PERSONAL STATEMENT OF RESEARCH ACCOMPLISHMENTS AND FUTURE DIRECTIONS

Mark Bailey
Department of Computer Science
University of Virginia

In graduate school, I have focused on retargetable systems software and programming languages. My research spans several fields, including compilers, optimization, computer architecture, testing, formal specification, and systems software. I continue to have interests in these fields, but my future research will not be limited to these areas. Below is a summary of my previous work, and my plans for future research.

My dissertation research concerns the specification of computer hardware for use in retargetable systems software such as compilers, linkers, assemblers, debuggers and simulators. The proliferation of microprocessors in recent years has made the development of such software more challenging than ever. The implementation of these applications is already difficult and time consuming. Consequently, it is essential that design efforts produce applications that are portable to amortize their high development costs across many hardware platforms.

Portability has two aspects: rehostability and retargetability. My work focuses on retargetability. Systems software requires information about the host computer system such as the machine’s instruction set, pipeline structure, timing information, and memory system. Previous work in the field mostly addressed the instruction set descriptions that are used to automatically generate code generators for compilers. These solutions are limited to use by a single compilation system and typically only specify one aspect of the computer system (e.g., instructions). My work has focused on the design and implementation of a general framework for building comprehensive computer system descriptions that can be used by a variety of systems software. By building these specifications in a reusable manner, we can share both the specifications and the tools that process them with many applications.

My description framework, called CSDL (Computer Systems Description Language), is an extensible system that provides a core description and facilities for applications to include both application-specific, and application-independent information about the target machine. CSDL manages the descriptions and provides a mechanism for building custom specifications for the client application. The core description is based on RTL’s (register transfer lists) from Davidson’s Portable Optimizer (PO). PO is the basis for the underlying technology for the GNU compilers. A paper in the *Workshop on Compiler Support for System Software* presents an overview of CSDL. This work is also a key component of an ARPA funded joint-project with Princeton, Virginia, and Stanford to build a national compiler infrastructure for use by the compiler research community.

To illustrate the usefulness of CSDL, I have built several extensions. Of particular note is a specification language for procedure calling conventions called CCL. Calling conventions, which are typically mixes of hardware and software conventions, vary drastically from machine to machine. My specification language captures the details of calling conventions in an application-independent way. Papers that describe the uses of CCL have been published in the *Proceedings of the 1995 ACM Symposium on Principles of Programming Languages*, and the *Proceedings of the 1996 ACM Conference on Programming Language Design and Implementation*. We have also submitted a journal article on CCL and its applications to *ACM Transactions on Programming Languages and Systems*.

For my master’s degree, I developed a tool for use in exploring programming language design. The tool, called FLECS, facilitates rapid prototyping of control structures in success/failure based languages. Control structures are modeled in Scheme using a continuation-passing-style interpreter. Other work involved the addition of object-oriented features to the Icon programming language.
I anticipate that, initially, my future research will extend my dissertation research. The CSDL framework presents unique opportunities for exploration of specifications of all aspects of computer systems. In many cases, such extensions are particularly amenable to undergraduate research projects and theses. There may also be opportunities for students to join the national compiler infrastructure project. Because of the applicability of my research to industry, I anticipate internship opportunities for my students as well as industrial funding for projects. I already have some contacts in industry and will continue to foster them. Finally, I would also like to see my future research include problems in other fields, such as issues in high-level language implementation, and I would like to pursue interdisciplinary projects with other departments.