

# Métodos de Acceso Espacio-Temporal: nuevas propuestas basadas en métodos existentes

Edilma O. Gagliardi <sup>(1)</sup>, María G. Dorzán <sup>(1)</sup>  
Fernando D. Carrasco, Juan C. García Sosa,  
Juan G. Gómez Barroso

Departamento de Informática

Facultad de Ciencias Físico, Matemáticas y  
Naturales

Universidad Nacional de San Luis, Argentina

{oli, mgdorzan, fdcarras, jcgarcia, jggomez}  
@unsl.edu.ar

Gilberto Gutiérrez Retamal

Departamento de Auditoría e  
Informática

Facultad de Ciencias Empresariales

Universidad del Bío-Bío, Chile

ggutierr@ubiobio.cl

## Resumen

Existen muchas aplicaciones que requieren manejar objetos espacio-temporales, es decir, objetos cuya posición espacial o forma cambia en distintos instantes de tiempo. Tales cambios deben ser manejados por un Sistema de Administración de Bases Datos el que debe proveer, entre otros servicios, métodos de acceso para procesar en forma eficiente consultas cuyos predicados contemplan relaciones espacio-temporales. Los tipos de consultas que han recibido mayor atención son a) *timestamp* o *timeslice*, que permite *recuperar todos los objetos que se encontraban en una cierta región en un instante dado t* y b) *Intervalo* la cual permite, *recuperar todos los objetos que encontraban en una cierta región en el intervalo de tiempo  $[t_1, t_2]$* . En este trabajo nos abocaremos al estudio de métodos de acceso espacio-temporal, con el objetivo de extenderlos para resolver eficientemente las consultas planteadas anteriormente, como también otros tipos de consultas tales como aquellas relacionadas con la trayectoria de los objetos u orientadas a los eventos que ocurren en el espacio en determinados instantes de tiempo.

## 1. Introducción

Las investigaciones en torno a los modelos de Bases de Datos Espaciales y Temporales se realizaron, en su mayoría, de manera independiente. Así, los trabajos de investigación en bases de datos espaciales se centralizaron en el modelado y la resolución de consultas, basándose en la geometría asociada a los objetos almacenados en una base de datos [2]. Y para el caso de las bases de datos temporales, se procuró incluir información acerca del pasado como atributos adicionales de los objetos. Actualmente, las Bases de Datos Espacio-Temporales en conjunto representan un tópico de estudio reciente, de mucho interés debido a la estrecha relación existente entre espacio y tiempo en una gran cantidad de aplicaciones en donde se debe modelar datos con componentes tanto espaciales como temporales.

Por ello, es necesario que una Base de Datos Espacio-Temporal sea capaz de representar modelos muy cercanos al mundo real, con todo el dinamismo que él implica, y administrar objetos que, básicamente, cambian su ubicación y/o forma a través del tiempo [3]. Se pueden encontrar ejemplos claros en áreas como las de transporte (vigilancia de tráfico), Ciencias Sociales (demografía), telecomunicaciones (telefonía celular), multimedia (películas animadas) e información geográfica (cambios de límites en terrenos), entre otras [8].

<sup>(1)</sup> Proyecto Tecnologías Avanzadas de Bases de Datos 22/F314, Departamento de Informática, UNSL.  
Proyecto AL2005 Geometría Computacional, UPM.

<sup>(2)</sup> Red Iberoamericana de Tecnologías del Software (RITOS2)

Por ejemplo, en un sistema de vigilancia satelital de una empresa de transportes puede ser interesante conocer la ubicación de las unidades a lo largo del tiempo. Así la información espacial se amplía con conceptos temporales. Para una aplicación como ésta resulta fundamental poder responder consultas con predicados espaciales y temporales, tales como: *¿Dónde se encuentra la unidad n° 12 ahora?* haciendo preciso el instante *ahora*; o *¿Qué unidades estaban más cercanas a la intersección de las rutas A y B ayer a las 14:00 hrs.?* también haciendo precisos los conceptos *más cercanas* y *ayer*. Otras podrían ser *¿Cuál fue el recorrido de la unidad n° 8 en la última semana?*, *¿Cuáles unidades se encontraban en la ruta A entre 04:00 hrs. y las 08:00 hrs. el día 14/02/05?*, *¿Cuáles unidades llegaron y cuáles salieron en la zona Z a las 20:00 hrs.?*, entre otras.

En este contexto surgen las Bases de Datos Espacio-Temporales, que permiten modelar en forma adecuada los objetos dinámicos del mundo real y responder en forma eficiente las consultas que involucran predicados espacio-temporales. Así como un Sistema de Administración de Bases de Datos (DBMS) provee métodos de acceso que evitan examinar todos los objetos al momento de responder consultas, también un Sistema de Bases de Datos Espacio-Temporal (STDBMS) debe contar con métodos que permitan construir índices que mejoren los tiempos de respuestas para las consultas espacio-temporales. Por ello, es fundamental contar con métodos de acceso que permitan seleccionar los objetos basándose en sus atributos espacio-temporales.

Esta presentación está organizada del siguiente modo. En la Sección 2 presentamos una introducción acerca de los métodos de acceso espacio-temporales; en la Sección 3, exponemos nuestros objetivos en esta línea de investigación; y en la última Sección, nuestro plan de trabajo.

## 2. Métodos de acceso Espacio-Temporal

A continuación abordamos el problema de indexación de objetos espacio-temporales y se discuten los principales métodos de acceso hasta ahora conocidos.

En primer lugar presentamos los tipos de consultas espacio-temporales más comunes. Estas consultas incorporan predicados espaciales que involucran instantes o intervalos de tiempo. Los tipos principales de consulta son [1, 5]:

*TimeSlice*: El resultado de la consulta consiste de los objetos que se encuentran en una determinada área en un instante de tiempo dado.

*Intervalo*: A diferencia de la consulta *TimeSlice* se tiene en cuenta un intervalo y no un instante de tiempo.

*Trayectoria*: Se recupera el conjunto de posiciones espaciales en las que un objeto ha permanecido en un intervalo de tiempo dado.

*Join*: Selecciona todos los pares de objetos que se encuentran espacialmente cercanos, durante un intervalo o instante de tiempo específico.

*Eventos*: Se recuperan eventos que han sucedido en una región en un instante dado. Estos eventos pueden ser objetos que han *aparecido* o *desaparecido* en una región en un cierto instante.

Se han propuesto varios métodos para construir índices para objetos espacio-temporales. Estos métodos se pueden categorizar de la siguiente manera [6]:

### 2.1 Métodos que tratan el tiempo como otra dimensión

El método de acceso 3D R-Tree, propuesto en [7], considera el tiempo como otro eje junto a las coordenadas espaciales. Un objeto el cual permanece inicialmente en  $(x_0, y_0)$  durante el tiempo  $[t_i, t_j]$  y luego en  $(x_1, y_1)$  durante el tiempo  $[t_j, t_k]$  puede ser modelado por dos segmentos de líneas en el espacio tridimensional identificadas por:  $[(x_0, y_0, t_i); (x_0, y_0, t_j)]$  y  $[(x_0, y_0, t_j); (x_0, y_0, t_k)]$ , las cuales

pueden ser indexadas por un R-Tree [3], considerando los objetos en tres dimensiones. Todos los límites finales de los intervalos de tiempo deben ser conocidos por anticipado para su correcto funcionamiento. Este método presenta como desventajas la necesidad de conocer por anticipado los intervalos de tiempo y la ineficiencia para procesar consultas del tipo timeslice. Sin embargo, entre sus ventajas se puede destacar el uso eficiente del espacio y el procesamiento de consultas del tipo intervalo.

Otro método que corresponde a esta categoría es el 2+3 R-Tree [4] que permite indexar información actual y pasada de objetos que se encuentran en movimiento. La idea principal es tener dos R-Tree de manera separada, uno de dos dimensiones para almacenar los objetos que tienen definido tiempo final, y un segundo R-Tree tridimensional (dos dimensiones espaciales y una dimensión temporal) para almacenar aquellos objetos que tienen definidos el tiempo inicial y final de permanencia en una determinada posición. Esta estructura permite mantener información histórica de los objetos como así también su trayectoria. Dependiendo del tiempo de la consulta, puede ser necesario hacer la búsqueda en ambos árboles.

## **2.2 Métodos que incorporan la información temporal dentro de los nodos de la estructura, sin considerar el tiempo como otra dimensión**

En el método de acceso RT-Tree [9], la información temporal se mantiene en los nodos del R-tree, es decir que extiende el contenido que normalmente tienen dichos nodos. Como las búsquedas son guiadas por la información espacial, las consultas con predicados en el dominio temporal no pueden ser eficientemente procesadas. Una gran desventaja de esta estructura es su ineficiencia para procesar consultas del tipo timeslice debido a la gran altura que puede alcanzar y al hecho de que la dimensión temporal juega un rol secundario. Sin embargo, se ha demostrado que el RT-Tree es una de las estructuras más eficientes en la utilización del almacenamiento.

Otro método que pertenece a esta categoría es el DR-Tree, el cual es un método de acceso espacio-temporal de tipo histórico que se basa en la idea expuesta en [1]. La idea básica del DR-Tree es mantener puntos de referencia con el componente espacial de los objetos para ciertos instantes de tiempo, los objetos “vivos” en esos instantes de tiempo se almacenan en una estructura de datos R-Tree. Los movimientos de los objetos entre puntos de referencia consecutivos se mantienen en una lista de movimientos denominada bitácora, ordenada de acuerdo al tiempo, y que permite reconstruir cualquier estado de la base de datos entre dos puntos de referencia consecutivos.

## **2.3 Métodos que usan múltiples versiones de la misma estructura para distintos instantes de tiempo, con reutilización de las partes invariantes**

En el HR-Tree [4,3], MR-Tree [9] y OLQ (*Overlapping Linear Quadtree*), la idea básica es que dado dos árboles, el más reciente corresponde a una evolución del más antiguo.

El HR-Tree es uno de los métodos más estudiados y es uno de los más eficientes para procesar consultas del tipo timeslice, aunque requiere espacio excesivo para almacenar la estructura. Por ejemplo, si sólo un objeto de cada nodo hoja se desplaza en un instante de tiempo, esto implica duplicar completamente el árbol en el instante siguiente.

Para eliminar las desventajas del HR-Tree se propone una variante denominada HR+-Tree [5]. Aquí, se trata de eliminar la redundancia de versión producida en el HR-Tree permitiendo un ahorro significativo de almacenamiento. Se ha demostrado [5] que el HR+-Tree sólo necesita alrededor de un 20% del espacio requerido por el HR-Tree sin degradar demasiado el procesamiento de consultas de tipo timeslice. Además mejora el procesamiento de consultas de tipo intervalo. Sin embargo, esta variante no resuelve el problema del almacenamiento completamente.

### 3. Objetivos

Nuestro trabajo de investigación se dedica al estudio y análisis del diseño de estructuras de datos y algoritmos que permitan resolver los distintos tipos de consulta antes mencionados en diversos métodos de acceso espacio-temporal. En particular, se han abordado dos métodos: DR-Tree y 2+3 R-Tree. Como ambos métodos apuntan a la resolución de los tipos de consultas *timeslice* e *intervalo*, hemos propuesto extensiones a las estructuras de datos subyacentes de forma tal que permitan resolver los tipos de consultas *evento* y *trayectoria*. A continuación damos un resumen de cada una de las estructuras tratadas.

#### 3.1 D\*R-Tree

En particular, el DR-Tree es un método de acceso espacio-temporal de tipo histórico que se basa en la idea expuesta en [1]. El método propuesto pretende mantener un equilibrio entre el espacio de disco utilizado por la estructura y el tiempo de acceso empleado en responder los distintos tipos de consulta. Nosotros proponemos extender este método, denominándolo D\*R-Tree, con las siguientes características:

- Se agrega un Índice para almacenar los instantes de tiempo que se establezcan como puntos de referencia.
- Los R-Tree's son utilizados para almacenar la ubicación espacial de los objetos. Cada R-Tree está asociado a un punto de referencia de tiempo, según el índice descrito en el punto anterior.
- Se usan Bitácoras para almacenar los movimientos realizados en instantes intermedios entre puntos de referencia almacenados en el índice de instantes.
- Se agrega un Índice para acceder a los últimos registros de movimiento en cada bitácora por cada objeto y con ello se mantiene el registro de la trayectoria de cada uno de los objetos en movimiento.

#### 3.2 I+3 R-Tree

El 2+3 R-Tree es una estructura que permite mantener datos de tipo espacio-temporal, el cual consta de dos R-Tree, uno bidimensional y otro tridimensional. La estructura propuesta, el I+3 R-Tree es una alternativa al 2+3 R-Tree en la cual se reemplaza el 2D R-Tree [3] de la estructura original, por un Índice.

El I+3 R-Tree consta de las siguientes características:

- Índice: en el cual se almacenan los cubos abiertos. Esto es aquello para los cuales su instante final en una posición aún no está definida. También se guardan las referencias necesarias a los cubos anteriores que describen la trayectoria del objeto.
- 3D R-Tree: Esta estructura se utiliza para almacenar los cubos cerrados, que representan las posiciones anteriores de los objetos y su tiempo de estadía en dicha posición. Cuando un objeto cambia su posición actual, se actualiza su posición en el índice, y se inserta en el 3D R-Tree un cubo que representa su posición anterior.

### 4. Trabajo futuro

Aunque se ha reconocido la necesidad de que los DBMS soporten tipos de datos espacio-temporales, estos no los implementan y en general, se desarrollan sistemas a medida utilizando plataformas generales de tipo espacio-temporales. Así surge la necesidad de un ambiente de trabajo

general flexible para soportar las clases de consultas espacio-temporales. Un enfoque es usar las estructuras de índices existentes y extenderlas a la problemática espacio-temporal. Y otro enfoque consiste en desarrollar herramientas de indexación específicas espacios-temporales. En este trabajo nos abocaremos al segundo enfoque, es decir al estudio de métodos de indexación espacio-temporal con el objetivo de diseñar estructuras de datos y algoritmos que permitan resolver eficientemente los tipos de consultas previamente descritos. Esta es un área de investigación abierta sobre la que existe poco trabajo realizado y es de utilidad dado que pocos sistemas tienen totalmente integrado el tipo dato espacio-temporal y más aún la incorporación de plataformas espacio-temporales para la resolución de consultas que involucran operaciones con predicados espacio-temporales.

Estos trabajos están enmarcados dentro de la línea de investigación Geometría Computacional y Bases de Datos, perteneciente al Proyecto Tecnologías Avanzadas de Bases de Datos 22/F314, Departamento de Informática, UNSL y en el marco de la Red Iberoamericana de Tecnologías del Software (RITOS2), subvencionado por CYTED, por lo que se ha establecido un grupo de interés en el tema conformado por docentes investigadores y alumnos avanzados de la UNSL y de la UBB.

## Referencias bibliográficas

- [1]. Gutiérrez Gilberto. Propuesta de un método de acceso espacio-temporal. II WorkShop de Bases de datos. (2003), Chillan/Chile (2003).
- [2]. Güting, R.H., An introduction to Spatial Database System. VLDB Journal (1994)
- [3]. Nascimento M., Silva J., and Theodoridis Y.. Access structures for moving points, (1998)
- [4]. Nascimento M., Silva J., and Theodoridis Y.. Evaluation of access structures for discretely moving points. In Spatio-Temporal Database Management, pages 171-188. (1999)
- [5]. Tao Y. and Papadias D.. Efficient historical R-tree. In IEEE International Conference on Scientific and Statistical Database Management. (2001)
- [6]. Theodoridis Y., Sellis Timos K., Papadopoulos A., and Manolopoulos Y.. Specifications for efficient indexing in spatiotemporal databases. In Proceedings 10th International Conference on Scientific and Statistical Database Management, Capri, Italy, pages 123-132. (1998)
- [7]. Theodoridis Y., Vazirgiannis M., and Sellis Timos K.. Spatio-temporal indexing for large multimedia applications. In International Conference on Multimedia Computing and Systems, pages 441-448. (1996)
- [8]. Wang, X., Zhou, X. y Lu, S. Spatiotemporal data modeling and management: a survey. Proceedings 36<sup>th</sup> International Conference on Technology of Object-Oriented Languages and System, pp. 202-211. (2000)
- [9]. Xu X., Han J., and Lu W. RT-tree: An improved R-tree index structure for spatio-temporal database. In 4th International Symposium on Spatial Data Handling, pages 1040-1049. (1990)