Research on general foundation platform based on MDA

Qingfeng Wu, Minwen Wu, Huailin Dong*

Software School, Xiamen University, Xiamen, China

Received 1 October 2014, www.cmnt.lv

Abstract

With the rapid development of informatization in various industries, the needs of information systems for government and enterprise continues rising, and the scale and complexity of such systems is increasing as well. Therefore, lack of reusability and reliability has become one urgent issue in traditional software design. Based on the core idea of model-driven development, this paper designs a J2EE-based business platform which has three subs systems: model generation subsystem, workflow information subsystem and role-based permissions subsystem. The implementation and practice of this platform indicates that such software development method based on platform can significantly improve the development efficiency. The basis functional modules and workflow applications can be implemented by configuring the software. Through the built-in user management, menu management, rights management and other public module, the platform can greatly improve software reusability, reducing the development effort and achieving perfect results.

Keywords: platform, MDA, general foundation platform, information systems, J2EE

1 Introduction

The brutal competition in the market is the biggest challenge faced by almost all enterprises. In order to obtain higher returns in the market competition, many companies are conducting internal information-based transformation [1, 2]. Information systems such as MIS, ERP, CRM, SCM, etc. are gradually perceived and applied by enterprises [3]. However, these systems cannot solve problems once for all in the information process. On one hand, most of enterprises have many specific business functions and management processes, the general universal product can't meet the different needs for enterprises internal management [4]. In this way, these part requirements are required to be customized. On the other hand, traditional software development techniques have high failure rate, low-level reusability, long project cycle and poor portability features, so it is difficult to realize the further development task of enterprise informatization [5].

The growing demands for software, workload, complexity of software development and high failure rate form the basic contradiction of the software industry. To solve this problem, people focus on the two following directions [6, 7]: one is software engineering and software project management, the other is the innovation of software technology. Among them, the most significant technical innovation is the software platform which is able to combine with the two directions researches. The application of the new technology into software platform's underlying base modules could support specific business modules to improve the portability and reusability of business systems [8].

On the basic of analysis above, this paper introduces the model-driven development theory, designing a universal basic platform based on model-driven architecture. Then through secondary development and application, the performance of the platform is verified.

This paper briefly introduces the core idea of platform MDA in the Section 2. In Sections 3 and 4, the universal platform is intensively analyzed and designed. In Section 5 the platform application in practice is presented. Finally, the conclusions and outlook are summarized.

2 Research progress in domestic and abroad

2.1 RESEARCH ON PLATFORM

At present, there are two main directions of the studies on software platforms: one is the software infrastructure platform and the other one is the business infrastructure software platform. Business infrastructure software platform refers to the business-oriented and driven platform, which can quickly build applications. All of the functional components, such as financial software, e-commerce, and management software which are developed based on this common platform. Through the combination with various components, the management of enterprise applications achieves a smooth and seamless connection [9,10].

Nowadays, many software companies in domestic and foreign have realized the advantages of platform and model-driven development, and developed a number of platform products based on MDA, and achieved some effects. However, these products still have some shortcomings [11-13]. On one hand, some platforms adopt the mode that compiler generates code to run. As a result, the deployment is complex and the management is difficult, such as Kingdee BOS. On the other hand, the coupling among some platforms and some specific industries is too deep. Besides, the model is added to the

* ***Corresponding author's*** e-mail: hldong@xmu.edu.cn

industry relevance model properties, which would lead to the great reduction of the platform applicability, such as Bask platform of Yili.

2.2 OVERVIEW OF MDA

In order to solve the problems in traditional hand-coded software development, the Object Management Group (OMG) puts forward the Model-Driven Architecture (MDA). Its architecture is shown in Figure 1. The MDA architecture could be divided into four levels: modelling techniques, middleware and platform, universal service, and the solutions of problem domain [14,15]. The key essence of MDA is to enhance the current development to a higher abstraction level - analysis model level. These models can be read by MDA tools, and through a variety of standard mapping automatically converted into C++ code, Java code, etc [16-18].

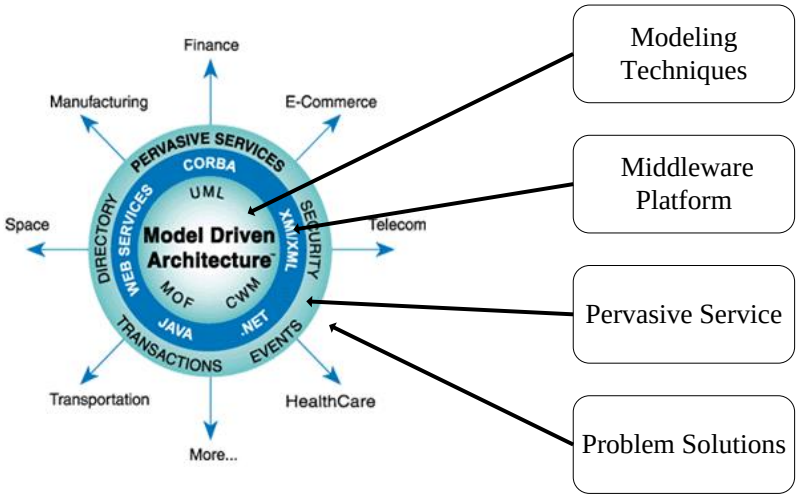

FIGURE 1 Architecture of MDA

MDA separates the system functional description and the implementation on the particular platform [19]. It divides the system model into two types: one is a platform-independent model (PIM) and the other one is platform-specific model (PSM). PIM is an abstract formal description of system functions and structure, which represents the business functions and behaviour. PSM is a concrete description of system functions and structure on the specific platform. It introduces some platform-specific information into PIM. Both PIM and PSM are described by UML.

MDA has three levels. The core are modelling standards UML, MOF and CWM and followed by some public services based on the core, such as directory services, transaction services, persistence and security, etc [20]. The outermost layer is the applications of MDA in various fields, such as communications, e-commerce, manufacturing and financial fields, etc [21].

MDA plays an important role on promoting software development efficiency and enhancing software portability, interoperability and maintainability.

3 Analysis of general foundation platform

3.1 GOAL ANALYSIS

This paper designs a general foundation platform based on MDA, mainly used in the construction of enterprise back-office information system. Through the modelling of business systems, the enterprise's business is divided into business objects and processes, and generates application by parsing engine. Therefore, the platform design goals are as follows:

1) This platform can provide the basic foundation for the business system which is developed on it. Besides, it supports secondary development and model generation of the platform, including user management, institutional management, menu management, rights management and cache management, etc.

2) The business object and business process models can be configured and saved by WEB tools. The platform can automatically generate related CRUD functions by parsing engine. In addition, it can maintain running state of processes for business forms and dispatch activities between uses.

3) The design of the platform model is required to provide generality and neutrality, without coupling with any industry businesses to maintain its versatility. The model can be as much applicable to all industries.

4) It's convenient for custom extensions and modifications. Furthermore, platform supports template technology, so the interface could be easily modified and expanded.

3.2 FUNCTION USE CASE ANALYSIS

The use case diagram of configuration management is shown in Figure 2. It shows three main use cases and eight sub-use cases of system configuration manager and system administrators. It includes some model metadata, CRUD function, and the design of business process and the configuration of menus, users, and permissions.

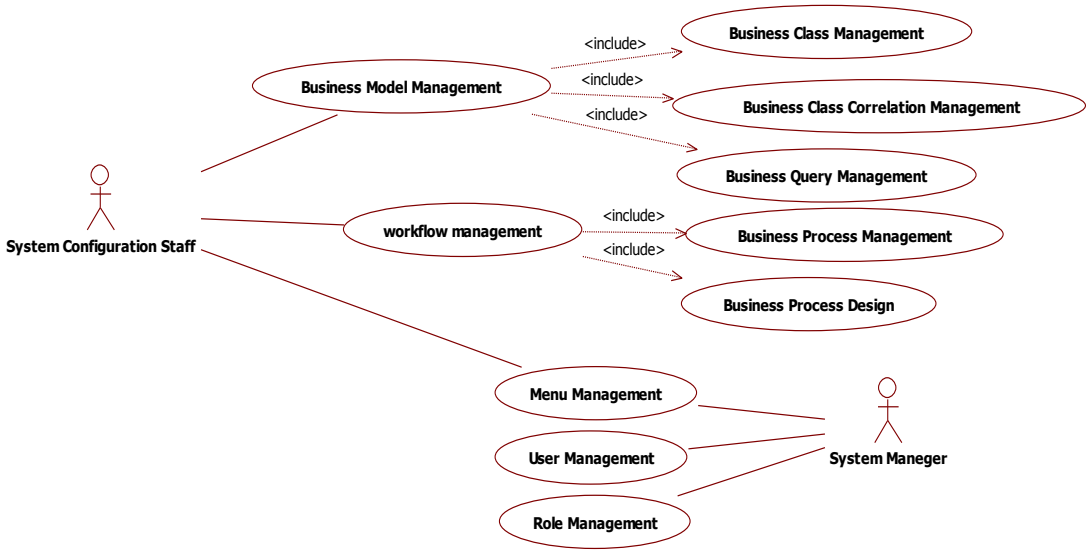

FIGURE 2 Use case of platform configuration management

The use case diagram of platform analysis and configuration is shown in Figure 3. It shows the eight sub-system use cases for system users. It contains the user's login and logout, load of workspace menu, generation of configuration page and the dispatch of process, etc.

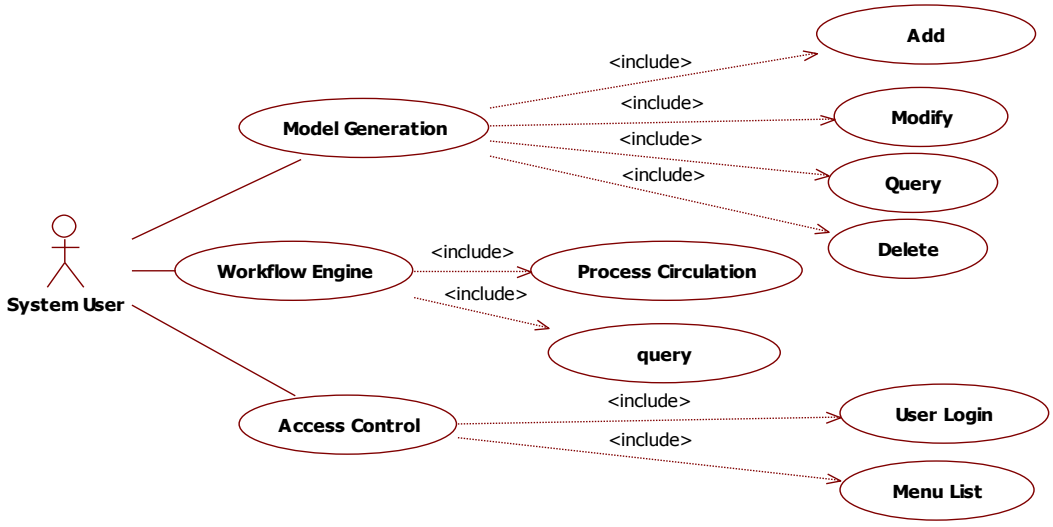

FIGURE 3 Use case of platform analysis configuration

4 Design of general foundation platform

4.1 DESIGN OF OVERALL ARCHITECTURE

The platform overall architecture is shown in Figure 4, in which the platform functions are divided into three levels. The first level is the base module, providing basic user role management, menu management, institutional management, cache management and control management. This part is not only the infrastructure of model-driven architecture, but also the key to provide interfaces to support custom coding development. The middle level is the model management and model parsing engine. The model management is composed by a series of modelling tools, which is responsible for input and persistence of metadata. The model parsing engine is responsible for loading the metadata, management of page template and program extension system. With the business models and processes parser, the model parsing engine can combine the metadata, page templates, and program extension together to produce the final application.

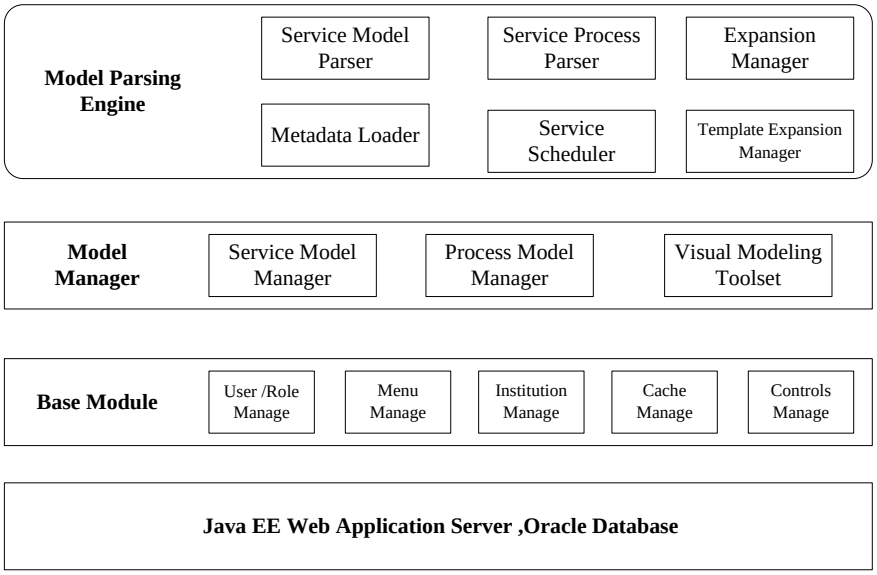

FIGURE 4 Platform overall architecture

4.2 TECHNICAL FRAMEWORK OF PLATFORM

The platform technical framework diagram is shown in Figure 5. The platform uses a lot of open source framework technology. The Persistence Layer uses Spring JDBC and Hibernate. The Service Layer adopts Spring as an integrated framework and uses the property of Spring AOP and IOC's to supervise the components in container, integrating Business Layer components such as EHCache. The Presentation Layer uses JSTL built-in Spring MVC and J2EE, the custom Tag components and Freemarker. Dhtmlx component and some JS component library are used on interface design.

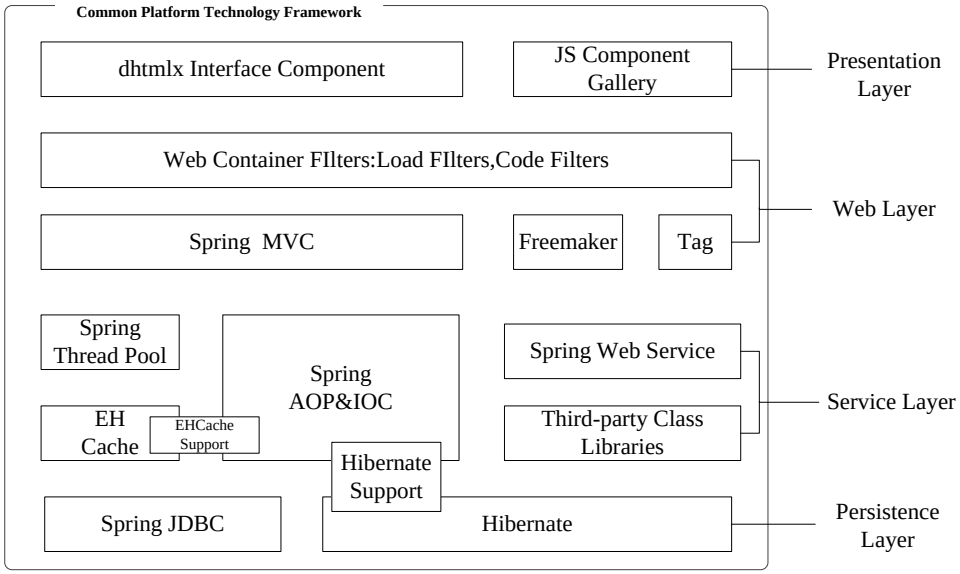

FIGURE 5 Platform technical framework

4.3 BASE MODULES OF PLATFORM

According to the analysis of requirements, the functions of the platform could be eventually divided it into three subsystems: model management systems, workflow management and rights management system, which is shown in Figure 6. Each subsystem contains some related functions. They are independent from each other and tightly integrated, working together to achieve the platform configuration development capabilities.

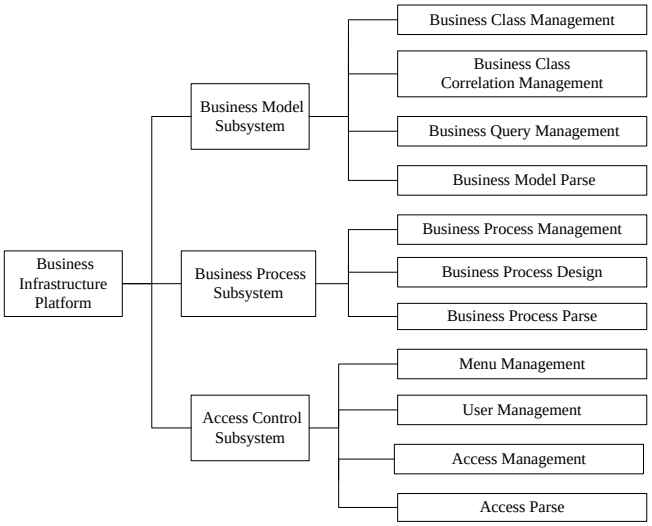

FIGURE 6 Platform function structure

The platform uses RBAC (Role Base Access Control) for access control. The system defines operable resources in advance and then assigns the available actions and menus for the role. Through the system authorization model to assign a role to one user, the operate authority for system users is its role authority union set.

Meanwhile, platform provides a cache manager based on EHcache. The cache manager provides two caches for users and platform which can be used to load some frequently accessed metadata to improve the performance of platform.

5 Application practice of platform

The goal of the platform is to satisfy business requirements quickly. When developing through the platform, the developer should understand the relationship between the platform and target system, and the entire process of development. As shown in Figure 7, application system is built on the foundation platform. Firstly, developers build applications metadata through the user interface configuration tool on the platform. And codes for general system are usually stored in the form of data in platform table space. After successful configuration, the platform will read the data in the table space automatically and instantiate applications by analyzing these data. Functional operations for uses are analyzed by parsing engine. Then the engine generates the actual database operations and interacts with data in the application table spaces.

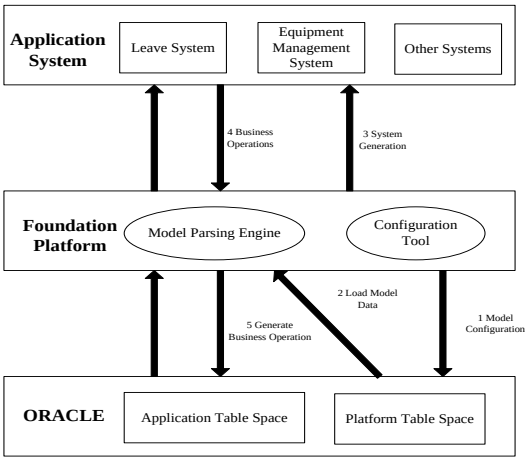

FIGURE 7 Platform and business systems architecture

The MDA-based business platform is aimed at solving some serious problems in the development of information system with least cost. Such problems cover heavy workload, high failure rate, easy changes in demand and so on. And one advantage of the platform is that it could be widely used in various industries. This platform is high efficiency, stability, portability, which can combine with the object-oriented development very well. The system developer can construct one business system directly by working through modelling and configuration. In this way, people can replace the traditional pure coding software

development to some extent, greatly improving the software reusability.

6 Conclusions and outlook

The traditional development methods can't meet the need of growing complexity and scale of enterprise information systems. It not only has long development cycle, high failure rate and low software reusability, but also is difficult to adapt to the changes in demand. In this situation, this paper draws on the ideas of MDA, and design one universal platform based on MDA architecture. This platform integrates the basis functions, business model, business process which can significantly improve reusability and efficiency of software development.

Acknowledgements

Supported by the National Natural Science Foundation of China under Grant No. 61202142; the Scholarship of Overseas Studies in Fujian Province; the Key Technology R&D Program of Xiamen, Fujian under Grant No. 3502Z20103001, 3502Z20101002; the Leading Academic Discipline Program, "Project 211 (the 3rd phase)" of Xiamen University; the Fundamental Research Funds for the Central Universities, Xiamen University under Grant No. 2011121023, CXB2012012, CXB2012013, 201212G007, CXB2013016.

References

[1] Anaya R, Quintero 2007 MDA and the Role of Models in the Software Development Process *EIA journal* 8 131-46
[2] Panfilenko D V 2014 Recommendations for impact analysis of model transformations from the requirements model to the platform-independent model *Proceedings of the 16th International Conference on Enterprise Information Systems* 2 428-34
[3] Van der Aalst, Rozinat, Russell 2010 Business Process Simulation: How to get it right? *In: Handbook on Business Process Management, International Handbooks on Information Systems Springer-Verlag* 317-42
[4] Zorzan, Riesco, 2008 Transformation in QVT of software Development Process Based on SPEM to Workflows
[5] Bocciarelli, Ambrogio 2011 A BPMN extension for modelling non-functional properties of Business Processes *In: Proceedings of the Symposium On Theory of Modelling and Simulation*
[6] Bocciarelli, Ambrogio 2011 Performability-oriented Description and Analysis of Business Processes *In: Beckmann, J.A. (ed.) Business Process Modelling: Software Engineering, Analysis and Applications. Nova Publishers, New York*
[7] Braghetto, Vincent and J Performance evaluation of business processes through a formal transformation to san *In: Proceedings of the 8th European conference on Computer Performance Engineering Springer-verlag Berlin Heidelberg*
[8] Yang, F, Shen W, Tan W, Ghenniwa H 2010 A framework for service enterprise workflow simulation based on multi-agent cooperation *In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics Istanbul Turkey* 2587-94
[9] Xue Z, Wu Z 2008 Study on Model Driven Pattern of Software Development *Computer technology and development, Technology and development* (2) 24-6
[10] Smith R M, Trivedi K S 1988 *IEEE Trans Compute* **37**(4) 406-17
[11] Craftware. UML Executable and MDA. http://www.craftware.net/es/ visited 08/2012
[12] EMFT. Eclipse Modelling Framework Technology. http://www.eclipse.org/modeling/emft/ visited 09/2012
[13] Dahman K, Charoy F, Godart C 2010 Generation of component based architecture from business processes: model driven engineering for SOA *In: European Conference on Web Services* 155-62
[14] Wang Y, Chen W 2006 Application Research on Advanced Model-driven Developing Method *Computer Engineering* (13) 63-5
[15] Liu X 2012 Research on Model-driven software developing method *China CIO News* (9) 98-9
[16] Qiao Y 2009 Research and design of business platform based on component [D] *Jiang Su University*
[17] Calinescu R, Grunske L 2011 s *IEEE Trans Soft Eng* **37**(3) 387-409
[18] Xie P, Yu J 2012 The research on MDA model transformation oriented to J2EE *Microcomputer & Its Applications* (17)
[19] Zeng Y, Zhou J, Sun Z 2012 Research on MDA-supported design pattern modelling and model transformation method *Computer Engineering and Applications* **48**(1) 76-80
[20] Qi Tie-lin, Li Ya-fen, Wang Pu 2011 Research on model transformation method in model transformation platform of MDA *Computer Engineering and Design* **32**(1) 202-5

Authors

Qingfeng Wu, September 20, 1977, China.

Current position, grades: associated professor at the Xiamen University, the Vice Director of the Software Engineering Lab.
University studies: BS degree in system engineering, PhD degree in Computer Science from Xiamen University in 2000 and 2007, respectively
Scientific interest: software engineering; visual information processing; computer vision.
Publications: 50 papers, 2 patents in computer vision.

Minwen Wu, December 13, 1989, China.

Current position, grades: Master.
University studies: BS degree and M.S. degree in Computer Science from Xiamen University in 2012 and in 2014, respectively.
Scientific interest: software engineering, information management, digital media technology.

Huailin Dong, September 05, 1957, China.

Current position, grades: professor at the Xiamen University, Vice Dean of the Software School.
University studies: BS degree in Mathematics from Xiamen University in 1980.
Scientific interest: software engineering; image processing; information retrieval.
Publications: more than 70 papers.