



UNIVERSIDAD NACIONAL DE ROSARIO

MONOGRAFÍA
BASES DE DATOS AVANZADAS

Bases de Datos Espaciales y Temporales

Autores:

Federico Badaloni
David Giordana

Docentes:

Claudia Deco
Cristina Bender

Departamento de Ciencias de la Computación
Facultad de Ciencias Exactas, Ingeniería y Agrimensura
Av. Pellegrini 250, Rosario, Santa Fe, Argentina

2 de octubre de 2020

Resumen

El resumen de la monografía.

Índice general

Índice general	IV
1 Introducción	1
2 Bases de Datos Espaciales	3
2.1. Introducción	3
2.2. Modelos Teóricos	3
2.2.1. Realms	4
2.2.2. Actualizaciones en realms	5
2.2.3. Modelo Funcional	6
2.2.4. Modelos Basados en Objetos	7
2.2.5. SDTs	7
2.2.6. Operaciones Espaciales	8
2.3. Procesamiento de consultas espaciales	8
3 Bases de Datos Temporales	9
3.1. El dominio del tiempo	9
3.1.1. La estructura del tiempo	9
3.1.2. La densidad del tiempo	11
4 Bases de Datos Espaciotemporales	13
4.1. Introduccion	13
4.2. El tiempo como una cuarta dimension	13
4.3. Conceptos	13
4.4. Queries espacio temporales	13
4.5. Algunos modelos	13
5 Conclusiones	15
Bibliografía	17
A Titulo del Apendice	19
A.1. Titulo de la seccion	19

Capítulo 1

Introducción

La idea general es presentar espaciales como algo practica-¿teoria, temporales como teoria-¿practica y despues espacio temporales como una mezcla de ambas culturas”.

Intro hablando sobre el marco general de la mono, detallando que va a tratar cada capitulo y como esta estructurada

En esta trabajo, se presentan conceptos generales y el estado actual de la investigación científica en el área de las Bases de Datos Espaciales, Temporales y Espacio-Temporales.

En el capitulo 1 se presentan conceptos sobre Bases de Datos Espaciales, antecedentes importantes en el procesamiento de datos espaciales como los sistemas GIS, los modelos teóricos que se usan actualmente, diversas aplicaciones y generalizaciones teóricas que nos serán útiles para combinarlas con modelos temporales.

En el capítulo 2 se presentan modelos de Bases de Datos Temporales, las distintas formas de modelarlas teóricamente, el consenso obtenido entre distintos modelos y posterior extensión al estándar SQL y se compara contra el proceso de desarrollo de las Espaciales.

En el capítulo 3 se examina como de la combinación de ambas ramas de investigación y desarrollo surgen las Bases de Datos Espacio-Temporales, las distintas tendencias heredadas de las diferencias historicas entre Temporales y Espaciales que derivan en una variedad de modelos especializados para aplicaciones especificas y se concluye con un análisis de como los autores de este trabajo ven la posible evolución de esta area

Capítulo 2

Bases de Datos Espaciales

2.1. Introducción

Los sistemas de bases de datos espaciales (SDBMS) son los DBMS que incorporan capacidades de representación y manipulación de datos geométricos en un marco de referencia dado.

La principal aplicación de los SDBMS son los Sistemas de Información Geográfica (GIS): software que provee mecanismos de análisis y visualización de de datos geográficos. Los datos geográficos son datos espaciales cuyo marco de referencia es la superficie terrestre. Los GIS implementan en sus herramientas un gran conjunto de técnicas desarrolladas por cartógrafos y que son previas al desarrollo de la informática. Este hecho dota a la investigación de las Bases de Datos Espaciales de un *carácter multidisciplinario* que puede señalarse como una de las causas del rápido avance de la misma.

A su vez, existen otras aplicaciones para los SDBMS tales como la representación de circuitos, datos astronómicos, moléculas y muchas otras. En general, los modelos teóricos son los mismos y lo que cambia es simplemente el marco de referencia en el que se representan los datos. En este trabajo haremos foco en las aplicaciones a los GIS, pero el lector no debe perder de vista que todos los conceptos discutidos son aplicables a estas otras áreas.

2.2. Modelos Teóricos

El desarrollo de modelos teóricos para las Bases de Datos Espaciales debe entenderse en el contexto histórico discutido en la sección anterior. Los primeros GIS implementaban todas las operaciones sobre datos espaciales a nivel de aplicación, guardando sus datos en bases de datos convencionales. A su vez, los primeros SDBMS intentaron simplemente mover esta lógica al nivel de base de datos, pero sin desarrollar un marco teórico. Posteriormente, algunos investigadores comenzaron a dar definiciones formales de Tipos de Datos Espaciales (SDTs).

En las secciones 2.2.1 y 2.2.2 analizaremos brevemente el Álgebra de RoSE [2] (Robust Spatial Extension), un modelo formal que consideramos uno de los más elegantes

y significativos del campo. Luego, en las secciones 2.2.3 y 2.2.4 veremos los principales paradigmas de modelado que se utilizan en los SDBMS actuales, el modelo funcional y el modelo basado en objetos.

2.2.1. Realms

Según Güting y Schneider [2], un modelo formal para Bases de Datos Espaciales debe ser:

- General: los objetos geométricos usados como valores de SDTs deben ser tan generales como sea posible. Por ejemplo, un valor región debe poder representar una colección de áreas disjuntas, cada una de las cuales puede tener agujeros. Mas precisamente, los dominios de los tipos de datos punto, línea y región deben ser cerrados respecto a la unión, intersección y diferencia de sus conjuntos de puntos subyacentes.
- Riguroso: la semántica de los SDTs, es decir, los posibles valores para los tipos y las funciones asociadas con las operaciones, deben estar definidas formalmente para evitar ambigüedades para el usuario y el implementador.
- De resolución finita: las definiciones formales deben tener en cuenta las capacidades de representación finitas de las computadoras. Delegarle al programador la responsabilidad de cerrar la brecha entre teoría y práctica en este punto lleva a errores numéricos y topológicos.
- Geométricamente consistente: distintos objetos espaciales pueden estar relacionados mediante restricciones geométricas. Las definiciones de los SDTs ayudan a mantener esa consistencia.

El Álgebra de RoSE se basa en la incorporación a los DMBSs del concepto de realm, un conjunto finito de puntos y segmentos sin intersecciones sobre los cuales se posicionan todos los datos espaciales de una base de datos. Con este enfoque, los datos no se crean al darle valor a un atributo, sino que se *seleccionan* del conjunto de valores existentes en el realm. Este modelo permite asegurar que el álgebra espacial es cerrada respecto a un realm. Es decir, que las primitivas geométricas y las operaciones sobre realms se definen sobre aritmética entera, libre de errores de redondeo que podrían aparecer si el modelo se definiera sobre un espacio euclídeo.

Otra ventaja de los realms es que permiten aislar todo el computo de puntos en las operaciones de actualización del realm. No se computan intersecciones durante consultas de búsqueda.

Para mapear los segmentos con intersecciones de una aplicación a un conjunto de segmentos sin intersecciones de un realm, se utiliza redibujado y geometría de resolución finita [1].

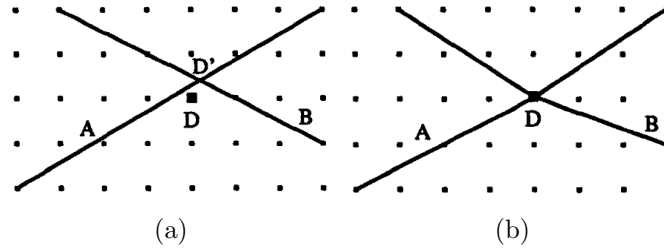


Figura 2.1: Redibujado de la intersección de dos segmentos.

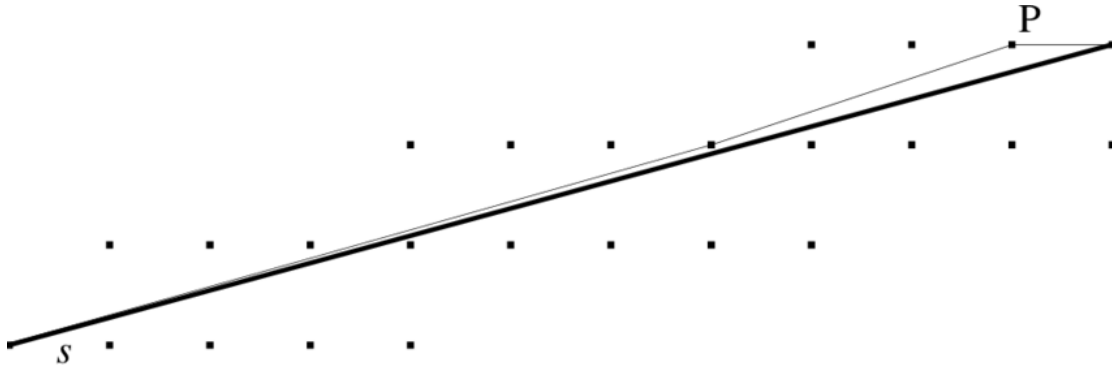


Figura 2.2: Envoltura de un segmento.

2.2.2. Actualizaciones en realms

Veamos mas en detalle que ocurre en este modelo cuando se realiza una actualización que requiere la insercion de un nuevo segmento. Esto es importante porque los datos geométricos provenientes de la aplicación no son no-intersecantes como requiere el modelo. El problema fundamental es que usualmente los puntos de intersección no se corresponden con puntos de la grilla del realm. La solución es aplicar redibujado: se modifican los segmentos de forma tal que la intersección se mueva al punto mas cercano que sí esté en la grilla.

Esto puede verse en las figuras 2.1a y 2.1b: la intersección entre los segmentos A y B (D') no se corresponde con un punto de la grilla, pero podemos partir a cada uno de ellos en dos, de manera que todos los segmentos resultantes tengan a D como uno de sus extremos.

Pero ahora nos surge un nuevo problema que es la preocupación de que aplicando esta operación sucesivamente los segmentos se modifiquen cada vez mas, introduciendo cada vez mas error en la representacion. Podemos acotar este error que se introduce, usando el concepto de envoltura. La envoltura de un segmento es el conjunto de puntos de la grilla que se encuentran inmediatamente arriba, debajo o sobre el segmento. Si establecemos entonces la restriccion de que los puntos a usar para el redibujado deben caer sobre la envoltura del segmento, tenemos una cota para el error: la distancia entre puntos en la grilla. La figura 2.2 muestra un ejemplo de envoltura de un segmento.

De esta forma, Güting y Schneider definen un conjunto de operaciones geométricas primitivas robustas con error acotado que son la base para, a través de la composición de las mismas, definir de forma rigurosa todos los SDTs y las operaciones geométricas que nos pueden resultar interesantes para la representación de datos espaciales. Si bien el Álgebra de RoSE fue un hito en términos de rigurosidad teórica en los modelos de bases de datos espaciales, no es utilizada actualmente en ninguno de los SDBMSs actualmente en uso por la industria del software. De todas formas, su influencia puede verse en los modelos que se utilizan actualmente y que veremos a continuación.

2.2.3. Modelo Funcional

En esta sección, veremos el modelo funcional o modelo orientado a campos. Comenzaremos presentando un ejemplo de un problema espacial a modelar. Supongamos un parque nacional que cuenta con un conjunto de bosques, cada uno con una especie de árbol dominante.

En el modelo funcional, se pueden modelar los bosques del parque nacional como una función cuyo dominio es el conjunto de puntos en el mapa del parque nacional y el rango es un conjunto de especies de árboles. En este caso, se trata de una función escalonada (constante dentro en un mismo bosque y con saltos en los bordes entre bosques).

En general, para modelar un problema usando el paradigma funcional, se deben definir tres componentes [7]:

- Un sistema de referencia espacial: una grilla finita sobre el espacio subyacente, similar al concepto que presentamos en la sección 2.2.1 sobre el Algebra de RoSE. El ejemplo mas conocido es un sistema de referencia de la tierra por latitud y longitud.
- Campos simples: un conjunto finito de funciones computables $\{f_i : SF \rightarrow A_i, 1 \leq i \leq n\}$ donde SF es el sistema de referencia espacial y los A_i son dominios de atributos. La elección de estos dominios de atributos y las funciones que asignan elementos de los mismos a cada punto del espacio dependerá de la aplicación en cuestión.
- Operaciones de campos: especifican las relaciones e interacciones entre los distintos campos. Estas se dividen en:
 - Operaciones locales: operaciones en las que el valor del campo resultante solo depende de los valores de los campos de entrada en ese punto. La composición de campos es un ejemplo de operación local.
 - Operaciones focales: operaciones en las que el valor del campo resultante depende de los valores de los campos en una pequeña vecindad alrededor del punto. Este es el caso por ejemplo de operaciones que involucran gradientes.
 - Operaciones zonales: operaciones que involucran operadores agregados como el promedio o la integración. También son operaciones zonales aquellas que particionan el espacio en zonas.

Los modelos funcionales son buenos para representar problemas en los que las datos continuos como elevación, temperatura y variación del suelo. Tienen la ventaja de poder representar fenómenos espaciales amorfos o con contornos fluidos.

2.2.4. Modelos Basados en Objetos

Volviendo al ejemplo del parque nacional, si consideramos los lugares donde los bosques cambian, en un entorno idealizado donde los bordes entre bosques están claramente definidos, obtendremos los bordes de polígonos. A cada polígono se le puede asignar un identificador y un atributo no espacial (la especie de árbol dominante). Los bosques del parque pueden modelarse entonces como un conjunto de polígonos.

En el modelado basado en objetos, se abstrae información espacial en conjuntos de entidades únicas, distinguibles y relevantes llamadas objetos. Cada objeto tiene un conjunto de atributos que lo caracterizan, los cuales se dividen en atributos espaciales y no espaciales. En el ejemplo del parque nacional, un objeto bosque tiene un atributo espacial, un polígono, que representa su extensión espacial, y un atributo no espacial, su especie dominante, representada como un valor alfanumérico. Es interesante notar que un objeto puede tener varios atributos espaciales. Por ejemplo, un camino puede tener un polígono y una línea, para elegir cuál usar según la escala del mapa.

Los modelos de objetos son los mas comunes en problemas relacionados a representar redes de transportes o parcelas de tierra. Tienen la ventaja de ser mas intuitivos y cercanos a los modelos Entidad-Relación ya muy presentes en los DBMS más usados. En definitiva, la decisión de usar el paradigma funcional o de en objetos se basará en los requerimientos de la aplicación.

En las próximas secciones, detallaremos más los SDTs y operaciones espaciales en el modelo basado en objetos.

2.2.5. SDTs

Se han propuesto muchos conjuntos básicos de tipos espaciales para la representación de formas comunes en mapas. Actualmente, la más adoptada es el estándar OGIS [3] y es en la que nos basaremos. El estándar OGIS propone los siguientes tipos de datos:

- Geometría: es un tipo abstracto (no puede ser instanciado) del cual se derivan todos los otros tipos. Tiene asociado un sistema de referencia.
- Punto: describe la posición de objetos de cero dimensiones, como una oficina o una zona de campamento, en el caso del parque nacional.
- Cadena de Líneas: describe un objeto de una dimension a través de una sucesión de dos o mas puntos, que pueden representar un segmento o aproximar una curva. Por ejemplo, caminos o ríos.
- Polígono: objetos de 2 dimensiones, como un bosque.

- Colección de Geometrías: formas complejas formadas por conjuntos de otros tipos. Se dividen en multipuntos, multilineas y multipoligonos. Estos tipos son necesarios para asegurar que las operaciones geométricas sean cerradas. Por ejemplo, la intersección de un río con un bosque de forma cóncava pueden ser representado por un conjunto de líneas.

2.2.6. Operaciones Espaciales

En el modelo funcional, las relaciones disponibles quedaban definidas por las funciones de campos del modelo. Pero en un modelo basado en objetos, el espacio subyacente será el que determine las operaciones disponibles y las relaciones que pueden existir entre objetos. La siguiente tipología de espacios resume la presentada por [4]:

- Espacios Orientados a Conjuntos: son los espacios mas simples y generales. Permiten todas las relaciones usuales de conjuntos como unión, intersección, contención y pertenencia.
- Espacios Topológicos: los espacios topológicos son aquellos en los que las relaciones no se ven afectadas por transformaciones elásticas del mismo.
- Espacios Direccionales: las relaciones direccionales de un espacio pueden ser absolutas (en referencia a un sistema de referencia global), relativas (basadas en la orientación de un objeto) o basadas en un observador (un objeto especial designado como tal).
- Espacios Métricos: son aquellos espacios en los que la noción de distancia está bien definida. La función de distancia puede usarse para definir una topología y por lo tanto todo espacio métrico es también un espacio topológico.
- Espacios Euclidianos: Son espacios en los que los puntos pueden expresarse como la combinación lineal de los elementos de un sistema de referencia. Sobre espacios euclidianos pueden definirse todas las operaciones de las categorías anteriores.

2.3. Procesamiento de consultas espaciales

En esta sección exploraremos una técnica para la optimización de consultas espaciales sumamente importante, el paradigma de filtrado y refinamiento.

Los usuarios expresan las consultas a un motor de bases de datos utilizando un lenguaje declarativo como SQL. Esto quiere decir que el usuario solo define qué resultado quiere, pero es responsabilidad del motor de bases de datos determinar cuál es el proceso más eficiente para llegar a ese resultado y luego ejecutarlo. A esto le llamamos procesamiento de consultas.

Las consultas a su vez, pueden dividirse en las de escaneo simple o escaneo múltiple. Es una consulta de escaneo simple, cada registro de una tabla deberá ser accedido a lo sumo una vez durante el procesamiento de la consulta. La operación Join es el ejemplo típico de una operación de escaneo múltiple

Capítulo 3

Bases de Datos Temporales

3.1. El dominio del tiempo

Estudiaremos ahora el problema de representar datos temporales y relaciones temporales entre datos, ambos aspectos muy importantes de los fenómenos del mundo real. La habilidad de modelar esta dimension temporal es esencial para aplicaciones informaticas como econometría, bancos, control de inventario, contabilidad, leyes, registros médicos, sistemas de reservas de vuelos y muchas otras.

Podemos introducir la idea de bases de datos temporales como una generalización a la siguiente intuición. Una base de datos tradicional representa el estado del sistema en un momento específico: el presente. En este sentido, a medida que la base de datos se modifica, las versiones anteriores se pierden. Los datos desactualizados son borrados de la base de datos. Surge entonces la problemática de las aplicaciones para las cuales nos sería útil conservar estos estados anteriores y las relaciones entre sus datos.

Para poder plantear una implementación para una base de datos temporal es necesario comprender cómo se comporta y cómo se observa el tiempo. A lo largo de este capitulo exploraremos algunos modelos y representaciones del mismo junto con las elecciones para poder plantear una solución a los problemas temporales.

3.1.1. La estructura del tiempo

En el capítulo anterior vimos como la investigación de bases de datos espaciales se vio motivada por la necesidad práctica de mejorar el manejo de datos de los sistemas GIS. Las bases de datos temporales por su parte, fueron fuertemente impactadas por la investigación previa en lógica temporal de los años 70 [6]. Los trabajos fundacionales de la lógica temporal han dividido el estudio del tiempo en dos modelos estructurales principales: **lineal** y **ramificado** [5]. Comenzaremos entonces por estudiar estos modelos y luego mencionaremos brevemente otras posibilidades.



Figura 3.1: Ejemplo de modelo lineal del tiempo

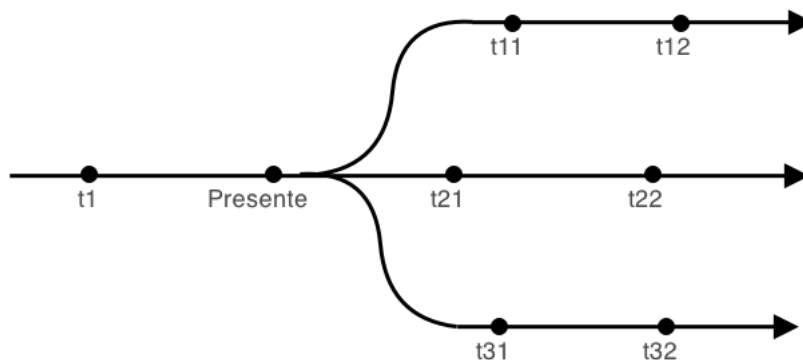


Figura 3.2: Ejemplo de modelo ramificado del tiempo

Modelo Lineal

Es el modelo más simple, donde el tiempo avanza del pasado al futuro de forma ordenada formando así una secuencia tal y como se ve en la figura 3.1.

Generalmente esta estructura es útil para mantener información histórica centrada en eventos del pasado. Por ejemplo: registros de transacciones de un banco, sistemas de control de inventario, etc.

Modelo ramificado

A diferencia del modelo anterior, el tiempo se presenta dos formas: se extiende de forma lineal desde el pasado hacia el presente formando una secuencia. Podemos visualizar este modelo en la figura 3.2 y se ramifica del presente al futuro creando un árbol con raíz en el *ahora* cuyas ramas se extienden hacia el futuro.

A esta forma de organizar el tiempo también se la conoce con el nombre de “modelo de los posibles futuros”. Debido a sus características es útil para representar y trabajar con información predicativa.

Elección entre estructuras temporales

Los modelos más generales del tiempo en la lógica temporal representan el tiempo como un conjunto arbitrario que cumple con la restricción de ser un *orden parcial*. Otros axiomas introducen otros modelos más refinados del tiempo. Por ejemplo: el tiempo

lineal puede ser simplificado agregando un axioma imponiendo un *orden total* en este conjunto. Un modelo recurrente, por otra parte, puede trabajarse con un *modelo cíclico* del tiempo.

A lo largo de este trabajo se utilizará un modelo lineal del tiempo ya que es el más sencillo y brinda todas las características necesarias en las bases de datos temporales.

3.1.2. La densidad del tiempo

Siguiendo con el modelo lineal, la densidad del tiempo puede variar en la línea. Esta se puede clasificar en dos grupos: *discretos* y *densos*.

Modelos Discretos

Estas representaciones son isomorfas a los números naturales, lo que implica que cada punto en el tiempo tiene un único sucesor. Cada número natural corresponde a una unidad no descomponible de tiempo de duración arbitraria. Estas reciben el nombre de *chronons*.

Modelos Densos

Por su parte, los modelos densos, presentan dos variaciones: pueden ser isomorfos tanto a los números racionales como a los reales. Como consecuencia, dados dos momentos cualquiera en el tiempo existe otro momento entre ellos.

Los modelos isomorfos a los reales también son conocidos como continuos. En ellos cada número real corresponde a un punto en el tiempo.

Chronons y la elección del modelo de densidad

Un **chronon** es la duración de tiempo más pequeña que puede ser representada. Cabe destacar que no se trata de un punto sino un segmento en la línea del tiempo.

A pesar de que el tiempo en sí mismo suele ser percibido como algo continuo, la mayoría de las propuestas para agregar una dimensión temporal a los modelos de datos relacionales están basados en un modelo discreto del tiempo. Existen varias razones para esta elección entre las que se destacan cuatro:

- Las medidas del tiempo son inherentemente imprecisas. Los instrumentos de medición temporal invariablemente reportan la ocurrencia de eventos en términos de chronons. No en “puntos” de tiempo. Por lo tanto, incluso los eventos así llamados “instantáneos” pueden ser medidos como si hubieran ocurrido durante un chronon, en el mejor de los casos.
- Las referencias del lenguaje más naturales del tiempo son compatibles con el modelo discreto del tiempo. Por ejemplo, cuando se dice que un evento ocurrió a las 4:30 PM, en realidad no se hace referencia a que el evento ocurrió en el “punto” del tiempo asociado a las 4:302 PM, sino a algún momento en el chronon asociado con las 4:30 PM aunque exista una variación de uno o dos minutos.

- Los conceptos de chronon y de intervalo permiten modelar naturalmente eventos que no son instantáneos pero que tienen duración.
- Cualquier implementación de un modelo de datos con dimensión temporal tendrá la necesidad de codificar el tiempo de forma discreta, es decir, que es inevitable discretizar el tiempo en algún punto.

Capítulo 4

Bases de Datos Espaciotemporales

4.1. Introduccion

Comparacion de las dos culturas de investigacion que ya discutimos para introducir que hay varias tendencias en el campo.

4.2. El tiempo como una cuarta dimension

En esta seccion hablamos de lo intuitivo que es combinar ambos problemas pero a la vez las complicaciones que trae.

4.3. Conceptos

Use the template *chapter.tex* together with the Springer document class SVMono (monograph-type books) or SVMult (edited books) to style the various elements of your chapter content in the Springer layout.

4.4. Queries espacio temporales

Mencionamos por arriba queries propias de espacio temporales que podrian ser interesantes. Algo estilo “ademas de las combinaciones obvias, hay algunas muy propias de espaciotemporales.”

4.5. Algunos modelos

Los mismos de las slides. La reflexion de que parte de diseñar una BDET es elegir el modelo que mas te sirva para el caso.

Capítulo 5

Conclusiones

Primero ponemos que espacio-temporales tiene problemas propios que no son la simple combinacion de resolver al mismo tiempo problemas espaciales y temporales, sino que emergen propiedades y problemas nuevos.

Terminamos con la conclusion del final de las slides. Que aun no hay consenso científico de si vamos a seguir ramificando o se va a encontrar un modelo general para todos los usos.

Bibliografía

- [1] D. H. Greene y F. F. Yao. «Finite-resolution computational geometry». En: *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*. 1986, págs. 143-152.
- [2] R. Gting, M. Schneider, P. Iv y F. Hagen. «Realm-Based Spatial Data Types: The ROSE Algebra». En: (dic. de 1995).
- [3] OGC. *Open GIS Consortium, Inc. OpenGIS® Simple Features Specification For SQL Revision 1.1*. 1999.
- [4] S. Shekhar y S. Chawla. *Spatial databases - a tour*. Ene. de 2003. ISBN: 978-0-13-017480-2. DOI: 10.1016/B0-12-369398-5/00337-6.
- [5] R. Snodgrass. «Temporal Databases». En: 19 (oct. de 1986), págs. 35-42. DOI: 10.1109/MC.1986.1663327.
- [6] R. Snodgrass. «Temporal Databases - Status and Research Directions.» En: *SIG-MOD Record* 19 (dic. de 1990), págs. 83-89. DOI: 10.1145/122058.122068.
- [7] M. Worboys y M. Duckham. «GIS, a Computing Perspective». En: (ene. de 2004).

Apéndice A

Titulo del Apendice

A.1. Titulo de la seccion