

Programming Languages and Compiler Construction

The research group „Programming Languages and Compiler Construction“ is interested in the design, implementation and application of programming languages intended to support the reliable implementation of complex systems. The research ranges from object-oriented design methods and the analysis of concurrent and distributed systems to the implementation and application of declarative programming languages, in particular, in the area of web-based systems.

During the period reported below, the research group worked on a new advanced implementation of the multi-paradigm language Curry and achieved, partially in collaboration with the Portland State University (Oregon, USA), new research results related to the application of declarative programming methods.

Results

The scientific work of the research group involved all areas related to declarative programming languages, e.g., design, semantics, implementation, development tools, and application of such languages. Declarative programming languages are based on clear mathematical foundations. They abstract from the underlying computer architecture, and thus provide a higher programming level leading to more reliable systems. In particular, much of the research is focused on the integration of the most important declarative programming paradigms: functional and logic programming.

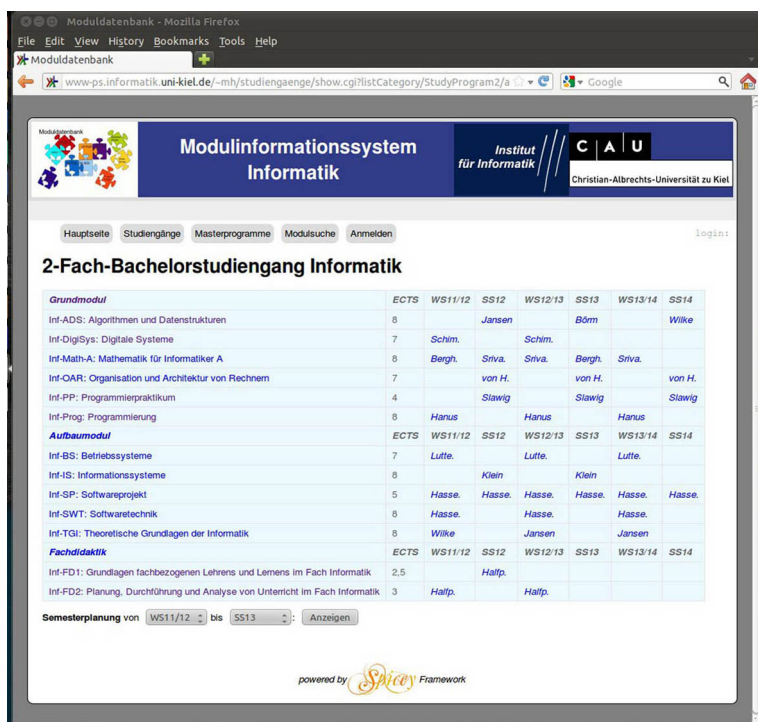


Fig. 1: The interface to the module database of the Institute of Computer Science

Related to the *application* of declarative languages, we worked on the design and implementation of a new web framework, called Spicey. Using Spicey, it is possible to generate a complete web application from a specification of the underlying complex data model in a few minutes. Since the generated implementation is a high-level declarative program, it is easy to adapt this to various customer requirements. In contrast to other web frameworks, ours exploits high-level declarative programming techniques, so that it yields reliable implementations that avoid data inconsistencies at various levels. We applied this system to implement a web-based interface to access the module database of the Institute of Computer Science which is used to plan the curriculum; it was quite positively evaluated by an accreditation committee.

Another application of declarative languages has been implemented together with Steffen Mazanek from the Universität der Bundeswehr (Munich). We developed a bidirectional transformation between BPMN (Business Process Modelling and Notation) and BPEL (Business Process Execution Language), two formalisms to deal with the management of business processes. Since our transformation from BPMN to BPEL is implemented in a functional logic language, the backward transformation from BPEL to BPMN is also obtained, so that one can work on both representations of business processes with automatic synchronization.

The application of functional logic languages has resulted in new design and programming patterns which exploit specific features of such languages. Hence, these patterns are not applicable to other kinds of programming languages. In order to support a systematic classification of such patterns, we collaborated with Prof. Antoy, Portland State University (Oregon, USA). We proposed a catalogue of design patterns together with typical application scenarios.

One of these design patterns has been exploited to provide a high-level processing of semistructured data that is frequently available as web data (in XML format). Due to the use of a modern multi-paradigm declarative language, it is possible to provide a framework to query and process such data in a few lines of code. Actually, the complete implementation is part of the corresponding scientific paper describing this approach, i.e. the source code of the publication is, at the same time, the program code used to perform such processing tasks. The code itself is not just a toy example but is used to integrate UnivIS data (from the XML-interface of the lecture information system of the university) into the module database of the institute.

In the area of *software technique* related to declarative languages, we collaborated with the Portland State University (Oregon, USA) and developed a transformation tool to support the development of reliable declarative programs written in the multi-paradigm programming language Curry. First, we set up a framework with precise notions of specifications, contracts, and assertions for declarative programs. Since specifications, contracts, assertions, and programs are written in the same language (Curry), Curry is used as a wide-spectrum language for software development. For instance, specifications are written in Curry and are, thus, executable. Hence, they can be used as an initial implementation. If this implementation is not efficient enough, a more efficient one can be developed, e.g., by using specific data structures and algorithms for the given problem domain. In this case one can use the initial specification as a contract or assertion to check the validity of the new implementation. This general idea is supported by a new tool (DSDCurry), which transforms, if necessary, specifications into implementations or assertions. Based on other work of our group on different methods for assertion checking, DSDCurry also support various kinds of assertion checks, like strict, lazy, or faithful assertions.

We also investigated several issues related to the *implementation* of functional logic programming languages. In particular, we developed a new implementation of Curry that is based on compiling Curry programs into purely functional Haskell programs. The advantage of this implementation is the explicit representation of non-deterministic computed results in the form of a search space. This provides for the application of various search strategies (depth-first, breadth-first, iterative deepening, parallel) to explore the search space. In addition to this flexibility, purely functional parts of declarative programs are compiled into Haskell programs almost without any overhead. Our first benchmarks show that this new implementation (KiCS2) is much faster than other existing Curry implementations for deterministic programs. For non-deterministic programs, KiCS2 can compete with or outperform other existing implementations of Curry. This implementation raised also some interest in the logic programming community so that we could present it as an invited talk at the 11th International Colloquium on Implementation of Constraint and Logic Programming Systems in Lexington (Kentucky, USA).



Personnel

Head of the group: Prof. Dr. Michael Hanus; Secretary: Ulrike Pollakowski
 Technical Staff: Dipl.-Ing. (FH) Thomas Heß



Fig. 2: Beginner students of the winter term 2011 (photo by Th. Wilke)

Scientific Staff:

M. SC. Björn Peemöller	01.01.-31.12.2011	CAU
Dipl.-Inf. Fabian Reck	01.01.-31.12.2011	CAU
Dr. Friedemann Simon	01.01.-31.12.2011	CAU

Lectures, Seminars, and Laboratory Course Offers

Winter 2010/2011

MSS0303: Masterabschlussseminar - Programmiersprachen, 2 hrs Seminar/Week,
Michael Hanus

Arbeitsgemeinschaft Informatik, Logik und Mathematik, 2 hrs Seminar/Week,
Michael Hanus (+ Rudolf Berghammer)

Inf-Prog: Programmierung, 4 (+ 2) hrs Lecture (+ Exercises)/Week,
Michael Hanus (+ Fabian Reck, Lars Prädel)

MS0303: Deklarative Programmiersprachen, 4 (+ 2) hrs Lecture (+ Exercises)/Week,
Michael Hanus (+ Björn Peemöller)

MS0306: Nebenläufige und verteilte Programmierung, 4 (+ 2) hrs Lecture (+ Exercises)/Week,
Frank Huch (+ Björn Peemöller, K. O. Kürtz)

Inf-ObjPro: Objektorientierte Programmierung, 2 (+ 2) hrs Lecture (+ Exercises)/Week,
Friedemann Simon

Inf-ObjProNF: Objektorientierte Programmierung (für Nebenfächler), 2 (+ 2) hrs Lecture (+ Exercises)/Week,
Friedemann Simon

Summer 2011

MSS0303: Masterabschlussseminar - Programmiersprachen, 2 hrs Seminar/Week,
Michael Hanus

Arbeitsgemeinschaft Informatik, Logik und Mathematik, 2 hrs Seminar/Week,
Michael Hanus (+ Rudolf Berghammer)

Inf-FortProg: Fortgeschrittene Programmierung, 3 (+ 2) hrs Lecture (+ Exercises)/Week,
F. Huch (+ Fabian Reck, Björn Peemöller, Th. Wilke)

NF-Inf-2: Systematisches Programmieren, 2 (+ 4) hrs Lecture (+ Exercises)/Week,

Friedemann Simon

NF-Inf-2-Phys: Systematisches Programmieren für Physiker, 2 hrs Lecture/Week,

Friedemann Simon

Winter 2011/2012

MSS0303: Masterabschlussseminar - Programmiersprachen, 2 hrs Seminar/Week,

Michael Hanus

Arbeitsgemeinschaft Informatik, Logik und Mathematik, 2 hrs Seminar/Week,

Michael Hanus (+ Rudolf Berghammer)

Inf-Prog: Programmierung, 4 (+ 2) hrs Lecture (+ Exercises)/Week,

Michael Hanus (+ Fabian Reck, Lars Prädell)

MS0302: Übersetzerbau, 4 (+ 2) hrs Lecture (+ Exercises)/Week,

Michael Hanus (+ Fabian Reck)

MS0304: Funktionale Programmierung, 4 (+ 2) hrs Lecture (+ Exercises)/Week,

Frank Huch (+ Björn Peemöller)

Inf-PtKI: Programmieretechniken für die Künstliche Intelligenz, 2 (+ 2) hrs Lecture (+ Exercises)/Week,

Friedemann Simon

NF-Inf-3: Programmieretechniken für die Künstliche Intelligenz für Nebenfächler, 2 (+ 2) hrs Lecture (+ Exercises)/Week,

Friedemann Simon

Further Cooperation, Consulting, and Technology Transfer

The research group collaborated with Portland State University (Sergio Antoy) and Universität der Bundeswehr (Steffen Mazanek) during the reported period.

Diploma, Bachelor and Master Theses

Catherine Antoniou, *A Client-Server Solution for Database Integration*, 01.02.2011

R. Gudschn, *Design-Aid for Graphical User Interfaces in Declarative Programming Languages*, 25.02.2011

J. Beck, *Transferbasierte Übersetzung vom Schwedischen ins Deutsche - implementiert in Prolog*, 30.09.2011

Dissertations / Postdoctoral Lecture Qualifications

B. Braßel, *Implementing Functional Logic Programs by Translation into Purely Functional Programs*, 20.10.2011

Publications

Published in 2011

S. Antoy, S. Hanus, *Contracts and Specifications for Functional Logic Programming*, Proceedings of the 21st Workshop on Logic-based Methods in Programming Environments (WLPE 2011), 47 - 61 (2011)

S. Antoy, M. Hanus, *New Functional Logic Design Patterns*, Proceedings of the 20th International Workshop on Functional and (Constraint) Logic Programming (WFLP 2011), Springer LNCS 6816, 19 - 34 (2011)

B. Braßel, S. Fischer, M. Hanus, F. Reck, *Transforming Functional Logic Programs into Monadic Functional Programs*, Functional and Constraint Logic Programming, Springer LNCS 6559, 30 - 47 (2011)

- B. Braßel, M. Hanus, B. Peemöller, F. Reck, *KiCS2: A New Compiler from Curry to Haskell*, Functional and Constraint Logic Programming, **Springer LNCS 6816**, 1 - 18 (2011)
- B. Braßel, M. Hanus, B. Peemöller, F. Reck, *Haskell for Logic Programmers: Compiling Functional Logic Programs to Haskell*, Proc. 11th International Colloquium on Implementation of Constraint and Logic Programming Systems (CICLOPS 2011), 1 - 18 (2011)
- M. Braßel, M. Hanus, B. Peemöller, F. Reck, *Implementing Equational Constraints in a Functional Language*, Proc. 19th International Conference on Applications of Declarative Programming and Knowledge Management (INAP 2011) and 25th Workshop on Logic Programming (WLP 2011), 22 - 33 (2011)
- M. Hanus, *Lazy and Enforceable Assertions for Functional Logic Programs*, Functional and Constraint Logic Programming, **Springer LNCS 6559**, 84 - 100 (2011)
- M. Hanus, *Declarative Processing of Semistructured Web Data*, Technical Communications of the 27th International Conference on Logic Programming (ICLP 2011), **LIPICs Vol. 11**, 198 - 208 (2011)
- M. Hanus, P. Scheider-Kamp, *Principles and Practice of Declarative Programming*, ACM Press, (2011)
- M. Hanus, *Declarative Processing of Semistructured Web Data*, IfI, CAU Kiel, **Research Report 1103**, (2011)
- M. Hanus, S. Koschnicke, *An ER-based Framework for Declarative Web Programming*, CoRR, **abs/1103.4133**, (2011)
- S. Mazanek, M. Hanus, *Constructing a Bidirectional Transformation between BPMN and BPEL with a Functional Logic Programming Language*, Journal of Visual Languages and Computing, **Vol. 22, No. 1**, 66 - 89 (2011)
- B. Braßel, S. Fischer, *From Functional Logic Programs to Purely Functional Programs Preserving Laziness*, 20th International Symposium on Implementation and Application of Functional Languages, **Springer LNCS 5836**, 25 - 42 (2011)

Presentations

- M. Hanus, *From Haskell to Curry and back to Haskell*, Portland State University, Portland, Oregon (USA), 28.04.2011
- Björn Peemöller, *KiCS2: A New Compiler from Curry to Haskell*, 28. Workshop GI-FG Programmiersprachen und Rechenkonzepte, Bad Honnef, 04.05.2011
- M. Hanus, *KiCS2: A New Compiler from Curry to Haskell*, pdxfunc meeting, Portland, Oregon (USA), 13.06.2011
- M. Hanus, *Declarative Processing of Semistructured Web Data*, 27th International Conference on Logic Programming, Lexington, Kentucky (USA), 09.07.2011
- M. Hanus, *Haskell for Logic Programmers: Compiling Functional Logic Programs to Haskell*, 11th International Colloquium on Implementation of Constraint and Logic Programming Systems (invited talk), Lexington, Kentucky (USA), 10.07.2011
- M. Hanus, *Contracts and Specifications for Functional Logic Programming*, 21st Workshop on Logic-based Methods in Programming Environments, Lexington, Kentucky (USA), 10.07.2011
- B. Peemöller, *KiCS2: A New Compiler from Curry to Haskell*, 20th International Workshop on Functional and (Constraint) Logic Programming, Odense, Denmark, 19.07.2011
- M. Hanus, *New Functional Logic Design Patterns*, 20th International Workshop on Functional and (Constraint) Logic Programming, Odense, Denmark, 19.07.2011
- Fabian Reck, *Implementing Equational Constraints in a Functional Language*, 16. Kolloquium Programmiersprachen und Grundlagen der Programmierung (KPS'11), Raesfeld Castle, Germany, 27.09.2011
- M. Hanus, *Implementing Equational Constraints in a Functional Language*, 19th International Conference on Applications of Declarative Programming and Knowledge Management and 25th Workshop on Logic Programming, Vienna, Austria, 28.09.2011

Further Activities and Events

M. Hanus:

programme committee member of TFP 2011 (12th International Symposium on Trends in Functional Programming), Madrid (Spain), May 2011,

programme committee member of the CICLOPS 2011 (11th International Colloquium on Implementation of Constraint and Logic Programming Systems), Lexington (Kentucky, USA), July 2011,

programme committee member of WFLP 2011 (20th Workshop on Functional and (Constraint) Logic Programming), Odense (Denmark), July 2011,

programme committee chair of PPDP 2011 (13th International Symposium on Principles and Practice of Declarative Programming) Odense (Denmark), July 2011,

programme committee member of WLP 2011 (25th Workshop on (Constraint) Logic Programming), Vienna (Austria), September 2011,

member of the Editorial Board of the Journal of Functional and Logic Programming,

member of the steering committee of the Symposia on Logic-based Program Synthesis and Transformation,

member of the steering committee of the ACM SIGPLAN Symposia on Principles and Practice of Declarative Programming,

member of the executive committee and vice-chair of the GLP (Gesellschaft für Logische Programmierung), German-speaking branch of the Association for Logic Programming (ALP),

member of the advisory board of the GLP (Gesellschaft für Logische Programmierung), German-speaking branch of the Association for Logic Programming (ALP),

member of the selection committee of the DAAD (German Academic Exchange Service) for the project-related support to scientific cooperation with Spain and Portugal,

member of the advisory board of the „Berufsakademie an der Wirtschaftsakademie Schleswig-Holstein“ ,

chair of the examinations board of computer science studies, University of Kiel,

member of the Senate Curriculum Committee, University of Kiel,

vice-member of the Senate Equal Opportunities Committee, University of Kiel,

research sabbatical at Portland State University (Oregon, USA) with Prof. Sergio Antoy, March-June 2011,

invited talk at CICLOPS 2011 (11th International Colloquium on Implementation of Constraint and Logic Programming Systems), Lexington (Kentucky, USA), July 2011.

F. Huch:

programme committee member of TFP 2011 (12th International Symposium on Trends in Functional Programming), Madrid (Spain), May 2011,

programme committee member of LOPSTR 2011 (21st International Symposium on Logic-Based Program Synthesis and Transformation), Odense (Denmark), July 2011,

organization of the 28th Workshop of the GI-Fachgruppe Programmiersprachen und Rechenkonzepte, Bad Honnef (Germany), May 2011,

member of the Steering Committee of Symposia on Implementation and Application of Functional Languages (IFL),

chair of the executive committee of the Fachgruppe „Programmiersprachen und Rechenkonzepte“ of the Gesellschaft für Informatik e.V.

B. Peemöller:

research stay at the National Institute of Informatics, Tokio, Japan, September 12-25, 2011,

attendance and administrative support of the International Conference on Functional Programming (ICFP 2011) and related workshops (WGP, Haskell Symposium, Haskell Implementor's Workshop, CUF), Tokio, Japan, September 2011.

F. Simon:

participation in seminars for students planning professional careers, „Computer Museum“ , representative of the Faculty of Engineering in the board of control.