Towards an Integration of Standard Component-Based Safety Evaluation Techniques with SaveCCM

Lars Grunske

School of Information Technology and Electrical Engineering
ARC Centre for Complex Systems,
University of Queensland,
4072 Brisbane (St.Lucia), Australia
grunske@itee.uq.edu.au

Abstract. To deliver complex functionalities in a cost effective manner, embedded software should ideally be developed with standardized interoperable components. At the same time, most of these embedded systems must be demonstrably safe and reliable. This paper aims to extend SaveCCM, a modelling language for component-based embedded systems, with standard safety evaluation models. Based on this extension, failure and hazard probabilities can be estimated early in the development process and can be used to check if a system can fulfil its safety requirements. The procedure of the safety evaluation is demonstrated with the case study of a computer assisted braking system.

1 Introduction

Modern safety-critical real-time systems in various application domains, such as automotive, avionic, defence and medical systems, are becoming increasingly complex ensembles of hardware and software components. Design and development of these complex component-based systems including their architectures is challenging, because systems and software engineers need to deal with strict non-functional requirements, such as safety, availability, reliability, performance, memory consumption and real-time requirements [1,2], while keeping development and life-cycle costs low and practicable. Therefore, component-based software engineering technologies in these domains must be capable of predicting dependability attributes of a system assembled from components. Currently, several component-based modelling languages and component frameworks aim to solve this problem. Examples are the PECT (Prediction-enabled Component Technology) [3] initiative of the Software Engineering Institute at the Carnegie Mellon University, the Ptolemy II project [4] of the University of California at Berkeley and the KOALA component model [5], which is used at Philips for embedded software in consumer electronic devices.

This paper focuses on SaveComp [6]; a recent component-based development framework for embedded control applications in automotive (vehicular) systems. This framework is based on a control flow paradigm and its basic aim is to create predictable component-based systems. The underlying formalism for the construction of SaveComp systems is the architecture description language SaveCCM (SaveComp Component Model). This language has a simple graphical syntax [6] and a formal semantics [7],

C. Hofmeister et al. (Eds.): QoSA 2006, LNCS 4214, pp. 199–213, 2006.

which uses the theory of timed automata [8]. Based on this semantics, system properties can be checked with the UPPAAL [9] model checker. Other prediction techniques [6,10] provide the ability to predict Worst-Case Execution Time (WCET) of components, Worst Case Response Time (WCRT) of an assembled system or resource utilization of a set of components. The early prediction of these correctness and real-time properties allows reducing the number of design failures in the software system. These design failures are also known as systematic failures, because they systematically occur during runtime, if the system is operated with a certain sequence of input requests. However, for a complete safety analysis, additionally random and wear-out failures of the hardware elements have to be considered. These failures occur stochastically distributed over the mission time of the system and they are generally quantified with probabilistic metrics [11], such as failure rates (λ), Mean Time To Failure ($MTTF$) or Mean Time Between Failures ($MTBF$).

To facilitate a quantitative safety analysis of component-based systems, encapsulated evaluation models, such as Component Fault Trees (CFT) [12] or State-Event Fault Trees (SEFT) [13] have been developed. Generally, an encapsulated evaluation model specifies all necessary information to reason about quality attributes independently from the deployment context and the environment of an architectural entity. In the case of safety evaluation, these encapsulated evaluation models describe possible failures of component services and enable the estimation of their failure probabilities. The overall goal of this paper is to identify if these models can be integrated within the SaveComp modelling framework. In detail, the contributions of this paper are as follows:

- Identify a suitable formalism for safety evaluation (failure specification and analysis) within SaveCCM
- Create a methodology to systematically generate safety evaluation models
- Validate the methodology with a complex case study
- Identify the limitation of this approach and proposed future research directions to overcome these limitations

This rest of this paper is structured as follows. Section 2 reviews related work and describes state-of-the-art techniques in the area of safety evaluation for component-based systems. The basic concepts of SaveCCM are introduced in Section 3. An integration of the standard safety evaluation techniques and SaveCCM is presented in Sections 4 and 5. The general aim of these sections is to describe how probabilistic failures can be specified and how failure propagation models can be generated. The presented approach will be validated in Section 6 with the case study of a computer assisted braking system. Finally, conclusions as well as relevant directions for future work are given in Section 7.

2 Related Work

The related work can be distinguished into two categories. The first category contains analysis techniques for quality attributes, which are preliminaries for safety evaluation and which have been already integrated with SaveCCM. The second category contains detailed approaches for safety evaluation for other component-based specification languages.