ORACLE

Oracle ACFS

Advanced Cluster File System

Snapshots Best Practices

November, 2024 | Version 1.04

DISCLAIMER

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

TABLE OF CONTENTS

INTRODUCTION

Oracle Advanced Cluster File System (hereinafter Oracle ACFS), is a file system that fulfills all industry standards, complies with POSIX and X/OPEN, and offers support for multiple Operating Systems such as Oracle Linux, Redhat, Novell SLES, Solaris and AIX. Oracle ACFS is deployed in a wide arrange of server platforms, ranging from traditional server environments, Oracle Engineered Systems such as the Oracle Exadata Database Machine and the Oracle Database Appliance, and it is steadily becoming the storage foundation of choice for diverse Cloud offerings.

Throughout years of constant growth and evolution, Oracle ACFS has broadened its scope as a cluster file system, incorporating diverse features and functionality to its offering and becoming the preferred storage management solution of choice for application files and Oracle Database files. Oracle ACFS allows for file based and filesystem-based snapshots, providing its user with the capability of provisioning test and development environments in a simple and efficient way.

The following technical brief is an introduction to Oracle ACFS Snapshot functionality and a walkthrough of best practices to leverage ACFS snapshot technologies for your working environments.

WHAT IS ORACLE ACFS

Architecture

Oracle ACFS, as part of Oracle Grid Infrastructure, is integrated with Oracle ASM, Oracle ADVM and Oracle Clusterware as shown in diagram 1 below. ACFS communicates with Oracle ASM to obtain ASM disk group storage addresses and uses these addresses to read and write ACFS data directly onto ASM storage for maximum performance. It also communicates with Oracle Clusterware to facilitate resource management for ACFS.

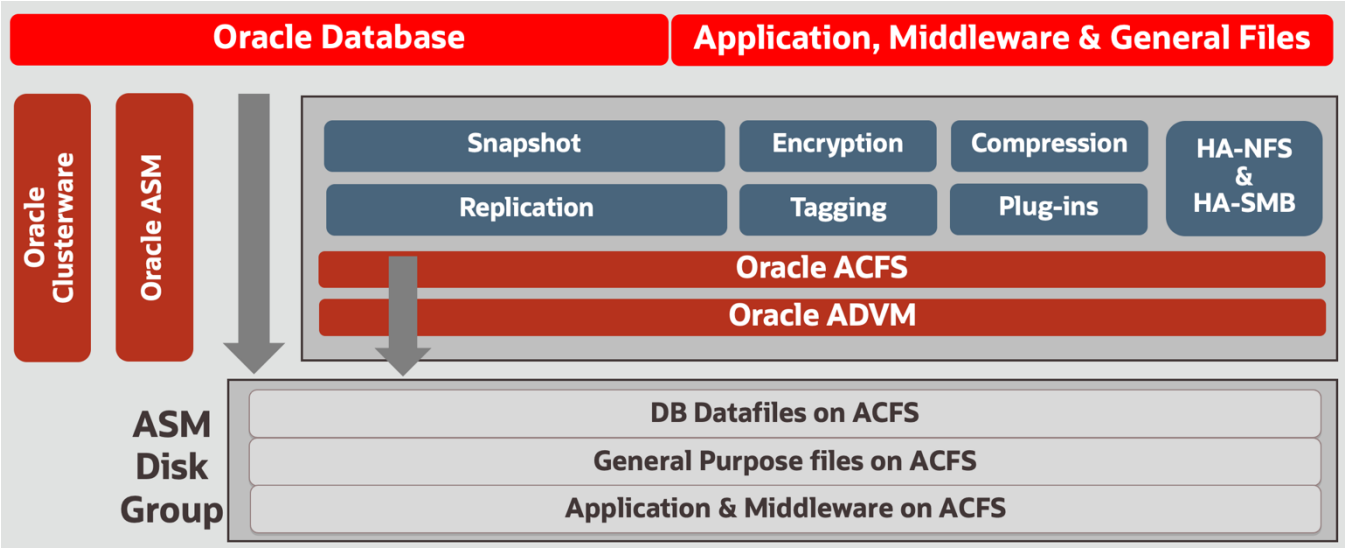

Diagram 1 – Oracle ACFS Architecture

Upon creation of an Oracle ADVM volume, a block device special file is created providing ACFS with a standard device interface for direct access to disk group devices. All ACFS I/O requests are transferred directly to ASM disk group storage as shown in the diagram. ACFS IO does not pass through the Oracle ASM instance.

ASM integration allows ACFS to participate in ASM storage management operations including balanced distribution of ASM disk group file extents, dynamic file resizing, and on-line add/remove of disk group storage devices. ACFS also benefits from ASM file extent striping and mirroring for performance and highly available storage access.

Grid Infrastructure integration allows for Oracle ACFS to leverage Clusterware resources like cluster membership state transitions, driver loading, automatic mounts and unmounts of file systems, and enabling and disabling of volumes; all these provide High Availability for both Oracle ACFS and Oracle ADVM resources.

Oracle ACFS is a complete storage solution that eliminates the need for expensive third-party solutions, bringing together high availability and best in class features to manage your storage needs. Oracle ACFS offers a complete cluster file system with the best performance, simplicity in its management, and high availability as part of the Grid Infrastructure stack. For more information, please review the Oracle ACFS Introduction technical brief at ACFS product page[1].

ORACLE ACFS SNAPSHOTS

Oracle ACFS snapshotting functionality provides the ability to generate sparse, point in time read-write or read-only copies of Oracle ACFS file systems or single files. Oracle ACFS snapshots use space-efficient copy-on-write functionality. In order to maintain a point-in-time view of the file system, Oracle ACFS file extent value updates are allocated new storage while snapshots maintain pointers to existing extent values.

1 https://www.oracle.com/goto/acfs

Oracle ACFS Snapshots are stored within the originating ACFS file system. If additional storage for files and snapshots is needed, Oracle ACFS allows for dynamic resizing of the file system. Oracle ACFS snapshots are ready and available to be used upon creation as long as the file system is mounted. Up to 1023 snapshots can be hosted on an Oracle ACFS file system. Oracle ACFS supports creation of snapshots of snapshots, providing full inheritance at any given point in the hierarchy. There is no performance penalty related to the number of snapshots or the depth of the snapshot hierarchy. There is no performance penalty for deleting snapshots from within the snapshot hierarchy.

As shown below, the process is straightforward and simple for users to leverage the functionality of Oracle ACFS Snapshot capabilities. Additional snapshot commands will also be briefly explained:

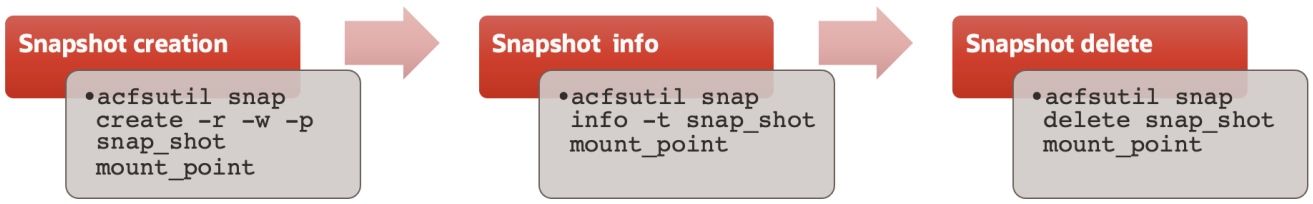

Diagram 2 – Oracle Snapshots

`acfsutil snap create -r -w -p parent_snapshot snap_shot mount_point` command creates a snapshot following the user specified options which are:

- `[-r|-w]` snapshots are by default read-only, if `-w` option is selected the snapshot will be read-write. A snapshot can be later converted from read-write to read-only or vice versa with the `acfsutil snap convert` command.
- `snap_shot` indicates the snapshot name. A snapshot name can be changed via the `acfsutil snap rename` operation.
- `-p parent_snap_shot` when performing a snapshot of a snapshot, the parent snapshot named must be specified via this option.
- `mount_point` this option specifies the file system mount point.
- a quota for a snapshot can be established through the `acfsutil snap quota` command.

`acfsutil snap info -t snap_shot mount_point` command provides information regarding a specific snapshot (specified by `snap_shot`) on the file system mounted on the specified mount point (`mount_point`). Option `-t` indicates that the output should contain a tree structure of the snapshot.

`acfsutil snap delete snap_shot mount_point` command deletes the specified snapshot (`snap_shot`) on the file system mounted on the specified mount point (`mount_point`).

For individual file snapshots, the following command utility is used:

`acfsutil fshare create source_file_path destination_file_path`

- `fshare` specifies that the snapshot to be created is for a single file (fshares are automatically created when the Linux reflink option to the cp command is used).
- `source_file_path` specifies the file to be used as a source for the snapshot.
- `destination_file_path` specifies the path where the snapshot is to be created.

Oracle ACFS introduced in 12c Release 2 the snap duplicate create and snap duplicate apply operations. This allows users to create a duplicate snapshot and a duplication stream to apply any changes in the original snapshot to the duplicate snapshot. Furthermore, users can create new Oracle ACFS file systems out of a snapshot and apply changes in the source snapshot to the newly created file system. Please review the Oracle Automatic Storage Management Administrator's guide for more information on snap duplicate create and apply operations.

Further snapshot enhancements have been introduced including snapshot links (18c) and remastering of a file system based on an existing snapshot (12cR2). An Oracle ACFS snapshot link presents an alternate path to a snapshot and its contents. Finally, remastering of an Oracle ACFS file system allows for an existing snapshot to be used as a base for a file system: all existing contents of the Oracle ACFS file system will be removed and replaced with the contents of the selected snapshot. Please see below, a basic explanation on how to create snapshot links and file system remastering:

Diagram 3 – Oracle Snapshots

`acfsutil snap link -s snap_shot -d path_to_link` command creates a snapshot link based on the specified snapshot and the provided link name. If used, the d option, deletes the specified snapshot link.

`acfsutil snap remaster -c -f snapshot volume` command remasters an Oracle ACFS file system, using as a base the specified snapshot in the specified volume. In case of any interruption, the remastering of the file system will continue by executing the command again with the `-c` option. Option `-f` will force the operation.

Oracle ACFS in conjunction with Oracle Multitenant, allows customers to leverage snapshots with pluggable database technology. Using copy on write technology, Oracle ACFS allows for the creation of snapshot clones of pluggable databases, further broadening the customer's choices for the provisioning of test and development environments. Customers can use the pluggable database clones for testing of new applications, run all their test scenarios, and more without ever jeopardizing production data. Databases stored on Oracle ACFS require just a few steps in order to leverage this functionality. This functionality allows for SQLplus to work in conjunction with ACFS, allowing customers to create PDB snapshot clones internally from the SQLplus terminal without having to switch to acfsutil commands. More information can be found on Oracle ACFS Administrators Guide[2].

Oracle ACFS Snapshot-based Replication

Oracle ACFS provides maximum efficiency and flexibility in asynchronous replication for your file system and Oracle Database files. Replication was first introduced in Oracle ACFS on release 11.2.0.2. This feature has seen constant evolution from log-based replication for application files only, to the current snapshot-based replication solution that supports entire file systems and application files. Oracle ACFS Replication provides a solution for replicating an Oracle ACFS file system across the network from a primary to a standby site. Complemented with Oracle Data Guard or Oracle GoldenGate, Oracle ACFS Snapshot-based Replication provides the customer with an end-to-end DR solution for all files.

Oracle ACFS Snapshot-based replication works by transferring the differences between consecutive snapshots from the primary file system to the standby file system using the simple ssh protocol. Once an initial snapshot is replicated from the active to the standby file system, the changes of the successive snapshots will continue to be replicated. Oracle ACFS Snapshot-based replication carries a timestamp that can be queried for the purpose of synchronizing database transactions with ACFS file system data.

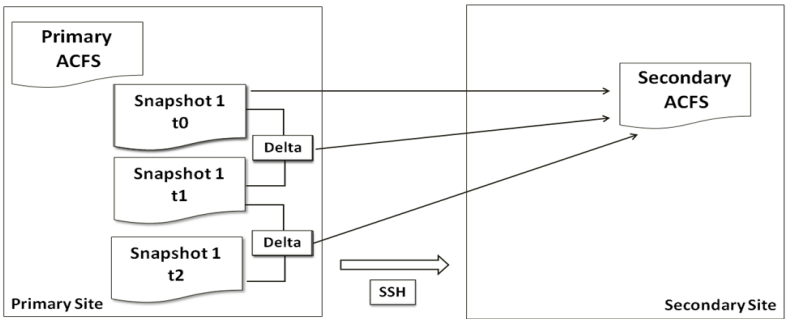

Diagram 4 – Oracle ACFS Snapshot-based Replication

[2] Oracle ACFS Administrator's guide https://docs.oracle.com/en/database/oracle/oracle-database/19/ostmg/acfs-advanced-topics.html#GUID-8D71BD92-78F9-40DF-8418-A000E201218B

Oracle ACFS Replication on primary sites running AIX, Linux, or Solaris works with standby sites running on any of the mentioned operating systems. Oracle ACFS Replication allows for encryption to be enabled, thus securing the replicated standby file system with any policies in place in the primary file system. Role reversal replication is introduced in Oracle ACFS 18c, allowing the original primary and standby locations to switch roles. One command enables users to change the original primary location to become the new standby and the original standby location to become the new primary. For more information on Oracle ACFS snapshot-based replication, please review the Oracle ACFS replication technical brief. [3]

Oracle ACFS Snapshots Best Practices

Oracle ACFS Snapshot Metadata Storage Requirements Considerations

Oracle ACFS File systems utilize metadata files for management purposes. One of the metadata files used is called the File Entry Table. This hidden file keeps the information about all of the files in the Oracle ACFS file system. A non-shared copy of the file system's File Entry Table is created upon Snapshot creation.

What storage considerations should users take when creating snapshots in an Oracle ACFS file system? For metadata purposes related to the File Entry Table file, let's use the following example to illustrate Oracle's best practice regarding snapshot metadata storage requirements:

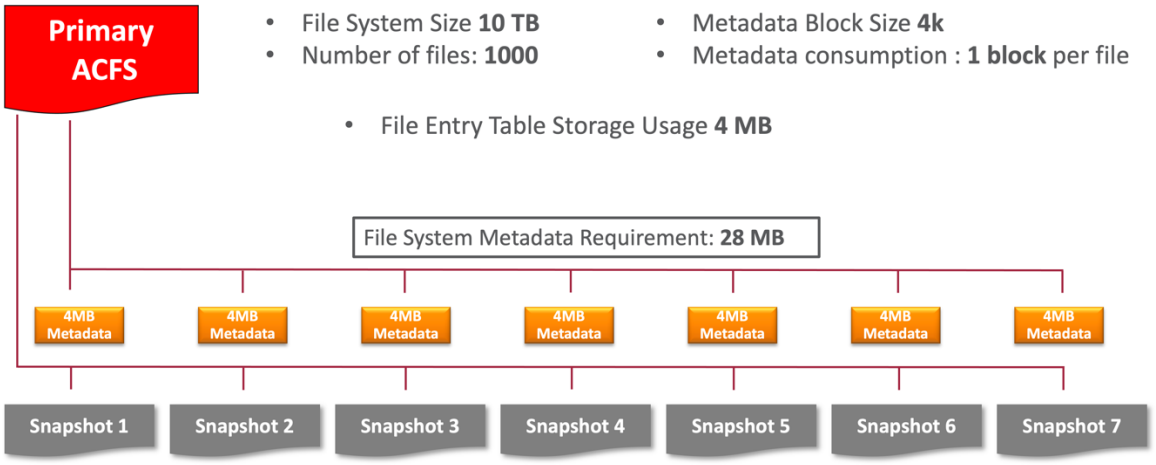

Diagram 5 – Oracle ACFS Snapshots Metadata Storage Requirements

The file system has seven snapshots maintained to keep weekly data in this example. Given the metadata block size, the consumption per file, and the File Entry Table storage usage, it is then inferred that a minimum of 28Mb of storage must be available at all times for metadata purposes.

Oracle ACFS Snapshot Storage Requirement Considerations

Oracle ACFS file systems require storage availability for snapshot creation and maintenance. How much available storage is required? Let's use the following example to illustrate Oracle's best practice regarding snapshot metadata storage requirements.

An Oracle ACFS File system is configured to maintain weekly data with seven snapshots. A snapshot is created daily while the oldest snapshot is deleted, thus, preserving the last seven days of data always available. It added daily storage to the file system averages 480 Mb, which will equate to a daily growth of the snapshots of the same number, 480 Mb. Taking into consideration storage for seven snapshots (3360 Mb) and metadata storage (28 Mb) to be set aside (as seen in the example above), the required total snapshot storage is 3388 Mb.

3 ACFS File System Snapshot-Based Replication: How to Setup Guide https://www.oracle.com/a/tech/docs/acfs-file-system-snapshot-based-replication.pdf

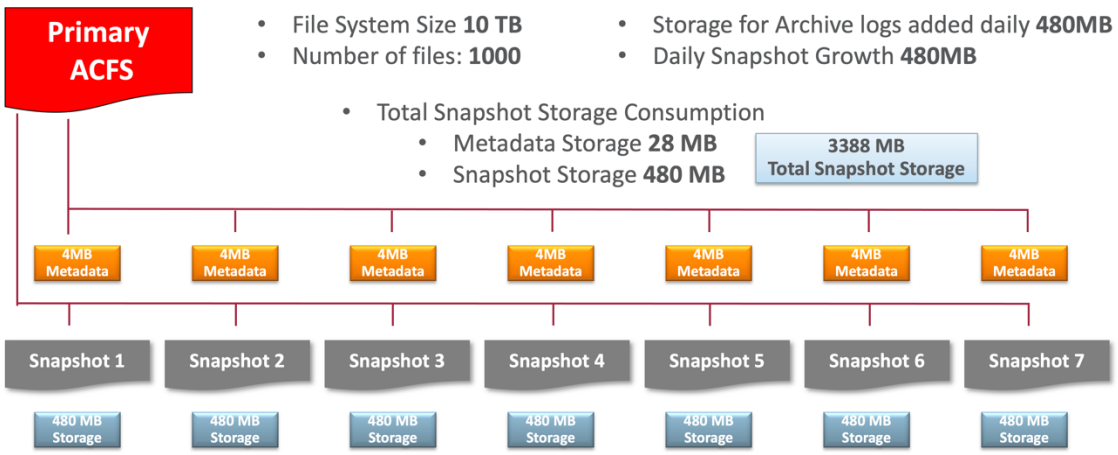

Diagram 6 – Oracle ACFS Snapshots Storage Requirements

Oracle ACFS Snapshot Storage Requirement Considerations for Replication

Oracle ACFS Snapshot-based replication starts up its process by creating a snapshot of the primary file system on the active site. This first snapshot is sent to the standby cluster and then applied to the standby file system; while this takes place, it is expected that this initial snapshot will use an increasing amount of storage while it is copied to the standby cluster file system. Tools such as 'df' can be used on the standby file system to gauge the progress of the copy and to project how much storage the primary snapshot will consume before the copy completes. Let's use the following example to illustrate Oracle's best practice regarding snapshot storage requirements for ACFS replication.

An Oracle ACFS file system is initiating replication with the following values, 10 Terabytes in initial size, 1000 files, and 480 Mb of archive logs added daily (which equates to 20Mb an hour). Let's assume that it takes 12 hours to send and copy the initial snapshot to the standby file system. Using the example above, we can assume that the storage requirement for metadata is 4Mb, and in the case of snapshot storage, the requirement is 240Mb; this since if the transport is done at an average of 20Mb an hour and the total time is estimated to be 12 hours, then 240Mb = 20Mb * 12. The complete storage requirements then for the initial snapshot is 244 Mb.

- File System Size **10 TB**
- Number of files: **1000**
- Storage for Archive logs added daily **480MB** or **20MB/hour**

- Metadata Storage for Snapshot 1 on Standby **4MB**
- Snapshot 1 Storage on Standby **1** snapshot * **20 MB** * **12** = **240 MB**
- Total Snapshot Storage **244 MB**

Diagram 7 – Oracle ACFS Snapshots Storage Requirements for Replication

Once the initial snapshot has been transported and applied from the active site's primary file system to the standby site's secondary file system, new snapshots will be created on the primary file system. The timing of snapshot creation can be configured to happen every specific time unit. In this case, the snapshots are made every hour; Oracle ACFS replication will verify the differences between each subsequent snapshot (delta) and then send over the network only the differences to be applied to the standby site's secondary file system. Upon successfully applying the delta on the standby site's secondary file system, Oracle ACFS replication deletes the older snapshot on the active site's primary file system; this guarantees that only two snapshots will require storage at any given time.

Under the premise explained above, the total storage consumed by replication snapshots after initialization should be 8 Mb of metadata storage and 47 Mb of Snapshot storage. Since the file system is dealing with two snapshots, this would mean 20 Mb per snapshot for the hour in between every snapshot creation and 20Mb/6 (3.33 Mb) for each snapshot since it would take ten minutes to create and send the deltas and there are six ten minute blocks in an hour. The total would be 8Mb of metadata storage and 47 Mb of Snapshot storage, hence 55 Mb.

- Metadata Storage for snapshots on Active **8MB**
- Snapshots Storage for snapshots on Active
  2 snapshot * **(20 MB + (20MB/6)) = 47 MB**

- Total Snapshot Storage **55 MB**

Diagram 8 – Oracle ACFS Snapshots Storage Requirements for Replication

Oracle ACFS Snapshot Deletion Storage Considerations

Oracle ACFS snapshots utilize two types of storage, inherited storage and allocated storage. Inherited storage is storage the snapshot is utilizing, and it was inherited from its parent; snapshots can also have sibling snapshots that may also be using this storage. Sibling snapshots are snapshots that share the same parent. Allocated storage was added to the snapshot because files were created within the snapshot, or existing files in the snapshot were overwritten or extended; this storage is only used by the snapshot that allocated it and any children snapshots it may have (not used by sibling snapshots).

Upon ACFS snapshot deletion, storage will be freed. However, there are conditions regarding which storage will be freed:

- Inherited storage that is not being shared with sibling snapshots.
- Allocated storage that is not being used by the snapshot's children, if any.
- It is important to remember that these allocations only occur with Read/Write Snapshots. Inheritance, however, does occur with both Read-only and Read/Write Snapshots.

The following examples illustrate in a simple graphical fashion.

Example 1

Let's start with a file system of 10 Terabytes, 1000 files, and 480Mb of storage archive logs added daily. Every day at the same time, a snapshot is created. On Monday, "Snapshot Monday" is made, and 480Mb of new archive logs storage is added to the file system throughout the day. Remember that "Snapshot Monday" does not use this storage as it was added to the file system after the snapshot was created.

The exact process happens on Tuesday; a snapshot is created, "Snapshot Tuesday," which shares the 480 Mb of new archive log storage added on Monday with the file system. 480Mb of new archive logs storage throughout the day is added to the file system. Remember that "Snapshot Tuesday" does not use this storage as it was added to the file system after the snapshot was created.

The exact process happens on Wednesday; a snapshot is created, "Snapshot Wednesday," which shares with the file system the 480 Mb of archive log storage added on Tuesday and shares with "Snapshot Tuesday" the 480 Mb of archive logs added on Monday. 480Mb of new archive logs storage throughout the day is added to the file system. Remember that "Snapshot Wednesday" does not use this storage as it was added to the file system after the snapshot was created.

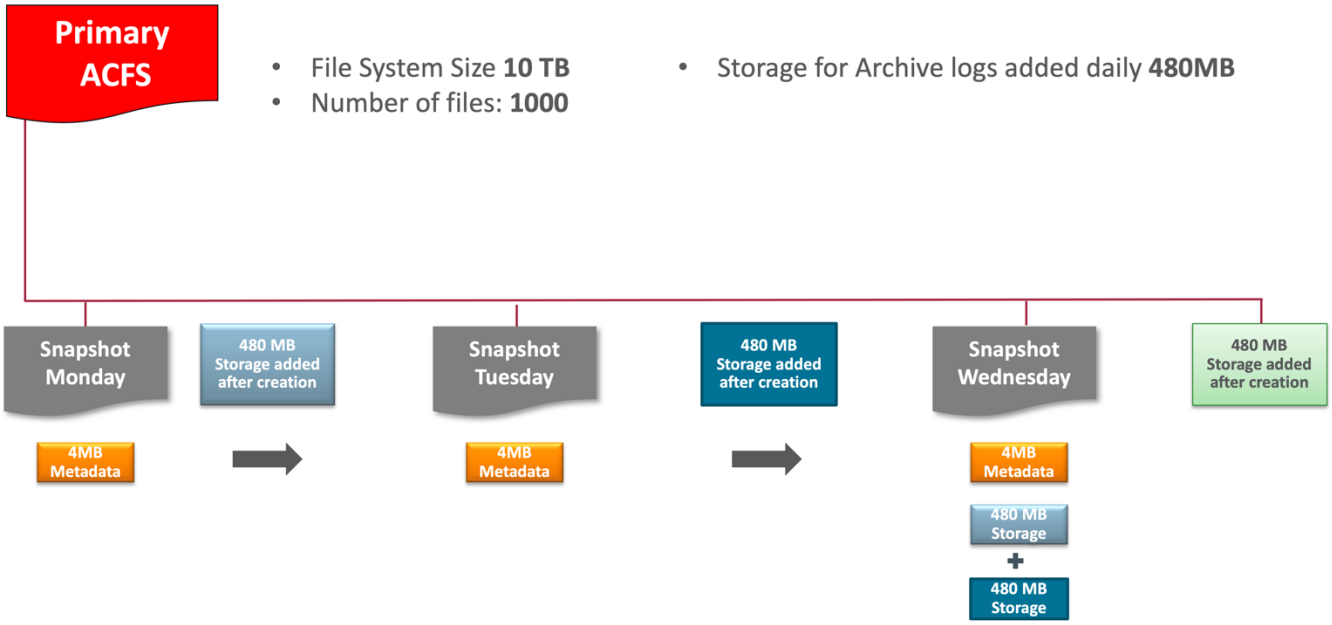

Diagram 9 – Oracle ACFS Snapshots Deletion Storage Considerations

How would storage behave when files are deleted in this scenario? Let's start by deleting from the ACFS file system the archive logs generated on Monday and Tuesday.

- "Snapshot Tuesday" and "Snapshot Wednesday" are still using the archive logs from Monday, so that they will inherit this specific storage.
- "Snapshot Wednesday" is still using the archive logs from Tuesday so that it will inherit this specific storage.

If "Snapshot Monday" were to be deleted, a minimal amount of storage would be freed since most of the hold the snapshot is using is being shared with the ACFS file system. The same thing applies if "Snapshot Tuesday" is deleted since it shares most of its storage with the ACFS file system and with "Snapshot Wednesday." However, if "Snapshot Wednesday" were to be deleted, 480Mb+ of storage would be freed as only this snapshot is using the storage for Tuesday's archive logs which were deleted manually.

Example 2

Let's continue with the same file system from Example 1. On Monday, "Snapshot Monday" is created, and throughout the day, 480Mb of new archive logs storage is added to the file system. Remember that "Snapshot Monday" does not use this storage as it was added to the file system after the snapshot was created. Now, new data is added to "Snapshot Monday" by creating files in that snapshot, and 100 Mb of new data is added.

The exact process happens on Tuesday; a snapshot is created, "Snapshot Tuesday," which shares the 480 Mb of new archive log storage added on Monday with the file system. 480Mb of new archive logs storage throughout the day is added to the file system. Remember that "Snapshot Tuesday" does not use this storage as it was added to the file system after the snapshot was created. Now, new data is added to "Snapshot Tuesday" by creating files in that snapshot, and 600 Mb of new data is added.

The exact process happens on Wednesday; a snapshot is created, "Snapshot Wednesday," which shares with the file system the 480 Mb of archive log storage added on Tuesday and shares with "Snapshot Tuesday" the 480 Mb of archive logs added on Monday. 480Mb of new archive logs storage throughout the day is added to the file system. Remember that "Snapshot Wednesday" does not use this storage as it was added to the file system after the snapshot was created. Now, new data is added to "Snapshot Wednesday" by creating files in that snapshot, and 300 Mb of new data is added.

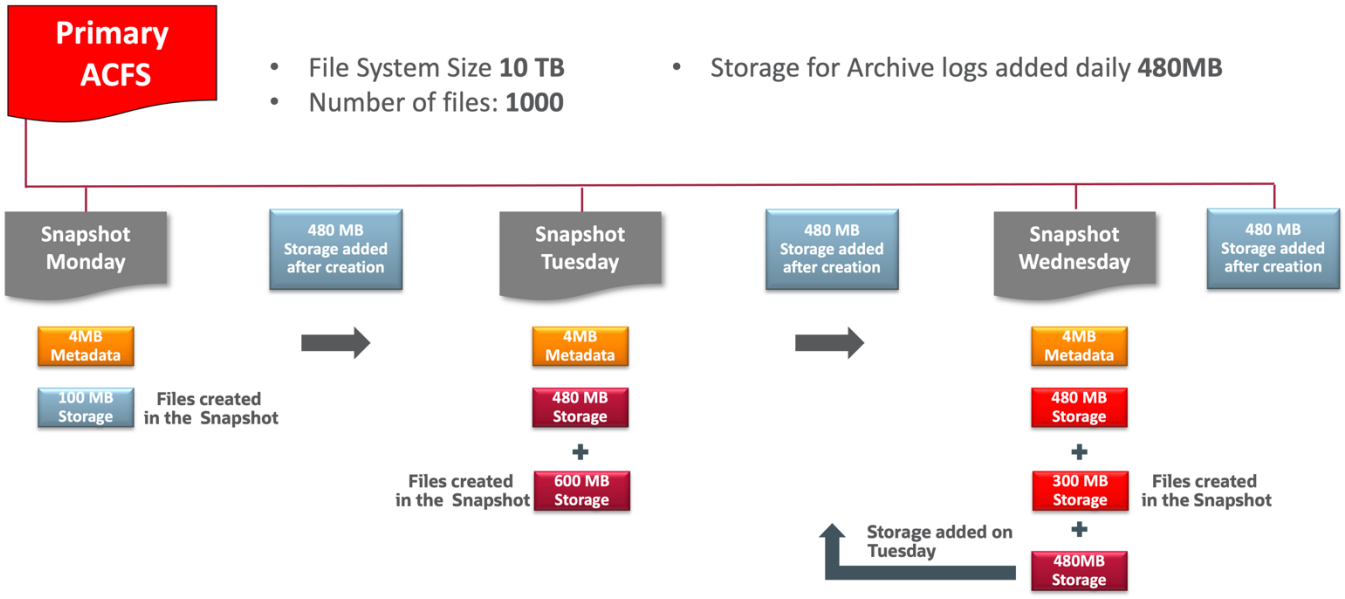

Diagram 10 – Oracle ACFS Snapshots Deletion Storage Considerations

How would storage behave when files are deleted in this scenario? Let's start by deleting from the ACFS file system the archive logs generated on Monday and Tuesday.

- "Snapshot Tuesday" and "Snapshot Wednesday" are still using the archive logs from Monday, so that they will inherit this specific storage.
- "Snapshot Wednesday" is still using the archive logs from Tuesday so that it will inherit this specific storage.

If "Snapshot Monday" were to be deleted, no inherited storage would be freed since most of them hold the snapshot shared with the ACFS file system. However, the 100 Mb of new data added only to this snapshot would be freed.

The same applies in case "Snapshot Tuesday" were to be deleted since it shares most of its storage with the ACFS file system and with "Snapshot Wednesday"; however, the 600 Mb of new data was added only to this snapshot would be freed.

Finally, if "Snapshot Wednesday" were to be deleted, 480Mb+ of storage would be freed as only this snapshot is using the storage for Tuesday's archive logs which were deleted manually; in addition, the 300 Mb of new data added to this snapshot would be freed as well.

Tips and Tricks

The *acfsutil info fs* command can be used to determine how much storage is being used by the File Entry Table in the file system:

```
# acfsutil info fs /usmmnt
/usmmnt
    ACFS Version: 21.0.0.0.0
    on-disk version:       51.0
    ACFS compatibility:    21.0.0.0.0
    flags:        MountPoint,Available,KiloSnap
    creation time:   Tue Sep 08 06:58:20 2020
    mount time:      Tue Sep 08 06:58:22 2020
    mount sequence number: 12
    number of nodes:       2
    allocation unit:       4096
    metadata block size:   4096
    volumes:      1
    total size:   128428539904  ( 119.61 GB )
    total free:   124618858496  ( 116.06 GB ) ( 97.03% free )
    file entry table allocation: 75890688   (  72.38 MB ) ( 18528 entries )
    number of volume logs: 2
    primary volume: /dev/dsk/c1d1s6
        label:
        state:                 Available
        major, minor:          240, 14
        logical sector size:   512
        size:                  128428539904  ( 119.61 GB )
        free:                  124618858496  ( 116.06 GB ) ( 97.03% free )
        metadata read I/O count:         24241
        metadata write I/O count:        2375
        total metadata bytes read:       250458112  ( 238.86 MB )
        total metadata bytes written:    355131392  ( 338.68 MB )
    number of snapshots:  2
    snapshot space usage: 151781376  ( 144.75 MB )
    replication status: Disabled
    compression status: Disabled
```

The *acfsutil snap info* command can be used to determine how much storage is being used by the file entry table in each snapshot and also how much storage has been added to that snapshot (new files, extended files, etc.):

```
# acfsutil snap info /usmmnt
snapshot name:               s1
snapshot location:           /usmmnt/.ACFS/snaps/s1
RO snapshot or RW snapshot:  RW
parent name:                 /usmmnt
snapshot creation time:      Tue Sep 08 07:08:54 2020
file entry table allocation: 84279296   (  80.38 MB ) ( 20576 entries )
storage added to snapshot:   224501760   ( 214.10 MB )

snapshot name:               s2
snapshot location:           /usmmnt/.ACFS/snaps/s2
RO snapshot or RW snapshot:  RW
parent name:                 /usmmnt
snapshot creation time:      Tue Sep 08 07:25:12 2020
file entry table allocation: 92667904   (  88.38 MB ) ( 22624 entries )
storage added to snapshot:   728162304   ( 694.43 MB )

    number of snapshots:  2
    kilosnap state:       Enabled
    snapshot space usage: 952664064  ( 908.53 MB )
```

Besides the file entry table, other metadata may consume storage within snapshots. For huge files, index metadata blocks manage file extents. The amount of storage used for this is not explicitly listed in the *acfsutil snap info* output but is accounted for in the *snapshot space usage* number.

If ACFS snapshots consume more space than anticipated, the file system can be grown using the *acfsutil size* command. The following command will grow the file system and the underlying ADVM volume by 1TB, assuming that the underlying disk group has enough storage:

```
# acfsutil size +1TB /usmmnt
```

The file system can also be grown to a particular size, such as 10TB:

```
# acfsutil size 10TB /usmmnt
```

Alternatively, auto-resize can be enabled to grow the file system automatically when it begins to become full. The following command will grow the file system automatically by 1TB up to a maximum of 32TB when free space in the file system falls below 1TB or 10% of the size of the file system:

```
# acfsutil size –a 1TB –x 32TB /usmmnt
```

Troubleshooting

One of the most common sources of confusion related to ACFS snapshots is the apparent discrepancy between the individual snapshot storage usage and the total snapshot storage usage reported by *acfsutil snap info* and *acfsutil info fs*. Consider the following output from *acfsutil snap info /usmmnt*:

```
# /sbin/acfsutil snap info /usmmnt
snapshot name:               s1
snapshot location:           /usmmnt/.ACFS/snaps/s1
RO snapshot or RW snapshot:  RW
storage interest tracking:   Enabled
parent name:                 /usmmnt
snapshot creation time:      Wed Sep  9 12:22:14 2020
file entry table allocation: 262144   ( 256.00 KB ) ( 64 entries )
storage added to snapshot:   537133056   (  512.25 MB )


snapshot name:               s2
snapshot location:           /usmmnt/.ACFS/snaps/s2
RO snapshot or RW snapshot:  RW
storage interest tracking:   Enabled
parent name:                 /usmmnt
snapshot creation time:      Wed Sep  9 12:22:14 2020
file entry table allocation: 262144   ( 256.00 KB ) ( 64 entries )
storage added to snapshot:   262144   ( 256.00 KB )



snapshot name:               s3
snapshot location:           /usmmnt/.ACFS/snaps/s3
RO snapshot or RW snapshot:  RW
storage interest tracking:   Enabled
parent name:                 /usmmnt
snapshot creation time:      Wed Sep  9 12:22:14 2020
file entry table allocation: 262144   ( 256.00 KB ) ( 64 entries )
storage added to snapshot:   262144   ( 256.00 KB )


    number of snapshots:  3
    kilosnap state:       Enabled
    storage interest tracking: Enabled
    snapshot space usage: 1611399168  (   1.50 GB )
```

This output shows that the file system has three snapshots. The last line states that the total snapshot space usage is 1.50 GB. But at the end of each snapshot display, the storage added to the snapshot is also displayed. These are 512.25 MB, 256 KB, and 256 KB. If these are added together, they total only slightly less than 513 MB and much less than the 1.50 GB reported at the end of the output.

This apparent discrepancy is because the individual snapshot storage output is reporting the amount of storage added to the snapshot using file creations or writes to files in that snapshot. It also includes the storage used by the File Entry Table for the snapshot. By contrast, the final line of output, snapshot space usage, is also reporting this, but, in addition, it includes the amount of storage that the snapshots, as a group, have inherited from the file system. The above output was generated after executing the following :

1. `mkfs...`
2. `mount...`
3. Create three 1G files in the file system.
4. `/sbin/acfsutil snap create -w s1 /usmmnt`
5. `/sbin/acfsutil snap create -w s2 /usmmnt`
6. `/sbin/acfsutil snap create -w s3 /usmmnt`
7. Overwrite the first 512 MB of one of the three files in snapshot s1. This causes s1 to get 512 MB of new storage for the new data. This is why its display shows 512.25 MB of storage added to it.
8. Overwrite the first 512 MB of two of the three 1G files in the file system. This causes the file system to give the old storage for the first half of those two 1G files to the snapshots to share while it allocates 512 MB of new storage for each of the two files being overwritten. This is known as copy-on-write.

The output at the end of the display, then, is the sum of the File Entry Tables in the snapshots (256KB * 3), from steps 4-6 above, the storage added to the snapshots by writes to their files(512 MB), from step 7 above, and the storage inherited by the snapshots from the file system when the files in the file system were partially overwritten, (1 GB), from step 8 above.

Note that *`acfsutil info fs`* also reports the same total snapshot space usage as part of its output:

```
# /sbin/acfsutil info fs /usmmnt
/usmmnt
    ACFS Version: 21.0.0.0.0
    on-disk version:       51.0
    ACFS compatibility:    21.0.0.0.0
    flags:        MountPoint,Available,KiloSnap,GBMChksum
    creation time:   Thu Sep  10 04:15:18 2020
    mount time:      Thu Sep  10 04:15:19 2020
    mount sequence number: 132
    number of nodes:       1
    allocation unit:       4096
    metadata block size:   4096
    volumes:      1
    total size:   53687091200  (  50.00 GB )
    total free:   48251514880  (  44.94 GB ) ( 89.88% free )
    file entry table allocation: 262144   ( 256.00 KB ) ( 64 entries )
    number of volume logs: 1
    primary volume: /dev/xvdb
        label:
        state:                 Available
        major, minor:          202, 16
        logical sector size:   512
        size:                  53687091200  (  50.00 GB )
        free:                  48251514880  (  44.94 GB ) ( 89.88% free )
        metadata read I/O count:         5286
        metadata write I/O count:        376
        total metadata bytes read:       43601920  (  41.58 MB )
        total metadata bytes written:    41525248  (  39.60 MB )
    number of snapshots:  3
    snapshot space usage: 1611399168  (   1.50 GB )
    replication status: Disabled
    compression status: Disabled
    storage interest tracking: Enabled
```

CONCLUSION

Oracle Advanced Cluster File System offers a complete, stable, and mature storage solution that has evolved through the years to become the storage foundation for cloud architectures worldwide.

Oracle ACFS offers a universal way to manage your data, providing high availability, high performance, scalability, simplicity in its administration, data integrity, fast recovery, and savings in licensing costs. Customers use Oracle ACFS in over 60 countries, and its features and functionalities are leveraged across all industries, including 40 of the top S&P 500 companies.

Throughout the pages of this technical brief, customers have learned in-depth about ACFS snapshot's functionality, best practices, troubleshooting and tips and tricks to further their usage.

CONNECT WITH US

Call +1.800.ORACLE1 or visit oracle.com.
Outside North America, find your local office at oracle.com/contact.

 blogs.oracle.com

 facebook.com/oracle

 twitter.com/oracle

Oracle ACFS 23ai – Snapshots Best Practices
November, 2024
Author: Tim Mark
Contributing Authors: Ricardo Gonzalez – ACFS Product Management | Oracle ACFS Development team