

# Lenguaje SQL

**Introducción**

**Estructura del lenguaje SQL**

# Introducción



¿Qué es una BBDD?



Diseño



Arquitectura BBDD



Estructura BBDD Oracle



Configuración Conexión



Aplicaciones



SQL

# ¿Qué es una Base de Datos?

Conjunto de datos estructurado según un determinado modelo de datos y almacenado en soporte informático, al que tienen acceso personas y aplicaciones. (diccionario.raing.es)

00220

AFFORESTATION - D (PL)

AFGHANISTAN - A

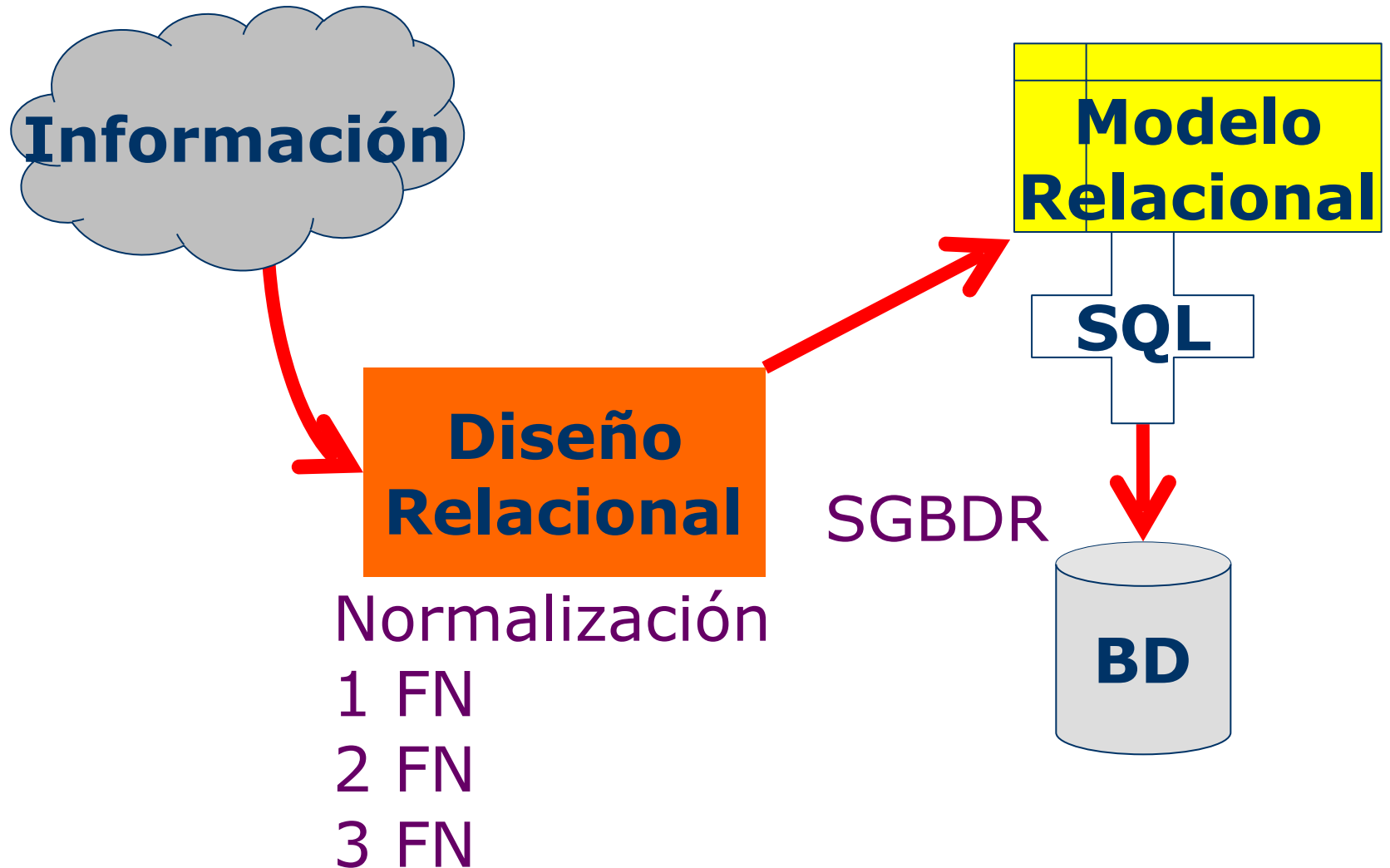
# Diseño de BBDD



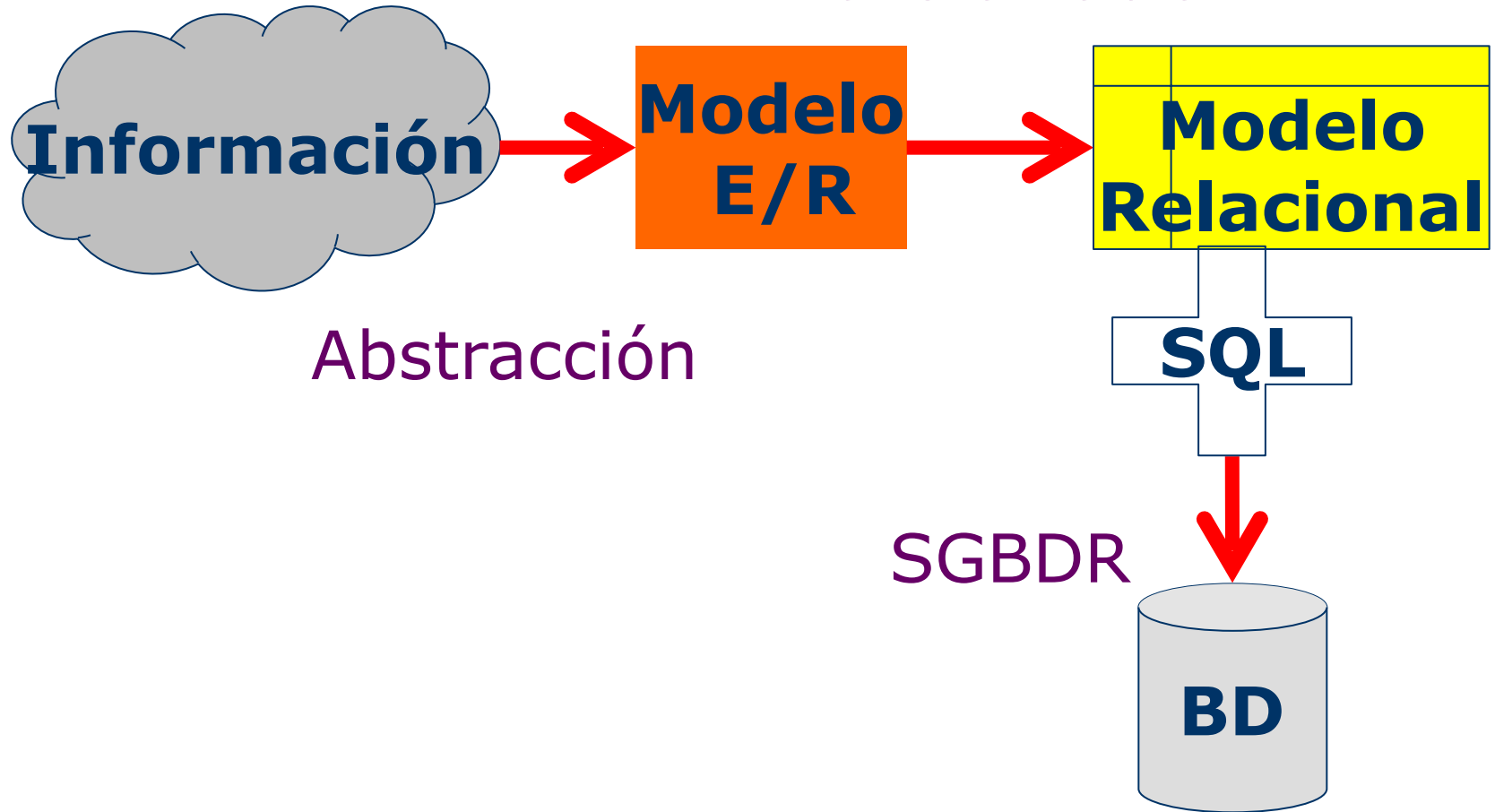






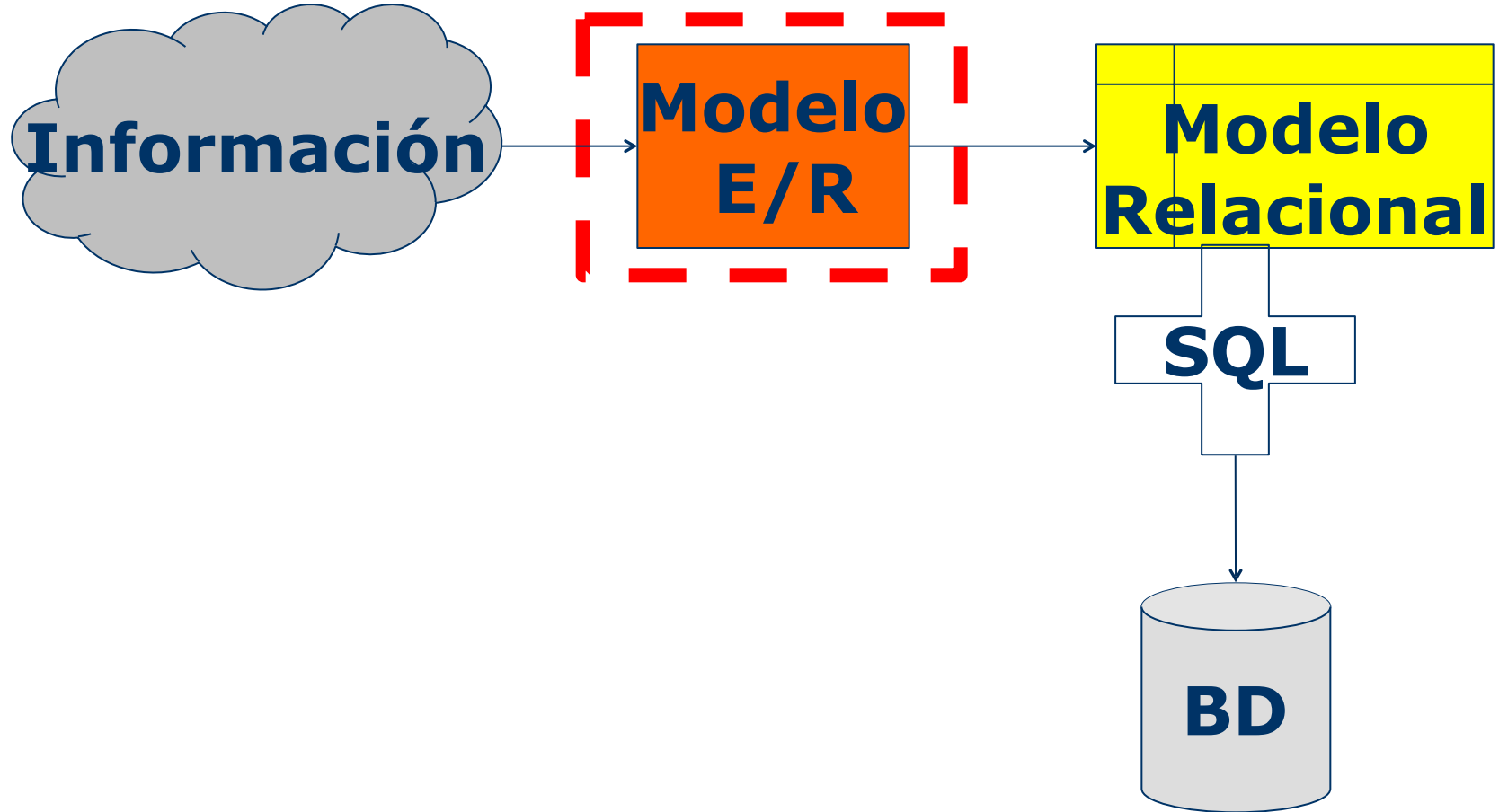


## Paso a Tablas

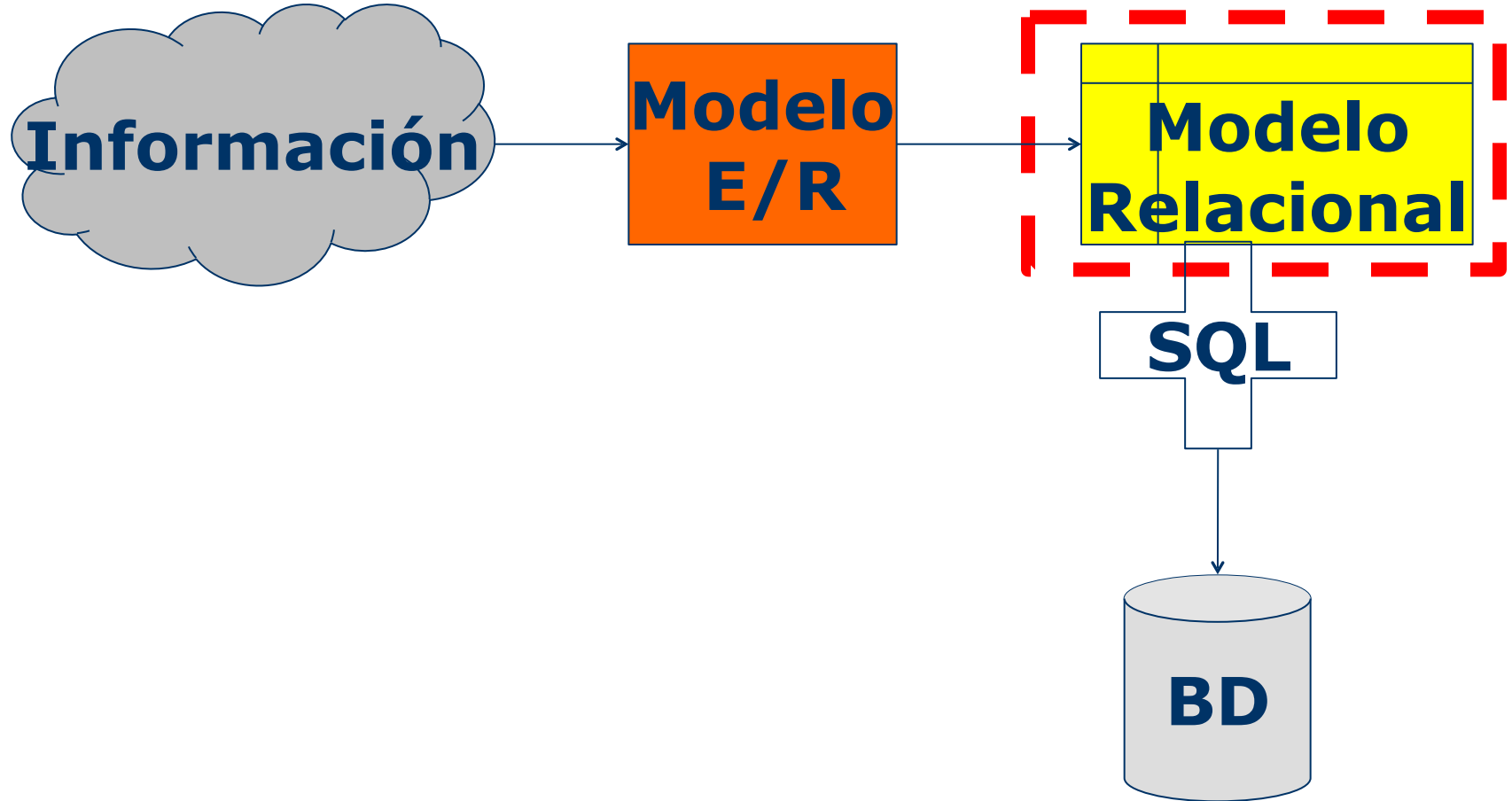




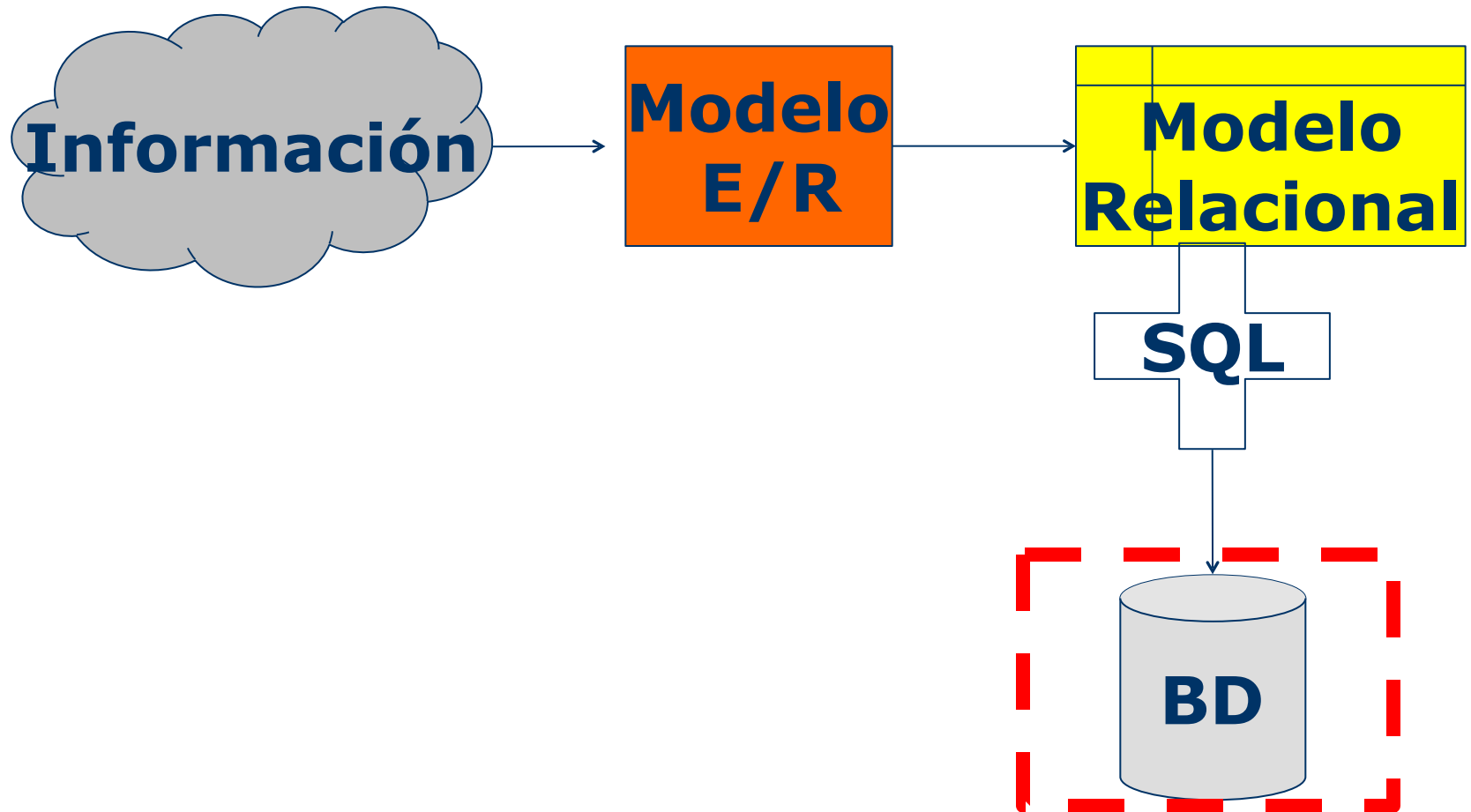
# Nivel Conceptual



# Nivel Lógico



# Nivel Físico



# Nivel Conceptual

Se desea guardar información de los departamentos, nombre y localidad.

De los empleados: nombre, salario, trabajo, y el jefe.

Un empleado está solo en 1 departamento.

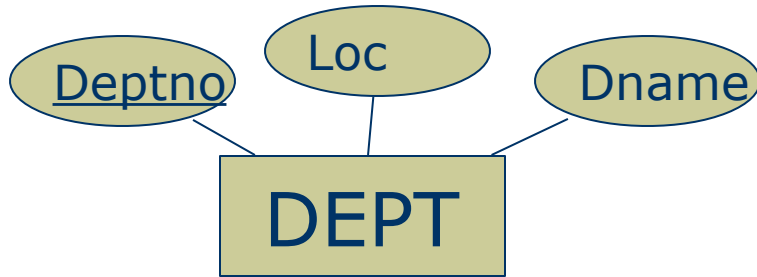
En un departamento hay varios empleados.

Un empleado sólo tienen 1 jefe.

Un jefe, puede serlo de varios empleados.

---

# Modelo Entidad-Relación



Se desea guardar información de los departamentos, nombre y localidad.

De los empleados: nombre, salario, trabajo, y el jefe.

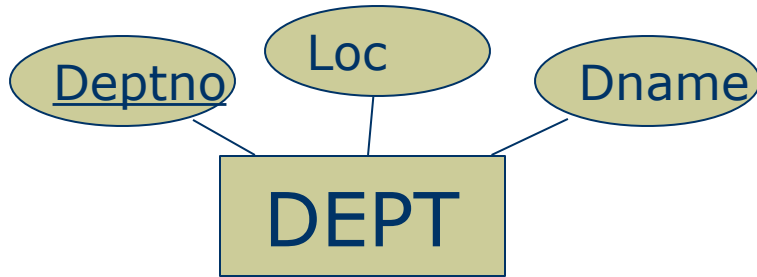
Un empleado está solo en 1 departamento.

En un departamento hay varios empleados.

Un empleado sólo tienen 1 jefe.

Un jefe, puede serlo de varios empleados.

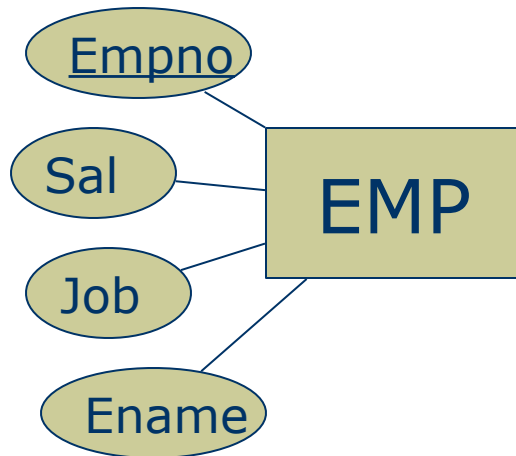
# Modelo Entidad-Relación



Se desea guardar información de los departamentos, nombre y localidad.

De los empleados: nombre, salario, trabajo, y el jefe.

Un empleado está solo en 1 departamento.

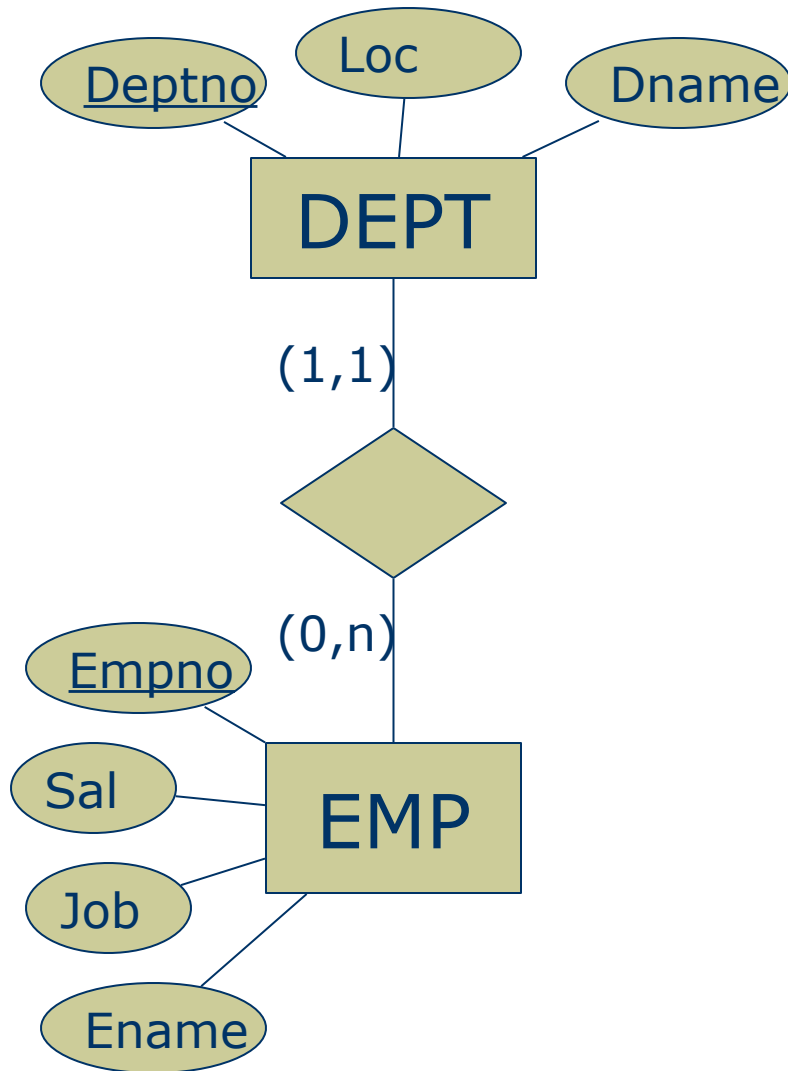


En un departamento hay varios empleados.

Un empleado sólo tienen 1 jefe.

Un jefe, puede serlo de varios empleados.

# Modelo Entidad-Relación



Se desea guardar información de los departamentos, nombre y localidad.

De los empleados: nombre, salario, trabajo, y el jefe.

Un empleado está solo en 1 departamento.

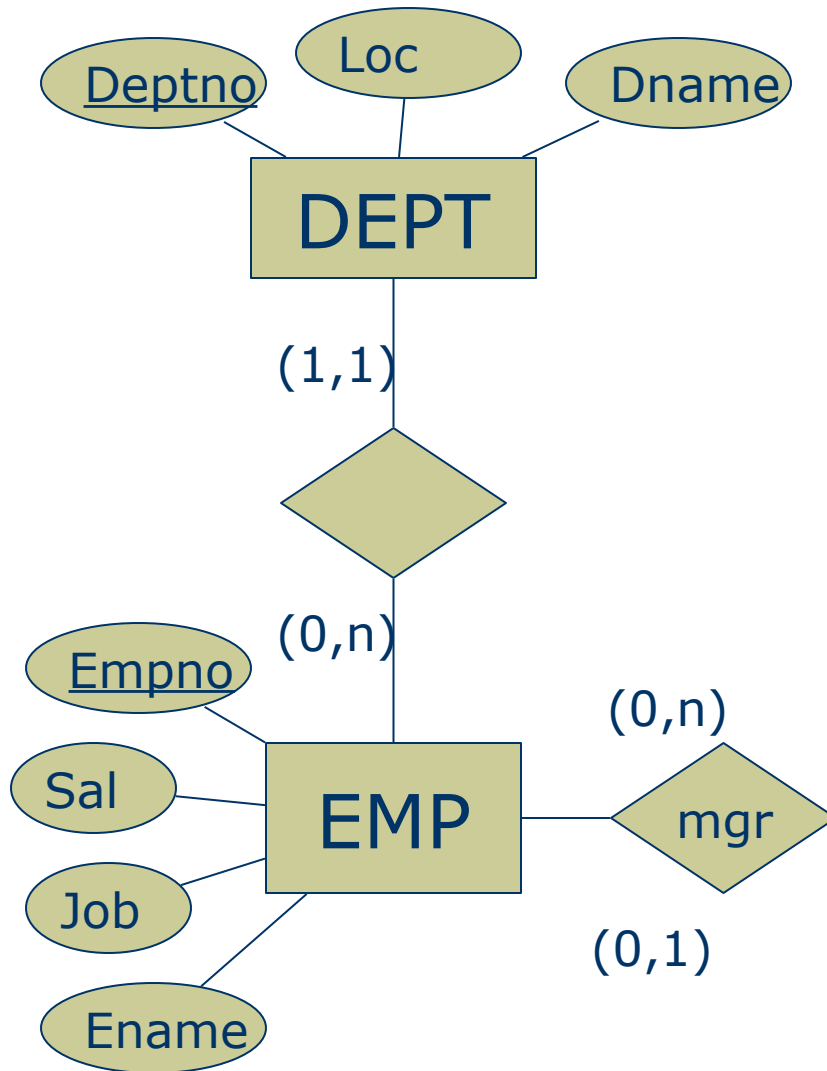
En un departamento hay varios empleados.

Un empleado sólo tienen 1 jefe.

Un jefe, puede serlo de varios empleados.



# Modelo Entidad-Relación



Se desea guardar información de los departamentos, nombre y localidad.

De los empleados: nombre, salario, trabajo, y el jefe.

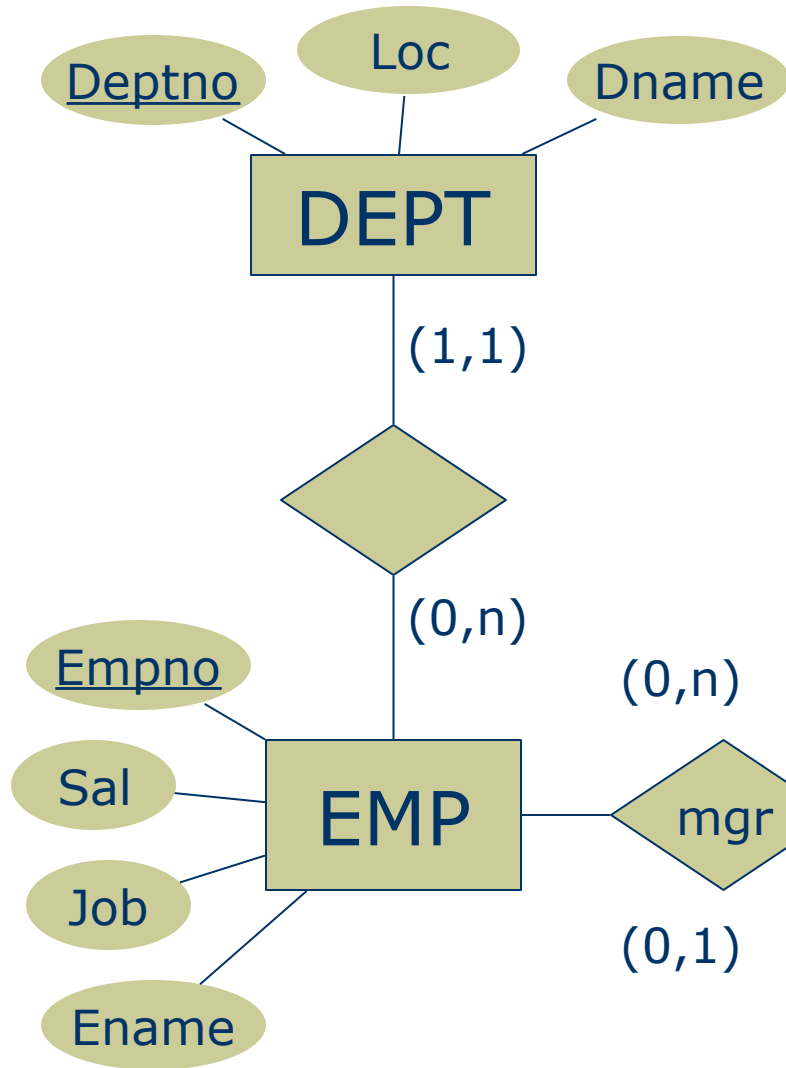
Un empleado está solo en 1 departamento.

En un departamento hay varios empleados.

Un empleado sólo tienen 1 jefe.

Un jefe, puede serlo de varios empleados.

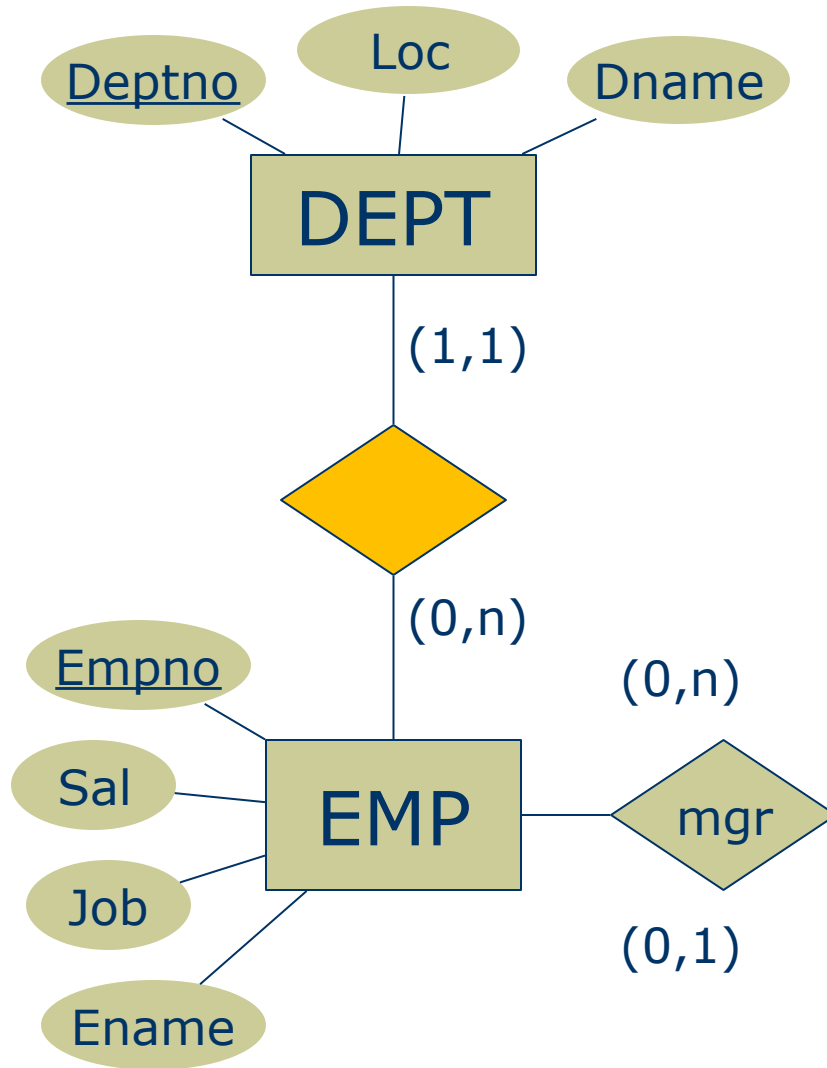
# Modelo Relacional



Deptno	Dname	Loc
10		
20		
30		
40		

Empno	Sal	...
7878		
7879		
7979		
7576		
7675		

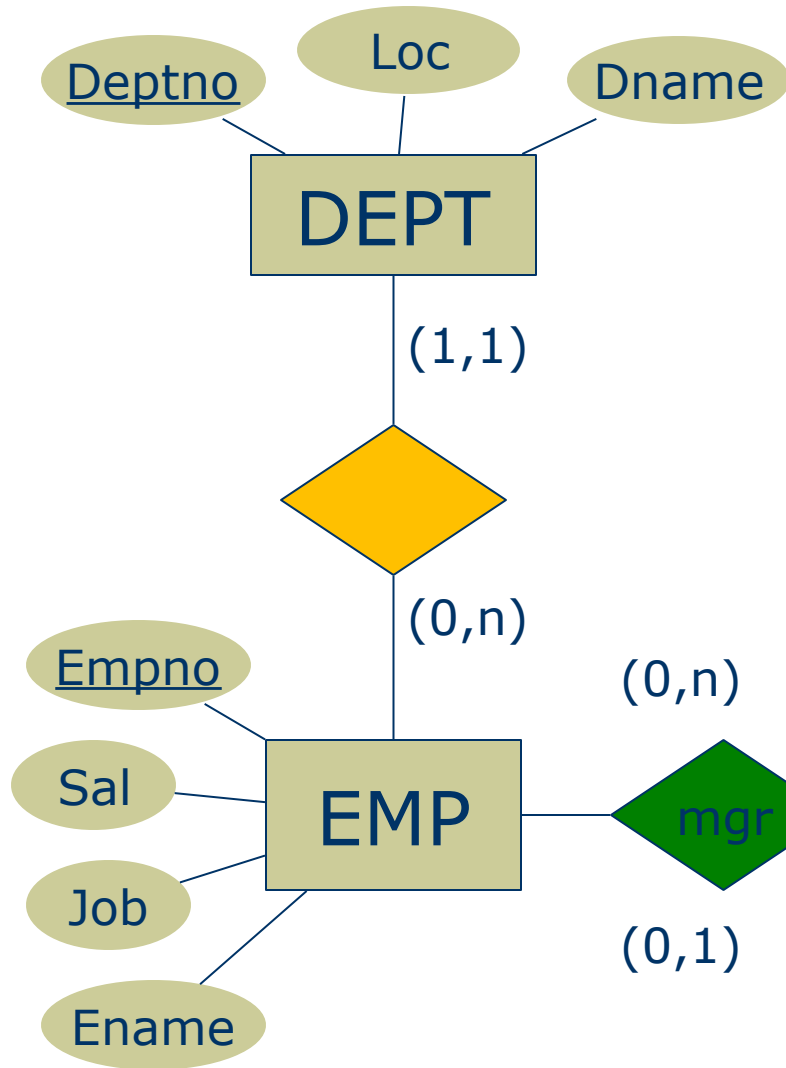
# Modelo Relacional



Deptno	Dname	Loc
10		
20		
30		
40		

Empno	Sal	...	Deptno
7878			10
7879			20
7979			10
7576			30
7675			

# Modelo Relacional



Deptno	Dname	Loc
10		
20		
30		
40		

Empno	Sal	...	Deptno	Mgr
7878			10	
7879			20	7878
7979			10	7878
7576			30	7979
7675				7675

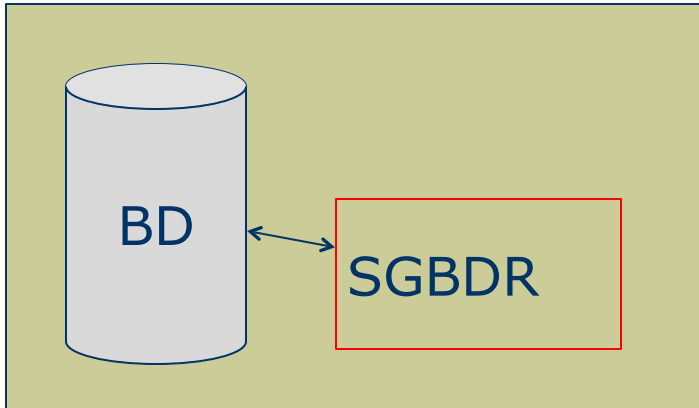
# Arquitectura



## Servidor Base de Datos



## Servidor Base de Datos



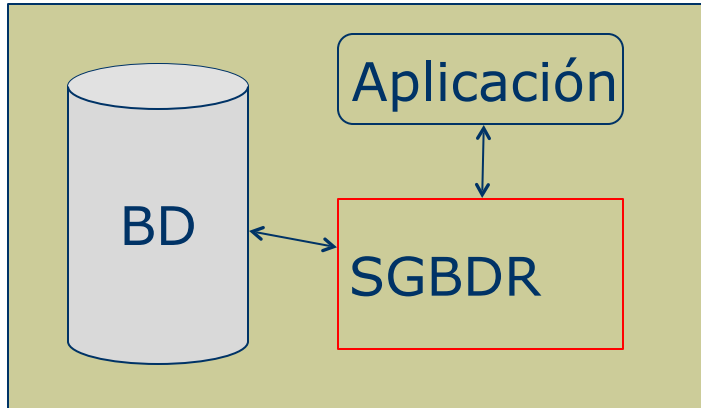
Sistema gestor de base de datos:

Oracle  
MySQL  
DB2  
SQL Server

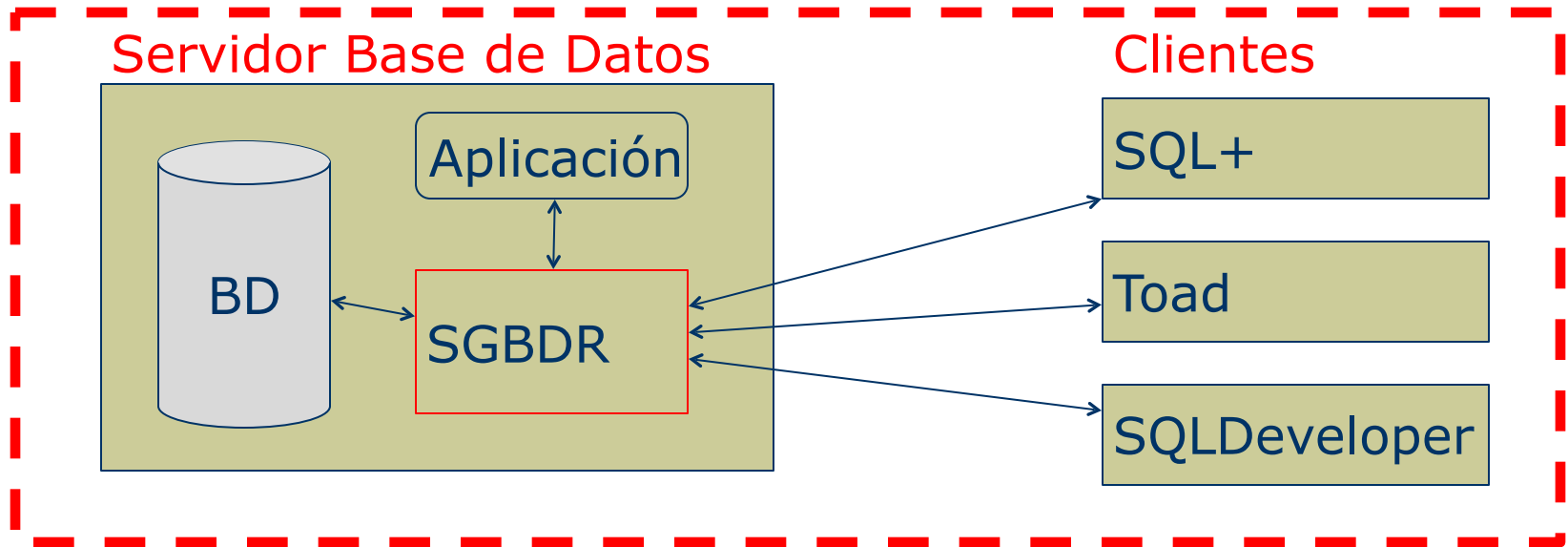


# Servidor

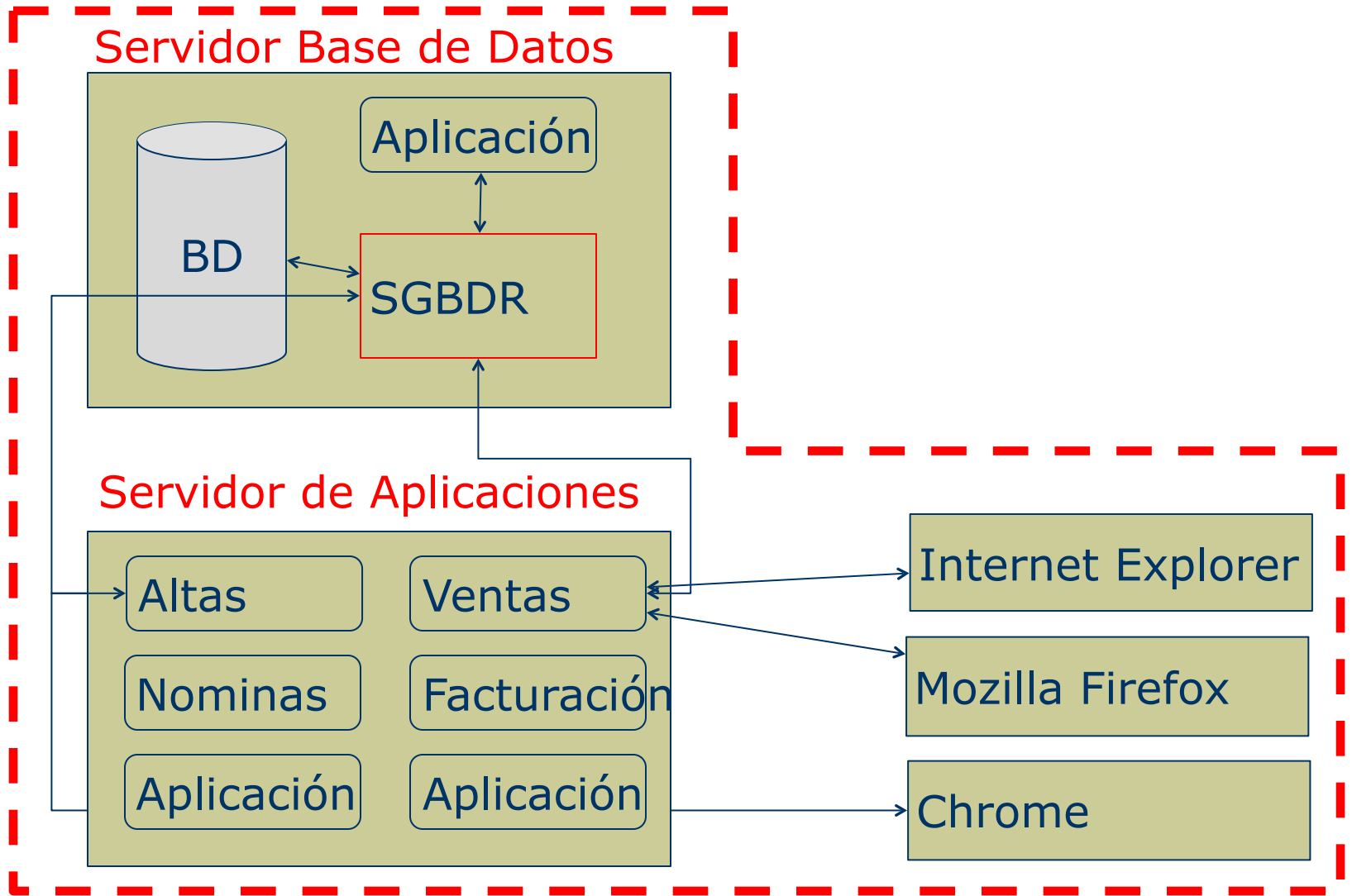
## Servidor Base de Datos



# Cliente-Servidor



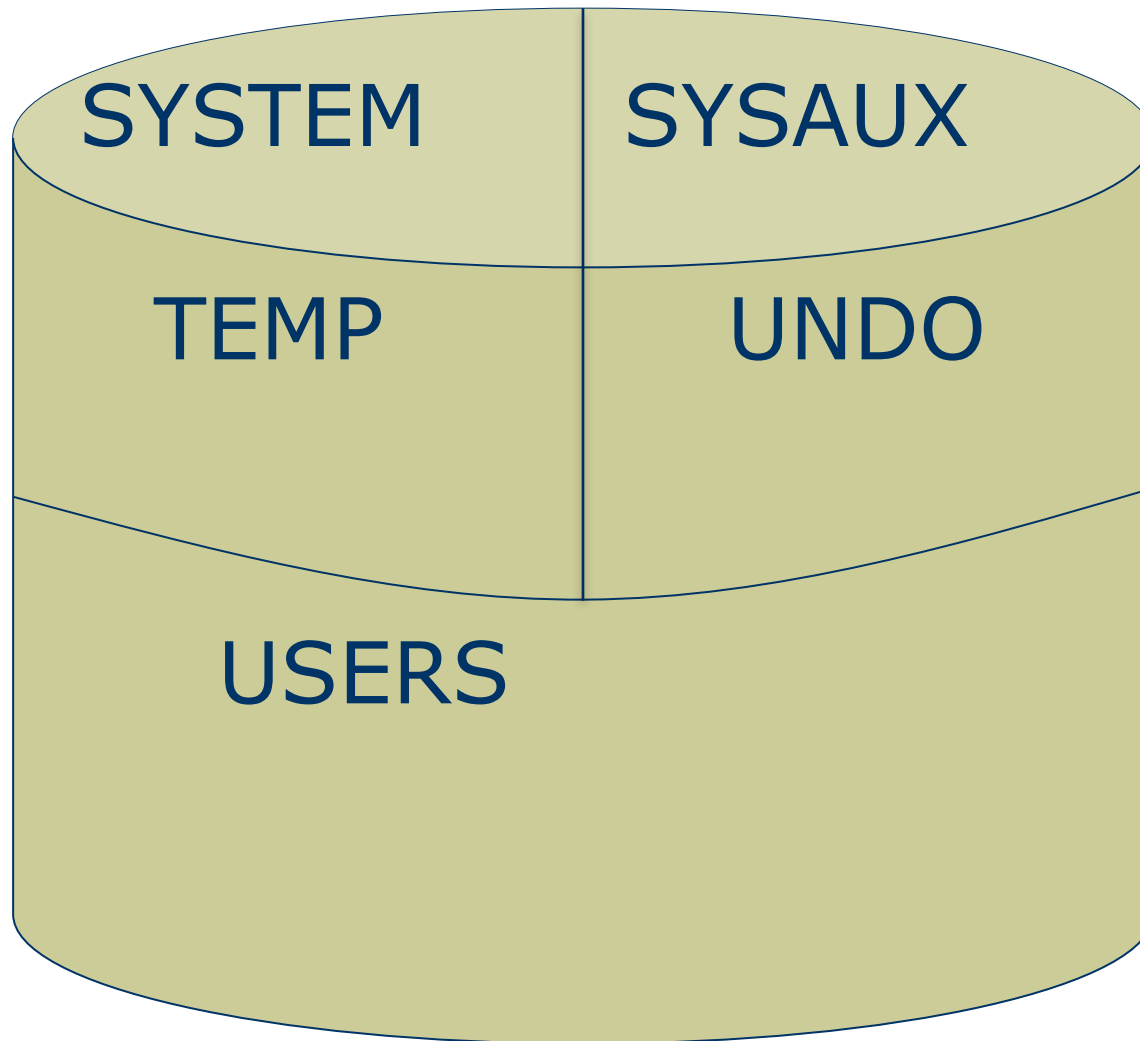
# Web



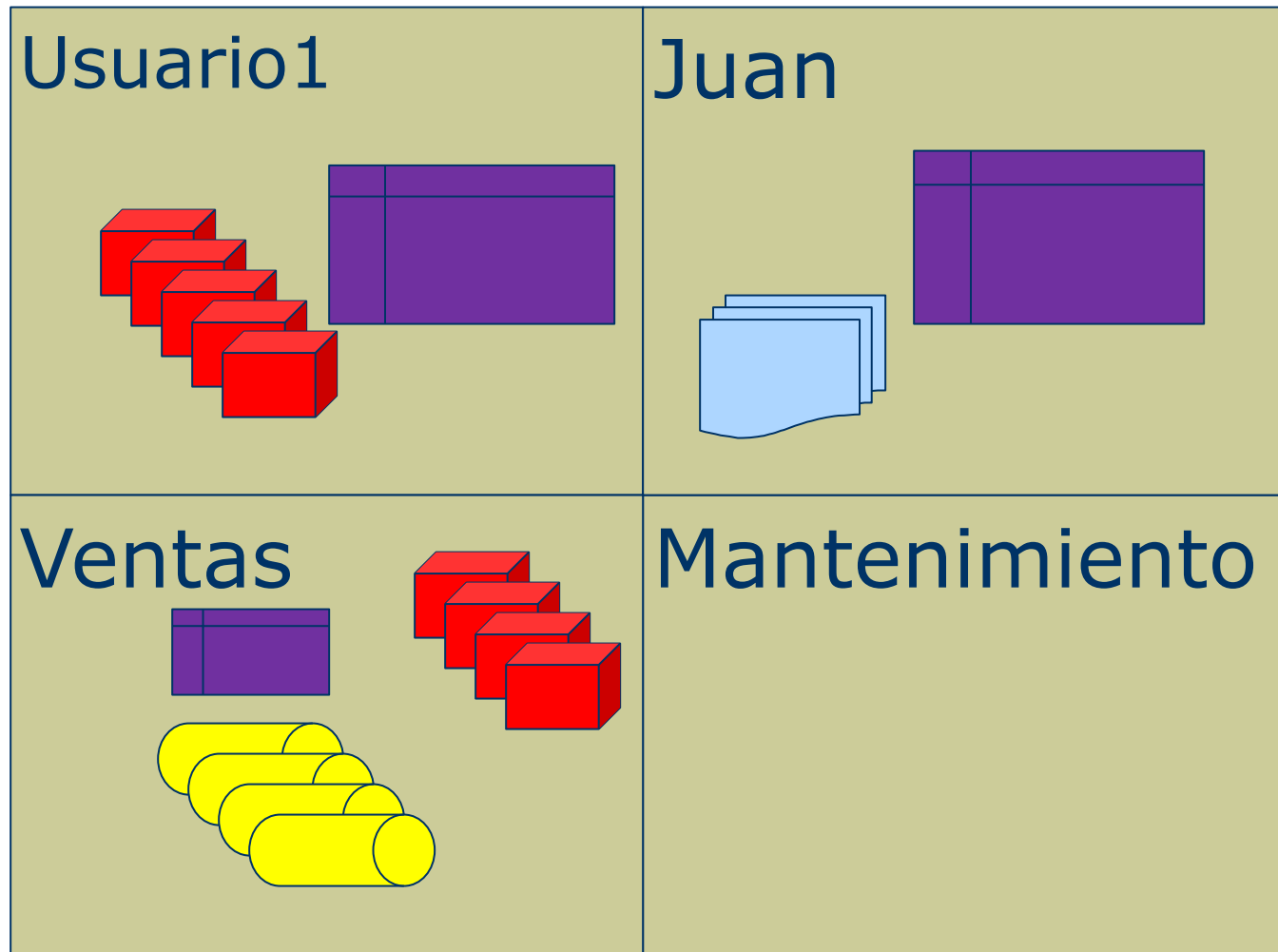


# Estructura

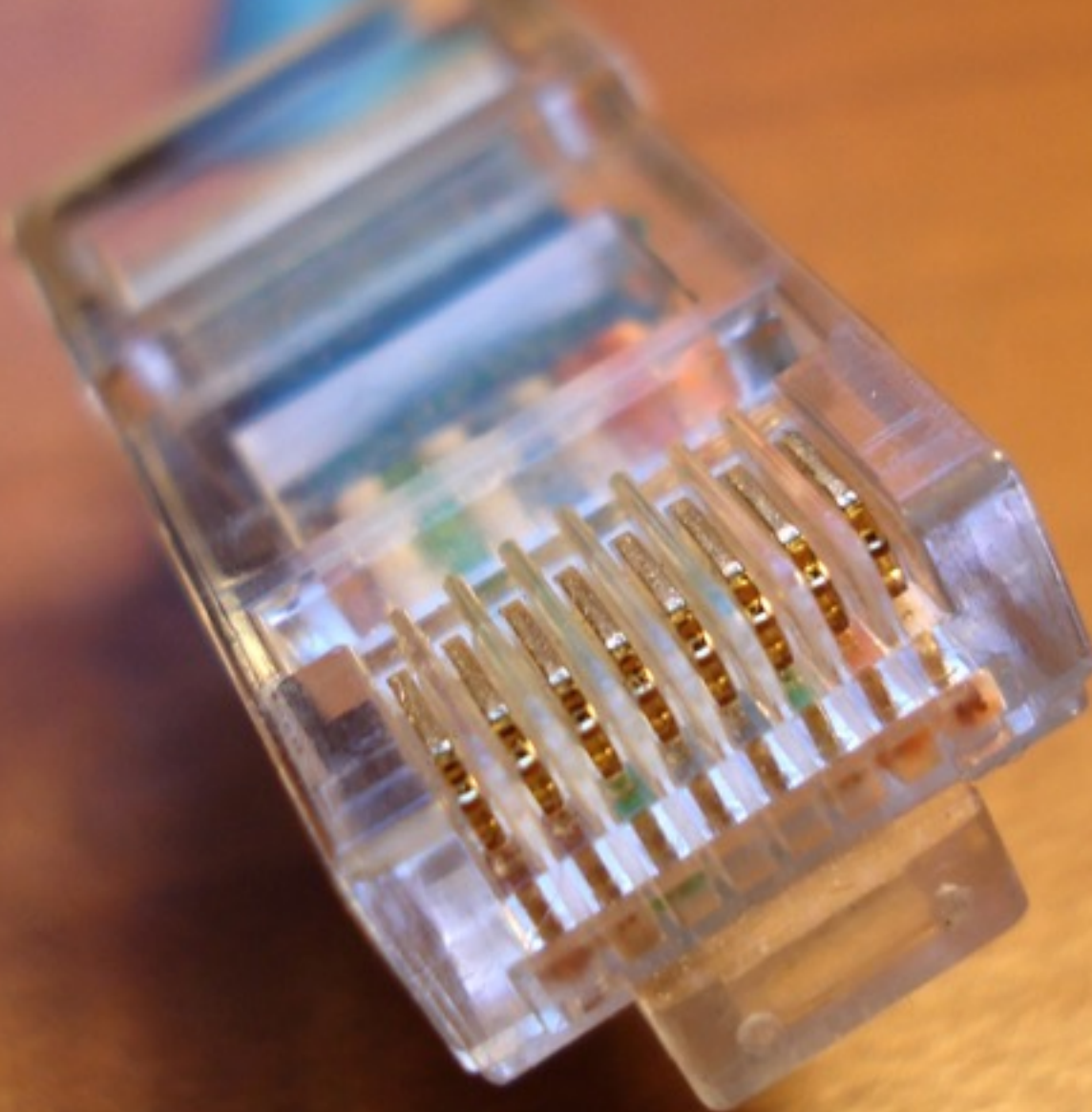
# Tablespaces



# Esquemas



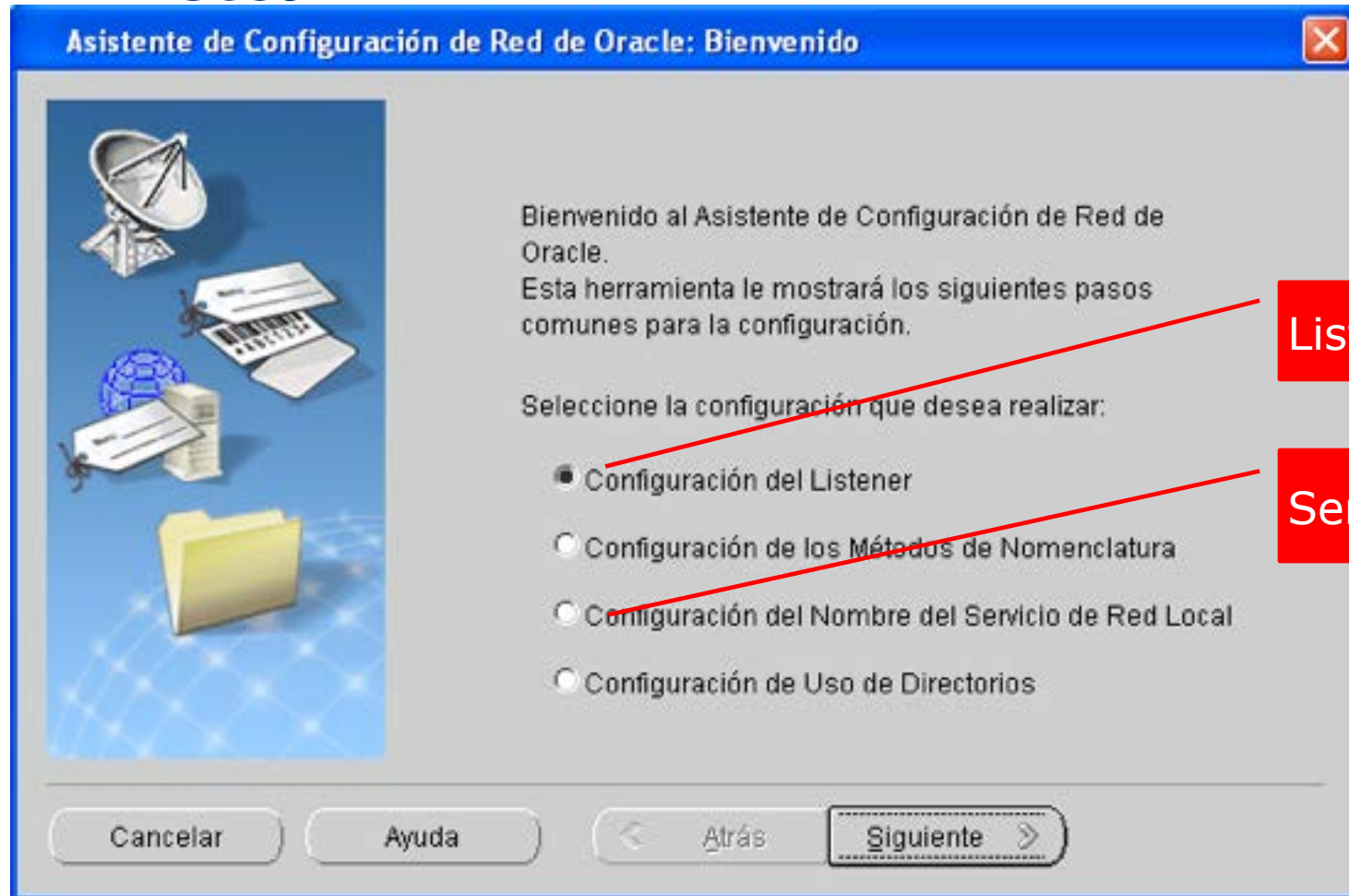
# Conexión















# Configuración de Red

netca



# Servicios y Archivos

## Insrctl

 OracleDBConsolejp	Manual
 OracleJobSchedulerJP	Deshabilitado
 OracleJobSchedulerORCL	Deshabilitado
 OracleMTSRecoveryService	Manual
 OracleOraDb10g_home1iSQL*Plus	iSQL*Plus ... Manual
 OracleOraDb10g_home1TNSListener	Manual
 OracleOraDb11g_home1ClrAgent	Manual
 OracleOraDb11g_home1TNSListener	Manual
 OracleServiceJP	Manual
 OracleServiceORCL	Manual

\$ORACLEHOME/NETWORK/ADMIN

LISTENER.ORA

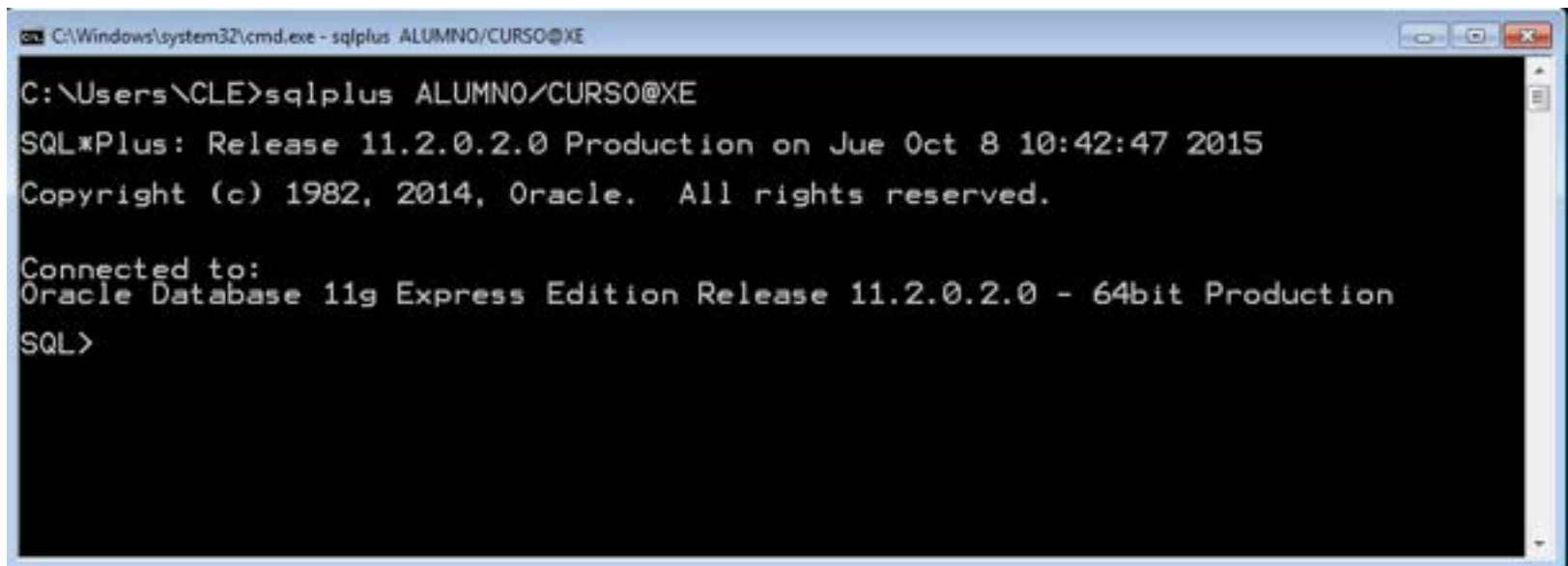
TNSNAMES.ORA

# Aplicaciones



# SQLPLUS

sqlplus usuario/contraseña@nombreServicioBD

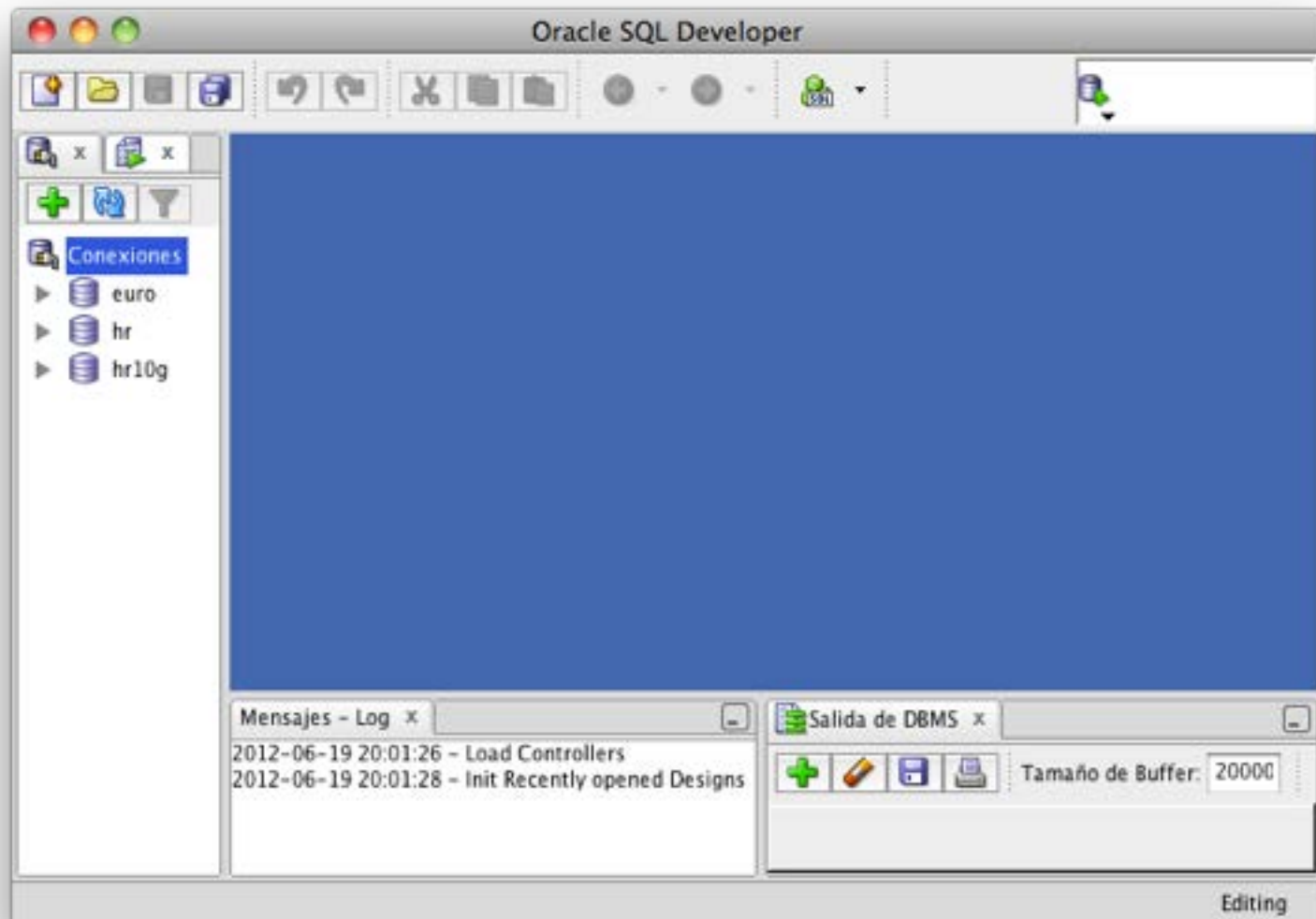


```
C:\Windows\system32\cmd.exe - sqlplus ALUMNO/CURSO@XE

C:\Users\CLE>sqlplus ALUMNO/CURSO@XE
SQL*Plus: Release 11.2.0.2.0 Production on Tue Oct 8 10:42:47 2015
Copyright (c) 1982, 2014, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production
SQL>
```

# SQL Developer





# SQL



# SQL

## Lenguaje Estándar de Consulta

SQL es un lenguaje de consulta, no de programación.



# Tipos de Datos

## NUMBER

NUMBER	cualquier número real
NUMBER (n)	n dígitos de un número entero
NUMBER (n,m)	n dígitos de las cuales m son decimales

## VARCHAR2

VARCHAR2 (n)	de longitud n
--------------	---------------

## DATE

# Estructura del lenguaje SQL



**SELECT**



**DML:** INSERT, UPDATE, DELETE, MERGE



**Instrucciones de transferencia:**  
**ROLLBACK, COMMIT**



**DDL:** CREATE, ALTER, DROP, RENAME , TRUNCATE.



**DCL:** GRANT, REVOKE.

**SELECT**



# SELECT

**SELECT** [**ALL**|DISTINCT]  
{ \* | {columna | expresión} [[AS] *alias*], ... }  
**FROM**  
    {[esquema.]{tabla|vista} | (subconsulta)}  
    [*alias*][, ...]  
[**WHERE**           lista\_de\_condiciones]  
[**GROUP BY**       lista\_de\_columnas]  
[**HAVING**          condiciones\_de\_grupo]  
[**ORDER BY**       columna [**ASC**|DESC]]  
;

# SELECT

```
SELECT *  
FROM emp;
```

# SELECT

```
SELECT ename, sal, sal+comm SalTotal  
FROM emp;
```

# SELECT

```
SELECT ename, sal, sal+comm SalTotal  
FROM emp  
WHERE deptno=20;
```

# SELECT

```
SELECT ename, sal, sal+comm SalTotal  
FROM emp  
WHERE deptno=20  
ORDER BY 1 ASC, 3 DESC;
```



# CASE

```
SELECT ename,  
        CASE deptno  
          WHEN 10 THEN  
            'Diez'  
          WHEN 20 THEN  
            'Veinte'  
          ELSE  
            'Otro'  
        END depart  
FROM    emp;
```

# CASE

```
SELECT ename,  
      CASE  
        WHEN deptno=10 THEN  
          'Diez'  
        WHEN deptno=20 THEN  
          'Veinte'  
        ELSE  
          'Otro'  
        END depart  
FROM    emp;
```

---

# Condiciones

**WHERE** cond1  
AND ((cond2 OR cond3)  
AND cond4)

=

>

<

>=

<=

<>

!=

~ =

IN

ALL

BETWEEN

LIKE

NOT

NULL

# Funciones Predefinidas

ROUND  
TRUNC  
NVL  
COUNT  
MAX  
MIN  
SUM  
AVG

LENGTH  
UPPER  
LOWER  
INITCAP  
RPAD  
LPAD  
SUBSTR

SYSDATE  
USER

TO\_NUMBER  
TO\_DATE  
TO\_CHAR

# GROUP BY

```
SELECT      SUM(sal), deptno  
FROM        emp  
GROUP BY    deptno;
```

SUM

MAX

MIN

COUNT

AVG

# HAVING

```
SELECT      SUM(sal), deptno  
FROM        emp  
GROUP BY    deptno  
HAVING      SUM(sal) > 2500;
```

SUM

MAX

MIN

COUNT

AVG

# JOINS

```
SELECT EMP.deptno, empno, dname  
FROM EMP, DEPT  
WHERE EMP.deptno=DEPT.deptno;
```

FK

Empno	Sal	...	Deptno
7878			10
7979			20
7676			10
7575			30
7474			

PK

Deptno	Dname	Loc
10	VENTAS	
20	ADMIN	
30	CONTAB	
40	DIR	



# JOINS

Empno	Sal	...	Deptno
7878			10
7979			20
7676			10
7575			30
7474			

Deptno	Dname	Loc
10	VENTAS	
20	ADMIN	
30	CONTAB	
40	DIR	

EMP.deptno	empno	dname
10	7878	VENTAS
20	7979	ADMIN
10	7676	VENTAS
30	7575	CONTAB



# JOINS

```
SELECT EMP.deptno,ename,dname  
FROM EMP, DEPT  
WHERE EMP.deptno=DEPT.deptno  
        AND sal>3000;
```

# JOINS

```
SELECT E.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno = D.deptno  
        AND sal > 3000;
```

# JOINS

```
SELECT E.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno = D.deptno(+);
```

Empno	Sal	...	Deptno
7878			10
7979			20
7676			10
7575			30
7474			

Deptno	Dname	Loc
10	VENTAS	
20	ADMIN	
30	CONTAB	
40	DIR	

# JOINS

Empno	Sal	...	Deptno
7878			10
7979			20
7676			10
7575			30
7474			

Deptno	Dname	Loc
10	VENTAS	
20	ADMIN	
30	CONTAB	
40	DIR	

E.deptno	empno	dname
10	7878	VENTAS
20	7979	ADMIN
10	7676	VENTAS
30	7575	CONTAB
	7474	

# JOINS

```
SELECT E.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno = D.deptno(+);
```

E.deptno	empno	dname
10	7878	VENTAS
20	7979	ADMIN
10	7676	VENTAS
30	7575	CONTAB
	7474	

# JOINS

```
SELECT E.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno(+) = D.deptno;
```

E.deptno	empno	dname
10	7878	VENTAS
20	7979	ADMIN
10	7676	VENTAS
30	7575	CONTAB
		DIR

# JOINS

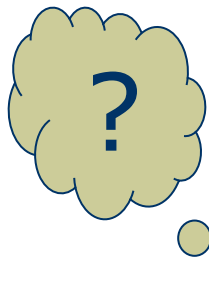
Empno	Sal	...	Deptno
7878			10
7979			20
7676			10
7575			30
7474			

Deptno	Dname	Loc
10	VENTAS	
20	ADMIN	
30	CONTAB	
40	DIR	

E.deptno	empno	dname
10	7878	VENTAS
20	7979	ADMIN
10	7676	VENTAS
30	7575	CONTAB
		DIR

# JOINS

```
SELECT E.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno(+) = D.deptno;
```



E.deptno	empno	dname
10	7878	VENTAS
20	7979	ADMIN
10	7676	VENTAS
30	7575	CONTAB
.		DIR



# JOINS

```
SELECT D.deptno, ename, dname  
FROM EMP E, DEPT D;
```

```
SELECT D.deptno, ename, dname  
FROM EMP CROSS JOIN DEPT D;
```

# JOINS

```
SELECT D.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno=D.deptno;
```

```
SELECT deptno, ename, dname  
FROM EMP JOIN DEPT  
      USING(deptno);
```

# JOINS

```
SELECT D.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno=D.deptno;
```

```
SELECT D.deptno, ename, dname  
FROM EMP E JOIN DEPT D  
ON(E.deptno=D.deptno);
```

# JOINS

```
SELECT D.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno(+) = D.deptno;
```

```
SELECT deptno, ename, dname  
FROM EMP RIGHT JOIN DEPT  
USING(deptno);
```

# JOINS

```
SELECT D.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno=D.deptno(+);
```

```
SELECT deptno, ename, dname  
FROM EMP LEFT JOIN DEPT  
USING(deptno);
```

# JOINS

```
SELECT D.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno(+) = D.deptno(+);
```

```
SELECT deptno, ename, dname  
FROM EMP FULL JOIN DEPT  
USING(deptno);
```

# JOINS



```
SELECT D.deptno, ename, dname  
FROM EMP E, DEPT D  
WHERE E.deptno(+) = D.deptno(+);
```

```
SELECT deptno, ename, dname  
FROM EMP FULL JOIN DEPT  
USING(deptno);
```

# Subconsultas

```
SELECT   ename, sal
FROM     emp
WHERE    deptno IN
           (SELECT deptno
            FROM      dept
            WHERE loc='DALLAS');
```



# Subconsultas

```
SELECT      deptno
FROM        emp
GROUP BY    empno
HAVING COUNT(empno) =
            (SELECT MIN(COUNT(empno))
             FROM      emp);
```

# Subconsultas

```
SELECT  ename,sal,loc,media
FROM
    (SELECT ename,sal,loc
     FROM emp
      JOIN dept USING (deptno)) t1
JOIN
    (SELECT AVG(sal) media,loc
     FROM emp
      JOIN dept USING (deptno)
     GROUP BY loc) t2
USING (loc);
```

# Subconsultas

```
SELECT deptno,  
      (SELECT SUM(sal)  
       FROM emp  
       WHERE deptno=D.deptno) Suma  
FROM dept D;
```



# DML

# INSERT

```
INSERT INTO dept  
VALUES(50,'TC','MADRID');
```

# INSERT

```
INSERT INTO dept(  
    deptno,  
    dname)  
VALUES(  
    60,  
    (SELECT department_name  
     FROM departments  
     WHERE department_id=260)  
    );
```

# INSERT

```
INSERT INTO dept
  SELECT      department_id,
                department_name,
                city
FROM departments
      JOIN locations USING (location_id)
WHERE department_id IN (80,90)
```

# DELETE

```
DELETE dept  
WHERE deptno=60;
```



# UPDATE

```
UPDATE emp  
SET      sal=3000  
WHERE deptno=20;
```

# UPDATE

```
UPDATE emp E
SET sal=(SELECT AVG(sal)
          FROM emp
          WHERE deptno=E.deptno);
```

# MERGE

```
MERGE INTO empleados e1
USING emp e2
ON(e1.empno=e2.empno)
WHEN MATCHED THEN
    UPDATE SET e1.sal=e2.sal, e1.job=e2.job
    WHERE e1.deptno IN (20,30)
    DELETE
    WHERE e1.empno=7521
WHEN NOT MATCHED THEN
    INSERT (e1.empno, e1.ename, e1.deptno)
    VALUES(e2.empno, e2.ename, e2.deptno)
    WHERE e2.hiredate IS NOT NULL;
```

# Instrucciones Transferencia



# **Instrucciones de transferencia**

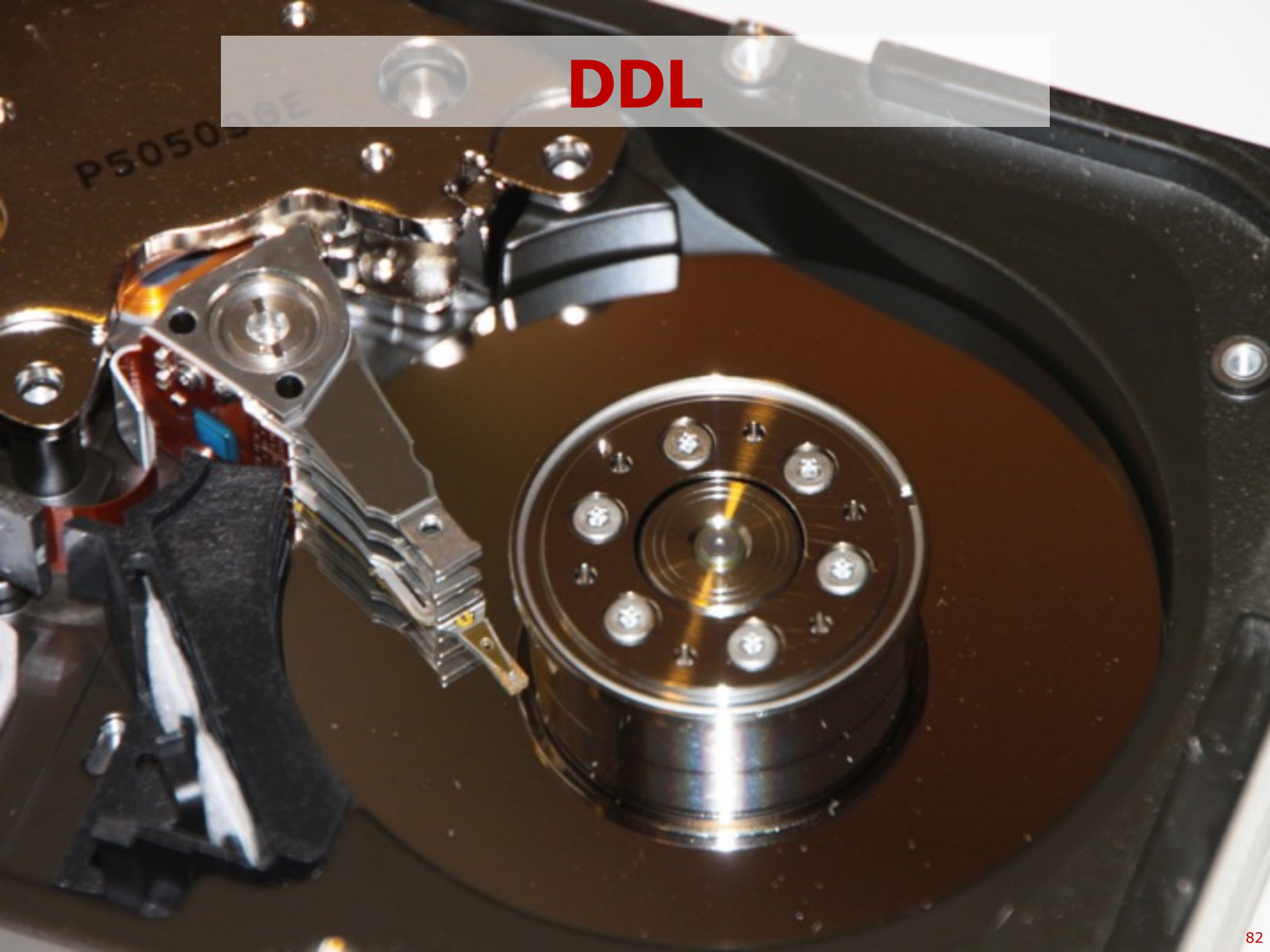
**COMMIT**

**ROLLBACK**

**SAVE POINT**



**DDL**



# Tipos de Constraints

**PRIMARY KEY**  
**FOREIGN KEY**  
**UNIQUE**  
**CHECK**

# PRIMARY KEY

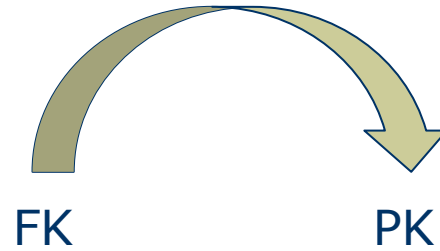
Nombre
Maria
MARIA
María
<del>María</del>



# PRIMARY KEY

Nombre	Apellido
Maria	Lopez
MARIA	López
María	Lopez
Mary	
María	López

# FOREIGN KEY



Empno	Sal	...	Deptno
7878			10
7879			20
7979			10
7576			30
7675			
<del>7777</del>			<del>77</del>

Deptno	Dname	Loc
10		
20		
30		
40		

# UNIQUE

Nombre
Maria
MARIA
María
<del>María</del>

# Constraints

- **Desactivada**

```
ALTER TABLE      nombre_tabla  
DISABLE CONSTRAINT nombre_constraint;
```

- **Obligatoria**

```
ALTER TABLE      nombre_tabla  
ENFORCE CONSTRAINT nombre_constraint;
```

- **Activada**

```
ALTER TABLE      nombre_tabla  
ENABLE VALIDATE CONSTRAINT nombre_constraint;
```

# DCL



# Constraints

**GRANT**

**REVOKE**

# Licencia

**Esta obra está sujeta a la licencia Reconocimiento-NoComercial 4.0 Internacional de Creative Commons. Para ver una copia de esta licencia, visite:**

**<http://creativecommons.org/licenses/by-nc/4.0/>.**

## **You are free to:**

**Share** — copy and redistribute the material in any medium or format

**Adapt** — remix, transform, and build upon the material

The licensor cannot revoke these freedoms as long as you follow the license terms.

## **Under the following terms:**

**Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

**NonCommercial** — You may not use the material for commercial purposes.