

Tutorium: Einführung in Funktional-Reaktives Programmieren mit Elm

Janis Voigtländer

Universität Bonn, Institut für Informatik

<http://www.janis-voigtlaender.eu/>

Elm (<http://elm-lang.org/>) ist eine funktionale Programmiersprache, die als JavaScript-Alternative positioniert wird. Sie übernimmt viele Sprachmerkmale aus der ML-Familie und aus Haskell, etwa:

- strenge, statische Typung und Typinferenz
- algebraische Datentypen, Polymorphie, Records mit Subtyping
- pur wie Haskell (mit per Typsystem kontrollierten Seiteneffekten)
- strikt auswertend wie ML

Einfache Haskell-Programme sind oft bis auf kleine syntaktische Differenzen (etwa Umbenennung von Schlüsselworten) auch Elm-Programme. Andererseits gibt es bewusste Abweichungen hinsichtlich Sprachmerkmalen (etwa bisher keine Typklassen) und „Philosophie“. Vorderstes erklärtes Zieleinsatzgebiet ist Web-Frontend-Programmierung. Entsprechend gut ist die Unterstützung für deklaratives HTML, Canvas etc. Der existierende Compiler übersetzt nach JavaScript.

Dass Wachstum der Elm-Community vor allem durch „Konvertierung“ von JavaScript-Programmierern, nicht von Haskell-Programmierern, angestrebt wird, schlägt sich auch in Entscheidungen zum Sprachdesign und Tooling wieder. Dennoch, oder vielleicht gerade deswegen, kann Elm auch für ein Haskell-Publikum interessant sein. Ziel des Tutoriums ist eine Einführung in Elm. Insbesondere soll das Konzept Funktional-Reaktive Programmierung (FRP) verwendet werden. FRP ist auch in Haskell möglich (es gibt mehrere entsprechende Bibliotheken), in Elm ist das Konzept jedoch zentral in der Sprache verankert, wegen des Zieleinsatzgebietes (Reaktivität für Web-Frontends).

Bei am Tutorium Teilnehmenden werden mindestens Kenntnisse vorausgesetzt, wie sie typischerweise in einer einsemestrigen Vorlesung zu Haskell erworben werden. Details zur notwendigen technischen Vorbereitung (Installation etc.) werden noch auf <https://github.com/jvoigtlaender/Elm-Tutorium> bekannt gegeben.