SLOVENSKI STANDARD
SIST EN 9115:2013

01-julij-2013

Sistemi vodenja kakovosti - Zahteve za organizacije za zračni promet, astronavtiko in obrambo - Dobavljivost programske opreme (Dopolnilo k EN 9100)

Quality Management Systems - Requirements for Aviation, Space and Defence Organizations - Deliverable Software (Supplement to EN 9100)

Qualitätsmanagementsysteme - Anforderungen an Organisationen der Luftfahrt, Raumfahrt und Verteidigung - Lieferbare Software (Ergänzung zu EN 9100)

Systèmes de management de la Qualité - Exigences pour les Organisations de l'Aéronautique, l'Espace et la Défense - Logiciel livrable (Supplément à l'EN 9100)

Ta slovenski standard je istoveten z: **EN 9115:2013**

ICS:

03.120.10	Vodenje in zagotavljanje kakovosti	Quality management and quality assurance
49.020	Letala in vesoljska vozila na splošno	Aircraft and space vehicles in general
95.020	Vojaška tehnika. Vojaške zadeve. Orožje	Military engineering. Military affairs. Weapons

SIST EN 9115:2013 **en,fr,de**

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 9115

January 2013

ICS 03.120.10; 49.020

English Version

Quality Management Systems - Requirements for Aviation, Space and Defense Organizations - Deliverable Software (Supplement to EN 9100)

Systèmes de management de la Qualité - Exigences pour les Organisations de l'Aéronautique, l'Espace et la Défense - Logiciel livrable (Supplément à l'EN 9100)

Qualitätsmanagementsysteme - Anforderungen an Organisationen der Luftfahrt, Raumfahrt und Verteidigung - Mitgelieferte Software (Ergänzung zu EN 9100)

This European Standard was approved by CEN on 18 June 2011.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

Management Centre: Avenue Marnix 17, B-1000 Brussels

Contents

Page

Foreword

This document (EN 9115:2013) has been prepared by the Aerospace and Defence Industries Association of Europe - Standardization (ASD-STAN).

After enquiries and votes carried out in accordance with the rules of this Association, this Standard has received the approval of the National Associations and the Official Services of the member countries of ASD, prior to its presentation to CEN.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by July 2013, and conflicting national standards shall be withdrawn at the latest by July 2013.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

According to the CEN/CENELEC Internal Regulations, the national standards organisations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

This document standardizes, to the greatest extent possible, the software quality management system requirements for the aviation, space, and defense industry. This was accomplished through the harmonization of quality management system requirements from international aviation, space, and defense software standards and other applicable documents. The establishment of common requirements for use at all levels of the supply-chain by organizations around the world should result in improved quality, schedule, and cost performance by the reduction or elimination of organization unique requirements and wider application of good practice.

SUMMARY/RATIONALE

The 9115 document supersedes AS9006, "Deliverable Aerospace Software Supplement for AS9100A, Quality Management Systems — Aerospace — Requirements for Software", published in March 2003. The AS9006 standard was published as an Americas Aerospace Quality Group (AAQG) sector specific document.

This is the initial release of 9115, which is an international supplement to 9100 providing clarification of the corresponding 9100 requirements, as necessary, for deliverable software. In some cases, where clarification is needed, it was necessary due to the complexity of software to decompose "shall" statements in 9100 into more granular requirements. Where no software clarification is required of the 9100 requirements, the following phrase will be presented: "The requirements of 9100 apply. No clarification required for software."

NOTE ***This document must be used in conjunction with EN 9100; references throughout the text to EN 9100 are understood to mean EN 9100:2009.***

0 Introduction

0.1 General

The requirements of EN 9100 apply. No clarification required for software.

0.2 Process approach

The requirements of EN 9100 apply. No clarification required for software.

QUALITY MANAGEMENT SYSTEMS — REQUIREMENTS

1 Scope

1.1 General

The requirements of EN 9100 apply with the following clarification for software.

This document supplements the EN 9100 standard requirements for deliverable software and contains quality management system requirements for organizations that design, develop, and/or produce deliverable software for the aviation, space, and defense industry. This includes, as required, support software that is used in the development and maintenance of deliverable software. The deliverable software may be stand-alone, embedded, or loadable into a target computer.

Where the use of Hardware Description Language (HDL) or high order language is utilized as the design source of electronic hardware [e.g., Application Specific Integrated Circuit (ASIC), Programmable Logic Device (PLD)], the organization and customer shall agree on the extent of applicability of this supplement.

NOTE 1 For airborne electronic hardware guidance, see RTCA/DO-254 or EUROCAE ED-80; and for product realization requirements, see EN 9100.

Where Commercial-off-the-Shelf (COTS) or non-developmental software is integrated into a deliverable product, the organization and customer shall agree on the extent of applicability of this supplement.

For the purposes of this document, the terms “product” and “software product” are considered synonymous.

NOTE 2 This document is independent of the life cycle models (e.g., waterfall, spiral, evolutionary, incremental) or methodology (e.g., objected oriented design, unified modeling language, agile).

1.2 Application

The requirements of EN 9100 apply with the following clarification for software.

Exclusions to requirements in Clause 7 should only be considered after analysis of software attributes (e.g., size, safety, security, complexity, criticality, risk).

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE 1 The requirements of EN 9100 apply with the following clarification for software.

EN 9100:2009, *Quality Management Systems — Requirements for Aviation, Space and Defence Organizations*

NOTE 2 Documents referenced in this document, other than the normative references (i.e., 9100, ISO 9000) are listed in the Bibliography. For undated references, the latest edition of the referenced document (including any amendments) applies. The referenced documents are “informative” references; the requirements of these referenced documents do not add any additional requirements to this standard.

3 Terms and definitions

For the purposes of this document, the terms and definitions given in EN 9100 and ISO 9000 apply. The following terms and definitions are included to support the understanding of this document.

3.1
baseline
the approved, recorded configuration of one or more configuration items, that thereafter serves as the basis for further development, and that is changed only through change control procedures

[SOURCE: RTCA/DO-178, EUROCAE ED-12]

3.2
Commercial-Off-The-Shelf (COTS) software

commercially available applications sold by vendors through public catalog listings. COTS software is not intended to be customized or enhanced. Contract-negotiated software developed for a specific application is not COTS software.

[SOURCE: RTCA/DO-178, EUROCAE ED-12]

Note 1 to entry: COTS software is a type of non-developmental software.

3.3
configuration item
one or more hardware/software entities treated as a unit for configuration management purposes or software life cycle data treated as a unit for configuration management purposes

[SOURCE: based on RTCA/DO-178, EUROCAE ED-12]

3.4
critical items
the definition in EN 9100, Clause 3.3, applies with the following clarification for software

Critical items in software are those characteristics, requirements, or attributes that have been determined to be most important to achieve product realization (e.g., safety, maintainability, testability, usability, performance). Critical items should be adequately managed and appropriate action taken to ensure visibility throughout the product life cycle. For example, in a flight control system software response time can be elevated to a critical item to ensure performance characteristics are met. Furthermore, if the project has specific testability requirements, cyclomatic complexity may become a critical item.

3.5
cyclic redundancy check (CRC)
a type of function that takes as input a data stream of any length and produces as output a value of a certain space, commonly a 32-bit integer. A CRC can be used to detect alteration of data during transmission or storage.

3.6
digital signature
a type of asymmetric cryptography used to express compliance with the security properties of a handwritten signature on paper, also referred to as a digital signature scheme

3.7
key characteristic
the definition in EN 9100, Clause 3.4, applies with the following clarification for software

Key characteristics in software are those measurable attributes where variability can be measured by the project and can, if left unchecked, adversely impact the project or product in areas (e.g., schedule, cost, maintainability, testability, reliability, portability). Examples of key characteristics include defect severity, complexity factors, nested menus, memory, timing, response time, and throughput targets.

3.8
Monitoring
the act of witnessing or inspecting selected instances of test, inspections, or other activities, or records of those activities, to assure that the activity is under control and that the reported results are representative of the expected results

Monitoring is usually associated with activities done over an extended period of time where 100 % witnessing is considered impractical or unnecessary. Monitoring permits authentication that the claimed activity was performed as planned.

[SOURCE: RTCA/DO-178, EUROCAE ED-12]

3.9
non-developmental software
deliverable software that is not developed under the contract, but is provided by the organization, customer, or a third party (e.g., reused software, customer furnished software, COTS software, open source software)

3.10
phase
a collection of processes, activities, tasks, and outcomes within the software life cycle

3.11
release
a particular version of a configuration item that is made available for a specific purpose (e.g., test release)

[SOURCE: ISO/IEC 12207]

3.12
reliability
the probability of failure-free operation of a computer program in a specified environment for a specified time

[SOURCE: based on IEEE-STD-982.1]

Note 1 to entry: Software reliability requirements should consider the level and manner of fault and failure detection, isolation, fault tolerance, and recovery expected to be fulfilled by the software.

3.13
risk
the definition in EN 9100 (see 3.1) applies. No clarification required for software.

3.14
robustness
the extent to which software can continue to operate correctly despite invalid inputs

[SOURCE: RTCA/DO-178, EUROCAE ED-12]

Note 1 to entry: Robustness, in the software context, means that the organization has utilized techniques (e.g., exception handling, redundancy, related verification techniques).

3.15
secure hash algorithm
cryptographic functions that compute a fixed-length digital representation, known as a message digest, of an input data sequence of any length

3.16
software
computer programs, associated documentation, and data pertaining to the operation of a computer system

[SOURCE: based on RTCA/DO-178, EUROCAE ED-12]

Note 1 to entry: The executable programs and data that are embedded in hardware devices are considered to be included in this definition (i.e., firmware).

Note 2 to entry: Firmware is the combination of a hardware memory device loaded with computer instructions and/or digital data that reside as read-only software on a device that a computing system can read. The software cannot typically be readily modified under program control.

3.17
software life cycle
the period of time that begins with the decision to produce or modify software and ends when the software product support is no longer required. The software life cycle contains phases.

Note 1 to entry: The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and sometimes, retirement phase

[SOURCE: IEEE-STD-610.12]

3.18
software product
the set of computer programs and associated documentation and data intended for, or required by, a customer; or any intended output resulting from the product realization process

Note 1 to entry: A software product may be designated for delivery, an integral part of another software or hardware product, or used in the development process.

3.19
special requirements
the definition in EN 9100, Clause 3.2, applies with the following clarification for software

Examples of special requirements that may introduce high risk for software include: the introduction of a new compiler, new advanced modeling technique, qualification of tools, novel test equipment capabilities, introduction of a new type of interface, or novel customer technical requirements. These requirements are included in the risk management process.