An Integrated Process for Developing Safety-critical Systems using Agile Development Methods

Zhensheng Guo
Siemens AG, Corporate Technology
Erlangen, Germany
Joe.guo@siemens.com

Claudia Hirschmann
Siemens AG, Corporate Technology
Erlangen, Germany
Claudia.hirschmann@siemens.com

Abstract - **This paper proposes a novel idea for developing safety-critical software-intensive systems by the use of agile development models. This idea comes from the dramatic increase of the complexity for constructing technical systems such as trains, cars and medical devices. Iterative and incremental development becomes more and more popular and effective in such fields. However, the development of safety-critical systems is strictly defined by relevant safety standards. A kind of finish-to-start relationship between the development phases is required implicitly. This paper explains an idea to integrate an iterative incremental development process into the strict safety development lifecycle. At the end of this short paper, an overview of the further activities is presented.**

Keywords-safety-critical software-intensive systems; IEC 61508; EN 50128; agile software development process; Scrum

I. INTRODUCTION

Software is replacing traditional mechanical and electrical components with an extremely high speed and huge extends. Many of such components are used in safety-critical systems, where the malfunctions of software could cause the damage of equipment, environmental pollution, even injury or death of human being. Typical examples of such systems are trains, cars, aircrafts, medical devices and nuclear power plants. To develop high quality software components, it is nowadays essential to use a suitable development process model. In [15], several typical process models are presented and compared for the suitability for developing safety-critical systems: One of the core results of the referenced paper is that strict finish-to-start process models such as the V- Model XT [14] are suitable to construct safety-critical systems, whereas agile process models, like Scrum, are not recommended.

Agile software development methods follow the principle of 'plan – build – revise' in short iterative cycles. At the end of each cycle they deliver incrementally ready product features. Agile methods provide good measures to handle the reality of software development with its late requirements, need for flexibility and fast reaction times; see [7].

The bases behind agile processes are the Manifesto for Agile Software Development [1] and the Principles behind the Agile Manifesto [2]. Agile processes welcome changing requirements, deliver working software frequently and in close cooperation with the customer trusting the motivated development team. But, agile methods, like e.g. Scrum [8], do not care about dedicated roles for quality management or safety management, or documentation.

One of the most used agile methods is Scrum (see [6]), which is continuously enhanced, like in "The Scrum Primer, version 1.2" [3] or "The Scrum Guide" [4].

Application of agile methods in safety-critical, regulated environments (as in medical technology) is discussed in science and industry, see the ScrumMed conference [5][9][10]. Application of agile methods in safety-critical systems with focus on IEC 61508 or EN 50128 is still a gap which we want to fill with our idea.

Nonetheless, many industry domains are using agile methods to reduce the project risk and increase better orientation, flexibility, transparency and even quality: So, there is a gap and practical need for finding out how to use iterative process models and agile methods save and effectively for developing safety-critical systems.

There is no specific process model required in the safety standards such as IEC 61508 [12] and EN 50128 [13]: in IEC 61508 a safety lifecycle is defined and required, in EN 50128 waterfall model and V-Model is referenced but not required. The safety standards only require certain activities and documentation with some quality according to the corresponding Safety Integrity Level (SIL). The appropriate process model can be decided by the individual project whereat the new version of EN 50128 even mentions the consideration of iterative development.

Therefore, an iterative process model for developing safety-critical systems becomes important: integrating the benefits without letting the drawbacks from safety view in.

II. IDEA FOR DEVELOPING SAFETY-CRITICAL SYSTEMS USING AGILE DEVELOPMENT METHODS

Our idea is to map the activities of an agile process model into the safety lifecycle. The agile process model will be used, but not as the only process model. The traditional strict finish-to-start process model will be used as well. This limitation has the benefit that the required activities and documents of the individual phases are done in the required sequence of the safety standards.

The following section explains how the agile process model can be mapped into the safety lifecycle.

In order to make our idea general, we take the software safety lifecycle from the mother safety standard, IEC 61508,

 ISBN: 978-1-61208-230-1

part 3. In the software safety lifecycle in IEC 61508 part 3, the activities and documents, which shall be done after each individual phase, are readily identifiable from the name of each phase of safety lifecycle.

Scrum shall contain the following activities and artifacts: Product backlog (user requirements) and Release plan; Sprint planning meeting and Sprint Backlog; Code including documentation after each sprint / increment; Review meeting and results; Retrospective.

Now, the Scrum activities and artifacts will be arranged in the phases of the safety lifecycle as described in Figure 1.

Figure 1 illustrates the idea of the solution for developing safety-critical systems using agile development methods from the Software safety lifecycle view point. It shows which agile elements would go into which step of the safety lifecycle: The Product Backlog from Scrum method will be applied for the SW safety requirements specification additionally. SW design & development will be performed according to Scrum's Sprints including the Sprint planning meeting, Sprint Backlog, the Daily Scrum Meetings, Sprint Review, and regular Sprint Retrospectives. The Sprint Review and Sprint Retrospective element from Scrum will be applied for the Software safety lifecycle step of SW aspects of system safety validation, as well, to facilitate stakeholder involvement and continuous improvement of the product and process.

The Sprints for the implementation of the different features could be performed in parallel and sequentially, depending on the dependability of the features that are implemented in the sprints and corresponding safety requirements respectively; so, the whole Software will be implemented iteratively through as many Sprints as needed.

Figure 2 illustrates the idea of the solution for developing safety-critical systems using agile development methods from the Scrum view point. It shows the Scrum method's framework form Product Backlog over Release Plan, Sprint Planning Meeting, Sprint Backlog into the Sprints ending with the Sprint Review Meeting and Sprint Retrospective Meeting, whose results flow back into the planning for the next Sprint. In order to ensure accurate consideration of safety issues, a Safety Manager will care for safety assurance throughout the whole process. So the Safety Manager will check the Product Backlog for certain safety goals and care for proper ranking and arrangement in the Release Plan,, Sprint Backlog and Sprint Review Result, and will monitor and track safety items during the Daily Scrum Meetings and the Sprint Retrospective Meetings. He supports the Scrum Master in all sprints, takes part in the Sprint Review and in the Sprint Retrospective to care for proper documentation, etc.

A second step of this integrated process is to integrate the Start and Done- criteria as a refinement of [11] into the software safety lifecycle. Such criteria define start and end of each sprint in the overall safety lifecycle.

The Done Criteria that are checked after each Sprint to determine whether a task of the Sprint is completely done or not, is steadily maintained and kept by the Safety Manager. These Done Criteria, which serve as checklist, will include for example safety relevant review activities, documentation, and safety measures.

One crucial factor for use of such an integrated process model is the correct definition of the sprints according to the separation of the non-safety-critical functions from the safety-critical functions and the decomposition of the safety functions regarding their criticalities and dependencies.

III. FUTHER ACTIVITIES

We are planning to use this integrated process model in the Scrum teams for developing safety-critical systems. Important is that the overall safety activities will be well-integrated and advised by the Scrum master and Safety manager. Of course, the recommended and required techniques according to the different SILs, Change Management like other required activities from the relevant safety standards will be used and integrated in each sprint.

IV. CONCLUSION

In this paper, we presented a novel idea to integrate agile methods / Scrum into the safety lifecycle to enable iterative incremental development in safety-critical systems.

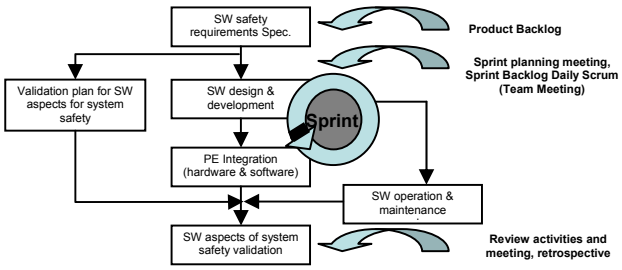

Figure 1. Software safety lifecycle with Scrum elements.

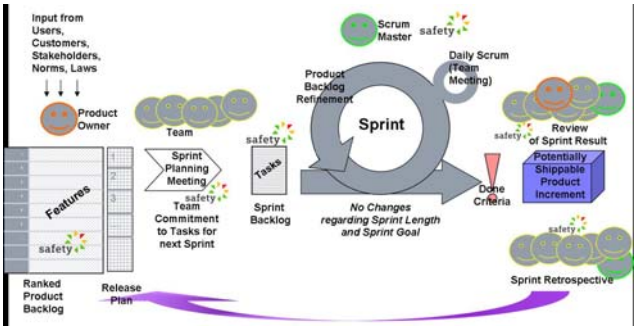

Figure 2. Safety-oriented Scrum process framework.

V. REFERENCE

[1] http://agilemanifesto.org/. [retrieved: September, 2012].

[2] http://agilemanifesto.org/principles.html. [retrieved: September, 2012].

[3] Pete Deemer, Gabrielle Benefield, Craig Larman, and Bas Vodde: "The Scrum Primer, version 1.2", 2010, pp. 4-16.

[4] Ken Schwaber and Jeff Sutherland: "The Scrum Guide, The Definitive Guide to Scrum: The Rules of the Game", 2011, pp. 3-15.

[5] Jörg Bindner and Claudia Hirschmann: "Agil in Medical Technology? Hand in Hand with Quality Management" (Transl.), published in OBJECTspectrum, Agility/2009, pp. 1-5.

[6] Results from Scott Ambler's March 2006 'Agile Adoption Rate Survey' posted at www.ambysoft.com/surveys/. [retrieved: September, 2012].

[7] Computerwoche: „Agile Methods in Comparison" (Transl.), http://www.computerwoche.de/software/software-infrastruktur/2352712/ , April 2012. [retrieved: September, 2012].

[8] Roman Pichler: „Scrum – Using Agile Project Management Successfully" (Transl.), 2008, pp. 7-123.

[9] ScrumMed, Conference for Scrum in Medical Technology.

[10] Andrea Heck: „How is a large software project in medical technology getting agile" (Transl.), Colloquium at Georg Simon Ohm University of Applied Sciences.

[11] Jeff Sutherland: Sprint Ready and Done threshold, 2009.

[12] IEC 61508, Functional safety of electrical/electronic/programmable electronic safety-related systems, International Electrotechnical Commission, 2010, pp. 19-20.

[13] EN 50128, European Standard of Railway applications - Communication, signalling and processing systems - Software for railway control and protection systems. 2011, pp. 10-26.

[14] V-Modell XT Version 1.3 English, Federal Republic of Germany, 2006, pp. 10- 20.

[15] Adrien Mouaffo, Zhensheng Guo, Mahmudul Huq, Dieter Rombach, and Peter Liggesmeyer, Tool support for a safety- and security- based assessment model for software engineering processes, in Software Process Improvement And Capability dEtermination (SPICE) Conference, 2010.

 ISBN: 978-1-61208-230-1