Flexible structural constraints in XQuery Full-Text

Emanuele Panzeri
University of Milano Bicocca
panzeri@disco.unimib.it

Motivation

BaseX for : **XML Retrieval**

- Focused Retrieval of XML fragments
- Information Retrieval techniques with highly-structured documents

Query Language for XML Retrieval: **XQuery Full-Text**

- Full-Text searches within XML elements
- Fragment scoring only on content matching, no flexibility for structural constraints
- **User is required to know in advance the heterogeneous document structure in the collections**

A flexible extension of XQuery Full-Text

Support user approximate query specification with respect to document structure:

- Vagueness in descending axis: **Below** axis
- Vagueness in "nearby" nodes: **Near** axis

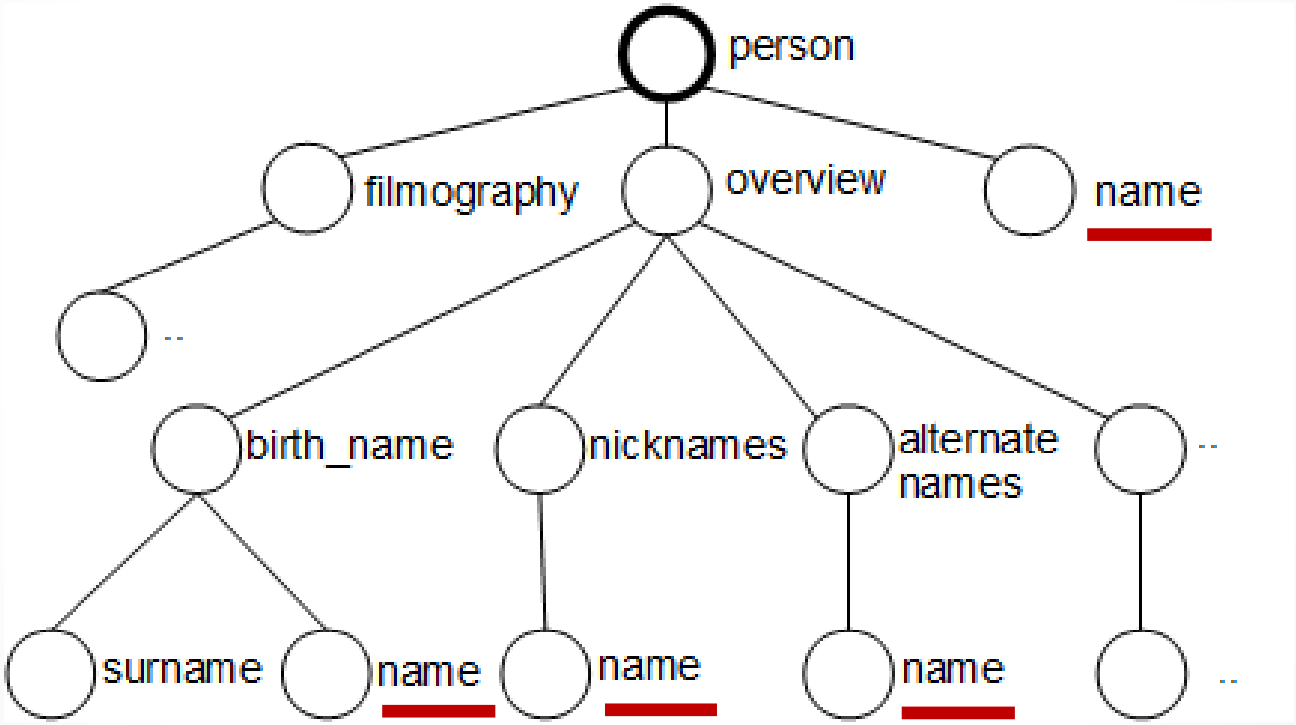
Each constraint provides a computation of a **path relevance score**

for each matched node

[1] E. Panzeri, G. Pasi, An Approach to Define Flexible Structural Constraints in XQuery, AMT 2012, Macau

XML Querying: an example

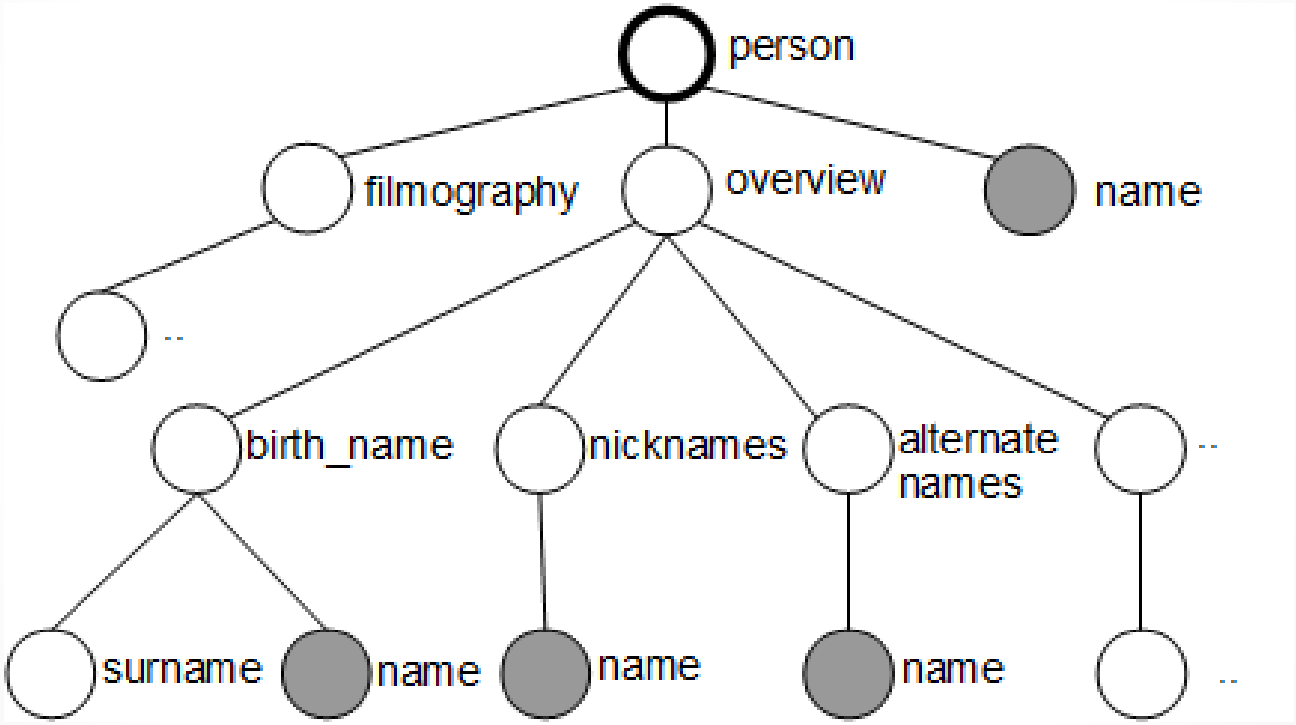
Flexibility offered by XQuery (Full-Text)



Structure reflects the semantics of the data

XML Querying: an example

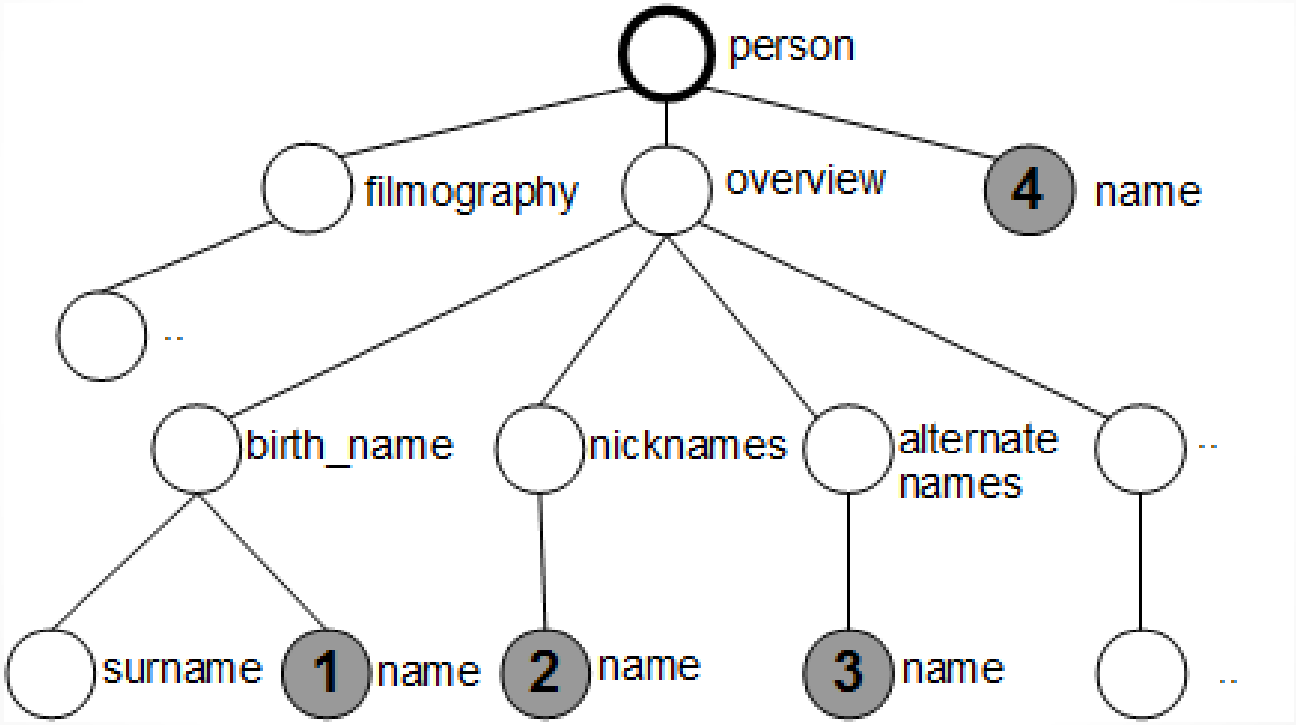
Flexibility offered by XQuery (Full-Text)



XPath query: **person /descendant:: name**

XML Querying: an example

Flexibility offered by XQuery (Full-Text)

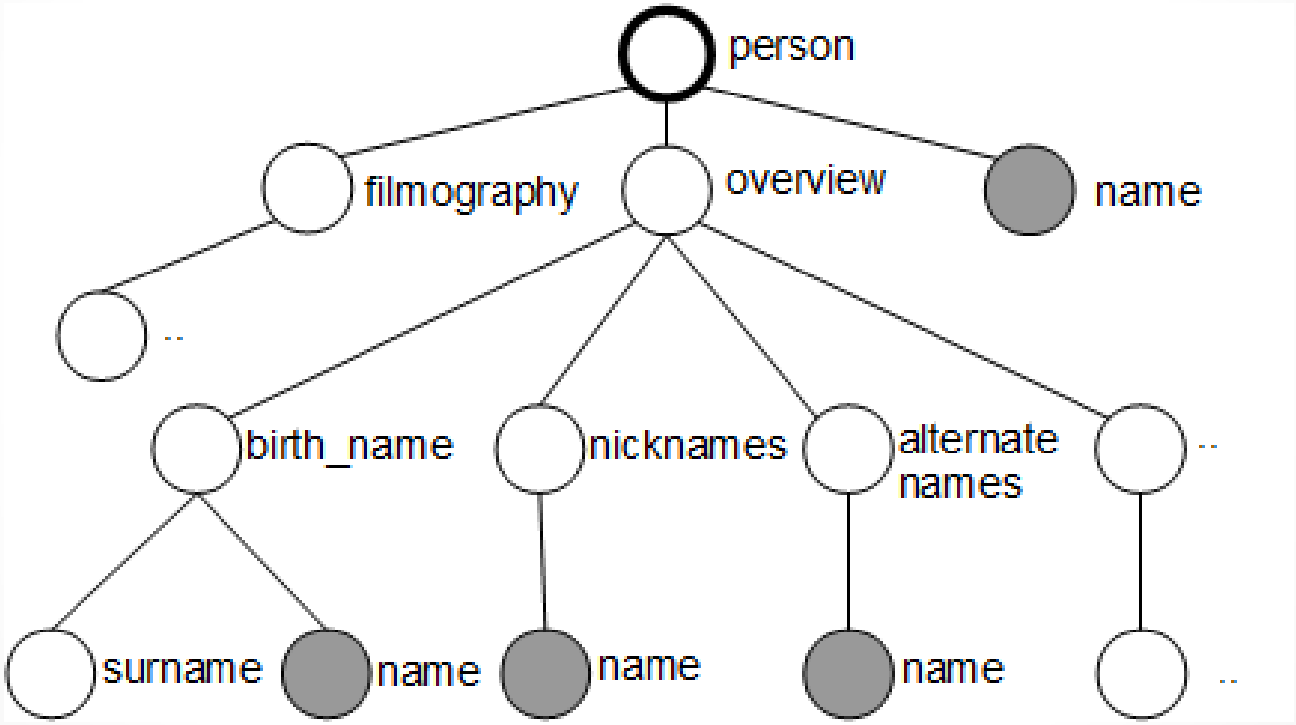


XPath ranking order: **none**

Elements are retrieved as found in the document source

The flexible extension

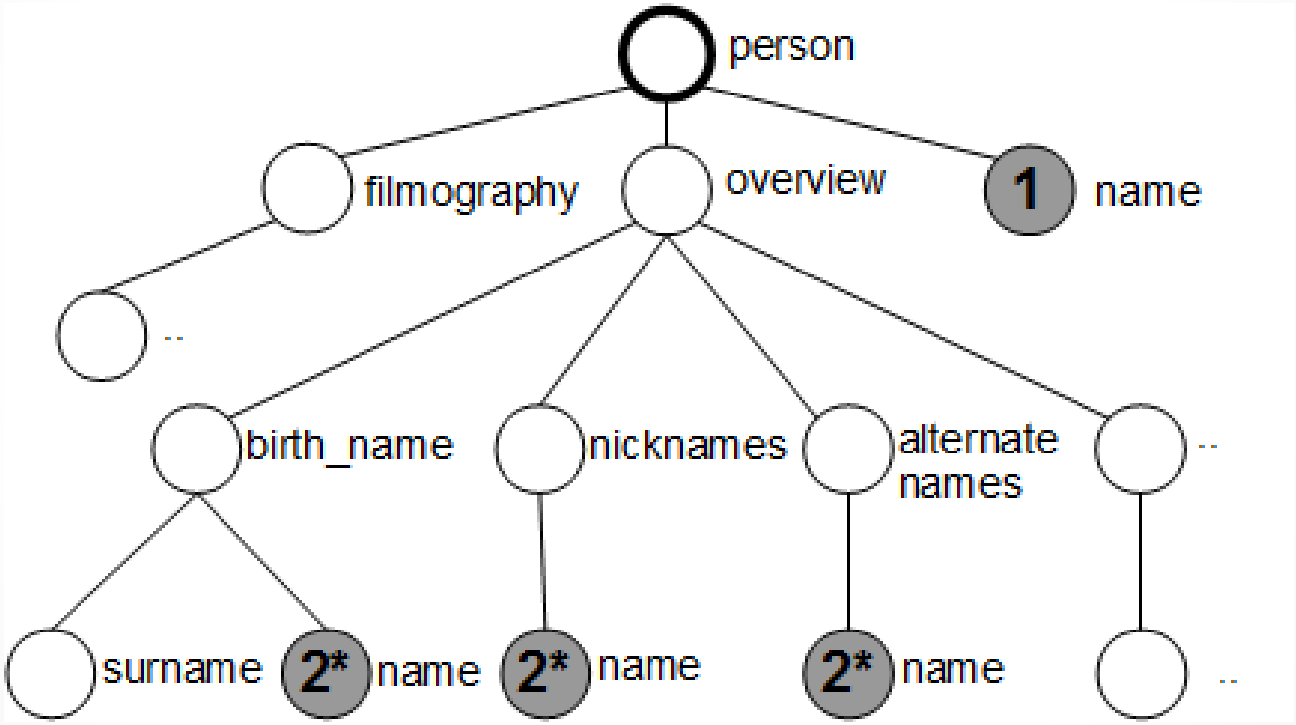
Adding flexible structure matching with the **Below** constraint



Flexible XPath query: **person /below:: name**

The flexible extension

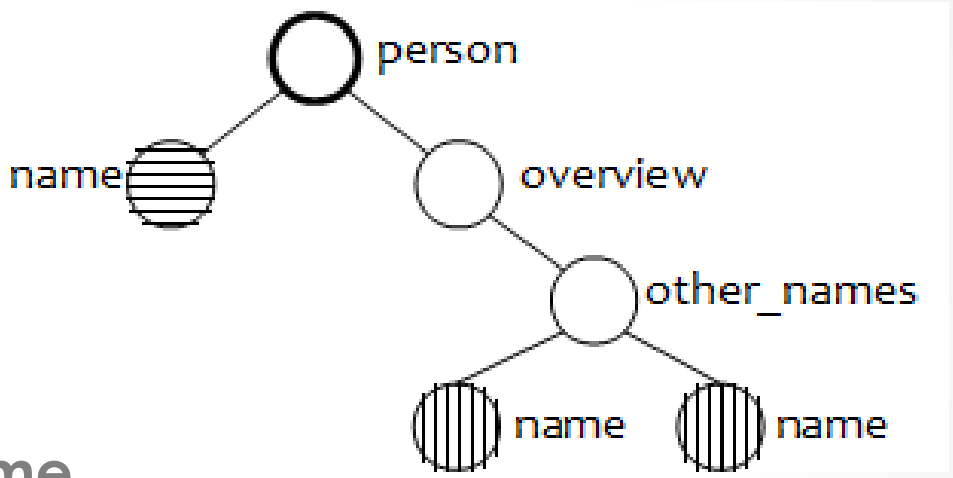
Adding flexible structure matching with the **Below** constraint



Flexible XPath ranking order: **Nodes are ranked by the distance from the context node (*person*)**

The Below constraint

Below Example:



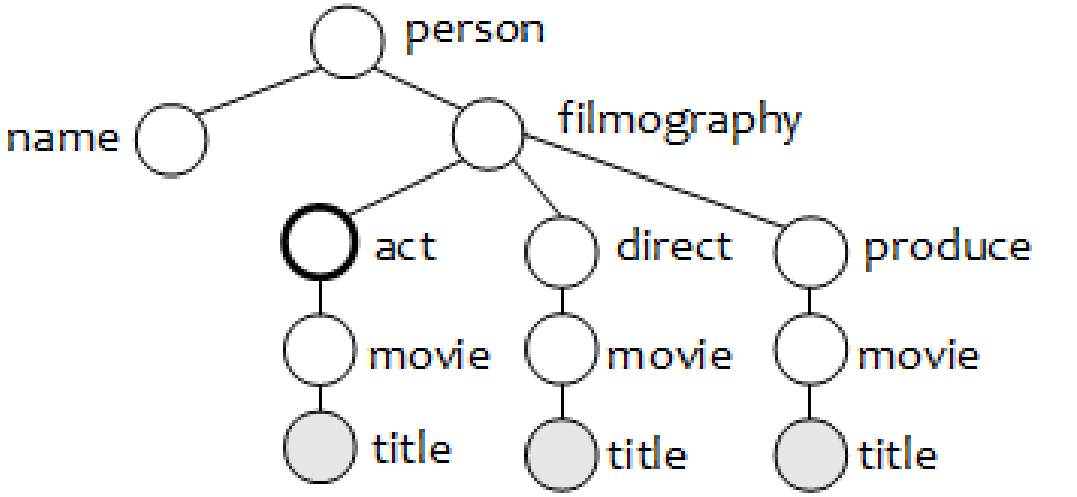
Query: **person /below::name**

Results:

1. person/name (score = 1.0)
2. Person/overview/other_names/name[1] (score = 0.3)
3. Person/overview/other_names/name[2] (score = 0.3)

The Near constraint

Near Example:

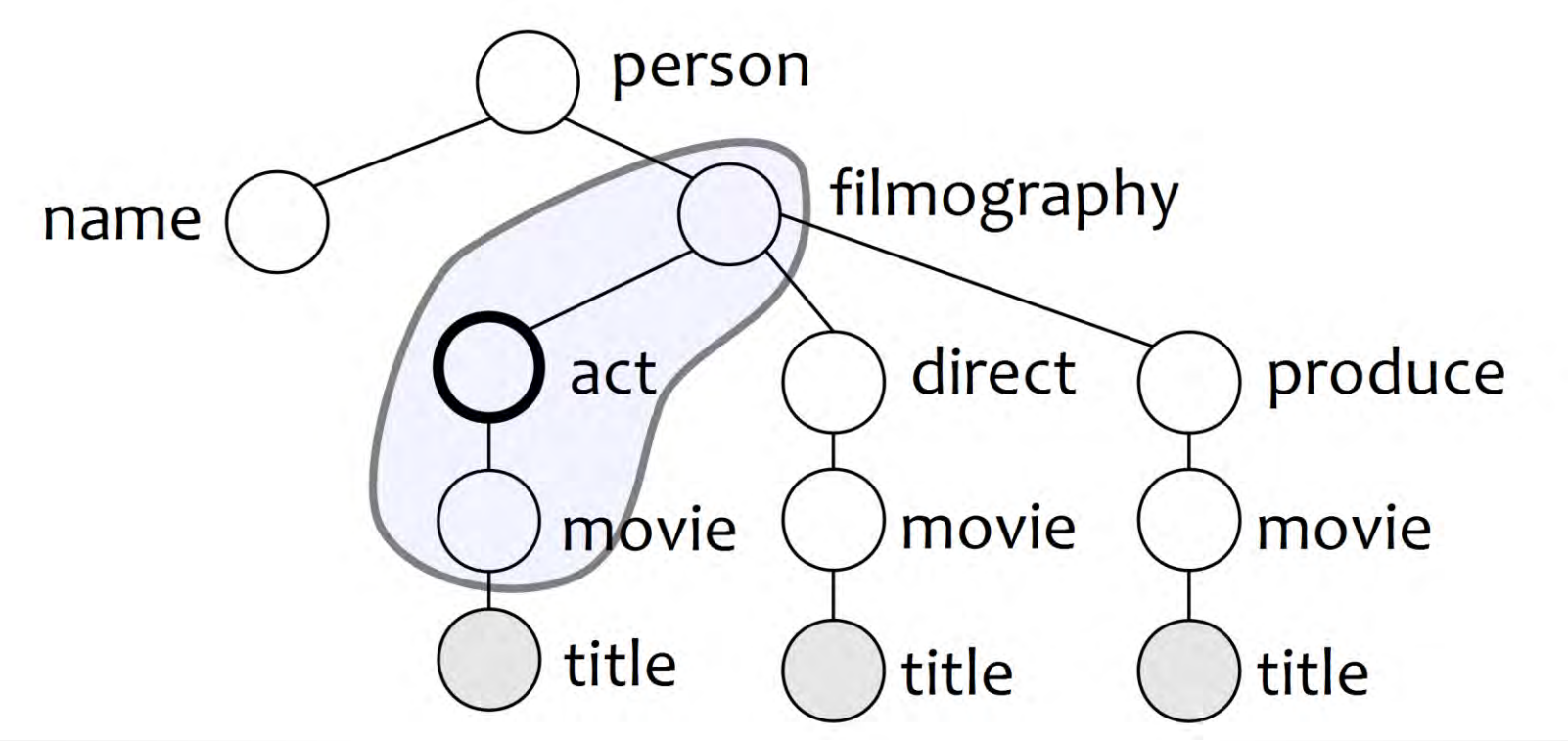


Query: **filmography/act /near::title**

Results:

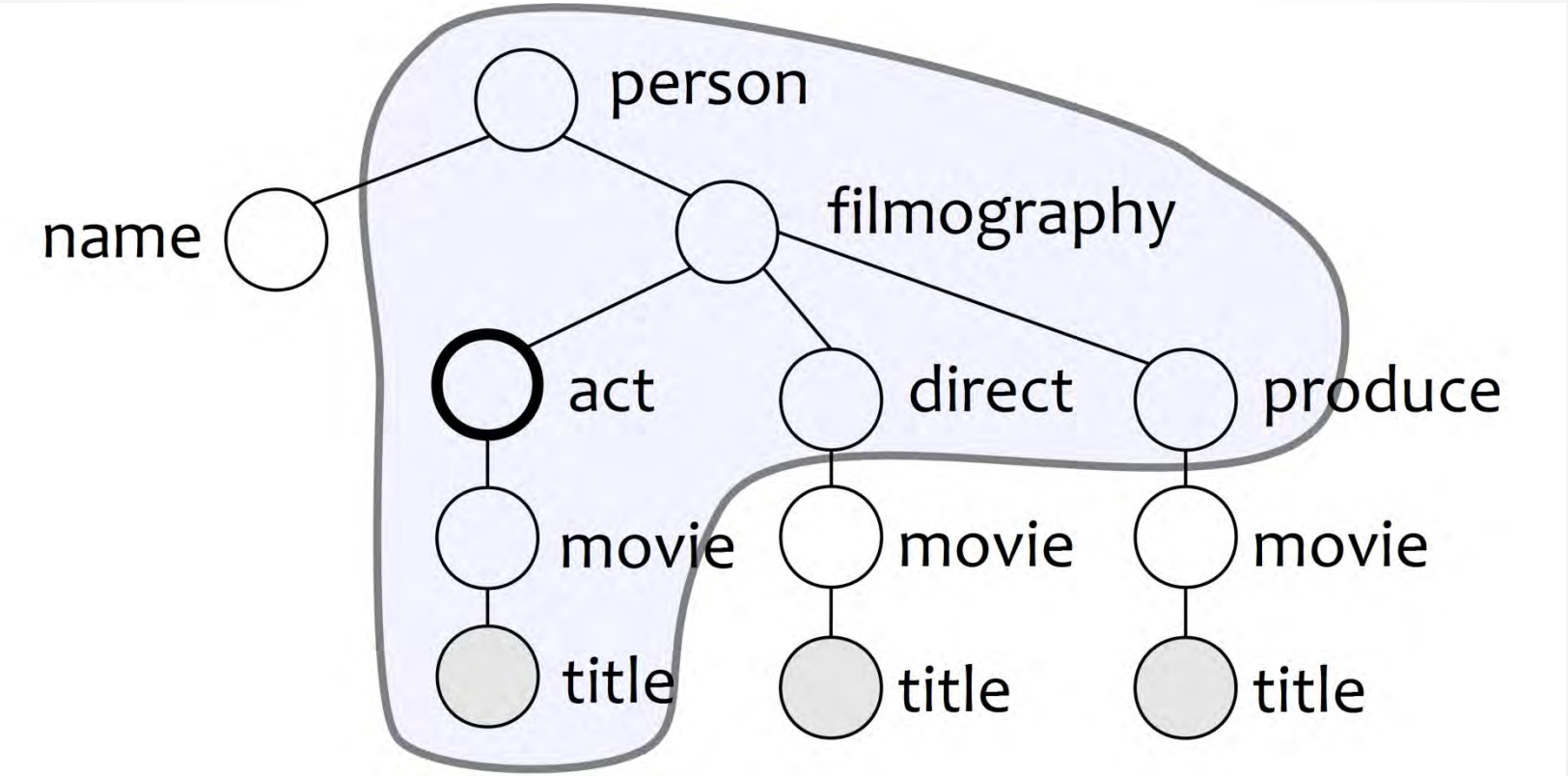
1. person/filmography/act/movie/title (score = 0.5)
2. person/filmography/direct/movie/title (score = 0.25)
3. person/filmography/produce/movie/title (score = 0.25)

The Near Evaluation



Near(1) evaluation

The Near Evaluation



Near(2) evaluation

XQuery FT extension

Structural-Scores have been integrated in XQuery FLWOR clauses

XQuery **FOR** clause example:

```
ForClause  ::= "for" "$" VarName TypeDeclaration?
    PositionalVar?  FTScoreVar? StructScoreVar?
    "in" ExprSingle ("," "$" VarName  TypeDeclaration?
     PositionalVar? FTScoreVar? "in" ExprSingle)*

FTScoreVar      ::= "score" "$" VarName

StructScoreVar::= "score-structure" "$" VarName
```

XQuery FT extension

Example query with **below** constraint and double fragment scoring (both Full-Text and structure)

```
for $item score $scoreFT score-structure $scoreS in
   person/below:name[text() contains text "brad"]
order by $scoreS
return <i scS='{$scoreS}' scFT='{$scoreFT}'>$item</i>
```

Results:

1. `<i scS="1" scFT="1">Brad</i>`
2. `<i scS="0.3" scFT="0.62">Brad Winsley</i>`

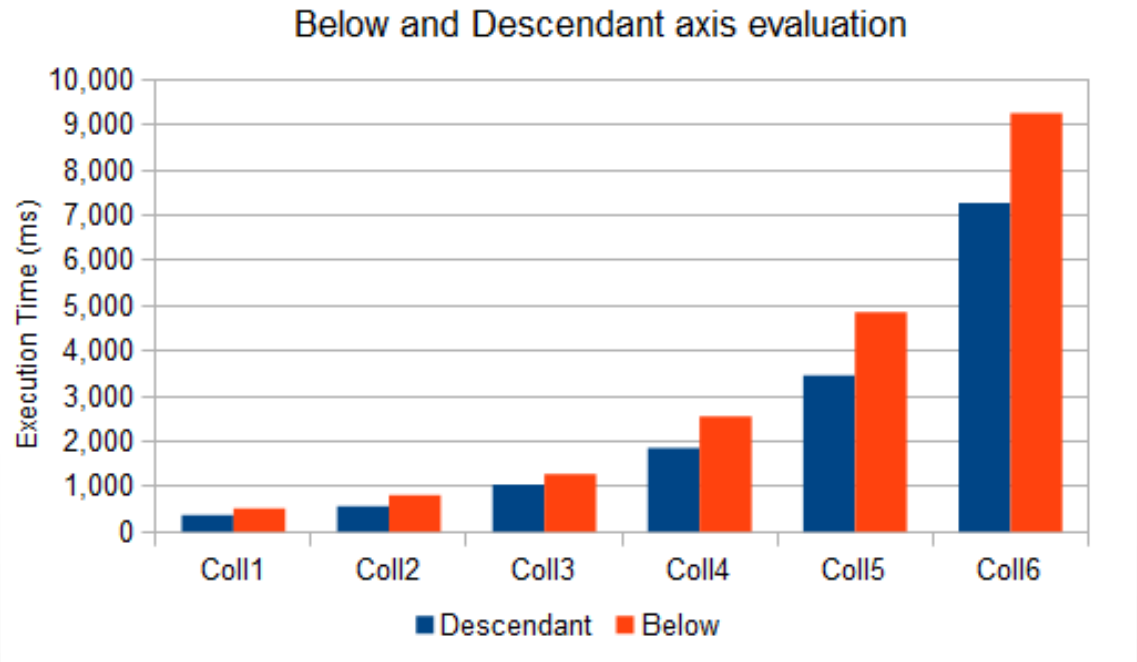
Implementation

Implementation on top of a XML Query Engine
(proof of concept)

- Integration of Near and Below in XQuery parser
  BaseX data-structures allow flexible structural matching
- FLWOR clauses integration with structural scores
  allow user to define scores aggregation
- Handling of ***branching queries*** with structural scores
- Implemented ***reverse-axis*** for BaseX internal query optimization and rewriting

Preliminary Results

- Below and Near constraints have been evaluated in comparison (if possible) with their XPath counterparts



The Near axis can not be compared due to its singularity in node identification (there is no XPath counterpart)

Future Work

Ongoing and Future work

- BaseX with flexible constraints for Patent Retrieval
- Improve Near and Below evaluation performances with ad-hoc data structures other than BaseX ones
- User evaluation of flexible constraints specification
- BaseX as an Information Retrieval System?