

プログラミング言語の設計と実装

Design and Implementation of Programming Languages

キーワード：言語処理系、コンパイラ /key words: programming, compiler

木山 真人 助教 Ph.D. / Masato KIYAMA Research Associate., Ph.D.

情報・エネルギー部門 先端工学第四分野（ビッグデータ） / Big Data Science and Technology

E-mail : masato@cs.※ Tel : 096-342-3607

●拡張性のあるパーサジェネレータの作成

拡張性のあるPEGパーサジェネレータを開発している。拡張性を確保するため、Extensible Effectsという機構を参考に新たなMonad Transformerの機構を開発している。パーサでは、状態の保存とエラー時の処理が主要な部分となる。これはMonadで表現でき、Monad Transformerで拡張すれば、パーサとしては問題ない。しかし、別のアプリケーションに拡張するためには、さらなる発展が必要となり、拡張性の議論が必要である。Monadは関数型言語のプログラミング、特にHaskellにおいて、プログラミングの根幹的な部分である。その根幹的な部分を組み合わせる方法であるMonad Transformerは非常に重要な部分となる。MonadとMonad Transformer、さらに新たな機構を用いてパーサ自体を最小限の状態から構築する考えが本研究の斬新的なアイデア部分であり、この機能を中心にパーサを作成することで、拡張性を確保できると考えている。

Extensible PEG parser generator : We develop extensible PEG parser generator in Haskell and new Monad Transformer to provide an extensibility with users. The main parts of parser are storing states and error process. These parts can be expressed by Monad and be extend by Monad Transformer. However Improvement and discussion of new Monad Transformer is needed to adapt extensibility to another applications. Monad is an essential concept in Haskell. A monad is a way to structure computations in terms of values and sequences of computations using those values. Our novel idea is that we can construct a parser from minimal state using new monad transformer. That method provides an extensibility with users.

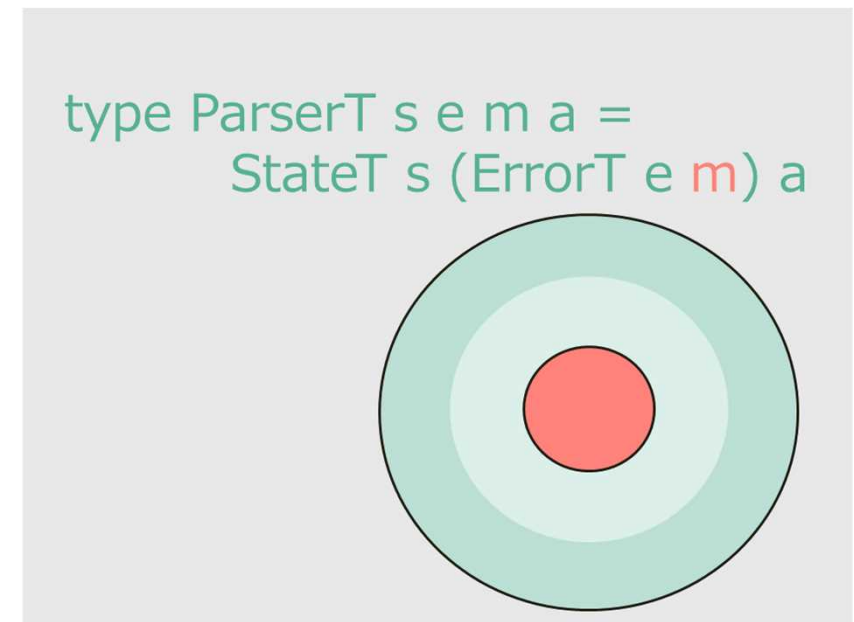


Figure Concept of Extensible PEG Parser Generator