

Diploma thesis about  
Enhancing and Improving the Capabilities of the TDL Editor Suite

Developing TDL applications is supported by several tools that form up the so-called TDL toolchain. The *TDL editor suite* is one of those tools. A TDL application usually consists of several TDL modules that are defined by a textual representation of the TDL modules that has a certain syntax. This textual representation is called TDL code. By contrast, the major task of the TDL editor suite is to define TDL modules by using a graphical user interface which eases the development of TDL modules.

The TDL editor suite is well-integrated into Simulink. Simulink is the de-facto standard modeling tool widely used in the industry. Additionally, the suite supports code generation (i.e. generating of TDL code and controlling Simulink's Real-time Workshop Embedded Coder for generating functionality code, e.g. task functionality). It is implemented in Java and its implementation principles are heavily based on the MVC pattern as well as on a plug-in architecture. The latter is used to support different platforms for which code has to be generated.

Although many features are already included in the current version of the TDL Editor suite, various usability aspects need improvements. This diploma thesis, which consists of two parts, addresses these issues.

The TDL editor suite consists of several editor types that define different aspects of a TDL module. The major enhancement of the TDL editor suite, as the first part of the thesis, is the implementation of an additional editor type, the so-called *platform editor*. Although it is possible in the current implementation to create platforms (i.e. nodes a distributed system consists of) a module should run on, there is no comfortable way to do this in a graphical style. Thus the user should be able to graphically define the topology and characteristics of a distributed system first. Then the user should be able to assign TDL modules to the nodes of the system. The platform editor should possess the ability to initiate the required code generation and compilation steps for the whole system. A case study should be implemented for a specific hardware platform provided by the department of computer science.

The second part of the thesis is an analysis and improvement of the HCI capabilities of the existing TDL editor suite. Based on the results of the analysis suggestions of improvements have to be worked out and those improvements have to be implemented too.

Advisor: Univ.-Prof. DI Dr. Wolfgang Pree

Assistant advisors: DI Gerald Stieglbauer, DI Dr. Josef Templ

Diploma student: Martin Hoeggerl

E-mail: martinhoeggerl@gmx.net

Matrikelnummer: 0120076

Address: Mozartstraße 64, 9020 Klagenfurt

Phone number: 0676 5998685

Start of the thesis: 01.08.2005

End of the thesis: Spring 2006