Creation of an IEC 62304 compliant Software Development Plan

Peter Rust, Derek Flood, Fergal McCaffery
{peter.rust, derek.flood, fergal.mccaffery}@dkit.ie

Abstract

Organizations engaged in medical device software development are required to demonstrate compliance with a set of medical device standards and regulations before the device can be marketed. One such standard IEC 62304, Medical device software - Software life-cycle processes, defines the processes that are required in order to develop safe software. Demonstrating compliance with IEC 62304 can be problematic for organizations that are new to or have limited experience in the domain. The standard defines what processes must be carried out, but does not state how. In a review of a number of such organisations it was found that the development of a software development plan proved to be a difficult task. In this work we have created a software development plan template to assist organisations with this arduous task.The software development plan template will be validated with these organisations as part of the future work.

Keywords

Regulatory compliance, Software Process Improvement, Software Process Improvement Roadmaps, IEC 62304, Medical device Software Development Plan

1 Introduction

Medical devices have been around for centuries but it is only in the last decades of the twentieth century that software has become widespread in the operation and control of some kinds of medical devices [5]. It is because of the critical nature of medical device software and due to the increase in the number of recalls of medical devices arising from software failures that regulatory bodies acted to try and rectify this growing trend.

To address these issues international standards organisations have developed a number of medical device standards which aim to regulate how organisations implement medical device software. These standards outline what organisations must do to ensure the development of quality medical device software processes, however they do not specify how they should do it. Existing software process improvement frameworks such as MDevSPICE® (formerly known as Medi SPICE) allow organisations to examine their existing processes in light of these standards but do not provide specific detail on how to implement the processes.

In previous work, an IEC 62304 implementation roadmap has been developed [8] and is currently being prepared for validation by industry experts. Through contact with software development organisations, the first element causing a major difficulty was the creation of a software development plan as described in Section 5 of IEC 62304. These organisations did not have the experience to develop such a document. This paper describes the development of a software development plan template that complies with IEC 62304 and would be suitable for small to medium size medical device software development organisations.

2 Related Work

2.1 Medical Device Software Quality

Wallace and Kuhn [5] describe how in the years, 1983 to 1991 6% of the recalls registered with the FDA were due to software failures and how for the years 1994 to 1996 this had risen to 10%. ANSI/AAMI/SW68 Medical device software - Software life cycle processes [6] was adopted in 2001 and its stated purpose was to reduce the time required for regulatory review of medical device software by reducing the material that must be reviewed while providing a development process that will consistently produce high quality, safe medical device software. IEC 62304 was introduced in 2006 and is based on ANSI/AAMI/SW68 with a number of significant additional requirements. IEC 62304 has been adopted by the ANSI as an US national standard (replacing ANSI/AAMI/SW 68). However the number of medical device recalls registered with the FDA that related to software issues has continued to increase. Alemzadeh et al.[7] describe how 33.3% of Class I (presenting a high risk of severe injury or death to patients) recalls between 2006 and 2011 were software related. The standards state clearly what is required by medical device software development organisations, but do not tell the organisation how to implement these requirements.

The development of safe medical device software requires quality management, risk management, and good software engineering [1]. The purpose of IEC 62304 Medical device software — Software life-cycle processes [2] is to define the lifecycle requirements for medical device software development and to establish a common framework for medical device software life cycle processes. IEC 62304 also requires a medical device software development organisation to have a quality management system in place that demonstrates the ability to provide medical device software that consistently meets customer requirements and applicable regulatory requirements. ISO 13485 Medical devices - Quality management systems - Requirements for regulatory purposes [3] is one such standard. IEC 62304 also requires that a risk management process complying with ISO 14971 [4] be applied to the software development life cycle processes.

2.2 Roadmapping

The roadmapping process is established and proven in the technology domain and continues to be adopted in many other fields of endeavour. Phaal [9] lists over 2000 public domain roadmaps organized by topic including chemistry, construction, defence, energy, transport and many more. A number of large companies use roadmapping to develop their strategic planning going forward. NASA embraced roadmapping in 2005 [10] arising out of a number of cost overruns in their development budgets.

Within the SPI domain, the number of published roadmaps is limited. McFeeley et al., [11] have developed a high level process improvement roadmap and describe how their roadmap is intended to provide an organization with a guide to forming and carrying out an SPI program.

Höss et al., [12] launched a pilot project to acquire skills in implementing IEC 62304 in a hospital-based environment (in-house manufacture). They concluded that the pilot project carried out at their facility clearly demonstrated that the interpretation and implementation of IEC 62304 is not feasible without appropriately qualified staff. They recognized that it could be carried out by a small team with limited resources although the initial effort is significant and a learning curve must be overcome.

It can be seen that applying the roadmapping process to IEC 62304 and generating a roadmap that will aid medical device software development organizations in the implementation of IEC 62304 is a necessary and justified step.

Flood et al. [13][14] have already applied the roadmapping process to ISO 14971 and IEC 62366 and these roadmaps have been validated with industry experts. A roadmap has also been developed for traceability in the medical device domain leaving the development of an IEC 62304 roadmap as the

last piece of the puzzle.

IEC 62304 is not a standalone standard and the manufacturer of a medical device is responsible for ensuring compliance with the other relevant standards. Irrespective of the lifecycle model chosen, the processes defined in the standard must form part of the model and be implemented during the development of the medical device software. One method organizations have of doing this is through mapping the standard to their particular life cycle model. The IEC 62304 implementation roadmap will remove this step in the software development process as the requirements of IEC 62304 are already mapped to the defined processes, identified as Activities and any gaps that exist in the organizations processes will be detected.

2.3 General Software Development Planning

The Institute of Electrical and Electronics Engineers (IEEE) produced Standard 1058:1998 Standard for Software Project Management Plans [15] to specify the format and content of software project management plans. When ISO/IEC/IEEE 16326:2009 Systems and software engineering — Life cycle processes — Project management, which harmonised ISO/IEC TR 16326:1999 and IEEE Standard 1058:1998, was introduced, software development plans were identified as separate entities. Section 5.9 Additional plans (Clause 9 of the PMP) states "*For projects dealing with software intensive systems or software products, these additional requirements are usually documented in two additional plans created at a lower level of abstraction than the PMP. These additional plans are the system engineering management plan (SEMP) and the software development plan (SDP)*." These standards state what must be contained within a plan but do not give examples of such a plan. McConnell, in his Software Project Survival Guide [16], details a software project development plan template, based on IEEE 1058 – 1998. One of the main features of this plan is that it separates the managerial processes, the technical processes and the work packages, schedule and budget.

3 A roadmap for IEC 62304

The definition of a Roadmap for the purposes of applying the roadmapping process to this and the other standards in the domain is *"A series of Activities, comprised of Tasks that will guide an organization through the use of specific "How To's" towards compliance with regulatory standards".*

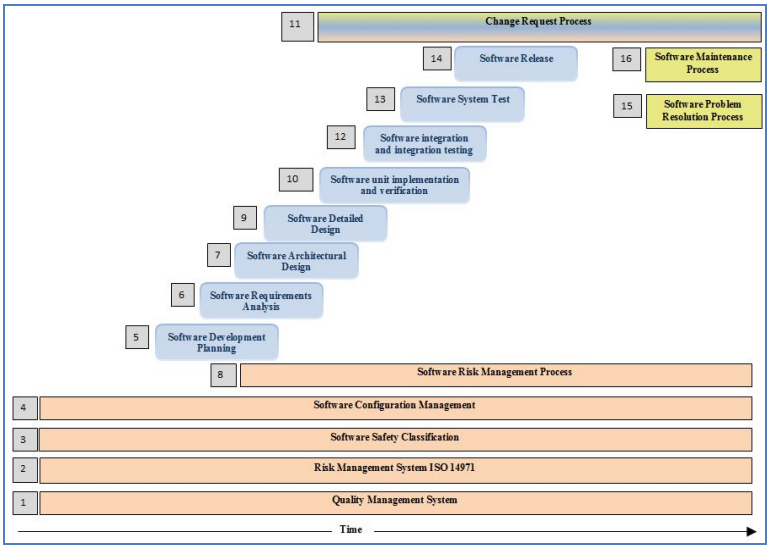

Figure 1 Metaphor for Roadmap

A roadmap for the implementation of IEC 62304 has been developed and the metaphor detailed in figure 1 was developed to aid organisations in the visualisation of the activities that were required and

also to give an indication of the timeline associated with the implementation of the processes required by IEC 62304.

It can be seen in this figure that a number of the required processes are on-going right throughout the development of the project and in some cases, right to the end of the life of the product. These processes have been defined to ensure the safety of all users of the medical device, including operators, patients and healthcare professionals.

The roadmap outlines the stages at which each process should be introduced into the organisation and these processes may be performed multiple times through the life of the project. As can be seen organisations must first start with the implementation of a quality management system and a risk management process compliant with ISO 14971. The organisation must then establish the classification of the device and ensure that for all artefacts produced as part of the project are controlled and uniquely identified, including all modifications and revisions.

At this point the organisation would be in a position to begin the development of the medical device software. The first stage in this process is to plan the software development process including all of the necessary stages from requirements analysis through to releasing the process.

4 Research Method

The aim of this work is to assist organisations with the implementation of IEC 62304. To meet this aim we began by examining the current software development practices of two organisations new to the medical device domain in light of the IEC 62304 standard. This work revealed that the most prominent issue was a lack of a software development plan.

To assist these organisations in the creation of the software development plan the following research method was undertaken.

1. **Examine current software development practices within the medical device organisations:** The first stage in the implementation of the roadmap was to examine the organisations existing processes and determine which elements were most urgently required. This examination revealed that the most prominent issue faced by these organisations was the lack of a software development plan.
2. **Examine general Software Development Plans and compare them with the requirements of IEC 62304:** The requirements of IEC 62304 were mapped into the template and a comparison made between the contents of the template and the requirements of IEC 62304. Details from the annexes were also included so that the rationale behind the requirements were understood and could be easily referenced by the authors of the actual medical device software development plan.
3. **Develop Generic Software Development Plan template which satisfies the requirements of IEC 62304:** The outcome of the comparison process was a generic medical device software development plan that encompassed all elements of the original project plan, elements derived from best practice, the requirements of IEC 62304 and the rationale from the annexes of the standard.
4. **Examined the organisations existing planning documentation in light of the template:** Once the template was developed the organisations existing planning documentation was examined to determine if additional elements were required to complete the software development plan template developed in Stage 3.

5 Results

5.1 Examination of Current Software Development Practices

As described previously, contact was made with two software development organisations who were

planning to enter the medical device software development domain. Organisation A is a small software development house with approximately fifteen employees. They are specialists in the mobile application market. Organisation B is an organisation who as part of a multinational company has experience of manufacturing medical devices and some limited experience in developing medical device software but who have identified the need to set up a separate division to take responsibility for the development of medical device software that will form part of any medical device that they manufacture. Both organisations have already begun their journey on the road to IEC 62304 compliance by investing in a quality management system complying with ISO 13485 [3] (QMS) and a risk management system complying with ISO 14971 [4] (RMS). Their question was "what do we have to do now?"

Both organisations had commenced Activities 1 (QMS) and 2 (RMS). The software safety classification of Class C is assigned at the start of the project (Activity 3) and remains as such until the software risk management process (Activity 8) is commenced and the results of this process are established. The software configuration management process (Activity 4) was established in both organisations to a certain degree, established enough for them to commence planning the software development (Activity 5). Both organisations had difficulty with this stage. Reviewing their existing documents established that Standard Operating Procedures (SOP) were the methods employed to detail, describe and record the processes carried out by their staff. These procedures were not robust enough and did not cover all the requirements of IEC 62304. Neither organisation had what could be described as a Software Development Plan (SDP) as required by IEC 62304. This is the point where the "How To" element of the roadmap was required.

5.2 Comparison of general Software Development Plan with IEC 62304

There are a number of software project development plan templates available. Most of these are based on the IEEE standard. Because standalone software can be a medical device in its own right [17], a decision was made to base this software development plan template on the software project management plan template. The template refined by McConnell was used because of his standing in the software engineering world. Medical device software development organisations operate in a business environment as well as a safety critical one. The sections and references to budget were retained to reflect this reality.

Using the template detailed by McConnell, a comparison was made with IEC 62304 to identify any limitations that were apparent. Section 5 of IEC 62304 is titled Software Development Process and Sub-Section 5.1 (Software Development Planning) contains eleven sub clauses, 5.1.1 through 5.1.11, that detail the requirements for the software development planning stage. These are included in the roadmap under the Activity Title - Software Development Planning and grouped into five tasks as detailed in Table 1:

Table 1 Tasks associated with Activity 5 – Software Development Planning

Task	Description
5.1	Establish a Software Development Plan
5.2	Update the Software Development Plan
5.3	Ensure that the plan includes references to or plans for the following elements. (Task 5.3 references thirteen separate elements required by IEC 62304).
5.4	Identify the supporting items used to develop the medical device software, which could impact the medical device software.
5.5	Plan to place configuration items under documented configuration management control before they are verified.

These five tasks reference seventeen elemental requirements from Sub-Section 5.1.

The limitations identified to the software project management plan described by McConnell included a limitation related to the question of risk. In the general software engineering world, risk to the project is a concern whereas for the medical device domain IEC 62304 requires that the software risk manage-

ment process “identify software items that could contribute to a hazardous situation identified in the medical device risk analysis activity of ISO 14971”. Safety to the patient, the user and the environment are of paramount importance. Section 3.3 was retitled to Software Risk Management, mapped to IEC 62304 section 5.1.7 Software risk management planning, which in turn references the Software Risk Management Process described in Section 7 of IEC 62304.

A second limitation of the generic plan was that Section 1.1 Project Overview did not highlight the importance of safety and producing safe software as an objective. Reference to safety is a major element of this section in the template, so that from the beginning of the project, safety is to the forefront.

5.3 Generic Software Development Plan template

These requirements were mapped to the most appropriate headings in the revised template above. As an example, IEC reference 5.1.7 details the requirements with regard to software risk management and this was mapped to the heading 3.3 (Software Risk Management) in the software development plan. This mapping of each of the requirements of Sub – Section 5.1 to the template ensures that the plan encompasses all of the requirements of IEC 62304. The software safety classification associated with each requirement is also mapped. It is important to note that the plan is a “living” document and will be continuously reviewed and updated during the software lifecycle process.

The requirements of Section 5.1 of IEC 62304 were mapped to the template to ensure that all requirements were included in a particular section of the template.

The final template reflects the requirements of a software development plan for the medical device domain and by the addition of elements from current best practice, the template now comprises the following sections:

Title Page
Signature Page
Revision History
Preface
Contents
List of Figures
List of Tables
List of Definitions, Abbreviations and Acronyms
1. Introduction
    1.1 Software Project Overview (referencing main project if appropriate)
    1.2 Software Project Deliverables
    1.3 Schedule and Budget Summary
    1.4 Evolution of the Software Development Plan
    1.5 Reference Materials
2. Software Project Organization (referencing main project if appropriate)
    2.1 Process Model
    2.2 Organizational Structure
    2.3 Organizational Boundaries and Interfaces
    2.4 Project Responsibilities
        2.4.1 RACI Matrix
3. Managerial Process
    3.1 Management Objectives and Priorities
    3.2 Assumptions, Dependencies, and Constraints
    3.3 Software Risk Management
    3.4 Monitoring and Controlling Mechanisms
    3.5 Staffing Plan
    3.6 Measurement and Analysis
4. Technical Process
    4.1 Standards, Methods, Tools, and Techniques
    4.2 Software Documentation
    4.3 Project Support Functions
5. Work Packages, Schedule, and Budget
    5.1 Work Packages
    5.2 Dependencies
    5.3 Resource Requirements
    5.4 Budget and Resource Allocation
    5.5 Schedule
6. Additional Components
7. Index
8. Appendices

This template is easily tailored to match the requirements of any medical device software development organisation. The plan can be tailored by means of including elements of their processes under specific headings or in the Appendices. IEC 62304 only requires as a minimum that the SDP references the plans to be included. Therefore the organisation is able to identify the elements of the plan that will be required going forward and be able to adequately plan for and resource these processes.

5.4 Review of Organisations Documentation

The organisation's existing planning documentation was reviewed and a comparison made with the template. A number of elements were identified as either lacking in detail or were non-existent. Further follow up interviews elicited more information on the existing processes being used and the outcome of the review noted the following elements that required action:

- **Safety** - the main driving force behind the development of the standards for medical device and medical software development is to identify the processes that are required to produce safe design and maintenance of medical devices and software. The need to keep safety at the forefront of all discussions, designs and implementations is the element that needs to be included and highlighted within the SDP.
- **Maintenance** – The last process identified in the roadmap is that of Maintenance, but nonetheless a very important feature of medical device software development. Small organisations entering the medical device software development arena need to identify and plan for this process.
- **Language** – the medical device domain requires a whole new lexicon. It is not expected that developers become medical experts, but they need to understand the basic definitions and treatment methods associated with the device being designed. Understanding acronyms, units of measurement and dosage levels for instance could be crucial. The training requirements of the staff engaged on the project are required to be identified in the SDP.
- **Error handling** – in the general software development domain designing what happens when an error occurs will probably not cause harm to the user. In safety critical domains the results of error handling can be life threatening. The importance of error handling and the need to consider a much wider set of circumstances and implications is crucial in safety critical domains.

6 Discussion

IEC 62304 is an important and substantive standard. The purpose in developing an implementation roadmap is to guide medical device software development organisations of all maturity to regulatory compliance. Organisations who want to enter or who have just started their journey in entering the market find it difficult to decide how to undertake the processes necessary or even adapt or improve the processes that they have in place so that they can comply with all the requirements of IEC 62304. The roadmap, when complete, will contain a repository of the "How To's" necessary to guide such organisations.

IEC 62304 defines the processes required for the development of safe software for the medical device domain but does not tell the organization "how to" carry out the processes. The generated roadmap when completed will fill this gap. The experienced gained so far in generating the roadmap and interacting with organisations operating in the medical device software development domain has been invaluable in the understanding of the needs of these organisations and the best manner in which these needs can be fulfilled.

7 Conclusions

Organizations that are engaged in or wish to become engaged in the medical device software development domain are placed under a high level of scrutiny by the regulatory bodies tasked with ensuring that the medical device organization is compliant with all the standards. These standards identify the requirements the medical device organization must satisfy without telling them how to achieve compliance. This can hinder both the development of new medical devices and existing software houses entering the medical device domain due to the range of methods available for implementing the

standards.

Building on the previous work in developing the IEC 62304 implementation roadmap this paper has described the development of the first “How To” that will form the first block in the repository of “How To’s” for the roadmap. These “How To’s” are the essential ingredient that an organisation needs in starting the journey on the road to IEC 62304 regulatory compliance.

8 Future Work

The fourth stage of this work is to validate the roadmap through expert review. A number of experts will be recruited for the validation from a diverse range of back-grounds including those who work in the medical device domain and use the standards on a regular basis, assessors who regulate organizations using the standard, academics with the appropriate expertise, and members of the standards committee.

It was envisioned that the fifth stage, the identification of the “how to’s” for the achievement of the Tasks defined in the generated roadmap and the building of a repository to house them would only commence following validation of the roadmap. However, as is the case in many endeavours, “there is a tide in the affairs of men, which taken at the flood, leads on to fortune” (Julius Caesar Act 4, scene 3), it was decided to take the opportunity when it arose to work with both organisations in developing an SDP template. The plan template is currently being evaluated by the organisations and the results will be evaluated in due course. Further “How To’s” will be developed through interaction with organizations that are close to regulatory compliance and by assessment of their processes. This will enable the development of a robust repository of “How To’s” and future implementations of the roadmap in medical device organizations.

9 Acknowledgement

This research is supported 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.

10 Literature

[1] S. Eagles, “62304 and TIR32 Training Slides.” [Online]. Available: https://goo.gl/lr0YpT [Accessed: 12-Mar-2015].

[2] “IEC 62304:2006 (2006) Medical device software—Software life cycle processes. Geneva, Switzerland, IEC.,” vol. 3. 2006.

[3] “ISO 13485:2003 - Medical devices -- Quality management systems -- Requirements for regulatory purposes,” *Geneva, Switzerland, ISO*. 2003.

[4] “ISO 14971:2007 - Medical devices -- Application of risk management to medical devices,” *Geneva, Switzerland, ISO*. 2007.

[5] D. R. Wallace and D. R. Kuhn, “Failure Modes in Medical Device Software: an Analysis of 15 Years of Recall Data,” *Int. J. Reliab. Qual. Saf. Eng.*, vol. 08, no. 04, pp. 351–371, 2001.

[6] AAMI, “ANSI/AAMI SW68:2001 Medical device software — Software life cycle processes.” 2001.

[7] H. Alemzadeh, R. K. Iyer, Z. Kalbarczyk, and J. Raman, “Analysis of Safety-Critical Computer Failures in Medical Devices,” *IEEE Secur. Priv.*, vol. 11, no. 4, pp. 14–26, Jul. 2013.

[8] P. Rust, D. Flood, and F. McCaffery, “Software Process Improvement & Roadmapping – A Roadmap for Implementing IEC 62304 in Organizations Developing and Maintaining Medical Device Software,” in *SPICE 2015*, 2015.

[9] R. Phaal, “Public Domain Roadmaps.” Centre for Technology Management University of Cambridge, 2011.

[10] L. Hall, “Space Technology Roadmaps: The Future Brought To You By NASA.” Brian Dunbar, 07-Jun-2013.

[11] R. McFeeley, D. McKeehan, and T. Temple, *Software Process Improvement Roadmap*. Software Engineering Institute, 1995.

[12] A. Höss, C. Lampe, R. Panse, B. Ackermann, J. Naumann, and O. Jäkel, “First experiences with the implementation of the European standard EN 62304 on medical device software for the quality assurance of a radiotherapy unit.,” *Radiat. Oncol.*, vol. 9, p. 79, 2014.

[13] D. Flood, F. McCaffery, V. Casey, and G. Regan, “A Methodology for Software Process Improvement Roadmaps for Regulated Domains - Example with IEC 62366,” *Commun. Comput. Inf. Sci.*, vol. 364 CCIS, pp. 25–35, 2013.

[14] D. Flood, F. McCaffery, V. Casey, R. McKeever, and P. Rust, “A Roadmap to ISO 14971 Implementation,” *J. Softw. Process Evol. Accept. Awaiting Publ.*

[15] IEEE Standards Board, “IEEE 1058:1998 Standard for Software Project Management Plans,” 1998.

[16] S. McConnell, *Software Project Survival Guide*. Microsoft Press, 1997.

[17] M. McHugh, F. McCaffery, and V. Casey, “Standalone software as an active medical device,” in *Communications in Computer and Information Science*, 2011, vol. 155 CCIS, no. 2011, pp. 97–107.

11 Author CVs

Peter Rust

Peter Rust is a post graduate researcher in the Regulated Software Research Centre in DkIT. His PhD research addresses the difficulty experienced by organisations new to the medical device domain through the use of software process improvement roadmaps.

Dr Derek Flood

Dr. Derek Flood is a lecturer of computer science within the department of computing at Dundalk institute of Technology. Derek has previously worked as a post-doctoral research assistant at Oxford Brookes University where his principle research interests included usability of mobile applications and their implications on the performance of end users. In addition Dr. Flood has also examined the use of roadmaps for the implementation of medical device standards.

Dr Fergal McCaffery

Dr Fergal Mc Caffery is a Science Foundation Ireland (SFI) Principal Investigator. He is a Senior Lecturer in the Department of Computing and Mathematics, Dundalk Institute of Technology (DkIT). He is Director of the Regulated Software Research Centre in DkIT and the Medical Device Software Engineering competency area in Lero. He has been awarded SFI funding through the Stokes Lectureship, Principal Investigator and CSET Programmes to research the area of medical device software. He has published over 170 peer-reviewed conference and journal papers and is on the editorial board/programme committee for a number of leading software engineering conferences and journals.