Temporal and Real-Time Databases: A survey
Gultekin Ozsoyoglu and Richard T. Snodgrass
1995
Presented by Jian Xu
Discussion by Lan Wu

Outline
• Temporal database
• Real-time database

Temporal Database
• Time as an important domain
• Valid time and transaction time
  – Valid time models events in reality
  – Trans. time models facts known by database
• Temporal data, Temporal database
  – A database that supports some aspect of time.

An example (from wiki)
• a short biography of John Doe.
  – born on April 3rd, 1975 in Smallville. His birth was registered on April 4th, 1975.
  – He went to live on his own in Bigtown. Although he moved out on August 26th, 1994, he forgot to register the change of address officially.
  – He updated his record on December 27, 1994.
  – John Doe was accidentally hit by a truck on April 1st, 2001. The coroner reported his date of death on the next day.

Example (cont.)
• Recording valid time

<table>
<thead>
<tr>
<th>Name</th>
<th>Place</th>
<th>Start-T</th>
<th>End-T</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Doe</td>
<td>Smallville</td>
<td>3-Apr-1975</td>
<td>= 26-Aug-1993</td>
</tr>
<tr>
<td>John Doe</td>
<td>Bigtown</td>
<td>26-Aug-1993</td>
<td>= 1-Apr-2001</td>
</tr>
</tbody>
</table>

Example (cont.)
• Need for Transaction time

<table>
<thead>
<tr>
<th>Name</th>
<th>Place</th>
<th>VTStart</th>
<th>VTEnd</th>
<th>TTStart</th>
<th>TTEnd</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Doe</td>
<td>Smallville</td>
<td>03/04/1975</td>
<td>=</td>
<td>04/04/1975</td>
<td>27/12/1994</td>
</tr>
<tr>
<td>John Doe</td>
<td>Bigtown</td>
<td>26/08/1993</td>
<td>=</td>
<td>27/12/1994</td>
<td>=</td>
</tr>
</tbody>
</table>
Some approaches

- Temporal Relational
  - Data model
    - Extend relational model to support temporal data
  - Query languages
    - SQL based: TSQL2, HSQL, TDM...
    - RA based: Legol, TRA...
    - Other variations: Quel, QBE based

- Temporal Object-Oriented
  - Data models
    - Caruso, TIGUKAT, TOODM...
  - Query languages
    - SQL based: VISION, TOOSQL

A close look at TSQL2

- Temporal query language TSQL2
  - Record a media plan in a Temporal Database
  
  CREATE TABLE NBCShows
  ( ShowName Char(30) NOT NULL,
    InsertionLength INTERVAL SECOND,
    Cost Integer
  ) AS VALID STATE YEAR(2) TO NBCSeason;

  - Insert temporal record into the table

  INSERT INTO NBCShows
  VALUES ('Roseanne', INTERVAL '30',
  SECOND, 251000) VALID TIMESTAMP
  'Spring season 1994';

A close look at TSQL2

- Query the database!

  Example 2 How long has the Roseanne show run?

  SELECT SNAPSHOT ShowName,
  CAST(VALID(N) TO INTERVAL DAY)
  FROM NBCShows(Showname) AS N
  WHERE N.ShowName = 'Roseanne'

Temporal-Query processing

- Focus on Temporal-Relational db.
- Query optimization
  - Highly desired
    - DB size grows monotonically
  - More Involved
    - predicates involving time are harder to optimize
    - Inequalities are common
  - Opportunities exist
    - Time is one-dimension, single direction
Temporal Operations

- **New joins**
  - As time is presented as intervals, joins over temporal field involves interval comparison.
  - Implicit valid-time selection, only rows which are valid at the same time are joined
- **Temporal Indexes**
  - R-tree *designed for objects that has spans over dimensionalities*
  - Interval-tree *designed for intervals*

(looks) Related Models

- **Temporal**
  - Record time related tuples and query them
- **Time series**
  - Identifying the nature of the phenomenon represented by the sequence of observations
  - Forecasting
- **Stream**
  - Another sequential model

A note for stream system

- A stream can be views as a queue of data arriving at a port of your computer
- Stream elements can be time-stamped and treated as temporal data
- A number of temporal queries also apply to stream data
- Stream also have other distinctive characteristics and requirements

Have a break

The Real-time World

- **Real-time system**
  - Rapid reaction to events
  - Industry control, security alert ...
- **Real-time database**
  - Operation need to be finished timely
  - Validness of a results depends also on time

Motivations

- **Real-time system is commonly used.**
  - Aero-craft control system
  - Various of Surveillance, Detection, and Tracking tasks
- **Data collected by RT-sys need also be processed in real time**
  - To support decision
  - To ensure timely reaction
New types of transactions

- Hard, soft and firm transactions
  - Hard: transaction should never miss deadline
  - Soft: increasing penalty on miss of deadline
  - Firm: 0-1 penalty

- Transaction type affects policy and strategy on transaction-processing

Discussion

- Give examples for
  - Hard transaction
  - Soft transaction
  - Firm transaction
  and compare their common characters and differences.

Other factors

- Factors characterize real-time trans.
  - Transaction arrival pattern
  - Data access type
  - Accessed-object properties
  - Knowledge of items to be used
  - CPU and I/O time knowledge

Consistency

- Internal and external consistency
  - Internal: data satisfies constraints
  - External: data always reflect reality
  both with respect to transactions

- For real-time database, maintaining external consistency is of great importance

New Trans. Management Ideas

- Compare real-time transaction to conventional trans., management.
  - Trans. cooperate v.s. compete
  - Diff. in resolving data and resource conflicts
    - DBMS, maximize resource utilization
    - RT-DBMS, satisfy time constraints

Discussion

- Compare transaction management in conventional database to what in real-time database and reason why transaction cooperative is preferred.

- Do you find other factors that may shape the transaction management differently in the two types of systems?
Trans. processing in RT-DB

• Processing Hard-Deadline transaction
  – database need to ENSURE the transaction can be finished on time
  – Needs complete knowledge on above factors
  – Utilize of real-time task scheduling

Trans. processing in RT-DB

• Processing Soft-Deadline/Firm-Deadline
  • scheduling
    – Earliest-deadline first
    – Highest-value first
    – least-slack time first
    – dynamic priority assignment
  • Concurrency control
    – lock-based protocols
    – Timestamp ordering protocols
    – Optimistic concurrency control protocols

Conclusion

• Both types of db are desired and has wide application
• A lot of problems need to be investigated

• Spatial-Temporal database
• Stream system addresses some real-time applications.