RQL: a Query Language for Implications

Jean-Marc Petit
(joint work with B. Chardin, E. Coquery and M. Pailloux)

INSA Lyon
CNRS and Université de Lyon

Dagstuhl Seminar – 12-16 May 2014
"Horn formulas, directed hypergraphs, lattices and closure systems: related formalisms and applications"

This work was partly funded by the ANR DEFIS 2009 program, DAG project (`http://liris.cnrs.fr/dag`).

Lattices, Closure Systems, Implications

Lattice $\longleftrightarrow$ **Closure System** $\longleftrightarrow$ **Closure operator** $\longleftrightarrow$ **Implication basis**

- Cf talks of Karell Bertet, Sergeï Kustnetzov, Marcel Wild and others ...

What about the data ?

- Binary relations (aka formal context in FCA)
- Relational databases (**functional dependencies**)

$\Rightarrow$ two (more or less hermetic) faces of the same coin

Lattices, Closure Systems, Implications

Lattice $\longleftrightarrow$ **Closure System** $\longleftrightarrow$ **Closure operator** $\longleftrightarrow$ **Implication basis**

- Cf talks of Karell Bertet, Sergeï Kustnetzov, Marcel Wild and others ...

What about the data ?

- Binary relations (aka formal context in FCA)
- Relational databases (**functional dependencies**)

$\Rightarrow$ two (more or less hermetic) faces of the same coin

Implications, Functional Dependencies and Beyond

A pattern mining perspective

Problem statement (sketch)

Given a dataset D and a semantics for implications, how to discover implications satisfied in dataset D?

- Semantics of implications
  - Implications in binary relation $\neq$ functional dependencies in DB
  - same syntax, same reasoning on rules (reflexivity, augmentation, transitivity) but different semantics wrt the data

Implications, Functional Dependencies and Beyond

A pattern mining perspective

Problem statement (sketch)

Given a dataset D and a semantics for implications, how to discover implications satisfied in dataset D?

- Semantics of implications
  - Implications in binary relation $\neq$ functional dependencies in DB
  - same syntax, same reasoning on rules (reflexivity, augmentation, transitivity) but different semantics wrt the data

Semantics of implications

Let b_0 be a binary relation (given by a $\{0,1\}$-relation)

$$b_0 \models X \to Y \Leftrightarrow \forall t \in b_0 \\ (\forall A \in X \ t.A = 1) \Rightarrow (\forall A \in Y \ t.A = 1)$$

Let $d = \{r_0, r_1, ..., r_n\}$ be a relational database

$$r_0 \models X \to Y \Leftrightarrow \forall t_1, t_2 \in r_0 \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_p})) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_n})), \\ \forall t_2 \in \pi_X(\sigma_{F'}(r_{j_0} \bowtie \ldots \bowtie r_{i_n})) \\ \text{such that } (t_1.rank = t_2.rank + 1) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

Semantics of implications

Let b_0 be a binary relation (given by a $\{0,1\}$-relation)

$$b_0 \models X \to Y \Leftrightarrow \forall t \in b_0 \\ (\forall A \in X \ t.A = 1) \Rightarrow (\forall A \in Y \ t.A = 1)$$

Let $d = \{r_0, r_1, ..., r_n\}$ be a relational database

$$r_0 \models X \to Y \Leftrightarrow \forall t_1, t_2 \in r_0 \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_p})) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_n})), \\ \forall t_2 \in \pi_X(\sigma_{F'}(r_{j_0} \bowtie \ldots \bowtie r_{i_n})) \\ \text{such that } (t_1.rank = t_2.rank + 1) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

Semantics of implications

Let b_0 be a binary relation (given by a $\{0,1\}$-relation)

$$b_0 \models X \to Y \Leftrightarrow \forall t \in b_0 \\ (\forall A \in X \ t.A = 1) \Rightarrow (\forall A \in Y \ t.A = 1)$$

Let $d = \{r_0, r_1, ..., r_n\}$ be a relational database

$$r_0 \models X \to Y \Leftrightarrow \forall t_1, t_2 \in r_0 \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_p})) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_n})), \\ \forall t_2 \in \pi_X(\sigma_{F'}(r_{j_0} \bowtie \ldots \bowtie r_{i_n})) \\ \text{such that } (t_1.rank = t_2.rank + 1) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

Semantics of implications

Let b_0 be a binary relation (given by a $\{0,1\}$-relation)

$$b_0 \models X \to Y \Leftrightarrow \forall t \in b_0 \\ (\forall A \in X \ t.A = 1) \Rightarrow (\forall A \in Y \ t.A = 1)$$

Let $d = \{r_0, r_1, ..., r_n\}$ be a relational database

$$r_0 \models X \to Y \Leftrightarrow \forall t_1, t_2 \in r_0 \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_p})) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

$$d \models X \to Y \Leftrightarrow \forall t_1 \in \pi_X(\sigma_F(r_{i_0} \bowtie \ldots \bowtie r_{i_n})), \\ \forall t_2 \in \pi_X(\sigma_{F'}(r_{j_0} \bowtie \ldots \bowtie r_{i_n})) \\ \text{such that } (t_1.rank = t_2.rank + 1) \\ (\forall A \in X \ t_1.A = t_2.A) \Rightarrow (\forall A \in Y \ t_1.A = t_2.A)$$

Semantics of implications (cont'ed)

$$d \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_0 \bowtie \ldots \bowtie r_n))$$
$$(\forall A \in X(2 * ABS(t_1.A - t_2.A)/(t_1.A + t_2.A) < 0.1))$$
$$\Rightarrow (\forall A \in Y(2 * ABS(t_1.A - t_2.A)/(t_1.A + t_2.A) < 0.1))$$

$$d \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_0 \bowtie \ldots \bowtie r_n))$$
$$(\forall A \in X\ t_1.A \leq t_2.A) \Rightarrow (\forall A \in Y\ t_1.A \leq t_2.A)$$

$$r_0 \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2, t_3 \in r_0$$
$$(\forall A \in X(t_1.A \leq t_2.A) \wedge (t_3.A \leq t_2.A))$$
$$\Rightarrow (\forall A \in Y(t_1.A \leq t_2.A) \wedge (t_3.A \leq t_2.A))$$

Semantics of implications (cont'ed)

$$d \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_0 \bowtie \ldots \bowtie r_n))$$
$$(\forall A \in X(2 * ABS(t_1.A - t_2.A)/(t_1.A + t_2.A) < 0.1))$$
$$\Rightarrow (\forall A \in Y(2 * ABS(t_1.A - t_2.A)/(t_1.A + t_2.A) < 0.1))$$

$$d \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_0 \bowtie \ldots \bowtie r_n))$$
$$(\forall A \in X \ t_1.A \leq t_2.A) \Rightarrow (\forall A \in Y \ t_1.A \leq t_2.A)$$

$$r_0 \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2, t_3 \in r_0$$
$$(\forall A \in X(t_1.A \leq t_2.A) \wedge (t_3.A \leq t_2.A))$$
$$\Rightarrow (\forall A \in Y(t_1.A \leq t_2.A) \wedge (t_3.A \leq t_2.A))$$

Semantics of implications (cont'ed)

$$d \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_0 \bowtie \ldots \bowtie r_n))$$
$$(\forall A \in X(2 * ABS(t_1.A - t_2.A)/(t_1.A + t_2.A) < 0.1))$$
$$\Rightarrow (\forall A \in Y(2 * ABS(t_1.A - t_2.A)/(t_1.A + t_2.A) < 0.1))$$

$$d \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2 \in \pi_X(\sigma_F(r_0 \bowtie \ldots \bowtie r_n))$$
$$(\forall A \in X \ t_1.A \leq t_2.A) \Rightarrow (\forall A \in Y \ t_1.A \leq t_2.A)$$

$$r_0 \models X \rightarrow Y \Leftrightarrow \forall t_1, t_2, t_3 \in r_0$$
$$(\forall A \in X(t_1.A \leq t_2.A) \wedge (t_3.A \leq t_2.A))$$
$$\Rightarrow (\forall A \in Y(t_1.A \leq t_2.A) \wedge (t_3.A \leq t_2.A))$$

Main question

Towards a query language to express semantics of implication

Does there exist syntactic restrictions to define "semantics of implications" such that those implications define a **closure system**?

Contribution.

From a theoretical point of view:

- Definition of *SafeRL*, a logical rule query language from the safe tuple relational calculus (TRC a.k.a SQL)
- Main property: Every *SafeRL* query defines a closure system.

From a practical point of view

- RQL: a SQL-like query language derived from *SafeRL*
- Web-based application opens to everyone
  `http://rql.insa-lyon.fr`

Ingredients for a logical rule query language.

Variables, syntax and semantics !

- Number of tuples ?
  1 for implication, 2 for FD and variants;
  In fact, could be 3, 4 ...
- Tuple defined over what ?
  the result of a SQL expression $\Rightarrow \psi$ TRC formula
- Condition to be verified?
  logical formulas on tuples and every attribute of LHS+RHS
  $\Rightarrow \delta$ $\mathcal{RL}$ formula

$SafeRQL$ queries: Syntax

Definition (Syntax)

$$Q = \{ X \to Y \mid \forall t_1 \ldots \forall t_n \left[\psi(t_1, \ldots, t_n) \wedge \left(\forall A \in X(\delta(A, t_1, \ldots, t_n)) \to \forall A \in Y(\delta(A, t_1, \ldots, t_n)) \right) \right] \}$$

- X, Y free schema variable
- ψ is a safe TRC formula (i.e. SQL), δ a $\mathcal{RL}$ formula
- $t_1, \ldots, t_n$ are free tuples variables in ψ and δ, A is a free attribute variable in δ
- $sch(Q) = \bigcap_{i=1}^{n} sch(t_i)$

TRC Formulas

Definition (Syntax)

$$\psi ::= \mathtt{R}(t) \mid t_1.\mathtt{A}\Box t_2.\mathtt{B} \mid t.\mathtt{A}\Box \mathtt{c} \mid \neg\psi \mid \psi_1 \wedge \psi_2 \mid \exists t : \mathtt{X}\ (\psi)$$

Definition (Semantic)

- $\langle d, \sigma\rangle \models \mathtt{R}(t)$ if $\sigma(t) \in d(\mathtt{R}), \mathtt{R} \in \mathbf{R}$
- $\langle d, \sigma\rangle \models t_1.\mathtt{A}\Box t_2.\mathtt{B}$ if $\sigma(t_1)(\mathtt{A})\Box\sigma(t_2)(\mathtt{B})$
- $\langle d, \sigma\rangle \models t.\mathtt{A}\Box \mathtt{c}$ if $\sigma(t)(\mathtt{A})\Box\mathtt{c}$
- $\langle d, \sigma\rangle \models \neg\psi$ if $\langle d, \sigma\rangle \not\models \psi$
- $\langle d, \sigma\rangle \models \psi_1 \wedge \psi_2$ if $\langle d, \sigma\rangle \models \psi_1$ and $\langle d, \sigma\rangle \models \psi_2$
- $\langle d, \sigma\rangle \models \exists t : \mathtt{X}\ (\psi)$ if there exists a tuple $\mathtt{t}$ over $\mathtt{X}$ such that $\langle d, \sigma_{t\mapsto \mathtt{t}}\rangle \models \psi$

$\mathcal{RL}$ Formulas

Definition (Syntax)

$$\delta ::= t_1.A \sqsubset t_2.B \mid t.A \sqsubset \mathsf{c} \mid \neg\delta \mid \delta_1 \wedge \delta_2$$

Definition (Semantic)

- $\langle \sigma, \rho \rangle \models t_1.A \sqsubset t_2.B$ **iff** $\sigma(t_1)(\rho(A)) \sqsubset \sigma(t_2)(\rho(B))$
- $\langle \sigma, \rho \rangle \models t.A \sqsubset \mathsf{c}$ **iff** $\sigma(t)(\rho(A)) \sqsubset \mathsf{c}$
- $\langle \sigma, \rho \rangle \models \neg\delta$ **iff** $\langle \sigma, \rho \rangle \not\models \delta$
- $\langle \sigma, \rho \rangle \models \delta_1 \wedge \delta_2$ **iff** $\langle \sigma, \rho \rangle \models \delta_1$ and $\langle \sigma, \rho \rangle \models \delta_2$

Safe$\mathcal{RL}$ queries: Semantics

Definition (Semantic)

$\langle d, \Sigma \rangle \models \langle \psi, \delta \rangle$ if for all σ s.t.. $\langle d, \sigma \rangle \models \psi$:

$$\text{if } \forall \mathtt{A} \in \Sigma(X)\ \langle \sigma, \rho_{A \mapsto \mathtt{A}} \rangle \models \delta$$
$$\text{then } \forall \mathtt{A} \in \Sigma(Y) \langle \sigma, \rho_{A \mapsto \mathtt{A}} \rangle \models \delta$$

$$ans(Q, d) = \{\Sigma(X) \to \Sigma(Y) \mid \langle d, \Sigma \rangle \models \langle \psi, \delta \rangle\}$$

Two main results.

THM

Let Q be a $SafeRL$ query over a database d.

1. $ans(Q, d)$ defines a closure system $C(Q)$ over $sch(Q)$
2. There exists a SQL query Q' over d such that Q' computes a base $B(Q)$ of C(Q), i.e. $IRR(Q) \subseteq B(Q) \subseteq C(Q)$

- $B(Q)$: *agree sets* for FD and *binary relation* for implications
- Proof of 1. similar to the proof given for Functional Dependencies by Mannila and Raiha 1994, Demetrovics and Thi 1995.
- Proof of 2. a bit tricky

Computing the Direct Canonical Basis.

Let Q be a $Safe\mathcal{RL}$ query against d and $B(Q)$ such that $IRR(Q) \subseteq B(Q) \subseteq C(Q)$

- For each attribute A in $sch(Q)$

  $Max(A, Q) = max_{\subseteq}\{X \in B(Q) | A \notin X\}$

  $\rightarrow$ equivalent to a CNF

  $CMax(A, Q) = \{sch(Q) \setminus X | X \in Max(A, Q)\}$

  $\rightarrow$ seen as an hypergraph

  $Lhs(A, Q) =$ minimal transversal of $CMax(A, Q)$

  $\rightarrow$ equivalent to a DNF

- $\bigcup_{A \in sch(Q)}\{X \rightarrow A \mid X \in Lhs(A, Q)\}$: direct canonical basis

RQL: a Practical Language fro $SafeRL$

Recall

$$Q = \{X \rightarrow Y \mid \langle \psi(t_1, \ldots, t_n), \delta(A, t_1, \ldots, t_n) \rangle\}$$

RQL has 5 clauses (look and feel of SQL):

FINDRULES
OVER $A_1, ..., A_n$
SCOPE $t_1(SQL_1), ..., t_n(SQL_n)$
WHERE $condition(t_1, ..., t_n)$
CONDITION ON A IS $\delta(A, t_1, ..., t_n)$

Examples

```
FINDRULES
OVER Empno,Lastname,Workdept,Job,Sex,Bonus
SCOPE t1,t2 Emp
CONDITION ON A IS t1.A = t2.A;
```

```
FINDRULES
OVER Empno, Lastname, Workdept, Job, Sex, Bonus, Mgrno
SCOPE t1 Emp
CONDITION ON A IS t1.A IS NULL
```

Examples

```
FINDRULES
OVER ...
SCOPE t1,t2,t3 sensors
WHERE t2.time = t1.time+interval 1 minute AND
t3.time = t2.time+interval 1 minute
CONDITION ON A IS t1.A < t2.A AND t2.A > t3.A;
```

RQL query processing.

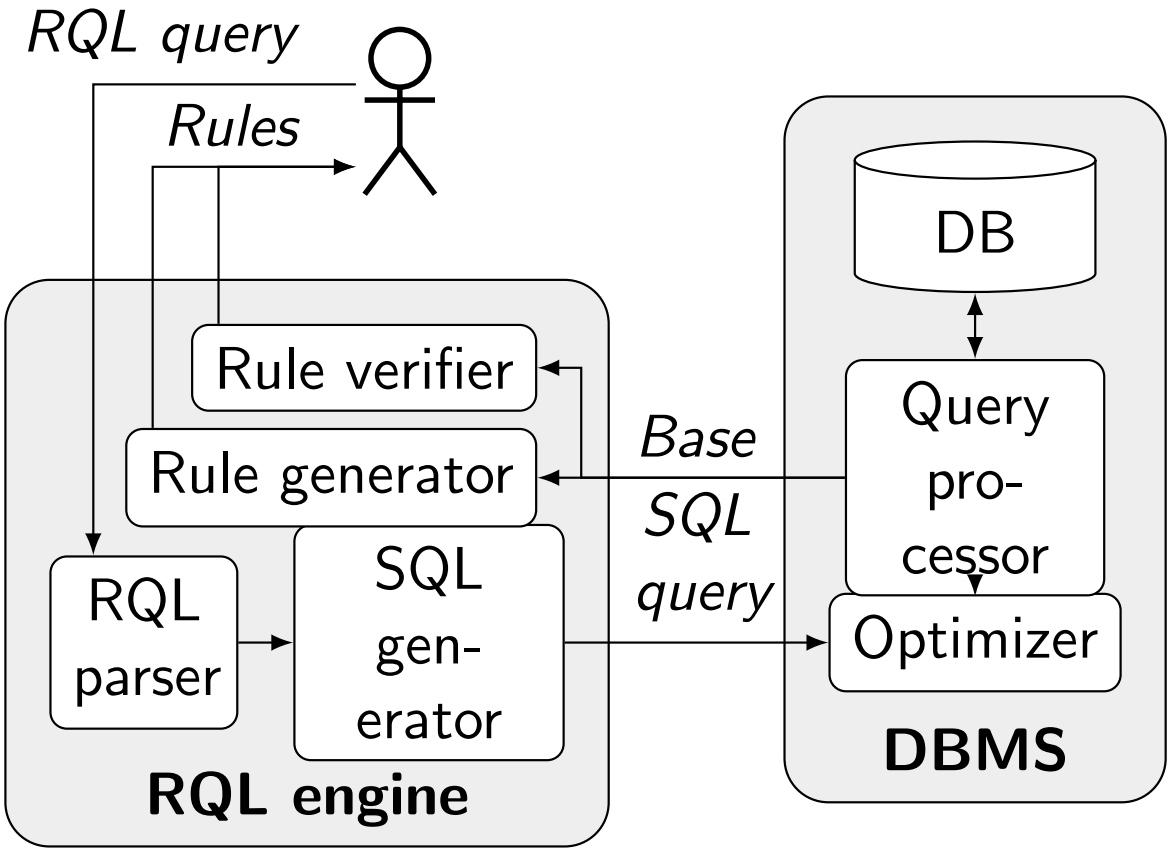

Figure: RQL queries processing overview

RQL Web Interface

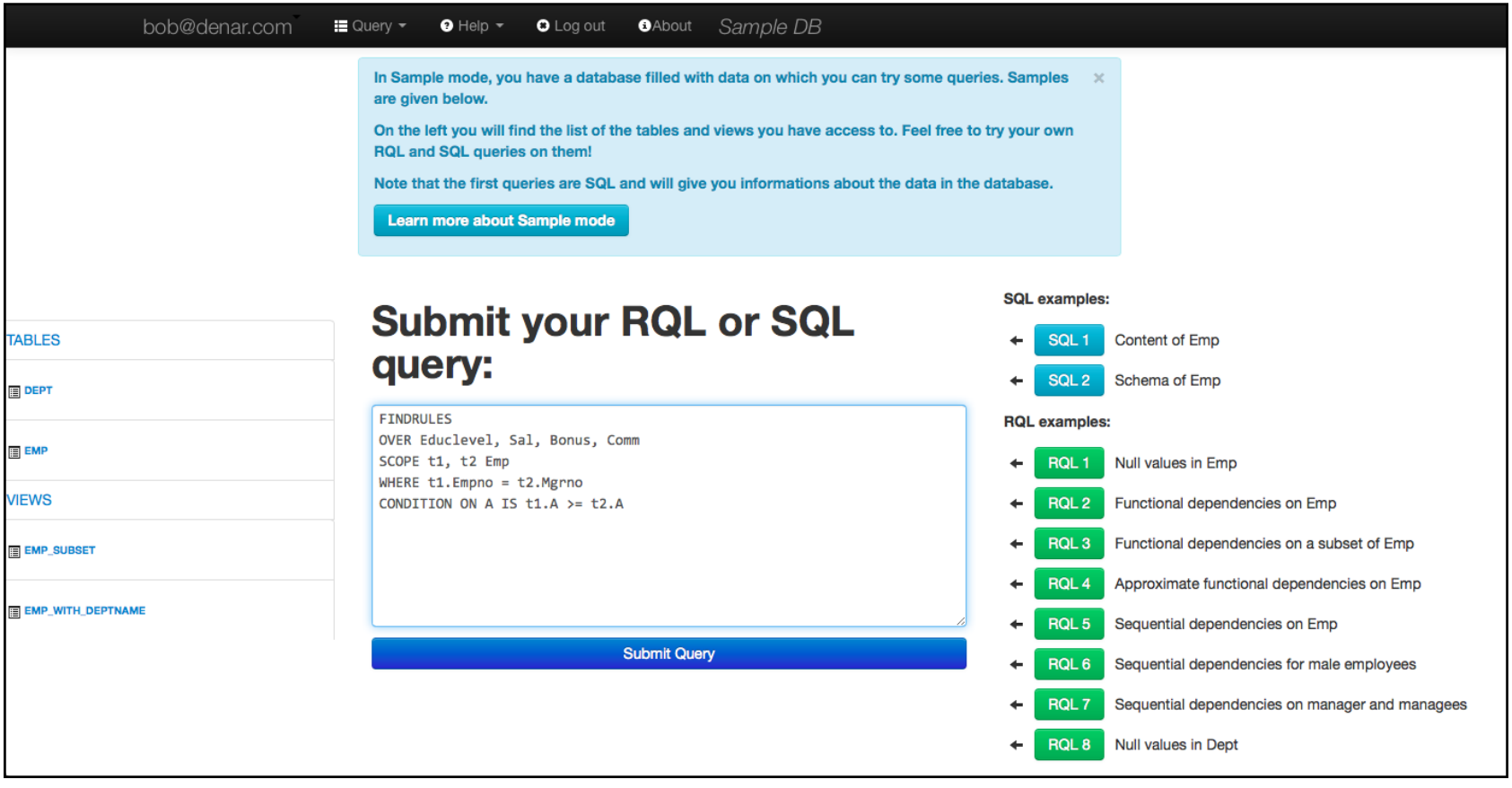

Figure: RQL Interface

RQL Web Interface

Rule verification:

The rule **Sal Educlevel ➜ Bonus** is **false**

Counter-example:

EMPNO	LASTNAME	WORKDEPT	JOB	EDUCLEVEL	SEX	SAL	BONUS	COMM	MGRNO
10	SPEN	C01	FINANCE	18	F	52750	500	4220	20
20	THOMP	null	MANAGER	18	M	41250	800	3300	null

Generated query:

```
1. SELECT t1.*, t2.*
2. FROM Emp t1, Emp t2
3. WHERE (t1.Sal >= t2.Sal AND t1.Educlevel >= t2.Educlevel)
4. AND CASE WHEN (t1.Bonus >= t2.Bonus) THEN 1 ELSE 0 END = 0
5. AND rownum <= 1
```

Figure: Counter-example with RQL

Summary

- RQL: a practical language to express different semantics for implication
- Discovery of implications seen as a Query Processing problem
- Powerful techniques for data analysts to interact with her data through counter examples (similar to "DB design by example" with Armstrong databases)
- Advantages
  - Query the data where they are : in DBMS !
  - Use/extend DB languages for pattern mining problems
  - RQL: easy to learn language for SQL-aware data analysts
  - Try it out ! http://rql.insa-lyon.fr

For Further Reading I


S. Abiteboul, R. Hull, and V. Vianu, *Foundations of Databases*. Addison-Wesley, 1995.


B. Ganter and R. Wille, *Formal Concept Analysis*. Springer, 1999.


N. Caspard and B. Monjardet, "The lattices of closure systems, closure operators, and implicational systems on a finite set: A survey," *Discrete Applied Mathematics*, vol. 127, no. 2, pp. 241–269, 2003.


H. Mannila and K.-J. Räihä, "Algorithms for inferring functional dependencies from relations," *Data and Knowldege Engineering*, vol. 12, no. 1, pp. 83–99, 1994.


J. Demetrovics and V. D. Thi, "Some remarks on generating Armstrong and inferring functional dependencies relation," *Acta Cybernetica*, vol. 12, no. 2, pp. 167–180, 1995.


M. Agier, J.-M. Petit, and E. Suzuki, "Unifying framework for rule semantics: Application to gene expression data," *Fundam. Inform.*, vol. 78, no. 4, pp. 543–559, 2007.

For Further Reading II


S. Lopes, J.-M. Petit, and L. Lakhal, "Functional and approximate dependency mining: database and FCA points of view," *J. Exp. Theor. Artif. Intell.*, vol. 14, no. 2-3, pp. 93–114, 2002.


M. Agier, C. Froidevaux, J.-M. Petit, Y. Renaud, and J. Wijsen, "On Armstrong-compliant Logical Query Languages," in *4th International Workshop on Logic in Databases (LID 2011) in conjunction with EDBT/ICDT conference*, G. H. L. Fletcher and S. Staworko, Eds. ACM, Mar. 2011, pp. 33–40.


B. Chardin, E. Coquery, B. Gouriou, M. Pailloux, and J.-M. Petit, "Query Rewriting for Rule Mining in Databases," in *Languages for Data Mining and Machine Learning (LML) Workshop@ECML/PKDD 2013*, B. Crémilleux, L. D. Raedt, P. Frasconi, and T. Guns, Eds., Sep. 2013, pp. 1–16.