# MAGIK: Managing Completeness of Data

http://magik-demo.inf.unibz.it/public-version/

8th Reasoning Web Summer School (Reasoning Web 2012)

September 3 – 8, 2012 Vienna, Austria

**Ognjen Savkovic**
Free University of Bozen-Bolzano, Italy
savkovic@inf.unibz.it
joint work with **Sergey Paramonov, Mirza Paramita, and Werner Nutt**

Freie Universität Bozen
Libera Università di Bolzano
Free University of Bozen · Bolzano

## Data Quality: State of the Art

### What is Data Quality?

- **Data Quality (DQ) is a perception of data's fitness to serve its purpose in a given context.**
- DQ problems cost U.S. businesses around **$600 billion annually** [TWDI Journal]
- DQ problems are characterized by usually independent dimensions**,** such as **completeness, correctness, consistency, accuracy, timeliness…** (**no silver bullet for all DQ problems**). For example, data might be complete but inaccurate.

### What is Data Completeness?

- **Is all necessary data present?**
- **Completeness Measure:** The extent to which data are of sufficient breadth, depth, and scope for the task at hand (e.g., **query answering**)**.** A database might be incomplete in general but **still sufficiently complete** for the task at hand.
- **Application domains** includes areas like **Data Warehousing, data generated by business processes (workflow),** etc.

### Research Motivation

- **Managers** see the data throughout **Reports/Dashboards**
- **Data quality is a major concern** for decision support data (according to interviews with IT experts in companies and School Administration)
- **Diffuse market of enterprise solutions**: very expensive, not sufficiently comprehensive, non-standardized solutions


### Research Goals

- Take advantage of widely present **meta-information that accompanies data** (e.g., **business processes workflows, ETL processes, master data management systems, etc.) to assess data quality aspects**.
- Track back **the causes of bad data quality and propose fixes.**
- **Design and implement** algorithms that automate the proposed solutions.


## MAGIK at Work: School Information System

### Scenario

**The school administration (e.g., school personnel or school administrative process) provides Completeness Statements (meta-information) that holds over the existing data (that is in general incomplete).**

### School Schema

***pupil(name,level,code)*** … *pupils*
***class(level,code,dept)*** … *every class belongs to a department*
***langAtt(name,language)*** … *pupils attend language courses*

### Plain Reasoning

**Statement 1**: *We are complete for all pupils.*
`TABLE: pupil(Name,Level,Class) WHERE:`

**Statement 2**: We are complete for all pupils in the class **'1a'**.
`TABLE: pupil(Name,Level,Class) WHERE: Level=1 AND Class='a'`

**Query 1**: *Who are the pupils at the 1st level?*

```
SELECT p.name
FROM pupil AS p
WHERE p.level='1'
```

*Can we answer **Query 1** completely under the assumption of **Statement 1**?*
***Query 1 is complete, because Statement 1 guarantees that the data asked by the query is present in the database.***

*Can we answer **Query 1** completely under the assumption of **Statement 2**?*
***Query 1 is NOT complete, because Statement 2 guarantees completeness only for a specific part of the data asked by the query. Other parts might be incomplete.***

### Reasoning under Foreign Keys (FKs)

**Additional meta-information about the data we can be obtained from the schema foreign keys imposed over the data.**
**FK 1**: `pupil(level,code) REFERENCES class(level,code)`
**FK 2**: `langAtt(name) REFERENCES pupil(name)`

**Statement 3:** *We are complete for French learners.*
`TABLE: langAtt(Name,Lang) WHERE: Lang='french'`

**Query 2**: *Which science pupils learn French?*

```
SELECT p.name
FROM pupil AS p, class AS c, langAtt AS l
WHERE p.name=l.name AND l.lang='french'
AND p.level=c.level AND p.code=c.code
AND c.branch='science'
```

Can we answer **Query 2** completely under the assumption of **Statement 3** and foreign keys **FK1** and **FK2**?
***Query 1 is complete, because we are complete for all language attendances and in addition FK1 and FK2 guarantee that for every such language attendance record exists corresponding pupil and class record.***

### Reasoning under Finite Domains Constraints (FDs)

**Sometimes, the schema constrains a table column so that only predefined values can appear.**
**FD 1**: *In the table pupil the 3rd column (code) can be either **a** or **b**.* `pupil(code) IN {a,b}`

**Statement 2**: *We are complete for all pupils in the class **'1a'**.*
`TABLE: pupil(Name,Level,Class) WHERE: Level=1 AND Class='a'`

Can we answer **Query 1** completely under the assumption of **Statement 2** and **FD1**?
***Query 1 is NOT complete, because other classes (e.g., '1b') can exists.***

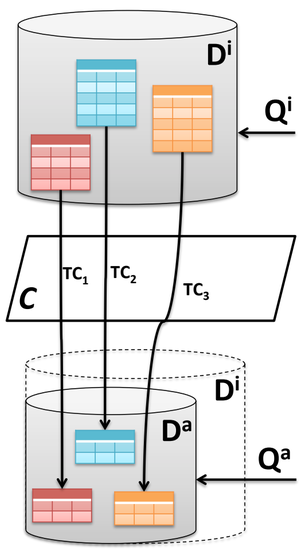
What is incomplete wrt **Query 1**? *Considering that we are complete for 1a and there can be only class 1b at the 1st level, to become complete for all 1st level classes we need to be complete for class 1b.*

**MAGIK** suggests to us ( ) to complete the data for the class '1b' and to confirm this with the following statement:
`TABLE: pupil(Name,Level,Class) WHERE: Level=1 AND Class='b'`
*Is **Query 1** complete if in addition to Statement 3 and FD1 we consider the* statement proposed by MAGIK?
***Query 1 is complete, because we are complete for all 1st level classes (namely, classes '1a' and '1b').***

## Formalization of the Problem

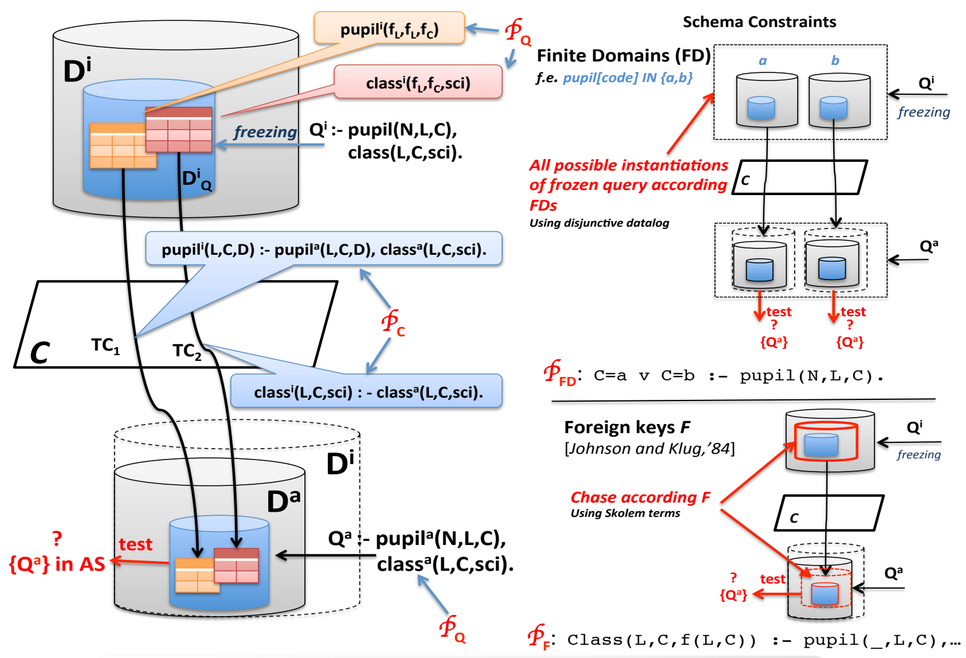
**Provided meta-information (called Table Completeness (TC) statements) that some parts of the existing database (called available database - $D^a$) is complete can we guarantee (deduce) that a query answer is the same (called Query Completeness (QC)) as if the query was evaluated over the complete database (called ideal database - $D^i$)?**

- To express partial completeness of database we use **table completeness (TC)** statements [A. Levy '96]
- For example, ($TC_1$) **we are complete for science pupils**, we express using the notation
  - `TABLE: pupil(Name,Level,Code)` (← complete table)
    `WHERE: class(Level,Code,science)` (← condition)
  - Alternatively, we can express this using datalog notation
    $\texttt{pupil}^a(N, L, C) \leftarrow \texttt{pupil}^i(N, L, C) \wedge \texttt{class}^i(L, C, \text{science})$.
- Let $D^a = \{\texttt{class(1,a,sci)}\}$, $D_1^i = \{\texttt{class(1,a,sci)}\}$ and $D_2^i = \{\texttt{class(1,a,sci)}, \texttt{pupil(john,1,a)}\}$
  - $(D^a, D_1^i)$ **satisfies** $TC_1$, but $(D^a, D_2^i)$ **doesn't satisfy** $TC_1$
  - Similarly for a query $Q(N) \leftarrow \texttt{pupil(N,L,C)}$:
    $Q$ **is complete** under $(D^a, D_1^i)$
    $Q$ **is NOT complete** under $(D^a, D_2^i)$



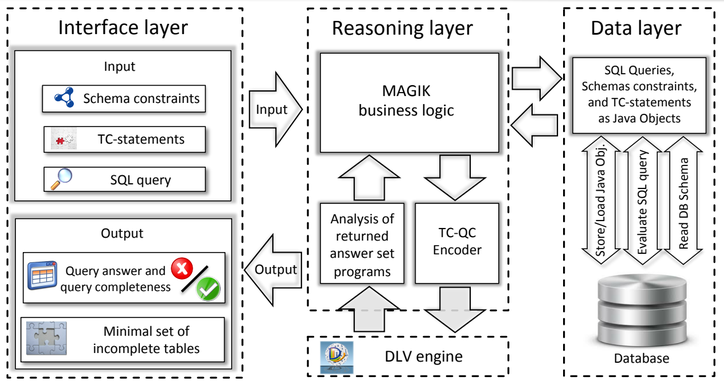

## Implementation

### Encoding of the Problem in Logic Programming (Answer Set Programming)




**Completeness** of **Q** follows from TCs in ***C***, FKs in ***F*** and FDs in ***FD*** ***iff*** the atom $\mathbf{Q}^a$ is in every **answer-set** of the program $\mathcal{P}_Q \cup \mathcal{P}_C \cup \mathcal{P}_F \cup \mathcal{P}_{FD}$

### System Architecture




**System features:**

- **Web based** application written in **JAVA** that posses powerful GUI that allows:
  - **Creation of a database schema** and **extraction of a schema** from the database catalog.
  - A user to **create/remove/edit TC-statements, FKs, FDs and Queries**.
  - **SQL `select-project-joins` queries** that in addition can contain the key word **`DISTINCT`** or grouping with the aggregate function **`count(*)`**.
- Runs using technologies/tools such as **Java Server Pages (JSP), Apache-Tomcat, Ubuntu-Linux, PostgreSQL, Hibernate and DLV.**

## Summary and Publications

- The **first realized** system that can reason about **query completeness** based on **partially complete database (reasoning about TC-QC entailment)**
- We gone beyond original **TC-QC problem**, and we investigate the impact of **Schema Constraints**, like **Foreign keys** and **Finite Domains**, on TC-QC entailment.
- We developed a component for **explanations and suggestions**, that in the case the query is not complete indicates which parts of a database are incomplete w.r.t. the query.

**DEMO-PAPER: MAGIK: Managing Completeness of Data**
*Ognjen Savkovic, Mirza Paramita, Sergey Paramonov, and Werner Nutt*
***Proc. of the 21st ACM Int. Conf. on Information and Knowledge Management (CIKM 2012)***