1. Background

- Vast online information environments
  - billions of digital documents
  - many different natural languages
  - distributed document production and publication: no generally agreed rules
  - general lack of control in the process
  - much spam and other unwanted material

2. Ontologies

- Ontologies model semantics
  - concepts
  - rich relationships
  - support inference
  - application means resource annotation
  - closely related to thesauri

- Belief: ontologies can solve the vocabulary problem
  - represents the semantics of resources (documents) better than pure natural language
  - retrieval becomes correct and accurate
  - desired: a universal world model, and a controlled language for description and reasoning about this model

Issues in Classification and Indexing

- Index languages -
  - modeling - coverage, viewpoint
  - maintenance - ageing, cost

- Indexing -
  - specificity, exhaustivity, consistency
  - cost - where paid, who pays?
  - The over-specificity the devices created often lead to poor recall and thus they were soon mostly abandoned
Any Room for Ontologies?

Should one thus discard ontologies? or other vocabulary control tools?

In practice, realism tells us that:
- There will never be a comprehensive & up-to-date ontology – cf. UDC, which had large development community support
- No one will annotate for free, for ever & consistently
- No one can do that exhaustively and from many viewpoints emerging, e.g., in future
- In fact, less than 0.3% of web pages had Dublin Core metadata (Rasmussen 2003)

There is no alternative to searching unannotated collections. Automatic annotation does not solve the problem - if one aims at the good semantics required in the Semantic Web.

Searching Unannotated Collections through Ontologies

Searching ontologies can:
- Provide conceptual organization
- Support direct access to textual content
- Translate between concepts and textual variation
- Translate between natural languages
- Hide search engines/query languages
- May support other media/structures/features
- Be light-weight, narrow, and no world models
- Be personal, group or small community support
- Be versions, mutually incoherent, easily modifiable
- Be easily disposable, perhaps tradable

Searching through Ontologies

Need to solve the vocabulary problem from concepts to textual expressions:
- Three layers:
  - Concepts - for user interaction
  - Expressions - for system use
  - Strings to match - for system use

Need to provide a handy concept browser and query constructor.

QUCCOO: Principles

QUCCOO: Query Construction with Ontologies for direct content access:
- Based on the three levels...
- Aims to provide independence of:
  - Expression variability (nutraceutical?)
  - Natural language (French?)
  - Collection (intranet, Web,...)
  - Indexing (lemmatization, compounds?)
  - Availability of metadata & world model
  - Engine & query language (Lemur, Trip, Google,...)
- You just select your concepts, targets and go!
- Point, click and go.

QUCCOO: status

Web application, uses state-of-the-art Servlet technology
- Supports diverse full-text database engines (Trip, InQuery, etc.) as well web search engines (e.g., Google)
- Supports diverse collections
- Intuitive: simple interface to access information
- Supports multilingual search and various index types
QUCCOO: Architecture

Client-Applet

Server-Servlet

Request

Concepts

Query

Expand Query

Ontology server

Results

Document servers

Feza Baskaya – Anne Kakkonen - University of Tampere - 2007

Quccoo - interface

Search box

Options button

Concepts given by user

Search button

Ontology Tree

Feza Baskaya – Anne Kakkonen - University of Tampere - 2007

Quccoo - interface

Search Engine Selection

Liberality Selection

Expansion Level Selection

Feza Baskaya – Anne Kakkonen - University of Tampere - 2007
Quccoo - interface

Trip Database Engine results

Ontology in Finnish

Quccoo - interface

Inquery Database Engine Results

Ontology in Finnish

Quccoo - interface

Google Results in Finnish

Ontology in Finnish

Quccoo - interface

Google Results in English

Ontology in Finnish

Quccoo - interface

Google Results in Swedish

Ontology in Finnish

Quccoo - interface

Google Results with Logging Facility

Ontology in Finnish
Quccoo - interface

Ontology in English

Google Results in English

Quccoo - interface

Ontology in English

Google Results in Finnish

Quccoo - interface

Extra keyword(s) added by user

4. ShOE: Creating ontologies

- Search ontology editor - for creating ontologies
  - supports the 3 layer architecture of QUCCOO
  - intuitive; easy to learn and use
  - automatic support for the human editor

- Multilingual in many aspects
  - GUI, User Interface language can be changed
  - Concepts names can be edited/displayed on-the-fly in different languages
  - Expressions can be edited/displayed on-the-fly in different languages.

ShOE: implementation

- Well-designed modular object-oriented architecture based on MVC paradigm
- Platform independent; written in Java
- Flexible; e.g. uses XML as file format, configurable tables, with XML configurable menu structure
- Robust
- Extensible via Plug-ins

ShOE - Main window

Concept hierarchy tree

Expression window
6. Conclusion

- ShOE and QUCCOO are one answer to problems in semantic information access.
  - Light-weight, disposable search ontologies for full content access.
  - Independencies:
    - Collections (partially), indexing ways.
    - Availability of metadata / annotations.
    - Changes of needs, variability of "world models".
    - Search engines, query languages.
    - Vocabulary variation and natural languages.
  - A compromise, different from semantic annotation or indexing, with control at the user end.

User testing

Cross-language Web search

- Test persons:
  - 40 students from the University of Tampere and Pirkanmaa polytechnic.
- Ontology:
  - Combination of two ontologies: Food concepts and geographical concepts.
- 2 interfaces:
  - QUCCOO + interface without ontology (basic Google search).
- 4 simulated search tasks:
  - Two tasks with one interface and two with the other.

Analysis

- Log files
  - Queries.
  - Relevance assessments (scale 0-4).
- Questionnaires
  - Opinions about ontology and QUCCOO-interface.

Results: search success

- No significant difference between systems.
  - QUCCOO performed better when strong query structure was needed ("alcoholic beverage").
  - In most self-formulated queries no phrases were used.
  - QUCCOO helps persons who are not used to formulate structured queries.

Results: opinions

- "Structure of the ontology was logical."
- "Finding search concepts needed in the tasks in ontology was easy."
- "Using the ontology was effortless."
  - 92% agreed in all.
Results: opinions
- 32/40 thought that QUCCOO-interface was easier to use
- 32/40 liked QUCCOO better
- Why?
  - Helped users to clarify task topic and to find related search keys
  - Made cross-language search easy (in 80% of direct searches some dictionary was used to help query formulation)

Discussion
Thank you!
Over to you ... questions?