Peer Data Management Systems
Concepts and Approaches

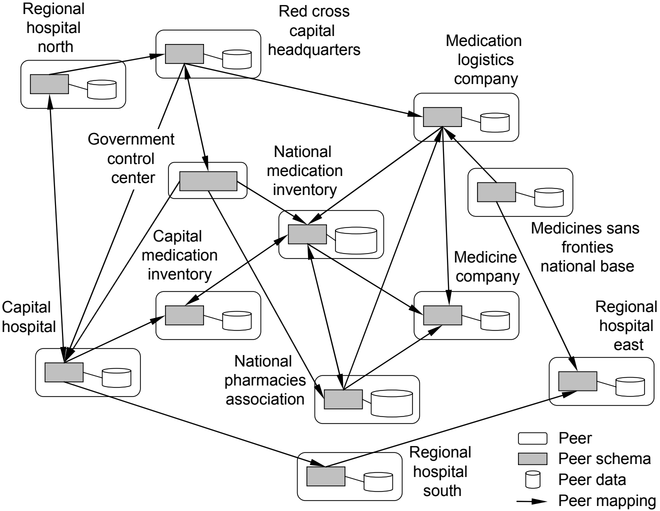
Armin Roth

HPI, Potsdam, Germany

Nov. 10, 2010

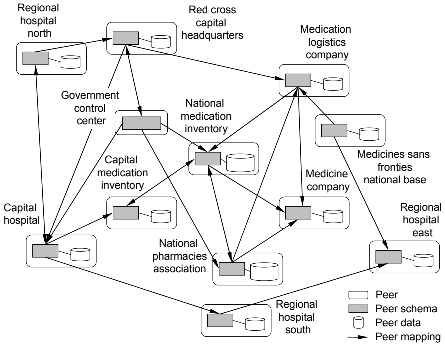
Agenda

Large-scale Information Sharing

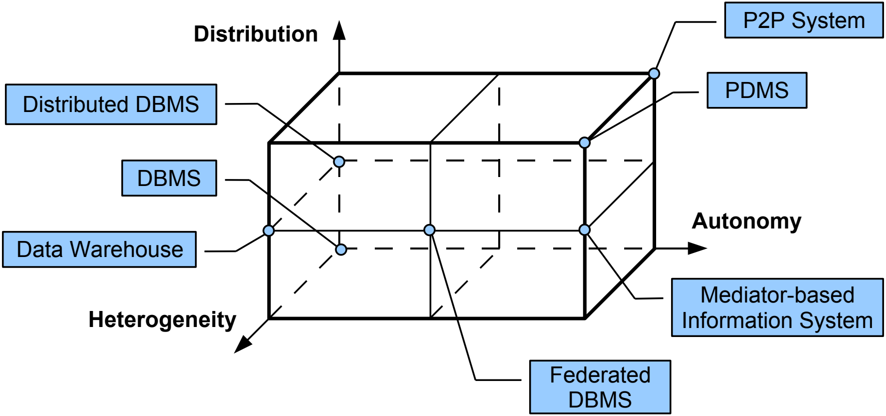


PDMS

- Heterogeneity
- Peer Autonomy
- Mediator: Queries passed to neighbors
- Flexibility
- High Redundancy
- Information Loss



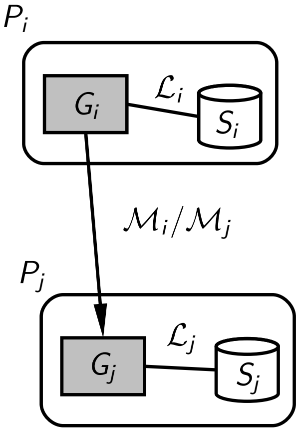
Distributed Information Systems



[OV99]

General System Model

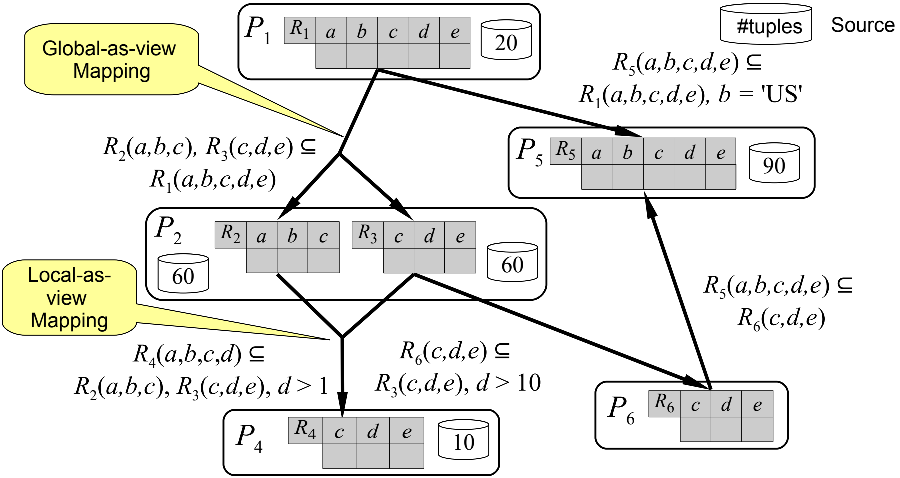
- PDMS set $\mathcal{P}$ of peers P_i with $P_i = \{G_i, S_i, \mathcal{L}_i, \mathcal{M}_i\}$:
  - Peer schema G_i
  - Local schema S_i
  - Local mappings $\mathcal{L}_i$
  - Peer mappings $\mathcal{M}_i$
- Peer mappings $m \in \mathcal{M}_i \cup \mathcal{M}_j$ are assertions $\phi_{G_i} \rightsquigarrow \phi_{G_j}$ resp. $\phi_{G_j} \rightsquigarrow \phi_{G_i}$ with queries ϕ_{G_i} and ϕ_{G_j} of *different* arity



Peer Mappings

- Different peers P_i, P_j heterogeneous in
  - Data model
  - Schema
  - Query language
  - Data schema interplay [BCHL05]
  - Intens./extens. completeness
- Language of mapping assertions $\phi_{G_i} \rightsquigarrow \phi_{G_j}$ must bridge all these types of heterogeneity [MBDH02]

Example



Global-as-view Mapping
Local-as-view Mapping
#tuples
Source
P1
R1 a b c d e
20
P5
R5 a b c d e
90
P2
R2 a b c
R3 c d e
60
60
P4
R4 c d e
10
P6
R6 c d e
$R_5(a,b,c,d,e) \subseteq R_1(a,b,c,d,e),\ b = \text{'US'}$
$R_2(a,b,c),\ R_3(c,d,e) \subseteq R_1(a,b,c,d,e)$
$R_5(a,b,c,d,e) \subseteq R_6(c,d,e)$
$R_4(a,b,c,d) \subseteq R_2(a,b,c),\ R_3(c,d,e),\ d > 1$
$R_6(c,d,e) \subseteq R_3(c,d,e),\ d > 10$

Semantics of PDMS Query Answering [CGLR04]

- Special case: all queries in mapping assertions $\in$ CQ
- Semantics of an *individual* peer: FOL theory T_{P_i}
- (Global) source database $\mathcal{D}$
- Set of all models of PDMS $\mathcal{P}$ wrt. $\mathcal{D}$:

$$sem^{\mathcal{D}}(\mathcal{P}) = \{\mathcal{I} \mid \mathcal{I} \text{ is a model of all } T_{P_i} \text{ based on } \mathcal{D} \wedge \mathcal{I} \text{ satisfies all } \mathcal{M}_i\}$$

- Meaning of $\mathcal{I}$ satisfying $\mathcal{M}_i$ varies in different approaches for peer mappings

Applications for PDMS

- Fusion of organisations
- Semantic Web [HIMT03, HHNR05]
- Disaster Management [HIST03]
- Groupware [ANR07]
- In general:
  Large, loosely coupled integrated information systems

System Model [HRZ+08]

Category	Possible Alternatives
Data model	Relational XML (incl. web services) RDF
Topology	Arbitrary Arbitrary without cycles
Mapping language	GLaV Subset of FOL Mapping tables Data schema interplay (e.g., HePToX)

Semantics

- Expressiveness and interpretation of mapping language determines semantics of
  - query answering
  - data exchange
- 2 principal approaches
  1. *Global reasoning*: Mappings are interpreted as material logical implication
  2. *Local reasoning*: Only exchange of certain answers

Autonomy/Modularity

- Important category in distributed systems with many stakeholders
- Types:
  - Design autonomy (modeling, naming)
  - Communication autonomy (decide about cooperations)
  - Execution autonomy (scheduling of requests)
- Influenced by
  - Semantics
  - Functional requirements
    (e.g., update propagation, global catalog)

Piazza [HIST03]

Data model	Relational, XML
Mapping language	GLaV, definitional mappings
Query language	CQ
Peer autonomy	Global catalog
Semantics of query answering	Open-world wrt. certain peer
Query optimization	Containment-based pruning at query planning time

Hyper [CGL+04, CGLR04]

Data model	Relational
Mapping language	GLaV
Query language	CQ
Peer autonomy	Preserved
Semantics of query answering	Based on epistemic logic, exchange of certain answers
Query optimization	none
Other	Inconsistency tolerance

Hyperion [AKK+03, KAM03]

Data model	Relational (others also possible)
Mapping language	Generalization of GLaV
Query language	CQ, value search
Peer autonomy	Preserved
Semantics of query answering	Open-world and closed-world possible
Query optimization	unknown
Other	Update propagation

Hyperion

- Highly dynamic and scalable
- Schema mapping expressions
- Mapping tables:
  - Correspondences between data values
  - Many-to-many mappings
  - Automatically inferring new entries
  - Respect autonomy of the peers
  - Supports value search (point queries)

Hyperion: Semantics of Mapping Tables

- Mapping table: $\mathcal{X} \rightarrow \mathcal{Y}$ with sets of attribute values resp. variables $\mathcal{X}$, $\mathcal{Y}$ (many-to-many)
- Semantics of practical interest: *closed-open-world*, *closed-closed-world*
- Influences combination of mapping tables

	Open-world	**Closed-world**
present $\mathcal{X}$-value	Any $\mathcal{Y}$-value	indicated $\mathcal{Y}$-values
missing $\mathcal{X}$-value	Any $\mathcal{Y}$-value	no $\mathcal{Y}$-value

Hyperion: Example

GDB id	SwissProt id	MIN id
GDB:120231	P21359	162200
GDB:120231	O00662	193520
GDB:120232	P35240	101000

GDB id	SwissProt id
GDB:120231	O00662

GDB id	MIM id
GDB:120233	162030

Logical Relational Model [SGMB03]

- Domain relation: any subset of $dom_i \times dom_j$
- Relational space: set of local databases and a domain relation
- Coordination formula:
  $CF ::= i : \phi \mid CF \rightarrow CF \mid CF \wedge CF \mid CF \vee CF \mid \exists i : x.CF \mid \forall i : x.CF$
  ($i \in$ set of peers)
- Example:
  $\forall(\text{Doc} : fn, ln, pn, gender, pr).$
  $(\text{Doc} : \texttt{Patient}(1234, fn, ln, pn, gender, pr) \rightarrow$
  $\text{Hospital} : \exists(hid, n, a).\texttt{Patient}(hid, 1234, n, gender, a, \texttt{Davis}, pr) \wedge$
  $n = concat(fn, ln)))$
- Query answering: coordination formulas as deductive rules

Logical Relational Model

Data model	Relational
Mapping language	Coordination formulas: Subset of FOL (implication, conjunction, disjunction, universal and existential quantification wrt. different domains)
Query language	Equal to mapping language
Peer autonomy	Preserved (recursive local reasoning)
Semantics of query answering	Local reasoning (satisfyability of coordination formulas)
Query optimization	unknown
Other	Update propagation (using coordination formulas)

Humboldt Peers [Rot07]

Data model	Relational
Mapping language	extensionally sound GaV: $\forall\bar{x}\forall\bar{y}(\phi_S(\bar{x},\bar{y}) \rightarrow \exists\bar{z}\ g(\bar{x},\bar{z}))$ extensionally sound LaV: $\forall\bar{x}\forall\bar{y}(s(\bar{x},\bar{y}) \rightarrow \exists\bar{z}\ \phi_G(\bar{x},\bar{z}))$
Query language	CQ with semi-interval selections
Peer autonomy	Highly preserved
Semantics of query answering	Exchange of certain answers
Query optimization	Completeness-driven pruning, limitation of resource consumption
Other	Cardinality estimation based on query feedback

Active XML [ABM08]

Data model	XML with web service invocations
Mapping language	web services
Query language	XQuery, XPath
Peer autonomy	Limited
Semantics of query answering	Reasoning encapsulated by web services
Query optimization	Several techniques considering embedded web service calls

Conclusion

- PDMS: flexible architecture for large-scale information sharing
- Main system characteristics: mapping and query languages, peer autonomy, semantics
- Semantics depend on interpretation of mappings
- Comparison of existing PDMS approaches

Future Research

- Reduce redundancy in query answering
- Considering data quality in query answering
- Building and optimizing of network of peers and mappings
- Dealing with different/varying data models and query languages
- Approximative query processing and non-standard query operators (e.g., top-k)

References I

[ABM08] S. Abiteboul, O. Benjelloun, and T. Milo.
The Active XML project: an overview.
VLDB J., 17(5):1019–1040, 2008.

[AKK+03] M. Arenas, V. Kantere, A. Kementsietsidis, I. Kiringa, R. J. Miller, and J. Mylopoulos.
The Hyperion project: From data integration to data coordination.
ACM SIGMOD Record, 32(3):53–58, 2003.

[ANR07] Alexander Albrecht, Felix Naumann, and Armin Roth.
Networked PIM using PDMS.
In *Proc. of the Workshop on Networking Meets Databases (NetDB)*, 2007.

[BCHL05] A. Bonifati, Q. Chang, T. Ho, and L.V.S. Lakshmanan.
HepToX: Heterogeneous peer to peer XML databases.
Technical report, U. of British Columbia and Icar CNR, Italy, 2005.

[CGL+04] D. Calvanese, G. De Giacomo, M. Lenzerini, R. Rosati, and G. Vetere.
Hyper: A framework for peer-to-peer data integration on grids.
In *Proc. of the Int. Conference on Semantics of a Networked World: Semantics for Grid Databases (ICSNW 2004)*, 2004.

References II

[CGLR04] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati.
Logical foundations of peer-to-peer data integration.
In *Proc. of the Symposium on Principles of Database Systems (PODS)*, 2004.

[HHNR05] Ralf Heese, Sven Herschel, Felix Naumann, and Armin Roth.
Self-extending peer data management.
In *Proc. of the Conf. Datenbanksysteme in Business, Technologie und Web (BTW)*, Karlsruhe, Germany, 2005.

[HIMT03] Alon Y. Halevy, Zachary Ives, Peter Mork, and Igor Tatarinov.
Piazza: Data management infrastructure for semantic web applications.
In *Proc. of the Int. World Wide Web Conf. (WWW)*, 2003.

[HIST03] Alon Y. Halevy, Zachary Ives, Dan Suciu, and Igor Tatarinov.
Schema mediation in peer data management systems.
In *Proc. of the Int. Conf. on Data Engineering (ICDE)*, 2003.

[HRZ+08] Katja Hose, Armin Roth, Andrï¿½ Zeitz, Kai-Uwe Sattler, and Felix Naumann.
A research agenda for query processing in large-scale peer data management systems.
Information Systems, 33(7-8):597–610, 2008.

References III

[KAM03] A. Kementsietsidis, M. Arenas, and R. J. Miller.
Mapping data in peer-to-peer systems: Semantics and algorithmic issues.
In *SIGMOD 2003*, pages 325–336, 2003.

[MBDH02] J. Madhavan, P. A. Bernstein, P. Domingos, and A. Y. Halevy.
Representing and reasoning about mappings between domain models.
In *Proc. of the National Conf. on Artificial Intelligence (AAAI)*, 2002.

[OV99] M. T. Özsu and P. Valduriez.
Principles of distributed database systems.
Prentice Hall, 2nd edition, 1999.

[Rot07] Armin Roth.
Completeness-driven query answering in peer data management systems.
In *Proc. of the VLDB 2007 PhD Workshop*, 2007.

[SGMB03] L. Serafini, F. Giunchiglia, J. Mylopoulos, and P. A. Bernstein.
Local relational model: A logical formalization of database coordination.
In *Proc. of CONTEXT*, 2003.