

ACERCA DE LA MODELACIÓN DE DATOS

Por: Lic. Ana M^o. García Pérez, Dra. Luisa González González, Lic. Abel Rodríguez Morffi *

RESUMEN

En este artículo se presenta un esbozo de la problemática de la modelación de datos. Se describen los principales modelos de datos conocidos resaltando sus bondades y deficiencias y utilizando un conjunto de referencias a artículos y textos de relevancia. Se exponen los requerimientos de nuevas aplicaciones intensivas de datos, que han originado investigaciones en la búsqueda de nuevos modelos que los satisfagan.

Palabras Clave: Bases de Datos, Modelos de datos, Representación del Conocimiento.

INTRODUCCION

Un *modelo de datos* según [Brodie y otros (1984)] es una colección de conceptos bien definidos matemáticamente que ayudan a considerar y expresar las propiedades estáticas y dinámicas de una aplicación, y aunque la mayoría de los modelos de datos han evolucionado intuitivamente y no han sido formalmente definidos, se supone generalmente que de una aplicación deben caracterizar:

- Propiedades estáticas tales como objetos, sus propiedades (llamadas con frecuencia atributos) e interrelaciones entre objetos (consideradas en ocasiones como propiedades especiales de los objetos).
- Propiedades dinámicas tales como operaciones sobre los objetos, propiedades de las operaciones e interrelaciones entre las mismas.

- Reglas de integridad sobre los objetos que reflejen un estado válido de la base de datos.

El resultado de la modelación de datos es una representación que tiene dos componentes: propiedades estáticas, definidas en un esquema y propiedades dinámicas, definidas como especificaciones para programas de aplicación, solicitudes y reportes. Un *esquema o modelo conceptual* consiste en una definición de todos los tipos de objetos de una aplicación incluyendo sus atributos, interrelaciones y restricciones estáticas.

En correspondencia con el esquema, en ocasiones llamado metadato, se tiene un depósito de datos llamado base de datos o instancia del esquema.

Las herramientas asociadas a un modelo de datos son los lenguajes para definir, manipular y recuperar la información. La mayoría de los SGBD contienen un Lenguaje de Definición de Datos (LDD) para la definición de esquemas, un Lenguaje de Manipulación de Datos (LMD) para escribir programas y un Lenguaje de Solicitud, del inglés *Query Language (QL)*, para formular solicitudes. Un SGBD es un sistema que instrumenta las herramientas asociadas con un modelo de datos.

El modelo de datos relacional

El modelo de datos relacional se apoya en el concepto matemático de relación. Una relación es un conjunto de n-uplos. Un uplo

* Docentes investigadores de la Facultad de Matemática, Física y Computación, Universidad Central de Las Villas, Cuba.

puede representar entidades u objetos de un mismo conjunto así como también interrelaciones entre entidades de diferentes conjuntos. En este último caso deben observarse ciertos criterios sobre la representación de la interrelación en uno de los conjuntos o su representación totalmente independiente. Los criterios fundamentales que se manejan son: representar explícitamente la interrelación mediante las claves de los conjuntos que están siendo interrelacionados y cualquier otro atributo que requiera la interrelación o representarla implícitamente en uno de los conjuntos (esto es posible cuando la cardinalidad de la interrelación contiene al menos un uno y su grado es uno o dos).

Por su simplicidad y base matemática, el modelo relacional ha proporcionado más fundamentos intuitivos que sus principales competidores: el modelo jerárquico y el de redes. Por la novedad e importancia del modelo relacional se le concedió a Codd en 1981 el premio Turing.

La popularidad del modelo relacional es ampliamente reconocida por diversas razones, como son: es conceptualmente simple, fácil de usar y de instrumentar, idealmente adecuado para modelar problemas que puedan ser descritos naturalmente en forma tabular; proporciona una importante capacidad de recuperación de conjuntos de artículos simultáneamente a partir de los operadores del álgebra y el cálculo relacional; tiene fundamentos teóricos muy fuertes y tiene formalizadas técnicas de modelación y diseño como la teoría de la normalización, la teoría de diseño relacional lógica [Teorey y otros (1986)] y la técnica de modelación de objetos [Blaha y otros (1988)], entre otras. Una contribución importante de este modelo es que ayuda a alcanzar un alto grado de independencia de

datos, motivación original para el desarrollo del modelo.

A pesar de sus indudables ventajas existen variados ejemplos de áreas en que el modelo relacional presenta deficiencias, al menos como fue originalmente formulado; las áreas que son generalmente mencionadas en la literatura son CAD-CAM, CASE (Computer-Aided Software Engineering) y la automatización de la actividad de oficina.

Las deficiencias que se señalan al modelo se refieren fundamentalmente a que es demasiado simple para reflejar la complejidad del mundo real; en aplicaciones ingenieriles no permite modelar objetos en forma natural, ni manipular y recuperar tales objetos en forma familiar a ingenieros; en particular no permite manipular objetos como una entidad completa de la base de datos, sino que éstos son representados en el esquema conceptual con restricciones impuestas por el modelo que no necesariamente constituyen una transformación natural. La restricción de la primera forma normal incide en que se pierda la semántica inherente a algunos problemas reales; lo que hace necesario por lo general, para reconstruirlos, la ejecución de acoples mediante claves foráneas.

A muchos sistemas relacionales les falta capacidad para especificar restricciones de integridad en la definición de la base de datos. El problema es trasladado a los programadores de aplicación reduciendo su productividad y permitiendo que importantes pruebas sean omitidos o especificados de manera inconsistente por diferentes programadores.

Los modelos de datos semánticos

Desde la década del 70 han evolucionado numerosos modelos de datos como un

intento de representar contenido semántico en los datos. El incremento de propiedades semánticas al modelo relacional es un objetivo señalado por Codd desde el año 1979 [Codd (1979)] y ha sido objeto de posteriores investigaciones; los modelos de datos semánticos surgen como un requerimiento para dar mayor expresividad conceptual a los modelos de datos, incluido el relacional.

El concepto de modelo de datos semántico se ha usado para reflejar la representación de un amplio rango de información significativamente más convincente que los anteriores modelos; contribuciones importantes aparecen en [Smith y Smith (1976), Hammer y McLeod (1981)].

La idea básica de la modelación semántica es identificar un conjunto de construcciones generalmente útiles en el sentido en que se presentan de alguna manera en una amplia variedad de aplicaciones [Date (1990)]. Ejemplos de tales construcciones son las claves primarias, claves foráneas y dominios del modelo relacional [Ullman (1990)]. Si en un área de aplicación se pueden identificar y formalizar tales construcciones con sus operadores respectivos, pueden incorporarse a sistemas de bases de datos; sin embargo, los intentos de proporcionar construcciones útiles con un nivel de abstracción superior al modelo relacional han tenido poco éxito. Es ampliamente reconocido que a las generaciones precedentes de modelos de datos les faltan soporte para interrelaciones, abstracción de dato, herencia, restricciones de integridad, objetos no estructurados y propiedades dinámicas de una aplicación; no obstante ninguna propuesta ha tenido aceptación excepto el impacto que han tenido en las metodologías de diseño.

Las construcciones de los modelos semánticos están más estrechamente relacionadas con la descripción del mundo real que con aspectos de representación física.

Los modelos de datos semánticos han sido fundamentalmente usados para la representación de esquemas y como herramientas de diseño. El modelo de datos semántico más popular es el Entidad-Relación (ER) propuesto por Chen [Chen (1976)]. Esta es la técnica más ampliamente aceptada para la modelación conceptual de los datos; como su nombre indica este modelo soporta entidades e interrelaciones. Las interrelaciones originalmente ofrecidas son la is-a, asociación y agregación. Las restricciones para las operaciones de inserción y eliminación se definen usando dependencias de existencia entre entidades débiles y fuertes.

A pesar de su utilidad, el modelo ER no abarca completamente los requerimientos de la modelación, especialmente para aplicaciones complejas; al mismo le faltan subestructuras para entidades e interrelaciones, lo cual es aclamado por investigadores que desarrollan extensiones del modelo. Representativo de esta tendencia es el modelo Entidad-Relación-Extendido expuesto por Teorey en su metodología de diseño relacional lógica [Teorey y otros (1986)], la misma considera una técnica de modelación gráfica que soporta los conceptos básicos de entidades, jerarquías de generalización y de subconjunto, agregación y asociación.

Los modelos orientados a objetos

Se han realizado numerosas investigaciones relacionadas con la tecnología orientada a objetos en áreas como los lenguajes de programación, técnicas para el análisis y

diseño de sistemas, sistemas de bases de datos, etc. El paradigma orientado a objetos se caracteriza en un sentido general por una agrupación de información con el concepto o entidad con que se relaciona. Este paradigma se usa ampliamente. Una de las razones para este creciente interés es que permite modelar muchos conceptos del mundo real en una forma directa y natural [Khoshafian (1990)].

La tecnología orientada a objetos en bases de datos está relacionada con el objetivo de hacer los sistemas más inteligentes al proporcionar construcciones no orientadas a la computadora tales como artículos y atributos; en su lugar el usuario trata con objetos, más estrechamente relacionados con su contrapartida en el mundo real.

El concepto de objeto se refiere a una colección de información que incluye atributos y métodos. Los métodos son grupos de sentencias que se refieren a los atributos. Los objetos pueden ser también interrelacionados. Los atributos, métodos e interrelaciones pertenecientes a un objeto definen su tipo. Los objetos tienen una identidad única que nunca cambia, frecuentemente referida como su ID. Los atributos e interrelaciones de un objeto (su estado) pueden cambiar con el tiempo pero su identidad no. Un objeto nunca es identificado por su estado sino por su ID. En la práctica los ID deben ser generados arbitrariamente y ser incluidos como atributos para asegurar la unicidad deseada.

El mecanismo de herencia es usado para compartir atributos y métodos de un tipo más general a uno más específico. Detalles de éstos y otros conceptos asociados al paradigma orientado a objetos pueden ser

profundizados en [Date (1990), Patón y Díaz (1991)].

Durante los últimos años se han propuesto varios sistemas orientados a objetos como una nueva alternativa para aplicaciones técnicas. En el área de las bases de datos se distinguen dos tipos de orientación a objetos: estructural y de comportamiento [Patón y Díaz (1991)].

La orientación a objetos estructural tiene sus orígenes en las técnicas de bases de datos. Versiones típicas las constituyen extensiones de los modelos relacional y de ER. Los modelos de datos orientados a objetos estructuralmente proporcionan facilidades para transformar y recuperar objetos complejos en estructuras de la base de datos pero carecen de construcciones para manipular estos objetos de manera familiar a los usuarios.

La orientación a objetos por comportamiento tiene sus orígenes en los lenguajes de programación y es más orientada a la aplicación.

Los SGBD orientados a objetos (SGBDOO) siguen por lo general dos tendencias: extensiones de SGBD relacionales o extensiones de lenguajes de programación orientados a objetos; en ambas tendencias está presente el objetivo de apoyarse en herramientas existentes.

Los modelos de objetos complejos

El concepto de objeto complejo aunque se considera un importante principio de la orientación a objetos, aparece muy vinculado a la problemática de objetos compuestos.

Se denomina *objeto complejo* a aquél en que un atributo puede ser una referencia a otro objeto o conjunto de objetos (Bertino y Martino, 1993).

Los objetos compuestos han originado un tratamiento adicional de la interrelación PART-OF («parte de»), esta interrelación expresa de manera declarativa dos cosas entre un objeto y sus componentes, primero una taxonomía y segundo una referencia entre instancias.

Los modelos de objetos complejos intentan extender el modelo relacional para permitir solicitudes más expresivas, mientras mantiene sus sólidos fundamentos teóricos [Khoshafian (1990)]; estos modelos contienen una o más de las siguiente extensiones: conjuntos de valores atómicos, atributos tipo uplo, conjuntos de uplos (relaciones anidadas), identidad de objeto, etc.

Los SGBDOO han sido influenciados por las alternativas de modelación de los modelos semánticos y de objetos complejos.

En la actualidad también se maneja la tendencia hacia la integración de ambos modelos: el relacional y el orientado a objetos (Date, 1994)

Los modelos de datos en el diseño Ingenieril

Desde inicios de la década del 80 el modelo relacional emergió como una organización de datos que rápidamente fue usada por usuarios con intereses no sólo investigativos sino también de aplicaciones prácticas; las aplicaciones CAD fueron unas de las primeras en examinarse, en parte debido al crecimiento en ese tiempo de la industria asociada, así como también a causa del creciente interés alrededor del diseño de circuitos en las universidades. En pocos años surgieron tres tendencias para el desarrollo de nuevos modelos de datos. La primera fue extender el modelo relacional, la segunda extender el modelo ER y la tercera desarrollar

nuevos modelos de datos basados en los conceptos orientados a objetos.

Es difícil modelar jerarquías con tablas planas, además el concepto de transacción considerado por la mayoría de los sistemas relacionales no se adapta bien a las necesidades del diseño; alrededor de ambos aspectos es que se han hecho las principales extensiones al modelo relacional.

Uno de los primeros y más importantes artículos que describen extensiones del modelo relacional para manipular datos de diseño ingenieril fue publicado en [Haskin y Lorie (1982)], en el mismo se muestran extensiones del sistema R con objetos complejos y mecanismos para transacciones.

Una de las formas en que el modelo ER se extendió para aplicaciones CAD fue añadiendo nuevas interrelaciones en el modelo tales como is-derived-from, is-part-of, is-equivalent-to, etc. Ejemplo de una extensión hecha al modelo es el sistema Versión Server [Katz (1986)].

La mayoría de los modelos de datos soportan conceptos tales como objetos, operaciones y restricciones; sin embargo hay diferencias en los formalismos proporcionados para su representación. De los tres conceptos, los especialistas en computación manejan con mayor frecuencia los dos primeros, mientras que lo relacionado con las restricciones ha recibido menor atención, y precisamente resultan ser de un interés especial en las aplicaciones ingenieriles.

Las restricciones de integridad son de los problemas que emergen de la representación del conocimiento [Goutas y otros (1991)]. En [Ballou y otros (1988)] se presenta una propuesta de integración de

bases de datos deductivas con sistemas de bases de datos orientados a objeto que parece ser atractiva. En [Goutas y otros (1991)] se toma dicha idea y se plantea la incorporación de reglas en un SGBDOO para representar restricciones que reflejen las dependencias entre objetos.

Los modelos de datos en aplicaciones multimedia

La utilización de multimedia redundante en grandes beneficios cuando se trata de manejar grandes bases de datos de distintos medios. Los documentos voluminosos, las bases de datos con gran cantidad de referencias cruzadas o aquellas que han de ser consultadas por tipos de personas con intereses muy diferentes para recuperar información muy divergente, son candidatos claros a este mundo.

Podría pensarse que puesto que los conceptos intuitivos que manejamos en hipermedia (hipertexto multimedia) son fáciles de comprender y de utilizar por no expertos, la creación de un sistema hipertexto debe tener baja complejidad, y sin embargo no es esto lo que ocurre.

Es bastante común perderse en el proceso de navegación. Lo que suele definirse como «pérdida en el hiperespacio o en la tela de araña», se caracteriza de varias maneras: el lector no sabe volver a un nodo que le interesaba, o se olvida de lo que estaba buscando, o cambia de objetivo en la búsqueda. Se observa la falta de estructuras necesarias.

El sueño del hipertexto es que el lector pueda tener un rol más activo, como: añadir notas al hipertexto, que pueda reproducir el camino que ha seguido y que cree enlaces entre materias entre las que no existe previamente

uniones si su experiencia le hace ver que dicha relación existe.

Un ambiente de diseño de hipertextos debe integrar herramientas que no sólo faciliten y hagan eficiente el proceso, sino también que manipulen aspectos formales (Nanard y Nanard, 1995).

Una estructura de hipertexto debe ser descrita en términos de tipos lógicos, independientemente del contenido real de los nodos. De esta forma, trabajando en términos de instancias y en términos de abstracciones, se permitirá al diseñador alternar entre los modos top-down y bottom-up.

Para facilitar la creación de prototipos, los ambientes de trabajo deberán permitir al diseñador conceptualizar una estructura general y ayudarle a instanciarla.

Los modelos de objetos complejos deben jugar un papel importante en la formalización de la conceptualización, he ahí un problema de investigación actual.

CONCLUSIONES

En la actualidad han emergido un conjunto de nuevas aplicaciones intensivas de datos que demandan la formalización de modelos de datos que permitan expresar, de la manera más natural y directa posible, la naturaleza de los objetos del mundo real. Ya que entre los objetos de estas aplicaciones existen numerosas interdependencias, se requiere que los modelos no sólo representen la estructura de los objetos individuales sino también las interrelaciones entre éstos y su comportamiento a través del tiempo.

BIBLIOGRAFÍA

- BALLOU, N.; H. CHOU, J. F. GARZA, W. KIM, C. PETRIE, D. RUSSINOFF, D. STEINER y D. WOELK.(1988) Coupling an expert system shell with an Object-Oriented Database System. J.O.O.P. Vol. 1, N° 2, Junio 1988, pp 12-21.
- BERTINO, E. and L. MARTINO.(1993). Object-Oriented Database Systems: concepts and architectures. Addison-Wesley Publishing Co.
- BLAHA, M.R.; W. J. PREMERLANI y J. RUMBAUGH (1988) Relational Database Design using an object-oriented methodology. Comm of the ACM 31 (4), Abril 1988, pp. 414-427.
- BRODIE, M.; J. MYLOPOULOS y J. SCHMIDT (1984). On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases, and Programming Languages. New York, Springer-Verlag.
- CHEN, P. P. (1976). The entity-relationship model: toward a unified view of data. ACM Trans. Database Syst., Vol.1, N° 1, Marzo, 1976, pp. 9-36.
- CODD, E. F. (1979) Extending the database relational model to capture more meaning. ACM Trans. Database Syst., Vol. 4, N° 4, Diciembre, 1979, pp. 397-434.
- DATE, C. J. (1990) Object-Oriented Systems. New York, Addison-Wesley, Reading, Mass.
- DATE, C. J. (1993) Introduction to Database Systems, Vol. 1 5th. ed, Addison-Wesley Publishing Co.
- DATE, C. J. (1994) Oh Oh Relational: toward an OO/Relational Rapprochement. Database Programming and Design. Oct. 1994 7 (10), 23-27.
- DEUTSCH, P. L. (1991) Object-Oriented Software Technology. IEEE computer 24 (9), 112-113.
- DÍAZ, O. y N. W. PATÓN (1991) Sharing behaviour in an object-oriented database using a rule-based mechanism. In Proc. 9th British National Conference on Databases, 1991, pp. 1-21.
- GOUTAS, S., P. SOUPOS y CHRISTODOULAKIS (1991) Formalization of object-oriented database model with rules. Information and Software Technology, Vol.33, No.10, Diciembre, 1991, pp. 741-757.
- HAMMER, M. y D. McLEOD. (1981) Database descriptions with SDM: A semantic database model. ACM Trans. on Database Systems 6 (3) Sept. 1981, pp. 351-386.
- HASKIN, R.L. y R. A. LORIE. (1982) On extendings the functions of a relational database system. In Proc. ACM SIGMOD CONF. New York, Junio, 1982, p. 207-212.

HUGHES, J. G. (1991) Object-Oriented Databases. Prentice-Hall, C.A.R. Hoare series ed.

KATZ, R., M. ANWARRUDIN y E. CHANG. (1986) A version server for computer-aided design data. In Proc. ACM/IEEE Design Automation Conf. Las Vegas, Junio 1986, pp. 27-33.

KOSHAFIAN, S., M. J. FRANKLIN y M. CAREY. (1990) Storage management for persistent complex objects. Information Systems, 15 (3), 1990, pp. 303-320.

NANARD, J. y M. NANARD. (1995) Hypertext design environments and the hypertext design process. Comm of the ACM Agosto 1995 38 (8), pp. 49-56.

ORTEGA, M., F. RUIZ, J. BRAVO, J. RUIZ y E. DOMÍNGUEZ. (1995) Hipermedia. En Proc. Conferencia Internacional sobre Informática Educativa, Universidad de Oviedo, España, 1995.

PEHONG, CHEN, M. A. HARRISON. (1988) Multiple representation document development. IEEE Computer, January 1988, 15-31.

SMITH, J. M. y D. C. SMITH. (1976) Database abstractions: Aggregation and Generalization. ACM TODS, 2 (2) Junio 1976, pp. 105-133.

TEOREY, T., D. YANG y J. P. FRY (1986) A logical design methodology for relational databases using the extended entity-relationship model. ACM Computing Surveys, 18 (2) Junio 1986, pp. 197-222.

ULLMAN, J. D. (1990) Principles of Database and Knowledge-based Systems. Computer Science Press, Rockville.