

# Chapter 10: Temporal Database

# Introduction

- Time is an important aspect of all real-world phenomena.
- Every activity or change in the real world takes place in the context of time.
- Today, in an age dominated by technology and information systems, the recording of time is even more vital and essential to many computer applications .
- Conventional DBMS's lack the capability to record and process time-varying aspects of the real world , which provide us with the current database state at any given time .
- Temporal database has the ability to store different states (past, current, future ) of database .

# Temporal Database

What are the (Temporal Database , Main Temporal Time Aspects , Time-Stamping Approaches) ?

- **Temporal Database** : is a database which supports some aspects of time and store time-varying data .
- **Main Time Aspects** : Valid-Time, Transaction-Time, Bi-Temporal .
- **Valid Time** :is the time when the fact is true in the mini-world , Valid time is usually supplied by the user.
- **Transaction Time** : is the time when the fact is currently active in the database, Transaction time is usually supplied by the system .
- **Bi-Temporal Time**: A fact could have a bi-temporal time, when it involves both valid and transaction times . It support the Proactive , Retroactive , and simultaneous change .

# Temporal Database Concepts

## Time Representation, Calendars, and Time Dimensions

- Time is considered **ordered sequence** of **points** in some **granularity**
  - Use the term **choronon** instead of point to describe minimum granularity

# Temporal Database Concepts

## Time Representation, ... (cont.)

- A **calendar** organizes time into different time units for convenience.
  - Accommodates various calendars
    - Gregorian (western)
    - Chinese
    - Islamic
    - Hindu
    - Jewish
    - Etc.

# Temporal Database Concepts

## Time Representation, ... (cont.)

- Point events
  - Single time point event
    - E.g., bank deposit
  - Series of point events can form a time series data
- Duration events
  - Associated with specific time period
    - Time period is represented by start time and end time

# Temporal Database Concepts

## Time Representation, ... (cont.)

- Transaction time
  - The time when the information from a certain transaction becomes valid
- Bitemporal database
  - Databases dealing with two time dimensions

# Temporal Database Concepts

## Incorporating Time in Relational Databases Using Tuple Versioning

- Add to every tuple
  - Valid start time
  - Valid end time



# Time-Stamping Approaches

- **Time stamping** : is a way of associating temporal time aspects(Valid Time, Transaction Time , or Bi temporal ) with the attribute which has varying values over the time.
  
- *Time-stamp approaches* :
  - 1- *Tuple-Timestamp using 1NF.*
    - Tuple-Timestamp Single Relation (TTSR) .
    - Tuple-Timestamp Multiple Relation (TTMR) .
  
  - 2- *Attribute-timestamp using N1NF .*

# Temporal Database Concepts : TTSR

## (a) EMP\_VT

Name	<u>Ssn</u>	Salary	Dno	Supervisor_ssn	<u>Vst</u>	Vet
------	------------	--------	-----	----------------	------------	-----

## DEPT\_VT

Dname	<u>Dno</u>	Total_sal	Manager_ssn	<u>Vst</u>	Vet
-------	------------	-----------	-------------	------------	-----

## (b) EMP\_TT

Name	<u>Ssn</u>	Salary	Dno	Supervisor_ssn	<u>Tst</u>	Tet
------	------------	--------	-----	----------------	------------	-----

## DEPT\_TT

Dname	<u>Dno</u>	Total_sal	Manager_ssn	<u>Tst</u>	Tet
-------	------------	-----------	-------------	------------	-----

## (c) EMP\_BT

Name	<u>Ssn</u>	Salary	Dno	Supervisor_ssn	<u>Vst</u>	Vet	<u>Tst</u>	Tet
------	------------	--------	-----	----------------	------------	-----	------------	-----

## DEPT\_BT

Dname	<u>Dno</u>	Total_sal	Manager_ssn	<u>Vst</u>	Vet	<u>Tst</u>	Tet
-------	------------	-----------	-------------	------------	-----	------------	-----

**Figure 26.7**

Different types of temporal relational databases. (a) Valid time database schema. (b) Transaction time database schema. (c) Bitemporal database schema.

# Temporal Database Concepts : TTSR

**Figure 26.8**

Some tuple versions in the valid time relations EMP\_VT and DEPT\_VT.

## EMP\_VT

Name	<u>Ssn</u>	Salary	Dno	Supervisor_ssn	<u>Vst</u>	Vet
Smith	123456789	25000	5	333445555	2002-06-15	2003-05-31
Smith	123456789	30000	5	333445555	2003-06-01	Now
Wong	333445555	25000	4	999887777	1999-08-20	2001-01-31
Wong	333445555	30000	5	999887777	2001-02-01	2002-03-31
Wong	333445555	40000	5	888665555	2002-04-01	Now
Brown	222447777	28000	4	999887777	2001-05-01	2002-08-10
Narayan	666884444	38000	5	333445555	2003-08-01	Now

...

## DEPT\_VT

Dname	<u>Dno</u>	Manager_ssn	<u>Vst</u>	Vet
Research	5	888665555	2001-09-20	2002-03-31
Research	5	333445555	2002-04-01	Now

...

# TTSR concept and example

- In (TTSR) model two additional time attribute (VST,VET) are added to the relation to support the time interval base.
- Therefore when the user, for instance, updates the manager of 'Fekry' employee from 102 to 101 , as illustrated in the following table ( see TR#6-U) .
- A new version will be created as a copy from the current version with the new manger value '101'.
- The VET value of old version will be changed to 24/5/2010 which entered by the user instead of NOW value (see TR# 3-I) ,
- the value of VST '25/5/2010' for the new version it will be entered by the user (see TR#6-U) and VET will be NOW value if it had not been entered .

# Employee example based on TTSR

<u>empn</u> <u>o#</u>	ENAM E	HIREDATE	JOB *	SALA RY *	MANGE R*	<u>VST</u>	<u>VET</u>	TR#
101	Essa	1/1/2007	DB <u>Desinger</u>	4500		1/1/2007	NOW, 25/12/2009	1-I
101	Essa	1/1/2007	DB Consultant	4500		26/12/2009	NOW, 31/12/2009	4-1- U
101	Essa	1/1/2007	DB <u>Desinger</u>	30000		1/1/2010	NOW, 31/12/2018	4-2- U
101	Essa	1/1/2007	President	30000		10/1/2019	NOW	5-3- U
102	Ali	4/1/2007	Analyst	3000	101	1/1/2007	NOW	2-I
103	<u>Fekry</u>	5/1/2007	Teacher	3000	102	5/1/2007	NOW, 24/5/2010	3-I
103	<u>Fekry</u>	5/1/2007	Teacher	3000	101	25/5/2010	NOW 30/5/2010	6-U 7-D

# Tuple-Timestamp Multiple Relation (TTMR)

- In **TTMR** approach, a relation is decomposed as follows:
- the time-varying attributes (temporal attribute) in the relation are decomposed into multiple relations,
- where each time-varying attribute is incorporated with additional time-stamp columns in a separate relation.
- Non-temporal attributes are still existed in the original relation with adding two additional attributes to record the object lifespan as illustrated in next figure.

# (TTMR) example

- As shown in the Figure in the next slide. holds the none time varying attributes (EMPNO#, ENAME, HIREDATE) plus the two additional time-stamp attributes (Lifespan Start time (LSST) , Lifespan End time (LSET)) instead of (VST,VET) as an interval based type, where LSST identify the time of object creation in the reality, the LSET identify the time for logical delete of the tuple. The time varying attributes ( JOB, SALARY ,MANAGER) are respectively decomposed into separate relation as illustrated in Figure1 ( see Table B, Table C and Table D). Therefore, when the users insert the object of 'Essa' employee in the database by applying the transaction T1 in the transaction list as an example ,then the none time attributes will be stored as new record in Table A(see TR# 1-I) , and the JOB attribute value will be stored as new record in Table B(see TR# 1-I), SALARY attribute value will be stored as new record in Table C (see TR# 1-I) and the MANAGER attribute value will be stored as new record in Table D(see TR# 1-I). Additionally, when the user update the salary of employee 'Essa' from 4500 to 30000 by applying the transaction T4, a new tuple version is stored as new record in table C(see TR# 4-2-U) , while The VET value of old version will be changed to the end time value 31/12/2009 instead of 'NOW" value (see TR# 1-I) and the VST will be entered by the user for the new version.



# Employee example based on TTSR

<u>empno#</u>	ENAME	HIREDATE	VST	VET	TR#
101	Essa	1/1/2007	1/1/2007	NOW	1-I
102	Ali	4/1/2007	1/1/2007	NOW	2-I
103	Fekry	5/1/2007	5/1/2007	30/5/2010	3-I-7-D

A. Employee relation with Constant Attribute

<u>empno#</u>	JOB	VST	VET	TR#
101	DB Desinger	1/1/2007	NOW, 25/12/2009	1-I
101	DB Consultant	26/12/2009	NOW, 31/12/2018	4-1-U
101	President	10/1/2019	NOW	5-3-U
102	Analyst	1/1/2007	NOW	2-I
103	Teacher	5/1/2007	NOW, 30/5/2010	3-I 7-D

B. Employee\_job relation

<u>empno#</u>	SALARY *	VST	VET	TR#
101	4500	1/1/2007	NOW, 31/12/2009	1-I
101	30000	1/1/2010	NOW	4-2-U
102	3000	1/1/2007	NOW	2-I
103	3000	5/1/2007	NOW, 30/5/2010	3-I 7-D

C. Employee\_salary relation

<u>empno#</u>	MANGER*	VST	VET	TR#
101	NULL	1/1/2007	NOW	1-I
102	101	1/1/2007	NOW	2-I
103	102	5/1/2007	NOW-24/5/2010	3-I
103	101	25/5/2010	NOW, 30/5/2010	6-U 7-D

D. Employee\_Mnager relation



# Temporal Database Concepts

## Incorporating Time in Object-Oriented Databases Using Attribute Versioning

- **Time varying attribute**

- An attribute that changes over time
  - E.g., age

- **Non-Time varying attribute**

- An attribute that does **not** changes over time
  - E.g., date of birth