Recife, July 22nd, 2008

Semantic-based Query Reformulation for PDMS

Damires Yluska de Souza Fernandes

[dysf@cin.ufpe.br]

Ana Carolina Salgado

Advisor

Patricia Tedesco

Co-advisor

[acs,pcart]@cin.ufpe.br

Outline

- Motivation
  - ✓ PDMS and Query Reformulation
  - ✓ The SPEED System
- Contributions and Current Status of the Work
- Concluding Remarks

Outline

- Motivation
  - ✓ PDMS and Query Reformulation
  - ✓ The SPEED System
- Contributions and Current Status of the Work
- Concluding Remarks

Motivation

- Relevant research issues in ***Data Integration*** have been pointed out [Halevy et al. 2006]:
  - Generation of Schema ***mappings***;
  - Adaptive ***query processing***;
  - XML as a common syntactic format for sharing data among data sources;
  - ***Model management***, through an algebra for manipulating schemas and mappings;
  - ***Peer-to-Peer Data Management*** and;
  - The application of ***Artificial Intelligence*** to Data Integration.

Our work addresses the problem of ***semantic-based query reformulation*** in a Peer-to-peer Data Management environment

Peer Data Management Systems - PDMS

PDMS = benefits of P2P networks
+
the richer semantics of databases [Zhao 2006]

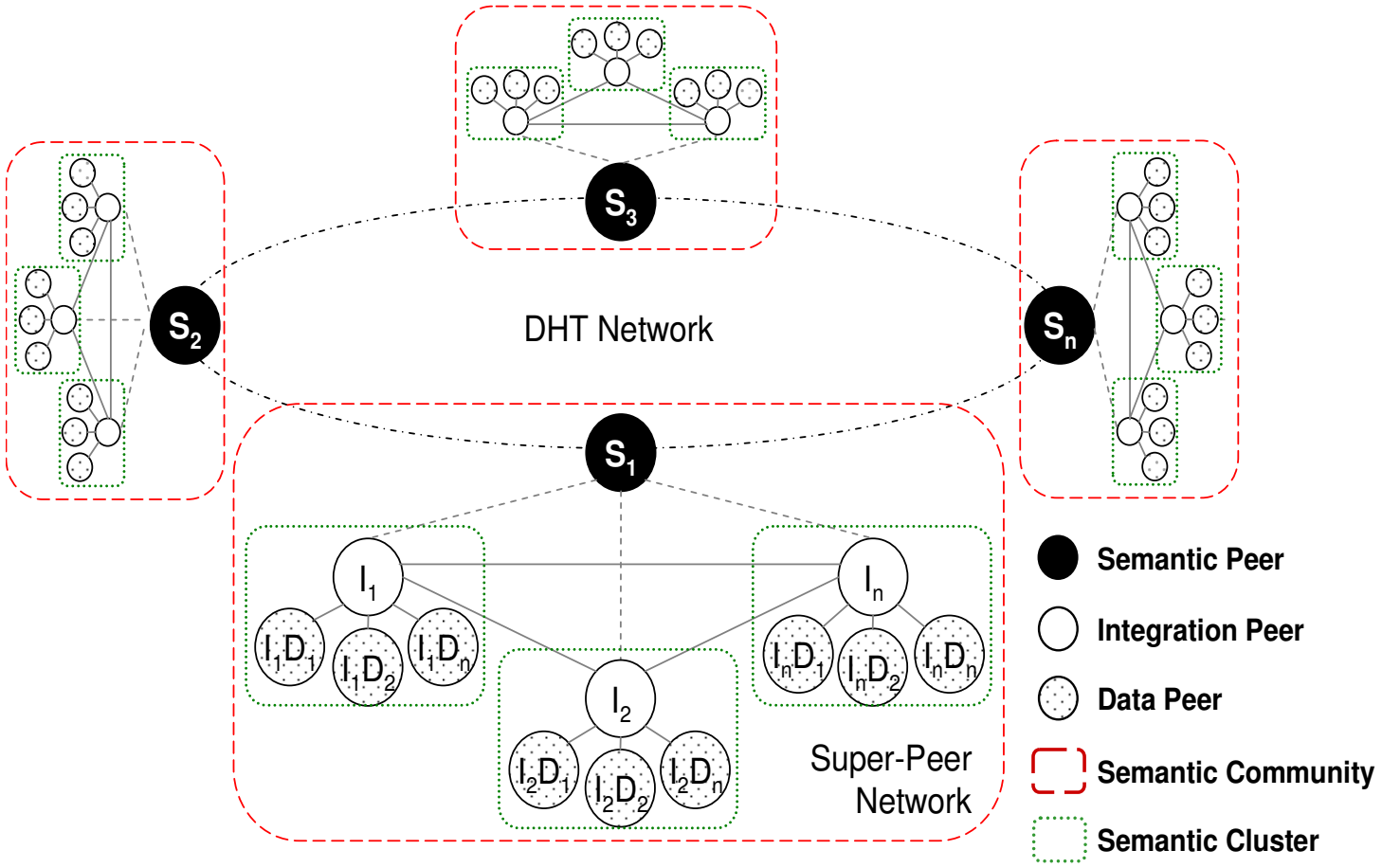
- PDMS can be used for data exchanging, query answering and information sharing
- They consist of a ***set of peers***
- They ***do not*** consider a single global schema
- Queries submitted at a peer are answered with data residing at that peer and with data that is reached along paths of ***mappings*** through the network of peers.

Query Reformulation in PDMS

- The key step in ***query processing*** in a PDMS is reformulating a peer's query over other peers on the available semantic paths.
- ***Query reformulation*** is a process where a query ***Q1*** is translated into a query ***Q2***, across ***mappings***, in such a way that:
  - ***i.*** ***Q2*** contains correct answers;
  - ***ii.*** $Q2 \subseteq Q1$ and;
  - ***iii.*** ***Q2*** provides all the possible answers for ***Q1***.
- It may be divided into two phases:
  - ***Query rewriting***, where the output is a query expression (***Q2***);
  - ***Query answering***, where the result is the set of all possible answers for such query expression

The SPEED System

➢ ***SPEED: Semantic PEEr-to-Peer Data Management System***



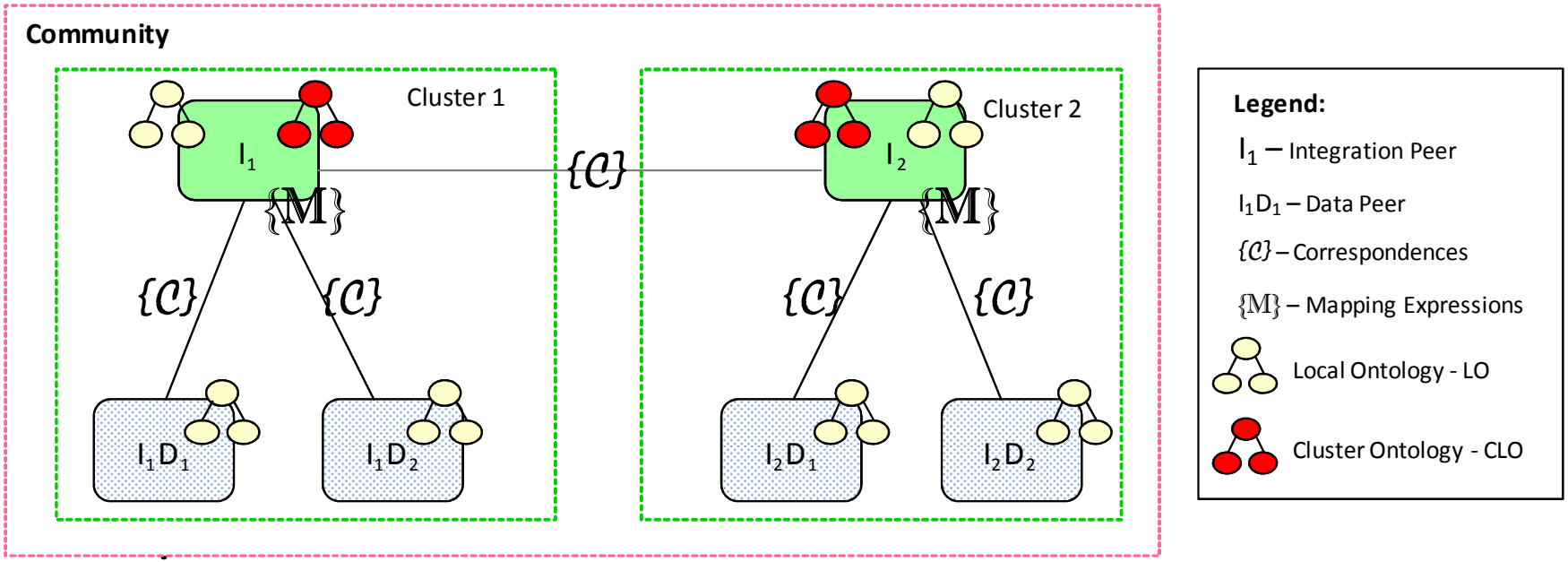
The SPEED System

- Ontologies Categories:
  - ✓ ***Local Ontologies***, resembling the structure of the data sources stored in data peers and integration peers; and
  - ✓ ***Domain Ontologies***, containing concepts and properties of a particular knowledge domain.
    - A **Community Ontology (CMO)** is a domain ontology offered by a semantic peer which is used as a semantic reference by all current clusters within the community.
    - A **Cluster Ontology (CLO)** is a domain ontology stored in an integration peer. It is obtained through the merging of the local ontologies representing data peers' and integration peer's exported schemas.

Outline

- Motivation
  - ✓ PDMS and Query Reformulation
  - ✓ The SPEED System
- Contributions and Current Status of the Work
- Concluding Remarks

Contributions and Current Status of The Work



- Our ***approach***: reformulating a submitted query into another "meaningful" query according to the semantic correspondences between the integration peers' ontologies (CLOs).
  - ✓ We focus on using the semantics behind the ***correspondences*** among semantic neighbors in order to execute more ***enriched*** queries over them.

Semantic Correspondences Definition

- **Definition 1 (Correspondence)**. A ***correspondence*** from ontology Oi to ontology Oj is an expression defined as follows:
  1. i:x ⊑→ j:y, an *isSubConceptOf* correspondence (specialization)
  2. i:x ⊒→ j:y, an *isSuperConceptOf* correspondence (generalization)
  3. i:x ≡→ j:y, an *isEquivalentTo* correspondence (equivalence)
  4. i:x ▷→ j:y, an *isPartOf* correspondence
  5. i:x ◁→ j:y, an *isWholeOf* correspondence
  6. i:x ≈→ j:y, an *isCloseTo* correspondence (closeness)
  7. i:x ≢→ j:y, an *isDisjointTo* correspondence (disjointness)
  - ✓ where x and y are either two concepts or two slots (properties) of Oi and Oj respectively representing two ontologies in the system (a CLO and a LO or two CLOs).

Contributions and Current Status of The Work

- We have also defined a set of ***query rewriting rules*** which are used to produce results closer to the user's expectations.
  - ✓ These rules are being formalized in Description Logics (DL)
- The query rewriting process takes into account:
  - ✓ The ***context*** of the user, of the query and of the environment
  - ✓ The ***semantics*** behind the correspondences between neighbors peers.
    - As a result, queries may be enriched and their results may be more relevant
- Two algorithms:
  - ✓ Query Reformulation between two neighbors peers
  - ✓ Query Routing among integration peers

Concluding Remarks

- This work aims to develop a ***semantic-based query reformulation*** process for the ***SPEED system***.
- Until now, we have:
  - ✓ A contextual ontology – ***CODI***, to be used in the overall query execution process, in order to take advantage of employing contextual elements.
  - ✓ The ***Semantic Correspondences Definition***, formalized in DL
  - ✓ A set of ***Query Rewriting Rules***
  - ✓ The ***algorithms*** for Query Rewriting and Routing, which are being formalized
- Ongoing work:
  - ✓ The implementation of the semantic correspondences identification
  - ✓ The implementation of the query reformulation process in SPEED

Recife, July 22nd, 2008

Semantic-based Query Reformulation for PDMS

Damires Yluska de Souza Fernandes

[dysf@cin.ufpe.br]

Ana Carolina Salgado

Advisor

Patricia Tedesco

Co-advisor

[acs,pcart]@cin.ufpe.br