Safety Critical Software Process Assessment: How MDevSPICE® Addresses the Challenge of Integrating Compliance and Capability

Paul Clarke[1], Marion Lepmets[1], Alec Dorling[1], Fergal McCaffery[1]

[1] Regulated Software Research Centre, Dundalk Institute of Technology,
Ireland
{paul.clarke, marion.lepmets, alec.dorling, fergal.mccaffery}@dkit.ie

Abstract. One of the primary outcomes of a software process assessment is visibility of the capability of a software process which among other things, informs us of the ability of a process to deliver consistent product quality levels. In safety critical domains, such as the medical device sector, high product quality – and particularly high product safety - is an important consideration. To address this safety concern, the medical device sector traditionally employs audits to determine compliance to software process standards and guidance. Unlike an audit which results in a pass/fail outcome, an assessment provides a process capability profile which identifies areas for improvement and enables a comparison with broader best practice. MDevSPICE® integrates the various medical device software standards and guidance within the infrastructure of a SPICE assessment model, thus encompassing aspects of compliance and capability. This paper describes some of the key enablers of this integration.

Keywords: Safety Critical Software · Medical Device Software · Software Process Improvement · Software Process Assessment · MDevSPICE®.

1 Introduction

Safety critical software is software which if not operating correctly can result in harm or even death to humans [1]. It is therefore the case that safety critical software development should take additional steps beyond general software development to specifically address safety considerations. Medical device software is of a safety critical nature and regulators have implemented legal requirements (or regulations) which must be met prior to placing a device on the market. Whereas general software development studies have demonstrated that software developers may be unwilling to embrace a strong software process focus [2], such processes are a mandatory requirement in the medical device sector. These regulations are typically regional in application, with the Food and Drug Administration (FDA) and the European Commission (EC) regulating for the US and EU respectively.

In the case of the EU, medical device regulation is contained in the Medical Device Directive (MDD) 93/42/EEC [3], the Active Implantable Medical Device Directive

(AIMDD) 90/385/EEC [4], and the In-vitro Diagnostic (IVD) Medical Device Directive 98/79/EC [5] – with MDD 2007/47/EC [6] amending these earlier directives. In the US, the FDA advances medical device regulation through Code of Federal Regulation (CFR) Title 21, Chapter I, Subchapter H, Part 820 [7]. In both the EU and the US regulations, provisions are made for classifying medical devices depending on the role of the device, ranging from Class I to Class III depending on the extent of the role of the device in supporting or sustaining human life. Various standards and guidance exists to support manufacturers developing medical devices in adhering to the regulations, and compliance to these standards will generally enable market access. Primary among these standards are ISO 14971 [8], ISO 13485 [9], IEC TR 80002-1 [10], IEC 62304:2006 [11], IEC TR 80002-3 [12], IEC 62366 [13], IEC 60601-1 [14], IEC 82304 [15]), the FDA guidance documents on premarket submission [16], off-the-shelf software use [17] and software validation [18]. These standards and guidance documents are presented in Figure 1 and in further detail in [19].

Despite the existence of regulation, standards and guidance for medical device software, the proportion of medical devices being recalled owing to software errors is growing. From a base of less than 10% for most of the 1990s, the proportion of medical device recalls attributable to software errors hit 24% in 2011 [20] and this trend would appear to be set to continue upwards. Although one of the reasons for this growth is undoubtedly the increasing use of software in medical devices, other factors such as inadequacies in the software development process could have a role to play. To help address this undesirably upward trend in the proportion of medical device recalls attributable to software errors, the introduction of a SPICE-based process assessment may be of benefit, especially given the significant positive impact that SPICE models have had within other safety critical sectors, e.g. Automotive SPICE [21].

2 Integrating Medical Device Standards into SPICE

While there are potentially significant benefits to be derived from the use of SPICE based assessment in the medical device sector, the task of integrating the existing medical device standards and guidance into the SPICE framework is a challenging one. Medical device standards and guidance are rich in detail, varied in origin, and sometimes with overlapping content. As a result, a significant burden of effort is required to integrate these disparate sources into a single, comprehensive framework. Furthermore, SPICE frameworks do not typically trace the origin of different process requirements – rather, a SPICE assessment is performed against the accumulated best practice that is incorporated into the framework. Therefore, the following significant challenge emerged during the development of MDevSPICE®:

> **Challenge:** How can the origin of different process requirements be carried forward into a SPICE framework such that assessments can also assist manufacturers in addressing their basic standards compliance requirements?

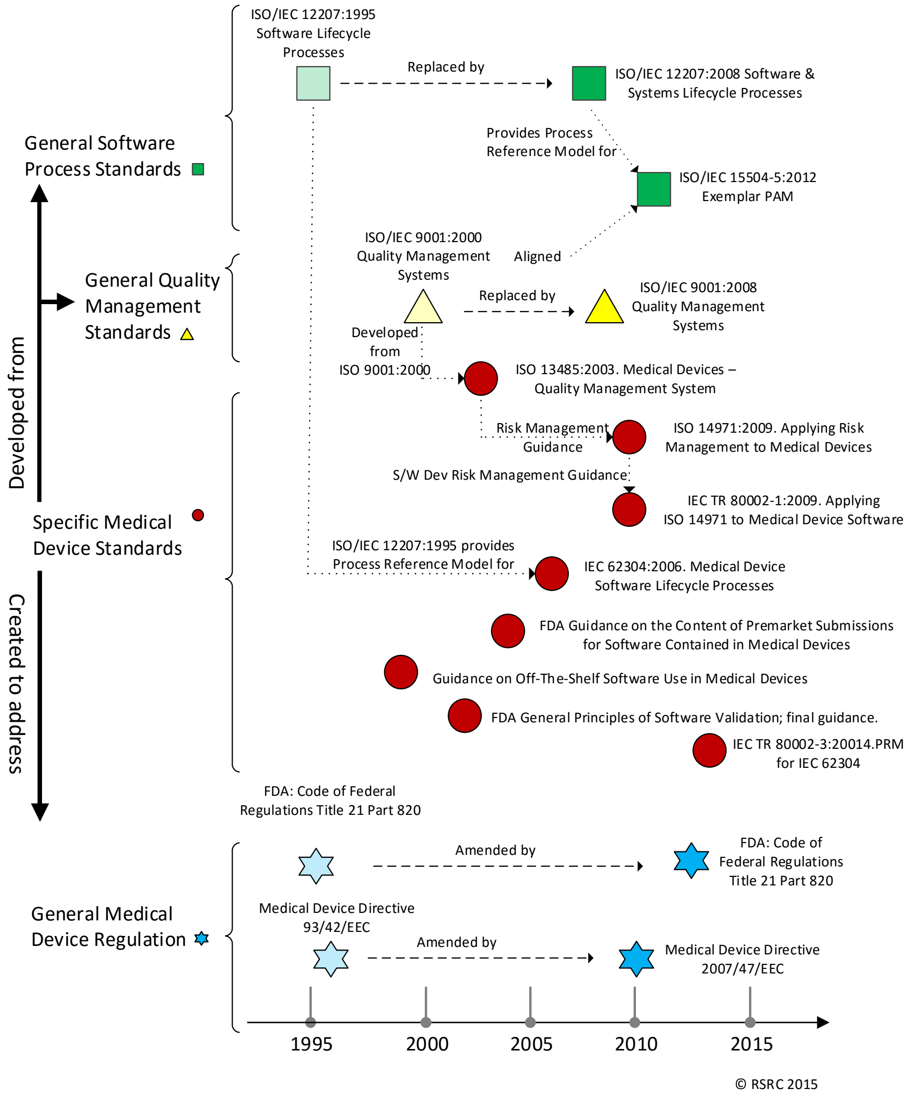

Fig. 1. Medical Device Standards and Regulation

2.1 Process Lineage

To address the challenge identified above, we determined to identify the source for various specific standards and guidance contained within the MDevSPICE® Process Assessment Model (PAM). All MDevSPICE® assessments operate upon the basic assumption that IEC 62304 is within the scope of an assessment – as it is the globally recognised standard that should be adopted when performing medical device software development. Therefore, IEC 62304 requirements form the foundation of MDevSPICE® [22]. In fact, an important foundation to this was the publication of IEC 80002-3 [12] by the developers of MDevSPICE® as a Process Reference model for IEC 62304. IEC 80002-3 was then used as the starting point for the development of the MDevSPICE® PAM. Software safety classification (detailed in IEC 62304) is a concept similar to medical device classification, with three classes existing and these being determined based upon the worst possible consequence in the case of a software failure.

Building on IEC 62304, additional details incorporated from aligned standards and guidance documents retain information regarding the source of such details. For example, in the case of the Software Architectural Design process, when describing the software architecture (which is the first base practice for this process), there is an explicit reference to an item of FDA Guidance on the use of off the shelf software, specifically addressing the treatment of Software Of Unknown Provenance (SOUP) as follows:

FDA on OTS: *identify the expected design limitations of the SOUP Software.*

There are many further examples of such additions and indications in the MDevSPICE® PAM, for instance, in the Software Requirements Analysis process, there is an addition to base practice five *Verify Software Requirements*, as follows:

FDA on Validation: *A software requirements traceability analysis should be conducted to trace software requirements to (and from) risk analysis results.*

While software process adaptation is considered advantageous in a general sense [23], it is however a requirement for medical device manufacturers since regulation (and corresponding standards and guidance) is subject to change. In this respect, MDevSPICE® is of particular benefit to manufactures. Through our work in the Regulated Software Research Centre, the creators of MDevSPICE® will continue to work with the international standards organisations to develop and improve standards and guidance, with corresponding updates being further applied to the MDevSPICE® framework. Thus, manufacturers can continually adopt MDevSPICE® for the purpose of staying abreast of standards and guidance evolution.

2.2 Assessment Questions

To further aid MDevSPICE® Assessors, a suite of detailed questions has been developed as a counterpart to the PAM. The detailed questions are directly linked to the PAM, with specific questions designed to address aspects of the PAM that have been incorporated from sources other than IEC 62304 and which therefore could be important within the context of determining approximate standard or guidance compliance. It should be noted that it is not the intention to use MDevSPICE® assessments to certify compliance to individual standards. Instead, MDevSPICE® is used to determine the capability of the software development process relative to the accumulated best practice information available. Having process lineage to individual source standards and guidance, and providing a set of corresponding questions does however permit a reasonably accurate approximation to standards compliance and it is possible that in the fullness of time and subject to robust validation, MDevSPICE® could concurrently address the capability consideration that is central to all SPICE assessments (and which enables targeted process improvement and supplier selection) in tandem with undertaking a compliance audit (which would enable market access).

3 Conclusion

Safety critical software development is often subject to regulation, with that regulation being realized through the implementation of associated standards and guidance. The aim of regulations is to reduce the risk of harm to humans in so far as is possible. This is achieved through the adoption of a robust software development process – with such a process effectively increasing product quality and thereby safety. Process assessment frameworks such as SPICE have been designed to achieve (among other things) higher levels of product quality and have been used to good effect in safety critical domains such as the automotive sector. Therefore, MDevSPICE® was developed for the medical device sector to deliver a view on process robustness though the capability lens, while simultaneously providing an approximation to standards and guidance compliance. However, standards compliance and process capability are not necessarily natural bedfellows, and the creation of MDevSPICE® has had to incorporate some innovations to enable the integration of both concerns. Specifically in this respect, MDevSPICE® retains linkage to the source standards. To further enable the harmonization of compliance and capability considerations, the MDevSPICE® PAM has an associated set of questions that align with the process components in the PAM. Through the MDevSPICE® process question set, MDevSPICE® Assessors can consistently examine both process capability and standards alignment in a single engagement, and this, we believe, represents an important step forward for medical device software development.

Acknowledgments. This research is supported by Enterprise Ireland and the European Regional Development Fund (ERDF) under the National Strategic Reference Framework (NSRF) 2007-2013, grant number CF/2012/2631, and in part by the Science Foundation Ireland Principal Investigator Programme, grant number 08/IN.1/I2030 and by Lero - the Irish Software Research Centre (http://www.lero.ie) grant 10/CE/I1855 & 13/RC/20194.

References

1. Turk, D., France, R., Rumpe, B.: Limitations of agile software processes. In: Proceedings of Third International Conference on eXtreme Programming and Agile Processes in Software Engineering Italy (2002)
2. Clarke, P., O'Connor, R.V., Yilmaz, M.: A hierarchy of SPI activities for software SMEs: Results from ISO/IEC 12207-based SPI assessments. In: Mas, A., Mesquida, A., Rout, T., O'Connor, R.V., Dorling, A. (eds.) SPICE 2012. CCIS, vol. 290, pp. 62-74. Springer, Heidelberg (2012)

3. European Commission, Directive 93/42/EEC of the European Parliament and of the Council concerning medical devices, in OJ o L 247 of 2007-09-21. 1993: EC, Brussels, Belgium.
4. European Commission, Council directive 90/385/EEC on active implantable medical devices (AIMDD). 1990: Brussels, Belgium.
5. European Commission, Directive 98/79/EC of the European parliament and of the council of 27 October 1998 on in vitro diagnostic medical devices. 1998: Brussels, Belgium.
6. European Commission, Directive 2007/47/EC of the European Parliament and of the Council concerning medical devices, OJ no L247 2007-09-21. 2007, EC: Brussels, Belgium.
7. FDA. Chapter I - Food and drug administration, department of health and human services subchapter H - Medical devices, Part 820 - Quality system regulation. [cited 2015 06.03]; Available from: http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?CFRPart=820.
8. ISO 14971:2007, Medical Devices — Application of risk management to medical devices. ISO: Geneva, Switzerland. (2007)
9. ISO 13485:2003, Medical devices — Quality management systems — Requirements for regulatory purposes. ISO: Geneva, Switzerland. (2003)
10. IEC/TR 80002-1:2009, Medical device software Part 1: Guidance on the application of ISO 14971 to medical device software. BSI: London. (2009)
11. IEC 62304:2006, Medical device software—Software life cycle processes. IEC: Geneva, Switzerland. (2006)
12. IEC/TR 80002-3:2014, Medical Device Software - Part 3: Process reference model for medical device software life cycle processes (IEC 62304). ISO: Geneva, Switzerland (2014)
13. IEC 62366:2007, Medical devices - Application of usability engineering to medical devices. IEC: Geneva, Switzerland. (2007)
14. BS EN 60601-1:2005 Medical electrical equipment – Part 1: General requirements for basic safety and essential performance. IEC: Geneva, Switzerland. (2005)
15. IEC/CD 82304:2014, Health Software - Part 1: General Requirements for Product Safety. ISO: Geneva, Switzerland. (2014)
16. US FDA Center for Devices and Radiological Health, Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices. CDRH: Rockville (2005)
17. US FDA Center for Devices and Radiological Health, Off-The-Shelf Software Use in Medical Devices; Guidance for Industry, medical device Reviewers and Compliance. CDRH: Rockville (1999)
18. US FDA Center for Devices and Radiological Health, General Principles of Software Validation; Final Guidance for Industry and FDA Staff. CDRH: Rockville (2002)
19. Lepmets M, Clarke P, McCaffery F, Finnegan A, Dorling A. Development of MDevSPICE® - the Medical Device Software Process Assessment Framework. To appear in: Journal of Software: Evolution and Process.
20. FDA. FDA News on Software Failures Responsible for 24% of all Medical Device Recalls. 2012 [cited 2015 06.03]; Available from: http://www.fdanews.com/newsletter/article?articleId=147391&issueId=15890.
21. Automotive SIG, Automotive SPICE Process Assessment V 2.2. 21 August (2005)
22. McCaffery, F., Clarke, P., Lepmets, M..: A Lightweight Assessment Method for Medical Device Software Processes. In: Mitasiunas, A., Rout, T., O'Connor, R., Dorling, A. (eds) SPICE 2014. CCIS, vol. 447, pp.144-156. Springer, Heidelberg (2014).
23. Clarke, P., O'Connor, R.V.: An Approach to Evaluating Software Process Adaptation. In: O'Connor, R.V., Rout, T., McCaffery, F., Dorling, A. (eds.) SPICE 2011. CCIS, vol. 155, pp. 28-41. Springer, Heidelberg (2011)