Ontology-Driven Sub-Query Extraction in UnIT-Net IEDI

Vadim Ermolayev
Natalya Keberle
Vladimir Shapar
Vladimir Vladimirov

Zaporozhye State University,
Ukraine

UnIT-Net: IT in University Management Network
TEMPUS/TACIS MP-JEP-2010-2003

ISTA’04, Salt Lake City, 15-17.07.2004
Title: IT in University Management Network

Objective(s):
- Creation of the Ukrainian National “Network of Excellence”
- Dissemination of the best practices – IT in University Management
- Elaboration of the Specifications recommending reasonable ways of using IT in University Management
- Design and implementation of the Research Prototype of the National Infrastructure for Electronic Data Interchange (mediator-wrapper, hybrid knowledge representation)

Participants:
- Kherson State University (project coordinator)
- Ministry of Education and Science of Ukraine
- Kharkiv national University
- Zaporozhye State University
- University of Nice – Sofia Antipolis, France
- Glasgow Caledonian University, UK

URL: http://www.unit-net.org.ua/
I have a query to all of you in terms (and in language) that I understand.

We provide IR-s annotated in terms suitable for us.
And, normally, we do not care about the others.
Q1: How to add a new IR to a Retrieval System?

- **A1.1**: Wait until its owner volunteers to provide the annotation *(IR registration)* – **UnIT-Net**
- **A1.2**: Crawl the Web for publicly available IR-s

Q2: How to align our beliefs on IR to its actual state?

- **A2.1**: Wait until the owner informs about the changes, then align – **UnIT-Net**
- **A2.2**: Mine the changes from IR-s regularly. Align accordingly
Problems to Solve:

- **Q3**: How to formulate & perform queries to such an IR Grid?

- **A3**: UnIT-Net
  - Assist a User to **formulate** his query in terms familiar to him
  - **Transform** the query to the terms commonly accepted for the Domain
  - **Decompose** the query to the set of sub-queries
  - **Forward** sub-queries to appropriate IR wrappers
  - ...
  - (**Fuse** and) **Return** the **results** to the User
IEDI Architecture in a Nutshell

User Layer

Mediator Layer

Process queries

IR Wrapper Layer

IR Layer

Request to formulate a query

Q-ry Results in terms of MDO

a User

IEDI Mediator

Query Formulation Server

Sub-Query Extraction Server

Sub-Query Execution Server

Results Mark-Up Translation Server

MKB

AUPO

MDO

IRDMO

IR Wrapper Web Service

IR Wrapper Web Service

IR Wrapper

IR Wrapper

IR Wrapper

IR Wrapper

IR

IR

IR Providers

Knowledge Engineers

Register IR-s

Maintain IR changes
Ontology and Language Aspect:

Query Formulation -> RDQL

Query Transformation -> RDQL

Sub-Query Extraction

Mediator

IR Wrapper

Perform Sub-Queries

- semi-automatic

- automatic

UPO

MDO

MDO Core

IRO

OWL-DL

OWL-DL

OWL-DL

OWL-DL
A Walkthrough Example

- Return the list of the 1-st year CS students who:
  - had received maximal grade in Mathematics at the entrance examinations
  - and have failed to pass the 1-st Term examination in any basic course in Mathematics
Return the list of the 1-st year CS students who:

- had received **maximal grade in Mathematics** at the entrance examinations
- and have **failed** to pass the 1-st Term examination in **any basic course in Mathematics**

Mathematics:
- Math Analysis
- Linear Algebra
- Analytical Geometry
- ...

Ontology

Univ. Entrant IR

CS Student IR

A Walkthrough Example
A Walkthrough Example

Return the list of the 1-st year CS students who:

- had received maximal grade in Mathematics at the entrance examinations
- and have failed to pass the 1-st Term examination in any basic course in Mathematics

Mathematics:
- Math Analysis
- Linear Algebra
- Analytical Geometry
...
MDO and IRO-s for the Example
IRO Registration is NOT a Simple Join

Note:
- IR-Domain Mapping Ontology (IRDMO) provides only minimally necessary mappings:
  - Slots defined for the concept
  - But NOT the slots inherited from its superclass

Requires Late Binding
Retrieve "CS" students who had got maximum grade (5) in "Mathematics" at the University entrance examinations and have failed to pass the 1-st semester examination (grade $\leq 2$) in "Math Analysis" or "Linear Algebra".

Display: firstname, secondname, lastname, spec_name, discipline_name
**Sub-Query Extraction**

**Steps 1,2:** Find **Determining Concepts (DC)** their properties are listed in the **WHERE** clause of our query.
Mappings Recorded in IRDMO
(MDO ↔ Stud IRO)

- Late Binding required to resolve the mapping
Resolved Mappings: MDO↔Stud IRO

Steps 3,4: Resolve Concept and Slot Mappings
Result: a (subset of) Billet Query (ies) per IR

Orphan elements

Student
  surName
  secondName
  name

Speciality
  specialityName = “CS”

Session Exam
  examName = “Math Analysis”
  OR “Linear Algebra”
  grade <= 2
  semesterNum = 1

Exam
  exams_passes

Entrant
  entrantgrade = 5

Session
  examName = “Math Analysis”
  OR “Linear Algebra”
  grade <= 2
  semesterNum = 1

Discipline
  disciplinename = “Mathematics”

Orphan elements – to be deleted at Query Clarification - Step 5
ODSQE at a Glance

Features:
- IRDMO provides only minimally necessary mappings.
- Late Binding is used to resolve mappings in subclass/superclass chains (further development - meronymy).
- Multiple mappings result in series of Billet Queries per IR.
- MDO Core is used to fix the minimal set of concepts common to all IR-s.
ODSQE Properties

- Builds the set of IRO queries for a specific MDO query (**existence**)

- Builds the unique set of IRO queries for a specific MDO query (**uniqueness**)

- The recall of a hypothetic MDO query is less or equal to the combined recall of the set of the produced IRO queries (**complete coverage**)

To Conclude …

- Initial **proof-of-concept implementation** of ODSQE is done
- **UnIT-Net IEDI** Sub-Query Extraction Server is **under development**

**Further development:**
- Richer semantic relationships
- More intelligence in Late Binding for non-equivalent concepts
That’s it …

Shall be happy to know the answers

These slides are available from: http://eva.zsu.zp.ua/eva_personal/evapubs.htm