Co-Engineering-in-the-Loop

Thomas Gruber[1], Christoph Schmittner[1], Martin Matschnig[2], and Bernhard Fischer[2]

[1]AIT Austrian Institute of Technology GmbH, Giefinggasse 4. 1210 Vienna, Austria
thomas.gruber@ait.ac.at
christoph.schmittner@ait.ac.at
[2]Siemens Aktiengesellschaft Österreich, Siemensstraße 90, 1210 Vienna, Austria
martin.matschnig@siemens.com
bernhard.bf.fischer@siemens.com

Abstract. System safety standards have been available for two decades. Remarkably, none of the functional safety standards gave detailed guidance on how to treat potential security risks; security was – if at all – only mentioned in a small remark. However, the way how systems are built has changed; today's safety-critical systems are more and more integrated in networks and, thus, the old paradigm of isolated systems is not any more valid. It has been recognized that safety and security, and since recently also performance, need to be treated in combination: Co-engineering is required. After a short glance at the state of the art in co-engineering methods and in respective standardization, the paper describes the approach of co-engineering with interaction points taken in the ECSEL project AQUAS, which has been running since May 2017. The methodology is illustrated with first details on how the co-engineering approach for the concept phase is realized in the industrial drive use case provided by Siemens AG Austria.

Keywords: Co-engineering, Product Lifecycle, Industrial drives, Safety, Security, Performance, Interaction point.

1 Introduction

System safety considerations look back on a long tradition; the first edition of the generic Functional Safety standard IEC 61508 [1] was issued in 1998. However, none of the Functional safety standards gave detailed guidance on how to treat potential security risks; security was – if at all – only mentioned in a small remark. Instead, the assumption was that safety-critical systems usually have to be separated from the outside world in a way that attacks that could compromise them were possible only with physical access.

The way how systems are communicating has changed; today's safety-critical systems are more and more integrated in networks and, thus, the old paradigm of isolated systems is not any more valid (e.g. Industry 4.0 [21]). Real events like the steel mill attack in Germany [2] or hackers causing power outages [22] attracted attention even in a wider public. It is therefore increasingly understood that attacks can compromise

safety and, therefore, security considerations are inevitably necessary also for safety-critical systems.

Several research projects like MERGE [9], AMASS [25] or AQUAS [27] have treated or are currently dealing with co-engineering, i.e. the concurrent treatment of more than one quality attribute in order to address risks of different origin. Primary target of the projects was the interplay between safety and security, but in recent projects like AMASS and AQUAS, the scope has been conceptually extended to cover more quality attributes, in particular performance. The problem the projects are trying to solve is that solutions like for instance risk mitigation measures targeting one quality attribute often have a negative impact on another one. These trade-offs need to be handled properly, and the projects try out different approaches to reach a balanced set of measures addressing the different concerns.

This paper presents the concepts of the interaction-point-based safety-security-performance co-engineering approach that is currently being elaborated in the AQUAS project. It is structured as follows: Chapter 2 describes existing approaches for co-engineering and current guidance given by standards. Chapter 3 introduces the general AQUAS approach for co-engineering and explains then its application in the industrial drive case study for the concept phase. An outlook on future work is given in chapter 4.

2 State of the art

As explained above, the industry relied for a long time on the paradigm that safety-critical systems are separated from the outside world. With today's systems getting more and more networked attacks can compromise system safety. Thorough security analysis and the respective risk prevention and risk mitigation measures must therefore be deployed for safety-critical systems. Safety and security measures require the reservation of performance-related capabilities in order to provide their service when needed. For example, a safely-limited speed function (IEC 61800 - Adjustable speed electrical power drive systems [3]) for e-motors has to take action very quickly, but should not deteriorate the overall system function (e.g. the positioning of a motor axis should still be precise). In the industrial domain one of the most important performance factors is the cycle response time in control loops (hard real-time), which is impacted by both, safety and security measures. State-of-the-art approaches treat the approximation of performance by simulation and experience data and they handle safety and security separately. This means that the design cycle has to be run several times and even then the gained results might differ at a large scope from what was originally intended.

A said in the Introduction, the mutual influence between the measures addressing various quality attributes made a trade-off analysis necessary. This leads to the concept of co-engineering, for which different approaches are under development in several research projects. Also standardization groups from different domains have reacted and they are offering guidance for treating security in safety-critical systems. The following subsections explain some of these developments.

2.1 Co-engineering

Co-engineering means that, in a system development phase, the engineering processes targeting different quality attributes are not anymore performed fully independent, but there are interactions of some kind between them. In the following, three existing co-engineering approaches are outlined shortly. They focus on the mutual influence between safety and security. The interrelation of these two quality attributes has hardly been studied up to now except for performance in the sense of human performance in the context of safety management.

SAHARA (Security Aware Hazard Analysis and Risk Assessment).
[8] presents a framework for the security aware identification of safety hazards for the automotive domain. The method enhances the inductive analysis method HARA (Hazard Analysis and Risk Assessment), which is requested by ISO 26262 [6], to cover also threats defined in Microsoft's Threat Model STRIDE (Spoofing identity, Tampering with data, Repudiation, Information disclosure, Denial of service and Elevation of privilege) [7]. The STRIDE approach allows quantifying the probability of occurrence and the impacts of security issues on safety concepts (safety goals). Each system component is analysed for its susceptibility to threats and, subsequently, all identified threats can be mitigated to ensure system security.

FMVEA (Failure Modes, Vulnerabilities and Effects Analysis)

The safety and security co-analysis method FMVEA [10] was developed in the context of the Arrowhead [28] project and extends the established FMEA (Failure Mode and Effects Analysis, see for instance [11]) with security related threat modes.

The failure part of the method consists, like in the FMEA, of failure cause, failure mode, and effect. The novelty is that security related parts are added here, including vulnerability, threat agent, threat mode and effect. Depending on the level of analysis, a vulnerability can be an architectural weakness or a known software vulnerability. Compared to safety, security requires not only a weakness but also an element, which is exploiting this weakness. This can be a software or a human attacker.

Different threat modelling concepts can be used for the identification of threat modes such as CIA (confidentiality, integrity, availability) [12], summarizing security properties an attack could exploit, or also STRIDE. Based on the severity of the effect, measured in terms of financial damage, loss of confidentiality or privacy and on the operational or safety impact and, finally, the likelihood of the failure or threat, the criticality is measured. In the likelihood context, the system properties and attacker properties have to be investigated. As a result, the FMVEA yields a semi-quantitative measure for the risk of each individual threat and failure mode, and, accordingly, security controls can be chosen based on the associated risk.

The Communication approach of SAE J3061

The Automotive security guidebook SAE J3061 [13] provides flexible guidelines for treating security in automotive systems. One of the recommended practices is a safety-security co-engineering process with "potential communication paths" between the yet separate safety- and security-related lifecycles.

These “potential communication paths” activities correspond to the “interaction points”, which are defined for the AQUAS approach. AQUAS, however, has also performance in the focus, which increases the complexity of the interactions. Therefore, the project is investigating ways how to shape these interactions and at which points of the PLC (product life cycle) they should take place. There are more approaches for safety-security co-engineering, e.g. STPA-Sec [23] or combined Fault and Attack trees [24]. We restricted our selection to those most closely related to the AQUAS approach.

2.2 Standardization

As mentioned in the Introduction, Functional Safety standards have a 20 years long tradition since IEC 61508 [1] has been issued. IT security standards like the ISO 27000 [5] series or the common criteria [4] appeared slightly later. The development of targeted security standards for IACS (Industrial Automation and Control Systems) started much later. Here, two standards with relevance to the AQUAS methodology or the Industrial domain are shortly outlined.

The security standards for automation and control systems usually provide methods for security analysis and techniques for security controls. Most of them provide also lifecycle models, which, however, consist of separate process flows for safety and security, connected by interactions.

The guidebook **SAE J3061** [13] provides set of high-level guiding principles for Cybersecurity engineering in the automotive industry and establishes a framework reaching from the concept phase through production, operation, and service until the decommissioning.

IEC 62443 “Industrial communication networks -Security for industrial automation and control systems” [14, 15, 16, 17, 18, 19, 20] is a series of standards and technical reports with the goal of improving safety, availability, integrity, and confidentiality of IACS (Industrial Automation and Control Systems). The standards define Procedures for implementing electronically secure IACS. Their guidance applies to end-users, system integrators, security practitioners, and control systems manufacturers responsible for manufacturing, designing, implementing, or managing IACS.

Out of these standards, the main concepts relevant for the AQUAS approach and in particular for the Industrial Drive use case are the concept of partitioning the system into zones and conduits, which allows a structured cybersecurity risk analysis with targeted measures for the safety-critical zones, and the concept of security levels, which links the identified risk to security requirements for the IACS components.

3 Co-engineering approach with interaction points

3.1 The general approach

The AQUAS approach is based on separate activities for the different quality attributes running in parallel and an interaction point which brings together experts for all

concern in order to evaluate the compatibility of the results. Fig. 1 shows an AQUAS safety/security/performance co-engineering process with an interaction point.

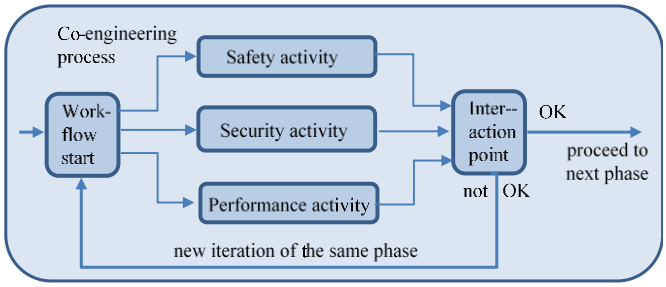

Fig. 1. AQUAS co-engineering process with separate activities and interaction point.

When the workflow starts, activities for all considered quality attributes (here: safety, security and performance) are triggered, ant they run independently in parallel until they yield their results. When all (here: three) activities have finished, the experts for safety, those for security, and those for performance hold a meeting together in order to verify whether the results of the parallel activities are compatible. This means, they include the mutual influences between the results in an overall evaluation and determine whether the goals of the Co-engineering process w.r.t. all considered quality attributes are met. If they are compatible ("OK"), then the workflow proceeds to the next development lifecycle phase. Otherwise ("not OK") the workflow goes back into a new iteration of the same phase, and all three activities are conducted again.

To illustrate the process, we can think of a safety/security/performance analysis process in the concept phase. They analyze the system model or structure for safety, security and performance properties and yield, as a result, three sets with safety, security and performance requirements. These may be contradictory, and in this case the mitigation measures for the three concerns have to be modified in order to meet all requirements. One of the goals of the AQUAS project is to minimize the necessary count of iterations until the results fulfill all safety, security and performance goals. The better the experts for the concerns understand each other, the easier is it to quickly find a solution compatible with safety, security and performance criteria.

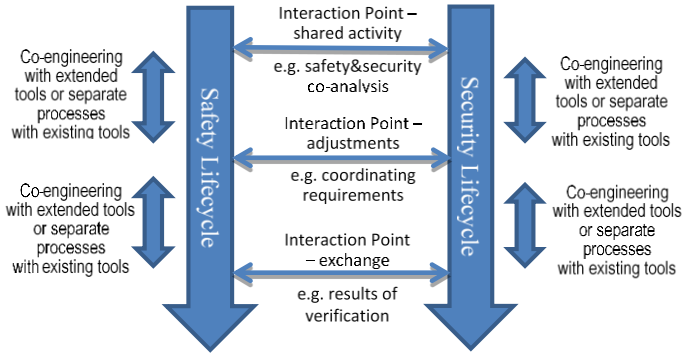

Fig. 2. AQUAS co-engineering process throughout the Product Lifecycle.

The AQUAS approach is applicable to the entire PLC (Product Life-Cycle). As an example, **Fehler! Verweisquelle konnte nicht gefunden werden.** shows safety and security co-engineering for a few PLC phases. The approach with parallel engineering processes for safety and security is in line with standards, in particular the guidebook SAE J3061 [13] for the automotive Industry, and the series of IEC 62443 security standards for the Industrial domain, from which several parts are not finished yet.

The AQUAS project plans the application of tools for activities. In the above example, a safety, a security and a performance analysis tool could be launched in parallel, and their results checked for compatibility in the interaction point. On the other hand, there are combined methods like for instance the FMVEA, which has been described in the previous chapter. In this case, the tool implements two activities in one process, in our example safety and security (co-)analysis. Then the interaction verifies only the compatibility between the FMVEA and the performance tool results.

The decision whether to use combined co-engineering tools or established tools for single quality attributes is individually possible for all phases. Moreover, the AQUAS approach as such can be deployed for a single phase only while, in the other phases, the company continues using the established legacy technologies, for which the staff is already trained. This flexibility allows a smooth transition from current company practices to the AQUAS framework with low effect on business continuity and low cost for tools and training. The following subchapter brings an example for the use of AQUAS concepts in the Industrial Drive use case provided by Siemens AG.

3.2 Detailed example for Interaction Points in the Concept phase

Automating workflows across multiple iterations of system development helps to accelerate the development flow while avoiding wrong or incomplete process chains caused by human error in the case of manually managing the activities. The Eclipse RCP-based tool WEFACT (Workflow Engine for Analysis, Certification and Test) support defining the, if applicable too-based, activities in the product lifecycle as well as their sequence (predecessor, successor), and then executing the workflow automatically. The concept allows also forking the process flow and combining the results after completing the parallel activities, as it is needed for the interaction point concept in AQUAS. WEFACT traces moreover whether the executed activities have been accomplished successfully and enables automated iterations in case the overall result of the parallel activities is not satisfactory (e.g. contradictions between resulting safety, security and performance requirements).

In the first step of the Industrial Drive use case, WEFACT is supporting the automated workflow of first performing Safety, Security and Performance Analyses, and finally starting interactions between the quality attribute-specific analysis processes. In case the interactions yield incompatibilities between the quality attribute specific processes, WEFACT leads the workflow back to a second iteration of the analyses. The above explained capabilities of WEFACT allow to instantiate exactly the process flow structure which is needed for the implementation of co-engineering with interaction points. **Fig. 3** shows an example for a part of a multi-concern assurance process.

The process is usually an iterative one, therefore the exemplary interaction points in the figure are traversed more than once.

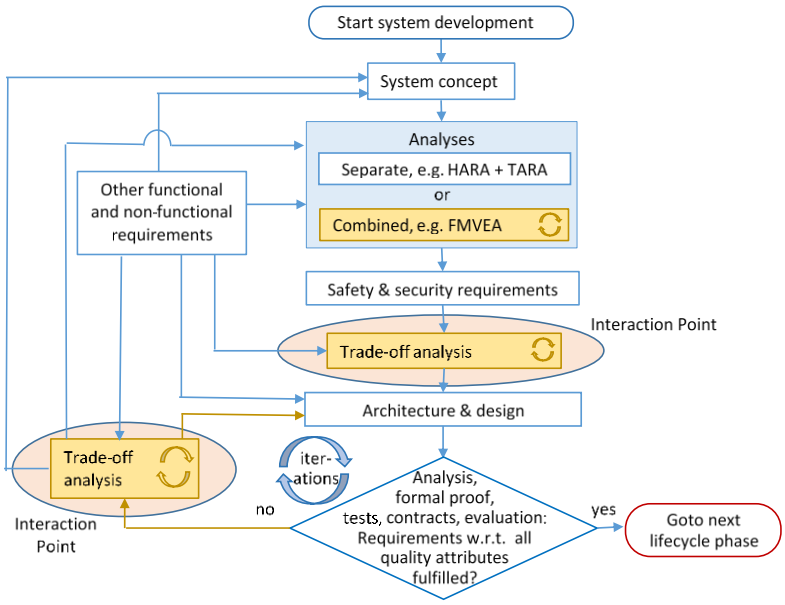

Fig. 3. The iterative lifecycle process modelled in WEFACT

The AQUAS approach allows separate as well combined processes for the individual quality attributes under consideration (e.g. safety, security, performance); at least in the case of separate processes, after performing them, an interaction point is needed to analyze the trade-offs between the potentially contradictory quality attributes treated in the separate parallel processes.

Co-engineering Example in the Industrial Drives Use Case

Within the scope of AQUAS, the shown approach is applied to five very relevant application domains: Air Traffic Management, Medical Devices, Railway, Space and Industrial Drives. Industrial drives are the backbone of many automated industrial processes. Motion control is an essential part for machinery construction and industrial automation. Motion control platforms aim to precisely control electric motors (e-motor) under consideration of safety and security requirements. They are usually realized as Programmable Logic Control (PLC) applications based on microcontroller, FPGA and ASIC solutions. Typical applications are wood/ceramics/glass/stone processing, handling systems, packaging, plastics and textile machines, milling machines, lathes, handling systems, grinders, laser processing, storage and retrieval machines, extruders, winders, rolling machines, tooling machines and many more.One suitable example out of the domain is an FPGA-based generic motion control platform. This demonstrator acts as a test vehicle for piloting the AQUAS methodology. In particular, a virtual prototype of a motion control system is developed which will enable upfront performance considerations and assessment of safety and security features without having the live system at hand.

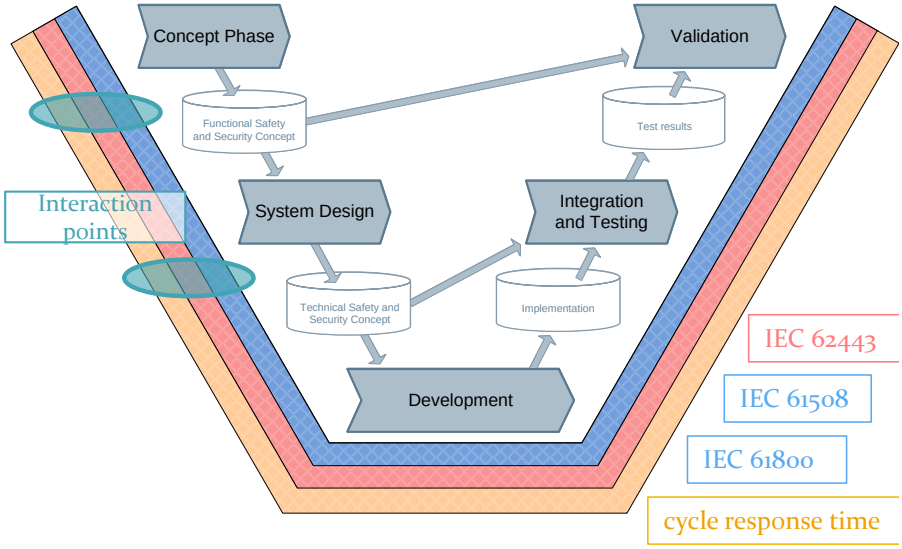

Fig. 4. Generalized PLC in the industrial drives use case for piloting co-engineering

The use case follows the generalized product life-cycle (PLC) depicted in **Fig. 4**. It has the advantage of being domain-independent and small in size – making it easier to integrate it into a fully-blown PLC. In the following we are giving an enhanced flow (we call it AQUAS Life Cycle (ALC)) as an example for co-engineering in the Concept Phase (see Fig. 5). Note that the evaluation of that flow is one of the research questions in AQUAS. When the applied concept proves to be advantageous, other phases will be modelled as well in this style at later phases of AQUAS.

Co-engineering with Safety, Security and Performance in the Concept Phase

The major *goal* is to have a well-balanced and stable set of safety-security-performance requirements in place that will hold in subsequent PLC phases.

ALC start:

We start with an initial set of artefacts, including functional, safety, security, other non-functional requirements and a preliminary system architecture.

ALC Step 1 – Safety/Security/Performance Analysis:

The system is analyzed concurrently by analysis activities with specialized methods/tools with focus set on the system attributes safety, security or performance. **Fig. 5** depicts such analysis activities paradigmatically: Two for safety analysis, two for security analysis and one performance analysis activities. Applying more than one distinct analysis activities for the same system attribute has the advantage of possibly discovering new requirements and contradicting requirements that might have been undetected by applying only a single analysis activity. The goal for each analysis activity is to create a refined set of requirements.

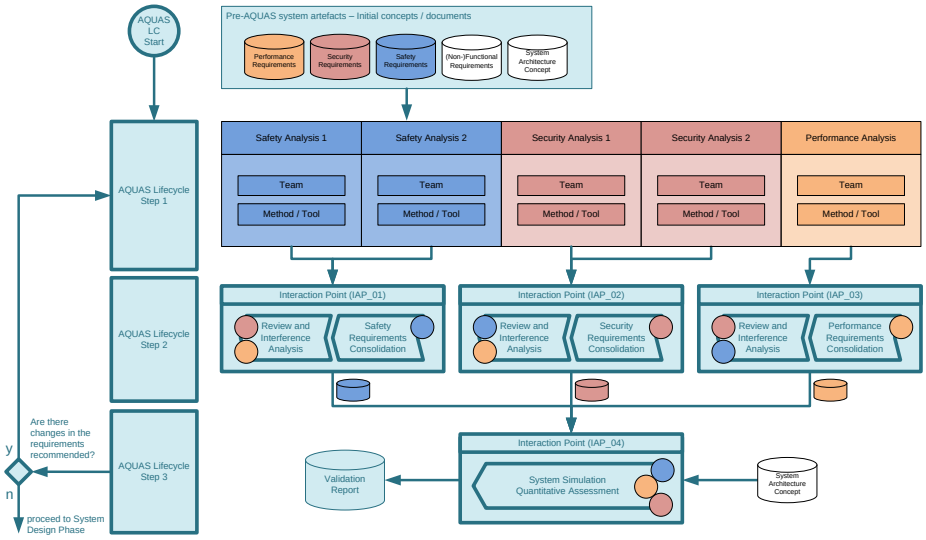

Fig. 5. Co-engineering example with Interaction Points in the Concept Phase

ALC Step 2 – Consolidation, Review, Interference - (IAP_01 to IAP_03)

Results emerging from the various analysis methods with the same attribute focus have to be consolidated and then checked for interference with requirements of other system attributes. In AQUAS we approach that by introducing the concept of *Interaction Points (IAP)*.

The resulting requirements collections that have the same attribute (e.g. safety) from the previous phase are ***consolidated***. During the consolidation activities the requirements coming from the different analysis methods and tools are merged, duplicates are eliminated and contradicting requirements identified. These activities are done by at least one additional expert from the same analysis attribute (e.g. safety requirements are consolidated with a safety analysis expert).

When all requirement consolidations are finished, then each requirement collection is reviewed and analyzed for ***interference*** by expert from the two other analysis attributes (arising discussions may still include the expert(s) from the originating requirements collection). For example the safety requirements collection is reviewed by at least one expert with security-, and one expert with performance background. In this context, interference analysis means a discussion-based analysis of requirements, where requirements that influence each other are identified and marked (linked to each other). The procedure for security and performance is analogous.

Consolidation, review and interference analysis activities are combined in interaction points, with each interaction point has a focus on either safety or security or performance. The outcomes of the interaction points are requirement collections for safety, security and performance.

ALC Step 3 – Validation (IAP_04)

In this phase the goal is to give a statement (by quantitative assessment) on the condition and the validation of the current requirements collection and the system architecture concept. The system is modelled based on the currently, preliminary,

system architecture concept. Some information on the system is not matured in this early PLC phase. Assumptions have to be made, e.g. timing information for different system components. The better the assumptions are the more accurate simulation results and statements on the validation of the current system architecture against the current collection of requirements will be. That interaction point might be triggered again in later phases of the PLC, when more accurate information is available. Such a re-triggering might reinforce the system architecture concept and requirements when previously taken assumptions were sufficient, or reject the current system architecture with its requirements in the other case.

The output of that interaction point is a validation report that can be used to give recommendations on the current system architecture and requirements. That information is crucial for either a re-iteration of steps 1 to 3 or the advancement to the next PLC phase ("System Design").

ALC – Criteria for Transitioning to the next phase

After ALC step 3, a transition to the next phase and thus leaving the iteration loop of steps one to three, may be done if:

- There are no more changes to the requirements recommended by any ALC step.
- Each analysis method (ALC Step 1) was run at least twice.
- The output of each interaction point is the same as the one from the previous round, i.e. there is no more new knowledge gained.

These conditions might cause some overhead, but with these in place there is a higher confidence for having a contradiction-free and complete set of requirements. Analysis methods have to be re-run after interaction points if the interaction points change the requirement collections, because requirements added, changed or dismissed by a concurring analysis method might be evaluated differently by the other analysis method in place. This means that only after a complete run through steps 1 to 3 without any changes to requirements and system architecture the concept gives better confidence.

4 Conclusions and outlook

Currently (May 2018) the AQUAS project has just passed the first project year out of three; the consortium is gaining experience and working on the refinement of the interaction point and co-engineering approach, which has been shortly presented in the previous chapter. AQUAS goes beyond the scope of SAHARA which addresses only the analysis phase, and it integrates the FMVEA approach as a possible combined co-engineering method.

By using WEFACT, AQUAS provides flexibility w.r.t. manual and tool-based processes in the workflow, without any further requirements for tool adaptation. As co-engineering phases can be adapted to the AQUAS approach individually while the rest of the lifecycle model continues using the established company practices, a transition to the AQUAS approach can be performed step-by-step. This results in the ad-

vantage of a smooth transition from legacy to new processes, avoiding risk and cost of a changing to entirely new lifecycle processes.

This distinguishes the AQUAS approach also from the related AMASS project [AMASS], which focuses on multi-concern assurance and develops a model-based, open-source assurance framework with mainly integrated tools and the possibility for external tools to be used via adapters.

In the remaining two years of the project runtime, exemplary processes for the AQUAS approach will be implemented and evaluated, and more development phases will be considered. With experience from the case studies, we expect to have a clearer view on how the interaction points shall be organized and where in the product lifecycle they shall be located. This will enable us to address, apart from Co-engineering and Product Life-Cycle, the third main goal of AQUAS – Influencing Standardization. As mentioned, there are already comparable approaches described in standards, but we expect to contribute with detailed guidance from experience

Acknowledgements: The work published here is based on research in the AQUAS project that has been funded by the ECSEL Joint Undertaking and the Austrian Ministry for Transport, Innovation and Technology (BMVIT) in the program "ICT of the Future" and the Austrian Research Promotion Agency (FFG) under Grant Agreement number 737475.

References

1. IEC_61508-1_Ed.2.0. (2010). Functional safety of electrical/electronic/programmable electronic safety-related. Part 1-6.
2. BBC report "Hack attack causes 'massive damage' at steel works", http://www.bbc.com/news/technology-30575104 (access May 2018)
3. IEC 61800 Adjustable speed electrical power drive systems Part 1-7
4. ISO/IEC 15408-1:2009 Information technology -- Security techniques -- Evaluation criteria for IT security -- Part 1: Introduction and general model
5. ISO/IEC 27000:2018 Information technology - Security techniques - Information security management systems - Overview and vocabulary
6. ISO - International Organization for Standardization. ISO 26262 Road vehicles Functional Safety Part 1-10, 2011.
7. Riccardo Scandariato, Kim Wuyts, Wouter,Joosen, Microsoft Corporation. A descriptive study of Microsoft's threat modeling technique, http://scholar.google.at/scholar_url?url=https://lirias.kuleuven.be/bitstream/123456789/424554/1/rej-stride.pdf&hl=de&sa=X&scisig=AAGBfm37KabrVfVeveavNgju0SLcmf351w&nossl=1&oi=scholarr (access May 2018.
8. Georg Macher, Harald Sporer, Eric Armengaud and Christian Kreiner : SAHARA: A Security-Aware Hazard and Risk Analysis Method (2015)
9. The_MERgE_Project, "D3.4.4: Recommendations for Security and Safety Co-engineering", http://www.merge-project.eu/merge-deliverables, 2016.
10. C. Schmittner, T. Gruber, P. Puschner and E. Schoitsch, "Security Application of Failure Mode and Effect Analysis (FMEA)," Safecomp, Proceedings, 09 2014.

11. IEC 60812 “Analysis techniques for system reliability – Procedure for failure mode and effects analysis (FMEA)”, 2nd ed., 2006.
12. https://www.techrepublic.com/blog/it-security/the-cia-triad/ (access 26 May, 2018)
13. SAE J3061, “Cybersecurity Guidebook for Cyber-Physical Vehicle Systems", 2016.
14. IEC/TS_62443-1-1, Industrial communication networks – Network and system security – Part 1-1: Terminology, concepts and models, 2009.
15. IEC_62443-2-1, Industrial communication networks – Network and system security – Part 2-1: Establishing an industrial automation and control system security program, 2010.
16. IEC_62443-3-1, Industrial communication networks – Network and system security – Part 3-1: System security requirements and security levels, Draft.
17. IEC_62443-3-2, Industrial communication networks – Network and system security – Part 3-2: Security risk assessment and system design, Draft.
18. IEC_62443-3-3, Industrial communication networks – Network and system security – Part 3-3: System security requirements and security levels, 2013.
19. IEC 62443-4-1 Security for industrial automation and control systems - Part 4-1: Secure product development lifecycle requirements, 2018.
20. IEC 62443-4-2 Industrial communication networks - Security for industrial automation and control systems - Part 4-2: Technical security requirements for IACS components, Draft, 2017.
21. Industry 4.0 - Wikipedia, https://en.wikipedia.org/wiki/Industry_4.0.
22. Hackers trigger yet another power outage in Ukraine, ars TECHNICA, https://arstechnica.com/information-technology/2017/01/the-new-normal-yet-another-hacker-caused-power-outage-hits-ukraine/.
23. C. Schmittner, Z. Ma, P. Puschner: "Limitation: Modelling Adversary Effort and Improvement of STPA-Sec for Safety and Security Co-analysis"; Vortrag: SAFECOMPSupply Chain Risks," 2016 12th European Dependable Computing Conference (EDCC), Gothenburg, 2016, Trondheim; 20.09.2016; in: "Computer Safety, Reliability, and Security SAFECOMP 2016 Workshops, ASSURE, DECSoS, SASSUR, and TIPS", Springer, (2016), ISBN: 978-3-319-45480-1; S. 195 - 210.
24. IgorNaiFovino, MarceloMasera, AlessioDeCian: Integrating cyber attacks within fault trees, in Elsevier, Reliability Engineering and System Safety 94 (2009), pp 1394–1402.
25. The AMASS project. https://www.amass-ecsel.eu/
26. I. Gashi, A. Povyakalo and L. Strigini, "Diversity, Safety and Security in Embedded Systems: Modelling Adversary Effort and Supply Chain Risks," 2016 12th European Dependable Computing Conference (EDCC), Gothenburg, 2016, pp. 13-24.
27. The AQUAS project https://aquas-project.eu/
28. The Arrowhead project, http://www.arrowhead.eu/