

A Critical Evaluation of a Methodology for the Generation of Software Process Improvement Roadmaps

Derek Flood, Fergal Mc Caffery, Gilbert Regan, and Val Casey

{gilbert.regan,fergal.mccaffery,
kevin.mcdaid,derek.flood}@dkit.ie

Abstract. For medical device organisations to market their devices in specific geographic regions they must adhere to the regulations of that region. These regulations often recommend that organisations adhere to specific standards and guidance documents which specify “what” must be achieved without specifying “how” this may be done. Due to changes to the medical device directive, which governs the development of medical devices within the EU, in March 2010, software can now in its own right be considered a medical device. This change has meant that a number of software organisations developing software for the medical device domain must now adhere to the same regulations as other medical device manufacturers. In this work we present a concept for a Software Process Improvement (SPI) roadmap to guide such organisations through the task of implementing medical device standards and guidance documents. In addition we present and evaluate a methodology that can be used to create a SPI roadmap from a set of requirements such as the aforementioned standards and guidance documents.

Keywords: software process improvement roadmaps, medical device standards, software, usability, risk management, research method.

1 Introduction

Software can be easily used to configure a medical device without the need for expensive and time consuming hardware changes [1]. In 2006, Faris et al. [2] estimated that approximately half of all medical devices on the US market contained software. Complexity of software has been increased dramatically, posing higher risks of software malfunction and miss-application. Between 2005 and 2009, 87 models of infusion pumps were recalled due to safety problems [3]. In response to this, a whitepaper on the use of infusion pumps produced by the FDA, reports that “many of the problems that have been reported are related to software defects”.

Although this is only one example, recent trends show that an increasing number of medical devices are being recalled due to software failures. Due to the increasingly important role of software in these devices, software is now included in the EU’s definition of a medical device [4] subjecting it to the same processes and standards as other medical devices.

To ensure compliance, organisations are facing the challenge of implementing a number of medical device standards and guidance documents. These standards and guidance documents clearly define what must be achieved without providing specific methods for achieving them [5].

In this work we aim to alleviate this problem through the use of a series of software process improvement roadmaps. These roadmaps will not only outline what an organisation must do and when it should be introduced (in line with the software development lifecycle), but will also provide specific guidance on the best way to achieve these requirements for individual organisations.

Previous work by the authors [6] has outlined the structure of these roadmaps and proposed a methodology for their development. In this paper we aim to re-evaluate this methodology in light of its application to two medical device standards, IEC 62366[7] and ISO 14971[8], and to share our experiences in the application of the methodology to allow future researchers learn from these experiences.

The remainder of this paper is structured as follows: Section 2 outlines the background to this work and discusses the importance of software to the medical device domain. Section 3 examines the role of software processes and software process improvement within the medical device domain. In section 4 we describe the structure of the roadmaps and a methodology for their development. Section 5 then discusses how the roadmap development methodology was applied to two international standards and discusses the impact this will have on the methodology in the future. Section 6 then concludes the paper with an outline of how this work will progress.

2 Medical Device Regulations, Standards and Guidance Documents

In order to sell a medical device within the European Union (EU), the medical device organisation must demonstrate that they are compliant with the regulations set forth by the EU. Similarly, to sell medical devices within the US the organisation must demonstrate compliance with the FDA regulations [9]. In order to help organisations achieve compliance with these regulations, the EU and FDA have published standards and guidance documents that address specific aspects of the regulations and also recommend compliance with harmonised and consensus international standards, such as IEC 62304 [12] and ISO 13485 [10]. ISO 13485 Quality management system (QMS) ensures that the processes used during the development and production of a medical device are defined and monitored to ensure high quality products are developed. This standard is referred to by the European regulations and has recently been accepted by the FDA as adequate fulfilment of the requirements of a QMS.

As part of the QMS, organisations must perform risk management activities. ISO 14971:2007 [8] describes the requirements of a risk management process for medical device development. This standard identifies 6 key stages of a risk management; risk analysis, risk evaluation, risk control, evaluation of overall residual risk acceptability, risk management report, and production and post-production information.

During the development of a medical device, it is important to consider how the user will interact with it. Usability (the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use [11]) can be a source of great risk. The IEC 62366 [7] standard defines a usability engineering process that can help medical device developers produce usable products thereby reducing the risk of use errors.

IEC 62304:2006 – Medical device software – Software life cycle processes [12] provides specific guidance on how to perform software development activities for software that is to be incorporated in a medical device. It is therefore used to develop medical device software for both the European and US markets.

3 Software Process Improvement within the Medical Device Domain

There has been very limited adoption of software process improvement within the medical device domain [4]. In addition existing generic SPI models, such as the CMMI® and ISO/IEC 15504-5:2012 (SPICE), do not provide sufficient coverage to achieve medical device regulatory compliance [15] [2] [13] [1]. To address this issue a medical device specific SPI framework, titled Medi SPICE, is being developed [16].

The objective of undertaking a Medi SPICE assessment is to determine the state of a medical device organisation's software processes and practices, in relation to regulatory requirements and best practices with the goal of identifying areas for undertaking process improvement [15] [13]. It can also be used as part of the supplier selection process [17].

Medi SPICE is based on ISO/IEC 15504-5:2012 [14], IEC 62304:2006 [12] and ISO/IEC 12207:2008 [18]. It is being developed in line with the requirements of ISO/IEC 15504-2:2003 [19] and contains a Process Reference Model (PRM) and Process Assessment Model (PAM). It also incorporates the requirements of the relevant medical device regulations, standards, technical reports and guidance documents.

The Medi SPICE PRM consists of 44 processes and 15 sub-processes with clearly defined purpose and outcomes that must be accomplished to achieve that purpose. The Medi SPICE PAM which is related to the PRM, forms the basis for collecting evidence that may be used to provide a rating of process capability.

4 Software Process Improvement Roadmaps

In this work we propose the use of a software process improvement framework for the implementation of medical device standards. Unlike traditional SPI models, the goal of the roadmap implementation framework is not to improve existing processes but to implement the processes necessary to meet the requirements of a specific standard. Initially this work will focus on the development of SPI roadmaps for key medical device standards; IEC 62366, ISO 14971, ISO 13485 and IEC 62304.

For the purposes of this work we define a roadmap as: A series of milestones, comprised of goals, that will guide an organisation, through the use of specific activities, towards compliance with regulatory standards.

The roadmap is divided into two levels. The first level defines the goals, grouped into milestones that the organisation should achieve throughout the SPI initiative. And contains no detail relating to how the goals should be achieved. This is done for two reasons. Firstly, by presenting the roadmap as a series of goals traceability to the relevant standard can be easily achieved. Secondly, the high-level roadmap can form a basis for communication across the industry as the same high-level roadmap can be applied to all organisations. The second level roadmap contains specific guidance for organisations on how to achieve the goals outlined in the high level roadmap and is comprised of multiple activities that can achieve each goal so that the most suitable activity can be presented to an organisation wanting to implement the roadmap.

4.1 Roadmap Development Methodology

The following approach is similar to the transformation method presented in [20] for the construction of ISO/IEC 15504-2 compliant process assessment and process reference models. The goal of the transformation methodology presented in [20] was to develop a process reference model and a process assessment model. As the goal of this methodology is to develop a roadmap for implementing medical device standards it was necessary to alter the methodology to account for the order of implementation and the distinction between the goals and activities (or practices in ISO 15504) in the roadmap.

The methodology used for the development of the roadmaps is as follows:

1. **Identify requirements of the standard:** (The requirements will henceforth be known as ‘goals’ to differentiate the roadmap from the standard). This will be achieved through manual analysis of the standard.
2. **Logically group all goals.** Goals are grouped based on the stage of the software development lifecycle at which they occur. However as some goals are performed throughout the lifecycle, these goals should be grouped together and placed at or before the first stage at which they are performed.
3. **Separate grouped goals in line with ISO/IEC 15504 capability levels.** These groups are separated based on the capability level at which the requirements should be performed. These groups form the milestones of the roadmap.
4. **Order the milestones based on the capability level and logical groups.** All milestones containing level 1 goals should be implemented first in the order in which they will occur in the development process, followed by all milestones containing level 2 goals, and subsequently by all milestones containing level 3 goals until all of the milestones are in order.
5. **Validate generated roadmap.** The generated roadmap should be validated with industry experts. Members of the standards committee could also assist with the validation. Interviews or workshops are methods that could be used. A Delphi study could also be used. The validation should aim to ensure that:

- The goals are correctly grouped
 - The milestones are in the correct order for implementation
6. **Identify activities that can meet the identified goals.** This can be done through a systematic literature review and/or case studies with organisations already implementing the standard.
 7. **Validate activities in host organisation.** This will involve the generation of a roadmap for the host organisation and then undertaking a software process improvement initiative to implement the roadmap.

5 Evaluation

This section presents two case studies that have used the roadmap development methodology to develop and validate a high-level roadmap for two medical device standards (IEC 62366 and ISO 14971). A full description of the developed roadmaps is beyond the scope of this experience report.

5.1 Validation Methodology

To validate each of the roadmaps an expert evaluation was used. There were two aims established for each validation:

1. To determine if the goals are appropriately grouped into milestones
2. To determine if the ordering of the milestones is appropriate for implementation in a medical device organisation.

Experienced personnel within each of the two domains, risk management and usability of medical devices, were asked to complete the on-line questionnaire illustrated in Figure 1. The questionnaire showed participants each milestone in turn and asked them to state whether they thought each goal belonged in the milestone it was included in. In addition the participants were asked to rate on a 5 point likert scale (where 1 = strongly disagree and 5 = strongly agree) whether they agreed with the following statement; *The order of this milestone within the roadmap is correct.* The participants were also provided with the opportunity to add any additional comments they felt were relevant.

In addition to this, the online questionnaire also provides the user with the opportunity to state at what capability level, in line with 15504-2, each goal should be accomplished at. As the participants who took part in the study were experts in medical device standards and not software process improvement these results were not included in the study. However we did manage to recruit 1 software process improvement expert whose feedback is included.

Milestone 3: Risk Evaluation

Goal	Level <input type="checkbox"/>	Suitable <input type="checkbox"/>
Estimate associated risks or list possible consequences	Please Select ▾	<input type="radio"/> Yes <input type="radio"/> No <input checked="" type="radio"/> Don't Know
Decide if risk reduction is necessary for each hazardous situation	Please Select ▾	<input type="radio"/> Yes <input type="radio"/> No <input checked="" type="radio"/> Don't Know
Document results of risk evaluation	Please Select ▾	<input type="radio"/> Yes <input type="radio"/> No <input checked="" type="radio"/> Don't Know

For the following statements indicate how strongly you agree or disagree. (1 = strongly disagree; 5 = strongly agree)

The order of this milestone within the roadmap is correct

1 (Strongly Disagree)
 2
 3
 4
 5 (Strongly Agree)

Additional Comments

Match case

Fig. 1. Screenshot from online questionnaire

5.2 Case Study 1- IEC 62366

The first case study conducted applied the roadmap development methodology to the IEC 62366 standard. This standard outlines the requirements for a usability engineering process and describes what needs to be done to minimise use related risks. The standard requires the development of a number of documents, including a usability engineering file which should at least reference all of the documentation relating to the usability engineering process.

Step 1 of the methodology identified 44 goals, that were separated into 10 milestones (step 3) that are implemented throughout the software development lifecycle. Table 1 shows the number of goals that were included in each of the milestones for the IEC 62366 roadmap. It can be seen that the number of goals range from 2 to 7 per milestone.

Table 1. Initial IEC 62366 Roadmap

Milestone	# of goals	Milestone	# of goals
Task	5	Training	4
Usability Specification	5	Verification	4
Risk Management	7	Validation	4
Implementation	2	Validation Management	3
Documentation	6	Process	4

Once the roadmap was produced it was validated by 5 participants with experience of usability engineering for medical devices. In total 18 individuals were contacted in relation to the validation however, only 5 agreed to participate in the study. Overall the participants felt that the initial roadmap was well structured, however they did feel that the last two milestones should be implemented earlier. It was felt that the Process milestone should be the first milestone as it defines and maintains the overall process of usability engineering.

Although a full overview of the results is beyond the scope of this paper, it is important to mention that the results obtained in relation to the capability level of each goal provided little agreement among the participants. This result may be explained by the participants' area of expertise being in the area of usability engineering and not in ISO 15504 capability levels.

As a result of the validation, the roadmap was revised to include only 39 goals (some of the original goals were merged where the documentation of an activity and the activity itself were separate goals) divided into 9 milestones. Two milestones, validation and validation management, were merged to form a single goal as these were originally separated based on their capability level. Table 2 shows the number of goals by milestone for the revised roadmap.

Table 2. # of goals in the revised IEC 62366 roadmap by milestone

Milestone	# of goals	Milestone	# of goals
Process	3	Documentation	6
Task	3	Training	4
Usability specification	4	Verification	4
Risk Management	6	Validation	7
Implementation	2		

5.3 Case Study 2 – Roadmap for ISO14971

The second case study applied the roadmap development methodology to the ISO 14971 standard. ISO 14971 describes the risk management process that should be applied during the development of medical devices. The standard itself is not limited to software but can apply to any type of medical device. The standard outlines a 6 phase risk management process ranging from risk analysis, which is the identification of possible risks posed by the medical device to Production and post-production management of any residual risks.

The roadmap generated by the roadmap development methodology contained 51 goals divided among 14 milestones. Table 3 shows that there are between 1 and 7 goals per milestone in the roadmap. As risk management is an on-going activity with each stage being repeated throughout the software development lifecycle, the roadmap should be used to introduce the goals early in the product lifecycle so that the necessary activities are in place when needed.

Table 3. # of goals per milestone

Milestone	# of Goals	Milestone	# of Goals
Initial Planning	6	Pre-Production	1
Risk Analysis	4	Post-Production	2
Risk Evaluation	3	Management Planning	4
Risk Control	3	Staff Planning	4
Verification of Risk Control	5	Final Review	2
Residual Risk	7	Risk Management System Review	3
Pre-Release	6	Traceability	1

As was found in case study 1, the validation of the roadmap found that a number of the milestones, which contained goals believed to be at capability level 2 (Managed process- the process meets the requirements for capability level 1 where the process is *performed*, and is now implemented in a *managed* fashion), were introduced too late in the roadmap and as such should be introduced earlier. In addition it was also found that in a number of cases the separation of an activity from its documentation was unnecessary and these goals should be grouped into a single goal.

The validation resulted in a number of changes to the roadmap. The resulting roadmap contains 44 goals divided among 14 milestones. Three of the final 4 milestones were moved so that they would be implemented at the beginning of the implementation process. The final number of goals and order of the milestones can be seen in table 4 (read from top to bottom, then left to right).

Table 4. # of goals per milestone in the ISO 14971 Roadmap

Milestone	# of Goals	Milestone	# of Goals
Initial Planning	6	Verification of risk Control	3
Management Planning	3	Residual Risk	7
Staff Planning	3	Pre-Release	6
Traceability	1	Pre-Production	1
Risk Analysis	4	Post-Production	1
Risk Evaluation	3	Risk Management System Review	3
Risk Control	2	Final Review	1

6 Discussion

The case studies described above have provided a lot of insight into the methodology and have highlighted a number of issues that can arise when applying it to medical device standards.

The first stage is to determine the requirements of the standards. The standards contain a lot of supporting information which can be difficult to discern from the requirements. The authors judgement was used in determining this and the validation found that these judgements were correct. Additionally, it was found that the

standards use consistent terminology to describe what needs to be done in order to be in compliance with the standard.. This consistency could also allow for the use of Natural Language Processing (NLP) techniques to identify the requirements from the standard.

During the application of step 2 (logically group all goals) it was identified that not all processes can adhere to the software development lifecycle. In the cases outlined above it was found that there were logical groupings that could be easily identified however, some requirements would not easily fit into these groupings. For example in case study 2 it can be seen that 2 of the milestones contained only a single goal. Although the methodology itself can be quite flexible it was found in both case studies that dividing the goals based on their capability level provided little benefit as most of the goals should be implemented by a level 1 organisation. During case study 1, an ISO 15504-2 expert was asked to review the goals to determine if they were assigned an appropriate capability level. This expert remarked that as they are a requirement of the standard they should all be assigned a capability level of 1. For this reason it has been determined that this step (step 3) should be removed from the methodology.

In the cases outlined above it was found that the logical groupings provided a clear path to implementation. The use of the ISO 15504 capability levels in this step however, did cause problems. Some of the goals that were determined to be implemented at level 2 are necessary at the start of the implementation, for example the development of a standard operating procedure. Including these levels in the ordering of the milestones lead to the development of inaccurate roadmaps that was quickly identified by the experts. The case studies also revealed that it is important to select the correct method for validation. In both cases an online form was used to collect the opinions of the validators on the roadmap. Although this approach provided sufficient validation for the two standards selected, this may not scale well due to the large number of inputs that would need to be completed. In case study 2 one participant opted to email their comments directly to the author instead of completing the online form. To address this issue future validation studies may instead opt to take on a different format. One possibility is the use of a workshop whereby the participants are co-located and presented with the roadmap and provided an opportunity to discuss the roadmap in much more detail.

During the case studies it was found that the separation of the documentation of an activity from the activity itself, as was done in a number of cases, should not be done. One of the validators remarked that “If it’s not documented, it’s not done”, and suggested that the documentation of an activity should not be a separate goal but incorporated into the activity that is being performed.

7 Revised Roadmap Development Methodology

As a result of the validation the methodology now consists of:

1. **Identify requirements of the standard:** It is important to ensure that requirements of the standard are identified and distinguished from supporting advice. These requirements should then be phrased as goals of the roadmap.

2. **Logically group all goals.** Each group should not contain too many goals. If this is the case they may be separated into multiple sub-groups. The resulting groups will form the milestones for implementation.
3. **Order the milestones based on the logical groups.** The milestones should be ordered in a way that is compatible with implementation in a software development process to ensure that organisations suffer limited disruption due to the implementation.
4. **Validate generated roadmap.** The validation should be performed with industry experts and evaluate the roadmap to answer the following questions:
 - Are the goals appropriately grouped into milestones?
 - Is the ordering of the milestones appropriate for implementation in a medical device organisation?
5. **Identify activities that can meet the identified goals.** This may be done through a systematic literature review and/or case studies with organisations already implementing the standard.
6. **Validate activities in host organisation.** Generate a roadmap for the host organisation through collaboration between the organisation and industry experts and then undertaking a software process improvement initiative to implement the roadmap.

8 Limitations

The validation described above did not include development of the activities repository or industry validation of a complete roadmap. For this reason the validation presented above is limited to the first 5 steps of the methodology which can be used to develop a high-level roadmap. This in itself is a vital aspect to the development of software process improvement roadmaps.

The development methodology has been applied to two standards within the medical device domain. Before it can be established that the methodology can be applied as is to other domains, such as the automotive or aerospace domains, it must be validated within these domains.

9 Conclusions and Future Work

The implementation of any standard required in the development of medical devices can be a complex and time consuming issue. SPI roadmaps provide specific activities, in-line with the medical device standards, for an organisation to implement the chosen standard in a way that complements existing software development lifecycle processes.

This paper outlines how such roadmaps can be developed through the use of the roadmap development methodology and report on the application of this methodology to two medical device standards, IEC 62366 and ISO 4971. In light of these case studies it was deemed that the methodology can be used to develop high level software process improvement roadmaps that would be well received by the medical device community.

In the future this work will examine the use of Natural Language Processing (NLP) techniques to assist in the development of such roadmaps, in line with the roadmap development methodology. NLP techniques could be used to automatically identify the requirements of a medical device standard, the first step in the presented methodology. This could greatly simplify the process of roadmap development as the identification of such requirements can be a time consuming task.

Using the roadmaps presented here, it is intended to recruit a number of medical device organisations to implement the roadmaps to evaluate how well the proposed roadmaps work in an industry setting. After an initial evaluation of the organisations existing processes, a customised roadmap will be developed for the organisation and they will be guided through its implementation, until they have met all of the requirements of the medical device standard

Acknowledgement. This research is supported by the Science Foundation Ireland (SFI) Stokes Lectureship Programme, grant number 07/SK/I1299, the SFI Principal Investigator Programme, grant number 08/IN.1/I2030 (the funding of this project was awarded by Science Foundation Ireland under a co-funding initiative by the Irish Government and European Regional Development Fund), and supported in part by Lero - the Irish Software Engineering Research Centre (<http://www.lero.ie>) grant 10/CE/I1855.

References

- [1] Lee, I., Pappas, G., Cleaveland, R., Hatcliff, J., Krogh, B., Lee, P., Rubin, H., Sha, L.: High-Confidence Medical Device Software and Systems. *Computer* 39(4), 33–38 (2006)
- [2] Faris, T.H.: *Safe and Sound Software: Creating an Efficient and Effective Quality System for Software Medical Device Organizations*. ASQ Quality Press (2006)
- [3] Food and Drugs Administration (FDA), Infusion Pumps Improvement Initiative (2010), <http://www.fda.gov/medicaldevices/productsandmedicalprocedures/GeneralHospitalDevicesandSupplies/InfusionPumps/ucm205424.html> (accessed December 7, 2012)
- [4] McHugh, M., McCaffery, F., Casey, V.: Standalone Software as an Active Medical Device. In: O'Connor, R.V., Rout, T., McCaffery, F., Dorling, A. (eds.) *SPICE 2011*. CCIS, vol. 155, pp. 97–107. Springer, Heidelberg (2011)
- [5] McCaffery, F., Dorling, A., Casey, V.: Medi SPICE: An update. In: *International Conference on Software Process Improvement and Capability Determinations (SPICE)*, Pisa, Italy, May 18–20, pp. 195–198 (2010)
- [6] Flood, D., Mc Caffery, F., Casey, V., Regan, G.: A Methodology for Software Process Improvement Roadmaps for Regulated Domains – Example with ISO 62366. In: McCaffery, F., O'Connor, R.V., Messnarz, R. (eds.) *EuroSPI 2013*. CCIS, vol. 364, pp. 25–35. Springer, Heidelberg (2013)
- [7] IEC 62366:2007 “Medical Devices – Application of usability engineering to medical devices” Switzerland, ISO
- [8] ISO 14971 – Medical Devices – Application of risk management to medical devices, Switzerland, ISO (2007)

- [9] Burton, J., McCaffery, F., Richardson, I.: A risk management capability model for use in medical device companies. In: International Workshop on Software quality (WoSQ 2006). ACM, Shanghai (2006)
- [10] ISO 13485:2003, Medical devices — Quality management systems — Requirements for regulatory purposes, 2nd edn. Geneva, Switzerland, ISO (2003)
- [11] ISO/IEC, ISO 9241: Ergonomic requirements for office work with visual display terminals (VDT)s - Part 11 Guidance on usability (1998)
- [12] IEC 62304:2006, Medical device software—Software life cycle processes. Geneva, Switzerland, IEC (2006)
- [13] Humphrey, W.S., Snyder, T.R., Willis, R.R.: Software process improvement at Hughes Aircraft. IEEE Software 8(4), 11–23 (1991)
- [14] ISO/IEC 15504-5:2012, Information technology - Process Assessment - Part 5: An Exemplar Process Assessment Model. Geneva, Switzerland, ISO (2012)
- [15] McCaffery, F., Dorling, A.: Medi SPICE Development. Software Process Maintenance and Evolution: Improvement and Practice Journal 22, 255–268 (2010)
- [16] Casey, V., McCaffery, F.: Development of the medi SPICE PRM. In: Mas, A., Mesquida, A., Rout, T., O'Connor, R.V., Dorling, A. (eds.) SPICE 2012. CCIS, vol. 290, pp. 265–268. Springer, Heidelberg (2012)
- [17] Casey, V.: Virtual Software Team Project Management. Journal of the Brazilian Computer Society 16, 83–96 (2010)
- [18] ISO/IEC 12207:2008, Systems and software engineering - Software life cycle processes. Geneva, Switzerland, ISO (2008)
- [19] ISO/IEC 15504-2 (2003) - Software engineering — Process assessment — Part 2: Performing an assessment, Geneva, Switzerland (2003)
- [20] Barafort, B., Renault, A., Picard, M., Cortina, S.: A transformation process for building PRMs and PAMs based on a collection of requirements – Example with ISO/IEC 20000. In: SPICE 2008, Nuremberg, Germany (2008)