ABSTRACT

Ph.D. thesis: A Domain-Specific Language for Protocol Stack Implementation in Embedded Systems

Author: Yan Wang

Embedded network software has become an area of increasing importance for both research and industry as more and more applications are built on networked embedded systems. Modern devices and applications require newly designed or revised protocols which have to be implemented. Also, well-known infrastructure protocol stacks have to be reimplemented on new hardware platforms and software architectures. However, implementing protocol stacks for embedded systems remains a time-consuming and error-prone task due to the complexity and performance-critical nature of network software. It is even more so when targeting resource constrained embedded systems, as implementations also have to minimize energy consumption and meet memory constraints.

This thesis addresses how to facilitate protocol stack implementations for embedded systems and how to determine and control their resource consumption by means of a language-based approach. It aims at a domain-specific language (DSL) that supports abstractions suitable for the implementation of protocol stacks. Language technologies in the form of a type system, a runtime system and compilation can then be used to generate efficient implementations.

In the work presented in this thesis, we give background on DSL implementation techniques. We also investigate common practices in network protocol development to determine the potential of DSLs for embedded network software. Finally, we propose a domain-specific embedded language (DSEL), Protege (Protocol Implementation Generator), for declaratively describing overlaid protocol stacks. In Protege, a high-level packet specification is dually compiled into an internal data representation for protocol logic implementation, and packet processing methods which are then integrated into the dataflow framework of a protocol overlay specification. The Protege language offers constructs for finite state machines to specify protocol logic in a concise manner, close to the protocol specification style. Protege specifications are compiled to highly portable C code for various architectures.

Four attached papers report our main results in more detail: an embedded implementation of the data description calculus in Haskell, a compilation framework for generating packet processing code with overlays, an overview of the domain-specific language Protege, including embedding techniques and runtime system features, and a case study implementing an industrial application protocol.

Contact:

Yan Wang
Tel: +46 35 167464
E-mail: yan.wang@hh.se

Halmstad university, School of Information Science, Computer and Electrical Engineering
Box 823, SE-301 18 Halmstad, Sweden