

# CSE 4409: Database Management Systems II

Dr. Abu Raihan Mostofa Kamal

December 8, 2021



# Course Outline

**Pre-requisite:** **CSE 4307** (Database Management Systems)

## Syllabus: Part II

I. Relational Tables and storage mapping. Introduction to Programmable SQL (or Similar), Basic constructs, Composite Data-types, Functions and Procedures, Cursor, Triggers, Package, Dynamic SQL, Exception Handling, Data dictionary, Collection.



# Storage: Tables and objects in general

So far,

Upto now the user has **no idea where** the object is stored and **how** the storage is maintained. Now we will explore how a database designer can **control** it precisely.



# Tablespace, Datafiles and Objects

The following block-diagram shows the relationship.

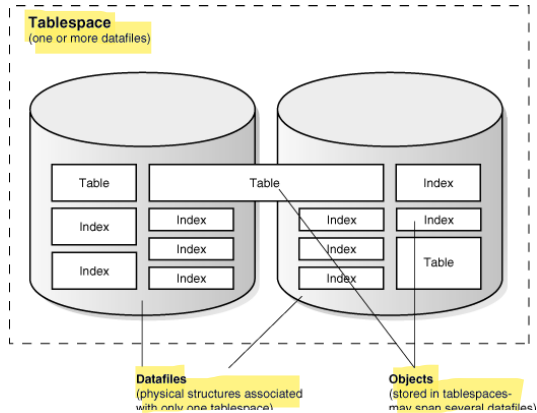


Figure 1: Tablespace, Datafiles and Objects



## Tablespace, Datafiles and Objects (Cont.)

Although databases, tablespaces, datafiles, and segments are **closely related**, they have important **differences**:

- **Databases and tablespaces.** An Oracle database is comprised of one or more **logical storage units** called **tablespaces**. The database also has a lot more (background process). The database's data is collectively stored in the database's tablespaces.
- **Tablespaces and datafiles.** Each tablespace in an Oracle database is comprised of one or more **operating system files** called **datafiles**. A tablespace's datafiles physically store the associated database data on disk.



## Tablespace, Datafiles and Objects (Cont. 2)

Although databases, tablespaces, datafiles, and segments are **closely related**, they have important **differences**:

- **Databases and datafiles.** A database's data is collectively stored in the datafiles that constitute each tablespace of the database. For example, the simplest Oracle database would have one tablespace and one datafile. A more complicated database might have three tablespaces, each comprised of two datafiles (for a total of six datafiles).
- **Tablespace.** Tablespaces are the **bridge** between certain physical and logical components of the Oracle database.

(Tablespaces are where you store Oracle database objects such as tables, indexes and rollback segments. [A Rollback Segment is a database object containing before-images of data written to the database. Rollback segments are used to: i) Undo changes when a transaction is rolled back ii) Recover the database to a consistent state in case of failures ] )



# Basic Operations on Tablespace

Basics of Space Management: Introduction to Data Blocks, Extents, and Segments:

- **Data blocks.** At the finest level of granularity, Oracle stores data in data blocks (also called logical blocks, Oracle blocks, or pages). One data block corresponds to a specific number of bytes of physical database space on disk.
- **Extent.** The next level of logical database space is an extent. An extent is a specific number of contiguous data blocks allocated for storing a specific type of information.
- **Segment.** The level of logical database storage above an extent is called a segment. A segment is a set of extents.

Note:

*Oracle allocates space for segments in units of one **extent**. When the existing extents of a segment are full, Oracle allocates another extent for that segment.*



# Blocks, Extents and Segments

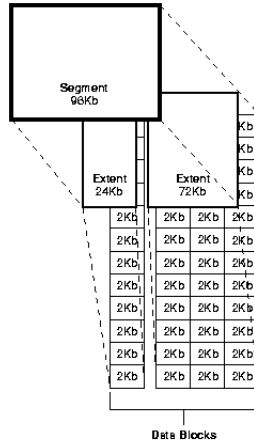


Figure 2: Oracle Storage Management





# Mapping between user, object and tablespace: Example

## Step 1: Create a Tablespace first.

```
1 CREATE TABLESPACE mytspace  
2 DATAFILE '/u02/oracle/data/lmtbsb01.dbf' SIZE 50M  
3 EXTENT MANAGEMENT LOCAL AUTOALLOCATE;  
4
```

**AUTOALLOCATE** causes the tablespace to be system managed with a minimum extent size of 64K.



## Mapping between user, object and tablespace: Example (Cont.)

**Step 2 (a):** Create an user and assign that user to a specific tablespace.

```
1  
2 CREATE USER iutlearner  
3 IDENTIFIED BY test123  
4 DEFAULT TABLESPACE mytspace;  
5  
6
```



## Mapping between user, object and tablespace: Example (Cont.)

**Step 2 (b):** Create a specific table and assign a tablespace with it (this will overrule previous).

```
1  
2 create table students  
3 (ID number primary key,  
4 Name varchar2(50),  
5 DOB date,  
6 Program varchar2(30)  
7 ) tablespace NEW_TBSPACE;  
8
```

- Use *DBA\_FREE\_SPACE* data-dictionary to find out the free space for a tablespace.
- Use *ALL\_TABLES* data-dictionary to find information of each table along with its tablespace.



## Mapping between user, object and tablespace: Example (Cont.)

**Step 3:** How to get information about free available space for a tablespace. Use DBA\_FREE\_SPACE data-dictionary.

```
1  SELECT TABLESPACE_NAME ,  
2  SUM(BYTES)/1024/1024/1024 "FREE SPACE(GB)"  
3  FROM DBA_FREE_SPACE GROUP BY TABLESPACE_NAME;
```



## Tablespace: Adding space

You can add data-files to an existing tablespace:

```
1 ALTER TABLESPACE  
2 users  
3 ADD DATAFILE  
4 '/ora01/oracle/oradata/booktst_users_02.dbf '  
5 size 500m
```



# More Operations on Tablespace

- Online and Offline Tablespaces
- Read-Only Tablespaces

```
1 ALTER TABLESPACE mytspace READ ONLY; ---No insert,update,delete  
   allowed  
2 ALTER TABLESPACE mytspace READ WRITE;  
3  
4 ALTER TABLESPACE mytspace OFFLINE; --all data will be non-visible  
5 ALTER TABLESPACE mytspace ONLINE;  
6  
7  
8
```

