

# XQueryの等式変換とそのコストモデル

## XQuery Equalities and its Cost Model

日高宗一郎  
Soichiro HIDAKA

加藤弘之  
Hiroyuki KATO

吉川正俊  
Masatoshi YOSHIKAWA

### 何がわかる？

プログラムをより速く実行させる手法である最適化の一つとして、プログラムを構成する要素毎にそれと同じ働きでより速い要素に書き直す手法があります。この置き換えについて、本当に「同じ」かどうか、どれくらい「速い」のかを厳密に示す方法を明らかにします。

### どんな研究？

木構造で表現されたデータを格納するデータベースへの照会をするプログラムであるXQueryについて、構文要素が持つ数学的な性質を利用して自動的に変形を重ね、答えがより速く返ってくるものを書き直すことを目指しています。

## 状況設定

データベースの規模が大きくなるにつれ、書き換えによる最適化の効果も大きくなり、従来多くの変換規則が提案されているが、

証明が伴っていないため安心して用いることが出来ないものがある。  
独立に考案された類似の変換規則の間関係が判然としなない。  
効果の大きい変換規則の適用のために補助的に行う変換が、最適化の邪魔をしているかどうか心配である。  
プログラムの規模が大きくなると、どの部分が最も性能に影響を及ぼしているかを見極めるのが難しい。

等の問題がある。  
XQueryには厳密な形で定義された仕様があり、厳密な議論に生かすことが出来る。

## 研究状況

### 等式変換について

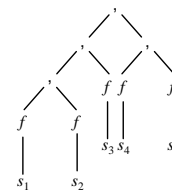
- XQuery構文毎に成り立つ基本的な変換規則を洗い出した。
- XQueryを特徴づける基本的な構文について、共通する別の構文に帰着させられることが分かった。
- 既に提案されているXQueryの変換規則の等価性を検証した。

### コストモデルについて

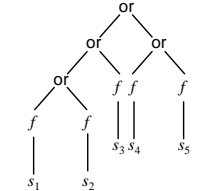
- 変換のプログラムの速さを定量的に比較出来るように、プログラムのコストを計算する式を開発した。
  - ・プログラムの成り立ちに従いコストも要素毎に積み上げていく形式を取っている。
  - ・このコスト計算モデルを用いて、補助変換をコストを「増加させる可能性のあるもの」、「変えないもの」等に分類することが出来た。
  - ・具体的なコストを見積もるためにはデータベースの中身を知らなければならないが、データベースの具体的な中身によらないコスト計算を可能にするため、コストの式を相対的なものにし変換前後のコストの差分という形で変換の損得が計算出来るようにしたことが特徴である。

**コスト関数** $C(e)$  部分式のコスト関数、サイズ関数と確率関数を使って再帰的に記述される  
**サイズ関数**(補助的関数) $S(e)$  部分式のサイズ関数および確率関数を使って記述される(部分式のコストに依存しない)  
**確率関数**(補助的関数) $P(e)$  部分式の確率関数とサイズ関数で記述される(部分式のコストに依存しない)

for  $Sx$  in  $(s_1, s_2, s_3, s_4, s_5)$   
return  $f(Sx)$



some  $Sx$  in  $(s_1, s_2, s_3, s_4, s_5)$   
satisfies  $f(Sx)$



構文要素に共通する計算パターンの抽出

```
+S(distinct-values($auction/site/people/person/profile/interest/@category))
·(S($auction/site/people/person) ·P($t/profile/interest/@category = $i)
·C(<personne>
...
<revenu> { fn:data($t/profile/@income) } </revenu> (: 1 :)
...
<ville> { $t/address/city/text() } </ville> (: 2 :)
...
</personne>)
-C(<revenu> { fn:data($t/profile/@income) } </revenu>) (: 1' :)
-P(<revenu> { fn:data($t/profile/@income) } </revenu> > 10000)
·C(<ville> { $t/address/city/text() } </ville>) (: 2' :)
+(C($p1/statistiques/revenu)
+P($p1/statistiques/revenu > 10000)
·C($p1/coordonnees/ville))
)
+C(<categorie><id>{ $i }</id>{ $p }</categorie>)
)
+C($view/personne)
```

コスト変化計算の具体例