

Zwei Fliegen mit einer Klappe: Bidirektionales Serialisieren und Deserialisieren

Stefan Wehr (wehr@factisresearch.com)

factis research GmbH, Freiburg im Breisgau

19. April 2011

Programme müssen häufig Werte von Datentypen in eine externe Repräsentation konvertieren (Serialisieren) oder aus einer externen Repräsentation einlesen (Deserialisieren). Hierfür gibt es eine Vielzahl von Tools, die aus einer Spezifikation den entsprechenden Code zur Serialisierung und Deserialisierung generieren. Dabei verliert der Programmierer jedoch häufig die Kontrolle über die Form der Datentypen oder das genau Format der serialisierten Daten.

Basierend auf einer Idee von Rendel und Ostermann [2] demonstriert der Vortrag, wie man in einer eingebetteten Sprache Serialisierung und Deserialisierung auf kompakte Art spezifizieren kann und dabei trotzdem volle Kontrolle über Form der Datentypen und das Serialisierungsformat behält. Darüberhinaus bietet der Ansatz folgende Vorteile:

- Die Spezifikation ist bidirektional: unter Einhaltung klar definierter Bedingungen ist der Serialisierungscode invers zum Deserialisierungscode.
- Die Spezifikationsprache ist um verschiedene Ausgabeformaten erweiterbar. In der Praxis wurden als Ausgabeformate bisher XML, reiner Text und HL7¹ Nachrichten erprobt. Es ist sogar möglich, den Begriff “Ausgabeformat” noch weiter zu fassen und z.B. Datentypen in HTTP Antworten bzw. Anfragen zu (de-)serialisieren.

Im zweiten Punkt zeigt sich ein wichtiger Unterschied zur Arbeit von Rendel und Ostermann sowie zu verwandten Arbeiten von Kennedy [1] und der Haskell Bibliothek HXT:² alle diese Arbeiten unterstützen lediglich ein fixes Ausgabeformat (Text, rohe Bytes bzw. XML).

Bisher liegen Implementierungen der Spezifikationsprache für Haskell und Scala vor.³ Beide Implementierung wurden bereits im Rahmen kommerzieller Projekte praktisch erprobt.

Literatur

- [1] Andrew Kennedy. Pickler combinators. *Journal of Functional Programming*, 14(6):727–739, 2004.
- [2] Tillmann Rendel and Klaus Ostermann. Invertible syntax descriptions: unifying parsing and pretty printing. In *Proceedings of the third ACM Haskell symposium on Haskell*, pages 1–12, New York, NY, USA, 2010. ACM.

¹<http://www.hl7.org/>

²<http://www.fh-wedel.de/~si/HXmlToolbox/>

³<http://darcs.umidev.de/pub/roundtrip/>, <http://darcs.umidev.de/pub/roundtrip-scala/>