

発表概要

## 抽象構文木に基づいた解析表現文法への静的型付け

山口 大輔<sup>1,a)</sup> 倉光 君郎<sup>1,b)</sup>

2017年3月4日発表

多くのパーサジェネレータでは、パーサが受理する言語のほかにパーサが受理した文字列に対して行う処理の定義を与える必要がある。Nez は PEG (Parsing Expression Grammar, 解析表現文法) をベースとした open grammar 言語であり、これらの定義を Nez では統合して行うことができる。しかし、Nez は型なし言語である。本発表では、Nez に対して型システムを導入する。型は、生成されるパーサのパーズ結果が成功である場合に得られる AST (Abstract Syntax Tree, 抽象構文木の構造) に対応する。これによって、無意味な構文定義を型システムによって静的に検出することが可能になるとともに、型の持つ情報から、AST の概形を事前に予想することができる。

## Statically Typing for Parsing Expression Grammars Based on Abstract Syntax Tree

DAISUKE YAMAGUCHI<sup>1,a)</sup> KIMIO KURAMITSU<sup>1,b)</sup>

Presented: March 4, 2017

In most parser generators, developers need to give definitions of both a language accepted by the parser and operations for the strings accepted. Nez is a PEG (Parsing Expression Grammar)-based open grammar language that allows developers to integrate operations into the language. However, Nez is an untyped language. In this presentation, we describe the type system whose type is correspond to abstracted structure of AST (Abstract Syntax Tree). This type system detects error syntax definitions statically and provides the information about the structure of AST constructed from matched string.

---

<sup>1</sup> 横浜国立大学  
Yokohama National University, Yokohama, Kanagawa 240-8501, Japan

a) yamaguchi-daisuke-bf@ynu.jp

b) kimio@ynu.ac.jp