

Integridad

Un sistema de gestión de bases de datos (DBMS), centralizado o distribuido, debe garantizar la integridad de una base de datos, condición que se verificará si dicha base de datos satisface un conjunto de reglas denominadas restricciones de integridad semántica o simplemente *restricciones de integridad*, que pueden estar explícita o implícitamente declaradas en el diccionario de datos. Por ejemplo, una restricción de integridad puede imponer la igualdad de los valores de dos atributos de una base de datos (el valor del atributo W# en la relación BEBER debe ser igual al valor del atributo W# en la relación VINOS), o puede manifestar una relación compleja entre múltiples datos (la suma de las cantidades en la relación BEBER no deberá exceder a la suma de las cantidades en la relación COSECHAR). El problema del mantenimiento de la integridad, se traduce en el problema de rechazar aquellas actualizaciones que puedan violar las restricciones de integridad. La invalidez de tales actualizaciones puede ser causada por varios motivos:

1. Errores en la entrada de datos.
2. Accesos concurrentes a la bases de datos.
3. Fallos en el software o en el hardware.
4. Modificaciones no autorizadas de los datos.

Por estas razones, el mantenimiento de la integridad de una base de datos requiere:

1. Un control de la integridad semántica.
2. Un control de concurrencia.
3. Un control de veracidad
4. Protección contra accesos no autorizados.

Este tema estará dedicado al control de la integridad semántica.

Los mecanismos encargados del *control de integridad semántica* comprobarán qué restricciones de integridad no se violarán al intentar actualizar una base de datos. Asegurarán la integridad o bien rechazando aquellas actualizaciones que conduzcan a estados inconsistentes, o bien ejecutando acciones específicas que compensen el efecto de dichas actualizaciones. Un estado inconsistente de una base de datos se caracteriza por el hecho de que algunas restricciones de integridad son falsas. Para evitar estos estados inconsistentes se han propuesto dos tipos de métodos: *Métodos de detección*, que verifican que todas las restricciones de integridad se cumplen después de una o más actualizaciones, rechazándolas en caso contrario. Métodos preventivos, que verifican, antes de actualizar la base de datos, que no se viole ninguna de las restricciones de integridad.

1 Definición de restricciones de integridad

En este apartado, se clasificarán las restricciones de integridad de acuerdo a sus funciones. En el nivel más alta, se distinguirá entre restricciones de integridad estructural y restricciones de integridad de comportamiento. En un segundo nivel, se considerarán otros tipos de restricciones.

1.1 Restricciones de integridad estructural

Definición:

Restricción inherente a un modelo de datos, que expresa propiedades semánticas específicas de las estructuras básicas que utiliza el modelo.

Una restricción estructural se especifica, generalmente, como parte del esquema de definición de bases de datos. En el modelo relacional se pueden distinguir los siguientes tipos de restricciones de integridad estructural:

1. *Restricciones de llave única.* Especifican que una relación tiene una llave no nula compuesta por uno o más atributos. Por ejemplo, el número de vino (W#) es la llave primaria de la relación VINOS.
2. *Restricciones de integridad referencial.* Especifican que parte (o toda) de la llave de una relación se utiliza como llave en otra relación. Por ejemplo, el número de vino (W#) de la relación BEBER referencia a la llave primaria de la relación VINOS.
3. *Restricciones de dominio.* Restringen los valores que puede tener un atributo a un conjunto específico. Entre las restricciones de dominio se pueden distinguir: restricciones de tipo, que especifican el formato del dominio y las operaciones permitidas (el atributo CANTIDAD es de tipo real), y las restricciones de rango, que restringen los posibles valores de un dominio (por ejemplo, el atributo CANTIDAD debe tomar valores en un rango comprendido entre 10 y 30).
4. *Restricciones no nulas.* Elimina el valor nulo del conjunto de posibles valores para un atributo determinado. Por ejemplo, no se permite el valor nulo para el atributo VIÑEDO en la relación VINOS.

Las restricciones estructurales fueron introducidas en el modelo relacional para extender la semántica de los datos y poder modelar ciertas capacidades. Concretamente, las restricciones de llave única aparecieron para evitar la existencia de tuplas duplicadas en una relación, es decir, para poder captar, sin ningún tipo de ambigüedad, las entidades del mundo real. Por otra parte, las restricciones de llave y las restricciones de dominio son insuficientes para modelar la semántica de los vínculos entre dos o más relaciones. Fue en 1981 cuando se introdujo la integridad referencial como un tipo de restricción estructural, permitiendo al diseñador declarar en una relación llaves externas que captan la semántica de los vínculos entre relaciones.

1.2 Restricciones de integridad de comportamiento

Definición:

Restricción de integridad que modela una propiedad semántica de la base de datos y que regula el comportamiento de las aplicaciones

Las restricciones de integridad de comportamiento se introducen en la definición del esquema como un añadido. En particular, en el modelo relacional se distinguen varios tipos de restricciones de comportamiento, pudiendo expresarse, muchas de ellas, mediante aserciones en lógica de primer orden. Las restricciones de comportamiento pueden representar relaciones internas dentro de una relación, tales como:

Dependencias funcionales, por ejemplo, el número de vino (W#) determina de forma única al viñedo.

Dependencias multivaluadas, por ejemplo, en una relación PROFESOR = (PROF, CLASES, ALUMNOS), las clases impartidas por un profesor son independientes de los nombres de sus alumnos.

Dependencias de conjunto, que restringen los posibles valores de una función de conjunto en una relación; por ejemplo, la suma de las cantidades bebidas de vino no debe exceder de 10.

Las restricciones de comportamiento también pueden definir interrelaciones, tales como:

Dependencias de inclusión, que especifican que una o más columnas de una relación están incluidas en otras relaciones; por ejemplo, la columna CIUDAD de la relación BEBEDOR debe aparecer como columna en la relación ESTADOS.

Dependencias ecuacionales, que imponen la igualdad o desigualdad de dos expresiones aritméticas que se calculan utilizando valores de atributos o funciones de valores de atributos (incluyendo funciones de conjunto); por ejemplo, la cantidad de stock para cada vino debe ser siempre igual a la cantidad comprada menos la cantidad bebida.

Otro tipo importante de restricciones de comportamiento es que tiene que ver con la *invarianza ecuacional*. Este tipo de restricción, impone que el resultado de una expresión aritmética que se calcula utilizando valores de atributos o funciones de valores de atributos (incluyendo funciones de conjunto), no debe cambiar por las actualizaciones realizadas por un programa. Por ejemplo, la suma de los balances en la base de datos de un banco debe permanecer constante.

Finalmente, indicar que ciertas restricciones de integridad son dependientes del tiempo; de esta forma, se puede imponer, por ejemplo, que solamente se permitirán determinados cambios en los datos en determinados instantes, o cambios relativos a otros cambios; por ejemplo, el atributo GRADUACION de la relación VINOS sólo puede decrecer. Este tipo de restricciones, se denominan *restricciones de transición*.

2 Manipulación de restricciones de integridad

En los sistemas de gestión de bases de datos relacionales, las restricciones de integridad se manipulan utilizando lenguajes no procedurales. En SQL, determinadas restricciones

estructurales se pueden definir al crear una relación, dentro de la instrucción CREATE. Esto incluye restricciones de llave única (cláusula PRIMARY KEY), restricciones de dominio (CHAR, NUMBER, ... para las de tipo y cláusula CHECK para las de rango), restricciones no nulas (cláusula NOT NULL) y restricciones de integridad referencial (cláusula REFERENCES). Además, en determinadas versiones de SQL se incluye, para definir restricciones de comportamiento, la siguiente instrucción:

```
ASSERT <nombre de restricción> ON [<operación> OF] <nombre de relación> :  
    <condición de búsqueda>
```

donde “nombre de restricción” es un nombre que se da a la restricción para poder referenciarla en el diccionario de datos. También es posible especificar restricciones que se tiene que cumplir solamente cuando se ejecutan determinadas operaciones sobre la relación. La “operación” es una de las operaciones de actualización típicas de SQL (UPDATE, INSERT o DELETE). La “condición de búsqueda” es una condición de búsqueda de SQL, por ejemplo, “la media de graduación de los vinos no debe exceder de 12”, se puede expresar de la siguiente forma:

```
ASSERT MEDVINO ON VINOS: AVG(GRADUACION) ≤ 12
```

O, por ejemplo, “un vino únicamente se podrá eliminar de la relación VINOS, si las correspondientes cantidades de ese vino en la relación BEBER son 0”:

```
ASSERT DELVINO ON DELETION OF VINOS:  
(SUM(CANTIDAD) FROM BEBER WHERE BEBER.W# = VINOS.W#) = 0
```

También es posible usar las palabras clave OLD y NEW para especificar restricciones de transición (dependientes del tiempo), por ejemplo, “una graduación solamente puede decrecer”:

```
ASSERT UPDVINO ONUPDATE OF VINOS:  
NEW.GRADUACION < OLD.GRADUACION
```

Un DBMS que ofreciera un subsistema completo de control de integridad, debería incluir un lenguaje de manipulación de restricciones de integridad. Las características principales de ese lenguaje deberían ser simplicidad y completitud, en el sentido de que todas las restricciones de integridad que se especifiquen se puedan definir con instrucciones simples.

Por simplicidad, considérese que el efecto de violar una restricción de integridad es siempre el de rechazar las tuplas que no satisfacen dicha restricción. Sin embargo, algunos sistemas de gestión de bases de datos son más sofisticados. Por ejemplo, existen determinados DBMSs, entre ellos ORACLE, que permiten especificar ciertas acciones compensatorias. De forma más general, los *disparadores* (*triggers*) pueden combinarse con las restricciones de integridad para especificar las acciones a realizar en el caso de que una actualización viole una restricción de integridad.

Definición:

Un disparador (trigger) es una regla que especifica una acción y que se activa cuando una condición es verdad en la base de datos.

Un disparador, a menudo, se expresa como una regla de la forma: IF <condición de búsqueda> THEN <acción>. La acción, generalmente, es una instrucción de actualizar (UPDATE), insertar (INSERT) o borrar (DELETE). De esta manera, los disparadores se pueden utilizar para restaurar la consistencia de una base de datos tras la violación de una restricción. Por ejemplo, si W# en la relación BEBER se refiere a un vino que debe

existir en la relación VINOS (es decir, existe una restricción de integridad referencial de BEBER a VINOS), la integridad de la base de datos después de borrar un determinado vino en VINOS, puede ser reintegrada por un disparador cuya acción fuera borrar todas las tuplas de BEBER que hagan referencia al vino borrado de VINOS.

Bibliografia

1. G. Gardarin and P. Valduriez: *Relational database and knowledge bases*, Addison Wesley, 1989.