

Representing Software Models and Database Schemas in Ontologies

Jeff Z. Pan

Department of Computing Science

University of Aberdeen, UK

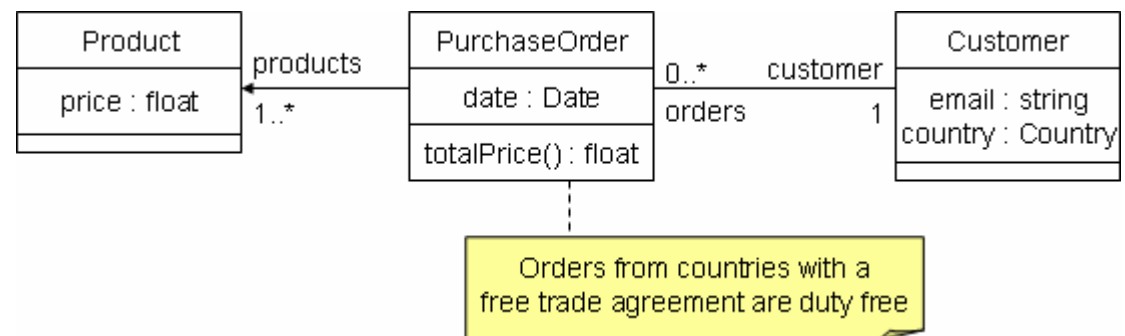


Why Representing in Ontologies (I)

Reuse of UML class diagrams (ER diagrams)

“extend our online shop to allow access of the product catalogue through a Web Service”

“port our system to a new platform”



Why Representing in Ontologies (II)



- *Model-View-Control* architecture [BMRSS1996]
 - Visual components are easier to **reuse**
 - Class diagrams often only hard-coded

- W3C Task Force on Software Engineering
 - At the Semantic Web Best Practices and Deployment WG
 - <http://www.w3.org/2001/sw/BestPractices/SE/>

Ontologies: Benefits



- Reuse and interoperability
 - RDF and OWL models can be shared among applications and on the web
- Flexibility
 - RDF and OWL models can operate in an open environment in which classes can be defined dynamically etc
- Consistency and Quality Checking
- Reasoning

Ontologies: Disadvantages



- New concept for many users
- Reasoning complexity is high

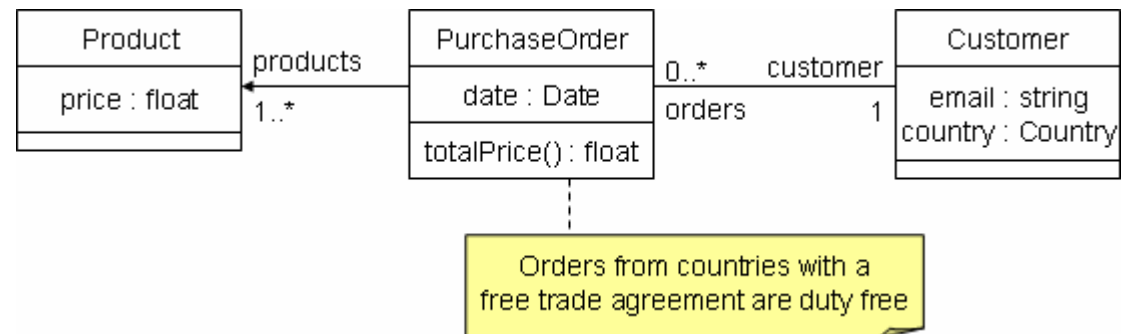
Ontologies: Disadvantages



- New concept for many users
 - Easy to make mistakes
 - Need best practices
 - Need design patterns
 - Need usability support (in general)
- Reasoning complexity is high

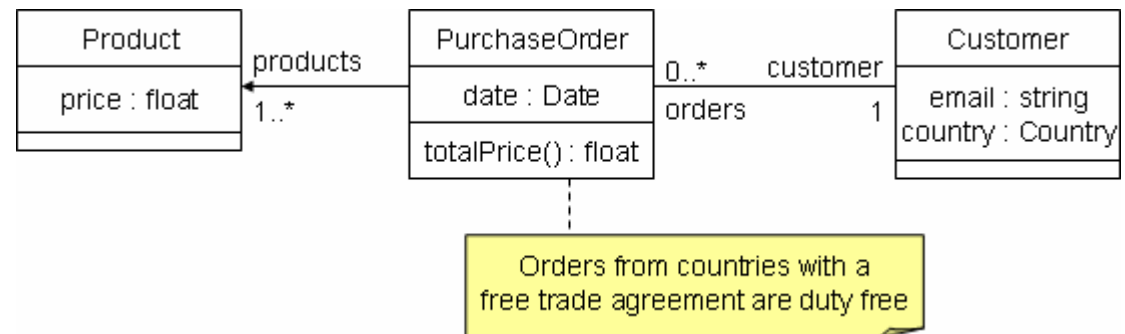
Representing Attributes

- Q: How to represent the **email** attribute of the Customer class?
- S1:
 - DatatypeProperty (email domain(Customer) range(xsd:string))



Representing Attributes

- Q: How to represent the **email** attribute of the Customer class?
- S2:
 - `DatatypeProperty` (email domain(Customer) range(xsd:string))
 - `Class` (Customer partial someValuesFrom(xsd:string))



Representing N-ary relations (I)



- Example: referee(X, P_1, P_2)
 - X is the referee in a chess game between players P_1 and P_2

Representing N-ary relations (II)



- Questions:
 - Is the following approach a proper solution?
 - Can you think of any solutions?
- S1:
 - A new auxiliary resource chessGame1
 - Three new properties ref, player1, and player2
 - **Individual** (chessGame1 **value** (ref X)
value (player1 P₁)
value (player2 P₂))

Representing N-ary relations (II)



- Questions:
 - Is the following approach a proper solution?
 - Can you think of any solutions?
- S2:
 - A new auxiliary resource chessGame1
 - Three new properties ref, player1, and player2
 - **Individual** (chessGame1 **value** (ref X)
value (player1 P₁)
value (player2 P₂))
 - **ObjectProperty** (ref **Functional**)
 - ...

More Best Practices and Design Patterns ...



- Check the homepage of the W3C Semantic Web Best Practices and Deployment WG

✓ <http://www.w3.org/2001/sw/BestPractices/SE>

Ontologies: Disadvantages



- New concept for many users
- Reasoning complexity is high

Query Answering is Hard

- **Open problem**: whether query answering in OWL DL is decidable
- And it is known to be hard
 - *SHIQ* with queries allowing only simple properties in [Ortiz et al., 2006]
 - Data complexity: co-NP-complete
 - *SHIQ* [Glimm et al., 2007]:
 - Combined complexity: 2EXPTIME
 - Data complexity: co-NP-complete
 - In fact, co-NP-complete even for **very small fragments** of *SHIQ* [Calvnesse et al., 2006], e.g. allowing
 - Full negations in the left hand side of class inclusions, or
 - Disjunctions or universal restrictions in the right hand side of class inclusions

OWL 2: The Next OWL

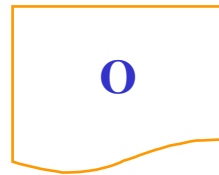


- OWL 2 is on its way
 - Extend OWL DL with small but useful set of features
 - W3C WG:
 - <http://www.w3.org/2007/OWL/>
- **Profiles**
 - Tractable fragments
- Hard decision:
 - Trade-off between **Expressiveness** and **scalability**
 - Users can **either** choose the full OWL 2 language
 - **Or** some tractable fragments

Approximations

- Approximation of queries [Struckenschmidt and van Harmelen, 2002; Wache et al., 2005]

$Q \rightarrow Q_{\text{Simple}}$



- Approximations of ontologies [DLDB, 2004; OWLJessKB, 2003]

Q

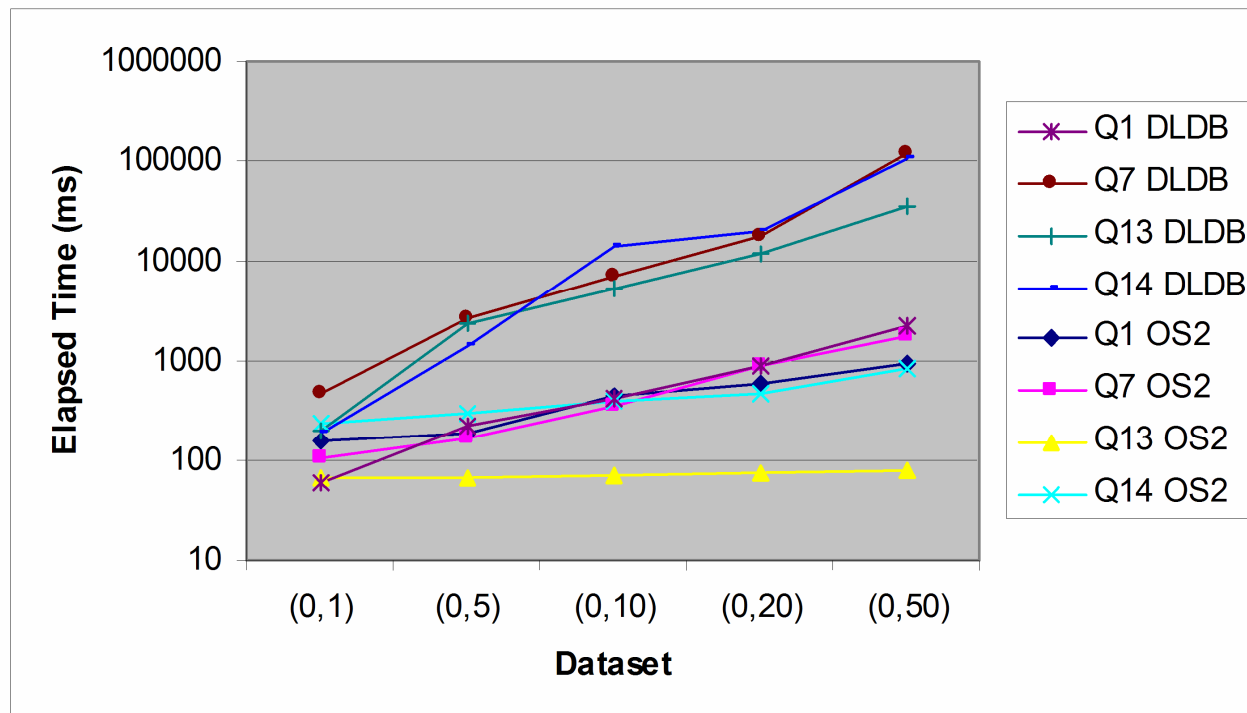


Approximating ontologies

- **Problem:**
 - It is still an open problem whether query answering in OWL DL is decidable
 - Syntactic approximations can guarantee neither soundness nor completeness
 - Existing syntactic approximations can guarantee neither soundness nor completeness
- **Example (syntactic approximation)**
 - An OWL DL source ontology $O = \{\exists_{\geq 50} \text{goodF} \sqsubseteq P$ (anyone with at least 50 good friends is popular), $\text{goodF} \sqsubseteq \text{hasF}$ (if A has good friend B, then A has friend B), $\text{goodF}(\text{john}, \text{kate})$ (john has a good friend kate)}
 - A syntactic approximation in *ALC*: $O' = \{\exists_{\geq 1} \text{goodF} \sqsubseteq P$ (anyone with at least a good friend is popular), $\text{goodF}(\text{john}, \text{kate})$ (john has a good friend kate)}
 - **Unsoundness:** O' entails $P(\text{john})$ (john is popular) but O does not
 - **Incompleteness:** O entails $\text{hasF}(\text{john}, \text{kate})$ (john has a friend kate) but O' does not

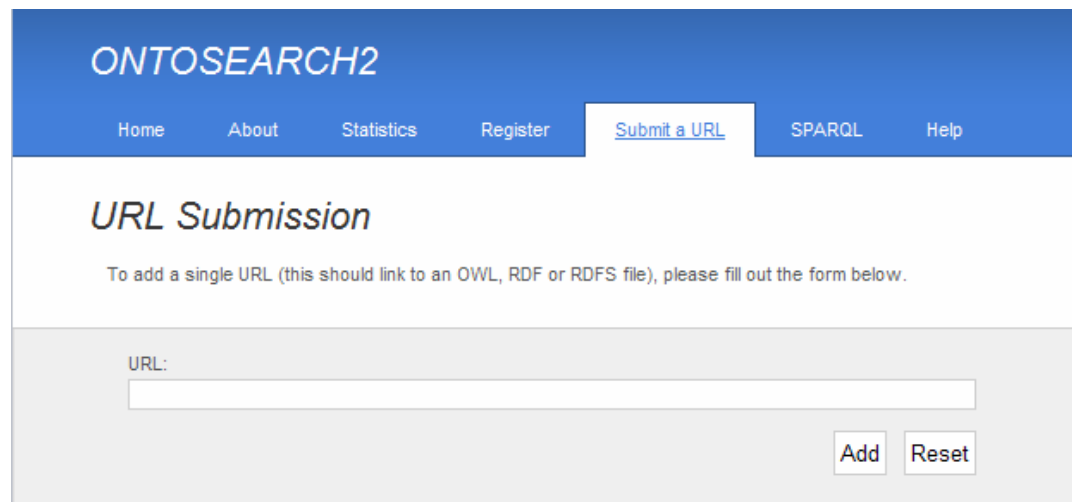
Semantic approximation

- Semantically approximate an source OWL ontology O_s with its least upper bound O_t
 - Implemented in ONTOSEARCH2 [AAAI2007]
- DLDB vs. ONTOSEARCH2



ONTOSEARCH2

- Submit ontologies to ONTOSEARCH2

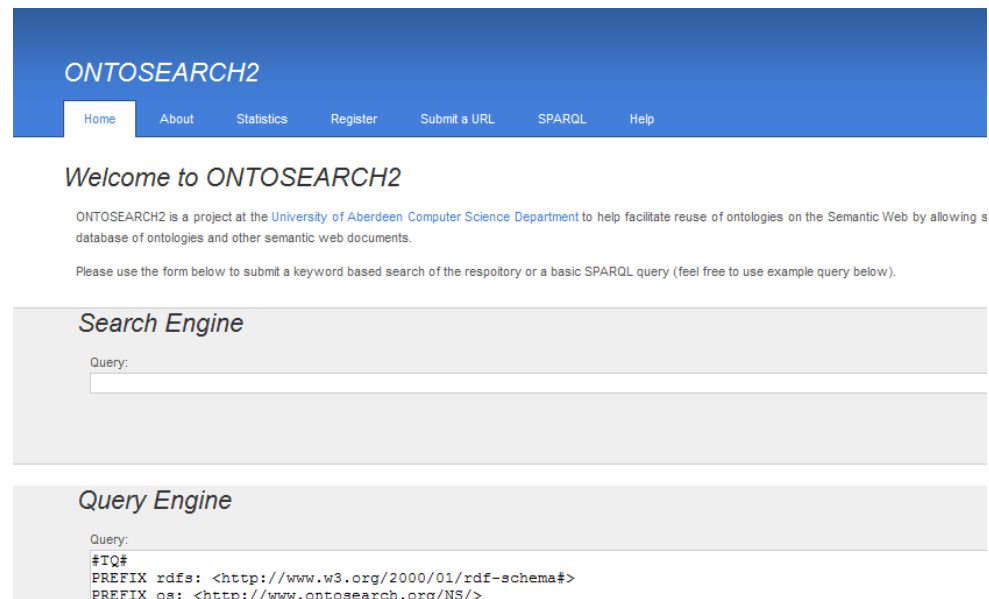


The screenshot shows the ONTOSEARCH2 website interface. At the top, there is a blue navigation bar with the logo 'ONTOSEARCH2' and several menu items: 'Home', 'About', 'Statistics', 'Register', 'Submit a URL' (which is highlighted), 'SPARQL', and 'Help'. Below the navigation bar, the page title is 'URL Submission'. A sub-header reads: 'To add a single URL (this should link to an OWL, RDF or RDFS file), please fill out the form below.' The form itself is a light gray box containing a text input field labeled 'URL:' and two buttons: 'Add' and 'Reset'.

- How do you know that your ontology is DL-Lite?
 - Use the checker - <http://www.ontosearch.org/DLLiteChecker/>

ONTOSEARCH2 – Implementation

- ONTOSEARCH2 (<http://www.ontosearch.org/>)
 - ontology crisp/fuzzy query answering
 - ontology searching



The screenshot shows the ONTOSEARCH2 website interface. At the top, there is a blue navigation bar with the title "ONTOSEARCH2" and a menu with links: Home, About, Statistics, Register, Submit a URL, SPARQL, and Help. Below the navigation bar, the main content area is white. It starts with the heading "Welcome to ONTOSEARCH2" followed by a paragraph explaining the project's purpose: "ONTOSEARCH2 is a project at the University of Aberdeen Computer Science Department to help facilitate reuse of ontologies on the Semantic Web by allowing a database of ontologies and other semantic web documents." Below this is a note: "Please use the form below to submit a keyword based search of the repository or a basic SPARQL query (feel free to use example query below)." There are two search engines provided. The first is labeled "Search Engine" and has a text input field with the label "Query:". The second is labeled "Query Engine" and has a text input field with the label "Query:" containing the following example query:

```
#IQ#  
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
PREFIX os: <http://www.ontosearch.org/NS/>
```



MOSTPROJECT

<http://most-project.eu>

- MOST
 - Marrying Ontologies with Software Technologies
 - will improve software engineering
 - by leveraging ontology and reasoning technology

- Approximation technologies will be developed
 - Good balance between expressiveness and scalability
 - Given certain tasks in ODSD (ontology-driven software development)

Thank you!

Jeff Z. Pan

Department of Computing Science
University of Aberdeen, UK

