

Этюды с парсер-комбинаторами

Дмитрий [Юрьевич] Булычев

СПБАУ

30 мая 2012

Санкт-Петербург

Парсер-комбинаторы

Идея: выразить синтаксический анализатор через элементарные распознаватели и функции высшего порядка, соответствующие синтаксическим операторам (например, EBNF).

- Graham Hutton and Erik Meijer. **Monadic parser combinators** // J. of Functional Programming, №1, 1993.
- Daan Leijen and Erik Meijer. **Parsec: Direct Style Monadic Parser Combinators for the Real World** // Technical Report, University of Utrecht, 2001.
- Bryan Ford. **Packrat Parsing: Simple, Powerful, Lazy, Linear Time** // ICFP'2002.

Pro et contra

- + Хорошо интегрируются в среду основного языка реализации;
- + Разбирают более релевантный класс языков;
- + Допускают простую реализацию многоуровневых и расширяемых анализаторов.

- Императивность описания;
- Трудности с восстановлением после ошибок и их диагностикой;
- Проблемы с производительностью.

Этюд №1: левая рекурсия и “LMF”

Классические парсер-комбинаторы плохо переносят леворекурсивные спецификации. Устранение левой рекурсии статически вообще говоря невозможно:

```
list[item]: item ("," item)*
```

Эффективные парсер-комбинаторы чувствительны к порядку альтернатив (Longest-Match First Rule):

```
expression:  
  identifier  
| identifier "+" expression
```

Этюд №1 (окончание)

Способы борьбы:

- Richard A. Frost and Rahmatullah Hafiz. **A New Top-Down Parsing Algorithm to Accommodate Ambiguity and Left Recursion in Polynomial Time** // ACM SIGPLAN Notices, Vol. 41 (5), May 2006;
- Richard A. Frost, Rahmatullah Hafiz, and Paul Callaghan. **Parser Combinators for Ambiguous Left-Recursive Grammars** // PADL 2008, LNCS 4902, Springer-Verlag Berlin Heidelberg 2008;
- Laurence Tratt. **Direct Left-Recursive Parsing Expression Grammars** // EIS-10-01, School of Engineering and Information Sciences, Middlesex University, October 2010.

Этюд №2: диагностика и восстановление после ошибок

Очень просто точно сообщить о первой ошибке. Очень трудно восстановиться и сообщить об остальных.

Continuation-based Applicative Parser Combinators:

- S. Doaitse Swierstra and Pablo R. Azero Alcocer. **Fast, Error Correcting Parser Combinators: A Short Tutorial** // University of Utrecht, 1999;
- S. Doaitse Swierstra. **Combinator Parsers: From Toys to Tools** // ICFP Workshop on Haskell, 2000.

Этюд №2 (окончание)

Можно по-другому: используя понятие расстояния между строчкой и языком:

- Gordon Lyon. **Syntax-Directed Least-Errors Analysis for Context-Free Languages: A Practical Approach** // CACM, Vol. 17, № 1, 1974;
- Julia Dain. **Minimum Distance Error Correction** // Dept of Computer Science University of Warwick, UK, 1987.
- Giovanni Pighizzini. **A Parallel Minimum Distance Error-Correcting Context-Free Parser** // Theoretical Computer Science, Proc. of the Fourth Italian Conference, 1992;

Этюд №3: парсер-комбинаторы и генераторы времени исполнения

Идея: генерировать эффективный анализатор сразу перед его применением. Помогает решить многие проблемы, включая проблему эффективности.

Уже реализовано в некоторых парсер-генераторах.

Пока не применялось для парсер-комбинаторов.

По-другому: смешанные вычисления (суперкомпиляция) для синтаксических анализаторов.

Экспериментальная база

- Objective Caml (<http://caml.inria.fr>);
- Ostop (<http://caml.inria.fr/cgi-bin/hump.en.cgi?contrib=513>)
— монадические парсер-комбинаторы с произвольным lookahead + синтаксическое расширение;
- HaSCoL, Oberon0 — готовые синтаксические анализаторы, на которых можно что-нибудь испытать.

Спасибо за внимание!

- Вопросы?
- Предложения?
- Пожелания?