Modeling of Agile Avionics Software Development Processes through the Application of an Executable Process Framework

Patrick KINGSBURY, André WINDISCH
Cassidian Air Systems
Rechliner Strasse
85077 Manching, Germany

and

Wolfram HARDT
Chemnitz University of Technology
Department of Computer Science
09107 Chemnitz, Germany

ABSTRACT

In software intensive avionics projects the problem of missing adherence to the complex process landscape has been known for decades. This problem is significantly aggravated when the combination of business needs, such as improving productivity and responsiveness to technical changes are required in addition.
The modeling of the Agile Avionics Software Development Processes through the Application of an Executable Process Framework shows first useful results in improving the situation of missing process adherence and is increasing transparency of process changes.

Keywords: Process Modeling, Complex Avionics System Software, Executable Process Framework, Agile Avionics Software development, BPMN2.0

1. INTRODUCTION

In today's avionics software development the survival and growth of business requires effective means to align organizational business objectives with software project management and software processes.

The continuous technological advancement of computer technology over the past decades is accompanied by a similar growth of the complexity of avionics systems which in turn caused an exponential increase of the complexity of aircraft software [9] as indicated by figure 1. For decades this increasing software complexity has been standing in strong contrast to the problem of insufficient or missing software process adherence in the complex avionics software engineering process landscape. The results are observed in many civil and military aviation programs leading to severe cost and schedule overruns. It's not that the software doesn't work; it's the traceability of the software [7], i.e. the proof that it has been developed according to the standards.

Furthermore, this problem is significantly aggravated in a competitive environment where improved productivity, faster time to market and better quality are required. Traditionally the approach to Avionics software development follows the waterfall lifecycle model as depicted by figure 2 which provides less development speed compared to the Agile lifecycle model shown in figure 3.

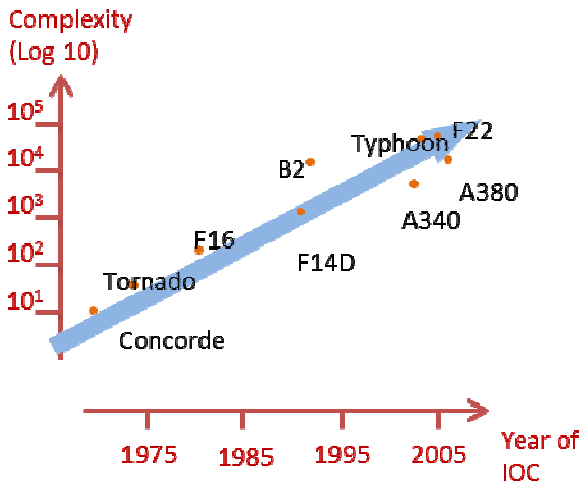

Figure 1: Increasing Complexity of Aircraft Avionics

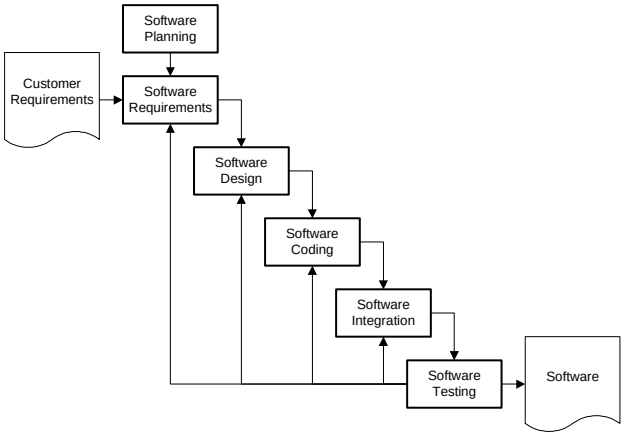

Figure 2: Waterfall Lifecycle Model

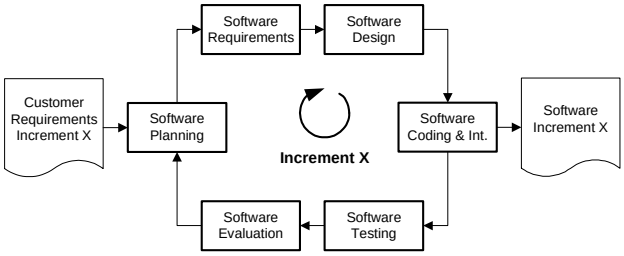

Figure 3: Agile Lifecycle Model

This paper describes the new idea to model the complete avionics software process landscape incorporating both development and certification standards with the Business Process Modeling Notation 2.0 (BPMN). The first result of this research project currently undertaken at Cassidian is the development of a BPMN 2.0 based process framework for the specification and deployment of complex agile avionics software engineering processes.

Besides its formal static semantics the BPMN standard also specifies execution semantics for the implementation of business processes in corporate IT infrastructures. In the context of our research project this feature has been used to deploy and execute the complex avionics software development process landscape transparently via web-browser in the complex software development environment.

Through the application of this process framework the introduction of the Agile Lifecycle Model for the avionics software development became feasible in the context of traceability for certification.

With this solution two new ideas are presented to the area of software engineering and process modeling: (a) to use the Business Process Modeling Notation (BPMN 2.0) for the formal specification of all software project management and software engineering processes and (b) to use a process engine to deploy and execute agile avionics software development processes.

The remainder of this paper is structured as follows: chapter 2 will provide a brief overview on related work before the executable process framework is discussed in chapter 3. Subsequently, in chapter 4 the formal process models for agile avionics software engineering and their application in the executable process framework are presented. Chapter 5 concludes with an outlook on future activities.

2. RELATED WORK

Today the area of process modeling and execution is mainly restricted to the business process level. In particular the new BPMN 2.0 standard has gained wide acceptance in industry. Several application areas have been reported, such as internal process management in large health care institutions [4], customer management [4] and process migration in telecommunications [5], customer support management in aerospace [6], and many others.

This acceptance in industry is based on the need for a common, cross-domain process standard which not only supports the modeling, but also the static verification of complex process landscapes and their deployment on enterprise IT-infrastructures. BPMN 2.0 fulfils all these requirements: its static semantics are formally defined by a UML-Metamodel and its execution semantics in terms of WEB services.

Although the new BPMN 2.0 standard explicitly lists engineering processes as a possible area of application, no references to Avionics software development processes could be found. One of the reasons could be, that traditionally the application of software process standards in industry is defined by a set of authorized planning documents (e.g. software development plan, software verification plan, etc.) which specify the individual processes, their inputs / outputs, and the process stages to be performed. Even though these processes are usually depicted in some graphical form, no formal process modeling is applied.

In the context of the EUREKA-ITEA AGILE Projects A. Wils et al [15] investigated the applicability of agile

methods to the embedded software domain. At a first glance the combination of agile development with certification of Avionics Software seems to be a contradiction, but it is feasible. However, no particular agile process solution was presented. No further publications on agile software development for avionics systems have been found.

3. THE EXECUTABLE AVIONICS SOFTWARE ENGINEERING PROCESS FRAMEWORK (EASE-P)

Framework Requirements

The lack of formal process modeling in the agile avionics software engineering domain manifests itself in inconsistent software process planning documents and insufficiencies in software engineering process adherence. The consequences are severe project delays, cost overruns, and quality problems - the most recent one being reported in [7]. From our experience the reasons for project failure are manifold, however, the top 3 addressed by this paper are:

(1) Inconsistencies in the software engineering process landscape
(2) Lack of adherence to software engineering processes or methods
(3) Insufficient project metrication, solely based on Earned Value Management (EVM)

The source of problem (1) is the complexity of the process landscape required for agile avionics software engineering. Figure 4 shows the landscape of the relevant processes of which most not only run in parallel but are also of a highly iterative nature. However, the resulting complex process interaction pattern are typically neither modeled nor verified. Instead, different process areas are defined by software planning documents which provide an informal picture of the processes, textually detail their activities, and describe their input / output relation with other processes.

The non-adherence to software engineering processes (2) cannot just cause major project delays but also endanger Avionics software certification. Although detailed textual descriptions of all processes and process stages exist, the overall complexity of the process landscape obstructs the situational awareness of the individual software engineers. This problem is intensified by the fact that engineers are typically assigned to one process stage only, e.g. software requirements analysis, and typically have very different educational backgrounds and skills.

The third reason for project failure addressed by this paper is the one-dimensional project measurement and control process implemented in most organizations. The standard approach is the utilization of Earned Value Management (EVM), i.e. project progress is measured in terms of man hours spent vs. values earned in terms of project milestones achieved. However, a project milestone does not denote quantitative and qualitative product information. To get a clear picture of productivity rates and product quality the EVM system has to be complemented with a product metrication process.

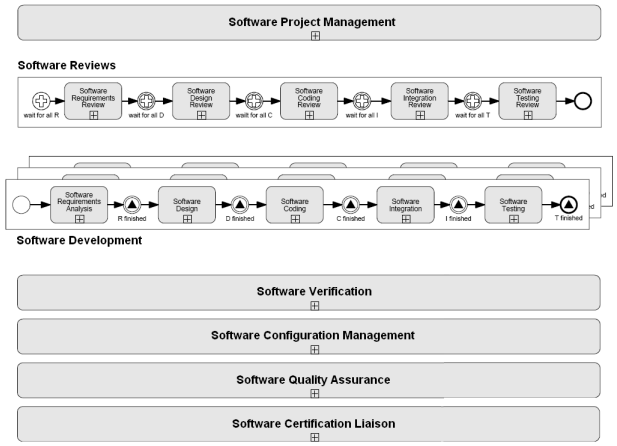

Figure 4: The agile avionics software process landscape

To eliminate the aforementioned deficiencies in Avionics software engineering a new approach to process modeling and execution is required. This new approach should

- Utilize formal process specifications which lend themselves to the application of formal verification techniques in order to eliminate process inconsistencies
- Provide explicit and graphical process guidance to increase the individual situational awareness and to reduce the impact of personal educational backgrounds and skill sets
- Support the implementation and integration of product metrication's to complement the traditional EVM based project control process.

Framework Concept

This paper proposes the new idea to adopt the business process modeling and execution approach to the domain of Avionics software project management and software engineering. All processes on the business, project, and

engineering levels are then specified in the unified formal notation BPMN 2.0. Based on the static semantics of this notation formal verification approaches, e.g. model checking, can be applied which enable the detection and elimination of inconsistencies in process interaction and process data exchange by simulation.

Besides this strong advantage the utilization of BPMN 2.0 offers additional benefits for Avionics software project management and Avionics software engineering. This paper proposes the new idea to deploy and execute the formally specified processes on a process execution engine such as jPBM [8]. However, this engine not only executes the individual project and engineering processes but also ensures overall process orchestration. Based on the graphical syntax of BPMN 2.0 both execution and orchestration of these processes can be graphically represented in a tool to provide an explicit visual guidance for the software engineers and to reduce the direct impact of personal educational background and skills. Moreover, for training purposes the process execution can be simulated for training projects. This allows for a seamless integration of training and engineering activities.

For avionics software project management the presented approach can be extended by integrating product metrication activities. This requires the specification and deployment of product metrication activities in the scope of the project measurement and control process and the implementation and integration of metrication procedures on the process execution engine. The invocation of these metrication procedures is then triggered whenever the corresponding activities of the project measurement and control process are executed. The gathered quantitative and qualitative product data – i.e. number of base lined requirements, implemented LoC, completed test procedures – provide a far more detailed project status than that solely based on EVM.

Framework Implementation

The conceptual ideas presented in the previous section were validated in the aerospace industry by means of an implementation prototype. The resulting EASE-P process framework combines existing tools, such as configuration management systems and task databases, with a new process execution engine and process visualization tools.

The overall tool architecture of the EASE-P process framework is depicted by figure 5. It utilizes a web-based client / server architecture where the jPBM process execution engine is integrated on the web server. The processes can be visualized and controlled via standard web browsers interactively. However, the type of interactions allowed is restricted for the different users depending on their role and responsibility in the project. The implementation of this approach is based on system access rights which also govern access to the framework tools. This ensures for instance that project metrication can be executed only from an account with project management rights. However, the same account is not permitted to introduce software configuration baselines or to check-in source code.

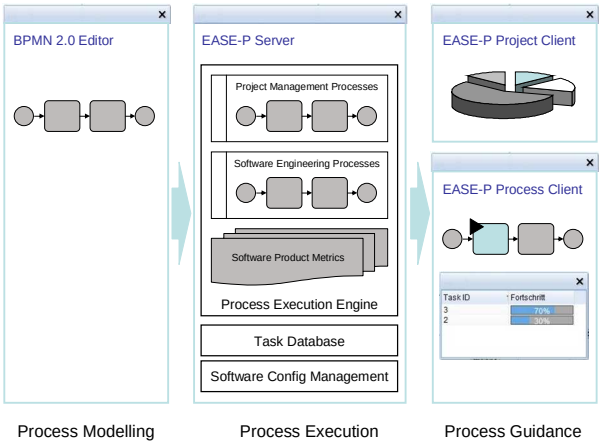

Figure 5: The executable avionics software engineering process (EASE-P) framework

The integration of the process execution engine and the existing tools is based on web-services. This approach is conforms to that used on business level and hence allows the future integration of the EASE-P framework into the enterprise IT-infrastructure.

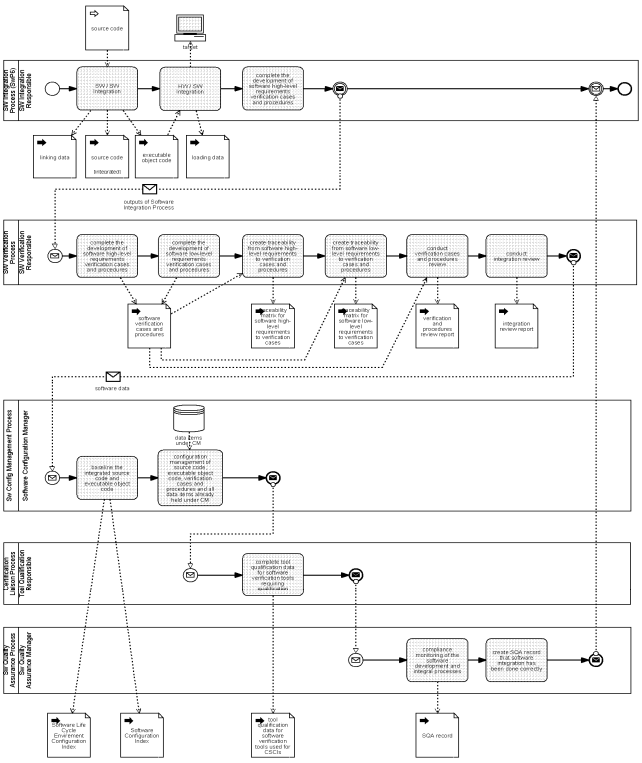

Figure 6: BPMN specification of software integration process [14]

4. AGILE AVIONICS SOFTEWARE ENGINEERING PROCESS FRAMEWORK

The implementation prototype of the EASE-P process framework was used to model and deploy the agile avionics software development processes as depicted by figure 4. Thereby, the top-level BPMN specification closely follows the planning document structure defined by the avionics software certification standard DO-178B [3]. In this model the agile software development process formally captures all activities usually described by the Software Development Plan (SDP). The same applies to software verification, quality assurance, configuration management, and certification liaison which are traditionally defined by the Software Verification Plan (SVP), the Software Quality Assurance Plan (SQAP), the Software Configuration Management Plan (SCMP), and the Plan for Software Aspects of Certification (PSAC).

The only exception to this rule is the separate specification of the software review activities which normally constitute one specific part of the software verification activities. However, this modification was necessary due to the fact that we allow multiple software development process instances to be executed in parallel in order to achieve a agile software development. Figure 7 shows the holistic view on the agile avionics software process realization.

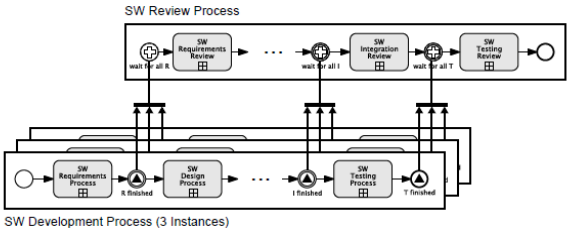

Figure 7: Holistic view on agile avionics software process [14]

In this context the explicit software review process is used to synchronize the parallel development phases before the formal software review is conducted, which is mandatory to fulfill the certification requirements for Agile avionics software development.

The presented modeling approach extensively uses BPMN process composition to roll-up the implementation details. As an example consider the software integration process shown as a single process box by figure 4. The formal specification of the detailed activities of this process is depicted by figure 6. In this BPMN specification the parallel execution of software processes is modeled by means of BPMN pools each of which encapsulates the process-specific sequence of activities. The information and data flow between these processes are modeled in terms of BPMN events which trigger and synchronize the internal activities of the concurrently executing processes.

The process models have been deployed on the EASE-P process framework which provides graphical process guidance to software project managers and software engineers.

5. CONCLUSION

This paper presented the two new ideas to implement agile software development processes for avionics software engineering: (a) to use the Business Process Modeling Notation (BPMN 2.0) for the formal specification of all software project management and software engineering processes and (b) to use a process engine to deploy and execute agile avionics software development processes.

Besides the strong advantage that BPMN 2.0 provides both a formal process specification semantics and an execution semantics the EASE-P Framework offers the following additional advantages:

1. Visualization of all software processes and their complex interaction to both software developers and project managers

2. Process guidance for all software developers through step-wise process execution to ensure subsequent avionics software certification

3. Situational awareness at each state for project managers through integration of metrication to ensure schedule adherence, productivity level and objective metrication based on development artifacts for Software Project Management which complements the traditional EVM approach

The presented approach shows that all necessary agile processes for the development of certifiable embedded Avionics software can be specified in BPMN 2.0 and integrated into EASE-P process framework. These processes and interactions have been based on the relevant standards for software development ISO/IEC 12207 und DO-178B.

REFERENCES

[1] Allweyer, Thomas. **BPMN 2.0 – Introduction to the Standard for Business Process Modelling.** Books on Demand GmbH, Norderstedt, Germany, 2009. ISBN 978-3-8391-4985-0

[2] International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC). **ISO/IEC 12207 – Information Technology – Software Life Cycle Processes.** 1995

[3] Radio Technical Commission for Aeronautics (RTCA). **DO-178B – Software Considerations in Airborne Systems and Equipment Certification.** 1992

[4] Signavio. **Introduction of a common process platform at AOK.** (In German). http://www.signavio.com/de/referenzen/kunden.html, 2010.

[5] Casewise Ltd. **How PBM and BPMS tools are supporting migration capabilities for Alcatel-Lucent.** http://www.casewise.com/NR/rdonlyres/45EF7DA6-55B5-4013-A2FF-F9ECBDEAE68F/0/alcatellucent.pdf, 2010.

[6] Interfacing Technologies Corporation. **Case Studies – Bombardier Aerospace.** http://interfacing.com/Literature/Business-process-case-studies, 2010.

[7] Reuters - **Brake software latest threat to Boeing 787 –** 15.07.08

[8] Red Hat. **Jboss jBPM Process Execution Engine**. http://www.jboss.org/, 2010.

[9] Robert A. Dietrick, Major, USAF: **Impact of Weapon System Complexity on Systems Acquisitions** - Air Command and Staff College Air University 2006

[10] From Wikipedia, the free encyclopedia - **Education**

[11] Silvia T. Acuna Universidad Autonoma de Madrid Spain and Natalia Juristo Universidad Politecnica de Madrid Spain - **Software Process Modeling**

[12] From Wikipedia, the free encyclopedia - **Productivity paradox**

[13] Berglas, Anthony (2008) - **Why it is Important that Software Projects Fail**

[14] F. Triebsch, Thesis - **Konzeption und Umsetzung eines Frameworks für die Modellierung, Ausführung und Metrikation von Agilen Prozessen in der Avioniksoftwareentwicklung**

[15] Wils, Andrew et al - **Agility in the avionics software world. K.U.Leuven DistriNet, 2006** http://www.agileitea.org/public/papers/agileavionics.pdf

[16] Beck, Kent et al - **Manifesto for Agile Software Development.** http://www.agilemanifesto.org**, (23.07.2010)**

[17] Sommerville, Ian – **Software Engineering 6th Edition**, Addison Wesley