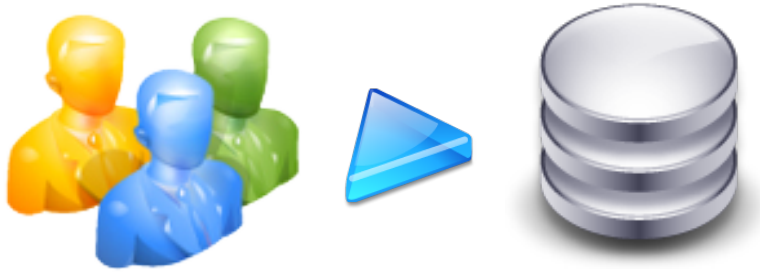
ROMA TRE
UNIVERSITÀ DEGLI STUDI



Using Keywords to Find the Right Path through Relational Data

*Roberto De Virgilio, **Antonio Maccioni** and Riccardo Torlone*

SEBD 2012 ***Venice, 26-06-2012***

Motivation Scenario

Number	LastName	FirstName	Position	Goals

Tabella di database

LastName	FirstName	Position

Tabella di recordset

The Problem

- **Keyword Search on Relational Databases**
  - **INPUT**: keyword query. Ex. **Q = {Database, Matters}**
  - **OUTPUT**: tuples matching keywords connected by foreign keys.

T_4: **Author**

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_5: **Publication**

	title	year
t_{15}	Data Integration	2002
t_{16}	Foundation Matters	2002

T_1: **Person**

	name	area
t_1	Watson	Database
t_2	Lenzerini	Database
t_3	Date	Database
t_4	Hunt	Inf. Systems

T_2: **Affiliated**

	professor	department
t_5	Watson	x123
t_6	Lenzerini	cs34
t_7	Date	cs34
t_8	Hunt	m111

T_3: **Department**

	id	dname	director
t_9	x123	CS	Watson
t_{10}	cs34	IE	Hunt
t_{11}	ee67	EE	Date
t_{12}	m111	ME	Hunt

The Problem

- **Keyword Search on Relational Databases**
  - **INPUT**: keyword query. Ex. **Q = {Database, Matters}**
  - **OUTPUT**: tuples matching keywords connected by foreign keys.

T_4: **Author**

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_5: **Publication**

	title	year
t_{15}	Data Integration	2002
t_{16}	Foundation Matters	2002

T_1: **Person**

	name	area
t_1	Watson	Database
t_2	Lenzerini	Database
t_3	Date	Database
t_4	Hunt	Inf. Systems

T_2: **Affiliated**

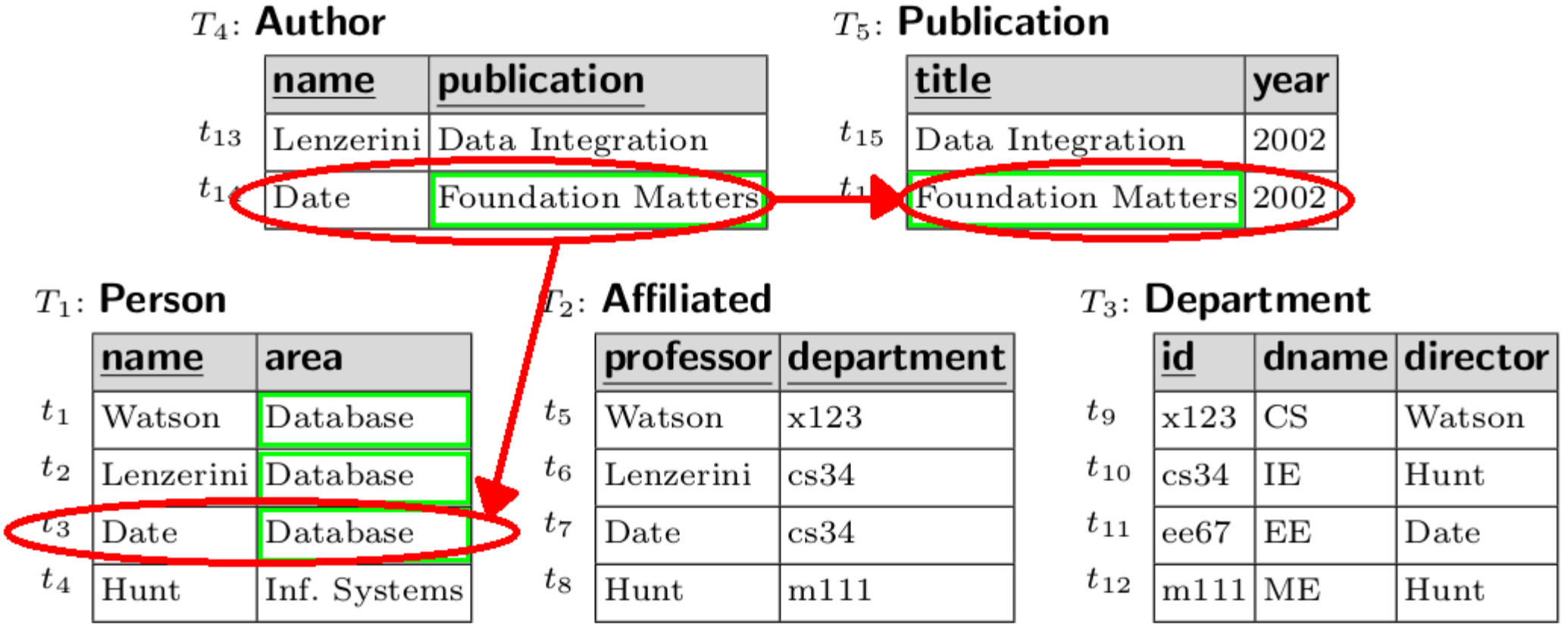
	professor	department
t_5	Watson	x123
t_6	Lenzerini	cs34
t_7	Date	cs34
t_8	Hunt	m111

T_3: **Department**

	id	dname	director
t_9	x123	CS	Watson
t_{10}	cs34	IE	Hunt
t_{11}	ee67	EE	Date
t_{12}	m111	ME	Hunt

The Problem

- **Keyword Search on Relational Databases**
  - **INPUT**: keyword query. Ex. **Q = {Database, Matters}**
  - **OUTPUT**: tuples matching keywords connected by foreign keys.



State Of The Art

- **Schema-free**: graph based algorithms
  [Kacholia et al., VLDB 2005] [He et al., SIGMOD 2007]
  - In-memory elaboration
  - Lot of I/O processing

  RDBMS features are not exploited [Qin et al., SIGMOD 2009]

- **Schema-based**: query rewriting
  [Luo et al., SIGMOD 2007] [Bergamaschi et al., SIGMOD 2011]
  - Produce and Execute lot of SQL queries
  - Empty results

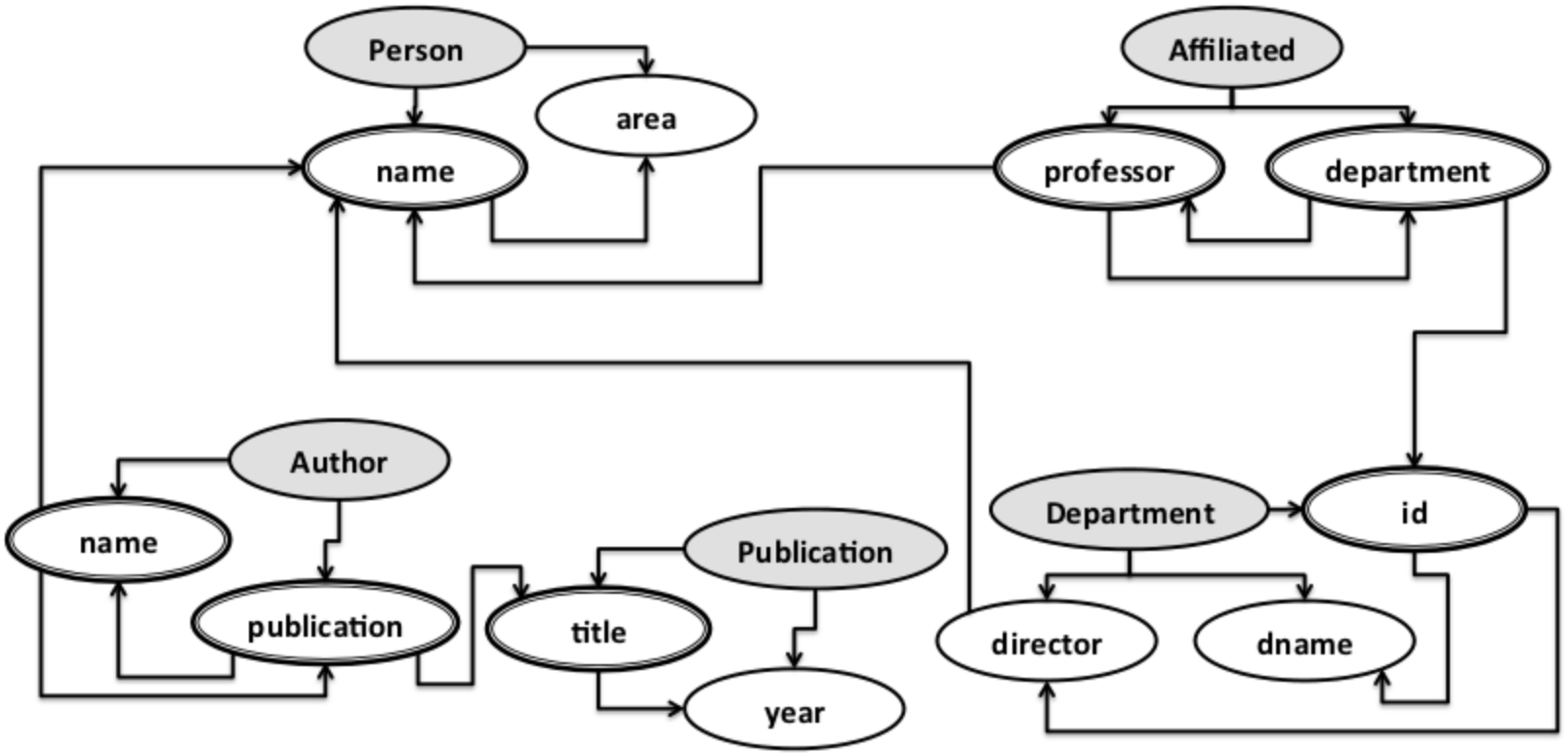
  RDBMS features are not WELL exploited [Qin et al., SIGMOD 2009]

The Idea

- Avoid ***middleware approaches*** and ***graph navigations***.
- Build the solutions just making use of the RDBMS.
- Lean *exploration* guided by the use of **paths** resulting from the logical schema.

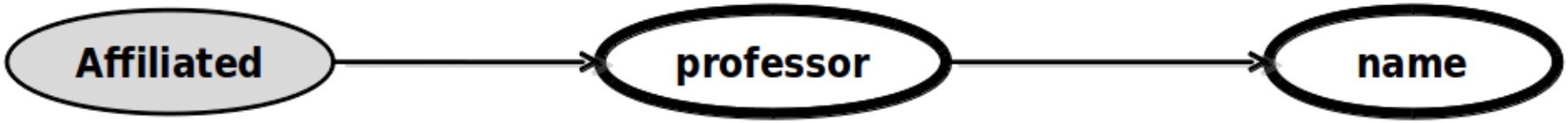
The Schema Graph

- Logical schema of the DB as a graph $SG = (V_s, E_s)$, where $V_s = (T, A)$ and $E_s \subseteq (T \times A) \cup (A \times A)$

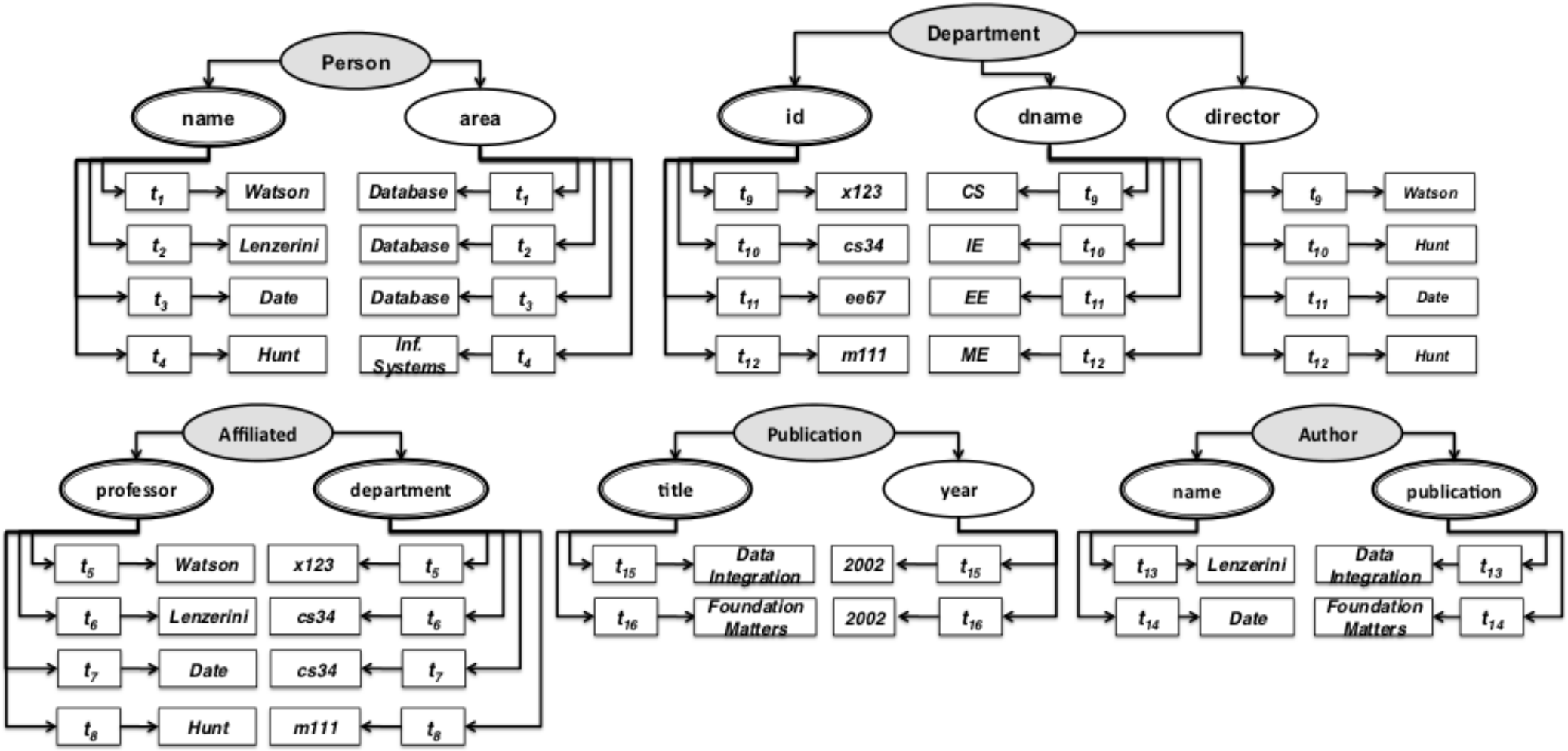


The Schema Paths

- A sequence $v_1 \rightarrow v_2 \rightarrow \ldots \rightarrow v_f$ where
  - $v_1 \in T$
  - $(v_i, v_{i+1}) \in E_s$

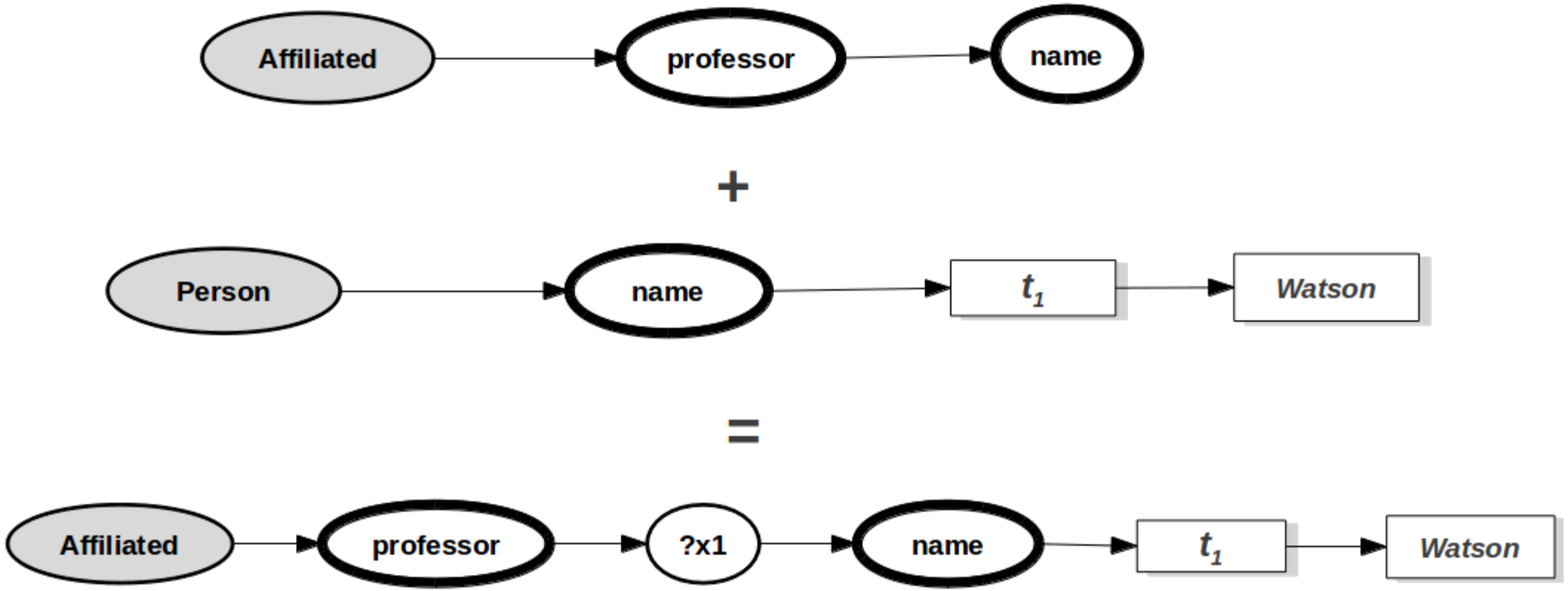


Data Graph And Data Paths



Expanded Data Paths

- Schema Path + Data Path = Expanded Data Path



The Approach

QUERYING

+

CLUSTERING

+

BUILDING

1. Retrieve the data paths
2. Expand the data paths

The Approach

QUERYING

+

CLUSTERING

+

BUILDING

1. Retrieve the data paths
2. Expand the data paths
3. Cluster expanded data paths
4. Sort the paths within a cluster

The Approach

QUERYING

+

CLUSTERING

+

BUILDING

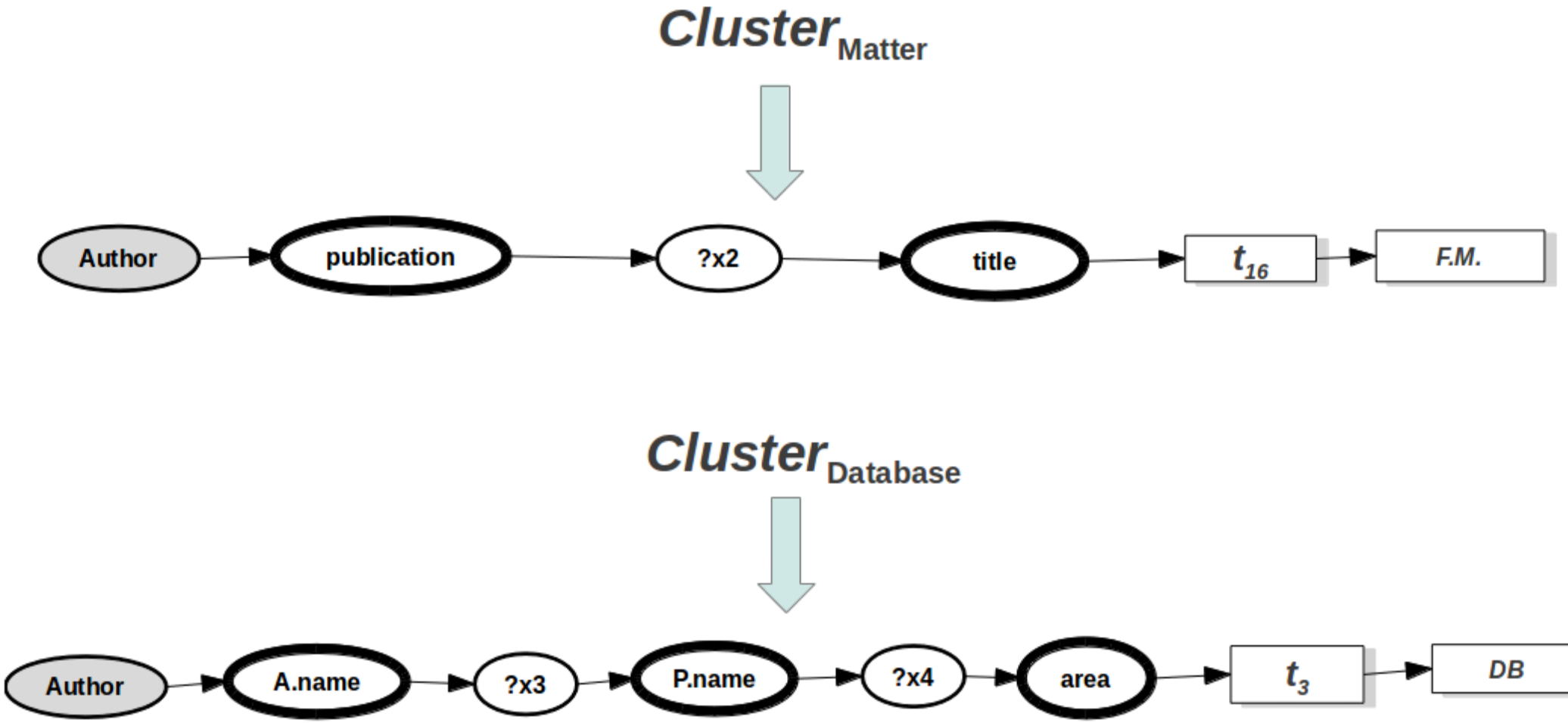
1. Retrieve the data paths
2. Expand the data paths
3. Cluster expanded data paths
4. Sort the paths within a cluster
5. Combine the paths
6. We explore the paths singularly

Clustering

$$cl_{Matters}: \left(\begin{array}{l} dp'_1 : \texttt{Author} \rightarrow \texttt{Author.publication} \rightarrow \texttt{t}_{14} \rightarrow F.M. \\ dp'_2 : \texttt{Publication} \rightarrow \texttt{Publication.title.} \rightarrow \texttt{t}_{16} \rightarrow F.M. \\ dp'_3 : \texttt{Author} \rightarrow \texttt{Author.name} \rightarrow ?x_1 \rightarrow \texttt{Author.publication} \rightarrow \texttt{t}_{14} \rightarrow F.M. \\ dp'_4 : \texttt{Author} \rightarrow \texttt{Author.publication} \rightarrow ?x_2 \rightarrow \texttt{Publication.title} \rightarrow \texttt{t}_{16} \rightarrow F.M. \\ \ldots \end{array} \right)$$

$$cl_{Database}: \left(\begin{array}{l} dp'_5 : \texttt{Person} \rightarrow \texttt{Person.area} \rightarrow \texttt{t}_1 \rightarrow Db \\ dp'_6 : \texttt{Person} \rightarrow \texttt{Person.area} \rightarrow \texttt{t}_2 \rightarrow Db \\ dp'_7 : \texttt{Person} \rightarrow \texttt{Person.area} \rightarrow \texttt{t}_3 \rightarrow Db \\ \ldots \\ dp'_8 : \texttt{Author} \rightarrow \texttt{Author.name} \rightarrow ?x_3 \rightarrow \texttt{Person.name} \rightarrow ?x_4 \rightarrow \texttt{Person.area} \rightarrow \texttt{t}_3 \rightarrow Db \\ \ldots \end{array} \right)$$

Building


Cluster Matter
Author
publication
?x2
title
t_{16}
F.M.
Cluster Database
Author
A.name
?x3
P.name
?x4
area
t_3
DB

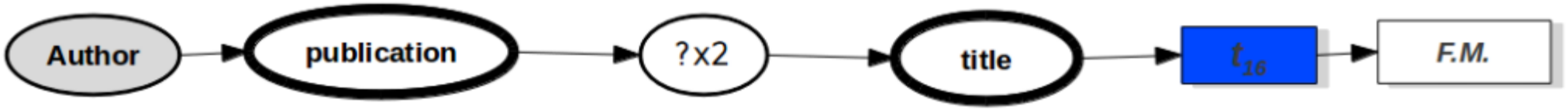
Backward Exploration

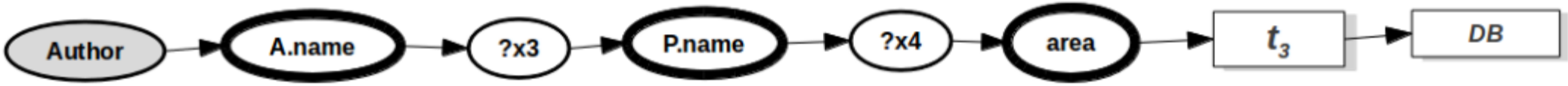
T_4: **Author**

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_5: **Publication**

	title	year
t_{15}	Data Integration	2002
t_{16}	Foundation Matters	2002





$$S = \{t_{16}\}$$

Backward Exploration

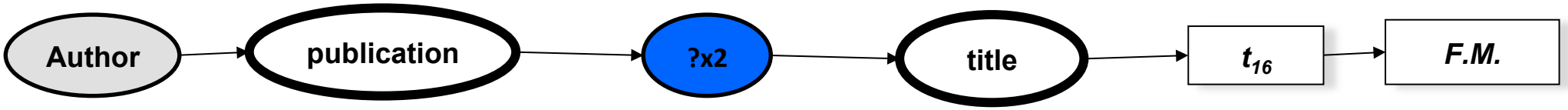
T_4: **Author**

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_5: **Publication**

	title	year
t_{15}	Data Integration	2002
t_{16}	Foundation Matters	2002

$$\sigma_{publication='Foundation\ Matters'}(Author)$$



$$S = \{t_{16}\}$$

Backward Exploration

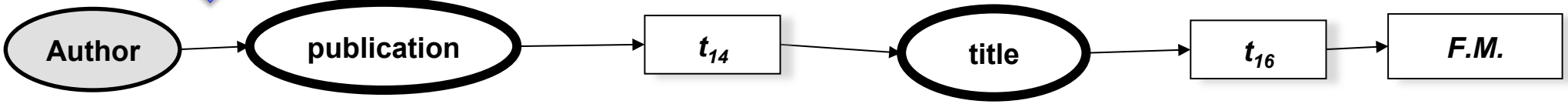
T_4: **Author**

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_5: **Publication**

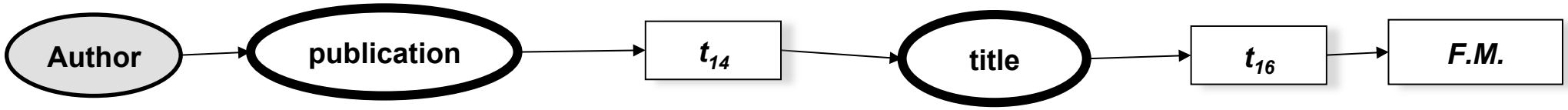
	title	year
t_{15}	Data Integration	2002
t_{16}	Foundation Matters	2002

$$\sigma_{publication='Foundation\ Matters'}(Author)$$



$$S = \{t_{14}, t_{16}\}$$

Backward Exploration

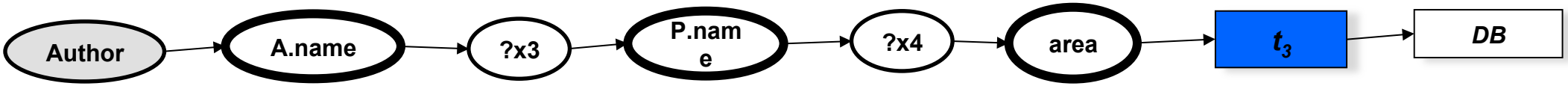


T_4: Author

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

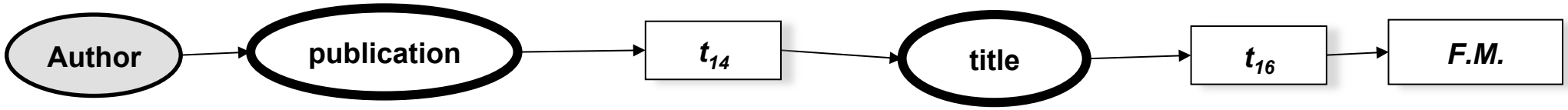
T_1: Person

	name	area
t_1	Watson	Database
t_2	Lenzerini	Database
t_3	Date	Database
t_4	Hunt	Inf. Systems



$$S = \{t_3, t_{14}, t_{16}\}$$

Backward Exploration



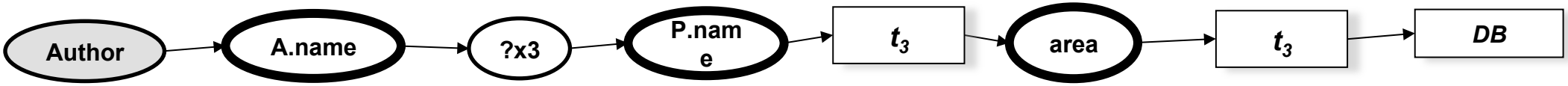
T_4: Author

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_1: Person

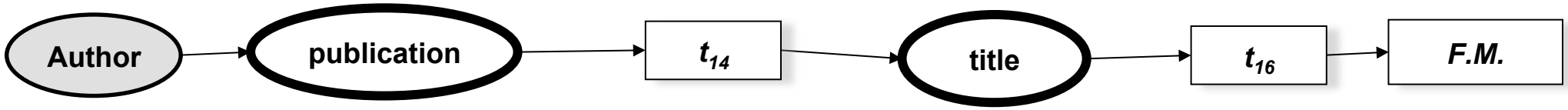
	name	area
t_1	Watson	Database
t_2	Lenzerini	Database
t_3	Date	Database
t_4	Hunt	Inf. Systems

$$\pi_{name}(\sigma_{tid=t_3}(Person))$$



$$S = \{t_3, t_{14}, t_{16}\}$$

Backward Exploration



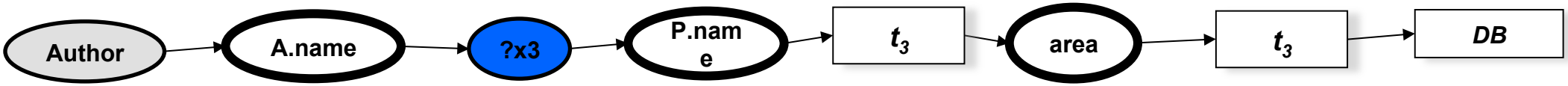
T_4: Author

	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_1: Person

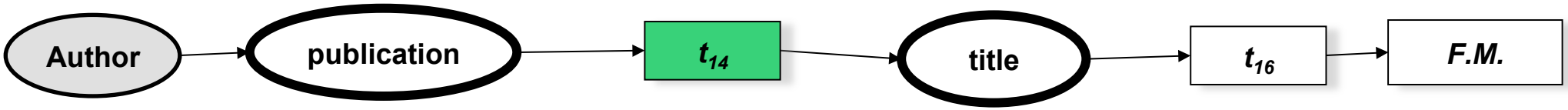
	name	area
t_1	Watson	Database
t_2	Lenzerini	Database
t_3	Date	Database
t_4	Hunt	Inf. Systems

$$\sigma_{name='Date'}(Author)$$



$S = \{t_3, t_{14}, t_{16}\}$

Backward Exploration



T_4: **Author**

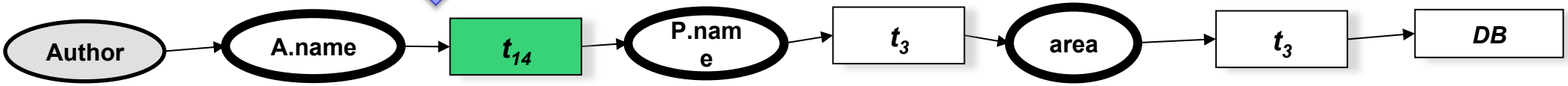
	name	publication
t_{13}	Lenzerini	Data Integration
t_{14}	Date	Foundation Matters

T_1: **Person**

	name	area
t_1	Watson	Database
t_2	Lenzerini	Database
t_3	Date	Database
t_4	Hunt	Inf. Systems

$$\sigma_{name='Date'}(Author)$$

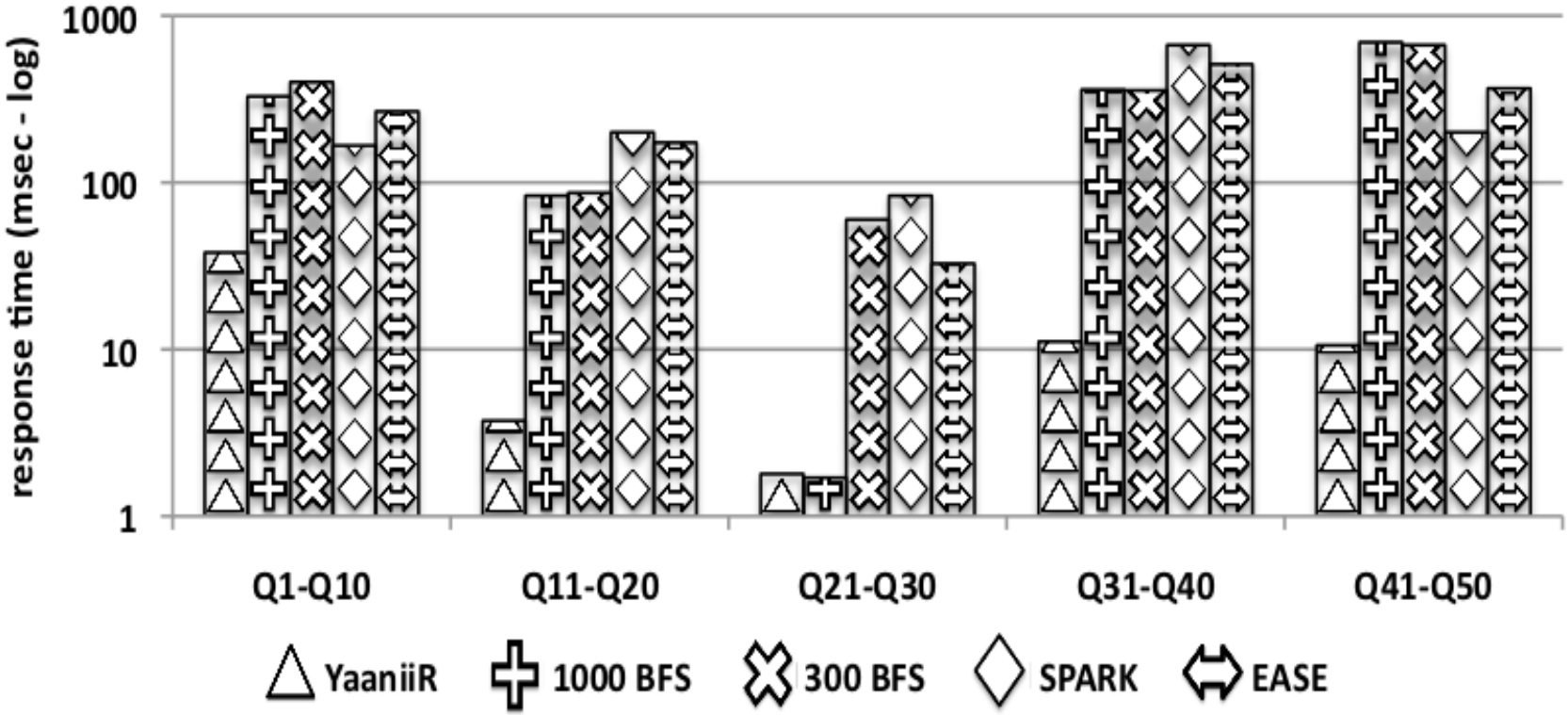
S = {t_3, t_{14}, t_{16}}



Implementation & Experiments

Implementation in **PostgreSQL 9.1** (*PL/pgSQL*)

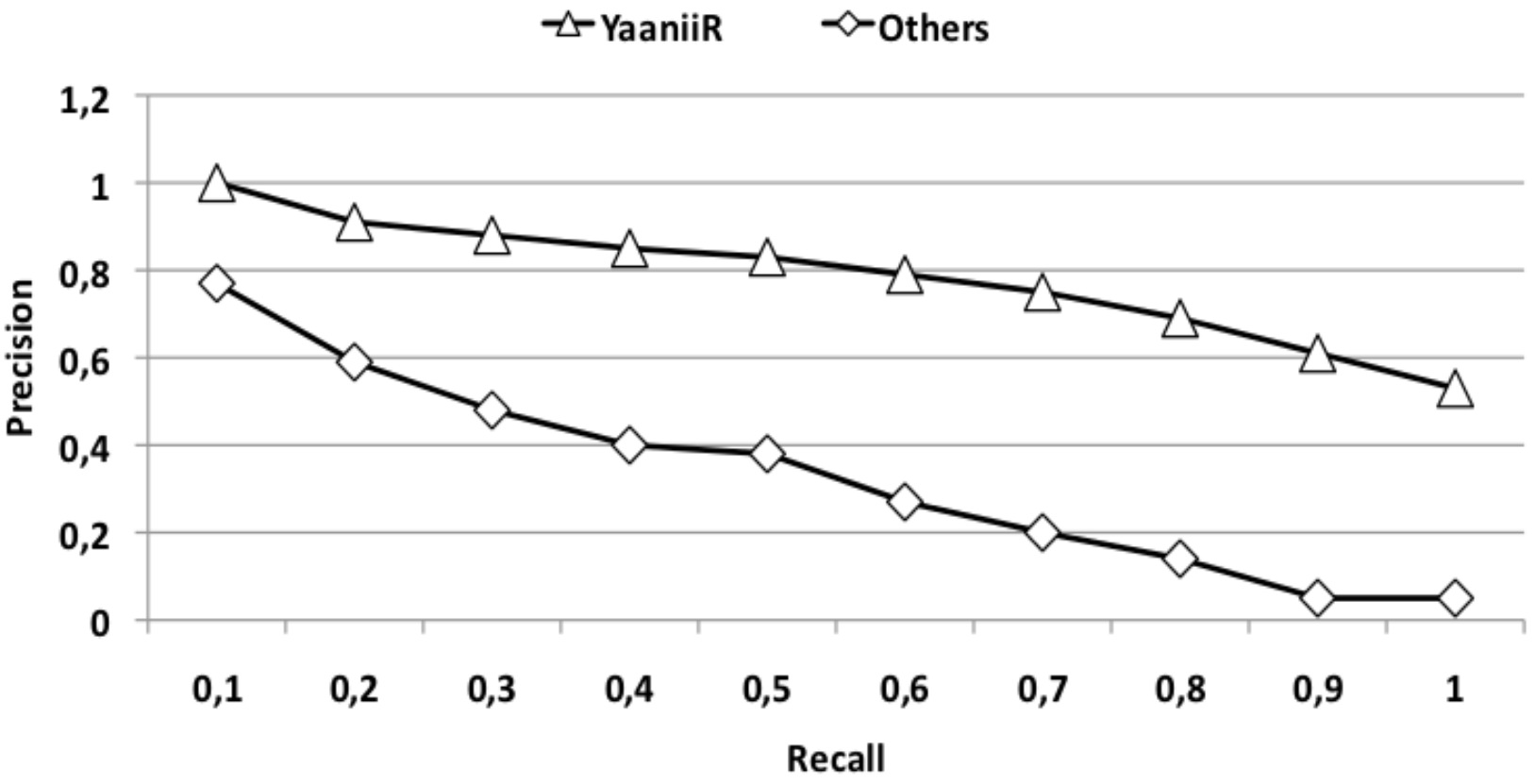
Tested with a common benchmark [*Coffman et al., CIKM 2010*]



Implementation & Experiments

Implementation in **PostgreSQL 9.1** (*PL/pgSQL*)

Tested with a common benchmark [*Coffman et al., CIKM 2010*]



Conclusion

- A strategy to **Structured Keyword Search** that:
  - Works exploiting the RDBMS engine
  - Retrieves solutions in order
  - Uses only operations of ***selection*** and ***projection***

Future Works

- *Forward exploration*.
- *Query Execution Plan* comparison with query rewriting approaches.
- Exploit *IR functionalities* on top of the approach.

Thanks for the attention…