

発表概要

# OCommand : OCaml 上の型安全な シェルプログラミングのための領域特化言語の提案

朝倉 泉<sup>1,a)</sup> 増原 英彦<sup>1,b)</sup> 青谷 知幸<sup>1,c)</sup>

2014年6月20日発表

OCaml プログラム中からシェルコマンドを型安全に実行するための領域特化言語 OCommand を提案する。OCommand はコマンドの仕様である、出力行の各欄の型とコマンドオプションによって起きる変化の記述を受け取り、コマンド起動関数とコマンドオプションを表す値定義を持つ OCaml モジュールを生成する。生成された起動関数は、複数のオプション値を受け取りコマンドを実行し、オプションに応じてコマンド出力行をパースして型付きのフィールドからなるレコードを返す。出力行レコードにおけるフィールドの型や存在の変化を静的に検査するため、一般化代数的データ型を用いた。Camlp4 を用いて実現した処理系を用いて、これまでに `ls` や `ps` などのコマンドを扱えることを確めている。

## OCommand: A Domain Specific Language for Type Safe Shell Programming in OCaml

IZUMI ASAKURA<sup>1,a)</sup> HIDEHIKO MASUHARA<sup>1,b)</sup> TOMOYUKI AOTANI<sup>1,c)</sup>

Presented: June 20, 2014

This presentation proposes We proposes a domain-specific language (DSL), called OCommand, for executing shell commands from within OCaml programs in a type safe way. OCommand takes a command specification consisting of types of output columns from the command and effects of command options on the types of the columns, and generates an OCaml module that contains a command-executing function and a set of values representing the command options. The command-executing function takes options as arguments, runs the command, parses the output lines, and returns them as a list of records. In order to statically-check existence and types of fields of the output record that can be changed by command options, we used generalized algebraic data type. With our implementation constructed by using Camlp4, we successfully handle typical Unix commands like `ls` and `ps` by using OCommand.

---

<sup>1</sup> 東京工業大学大学院 数理計算科学専攻  
Department of Mathematical and Computing Sciences,  
Tokyo Institute of Technology, Meguro, Tokyo 152-8550,  
Japan

a) asakura.i.aa@m.titech.ac.jp

b) masuhara@acm.org

c) aotani@is.titech.ac.jp