Chapter 6
A UML–Compliant Approach for Intelligent Reconfiguration of Embedded Control Systems

Amen Ben Hadj Ali
Tunis El Manar University, Tunisia

Samir Ben Ahmed
Tunis El Manar University, Tunisia

Mohamed Khalgui
Xidian University, China

Antonio Valentini
O3neida Europe, Belgium

ABSTRACT

The chapter deals with UML-based design of Reconfigurable Embedded Control Systems (RECS). The different software architectural configurations of the control system are described as a set of inter-connected software components. A software agent is defined in order to handle dynamic reconfiguration scenarios. The agent has the ability to interact with users and to monitor the system's environment in order to apply valid reconfiguration scenarios at different levels of the system's architecture. In order to address all possible forms of reconfiguration, three architectural levels are defined. A set of UML-compliant metamodels are proposed in order to describe the knowledge about the reconfiguration agent, the system architecture, the reconfiguration scenarios, and the reconfiguration events. The validity of the reconfiguration scenarios is checked by using a UML-based environment that allows evaluation of the architectural and reconfiguration constraints. The proposed reconfiguration approach is applied to two industrial production systems, FESTO and ENAS.

INTRODUCTION

In the context of modern Embedded Control Systems (ECS), one of the most important challenges is the tradeoff between performance and rapid response to market changes and customer needs (Leitão, 2004). This tradeoff is better obtained when addressed early in the development process at design time. Indeed, modern ECS become more complex because they incorporate increasing amounts of software. In order to counter the effect of growing complexity, ECS are often designed in

a component-based fashion as a network of inter-connected software components using different technologies that have been proposed for this aim such as IEC61499 (IEC-Part1, 2005; Lewis, 2001). Consequently, even small changes in the system design or any failure at run-time require a cost- and time-intensive effort to adapt the system. One of the most promising directions to address these issues is the reconfiguration which refers to the process of modifying the system's structure and behavior during its execution. Being reconfigurable is important for reacting fast to sudden and unpredictable requirement changes with minimum cost and risk. Within the literature of ECS, two reconfiguration policies could be identified depending on the way of re-configuration execution: static (offline) (Angelov, Sierszecki & Marian, 2005) or dynamic (online) reconfiguration. In the last case, two sub-classes exist: manual reconfigurations to be executed by users (Rooker, Sunder, Strasser, Zoitl, Hummer & Ebenhofer, 2007) and automatic (intelligent) reconfigurations to be performed by intelligent agents (Brennan, Vrba, Tichý, Zoitl, Sünder, Strasser & Marík, 2008).

In last years, researches have been conducted on architectures and software engineering in order to enable development of reconfigurable ECS (Khalgui, 2011). The architecture of RECS can be described by different software configurations such that each one is designed by a set of inter-connected software components (controllers) and each one of these components has to control a part of the system (Khalgui & Mosbahi, 2010). The reconfiguration of ECS corresponds to the execu-tion of reconfiguration scenarios on the software architecture of the system (Khalgui & Mosbahi, 2011). A reconfiguration scenario is an ordered sequence of reconfiguration operations that consist of adding/removing controllers, adding/removing connections between them or updating their data or code. Each operation corresponds to a transi-tion from one configuration to another and it is triggered under particular conditions as response to reconfiguration requests. A request represents a need to improve the system's performance, or to recover and prevent hardware/software errors, or also to adapt the system's behavior to new require-ments according to the environment's evolution.

In our research work we are focusing on the design of reconfigurable agents-based ECS in a platform independent way and with flexible recon-figuration spectrum covering manual, automatic and hybrid (i.e. a combination between manual and automatic) execution. For this purpose, we propose a UML-based approach for the design and validation of such systems. The Unified Modelling Language (OMG-UML, 2010) is the general-purpose language for modeling intensive-software systems. The proposed approach is the first to our knowledge to deal with UML-based design and validation of reconfigurable agents-based ECS which is an attempt to answer three research questions: (1) how to model reconfigura-tion and system's architecture, (2) how to execute reconfigurations on systems, and (3) how to ensure that reconfiguration agent brings the system into correct and safe behaviors.

In order to answer these questions, there are two ingredients of the proposal, the specification and the validation of the solution. The specification of the solution is covered by a set of UML-com-pliant metamodels enabling to specify the RECS architectures with their constraints, the intended reconfiguration scenarios and the architecture of the reconfiguration agent. The specification of RECS is based on the central concept of control-ler components described as UML components (OMG-UML, 2010) which are self-contained units, with ports as interfaces for external com-munications. Therefore, the concept of controller component used in this paper corresponds to a platform independent software unit that can be instantiated in several specific platform languages

Related Content

Measuring Developers' Software Security Skills, Usage, and Training Needs

Tosin Daniel Oyetoyan, Martin Gilje Gilje Jaatunand Daniela Soares Cruzes (2022). *Research Anthology on Agile Software, Software Development, and Testing (pp. 2026-2048).*

www.irma-international.org/chapter/measuring-developers-software-security-skills-usage-and-training-needs/294556

Enriched Conceptualization of Subtyping

Terry Halpin (2009). *Innovations in Information Systems Modeling: Methods and Best Practices (pp. 1-16).*

www.irma-international.org/chapter/enriched-conceptualization-subtyping/23781

A Modified Parallel Heapsort Algorithm

Hiroaki Hirataand Atsushi Nunome (2020). *International Journal of Software Innovation (pp. 1-18).*

www.irma-international.org/article/a-modified-parallel-heapsort-algorithm/256233

The Role of Information Technology Managers in the Significant Company in Case of Natural Disasters in Qatar

Salem Al-Marriand Muthu Ramachandran (2010). *Handbook of Research on Software Engineering and Productivity Technologies: Implications of Globalization (pp. 364-371).*

www.irma-international.org/chapter/role-information-technology-managers-significant/37042

Line-of-Sight Detection Using Center of Gravity with Pixel Number Variation

Takahito Niwa, Ippei Toriiand Naohiro Ishii (2017). *International Journal of Software Innovation (pp. 64-76).*

www.irma-international.org/article/line-of-sight-detection-using-center-of-gravity-with-pixel-number-variation/182537