Processing XML Full-Text Data

Sebastian Gath

Explorative Analysis and Visualization of Large Information Spaces

Universität Konstanz

1 Background

- XML documents serve variable contents with total flexible structure
- for efficient processing XML databases are essential
- special class of XML documents containing a lot of textual content (e.g. Wikipedia)

2 Research Questions

Observations

- XML data is sorted in different ways, e.g. native databases
- XPath/XQuery define various querying opportunities with structural and content based queries
- versatile field of optimizations:
  grammar transformations, index support in particular for documents with textual contents

Questions

- how to use the optimization potential of indexes for content based queries?
- how to design indexes with respect to the usecases of XPath?
- how to address further requirements, like fuzzy search?
- how to deal with large result sets and limited display space?

3 BaseX [1]

W3C XQuery Full-Text 1.0 WorkingDraft

- combined querying of text-content and structural parts
- powerful grammar with conjunctions, disjunctions, negations, distances, scopes, windows and wildcards

Full-Text Details of BaseX

- Trie basex index structure for text nodes
- support of exact, range and wildcard search
- disk and main memory based storage
- optional compression mechanisms, bulk loading facility
- special index structure for fuzzy search (library context)

XML Document Fragment

```
<page>
   <title>Fuzzy string searching</title>
   <text>Fuzzy string search is the name that is used for a category of techniques for string
searching/finding strings that approximately match some given pattern string.
It may also be known as approximate or inexact matching.  ...
   </text>
</page>
```

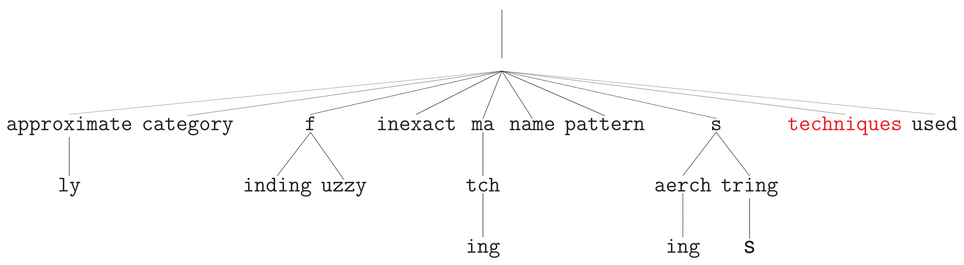
Example Query

```
 //* [text() ftcontains "techniques"]
```

Result

```
   <text>Fuzzy string search is the name that is used for a category of techniques for string
searching/finding strings that approximately match some given pattern string.
It may also be known as approximate or inexact matching.  ...
   </text>
```

Full-Text Index Structure (Trie)



Full-Text Index Structure
Arraybased Storage

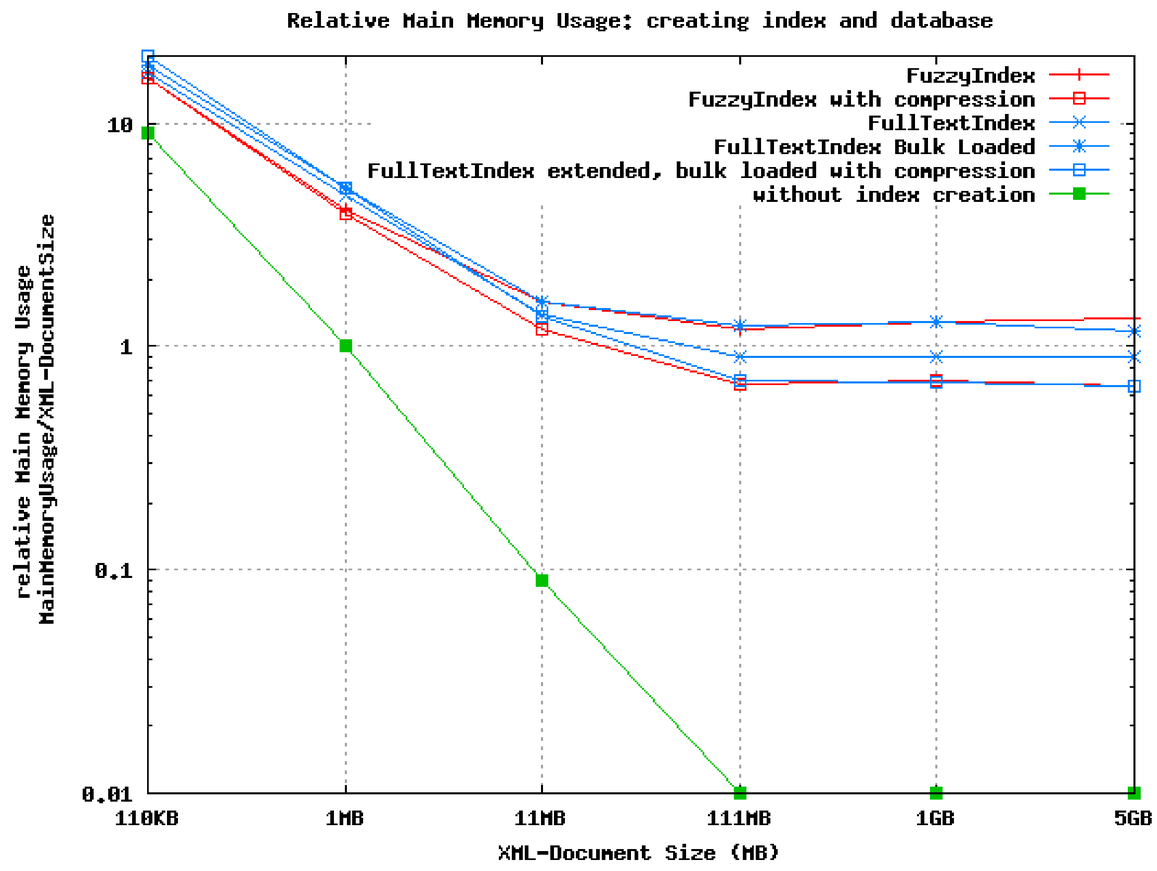
ID	Token	Next Nodes	Pre	Pos
1	approximate	2, l	5	119
2	ly		5	83
3	category		5	30
4	f	6, i, 5, u		
5	uzzy		3, 5	0, 0
6	inding		5	67
7	inexact		5	131
8	match	9, i	5	97
9	ing		5	139
10	name		5	20
11	pattern		5	103
12	s	15, e, 14, t		
13	ing		3, 5	13, 57
14	tring	16, s	3, 5, 5, 5	6, 6, 50, 111
15	earch	13, i	5	13
16	s		5	57
17	techniques		5	39
18	used		5	25
0	1,a, 3,c, 4,f, 7,i, 8,m, 10,n, 11,p, 12,s, 17,t, 18,u			

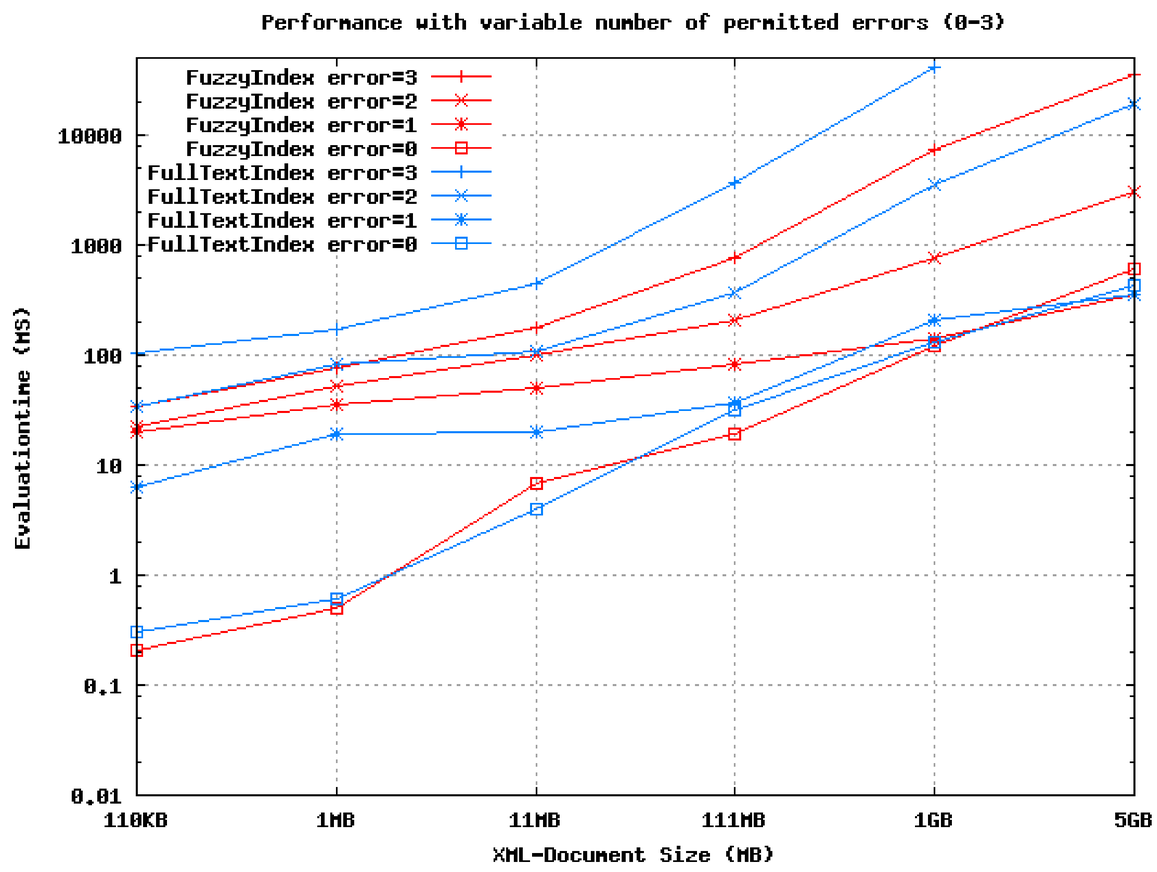
Fuzzy Index Structure
Token Length and Token Index

Tokenlength	4	5	6	7	8	9	10	11	13
ID	0	2	4	6	10	12	13	14	15

ID	Token	Pre	Pos
0	name	5	20
1	used	5	25
2	fuzzy	3, 5	0, 0
3	match	5	97
4	search	5	13
5	string	3, 5, 5, 5	6, 6, 50, 111
6	finding	5	67
7	inexact	5	131
8	pattern	5	103
9	strings	5	75
10	category	5	30
11	matching	5	139
12	searching	3, 5	13, 57
13	techniques	5	39
14	approximate	5	119
15	approximately	5	83

4 Results





References

[1] Christian Grün, Alexander Holupirek, and Marc H. Scholl. Visually Exploring and Querying XML with BaseX. In *12. GI-Fachtagung für Datenbanksysteme in Business, Technologie und Web (BTW 2007)*, Aachen, Germany, March 2007. (Demo).