Exception Handling in Haskell

William S. Fisher

August 10, 2016

Abstract

Implementing exception handling in Haskell. Unlike other libraries, use named exception handlers. Use the λ^{try}-calculus to formalize and explore a series of translations between multiple calculi to arrive at a translation into Haskell. Explore properties of this translation including soundness and completeness. Publish useable Haskell library.

Acknowledgements

Thanks me

Contents

Chapter 1

Introduction

Hel

Chapter 2

Background

2.1 λ-Calculus

2.2 Haskell

2.3 Logic, Types, and their Computation Interpretation

2.3.1 Continuations

2.3.2 Delimited-Continuations

2.4 $\lambda\mu$-Calculus

2.5 λ^{try}-Calculus

2.6 Delimited-Continuation Calculus

Chapter 3

DCC Interpreter

3.1 Interpreter

Chapter 4

Translations

4.1 λ^{try}-to-$\lambda\mu$

4.2 $\lambda\mu$-to-DCC

4.3 λ^{try}-to-DCC

Chapter 5

Conclusion

5.1 Evaluation

5.2 Conclusion

5.3 Future Work

Bibliography

[1] R. Kent Dybvig, Simon L. Peyton Jones, and Amr Sabry. A monadic framework for delimited continuations. *J. Funct. Program.*, 17(6):687–730, 2007.