XIRQL: Eine Anfragesprache für Information Retrieval in XML-Dokumenten

Norbert Fuhr
Universität Duisburg
Outline of Talk

I. XML retrieval
II. XIRQL: XML IR Query Language
III. XIRQL vs. XQuery
IV. User Interface
V. INEX: Initiative for the Evaluation of XML Retrieval
VI. Summary
I. XML documents

<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
This is the introduction to the document. It describes the XML Query Language XQL. The section on examples provides syntax of XQL, and we also discuss the syntax of XQL in detail.
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: XPath (XQL)
XPath

Path condition: parent/child node
chapter/chapter/heading
XPath

Path condition: ancestor-descendant
chapter//heading
XPath

Filter wrt. structure: //chapter[heading]
XPath

Filter wrt. content:

/document[@class="H.3.3" ∧ author="John Smith"]
XPath properties

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

Extend XPath by:

- Probabilistic retrieval with weighted document indexing
- Relevance-oriented search (irrespective of structure)
- (Extensible) data types with vague predicates
- Structural relativism
II.1 Probabilistic Retrieval with XIRQL

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

/\document[.//heading ∋ "XML" ∨ .//section///* ∋ "XML"]}
Weighting of term occurrences in documents

a) Weighting wrt. single query conditions

\[ P(././heading \ni \text{"XML"},d) = 0.5 \]
\[ P(././section/\text{*} \ni \text{"XML"},d) = 0.7 \]

→ 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[./* \є "XQL" \∧ ./* \є "syntax"]

[5,XQL] \∧ [5,syntax]
Event expressions

/document/chapter [.//* ↗ "XQL" ∧ .//* ↗ "syntax"]

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

(as in probabilistic Datalog)

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:

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

\[ P(e) = \sum_{i=1}^{n} (-1)^{i-1} \left( \sum_{1 \leq j_1 < \ldots < j_i \leq n} P(C_{j_1} \land \ldots \land C_{j_i}) \right) \]
II.2 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 specificity of answers
Index nodes for relevance-oriented search

This...
Augmentation

...by disjunction

Example query: syntax ∧ example
Augmentation

...by disjunction

Example query: XQL
Augmentation

...with augmentation weight

Example query: XQL
II.3 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)
II.4 Structural Relativism

- Drop distinction attribute/element:
  ~author searches for attribute or element

- Generalize to data types:
  #personname searches for attributes/elements of specific data type

- Exploit ontology over element names:
  region – country – continent

- Edit distance on paths:
  author=“Smith” vs. author/name vs. author/name/lastname
III. XIRQL vs. XQuery

XQuery (proposed as standard XML query language by W3C WG):

- No IR support (weighting, vague predicates, relevance-oriented search, structural relativism)
- Aggregation operators (sum, count, min, max, avg)
- Restructuring of results
XIRQL as IR extension of XQuery subset

- XQuery structure:
  FOR PathExpression
  WHERE AdditionalSelectionCriteria
  RETURN ResultConstruction

- XIRQL subset:
  FOR $X IN PathExpression
  RETURN $X
IV. User Interface

- Query formulation
- Result visualization
Query Formulation: Layout-oriented

**Author:** Terence R. Smith (University of California, Santa Barbara)

**Journal:** COMPUTER

**Issue:** Vol. 29, No. 5

**Publication Date:** MAY 1996

**Pages:** pp. 54-60

**Abstract:**

ADL will provide on-line public access to a large amount of other information referenced in geographic data currently is found only at major research centers.

```xml
send xirql-query to hyrex

ARTICLE[FM/HDR/AU/* $soundex$ "smith" $and$ FM/HDR/HDR1/TI/* $stem$ "digital"]/* $stem$ "visualization"
```
Query Formulation: Structure-oriented

```
ARTICLE[FM/HDR/AU/!^sounded!^"smith" $and$ FM/HDR/HDR1/TI/!^"digital"!]"visualization"
```

The Visualization of...
Visualization of Results: Textbars
Visualization of Results: Treemaps
V. INEX: Initiative for the Evaluation of XML Retrieval

- Initially 50 groups from 20 countries (finally 27 active)
- Documents:
  7 years of IEEE-CS journals (12107 articles, 494 MB)
- Queries:
  30 content-only, 30 content+structural conditions
- Results due: September 15, 2002
- Relevance judgements due: November 20
- Final Workshop: December 9-11, 2002
Example query

- **Title:** Nonmonotonic Reasoning
- **Description:**
  Retrieve all articles from the years 1999-2000 that deal with nonmonotonic reasoning. Do not retrieve articles that are calendar/calls for papers.
- **Condition:**

```
/article=./bdy/sec ∋ “nonmonotonic reasoning” ∧
¬ .//tig/atl ∋ “calendar”]
```
VI. Summary

- Data-centric vs. document-centric view on XML (database vs. IR view)
- IR methods for XML must support uncertainty and vagueness...
XIRQL: XML query language implementing

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

HyREX: Open source XML retrieval engine:
http://ls6-www.cs.uni-dortmund.de/ir/projects/hyrex