SLOVENSKI STANDARD
SIST EN 62628:2012

01-december-2012

Navodilo o programskih vidikih zagotovljivosti

Guidance on software aspects of dependability

Leitlinien zu Softwareaspekten der Zuverlässigkeit

Lignes directrices concernant la sûreté de fonctionnement du logiciel

Ta slovenski standard je istoveten z: **EN 62628:2012**

ICS:

03.120.01	Kakovost na splošno	Quality in general
35.020	Informacijska tehnika in tehnologija na splošno	Information technology (IT) in general

SIST EN 62628:2012 **en**

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 62628

September 2012

ICS 03.120.01

English version

Guidance on software aspects of dependability
(IEC 62628:2012)

Lignes directrices concernant la sûreté de fonctionnement du logiciel
(CEI 62628:2012)

Leitlinien zu Softwareaspekten der Zuverlässigkeit
(IEC 62628:2012)

CENELEC

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

Management Centre: Avenue Marnix 17, B - 1000 Brussels

Ref. No. EN 62628:2012 E

Foreword

The text of document 56/1469/FDIS, future edition 1 of IEC 62628, prepared by IEC/TC 56, "Dependability" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 62628:2012.

The following dates are fixed:

• latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement	(dop)	2013-06-12
• latest date by which the national standards conflicting with the document have to be withdrawn	(dow)	2015-09-12

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

Endorsement notice

The text of the International Standard IEC 62628:2012 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

IEC 62508	NOTE	Harmonized as EN 62508.
IEC 60300-1	NOTE	Harmonized as EN 60300-1.
IEC 60300-2	NOTE	Harmonized as EN 60300-2.
IEC 60300-3-3	NOTE	Harmonized as EN 60300-3-3.
IEC 62347	NOTE	Harmonized as EN 62347.
IEC 61160	NOTE	Harmonized as EN 61160.
IEC 61078	NOTE	Harmonized as EN 61078.
IEC 61025	NOTE	Harmonized as EN 61025.
IEC 61165	NOTE	Harmonized as EN 61165.
IEC 62551[1)]	NOTE	Harmonized as EN 62551[1)].
IEC 60812	NOTE	Harmonized as EN 60812.
IEC 60300-3-1	NOTE	Harmonized as EN 60300-3-1.
IEC 61508-3	NOTE	Harmonized as EN 61508-3.
IEC 62429	NOTE	Harmonized as EN 62429.
IEC 61014	NOTE	Harmonized as EN 61014.
IEC 61164	NOTE	Harmonized as EN 61164.
IEC 62506[1)]	NOTE	Harmonized as EN 62506[1)].

[1)] To be published.

Annex ZA
(normative)
Normative references to international publications with their corresponding European publications

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 When an international publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

Publication	Year	Title	EN/HD	Year
IEC 60050-191	-	International Electrotechnical Vocabulary (IEV) - Chapter 191: Dependability and quality of service	-	-
IEC 60300-3-15	-	Dependability management - Part 3-15: Application guide - Engineering of system dependability	EN 60300-3-15	-

IEC 62628

Edition 1.0 2012-08

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Guidance on software aspects of dependability

Lignes directrices concernant la sûreté de fonctionnement du logiciel

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX XB

ICS 03.120.01

ISBN 978-2-83220-303-3

CONTENTS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

GUIDANCE ON SOFTWARE ASPECTS OF DEPENDABILITY

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62628 has been prepared by IEC technical committee 56: Dependability.

The text of this standard is based on the following documents:

FDIS	Report on voting
56/1469/FDIS	56/1480/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

INTRODUCTION

Software has widespread applications in today's products and systems. Examples include software applications in programmable control equipment, computer systems and communication networks. Over the years, many standards have been developed for software engineering, software process management, software quality and reliability assurance, but only a few standards have addressed the software issues from a dependability perspective.

Dependability is the ability of a system to perform as and when required to meet specific objectives under given conditions of use. The dependability of a system infers that the system is trustworthy and capable of performing the desired service upon demand to satisfy user needs. The increasing trends in software applications in the service industry have permeated in the rapid growth of Internet services and Web development. Standardized interfaces and protocols have enabled the use of third-party software functionality over the Internet to permit cross-platform, cross-provider, and cross-domain applications. Software has become a driving mechanism to realize complex system operations and enable the achievement of viable e-businesses for seamless integration and enterprise process management. Software design has assumed the primary function in data processing, safety monitoring, security protection and communication links in network services. This paradigm shift has put the global business communities in trust of a situation relying heavily on the software systems to sustain business operations. Software dependability plays a dominant role to influence the success in system performance and data integrity.

This International Standard provides current industry best practices and presents relevant methodology to facilitate the achievement of software dependability. It identifies the influence of management on software aspects of dependability and provides relevant technical processes to engineer software dependability into systems. The evolution of software technology and rapid adaptation of software applications in industry practices have created the need for practical software dependability standard for the global business environment. A structured approach is provided for guidance on the use of this standard.

The generic software dependability requirements and processes are presented in this standard. They form the basis for dependability applications for most software product development and software system implementation. Additional requirements are needed for mission critical, safety and security applications. Industry specific software qualification issues for reliability and quality conformance are not addressed in this standard.

This standard can also serve as guidance for dependability design of firmware. It does not however, address the implementation aspects of firmware with software contained or embedded in the hardware chips to realize their dedicated functions. Examples include application specific integrated circuit (ASIC) chips and microprocessor driven controller devices. These products are often designed and integrated as part of the physical hardware features to minimize their size and weight and facilitate real time applications such as those used in cell phones. Although the general dependability principles and practices described in this standard can be used to guide design and application of firmware, specific requirements are needed for their physical construction, device fabrication and embedded software product implementation. The physics of failure of application specific devices behaves differently as compared to software system failures.

This International Standard is not intended for conformity assessment or certification purposes.

GUIDANCE ON SOFTWARE ASPECTS OF DEPENDABILITY

1 Scope

This International Standard addresses the issues concerning software aspects of dependability and gives guidance on achievement of dependability in software performance influenced by management disciplines, design processes and application environments. It establishes a generic framework on software dependability requirements, provides a software dependability process for system life cycle applications, presents assurance criteria and methodology for software dependability design and implementation and provides practical approaches for performance evaluation and measurement of dependability characteristics in software systems.

This standard is applicable for guidance to software system developers and suppliers, system integrators, operators and maintainers and users of software systems who are concerned with practical approaches and application engineering to achieve dependability of software products and systems.

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.

IEC 60050-191, *International Electrotechnical Vocabulary – Chapter 191: Dependability and quality of service*

IEC 60300-3-15, *Dependability management – Part 3-15: Application guide – Engineering of system dependability*

3 Terms, definitions and abbreviations

For the purposes of this document, the terms and definitions given in IEC 60050-191, as well as the following apply.

3.1 Terms and definitions

3.1.1
software
programs, procedures, rules, documentation and data of an information processing system

Note 1 to entry: Software is an intellectual creation that is independent of the medium upon which it is recorded.

Note 2 to entry: Software requires hardware devices to execute programs and to store and transmit data.

Note 3 to entry: Types of software include firmware, system software and application software.

Note 4 to entry: Documentation includes: requirements specifications, design specifications, source code listings, comments in source code, “help” text and messages for display at the computer/human interface, installation instructions, operating instructions, user manuals and support guides used in software maintenance.

3.1.2
firmware
software contained in a read-only memory device, and not intended for modification

EXAMPLE Basic input/output system (BIOS) of a personal computer.

Note 1 to entry: Software modification requires the hardware device containing it to be replaced or re-programmed.

3.1.3
embedded software
software within a system whose primary purpose is not computational

EXAMPLES Software used in the engine management system or brake control systems of motor vehicles.

3.1.4
software unit
software module
software element that can be separately compiled in programming codes to perform a task or activity to achieve a desired outcome of a software function or functions

Note 1 to entry: The terms "module" and "unit" are often used interchangeably or defined to be sub-elements of one another in different ways depending upon the context. The relationship of these terms is not yet standardized.

Note 2 to entry: In an ideal situation, a software unit can be designed and programmed to perform exactly a specific function. In some applications, it may require two or more software units combined to achieve the specified software function. In such cases, these software units are tested as a single software function.

3.1.5
software configuration item
software item that has been configured and treated as a single item in the configuration management process

Note 1 to entry: A software configuration item can consist of one or more software units to perform a software function.

3.1.6
software function
elementary operation performed by the software module or unit as specified or defined as per stated requirements

3.1.7
software system
defined set of software items that, when integrated, behave collectively to satisfy a requirement

EXAMPLES Application software (software for accounting and information management); programming software (software for performance analysis and CASE tools) and system software (software for control and management of computer hardware system such as operating systems).

3.1.8
software dependability
ability of the software item to perform as and when required when integrated in system operation

3.1.9
software fault
bug
state of a software item that may prevent it from performing as required

Note 1 to entry: Software faults are either specification faults, design faults, programming faults, compiler-inserted faults or faults introduced during software maintenance.

Note 2 to entry: A software fault is dormant until activated by a specific trigger, and usually reverts to being dormant when the trigger is removed.

Note 3 to entry: In the context of this standard, a bug is a special case of software fault also known as latent software fault.

3.1.10
software failure
failure that is a manifestation of a software fault

Note 1 to entry: A single software fault will continue to manifest itself as a failure until it is removed.

3.1.11
code
character or bit pattern that is assigned a particular meaning to express a computer program in a programming language

Note 1 to entry: Source codes are coded instructions and data definitions expressed in a form suitable for input to an assembler, compiler, or other translator.

Note 2 to entry: Coding is the process of transforming of logic and data from design specifications or descriptions into a programming language.

Note 3 to entry: A programming language is a language used to express computer programs.

3.1.12
(computer) program
set of coded instructions executed to perform specified logical and mathematical operations on data

Note 1 to entry: Programming is the general activity of software development in which the programmer or computer user states a specific set of instructions that the computer must perform.

Note 2 to entry: A program consists of a combination of coded instructions and data definitions that enable computer hardware to perform computational or control functions.

3.2 Abbreviations

ASIC	Application specific integrated circuit
CASE	Computer-aided software engineering
CMM	Capability maturity model
CMMI	Capability maturity model integration
COTS	Commercial-off-the-shelf
FMEA	Failure mode and effects analysis
FTA	Fault tree analysis
IP	Internet protocol
IT	Information technology
KSLOC	Kilo-(thousand) source lines of code
ODC	Orthogonal defect classification
RBD	Reliability block diagram
USB	Universal serial bus

4 Overview of software aspects of dependability

4.1 Software and software systems

Software is a virtual entity. In the context of this standard, software refers to procedures, programs, codes, data and instructions for system control and information processing. A software system consists of an integrated collection of software items such as computer programs, procedures, and executable codes, and incorporated into physical host of the processing and control hardware to realize system operation and deliver performance functions. The hierarchy of the software system can be viewed as a structure representing the system architecture and consisting of subsystem software programs and lower-level software units. A software unit can be tested as specified in the design of a program. In some cases,