

Software Qualitätssicherung für Geschäftsprozesse mit eingebetteten Systemen

Padma Iyengar^{1,2}

¹Software Engineering Research Group, University of Osnabueck, Germany

²Institute of Computer Engineering, UAS, Osnabrueck, Germany
piyengha@uos.de

1 Abstract

Für die Qualitätssicherung eines Geschäftsprozesses, ist es unerlässlich, die Geschäftsprozesskette ganzheitlich zu betrachten. Das schließt die Betrachtung jeder einzelnen funktionalen Komponente ein. Dies ist besonders bei Geschäftsprozessen schwierig, die eingebettete Systeme involvieren. Deren Software-Komponenten sind nur schwer zugänglich. Die Software-Komponenten in den eingebetteten Systemen haben aber besondere Relevanz für das Einhalten von Echtzeitbedingungen innerhalb eines Geschäftsprozesses. Die Verletzung dieser Bedingungen kann katastrophale und lebensbedrohliche Folgen haben. Erst in der neueren Forschung werden auch eingebettete Systeme auf höherer Ebene modelliert, generiert und insbesondere in ersten Ansätzen auf höherer Ebene getestet [KSLB03].

Das vorgestellte modell-basierte Test-Framework (basierend auf Model-Driven Architecture (MDA) [Obj11]) stellt eine Umgebung zur Qualitätssicherung für eingebettete Geräte bereit, welche sich in die Methoden zur Qualitätssicherung der gesamten Geschäftsprozessmodellierung integrieren lässt. Wie in Abb. 1 zu sehen ist, sind die beiden Bereiche Software-Entwicklung und Testen in zwei Phasen aufgeteilt, welche durch Model Driven Development (MDD) beziehungsweise Model-Based Testing (MBT) gekennzeichnet sind. Die Software, welche während der MDD-Phase entwickelt wird, ist Gegenstand der Qualitätssicherung. Die Qualitätssicherung wird durch das Test-Framework sicher gestellt, das in der MBT-Phase erzeugt wird (vgl. Abb. 1). Der Algorithmus zum Generieren der Testumgebung ist in der Programmiersprache Java implementiert und mit einem MBT-Tool integriert. Die Ergebnisse der Qualitätssicherung während der MBT-Phase werden als UML-Diagramme (z.B. Sequenzdiagramme und Zeitverlaufdiagramme) automatisch generiert. Durch den Einsatz von Unified Modeling Language (UML)- Diagrammen [Obj11] wird die Qualitätssicherung auch der am Geschäftsprozess beteiligten Hardware auf einer höheren Ebene unterstützt. So kann diese Umgebung zur Qualitätssicherung leicht in die Qualitätssicherung des gesamten Geschäftsprozesses integriert werden, da die Qualitätssicherung eines Geschäftsprozesses ebenfalls auf einer höheren Modellierungsebene durchgeführt wird (z.B. mit UML, EPC oder BPMN).

Zur Veranschaulichung betrachten wir beispielhaft einen Geschäftsprozess, der bei einer Funkenlöschanlage [Neu11] zum Einsatz kommt. Dieser Prozess umfasst die Aufgaben,

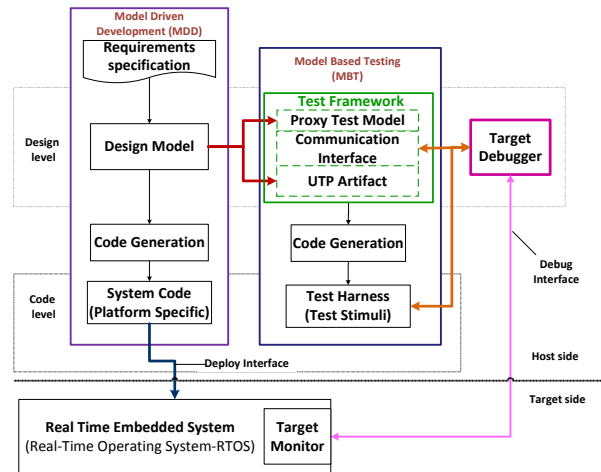


Abbildung 1: Integrierte modell-basierte Architektur und Test-Framework

die im Brandfall bei Löscharbeiten in der holzverarbeitenden Industrie durchgeführt werden müssen. Während die Aspekte der Qualitätssicherung für die verschiedenen funktionalen Komponenten in einem solchen Prozess vollständig berücksichtigt sind, sind die Eigenschaften der Software auf dem eingebetteten Gerät (z.B. auf der zentralen Steuerkonsole) schwieriger zu ermitteln. In der oben erwähnten Geschäftsprozesskette müssen die Daten, welche auf den eingebetteten Geräten verarbeitet werden, in Echtzeit übertragen werden - also ohne Verzögerung und Latenz. Eine Verzögerung bei der Auslösung des Alarms und/oder der Einleitung des Feuerlösch-Prozesses ist nicht tolerabel.

Der vorgestellte Ansatz nutzt für die Qualitätssicherung der eingebetteten Software UML-Sequenzdiagramme, die das Verhalten der eingebetteten Software-Systeme (z.B. auf der zentralen Steuerkonsole) beschreiben. So können die Ergebnisse der Qualitätssicherung nun auch als UML-Interaktionsdiagramm mit einer übergeordneten Notationen integriert werden, was einer ganzheitlichen Sicht entgegen kommt. Dies ist ein großer Vorteil für die Integration mit übergreifenden Qualitätssicherungsprozessen für alle funktionalen Komponenten der Geschäftsprozesskette.

Literatur

- [KSLB03] G. Karsai, J. Sztipanovits, A. Ledeczi und T. Bapty. Model-integrated development of embedded software. *Proceedings of the IEEE*, 91(1):145 – 164, January 2003.
- [Neu11] Neuentwicklung einer Funkenloeschzentrale. <http://www.willert.de/assets/Vortraege/Entwicklung-einerFunkenloeschanlage-f.-d.-Holzindustrie-mit-Rhapsody-von-der-Heide-Boerries-Innovate-IBM-2010.pdf>, 2011.
- [Obj11] Object Management Group. <http://www.omg.org>, 2011.