HP Technology Forum 2006

Strategies for Migration from Alpha and VAX to HP Integrity

Robert Gezelter, CDP, CSA, CSE
Principal,
Robert Gezelter Software Consultant

GET CONNECTED
People. Training. Technology.

“Failure is not an option.”

Gene Krantz, Lead Flight Director (ret)

NASA

End Users and not ISVs

Differences between ISV and End-User

- ISVs rebuild as a matter of routine for releases
- ISVs have a more controlled source base
- Release Delay
  - To an ISV – an annoyance
  - To an End User – business interruption
- Most organizations are somewhere between the two extremes

Business Issues

- Maintenance of operations
- New business needs
- Technology upgrades/refresh
- Upgrades/Refresh – low priority until interference with commitments

Persuading Non Technical Management

Persuading Business (non-Technical) Management

- Non-disruption of commitments
- Operations
- Business Expansion
- Long term costs
- Short term costs

OpenVMS has been here twice

The OpenVMS Philosophy – *Leverage through Architecture*

- Scaffolding minimization
- Flexibility by implication
- Small changes can be cascaded by the architecture
- A form, if you will, of an architectural calculus

invent

Planning for Cutover

Cutover Issues

- Worse than a non-cutover is a failed cutover
- “As Planned” will take care of itself
- Plan for when things do not go as planned
- Pilots, divers, mountain climbers, and others
- Timing –
  - “Point of Safe Return”
  - Iron Clad Mission Rules

OpenVMS Technologies and Implications

OpenVMS Tools for Seamless Transitions from Alpha/VAX to Integrity

- Image Translation (VEST, AEST)
- Mixed architecture OpenVMS clusters
- Compatible compilers
- Logical Names
- Host Based Volume Shadowing

In the beginning – There was PDP-11 Emulator Mode

- At 1.0, vast majority of non-privileged code ran in 16-bit mode
- VAX/VMS went from initial architectural work to first release in late 1977
- 18 month timeline – hardware and software
- 16-bit code replaced in successive stages
- Allowed prioritization of engineering resources

VAX to ALPHA – Part Deux

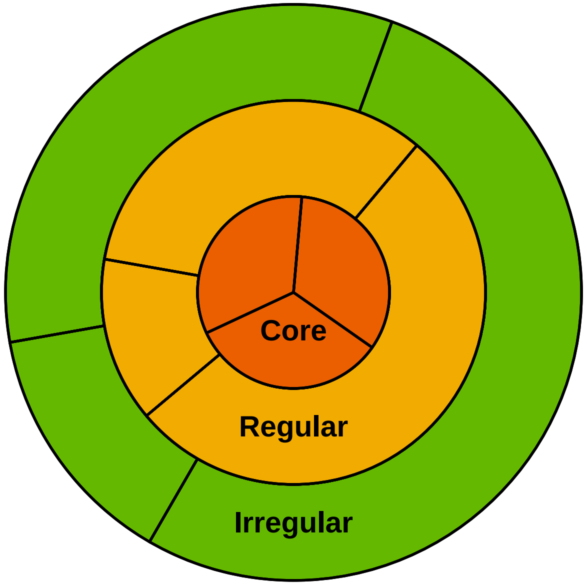
- MACRO-32 compiler
- VEST
- Far larger code base, similar timeline
- OpenVMS MONITOR utility
  - Translated from initial release through 7.3-2
  - Unnoticed by overwhelming majority of users
  - A perfect example of why this strategy works

Alpha to Integrity – Deja Vu

- MACRO-32 compiler
- AEST
- A tried and proven, low risk approach

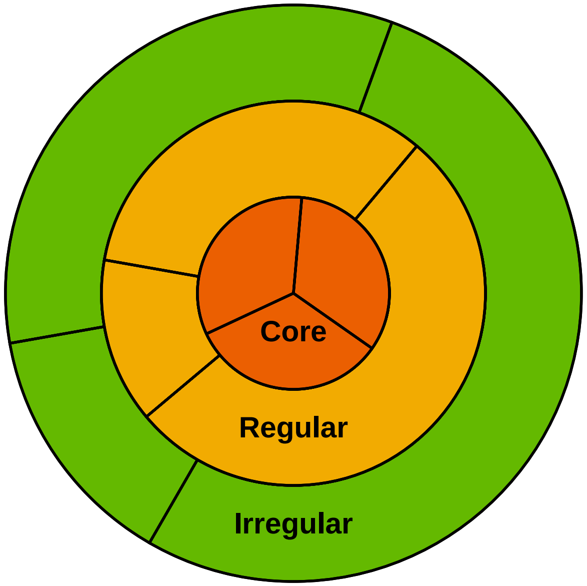
Characterize your Systems and Applications

- Mission Critical/Routine/Irregular
- Criticality
- Complexity
- Dependencies
- Visibility



Your (Organization's) Comfort Zone

- Management's outlook
- Technical perspective
- De-rate positive results



Your (Organization's) Comfort Zone – Part Deux

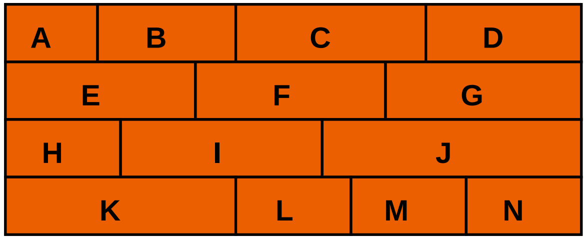
A continuum between two extremes

- Core workload first, Outriders last
  - First cutover inherently has a higher risk
  - Easier to justify resources/budget
- Outriders first, Core workload last
  - Builds confidence with low risk
  - Harder to justify resources/budget
- Reality – Infinite range of possibilities between extremes
  - Carefully consider politics and commitments
  - On balance, outriders are an easier/safer way 1st step

Image Translation

- Translates binary executables from one architecture to another
- Does not require up-to-date source files
- Bug-for-bug compatibility
- Some loss of performance is expected

How do we get from Alpha to Integrity?



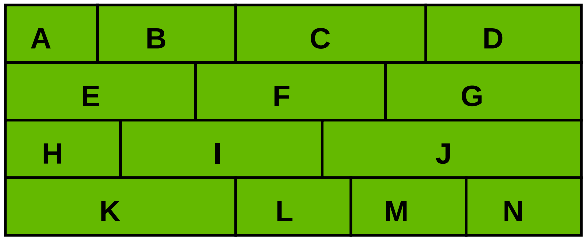
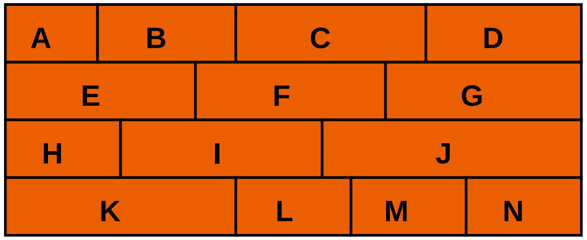


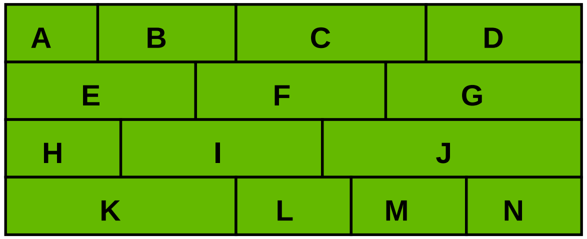
Image Translation – An Alpha/Integrity Example

```
$ aest helloworld
$ run helloworld_av
Hello world!
$
```

AEST – Alpha Image Translator

- Executable images linked at 6.1 or later
- Any supported language
- Previously VEST'ed images
- Main programs or shareable images



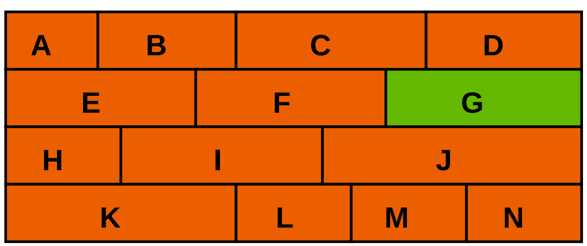


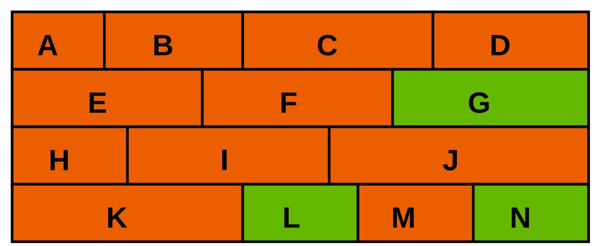
Use of Translated Images – Divide, Mix 'n Match

- Break up Alpha images into collections of shareable images
- Compile on Alpha
- Translate to Integrity
- Going native is now an incremental process
- Compile/link with /TIE and /NONATIVE_ONLY

Incremental Process of Translation to Native

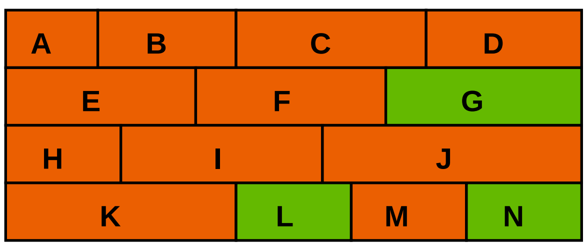
- One/more shareable image at a time
- Start with fully translated environment
- Replace components
- Efforts can run in parallel (e.g. Group, Process Logical Names)

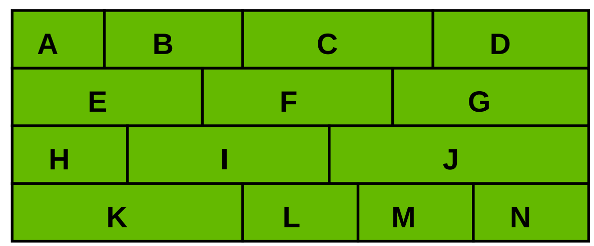




Till all code in all applications is native

- Long Term Goal – Pure native environment
- Long Term means Long Term – Remember MONITOR

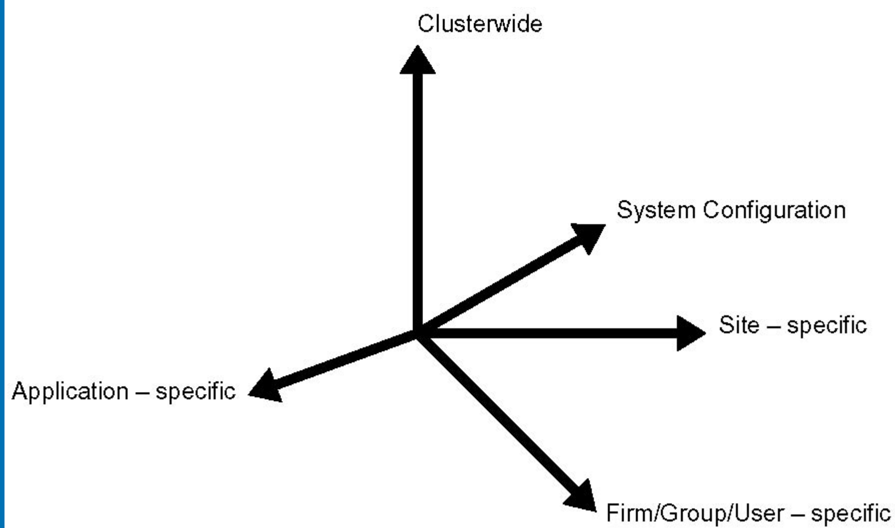


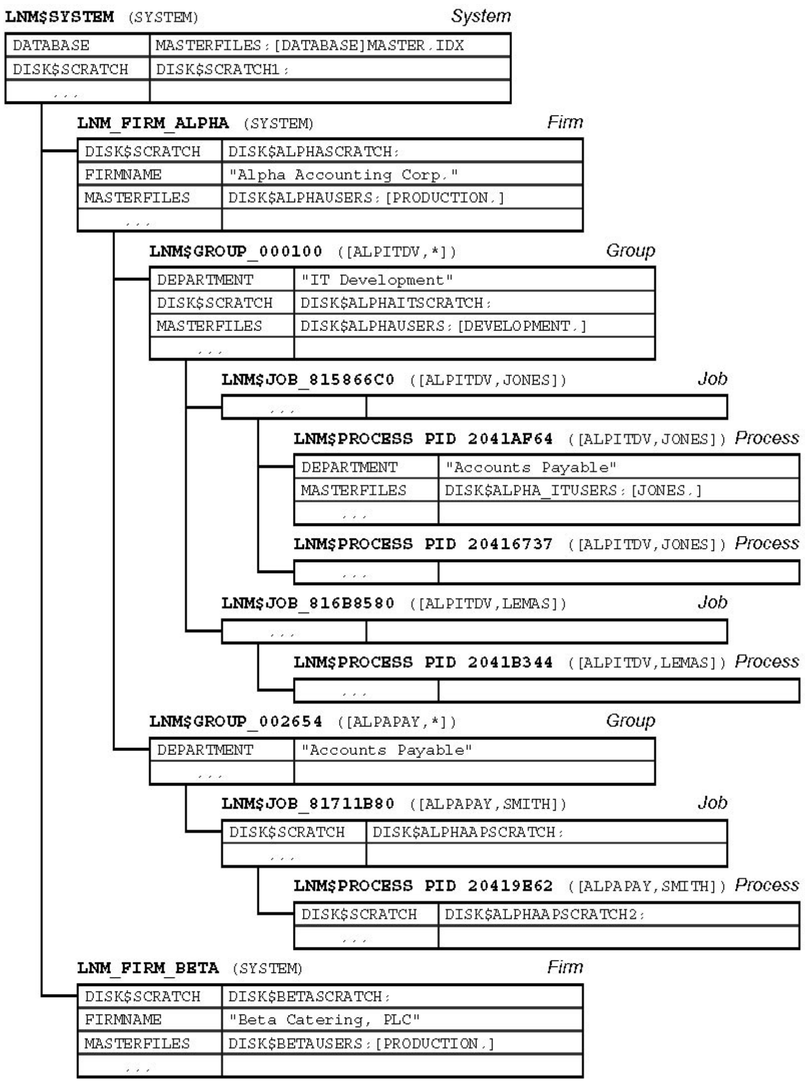


Hierarchical Logical Names

- Out of the box Logical names can be different for different groups, users, jobs, and processes
- Additional levels can be added by (supported) alternations to LNM$_FILE_DEV
- Reference: OpenVMS Technical Journal, V3

Hierarchical Logical Names II





From *Inheritance Based Environments in OpenVMS Systems and VMSClusters,* OpenVMS Technical Journal, V3

Mixed Architecture OpenVMS clusters

- Machine population based on workload
- Image translation allows decoupling of machine inventory
- Start with Alpha OpenVMS Cluster, small Integrity
- Acquire assets; Retire assets as needed
- End game is small Alpha/VAX; otherwise pure Integrity

Host Based Volume Shadowing

- Migrate storage environment regardless of underlying technologies
- NO INTERRUPTION OF USERS
- Zero Window cutover

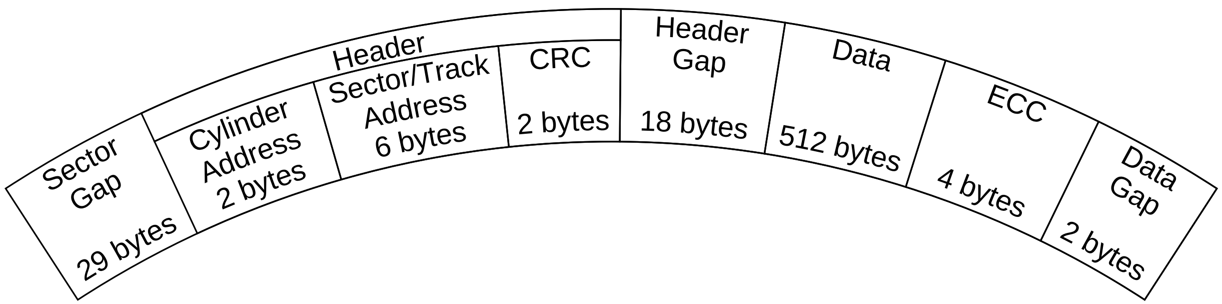
Today's Environment

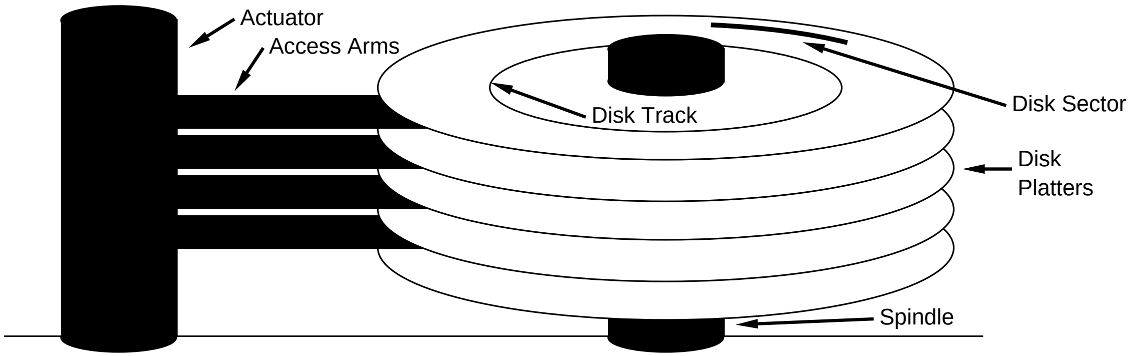
- 24 x 7; 366 days a year
- Backups are critical
- Halting Production is "not an option"
- Cluster uptime needs to be measured in years, not months.
- Technology change is inevitable, technology changes can be delayed; but not frozen

Today’s Tools

- OpenVMS – on Alpha and Integrity
- Shadowing
  - Storage Management v. redundancy
- Enterprise arrays
- Managing workload
- LD (“Logical Disk”)

Disks are Disks

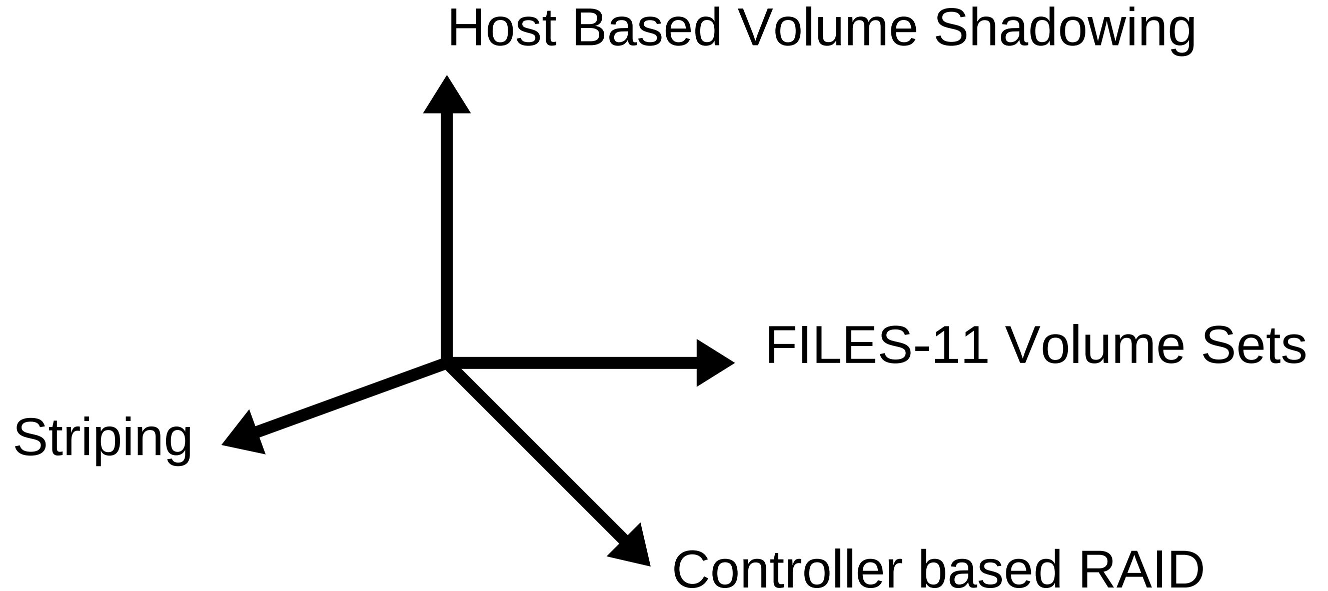




Storage Strategy

- Disruption/Downtime/Non-Availability – Evil
- *Housekeeping* load is undesirable, but tolerable if prudently scheduled
- Growth works; shrink does not

Independent Axes of Storage Configuration



- Host-based shadowing is independent of other techniques
- Each technique has its benefits, and weaknesses
- Each “flavor” has its place in your toolkit

Disruptive Events

- BITMAP.SYS expansion
- INDEXF.SYS expansion
- Global changes in directory structure
- Global changes in file ownership/protection

BITMAP.SYS

- One bit/cluster on the disk, maximum size 65535 blocks
- In today's terms, the bitmap is a negligible burden:
  - each bitmap block represents 4096 clusters
  - largest possible bitmap is approximately 33.55 MB
  - on a 1GB disk, there are approximately 2,000,000 blocks
  - the maximum bitmap is less than 0.00005% of the disk
- Maximizing the size of the bitmap is insignificant on large disks

INDEXF.SYS

- At least one file header per file, possibly more
- Initial default allocation is 16; which leaves space for 10 files (6 entries are used by the FILES-11 reserved files)
- INDEXF.SYS is limited to a single file header
- INDEXF.SYS will extend as needed, but fragmentation of extensions means that HEADERFULL will occur long before the volume runs out of space.

Global Changes in Directory Structure

- Execution disruptive
- Profile disruptive
- Requires synchronization over potentially entire user base, unlikely to be achieved
- Use rooted logical names to separate departments, applications, and other logical groupings of files

Global Changes in File Ownership/Protection

- Execution disruptive
- User confusion
- Problem report intensive

Overall Strategic Goals

- Avoid interruptions and unavailability at all costs
- Assimilate multiple generations of storage devices without interruption

Tactical Goals

- Avoid shortages of file headers in INDEXF.SYS
- Avoid volume reorganization caused by expansion of BITMAP.SYS

Implementation Tactics

- Use OpenVMS Volume Shadowing to migrate volumes to larger physical volumes online
- Use INITIALIZE command to set the stage:
  - /LIMIT to force bitmap to allow for future expansion
  - /MAXIMUM_FILES to large value
  - /HEADERS to value commensurate with /MAXIMUM_FILES and /CLUSTER
- Test procedures using Logical Disks; experiment without
  - needing additional hardware
  - avoid large shadow copies during tests, the facilities do scale

Implementation Tactics (cont'd)

- Experiments on small, offline systems are infinitely cheaper than encountering problems when working with live systems.
  - Small test systems are invaluable for testing SYSGEN parameters relating to shadowing
  - It is well worth using a DS (Alpha), rx1600/2600 or other older/smaller machine to experiment with

The Plan

- Make up; then break up
- Straddling

- (diagram goes here)

Use LD to Create Logical Drives

```
$ ld create pseudodisk1.dsk/size=15000
$ ld create pseudodisk2.dsk/size=15000
$ ld create pseudodisk3.dsk/size=30000
$ ld create pseudodisk4.dsk/size=30000
$ ld create pseudodisk5.dsk/size=45000
$ ld create pseudodisk6.dsk/size=45000
```

Initialize and Build the Initial Shadow Set (Stage A)

```
$ ld connect pseudodisk1.dsk/symbol
%LD-I-UNIT, Allocated device is $1$LDA16:
$ allocate ld16 shadowmember1:
%DCL-I-ALLOC, _$1$LDA16: allocated
$ ld connect pseudodisk2.dsk/symbol
%LD-I-UNIT, Allocated device is $1$LDA17:
$ allocate ld17 shadowmember2:
%DCL-I-ALLOC, _$1$LDA17: allocated
$ initialize/shadow=(shadowmember1:,shadowmember2:)-
  /structure=5-
  /cluster=3/limit=3145728/erase shadowtest
$ deallocate shadowmember1:
$ deallocate shadowmember2:
$ mount/system
 dsa/shadow=(shadowmember1,shadowmember2) shadowtest
%MOUNT-I-MOUNTED, SHADOWTEST mounted on _DSA9999:
%MOUNT-I-SHDWMEMSUCC, _$1$LDA16: (ALFA) is now a
 valid member of the shadow set
%MOUNT-I-SHDWMEMSUCC, _$1$LDA17: (ALFA) is now a
 valid member of the shadow set
```

Create a Directory on the shadow set

```
$ create/directory disk$shadowtest:[gezelter]
$ show device disk$shadowtest

Device                 Device            Error    Volume         Free  Trans Mnt
 Name                  Status            Count     Label        Blocks Count Cnt
DSA9999:               Mounted               0  SHADOWTEST       14547     1   1
$1$LDA16: (ALFA)  ShadowSetMember            0  (member of DSA9999:)
$1$LDA17: (ALFA)  ShadowSetMember            0  (member of DSA9999:)
```

Move some files to the shadowset

```
$ copy *.*/exclude=(*.dsk;*,*.dir;*) disk$shadowtest/log
%COPY-S-COPIED, SYS$SYSDEVICE:[GEZELTER]ACCOUNTS.TMP;27 copied to
disk$shadowtest:[GEZELTER]ACCOUNTS.TMP;27 (2 blocks)
%COPY-S-COPIED, SYS$SYSDEVICE:[GEZELTER]ADDSHADOWMEMBER.COM;3 copied to
disk$shadowtest:[GEZELTER]ADDSHADOWMEMBER.COM;3 (1 block)
%COPY-S-COPIED, SYS$SYSDEVICE:[GEZELTER]ALPHAFTPAKSJUNE.COM;1 copied to
disk$shadowtest:[GEZELTER]ALPHAFTPAKSJUNE.COM;1 (36 blocks)
                                        • …
%COPY-S-COPIED, SYS$SYSDEVICE:[GEZELTER]TCPIP$FTP_SERVER.LOG;11 copied to
disk$shadowtest:[GEZELTER]TCPIP$FTP_SERVER.LOG;11 (1 block)
%COPY-S-COPIED, SYS$SYSDEVICE:[GEZELTER]UNZIP.EXE;1 copied to
disk$shadowtest:[GEZELTER]UNZIP.EXE;1 (278 blocks)
%COPY-S-COPIED, SYS$SYSDEVICE:[GEZELTER]X.TMP;1 copied to
disk$shadowtest:[GEZELTER]X.TMP;1 (1 block)
%COPY-S-COPIED, SYS$SYSDEVICE:[GEZELTER]ZIP.EXE;1 copied to
disk$shadowtest:[GEZELTER]ZIP.EXE;1 (194 blocks)
%COPY-S-NEWFILES, 22 files created
$ show device disk$shadowtest

Device                    Device           Error    Volume         Free  Trans
  Mnt
 Name                     Status           Count     Label        Blocks Count
  Cnt
DSA9999:                  Mounted              0  SHADOWTEST        6936     1
  1
$1$LDA16:     (ALFA)  ShadowSetMember      0  (member of DSA9999:)
$1$LDA17:     (ALFA)  ShadowSetMember      0  (member of DSA9999:)
```

Detailed Look at the Shadow Set

```
$ show device disk$shadowtest/full

Disk DSA9999:, device type Foreign disk type 1, is online, mounted, file-
  oriented device, shareable, available to cluster, error logging is
  enabled,
    device supports bitmaps (no bitmaps active).

    Error count                    0    Operations completed
  366
    Owner process                 ""    Owner UIC
  [SYSTEM]
    Owner process ID        00000000    Dev Prot
  S:RWPL,O:RWPL,G:R,W
    Reference count                1    Default buffer size
  512
    Total blocks               15000    Sectors per track
  11
    Total cylinders              124    Tracks per cylinder
  11
    Logical Volume Size        15000    Expansion Size Limit
  3158016

    Volume label         "SHADOWTEST"   Relative volume number
  0
    Cluster size                   3    Transaction count
  1
    Free blocks                 6936    Maximum files allowed
  393216
    Extend quantity                5    Mount count
  1
    Mount status              System    Cache name
  "_$1$DKA100:XQPCACHE"
    Extent cache size             64    Maximum blocks in extent cache
  693
```

And the Shadow Sets Members

```
Disk $1$LDA16:, device type Foreign disk type 1, is online, member of shadow
   set DSA9999:, shadow set virtual unit.

   Error count                    0    Shadow member operation count
   583
   Allocation class               1

Disk $1$LDA17:, device type Foreign disk type 1, is online, member of shadow
   set DSA9999:, shadow set virtual unit.

   Error count                    0    Shadow member operation count
   579
   Allocation class               1
```

First Crisis: Space Shortage

- There is a shortage of free space on DISK$SHADOWTEST
- There is clearly no shortage of file headers
- Resolution:
  - switch to larger volumes without interrupting users and applications
  - straddle to the new, larger volumes; then release smaller volumes

Create the New Shadow Set Member

```
$ ld connect pseudodisk3.dsk/symbol
%LD-I-UNIT, Allocated device is $1$LDA18:
$ allocate ld18 newmember
%DCL-I-ALLOC, _$1$LDA18: allocated
$ initialize newmember: scratch_disk
$ deallocate newmember
$ mount/system disk$shadowtest/shadow=newmember-
  /policy=verify_label shadowtest
%MOUNT-I-MOUNTED, SHADOWTEST mounted on _DSA9999:
%MOUNT-I-SHDWMEMCOPY, _$1$LDA18: (ALFA) added to the shadow set
  with a copy operation
%MOUNT-I-ISAMBR, _$1$LDA16: (ALFA) is a member of the shadow set
%MOUNT-I-ISAMBR, _$1$LDA17: (ALFA) is a member of the shadow set
$
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:25:22.61  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVINICPY, initiating copy operation on
  _DSA9999: at LBN: 0, I/O size: 127 blocks, ID number: 0400076E.

%%%%%%%%%%%  OPCOM  23-AUG-2005 06:25:37.64  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVNORMAL, successful completion of copy
  operation on device _DSA9999: at LBN: 15000, ID number:
  0400076E.
```

Then Release one of the Smaller Volumes

```
$ DISMOUNT LD16
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:25:39.81  %%%%%%%%%%%
$1$LDA16: (ALFA) has been removed from shadow set.
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:25:40.62  %%%%%%%%%%%
DSA9999: shadow set has been reduced.
```

Complete the Straddle: Add the Second Stage 2 Volume

```
$ ld connect pseudodisk4.dsk/symbol
%LD-I-UNIT, Allocated device is $1$LDA19:
$ allocate ld19 newmember
%DCL-I-ALLOC, _$1$LDA19: allocated
$ initialize newmember: scratch_disk
$ deallocate newmember
$ mount/system disk$shadowtest/shadow=newmember-/policy=verify_label
  shadowtest
%MOUNT-I-MOUNTED, SHADOWTEST mounted on _DSA9999:
%MOUNT-I-SHDWMEMCOPY, _$1$LDA19: (ALFA) added to the shadow set with a copy
  operation
%MOUNT-I-ISAMBR, _$1$LDA17: (ALFA) is a member of the shadow set
%MOUNT-I-ISAMBR, _$1$LDA18: (ALFA) is a member of the shadow set
$
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:26:46.71  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVINICPY, initiating copy operation on _DSA9999: at LBN:
  0, I/O size: 127 blocks, ID number: 0400076B.
$
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:27:00.70  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVNORMAL, successful completion of copy operation on
  device _DSA9999: at LBN: 15000, ID number: 0400076B.
$ DISMOUNT LDA17
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:27:30.90  %%%%%%%%%%%
$1$LDA17: (ALFA) has been removed from shadow set.
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:27:32.70  %%%%%%%%%%%
DSA9999: shadow set has been reduced.
```

Now Check the Space Situation

```
$ show device disk$shadowtest

Device                  Device           Error    Volume         Free  Trans
  Mnt
 Name                   Status           Count     Label        Blocks Count
  Cnt
DSA9999:                Mounted              0  SHADOWTEST        6936      1
  1
$1$LDA18:     (ALFA)    ShadowSetMember      0  (member of DSA9999:)
$1$LDA19:     (ALFA)    ShadowSetMember      0  (member of DSA9999:)
```

- The old, 15K block volumes have been replaced with 30K block volumes without interrupting users
- We now need to use Dynamic Volume Expansion to increase the space available to users

```
$ set volume/size=30000 disk$shadowtest
$ show device disk$shadowtest

Device                  Device           Error    Volume         Free  Trans
  Mnt
 Name                   Status           Count     Label        Blocks Count
  Cnt
DSA9999:                Mounted              0  SHADOWTEST       21936      1
  1
$1$LDA18:     (ALFA)    ShadowSetMember      0  (member of DSA9999:)
$1$LDA19:     (ALFA)    ShadowSetMember      0  (member of DSA9999:)
```

Time Elapses: déjà vu – We are again out of available space

```
$ ld connect pseudodisk5.dsk/symbol
$ LD-I-UNIT, Allocated device is $1$LDA20:
$ allocate ld20 newmember
%DCL-I-ALLOC, _$1$LDA20: allocated
$ initialize newmember: scratch_disk
$ deallocate newmember
$ mount/system disk$shadowtest/shadow=newmember-
/policy=verify_label shadowtest
%MOUNT-I-MOUNTED, SHADOWTEST mounted on _DSA9999:
%MOUNT-I-SHDWMEMCOPY, _$1$LDA20: (ALFA) added to the shadow set with a copy
operation
%MOUNT-I-ISAMBR, _$1$LDA18: (ALFA) is a member of the shadow set
%MOUNT-I-ISAMBR, _$1$LDA19: (ALFA) is a member of the shadow set
$
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:28:25.82  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVINICPY, initiating copy operation on _DSA9999: at LBN:
0, I/O size: 127 blocks, ID number: 05000764.
$
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:28:45.85  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVNORMAL, successful completion of copy operation on
device _DSA9999: at LBN: 30000, ID number: 05000764.
$ dismount lda18
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:28:59.67  %%%%%%%%%%%
$1$LDA18: (ALFA) has been removed from shadow set.
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:28:59.84  %%%%%%%%%%%
DSA9999: shadow set has been reduced.
```

Complete the second straddle

```
$ ld connect pseudodisk6.dsk/symbol
%LD-I-UNIT, Allocated device is $1$LDA21:
$ allocate ld21 newmember
%DCL-I-ALLOC, _$1$LDA21: allocated
$ initialize newmember: scratch_disk
$ deallocate newmember
$  mount/system disk$shadowtest/shadow=newmember-
/policy=verify_label shadowtest
%MOUNT-I-MOUNTED, SHADOWTEST mounted on _DSA9999:
%MOUNT-I-SHDWMEMCOPY, _$1$LDA21: (ALFA) added to the shadow set
  with a copy operation
%MOUNT-I-ISAMBR, _$1$LDA19: (ALFA) is a member of the shadow set
%MOUNT-I-ISAMBR, _$1$LDA20: (ALFA) is a member of the shadow set
$
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:29:26.90  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVINICPY, initiating copy operation on
  _DSA9999: at LBN: 0, I/O size: 127 blocks, ID number: 04000761.
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:29:46.92  %%%%%%%%%%%
Message from user SYSTEM on ALFA
%SHADOW_SERVER-I-SSRVNORMAL, successful completion of copy
  operation on device _DSA9999: at LBN: 30000, ID number:
  04000761.
$ dismount lda19
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:29:56.34  %%%%%%%%%%%
$1$LDA19: (ALFA) has been removed from shadow set.
%%%%%%%%%%%  OPCOM  23-AUG-2005 06:29:56.90  %%%%%%%%%%%
DSA9999: shadow set has been reduced.
```

Complete the process of making space available to users

```
$ show device disk$shadowtest

Device                     Device            Error    Volume         Free  Trans
  Mnt
 Name                      Status            Count     Label        Blocks Count
  Cnt
DSA9999:                   Mounted               0  SHADOWTEST       21936     1
  1
$1$LDA20:     (ALFA)  ShadowSetMember        0  (member of DSA9999:)
$1$LDA21:     (ALFA)  ShadowSetMember        0  (member of DSA9999:)
```

- As in the previous iteration, we need to use the SET VOLUME/SIZE command to make the full extent of the new shadow set volumes available for use

```
$ set volume/size=45000 disk$shadowtest
$ show device disk$shadowtest

Device                     Device            Error    Volume         Free  Trans
  Mnt
 Name                      Status            Count     Label        Blocks Count
  Cnt
DSA9999:                   Mounted               0  SHADOWTEST       36936     1
  1
$1$LDA20:     (ALFA)  ShadowSetMember        0  (member of DSA9999:)
$1$LDA21:     (ALFA)  ShadowSetMember        0  (member of DSA9999:)
```

What about the files we placed on the shadow set?

- Our files are still where we placed them
- There was no interruption of file access or availability at any point in the preceding two generations of storage devices

```
$ dir disk$shadowtest

Directory DISK$SHADOWTEST:[GEZELTER]

ACCOUNTS.TMP;27    ADDSHADOWMEMBER.COM;3
ALPHAFTPAKSJUNE.COM;1                DECW$ENDSESSION.DECW$XAUTH;13
decw$sm.log;13     DEPARTMENTS.TMP;2  DEPARTMENTS.TXT;3  DKA300.LST;1
HPWORLD.ZIP;1      INFO-ZIP.ZIP;1     LOGIN.COM;78       MAKEACCOUNTS.COM;21
MAKESIGN.COM;1     NET$SERVER.LOG;13  openvms-alphapak.txt;1
RIGHTSLIST.DAT;1   SHADEXAMPLE1.COM;8 SYSUAF.DAT;1
TCPIP$FTP_SERVER.LOG;11              UNZIP.EXE;1        X.TMP;1
ZIP.EXE;1

Total of 22 files.
$
```

Summary

- Users have uninterrupted access to data throughout this process. To recap:
  - two sets of storage device transitions
  - size of individual volume grew 300% from
    - an initial volume size of 15K blocks
    - to an intermediate stage size of 30K blocks; and
    - finally to a size of 45K blocks
  - all the space on the volumes was available at every stage

Besides Increasing Space How Can This Be Used?

- Changes in striping
- Changes in physical hardware
- Any change that does not affect underlying FILES–11 structures (e.g., cluster factor); everything else is limited only by your imagination

Example Command Files Used for this exercise

The command files used for the demonstration will be posted on my www site at:

http://www.rlgsc.com/hptechnologyforum/2006/index.html