



# ***Healthy System Means Healthy System Manager***

vision

*Presented by*

*Lori Spencer*

[www.parsec.com](http://www.parsec.com) | 888-4-PARSEC



vision

**To Download this Presentation, please visit:**  
**<http://www.parsec.com/public/HealthyManager.pdf>**

**To E-mail Lori**  
**[Lspencer@parsec.com](mailto:Lspencer@parsec.com)**

**[www.parsec.com](http://www.parsec.com) | 888-4-PARSEC**



**PARSEC Group**  
**Our Trainers Consult. Our Consultants Train.**



# Topics

- Objective
- Monitoring Uptime
- Volume Management
- Security Auditing
- Accounting Log File
- Questions





# Objective

This presentation takes the system manager beyond the typical DCL SHOW command, although some will be presented along with some DCL command procedures, but to help system manager understand concepts and utilities that normally would not be implemented.





# Monitoring Uptime

- Monitoring uptime goes beyond the typical SHOW command we will look at:
  - What should you do if the system crashes?
  - Why didn't the system reboot?
  - What should I do if the system doesn't respond?



# Monitoring Uptime Topics

- Commands to monitor uptime
- Why did the system reboot?
- System will not reboot!
- System is not responding
- System Crashed, now what?





# Command to Monitor Uptime

DCL examples:

Can be put in your LOGIN.COM

```
$ show system/noprocess/output=up.lis
$ open upt up.lis
$ read upt val
$ uptime = f$extract(54,999,val)
$ write sys$output "System ''uptime'"
$ close upt
$ purge up.lis
$ exit
```

Or use the PIPE command:

```
$ PIPE show system/noprocess | (read sys$input val ; -
  write sys$output -
    "System " + f$extract(f$locate("Uptime",val),999,val))
$ exit
```



# Commands to Monitor Uptime

Examples:

```
$ show system/noprocess
```

```
OpenVMS V8.3  on node CLASS2  29-MAR-2007 11:24:18.87  Uptime  5 21:54:47
```

```
$ show system/noprocess/cluster
```

```
OpenVMS V8.3  on node CLASS2  29-MAR-2007 11:24:22.48  Uptime  5 21:54:50
```

```
OpenVMS V8.3  on node CLASS8  29-MAR-2007 11:24:22.49  Uptime  0 18:31:05
```

```
OpenVMS V7.3  on node CLASS9  29-MAR-2007 11:24:22.53  Uptime  6 01:59:23
```

```
OpenVMS V7.3-2 on node JOKER  29-MAR-2007 11:24:22.54  Uptime  2 00:53:21
```

```
OpenVMS V8.2  on node YIPPIE  29-MAR-2007 11:24:22.56  Uptime  2 00:16:33
```

- Notice node CLASS8 has only been up 18 hours, you will need to determine if it crashed, rebooted itself, or a schedule reboot was performed.



# Why did the System Reboot?

- Schedule reboot
  - Schedule reboots should be coordinated through the system manager, management and users.
  - These should not be an issue, but if it is, then you have a employee issue, not an OpenVMS issue!



# Why did the System Reboot?

- Unscheduled reboot
  - 90% are accidental.
  - Mostly happens when someone at the console terminal accidentally hits a CTL-P and then panics!
  - DO NOT panic, just enter CONTINUE at the >>> prompt within a timely fashion. Most systems will recover.
  - Power outages also cause unscheduled reboots.
  - System crashes! Unlikely, because we all know that OpenVMS doesn't crash!
  - But if it did you need to determine why!



# Continuing a System from the Console

Examples:

```
$ show system/noprocess
```

```
OpenVMS V8.3  on node CLASS2  29-MAR-2007 16:23:51.52  Uptime  0 00:04:35
```

```
$ CTL-P
```

```
halted CPU 0
```

```
CPU 1 is not halted
```

```
CPU 2 is not halted
```

```
CPU 3 is not halted
```

```
halt code = 1
```

```
operator initiated halt
```

```
PC = ffffffff857b2ac8
```

```
P00>>>cont
```

```
continuing CPU 0
```

```
$ show system/noprocess
```

```
OpenVMS V8.3  on node CLASS2  29-MAR-2007 16:24:00.90  Uptime  0 00:04:45
```



## System will not Reboot!

- Check the console and if the system displays a BUGCHECK message on the console and shuts itself down, it means the system encountered a problem that made further operation impossible or dangerous.
- If the system does not automatically reboot then make sure your system is set to boot automatically by checking the console parameter AUTO\_ACTION to ensure that it is set to RESTART.
  - The system attempts to write a crash dump to the dump file, and after the dump write completes, this makes the system try to reboot itself automatically.
  - SRM console command is:
    - PO>>> set auto\_action restart



# System is not responding!

- If the system stops responding to your commands (that is, the system “hangs”), there is a possible failure in a system software or hardware component
- If is the case then try to generate a crash dump and then reboot.
- **DO NOT JUST POWER CYCLE THE SYSTEM.** If you do you will never know why it “Hung”, and will probably do it again.



# System is not responding!

- How do I generate a Crash Dump
  - Run OPCCRASH from the console if possible.
  - If unable to run OPCCRASH, then HALT the system and CRASH.



# System is not responding!

Examples:

```
$ mcr opccrash
```

```
Quorum: 3 (of 5 votes); this node contributes 1 vote  
Cluster has no voting quorum disk.
```

```
**** Starting compressed selective memory dump at 29-MAR-2007 16:25...
```

```
.....  
** System space, key processes, and key global pages have been dumped.  
** Now dumping remaining processes and global pages...
```

```
.....  
...Complete ****
```

```
SYSTEM SHUTDOWN COMPLETE
```

```
halted CPU 0  
halt code = 5  
HALT instruction executed  
PC = ffffffff80087b24  
P00>>>
```



# System is not responding!

Examples:

```
$ CTL-P
```

```
halted CPU 0
```

```
. . .
```

```
P00>>>crash
```

```
CPU 0 restarting
```

```
**** OpenVMS Alpha Operating.
```

```
System V8.3      - BUGCHECK ****
```

```
** Bugcheck code = 0000064C: OPERCRASH, Operator forced system crash
```

```
** Crash CPU: 00000000      Primary CPU: 00000000      Node Name: CLASS2
```

```
** Supported CPU count:    00000004
```

```
** Active CPUs:           00000000.0000000F
```

```
** Current Process:       NULL
```

```
** Current PSB ID:        00000001
```

```
** Image Name:
```



# System is not responding!

Examples:

```
**** Starting compressed selective memory dump at 29-MAR-2007 16:16...  
.....  
** System space, key processes, and key global pages have been dumped.  
** Now dumping remaining processes and global pages...  
.....  
...Complete ****
```

halted CPU 0

halt code = 5

HALT instruction executed

PC = ffffffff80087b24

CPU 0 booting

(boot dkb300.3.0.5.1 -flags 0,0)

. . . .



# System crashed, now what?

- This is not a crash dump analysis session. But system manager can do some preliminary work.
  - Copy the crash dump
  - Gather up the error log file
  - Contact your support provider or PARSEC Group (which can become your support provider) to analyze crash dump and error log.





# System crashed, now what?

- Do not use BACKUP to move SYSDUMP.DMP
  - It is marked /NOBACKUP for starters
  - BACKUP only allocates and sets the file high-water mark to zero without copying any data
  - BUGCHECK writes the dump (it doesn't know about HWM) – so all looks well. But: SDA (& DUMP) gets zeroes back whenever the file is read – leads to the name “phantom dump”
  - %SDA-E-BADHWM error starting in V8.2
  - Can be fixed by most of the time by:
    - \$ SET VOLUME/NOHIGHWATER ddcn: ! If necessary
    - \$ SET FILE/END ddcn:[SYSn.SYSEX]SYSDUMP.DMP
    - \$ SET VOLUME/HIGHWATER ddcn: ! If necessary



# System crashed, now what?

- Do not use DCL COPY to save contents of a system dump (or BACKUP/IGNORE=NOBACKUP)
- Multiple reasons to use SDA COPY
  - BUGCHECK probably didn't use the entire file
    - SDA COPY only saves used blocks
  - Integrity system dumps need process unwind data
    - SDA COPY collects it and appends it to the copy



# System crashed, now what?

- Multiple reasons to use SDA COPY
  - File ID to filename translation data may be useful
    - SDA COPY collects it and appends it to the copy
  - SDA COPY will compress the dump if originally written as a raw dump
  - Only copies dump file if it is a valid dump
- Why not create a command procedure to do some work for you?
  - If saving to an alternate drive, mount device in SYCONFIG.COM.
  - Create logical pointing location of command procedure in SYLOGICALS.COM



# System crashed, now what?

## Examples

```
$ anal/crash sys$system:sysdump.dmp
```

OpenVMS system dump analyzer

...analyzing an Alpha compressed selective memory dump...

Dump taken on 29-MAR-2007 16:25:33.54 using version V8.3

OPERATOR, Operator requested system shutdown

```
SDA> copy storage:[crash_files]crash_mar29.dmp
```

```
%SDA-I-COLLECTING, collecting file and/or unwind data
```

```
SDA> exit
```

```
$
```



# System crashed, now what?

## Examples

```
$ type sys$manager:savedump.com
```

```
$!   Print dump listing if system just failed.
```

```
$!
```

```
$ analyze/crash_dump sys$system:sysdump.dmp
```

```
COPY STORAGE:[CRASH_FILES]SAVEDUMP.DMP    ! Save the dump file
```

```
SET OUTPUT SYSDUMP.LIS                      ! Create a listing file
```

```
CLUE CRASH
```

```
SHOW CRASH                                ! Display crash info
```

```
SHOW STACK                                ! Show current stack
```

```
SHOW SUMMARY/IMAGE                        ! List all active processes
```

```
SHOW PROCESS/PCB/PHD/REG                  ! Display current process
```

```
SHOW SYMBOL/ALL                           ! Print system symbol table
```

```
EXIT
```

```
$ show log/full clue$site_proc
```

```
"CLUE$SITE_PROC" [super] = "SYS$MANAGER:SAVEDUMP.COM" (LNM$SYSTEM_TABLE)
```



# Volume Management

- Free space monitoring
- Shadow set members
  - Basic Shadow Terminology
  - Why is there a shadow copy?
  - Why is there a shadow merge?
  - Copy and merge fence
- Analyzing Disks Volumes





# Free Space Monitoring

- In OpenVMS unlike Unix systems 0 blocks means 0 blocks
- If system disk gets to 0 blocks system will hang
- Database disk especially need monitoring
- Several ways to monitor disk space, but command procedure in a batch queue is the most reliable.
- By default, disk space is shown in blocks. It can be displayed in bytes by either:
  - `$ show dev/unit=bytes d`
  - `$ set process/unit=bytes`





# Free Space Monitoring

- The following shows a command procedure that will monitor disk space.
- It's a simple command procedure which you can modify for your specific needs.



# Free Space Monitoring

Examples:

Here is the calculation.

```
$! DISKSPACE.COM - PARSEC Group 03/29/07
```

```
. . .
```

```
$ GET_DISK_NAME:
```

```
$ READ INFO_FILE DISK_RECORD /END_OF_FILE=EOF_LABEL
```

```
$ DISKNAME = F$ELEMENT(0, ":", DISK_RECORD)
```

```
$ TOTAL_SPACE = F$GETDVI(DISKNAME, "MAXBLOCK")/10
```

```
$ FREE_SPACE = F$GETDVI(DISKNAME, "FREEBLOCKS")/10
```

```
$ TOTAL_USED = TOTAL_SPACE - FREE_SPACE
```

```
$ CAPACITY = (TOTAL_SPACE - FREE_SPACE) * 100 / TOTAL_SPACE
```

```
. . .
```



# Free Space Monitoring

Example Output:

| DEVICE NAME  | TOTAL SPACE | FREE SPACE | TOTAL USED | %CAPACITY USED |
|--------------|-------------|------------|------------|----------------|
| \$5\$DQA0    | 5863334     | 5289417    | 573917     | 9              |
| \$16\$DKA0   | 411048      | 276732     | 134316     | 32             |
| \$16\$DKA100 | 83252       | 55471      | 27781      | 33             |
| \$16\$DKA200 | 83252       | 82460      | 792        | 0              |
| \$16\$DKA300 | 411048      | 270023     | 141025     | 34             |
| \$18\$DRA0   | 832307      | 744431     | 87876      | 10             |
| \$18\$DRA1   | 832307      | 101645     | 730662     | 87             |
| \$22\$DKA300 | 3555638     | 938304     | 2617334    | 73             |
| \$22\$DKA506 | 4189014     | 488588     | 3700426    | 88             |
| \$22\$DKA507 | 2512793     | 1280462    | 1232331    | 49             |
| \$32\$DKA0   | 7113296     | 3233664    | 3879632    | 54             |
| \$32\$DKA100 | 7113296     | 5616547    | 1496749    | 21             |
| \$82\$DKA0   | 205086      | 3979       | 201107     | 98             |



# Basic Shadow Terminology

- HBVS Host Based Volume Shadowing
  - OpenVMS version of RAID1 implementation
- VU Virtual Unit
  - The volume that is mounted whose device name is DSA
- Shadow Set
  - The volume that is mounted using the letters DSA
- SSM Shadow Set Member
  - Maximum of three devices can form a shadow set



# Shadow States

- Shadow devices can be in one of two states:
  - ***Transient State***, when one or more of its members are undergoing a copy or a merge operation.
  - ***Steady State***, which means all of its members are known to contain identical data.



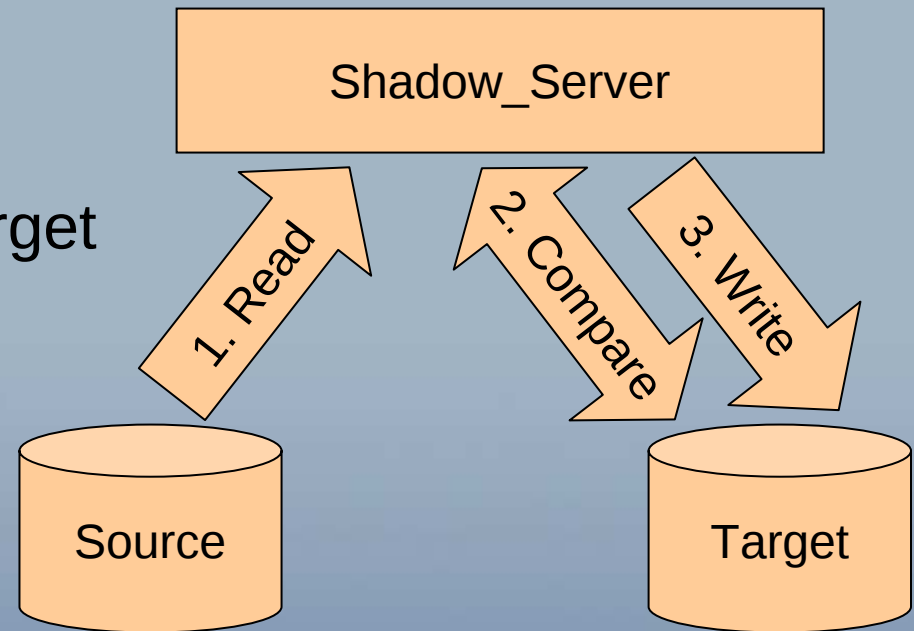
# Why is there a shadow copy?

- The DCL MOUNT command initiates a copy operation, when a disk is added to an existing shadow set.
- Copy operation duplicates data on a source disk to a target disk
- Starts at first Logical Block on disk (LBN zero) and processes 127 blocks at a time from beginning to end.
- Read and writes access continue while a disk(s) are undergoing a copy operation.



# Why is there a shadow copy?

1. Read from source
2. Compare with target
3. If different, write data to target and start over at Step 1.





# Creating a VU

```
$ init/system/shadow=(VDA17:,VDA16:,VDA15:) disk30  
$ mount/system dsa30 /shadow=($1$VDA17:,$1$VDA16:,$1$VDA15:) disk30
```

```
%MOUNT-I-MOUNTED, DISK30 mounted on _DSA30:
```

```
%MOUNT-I-SHDWMEMSUCC, _$1$VDA17: (CLASS3) is now a valid member of the shadow set
```

```
%MOUNT-I-SHDWMEMSUCC, _$1$VDA16: (CLASS3) is now a valid member of the shadow set
```

```
%MOUNT-I-SHDWMEMSUCC, _$1$VDA15: (CLASS3) is now a valid member of the shadow set
```



# Adding to a VU

```
$ show dev dsa20
```

| Device Name | Device Status            | Error Count | Volume Label       | Free Blocks | Trans Count | Mnt Cnt |
|-------------|--------------------------|-------------|--------------------|-------------|-------------|---------|
| DSA20:      | Mounted                  | 0           | DISK20             | 2849        | 1           | 1       |
| \$1\$VDA19: | (CLASS3) ShadowSetMember | 0           | (member of DSA20:) |             |             |         |
| \$1\$VDA20: | (CLASS3) ShadowSetMember | 0           | (member of DSA20:) |             |             |         |

```
$ mount/system/confirm dsa20 /shadow=$1$VDA18: disk20
```

```
%MOUNT-F-SHDWCOPYREQ, shadow copy required
```

```
Virtual Unit - _DSA20: Volume Label - DISK20
```

```
Member Volume Label Owner UIC
```

```
_ $1$VDA18: (CLASS3) DISK0 [1,1]
```

```
Allow FULL shadow copy on the above member(s)? [N]:y
```

```
%MOUNT-I-MOUNTED, DISK20 mounted on _DSA20:
```

```
%MOUNT-I-SHDWMEMCOPY, _ $1$VDA18: (CLASS3) added to the shadow set with a copy operation
```

```
%MOUNT-I-ISAMBR, _ $1$VDA19: (CLASS3) is a member of the shadow set
```

```
%MOUNT-I-ISAMBR, _ $1$VDA20: (CLASS3) is a member of the shadow set
```

```
$ show dev dsa20
```

| Device Name | Device Status            | Error Count | Volume Label       | Free Blocks | Trans Count | Mnt Cnt |
|-------------|--------------------------|-------------|--------------------|-------------|-------------|---------|
| DSA20:      | Mounted                  | 0           | DISK20             | 2849        | 1           | 1       |
| \$1\$VDA18: | (CLASS3) ShadowSetMember | 0           | (member of DSA20:) |             |             |         |
| \$1\$VDA19: | (CLASS3) ShadowSetMember | 0           | (member of DSA20:) |             |             |         |
| \$1\$VDA20: | (CLASS3) ShadowSetMember | 0           | (member of DSA20:) |             |             |         |

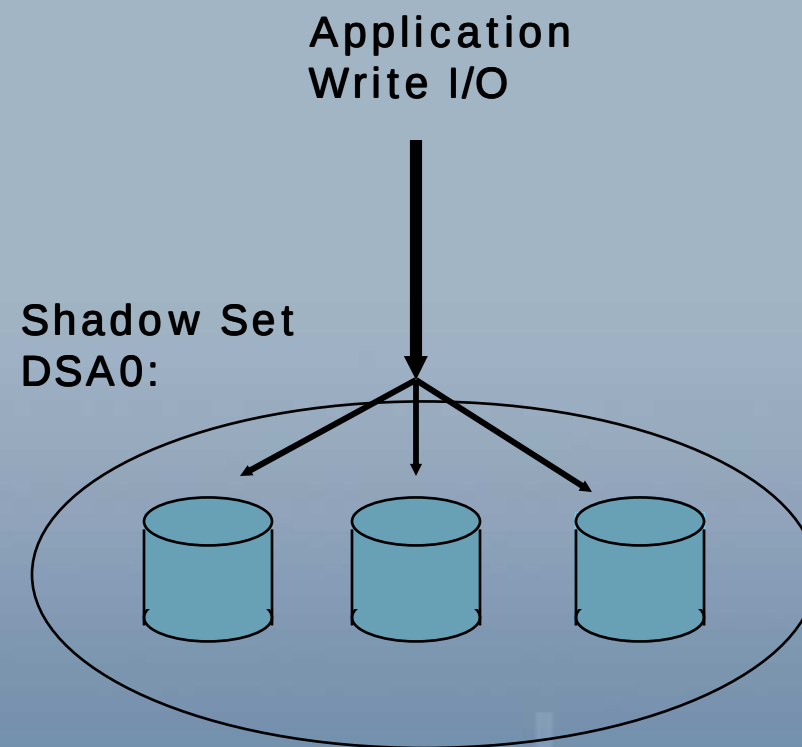


# Mini-Copy Scenario

- Mini-copy is a streamline copy operation
- A write bitmap tracks writes to a shadow set and is used to direct mini-copy operation
- Prior to the removal of a shadow set member, writes are sent directly to the shadow set
- To create the bitmap you specify /POLICY=MINICOPY when you DISMOUNT or MOUNT a shadow set member

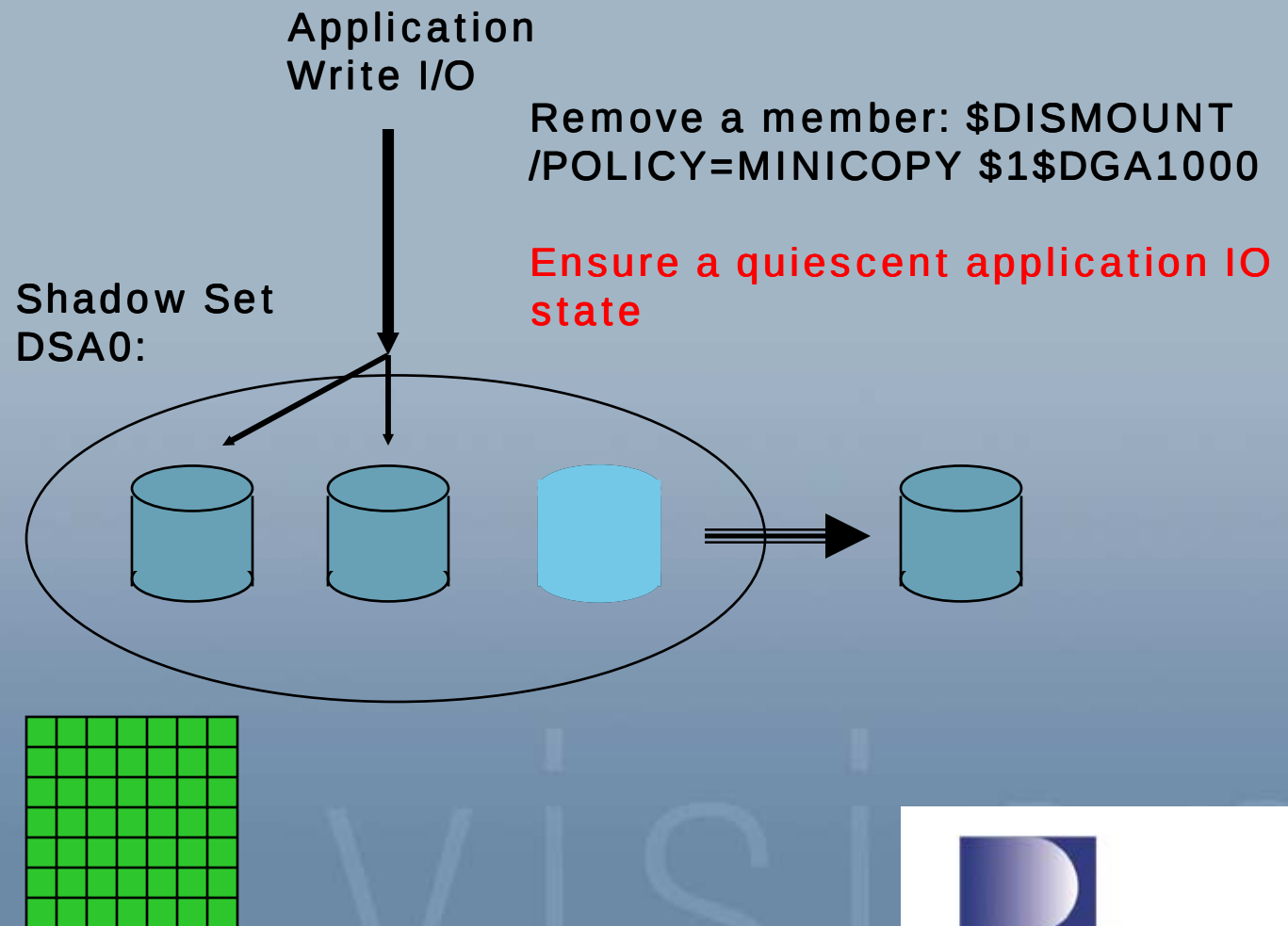


# Mini-Copy



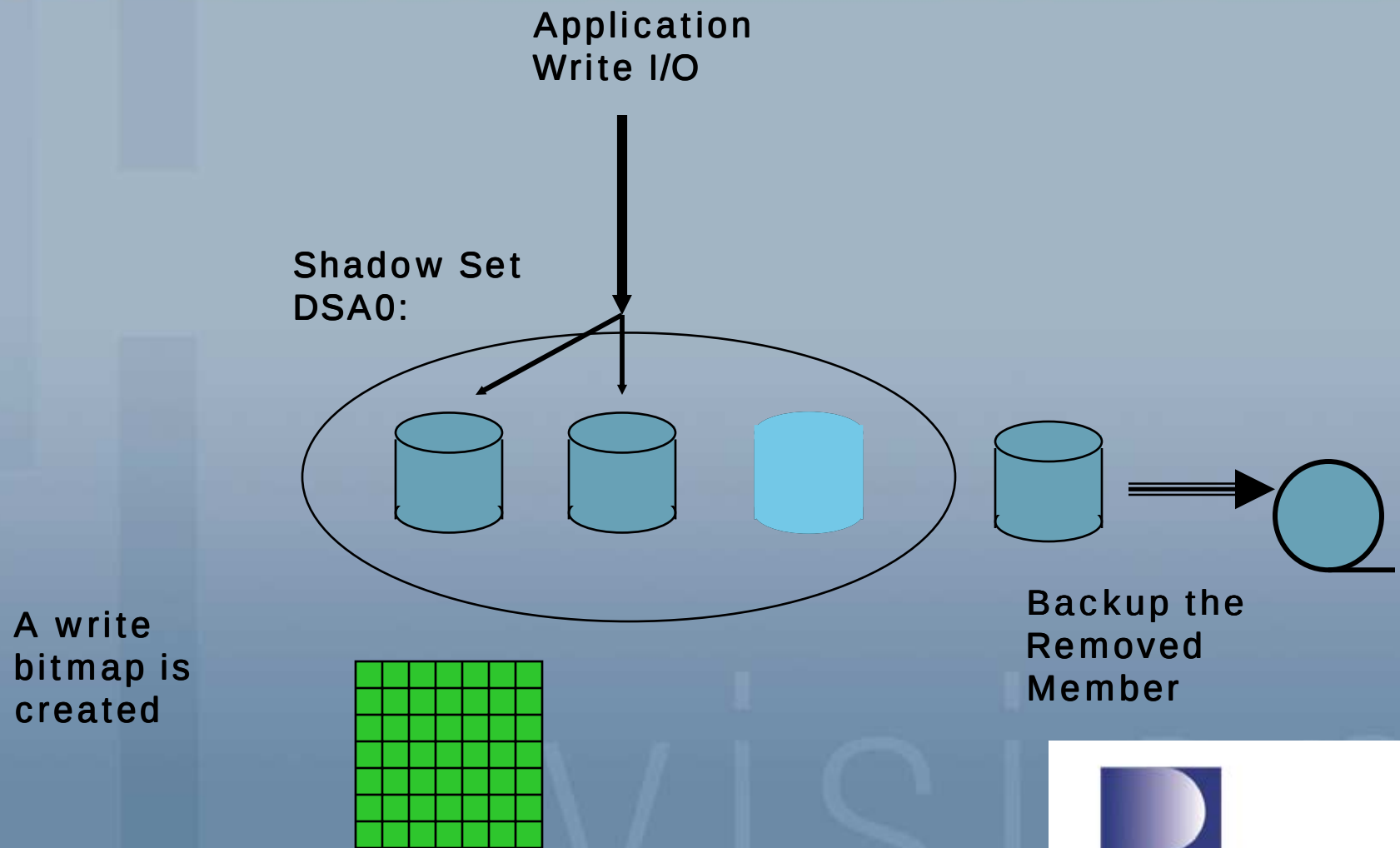


# Mini-Copy



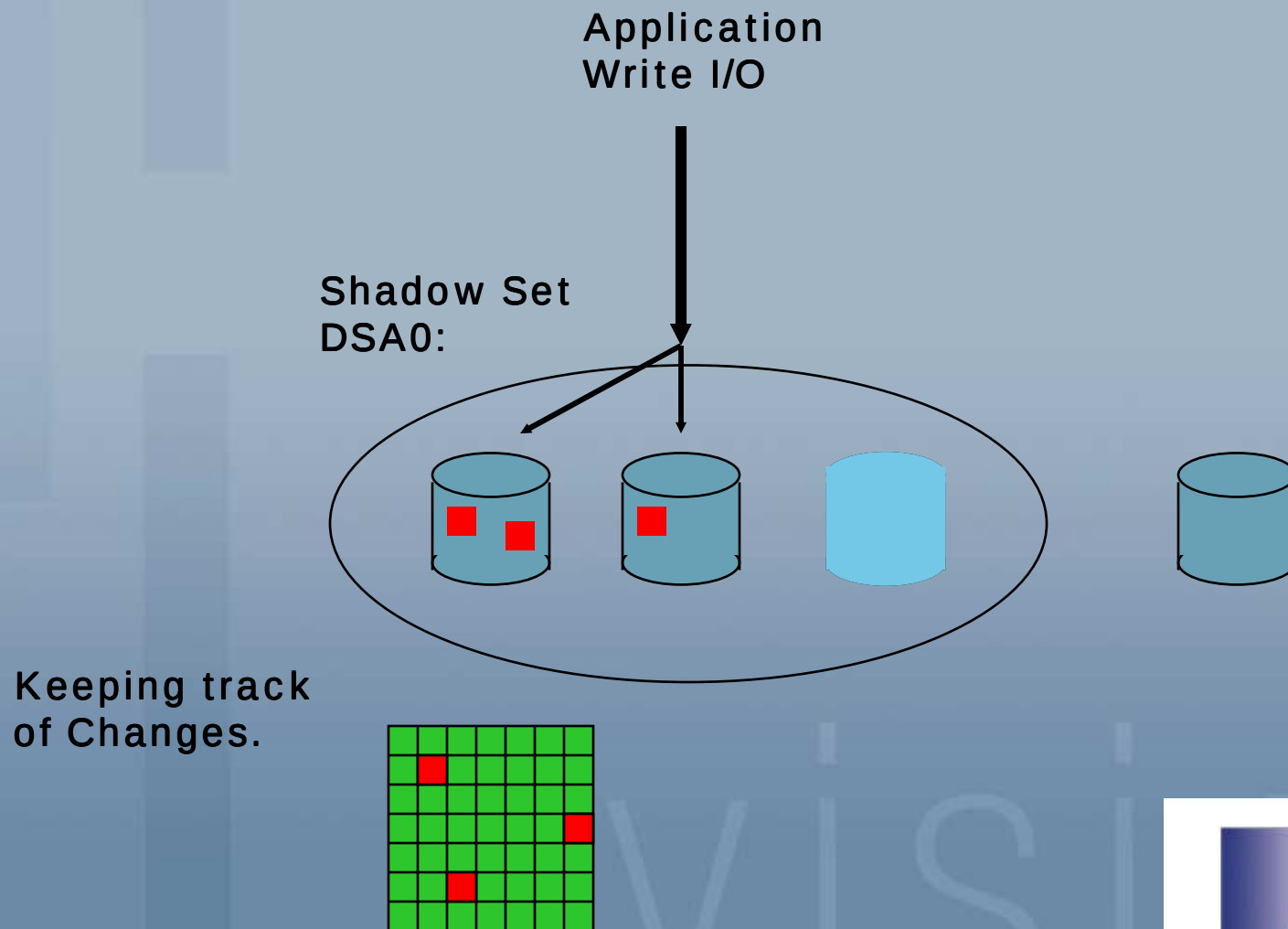


# Mini-Copy



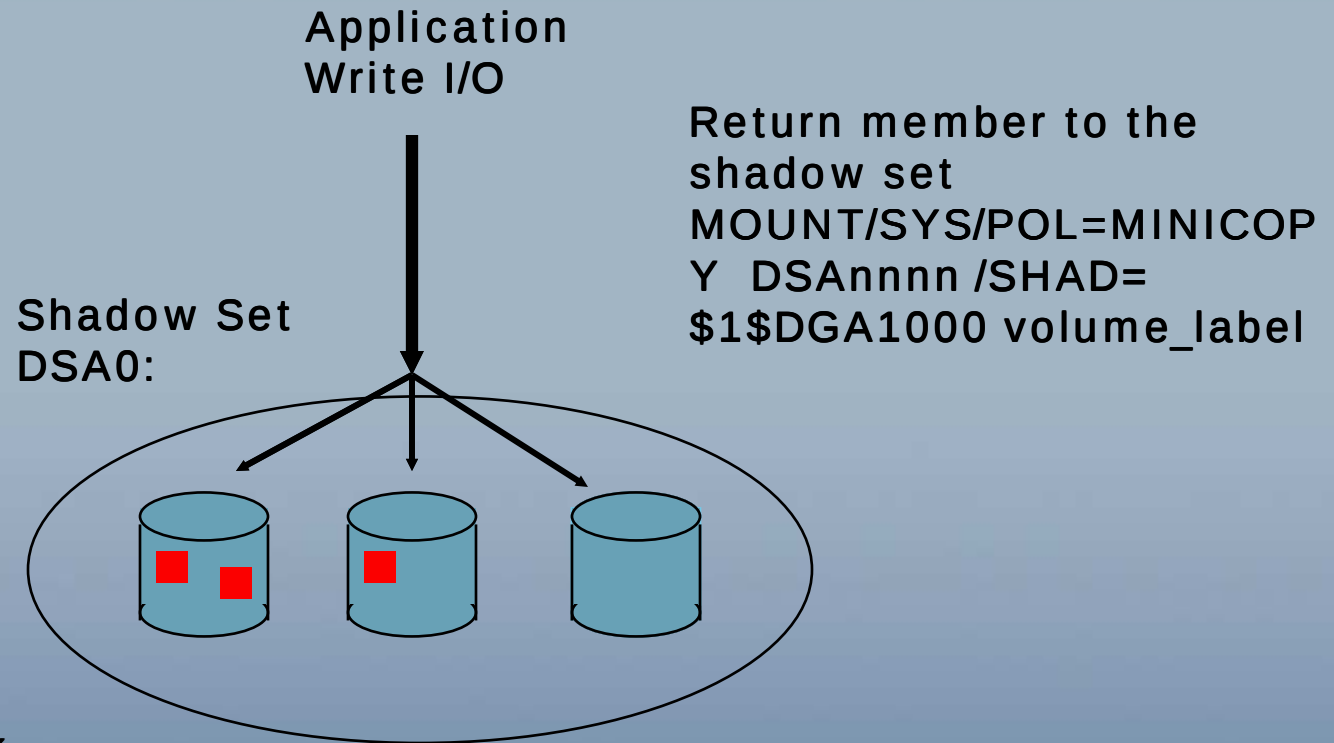


# Mini-Copy

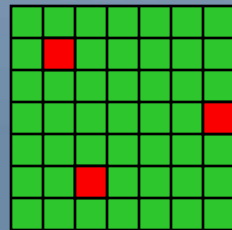




# Mini-Copy



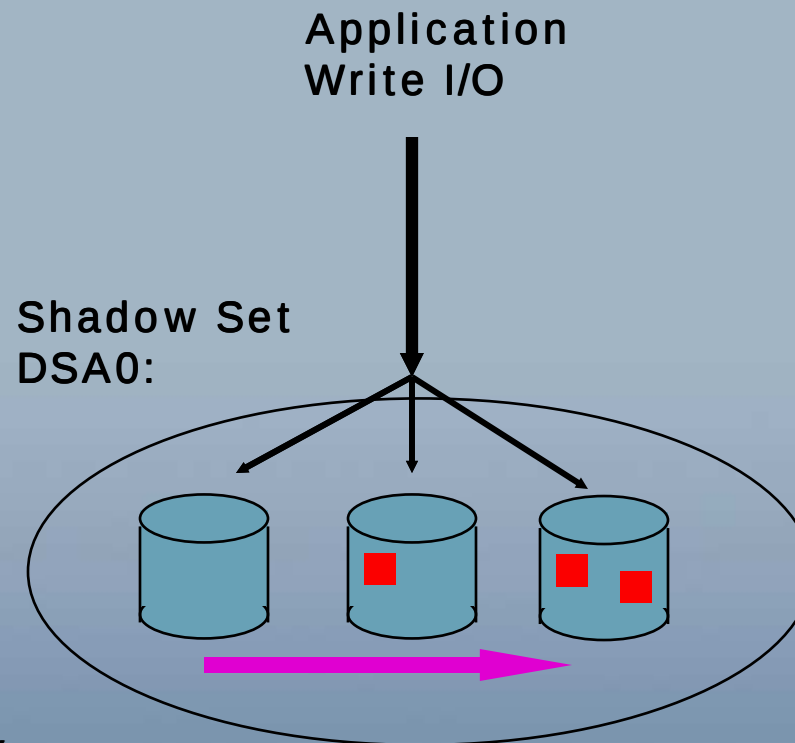
Keeping track  
of Changes.



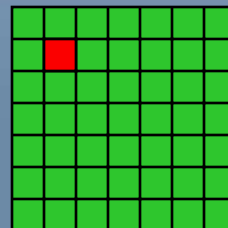


# Mini-Copy

Now, we only copy the data which has changed!

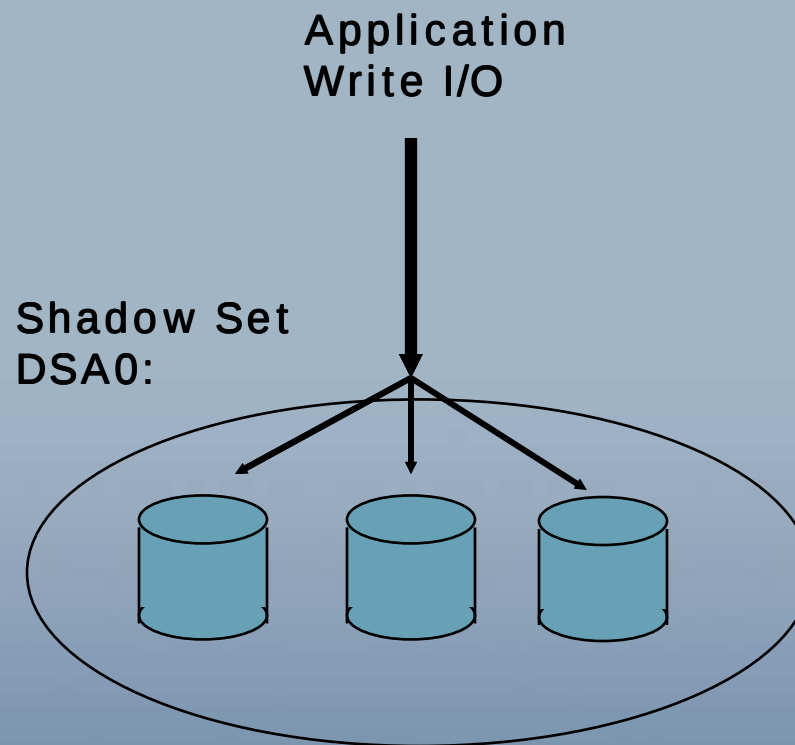


Keeping track of Changes.





# Mini-Copy



The bitmap is deleted for the added member

Shadow set is fully usable and consistent



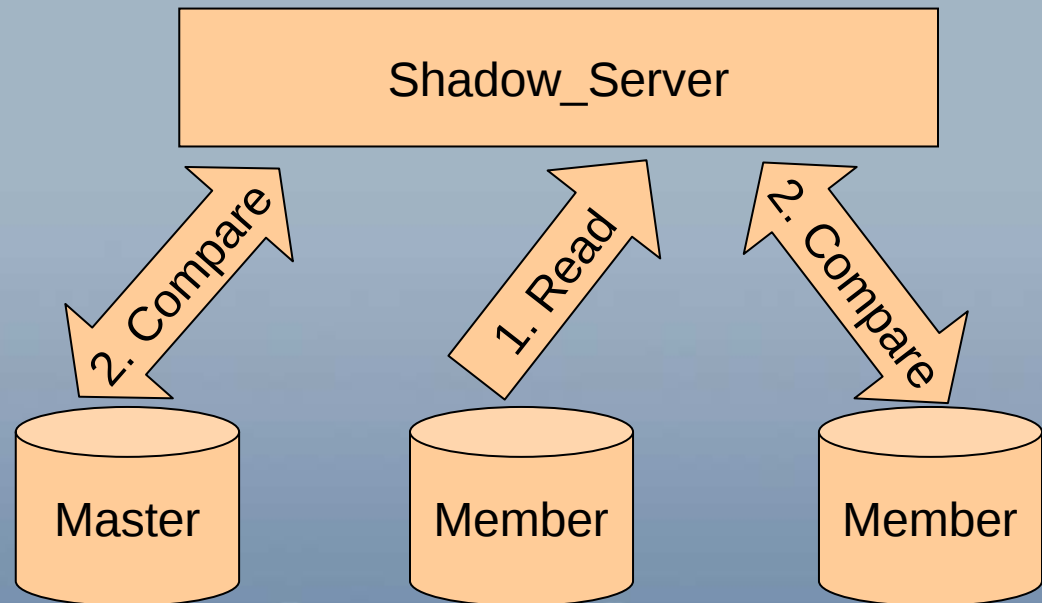
# Why is there a shadow merge?

- When a shadow set is “improperly” dismounted by a system a mandatory merge operation occurs.
  - An improper dismount (crash) can cause an application write I/O that is “in flight”, to write data to an indeterminate number of the shadow set members.
  - If a system aborts a shadow set and it has write I/O outstanding in its internal queues...a merge must be done
- Simply put, a merge operation insures that all devices contain identical data on ***all*** LBNS.



# Why is there a shadow merge?

1. Read from any member
2. Compare with other member(s)
3. If different, do a Fix-Up: halt all I/Os to the shadowset, fix up differences using data from the Master member, then allow I/Os to continue





# Host Based Mini-Merge

- Host based mini-merge is available for OpenVMS 7.3-2 with remedial patch
- Integrated in OpenVMS V8.2 for Alpha and Integrity
- Host based mini-merge depends on bitmaps and policies for information on mini-merge operations





# Host Based Mini-Merge

**\$ show shadow dsa20**

\_DSA20: Volume Label: IA64SYS

Virtual Unit State: Steady State

**Enhanced Shadowing Features in use:**

**Host-Based Minimerge (HBMM)**

|                     |      |               |         |
|---------------------|------|---------------|---------|
| VU Timeout Value    | 3600 | VU Site Value | 0       |
| Copy/Merge Priority | 5000 | Mini Merge    | Enabled |
| Served Path Delay   | 30   |               |         |

HBMM Policy

HBMM Reset Threshold: 50000

HBMM Master lists:

Any 1 of the nodes: PARSEC, BEAGLE

HBMM bitmaps are active on PARSEC

Modified blocks since bitmap creation: 254

Device \$32\$DKA0

|           |   |        |
|-----------|---|--------|
| Read Cost | 2 | Site 0 |
|-----------|---|--------|

|                |    |
|----------------|----|
| Member Timeout | 10 |
|----------------|----|

Device \$32\$DKA100

|           |     |                      |
|-----------|-----|----------------------|
| Read Cost | 501 | Master Member Site 0 |
|-----------|-----|----------------------|

|                |    |
|----------------|----|
| Member Timeout | 10 |
|----------------|----|

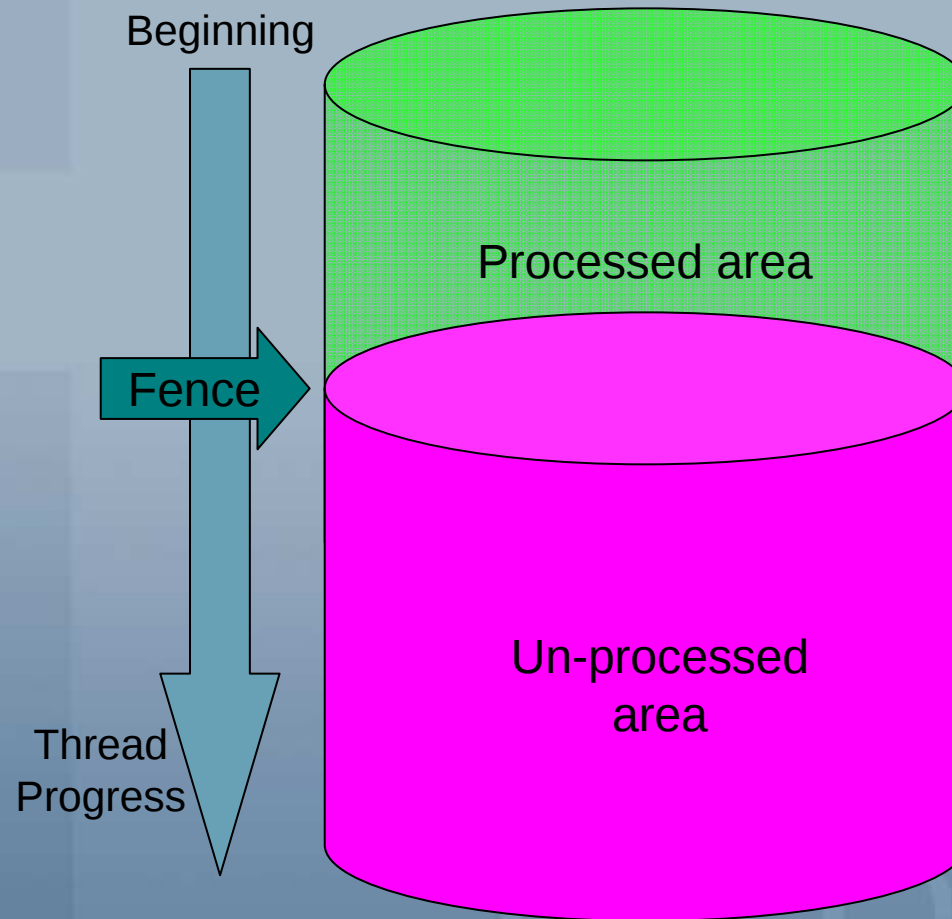


# Copy and Merge Fence

- For both Merge and Copy operation there is an imaginary fence.
  - It separates the unprocessed and processed portion of the volume
  - Is specified by completed LBN value
  - Is periodically distributed cluster wide
  - LBNs at or below the fence have been processed
  - LBNs above the fence have not been processed



# Why is there a shadow copy?





# Analyzing Disk Volumes

- The ANALYZE/DISK utility examines and repairs the OpenVMS file structure.
- It checks the readability and validity of the OpenVMS file structure.
- The utility write locks the volume when performing a repair
- Care should be taken when running this utility, because it could cause more damage than it fixes.





# Analyzing Disk Volumes

- The utility has three modes:
  - Command to report errors
    - \$ ANALYZE/DISK\_STRUCTURE device\_name
  - Command to report and repair errors
    - \$ ANALYZE/DISK\_STRUCTURE device\_name
  - Command to report errors and selectively repair errors
    - \$ ANALYZE/DISK\_STRUCTURE/REPAIR/CONFIRM device\_name



# Analyzing Utility Warning!

- Analyze Utility can cause more problems then it fixes!
  - If you are logging a lot of hardware errors or if you suspect severe corruption, DO NOT run Analyze Utility in repair mode.
  - Try and get a physical backup to a save\_set first.
  - Next run Analyze to get a report.
  - Then try and run Analyze in repair mode.
  - With the physical backup you can always get back to the original state of the disk.
- The next slide will show what happened when Analyze with repair was run and no back up of the device was completed.
  - They did loose data!
  - PARSEC did recover some of the data.



Analyze/Disk\_Structure for \_XMIT\$DKB3: started on 18-DEC-2006 14:55:06.85

%ANALDISK-I-OPENQUOTA, error opening QUOTA.SYS

-SYSTEM-W-NOSUCHFILE, no such file

%ANALDISK-W-ALLOCCLR, blocks incorrectly marked allocated

LBN 5563170 to 5563283, RVN 1

%ANALDISK-W-BADDIRENT, invalid file identification in directory entry

[XMIT\_DATA.20061218]0000005036-HQB-002.OSB;1

-ANALDISK-I-BAD\_DIRHEADER, no valid file header for directory

%ANALDISK-W-BADDIRENT, invalid file identification in directory entry

[XMIT\_DATA.20061218]0000005036-JNE-002.OSB;1

-ANALDISK-I-BAD\_DIRHEADER, no valid file header for directory

%ANALDISK-W-BADDIRENT, invalid file identification in directory entry

[XMIT\_DATA.20061218]0000005036-JPY-002.OSB;1

-ANALDISK-I-BAD\_DIRHEADER, no valid file header for directory

%ANALDISK-W-BADDIRENT, invalid file identification in directory entry

[XMIT\_DATA.20061218]SPLIT\_CBSSRV\_CLS\_XMT\_145503.LOG;1

-ANALDISK-I-BAD\_DIRHEADER, no valid file header for directory

%ANALDISK-W-FREESPADRIFT, free block count of 65559543 is incorrect (RVN 1);

the correct value is 65558805



# Security Audit

- One of the more important aspect of maintaining a healthy system is security auditing. We will look at basic components of security auditing and also how to generate reports and pinpoint any issues that may have arose.





# Security Audit

- Security Audit Basics
- Components Involved in Security Auditing
- Security Audit Reporting





# Security Audit Basics

- OpenVMS can report security events as either “security audits” or “security alarms” or both.
  - Security Audits is a log of security events that is stored in a binary file and may be reviewed later.
  - Security Alarms are text message describing a security event sent to security operators.
    - Multiple terminals can be enabled as a security operator terminal.
    - Security alarms are used to notify system managers of an event that has or is occurring so they can take action.
    - You must have both OPER and SECURITY privilege to enable a terminal for security alarm messages.



# Components Involved in Security Auditing

- Audit Server Process
  - The Audit Server process performs the following actions:
    - Logging security events to the cluster-wide security audit file
    - Formats security alarms for reporting to security operators and operator log file
    - Monitor system-wide resources needed to log security events
    - Prevent the loss of security information when resources are depleted
  - Stop and start the server with the following command:
    - `$ SET AUDIT/SERVER=EXIT`
    - `$ SYS$SYSTEM:STARTUP AUDIT_SERVER`



# Components Involved in Security Auditing

- Audit Server Data File
  - The file VMS\$AUDIT\_SERVER.DAT contains information about the location of the security audit log file.
  - Should be shared by all nodes in a cluster for a single security domain.
  - Default location is SYS\$COMMON:[SYSMGR].
  - Can be moved to another location by defining the logical VMS\$AUDIT\_SERVER



# Components Involved in Security Auditing

- Security Audit Log File
  - All security audit events are logged to the security audit log file which is a binary file.
  - View information in the file using the DCL command ANALYZE/AUDIT.
  - The filename is  
SYS\$MANAGER:SECURITY\_AUDIT.AUDIT\$JOURNAL.
  - Can be moved to another location by issue the DCL command \$ SET AUDIT/DESTINATION=destination.



# Security Reporting

- Keeping track of all the security information possible is useless if it is not possible to generate reports on the information.
- We will be looking at the DCL ANALZE/AUDIT command





# Security Reporting

- The ANALYZE/AUDIT command is used to review security audit events. It is capable of generating four types of outputs which are:
  - Summary Report
  - Brief Report
  - Full Report
  - Binary Output File





# Summary Report

- This report provides a count of the number of each type of events included. It does not have any details about those events.
- This report is strictly a glimpse of the events and is a sound starting point for system managers to look deeper into security issues on the system.
- It is recommended to run this daily.





# Summary Report

Example

**\$ analyze/audit/summary/since=1-feb-2007/before=1-mar-2007 common:**

|                            |            |                         |       |
|----------------------------|------------|-------------------------|-------|
| Total records read:        | 893021     | Records selected:       | 98306 |
| Record buffer size:        | 881        |                         |       |
| Successful logins:         | 255        | Object creates:         | 852   |
| Successful logouts:        | 38         | Object accesses:        | 20658 |
| Login failures:            | 339        | Object deaccesses:      | 9186  |
| <b>Breakin attempts:</b>   | <b>52</b>  | Object deletes:         | 946   |
| <b>System UAF changes:</b> | <b>102</b> | Volume (dis)mounts:     | 9     |
| Rights db changes:         | 32         | System time changes:    | 0     |
| Netproxy changes:          | 0          | Server messages:        | 0     |
| Audit changes:             | 26         | Connections:            | 0     |
| Installed db changes:      | 0          | Process control audits: | 526   |
| <b>Sysgen changes:</b>     | <b>1</b>   | Privilege audits:       | 2768  |
| NCP command lines:         | 16         | Persona audits:         |       |



# Brief and Full Reports

- After reviewing the summary report, it is decided that more information is needed about the “Audit changes” so a brief report is generated on those items.
- Next we are going to see who made the SYSGEN parameter change from the full report.





# Brief Report

Example:

```
$ analyze/audit/summary/since=1-feb-2007/before=1-mar-2007 -  
_$_ /event_type=audit common:
```

| Date / Time             | Type  | Subtype         | Node   | Username | ID       |
|-------------------------|-------|-----------------|--------|----------|----------|
| 13-FEB-2007 10:13:33.60 | AUDIT | ALARM_TERMINATE | CLASS2 | SAUER    | 20C00240 |
| 20-FEB-2007 10:13:33.68 | AUDIT | AUDIT_INITIATE  | CLASS3 | SYSTEM   | 23200105 |
| 20-FEB-2007 10:13:33.76 | AUDIT | ALARM_STATE     | CLASS2 | SPENCER  | 29C000CB |
| 20-FEB-2007 10:13:33.84 | AUDIT | AUDIT_INITIATE  | CLASS2 | WILLIAMS | 23400256 |
| 21-FEB-2007 10:13:33.92 | AUDIT | ALARM_INITIATE  | CLASS3 | SYSTEM   | 24200085 |
| 22-FEB-2007 10:13:33.99 | AUDIT | AUDIT_TERMINATE | CLASS3 | SAUER    | 22C03883 |
| 24-FEB-2007 10:13:34.07 | AUDIT | ALARM_STATE     | CLASS2 | SPENCER  | 29C000CB |
| 24-FEB-2007 10:13:34.14 | AUDIT | AUDIT_STATE     | CLASS2 | SPENCER  | 29C000CB |
| 24-FEB-2007 10:13:34.22 | AUDIT | ALARM_STATE     | CLASS2 | SPENCER  | 29C000CB |
| 26-FEB-2007 10:13:34.30 | AUDIT | AUDIT_STATE     | CLASS2 | PARSEC   | 28C0884B |
| 26-FEB-2007 10:13:34.38 | AUDIT | ALARM_STATE     | CLASS2 | PARSEC   | 28C0884B |
| 26-FEB-2007 10:13:34.45 | AUDIT | AUDIT_STATE     | CLASS2 | PARSEC   | 28C0884B |
| 26-FEB-2007 10:13:34.52 | AUDIT | ALARM_STATE     | CLASS2 | PARSEC   | 28C0884B |

.  
. .  
.

vision



PARSEC Group  
Our Trainers Consult. Our Consultants Train.



# Full Report

Example:

```
$ analyze/audit/full/since=1-feb-2007/before=1-mar-2007/event_type=sysgen common:
```

```
Security Audit Analysis Utility
```

```
-----
```

```
Security audit (SECURITY) on CLASS9, system id: 1040
```

```
Auditable event:          SYSGEN parameter set
```

```
Event time:              2-FEB-2007 15:07:20.56
```

```
PID:                     28C00123
```

```
Process name:            SPENCER
```

```
Username:                SPENCER
```

```
Process owner:           [STAFF, SPENCER]
```

```
Terminal name:           RTA1:
```

```
Image name:              $16$DKA0:[SYS0.SYSCOMMON.][SYSEXEXE]SYSGEN.EXE
```

```
Parameters write:        $16$DKA0:<SYS10.SYSEXEXE>VAXVMSSYS.PAR;1
```

```
Parameters inuse:        Default
```

```
Startup:                 New:          SYS$SYSTEM:STARTUP1.COM
```

```
Original: SYS$SYSTEM:STARTUP.COM
```

```
SCSSYSTEMID:             New:          5367
```

```
Original: 0
```

```
SCSNODE:                 New:          BATGRL
```

```
Original:
```



# Accounting

- The accounting report will have the system understand how the system is used and by whom.
- Originally designed for accountants to charge system resources usage back to the users of the system, which isn't being used much today.
- Accounting data can track how users are utilizing the system and this information can help system managers detect unusual situations.





# Accounting

- What is being tracked
- What to look for
- Accounting File
- Generating reports





# What is being Tracked

- Proces - any process termination
- Image - image execution
- Interactive - interactive job termination
- Login Failure – failed login attempts
- Subprocess – Subprocess termination
- Detached – detached job termination
- Batch – batch job termination
- Network – network job terminal
- Print – all print jobs
- Message – user messages





# What to look for

- Most important thing to look for in accounting information is anything unusual.
- The following is a guideline on what to look for, which doesn't necessarily indicate a problem, but only flags that should be considered.
- But in general, it is now most useful as an troubleshooting aide.





# What to look for

- Unknown usernames
  - Users who normally do not log on to this system indicates a possible intrusion of the system.
- Unusual usage patterns
  - Look for users who are using the system on a weekend that normally uses the system only during the week. Also, consider times of the day the users are using the system.
- Unusual system resource usage
  - Check for process that is using an unusually large or small amount of resources compared to normal. This is an indication of the process doing things out of the ordinary.



# What to look for

- Unexpected sources of login
  - If a process logs in from a network connection for a user that is normally working at their desk on an OpenVMS workstation may be an indication the user is trying to hide something.
- If using for troubleshooting, the last status field



# Accounting File

- The accounting file is named `SYS$MANAGER:ACCOUNTNG.DAT`
- Can be moved to another disk or directory with the logical name `ACCOUNTNG`.
- Protect this file from processes that do need access to it, because it does contain usernames.



# Accounting File

- Much like the Security Auditing, Accounting can generate reports. Use the ACCOUNTING command to generate these reports. ACCOUNTING is capable of generating four types of outputs which are:
  - Brief Accounting Report
  - Full Accounting Report
  - Binary version of selected or rejected records
  - Summary report on selected items



# Accounting File

Example:

**\$ accounting/since/brief**

| Date / Time          | Type    | Subtype     | Username | ID       | Source | Status   |
|----------------------|---------|-------------|----------|----------|--------|----------|
| 30-MAR-2007 08:13:41 | PROCESS | SUBPROCESS  | SPENCER  | 2E4005DF |        | 00010001 |
| 30-MAR-2007 08:13:41 | PROCESS | SUBPROCESS  | SPENCER  | 2E4005DE |        | 00002BD4 |
| 30-MAR-2007 08:24:34 | PROCESS | SUBPROCESS  | SPENCER  | 2E40061A |        | 00010001 |
| 30-MAR-2007 08:24:34 | PROCESS | SUBPROCESS  | SPENCER  | 2E400619 |        | 00002BD4 |
| 30-MAR-2007 13:49:46 | LOGFAIL |             | <login>  | 2E400803 | TNA6:  | 00D38064 |
| 30-MAR-2007 13:52:07 | PROCESS | INTERACTIVE | MEHLHOP  | 2E400804 | TNA7:  | 10000001 |
| 30-MAR-2007 14:15:11 | PROCESS | SUBPROCESS  | SPENCER  | 2E400828 |        | 10000001 |
| 30-MAR-2007 14:15:11 | PROCESS | SUBPROCESS  | SPENCER  | 2E400829 |        | 00010001 |





# Accounting File

- Lets look at more information about the process with the ID of 2E400803, so lets generate a FULL report.





# Accounting File

Example:

**\$ accounting/full/id=2E400803**

LOGIN FAILURE

-----

|                                                                   |                 |                    |                         |
|-------------------------------------------------------------------|-----------------|--------------------|-------------------------|
| Username:                                                         | <login>         | UIC:               | [SYSTEST,SYSTEM]        |
| Account:                                                          | <login>         | Finish time:       | 30-MAR-2007 13:49:46.74 |
| Process ID:                                                       | 2E400803        | Start time:        | 30-MAR-2007 13:49:07.10 |
| Owner ID:                                                         |                 | Elapsed time:      | 0 00:00:39.64           |
| Terminal name:                                                    | TNA6:           | Processor time:    | 0 00:00:00.04           |
| Remote node addr:                                                 |                 | Priority:          | 4                       |
| Remote node name:                                                 |                 | Privilege <31-00>: | 0010C000                |
| Remote ID:                                                        | TELNET_0A64009F | Privilege <63-32>: | 00000000                |
| Remote full name:                                                 | 10.100.0.159    |                    |                         |
| Posix UID:                                                        | -2              | Posix GID:         | -2 (%FFFFFFFFE)         |
| Queue entry:                                                      |                 | Final status code: | 00D38064                |
| Queue name:                                                       |                 |                    |                         |
| Job name:                                                         |                 |                    |                         |
| Final status text: %LOGIN-F-CMDINPUT, error reading command input |                 |                    |                         |
| Page faults:                                                      | 90              | Direct IO:         | 16                      |
| Page fault reads:                                                 | 3               | Buffered IO:       | 24                      |
| Peak working set:                                                 | 1616            | Volumes mounted:   | 0                       |
| Peak page file:                                                   | 169216          | Images executed:   | 1                       |



# Process Management

- At times it is necessary to look at, or change, a currently executing process.
- System managers with WORLD privilege may look at the process with the SHOW PROCESS command.
- The /ID qualifier specifies the process ID of the desired process.





# Process Management

**parsec> show process/all/id=23E00E6B**

29-MAR-2007 09:43:54.19      User: PARSEC                      Process ID:      23E00E6B  
                                 Node: CLASS3                      Process name: "PARSEC"

Terminal:                      TNA23: (Host: tpg.parsec.com Port: 1141)

User Identifier:              [PARSEC]

Base priority:                7

Default file spec:      STAFF:[PARSEC]

Number of Kthreads: 1

Devices allocated:      CLASS3\$TNA23:

Process Quotas:

Account name:

CPU limit:                      Infinite      Direct I/O limit:                      150

Buffered I/O byte count quota:      99808      Buffered I/O limit:                      150

Timer queue entry quota:                      10      Open file quota:                      150

Paging file quota:                      43712      Subprocess quota:                      10

Default page fault cluster:                      64      AST quota:                      248

Enqueue quota:                      2000      Shared file limit:                      0

Max detached processes:                      0      Max active jobs:                      0





# Process Management

## Accounting information:

|                     |               |                        |        |
|---------------------|---------------|------------------------|--------|
| Buffered I/O count: | 159           | Peak working set size: | 4096   |
| Direct I/O count:   | 46            | Peak virtual size:     | 171616 |
| Page faults:        | 782           | Mounted volumes:       | 0      |
| Images activated:   | 8             |                        |        |
| Elapsed CPU time:   | 0 00:00:00.29 |                        |        |
| Connect time:       | 0 00:03:57.55 |                        |        |

## Authorized privileges:

NETMBX          SETPRV          TMPMBX

## Process privileges:

|        |                              |
|--------|------------------------------|
| ALTPRI | may set any priority value   |
| NETMBX | may create network device    |
| TMPMBX | may create temporary mailbox |

## Process rights:

|             |          |
|-------------|----------|
| PARSEC      | resource |
| INTERACTIVE |          |
| REMOTE      |          |

## System rights:

SYSS\$NODE\_CLASS3





# Process Management

Auto-unshelve: on  
Image Dump: off  
Soft CPU Affinity: off  
Parse Style: Traditional  
Case Lookup: Blind  
Units: Blocks  
Home RAD: 0  
Scheduling class name: none  
Process Dynamic Memory Area

|                        |        |                           |       |
|------------------------|--------|---------------------------|-------|
| Current Size (KB)      | 128.00 | Current Size (Pagelets)   | 256   |
| Free Space (KB)        | 114.73 | Space in Use (KB)         | 13.26 |
| Largest Var Block (KB) | 114.21 | Smallest Var Block (By)   | 96.00 |
| Number of Free Blocks  | 5      | Free Blocks LEQU 64 bytes | 0     |

There is 1 process in this job:  
PARSEC (\*)





# Process Management

- The SET PROCESS command may be used to change the attributes of the executing process.
- Additionally, use the /ID qualifier to specify the process ID of the desired process.
- The STOP command may be used to abort a process.





# Process Management

```
$ show system/process=parsec
```

```
OpenVMS V8.3  on node CLASS3  29-MAR-2007 09:54:59.05  Uptime  6 20:39:55
  Pid      Process Name      State  Pri      I/O      CPU      Page flts  Pages
23E00E6B PARSEC              COM    7      435    0 00:04:35.56      1231    125
```

```
$ set process/suspend/id=23E00E6B
```

```
$ show system/process=parsec
```

```
OpenVMS V8.3  on node CLASS3  29-MAR-2007 09:55:24.92  Uptime  6 20:40:21
  Pid      Process Name      State  Pri      I/O      CPU      Page flts  Pages
23E00E6B PARSEC              SUSP   7      435    0 00:04:55.51      1231    125
```

```
$ set process/resume/id=23E00E6B
```

```
$ show system/process=parsec
```

```
OpenVMS V8.3  on node CLASS3  29-MAR-2007 09:55:46.11  Uptime  6 20:40:42
  Pid      Process Name      State  Pri      I/O      CPU      Page flts  Pages
23E00E6B PARSEC              COM    7      435    0 00:05:10.21      1231    125
```

```
$ stop/id=23E00E6B
```

or

```
$ stop parsec
```





# Simple Performance Monitoring

- The Monitor utility is part of OpenVMS and can display system statistics on an ongoing basis.
- It does not have the ability to show trends or graph historical data
- It is best used to look at a live system for performance problems
- Can create binary recording files, which can
  - Be played back, possibly at a different interval
  - Converted to CSV files by an hp supplied utility for analysis using T4 (discussed later)



# Simple Performance Monitoring

- The Monitor utility has the following command syntax:  
MONITOR [/command qualifier[,...]] classname[,...] [/classname-qualifier[,...]]
- The following are useful Monitor qualifiers
  - /BEGINNING – Start time
  - /ENDING – End time
  - /BY\_NODE – Displays performance data by node
  - /[NO]DISPLAY – Specify /nodisplay when in batch mode
  - /INPUT – Input recording file
  - /INTERVAL – Sampling interval
  - /RECORD – Create an output binary recording file
  - /SUMMARY – Summarizes monitor data



# Simple Performance Monitoring

- The following classes can be specified via the Monitor utility:

- |               |                   |             |
|---------------|-------------------|-------------|
| ■ ALL_CLASSES | FILE_SYSTEM_CACHE |             |
| ■ DISK        | DLOCK             | FCP         |
| ■ CLUSTER     | IO                | DECNET      |
| ■ LOCK        | MODES             | MSCP_SERVER |
| ■ PAGE        | STATES            | RLOCK       |
| ■ RMS         | SCS               | PROCESSES   |
| ■ SYSTEM      | TIMER             | TRANSACTION |
| ■ VBS         | VECTOR            |             |



# Simple Performance Monitoring

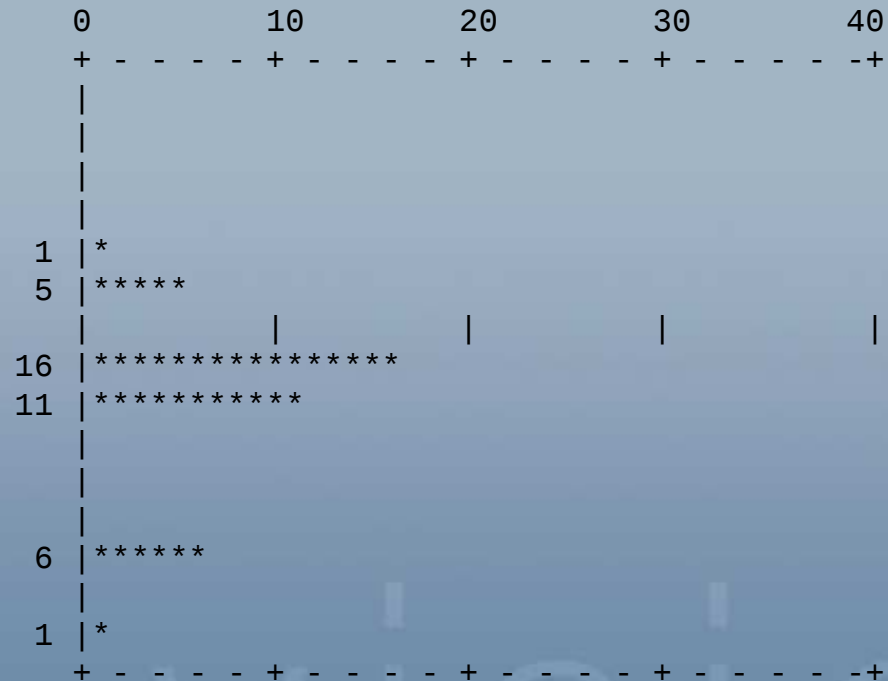
OpenVMS Monitor Utility

```
+-----+  
| CUR |  
+-----+
```

PROCESS STATES  
on node CLASS2  
29-MAR-2007 10:24:21.86

Collided Page Wait  
Mutex & Misc Resource Wait  
Common Event Flag Wait  
Page Fault Wait  
Local Event Flag Wait  
Local Evt Flg (Outswapped)

Hibernate  
Hibernate (Outswapped)  
Suspended  
Suspended (Outswapped)  
Free Page Wait  
Compute  
Compute (Outswapped)  
Current Process



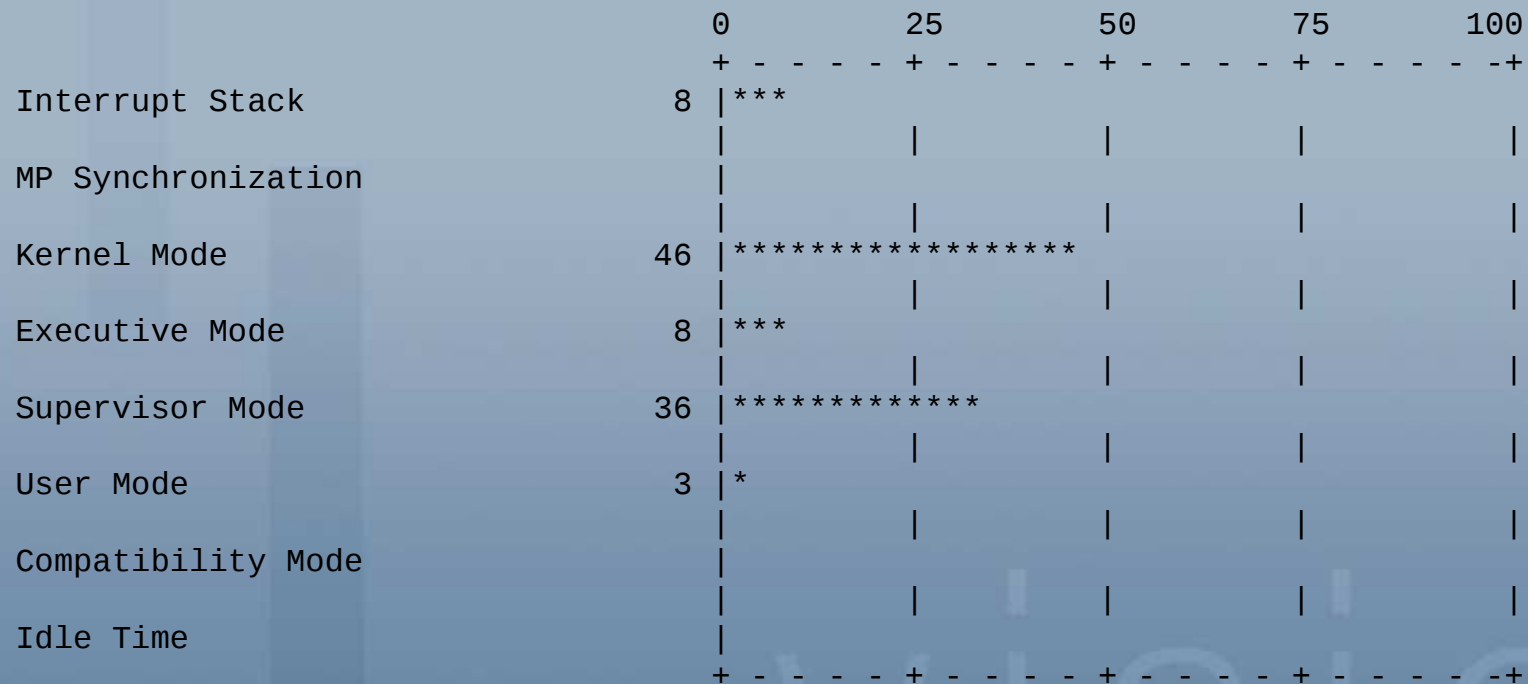


# Simple Performance Monitoring

OpenVMS Monitor Utility

```
+-----+  
| CUR |  
+-----+
```

TIME IN PROCESSOR MODES  
on node PARSEC  
29-MAR-2007 10:30:48.06





# Simple Performance Monitoring

OpenVMS Monitor Utility

TOP CPU TIME PROCESSES  
on node PARSEC  
29-MAR-2007 10:37:05.92

|          |            | 0  | 25        | 50 | 75        | 100 |           |   |           |   |
|----------|------------|----|-----------|----|-----------|-----|-----------|---|-----------|---|
|          |            | +  | - - - - - | +  | - - - - - | +   | - - - - - | + | - - - - - | + |
| 2020016A | BATCH_511  | 46 | *****     |    |           |     |           |   |           |   |
|          |            |    |           |    |           |     |           |   |           |   |
| 20200168 | SAUER_1    | 40 | *****     |    |           |     |           |   |           |   |
|          |            |    |           |    |           |     |           |   |           |   |
| 2020005F | TNT_SERVER | 3  | *         |    |           |     |           |   |           |   |
|          |            |    |           |    |           |     |           |   |           |   |
| 2020012D | SAUER      | 2  |           |    |           |     |           |   |           |   |
|          |            |    |           |    |           |     |           |   |           |   |
| 20200041 | SWAPPER    | 1  |           |    |           |     |           |   |           |   |
|          |            |    |           |    |           |     |           |   |           |   |
| 2020016F | FAULTER23  | 1  |           |    |           |     |           |   |           |   |
|          |            |    |           |    |           |     |           |   |           |   |
| 20200052 | TP_SERVER  | 1  |           |    |           |     |           |   |           |   |
|          |            |    |           |    |           |     |           |   |           |   |
|          |            | +  | - - - - - | +  | - - - - - | +   | - - - - - | + | - - - - - | + |

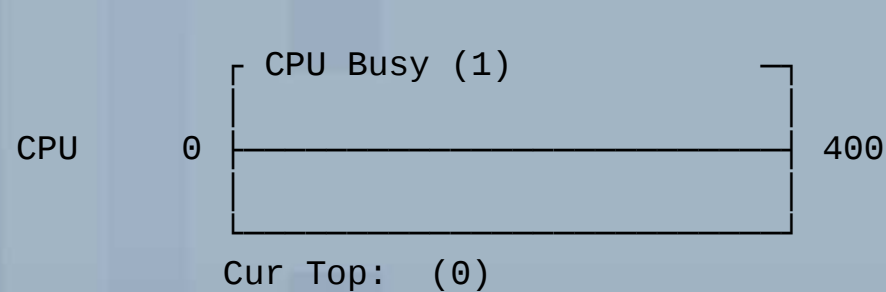


# Simple Performance Monitoring

Node: CLASS2  
Statistic: CURRENT

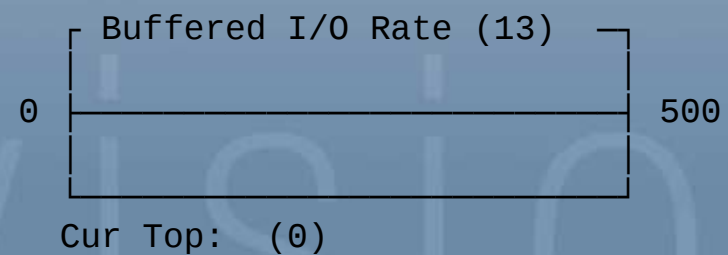
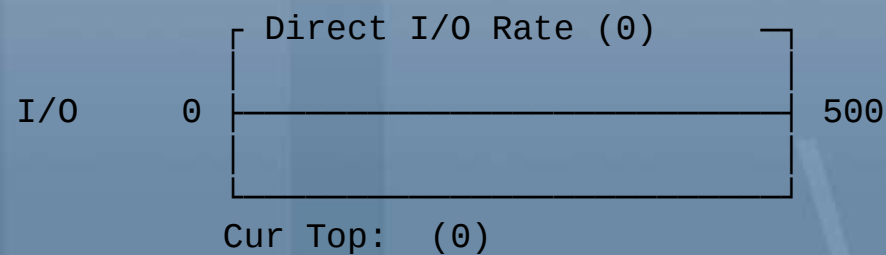
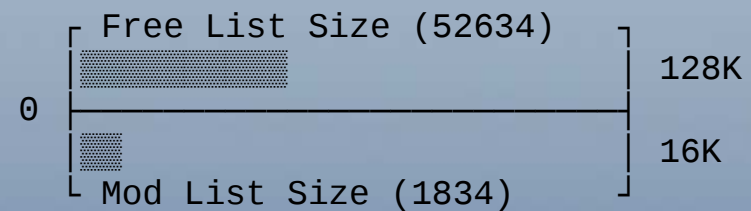
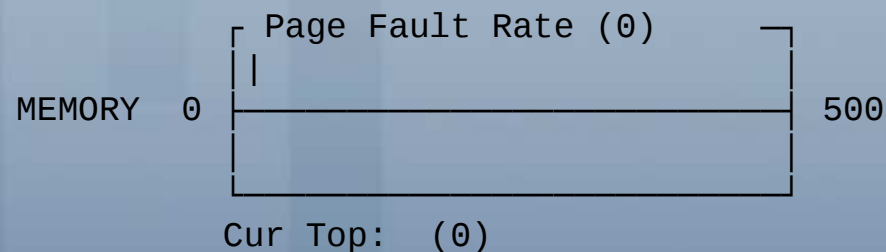
OpenVMS Monitor Utility  
SYSTEM STATISTICS

3-APR-2007 09:01:59



Process States

|           |    |        |   |
|-----------|----|--------|---|
| LEF:      | 11 | LEF0:  | 0 |
| HIB:      | 25 | HIB0:  | 0 |
| COM:      | 0  | COM0:  | 0 |
| PFW:      | 0  | CUR:   | 1 |
| MWAIT:    | 0  | Other: | 0 |
| Total: 37 |    |        |   |





# T4

- T4 – Tabular Timeline Tracking Tool
- Runs on OpenVMS
- Automatically creates historical archive
- Draws from multiple data sources
- Multiple performance metrics per source
- Merges to a **synchronized timeline** view
- Creates two-dimensional table (CSV)
  - CSV files can be imported to excel and other programs to create performance graphs



# Acquiring T4

- You can download the T4V4 tool kit from <http://h71000.www7.hp.com/OpenVMS/products/t4/index.html>
- At this site, you can find the T4 kit, as well as the readme file, which is VERY beneficial
- T4V33 tool kit ships with the release of OpenVMS V7.3-2 in SYS\$ETC:
- T4V34 tool kit ships with the release of OpenVMS V8.2 in SYS\$ETC:
- T4 collection can be a useful **adjunct** to your existing performance management program.



# TLVIZ

- TLViz (Time Line Vizualizer) is an HP internal tool, developed and used by OpenVMS Engineering to simplify and dramatically speed up the analysis of T4 style CSV files.
- TLViz is a Windows NT PC utility (written in Visual Basic) that allows you to quickly generate performance graphs using T4 generated CSV files
- Download the latest version from <http://h71000.www7.hp.com/OpenVMS/products/t4/index.html>
- The following example illustrates the use of TLViz

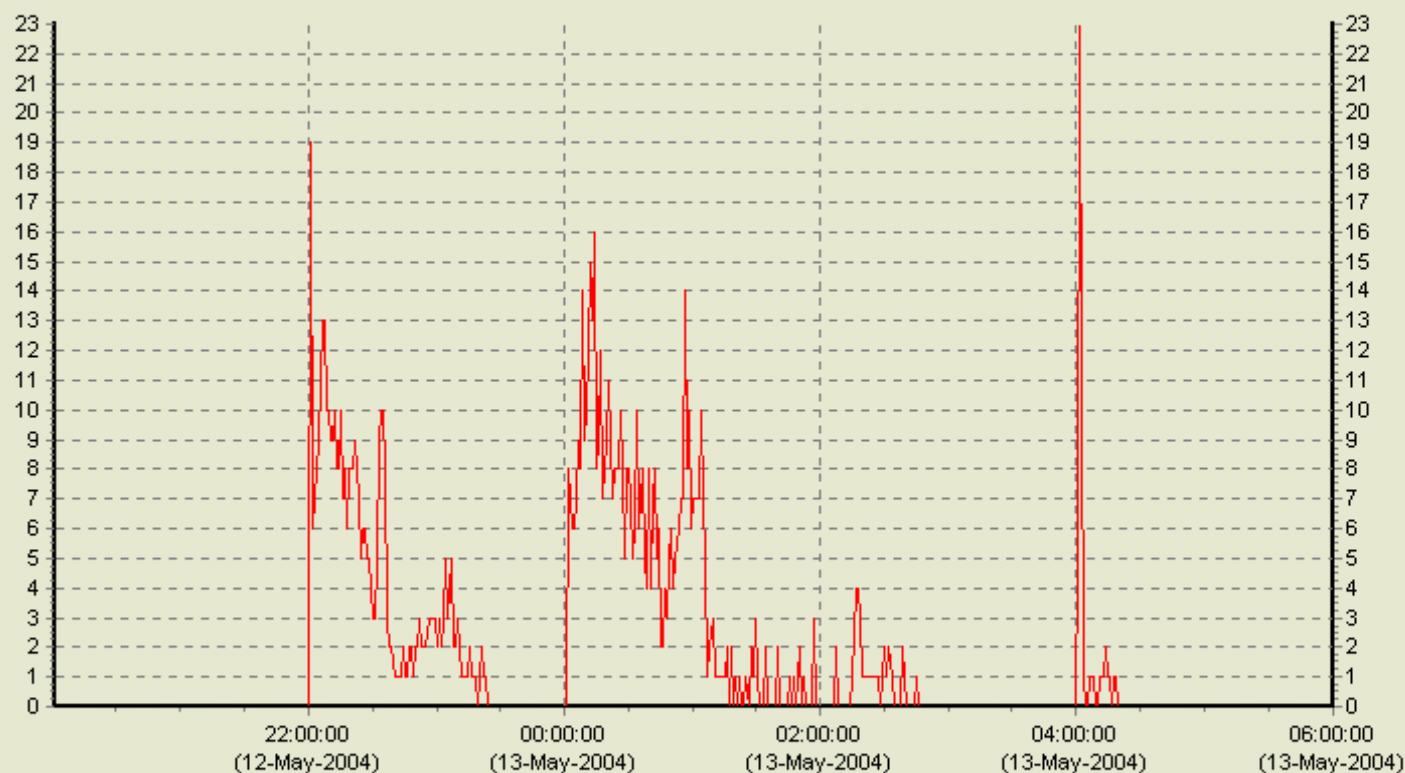


# T4 TimeLine Visualization (TLVIZ V1.4-30)

File Options Help

[ACLG]LifeTime < 1 min  
[ACLG]LifeTime < 2 mins  
[ACLG]LifeTime < 5 mins  
[ACLG]Logins  
[ACLG]Logouts  
[LCK73]Busy %  
[LCK73]Req Count/Sec  
[LCK73]Req Time Acc/Sec  
[LCK73]Spin Time Acc/Sec  
[MON.DISK]OpRate  
[MON.DISK]QLen  
[MON.DISK]RespT  
[MON.DLOC]Deadlock messages  
[MON.DLOC]DIR functions (Incom  
[MON.DLOC]DIR functions (Outg  
[MON.DLOC]Incoming BLKASTs  
[MON.DLOC]Local BLKASTs  
[MON.DLOC]Lock conversions (I  
[MON.DLOC]Lock conversions (L  
[MON.DLOC]Lock conversions (C  
[MON.DLOC]Lock DEQs (incomir  
[MON.DLOC]Lock DEQs (local)  
[MON.DLOC]Lock DEQs (outgoir  
[MON.DLOC]New Locks (Incomir  
[MON.DLOC]New Locks (Local)  
[MON.DLOC]New Locks (Outgoir  
[MON.DLOC]Outgoing BLKASTs  
[MON.FCP]FCP calls  
[MON.FCP]FCP Cpu  
[MON.FCP]FCP Disk space alloc.  
[MON.FCP]FCP erases  
[MON.FCP]FCP faults  
[MON.FCP]FCP Filename lookups  
[MON.FCP]FCP Files created  
[MON.FCP]FCP Files opened

Node(s) : PARSEC



Stack

Correlate

UnStack

Undo  
Zoom/Scroll

C:\WINDOWS\Desktop\parsec.csv



vision



# Questions??



vision



PARSEC Group  
Our Trainers Consult. Our Consultants Train.

# Question & Answer



vision

*Presented by  
Lori Spencer*

[www.parsec.com](http://www.parsec.com) | 888-4-PARSEC  
[lspencer@parsec.com](mailto:lspencer@parsec.com)



PARSEC Group  
Our Trainers Consult. Our Consultants Train.



vision

**To Download this Presentation, please visit:**  
**<http://www.parsec.com/public/HealthyManager.pdf>**

**To E-mail Lori**

**[lspencer@parsec.com](mailto: lspencer@parsec.com)**  
**www.parsec.com | 888-4-PARSEC**



**PARSEC Group**  
**Our Trainers Consult. Our Consultants Train.**