Safe Design Arguments (SDAs) – tips ane examples

Piotr Gorczyca[1][0000−0002−6613−6061], Dörthe Arndt[1][0000−0002−7401−8487], Martin Diller[2][0000−0001−6342−0756], Pascal Kettmann[1][0009−0009−9461−7952], Stephan Mennicke[3][0000−0002−3293−2940], and Hannes Strass[1][0000−0001−6180−6452]

[1] Computational Logic Group, Institute of Artificial Intelligence
[2] Logic Programming and Argumentation Group, Institute of Artificial Intelligence
[3] Knowledge-Based Systems Group, Institute for Theoretical Computer Science
[1,2,3] Faculty of Computer Science, TU Dresden, Germany
firstname.lastname@tu-dresden.de

1 SDAs

The central notion of the RISKMAN ontology & shapes is that of a *Safe Design Argument* (SDA), which is a building block of the so-called SDA trees. SDA trees can be seen as simplifications of *Assurance Cases* (ACs) [2], in that they also represent structures of measures mitigating certain risk, but the AC notions (such as e.g. *claim*, *strategy*, *evidence* or *instantiation*) are by design built into the SDAs. SDAs by being uniform building blocks help to avoid the complexity of modelling risk mitigation strategies using AC approach.

SDAs allow to conceal concrete and potentially manufacturer-sensitive data by capturing only the abstract idea of risk mitigation. That enables their transparency and reusability: these abstract ideas are expected to reside in an open source repository from which they could be pulled by manufacturers whenever needed. Popular SDA would gain credability through frequent implementation (and hence certification) which would in turn hopefully lead to a so-called *safety competition*: a state in which SDAs addressing the same risk are competing to be recognized as the safest.

For a concrete risk management file submission an abstract idea is not enough: it is necessary to provide the details of implementation of the SDA within the device as well as the evidence of the implementation. This requirement is realized by means of an *Implementation Manifest*, which is a piece of information supplied to an SDA containing the aforementioned information. An SDA containing an Implementation Manifest is called a *Safe Design Argument Implementation* (SDAI).

SDA (SDAI) is the base, abstract superclass of *Risk SDA* (*Risk SDAI*) and *Assurance SDA* (*Assurance SDAI*), the two SDA subclasses to be used in practice. Whenever given SDA is referring to some state of the art *Safety Assurance* – e.g. a section of a norm or standard mentioning a particular way of handling a risk, we speak about the latter – Assurance SDA(I). Otherwise, in the case of the absence of the Safety Assurance, we are dealing with Risk SDA(I).

As an example, consider a scenario in which a risk of electrocution appears and, as prevention, the manufacturer decides to use electrical insulation. The concrete value of distance through insulation has been decided to be 0.5mm. The structure of a corresponding Risk SDA is depicted below:

$$\overbrace{\underbrace{\text{Electrical insulation}}_{\textit{Risk SDA}} \mid \overbrace{\text{dist=0.5mm}}^{\textit{Impl. manifest}}}^{\textit{Risk SDAI}}$$

On the other hand, assume that the manufacturer is aware that in their scenario (under adequate assumptions), a mitigation specified in IEC 60601-1 [4] is applicable, which further specifies the insulation distance to be at least 0.4mm. In this case the manufacturer could refer to the respective clause of IEC 60601-1 (clause 8.8.2) as the Safety Assurance of the related SDA. The structure of such Assurance SDA is shown below:

$$\overbrace{\text{Electrical insulation} \mid \overbrace{\text{IEC 60601-1, s 8.8: min. dist.} > 0.4\text{mm}}^{\textit{Safety Assurance}} \mid \overbrace{\text{dist=0.5mm}}^{\textit{Impl. Manifest}}}^{\textit{Assurance SDAI}}$$

The above example illustrates the difference between Risk- and Assurance SDA(I)s. An additional reference to a state-of-the-art measure in a form of a Safety Assurance can help in establishing SDAs credability and traceability in case of an unforeseen event.

The hierarchical relation between an SDAs similar to that known from AC, in that the children SDAs jointly realize the goal/claim of its parent. The RISKMAN approach additionally imposes the two following constraints on the structure of SDA trees:

1. all children nodes of Assurance SDA(I)s must again be Assurance SDA(I)s,
2. all leaf nodes must have an Implementation Manifest (effectively, must be SDAIs).

Non-leaf nodes may have Implementation Manifests, but it is not strictly required.

2 Insulin infusion pump example

Insulin infusion pumps aid in regulating blood glucose levels, especially of patients with diabetes, by administering fast-acting insulin via a catheter inserted beneath the skin. Based on the risk assessment for a generic infusion pump by Zhang et al. [4, 3], Fig. 1 shows a controlled risk and associated SDA that can be extracted from a risk management file that follows VDE Spec 90025.

[4] IEC 60601 is a series of technical standards published by the International Electrotechnical Commission, addressing safety aspect of medical electronical devices.

Following Zhang et al. [4, entry 4.3.9 in Table 4 in the appendix], the risk stems from a "non-audio alarm malfunction" hazard (with associated id hz in the figure). Specifically, the vibration mechanism of the non-audio alarm integrated into the pump may fail (event ev_1). Then, the patient may not become aware of an issue (event ev_2), which can lead to the patient receiving less insulin (hazardous situation hs) and the patient losing consciousness (harm hr). Apart from the information for the domain specific hazard (dsh) required by VDE Spec (dashed boxes in the figure exemplarily group related elements), the RISKMAN ontology also allows to refer to terminology for medical device problems put forward by the International Medical Device Regulators Forum (IMDRF) [1] (field with id dp). The SDA (sd_0, based on the work of Zhang et al. [3, Table 3]) consists of three sub-SDAs and expresses that there are alternative means of alerting the patient. The first sub-SDA (sd_1) specifically expresses that the alarm condition is also indicated through visual signals. Moreover, the second sub-SDA (sd_2) indicates that this notification is recurring. The third SDA (sd_3) expresses that there is also an additional audio alarm that will start unless the patient acknowledges the vibration or blinking. Moreover, according to sub-SDA (sd_5) the audible signal is in accordance with regulations, here the assurance is IEC 60601 (sa). Thus, sub-SDA (sd_5) is the only assurance SDA (indicated by the pink colour); all other SDAs are risk SDAs (purple). On the other hand, as required by VDE Spec, each leaf SDA is an SDAI, with the associated implementation manifests (im_1,im_2,im_4,im_5) pointing to implementation details and documentation.

References

1. International Medical Device Regulators Forum: Terminologies for Categorized Adverse Event Reporting (AER): terms, terminology and codes. IMDRF code: IMDRF/AE WG/N43FINAL:2020 (Edition 4) (April 2020), https://www.imdrf.org/sites/default/files/docs/imdrf/final/technical/imdrf-tech-200318-ae-terminologies-n43.pdf
2. Weinstock, C.B., Goodenough, J.B.: Towards an assurance case practice for medical devices. Tech. rep., Software Engineering Institute, Carnegie Mellon University (2009), Technical Note CMU/SEI-2009-TN-018
3. Zhang, Y., Jetley, R., Jones, P.L., Ray, A.: Generic safety requirements for developing safe insulin pump software. Journal of diabetes science and technology **5**(6), 1403–1419 (2011)
4. Zhang, Y., Jones, P.L., Jetley, R.: A hazard analysis for a generic insulin infusion pump. Journal of diabetes science and technology **4**(2), 263–283 (2010)

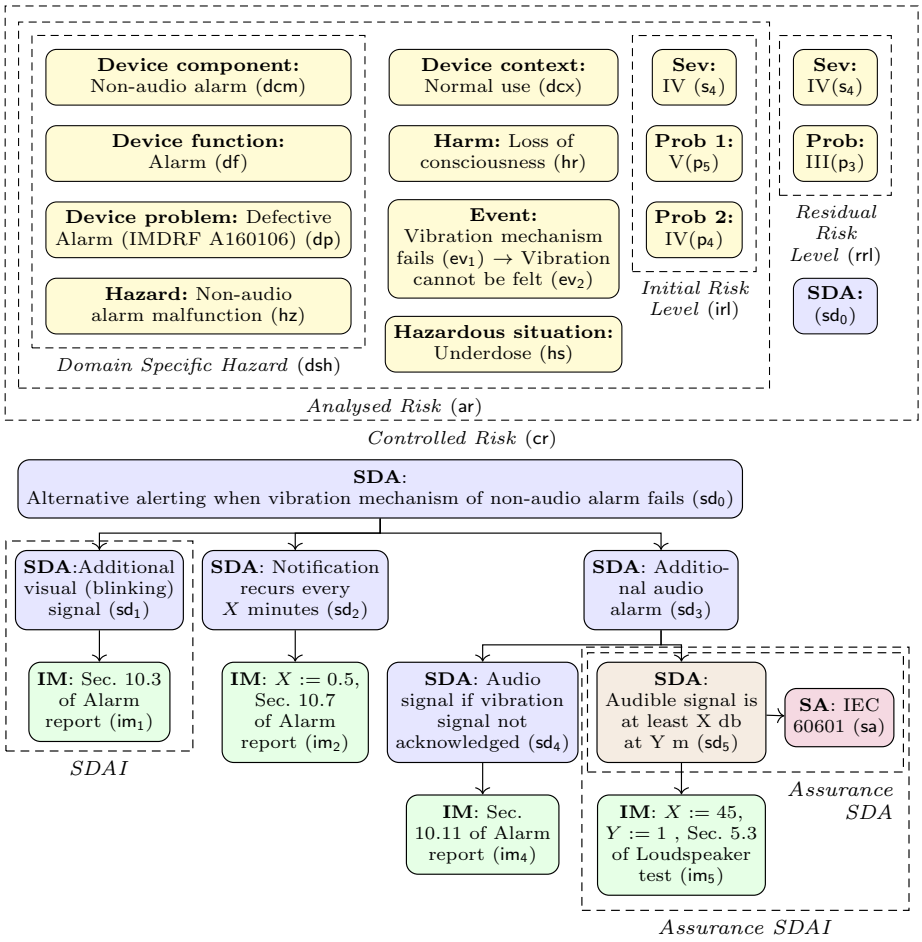

Fig. 1. Graphical representation of the data of a controlled risk (top figure) and associated SDA (bottom figure) provided within a risk management file for an infusion pump. The dashed indicate examples of composite classes, e.g. (sd_1) and (im_1) together constitute an SDAI. Note that these are just single examples of SDAIs, as any pair of an SDA and its related Implementation Manifest jointly realize an SDAI, e.g. the pair (sd_2) + (im_2) or (sd_4) + (im_4).