National Institute of
Advanced Industrial Science
and Technology
AIST

AIST-PS-2009-007

User Oriented Dependability

Yoshiki Kinoshita

Research Center for Verification and Semantics (CVS)

算譜科学研究速報
Programming Science
Technical Report

Research Center
for Verification and Semantics

独立行政法人 産業技術総合研究所 システム検証研究センター 発行
Published by Research Center for Verification and Semantics
National Institute of Advanced Industrial Science and Technology

User Oriented Dependability*

Yoshiki Kinoshita†

2009.7.14

Seeking for the proper concept of user orietend dependabiliy of information systems, I discuss on some topics which should be essential for it. They include the dependability of open systems, certification, risks and their reduction. I emphasise that my discussion is parameterised by the concrete attributes of dependability in question, as one must not restrict one's attention to a particular concrete attribute, such as safety or security, when one needs dependabilit of open systems.

1 Introduction

I seek for the concept of user oriented dependability of information systems, in the level which is independent of concrete attributes. I start from the observation that systems cannot be freed from risks and achievement of dependability is closely related to, or even reduced to risk reduction.

In the seminal paper [1], they discussed five attributes, i.e., reliability, availability, safety, integrity, maintainability, of dependability. Each attribute has its own own context of risks. Means for risk reduction accordingly has to be discussed for each attribute. I wish to discuss, however, about generality amongst those contexts of risks and means of risk reduction.

"Safety culture" is a good example. It is anotion introduced as a means for risk reduction in the context of safety, which says that there is no goal in the attempt of pursuit of safety and continuous effort of improving the situation is essentially important. But that does apply for other attributes such as reliability, availability, etc.

Starting from a very general, not seemingly meaningful definition of dependability, I try to extract general and useful thoughts on risks as well as risk reduction. Given a concrete attribute of dependability in concern, these thoughts can be instatiated so that concrete means of risk reduction is obtained. At least, such is what I try to produce as a result of this research.

User orientedness Dependability of information systems is often discussed from engineering point of view. However, engineers are not the only stakeholders of information systems; there are also the users and even notified bodies which issues various certifications for systems are quite involved in the dependability issues. Among these stakeholders, those who wishes the dependability to the largest extent is the user. In that sense, the notion of dependability is inherently user oriented. Yet I retain the word "user oriented" in order not to forget about the users' benefit in thinking about dependability.

---

* This paper is presented on the occasion of Dependable System Workshop held at Oonuma, Jul. 14–16, 2009.

† CVS, AIST

Life cycle One implication of user orientedness is the need for treatment of the whole life cycle of the system, rather than the system itself. Not only the development of the system matters, but acquisition process, maintainance, operation, and even disposal process play important roles to realise dependability.

Open system The user of information system wishes to be equipped with unexpected failure caused by unforeseen faults. Such is a problem of open system. One must accept the fact that there is no hope for complete enumeration of possible failures and faults.

Avizienis, Laprie, Randell and Landwehr analysed the basic concepts of dependable computing in their seminal paper [1]. In their analysis, I do not find any evidence that that they take, or at least took at the time of writing [1], an open system standpoint with respect to their thougt on dependable systems. Yet, they define dependability as "the ability to avoid service failures that are more frequent and more servere than is acceptable," and this does not put any restriction on failures. So, the idea of open system does not contradict with their analysis, although it seems such an idea did not exist in the authors of [1] at the time of writing it.

We shall investigate the notion of dependability, taking the open system point of view.

Risk reduction According to the above definition of dependability, to gain dependability means to reduce risk or to cut down the possibility of something bad happening in the future, rather than to increase the possibility of something good. Systems have to live with risks. Risks cannot be completely eliminated from information systems. I have to accept it and the question is how I can reduce risks and to what extent.

Abstract means for dependability Under open system standpoint, there is no hope for complete enumeration of failures and faults. Therefore, the means for realisation of dependability under open system standpoint is generic with respect to concrete attributes one wishes to obtain, e.g., safety and reliability. The means for dependability is abstract in that sense.

Generic and abstract means, however, can be discussed without going into the details of the attributes to be realised. For instance, "safety culture" is discussed in the realisation of safety of systems, where it is argued that there is no goal of safety achievement but the continuous effort is the essense. But such is valid not only in realisation of safety but also in realisation of reliability and other attributes. I shall be interested in investigatigation of means at such abstract level.

Certification Going back to the definition of dependability in [1], it uses the word "acceptability," which leads to a relativity in dependability. The notion of integrity level of systems comes in here. More imporant is that users do not have professional skill as for evaluation of information systems, or they do not wish to spend the cost for evaluation, so they need an assessment of the system by an authoritative body or person; they may even need certification of assessment in some cases. Assessment is being done in many contexts, so it would be valuable to clarify the whole action, with a list of stakeholders.

2 Dependability of open systems

I begin with the notion of open systems and its dependability.

2.1 Open systems

The notion of "open systems" was introduced in phisics and chemistry, and nowadays also used in informatics context[2]. These two uses of the word, however, seems to be different, although there is a common spirit in both. In phisics and chemistry, an open system is defined to be a system where there is a flow of matter or energy between itself and the world outside it.

Likewise, an open information system may be defined to be a system where there is a flow of information in and out of itself. But it does not seem to be enough; not only the information manipulated by the system but also the information about the system, that is, the description of the system is also subject to change from time to time. Open information system comes with uncertainty, as its own description is subject to change. I do not go more into detail here as for the definition of open information systems, which deserves another place for investigation, but uncertainty seems to be a key to the notion.

2.2 Uncertainty

As for uncertainty of an information system, there are at least three different properties which should be clarified before going futher. These three kinds are nondeterminism, incompleteness and indefiniteness.

Nondeterminism (Pascal, Bayes, Zadeh)
: Nondeterminism definitely captures at least a part of the notion of uncertainty. There are several kinds of nondeterminisms in practice, like probability, degree of belief (Bayesian interpretation of probability), fuzzyness. In these cases, however, these notions for nondeterminism are treated as objects of a theory; there are theories which describes the complete properties of these notions. In this case, uncertainty is thoroughly controlled by the theory which describes them.

Incompleteness
: The (first) incompleteness theorem by Gödel says that every consistent effectively generated formal theory that proves theorems in Peano arithmetic has an arithmetical statement that is true but not provable in it. The theory does not have the thorough control in this case and in that sense, there is a kind of uncertainty here. This, however, is not very important because such a pathological example as a true but unprovable proposition appears only in a very complicated cases and does rarely appear in information processing.

Indefiniteness (Kripke)
: The kind of uncertainty most relevant to the current situation is indefiniteness, which Kripke discussed in his seminal plus-quas example. His point is essentially that the interpretation of every description may differ from person to person so there is always a room of misunderstanding with each other.

2.3 Definition of dependability

Extensive attempts has been taken in [1], for instance, on the notion of dependability from the aspect of attributes, threats, means to reduce threats. From open systems standpoint, however, such analytic approach is not very effective because, as soon as a list of attributes for dependability is given for example,

an attribute which is not on the list could be given immediately as a counterexample for the adequateness of the list. There are infinite number of attributes for dependability and it seems one cannot list up all the attributes; this is one of the implications of open system view on information systems.

I take the open system view on information systems. Therefore, I leave the problem of enumerating all attributes for dependability to other opportunity; I shall take a parameter which varies among the attributes of dependability and I concentrate upon investigation of the general method of risk reduction, whatever the concrete attribute would be. I expect there should be some generality amongst the methods of risk reduction for each concrete attribute such as safety and security.

So, my (tentative) definition of a dependable system is "a system being as it should be." This is quite an abstract, general definition, yet there seems to be a room for investigatigation of a means for realisation of such a system, i.e., a common means amongst the means for realisation of concrete attributes in question. I could also discuss about the origin of those risks, about the situation where there is no risk at all and the way leading to such an ideal situation but, in this paper, I shall briefly touch on the last of these three topics.

3 Risks

In the introduction, I saw that to obtain dependability is to reduce risks. Also I saw the possibility of talking about generic risks which can be instantiated to each attribute for dependability if necessary. In this section, I discuss about risks of information systems. The discussion should be in four steps:

1. what are risks
2. what are causes of risks
3. what are risk reduction
4. means for risk reduction

In this paper, however, I discuss only about 1. and 4., as I need more thoughts for other steps.

3.1 What are risks

Risks of information systems are called threats in [1] and hazards in the context of safety. There are many attempts to classify risks, but they are always for a concrete attribute of dependability; I try, however, to discuss risks in so abstract level that the concrete attribute does not matter.

In that abstract level, there are at least four kinds of risks for information systems: development, aging, malfunctioning and disposal. Surprisingly, most risks I think of in practice can be classified into one of these four.

Development Development of systems is a risk, as it can create many kinds of causes of failures. Every bug and security hole is embedded during the development; only, the developer does it without knowing.

Aging Aging is a risk. For instance, improper processing of data queues which creates inaccessible memory blocks tends to become observable as the system continues to operate for a long time. It

is aging of software, not of hardware. The fault may simply be a bug which cannot be detected easily, but the situation could be more complicated when hardware issue is involved. Hardware of course changes its behaviour as it becomes older, and even if it is only a change of behaviour within the error range, the software may not allow it, and the combination of the hardware and software may cause a failure. Such a case would be something on the border between a bug and non-bug.

Malfunctioning Malfunctioning is a risk. Malfunctioning of a system is often the way how its bugs are found. It is not obvious to relate the malfunctioning phenomenon and its cause in the software of the system or its lifecycle process; such diagnostic process for systems could have an analogy with medical process for human being.

Displosal Disposal is a risk which occurs at the last phase of a system's life cycle. Many problems concerning systems may occur at disposal. For instance, the data in the computer disk to be disposed must be carefully erased in order to keep the confidential information of the user of the information system. Another thing to consider is the inheritance of data and operation of the system.

I could further add the following four kinds.

- Being unable to obtain what is necessary.
- Being connected to another system with bad interface.
- Being disconnected from another system with good interface.
- Other risks caused by the running of the system.

3.2 Means to reduce risks

I propose eight generic means for risk reduction. These means are intended for stakeholders of the system in concern. The adequateness of this list, however, must be validated in further work; there may be other generic means which should be added to this list.

Right view Correct understanding of the system and its environment provides a correct start for an attempt of risk reduction. Such understanding could be achieved, for instance, by building a mathematical model of the system. Being mathematical or not, one needs a precise, logically consistent model.

Right intention The stakeholders must intend to reduce risks explicitly and clearly because nothing would be achieved without indending it explicitly and clearly. An explicit and precise plan for building and running the system should be made, even though there would be no complete specification which needs no later modification. Asking for specification does not contradict with the open system view on systems if inadequateness and inconsistency of specification are also taken into account by the stakeholders.

Right speech It is well-known that the communication between the stakeholders of system is the sink of risks. So, using the language in a right way is essential to the reduction of problems around

systems. This includes the precise use of words in the professional communication as well as selection and use of the proper vocabulary.

Right action The operation of the system must be conducted in the way that the designer of the system specifies.

Right livelihood Every system has its accountance matter. There should be no doubt in the finance of the system, i.e., the income for the project for running the system should be obtained in a right way. Otherwise, the whole system would get into trouble sooner or later. This item is tightly connected to integrity of system.

Right effort Dependability is obtained through continuous effort for improvement. It is typical in "safety culture" where people say there is no goal in safety improvement. There should be no stop. Dependability related attributes, safety for instance, are not something to be achieved, but always to be made better. It is not the matter of result but of the whole process.

Right mindfulness Mindfulness (delivering the attention) does not only contribute to improve the system not only from an effectivity point of view, but also from esthetic point of view.

Right concentration The whole system, including the people working around the system, should work smoothly without noise, not only in physical sense but also in abstract sense. If a system works somewhat in a ragging way or gawky manner, there would be higher risks.

4 Certification and Assessment

Certification is an act that an assessor makes a judgment of whether or not a certificate about properties of the assessed person or thing, hereafter simply called the assessed, is to be issued. It is initiated by certification application; the person who applies for certification is the certification applicant. The certification applicant is asked for a documentation about the assessed. The documentation contains the information required for assessment. In addition to the assessor, there is also an evaluator whose role is to evaluate the assessed (therefore, the assessed could also called the evaluated) as for the attribute in concern by inspecting the documentation and the assessed itself, and to produce an evaluation report which is a result of this evaluation process. An evaluation report is expected to be given to the assessor and contains the data required for judgment of whether the certificate is to be issued. The evaluation report may be written quantitatively or qualitatively and, even if it is written in a quantitative manner, multiple quantitative measures are often used in the evaluation report. The assessor examines the evaluation report and makes accordingly the judgment whether or not the certificate is to be issued; this final process is called assessment.

Example 4.1 (Entrance Examination)
One of familiar examples of certification is entrance examinations of schools. In this case, the asseossor is the school or the committee formed for the entrance examination and the assessed person is the candidate; in this case, the candidate is also the certification applicant. The judgment is made whether the candidate is to enter the school or not. The evaluator is a group of teachers or the committee formed for marking the exam paper. The documentation is the exam paper. The evaluation is the process of

marking the exam papers. There is usually an exam for each subject so the result of assessment, i.e., the assessment report consists of a set of marks; each mark provides a quantitative measure and the tuple consisting of all marks provides a nonquantitative measure. The assessor examines the evaluation report and makes the final judgment for each candidate. There are a number of ways for the assessment. A typical approach is to take the sum of each marks or, equivalently, the average of marks. Some approach may firstly drop off candidates who get less mark than the determined limit in some subject and then takes the best candidates in terms of the sum or mark. The role of assessor is to select one of these possitilibies.

Example 4.2 (due to S. Matsuoka)
Another example would be type approval in legal metrology. There, the assessor is the notified body, e.g., the accredited body in charge of type approval. The assessed is a measurement instrument type and the manufacturer who designs the instrument type is the certification applicant. The manufacturer gives the type approval documentation, which corresponds the documentation in my terminology, to the notified body where testing is often carried on. So, the assessment process is type approval examination process and its result, i.e., the judgment whether the type in concern is to be approved or not, is the type approval certificate. The assessor is the person in charge of it, e.g., the head of the notified body.

Example 4.3 (due to A. Kishimoto)
In the case of chemical risk assessment, the assessor is the regulatory body and the assessed is a chemical substance which the manufacturer, often being the certification applicant, asks for approval of commercial distribution. The documentation is that to be supplied for the proof of safety; the manufacturer gives it to the regulatory body. The evaluator is the subcommittee formed by the appointed experts. The evaluation done by the subcommittee usually takes form of a draft report. Its results, i.e., the level of risk, may either be quantitative or qualitative. The judgment is whether the chemical substance in concern is to be approved or not for commercial distribution. The judgment is usually made by the upper committee.

The actions of evaluation and assessment can further be detailed. Evaluation can be divided into validation, verification and testing. Validation is the process of checking whether the documentation really contains the information required for evaluation. For verification, one checks the correctness of the contents of documentation. Testing is the examination of the assessed to see whether it has the property which the documentation declares is owned by the assessed in the documentation.

In other contexts, assessment is a process of estimating the value (size, strength, probability, etc.) of some property (risk (benefit), safety, etc.)

Acknowledgment

This work has been done as a part of the project “User Oriented Dependability” which the author is conducting as a theme of CREST DEOS (Dependable Embedded Operating System) programme. The author awes much to many discussions with the project members and the participants of other projects

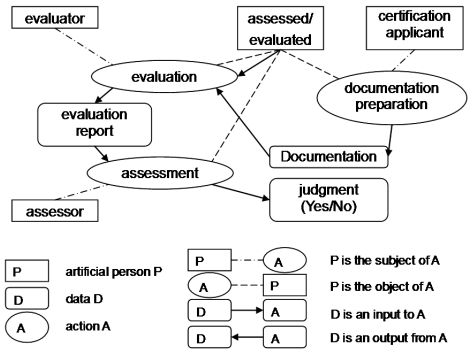

Figure1

in DEOS programme. In particular, Toshinori Takai and Makoto Takeyama read a draft of this paper carefully and gave many valuable and encouraging comments. Mario Tokoro gave the author a comment about open sysem standpoint. The author, however, is responsible for any errors, mistakes or faults remaining in this paper.

References

[1] Avizienis, A., Laprie, J.-C., Randell, B., and Landwehr, C.: Basic Concepts and Taxonomy of Dependable and Secure Computing, *IEEE Transactions on Dependable and Secure Computing*, Vol. 1(2004), pp. 11–33.

[2] Tokoro, M.(ed.): *Open system science*, NTT publications, 2009. in Japanese.

利用者指向ディペンダビリティ (in English)
(算譜科学研究速報)
発行日：2009 年 7 月 13 日
編集・発行：独立行政法人 産業技術総合研究所 (システム検証研究センター)
同連絡先：〒560-0083 大阪府豊中市新千里西町 1-2-14 三井住友海上千里ビル 5F
TEL：06-4863-5025
e-mail：informatics-inquiry@m.aist.go.jp

User Oriented Dependability
(Programming Science Technical Report)
13 July 2009
(Research Center for Verification and Semantics (CVS))
National Institute of Advanced Industrial Science and Technology (AIST)
5F Mitsui Sumitomo Kaijo Senri Bldg., 1-2-14, Shinsenrinishi-machi, Toyonaka, Osaka 560-0083 Japan
TEL：+81-6-4863-5025
e-mail：informatics-inquiry@m.aist.go.jp

AIST01-J00022-97