Deliverable 1.3

Report on the Evolution of Co-Engineering Standards

This project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737475. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and Spain, France, United Kingdom, Austria, Italy, Czech Republic, Germany.

DISSEMINATION LEVEL		
X	**PU**	Public
	CO	Confidential, only for members of the consortium (including the Commission Services)

COVER AND CONTROL PAGE OF DOCUMENT	
Project Acronym:	AQUAS
Project Full Name:	Aggregated Quality Assurance in Systems
Grant Agreement No.:	737475
Programme	ICT-1: Cyber-Physical-Systems
Instrument:	Research & innovation action
Start date of project:	01.05.2017
Duration:	36 months
Deliverable No.:	D1.3
Document name:	Report on the Evolution of Co-Engineering Standards
Work Package	WP1
Associated Task	Task 1.3
Nature [1]	R
Dissemination Level [2]	PU
Version:	1.0
Actual Submission Date:	31-10-2018
Contractual Submission Date	31-10-2018
Editor: Institution: E-mail:	John Favaro Intecs john.favaro@intecs.it

[1] **R**=Report, **DEC**= Websites, patents filling, etc., **O**=Other

[2] **PU**=Public, **CO**=Confidential, only for members of the consortium (including the Commission Services)

Change Control

Document History

Version	Date	Change History	Author(s)	Organisation(s)
0.1.0	28-08-2018	Document template creation and initial contents	John Favaro, Erwin Schoitsch, Christoph Schmittner	INT, AIT
0.1.1	16-09-2018	Medical Use Case contribution	Ricardo Ruiz	RGB
0.1.2	18-09-2018	Space Use Case contribution	Jaime González Martínez	TAS
0.1.3	18-09-2018	Railway UC Contribution	Nicolas Ayache	Clearsy
0.1.4	01.10.2018	OMG / MARTE contribution	Laurent Rioux	TRT
0.1.5	05.10.2018	ECSS, OMG, INCOSE contributions	Silvia Mazzini	INT
0.1.6	08.10.2018	Human factors in medical use case	Marwa Gadala, Lorenzo Strigini	CITY
0.1.7	10.10.2018	Safety / Security / Performance related V&V processes	Daniel Kästner, Reinhold Heckmann	AbsInt
0.1.8	15.10.2018	Industrial Drive UC Contribution	Martin Matschnig	Siemens / SAG
0.1.9	17.10.2018	ISO, IEC and IoT Standards recent evolutions added	Erwin Schoitsch, Christoph Schmittner	AIT
0.2	22.10.2018	First version for internal review	John Favaro	INT
0.2.1	22.10.2018	Contribution on Arcadia	Charles Robinson	TRT
0.2.2	25.10.2018	Reviewed version	Martin Matschnig	Siemens / SAG
0.2.3	25.10.2018	Further contributions on OMG	Silvia Mazzini	INT
0.2.4	26.10.2018	2nd version for internal review	John Favaro	INT
0.2.5	29.10.2018	New figure and text for Human Factors contribution	Marwa Gadala, Lorenzo Strigini	CITY
1.0	30.10.2018	Final version for delivery	John Favaro	INT

Distribution List

Date	Issue	Group
	Document alignment	application.domain.leads@aquas-project.eu
	Final Version	application.domain.leads@aquas-project.eu, leaders@aquas-project.eu

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

Executive Summary

The interplay between dependability attributes such as safety / security / performance is being increasingly accepted by all involved stakeholders and discussions on how to react to this development in standardization is ongoing. Related standards in multiple domains are currently under revision or (especially in the case of security standards) for the first time under development. For IoT and the increasingly open and dynamic systems, it will be necessary to regulate and consider multiple dependability attributes. Due to the continuing involvement of AQUAS partners in standardization activities, AQUAS has a window of opportunity to influence standardization. The AQUAS work on the evolution of co-engineering standards takes as its point of departure a body of existing work that is relevant to the project in various ways:

- Existing standards that directly govern the development of the demonstrators in the AQUAS use cases;
- Existing standards that are not specifically relevant to the selected use cases in AQUAS, but which have already addressed aspects of co-engineering relevant to AQUAS objectives;
- Ongoing standardisation initiatives which might not necessarily have produced results yet, but can be a source of guidance for AQUAS.

In multiple domains which already have safety as an established property, security is becoming a new issue. The introduction of performance into co-engineering is extremely recent, and few standards are treating it to date. Thus, we must rely currently on experience with cybersecurity and safety dimensions in the standards developing organizations to understand how they are currently addressing the topic of co-engineering. The establishment of a unified risk assessment / management regime in co-engineering for all three dimensions remains a significant challenge. Furthermore, each of the three co-engineering dimensions treats “best practices” in development in different ways, making it difficult to harmonize the standardisation of development according to each of the three dimensions. This also remains a challenge for co-engineering standards.

The AQUAS consortium is well aware that it will not always be possible to synchronize its standards-influencing efforts with the windows of opportunity that will arise during the evolutionary cycles of the standardization groups. Nevertheless, activities can be engaged that are useful even in the absence of perfect synchronisation with the standards renewal cycles. The following approaches are under consideration:

- Reports and change request packages valid for future revision cycles;
- Presentations to standards committees and working groups;
- Guidelines for the usage and interpretation of standards in particular ways.

In order to channel upcoming standards evolution activities in the project into the most effective paths, the project is putting into place mechanisms for tracking progress toward the achievement of the objectives of these activities.

Another activity recently launched is the harmonization of standards terminology, to ensure that any proposals resulting from AQUAS work are consistent with the directions being taken by the standardisation groups, in order to avoid obstacles that result from extreme mismatches both at the conceptual and at the terminological levels.

1 Introduction: Current Challenges in Standards Evolution

The interplay between dependability attributes such as safety / security / performance is being increasingly accepted by all involved stakeholders and discussions on how to react to this development in standardization is ongoing. Related standards in multiple domains are currently under revision or (especially for security standards) for the first time under development. For IoT and the increasingly open and dynamic systems, it will be necessary to regulate and consider multiple dependability attributes. Due to the ongoing involvement of AQUAS partners in standardization activities, AQUAS has a window of opportunity to influence standardisation.

However, it is still difficult to address such issues in a cross-domain way. Different domains have established safety standards, and security standards are partially designed to interact and extend existing standards. Therefore, we do not expect much overlap between the domains in standardization. A positive counterexample is the acceptance of IEC 62443 [10] as a template for future cybersecurity standards for additional domains like railways.

Another challenge is related to the current lack of focus on **performance** considerations in standardisation activity, in contrast to the increasing interest in safety / security co-engineering. This issue is finally being given a push by the emergence of autonomous applications in multiple domains, and standardization groups are beginning to realize the relevance and importance of the performance dimension. But the challenge is significant and open-ended – representing at the same time a potential opportunity for AQUAS.

Besides multi-concern standardization, tool interoperability will also play an important role in the success of the AQUAS activities. Only accepted and well-specified interoperability standards will allow the seamless interoperability between AQUAS internal and external tools and support the automation of co-engineering processes.

Each of the above-described challenges will be addressed in more detail in the sections that follow.

The AQUAS project has set for itself the following objectives (the numbers correspond to those mentioned in the Description of Work):

- **Objective 9**: Contribute to the improvement of standards to address co-engineering, by submission of change requests to at least 1 standard for each of the AQUAS use case domains.
- **Objective 10**: To promote awareness and bring results of AQUAS into at least two international standards in the functional safety and security area with respect to safety, security and performance co-engineering.
- **Objective 11**: To influence actively two international standardization groups focused on frameworks for the coordination of safety, security and reliability of automation.
- **Objective 12**: To promote awareness and bring results of AQUAS into at least two other international engineering standards, such as OMG, or FMI.

In the following sections, a basis for the AQUAS approach to meeting these objectives will be identified.

2 Standards influencing AQUAS

The AQUAS work on the evolution of co-engineering standards takes as its point of departure a foundation of existing work that is relevant to the project in various ways:

- Existing standards that directly govern the development of the demonstrators in the AQUAS use cases. These represent “bedrock” in the project: real standards that have a concrete effect on real application development;
- Existing standards that are not specifically relevant to the selected use cases in AQUAS (for example, they might be in different domains), but which have already addressed aspects of co-engineering that are relevant to AQUAS objectives and which could help to achieve those objectives;
- Standardisation initiatives that are currently ongoing, but have not necessarily produced results yet. These can provide valuable guidance in how to proceed in our own work.

The following sections further elaborate on these points.

2.1 Standards and the AQUAS Use Cases

2.1.1 UC1 Air Traffic Management

The ATM use case offers one of the first opportunities for the AQUAS project to break potential new ground in the examination of **performance** as a first-class citizen in the treatment of co-engineering in standards. As reported earlier [4], the so-called SWIM profiles [1][2][3] developed within the SESAR projects are the primary sources of requirements on performance and security. But these profiles were originally developed in the context of commercial aviation, long before the advent of remote-controlled, unmanned vehicles (drones), and therefore placed their principal emphasis on human safety. With the increased use of telecommunications, and the attendant need for ultra-reliable low-latency communications and enhanced broadband communications capacity, **performance** has become an integral factor in requirements engineering for this category of application. The mission critical nature of both military and civilian drone applications, combined with the reliance on wireless communications, has also promoted security aspects to first-class citizenship. Now true co-engineering of safety, performance, and security is needed for this class of application.

Therefore, the AQUAS analyses of the trade-offs between safety, performance, and security in the ATM use case will provide a unique opportunity to integrate co-engineering of Unmanned Aerial Vehicles into the SWIM standardisation infrastructure.

2.1.2 UC2 Medicine

This section updates the similar section in Deliverable 2.2.2 [5] by adding some additional standards that are being considered; updates are in **boldface**.

Medical devices are strongly regulated in the European Union. The current medical devices directive 93/42/EEC is still applicable, but a new medical device regulation 2017/745 was approved in 2017 and will be fully applicable in 2020. These regulations specify the requirements that a medical device legally placed on the European market must satisfy, and state that a sufficient condition for satisfying requirements is compliance "with the relevant national standards adopted pursuant to the harmonized standards", which makes such compliance with the latter the natural path for industry to follow. Standards currently harmonised with the medical devices directive 93/42/EEC can be found at [9].

The main standard considered in the medical use case is EN 60601-1:2006, titled, "Medical Electrical Equipment. General requirements for basic safety and essential performance". This standard is a

general safety standard applicable to any type of medical device. It has several associated collateral standards that are mandatory when applicable. In this use case, the relevant ones (which are technically equivalent to the international IEC 60601-standards series) are:

- EN 60601-1-2: 2007. "Collateral standard: Electromagnetic compatibility"
- EN 60601-1-6: 2010. "Collateral standard: Usability"
- EN 60601-1-8: 2007. "Collateral standard: General requirements, tests and guidance for alarm systems in medical electrical equipment and medical electrical systems"
- EN 60601-1-10: 2008. "Collateral standard: Requirements for the development of physiologic closed-loop controllers"

There are also particular standards related in the standard series, that are related to the safety of specific types of medical devices. The standards applicable to this use case are:

- EN 60601-2-10:2015. "Particular requirements for the basic safety and essential performance of nerve and muscle stimulators". This standard specifies particular requirements related to the measurement of neuromuscular transmission.
- EN 80601-2-30: 2010. "Particular requirements for the basic safety and essential performance of automated non-invasive sphygmomanometers". This standard specifies particular requirements related to the non-invasive measurement of blood pressure.

Although the IEC 60601 series also covers aspects such as product life cycle, there are specific standards that directly address such aspects in more detail:

- EN 62304:2006. "Medical device software - Software life cycle processes". The EN 62304 standard requires following the well-known V-model for the software life cycle processes of a medical device, but the rest of the standards normally include specific requirements not related with the product life cycle.
- **ISO 13485:2016. "Medical Devices - Quality Management Systems - Requirements for Regulatory Purposes"**
- **IEC 61508: "Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems" covering aspects to be considered when electrical/electronic/programmable electronic systems are used to carry out safety functions"**

In AQUAS we have paid special attention to the effects of usability standards on safety, performance and security:

- EN 62366-1:2015. "Medical devices - Application of usability engineering to medical devices" and as guidance to this standard: **IEC/TR 62366-2:2016. "Guidance on the Application of Usability Engineering to Medical Devices".**

For issues related to symbols and labelling of medical devices, the following standards are also considered in this use case:

- **ISO 15223-1:2016. "Medical Devices – Symbols to be used with medical device labels, labelling and information to be supplied".**
- **IEC 60878:2015. "Graphical symbols for electrical equipment in medical practice"**

Some of the aforementioned standards have issued several amendments since their approval and are periodically revised.

According to medical device regulation in the European Union, it is mandatory to perform a risk assessment starting with the initial design of a medical device and during all phases of its lifecycle. This risk assessment must be performed in compliance with the following standard:

- EN ISO 14971:2012. "Medical device - Application of risk management to medical devices"

There are **no harmonised standards related to cybersecurity**, but some security-related standards that should be considered are:

- ISO 27799:2016. "Health informatics - Information security management in health using ISO/IEC 27002". It is a guideline for the application of the ISO/IEC 27002 standard to health informatics. ISO 27799 and ISO/IEC 27002 taken together define what is required in terms of information security in healthcare to maintain the confidentiality, integrity and availability of personal health information.
- ISO/IEEE 11073. "Health informatics - Medical health device communication". This set of standards enables communication between medical, health care and wellness devices with external computer systems.
- HL7 Standards. This family of standards is related to clinical information exchange and is widely used for the communication between medical devices and Patient Data Management Systems (PDMS) in hospital environments.

2.1.3 UC3 Railway

The basic safety-related standards for railways are EN 50126, EN 50128, EN 50129 and EN 50159 (See [6]).

- EN 50126 – Railway applications – The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS) – Part 1: Basic Requirements and generic process.
- EN 50128 - Railway applications - Communication, signalling and processing systems - Software for railway control and protection systems.
- EN 50129 - Railway applications – Communications, signalling and processing systems – Safety related electronic systems for signalling.
- EN 50159 - Railway applications - Communication, signalling and processing systems. Safety-related communication in transmission systems.

In all these standards “security” is not mentioned except in the context of physical access, based on the traditional isolation of railway signalling and communication systems from regular public systems. With increased use of public facilities and wireless communication and control systems, e.g. the European Train Control System, the “security-aware safety” considerations in standardization are now starting also in the railway sector. DKE, in Germany, is integrating requirements from IEC 62443 in the railway standards (proposal, addressing EN 50129 and EN 50159 issues) by DIN VDE V 0831-104 “Electric signalling systems for railways – Part 104: IT Security Guideline based on IEC 62443”. Work is now transferred to CENELEC TC9X and tackles the cybersecurity issue not only from the signalling safety and communication viewpoint, but also from the top-level viewpoint (see Figure 1).

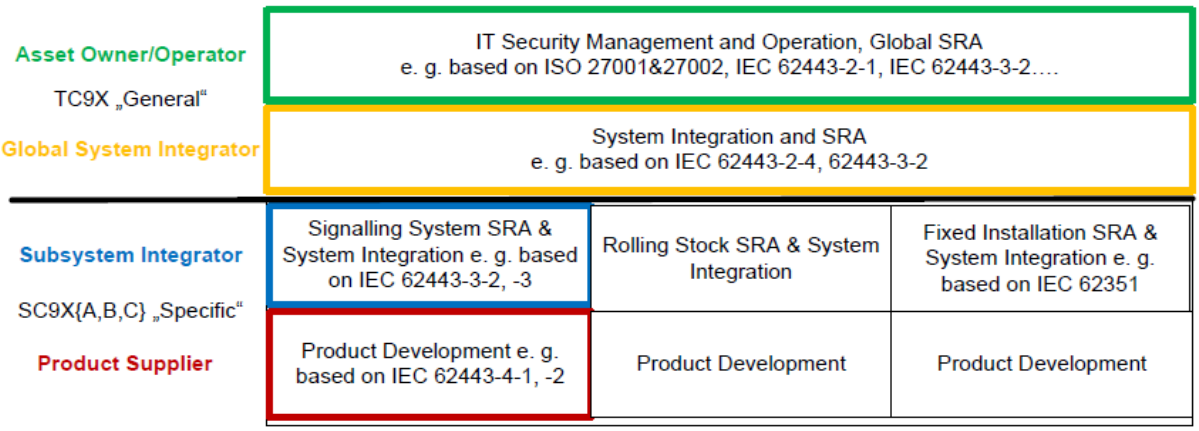

Figure 1: Considering Cybersecurity in Railway Standards

An example of an adaptation of other railway standards, e.g. for rolling stock, to the functional safety standards as already common in signalling, may be found in the new EN 50657 “Railways Applications – Rolling stock applications – Software on Board Rolling Stock”. After a withdrawal of the old version, a new one has been issued (1 December 2017). It replaces EN 50128 for rolling stock and takes over most concepts from that standard. It is a bit less strict with respect to low safety integrity, for instance lower documentation requirements, and in this sense SIL0 has been renamed to “Basic integrity”.

Based upon discussions with experts, it appears that the aforementioned standards cover development processes regarding safety and security in a rather open way: the objectives are described, but there are no constraints on the means. Achieving the required level of safety or security entirely depends on the safety or security demonstrations. For instance, if one is using tests in a part of the project, they will have to provide a demonstration as to why they satisfy the considered requirements. But the same approach would apply if another technique, say formal methods, was used.

2.1.4 UC4 Industrial Drive

As stated in earlier deliverables, the governing **safety-related** standard for this use case is IEC 61508 – “Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems”. [11] With respect to functional safety, the standard “Functional safety for adjustable speed electrical power drive systems” (IEC 61800-5-2) defines safety functions performing monitoring and/or safety-relevant controlling tasks, e.g. Safe torque off (STO). It was discussed previously [7] that, for example, this standard contains requirements (such as uncontrolled communication during an emergency situation) that could create safety and security interference.

However, the newer developments concern **cybersecurity standardisation** in this sector. As reported earlier in the AQUAS project, IEC 61508 is now in a relatively stable phase, having recently undergone a significant revision (Second Edition in 2010), in which some aspects of cybersecurity are touched upon, particularly in the section on Hazard Analysis. The new development to report is the planned set of IEC 62443 standards.

IEC 62443 “Industrial communication networks -Security for industrial automation and control systems” 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). Figure 2 gives an overview on the planned set of standards and technical reports.

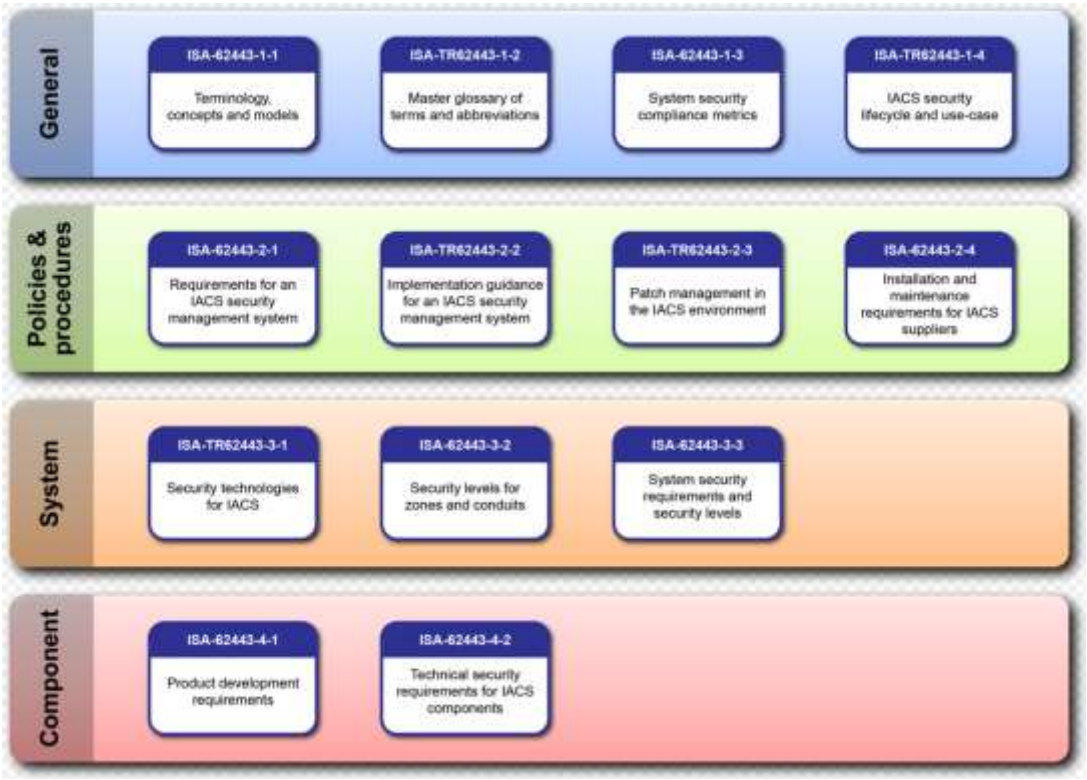

Figure 2: Planned parts of IEC 62443 (*source: IEC*)

The standards define procedures for implementing electronically secure IACS. Their guidance applies to end-users (i.e. asset owner), system integrators, security practitioners, and control systems manufacturers responsible for manufacturing, designing, implementing, or managing IACS.

In these standards, the main concepts relevant for the AQUAS approach and in particular for the Industrial Drive use case are:

- The zone-conduit concept described and the respective workflow in IEC 62443-3-2 and IEC 62443-3-3, which allows a structured cybersecurity risk analysis with targeted measures for the safety-critical zones, and
- The concept of security levels (SL): SL-T (target SL), SL-C (SL capabilities), and SL-A (achieved SL) introduced also in IEC 62443-3-2.

The IEC 62443 series of standards is intended to be used across industrial control segments and has been approved by many countries. The concepts have also influenced the railway domain and have commonalities with the AQUAS approach.

2.1.5 UC5 Space Multicore

The European Cooperation for Space Standardization (ECSS) represents a cooperative effort of the European Space Agency (ESA), national space agencies and European industry associations for the development of a coherent, single set of consistent space standards for use by the entire European Space Community. The objective of creating this organization was to produce standards to be used throughout the European space business. Therefore, the European Space Agency (ESA) contractors must adhere to the standards created by this organization [8].

The result of this effort is the ECSS series of Standards (ST), Handbooks (HB) and Technical Memoranda (TM) organized in four branches:

- M: Management Standards

- Q: Product Assurance Standards
- E: Engineering Standards
- U: Usability Standards

Among them, the most relevant standards for the AQUAS project are:

- ECSS-Q-ST-30 “Space product assurance (dependability)” defines the requirements for a dependability assurance programme in space projects. This standard calls for the use of dependability analysis techniques, tailored to match the generic requirements in each project, to address the hardware, software and human functions composing the system.
- ECSS-Q-ST-40 “Space product assurance (safety)” defines the safety programme and the technical safety requirements for space projects.
- ECSS-E-40 “Software” focuses on space software engineering process requirements and their expected outputs, putting a special emphasis on the system-software relationship and on the verification and validation of software items.
- ECSS-Q-ST-80 “Space product assurance – Software product assurance” defines a set of software product assurance requirements to be used for the development and maintenance of software for space systems. The objective is to provide adequate confidence to the customer and to the supplier that a software (developed or reused) satisfies its requirements throughout the system lifetime. In particular, that the software performs properly and safely in its operational environment and meets the quality objectives agreed for the project. The requirements defined in the ECSS-Q-ST-80 standard deal with quality management, process definition and quality characteristics of software products during the whole project life cycle.

A gap analysis of these standards is presented in a subsequent section (Section 3.2.1).

2.2 Transversal standards activity influencing AQUAS

There is current standardisation activity that is not immediately associated with a particular domain, and therefore is to be analysed separately. Primarily this concerns tool interoperability, and its relationship to tools being employed in AQUAS such as medini analyse [17] and CHESS [18].

2.2.1 OMG standards activity

The OMG (Object Management Group) is an international, open membership, not-for-profit organization for the development of technology standards. OMG standards are driven by vendors, end-users, academic institutions, and government agencies. OMG Task Forces develop enterprise integration standards for a wide range of technologies.

The OMG technology adoption process [23] is quite elaborate: it starts with a Request for Proposals (RFP). An RFP is a statement of industry need and an invitation to the software supplier community to provide a solution, based upon requirements stated within. The process of identifying need is a culmination of experience within an OMG technical group (be it a Task Force, a Special Interest Group or a Subcommittee) and solicitation of industry recommendation. Any Contributing, Domain or Platform Member of the OMG in good standing may propose specifications for adoption by OMG in response to an RFP. The initial submissions in response to an RFP are developed and presented to the sponsoring Task Force, which provides feedback to the submitter(s), a determination is made for the need for revised submissions. Revised submissions can be iterated more times until the approval by the OMG Membership is reached, and a further finalization step is issued.

Several standards and initiatives at OMG are related to AQUAS activities. The OMG “modelling standards”, such as the Unified Modelling Language (UML), and in particular the Systems Modelling

Language (SysML) and MARTE (Modelling and Analysis of Real-time and Embedded systems), are at the very heart of AQUAS tool interoperability activities.

SysML is a general-purpose modelling language for systems engineering applications. It supports the specification, analysis, design, verification and validation of a broad range of systems and systems-of-systems.

MARTE is a UML profile defining foundations for model-based description of real time and embedded systems. These core concepts are then refined for both modelling and analysing concerns. Modelling parts provides support required from specification to detailed design of real-time and embedded characteristics of systems. In addition, facilities to annotate models with information required to perform specific analysis are provided. MARTE focuses especially on performance and schedulability analysis. It is a general framework for quantitative analysis which may be specialized for other kind of analysis.

They constitute the formal basis for specifying the artefacts to be modelled, annotated for analysis and exchanged among many of the AQUAS tools, and are being followed closely by some AQUAS partners, such as TRT, CEA and Intecs.

2.2.2 OSLC

In the context of software and system interoperability and integration, the Open Services for Lifecycle Collaboration (OSLC) initiative is a joint effort between academia and industry to improve data sharing and interoperability among applications by applying the Linked Data principles: "*1) Use URIs as names for things. 2) Use HTTP URIs so that people can look up those names. 3) When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL) and 4) Include links to other URIs, so that they can discover more things*". See Figure 3.

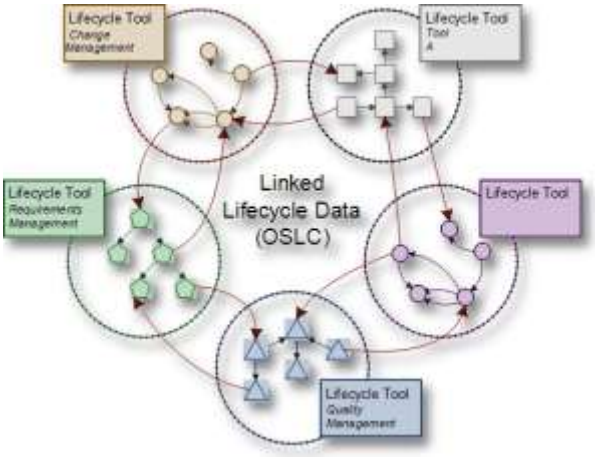

Figure 3: OSLC Concept of Linked Lifecycle Data

Led by the OASIS OSLC working group[3], OSLC is heavily based on web standards (as can be seen in the description above), in particular RDF for providing a common *data model*, and HTTP for providing a common *protocol*.

The objective, particularly relevant to AQUAS, is to provide Product Lifecycle Management tools with agreement on how and which data to share. One issue is that the artefacts generated during the AQUAS lifecycle are not necessarily already defined, in part because of the CPS nature of AQUAS relevant systems. For example, simulation models or physical circuits are examples of potential

[3] http://www.oasis-oslc.org/

artefacts whose OSLC "resource shape" is not yet defined. These may have to be defined during the AQUAS project, depending on their appearance in the use cases.

Essentially, by taking advantage of the Linked Data principles and Web standards and protocols, the OSLC effort is attempting to create a family of web-based specifications for products, services and tools that support all the phases of the software lifecycle. A number of industry platforms such as PTC Integrity[4], Siemens Team Center[5], IBM Jazz Platform[6] or HP PLM[7] are now offering OSLC interfaces for different types of artefacts.

Note, however, that data exchange does not necessarily imply integration. From service providers to data items, an integration strategy is required to represent, store, search and coordinate collaboration between software artefacts metadata and contents. In this light, the OSLC initiative is currently following this approach, and is having an impact on the main players in the software and systems industry. Nevertheless, it only covers a restricted type of artefacts and some crosscutting and basic services for reuse, such as indexing or retrieval, must be provided by all third-parties. Within AQUAS, these restrictions will be taken into consideration with regard to the effort that would be required to fill the existing gaps in the OSLC conceptual and practical offerings, in order to decide on the relative costs and benefits of an OSLC oriented solution to interoperability of AQUAS tools.

It should be mentioned that the outcomes of the Horizon 2020 Support Action CP-SETIS on tool interoperability, which are based on the OSLC approach and have been extended to further interoperability standards and guidelines as a multi-standards platform, is supported by ARTEMIS-IA as a small community project (the IOS-ICF, *Interoperability Coordination Forum*). AQUAS partner AIT was a partner in CP-SETIS driving the standardization agenda for CPS, and is further involved in ICF.

Note also that OSLC will be further treated in AQUAS Deliverable D4.2 (November 2018).

2.2.3 INCOSE

The International Council on Systems Engineering (INCOSE) is a not-for-profit membership organization founded to develop and disseminate the interdisciplinary principles and practices that enable the realization of successful systems. INCOSE is focused on producing state-of-the-art work products that support and enhance the Systems Engineering discipline's visibility in the world.

INCOSE promotes a Systems Engineering Vision [24] to inspire and guide the direction of systems engineering across diverse stakeholder communities, which include different engineering disciplines and tool vendors and prepare the System Engineering Handbook [25] to describe key processes and activities performed in systems engineering.

INCOSE, together with the Institute of Electrical and Electronics Engineers Computer Society (IEEE-CS), and the Systems Engineering Research Center (SERC) provides also the funding and resources needed to sustain and evolve the Guide to the Systems Engineering Body of Knowledge (SEBoK) [26] and make it available as a free and open resource to all. The SEBoK provides a compendium of the key knowledge sources and references of Systems Engineering organized and explained to assist a wide variety of users.

2.3 Automotive sector standards activity

AQUAS does not have a use case in the automotive sector. However, there is vigorous ongoing standardisation activity, in great measure due to commercial pressure resulting from the race towards

[4] http://www.ptc.com/application-lifecycle-management/integrity

[5] http://www.plm.automation.siemens.com/en_us/products/teamcenter/

[6] https://jazz.net/

[7] http://www8.hp.com/us/en/business-services/it-services.html?compURI=1830395

vehicle autonomy, advanced intelligent transport system applications, and resolution of automotive cybersecurity issues. Of particular relevance to AQUAS is the remarkable fact that the current automotive standardisation activity involves three standards in evolution *at this time* (Autumn 2018) that address exactly the **three dimensions** treated by AQUAS and require interactions between these properties:

- **Safety**. The first version of ISO 26262 was published in 2011. While the standard was a huge success and adapted by the automotive industry, technological developments like the increased usage of assistant functions, increased connectivity and the rising importance of software required a revision and update of the standard. ISO 26262 Ed. 2 has been scheduled for publication by the end of 2018. The currently available draft contains a section on the interaction between safety and security and a requirement to define communication channels between safety and security. [13]
- **Performance**. Safety of The Intended Functionality – SOTIF: For automated or autonomous vehicles safety is not only endangered by failures in the classical sense, e.g. a hardware element is failing, or a software has a design error, but also by misinterpretations of sensor signals or lacking combination of sensor data and processing. SOTIF is a newly developed standard (ISO PAS 21448 – Publicly Available Specification) which addresses such issues. Of special interest to AQUAS is the fact that **inadequate performance** is explicitly considered in the standard as having potential impact on the other dimensions, in particular safety. This is nearly unique in the current standardisation landscape (although it is likely to become increasingly important, due to the automation of applications in diverse sectors from avionics to robotics and other related transport sectors). The PAS was planned for publication by the end of 2018. [14]
- **Security**. Due to increasing connectivity, V2X communication and the shift of functionality towards software and more complexity that increases the need for Over the Air Updates (OTA), cybersecurity is increasingly important for dependable automotive systems. Recently demonstrated hacker attacks on automotive control systems via maintenance or entertainment channels have highlighted the necessity as well. Therefore SAE, who created already SAE J3061 as Guideline for Automotive cybersecurity engineering, and ISO have joined forces towards an Automotive Cybersecurity Standard (ISO/SAE JWG1, ISO TC22 SC32 WG 11, for ISO 21434). The standard has been scheduled for publication in 2020. Similar to ISO 26262:2018 ISO/SAE 21434 should contain a section for the interaction from safety to security.
- **Extended vehicle standards.** ISO TC22 SC31 has moved the extended vehicle standards, who handle the interaction between the vehicle and its environment (including other vehicles, road signs, sensors on other objects in the traffic flow, web interfaces, etc.) into a new WG 10 (time-critical extended vehicle applications). Sensor interfaces for automated driving functions remain in WG9. To harmonize somehow the standardization activities of various ISO committees towards automated driving and provide recommendations, TC22 has created an AG1 “Automated Driving Ad-hoc Group” (ADAG). In this context, security and safety interaction may become an issue as well.

2.4 Framework-oriented standards activity

Standards are sets of requirements that have to be met in order to consider something “compliant”. Most of AQUAS is concerned with domain specific standards. **Frameworks**, in contrast, are “merely” sets of best practices and reference models. They can be used in the generation of domain-specific standards; they can be used in the absence of agreed standards; they can be used in order to help harmonize different standardisation activities. Indeed, framework-oriented standards are becoming more and more important in the standardisation environment for all of these reasons, and in particular

the last – their ability to contribute to harmonization of multiple standardization activities. This is why AQUAS set an objective to also target framework-oriented standardisation activities for potential influence.

As a specific example, consider framework-oriented standardisation for the Internet of Things. In the area of IoT [22], ETSI and AIOTI are working jointly on standardization, with particular focus on communication. Here security plays a dominant role, and therefore the challenge in this context is to bring "safety" into some fundamental statements.

In ISO/IEC JTC1 SC41, *Internet of things and related technologies*, a joint committee of ISO and IEC in areas of common interest, Framework and Architectural standards (WG3), and Interoperability standards (WG4) are arising, besides WG5, Applications. Besides particular standards for sensor networks and wearables, the following evolving ones are of general interest here:

ISO/IEC CD 21823-1 [Under development]
Internet of things (IoT) -- Interoperability for internet of things systems -- Part 1: Framework

ISO/IEC NP 30147 [Under development]
Information technology -- Internet of things -- Methodology for trustworthiness of IoT system/service

ISO/IEC NP 30149 [Under development]
Internet of things (IoT) -- Trustworthiness framework

ISO/IEC JTC1 SC41 has a liaison with ISO/IEC JTC1 SC27, *IT- Security techniques*, which hopefully becomes effective in our sense. SC41 has a Study Group on Blockchain (Security!) (AHG 18) and a Study Group on Societal and human factors in IoT based services (AHG 17), besides other groups and liaisons.

Elsewhere in this document examples of framework-oriented standardisation activities in systems engineering (PMBOK, Arcadia) and the IEC 61508 standard (see e.g. Section 3.2.2) are presented.

2.5 Human Factors

Human factors have essential roles in safety, security and system performance. But similar to safety and security, they are generally covered by specific professional groups and specific standards. Thus, in line with AQUAS objectives it was important to analyse these specific standards for such essential relationships. This emerged in the medical use case.

Additionally, it was identified in IEC 61508-3 preparation for Edition 3 that human factors are insufficiently covered in context of functional safety. So a working group was started in IEC SC65A, WG 17, for IEC TS 62879, "Human factors – functional safety". AQUAS partner AIT initiated that human factors do not only impact safety in the conventional manner as described below, but that particularly "security" has a strong impact and is mainly endangered by human interference ("hackers"). This is definitely influenced by the AQUAS co-engineering concept, and now in an early stage of drafting.

In the medical domain, we identified several gaps and defects in current human factors/usability standards. This section outlines our observations and some possible remedies which could take the form of a commentary in the standards that designers can use to improve device safety and security, while being a possible proposal for updating the standards. A more detailed report for publication is currently in preparation. This section summarizes its major contributions.

Although we analysed many of the standards outlined in Section 2.2.2, our observations relate specifically to the following standards:

- EN 62366-1:2015: "Medical devices - Application of usability engineering to medical devices"
- IEC/TR 62366-2:2016: "Guidance on the Application of Usability Engineering to Medical Devices"

- EN 60601-1-8: 2007: "Collateral standard: General requirements, tests and guidance for alarm systems in medical electrical equipment and medical electrical systems"

Conceptual gaps in the standards that can lead to tunnel vision

We have observed some conceptual gaps in the medical standards. These are areas we highlight where the current wording of the discussion in the standards can lead to tunnel vision by readers: while highlighting some problems and solutions, it may cause standard users to ignore others. Two of the major conceptual gaps we aim to address in our proposals for changes to the medical usability standards are described below in Sections 2.5.1 and 2.5.2: (1) effects of usability on medical device security, and (2) causes of use errors beyond shortcomings in user interface design.

2.5.1 Lack of Consideration of Effects of Human Factors on Security

Usability directly affects device safety, performance and security, and although the former two are somewhat discussed in the standards, the relationship between usability and security is often overlooked. This is of concern. In particular, trade-offs between safety, security, performance are in fact decided by the effects on human behaviour of various design decisions. If the task of taking into account human nature is left to specialists in human factors as a stand-alone task, these specialists may be able to improve "usability" in a narrow sense (e.g. fonts, colours, size of buttons) but these most important trade-offs would be decided almost inadvertently through design decisions (about hardware, algorithms, configuration details, procedures of use) taken without the necessary joint consideration of all effects of system design.

The relationship between usability and safety is especially important; “the majority of medical device incident reports can primarily be attributed to use error” [27]. Emphasizing this role adds to the significance of usability, which some designers may consider a minor issue. For example, displaying dose limits on a user interface display not only “can reduce the burden on the users’ memory and increase their confidence when programming the pump”, but also help prevent a harmful dose [IEC 62366-2].

Another significant relationship mentioned in the medical standards is that between usability and performance. As an example of this relationship, IEC 62366-2 discusses how high task performance might increase the safety of a device as it prevents delay of urgent therapy, but that it might also introduce new hazards if critical confirmation steps are not incorporated. On the other hand, slow task performance could “lead a well-meaning user to pass over steps in a procedure to increase speed of the procedure. This can result in a higher probability of use error linked to a potentially unacceptable risk”.

Besides the strong relationships between usability, safety and performance, new advances in medical devices have introduced another important relationship, which is often overlooked: the relationship between usability and security. Recent discussions in the AQUAS project between security and human factors experts highlighted such trade-offs. For example, user satisfaction increases when users are offered a wireless medical device compared to one with several connective wires that make it harder to use and move; however, a wireless design also introduces a range of new security vulnerabilities. As another example, security of use of a device may be enhanced with use of user authentication that prevents malicious/unintentional use; however, this may also prove to be a nuisance to some users, especially if repetitively required. Most importantly, in case of an emergency, authentication may inhibit a clinician’s ability to respond in a timely manner, thus posing a safety hazard. This exemplifies our above-mentioned concern that usability-related decisions are really about crucial tradeoffs between system properties, and thus require full combined analysis of all their effects.

This latter example depicts how all four factors: safety, security, performance, and usability may interact and influence manufacturers' design decisions (with regards to the level of authentication they may require as part of their device design). *We suggest that, similar to other relationships, discussion and examples in the standards of the human factor aspects of security may help designers consider and prepare for possible trade-offs that may arise.*

Also important to note is that such considerations of the effect of usability and relevance of human factors on other important qualities are best explored in the early stages of development and revisited later in the project, rather than only retrospectively considered when the device is in its later stages of development or, worse, in use. Early inclusion of such analyses is likely to guide important design decisions at the start when changes are less costly.

2.5.2 Incomplete Consideration of Causes of Use Errors: Beyond Shortcomings in UI Design

While the standards we examined document advances in the understanding of the effects of usability on safety and performance, and thus the responsibility of designers to address usability concerns, they often take too narrow a view of the causes of use errors. The standards repeatedly focus on the importance of user interfaces in a somewhat narrow sense: readability of displays, likelihood of confusion between buttons, etc. This is the area of causes of slips, typically (accidental errors in the execution of well-planned actions), but higher-level mistakes (misunderstandings of a situation and of what must be done) and intentional violations of procedures are also important hazards, affected by design and thus to be addressed in design and validation.

An important, recurring term in the usability standards is *use error*, defined as "user action or lack of user action while using the medical device that leads to a different result than that intended by the manufacturer or expected by the user" [EN 62366-1]. As explained in the standards, the term *use error* "was chosen over the more commonly used terms" *user error* or *human error* because not all such errors are "the result of oversight or carelessness by the user" [EN 62366-1]. In fact, the standards suggest that it is inappropriate to start by blaming the users: "although human beings are imperfect, it is inappropriate to blame the user when problems occur during summative evaluation. The key in any analysis of use errors, close calls or use difficulties is to intensely search for a design-based root cause before attributing the use error to the user." [IEC 62366-2]

We agree that renaming user errors to use errors is certainly a positive change as it removes judgment from the user. However, current explanations now seem to shift much of this blame onto user interface (UI) designers alone, so that other aspects of design are comparatively under-emphasised. This idea is repeated frequently throughout the standards: "much more commonly, use errors are the direct result of poor user interface design" [EN 62366-1], "user interface design shortcomings can lead to use errors" [IEC 62366-2], and "the application of usability engineering is a principle means to reduce medical device unacceptable risk and improve patient care by reducing the potential for harmful use error through enlightened user interface design" [IEC 62366-2].

It is certainly true that non-intuitive displays, hard-to-learn controls, confusing menus, or ambiguous alarm signal messages - all examples of user interface design flaws - may lead to use errors. However, abundant experience, research literature and medical device incident reports reveal that these are not the only causes. Thorough identification of use-related hazards must consider: (1) the users, (2) the use environment, (3) the device design (including the user interface) and (4) the complex interactions between them. The diagram below lists examples of factors in each of these categories, and depicts how user interface is but a single player in a web of potential causes. *Not all these causes are discussed in the standards, and too much of the focus is on user interface design.* Quite often the environmental causes of use errors are in procedures, user workloads, responsibilities, etc., which while not under

direct control of the designers are affected by documentation and labelling recommendations that may need to be informed by the supplier.

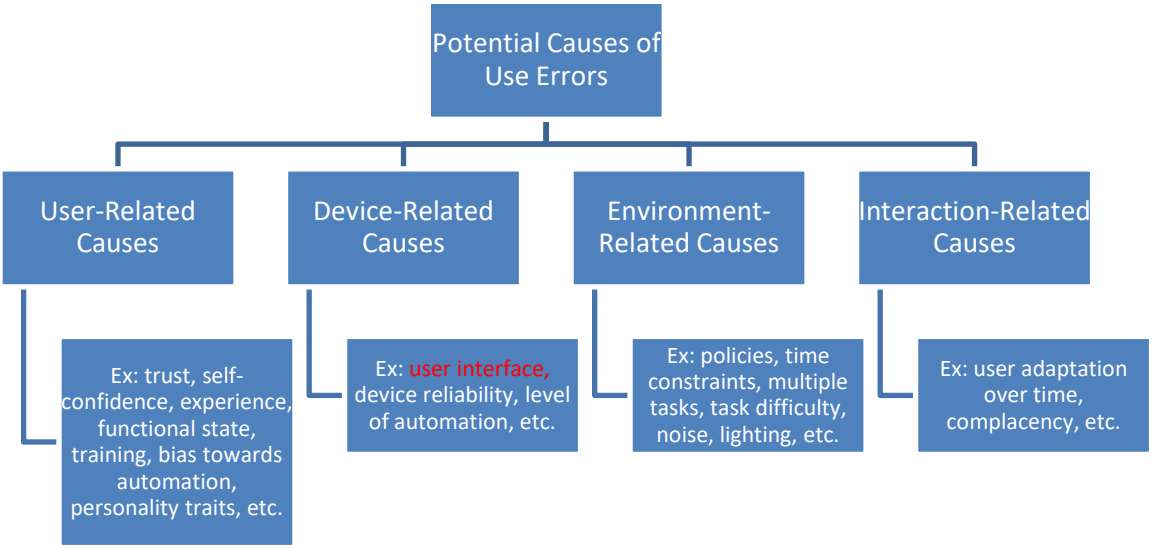

Figure 4: Potential causes of use errors beyond shortcomings in user interface design

As an example of how various factors in Figure 4 may interact to trigger a use error, consider a scenario where a physiological patient parameter has reached a dangerously low level and thus warrants immediate user action. To start with, user action is likely influenced by whether the device algorithm is designed to detect this danger with high enough probability (i.e., tool reliability). In case the device does detect this danger, the alarm signal emitted then needs to be visually and/or aurally communicated effectively to the user (i.e., user interface). However, in order to prevent a hazardous situation, it does not only matter whether the alarm is clear and audible, but also whether in practice it will lead to correct user action (with high enough probability). This may depend on environmental factors such as whether the user is busy dealing with another, simultaneous task (i.e., multiple tasks). It also depends on user factors such as the user's *mental model*: their "conceptual model of how the [device] works and is structured" [EN 60601-1-10]. In turn, mental models are based on users' knowledge and thus depend not only on their training but also on their experience of interaction and learning curve when using the device (i.e., user adaptation).

Just as the standards have moved away from placing the blame entirely on the user, we argue it is inappropriate to shift this blame onto the user interface designers. While in the standards there is a need to focus on the role of sound user interface design to ensure that designers take certain precautions, designers who focus solely on the role of user interface are likely to overlook other causes and thus fail to adopt mitigations in their designs to address these causes. We use Table 1 to help illustrate the danger of such tunnel-vision. The first two columns are taken directly from the standards [IEC 62366-2] and describe given use errors and user interface design shortcomings that may cause them. We add the third column to illustrate other plausible non-user interface causes of the same use errors, which would require different mitigations.

Use Error	User Interface Design Shortcomings	Other Possible Causes Not Related to User Interface Design
Users fail to detect a dangerous increase in heart rate because alarm limit is set too high and users do not look at medical device display because they are over-reliant on the alarm system	User-adjusted high and low alarm limits on a heart-rate monitor are not continuously displayed	User chose inappropriate alarm limits either due to inexperience or in an effort to reduce the device alarm rate which they find distracting *(Device-related cause)*
User ignored a warning label telling the user to disconnect the patient tube before turning the medical device off	The medical device did not require the user to confirm patient disconnection before powering-off	User at the end of a long medical procedure is fatigued and overlooks the importance of this step. Or hospital protocols, which the user is accustomed to, dictate that all equipment must be turned off before disconnecting from the patient. *(User-related cause /* *Environment-related cause)*
User disregarded a warning symbol and allowed a portable medical device to run out of battery power	The warning symbol was not sufficiently attention-getting	Lack of reaction to an alarm could also be caused by other factors such as the "cry wolf" effect. Paradoxically, designers can make devices more sensitive only to find that user decisions become less sensitive. In other words, it may not be that the user did not see/hear the warning but that their experience with the device's frequent alarms has led them to ignore it. *(Interaction-related cause)*

Table 1: Various Causes of Use Errors

Source: IEC 62366-2

As can be seen from the third column, our concerns are not denying the role of effective user interface design, but emphasizing that some use errors can be the result of other user, environmental, or interaction-based issues. Although some of these other causes can perhaps be remedied using the same design mitigations that address interface-design shortcomings, importantly, some of them will require different strategies; thus, highlighting the importance of taking a holistic approach to analysis of the causes. For example, in the third row in Table 1, making the warning symbol more attention-getting not only does not address the "cry wolf" effect discussed in the third column, but may even exacerbate it.

This important category – errors in response to alarms – of use errors, is a good illustration of how the recommendations in the standards may be simplistic, rightly highlighting the risk from very basic design flaws but not alerting the designers to subtler causes of the same errors, that also require care in design. The complex interactions of multiple factors that can lead to use error of this kind are exemplified in the publication by Alberdi et al. [28], which documents multiple causal chains that may

lead to these use errors. By heeding these more extensive explanations of use errors, the standards could avoid a narrow focus on user interface alone and protect against tunnel vision in design.

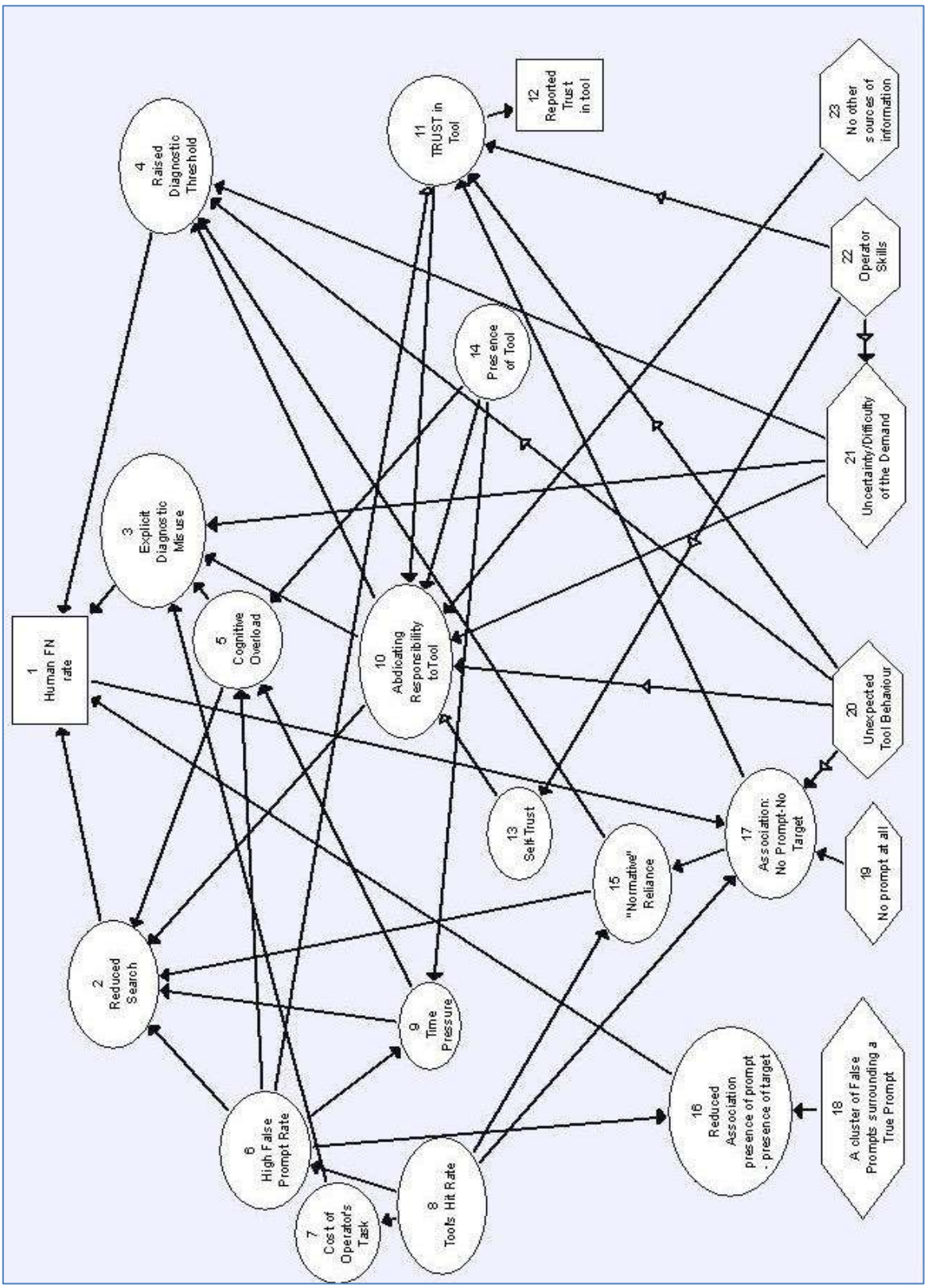

Figure 5: Analysis of possible causes of non-responses to alarms

Figure 5 shows an analysis of possible causes, *not linked to user interfaces*, of non-response to alarms, (from [28]). The graph is meant to assist designers in identifying the causal chains leading to undesired effects so that they can interrupt the chain with appropriate mitigations. This example highlights how a narrow view of use error causes may cause designers to choose inadequate or even counterproductive mitigations.

We believe this holistic approach to potential causes of use errors is becoming increasingly important. Medical devices are now used with increased frequency in busy environments, with new distractions. Also, patient care is evolving, sometimes moving to new environments such as private homes, or sometimes allowing use by less skilled or even unskilled users and patients; thus, wider considerations that incorporate such factors are becoming more important than ever.

IMPROVED CLARITY AND CONSISTENCY OF DEFINITIONS

The examples of conceptual gaps discussed above represent high-level observations of the medical standards; equally important are lower-level observations regarding the definitions of specialized terms used in the standards. Precise and consistent definitions ensure a common understanding of the standards by all the different people involved: managers, engineers, researchers, clinicians, business professionals, etc. Holistic and thorough definitions can also help highlight causes or consequences of hazards that may otherwise be overlooked. Our analysis of the medical standards reveals that crucial concepts such as *alarm system, alarm condition* and *alarm signal* are defined in ambiguous/inconsistent ways that may lead users of the standards to overlook certain hazards [EN 60601-1-8:2006].

LATEST METHODOLOGIES AND TECHNIQUES FOR EFFECTIVE USABILITY

An important part of the standards is the summary of the latest research on accepted techniques and methodologies that may be used to achieve effective usability. We note a thorough discussion of a variety of techniques. However, we also note an over-simplification of important concepts that should be considered, which we address in our proposals, supported with more recent references.

INCREASED, ENRICHING EXAMPLES TO ENHANCE UNDERSTANDING OF THE STANDARDS

We also highlight several areas in the standards where we propose that discussions would benefit from effective examples. We rely on research literature and manufacturer-related material for relevant and practical examples. By proposing use of these examples, we aim to clarify the points under discussion, increase their significance, and deepen readers' understanding – for example, by revealing less obvious issues such as: (1) how a desirable design goal can lead to a use error when considered in isolation, (2) how factors can interact and trigger one another to lead to a use error, and (3) how users can be affected by a device without actually conforming to its advice.

2.5.3 Human Factors in Automotive Standards

Within the automotive standards, there is work in human factors in the context of the SOTIF standardisation work described previously.

Driver interaction with a highly automated vehicle is clearly very sensitive to human factors, and therefore has been brought into the scope of the SOTIF. The approach is based upon the *Human Factors Analysis and Classification System* (HFACS) [31].

An example of the approach is presented in Table 2.

Performance limitation scenario	1) Stakeholders	2) Misuse causes		3) interactions between driver and system/vehicle	Misuse scenario 4) consider condition of environment
		process	Guide words		
"While operating autonomously on a highway, the vehicle cannot estimate the location of the lane boundary due to a performance limitation. The vehicle starts to leave the lane and the driver is notified to take control."	Driver …	Recognition	1. Do not understand	Operation (Usage)	…
				Vehicle behaviour	…
				Warning/information	"Driver does not take over control of the vehicle and vehicle departs lane because driver does not know meaning of the warning"
			2. False recognition	Operation (Usage)	…
				Vehicle behaviour	…
				Warning/information	…
		Judgment	3. Judgment error/ misjudgement	…	…
		Action	4. Slip/Mistake	…	…
			5. Intentional "driver vacated seat"	…	…
			6. Unable "Driver not paying attention Driver asleep"	…	…
	…	…		…	…

Table 2: Examples of misuse scenarios in the SOTIF

(*Source: SOTIF*)

Although AQUAS does not have an automotive use case, this demonstrates the general interest of this dimension across mission-critical domains.

2.5.4 Human factors in Space Standards

In the ECSS family of standards, ECSS-E-ST-10-11C – *Human factors engineering* (31 July 2008) forms part of the System engineering branch of the Engineering area. As such it is intended "… to assist in the consistent application of human factors engineering to space products by specifying normative provisions for methods, data and models to the problem of ensuring crew safety, well-being, best performance, and problem avoidance in space system and payload operations".

Note the link to both safety and performance in the descriptive text of the standard – more evidence of the relevance of human factors to the AQUAS objectives.

3 Co-Engineering Gap Analysis of Current Standards

3.1 Current co-engineering issues in standards development

In multiple domains, which already have safety as an established property, security is becoming a new issue. Due to increased interconnectivity and usage of Commercial Off The Shelf (COTS) components in safety-critical systems there is an growing threat to the cybersecurity of safety-critical systems. Usage of COTS components leads to common vulnerabilities in different systems and more people with the skills to identify and exploit vulnerabilities. In addition, cyberattacks are increasingly used as a new method for covered attacks by state-connected or terrorist groups. Cooperation between such groups and semi-commercial hackers who search for and sell zero-day exploits and malware kits leads to threat actors with increased expertise and resources. This, combined with the increased attack potential, leads to a rising threat landscape for safety-critical systems.

The rise of autonomous systems in multiple domains has led to the need to include **performance** as part of the mix, since it has been observed that inadequate performance (e.g. of advanced sensors) can have an effect on the other dimensions, particularly of safety. In addition, it has been recently observed that measures taken to enhance performance have had serious consequences on the other dimensions. For example, the Meltdown and Spectre bugs arising from attempts to enhance the performance of Intel processors have seriously compromised their security characteristics.

The introduction of performance into co-engineering is extremely recent, and few standards are treating it to date (a notable exception is the automotive SOTIF). Thus, we must rely for now on experience with cybersecurity and safety dimensions in the standards developing organizations in order to see how they are currently addressing the topic of co-engineering.

In most domains there has been a long discussion about how to address this new challenge. The discussion has focused mainly on how to address the issue of security in safety critical domains in the standards. Discussed approaches include:

1. Use established security standards for security engineering in safety-critical domains
2. Extend established safety standards with security engineering in safety-critical domains
3. Develop own security standards for security engineering in safety-critical domains

Most domains decided on approach 3 while also integrating links from safety to security in their safety standard, which is very important, since cooperation between both areas is crucial for success. The following is a brief summary of the reasons why it was necessary to develop specific security standards for safety-critical domains. Since the development of security standards is still ongoing, this list does not include a solution to the approaches, but presents challenges when trying to apply “standard” IT security to safety critical systems or trying to use “standard” safety approaches for cybersecurity. As noted earlier, the inclusion of the performance dimension adds another element of complexity that has yet to be addressed in the standards developing organisations, and could become therefore an issue of interest to contribute for AQUAS.

3.1.1 Risk assessment and management

At first glance, risk management in cybersecurity and safety is similar. Based on an initial risk assessment measures for risk reduction or mitigation are implemented. During operation incidents are monitored and, if evidence shows that the risk management is insufficient, additional efforts are required. But while safety assumes a *random* distribution of failures over time and components, security needs to consider an *intelligent* attacker. Attacks are timed to maximize the impact and if an incident is detected the system is often compromised in multiple additional ways. In addition, safety relies for risk assessment on existing information about past systems to enumerate the risk and

determine an acceptable level. The combination of our limited experience of the new forms of advanced persistent threats and the growing interconnection of critical components reduces the usefulness of past experiences. Hidden interdependencies like reliance on common infrastructure (time or position server) or components (same variant of SW or encryption library) leads to single attacks which brings down many different systems. In previous work, AQUAS researchers have observed that it is inherently infeasible to associate some kind of probability with security risk, whereby it is not infeasible for safety. [15][16]

Current risk-management techniques from safety are blind to intelligent attackers and have no usable existing data. Security analysis misses the cyber-physical dimension, e.g. the impact on the real world and consideration of system environment. **Performance risk analysis** to date has been nearly entirely associated with nominal functional operation (often linked to commercial issues such as minimal acceptable performance by customers), with little or no relationship established to safety and security dimensions.

Therefore, the establishment of a unified risk assessment / management regime in co-engineering for all three dimensions remains a significant challenge.

3.1.2 Incident reporting and sharing

A common best practice in all dimensions is the recording of incidents – for example, for compiling "lessons learned". However, there are pressures of different kinds that inhibit incident sharing.

To date, the recording of **performance incidents** has been generally kept proprietary by the manufacturers when the incidents did not have a clear impact on either safety or security. This is generally in order to protect brand reputation – that is, to fix defects without adverse publicity.

In the safety dimension, while sharing of safety incidents increases the level of achievable safety for all, sharing of security risks can, in the worst case, increase the risk level for all. Especially for safety-critical legacy devices which are often not continuously connected closing of vulnerabilities is a time-consuming process. Publishing vulnerabilities leads to a "window of vulnerability" which can exist for quite some time. In addition, processes and responsibilities for sharing of vulnerabilities are currently in definition and not established in industry. Therefore, processes and responsibilities for cybersecurity incident sharing are not defined. Especially with shared components domain specific sharing can lead to risks to other domains. Existing security sharing policies cannot be copied to the safety domain. Existing performance related incident recording policies are generally unique to the manufacturing organization and often kept as proprietary as possible in order to protect IP and avoid negative commercial consequences. Now that performance is being related to safety and security impacts, this will have to change, while acknowledging the commercial pressures on manufacturers.

3.1.3 Safety / Security / Performance related development processes

Nearly every standard regulating mission-critical systems development specifies a development **process** (the left-hand side of the V-model) together with a set of recommended best practices according to the dimension being treated by the standard. For example, a standard focusing on performance might focus on algorithm quality or recommended performance benchmarks. A safety engineering standard might focus on a set of recommended "safety patterns" that are well trusted in the community. But they can come into conflict.

For example, the need to consider cybersecurity threats reduces the usability of established safety engineering patterns. Redundancy and diversity are well-established safety mechanisms. Software-based systems rely mainly on diversity, e.g. having two different versions of software or even systems for the same task. Considering cybersecurity this increases the potential attack surface and requires auditing and ensuring security of two supply chains, including checking all used COTS elements and vetting suppliers and involved developers. Established safety design-patterns lead to an increase in

cybersecurity risks and there are no easy solutions, either from an architecture or from a process side. New architectures and design concepts need to consider safety and cybersecurity. An example is the potential usage of cryptography for security (confidentiality) *and* safety (error detection).

On the other hand, diversity may be used to detect certain types of anomalies since it is more unlikely that both (or all, if more channels are available) channels are attacked by the same means. If multiple diversity is available, this would allow continued operation while the infected or disrupted channel is cleaned. Pure homogenous redundancy is prone to react in a malicious way to the attack at the same time.

Finally, redundancy is also a commonly used technique in order to ensure adequate **performance**, including availability (in terms of nominal functionality). But its interpretation in performance-related development is clearly different from that of safety and security. AQUAS researchers (especially CITY) have been particularly active in research in different types of redundancy and their effect on safety and security.

In summary, each of the three dimensions treats “best practices” in development in different ways, making it difficult to harmonize the standardisation of development according to each of the three dimensions. This remains a challenge for co-engineering standards.

3.1.4 Safety / Security / Performance related testing, analysis and V&V processes

Whereas the development process addresses the left-hand side of the classic “V model”, the testing, analysis, and verification / validation processes address the right-hand side. Standards such as the DO-178B (“Software Considerations in Airborne Systems and Equipment Certification”) emphasize the importance of software verification. Verification is defined as a technical assessment of the results of both the software development processes and the software verification process. Sec. 6.0 of the DO-178C states that “verification is not simply testing. Testing, in general, cannot show the absence of errors.” The standard consequently uses the term "verify" instead of "test" when the software verification process objectives being discussed are typically a combination of reviews, analyses and test. The purpose of the software verification process is to detect and report errors that may have been introduced during the software development processes. Removal of the errors is an activity of the software development processes. The general objectives of the software verification process are to verify that the requirements of the system level, the architecture level, the source code level and the executable object code level are satisfied, and that the means used to satisfy these objectives are technically correct and complete. At the code level, the objective is to detect and report errors that may have been introduced during the software coding process. The non-functional safety properties are explicitly listed as a part of the accuracy and consistency verification objective at the code level, including stack usage, worst-case execution timing and absence of runtime errors.

Performance testing is well aligned with classic testing and V&V, with a long history of structuring the process and automating the execution to reduce human effort and increase test coverage. While performance testing typically focuses on maximizing throughput or minimizing latency, safety-oriented performance analyses focus on ensuring compliance with predefined performance limits, in particular resource constraints such as stack size or real-time deadlines. A technique which can provide guarantees that such limits are met is sound static code analysis at the executable object level. Meeting resource constraints is a typical requirement of safety standards.

Safety testing increases complexity further, whereby different phases in the lifecycle rely on defined test approaches to ensure correct implementation of safety measures and sufficient risk reduction.

The situation changes with security. Security testing follows different strategies and the highest level is human testing (penetration testing) which is only partially structured and difficult to automate. Identifying and using overlaps between security, performance, and safety testing has the potential to reduce effort. One such overlap exists at the code level. A common safety requirement is that runtime

errors due to undefined or unspecified behaviour of the programming language must be prevented since they can cause erratic and erroneous behaviour and, hence, may provoke safety hazards. Examples are division by zero, buffer overflows, or data races. At the same time these programming defects also represent the most important security vulnerabilities at the code level which enable data leaks, code injection and denial-of-service attacks. Sound static analysis at the code level can detect all such runtime errors and constitutes a common activity in the safety and security life cycle.

AQUAS results in this area could become a valid input to standardisation efforts in the various domains.

3.2 Gaps in selected current standards

In the following sections, representative standards are selected for analysis as examples of issues in co-engineering gap analysis.

3.2.1 ECSS standards

The ECSS standards family is chosen here as a representative of the problem of integrating separate, pre-existing standards for safety, security, and – in *some* cases – performance.

In Space projects, currently the standards for Safety, Security, and Performance are handled **separately**. In the ECSS, dependability and safety are the only aspects that are handled together which define the criticality of the software. For defining this, no Security or Performance aspects are considered. Safety and dependability standards are defined separately in the ECSS-Q-ST-40C and ECSS-Q-ST-30C documents. It would be beneficial as a standard evolution that the documents ECSS-Q-ST-80C and ECSS-E-ST-40C (software quality and software engineering respectively) pointed to a concrete set of security and performance standards so they can be used for the selection of the software criticality and tailoring.

For Space software projects, there are currently **two clear scenarios**: one scenario for which there are performance standards that can be used, and another scenario for which it would not be realistic to implement performance standards. Therefore, trying to introduce a specific performance standard across *all* Space projects would not be advisable. Regarding security, we believe standards are in the same situation as described for performance.

The ECSS software standards do point to dependability and safety standards. As we study co-engineering in the context of AQUAS, we believe it would be advantageous to have a link to a concrete set of security and performance standards in order to have all these aspects considered when defining the software criticality. Currently the standards and the development efforts are tailored according to the criticality of the software by means of an applicability matrix. Therefore, this tailoring would need to consider when safety and security aspects are applicable.

There has been a recent evolution of the current ECSS standards, where the definition of the criticality categories and how to perform dependability and safety engineering in a project have been revised and partially harmonized. In July 2016, the ESA Board for Software Standardisation and Control released a first version of a "Secure Software Engineering Standard", ESSB-ST-E-009 and a companion "Glossary of Secure Software Engineering Terms", ESSB-ST-E-009 Issue 1. The purpose of these documents is to enhance the E-40 software development process in the area of secure software development.

However, the ECSS standards still do not consider the safety, security or performance aspects *together* for the classification of projects according to criticality categories. Within AQUAS, we propose to aim our efforts in standards evolution towards this goal.

3.2.2 Project Management Body of Knowledge (PMBOK)

As a representative example of a framework-oriented standard, the Project Management Body of Knowledge (PMBOK) [29][30] has been selected for gap analysis. The PMBOK is an exceptional and widely recognized "summa" of project management competences, used worldwide to train managers (see *PMI Project Management Institute*). However, there is room for improvement in the area of co-engineering and its sister concepts of multi-discipline, concurrent / simultaneous engineering.

In the current version of the PMBOK, the term "co-engineering" is never found. The term "concurrent engineering" is also never found, nor is "simultaneous engineering". The term "discipline" is used only very generically. The term "speciality" is used occasionally to indicate a specific technical area (such as mechanical or electronic), and discussed in the project organization. But the concept is not further elaborated.

Parallel work is only marginally reported in *Fast Tracking schedule compression* as an approach where "…activities or phases normally done in sequence are performed in parallel for at least a portion of their duration." Clearly, this interpretation of parallel activities does not capture the full semantics of co-engineering.

As a framework-oriented standard, it is reasonable and correct that the PMBOK not focus merely on specific concerns like safety, performance, and security, but on the general concepts of multi-concern / discipline project management – providing exactly the conceptual framework within which co-engineering in the AQUAS sense can thrive. In this sense, we propose the following recommendations for the PMBOK:

Future project managers need to be trained more on organizing and controlling a project through its many disciplines / specialties (e.g. mechanical, optical, electrical, safety, security, performance, energy, waste, etc.). This should become more evident from the WBS (work breakdown structure) and OBS (organization breakdown structure).

The many disciplines should proceed in parallel in a harmonized way, conflicts shall be identified and resolved, trade-off established. The PMBOK may call this "co-engineering" (or another suitably general term), where a pool of experts from the many different disciplines / specialties cooperate on a continuous basis to take major design decisions. Likely, infrastructure facilities may be required to improve co-engineering (e.g. co-presence, co-editing. etc.).

Codifying such a general conceptual basis for co-engineering in the PMBOK could make a real contribution to facilitating its instantiation in the various domain-specific standards.

3.2.3 Arcadia – Engineering Methodology for System, Software and Hardware Architectural Design

The **ARC**hitecture **A**nalysis & **D**esign **I**ntegrated **A**pproach (ARCADIA) emerged initially within Thales in 2007 and is in the process of becoming an open standard. It was created in response to the need for a common approach, placing collaboration at the heart of engineering and so reducing the communication barriers between engineering teams, within and across all structured engineering activities of the PLC (see Figure 6). This global method can be customised to specific domains and enables greater traceability assurance, engineering efficiency and mastery of increasing system complexity. The Arcadia methodology is divided into five engineering parts: Operational Analysis Model; System Functional and Non-Functional Need Model; Logical Architecture Model; Physical Architecture Model; Product Breakdown. These are all fed by information related to scenarios, data models, data flow, functional chains/operational processes, modes and states.

Discussions for a collaborative analysis of Arcadia by AQUAS partners was commenced early in the project by TRT. At this point Thales Global Services who provide leadership for the Arcadia methodology were contacted about AQUAS plans to provide a change request during the project. Inside the Consortium, interest has gathered from partners including Tecnalia, the CEA, All4Tec and Magillem. The analysis is planned to begin in November 2018 with the main goal to provide recommended updates to the Arcadia Methodology with respect to SSP co-engineering. Actions will include a review of how SSP interactions are managed in this methodology (within and across PLC stages), how it relates to the AQUAS methodology, derived recommendations beneficial to Arcadia, type of tooling to support the co-engineering and the submitted change request.

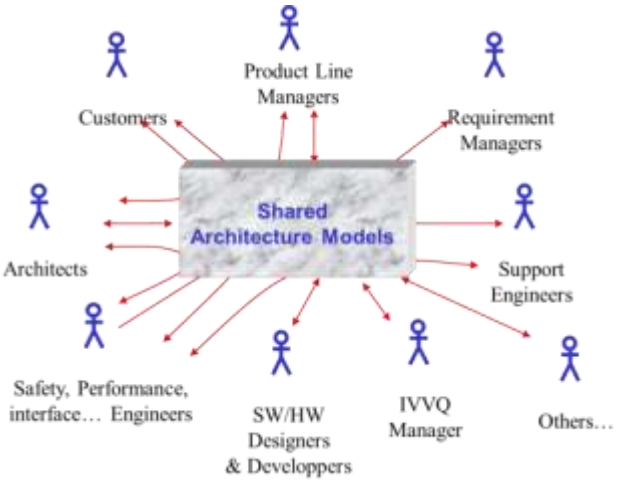

Figure 6: Enhanced connection of stakeholders

3.3 Current approaches being pursued by standards developing organisations

In the following, some examples of co-engineering approaches being pursued by SDOs are given (not necessarily in the specific domains of the AQUAS use cases). They are primarily concerned only with safety and security.

3.3.1 IEC TC 45, SC45A - Nuclear Power Plants

The series of nuclear power plants safety and associated cybersecurity standards are a good example of a very good separation of concerns in the documents, and, on the other hand, integration of co-engineering aspects by a coordinated approach. They chose a three-step approach:

- **IEC 61513** (*Nuclear power plants. Instrumentation and control important to safety. General requirements for systems*): focusing on safety (Nuclear Safety domain standard interpreting IEC 61508)
- **IEC 62589** *(Nuclear power plants - Instrumentation and control systems - Requirements for coordinating safety and cybersecurity):* setting up “fundamental principles” to protect the safety objectives despite cybersecurity threats, avoiding adverse impact of cybersecurity counter measures. This represents the “safety first” viewpoint, view on security measures from the safety point of view.

  Some of the “fundamental principles” are (examples, excerpt):

  5.2 Fundamental Principles

  - **Cybersecurity** shall **not interfere** with the **safety objectives** of the plant and shall **protect their realisation**. It shall not compromise the efficiency of the diversity and defence-in depth features…
  - **Cybersecurity requirements** impacting the overall I&C architecture shall be **addressed**…

- Implementation of **cybersecurity features** shall **not adversely impact** the required performance (including response time), effectiveness, reliability or operation **of functions** important to **safety**.
- The **failure modes and consequences** of cybersecurity features on the functions important to safety shall be **analysed** and **considered**.
- Any **architectural property or characteristics** initially designed for safety reason (e.g., independence between systems), and later considered as a potential cybersecurity counter-measure … should be re-examined on purpose … to confirm its cybersecurity added-value

- **New work item NP 45A/1091/ IEC 62XXX:** "*Nuclear power plants - Instrumentation and control systems - Security controls*", specifically focusing on the selection and application of computer security controls from the included security controls catalogue" (based on IEC 62645, top level document for cyber security, and IEC 61513
  - To ensure consistent understanding of the process of the selection and application of cyber security controls;
  - To ensure consistent understanding of what security controls are recommended and optional for the security baseline and the security degrees S1, S2 and S3 (Catalogue)
  - to describe a method for crediting/inheriting existing security controls and safety provision for I&C systems important for safety
  - To describe a method for applying compensatory security controls in case recommended security controls cannot be implemented
  - To describe a method for handling of the legacy topic.

3.3.2 IEC TC 44, Safety of Machinery – Electro-technical aspects

With respect to the general domain of machinery safety, the following activities are observed:

- **IEC 60261** *("Safety of machinery: Functional safety of electrical, electronic and programmable electronic control systems"),* is the domain standard interpreting IEC 61508 (plus complementary ISO 13849-1 for other safety aspects than E/E), for security of IACS (Industrial Automation and Control Systems) is IEC 62443 the basic standard.

- **Cybersecurity:** Originally, the idea was to separate safety of machinery (responsibility of the manufacturer of the machine) and cybersecurity responsibility (on the shoulders of the OEM/integrator). This early concept was rejected by the IEC ACOS (Advisory Committee on Safety) as well as by ISA 99 (International Society for Automation, before known as "Instrument Society of America") with the argument, that a separation of requirements cannot be at this low level. (Informal comment: "Just a firewall is not sufficient!").

- **New work item has been** started: *"Security aspects related to functional safety of safety-related control systems"*: now a much better approach "to consider the security aspects in context of safety of machinery", a similar approach as the second level standard of IEC TC45 SC45A, Nuclear Plants)

  The following aspects will be considered:

  - what is the relationship between safety and security?
  - vulnerabilities can be the result of systematic faults which can lead to a hazardous situation of the machine;
  - vulnerabilities may impact the integrity and availability of the safety-related control system to properly perform its function(s);
  - reasonably foreseeable misuse (see ISO 12100), e.g. typical use case definition and application of a corresponding threat model.

4 AQUAS influencing standards

4.1 Overall approaches to influencing standards

It is well known that standards tend to have a long evolutionary lifecycle – not merely because of "inertia", but also because an essential characteristic of a standard is the need for a sufficient period of guaranteed stability to fulfil its role as a point of reference and compliance within its community of users. The AQUAS consortium is well aware that it will not always be possible to synchronize its standards-influencing efforts with the windows of opportunity that will arise during the evolutionary cycles of the standards developing organisations.

Nevertheless, activities can be engaged that are useful even in the absence of perfect synchronisation with the standards renewal cycles. The following approaches are under consideration:

- **Reports and change request packages**. Even in the cases where the SDOs are not currently in a public consultation or updating cycle, a package can still be prepared containing reports and/or change requests that may be presented at the next possible updating cycle of a standard, also beyond the end of the project. This is particularly viable when a member of the AQUAS consortium is a member of the relevant standards committee and can take direct responsibility for presentation of the change requests at the opportune moment. In a sense, this may be considered a type of "leaving a legacy for posterity".
- **Presentations to standards committees and working groups**. Even outside of updating cycles, standards working groups are often active – for example, to collect experience reports and suggestions from users for future revisions of the standard (which may still be years away). It is common usage to present new ideas and technologies to these working groups to inform them of recent developments to take into consideration. AQUAS members can prepare targeted informative presentations for such working groups. Indeed, this has been formalized in the awareness-raising objective of AQUAS.
- **Guidelines**. It has become common practice to prepare guidelines for the usage and interpretation of standards in particular ways, especially when the standard is unlikely to be updated for several years. In this way, an informal type of *ad hoc* modification to the standard is achieved. For example, a guideline entitled "Co-engineering with Standard X" could provide a set of recommended best practices that augment the standard with facilities for achieving co-engineering (e.g. additional intermediate steps), but remain conformant with the text of the standard in its current form.

4.2 Current approaches to meeting specific standardisation objectives

The following sections summarize current status as of deliverable submission with regard to addressing the specific objectives of the project.

4.2.1 Objective 9: Change Requests

Contribute to the improvement of standards to address co-engineering, by submission of change requests to at least 1 standard for each of the AQUAS use case domains.

The use cases provide the most immediate potential opportunities to submit change requests, because AQUAS partners are adequately positioned in a number of cases. DO-278 is heavily applied in the **ATM use case**, and partner Thales will be in the position to make change requests toward the standard. In the **medical use case**, there are three potential standards improvements (IEEE 11073, EN62304, EN-60601-1-10). As discussed previously in this document, the **rail carriage mechanisms** use case involves the EN 5012X standards, where partner ClearSY is deeply involved, and so is Intecs. The **Industrial Drive**

use case involves IEC 61508, as discussed earlier, where partner AIT is involved (see also discussion in Section 4.2.3), and also ClearSy. Furthermore, there is a set of standards around IEC 62443. Several ECSS standards are relevant to **Space Multicore Architectures**, including E40, Q80, Q30, Q40, and all providers to this Use Case are able to contribute change requests to these respective standards.

As noted in the introduction to this section, the fact that AQUAS partners are well positioned to submit change requests when suitable AQUAS results are available does not guarantee that this can happen within the duration of the project, depending on the revision cycles of the specific standards. However, the strategies outlined above (e.g. packages containing analyses and change requests) can be employed regardless of revision cycles and ensure that AQUAS produces a set of change requests that may be submitted also beyond the lifetime of the project.

4.2.2 Objective 10: Promoting awareness

To promote awareness and bring results of AQUAS into at least two international standards in the functional safety and security area with respect to safety, security and performance co-engineering.

As noted previously, awareness-raising initiatives are valid alternatives even in the absence of perfect synchronisation with revision cycles, and AQUAS partners are currently involved in various initiatives of these kinds.

Partner AIT presented the AQUAS approach of interaction points in a meeting aimed at the development of ISO/SAE 21434 as a potential approach how to instantiate the communication channel and use them in applied processes. Communication channels from safety to security are currently already required in the FDIS version of ISO26262 Edition 2. The challenge is that the current definition of “communication channel” is based on exchanged information and there is no process specifying *how* such information could be exchanged. The interaction points [20][21] from AQUAS are a potential way to specify how a communication channel could be defined in a company-specific process.

In the medical devices use case, Partner RGB has provided contact information the leader of a medical cybersecurity group, and is a potential point of influence for AQUAS.

There are new developments with regard to the new generation of trains, e.g. the European Railway Traffic Management System. For example, in CENELEC TC9X, work is going on with respect to integration of security considerations in the existing standards landscape. DIN VDE V 0831-104 “Electric signalling systems for railways – Part 104: IT Security Guideline based on IEC 62443” was a “forerunner” in proposing to include the lower SL (Security level 1) of IEC 62443-1 in the safety standards. AQUAS partner AIT has made contacts with SC9X SGA16, which recommended to consider security in the CLC TC9X standards’ framework. The chairperson, together with AIT and others, organizes the so-called “Safety meets Security” workshops of “Hanser Tagungen”, where the co-engineering aspects are disseminated. Therefore, there may be some impact in the longer term of evolution of AQUAS standardization work.

Finally, AQUAS partner INT is a member of INCOSE, and will promote awareness of INCOSE inputs in the AQUAS project and bring results of AQUAS to the INCOSE members or conferences.

4.2.3 Objective 11: Influencing framework-oriented standardisation groups

To influence actively in two international standardization groups focused on frameworks for the coordination of safety, security, and reliability of automation.

AQUAS partner AIT is involved in this area, where framework standards are being set up in IEC TC 65 on Smart Manufacturing and in IEC TC 65 SC65A in updating IEC 61508 towards Ed. 3. Preliminary work has started, and the official stability phase of Ed. 2, 2010, will be extended to have more time to prepare a solid Ed. 3 considering new paradigms (particularly in software) and on transversal topics like co-engineering or consideration of cybersecurity in safety standards. The TC65 AhG1, later known

as WG20, together with TR 63069, tried to fill the gap in “Framework for functional safety and security”. The co-engineering aspect (a subchapter) and the interaction between safety and security teams are described, but it was a compromise: contrary to the AQUAS approach, the interaction is not described in a fully symmetric manner.

To speed up the process for a modernized IEC 61508 as a system standard, WG 18 was created in SC65A for IEC 63187, “Functional safety - Framework for safety critical E/E/PE systems for defence industry applications”, which plans to overcome the IEC 61508 weaknesses versus large systems, systems engineering concepts, and concepts like the interaction between safety & security (this is already marked as a separate chapter).

In Smart Manufacturing, an Ad-hoc group AhG3 was started in IEC TC65, which delivered a report and starts as WG 23. Together with ISO TC 184 (Automation systems and Integration), JWG 21 was founded on Smart Manufacturing – Reference models. A Cybersecurity Task Force was founded for AhG3 to look at the safety – cybersecurity issues in context of smart manufacturing. Here again, AQUAS Partner AIT is trying to raise awareness and get consensus on the co-engineering and interaction point concepts – but it is a hard fight to achieve consensus in this direction.

Finally, within the framework-oriented standards on the Internet of Things (see also Section 2.4), Austrian partners are active mainly on national level, where influencing development towards common safety and security issues remains a challenge at this stage.

4.2.4 Objective 12: Promoting awareness in other standardisation groups

To promote awareness and bring results of AQUAS into at least two other engineering international standards, such as OMG, or FMI.

A major action is currently underway for the MARTE OMG standard, involving the preparation of Request For Information (RFI) for MARTE 2.0, envisioned as an extended version with capacities to manage complementary DSLs, plus various technologies to serve IoT domain applications and other kinds of specialised analysis for non-functional properties in conjunction with software and hardware design.

AQUAS partners have been involved in this initiative since the beginning, to create opportunities to bring AQUAS results in the MARTE standard.

Over the first year of life of AQUAS, we have progressed mostly in the collection and submission to OMG of the interests of partners and the definition of the AQUAS high-level requirements for MARTE 2.0. This activity has been coordinated internally by INT and TRT, with interest expressed by CEA, INT, MAGILLEM, MTTP, UNIVAQ, ITI, TRT and CITY and also coordinated externally with the MegaM@Rt2 and the AMASS ECSEL projects.

As a result, AQUAS partners now have a first version of collected requirements that will be used to submit the requested information for MARTE 2.0. The principal requirements gathered to date concern the expansion of its modelling and annotation capabilities for current evolution of real-time embedded systems (e.g. CPS, IoT, and Industry 4.0) and their necessary quality attributes (with emphasis on dependability, safety, and security attributes).

RFI submissions will help OMG to prepare the initial draft for the MARTE 2.0 Request for Proposal (RFP). AQUAS partners plans also to participate in an initial submission team that responds to the RFP. If possible, we may even contribute to the revised submission in the course of the project.

5 Conclusions and Further Activities

The submission of this deliverable is occurring at approximately the halfway point in the AQUAS project. The use cases are beginning to produce first results and elicit actionable requirements for standards evolution. As the previous discussions in this document have illustrated, ongoing contacts continue and new ones are being established, as well as new initiatives planned. Section 4.2 of this document has outlined the specific initiatives underway and indicated which partners are involved.

In addition, this section describes a further set of internal activities that are intended to keep the standards evolution activity focused and on track throughout the project.

Tracking progress toward objectives

In order to channel upcoming activities in the remainder of the project into the most effective paths, both the expert advisory board and the project reviewers have recommended to put into place mechanisms for tracking progress toward the achievement of the objectives of these activities. This is currently being implemented for the Standards Evolution Goal in the harmonized spreadsheets over the three goals.

As a specific example, a set of general challenges in the area of standards evolution was identified, and a preliminary set of potential progress indicators against these challenges was identified, as illustrated in Table 3.

Challenge	Progress Indicator
How to provide visibility of challenges and progress, addressing priorities and decisions (supported in AQUAS or later).	Number of presentations either in AQUAS related meetings (e.g. EAB) or public conferences
Industry may have reservations to adopt an approach which is not reflected in current standards.	Number of explicit contacts established with companies on the question of standards-based co-engineering
There are domains in which integrated approaches to safety and security are not fostered by the governing standards –or even implicitly discouraged.	Number of papers or public reports (including AQUAS deliverables) arguing integrated standards approaches

Table 3: Preliminary progress indicators against general SE challenges

Identification of key participants and skill sets

As part of the overall initiative to track progress, identification of key participants from the consortium (in terms of both person/months available for commitment to the activity and skills / contacts available for this activity) has begun, so that maximum involvement over the consortium can be achieved, and opportunities for exploiting consortium-internal resources are not missed. This is likewise being managed in the spreadsheets made available consortium-wide for this purpose.

Harmonization of terminology

Another important activity launched over all three AQUAS goals was the **harmonization of terminology**. This was also the result of a recommendation from the external advisory board: ensure that any proposals resulting from AQUAS work are consistent with the directions being taken by the standardisation groups, in order to avoid the ugly surprises that result from extreme mismatches both at the conceptual and at the terminological levels.

First efforts began in this direction shortly after Month 12 of the project, reacting to first external advisory board recommendations. Table 4 presents an extract from the (much larger) table currently being maintained, in an attempt to arrive at a proposed harmonisation of terminology.

As can be seen in the table, there are already problematic areas. For example, "risk" is defined and understood in many different ways in different standards, and it will be a challenge for AQUAS to arrive at an operational understanding of risk that is sufficiently robust to support its methodology and at the same time satisfy the interpretations of the target standards that AQUAS will later approach. Nevertheless, the AQUAS consortium recognizes the importance of the recommendation to analyse these terminological issues as an integral aspect of a viable initiative to influence standards.

Another problematic area is the terminology involving the **performance** parameter. Hardly any standards to date even define performance as part of any formal glossary. Only with the advent of the most recent standards governing autonomous applications (e.g. aerial drones and self-driving road vehicles) are we beginning to see the performance parameter inserted into controlled vocabularies – and even then, there is only a top-level, relatively generic definition. Clearly, this parameter in particular is only beginning to be treated systematically in standards, and the AQUAS project could be instrumental in formalizing its place in co-engineering.

Term	Definition	Type	Source
Safety	State where an acceptable level of risk is not exceeded. This may apply to the system or its environment (particularly to people).	Safety	ECSS / CRR
Risk	The level of impact on organizational operations (including mission, functions, image, or reputation), organizational assets, or individuals resulting from the operation of an information system given the potential impact of a threat and the likelihood of that threat occurring.	Transverse	FIPS 200
Safety Integrity Level	Discrete level, corresponding to a range of safety integrity values	Safety	IEC 61508
Security level	Level corresponding to the required effectiveness of countermeasures and inherent security properties of devices and systems for a zone or conduit based on assessment of risk for the zone or conduit	Security	IEC 62443
Performance limitation	Insufficiencies of the function itself	Performance	SOTIF
Trade-off	Decision-making actions that select from various requirements and alternative solutions on the basis of net benefit to the stakeholders	Transverse	ISO/IEC 15288:2015

Table 4: Extract from terminology harmonisation table

A second version of this deliverable in Month 30 will report on subsequent progress towards the standards evolution objectives.

6 References

[1] "SWIM-TI Yellow Profile Technical Specification 3.1", 14.01.04.D43-004, 00.01.00, December 2015.

[2] "SWIM-TI Blue Profile Technical Specification 3.1", 14.01.04.D43-005, 00.01.00, December 2015.

[3] "SWIM-TI Purple Profile Technical Specification 3.1", 14.01.04.D43-006, 00.01.00, December 2015.

[4] D2.2.1 – Demonstrator Architecture – Air Traffic Management, AQUAS project, http://aquas-project.eu, 31.03.2018.

[5] D2.2.2 – Demonstrator Architecture – Medical, AQUAS project, http://aquas-project.eu, 31.03.2018.

[6] D2.2.3 – Demonstrator Architecture – Railways, AQUAS project, http://aquas-project.eu, 31.03.2018.

[7] D2.2.4 – Demonstrator Architecture – Industrial Drive, AQUAS project, http://aquas-project.eu, 31.03.2018.

[8] D2.2.5 – Demonstrator Architecture – Space Multicore, AQUAS project, http://aquas-project.eu, 31.03.2018.

[9] https://ec.europa.eu/growth/single-market/european-standards/harmonised-standards/medical-devices_en

[10] IEC 62443, Industrial communication networks – Network and system security, 2010.

[11] IEC 61508:2010 (ed. 2), Functional safety of electrical/electronic/programmable electronic safety-related systems, 2010.

[12] IEC 61800, Adjustable speed electrical power drive systems, 2015.

[13] ISO26262 – Road Vehicles – Functional Safety – First Edition 2011, Second Edition 2018.

[14] ISO PAS 21448 – Safety of the Intended Functionality – October 2018.

[15] D2.3 – Safety and Security Balanced Mechanisms, SESAMO project, http://sesamo-project.eu, 10 November 2014.

[16] D2.1 – Security and Safety Modelling, SESAMO project, http://sesamo-project.eu, 29 May 2013.

[17] Medini Analyze, ANSYS medini Technologies AG, http://www.medini.eu/.

[18] CHESS, INTECS, http://www.intecs.it/eng/rd_dettagli.asp?ID_RD=5.

[19] D2.1.4 – Domain Environment Industrial Drives, AQUAS project, http://aquas-project.eu, 31.10.2017.

[20] Various documents and discussions on interaction points, https://svn.trt.thalesgroup.com/repos/aquas/Work_Packages/WP3%20-%20Methodology/WP3-methodDiscussions

[21] How to specify interaction points, https://svn.trt.thalesgroup.com/repos/aquas/Work_Packages/WP3%20-%20Methodology/WP3-methodDiscussions/interactionPointScheduling_v02.pdf

[22] MALINA, L.; HAJNÝ, J.; FUJDIAK, R.; HOŠEK, J. On Perspective of Security and Privacy- Preserving Solutions in the Internet of Things. Computer Networks, 2016, vol. 102, no. 2016, p. 83-95. ISSN: 1389-1286.

[23] OMG, The OMG Hitchhiker's Guide - A Handbook for the OMG Technology Adoption Process, OMG Document: omg/2008-09-02, version 7.8, 2008.

[24] https://www.incose.org/docs/default-source/aboutse/se-vision-2025.pdf

[25] INCOSE. 2015. Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, 4th Edition

[26] https://www.sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK)

[27] van der Peijl, J., Klein, J., Grass, C., & Freudenthal, A. (2012). Design for risk control: the role of usability engineering in the management of use-related risks. Journal of biomedical informatics, 45(4), 795-812.

[28] Alberdi, E., Strigini, L., Povyakalo, A. A. & Ayton, P. (2009). Why Are People's Decisions Sometimes Worse with Computer Support? Computer Safety, Reliability, and Security, Proceedings, 5775, 18 - 31.

[29] A Guide to the Project Management Body of Knowledge – PMBOK Guide Fifth Edition, 2013 (about 580 pages)

[30] ISO 21500 Guide on Project Management (was modelled around PMBOK although there are some differences)

[31] The Human Factors Analysis and Classification System –HFACS – Scott A. Shappell, FAA Civil Aeromedical Institute Oklahoma City, OK 73125 Douglas A. Wiegmann; University of Illinois at Urbana-Champaign Institute of Aviation Savoy, IL 61874; February 2000 Final Report. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

7 Glossary

ATM	Air Traffic Management
AQUAS	Aggregated Quality Assurance in Systems
CE	Co-Engineering
CENELEC	European Committee for Electrotechnical Standardization
CO	Confidential
COTS	Commercial Off The Shelf
CPS	Cyber Physical System
DKE	Deutsche Kommission Elektrotechnik Elektronik im DIN und VDE
DSL	Domain Specific Language
ECSS	European Cooperation for Space Standardization
EN	European Norm
ESA	European Space Agency
EUC	Equipment under control
FPGA	Field Programmable Gate Array
HTTP	Hyper Text Transfer Protocol
IACS	Industrial automation and control system
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council on Systems Engineering
IOT	Internet of Things
ISO	International Standards Organization
JWG	Joint Working Group
MARTE	Modelling and Analysis of Real-time and Embedded Systems
OASIS	Organization for the Advancement of Structured Information Standards
OMG	Object Management Group
OSLC	Open Services for Lifecycle Collaboration
OTA	Over the Air Updates
PAS	Publicly Available Specification
PLC	Product life-cycle
PLM	Product Lifecycle Management
RDF	Resource Description Framework
RFI	Request For Information

RFP	Request for Proposals
RRM	Risk reduction measures
SAE	Society of Automotive Engineers
SDO	Standards Developing Organisation
SEBoK	Systems Engineering Body of Knowledge
SESAR	Single European Sky ATM Research
SIL	Safety integrity level
SOTIF	Safety of The Intended Functionality
SPARQL	SPARQL Protocol and RDF Query Language
SRS	Safety-related system
SSP	Safety, Security and Performance
S/S/P	Safety / Security / Performance
SWIM	System Wide Information Management
SysML	Systems Modeling Language
TC	Technical Committee
UAV	Unmanned Airborne Vehicle
UML	Unified Modeling Language
URI	Universal Resource Identifier
WG	Working Group
WP	Work Package