

BASES DE DATOS TEMPORALES

Una **BASE DE DATOS TEMPORAL** es aquella que contiene datos históricos en vez (o además) de datos actuales.

Docentes : Ing. Fernando Sato

(C.J. Date).

A.S. Sebastian Trossero

Versión: 20190424

RESUMEN

- Motivación y origen BD Temporales
- Definición
- Características
- Tipos:
 - Tiempo Valido
 - Tiempo Transaccional
 - Bitemporales
- Caso de Estudio
- Algebra Temporal
- Implementaciones

Bases de Datos Temporales

Motivación

→ Problema

Tenemos una relación como la siguiente:

Empleado (dni, nombre, fec_nac, sueldo, titulo)

Mas alla del dominio de cada dato,

¿Que clasificación podemos hacer en cuanto a su variación en el tiempo?

¿Que aspectos motivan la variación?



Bases de Datos Temporales

Motivación → Análisis

Tenemos una relación como la siguiente:

Empleado

Atributos Invariantes:

dni, nombre???, fec_nac

Atributos Variantes con el tiempo:

sueldo, titulo



Bases de Datos Temporales

Motivación

→ Problema

Una base de datos convencional (NO TEMPORAL), tiene solamente datos actuales, en general pierden los datos historicos:

Empleado (dni, nombre, fec_nac, sueldo, titulo)

¿Como plantearía esta relación para llevar la historia de salarios y de titulos?



Bases de Datos Temporales

Motivación

→ Solución

Situación de Partida:

Empleado (dni, nombre, fec_nac, sueldo, titulo)

Solución

Agregamos las fechas de inicio de cada novedad y su variación como una relación repetitiva:

**Empleado (dni, nombre, fec_nac,
 {desde, sueldo},
 {fecha_graduacion,titulo}
)**

Bases de Datos Temporales

Motivación

→ Solución

Situación de Partida:

Empleado (dni, nombre, fec_nac, sueldo, titulo)

Solución

Normalizamos:

Empleado (dni, nombre, fec_nac)

EmpleadoSueldos (dni, desde, hasta, sueldo)

EmpleadoTitulos(dni, fechagraduacion, titulo)

Bases de Datos Temporales

Motivación → Problema

Nueva Relación Empleado y Relaciones Secundarias:

Empleado (dni, nombre, fec_nac)

EmpleadoSueldos (dni, desde, hasta, sueldo)

EmpleadoTitulos(dni, fechagraduacion, titulo)

¿Como obtenemos ahora el sueldo actual
de la persona cuyo dni es 17044664?



Bases de Datos Temporales

Motivación → Problema

Nueva Relación Empleado y Relaciones Secundarias:

Empleado (dni, nombre, fec_nac)

EmpleadoSueldos (dni, desde, hastya, sueldo)

EmpleadoTitulos(dni, fechagrgraduacion, titulo)

Desde la Situación de Partida es facil:



Empleado (dni, nombre, fec_nac, sueldo, titulo)

```
Select sueldo from Empleado where dni = 17044664;
```

Bases de Datos Temporales

Motivación

→ Solución

Nueva Relación Empleado y Relaciones Secundarias:

Empleado (dni, nombre, fec_nac)

EmpleadoSueldos (dni, desde, hasta, sueldo)

EmpleadoTitulos(dni,fechagraduacion,titulo)

Que solución Estructural plantearíamos?



Que solución con sql basico?

Que objeto podría encapsular esta ultima solución?

Bases de Datos Temporales

Motivación

→ Solución

Nueva Relación Empleado y Relaciones Secundarias:

Empleado (dni, nombre, fec_nac, **sueldoactual, tit..)**

EmpleadoSueldos (dni, desde, sueldo)

EmpleadoTitulos(dni,fechagraduacion,titulo)

Que solución Estructural plantearíamos?

Información redundante controlada por regla activa (sueldoactual).



Que solución con sql basico?

Y con sql?

Que objeto utilizaría para esta ultima solución?

Bases de Datos Temporales

Motivación

→ Solución

Nueva Relación Empleado y Relaciones Secundarias:

Empleado (dni, nombre, fec_nac)

EmpleadoSueldos (dni, desde, hasta, sueldo)

EmpleadoTitulos(dni,fechagraduacion,titulo)

Que solución Estructural plantearíamos?

Información redundante controlada por regla activa
(sueldoactual).

Que solución con sql basico?

```
Select e.dni, nombre, fec_nac, s.sueldo
from empleado e
join empleadosueldos s on e.dni = s.dni
where desde <= current_date and hasta >= current_date
```



¿Funciona?

Bases de Datos Temporales

Motivación

→ Solución

Nueva Relación Empleado y Relaciones Secundarias:

Empleado (dni, nombre, fec_nac)

EmpleadoSueldos (dni, desde, hasta, sueldo)

EmpleadoTitulos(dni,fechagraduacion,titulo)

Que solución Estructural plantearíamos?

Redundancia controlada por regla activa

Que solución con sql basico?

```
Select e.dni, nombre, fec_nac, s.sueldo
from empleado e
join empleadosueldos s on e.dni = s.dni
where desde <= current_date and
       coalesce(hasta,current_date) >= current_date
```



¡Funcional!

Bases de Datos Temporales

Motivación

El elemento tiempo en una base de datos avanzada requiere un tratamiento especial, pensemos en preguntas del tipo:

¿Que sueldo tuvo Juan desde octubre de 2005 a marzo de 2006?

¿Cual fue el sueldo inicial de Juan?

dni	nombre	fec_nac	desde	hasta	sueldo
17044664	Juan	27/06/1984	16/03/2003	22/11/2005	6000.00
17044664	Juan	27/06/1984	23/11/2005	NULL	85000.00
24333121	Maria	02/08/1993	12/08/2010	NULL	6000.00

¿Que evolución a tenido su sueldo año por año, o por periodo de cambio?

Bases de Datos Temporales

Introducción → Aplicaciones

Existen muchos dominios en los que se necesita tratamiento temporal para mantener sus datos:

Seguros: Vigencias de Polizas, Pagos de Poliza, Administración de Siniestros, etc.

Salud: Historias Clínicas.

Registros Civiles: Identificación de personas, Estado Civil, etc?

Universidades-Facultades: Planes de Estudio, Estado académico de alumnos, etc.

En la realidad en todo sistema encontramos dominios que merecen un diseño y tratamiento especial.

Bases de Datos Temporales

Definición

Una Base de Datos Temporal gestiona, implementa y trata con especial énfasis los aspectos temporales, teniendo un modelo de datos temporal.

Se espera contar con una versión temporal de SQL que permita facilitar estas gestiones.

Hablando en terminos coloquiales:

"Es aquella que almacena y trata Datos Historicos y Actuales."

Ampliando el problema pensemos:

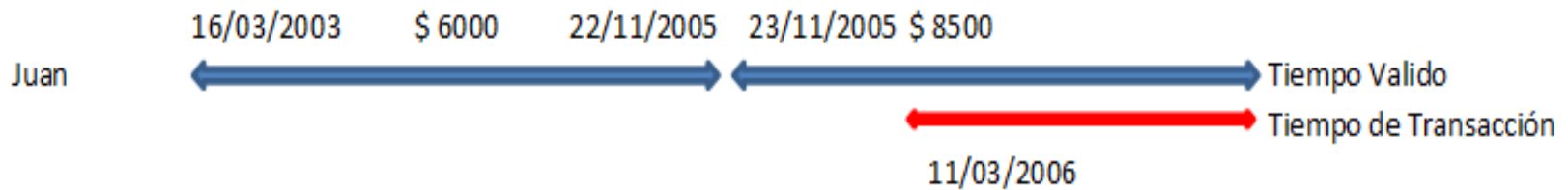
- *Que sucede si el aumento de Juan del 23/11/2005 es producto de una negociación que se cierra definitivamente el 11-03-2006?*

Bases de Datos Temporales

Dimensiones Temporales

Representando esto en un eje de tiempo:

Que sucede si el aumento de Juan del 23/11/2005 es producto de una negociación que se cierra definitivamente el 11-03-2006?



Es decir además del concepto periodo valido de un hecho en el mundo real, puede ser "no conocido" de acuerdo al tiempo de registro de la transacción, esta sutil diferencia define las dos dimensiones temporales:

- Tiempo de **Validez** o Valido de un hecho.
- Tiempo de **Transacción**.

Bases de Datos Temporales

Tiempo Valido y Tiempo de Transacción

Definiciones:

- ♦ **TIEMPO DE VALIDEZ** de un hecho: Es el periodo en que el hecho es cierto en el mundo real.
- ♦ **TIEMPO DE TRANSACCION** de un hecho: Cuando este se registra en la base de datos, o podemos pensarlo cuando la base de datos lo reconoce.

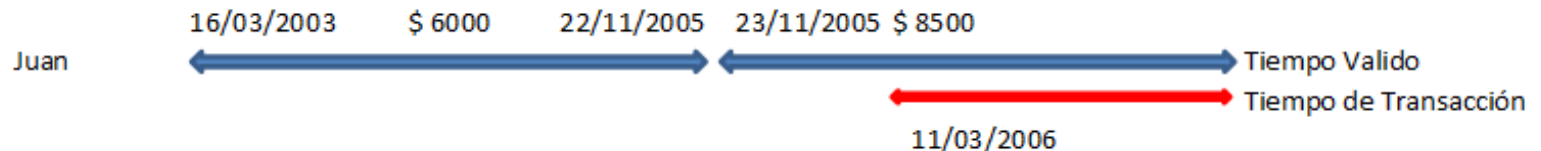
El aumento de Juan del **23/11/2005** es producto de una negociación que se cierra definitivamente y se modifica el **11-03-2006**.

Bases de Datos Temporales

Introducción

Elementos de Tiempo en Bases de Datos: Estos posibilitan consultas tales como ...

- **Simples:** ¿Que paso en ese momento?
Ej *Cual era el salario de Juan cuando ingreso?*



- **Complejas:** ¿Lo que ha sucedido desde entonces hasta ahora?
Ej *Que Trabajadores recibieron aumento el ultimo año?*

Bases de Datos Temporales

Concepto Granularidad

Analicemos las siguientes relaciones:

Empleado (dni, nombre, estadoCivil, fechaNacimiento, HorarioEntrada)

InscripcionAlumno (libUniversitaria, fechaInscripcion, anioLectivo)

- Cuales son los datos que requieren tratamiento temporal?

Bases de Datos Temporales

Concepto Granularidad

Analicemos las siguiente relaciones:

Empleado (dni, nombre, estadoCivil, fechaNacimiento, horarioEntrada)

InscripcionAlumno (libUniversitaria, fechaInscripcion, anioLectivo)

Cuales son los datos que requieren tratamiento temporal?

Que representación plantearía para **horarioEntrada**?

Bases de Datos Temporales

Concepto Granularidad

Granularidad Temporal:

La **granularidad** temporal expresa una restricción en la cual la máxima frecuencia de actualización de un atributo es la ***granularidad elegida***.

La granularidad temporal es una medida de tiempo que puede ser expresada en:

Años, Meses, Días, Horas, Minutos, Segundos, etc.

Nota: La consecuencia de elegir una granularidad (la mínima posible) es que dos eventos que tengan lugar con el mismo valor se consideraran simultáneos, aunque realmente no lo sean.

Bases de Datos Temporales

Concepto Granularidad

Mas Sobre Granularidad Temporal: Relaciones

A su vez, dado un conjunto de granularidades existen unas relaciones de orden o equivalencia entre ellas, por ejemplo, la granularidad semanas es más fina que la granularidad meses y menos fina que la granularidad días.

Estas relaciones de orden entre diferentes granularidades deben ser especificadas en la base de datos. Por otro lado, la granularidad de la información se puede ajustar, moviéndose de algo más fino (fino) a datos más grueso (coarser), a esto se le conoce como **rollup**. Si ocurre lo opuesto, ir de algo grueso a algo más fino ocurre un **drill down**.

Bases de Datos Temporales

Concepto Granularidad

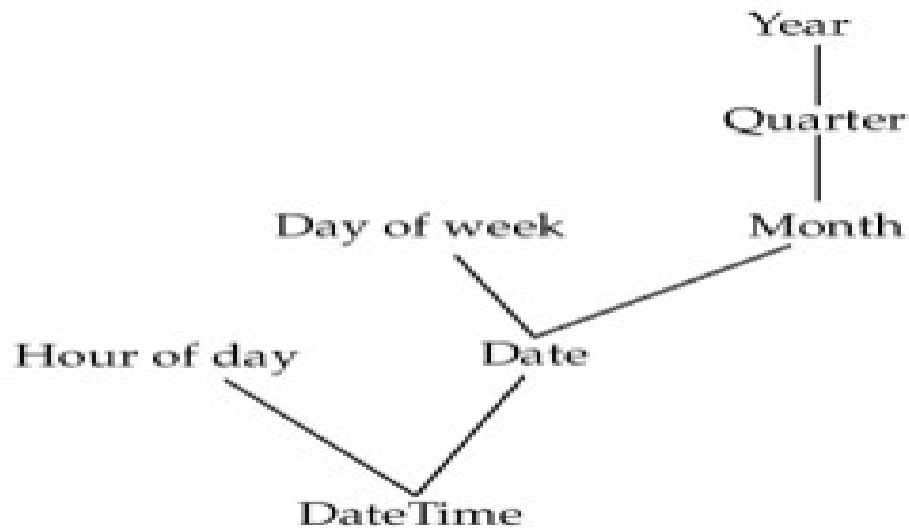
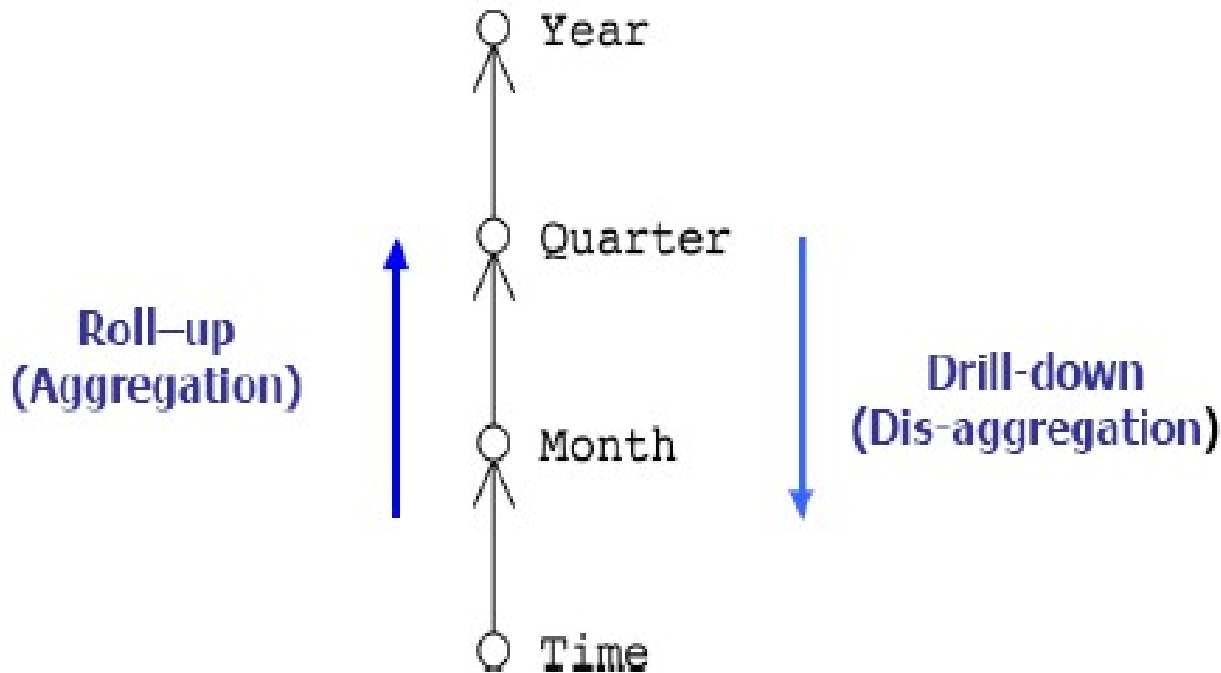


Figura .: Equivalencia de granularidades

Bases de Datos Temporales

Concepto Granularidad

- **Roll-up:** generalización y agregación.
- **Drill-down:** especialización y desagregación.



Bases de Datos Temporales

Espacio de Discusión Granularidad

- **Caso alumnos:** Se desea modelar una tabla con las siguientes relaciones:

Lu	nombre	sexo
Nacio	ingreso	abandono
SeGraduo	promedio	

- Se pide:
 - 1) Clasificar los atributos de acuerdo a su relación con la temporalidad.
 - 2) Determinar el tipo de datos propuesto para cada uno.
 - 3) Determinar las restricciones de cada atributo (cconstraint y tconstraint).
 - 4) Genera DDL necesaria.
 - 5) Plantear una nueva relación que contenga además de la información de base los siguientes roll-up: año de graduación, nro de trimestre en el que abandono.
 - 6) Que usaría para generalizar el concepto de trimestre?

Bases de Datos Temporales

Espacio de Discusión Granularidad

- **Caso** alumnos: Se desea modelar una tabla con las siguientes relaciones:

Lu	nombre	sexo
Nacio	ingreso	abandono
SeGraduo	promedio	

- Se pide:
 - 1) Clasificar los atributos de acuerdo a su relación con la temporalidad.
 - 2) Determinar el tipo de datos propuesto para cada uno.
 - 3) Determinar las restricciones de cada atributo (cconstraint y tconstraint).
 - 4) Genera DDL necesaria.
 - 5) Plantear una relación tabla derivada (subselect en from) que contenga además de la información de base los siguientes roll-up: año de graduación, nro de trimestre en el que abandono.
 - 6) Que usaría para generalizar el concepto de trimestre?
 - 7) Genere una función denominada nrotrimestre que reciba como parámetro una fecha genere un entero con el nro de trimestre.
 - 8) Genere otra función denominada trimestre pero que genere un entero con el año y el trimestre, ej (input '2017/05/02' output 20172).

Bases de Datos Temporales

Concepto Granularidad - Chronon

Chronon:

Se define como la duración de tiempo más corta, al instante temporal más breve, soportado por el DBMS.

Esta unidad de tiempo indivisible generalmente es el *milisegundo*. Esta magnitud resulta suficiente para la mayoría de los dominios de aplicaciones administrativas.

□ Ejemplos:

- para el dominio de un hospital, el registro de los nacimientos se requiere un chrono de minutos,
- para una carrera de formula uno, se requiere un chrono de milisegundos, Ej tiempo 1:32:16.019.

Bases de Datos Temporales

Concepto Granularidad - Chronon

Chronon:

Ejemplo Critico: _para un dominio de formula uno, se requiere un chrono de milisengundos.

Ej tiempo 1:22:16.019.

—

Live Timing			
P	#	NAME	BEST
1	44	L.HAMILTON	1:22.188
2	5	S.VETTEL	1:22.456
3	77	V.BOTTAS	1:22.481
4	7	K.RAIKKONEN	1:23.033
5	33	M.V	
6	8	R.G	
7	19	F.M	
8	55	C.S	
9	26	D.K	
10	3	D.R	



Bases de Datos Temporales

Concepto Tiempo – Sql

Podemos plantear las siguientes equivalencias con datos primitivos de SQL no temporal.

Granularidad	Tipo de Dato	Descripción
Dia	Date	Fecha
Instante Absoluto	TimeStamp o DateTime	Fecha, Hora, Minuto, Segundo y Milesima Seg
Instante Relativo	Time	Hora, Minuto, Segundo y Milesima Seg

- Nota: El resto de los datos se pueden tratar con otros tipos primitivos sql92, definiendo dominios. Ejemplo:

```
Create Domain danio as integer check(value > 0);
```

Bases de Datos Temporales

Concepto Tiempo - Tsql

Podemos plantear los siguientes tipos de datos para una versión de Sql Temporal.

Tipo	Descripción
Time	Hora con fracciones de segundo [H:M:S:mS]
Date	Fecha [Y:M:D]
TimeStamp	Instante de Tiempo, Fecha + Hora
Interval	Duración de tiempo relativa, como 10 días o 25 minutos
Period	Periodo, duración de tiempo absoluta. TimeStam + Interval.

□ Ejemplo de periodos:

- Periodo de Vacaciones 2016 ['2017/01/02' – '2017/01/16')
 - Significa desde el 2 de enero incluido hasta 16 enero (excluido).

Bases de Datos Temporales

Tipos de Datos

Existen 3 tipos de datos en una base de datos temporal:

Datos Estaticos: Este tipo de datos es definido como una constante sobre todo el universo de tiempo. Es decir, se considera eterno en contraste con el temporal el cual es valido para un intervalo o periodo especifico de tiempo.

Datos Temporales: Es el tipo de dato para casos que varían en el tiempo, datos dinámicos.

Datos Instantaneos: *Un valor para este tipo de datos es valido solamente para el instante actual. Se aplica a datos estaticos y temporales. Edad de una persona, interes ganado en un plazo fijo, valor de una acción.*

Bases de Datos Temporales

Tipos de Dato TEMPORAL

El tipo de **Dato Temporal** es el más importante de la base de datos temporal.

El tipo de **Dato Temporal** es “una unión finita de intervalos”

Este elemento es el que caracteriza la tecnología que tiene un gestor de base de datos temporal específico para construir relaciones temporales.

Shashi Adia y Sunil S. Nair

Bases de Datos Temporales

Tipos de Información Temporal

Una BD temporal guarda información estática y dinámica sobre cuando se producen determinados eventos o cuando se consideran determinados hechos válidos. En gral estos hechos se clasifican en:

Hechos de Punto: Se utiliza en todos los hechos que en el mundo real se consideran instante de unidad(grano). Esta granularidad puede ser día (date) o instante (timestamp) e incluso año, década, siglo, etc.

Hechos de Duración: Se utiliza en todos los hechos que en el mundo real se consideran intervalos de tiempo con inicio. Su definición consiste de un punto (Instante, fecha, etc.) de inicio y uno de fin; el inicio y fin se construyen sobre una misma escala (granularidad o grano).

Bases de Datos Temporales

Tipos de Información Temporal

¿Que pasa con la Implementación de cada uno de ellos?

Hechos de Punto: Se implementan con variables convencionales de tiempo de las bases de datos relacionales: timestamp, date, semana, mes, trimestre, año, etc.

Hechos de Duración: (Periodos) Se implementan usando atributos desde hasta, es decir, 2 variables convencionales. Ej Suplencia en el cargo tesorero (01-07-2011 , 20-3-2015).

¿Y Si el Periodo es abierto?,

¿es decir se cumple hasta ahora?



Bases de Datos Temporales

Tipos de Información Temporal

¿Que pasa con la Implementación de cada uno de ellos?

En el modelado se ha usado:

(∞ , forever, uc: until changed) para representar esta situación.

Tratamiento con BD no temporal: Utilizan la constante figurativa **NULL**. ¿Ventajas o Desventajas?

Tratamiento BD temporales: No Utilizan
Una representación específica.

Solución
a periodos
abiertos

Results		Messages				
	dni	nombre	fecha_nacimiento	sueldo	ttinicio	ttfin
1	10	Juan	1999-06-02	70000.00	2017-05-12 11:42:25.1464953	9999-12-31 23:59:59.9999999

Bases de Datos Temporales

Tipos de Información Temporal

Dada la siguiente relación determine que atributos son del tipo hechos de punto, de duración y cuales son atemporales y defina su granularidad.

Para el caso de los atributos temporales definir cuales son implementados como instantaneos. ¿y analice si algún atributo temporal admite periodo abierto?

`AutoAlquiler (patente, marca, modelo,
aniofabricacion, estado , dueños, dueño_actual,
precioalquiler)`

`Estado asociado al dominio`

`('libre', 'alquilado', 'fuera de servicio')`

Bases de Datos Temporales

Tipos de Relaciones-según R SNODGRASS

□ Relaciones SNAPSHOT

Las bases de datos SNAPSHOT no contemplan ni el tiempo de validez ni de transacción, no almacenan ninguna de estas referencias temporales en el modelo que emplean .

□ Relaciones TIEMPO DE TRANSACC.

□ Relaciones TIEMPO VALIDO

□ Relaciones BI – TEMPORALES

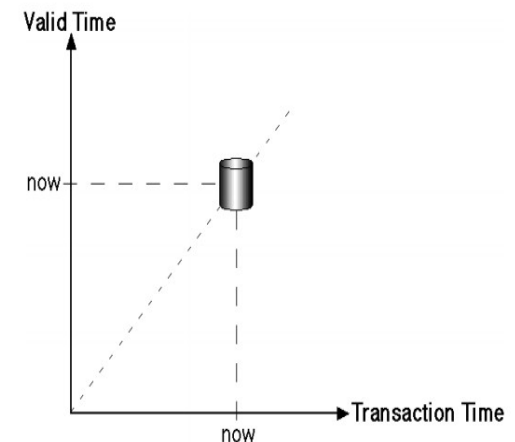


Figura 3: Snapshot Databases

Bases de Datos Temporales

Tipos de Relaciones-según R SNODGRASS

- Relaciones SNAPSHOT
- Relaciones TIEMPO DE TRANSACCION(rollback)

Las bases de datos ROLLBACK, en cambio, contienen unicamente tiempo de transacción, nos permiten devolver la base de datos a un estado anterior en caso que sea necesario, pero no permiten conocer en que periodo de tiempo fue valida la información en el mundo real. Ejemplo tecnología FlashBack de ORACLE.

- Relaciones TIEMPO VALIDO (históricas)
- Relaciones BI – TEMPORALES

Bases de Datos Temporales

Tipos de Relaciones-según R SNODGRASS

- Relaciones SNAPSHOT
- Relaciones TIEMPO DE TRANSACCION
- Relaciones TIEMPO VALIDO (históricas)

Las bases de datos HISTORICAS contemplan unicamente el tiempo de validez, almacenan la información que conocemos como valida, tanto presente como pasado o futura, pero los cambios producidos en la información (como ha sido la evolución de las actualizaciones realizadas) no quedan almacenados.

- Relaciones BI – TEMPORALES

Bases de Datos Temporales

Tipos de Relaciones-según R SNODGRASS

- Relaciones SNAPSHOT
- Relaciones TIEMPO DE TRANSACCIÓN
- Relaciones TIEMPO VALIDO

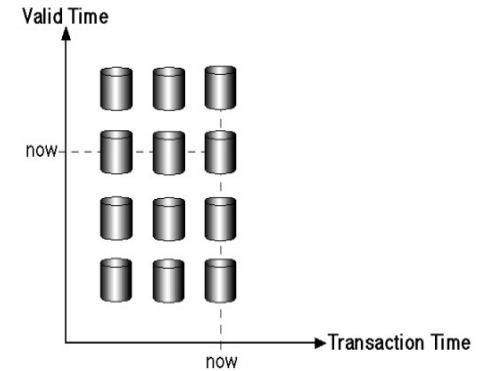


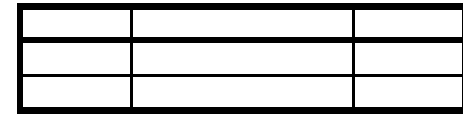
Figura 4: Base de Datos Bitemporal

- Relaciones **BI-TEMPORALES**: Las bases de datos BI-TEMPORALES, soportan tanto el tiempo de transacción como el de validez, por lo que nos dan una visión más precisa de la evolución que ha sufrido la información tanto sobre sus periodos de validez como de sus actualizaciones, al contemplar los diferentes estados por los que pasó la información, y permitir su recuperación.

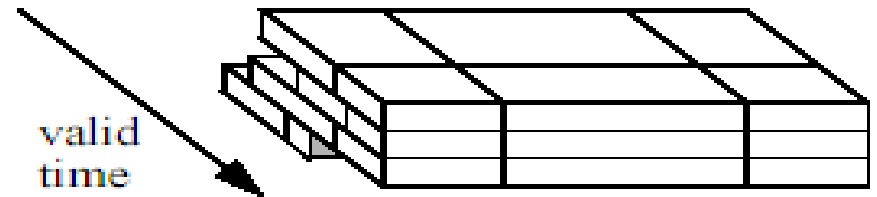
Bases de Datos Temporales

Relaciones Temporales

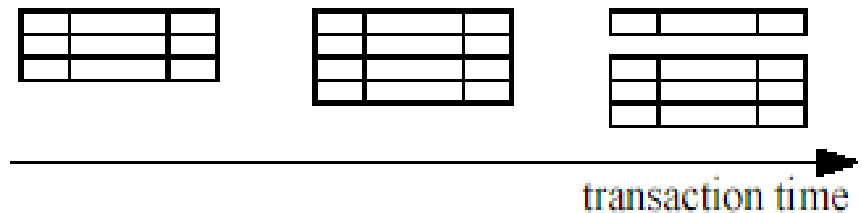
- Relaciones Snapshot



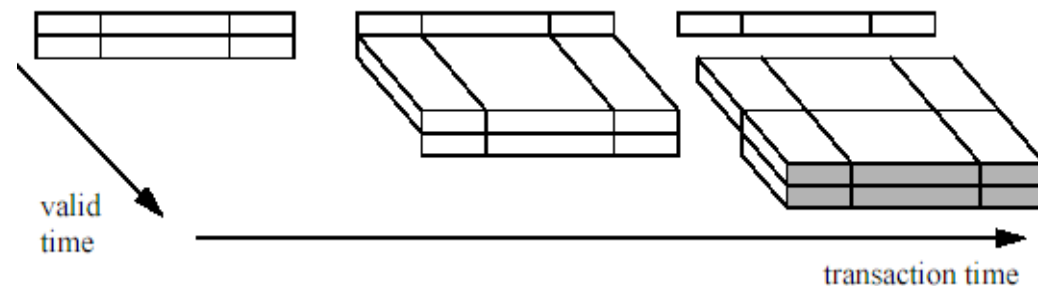
- Rels Tiempo Valido



- Rels Tiempo Transacción



- Rels Bitemporales



Bases de Datos Temporales

Tipos de Relaciones-según R SNODGRASS

Una tabla en una Base de Datos Temporal puede ser:

- **Tabla snapshot** (almacenamiento de datos atemporales y/o instantaneos),
- o **tabla de tiempo valido**,
- o **tabla de tiempo de transacción**
- o una **tabla bitemporal**.

Bases de Datos Temporales

Motores Temporales

¿Que es una RDBMS temporal?

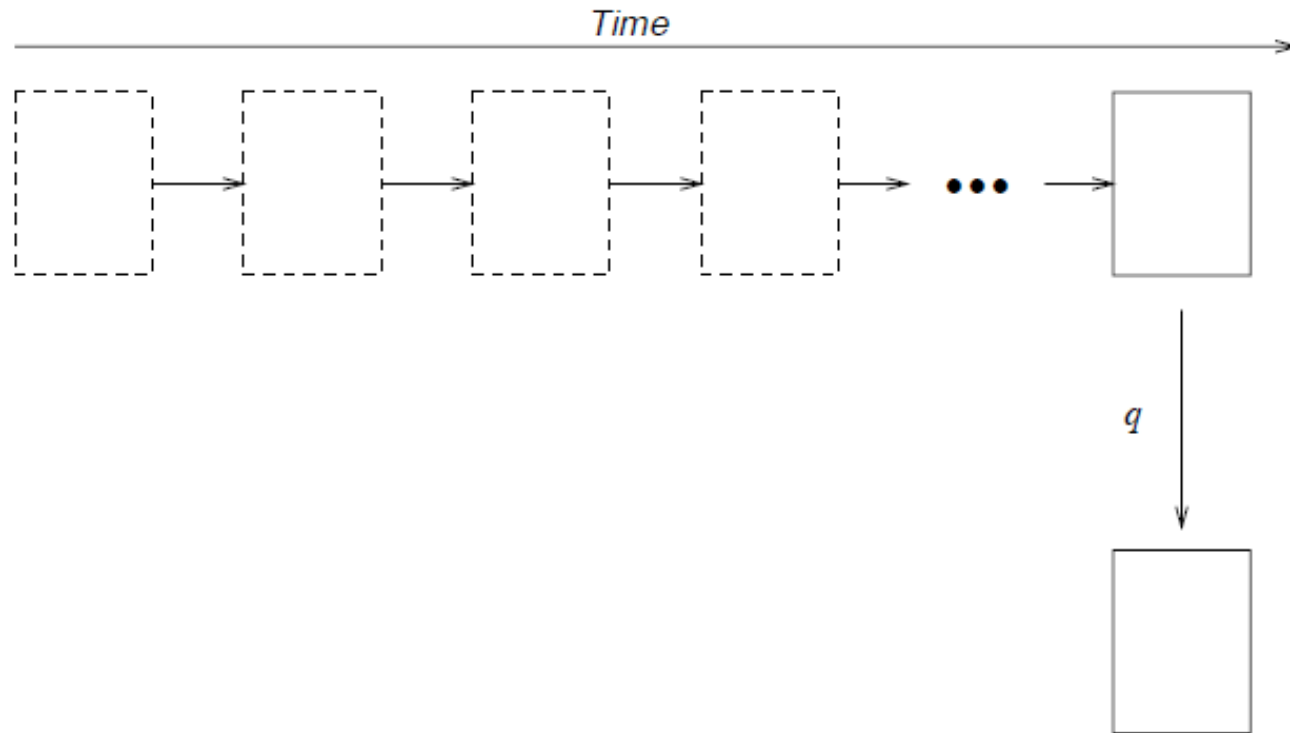
- Es un RDBMS que soporta ambos: TIEMPO VALIDO y TIEMPO TRANSACCION
- Dicho de otra forma: Es un RDBMS que permite gestionar Bases de Datos Bi-Temporales.

RESUMEN

- Motivación y origen BD Temporales
- Definición
- Características
- Tipos:
 - **Tiempo Valido**
 - Tiempo Transaccional
 - Bitemporales

Bases de Datos Temporales

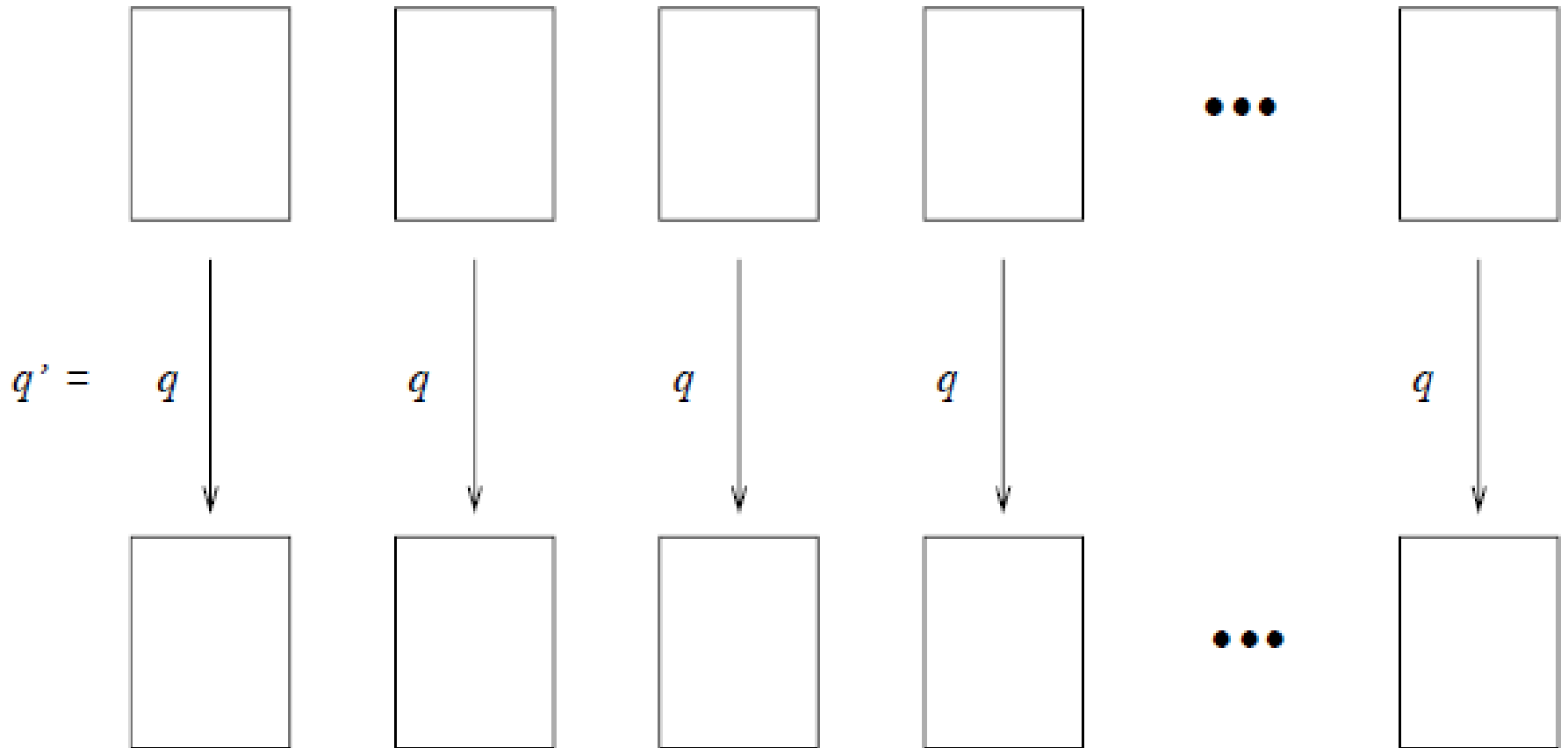
Relaciones SNAPSHOT o Instantanea



q : Es un query sql Standart. Salario de un empleado.

Bases de Datos Temporales

Relaciones Temporal (TIEMPO – VALIDO)



q' : query TSQL histórico

Por Ej: Trae la historia de los salarios mensuales pagados a los empleados.

Bases de Datos Temporales

Tiempo de Transacción

- Por que se necesitan estas relaciones?
 - Muchas veces las aplicaciones deben llevar un registro de todos los estados de la base de datos, generalmente por exigencias de auditoria o seguridad (FlashBack).
 - En estos casos no se permiten eliminar cambios de estados pasados.

En lugar de ello se utilizan transacciones de compensación.

Bases de Datos Temporales

Tiempo de Transacción

- *El **tiempo de transacción** siempre es menor o igual al instante actual.
NO puede ser FUTURO.*
- En Contraposición:
 - El **Tiempo válido** puede referirse a un futuro.
 - Ejemplo: Fecha prevista de vacaciones (próximo año) del 01-2-año proximo al 17-2-año proximo.

Bases de Datos Temporales

Tiempo de Transacción

- El tiempo evoluciona en forma discreta, en el mismo orden que el nro de transacción:
 - $T1 < T2 < T3 < T4 \dots$ ($<$ antes que)
- Una fila (F) se amplía con un periodo $[tt.start, tt.end]$.
- Cuando insertamos una fila en un instante $t1$ se actualizan los atributos temporales $\rightarrow [t1, uc]$.
 - El pasado **NO SE PUEDE CAMBIAR.**
 - Características (Rollback de DB)-volver la BD a un estado anterior.

Bases de Datos Temporales

Tiempo de Transacción

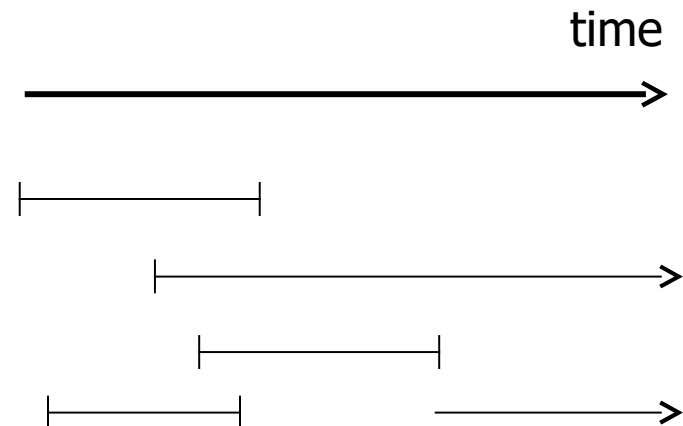
□ Los DELETE son Lógicos → Nunca Físicos

- Cuando una fila es deleteada en el tiempo t_2 , sus atributos temporales cambian de:

$[t_1, uc]$ a $[t_1, t_2]$ (tiempo de vida).

- Características: Una fila esta “viva” si el extremo del periodo tiene until changed o infinito. Ej: $[t_n, uc]$

eid	sueldo	t.start	t.end
10	20K	9/93	12/94
20	50K	4/94	uc
33	30K	5/94	6/95
10	50K	1/95	uc



Bases de Datos Temporales

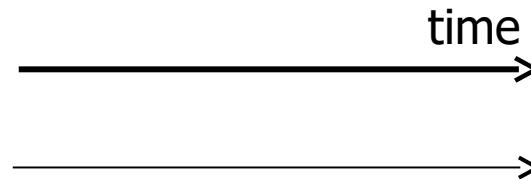
Tiempo de Transacción

□ Veamos los detalles

○ Con fecha 9/93 hacemos

→ *Insert into empleados values (10,20k) ;*

eid	sueldo	t.start	t.end
10	20K	09/93	uc



Bases de Datos Temporales

Tiempo de Transacción

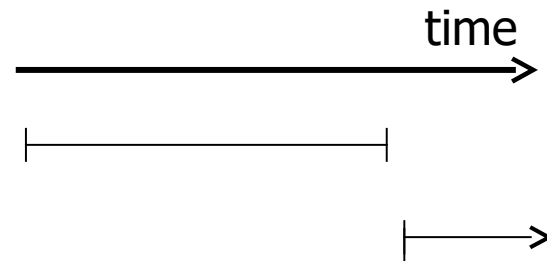
□ Luego

Insert into empleados values (10,20k);

○ Con fecha 1/95 hacemos

→ *Update empleados set sueldo = 50k where eid=10;*

eid	sueldo	t.start	t.end
10	20K	09/93	12/94
10	50K	01/95	uc



*Que resultado da: select * from empleados where eid=10?*

eid	sueldo
10	50K

Bases de Datos Temporales

Tiempo de Transacción

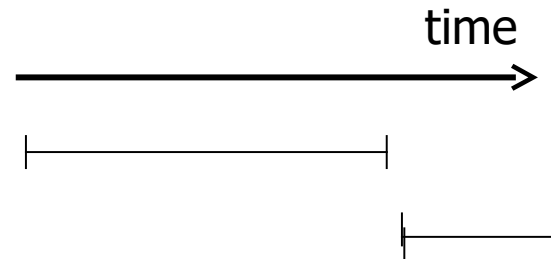
□ Luego

Insert into empleados values (10,20k) ;
update empleados set sueldo = 50k where eid = 10 ;

○ Con fecha 3/95 hacemos

→ *delete from empleados where eid = 10 ;*

eid	sueldo	t.start	t.end
10	20K	09/93	12/94
10	50K	01/95	03/95

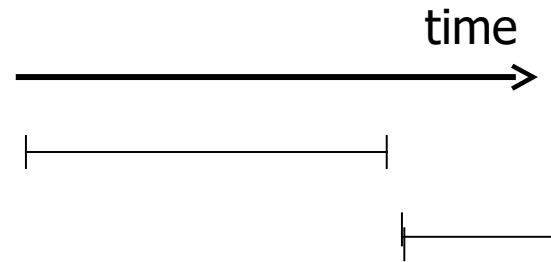


Bases de Datos Temporales

Tiempo de Transacción

□ Y ahora?

eid	sueldo	t.start	t.end
10	20K	09/93	12/94
10	50K	01/95	03/95



Que resultado da:

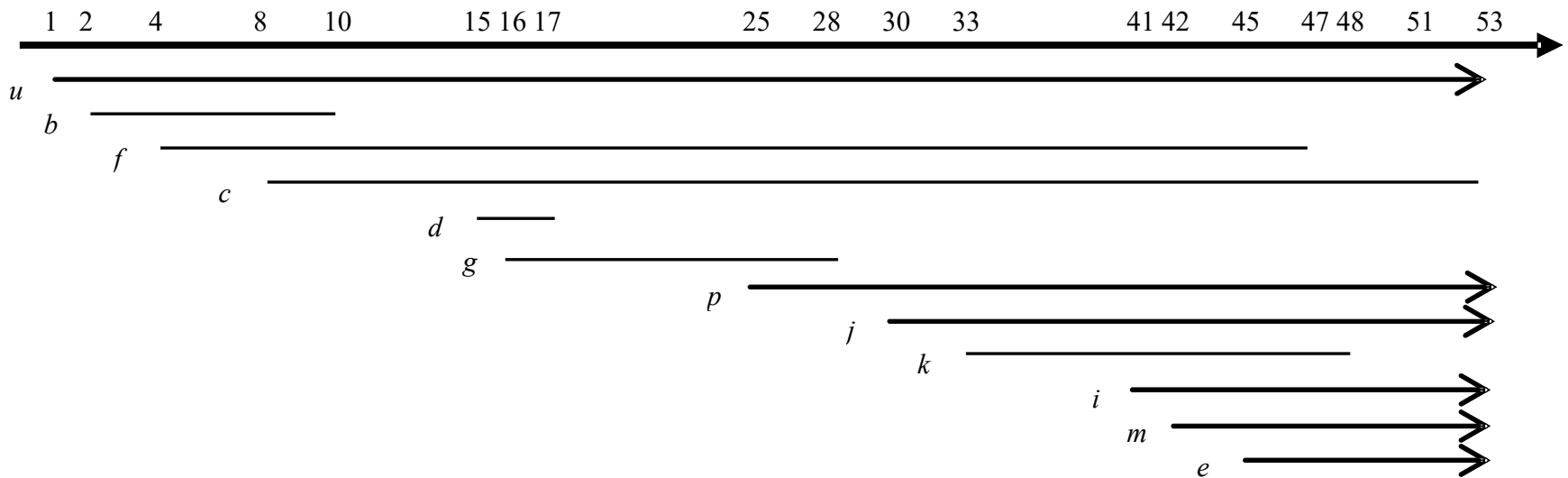
```
select * from empleados where eid=10?
```

eid	sueldo
-----	--------

Nota: Las filas “vigentes” son filas que contienen **UC** en el atributo de sistema de bdt t.end.

Bases de Datos Temporales

Tiempo de Transacción



Las Bases de datos evolucionan a través de inserciones (insert), modificaciones (update) y Eliminaciones (delete).

Nota: El tiempo transaccional "marcha" en forma monótona hacia adelante y se encuentra limitado por el instante de creación de la Base de Datos y el momento corriente.

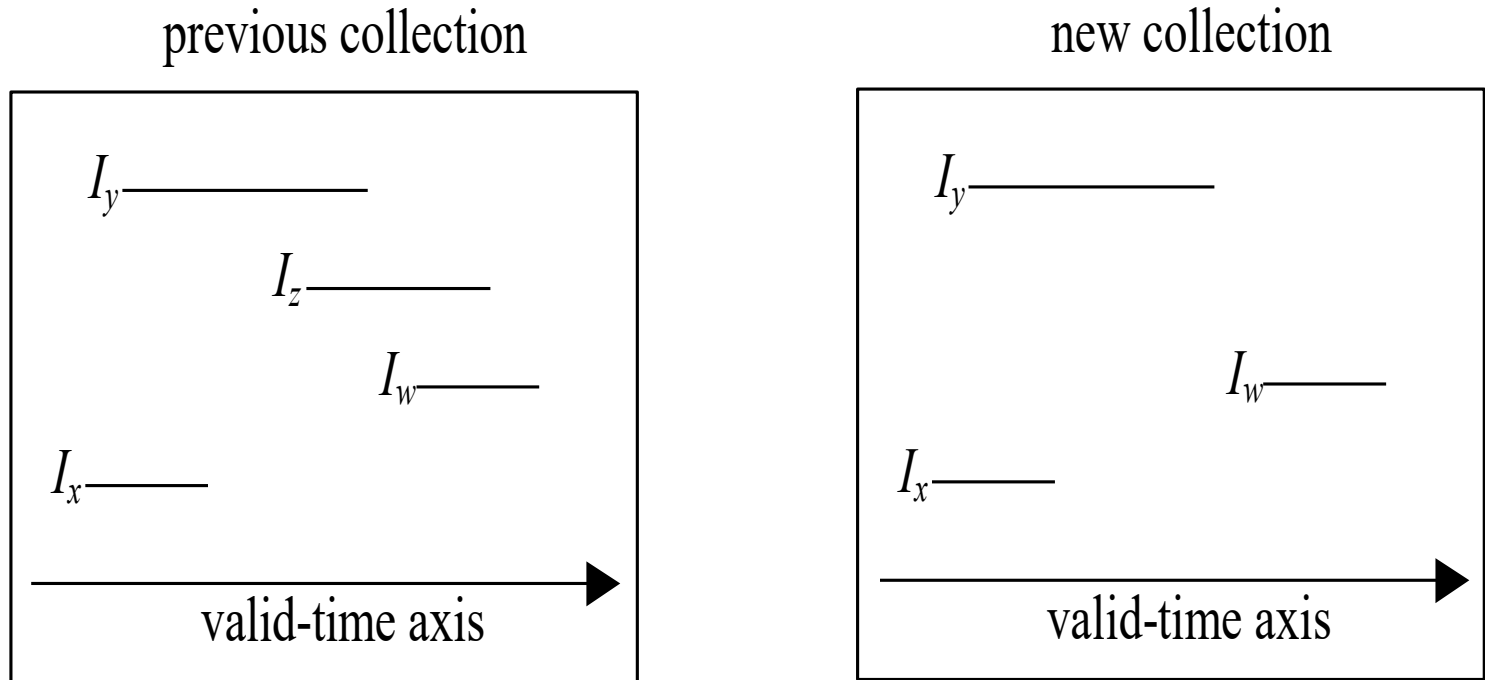
Bases de Datos Temporales

Tiempo de Validez

- El tiempo evoluciona continuamente.
- Cada objeto tiene asociado su periodo de tiempo de validez (Ej. Vigencia de una tarjeta de credito)
- Soporte total sobre Periodos de datos para las operaciones:
 - Delete
 - Insert
 - Update

Bases de Datos Temporales

Tiempo de Validez



- Las actualizaciones (nueva colección) agregan, quitan o modificandos generando información historica.

Bases de Datos Temporales

Tiempo de Validez

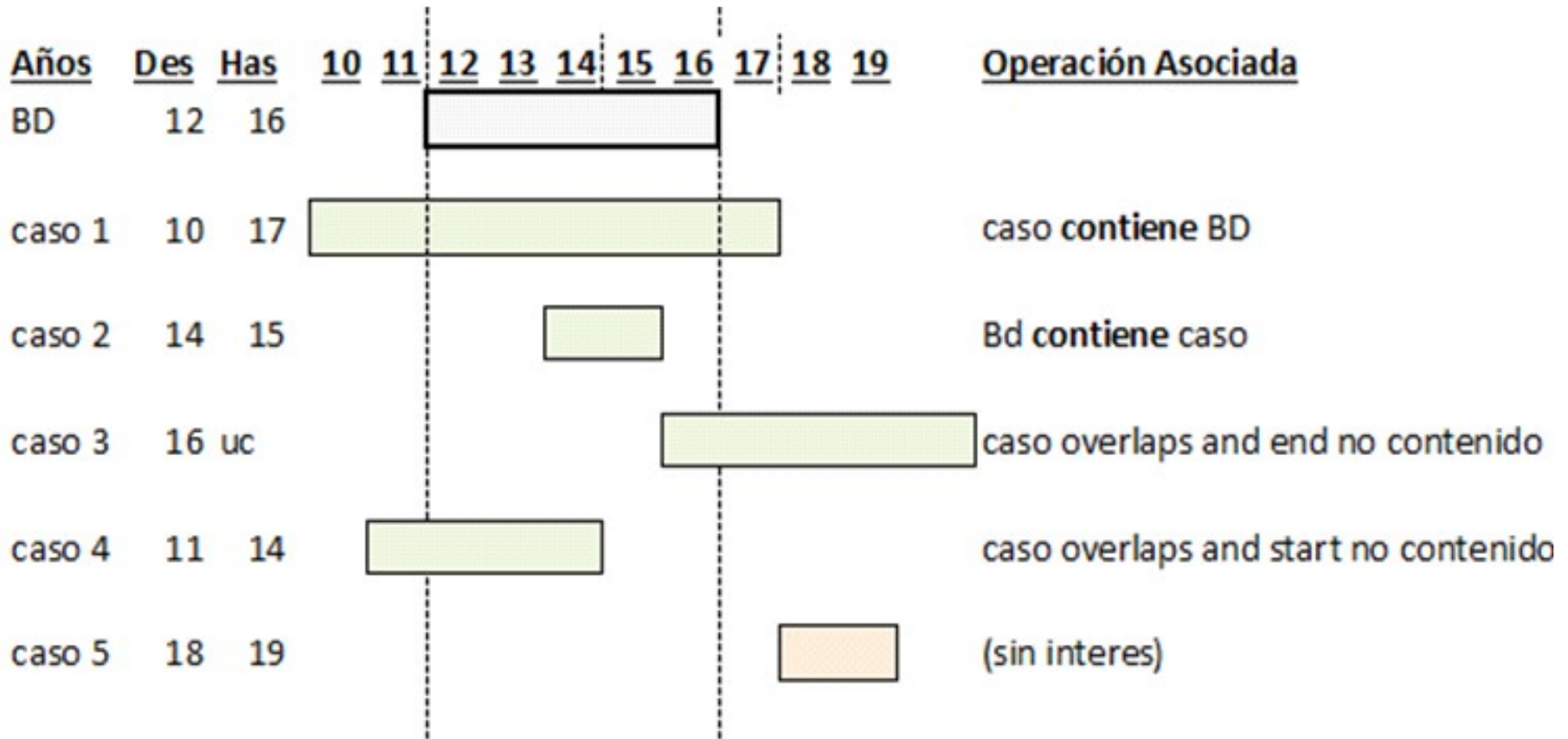
- Los **Delete y Update** son **físicos**:
 - Con lo cual no hay manera de saber los estados en periodos anteriores.

Que Situaciones puede generar un DELETE o UPDATE



Bases de Datos Temporales

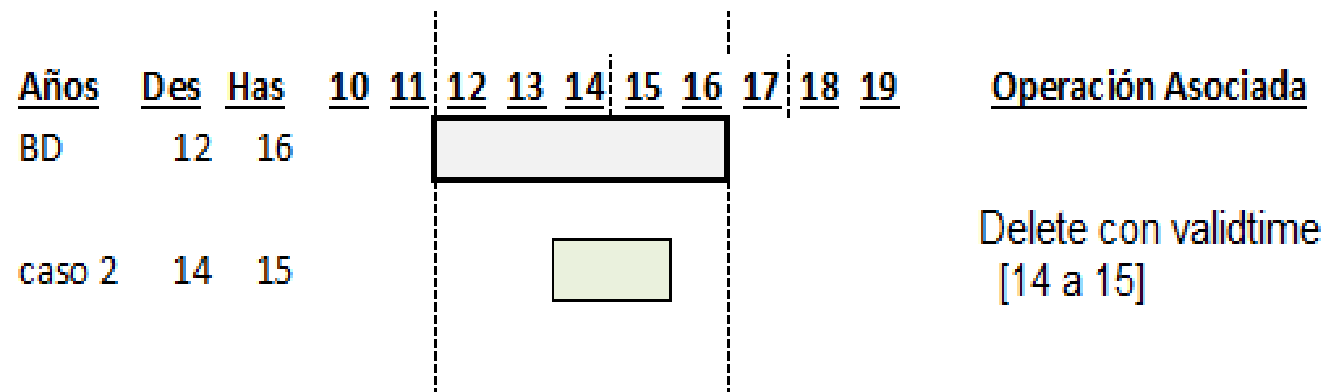
Analisis de la situación con DELETE



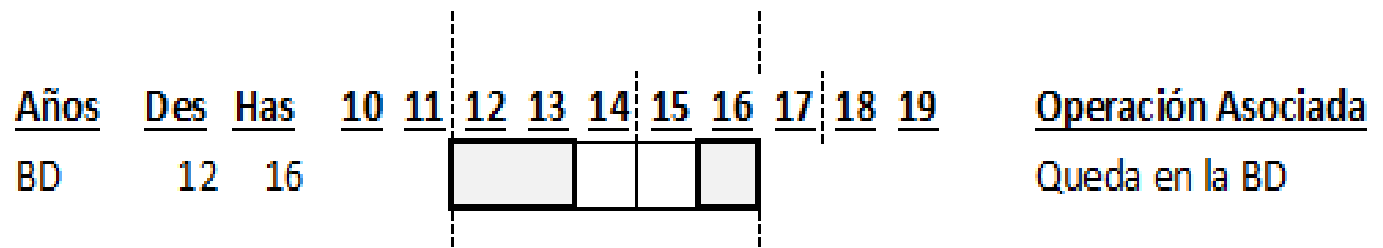
Bases de Datos Temporales

Tiempo de Validez

- **Delete** con validtime [14 a 15]



- Efecto causado



Bases de Datos Temporales

Ej de Relaciones de TIEMPO VALIDO

Relación con TIEMPO-VALIDO

```
CREATE TABLE Employee(  
  Name VARCHAR(30),  
  Manager VARCHAR(30) VALIDTIME REFERENCES  
    Employee (Name),  
  Dept VARCHAR(20)) AS VALIDTIME PERIOD (DATE)
```

TIEMPO VALIDO : para Manager y Dept.

Nota: La DDL de creación de tablas deberá tener un mecanismo declarativo para especificar que atributos tendrán tiempos validos (historia).

Bases de Datos Temporales

Ej Relaciones de TIEMPO VALIDO

LISTA TODOS LOS EMPLEADOS QUE NO SON GERENTES

VALIDTIME

SELECT Name FROM Employee

EXCEPT

SELECT Manager FROM Employee

TIEMPO VALIDO : para Manager

Nota: Las DML de consultas deberán tener un mecanismo para describir si las consultas son actuales o historicas.

Bases de Datos Temporales

Ej Relaciones de TIEMPO VALIDO

LISTA LA CANTIDAD DE CAMBIOS DE DEPARTAMENTO

VALIDTIME

```
SELECT Dept, COUNT(*)      FROM Employee GROUP BY Dept
```

TIEMPO VALIDO : Para Dpto.

Bases de Datos Temporales

Ej Relaciones de TIEMPO VALIDO

CAMBIA EL GERENTE DEL DEPARTAMENTO 'Tools' en el año 1994 a 'Bob'

```
VALIDTIME PERIOD '[1994-1-1 - 1994-12-31]'
```

```
UPDATE Employee SET Manager = 'Bob' WHERE Dept = 'Tools'
```

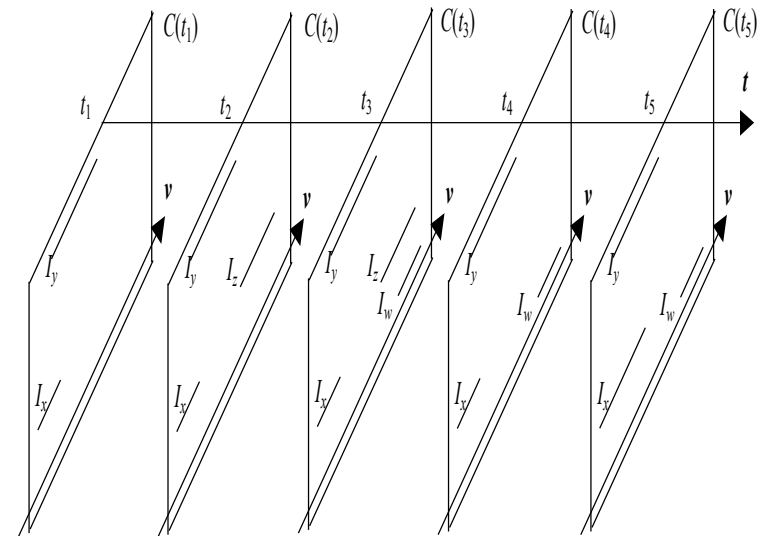
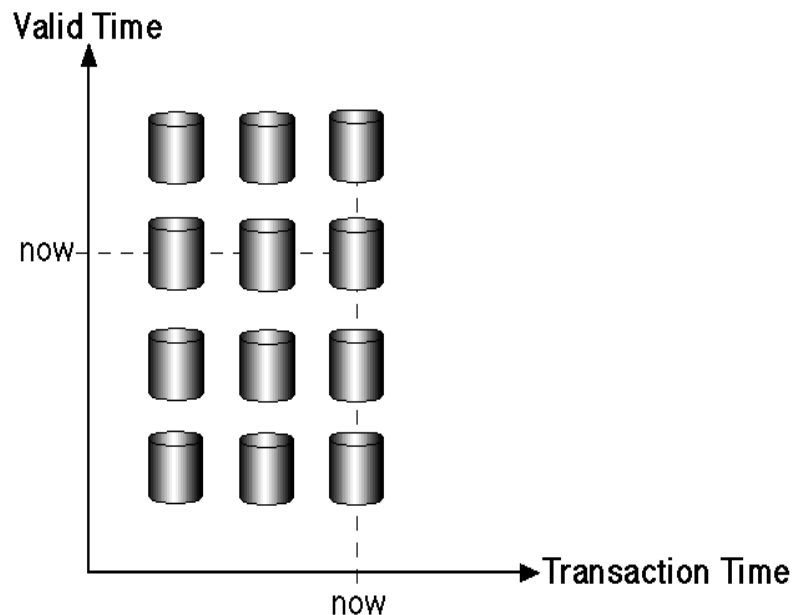
TIEMPO VALIDO : Para Manager y Dpto.

Nota: La DML de actualización deberán tener un mecanismo para expresar el periodo de validez.

Bases de Datos Temporales

Bi-Temporales

Los estados de una base de datos Bi-Temporales están representados de las siguientes maneras:



Bases de Datos Temporales

Ejemplo de BD Bitemporales

Emp	Dept	Valid Time	Transaction Time
Jake	Shipping	[1995-06-10 - 1995-06-16)	[1995-06-05 - 1995-06-10)
Jake	Shipping	[1995-06-05 - 1995-06-21)	[1995-06-10 - 1995-06-15)
Jake	Shipping	[1995-06-10 - 1995-06-16)	[1995-06-15 - 1995-06-20)
Jake	Loading	[1995-06-10 - 1995-06-16)	[1995-06-20 - 9999-12-31)

Figure 1: A Table With Both Valid-Time and Transaction-Time Support

- Contratación de Jake – refleja distintos tiempos de transacción por modificaciones practicadas, solo se refleja un cambio de Dept a Loading.

Bases de Datos Temporales

Ejemplo de BD Bitemporales

¿Quien actualiza cada dimensión?

- **Tiempo Valido:** Lo actualiza el **usuario** cuando se presenten cambios en el mundo real.
- **Tiempo de Transacción:** Es unicamente modificado (agregando información) por el **sistema gestor** de BD.

RESUMEN

- Motivación y origen BD Temporales
- Definición
- Características
- Tipos:
 - Tiempo Valido
 - Tiempo Transaccional
 - Bitemporales
- **Caso de Estudio**
- Algebra Temporal
- Implementaciones

Bases de Datos Temporales

Caso Estudio (I)

- Bernardo Sabina nació un soleado 6/3/1985 en Zaragoza. Su madre registró su nacimiento al día siguiente.
- Tras terminar sus estudios de Ingeniería el 15/6/2007, Bernardo se mudó ese mismo día a Torre vieja a vender hamburguesas con queso.
- Sin embargo, no registró su mudanza hasta el 25 de junio, ya que tenía una competición nacional de ping pong.
- Pese a tener un futuro prometedor, Bernardo murió el 20/09/2012 de un ladrillazo en la cabeza, cuando iba a ver la presentación de su hermano en un partido de fútbol. El equipo forense registró su muerte el mismo día.

Bases de Datos Temporales

Caso Estudio (II)

¿Qué granularidad sugiere?

¿Qué transacciones SQL se realizarían con una base de datos convencional?

¿Como representaría la evolución en una base de datos con tiempo de validez?

¿Como representaría la evolución en una base de datos bitemporal?

Bases de Datos Temporales

Caso Estudio (III)

Fecha	Hecho ocurrido	Acción de la BD	Vista en la BD
6/03/1985	Nace Bernardo.	-	-
7/03/1985	Se registra su nacimiento.	Inserción: (Bernardo,Zaragoza)	Bernardo vive en Zaragoza.
15/06/2007	Bernardo se muda a Torre vieja.	-	Bernardo vive en Zaragoza.
25/06/2007	Bernardo registra la mudanza.	Actualización: (Bernardo,Torre vieja)	Bernardo vive en Torre vieja.
20/09/2012	Bernardo muere; se registra el hecho.	Borrado: Bernardo	-
[...]	[...]	[...]	[...]

Bases de Datos Temporales

Caso Estudio (IV)

Tiempo de Validez (TV): lo habíamos definido como →
El periodo en el que un hecho es cierto en el mundo real.

Por ejemplo:

El TV de (Bernardo, Zaragoza) es 06/03/85-15/06/07.

Nombre	Ciudad	TVI	TVF
Bernardo	Zaragoza	06/03/1985	∞

Bases de Datos Temporales

Caso Estudio (IV)

Tiempo de Validez (TV):

Nombre	Ciudad	TVI	TVF
Bernardo	Zaragoza	06/03/1985	∞

Nombre	Ciudad	TVI	TVF
Bernardo	Zaragoza	06/03/1985	15/06/2007
Bernardo	Torre vieja	15/06/2007	∞

Bases de Datos Temporales

Caso Estudio (V)

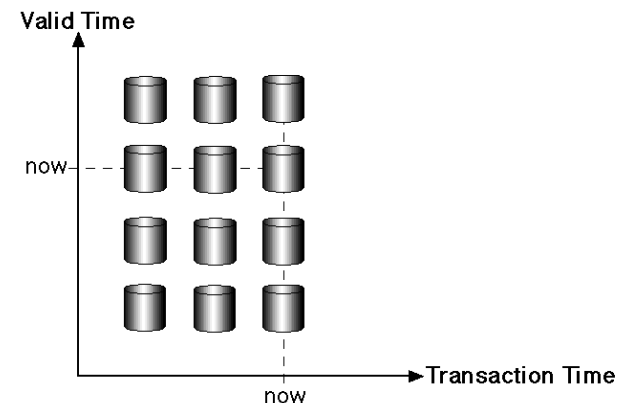
Tiempo de Transacción (TT): lo habíamos definido como →

El tiempo en el que se ha incluido el hecho en la BD.

Datos Bitemporales (TT): lo habíamos definido como →

Los tiempos que combinan (almacenan) TV y TT.

- *Tiempo de Validez inicial (TVI).*
- *Tiempo de Validez final (TVF).*
- *Tiempo de Transacción inicial (TTI).*
- *Tiempo de Transacción final (TTF).*



Bases de Datos Temporales

Caso Estudio (VI)

Datos Bitemporales (TT)

#Reg	Nombre	Ciudad	TVI	TVF	TTI	TTF
1	Bernardo	Zaragoza	06/03/85	∞	07/03/85	25/06/07
2	Bernardo	Zaragoza	06/03/85	15/06/07	25/06/07	now
3	Bernardo	Torre vieja	15/06/07	∞	25/06/07	20/09/12
4	Bernardo	Torre vieja	15/06/07	20/09/12	20/09/12	now

Bases de Datos Temporales

Aactualizaciones Tiempo de Validez (I)

Como trabajan las operaciones de Actualización?
Insert / Delete y Update

#Reg	Nombre	Ciudad	TVI	TVF	TTI	TTF
1	Bernardo	Zaragoza	06/03/85	∞	07/03/85	25/06/07
2	Bernardo	Zaragoza	06/03/85	15/06/07	25/06/07	now
3	Bernardo	Torre vieja	15/06/07	∞	25/06/07	20/09/12
4	Bernardo	Torre vieja	15/06/07	20/09/12	20/09/12	now

Bases de Datos Temporales

Actualizaciones Tiempo de Validez (II)

Funcionamiento de INSERT

Se crea una versión actual del dato, TVI correspondiente a los datos ingresados y TVF = ∞ .

Por ejemplo:

El TV de (Bernardo, Zaragoza) desde 06/03/85.

Nombre	Ciudad	TVI	TVF
Bernardo	Zaragoza	06/03/1985	∞

Bases de Datos Temporales

Actualizaciones Tiempo de validez (III)

Funcionamiento de DELETE

Actualiza el TVF del dato y pasa a ser Histórico (sin ningún registro actual).

F

Por ejemplo:

```
VALIDTIME Date '2012/09/20'  
DELETE FROM PERSONAS;
```

Nombre	Ciudad	TVI	TVF
Bernardo	Zaragoza	06/03/1985	20/09/12

Bases de Datos Temporales

Actualizaciones Tiempo de Validez (IV)

Funcionamiento de UPDATE

Eliminación + Insert

F

Por ejemplo:

VALIDTIME Date '2007/06/15'

```
UPDATE PERSONAS SET CIUDAD = 'TORREVIEJA'
WHERE NOMBRE = 'BERNARDO';
```

#Reg	Nombre	Ciudad	TVI	TVF	
2	Bernardo	Zaragoza	06/03/85	15/06/07	← DELETE
3	Bernardo	Torre vieja	15/06/07	∞	← INSERT

Bases de Datos Temporales

PRIMARY KEY (I)

Las claves primarias ya no son válidas.

¿Qué claves primarias usamos en BDT?

#Reg	Nombre	Ciudad	TVI	TVF	TTI	TTF
1	Bernardo	Zaragoza	06/03/85	∞	07/03/85	25/06/07
2	Bernardo	Zaragoza	06/03/85	15/06/07	25/06/07	now
3	Bernardo	Torreveja	15/06/07	∞	25/06/07	20/09/12
4	Bernardo	Torreveja	15/06/07	20/09/12	20/09/12	now

Bases de Datos Temporales

PRIMARY KEY (II)

¿Qué claves primarias usamos en BDT?

#Reg	Nombre	Ciudad	TVI	TVF	TTI	TTF
1	Bernardo	Zaragoza	06/03/85	∞	07/03/85	25/06/07
2	Bernardo	Zaragoza	06/03/85	15/06/07	25/06/07	now
3	Bernardo	Torre vieja	15/06/07	∞	25/06/07	20/09/12
4	Bernardo	Torre vieja	15/06/07	20/09/12	20/09/12	now

Bases de Datos Temporales

PRIMARY KEY (III)

¿Qué claves primarias usamos en BDT?

Si usamos una BD convencional, utilizaríamos como clave “Nombre” (Bernardo).

• Una clave primaria en BDT debe incluir:

- Características de clave primaria de BD convencional
- Un periodo (y por tanto sus restricciones)
- Restricciones de contigüidad y solapamiento.

- Si insertas un nuevo dato con un periodo no contiguo, **WARNING!**
- Si insertas un nuevo dato con un periodo que se solapa con otro, **ERROR!**

• Una alternativa de clave para Persona sería:

PRIMARY KEY (Nombre, TVI, TVF)

Pero: ¿que sucede con los UC? ¿Es mínima esta clave?

Bases de Datos Temporales

PRIMARY KEY (IV)

**¿Qué sucede con la integridad cuando por ej
Administramos posesión de recursos físicos únicos?**

Tendríamos que contar con mecanismos que nos permitan que no haya superposición entre periodos, mecanismos declarativos.

Ej: un vehículo no se puede alquilar a dos personas distintas en un mismo periodo.

- **pk** con clausula WITHOUT OVERLAPS.
- **Referencias de integridad temporal** con OVERLAPS, CONSTAINS, etc.

SQL 2011 provee mecanismos para generar claves primarias y restricciones sin superposición (OVERLAPS).

Bases de Datos Temporales

CONSTRAINT (I)

Aparecen nuevas restricciones.

¿Cuáles son y cómo las tratamos?';

Bases de Datos Temporales

CONSTRAINT (II)

¿Qué Restricciones Nuevas aparecen?

Restricciones Nuevas

- Restricciones propias de los periodos, por $TVF > TVI$.
- Restricciones propias de los periodos, TVI no puede ser nulo.
- Restricciones de contigüidad y solapamiento.
 - Si insertas un nuevo dato con un periodo no contiguo, **WARNING!**
 - Si insertas un nuevo dato con un periodo que se solapa con otro, **ERROR!**

Bases de Datos Temporales

IMPLEMENTACIONES

Se pueden realizar distintos tipos de implementaciones que cumplan con el modelo propuesto, de hecho una solución **Monolítica** (Una sola tabla contiene todas las tuplas actuales e históricas) sería totalmente factible. Pero existen otras alternativas de implementación.

1.**Partición horizontal:** crear dos tablas:

- una para la información válida
- otra para la información histórica

2.**Partición vertical:** se distribuyen los atributos de la relación temporal en distintas relaciones, los atributos que se actualizan a la vez se ponen en la misma relación.

Bases de Datos Temporales

IMPLEMENTACIONES

Por ejemplo tomemos:

Persona (Nombre,
Ciudad (Granularidad Día → Fecha),
montoBienesPersonales (Granularidad Año))

La solución **Monolítica** (Una sola tabla contiene todas las tuplas actuales e históricas) sería totalmente factible.

Persona (Nombre, Ciudad, montoBienesPersonales,
TV+TT)

Bases de Datos Temporales

IMPLEMENTACIONES

Por ejemplo tomemos:

Persona (Nombre,
Ciudad (Granularidad Día → Fecha),
montoBienesPersonales (Granularidad Año))

- 1.Solución Partición horizontal:** crear dos tablas:
- una para la información válida (actual)
 - otra para la información histórica

Persona (Nombre, Ciudad, montoBienesPersonales)
PersonaHistoria (Nombre, Ciudad, montoBienesPersonales,
+ TV + TT)

Pregunta → Como se gestionaría automáticamente?

Bases de Datos Temporales

IMPLEMENTACIONES

Por ejemplo tomemos:

Persona (Nombre,
Ciudad (Granularidad Día \rightarrow Fecha),
montoBienesPersonales (Granularidad Año))

2.Partición vertical: se distribuyen los atributos de la relación temporal en distintas relaciones, los atributos que se actualizan a la vez se ponen en la misma relación.

Persona (Nombre, Ciudad, montoBienesPersonales)

PersonaDomicilio (Nombre, Ciudad, + TV + TT)

PersonaBiensPerso (Nombre, monto, + TV + TT)

Pregunta \rightarrow Como se gestionaría automáticamente?

RESUMEN

- Motivación y origen BD Temporales
- Definición
- Características
- Tipos:
 - Tiempo Valido
 - Tiempo Transaccional
 - Bitemporales
- Caso de Estudio
- **Algebra Temporal**
- Implementaciones

Bases de Datos Temporales

Álgebra Temporal

Existen un conjunto de operadores que forman el Álgebra Temporal, estos se dividen los siguientes grupos:

- Operadores de Inclusión y Extracción de límites de periodos.
- Operadores de Comparación.
- Operadores de Periodos que producen Periodos.
- Operadores sobre conjuntos de Periodos.

Bases de Datos Temporales

Operadores Algebra Temporal (I)

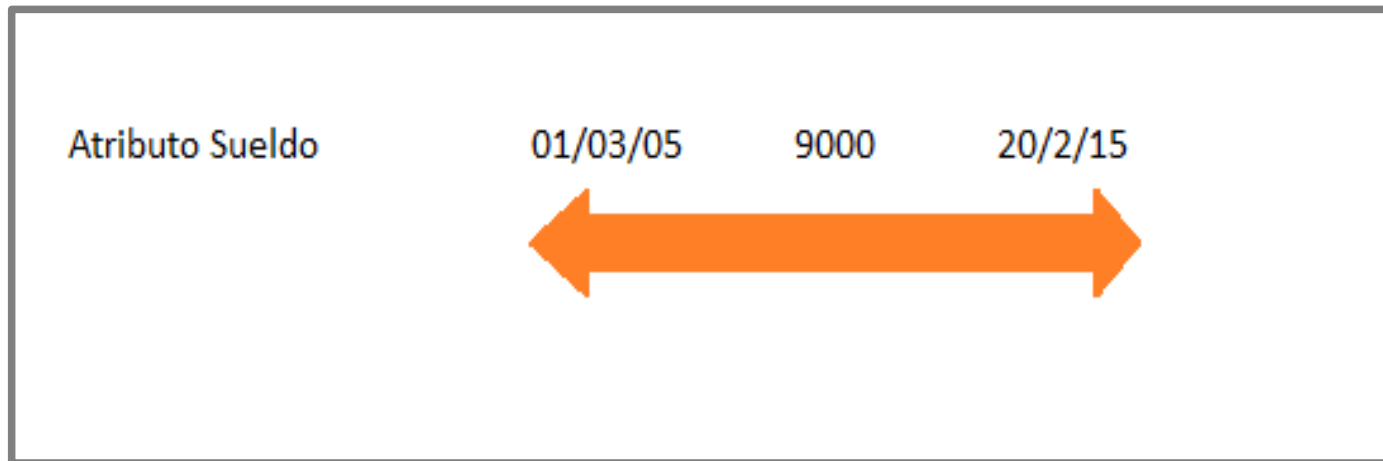
Operadores de Extracción de Limites de Periodos.

- **START**(I): primer valor del periodo I. Para el Ej: 01/03/05.
- **END**(I): ultimo valor del periodo I . Para el Ej: 20/02/15.

$START(I) \leq END(I)$

plpgsql: lower, upper
ver tb: lower_inc,
upper_inc

Ejemplo:



Bases de Datos Temporales

Operadores Algebra Temporal (II)

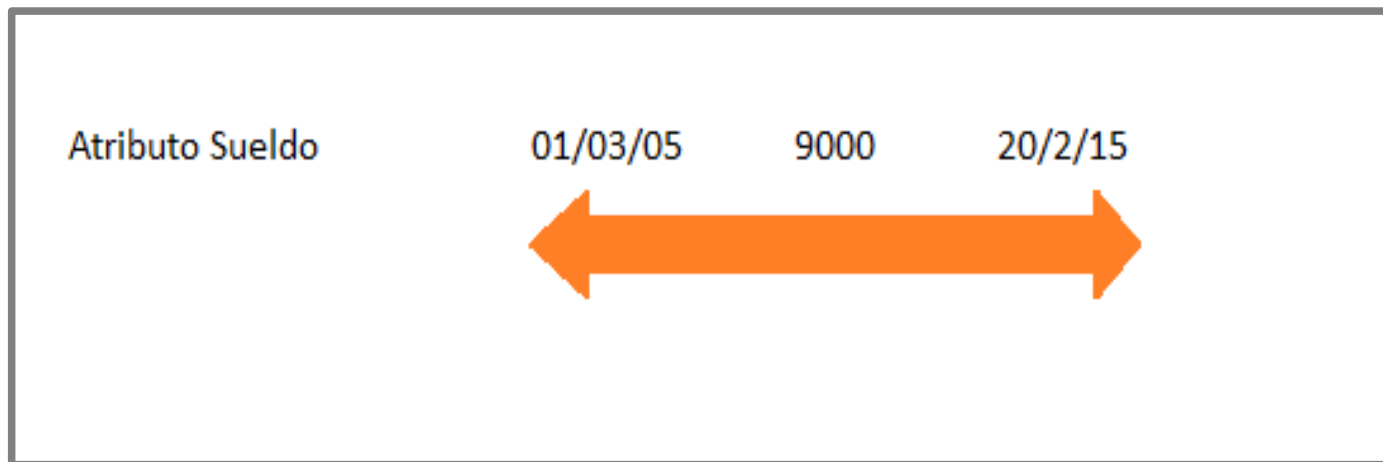
Operador de Inclusión de punto en Periodos.

- $s \text{ IN } I$: si s es un valor que cumple que $s \text{ IN}(i)=\text{TRUE}$, si y sólo si se cumple:

$$((\text{START}(I) \leq s) \text{ AND } (s \leq \text{END}(I)))$$

plpgsql: $s \leq @ I$

Ejemplo: Para el periodo ('05/03/01','20/02/15') y $s = '10/6/09'$ → True



Bases de Datos Temporales

Operadores Algebra Temporal (III)

Operadores de comparación (I):

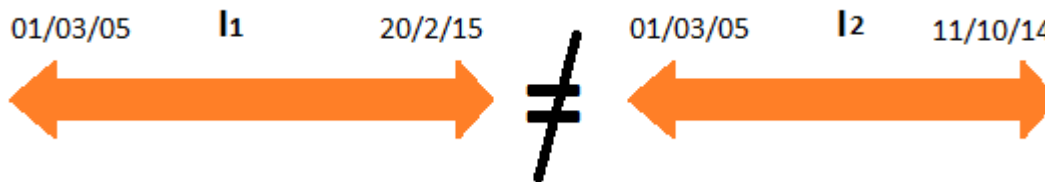
Sean $I1 = [i1, f1]$; $I2 = [i2, f2]$; Periodos plpgsql: por ej daterange

IGUALDAD: $I1 = I2$ si y sólo si $i1 = i2$ y $f1 = f2$

Ej1:



Ej2:



Bases de Datos Temporales

Operadores Algebra Temporal (IV)

Operadores de comparación (II):

Sean **I1** = [i1, f1]; **I2** = [i2, f2] ; periodos

plpgsql: **I1 << I2**

I1 BEFORE I2:

Es cierto si y sólo si $f1 < i2$ es verdadero [i1,f1][i2,f2]

Ej 1:



Ej 2:



Bases de Datos Temporales

Operadores Algebra Temporal (V)

Operadores de comparación (III):

Sean $I1 = [i1, f1]$; $I2 = [i2, f2]$; periodos,

$I1$ **MEETS** $I2$:

plpgsql: $I1 -|- I2$

Es cierto si y sólo $i2 = f1 + 1$ ó $i1 = f2 + 1$, es decir, alguno es verdadero $[i1, f1][i2=f1+1, f2]$ ó $[i2, f2][i1=f2+1, f1]$

Ej 1:



Ej 2:



Bases de Datos Temporales

Operadores Algebra Temporal (VI)

Operadores de comparación (IV):

Sean $I1 = [i1, f1]$; $I2 = [i2, f2]$; periodos,

$I1$ **OVERLAPS** $I2$:

Es cierto si y sólo si $i1 \leq i2 \leq f1$ son ambos ciertos.

Ej 1:



Ej 2:



Bases de Datos Temporales

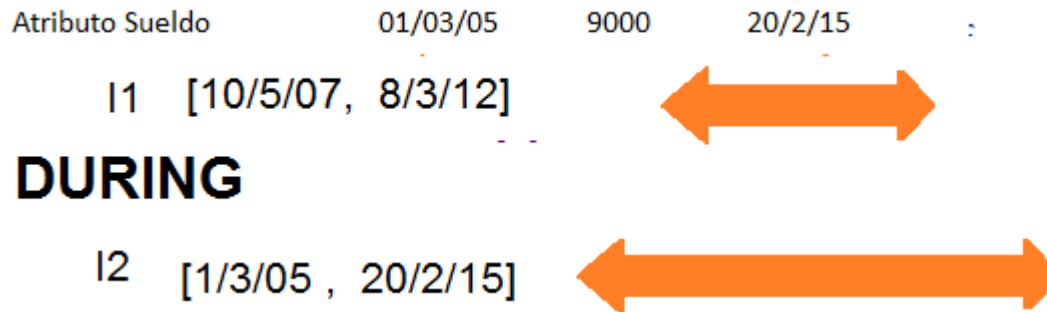
Operadores Algebra Temporal (VII)

Operadores de comparación (V):

Sean **I1= [i1, f1]**; **I2=[i2, f2]** ; periodos de tipo INTERVAL (tp),
donde $I1 = I2$ si y sólo si $i1 = i2$ y $f1 = f2$

I1 DURING I2:

Es cierto si y sólo si $i2 \leq i1$ y $f2 \geq f1$ son ambos verdaderos
[i2 [i1, f1]f2]



Bases de Datos Temporales

Operadores Algebra Temporal (VIII)

Operadores de comparación (VI):

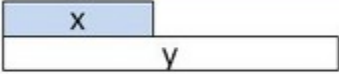
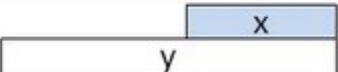
Sean $I1 = [i1, f1]$; $I2 = [i2, f2]$; periodos de tipo INTERVAL (tp), donde $I1 = I2$ si y sólo si $i1 = i2$ y $f1 = f2$

I1 START I2:

Es cierto si y sólo si $i1 = i2$ y $f1 \leq f2$ son ambos verdaderos
[$i1=i2, f1$] $f2$]

I1 FINISHES I2:

Es cierto si y sólo si $f1 = f2$ y $i1 \geq i2$ son ambos verdaderos
[$i2$] $[i1, f1=f2]$

Relation	Illustration
X starts Y	
X finishes Y	

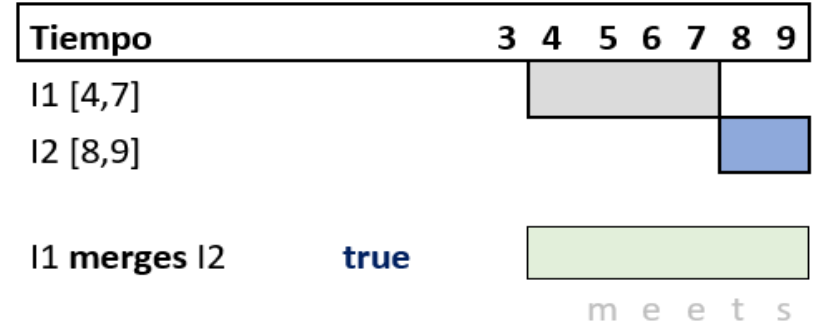
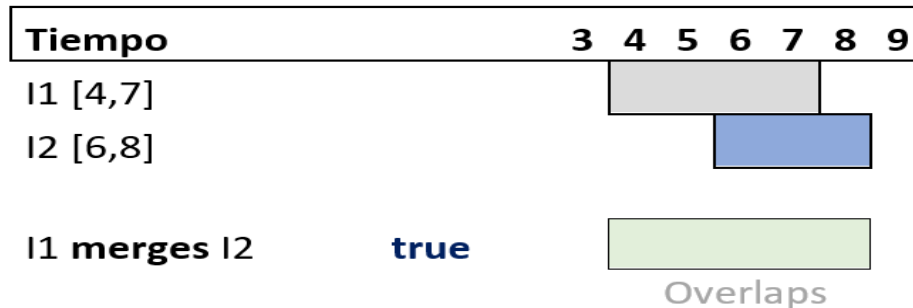
Bases de Datos Temporales

Operadores Algebra Temporal (IX)

Operadores de comparación (VII):

I1 **MERGES** I2:

Es cierto si y sólo si i1 **MEETS** I2 es verdadero o si I1 **OVERLAPS** I2 es verdadero.



Bases de Datos Temporales

Operadores Algebra Temporal (X)

Operadores de comparación (VII):

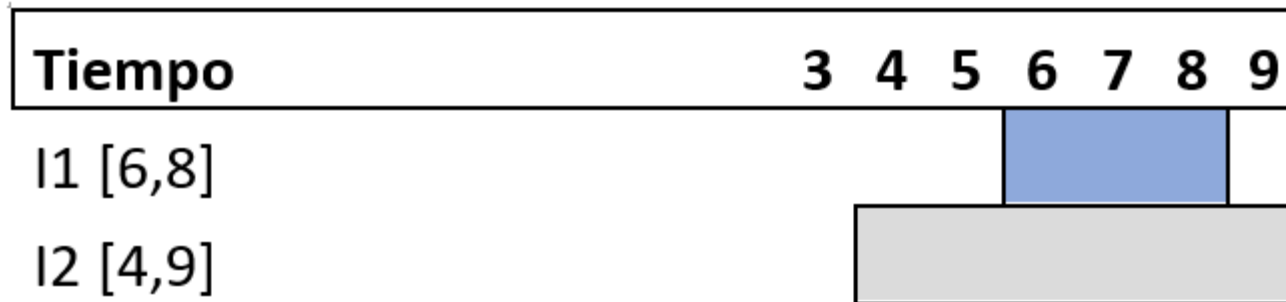
DURATION(I1):

Devuelve la cantidad de puntos que hay en el periodo I1.

I2 **CONTAINS** I1:

Es cierto si y sólo si I1 DURING I2 es verdadero.

Inverso de During



I2 contains I1

true

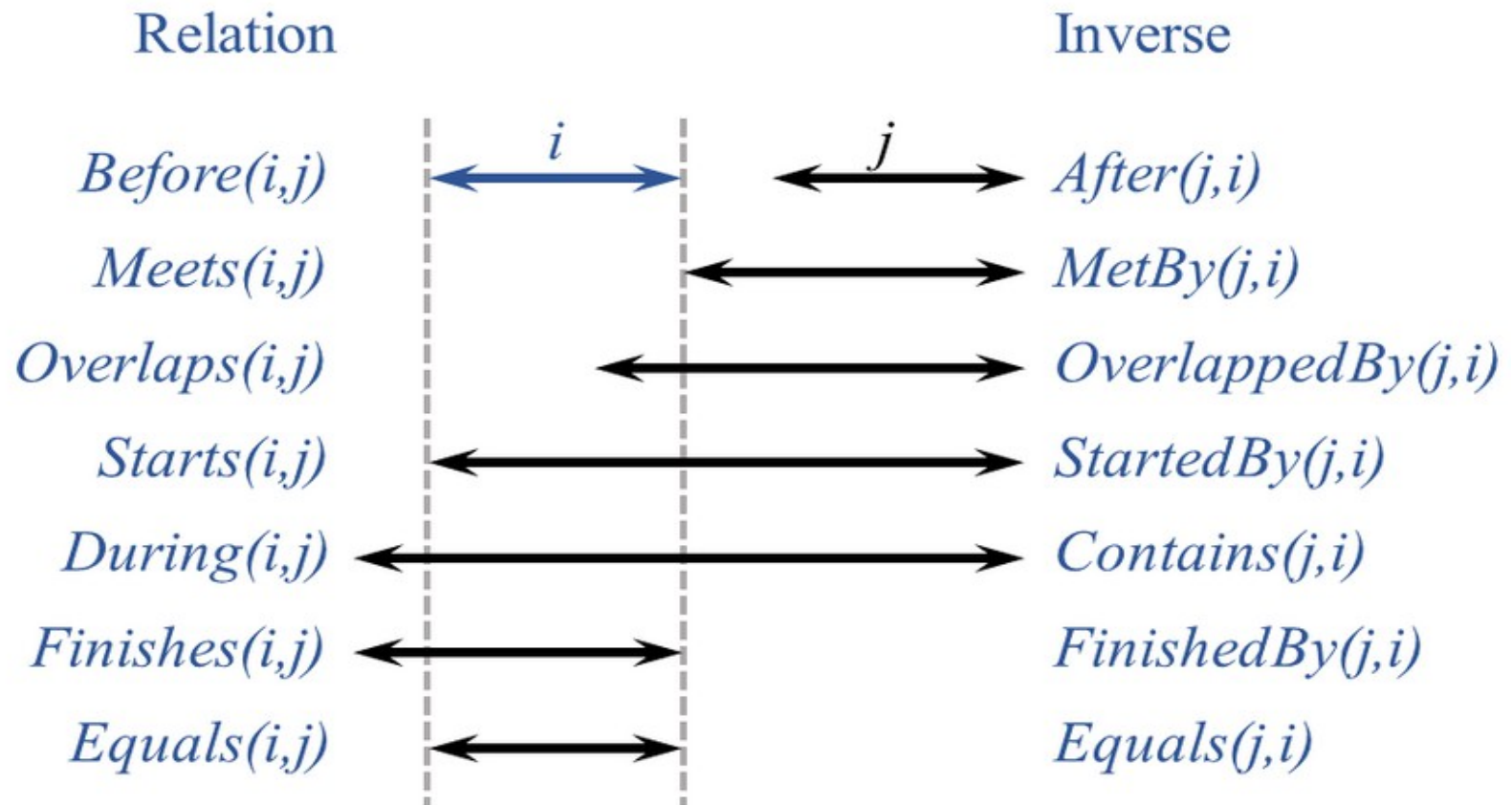
duration(I2)

6

Bases de Datos Temporales

Operadores Algebra Temporal (XI)

Resumen de Operadores y sus Inversos
Para aplicar entre Periodos.



Bases de Datos Temporales

Operadores Algebra Temporal (XII)

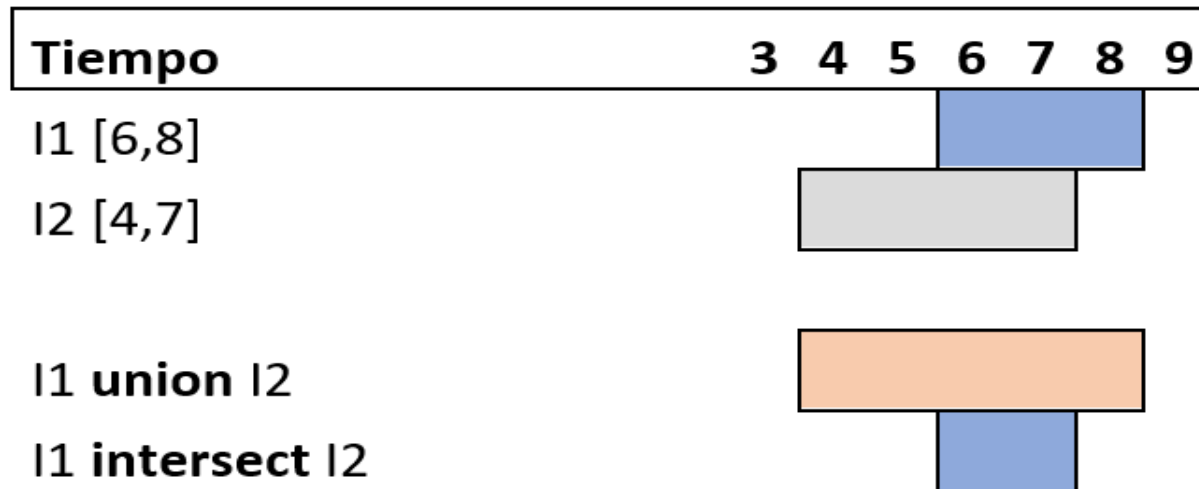
Operadores de Periodos que Producen Periodos:

I1 **UNION** I2:

Produce $[\text{MIN}(i1, i2), \text{MAX}(f1, f2)]$ cuando I1 **MERGES** I2 es verdadero y en caso contrario indefinido.

I1 **INTERSECT** I2:

Produce $[\text{MAX}(i1, i2), \text{MIN}(f1, f2)]$ cuando I1 **OVERLAPS** I2 es verdadero y queda indefinido en otro caso.



Bases de Datos Temporales

Operadores Algebra Temporal (XIII)


Operadores sobre Conjuntos de Periodos:

COALESCE:

Agrupar Periodos en unidades mayores.

Agrega, temporalmente, varias filas en una o mas.

Salary	Start	End
6000	1995-01-01	1995-05-31
7000	1995-06-01	1995-09-30
7000	1995-10-01	1996-01-31
7000	1996-02-01	1996-12-31



Salary	Start	End
6000	1995-01-01	1995-05-31
7000	1995-06-01	1996-12-31

UNFOLD:

Desglosa Periodos en unidades menores.

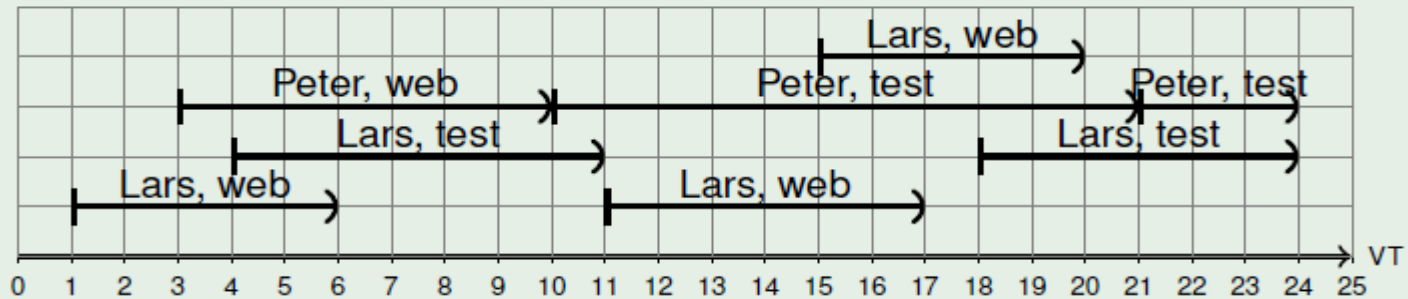
Podríamos decir que es la operación inversa a coalesce temporal.

Bases de Datos Temporales

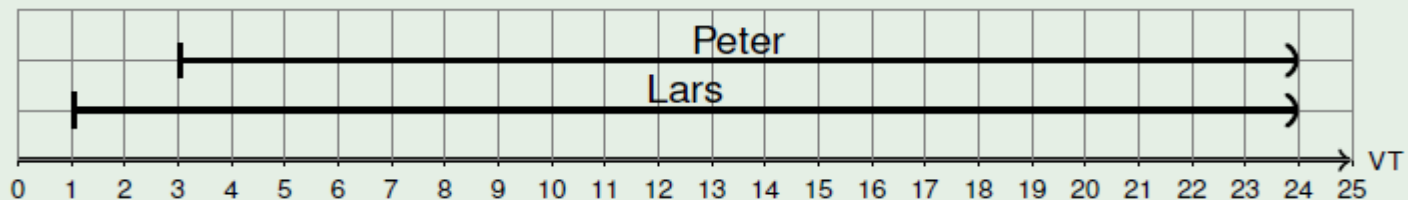
Operadores Algebra Temporal (XIV)

COALESCE: Agrupa Periodos en unidades mayores.

Ejemplo (¿Cuanto Tiempo han Trabajado Peter y Lars con nosotros?)



Resultados



Operaciones Efectuadas

- Seleccionar el nombre del empleado
- Coalescing el resultado en el numero menor de filas

Bases de Datos Temporales

Operadores Algebra Temporal (XV)

UNFOLD: Desglosa Periodos en unidades menores.

Es decir **UNFOLD** sobre una relación R con respecto a un atributo A, reemplaza cada tupla por un conjunto de tupla s que mantienen los mismos valores para todos los atributos de la tupla excepto para el atributo A, para el cual aparecerá cada uno de los elementos simples que pertenecen a ese Periodo.

Por ejemplo hacer un UNFOLD por año de una variable sueldo que tiene granularidad (periodo de date).

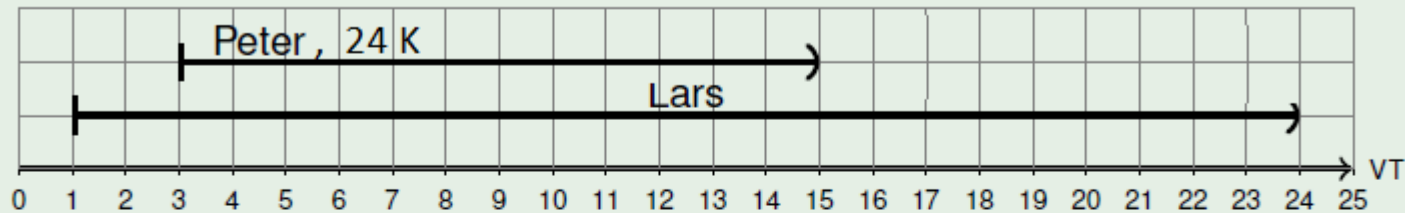
Caso opuesto a **Coalesce**

Bases de Datos Temporales

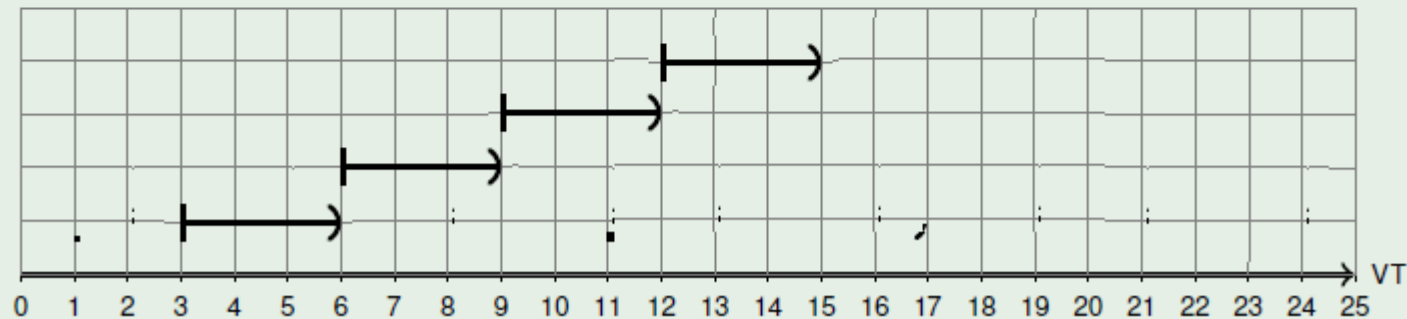
Operadores Algebra Temporal (XVI)

UNFOLD: Desagrega una fila en varias.

Ejemplo, ¿Determine los salarios de Peter (una o mas filas), por trimestre?



Resultados



RESUMEN

- Motivación y origen BD Temporales
- Definición
- Características
- Tipos:
 - Tiempo Valido
 - Tiempo Transaccional
 - Bitemporales
- Caso de Estudio
- Algebra Temporal
- **Implementaciones**

Bases de Datos Temporales

Sistemas de Gestion BDT (I)

Los SGBD comerciales (**Oracle, PostgreSQL, SqlServer, O2...**) NO nacieron con la capacidad de realizar gestión bitemporal de datos.

Paulatinamente van incorporando esta funcionalidad.

Bases de Datos Temporales

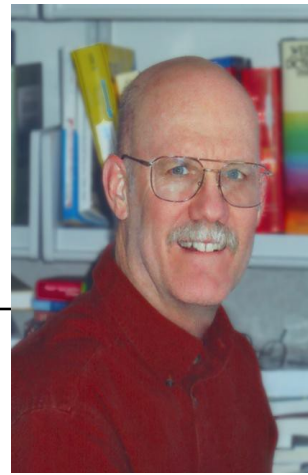
Sistemas de Gestion BDT (II)

Existen varias estrategias desarrolladas:

- Extensiones a motores existentes **TSQL2**:
 - ✓ **Base de Datos Oracle** - Oracle Workspace Manager es una característica de Oracle que permite datos temporales.
 - ✓ **PostgreSQL versión 9.2** Varió los tipos de datos que son capaces de implementar todas las características de la pgFoundry temporal contribuyó extensión.
 - ✓ **Microsoft SQL Server** introduce Tablas temporales como una característica de SQL Server 2016.
- *Auténticos* SGBD temporales
 - ✓ **TimeDB**
 - ✓ Tiger System
 - ✓ Time Series Cartridge

Bases de Datos Temporales

Sistemas de Gestion BDT (III)

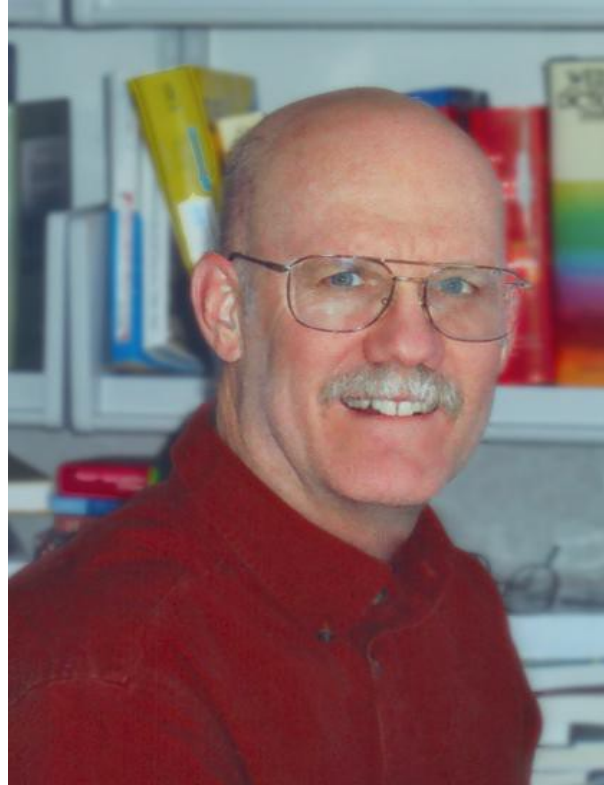


TSQL2 es una extensión temporal a SQL92, posiblemente, la más conocida

- En 1994 un comité encabezado por Richard T. Snodgrass (Universidad de Arizona) produjo el lenguaje de especificación ACM SIGMOD Record, semilla de TSQL2, que fue diseñado en el mismo año
- Las construcciones e ideas de TSQL2 han sido incorporadas a SQL2011 y SQL3

Bases de Datos Temporales

Sistemas de Gestion BDT (III)



Richard T. Snodgrass (Universidad de Arizona)

Bases de Datos Temporales

Sistemas de Gestion BDT (III)



- TimeDB es un SGBDT basado en SQL
- Fue desarrollado por Andreas Steiner (Instituto Federal Suizo de Tecnología, Zurich) en su Ph.D
- Su última versión(2.0) está basada en Java, usa JDBC, posee una API y ofrece mayor funcionalidad
- Soporta TSQL2

Bases de Datos Temporales

TimeDB – Ejemplo (I)

Vamos con el Ejemplo de relación temporal:

Persona (Nombre, Ciudad)

```
CREATE TABLE PERSONA  
(Nombrechar[20], Ciudad char[20])  
AS VALIDTIME AND TRANSACTIONTIME
```

Notar que el SG debería permitir opcionalmente especificar la granularidad (hora, día, mes...).

Bases de Datos Temporales

TimeDB – Ejemplo (II)

Queremos realizar unas consultas:

Datos Actuales:

VALIDTIME

```
SELECT Ciudad FROM Persona WHERE Nombre = 'Bernardo'
```

Datos actuales: Donde vive Bernardo ahora.

Datos Históricos:

TRANSACTIONTIME

```
SELECT Ciudad FROM Persona WHERE Nombre = 'Bernardo'
```

Datos Históricos: Donde ha vivido Bernardo.

Bases de Datos Temporales

TimeDB – Ejemplo (III)

Queremos realizar unas consultas:

Datos Actuales e Históricos Combinados:

VALIDTIME AND TRANSACTIONTIME

```
SELECT Ciudad FROM Persona WHERE Nombre = 'Bernardo'
```

Datos actuales e históricos: Donde vive Bernardo ahora y donde ha vivido.

Bases de Datos Temporales

CONCLUSIONES (I)

Las SGBDT amplían la potencia de muchos de los tipos de SGBD existentes en la actualidad.

- En una BDT tenemos la evolución de los datos gracias a marcas en el tiempo.
- Podríamos crear una BDR actual a partir de una BDT.
- Podríamos crear una BDR a una fecha determinada a partir de una BDT (Rollback de Base de Datos)
- Sin embargo, si no se desea una gestión temporal, podríamos estar usando recursos innecesariamente.

Bases de Datos Temporales

CONCLUSIONES (II)

La principal desventaja es que los SGBDT poseen un tiempo de respuesta inferior a los SGBDR para relaciones temporales.

¿Y si se consigue eficiencia máxima en el tratamiento de las fechas?

¿Desaparición de las BDR?



Bases de Datos Temporales

Fuentes

Libro: Introducción a los SISTEMAS DE BASES DE DATOS

Autor: C.J. Date

Editorial: Addison Wesley

Libro: Fundamentos de Sistemas de Bases de Datos

Autor: Elmasri / Navathe

Editorial: Pearson

Libro: Fundamentos de Bases de Datos

Autor: Silberschatz / Korth / Sudarshan

Editorial: Mc Graw Hill