



ISSN: 2319-5967

ISO 9001:2008 Certified

International Journal of Engineering Science and Innovative Technology (IJESIT)

Volume 2, Issue 2, March 2013

# Software Development Life Cycle: Enterprise Requirement Mapping

NoumanMaqboolRao<sup>\*</sup>, Majid Zaman<sup>\*\*</sup>, Muheet Ahmed Butt<sup>\*\*\*</sup>

<sup>\*</sup>Higher Education Commission (HEC) of Pakistan, Islamabad, Pakistan.

<sup>\*\*</sup>Scientist, Directorate of Information Technology, University of Kashmir, Srinagar, J&K, India.

<sup>\*\*\*</sup> Scientist, PG Department of Computer Science, University of Kashmir, Srinagar (J&K), India

*Abstract - Model Driven Architecture (MDA) is an conceptualization for determining scheme in terms of models, and then represent the evolution procedure through activity model transmutation. The System Development Life Cycle (SDLC) is a procedure for assuring that all structural and user necessities and authority strategical goals and objectives are met. The SDLC furnish a organized and replaceable procedure for all state of any scheme improvement effort. A generic lifecycle for MDA-based software development that can be used as a basis for conception of MDA-based method through a Method Engineering (ME) process has been proposed. This generic life cycle is studied and analyzed in this paper.*

**Index Terms— Model Driven Architecture, SLDC**

## I. INTRODUCTION

The MDA [1] is a new way of development use and compositions qualify, supported on a platform-independent model (PIM) of the utilization or description concern practicality and conduct. An absolute MDA description consists of a explicit platform-independent base framework, plus one or more platform-specific models (PSM) and sets of computer programs definitions, each depicting how the base model is implemented on a assorted middleware platform. A complete MDA usage consists of a explicit PIM, plus one and complete execution, one on each level that the utilization developer determine to support. MDA development centering introductory on the practicality and behavior of a apportioned utilization or system, undistorted by idiosyncrasies of the technology platform or platforms on which it will be implemented. In this way, MDA disassociate execution details from business functions. The SDLC is System Development Life Cycle [2], defined by many other ways by different authors. The SDLC propose the usage of a step by step formulation to evolve a scheme. The SDLC is having six to eight phase but it changes between them by different content; In order to modify the improvement procedure and produce choice product SDLC permit the expert to continue step by step for the investigation and arrangement of the information system. The expert must start from phase first and continue next simultaneously. Each phase having enquiry, analyst has to reply the query to accomplish the outcome in successful way. Each step execute reusable activities, although the activities are separate from each other in a series but in actual cognizance they are extremely associated. Activities in all phase are sporadic but if in a phase difficulty happen then it might necessitate alteration by the expert in preceding stage to resolve it, each phase is distinct individually to execute circumstantial act but interconnected as the choice outcome cannot be bring forth without the action among them. It is more appropriate to say that activities in SDLC are performed in full swing overlap each other to make a useful quality end product.

## II. MODEL DRIVEN ARCHITECTURE (MDA)

The MDA is a model driven architecture, is a software development framework defined by OMG supported on automated transmutation of framework. By using modeling languages as declaratory scheduling communication the consequence produced are elevation of abstract level for programmers and reinforced productivity, choice, longevity outlooks. The MDA is brought up to sphere assets in case if level changes it should not affect technology. MDA render the result by separate content and processing philosophy from engineering particular. The MDA usage affects Platform Independent Model (PIM), Platform Specific Model (PSM) and code. To correspond various exemplary aggregate instrument are used, like Unified Modeling Language (UML) Meta-Object Facility (MOF) and Common Warehouse Meta-model (CWM) etc. MDA is model-driven because it renders a way for exploitation of models to straightforward the course of study of understanding, plan, expression, deployment, action, upkeep and alteration. MDA Based methodologies also used Situational Method Engineering (SME).The use of SME is to fit the undertaking accordant to definite situation. There are different types like



ISSN: 2319-5967

ISO 9001:2008 Certified

International Journal of Engineering Science and Innovative Technology (IJESIT)

Volume 2, Issue 2, March 2013

Paradigm based SME, Generic instantiable process lifecycle and the Extension based approach. The detail of CIM, PIM, PSM and code generation is given below:

**A. CIM**

The CIM seizure the business cognition without citation to a peculiar system execution or technology, and it would stay the same even if the system were enforced automatically, or in computer software. The CIM is represented with a cognition that is acquainted to business people. Therefore, it plays an essential role in construction the gap between proficient about the business, and software engineers.

**B. PIM**

The platform independent model is accountable for ceremonial description of the construction and mapping of a system except the platform detail. Normally UML is used to symbolize it sometime any other tool can be used instead of UML.

**C. PSM**

The platform specific model is accountable for the practicality in a specific platform that is nominal in PIM in general. It takes PIM as its input signal and transmute PIM into PSM. It exercise UML to make Platform specific model. Function is carried out at this stage for transmutation of PIM into PSM accordant to desire platform. The PIM provide the goodness of verification accordant to demand before transforming in to PSM.

**Coding**

At this phase coding is finalized. An MDA instrument creates most of the execution of code selected by the developer and some portion is done extremity by the developer. Finally the end product is produced.

**III. MAPPING OF SDLC WITH MDA-SDLC**

The generic life cycle focused on the procedure portion of methodological analysis. The generic SDLC dwell of five stage and activities in them. Each stage in a chronological sequence have its own activity to execute. Project Initiation Phase, PIM development Phase, PSM and Code development phase, Deployment Phase, Maintenance Phase are the stages of having stages and activities in them. The mapping of MDA-SDLC with SDLC is given below. MDA-SDLC is not a solid methodology but it's a general process.

**Table 1: Mapping of SDLC with MDA-SDLC**

1) <b>Recognition of need preliminary survey / initial investigation</b>	<b>Phase 1) "Project Initiation"</b> 1) CIM definition  2. Requirement Specification 3. Obtaining Funding & Support (resources) 5. Define General Plan
2) <b>Feasibility Study</b>	
3) <b>Analyzing</b>	<b>Phase 2) "PIM Development"</b> 1. Produce Analysis PIM (System Functionalities) 2. Architectural Design
4) <b>Design</b>	<b>Phase 3) "PSM &amp; Code development phase"</b> 2. General Code 3. * Test
5) <b>Implementation</b>	<b>Phase 4) "Development"</b> 1. Transition of the system to the user environment. 2. * Finalizing of the system and user documentation.
6) <b>Post-Implementation</b>	<b>Phase 5) Maintenance</b> 1. * Maintenance



ISSN: 2319-5967

ISO 9001:2008 Certified

International Journal of Engineering Science and Innovative Technology (IJESIT)

Volume 2, Issue 2, March 2013

#### IV.SHORTCOMINGS IN MDA-SDLC

Mapping of MDA-SDLC with orthodox SDLC displays that following activities/phases are not enclosed in the MDA-SDLC i.e. these are the major shortcomings of MDA-SDLC.

##### **A. Modification and Enhancement Phase**

System should have the capability for change if requisite not only the upkeep characteristic is adequate to make instrumentality efficient and certain. Such as if approaching improvement is requisite or there is a need for up gradation, the scheme should be negotiable adequate to assimilate the alteration. Improvement must take place in such a way that by adding excess characteristic with in the body of actual instrumentality consecutive substance should not be disturbed. Organizational responsibility can alter with times which request for system improvement.

##### **B. Training to User**

Developers follow assorted formulation to form the scheme easy and intelligible for the public users. Software documentation is virtually constantly provided to the user along with the instrumentality there are manual to use the system and for troubleshooting, but the much better conceptualization is the applicative presentation of the system by the developer. Such demonstration and demos delivered by the developer make system very easy to use and manage problem in better way.

##### **C. Quality Check and Evaluation after each Phase**

There essentially be assessment after prescribed intervals. Overall rating after each stage must be transport out by action from user to sustain that improvement procedure is going accordant to his responsibility and to make sure that development unit is on right course and impermanent according to defined standards. In bid to make quality production senior supervisor must be accountable for administration exhibit to realize the current undertaking for new team member .Quality of software should not be compromised by any upset during development. Choice check act after the first stage of MDA-SDLC to make certain that what the person needful developer realize its well, and what developer unit is going to suggest is that which user request for. The choice draft activity will be multipurpose to corroborate that the responsibility exemplary is meeting the modular; project substructure is also up to the mark, the origin for the undertaking funding will also be accessible in near future and over all base content for PIM is from valid beginning and is that which is required.

#### V. CONCLUSION

In this paper conventional SDLC is interpreted as standard. MDA-SDLC is mapped with conventional SDLC. It is recovered that the MDA-SDLC deficiency some very crucial stage or activities such as alteration and improvement stage, user preparation stage and Superior appraisal and assessment stage.

#### REFERENCES

- [1] I. Mukerji, J. Miller, MDA Guide Version 1.0.1, OMG, 2003.
- [2] <http://www.oft.state.ny.us/pmmp/guidebook2/Phase.pdf> .
- [3] A. Gavras, M. Belaunde, L. Ferreira Pires, J. P. Andrade Almeida, "Towards an MDA-based development methodology", Proc. First European Workshop on Software Architecture (EWSA2004), Enschede, Netherlands, 2004, pp. 71-81.
- [4] X. Larrucea, A. B. G. Diez, J. X. Mansell, "Practical Model Driven Development process", Proc. Second European Workshop on Model Driven Architecture (MDA), Canterbury, UK, 2004.
- [5] T. Hildenbrand, and A. Korthaus, "A Model-Driven Approach to Business Software Engineering", Proc. 8th World Multi-Conference on Systemic, Cybernetics and Informatics (SCI 2004), Volume IV Information Systems, Technologies and Applications, Orlando, Florida, USA, 2004, pp. 74-79.
- [6] M. Gervais, "ODAC: An Agent-Oriented Methodology Based on ODP", Journal of Autonomous Agents and Multi-Agent Systems, 7(3), 2003, pp. 199-228.
- [7] M. Gervais, "Towards an MDA-Oriented Methodology", Proc. 26th Annual International Computer Software and applications Conference (COMPSAC'02), Oxford, England, 2002.
- [8] S. Kim, H. G. Min, J. S. Her, S. H. Chang, "DREAM: A practical product line engineering using model driven architecture", Proc. ICITA '05, Australia, 2005, pp. 70-75.
- [9] N. Guelfi, R. Razavi, A. Romanovsky, S. Vandenberg, "DRIP Catalyst: an MDE/MDA Method for Fault-tolerant Distributed Software Families Development", Proc. OOPSLA & GPCE 2004 workshop on best practices for Model Driven Development, 2004.
- [10] Mohsen Asadi, MahdyRavakhah, Raman Ramsin, An MDA-based System Development Lifecycle, Second Asia International Conference on Modeling& Simulation, Kuala Lumpur, Malaysia 13 – 15 May 2008.