Week #1: Database Structure and ASM

Oracle is based on the client–server architecture. The Oracle server consists of the database (the raw data, including log and control files) and the instance (the processes and system memory on the server that provide access to the database). An instance can connect to only one database. The database consists of a logical structure, such as the database schema, and a physical structure, containing the files that make up an Oracle database.

1.1 Oracle's Logical Database Structure

At the logical level, Oracle maintains *tablespaces*, *schemas*, and *data blocks* and *extents/segments*.

Tablespaces

An Oracle database is divided into logical storage units called tablespaces. A tablespace is used to group related logical structures together. For example, tablespaces commonly group all the application's objects to simplify some administrative operations.

Every Oracle database contains a tablespace named SYSTEM, which is created automatically when the database is created. The SYSTEM tablespace always contains the system catalog tables (called the data dictionary in Oracle) for the entire database. A small database might need only the SYSTEM tablespace; however, it is recommended that at least one additional tablespace is created to store user data separate from the data dictionary, thereby reducing contention among dictionary objects and schema objects for the same datafiles.

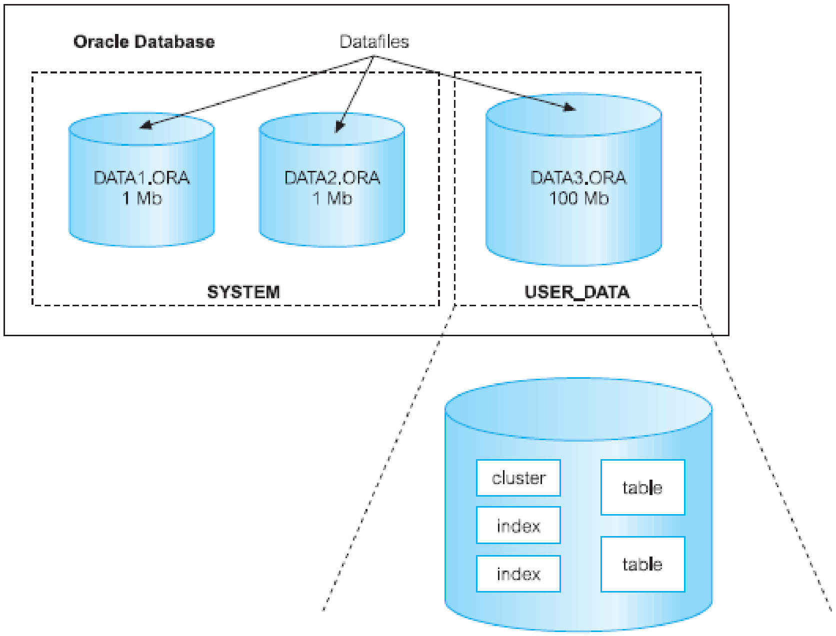

Figure 1.1: *Relationship between an Oracle database, tablespaces, and data files.*

Figure 1.1 illustrates an Oracle database consisting of the SYSTEM tablespace and a USER_DATA tablespace.

A new tablespace can be created using the CREATE TABLESPACE command,

For example:

CREATE TABLESPACE user_data
DATAFILE 'DATA3.ORA' **SIZE** 100K
EXTENT MANAGEMENT LOCAL
SEGMENT SPACE MANAGEMENT AUTO;

A table can then be associated with a specific tablespace using the CREATE TABLE or ALTER TABLE statement, for example:

CREATE TABLE PropertyForRent (propertyNo **VARCHAR2**(5) **NOT NULL**, . . .)
TABLESPACE user_data;

If no tablespace is specified when creating a new table, the default tablespace associated with the user when the user account was set up is used.

Users, Schemas and Schema Objects

- A **user** (sometimes called a username) is a name defined in the database that can connect to, and access, objects.
- A **schema** is a named collection of schema objects, such as tables, views, indexes, clusters, and procedures, associated with a particular user. Schemas and users help DBAs manage database security.

To access a database, a user must run a database application (such as Oracle Forms or SQL*Plus) and connect using a username defined in the database. When a database user is created, a corresponding schema of the same name is created for the user. By default, once a user connects to a database, the user has access to all objects contained in the corresponding schema. As a user is associated only with the schema of the same name; the terms "user" and "schema" are often used interchangeably.

> **NOTE**: There is no relationship between a tablespace and a schema: Objects in the same schema can be in different tablespaces, and a tablespace can hold objects from different schemas.

Data blocks, extents, and segments

The **data block** is the smallest unit of storage that Oracle can use or allocate. One data block corresponds to a specific number of bytes of physical disk space. The data block size can be set for each Oracle database when it is created. This data block size should be a multiple of the operating system's block size (within the system's maximum operating limit) to avoid unnecessary I/O. A data block has the following structure:

- *Header.* contains general information such as block address and type of segment.
- *Table directory.* contains information about the tables that have data in the data block.
- *Row directory.* contains information about the rows in the data block.
- *Row data.* contains the actual rows of table data. A row can span blocks.
- *Free space.* allocated for the insertion of new rows and updates to rows that require additional space. As of Oracle8i, Oracle can manage free space automatically, although there is an option to manage it manually.

The next level of logical database space is called an **extent**. An extent is a specific number of contiguous data blocks allocated for storing a specific type of information. The level above an extent is called a **segment**. A segment is a set of extents allocated for a certain logical structure. For example, each table's data is stored in its own data segment, and each index's data is stored in its own index segment.

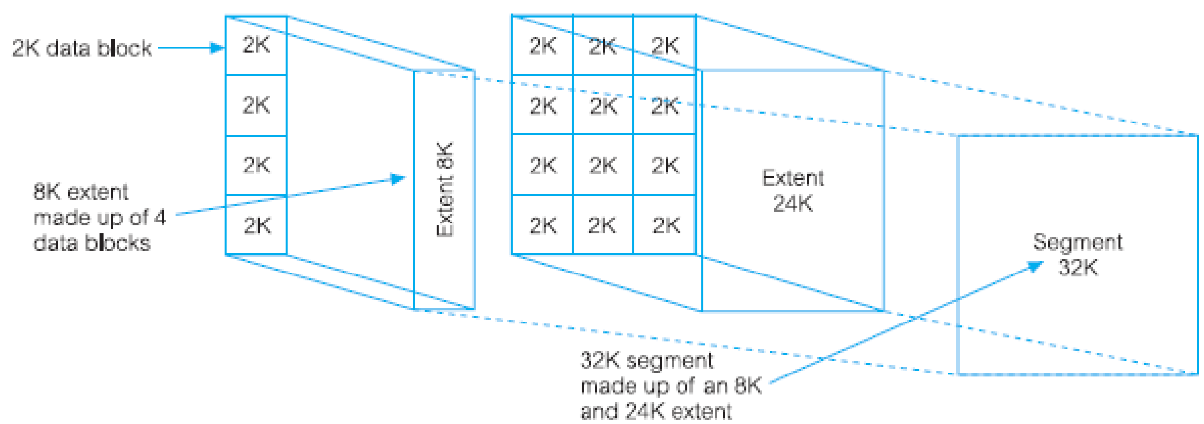

Figure 1.2: *Relationship between Oracle data blocks, extents, and segments.*

1.2 Oracle's Physical Database Structure

The main physical database structures in Oracle are datafiles, redo log files, and control files.

Datafiles

Every Oracle database has one or more physical datafiles. The data of logical database structures (such as tables and indexes) is physically stored in these datafiles. One or more datafiles form a tablespace. The simplest Oracle database would have one tablespace and one datafile. A more complex database might have four tablespaces, each consisting of two datafiles, giving a total of eight datafiles.

Redo log files

Every Oracle database has a set of two or more redo log files that record all changes made to data for recovery purposes. Should a failure prevent modified data from being permanently written to the datafiles, the changes can be obtained from the redo log, thus preventing work from being lost.

Control files

Every Oracle database has a control file that contains a list of all the other files that make up the database, such as the datafiles and redo log files. For added protection, it is recommended that the control file should be multiplexed (multiple copies may be written to multiple devices). Similarly, it may be advisable to multiplex the redo log files as well.

The Oracle instance

The Oracle instance consists of the Oracle processes and shared memory required to access information in the database. The instance is made up of the Oracle background processes, the user processes, and the shared memory used by these processes. Among other things, Oracle uses shared memory for caching data and indexes as well as storing shared program code. Shared memory is broken into various memory structures, of which the basic ones are the System Global Area (SGA) and the Program Global Area (PGA).

- *System global area.* The SGA is an area of shared memory that is used to store data and control information for one Oracle instance. The SGA is allocated when the Oracle instance starts and deallocated when the Oracle instance shuts down. The information in the SGA consists of the following memory structures, each of which has a fixed size and is created at instance startup:
    - *Database buffer cache*. This contains the most recently used data blocks from the database. These blocks can contain modified data that has not yet been written to disk (dirty blocks), blocks that have not been modified, or blocks that have been written to disk since modification (clean blocks). By storing the most recently used blocks, the most active buffers stay in memory to reduce I/O and improve performance.

- *Redo log buffer.* This contains the redo log file entries, which are used for recovery purposes. The background process LGWR (explained shortly) writes the redo log buffer to the active online redo log file on disk.
- *Shared pool.* This contains the shared memory structures, such as shared SQL areas in the library cache and internal information in the data dictionary. The shared SQL areas contain parse trees and execution plans for the SQL queries. If multiple applications issue the same SQL statement, each can access the shared SQL area to reduce the amount of memory needed and to reduce the processing time used for parsing and execution. We discuss query processing in Chapter 23.
- *Large pool.* This is an optional memory area intended for large memory allocations (eg. for buffers for Recovery Manager (RMAN) I/O slaves).
- Java pool. This area stores all session-specific Java code and data within the Java Virtual Machine (JVM).
- *Streams pool.* This area stores buffered queue messages and provides memory for Oracle Streams processes. Oracle Streams allows information flow (such as database events and database changes) to be managed and potentially propagated to other databases.
- *Fixed SGA.* This is an internal housekeeping area that contains various data such as general information about the state of the database and the Oracle instance and information communicated between Oracle processes, such as information about locks.

* Program global area. The PGA is an area of shared memory that is used to store data and control information for an Oracle process. The PGA is created by Oracle Database when an Oracle process is started. One PGA exists for each server process and background process. The size and content of the PGA depends on the Oracle server options installed.
* Client processes. Each client process represents a user's connection to the Oracle server (for example, through SQL*Plus or an Oracle Forms application). The user process manipulates the user's input, communicates with the Oracle server process, displays the information requested by the user and, if required, processes this information into a more useful form.
* Oracle processes. Oracle (server) processes perform functions for users. Oracle processes can be split into two groups: server processes (which handle requests from connected user processes) and background processes (which perform asynchronous I/O and provide increased parallelism for improved performance and reliability). This are the following background processes:
    - *Database Writer (DBWR).* The DBWR process is responsible for writing the modified (dirty) blocks from the buffer cache in the SGA to datafiles on disk. An Oracle instance can have up to ten DBWR processes, named DBW0 to DBW9, to handle I/O to multiple datafiles. Oracle employs a technique known as write-ahead logging, which means that the DBWR process performs batched writes whenever the buffers need to be freed, not necessarily at the point the transaction commits.

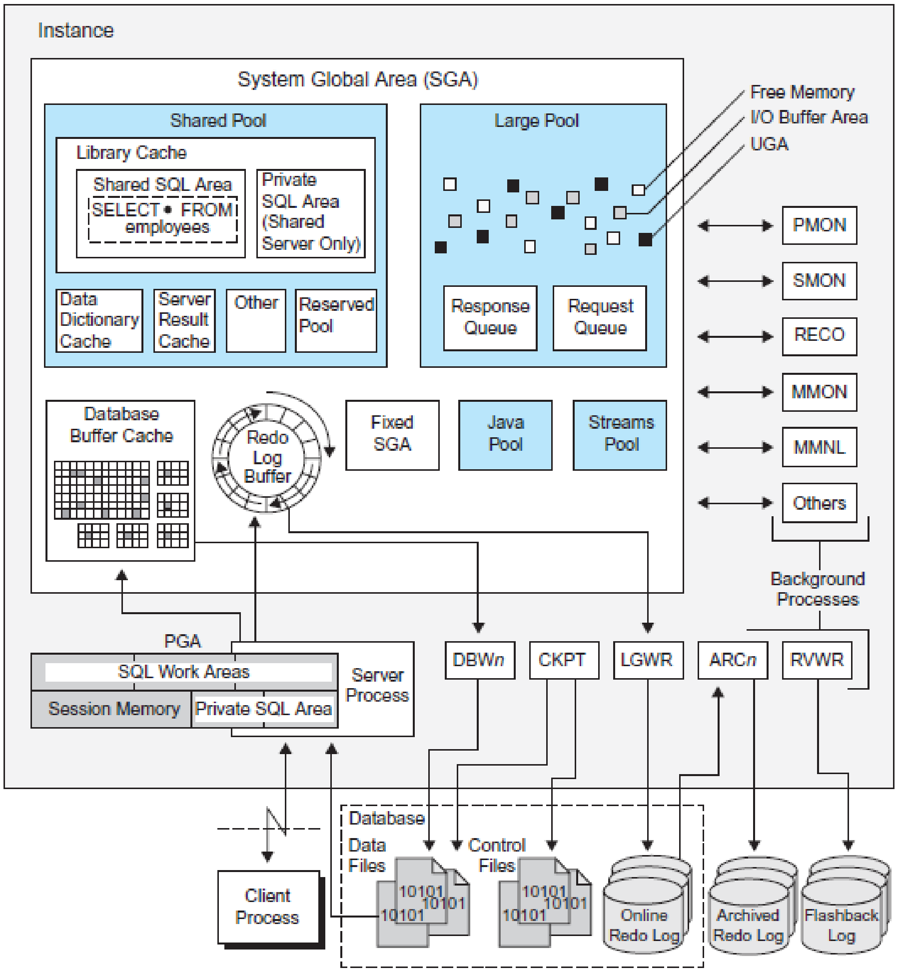

Figure 1.3: *The Oracle architecture (from the Oracle documentation set).*

- *Log Writer (LGWR).* The LGWR process is responsible for writing data from the log buffer to the redo log.
- *Checkpoint (CKPT).* A checkpoint is an event in which all modified database buffers are written to the datafiles by the DBWR. The CKPT process is responsible for telling the DBWR process to perform a checkpoint and to update all the datafiles and control files for the database to indicate the most recent checkpoint. The CKPT process is optional and, if omitted, these responsibilities are assumed by the LGWR process.
- *System Monitor (SMON).* The SMON process is responsible for crash recovery when the instance is started following a failure. This includes recovering transactions that have died because of a system crash. SMON also defragments the database by merging free extents within the datafiles.
- *Process Monitor (PMON).* The PMON process is responsible for tracking user processes that access the database and recovering them following a crash. This

includes cleaning up any resources left behind (such as memory) and releasing any locks held by the failed process.

- *Archiver (ARCH).* The ARCH process is responsible for copying the online redo log files to archival storage when they become full. The system can be configured to run up to 10 ARCH processes, named ARC0 to ARC9. The additional archive processes are started by the LWGR when the load dictates.
- *Recoverer (RECO).* The RECO process is responsible for cleaning up failed or suspended distributed transactions.
- *Flashback Writer or Recovery Writer (RVWR).* When flashback is enabled or when there are guaranteed restore points, the RVWR process writes flashback data to flashback database logs in the flash recovery area. Flashback tools allow administrators and users to view and manipulate past states of an Oracle instance's data without recovering the database to a fixed point in time.
- *Manageability Monitor (MMON)*. This process performs many tasks related to the *Automatic Workload Repository (AWR)*. AWR is a repository of historical performance data that includes cumulative statistics for the system, sessions, individual SQL statements, segments, and services. Among other things, by default, the MMON process gathers statistics every hour and creates an AWR snapshot.
- *Manageability Monitor Lite (MMNL)*. This process writes statistics from the Active Session History (ASH) buffer in the SGA to disk. MMNL writes to disk when the ASH buffer is full.

1.3 Automatic Storage Management (ASM)

REFERENCES:

Authors, C., Abraham, J., Bagal, P., Bauer, M., Belden, E., Bridge, B., … Nelson, B. (2012). Oracle ® Automatic Storage Management Administrator ' s Guide. 2(February). Retrieved from https://docs.oracle.com/cd/E11882_01/server.112/e18951.pdf

Connolly, T., & Begg, C. (2015). Database Systems (Sixth Edit; M. Horton, M. Goldstein, L. Dent, K. K. Neelakantan, & S. Disanno, eds.). Pearson Education Limited.