Towards a Context Ontology to Enhance Data Integration Processes

Damires Souza (dysf@cin.ufpe.br)
Rosalie Belian (rbb@cin.ufpe.br)
Ana Carolina Salgado (acs@cin.ufpe.br)
Patricia Tedesco (pcart@cin.ufpe.br)

Outline

- Data Integration X Context
  - Schema Reconciling
  - Query Processing
- CODI-Context Ontology for Data Integration
- Some Considerations
- Conclusion and Further Work

Data Integration Systems

- Need to query across multiple autonomous and heterogeneous data sources
- Dynamic environment
  - It deals with less information and control
  - It is more difficult to plan tasks

We need more semantics and control that may be only acquired on the fly

Context

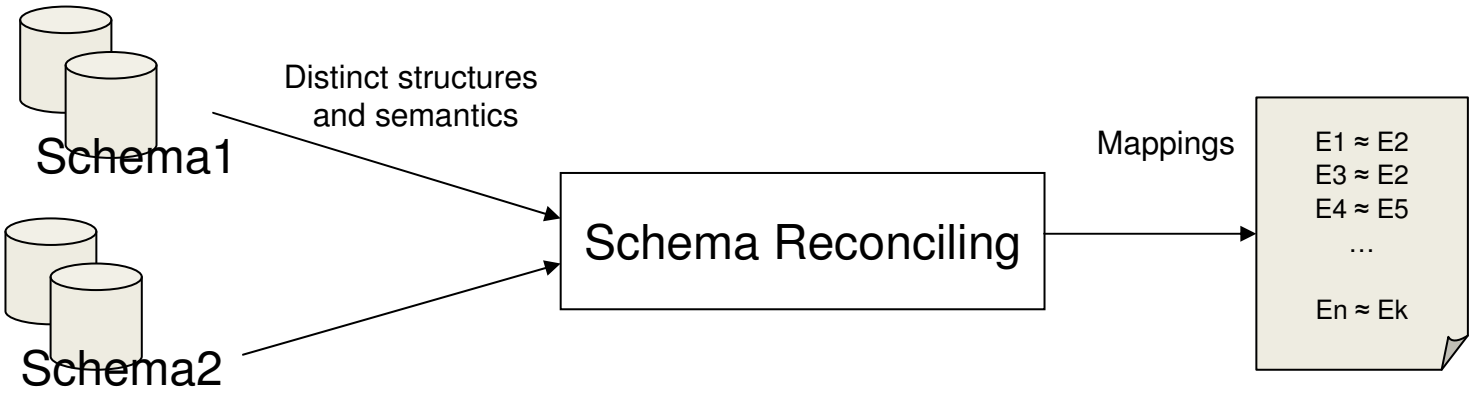
- **Circumstantial elements** that make a certain situation unique and comprehensible (Dey 2001)
- A set of elements surrounding a domain entity which are considered relevant in a specific situation and in a given time
  - **Domain Entity** : person, procedure, an inter-schema mapping

Contextual Elements (CE) refer to pieces of data, information or knowledge that can be used to define the *Context* (Vieira et al. 2007)

Context in DI Processes

- **Schema Reconciling**
  - Identifying in which context the schema elements occur and determining the semantic affinity between them
- **Query Processing**
  - Providing users with more meaningful and complete answers according to the context acquired at query submission and execution time

Schema Reconciling



Schema Reconciling

- This process usually executes:
  - A **preprocessing** routine
    - translates schemas into a common format and makes schema element names processable;
  - A **schema matching** and **mapping** routine
    - produces inter-schema mappings

✓ Element names can have different meanings depending on the semantic context

✓ CEs may provide a more accurate semantic interpretation (allowing restrictions or characterizations of an element name according to a specific semantic context)

Query Processing in PDMS

- **PDMS**: Peer Data Management System
  - It is a natural step beyond data integration systems
  - The single logical schema is replaced by an interlinked **collection of semantic mappings**
  - **Its dynamicity** must be dealt with accordingly
    - Context may be used as a way to deal with such dynamicity.

Context-based Query Processing in PDMS (1/2)

Step	Contextual Elements
Query Submission	User preferences User interface type Submission peer
Query Analysis	Required entities, attributes and operators
Relevant Peers' Establishment	Data model Peer availability Required operators

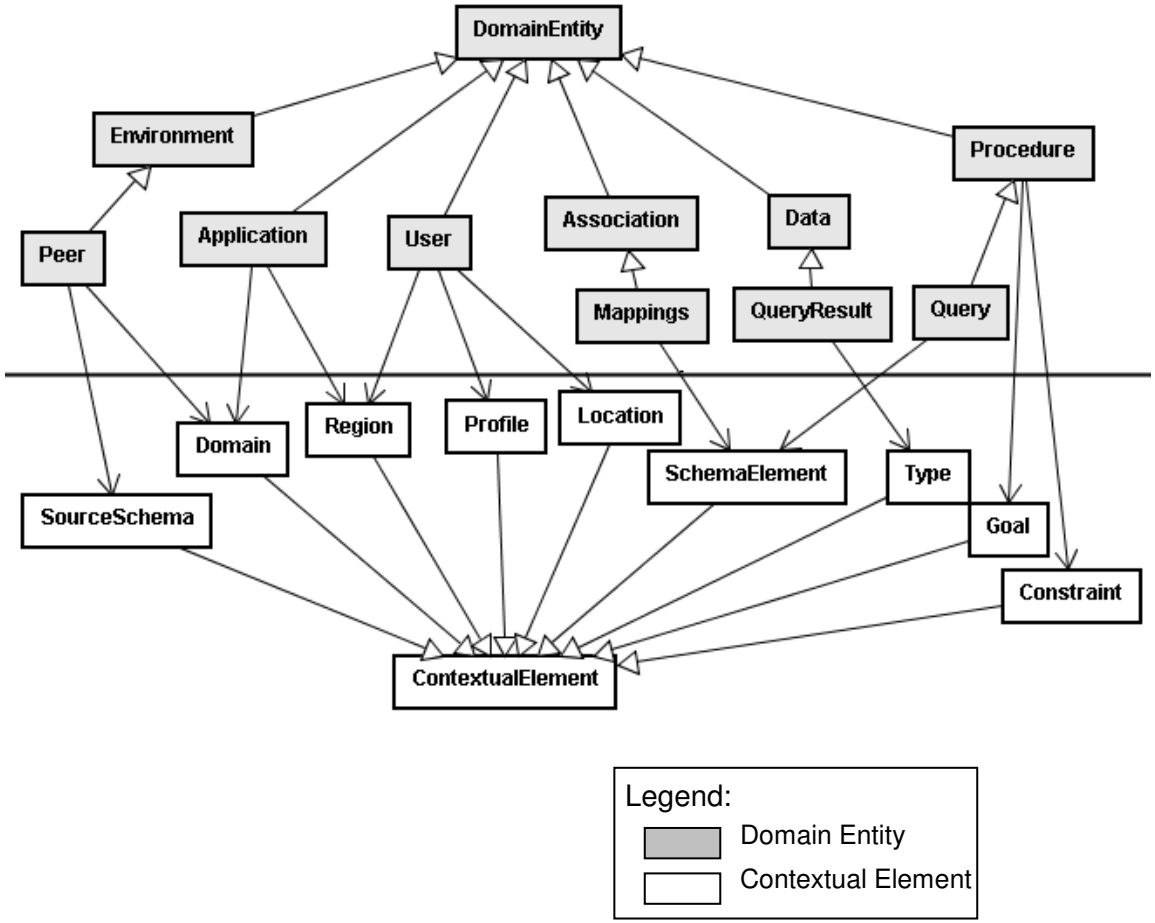
Context-based Query Processing in PDMS (2/2)

Step	Contextual Elements
Query Reformulation	Semantic inter-schema mappings between peers
Query Execution and Answer Integration	Query context
Result Presentation	User's preference Query interface type

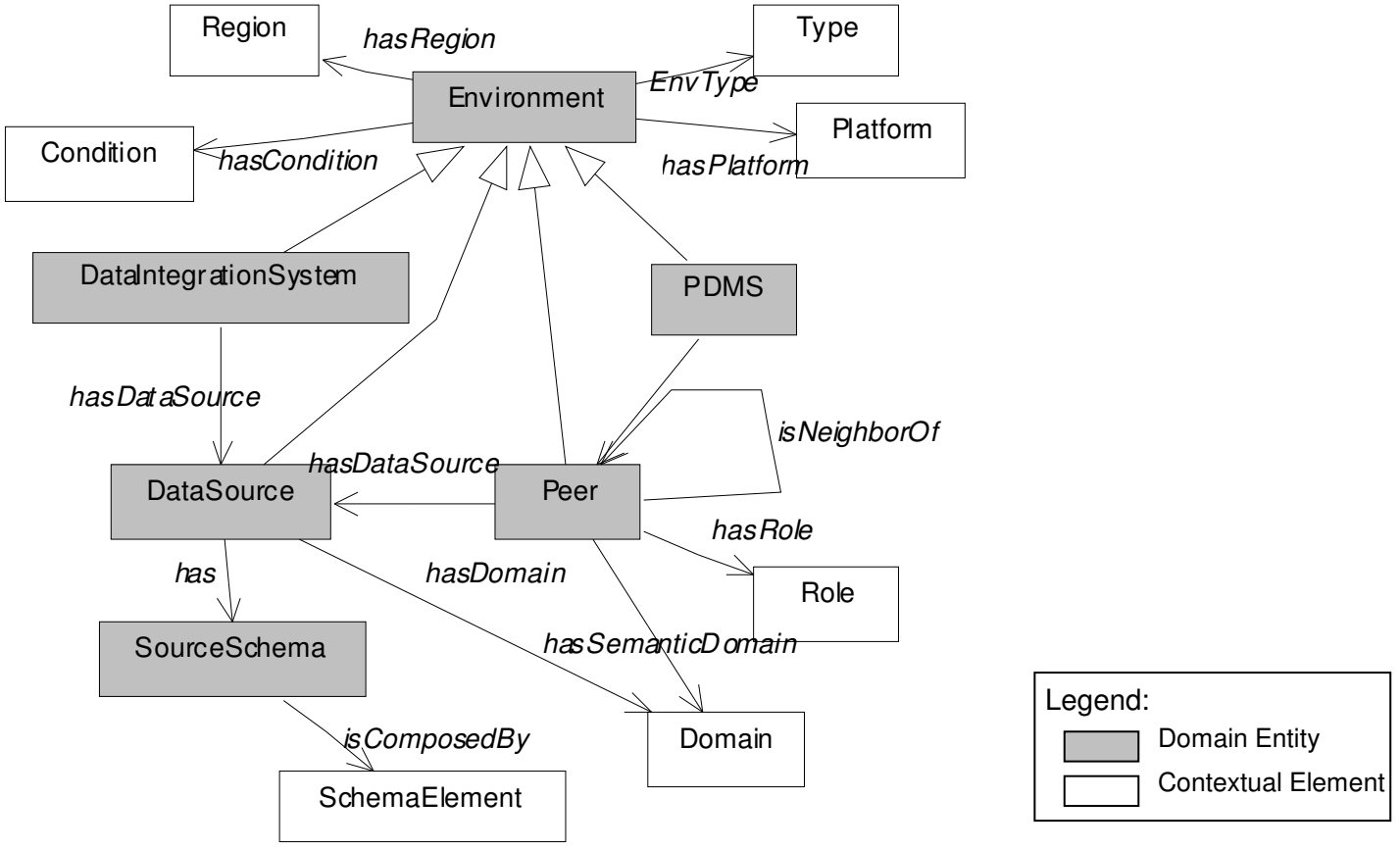
CODI – A Context Ontology for DI

Represent context

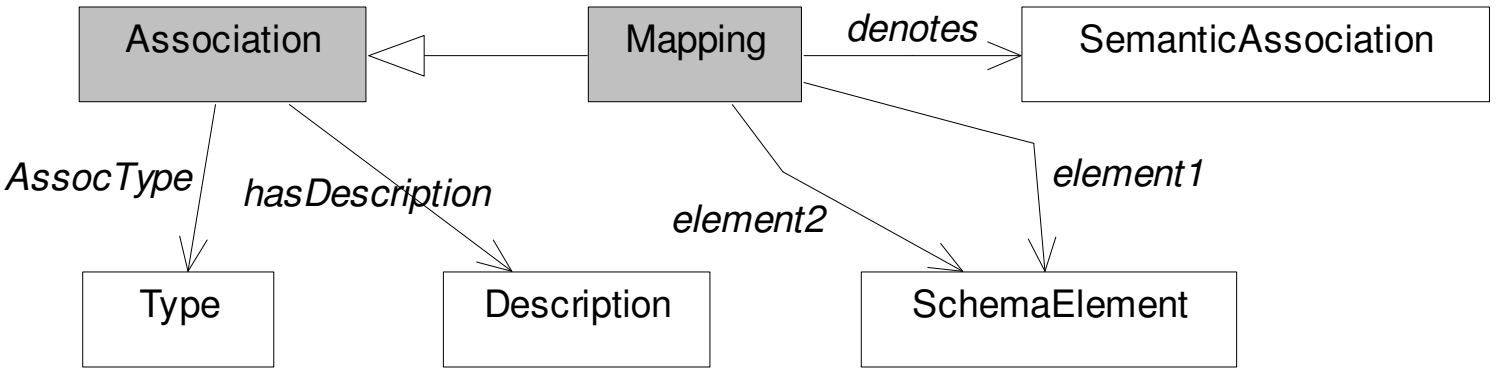
- **Domain Entities**
  - *User, environment, data, procedure, association and application*
- **Contextual Elements**
  - *Domain, schemas, region, profile, type, constraint …*

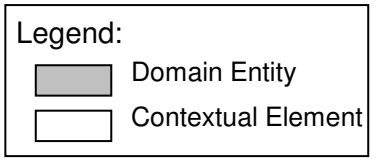


Environment Entity and its CEs

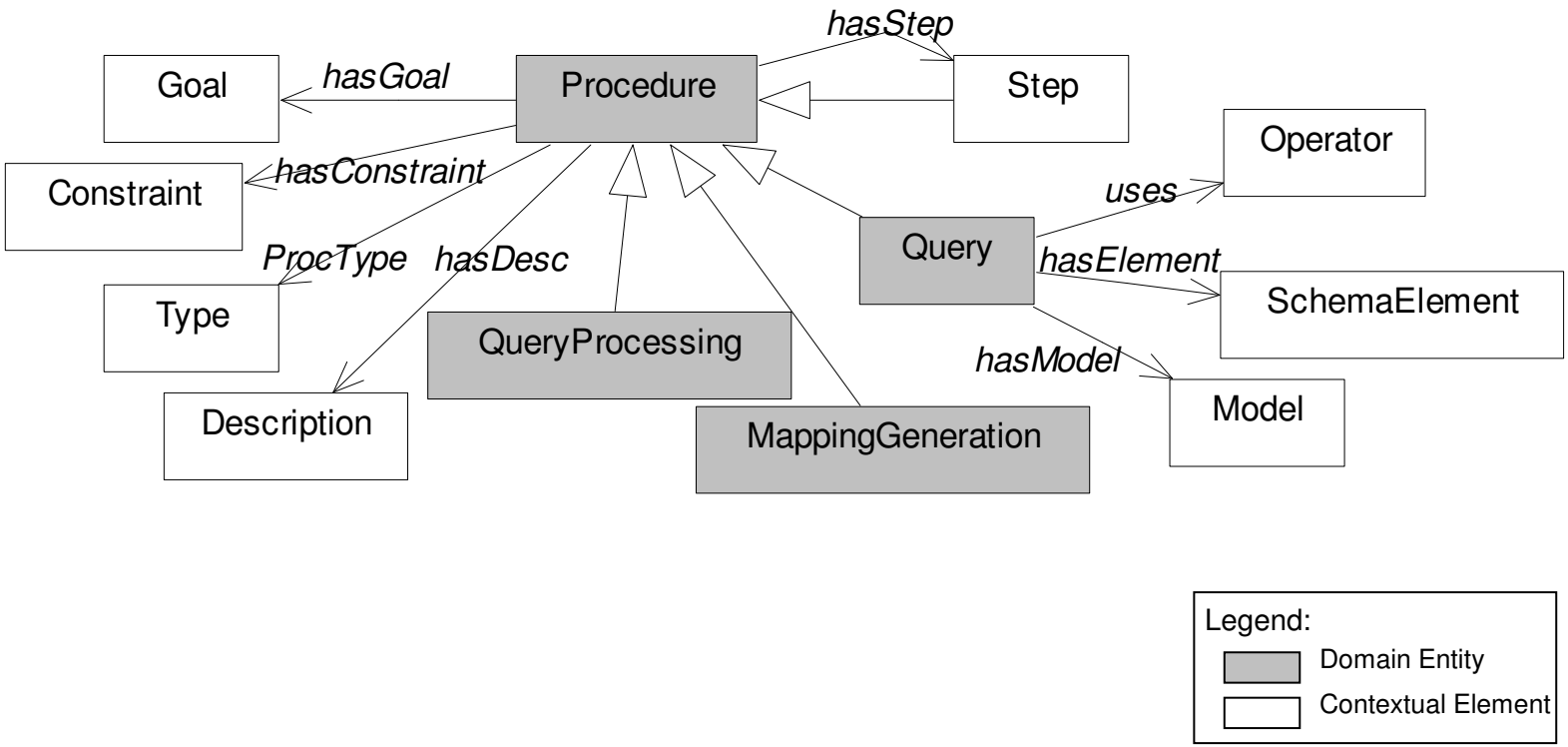


Association Entity and its CEs



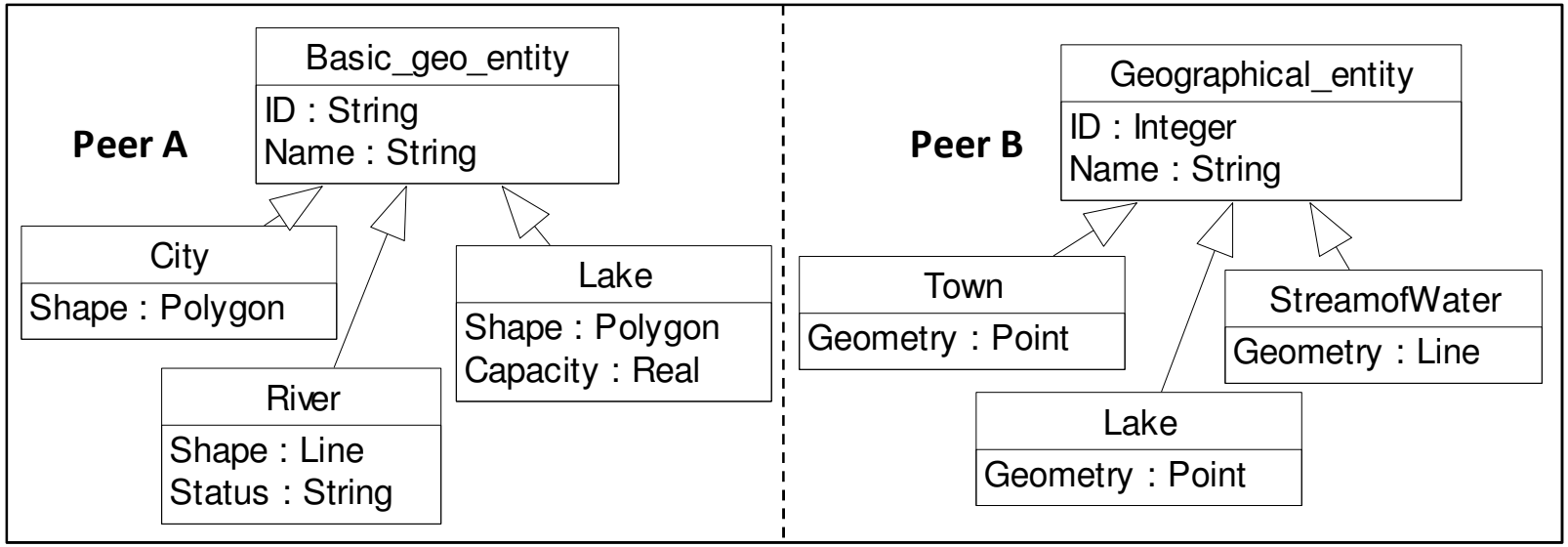


Procedure Entity and its CEs



A Motivating Scenario

- Brazilian Hydrographic System
  - PDMS environment
  - Peers A and B



Using CODI – Query processing

- Spatial query **Q**:

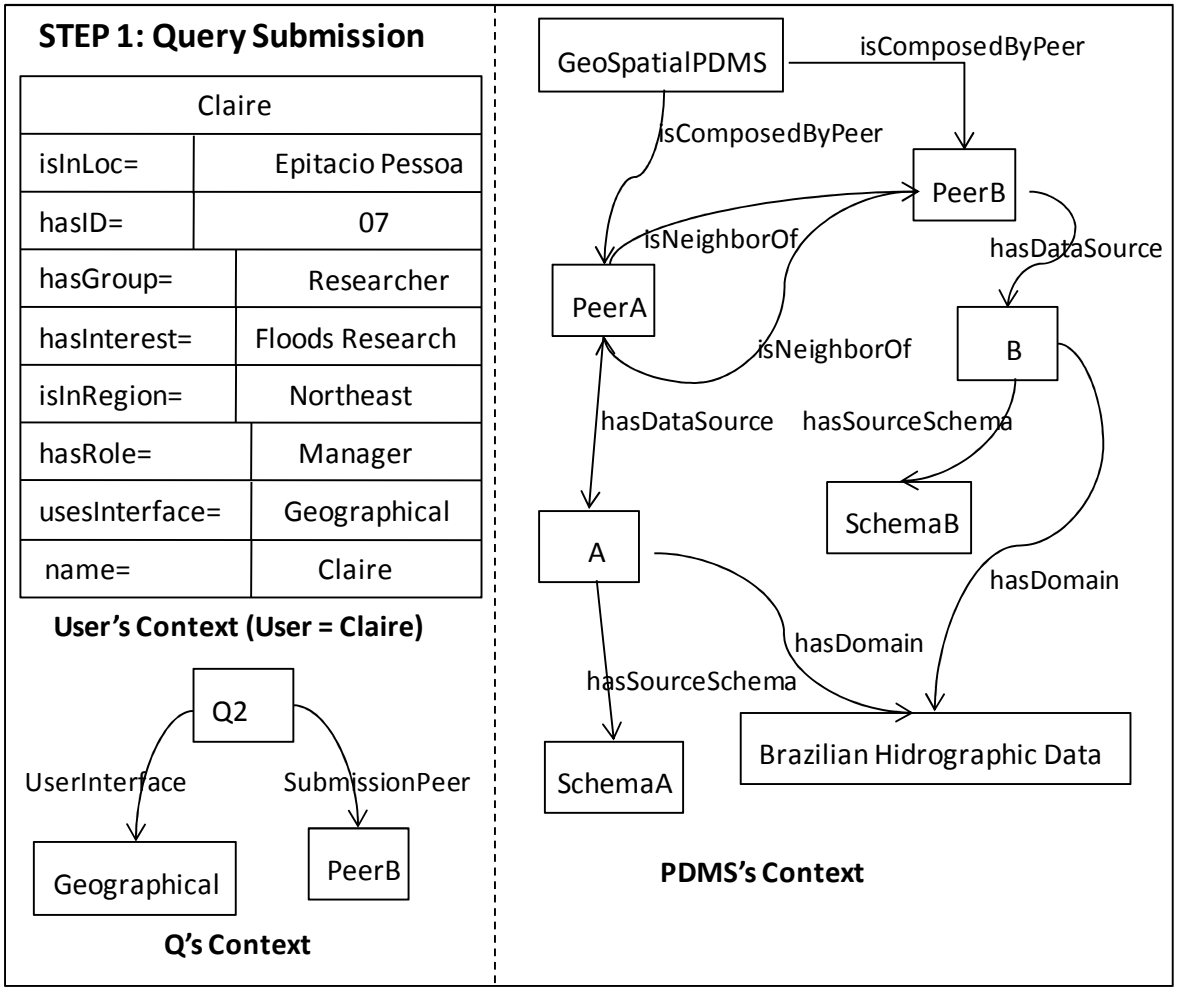
  SELECT R.Name, C.Name FROM River R, City C

  WHERE Cross(R.Shape,C.Shape)=1;"

  "For all the rivers, find the cities through which they pass".

- Q's submission is done in **Peer B**

Query Submission



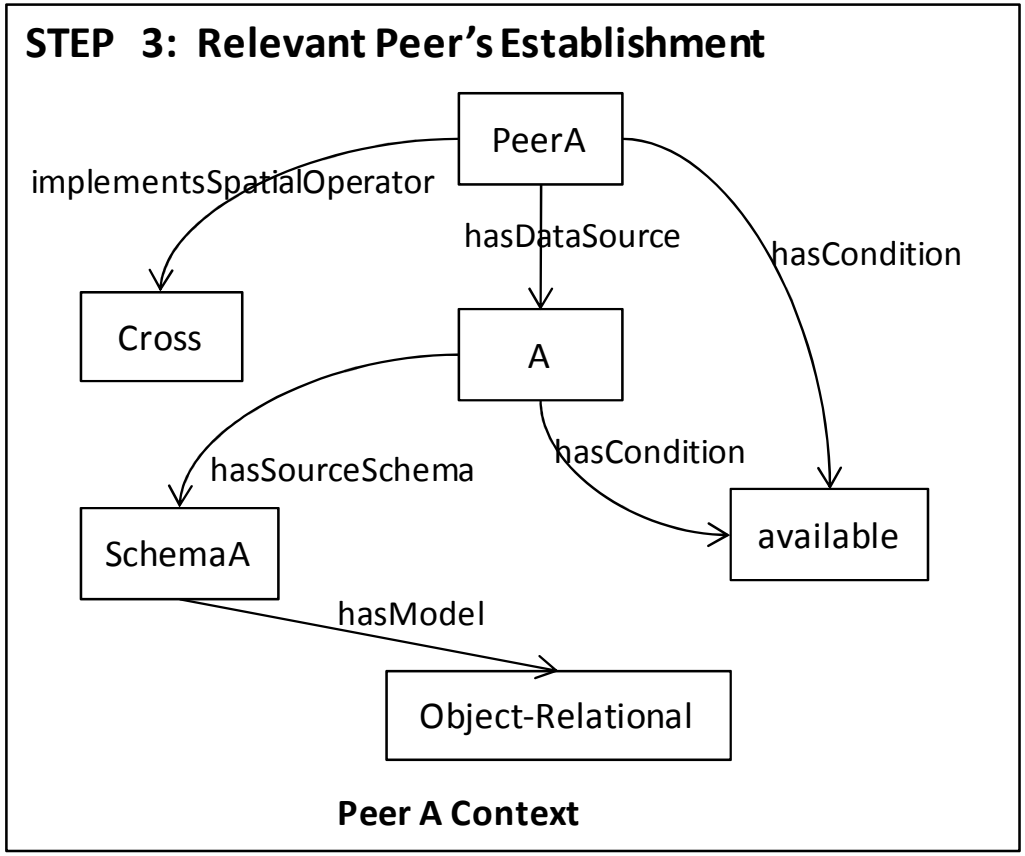
Query Analysis

STEP 2: Query Analysis

Q2	
isExecutedIn=	PeerB
asksForCondition=	Cross(R.Shape, C.Shape) = 1
hasModel=	Object-Relational
asksForAttribute=	B.City.Name
	B.River.Name
hasEntity=	B.River
	B.City
UserInterface=	Geographical
SubmissionPeer=	PeerB
usesOperator=	Cross
hasRestriction=	GeographicalResult
hasFinality=	GetRiversCrossCities
hasDescription=	For all rivers, find the cities...

Q's Context

Relevant Peers' Establishment



Mappings between Schema A and B

03	
element1 =	A.Lake
hasSemanticAssociation =	isEquivalentTo
element2 =	B.Lake
AssocID =	03

04	
element1 =	A.StreamofWater
hasSemanticAssociation =	isEquivalentTo
element2 =	B.River
AssocID =	04

08	
element1 =	A.StreamofWater.Name
hasSemanticAssociation =	isEquivalentTo
element2 =	B.River.Name
AssocID =	08

09	
element1 =	A.Town.Geometry
hasSemanticAssociation =	isSimilarTo
element2 =	B.City.Shape
AssocID =	09

Some A-B Mappings

Query Reformulation

Q2RefA:

```
SELECT S.Name, T.Name
FROM StreamofWater S,
     Town T
WHERE
     Cross(R.Shape,C.Shape)
     =1;”
```

STEP 4: Query Reformulation

Q2REFA	
isExecutedIn =	Peer_A
asksForCondition =	Cross(SW.Geometry,T.Geometry)=1
hasModel =	Object-Relational
asksForAttribute =	A.StreamofWater.Name
	A.Town.Name
hasEntity =	A.StreamofWater
	A.Town
isReformulationOf =	Q2
usesOperator =	Cross
hasRestriction =	Geographical Result
hasFinality =	GetRiversCrossCities
hasDescription =	For all the rivers, find the cities thro…
name =	Q2REFA

Q2REFA is a reformulation of Q

Some Considerations (1/3)

- Representing context information using an **ontology** brings various benefits:
  - Provides **concept subsumption**, concept consistency and instance checking
  - Allows to **organize knowledge**
  - Eases execution of **semantic queries**
  - Allows defining **constraints** and **reasoning rules**

Some Considerations (2/3)

- A PDMS environment is **highly dynamic**, so we have to obtain the context:
  - Around the query (its semantics),
  - Around the peers (availability),
  - About mappings (may be of different types)
  - Concerning the user (preferences, interface)

Some Considerations (3/3)

- **By using context**, the system is able to adapt and react to different user's queries and needs.
- **Without context**, query processing would be limited by not dealing with some information that can just be acquired on the fly.

Conclusions

- Complexity of DI environments **makes context a necessity**
- Differences related to other approaches:
  - **Contextual Elements** are defined according to **Domain Entities** that have been identified as relevant
  - Broader range of concepts used in DI processes
    - Two new entities – **procedure and environment**
  - **Ontology** is used as a context representation model
    - A formal framework
    - With reasoning services

Conclusions and Further Work

- CODI was initially used by a schema reconciling process
  - Preliminary results have shown considerable improvements
- We are working on the implementation in a PDMS environment
- **Further work**
  - Development of additional scenarios
  - Integration of this work with a context manager

Towards a Context Ontology to Enhance Data Integration Processes

Damires Souza (dysf@cin.ufpe.br)
Rosalie Belian (rbb@cin.ufpe.br)
Ana Carolina Salgado (acs@cin.ufpe.br)
Patricia Tedesco (pcart@cin.ufpe.br)