Chapter 17
Specification and Validation of Real Time Systems

Olfa Mosbahi
Campus Universitaire, Tunisia

ABSTRACT

The chapter presents a specification technique borrowing features from two classes of specification methods, formal and semi-formal ones. Each of the above methods have been proved to be useful in the development of real-time and critical systems and widely reported in different papers (Bruel, 1996; Clarke & Wing, 1996; Cohen, 1994; Fitzgerald & Larsen, 1994; Ghezzi, Mandrioli & Morzenti, 1990). Formal methods are based on mathematical notations and axiomatic which induce verification and validation. Semi-formal methods are, in the other hand, graphic, structural and user-friendly. Each method is applied on a suitable case study, that we regret some missing features we could find in the other class. This remark has motivated our work. We are interested in the integration of formal and semi-formal methods in order to lay out a specification approach which combines the advantages of these two classes of methods. The proposed technique is based on the integration of the semi-formal method STATEMATE (Harel, 1997; Harel, 1987) and the temporal logic FNLOG (Sowmya & Ramesh, 1997). This choice is justified by the fact that FNLOG is formal, deals with quantitative temporal properties and that these two approaches have a compatibility which simplifies their integration (Sowmya & Ramesh, 1997). The proposed integration approach uses the notations of STATEMATE and FNLOG, defines various transformation rules of a STATEMATE specification towards FNLOG and extends the axiomatics of the temporal logic FNLOG by new lemmas to deal with duration properties. The chapter presents the various steps of our integration approach, the proposed extensions and illustrates it over a case of critical real-time systems: the gas burner system (Ravn, Rishel & Hansen, 1993).
INTRODUCTION

Critical real-time systems are complex and require a high level of safety and reliability. To reduce this complexity and to reach a high required degree of reliability and safety, it would be quite interesting to lay out a specification approach which simplifies the requirement description, deals with mathematical notations inducing verification and validation, and allows the description of quantitative temporal properties. Thus, it comes the idea of integrating formal (Bruel, 1996; Bus sow, Wolfgang & Grieskamp, 1998; Clarke & Wing, 1996; Jahanian & Mok, 1996; Ostroff, 1994) and semi-formal approaches in order to lay out a specification approach which combines the advantages of these two classes of methods. Semi formal methods are graphic, structural and user-friendly; formal methods are based on mathematical notations and axiomatic inducing proofs. In this paper, we propose a specification technique integrating STATEMATE (Harel, 1987; Harel, 1997) as a semi-formal method and the temporal logic FNLOG (Sowmya & Ramesh, 1997) as a formal one. Several reasons justify the choice of these methods. STATEMATE (Harel, 1987) is a graphic formalism; covers the various aspects of a complex system (data, functionality, control and structure); deals with a tool allowing the checking of syntax and the validation by simulation. The temporal logic FNLOG (Sowmya & Ramesh, 1997) provides a requirement specification language that allows a concise expression of quantitative properties. The proposed specification and validation approach introduces an integration method using STATEMATE and FNLOG notations and proposes transformation rules, and an extension of FNLOG axiomatic to reason about duration properties.

The chapter is organized as follows: section 2 presents a background for the used methods: STATEMATE and FNLOG. The reader interested by more details can be referred to papers (Harel, 1987; Harel, 1997; Harel, Pnueli, Schmidt & Sherman, 1987) for STATEMATE and to papers (Sowmya & Ramesh, 1997; Sowmya & Ramesh, 1992) for FNLOG. Section 3 presents the proposed method combining STTEMATE and FNLOG. Section 4 presents a case study illustrating the proposed method. We end by a conclusion and a future work.

BACKGROUND

General View STATEMATE

STATEMATE is a graphic specification method for complex real-time reactive systems. In this section, we present a formal description of real-time reactive systems, an informal description of STATEMATE as a specification method. In STATEMATE, the descriptions used to capture the system specification are organized into three views, or projections, of the system: functional, behavioral and structural. The functional view describes the system’s functions, processes and activities. The behavioral view describes the system’s behavior over time, including the dynamics of activities, their control and timing behavior, the states and modes of the system, and the conditions and events that
Related Content

Legacy Systems towards Aspect-Oriented Systems
[www.igi-global.com/chapter/legacy-systems-towards-aspect-oriented-systems/135231?camid=4v1a](www.igi-global.com/chapter/legacy-systems-towards-aspect-oriented-systems/135231?camid=4v1a)

Empowering Microfinance Processes through Hybrid Cloud Based Services
[www.igi-global.com/article/empowering-microfinance-processes-through-hybrid-cloud-based-services/134431?camid=4v1a](www.igi-global.com/article/empowering-microfinance-processes-through-hybrid-cloud-based-services/134431?camid=4v1a)

Knowledge Integration in Problem Solving Processes: A Case Study - Perceptions of Workers
[www.igi-global.com/article/knowledge-integration-in-problem-solving-processes/119656?camid=4v1a](www.igi-global.com/article/knowledge-integration-in-problem-solving-processes/119656?camid=4v1a)

Using the Cloud for Testing NOT Adjunct to Development
[www.igi-global.com/chapter/using-cloud-testing-not-adjunct/72233?camid=4v1a](www.igi-global.com/chapter/using-cloud-testing-not-adjunct/72233?camid=4v1a)