Are

We

IEC62304

Compliant?

- Wat is professionele sw
  - upto Date plan
  - . reqs managemeten
  - Code reviews
  - Code check ers
  - Code coverage check ers
  - Architectuur
  - If spec
  - Tracing
  - Sw versioning and baselining
  - Crpr ccb
  - Risk management
  - Soup management

- Single fault condition on HW?
- Voorbeeld TENS beter uitwerken
- FDA sheet erbij
- Clarify standard is not mandatory

Paper summary

- **Is your medical device software compliant with the current medical device standards**
- With the launch of the standard for medical device software, the IEC62304, you as manufacturer have to adhere to a range of rules before you can release it to the market. Regulators have become stricter over the years especially on the software field, since software becomes more complex and the amount of software code in medical devices is growing.
- Are you as manufacturer aware of your obligation to be compliant with the standard on software the IEC62304?
- Do you know how to implement this standard in your software development process efficiently and effectively?
- If you can answer both questions with “yes”, skip this presentation, if you cannot please join.

The latest version of this presentation

- Is to be found on my website www.processvision.nl

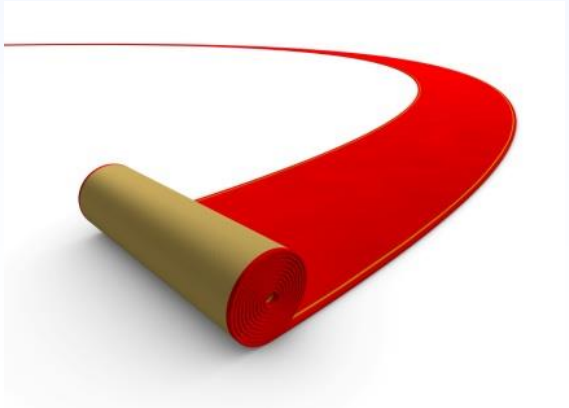
Willem van den Biggelaar

- Quality and Medical Regulatory Consultant
- Certified DEKRA auditor for ISO 9001 / ISO 13485
- Setup several ISO 9001 / 13485 certified Quality Management Systems (QMS)
- Previous jobs
  - Quality Assurance Officer (5 years)
  - System Tester (1 year)
  - Embedded software engineer (7 years)

And what is your experience with IEC62304

Content

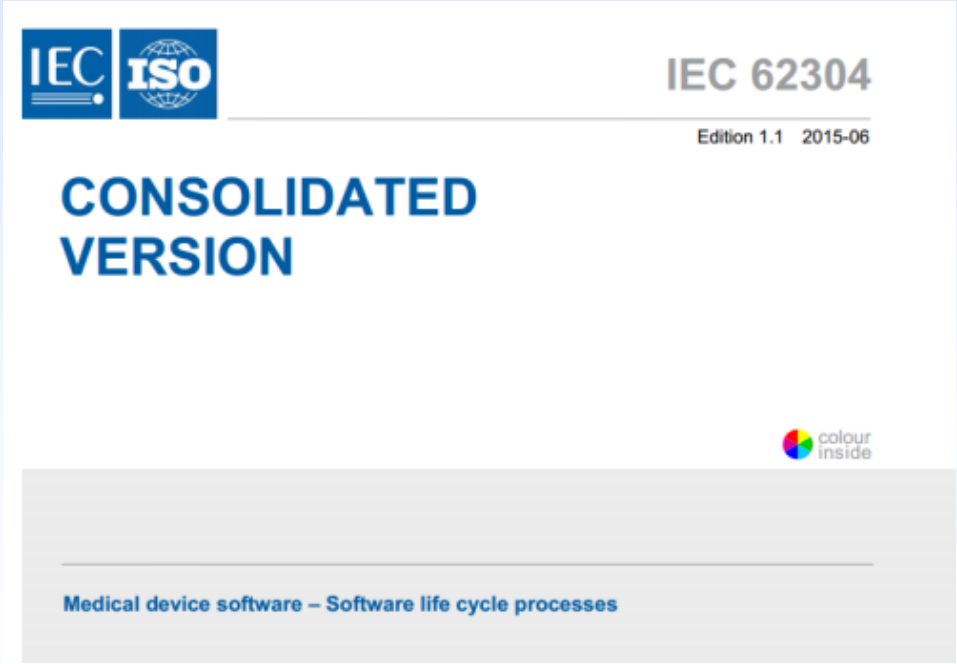
- What is the IEC62304
- Software Classification
- Implementation of the IEC62304



WHAT IS THE IEC62304

Just to set the scope…….IEC62304:2006 v1.0

- Is the de-facto process standard for the development of medical device software
- New 2.0 version on it's way, version 1.1 already available



Accepted for Europe, USA, China and several other countries

Recognized Consensus Standards

FDA Home ▸ Medical Devices ▸ Databases

1 to 2 of 2 Results
Standards Designation Number: *62304*

Results per Page 10

New Search | Export To Excel | Help

Recognition Number	Standard Developing Organization	Standard Designation Number And Date	Title Of Standard	FR Publication Date	Specialty Task Group Area
13-32	AAMI ANSI IEC	62304:2006	Medical Device Software - Software Life Cycle Processes	08/20/2012	Software/Informatics
13-79	IEC	62304 Edition 1.1 2015-06	Medical Device Software - Software Life Cycle Processes	04/04/2016	Software/Informatics

Cenelec	EN 62304:2006 Medical device software - Software life-cycle processes	27/11/2008		
	EN 62304:2006/AC:2008	18/01/2011		

Code of China

Chinese Standard Classification | Professional Classification | ICS Classification | Latest

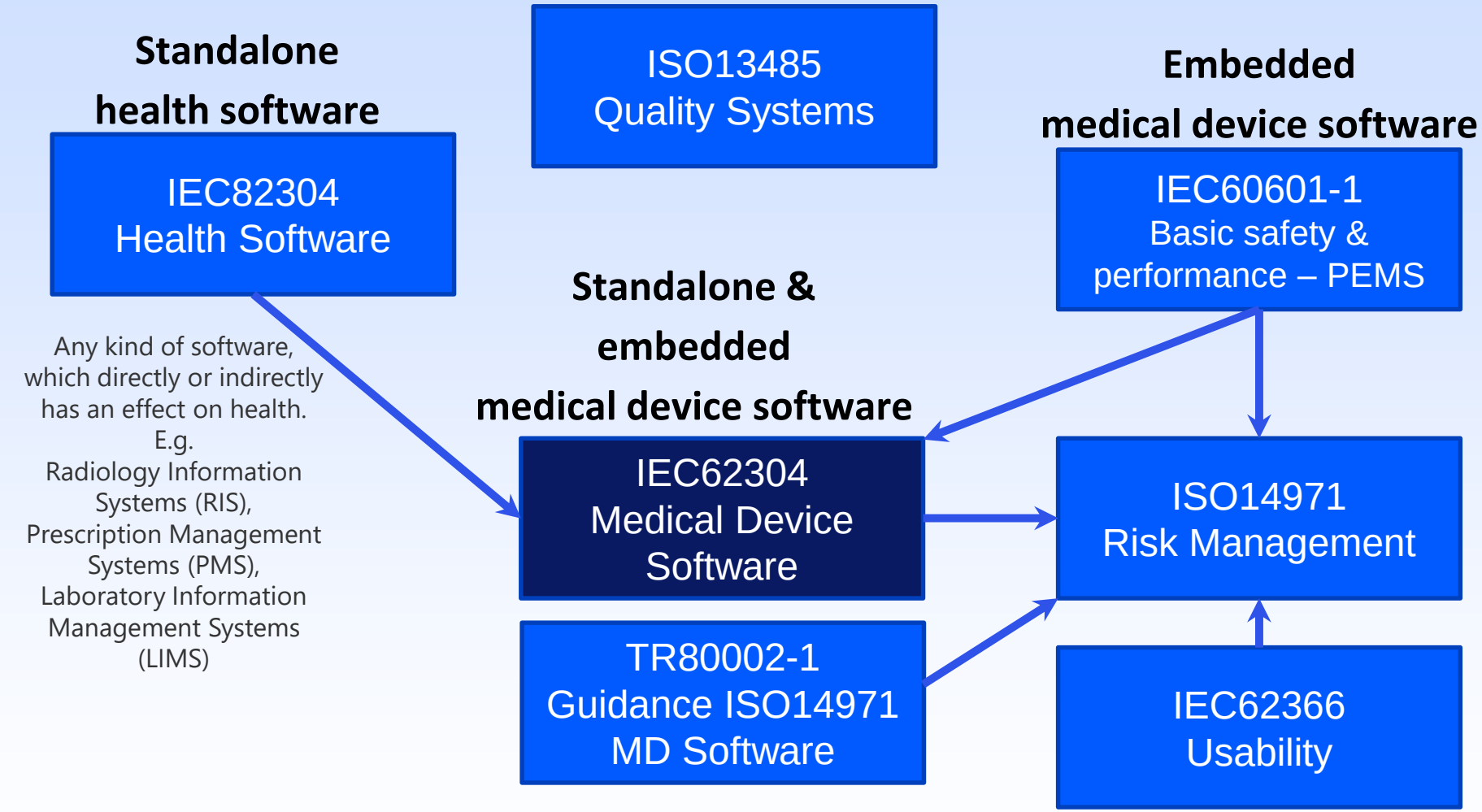
Search | Advanced Search

Position: Standard

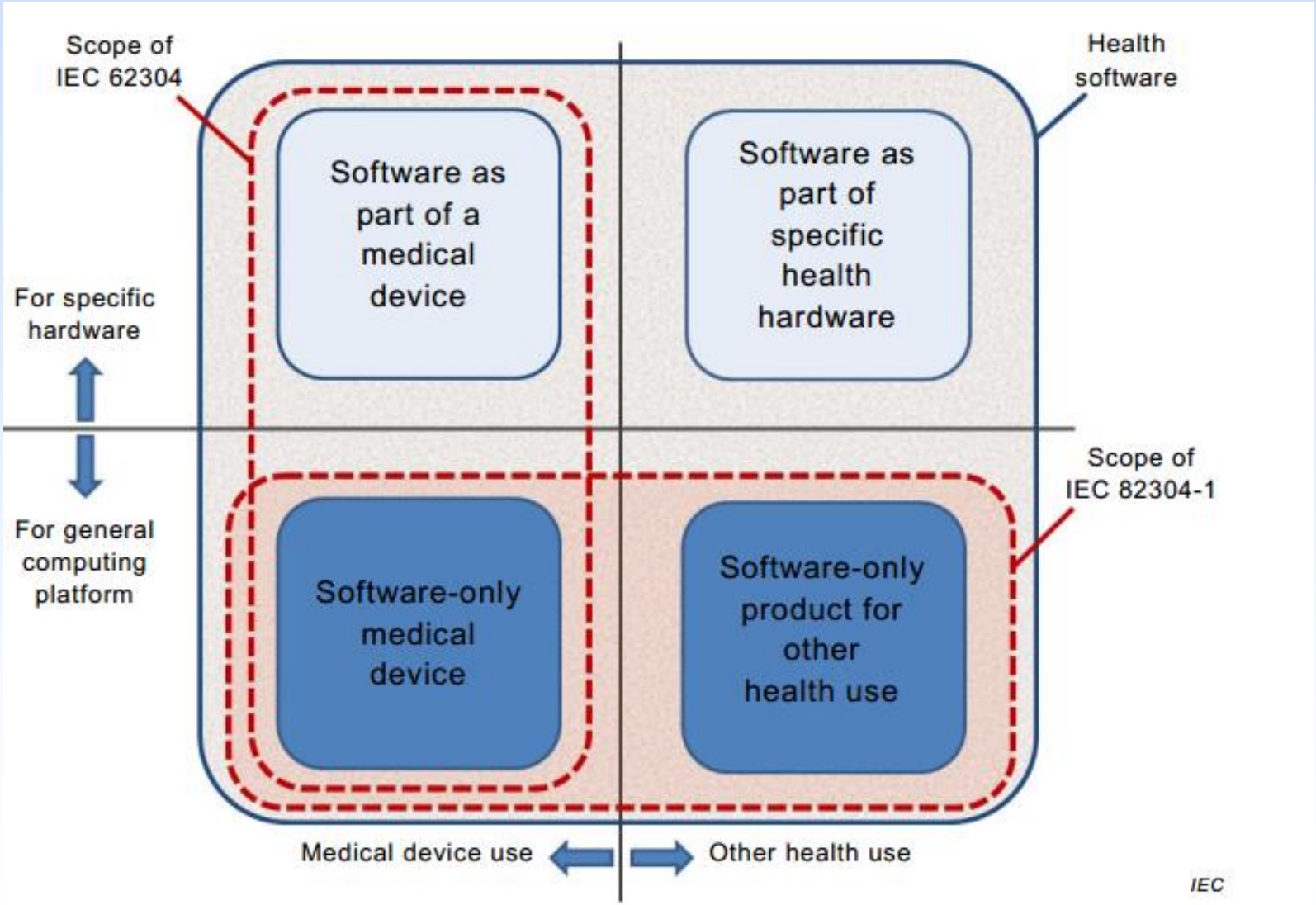
YY/T 0664-2008 Medical device software - Software life cycle processes (English)

Standard No.	YY/T 0664-2008	**Status**	Latest
Language		**File Format**	PDF
Price	320 USD	**Delivery Time**	within 1 busines

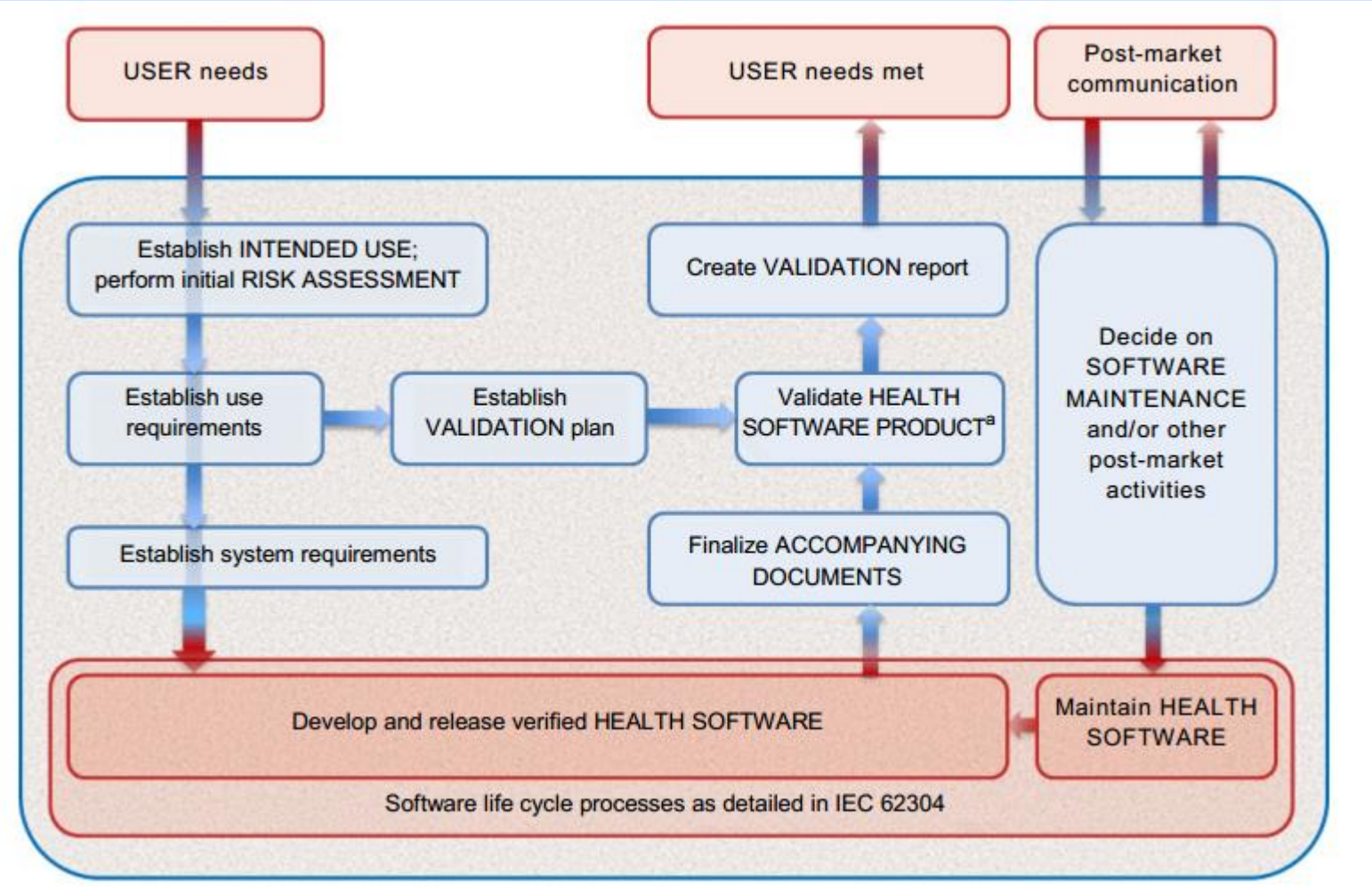
Relation with other standards



Scope IEC62304 versus IEC82304



Usage of IEC62304 for IEC82304



IEC62304: Software Validation not covered

This standard does not cover validation and final release of the MEDICAL DEVICE, even when the MEDICAL DEVICE consists entirely of software.

- Standalone Software: IEC82304:2016 (Guidance)
- Embedded software: IEC60601-1 PEMS

14.11 * PEMS VALIDATION

A PEMS VALIDATION plan shall include the validation of BASIC SAFETY and ESSENTIAL PERFORMANCE, and shall require checks for unintended functioning of the PEMS.

The PEMS VALIDATION shall be performed according to the PEMS VALIDATION plan. The results of PEMS VALIDATION activities shall be documented.

The person having the overall responsibility for the PEMS VALIDATION shall be independent of the design team. The MANUFACTURER shall document the rationale for the level of independence.

No member of a design team shall be responsible for the PEMS VALIDATION of their own design.

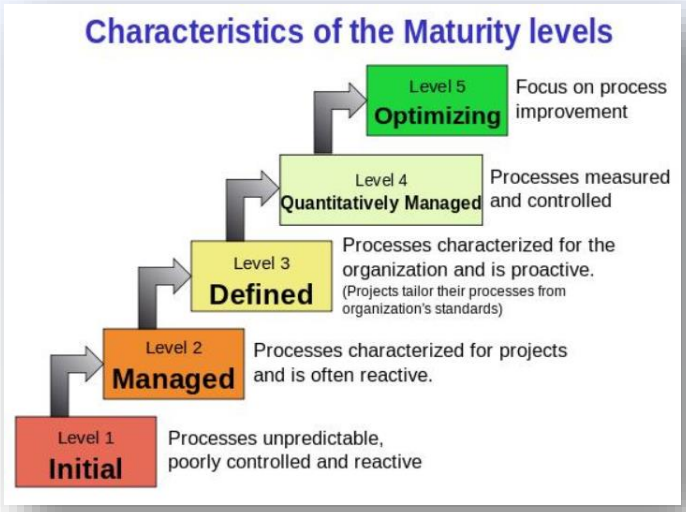
Overview IEC62304

3. Terms and definitions
4. General Requirements
5. Software Development
6. Software Maintenance
7. Software Risk Management
8. Software Configuration Management
9. Software Problem Resolution

So what is IEC62304 about?

Professional Software Development

Compare it with "good old" CMM for software



SOFTWARE CLASSIFICATION

How to define the Software classification

Class A

- No contribution to Hazardous situation OR
- Contributes to Hazardous situation but acceptable risk after external control measures

Class B

- Contributes to Hazardous situation
- Unacceptable risk after external control measures
- Resulting possible harm = non serious injury

Class C

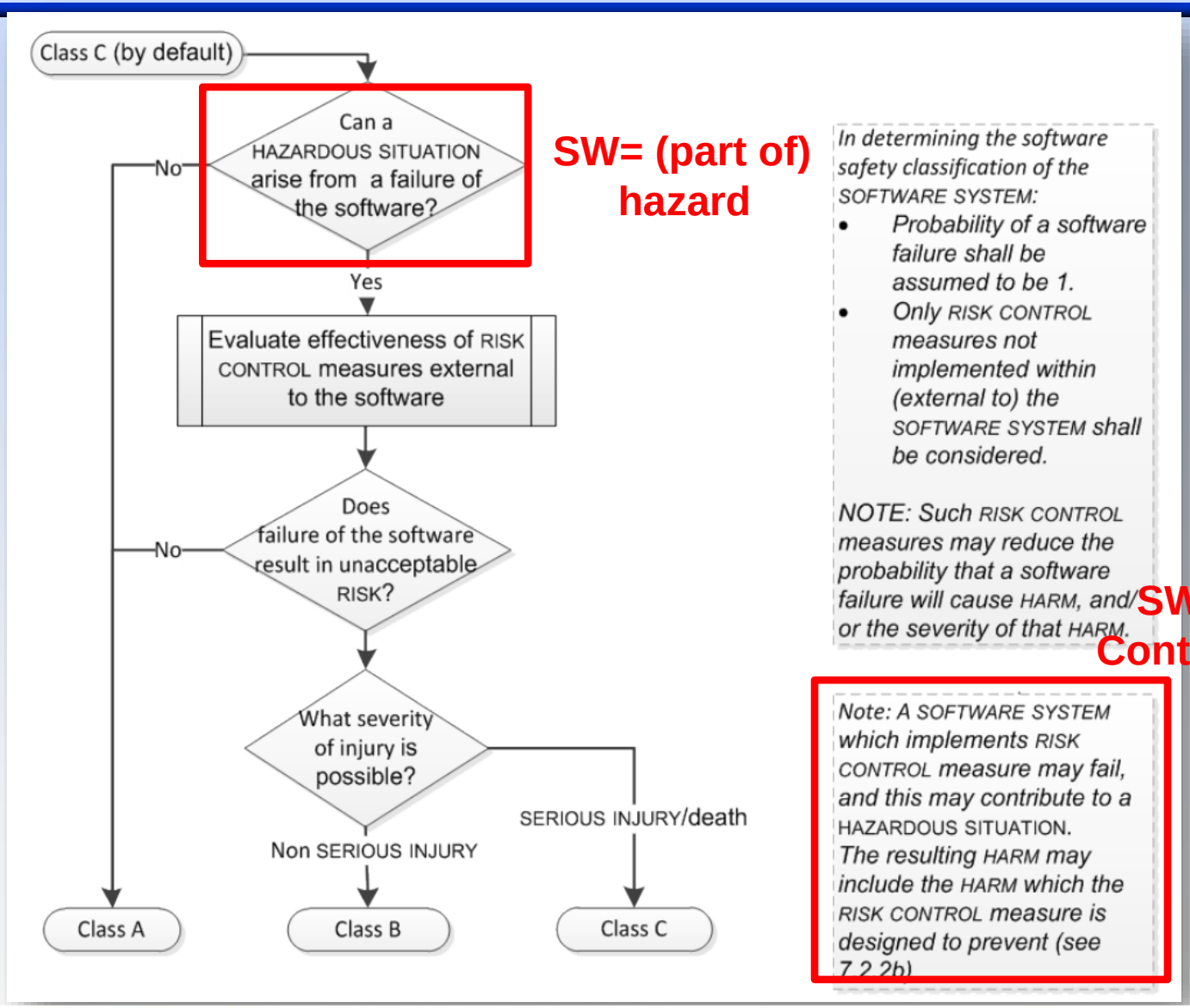
- Contributes to Hazardous situation
- Unacceptable risk after external control measures
- Resulting possible harm = serious injury or death

Impact of classification: applicable clauses (Ed 1.1)

Large impact on project timing & budget

		A	B	C
5	Software Development	22	46	52
5.1	Development Planning	7	10	11
5.2	Requirements Analysis	5	6	6
5.3	Architectural Design	0	4	5
5.4	Detailed Design	0	1	4
5.5	Software Coding	1	4	5
5.6	Integration and Testing	0	8	8
5.7	System Testing	5	5	5
5.8	Release	4	8	8
6	Software Maintenance	8	8	8
6.1	Establish Software Maintenance Plan	1	1	1
6.2	Problem and Modification Analysis	5	5	5
6.3	Modification Implementation	2	2	2
7	Software Risk Management	1	12	12
7.1	Analysis of software contributing to Hazards	0	5	5
7.2	Risk Control Measures	0	2	2
7.3	Verification of Risk Control Measures	0	2	2
7.4	Risk Management of Software Changes	1	3	3
8	Software Configuration Management	7	7	7
8.1	Configuration Identification	3	3	3
8.2	Configuration Control	4	4	4
8.3	Configuration Status Accounting	1	1	1
9	Software Problem Resolution	8	8	8

Classification Decision tree



NOTE 1 External RISK CONTROL measures can be hardware, an independent SOFTWARE SYSTEM, health care procedures, or other means to minimize that software can contribute to a HAZARDOUS SITUATION.

Classification checks – Rule of thumb

- Class B is appropriate in most cases
- Class C for high risk devices
  - infusion pumps, radiotherapy devices, pacemakers etc..
- Class A exceptional
  - minor software functions.
- Check risk management file:
  - If there are risks associated with the software, class A is unlikely

An example from real life – a TENS device

2.2 The process for [redacted] software device

1. Check the syststem risk analysis (HBSRA) if
   a. Software is (part of) a hazard
   b. Software contributes to a control measure
2. If software is (part of) a hazard
   a. The probability of the hazard must be set to >1000ppm
   b. The classification of the software is directly related to the severity of the hazard:
      i. Severity is 0 (no injury) → class A
      ii. severity is 1 or 2 → class B
      iii. severity is 3 or 4 → class C
   c. If it is class B/C has a hardware control measure that reduces the consequence of failure or the probability, it can be reclassified to A/B
3. If Software contributes to a control measure
   a. If the software control measure fails and there is also a hardware control measure that reduces the consequence of failure or the probability it can be reclassified

3 Actual Classification

Baseline 1.0 of HBSRA is used as input.

The following hazards are software related

HBSRA id	Software type	Severity	Initial Classification of software	Hardware control measure present that fulfils criteria of section 2.2?	Reclassification
HBSRA31	Software is (part of) a hazard	1	B	Yes	A
HBSRA82	Software is (part of) a hazard	1	B	Yes	A

Software does not contribute to a control measure

4 Decision

Taking the above into account it is decided that the software classification for both the embedded software in the [redacted] device and the mobile software app are **class A.**

What I have seen gone wrong

- Not implementing IEC62304 for class A software
- Legal manufacturer (system responsible) dictates that software to be developed by supplier X is class B
- System Risk Analysis NOT used as input
- Software control measures used to degrade classification
- No proper rationale why software is A or B
- IEC62304 classification equal to FDA Level of concern

Implementation of the IEC62304

Implementation of IEC62304

- Go through the standard, clause by clause
- Adapt your software process accordingly
- Adapt your software templates and checklists accordingly
- Adapt your software environment accordingly (version control, coding checkers, ….)

Example software requirement template

5.2.2 Software requirements content

As appropriate to the MEDICAL DEVICE SOFTWARE, the MANUFA
software requirements:

a) functional and capability requirements;

NOTE 1 Examples include:

- performance (e.g., purpose of software, timing requirements),
- physical characteristics (e.g., code language, platform, operating system),
- computing environment (e.g., hardware, memory size, processing unit, tim
  which the software is to perform, and
- need for compatibility with upgrades or multiple SOUP or other device versi

b) SOFTWARE SYSTEM inputs and outputs;

NOTE 2 Examples include:

- data characteristics (e.g., numerical, alpha-numeric, format)
- ranges,
- limits, and
- defaults.

c) interfaces between the SOFTWARE SYSTEM and other SYSTEMS;

d) software-driven alarms, warnings, and operator messages;

e) SECURITY requirements;

NOTE 3 Examples include:

- those related to the compromise of sensitive information,
- authentication,
- authorization,
- audit trail, and
- communication integrity,
- system security/malware protection.

f) user interface requirements implemented by software;

NOTE 4 Examples include those related to:

5.2.6 Installation and acceptance requirements

Define the installation and acceptance requirements at the operation and m
Ensure that released software can be reliably delivered to the point of use w
**Example is implementation of MD5 algoritm at installation of software and validatio
ment to be updated by the user.**

5.3 Safety Requirements

Table 5.2: Safety requirements

ID	Description

5.4 Security Requirements

Examples include:

- those related to the compromise of sensitive information,
- authentication,
- authorization,

[3] Refer to tagsd in RMF and/or design FMEA's

- audit trail, and
- communication integrity,
- system security/malware protection

Viewpoints for peer review process

5.2.6 Verify software requirements

The MANUFACTURER shall verify and document that the software requirements:

a) implement SYSTEM requirements including those relating to RISK CONTROL;
b) do not contradict one another;
c) are expressed in terms that avoid ambiguity;
d) are stated in terms that permit establishment of test criteria and performance of tests;
e) can be uniquely identified; and
f) are traceable to SYSTEM require

5.4.4 Verify detailed design

The MANUFACTURER shall verify and document that the software detailed design:

a) implements the software ARCHITECTURE; and
b) is free from contradiction with the software ARCHITECTURE.

5.6.2 Verify software integrati

The MANUFACTURER shall verify that the SOFTWARE UNITS have been integrated into SOFTWARE ITEMS and/or the SOFTWARE SYSTEM in accordance with the integration plan (see 5.1.5) and retain records of the evidence of such verification.

5.7.4 Evaluate SOFTWARE SYSTEM testing

The MANUFACTURER shall EVALUATE the appropriateness of VERIFICATION strategies and test procedures.

The MANUFACTURER shall verify that:

a) all software requirements have been tested or otherwise VERIFIED;
b) the TRACEABILITY between software requirements and tests or other VERIFICATION recorded; and
c) test results meet the required pass/fail criteria.

The ultimate compliance check

Fill the TRF for IEC62304 a.s.a.p. in the project

Clause	Requirement + Test	Evidence	Verdict (Pass/Fail)[1]
5.2	SOFTWARE REQUIREMENTS ANALYSIS		
5.2.1	[A, B, C] For each SOFTWARESYSTEM of the MEDICALDEVICE, the MANUFACTURER defined and documented SOFTWARESYSTEM requirements from the SYSTEM level requirements	[DDX-SRS]	Pass
5.2.2	[A, B, C] As appropriate to the MEDICALDEVICESOFTWARE, the MANUFACTURER included in the software requirements:		
	a) functional and capability requirements	[DDX-SRS]	Pass
	b) SOFTWARE SYSTEM inputs and outputs	[DDX-SRS]	Pass
	c) interfaces between the SOFTWARESYSTEM and other SYSTEMS	[DDX-SRS]	Pass
	d) software-driven alarms, warnings, and operator messages	[DDX-SRS]	Pass
	e) SECURITY requirements	N/A, not possible to update the firmware	N/A
	f) usability engineering requirements that are sensitive to human errors and training	[MM-URD]	Pass
	g) data definition and database requirements	[DDX-SRS]	Pass
	h) installation and acceptance requirements of the delivered MEDICAL DEVICE SOFTWARE at the operation and maintenance site or sites	N/A, consumer product without installation	N/A
	i) requirements related to methods of operation and maintenance	N/A, consumer product without installation	N/A
	j) user documentation to be developed	[MM-DFU]	Pass
	k) user maintenance requirements	[MM-DFU]	Pass
	l) regulatory requirements	[MM-CP]	Pass

Outsource your software development?

- If you outsource……..
- Select your supplier based on
- **Prooven** experience with medical device software
- With at least **class B** development

What I have seen gone wrong

- IEC6304 implementation in QMS not detailed out
- IEC62304 TRF filled in only at the end of the project (needed for submission to the notified body)
- No static or dynamic code checkers at all
- Outsource based on time/money not on software competence

Summary

- Always use IEC62304 for all medical device software
- Have a proper software classification process
- Use the system risk analysis
- Inject IEC62304 requirements in your QMS in detail
- Choose supplier on proof on MD SW experience
- Use the TRF IEC62304 for compliance check


The
End