

**Modelado de Datos en el Mundo Real:
Teoría vs. Práctica**

**Tema 5:
Reglas del Negocio**

Laura Rivero, Viviana Ferraggine
2010 Grupo Base de Datos y Procesamiento de Señales

Reglas del Negocio

¿Por qué varias definiciones? Porque hay varias **perspectivas** o puntos de vista

- De los **sistemas de información** → hechos del sistema que se registran como datos y restricciones sobre los cambios de valores de estos hechos.
- Del **negocio** → conducta de las personas en el sistema empresarial.
- De lo **comercial**.

... Una regla de negocio es la evidencia de que existe una obligación respecto a la conducta, acción, práctica o procedimiento dentro de una determinada actividad o ámbito.

Reglas del Negocio

- Características importantes de una regla de negocio:
 - Debería haber una motivación explícita para ello.
 - Debería haber un régimen de aplicación indicando cuáles son las consecuencias si la regla no se cumple.
- Existen diferentes clasificaciones de acuerdo al punto de vista de los autores.

Tipos de Reglas del Negocio

Clasificación de las reglas del negocio según Gottesdiener (1997)

Término, palabra, frase o conjunto de ellas que tiene un significado preciso en la organización. Pueden representar objetos o entidades o bien atributos o propiedades

Hechos que relacionan los términos, Aserciones que establecen un vínculo semántico entre dos o más términos

Restricciones, Son circunstancias bajo las cuales una transacción, decisión, acción u operación de negocio debería ser rechazada.

Habilitadores de acciones, Circunstancias que habilitan a la organización para ejecutar acciones previamente establecidas

Derivaciones, Permiten extraer nuevo conocimiento a partir de conocimiento ya existente a través de algoritmos explícitamente definidos

Inferencias, Indican la forma de convertir conocimiento o información en otra forma

Tipos de Reglas del Negocio

Clasificación de reglas según la metodología de Ross, 1997 : siete familias caracterizadas por el tipo de verificación que realizan, y una segunda clasificación específica para tipos de reglas derivativas

- Reglas verificadoras de instancias, de tipo y de posición:
- Reglas de verificadores funcionales:
- Reglas para evaluadores comparativos y matemáticos:
- Reglas para controladores de proyección o reparadoras:

Tipos de Reglas del Negocio

Clasificación de de reglas según ORM, Halpin, 2003 a 2005

- Unicidad simple en asociaciones binarias (N:1, 1:1)
- Restricciones mandatorias en asociaciones binarias((1,N) : (0,N))
- Restricciones de unicidad interna aplicables a asociaciones binarias y n-arias (N:1:1)
- Restricciones de unicidad externas básicas
- Restricciones de unicidad externa involucrando caminos de ensamble o andamios
- Restricciones mandatorias sobre roles en asociaciones n-arias
- Restricciones mandatorias disjuntivas (restricciones inclusive-or sobre conjuntos de roles)
- Restricciones de valores : (dominios)
- Restricciones de comparación de conjuntos (inclusiones, exclusiones, etc.)
- Restricciones de frecuencia interna en roles simples: (N, >=5)
- Restricciones de frecuencia interna aplicada a secuencias de roles múltiples
- Restricciones de frecuencia externa

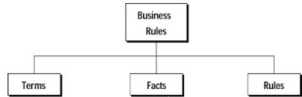
Reglas del Negocio: Enfoques (Barbara von Halle)

Se basa en 4 principios simples:

- (S) Separar. Separar las reglas de todos los otros aspectos de los requisitos.
- (T) (trace) seguimiento. Mantener la conexión de las reglas en dos direcciones. La primera desde el origen hacia la regla y la segunda desde la implementación de la regla hacia el origen.
- (E) Externalizar. Expresar la regla en un formato comprensible para un usuario no técnico, esto ayuda a que los líderes puedan verificarlas y medir su eficacia.
- (P) Posición. Significa tener en claro cual es la posición de la regla y poder construir el modelo de datos de una manera flexible de manera tal que los cambios de reglas no requieran cambios costosos en la base de datos.

Fuente: Barbara Von Halle - Business Rules Applied: Building Better Systems Using the Business Rules Approach

Reglas del Negocio: Enfoques (Barbara von Halle)



- Términos: es un sustantivo o una frase sustantiva con una definición determinada.
- Hechos: es una afirmación que conecta términos, a través de preposiciones y frases verbales, indicando observaciones relevantes para la organización.
- Reglas: afirmaciones declarativas

Terminos (von Halle)

- Un término puede definirse como:
 - Un concepto, por ejemplo cliente.
 - Una característica de un concepto, como por ejemplo Código de calificación crediticia del cliente.
 - Un valor, como por ejemplo femenino.
 - Un conjunto de valores, por ejemplo los días de trabajo (Lunes, Martes, Miércoles, Jueves, Viernes)
- Para un modelador de datos los términos probablemente serán entidades, atributos, dominios o constantes.

Hechos (von Halle)

- Ejemplos de hechos son los siguientes:
 - El cliente puede realizar un pedido
 - La orden tiene renglones
 - El cliente tiene un código de calificación crediticia
- Los términos y son los hechos detrás del semántica reglas.
- Se convertirá en la base de un modelo lógico de datos y base de datos física y tal vez un modelo de objetos de negocio.
- (Nota: Si usted es un modelador de datos, hechos que resultan ser relaciones entre las entidades o asociaciones de atributos a una entidad. Información que ayuda en el diseño del esqueleto de la BD.

Terminos + Hechos = (documentado) esquema-BD + Atributos

Reglas

- Es una afirmación declarativa donde se aplica la lógica o el cálculo de los valores.
- Una regla resulta de descubrir una nueva información o una decisión sobre una acción.
- Ejemplos de reglas son:
 - Un nuevo cliente no debe realizar un pedido cuya orden supere los \$ 1000.
 - El envío de un pedido de un cliente existente que no ha pagado su última factura, se retrasará hasta que se reciba el pago.
- Por lo tanto, una regla se convierte en una lógica ejecutable, que utiliza información de entrada y crea como salida información o acción

Reglas: Restricciones Mandatorias (von Halle)

- "Una sentencia completa que expresa una circunstancia incondicional que debe ser verdadera (o falsa) para un evento del negocio"
 - El cliente no puede tener mas de 10 ordenes abiertas
 - El total de ordenes no puede exceder el limite de credito
- Restricciones de tupla, tabla, base de datos
- Puede ser una restricción de transacción
 - Depende de la circunstancia incondicional

Reglas: Guías (von Halle)

- "Una sentencia que expresa una alerta acerca de una circunstancia que debe ser verdadera o falsa"
 - Un cliente no debe tener mas de 10 ordenes abiertas
- No restringe un estado de la BD
 - Algo que la interfaz del usuario debería o podría indicar
 - Entonces el usuario podría decidir continuar o no
 - Podría convertirse en una restricción de transacciones
 - Introduciendo una tabla de "guías violadas"

Reglas: Facilitadores de acciones (von Halle)

- "Una sentencia que testea las condiciones y si es verdadera, inicia un evento, un mensaje u otra actividad" (fuera de la db)
 - Si la orden es válida, iniciar el procesamiento de la orden
 - Si el cliente es de alto riesgo, notificar a los administradores
- No es completamente una restricción de estados de la BD: el autor prefiere que se realicen como restricciones de transacciones
- Mediante la introducción de una tabla de "acciones a realizar"
Construir un agente que monitoree la tabla y realice las acciones

Reglas: Calculos (von Halle)

- "Una sentencia que provee un algoritmo para arribar a un valor de un término"
 - Monto total de la orden = suma de los montos de los items
- Si el término está almacenado (es decir es parte del esquema de la BD)
 - El término es *redundante* y el cálculo es una restricción estática (tupla, tabla o BD)
- Sino,
 - Probablemente sea un código centralizado que sea utilizado por otras aplicaciones. (por ej. Función que calcula la suma de los montos)

Reglas: Inferencias (von Halle)

- "Una sentencia que testea una condición si es verdadera, establece la verdad de un nuevo hecho"
 - Si el cliente no tiene facturas pendientes, entonces tiene un estado de cliente preferencial.
 - Si el cliente es preferencial, entonces sus nuevas ordenes tendrán un 20 por ciento de descuento
- El resultado de ejecutar una regla de inferencia es una nueva información → puede ser una nueva instancia de una entidad o un nuevo valor de un atributo

Resumiendo (von Hallen)

- Los términos definen conceptos de datos y detalles
- Los hechos definen las asociaciones entre los datos
- Las restricciones y las guías verifican los datos
- Los cálculos dan como resultados nuevos valores de datos
- Las inferencias dan como resultados nuevas conclusiones acerca de datos
- Los facilitadores de acción evalúan valores de datos antes de iniciar la acción

Reglas del Negocio: Enfoque Date

- **Chris Date. Twelve Rules for Business Rules. Chris J. Date . 2000.**
- **Definiciones:**
 - **Aplicación de software:** su propósito es implementar una porción del **trabajo de la organización**
 - **Trabajo de la organización:** está **especificado** como un conjunto de definiciones (definiciones de datos, de accesos, de formularios, etc.)
 - **Especificaciones:** son por naturaleza **declarativas**, también las hay **formales** o bien llamadas **reglas del negocio**
 - **Reglas del negocio:** son compilables.

El conjunto de reglas que constituyen la especificación declarativa para una aplicación dada ES el código fuente para dicha aplicación.

Reglas del Negocio: Enfoque Date

Las 12 meta-reglas

1. Las BR deben ser ejecutables y compilables.
2. Las BR deben ser declarativas y expresivamente completas
3. Las BR pueden ser de los siguientes tipos:
 - Reglas de presentación
 - Reglas de aplicación
 - Reglas de base de datos
 - Las 2 últimas pueden ser de cálculo, de restricción o de inferencia.
4. Secuencia de declaración vs. Secuencia de ejecución
5. El motor de reglas se puede asimilar a un DBMS: posee las estructuras conceptuales Externa-Conceptual-Interna

Reglas del Negocio: Enfoque Date

ContinuaciónLas 12 meta-reglas

6. El motor de reglas es un DBMS relacional, asumiendo que los dominios y las relaciones son necesarias y suficientes.
7. Administrador de reglas.
8. Tipos de restricciones:
 - Restricciones de dominio (o tipo)
 - Restricciones de columna (o atributo)
 - Restricciones de tabla (o relación)
 - Restricciones de base de datos
9. Extensibilidad
10. Independencia de la plataforma
11. No subvertir: significa que no se deben violar los principios del DBMS
12. Automatización completa

Reglas de Presentación (Date)

- "Reglas que se aplican a aspectos de presentación de una aplicación"
 - A realizar en la interface del usuario: los formularios, las impresiones, los mensajes de error, etc.
 - Para facilitar la uniformidad y consistencia de las aplicaciones
 - "Muchas personas tienden a no incluir estas cosas bajo el título general de Regla del Negocio"
- Entonces se debe buscar que:
- El IDE facilite la "uniformidad y consistencia".

Reglas de Aplicacion (Date)

- "Reglas que aplican a los aspectos específicos de la función comercial per se"
 - "Otros autores plantean tratarlas junto a las de base de datos ya que es difícil trazar una línea divisoria entre ambas."
- Adicionalmente otros opinan que se debe extender el esquema de la base de datos

Restricciones de dominio (Date)

- "Especifica valores legales para un cierto dominio (tipo)"
 - Pueden ser tanto tipos definidos por el sistema o por el usuario (de complejidad arbitraria)
 - Tipo cantidad: los valores lícitos deben ser tales que puedan representarse por, precisamente, los enteros 1, 2, ..., 5000
- Las restricciones de dominio, definiendo un atributo establecen el conjunto de valores posibles

Restricciones de columna (Date)

- "Especifica los valores legales para una columna (atributo)"
 - Una declaración al efecto que los valores de la columna son tomados de algún dominio dado.
 - ...en su libro no esta clara la diferencia entre restricción de columna y de tabla!!

Restricciones de Tabla (Date)

- Cualquiera de las siguientes puede ser una restricción de tabla:
 - Una definición de clave candidata
 - Un chequeo que involucre toda la tabla (en su ejemplo del libro esto no sucede!!! Se muestra un chequeo de columna)
 - También estaba en su libro "An introduction of Database Systems" la definición de foreign key, que luego cambió su clasificación a aserción (por el hecho de involucrara 2 relaciones)

Restricciones de base de datos (Date)

- "Son restricciones que se refieren al menos a dos tablas"
 - Foreign key
 - Los proveedores que están con estado X no deben enviar mas de Y cantidades de partes

Enfoques de Reglas del Negocio

Origen	Taxonomia
Business Rules Project	Structural assertion (terms, facts) Action assertions (integrity constraints, conditions, authorizations) Derivations (calculations, inferences)
Ron Ross, Database Research Group	terms facts (relationship types, subtyping, attribute) rules (computations, rejecters, projectors)
Barbara von Halle, Knowledge Partners	definitions facts constraints derivations Inferences
.....

Tipos vs Clasificación de Reglas del Negocio

<p>Tipos generales de reglas vs</p> <ul style="list-style-type: none"> ▪ Restricción – debe o no debe pasar ▪ Heurísticas – guías, pero no un debe ▪ Inferencia – dada una condición, asumir otra condición (cambio de estado) ▪ Tiempo – actividad basada en lapsos de tiempo (polling) ▪ Triggers – relaciones de causa - efecto (evento) 	<p>Clasificación de reglas</p> <ul style="list-style-type: none"> ▪ Restricción: Forzar o rechazar una condición. Probablemente las más comunes. ▪ Inferencia: Siempre se expresa como que algo positivo va a suceder (respuesta) frente a un evento específico que ya ha sucedido. (Si / Entonces) ▪ Controlador: Control de procesos y de datos. ▪ Trigger: Si se cumple la condición entonces se dispara la acción
--	--

Nota: La clasificación de reglas no tiene heurísticas porque no es posible implementarlas

Tipos de expresiones de reglas

Business Conversation Piece	→	Natural Language Version	→	Rule Specification Language Version	→	Rule Implementation Language Version
-----------------------------	---	--------------------------	---	-------------------------------------	---	--------------------------------------

<ul style="list-style-type: none"> • May not be relevant • May not be atomic • May not be declarative • May not be precise • May be incomplete • May not be reliable • May not be authentic • May be redundant • May be inconsistent 	<ul style="list-style-type: none"> • Relevant • Atomic • Declarative • Not totally Precise • May be incomplete • Reliable • May not be authentic • May be redundant • May be inconsistent 	<ul style="list-style-type: none"> • Relevant • Atomic • Declarative • Precise • Complete • Reliable • Authentic • Unique • Consistent 	<ul style="list-style-type: none"> • Executable • May be procedural
---	--	---	---

Fuente: Business Rules Applied. Barbara von Hallen

Regla del Negocio

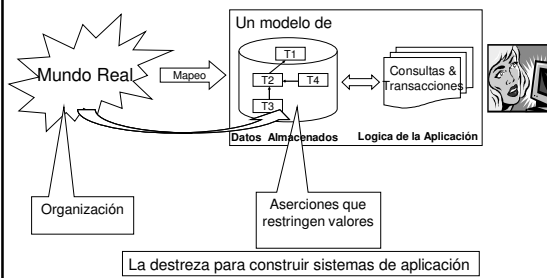
Desde el punto de vista del Modelo de Datos

The diagram illustrates the process of business rule implementation from a data model perspective. It starts with the 'Universo del Discurso' (Discourse Universe) leading to 'Análisis de Requisitos' (Requirements Analysis), then 'Diseño Conceptual' (Conceptual Design), 'Diseño Lógico' (Logical Design), and finally 'Diseño Física' (Physical Design) leading to 'Implementación BD' (DB Implementation). A small table is shown with columns: ID, Nombre, Fecha, Código, Estado, Fecha Alta, Fecha Baja. Rows include: ALUMNO (UNI, Nombre, Dirección,...), REALIZAR (UNI, Nombre), and CURSO (Nombre, Libro, WWW,...).

Reglas del Negocio: Enfoque Koppelaars

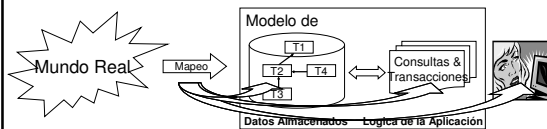
- Regla del Negocio = "toda **aserción** que restringe los posibles valores en la base de datos".
 - **Aserción:**
 - > 1.1. Acción y efecto de afirmar o dar por cierto algo.
 - > 2.1. Proposición en que se afirma o da por cierto algo.
 - **Aserción** → establece **QUE**, no **COMO**
 - Estado de la base de datos → Conjunto de valores actuales de nuestras tablas
 - Completamente expresable en términos del **esquema de la BD**
 - Esquema de la BD → estructura de tabla/columna
- Esta definición implica que si una regla no puede ser parafraseada (transformada, traducida) en términos de los valores de las tablas (los valores en un estado de la base de datos) que implementan el modelo de datos, entonces no puede considerarse una regla del negocio para este enfoque.
- Si la regla debe considerarse → el modelo de datos debe extenderse.

Reglas del Negocio: Enfoque Koppelaars



Reglas del Negocio: Enfoque Koppelaars

- Este enfoque no sugiere que otras 'divagaciones' no sean relevantes...
- Implica que → las BR deben estar claramente separadas de otras reglas que limitan los valores en un estado de la base de datos.



COMO versus QUE

- Si el estado de un proveedor (P) es un valor < 20 entonces la cantidad de todo envío (E) debe ser < 500 .
- Si la cantidad del envío es un valor mayor que 499 entonces el estado del proveedor debe ser mayor o igual a 20.

Que

Cómo

Para todo r en $Join(P,E)$: $estado < 20 \Rightarrow cantidad < 500$

Definición formal de una base de datos

- Un estado de una base de datos en un momento determinado es el conjunto de tablas \rightarrow cada tabla tiene su estado válido (que se ajusta a las reglas del negocio)
- En el transcurso del tiempo ocurren transacciones que cambian el estado de la base de datos en nuevos estados de la base.
- Se quiere modelar el conjunto de todos los estados válidos
 - Un estado de la BD es un conjunto admisible de tablas
 - Las tablas están compuestas por un conjunto admisible de tuplas
 - Las tuplas están compuestas por un conjunto admisible de atributos
 - Un atributo es un valor admisible tomado de un conjunto de valores establecidos
- El conjunto de estado permitidos de la BD se denomina **Universo**
 - Database = variable bastante compleja
 - Universo de la BD = sus tipos de datos...
- Limitar los posibles valores en un estado de la base de datos

Definición formal de una base de datos

Universo de transacciones

TX

Universo de la Base de Datos

DB

Universo de Tablas

T1

T2

...

Tn

Universo de Tuplas

t1

t2

...

tn

Conjunto de valores de atributos

A1

A2

...

An

Esquema de la Base de Datos

S

Restricción de Atributo

- Construcción bottom-up de un Universo de BD
- Esquema de la Base de Datos
 - Por tabla, ¿cuales son los atributos?
 - Por tabla, por atributo, ¿cual es su conjunto de valores?

Conjunto de valores de Atributo

Esquema de la Base de Datos

Restricción de Atributo

- Definición de conjunto de valores de los atributos
 - Aserciones respecto de que valores de atributos deben garantizarse
 - Pueden ser especificadas y chequeadas sin tener que referirse a otros valores de atributos dentro de una tupla.
 - Ámbito de la aserción = **el atributo**

Los valores permitidos para tarea son 'GERENTE', 'SECRETARIO' y 'VENDEDOR'

$t(\text{TAREA}) \in \{\text{'GERENTE'}, \text{'SECRETARIO'}, \text{'VENDEDOR'}\}$

Restricción de Tupla

- Tuplas posibles ("Producto cartesiano del conjunto de valores")
 - + **restricciones de tupla** → Universo de Tuplas

Universo de Tuplas

Conjunto de valores de atributos

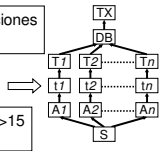
Esquema de la Base de Datos

Restricción de Tupla

- Aserciones entre diferentes atributos de una misma tupla
 - Pueden ser especificadas y chequeadas sin tener que referirse a otras tuplas.
 - Ámbito de la aserción = **la tupla**

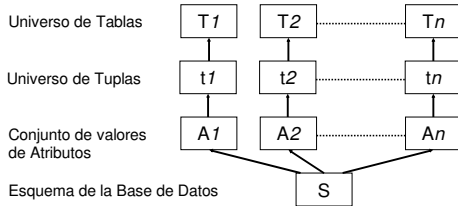
Los gerentes deben tener mas de 15 días de vacaciones
Las vacaciones no pueden exceder las 3 semanas

$t(\text{TAREA}) = \text{'GERENTE'} \Rightarrow t(\text{DIAS_VACACIONES}) > 15$
 $(t(\text{ULTIMO_DIA}) - t(\text{PRIMER_DIA}) + 1) \leq 21$



Restricción de Tabla

- Conjunto de tablas posible
 - + restricciones de tabla → Universo de Tablas

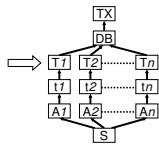


Restricción de Tabla

- Aserción entre diferentes tuplas de una misma tabla o grupo de tuplas de una misma tabla.
 - Pueden ser especificadas y chequeadas sin tener que referirse a otras tablas.
 - Ámbito de la aserción = **la tabla**

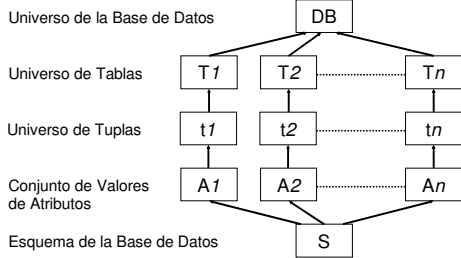
Todas las SECRETARIAS deben tener menos días de vacaciones que los VENDEDORES

$(\forall c \in T: c(\text{TAREA}) = \text{'SECRETARIA'} \Rightarrow$
 $(\forall s \in T: s(\text{TAREA}) = \text{'VENDEDOR'} \Rightarrow$
 $c(\text{DIAS_VACACIONES}) <$
 $s(\text{DIAS_VACACIONES})$
 $)$



Restricción de Base de Datos

Posibles estados de la Base de Datos (Producto Cartesiano del universo de tablas) + Restricción de base de datos → **Universo de Base de Datos**

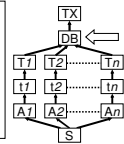


Restricción de Base de Datos

- Aserciones entre diferentes tablas de la misma base de datos en un mismo estado de la base de datos
 - Ámbito de la aserción = **estado de la base de datos**

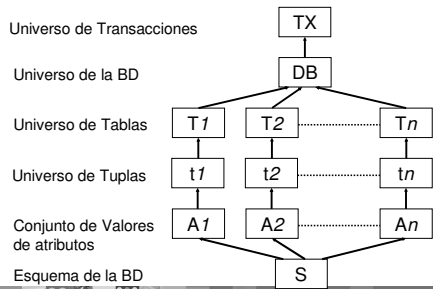
Las vacaciones no podran exceder el máximo permitido

```
(∀ev(EMP):
  (∑s: sev(VAC) ∧ s(EMPNO)=t(EMPNO):
    { deWORKDATE
      s(PRIMER_DIA) ≤ s(ULTIMO_DIA) }
  )
  ≤ t(DIAS_VACACIONES)
)
```



Restricciones de transición

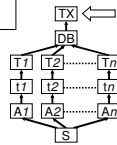
Referirse a más de un estado de la base de datos
+ restricciones de transición → **Universo de Transacciones**



Restricciones de transición

- Aserciones que contienen estados de transición
 - Ámbito de la aserción = **estados Inicio + Resultado BD**

- Los días de vacaciones solo pueden incrementarse
- No se permiten actualizaciones de identificadores únicos
- Los registros insertados de períodos de vacaciones deben ser "los más recientes".

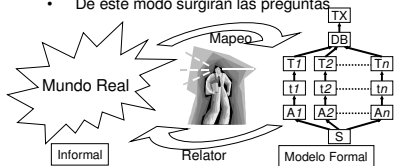


Nota Importante...

- Una regla del negocio no puede ser clasificada sin el contexto del esquema de la base de datos
- Pero a veces se pueden encontrar diferentes formas de representación, por ejemplo una jerarquía disjunta parcial:
 - Normalmente es una restricción que involucra varias tablas (de base de datos), pero:
 - Se puede agregar un atributo a la tabla padre y esto se vuelve una restricción de atributo.

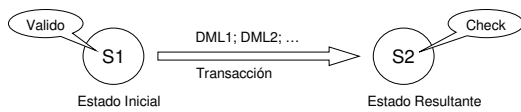
Beneficios de este enfoque

1. Clasificación dirigida al ámbito de los datos → fuertemente relacionada con las características de implementación
2. Especificación formal → nunca causa ambigüedad
3. Pero, no se debe confrontar a los usuarios con este formalismo
 - Los profesionales de IT trabajan con los usuarios de un modo informal
 - Continúan tratando de mapear al modelo
 - De este modo surgiran las preguntas

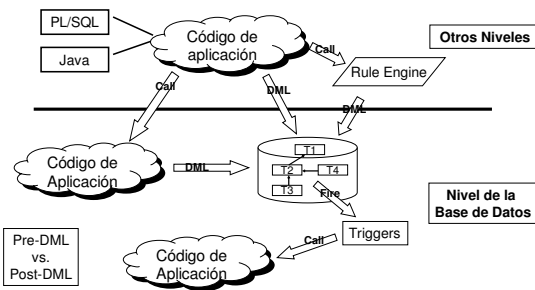


Implementación de Reglas del Negocio

- Diferentes estrategias de implementación



Estrategias de Implementación



Implementación Post-DML

- Declarativo
 - Sintaxis de creación de tablas
 - Implementación fácil, eficiente, mantenible
 - **Que = Como!**
- Procedural
 - Triggers asociados con las tablas
 - Insert, Update, Delete
 - Before DML-statement, after statement
 - Before each row, after each row
 - **Que = Como!** → potencialmente complejo...

/

Restricciones de Atributos

- Uso declarativo en la clausula check

```
Create Table Emp
(empno      number(4,0)      not null
             check (empno>0),
enombre     varchar2(20)    not null,
tarea       varchar2(10)    not null
             check (tarea in
('GERENTE','SECRETARIO','VENDEDOR')),
dias_vacaciones number(3,0) not null
             check (dias_vacaciones>10)
);
```

- Es posible darle un nombre al constraint
- Desconocido por NULL evalúa a TRUE

Restricciones de Tuplas

- Uso declarativo en la clausula check (out-of-line) check

```
-Create Table Emp
(empno      number(4,0)      not null
enombre     varchar2(20)    not null,
tarea       varchar2(10)    not null,
dias_vacaciones number(3,0) not null,
             check (tarea<=>'GERENTE' or dias_vacaciones >15)
);
```

TAREA='GERENTE' ⇒ DIAS_VACACIONES>15

A ⇒ B, es lógicamente equivalente a: ¬A ∨ B

Restricciones de tabla (1)

- Declarativas → solamente claves únicas, claves primarias, autoreferencias

```
Create Table Emp
(empno      number(4,0)      not null <empno restricción de atributo>,
ename      varchar2(20)    not null <ename restricción de atributo>,
tarea      varchar2(10)    not null <tarea restricción de atributo>,
dias_vacaciones number(3,0) not null <dias_vacaciones restricción de atributo>,
<Emp restricción de fila>,
primary key (empno));
```

- Todas las restantes requieren implementación procedural

Restricciones de tabla (2)

- Implementación procedural de checks
 1. Identificar las sentencias de DML que producen violaciones
 2. Derivar los mínimos chequeos para cada caso
 3. Construir los triggers
 4. Investigar la concurrencia

Restriccion de Tablas (3)

1. Sentencias de DML que pueden violar
 - Predicado para las columnas involucradas (QUE)

```
/*No se permiten vacaciones adyacentes */  
( $\forall s, t \in T: (s(EMPNO) = t(EMPNO) \wedge s(PRIMER\_DIA) < t(PRIMER\_DIA))$   
   $\Rightarrow (\exists d \in DIALABORAL: s(ULTIMO\_DAY) < d < t(PRIMER\_DIA))$   
)
```

- Construir tablas de casos
 - 1 Insert
 - 1 Delete
 - 7 Updates
 - > 2³-1

Caso	EF	Pdia	Ultia
1	X		
2		X	
3			X
4	X	X	
5		X	X
6	X		X
7	X	X	X

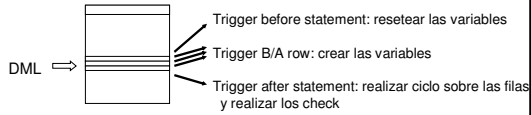
Restriccion de Tablas (4)

2. Derivar los mínimos chequeos
 - Puede ser formalmente hecho utilizando reglas de reescritura del cálculo de predicados
 - Comenzar con Insert, luego Delete, luego Update(s)
 - si el delete no requiere check
 - => todos los updates requerirán 'insert'-checks
 - Updates conflictivos → Restricciones de Transición extra
 - Construir función(es) PL/SQL para implementar check(s)
 - Típicamente tienen PK, rowid u otros valores de columna de las filas afectadas como parámetro de entrada

Restricciones de Tabla (5)

3. Construir triggers

- Llamadas funciones de chequeo, implementadas como triggers before, for each row
 - Deben leer otras filas en la misma tabla (que en algunos DBMS trae problemas, ej. Oracle)



Restricciones de Tabla (6)

4. Investigar la concurrencia

- Usuario A:
 - insert into vac values (1,sysdate,sysdate+5), exitoso
- Usuario B:
 - insert into vac values (1,sysdate+6,sysdate+8), exitoso
- Usuario A y B realizan el commit (en cualquier orden)
 - Violación → los registros de vacaciones estan adyacentes
- En este caso el trigger debería realizar un lock del registro del empleado

Detalles sobre concurrencia

- Regla general:
 - Realizar un lock de todos los datos utilizados en el check
 - 'Select for update' da locks exclusivos
- Para los chequeos el 'read' locks será suficiente en la mayoría de los casos
Un read lock:
 - Impide que otras transacciones promuevan un write lock sobre la fila

Restricciones de base de datos

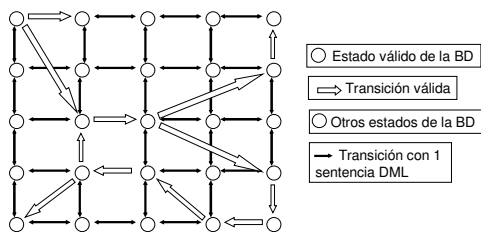
- Declarativas → Solamente Foreign Key > 1 tabla
- Todo el resto procedurales
 - Mismo proceso que el utilizado para las restricciones de tablas
 - Investigar los casos por tabla involucrada
 - Potencialmente muchos casos a ser investigados ($2^{n+1} + 2^{n+1} + \dots$)

Restricciones de transición

- Usualmente son simples
 - No se permite que se decrementen los salarios
- Usa :old y :new disponibles en los triggers for each row

- Necesita una sub-clasificación
 - Y guías de implementación

La causa para diferir...



La causa para diferir: SQL...

- Las sentencias de DML operan sobre una única tabla
 - Algunas transiciones requieren DML sobre más de una tabla
 - Restricciones de base de datos
- "Cada producto tiene al menos 2 componentes" y "cada componente pertenece a un producto"
 - Se debe elegir que una de las dos sea implementada como diferida

La causa para diferir: SQL...

- La sentencia DML opera de un solo modo
 - Insert, o Update, o Delete
 - Algunas transiciones requieren más de un tipo de sentencia de DML para la misma tabla (tuplas diferentes)
 - Restricciones de tabla
- La cantidad de productos de tipo A más 2 veces la cantidad de productos de tipo B debe ser igual a 0 o 100
 - Se comienza insertando 100 productos de tipo A
 - No se puede ingresar productos de tipo B sin violar la restricción de manera temporal .

Bibliografía

- Twelve Rules for Business Rules - 2000 C. J. Date.
<http://www.versata.com/documents/ChrisDate.whitepaper.pdf>
- Business Rules Group <http://www.businessrulesgroup.org/home-brg.shtml>
- Business Rules Community <http://www.brcommunity.com/>
