

The Proposed Blended-MDA for Software Modeling in Architecture Phase

Sidra Sultana¹, Fahim Arif², Iqra Javaid³

¹(Department of Computer Software Engineering, MCS/ NUST, Pakistan)

²(Department of Computer Software Engineering, MCS/ NUST, Pakistan)

³(Department of Computer Software Engineering, MCS/ NUST, Pakistan)

ABSTRACT: The Model Driven Architecture is the modeling approach that ensures the re-usability, portability and inter-operability of the software. For the structuring of the system it provides basic guidelines to be modeled. Separating the architecture from the design is one of the main aim of MDA. Functional requirements of the system are being analyzed by the detailed design of the system (for instance, use cases) and for non-functional requirements, the infrastructure is provided by the architecture of the software ensuring performance, reliability, scalability etc. In this research paper, some case studies are extensively analyzed in the domain of web applications. Functional and non-functional aspects are analyzed in the comparative analysis table. Finally, the Blended-MDA Framework is proposed for the case study entitled as Online Event Manager.

Keywords: Model Driven Architecture (MDA), Computation Independent Viewpoint (CIV), Platform Independent Viewpoint (PIV), Computation Independent Model (CIM), Platform Specific Viewpoint (PSV), Platform Independent Model (PIM), Platform Specific Model (PSM), Portability, Reusability, Inter-operability, Code Generator

I. INTRODUCTION

Object Management Group (OMG) proposed the Model Driven Architecture in 2001. Requirements of the software system and its environment are focused by Computation Independent ViewPoint (CIV). Platform Independent Viewpoint (PIV) helps portability by providing operations independent of the technology platform under use and Platform Specific Viewpoint (PSV) handles system specific concerns by providing system related portability configurations [1]. Description of the domain and its related requirements is handled by Computation independent model (CIM). Plat Independent Model (PIM) is the component of MDA that provides the representation of the conceptual design considering the functional requirements of the system and later on technological (web, mobile etc.) and architectural changes can be incorporated through this vital component of MDA. Platform specific model (PSM) elaborates the system specific configuration of the operations as per multiple platforms, the specific aspect of design are handled by PSM that depends on the system customization and specific operations.

Section II offers a brief literature survey. The comparative analysis is performed in Section III. The Proposed Framework is presented in Section IV. Section V provides a Conclusion. Future Work is given in Section VI.

II. LITERATURE SURVEY

In [1], the authors addressed that MDA tool set provides set of modeling notations for specifying different layers of an enterprise application and a set of code generators that transform these models into platform-specific implementations. Paper discussed experiences of introducing MDA approach, supported by MDA tool set, in an organization that provides a wide variety of IT consultancy and software development services. An application is treated in terms of models for different architectural layers which are application, database and GUI layer. Moreover, encoding of design strategies, guidelines and best practices into the code generators resulted in uniformly high code quality.

In [2], a commercial bank, known as prosperous bank, aims at developing an Enterprise application integration (EAI) project to facilitate its users with functionalities like integrating transactional messaging system with other systems such as legacy applications database storage, messaging systems. Functional requirements of EAI are

- a) Transaction Integrity
- b) Performance
- c) High Availability
- d) Flexibility
- e) Consistent design

In context of implementing Enterprise Web Application, [3] follows the Model Driven Development Methodology. The Platform Independent Model of systems can be used to create the ADI model, an abstraction of the interaction between the browser and the business logic of the system. In order to analyze the ADI, author applied a further MDA transformation to create a corresponding Alloy model.

The development process that conforms to the framework in [4] is decomposed into following steps: Firstly, a general business function PIM is built using the EDOC profile. According to the functional decomposition, PIM is broken down into sub PIMs where each of the PIMs will serve independently and thus can be further utilized to be implemented in a web service. Afterward, it transforms each of these sub EDOC PIMs into interface model of its corresponding web service. Finally, implemented each sub EDOC PIM, through a series of model transformation, on specific platforms on which these services will be implemented.

In [5], goals are achieved with the help of multiple applications involved that ultimately results in a heavy and complex system. Business system has GUI connected with database repository for storage purposes. Model View Controller (MVC) is a basic software development approach adopted for such applications. Using traditional techniques, problems like maintenance and requirement updating arises once the system is deployed. An approach has been purposed in the paper base on UML profiles creating models. Analysis model comprise of stake-holder's requirements, application component structure and communication. Design model created by embedding more implementation details to the analysis model. Domain data model is created as a further detailed extension of analysis or design model and can be transformed to PSM.

In context of [6], developing E-learning system using traditional techniques came up with the problems of interoperability as well as portability issues. Learning Technology System Architecture (LTSA) is a high level layered architecture for developing such information systems. J2EE architecture is also used for developing E- Learning systems but if it is needed to port such E-Learning system to a different platform then it will be time consuming and heavy budgeting is required to carry out such task. To solve this problem MDA based E-Learning system has been purposed.

III. COMPARATIVE ANALYSIS

The research is based on critically analyzing six research papers in terms of identifying the parameters related to the implementation of MDA in Web application and then tracing those parameters in the selected papers. Comparative analysis is presented in a tabular form followed by the brief description of each technique and the illustrative example overview presented in the paper. According to [1], MDA approach benefits enterprise products mostly resulting in easier change management, improved platform independence, better code quality and increasing productivity but not suitable for small and medium size projects. Instead, a meta data-driven aspect oriented approach was found more suitable in these projects. This paper concluded that the Focus of MDA community is on standardizing modeling notations and transformations in order to facilitate tool interoperability but it should now move towards evolving suitable development methods which should also address needs of small to medium sized projects. MDA process used by prosperous bank [2] is depicted by the Fig. 1,

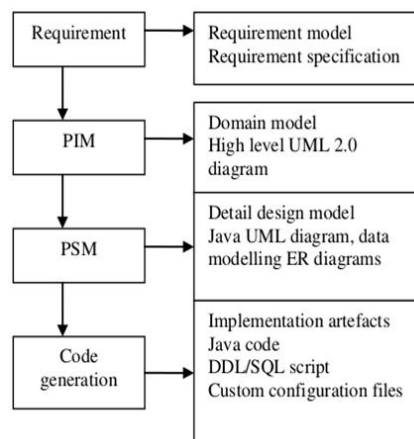


Fig. 1 MDA in prosperous bank [2]

While analyzing the ADI, it is possible to identify a group of bugs, such as the Amazon bug. Finally the approach presented in the paper is explained via an example of an e-commerce system i.e. an Online Book store system. In the case study different UML diagrams are used and in the class diagram the information that belongs to the business logic is stereotyped as entity. The aim is to analyze the interaction of the user, through the web browser, with the business logic of the online bookstore system. The Browser is a class that represents the web browser that the user accesses for the web site. A major requirement of the model depicted is to guarantee the integrity of the system by ensuring that the contents of the shopping cart and the list of items on the browser are identical.

The E-Library provides readers with online literature searching, books borrowing, VOD service, net resources searching, and disk resources searching. Readers can make use of the E-Library via the Internet, and pay for it via bank. Firstly, build a PIM using the EDOC profile to describe the general functions of the E-Library. Secondly, break down the general PIM into sub PIMs each of which can provide service independently. Thirdly, pre-process these sub PIMs and add necessary details to them. Finally apply the transformation rules to get PSMs. Comparative analysis of MDA implementation on various web application is tabulated in Table I.

Table I Comparative Analysis Of Mda Implementation On Various Web Applications

Paper Title	MDA and Analysis of Web Applications	A Model Driven Development Framework for Enterprise Web Services	Introducing MDA in a large IT consultancy organization	Successful Implementation of MDA in a commercial bank	MDA-based Development of E-Learning System	MDA Approach for Maintenance of Business Applications
Case Study Title	Online Bookstore	E-Library System	Implication of proposed approach on enterprise product line, several large and some small projects	Enterprise application development	E-learning System	Verification Office Application system.
Approach	Use of Alloy analyzer	Transformation from EDOC (Enterprise Distributed Object Computing) Model to web service interface model	Introduction of MDA toolset in IT consultancy and software development service provider organization	High (decouples business domain from platform specific details)	Use of learning technology system architecture(LTSA)	Semi automated code generation.
Generality	High	Can work with heterogeneous and distributed environment	Low	Low	Moderate	High
Portability	Moderate	High	Moderate	Moderate	High	High
Reusability	High	High	Moderate	Moderate	High	High
Interpretability	Low	Moderate	Low	Moderate	Moderate	High
Productivity	Moderate	High	High	High	Moderate	High
Code Quality	Moderate	Very High	High	Not mentioned	Moderate	High
Trace ability	Not Mentioned	Not Mentioned	Not Mentioned	Not Mentioned	Not Mentioned	Low
Maintainability	Low	High	High	Moderate	Low	Low
Level of interaction between business logic and Web browser	High	Very High	Low	Low	Low	Low
Automated support tool	UML2Alloy: transformation from the UML to Alloy ArgoUML	Transformation rules are applied	Not Mentioned	Not Mentioned	ArcStyler	Atlas transformation language (ATL) tool. MOFScript tool.

Verification Office Application [5] is taken as a business application case study where the employees at a medical lab require a system to manage the data regarding customers, billing information, test results, medical reports and devices. The application based 2-tier architecture where first tier is a user interface and the second tier is based on database manipulation functions where communication between them is carried out by domain data objects in first tier. Further automated support is provided for model-to-model and model to semi-automated code transformation using Atlas transformation language (ATL) tool and MOF Script tool respectively. The benefits of the purposed approach are the maintenance and ease of system updating as well as providing traceability between high-level design changes.

IV. PROPOSED FRAMEWORK

For applying the findings in a framework for a proposed case study, authors have considered an event management website which provides the facilities to create a personalized event calendar after creating an account. Different types of accounts are there, which are organizational account, personal account and student account. Back-end functionalities are the administrative control including updating, editing and deleting i.e., canceling event and account etc. At CIM level these requirements are gathered independent of their technicalities related to domain and implementation, then the analysis of these requirements lead to some model having design details independent of platform and implementation language/ tools details. PSM of the system represent fine-grain details of framework to be used. Finally code generator results in a software application facilitating web application.

There are some queries as mentioned below that requires answers in terms of explicitly mentioning the functional and non-functional requirements in MDA framework

1. What are the possible most occurring threats in a way to implement MDA in this specific project?
2. If the policies defining the events in case of organizational account changes then how could the modifications be propagated to the design of MDA so that the events remain unaffected?
3. If the domain of a person in case of personal account is modified then how the domain requirements could ensures the conformance with the accurate event management?
4. How to ensure efficient access to a database having millions of record to be searched through in order to check the availability of venue and time for a certain newly added event in case of adapting pure MDA approach?

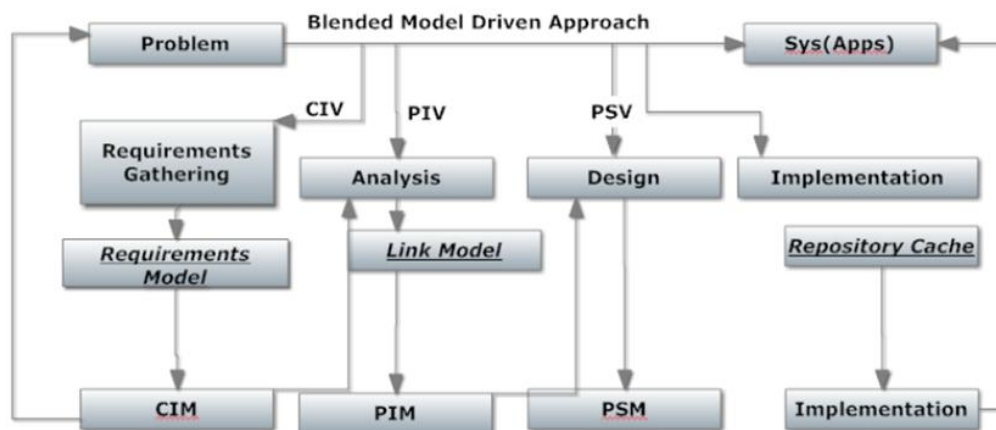


Fig. 2 Blended Model Driven Architecture

Fig. 2 answers all of the above questions that depict a need of some blended MDA approach that is the need of hour to add the efficiency, availability and modifiability quality attributes with the three basis quality attributes that are portability, inter-operability and re-usability provided by adapting MDA approach.

Link Model is there to separate the business policies from the design model at PIM level to facilitate modification and evolution of software. Changes in business policies in case of all three types of user accounts is silent with respect to the functionality degradation of software. There should be some cache table to be introduced in the database that have all the important fields of different tables of the database to ensure efficient searching instead of searching multiple fields in multiple table. The cache table is there in the central repository called RUBRIC model. To facilitate the domain level changes that are the requirement management, model based approach in requirement gathering process is required to map the model of requirements to the model of design. Thus the blended MDA will take the above mentioned shape shown in Fig. 2.

V. CONCLUSION

The research work proposed facilities the main stream of CMS (Content Management Systems) by providing additional quality measures like efficiency, availability and modifiability by embedding the existing software engineering, software design and software architecture concepts in such a way that meta models of whole MDA is preserved. Each model is thus providing a loose coupling with strong cohesion aimed at a good software architecture approach. Good software is the one providing conformance to user requirements thus meeting the quality needs of the system. In order to make MDA more approachable and implementable in every domain, it could be tested against other quality attributes like feasibility, testability, maintainability, extensibility and scalability, while keeping the complexity measure as low as possible.

REFERENCES

- [1] Kulkarni, V.; Tata Res. Dev. & Design Centre, Pune; Reddy, S.;: Introducing MDA in a large IT consultancy organization, IEEE, Software Engineering Conference, 2006. APSEC 2006. 13th AsiaPacific, 2006, 6-8 Dec. 2006, pp. 419 - 426.
- [2] Choong Koon Fong.;: Successful Implementation of Model Driven Architecture, APAC Support Center, 2007, A Borland White Paper.
- [3] Behzad Bordbar; Kyriakos Anastasakis, MDA and Analysis of Web Applications, School of Computer Science, University of Birmingham, Birmingham, 2007.
- [4] YU Xiaofeng; HU Jun; ZHANG Yan et al, A Model Driven Development Framework for Enterprise Web Services, Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference, EDOC, 2006, pp. 75-84
- [5] Mila Keren; Andrei Kirshin; Julia Rubin; Ahto Truu, MDA Approach for Maintenance of Business Applications, Model Driven Architecture –Foundations and Applications, Lecture Notes in Computer Science Volume 4066, ECMDA-FA 2006, LNCS 4066, pp. 40 – 51
- [6] Hong Wang ; Carleton Univ., Ottawa, Ont., Canada ; Dong Zhang ; Jun Zhou, MDA-based development of e-learning system, Computer Software and Applications Conference, 2003. COMPSAC 2003, pp. 684 - 689
- [7] John D. Poole, Model-Driven Architecture: Vision, Standards And Emerging Technologies, ECOOP, 2001.
- [8] Youcong Ni ; Wuhan, China ; Shi Ying ; Linlin Zhang ; Jing Wen , Modeling Aspect-Oriented Software Architecture, Industrial and Information Systems, International Conference on Industrial and Information Systems, 2009, IIS '09, pp. 108 - 113
- [9] Jiang Guo, An approach for modeling and designing software architecture, Engineering of Computer-Based Systems, 2003. Proceedings. 10Th IEEE International Conference and Workshop, pp. 89 - 97
- [10] Jiang Guo ; Yuehong Liao, The scheduling algorithms in software architecture modeling, Engineering of Computer-Based Systems, 2004. Proceedings. 11th IEEE International Conference and Workshop, pp. 36 – 43
- [11] Ortega, D. ; Silvestre, L. ; Bastarrica, M.C. ; Ochoa, S.F., A Tool for Modeling Software Development Contexts in Small Software Organizations, Chilean Computer Science Society (SCCC), 2012 31st International Conference, pp. 29-35
- [12] Chang-Guo Guo ; Xiao-Ling Li ; Jun Zhu., A Generic Model for Software Monitoring Techniques and Tools, Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference, pp. 61-64
- [13] Alves Pereira, J. ; Souza, C. ; Figueiredo, E. ; Abilio, R. , Software Variability Management: An Exploratory Study with Two Feature Modeling Tools, Software Components, Architectures and Reuse (SBCARS), 2013 VII Brazilian Symposium, pp. 20-29
- [14] T. K. Zirkel, S. F. Siegel, and T. McClory, Automated Verification of Chapel Programs Using Model Checking and Symbolic Execution, NASA Formal Methods Lecture Notes in Computer Science Volume 7871, 2013, Pp 198-212
- [15] D. Remenska, J. Templon, T. A.C. Willemse, P. Homburg, K. Verstoep, A. Casajus, and H. Bal, From UML to Process Algebra and Back: An Automated Approach to Model-Checking Software Design Artifacts of Concurrent System, NASA Formal Methods Lecture Notes in Computer Science Volume 7871, 2013, Pp 244-260.
- [16] S. Lyde, M. Might, Extracting Hybrid Automata from Control Code, NASA Formal Methods Lecture Notes in Computer Science Volume 7871, 2013, Pp 447-452
- [17] N. Decker, M. Leucker, and D. Thoma, jUnit RV —Adding Runtime Verification to jUnit, NASA Formal Methods Lecture Notes in Computer Science Volume 7871, 2013, Pp 459-463.
- [18] Shivers, O.G.: Control-Flow Analysis of Higher-Order Languages. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA (1991)
- [19] Van Horn, D., Might, M.: Abstracting abstract machines. In: ICFP 2010: Proceedings of the 15th ACM SIGPLAN International Conference on Functional Programming, pp. 51–62. ACM Press (2010)
- [20] Cousot, P.: Integrating physical systems in the static analysis of embedded control software. In: Yi, K. (ed.) APLAS 2005. LNCS, vol. 3780, pp. 135–138. Springer, Heidelberg (2005)
- [21] Cousot, P., Cousot, R.: Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In: Conference Record of the Fourth ACM Symposium on Principles of Programming Languages, pp. 238–252. ACM Press, New York (1977)
- [22] Bouissou, O.: From control-command synchronous programs to hybrid automata. In: Analysis and Design of Hybrid Systems, pp. 291–298 (2012)
- [23] Najafi M., H. Haghighi: A formal mapping from Object-Z specification to C++ code, Scientia Iranica D (2013) 20(6), pp. 1953-1977
- [24] Najafi M., Haghighi H., Zohdi N. T.: A survey on formal, object-oriented program development approaches, Scientia Iranica D (2015) 22(3), pp. 1001-1017
- [25] M. Bashiri and S.G. Miremadi: Perform-ability guarantee for periodic tasks in real-time systems, Scientia Iranica D (2014) 21(6), pp. 2127-2137
- [26] Ralf W., Nils J., Erika A., Joost-Pieter K. High-Level Counter examples for probabilistic automata, Logical Methods in Computer Science Vol. 11(1:15)2015, pp. 1–23
- [27] Rahim, M.A.B., Arif, F.: Translating Activity Diagram from Duration Calculus for Modeling of Real-Time Systems and its Formal Verification using UPPAAL and DiVinE, Mehran University Research Journal of Engineering & Technology, Volume 35, No. 1, January, 2016, pp. 139-154
- [28] Rahim, M.A.B., Ahmad, J., and Arif, F.: Parallel verification of UML using DiVinE tool, 5th International Conference on Computer Science and Information Technology, Amman,