Semantic Information Integration

*Schema Mappings, Data Exchange, and Metadata Management*
Phokion Kolaitis, 2005

*Data Integration: A Theoretical Perspective*
Maurizio Lenzerini, 2002

Presented by:
Paea LePendu
paea@cs.uoregon.edu
Outline

➲ Introduction & Motivation
➲ Important Contributions
➲ Background
➲ An Example
➲ Challenges
➲ Active research areas
Integration \supseteq Data Exchange \supseteq Query Answering

Data Federation
main task: query answering

Data Migration
main task: data translation
Integration Scenarios

Integration depends on your task perspective.
- A acquires B
- A gives to B
- A and B share with C
- A and B cooperate

What is the Source? Target?

What is the goal? (Query, Update)

How often? (Once, Regularly, Rarely)
Outline

➲ Introduction & Motivation
➲ Important Contributions
➲ Background
➲ An Example
➲ Challenges
➲ Active research areas
Contributions (Lenzerini)

- Formalizing Integration Systems
  - $I = \langle G, S, M \rangle$
  - *certain answers vs. possible answers*

- Logical Foundation
  - Soundness
  - Completeness

- Global as View vs. Local as View

- Query Processing

- Inconsistency

- Reasoning

Introduction → Motivation → Contributions → Theoretical Grounding → Future Work
Contributions (Kolaitis)

✸ Separate exchange
  ● A specific sub-problem
  ● Unique challenges

✸ Constraint satisfaction
  ● Tuple generating dependencies
  ● Universal solutions

✸ Complexity analysis
  ● Decidability
  ● Queries, Composition
The difference is in MATERIALIZATION!

**Data Federation**
virtual
requirement: on-the-fly, fast

**Data Migration**
materialized
requirement: stable, consistent
Global as View
Source: P. Atzeni [ICDE 2006]

GAV, in integration

CREATE VIEW R1
AS SELECT ...
FROM R11, R12

---

R1, R2

---

R11, ...

---

R12, ...
CREATE VIEW R11
AS SELECT ...
FROM R1, R2
<table>
<thead>
<tr>
<th>GAV</th>
<th>vs.</th>
<th>LAV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not modular</td>
<td>Modular--adding new sources is easy</td>
<td></td>
</tr>
<tr>
<td>- Addition of new sources changes the mediated schema</td>
<td></td>
<td>Very flexible--power of the entire query language available to describe sources</td>
</tr>
<tr>
<td>Can be awkward to write mediated schema without loss of information</td>
<td></td>
<td>Reformulation is hard</td>
</tr>
<tr>
<td>Query reformulation easy</td>
<td>- Involves answering queries only using views (can be intractable—see below)</td>
<td></td>
</tr>
<tr>
<td>- reduces to view unfolding (polynomial)</td>
<td></td>
<td>Best when</td>
</tr>
<tr>
<td>- Can build hierarchies of mediated schemas</td>
<td></td>
<td>- Many, relatively unknown data sources</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- possibility of addition/deletion of sources</td>
</tr>
<tr>
<td>Best when</td>
<td></td>
<td>- Information Manifold, InfoMaster, Emerac, Havasu</td>
</tr>
<tr>
<td>- Few, stable, data sources</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- well-known to the mediator (e.g. corporate integration)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Garlic, TSIMMIS, HERMES</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Formal Modeling

I = \langle G, S, M \rangle

Mappings (M)

\begin{align*}
qs & \sim qG, \\
qG & \sim qS
\end{align*}

Certain Answers

Sound

\begin{align*}
& s^D \subseteq qG^B \\
& \forall x \ s(x) \rightarrow qG(x)
\end{align*}

Possible Answers

Complete

\begin{align*}
& s^D \supseteq qG^B \\
& \forall x \ qG(x) \rightarrow s(x)
\end{align*}
Constraints as Dependency Graph

\[ H(x_1, x_2) \rightarrow \exists y H(x_2, y) \]

\[ H(x_1, x_2) \rightarrow \exists y H(x_1, y) \]

\[ H(x_1, x_2) \rightarrow \exists y H(y, x_2) \]

\[ H(x_1, x_2) \rightarrow \exists y H(y, x_2) \]

Outline

- Introduction & Motivation
- Important Contributions
- Background
- An Example
- Challenges
- Active research areas
What is a view?

- Datalog
- SQL
Datalog

➾ GAV:: \( R1(x,y,...) \) :- \( R11(x,y,...) \land R12(x,y,...) \)

➾ LAV:: \( R11(x,y,...) \) :- \( R1(x,y,...) \land R2(x,y,...) \)
SQL

- **GAV**:: \( R_1 \) AS SELECT \( x, y, \ldots \) FROM \( R_{11}, R_{12} \)
- **LAV**:: \( R_{11} \) AS SELECT \( x, y, \ldots \) FROM \( R_1, R_2 \)
Is a view a logical implication?

◮ GAV:: \[ R_1(x,y,...) :- R_{11}(x,y,...) \land R_{12}(x,y,...) \]

\[ \text{e.g., } R_{11}(x,y,...) \land R_{12}(x,y,...) \rightarrow R_1(x,y,...) \]

◮ LAV:: \[ R_{11}(x,y,...) :- R_1(x,y,...) \land R_2(x,y,...) \]

\[ \text{e.g., } R_1(x,y,...) \land R_2(x,y,...) \rightarrow R_{11}(x,y,...) \]
So what does it really mean?

I = \langle G, S, M \rangle

Mappings (M)

\[ qs \sim qG, \quad qG \sim qS \]

Certain Answers

Sound

\[ s^D \subseteq q_B^G \]

\[ \forall x \ s(x) \rightarrow q_G(x) \]

Possible Answers

Complete

\[ s^D \supseteq q_B^G \]

\[ \forall x \ q_G(x) \rightarrow s(x) \]
Outline

- Introduction & Motivation
- Important Contributions
- Background
- An Example
- Challenges
- Active research areas
What is a data warehouse?

Source-1, Source-2 → Target

rdb1, rdb2 → dw1

(note: color coding guide = source, target)
The Source

Source Database (rdb1)

Fig. 3. Relational schema rdb1
The Target

Target Database (data warehouse - dw1)

Fig. 4. Star schema of data warehouse dw1
Goal: source $\rightarrow$ target
Fig. 5. The structure of $map_1$. Dotted lines are containment relationships. Solid lines are relationships to the domain and range of mapping objects.
Mappings – Global as View

create view dw1.Products (ProductID, ProductName, BrandID, BrandDescription) as
select P.ProductID, P.ProductName, B.BrandID, B.BrandDescription
from rdb1.Brands B, rdb1.Products P
where B.BrandID=P.BrandID
Outline

➲ Introduction & Motivation
➲ Important Contributions
➲ Background
➲ An Example
➲ Challenges
➲ Active research areas
Consistency and Constraints

- Mappings are not the end of the story!
- Redundancy
  - Object reconciliation -- “are X and Y the same?”
- Conflicts and inconsistency
  - “as sound as possible”
  - Data Lineage
  - Majority consensus
- Constraint satisfaction
  - Tuple generating (multi-valued) dependencies
  - Equality generating dependencies
  - Universal solutions, cores, weakly acyclic tdgs
Discussion

- Formalizing Integration Systems
  - \( I = \langle G, S, M \rangle \)
  - certain answers vs. possible answers

- Logical Foundation
  - Soundness, Completeness

- GAV vs. LAV

- Query Processing

- Inconsistency, Constraints

- Reasoning vs. Rewriting vs. Unfolding
Consistency and Constraints

➲ Mappings are not the end of the story!

➲ Redundancy
  ● Object reconciliation -- “are these really the same?”

➲ Conflicts and inconsistency
  ● “as sound as possible”
  ● Data Lineage
  ● Majority consensus

➲ Constraint satisfaction
  ● Tuple generating (multi-valued) dependencies
  ● Equality generating dependencies
  ● Universal solutions
Outline

- Introduction & Motivation
- Important Contributions
- Background
- An Example
- Challenges
- Active research areas
Active Research

➲ The use of Ontologies in Integration
➲ Object reconciliation
➲ Consistency and Completeness
➲ Semantics

Introduction → Motivation → Contributions → Theoretical Grounding → Future Work
Thank you.
References

➲ M. Lenzerini. Data integration: A theoretical perspective. [PODS 2002]

➲ P. Kolaitis. Schema Mappings, Data Exchange, and Metadata Management. [PODS 2005]

➲ P. A. Bernstein and E. Rahm. Data warehouse scenarios for model management. [ER 2000]