Incremental declaration handling in Open Source Modelica

Peter Aronsson, Peter Fritzson, Levon Saldamli, Peter Bunus
Dept. of Computer and Information Science,
Linköping University, SE-581 83 Linköping, Sweden
(petar,petfr,levsa,petbu)@ida.liu.se

15 April 2002

Abstract

This paper presents the recent development in the Open Source Modelica project started at the Programming Environments Laboratory (PELAB) at Linköping University. The goal of this project is to develop a freely available Open Source implementation of a Modelica compiler to a large extent based on a formal specification of Modelica.

Current activities in the Open Source Modelica project, besides upgrading the specification to the latest version of the Modelica language specification (Modelica 2.0), is to adapt the compiler for working in an incremental modeling and simulation environment which is also capable of performing scientific calculations in an efficient way. Such an environment will be built within the scope of the Open Source Modelica project combined with other research activities, like debugging Modelica code and adding Partial Differential Equation support to Modelica, taking place at the Programming Environments Laboratory at Linköping University.

The purpose of having an interactive and incremental modeling, simulation and computation environment is to introduce and strengthen the incremental development process in the field of modeling and simulation. In an incremental development process, the software is evolved gradually and interactively, hence ensuring the quality of the software by performing interactive tests as the software evolve.

For support of such an environment, there is a need for an incremental and interactive symbol table in the Open Source Modelica compiler. This symbol table contains model definitions with their component declarations, equations, and algorithms, i.e. all information that is needed to produce the flat set of equations and algorithms from a Modelica model. This incremental symbol table should have support for adding new model definitions, retrieval of definitions (both Modelica definitions and the flat definitions, i.e. the flat set of equations), etc. In this paper, we present an initial design of this symbol table written in RML. RML is a language for writing natural semantics that allows for efficient code generation, making it possible to generate an efficient compiler from a specification (written in RML).

In conclusion, the Open Source Modelica project will with its incremental and interactive environment make it possible for engineers to work in an iterative and efficient way when building models, performing model refinements and simulation experiments. Thus, the effort needed to build models and implement scientific calculations (in the form of Modelica functions) will decrease with the use of such an incremental and interactive environment.