

Oracle Core Files - Complete Guide

<https://digitalksystems.com>, digitalk.fmw@gmail.com
https://www.youtube.com/channel/UCCGTnI9vvF_ETMhGUXGdFWw

1. DATA FILES

File Extension: .dbf

Location: Varies by storage setup (typically under \$ORACLE_BASE/oradata)

What They Contain: Actual database data—tables, indexes, segments, and all user information stored in tablespaces.

Purpose: Store permanent data on disk. Every table, index, and object you create goes into a datafile belonging to a tablespace.

Key Characteristics:

- One or more datafiles per tablespace
- Size can be fixed or autoextend
- Directly mapped to physical tablespaces
- Lost datafile = data loss unless backed up

Example:

- /u01/oracle/oradata/ORCL/system01.dbf (SYSTEM tablespace)
- /u01/oracle/oradata/ORCL/sysaux01.dbf (SYSAUX tablespace)
- /u01/oracle/oradata/ORCL/users01.dbf (USERS tablespace)
- /u01/oracle/oradata/ORCL/temp01.dbf (TEMP tablespace)
- /u01/oracle/oradata/ORCL/undo01.dbf (UNDO tablespace)

Query to Check:

```
SQL> SELECT FILE_ID, FILE_NAME, TABLESPACE_NAME FROM DBA_DATA_FILES;
```

2. CONTROL FILE

File Extension: .ctl

Location: Multiple copies for redundancy (e.g., /u01/oracle/oradata, /u02/oracle/oradata, /u03/oracle/oradata)

What It Contains: Database metadata—datafile locations, tablespace names, archive log history, backup information, database name, created date.

Purpose: The "roadmap" of the database. Oracle reads it at startup to know where all datafiles are and what state the database should be in.

Key Characteristics:

- Multiple copies stored (minimum 2, preferably 3)
- Small binary file (typically 5-20 MB)
- MUST be readable and writable
- Required for every database startup
- Contains SCN (System Change Number)—tracks database consistency

Why Multiple Copies? If one control file is lost and you have copies, database continues running. If ALL copies are lost, database cannot start.

Example:

```
/u01/oracle/oradata/ORCL/control01.ctl  
/u02/oracle/oradata/ORCL/control02.ctl  
/u03/oracle/oradata/ORCL/control03.ctl
```

Query to Check:

```
SQL> SELECT NAME FROM V$CONTROLFILE;  
SQL> SHOW PARAMETER control_files;
```

What Happens if Lost:

- If one copy lost: Database continues, but restart is risky without backup
- If all copies lost: Database cannot start (ORA-00205: identification of datafile failed)

3. REDO LOG FILES

File Extension: .log or .rdo

Location: Multiple locations for redundancy (e.g., /u01/oracle/oradata, /u02/oracle/oradata)

What They Contain:

- Sequential record of every change made to the database (INSERT, UPDATE, DELETE, COMMIT operations).

Purpose:

- Capture all transactions in real-time. Used for crash recovery and point-in-time recovery.

Key Characteristics:

- Organized in groups (Log Group 1, 2, 3, etc.)
- Each group has multiple members (copies) for protection
- Written sequentially—once filled, switches to next group
- When last group is filled, wraps back to first group (circular buffer)
- Small files compared to datafiles (typically 50 MB - 5 GB)
- Continuously written during database operation

How They Work:

1. Active redo log group receives all changes
2. When filled, automatic log switch occurs
3. New group becomes active, previous group becomes inactive
4. When inactive group is archived (ARCHIVELOG mode), it's copied to archive log location
5. After archiving, group can be reused

Example Structure:

Group 1: /u01/oracle/oradata/ORCL/redo01a.log (member 1)
/u02/oracle/oradata/ORCL/redo01b.log (member 2)

Group 2: /u01/oracle/oradata/ORCL/redo02a.log (member 1)
/u02/oracle/oradata/ORCL/redo02b.log (member 2)

Group 3: /u01/oracle/oradata/ORCL/redo03a.log (member 1)
/u02/oracle/oradata/ORCL/redo03b.log (member 2)

Query to Check:

```
SQL> SELECT GROUP#, MEMBERS, STATUS FROM V$LOG;  
SQL> SELECT GROUP#, MEMBER FROM V$LOGFILE;
```

What Happens if Lost:

- If one member of a group lost: Database switches to next group, can recover from intact members
- If all members of active group lost: Database cannot continue (crash recovery impossible)

4. ARCHIVE LOG FILES

File Extension: .arc or .log

Location: Archive log destination (e.g., /u01/oracle/archive or /u02/oracle/archive)

What They Contain: Permanent copy of redo log files after they have been filled and archived.

Purpose: Enable point-in-time recovery and backup strategy. Without archive logs, you can only recover to last backup point.

Key Characteristics:

- Created only when database is in ARCHIVELOG mode
- Generated when redo log group fills and switches
- Named sequentially with archive sequence number

- Can be backed up and deleted to free space
- Essential for hot backups and recovery
- Typically large files (100 MB - several GB)

Naming Convention: Archive logs are usually named with sequence numbers and timestamps:

```
archive_1_20240115_098765432.arc
archive_2_20240115_098765433.arc
archive_3_20240115_098765434.arc
```

Example: If database generates 5 redo log switches per hour, archive directory will accumulate archive logs continuously.

Query to Check:

```
SQL> SELECT NAME FROM V$ARCHIVED_LOG WHERE DELETED = 'NO';
SQL> ARCHIVE LOG LIST; -- Shows archive mode status and destination
```

ARCHIVELOG vs NOARCHIVELOG Mode:

- NOARCHIVELOG:
 - Redo logs are overwritten (circular), no archive logs created. Recovery only to last backup.
- ARCHIVELOG:
 - Redo logs are archived before being overwritten. Point-in-time recovery possible.

Check Current Mode:

```
SQL> SELECT LOG_MODE FROM V$DATABASE;
-- Returns: ARCHIVELOG or NOARCHIVELOG
```

5. PFILE (Parameter File)

File Name: init<SID>.ora

Location: \$ORACLE_HOME/dbs/init<SID>.ora (e.g., /u01/oracle/product/19c/dbs/initORCL.ora)

What It Contains: Plain text file with database startup parameters and initialization settings.

Purpose: Tells Oracle how to start the database—memory allocation, archive log settings, database name, control file locations, etc.

Key Characteristics:

- Plain text file (human-readable)
- One parameter per line
- Read during database startup (STARTUP command)
- Changes require database restart to take effect
- Older method (replaced by SPFILE in Oracle 9i+)

Example Contents:

- *.db_name='ORCL'
- *.instance_number=1
- *.memory_target=2G
- *.processes=300
- *.open_cursors=600
- *.db_recovery_file_dest='/u01/oracle/fra'
- *.control_files='/u01/oracle/oradata/ORCL/control01.ctl',
- '/u02/oracle/oradata/ORCL/control02.ctl'
- *.log_archive_dest_1='LOCATION=/u01/oracle/archive VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=ORCL'

Query to Check:

```
SQL> SHOW PARAMETER spfile; -- Shows which file is being used
```

Disadvantages:

- Manual editing needed
- Easy to make typos
- No validation until restart
- Different file on each server (RAC environments)

6. SPFILE (Server Parameter File)

File Name: spfile<SID>.ora

Location: \$ORACLE_HOME/dbs/spfile<SID>.ora (e.g., /u01/oracle/product/19c/dbs/spfileORCL.ora)

What It Contains: Binary file containing database parameters (same information as PFILE).

Purpose: Modern replacement for PFILE. Stores initialization parameters in binary format for centralized management.

Key Characteristics:

- Binary file (not human-readable)
- Can be modified with ALTER SYSTEM command without restart (SCOPE=BOTH)
- Changes persist across restarts
- Single source of parameters (centralized)
- Creates text backup file spfile<SID>.ora.bak
- Only one SPFILE per database instance

Advantages over PFILE:

- Dynamic parameter changes (some parameters)
- No manual editing needed
- Automatic backup of changes
- Validation before applying
- Easier for RAC (centralized on shared storage)

Example Dynamic Parameter Change:

```
SQL> ALTER SYSTEM SET SGA_TARGET=3G SCOPE=BOTH;  
-- Changes take effect immediately and persist after restart
```

Create SPFILE from PFILE:

```
SQL> CREATE SPFILE FROM PFILE;
```

Create PFILE from SPFILE:

```
SQL> CREATE PFILE FROM SPFILE;
```

Query to Check Current Settings:

```
SQL> SHOW PARAMETER db_name;  
SQL> SHOW PARAMETER processes;
```

7. TEMP FILES / TEMPORARY TABLESPACE

File Extension: .tmp

Location: Typically under \$ORACLE_BASE/oradata

What They Contain: Temporary data used during SQL operations—sorting, joining, aggregation.

Purpose: Provide temporary working space for large operations without using permanent storage.

Key Characteristics:

- Contents lost after operation completes
- Not recovered after crash
- Used for sort areas, hash joins, temporary segments
- Multiple temp files possible in TEMP tablespace
- Size can be autoextended
- Not included in database backups (can be recreated)

Example:

/u01/oracle/oradata/ORCL/temp01.dbf

When Used:

```
-- Sorts on large tables
SELECT * FROM large_table ORDER BY column1, column2;
-- Joins of large tables
SELECT * FROM table1 JOIN table2 ON table1.id = table2.id;
-- GROUP BY operations on large datasets
SELECT column1, COUNT(*) FROM large_table GROUP BY column1;
```

Query to Check:

```
SQL> SELECT FILE_ID, FILE_NAME, TABLESPACE_NAME FROM DBA_TEMP_FILES;
```

8. UNDO TABLESPACE (UNDO FILES)

File Extension: .dbf

Location: Typically under \$ORACLE_BASE/oradata

What They Contain: Before-images of changed data—used to rollback uncommitted transactions and provide read consistency.

Purpose: Allow transactions to be rolled back and provide consistent read views for queries.

Key Characteristics:

- Automatically managed (AUM - Automatic Undo Management)
- Data persists for transaction duration, then reused
- Size must accommodate all concurrent transactions
- Insufficient undo can cause transaction failures
- One active undo tablespace per instance
- In RAC, each instance has its own undo tablespace

Example:

```
/u01/oracle/oradata/ORCL/undo01.dbf
```

How It Works:

1. Transaction starts: INSERT INTO employees VALUES (...)
2. Old data (before change) written to UNDO tablespace
3. New data written to DATA file
4. If COMMIT: undo data marked for reuse
5. If ROLLBACK: undo data replayed, original data restored

Query to Check:

```
SQL> SELECT TABLESPACE_NAME, STATUS FROM DBA_TABLESPACES WHERE CONTENTS='UNDO';
SQL> SHOW PARAMETER undo_tablespace;
```

9. INITIALIZATION PARAMETER FILES SUMMARY

File Type	Format	Location	Readable	Editable	Modern
PFILE	Text	\$ORACLE_HOME/dbs/init<SID>.ora	Yes	Manual edit	No (legacy)
SPFILE	Binary	\$ORACLE_HOME/dbs/spfile<SID>.ora	No	ALTER SYSTEM	Yes (current)

Complete File Organization Example

Database: ORCL

Data Files:

/u01/oracle/oradata/ORCL/system01.dbf
/u01/oracle/oradata/ORCL/sysaux01.dbf
/u01/oracle/oradata/ORCL/users01.dbf
/u01/oracle/oradata/ORCL/temp01.dbf
/u01/oracle/oradata/ORCL/undo01.dbf

Control Files:

/u01/oracle/oradata/ORCL/control01.ctl
/u02/oracle/oradata/ORCL/control02.ctl
/u03/oracle/oradata/ORCL/control03.ctl

Redo Log Files:

Group 1: /u01/oracle/oradata/ORCL/redo01a.log, /u02/oracle/oradata/ORCL/redo01b.log
Group 2: /u01/oracle/oradata/ORCL/redo02a.log, /u02/oracle/oradata/ORCL/redo02b.log
Group 3: /u01/oracle/oradata/ORCL/redo03a.log, /u02/oracle/oradata/ORCL/redo03b.log

Archive Log Files:

/u01/oracle/archive/archive_1_20240115_098765432.arc
/u01/oracle/archive/archive_2_20240115_098765433.arc
/u01/oracle/archive/archive_3_20240115_098765434.arc

Parameter Files:

/u01/oracle/product/19c/dbs/spfileORCL.ora
/u01/oracle/product/19c/dbs/initORCL.ora (or created from SPFILE)

Alert Log:

/u01/oracle/diag/rdbms/orcl/ORCL/trace/alert_ORCL.log

Recovery Scenario: Why Each File Matters

Scenario: Database crashes and all files are corrupt

1. **SPFILE** (or PFILE): Restore to know what database configuration was
2. **Control Files**: Restore to know where all datafiles are located
3. **Datafiles**: Restore to get all data back
4. **Redo Logs**: Used during recovery to replay recent changes
5. **Archive Logs**: Used to recover to point just before crash
6. **Undo Tablespace**: Helps maintain consistency during recovery

Without any one of these: Complete recovery may not be possible.

Checklist: Essential Files to Backup

- All datafiles from all tablespaces
- All control files (at least 1 copy)
- SPFILE (or PFILE if SPFILE doesn't exist)
- Archive logs (if ARCHIVELOG mode enabled)
- Redo logs (automatically protected by multiple members)

Not needed in backup:

- Temp files (can be recreated)
- Active redo logs (captured in archive logs)

DigiTalk Udemy Course Links and Coupon Codes:

Mastering Apache Web Server 2025 — From Basics to Advance

<https://www.udemy.com/course/mastering-apache-web-server-2025-from-basics-to-advance/?referralCode=sCDB8D72D2925F9E7F0C>

Coupon Code: C8097E29A6BDAC103F00

Mastering SSL in Oracle WebLogic (12c, 14c & 15c)

<https://www.udemy.com/course/mastering-amazon-web-services-aws-from-zero-to-cloud-expert/?referralCode=EB7C9CA90D3921152859>

Coupon Code: FE8A7CEF11510CB60029

Mastering Ansible Administration

<https://www.udemy.com/course/mastering-ansible-administration/?referralCode=119798378DB4239680EB>

Coupon Code: D5FBA7282AF52C4D1383

WebLogic Server Administration

<https://www.udemy.com/course/oracle-weblogic-server-12c-and-14c-administration/?referralCode=340FBB94DBFEE1A8CC09>

Coupon Code: 68812FE2DD5454B0D72B

Mastering Amazon Web Services AWS: From Zero to Cloud Expert

<https://www.udemy.com/course/mastering-amazon-web-services-aws-from-zero-to-cloud-expert/?referralCode=EB7C9CA90D3921152859>

Coupon Code: 78709D23AE0D12D4F524

Tomcat Administration:

<https://www.udemy.com/course/mastering-apache-tomcat-from-basics-to-advanced/?referralCode=E6B1CB09E9944CEE8D72>

Coupon Code: 9C0A242432A1F7A37CE6

Docker Administration

<https://www.udemy.com/course/mastering-docker-administration-from-fundamentals-to-expert/?referralCode=42163FA122015C6EE67D>

Coupon Code: 747BB0BF4BB6A8F4CC64

Kubernetes Administration

<https://www.udemy.com/course/mastering-kubernetes-administration-from-intro-to-advanced/?referralCode=7E0250BEC21F46192834>

Coupon Code: 05E2222A535CF8E4CA3F

WebSphere Administration

<https://www.udemy.com/course/mastering-ibm-websphere-9x-administration/?referralCode=F372B3EA78D7BEA92416>

Coupon Code: F69A71A6EAAAA489114C

Mastering SSL (TLS), Keys and Certificates Management

<https://www.udemy.com/course/mastering-ssl-tls-keys-and-certificates-management/?referralCode=DB9F1429CB01921969F6>

Coupon Code: 2B97372633E06F65ED94

SOA Suite Administration

<https://www.udemy.com/course/mastering-oracle-soa-suite-12c-administration/?referralCode=2BCF3895067AA7BC22E0>

Coupon Code: 17B54EDB2B77C249B290

JBoss 8 Administration

<https://www.udemy.com/course/mastering-jboss-eap-8-administration-from-intro-to-advanced/?referralCode=D095CE8B1EE9D065C822>

Coupon Code: D92B4A52593DBADE5618

OHS Administration

<https://www.udemy.com/course/mastering-oracle-ohs-http-12c-web-server-administration/?referralCode=165F6A3A8B662CEA1489>

Coupon Code: 787A792CAEA24F4B7FCF

Mastering Oracle Internet Direct (OID) Administration

<https://www.udemy.com/course/mastering-oracle-internet-direct-oid-administration/?referralCode=906E9AC0059A29A3CD0D>

Coupon Code: DD15BE40BBF796BDD972

For any query or concerns please write us on digitalk.fmw@gmail.com