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

V_1

ID	Name	Department	Pay
100	Sam	CS	3000

```
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

V_2

ID	Name	Department
100	Sam	CS
200	John	EE

Employee_Salary

ID	Salary
100	3000
200	4000

- How to connect the provenance of data created under different schemas?

Next

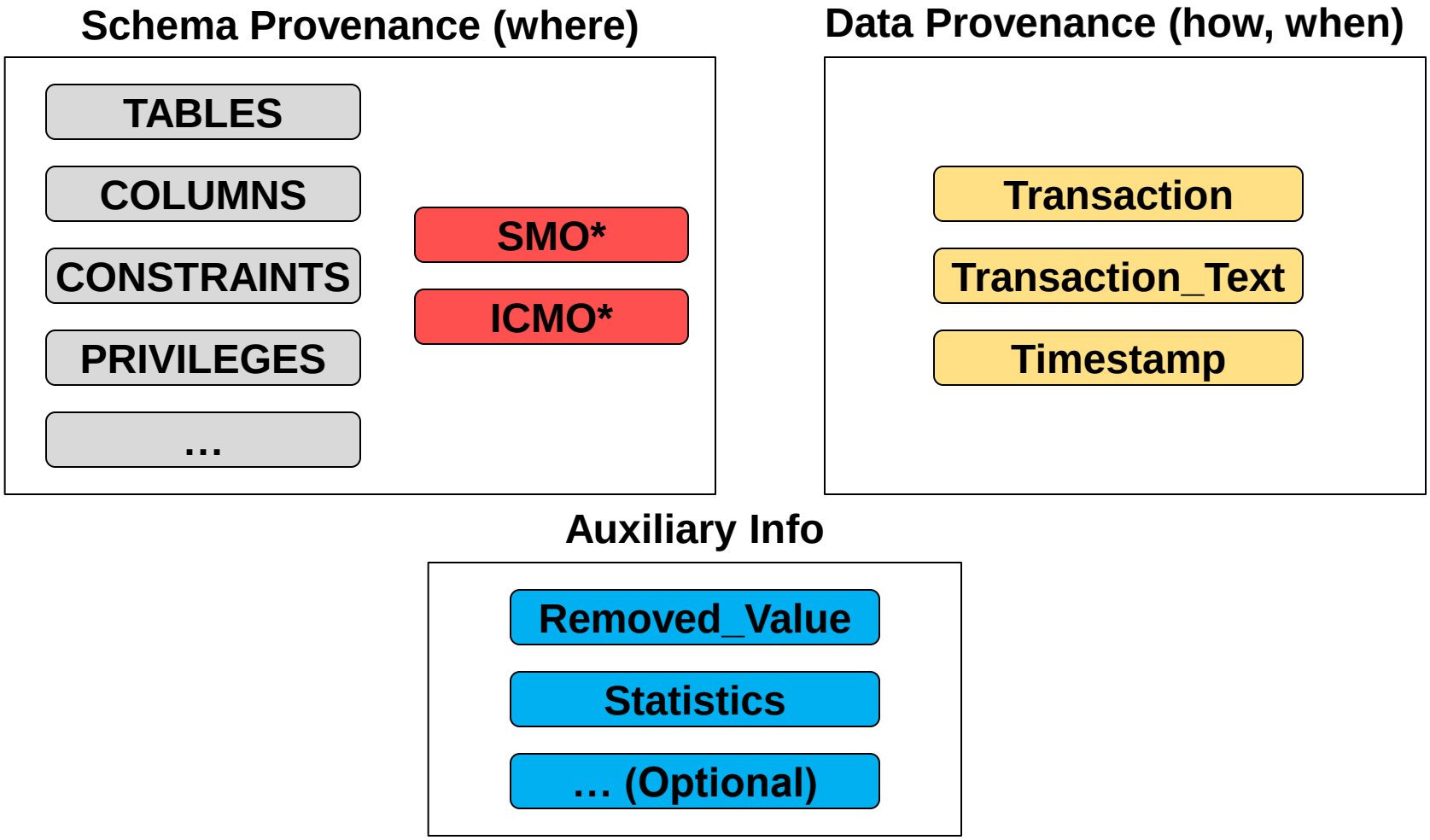
AM&PM System

- 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 Modification Operators(SMO) and Integrity Constraints Modification Operators(ICMO) [H. Moon 2008]

Model

Tables

V	Name	TS	…
1	*Employee*	t0	…
2	*Employee_info*	t3	…
3	*Empoyee_Salary*	t3	

Table_Constraints

Table_Privileges

Columns

Column_Privileges

Version

… …

SMO

ID	Text	Source	Target	TS
1	***smo1***	V1	V2	t2
2	***smo2***	V1	V2	t3

ICMO

ID	Text	Source	Target	TS

smo1: RENAME COLUMN Payment IN Employee To Salary

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

Transaction

ID	User	TS	…
1	App	t1	…

Transaction_Text

ID	Text
1	***tran***

Employee_Salary_Timestamp

Employee_Info_Timestamp

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

Information Schema | **Schema Evolution** | **Data Provenance**

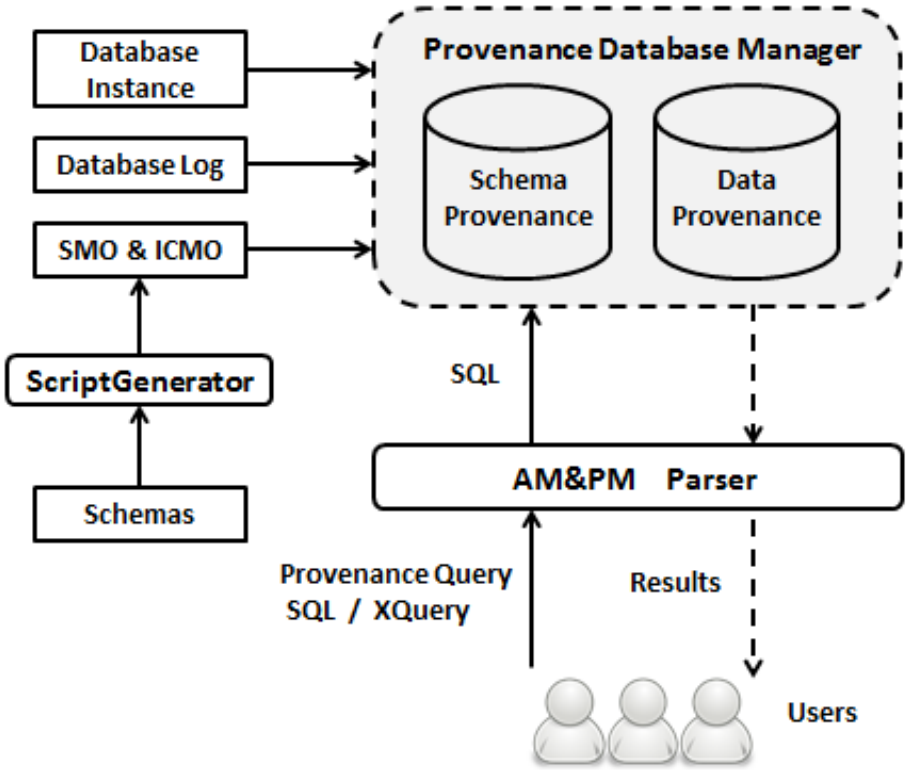
Previous

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

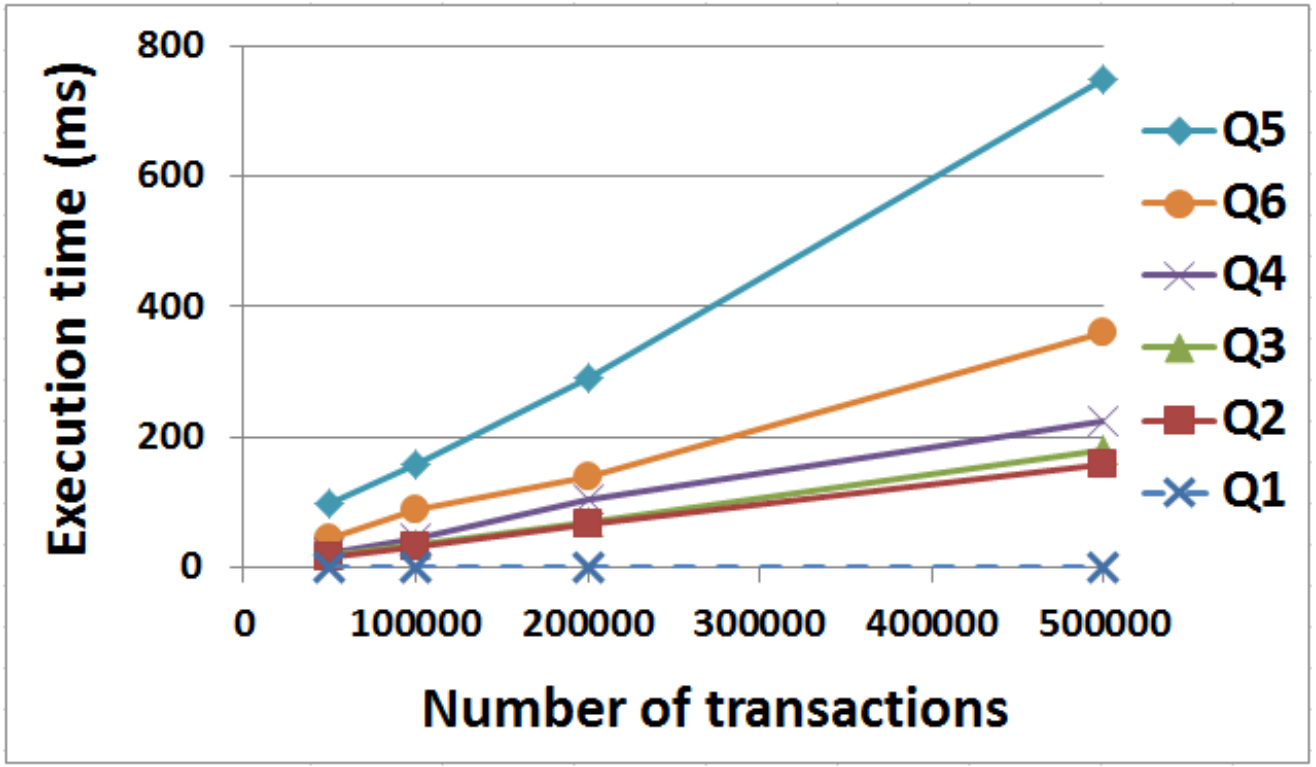
- We perform some preliminary experiments to evaluate the execution time of provenance queries on AM&PM provenance database
- Datasets
  - Synthetic Dataset: California Traffic

Experiments

Query	Type	Description
Q1	when	Find the creation time of the tuple with id 2357 in highway accident
Q2	how	Find the transaction which generates the tuple with id 19009 in highway condition
Q3	aggregate	Find the number of accidents happening on 04/02/2012
Q4	aggregate	Find the number of highway condition records on 04/03/2012
Q5	temporal join	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”
Q6	temporal join	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”

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

- [H. Moon 2008] H. J. Moon, C. A. Curino, A. Deutsch, C.-Y. Hou, and C. Zaniolo. Managing and querying transaction-time databases under schema evolution, VLDB, 2008.
- [F. Wang 2008] Fusheng Wang, Carlo Zaniolo, and Xin Zhou. Archis: an xml-based approach to transaction-time temporal database systems. The VLDB Journal, 17(6):1445-1463, November 2008.
- [J. Cheney 2009] Provenance in Databases: Why, How, and Where, James Cheney and Laura Chiticariu and Wang-Chiew Tan (2009), Foundations and Trends® in Databases: Vol. 1: No 4, pp 379-474
- Schema Evolution Benchmark: C. Curino and C. Zaniolo. Pantha rhei benchmark datasets. http://yellowstone.cs.ucla.edu/schema-evolution/index.php/Benchmark_home