

Bachelorarbeit: Funktionale reaktive Programmierung in Java

Funktionale reaktive Programmierung (http://en.wikipedia.org/wiki/Functional_reactive_programming), kurz FRP, ist ein neues Programmierparadigma für reaktive, ereignisorientierte Programme, welche auf funktionalen Prinzipien beruht. Typische Anwendungsbereiche sind z.B. graphische Benutzeroberflächen oder ereignisorientierte Systeme Steuerungen.

Die wesentliche Idee von FRP ist die Verarbeitung der zeitlichen Abfolge von Ereignissen über eine Kette von Funktionen zu modellieren. Dabei kann ein Ereignis eine Message oder die Änderung eines Datenwertes sein. Tritt ein Ereignis auf, löst das automatisch eine Kette von Reaktionen aus, die über funktionale Zusammenhänge beschrieben sind, bis schließlich ein Ergebniszustand erreicht ist.

Ein attraktives System für FRP ist Scala.React¹. Das System basiert auf folgenden grundlegenden Konzepten:

- *Ereignisströme (Events)*: Stellen eine zeitliche Folge von Ereignissen dar.
- *Signale (Signal)*: Sind zeitliche Werteverläufe, die sich ereignisorientiert ändern.
- *Variablen (Var)*: Sind Signale, bei denen man einen Wert setzen kann.
- *Beobachter (Observer)*: Reagieren auf Änderungen von Signalen und auf Ereignisse.
- Der Abbildung von Signalen und Ereignissen über funktionale Ketten.
- Der Propagierung von Ereignissen und Änderungen von Signalen über funktionale Abhängigkeiten.

In dieser Bachelorarbeit soll ein ähnliches Programmsystem für FRP in Java implementiert werden. Die Implementierung soll auf den besonderen Features von Java aufgebaut werden. Des Weiteren soll in der Bachelorarbeit ein Vergleich mit anderen Systemen für FRP in Java durchgeführt werden.

In einer weiteren Bachelorarbeit soll ein analoges System in C# implementiert werden. Die beiden Bachelorarbeiten können in enger Kooperation realisiert werden und sollten dann einen Vergleich mit dem jeweils anderen System beinhalten.

Betreuung: Dr. Herbert Prähofer

¹ <http://infoscience.epfl.ch/record/176887/files/DeprecatingObservers2012.pdf>