PARSEC Group

Our Trainers Consult. Our Consultants Train.

Find the latest webinars at:

OpenVMS PLANET.org

News, Views, Tips, Forums, Jobs, Resources, and More!

Comparing the Advanced FileSystem/AdvFS (Tru64 UNIX) & Logical Volume Manager/LVM (HP-UX)

Presented by

John Lanier

www.parsec.com | 888-4-PARSEC

INTRODUCTION

When talking in terms of **filesystems**, we are talking (more or less) about the ability of the operating system to perform administrative tasks with relationship to:

- Files used by/for the system
- Files used by users on the system
- How disk space will be allotted to different files on the disk
- Managing/monitoring disk usage as a whole

This presentation will focus on key components of 2 filesystems: **AdvFS (Tru64 UNIX)** and **LVM (HP-UX),** with relationship to their architecture, implementation and system management capabilities. We will also discuss some of the similarities and differences that exist between **AdvFS** and **LVM**, namely:

1. Transaction Logging
2. Logical volume Maxims
3. Online resizing
4. Online backup
5. Comparing AdvFS File Domains and LVM Volume Groups

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS vs. JFS→OnLineJFS→VxFS→LVM

- Both are **log-based, journaled file systems** with comparable capabilities and are managed using similar concepts.
- AdvFS filesets are not supported on HP-UX.
- **Filesets** are the mountable portions of **domains**, the latter being a **pool of storage** that contains **logical volumes** (disk partitions, RAID-enabled LUNs, LSM volumes) for use by AdvFS.
- The **Logical Volume Manager (LVM)** can be considered a subsystem for managing disk space.
- **LVM** allows the user to consider the disks (a.k.a. physical volumes) as a pool of data storage, consisting of equal-sized **extents**.
- The default size of an extent is **4 MB**.
- **Extent(s): Contiguous area(s) / block(s) on disk**.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM (At a Glance)

- **LVM** consists of groupings of physical volumes that are organized into **volume groups**.
- A volume group can consist of one or more physical volumes.
- There can be more than one volume group in the system.
- Once created, the volume group, not the disk, is the basic unit of data storage.
- While one might move disks from one system to another, with LVM one would move a volume group from one system to another; thus it is often convenient to have multiple volume groups on a system.

LVM (At a Glance; cont.)

- Volume groups can be subdivided into virtual disks, or **logical volumes**.
- A **logical volume** can span a number of physical volumes or represent only a portion of one physical volume.
- The pool of disk space that is represented by a volume group can be divided into logical volumes of various sizes.
- The size of a logical volume is determined by its number of extents.
- Once created, logical volumes can be treated just like disk partitions.
- Logical volumes can be assigned to:

→Filesystems

→Swap or dump devices

→Raw access (underlying storage that bypasses data caching techniques at the block device file level).

Installing AdvFS

- AdvFS is bundled with Tru64 UNIX Version 5.x (it is also the default filesystem as of Tru64 UNIX V5.x).
- Basic commands and features are part of the initial OS install and licensing (**one logical volume per domain, showfdmn/showfset, etc.).**
- “Advanced Utilities” are a layered product installed separately; requires additional licensing (**addvol, rmvol, clonefset, defragment, mktrashcan, etc.).**

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Installing LVM

When performing a “cold” (i.e. fresh) install on an HP-UX system, the following choices are provided to configure the filesystem type:

- VERITAS Volume Manager (VxVM) with VxFS
- Whole disk with VxFS
- **Logical Volume Manager (LVM) with VxFS ←Default (recommended)**

AdvFS Features

- Write-ahead logging for reduced file-system (FS) inconsistencies and rapid recovery
- Reconfiguration and tuning can be done online without service interruption
- Uses an extent-based file allocation scheme for fewer I/O transfers of a larger size increasing sequential read/write throughput
- Supports multivolume FS, including file-level striping across several disks
- GUI interface available (dtadvfs/CDE, dxadvfs/DECWindows Motif)
- Large scale FS support for larger files and FS size (4 TB).
- Unified Buffer Cache (UBC) for dynamic allocation of memory used for caching disk I/O
- Ability to clone filesets for ‘point in time’ snapshots allowing online backups
- Online resizing (“addvol/rmvol”)
- Online defragmentation (“defragment” utility)
- File undelete (mktrashcan)

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM Features

- Allows for management of physical disks as logical volumes.
- A file system can span multiple physical disks; multi-volume FS's.
- Flexible means of defining boundaries of disk space; boundaries are independent of one another.
- Journaled FileSystem (JFS): Intent log (journal) is referenced for reduced file-system inconsistencies & rapid recovery after a crash.
- **Online JFS**: FS reconfiguration and tuning can be done online without being in single-user mode. **Add-on product.**
- JFS/OJFS uses an "extent-based" file allocation scheme

  →Physical Extents: Contiguous disk blocks on a physical volume.

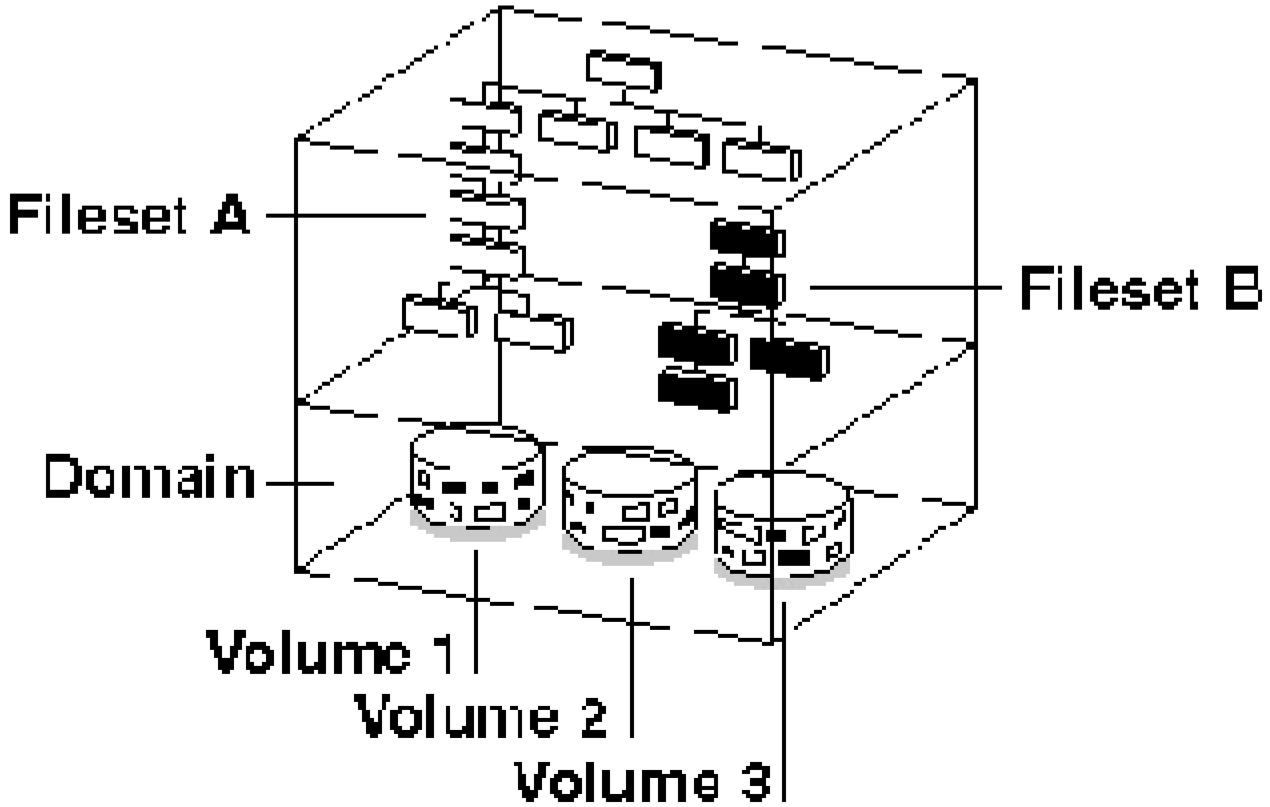
  →Logical Extents: Same size as physical extents in a volume group; can map to physical extents on different disks.
- GUI interface available (SAM)
- FS support for larger files and FS size (4 TB).
- **Online resizing supported, but not without Online JFS.**

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS & LVM Architecture (At a Glance)

- **The directory hierarchy layer**
  - File-naming
  - Creating and opening files
  - Reading to/writing from files
- **The physical storage layer**
  - Write-ahead/Transaction/Intent logging
  - Caching (speed in retrieving most recently accessed data)
  - File allocation
  - Physical disk I/O functions

AdvFS Design (At a Glance)



LVM/JFS Design (Bottom-up, At a Glance)

JFS→Journaled File System; fast err/crash recovery

Volume→Device used for a filesystem, swap space, or raw data

Physical Volume→Disk not initialized for LVM use

Logical Volume→Space defined in a volume group

Volume Group→Collection of logical volumes

Filesystem→Means of managing files & directories

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Some Notes on Logical Volume Manager (LVM) Versus VxVM

- Do not be confused about the fact that Logical Volume Manager (LVM)performs a different set of functions/tasks than that of VxVM. They are two volume management products available within the HPUX operating system, but the two do not always work together in tandem.
- LVM deals more in the realm of logical volume group management and administration (see “man –k vg” for details).
- VxVM deals more in the realm of logical volume management and administration (see “man –k vx” for details).
- By this comparison, VxVM in HPUX is very similar in nature to LSM in Tru64 UNIX.

AdvFS Cloning (clonefset)

- A clone is a read-only version of the fileset, and only the structure is copied at first.
- Utilizes a “Copy-On-Write” concept; files written to are copied over to the clone, maintaining a true point in time version with a minimum of space.
- Online backups can be done from the clone, minimizing impact to active users.
- A daily snapshot can be mounted as a method to reduce restores by providing a copy of “yesterday’s” files.

Example of AdvFS Cloning

- Create the clone:
  - clonefset users_domain home1_fset home1_clone
- Mount / unmount the clone:
  - mount users_domain#home1_clone /yesterday_home1
  - umount /yesterday_home1
- Remove the clone:
  - rmfset users_domain home1_clone
- Rename the clone:
  - clone filesets cannot be renamed, they must be destroyed and recreated with the new name

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM Online Backup & Other Utilities

- Online backup requires existence of OnLine JFS

→Includes a filesystem “snapshot” capability

- Use the “fsadm” command with OnLine JFS to perform online backups
- Other features of the Online JFS component:

→Online defragmentation

→Online resizing

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS Volume Structure

- Each file domain has several structures:
  - Bitfile metadata table (BMT) : used to store file data structure, attributes, extent maps, fileset attributes and POSIX file statistics
  - Storage bitmap : use to track free and allocated disk space
  - Miscellaneous metadata bitfile : maps areas of the volume that are not AdvFS metadata, like the disk label and boot block information
  - Transaction log : only on one volume, this stores changes until they are written out to disk
  - Root tag directory : only on one volume, defines the location of all filesets in the file domain

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Example of Creating a Domain and Fileset with AdvFS

1. Create an AdvFS domain:

```
root@artemis in / --> mkfdmn /dev/disk/dsk5c app_domain
```

(NOTE: This could have been dsk5g, etc – whole disks are not required)

2. Create an AdvFS fileset:

```
root@artemis in / --> mkfset app_domain app1_fset
```

3. Create a mountpoint:

```
root@artemis in / -->mkdir /app1
```

4. Mount the domain and fileset:

```
root@artemis in / --> mount app_domain#app1_fset /app1
```

5. Confirm the results:

```
root@artemis in / --> df -h /app1
Filesystem              Size     Used   Available Capacity  Mounted on
app_domain#app1_fset   8184M     16K       8179M     1%       /app1
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

JFS/LVM Design

JFS→Journaled File System; fast err/crash recovery

Volume→Device used for a filesystem, swap space, or raw data

Physical Volume→Disk not initialized for LVM use

Logical Volume→Space defined in a volume group

Volume Group→Collection of logical volumes

Filesystem→Means of managing files & directories

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM usage checklist

The basic steps to be able to use LVM are as follows:

✓Identify the disks to be used for LVM

✓Create an LVM data structure on each identified disk (see “man pvcreate”)

✓Collect all the physical volumes to form a new volume group (see “man vgcreate”)

✓Create logical volumes from the space in the volume group (see “man lvcreate”)

✓Use each logical volume as if it were a disk section (create a file system, or use for raw access)

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Example of Constructing a Filesystem Using LVM (1 of 4)

1. Create a physical volume from disk c3t0d0:

```
root@merc72 in / --> pvcreate  -f /dev/rdsk/c3t0d0
Creating "/etc/lvmtab".
Physical volume "/dev/rdsk/c3t0d0" has been successfully created.
```

2. Create a volume group:

From "man vgcreate":

```
.........

The minor number for the group file should be unique among all
the volume groups on the system.  It has the format 0xNN0000,
where NN runs from 00 to ff.  The maximum value of NN is
controlled by the kernel tunable parameter maxvgs.
.........
 -s pe_size             Sets the number of megabytes in each physical
                         extent, where pe_size is expressed in units
                         of megabytes (MB) in the range 1 to 256.
                         pe_size must be equal to a power of 2 (1, 2,
                         4, 8, etc.).  The default value for pe_size
                         is 4 (four megabytes).
.........
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Example of Constructing a Filesystem Using LVM (2 of 4)

<u>So I create the volume group and add disk "c3t0d0" to the volume group:</u>

```
root@merc72 in / --> mknod /dev/rz1cbvg/group c 64 0x000000
root@merc72 in / -->
root@merc72 in / --> file /dev/rz1cbvg/group
/dev/rz1cbvg/group:     character special (64/0)
root@merc72 in / -->
root@merc72 in / --> vgcreate -s 2 /dev/rz1cbvg /dev/dsk/c3t0d0

Increased the number of physical extents per physical volume to 2045.
Volume group "/dev/rz1cbvg" has been successfully created.
Volume Group configuration for /dev/rz1cbvg has been saved in
/etc/lvmconf/rz1cbvg.conf
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Example of Constructing a Filesystem Using LVM (3 of 4)

3. Create the logical volume (1GB), so that I can use it for FS purposes:

```
root@merc72 in / --> lvcreate -L 1000 /dev/rz1cbvg

Logical volume "/dev/rz1cbvg/lvol1" has been successfully created with
character device "/dev/rz1cbvg/rlvol1".
Logical volume "/dev/rz1cbvg/lvol1" has been successfully extended.
Volume Group configuration for /dev/rz1cbvg has been saved in /etc/lvmconf/rz1cbvg.conf
```

From "man lvcreate":

```
 -L lv_size             Allocate space to the logical volume,
                        specified in megabytes.  lv_size is a
                        decimal value in the range 1 to 16777216
                        (the implementation limit).  lv_size is
                        rounded up to the nearest multiple of
                        the logical extent size, equivalent to
                        the physical extent size defined for the
                        volume group by the vgcreate command
                        (see vgcreate(1M)).  The default is
                        described above.
 -n lv_name             Set the name of the new logical volume
                        to lv_name, where lv_name is a simple
                        file name, not a path name.
..........
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Example of Constructing a Filesystem Using LVM (4 of 4)

4. Create the filesystem:

```
root@merc72 in / --> newfs -F vxfs /dev/rz1cbvg/rlvol1  <--character (raw) device

    version 5 layout
    1024000 sectors, 1024000 blocks of size 1024, log size 16384 blocks
    unlimited inodes, largefiles not supported
    1024000 data blocks, 1007288 free data blocks
    32 allocation units of 32768 blocks, 32768 data blocks
    last allocation unit has 8192 data blocks
```

5. Mount the filesystem:

```
root@merc72 in / --> mkdir /opt2
root@merc72 in / --> mount /dev/rz1cbvg/lvol1 /opt2
root@merc72 in / --> bdf /opt2

Filesystem          kbytes    used   avail %used Mounted on
/dev/rz1cbvg/lvol1 1024000   16725  944328    2% /opt2
```

6. If I want this to survive a reboot, I need to add it to /etc/fstab:

```
root@merc72 in / --> grep /opt2 /etc/fstab
/dev/rz1cbvg/lvol1 /opt2 vxfs delaylog 0 2
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS Examples

- The mount and umount commands work the same with AdvFS, with domain and fileset being stated:
  - mount domain#fileset /mnt
  - umount /mnt
- The /etc/fstab entries have domain#fileset, but also have the filesystem type noted:

```
#cat /etc/fstab
root_domain#root  /      advfs    rw       0 1
/proc             /proc  procfs   rw       0 0
usr_domain#usr    /usr   advfs    rw       0 2
usr_domain#var    /var   advfs    rw       0 2
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM Daemons

- lvmkd – watches the lvm queue for jobs and submits them as processes managed by the kernel
- lvmschedd – periodically reschedule I/O requests that were queued because no memory was available to allocate data structures
- lvmdevd – handles volume group recovery, one per volume group
- lvmattachd – keeps track of NOT_ATTACHED physical volumes and attaches them to the volume group when the become accessible

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS Daemons

- advfsd: daemon process needed for the AdvFS GUI (dtadvfs)
- Issues commands and gathers system information on behalf of the GUI
- /usr/var/advfs/daemon/socket/hosts.allow
  - List of hosts on which advfsd will allow the GUI to run
  - Hostnames must be on separate lines
- /usr/var/advfs/daemon/socket/gui.passwd
  - If it exists, is used to force dtadvfs user to provide a unique password when connecting to the host
  - Do not use root password!

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS Daemons (cont.)

- Automatically started at bootup
  - To avoid startup, move /sbin/rc3.d/s53advfsd and start manually
- Allows SNMP clients (like NetView or Performance Manager) to request AdvFS file system information, but NOT to issue system configuration info.
- /usr/var/advfs/daemon/disks.ignore
  - Each line is the name of a disk to ignore by the daemon
  - IE: to ignore /dev/disk/dsk5b, enter line: dsk5
  - All or nothing, on a partition level
  - Good to remove a failed disk until replaced

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS Commands

- Base System Commands
  - Already installed for 5.x systems
  - No licensing required
  - Covers all basic tasks/needs
- Advanced Utilities
  - Installed from Associated Products CD-ROM or via a “full” install (all subsets, versus “mandatory” subsets, are selected for installation)
  - Requires an additional license
  - Provides additional, valuable services (volume spanning, online FS expansion, defragment, file undelete, etc.)

AdvFS Base System Commands (1 of 4)

- advfsstat : display system statistics
- advfscan : locate AdvFS partitions on disks
- chfile : change file attributes
- chfsets : change fileset attributes
- chvol : change volume attibutes
- defragment : makes files more contiguous
- edquota : edit group quotas
- fixfdmn : checks and repairs corrupted domains
- mkfdmn : make a new domain
- mkfset : make a new fileset (in existing domain)
- mountlist : checks for mounted AdvFS filesets

AdvFS Base System Commands (2 of 4)

- ncheck : create a list of files (pathnames/i-numbers) on filesets
- nvbmtpg : display formatted BMT pages
- nvfragpg : display formatted frag pages
- nvlogpg : display formatted log pages
- nvtagpg : display formatted tag pages
- quot : summarize fileset ownership
- quota : display disk usage and limits
- quotacheck : checks file system quota consistency
- quotaoff : turn off user/group quotas
- quotaon : turn on user/group quotas

AdvFS Base System Commands (3 of 4)

- renamefset : rename a fileset
- repquota : summarized disk use and quotas for filesets
- rmfdmn : remove domain
- rmfset : remove fileset
- salvage : recover file data from damaged domains
- savemeta : saves snapshot of a domain’s metadata
- shblk : display unformatted disk blocks
- shfragbf : display frag file information
- showfdmn : display domain attributes
- showfile : display file attributes
- showfsets : display fileset attributes within a domain

AdvFS Base System Commands (4 of 4)

- switchlog : move AdvFS log file to a different volume in a domain
- tag2name : print path of a file, given tag number
- vdf : display disk info for AdvFS domains and filesets
- vdump : back up filesets
- verify : check for and repair file system inconsistencies
- vfilegp : display pages of an AdvFS file
- vrestore : restore files saved with vdump
- vsbmpg : display a page from a storage bitmap (SBM) file

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS Utilities & Commands

- **addvol** : add a volume to an existing domain
- **advfsd** : starts AdvFS gui demon
- **balance** : balance the % used space between volumes
- **clonefset** : creates a ‘point in time’ read-only copy of a fileset
- **dmapi** : accesses the Data Management API
- **dtadvfs** : starts the AdvFS gui
- **lsmsa** : runs the LSM Storage Administrator
- **migrate** : move the location of a file within a domain
- **mktrashcan** : attach directories to a trashcan directory, to store deleted files
- **rmtrashcan** : removes specified directory from the trashcan directory
- **rmvol** : remove a volume from an existing domain
- **shtrashcan** : show the trashcan directory attached to a specified directory
- **stripe** : interleave storage allocation of a file across two or more volumes in one domain.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS Maxims

- **Up to 100 active (mounted) file domains per system**
  - →**Active**: at least 1 fileset per domain is mounted
- **Up to 250 volumes per active (mounted) domain**
  - →Recommended to have 3 volumes per domain
  - →Reduces risk of domain failure
- **Unlimited number of filesets per domain**
  - →More filesets, more flexibility with FS mgmt.
  - →More filesets, more management overhead
- **Quota values larger than 2 TB are supported**
  - →Tru64 Version 5.1 & higher

LVM Commands (1 of 3)

- lvchange : change the characteristics of logical volumes
- lvlnboot : create root, primary & secondary swap and dump volumes.
- lvcreate : creates a logical volume
- lvextend : increase disk space on a logical volume
- lvreduce : decrease disk space on a logical volume
- lvremove : remove logical volume(s) from a volume group
- lvsplit : split a mirrored logical volume into two logical volumes
- lvmerge : reverses lvsplit to make a single logical volume
- lvsync : synchronize stale mirrors

LVM Commands (2 of 3)

- pvcreate : make a disk an LVM disk
- pvdisplay : display info about physical volumes in a group
- pvchange : set physical volume characteristics
- pvmove : move allocated physical extents within a volume group
- pvremove : remove LVM header info and releases disk from LVM
- vgcreate : create a volume group
- vgdisplay : display info on all volume groups
- vgchange : activate/deactivate volume group(s)
- vgextend : extend a volume group (add one or more disks)

LVM Commands (3 of 3)

- vgreduce : reduce a volume group (remove disks)
- vgscan : scan all disks, looking for logical volume groups
- vgsync : synchronize mirrors that are stale in one or more logical volumes
- vgremove : removes volume group definition from the system
- vgexport : removes volume group from the system
- vgimport : adds volume group by scanning for exported groups [vgexport]
- vgcfgbackup : create or update a configuration backup file
- vgcfgrestore : display or restore configuration from file

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM Maxims

SUMMARY: Supported limits in LVM for HP-UX (11.00, 11.11 and 11.23)

SOURCE: http://docs.hp.com/en/6054/Limits_wp.htm

Parameter	Operation to change the value	Min-Default-Max Values
Max VGs per system	Kernel tunable ‘**maxvgs**’	0 – 10 – 256
Max LVs Per VG	**vgcreate(1M) -l ‘max_lv’**	1 – 255 – 255
Max PVs per VG	**vgcreate(1M) –p ‘max_pv’**	1 – 16 – 255
Max Extents per PV	**vgcreate(1M) –e ‘max_pe’**	1 – 1016 – 65535 The default value may get adjusted to (Disk space/ Extent size) of the first PV used for creating the VG, if that value is greater than 1016.
Extent Size	**vgcreate(1M) –s ‘pe_size’**	1 – 4 – 256 Mbytes

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM Maxims (cont.)

Parameter	**Operation to change the value**	**Min-Default-Max Values**
Effective size of PV	**pvcreate(1M) –s 'disk_size'** The following patches are required for supporting disks larger than 256Gb: **11.00** – Patch superseding PHKL_30553 **11.11** – PHKL_30622 **11.23** -- PHKL_31500	Extent Size –Disk capacity – 4 TB
Size of a LV	**lvcreate(1M)/lvextend(1M)-l \** **'le_number' | -L 'lv_size'**	**0 – 0 – 2 TBytes** **[I/Os beyond 2TB will be rejected]**

PARSEC Group
Our Trainers Consult. Our Consultants Train.

LVM Maxims (cont.)

Parameter	Operation to change the value	Min-Default-Max Values
Mirror copies per LV	**lvcreate(1M)/lvextend(1M) \ -m 'mirror_copies'**	0 – 0 – 2
Stripes of LV	**lvcreate –I 'stripes'**	2 – None (Must specify) – Max PVs per VG
Stripe size of LV	**lvcreate –I 'stripe_size'**	4 – None (Must specify) – 32768 Kbytes

PARSEC Group

Our Trainers Consult. Our Consultants Train.

JFS and AdvFS Differences

Some differences between JFS and AdvFS

JFS:

→Can specify a block size and intent log size other than the default

AdvFS:

→Can specify the number of pages for the log file

→Can specify whether or not to use frag files for files or fragments smaller than 8 KB

→Does not support specifying a block size

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Mount Syntax (JFS/VxFS)

mount syntax (JFS/VxFS):

To allow system calls to return before the intent log is written:
#mount -o delaylog
#mount -o tmplog

To bypass the use of an intent log:
#mount -o nolog (Recommended only for temporary file systems)

To log the data separately from the file system changes:
#mount -o nodatainlog (This option is used on systems with disks that do not support bad block re-vectoring).

To clear (zero out) all extents assigned to a file system before any data is written: **#mount -o blkclear** (this option is used in increased data security environments).

Reference "man 1m mount_vxfs" for details.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

Mount Syntax (AdvFS)

<u>mount syntax (AdvFS)</u>

- To use Atomic -Write Data Logging (data written to buffer cache as well as the transaction log in 8KB increments):

  #mount -o adl

- To dual-mount a fileset even though it has the same domain ID as another mounted fileset:

  #mount -o dual (used to mount cloned filesets)

- To flush to disk the file access time changes for reads of regular files:

  #mount -o atimes

- To mount a file system even if it was not cleanly un-mounted:

  #mount -o dirty

Reference "man 8 mount" for details.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons

<u>Operations on Log-Based File Systems</u>

HP-UX: JFS, OnLineJFS

Tru64 UNIX: AdvFS

Both support operations for the creating, mounting, un-mounting, extending, dumping, and restoring of file systems.

AdvFS and JFS Comparisons (Creating Filesystems)

<u>Creating File Systems</u>

- HP-UX & Tru64 UNIX both use newfs to create a file system.
- Tru64 UNIX uses newfs to create UFS filesystems.
- Tru64 UNIX uses mkfdmn/mkfset to create AdvFS filesystems.
- UFS in Tru64 UNIX is similiar to HFS in HP-UX
- UFS & HFS have their roots in the "original" UNIX filesystem;

  not as robust when compared to AdvFS and JFS.
- The HP-UX command “newfs” is an implementation of the HP-UX command “mkfs”.
- The command "newfs_vxfs" is also available specifically for the VxFS file system.
- Tru64 and HP-UX both use the “mkdir” command to create mount points for file systems.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons (Mounting Filesystems)

Mounting and Unmounting File Systems

- Tru64 & HP-UX use the mount & umount commands to mount & unmount file systems.
- Some options differ that specify the FS type.
- HP-UX: Filesystem type is specified with the

  "mount -F" option.
- Tru64 UNIX: "mount -t FStype" specifies the filesystem type.
- HP-UX: The command "mount_vxfs" exists for VxFS filesystems.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons (Extending Filesystems)

Extending File Systems

HP-UX: Reference the **extendfs** command

Also reference "**extendfs_vxfs**" for VxFS filesystems.

Tru64 UNIX offers the "mount -o/-u extend" option (new as of 5.1B).

From "man mount" (Tru64 UNIX 5.1B):

There are several steps needed in order to expand a file system:

1. Make additional storage space available on the underlying storage device (that is, a LSM volume or hardware RAID LUN).

2. For non-LSM volumes, modify the disk label to include additional storage. See the disklabel reference page for more information.

3. Use the mount command with the extend option to allow the file system to use the additional storage.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons (Extending Filesystems; cont.)

For an unmounted AdvFS fileset, the following is an example of volume expansion:

```
# mount -o extend domain#fileset /ausr1
```

For a mounted AdvFS fileset, the following is an example of volume expansion:

```
# mount -u -o extend domain#fileset /ausr2
```

For an an unmounted UFS file system, the following is an example of volume expansion:

```
# mount -o extend /dev/disk/dsk0g /useref
```

For a mounted UFS file system, the following is an example of volume expansion:

```
# mount -u -o extend /dev/disk/dsk0h /useracct
```

Refer to the Tru64 UNIX System Administration manual for more information.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons (Dumping and Restoring Filesystems)

GUI (HP-UX)
Can use the "**SAM**" utility to archive file systems.

GUI (Tru64 UNIX)
Can use the "**dxarchiver**" graphical user interface.

HP-UX
dump/restore:
Used to dump data from & restore it to an HFS file system.
vxdump/vxrestore:
Used to dump data from & restore it to a VxFS file system.

Tru64 UNIX
dump/restore:
Used to dump data from & restore it to a UFS file system.
vdump/vrestore:
Used for dumping data from and restoring it to AdvFS/UFS.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons (Advanced File-naming Techniques)

Listing File Names and File System Statistics

HP-UX

The "**ff**" command lists file names and statistics for a **VxFS** filesystem. This command reads the **inode list (i-list)** and directories of each special file specified, and displays the path name for each saved inode, as well as other requested information, such as files whose inodes have changed or been accessed in a specified number of days.

Tru64 UNIX

The **ncheck** command provides similar functionality as "ff" but for **AdvFS**. The **ncheck** utility lists the inode number (**i-number; UFS) or tag (AdvFS)** and pathname for files in a local file system.

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons (Advanced File-Naming Techniques; cont.)

Use of the "ff" command (HP-UX):

```
root@merc72 in / --> ff -F vxfs /dev/vx/dsk/rootdg/homevol | more
vxfs ff: /dev/vx/dsk/rootdg/homevol: 423 files selected
.       2
./lost+found    3
./lanier/ISEE_hpux/installation.htm     4
./jbridge       5
./lanier/.VRTSob        6
./jbridge/.cshrc        7
./jbridge/.exrc 8
./jbridge/.login        9
./jbridge/.profile      10
./jbridge/scripts       11
./jbridge/scripts/scripts.tar   12
………
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

AdvFS and JFS Comparisons (Advanced File-naming Techniques; cont.)

Use of the "ncheck" command (Tru64 UNIX):

```
root@marquis in / --> ncheck root_domain#root | more
root_domain#root:
3       /.tags/.
4       /quota.user
5       /quota.group
6       /cluster/.
13      /etc/.
14      /mnt/.
15      /opt/.
16      /sbin/.
18      /dev
19      /tmp
………
```

PARSEC Group
Our Trainers Consult. Our Consultants Train.

REFERENCES

- LVM Documentation:

  http://docs.hp.com/en/B2355-60103/lvm.7.html

- Tru64 UNIX Online Documentation (all versions):

  http://h30097.www3.hp.com/docs/pub_page/doc_list.html

- HPUX Online Documentation (HPUX 11i V2):

  http://docs.hp.com/en/hpux11iv2.html

- Combined Tru64 UNIX & HPUX Documentation:

  http://docs.hp.com/en/index.html

PARSEC Group
Our Trainers Consult. Our Consultants Train.

PARSEC Group

Our Trainers Consult. Our Consultants Train.

Find the latest webinars at:

OpenVMS PLANET.org

News, Views, Tips, Forums, Jobs, Resources, and More!

QUESTIONS?

www.parsec.com | 888-4-PARSEC