XIRQL: A Query Language for Information Retrieval in XML Documents

Norbert Fuhr, Kai Großjohann
University of Dortmund
Germany
<book class="H.3.3">
  <author>John Smith</author>
  <title>XML Retrieval</title>
  <chapter> <heading>Introduction</heading>
    This text explains all about XML and IR.
  </chapter>
  <chapter>
    <heading>XML Query Language XQL</heading>
    <section>
      <heading>Examples</heading>
    </section>
    <section>
      <heading>Syntax</heading>
      Now we describe the XQL syntax.
    </section>
  </chapter>
</book>

Elements:
- start tag
- end tag
- content
- attribute
Introduction

XML Query Language XQL

We describe syntax of XQL

Examples

Syntax
XML query languages

- **Data-centric view:** XML as exchange format for structured data
- **Document-centric view:** XML as format for representing the logical structure of documents

*W3C WG proposal* for XML query language: **XQuery**

Focuses on data-centric view here:

- Information Retrieval for document-centric view
- Starting point: XQL (XPath)
Path condition: parent/child node chapter/heading
Path condition: ancestor-descendant chapter//heading
Filter wrt. structure:  
//chapter[heading]
Filter wrt. content:

\[ /\text{document[@class="H.3.3" \land author="John Smith"]} \]
XQL properties

✓ Conditions wrt. logical structure
✓ Conditions wrt. content
✓ Results are always complete elements
- Boolean Retrieval (poor retrieval quality)
- Relevance-oriented search (irrespective of structure) not supported
- Few data types only
XIRQL: XML IR Query Language

Extend XQL by:

- probabilistic Retrieval with weighted document indexing
- Relevance-oriented search (irrespective of structure)
- (Extensible) data types with vague predicates
Probabilistic Retrieval with XIRQL

**Problem:** weighting of different forms of occurrence of terms

```
/document[./heading \e "XML" \v ./section//\* \e "XML"]
```
Weighting of term occurrences in documents

a) Weighting wrt. single query conditions
→ Possible overlapping of query conditions
→ Dependent probabilistic events
→ Only probability intervals for answers
→ No linear ranking of documents
Weighting of term occurrences in documents

b) Weighting wrt. document parts
   
   → Term weighting depends on context of term occurrence
   
   → All occurrences within same context refer to same probabilistic event
   
   → Only identical and independent events
   
   → Point probabilities for answers
   
   → Linear ranking of documents
Index nodes as units for term weighting

Application of known indexing functions (e.g. tf*idf)
Probabilistic events and event expressions

**Problem:** combination of term weights consistent with probability theory

- *Basic event:* term occurrence in an index node
- Basic events are independent (different terms, same term in different index nodes)
- *Event expressions* describe combination of basic events in a document wrt. a query
Event expressions

//section[./* $\ni$ "XQL" $\land$ ./* $\ni$ "syntax"]

[5,XQL] $\land$ [5,syntax]
Event expressions

/event [/* "XQL" ∧ /* "syntax"]

([3,XQL] ∨ [5,XQL]) ∧ [5,syntax]
Evaluation of event expressions

1. Transform event expression into disjunctive normal form

   \[ e = C_1 \lor \ldots \lor C_n \]

   \( C_i \): Conjunction of event atoms

   Event atom: positive or negated basic event

2. Application of inclusion/exclusion formula:

   \[
   P(e) = P(C_1 \lor \ldots \lor C_n)
   \]

   \[
   P(e) = \sum_{i=1}^{n} (-1)^{i-1} \left( \sum_{1 \leq j_1 < \ldots < j_i \leq n} P(C_{j_1} \land \ldots \land C_{j_i}) \right)
   \]
Relevance-oriented search

(Queries irrespective of document structure)

1) Restrict possible answers
   (not all elements suitable)
2) Retrieval strategy: return most specific element satisfying the query
   but: combination with weighted indexing?

Solution:

1) Index nodes as roots of possible answers
2) Augmentation as concept for computing tradeoff between indexing weights and smallness of answers
Index nodes for relevance-oriented search

author: John Smith

chapter: Introduction

title: XML Retrieval

class="H.3.3"

This...

XML Query Lang. XQL

We describe syntax of XQL

Examples

Syntax
Augmentation

...by disjunction

Example query: syntax ∧ example
Augmentation

...by disjunction

Example query: XQL
Augmentation

...with augmentation weight

Example query: XQL
XIRQL: Data types with vague predicates

XML markup allows for detailed markup of text elements

- Exploit markup for more precise searches
- Consider also vagueness and imprecision of IR
- Data types with vague queries
  
  ``Search for an artist named Ulbrich, living in the Rhine-Main area of Germany about 100 years ago``

  Ernst Olbrich, Darmstadt, 1899

- (Extensible) data types for document-centric view
  
  (person names, dates geographic locations, classifications/images, audio,...)
Extensible type hierarchy

- Extensible type hierarchy with vague predicates for each data type
  1) **text**: substring-match
  2) **Western language**: single word search, truncation, word distance
  3) **English text**: stemming, noun phrases

- Data types of XML documents defined in extended DTD (XML schema)
Processing of XIRQL queries

1. Translation into path algebra
   (results are always complete elements of original documents)

2. Query optimization

3. Development of algorithms for best match queries
   a) Access paths with ranking wrt. single conditions
      (Pfeifer & Fuhr 93, Fagin 96, Güntzer et al. 00)
   b) Access paths ordered by document (text search)
      (Buckley & Lewit 85, Moffat & Zobel 97)
Summary and conclusions

XIRQL supports

- Combination of structural conditions with probabilistic weighting
- Relevance-oriented search by augmentation
- Extensible data types with vague predicates

HyREX (Hypermedia Retrieval Engine for XML):
   Open source prototype implementing XIRQL