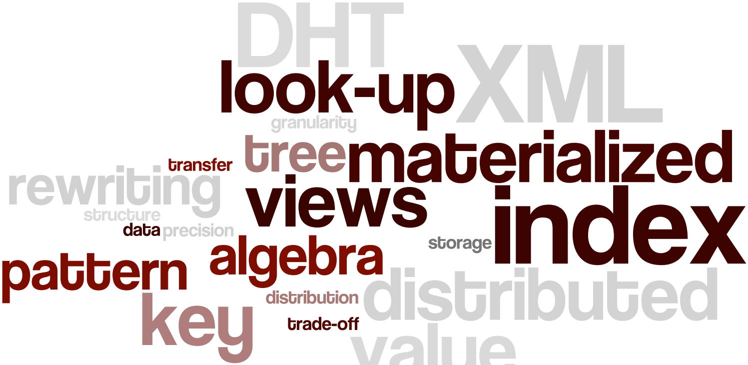
LiquidXML: distributed XML management in P2P

Ioana Manolescu

Gemo/IASI group
INRIA Saclay–Île-de-France and LRI, Université de Paris Sud-11

December 3, 2009

This talk at a glance



Background

1998-2001	PhD at INRIA Rocquencourt with Daniela Florescu: XQuery rewriting, XML materialized views
2002	PostDoc in Politecnico di Milano, Italy
2003-now	*Chargéee de Recherche* INRIA, Gemo group. Lots of research and teaching on (distributed) XML
2005-2006	W3C XQuery Working group (Update task force)

Distributed data management

Old goal (1970)

Distributed data management

Old goal (1970)

- distributed versions of industrial-strength DBMSs
- massively parallel with map/reduce

Distributed data management

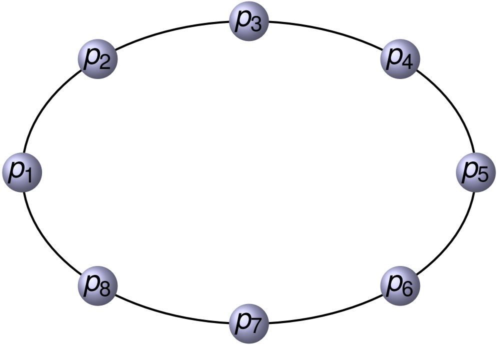
Old goal (1970)

- distributed versions of industrial-strength DBMSs
- massively parallel with map/reduce

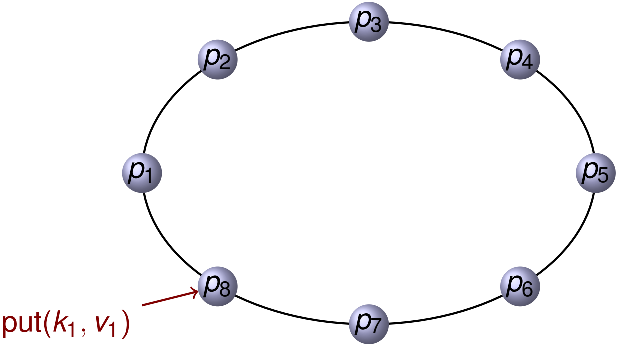
Still missing: the flexible federation

- high independence of the sites: when to be in, what to store
- data distribution transparency
- . . . with the usual performance requirements

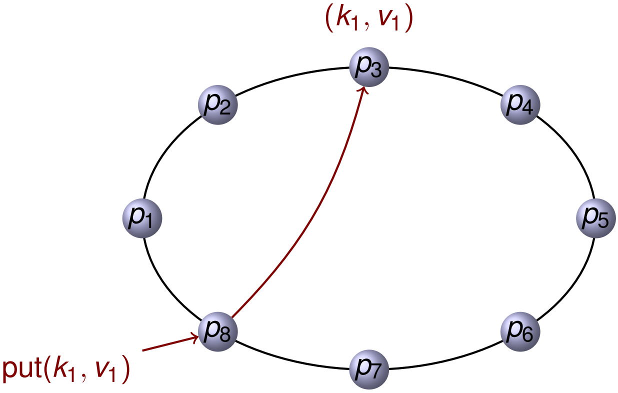
Distributed hash tables



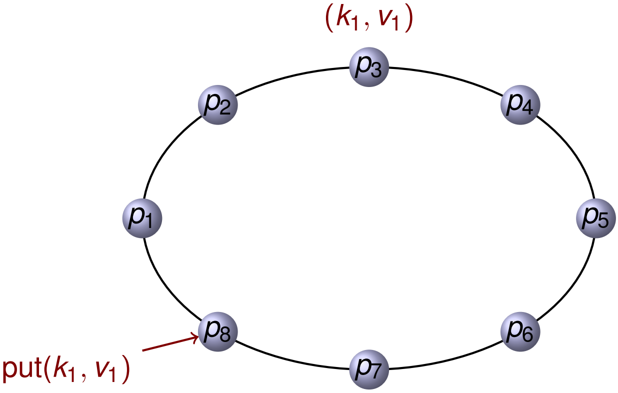
Distributed hash tables



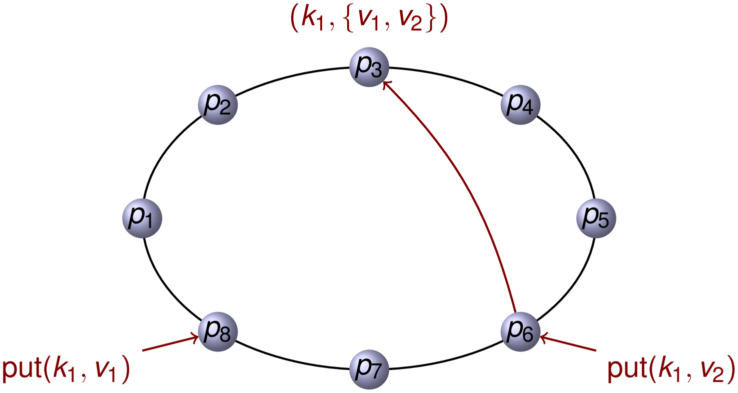
Distributed hash tables



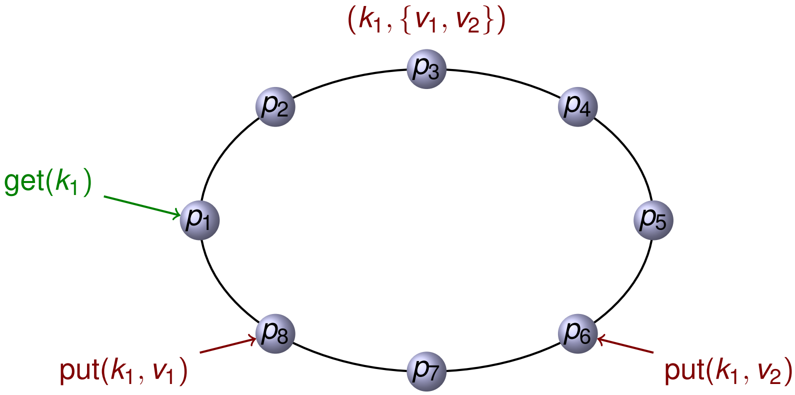
Distributed hash tables



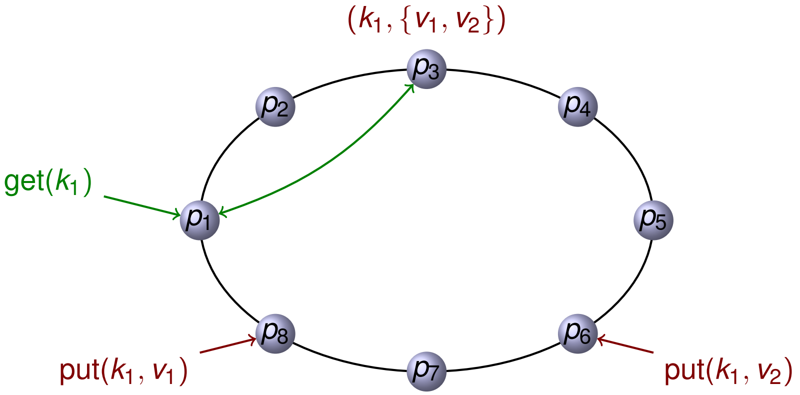
Distributed hash tables



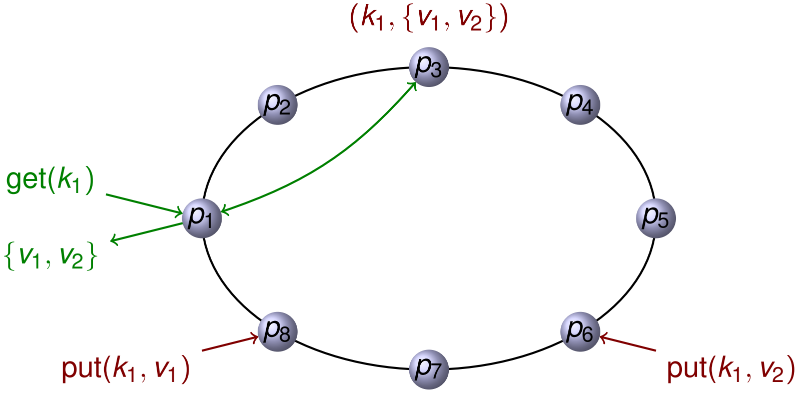
Distributed hash tables



Distributed hash tables



Distributed hash tables



From DHTs to distributed data management

DHTs provide:

- logical network maintenance
- efficient message routing
- shared (key, value) repository

From DHTs to distributed data management

DHTs provide:

- logical network maintenance
- efficient message routing
- shared (key, value) repository

Stil need:

- data indexing algorithms

From DHTs to distributed data management

DHTs provide:

- logical network maintenance
- efficient message routing
- shared (key, value) repository

Stil need:

- data indexing algorithms
- storage for application data and even DHT index data

From DHTs to distributed data management

DHTs provide:

- logical network maintenance
- efficient message routing
- shared (key, value) repository

Stil need:

- data indexing algorithms
- storage for application data and even DHT index data
- local query processing

From DHTs to distributed data management

DHTs provide:

- logical network maintenance
- efficient message routing
- shared (key, value) repository

Stil need:

- data indexing algorithms
- storage for application data and even DHT index data
- local query processing
- distributed query processing: operators, including data transfers, optimization . . .

Building XML stores on DHTs

- Peers retain control over the data they store/publish
  - no global schema
  - documents published independently
  - annotations, triples, links can freely connect content
- peers collaborate for storing the index
- load balancing

Building XML stores on DHTs

- Peers retain control over the data they store/publish
  - no global schema
  - documents published independently
  - annotations, triples, links can freely connect content
- peers collaborate for storing the index
- load balancing

Systems

- XML indexing: KadoP [AMP05, AMP+08]

Building XML stores on DHTs

- Peers retain control over the data they store/publish
  - no global schema
  - documents published independently
  - annotations, triples, links can freely connect content
- peers collaborate for storing the index
- load balancing

Systems

- XML indexing: KadoP [AMP05, AMP+08]
- XML materialized views in P2P networks: ViP2P [MZ09] http://vip2p.saclay.inria.fr

ViP2P

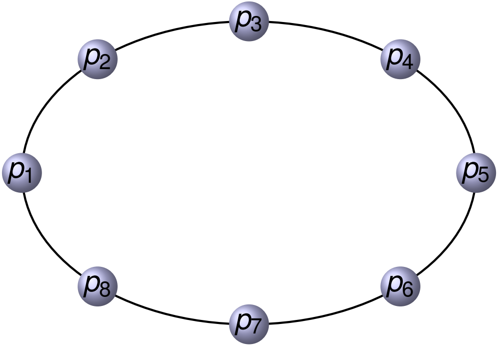
Functionalities

1. Re-distribute XML content to subscriber queries (XPath and more)
2. Answer snapshot queries based on the existing subscriptions

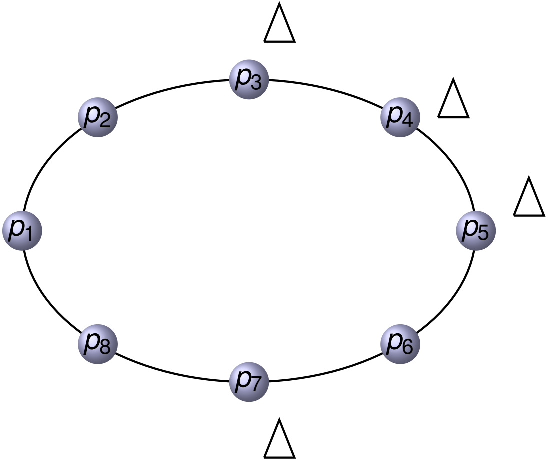
Joint work with:

S. Zoupanos, A. Tilea, K. Karanasos, J. Camacho-Rodriguez, A. Katsifodimos, S. Julean, J. Leblay

ViP2P: views in peer-to-peer



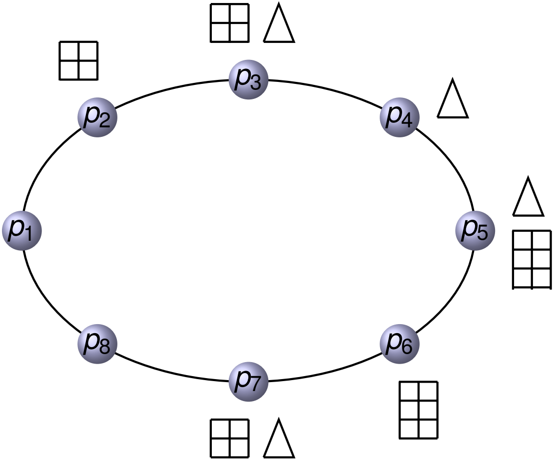
ViP2P: views in peer-to-peer



The peers may store:

- documents

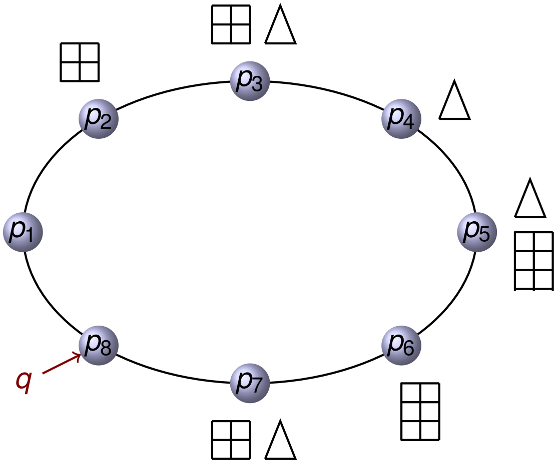
ViP2P: views in peer-to-peer



The peers may store:

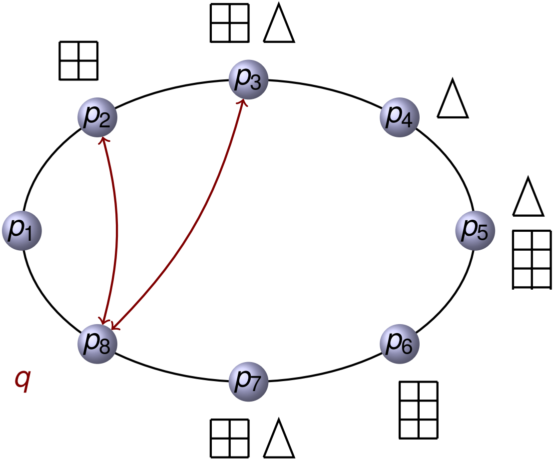
- documents
- views

ViP2P: views in peer-to-peer



When q arrives:

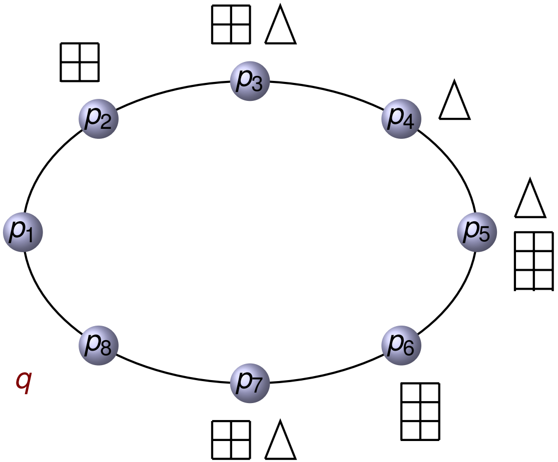
ViP2P: views in peer-to-peer



When *q* arrives:

- view definition lookup

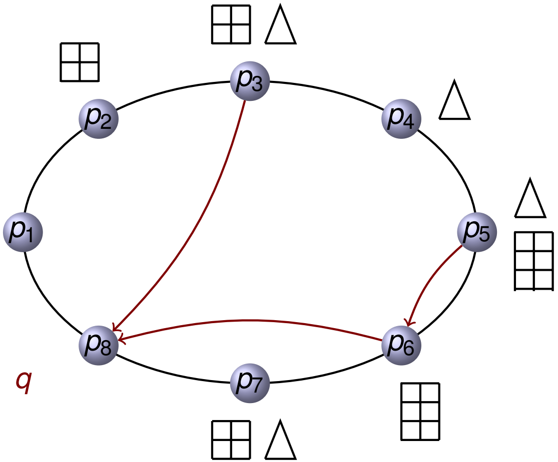
ViP2P: views in peer-to-peer



When q arrives:

- view definition lookup
- rewriting

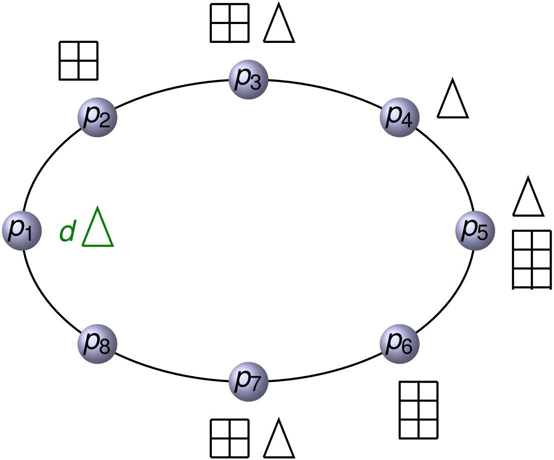
ViP2P: views in peer-to-peer



When q arrives:

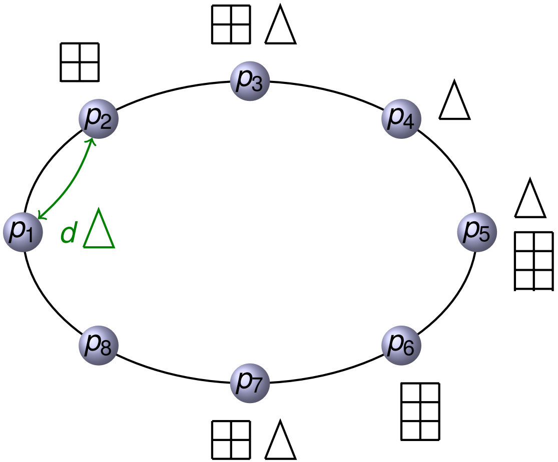
- view definition lookup
- rewriting
- execution of physical plan

ViP2P: views in peer-to-peer



When *d* arrives:

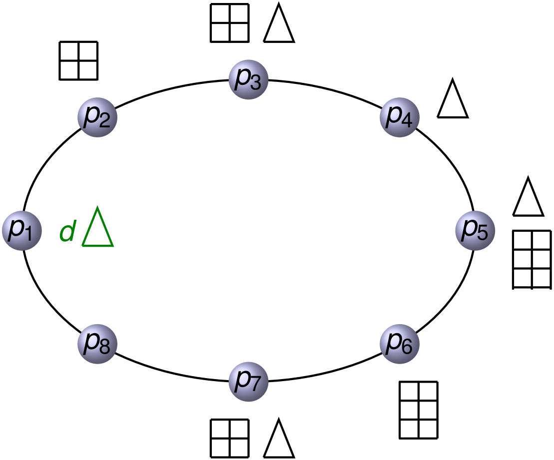
ViP2P: views in peer-to-peer



When *d* arrives:

- search view definitions for which $v_i(d) \neq \emptyset$

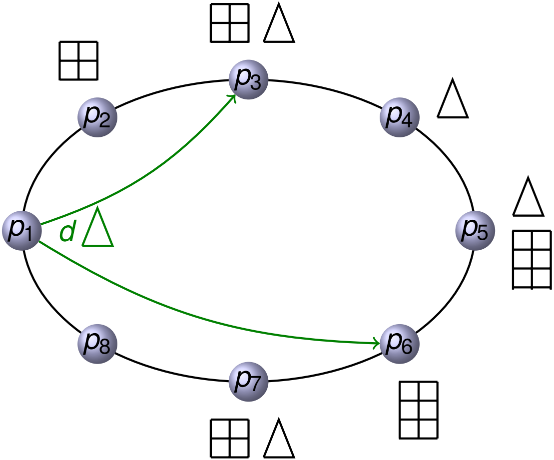
ViP2P: views in peer-to-peer



When d arrives:

- search view definitions for which $v_i(d) \neq \emptyset$
- compute $v_i(d)$

ViP2P: views in peer-to-peer



When d arrives:

- search view definitions for which $v_i(d) \neq \emptyset$
- compute $v_i(d)$
- send results

View and query language

Tree pattern language extending XPath

1. Parent-child and ancestor-descendant axis
2. Nodes: element, attribute, word in text
3. From each node, one may store:
   - full image (content)
   - text value
   - identifier

View and query language

Tree pattern language extending XPath

1. Parent-child and ancestor-descendant axis
2. Nodes: element, attribute, word in text
3. From each node, one may store:
    - full image (content)
    - text value
    - identifier

More features (nesting and optionality) get closer to XQuery

View and query language

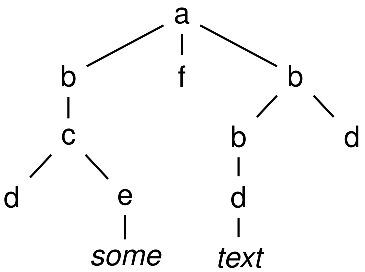
Tree pattern language extending XPath

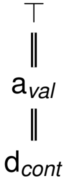
1. Parent-child and ancestor-descendant axis
2. Nodes: element, attribute, word in text
3. From each node, one may store:
    - full image (content)
    - text value
    - identifier

More features (nesting and optionality) get closer to XQuery

Tree pattern = the data needs of a query

Sample view (1)

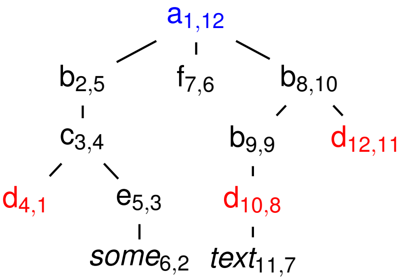


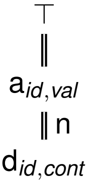


for $x in //b,
$z in $x//d
return $x/text(), $y

a_{val}	d_{cont}
some text	⟨d/⟩
some text	⟨d⟩text⟨/d⟩
some text	⟨d/⟩

Sample view (2)





<table>
<tr><th>a_{id}</th><th>a_{val}</th><th></th></tr>
<tr><td>1,12</td><td>some text</td><td>
<table>
<tr><th>d_{id}</th><th>d_{cont}</th></tr>
<tr><td>4,1</td><td>⟨d/⟩</td></tr>
<tr><td>10,8</td><td>⟨d⟩text⟨/d⟩</td></tr>
<tr><td>12,11</td><td>⟨d/⟩</td></tr>
</table>
</td></tr>
</table>

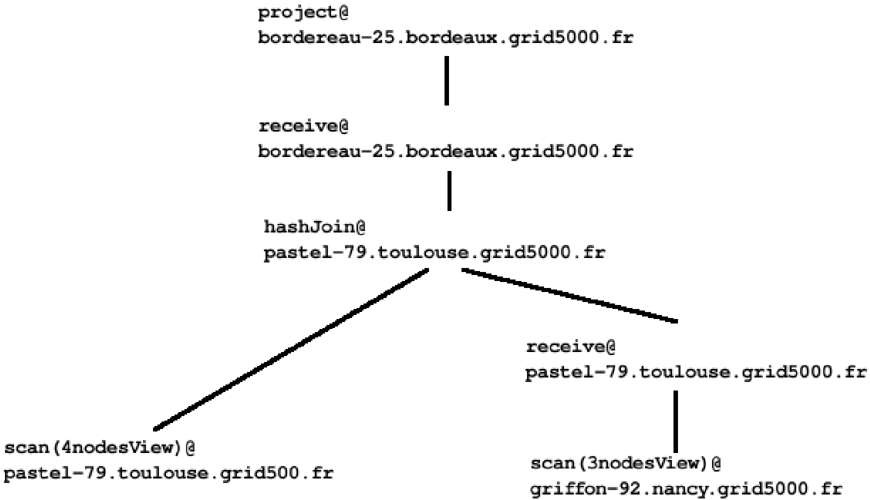
View-based rewriting

1. Rewrite $//a//b_{cont}$ using $//a_{cont}$
2. Rewrite $//a[//c/d[e = 5]/f]$ using $//a_{cont}$
3. Rewrite $//a[//b]//c_{cont}$ using $//a_{id}[//b]$ and $//a_{id}//c_{cont}$
4. Rewrite $//a[//b]//c_{cont}$ using $//a_{id}[//b]$ and $//c_{id,cont}$

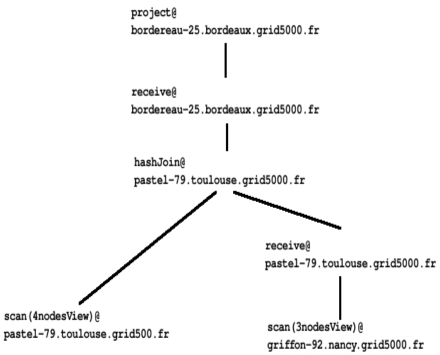
ViP2P platform

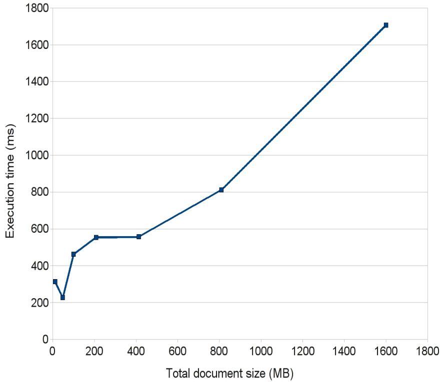
- Fully implemented using Java 6
- Used Berkeley DB (version 3.3.75) to store view data
- Used FreePastry (version 2.1) as our DHT network
- Experiments carried on *Grid5000* using **250 machines**
- **1000 ViP2P peers** were deployed [MZ09]

Query execution: sample plan



Query execution





Related works

Distributed data management [ÖV99, Kos00]

XPath query rewriting [BOB+04, XO05, CDO08, TYÖ+08]

- XPath: wildcard *, union
- Rewritings: intersection, navigations, joins

DHT-based relational data management [LHSH04, HRVM08, APV07]

DHT-based XML indexing [GWJD03, BC06, SHA05, AMP+08]

DHT-based shared XML caches [LP08]

Layered architecture for Web content warehousing [AAC+08]

RDF querying and reasoning on DHT [KMK08, LIK06]

Perspectives

1. Views and queries over annotated documents (K. Karanasos, J. Camacho-Rodriguez)
    - Subscription language for XML with annotations
    - New algorithms for evaluating subscriptions, for rewriting queries
2. Self-adaptive views (A. Katsifodimos)
    - Automatically determining subscriptions/views for the local and global interest
    - Views: selfish, compulsory, collaborative
    - Cost model for adaptation

[AAC+08] Serge Abiteboul, Tristan Allard, Philippe Chatalic, Georges Gardarin, A. Ghitescu, François Goasdoué, Ioana Manolescu, Benjamin Nguyen, M. Ouazara, A. Somani, Nicolas Travers, Gabriel Vasile, and Spyros Zoupanos.
Webcontent: efficient P2P warehousing of web data.
PVLDB, 1(2):1428–1431, 2008.

[AMP05] S. Abiteboul, I. Manolescu, and N. Preda.
Constructing and querying peer-to-peer warehouses of XML resources.
In *ICDE '05: Demo Session*, 2005.

[AMP+08] Serge Abiteboul, Ioana Manolescu, Neoklis Polyzotis, Nicoleta Preda, and Chong Sun.
XML processing in DHT networks.
In *ICDE*, pages 606–615, 2008.

[APV07] Reza Akbarinia, Esther Pacitti, and Patrick Valduriez.
Data currency in replicated DHTs.

In *SIGMOD Conference*, pages 211–222, 2007.

[BC06] Angela Bonifati and Alfredo Cuzzocrea.
Storing and retrieving XPath fragments in structured P2P networks.
Data Knowl. Eng., 59(2), 2006.

[BOB+04] A. Balmin, F. Ozcan, K. Beyer, R. Cochrane, and H. Pirahesh.
A framework for using materialized XPath views in XML query processing.
In *VLDB*, 2004.

[CDO08] Bogdan Cautis, Alin Deutsch, and Nicola Onose.
XPath rewriting using multiple views: Achieving completeness and efficiency.
In *WebDB*, 2008.

[GWJD03] L. Galanis, Y. Wang, S.R. Jeffery, and D.J. DeWitt.
Locating data sources in large distributed systems.

In *VLDB*, 2003.

[HRVM08] Rabab Hayek, Guillaume Raschia, Patrick Valduriez, and Noureddine Mouaddib.
Summary management in P2P systems.
In *EDBT*, pages 16–25, 2008.

[KMK08] Zoi Kaoudi, Iris Miliaraki, and Manolis Koubarakis.
RDFS reasoning and query answering on top of DHTs.
In *International Semantic Web Conference*, pages 499–516, 2008.

[Kos00] Donald Kossmann.
The state of the art in distributed query processing.
ACM Comput. Surv., 32(4):422–469, 2000.

[LHSH04] Boon Thau Loo, Ryan Huebsch, Ion Stoica, and Joseph M. Hellerstein.
The case for a hybrid P2P search infrastructure.
In *IPTPS*, pages 141–150, 2004.

[LIK06] Erietta Liarou, Stratos Idreos, and Manolis Koubarakis.
Evaluating conjunctive triple pattern queries over large structured overlay networks.
In *International Semantic Web Conference*, pages 399–413, 2006.

[LP08] Kostas Lillis and Evaggelia Pitoura.
Cooperative XPath caching.
In *SIGMOD Conference*, pages 327–338, 2008.

[MZ09] Ioana Manolescu and Spyros Zoupanos.
Materialized views for P2P XML warehousing.
Journées de Bases de Données Avancées, 2009.

[ÖV99] M. Tamer Özsu and Patrick Valduriez.
Principles of Distributed Database Systems, Second Edition.
Prentice-Hall, 1999.

[SHA05] Gleb Skobeltsyn, Manfred Hauswirth, and Karl Aberer.

Efficient processing of XPath queries with structured overlay networks.
In *OTM Conferences (2)*, 2005.

[TYÖ+08] Nan Tang, Jeffrey Xu Yu, M. Tamer Özsu, Byron Choi, and Kam-Fai Wong.
Multiple materialized view selection for XPath query rewriting.
In *ICDE*, pages 873–882, 2008.

[XO05] W. Xu and M. Ozsoyoglu.
Rewriting XPath queries using materialized views.
In *VLDB*, 2005.