$ if inner .gt. 20000 then goto end_loop1
$ inner=inner+1
$ dcl$ctrlt=F$FAO("Inner loop count is !SL !/ Outer loop count is !SL",inner,outer)
$ goto loop1
$ end_loop1:
$ inner=0
$ outer=outer+1
$ goto loop
```

```
IPL31> @ctrlt_loop
IPL31::GUY 10:46:37   (DCL)   CPU=00:03:42.68 PF=13453 IO=6743 MEM=187
Inner loop count is 12306
Outer loop count is 0
IPL31::GUY 10:46:43   (DCL)   CPU=00:03:49.19 PF=13455 IO=6744 MEM=187
Inner loop count is 19200
Outer loop count is 2
```

Simple DCL procedure demonstrating Customizing CTRL-T output

```
 IPL31> ty ctrlt_looper.c
          #include <descrip>
          void main()
          {
          int counter=0;
          $DESCRIPTOR(sym_name,"dcl$ctrlt");
          static struct dsc$descriptor_s value_desc;
          char buffer[256]={0};

                   value_desc.dsc$b_dtype = DSC$K_DTYPE_T;
                   value_desc.dsc$b_class = DSC$K_CLASS_S;
                   while (1){
                              counter++;
                              sprintf(buffer,"Counter is %d",counter);
                              value_desc.dsc$a_pointer = buffer;
                              value_desc.dsc$w_length =  strlen(buffer);
                              lib$set_symbol(&sym_name,&value_desc);
                 }
         }
IPL31> r ctrlt_looper
IPL31::GUY 10:47:27 CTRLT_LOO CPU=00:03:53.26 PF=13631 IO=6784 MEM=335
Counter is 216766
IPL31::GUY 10:47:28 CTRLT_LOO CPU=00:03:54.45 PF=13631 IO=6785 MEM=335
Counter is 338429
```

Simple C program demonstrating Customizing CTRL-T output

Customizing CTRL-T output

- Displaying the name of the current procedure

```
$!
$ if f$trnlnm("DCL$CTRLT_DEBUG") .NES. ""
$ then
$       dcl$ctrlt =  f$parse(f$environment("procedure"),,,"NAME") -
                   + f$parse(f$environment("procedure"),,,"TYPE")
$ endif
$
```

- With V8.3, when SYS$OUTPUT is redirected, CTRL-T output will still be displayed on the terminal.

Remote CTRL-T

- Introducing the concept of remote CTRL-T
  - CTRL-T can now display standard CTRL-T information about remote processes.
  - Remote may be on a different system in the cluster
  - The symbol DCL$CTRLT_PID should contain the PID of the remote process

Remote CTRL-T

```
Running on node BLUSKY....hitting CTRL-T
$
BLUSKY::SYSTEM 17:40:55   (DCL)   CPU=00:00:00.16 PF=212 IO=98 MEM=146
$
$! Now define the new symbol
$!
$ dcl$ctrlt_pid="23800436"
$
$! Hit CTRL-T again
$!
IPL31::GUY 17:41:12 LOOPER    CPU=01:28:05.17 PF=2700 IO=594 MEM=322
$
IPL31::GUY 17:41:14 LOOPER    CPU=01:28:07.02 PF=2700 IO=594 MEM=322
$
```

New Permanent DCL symbols

- On image rundown DCL populates $SEVERITY and $STATUS
- Added $FACILITY and $IDENT

```
$ exit  %x10911a02

$ show symbol $status
  $STATUS == "%X10911A02“

$ show symbol $facility
  $FACILITY == "%X00000091“

$ show symbol $ident
  $IDENT == "%X00000340“

$ show symbol $severity
  $SEVERITY == "2"
```

SHOW DEVICE/FULL

- Enhanced information for LAN devices
- LAN-specific $GETDVI item codes added

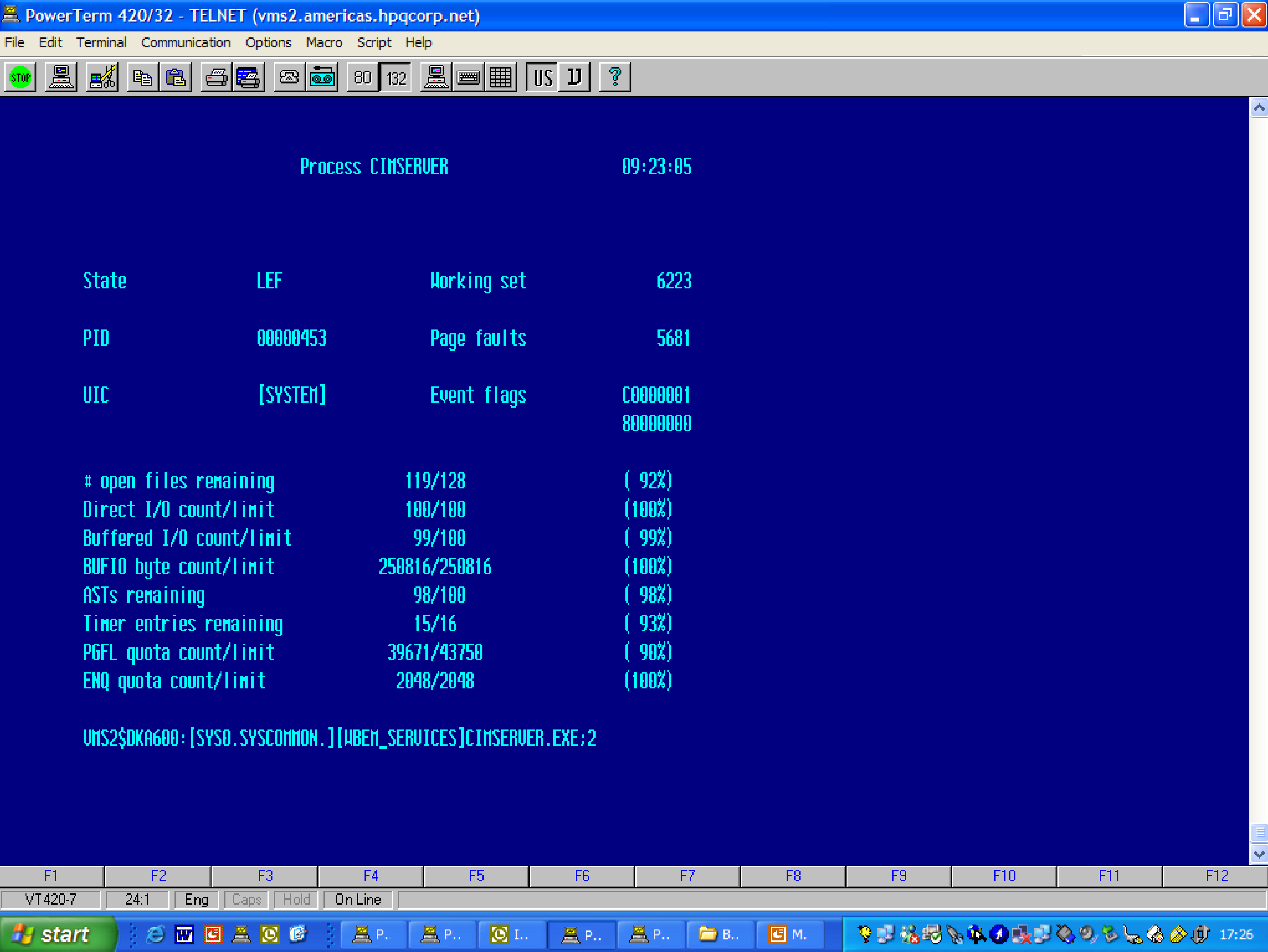
```
Device EWA0:, device type DEGXA, is online, network device, error logging is
    enabled, device is a template only.

    Error count                    1    Operations completed                  0
    Owner process                 ""    Owner UIC                      [SYSTEM]
    Owner process ID        00000000    Dev Prot              S:RWPL,O:RWPL,G,W
    Reference count                0    Default buffer size                 512
    Current preferred CPU Id       0    Fastpath                              1
    Current Interrupt CPU Id       0

Operating characteristics: Link up, Full duplex, Autonegotiation, Jumbo frames.

    Speed (Mbits/sec)           1000
    Def. MAC addr  00-D0-59-61-6A-B2    Current MAC addr      00-D0-59-61-6A-B2
```

SHOW DEVICE/FULL

```
$ sho dev/ful ewa5

Device EWA5:, device type DEGXA, is online, network device, error logging is
    enabled.

    Error count                    0    Operations completed                  2
    Owner process           "NETACP"    Owner UIC                      [SYSTEM]
    Owner process ID        39800425    Dev Prot              S:RWPL,O:RWPL,G,W
    Reference count                1    Default buffer size                1498

Operating characteristics: Full duplex, Autonegotiation, Jumbo frames.

    Speed (Mbits/sec)           1000
    Def. MAC addr  00-D0-59-61-6A-B2    Current MAC addr      00-D0-59-61-6A-B2
    Protocol name             DECNET    Protocol type                     60-03
```

SHOW PROCESS

- SHOW PROCESS/CONTINUOUS now supports the ‘Q’ option….
- ‘Q’ = Quota
  - While the continuous display is running, it is now possible to hit ‘Q’ and dynamically monitor the process quotas

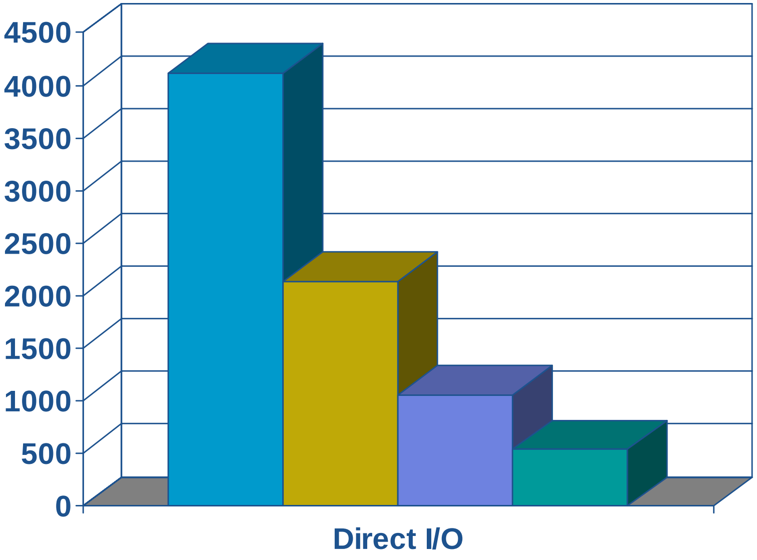


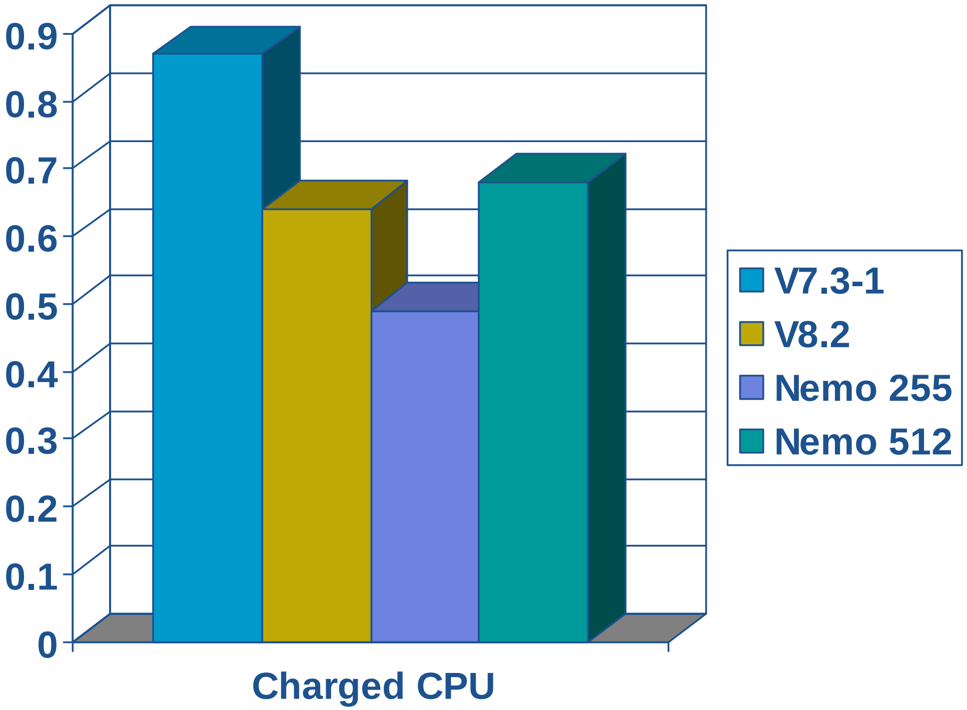
```
                                  Process CIMSERVER                     09:23:05



         State                LEF                 Working set               6223

         PID                  00000453            Page faults               5681

         UIC                  [SYSTEM]            Event flags           C0000001
                                                                        80000000

         # open files remaining            119/128                 ( 92%)
         Direct I/O count/limit            100/100                 (100%)
         Buffered I/O count/limit           99/100                 ( 99%)
         BUFIO byte count/limit         250816/250816              (100%)
         ASTs remaining                     98/100                 ( 98%)
         Timer entries remaining            15/16                  ( 93%)
         PGFL quota count/limit          39671/43750               ( 90%)
         ENQ quota count/limit            2048/2048                (100%)

         VMS2$DKA600:[SYS0.SYSCOMMON.][WBEM_SERVICES]CIMSERVER.EXE;2
```

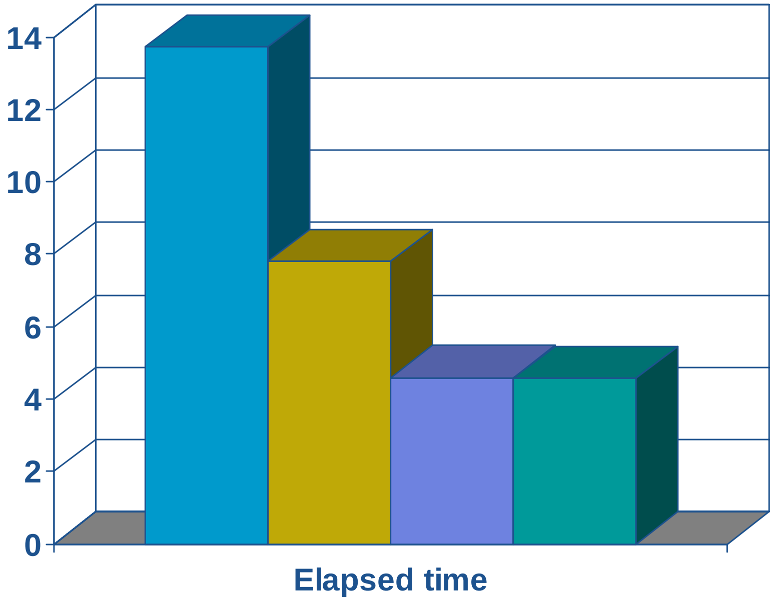
COPY

- V8.2 added the /BLOCK_SIZE qualifier
  - Default I/O size is 124 blocks
  - Maximum I/O size is 127 blocks

- V8.3 removes the I/O size limit
  - Copy has been modified to use RAB64
  - Can’t exceed maximum I/O size supported by the port driver
  - VCC_MAX_IO_SIZE







65MB file

DPWS/RZ29

DIFFERENCES

- DIFF/IGNORE=SPACE compresses multiple spaces and tabs down to one space before comparing
- **/IGNORE=WHITE_SPACE** removes all spaces and tabs before comparing
- In F$EDIT terminology, COMPRESS versus COLLAPSE
- Very useful when looking at code written by different people with different coding preferences

```
status = routine(a,b,c)
      Vs.
status = routine (a,b,c)
```

Lexical Functions

- F$LICENSE now supports 3rd party producers
  - Optional producer argument, DEC/HP assumed if omitted.

```
$ write sys$output f$license("PLI","KEDNOS")
TRUE
```

- F$CUNITS – New lexical function
  - F$CUNITS (number to convert, from_units, to_units)
  - The first argument is limited to 32bits
  - Currently only knows how to convert blocks to bytes
  - What else do you need?

```
$ write sys$output f$cunits(4432216,"blocks","bytes")
2.11GB
```

Lexical Functions

- F$MATCH_WILD
  - Performs wildcard matching between candidate and pattern string
  - Returns TRUE if the strings match
  - Syntax
    - F$MATCH_WILD (CANDIDATE, PATTERN)

```
$ write sys$output f$match_wild ("This is a candidate","*c%%d*")
TRUE
$
```

SEARCH / STATISTICS

- SEARCH/STATISTICS now defines several DCL symbols with statistics information

```
Files searched:               125       Buffered I/O count:       602
Records searched:           15575       Direct I/O count:         135
Characters searched:       842598       Page faults:               36
Records matched:               45       Elapsed CPU time:  0 00:00:00.26
Lines printed:                 97       Elapsed time:      0 00:00:02.87

$ sh sym search*
  SEARCH$CHARACTERS_SEARCHED = "842598"
  SEARCH$FILES_SEARCHED = "125"
  SEARCH$LINES_PRINTED = "97"
  SEARCH$RECORDS_MATCHED = "45"
  SEARCH$RECORDS_SEARCHED = "15575"
```

DIRECTORY & MAGTAPES

★To DIRECTORY….All blocks are created equal ☺

```
$ dir mkb100:[000000]/siz

        Directory MKB100:[]

        LEEHE.BCK;1                        520KB
        TEST1.BCK;1                        619KB
        TEST2.BCK;1                        619KB
        TEST3.BCK;1                         74KB

        Total of 4 files, 1.78MB

 $ dir mkb100:[000000]/siz

        Directory MKB100:[]

        LEEHE.BCK;1                       8.13MB
        TEST1.BCK;1                       9.67MB
        TEST2.BCK;1                       9.67MB
        TEST3.BCK;1                       9.17MB

        Total of 4 files, 36.65MB
```

MONITOR

- New “TOP” display
  - Top processes that use Kernel, Executive, Supervisor and User mode on the system
  - MONITOR PROCESS /TOPKERNEL
    /TOPEXEC…..
- Align class added to monitor alignment faults rate

```
                          OpenVMS Monitor Utility
                      ALIGNMENT FAULT STATISTICS
                            on node IPL31
                        3-JAN-2006 15:32:59.66

                                  CUR        AVE        MIN        MAX


Kernel Alignment Faults       1655.00    1613.00    1356.00    2068.00
Exec Alignment Faults         8525.00    8657.79    7499.00   10527.00
Super Alignment Faults           0.00       0.00       0.00       0.00
User Alignment Faults         1294.00    1267.20    1084.00    1628.00

Total Alignment Faults       11474.00   11538.00   10011.00   14223.00
```

Queue Manager

★ Increase batch queue job limit to 65535 (was 255)
  - V7.3-2 TIMA kit

- Performance enhancement to SYS$SNDJBC
  - Avoid heavy alignment faults (2000 faults per submit)
  - Pad the message being exchanged between the job controller and the queue manager
  - The old algorithm used when Queue Manager is running on non V8.3 version
  - In mixed version cluster run the Queue Manager on V8.3 node for optimal performance

65535 SPAWNed processes

★ Maximum number of spawned subprocesses increased to 65535

- Previous limit was 255
  - When this limit is reached - excessive CPU time (with spinlocks held) trying (and failing) to create more processes
- Numeric portion of the spawned process name increased from byte to word
  - ***Username portion may be truncated from 11 to 9 characters***
  - Set Bit 2 in DCL_CTLFLAGS to restore previous behavior (255 spawned process)
  - ***%DCL-S-SPAWNED, process GUY_47132 spawned***

General Enhancements (1 of 6)

- SYNCHRONIZE/TIME_OUT
  - Allows specifying the number of seconds to wait before terminating the SYNCH command

```
BLUSKY> submit looper
Job LOOPER (queue SYS$BATCH, entry 4) started on SYS$BATCH
BLUSKY> synch/entry=4/time_out=5
%QUEMAN-W-TMOEXP, timeout period expired
```

- New common qualifier keyword /SINCE=JOB_LOGIN
  - JOB_LOGIN is the login time of the master process in the job
  - PIPE creates a subprocess for each pipe segment therefore /since=login can’t be used in a PIPE

```
IPL31> pipe dir/sin=login | sea sys$input test
%SEARCH-I-NOMATCHES, no strings matched
IPL31> pipe dir/sin=job_login | sea sys$input test
TEST.TXT;1
```

General Enhancements (2 of 6)

- Assure SET LOGIN/INTERACTIVE succeeds during startup
  - No response from the console
  - No response for an interactive login attempt
    - Typically occurs when VMS$BASEENVIRON-050_VMS.COM terminated unexpectedly
- Case sensitivity support in cluster_config(_LAN).com
- DEASSIGN/NOLOG
  - The completion status will be set to success even if the logical name does not exist (instead of %SYSTEM-F-NOLOG)
  - No output is being displayed
- SHOW DEVICE
  - Performance enhancement to device scanning algorithm

General Enhancements (3 of 6)

- Maximum size of the DCL prompt has been increased to 64 characters (was 32)
  - Allows fancier prompts using escape sequences

- Target account for LMF compliance reports may be controlled by setting

  LMF$COMPLIANCE_CONTACT_ACCOUNT

★ Unlimited license support added to the Galaxy license server (GLX$LICENSE_SERVER)

★ SET COMMAND/RMS_RELATED_CONTEXT

General Enhancements (4 of 6)

- READ/WAIT
  - Wait if the record is currently locked by another stream
  - May be combined with /TIME_OUT
  - Sets RAB$V_WAT
- READ/KEY/MATCH={LT|LE}
  - READ/KEY only supports finding matching records with value equal (EQ), greater (GT), or greater or equal (GE) than a key
  - The new keywords add support for finding matching records with value less (LT) or less equal (LE) than a key

General Enhancements (5 of 6)

- ANALYZE/SSLOG
  - Support has been added for selecting entries based on CPU, kernel thread and Pthread IDs.
- SEARCH/WILDCARD_MATCHING
  - Two new keywords RELAXED and STRICT

```
IPL31> ty test.txt
first line
second line
third line

IPL31> sea test.txt "l*n"/wild=relax           ! V8.2 behavior, realx may
                                              be omitted
first line
second line
third line
IPL31> sea test.txt "l*n"/wild=strict          ! Asterisks not appended
                                                 match not found
%SEARCH-I-NOMATCHES, no strings matched
IPL31> sea test.txt "*l%n%"/wild=strict        ! Be a little more specific
                                                 and match found
first line
second line
third line
```

General Enhancements (6 of 6)

- ANALYZE/MEDIA/EXERCISE
  - Easy way for erasing media (pattern may be specified)
  - Default I/O size increased to 256 blocks
  - Cut the number of I/Os in half
- SHOW LICENSE/HIER**/ALL**
  - Displays all licenses defined in the Operating Environment Database (used to display loaded licenses only)
- B2B support in SHOW QUEUE

Agenda

- V8.3 new features
- ***Licensing changes***
- Prior releases
- Latest happenings with OpenVMS BACKUP

Current licensing policy

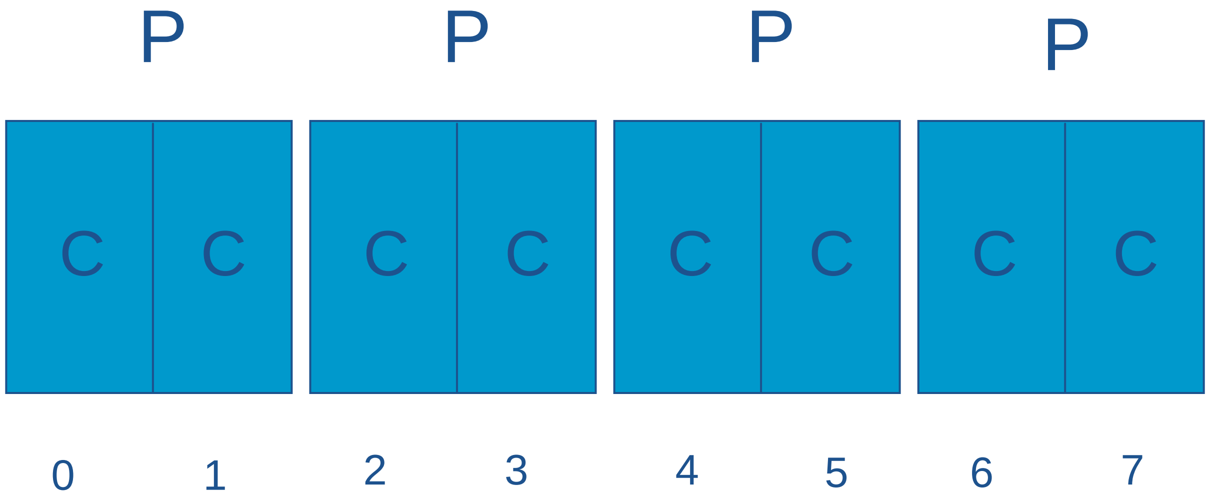
- HP licenses Integrity systems by the number of processors
  - 1 processor = 1 unit

```
$ show license/char
OpenVMS I64/LMF Charge Information for node NYANGA
This is an HP rx4640  (1.50GHz/6.0MB), with 4 CPUs active
This platform supports up to 4 CPU socket(s)
Type: PPL,   Units Required: 4  (I64 Per Processor)
```

HW Terminology – Processors/Cores

- The next chip generation, named Montecito, has 2 cores per processor
- The current rx2600 is 2P/2C
- Upgrading to Montecito will make it 2P/4C
- Upgrading the 4P/4C rx4640 will make it 4P/8C
- What happened to CPUs?
- From the OpenVMS viewpoint, what has always been seen as a CPU is now a core.
- $ SHOW CPU
- $ START CPU

OpenVMS Naming of a 4P/8C



Active CPUs: 0-7

New licensing policy

- HP is switching to license integrity systems based on the number of cores using new type of licenses – PCL
- Per Core License
- Each Core requires 1 PCL unit

```
$ show  license/char
OpenVMS I64/LMF Charge Information for node SD00
This is an HP SD64A  (1.50GHz/6.0MB), with 6 cores active
This platform supports up to 64 processor socket(s)
Type: PPL,   Units Required: 6  (I64 Per Processor)
Type: PCL,   Units Required: 6  (I64 Per Core)
```

- Cosmetic change only for non Montecito based systems
- 8P/16C RX7640 will require 16 PCL units

PCL

- PCL licenses may only be loaded on IA64
- PCL licenses may be managed by both Alpha & IA64
- PCL & PPL may be combined
- Full PAKGEN support

```
$ LICENSE REGISTER TEST_PCL_PAK -
/ISSUER=HP -
/AUTHORIZATION=TESTING123 -
/PRODUCER=HP -
/UNITS=50 -
/TERMINATION_DATE=1-FEB-2006  -
/OPTIONS=(IA64,PCL) -
/CHECKSUM=2-IYPC-LMEA-MEIF-MIRE
```

Release Vehicle

- PCL support ships with OpenVMS V8.3
  - Alpha & IA64
- Alpha – support for PCL management
  - VMS732_LMF-V0200
  - VMS82A_LMF-V0200
- IA64 – support for managing & loading PCL licenses
  - VMS82I_LMF-V0300
  - VMS821I_LMF-V0200

Agenda

- V8.3 new features
- Licensing changes
- ***Prior releases***
- Latest happenings with OpenVMS BACKUP

TIMA kits

- ***VMS821I_LIBRTL-V0100***
  - Prerequisite for RDB on IA64
  - Includes a change to LIB$VM_MALLOC which is called by the CRTL malloc() function

TIMA kits

- ***VMS732_DCL-V0700***
  - New optional format keywords for F$DELTA_TIME
    - ASCTIM (default)
    - DCL
  - SPAWNed process limit increased to 65535 per username (more on that later)
  - Fixes memory leak when encountering FNF error

Translated Images

- Upgrade to V8.3
  - Critical Translated Image Environment (TIE) fixes
    - Fixes are in the area of calling native IA64 routines
- New Binary Translator – V2.0
  - Bug fixes
  - Performance improvements
  - Support Pascal images

Latest Happenings with OpenVMS BACKUP

BACKUP & DVE

- OpenVMS V7.3-2 added Dynamic Volume Expansion (DVE) support
- Two new terms introduced: Logical Volume size and Expansion Volume size
- Controlled by the /SIZE & /LIMIT qualifiers to the INITIALIZE & SET VOLUME commands

```
Error count                    0    Operations completed              8936
 Owner process          "_VTAT7:"    Owner UIC                   [NPAR_BUILD]
 Owner process ID        2817CDDC    Dev Prot             S:RWPL,O:RWPL,G:R,W
 Reference count                2    Default buffer size                 512
 Current preferred CPU Id       0    Fastpath                              1
 Total size               33.91GB    Sectors per track                    96
 Total cylinders             7719    Tracks per cylinder                  96
 Logical Volume Size       2.38GB    Expansion Size Limit           190.96GB
 Allocation class               5
```

BACKUP & DVE

- BACKUP had no knowledge about DVE
  - DVE characteristics of a device were lost when image backup performed
- Customers had to manually set DVE characteristics (assuming they noticed it was lost ;-)
- BACKUP now fully supports DVE
  - VMS732_BACKUP-V0600

BACKUP & DVE – Expansion size

- The volume expansion size is being recorded in the save-set header
- BACKUP/LIST displays the expansion size if it exists in the save-set
- When restoring a save-set (using /image) or performing disk-to-disk image backup, the target device inherits the expansion size limit of the input device
- New qualifiers
  - /IGNORE=LIMIT prevents the target device from inheriting the expansion size
  - /LIMIT=n added to allow overriding the expansion size stored in the save-set header
    - Corresponding to $INIT/LIMIT

BACKUP & DVE – Logical size

- By default logical size is not preserved
  - Restoring image backup of 4GB disk to a 36GB disk will only result in 4GB of usable disk space
- BACKUP/SIZE
  - Instructs BACKUP to preserve the logical volume size during a restore operation
- BACKUP/SIZE=n
  - Instructs BACKUP to initialize the target device to have a logical volume size of n

BACKUP & DVE – the fine prints

- Did you know….
  - BACKUP/NOINIT initializes the target device…
    - Yes this is not a mistake !

- DVE characteristics will not be preserved if /NOINIT is specified
  - The target device is mounted foreign and we can’t retrieve the logical volume size and the expansion size
    - Use /LIMIT & /SIZE if you must

```
IPL31> back IA64:[KITS]I64XB3X.BCK/sav dka100:/ima/noini
%BACKUP-I-LOGNOTPRES, logical volume size of volume DKA100: not preserved
%BACKUP-I-LIMITNOTPRES, expansion size limit of volume DKA100: not preserved
```

Encryption support

- Starting with OpenVMS V8.2 the Encrypt product is covered by the base O/S license
  - No separate license required
- Starting with V8.3 the Encrypt product is integrated into the base O/S
  - No separate installation required
- BACKUP supports creating encrypted save-sets using the /ENCRYPT qualifier

Encryption support

- How does it work?
  - At run-time, BACKUP generates a random encryption key used for encrypting the save-set records
    - Random (time based) string is XOR'd with the user's command and encrypted against itself
  - The encryption key is being encrypted using a user provided key and stored in the save-set header
  - Decryption – the encryption key is retrieved by decrypting the key stored in the header using the user provided key
  - BACKUP/ENCRYPT=(…)
    - NAME
    - ALGORITHM
    - VALUE

Encryption support

- Existing limitations
  - Uses DESCBC algorithm (old, slow & expensive)
  - User provided algorithm is only used for encrypting the key
    - DESCBC is still used for encrypting the data

- V8.3 adds AES encryption support to BACKUP
  - Modern & stronger encryption
  - User provided algorithm is used for encrypting the data
  - DESCBC is still used by default (to maintain backward compatibility)
  - The following algorithms supported by BACKUP:
    - **AESCBC128**, AESCBC192, AESCBC256, AESECB, AESCFB and AESOFB

Encryption support - examples

- Create an encrypted save-set, specify the encryption key at run-time

```
$ backup *.com coms.bck/sav/encrypt=alg=aes
Enter key value:
Verification:
```

- Create an encrypted save-set, create the encryption key from DCL

```
$ encryp/create_key guy "This is a nice key called guy"/aes
$ backup *.com coms.bck/sav/encrypt=(alg=aes,name=guy)

$ backup coms.bck/sav […] /encrypt=(alg=aes,name=guy)
```

Conflict in verbs

- Starting with V8.3 the DECRAM verb has been removed
  - Avoid conflict between DECRAM & DECRYPT

- Any procedure using the DECRAM verb should be modified to use a foreign command
  - $DECRAM == “$MDMANAGER”

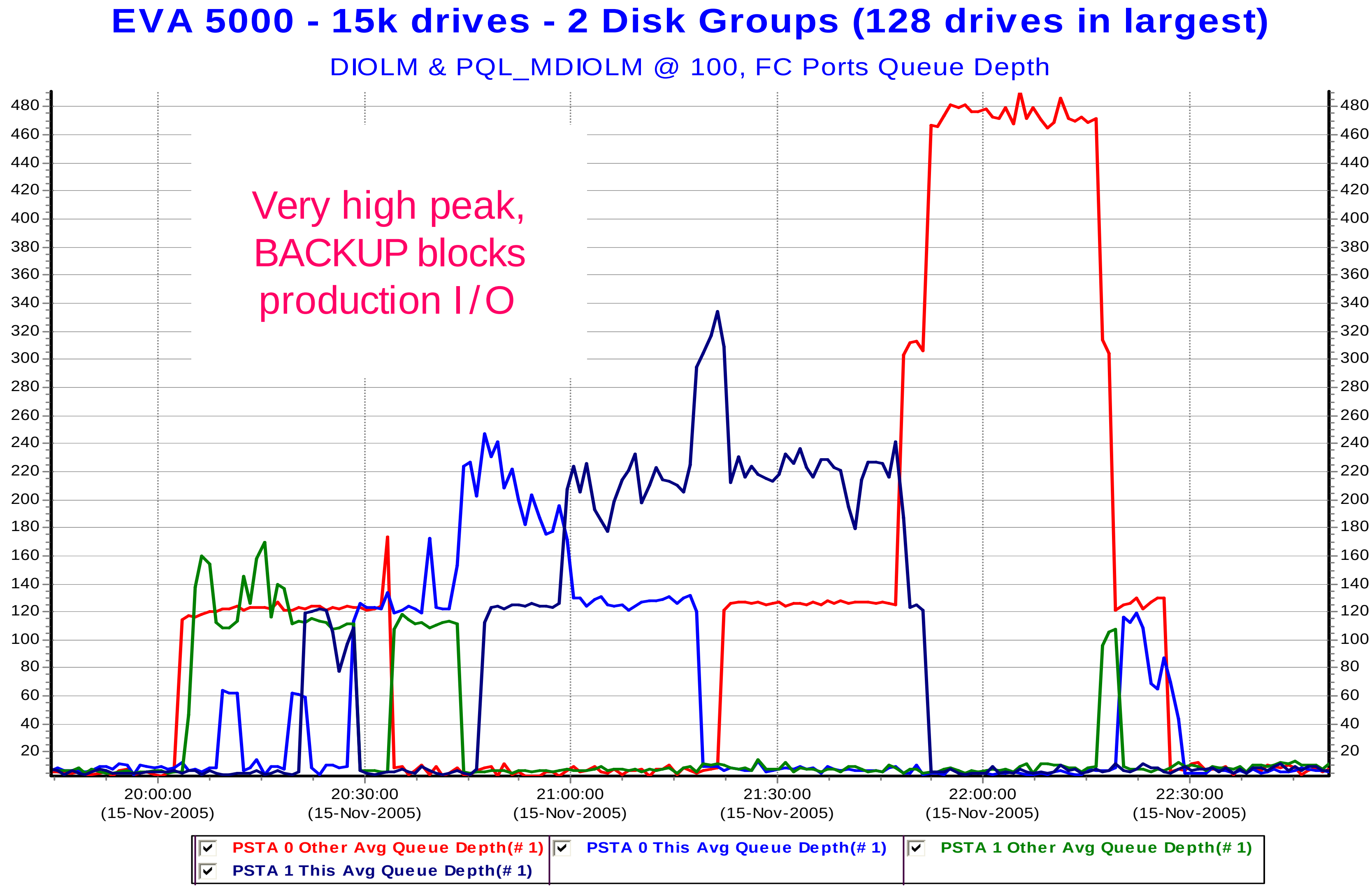
Disk Queue Load

- BACKUP uses a large buffer pool to read file data in an optimized manner
- Algorithm was designed 20 years ago
- All reads for the entire buffer pool are issued concurrently
- Over time buffer pools have gotten larger & the I/O subsystem’s tolerance of being flooded with large number of I/Os has decreased
  - Especially true with EVA & XP storage controllers
- A picture is worth a thousand words….

Impact of DIOLM on Storage Arrays

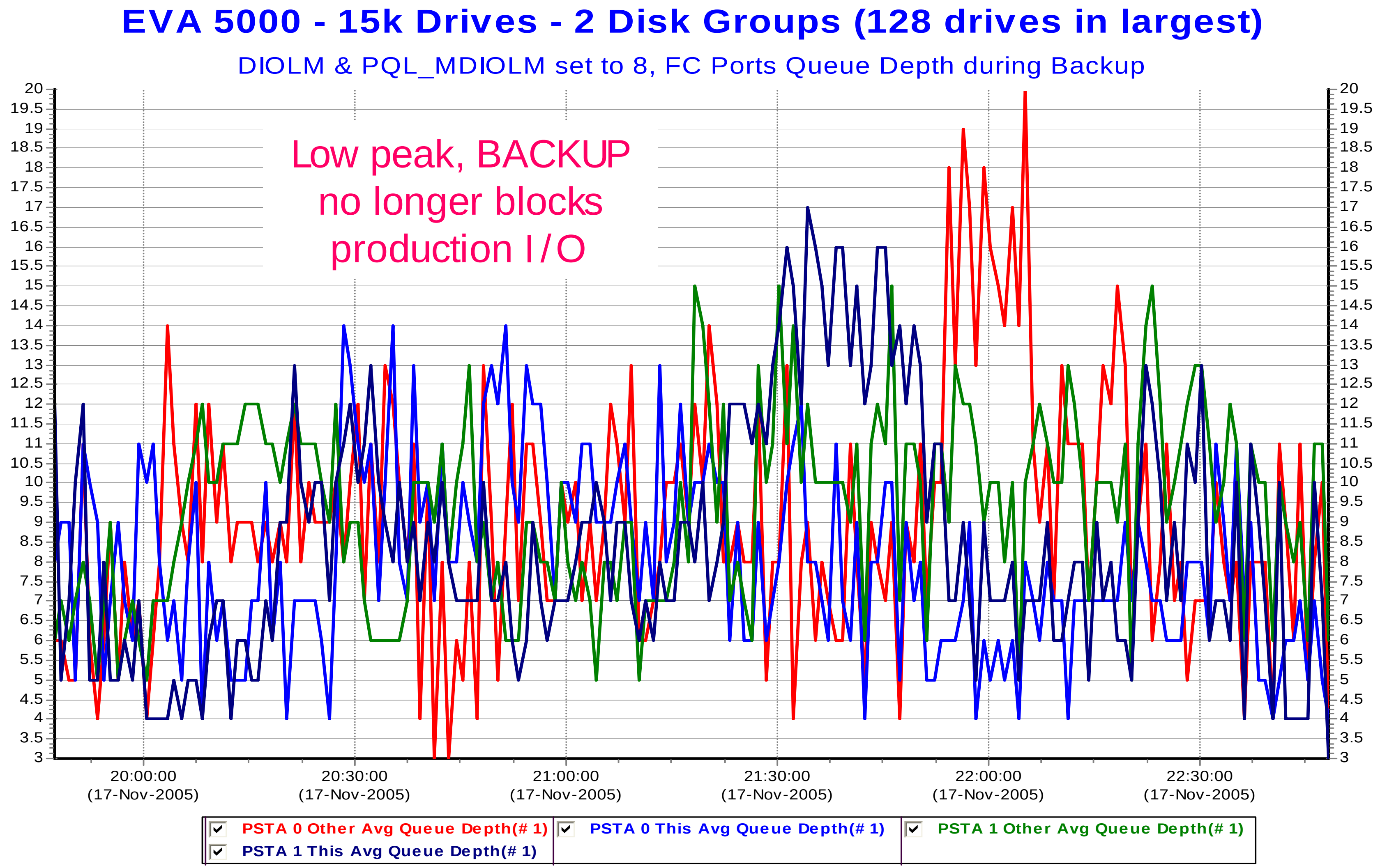
- Customer’s production environment experienced
  - Poor performance during BACKUP jobs
  - Using latest EVA VCS, Drive and HBA firmware
  - Using recommended DIOLM values of 100
- Recommended dropping DIOLM to only 8
  - After system startup
  - Set DIOLM on the BACKUP account to 8
  - Before BACKUP jobs set PQL_MDIOLM to 8
  - After BACKUP jobs return PQL_MDIOLM to 100
- Performance better by an order of magnitude !!

FC Queue Depth



Very high peak, BACKUP blocks production I/O

FC Queue Depth – Better Balance



Low peak, BACKUP no longer blocks production I/O

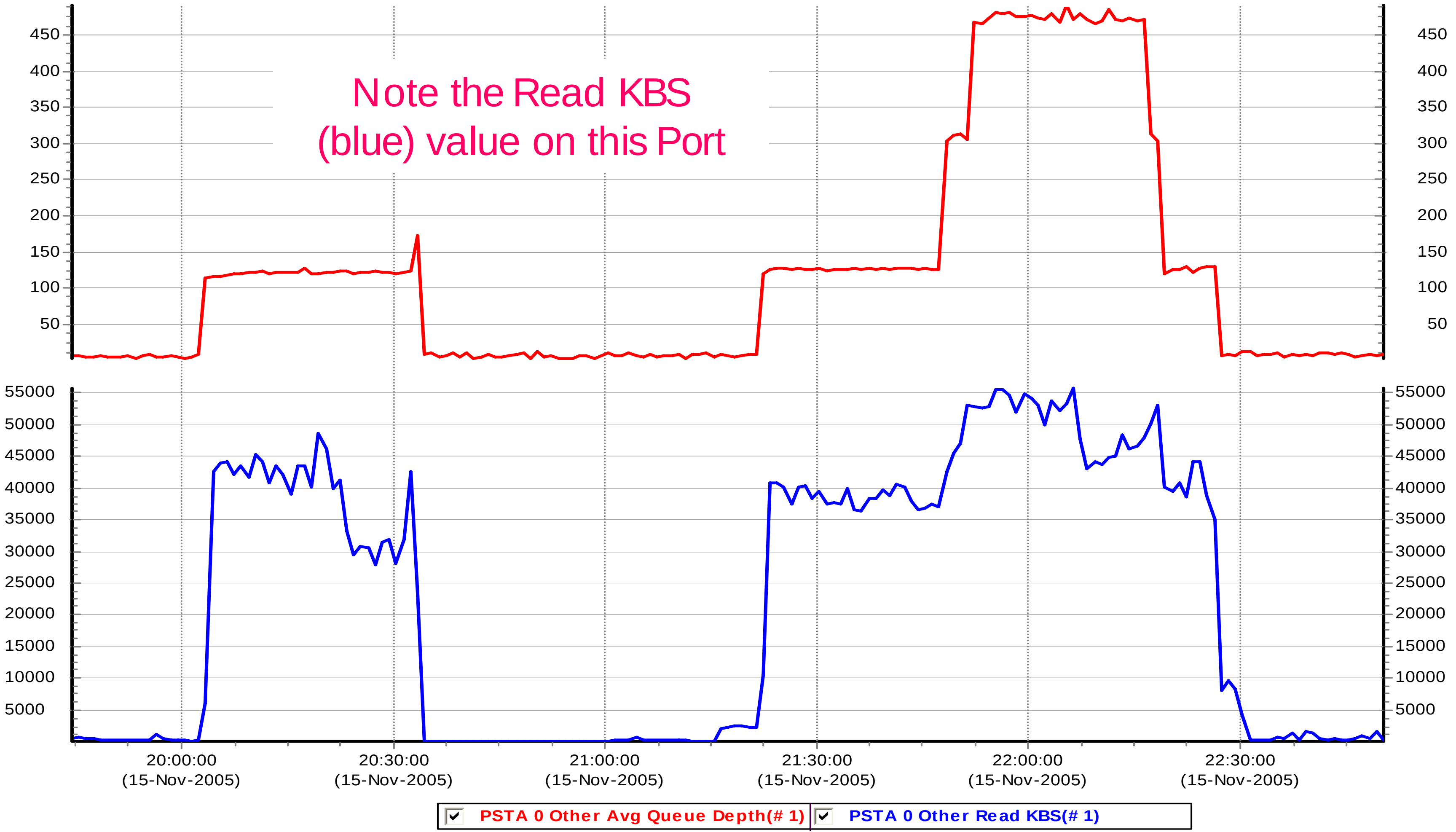
FC Queue Depth – Conclusions

- A SAN Storage Controller is a shared resource:
  - Production and backup often use the same
    - FC Ports
    - Cache
    - Back end
  - Heavy activity by one host can impact all hosts
    - Heavy queue on FC port delays I/O for other hosts
    - Heavy use of queue decreases cache effectiveness for all
    - Heavy back end usage decreases availability for all
- When possible, keep the FC Port Queue Depth as low as possible. DIOLM of 8 decreases demand, but keep tapes streaming.

Port Queue Depth vs. Read KBS

EVA 5000 - 15k drives - 2 Disk Groups (128 drives in largest)

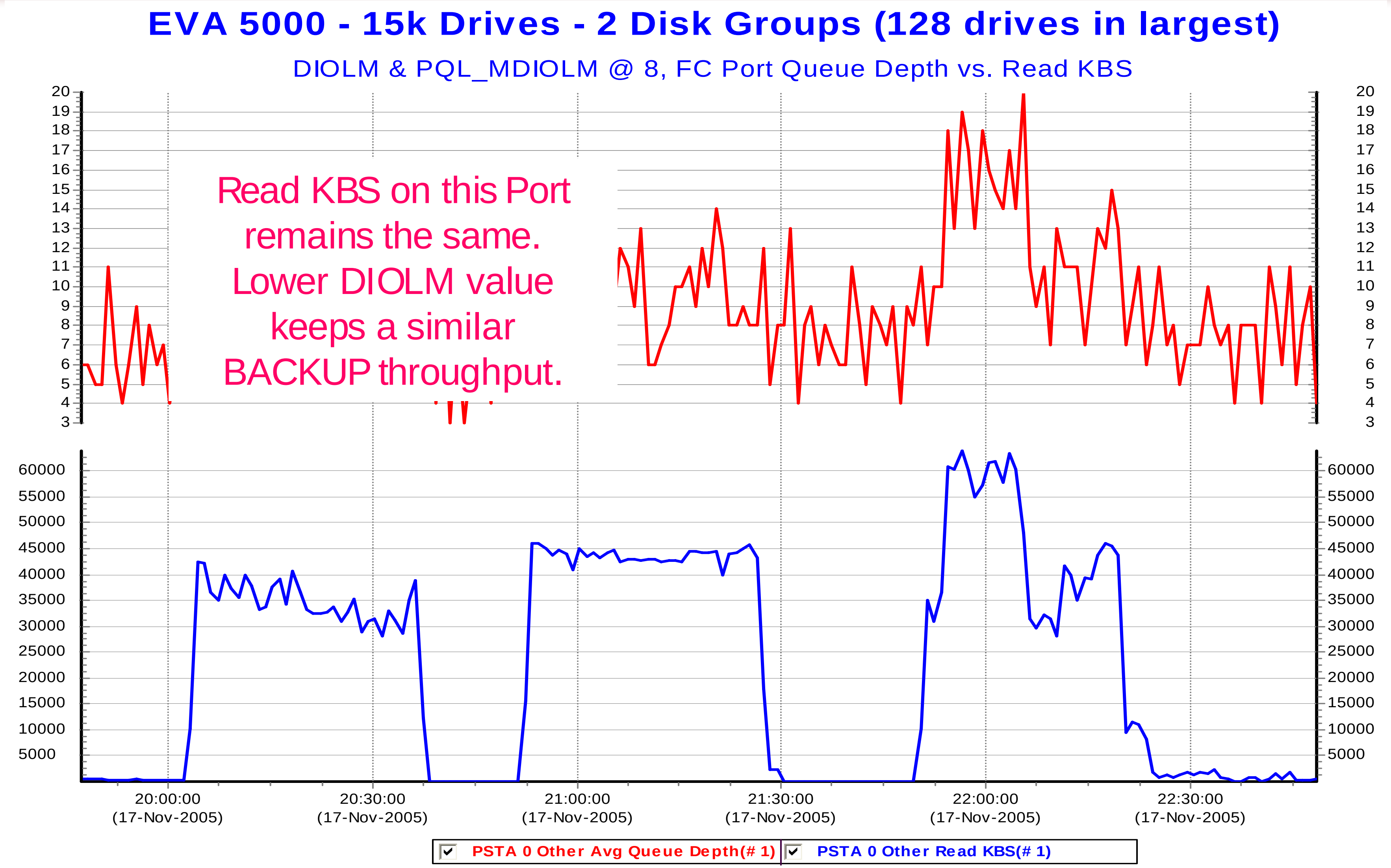
DIOLM & PQL_MDIOLM @ 100, FC Port Queue Depth vs. Read KBS



Note the Read KBS (blue) value on this Port

BRUDEN · OSSG
On-Shore Systems Group

Port Queue Depth vs. Read KBS



Read KBS on this Port remains the same. Lower DIOLM value keeps a similar BACKUP throughput.

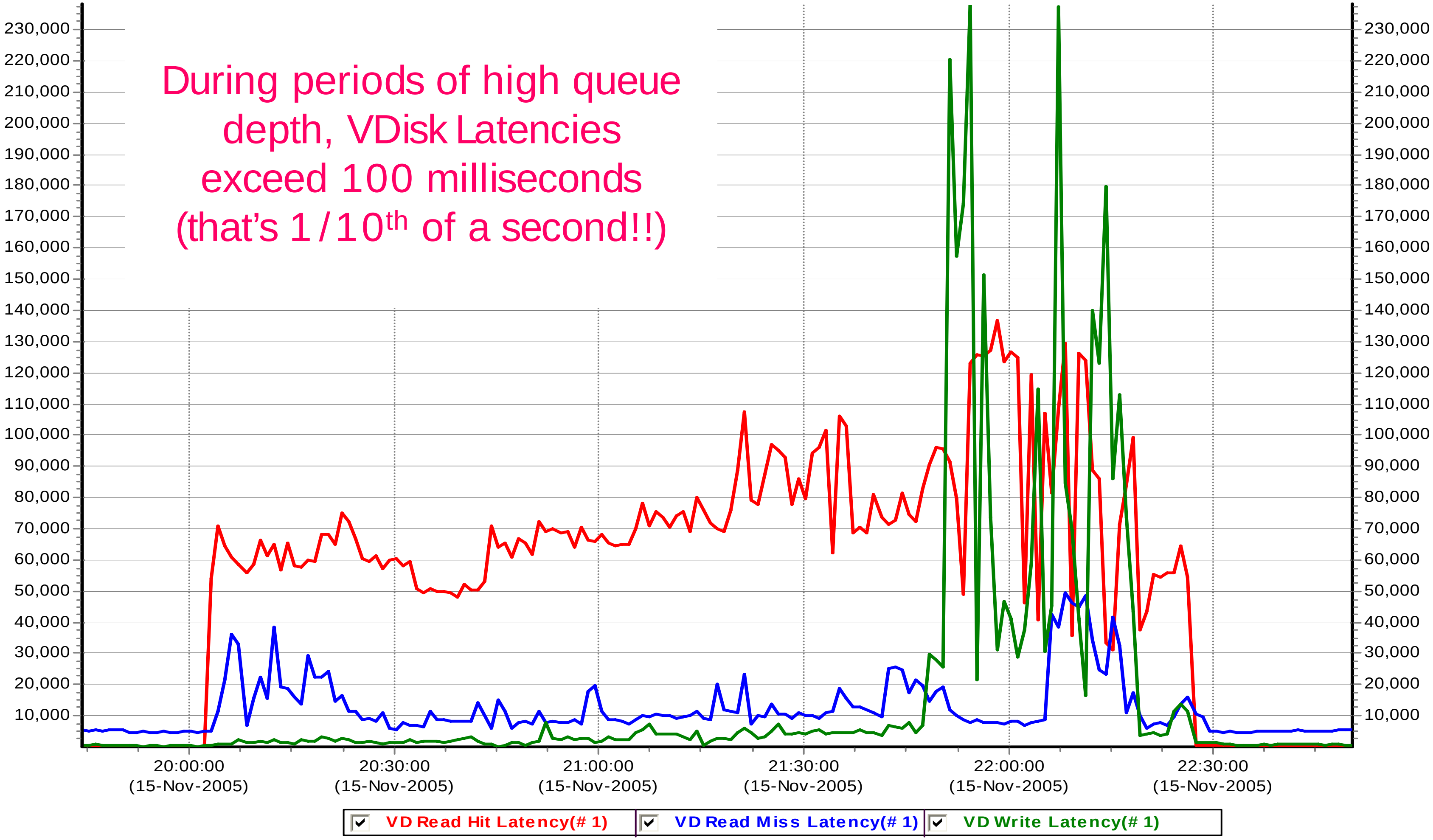
QDepth vs. Read KBS – Conclusions

- QDepth at DIOLM 8 is only 1/20th of the queue depth when DIOLM is set to 100.
  - DIOLM at 100 = QDepth of 400
  - DIOLM at 008 = QDepth of 020
- Yet, the Read KBS remains the same. OpenVMS does not issue all the I/Os at once. However, it can issue enough to keep the data moving at the same rate.
  - DIOLM at 100 = Read KBS at 40-50MB/sec
  - DIOLM at 008 = Read KBS at 40-50MB/sec

Impact of High FC Queue Depth

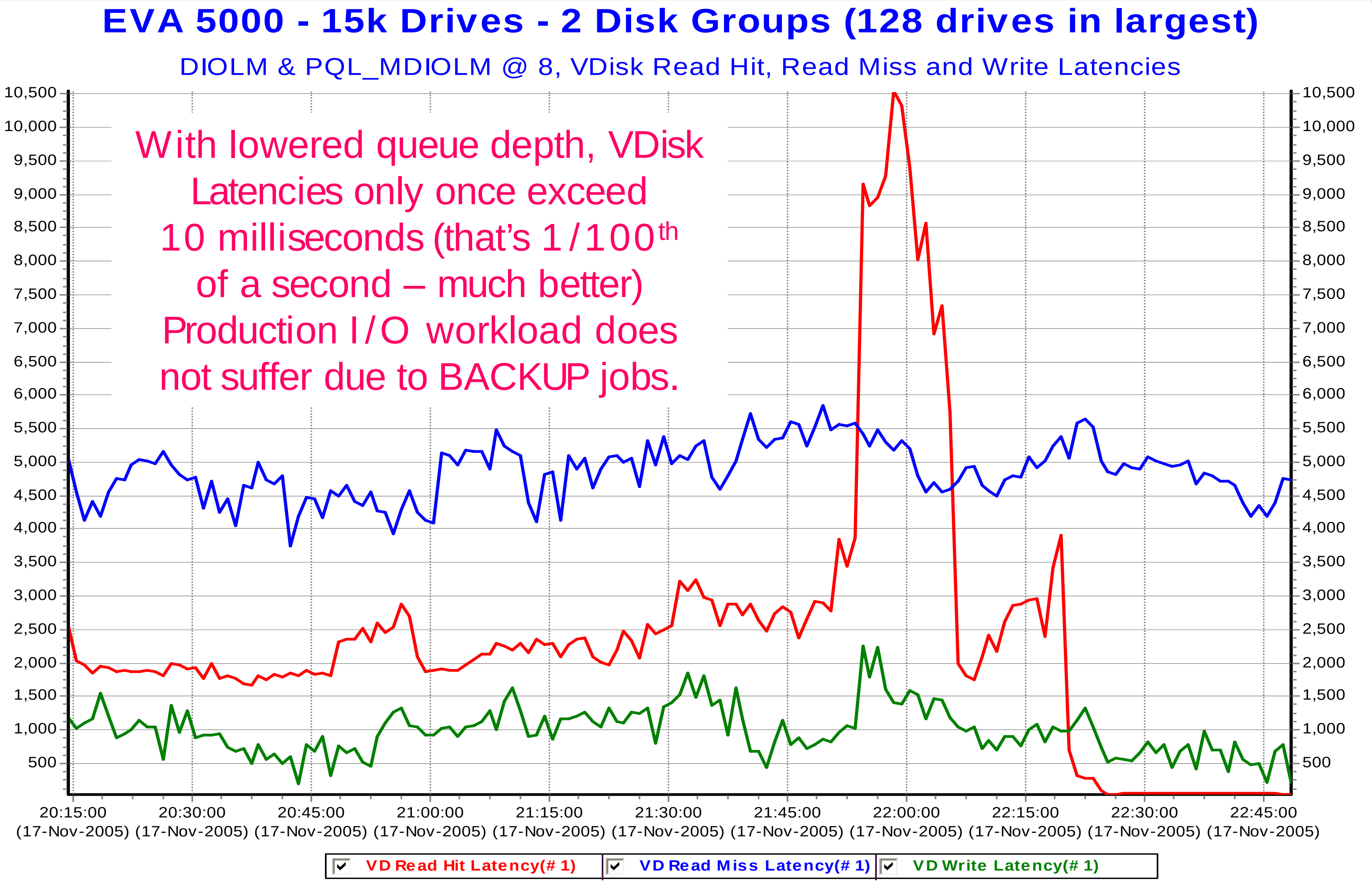
EVA 5000 - 15k drives - 2 Disk Groups (128 drives in largest)

DIOLM & PQL_MDIOLM @ 100, VDisk Read Hit, Read Miss and Write Latencies



During periods of high queue depth, VDisk Latencies exceed 100 milliseconds (that's 1/10th of a second!!)

Impact of Lowered FC Queue Depth



With lowered queue depth, VDisk Latencies only once exceed 10 milliseconds (that's 1/100th of a second – much better) Production I/O workload does not suffer due to BACKUP jobs.

VDisk Latencies – Conclusions

- Latencies with DIOLM at 8 stay below 10ms.
  - DIOLM at 100 = Latencies significantly exceed 100ms
  - DIOLM at 008 = Latencies stay below 10ms
- Read Miss Latency achieves an excellent 5ms
- Read Hit Latency achieves an outstanding 2.5ms
  - Some spikes as extra BACKUP jobs start
  - Even under added load, the latencies remain acceptable
- Write Latency stays at 1ms.
  - Compare that to more than 200ms during peak when DIOLM was at 100.

Disk Queue Load

- Something had to be done….
- *Old behavior*
  - *issue as much I/Os possible allowed by DIOLM*
  - *Continue issuing I/Os until we hit SS$_EXQUOTA*
  - *Wait for I/Os to complete and flood the I/O subsystem again*
- *The New algorithm*
  - *Issue the disk reads in n parallel AST threads*
  - *The completion AST of each thread issues the next I/O*
  - *By default use 8 parallel I/O threads*
  - *Number of threads controlled by new /IO_LOAD qualifier*

Disk Queue Load

- Better performance
  - Idle EVA controller showed 15% reduction in elapsed time
  - Results are not linear – busy controllers will witness more significant (dramatic !) improvement
  - Direct attached SCSI disk showed ~5% improvement
  - YMMV

- RMS optimization when writing a save-set to disk
  - set the WBH & RAH bits

30% reduction in elapsed time

Improved CTRL-T information

- Traditional CTRL-T information showed the name of the current file being saved/restored and the total number of save-set blocks processed

```
$ backup IA64:[KITS]I64XB37.BCK/sav $5$dka100:/ima
MIKAXP::_VTA61: 14:36:38 BACKUP    CPU=00:00:00.86 PF=908 IO=2192 MEM=256
 Restoring file: $5$DKA100:[DWMOTIF_SUPPORT_I64XB37.KIT]HP.SI$COMPRESSED;1
 Saveset volume:1, saveset block:266 (32256 byte blocks)
```

- Can you tell how much data restored so far?
- Can you tell when the restore will be done?
- Something had to be done…
- ***Introducing the new & improved CTRL-T***

Improved CTRL-T information

```
IPL31::_VTAT7: 14:46:27 BACKUP    CPU=00:00:03.40 PF=6298 IO=18408 MEM=465
 Restoring file: DKA100:[DWMOTIF_SUPPORT_I64XB37.KIT]HP.PCSI$COMPRESSED;1
 Saveset volume:1, saveset block:720 (32256 byte blocks)
 22.14MB restored out of 1.18GB, 1% completed
 Restore rate: 965KB/sec, estimated completion time: 15:07:31.85
```

Improved CTRL-T information

- Note the message says ESTIMATED !!!
- Fancy CTRL-T message displayed when
  - Restoring a saveset
  - Creating an image backup
  - Counters are updated when a file marked nobackup is encountered
- When the amount of data to be saved is unknown, only the rate and total amount of data processed so far is displayed
- CTRL-T can not be used in batch….and BACKUP is usually running in batch…

BACKUP/PROGRESS_REPORT=n

- /PROGRESS_REPORTS writes CTRL-T style message to the output device every given interval
- n is the number of seconds between intervals

```
$ back IA64:[KITS]I64XB37.BCK/sav dka100:/ima/progress=10
%BACKUP-I-PROGRESS, progress report generated at  4-JAN-2006 15:00:54.47
 Restoring file: DKA100:[DWMOTIF_SUPPORT_I64XB37.KIT]HP1.PCSI$COMPRESSED;1
 Saveset volume:1, saveset block:170 (32256 byte blocks)
 5.22MB restored out of 1.18GB, 0% completed
 Restore rate: 535KB/sec, estimated completion time: 15:39:28.28
```

CRC

- Tape drives are getting faster….
  - Ultrium-960 can write @160 MB/sec
- BACKUP completes faster but during this time the CPU gets overloaded (calculating CRC)
  - 90% CPU utilization on DS25 writing to Ultrium-460 drive (@ 40MB/sec)
  - May impact the availability of other applications on the system

CRC

- Performance enhancement made to LIB$CRC
  - 30% - 50% reduction in CPU consumption
  - ~50% increase in throughput

```
$ r crc2
500 buffers of size = 32768 bytes
 lib$crc latency   228.6628 msec
Total bytes processed = 16384000
Rate =     68.3321 Mbytes/sec

$ r crc2
500 buffers of size = 32768 bytes
 lib$crc latency   152.2836 msec
Total bytes processed = 16384000
Rate =    102.6046 Mbytes/sec
```

- This is a short term solution…..

“lost” saveset attributes

- A saveset transferred using FTP or compressed and decompressed using ZIP will lose it’s RMS attributes
- An attempt to process the saveset will fail
  - %BACKUP-F-NOTSAVESET

- Fortunately the correct RMS settings are stored in the saveset header
- Many procedures for fixing this are floating around…..and now….drum roll please…… BACKUP can do it out of the box

BACKUP / REPAIR

- /REPAIR instructs BACKUP to attempt and restore the correct RMS attributes

```
$ backup images.bck/sav [.exes]/repair
%BACKUP-I-REPAIRED, saveset attributes changed to RFM=FIX, MRS=32256
IPL31::GUY 14:58:58 BACKUP    CPU=00:00:10.89 PF=7765 IO=71628 MEM=409
Restoring file: UPS$:[000000.EXES]BACKUPSHR.EXE;16
```

Misc. updates

- BACKUP$_STARTVERIFY, STARTRECORD and STARTDELETE modified to include the current time

```
%BACKUP-I-STARTVERIFY, starting verification pass at  4-JAN-2006 15:13:19.88
```

- When performing image backup of 18GB disk, VAX will fail to generate a valid boot block on the target device
  - Occurs when VMB.EXE lives on LBN 16777216 (or higher)
  - Fix is available

Potential future projects

- More performance improvements
- Larger tape blocks
- Larger disk reads
- Tape shadowing
- Performance monitoring (run-time stats)
- Preserving boot options

Project X

Compression support

- We have been toying with the idea of adding compression support to BACKUP
- No common tool for compressing data
    - ZIP does not support files exceeding 4GB
    - SPOOL was not ported to IA64
- The size of the VMS kit is getting larger and larger and will soon exceed the size of a single DVD
- Became an emergency when we needed to ship FT version of V8.3
    - Resulted in the port of gZIP to OpenVMS
- We went shopping for a compression algorithm

Compression Benchmark

- 1.19GB save-set containing the XB49 kit
  - Compressed using ZIP – 600.63MB
  - Compressed using LZW – 823.21MB
  - Compressed using ZLIB – 602.45MB

- 28.03MB PCSI kit
  - Compressed using PCSI – 13.33MB
    - DCX
  - Compressed using ZIP – 5.31MB
  - Compressed using ZLIB – 5.30MB

ZLIB

- Free
- Legally unencumbered
- May be included in commercial applications
- Lossless data compression library
- Never expands the data
- Unlike DCX may be used to compress stream of bytes
  - Does not need to analyze the file in advance
- Maximum compression factor 1:1000
  - 50MB file filled with zeros compressed to 49KB
  - Realistic numbers are in the range of 2:1 to 5:1

- http://www.zlib.net

Compression support in BACKUP

- ZLIB routines are shipping in a new shareable image
  - SYS$LIBRARY:COMPRESS$SHR.EXE
  - SYS$LIBRARY:COMPRESS$SHR_EV6.EXE (alpha only)
- New qualifier - /DATA_FORMAT=COMPRESSED
  - Instructs BACKUP to create a compressed save-set
  - Does not need to be specified on the restore command
- Initially viewed as a feature to save space….but turned out to be a significant performance improvement

Compression support in BACKUP

- The slower the I/O subsystem the bigger the performance win
  - Big win for network operations
  - Big win for MSCP served devices
  - doubles the CPU consumption
- On average BACKUP completes 2-5 times faster
- 800MB dump file was compressed to 75MB
- No support for writing compressed savesets to tapes
- Do not encrypt & compress…..at least for now

Compression support in BACKUP

- Compressed save-sets have variable length records
  - Instead of fix
- First record is not compressed
  - Detect compressed savesets
  - Force largest record = value of /BLOCK_SIZE
- To fix attributes after file transfer
  - SET FILE X.CBCK/ATTRIB=(RFM=VAR,LRL=32256,MRS=32256)

```
IPL31> dir cxx*.exe/siz=all

Directory SYS$COMMON:[SYSEXE]

CXX$COMPILER.EXE;1   39.68MB/39.68MB
CXX$DEMANGLE.EXE;1      41KB/42KB
CXX$LINK.EXE;1         166KB/166KB

Total of 3 files, 39.89MB/39.89MB

IPL31> backup cxx*.exe cxx.bck/sav/log
%BACKUP-W-NOFILES, no files selected from DSA5:[SYS0.][SYSEXE]CXX*.EXE;*
%BACKUP-S-COPIED, copied DSA5:[SYS0.SYSCOMMON.][SYSEXE]CXX$COMPILER.EXE;1
%BACKUP-S-COPIED, copied DSA5:[SYS0.SYSCOMMON.][SYSEXE]CXX$DEMANGLE.EXE;1
%BACKUP-S-COPIED, copied DSA5:[SYS0.SYSCOMMON.][SYSEXE]CXX$LINK.EXE;1

IPL31> dir cxx.bck/siz

Directory SYS$SYSROOT:[SYSEXE]

CXX.BCK;1            44.60MB

Total of 1 file, 44.60MB
```

Compression support in

```
IPL31> dir cxx*.exe/siz

Directory SYS$COMMON:[SYSEXE]

CXX$COMPILER.EXE;1   39.68MB
CXX$DEMANGLE.EXE;1      41KB
CXX$LINK.EXE;1         166KB

Total of 3 files, 39.89MB

IPL31> backup cxx*.exe cxx.cbck/sav/data=comp/log
%BACKUP-W-NOFILES, no files selected from DSA5:[SYS0.][SYSEXE]CXX*.EXE;*
%BACKUP-S-COPIED, copied DSA5:[SYS0.SYSCOMMON.][SYSEXE]CXX$COMPILER.EXE;1
%BACKUP-S-COPIED, copied DSA5:[SYS0.SYSCOMMON.][SYSEXE]CXX$DEMANGLE.EXE;1
%BACKUP-S-COPIED, copied DSA5:[SYS0.SYSCOMMON.][SYSEXE]CXX$LINK.EXE;1
%BACKUP-I-COMPRESS, data compressed by 66%

IPL31> dir cxx.cbck/siz

Directory SYS$SYSROOT:[SYSEXE]

CXX.CBCK;1           14.82MB
```

Nous avons resolu les problemes du monde

Questions?

BRUDEN-OSSG thanks you for attending this session.

See us at www.BRUDENOSSG.com for:

- *Performance analysis*
  - *(Performance Results Or No Expense)*
- *Porting assistance*
- *Special OPS (OpenVMS Programming Services)*