

Language-Integrated Querying for Haskell

Alexander Ulrich

March 29, 2013

Abstract

We present *Database-Supported Haskell* (DSH), a library for language-integrated database queries in Haskell. Program fragments supposed to be executed on a database system (i.e. database queries) are expressed using familiar combinators from the standard list prelude and are entirely executed on the database system. Programming with the expressive list comprehension notation is also supported. A query's result may consist of basic types and arbitrarily nested combinations of lists and tuples. In addition, any (non-recursive) algebraic data type is supported. The implementation avoids unnecessary data transfer and context switching between the database coprocessor and the programming language runtime: a DSH program fragment is implemented by a small number of relational queries whose number is statically determined by the program fragment's type and does not depend on the database size.

Wir stellen Database-Supported Haskell (DSH) vor, eine Bibliothek zur Integration von Datenbankabfragen in Haskell. Programmfragmente, die auf einem Datenbanksystem ausgeführt werden sollen (d.h. Datenbankabfragen) werden mittels der gewohnten Listenkombinatoren aus der Haskell-Standardbibliothek formuliert und werden vollständig auf dem Datenbanksystem ausgeführt. Neben Kombinatoren können Abfragen auch mittels der ausdrucksstarken Notation für Listen-Comprehensions formuliert werden. Das Ergebnis einer Query kann aus Basistypen sowie beliebig verschachtelten Kombinationen von Listen und Tupeln bestehen. Zusätzlich werden alle (nicht-rekursiven) algebraischen Datentypen unterstützt. Die Implementation der Bibliothek vermeidet unnötige Datenübertragungen und Kontextwechsel zwischen dem Datenbanksystem und dem Haskell-Heap: Ein DSH-Fragment wird durch eine kleine, statisch durch den Typ des Ergebnisses bestimmte Anzahl von relationalen Queries implementiert, die nicht von der Größe der Datenbank abhängt.

The DSH library [1] has primarily been described in [3] and is covered in a number of other publications, e.g. [4, 5, 2]. The presentation will briefly sketch the current state of query libraries for Haskell and then give an example-driven introduction to the DSH library.

References

- [1] Database Supported Haskell. <http://hackage.haskell.org/package/DSH>.
- [2] George Giorgidze, Torsten Grust, Iassen Halatchliyski, and Michael Kummer. Analysing the Entire Wikipedia History with Database Supported Haskell. In *Proc. PADL*. ACM, 2013.

- [3] George Giorgidze, Torsten Grust, Tom Schreiber, and Jeroen Weijers. Haskell Boards the Ferry: Database-Supported Program Execution for Haskell. In *Proc. IFL*. Springer, 2010.
- [4] George Giorgidze, Torsten Grust, Nils Schweinsberg, and Jeroen Weijers. Bringing Back Monad Comprehensions. In *Proc. Haskell Symposium*. ACM, 2011.
- [5] George Giorgidze, Torsten Grust, Alexander Ulrich, and Jeroen Weijers. Algebraic data types for language-integrated queries. In *Proc. DDFP*. ACM, 2012.