


Potapov I.V., Baeva M.A.

TERMINOLOGY ISSUES RELATED TO RELIABILITY OF PROGRAMS AND SOFTWARE

This article deals with the elaboration of definition of software reliability. This task is considered within the context of a generic terminological problem arisen due to the necessity to coordinate the Russian terminology with the definitions used in international practice. The paper represents a list of definitions of software dependability assumed to be a complex property. Several software properties specifying reliability are considered. The article also contains the table of applicability of these properties to describe the reliability of various types of software programs.

Keywords: *reliability of software.*

Introduction

Today the reliability experts are paying more attention to the issues of terminology. This is due to the necessity to revise the main standard "Dependability in equipment" which defines the terminology in this field, and to the necessity to coordinate the major applicable terms with the international standards. The detailed information about the ways how these issues are discussed as well as about associated difficulties is provided the papers [1, 2].

Similar difficulties can be faced by the specialists exploring the issues of reliability of information systems (IS). In particular, this applies to IS software. Here the terminological difficulties occur already at the use of basic concepts. For instance, in the standards GOST 19781-90 "Software of data processing systems. Terms and definitions" and GOST 28806-90 "Software quality. Terms and definitions," which actually should be used jointly on the mutually complementary basis, in fact one and the same term is called in different ways: "Software" and "Software tool". Curiously, GOST 19781–90 standard does not contain this term in English, although in the English title of the standard the term "software" is applied. On the other hand, the term "software program" has the same definitions in these standards, and that is important because it is software programs that we shall speak about in this article. Moreover, let us underline that this paper can also use the term "Software tool" in the definition according to GOST 28806–90 (the annex to the standard stipulates that this term includes the scope of the term "Software"). It seems that the most important thing here is the fact that software programs and software tools (ST) can be understood by the readers in the general meaning including other terms used in references: "software systems", "software complexes", etc.

The standards related to dependability of software programs and software tools are also interpreted with terminological difficulties. Let us point out the main reasons. Firstly, there are different definitions related to the terminology of ST dependability, including the translated ones, taken from international standards. Secondly, when these definitions and terms are translated, the existing standardized terminology also should be taken into account and it may cause difficulties. Thirdly, the existing effective standard GOST 27.002–89 that defines the most amount of terminology in the field of dependability is oriented towards technical

facilities, but not towards programs for these facilities, i.e. the substantial differences of ST are not covered in this standard. These reasons for terminological difficulties are supplemented by the standards dedicated to ST quality. Such standards consider the term "reliability" only as one of properties specifying quality. There will be more detailed information about it further in the text. It should be added that the above mentioned difficulties are essential within the whole topic of ST dependability [3].

Along with the above we can conclude about the necessity of in-depth study of terminology issues related to ST dependability that is quite consistent with a general tendency.

Problem description

Based on the abovementioned, there appears the necessity to study the existing variety of terms in order to choose the most appropriate ones. The problem could be described in more detail: it is necessary to consider several known definitions of ST dependability and, for the most important properties within the ST dependability, to perform qualitative assessment of the possibility of their practical application for a wide variety of software programs. In other words, it is proposed to perform the analysis of current terminology searching for definitions which at least partially could be used as the operational ones, whose concept is worded in [2]. It is quite a complicated task, and that is why this article covers only part of the whole work.

To solve this problem, let us select the definitions used in scientific literature and in standards, which reveal the complex ST dependability property. Then let us check the applicability of certain wordings for different types of software programs, listed in All-Russian Classification of Products (OKP). According to the results of analysis we shall try to draw some conservative conclusions about the possibility of practical application of terms under consideration.

Before we proceed with the performance of the aboveindicated task, we should make out what is ST considered as the object of application of a traditional dependability methodology, which stipulates that it is necessary to study the dependability of a system as a whole depending on the reliability of its components. We should understand what a ST component is. It will help to take the proper direction to solve the problem.

The respective literature traditionally covers software systems, or software complexes, consisting of software modules, software programs and sub-programs. If to take a particular software program as an object of study, it may be considered, for instance, as a combination of sub-programs or other functionally complete units. But software modules and sub-programs are the software programs themselves: the above mentioned GOST 19781–90 standard puts an expression "software program" in the beginning of the definition of "software module" and "sub-program". Therefore, the study of these components does not change the view on the ST dependability issue.

It is proposed to consider ST as a set of functions they realize to control information systems. Then the element of the program shall be a separate function which controls the storage, processing, delivery of information and other work performed by IS. In addition, it is possible to rank the functions by their importance which may be useful for the study of a number of related issues (for example, for risk assessment). Consideration of separate software functions or information services may offer an advantage of definition of reliability indices of IS and software programs. It will be shown later that this approach is consistent with international standards and generalized concept of dependability in the sphere of computer sciences.

Terminology analysis

Let us consider several definitions of the main terms taken from different sources, to analyze their applicability for ST.

We shall start with the basic standard GOST 27.002–89 "Dependability in equipment. General principles. Terms and definitions" [4]. In accordance with this standard, dependability is defined as "the property of an object to maintain within the specified time the values of all parameters specific to the ability to perform the required functions under set modes and conditions of operation, maintenance, storage and transportation". This standard [4] also points out that depending on the object's purpose and application conditions, the dependability may include combinations of reliability, durability, maintainability and storability. It is an important note which will further come useful. We need to note here that the properties storability and durability as they are defined in [4] are apparently not essential for a wide range of ST. Perhaps, the notion of durability (together with the notion of a limiting state) can be applied for complicated software systems of real time. The definition of dependability itself could be called as "parametric", i.e. it is necessary to specify the parameters and their variation limits for the application of this definition. It is a quite difficult task for software programs, though it is still possible to solve. It seems that the main inconvenience of this dependability definition is statement of necessity "to maintain within the specified time", as it is probably related to the real time software systems only [5]. Hereafter we shall give some more definitions with the mentioned time intervals of functioning. Presumably it is connected with the fact that dependability of technical systems is often understood as reliability, but this word is related to a "probability of failure-free operation on a certain time interval". However, this measure is not always applicable for software programs.

The standard GOST 28195–89. "Quality control of software systems. General principles" [6] does not contain any definitions of terms, but there is a description of quality measures, ST dependability measures, reflecting the "ability of ST in certain application fields to carry out the specified functions in accordance with software documentation under conditions of deviations in the operational environment

caused by hardware, input data errors, service errors and other destabilizing effects". Here there is no consideration of the fact that software documents, as well as the software programs themselves, may contain errors. Moreover, the above listed deviations refer not to ST, but to the environment. However, the standard GOST 28806–90 "Software quality. Terms and definitions" [7] gives the definition of ST reliability in accordance with the title: "Set of properties representative of the ability of a software tool to maintain the specified level of performance availability in the required conditions during a specified time interval". Again we can see the statements about "specified time intervals", the meaning of which is rather doubtful for ST functioning. There is enough information about it in the ST reliability literature. It is interesting that this standard [7] provides a note right after the ST dependability definition, stipulating that "the number and mode of software tool failures … do not depend on time". And the level of performance availability we are speaking about in this definition of dependability is defined as a "degree of needs satisfaction represented by a certain number of values of software tool quality characteristics". I think the word "needs" is the most interesting here. Which needs are we speaking about? Let us assume these are the users' needs. Further, the attachments contain the information about sub-characteristics of ST reliability, which include completeness, fail safety and recoverability of ST. Fail safety and recoverability are defined through the possibility to maintain or recover the same "level of performance availability", and the definition of completeness contains a word expression "failure rate" which is not defined in the text. In short, these terms cause new questions.

We will follow up on the above with the consideration of the standard GOST R ISO/IEC 9126–93 "Information technology. Software product evaluation. Quality characteristics and guidelines for their use" [8]. This standard defines dependability as "a set of attributes related to the ability of software to keep the quality level of functioning under the specified conditions during a specified time period". Here again the questions of a "time period" occur especially with consideration of a note indicating that software programs are not exposed to "ageing". On the other hand, this note has a reference to another standard, which points that the given definition of dependability has been extended to the maintaining of "its quality level of functioning" instead of the "performance of a required function". Probably, it is essential for quality matters. An attachment to the standard [8] offers the reliability measures which are to a large extent comparable to the above mentioned measures from the standard [7]: recoverability, fault tolerance and stability.

Now let us check the definition of this term in other international standards. Let us start with IEEE Std 610.12–1990 "Glossary of Software Engineering Terminology" [9], which defines reliability as "the ability of a system or component to perform its required functions under stated conditions for a specified period of time". It could be understood as the "ability of a system or component to perform its required functions under stated conditions for an exactly specified time interval". Unlike a clear Russian definition of GOST 27.002–89, this term, as well as a number of other terms translated from English, has some kind of ambiguity (maybe it is just seeming) as to what the "stated conditions" do refer: either they mean the conditions of functioning of a component/a system, or they concern the parameters of the functions performed. Maybe, it is connected with the custom of defining reliability in a parametric way.

We have already considered the international standard related to ST quality above [8]. This standard has a later revision, consisting of several parts which shall be observed further in this paper. The standard [10] defines the reliability (when translated into Russian) as "the ST ability to maintain a specified level of performance under specified conditions". It is obvious that unlike the above mentioned definition from [8], there are no words about a "time period" here. Probably, this wording was added at the preparation of this document [8] to ensure coordination with a basic concept from GOST 27.002–89 or IEEE Std 610.12–1990 (it is just a supposition).

The listed definitions of ST dependability are by far not the complete list of possible variants. You can see other wordings in different materials. The most attention should be given to the definition drawn on the basis of a modern view. Let us point the definition given in paper [11]: "functional reliability is a set of properties determining the ability of software with an acceptable level of faultlessness, to perform a correct transformation of input data into the output results under the given conditions, keeping the output results within acceptable limits". The paper [11] also contains the list of major attributes of functional reliability, with "faultlessness" and "correctness" among them. It should be noted additionally that the attributes of functional reliability of software programs [11] also include the definition of "failure-free performance", which differs from the definition of "faultlessness". The point is that failure-free performance is interpreted as the ability of software programs not to cause functional failures of information systems, and the faultlessness as the ability of software programs "to function without faults". Basically, it is logical, as we can presume that it is not a software program that fails, but the information processing system controlled by this software program.

Here an important question arises again – whether it is necessary to consider the reliability of software programs as separate independent components of information systems, they are executed in. In the technical report dedicated to fundamental concepts of dependability [12], a group of international specialists performs a complex evaluation of dependability of computer-based systems. They use the term *dependability*, which is more generalized than the term *reliability,* and define it as the "ability to deliver service that can justifiably be trusted". For an alternative definition of dependability they give the rule allowing to check on whether it is possible to trust the provided maintenance, which could be understood as the "the ability of a system to avoid failures that are more frequent or more severe, and outage durations that are longer, than is acceptable to the

user(s)". In principle, this all can be used when considering the ST reliability. Among the properties of IS dependability, described in [12], we can distinguish maintainability – ability to undergo repairs and modifications and reliability – continuity of correct service. These properties have already been observed earlier, but they were rendered a bit differently.

Table of software system reliability properties

Let us now check which properties can describe the reliability of ST of different applications. Table 1 lists the properties attributable to the concept of reliability in accordance with the abovementioned definitions. The columns represent several software tools and information products of computer engineering as per the OKP classification. At the point of a line-column crossing a note is put indicating the possibility of inclusion of this property under the study of reliability of the ST concerned: «+» – this property is convenient to consider as one of the reliability components of the respective ST type, «-» – this property is not convenient to consider as one of the reliability components of the respective ST type, «+/-» – it is difficult assess this property for its applicability to the characteristic of reliability of the respective ST type.

Table 1

Property	Software tools									
	1)	2)	3)	4)	5)	6)	7)	8)	9)	10)
Reliability (G)	+	+	+	+	+	+	+	+	+	-
Maintainability	+	+	+/-	+	+	+	+	+	+	+
Fault tolerance	+	+	-	+/-	-	+	+/-	+	+	-
Stability	+	-	-	-	+/-	+	+/-	+/-	-	-
Faultlessness	+	+	+	+	+	+	+	+	+	-
Reliability	+	-	-	-	+	+	+	+/-	-	-
Availability for recovery	+	+	-	-	+/-	+	+	+/-	+/-	-
Maintainability (ALR)	+	+	-	-	+/-	+	+	+/-	+/-	-
Reliability (ALR)	+	+/-	-	+/-	+	+	+	+	-	-
Security	+	+	-	-	-	+	-	+	+	+
Testability	-	+	+	-	-	-	-	-	+	-
Safety	+	+/-	-	-	+/-	+	+	+/-	+/-	-
Confidentiality	+	+	-	-	-	+	-	+	+	+

Let us list the properties and the considered ST types of Table 1 with the determination of OKP codes.

We shall start with the properties list. *The property of reliability* (marked with "G" in Table 1) by GOST 27.002–89 [4] and *properties of maintainability, fault tolerance and stability* by GOST R ISO/IEC 9126–93 [8] are specified. Further, using the definitions from [11], *the property of faultlessness* – the ability of ST to perform without errors, *the property of reliability* – the ability of ST not to cause IS functional failures, *availability for recovery* – the ability of a software program to eliminate a software fault and to reset for a new execution and recovery of data in case of a functional failure, *the property of security* – the ability of ST to prevent from an unauthorized access to software programs and data and *the property of testability* – the property specifying the completeness and efficiency of the detection of errors in intermediate and output results. In addition, let us consider several properties taken from paper [12], although they are defined in a generalized sense and refer more to IS rather than to software programs: *the property of maintainability* (marked with "ALR" in Table 1) – the ability to modify and restore the operability, *the property of reliability* (marked with "ALR" in Table 1) – the continuity of correct operation, *the property of safety* – absence of disastrous effects for users and external systems and *the property of confidentiality* – absence of unauthorized disclosure.

Let us further enumerate the considered ST properties with the indication of OKP codes:

1. 50 1000 8 System software;
2. 50 2000 0 Software of general use;
3. 50 3000 3 Application software for research and development;
4. 50 4000 6 Application software for design;
5. 50 5100 2 Software for local microprocessor systems of process regulating and control;
6. 50 5200 6 Software for automated process control systems (APCS); 50 5300 3 Software for control of flexible computerized manufacturing systems (FMS);
7. 50 5400 3 Software for control of moving objects;
8. 50 5500 0 Software for automated workstations;
9. 50 6000 1 Application software to solve organizational and economic tasks;
10. 50 8000 7 Software and information products.

Conclusions and tasks for the future

By analyzing Table 1, we can conclude that the ST dependability includes several combinations of the considered properties, specified by the types of software programs. If to consider the whole set of ST, then, the universal properties which describe the dependability to the fullest extent, are the faultlessness and failure-free performance (this property can be considered in the wording of [12], and in the definition of [4], if it is possible to define all the requirements and to set all the parameters as this standard does stipulate). Additionally we shall underline the property of maintainability important for practice. This property is particularly necessary for ST intended to solve the control tasks. Table 1 shows that the properties of availability to recovery and maintainability are similar in applicability, though the wordings are different.

The consideration of Table 1 in columns let us conclude that ST of system purpose, APCS and FMS control can be characterized by all the above listed dependability attributes. It is explained by their complexity and importance for the IS and external systems operation.

The main task of this paper is to find an appropriate range of ST properties (or attributes), which can help to determine what is understood by the reliability of IS soft-

ware components. For this purpose different definitions of reliability have been considered, as well as the properties included into the definitions of this term. We have checked the possibility to apply this or that attribute under the description of reliability of different ST. It should be noted that we have not considered the measures required for the quantification of these properties. This is the task of future works. It is seen in the context of study of the system software reliability through the reliability of its elements which are separate functions for reception, handling, storage and output of information. In other words, the most attention should be focused on the ability of ST to perform functions it was developed for, as the ability of a software program to perform the required functions determines the system capability in general.

As the tasks for the processing and further construction of the table provided in this paper to show the correspondence of the reliability properties to different STs (Table 1), a more detailed consideration of classification and flexibility of attributes could be proposed.

References

1. **Netes V.A., Tarasiev Y.I., Shper V.L.** Topical issues of standardization of terminology in the field of reliability // Dependability. 2014. No. 2 (49). P. 116–119.

2. **Netes V.A., Tarasiev Y.I., Shper V.L.** How to define what is “reliability” // Dependability. 2014. No. 4 (51). P. 3–14.

3. **Potapov I.V.** Terms of reference of software reliability // Dependability. 2015. No. 1 (52). C. 53–57.

4. GOST 27.002–89. Industrial product dependability. General principles. Terms and definitions.

5. **Lipaev V.V.** Software reliability. M.: SINTEG, 1998. 232 p.

6. GOST 28195–89. Quality control of software systems. General principles.

7. GOST 28806–90. Software quality. Terms and definitions.

8. GOST R ISO/IEC 9126–93. Information technology. Software product evaluation. Quality characteristics and guidelines for their use.

9. IEEE Std 610.12–1990. IEEE Standard Glossary of Software Engineering Terminology.

10. ISO/IEC FDIS 9126–1:2000. Information technology – Software product quality – Part 1: Quality model.

11. **Shubinsky I.B.** Functional reliability of information systems. Methods of analysis. M.: LLC Journal “Dependability”, 2012. 296 p.

12. **Avizienis A., Laprie J.-C., Randell B.** Fundamental concepts of dependability. LAAS-CNRS, Technical Report N01145, Apr. 2001. P.21.