

The Research of Trusted Attribute Based on Model-Driven of MDA

Kemeng He¹, Yushan Sun², Zhang Qin¹, Yuqiang Sun¹ and Yuwan Gu^{1,*}

¹School of Information Science & Engineering, Changzhou University, Jiangsu, Changzhou, 213164, China

²Department of Computer, Henan Institute of Engineering, Henan, Zhengzhou, 451191, China

Abstract: The research methods of trusted attribute based on model-driven for credibility problems in the development process of model-driven software proposed in the paper. An overall analysis for software requirements is carried out, considering the business specification and the need to achieve the function, testing the Integrity and accuracy of model in the PIM phase. Then, is a need to achieve the necessary component services in the concrete platform for implementing the business logic of PIM, studying the symmetry and reliability of model in the PSM phase. Finally, the use of transformation method of element mark to achieve the conversion of platform model, testing the transfer ratio of model in the phase of PIM transformed to PSM. The research of trusted attribute based on model-driven of MDA has effectively improved reliability and developing efficiency of software.

Keywords: Model-driven, trusted attribute, PIM, PSM.

1. RESEARCH BACKGROUND

With the development of the software technique, various types of software have become integral in all the fields of national economy and national defense construction. But the software system is increasingly becoming large and complex. The software defect problems and system flaws are also increasing. Many software systems become fragile and those applications are not run most of the time, which often have a variety of failure and fault, so whether the software can run successfully, when needed and bring safe and reliable service to the user is the most concerned problem. This is termed as the "software trustworthiness" problem [1].

Research on a trusted construction method for electric power production management system of BAO Tie and LIU Shu-fen et al points out that with the increasing demand of trust and the development of trusted technology, trusted technology will be applied to the industrialized production practice in large scale and further research can be done on a lot of content [2]. So the research of trusted attribute based on model driven is proposed in this paper. MDA proposed by OMG includes PIM, PSM and PIM transformed to PSM etc. The Integrity and accuracy of model are tested in the PIM phase, and the symmetry and reliability of models are tested in the PSM phase, and the transfer ratio of model are tested in the phase of PIM transformed to PSM in the paper [3].

2. MDA FRAMEWORKS IS BRIEF INTRODUCTION

MDA is a software development framework defined by OMG, or is a software development standard. The system

software is based on various models, and drives the development of system through the model transformation, and on this basis designs a portable operable application program [4].

MDA regards the modeling language as a programming language, and is not just the design language. IT technology will be a series of new technological trend. These technologies include development based on component, design patterns, middleware, illustrative constraint, abstract, multilayer system, enterprise application integration and design of contract and so on.

MDA separated the model of software system to be PIM and PSM in the MDA proposed by OMG, and can unify them through the conversion rules at the same time in such a way that it gets rid of the dilemma brought by the change of requirements. PIM is the high hierarchical abstraction for the system, which does not include any information of implementation technology. PSM is a model of specific platform. we firstly use platform independent language to build the PIM in the MDA framework, then transform PIM into PSM according to the specific platform and the mapping rules of implementation language, at last PSM generates application code and test framework [5]. Combined with the stage of software design, as shown in Fig. (1).

MDA life cycle is the same as traditional life cycle, and their main difference is that work piece is created in the development process. MDA life cycle includes PIM, PSM and code. PIM is a model of the high hierarchical abstraction and is independent of any implementation technology. The PIM is converted to one or more PSM. PSM is tailored to a specific implementation technology. The final step of development is that each PSM is transformed into code, and PSM is closely related with the application technology. The traditional development process is transformation from model to model, or the transformation from model to code is done by

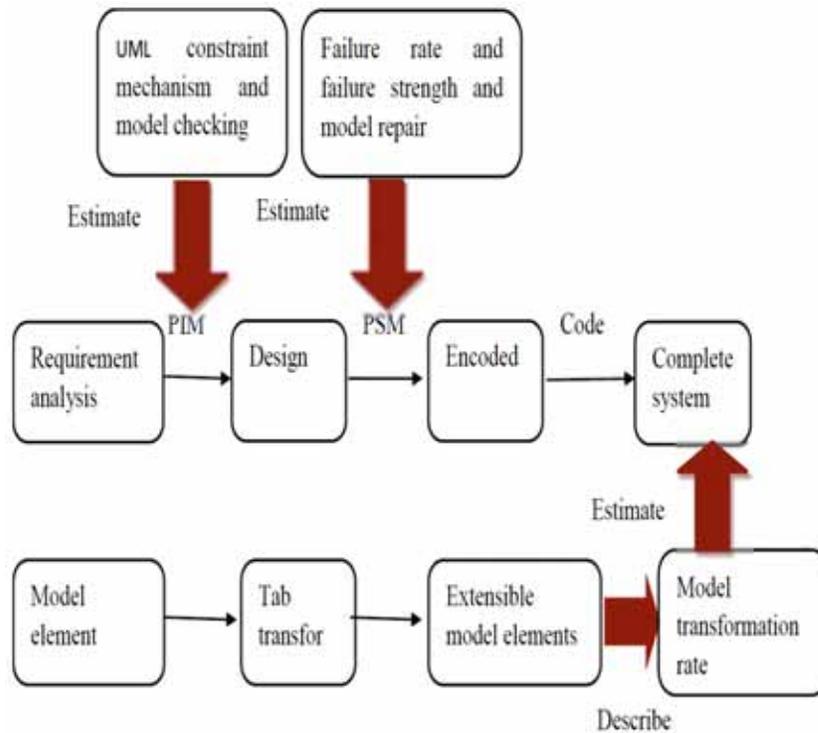


Fig. (1). Model driven software development process.

hand. But the transformation of MDA is done automatically by the tool. From PIM to PSM, and from PSM to code can be implemented by the tool. PIM, PSM, and Code model are regarded as designed work piece of the software development life cycle, and are the document and the chart in the traditional development mode. Importantly, they represent different hierarchical abstraction in the system, to view our system from the different angle of views, the high hierarchical PIM transformed PSM's ability to improve the level of abstraction. This can make developers more clearly understand the whole system architecture and is not polluted by concrete realization technology; it reduces the workload of developers on the complex systems at the same time.

3. THE MODELING STEPS BASED ON MDA

3.1. PIM Model

UML has a highly abstract modeling in the application domain, and this model has no relation to the technology of its realization (or the underlying technology) completely. This model is called a PIM. The Platform Independent Model separated the structure and function of the system from the special technical details, which has nothing to do with the concrete realization technology and hardware environment. It is a model of the most abstract and the highest coordination, and it is the core of MDA. The entire MDA development project originates from the creation of PIM, and this level of PIM express business function and behavior, through the technical details to ensure that there is no loss. The applied modeling tools of MDA include Pervasive Services and dominical tools, and allow the application system to use and combine them through the options menu [6].

3.2. The PIM to PSM Mapping

Mapping in MDA uses a series of rules and technology to transforms a model into another model, and the model mapping is to realize the transformation of models. PIM improves itself according to the mapping technology and is transformed into PSM through automatic transformation tools when the PIM is improved to a certain extent and then the generated PSM perfects itself according to the mapping technology again. If the transform tool is represented as a black box, the input is a model and the output will be another model. For example, we can use the applied, proved and automated UML to J2EE conversion mode between design models expressed in UML and the implementation model expressed in J2EE.

3.3. PSM Model

We can build a PSM model in the defined mapping steps through the system model of PIM. PIM can be mapped into a single model or for multi-platform. Established platform specific model is of a formal specification based on specific target platform; PSM will use a particular implementation technique to describe the system. It will use a variety of architecture, such as EJB, database model, COM module etc. [7].

4. PRIMARY COVERAGE

4.1. Research of Trusted Attribute in PIM

We must know what the target system must do, when we analyze the function demand of the software, and give complete, accurate, clear and specific requirements on the target system [8]. Some users understand the issues they face and

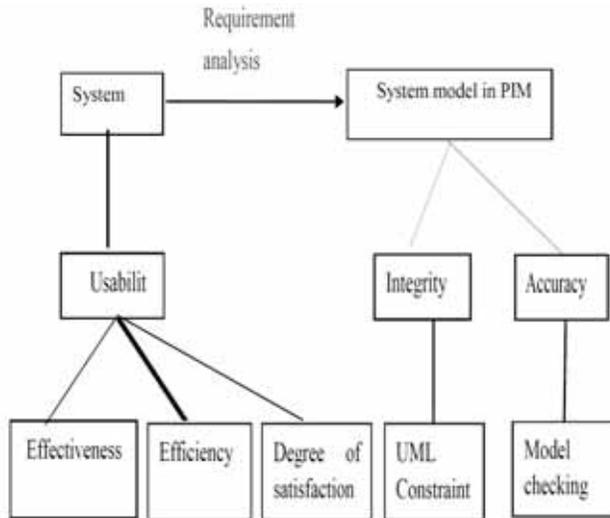


Fig. (2). Trusted attribute in system and trusted attribute in PIM.

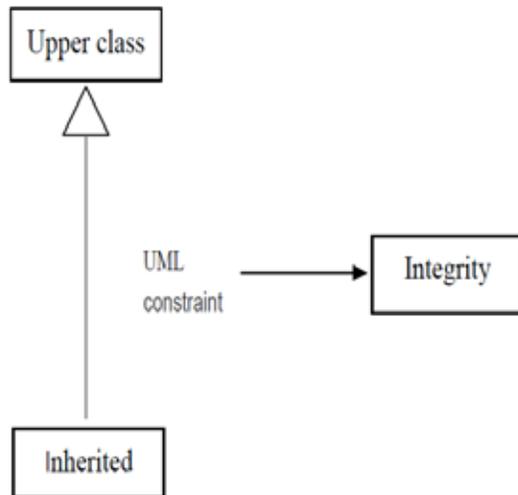


Fig. (3). UML constraint.

know what must be done. But most often cannot completely and accurately express their demands and do not know how to use computers to solve their problems. Software developers know how to use software to realize people's requirements, but the specific requirements of users are not completely clear. So system analysts must cooperate closely with users in the requirements analysis phase, when the system model is established in order to get system requirements and system model through user confirmation. We create many scene models which also are various functions achieved by the target when building a PIM, so we must consider the usability of software, this can ensure the integrity and accuracy of the model in the PIM phase [9]. As shown in Fig. (2).

The software usability refers to the user use of the product to achieve specific concrete tasks in specific environments, the effectiveness of the interactive process, interaction efficiency and user satisfaction [10]. Hereinto:

(1) Effectiveness refers to the degree of accuracy and completeness when users complete a specific task.

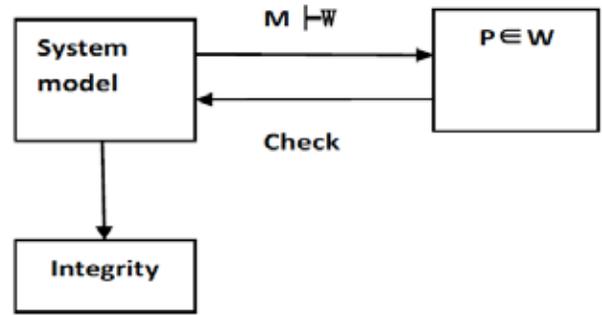


Fig. (4). Model checking.

(2) Efficiency refers to the ratio between getting the degree of accuracy, completeness and resources when users complete a specific task.

(3) Satisfaction refers to the user's subjective satisfaction and acceptance in the process of using the product.

The integrity of the model in PIM is the UML constraints. Constraint is a mechanism limiting element semantics in PIM. Constraints can be added in a class or object, the more common case is that constraints can be added in relationship, and constraints participate in class or object in this relationship [11]. As shown in Fig. (3).

The accuracy of the model by model checking to verify the model established in PIM [12]. Model checking is a formal verification method of finite state systems, and it is the basic matters based on the state search and the natural extension of simulation and test methods The exhaustiveness of search depends on building accurate model for the system, which causes certain difficulty for modeling, but this can ensure accuracy of the generated model. The basic idea of model checking method is that the search through the state space identify whether system meets any specification. Namely, given a system P and the specification of W, which generates an M corresponding to the system model, then prove $M \models W$. (w is true in one interpretation m) Namely, specification formula W is established in the model M, and this proves that the system P meets the specification W. As shown in Fig. (4).

Through the above research on trusted problems, we use constraints and test of the model to research the integrity and accuracy of the model in the PIM, which enhances the credibility of developed software based on MDA through the integrity and the accuracy of the model.

4.2. Research of Trusted Attribute in PSM

The PSM model is implemented by the form of the extensible UML model in the PSM phase. The UML element model emerges by the extensible UML model described by specific implementation technology through mapping rules, and then the extensible UML model is transformed into the code of specific implementation platform. For example, UML combined with EJB forms the extensible UML model, thus this model can be transformed into a code on JAVA platform. Platform specific model in the mapping process

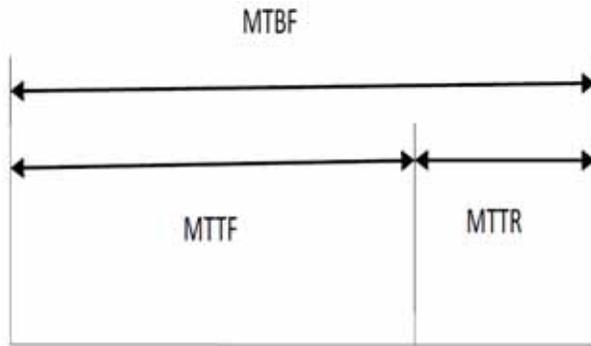


Fig. (5). Relationships of MTTF, MTTR and MTBF.

must correspond with the platform independent model; otherwise establishing specific model may produce errors, eventually leading to the errors of the realization code. Moreover, we should pay attention to the reliability of the model at this stage. Because the established model in the platform specific model is transformed into the concrete implementation code, so we must ensure the reliability of the model. Reliability refers to the probability that model cannot cause the system to lose effectiveness under specified conditions, within the prescribed time. We usually use failure rate, failure strength and MTTF, MTTR, MTBF etc. to describe the reliability of the model.

If the system is not up to or can't meet the user requirements, PSM described by the specific implementation technology is unreliable through mapping. So we need to consider the failure rate and failure strength of the model and repair the model. The failure rate and failure strength is higher or average if the failure time of the model is long, and the dependability of developed software is low.

(1) The failure rate and failure strength

The failure rate refers that in the case if model have not yet encountered failure at the moment t in time, the model has the failure probability in unit time.

The failure strength refers to the change rate of the average failure number by time, namely the assumption that model encounters the failure number $N(t)$ at the moment t , apparently $N(t)$ is a random number, and $N(T)$ is different by the change of time t , namely $\{N(t) \ t > 0\}$ is a stochastic process. Suppose that $u(t)$ is the mean value of the random variable $N(t)$, then $u(t) = E[N(t)]$, so $\lambda \ t = du(t)/dt$ is the failure strength at the moment t .

(2) MTTF MTTR and MTBF

MTTF MTTR and MTBF are 3 simplified reliability parameters.

MTTF is the Mean Time To Failure, $MTTF = 1/\lambda \ t$.

MTTR is the Mean Time To Repair.

MTBF is the Mean Time Between Failures, the relationship between the three of them as shown in Fig. (5).

We can see from Fig. (5) that $MTBF = MTTR + MTTF$

According to the above research on trusted problems, we use the failure rate, failure strength and MTBF of the model

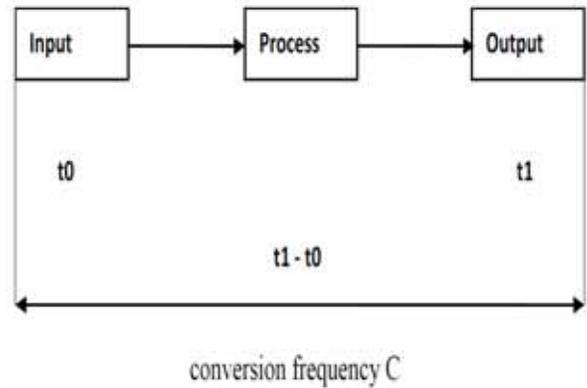


Fig. (6). Transformation process.

to measure the credibility of the model in a platform specific model, thus we know credibility of the software based on MDA.

4.3. Research of Trusted Attribute in the Transformation Process from Platform Independent Model to the Platform Specific Model

The system function described by UML class diagram forms the system model, it keeps the design result in the form of text and can produce the platform specific model of the extensible UML through mapping, so mapping becomes transformation technology from the PIM to the PSM. We chose a specific platform before conversion, and a mapping rule corresponding to the platform. This mapping rule contains a series of predefined markers. These markers are used to add tags to the model elements, which guide the process of model transformation. The transformation process is that, we initially add tags to model elements, then use the mapping rules corresponding to the platform, transform the marked model and generate the corresponding PSM. This method is used to implement platform transformation in the stage of transformation from PIM to the PIS [13].

We define a general transformation formula: $Transform = \langle Input \rangle + \langle Process \rangle + \langle Output \rangle$. Input is the input of transformation, and process is the process of dealing with transformation, and Output is the output results of transformation. Assume that at time t , the time after in which model transformed one timece is $t1$. Writing down the frequency of using the formula Transform is C . Then transfer ratio $P = C/(t1 - t0)$; If the developed software can't satisfy user's demand, we will change the system function model, or modify PSM model of specific implementation technology, obtaining a transformation of the model again, and the number of model conversion can increase, thus P gets large. So we use transfer ratio P to describe the software credibility. As shown in Fig. (6)

Through the above research on trusted problems, we use the transformation method of element tag to evaluate transfer ratio of the model in the PIM transformed from the model transformation stage, and we can know the trusted degree of developed software based on MDA according to the contrast transfer ratio of the model.

CONCLUSION

The trust attribute is researched, using the idea of model driven software development. The research of trusted attribute based on model driven proposed in this paper, includes the research of trusted attribute in the PIM, the research of trusted attribute in the PSM and the research of trusted attribute in the transformation process from PIM to the PSM. We can transform the credibility problem of software by researching a trusted attribute of model driven. Using model driven to improve the efficiency of software development, enhancing the portability of the software, interoperability, maintainability, and improving the convenience of documentation etc. However, this research considered only trusted attribute of system mode in MDA development, no actual consideration to the index of credibility and the cost in the phase of software development. Therefore, in the following research, we can research the trusted attribute of each phase in the software development.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Supported by The National Natural Science Fund 11271057, 51176016 and the project of general office of Broadcasting and Television (GD10101) and Natural Science Fund in Jiangsu (BK2009535) and Natural Science Fund in Zhejiang (Y1100314) and Jiangsu Province ordinary university innovative research project (CXZZ13_0691).

REFERENCES

- [1] J. Ouyang, H. Wang, and D. Shi, "Research of trust in pervasive computing", *The Research of Computer Application*, vol. 25, no. 2, pp. 3521-3524, 2008.
- [2] L. S. Boyi, and X. Wang, "Research on trusted structure of electric power production management system", *Chinese Journal of Electronics*, vol. 38, no. 9, pp. 2166-2171, 2010.
- [3] P. Harmon, "The OMG'S model driven architecture and BPM", *Business Process Trends*, vol. 2, no. 5, pp. 1-3, 2004.
- [4] A. Kleppe, J. Warmer, and W. Bast, "*MDA Explained-The Practice and Promise of The Model Driven Architecture*" Beijing: Posts & Telecom Press, 2003.
- [5] Kleppe, J. Warmer, and W. Bast, "*MDA Explained-The Model Driven Architecture: Practice and Promise*", Boston: Addison Wesley Professional 2003.
- [6] G. Caplat, and J. L. Sourrouille, "Model mapping using formalism extensions", *IEEE Software*, vol. 22, no. 20, pp. 44-51, 2005.
- [7] X. Zhong, and X. Zhao, "Method for software trustworthiness evaluation supporting dynamic and multiple attributes", *Computer Engineering and Science*, vol. 35, no. 6, pp. 107-112, 2013.
- [8] T. Bao, and B. Li, "Research on a formal description method of network behavior based on communication sequential process" In: *ICPCA08-2008 3rd International Conference on Pervasive Computing and Application*, 2008, pp. 225-229.
- [9] A. A. Igirdas, C. L. Jwan, and R. Brian, "Basic concepts and taxonomy of dependable and secure computing", *IEEE Transaction on Dependable and Secure computing*, vol. 1, no. 1, pp. 11-33, 2004.
- [10] R. L. Michael, "*Handbook of software reliability engineering*", IEEE Computer Society Press, USA 2006.
- [11] L. Fuentes, and P. Sanchez, "A generic MOF metamodel for aspect-oriented modeling", *Proceedings of the 3rd International Workshop on Model-Based Methodologies for Pervasive and Embedded Software*, Postdam, Germany 2006, pp. 113-124.
- [12] J. Yuan, and M. Huang, "Product configuration method based on unified modeling language and conditional con-straint satisfaction problem", *Computer Engineering and Applications*, vol. 49, no. 14, pp. 1-24, 2013.
- [13] M. Chen, C. Deng, and L. Ren, "A research on the written interface development framework based on model driven", *Chinese Journal of Electronics*, vol. 39, no. 6, pp. 268-274, 2011.

Received: November 10, 2014

Revised: January 07, 2015

Accepted: January 21, 2015

© He *et al.*; Licensee Bentham Open.

This is an open access article licensed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.