**A Software Process Development,**

**Assessment and Improvement Framework,**

**for the Medical Device Industry**

*F. McCaffery2, P. Donnelly1, A. Dorling3, F.G. Wilkie2*

*1IQ Solutions, Castlewellan, Northern Ireland.*

*2Centre for Software Process Technologies, Jordanstown, Northern Ireland.*

*3InterSPICE, Cambridge, England.*

Email: *f.mccaffery@ulster.ac.uk*, *peter@iqsolution.com*, *alec.dorling@interspice.uk.com*, *fg.wilkie@ulster.ac.uk*

**Abstract**

*This paper describes a software process development, assessment and improvement framework, structured to ensure regulatory compliance for the software developed in medical devices. Software is becoming an increasingly important aspect of medical devices and medical device regulation. Medical devices can only be marketed if compliance and approval from the appropriate regulatory bodies of the Food and Drug Administration (US requirement), and the European Commission under its Medical Device Directives (CE marking requirement) is achieved.*

 *Integrated into the design process of medical devices, is the requirement of the production and maintenance of a device technical file, incorporating a design history file. Design history illustrates the well documented, defined and controlled processes and outputs, undertaken in the development of medical devices and for our particular consideration with this framework - the software components.*

**1. Introduction**

 Of increased interest to medical device companies, particularly small businesses, is to enhance their software development processes beyond the sole need to ensure regulatory compliance, into an environment that integrates compliance, best practice and process improvement.

Both patients and medical professionals have to rely on the proper functioning of medical devices that need to be operated safely. One of the main current issues is the assessment of appropriate safety integrity levels (SILs) for medical devices.

The software framework introduced in this paper will address an opportunity to integrate the regulatory issues, process improvement mechanisms and safety integrity in order to achieve greater customer satisfaction, faster time to market and improved software quality.

**2. The medical device industry**

According to the Institute of Medicine report ‘To Err is Human’[11], between 4400 to 9800 people die in hospital from preventative medical errors. The report also says that more people die every years a result of medical errors than from motor vehicle accidents, breast cancer or AIDS.

Like most industries, the healthcare industry depends on computer technology to perform many of the functions ranging from financial management and patient information to patient treatment[12].

The use of software in medical devices has become widespread in the last two decades. Medical devices with software include those that are supplied and used entirely in hospitals and other health facilities, as well as consumer items such as blood pressure monitors. Many medical devices, and their software, operate in real time – monitoring, diagnosing, or controlling a physiological process as it changes.

The complexity and risk profile of medical devices vary widely and range from a consumer digital thermometer for minor diagnosis, and an implanted artificial heart that is critical to preserving a patient’s life, to a therapeutic X-ray machine with a computer user interface, programmable software controlled therapy and anatomical and biophysical modelling in the software, which is operated under a high level of professional staff supervision[13].

A recent analysis of medical device recalls highlights the diverse nature of medical device software failures.The Federal Drug Administration (FDA) found that during 1983 – 1987 approximately 44% of the quality problems that led to voluntary recalls of medical devices were attributed to errors or deficiencies designed into particular medical devices rather than having been inserted during the manufacturing phase. The study also recognised software quality management practices as a means to prevent failure[14].

In the medical device industry, the software used to control a device takes on an additional role - it must help ensure the safety of the user and the patient. There are many challenges to implementing safe software. Software design needs to include deliberate engineering practices and rigorous risk analysis and mitigation needs to be performed whilst at the same time simultaneously addressing potential device failures that may be introduced by the software itself[15].

**3. Background**

Medical device manufacturers, like businesses in most industries, have based the core of their quality system on the ISO9000:2000[20] family of standards. The internal quality system of each medical device company has then been enhanced to ensure compliance with additional standards that relate specifically to the nature of their products and the market they address. ISO 13485:2003, Medical devices - Quality management systems - Requirements for regulatory purposes[29], for instance, is based on quality management system requirements currently contained in medical device regulations as well as those appropriate requirements contained in ISO 9001:2000.

A recent study carried out by IQ Solutions with the medical device companies in N. Ireland has revealed that the software development process has been predominately based on the need to comply with the US Food and Drug Administration (FDA) [1] and European Medical Device Directives (MDD) [2] regulations. The software life cycle process itself has not been of primary focus, only that the elements are in place that satisfy the regulatory requirements.

It is believed that a Software Development Roadmap (SDR) and Software Process Improvement (SPI) exercise would greatly enhance the design control procedures currently being implemented in company quality systems. The major rationale for accepting the software framework is to improve software quality and reduce time to market. Software quality is defined as software that meets its functional and non-functional requirements without lengthy rework, including regulatory compliance, without any inconsistencies. Reducing time to market is often based on continuous improvement of processes implemented for the design, development and manufacture of medical devices and products.

In addition, a recent survey undertaken by the Centre for Software Process Technologies (CSPT), indicated that N. Ireland health technology companies have limited awareness of standards applied to software development, such as ISO/IEC15504 (SPICE)[3]. Few organisations are aware of the concept of software process improvement and more alarmingly, the importance of software process improvement for increasing market penetration worldwide, particularly in the U.S.A. If this is not addressed, health technology companies will experience difficulties in competing in markets where advanced software systems are required and time to market is crucial.

Interestingly the prime factor here is not simply cost, but the quality agenda as a whole, with competitor companies possibly having already engaged in process improvement programs. Indeed, in countries such as China[4] there is considerable recognition of the importance of quality in software development with the Chinese government not only setting up government agencies with responsibility for building the maturity of the Chinese software industry but also offering subsidies to software enterprises that engage in SPIbased maturity evaluations.

**4. SPI framework**

IQ Solutions and the CSPT are developing a software development framework for the medical device sector that integrates existing regulatory requirements for the control of the design, development, maintenance and support of software, with a software process improvement mechanism that will improve productivity, time to market and quality. The framework is developed and delivered within the design control component of the ISO9000 standard and related standards, thereby not requiring additional standard compliance and specific auditing by a notified body.

The framework has received interest from other sources. There is a specific interest to address software in the medical device sector currently coming from Sweden, Italy, Germany, Switzerland and the USA. To channel this interest the SPICE User Group is to launch the MediSPICE initiative to bring the interested parties together.

The approach for delivering an SPI framework is to establish a model (implemented as illustrated in **figure 1**) that addresses the specific software development requirements from relevant regulatory bodies, standards, customer expectations, and integrates those constraints with a SPI framework. The proposed SPI framework will be based on identifiable and relevant components of existing SPI models, such as ISO/IEC15504 (SPICE) and CMMI[5]. The model will be flexible in that relevant elements of the SPI framework may be adopted as required to provide the most significant benefit to the business.

The major process improvement frameworks that currently exist, namely ISO/IEC15004 (SPICE) and CMMI, do not address regulatory requirements of the medical device industry. The US Food and Drug Administration (FDA) and the European Medical Device Directives (MDD) are the regulatory standards that control the design, development, manufacturing and support of products within the medical device industry.

Figure 1: Software framework model

M

O

D

E

L

SPI

Customer

Regulatory bodies

Standards

**Best practices**

Regulatory compliance

Quality

Customer

satisfaction

Time to market

**5. Project Outline**

In order to deliver an endorsed framework it was essential that a steering group was formed with members from various medical device companies and a notified body with experience in auditing medical device companies. The involvement of medical device companies also adds an ownership element to the model and should improve its acceptance and implementation within each company.

The software development model will cover the complete lifecycle, defined by default, as the V-Model in

Figure **2**. No restriction will be made on the development life cycle processes undertaken by individual companies, although it is understood that companies within the medical device sector typically implement this V-Model. The model will be expanded to incorporate applicable aspects of software improvement programs such as ISO/IEC15504 (SPICE) in addition to checking that all applicable standards and regulations are adhered to in a seamless manner and are focused upon software process improvement.

The framework will also ensure that developed, reused or off-the-shelf[6] software satisfies functional, operational and regulatory requirements throughout the system lifecycle. As each stage of the life cycle can be performed using various methodologies and project management processes, the model will be flexible to support the implementation and integration of preferred methodologies into the framework.



**Figure 2: Software lifecycle V model**

**6. Design**

The framework will provide a software development roadmap, which complies with the regulatory guidance criteria, while introducing best practice techniques and methodologies that can be selected as required. For example a risk management process can adopt a Failure Mode and Effects Analysis (FMEA) method of identifying, mitigating and tracking risk and hazard issues. Also the continuous management of verification and validation testing can be controlled and planned under a structured Test Process Improvement (TPI) model.

The framework will provide a means of assessing software engineering capability in twelve areas that have been defined by the FDA[7]as:

1. Level of Concern
2. Software Description
3. Device Hazard and Risk Analysis
4. Software Requirements Specification
5. Architecture Design
6. Design Specifications
7. Requirements Traceability Analysis
8. Development
9. Validation, Verification and Testing[8]
10. Revision Level History
11. Unresolved Anomalies
12. Release Version Number

To ensure compliance whilst also achieving software process improvement the following six software process have been identified as areas to be focused upon as a mapping can be made from these process areas to the twelve areas listed above.

1. Requirements Management
2. Risk Management
3. Project Management
4. Configuration Management
5. Verification, Validation and Testing
6. Software Design and Development

The model will help companies to measure their organisational capability and to track progression and achievements in each of the six process areas and against process capability levels defined in ISO/IEC 15504-2[16].

**7. Assessment framework**

The assessment framework will provide a method for the assessment of software processes in medical device organisations. The framework will include an assessment model that will provide a structured approach for assessment and that may be used by organisations involved in the development, maintenance and support of software with the following purposes:

* Understanding the state of a company’s own processes for process improvement;
* Determining the capability of a company’s own processes for a particular contract;
* Determining the capability of another organisation’s processes for a particular contract.

The assessment model will include coverage of all associated medical device regulatory standards. The project will take a similar lead in the approach to defining the assessment model based on SPICE for Space (S4S) in the space sector, and AutomotiveSPICE in the automotive sector that have adapted the ISO/IEC 15504 assessment model to applicable domain standards. Additionally, the outcome will have similarities to the Capability Assessment Tool[10] that was developed by the Office of Government Commerce, in that it helps an organisation to measure organisational capability and to track progression and achievement in key process areas (with a mapping back to the twelve FDA key areas).

The framework will not provide requirements but rather guidelines and recommendations and ensure it endorses the essence of the SPICE and CMMI frameworks.

**7.1. Contents of the framework**

The contents of the framework will include a description of thefollowing:

* Introduction to the framework;
* Intended audience and implementation strategy;
* Software Development Roadmap - a suggested step-by-step guide to developing software including:
	+ Life cycle development models and milestones,
	+ Roadmap checklists,
	+ Recommended best practices;
* Architecture consisting of:
	+ An assessment model,
	+ An assessment method,
	+ Assessment activities;
* Process assessment and improvement concepts
	+ Process improvement,
	+ Process capability determination,
* Compliance with medical device regulations;

**7.2. Software development roadmap**

Of key importance to medical device companies is a model that identifies the major software development activities and indicates how they should be undertaken not only to achieve complete, efficient and timely software development projects, but that also provides complete regulatory compliance and operates within recommended guidelines.

The roadmap will be based on the V-Model as illustrated in **figure 2**, which is commonly accepted as the default product development model and is recommended under FDA guidelines[7].

It is the intention of the framework not only to encourage the implementation of a process model outlining specific development phases and how they interact, but also to examine how these phases are undertaken for their content, completeness and coverage. In this circumstance best practices from the software industry will be presented and checklists will be provided.

* 1. **Process Assessment**

Process assessment is an integral part of software process improvement and (see **Figure 3**)[9] provides a way to measure the capability of selected processes in an organisation against a target capability profile. Analysis of the assessment results enables companies to prioritise which processes should be improved in order to increase their effectiveness in achieving their business goals. The assessment results will also indicate the risks involved in undergoing a project using the assessed processes. This enables determination of how effective they are in achieving their goals, and to identify significant causes of poor quality, or overruns in time or cost. These provide the criteria to prioritise process improvements.

Conformance assessments will be performed with the objective of determining how a company’s processes adhere to the framework. The assessments will be based on the interpretation of both the company’s historical business and product development information and metrics, and on the rating scheme used in the process assessment. When performing a conformance assessment any requirements that are specified by the regulatory bodies (e.g. FDA regulations) will identified as mandatory.

is

examined

by

Process

Process

Assessment

Process

Improvements

Capability

Determination

Conformance

Assessment

leads

to

leads

to

motivates

motivates

leads

to

identifies

capability

and risks

identifies

changes to

identifies

capability

and risks

**Figure 3: Objectives of Process Assessment**

**7.3.1. Process assessment model.** The process assessment model is composed of two main components:

* Process areas
* Capability scale.

The process area component is introduced by listing the process areas that are deemed applicable to the medical device industry. A description of each process area is complemented with details of how to fulfil the criteria of the process area. In addition a list of potential work outputs are represented together with their corresponding term in associated medical device standards.

The capability scale section is introduced by listing the capability scale against which each process is measured. The capability scale will be based upon ISO/IEC 15504-2 capability scales.

**7.3.2. Process Assessment activities.** A medical device software process assessment will consist of the following activities:

* Initiation;
* Planning;
* Briefing;
* Information and metrics acquisition;
* Information and metrics validation;
* Process rating;
* Reporting and inputs to the risk management process.

A detailed breakdown of each activity into tasks will be described in the framework. Each process included in the assessment will be assessed on the basis of available evidence.

The evidence may be gathered using interview, inspecting organisational documents or analysing metrics. The information collected for each process area will meet assessment objectives and scope. Information that supports a particular process rating will be recorded and maintained as evidence to substantiate the ratings and to verify compliance with the requirements.

**7.3.3. Risk management.** The framework will include a last important step “Reporting and providing inputs to the Risk Management Process”. The risk management process area is both a process area and the final step of the assessment activities, as it is important that risk management is assigned a separate process area and that issues found in other process areas are also reported into the risk management process.This is particularly important to software developed or implemented in the medical device industry, as it is necessary to implement a risk management strategy throughout development. The risk management guidelines within the framework will acknowledge and endorse the ISO14971:2000 standard for “Application of Risk Management to Medical Devices”.

In addition to ISO14971, best practice models such as FMEA for medical devices will be introduced to identify, assess, reduce, accept, and control software risks in a systematic proactive, comprehensive and cost effective manner taking into account the project’s technical and programmatic constraints. Risk identification within the framework will consist of 5 stages

Stage 1 – Creation of a forum for risk assessment.

Stage 2 – Systematically identify risk.

The Software Risk Taxonomy checklist (that includes checklists relating to the regulatory standards and the SEI Software Risk Taxonomy checklist[10]) is used as a tool to help identify risks.

Stage 3 –Define the Risk Attributes

Identified risks are clarified and major risk attributes such as the probability of occurrence, the loss to the project should it occur, and the time frame for action are defined.

Stage 4 – Document the identified risk in a risk management form.

Stage 5 – Communicate the identified risks to the stakeholders.

Risks are prioritised according to their exposure (the probability of the risk occurring multiplied by the impact of the occurrence of this risk on the project). This enables project critical issues to be highlighted to management and if necessary to the steering group committee.

The risk management process benefits from the inputs from the process assessment in aspects related to the identification of processes triggered by a project risk and an evaluation of potential risk.

**7.4. Software Process Improvement**

Software process improvement is a continuous process that an organisation follows in a cyclic improvement path[9]. Improvement will be achieved by following this path and adopting specific improvement measures such as the introduction of new or changed practices into established processes and removing inefficient practices. An important step within the software process improvement cycle is the gathering of information. This information is required to establish the current state and subsequently to confirm the improvements by comparing the initial process assessment results with the re-assessment results gathered after the implementation of the improvements. The degree of improvement depends on the assessment criteria used rather than the specific activities of the improvement process.

The framework will provide guidelines for the implementation of each of the activities in the improvement cycle as well as the expected results. Additionally, guidelines will to be provided in terms of recognition of improvement results.

**8. Safety Integrity levels**

Patients and medical professionals have to rely on the proper functioning of medical devices that need to be operated safely. The different stakeholders in the medical domain need to assess user needs, identify context specific risks, and specify functional requirements within a variety of operational contexts[17].

One of the main current issues is the assessment of appropriate safety integrity levels (SILs) for medical devices. The topic of SILs requires special consideration with regard to the weighting of expected benefits of therapeutic or diagnostic functions against risks for patients in different classes of initial health condition.

Active medical devices are more and more dependent on and controlled by programmable electrical/electronic (PE) systems. IEC 61508[18] is an international standard for functional safety of electrical and/or electronic and/or programmable electronic (E/E/PE) safety related systems. IEC 61508 defines appropriate means for achieving functional safety in the system. Part 3 of the7-part standard concerns itself with Software Requirements.

The range of E/E/PE safety-related systems to which IEC 61508 can be applied includes other areas in addition to medical devices for instance:

* Emergency shut down systems;
* Crane automatic safe-load indicators;
* Guard interlocking and emergency stopping systems for machinery;
* Railway signalling systems;
* Fly-by-wire operation of aircraft flight control surfaces;
* Automobile anti-lock braking systems;
* Remote monitoring, operation or programming of network-enabled process plant.

As a general standard for safety, the IEC 61508 standard defines four Safety Integrity Levels (SIL). The levels are specified in terms of average probability of failures per usage for low demand systems and probability of failures per hour for high demand (continuous usage) systems.

IEC 61508 recommends the usage of a number of software practices for safety critical systems. These recommendations become more stringent as the required safety integrity level increases. The recommendations of the IEC 61508 standard focus on two productivity related parameters - one is termed software product verifiability and the other software process capability.

Part 3 of the IEC 61508 standard relates to software requirements in safety critical systems. The general requirements of Part 3 specify the employment of such processes as quality assurance, configuration management, documentation and planned and tracked life cycle. Part 3 also enumerates considerable number of recommended software practices.

When performing a software process assessment it will increasingly become important to establish the relevant target capability level for a SIL. There is currently a wide ranging set of views that surround the ability to establish the relationship between capability / maturity levels and software criticality. There are several opposing opinions and approaches to the problem that need to be analyzed and validated before including in the assessment framework for medical devices.

*Some of the wide ranging views are detailed below:*

In using IEC 61508 SILs within the automotive sector, the MISRA guidelines [25,26] have made a basic assumption that an organization must have an ISO9001 software quality system at SIL level 0 (lowest level). If this assumption is true then the starting point for any target levels would be capability level 3 for all processes encompassed within an ISO9001 quality system.

In relating IEC 61508 SILs to ISO/IEC TR 15504 Capability Levels empirical work from the SPICE trails and COCOMO II[19] has proffered a correspondence between SILs and CLs. The stated correspondence relates SILs 1-2 to CL2 and SILs 3-4 to CLs 3-4 respectively for relevant engineering processes. Data collected from the SPICE trials however indicates that organizations do not live up to the required capability levels for safety critical applications.

The literature provides many examples of practices, tools and techniques that are recommended for the different SIL levels[20,21,27]. The question then arises whether the relationship between SILs and CLs are two dimensional or three dimensional. Whilst there may be a relationship of practices to capability levels, there may also be a need to assess on a more rigorous basis work product characteristics and resource and infrastructure characteristics at all levels depending on the SIL.

Sources at the Defense Systems Directorate (USA) and Defence Material Organization (Australia) state that there is no correlation between CMMI/CMM capability/maturity levels and integrity-levels / levels-of-assurance. Achieving a certain CMMI/CMM capability / maturity level does not guarantee the ability to develop high integrity systems. It is stated that it is easy to confuse between them if you are not familiar with CMMI and/or safety.

Current initiatives within CMMI and iCMM are integrating an integrity assurance process area within the models to deliver a context for safety/security and the synthesizing and harmonizing of safety/security practices[23]. A similar viewpoint has been developed by the +SAFE initiative in Australia[24].

There is also a raging debate in UK safety critical circles as to whether there is actually any evidence that the ‘accepted wisdom’ defined in the standards is effective?[22]. It has been suggested that an ‘evidence based approach’ may be better than ‘accepted wisdom’ in process based standards. The standards themselves contain a lot of sensible advice and guidance albeit that not much about safety, only quality and reliability.

**9. Conclusion**

The medical device industry in Northern Ireland is characterised by many small new-start indigenous companies. Of particular importance to these companies is the need to develop medical devices in full compliance with the appropriate regulatory bodies that govern the sale and marketing of medical devices throughout the world. The key business goals of - cost effective development and speed to market, are fundamental factors for all companies, but for small new-start companies this is critical. The studies and assessment of the local medical device industry illustrates that an assessment framework as proposed in this paper will provide a huge benefit to participating companies as business goals and regulatory compliance may both be achieved. Once the framework is established further work will need to be progressed to relate capability levels to safety integrity levels to address relationships between the relevant standards.

**10. References**

1. FDA Regulations. "Code of Federal Regulations 21 CFR Part 820." June 1997.
2. European Council, “Council Directive 93/42/EEC Concerning Medical Devices”, 14 June 1993.
3. ISO/IEC TR 15504:1998(E), Information Technology – Software Process Assessment, Parts 1-9, Type 2 Technical report.
4. Ju Dehua., China’s Budding Software Industry, IEEE Software May/June 2001. pp 92-95
5. Capability Maturity Model® Integration (CMMISM) for Software Engineering (CMMI-SW, V1.1, Version 1.1, August 2002)
6. FDA/CDRH Guidance Document. "Guidance for Off-the-Shelf Software Use in Medical Devices." September 1999
7. FDA/CDRH Guidance Document. "Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices." May 1998.
8. FDA/CDRH Guidance Document. "General Principles of Software Validation." June 1997.
9. ECSS-Q-80-02, A standard for software process assessment and improvement for the European Space Domain: Patricia Rodriguez-Dapena, Joint ESA-3rd International Spice Conference on Process Assessment and Improvement, 17-21st March 2003, Noordwijk, pp 7-14.
10. Capability Assessment Tool. PSBS News Summer 2002, pp 12-13.Carr M, Konda S, Monarch I, Ulrich F, Walker C. Taxonomy based identification. Technical report CMU/SEI-93-TR-6. Pittsburgh, PA: SEI, Carnegie Mellon University, 1993.
11. To Err is Human: Buiding a Safer Health System, Linda Kohn, Janet Corrigan, Molla Donaldson, eds, National Academy Press 2000.
12. Failure Modes in Medical Device Software; An analysis of 15 Years of Recall data, Dollores R. Wallace and D. Richard Kuhn, NIST.
13. Regulation of medical devices involving software in Australia – an overview, John Jamieson, 6th Australian Workshop on Safety Critical Systems and Software, Brisbane 2001.
14. Applying requirements management to medical devices utilizing software, Dean A. Leffingwell, Don R. Widrig and Wayne T. Morrissey, Rational Software Corporation 1997.
15. Developing safety-conscious software for medical devices, Timothy Cuff and Steven Nelson, Medical Devicelink archive January 2003..
16. ISO/IEC 15504-2 Process Assessment: Requirements, ISO 2003.
17. Medical devices subgroup briefing paper, EWICS Medical devices subgroup [www.ewics.org](http://www.ewics.org)
18. “IEC 61508 Functional safety of electrical/electronic/programmable electronic safety-related system. Parts 1-7”, International Electro-technical Commission. IEC 61508. 1999
19. O. Benediktsson; *Safety critical software and development productivity*; The Second World Congress on Software Quality, Yokohama, Sept 25.-29, 2000. <http://www.calpoly.edu/~pmcquaid/2WCSQ>
20. Safety in the Process Industries – Guiding Principles to the implementation of IEC 61508 [www.sipi61508.com](http://www.sipi61508.com)
21. The application of IEC 61508 in the Automotive Sector - Gunter Glöe, Folkert Jurgens, Gerhard Rabe – TUV Nord Gruppe, Hamburg / Hannover , 5th International Symposium Programmable Electronic Systems in Safety Related Applications, 7-8 May 2002
22. Software Safety: Where’s the Evidence? – John A McDermid, Proc. Sixth Australian Workshop on Industrial Experience with Safety Critical Systems and Software, Brisbane, Australia. Conferences in Research and Practice in Information Technology, 3. Lindsay, P., Ed. ACS. 1-6. <http://crpit.com/Vol3.html>
23. Integrity Assurance: Integrating Safety and Security with use of CMMi – Joe Jarzombek, Defense Systems Directorate, April 2003. <http://www.incose.org/so-md/ChapterMeetingMinutes/Presentations/Microsoft%20PowerPoint%20-%20NavAirCMMI-IntegrityAssurance.pdf>
24. Use of +SAFE in acquisition: Lessons learned in Australian Defense Material Organisation – Matt Ashford, Software Intensive Systems Acquisition Resources and Analysis Directorate, Office of the Secretary of Defense
25. MISRA "Development Guidelines for Vehicle Based Software", ISBN 0 9524156 0 7, November 1994. <http://www.misra.org.uk/index.htm>
26. MISRA Report 2, "Integrity", February 1995. <http://www.misra.org.uk/index.htm>
27. Relating SILs to CLs – Research work in progress at Daimler Chrysler Research Laboratories (unpublished work – not in the public domain) AutomotiveSIG, The Procurement Forum, March 2003.
28. ISO 9000:2000 Quality management systems (series). [www.iso.ch](http://www.iso.ch)
29. ISO 13485:2003 Medical devices -- Quality management systems -- Requirements for regulatory purposes. [www.iso.ch](http://www.iso.ch)