

# Подход к переносимой реализации языка XQuery

Паращенко Олег

8 июля 2004 г.

Protva XQuery

Санкт-Петербург

Санкт-Петербургский Государственный Университет

## Аннотация

XQuery, язык запросов XML, можно рассматривать как обобщённый интерфейс к древовидным данным. В работе обосновывается идея создания переносимой системы XQuery, которая может обрабатывать любые деревья, и предлагается реализация с помощью виртуальной машины.

## 1 Зачем обычным приложениям XQuery

Один из подходов к обработке XML[2] — это представление его в виде дерева, для навигации по которому используется XPath[3]. На основе XPath созданы языки для преобразования XML: XSLT[4] и XQuery[5]. XSLT является языком шаблонов, а XQuery можно представить как значительно расширенный XPath.

XML можно рассматривать как внешнее текстовое представление древовидных структур, а связанные с XML стандарты — как способы обработки таких данных. Такой подход может оказаться полезным для некоторых видов программ, например,

- для текстовых процессоров и
- компиляторов и интерпретаторов.

В подобных приложениях важную роль играют сложные запросы к деревьям, например:

- найти все заголовки, не помеченные как удалённые, и создать из них оглавление;
- найти присваивания вместо сравнений в блоках условий команд if, for, while и им подобных.

Прямолинейная реализация таких приложений зачастую сложна, а код оказывается более низкого уровня чем решаемая задача. В качестве альтернативы предлагается использовать встроенный XQuery, что должно упростить написание программ.

## 2 Подход к реализации

Основная проблема при разработке системы XQuery — это большой объём работ, поэтому не стоит писать XQuery каждый раз заново. Лучше создать переносимую реализацию и портировать её в приложения с минимальными затратами.

Для этого предлагается разработать виртуальную машину для обработки деревьев и систему XQuery на её основе. Если какому-либо приложению требуется поддержка XQuery, то достаточно будет реализовать эту машину.

Чтобы упростить разработку, код пишется и отлаживается на языке Common Lisp, а затем транслируется в ассемблер виртуальной машины. За основу взята система CL-XML[6]. James Anderson, её автор, проделал большую работу, так что CL-XML уже обрабатывает все примеры из спецификации XQuery.

CL-XML переводит XQuery во внутреннее промежуточное представление в виде абстрактного синтаксического дерева (AST). Этот формат можно использовать для высокоуровневой оптимизации запросов.

Большая часть языковых конструкций Common Lisp реализована с помощью макросов. Если взять CL-XML и раскрыть все макросы, то получится программа, состоящая из примитивов. Набор полученных конструкций является черновой версией виртуальной машины.

Реализация диалекта минимального Lisp (или Scheme) не составляет сложности, так как является хорошо изученной задачей [7].

Отдельный интерес представляет трансляция нашей виртуальной машины на LLVM[8] — промежуточное представление и набор инструментов для оптимизации программ. Благодаря LLVM и высокоуровневой оптимизации, код вычисления XQuery может быть эффективнее вручную написанного аналога.

## 3 Статус

Проект находится в начальной стадии. Пока только готов план работ, проведены первые эксперименты и подобраны инструменты для реализации. Промежуточные отчёты будут выкладываться на странице проекта Protva XQuery[1], код будет распространяться по лицензии LGPL.

На конференции будут представлены черновая версия виртуальной машины и первые результаты.

Первой вехой проекта будет добавление поддержки XQuery в XSLT-процессор xsltproc[9].

## Список литературы

- [1] Проект Protva XQuery <http://xmlhack.ru/protva/>
- [2] Extensible Markup Language (XML) <http://www.w3.org/XML/>
- [3] XML Path Language (XPath) <http://www.w3.org/TR/xpath>

- [4] The Extensible Stylesheet Language Family (XSL) <http://www.w3.org/Style/XSL/>
- [5] XML Query (XQuery) <http://www.w3.org/XML/Query>
- [6] CL-XML: Common Lisp support for XML <http://cl-xml.org/>
- [7] Scheme Compiler Technology/Implementation Techniques and Optimization <http://library.readscheme.org/page8.html>
- [8] The LLVM Compiler Infrastructure Project. <http://llvm.cs.uiuc.edu/>
- [9] The XSLT C library for Gnome <http://xmlsoft.org/XSLT/>