???!

Van Cam Pham, Ansgar Radermacher, Sebastien Gerard

CEA-List, Laboratory of Model-Driven Engineering for Embedded Systems (LISE)

Gif-sur-Yvette, France

Email: first-name.lastname@cea.fr

Abstract—UML state machine and their visual representations are much more suitable to describe logical behaviors of system entities than any equivalent text based description such as IF-THEN-ELSE or SWITH-CASE constructions. Although many industrial tools and research prototypes can generate executable code from such graphical language, generated code could be manually modified by programmers. After code modifications, round-trip engineering is needed to make the model and code consistent, which is a critical aspect to meet quality and performance constraint required from project manager today. Unfortunately, current UML tools only support structural concepts for round-trip engineering such as those available from class diagrams. In this paper, we address the round-trip engineering of UML state-machine and its related generated code. We propose a round-trip engineering approach consisting of a forward process which generates code by using transformation patterns, and a backward process which is based on code pattern detection to update the original state machine model from the modified generated code. We implemented a prototype and conducted several experiments on different aspects of the round-trip engineering to verify the proposed approach.

I. Introduction

II. Approach

State

III. Conclusion

Acknowledgment

This work is motivated by....