Generalization of an Active Electronic Notebook for Teaching Multiple Programming Languages

Mohsen Torabzadeh-Tari, Peter Fritzson, Adrian Pop, Martin Sjölund
PELAB – Programming Environment Lab, Dept. Computer Science
Linköping University, SE-581 83 Linköping, Sweden
{mohto, petfr, adropo, marsj}@ida.se

Abstract — In this paper we present a generalization of the active electronic notebook, OMNotebook, for handling multiple programming languages for educational purposes. OMNotebook can be an alternative or complementary tool to the traditional teaching method with lecturing and reading textbooks. Experience shows that using such an electronic book will lead to more engagement from the students. OMNotebook can contain technical computations and text, as well as graphics. Hence it is a suitable tool for teaching, experimentation, simulation, scripting, model documentation, storage, etc.

OMNotebook is part of the open source platform OpenModelica. It is already used for the course material DrModelica in teaching the Modelica language but can easily be adapted to other programming languages which is also shown in this paper. The notebook can also be adapted to other areas, such as physics, chemistry, biology, biomechanics etc., where phenomena can be illustrated by dynamic simulations within the notebook.

The idea behind this paper is to show that by using a standardized interface the notebook can be extended to any computer language, i.e., being language independent. This is shown in the form of an implementation and adaptation of the notebook to support the Scheme language.

Keywords- OpenModelica, Scheme, DrModelica, Electronic Notebook

I. INTRODUCTION

In this paper we introduce a research project for generalizing the modern object-oriented equation-based modeling and simulation environment OpenModelica towards also supporting other programming languages than Modelica, [1]. In this way the active student has a common platform for learning programming languages as well as given the opportunity to experiment with physical phenomena by using interactive electronic book, OMNotebook, [2].

This kind of interactive courses based on electronic books allows experimentation and dynamic simulation as well as execution of computer programs. The OMNotebook can contain program code, text, links, pictures, video, virtual and scientific visualizations. OMNotebook is an active electronic book that makes it is possible to integrating applied sciences in physics, human biology [3], mathematics and computer science.

This work was supported by the EU project Lila and by the ITEA2 OPENPROD project.
formalism for compiler semantics. This formalism allows efficient compilation combined with optimized C code. This was later (2006) replaced by an extension to Modelica itself, MetaModelica, and the whole compiler was migrated to MetaModelica.

The OpenModelica environment compiler translates the Modelica model into a flat Modelica code first and then into C code after a couple of more steps. Also an interactive command handler, i.e., a shell, for executing Modelica scripts and functions etc., is also provided in the environment.

The OpenModelica environment, shown in Fig. 1, consists of several interconnected subsystems. The debugger currently provides debugging of an extended algorithmic subset of Modelica, MetaModelica.

The interactive session handler interface is used in this paper for communication between the OMNotebook and the Scheme interpreter.

C. OMNotebook – the Active Electronic Notebook

The OpenModelica Notebook editor, OMNotebook, provides an active electronic notebook including an editor. The notebook is active in the sense that models inside the book can be changed and executed, it is not just a passive textbook or html page. This is one of the first open source efforts that makes it possible to create interactive books for educational purposes in general, and more specifically for teaching and learning programming. This functionality allows the usage of interactive hierarchical text documents where underlying chapters and sections can be represented and edited. OMNotebook supports functionality for Modelica model simulations, text, images and interactive linking between those. Furthermore, via the external interface, Scheme programs and other codes can be evaluated.

The notebook is currently being used for course material (DrModelica) in teaching the Modelica language and object-oriented modeling and simulation, (see Fig 2), but can easily be adapted to electronic books teaching other programming languages which is demonstrated in this paper. OMNotebook can also easily be used in other areas such as physics, biology chemistry, biomechanics etc., where phenomena can be illustrated by dynamic simulations within the book.

Traditional teaching methods with lecturing and reading a textbook are often too passive and don’t engage the student. The option presented in this paper with an active notebook, however, facilitates the learning process, e.g. to run programs and exercises within the book, and mix lecturing with exercises and reading in the interactive book.
on different tool and technology but instead focus on the information. An implementation of the factorial function using OMScheme is shown in Fig 3.

D. PLT DrScheme

The PLT Scheme is a dialect of the Scheme programming language, [5]. MzScheme is the interpreter behind the PLT Scheme for compiling syntactically valid programs into an internal bytecode representation before evaluation. The graphical user interface toolkit is named MrEd. DrScheme is an integrated development environment based on MzScheme, i.e., a MrEd application, with support for embedding third-party extensions. DrScheme provides developers with modular development tools, e.g. syntax or flow analyzers. The C API provided by this environment is embedded in OMNotebook and used in this paper for evaluating Scheme programs.

III. CONCLUSIONS

In this paper we outline the basic ideas of an active notebook for educational purpose intended for handling multiple programming languages. An early prototype is being developed for the Lisp dialect Scheme. This interactive E-book, OMNotebook, has been used successfully in both graduate and workshop courses for the Modelica language. OMScheme takes this idea further, combining an additional programming language with innovative teaching concepts.

ACKNOWLEDGMENT

This work was supported by the EU project Lila and the ITEA2 OPENPROD project.

REFERENCES