Enhancing Content-And-Structure Information Retrieval Using a Native XML Database

Jovan Pehcevski, James Thom
School of CS and IT, RMIT University, Melbourne, Australia
email : {jovanp, jat}@cs.rmit.edu.au

Anne-Marie Vercoustre
INRIA, Rocquencourt, France
email : anne-marie.vercoustre@inria.fr

Overview

- Introduction
- INEX 2003: XML Retrieval Tasks
- XML Retrieval Approaches
  - Full-Text Information Retrieval Approach
  - Native XML Database Approach
  - Hybrid XML Retrieval Approach
- Ranking the Native XML Database Output
- Experiments and Results
- Conclusion and Future Work

Introduction

- Given the structural information explicitly present in an XML document collection, XML retrieval systems aim at more focus (by identifying and returning *document components* that are estimated likely to be *relevant* to users' information needs).

- We analyse and evaluate three systems implementing different approaches to XML retrieval:
  - *Full-text Information Retrieval Approach*, by using **Zettair** – a compact and fast full-text search engine;
  - *Native XML Database Approach*, by using **eXist** – an open source native XML database; and
  - *Hybrid XML Retrieval Approach*, which uses eXist to produce the final answers from likely relevant documents retrieved by Zettair.

INEX 2003: XML Retrieval Tasks

- The main task: **ad-hoc retrieval** of XML documents
  - searching a *static* set of documents using a *new* set of retrieval topics

- Comprises two sub-tasks:
  - a *Content-Only (CO)* sub-task, which involves 36 CO topics; and
  - a *Content-And-Structure (CAS)* sub-task, which involves 30 CAS topics and includes:
    - a *SCAS* sub-task, where structural constraints in a query are *strictly* matched;
    - a *VCAS* sub-task, where structural constraints in a query are treated as *vague* conditions.

- We focus on improving XML retrieval for CAS topics, in particular using the SCAS retrieval sub-task.

INEX 2003 CAS Topic Example

<INEX_topic topic_id=“86" query_type="CAS" ct_no=“107">

<Title> //sec[about(., 'mobile electronic payment system')] **</Title>**

<Description>

Find sections that describe technologies for wireless mobile electronic payment systems at consumer level.

</Description>

<Narrative>

To be relevant, a section must describe security-related technologies that exist in electronic payment systems that can be implemented in hardware devices.
The main interests are systems that can be used by mobile or handheld devices.
A section should be considered irrelevant if it describes systems that are Designed to be used in a PC or laptop.**</Narrative>**

<Keywords>

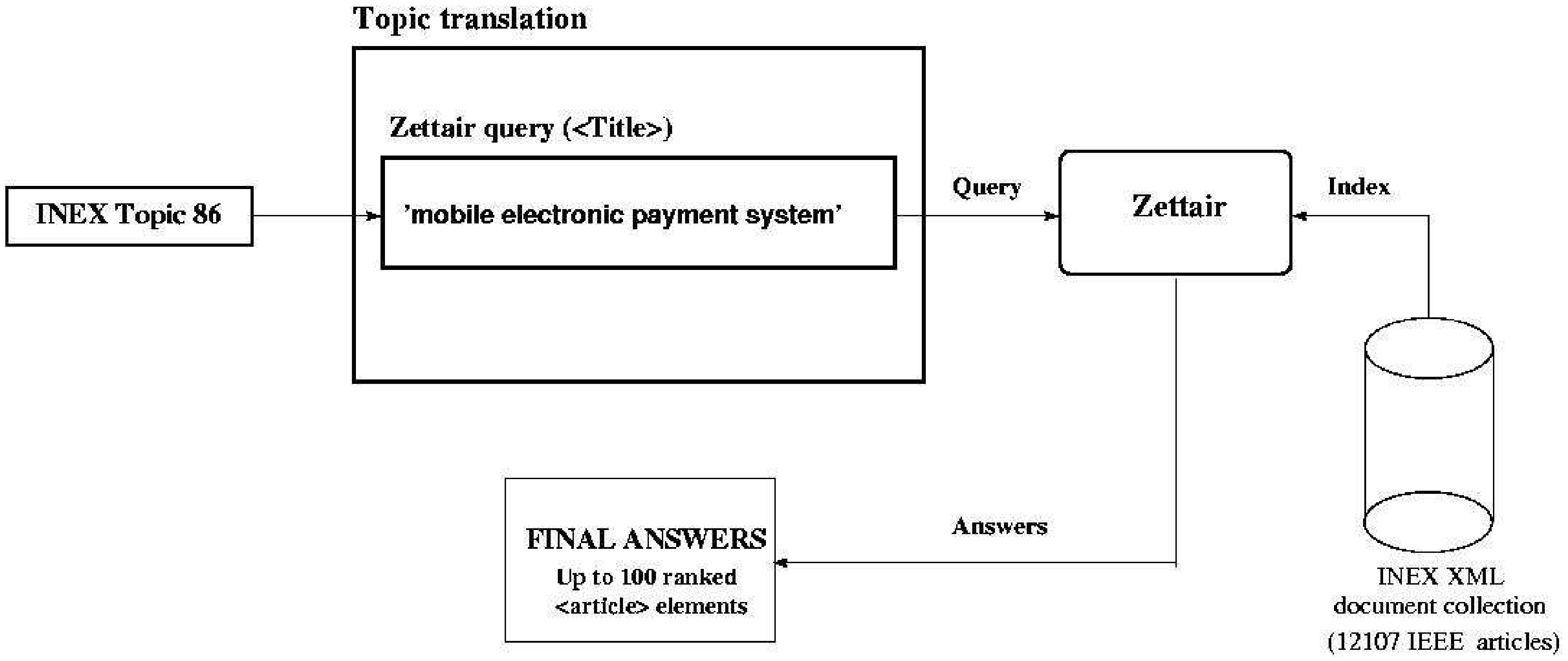
mobile, electronic payment system, electronic wallets, e-payment, e-cash, wireless, m-commerce, security **</Keywords>**

</INEX_topic>

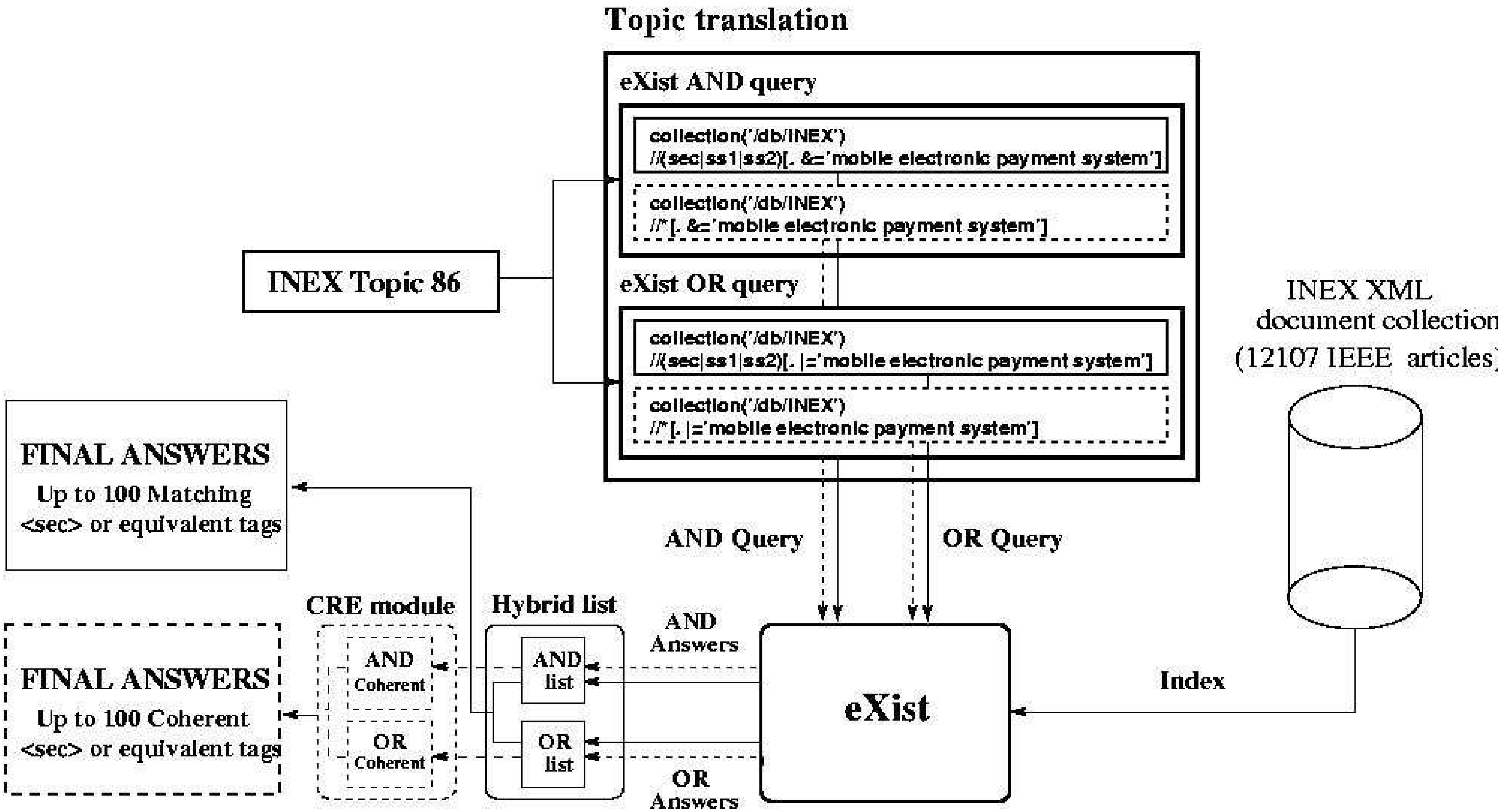
INEX 2003 CAS Topic Categories

- INEX 2003 introduces 30 CAS topics in total. Out of the topic titles we distinguish two categories of CAS topics.
    - The first category of topics seeks to retrieve full articles rather than more specific elements within articles as final answers. We refer to such topics as *Article* topics.
    - The second category of topics seeks to retrieve more specific elements within articles rather than full articles as final answers. We refer to such topics as *Specific* topics.

Full-Text Information Retrieval Approach



Native XML Database Approach



eXist AND query

```
collection('/db/INEX')
//(sec|ss1|ss2)[. &='mobile electronic payment system']
```

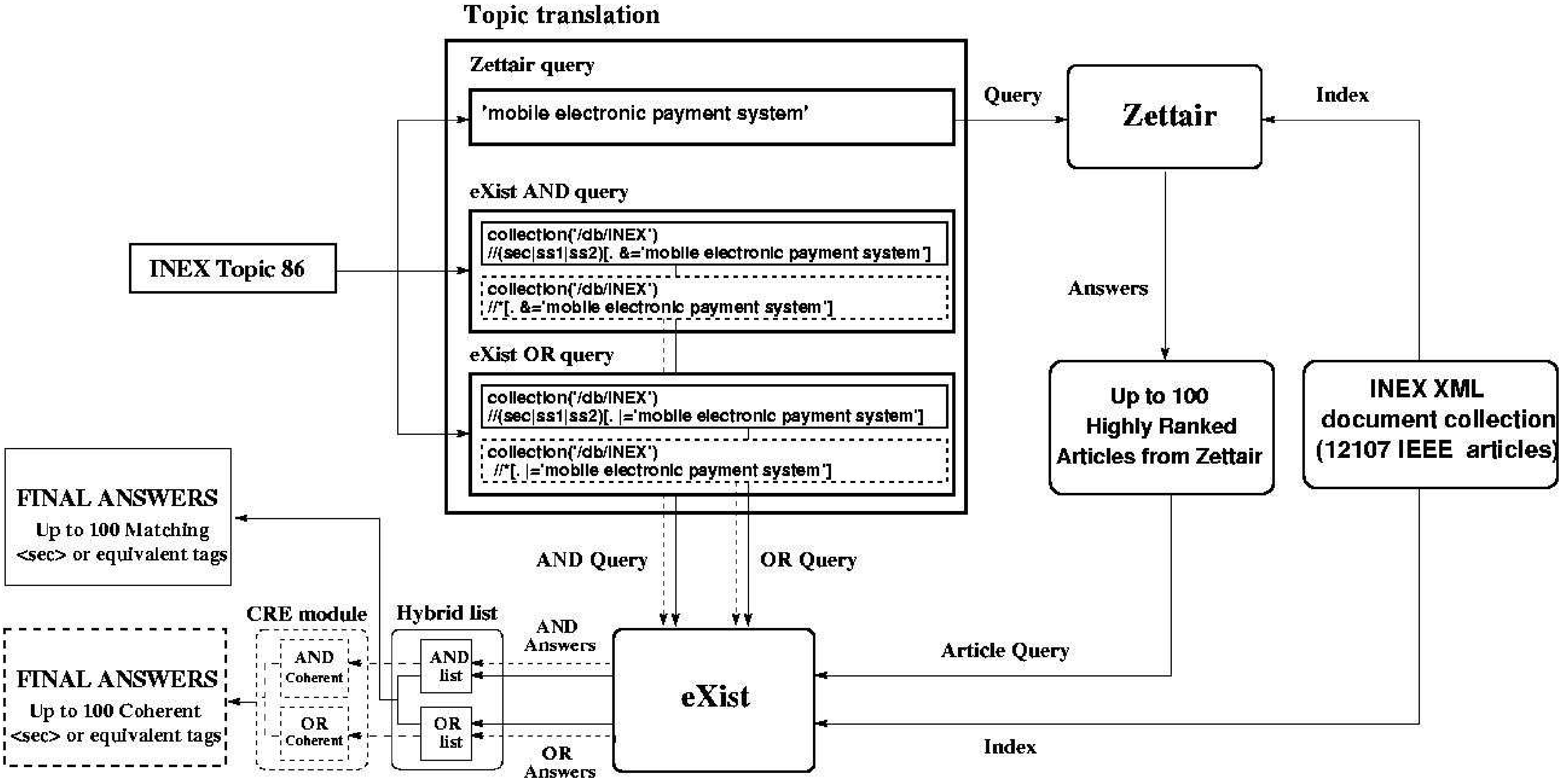
```
collection('/db/INEX')
//*[. &='mobile electronic payment system']
```

eXist OR query

```
collection('/db/INEX')
//(sec|ss1|ss2)[. |='mobile electronic payment system']
```

```
collection('/db/INEX')
//*[. |='mobile electronic payment system']
```

Hybrid XML Retrieval Approach



```
collection('/db/INEX')
//(sec|ss1|ss2)[. &='mobile electronic payment system']
```

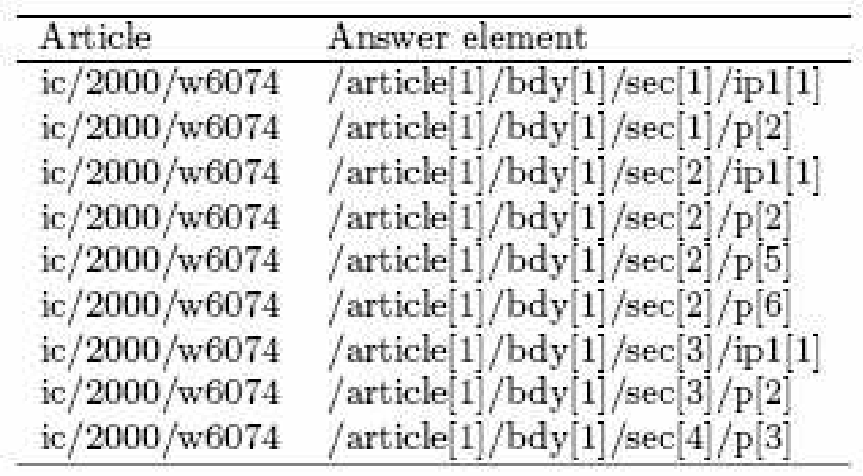
```
collection('/db/INEX')
//*[. &='mobile electronic payment system']
```

```
collection('/db/INEX')
//(sec|ss1|ss2)[. |='mobile electronic payment system']
```

```
collection('/db/INEX')
 //*[. |='mobile electronic payment system']
```

Ranking the Native XML Database Output

- For a particular article in the final answer list, a *Coherent Retrieval Element (CRE)* is an element that contains *at least* two matching elements, or *at least* two other Coherent Retrieval Elements, or a combination of a matching element and a Coherent Retrieval Element.
- In all three cases, the containing elements of a Coherent Retrieval Element should represent either its *different children* or each different child's *descendants*.



Article	Answer element
ic/2000/w6074	/article[1]/bdy[1]/sec[1]/ip1[1]
ic/2000/w6074	/article[1]/bdy[1]/sec[1]/p[2]
ic/2000/w6074	/article[1]/bdy[1]/sec[2]/ip1[1]
ic/2000/w6074	/article[1]/bdy[1]/sec[2]/p[2]
ic/2000/w6074	/article[1]/bdy[1]/sec[2]/p[5]
ic/2000/w6074	/article[1]/bdy[1]/sec[2]/p[6]
ic/2000/w6074	/article[1]/bdy[1]/sec[3]/ip1[1]
ic/2000/w6074	/article[1]/bdy[1]/sec[3]/p[2]
ic/2000/w6074	/article[1]/bdy[1]/sec[4]/p[3]

Table 1: eXist list of CO matching elements

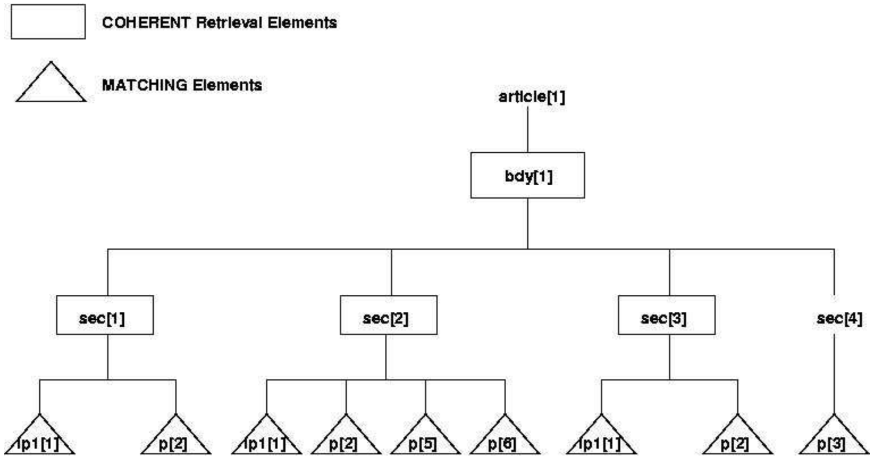
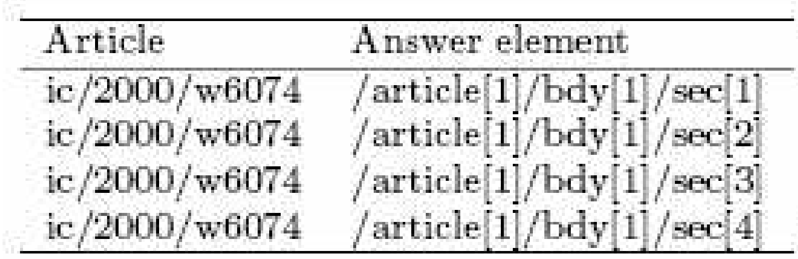


Figure 1: CREs – a tree-view example

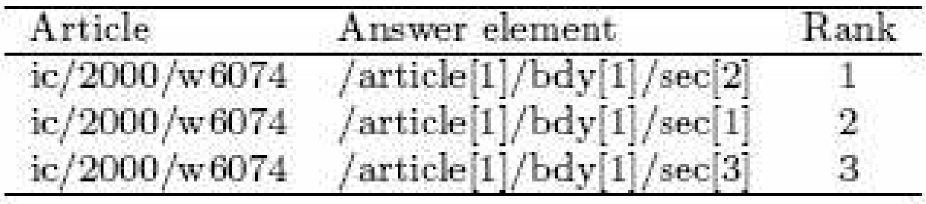
Ranking the Native XML Database Output…

- The following XML-specific criteria are used to calculate the ranking values of the CREs (in descending order of importance):
  - The number of times a CRE appears in the absolute path of each matching element in the answer list (the more often it appears, the better);
  - The length of the absolute path of a CRE (the shorter it is, the better);
  - The ordering of the XPath sequence in the absolute path of a CRE (nearer to beginning is better); and
  - Since we are dealing with SCAS retrieval task, only CREs that satisfy the granularity constraints in a CAS topic will be considered as answers.



Article	Answer element
ic/2000/w6074	/article[1]/bdy[1]/sec[1]
ic/2000/w6074	/article[1]/bdy[1]/sec[2]
ic/2000/w6074	/article[1]/bdy[1]/sec[3]
ic/2000/w6074	/article[1]/bdy[1]/sec[4]

Table 2: eXist list of CAS matching elements



Article	Answer element	Rank
ic/2000/w6074	/article[1]/bdy[1]/sec[2]	1
ic/2000/w6074	/article[1]/bdy[1]/sec[1]	2
ic/2000/w6074	/article[1]/bdy[1]/sec[3]	3

Table 3: Ranked list of CREs

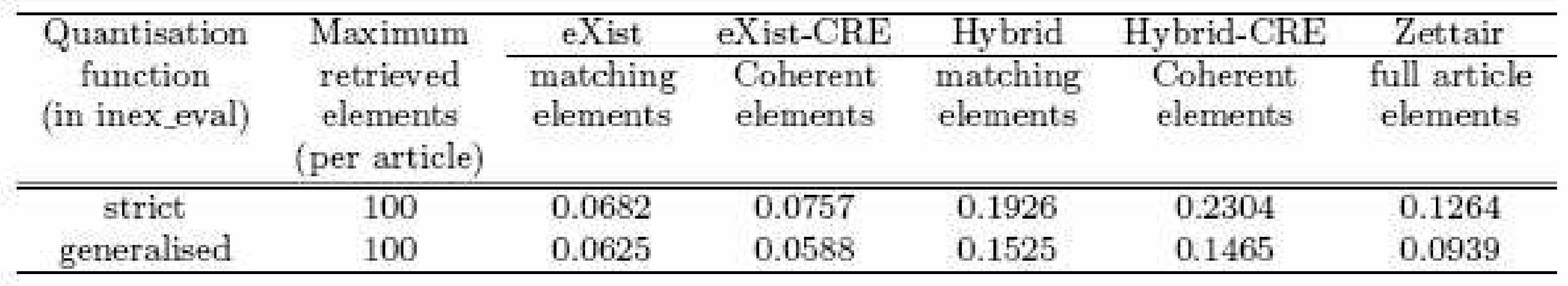
Evaluating the XML Retrieval Effectiveness

- INEX 2003 uses two relevance criteria:
  - *exhaustivity,* the degree to which a document component covers the concepts requested by a topic.
  - *specificity,* the extent to which a document component focuses on the topic.
  - both criteria use a 4-point relevance scale: {0, 1, 2, 3}
- The *inex_eval* evaluation metric uses two quantisation functions:
  - *strict* - evaluates whether an XML retrieval approach is capable of retrieving highly exhaustive and highly specific document components;
  - *generalised* - credits document components according to their degree of relevance.

Example: f_{strict} = **1**, if *exaustivity*=3 and *specificity*=3; **0** otherwise.

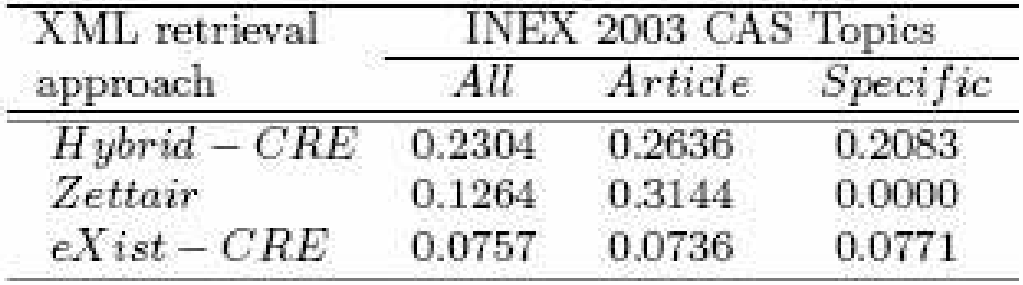
Experiments and Results

- Comparison of XML Retrieval Approaches
  (using different quantisation functions):



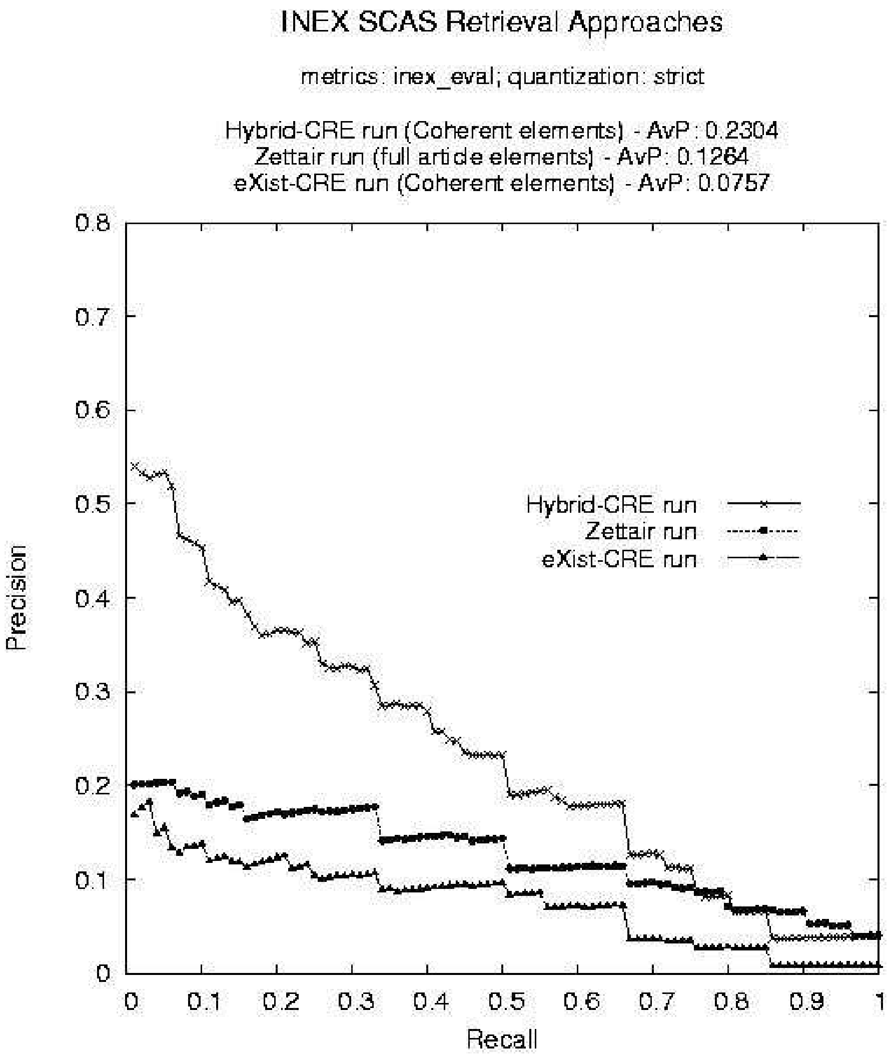
Quantisation function (in inex_eval)	Maximum retrieved elements (per article)	eXist	eXist-CRE	Hybrid	Hybrid-CRE	Zettair
		matching elements	Coherent elements	matching elements	Coherent elements	full article elements
strict	100	0.0682	0.0757	0.1926	0.2304	0.1264
generalised	100	0.0625	0.0588	0.1525	0.1465	0.0939

- Analysis based on different CAS topic categories
  (using strict quantisation function):



XML retrieval approach	INEX 2003 CAS Topics		
	All	*Article*	*Specific*
$Hybrid - CRE$	0.2304	0.2636	0.2083
$Zettair$	0.1264	0.3144	0.0000
$eXist - CRE$	0.0757	0.0736	0.0771

Experiments and Results…



Conclusion and Future Work

- The full-text information retrieval system yields effective retrieval for CAS topics where full article elements are retrieved.
- We observed poor performance for the native XML database for CAS topics where more specific elements within articles (as well as full articles) are retrieved.
- In order to support CAS XML retrieval that combines both topic categories, we have developed and evaluated an XML retrieval system that uses a *hybrid* approach to XML retrieval.
- For ranking the native XML database output we have also developed a retrieval module that identifies and ranks Coherent Retrieval Elements (CREs) with appropriate levels of retrieval granularity.
- Our hybrid XML retrieval system with the CRE retrieval module yields an effective content-and-structure XML retrieval.

Conclusion and Future Work…

- We plan to undertake the following extensions of this work in the future.
  - We aim to investigate the possibility of applying Zettair for ranking CREs coming out of different answer lists.
  - We also aim to investigate the optimal combination of Coherent Retrieval and matching elements in the final answer list, which could be applied to CAS as well as to CO retrieval topics.

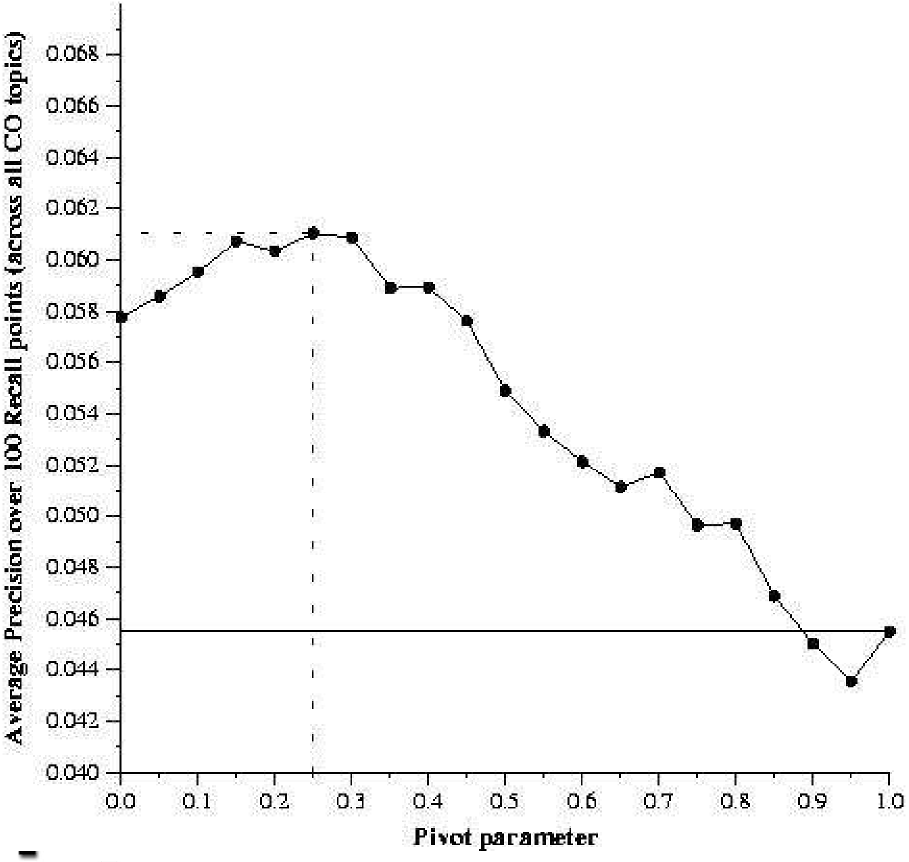
Questions ???

Efficiency considerations

- **INEX XML document collection**
  - 12,107 XML articles of IEEE Computer Society's publications from 12 magazines and 6 transactions for the period of 1995-2002.
  - 494 MB in size.
- **Zettair**
  - the size of the index takes roughly 26% of the total collection size.
  - time taken to index the entire INEX collection on a system with a Pentium4 2.66GHz processor and a 512MB RAM memory running Mandrake Linux 9.1 is around *70 seconds*.
- **eXist**
  - the size of the index is roughly twice as big as the total collection size.
  - time taken to index the entire INEX collection on a system with a Pentium 4 2.6GHz processor and a 512MB RAM memory running Mandrake Linux 9.1 is around 2050 seconds.

Pivoted Cosine Normalization with Zettair…

- Training pivoted cosine normalisation with Zettair on the INEX 2002 test collection. Terms appearing in <Title> part of INEX topics were used to construct the Zettair queries. The AP values are calculated by using strict quantisation function in the *inex_eval* evaluation metric.


Average Precision over 100 Recall points (across all CO topics)
Pivot parameter