Data Integration: Provenance

Jan Chomicki

University at Buffalo
Annotations recording how a tuple in the query result was produced from the database.

Different kinds

• **Why-provenance (lineage):** return a relevant part of the database
• **How-provenance:** keeping track of individual derivations
• **Where-provenance:** keeping track of individual attribute values
Why-provenance

Query language
Relational algebra:
- without set difference (positive RA)
- without renaming (for simplicity)
- with constant singleton relations: \{u\}

Notation
- \(Q\): query
- \(t\): tuple
- \(R_1, \ldots, R_k\): relation names
- \(D\): database instance consisting of relation instances \(r_1, \ldots, r_k\)
- \(S \uplus T = \{S \cup T \mid S \in S \land T \in T\}\)
Why-provenance: definition

Tuple annotations: sets of sets of facts.

Definition

\[
\text{Why}(\{u\}, D, t) = \begin{cases} 
\emptyset & \text{if } t = u \\
\emptyset & \text{otherwise}
\end{cases}
\]

\[
\text{Why}(R_i, D, t) = \begin{cases} 
\{\{R_i(t)\}\} & \text{if } t \in r_i \\
\emptyset & \text{otherwise}
\end{cases}
\]

\[
\text{Why}(\sigma_C(Q), D, t) = \begin{cases} 
\text{Why}(Q, D, t) & \text{if } t \text{ satisfies } C \\
\emptyset & \text{otherwise}
\end{cases}
\]

\[
\text{Why}(\pi_X(Q), D, t) = \bigcup \{ \text{Why}(Q, D, u) \mid u \in Q(D) \land t = u[X] \}
\]

\[
\text{Why}(Q_1 \cup Q_2, D, t) = \text{Why}(Q_1, D, t) \cup \text{Why}(Q_2, D, t)
\]

\[
\text{Why}(Q_1 \bowtie Q_2, D, t) = \text{Why}(Q_1, D, t[U_1]) \uplus \text{Why}(Q_2, D, t[U_2])
\]
Why-provenance: properties

Empty provenance

If $\text{Why}(Q, D, t) = \emptyset$, then $t \notin Q(D)$.

Nonempty provenance

If $J \in \text{Why}(Q, D, t)$, then $J \subseteq D$ and $t \in Q(J)$. 
Tuple annotations: values from a special domain $\mathcal{K}$.

The properties of $\mathcal{K}$

$\mathcal{K} = (K, 0, 1, +, \cdot)$ is a commutative semiring:

- Addition (+) is associative, commutative and has identity 0
- Multiplication (⋅) is associative, commutative and has identity 1
- For all $x$: $x \cdot 0 = 0 \cdot x = 0$
- Multiplication distributes over addition.

Examples of $\mathcal{K}$

- **Booleans**: relations as sets
- **natural numbers**: relations as bags
- **polynomials**: how-provenance
How-provenance

Tuple annotations

- tuples not in the database: 0
- tuples in the database: tuple identifiers
- tuples in a query result: polynomial expressions encoding tuple derivations

Definition

\[
\begin{align*}
\text{How}(\{u\}, D, t) & = \begin{cases} 
1 & \text{if } t = u \\
0 & \text{otherwise}
\end{cases} \\
\text{How}(R_i, D, t) & = \begin{cases} 
V & \text{if } t \in r_i \text{ with annotation } V \\
0 & \text{otherwise}
\end{cases} \\
\text{How}(\sigma_C(Q), D, t) & = \begin{cases} 
\text{How}(Q, D, t) & \text{if } t \text{ satisfies } C \\
0 & \text{otherwise}
\end{cases} \\
\text{How}(\pi_X(Q), D, t) & = \sum \{ \text{How}(Q, D, u) \mid \text{How}(Q, D, u) \neq 0 \land t = u[X] \}
\end{align*}
\]
Union

\[ \text{How}(Q_1 \cup Q_2, D, t) = \text{How}(Q_1, D, t) + \text{How}(Q_2, D, t) \]

Join

\[ \text{How}(Q_1 \bowtie Q_2, D, t) = \text{How}(Q_1, D, t[U_1]) \cdot \text{How}(Q_2, D, t[U_2]) \]

Recovering Why-provenance

\[ \mathcal{K} = (\mathcal{P}(\mathcal{P}(\text{Facts})), \emptyset, \{\emptyset\}, \cup, \sqcup) \]

where \( \mathcal{P} \) is the powerset operator and \( \text{Facts} \) is the set of all facts.