MDA Tool Support for Model Driven Software Evolution: A Survey

Madhavi Karanam
Computer Science and Engineering Department,
Sri Venkateswara Engineering College, Suryapet, Telangana, India
Email: bmadhaviranjan@yahoo.com

Abstract: ***The explosive growth of model driven approaches and the emergence of Model Driven Engineering (MDE) have led to the Model-Driven Software Evolution (MoDSE) which is a new paradigm for software evolution. Growing interest in the model driven approaches has largely increased the number of tools into the development environment. Using the modeldescribed by analysts, software tools will do the rest of the task,generating software that will comply with customer's defined requirements.Previous research has shown that the user often do not use or know all of the tools available in the model evolution environment that they regularly use. The common solution to this problem is to provide a means to search through passive help documents. So, choosing the right tool has become very much essential because of the diverse tools usage for MoDSE tasks.The main aim of this paper is to explore the MDA tools to know the automation support provided by these tools in model driven evolution process. It is also aim to know the general list of tools, ease of use, which meets the requirements, support for platform independent models, free and commercial tools etc. It is observed from the survey of various tools that no single tool serves the purpose and there is an enough automation support for activities of Model Driven Software Evolution.***

Key Words: Model Driven Software Evolution, Model Driven Architecture, MDA Tools.

I. INTRODUCTION

Model Driven approaches have become a new software trend in software development process. MDE needs a new paradigm for software evolution which is known as MoDSE [1]. Many CASE tools have evolved due to wide usage of model driven approaches. Tools are used for different activities of model driven evolution such as model transformation, model mapping etc. The contradictory experiences with MDA and UML tools appear puzzling and difficult to interpret. Tools do much work in model driven approaches [7]. So, it is very much essential to choose the tools carefully. Basically, an MDA tool is a tool used to develop, interpret, compare, align, measure, verify, transform, models or meta models. In MDA approach we have essentially two kinds of models: *initial models* are created manually by human agents while *derived models* are created automatically by programs. For example, an analyst may create a UML initial model from its observation of some loose business situation while a Java model may be automatically derived from this UML model by a Model transformation operation which can be done with the help of automated tools. These tools perform more than one of the desired functions. For example, some creation tools may also have transformation and test capabilities. There are other tools that are solely for creation, solely for graphical presentation, solely for transformation, etc.

There is an increasing need for more disciplined techniques and engineering tools to support a wide range of model evolution activities, including model-driven software evolution, model differencing, model comparison, model refactoring, model consistency, model versioning and merging, and (co-)evolution of models.The research presented here suggests that focusingthe specific outcome expectations, features, categories and characteristics of various MDA tools, it may be able to make sense of the apparently inconsistent findings. This paper describesa conceptualized review of tools which automates the model driven evolution process. Such a perspective allows users to anticipate, explain, and evaluate different experiences and consequences following the introduction and intention of the tools.However, in MoDSE automation required for tasks such as model mapping, model merging, model integration, model transformation, model consistency etc, are identified in [1, 2,6, 7]. Much of the literature on these tools has intended to focus on discussion forums, panels, comparison strategies and frameworks. Discussion forum and/or panel are where the users can share their ideas and answering the questions of the audience [15,16, 17, 18]. Framework determines the comparison strategies for features of tools of same category under a uniform platform [2, 3, 5, 6, 8, 10,12]. Thus, this paper is intended to provide the survey on various MDA tools,evaluation criteria to compare, assessment of thesetools and their capability for model driven evolution automation.

The remainder of the paper is structured as follows. Section 2 reviews the related work. Survey on MDA tools for model driven evolution automation which is the primary contribution of this paper is discussed in the section 3. Section 4 closes with conclusions and future work.

II. RELATED WORK

There have been various studies characterizing and/or comparing MDA/MDE/UML tools for model driven evolution purpose. Most of these are focused toward limited number of tools that are discussed, evaluated and compared.Existing studies for comparing and evaluating MDA tools are reviewed in this section.

Clic. T.et.al, proposed an evaluation framework for MDA tools in [2] and outlines with a set of software requirements. Evaluation Criteria consists of six criteria groups and each criteria group is expanded into a set of 54 detailed subcriteria.They also came up with the "portrait" of an ideal MDA tool.

The purpose of the study conducted in [4] is to compare general purpose tools available that enable to put in practice the principles and aimed at generating a wide variety of applications composed by interactive multimedia and artificial intelligence components. This study also discusses the technical features like purpose, language, platform, meta models, etc and identifity features such as license, release updated, last year release etc. But in this paper only four tools AndroMDA, ArcStyler, Borland Together, and Eclipse Modelling were discussed.

Integrated constrain support in MDA tools evaluated in [5]. The different tools were classified in Categories like CASE tools, MDA specific tools, MDD methods and OCL tools. MDA-tools considered in the classification are closest to the MDA standard. Only few tools such as Poseidon, Rational Rose, Magic Draw, Objecteering/UML , Together ,ArcStyler, OptimalJ and AndroMDA etc, have selected for the comparison and evaluation purpose. The support of current tools regarding the automatic generation of the code required to enforce the Integrated Constraints specified in a PIM also surveyed. The main shortcomings encountered are the lack of expressivity and efficiency in integrated constraints.

MDA tools are categorized in three ways in [8]. The first, whether the tool is open source or commercial, will help to choose a tool that is right for the culture of an organization, among other things. The second, whether the tool offers a partial or complete MDA solution, will helps in such considerations as cost, quality, and flexibility. The final category, whether the tool generates code from the model or executes the model. It is not mentioned what are the tools that fits in to the specified categories.

A short comparison of the three MDA tools was presented in [11]. It was focused on the concepts behind the tools as well as how to use them. The tools considered for comparison are ArchitectureWare, AndroMDA, and openMDX. The comparison strategy presented might be useful for evaluation purposes to find out about the differences in approach, features and concepts which are to be considered as the implementation of OMG's MDA specification.

IBM Rational software has several products that support MDA and Model Driven Development (MDD) in varying capacities [13]. These tools fall into three basic categories such as general-purpose, domain-specific, and supporting. For example Rational Software Architect is in general-purpose category, IBM Rational Systems Developer in domain-specific category, and IBM WebSphere Business Modeler in the supporting category. The usage of these tools in MDA was described. Here only IBM products are considered for categorization.Tariq and Akhter in [14] established a set of characteristics which are essential to understand a MDA tool. This study was applied to 10 commercial tools.

Computer Aided Software Engineering tool Community [15] is an open access web application. In which the different categories of the CASE tools like MDA, UML, reverse engineering, agile modeling etc. are listed. The key functions, external links, and rating of the tools are provided. modelbased.net forum [16], Objects by Design Forum [17] and UML Forum [18] are dedicated to tools and information related to model-driven system development, aiming at supporting OMG's vision MDA are provided.These forums/communities provide the overview and resource links of MDA oriented tools, UML, MOF, and Model transformation tools.

Above mentioned frameworks, evaluation criteria's, forums, panels and communitiesprovide only specific features/characteristics/evaluation criteria's of MDA tools in different aspects. But, there is no specific study in model MoDSE and tools support for various activities. Therefore, this paper aims to discuss a huge list of tools which are available in current market for various activities of MoDSEautomation specifically.

III. MODEL DRIVEN SOFTWARE EVOLUTION AUTOMATION

A widely accepted statement is that tools are needed in order to put in practice in an effective way all the techniques involved in the construction of a Model Driven Software Development (MDSD)-based method. Without tools which automate the steps that must be carried out during the application of such kind of methods, most of the promised benefits can not be obtained [21]. This statement reveals the vital role of the automated tools in any software engineering task. So, this paper aims to collect information about the various MDA tools.

These tools are considered for selection, assessment and comparison under a uniform platform. Only MDA tools are considered in this paper because of suitability of these tools in MoDSE activities.

List of all the MDA tools which have not covered in previous works have chosen in this survey. Tools which believed that meet a series of requirements are suitable for automating model driven evolution tasks are considered in this paper. An extensive research has done to put together all the tools which are available in the market as MDA tools. Although both MDA and MoDSE are emerging technologies, in the last few years its popularity caused the creation of a considerable number of tools based on it. However, this survey provides the various activities involved in the model driven evolution process and the tools automation support.

A. Model Driven Software Evolution Activities

This section describes brief introduction, and diverse activities which are collected from the existing literature.While MDE promises to improve productivity and maintainability, wide spread adoption and scaling to large software systems needs research into evolution of model based systems, scope and expressivity of modeling languages, and into interaction and integration of models. Thus, MDE requires a new paradigm for software evolution which could lead to an emergence of Model-Driven Software Evolution (MoDSE). Deursen et.al, have also brought a research proposal which is entitled as 'Model-Driven Software Evolution' [1]. The research projects such as MoDSE project at Delft University of technology, Netherlands, ARC project at University of MONS, Belgium and MODEVO project at Uuniversity of Limerick, Ireland are focus to develop an approach includes methods, techniques, tool support, and stakeholder's concerns, views etc. to understand MoDSE. So, there is an increasing need for more disciplined techniques, approaches and engineering tools to support a wide range of model evolution activities including model-driven software evolution, model difference, model comparison, and model merging etc.

A single model is not sufficient and several models of different modeling languages as well as some code in programming languages are essential to develop an entire software system. Due to variety of models and modeling languages, there is a need to have model interaction, integration, mapping and transformation among the diverse models. In this manner model-driven engineering abandon from traditional software engineering with its massive development platform. In place of one or two programming languages MDE bring forward a multitude of languages that are themselves artifacts of development process. When models are incrementally introduced in legacy system, models need to interact with legacy code. Thus, there is a need to have an interaction between model and code. This kind of interaction can be provided thorough an interface in any modeling language. Hence, there is a need to have an interaction between models.

Model-Driven Software Evolution requires many types of transformation like model to code, model to model, and code to model. These transformations are done by using transformation language by the choice. Therefore, major activities of MoDSE are model interaction, model integration, model mapping, model transformation, and model merging etc are considered in this paper.

B. MDA Tools

There is an increasing need for more disciplined techniques and engineering tools to support a wide range of model evolution activities as mentioned in above section. Many CASE tools have been evolved due to wide usage of model driven approaches. Tools are needed at different stages of model driven evolution. So, here the question arises "how do you choose the right tool?" Tools do much work in model driven approaches. So, it is very much essential to choose tools carefully.

MDA tools have a major role in MDA-based software development such as Model Driven Software Development and Model Driven Software Evolution etc. They provide development of software applications by creating a PIM model of a system and its transformation and mapping it into the PSM model. MDA tools also provide further transformation of the PSM model to the fully executable program code in several programming languages. Currently, the OMG did not specify a document that indicates which features a modeling tool needs to incorporate to be MDA-compliant. However, according to the OMG website [19] today on the market there are over fifty MDA tools that support one or more major features of the MDA approach. Each of them has different strengths, so it is up to developers to choose a tool that best fits their needs. More number of tools which are not yet covered in previous research works is reviewed in this paper to implement this study.

C. MDA Toolsfor MoDSE Activities

This section discusses the various MDA toolsand their automation support for activities of MoDSE process. Majority ofMDA tools are considered for this survey. Tool name, MoDSE activity, and its automation support for model driven evolution process is described in table 1.

IV. CONCLUSIONS AND FUTURE WORK

This survey has described the MDA tools support for various activities of model driven software evolution. An extended survey also investigated several MDA tools available in the market toverify the automation support for various activities. So, in this paper major MDA tools which are not yet covered in previous research works and their automation support for MoDSE activities are discussed.From this survey it is observed that all tools provide automation for model transformation which is the major activity of model driven evolution. But for the code transformation only the programming languages differ from one tool to other. Moskitt tool is named as MDA tool but it supports only UML diagrams. It is also noticed that UML models are also essential for model driven evolution. So, majority of the tools support UML models as well as other models like BPML, WSDL etc.

From this survey it is believed that the different activities of MoDSE have automated support from different tools. But no single tool supports all the activities still activities like model merging, model mapping etc., have less tool support. So, these kind of activities should be included in more number of tools and it is possible to have an ideal tool for all the activities can be considered as future work.

TABLE 1. MDA TOOL VS MODSE ACTIVITY

MDA Tool Name	**MoDSE Activity**	**Automation**	**Others**
Andro MDA	Model Transformation	PIM to PSM PSM to PIM Supporting Programming languages like Java, C,C++, C#, .Net,PHP	Non-Commercial
Arcstyler	Model Transformation UML 2.3 Profile	PIM to PSM Supporting Programming languages like Java, .Net, CORBA 14 diagrams	Commercial
Acceleo	Model Transformation UML 2.3 Profile BPML Technology Standards Meta Models	PIM to PSM PSM to PIM Supporting Programming languages like Java, .Net, PHP, Python 14 diagrams Business Process Diagram XMI, MOF QVT	Commercial
Aonix	Model Transformation UML 2.3 Profile Technology Standards	PIM to PSM Supporting Programming languages like Java, C++, CORBA , Ada 95 13 diagrams XMI	Commercial
BoUML	Model Transformation Technology Standards	PIM to PSM PSM to PIM Supporting Programming languages like Java, C++, PHP, Python XMI, XML	Non-Commercial
BluAge	Model Transformation	PIM to PSM PSM to PIM Supporting Programming languages like Java, .Net Executable Code generation	Commercial

	Data Modeling UML 2.3 Profile	UML Profile, Oracle 4 diagrams	
BluePrintME	UML 2.3 Profile Model Transformation	10 diagrams PIM to PSM PSM to PIM Supporting Programming language Java Executable Code generation	Non-Commercial
Enterprise Architect	UML 2.3 Profile Model Transformation Data Modeling Technology Standards Testing Strategies Test report generation Model Repository Requirements Engineering	14 diagrams PIM to PSM PSM to PIM Supporting Programming languages like Java, .Net, C++, C#, Visual Basic, Delphi, PHP, Python Executable Code generation UML Profile Oracle, MySQL, MS Access XMI, XML schemas, BPML profile, WSDL profile, SysML profile Unit, Integration, System, Acceptance, Scenario testing RTF, HTML, Excel Sheet, Word UML profile	Community edition, professional edition, enterprise edition
GMT	Model Transformation Technology Standards Meta Models	PIM to PSM XMI schema QVT	Non-Commercial
Kermeta	Technology Standards Meta Models	OCL QVT, DSL, UML, MOF	Commercial
MagicDraw	UML 2.3 Profile Model Transformation Data Modeling Technology Standards Model Synchronization Model Decomposition Model Integration Model Difference Model Search	12 diagrams PIM to PSM PSM to PIM Supporting Programming languages like Java, C++, C# Executable Code generation UML Profile Oracle, MySQL XML schema, BPML profile, WSML profile, OCL Compare two versions of same model	Commercial

	Model Repository		
Md Workbench	Model Transformation Technology Standards	PIM to PSM PSM to PIM Supporting Programming languages like Java XMI,XML schemas	Commercial
Modelio	UML 2.3 Profile Model Transformation Technology Standards Model Search Model Extension Meta models	8 diagrams PIM to PSM PSM to PIM Supporting Programming languages like Java, C++, C# Executable Code generation XML, XML schemas, BPML profile, WSML profile, SysML profile UML meta models	Commercial
Moskitt	UML 2.3 Profile	5 diagrams	Non-Commercial
Modfact	UML 2.3 Profile Model Transformation Model Repository Meta Models	13 diagrams PIM to PSM PSM to PIM Supporting language CORBA MOF, UML meta models	Non-Commercial
Objecteering MDA & Objecteering MDA Modeler	UML 2.3 Profile Model Transformation Model Repository Meta Models Requirements Engineering Technology Standards	14 diagrams PIM to PSM PSM to PIM Supporting language C++, C#, Java MOF, UML meta models UML profile UML, BPML, SysML profiles	Commercial
OptimalJ	UML 2.3 Profile Model Transformation Technology Standards Database Modeling	14 diagrams PIM to PSM PSM to PIM Supporting language CORBA MOF, UML meta models MySQL	Commercial
Powerdesigner	UML 2.3 Profile Model Transformation Technology Standards Database Modeling	14 diagrams PIM to PSM PSM to PIM Supporting languages Java, C#,.Net XML schema, BPML profile MySQL	Commercial

RISE	UML 2.3 Profile Model Transformation Technology Standards Database Modeling	14 diagrams PIM to PSM Supporting languages C#, .Net, PHP XML schema UML profile MySQL	Commercial
StarUML	UML 2.3 Profile Model Transformation	14 diagrams PIM to PSM PSM to PIM Supporting language Java	Non-Commercial
XactiumXMFmosiac	UML 2.3 Profile Model Transformation Technology standards Meta models	9 diagrams Executable code generation MOF, OCL QVT	Commercial

REFERENCES

[1] Arie van Deursen, Eelco Visser, and Jos Warmer.: "Model-Driven Software Evolution: A Research Agenda",In Dalila Tamzalit (Eds.), *Proceedings 1st International Workshop on Model-Driven Software Evolution*, University of Nantes, 2007. pp. 41-49.

[2] Calic T. 2008. Exploration of Model Driven Architecture capabilities via Comparative Utilization of MDA Tools, Proceedings of 5th International Conference on Information Technology, 44-50.

[3] Chatterjee S., 2007. MDA Tools Evaluation, Jax Magazine.

[4] Elias Palacios-Gonzalez, Hector fenandez-Fernandez, et.al. General Purpose MDE Tools. International Journal of Interactive Multimedia and Artificial Intelligence, Vol.1, No.1,ISSN-1998-1660. 72-75.

[5] Jordi Cabot, Ernest Teniente, 2006. Constraint Support in MDA tools: a Survey, Proceedings of 2nd European Conference on Model Driven Architecture, LNCS, 256-267.

[6] Janssen J.W., 2004. Evaluation of Current Tool Support for the Model Driven Architecture, MS Thesis, Uniersity of Twente, Enschede, Netherlands.

[7] Michel Hoste, Jorge Pinna Puissant, Tom mens,2008. Challenges In Model_ Driven Software Evolution, Technical report of BENEVOL Workshop, Technische University Eindhoven, December, 08-30.

[8] Mikko Konito, 2005. Architectural manifesto:choosing MDA tools, Three categories for evaluation, Model Driven Architecture(MDA) Information Center, IBM.

[9] Moke Riley, 2006. A Special Guide-MDA and UML Tools: CASE 2.0 – or the Developer's Dream, Dr.Dobb's Journal.

[10] Mukerji J. and Miller J. 2003. MDA Guide V10.1(omg/03-06-01), Technical report, OMG.

[11] Peter Wittmann , "Comparison of MDA tools",www.wittmannclan.com

[12] Philip Liew, Kostas Kontogiannis, Tack Tong, 2004. A Framework for Business Model Driven Development, Proceedingsof the International Workshop on Software Technology and Engineering Practice.

[13] Rational Product Support for MDA, Model Driven Architecture(MDA) Information Center, IBM.

[14] Tariq N.A., Akhter N. 2005. Comparison of Model Driven Architecture (MDA) Based Tools, MS Thesis, Royal Institute of Technology, Stockholm, Sweden.

[15] Computer Aided Software Engineering Tools Community,**http://case-tools.org**

[16] modelbased.net, www. modelbased. Net

[17] Objects by Design Forum, Jelsoft Enterprises Limited. http://forums.objectsbydesign.com

[18] UML Forum, http://www.uml-forum.com

[19] OMG's website. "Committed Companies and Their Products". 2006. http://www.omg.org/mda/committed-products.htm.

[20] Object Management Group (OMG), 2007. http://www.omg.org.

[21] Carlos Cetina, Estefania Serval, Javier Munoz and Vicente Pelechano. Tool Support for Model-Driven Dvelopment of Pervasive Systems, 4th International Workshop on Model-Based Methodologies for Pervasive and Embedded Software, 2007.