



Modelo Físico DBMS Oralce

AULA PL03

Hugo Peixoto

2018 – 2019 Universidade do Minho



Conteúdo da UC

<http://hpeixoto.github.io/nosql>

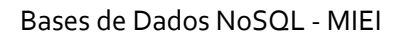


- Relational Database
- Oracle
- Tablespaces | Datafiles
- Manage Database Size
- Objects
- Create Table
- SQL: create table | insert and select statements



Relational Database

- A database that conforms to the relational model. The relational model has the following major aspects:
 - Structures
 - Well-defined objects store or access the data of a database.
 - Operations
 - Clearly defined actions enable applications to manipulate the data and structures of a database.
 - Integrity rules
 - Integrity rules govern operations on the data and structures of a database.



The diagram illustrates a database schema for protein analysis, featuring several interconnected tables:

- Alleles**: Contains columns for `nm_allele` (VARCHAR(20)), `nm_organism` (VARCHAR(45)), and an index.
- NetCTL**: Contains columns for `id_protein` (VARCHAR(20)), `ds_pep` (VARCHAR(20)), `ps_start` (INT(11)), `ps_end` (INT(11)), `nm_allele` (VARCHAR(20)), `vl_score` (FLOAT), and an index.
- NetMHC**: Contains columns for `id_protein` (VARCHAR(20)), `ds_pep` (VARCHAR(20)), `ps_start` (INT(11)), `ps_end` (INT(11)), `nm_allele` (VARCHAR(20)), `vl_score` (FLOAT), `ds_bind` (VARCHAR(20)), `ds_core` (VARCHAR(20)), and an index.
- NetMHCIi**: Contains columns for `id_protein` (VARCHAR(20)), `ds_pep` (VARCHAR(20)), `ps_start` (INT(11)), `ps_end` (INT(11)), `nm_allele` (VARCHAR(20)), `vl_score` (FLOAT), `ds_bind` (VARCHAR(20)), `ds_core` (VARCHAR(20)), and an index.
- Proteome**: Contains columns for `id_protein` (VARCHAR(20)), `ds_protein` (VARCHAR(120)), `tx_sequence` (TEXT), `nm_genome` (VARCHAR(20)), and an index.
- BepiPred**: Contains columns for `id_protein` (VARCHAR(20)), `ds_pep` (VARCHAR(20)), `ps_start` (INT(11)), `ps_end` (INT(11)), `tm_pep` (INT(11)), `vl_score` (FLOAT), `vl_sd` (FLOAT), and an index.
- WoLF PSORT**: Contains columns for `id_protein` (VARCHAR(20)), `nm_local` (VARCHAR(20)), `vl_score` (FLOAT), and an index.
- AAP12**: Contains columns for `id_protein` (VARCHAR(20)), `ds_pep` (VARCHAR(20)), `ps_start` (INT(11)), `ps_end` (INT(11)), `vl_score` (FLOAT), and an index.

Relationships are defined as follows:

- Alleles** has a one-to-many relationship with **NetCTL**, **NetMHC**, and **NetMHCIi**.
- Proteome** has a one-to-many relationship with **NetCTL**, **NetMHC**, **NetMHCIi**, **BepiPred**, **WoLF PSORT**, and **AAP12**.
- BepiPred** has a one-to-many relationship with **NetCTL**.
- WoLF PSORT** has a one-to-many relationship with **NetMHCIi**.
- AAP12** has a one-to-many relationship with **NetMHCIi**.



ORACLE®



Oracle Database

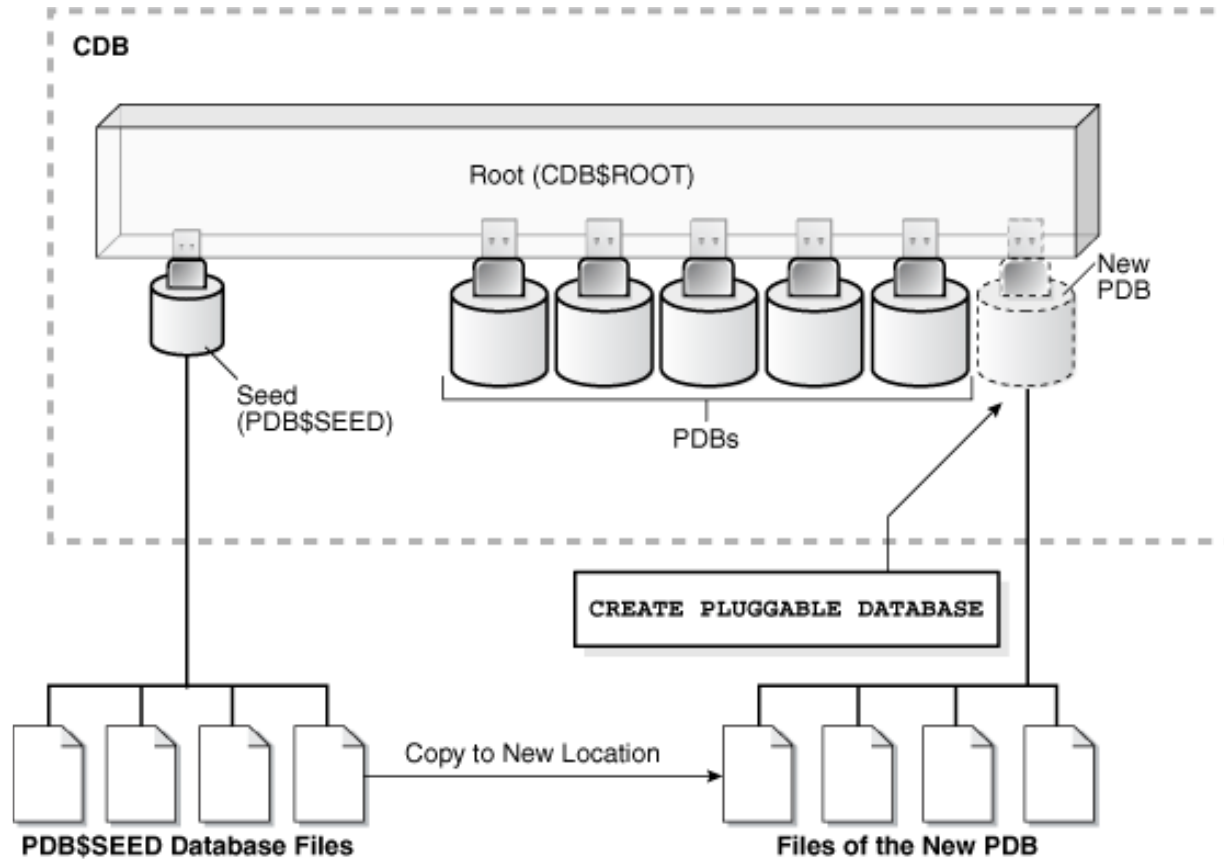
Schema Objects:

- A schema is a way to logically group objects in a single collection and provide a unique namespace for objects

User account + collection of all objects therein

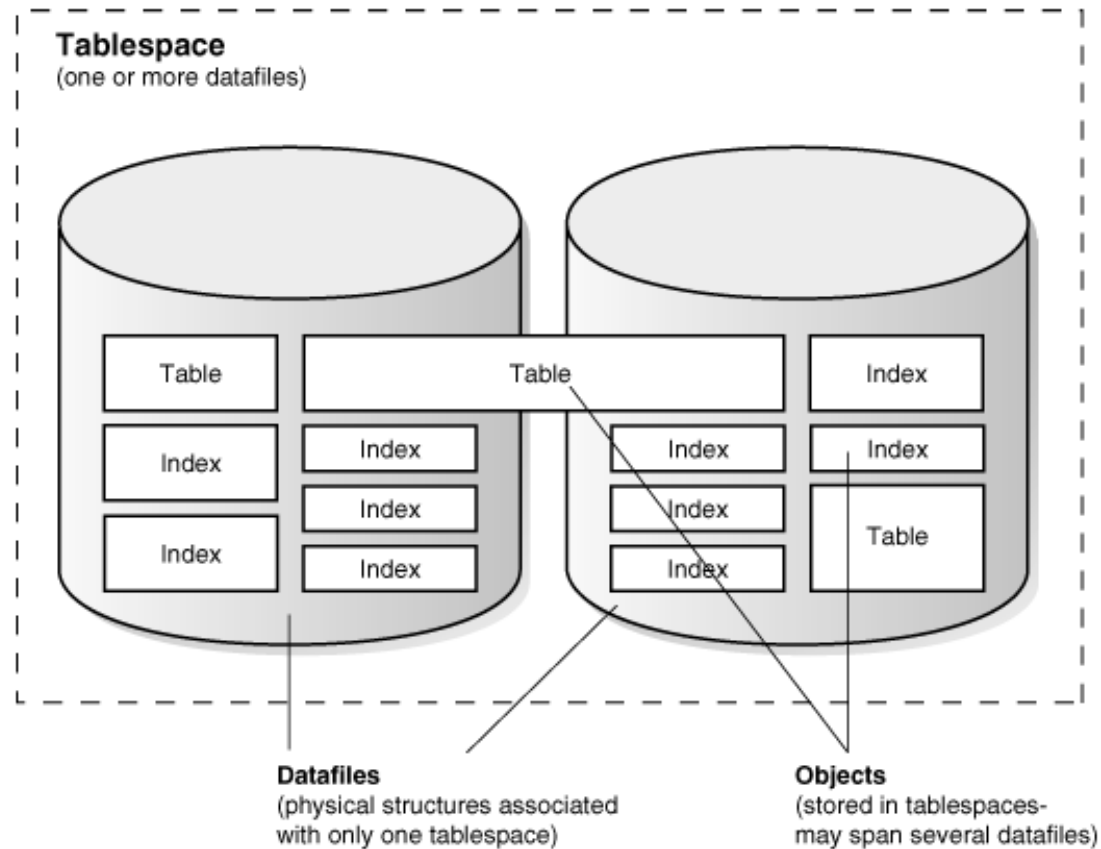


Oracle Database 12c





Tablespaces | Datafiles | Objects





TableSpaces | DataFiles

Databases, tablespaces, and datafiles are closely related, but they have important differences:

An Oracle database consists of one or more logical storage units called **tablespaces**, which collectively store all of the database's data

Each tablespace in an Oracle database consists of one or more files called **datafiles**, which are physical structures that conform to the operating system in which Oracle is running.

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.

Another database can have three tablespaces, each consisting of two datafiles (for a total of six datafiles).



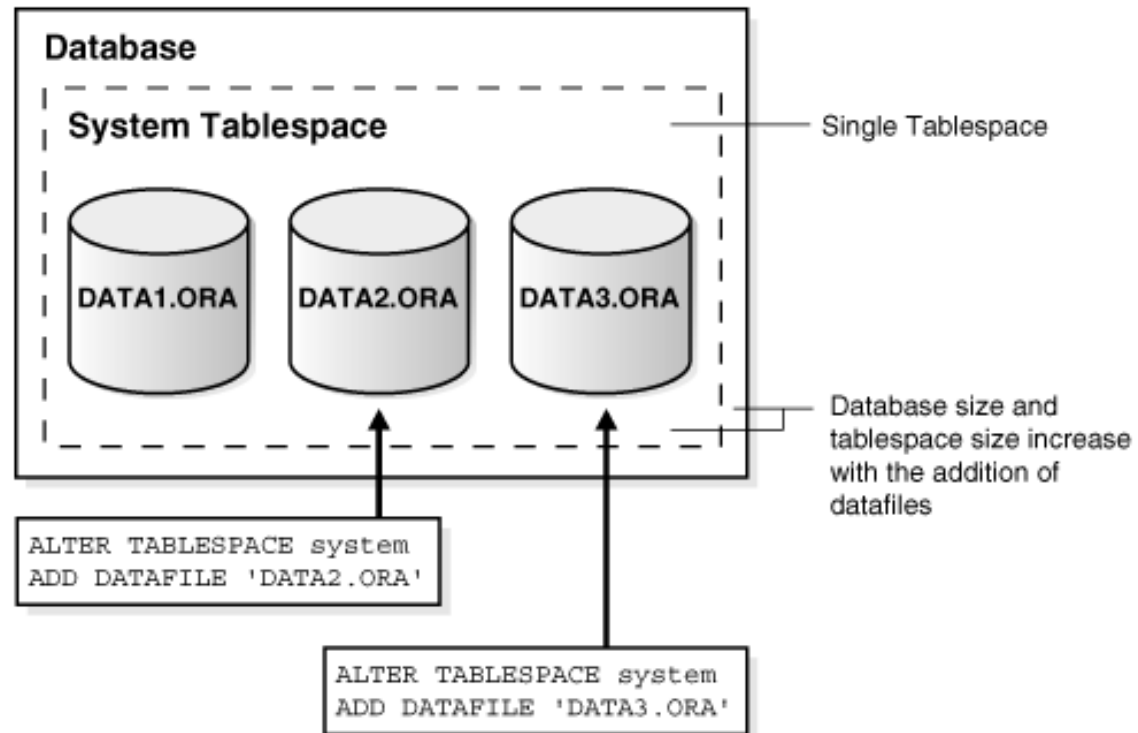
Database Size

You can enlarge a database in three ways:

- 1) Add a datafile to a tablespace
- 2) Add a new tablespace
- 3) Increase the size of a datafile

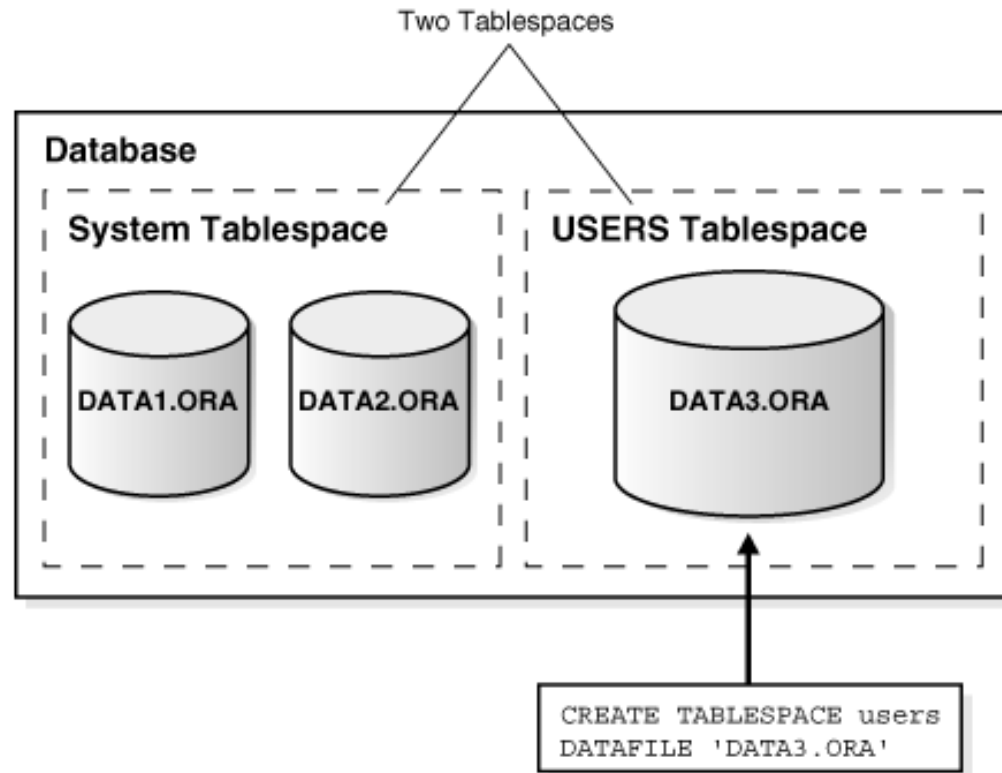


Database Size : Add datafile



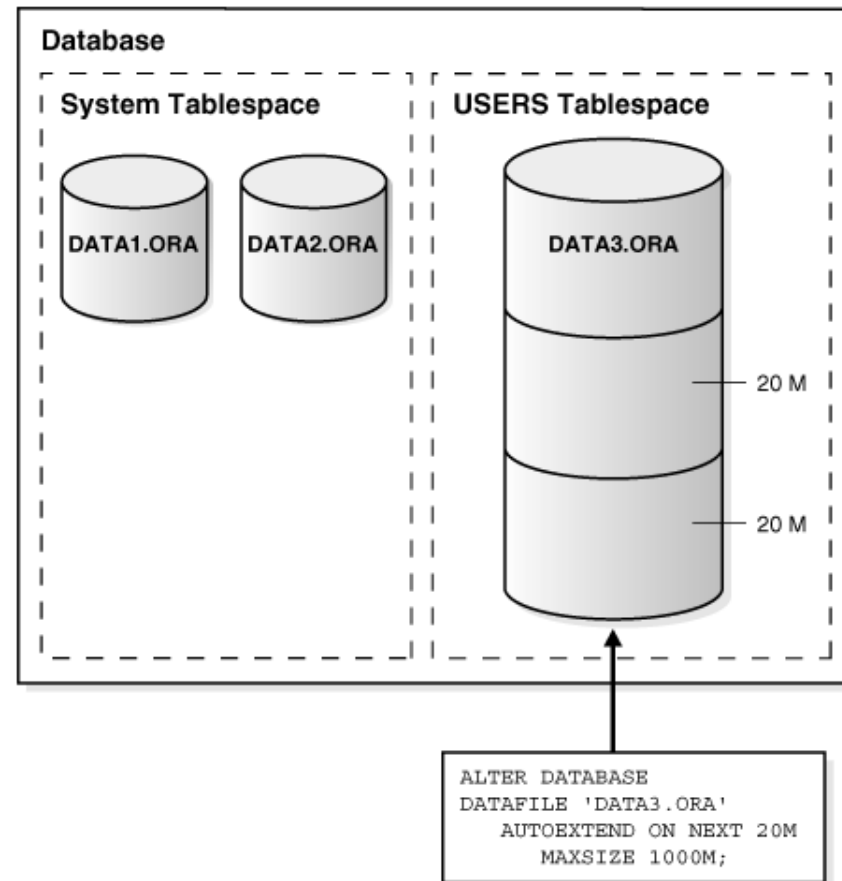


Database Size : New Tablespace





Database Size : Datafile size





Objects

- Tables;
- Views;
- Materialized Views;
- Dimensions;
- Sequences;
- Synonyms;
- Indexes;
- Databaselinks;
- Stored Procedures;
- ...



Objects: Tables

Diagram illustrating a table structure with annotations:

- Rows**: Points to the vertical axis of the table.
- Columns**: Points to the horizontal axis of the table.
- Column names**: Points to the header row of the table.
- Column not allowing nulls**: Points to the **DEPTNO** column.
- Column allowing nulls**: Points to the **COMM** column.

	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7329	SMITH	CLERK	7902	17-DEC-88	800.00	300.00	20
7499	ALLEN	SALESMAN	7698	20-FEB-88	1600.00	300.00	30
7521	WARD	SALESMAN	7698	22-FEB-88	1250.00	500.00	30
7566	JONES	MANAGER	7839	02-APR-88	2975.00		20



Objects: Views

Base Table

employees						
employee_id	last_name	job_id	manager_id	hire_date	salary	department_id
203	marvis	hr_rep	101	07-Jun-94	6500	40
204	baer	pr_rep	101	07-Jun-94	10000	70
205	higgins	ac_rep	101	07-Jun-94	12000	110
206	gietz	ac_account	205	07-Jun-94	8300	110

View

staff				
employee_id	last_name	job_id	manager_id	department_id
203	marvis	hr_rep	101	40
204	baer	pr_rep	101	70
205	higgins	ac_rep	101	110
206	gietz	ac_account	205	110



Objects: Sequences

Sequence numbers are Oracle integers of up to 38 digits defined in the database.

A sequence definition indicates general information, such as the following:

- The name of the sequence

- Whether the sequence ascends or descends

- The interval between numbers

- Whether Oracle should cache sets of generated sequence numbers in memory



Objects: Synonyms

A synonym is an alias for any table, view, materialized view, sequence, procedure, function, package, type, Java class schema object, user-defined object type, or another synonym.

Synonyms are often used for security and convenience. For example, they can do the following:

- Mask the name and owner of an object
- Provide location transparency for remote objects of a distributed database
- Simplify SQL statements for database users
- Enable restricted access similar to specialized views when exercising fine-grained access control



Objects: Indexes

Indexes are optional structures associated with tables.

You can create indexes on one or more columns of a table to speed SQL statement execution on that table.



SQL: Create Table

Example:

```
CREATE TABLE EDITORA (  
    "ID_EDITORA" NUMBER(3,0) NOT NULL ENABLE,  
    "NOME" VARCHAR2(200 BYTE) NOT NULL ENABLE,  
    CONSTRAINT "EDITORA_PK" PRIMARY KEY ("ID_EDITORA")  
);
```



SQL: Insert statment

Example:

```
> insert into review values (7,7, to_date('19-12-2017', 'dd-mm-yyyy'), 'MAU');
```



SQL: Select statment

Example:

```
> select count(*) from review;
```



Modelo Físico DBMS Oralce

AULA PL03

Hugo Peixoto

2018 – 2019 Universidade do Minho