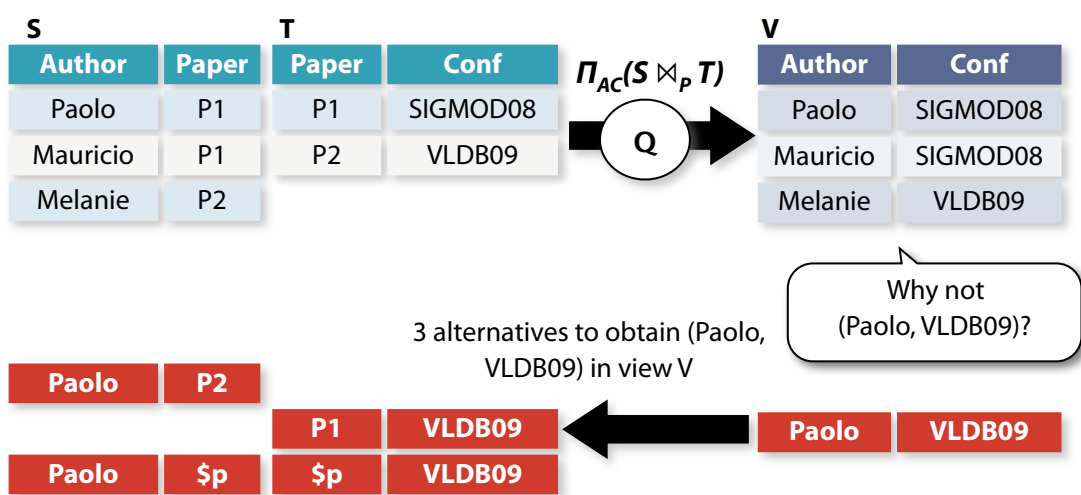


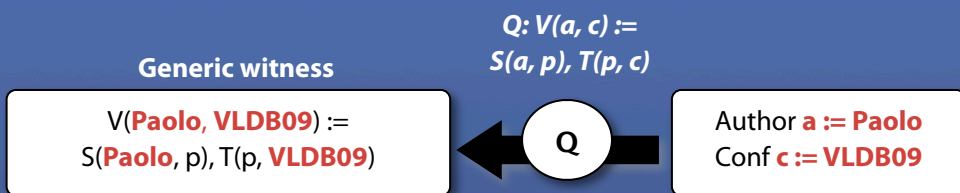
# Artemis: A System for Analyzing Missing Answers



## Introduction & Motivation

- Understanding and debugging SQL queries is generally a tedious process.
- Artemis is a step towards debugging and understanding SQL queries by enabling analysis of missing answers.
- Same problem and motivation as in [2].
  - Artemis allows for simultaneous analysis of multiple (correlated) missing answers.
  - Analysis supported by generated explanations, based on source-data (as opposed to query-based explanations [3]).
- Flexible trust mechanism, not limited to tables and subsets of attributes.
- Constraint solving over conditional tables.

## Generic witness



- A generic witness is a concise pattern that every generated explanation will match.
- It describes the combination of tuples required in the sources in order to produce the missing tuples.
- Takes as input a set of queries  $Q$  and a set of missing tuples  $E$ .

## Artemis Features

- Considers key, unique, and foreign key constraints.
- Determines optimal explanations for the insertion of multiple tuples.
- Allows to leave attribute values blank.
- Allows to correlate attribute values across tuples using named variables.
- Permits to mark views and tables as immutable or with the constraint of having minimal number of side-effects.

## Source c-views

**c-S**

Author	Paper	cond
Paolo	P1	TRUE
Mauricio	P1	TRUE
Melanie	P2	TRUE
Paolo	v1	$v1 \neq P1$

**c-T**

Paper	Conf	cond
P1	SIGMOD08	TRUE
P2	Vldb09	TRUE
v2	Vldb09	$v2 \neq P2$

- A c-table includes a condition for each tuple in the table.
- Semantics: A tuple in the table exists if its condition evaluates to true.
- Existing tuples in the source database  $D$  get a true condition.
- Each pattern in the generic witness is added as c-tuples in  $D$ .
- The condition depends on constraints implied by the generic witness, key constraints, and foreign key constraints.

## Target c-views

**c-V**

Author	Conf	cond
Paolo	SIGMOD08	TRUE
Mauricio	SIGMOD08	TRUE
Melanie	Vldb09	TRUE
Paolo	SIGMOD08	$v1 \neq P1 \ \& \ v1 = P1$
Paolo	Vldb09	$v1 \neq P1 \ \& \ v1 = P2$
Paolo	Vldb09	$v2 \neq P2 \ \& \ v2 = P1$
Mauricio	Vldb09	$v2 \neq P2 \ \& \ v2 = P1$
Melanie	Vldb09	$v2 \neq P2 \ \& \ v2 = P2$
Paolo	Vldb09	$v1 \neq P1 \ \& \ v2 \neq P2 \ \& \ v1 = v2$

- We produce a set of target-side c-tables by executing all queries in  $Q$  over the source c-views.
- The condition of a tuple  $t$  in these c-tables is the conjunction of the conditions of tuples in the provenance of  $t$  and the join conditions.

## Candidate Explanations

VARIABLES v2:PAPER, i1:BOOL, i2:BOOL, i3:BOOL, s:INT

SOLVE

$(v2 \neq P2 \ \& \ v2 = P1) \ \& \ ((v2 \neq P2 \ \& \ v2 = P1) \ \<=> \ i1) \ \& \ ((v2 \neq P2 \ \& \ v2 = P2) \ \<=> \ i2) \ \& \ ((v2 \neq P2 \ \& \ v2 = P2) \ \<=> \ i3) \ \& \ (s = i1 + i2 + i3)$

MINIMIZING s

Paolo	Vldb09	$v1 \neq P1 \ \& \ v1 = P2$
Paolo	Vldb09	$v2 \neq P2 \ \& \ v2 = P1$
Mauricio	Vldb09	$v2 \neq P2 \ \& \ v2 = P1$
Melanie	Vldb09	$v2 \neq P2 \ \& \ v2 = P2$

- We determine which c-tuples in the c-tables for  $Q$  represent missing tuples from  $E$ .
- We combine constraints of tuples matching with missing tuples from  $E$  and constraints for non-matching tuples not originally in view.
- We solve the combined constraint using a constraint solver [1].

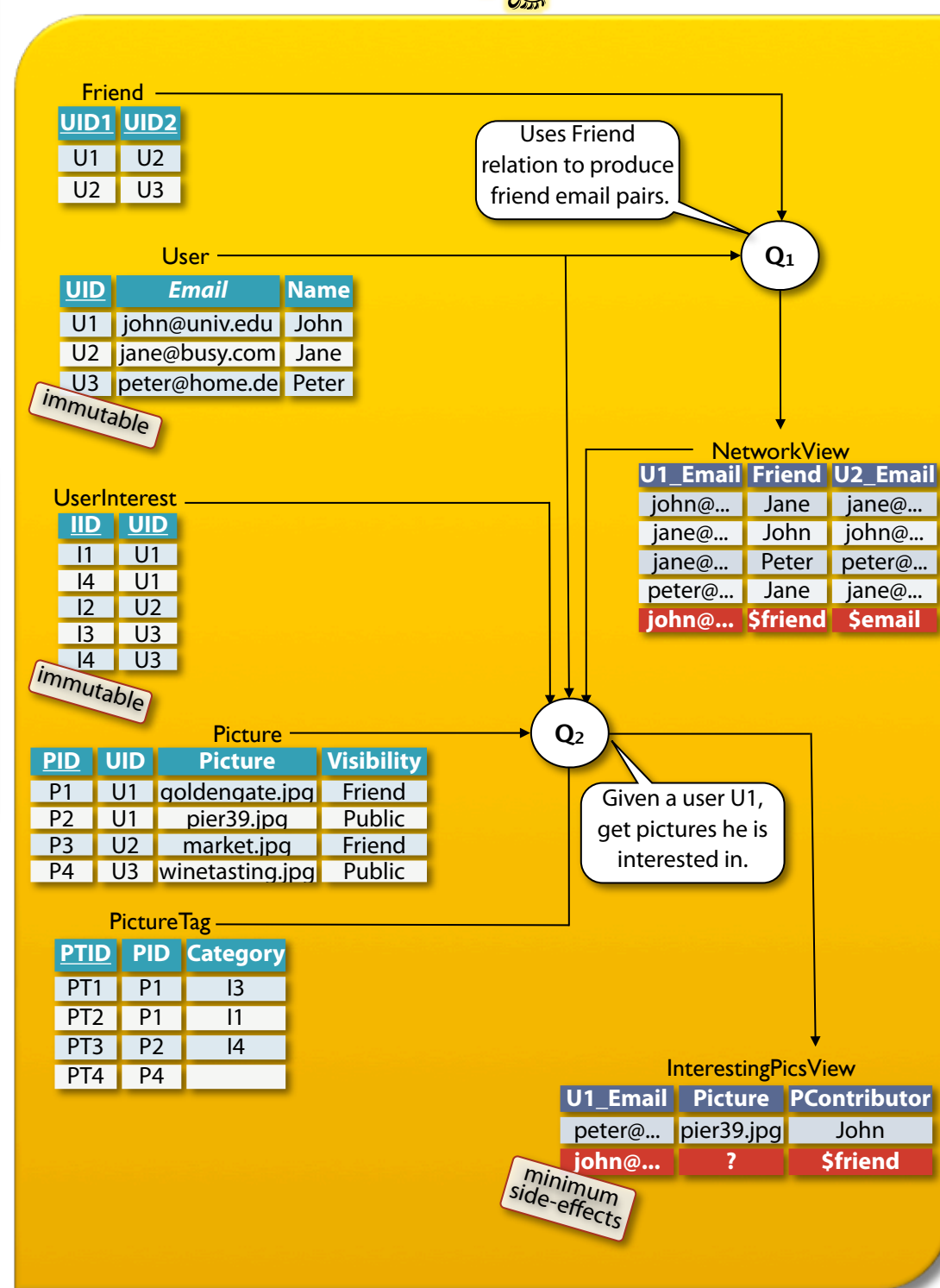
## Filter, Rank & Output

Author	Conf	Candidate explanations (CE)
Paolo	SIGMOD08	Never a side-effect
Paolo	Vldb09	CE1: Insert (Paolo, P2) into S
Paolo	Vldb09	CE2: Insert (P2, Vldb09) into T
Mauricio	Vldb09	Side-effect of CE2
Melanie	Vldb09	Never a side-effect
Paolo	Vldb09	CE3: Insert (Paolo, \$p) to S, (\$p, Vldb09) to T

- Apply user specified filters to candidate explanations output by previous step.
- For instance number of insertions
- Return the explanations, potentially ranked by user-specified functions.
- For instance minimum number of side-effects.

Filter: #insertions < 2  
Ranking: #side-effects, ASC

Solutions
Sol.1: Insert (Paolo, P2) into S
Sol.2: Insert (P2, Vldb09) into T



## The Demonstration

- Artemis is implemented as an Eclipse Plugin.
- Two scenarios:
  - Photoshare (example above)
  - TPC-H inspired scenario
- We show several debugging scenarios in these scenarios to illustrate:
  - How to use the Artemis Plugin.
  - How Artemis helps in understanding and debugging SQL queries.
  - How the Artemis algorithm works internally.
- Artemis is far from being complete yet: future work includes improving efficiency, extending the debugging capabilities, and improving visualization..

## References

## Contact Information



**Melanie Herschel**  
Universität Tübingen  
Tübingen, Germany  
melanie.herschel@uni-tuebingen.de

**Mauricio A. Hernández**  
IBM Almaden Research Center  
San Jose, CA, USA  
mauricio@almaden.ibm.com



**Wang Chiew Tan**  
UC Santa Cruz  
Santa Cruz, CA, USA  
wctan@cs.ucsc.edu

- Melanie Herschel, Mauricio A. Hernández, Wang Chiew Tan. Artemis: A System for Analyzing Missing Answers. In PVLDB 2009.
- I. P. Gent, C. Jefferson, I. Miguel. Minion: A Fast Scalable Constraint Solver. In ECAI 2006.
- A. Chapman, H. V. Jagadish. Why not? In SIGMOD 2009.
- J. Huang, T. Chen, A. Doan, J. F. Naughton. On the Provenance of Non-answers to Queries over Extracted Data. In PVLDB 2008.

Stay up-to-date on SQL debugging: <http://www-db.informatik.uni-tuebingen.de/research/nautilus>

