

DSL4CS: Co-simulation Language for Heterogeneous CPS based on Gemoc*

Abstract—Domain specific language is ... co-simulation: We propose a co-simulation language, introduce the metamodel, concrete syntax and semantics. We implement it based on Gemoc framework.

Index Terms—co-simulation, DSL, metamodel, Gemoc, CPS

I. Introduction

II. Co-simulation Language for Heterogeneous CPS (DSL4CPS)

The design of complex systems often relies on several Domain Specific Modeling Languages (DSMLs) that may pertain to different theoretical domains with different expected expressiveness and properties. As a result, several models conforming to different DSMLs are developed and the specification of the overall system becomes heterogeneous.

Meta-models, as defined here, provide: A language representation with strict notation and grammar. This is used to represent and store the meta-model and to share meta-model information between different tools and people. Model abstraction, where each external model is represented individually in a uniform and simulation tool independent way. A generic and uniform way to connect various simulation tools. Platform independent models. No operating system, network, or other co-simulation platform dependent parameters are stored in the meta-model. A single meta-model simulator application can start the simulation tools and control the co-simulation based on the meta-model.

A. Meta Model for DSL4CPS

B. Concrete Syntax for DSL4CPS

rules, director, coordinate pattern

C. Semantics for DSL4CPS

extend Gemoc to support DSL4CS

III. Extend Gemoc to Support DSL4CPS

A. Design Meta Model Using EMF

B. Generate Concrete Syntax Based on Xtext

C. Define Semantics with CCSL

IV. Implementation and Case Study

A. Tool Implementation based on Gemoc

B. Case Study

V. Related Works

VI. Conclusions and Acknowledgement

References