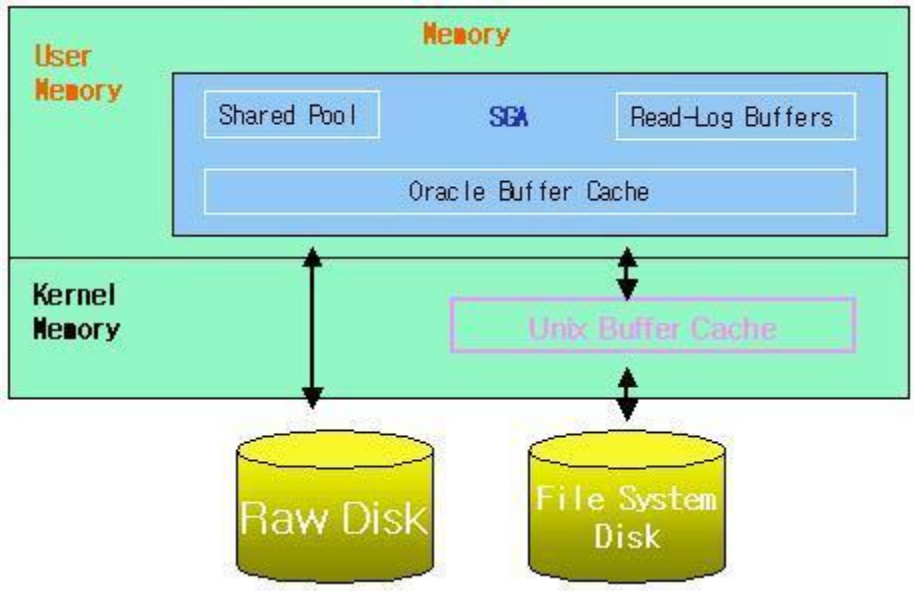
OCFS, ASM,ACFS, RAW Devices and Regular File System

Oracle, as a multi-platform product, must to have the ability to function with more than one file system and drive device configuration. There are several different storage options for a Database administrator to choose from. It is important to understand what the storage options are available to you and what the pros and cons are. In this post, we will be discussing the different options that are available with an Oracle implementation. The different options are:

1. Raw Devices.
2. Cooked file systems.
3. Oracle Clustered File System (OCFS)
4. ASM
5. ASM Cluster File System (ACFS)
6. NFS

Raw Devices



A raw device is either an entire disk, or a partition of a disk, that is not written to or controlled directly by the OS. From the OS, the contents of RAW devices are not visible from the command line. In the case of Oracle, this means you cannot view or manipulate the data files in the raw partition using standard UNIX commands. This in itself, makes RAW devices a bit scary if you have not dealt with them before. Also, with the advent of ASM, Oracle actually prefers the use of ASM over the direct use of Raw devices

Pros:

1. Raw devices tend to be faster in I/O than with "cooked" files systems, because the writing of data to disk is done by Oracle and skips the Operating systems I/O buffer. Also, typical file system maintenance is not required by the operating system, thereby reducing this overhead.
2. Very good for REDO files, as they are very write intensive.

Cons:

1. The data files in a RAW partition/device cannot be viewed or manipulated through the OS command line. There's no way to get an accurate picture at an OS level of how much disk space is in use – no df, find, ls -l
2. Due to the in ability to view or manipulate the data files, databases on Raw devices can be more difficult to maintain.
3. Oracle treats each raw device or raw partition as one file – can result in many many raw devices required.
4. Hence the configuration of the storage space is inflexible because only one database file is permitted for each raw device (and therefore for each partition). The sizes of the individual partitions must be adjusted to the sizes of the database files. This makes later relocation of the database files to other partitions more difficult.
5. As of Oracle 12c, raw devices are no longer supported by Oracle.

Cooked file systems

A cooked file system, is a regular file system that is managed by the operating system. In layman's terms, it is the file system used when installing your OS and creating partitions in the OS. Nothing other than a standard installation needs to be done the DBA/SA. Any OS operation can be performed in these files systems. You can browse the directories and manipulate the files from the operating system interface. It is very common to use cooked file systems with oracle. More than likely, that is what you currently have your database on.

Pros:

1. The DBA can view and manipulated the database files. This is much easier when needing move data file, or backup and restore data files.
2. This is the default method of installation and management for Oracle.

Cons:

1. I/0 may be slower due to the fact that Oracle must use the operating system disk cache, instead of writing to disk directly. The OS becomes and extra step when writing to disc.

OCFS

Oracle **C**lustered **F**ile **S**ystem, is a file system format that is proprietary to Oracle. This file system is designed with clustering in mind, hence the name. OCFS was designed by oracle to make it possible for DBA's to run a RAC on a shared file system, without having to use RAW devices. There are other clustered file systems on the market, however oracle offers OCFS at no cost. The initial version of OCFS (OCFS 1) targeted making clustered data storage easier to manage. Only database files, such as data files, control files, and redo log files (archived also) can be stored in these files system (in addition to the files the file system keeps there to maintain the shared cluster storage). Oracle has now release OCFS2, which has been expanded to include the storage of Oracle binaries and scripts. OCFS is not included with every operating system. Certain OS's are now including OCFS as an option for file system formatting, but for others the OCFS software will need to be downloaded from Oracle.

Work on a cluster filesystem at Oracle started many years ago, in the early 2000's when the Oracle Database Cluster development team wrote a cluster filesystem for Windows that was primarily focused on providing an alternative to raw disk devices and help customers with the deployment of Oracle Real Application Cluster (RAC). Oracle RAC is a cluster technology that lets us make a cluster of Oracle Database servers look like one big database. The RDBMS runs on many nodes and they all work on the same data. It's a Shared Disk database design. There are many advantages doing this but I will not go into detail as that is not the purpose of my write up. Suffice it to say that Oracle RAC expects all the database data to be visible in a consistent, coherent way, across all the nodes in the cluster. To do that, there were/are a few options : 1) use raw disk devices that are shared, through SCSI, FC, or iSCSI 2) use a network filesystem (NFS) 3) use a cluster filesystem(CFS) which basically gives you a filesystem that's coherent across all nodes using shared disks. It is sort of (but not quite) combining option 1 and 2 except that you don't do network access to the files, the files are effectively locally visible as if it was a local filesystem. So OCFS (Oracle Cluster Filesystem) was born.

The first version of OCFS was really primarily focused on replacing the use of Raw devices with a simple filesystem that lets you create files and provide direct IO to these files to get basically native raw disk performance. Cache coherency was easy since it was basically always direct IO down to the disk device and this ensured that any time one issues a write() command it would go directly down to the disk, and not return until the write() was completed. Same for read() any sort of read from a datafile would be a read() operation that went all the way to disk and return. We did not cache any data when it came down to Oracle data files.

Pros:

1. The file system was designed with Oracle clustering in mind and it is free.
2. Eliminates the need to use RAW devices or other expensive clustered file systems.
3. With the advent of OCFS2, binaries, scripts, and configuration files (shared Oracle home) can be stored in the file system. Making the management of RAC easier.

Cons:

1. With OCFS version 1, regular files cannot be store in the file system, however this issue is eliminated with OCFS2.
2. Adds another product/solution to the database stack.
3. Can add performance overhead
4. Except for Linux, all of the clustered filesystem options are provided by a third-party vendor.

ASM

Automatic Storage Management is a feature of Oracle that allows the Database administrator to manage logical volumes for the database. When setting up ASM, the disk groups that are managed are place on a raw partition. During ASM set up, the installation software will perform a "disk discovery" for raw partitions available. Because ASM does not use the OS file system, you get the same I/O performance benefits as you do with standard RAW devices. In addition to the performance benefits, ASM by design, makes the management of the contents of the raw device easier, by acting as the volume manager (actually it is the volume manager). ASM runs as a separate instance from Oracle (it is recommended to have a separate home, for upgrade and maintenance reasons). Oracle accesses the ASM instance in order to read from the disk. In order for Oracle to see the contents on the RAW partition(s), ASM must be up and running. This differs for standard RAW devices, because ASM is accessing the devices, not the Oracle instance. The ASM instance is what manages the disk groups where the data is stored. Also, ASM for both single instance databases as well as clustered databases and is the preferred storage method for Clustering by Oracle.

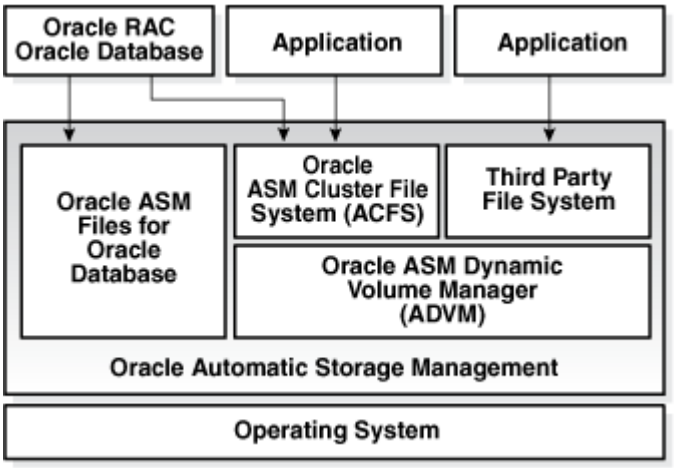
Pros:

1. Same I/O benefits as with RAW devices.
2. Oracle provides an interface to manage disk groups.
3. ASM provides file level striping and mirroring.

Cons:

1. ASM only support data base files. Standard files(binaries,etc) are not allowed.
2. ASM files can only be managed using RMAN. These management tasks include moving files from one disk group to another as well as backup and recovery operations.
3. The files maintained by ASM are not visible to the OS.

ACFS



Oracle in 11gR2 extends the functionality of ASM and introduces a new ASM Cluster File System (ACFS) based on ASM. ACFS is a general purpose cluster file system that allows storing all type of data files in similar manner to any Linux file system for example ext3. Oracle recommends using ASM for storing database files as it was conceptually done since the first introduction of ASM in release10g. ACFS cannot be used to install the Oracle 11gR2 Grid

Infrastructure that is, Oracle 11gR2 also uses ASM to store the Grid Infrastructure OCR and the voting disk.

ACFS can be used as a file system for Oracle homes storing the Oracle binaries.

Oracle 11gR2 new features enable us to create, mount and manage ACFS using familiar Linux commands. ACFS supports snapshots and dynamically online resizing of existing file systems using internally ASM Dynamic Volume Manager (ADVM). ACFS is based on volumes that are created in a disk group. Once created volume is identified by a volume device, that is used by Linux to address the volume and this volume device is used to create the ACFS.

To use and deploy ACFS the compatibility in ASM has to be at least 11.2 for ASM, RDBMS and ADVM.

NFS

Network File Systems are also used by Oracle as a storage options. These file systems are stored on NAS (Network Attached Storage) devices/fliers. Netapp and EMC are providers of NAS products. NFS mounts are seen as remote devices, and not local to the machine as with a SAN. Because NFS is stored on a remote device on the Network, through put may not be as good as with a SAN or local device, so systems with a high number of transactions may not benefit from this. Oracle as a list of certified NFS configurations.

Pros:

1. The cost of NAS products is generally cheaper.

Cons:

1. There is a finite list of certified configurations by Oracle.
2. I/O through put can be slower, and can be effected by additional network traffic if not isolated.

ASM is the most popular choice from Oracle 10g onwards and for RAC as well as non-RAC environments, it's adoption has been really big. Every environment is different though and you would need to weigh the pros and cons of each option.