Model Transformation and Code Generation using Verified Requirements

Prof. Nabendu Chaki
University of Calcutta, India

Model transformation represents the daunting challenge of converting higher-level abstract models to platform-specific implementation models. These, in turn, may be used for automated code generation. The importance of transforming a goal model to a finite state model lies in the effort to perform model checking effectively on goal models! Model checking helps requirement analysts to verify whether goal models comply with system regulations and standards. Standard model checkers accept extended finite state models as input and verify temporal properties. Transforming finite state models to other sequential models should be easier than transforming a sequence agnostic goal model. We have used goal models to identify and detect errors, conflicts, or issues that are often handled in the later phases of the lifecycle. The main motivation behind this research is to help developers identify, analyse and rectify errors in the requirements phase itself and to automate code generation from there. Thus, the source code, generated using automation from the verified and validated set of requirements, would be free from errors and conflicts that are negotiated at the early stage itself. We also propose techniques to leverage greater benefits from goal models by extending their analytical capabilities.