Reconciling the ISO 26262-compliant and the Agile Documentation Management in the Swedish Context

Barbara Gallina

with contribution from and Mattias Nyberg

1 Mälardalen University, Västerås, Sweden
barbara.gallina@mdh.se
2 Scania AB, Södertälje, Sweden
mattias.nyberg@scania.com

Work supported by the Swedish Foundation for Strategic Research via the SSF Gen&ReuseSafetyCases project

17th March 2016, 4th Scandinavian Conference System and Software Safety (SCSSS)
Context and motivation

Focus on document management (work products), safety management

- ISO 26262
- Agile manifesto
- "Swedish“ business/management culture

➔ Conflicting requirements and characteristics

How can reconciliation be enabled?
Talk outline

• Background
  – ISO 26262
  – Agile manifesto
  – Swedish (business/management) culture
  – Safety culture and state of practice at Scania
  – Forces to be taken into consideration

• Challenge

• Soft and hard pieces of solutions

• Conclusion and future work
ISO 26262-compliant documentation management

Concept of “Work Product” is central.

Quoting from Part 8

• “The identification of a work product in ISO 26262 shall be interpreted as a requirement for documentation”

• “The documentation process shall be planned in order to make documentation available. The documents should be:
  – a) precise and concise,
  – b) structured in a clear manner,
  – c) easy to understand by the intended users, and
  – d) maintainable”
ISO 26262-compliant safety culture

- Planning, coordinating and documenting the safety activities of all phases of the safety lifecycle are key management tasks.
- The organization shall create, foster, and sustain a safety culture that supports and encourages the effective achievement of functional safety.
  - Annex B → accountability is traceable; reward system; self-disclosure & disclosure; usage of diversity; defined, traceable and controlled process
- The organization shall institute, execute and maintain organization-specific rules and processes to comply with the requirements of ISO 26262.
Agile Manifesto

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation;
- Responding to change over following a plan.

→ radically informal and flexible
“Swedish” (business/management) culture
- from [8] and [9]

- Consensus-based informal decision making
- Flat hierarchy
- Trust
- Team-focused leadership
- "Lagom" – opinions are not strongly expressed
- and "fika" – informal communication
Safety culture and state of practice at Scania

• “Respect for the individual”
• Imposed processes, assigned jobs, and status rather than merit are not so welcome.
• Roles are weakly defined
• Destructive interference is carefully avoided
• “Elimination of all forms of waste”.

➞ commonalities with agile principles
Safely manager
Monitors progress against safety plan
Closely related to the creation of safety case

Safety manager

Monitors progress against safety plan

17th March 16, SCSSS
Forces to be taken into consideration

ISO 26262
Etc.

http://www.lindsay-sherwin.co.uk/guide_managing_change/html_change_strategy/07_mintzberg.htm
Talk outline

- Background
- Challenge
- OSLC: overview
- OSLC-based soft and hard pieces of solutions
- Conclusion and future work
A true challenge

- ISO 26262
- Agile manifesto
- “Swedish” business/management culture
- Forces-balance

How can we reconcile?
OSLC – Open Services for Lifecycle Collaboration

• Standard that:
  – targets development tools used during a product’s life cycle
  – enables tools integration and interoperability as well as traceability among their artifacts.
  – builds on top of Linked Data, Resource Description Framework, and HTTP protocol.

Remark: Each work product is described as an HTTP resource, identified via an URI. Work products are manipulated via HTTP methods.
OSLC – Open Services for Lifecycle Collaboration

OSLC Community - open-services.net, “Oslc diagram.”
In which way may OSLC be used to reconcile agile and ISO 26262 in the Swedish context?
Soft solution: Open-minded Teams for Lifecycle Collaboration

ISO 26262: safety manager can delegate tasks!

- Work performed by the different teams
- Replace safety manager by a safety case generator
- Avoid the introduction of additional hierarchical roles
- Flat hierarchy is preserved

- A safety manager should be appointed to guarantee the continuous integration of best practices, which should be suggested to the various teams

A safety manager should be mindful and vigilant

Adapted from the original OSLC figure
Hard solution: OSLC-based interoperable tools

Safety-case generator: Consumer of evidence
Producer of evidence-supported composable argument-fragments, contributing to showing that the product is acceptable safe
Hard solution: OSLC-based interoperable tools

Ongoing work
Hard solution: OSLC-based interoperable tools

NOTE Within the figure, the specific clauses of each part of ISO 26262 are indicated in the following manner: "m-n", where "m" represents the number of the part and "n" indicates the number of the clause, e.g. "4.7" represents Clause 7 of ISO 26262-4.

5.4.5 For each sub-phase of software development, the selection of the following, including guidelines for their application, shall be carried out:

a) methods; and

b) corresponding tools.

5.4.6 The criteria that shall be considered when selecting a suitable modelling or programming language are:

a) an unambiguous definition; EXAMPLE Syntax and semantics of the language.

b) the support for embedded real-time software and runtime error handling; and

c) the support for modularity, abstraction and structured constructs. Criteria that are not sufficiently addressed by the language itself shall be covered by the corresponding guidelines, or by the development environment.

NOTE 1 The selected programming language (such as ADA, C, C++, Java, Assembler or a graphical modelling language) supports the topics given in 5.4.7. Programming or modelling guidelines can be used to comply with these topics.

NOTE 2 Assembly languages can be used for those parts of the software where the use of high-level programming languages is not appropriate, such as low-level software with interfaces to the hardware, interrupt handlers, or time-critical algorithms.

Taken from ISO 26262 [ISO11]
Hard solution: OSLC-based interoperable tools

---

**NOTE** Within the figure, the specific clauses of each part of ISO 26262 are indicated in the following manner: "m-n", where "m" represents the number of the part and "n" indicates the number of the clause, e.g. "4.7" represents Clause 7 of ISO 26262-4.

**Figure 2 — Reference phase model for the software development**

5.4.5 For each sub-phase of software development, the selection of the following, including guidelines for their application, shall be carried out:

a) methods; and

b) corresponding tools.

5.4.6 The criteria that shall be considered when selecting a suitable modelling or programming language are:

a) an unambiguous definition; EXAMPLE Syntax and semantics of the language.

b) the support for embedded real time software and runtime error handling; and

c) the support for modularity, abstraction and structured constructs.

Criteria that are not sufficiently addressed by the language itself shall be covered by the corresponding guidelines, or by the development environment.

**NOTE 1** The selected programming language (such as ADA, C, C++, Java, Assembly or a graphical modelling language) supports the topics given in 5.4.7. Programming or modelling guidelines can be used to comply with these topics.

**NOTE 2** Assembly languages can be used for those parts of the software where the use of high-level programming languages is not appropriate, such as low-level software with interfaces to the hardware, interrupt handlers, or time-critical algorithms.
Hard solution: OSLC-based interoperable tools

- Current modeling:
  - OSLC-domains extension to include a subset of ISO 26262-Part 6 extended with Scania interpretation and tailoring
Conclusion and future work

- ISO 26262 in the agility-oriented Swedish context
- OSLC-based soft and hard solution

To be done:
- Further development of the proposed solution towards a PoC aimed at demonstrating the (semi)automatic traceability and work-products compilation → towards a safety case
  - Via the ongoing and planned master theses
    - [Master-thesis-1]
    - [Master-thesis-2]
    - [Master-thesis-3]
    - [Master-thesis-4]
  - Via doctoral students → AMASS
References

Thank you for your attention!

Discussion time…