Provenance Management in Databases under Schema Evolution

Shi Gao, Carlo Zaniolo
Department of Computer Science
University of California, Los Angeles
Provenance under Schema Evolution

- Modern information systems, particularly big science projects, undergo frequent database schema changes.
  - Mediawiki, 323 schema versions in 9 years
  - Atutor, 216 schema versions in 7 years
  - KtDMS, 105 schema versions in 6 years

- Therefore, we need an integrated provenance management for both data and metadata under schema evolution.
# Motivating Example

## Employee

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Department</th>
<th>Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Sam</td>
<td>CS</td>
<td>3000</td>
</tr>
</tbody>
</table>

Data Update: INSERT INTO Employee VALUES (200, ‘John’, ‘EE’, 4000)

Schema Change: RENAME COLUMN Pay IN Employee To Salary

DECOMPOSE TABLE Employee INTO Employee_Info(ID, Name, Department), Employee_Salary(ID, Salary)

## Employee_Info

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Department</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Sam</td>
<td>CS</td>
</tr>
<tr>
<td>200</td>
<td>John</td>
<td>EE</td>
</tr>
</tbody>
</table>

## Employee_Salary

<table>
<thead>
<tr>
<th>ID</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>3000</td>
</tr>
<tr>
<td>200</td>
<td>4000</td>
</tr>
</tbody>
</table>

How to connect the provenance of data created under different schemas?
AM&PM System

- Archived Metadata & Provenance Manager
- Goal: Manage the combined provenance of data and metadata under schema evolution
  - Extend the SQL Information Schema to archive the provenance of metadata
  - Provide a timestamp representation of the provenance database
  - Facilitate the expression of complex temporal query
Model

- A relational model that stores:
  - Data Provenance
    - The information of data updates and transactions applied to the content of database
  - Schema Provenance
    - The information of past schema versions and the history of schema evolution
  - Auxiliary Information
    - e.g. the removed values and database statistics
## Model

### Schema Provenance (where)

- TABLES
- COLUMNS
- CONSTRAINTS
- PRIVILEGES
- ...

### Data Provenance (how, when)

- Transaction
- Transaction_Text
- Timestamp

### Auxiliary Info

- Removed_Value
- Statistics
- ... (Optional)

* Schema Modification Operators (SMO) and Integrity Constraints Modification Operators (ICMO)  [H. Moon 2008]
**Model**

### Tables

<table>
<thead>
<tr>
<th>V</th>
<th>Name</th>
<th>TS</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Employee</td>
<td>t0</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>Employee_info</td>
<td>t3</td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>Employee_Salary</td>
<td>t3</td>
<td></td>
</tr>
</tbody>
</table>

### Table_Constraints

### Table_Privileges

### Columns

### Column_Privileges

### Version

... ... ...

**Information Schema**

**Schema Evolution**

**Data Provenance**

### SMO

<table>
<thead>
<tr>
<th>ID</th>
<th>Text</th>
<th>Source</th>
<th>Target</th>
<th>TS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>smo1</td>
<td>V1</td>
<td>V2</td>
<td>t2</td>
</tr>
<tr>
<td>2</td>
<td>smo2</td>
<td>V1</td>
<td>V2</td>
<td>t3</td>
</tr>
</tbody>
</table>

### ICMO

### Transaction

<table>
<thead>
<tr>
<th>ID</th>
<th>User</th>
<th>TS</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>App</td>
<td>t1</td>
<td>...</td>
</tr>
</tbody>
</table>

### Transaction_Text

<table>
<thead>
<tr>
<th>ID</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>tran</td>
</tr>
</tbody>
</table>

**smo1**: RENAME COLUMN Payment IN Employee To Salary

**smo2**: DECOMPOSE Table Employee INTO Employee_Info(ID, Name, Department), Employee_Salary(ID, Salary)

**tran**: INSERT INTO Employee VALUES (200, ‘John’, ‘EE’, 4000)
Provenance Queries

- Data Provenance Queries
  - Trace the provenance of data. For example, when the data is inserted and which transactions help generate the data

- Schema Provenance Queries
  - Trace the provenance of schema elements (tables and columns)

- Queries on Statistics
  - Statistical queries about the database content and schema
Architecture

- **Backend Database**
  - MySQL 5.1

- **Provenance DB Manager**
  - Parse input data
  - Construct provenance DB

- **AM&PM Parser**
  - Translate XQuery to SQL [F. Wang 2008]
  - Check syntax
Experiments

- We perform some preliminary experiments to evaluate the execution time of provenance queries on AM&PM provenance database

Datasets
- Synthetic Dataset: California Traffic
## Experiments

<table>
<thead>
<tr>
<th>Query</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>when</td>
<td>Find the creation time of the tuple with id 2357 in highway accident</td>
</tr>
<tr>
<td>Q2</td>
<td>how</td>
<td>Find the transaction which generates the tuple with id 19009 in highway condition</td>
</tr>
<tr>
<td>Q3</td>
<td>aggregate</td>
<td>Find the number of accidents happening on 04/02/2012</td>
</tr>
<tr>
<td>Q4</td>
<td>aggregate</td>
<td>Find the number of highway condition records on 04/03/2012</td>
</tr>
<tr>
<td>Q5</td>
<td>temporal join</td>
<td>Find the ids of accidents happening in the area of West Los Angeles between “04/04/2012 18:00:00” and “04/04/2012 23:00:00”</td>
</tr>
<tr>
<td>Q6</td>
<td>temporal join</td>
<td>Find the descriptions of highway condition updates happening in the area of Central LA between “04/04/2012 18:00:00” and “04/04/2012 23:00:00”</td>
</tr>
</tbody>
</table>

**Table: Data Provenance Queries for Evaluation**
Experiments

The performance of data provenance queries

- **Dataset: California Traffic**
  - The values are sampled from a small real-world traffic dataset
Conclusion

- AM&PM provides a simple way to support provenance management under schema evolution.
- Provenance queries on both data provenance and schema provenance are efficiently supported.

Ongoing work:
- Column store
- Provenance query rewriting
- ...

Thank You!
Reference


