Software quantitative safety analysis methodology using quality sub-factors in a metro microprocessor signaling system

J. Batista Camargo Jnr & J. E. Proneça Páscoa
Safety Analysis Group -GAS, Department of Computer Engineering and Digital Systems-PCS, Escola Politecnica da Universidade de São Paulo, Brazil.

Abstract

The purpose of this paper is to present the application of a software quantitative safety analysis to a metro signaling system. The first step is to determine the software quality sub-factors that can adequately represent the software safety quality factor. For this purpose, this work is based on the ISO/IEC 9126 Standard - Information Technology - Software Product Quality and on the ISO/IEC 14598 Standard - Software Engineering - Product Evaluation. The chosen quality sub-factors that represent the safety quality factor are: accuracy, security, maturity, completeness, fault/error tolerance, understandability, simplicity, consistency, auto-description, time behavior, analyzability, testability, modularity and traceability. The last step corresponds to the application of some of this safety sub-factor metrics over a metro signaling system implemented with microprocessor technology. Finally, some considerations are made in order to associate the desired safety level of the signaling system with the expected values of the software quality sub-factors adopted.

1 Introduction

In respect to the software system safety the software engineering makes look like to be currently in a primitive period of training. The enormous flexibility of software opposes it the techniques that they look to become software safest and hinders the development of absolutely safe software. In the biggest part of the times, the software safety still is seen as only one of the factors that compose a model of quality of software.
The models of quality of software, that had appeared in the decade of 1970, still they had not been unified. Different approaches of quality of software published in norms, standards or articles or used internally inside of the organizations exist currently. More recently, some models of software safety have been published. In the current period of training, however, these models of software safety are equivalent the models of quality of software modified to be applied in the safety of specific systems, such as atomic plants, military weapons, etc.

The objective of this work is to propose of a model of software safety that can be applied the critical systems. Basically, the model include: the definition of a set of factors and sub-factors that can be used to evaluate the safety of a product of critical software; the selection of a set of metric that allow to evaluate quantitatively the safety of the critical system; a method of evaluation of the factors and sub-factors from metrics and a method of evaluation of the safety of the product of software from the factors and the sub-factors.

The term factor is used in this work to assign to a feature or attribute that can be used to evaluate the safety or the quality of a software product. The norms and the consulted bibliography are not consistent how much to the terminology used in each publication [Ol]. In this work, the diverse nomenclatures consistently had been unified for the terms factor and sub-factor.

2 The Proposed Software Safety Model

This chapter describes the software safety model adopted in this work. A hierarchy of factors and sub-factors of quality related with the safety will form this model

2.1 The Methodology Used for the Selection of the Safety Factors

This section summarises the methodology adopted in this chapter to select the factors and sub-factors of quality, of a software product, that can serve for the evaluation of the safety of a product of critical software. In summary, the methodology will follow the following stages:

1) Analysis of the factors of the quality and safety models
   a) Selection of the factors related with safety
   b) Discarding of the factors not related with safety
   c) Reclassification of some factors as sub-factors

2) Analysis of the sub-factors of the quality and safety models
   a) Selection of the sub-factors related with safety
   b) Discarding of the sub-factors not related with safety
   c) Classification of the sub-factors into factors

3) Combination of the selected factors and sub-factors as candidates

4) Smoothing of the factors and sub-factors candidates forming the final model.
Initially it will be analysed, individually, each factor that is part of the quality and safety models. In the stage, it will be selected, through justifications, the factors related with the system safety candidates to be part of the final safety model considered in the end of this paper. In the stage 1b, the factors not related with the safety will be discarded. In the stage 1c, some of the factors of the considered quality models will be reclassified as sub-factors. The result of stage 1 is a set of factors candidates as safety factors that are part of the safety model. In stage 2, the process of selection of factors will be repeated to select the sub-factors candidates to be part of the final safety model. Each sub-factor that is part of the considered models will be analysed individually. In the stage 2a, it will be selected, through justifications, the sub-factors related with the safety as candidates to be part of the safety model.

In the stage 2b, the sub-factors not related with the safety will be discarded. In the stage 2c, the safety sub-factors will be classified as being part of a safety factor. The result of stage 2 is a set of sub-factors candidates as safety sub-factors that are part of the final safety model. In stage 3, the factors candidates to be part of the final safety model, resultant of stage 1, will be matched with the safety sub-factors candidates of stage 2, forming a hierarchy of factors and sub-factors candidates to be part of the final safety model.

Finally, in stage 4, the hierarchy of factors and sub-factors candidates, resultant of stage 3, will suffer some smoothing to form the final hierarchy of factors and sub-factors that will compose the final safety model. Other not enclosed factors or sub-factors in the analyzed models of quality could be added. These upgrades must keep the compatibility with the analyzed models, without entering in conflict with the norms that serve of base for this work.

2.2 Selection of the Safety Factors and Sub-factors

Before initiating the selection of the safety factors and sub-factors, it is verified that the considered quality and safety models include similar quality factors. It must be noticed that it does not have consensus related with the terminology.

2.2.1 Selection of the Factors

In principle, the factors that already are enclosed in the safety models, will be considered important for this work, therefore already they had been considered important by the respective authors. On the other hand, this criterion does not apply the factors of the quality models that are not enclosed in the safety models. The pre-selected factors as candidates of the safety factors, are the following ones: Accuracy, Capacity, Reliability, Efficiency, Functionality, Integrity, Inviolability, Maintainability, Portability, Reviseability, Robustness, Safety, and Usability. [02]

2.2.2 Discarding of Factors

The following quality factors had been discarded in:

Effectiveness: This factor is not related with safety. It refers to the degree of accuracy and completeness that the user reaches its objectives.
Flexibility: This factor is not related with safety. In the truth, when adding different possibilities of use, the factor flexibility can compromise simplicity, reliability and safety.

Interoperability: This factor is not related with the safety. In the truth, when adding different possibilities of operation matched with other systems, the factor interoperability can compromise simplicity, reliability and safety.

Portability: This factor is not related with the safety. In the truth, when adding different environments of use, the portability factor can compromise simplicity, reliability and safety. This factor is enclosed in the safety model of the Canadian norm CE-1001-Std only as a secondary objective.

Productivity: This factor is not related with the safety. It refers to the effort, time and money to reach the objectives.

Reusability: This factor is not related with the safety. In the truth, if the software component add different possibilities of use, the reusability factor can compromise simplicity, reliability and safety.

Satisfaction: This factor is not related with the safety. One mentions the attitude and feelings to it of the user in relation to the software product.

2.2.3 Reclassification of the Factors

In this stage of the methodology, with the objective to form the hierarchy of factors and sub-factors had been reclassified as sub-factors.

Accuracy: This factor will be reclassified as sub-factor of the functionality factor. In the truth, accuracy already is a sub-factor of the factor functionality, in model ISO/IEC 9126.

Capacity: This factor will be reclassified as sub-factor of the functionality factor. Correctness: This factor will be reclassified as sub-factor of the factor reliability. Integrity: This factor will be reclassified as sub-factor of the functionality factor. Inviolability: This factor will be reclassified as sub-factor of the functionality factor.

Reviewability: This factor will be reclassified as sub-factor of the reliability factor. Robustness: This factor will be reclassified as sub-factor of the reliability factor.

Safety: Safety is a factor in the vision of quality in use of model ISO/IEC 9126. Testability: In principle, critical software must intensively be tested. Tests of unit, integration, system and acceptance must be made and validation. This factor will be reclassified as sub-factor of the maintainability factor.

2.2.4 Analysis of the Sub-factors

The process used above analyzing the factors used for the safety and quality models can be repeated for the sub-factors.

2.2.5 Selection of the Sub-factors

The sub-factors candidates the safety sub-factors are sub-factors that are related with the safety, but that still they had not been definitively enclosed in the final safety model. With the objective to form the hierarchy of factors and sub-factors
the following sub-factors will be fit inside of determined factors. The sub-factors candidates the safety sub-factors are related following, together with justifications:

**Accuracy:** A software component that calculates parameters of actuators or inexact outputs can lead the system the unsafe states. This sub-factor will be part of the factor functionality, as it occurs in model ISO/IEC 9126.

**Analysability:** The diagnosis of the causes of an eventual bad functioning and the identification of the software component that it needs to be corrected are basic for the increase of the safety of a critical system. This sub-factor will be part of the factor maintainability, as it occurs in model ISO/IEC 9126.

**Auto description:** This sub-factor is important for the agreement of the functioning of a component of critical software, with the objective to verify it and to diminish the possibility of introduction of new errors in future maintenance. Normally, this sub-factor would have to be part of the factor maintainability. However, under the cited point of view, this sub-factor will be classified as being part of the factor reliability.

**Completeness:** A necessary component of software to be full specified and implemented to hinder that defaults take the system the unsafe states. This sub-factor will make part of the factor reliability.

**Temporal Behaviour:** In the critical systems, this sub-factor generally is associated timeouts, or either, the computational system delay excessively to act. However, this sub-factor also considers premature actions. For example, a valve can requires a performance in a specific interval of time, nor very early, nor very late. This sub-factor will be part of the factor efficiency, as it occurs in model ISO/IEC 9126. Standardised Communication: The use of types of data and not standardised structures of data can originate errors that lead the system unsafe situations. This sub-factor will be part of the factor reliability.

**Concision:** It is a basic aspect of critical software, therefore how much lesser it will be the size of the software components, more easy it will be the agreement and the verification of the safety of the system. This sub-factor will be part of the reliability factor.

**Consistency:** The components of the software product must be projected and implemented in mode uniform to facilitate the detention of errors that can take the system unsafe situations. This sub-factor will be part of the factor reliability.

**Standardised data:** The use of types of data and not standardised structures of data can originate errors that lead the system unsafe situations. This sub-factor will be part of the factor reliability.

**Efficiency of execution:** This sub-factor affects the timing that generally is related with the safety. This sub-factor will be part of the factor efficiency, as it occurs in model ISO/IEC 9126.

**Stability:** The immunity of the software components the corrections or modifications in other components is important for critical software. The main
Objective is to prevent new errors introduced for corrections. This sub-factor will be part of the maintainability factor, as it occurs in model ISO/IEC 9126.

**Instrumentation:** The auto verifiability mechanisms help to detect errors in critical software facilitating to the debugging of software and the tolerance the errors. This sub-factor will be part of the maintainability factor.

**Understandability:** This sub-factor is important for the user [03], for the development of software [04] and for the maintenance of software. In principle, the agreement easiness reduces the probability of errors in the use, development or maintenance. This sub-factor will be part of the factor usability, as it occurs in model ISO/IEC 9126.

**Inviolability:** The access not authorised to the system and the hard link with integrity of the information can take the system inadvertently the dangerous states or for bad faith. This sub-factor will be part of the factor functionality, as it occurs in model ISO/IEC 9126.

**Modularity:** This sub-factor magnifies the safety of software when facilitating the phase of individual tests, integration and system, as well as facilitating to the validation and the maintenance of software. This sub-factor will be part of the maintainability factor.

**Traceability:** A necessary component of software to be traceable to allow its verification in relation to the safety. This sub-factor will be part of the factor reliability.

**Recoverability:** This sub-factor is similar the availability and is related with the time that follows the occurrence of a bad functioning, which can include the deactivation total of critical software. The capacity of the product of software commit the level of specified performance and to recoup the data, in feed case, is important for the safety, particularly in protection systems. This sub-factor will be part of the factor reliability, as it occurs in model ISO/IEC 9126.

**Simplicity:** It is a basic aspect of critical software, therefore how much bigger it will be the complexity of software, greater will be the difficulty to verify the safety of the system. This sub-factor will be part of the factor reliability.

**Testability:** This sub-factor will be part of the maintainability factor, as it occurs in model ISO/IEC 9126.

**Error Tolerance:** Errors must be tolerated therefore, in the practical one, are very difficult to prove the inexistence of errors in the software product. This sub-factor will be part of the factor reliability, as it occurs in model ISO/IEC 9126.

**Failure Tolerance:** Failures must be tolerated therefore, in the practical one, are very difficult to prove the inexistence of failures the software product. This sub-factor does not eliminate the correction necessity, only tries complements it. This sub-factor will be part of the factor reliability, as it occurs in model ISO/IEC 9126.

### 2.2.6 Discarding of Sub-factors

The following sub-factors of quality had been discarded for shown ratios below:
Adaptability: This sub-factor is not related with the safety. In the truth, the adaptations of critical software must be prevented and in case that they are necessary, a re-evaluation of software is necessary.

Adequacy: This sub-factor is not related with the safety of the system.

Learnability: This sub-factor is not related with the software safety.

Attractiveness: This sub-factor is not related with the safety.

Auditability: This sub-factor is on the operational procedures to be adopted of that to the safety of the system.

Capacity to substitute: This sub-factor is not related with the safety. In the reality, one expects that the substitution or update of critical software be made very little times.

Coexistence: This sub-factor is not related with the safety. In the reality, the sharing of computational features between the product of critical software and other products of software, is total inadvisable.

Conformity: The model of norm ISO/IEC 9126 defines six features as being conformity with standards, conventions, regulations or laws related with each one of the six features that the model defines. Of the point of view of the software safety, these sub-factors are very generic. Therefore, these six sub-factors will be discarded.

Expandability: This sub-factor is not related with the safety. In the truth, the expansion capacity can stimulate modifications in software that can compromise simplicity, reliability and safety of critical software.

Generality: This sub-factor is not related with the safety. In the truth, the excess of possibilities of use of the software components magnifies the complexity and the probability of errors, which can compromise the reliability and the safety of critical software.

Independence of the hardware: This sub-factor is not related with the safety, therefore each implementation must be re-evaluated.

Independence of software: This sub-factor is not related with the safety, therefore each implementation must be re-evaluated.

Instalability: This sub-factor is not related with the safety. One expects that the installation of critical software be made very little times.

Interoperability: This sub-factor is not related with the safety. In the truth, when adding different possibilities of operation matched with other systems, the sub-factor interoperability can compromise simplicity, reliability and safety.

Maturity: The sub-factor maturity is excessively generic for the safety analysis. The description of norm ISO/IEC 9126 is very similar to the software failure tolerance. In this work, it will be assumed that the maturity concept is included in the failure tolerance, recoverability, completeness and correctness. Therefore, as this sub-factor is redundant for this work, it he will be discarded.

Changeability: This sub-factor will be discarded despite being one of the attributes of the safety model CE-1001-Std. In the case of critical software, the
amount of cares that must be taken in each modification dims this sub-factor, that represents the easiness with that modifications can be incorporated the software product. In principle, the amount of modifications applied to a product of critical software must be reduced, remaining only the modifications related with corrections and the addition of safety mechanisms.

Operability: This sub-factor is not related with safety of critical software, if well that it can be related with the safety of the system.

Training: This sub-factor is not related with the safety of critical software, if well that it can be related with the safety of the system.

Resources Use: This sub-factor is not related with the safety. The efficiency in the use of resources is on to the cost of the computational system. It must be noticed that another sub-factor, temporal behaviour, was defined separately and are important for the safety of the system.

2.2.7 Classification of the Sub-factors

In this stage of the methodology, with the objective of forms the hierarchy of factors and sub-factors that will be proposal the following sub-factors of safety will be fit as being part of some of the safety factors. The enclosed sub-factors in the previous stage related with the software safety and that they had still not been classified are the following ones:

Structure: This sub-factor will be classified as being part of the factor maintainability.

Predictability: This sub-factor will be classified as being part of the factor reliability. Verifiability: This sub-factor will be classified as being part of the factor reliability.

2.2.8 Factors and Sub-factors Candidates

The proposal of the software safety model will be made in a hierarchic form of factors and sub-factors keeping the maximum possible coherence with the analysed norms of quality in the previous sections. In this stage of the described methodology in the beginning of this chapter, the factors candidates will be matched with the sub-factors candidates to be part of the safety guard model that will be considered in the following stage.

2.2.9 Smoothing of the software Safety Model

Considered above of factors and sub-factors candidates to be used in the proposed safety model, can be simplified through the consolidation of the following sub-factors:

Robustness: This sub-factor is part of the model of CE-1001-Std quality and of the safety model of Lawrence.

Error Tolerance: This sub-factor is part of the quality model of McCall.

Failure Tolerance: This sub-factor is part of the quality model of ISO/IEC 9126. The sub-factors robustness, failure tolerances and error tolerance are equivalents, therefore all indicate the capacity of the system to continue to satisfy the requirements same on the occurrence of failures. Therefore, it is enough to keep only one of them. The sub-factor failure tolerance will be kept.
Auto-description: was defined as one of the metric ones of the model of McCall. In the stage auto-description was classified especially as one of the sub-factors of the factor reliability.

Reviewability: was defined as one of the primary objectives of quality for the norm of Canadian safety CE-1001-Std. It was reclassified as sub-factor of the factor reliability.

Verifiability: was defined as one of the attributes of quality for the norm of Canadian safety CE-1001-Std. It was classified as sub-factor of the factor reliability.

The Canadian norm CE-1001-Std also defines the Reviewability and Verifiability factors. As the factor reviewability had been reclassified as sub-factor, the two sub-factors had been equivalents. The metric auto-description of the model of McCall was interpreted in this work as a feature stand by for the verifiability. Therefore, it is enough to keep the sub-factor verifiability.

Integrity: was defined as one of the factors of quality of the simplified model. Also it was defined as one of the factors of the model of McCall. Integrity was reclassified as sub-factor of the factor functionality.

Inviolability: Inviolability was defined as one of the metric ones of the model of McCall. Also it was defined as one of the features of the model of norm ISO/IEC 9126. It was defined still as one of the qualities of the model of Lawrence. Inviolability was reclassified as sub-factor of the factor functionality. The sub-factors integrity and inviolability are equivalents, therefore both indicate the capacity of the system to resist the attempts of access not authorised to the system or the data in it contained. Therefore, it is enough to keep only one of them. The sub-factor inviolability will be kept, in ahead.

Concision: Concision was defined as one of the metric ones of the model of McCall. In the stage to, concision was classified as sub-factor of the factor reliability.

Simplicity: Simplicity was defined as one of the metric ones of the model of McCall. In the stage to, simplicity was classified as sub-factor of the factor reliability.

Of the point of view of system safety, the sub-factor concision implies in sub-factor simplicity and, therefore, these sub-factors are equivalents. Consequently, it is enough to keep only one of them. The sub-factor simplicity will be kept in ahead.

Efficiency of execution: Efficiency was defined as one of the quality factors and efficiency of execution was defined as a metric one of the model of McCall. Efficiency also was defined as one of the features of quality of model ISO/IEC 9126. Moreover, efficiency was defined as one of the secondary objectives of quality of the CE-1001-Std norm. So far, efficiency of execution was kept as a sub-factor of the factor efficiency.
Temporal behaviour: This sub-factor was defined as one of the features of quality of model ISO/IEC 9126. It was classified as a sub-factor of the factor efficiency.

The concept of efficiency of execution covers the concept of temporal behaviour. Although to cover the timing, the concept of efficiency of execution is on to the economy of features, which is less important of the point of view of the safety. Therefore, the sub-factor efficiency of execution will be discarded and the sub-factor temporal behaviour will be kept in the proposal of the final software safety model. After the consolidation above, the final stack of factors and sub-factors of safety is the following one, shown in below:

**Functionality:** Accuracy, Capacity, and Inviolability;

**Reliability:** Completeness, Standard Communication, Consistency, Correctness, Standard Data, Previewability, Traceability, Recoverability, Simplicity, Failure Tolerance, and Verifiability.

**Usability:** Learnability

**Efficiency:** Temporal Behaviour

**Maintainability:** Analysability, Stability, Structure, Instrumentation, Modularity, and Testability.

**Quality in Use:** Safety Mechanism

### 3 Conclusion

At moment, some of these sub-factors have been applied in the São Paulo metro signalling system that is implemented with microprocessor technology. It can be said the use of these factors and sub-factors representing the safety quality factor has been extremely useful and being constituted as reference guide in the development of new systems. Some metrics is now being selected to make the results more concrete. Much more research must be made in this field with the intention to reach a Software Safety Model. Based on this model, a qualitatively and quantitatively evaluation can be better developed.

### References


