Mastering Essential ASM Operations

Using Commands,SQL and PL/SQL Scripts

Asfaw Gedamu H

20/03/2024

Introduction

This document outlines the key steps for managing Automatic Storage Management (ASM) in Oracle Database. It covers querying ASM disk information, obtaining details about disk groups, dropping disks, monitoring rebalancing operations, and executing the runcluvfy.sh script for pre-RAC checks. It also enhances ASM and RAC administration by providing scripts for creating ASM disks, stopping/starting RAC clusters, creating password files, and modifying ASM rebalance power.

We delve into the heart of Oracle ASM—unraveling its intricacies, wielding its tools, and safeguarding our data. From disk management to synchronization, from passwords to rebalancing, this title encompasses the vital tasks that define our ASM odyssey. So, buckle up, fellow administrators, as we navigate the realms of storage, resilience, and precision.

Prerequisites

- Access to the Oracle database server with appropriate privileges (SYSDBA or ASM administration role)
- Familiarity with SQL, PL/SQL, and Bash scripting (optional, but beneficial)

Ideal Audience

- Database Administrators (DBAs)
- System Administrators supporting Oracle databases

1. What is ASM?

ASM is a storage management technology in Oracle Database that simplifies storage administration. It provides features like automatic disk group creation, failure handling, and rebalancing.

2. Getting ASM Disk Information

SQL Script

```
set pagesize 2000
set lines 2000
set long 999
col path for a54
select name, path, header_status, total_mb free_mb,
trunc(bytes_read/1024/1024) read_mb,
trunc(bytes_written/1024/1024) write_mb from v$asm_disk;
```

PL/SQL Script

```
DECLARE
  CURSOR c_disks IS
    SELECT disk_name, STATE, TYPE
    FROM dba_disks;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Disk Name' || CHR(9) || 'State' ||
CHR(9) || 'Type');
  FOR rec IN c_disks LOOP
    DBMS_OUTPUT.PUT_LINE(rec.disk_name || CHR(9) || rec.STATE ||
CHR(9) || rec.TYPE);
  END LOOP;
END;
/
```

How to do it?

1. Connect to the database server with SYSDBA or ASM administration privileges.
2. Execute the script:

```
@get_asm_disk_info.sql
```

3. The script outputs the disk name, state (ONLINE, OFFLINE, etc.), and type (NORMAL, REDUNDANCY, etc.).

3. Exploring ASM Diskgroup Details

SQL Script

```
SELECT group_name, STATE, TOTAL_MB, FREE_MB, USABLE_MB,
FREE_MB/TOTAL_MB*100 AS PERCENTAGE
FROM v$asm_diskgroup;
```

How to do it?

1. Connect to the database server.
2. Execute the script.
3. The script displays the disk group name, state, total megabytes (MB), free MB, and usable MB for each disk group.

4. Drop an ASM Disk

PL/SQL Script (Caution: Use with care!)

```
DECLARE
  disk_name VARCHAR2(255);
BEGIN
  -- Replace 'ASM_DISK_TO_DROP' with the actual disk name
  disk_name := 'ASM_DISK_TO_DROP';

  DBMS_OUTPUT.PUT_LINE('Dropping disk ' || disk_name || ' (This
action is irreversible)');
  DBMS_OUTPUT.PUT_LINE('Are you sure? (y/n)');
  IF UPPER(INPUT('Enter your choice: ')) = 'Y' THEN
    EXECUTE IMMEDIATE 'ALTER DISKGROUP ALL DROP DISK ' ||
disk_name;
    DBMS_OUTPUT.PUT_LINE('Disk dropped successfully.');
  ELSE
    DBMS_OUTPUT.PUT_LINE('Drop operation cancelled.');
  END IF;
END;
/
```

SQL Script

```
-----Dropping one disk:
alter diskgroup data drop disk DATA_ASM0001;
-----Dropping multiple disk:
alter diskgroup data drop disk DATA_ASM0001,DATA_ASM00002,
DATA_ASM0003 rebalance power 100;
---- Monitoring the rebalance operation:
select * from v$asm_operation;
```

How to do it?

1. **Important:** This script permanently removes an ASM disk. Use it with extreme caution and only after verifying all backups and consequences.
2. Edit the script to replace 'ASM_DISK_TO_DROP' with the actual disk name.
3. Execute the script.
4. Confirm the drop operation by entering y (yes) when prompted.

5. Monitor ASM Disk Rebalance

SQL Script

```
SELECT START_TIME, PERCENT_COMPLETE, OPERATION
FROM v$asm_operation
WHERE OPERATION LIKE 'rebalance%'
ORDER BY START_TIME DESC;
```

Alternatively:

```
set pagesize 299
set lines 2999
select GROUP_NUMBER,OPERATION,STATE,POWER,
ACTUAL,ACTUAL,EST_MINUTES from gv$asm_operation;
```

How to do it?

1. Connect to the database server.
2. Execute the script.
3. The script shows the start time, percentage completion, and operation type (e.g., rebalance group) for ongoing or recent rebalance operations.

6. Execute runcluvfy.sh for RAC Precheck

Bash Script

Note: The actual location of runcluvfy.sh might vary depending on your Oracle installation. You'll need to adjust the path accordingly.

```
# Replace '/path/to/oracle/grid/bin/runcluvfy.sh' with the
correct path

/path/to/oracle/grid/bin/runcluvfy.sh -precheck
```

Runcluvfy.sh script is available after unzipping the grid software.

```
Syntax:
./runcluvfy.sh stage -pre crsinst -n host1,host2,host3 -verbose
Example:
./runcluvfy.sh stage -pre crsinst -n classpredb1,classpredb2 -
verbose
```

How to do it?

1. Access the database server as a user with appropriate
2. Edit the script to replace `/path/to/oracle/grid/bin/runcluvfy.sh` with the actual path to the script on your system. You can typically find it in the Grid Infrastructure installation directory under `bin`.
3. Execute the modified script:

The script will perform pre-RAC checks and generate a report indicating any potential issues or configuration recommendations.

7. Copy ASM File to Remote ASM Instance

PL/SQL Script

```
DECLARE
  -- Replace placeholders with actual values
  remote_host VARCHAR2(255) := '<remote_hostname>';
  remote_sid VARCHAR2(255) := '<remote_sid>';
  remote_user VARCHAR2(255) := '<remote_username>';
  remote_password VARCHAR2(255) := '<remote_password>';
  local_file VARCHAR2(255) := '<local_file_path>';
  remote_file VARCHAR2(255) := '<remote_file_path>';
BEGIN
  DBMS_OUTPUT.PUT_LINE('Copying file ' || local_file || ' to '
|| remote_host || ':' || remote_file);

  EXECUTE IMMEDIATE DBMS_RLS.ADD_JOB(
    job_name => 'ASM_FILE_COPY',
    program_name => 'asmcmd cp -h ' || remote_host || ' -u ' ||
remote_user || ' -p ' || remote_password || ' ' || local_file ||
' ' || remote_file,
    schedule_type => 'NONE'
  );

  DBMS_OUTPUT.PUT_LINE('Job submitted. Check ASMCMD output for
details.');
END;
/
```

ASM file can as well be copied to remote asm instance(diskgroup) using asmcmd command.

```
SYNTAX - asmcmd> cp - -port asm_port  file_name
remote_asm_user/remote_asm_pwd@remote_host:Instancce_name:TARGET
_ASM_PATH
ASMCMD> cp --port 1521 s_srv_new21.dbf
sys/oracle@172.20.17.69.+ASM1:+ARCL/s_srv_new21.dbf
```

How to do it?

1. Edit the script to replace placeholders with actual values:
    - `<remote_hostname>`: Hostname of the remote ASM instance
    - `<remote_sid>`: SID of the remote database instance
    - `<remote_user>`: Username with SYSASM privilege on the remote instance
    - `<remote_password>`: Password for the remote user
    - `<local_file_path>`: Path to the ASM file on the local instance
    - `<remote_file_path>`: Destination path for the file on the remote instance
2. Execute the script.
3. The script submits an asynchronous job using `DBMS_RLS` to copy the file. Check ASMCMD output (usually in the alert log) for success or failure messages.

8. Mount/Dismount ASM Diskgroups

PL/SQL Script (Mount)

```
DECLARE
  diskgroup_name VARCHAR2(255) := '<diskgroup_name>';
BEGIN
  DBMS_OUTPUT.PUT_LINE('Mounting diskgroup ' || diskgroup_name);
  EXECUTE IMMEDIATE 'ALTER DISKGROUP ' || diskgroup_name || '
MOUNT';
  DBMS_OUTPUT.PUT_LINE('Diskgroup mounted successfully.');
END;
/
```

PL/SQL Script (Dismount)

```
DECLARE
  diskgroup_name VARCHAR2(255) := '<diskgroup_name>';
BEGIN
  DBMS_OUTPUT.PUT_LINE('Dismounting diskgroup ' ||
diskgroup_name);
  EXECUTE IMMEDIATE 'ALTER DISKGROUP ' || diskgroup_name || '
UNMOUNT FORCE';  -- Use FORCE with caution
  DBMS_OUTPUT.PUT_LINE('Diskgroup dismounted successfully.');
END;
/
```

To mount a diskgroup,(This is instance specific, for mounting on all nodes, run the same on all nodes)

```
SQL>alter diskgroup DATA mount;
or
asmcmd>mount DATA
```

To unmount a diskgroup,(This is instance specific, for unmounting on all nodes, run the same on all nodes)

```
SQL>alter diskgroup DATA dismount;
or
asmcmd>umount DATA
```

`To mount/Dismount all the diskgroups`

```
SQL>alter diskgroup ALL mount;
SQL>alter diskgroup ALL dismount;
```

How to do it?

1. **Mount:** Edit the mount script to specify the `<diskgroup_name>` to mount.
    - Execute the script to mount the diskgroup.
2. **Dismount:** Edit the dismount script to specify the `<diskgroup_name>` to dismount.
    - **Important:** Use `FORCE` with caution as it bypasses some checks. Ensure proper maintenance before dismounting.
    - Execute the script to dismount the diskgroup.

9. Drop ASM Diskgroup

PL/SQL Script (Caution: Use with care!)

```
DECLARE
  diskgroup_name VARCHAR2(255) := '<diskgroup_name>';
BEGIN
  DBMS_OUTPUT.PUT_LINE('Dropping diskgroup ' || diskgroup_name
|| ' (This action is irreversible)');
  DBMS_OUTPUT.PUT_LINE('Are you sure? (y/n)');
  IF UPPER(INPUT('Enter your choice: ')) = 'Y' THEN EXECUTE
```

```
IMMEDIATE 'DROP DISKGROUP ' || diskgroup_name;
DBMS_OUTPUT.PUT_LINE('Diskgroup dropped successfully.'); ELSE
DBMS_OUTPUT.PUT_LINE('Drop operation cancelled.'); END IF; END;
/
```

To drop a diskgroup, make sure the diskgroup has been dismounted from all the remote nodes, It should be mounted only on the local nodes, where we will run the drop command.

```
drop diskgroup DSMREDOA including contents;
```

1. **Important:** This script permanently removes an ASM diskgroup. Use it with extreme caution and only after verifying all backups and consequences.
2. Edit the script to replace `'<diskgroup_name>'` with the actual disk group name.
3. Execute the script.
4. Confirm the drop operation by entering `y` (yes) when prompted.

10. Clock Synchronization Status in RAC

Bash Script

```
# Check CTSS (Cluster Time Synchronization Service) status
/path/to/oracle/grid/bin/srvctl status ctiss -all

# Alternatively, use SQL to check NTP (Network Time Protocol)
SELECT STATUS FROM v$resource WHERE NAME = 'GLOBAL_RAC_TIME';
```

```
--Clock Synchronization across the cluster nodes
cd $GRID_HOME/bin
cluvfy comp clocksync -n all
--Check whether ctss or ntp is running
crsctl check ctss
CRS-4700: The Cluster Time Synchronization Service is in
Observer mode.
Observer means - Time sync between nodes are taken care by NTP
Active means - Time sync between nodes are taken care by CTSS
```

How to do it?

1. **CTSS:** Edit the script to replace `/path/to/oracle/grid/bin/srvctl` with the actual path to the `srvctl` command (usually in the Grid Infrastructure bin directory).
    - Execute the script. The output will display the status of CTSS on all RAC nodes.
2. **NTP:** Alternatively, you can use the provided SQL script to check the status of the NTP resource (`GLOBAL_RAC_TIME`) in `v$resource`. Execute the script in SQL*Plus or any other SQL client.

11. Create ASM Disk in Linux using oracleasm

Bash Script

```
# Replace placeholders with actual values
disk_name=<disk_name>
partition_path=/dev/<partition_path>

# Configure ASM library (if not already done)
oracleasm configure -i

# Initialize ASM library
oracleasm init

# Create ASM disk
oracleasm createdisk $disk_name $partition_path
```

```
--Check the asm disk labelling
#/etc/init.d/oracleasm querydisk /dev/sdn1
Device "/dev/sdn" is not marked as an ASM disk
--Create asm disk
# /etc/init.d/oracleasm createdisk ARCDATA /dev/sdn1
Marking disk "ARCDATA" as an ASM disk: [ OK ]
--Check the asm disk labelling
# /etc/init.d/oracleasm querydisk /dev/sdn1
Device "/dev/sdn1" is marked an ASM disk with the label
"ARCDATA"
--List the asm disks present
# /etc/init.d/oracleasm listdisks
ARCDATA
```

How to do it?

1. Edit the script to replace placeholders:
   - `<disk_name>`: Desired name for the ASM disk
   - `/dev/<partition_path>`: Path to the existing disk partition you want to use
2. Ensure the disk is properly partitioned beforehand.
3. Run the script as a user with appropriate permissions (typically the `oracle` user).
   - The script configures, initializes, and then creates the ASM disk.

12. Stop/Start Cluster in RAC Standalone

Bash Script (Stop)

```
# Stop CRS service
srvctl stop crs

# Stop database instances (optional)
lsnrctl stop && sqlplus /nolog @?/sbin/stop_inst
```

Bash Script (Start)

```
# Start CRS service
srvctl start crs

# Start database instances (optional)
lsnrctl start && sqlplus /nolog @?/sbin/start_inst
#Oracle RAC in standalone is known as oracle restart, where only
HAS(high availability service)  component is available.
crsctl stop has
crsctl start has
```

How to do it?

1. **Stop:** Edit the script if needed to exclude stopping database instances.
   - Execute the script to stop the Cluster Ready Services (CRS) and optionally the database instances.
2. **Start:** Edit the script if needed to exclude starting database instances.
   - Execute the script to start CRS and optionally the database instances.

Important: Ensure proper shutdown procedures are followed before stopping the cluster.

13. Create Password File in ASM DG

PL/SQL Script

```
DECLARE
  diskgroup_name VARCHAR2(255) := '<diskgroup_name>';
  password_file VARCHAR2(255) := '<password_file_path>';
  password VARCHAR2(30);
BEGIN
  DBMS_OUTPUT.PUT_LINE('Creating password file ' ||
password_file);
  DBMS_OUTPUT.PUT_LINE('Enter password (30 characters max): ');
  password := DBMS_INPUT.READ_STRING(30);

  EXECUTE IMMEDIATE 'ALTER DISKGROUP ' || diskgroup_name || '
SET ASM_PASSWORDFILE=' || CHR(39) || password_file || CHR(39);
  DBMS_OUTPUT.PUT_LINE('Password file created successfully.');
END;
/
```

Using Commands:

```
 --For oracle 12c  only
ASMCMD> pwcreate -dbuniquename {db_unique_name}
{file_path}  {sys_password}
ASMCMD> pwcreate --dbuniquename PRDPRE +DATA/PWDFILE/pwdPRDPRE
oracle
--For all version.
orapwd file='+DATA/orapwPRODPRE' ENTRIES=10
DBUNIQUENAME='PRODPRE'
```

How to do it?

1. Edit the script to replace placeholders:
   - `<diskgroup_name>`: Name of the ASM disk group
   - `<password_file_path>`: Path to the password file you want to create (ensure appropriate permissions)
2. Execute the script.

3. Enter a strong password (maximum 30 characters) when prompted. The script creates the password file and associates it with the disk group.

Important: Securely store the password file and restrict access.

14. Change ASM Rebalance Power

PL/SQL Script

```
DECLARE
  rebalance_power NUMBER := <power_value>;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Setting ASM rebalance power to ' ||
rebalance_power);
  EXECUTE IMMEDIATE 'ALTER SYSTEM SET ASM_REBALANCE_POWER=' ||
rebalance_power;
  DBMS_OUTPUT.PUT_LINE('Rebalance power changed successfully.');
END;
/
```

Alternatively:

```
-- Default value of asm_power_limit.
SQL> show parameter asm_power_limit
NAME                                 TYPE        VALUE
------------------------------------ ----------- -----------
asm_power_limit                      integer     1
-- Check for ongoing rebalance operations and their power.
select INST_ID,GROUP_NUMBER, OPERATION, STATE, POWER, EST_RATE,
EST_MINUTES from GV$ASM_OPERATION;
- Alter the asm rebalance.
alter diskgroup SALDATA rebalance power 4;
```

How to do it?

1. Edit the script to replace `<power_value>` with the desired rebalance power (between 0 and 100). Higher values prioritize rebalancing activities.
2. Execute the script. The script modifies the `ASM_REBALANCE_POWER` system parameter.

Important: Adjust the rebalance power with caution, considering your workload and performance needs.

15. Modify ASM User Password

PL/SQL Script

```
DECLARE
  old_password VARCHAR2(30);
  new_password VARCHAR2(30);
BEGIN
  DBMS_OUTPUT.PUT_LINE('Changing ASM user password');
  DBMS_OUTPUT.PUT_LINE('Enter current password: ');
  old_password := DBMS_INPUT.READ_STRING(30);

  DBMS_OUTPUT.PUT_LINE('Enter new password (30 characters max):
');
  new_password := DBMS_INPUT.READ_STRING(30);

  EXECUTE IMMEDIATE 'ALTER USER SYSASM PASSWORD IDENTIFIED BY '
|| CHR(39) || new_password || CHR(39);
  DBMS_OUTPUT.PUT_LINE('Password changed successfully.');
END;
/
```

Using Commands:

```
-- list asm users
ASMCMD> lspwusr
Username sysdba sysoper sysasm
SYS      TRUE   TRUE     TRUE
ASMSNMP  TRUE   FALSE    FALSE -- >

-- Modify user password
ASMCMD> orapwusr --modify asmsnmp
Enter password: ********
```

How to do it?

1. Execute the script.
2. Enter the current ASM user password (SYSASM) when prompted.
3. Enter a strong new password (maximum 30 characters) when prompted. The script modifies the SYSASM user password.

Important: Choose a complex password and avoid storing it in plain text.

16. Monitor ASM Diskgroup I/O

SQL Script

```
SELECT dg.NAME AS diskgroup_name,
       ROUND(AVG(sr.BYTES_READ / 1024 / 1024), 2) AS
avg_read_mbps,
       ROUND(AVG(sr.BYTES_WRITTEN / 1024 / 1024), 2) AS
avg_write_mbps
FROM v$asm_diskgroup dg,
     v$asm_disk d,
     v$asm_stat sr
WHERE dg.GROUP_ID = d.GROUP_ID
  AND d.DISK_ID = sr.DISK_ID
GROUP BY dg.NAME
ORDER BY dg.NAME;
```

How to do it?

1. Execute the script in SQL*Plus or any other SQL client.
2. The script displays the disk group name, average read MB/s, and average write MB/s for each ASM disk group.

17. Enable Tracing for asmcmd

Bash Script

```
# Replace <trace_file> with the desired output file path
asmcmd set tracelvl 10 tracefile <trace_file>
$export DBI_TRACE=1
$ asmcmd
```

How to do it?

1. Edit the script to replace `<trace_file>` with the path to the file where you want to capture `asmcmd` tracing information (ensure write permissions).
2. Execute the script as a user with appropriate permissions (typically the `oracle` user).
    - The script sets the trace level to 10 (comprehensive tracing) and redirects output to the specified file.

Important: Remember to disable tracing (using `asmcmd set tracelvl 0`) when finished to avoid performance overhead.

18. How to Change ASM SYS Password (Use with Extreme Caution!)

PL/SQL Script

```
DECLARE
  old_password VARCHAR2(30);
  new_password VARCHAR2(30);
BEGIN
  DBMS_OUTPUT.PUT_LINE('Changing ASM SYS password (This
operation is highly sensitive)');
  DBMS_OUTPUT.PUT_LINE('**WARNING:** Proceed with extreme
caution. Consider alternative security measures.');
  DBMS_OUTPUT.PUT_LINE('Enter current ASM SYS password: ');
  old_password := DBMS_INPUT.READ_STRING(30);

  DBMS_OUTPUT.PUT_LINE('Enter new ASM SYS password (30
characters max): ');
  new_password := DBMS_INPUT.READ_STRING(30);

  EXECUTE IMMEDIATE 'ALTER USER SYSASM PASSWORD IDENTIFIED BY '
|| CHR(39) || new_password || CHR(39);
  DBMS_OUTPUT.PUT_LINE('Password changed successfully.');
END;
/
```

Alternatively, you may use orapwd to recreate pwd file, or the folloiwng commands:

```
$ export ORACLE_SID=+ASM
$ asmcmd
ASMCMD> orapwusr --modify --password sys
Enter password: ******
```

```
ASMCMD> exit
```

How to do it?

1. **Important:** Only proceed if absolutely necessary and you understand the risks involved.
2. Execute the script.
3. Enter the current ASM SYS password when prompted.
4. Enter a strong new password (maximum 30 characters) when prompted. The script modifies the SYSASM user password.

19. Find and Update ASM SPFILE

Bash Script

```
# Find SPFILE location
spfile_location=$(srvctl config database -d <database_name> |
grep "^SPFILE" | cut -d':' -f2 | tr -d ' ')

# Update SPFILE (replace <spfile_content> with actual content)
echo "<spfile_content>" > $spfile_location

# Restart ASM instance
srvctl stop instance -d <database_name> -i <instance_name>
srvctl start instance -d <database_name> -i <instance_name>
```

Or just use the following commands:

```
-- To find current asm spfile:
$asmcmd
ASMCMD> spget
--- To update asm spfile to a new pfile:
$asmcmd
ASMCMD> spset +OCR_VOTING/CLUSTER/ASMPARAMETERFILE/spfileASM.ora
```

How to do it?

1. Edit the script to replace:

    - `<database_name>`: Name of the database
    - `<instance_name>`: Name of the ASM instance
    - `<spfile_content>`: New content for the SPFILE (ensure proper formatting)
2. Execute the script. The script retrieves the SPFILE location, updates it with the provided content, and restarts the ASM instance.

Important: Ensure the new SPFILE content is valid and thoroughly tested before deployment.

20. Copy SPFILE from One Diskgroup to Another

Important Notes

- Ensure the destination diskgroup is compatible with the SPFILE content.
- Thoroughly test the copied SPFILE in a non-production environment before deploying it in production.

Steps

A. **Obtain SPFILE Location from Source Diskgroup**

PL/SQL Script

```
DECLARE
  source_dg VARCHAR2(255) := '<source_diskgroup>';
  source_file VARCHAR2(255);
BEGIN
  -- Get SPFILE location from source diskgroup
  SELECT NAME INTO source_file
  FROM v$asm_file
  WHERE TYPE = 'SPFILE'
    AND FILE_ID = (SELECT VALUE FROM v$parameter WHERE NAME =
'spfile');

  DBMS_OUTPUT.PUT_LINE('SPFILE location in ' || source_dg || ':
' || source_file);
END;
/
```

- Replace `<source_diskgroup>` with the name of the diskgroup containing the current SPFILE.
- Execute the script. It retrieves and displays the SPFILE location.

B. Copy SPFILE to Destination Diskgroup

PL/SQL Script

```
DECLARE
  dest_dg VARCHAR2(255) := '<destination_diskgroup>';
  dest_file VARCHAR2(255) := '<destination_file_path>';
  source_file VARCHAR2(255); -- Replace with source SPFILE
location from previous script
BEGIN
  -- Update the script with the actual SPFILE location obtained
from step 1
  source_file := '<source_spfile_location>';  -- Replace with
actual source SPFILE path

  DBMS_OUTPUT.PUT_LINE('Copying SPFILE from ' || source_file ||
' to ' || dest_file);
  EXECUTE IMMEDIATE 'ALTER DISKGROUP ' || dest_dg || ' SET
SPFILE=' || CHR(39) || dest_file || CHR(39);
  DBMS_OUTPUT.PUT_LINE('SPFILE copied successfully.');
END;
/
```

1. **Source Diskgroup:** Execute the first script, replacing `<source_diskgroup>` with the name of the diskgroup containing the current SPFILE. The script retrieves the SPFILE location and displays it.
2. **Destination Diskgroup:**
   - Edit the second script:
     - Replace `<destination_diskgroup>` with the name of the diskgroup where you want to copy the SPFILE.
     - Replace `<destination_file_path>` with the desired path and filename for the SPFILE in the destination diskgroup.
   - Update the script with the actual SPFILE location obtained from the first script (replace `<source_spfile_location>`).
   - Execute the second script. The script copies the SPFILE from the source to the destination and updates the SPFILE location for the destination diskgroup.

Important: Ensure the destination diskgroup is compatible with the SPFILE content. Test the copied SPFILE thoroughly in a non-production environment before deploying it in production.

You may also use this command:

```
ASMCMD> spcopy +DATA/ASM/ASMPARAMETERFILE/registry.263.3728290
+MGMT/spfileCopyASM.ora
```

Conclusion

This guide has provided you with essential PL/SQL, SQL, and Bash scripts for core ASM management tasks. Remember to exercise caution when dropping disks and leverage the provided scripts to effectively query ASM information and monitor rebalancing operations. For pre-RAC checks, utilize the `runcluvfy.sh` script to ensure a smooth RAC configuration process.

By following these procedures and tailoring them to your specific environment, you can effectively manage ASM and optimize Oracle database storage utilization.