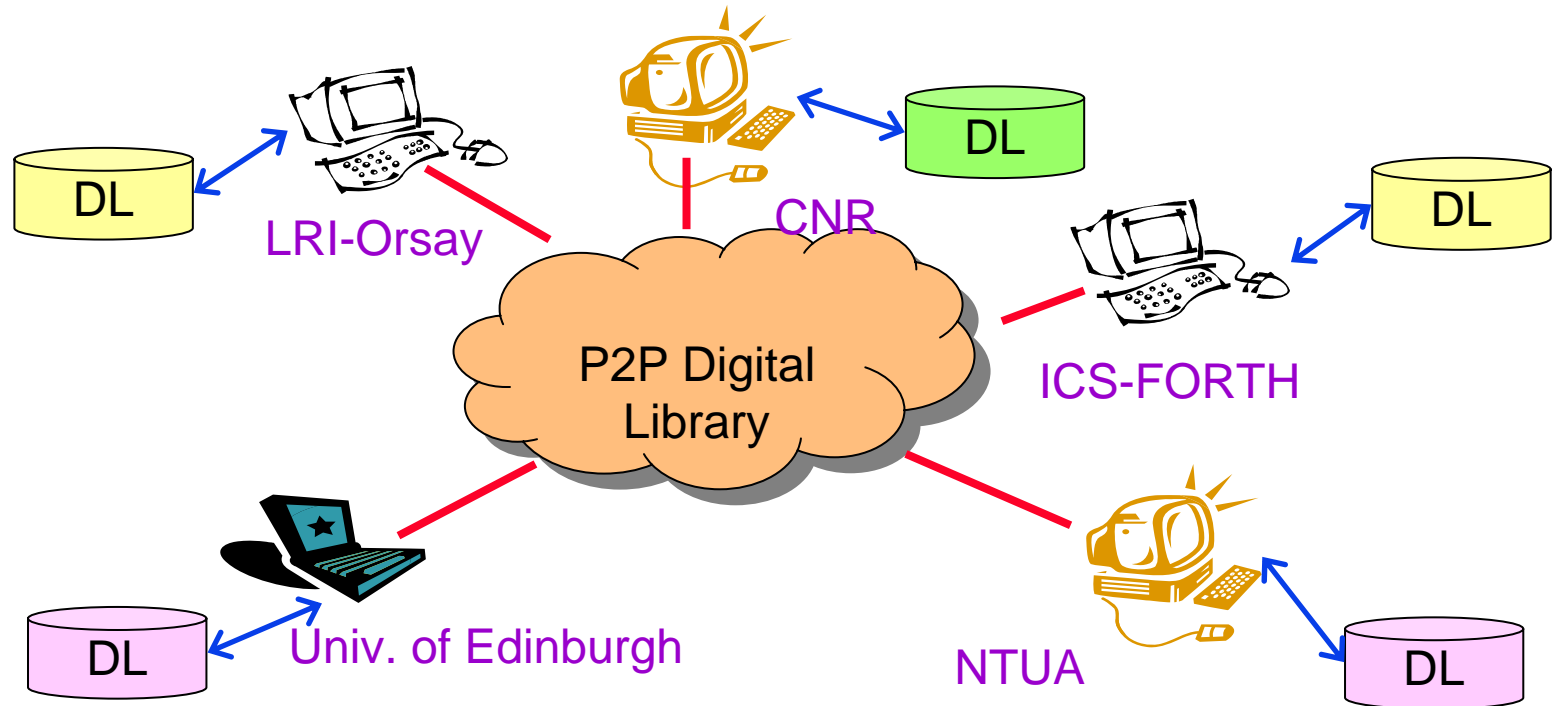# DELOS
All Tasks Meeting

Task 2.7
Application of the P2P Paradigm in Digital Libraries

Task leader: FORTH-ICS

Participants: NTUA-ICCS
UEDIN

The Vision

- DELOS envisions the availability of digital content on a global scale through Digital Libraries (DL) that "can be accessed, integrated and individualized for any user anytime and anywhere in possibly multiple comprehensive and efficient ways"

The Challenges

- P2P DL infrastructure must support advanced data management services on a very large scale (thousands of DL nodes, massive data)

  - Autonomy: no restrictions on how to organize the DLs nodes and what kind of query capabilities to offer
    - Support integrated access despite DL node heterogeneity

  - Decentralized sharing and management of data: DL nodes provide (access) resource descriptions to (from) others
    - Support on the fly collaboration of groups of DL nodes

  - Evolving network: DL node joins and leaves at free will
    - Support P2P dynamicity at a reasonable cost

P2P DLs
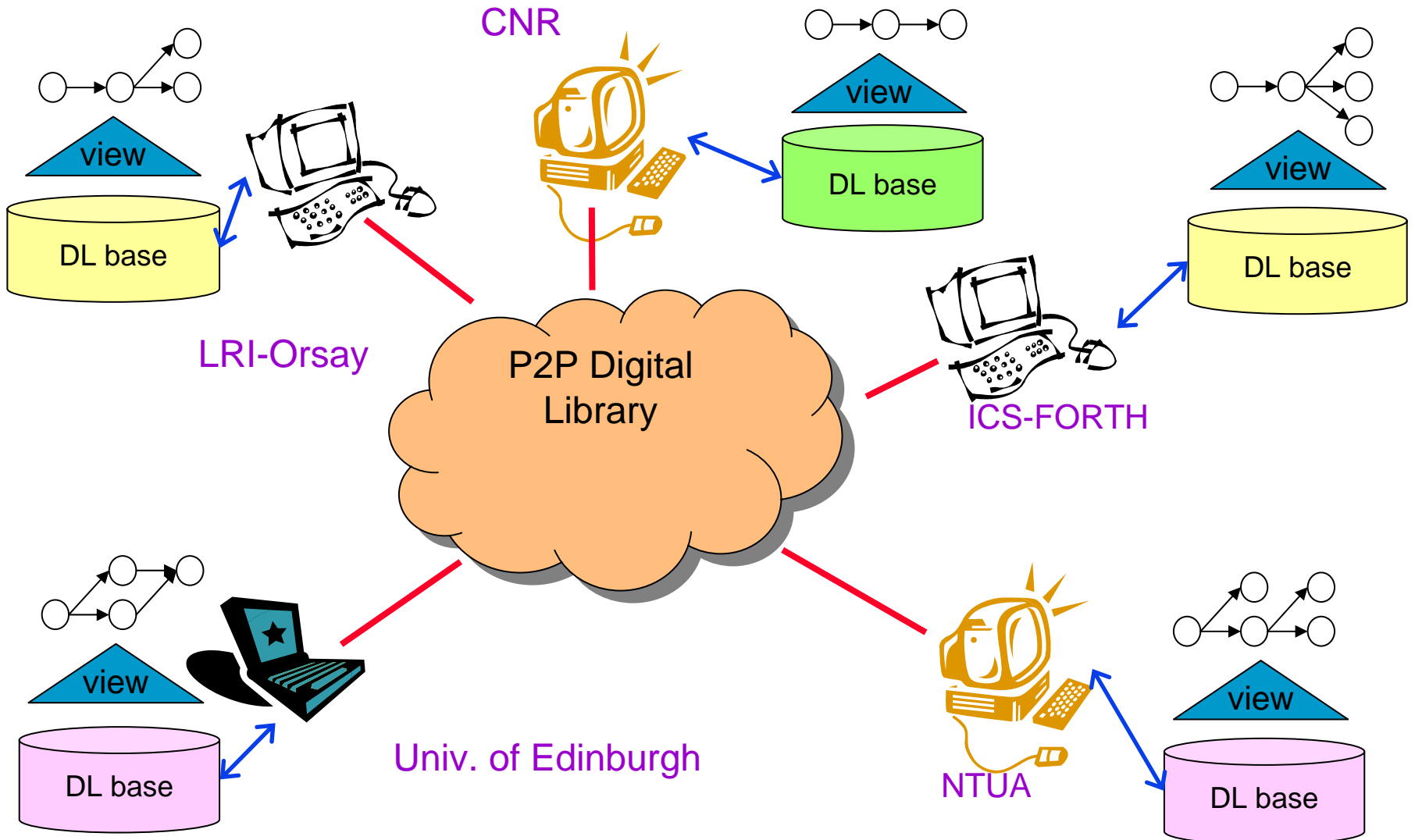
Issues Considered in Task 2.7

To design a fully functional P2P DL the following questions must be addressed:

- ◆ How DL nodes advertise their catalogs and pose queries?

- ◆ How a DL node's neighborhood is formulated?

- ◆ How queries are routed?

- ◆ How queries are processed?

- ◆ How queries are reformulated?

Issues Considered in Task 2.7

- How DL nodes advertise their catalogs and pose queries?
  - ◆ Every DL node provides RDF/S descriptions of resources available on the network
  - ◆ Every description base is advertised using views of globally known RDF/S schemas
  - ◆ Queries are formulated using expressive RDF/S graph patterns

- How a DL node's neighborhood is formulated?
  - ◆ DL nodes sharing the same semantics for a particular domain or application forms a Semantic Overlay Network (SON)
  - ◆ In a SON, a node's neighborhood includes nodes with closely semantically related descriptions through view subsumption
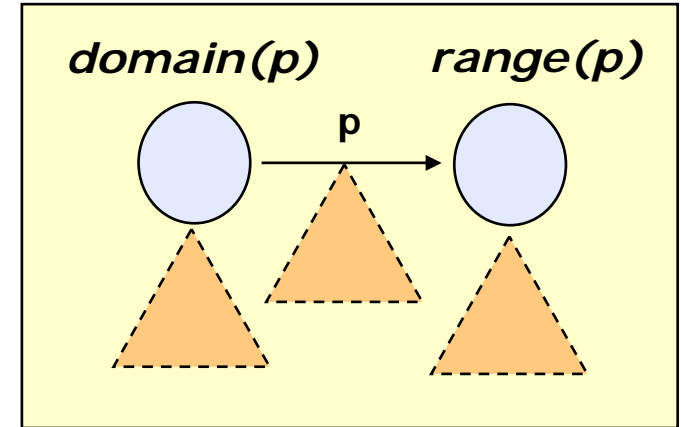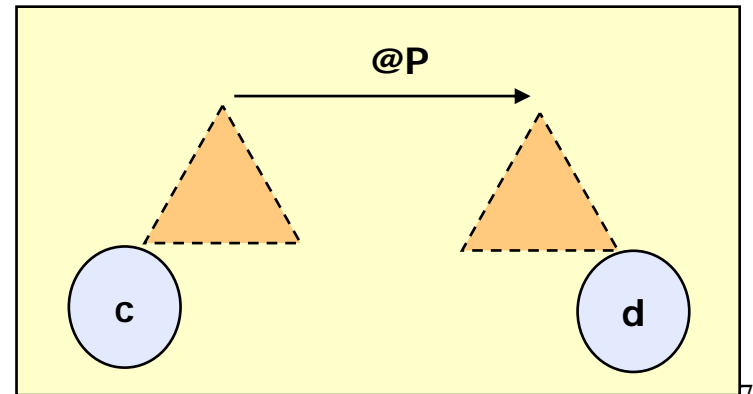
P2P DLs Building Blocks
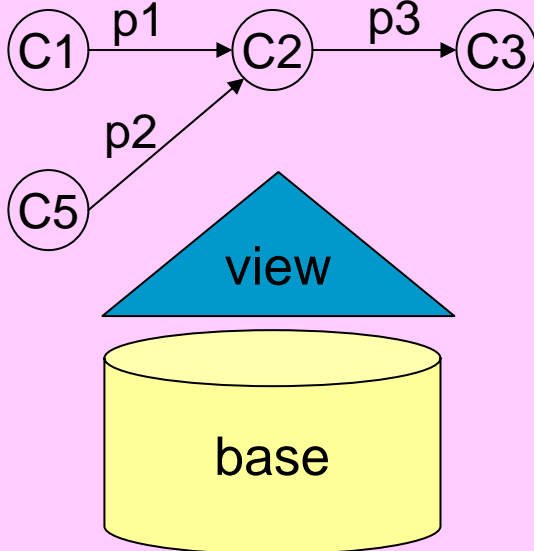
RDF/S Schema Graph
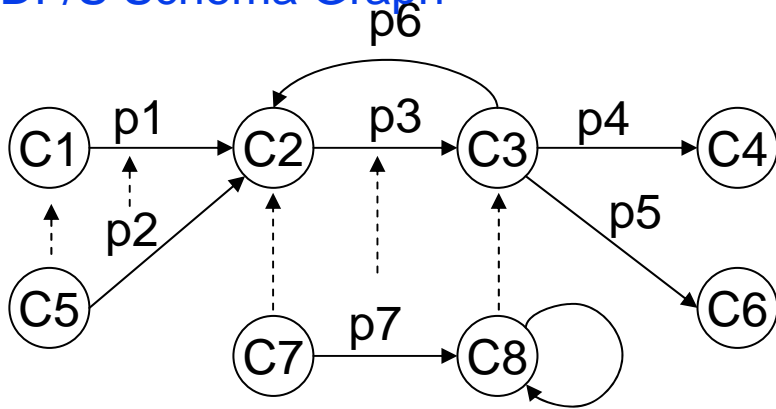
Peer

RDF/S Query Patterns

(X, p, Y)

(X;c, @P, Y;d)

Issues Considered in Task 2.7

- How queries are routed?

  - We have designed and implemented an effective and efficient query routing mechanism for P2P DLs, exploiting indexes on schema views (rather than data) of DL nodes

    - Avoid unlimited data migration

  - Given a query, we rely on those indexes to identify which DL nodes can actually contribute to the query answer

- How queries are processed?

  - Interleaved query routing and plan (runs in several iteration rounds)

    - Start with the original query, at each round smaller fragments of the query are considered, join these fragments to produce valid answers

  - Create and execute multiple query plan that when "unioned" offer completeness in the results

8

Issues Considered in Task 2.7

- Reformulate queries by taking into account a set of schema mapping rules
  - Local heterogeneity: translate RDF/S queries of a SON to underlying XML or relational databases
  - Global heterogeneity: translate RDF/S queries among different SONs

Current Prototypes and Future Work

- Implement, test and evaluate various algorithms for query routing and planning in different P2P DL settings:
  - ◆ Distributed Hash Tables (FORTH) -Simulator
  - ◆ Super-Peer Architecture (NTUA) -System
  - ◆ Ad hoc Architecture (UEDIN) -System

- Employ mapping rules established between the individual peer schemas to route and reformulate queries in a P2P DL network
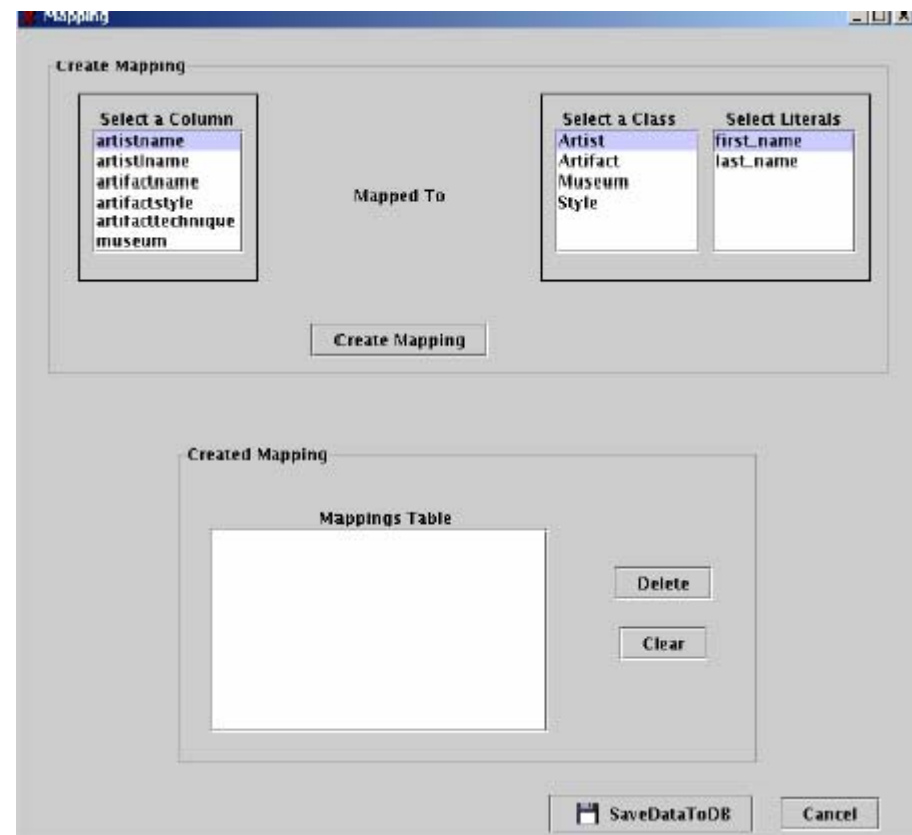
DEMO

- We have implemented a prototype P2P network of DL nodes. Our prototype provides the following functionality:
- **DL network formulation**
    - ◆ Every DL node stores its data in an RDBMS.
    - ◆ In order to join the P2P network, a DL node should either
        - select one of the RDFS schemas provided by the network, or
        - create an RDFS schema by applying certain kind of schema operations (like union, intersection, difference, selection) on the available RDFS schemas.
    - ◆ A tool assists the user to instantiate the RDFS classes by retrieving tuples from the database of the newly joined node
        - i.e., tuples from SQL queries become RDFS class instances.
    - ◆ The RDFS schemas provided are views of a given global schema.
    - ◆ A DL node can leave the network at its own will.
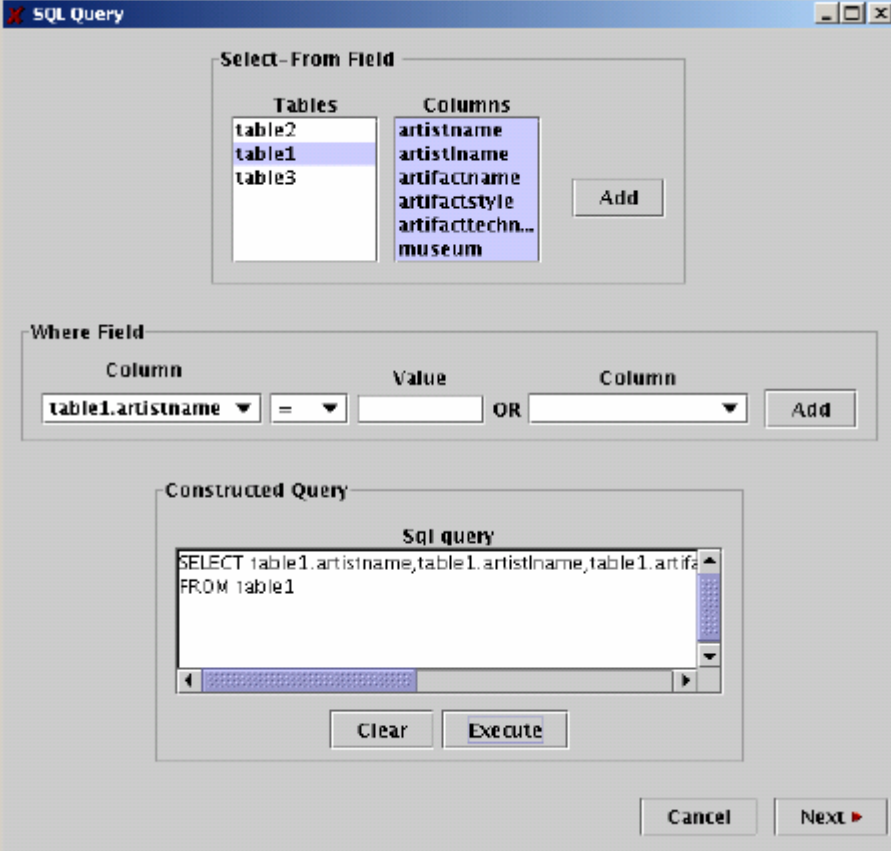
-Instantiate the RDFS classes by retrieving tuples from the database of the newly joined node.
-Tuples from SQL queries become RDFS class instances.

DEMO

- **DL querying**

  - A query initiated by a node is sent to its neighbouring nodes. Each one of those nodes sends the query to its neighbours, and so on.

  - The query is evaluated at each node and results are sent back to the first node.

  - Since DL nodes maintain different local schemas, the query is reformulated before its evaluation on a DL node to match its local schema. The reformulation is performed automatically using information from the global schema.
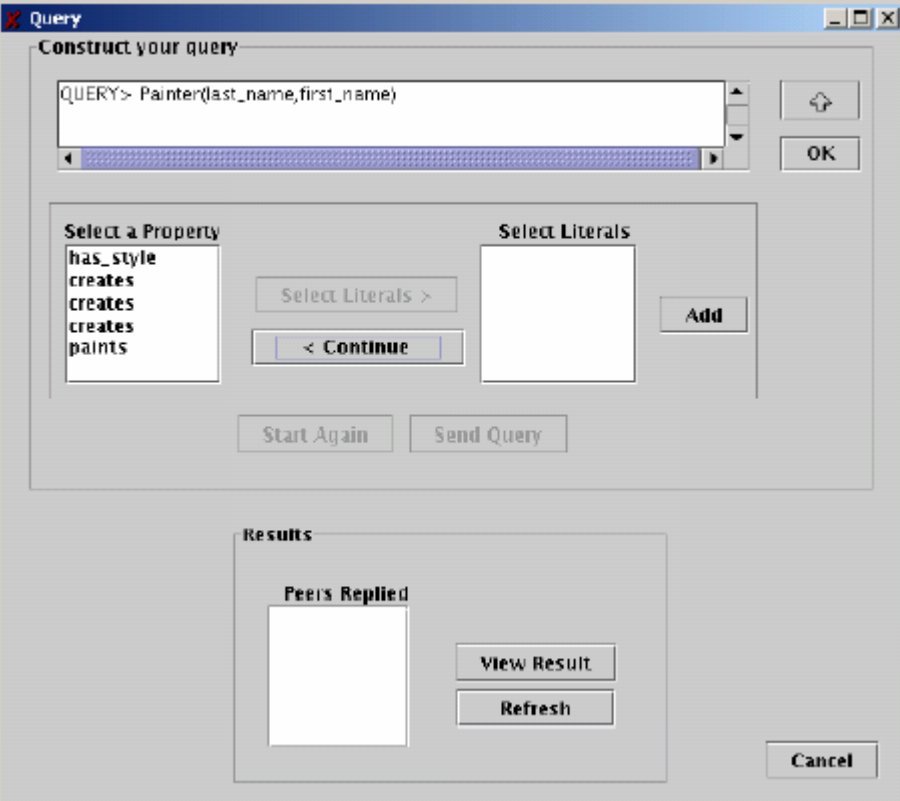
Pose queries in the DL network.
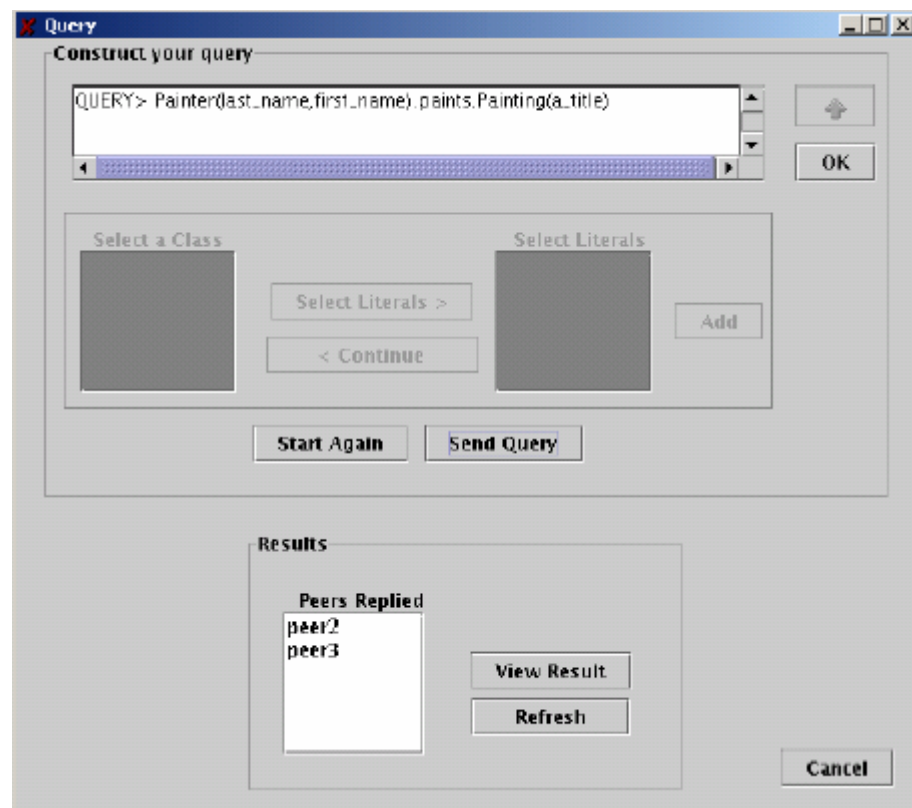
Results from DL node (peer) 2.

Publications/Reports

- [1] P2P Digital Libraries, G. Kokkinidis, L. Sidirourgos, V. Christophides, T. Dalamagas, T. Sellis, and S. Viglas, Poster Session of the 9th European Conference on Research and Advanced Technology for Digital Libraries, Vienna, Austria, September 18-23, 2005.

- [2] Semantic Query Routing and Processing in P2P Database Systems: The ICS-FORTH SQPeer Middleware, G. Kokkinidis and V. Christophides, First International Workshop on Peer-to-Peer Computing and Databases (P2P&DB'04), Co-located with EDBT 2004, Herakleion, Crete, Greece, March 14, 2004.

- [3] Efficient Query Routing in RDF/S schema-based P2P Systems, L. Sidirourgos, G. Kokkinidis and T. Dalamagas, Fourth Hellenic Data Management Symposium (HDMS'05), Athens, Greece, 2005.

- [4] Query Processing in RDF/S-based P2P Database Systems G. Kokkinidis, L. Sidirourgos, V. Christophides, at Semantic Web and Peer-to-Peer, S. Staab, H. Stuckenschmidt (eds.), Springer-Verlag, 2005.

- [5] Query Rewriting Using Mappings in P2P Systems, A. Dimitriou, T. Dalamagas, T. Sellis, S. Viglas, V. Christophides. Technical Report, 2005.

- [6] Exploring Partial Evaluation in P2P XPath Query Processing. P, Buneman, V. Christophides, Wenfei Fan, Anastasios Kementsietsidis, S. Viglas. Technical Report, 2005.