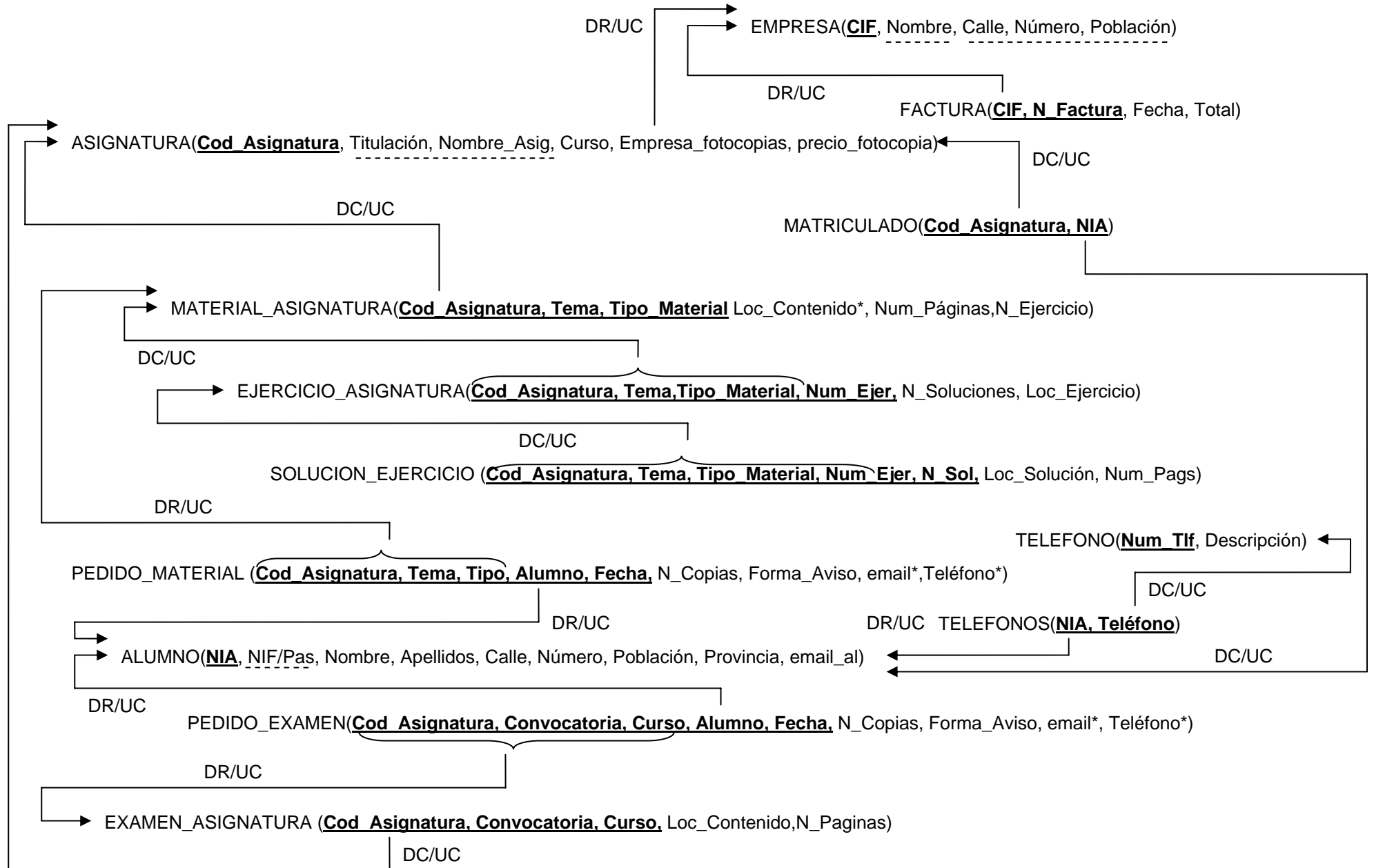


GRAFO RELACIONAL



SUPUESTOS SEMÁNTICOS

En la definición de las opciones de integridad referencial, se ha dado prioridad a la relación de pedidos, tanto de material como de exámenes. Por esto las relaciones PEDIDO_MATERIAL y PEDIDO_EXAMEN tienen restricciones de clave ajena con borrado restringido.

Varios alumnos podrán compartir un mismo teléfono.

Las interrelaciones que involucran a la relación EMPRESA han sido definidas con opción de borrado restringido para dar mayor relevancia a la información de las asignaturas y a la información de facturación de una empresa. No podrá eliminarse la información de una empresa mientras esté relacionada con alguna asignatura o con alguna factura.

DOMINIOS

- ✓ Titulación - Lista de todas las titulaciones ofertadas en la universidad.
- ✓ CIF - Concatenación de una letra y una secuencia de dígitos
- ✓ Tipo Material - Puede tomar los valores 'Apuntes', 'Ejercicios'
- ✓ Teléfono - Secuencia de 9 dígitos, 11 si es numeración internacional
- ✓ DNI/Pasaporte - Combinación de 9 caracteres alfanuméricos
- ✓ email_al - Cadena de caracteres terminada en "@estudia.es"
- ✓ Forma_Aviso - Toma valores en 'email', 'teléfono'

SEMÁNTICA NO RECOGIDA EN EL GRAFO RELACIONAL

CHECKS

Pedido-X.n_copias >= 1
Empresa.precio_fotocopia >= 0
Factura.total >= 0
Material_Asignatura.Num_Páginas > 0
Ejercicio_Asignatura.N_Soluciones >= 0
Material_Asignatura.N_Ejercicios >= 0

En Pedido_X, si Forma_Aviso = email entonces email <> null y telefono = null y viceversa

ASERCIONES

La fecha de la factura debe ser posterior a la de los pedidos que incluye

DISPARADORES

- ✓ Factura.Fecha -> Mensuales respecto a la fecha de factura anterior.
- ✓ Si material_asig.tipo = ejercicio al menos una tupla en ejercicio_asignatura
- ✓ Al menos un alumno por asignatura
- ✓ Al menos una asignatura por empresa
- ✓ Cálculo del número de soluciones para un ejercicio de una asignatura
- ✓ Cálculo del número de ejercicios de una asignatura
- ✓ Total de la factura obtenido como el 5% de lo que factura cada empresa (es decir, de la suma del producto del número de copias hechas por la empresa en el último mes por el precio por fotocopia).

3.- Implementación de disparadores

- a) Comprobar que un alumno no puede solicitar fotocopias de materiales de asignaturas en las que no está matriculado.

```
CREATE OR REPLACE TRIGGER esta_matriculado AFTER INSERT OR UPDATE OF  
Cod_Asignatura,NIA ON PEDIDO_MATERIAL  
FOR EACH ROW
```

```
DECLARE
```

```
/* Seccion de declaracion de variables, constante, cursores, etc. */
```

```
CURSOR cmatricula(inNIA VARCHAR2) IS SELECT Cod_Asignatura FROM MATRICULADO  
WHERE NIA = inNIA;
```

```
codasig MATRICULADO.Cod_Asignatura%TYPE;
```

```
aparece_matriculado BOOLEAN:=FALSE;
```

```
BEGIN
```

```
    OPEN cmatricula(:new.NIA);
```

```
    LOOP
```

```
        FETCH cmatricula INTO codasig;
```

```
        IF codasig = :new.Cod_Asignatura THEN
```

```
            aparece_matriculado:= TRUE;
```

```
        END IF;
```

```
        EXIT WHEN cmatricula%NOTFOUND;
```

```
    END LOOP;
```

```
    CLOSE cmatricula;
```

```
    IF aparece_matriculado = FALSE THEN
```

```
        RAISE_APPLICATION_ERROR(-20001,'El alumno no está matriculado en la asignatura  
para la que pide apuntes');
```

```
    END IF;
```

```
END esta_matriculado;
```

Para verificar si el disparador opera adecuadamente:

```
INSERT INTO
```

```
PEDIDO_MATERIAL(N_Copias,Forma_Aviso,email,Telefono,Tema,Tipo_Material,COD_ASIGNAT  
URA,NIA,Fecha)
```

```
VALUES(10,'email','mail@sitio.com','912323030','Tema1','apuntes','Asig1','100001','09/01/2008');
```

Esta operación debe realizarse correctamente

```
INSERT INTO
```

```
PEDIDO_MATERIAL(N_Copias,Forma_Aviso,email,Telefono,Tema,Tipo_Material,COD_ASIGNAT  
URA,NIA,Fecha)
```

```
VALUES(10,'email','mail@sitio.com','912323030','Tema1','apuntes','Asig1','100002','09/01/2008');
```

Esta operación debe dar error

4.- Consultas

a)

```
SELECT SUM(Total) As Facturacion, Nombre FROM (
  (SELECT
    SUM(Material_Asignatura.Num_Paginas*Pedido_Material.N_Copias*Asignatura.
    precio_fotocopia) As Total, Empresa.Nombre
  FROM Empresa, Asignatura, Material_Asignatura, Pedido_Material, Alumno
  WHERE Empresa.CIF = Asignatura.Empresa_fotocopias AND
  Asignatura.Cod_Asignatura = Material_Asignatura.Cod_Asignatura AND
  Material_Asignatura.Cod_Asignatura = Pedido_Material.Cod_Asignatura AND
  Pedido_Material.Fecha <= sysdate - 365
  Group BY Empresa.Nombre)
  UNION
  (SELECT
    SUM(Examen_Asignatura.Num_Paginas*Pedido_Material.N_Copias*Asignatura.
    precio_fotocopia) As Total, Empresa.Nombre
  FROM Empresa, Asignatura, Examen_Asignatura, Pedido_Examen, Alumno
  WHERE Empresa.CIF = Asignatura.Empresa_fotocopias AND
  Asignatura.Cod_Asignatura = Examen_Asignatura.Cod_Asignatura AND
  Examen_Asignatura.Cod_Asignatura = Pedido_Examen.Cod_Asignatura AND
  Pedido_Material.Fecha <= sysdate - 365
  Group BY Empresa.Nombre)
)
GROUP BY Nombre
ORDER BY Facturacion DESC;
```

b)

```
SELECT cp, N, COUNT(*) AS N_Alums FROM Asignatura, Matriculado, (SELECT AVG(Copias)
AS cp, Nombre_a As N FROM
((SELECT AVG(N_Copias) AS Copias,Asignatura.Nombre_asig as Nombre_a
FROM Matriculado, Asignatura, Pedido_Examen
WHERE Asignatura.Titulación = 'Ingeniería Informática' AND
Matriculado.Cod_Asignatura = Asignatura.Cod_Asignatura AND
Pedido_Examen.NIA = Matriculado.NIA
GROUP BY Asignatura.Nombre_asig
)
UNION
(SELECT AVG(N_Copias) AS Copias,Asignatura.Nombre_asig as Nombre_a
FROM Matriculado, Asignatura, Pedido_Material
WHERE Asignatura.Titulación = 'Ingeniería Informática' AND
Matriculado.Cod_Asignatura = Asignatura.Cod_Asignatura AND
Pedido_Material.NIA = Matriculado.NIA
GROUP BY Asignatura.Nombre_asig
))
GROUP BY Nombre_a) J
WHERE Asignatura.Nombre_asig = J.N AND
Matriculado.Cod_Asignatura = Asignatura.Cod_Asignatura
GROUP BY cp,N;
```

c)

```
SELECT Titulación, Tipo_Material, Ocurrencias
FROM (SELECT J.Titulación, J.Tipo_Material, SUM(frec) AS Ocurrencias
FROM ((SELECT COUNT(*) AS frec, Tipo_Material, Titulación
FROM Asignatura, Pedido_Material
WHERE Asignatura.Cod_Asignatura = Pedido_Material.Cod_Asignatura
```

```

GROUP BY Asignatura.Titulación, Tipo_Material
)
UNION
(SELECT COUNT(*) AS frec, 'Examen' As Tipo_Material, Titulación
FROM Asignatura, Pedido_Examen
WHERE Asignatura.Cod_Asignatura = Pedido_Examen.Cod_Asignatura
GROUP BY Asignatura.Titulación
)) J
GROUP BY J.Titulación, J.Tipo_Material)
WHERE Ocurrencias IN
(SELECT MAX(Ocurrencias) FROM
(SELECT J.Titulación, J.Tipo_Material, SUM(frec) AS Ocurrencias
FROM ((SELECT COUNT(*) AS frec, Tipo_Material, Titulación
FROM Asignatura, Pedido_Material
WHERE Asignatura.Cod_Asignatura = Pedido_Material.Cod_Asignatura
GROUP BY Asignatura.Titulación, Tipo_Material
)
)
UNION
(SELECT COUNT(*) AS frec, 'Examen' As Tipo_Material, Titulación
FROM Asignatura, Pedido_Examen
WHERE Asignatura.Cod_Asignatura = Pedido_Examen.Cod_Asignatura
GROUP BY Asignatura.Titulación
)) J
GROUP BY J.Titulación, J.Tipo_Material)
GROUP BY Titulación)

```

d)

```

SELECT DeExamen.Nombre, SUM(TotalMaterial), SUM(TotalExamen)
FROM (SELECT Empresa.Nombre,
SUM(Material_Asignatura.Num_Paginas*Pedido_Material.N_Copias*Asignatura.precio_fo
tocopia) As TotalMaterial
FROM Empresa,Pedido_Material,Material_Asignatura, Asignatura
WHERE Empresa.CIF = Asignatura.CIF AND
Asignatura.Cod_Asignatura = Pedido_Material.Cod_Asignatura
GROUP BY Empresa.Nombre) DeMaterial,
(SELECT Empresa.Nombre,
SUM(Examen_Asignatura.Num_Paginas*Pedido_Examen.N_Copias*Asignatura.precio_f
otocopia) As TotalExamen
FROM Empresa,Pedido_Examen,Examen_Asignatura, Asignatura
WHERE Empresa.CIF = Asignatura.CIF AND
Asignatura.Cod_Asignatura = Pedido_Examen.Cod_Asignatura
GROUP BY Empresa.Nombre) DeExamen
WHERE DeExamen.Nombre = DeMaterial.Nombre
GROUP BY DeExamen.Nombre;

```