

THE PARALLEL BETWEEN ELECTRONIC LEGAL MEASURING INSTRUMENTS AND (EMBEDDED) SOFTWARE IN THE CASE OF ESAFETY

Preliminary note : *This transposition is obviously a discussion paper, that has to be compared to the practise of eSafety stakeholders. As Welmec guides are in pdf format, I was not able to integrate them in this document.*

Personal analysis of the author and suggestions are in blue.

SUMMARY

ITS security and safety for eSafety are connected to the preservations of human life. Another field of public interest concerns metrology commonly called 'weight and measures'.

States have harmonized the evaluation of many measuring instruments and developed cooperation for a long time and they have published guides in the frame of WELMEC concerning the tests for the evaluation of measuring instruments, some of which concern electronics systems and communications and one concerns software requirements on the basis of the Directive on measuring instruments. The idea is to evaluate these WELMEC solutions for testing measuring instrument with embedded software and consider whether they could be adopted with profit by eSafety.

LEGAL METROLOGY

The legal measuring instruments have a role of judge and in case of good functioning and regular use their indication should be considered as the true value of the measurement. For reasons of public interest, public health, public safety, public order, protection of the environment, levying of taxes and duties, protection of the consumers and fair trading, public authorities are very concerned by these measurements.

To guaranty the indications of measuring instruments, MS and the Commission have settled very soon legislations and the control of the instruments is effective from the conception, to the manufacturing, the placing on the market, the maintenance and if need the installation and the conditions of use. The Measuring Instruments Directive 2004/22/EC ([MID](#)) has been adopted to harmonize MS legal metrology legislations.

MS and the Commission have settled Western European Legal Metrology Cooperation ([WELMEC](#)) to cope with these questions.

In the scope of MID, measuring instruments must respect operational specifications, and they should indicate value of the measured quantity in legal units (SI system with money units). They should wear indication of their measuring range and conditions of use.

They are submitted to a pattern approval tests where their metrological specifications are checked and they may receive a sealing plan to prevent unauthorized modification. The pattern approval applicant should expose the quality plan that will be used to maintain quality of the production. The instruments are checked at the initial control and are submitted if need to initial installation (where adjustments may be necessary), periodic control and use control.

Their maintenance is made by authorized persons and once repaired they are submitted to the same control as new instruments.

The emergence of electronics instruments, computers and communications has raised new questions needing a strong harmonization at the European level, so WELMEC has investigated these new questions publishing many guides, one of which concerns software-equipped measuring instruments. The [Welmec software guide](#) adds specification concerning software for measuring instruments covered by the MID, it make the distinction between devices supporting only legal measuring functions and multi application devices supporting other functions, proposes solutions to the problems connected to software certification, identification, check and download, software separation, software and parameters protection, protection of the interfaces, with different class of risks in consideration with the use of the instrument. Although the guide is oriented on instruments included in the regulations of the MID, the results are of a general nature and may be applied beyond MID.

ESAFETY

In ITS field, the functioning of electronics embedded equipments and the normal interactions between vehicles and infrastructure, vehicles and vehicles and vehicles and environment should not jeopardize security.

In ITS these items have been considered separately by stakeholders, so to get a better efficiency, it appears necessary to harmonize concepts structures and procedures, at an European level.

Vehicles should maintain during their life time a high level of security concerning their conformity to the specifications of the manufacturer and the compliance with the regulations.

NEW PARADIGM IN ITS

After a car accident, it was possible in case of electro mechanical equipments to prove the existence of non OEM equipment, that involved the responsibility of other persons than car manufacturers.

In the case of digital parameters or software in OEM hardware, it won't be possible to prove unauthorized intervention on the vehicle if these data have been erased during the accident. *Car manufacturers should be protected under such event as their responsibility could be involved with their insurance company. This involves that the conception of the interface of the vehicle should be so as to discourage unauthorized intervention and to give evidence that such intervention occurred.*

ANALYSIS OF WELMEC DOCUMENT IN VIEW OF SAFETY FOR eSAFETY

Foreword

The Welmec document takes into account the organization issued from the MID and concerns the additional specifications concerning embedded software. The structure of the document (in ANNEX) is explicit : it is a guide with many different requirements depending on the architecture of the instrument, the type of data that are stored, the transmission of data, the separation of software, and the download requirements, and risk classes depending on the use of the instrument and the level of conformity

It can be assumed that this Guide is a sensible approach to software on eSafety devices. The performance requirements that eSafety devices must meet, should provide a high level of protection. The conformity assessment should provide a high level of confidence. essential requirements that do not impede technical progress, preferably performance requirements should be specified and assessed.

In Legal Metrology the architecture defined by public authorities to maintain a high level of quality of the measuring instruments is always included, and it is necessary

to take this fact into account when reading this Guide, maybe to define the mainframe of this architecture for eSafety.

The state of the art in eSAfety technology, as in metrology, is subject to constant evolution which may lead to changes in the needs for conformity assessments

1 Introduction

Principles on the architecture of an electronic device with embedded software,
Principles of the chain of persons concerned by an electronic device during its life time, the system that is necessary to obtain a minimum level of quality, the traceability of the events concerning the life of the electronic device.

2 a Terminology that could be adapted to eSafety

Acceptable solution, Audit trail, Authentication, Basic configuration, Closed network, Open network, Communication interface, Fixed software, Integrated storage, Integrity of data and software, Software download, Software identification, Software separation, Risk class, Validation, Hash algorithm, Signature algorithm, Signature key, Public Key System (PKS), PKI Infrastructure, Certification of keys, Electronic signature, Trust Centre

2 b Terminology that need adaptation :

Built-for-purpose measuring instrument (type P), Measuring instruments using a universal computer (type U), Device-specific parameter, IT configuration, Legally relevant parameter, Long-term storage of measurement data, Sub-assembly, Transmission of measurement data, User interface

As the guide uses some terms of terminology to define the rules applicable to use it, it is necessary to adapt some terms of terminology

There is a need to specify some terms to transpose this guide to eSafety

I should suggest that for eSafety (type P) should concern built in for purpose computer and (type U) should concern a universal computer that can support many applications.

3 How to use this guide

This section describes the organisation of the guide and explains how to use it.

3.1 Overall structure of the guide

Consequently, there are three types of requirement sets:

1. requirements for two basic configurations (called type P and U),
2. requirements for four IT configurations (called extensions L, T, S and D)
3. instrument specific requirements (called extensions I.1, I.2, ...).

The first type of requirements is applicable to all instruments. The second type of requirements concerns the following IT functions: long-term storage of measurement data (**L**), transmission of measurement data (**T**), software download (**D**) and software separation (**S**). Each set of these requirements is only applicable if the corresponding function exists. The third type is instrument specific.

3.2 How to select the appropriate parts of the guide

This comprehensive software guide is applicable to a large variety of instruments. The guide is modular in form. The appropriate requirement sets can be easily selected by observing the following procedure.

Step 1: Selection of the basic configuration (*P or U*)

Step 2: Selection of applicable IT configurations (extensions *L, T, S and D*)

The sets selected depend only on the IT configuration. If an extension set is selected, then it must be applied in full.

Step 3: Selection of instrument specific requirements (extension *I*)

Step 4: Selection of the applicable risk class (extension *I*)

Select the risk class as defined in the respective instrument specific extension I.x, subchapter I.x.6. There, the risk class may be defined uniformly for a class of measuring instruments or further differentiated for categories, fields of application, etc. Once the applicable risk class has been selected, only the respective requirements and validation guidance need to be considered.

3.3 How to work with a requirement block

Each requirement block contains a well-defined requirement. It consists of a defining text, explanatory specifying notes, the documentation to be provided, the validation guidance and examples of acceptable solutions (if available). The content within a requirement block may be subdivided according to risk classes. This leads to the schematic presentation of a requirement block shown in Figure 3-3.

Title of the requirement

Main statement of the requirement (eventually differentiated between risk classes)

Specifying notes (scope of application, additional explanations, exceptional cases, etc.)

Documentation to be provided (eventually differentiated between risk classes)

Validation guidance for one risk class	Example of an acceptable solution for one risk class
Validation guidance for another risk class	Example of an acceptable solution for another risk class

Figure 3-3

4 Basic Requirements for Embedded Software in a Built-for-purpose Measuring Instrument (Type P)

see Welmec document

5 Basic Requirements for Software of Measuring Instruments using a Universal Computer (Type U)

see Welmec document

6 Extension L: Long-term Storage of Measurement Data

see Welmec document

7 Extension T: Transmission of Measurement Data via Communication Networks

This is an extension to the software requirements of the basic guides P and U. It must be used only if measurement data are transmitted via communication networks to a distant device where they are further processed and/or used for legally regulated purposes. This extension does not apply if there is no subsequent legally relevant data processing. If software is downloaded to a device subject to legal control the requirements of Extension D apply.

[This could concern car to car, car to infrastructure and infrastructure to infrastructure communications as communications between different eSafety functions within a car.](#)

7.1 Technical description

The set of requirements of this extension applies only if the device under consideration is connected to a network and transmits or receives measurement data that are legally relevant. In the following table three network configurations are identified. The simplest is an array of devices that are all subject to legal control. The participants are fixed at legal verification. A variant to this (closed network, partly under legal control), is a net with participants that are not subject to legal control but all are known and do not change during operation. An *open network* has no limitation in identity, functionality, presence and location of the participants.

Description of configurations

Closed network, completely under legal control

Only a fixed number of participants with clear identity, functionality and location are connected. All devices are subject to legal control. No devices exist in the network that are not subject to legal control.

Closed network, partly under legal control

A fixed number of participants with clear identity and location are connected to the network. Not all devices are subject to legal control and therefore their functionality is unknown.

Open network

Arbitrary participants (devices with arbitrary functions) can connect to the network. The identity and functionality of a participating device and its location may be unknown to other participants.

Any network that contains legally controlled devices with IR or wireless network communications interfaces shall be considered to be an open network.

Table 7-1: Technical description of a Type U measuring instrument.

7.2 Specific software Requirements for Data Transmission

see Welmec document

8 Extension S: Software Separation

Software separation is an optional design methodology that allows the manufacturer to easily modify non-legally relevant software. If software separation is implemented, then this extension shall be considered in addition to the basic requirements for types P and U.

8.1 Technical description

Software controlled measuring instruments or systems in general have complex functionality and contain modules that are legally relevant and modules that are not.

It is advantageous for the manufacturer and examiner – though it is not prescribed – to separate these software modules of the measuring system.

In the following table, two variants of software separation are described. Both variants are covered by the set of requirements.

Description

Software separation is realised independently from the operating system within an application domain, i.e., at the *programming language level* (**Low level software separation**).

Note: This feature is realisable in both built-for-purpose devices and universal computers.

The software modules to be separated are realised as independent objects in terms of the *operating system* (**High level software separation**).

Note: This type of separation is normally possible only with universal computers.

Example solutions are independently executable programs, dynamically linked libraries etc.

Table 8-1: Technical description of a Type U measuring instrument.

The protection against inadmissible changes of measurement values and parameters is only addressed indirectly as the programmer of software parts that are not subject to legal control must not give the user of the measuring system the opportunity of corruption. But this has in any case to be considered by the programmer (with or without separation) and the appropriate requirements are given in the basic parts P and U (Chapter 4 and 5) of the guide.

8.2 Specific software requirements for software separation

see Welmec document

9 Extension D: Download of Legally Relevant Software

This extension shall be used for the download of legally relevant software, e.g. bugfixes, updates, new applications, etc to measuring instruments of both types, P and U, as appropriate. These requirements are to be considered in addition to the basic requirements for Types P and Type-U described in Chapters 4 and 5 in the guide.

9.1 Technical Description

Software may be downloaded only to measuring instruments that are characterised by the following properties:

Hardware Configuration

The target device is subject to legal control. It may be a built-for-purpose measuring instrument (Type P) or one based on a universal computer (Type U).

Communications links for the download may be direct, e.g. RS 232, USB, over a closed network partly or wholly under legal control, e.g. Ethernet, token-ring LAN, or over an open network, e.g. Internet.

Software Configuration

The entire software of the target device may be legally controlled or it may have software separation. The download of legally relevant software must follow the requirements outlined below. If there is no software separation in the measuring instrument, then all of the requirements below apply to all downloads.

Table 9-1: Technical description of a Type U measuring instrument.

9.2 Specific Software Requirements

see Welmec document

10 Extension I: Instrument Specific Software Requirements

see Welmec document

11 Definition of Risk Classes

11.1 General principle

The requirements of this guide are differentiated according to (software) risk classes. Risks are related to software of the measuring instrument and not to any other risks. For convenience reasons, the shorter term “risk class” is used. Each measuring instrument must be assigned to a risk class because the particular software requirements to be applied are governed by the risk class the instrument belongs to. A risk class is defined by the combination of the appropriate levels required for software protection, software examination and software conformity. Three levels, low, middle and high are introduced for each of these categories.

11.2 Description of levels for protection, examination and conformity

The following definitions are used for the corresponding levels.

Software protection levels

Low: No particular protection measures against intentional changes are required.

Middle: The software is protected against intentional changes made by using easily-available and simple common software tools (e.g. text editors).

High: The software is protected against intentional changes made by using sophisticated software tools (debuggers and hard disc editors, software, development tools, etc).

Software examination levels

Low: Standard type examination functional testing of the instrument is performed. No extra software testing is required.

Middle: In addition to the low level, the software is examined on the basis of its documentation. The documentation includes the description of software functions, parameter description, etc. Practical tests of the software-supported functions (spot checks) may be carried out to check the plausibility of documentation and the effectiveness of protection measures.

High: In addition to the middle level, an in-depth test of the software is carried out, usually based on the source code.

Software conformity levels

Low: The functionality of the software implemented for each individual instrument is in conformity with the documentation approved.

Middle: In addition to the conformity level “low”, depending on the technical features, parts of the software shall be defined as fixed at type examination, i.e. alterable only with NB approval. The fixed part shall be identical in every individual instrument.

High: The software implemented in the individual instruments is completely identical to the approved one.

11.3 Derivation of risk classes

Out of the 27 theoretically possible level permutations, only 4 or at the utmost 5 are of practical interest (risk classes B, C, D and E, eventually F). They cover all of the instrument classes falling under the regulation of MID. Moreover, they provide a sufficient window of opportunity for the case of changing risk evaluations. The classes are defined in the table below.

Risk Class	Software Protection	Software Examination	Degree of Software Conformity
A	Low	Low	Low
B	Middle	Middle	Low
C	Middle	Middle	Middle
D	high	Middle	Middle
E	high	high	Middle
F	high	high	high

Table 11-1: *Definition of risk classes*

11.4 Interpretation of risk classes

Risk class A: It is the lowest risk class at all. No particular measures are required against intentional changes of software. Examination of software is part of the functional testing of the device. Conformity is required on the level of documentation. It is not expected that any instrument is classified as a risk class A instrument. However, by introducing this class, the corresponding possibility is held open.

Risk class B: In comparison to risk class A, the protection of software is required on the middle level. Correspondingly, the examination level is updated to the middle level. The conformity remains unchanged in comparison to risk class A.

Risk class C: In comparison to risk class B, the conformity level is raised to “middle”. This means, parts of the software may be declared as fixed at type examination. The rest of the software is required to be conform on the functional level. The levels of protection and examination remain unchanged in comparison to risk class B.

Risk class D: The significant difference in comparison to risk class C is the upgrade of the protection level to “high”. Since the examination level remains unaffected at “middle”, sufficiently informative documentation must be provided to show that the protection

measures taken are appropriate. The conformity level remains unchanged in comparison to risk class C.

Risk class E: In comparison to risk class D, the examination level is upgraded to “high”. The levels of protection and conformity remain unchanged.

Risk class F: The levels with respect to all aspects (protection, examination and conformity) are set to “high”. Like risk class A, it is not expected that any instrument is classified as a risk F instrument. However, by introducing this class, the corresponding possibility is held open.

12 Pattern for Test Report (Including Checklists) to the end of the document

see WELMEC document

CONCLUSION

The guide defines different risk classes, with increasing severity, the correspondence with eSafety architecture is open for discussion.

It addresses both, manufacturers of eSafety devices and conformity assessment of these devices.

It appears that the operational and systematic approach of WELMEC is very rich, and that it could probably be used as a mainframe to elaborate eSafety application specifications.

But many questions as the architecture of the eSafety device, the architecture of the controls and of the surveillance to be performed, the persons involved in the determination of operational specifications, in evaluating the performance in the quality system to sustain a high level of confidence, should be considered.

The existing standards used by eSafety stakeholders should be listed and compared with the Welmec guide recommendations.

There is a need that the different stakeholders consider this WELMEC guide and analyse it to give their advice on it.

For that purpose it should be interesting to try to define the typology of software to be considered (OS, general bookshelf functions, eSafety functions, communications functions etc;) in relation with the structure of the electronic system.

ANNEX : Structure of the Document

Foreword

1 Introduction

2 Terminology

3 How to use this guide

3.1 Overall structure of the guide

3.2 How to select the appropriate parts of the guide

3.3 How to work with a requirement block

3.4 How to work with the checklists

4 Basic Requirements for Embedded Software in a Built-for-purpose Measuring Instrument (Type P)

In this case we should transpose this to

Basic requirements for embedded software in an electronic device only designed for eSafety applications

4.1 Technical Description

4.2 Specific Requirements for Type P

5 Basic Requirements for Software of Measuring Instruments using a Universal Computer (Type U)

In this case we should transpose this to

Basic requirements for embedded software of an eSafety electronic device using a universal computer.

5.1 Technical Description

5.2 Specific Software Requirements for Type U

6 Extension L: Long-term Storage of Measurement Data

This could concerns the parameters storage

6.1 Technical description

6.2 Specific software requirements for Long-term Storage

7 Extension T: Transmission of Measurement Data via Communication Networks

7.1 Technical description

7.2 Specific software Requirements for Data Transmission

8 Extension S: Software Separation

8.1 Technical description

8.2 Specific software requirements for software separation

9 Extension D: Download of Legally Relevant Software

9.1 Technical Description

9.2 Specific Software Requirements

10 Extension I: Instrument Specific Software Requirements

10.1 Water Meters

10.2 Gas Meters and Volume Conversion Devices

10.3 Active Electrical Energy Meters

10.4 Heat Meters

10.5 Measuring Systems for the Continuous and Dynamic Measurement of Quantities of Liquids Other than Water

10.6 Weighing Instruments

10.7 Taximeters

10.8 Material Measures

10.9 Dimensional Measuring Instruments

10.10 Exhaust Gas Analysers

11 Definition of Risk Classes

11.1 General principle

11.2 Description of levels for protection, examination and conformity

11.3 Derivation of risk classes

11.4 Interpretation of risk classes

12 Pattern for Test Report (Including Checklists)

12.1 Pattern for the general part of the test report

12.2 Annex 1 of the test report: Checklists to support the selection of the appropriate requirement Sets

12.3 Annex 2 of the test report: Specific checklists for the respective technical parts

12.4 Information to be included in the type approval certificate

13 Cross Reference for MID-Software Requirements to MID Articles and Annexes

13.1 Given software requirement, reference to MID

13.2 Interpretation of MID Articles and Annexes by MID-Software Requirements

14 References and Literature

15 Index