

Bases de Datos Temporales

Nicolás Del Piano

Bases de Datos Avanzadas

Introducción

Introducción

El tiempo es un aspecto importante para los fenómenos del mundo real.

Introducción

El tiempo es un aspecto importante para los fenómenos del mundo real.

Los eventos ocurren en momentos de tiempo específicos.

Introducción

El tiempo es un aspecto importante para los fenómenos del mundo real.

Los eventos ocurren en momentos de tiempo específicos.

A veces, nos interesa saber con cierta certeza cuándo ocurrió tal evento, y poder compararlo con otros para obtener información de interés.

Introducción

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.
- ▶ Registros médicos.

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.
- ▶ Registros médicos.
- ▶ Sistemas de información geográfica.

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.
- ▶ Registros médicos.
- ▶ Sistemas de información geográfica.
- ▶ Operaciones bancarias.

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.
- ▶ Registros médicos.
- ▶ Sistemas de información geográfica.
- ▶ Operaciones bancarias.
- ▶ Data Warehousing.

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.
- ▶ Registros médicos.
- ▶ Sistemas de información geográfica.
- ▶ Operaciones bancarias.
- ▶ Data Warehousing.
- ▶ Sistemas de control de reservas (aerolíneas, hoteles, etc.).

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.
- ▶ Registros médicos.
- ▶ Sistemas de información geográfica.
- ▶ Operaciones bancarias.
- ▶ Data Warehousing.
- ▶ Sistemas de control de reservas (aerolíneas, hoteles, etc.).
- ▶ Aplicaciones científicas.

Introducción

Muchas de las áreas donde se aplican las bases de datos tienen naturaleza temporal:

- ▶ Control de inventario.
- ▶ Registros médicos.
- ▶ Sistemas de información geográfica.
- ▶ Operaciones bancarias.
- ▶ Data Warehousing.
- ▶ Sistemas de control de reservas (aerolíneas, hoteles, etc.).
- ▶ Aplicaciones científicas.
- ▶ ... y muchos aspectos más.

Relaciones no temporales

Relaciones no temporales

Id	Nombre	Estado	Sueldo
1	Juan	Activo	5700
2	Manuel	Activo	2300

Relaciones no temporales

Id	Nombre	Estado	Sueldo
1	Juan	Activo	5700
2	Manuel	Activo	2300

Muy comunes en los modelos relacionales.

Relaciones no temporales

Id	Nombre	Estado	Sueldo
1	Juan	Activo	5700
2	Manuel	Activo	2300

Muy comunes en los modelos relacionales.

Cada tupla representa un hecho que es verdadero ahora. Solo hay un estado representable de la base de datos: el actual (*current snapshot*).

Relaciones no temporales

Id	Nombre	Estado	Sueldo
1	Juan	Activo	5700
2	Manuel	Activo	2300

Muy comunes en los modelos relacionales.

Cada tupla representa un hecho que es verdadero ahora. Solo hay un estado representable de la base de datos: el actual (*current snapshot*).

A medida que el tiempo transcurre, los datos se van actualizando y modificando.

Relaciones no temporales

Id	Nombre	Estado	Sueldo
1	Juan	Activo	5700
2	Manuel	Activo	2300

Muy comunes en los modelos relacionales.

Cada tupla representa un hecho que es verdadero ahora. Solo hay un estado representable de la base de datos: el actual (*current snapshot*).

A medida que el tiempo transcurre, los datos se van actualizando y modificando.

Con este modelo, perdemos información.

Problemática

Las bases de datos convencionales representan el estado de la información en un instante de tiempo dado.

Problemática

Las bases de datos convencionales representan el estado de la información en un instante de tiempo dado.

Aunque la base de datos es actualizada, estos cambios son vistos como modificaciones del estado actual y los datos obsoletos son borrados.

Problemática

Las bases de datos convencionales representan el estado de la información en un instante de tiempo dado.

Aunque la base de datos es actualizada, estos cambios son vistos como modificaciones del estado actual y los datos obsoletos son borrados.

Por lo tanto, sólo podemos utilizar la información actual de la base de datos.

Las bases de datos convencionales representan el estado de la información en un instante de tiempo dado.

Aunque la base de datos es actualizada, estos cambios son vistos como modificaciones del estado actual y los datos obsoletos son borrados.

Por lo tanto, sólo podemos utilizar la información actual de la base de datos.

Esto genera un problema cuando queremos responder preguntas involucradas a intervalos de tiempo:

¿Cuáles empleados percibieron un aumento el mes pasado?

¿Qué es una Base de Datos Temporales?

¿Qué es una Base de Datos Temporales?

Un **DBMS Temporal** es un Sistema de Gestión de Base de Datos que proporciona herramientas para el manejo y control de Bases de Datos Temporales.

¿Qué es una Base de Datos Temporales?

Un **DBMS Temporal** es un Sistema de Gestión de Base de Datos que proporciona herramientas para el manejo y control de Bases de Datos Temporales.

Una **Base de Datos Temporales** es una Base de Datos que tiene dimensión del tiempo a través del almacenamiento de **datos temporales**.

¿Qué es una Base de Datos Temporales?

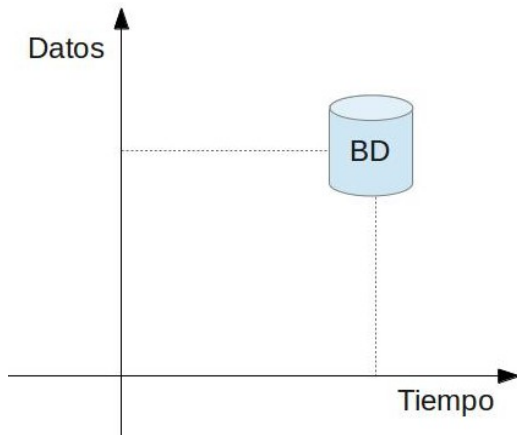
Un **DBMS Temporal** es un Sistema de Gestión de Base de Datos que proporciona herramientas para el manejo y control de Bases de Datos Temporales.

Una **Base de Datos Temporales** es una Base de Datos que tiene dimensión del tiempo a través del almacenamiento de **datos temporales**.

Proporcionan un marco que mantiene la historia de los cambios que se produjeron en la fuente de datos.

¿Qué es una Base de Datos Temporales?

Están diseñadas para la captura de la información que varía en el transcurso del tiempo.



Datos Temporales

Datos Temporales

Un dato temporal es un dato convencional al que se le asocia un período de tiempo para expresar valores temporales en la base de datos.

Datos Temporales

Un dato temporal es un dato convencional al que se le asocia un período de tiempo para expresar valores temporales en la base de datos.

Este agregado de información temporal se denomina *time-stamping*.

Datos Temporales

Un dato temporal es un dato convencional al que se le asocia un período de tiempo para expresar valores temporales en la base de datos.

Este agregado de información temporal se denomina *time-stamping*.

Al asociar el tiempo con la información, es posible almacenar diferentes estados de una base de datos.

Dimensión del tiempo

Las Bases de Datos Temporales almacenan dos dimensiones del tiempo:

- ▶ Tiempo Válido
- ▶ Tiempo Transaccional

Dimensión del tiempo: Tiempo Válido

Dimensión del tiempo: Tiempo Válido

Representa cuándo un hecho tiene validez, i.e, es verdadero en el mundo real.

Dimensión del tiempo: Tiempo Válido

Representa cuándo un hecho tiene validez, i.e, es verdadero en el mundo real.

Es independiente de si dicho evento fue registrado o no en la base de datos.

Dimensión del tiempo: Tiempo Válido

Representa cuándo un hecho tiene validez, i.e, es verdadero en el mundo real.

Es independiente de si dicho evento fue registrado o no en la base de datos.

Los Tiempos Válidos pueden encontrarse en el pasado, presente o futuro.

Dimensión del tiempo: Tiempo Válido

Representa cuándo un hecho tiene validez, i.e, es verdadero en el mundo real.

Es independiente de si dicho evento fue registrado o no en la base de datos.

Los Tiempos Válidos pueden encontrarse en el pasado, presente o futuro.

Todos los eventos tienen asociado un Tiempo Válido, pero no necesariamente son registrados en una base de datos.
Permiten llevar una historia de la BD.

Dimensión del tiempo: Tiempo Transaccional

Dimensión del tiempo: Tiempo Transaccional

Registra el período de tiempo donde un hecho fue almacenado en la base de datos.

Dimensión del tiempo: Tiempo Transaccional

Registra el período de tiempo donde un hecho fue almacenado en la base de datos.

Permiten consultas que muestren el estado de la base de datos en un tiempo dado.

Dimensión del tiempo: Tiempo Transaccional

Registra el período de tiempo donde un hecho fue almacenado en la base de datos.

Permiten consultas que muestren el estado de la base de datos en un tiempo dado.

Está acotado en ambos extremos; la creación de la base de datos y el tiempo presente.

Dimensión del tiempo: Tiempo Transaccional

Registra el período de tiempo donde un hecho fue almacenado en la base de datos.

Permiten consultas que muestren el estado de la base de datos en un tiempo dado.

Está acotado en ambos extremos; la creación de la base de datos y el tiempo presente.

Permiten hacer rollback de los datos.

Ejemplo

Ejemplo

El Señor X nace en Springfield el 12 de Mayo de 1956.



Ejemplo

El Señor X nace en Springfield el 12 de Mayo de 1956.



Su padre lo registra el 13 de Mayo de 1956.

Ejemplo

El Señor X nace en Springfield el 12 de Mayo de 1956.



Su padre lo registra el 13 de Mayo de 1956.

Se muda a Arroyos Cipreses el 3 de Agosto de 1980, pero olvida registrarse; lo hace el 16 de Agosto del mismo año.

Ejemplo

El Señor X nace en Springfield el 12 de Mayo de 1956.



Su padre lo registra el 13 de Mayo de 1956.

Se muda a Arroyos Cipreses el 3 de Agosto de 1980, pero olvida registrarse; lo hace el 16 de Agosto del mismo año.

Muere el 20 de Abril de 2004.

Ejemplo: no temporal

Nombre	ViveEn
Señor X	Springfield

↓ Update

Nombre	ViveEn
Señor X	Arroyos Cipreses

↓ Delete

Nombre	ViveEn
Señor X	Arroyos Cipreses

Ejemplo: Tiempo Válido

Nombre	ViveEn	Valid-From	Valid-To
Señor X	Springfield	12-May-1956	∞

↓ Update

Nombre	ViveEn	Valid-From	Valid-To
Señor X	Springfield	12-May-1956	2-Aug-1980

↓ Insert

Nombre	ViveEn	Valid-From	Valid-To
Señor X	Springfield	12-May-1956	2-Aug-1980
Señor X	Arroyos Cipreses	3-Aug-1980	∞

↓ Update

Nombre	ViveEn	Valid-From	Valid-To
Señor X	Springfield	12-May-1956	2-Aug-1980
Señor X	Arroyos Cipreses	3-Aug-1980	20-Apr-2004

Ejemplo: Tiempo Transaccional

Nombre	ViveEn	Transaction-From	Transaction-To
Señor X	Springfield	13-May-1956	∞

↓ Insert

Nombre	ViveEn	Transaction-From	Transaction-To
Señor X	Springfield	13-May-1956	16-Aug-1980
Señor X	Arroyos Cipreses	16-Aug-1980	∞

↓ Update

Nombre	ViveEn	Transaction-From	Transaction-To
Señor X	Springfield	13-May-1956	16-Aug-1980
Señor X	Arroyos Cipreses	16-Aug-1980	20-Apr-2004

Base de Datos Bi-Temporal

Base de Datos Bi-Temporal

Incluyen ambos tiempos, Válido y Transaccional.

Base de Datos Bi-Temporal

Incluyen ambos tiempos, Válido y Transaccional.

Proveen información **histórica** y permiten hacer **rollback** de los datos.

Nombre	ViveEn	Valid-From	Valid-To	Transaction-From	Transaction-To
Señor X	Springfield	12-May-1956	∞	13-May-1956	16-Aug-1980
Señor X	Springfield	12-May-1956	2-Aug-1980	16-Aug-1980	∞
Señor X	Arroyos Cipreses	3-Aug-1980	∞	16-Aug-1980	20-Apr-2004
Señor X	Arroyos Cipreses	3-Aug-1980	20-Apr-2004	20-Apr-2004	∞

Representación del tiempo válido y transaccional

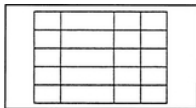


Figure 1. A snapshot relation.

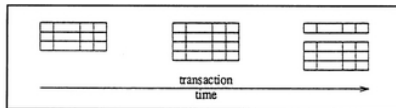


Figure 2. A rollback relation.

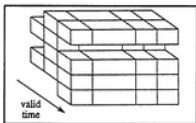


Figure 3. An historical relation.

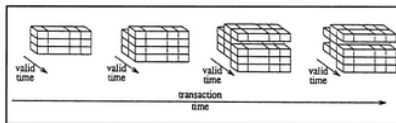


Figure 4. A temporal relation.

Extensiones Temporales

Extensiones Temporales

Hay dos formas de extender el modelo relacional para especificar requisitos temporales.

Extensiones Temporales

Hay dos formas de extender el modelo relacional para especificar requisitos temporales.

La forma mostrada en los ejemplos antes mencionados se denomina **marcaje de tuplas**.

Extensiones Temporales

Hay dos formas de extender el modelo relacional para especificar requisitos temporales.

La forma mostrada en los ejemplos antes mencionados se denomina **marcaje de tuplas**.

$$(attr_1, \dots, attr_n) \rightarrow (attr_1, \dots, attr_n, temp_attr_1, \dots, temp_attr_m)$$

Extensiones Temporales

Hay dos formas de extender el modelo relacional para especificar requisitos temporales.

La forma mostrada en los ejemplos antes mencionados se denomina **marcaje de tuplas**.

$$(attr_1, \dots, attr_n) \rightarrow (attr_1, \dots, attr_n, temp_attr_1, \dots, temp_attr_m)$$

La forma restante es denominada **marcaje de atributos** y usa atributos multivaluados.

Extensiones Temporales

Hay dos formas de extender el modelo relacional para especificar requisitos temporales.

La forma mostrada en los ejemplos antes mencionados se denomina **marcaje de tuplas**.

$$(attr_1, \dots, attr_n) \rightarrow (attr_1, \dots, attr_n, temp_attr_1, \dots, temp_attr_m)$$

La forma restante es denominada **marcaje de atributos** y usa atributos multivaluados.

Al mismo tiempo que permite la correspondencia 1:1 entre entidades y hechos reales, dificulta las actualizaciones y no cumple la 1NF.

Operadores de Allen

Operadores de Allen

Necesitamos una forma de comparar datos temporales.

Operadores de Allen

Necesitamos una forma de comparar datos temporales.

Allen (1983) propone un conjunto de operadores temporales lógicos para comparar intervalos de tiempo.

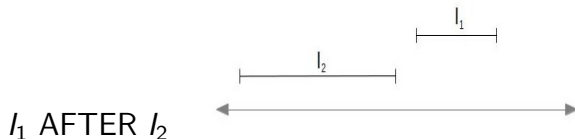
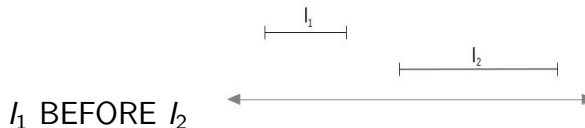
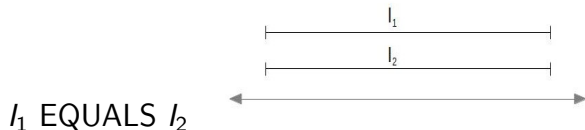
Operadores de Allen

Necesitamos una forma de comparar datos temporales.

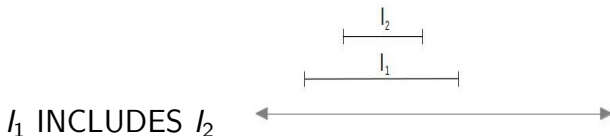
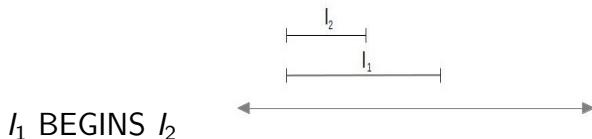
Allen (1983) propone un conjunto de operadores temporales lógicos para comparar intervalos de tiempo.

El operador es una función de tipo: $I_t \times I_t \rightarrow \{True, False\}$.

Algunos operadores temporales (1)



Algunos operadores temporales (2)



Modelo relacional temporal

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Utiliza marcaje de tuplas.

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Utiliza marcaje de tuplas.

Algunas de las características que debe ofrecer son:

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Utiliza marcaje de tuplas.

Algunas de las características que debe ofrecer son:

- ▶ Un tipo de datos para períodos de tiempo, incluyendo la habilidad de representar períodos infinitos (sin fin).

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Utiliza marcaje de tuplas.

Algunas de las características que debe ofrecer son:

- ▶ Un tipo de datos para períodos de tiempo, incluyendo la habilidad de representar períodos infinitos (sin fin).
- ▶ Soporte para tiempo válido, transaccional y tablas bitemporales.

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Utiliza marcaje de tuplas.

Algunas de las características que debe ofrecer son:

- ▶ Un tipo de datos para períodos de tiempo, incluyendo la habilidad de representar períodos infinitos (sin fin).
- ▶ Soporte para tiempo válido, transaccional y tablas bitemporales.
- ▶ Tiempo transaccional controlado por el sistema.

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Utiliza marcaje de tuplas.

Algunas de las características que debe ofrecer son:

- ▶ Un tipo de datos para períodos de tiempo, incluyendo la habilidad de representar períodos infinitos (sin fin).
- ▶ Soporte para tiempo válido, transaccional y tablas bitemporales.
- ▶ Tiempo transaccional controlado por el sistema.
- ▶ Consultas temporales.

Modelo relacional temporal

El modelo relacional temporal incorpora la semántica temporal en el modelo relacional.

Utiliza marcaje de tuplas.

Algunas de las características que debe ofrecer son:

- ▶ Un tipo de datos para períodos de tiempo, incluyendo la habilidad de representar períodos infinitos (sin fin).
- ▶ Soporte para tiempo válido, transaccional y tablas bitemporales.
- ▶ Tiempo transaccional controlado por el sistema.
- ▶ Consultas temporales.
- ▶ Predicados y operadores que actúen sobre intervalos de tiempo.

Lenguajes de consulta temporales

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Un lenguaje de consulta temporal es cualquier lenguaje de consulta para bases de datos temporales.

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Un lenguaje de consulta temporal es cualquier lenguaje de consulta para bases de datos temporales.

Las propiedades de un lenguaje de consulta temporal son:

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Un lenguaje de consulta temporal es cualquier lenguaje de consulta para bases de datos temporales.

Las propiedades de un lenguaje de consulta temporal son:

- ▶ Semántica declarativa

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Un lenguaje de consulta temporal es cualquier lenguaje de consulta para bases de datos temporales.

Las propiedades de un lenguaje de consulta temporal son:

- ▶ Semántica declarativa
- ▶ Implementación eficiente

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Un lenguaje de consulta temporal es cualquier lenguaje de consulta para bases de datos temporales.

Las propiedades de un lenguaje de consulta temporal son:

- ▶ Semántica declarativa
- ▶ Implementación eficiente
- ▶ Independencia en la representación

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Un lenguaje de consulta temporal es cualquier lenguaje de consulta para bases de datos temporales.

Las propiedades de un lenguaje de consulta temporal son:

- ▶ Semántica declarativa
- ▶ Implementación eficiente
- ▶ Independencia en la representación
- ▶ Expresividad de la consulta

Lenguajes de consulta temporales

Una base de datos temporales es un repositorio de información temporal.

Un lenguaje de consulta temporal es cualquier lenguaje de consulta para bases de datos temporales.

Las propiedades de un lenguaje de consulta temporal son:

- ▶ Semántica declarativa
- ▶ Implementación eficiente
- ▶ Independencia en la representación
- ▶ Expresividad de la consulta

Algunos ejemplos importantes son: TSQL, TQuel, HRDM, Backlog, etcétera. Muchos otros son derivados de estos.

Los DBMS Temporales deben respaldar un lenguaje de definición de datos temporales, un sistema de restricciones temporales, un lenguaje de manipulación de datos y un lenguaje de consultas temporales.

Los DBMS Temporales deben respaldar un lenguaje de definición de datos temporales, un sistema de restricciones temporales, un lenguaje de manipulación de datos y un lenguaje de consultas temporales.

Algunos ejemplos de DBMS Temporales son: TimeDB, Oracle Workspace Manager, PostgreSQL, IBM DB2, etcétera.

HISTORIA Y ESTANDARIZACIÓN

Hacia un modelo de datos temporal

Hacia un modelo de datos temporal

Hacia un modelo de datos temporal

La comunidad de Bases de Datos Temporales fue prolífica en la producción de modelos de datos temporales y lenguajes de consulta.

Hacia un modelo de datos temporal

La comunidad de Bases de Datos Temporales fue prolífica en la producción de modelos de datos temporales y lenguajes de consulta.

En los últimos 20 años, docenas de modelos relacionales de datos temporales fueron propuestos.

Hacia un modelo de datos temporal

La comunidad de Bases de Datos Temporales fue prolífica en la producción de modelos de datos temporales y lenguajes de consulta.

En los últimos 20 años, docenas de modelos relacionales de datos temporales fueron propuestos.

Richard Snodgrass propuso en 1992 que la comunidad de BDT realice extensiones a SQL.

Proceso de estandarización de SQL

Proceso de estandarización de SQL

INCITS(ANSI) DM32.2 responsable del estándar SQL en USA.

Proceso de estandarización de SQL

INCITS(ANSI) DM32.2 responsable del estándar SQL en USA.

ISO/IEC JTC 1/SC 32 Data Management and Interchange/WG 3 es un comité responsable del estándar SQL internacional.

Proceso de estandarización de SQL

INCITS(ANSI) DM32.2 responsable del estándar SQL en USA.

ISO/IEC JTC 1/SC 32 Data Management and Interchange/WG 3 es un comité responsable del estándar SQL internacional.

Muchas de las capacidades del estándar SQL fueron originadas en USA.

Proceso de estandarización de SQL

INCITS(ANSI) DM32.2 responsable del estándar SQL en USA.

ISO/IEC JTC 1/SC 32 Data Management and Interchange/WG 3 es un comité responsable del estándar SQL internacional.

Muchas de las capacidades del estándar SQL fueron originadas en USA.

La aprobación tiene un ciclo de 3 a 5 años.

Proceso de estandarización de SQL

INCITS(ANSI) DM32.2 responsable del estándar SQL en USA.

ISO/IEC JTC 1/SC 32 Data Management and Interchange/WG 3 es un comité responsable del estándar SQL internacional.

Muchas de las capacidades del estándar SQL fueron originadas en USA.

La aprobación tiene un ciclo de 3 a 5 años.

7 versiones del Standard SQL:

86(SQL-87), 89, 92(SQL2), SQL:1999(SQL3), SQL:2003, SQL:2008, SQL:2011

SQL Temporal: primer intento (1995-2001)

SQL Temporal: primer intento (1995-2001)

X3H2(ahora DM32.2) y WG 3 aprobaron el trabajo SQL/Temporal en 1995.

SQL Temporal: primer intento (1995-2001)

X3H2(ahora DM32.2) y WG 3 aprobaron el trabajo SQL/Temporal en 1995.

USA fue el primero en hacer la propuesta, añadiendo nuevas extensiones a SQL, basadas en el trabajo de Rick Snodgrass.

SQL Temporal: primer intento (1995-2001)

X3H2(ahora DM32.2) y WG 3 aprobaron el trabajo SQL/Temporal en 1995.

USA fue el primero en hacer la propuesta, añadiendo nuevas extensiones a SQL, basadas en el trabajo de Rick Snodgrass.

La propuesta de USA estaba basada en TSQL2, una extensión de SQL-92.

SQL Temporal: primer intento (1995-2001)

X3H2(ahora DM32.2) y WG 3 aprobaron el trabajo SQL/Temporal en 1995.

USA fue el primero en hacer la propuesta, añadiendo nuevas extensiones a SQL, basadas en el trabajo de Rick Snodgrass.

La propuesta de USA estaba basada en TSQL2, una extensión de SQL-92.

UK lanzó una propuesta similar. Conflictos entre las propuestas de USA y UK, terminaron en que ANSI e ISO decidieran en cancelar SQL/Temporal en 2001.

SQL Temporal: segundo intento (2008-2011)

SQL Temporal: segundo intento (2008-2011)

En 2008, un segundo intento se hizo presente. Empezó con la aceptación de la propuesta "system-versioned tables" llevada a cabo por INCITS DM32.2 y ISO/IEC JTC1 SC32 WG3.

SQL Temporal: segundo intento (2008-2011)

En 2008, un segundo intento se hizo presente. Empezó con la aceptación de la propuesta "system-versioned tables" llevada a cabo por INCITS DM32.2 y ISO/IEC JTC1 SC32 WG3.

No se resucitó SQL/Temporal, sino se añadió la propuesta a SQL/Foundation.

SQL Temporal: segundo intento (2008-2011)

En 2008, un segundo intento se hizo presente. Empezó con la aceptación de la propuesta "system-versioned tables" llevada a cabo por INCITS DM32.2 y ISO/IEC JTC1 SC32 WG3.

No se resucitó SQL/Temporal, sino se añadió la propuesta a SQL/Foundation.

Otra característica es "application-time period tables".

SQL Temporal: segundo intento (2008-2011)

En 2008, un segundo intento se hizo presente. Empezó con la aceptación de la propuesta "system-versioned tables" llevada a cabo por INCITS DM32.2 y ISO/IEC JTC1 SC32 WG3.

No se resucitó SQL/Temporal, sino se añadió la propuesta a SQL/Foundation.

Otra característica es "application-time period tables".

Las características temporales en SQL:2011 están inspiradas en las anteriores propuestas hechas en SQL/Temporal, pero con una sintaxis un poco diferente.

Inclusión de tipos de datos temporales:

Inclusión de tipos de datos temporales:

- ▶ DATE (10 posiciones) = YEAR, MONTH, DAY (yyyy-mm-dd)
- ▶ TIME (8 posiciones) = HOUR, MINUTE, SECOND (hh:mm:ss)
- ▶ TIMESTAMP (DATE, TIME, fracciones de segundo y desplazamiento de acuerdo al huso horario estándar)
- ▶ INTERVAL
Período de tiempo. Para incrementar/decrementar el valor actual de una fecha, hora o marca de tiempo.
Se califica con YEAR/MONTH o DAY/TIME para indicar su naturaleza.

TSQL2 (1)

TSQL2 (1)

Es un lenguaje de consulta temporal consensuado por un comité de grupos de investigación en bases de datos temporales.

TSQL2 (1)

Es un lenguaje de consulta temporal consensuado por un comité de grupos de investigación en bases de datos temporales.

Apareció como una extensión para SQL-92.

TSQL2 (1)

Es un lenguaje de consulta temporal consensuado por un comité de grupos de investigación en bases de datos temporales.

Apareció como una extensión para SQL-92.

La especificación incluye ideas y conceptos de Tiempo Válido, Tiempo Transaccional y Tablas Bitemporales.

TSQL2 (1)

Es un lenguaje de consulta temporal consensuado por un comité de grupos de investigación en bases de datos temporales.

Apareció como una extensión para SQL-92.

La especificación incluye ideas y conceptos de Tiempo Válido, Tiempo Transaccional y Tablas Bitemporales.

También conocido como extensiones ANSI X3.135.-1992 e ISO/IEC 9075:1992.

TSQL2 (2)

Incluye cuatro tipos de marcas de tiempo válido: intervalos, instantes, períodos y elementos.

TSQL2 (2)

Incluye cuatro tipos de marcas de tiempo válido: intervalos, instantes, períodos y elementos.

Asociadas a estas marcas existen tres categorías de operadores: extractores, constructores y comparadores.

TSQL2 (2)

Incluye cuatro tipos de marcas de tiempo válido: intervalos, instantes, períodos y elementos.

Asociadas a estas marcas existen tres categorías de operadores: extractores, constructores y comparadores.

Tiene una variable denominada NOW que indica el momento actual → tiempo de referencia.

TSQL2: Extractores

Operación	Operador
Extractor de eventos	BEGIN(event) END(period) BEGIN(element) END(event) END(period) END(element)
Extractor de períodos	FIRST(period) FIRST(element) LAST(period) LAST(element)

TSQL2: Constructores

Operación	Operador
Constructor de eventos	FIRST(event, event) LAST(event, event)
Constructor de períodos	PERIOD(event, event) INTERSECT(period, period)
Constructor de elementos	INTERSECT(element, element) element + element element - element

TSQL2: Comparadores

Operación	Operador
Comparador de eventos	event PRECEDES event event = event event OVERLAPS event event MEETS event event CONTAINS event
Comparador de períodos	period PRECEDES period period = period period OVERLAPS period period MEETS period period CONTAINS period
Comparador de elementos	element PRECEDES element element = element element OVERLAPS element element CONTAINS element

TSQL2: Operadores de Allen

Operador de Allen	Operador TSQL2
a BEFORE b	a PRECEDES b
a EQUAL b	a = b
a OVERLAPS b	a OVERLAPS b AND END(a) PRECEDES END(b)
a MEETS b	END(a) = BEGIN(b)
a DURING b	BEGIN(b) PRECEDES BEGIN(a) AND END(a) PRECEDES END(b)
a START b	BEGIN(a) = BEGIN(b) AND END(a) PRECEDES END(b)
a FINISH b	BEGIN(b) PRECEDES BEGIN(a) AND END(a) = END(b)

Ejemplos de consulta TSQL2: Tiempo Válido (1)

- > CREATE TABLE empleado(ename VARCHAR(12), eno INTEGER PRIMARY KEY, cumple DATE);
- > CREATE TABLE salario(eno INTEGER REFERENCES empleado(eno), sueldo INTEGER);
- > INSERT INTO empleado
VALUES('Homero', 1, DATE '1955-03-21');
- > INSERT INTO empleado
VALUES('Lenny', 2, '1956-09-18');
- > INSERT INTO salario VALUES(1, 2000);
- > INSERT INTO salario VALUES(2, 4000);

Ejemplos de consulta TSQL2: Tiempo Válido (2)

- > ALTER TABLE salario ADD VALIDTIME PERIOD(DATE);
- > ALTER TABLE empleado ADD VALIDTIME PERIOD(DATE);
- > INSERT INTO empleado
VALUES('Carl', 3, DATE '1955-08-10');
- > INSERT INTO salario VALUES(3, 3500);
- > COMMIT;

ename	eno	cumple	Válido
Homero	1	1955-03-21	[1995-02-01 - 9999-12-31)
Lenny	2	1956-09-18	[1995-02-01 - 9999-12-31)
Carl	3	1955-08-10	[1995-02-01 - 9999-12-31)

eno	sueldo	Válido
1	2000	[1995-02-01 - 9999-12-31)
2	4000	[1995-02-01 - 9999-12-31)
3	3500	[1995-02-01 - 9999-12-31)

Ejemplos de consulta TSQL2: Tiempo Válido (3)

Cambiar el nombre de 'Carl' por 'Carlos'.

- > VALIDTIME UPDATE empleado
SET ename = 'Carlos'
WHERE ename = 'Carl';
- > COMMIT;

ename	eno	cumple	Válido
Homero	1	1955-03-21	[1995-02-01 - 9999-12-31)
Lenny	2	1956-09-18	[1995-02-01 - 9999-12-31)
Carlos	3	1955-08-10	[1995-02-01 - 9999-12-31)

Ejemplos de consulta TSQL2: Tiempo Válido (4)

¿Quién percibió un aumento de sueldo?

- > UPDATE salario
 SET sueldo = 1.05 * amount
 WHERE eno = 3;
- > COMMIT;
- > NONSEQUENCED VALIDTIME SELECT ename
 FROM empleado AS E, salario AS S1, salario AS S2
 WHERE E.eno = S1.eno AND E.eno = S2.eno
 AND S1.sueldo < S2.sueldo AND
 VALIDTIME(S1) MEETS VALIDTIME(S2);

ename
Carlos

Ejemplos de consulta TSQL2: Tiempo Transaccional (1)

ename	eno	cumple	Válido
Homero	1	1955-03-21	[1995-02-01 - 9999-12-31)
Lenny	2	1956-09-18	[1995-02-01 - 9999-12-31)
Carlos	3	1955-08-10	[1995-02-01 - 9999-12-31)

- > ALTER TABLE empleado ADD TRANSACTIONTIME;
- > COMMIT;

ename	eno	cumple	Válido	Transacción
Homero	1	1955-03-21	[1995-02-01 - 9999-12-31)	[1995-07-01 - 9999-12-31)
Lenny	2	1956-09-18	[1995-02-01 - 9999-12-31)	[1995-07-01 - 9999-12-31)
Carlos	3	1955-08-10	[1995-02-01 - 9999-12-31)	[1995-07-01 - 9999-12-31)

Ejemplos de consulta TSQL2: Tiempo Transaccional (2)

- > UPDATE empleado
SET ename = 'Homero J.'
WHERE ename = 'Homero';
- > COMMIT;

ename	eno	cumple	Válido	Transacción
Homero	1	1955-03-21	[1995-02-01 - 9999-12-31)	[1995-07-01 - 2014-04-23)
Homero J.	1	1955-03-21	[1995-02-01 - 9999-12-31)	[2014-04-23 - 9999-12-31)
Lenny	2	1956-09-18	[1995-02-01 - 9999-12-31)	[1995-07-01 - 9999-12-31)
Carlos	3	1955-08-10	[1995-02-01 - 9999-12-31)	[1995-07-01 - 9999-12-31)

Ejemplos de consulta TSQL2: Tiempo Transaccional (3)

¿Cuándo trabajó algún empleado por más de seis meses?

ename	eno	cumple	Válido	Transacción
Homero	1	1955-03-21	[1995-02-01 - 9999-12-31)	[1995-07-01 - 2014-04-23)
Homero J.	1	1955-03-21	[1995-02-01 - 9999-12-31)	[2014-04-23 - 9999-12-31)
Lenny	2	1956-09-18	[1995-02-01 - 9999-12-31)	[1995-07-01 - 9999-12-31)
Carlos	3	1955-08-10	[1995-02-01 - 1995-03-31)	[1995-07-01 - 9999-12-31)

> VALIDTIME AND TRANSACTIONTIME SELECT ename, eno
FROM empleado
WHERE INTERVAL(VALIDTIME(empleado) MONTH) >
INTERVAL '6' MONTH;

ename	eno	cumple	Válido	Transacción
Homero	1	1955-03-21	[1995-02-01 - 9999-12-31)	[1995-07-01 - 2014-04-23)
Homero J.	1	1955-03-21	[1995-02-01 - 9999-12-31)	[2014-04-23 - 9999-12-31)
Lenny	2	1956-09-18	[1995-02-01 - 9999-12-31)	[1995-07-01 - 9999-12-31)

ESTADO DEL ARTE

Las BDT en la actualidad

Tipos de soluciones disponibles

Tipos de soluciones disponibles

Actualmente se pueden manipular los datos temporales de las siguientes formas:

Tipos de soluciones disponibles

Actualmente se pueden manipular los datos temporales de las siguientes formas:

- ▶ Usar un tipo de datos temporal integrado al DBMS y brindar soporte temporal con aplicaciones.

Tipos de soluciones disponibles

Actualmente se pueden manipular los datos temporales de las siguientes formas:

- ▶ Usar un tipo de datos temporal integrado al DBMS y brindar soporte temporal con aplicaciones.
- ▶ Implementar un tipo de dato abstracto para el tiempo.

Tipos de soluciones disponibles

Actualmente se pueden manipular los datos temporales de las siguientes formas:

- ▶ Usar un tipo de datos temporal integrado al DBMS y brindar soporte temporal con aplicaciones.
- ▶ Implementar un tipo de dato abstracto para el tiempo.
- ▶ Extender el modelo de datos no temporal a uno temporal.

Tipos de soluciones disponibles

Actualmente se pueden manipular los datos temporales de las siguientes formas:

- ▶ Usar un tipo de datos temporal integrado al DBMS y brindar soporte temporal con aplicaciones.
- ▶ Implementar un tipo de dato abstracto para el tiempo.
- ▶ Extender el modelo de datos no temporal a uno temporal.
- ▶ Generalizar un modelo de datos no temporal en uno temporal.

SQL:2011 (1)

El tiempo transaccional es manejado con *system-versioned tables*, que contienen el período de tiempo del sistema, y el tiempo válido es manejado con tablas que contienen *application-time period*.

SQL:2011 (1)

El tiempo transaccional es manejado con *system-versioned tables*, que contienen el período de tiempo del sistema, y el tiempo válido es manejado con tablas que contienen *application-time period*.

transaction time → system time

valid time → application time

SQL:2011 (1)

El tiempo transaccional es manejado con *system-versioned tables*, que contienen el período de tiempo del sistema, y el tiempo válido es manejado con tablas que contienen *application-time period*.

transaction time → system time

valid time → application time

Las técnicas e ideas de TSQL se tuvieron en cuenta por el comité, pero las extensiones sintácticas que se hicieron difirieron considerablemente de aquellas propuestas en TSQL.

SQL:2011 Application-Time tables

SQL:2011 Application-Time tables

El tipo de datos PERIOD para intervalos de tiempo, sigue no disponible. Es simulado usando pares de instantes (con la semántica [cerrado,abierto)).

SQL:2011 Application-Time tables

El tipo de datos PERIOD para intervalos de tiempo, sigue no disponible. Es simulado usando pares de instantes (con la semántica [cerrado,abierto)).

Application-time period tables son tablas que contienen una cláusula PERIOD, con un nombre del período definido por el usuario.

SQL:2011 Application-Time tables

El tipo de datos PERIOD para intervalos de tiempo, sigue no disponible. Es simulado usando pares de instantes (con la semántica [cerrado,abierto)).

Application-time period tables son tablas que contienen una cláusula PERIOD, con un nombre del período definido por el usuario.

Estas tablas contienen también dos columnas adicionales para almacenar el tiempo de inicio y fin de un período de un dato temporal.

SQL:2011 System-Versioned tables

SQL:2011 System-Versioned tables

Son tablas que contienen una cláusula PERIOD con un nombre de período predefinido (SYSTEM_TIME).

SQL:2011 System-Versioned tables

Son tablas que contienen una cláusula PERIOD con un nombre de período predefinido (SYSTEM_TIME).

Contienen dos columnas adicionales que se refieren a el inicio y fin de una transacción.

SQL:2011 System-Versioned tables

Son tablas que contienen una cláusula PERIOD con un nombre de período predefinido (SYSTEM_TIME).

Contienen dos columnas adicionales que se refieren a el inicio y fin de una transacción.

Ambos valores son seteados por el sistema.

SQL:2011 System-Versioned tables

Son tablas que contienen una cláusula PERIOD con un nombre de período predefinido (SYSTEM_TIME).

Contienen dos columnas adicionales que se refieren a el inicio y fin de una transacción.

Ambos valores son seteados por el sistema.

Preservan las versiones antiguas de las filas.

emp_name	dept_id	start_date	end_date
John	M24	1998-01-31	9999-12-31
John	J13	1995-11-15	1998-01-31
Tracy	K25	1996-01-01	2000-03-31

¿En cuál departamento estuvo John el 1 de Diciembre de 1997?

```
SELECT dept_id  
FROM empleados  
WHERE emp_name = 'John' AND start_date ≤ DATE  
'1997-12-01' AND end_date > DATE '1997-12-01';
```

emp_name	dept_id	start_date	end_date
John	M24	1998-01-31	9999-12-31
John	J13	1995-11-15	1998-01-31
Tracy	K25	1996-01-01	2000-03-31

¿En que departamento está John ahora?

```
SELECT dept_id  
FROM empleados  
WHERE emp_name = 'John' AND start_date ≤  
CURRENT_DATE AND end_date > CURRENT_DATE;
```

SQL:2011 (4)

Se borra la 3 fila el 16/4/2014:

```
DELETE FROM empleados  
WHERE emp_name = 'Tracy';
```

emp_name	dept_id	system_start	system_end
John	M24	1998-01-31	9999-12-31
John	J13	1995-11-15	1998-01-31
Tracy	K25	1995-11-15	2014-04-16

SQL:2011 vs TSQL

TSQL	SQL:2011
valid time transaction time	application time system time
timestamping	versioning
validtime table transactiontime table bitemporal table	application time period table system-versioned table system-versioned application time period table

TimeDB es un DBMS Bitemporal basado en SQL.

TimeDB es un DBMS Bitemporal basado en SQL.

Soporta un lenguaje de consulta, un lenguaje de manipulación de datos y un lenguaje de definición de datos.

TimeDB es un DBMS Bitemporal basado en SQL.

Soporta un lenguaje de consulta, un lenguaje de manipulación de datos y un lenguaje de definición de datos.

TimeDB brinda ATSQL2, un lenguaje de consulta basado en TSQL2, ChronoLog y Bitemporal ChronoLog.

TimeDB es un DBMS Bitemporal basado en SQL.

Soporta un lenguaje de consulta, un lenguaje de manipulación de datos y un lenguaje de definición de datos.

TimeDB brinda ATSQL2, un lenguaje de consulta basado en TSQL2, ChronoLog y Bitemporal ChronoLog.

Manipula los datos temporales buscando extender el modelo de datos relacional a uno temporal.

TimeDB es un DBMS Bitemporal basado en SQL.

Soporta un lenguaje de consulta, un lenguaje de manipulación de datos y un lenguaje de definición de datos.

TimeDB brinda ATSQL2, un lenguaje de consulta basado en TSQL2, ChronoLog y Bitemporal ChronoLog.

Manipula los datos temporales buscando extender el modelo de datos relacional a uno temporal.

Traduce sentencias TSQL en sentencias SQL estándar para luego ser ejecutadas en DBMS como Oracle, Sybase, etcétera.

TimeDB: TDDL

En TimeDB, una tabla bitemporal puede crearse de esta manera:

En TimeDB, una tabla bitemporal puede crearse de esta manera:

```
CREATE TABLE empleados (EmpID INTEGER, Name  
CHAR(30), Department CHAR(40),  
Salary INTEGER)  
AS VALIDTIME AND TRANSACTIONTIME;
```

TimeDB: TDML

Las siguientes sentencias insertan datos temporales a una tabla:

Las siguientes sentencias insertan datos temporales a una tabla:

```
VALIDTIME PERIOD '1985-1990'  
INSERT INTO empleados VALUES  
(10,'John','Research',11000);
```

```
VALIDTIME PERIOD '1990-1993'  
INSERT INTO empleados VALUES (10,'John','Sales',11000);
```

```
VALIDTIME PERIOD '1993-forever'  
INSERT INTO empleados VALUES (10,'John','Sales',12000);
```

TimeDB: TQL

Para hacer consultas:

Para hacer consultas:

VALIDTIME

```
SELECT * FROM empleados;
```

TRANSACTIONTIME

```
SELECT * FROM empleados;
```

VALIDTIME AND TRANSACTIONTIME

```
SELECT * FROM empleados;
```

Oracle Workspace Manager (1)

Oracle Workspace Manager (1)

Es una herramienta de Oracle Database que brinda a los desarrolladores la capacidad de manejar distintas versiones temporales de la base de datos.

Oracle Workspace Manager (1)

Es una herramienta de Oracle Database que brinda a los desarrolladores la capacidad de manejar distintas versiones temporales de la base de datos.

Se puede habilitar el soporte de tiempo válido al momento de creación de una tabla:

Oracle Workspace Manager (1)

Es una herramienta de Oracle Database que brinda a los desarrolladores la capacidad de manejar distintas versiones temporales de la base de datos.

Se puede habilitar el soporte de tiempo válido al momento de creación de una tabla:

- ▶ Crear la tabla de empleados y sus salarios:

```
CREATE TABLE empleados (  
    name VARCHAR(16) PRIMARY KEY,  
    salary NUMBER  
);
```


Oracle Workspace Manager (1)

Es una herramienta de Oracle Database que brinda a los desarrolladores la capacidad de manejar distintas versiones temporales de la base de datos.

Se puede habilitar el soporte de tiempo válido al momento de creación de una tabla:

- ▶ Crear la tabla de empleados y sus salarios:

```
CREATE TABLE empleados (  
    name VARCHAR(16) PRIMARY KEY,  
    salary NUMBER  
);
```
- ▶ Versionar la tabla. Especificar TRUE para el soporte de tiempo válido:

```
EXECUTE DBMS_WM.EnableVersioning('empleados','VIEW_WO_OVERWRITE',FALSE,TRUE);
```

Oracle Workspace Manager (1)

Es una herramienta de Oracle Database que brinda a los desarrolladores la capacidad de manejar distintas versiones temporales de la base de datos.

Se puede habilitar el soporte de tiempo válido al momento de creación de una tabla:

- ▶ Crear la tabla de empleados y sus salarios:

```
CREATE TABLE empleados (  
    name VARCHAR(16) PRIMARY KEY,  
    salary NUMBER  
);
```
- ▶ Versionar la tabla. Especificar TRUE para el soporte de tiempo válido:

```
EXECUTE DBMS_WM.EnableVersioning('empleados','VIEW_WO_OVERWRITE',FALSE,TRUE);
```
- ▶ Insertar las filas:

```
INSERT INTO empleados VALUES(  
    'Adams',  
    30000,  
    WMSYS.WM.PERIOD(TO_DATE('01-01-1990','MM-DD-YYYY'),  
        TO_DATE('01-01-2005','MM-DD-YYYY'))
```

Oracle Workspace Manager (2)

Oracle Workspace Manager (2)

El tipo de datos WM_PERIOD es usado para especificar el rango del tiempo válido.

Oracle Workspace Manager (2)

El tipo de datos WM_PERIOD es usado para especificar el rango del tiempo válido.

Las constantes del tiempo válido son:

DBMS_WM.MIN_TIME \approx 01-Jan-(-4712)

DBMS_WM.MAX_TIME \approx 31-Dec-9999

DBMS_WM.UNTIL_CHANGED es un TIMESTAMP que se comporta como MAX_TIME hasta que es modificado.

Oracle Workspace Manager (2)

El tipo de datos WM_PERIOD es usado para especificar el rango del tiempo válido.

Las constantes del tiempo válido son:

DBMS_WM.MIN_TIME \approx 01-Jan-(-4712)

DBMS_WM.MAX_TIME \approx 31-Dec-9999

DBMS_WM.UNTIL_CHANGED es un TIMESTAMP que se comporta como MAX_TIME hasta que es modificado.

Algunos operadores:

WM_OVERLAPS, WM_CONTAINS, WM_MEETS,
WM_EQUALS, WM_INTERSECTION, etcétera.

- ▶ Tópicos Avanzados de Bases de Datos. Cristina Bender, Claudia Deco.
- ▶ Temporal Databases. Richard T. Snodgrass.
- ▶ Temporal Query Languages: a Survey. Jan Chomicki.
- ▶ Temporal features in SQL:2011. Krishna Kulknarni.
- ▶ Extensions to SQL for Historical Databases. L. Sarda.
- ▶ Introduction to Temporal Database Research. Christian S. Jensen.