Jens Knoop, Uwe Zdun (Hrsg.): Software Engineering 2016,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2016 107

Evolution of Software in Automated Production Systems: Challenges and Research Directions

Birgit Vogel-Heuser[1], Alexander Fay[2], Ina Schaefer[3], and Matthias Tichy[4]

Abstract: Coping with evolution in automated production systems implies cross-disciplinary challenges along the system's life-cycle for variant-rich systems of high complexity. We provide an interdisciplinary survey on challenges and research directions in the evolution of automated production systems. After an initial discussion about the nature of automated production systems and their specific development process, we sketch in this extended abstract the challenges associated with evolution in the different development phases and a couple of cross-cutting areas.

Keywords: Automated Production Systems, Cross-Disciplinary Development, Challenges, Research Directions

1 Automated Production Systems

Automated production systems (aPS) form the backbone of the world's industrial production. They are highly specialized technical systems, which are comprised of mechanical, electrical and electronic parts and software, all closely interwoven. Software is the defining factor to realize modern trends in manufacturing as defined by mass customization, small lot sizes, high variability of product types, and a changing product portfolio during the lifecycle of an automated production system. Hence, the evolution of automated production systems always requires addressing cross-disciplinary evolution challenges.

aPS in special machinery and plant manufacturing industry are typically designed-to-order, i.e. they are unique systems, which are designed and implemented once a customer has awarded a contract to an aPS supplier. Until completion, such projects usually last between several weeks and several months. In order to shorten project durations and to reduce costs, reusable (partial) solutions are usually developed by system suppliers and combined to realize the automated production system. aPS are supposed to be in operation and continuously evolved for decades before they are finally taken out of operation and are demolished.

[1] Institute of Automation and Information Systems, Technische Universität München, Boltzmannstr. 15, 85748 Garching near Munich, Germany, email: vogel-heuser@tum.de
[2] Institute of Automation Technology, Helmut Schmidt University, Holstenhofweg 85, 22043 Hamburg, Germany, email: alexander.fay@hsu-hh.de
[3] Institute of Software Engineering and Automotive Informatics, Technische Universität Braunschweig, Mühlenpfordtstr. 23, 38106 Braunschweig, Germany, email: i.schaefer@tu-braunschweig.de
[4] Institute of Software Engineering and Programming Languages, Universität Ulm, 89069 Ulm, Germany, email: matthias.tichy@uni-ulm.de

2 Challenges and Research Directions

In the following, we discuss the evolution challenges and research directions covering the different development phases as well as several important cross-cutting aspects. We refer to the full paper [1] for a complete discussion of challenges, state of the art, and research directions illustrated using a simple production system.

A light weight and efficient way to define *requirements and system specification* for aPS and refine or change them during the design process needs to be developed for both functional and non-functional requirements. They should be formalized in a way that they are quantifiable but still technology-independent and cover all aspects of the aPS. Thus, their continuous fulfillment can be verified before and after an evolution step. Challenges in *system design* include the need to ensure consistency between design artefacts of the different disciplines as well as different aspects of the system to be built. Furthermore, the concept of technical debt needs to be investigated w.r.t. aPS. During *system realization and implementation of the design* of aPS lack of modularity concepts fulfilling the cross disciplinary requirements of aPS can result in reuse by copy&paste leading to clones. Additionally, the evolution during operation is performed by customer personal on-site which are experts in the production process but not in software engineering and work under stress and time pressure. That can lead to inconsistencies between design and implementation as well as unclear code structures. The main challenge for *validation and verification* under system evolution is to provide efficient techniques to establish the desired system properties after evolution without the need to re-verify the complete evolved system from scratch. In particular, compositional and incremental verification and validation techniques should be developed to cope with cross-discipline models and reduce the effort for re-establishing properties.

The main challenge concerning *variability management* of aPS is the management of multi-disciplinary variability in problem and solution space for functional requirements and non-functional requirements with different levels of abstraction and granularity. A particular problem is here to ensure the consistency of the variability models across disciplines.. There exists research to use *Model-Driven Engineering* for the development of aPS, the challenge is to avoid inconsistencies between the models and the generated code in case of changes on site. Finally, as aPS have a cross-disciplinary nature and many different artifacts are built during development ensuring *traceability* between all artifacts is an important challenge.

References

[1] Birgit Vogel-Heuser , Alexander Fay , Ina Schaefer , Matthias Tichy , Evolution of software in automated production systems - Challenges and Research Directions, The Journal of Systems & Software (2015), Elsevier, doi:10.1016/j.jss.2015.08.026