

Implementación del Índice Espacio-Temporal I+3 R-Tree en la Gestión de Emergencias

Resumen: En este artículo se describe el desarrollo de un prototipo para la atención de eventos de emergencia y la administración de los móviles correspondientes del Municipio de la localidad de Córdoba, Argentina, desarrollado en el marco de una tesis de Maestría en Ingeniería en Sistemas de Información, de la Facultad Regional Córdoba, de la Universidad Tecnológica Nacional.

Debido a la necesidad de gestionar móviles, que requieren el registro de sus posiciones en el tiempo y espacio, se utilizó la implementación del índice espacio-temporal I+3 R-Tree. Con esto se resolvió tanto la gestión de objetos móviles como la resolución de consultas relacionadas a la trayectoria que realizan los móviles en la atención de un evento.

Palabras Claves: Espacio-temporal, bases de datos espacio temporal, índices I+3 R-Tree, aplicación.

Abstract: This article describes the development of a prototype for attending emergency events and the administration of the corresponding mobiles of the Municipality of Córdoba, Argentine, developed within the framework of a Master's thesis in Engineering in Information Systems, Facultad Regional Córdoba, National Technology University.

Due to the need to manage objects that modify their position in time in this application, it was decided to implement the I+3 R-Tree space-time index. This solved both the management of mobile objects and the resolution of queries related to the path that the mobile perform in the care of an event.

Keywords: Temporal-space, databases, index, I+3 R-Tree, application.

Marcela M. Aballay¹, Anabella C. De Battista², Edilma O. Galiardi³

¹Maestría en Ing. en Sist. de Información, UTN, FRC, Ciudad Universitaria, Maestro M. Lopez esq. Cruz Roja s/n, 5000 Córdoba, Argentina.

²Dpto. Ing. en Sist. de Información, UTN, FRCU, Ing. Pereira 676, Concepción del Uruguay, Entre Ríos, Argentina.

³Dpto. de Informática, UNSL, Ejercito de Los Andes 950, San Luis, Argentina.

Mail: ¹marcelaaballay@yahoo.com.ar - ²debattistaa@frcu.utn.edu.ar - ³oli@unsl.edu.ar

INTRODUCCIÓN

En la actualidad la mayoría de los Sistemas Gestores de Bases de Datos trabajan con objetos que se pueden representar a través del modelo relacional compuesto básicamente de tablas, tuplas, campos y las relaciones establecidas entre ellos.

Si bien el modelo relacional es el más utilizado, para algunas aplicaciones en las que se requiere poder representar la forma y/o posición de objetos en el espacio y su evolución en relación al tiempo no resulta apropiado. Para estos tipos de datos se desarrollaron nuevos modelos como las bases de datos espaciales y espacio-temporales.

En algunas ocasiones también se requiere mantener distintos estados de la base de datos a través del tiempo.

Las bases de datos temporales permiten administrar los datos resguardando su estado pasado, actual e incluso futuro.

Si se pretende gestionar información que puede modificar su posición y/o forma con el correr del tiempo es imprescindible contar con índices específicos, en este caso se habla de Métodos de Acceso Espacio-Temporales. Estos índices evitan examinar todos los objetos de la base de datos al momento de responder consultas y resultan eficientes ante consultas que se realizan con alta frecuencia. La indexación se utiliza para optimizar la recuperación de datos almacenados en memoria secundaria y el tiempo de procesamiento de las consultas.

En este artículo se presenta un desarrollo en el que se implementa el índice espacio-temporal I+3 R-Tree [Gagliardi et al., 2009] para gestionar una aplicación concebida para la atención de eventos de emergencia y la administración de los móviles correspondientes de la Municipalidad de la localidad de Córdoba, Argentina.

Con el objetivo de brindar una introducción al

marco teórico que sustenta este trabajo se definen a continuación algunos conceptos relacionados:

Bases de datos espaciales

Las Bases de Datos Espaciales permiten procesar objetos con alguna referencia espacial y que poseen normalmente una estructura compleja. Un dato espacial puede ser un punto, una poligonal o un polígono [Güting et al., 2005; Berg et al., 2008]. Con un punto podrían representarse automóviles, edificios, estaciones de trenes, etc.; las poligonales se utilizan para modelar ríos, rutas, cableados, etc., y los polígonos se pueden utilizar para representar países o regiones. El principal ámbito de aplicación de las bases de datos espaciales es el de los Sistemas de Información Geográfica (SIG) [Shekhar & Chawla, 2003].

La recuperación y actualización de los tipos de datos espaciales se basan no sólo en el valor de ciertos atributos, sino también en la ubicación espacial del objeto. Por ejemplo, nos podría interesar obtener los terrenos geográficamente adyacentes a uno dado, o encontrar todas las estaciones de servicio por las que pasa una ruta.

Este tipo de bases de datos proporciona la estructura capaz de soportar datos espaciales, además de proveer un lenguaje de consulta y métodos de acceso eficientes [Theodoridis et al., 1996].

Consultas espaciales

Se enumeran a continuación los distintos tipos de consultas que pueden realizarse a una Base de Datos Espacial [Gaede et al., 1998; Shekhar et al., 2003; Gutiérrez Retamal, 2007]:

- Exact match query: dado un objeto encuentra todos los iguales en forma, posición y tamaño.
- Point query: encuentra los objetos que se superponen o cubren al punto consultado.
- Window query o range query: dado un rectángulo de dimensión d , encuentra los objetos que tienen al menos un punto en común con el rectángulo de consulta.
- Intersection query, region query u overlap query: dado un objeto devuelve los objetos de la base de datos que tienen al menos un punto en común con el objeto de consulta.
- Enclosure query: devuelve los objetos de la base de datos que encierran o contienen completamente al objeto de consulta.
- Containment query: devuelve los objetos de la base de datos que están encerrados por el objeto de consulta.
- Adjacency query: encuentra los objetos adyacentes al objeto de consulta.
- Nearest-neighbor query: devuelve los objetos de la base de datos que tienen la distancia mínima al objeto de consulta. En este caso la distancia se define como la distancia entre sus puntos más cercanos (Euclidiana o de Manhattan).
- Spatial join: dadas dos colecciones de objetos espaciales R y S y un predicado espacial P , encuentra todos los pares de objetos de $R \times S$ donde $P(R, S)$ es verdadero.

Bases de Datos Temporales

Este modelo de base de datos incorpora al tiempo como una dimensión, por lo que permiten asociar tiempos a los datos almacenados [Stantic et al., 2003; De Battista, 2008]. De acuerdo a la forma en que manejan el tiempo se puede establecer

una clasificación de las bases de datos temporales [Dyreson et al., 1994]:

- de tiempo transaccional donde el tiempo se registra de acuerdo al orden en que se procesan las transacciones,
- de tiempo vigente que almacenan el tiempo en que el hecho ocurrió en la realidad, que puede no coincidir con el momento de su registro
- bitemporales que integran la dimensión transaccional y la dimensión vigente a través del versionado de los estados, es decir, cada estado se puede modificar para actualizar el conocimiento de la realidad pasada, presente o futura, pero esas modificaciones se realizan generando nuevas versiones de los mismos estados.

Es decir que el tiempo válido identifica el momento de tiempo durante el cual la información es cierta en el mundo modelado y el tiempo transaccional indica el momento que los datos fueron incorporados a la base de datos [Güting et al., 2005].

Si se realiza una consulta teniendo en cuenta el tiempo vigente esta sería: encontrar todos los móviles activos para un tiempo determinado; en cambio si tomamos como parámetro el tiempo transaccional se consultaría: encontrar todos los móviles activos registrados en un tiempo determinado.

Bases de datos espacio-temporales

Las Bases de Datos Espacio-Temporales están compuestas por objetos espaciales que cambian su posición o forma a lo largo del tiempo [Mokbel et al., 2003; Schneider & Güting, 2005]. Este modelo permite administrar ambos componentes de un objeto al mismo tiempo, el espacial y el temporal.

Estas bases de datos poseen ciertas características [Ahn et al., 2001]:

- Tienen una estructura compleja que no puede ser representada por una tupla. Un dato espacial, entre otros, puede ser un punto, una línea o un polígono.
- Tienden a ser muy grandes. Por ejemplo un mapa puede ocupar varios gigabytes de almacenamiento.
- No existe un álgebra espacial o espacio temporal estándar, por lo tanto no hay un conjunto de operadores estandarizados.
- Se emplean operadores espaciales altamente dependientes de la aplicación en particular.
- Debido a la complejidad de los datos las operaciones espaciales implican un costo de procesamiento más alto que las operaciones relacionales.
- Por su dinamismo, requieren que la estructura de datos soporte frecuentes inserciones, eliminaciones y actualizaciones.
- Los operadores espaciales no cumplen con la propiedad de clausura. Por ejemplo, la intersección de polígonos no necesariamente da como resultado otro polígono.

Las principales consultas que se pueden realizar sobre una base espacio-temporal son [Mokbel et al., 2003]:

- TimeSlice: devuelve los objetos que se encuentran en una determinada área en un instante de tiempo dado.
- Intervalo: obtiene los objetos que se encuentran en una determinada área en un intervalo de tiempo.
- Eventos: se obtienen todos los eventos que sucedieron en una región en un instante dado.
- Trayectoria: recupera las posiciones espaciales en las que un objeto ha permanecido en un intervalo de tiempo dado.

- Consultas de los k vecinos más cercanos: dado un objeto, se devuelven los k objetos más cercanos al objeto de entrada para un instante o intervalo de tiempo.

Índices Espacio-Temporales

En el desarrollo de aplicaciones es fundamental el procesamiento eficiente de las consultas, por lo que se debe contar también con métodos adecuados para la recuperación de datos que eviten examinar todos los objetos de la base de datos en el momento de la consulta. Como respuesta a esto se implementan índices espacio-temporales que permiten seleccionar eficientemente solo aquellos objetos que son candidatos a formar parte de la respuesta.

Existe un método base de indexación espacial que implementan la mayoría de los Sistemas Gestores de Bases de Datos Espacio-Temporales (como Oracle, SQL y Postgres) que es el R-Tree, propuesto por Guttman en 1984 [Guttman et al., 1984].

En el ámbito de bases de datos espacio-temporales existen variantes del R-Tree, algunas de ellas consideran el tiempo como otra dimensión [Manolopoulos et al., 2006] como 3D R-Tree [Theodoridis et al., 1996], I+3 R-Tree [Gagliardi et al., 2009] y 2+3 R-Tree [Nascimento et al., 1998; Nascimento et al., 1999]. Existen otras variantes en las que el tiempo se incorpora como información dentro de la estructura de los nodos, como el RT-Tree [Xu et al., 1990].

En este desarrollo se propuso la utilización del índice I+3 R-Tree que permite manipular los puntos en movimiento, administra el tiempo como otra dimensión y responde a las consultas timeslice, intervalo, evento y trayectoria. Es una variante del 2+3 R-Tree, que se compone de dos R-Tree, uno de dos dimensiones para la información actual. Los objetos que no

tienen definido su tiempo final de estadía en una posición determinada. Otro de tres dimensiones, dos para el tiempo y una espacial, para la información pasada, que son los objetos que tienen tiempo inicial y final en una posición.

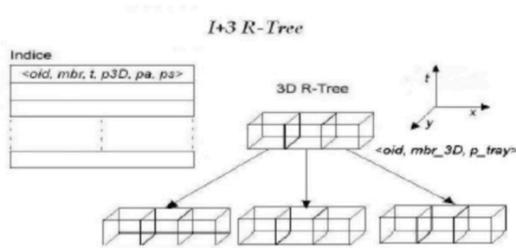


Fig. 1: Estructura del I+3 R-Tree.

La estructura del I+3 R-Tree (Figura 1) está formada por un R-Tree de tres dimensiones [Theodoridis et al., 1996] más una estructura denominada I que guarda los datos de las posiciones actuales de los objetos.

DESARROLLO

En este trabajo se presenta el desarrollo de un prototipo de aplicación para la atención de eventos de emergencia y la administración de los móviles correspondientes del Municipio de la localidad de Córdoba.

El procedimiento al que se da soporte con esta aplicación consiste básicamente en recibir una llamada a través de los operadores telefónicos, quienes la registran y derivan a un despacho (Figura 2).

Luego de registrada la llamada, ésta se puede derivar al despacho si requiere la intervención de algún móvil o finalizarla. En el despacho se cuenta con una vista de los eventos pendientes y en curso con su correspondiente localización en el mapa (Figura 3).

Para la asignación de un móvil a un evento pendiente se obtiene de la tabla de Móviles de la base de datos rela-

cional aquellos que están en estado “disponible” y se los representa gráficamente sobre un mapa, formando con ellos un diagrama de Voronoi [Berg, Cheong, Kreveld & Overmars, 2008] (Figura 4). De esta forma los despachantes pueden ubicar rápidamente cual es el móvil más cercano al evento que están atendiendo.

Una vez que se asigna un móvil a un evento pendiente, dicho móvil se incorpora a la vista del despacho y se actualizan los eventos pendientes y en curso (Figura 5).

Una vez finalizada la atención se cierra el evento y se libera el móvil, pasando a estar disponible para cubrir un nuevo evento.

De los móviles se registra periódicamente su posición, lo que permite posteriormente reconstruir su trayectoria en la atención de un determinado evento.

Para este desarrollo se tomaron como base los requerimientos del call center de emergencias de la Municipalidad de Córdoba. Dichos requerimientos fueron documentados con sus Actores y Casos de Uso (Figura 6).

El prototipo está programado en Java, utiliza como base de datos relacional PostgreSQL e implementa el método de acceso espacio-temporal I+3 R-Tree mediante librerías en Java.

Los principales objetos que se modelan a través de la base relacional son Llamadas, Eventos y Móviles (Figura 7).

Como se mencionó anteriormente, el índice I+3 R-Tree está conformado por dos estructuras 3D R-Tree y un Índice.

Las tuplas en el 3D R-Tree están representadas por los atributos <oid, mbr_3D, p_tray>

- oid: código identificador del objeto
- mbr_3D: región tridimensional cuya altura repre-

senta el intervalo temporal durante el cual el objeto se mantuvo en la posición espacial definida por su base :

- p_tray: puntero al cubo anterior correspondiente al mismo oid, utilizado para mantener un historial de trayectoria.

Las tuplas en el índice están representadas por los atributos <oid, mbr, t, p3D, pa, ps>

- mbr: región aproximada que ocupa actualmente el objeto
- t: tiempo de llegada del objeto a su ubicación actual
- p3D: puntero al cubo anterior correspondiente al mismo oid, utilizado para mantener un historial de trayectoria;
- pa: puntero al objeto insertado en el instante de tiempo inmediatamente anterior
- ps: puntero al objeto insertado en el instante de tiempo siguiente.

Esquemáticamente el I+3 R-Tree tiene el siguiente funcionamiento: inicialmente sólo se registran en el Índice los móviles con su posición inicial (el 3D R-Tree se encuentra vacío) (Tabla 1).

Oid	Mbr	t	p3D	Pa	Ps
om1	om1_posicion0	om1_tiempo0	null	null	puntero a om2
om2	om2_posicion0	om2_tiempo0	null	puntero a om1	Puntero a om3
om3

Tabla 1: Índice.

Al recibir información de una nueva posición del móvil om1 se actualiza el Índice y el 3D R-Tree (Tablas 2 y 3).

Oid	Mbr	t	p3D	Pa	Ps
om1	om1_posicion1	om1_tiempo1	puntero al primer cubo del R-Tree	null	puntero a om2
om2	om2_posicion0 (la de la base)	om2_tiempo0	null	puntero a om1	puntero a om3
om3

Tabla 2: Índice.

Oid	mbr_3D	p_tray
om1	posicion0 - Desde tiempo0 Hasta tiempo1	null

Tabla 3: 3D R-Tree

Y así sucesivamente con cada recepción de información (Tablas 4 y 5).

Para la visualización y georreferenciación se utilizaron las APIs de Google Maps ["Google Maps API | Google Developers", 2017]. De esta manera se puede localizar en el mapa de la ciudad los puntos donde se encuentran los eventos en curso y pendientes de atención.

En el caso del despacho, en la pantalla correspondiente a la asignación de un móvil para un evento, se implementó un diagrama de Voronoi. Esta estructura geométrica subdivide un área en regiones tomando como base una serie de pivotes representados por puntos geográficos. Cada una de estas regiones queda conformada por los puntos más cercanos al punto de referencia.

Oid	mbr	T	p3D	Pa	ps
om1	om1_posicion2	om1_tiempo2	puntero al segundo cubo del R-Tree	null	puntero a om2
om2	om2_posicion0 (la de la base)	om2_tiempo0	null	puntero a om1	puntero a om3
om3

Tabla 4: Índice.

Oid	mbr_3D	p_tray
om1	posicion0 - Desde tiempo0 Hasta tiempo1	null
om1	Posicion1 - Desde tiempo1 Hasta tiempo2	puntero al cubo anterior

Tabla 5: 3D R-Tree

En este caso el diagrama de Voronoi se genera dividiendo la ciudad de Córdoba en regiones, tomando como puntos de referencia las bases donde se encuentran los móviles disponibles, por ejemplo, hospitales.

De esta manera quedan conformadas regiones asociadas a un móvil, lo que permite ante un evento visualizar rápidamente, según la región donde se localiza, cuál es el móvil disponible más próximo para su atención.

Como en el prototipo no se dispone de una conexión real con los móviles para que transmitan cada determinado período su posición, se realizó una simulación, registrando en el I+3 R-Tree distintos tiempos y posiciones (latitud, longitud) para algunos móviles.

Luego de finalizada la atención de un evento, en base a los datos registrados en la estructura espacio-temporal se puede resolver la consulta de trayectoria de un móvil, que es uno de los principales requerimientos de información para este sistema.

RESULTADOS

El desarrollo de este prototipo ha resultado un desafío muy interesante tanto por los retos asumidos desde lo técnico como por la relevancia que adquiere el mismo al responder a las necesidades de gestión de móviles ante incidentes en la ciudad de Córdoba. La combinación del motor de Bases de Datos Relacional PostgreSQL con el índice métrico i+3 R-tree para tratar objetos en movimiento, ha resultado muy satisfactoria

y los resultados de las primeras pruebas son alentadores, dado que el prototipo ha respondido de manera adecuada a las consultas efectuadas, devolviendo correctamente los móviles disponibles en los radios más cercanos al lugar del evento.

DISCUSIÓN

Tratándose de un desarrollo con objetos en movimiento donde se deben considerar la componente espacial y temporal no se encontró en el mercado un sistema de gestión de base de datos que proporcione las facilidades de almacenamiento y consulta para tipos espacio-temporales.

Teniendo en cuenta las características de los móviles de este sistema se consideró la estructura del I+3 R-Tree como la más conveniente de implementar, teniendo en cuenta las siguientes consideraciones:

- Los objetos se mueven a gran velocidad.
- El tamaño y la forma de los objetos no son importantes, el interés se centra en su posición en el tiempo.
- El espacio no cambia y la cantidad de objetos no tiene mucha variación.
- Los objetos se mueven en áreas previamente establecidas.
- Se necesita consultar la trayectoria realizada por los objetos.

Otro aspecto importante que se resolvió a través de la utilización de este índice es el de las consultas. En este caso se necesita reproducir la trayectoria que realiza un móvil en la atención de un evento, consulta que no está disponible en todos los índices de acceso espacio-temporal.

Además, en el I+3 R-Tree, la lista secuencial de los objetos se encuentra ordenada por su identificador lo que

hace más eficiente la búsqueda de un determinado móvil. Uno de los aspectos más relevante de este trabajo

es que se pudo concretar la integración de ambos modelos, el relacional y el espacio-temporal.

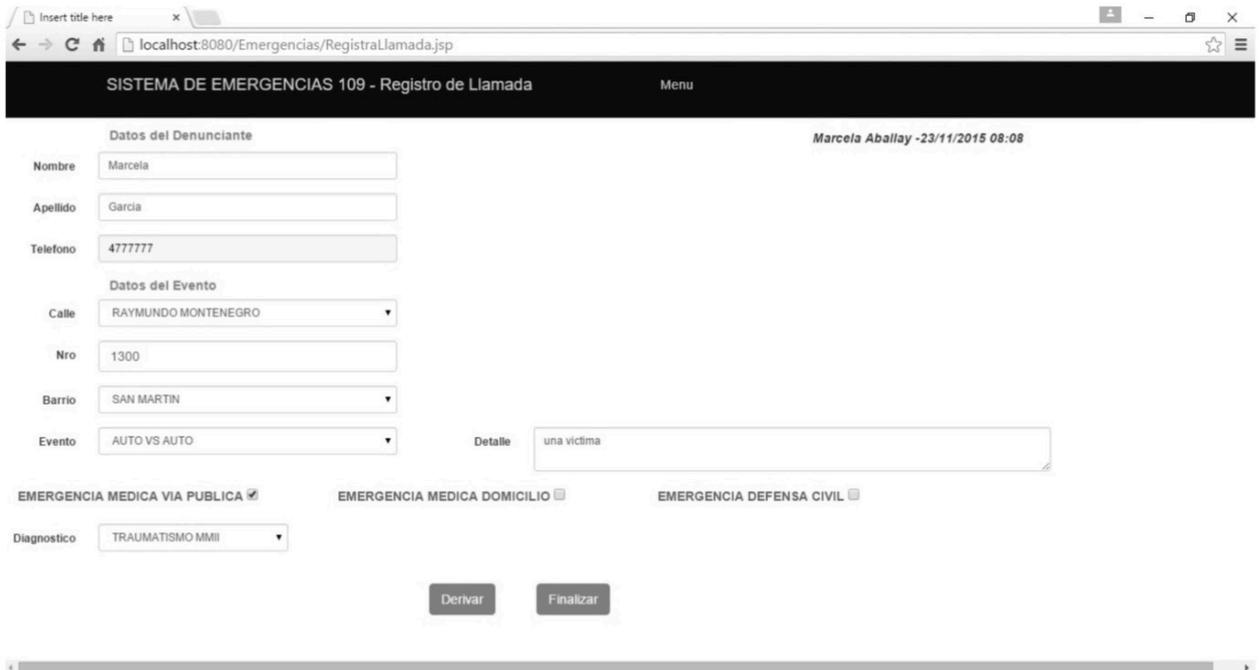


Figura 2: Pantalla para el registro de llamada.

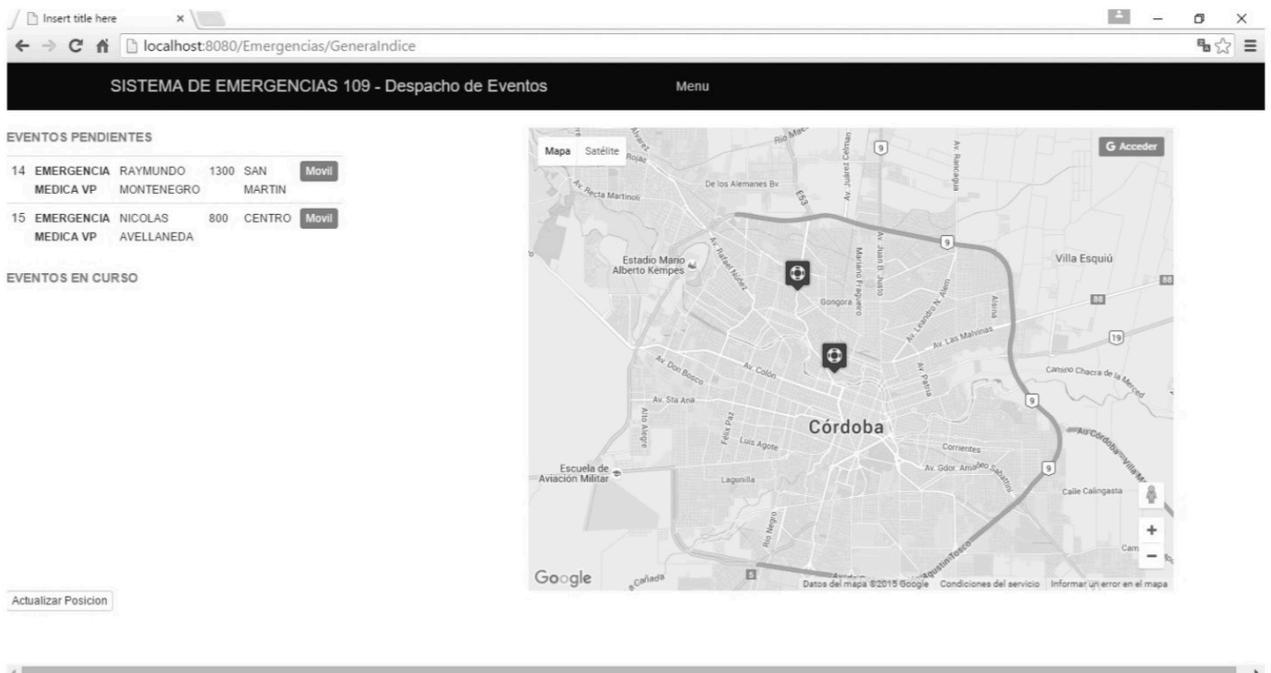


Figura 3: Vista del despacho con los eventos pendientes.

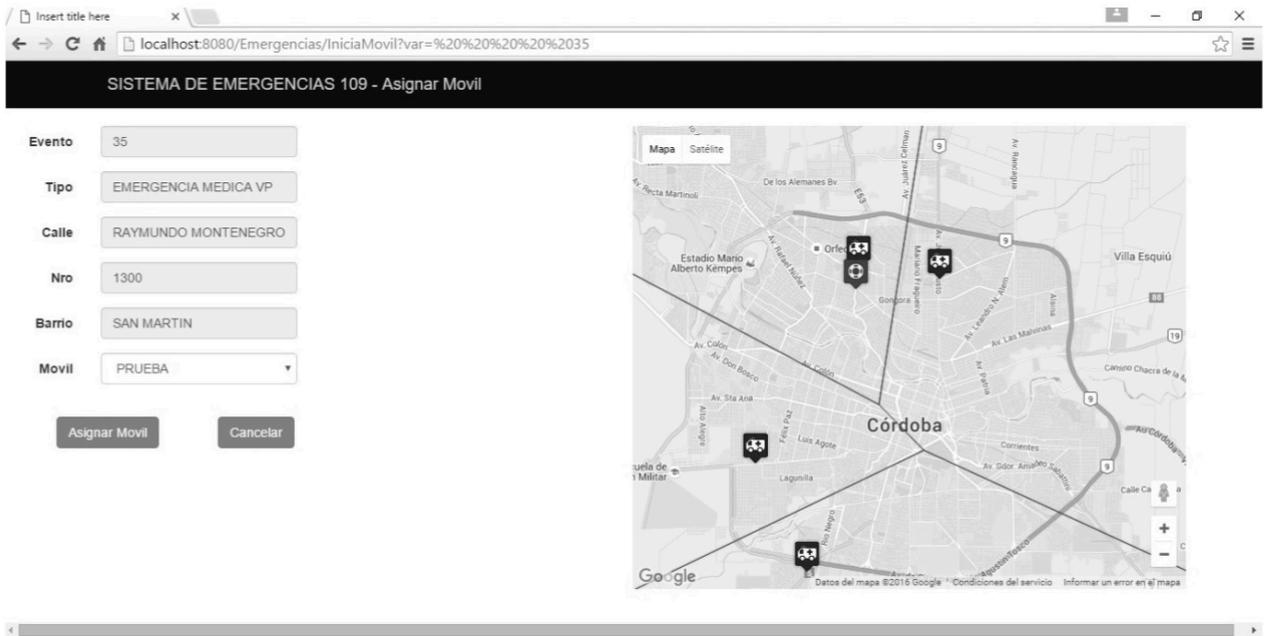


Figura 4: Vista del despacho con los móviles disponibles.

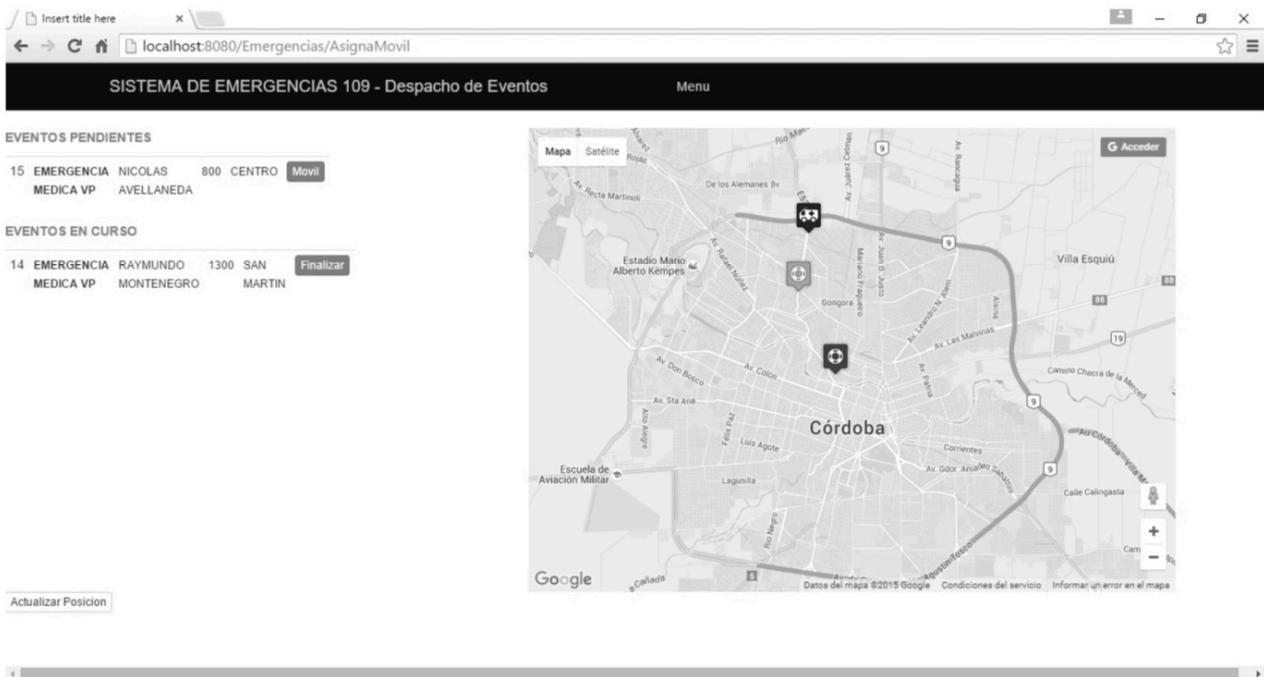


Figura 5: Vista del despacho con los eventos pendientes, en curso y móviles asignados.

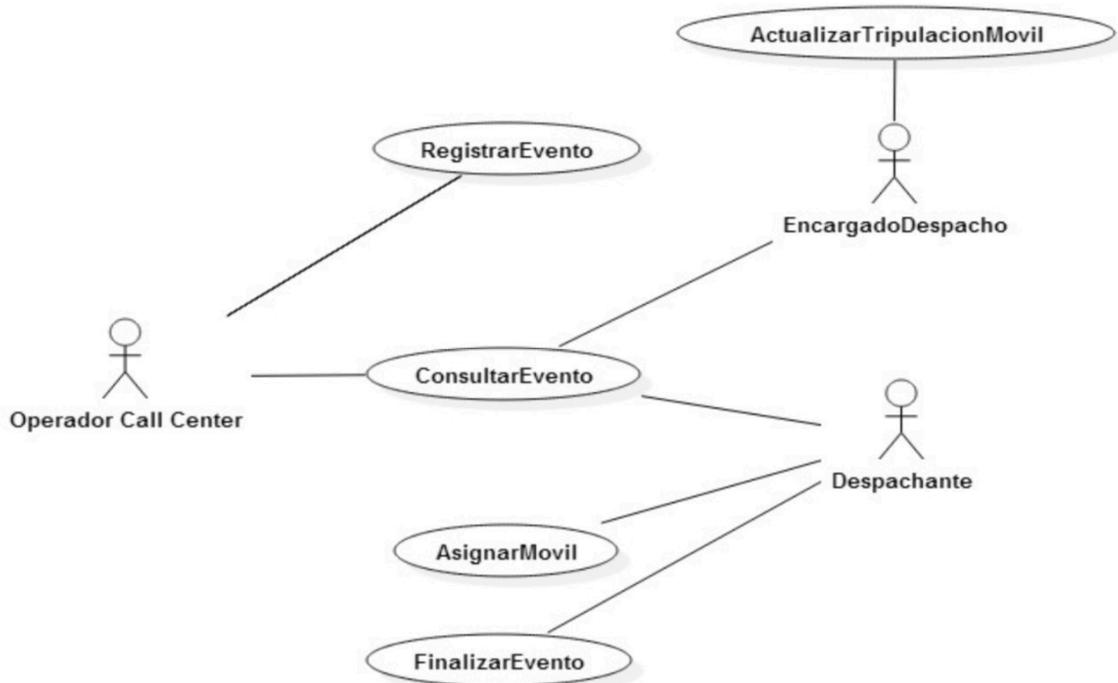


Figura 6: Diagramas de casos de uso.

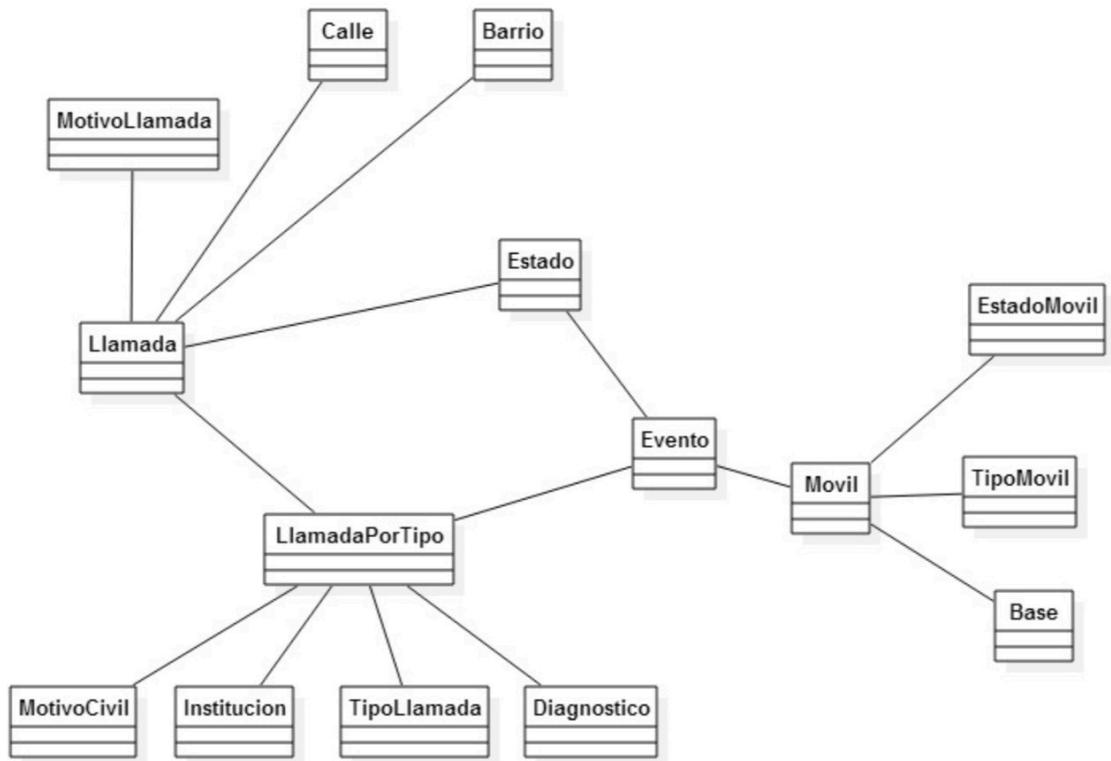


Figura 7: Esquema de base de datos relacional.

REFERENCIAS

- Ahn H, Mamoulis N, Wong H. (2001). *A Survey on Multi-dimensional Access Methods*. Institute of Information and Computing Sciences, Utrecht University, The Netherlands.
- Berg, M., Cheong, O., Kreveld, M., & Overmars, M. (2008). *Computational Geometry*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg.
- De Battista, A. (2008). *Búsquedas por similitud sobre objetos dinámicos (Magíster en Ciencias de la Computación con orientación en Bases de Datos)*. Universidad Tecnológica Nacional, Fac. Reg. Concepción del Uruguay.
- Dyreson C, Grandi F, Käfer W, Kline N, Lorentzos N, Mitsopoulos Y, Montanari A, Nonen D, Peressi E, Pernici B, Roddick J, Sarda N, Scalas M, Segev A, Snodgrass R, Soo M, Tansel A, Tiberio P, and Wiederhold G. (1994) *A consensus glossary of temporal database concepts*. SIGMOD Rec. 23, 1, 52-64.
- Gaede V, Günther O. (1998). *Multidimensional access methods*. ACM Comput. Surv. 30, 2. (pp 170-231).
- Gagliardi E, Carrasco F, García Sosa J, Gutierrez G. (2009) *I+3 R-Tree: un método de acceso espacio-temporal*. Actas del XV Congreso Argentino de Ciencias de la Computación. (pp 1031-1040).
- Google Maps API | Google Developers. (2017). Google Developers. Recuperado el 26 de Julio de 2017, desde <http://bit.ly/2dDZzLH>
- Gutiérrez Retamal, G. (2007). *Métodos de Acceso y Procesamiento de Consultas Espacio-Temporales (Doctor en Ciencias mención Ciencias de la Computación)*. Universidad de Chile.
- Güting R. (1994). *An introduction to spatial database systems*. The VLDB Journal 3, 4 (pp 357-399).
- Guttman A. (1984). *R-Trees: A dynamic index structure for spatial searching*. In *ACM SIGMOD Conference on Management of Data*. (pp 47-57). Boston. ACM.
- Manolopoulos, Y. (2011). *R-trees*. London: Springer.
- Mokbel M, Ghanem T, Walid G. (2003) *Spatio-temporal Access Methods*. IEEE Data Eng. Bull. 26 (2), (pp 40-49).
- Nascimento M, Silva J, and Theodoridis Y. (1998) *Access Structures for moving points*. Technical Report TR-33, TIME CENTER.
- Nascimento M, Silva J, and Theodoridis Y. (1999) *Evaluation of access structures for discretely moving points*. In *Proceedings of the International Workshop on Spatio-Temporal Database Management (STDBM '99)*, (pp 171-188), London, UK. Springer-Verlag.
- Shekhar, S., & Chawla, S. (2003). *Spatial database: a tour*. Upper Saddle River: Prentice Hall.
- Schneider, M., & Güting, R. (2005). *Moving Objects Databases (Morgan Kaufmann series in data management systems)*. Morgan Kaufmann Publishers.
- Stantic B, Thornton J, Sattar A. (2003). *Approach to Model NOW in Temporal Databases*. School of Information Technology Griffith University Gold Coast, Australia A Novel.
- Theodoridis Y, Vazirgiannis M, and Sellis T. (1996) *Spatio-temporal indexing for large multimedia applications*. In *Proceedings of the 1996 International Conference on Multimedia Computing and Systems (ICMCS '96)*. (pp 441-448), Washington, DC, USA. IEEE.
- Xu X, Han J, and Lu W. (1990). *RT-tree: An improved R-tree index structure for spatio temporal database*. In *4th International Symposium on Spatial Data Handling*, pages 1040-1049, 1990.