

# Bases de Datos Espacio-Temporales

Nicolás Del Piano

Bases de Datos Avanzadas

- ▶ Bases de Datos Espacio-Temporales
  1. Definición
  2. Importancia
  3. Aplicaciones
  4. Requerimientos
  5. Modelo Espacio-Temporal
- ▶ Lenguajes de Consulta
  1. Lenguaje de Güting
  2. Lenguaje SQL<sup>ST</sup>
- ▶ Estado del Arte

El objetivo es extender los modelos de información espacial para incluir tiempo y describir de forma más dinámica la realidad que se quiere representar.

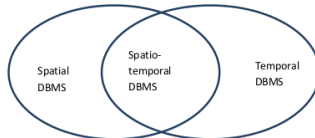
# Bases de Datos Espacio-Temporales

Se han convertido en un tópico muy importante en estos últimos años, ya que muchas de las aplicaciones como los Sistemas de Información Geográfica y los sistemas de localización necesitan almacenar datos espaciales con características temporales.

# Bases de Datos Espacio-Temporales

## STDB

*Un STDBMS es un DBMS cuyo aporte esencial es el manejo de datos espaciales a través del tiempo.*



**Figura:** Una idea de la relación entre la información espacial, temporal y espacio-temporal.

Muchas de las aplicaciones de bases de datos tratan el fenómeno espacio-temporal, y durante esta última década muchos grupos de investigación han investigado sobre la predicción del clima, prevención del atascamiento del tráfico, servicios de localización, movimiento de objetos, etcétera.

Proveer un modelo DBMS para dar soporte al fenómeno espacio-temporal se convirtió en un aspecto importante.

Sólo muy pocos STDBMS existen para satisfacer los tipos de datos y operaciones espacio-temporales.

El modelo espacio-temporal abarca aplicaciones:

- ▶ Demográficas
- ▶ Ecológicas
- ▶ Marketing
- ▶ Fenómenos naturales
- ▶ Militares
- ▶ Urbanísticas



# Requerimientos

- ▶ Representación eficiente del espacio y el tiempo.
  - ▶ Heredado de los modelos espaciales y temporales.
- ▶ Modelos de datos
  - ▶ Representa la evolución temporal de los objetos espaciales, a través de nuevos tipos de datos, las operaciones y relaciones entre ellos.
- ▶ Lenguajes de consulta

Una base de datos espacio-temporal necesita proveer un soporte DBMS para mover objetos que cambian continuamente de forma y posición.



**Figura:** Los GPS han introducido una fuerte demanda para implementar STDBMS eficientes.

# Modelo Espacio-Temporal

Los tipos de datos espacio-temporales permiten al usuario describir un comportamiento dinámico de objetos espaciales a través del tiempo.

El objeto espacial se mueve  $\Rightarrow$  los llamamos *moving objects*.

De la misma forma que los objetos espaciales, los espacio-temporales son añadidos como atributos de tipo en un modelo de datos de un DBMS.

# Tipos de Datos Espacio-Temporales

MPOINT (moving point), MLINE (moving line) y MREGION (moving region).

Conceptualmente:

MPOINT es una función  $f : \text{time} \rightarrow \text{point}$

MLINE es una función  $f : \text{time} \rightarrow \text{line}$

MREGION es una función  $f : \text{time} \rightarrow \text{region}$

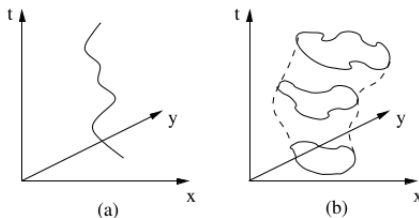


Figura: Ejemplos de un objeto *moving point* (a) y un objeto *moving region* (b).

# Operaciones y Predicados

Algunas de las operaciones que un modelo espacio-temporal debe ofrecer (además de las espaciales y temporales):

*deftime* : *mpoint*  $\rightarrow$  *periods*

*Disjoint* : *mpoint*  $\times$  *mregion*  $\rightarrow$  *bool*

*distance* : *mpoint*  $\times$  *mpoint*  $\rightarrow$  *mreal*

*Inside* : *mpoint*  $\times$  *mregion*  $\rightarrow$  *bool*

*intersection* : *mpoint*  $\times$  *mregion*  $\rightarrow$  *mpoint*

*meet* : *point*  $\times$  *region*  $\rightarrow$  *bool*

*min* : *mreal*  $\rightarrow$  *real*

*trajectory* : *mpoint*  $\rightarrow$  *line*

# LENGUAJES DE CONSULTA ESPACIO-TEMPORALES

# Lenguajes de Consulta Espacio-Temporales

Se han propuesto diversos lenguajes de consulta para este tipo de base de datos, como por ejemplo, SQL<sup>ST</sup> (Chen y Zaniolo, 2000), Güting (Güting y Schneider, 2005), el de Chomicki (Chomicki y Revesz, 1999) y STQL (Erwig y Schneider, 2002) entre otros.

En esta clase veremos y compararemos los dos primeros.

**Escenario:** incendio forestal (Güting et ál., 2000; Chen y Zaniolo, 2000).

Se maneja información sobre bosques, incendios y bomberos.



En Güting y Schneider (2005) se propone un lenguaje de consulta para el manejo de objetos móviles (se soportan cambios continuos de las geometrías, ya sea en forma y posición).

Se proponen tipos de datos móviles como MBOOL, MINTEGER, MPOINT, MLINE y MREGION, que combinan los tipos de datos primitivos y los espaciales con información temporal.

Lenguaje de tipo SQL.

## Creación de las tablas

Bosque (nombre: STRING, territorio: REGION)

Incendio (nombre: STRING, extensión: MREGION)

Bombero (nombre: STRING, ubicación: MPOINT)



<i>Bosque</i>	
Nombre	Territorio
Verde	
Santa Elena	

Figura: Tabla bosque para el lenguaje de Güting.

# Lenguajes de Consulta: Güting

<i>Bombero</i>	
Nombre	Ubicación
T. Montoya	{( 4/7/06, A ), (6/7/06 , B ), (7/7/06 , C )}
J. Vélez	{( 6/8/07, D ), (7/8/07, E ), (9/8/07, F )}

Figura: Tabla bombero para el lenguaje de Güting.







<i>Incendio</i>	
Nombre	Extensión
Gran L	{(5/7/06,  ) , (6/7/06,  ) , (7/7/06,  ) , (7/7/06,  ) }
Azul	{(24/3/07 ,  ) , (25/3/07 ,  ) }

Figura: Tabla incendio para el lenguaje de Güting.

SQL<sup>ST</sup> (Chen y Zaniolo, 2000) es una extensión minimalista del SQL estándar, ya que preserva su estructura y sólo añade tipos de datos temporales (DAY), espaciales (POINT, LINE y REGION) y operadores (ÁREA, OVERLAP, MOVING\_DISTANCE).

El tiempo se representa en forma discreta, así los cambios en las geometrías son discretos.

# Lenguajes de Consulta: SQL<sup>ST</sup>

## Creación de las tablas

```
CREATE TABLE bosque (nombre CHAR(30), territorio  
REGION)
```

```
CREATE TABLE incendio (nombre CHAR(30), extensión  
REGION, día DAY)
```

```
CREATE TABLE bombero (nombre CHAR(30), ubicación  
POINT, día DAY)
```

<i>Bosque</i>	
Nombre	Territorio
Verde	
Santa Elena	

**Figura:** Tabla bosque para el lenguaje SQL<sup>ST</sup>.

# Lenguajes de Consulta: SQL<sup>ST</sup>







Incendio		
Nombre	Extensión	Día
Gran L.	 (Área: 315)	5/7/06
Gran L.	 (Área: 503)	6/7/06
Gran L.	 (Área: 503)	7/7/06
Gran L.	 (Área: 114)	7/7/06
Azul	 (Área: 710)	24/3/07
Azul	 (Área: 480)	25/3/07

Figura: Tabla incendio para el lenguaje SQL<sup>ST</sup>.

# Lenguajes de Consulta: SQL<sup>ST</sup>

<i>Bombero</i>		
Nombre	Ubicación	Día
T. Montoya	A (31,52)	4/7/06
T. Montoya	B (40,47)	6/7/06
T. Montoya	C (34, 24)	7/7/06
J. Vélez	D (21,75)	6/8/07
J. Vélez	E (30, 66)	7/8/07
J. Vélez	F (34, 41)	9/8/07

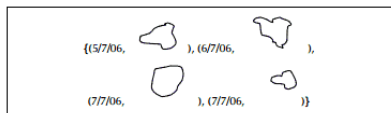
Figura: Tabla bombero para el lenguaje SQL<sup>ST</sup>.

# Ejemplos de consultas

¿Cuándo y dónde alcanzó el incendio "Gran L" su máxima extensión?

## Consulta: Güting

```
LET GranL = ELEMENT(SELECT extensión  
FROM incendio  
WHERE nombre = "Gran L");  
LET max_area = INITIAL(ATMAX(AREA(GranL)));  
ATINSTANT(GranL, INST(max_area));  
VAL(max_area);
```



ATINSTANT	VAL
6/7/06	503

**Figura:** La primera figura nos muestra el resultado de GranL. En la segunda, el resultado de la consulta.



# Ejemplos de consultas

¿Cuándo y dónde alcanzó el incendio "Gran L" su máxima extensión?

Consulta: SQL<sup>ST</sup>

```
SELECT día, extensión, AREA(extensión)
FROM incendio
WHERE nombre = "Gran L." AND AREA(extensión) =
      SELECT MAX(AREA(extensión))
      FROM incendio
      WHERE nombre = "Gran L");
```



<i>F1.día</i>	<i>F2.extensión</i>	<i>AREA</i>
6/7/06		503
7/7/06		503




Figura: Este es el resultado de la consulta.

# Ejemplos de consultas

¿Cuándo y dónde se expandieron los incendios más de 500 km<sup>2</sup>?

## Consulta: Güting

```
LET reg_grande = SELECT a_grande AS extensión
WHEN[FUN(r:region) AREA(r) > 500]
FROM incendio;
SELECT * FROM reg_grande
WHERE NOT(ISEMPTY(DEFTIME(a_grande)) );
```

reg_grande	
a_grande	
{(6/7/06,  ), (7/7/06,  )}	
{(24/3/07,  ), }	

LET square = FUN (m:integer) m \* m;

**Figura:** La primer figura nos muestra el valor de reg\_grande. En la segunda, la cláusula FUN.

# Ejemplos de consultas

¿Cuándo y dónde se expandieron los incendios más de 500 km<sup>2</sup>?

Consulta: SQL<sup>ST</sup>

```
SELECT día, extensión  
FROM incendio  
WHERE AREA(extensión) > 500;
```




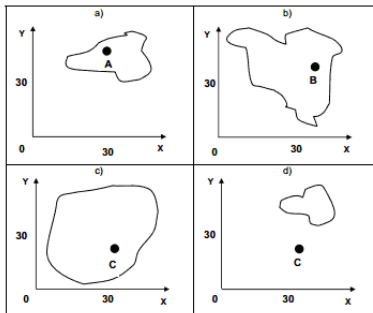
<i>F1.día</i>	<i>F2.extensión</i>
6/7/06	 (Área: 503)
7/7/06	 (Área: 503)
24/3/07	 (Área: 710)

Figura: El resultado de la consulta en SQL<sup>ST</sup>.

# Ejemplos de consulta

¿Cuánto tiempo estuvo el bombero *T. Montoya* dentro de *Gran L* y cuánta distancia recorrió dentro?



**Figura:** Ubicación del bombero *T. Montoya* dentro de *Gran L*: a) punto A y región 315 km<sup>2</sup> el 5/7/06, b) punto B y región de 503 km<sup>2</sup> el 6/7/06, c) punto C y región de 503 km<sup>2</sup> el 7/7/06 y d) punto D y región de 114 km<sup>2</sup> el 7/7/06.

# Ejemplos de consultas

¿Cuánto tiempo estuvo el bombero *T. Montoya* dentro de *Gran L* y cuánta distancia recorrió dentro?

Consulta: SQL<sup>ST</sup>

```
SELECT DURATION(bombero.día),  
       MOVING_DISTANCE(bombero.ubicación,  
                       bombero.día)  
FROM incendio, bombero  
WHERE incendio.día = bombero.día AND  
       incendio.nombre = "Gran L.^AND  
       bombero.nombre = "T. Montoya.^AND  
       INSIDE (bombero.ubicacion, incendio.extensión)
```

# Ejemplos de consulta

¿Cuánto tiempo estuvo el bombero *T. Montoya* dentro de *Gran L* y cuánta distancia recorrió dentro?

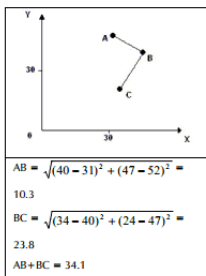


Figura: Ejemplo de la función MOVING\_DISTANCE.

# Ejemplos de consulta

¿Cuánto tiempo estuvo el bombero *T. Montoya* dentro de *Gran L* y cuánta distancia recorrió dentro?





<i>Incendio</i>			<i>Bombero</i>		
Nombre	Extensión	Día	Nombre	Ubicación	Día
Gran L		5/7/06	T. Montoya	A	5/7/06
Gran L		6/7/06	T. Montoya	B	6/7/06
Gran L		7/7/06	T. Montoya	C	7/7/06
Gran L		7/7/06	T. Montoya	C	7/7/06

Figura: Resultado parcial de la consulta en SQL<sup>ST</sup>.

<i>DURATION</i>	<i>MOVING_DISTANCE</i>
3	34.1

Figura: Resultado de la consulta en SQL<sup>ST</sup>.

# Ejemplos de consulta

¿Cuánto tiempo estuvo el bombero *T. Montoya* dentro de *Gran L* y cuánta distancia recorrió dentro?

## Consulta: Güting

```
SELECT tiempo AS  
DURATION (DEFTIME (INTERSECTION (ubicación,  
GranL))),  
distancia AS  
LENGTH(TRAJECTORY (INTERSECTION(ubicación, GranL)))  
FROM bombero  
WHERE nombre = "T. Montoya";
```

{( 5/7/06 , A ), (6/7/06 , B ), (7/7/06 , C )}

**Figura:** MPOINT resultante de INTERSECTION(ubicación, GranL).



## ESTADO DEL ARTE

# Actualidad de las STDB

Las STDB tratan con aplicaciones donde los tipos de datos son caracterizados por información espacial y temporal.

La implementación e investigación en este área ha empezado décadas atrás, cuando el manejo y manipulación de datos relacionada a los aspectos temporales y espaciales se hizo indispensable.

Sin embargo, no es una tarea sencilla dado la complejidad de las estructuras de datos junto con la representación y manipulación de esa información involucrada.

Una alternativa para lidiar con la información espacio-temporal, es construir un DBMS especializado para dar un soporte eficiente a los tipos de datos y consultas (**CONCERT** (Relly et al. 1997) y **SECONDO** (Dieker y Güting 2000)).

Cuando no sea posible usar un DBMS especializado, otra alternativa sería extender un DBMS (orientado a objetos o relacional), ya sea open source o comercial, para la manipulación y almacenamiento de objetos espacio-temporales.

Aunque los DBMS tienen capacidad de extenderse y mecanismos para soportar estas necesidades espacio-temporales, no existe un DBMS open source con una extensión espacio-temporal.

Otra alternativa es crear una arquitectura por capas sobre una DBMS ya existente.

*Oracle* brinda soporte espacio-temporal con las *versioned tables* de su Workspace Manager, pero es comercial.

Es una librería GIS open source que extiende el modelo relacional de un DBMS para el manejo del modelo de datos espacio temporal.



Figura: [www.terralib.org](http://www.terralib.org)

Soporta diferentes DBMS, incluyendo Oracle, PostgreSQL y MySQL. Se programa en C++.

Tiene su centro de desarrollo en Brasil y se distribuye bajo la licencia LGPL.

Existen varias aplicaciones asociadas para facilitar la visualización de consultas y el manejo del usuario.

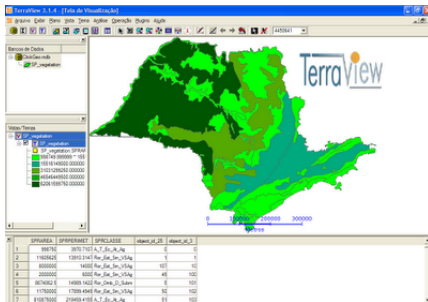


Figura: TerraView, un entorno gráfico para utilizar TerraLib.

# Problema

Existen numerosas bases de datos relacionales con extensiones espaciales y temporales, pero las bases de datos espacio-temporales no están basadas en el modelo relacional por razones prácticas.

Hasta ahora no hay RDBMS con extensiones espacio-temporales incorporadas.

TerraLib es como un medio externo que las utiliza para brindar estas herramientas.

# The Google Approach

Ya conocemos el inmenso trabajo de esta compañía en los aspectos espacio-temporales.

Entonces, ¿ cómo lidia Google este tipo de problemas?



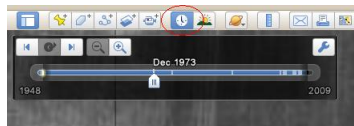
Keyhole Markup Language es una notación XML para representar datos geográficos en 3 dimensiones.

Permite describir y almacenar información geográfica así como también incorporar temporal.

KML es un estándar abierto y de su mantenimiento se encarga el Open Geospatial Consortium. Fue desarrollado para Google Earth y Google Maps.

# Google Earth

En Google Earth 5 se implementó la visualización de imágenes históricas. Es decir, podemos ver cómo se ha ido construyendo el edificio al lado de casa, o cómo era el entorno de nuestro pueblo años atrás.



**Figura:** Google Earth 5 y su herramienta histórica.

## El tiempo con vistas abstractas (fragmento).

```
<Placemark>  
  <name>Palace of Fine Arts in 2002</name>  
  <Camera>  
    <gx:TimeStamp>  
      <when>2002-07-09T19:00:00-08:00</when>  
    </gx:TimeStamp>
```

# KML: Ejemplo

Y también describir objetos espaciales:

## Descripción en el mapa de New York City

```
<?xml version="1.0">
<Document>
  <Placemark>
    <name>New York City</name>
    <description>New York City</description>
    <Point>

<coordinates>-74.0063,40.7141,0</coordinates>
    </Point>
  </Placemark>
</Document>
</kml>
```

# Otros formatos de datos espacio-temporales

- ▶ XML en sí
- ▶ GML
- ▶ NetCDF
  - ▶ Es un formato de archivo, diseñado para leer y escribir eficientemente matrices de datos.

- ▶ R. H. Güting and M. Schneider. Moving Objects Databases.
- ▶ Spatial and Spatio-Temporal Data Models and Languages, Markus Schneider.
- ▶ Claudia Deco y Cristina Bender. Tópicos Avanzados de Bases de Datos.
- ▶ Cyndy Xinmin Chen and Carlo Zaniolo. SQL<sup>ST</sup>: A Spatio-Temporal Data Model and Query Language.