Why and Where: A Characterization of Data Provenance

Peter Buneman, Sanjeev Khanna and Wang-Chiew Tan
University of Pennsylvania

Presenter: Fernando Seabra Chirigati

Motivation

Provenance

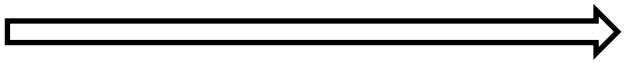
- Also known as “lineage” or “pedigree”
- Description of the origin, the source
- Importance
  - Accuracy and timeliness
  - Interpretation and understanding
  - Reproducibility
- Data Provenance
  - Derivation of a piece of data in a dataset

Data Provenance

Employee		
Name	DeptName	Salary
And	CBE	$50,000
Chris	CSE	$60,000
Robert	CSE	$55,000
Ryan	ECE	$40,000

Department	
DeptName	Address
CBE	6 MetroTech
CSE	2 MetroTech

Employee $\bowtie$ Department



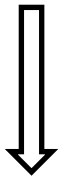
Name	DeptName	Salary	Address
And	CBE	$50,000	6 MetroTech
Chris	CSE	$60,000	2 MetroTech
Robert	CSE	$55,000	2 MetroTech

Why-Provenance

Data Provenance

Employee		
Name	DeptName	Salary
And	CBE	$50,000
Chris	CSE	$60,000
Robert	CSE	$55,000
Ryan	ECE	$40,000

```
SELECT Name, DeptName
FROM Employee
WHERE Salary > SELECT AVERAGE Salary
      FROM Employee
```

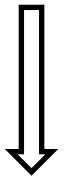


Chris	CSE
Robert	CSE

Data Provenance

Employee		
Name	DeptName	Salary
And	CBE	$50,000
Chris	CSE	$60,000
Robert	CSE	$55,000
Ryan	ECE	$60,000

```
SELECT Name, DeptName
FROM Employee
WHERE Salary > SELECT AVERAGE Salary
      FROM Employee
```

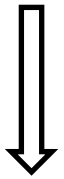


Chris	CSE
Robert	CSE

Data Provenance

Employee		
Name	DeptName	Salary
And	CBE	$50,000
Chris	CSE	$60,000
Robert	CSE	$55,000
Ryan	ECE	$60,000

```
SELECT Name, DeptName
FROM Employee
WHERE Salary > SELECT AVERAGE Salary
      FROM Employee
```



Chris	CSE
Ryan	ECE

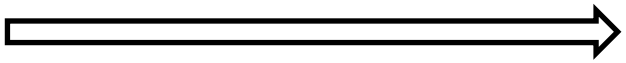
Data Provenance

Employee		
Name	DeptName	Salary
And	CBE	$50,000
Chris	CSE	$60,000
Robert	CSE	$55,000
Ryan	ECE	$40,000

Department	
DeptName	Address
CBE	6 MetroTech
CSE	2 MetroTech

Andy

Employee ⋈ Department



Name	DeptName	Salary	Address
And	CBE	$50,000	6 MetroTech
Chris	CSE	$60,000	2 MetroTech
Robert	CSE	$55,000	2 MetroTech

Where-Provenance

Existing Work

- Why-Provenance
  - Y. Cui and J. Widom. "Practical Lineage Tracing in Data Warehouses". ICDE, 2000.
    - Queries in the relational algebra
    - Semantic approach – restricted to SPJU queries
- Where-Provenance
  - No previous work

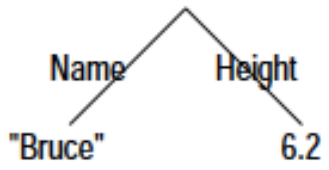
Main Idea

- Syntactic approach to compute provenance
  - Syntactic analysis of the query
- Both provenances
  - Why-Provenance
  - Where-Provenance
- Data of interest created by a database query
- Use of a general deterministic model
  - Relational databases
  - Semi-structured data (XML)

Deterministic Model

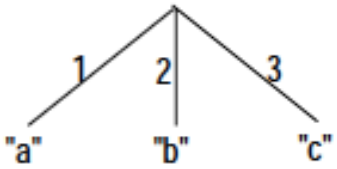
Syntax and Operations

- Values
  - $\{x_1:y_1, x_2:y_2, \ldots, x_n:y_n\}$



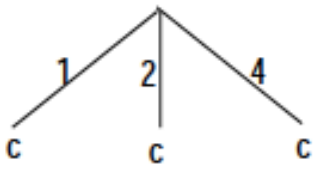
A record

{Name:"Bruce", Height:6.2}



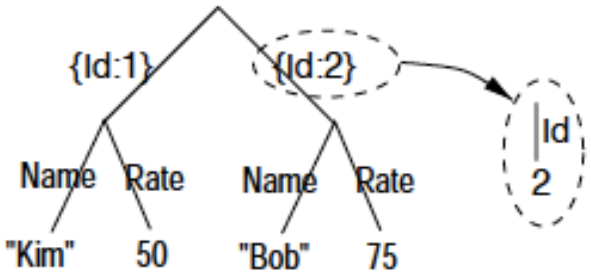
An array

{1:"a", 2:"b", 3:"c"}



A set

{1:c, 2:c, 4:c}

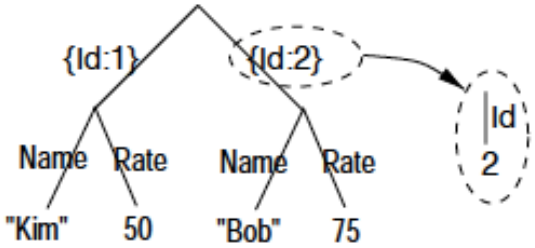


A relation

{ {Id:1} : {Name:"Kim", Rate:50},
{Id:2} : {Name:"Bob", Rate:75} }

Syntax and Operations

- Paths
  - $x_1.x_2.x_3. \ldots .x_n$
  - *e.g.*:
    - {Id:1}.Rate
- Abbreviation
  - $\{e_1:\{e_2:\{\ldots\{e_{n-1}:e_n\}\}\}\} = e_1.e_2. \ldots .e_{n-1}:e_n$
  - *e.g.*:
    - {{Id:1}:{Name:"Kim"}} = {Id:1}.Name:"Kim"



A relation

{ {Id:1} : {Name:"Kim", Rate:50},
{Id:2} : {Name:"Bob", Rate:75} }

Syntax and Operations

- Path Representation
  - {a:{1:c,3:d}} = {(a.1,c),(a.3,d)}
- Substructure
  - $z \sqsubseteq v$
  - Path representation of z is a subset of the path representation of v
  - *e.g.*:
    - z = a:{1:c,3:d} = {(a.1,c),(a.3,d)}
    - v = a:{1:c,2:b,3:d} = {(a.1,c),(a.2,b),(a.3,d)}

Syntax and Operations

- Deep Union
  - $v_1 \sqcup v_2$
  - Union of path representations
    - May be undefined
  - *e.g.*:
    - v_1 = {a:1,b.c:2} = {(a,1),(b.c,2)}
    - v_2 = {b.d:4,e:5} = {(b.d,4),(e,5)}
    - $v_1 \sqcup v_2$ = {(a,1),(b.c,2),(b.d,4),(e,5)} = {a:1,b:{c:2,d:4},e:5}

Deterministic Query Language

Query Fragment

- General Form

where $p_1 \in e_1,$
$\vdots$
$p_n \in e_n,$
$condition$
collect e

Example

Composers

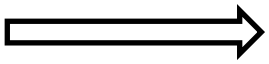
name	born	period
"J.S. Bach"	1685	"baroque"
"G.F Handel"	1685	"baroque"
"W.A Mozart"	1756	"classical"

Works

name	opus	title
"J.S. Bach"	"BMV82"	"I have enough."
"J.S. Bach"	"BMV552"	NULL
"G.F Handel"	"HMV19"	"Art thou troubled?"

```
{ Composers:
   {{name:"J.S. Bach"}: {born:1685, period:"baroque"},
    {name:"G.F. Handel"}: {born:1685, period:"baroque"},
    {name:"W.A. Mozart"}: {born:1756, period:"classical"}},
  Works: {{{name:"J.S. Bach"}.opus:"BMV82"}:
                    { title: "I have enough." },
          {{name:"J.S. Bach"}.opus:"BMV552"}:
                    { title: "-" },
          {{name:"G.F Handel"}.opus:"HMV19"}:
                    { title: "Art thou troubled?" }} }
```

where $\text{Composers}.x.\text{born}{:}u \in \text{D}$,
$u < 1700$
collect $\{\text{year}{:}u\}$:C



{year:1685}:C

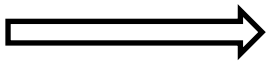
Well-Formed Query

- None of the patterns p_i is a single variable
- Each expression e_i:
  - Nested query
  - Expression that does not involve a query
- Comparisons
  - Between variables
  - Between variables and constants

Singular Expression

- Expression e is singular if $e \neq (e_1 \sqcup e_2)$
- Variables may only bind to singular values
  - They cannot bind subtrees

where $\text{Composers}.x : y \in \text{D},$
$\text{born}.u \in y,$
$u < 1700$
collect $x : y$



{{name:"J. S. Bach"}.born:1685,
{name:"G. F. Handel"}.born:1685}

Normal Form

- Q has the form $Q_1 \sqcup Q_2 \sqcup \ldots \sqcup Q_m$
- Each query Q_i has the form:

$$\begin{array}{ll} \text{where} & sp_1 \in D_1, \\ & \vdots \\ & sp_n \in D_n \\ & condition \\ \text{collect} & se \end{array}$$

Every well-formed query has an equivalent normal form!

Why-Provenance

Witness

- Idea
  - The structure of the why-provenance is related to finding the proof for the output
  - The collection of values that proves an output is called a witness for the output
  - *s* is a witness for *t*
    - $t \sqsubseteq Q(s)$
    - $s \sqsubseteq D$

Witness Basis

- $W_{Q,D}(t) = \{[\![p_0]\!]_\psi \sqcup \ldots \sqcup [\![p_n]\!]_\psi \mid \psi \in \Psi, t = [\![e]\!]_\psi\}$
  - Q is in normal form

Algorithm: Why(t, Q_i, D)

Let Δ denote the "where" clause of Q_i.
Let Δ' denote the deep union of patterns in Δ.
if there is a valuation ψ from e_i to t **then**
  Return the query "where $\psi(\Delta)$ collect $\psi(\Delta')$:C"
  (For simplicity, we did not serialize the output expression on the edge.)
else
  No query is returned
end if

Minimal Witness Basis

- $M_{Q,D}(t)$
  - Maximal subset of $W_{Q,D}(t)$
    - $\forall m \in M_{Q,D}(t)$, $\nexists w \in W_{Q,D}(t)$ such that $w \sqsubset m$

- Invariant under equivalent queries

Where-Provenance

Main Idea

- Connected to the witness idea
- For a specific value in the output
  - Determine output variable that was bound to the value
  - Identify pieces of input data that were bound to the output variable
- Difficulties in formalizing that…

Difficulties

- Output variable

$$Q_1 = \text{where } \text{Emps.}\{\text{Id}{:}x\}.\text{salary}{:}\$50\text{K} \in \text{D}, \\ \text{collect } \{\text{Id}{:}x\}.\text{salary}{:}\$50\text{K}$$

$$Q_2 = \text{where } \text{Emps.}\{\text{Id}{:}x\}.\text{salary}{:}y \in \text{D}, \\ y = \$50K \\ \text{collect } \{\text{Id}{:}x\}.\text{salary}{:}y$$

Difficulties

- Multiple pieces of data

$$
\begin{aligned}
Q_3 = \text{where}\ & \text{Emps.}\{\text{Id}{:}x\}.\text{salary}{:}y \in \text{D}, \\
& \text{Emps.}\{\text{Id}{:}x\}.\text{bonus}{:}y \in \text{D} \\
\text{collect}\ & \{\text{Id}{:}x\}.\text{new\_salary}{:}y
\end{aligned}
$$

$$
\begin{aligned}
Q_4 = \text{where}\ & \text{Emps.}\{\text{Id}{:}x\}.\text{salary}{:}y \in \text{D}, \\
& \text{Emps.}\{\text{Id}{:}x\}.\text{salary}{:}z \in \text{D}, \\
& \text{Emps.}\{\text{Id}{:}x\}.\text{bonus}{:}z \in \text{D} \\
\text{collect}\ & \{\text{Id}{:}x\}.\text{new\_salary}{:}\text{y}
\end{aligned}
$$

Difficulties

- Nested queries

$$Q_5 = \text{where } \mathrm{R}.x.y : z \in \mathrm{D},\ \mathrm{S}.x.y : z \in \mathrm{D} \text{ collect } x.y : z$$

$$Q_6 = \text{where } \mathrm{R}.x.y : z \in \mathrm{D},\ \mathrm{S}.t.u \in \mathrm{D},\ t : u \in (\text{where } \mathrm{R}.x.y : z \in \mathrm{D} \text{ collect } x.y : z), \text{ collect } \{x.y : z, t : u\}$$

Solution

- Still use the syntactic approach
- Restricted class of queries: traceable queries
  - Invariants under rewriting
  - Use of normal form
- Derivation Basis
  - Systematically explore the pieces of input data contributing to the output variable

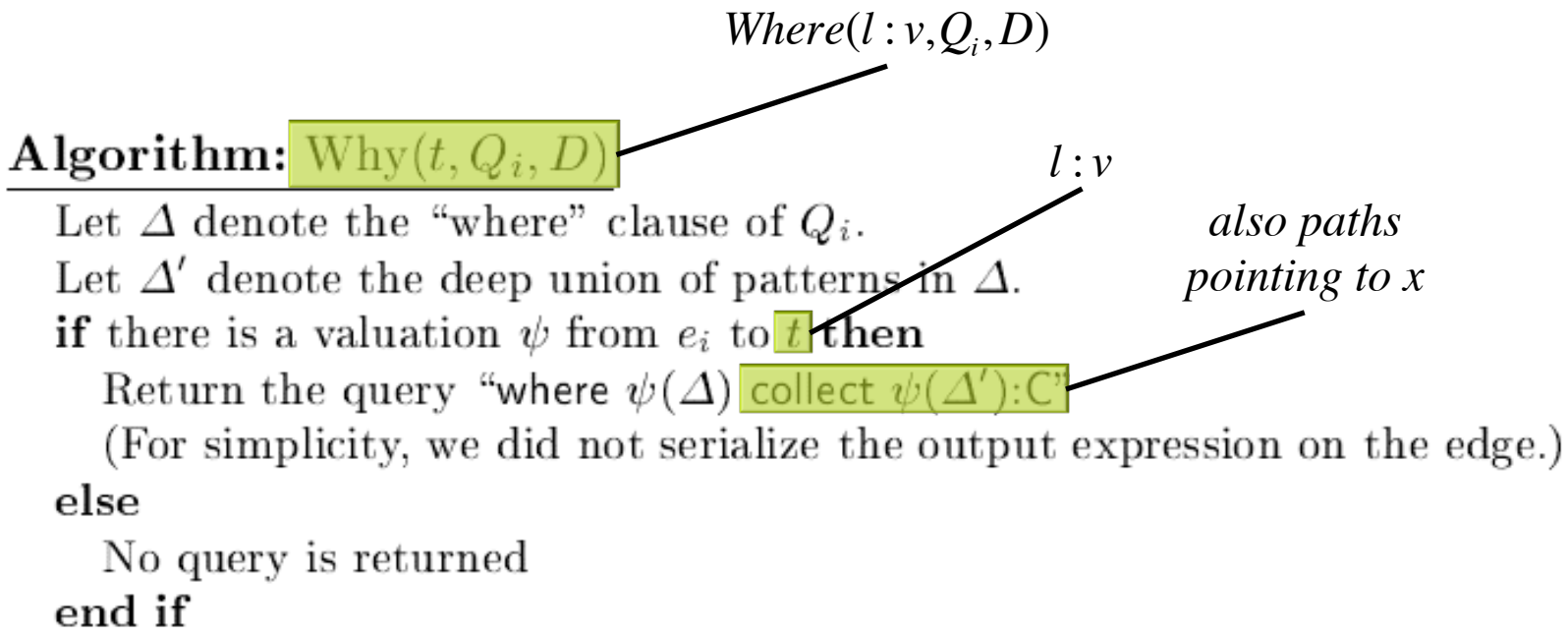
Traceable Queries

- A well-defined query Q is traceable if:
  - Each pattern matches either against a database constant or against a subquery
  - Every subquery in Q is a view, without sharing any variables in the outer scope
  - Only a singular pattern is allowed to match against a subquery
  - Pattern and output expression of the subquery consist of a sequence of distinct variables and have the same length

Traceable Queries

$$\begin{array}{l}\textsf{where}\ \ x : u \in D, \\ \qquad\qquad y : z \in \left(\begin{array}{l}\textsf{where}\ \ u : v \in D \\ \textsf{collect}\ u : v\end{array}\right), \\ \textsf{collect}\ x : z\end{array}$$

Derivation Basis



Conclusions

Conclusions

- A framework for describing and understanding provenance was presented
  - Why-Provenance
  - Where-Provenance
- Use of syntactic approach
  - Deterministic model

Oracle Total Recall

Oracle Database 11g Release 2

Provenance!

- Keep track of the contextual history of the data
  - Versions of the database
- Query against this history
- Flashback Data Archive (FDA)
  - Part of Oracle Flashback Technology
  - Feature responsible for storing the history

Versions of the Database

- When you commit a transaction, you have a new version of the database
  - Each version has a timestamp
  - There is a possible 3-second error in the timestamp
  - Precise time comes with the SCN (System Change Number)
    - SCNs are used internally
- New versions for both DML and some DDL operations

History Tables

- Creation of *history tables* (or *archive tables*)
  - One history table for each tracked table
- When one or ore columns are updated
  - A new row is inserted in the history table
  - It works as a before-image of the row before the transaction
- Undo records are used
  - Transactions and undo records are marked for archival
  - Undo records are only recycled after successfully archiving every transactions

Performance

- History table is internally partitioned
- Compression to reduce disk space requirements
- Multiple FDA processes during archiving
- Tests using TPC-C
  - Average response time impact: 5%

Step-by-Step Approach

- Either create a new tablespace or use an existing one
  - CREATE TABLESPACE tbs1DATAFILE ‘test.dbf’
    SIZE 40M ONLINE
- Create the flashback data archive for the tablespace
  - CREATE FLASHBACK ARCHIVE fda1
    TABLESPACE tbs1
    RETENTION 1 year
- Enable it on the desired tables
  - ALTER TABLE EMPLOYEE FLASHBACK ARCHIVE fda1

Querying

```
SELECT last_name, first_name, salary
FROM EMPLOYEES
AS OF TIMESTAMP TO_TIMESTAMP('2007-06-01 00:00:00','YYYY-MM-DD HH24:MI:SS')
WHERE employee_id=193;
```

```
SELECT last_name, first_name, salary
FROM EMPLOYEES
VERSIONS BETWEEN TIMESTAMP
TO_TIMESTAMP('2007-06-01 00:00:00','YYYY-MM-DD HH24:MI:SS') AND
TO_TIMESTAMP('2009-06-01 00:00:00','YYYY-MM-DD HH24:MI:SS')
WHERE employee_id=193;
```

Thank you!