

Investigación básica y aplicada en Bases de Datos Deductivas

H. Liberatori, J. Paganini, C. González, S. Figueroa, M. Tejerina, A. Vega, C. Castillo, V. Battezzati

Facultad de Ingeniería/ Universidad Nacional de Jujuy

Ítalo Palanca N° 10 San Salvador de Jujuy (4600) Tel 0388 4221582

hliberatori@hotmail.com; jhpaganini@fi.unju.edu.ar; scgonzalez@fi.unju.edu.ar; tkdfigueroa@hotmail.com;
mariotejerina@gmail.com; arielalejandrovega@gmail.com; ce_al_castillo@yahoo.com.ar,
vbattezzati@fi.unju.edu.ar

Resumen

Con el proyecto presente: “investigación básica y aplicada en Bases de Datos Deductivas”; se pretende avanzar en el estudio, aplicaciones y desarrollo de técnicas para el procesamiento de consultas recursivas para bases de datos deductivas, tendientes a ofrecer alternativas de nuevas prestaciones en Bases de datos relacionales. Constituyendo, por sus características en un encuentro entre conceptos teóricos provenientes de la Lógica Computacional, y del Álgebra Relacional para bases de datos; con técnicas de programación y desarrollo para la implementación de algoritmos concretos de aplicación.

Cabe destacar también que toda la investigación y desarrollo se formaliza y documenta con técnicas de especificación formal del método RAISE

El presente proyecto está categorizado con categoría “A”, la máxima, que lo sitúa en la línea de prioridades de la Facultad de Ingeniería de la UNJu; y se enmarca dentro de un proyecto general de la línea PICTO tendiente a implementar una base de datos de composición de alimentos para Argentina y Latinoamérica.

Otro aspecto que contempla el presente proyecto es el de contribuir a la formación de recursos humanos; ya que como parte del mismo se desarrolla una tesis de maestría en Ingeniería del Software y un proyecto final para el grado de Ingeniero Informático.

Palabras clave:

Bases de datos deductivas; Algoritmos Lógicos; RAISE, Datalog,

Contexto

El presente proyecto se inscribe dentro de las líneas prioritarias de investigación de la Facultad de Ingeniería. En el marco de la Universidad Nacional de Jujuy, está acreditado con categoría “A” y financiado por SECTER (Secretaría de Ciencia y Técnica y Estudios Regionales).

Los resultados a obtener, contribuirán al desarrollo de una Base de composición de Alimentos desarrollada mediante el auspicio de un

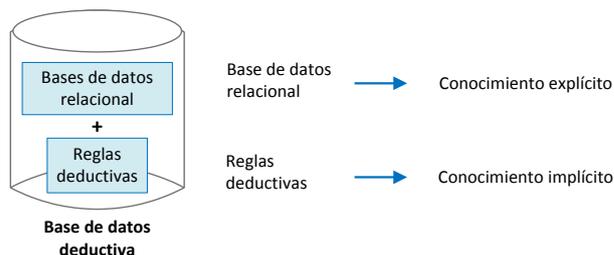
proyecto PICTO, de la Agencia Nacional de Promoción Científica y Tecnológica (FONCyT)

Los investigadores afectados a este proyecto provienen de las cátedras de Lógica Computacional y de la de Bases de Datos, a su vez integran el grupo del proyecto PICTO mencionado.

Introducción

Las bases de datos deductivas nacen de la necesidad de almacenar y utilizar “conocimiento” de una manera eficiente. Consisten de un conjunto de aserciones (hechos) referidos a la base de datos extensional y de un conjunto de reglas (axiomas) referidos a la base de datos intensional. La base de datos extensional se encuentra generalmente almacenada en una base de datos relacional y constituye el conocimiento básico de las bases de datos deductivas. Las reglas permiten obtener o deducir un nuevo conocimiento a partir de la base de datos extensional, el conocimiento no se encuentra almacenado directamente en la base de datos extensional. De esta manera, las bases de datos deductivas permiten inferir un nuevo conocimiento que puede ser utilizado para la toma de decisiones importantes de una organización.

Es conveniente situar las bases de datos deductivas en un contexto apropiado respecto de las Base de Datos Orientadas a Objetos. Estas últimas proporcionan un mecanismo de modelado natural para los objetos del mundo real, encapsulando su estructura junto con su comportamiento. En cambio las bases de datos deductivas extienden la capacidad expresiva de las bases de datos relacionales, proporcionando interrelaciones con el mundo real en forma de un conjunto de reglas, que permiten definir un conocimiento implícito a partir de datos existentes, como se muestra en la siguiente figura:



Una base de datos deductiva se compone de 3 conjuntos finitos: un conjunto de hechos, un

conjunto de reglas deductivas y un conjunto de restricciones de integridad:

- **Hechos:** se especifican de manera similar a las relaciones, excepto que no es necesario incluir los nombres de los atributos. Una tupla en una relación describe algún hecho del mundo real cuyo significado queda determinado en parte por los nombres de los atributos. En una base de datos deductiva, el significado del valor de un atributo en una tupla queda determinado exclusivamente por su posición dentro de la tupla.
- **Reglas:** presentan un parecido a las vistas relacionales, especifican relaciones virtuales que no están almacenadas, pero que se pueden formar a partir de los hechos, aplicando mecanismos de inferencia basados en las especificaciones de las reglas. La principal diferencia entre las reglas y las vistas, es que en las primeras puede haber recursión y por lo tanto, pueden producir relaciones virtuales que no es posible definir en términos de las vistas relacionales.

Mecanismos de Inferencia

Otra forma para interpretar el significado de las reglas implica definir un mecanismo de inferencia que el sistema utilice para deducir hechos a partir de reglas. Este mecanismo define una interpretación computacional del significado de las reglas.

El lenguaje de programación lógica Prolog se vale de su mecanismo de inferencia para definir el significado de las reglas y hechos en un programa. Sin embargo, en muchos programas simples de Prolog, el mecanismo de inferencia, infiere los hechos ya sea interpretando por la teoría de la demostración, o bien con un modelo mínimo en la interpretación por la teoría de modelos.

Los mecanismos de inferencia se basan en principios de la inteligencia artificial, basándose en mecanismos de inferencia tales como: inferencia ascendente (encadenamiento hacia adelante), inferencia descendente (encadenamiento hacia atrás). En el primer caso se parten de los hechos contenidos en la base de datos y se generan nuevos hechos utilizando las reglas de inferencia buscando llegar al hecho objetivo. En la inferencia descendente, por el contrario, se parte del hecho objetivo y se buscan hechos que lo satisfagan mediante el uso de reglas.

Los dos tipos principales de mecanismos de inferencia computacional se basan en la interpretación de reglas por la teoría de la demostración. Estos son los mecanismos de inferencia ascendente y descendente.

Evaluación Consultas Recursivas

Una de las características más notables de los sistemas de bases de datos deductivas es su soporte

para la recursión, mediante definiciones de reglas recursivas y por lo tanto, también de consultas recursivas. Una regla es recursiva cuando el predicado de la cabeza aparece también en el cuerpo. Existen diferentes tipos de reglas recursivas:

La mayor parte de las reglas de la vida real se pueden describir como reglas recursivas lineales. Se han definido algoritmos para ejecutar de manera eficiente conjuntos lineales de reglas.

El enfoque del procesamiento de consultas recursivas se clasifican de acuerdo a:

- Su Objetivo en: Evaluación y Reescritura.
- Técnicas de búsqueda en Bottom up y Top Down.

Técnicas para el procesamiento de consultas recursivas

Las consultas a bases de datos deductivas se pueden resolver mediante reglas no recursivas o reglas recursivas. Este último tipo de reglas presenta el problema de caer en bucles infinitos. Se han propuesto muchos métodos para resolver este problema, los cuales dieron como resultado una gran variedad de algoritmos que agregan la dificultad de elegir el adecuado que se deba aplicar.

En la clasificación de las técnicas de consulta se tienen en cuenta dos aspectos:

- Procesamiento de la consulta, que se puede descomponer en dos métodos:
 - Evaluación: se crea un esquema de evaluación de consultas que dé respuesta a la consulta y queda definida por: el dominio de aplicación y por un algoritmo que resuelva las consultas dadas por un conjunto de reglas.
 - Reescritura: se optimiza el esquema en una estrategia más eficiente, reescribiendo las reglas en función del argumento instanciado que proporciona la consulta.
- Mecanismo de inferencia que se utiliza para deducir hechos a partir de reglas. Los dos tipos principales de mecanismos de inferencia computacional se basan en la interpretación de reglas:
 - Encadenamiento hacia adelante (Bottom Up): el motor de inferencia parte de los hechos y aplica las reglas para generar hechos nuevos, éstos se comparan con el predicado que es el objetivo de la consulta para comprobar si coinciden.
 - Encadenamiento hacia atrás (Top Down): parte del predicado que es el objetivo de la consulta e intenta encontrar coincidencia con las variables que conduzcan a los hechos válidos en la base de datos.

La siguiente tabla muestra los aspectos definidos con anterioridad para la clasificación de las técnicas de consulta con ejemplos de los algoritmos:

Método de filtrado estático (static filtering)

Actualmente se está investigando, en el marco de este proyecto, el método de Filtrado estático, que introduce una optimización a los Métodos de Evaluación.

El método construye un grafo de axioma_relación a partir de un programa y una meta.

La optimización se logra imponiendo restricciones, las cuales deben ser cumplidas por las tuplas de la solución, es decir, los filtros se propagan desde el puerto de salida del nodo meta hacia todos los nodos relación, quedando las variables ligadas.

Las tuplas que no cumplan la condición son cortadas, tan pronto como sea posible, en la primera etapa de su flujo hacia el nodo meta. El método finaliza cuando un nodo de axioma no genera nuevas tuplas.

La estrategia asume un proceso de evaluación "Bottom_Up" a partir de un programa y una meta, construye un grafo llamado grafo axioma_relación, para ello se efectúa una reescritura de los argumentos para poder diferenciar el mismo argumento, en diferentes ocurrencias del mismo predicado.

Líneas de investigación y desarrollo

Este proyecto es nuevo, por lo tanto no posee una línea anterior a la cual pertenece. Sin embargo, puede establecerse que su línea de investigación deviene de dos fuentes; por un lado de un proyecto anterior sobre especificación formal y aplicaciones de la Lógica. ; y por otro de la necesidad de dotar de mecanismos de inferencia más eficientes a las bases de datos relacionales.

Resultados y Objetivos

Como resultados obtenidos se cita el estudio de PROLOG y DATALOG, como lenguajes lógicos de aplicación.

A la fecha, el equipo de investigación se encuentra abocado a la adaptación del método de filtrado estático para realizar consultas recursivas a la Base de Composición de Alimentos (ya implementada).

Como resultados futuros adaptar nuevos métodos de reescritura como por ejemplo reducción de variables y reducción de constantes.

Como objetivo final este proyecto propone desarrollar un sistema organizador y recuperador de la información, dotado de regla eurísticas sobre una base de datos relacional.

Formación de Recursos Humanos

Este proyecto se inició en Octubre del 2012, a la fecha no hay ninguna tesina de grado ni tesis presentada.

Sin embargo, se estima que para mayo se defienda una tesina (Proyecto Final) de grado; y en el curso de 2013 se culmine con una Tesis de Maestría acerca de Bases de Datos Deductivas.

Referencias

- E. F. Codd**, Relational Database: a practical foundation for productivity, Commun. ACM 25 (2) (1982) 109_117.
- M. P. Atkinson, F. Bancilhon, D. J. DeWitt, K. R. Dittrich, D. Maier, S. B. Zdonik**, The Object-Oriented Database System Manifesto., in: the First International Conference on Deductive and Object-Oriented Databases, 1989.
- M. Stonebraker, D. Moore, P. Brown**, Object-Relational DBMSs: Tracking the Next Great Wave, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1998.
- Kuhns, J.** "Logical aspects of question answering by computer" Technical report, Rand Corporation, RM-5428-Pr, 1967.
- Robinson, J. A.** "Logic: Form and Function: The Mechanization of Deductive Reasoning", University Press, Edinburgh, 1978.
- Kowalski, R.** "Logic for Problem Solving" North-Holland, Amsterdam, 1979.
- Codd, E.F.** "A Data Base sublanguage Founded on the Relational Calculus" Proc. of ACM SIGFIDET Workshop on Data Description, Access and Control, San Diego, California. 1971.
- Das, K.** "Deductive Databases and Logic Programming" Addison Wesley, 1992.
- Ramakrishnan, R.; Ullman, J.** "A Survey of Deductive Database Systems" Journal of Logia Programming 1995.
- Liu, M.** "Deductive Database Languages: Problems and solutions" ACM Computing Surveys, 31(1): 27-62, 1999.
- Wagner, G.** "Foundations of Knowledge Systems - with Applications to Databases and Agents", Kluwer Academic Publishers, 1998.
- R. Barbuti, P. Mancarella, D. Pedreschi and F. Turini.** Intensional negation of logic programs: Examples and implementation techniques. In Proc. International Joint Conference on Theory and Practice of Software Development (TAPSOFT'87), pages 96-110. Springer LNCS 250, 1987.
- R. Barbuti, P. Mancarella, D. Pedreschi and F. Turini.** A transformational approach to negation

in logic programming. *Journal of Logic Programming*, 8:201–228, 1990.

P. Bruscoli, F. Levi, and M.C. Meo. Compilative constructive negation in constraint logic programs. In *Proc. Colloquium on Trees in Algebra and Programming, (CAAP'94)*, pages 52–67. Springer LNCS 787, 1994.

Gallaire, H.; Minker, J. “Logic and Databases” Plenum Press 1978.

Levesque, H.J.; Lakemeyer, G. “The Logic of Knowledge Bases” MIT Press, 2001.

Leondes, C.T. (ed). “Knowledge-Based Systems. Vols. I-IV” Harcourt International, Academic Press 2000.

Tansel, A.; Clifford, J.; Gadia, S. et al. (eds.). “Temporal Databases” Benjamin Cummings, 1993.

Bouguettaya, Athman (Editor). “Ontologies and Databases”, Kluwer Academic Publishers, 1999.

Reiter, R. 1984. “Towards a logical reconstruction of Relational Database Theory”. *On Conceptual Modelling*. Springer-Verlag.

Lloyd, J.W. 1987. “Foundations of logic Programming”. Springer-Verlag.

Bjørner, Dines. “The SE Book”, Principle and Techniques of Software Engineering, Volume 3. Technical University of Denmark. February 2, 2004. Springer, Berlin, Heildenberg, New York, London, Paris, Milan Tokyo (free download DB Home Page) versión febrero / marzo 2004