Management of Semantic Web Data语义万维网

Zhihong Chong

http://cse.seu.edu.cn/people/zhchong/

WebDB.cn Open Group, Southeast University

Outline

- Problem
- Technique
- Improvement

Outline—Problem

- Model
  - triple/quadruple(三倍、四倍)
- Query
  - query/continuous query
  - pattern-matching
  - key-word tree query
- Architecture
  - central/distributed

Outline – Technique

- Schema oblivious/aware
- Vertical/horizontal segment
- B-tree Index平衡树索引
- Hash index散列索引

Triples

Subject (resource URI)	Predicate (property name)	Object (property value)

Fig. 1. Schema-oblivious representation

Property$_1$

Subject (resource URI)	Object (property value)

...

Property$_n$

Subject (resource URI)	Object (property value)

Class$_1$

Subject (resource URI)

...

Class$_m$

Subject (resource URI)

Fig. 2. Schema-aware representation

Subject	*FamilyName*	*GivenName*	*Phone*	*eMail*
ex:person1	Ding	Luping		lisading@WPI.EDU
ex:person2	Kuno	Harumi	123-456-7890	harumi.kuno@hp.com
ex:person3	Sayers	Craig		csayers@hpl.hp.com
ex:person4	Wilkinson	Kevin	123-555-7890	

Schema automation

Improvement:-Sparse Data Store

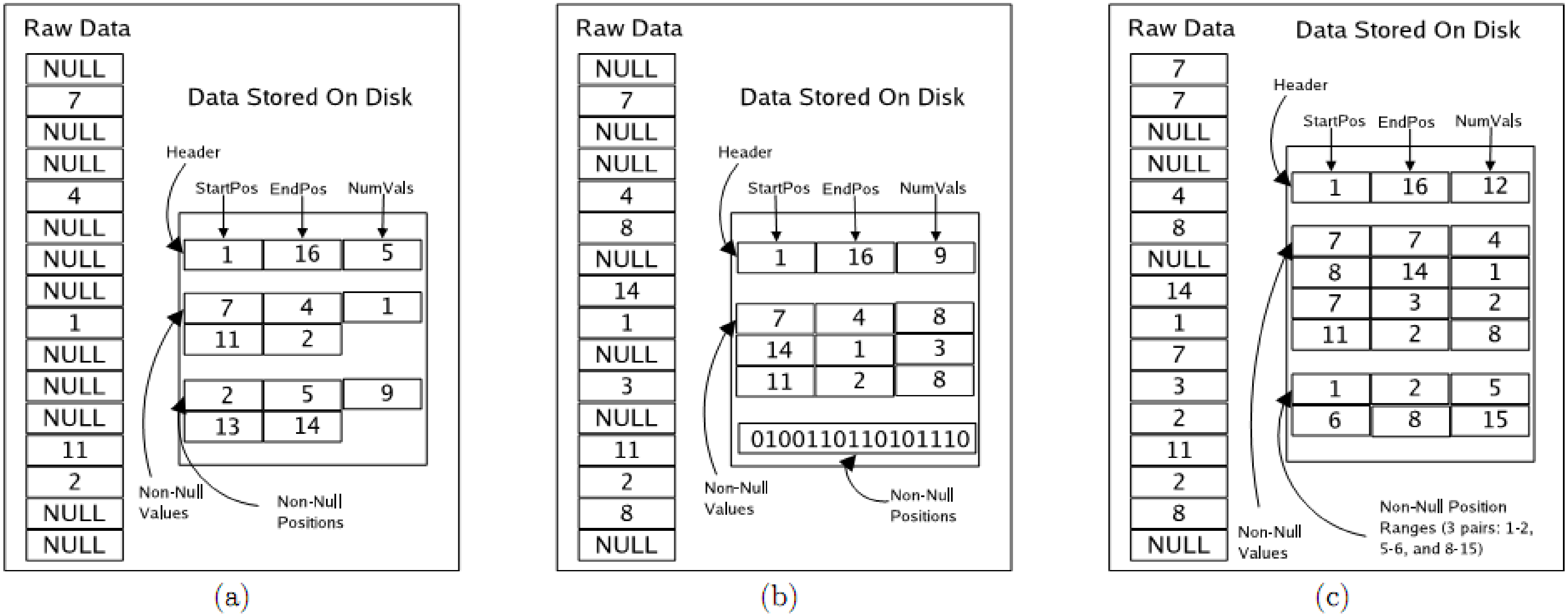


Figure 1: Positions represented using a list (a), a bit-string (b), and as ranges (c) for sparse columns

- Column-Stores For Wide and Sparse（稀疏） Data列存储
- CIDR2007

Oracle's Embedded Triple SQL-Query[1]

```
RDF_MATCH (
     Pattern          VARCHAR,
     Models           RDFModels,
     RuleBases        RDFRules,
     Aliases          RDFAliases,
   )
RETURNS AnyDataSet;
```

```
SELECT t.r reviewer, t.c conf, t.a age
FROM TABLE(RDF_MATCH(
  '(?r  rdf:type    Student)
   (?r  ReviewerOf  ?c)
   (?r  Age         ?a)',
  RDFModels ('reviewers'),
  NULL, NULL)) t
WHERE t.a < 25;
```

The RDF_MATCH invocation returns the following table:

r	c	c$type	a	a$type
John	IDBC2005	URI	24	xsd:int

- Graft RDF-Query into DBMS
  - Exploring DBMS capabilities
  - Materialized views as schema automation

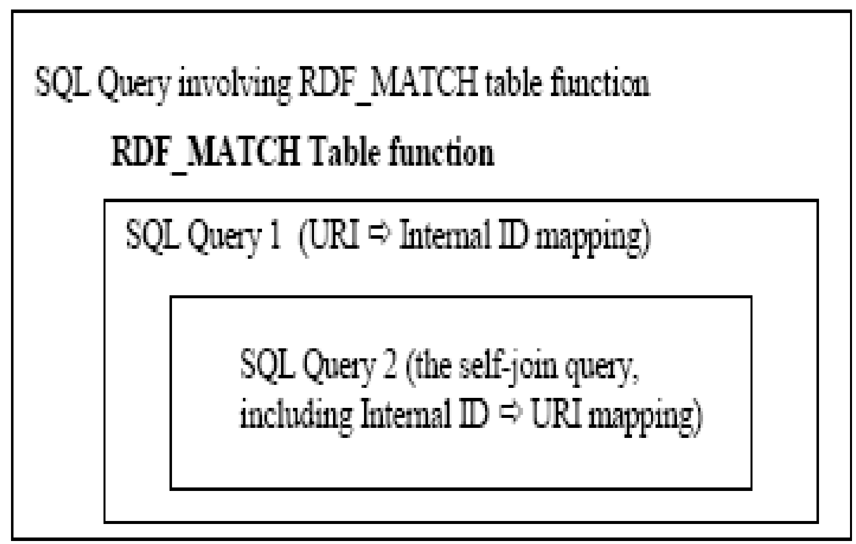


Figure 2: RDF_MATCH Implementation Overview

Beng Chin's Philosophy

Beng Chin approaches research problems and syste design with the philosophy that all algorithms an structures should be simple, elegant and yet efficient so that they can be easily grafted into existing systems and they are implementable, maintainable and scalable in actual applications. A good example would be his approach towards the design of new indexes; they are mainly B+-tree based -- simple and elegant in design, and efficient, robust and scalable in performance (eg. iDistance, Bx-tree, ST2B-tree).

http://www.comp.nus.edu.sg/~ooibc/

Hexastore: Sextuple Indexing for Semantic Web Data Management[2]

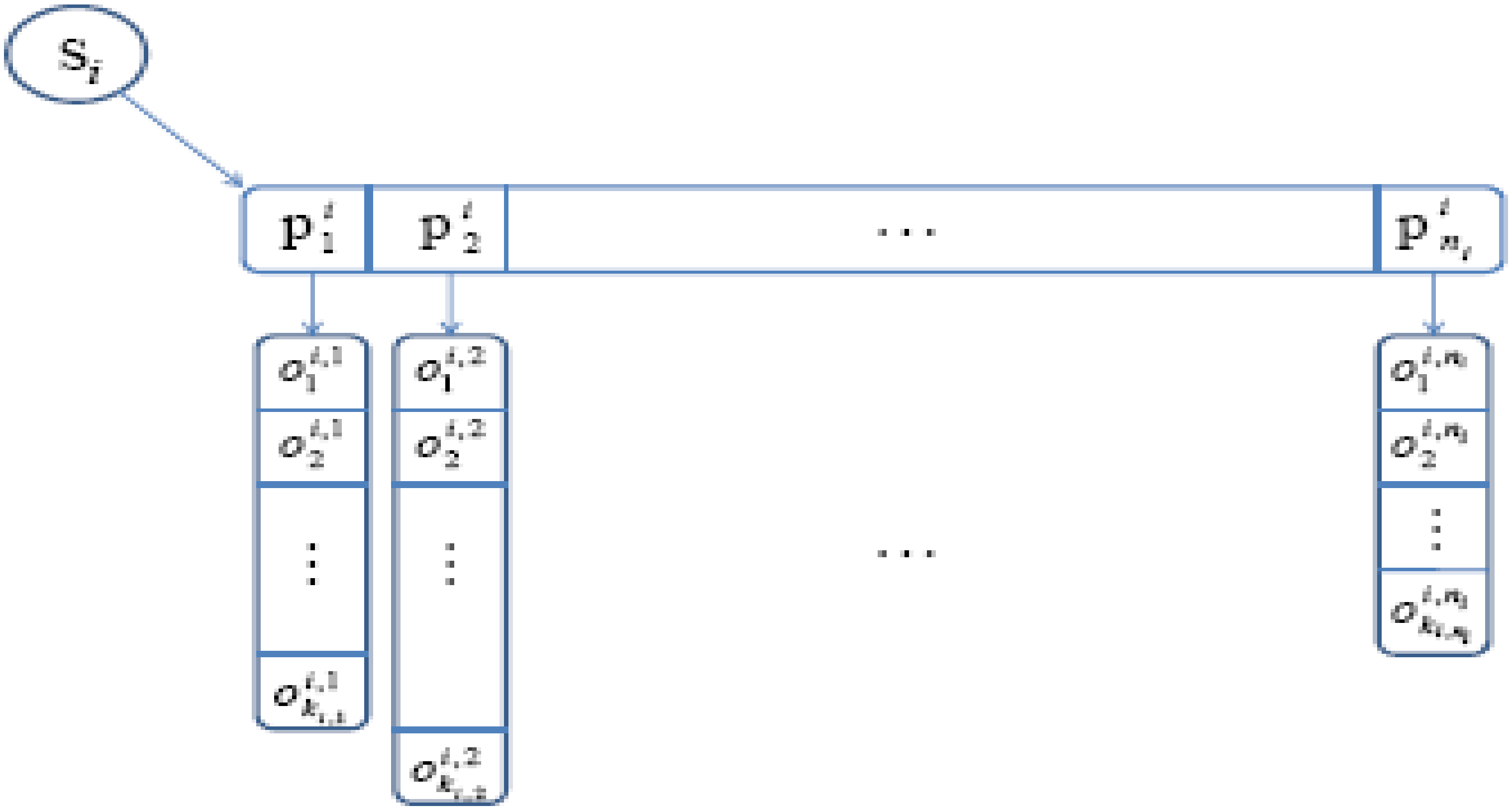


Figure 2: spo indexing in a Hexastore

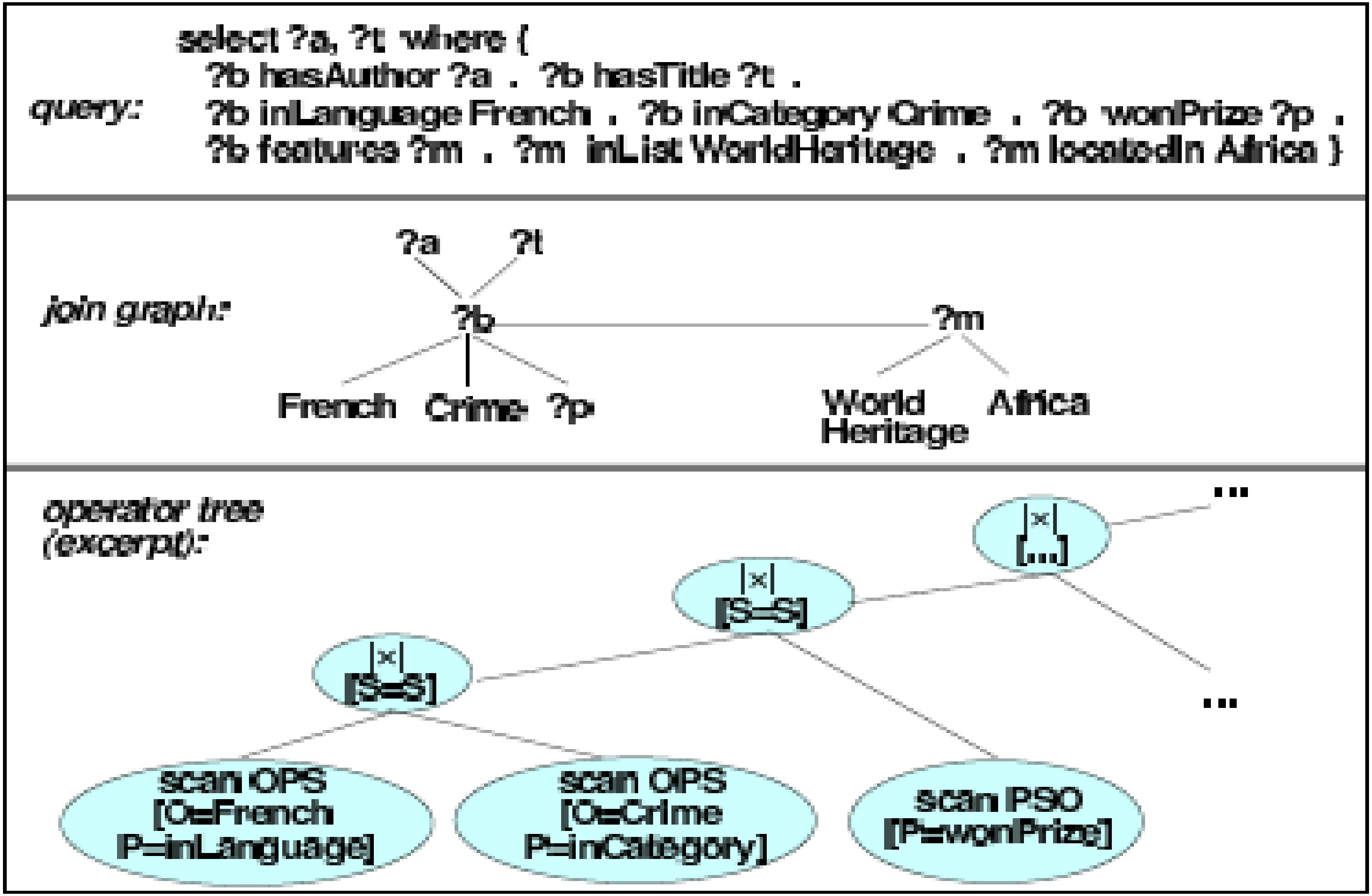


Figure 1: Query, Join Graph and Operator Tree

Sideway Information Passing[3]

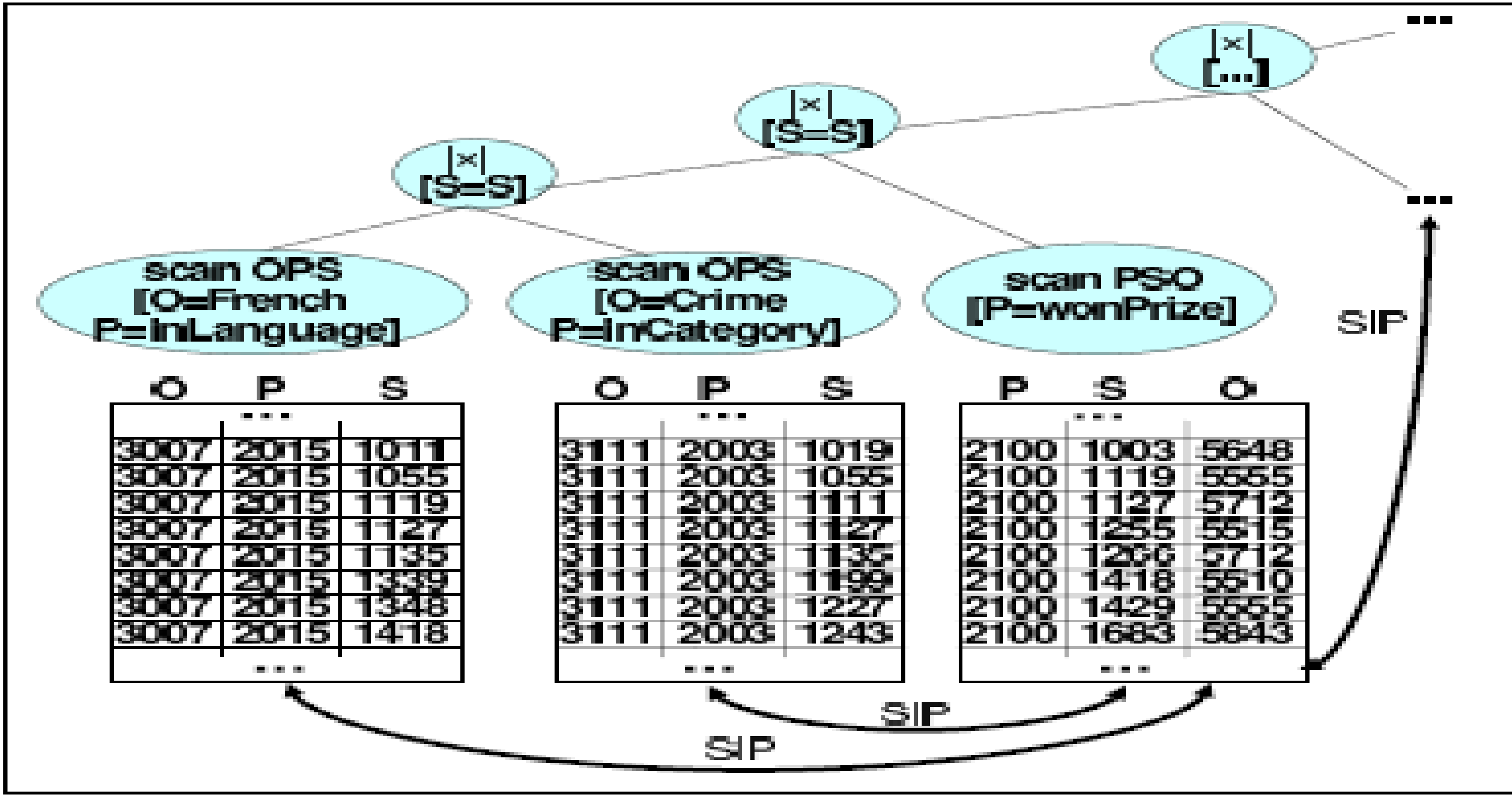


Figure 2: Operator Tree and its Index Inputs

Sideway Information Passing

1. Light-weight run-time method轻量级运行时方法
   - Run-time optimizing operator tree based light-weight statistics information collection
2. Pipelined executions流水线
3. Scale very well with increasing complex trees

Column-Store Support for RDF Data Management: not all swans are white[4]

Triple Patterns	
p1	(s, p, o)
p2	$(?s, p, o)$
p3	$(s, ?p, o)$
p4	$(s, p, ?o)$
p5	$(?s, ?p, o)$
p6	$(s, ?p, ?o)$
p7	$(?s, p, ?o)$
p8	$(?s, ?p, ?o)$

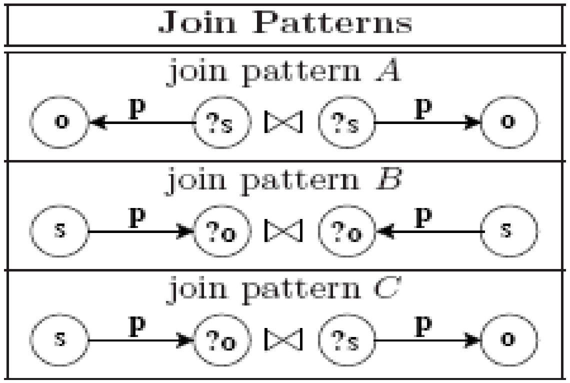


Figure 2: Simple RDF query patterns

Query	Pattern Coverage	
	Triple	Join
q1	p7	–
q2	p2,p8	A
q3	p2,p8	A
q4	p2,p8	A
q5	p2,p7	A, C
q6	p2,p7,p8	A, C
q7	p2,p7	A
q8	p6,p8	B

Table 2: Coverage of the query space

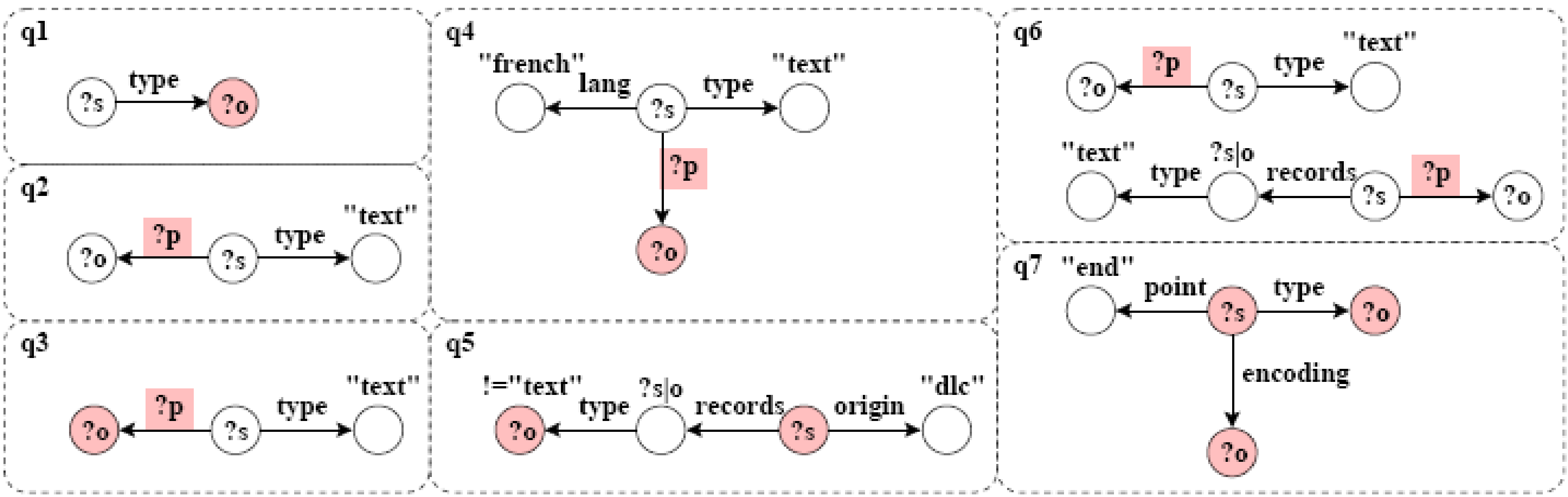


Figure 3: Graph Interpretation of queries q1 to q7

Column-Store Support for RDF Data Management: not all swans are white[4

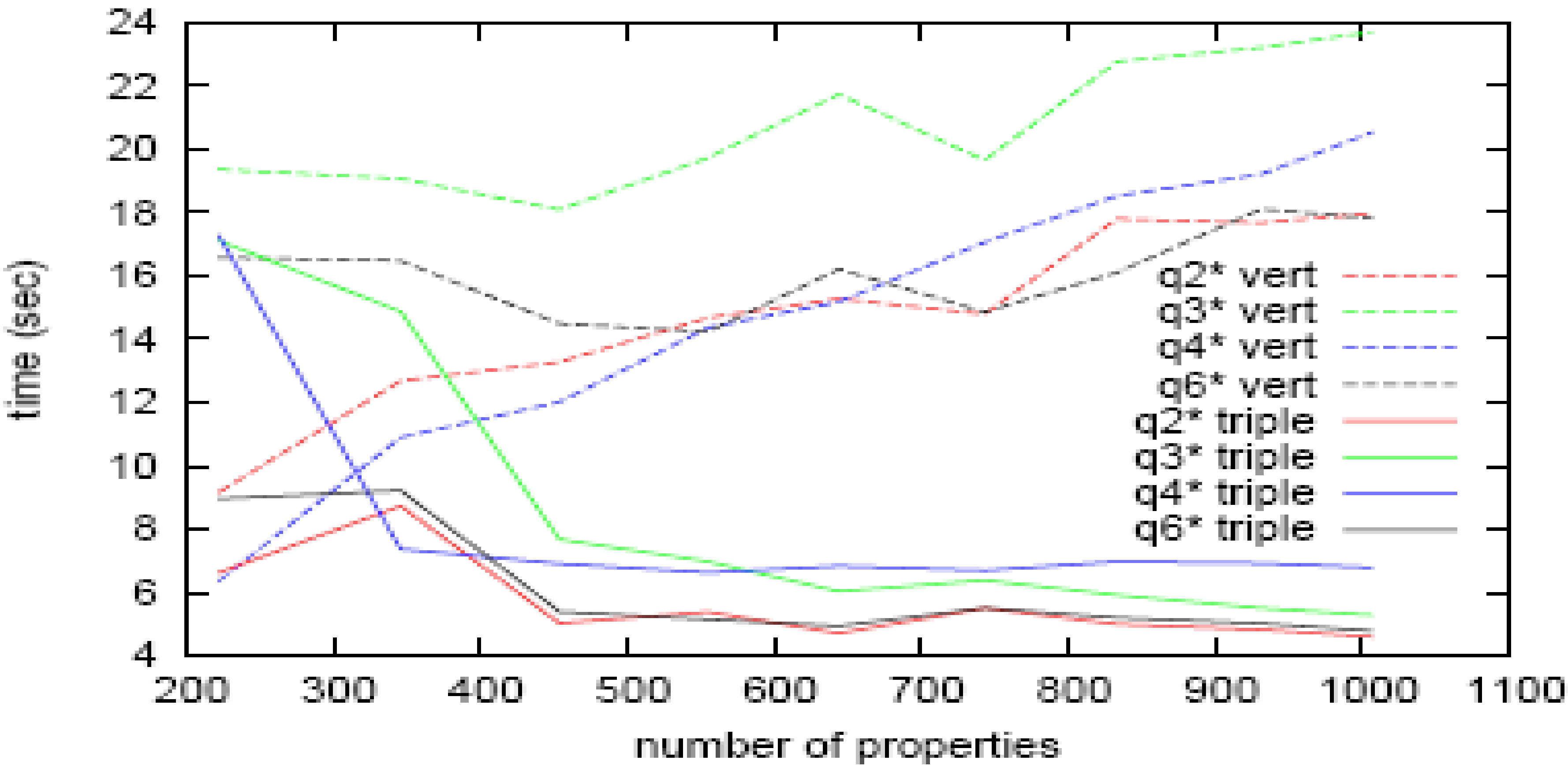


Figure 7: Scalability experiment

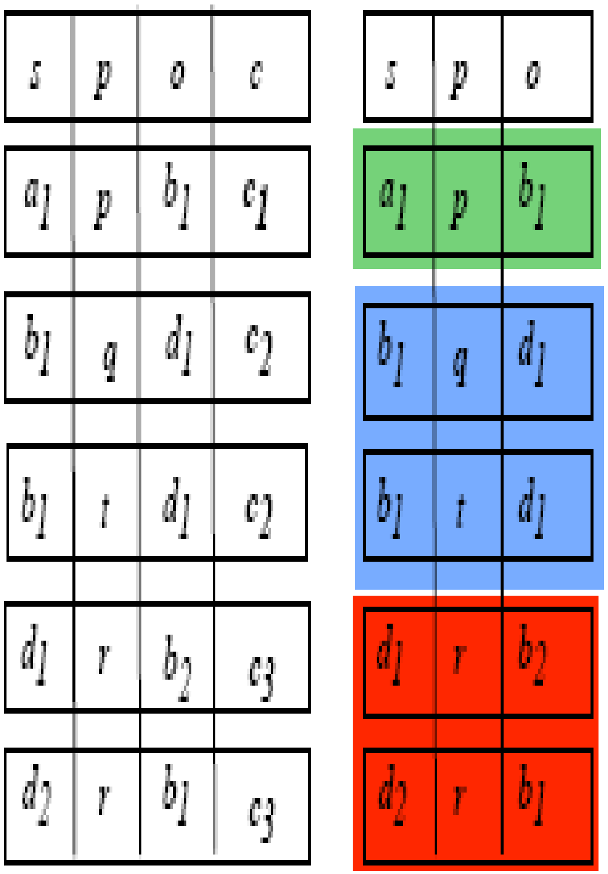


Fig. 1. Granularity Levels of Proven

	s	p	o	c
q_1	$\&NYT$	$endorses$	$\&B.\ Obama$	c_2
q_2	$\&NYT$	rdf:type	$Newspaper$	c_4
q_3	$Newspaper$	rdf:type	rdfs:Class	c_3
q_4	$Newspaper$	rdfs:subClassOf	$Mass\ Media$	c_3
q_5	$Mass\ Media$	rdfs:subClassOf	$Media$	c_5
q_6	$Candidate$	rdf:type	rdfs:Class	c_5
q_7	$\&B.\ Obama$	rdf:type	$Candidate$	c_5
q_8	$endorses$	rdf:type	rdf:Property	c_1
q_9	$endorses$	rdfs:domain	$Newspaper$	c_1
q_{10}	$endorses$	rdfs:range	$Candidate$	c_1
q_{11}	$Candidate$	rdfs:subClassOf	$Person$	c_1
q_{12}	$supports$	rdf:type	rdf:Property	c_1
q_{13}	$supports$	rdfs:domain	$MassMedia$	c_1
q_{14}	$supports$	rdfs:range	$Person$	c_1
q_{15}	$endorses$	rdfs:subPropertyOf	$supports$	c_2
q_{16}	$\&NYT$	$endorses$	$\&B.\ Obama$	c_1
q_{17}	$Media$	rdf:type	rdfs:Class	c_5

Fig. 2. Relation $Q(s, p, o, c)$

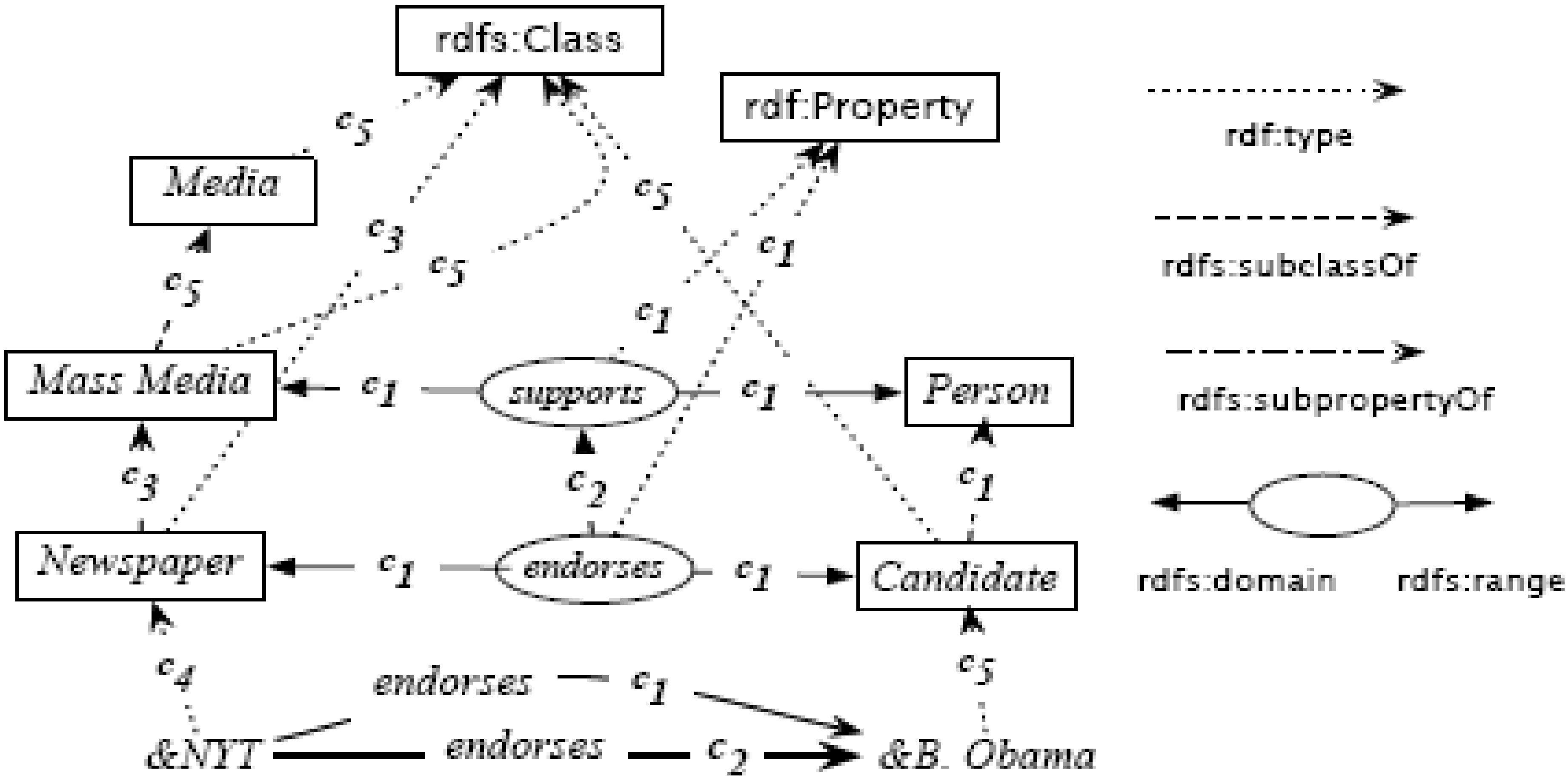
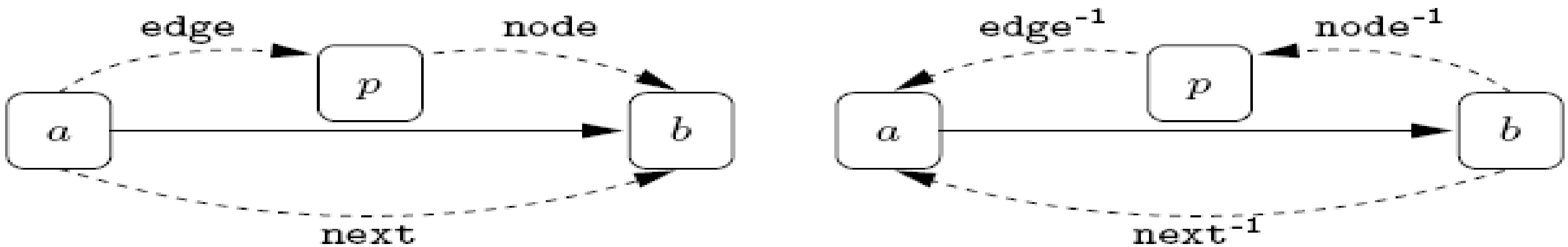


Fig. 3. Graph representation of $Q(s, p, o, c)$

Beyond Triples: Quadruple Storage

To my knowledge, no reported research on the storage of quadruples by now!

1. Sextuple Indexing may not be efficient
2. Complex queries on quadruples
3. It is related with uncertain data management, a hot research point.
   - (s, p, o, p’), where p’ means the possibility of (s, p, o).



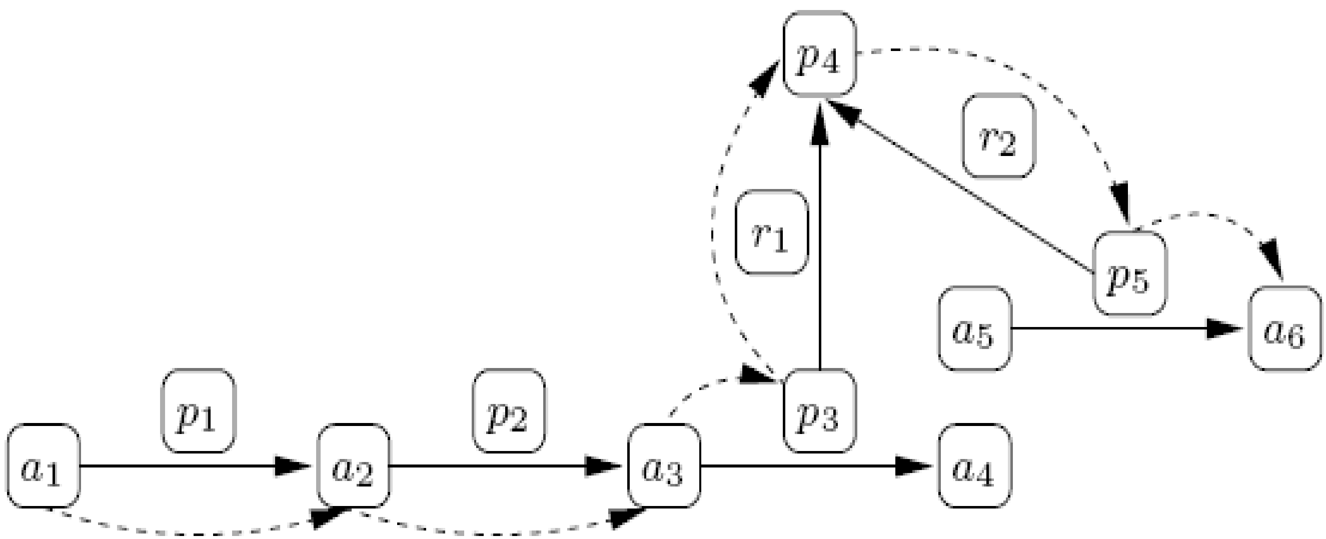
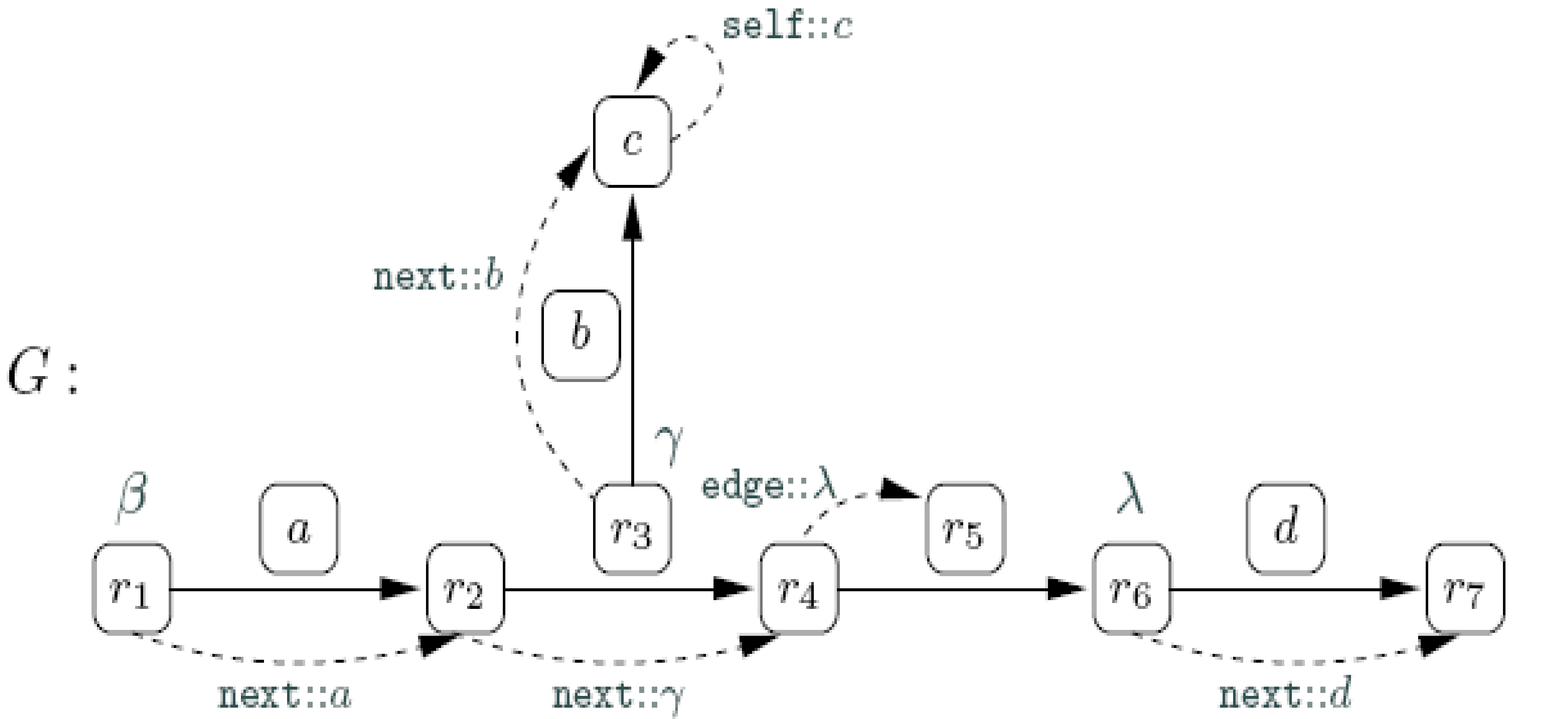


Fig. 4. Nodes a_1 and a_6 are connected by a path that follows the sequence of navigational axes `next/next/edge/next/next`$^{-1}$`/node`



$$\beta = \mathtt{next}::a/(\mathtt{next}::[\mathtt{next}::b/\mathtt{self}::c])^{*}/(\mathtt{edge}::[\mathtt{next}::d] \mid \mathtt{next}::a)^{+}$$

SPARQ2L:Towards support for subgraph extraction queries in RDF Database[9]

```
SELECT ??p
WHERE   {   ?x  ??p  ?x .
            ?z  compound:name  " Methionine" .
            PathFilter(containsAny(??p, ?z) ) }
```

```
SELECT ??p
WHERE   {   ?x  ??p  ?y .
            ?x  foaf:name  "salesPersonA".
            ?y  company:is_CIO ?z.
            ?z  company:name "CompanyY" .
            PathFilter( cost(??p) < 4 )      }
```

Complex Query

To my knowledge, no reported research on complex query in database literature!

1. Long join chain
2. Complex star join
3. Support inference
   - Materialization method may not be available?
   - Sideway information passing?
   - Others??

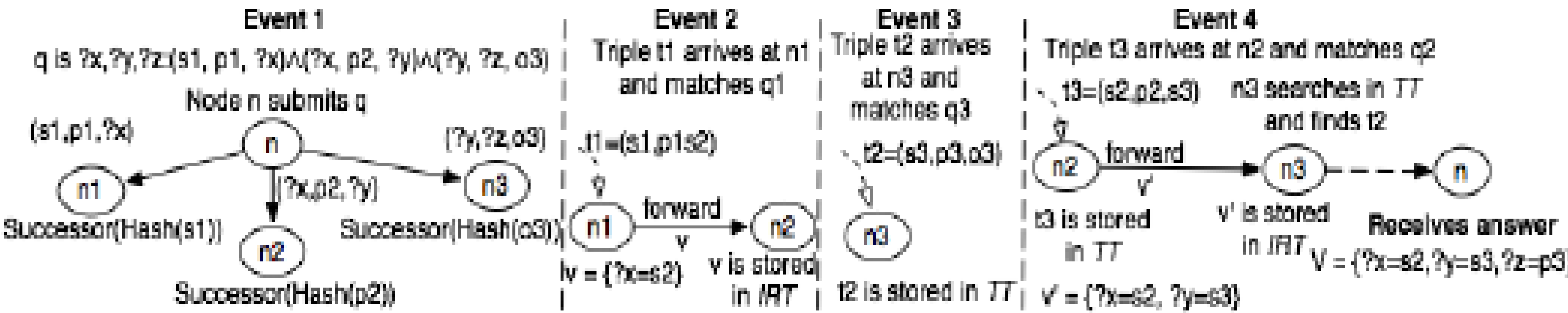


Fig. 1. The algorithm CQC in operation

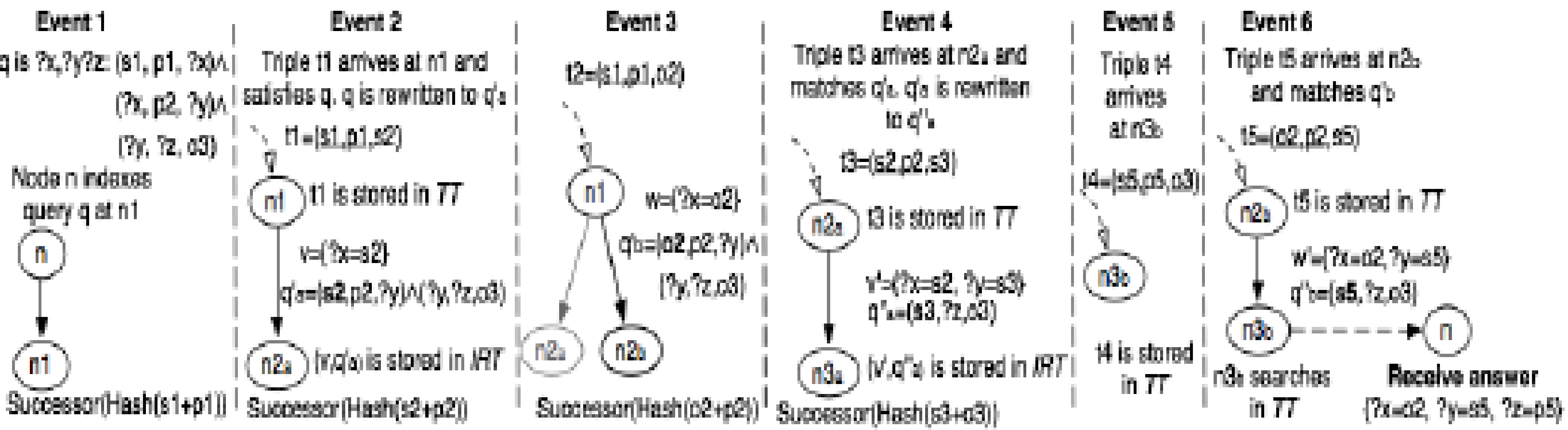


Fig. 2. The algorithm CSBV in operation

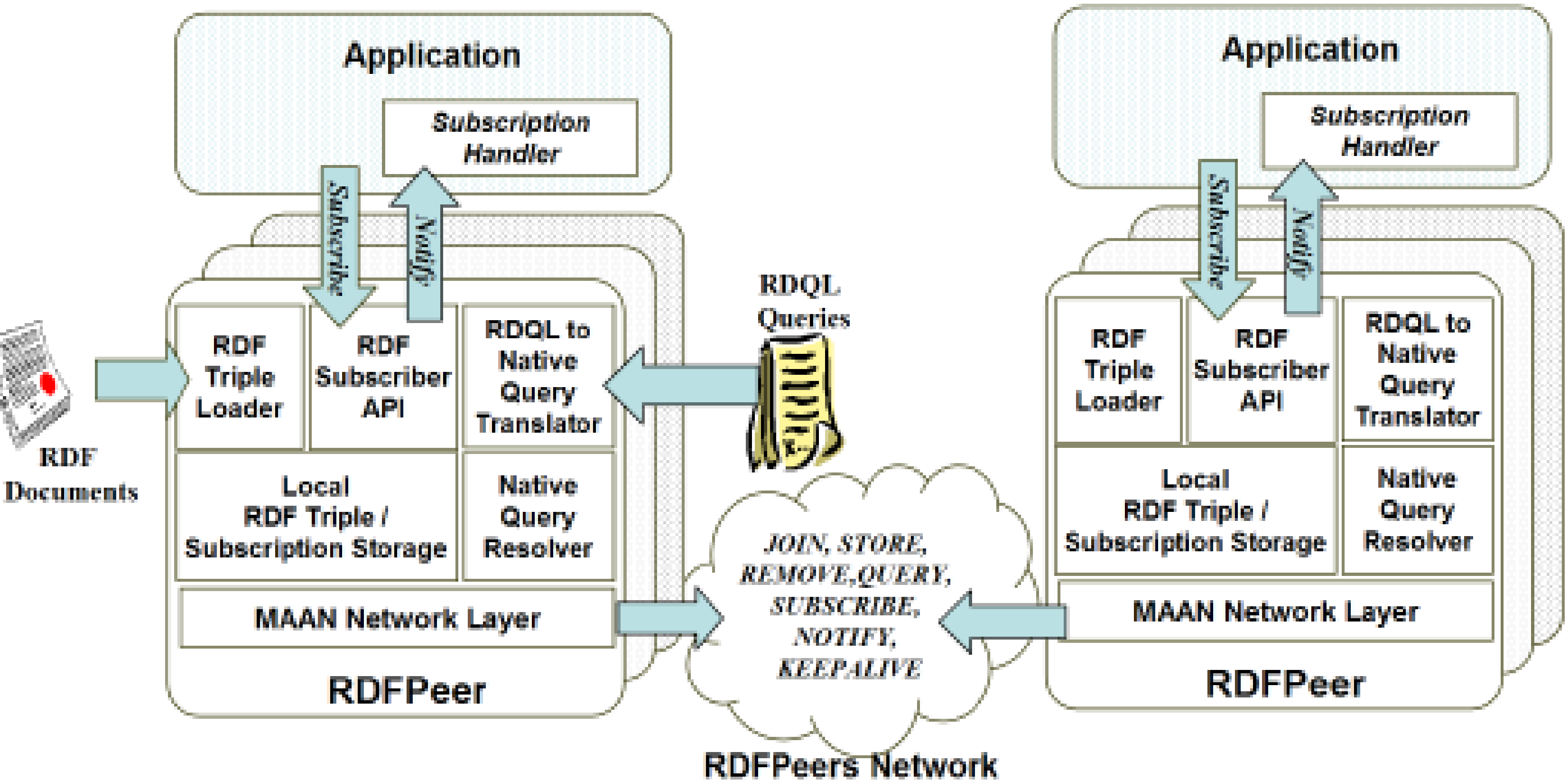


Fig. 1. The Architecture of RDFPeers

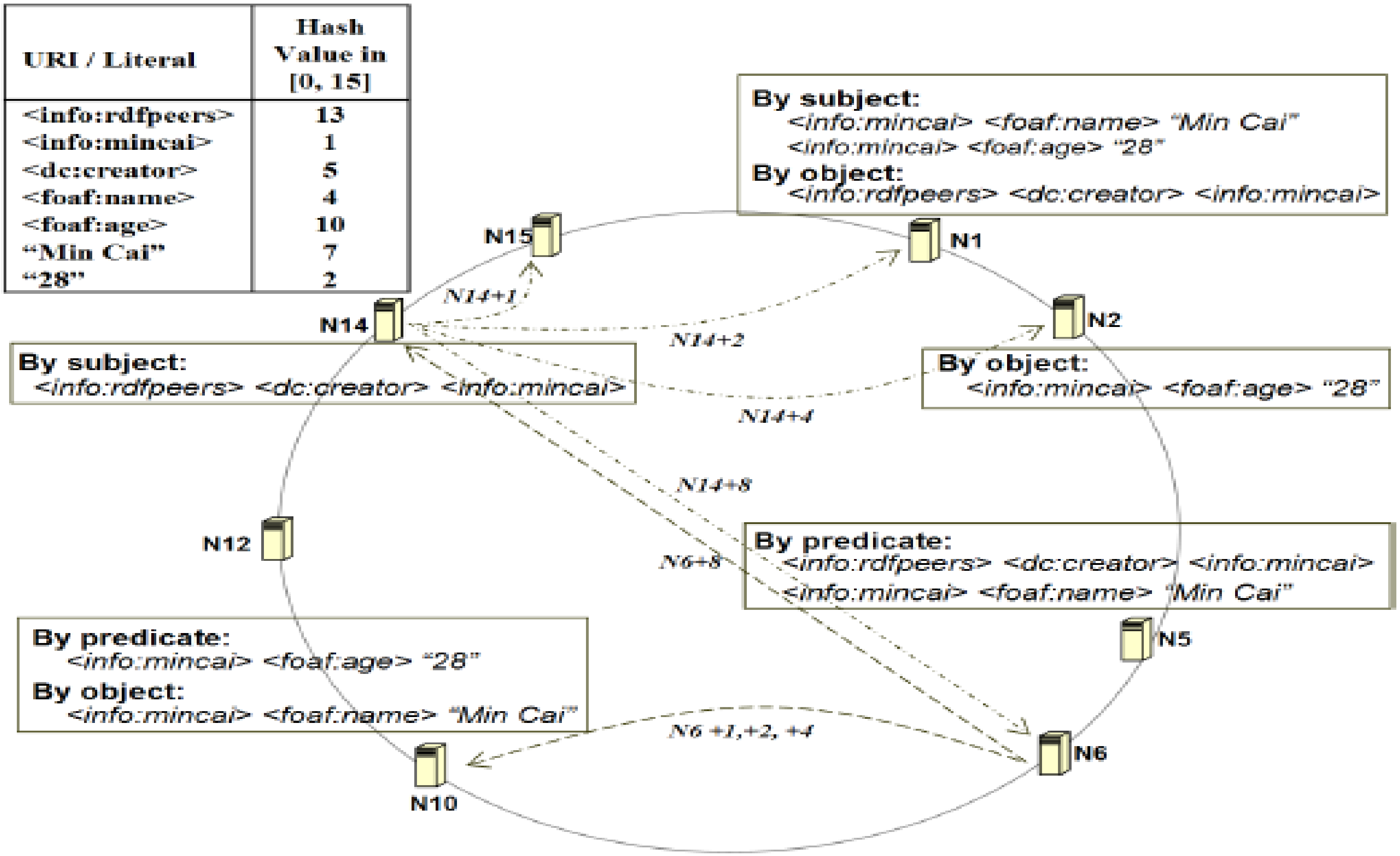


Fig. 3. Storing three triples into an RDFPeers network of eight nodes in an example 4-bit identifier space that could hold up to 16 nodes. (In reality a much larger identifier space is used, such as 128 bits.)

No.	Query Pattern	Cost	Query Semantics
Q1	$(?s, ?p, ?o)$	$O(N)$	find all possible triples
Q2	$(?s, ?p, o_i)$	$\log N$	given object o_i of any predicate, find the subjects and predicates of matching triples
Q3	$(?s, p_i, ?o)$	$\log N$	given predicate p_i, find the subjects and objects of the triples having this predicate
Q4	$(?s, p_i, o_i)$	$\log N$	given object o_i of predicate p_i, find the subjects of matching triples
Q5	$(s_i, ?p, ?o)$	$\log N$	given subject s_i, find all predicates and objects of the resource identified by s_i
Q6	$(s_i, ?p, o_i)$	$\log N$	given subject s_i, find its predicate that has object o_i
Q7	$(s_i, p_i, ?o)$	$\log N$	given subject s_i, find its object of predicate p_i
Q8	(s_i, p_i, o_i)	$\log N$	return this triple if it exists otherwise return nothing

Table 1
The eight possible atomic triple queries for exact matches. The cost is measured in the number of routing hops needed to resolve each query.

Reference

1. An efficient SQL-based RDF Querying Schema, VLDB’05
2. Hexastore: Sextuple Indexing for Semantic Web Data Management VLDB’08
3. Scalable Join Processing on Very Large RDF Graphs, SIGMOD’09
4. Column-Store Support for RDF Data Management: not all swans are white, VLDB’08
5. Coloring RDF Triples to Capture Provenance, ISWC’09
6. Foundations of RDF database, Reasoning Web 2009
7. Continuous RDF Query Processing over DHTs, ISWC’07
8. A Subscribable Peer-to-Peer RDF Repository for Distributed Metadata Management, Web Semantics: Science, Services and Agents on the World Wide Web 2004
9. SPARQ2L:Towards support for subgraph extraction queries in RDF Database, WWW’07

Thanks☺