Research on Software Quality Characteristics based on MBSE System

Jiangtao Che, Yangyang Zhang*, Wenpeng Li, Jianxun Guo

China Electronics Standardization Institute, Beijing, China

Abstract

MBSE (Model-based Systems Engineering) is an engineering method that uses graphical models to describe and design complex systems, which has the advantages of unambiguous knowledge representation, reusability of models and integration of system design. Based on MBSE, through modeling and simulation, the quality characteristics of product quality such as functionality, reliability and testability are improved. The important significance of MBSE is that it provides new ideas and methods for the software engineering field, helps to improve the quality and development efficiency of software products, and has broad development prospects in the software engineering field.

Keywords

MBSE, software quality, product quality mode, quality standard

1. Introduction

In the early stage of system engineering, the information generated by the system is described and recorded in the form of documents. However, with the continuous improvement of the scale and complexity of the system, the shortcomings and deficiencies of the traditional document-based system design method are becoming more and more prominent, such as inaccurate information representation, easy to generate ambiguity, difficult to find the required information from massive documents, and unable to connect with other engineering designs[1].

With the rapid development of computer and information technology and engineering technology in various fields, it is becoming easier and easier to describe the system with object-oriented, graphical and visual system modeling language, and the application proportion of model in the system development work is also increasing. Model based systems engineering (MBSE) came into being. MBSE method can effectively solve the problems of document-based system engineering methods in parameter acquisition and technical state management, and is a powerful tool to effectively deal with system complexity.

In 2007, the International Council on Systems Engineering (INCOSE) defined MBSE in its Systems Engineering Vision 2020 as "Model-based Systems Engineering [MBSE] is a paradigm that uses formalized representations of systems, known as models, to support and facilitate the performance of Systems Engineering [SE] tasks throughout a system's life cycle."[2] INCOSE stressed that MBSE is the development trend of future systems engineering methods and technologies, and a change in the field of systems engineering, and proposed the MBSE vision plan for the first time at the conference, and planned to realize the gradual maturity of MBSE theory and practice system from 2007 to 2020, which represents that MBSE will become an important development direction of systems engineering in the future.

MBSE has been recognized by many experts in the field of systems engineering, and is becoming the foundation of complex system design. Compared with traditional document-based system engineering methods, MBSE models become the core of complex system design from the requirements analysis stage to the evaluation stage. MBSE uses digital modeling instead of writing documents for system scheme design, and converts all the nouns, verbs, adjectives and parameters describing system structure, function, performance and specification requirements in design documents into digital model expressions. The system design of complex system is realized through the continuous evolution and iterative increase of the model, which has the advantages of unambiguous knowledge expression, reusability of the model,

5th International Workshop on Experience with SQuaRE Series andits Future Direction, December 04, 2023, Seoul, Korea
chejt@cesi.cn (J. Che); zhangyy@cesi.cn (Y. Zhang)
0000-0002-5456-1538 (J. Che); 0009-0006-4940-8527(Y. Zhang)

integration of system design, etc., and can well solve the problems and challenges brought by the increase of system complexity.

Compared with TSE(Traditional Systems Engineering), the characteristics of MBSE are reflected in three aspects: modeling language, modeling ideas and modeling tools. The main difference at the system modeling level is that the integration degree and executable degree of the system model are greatly improved, as shown in Table 1.

Table 1
Characteristics of TSE and MBSE

	TSE	MBSE
Modeling language	Natural modeling language	System modeling language
Modeling method	The main line is the function decomposition of the system	The main line is the layer decomposition of the system entity
Modeling tool	Computer (with natural language editing software installed), mainly for processing text	Computer (with system modeling language editing software installed), mainly to process graphical, visual and formal models
System model	It is composed of "discrete" reports, which rely on manual and manual correlation	Stored in the computer's model library and automatically correlated

2. Quality Characteristics of MBSE Systems

Compared with traditional software systems, the most important difference between MBSE based software systems is the formal modeling of the technical process throughout the whole life cycle, focusing on the formalization rather than the existence of modeling. Formal modeling has the following advantages:

- Unambiguous knowledge expression: MBSE describes the structure and behavior of the system using digital models that are based on clear rules and standardized symbols and terms, avoiding the semantic ambiguity or misunderstanding that can occur in traditional documents. Through digital models, MBSE can provide clear, consistent and accurate information that enables the design and functionality of the system to be understood and shared by all concerned, thereby improving the functional correctness of the system design.

- Reusability: The model in MBSE is independent of a specific project or system and can be reused in different projects or systems, which can not only improve work efficiency and reduce duplication of labor, but also reduce development costs and improve the quality and reliability of the system.

- Reliability: By building digital models, MBSE can conduct comprehensive simulation and testing of the system to identify and resolve potential problems and risks. This simulation can be carried out before the actual system is built, thus avoiding problems in actual operation and improving the reliability and stability of the system.

- Security: MBSE uses a modeling approach to design and analyze the system, which can more comprehensively consider the security requirements of the system, integrate the security concept from the beginning of the design, and reduce potential security risks. At the same time, MBSE can detect and solve potential security problems in advance through simulation and testing. Compared with traditional system testing methods, MBSE can test system security more comprehensively and systematically.

Combined with the characteristics of MBSE system, the mapping relationship between MSBE system and product quality characteristics can be established, as shown in Fig. 1.[3]

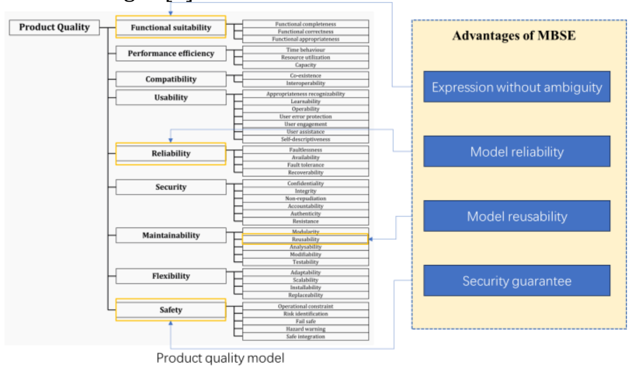

Figure 1: Relationship between MBSE system and product quality model

3. MBSE System and Software Quality Model

The model quality measurement includes two parts: the quality model and the quality measurement. The quality model gives the quality characteristics from which the quality of the model should be considered, and the quality measurement gives the quantification way of each quality characteristic. Based on the software product quality model in Fig. 1, combined with the characteristics of MBSE system, the model-based quality measurement m is formed after tailoring, as shown in Fig. 2.

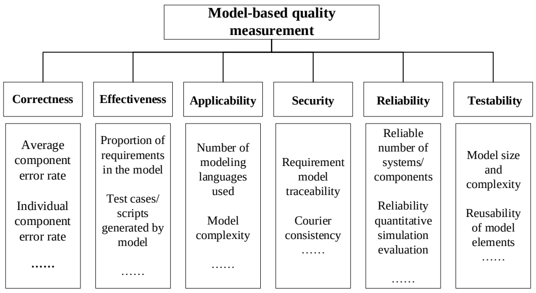

Figure 2: Model-based quality measurement model

A. Correctness

The main purpose of this work is to verify the consistency of the model with standards or modeling specifications. The model-based software test model is consistent with the rules of the formal description (modeling language, modeling guide), so that test cases or test data can be generated according to the model-based software test model without problems due to incorrect syntax.

The quality measure of correctness is divided into average component error rate and individual component error rate. The average component error rate can be used to assess the overall situation quality of the entire software system by the number of modeling requirements that are wrong/the number of all requirements in the model; The error rate of a single component can be measured by the number of incorrect modeling requirements in the concerned software configuration item/the number of all requirements in the concerned software configuration item, which is mainly to assess the model correctness of a single component in the software system.

B. Effectiveness

The main purpose of this work is to confirm the consistency between the model and user requirements and construction objectives. The content of the model is correct and expresses what needs to be expressed. The resulting product is "useful", the test script can be executed, the test case can be executed by the test engineer, and no errors can occur due to incorrect test cases.

Quality measures of effectiveness are to include:

The proportion of requirements in the model = the number of requirements in the model to complete modeling/the total number of requirements in the requirement specification;

Test cases/scripts generated by model = Number of test cases or scripts generated by the model/number of requirements modeled in the model.

C. Applicability

The main purpose of this work is to verify the consistency of the model with the test objectives and confirm the consistency with the test generation mechanism. Only when the model-based software test model is appropriate for the specified test objectives and the specified test generation mechanism can the products derived from the model-based software test model fully meet the expectations.

The quality measures of applicability would include:

The number of modeling languages used mainly evaluates the applicability of military software in model selection;

Model complexity is mainly measured by the number of nodes/elements in the model, the number of links in the model, the number of inbound nodes and the number of outbound nodes.

D. Security

To build a complete model system, first of all, we need to analyze the needs of users, establish the original demand and demand model, and establish the traceability relationship between them; Then the functional model and architecture model are modeled according to the requirements, and the traceability relationship between requirements and design is established. Finally, the system model after iteration is verified and evaluated.

In the system development based on MBSE, requirement analysis starts from the top-level requirement to model the requirements layer by layer, and establishes the requirement traceability relationship through the requirement model. The purpose of requirements analysis is to obtain the system requirements from the original requirements, so as to support the system architecture design.[4]

After completing the design of the requirement model, it is necessary to carry out functional analysis in the top-level design process. Analyze and model each functional component, design the realization method of the system use case through the use case model, build the functional architecture, and establish the traceability relationship between requirements and design. Function analysis reflects the system's activity and activity execution process by drawing activity diagram, and reflects the system's requirements. Moreover, through the corresponding relationship between functions and system requirements in the activity diagram, the functional model can also be linked to its corresponding requirements to complete the establishment of the traceability relationship between requirements and design.

Finally, on the basis of the functional model, the overall route and technical framework of the system are analyzed, and the system architecture model is designed. Architecture design is the transition from the top layer of the system to the software implementation layer. Through module definition, the relationship between modules is shown, the components and characteristics of the system are described, and the design and development of the system model is guided. The architecture design facilitates the refinement of the black box model and thus the conversion to the white box model of the module. Through architecture design, the requirements and functions of the system are assigned to components, ensuring the consistency of information transmission in the design process and realizing the accurate expression of the top-level framework of the system.

E. Reliability

Complex engineering systems are generally costly and high risk systems. If a large number of system reliability design problems can only be found after the physical construction integration, its change costs and expenses are inevitably unacceptable. Therefore, reliability design verification before construction is necessary for complex systems. However, at the same time, the reliability design verification of complex systems is very difficult, which is more difficult than the functional performance verification of the system. The reliability verification of the system should consider the impact of many variables such as the system's use scenario, environmental conditions, functional requirements, physical composition and personnel operation, which has long been a difficult problem in engineering.[5]

In the component/system integration verification stage, the reliability level of components and systems is virtual verified by means of component level and system functional reliability simulation analysis to determine whether they meet the requirements. This technical link needs a set of system reliability simulation verification method based on multi-physical model support, comprehensive use of all levels of the system, multi-domain physical simulation model, high fidelity simulation verification of reliability.

Relying on the high-precision integrated simulation model provided by MBSE, the multi-physical simulation model close to the real use scenario and environmental conditions is built early, and the system reliability is fully verified before the physical construction, and the weak link of the system reliability design is improved.

The key technical aspects of system reliability simulation and verification process based on multi-physical model include:

(1) Fault modeling of system components

Based on the multi-physical regional energy model of the system, various types of component fault simulation modeling are carried out, including mutation fault modeling, performance degradation fault modeling, logic error fault modeling, etc.

(2) Formalized requirement definition and verification

This paper mainly uses linear temporal logic and other formal languages to define reliability requirements by strict predicate and combinatorial logic, and uses model simulation to automatically output counterexamples of violating requirements, so as to verify whether the qualitative requirements of reliability are met.

(3) Reliability quantitative simulation evaluation

In the system reliability simulation evaluation, there are usually two technical approaches. One is based on the fault mode related data of the system components, using Monte Carlo sampling to evaluate the reliability. One is to use scenario model, operation model, fault mechanism model and multi-physical model to achieve quantitative evaluation of key component reliability and system functional reliability by task scheduling simulation.

F. Testability

The opportunities MBSE brings to system testability are mainly reflected in the modeling and sustainable simulation verification of the system design itself, which enables the virtual injection of faults into the system model to investigate their impact and evaluate the pros and cons of the test scheme, thus achieving a more objective, comprehensive and accurate testability application.

Test requirement is the basic basis for designing in-machine test and automatic test equipment, which is conducive to achieving the consistency of testability analysis, automatic test and fault diagnosis.[6]

Automating the testability analysis and design process based on MBSE requires solving the following key technologies.

- Complex system multi-level co-simulation verification technology. This is not only the core of future testability technology, but also a prerequisite for high-quality MBSE. The hardware and software mixed simulation implementation and cooperation and other contents support the simulation verification of system functions and performance.
- Fault data acquisition technology based on automatic fault injection and simulation. Automatic batch injection of faults in simulable models is used to ensure the coverage of real underlying faults. One of the key technical points is the transformation of low-level physical faults to high-level fault behavior descriptions, which makes it possible to implement credible fault behavior simulation at a higher level of abstraction (such as behavior level), reduce the dependence on design details, and improve simulation performance and operability.[7]
- Data-driven testability modeling, analysis and design techniques. Based on the above fault simulation data, feature extraction is carried out to obtain fault samples, and then the intelligent classification algorithm is studied according to the characteristics of the samples. Ideally, if the sample size is sufficient and balanced, a cross-validated testability analysis method can be used to quantitatively predict key testability indicators. Testability design is to optimize fault features according to optimization objectives such as the number of test points or test cost among many candidate features.[8]

Metrics and key performance indicators that may be monitored include:

- Manages and tracks the number of requirements in the MBT (Model-Based Testing)model and the percentage of requirements covered by the test cases.
- Size and complexity of the MBT model.

- The number of test cases and scripts generated, and the number of test cases/scripts produced per person per day.
- Number of requirement defects found in MBT modeling activities.
- The reusability of MBT model elements from one project to another.
- The extent to which the MBT model is used by the project stakeholders (business/development/test).
- Percentage increase in efficiency versus productivity of previous test design methods (lower cost testing).
- Percentage improvement in efficiency, defect finding rate compared to previous test design methods (better testing).

4. Conclusion

MBSE has a broad vision for improving the quality of software products. As the technology continues to evolve, MBSE methods and tools will continue to be refined to more accurately describe and simulate complex systems, further improving the functionality, reliability and maintainability of software products. At the same time, MBSE will be combined with new technologies such as artificial intelligence and big data to achieve more intelligent software design and quality management, and improve the quality and development efficiency of software products. In the future, MBSE is expected to become an important development trend in the field of software engineering and make greater contributions to the improvement of software product quality.

References

[1] Thomas A J, Jonathan M J, Paredis C J J. et al. Modeling Continuous System Dynamics in SysML [C]. Proceeding of the International Mechanics Conference and Exhibition. Seattle: 2007: 1-11.

[2] International Council on Systems Engineering (INCOSE). Systems engineering vision 2020[R]. Seattle, WA: International Council on Systems Engineering, 2007: 5-32.

[3] ISO/IEC 25010:2023. Systems and software engineering Systems and software Quality Requirements and Evaluation (SQuaRE) Product quality model.

[4] BAJAJ M, BACKHAUS J, WALDEN T, et al. Graphbased digital blueprint for model based engineering of complex systems[J]. INCOSE International Symposium, 2017, 27(1): 151-169.

[5] WU J, YAN S, ZUO M J. Evaluating the reliability of multi-body mechanisms: A method considering the uncertainties of dynamic performance[J]. Reliability Engineering & System Safety, 2016, 149:96-106.

[6] Cook D, Schindel W D. Utilizing MBSE patterns to accelerate system verification[J]. Insight, 2017, 20(1): 32-41.

[7] TANG X F, XU A Q, LI R F, et al. Simulation-based diagnostic model for automatic testability analysis of analog circuits[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and System, 2018, 37(7) : 1483 - 1493.

[8] TANG X F, XU A Q, NIU S C. KKCV-GA-based method for optimal analog test point selection[J]. IEEE Transactions on Instrumentation and Measurement,2017, 66(1) : 24 - 32