Table of Contents

Introduction ................................................. 4

FTContainsExpr ........................................ 5
Grammar .................................................. 5
AST ......................................................... 5

FTSelection .................................................. 6
Grammar .................................................. 6
AST ......................................................... 6

FTMatchOption ........................................... 7
Grammar .................................................. 7
AST ......................................................... 7

FTOrConnective .......................................... 8
Grammar .................................................. 8
AST ......................................................... 8

FTAndConnective ......................................... 9
Grammar .................................................. 9
AST ......................................................... 9

FTNegation .................................................. 10
Grammar .................................................. 10
AST ......................................................... 10

FTTimesSelection ....................................... 11
Grammar .................................................. 11
AST ......................................................... 11

FTDistanceSelection ................................... 12
Grammar .................................................. 12
AST ......................................................... 12

FTWindowSelection ..................................... 13
Grammar .................................................. 13
AST ......................................................... 13

FTScopeSelection ......................................... 14
Grammar .................................................. 14
AST ......................................................... 14
Introduction

This following document describes the conversion of XQuery Full Text expressions to an
AST representation. This document closely follows the layout and notation of the
previous XQuery document, which describes the conversion of XQuery
expressions to an AST representation.
FTContainsExpr

Grammar
FTContainsExpr ::=
  ComparisonExpr |
  FTScoreExpr |
  FTContainsExpr

AST

FTContainsExpr::searchCtx

is assigned the Expr on the left-hand of ftcontains.

FTContainsExpr::rootSelection

is assigned the FTSelection on right-hand side.

The optional FTSpecialCharsMatchOption is added to the options of the root selection.
FTSelection

Grammar:

FTSelection ::= FTOrConnective | FTSelection FTMatchOption

AST:

If an FTMatchOption is specified the corresponding FTMatchOption is added to FTSelection::options. See each FTSelection subclass for details how the particular FTSelection is processed.

The CreateTeXQueryContextSelection handles the presence of options.

Diagram:

[Diagram showing the structure of FTSelection]
FTMatchOption

Grammar:
FTMatchOption =
  FTStemMatchOption |
  FTCaseMatchOption |
  FTStopwordMatchOption |
  FTThesaurusMatchOption

AST:
See each FTMatchOption subclass for details on how it is processed.
The field `leftSel` is assigned the FTSelection on the left-hand side. The field `rightSel` is assigned the FTSelection on the right-hand side.
\[ FTAndConnective \]

Grammar:

\[ FTAndConnective \rightarrow \neg \neg FTAndConnective \]
\[ FTAndConnective \rightarrow \land \neg \neg \neg FTAndConnective \]

AST:

The left `sel` is assigned the `FTSelection` on the left-hand side. The right `sel` is assigned the `FTSelection` on the right-hand side.
FTNegation

Grammar

FTNegation := FTTimesSelection

FTNegation := "!!" FTNegation

AST

The field sel is assigned the nested FTSelection.
FTTimesSelection

Grammar

FTTimesSelection →
  FTTimesSelection |
  FTRangeSpec

FTRangeSpec →
  exactly? '{' AdditiveExpr '}' |
  at least '{' AdditiveExpr '}' |
  at most '{' AdditiveExpr '}' |
  from '{' AdditiveExpr '}' to '{' AdditiveExpr '}'

AST

FTRangeSpec::RangeSpecType = {ATMOST_RANGE, ATLEAST_RANGE, EXACTLY_RANGE, FROMTO_RANGE}

For FTRangeSpecType::type = ATMOST_RANGE, ATLEAST_RANGE, EXACTLY_RANGE, lbound == ubound are both assigned the nested XQuery expression. For FTRangeSpecType::type = FROMTO_RANGE, lbound is assigned the left nested expression, rbound is assigned the right nested expression.

FTTimesSelection::sel is assigned the nested FTSelection, FTTimesSelection::range is assigned the specified FTRangeSpec.
FTDistanceSelection

Grammar:

```
FTDistanceSelection ::= 
  FTWindowSelection  
  FTDistanceSelection "distance" FTRangeSpec ("word" | "words" | "sentence" | "sentences" | "paragraph" | "paragraphs")
```

For the FTRangeSpec production, see FTTimesSelection.

For notes on FTRangeSpec, see FTTimesSelection.

```
FTDistanceSelection::UnitType = {WORD_DISTANCE, SENTENCE_DISTANCE, PARA_DISTANCE}
```

FTDistanceSelection::sel is assigned the nested FTSelection,

FTDistanceSelection::range is assigned the specified FTRangeSpec.

FTDistanceSelection::unit is determined based on "word" | "words" | "sentence" | "sentences" | "paragraph" | "paragraphs" (plurals are same as singular forms).
FTWindowSelection

Grammar

FTWindowSelection ::= 
FTWindowSelection FTRangeSpec | FTWindowSelection FTWindowSelection

For notes on FTRangeSpec, see FTTimesSelection.

FTWindowSelection::UnitType = {WORD_WINDOW, SENTENCE_WINDOW, PARA_WINDOW}

FTWindowSelection::sel is assigned the nested FTSelection, FTWindowSelection::range is assigned the specified FTRangeSpec, FTWindowSelection::unit is determined based on "word" | "words" | "sentence" | "sentences" | "paragraph" | "paragraphs" (plurals are same as singular forms).
**FTScopeSelection**

**Grammar**

```
FTScopeSelection ::= 
  FTOrderSelection |
  FTScopeSelection ("same" | "different") ("sentence" | "paragraph")
```

**AST**

```
FTScopeSelection::sel is assigned the nested FTSelection.
FTScopeSelection::type is assigned based on the modifiers ("same" | "different") ("sentence" | "paragraph") specified.
```
FTOrderSelection::

FTOrderSelection ::= FTStringSelection

AST

FTOrderSelection::sel is assigned the nested FTSelection.
FTStringSelection

Grammar

FTStringSelection :=

  FTSelection | {InstanceOfExpr} ("any" | "all" | "phrase" | "any" word | "all" word)

AST

FTStringSelection::Type = {ANY_STRING_SELECTION, ALL_STRING_SELECTION, PHRASE_STRING_SELECTION, ANYWORD_STRING_SELECTION, ALLWORD_STRING_SELECTION}

The first branch of the production "FTStringSelection := FTSelection" is handled the same way the nested FTSelection is handled.

The interesting case is the second branch. FTStringSelection::type is determined based on the modifier ("any" | "all" | "phrase" | "any" word | "all" word) specified. The mapping to FTStringSelection::Type is obvious. If no modifier is specified, ANY_STRING_SELECTION is used. The searchTokens fields is set to the nested Expr, theInstanceOfExpr.
FTCaseMatchOption

Grammar:

FTCaseMatchOption =>
  | "lowercase" | "uppercase" | "case" "sensitive" | "case" "insensitive"

AST:

FTCaseMatchOption::CaseMatchValue = {CASE_SENSITIVE, CASE_INSENSITIVE, CASE_LOWER, CASE_UPPER}
$\text{FTStemMatchOption}$

$\text{Grammar}$

$\text{FTStemMatchOption ::= ("with" | "without") ("stems" | "stemming")}$

$\text{AST}$

$\text{FTStemMatchOption}$

AST is a $\text{Boolean}$.

$\text{"with"}$ translates to $\text{value} = \text{true}$, $\text{"without"}$ translates to $\text{value} = \text{false}$.

$\text{"stems"}$ and $\text{"stemming"}$ are equivalent.
FTStopwordsMatchOption

Grammar:

FTStopwordsMatchOption ::=

FTStopwordsOptionType = {WITH_STOPWORDS, WITHOUT_STOPWORDS_ADD, WITHOUT_STOPWORDS}

AST:

FTStopwordsMatchOption::FTStopwordsOptionType

\textit{add} is allowed only with \textit{with}. If \textit{with} \textit{stopwords} \textit{add} is used, \textit{FTStopwordsMatchOption::type} = \textit{WITH_STOPWORDS_ADD}; if only \textit{with} \textit{stopwords} is specified, \textit{FTStopwordsMatchOption::type} = \textit{WITH_STOPWORDS}; if \textit{without} \textit{stopwords} is specified, \textit{FTStopwordsMatchOption::type} = \textit{WITHOUT_STOPWORDS}.

\textit{FTStopwordsMatchOption::value} is assigned the optional \textit{Expr}; if none is specified, an \textit{Expr} generating an empty sequence is used.
FTTheaurusMatchOption

Grammar:

FTTheaurusMatchOption ::=

FTTheaurusMatchOption := AdditiveExpr |

AST

In case of "with thesaurus", FTTheaurusMatchOption::value is assigned the AdditiveExpr; otherwise, FTTheaurusMatchOption::value is an Expr generating an empty sequence.
FTSpecialCharsMatchOption

Grammar

FTSpecialCharsMatchOption :=
  "with" "special" "characters" |
  "without" "special" "characters"

AST

FTSpecialCharsMatchOption is parsed but no AST object is generated; currently a
NotImplementedException is thrown.