NTNU – Trondheim
Norwegian University of
Science and Technology

IAD
INFORMATION ACCESS DISRUPTIONS

Indexing and Querying of Overlapping Structures

Faegheh Hasibi

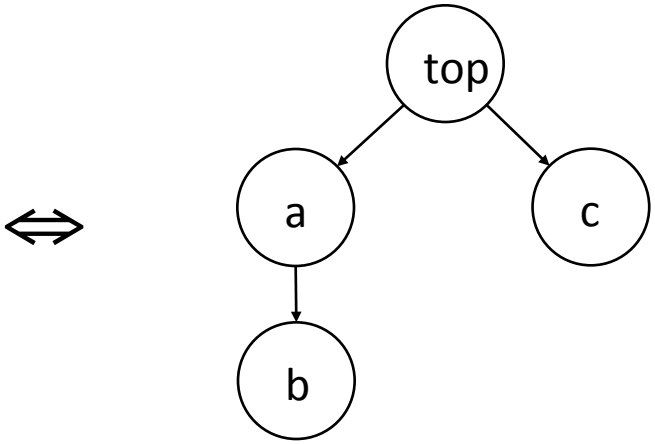
Adv.: *Svein Erik Bratsberg and Øystein Torbjørnsen*

Department of Computer and Information Science

Norwegian University of Science and Technology

What we have

```
<top>
  <a>
    <b  />
  </a>
  </c>
</top>
```



- Indexing methods: Dewey, PrePost,…
- Query processing: Twig Pattern Matching
- Query language: XPath, XQuery, …

But …

In real life information is not purely hierarchical.

- Example: Annotating structure of documents
  - Physical view: page/ line
  - Logical view: section/ paragraph/ sentence
- There are several solutions for **encoding overlaps in XML**
  - E.g.: TEI guidelines for non-hierarchical structures

```
<page>
    <sentence>
        <line>Example</line>
        Overlapping
</page>
<page>
        Section
    </sentence>
</page>
```

But …

In real life information is not purely hierarchical.

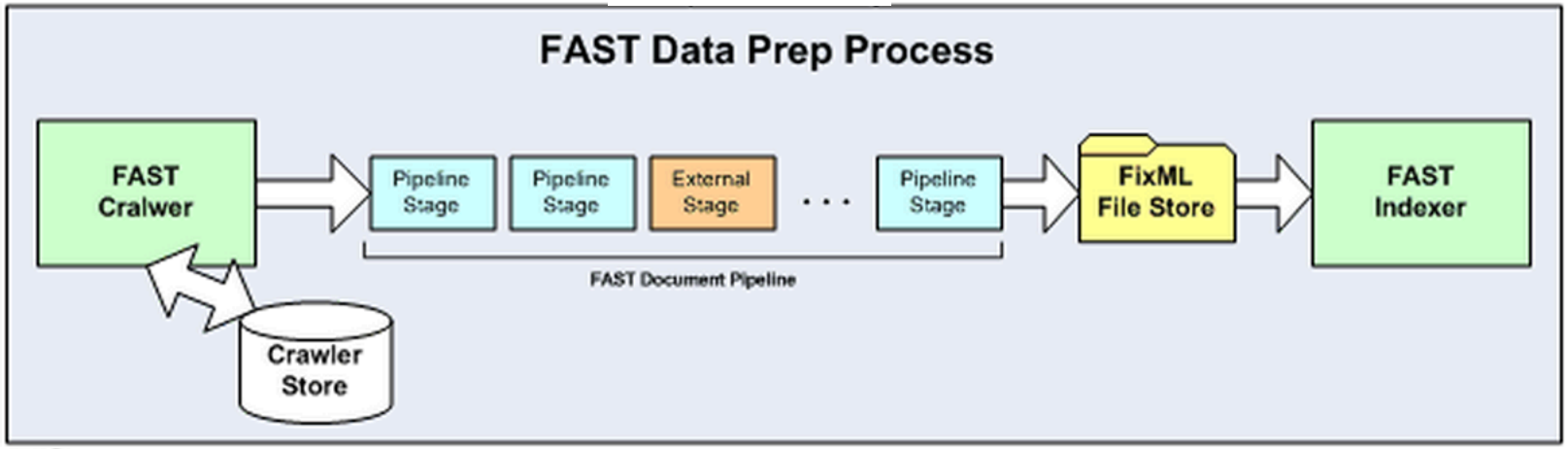
- Example: Annotating structure of documents
  - Physical view: page/ line
  - Logical view: section/ paragraph/ sentence
- There are several solutions for **encoding overlaps in XML**
  - E.g.: TEI guidelines for non-hierarchical structures

```
<page>
    <sentence n="1">
        <line>Example</line>
        Overlapping
    </sentence>
</page>
<page>
    <sentence n="2">Section</sentence>
</page>
```

Why Overlapping Matters?

Overlaps in **"Content Analysis" process of search engines**

- Microsoft FAST search server (ESP and FSIS):
  - Several components for analyzing input data
  - Each component adds annotations to input data
  - ➢ Annotations created by one component may overlap others



Research Questions

Main Research Question:

"How to efficiently handle overlaps in large-scale search engine?"

- **RQ1. Data model**
  Which data model can represent overlapping structures?
  How to map annotations to this data model?

- **RQ2: Indexing approach**
  How to index non-hierarchical structures?

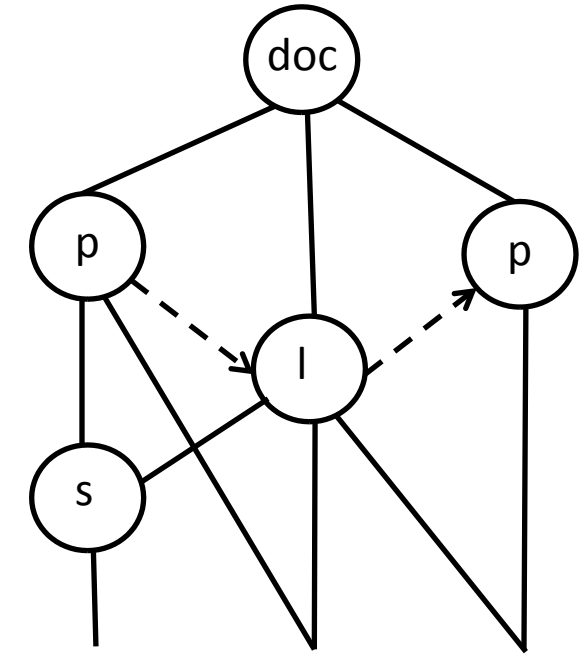
- **RQ3: Query processing approaches**
  - How to process queries of overlapping structures in search engines?
  - processing full text and structural queries

- **RQ4: Query language**
  - How users can ask queries on overlapping structures

Overlapping Data Model

Approach:
Extending GODDAG to a tree-like data model

- Each node can have multiple parents
- Inheritance is represented by solid arcs
- Overlapping is represented by dashed arcs
- **P - - -> S** means:
  1. Node P and node S are overlapping
  2. Node P precedes node S



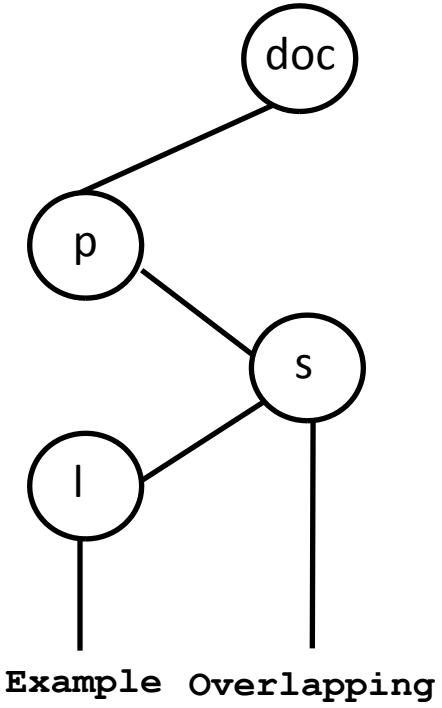
Example Overlapping section

Parsing Documents

Given adjacent opening tag <p> … <s>:

- Node <p> either contains or overlaps <s>
- Assume <p> contains <s>
- Parse the document like a tree

```
<doc>
  <p>
    <s>
      <l>Example</l>
      Overlapping
  </p>
  <p>
      Section
    </s>
  </p>
</doc>
```

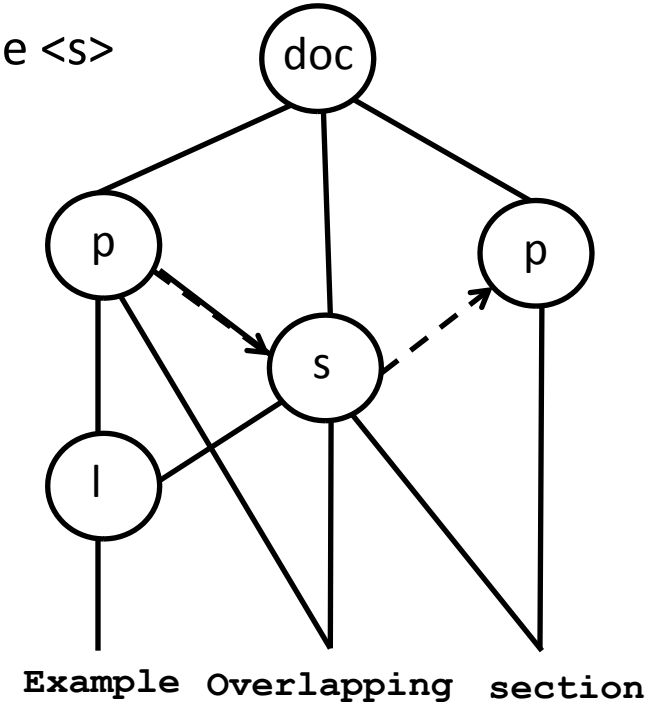


Parsing Documents

If node <p> and <s> are overlapping:

1. Change solid arc to dashed arc
2. Add a solid arc from parent of node <p> to node <s>
3. Add solid arcs from node <p> to children of node <s>

```
<doc>
  <p>
    <s>
      <l>Example</l>
      Overlapping
  </p>
  <p>
      Section
    </s>
  </p>
</doc>
```



Indexing Method

Approach: Extending XML Pre-post indexing method

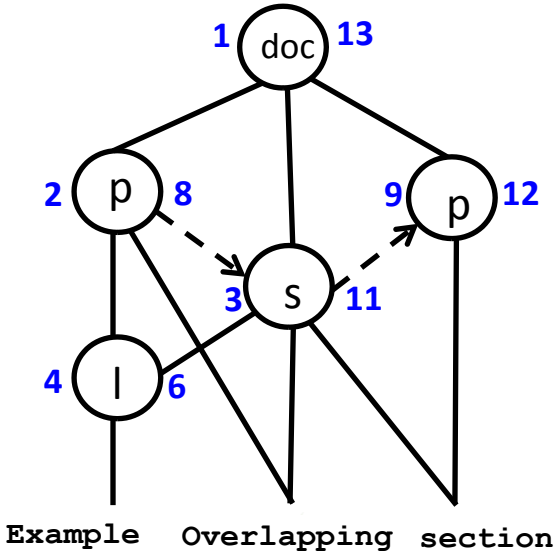
- Element index: (**docNo, start, end, parent(s)**)
  - **Start, end**: starting and end position of element
  - **Parent(s)**: start position of parent(s) node
- Text index: : (**docNo, position, parent(s)**)

(Element index)

<p>	(1, 2, 8, [1]) (1, 9, 12, [1]) ...
<s>	(1, 3, 11, [1]) ...
<l>	(1, 4, 6, [2, 3]) ...

(Text index)

Example	(1, 5, [4]) ...
Overlapping	(1, 7, [2, 3]) ...
Section	(1, 10, [3, 9]) ...



Indexing Method

The index can efficiently support:

- Overlapping:
  - <a> precedes and overlaps <b> IFF **a.start < b.start** AND **a.end < b.end**
- Ancestor-descendant:
  - <a> is ancestor of <b> IFF **a.start < b.start** AND **b.end < a.end**
- Parent-child:
  - <a> is parent of <b> IFF **b.parent = a.start**

Issue:

Not easy to maintain under dynamic updates

Challenges

1. The lack of **proper dataset**
   - A corpus collection that satisfies these properties:
     i. Real overlapping data
        - Most of overlapping datasets are artificially created for special purposes
     ii. Data with deep hierarchical structure
     iii. Large enough for indexing purpose
2. No appropriate framework
   - Existing open-source search engines can retrieve hierarchical structures
3. The lack of proper **baseline system**
   - No standard test set
   - Current approaches mostly use no indexing methods for handling overlaps

References

- T. Consortium, L. Burnard, and S. Bauman. “TEI P5: Guidelines for electronic text encoding and interchange”. TEI Consortium, 2012.
- F. ESP. Powering enterprise search with fast esp , 2008.
- C. Sperberg-McQueen and C. Huitfeldt. Goddag: “A data structure for overlapping hierarchies”. In Digital Documents: Systems and Principles, volume 2023 of Lecture Notes in Computer Science, pages 606–630, 2004.
- C. Zhang, J. Naughton, D. DeWitt, Q. Luo, and G. Lohman. “On supporting containment queries in relational database management systems”. In Proceedings of the 2001 ACM SIGMOD, pages 425–436, New York, USA, 2001.
- Grust, Torsten. "Accelerating XPath location steps." In Proceedings of the 2002 ACM SIGMOD, pp. 109-120. ACM, 2002.
- F. J. Burkowski. Retrieval activities in a database consisting of heterogeneous collections of structured text. In Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR ’92, pages 112–125, New York, NY, USA, 1992. ACM.