

BD Activas: Motivación

- ⌘ Los SGBD convencionales son “pasivos”. Sólo ejecutan preguntas o transacciones realizadas por los usuarios o por los programas de aplicación.
- ⌘ Para representar la semántica del mundo real proporcionan:
 - ☑ MODELO DE DATOS
 - ☒ Estructuras de Datos
 - ☒ Operadores para trabajar con las estructuras
 - ☒ Reglas de integridad
 - ☑ MODELO DE TRANSACCIONES
 - ☒ Posibilidad de definir transacciones pero sólo si los usuarios o aplicaciones lo solicitan explícitamente ¹

BD Activas: Motivación (2)

- ⌘ Los SGBD “pasivos” NO SON BUENOS para modelar comportamiento dirigido por sucesos.
- ⌘ Ejemplo: si el stock de un producto baja de un cierto umbral entonces solicitar más. Para implementarlo:
 - ☑ 1) En toda aplicación que modifique el stock de algún producto hay que añadir código que compruebe si se baja del umbral para solicitar más.
 - ☒ La semántica está distribuida por las aplicaciones.
 - ☒ Posiblemente es una fuente de errores.
 - ☑ 2) Realizando un programa que periódicamente “sondee” todas las condiciones ($\exists \text{stock}(i) < \text{umbral}(i) ?$)
 - ☒ Frecuencia de sondeo alta --> INEFICIENCIA
 - ☒ Frecuencia de sondeo baja --> INCONSISTENCIAS

BD Activas: Definición y Modelo de Conocimiento

- ⌘ Un Sistema de Bases de Datos Activas es un sistema que monitoriza situaciones de interés y que, cuando ocurren, dispara o activa la ejecución de una serie de acciones.
- ⌘ El comportamiento deseado se expresa en forma de Reglas Evento-Condición-Acción (ECA)
 - ⊗ ON evento
 - ⊗ IF condición
 - ⊗ THEN acción
- ⌘ Nota: Las reglas ECA provienen del paradigma de las Reglas de Producción (IF condición THEN acción) tratado en Inteligencia Artificial (sobre todo en Sistemas Expertos)

Modelo de Conocimiento

⌘ **ON evento** IF condición THEN acción

☒ **evento** puede ser un suceso primitivo:

- ☒ ocurre una operación con la BD (insert, ...)
- ☒ comienza / termina una transacción (commit,..)
- ☒ suceso externo: bajada de tensión
- ☒ suceso temporal: es primer día de mes
- ☒ suceso abstracto: violada una regla de integridad

☒ o un suceso compuesto:

- ☒ S1 OR S2 (sucede el suceso S1 o el S2)
- ☒ S1 AND S2 (suceden ambos sucesos)
- ☒ S1 ; S2 (sucede S1 y después S2)

Modelo de Conocimiento (2)

- ON evento **IF condición** THEN acción
 - Se cumple una determinada condición en la BD
 - el valor de un atributo es uno determinado
 - el valor nuevo a insertar es menor que el viejo
 - etc.
- ON evento IF condición **THEN acción**
 - Se dice que se ejecute algo automáticamente
 - un abort o rollback
 - mandar un mensaje al usuario
 - introducir / modificar datos en la base de datos
 - etc.

Modelo de Ejecución

⌘ Es el comportamiento de las reglas en tiempo de ejecución. Se debe conocer:

☑ 1) Cuándo se evalúan los eventos (la frecuencia, si se evalúan dentro de transacciones, etc.)

☒ **Durante la ejecución de una transacción puede haber momentos en los que la BD está inconsistente**

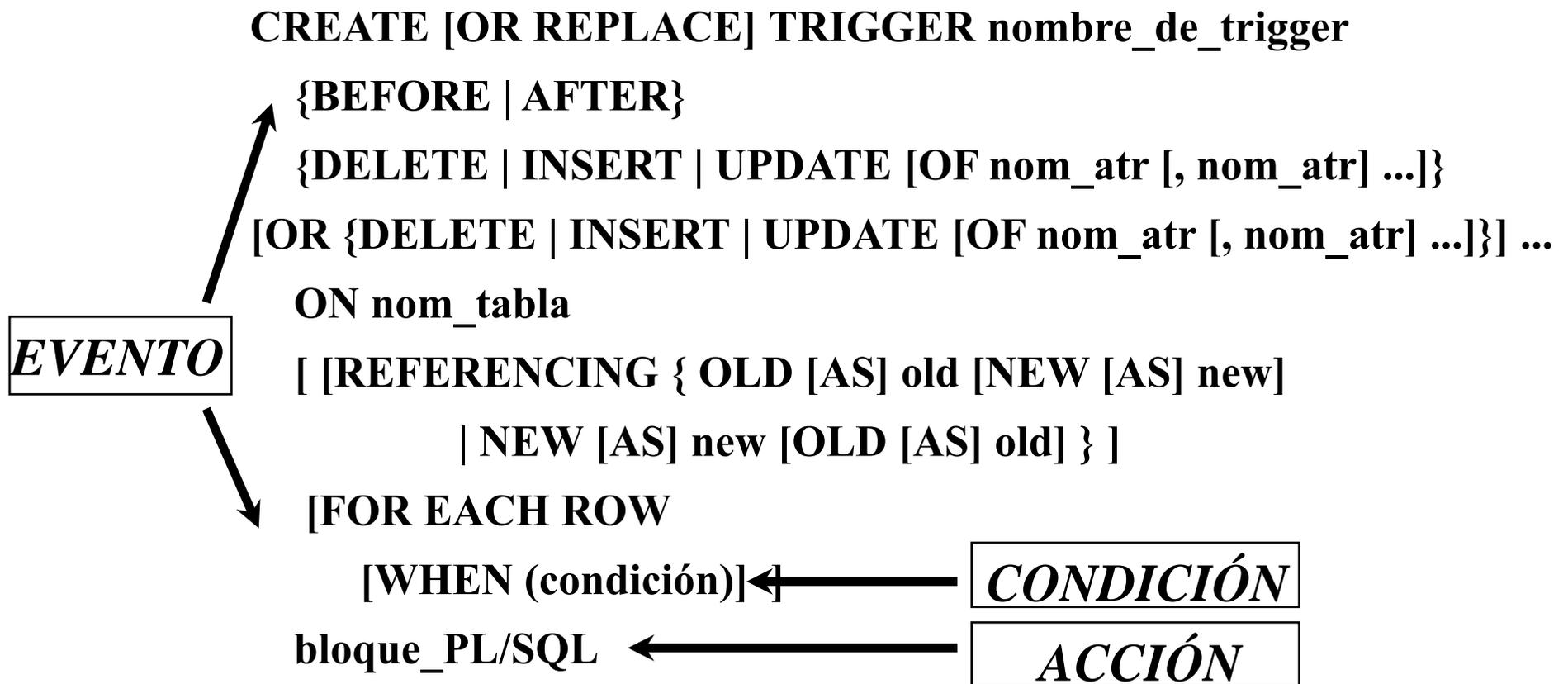
☑ 2) A qué reglas ECA se les evalúa antes la condición de entre las activadas por los eventos

☒ **¿Los eventos que ya han activado reglas pueden seguir activando otras? Ej: el evento "es el primer día del mes"**

☑ 3) Qué regla ECA se ejecutará la primera de entre las que cumplen la condición.

☒ **Relacionado con el problema del conjunto conflicto detectado por el motor de inferencia en S. Expertos**

Triggers en ORACLE 7



Triggers en ORACLE 7 (2)



- ⌘ OR REPLACE --> Reemplaza el trigger si ya existe
- ⌘ BEFORE/AFTER DELETE, INSERT, ... ON tabla
 - ☒ Indica si la acción (PL/SQL) se debe ejecutar antes o después de que se produzca el borrado, inserción o modificación de la tabla.
- ⌘ FOR EACH ROW indica que se ejecute la acción (si se cumple la condición) para cada tupla insertada, borrada,...
- ⌘ WHEN --> Es una condición SQL. No puede contener una pregunta SQL. Sólo SE PUEDE PONER la parte WHEN en triggers del tipo FOR EACH ROW
- ⌘ El bloque PL/SQL es la parte acción que ORACLE ejecuta cuando se produce el evento y se cumple la condición

Triggers en ORACLE 7 (3)

- Cuando los triggers son del tipo FOR EACH ROW, dentro del bloque PL/SQL se pueden utilizar las variables:
 - **:NEW** que contiene el **NUEVO** valor **INSERTADO** o **MODIFICADO**
 - El valor de **:NEW** se puede cambiar en triggers del tipo **BEFORE INSERT/UPDATE** pero no en triggers del tipo **AFTER**
 - así se podrá controlar el valor que se va a introducir.
 - **:OLD** que es el valor **BORRADO** o el valor viejo **MODIFICADO**.
 - Con **REFERENCING OLD AS mi_old ...** podemos renombrar **OLD** para que no haya problemas de nombres. Ej: una tabla se llama **OLD**
- Dentro del bloque PL/SQL se pueden realizar distintas acciones según se esté insertando, borrando o actualizando:
 - **IF INSERTING THEN ... END IF;**
 - **IF DELETING THEN ... END IF;**
 - **IF UPDATING ('nom_atr') THEN ... END IF;**

Restricciones en triggers

Oracle

- ⌘ El bloque PL/SQL que forma la parte acción no puede contener sentencias como COMMIT o ROLLBACK (ni CREATE..., ALTER...)
- ⌘ No se pueden crear triggers sobre tablas del sistema que forman el catálogo. Sería bueno para realizar acciones cada vez que se creara, borrara, etc. una tabla en la BD !!
- ⌘ Dentro de un trigger no se puede ni hacer SELECT de una **tabla mutante**, ni se puede cambiar la clave primaria, una clave ajena o claves únicas de una **tabla restringida**.
 - ☒ Una **tabla mutante** es aquella sobre la que se está haciendo un INSERT, un DELETE, un UPDATE o una tabla que puede ser afectada debido a una restricción DELETE CASCADE.
 - ☒ Una **tabla restringida** es aquella que es usada dentro de un trigger, por una sentencia SQL o para mantener una integridad referencial.
 - ☒ Una tabla no es considerada mutante ni restringida para triggers que NO SON del tipo FOR EACH ROW (excepto si el trigger se ha lanzado debido a una restricción DELETE CASCADE).

Consejos sobre Triggers Oracle



- ⌘ No definir triggers para definir restricciones de integridad que es posible definir de manera declarativa como REFERENCES, atributos NOT NULL, UNIQUE, etc.
 - ☑ Sí pueden servir para implementar los siguientes (no soportados todavía por Oracle)
 - ☒ ON DELETE / UPDATE SET NULL,
 - ☒ ON DELETE / UPDATE SET DEFAULT
- ⌘ No construir triggers recursivos.