Carleton
UNIVERSITY

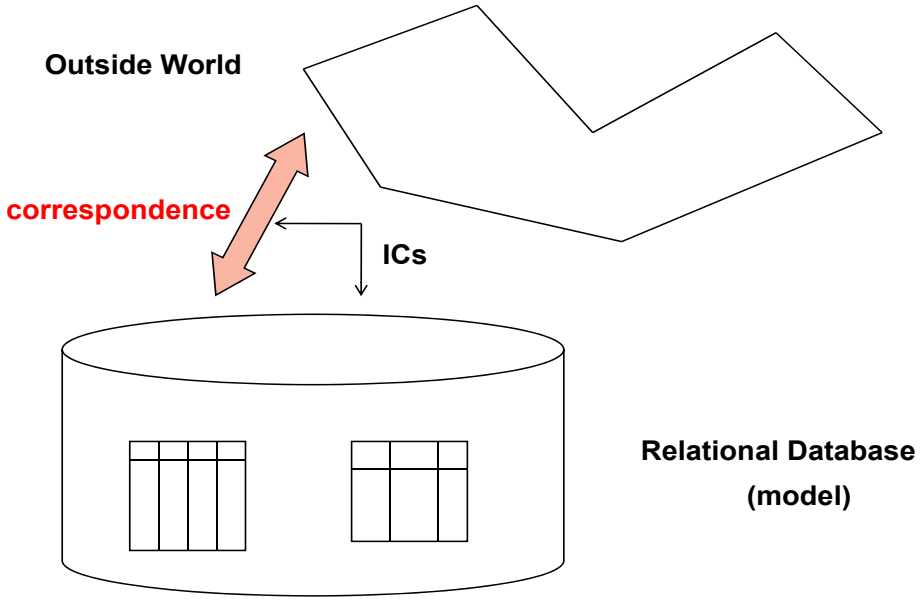
Dealing with Inconsistent Data Sources

Scientific and Technical Challenges

Leopoldo Bertossi

Carleton University
School of Computer Science
Ottawa, Canada

Consistency of Databases



A database instance D is a model of an outside reality

An integrity constraint on D is a condition that D is expected to satisfy in order to capture the semantics of the application domain

A set IC of integrity constraints (ICs) helps maintain the correspondence between D and that reality

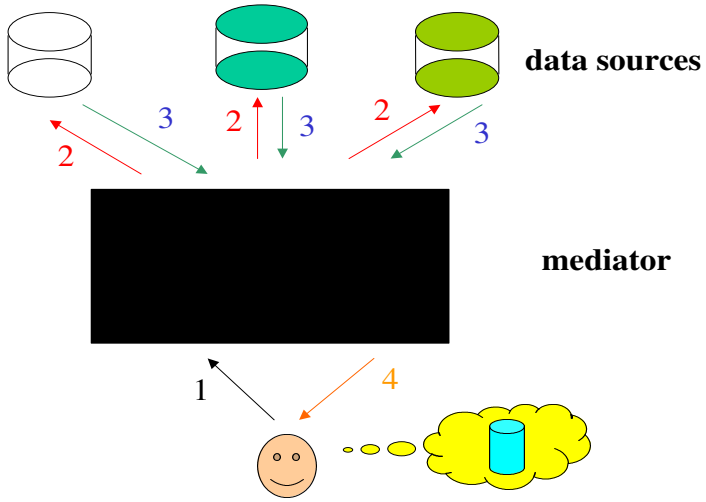
If D satisfies IC, we say that D is consistent

ICs can be expressed in symbolic languages of predicate logic

The database can be seen as a mathematical structure

For several reasons a database may become inconsistent with respect to a given set of desirable ICs

An important scenario: when data is virtually integrated from several autonomous and heterogeneous data sources



The user interacts with a mediator, having the feeling of being interacting with a single, real database, with a global schema

No way to maintain a set of global ICs satisfied!

What If the Database is Inconsistent?

Bringing it back to a consistent state may be difficult, impossible, nondeterministic, undesirable, unmaintainable, etc.

We may have to live with inconsistent data ...

The database (the model) is departing from the outside reality that is being modeled

However, the information is not all semantically incorrect

Most likely most of the data in the database is still "consistent"

Idea: (a) Keep the database as it is

(b) Obtain semantically meaningful information at query time; dealing with inconsistencies on-the-fly

Particularly appealing in virtual data integration ...

Characterizing Consistent Data

What is the consistent data in an inconsistent database?

What are the consistent answers to a query posed to an inconsistent database?

A mathematically precise definition is needed, that makes sense from the point of view of the intuitions behind the concept and of its applications

In (Arenas,Bertossi,Chomicki; PODS99) such a characterization was provided

Intuitively, the consistent data in an inconsistent database D is invariant under all minimal ways of restoring D's consistency

That is, consistent data persists across all the minimally repaired versions of the original instance: the repairs of D

Example: The instance D violates the functional dependency FD: $Name \longrightarrow Salary$

$Employee$	$Name$	$Salary$
	$Page$	5K
	$Page$	8K
	$Smith$	3K
	$Stowe$	7K

Two possible (minimal) repairs if only deletions/insertions of whole tuples are allowed: D_1, resp. D_2

$Employee$	$Name$	$Salary$
	$Page$	5K
	$Smith$	3K
	$Stowe$	7K

$Employee$	$Name$	$Salary$
	$Page$	8K
	$Smith$	3K
	$Stowe$	7K

$(Stowe, 7\text{K})$ persists in all repairs: it is consistent information

$(Page, 8\text{K})$ does not; actually it participates in the violation of FD

Consistent Query Answering

A consistent answer to a query $\mathcal{Q}$ from a database D that is inconsistent wrt IC is an answer that can be obtained as a usual answer to $\mathcal{Q}$ from every possible repair of D wrt IC

- $\mathcal{Q}_1 : Employee(x, y)?$

  Consistent answers: $(Smith, 3K), (Stowe, 7K)$

- $\mathcal{Q}_2 : \exists y Employee(x, y)?$

  Consistent answers: $(Page), (Smith), (Stowe)$

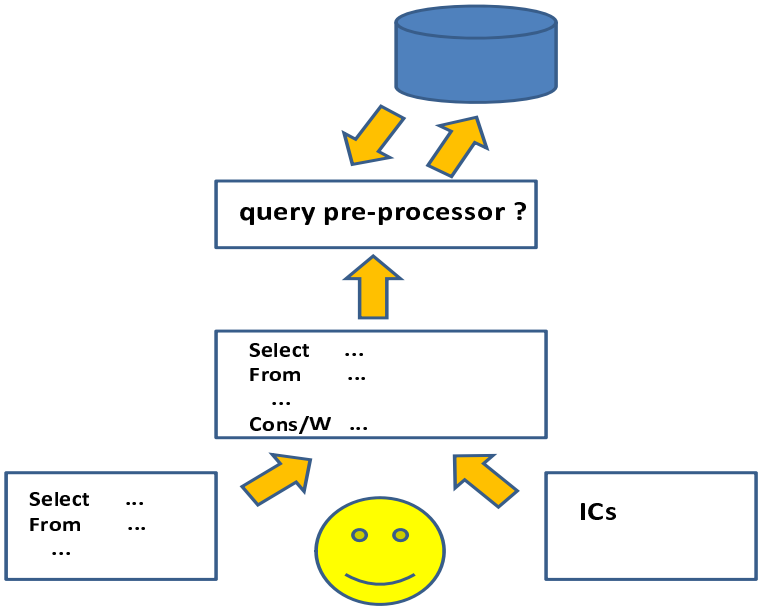
Consistent query answering (CQA) is different from classical data cleaning

However, CQA is relevant for data quality; an increasing need in business intelligence

It also provides concepts and techniques for data cleaning

Next DBMSs should provide more flexible, powerful, and user friendlier mechanisms for dealing with semantic constraints

In particular, they should allow to be posed queries requesting for consistent data; and answer them



Why not an enhanced SQL?

SELECT	Name, Salary
FROM	Employee
CONS/W	FD: Name -> Salary;

(FD not maintained by the DBMS)

All Kinds of Interesting Problems!

- Logical Problems

Query answering in databases follows classical predicate logic (relational calculus)

What is the logic followed by CQA?

What about compositionality? How to compute consistent answers combining consistent answers to subqueries?

- Algorithmic Problems

How to compute consistent answers?

We have to avoid as much as possible computing and materializing all possible repairs

Try to use the only instance D at hand, the inconsistent one ...

Is it possible to rewrite a query $\mathcal{Q}$ that expects consistent answers into a new query $\mathcal{Q}'$, whose usual answers from D are the consistent answers to $\mathcal{Q}$?

In the example, the query can be transformed into a standard SQL query to be posed to the original database

```
SELECT   Name, Salary
FROM     Employee
WHERE    NOT EXISTS (
           SELECT *
           FROM   Employee E
           WHERE  E.Name = Name AND
                  E.Salary <> Salary);
```

No repair generation is needed to answer this query!

It can be answered in polynomial time in data ...

Always possible?

- Mathematical Problems

What is the intrinsic complexity of CQA?

As a decision problem? As a data retrieval problem?

Expressive power of logical languages to rewrite queries for CQA?

Tractable vs. intractable cases

Approximation algorithms for intractable cases?

Use information theory to characterize degrees of consistency of databases as mathematical structures

- Computational Implementation

How to enhance DBMS for doing CQA?

How to couple a DBMS with a reasoning system for CQA?

• Applications

Many, where consistency of data is an issue ...

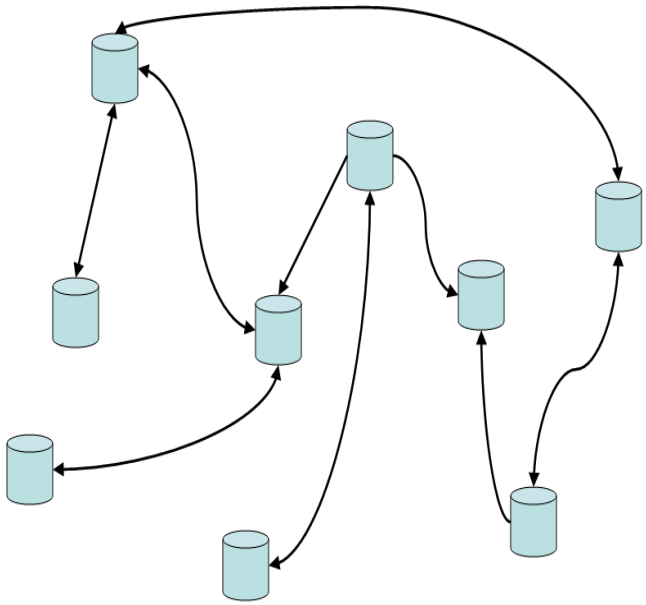
- Query answering in peer-to-peer data exchange systems

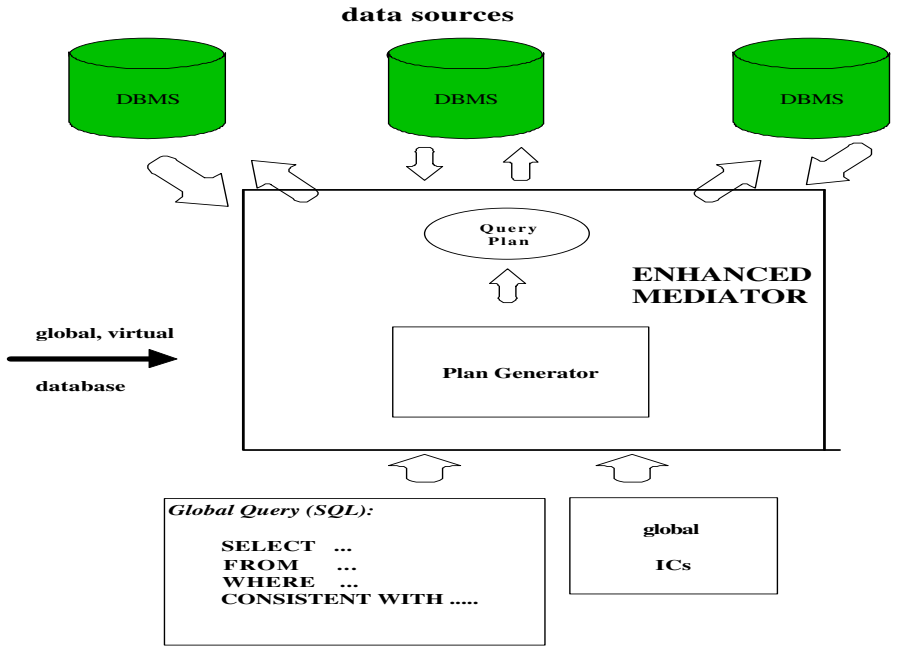
No centralized management

Peers exchange data at query answering time

Queries are posed to a peer who imports other peers' data or adjusts its own data

Trust relationships between peers may influence this process





- Virtual data integration

We can make the query plan generator to take into account the global ICs

The enhanced plan retrieves data from the sources, but returns only answers that are consistent with the global ICs

There are sufficiently expressive languages to express query plans

And can be evaluated using extended logic programming systems (answer set programming)

Query answers can be used to materialize an integrated and semantically correct global instance!

- Data quality and data cleaning

A vast territory still to be explored ...

Conclusions

Since the publication of the seminal PODS'99 paper much progress has been made around the problems mentioned before

A fast increasing number of papers have been and are being published on CQA

CQA has become quite relevant in databases and a growing community of researchers is working in it

There are still many open problems and many research avenues have not been explored yet

Given the diversity of challenges and potential applications, there is room for exciting interdisciplinary research

$\Longrightarrow$ mathematics, complexity theory, information theory, probability theory, information sharing agents, virtual data integration of biological databases, consistent integration of biological ontologies, CQA in the semantic web, ...