8
Managing Control Files

You can create, back up, and drop control files.

- **What Is a Control File?**
  Every Oracle Database has a **control file**, which is a small binary file that records the physical structure of the database.

- **Guidelines for Control Files**
  You can follow guidelines to manage the control files for a database.

- **Creating Control Files**
  You can create, copy, rename, and relocate control files.

- **Troubleshooting After Creating Control Files**
  After issuing the `CREATE CONTROLFILE` statement, you may encounter some errors.

- **Backing Up Control Files**
  Use the `ALTER DATABASE BACKUP CONTROLFILE` statement to back up your control files.

- **Recovering a Control File Using a Current Copy**
  You can recover your control file from a current backup or from a multiplexed copy.

- **Dropping Control Files**
  You can drop control files, but the database should have at least two control files at all times.

- **Control Files Data Dictionary Views**
  You can query a set of data dictionary views for information about control files.

> ✎ **See Also:**
>
> - *Oracle Database Concepts* for an overview of control files
> - Using Oracle Managed Files for information about creating control files that are both created and managed by the Oracle Database server

8.1 What Is a Control File?

Every Oracle Database has a **control file**, which is a small binary file that records the physical structure of the database.

The control file includes:

- The database name
- Names and locations of associated data files and redo log files
- The timestamp of the database creation
- The current log sequence number
- Checkpoint information

The control file must be available for writing by the Oracle Database server whenever the database is open. Without the control file, the database cannot be mounted and recovery is difficult.

The control file of an Oracle Database is created at the same time as the database. By default, at least one copy of the control file is created during database creation. On some operating systems the default is to create multiple copies. You should create two or more copies of the control file during database creation. You can also create control files later, if you lose control files or want to change particular settings in the control files.

8.2 Guidelines for Control Files

You can follow guidelines to manage the control files for a database.

- **Provide File Names for the Control Files**
  You specify control file names using the `CONTROL_FILES` initialization parameter in the database initialization parameter file. The instance recognizes and opens all the listed file during startup, and the instance writes to and maintains all listed control files during database operation.

- **Multiplex Control Files on Different Disks**
  Every Oracle Database should have at least two control files, each stored on a different physical disk.

- **Back Up Control Files**
  It is very important that you back up your control files. This is true initially, and every time you change the physical structure of your database.

- **Manage the Size of Control Files**
  The main determinants of the size of a control file are the values set for the `MAXDATAFILES`, `MAXLOGFILES`, `MAXLOGMEMBERS`, `MAXLOGHISTORY`, and `MAXINSTANCES` parameters in the `CREATE DATABASE` statement that created the associated database.

8.2.1 Provide File Names for the Control Files

You specify control file names using the `CONTROL_FILES` initialization parameter in the database initialization parameter file. The instance recognizes and opens all the listed file during startup, and the instance writes to and maintains all listed control files during database operation.

If you do not specify files for `CONTROL_FILES` before database creation:

- If you are not using Oracle Managed Files, then the database creates a control file and uses a default file name. The default name is operating system specific.

- If you are using Oracle Managed Files, then the initialization parameters you set to enable that feature determine the name and location of the control files.

- If you are using Oracle Automatic Storage Management (Oracle ASM), you can place incomplete Oracle ASM file names in the `DB_CREATE_FILE_DEST` and `DB_RECOVERY_FILE_DEST` initialization parameters. Oracle ASM then automatically creates control files in the appropriate places.

Related Topics

- **Creating Initial Control Files**
  The initial control files of an Oracle Database are created when you issue the `CREATE DATABASE` statement.

- Using Oracle Managed Files
  Oracle Database can manage the files that comprise the database.

- *Oracle Automatic Storage Management Administrator's Guide*

8.2.2 Multiplex Control Files on Different Disks

Every Oracle Database should have at least two control files, each stored on a different physical disk.

If a control file is damaged due to a disk failure, the associated instance must be shut down. Once the disk drive is repaired, the damaged control file can be restored using the intact copy of the control file from the other disk and the instance can be restarted. In this case, no media recovery is required.

The behavior of multiplexed control files is this:

- The database writes to all file names listed for the initialization parameter `CONTROL_FILES` in the database initialization parameter file.

- The database reads only the first file listed in the `CONTROL_FILES` parameter during database operation.

- If any of the control files become unavailable during database operation, the instance becomes inoperable and should be terminated.

> **Note:**
>
> Oracle strongly recommends that your database has a minimum of two control files and that they are located on separate physical disks.

One way to multiplex control files is to store a control file copy on every disk drive that stores members of redo log groups, if the redo log is multiplexed. By storing control files in these locations, you minimize the risk that all control files and all groups of the redo log will be lost in a single disk failure.

8.2.3 Back Up Control Files

It is very important that you back up your control files. This is true initially, and every time you change the physical structure of your database.

Such structural changes include:

- Adding, dropping, or renaming data files

- Adding or dropping a tablespace, or altering the read/write state of the tablespace

- Adding or dropping redo log files or groups

The methods for backing up control files are discussed in "Backing Up Control Files".

8.2.4 Manage the Size of Control Files

The main determinants of the size of a control file are the values set for the `MAXDATAFILES`, `MAXLOGFILES`, `MAXLOGMEMBERS`, `MAXLOGHISTORY`, and `MAXINSTANCES` parameters in the `CREATE DATABASE` statement that created the associated database.

Increasing the values of these parameters increases the size of a control file of the associated database.

> **See Also:**
>
> - Your operating system specific Oracle documentation contains more information about the maximum control file size.
> - *Oracle Database SQL Language Reference* for a description of the CREATE DATABASE statement

8.3 Creating Control Files

You can create, copy, rename, and relocate control files.

- **Creating Initial Control Files**
  The initial control files of an Oracle Database are created when you issue the CREATE DATABASE statement.

- **Creating Additional Copies, Renaming, and Relocating Control Files**
  You can create an additional control file copy for multiplexing by copying an existing control file to a new location and adding the file name to the list of control files.

- **Creating New Control Files**
  You can create new control files when all of the control files for the database have been permanently damaged and you do not have a control file backup or when you want to change the database name.

8.3.1 Creating Initial Control Files

The initial control files of an Oracle Database are created when you issue the CREATE DATABASE statement.

The names of the control files are specified by the CONTROL_FILES parameter in the initialization parameter file used during database creation. The file names specified in CONTROL_FILES should be fully specified and are operating system specific. The following is an example of a CONTROL_FILES initialization parameter:

```
CONTROL_FILES = (/u01/oracle/prod/control01.ctl,
                 /u02/oracle/prod/control02.ctl,
                 /u03/oracle/prod/control03.ctl)
```

If files with the specified names currently exist at the time of database creation, you must specify the CONTROLFILE REUSE clause in the CREATE DATABASE statement, or else an error occurs. Also, if the size of the old control file differs from the SIZE parameter of the new one, you cannot use the REUSE clause.

The size of the control file changes between some releases of Oracle Database, as well as when the number of files specified in the control file changes. Configuration parameters such as MAXLOGFILES, MAXLOGMEMBERS, MAXLOGHISTORY, MAXDATAFILES, and MAXINSTANCES affect control file size.

You can subsequently change the value of the CONTROL_FILES initialization parameter to add more control files or to change the names or locations of existing control files.

> **See Also:**
>
> Your operating system specific Oracle documentation contains more information about specifying control files.

8.3.2 Creating Additional Copies, Renaming, and Relocating Control Files

You can create an additional control file copy for multiplexing by copying an existing control file to a new location and adding the file name to the list of control files.

Similarly, you rename an existing control file by copying the file to its new name or location, and changing the file name in the control file list. In both cases, to guarantee that control files do not change during the procedure, shut down the database before copying the control file.

To add a multiplexed copy of the current control file or to rename a control file:

1. Shut down the database.

2. Copy an existing control file to a new location, using operating system commands.

3. Edit the `CONTROL_FILES` parameter in the database initialization parameter file to add the new control file name, or to change the existing control file name.

4. Restart the database.

8.3.3 Creating New Control Files

You can create new control files when all of the control files for the database have been permanently damaged and you do not have a control file backup or when you want to change the database name.

- When to Create New Control Files
  You must create new control files in certain situations.

- The CREATE CONTROLFILE Statement
  You can create a new control file for a database using the `CREATE CONTROLFILE` statement.

- Creating New Control Files
  You can create new control files for your database.

8.3.3.1 When to Create New Control Files

You must create new control files in certain situations.

It is necessary for you to create new control files in the following situations:

- All control files for the database have been permanently damaged and you do not have a control file backup.

- You want to change the database name.

  For example, you would change a database name if it conflicted with another database name in a distributed environment.

> **Note:**
>
> You can change the database name and DBID (internal database identifier) using the DBNEWID utility. See *Oracle Database Utilities* for information about using this utility.

8.3.3.2 The CREATE CONTROLFILE Statement

You can create a new control file for a database using the `CREATE CONTROLFILE` statement.

The following statement creates a new control file for the `prod` database (a database that formerly used a different database name):

```
CREATE CONTROLFILE
    SET DATABASE prod
    LOGFILE GROUP 1 ('/u01/oracle/prod/redo01_01.log',
                     '/u01/oracle/prod/redo01_02.log'),
            GROUP 2 ('/u01/oracle/prod/redo02_01.log',
                     '/u01/oracle/prod/redo02_02.log'),
            GROUP 3 ('/u01/oracle/prod/redo03_01.log',
                     '/u01/oracle/prod/redo03_02.log')
    RESETLOGS
    DATAFILE '/u01/oracle/prod/system01.dbf' SIZE 3M,
             '/u01/oracle/prod/rbs01.dbs' SIZE 5M,
             '/u01/oracle/prod/users01.dbs' SIZE 5M,
             '/u01/oracle/prod/temp01.dbs' SIZE 5M
    MAXLOGFILES 50
    MAXLOGMEMBERS 3
    MAXLOGHISTORY 400
    MAXDATAFILES 200
    MAXINSTANCES 6
    ARCHIVELOG;
```

> **Note:**
>
> - The `CREATE CONTROLFILE` statement can potentially damage specified data files and redo log files. Omitting a file name can cause loss of the data in that file, or loss of access to the entire database. Use caution when issuing this statement and be sure to follow the instructions in "Creating New Control Files".
>
> - If the database had forced logging enabled before creating the new control file, and you want it to continue to be enabled, then you must specify the `FORCE LOGGING` clause in the `CREATE CONTROLFILE` statement. See "*Oracle Database SQL Language Reference*".

> **See Also:**
>
> *Oracle Database SQL Language Reference* describes the complete syntax of the `CREATE CONTROLFILE` statement

ORACLE

8.3.3.3 Creating New Control Files

You can create new control files for your database.

Complete the following steps to create a new control file.

1. Make a list of all data files and redo log files of the database.

   If you follow recommendations for control file backups as discussed in "Backing Up Control Files" , you will already have a list of data files and redo log files that reflect the current structure of the database. However, if you have no such list, executing the following statements will produce one.

   ```
   SELECT MEMBER FROM V$LOGFILE;
   SELECT NAME FROM V$DATAFILE;
   SELECT VALUE FROM V$PARAMETER WHERE NAME = 'control_files';
   ```

   If you have no such lists and your control file has been damaged so that the database cannot be opened, try to locate all of the data files and redo log files that constitute the database. Any files not specified in step 5 are not recoverable once a new control file has been created. Moreover, if you omit any of the files that comprise the SYSTEM tablespace, you might not be able to recover the database.

2. Shut down the database.

   If the database is open, shut down the database normally if possible. Use the IMMEDIATE or ABORT clauses only as a last resort.

3. Back up all data files and redo log files of the database.

4. Start up a new instance, but do not mount or open the database:

   ```
   STARTUP NOMOUNT
   ```

5. Create a new control file for the database using the CREATE CONTROLFILE statement.

   When creating a new control file, specify the RESETLOGS clause if you have lost any redo log groups in addition to control files. In this case, you will need to recover from the loss of the redo logs (step 8). You must specify the RESETLOGS clause if you have renamed the database. Otherwise, select the NORESETLOGS clause.

6. Store a backup of the new control file on an offline storage device. See "Backing Up Control Files" for instructions for creating a backup.

7. Edit the CONTROL_FILES initialization parameter for the database to indicate all of the control files now part of your database as created in step 5 (not including the backup control file). If you are renaming the database, edit the DB_NAME parameter in your instance parameter file to specify the new name.

8. Recover the database if necessary. If you are not recovering the database, skip to step 9.

   If you are creating the control file as part of recovery, recover the database. If the new control file was created using the NORESETLOGS clause, you can recover the database with complete, closed database recovery.

   If the new control file was created using the RESETLOGS clause, you must specify USING BACKUP CONTROL FILE. If you have lost online redo logs, archived redo log files, or data files, use the procedures for recovering those files.

> **✎ See Also:**
>
> *Oracle Database Backup and Recovery User's Guide* for information about recovering your database and methods of recovering a lost control file

9. Open the database using one of the following methods:

   - If you did not perform recovery, or you performed complete, closed database recovery in step 8, open the database normally.

     ```
     ALTER DATABASE OPEN;
     ```

   - If you specified `RESETLOGS` when creating the control file, use the `ALTER DATABASE` statement, indicating `RESETLOGS`.

     ```
     ALTER DATABASE OPEN RESETLOGS;
     ```

   The database is now open and available for use.

8.4 Troubleshooting After Creating Control Files

After issuing the `CREATE CONTROLFILE` statement, you may encounter some errors.

- Checking for Missing or Extra Files
  After creating a new control file and using it to open the database, check the alert log to see if the database has detected inconsistencies between the data dictionary and the control file, such as a data file in the data dictionary includes that the control file does not list.

- Handling Errors During CREATE CONTROLFILE
  If Oracle Database sends you an error when you attempt to mount and open the database after creating a new control file, the most likely cause is that you omitted a file from the `CREATE CONTROLFILE` statement or included one that should not have been listed.

8.4.1 Checking for Missing or Extra Files

After creating a new control file and using it to open the database, check the alert log to see if the database has detected inconsistencies between the data dictionary and the control file, such as a data file in the data dictionary includes that the control file does not list.

If a data file exists in the data dictionary but not in the new control file, the database creates a placeholder entry in the control file under the name `MISSINGnnnn`, where *nnnn* is the file number in decimal. `MISSINGnnnn` is flagged in the control file as being offline and requiring media recovery.

If the actual data file corresponding to `MISSINGnnnn` is read-only or offline normal, then you can make the data file accessible by renaming `MISSINGnnnn` to the name of the actual data file. If `MISSINGnnnn` corresponds to a data file that was not read-only or offline normal, then you cannot use the rename operation to make the data file accessible, because the data file requires media recovery that is precluded by the results of `RESETLOGS`. In this case, you must drop the tablespace containing the data file.

Conversely, if a data file listed in the control file is not present in the data dictionary, then the database removes references to it from the new control file. In both cases, the database includes an explanatory message in the alert log to let you know what was found.

8.4.2 Handling Errors During CREATE CONTROLFILE

If Oracle Database sends you an error when you attempt to mount and open the database after creating a new control file, the most likely cause is that you omitted a file from the `CREATE CONTROLFILE` statement or included one that should not have been listed.

Typically, the error is `ORA-01173`, `ORA-01176`, `ORA-01177`, `ORA-01215`, or `ORA-01216`. In this case, you should restore the files you backed up in "Creating New Control Files" and repeat the procedure in that task, using the correct file names.

8.5 Backing Up Control Files

Use the `ALTER DATABASE BACKUP CONTROLFILE` statement to back up your control files.

You have two options:

- Back up the control file to a binary file (duplicate of existing control file) using the following statement:

  ```
  ALTER DATABASE BACKUP CONTROLFILE TO '/oracle/backup/control.bkp';
  ```

- Produce SQL statements that can later be used to re-create your control file:

  ```
  ALTER DATABASE BACKUP CONTROLFILE TO TRACE;
  ```

  This command writes a SQL script to a trace file where it can be captured and edited to reproduce the control file. View the alert log to determine the name and location of the trace file.

> ✎ **See Also:**
>
> – *Oracle Database Backup and Recovery User's Guide* for more information on backing up your control files
> – "Viewing the Alert Log"

8.6 Recovering a Control File Using a Current Copy

You can recover your control file from a current backup or from a multiplexed copy.

- Recovering from Control File Corruption Using a Control File Copy
  If a control file becomes corrupted, then you can recover it using a control file copy.

- Recovering from Permanent Media Failure Using a Control File Copy
  If there is permanent media failure, then you can recover by using a control file copy.

8.6.1 Recovering from Control File Corruption Using a Control File Copy

If a control file becomes corrupted, then you can recover it using a control file copy.

This method assumes that one of the control files specified in the `CONTROL_FILES` parameter is corrupted, that the control file directory is still accessible, and that you have a multiplexed copy of the control file.

1. With the instance shut down, use an operating system command to overwrite the bad control file with a good copy:

```
% cp /u03/oracle/prod/control03.ctl  /u02/oracle/prod/control02.ctl
```

2. Start SQL*Plus and open the database:

```
SQL> STARTUP
```

8.6.2 Recovering from Permanent Media Failure Using a Control File Copy

If there is permanent media failure, then you can recover by using a control file copy.

This method assumes that one of the control files specified in the CONTROL_FILES parameter is inaccessible due to a permanent media failure and that you have a multiplexed copy of the control file.

1. With the instance shut down, use an operating system command to copy the current copy of the control file to a new, accessible location:

```
% cp /u01/oracle/prod/control01.ctl  /u04/oracle/prod/control03.ctl
```

2. Edit the CONTROL_FILES parameter in the initialization parameter file to replace the bad location with the new location:

```
CONTROL_FILES = (/u01/oracle/prod/control01.ctl,
                 /u02/oracle/prod/control02.ctl,
                 /u04/oracle/prod/control03.ctl)
```

3. Start SQL*Plus and open the database:

```
SQL> STARTUP
```

If you have multiplexed control files, you can get the database started up quickly by editing the CONTROL_FILES initialization parameter. Remove the bad control file from CONTROL_FILES setting and you can restart the database immediately. Then you can perform the reconstruction of the bad control file and at some later time shut down and restart the database after editing the CONTROL_FILES initialization parameter to include the recovered control file.

8.7 Dropping Control Files

You can drop control files, but the database should have at least two control files at all times.

You want to drop control files from the database, for example, if the location of a control file is no longer appropriate.

1. Shut down the database.

2. Edit the CONTROL_FILES parameter in the database initialization parameter file to delete the old control file name.

3. Restart the database.

> ✎ **Note:**
>
> This operation does not physically delete the unwanted control file from the disk. Use operating system commands to delete the unnecessary file after you have dropped the control file from the database.

ORACLE

8.8 Control Files Data Dictionary Views

You can query a set of data dictionary views for information about control files.

The following views display information about control files:

View	Description
V$DATABASE	Displays database information from the control file
V$CONTROLFILE	Lists the names of control files
V$CONTROLFILE_RECORD_SECTION	Displays information about control file record sections
V$PARAMETER	Displays the names of control files as specified in the CONTROL_FILES initialization parameter

This example lists the names of the control files.

```
SQL> SELECT NAME FROM V$CONTROLFILE;

NAME
------------------------------------
/u01/oracle/prod/control01.ctl
/u02/oracle/prod/control02.ctl
/u03/oracle/prod/control03.ctl
```