Data Integration
in a Networked World

Karl Aberer
EPFL
karl.aberer@epfl.ch
http://lsir.epfl.ch/
http://www.mics.ch/
Overview

• Motivation: Semantic Interoperability
• Peer-to-peer Data Integration
  – Mapping Discovery
  – Semantics of Query Answering
  – Mapping Inconsistency
• Philosophical Excursion on Semantics
• Inconsistency in P2P Data Integration
  – Repairing Mappings
  – Consistent Mapping Discovery
• Whom to trust?
• Conclusions
Motivation

Semantics
Semantic Interoperability

• Semantic interoperability of information systems
  – has been a key data management problem
    .... and is continuously gaining in importance
• Examples
  – Business Data
  – Scientific Data
  – Personal Data
  – Web Data and Services

One of the major barriers to efficiency in financial networks is the staggering cost of integrating computer systems. Standards help, but the finance sector has multiple standards bodies, and each has a history of governing a silo. (David S. Frankel, SAP Labs, 2009)

World Wide Annual Integration plus Data Quality Costs: $1 Trillion / year “The problem is not in the plumbing. It’s in the semantics” (Michael Brodie invited talk at ISWC 2003)
Data Integration

- Is one approach to achieve semantic interoperability
  - Alternatives: standards, ontologies
- **Matching** and **mapping** of data structures
  - Based on structural and content features
  - Implemented, e.g., by schema matchers

\[ V_{\text{Client}}(\text{Member}) = \text{SELECT first + " " + last AS name FROM Member} \]
Schema Matchers

- Experimental evaluation
  - SAP business dataset
  - 3 schemas
- Matcher: AutoMappingCore, COMA

Typically 60% - 70% precision and recall
Global Schemas

**Global-as-View (GAV)**
- Person = $V_{\text{Person}}(\text{Client, Member, Partner, User, Contacts})$
- Approach does not scale: change in any local database affects global view

**Local-as-View (LAV)**
- Client = $V_{\text{client}}(\text{Person})$
- User = $V_{\text{user}}(\text{Person})$ etc.
- Complex query processing
- Approach does not scale: change in global schema affects all local views

Global schema needs to be kept up-to-date with all databases and serve all application needs.
Peer-to-peer to the rescue!

• Peer-to-peer data integration: no global schema
  – Nodes maintain few links locally
  – Network of matchings/mappings
  – Exploit transitive mapping composition
  – Queries can be issued at any node
Peer-to-peer, does it exist?

- Yes [Budura et al, FGCS 2007]
- SRS Bioinformatic library
  - Indexing and retrieval system
  - EMBL, SwissProt, Prosite, etc.
  - Semantic Graph: Links from one database to others by mapping identifiers
  - More than 380 databases and 500 (undirected) links
  - [http://srs.ebi.ac.uk/](http://srs.ebi.ac.uk/)
Structure of Semantic Graph

- Giant connected component: 187 nodes (out of 380)
- Powerlaw Topology
- Small-World Graph
  - Clustering coefficient = 0.32
  - Diameter = 9
Linked Open Data
And, yes, this looks more like P2P
The Network is the Business

• EU FP 7 project, 2010 - 2012
• Coordinated by SAP Research Israel
• Network approach to enterprise interoperability
  – Sharing and reuse of interoperability information
  – Peer-to-peer architecture
• For more information: http://www.nisb-project.eu
Overview

• Motivation: Semantic Interoperability
• Peer-to-peer Data Integration
  – Mapping Discovery
  – Semantics of Query Answering
  – Mapping Inconsistency
• Philosophical Excursion on Semantics
• Inconsistency in P2P Data Integration
  – Repairing Mappings
  – Consistent Mapping Discovery
• Whom to trust?
• Conclusions
Peer-to-peer Data Integration

Courtesy W. Litwin
Technical Problems

1. Semantics of Query Answering
2. Mapping Consistency
3. Mapping Discovery

http://www.morganclaypool.com/doi/abs/10.2200/S00338ED1V01Y201104DTM015
(free download 😊)
1. Mapping Discovery

• Use network to discover new mappings
• Adapt schema matching techniques to distributed settings
  – Attribute and relation annotations
  – Database probing
• Area still quite unexplored
2. Semantics of Query Answering

• Three approaches
  – Naïve operational semantics
  – Logic-based semantics
  – Incomplete mappings
Naïve Query Evaluation

- GAV mappings
  - Querying node considers it’s schema as global schema
- Recursively rewrite query
- Don’t visit nodes twice

```
Q(Client) ➔ Q(V_{client}(Partner)) ➔ Q(V_{client}(V_{partner}(Member)))
```
Logic-based Semantics

- Supports both GAV and LAV type of mappings among schemata [Halevy et al, ICDE 2003]
  - Mappings considered as global inter-schema constraints
  - Requires strong restrictions for tractable query processing
Incomplete Mappings

• Mappings might be incomplete (missing attributes)
  – Still partially translating queries might make sense
  – Projection: return NULL values
  – Selection: don’t consider conditions

• Measure the information loss and stop forwarding when above threshold [Aberer et al, WWW 2003]

  \[ \text{Projection similarity: } \operatorname{sim}_A(Q, Q_m) = \frac{\hat{w} \text{ReLU} \delta(A(Q))}{|\hat{w}|_2 |\delta(A(Q))|_2} \text{ where } \delta_i(A(Q)) = 1 \text{ if attribute } a_i(Q) \text{ could be mapped to an attribute in } Q_m \text{ and } w_i \text{ is the relative importance of attribute } a_i(Q) \text{ for the original query.} \]

  \[ \text{Selection similarity: } \operatorname{sim}_C(Q, Q_m) = \frac{\hat{w} \text{ReLU} \delta(C(Q))}{|\hat{w}|_2 |\delta(C(Q))|_2} \text{ where } \delta_i(C(Q)) = \text{selectivity}(C_i(Q)) \text{ if condition } c_i(Q) \text{ could not be mapped to a condition in } Q_m \text{ and } \delta_i(C(Q)) = 1, \text{ otherwise.} \]

  \text{selectivity}(C_i(Q)) \text{ is the selectivity of the condition for the relation to which it is applied.} \\
  w_i \text{ is the relative importance of condition } c_i(Q) \text{ for the original query.} \]
3. Inconsistent Mappings

- Big Problem: How to avoid/deal with inconsistency?
  - Mappers are unreliable
  - Peers are unreliable
  - Mapping languages are not sufficiently expressive
Problem is real!

- From SAP dataset

PostalAddress->Addresses->Address->Address
Path1: AdditionalTaxNumber->Tnumber
Path2: AdditionalTaxNumber->TaxNumbers
What is the impact?

• Naïve Evaluation
  – If we are lucky, the result is \{Anne, Bob, Charles, Zoltan, \ldots\}
  – If we are unlucky, the result is \{Anne, EPFL, SAP, Zoltan, \ldots\}

• Logics-based approach
  – We will be always unlucky, the result is
    \{Anne, Bob, Charles, EPFL, SAP, Zoltan, \ldots\}

• Any ideas what to do?
Overview

• Motivation: Semantic Interoperability
• Peer-to-peer Data Integration
  – Mapping Discovery
  – Semantics of Query Answering
  – Mapping Inconsistency
• Philosophical Excursion on Semantics
• Inconsistency in P2P Data Integration
  – Repairing Mappings
  – Consistent Mapping Discovery
• Whom to trust?
• Conclusions
Philosophical Excursion on Semantics
What is semantics?

• Standard response:
  “Mapping of a syntactic structure into a semantic domain”

Syntactic structure: database, knowledge base

Semantic domain: real-world
Semantic Web

• Real-world is a somewhat ill-defined and hard to compute concept
• Proposal: Substitute real-world by shared formal conceptualization [Gruber 93]

Syntactic structure: database, knowledge base
Semantic domain: ontology
The Issue with Ontologies

• What is the semantics of ontologies? After all, they are syntactic structures!
• Heterogeneous and evolving models and ontologies
The Correspondence Continuum

• Observation: Meaning is rarely a simple mapping from a syntactic structure to a semantic domain
• Continuum of (semantic) correspondences from symbol to (symbol to)* object [Smith 87]
• The meaning of a symbol is given by the composition of the semantic mappings that relate it to its root
Semantic Grounding

• Still: what should “relate to the root mean”?  
• The meaning of symbols can be explained by its semantic correspondences to other symbols alone (“Understanding understanding”) [Rapaport 93]

• Type 1 semantics  
  – understanding in terms of something else  
  – Problem: how to ground semantics?

• Type 2 semantics  
  – understanding something in terms of itself  
  – “syntactic semantics”: grounding through recursive understanding
Emergent Semantics

- Semantic correspondences form rich self-referential structures: recursive understanding
- Recursive understanding may converge against stable structures (fixpoints): *emergent semantics* [Aberer et al 2004]
Practical Conclusion

- If we have a mapping network with inconsistency, try to find a recursive procedure that eliminates inconsistency.
- In other words, establish a semantic agreement... by eliminating semantic disagreement.

= understanding
Overview

• Motivation: Semantic Interoperability
• Peer-to-peer Data Integration
  – Mapping Discovery
  – Semantics of Query Answering
  – Mapping Inconsistency
• Philosophical Excursion on Semantics
• Inconsistency in P2P Data Integration
  – Repairing Mappings
  – Consistent Mapping Discovery
• Whom to trust?
• Conclusions
Inconsistency in P2P Data Integration
Repairing Mappings

• Assume mappings are given
• Identify incorrect mappings
  – Feedback from query processing
• Types of feedback [Aberer et al, WWW 2003]
  – Query results: preservation of data dependencies, content similarity -> similar to mapping discovery
  – Transformed queries: transitive closures of mapping operations resulting in cycles and parallel paths
• Abductive reasoning: find best possible explanation in case of inconsistency
Estimating Probability of Correctness

• The result of applying a two equivalent composite mappings to a query should be identical to the original query
  – Allows to estimate the probability of correctness of

\[
Pr(f = 1|m_1, \ldots, m_k) = \begin{cases} 
1, & \text{if } m_1, \ldots, m_k \text{ are correct} \\
0, & \text{if exactly one element in } m_1, \ldots, m_k \text{ is incorrect} \\
\Delta, & \text{if several elements in } m_1, \ldots, m_k \text{ are incorrect}
\end{cases}
\]

\[m_1: \text{name} \rightarrow \text{companyname}\]
\[m_2: \text{name} \rightarrow \text{pname}\]
\[m_3: \text{companyname} \rightarrow \text{cname}\]
\[f = (\text{cname} = \text{pname}) = \text{FALSE}\]
Computing a Marginal Probability for one Mapping

- **Goal:** determine $P(m_i \mid f)$

\[
P(m_1 \mid f) = \sum_{m_2m_3} P(m_1,m_2,m_3,f)P(f)^{-1} = \sum_{m_2m_3} P(m_1)P(m_2)P(m_3)P(f \mid m_1,m_2,m_3)P(f)^{-1}
\]

(a priori probability) (known)

- **Maximum entropy principle:** a priori probability $P(m_i) = 0.5$

- **Bayes Theorem**

\[
P(m \mid f) = \frac{P(m)P(f \mid m)}{P(f)}
\]
Correlated Feedback

• But: feedbacks on different cycles are **correlated**
  – One wrong mapping will affect several cycles/paths
  – Need to express a **global** probabilistic model for the mapping graph:
    \[ P(m_1, ..., m_5 \mid f_1, ..., f_5) \]

• How to compute \( P(m_1 \mid f_1, ..., f_5) \)?

• Graphical representation of the problem
  [Cudre-Mauroux et al, ICDE 2006]
Deriving PDMS Factor-Graph from a P2P network
Solving PDMS Factor-Graphs

• How to infer probabilistic correctness?
  – Junction Tree
    • Centralized approach
    • Computational + communication overhead
  – Iterative Sum-Product by message passing
    • Correct only for tree structured networks
    • Approximate result for general graphs
Message Passing on Factor Graphs

- Given \( P(m_1, ..., m_4 \mid f_1, f_2) \), compute \( P(m_2 \mid f_1, f_2) \)

\[
\mu_{f_1 \rightarrow m_2}(m_2) = \sum_{m_1} P(m_1, m_2 \mid f_1)
\]

\[
\mu_{f_2 \rightarrow m_2}(m_2) = \sum_{m_3, m_4} P(m_2, m_3, m_4 \mid f_2)
\]

\[
P(m_1, m_2, m_3, m_4 \mid f_1, f_2) = P(m_1, m_2 \mid f_1)P(m_2, m_3, m_4 \mid f_2)
\]

\[
P(m_2) = \sum_{m_1,m_3,m_4} P(m_1, m_2, m_3, m_4 \mid f_1, f_2) = \sum_{m_1} P(m_1, m_2 \mid f_1) \sum_{m_3, m_4} P(m_2, m_3, m_4 \mid f_2)
\]

Two “factors”:
- computed independently
- sent as “message” to \( m_2 \)
Message Passing on Factor Graphs (2)

\[ P(m_1, m_2, m_3, m_4, m_5 \mid f_1, f_2, f_3) = P(m_1, m_2 \mid f_1) P(m_2, m_3, m_4 \mid f_2) P(m_4, m_5 \mid f_3) \]
Message Passing on Factor Graphs

- General Formulation
  - $n()$ denotes neighbors in factor graph

\[
\mu_{x \rightarrow f}(x) = \prod_{h \in n(x) \setminus \{f\}} \mu_{h \rightarrow x}(x)
\]

- Works correctly on trees
Distributed Implementation

- Message-Passing Schedules
  - Lazy (piggybacking on query forwarding)
    - No message overhead
  - Periodic
Evaluation: Convergence

(undirected example graph, prior 0.7 delta 0.1)
Consistent Mapping Discovery

• Previous approach
  – discovering mappings -> repairing mappings
  – does not take advantage of matching values from schema matchers

• Mapping discovery by exploiting schema matchers and considering violations of consistency

Matching Graph
Consistent Correspondence Graph

- Integrity constraints
  - No inconsistent cycles and parallel paths
  - No violation of triangle inequality by weights

Matching Graph

Instance – $I_0$
Selected highest ranked attribute correspondences

Instance $I$
Selected attributes to obtain consistent correspondence graph
Optimization Problem

• Given model $M$, integrity constraints $\Sigma$, optimization function $F$, find the instance $I_{opt}$ that is minimal-repair instance of top-rank instance $I_0$ and optimal w.r.t $F$.

• $F = \#\text{correct circles} + \#\text{correct parallel paths}$
Heuristics

Initialization
- Read all attribute correspondences
- Generate top-rank instance $I_0$

Pre-processing
(Transitive closure)
- Satisfy triangle inequality

Repair
(only deletion)
- Initialize an consistent instance $I_{min}$
- Find upper bound of repair distance

Optimization
(Exploit neighbor instance with deletion and insertion)
- Update upper bound
- Find minimal-repair instance with maximal number of correct circles and parallel paths

$I_0 \rightarrow I_{min} \rightarrow I_5 \rightarrow I_6 = I_{op}$

Optimal value
Overview

• Motivation: Semantic Interoperability
• Peer-to-peer Data Integration
  – Mapping Discovery
  – Semantics of Query Answering
  – Mapping Inconsistency
• Philosophical Excursion on Semantics
• Inconsistency in P2P Data Integration
  – Repairing Mappings
  – Consistent Mapping Discovery
• Whom to trust?
• Conclusions
Whom to trust?
Which mapping to trust?

• So far: tried to get rid of “bad” mappings/matches
• If we knew about who generates those mappings, the task might become simpler
• Trusted Entity Resolution [Cudre-Mauroux et al WWW 2009]
  – Entities identified in Web contents
  – Equivalence/Non-equivalence among entities stated but different sources
    • Algorithms, human users
    • Trusted or untrusted
Trusted Entity Resolution

- Idea: use consistency of entity resolution to
  - Improve entity matching
  - Assess trust in sources providing entity equivalence
Probabilistic Inference

Source Graph

Entity Graph

Reputation-Based Trust Management

Trust Values for Sources

Trust Constraint

Initial Link Values

Combined Value Functions / Priors for Links

Inferred Link Values

Graph Constraints
Result

Figure 9: Inference accuracy for networks of 50 entities, 150 links, 50 sources each declaring values for 1-10 links, and an initial population of legitimate sources gradually replaced by matchers or spammers.

Matchers predict links with 90% accuracy, spammers with 0%
Why to agree?

• Signaling approach
  – Non-strategic spamming behavior is detected
• Strategic behavior: what is the incentive
  – to share mappings with others?
  – to agree with others on consistent mappings?
• Economic or game-theoretic framework needed
• Ongoing work ...
Conclusion
Conclusion

• Peer-to-peer data integration
  – Promises to address scalability, but also trust and economic issues
  – Maps much better reality in terms of semantic interoperability

• Mappings
  – Are always approximate and uncertain
  – Life in a network context
  – Should be consistent as possible
  – Are an asset with value and privacy
References


