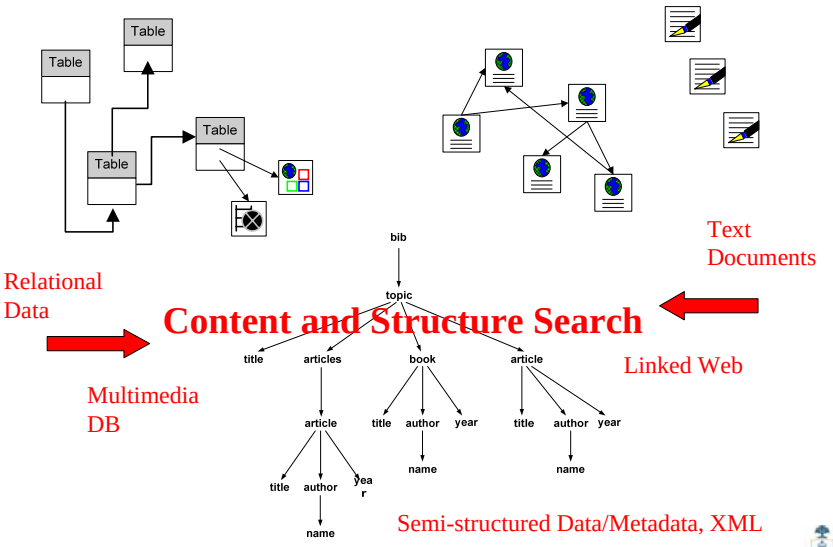
Flexible XML Retrieval using Summaries

Mariano P. Consens
University of Toronto

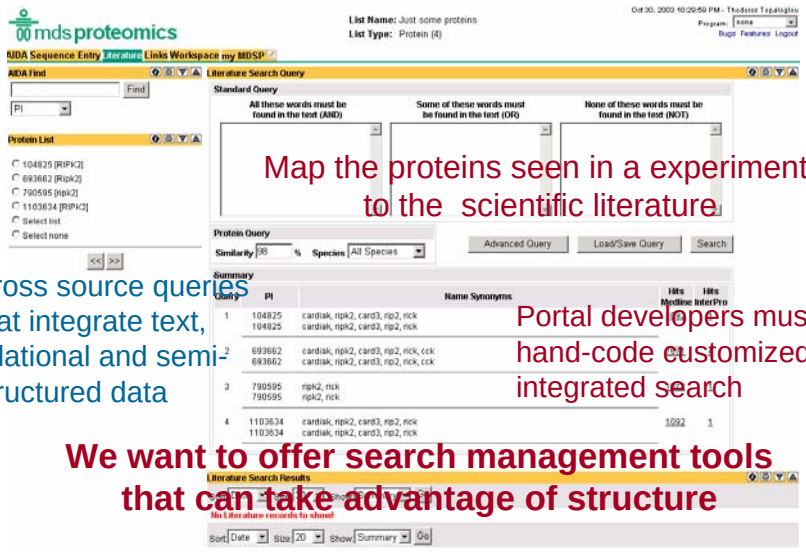
Outline

- Motivation
- What are Summaries
- Using Summaries in XML Retrieval
- Assessments and Performance
- Conclusions

Search/Rank any Data



Scenario: Proteomics Portal



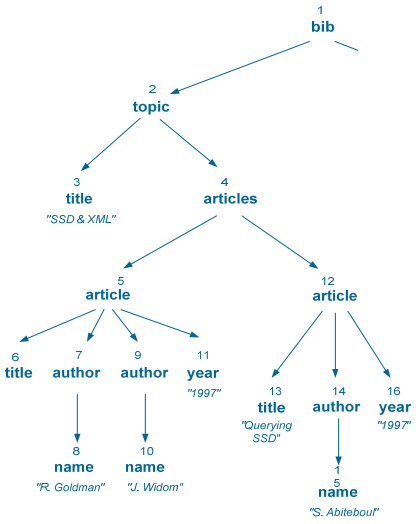
Outline

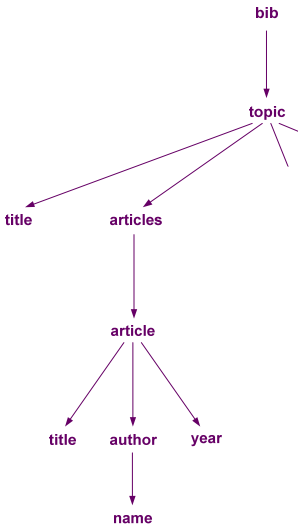
- Motivation
- What are Summaries
- Using Summaries in XML Retrieval
- Assessments and Performance
- Conclusions

Summaries for XML

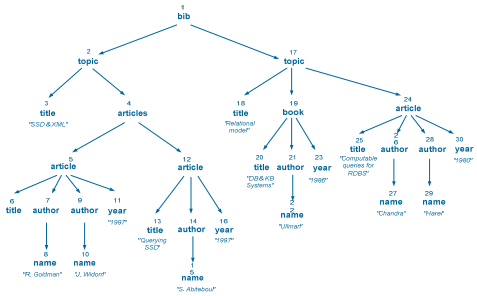
- Structural summaries are data structures that group together elements that cannot be distinguished w.r.t. some tree pattern (XPath query)
- Summaries locate specific fragments of XML (nodes, paths and subtrees)
- By accessing relevant data directly they help to avoid sequential scans of entire documents during structured query evaluation
- But they can be used to describe XML instances by keeping a synopsis of their structural properties

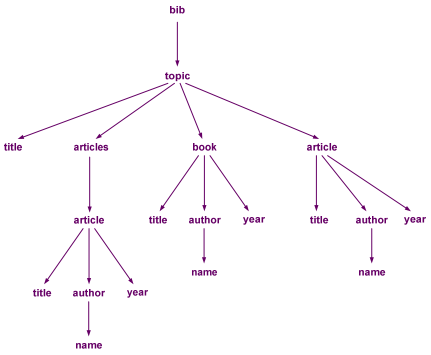
Incoming Path Summaries





Incoming Path Summaries

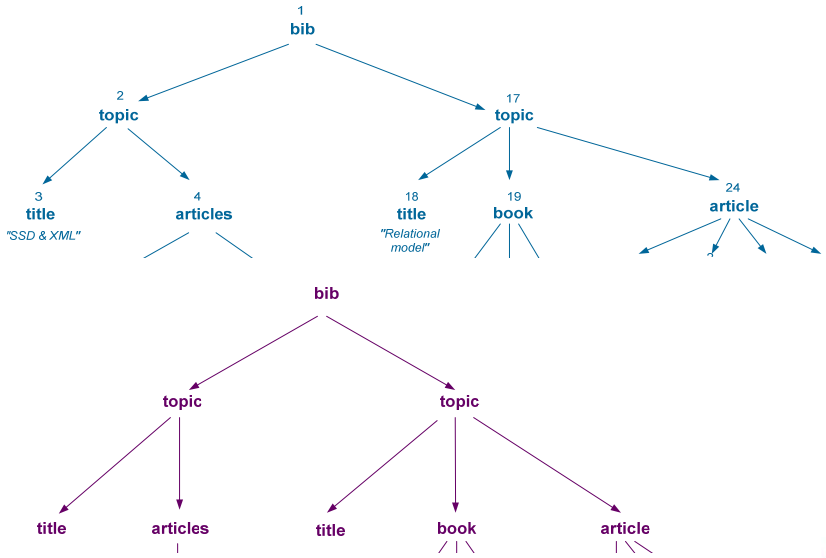




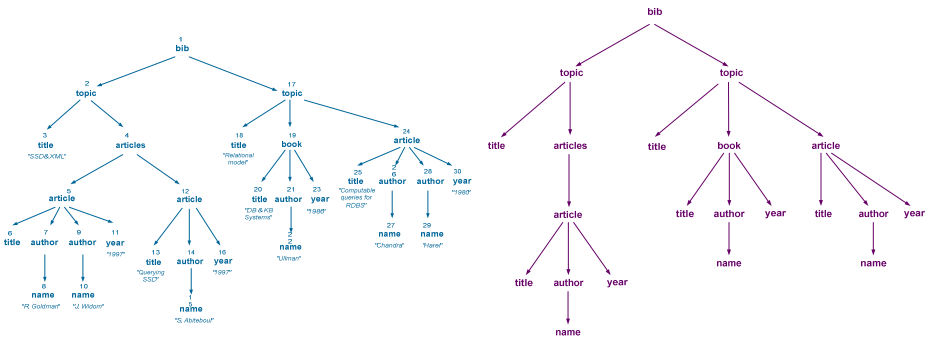
XSummary Framework

- A framework for describing XML structural summaries using XPath
  - Captures many proposals in the literature: region inclusion graphs (RIGs) [CM94], representative objects (ROs)[NUWC97], dataguides [GW97], reversed dataguides [LS00], 1-index, 2-index and T-index [MS99], ToXin [RM01], A(k)-index [KSBG02], F&B-Index and F+B-Index [KBNK02], HOPI [STW04], etc.
  - Provides a uniform approach for defining new classes of summaries
- Summaries are graphs whose nodes are identified by a summary identifier (SID)
- SIDs are like tags, but they are much more descriptive of the structure around them

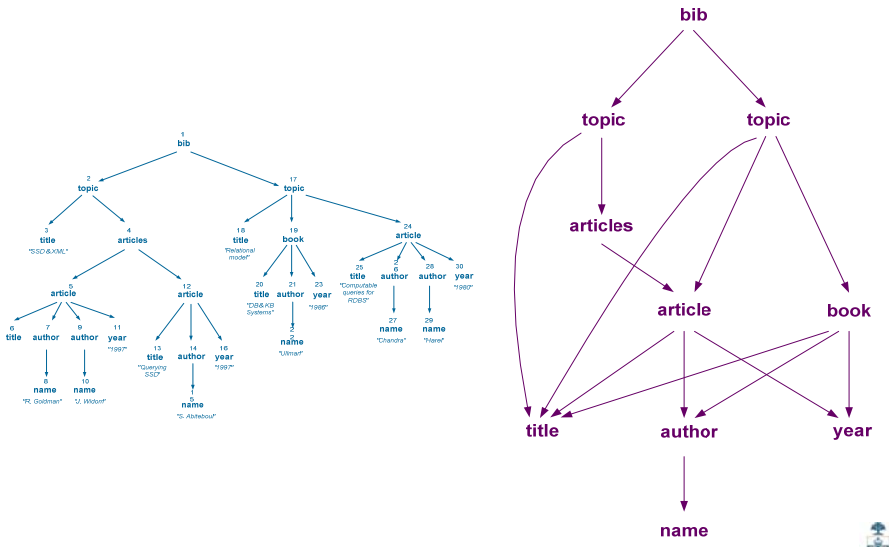
Incoming-Outgoing Summaries



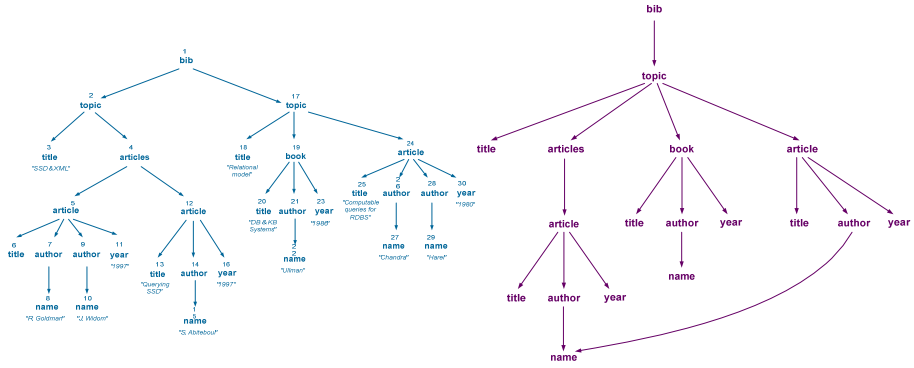
Incoming-Outgoing Summaries



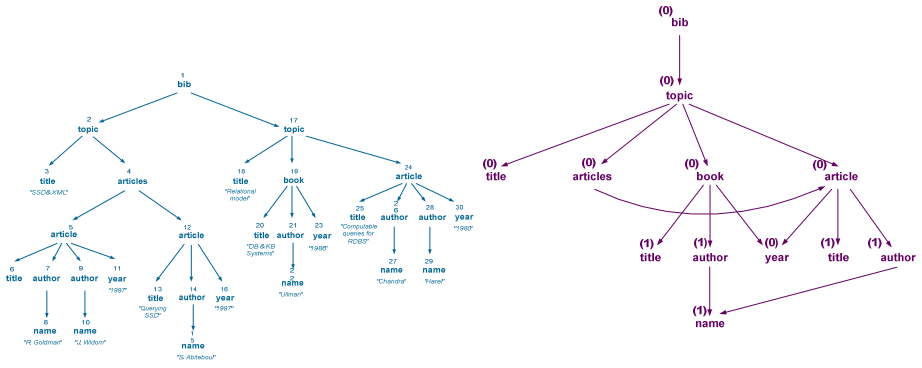
Outgoing Summaries



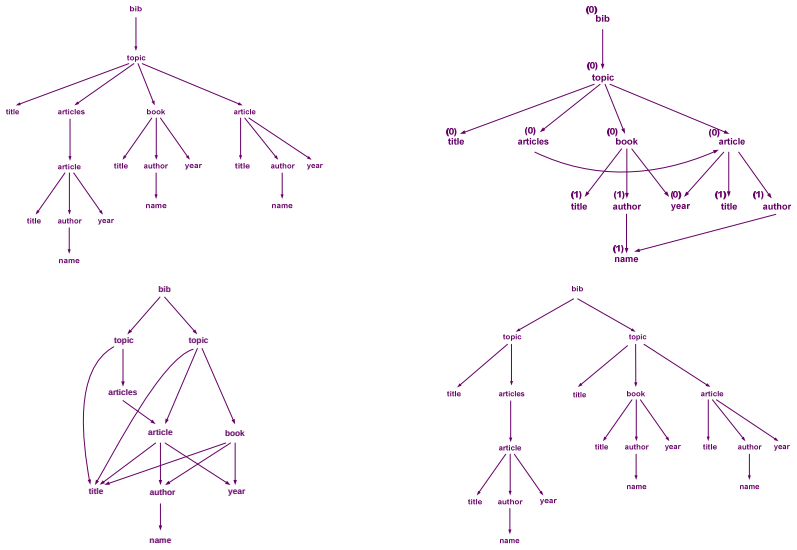
2-incoming Summaries



D(k) Summaries



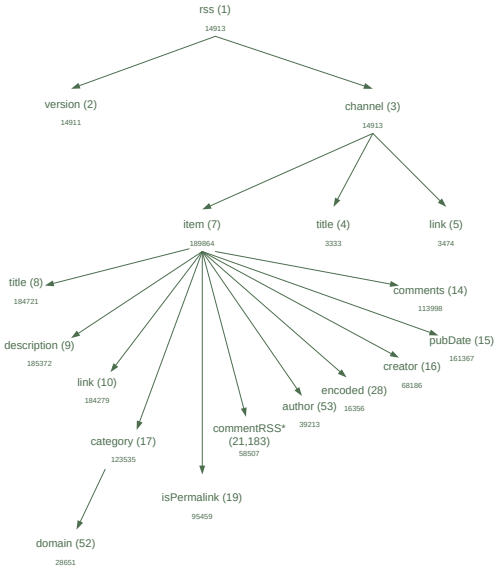
Summaries: plenty to choose



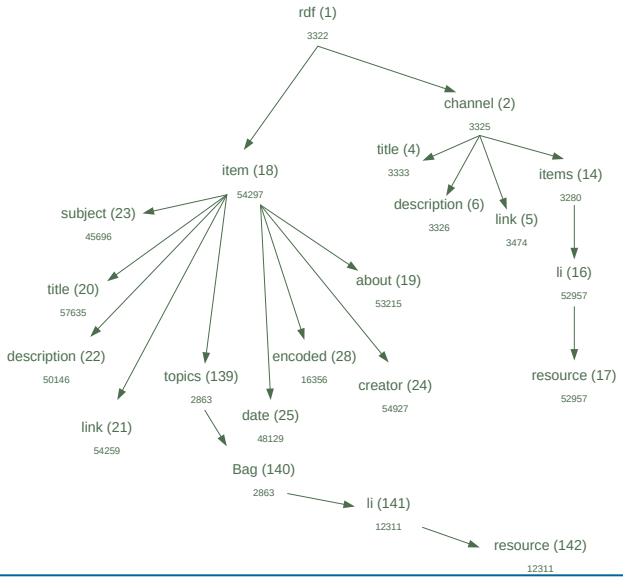
Summaries vs. Schemas

- DTDs and XML Schemas are used for validating instances
- Both are schemas in the database sense, and thus describe classes of documents and constrain their structure and contents
- However, they provide only a limited description of the instances that satisfy them and have no mechanism to locate specific instance fragments.
- In contrast, summaries are constructed for a particular instance and consequently provide a tighter description of the data.
- Summaries can be used in broader classes of applications, even when DTDs and XML Schemas are not present or very lax (semi-structured)

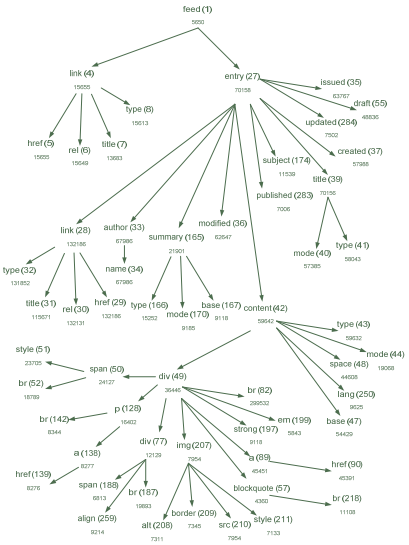
RSS Feed (extract)



RDF Feed (extract)



Atom Feed (extract)



XML Files	Sample1	Sample2
RSS	14913	9282
ATOM	5650	3093
RDF	3322	1875
Total	23885	14250

SIDs	Sample1	Sample2
RSS	3609	2905
ATOM	5724	4212
RDF	462	382
Total	9795	7499

Outline

- Motivation
- What are Summaries
- Using Summaries in XML Retrieval
- Assessments and Performance
- Conclusions

The TReX Search Engine

- Java-based system developed within Eclipse
  - Builds on top of open source components for storage (BerkeleyDB JE), tokenization (Lucene)
  - Extends standard XML parsing API (Stax) to incorporate tokenization and languages (TStax)
  - Builds upon the use of summaries in ToXop for structured XML query optimization and evaluation [Barta, Consens, Mendelzon VLDB'05]
- Participants:
  - S. Ali, X. Gu, Y. Kanza, F. Rizzolo, R. Stasiu

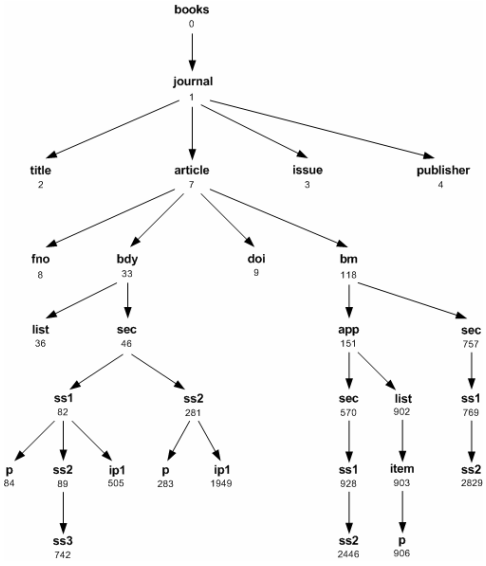
NEXI Queries from INEX 2005

Topic id	NEXI Query
202	//article[about(., ontologies)]//sec[about(., ontologies case study)]
203	//sec[about(., code signing verification)]
219	//sec[about(., learning object granularity)]
222	//article[about(., business strategies)]//sec[about(., electronic commerce e-commerce)]
223	//article[about(.//sec, wireless ATM multimedia)]
233	//article[about (.//bdy, synthesizers) and about (.//bdy, music)]
236	//article[about(., machine translation approaches -programming)]
260	//bdy//*[about(., model checking state space explosion)]
270	//article//sec[about(., introduction information retrieval)]
284	//article[about (.//bdy, thread implementation) and about (.//bdy, operating system)]

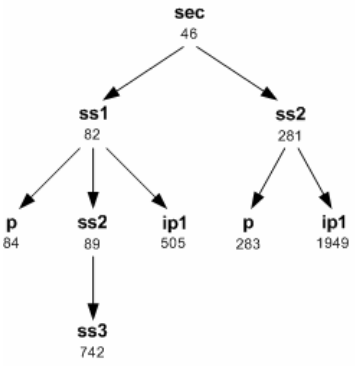
- Target queries are expressions combining structure and content conditions

  //article [about(.//sec, code signing)]

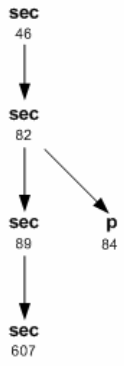
INEX Incoming Summary (extract)



INEX Summaries (fragment/stats)



incoming



alias incoming

Summary	*Incoming*	*Tag*
Number of SIDs with synonyms	7960	148
Number of SIDs without synonyms	11563	185

NEXI to SID Translation

Topic id	SID	Keyword
202	7, 46, 82, 89, 493, 607, 619, 630,761, 1995, 2239	ontologies, case, study
203	7, 46, 82, 89, 493, 607, 619, 630,761, 1995, 2239	code, signing, verification
219	7, 46, 82, 89, 493, 607, 619, 630,761, 1995, 2239	learning, object, granularity
222	7, 46, 82, 89, 493, 607, 619, 630,761, 1995, 2239	business, strategies, electronic, commerce, e-commerce
223	7, 46, 82, 89, 493, 607, 619, 630,761, 1995, 2239	wireless, ATM, multimedia
233	7, 33	synthesizers, music
236	7	machine, translation, approaches
260	7, 33	model, checking, state, space, explosion
270	7, 46, 82, 89, 493, 607, 619, 630,761, 1995, 2239	introduction, information, retrieval
284	7, 33	thread, implementation, operating, system

- A key aspect of TReX is to translate NEXI expressions into (SID,keyword) pairs
- Different summaries require different translations, hence different efficiency and effectiveness

Evaluation in TReX

- TReX uses two methods for computing NEXI queries
- Exhaustive Algorithm (EA): queries are computed in a one-pass merge of two types of indexes
  - Inverted file (can locate keyword positions in elements)
  - Summary-based structural indexes (can answer XPath queries).
- Threshold Algorithm (TA): the top-k answers to a query are computed from relevance-ordered posting lists (which are pre-computed using EA)

Ranking in TReX

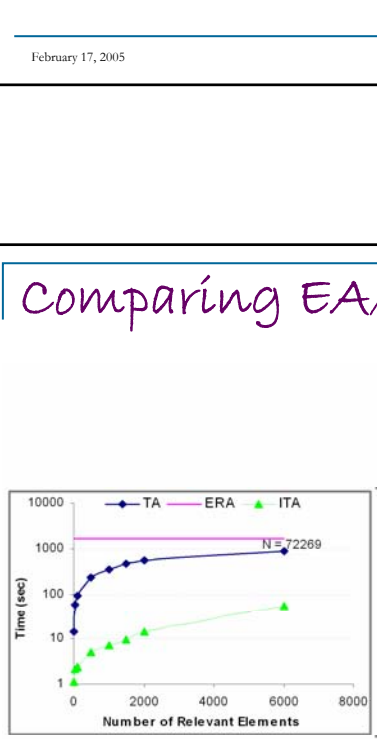
- EA can use different scoring functions
- The results presented here use the same variant of BM25 as TopX [Theobald, Schenkel, Weikum 2005], but with Tag replaced by SID (changes according to the summary used, coincides for Label/Tag)

Outline

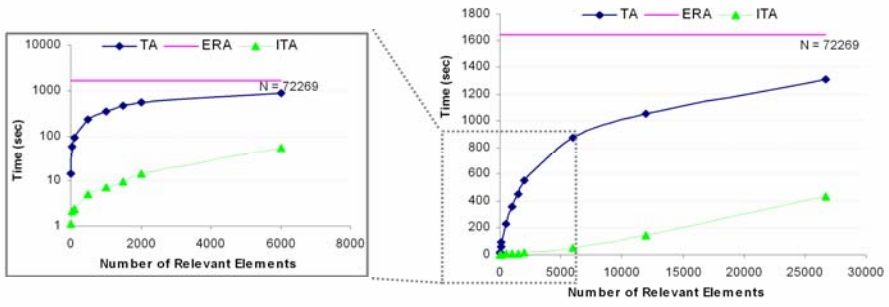
- Motivation
- What are Summaries
- Using Summaries in XML Retrieval
- Assessments and Performance
- Conclusions

A Glimpse of Experimental Results

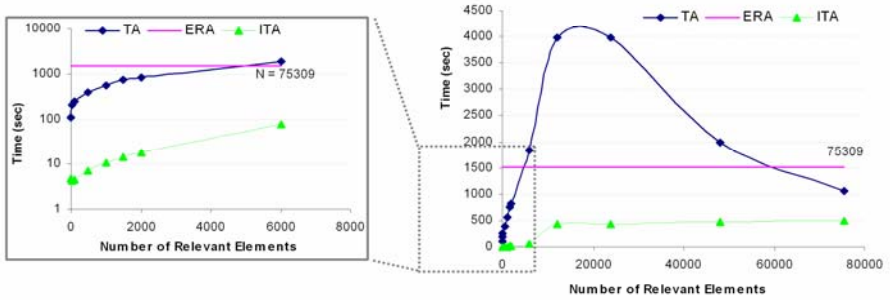
- Using the INEX 2005 collection and assessments
- Compare efficiency
- Compare effectiveness
- Keep in mind that changing summaries can influence both!



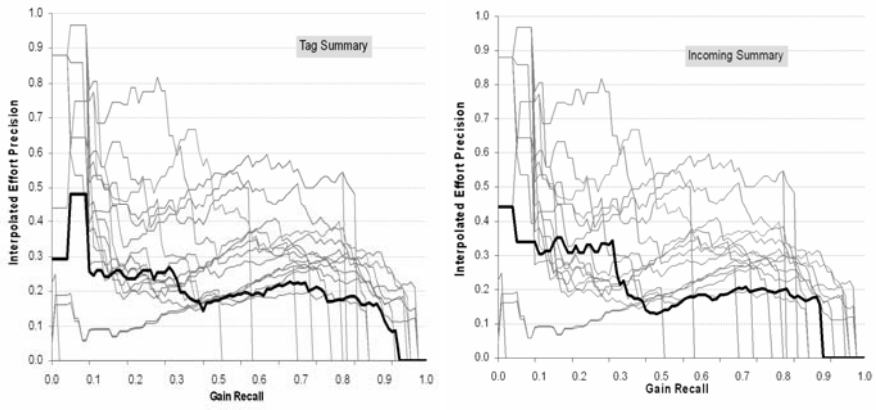
Comparing EA/TA Efficiency



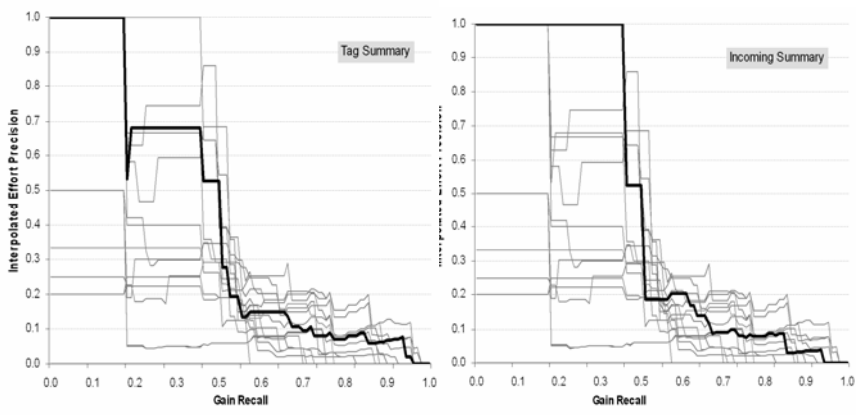
Comparing EA/TA Efficiency (II)



Comparing Summary Effectiveness



Comparing Summary Effectiveness



Outline

- Motivation
- What are Summaries
- Using Summaries in XML Retrieval
- Assessments and Performance
- Conclusions

Conclusions

- We need a better understanding of how structure impacts XML Retrieval
- Our approach is to build search management tools (TReX) with built-in flexibility to exploit retrieval options
- Key: achieve flexibility while retaining competitive performance
- Flexible use of summaries are crucial to achieving TReX objectives

Thanks

Mariano P. Consens
University of Toronto