Coalgebraic effects and their cohandlers in programming languages

(Efekty koalgebraiczne oraz ich kohandlery
w językach programowania)

Mateusz Urbańczyk

Praca inżynierska

Promotor: dr Maciej Piróg

Uniwersytet Wrocławski
Wydział Matematyki i Informatyki
Instytut Informatyki

30 czerwca 2020

Abstract

. . .

. . .

Contents

Chapter 1

Introduction

1.1 Problem Analysis

1.2 Problem Statement

1.3 Thesis Outline

. . . [1]

Chapter 2

Background

2.1 Algebraic Effects

2.2 Categorical Setting for Universal Algebra

2.3 Comodels and Duality

Chapter 3

Proposed Solutions

3.1 Dynamic Constraints Checking

3.2 Linear Types

3.3 Data-Flow Analysis

3.4 Cohandlers as a Separate Construct

3.5 Related Work

Chapter 4

Coeffectful Programming with Cohandlers

4.1 Examples

4.2 Usage guide

Chapter 5

Calculus of Freak language

5.1 Syntax

5.2 Typing Rules

5.3 Operational Semantics

5.4 Continuation Passing Style Transformation

Chapter 6

Implementation

6.1 Abstract Syntax Trees

6.2 Curried Translation

6.3 Uncurried Translation

6.4 Cohandlers

6.5 Type System

6.6 Source Code Structure

Chapter 7

Conclusion

7.1 Summary

7.2 Future work

Bibliography

[1] Sam Lindley, Conor McBride, and Craig McLaughlin. “Do be do be do”. In: *CoRR* abs/1611.09259 (2016). arXiv: 1611.09259. URL: http://arxiv.org/abs/1611.09259.