

演習 3 (4/23)
31010 佐藤秀明

選択した言語

- XSLT
 - もっとも普及している
 - オーソドックスな手法
- CDuce
 - 型の存在
 - Xduce を元に発展
- 2つの言語間の差異に興味

XSLT

- それ自体が xml 構造で記述される
- XSLT の名前空間に存在する要素、関数、演算子などを利用する
 - `<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">`

XSLT

- XML 構造の各ノードに対して変換規則を作成
 - `<xsl:template match="node1">`
`<xsl:apply-templates select="node2"/>`
`</xsl:template>`
 - 文脈自由文法風
- XPATH
 - XML 構造中の任意の位置にあるノードを表現できる
 - 絶対パス / 相対パス
 - 離れているノードの情報を用いた規則が作成できる

CDuce

- Ocaml に似た文法
- 型を導入
 - `type T = [S*];; type S = [T];;`
 - DTD に相当
 - 読み込んだ xml の型チェックをする
- let 束縛や高階関数などの作成が可能
- 関数のオーバーロード
 - 木構造を続けてたどる際などに有効

現状

- チュートリアル・リファレンスに目を通した
- 処理系のインストール完了（結構はまった）
- 双方の言語について、ごく簡単なプログラムを作成してみた

目標

- インターネット上にあるデータ（公開されているデータベースや WWW ページなど）を XML 形式で蓄積し、それを利用するような便利なアプリケーションを作成する。
- 各言語の紹介論文を読み、理解する。