AUTO-GENERATING MODELS FROM THEIR SEMANTICS AND CONSTRAINTS

Tanumoy Pati (James H. Hill) Department of Computer and Information Science, Purdue School of Science, Indiana University–Purdue University Indianapolis, Indianapolis, Indiana 46202

Model-Driven Engineering (MDE) facilitates building solutions in many enterprise application domains through the systematic use of graphical languages called domain-specific modeling languages (DSMLs). MDE tools, such as the Generic Modeling Environment (GME) and the Generic Eclipse Modeling System (GEMS), enable end-users to rapidly create such custom DSMLs. When DSMLs are coupled with constraint solvers, it is possible for DSML end-users to auto-generate solutions (*i.e.*, models) based on the goals of the constraint solver (*e.g.*, finding the optimal deployment for a set of software components given resource availability and resource needs). One requirement of using a constraint solver with a DSML, however, is that modelers have to create an initial model, also known as a “partial model”. This implies that it is the end-users responsibility to (1) understand how to use the DSML and (2) understand when they have created an appropriate partial model that is a candidate for completion using a constraint solver.

Our research therefore focuses on addressing the two problems mentioned above. We address the problems by analyzing the semantics and constraints of the DSML (*i.e.*, the meta-model). Based on our analysis, we then auto-generate as much of the model until we reach a point that requires user intervention. At that point, we present a set of operations (or moves) to the user and continue the process until the model is complete, or is solvable using a constraint solver.