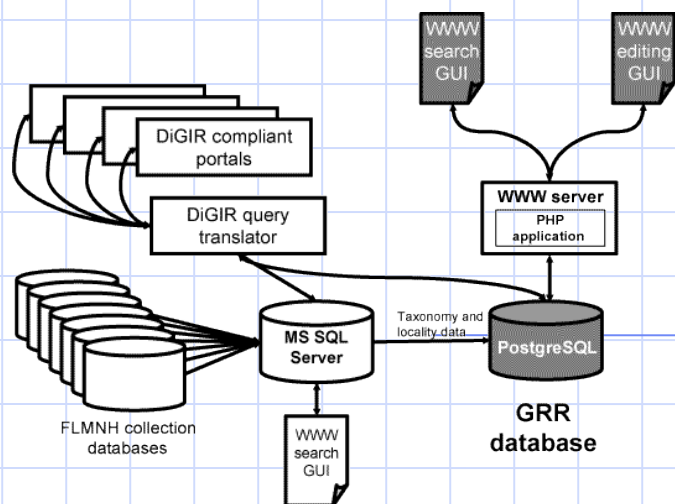




Universidad Tecnológica Nacional
Regional Paraná

Diseño y Adm. Base de Datos

Introducción a SQL (I) Consultas



Ing. Fernando Sato
fsatopna@gmail.com

Introducción a SQL

- **SQL (Structured Query Language).**
- **Es el Lenguaje mas importante en servidores relacionales.**
- **Es totalmente Declarativo.**
- **Standard de Mercado.**

Sub Lenguajes

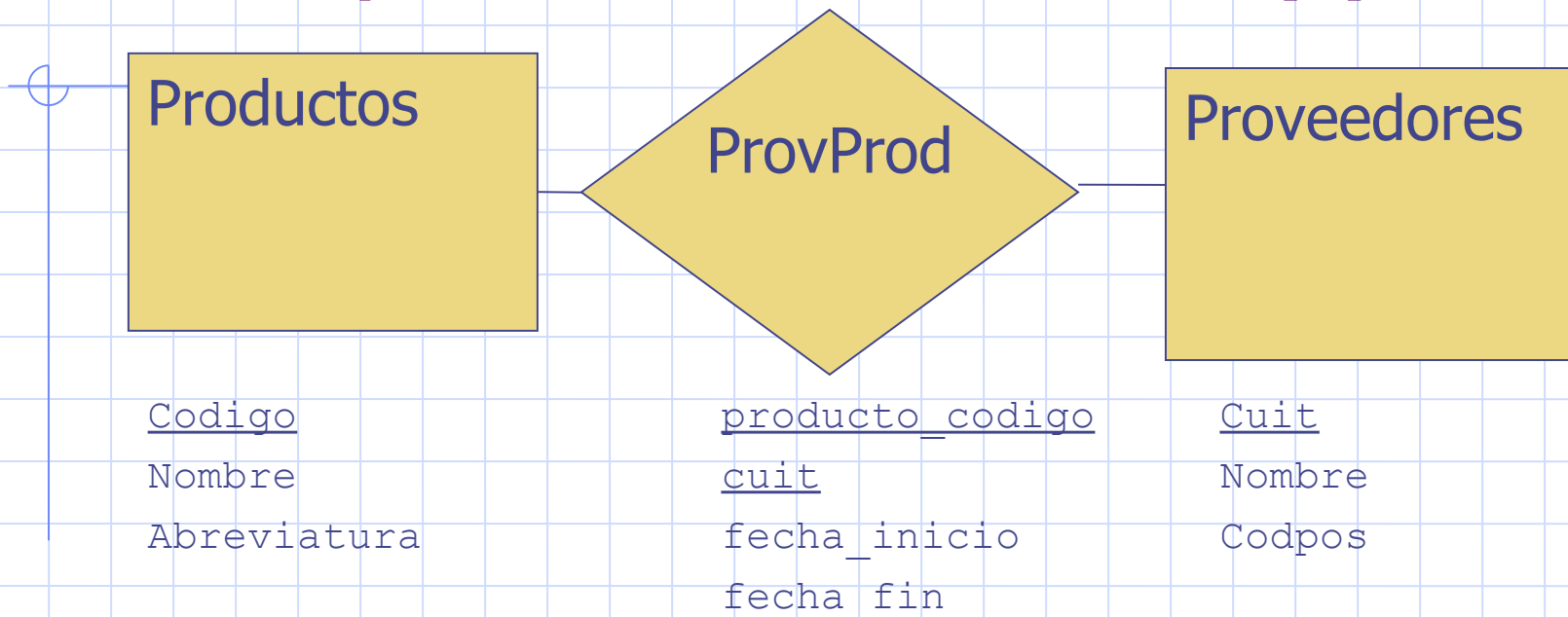
Se compone de 3 partes:

- DDL (Data Definition Language).
- DML (Data Manipulation Language).
- DCL (Data Control Language).

Historia SQL

- **1970 Edgard Codd** desarrolla el modelo Relacional.
- **1974-1979 IBM crea** System R - Sequel (precursor de SQL actual).
- **1986 ANSI** crea primer especificación (versión) de SQL.
- **1989, 1992, 1999, 2003 ANSI** actualizaciones de la especificación.
- **A la fecha** “casi” todos los vendedores de base de datos ofrecen productos que utilizan SQL.

Conceptos Básicos DDL (I)



<u>Productos</u>			<u>Proveedores</u>		
Codigo	Nombre	Abrev	Cuit	Nombre	Codpos
10	CDr Teltron	CDrT	20170446667	Garcia Pedro	3100
20	DVDr Tel	DVDrT	27170445134	Salas Marina	3000

<u>Proveedores_Productos</u>			
Producto_codigo	Cuit	Fecha_inici	Fecha_fin
10	20170446667	20/08/2009	
30	20170446667	10/07/2009	

Conceptos Básicos DDL(II)

Schema: Antes de crear tablas debo crear un schema.

```
CREATE DATABASE 'D:\cuentas\db\tsp.FDB' PAGE_SIZE 4096  
DEFAULT CHARACTER SET NONE;
```

En firebird un schema es equivalente a BD.

Existen Tipos de Datos Primitivos:

Caracteres: **char(*n*)**, **varchar(*n*)**

Enteros: **integer**, **smallint**

Reales: **numeric(*p*,*d*)**, **decimal(*p*,*d*)**

Coma flotante: **real**, **doble precision**, **float(*n*)**

Fecha: **date**, **time**, **timestamp**

Objetos grandes: **blob** y **clob** **sub_type(0 y 1)**

Conceptos Básicos DDL(III)

○ Creación de tablas : Se usa la DDL create table.

```
create table proveedores
( cuit char(11),      nombre varchar(50),
  codpos integer,    primary key (cuit) );
```

```
create table productos
( codigo integer primary key,
  nombre varchar(50),
  abreviatura char(6) );
```

```
create table proveedores_productos
( producto_codigo integer,      cuit char(11),
  fecha_inicio date,           fecha_fin date,
  primary key(producto_codigo, cuit));
```

DML – Select

Select: Sentencia mas importante de consulta de SQL.

Permite realizar prácticamente casi todas las operaciones del algebra relacional en forma declarativa.

Actualmente se utiliza además para monitorear información interna del servidor tal como, conexiones, memoria, almacenamiento físico y otras variables del servidor.

Select - Proyección

Proyección

Select *lista_proyección*
from *lista_relaciones*

Lista_relaciones

- Tablas (Relaciones Base)
- **Vistas (Relaciones Virtuales)**
- **Tablas Derivadas (Relaciones Resultado)**
- **Procedimientos**

Nota: Las listas se separan por coma.

Renombrado de atributos cuit as Cuit_cliente

Renombrado de Relaciones From proveedores as p

Select – Proyección (II)

Proyección

Select *lista_proyección*
from *lista_relaciones*

Lista_proyección

- Columnas (de algún miembro lista de relaciones)
- Constantes
- Operadores (+, - , *, /) (|| , SUBSTRING)
- Funciones Internas (Cast. Current_DATEs, CASE, etc)
- **UDF SUBSELECT de card 1 grado 1**
- **Funciones agregación**

Select – Proyección (III)

Proyección con filas únicas

Select [distinct] *lista_proyección*
from *lista_relaciones*

Lista_proyección

- Elimina filas repetidas.
- Debe estar solo una vez, después de la clausula select.

Select - Restricción

Restricción

Select *lista_proyección*
from *lista_relaciones*
[where *condiciones*]

Condición

- Operadores Lógicos: <, >, <=, >=, =, !=,
- Compuestas not, and, or between
- Operador especial like
- Se pueden emplear funciones casi los mismos elementos que se proyectan, con excepción de Agregación.

Nota: Las listas se separan por coma.

Select – Restricción II

Comparación de conjuntos

- EXISTS Al menos uno WHERE EXISTS (select
- IN comparación “=” de conjuntos (algun).
- *Problema con (c1, c2) IN (select c1, c2)*
- SOME para $>$, no se puede usar $>$ IN (conjunto)
 - Hay que usar “ $>$ SOME” (conjunto)
sql 99
- ALL Si queremos comprobar $>$ a todos los elementos del conjunto.
 - Hay que usar “ $>$ ALL” (conjunto)
sql 99

Select – Group By – Fxs agregate

Agrupamiento

```
Select lista_proyección  
from lista_relaciones  
[where condiciones]  
[group by lista_de_grupo]
```

Funciones de agregación:

- Count Sum
- Min max
- Avg

Select – Group By – Fxs aggregate

Agrupamiento

```
Select lista_proyección  
from lista_relaciones  
[where condiciones]  
[group by lista_de_grupo]  
[having condición_con_agregación]
```

Having:

- Restringe el conjunto resultado. Equivalente a where para select, pero solo se puede usar en group by.

Select – Order by

Orden

```
Select lista_proyección  
from lista_relaciones  
[where condiciones]  
[group by lista_de_grupo]  
[order by lista_de_ordenacion]
```

- Combinación con:
 - First
 - skip

Funciones de Fecha

Funciones que se pueden usar en proyección, restricción y group by

Funciones:

- Extract(year from campo_fecha) – Obtiene el año
- Extract(month from campo_fecha) – Obtiene el mes
- Extract(day from campo_fecha) – Obtiene el día

Select – valor NULL

Definición: Representa un valor desconocido, incierto.

Tratamiento en:

- Condicionales. IS
- Funciones de agregación.
- Función coalesce.

Select – Reuniones Internas externas



[Inner] Join

Left/Right [outer] Join

Full Join

Select – Sub select

En Proyección – Solo restringido a salidas de grado 1 y cardinalidad 1.

En Restricción

En lista de relaciones (Tablas Derivadas)

Select – UNION – INTERSECT – MINUS

Union: une 2 select de conjuntos homogéneos.

Intersect: intersecta 2 select de conjuntos homogéneos.
(no implementado en Firebird)

Minus: Diferencia de 2 select de conjuntos homogéneos.
(no implementado en Firebird)

Fuentes



Libro: Fundamentos de Bases de Datos

Autor: Silberschatz / Korth / Sudarshan

Editorial: Mc Graw Hill

Libro: Introducción a los SISTEMAS DE BASES DE DATOS

Autor: C.J. Date

Editorial: Addison Wesley

Fin



¿Preguntas?