

A Knowledge-Based Approach to Querying Heterogeneous Spatial Databases

Andrea Rodríguez

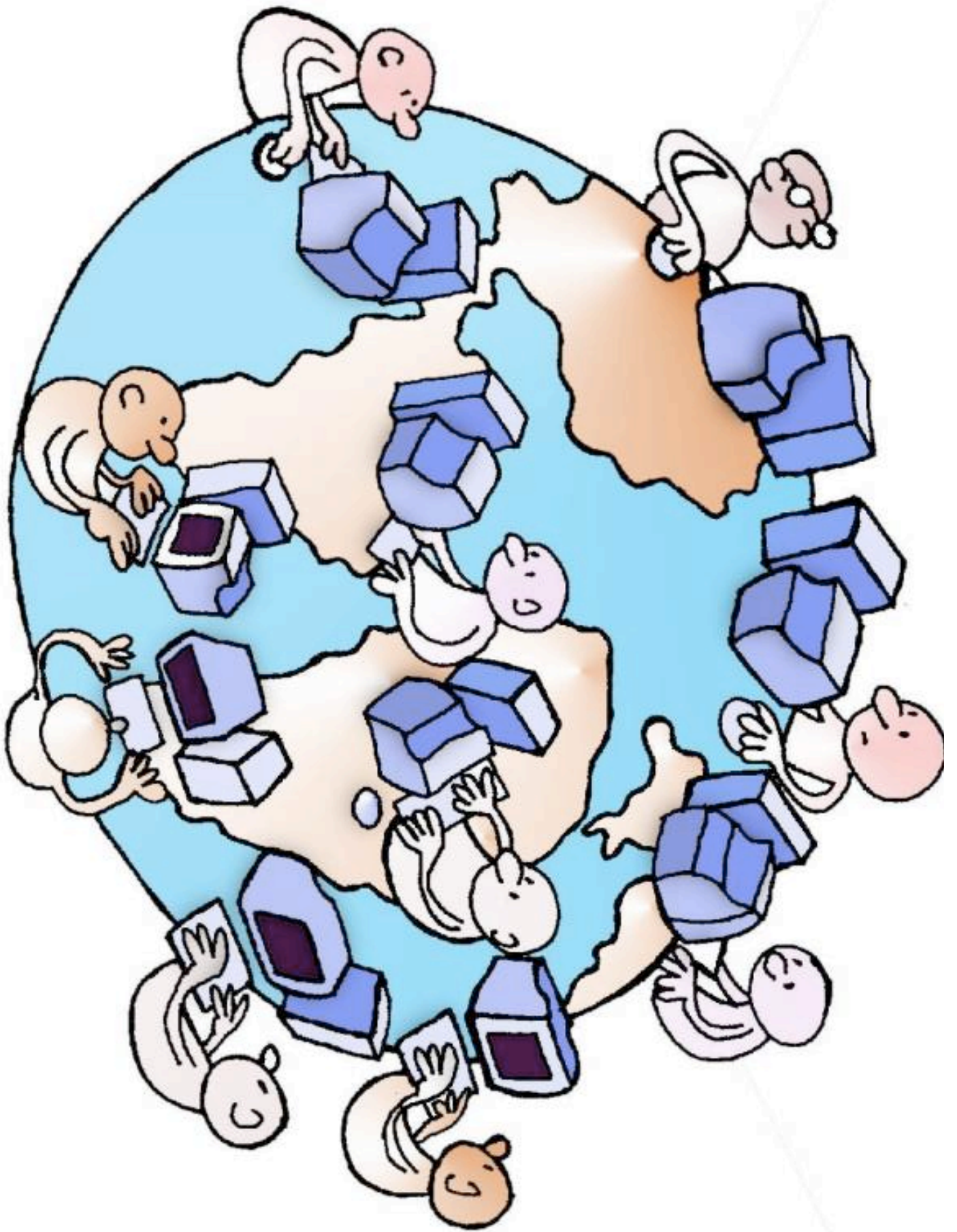
Universidad de Concepción - Chile

Department of Information Engineering and Computer Science

{andrea,mvaras}@udec.cl

Outline

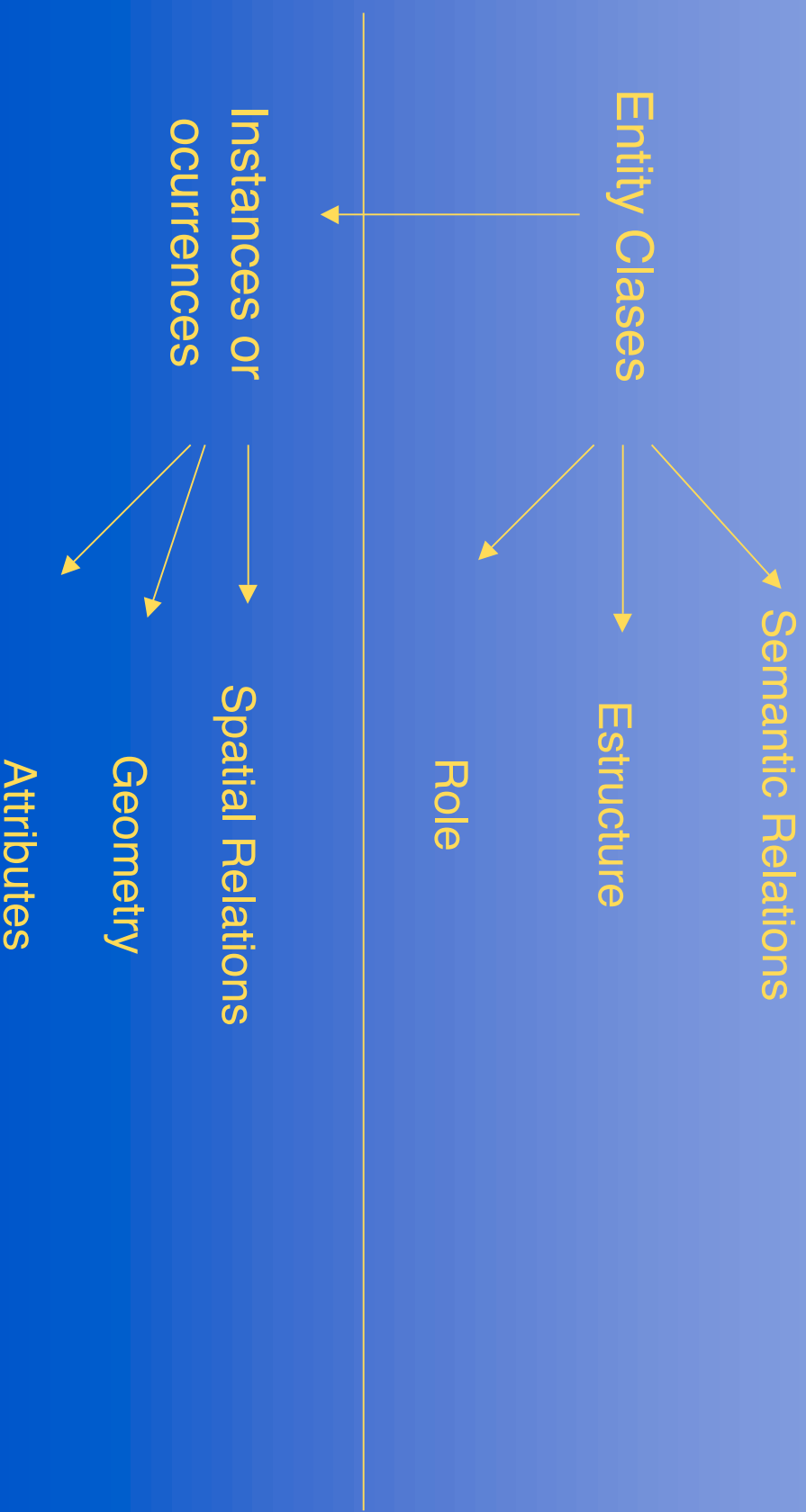
- Problem
- General Approach
- Ontology-Based Similarity
- DB Schema Similarity
- Conclusions and Future Work



Heterogeneous databases

- Syntactic level (data types and formats)
- Schematic level (schematic integration, query languages, and interfaces)
- Semantic level

Heterogeneous databases



Heterogeneous databases

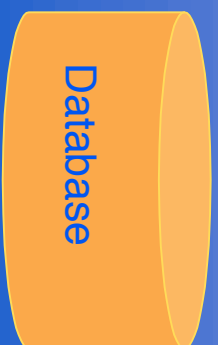
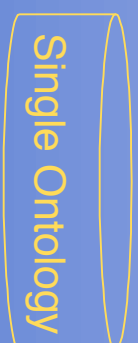
- Syntactic level (data types and formats)
- Schematic level (schematic integration, query languages, and interfaces)

Semantic level

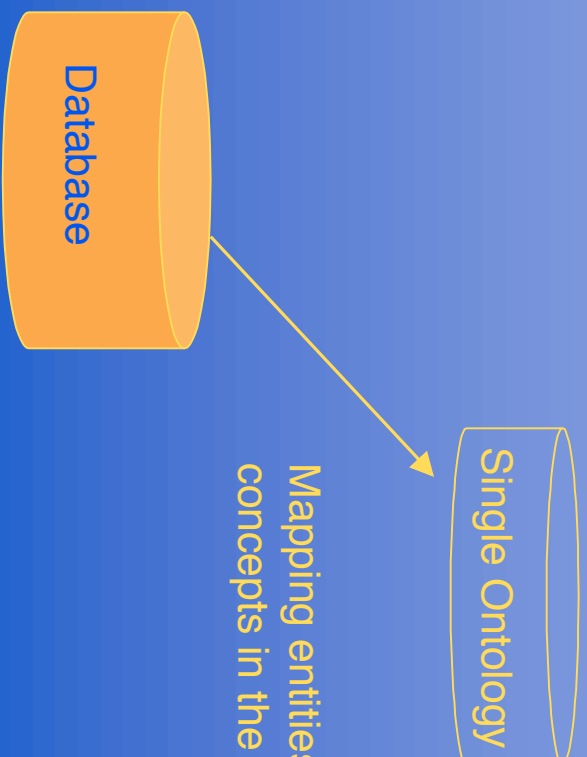
Ontology-Based Approach

- Rich semantic description
- Independence of data representation
- Abstraction of irrelevant information

Ontology-Based Approach

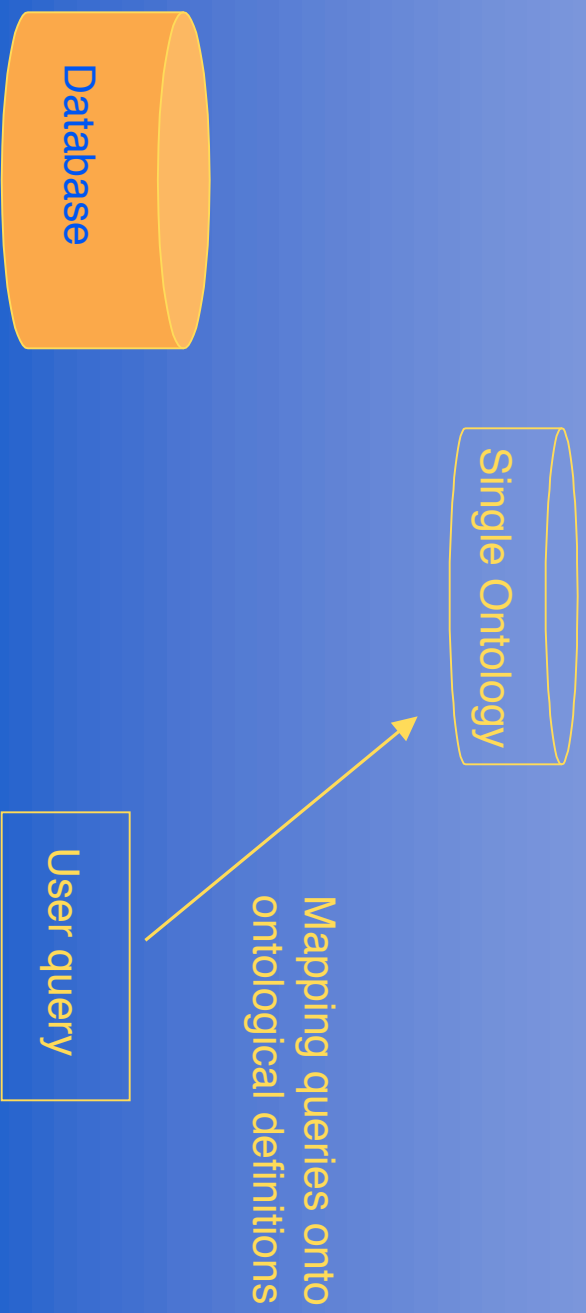


Ontology-Based Approach



Mapping entities in the database onto concepts in the ontology

Ontology-Based Approach



Issues

- Different conceptualizations imply different ontologies
- A single ontology forces to commitments and limits updates
- There exist well-defined ontologies for specific domains



Issues

Issues

Stadium



Athletic field



Issues of Multiple Ontologies

- Differences in the level of explicitness and formalization
- Ontology mismatches (polysemy, synonymy, overlapping)

General Approach

- by using a user ontology we allow users to express queries in their own terms.
- we expand the query to extract not only equivalent but also similar concepts.
- databases have no ontological descriptions of their stored entities so, we cannot compare, at the ontological level, different databases

General Approach

Query = { C_1, \dots, C_n }

- A query is expressed in terms of a user ontology

General Approach

Query = $\{C_1, \dots, C_n\}$



Similarity₁

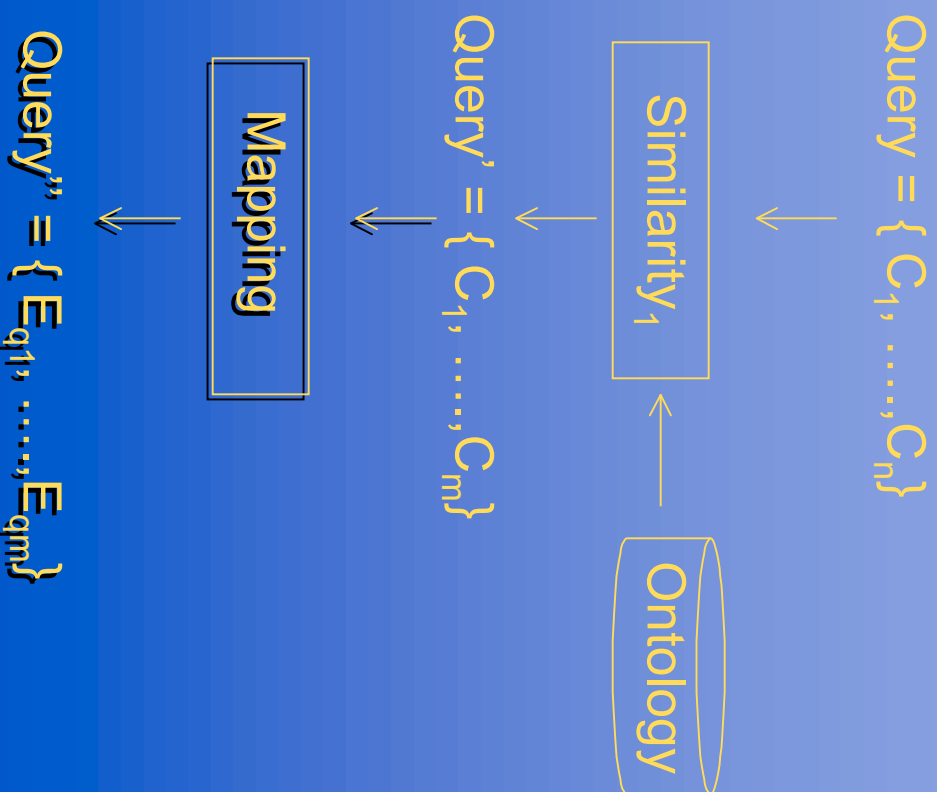


Query' = $\{C_1, \dots, C_m\}$

- A query is expressed in terms of a user ontology

- A similarity measure expands this query with similar concepts of this ontology

General Approach



- A query is expressed in terms of a user ontology
- A similarity measure expands this query with similar concepts of this ontology
- **The expanded query is mapped onto a database schema, query schema**

General Approach

Query” = $\{E_{q1}, \dots, E_{qm}\}$



Similarity₂



Database_a



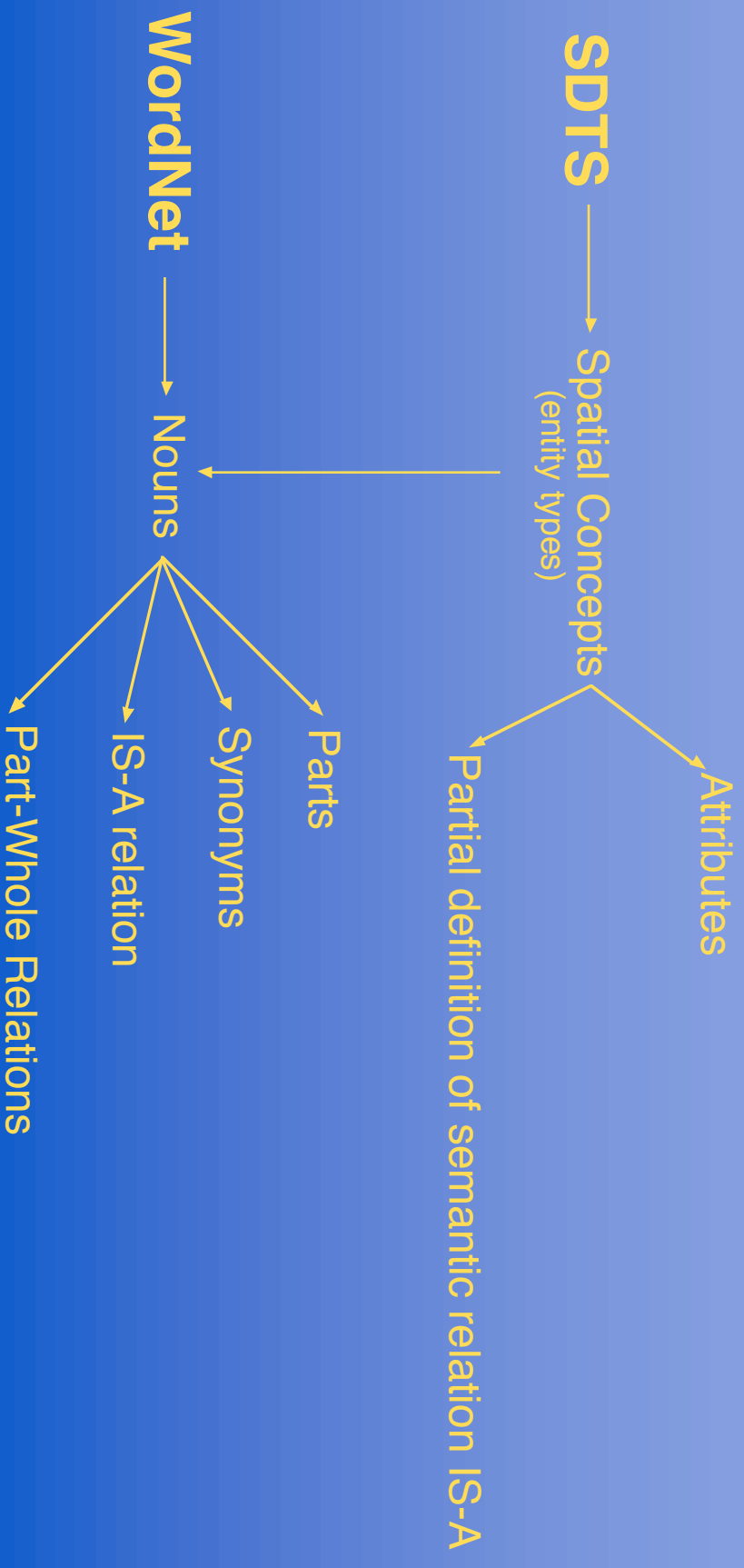
Solution_a = $\{E_{a1}, \dots, E_{ap}\}$

- The query schema is compared with each database schema
- Ranking based on the degree of similarity

User Ontology: Entity Classes



User Ontology: Entity Classes



Entity Class Definition

STADIUM

Informal Definition
large open unroofed structure in which athletic events are held

Synonyms:
bowl
arena

Edit

Semantic Relations

Is-a **Part-of** **Whole-of**

construction athletic field

Distinguishing Features

Parts	Functions	Attributes
athletic field spectator stands ticket office midfield dressing room foundation	recreate play practice	covered/uncovered sports type lighted/unlighted owner type user type name
Synonyms sports field playing field	Synonyms exercise	Synonyms

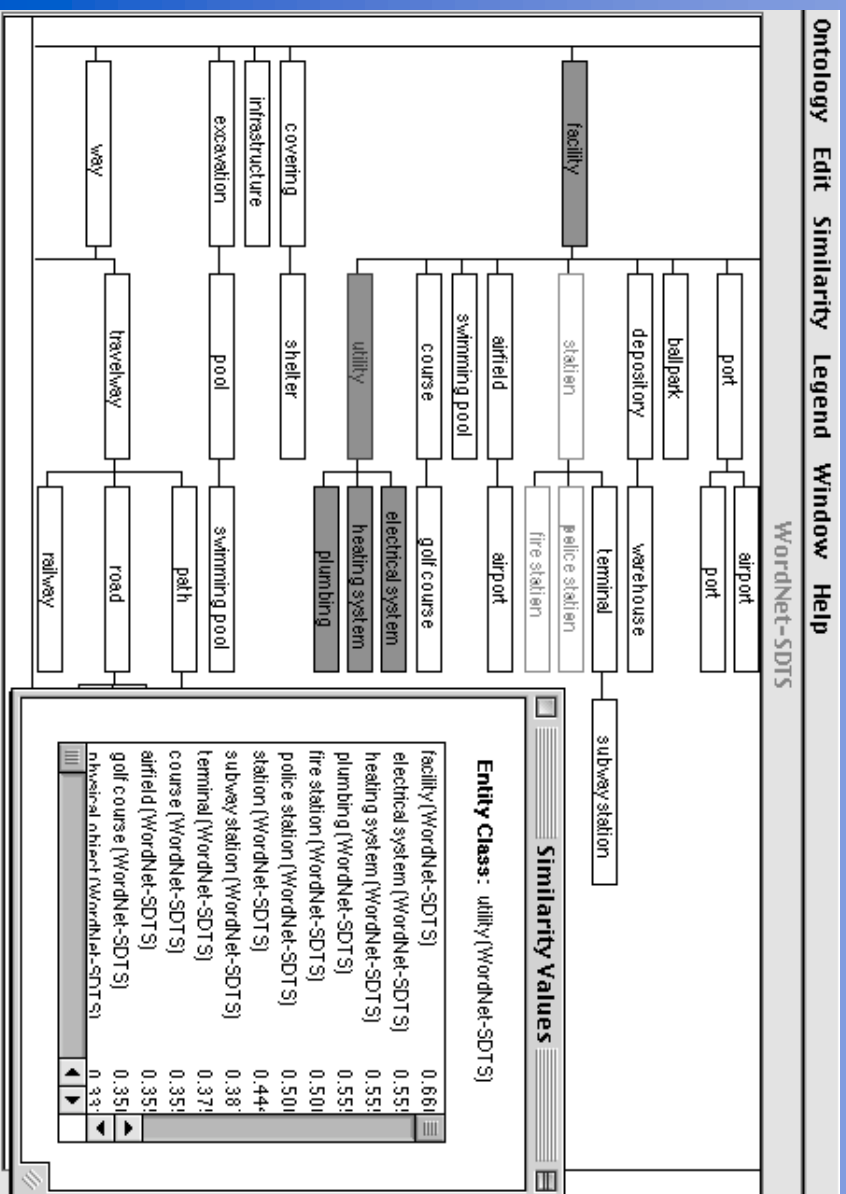
User Ontology: Similarity Assessment

$$S_t(c_1, c_2) = \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|} = \frac{|C_1 \cap C_2| + \alpha(c_1, c_2)(|C_1 \setminus C_2| + (1 - \alpha)(c_1, c_2))|C_2 \setminus C_1|}{|C_1 \cup C_2| + \alpha(c_1, c_2)(|C_1 \setminus C_2| + (1 - \alpha)(c_1, c_2))|C_2 \setminus C_1|}$$

where

- S_t similarity functions for type of distinguishing features t
- C_1, C_2 entity classes
- C_1, C_2 distinguishing features of type t for entity class c_1 and c_2
- $\alpha()$ coefficient of asymmetry
- $| |$ set cardinality

Semantic Similarity Evaluation



Database Schema: Relational Database

- Entities: names, attributes, primary key and foreign key
- Foreign keys (FK): relations that they belong and refer to
- Attributes: names

From Ontology to DB Schema

Semantic Relation Transformation

Is-A (1) An entity for each children

Part-Whole (1) New Relation

(2) Add Foreign Key

Whole-Of (1) New Relation

(2) Add Foreign Key

From Ontology to DB Schema

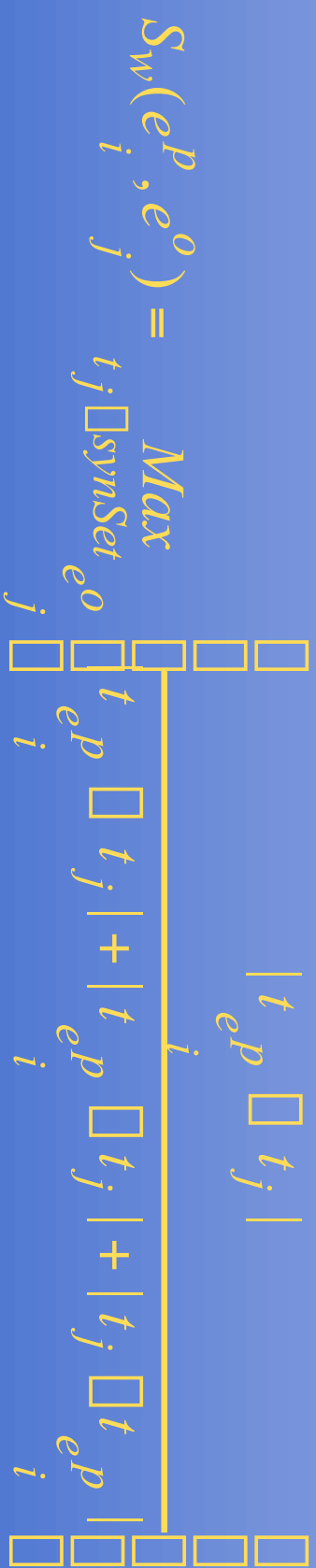
```
Entity_class{
  name: {heating_system}
  is_a: {utility}
  part_of: {}
  whole_of: {{pipeline, piping, pipe}}
```

```
Heating_system (FK_utility, FK_pipeline )
Foreign_key: FK_utility reference to Utility
Foreign_key: FK_pipeline reference to Pipeline
Pipeline (FK_heating_system, FK_plumbing_system )
Foreign_key: FK_heating_system reference to
Heatin_system
Foreign_key: FK_plumbing_system reference to
Plumbing_system
```

Comparing DB Schemas

- String Matching over entities' names, attributes domains and foreign keys' references.
- Matching of semantic neighborhood.

String Matching



Semantic Neighborhood

- Semantic Neighborhood □ Similarity
- Semantic relations:
 - In the query, relations are represented by foreign keys
 - In DB, the relations are represented by foreign keys or by domain value of an attribute.

Semantic Neighborhood

$$S_n(e_i^p, e_j^o) = \frac{\sum_{i=1}^n \text{Max}_{j_i} S_w(\text{FK}_{i, e_i^p}, \text{FK}_{j_i, e_j^o}) + \sum_{i=1}^m \text{Max}_{l, j} S_w(D_{l, i, e_i^p}, \text{FK}_{j, e_j^o})}{n + \sum}$$

Extending Comparisons to Instances

- Attributes: based on type of domains
- Spatial Relations: based on content measure
- Geometry: based on area, diagonal, and geometric type.

Conclusions

- Semantic Similarity combines:
 - Matching of attributes
 - Semantic distance
- Comparing DB Schemas combines:
 - String matching of names, attributes, and foreign keys.
 - Matching of semantic Neighborhood

Future Work

- Attributes
- Experimental results with large databases
- Extending this approach to the domain of the WWW