

A Proposed Framework for Comparing Tools to Reduce the Cost of Information Systems Development

¹MOHAMMAD HUSSEIN ALSAIDOMAR, ²Dr. Khaled Khalifah Allanqawi

Higher Institute of Energy

The Public Authority For Applied Education and Training, KUWAIT

DOI: <https://doi.org/10.5281/zenodo.14199359>

Published Date: 21-November-2024

Abstract: The rapid technical development in the world of software engineering and the rapid and continuous change in user requirements have created the need for software developers and systems analysts to identify and compare the advantages and disadvantages of tools for building software applications, especially modern tools that reduce the cost of developing information systems, and Model Driven Architecture tools are among the most important tools used to reduce the cost of developing information systems. This research is a step forward in identifying these tools that give an idea of the Model Driven Architecture tools that software developers and systems analysts can utilize to reduce the cost of that matter. Several criteria have been proposed to compare IS cost reduction tools, where six tools have been identified by presenting the advantages and disadvantages of each of them according to the proposed criteria. Knowing the most important advantages and disadvantages of these tools and comparing them will help system developers choose the tool that provides them with the required assistance and is compatible with the platform on which they work with the least cost, time, and effort.

Keywords: Model Driven Architecture (MDA) - Model transformation tools – Reducing.

1. INTRODUCTION

Software development is an expensive and slow process when compared to hardware development. It is very difficult and requires a lot of work as developers face many issues, with the emergence of any new technology, the effort required by the development team to take advantage of the new technology increases. Another issue is the changing customer requirements, which is one of the most important issues facing the team. Model Driven Architecture (MDA) tools help solve these issues and focus on software development from building the system model to writing the program in a programming language. Model-driven architecture tools are characterized by their ability to reduce the cost of developing information systems; hence we can call these tools as "**information system development cost reduction tools**".

This research will shed light on some of these tools in terms of their advantages and disadvantages, in addition to proposing a framework through which these tools can be compared through several criteria that help software developers choose the most appropriate tool that fulfills their development requirements with the least possible effort and time.

2. RESEARCH PROBLEM:

The development of information systems requires a high cost due to the rapid change in system requirements and the rapid progress in the world of information technology. With the existence of multiple tools that help reduce the cost of system development, it is difficult for the team to choose the most appropriate and useful tool for the system under development.

3. RESEARCH OBJECTIVE

The research aims to identify some Model Driven Architecture tools and mention their main advantages and disadvantages, in addition to proposing a framework for comparing these tools through several criteria developed by the researcher based on his experience, knowledge and study of the Model Driven Architecture technique and the tools that support this technique. The research also aims to highlight the strengths of each tool and the environment in which it operates.

4. IMPORTANCE OF THE RESEARCH:

The importance of the research lies in helping software developers to choose the tool that provides them with the required assistance compatible with the work platform on which they work with the least effort, time and cost, and showing the weaknesses of each tool, especially open source tools, will guide researchers and developers of this type of tools to work to avoid these points and improve the specifications of the tools that are developed to meet the requirements of users and keep pace with the rapid technological development and change with the least effort and cost.

5. RESEARCH METHODOLOGY

The research was prepared according to the descriptive and analytical method, where the concept of Model Driven Architecture technology and its tools will be presented, then six MDA tools will be studied, and a framework containing several criteria will be proposed through which MDA tools will be compared.

6. PREVIOUS STUDIES

There are many studies that have compared MDA tools, and the studies differed from each other either by the criteria or the tools that are compared.

Wang [1] proposed several criteria on which transformation tools can be evaluated, represented by the following points: Self-containment of auxiliary information - Ability to control the size of transformations - Simplicity - Bidirectional transformation - Can the transformations of this tool be used by other tools - Can the tool be enriched with some additional terms by the user - Use of common technical terms - Support for synthesis and reuse - Support for complex transformations such as M-N relationship transformation - Is the tool able to provide the user with a complete example - Power to execute the transformation -

Tool features (does the tool have strong support).

Pietraszek [2] proposed several criteria: manufacturer - model-to-model transformation - model-to-code transformation - whether the tool is freely available or a commercial version - UML support - the stage at which the tool can be used, then it is compared the following tools:

Optimal J - ArcStyler - Eclipse Modeling Framework - AndroMDA - Integranova Model Execution System - Enterprise Architect - Objecteering

Cernickins [3] also proposed several criteria for comparing MDA tools: Suitability for the phases of the system development life cycle - Functional capabilities (modeling - implementation - testing - documentation - project management) - Reliability - Reusability - Efficiency - Maintenance - Portability of the tool for different environments.

Palacios-Gonzalez et al [4] compared AndroMDA - ArcStyler - Borland Together - Eclipse Modelling by the following criteria: License type - Version currently available - Date of development of this version - Number of versions developed between 2007 and 2008 - URL of the tool - Modeling language used - Work platform required to make the tool - Work platform required to make the application - Programming languages into which code can be generated.

This study differs from previous studies by proposing some additional criteria for comparing tools, including Database generation - Reliability - Support for distributed analysis and design - Support for Operation - Support for error messages - Integration with the Eclipse environment - Synchronization - Support for Agile modeling - Trial version support for all functions.

7. MODEL DRIVEN ARCHITECTURE (MDA):

Model Driven Architecture is an initiative developed (introduced) by the Object Management Group (OMG) and through this technology, a model is built that is independent of any platform and can be converted to a model that depends on one or more specific platforms and from there to a programming code in one of the programming languages.

The Object Management Group (OMG) defines Model Driven Architecture as a method for organizing and managing the structure of organizations. This approach is supported by various tools and services that help define models and facilitate the automatic transformation of these models into different types.

Model Driven Architecture has gained increasing importance due to the advancements in information technology that these transformation tools rely on. This growing significance has led many prominent companies in the information technology sector, such as IBM, Oracle, Unisys, and IONA, to adopt this technology for developing their systems.

Model Driven Architecture relies on three basic parts:

- 1- Platform Independent Model.
- 2- Platform Specific Model.
- 3- Model transformation.

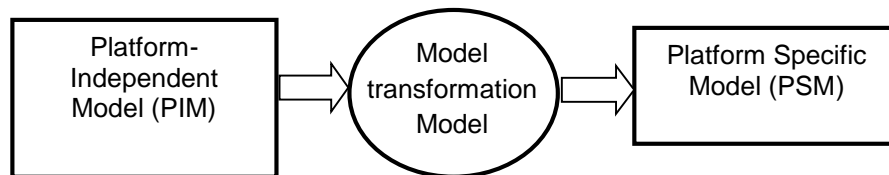


Figure 1: The main components of the MDA technique.

7-1 Basic principles of MDA technique:

The basics of MDA technology are based on three complementary ideas [6]:

- 1) Direct re-representation (modeling): In this way, attention is shifted during the development of information systems from focusing on the implementation technology (the platform on which the system will operate) to focusing on ideas and concepts related to the domain under development.
- 2) Automatic transformation: In this way, the ideas and concepts expressed during direct re-representation (modeling) are automatically converted into the required code that has specific technological specifications. In this way, the gap between the design and implementation stages has been bridged.
- 3) Open standards: Standards are considered one of the most effective reinforcements for progress in the technical field.

7-2 Benefits of using MDA technology:

The most important benefits of using MDA technology are [7]:

- Reducing the cost.
- Reducing the time required for the development process.
- Improving the quality of the information system being developed.
- Increasing the return on investment.
- The possibility of benefiting from any new technology in developing the system.

In addition to the previous benefits, we find that the productivity of the work team has improved during the development of the information system, as the use of this technology allows the work team to develop a larger number of information systems within a specific period of time. The reason for this is that the PSM is obtained automatically from the PIM, as well as the programming code is obtained automatically from the PSM or from the PIM, and thus the time required to build a system model for a specific technology and the programming code (when using traditional methods in developing systems) will be shortened and benefited from by focusing more on building a PIM with excellent accuracy and specifications.

The amount of cost reduction is not precise and is governed by many factors, and this is what the research will try to focus on and clarify.

8. MODEL-DRIVEN ARCHITECTURE TOOLS:

Many companies have introduced tools that support model-driven architecture. The coming years will witness growth in the market for model-driven architecture tools as a result of their adoption in developing information systems by large and well-known companies. The most important of these tools are:

Astah UML – Star UML – Argo UML - Software Ideas Modeler - Visual Paradigm – Enterprise Architecture – UMT – MTL – ATL – GMT – BOTL – OptmialJ - ArcStyler - Eclipse Modeling Framework – AndromDA - Integranova Model Execution System – Objecteering - Visual Studio 2012 – CodeFluent - Adaptive Framework - Gen it Architect - iUML and iCCG –...

Six tools were selected from the previous tools to study them and know their advantages and disadvantages with the aim of deducing the criteria by which model-driven architecture tools can be compared. The tools that were selected are: Astah UML - StarUML - ArgoUML - Software Ideas Modeler - Visual Paradigm UML - Enterprise Architect.

The following paragraphs will identify the previous six tools and the advantages and disadvantages that will be presented for each tool are the result of the researcher's work on these tools after installing them on the computer.

8-1 Astah UML Tool [8]

Astah is a new tool in the field of software design for small and large companies, in addition to the advantage of this tool with the feature of dynamic modification according to new work requirements, the tool also supports the possibility of cooperation between the work team by creating an idea diagram that allows the individual to send new ideas immediately to the work team and helps in the effective collaboration of more than one individual within the same group, which makes it one of the tools that support the use of light modeling technology (Agile Method). Light modeling is a set of principles and practices that can be applied during the system modeling process to accomplish the work effectively and quickly. This feature was not available in business development tools ten years ago, which made the work of software developers need time and effort to understand the work environment correctly, and the possibility of immediate modification to the project was not available. Astah Professional 6.8.0 version 37, which was released on 16/01/4201, will be dealt with. This version is distinguished from previous versions by the ability to fully support UML through the ability to generate UML diagrams directly from the idea diagram. Where all nodes in the idea diagram can be dragged and dropped into any UML diagram and thus will be automatically converted to (Classes, Use-Cases, Actors, States, and Activities).

8-1-1 Key Features:

- New UML diagrams are included in this release.
- Ease of design through the ability to build a mind map and then drag and drop it into UML diagrams, as it provides ready-made models and templates of the mind map.
- Includes flowcharts, data flow diagrams, relationship diagrams, entity ERD, requirements diagrams, and requirements tables.
- Export diagrams in different formats (BMP, PNG, JPG, GIF, TIFF, SVG, WMF, and (EMF.
- Export effective documentation and documentation with multiple HTML-RTF extensions)).
- Easy to use and considered a lightweight tool. Work can be shared by distributing work to several people and communicating with each other.
- Supports reverse engineering in Java only.
- Supports generating code in (C#, Java, and C++).
- Ability to convert class diagram to entity relationship diagram.

8-1-2 Disadvantages:

- Does not support specifying access permissions to the project for all those who communicate with them via the network.
- Cannot view the generated code within the tool.
- Does not support the ability to automatically restore the project in case of any error.
- Cannot import/export from XML.
- We cannot benefit from the user guide if there is no Internet.

8-2 Tool [9] StarUML:

The purpose of this tool is to build project models that can be integrated with other UML tools such as Rational Rose, and it also helps in evaluating UML diagrams by specifying the typical evaluation rules within the tool. This research will be worked on Star UML the Open Source UML/MDA platform version 5.0.2.1570.

8-2-1 Main features:

- Help can be obtained directly through the tool itself (Help).
- Supports all UML 1.4 and UML 2.0 diagrams.
- Supports reverse engineering of languages (C++, C#, Java).
- Supports inheritance as it appears in the generated code.
- Generate programming code in (C#, Java, and C++).
- Can import/export from XML.
- Export documentation with several extensions (.txt, .ppt, .doc)) and export diagrams as an image.
- Import from Rational Rose.
- Import from: Java2 enterprise 1.4, Java2 standard 1.3, .NET Base class-library, Microsoft foundation classes 6.0.
- Ease of use and helps in creating good typical UML diagrams, as there are 38 rules for evaluating the created UML models from which we can choose what we want.
- The ability to document the evaluation results in an Excel file.
- It contains options for controlling the work environment controlling the code generated into the languages you deal with, and controlling reverse engineering and the languages you deal with.
- An open-source tool, with the ability to add new options to the tool menus and define new functions by the user, thus the tool becomes suitable for the environment in which the user works [10].

8-2-2 Disadvantages:

- The generated code cannot be viewed within the tool.
- It does not support the ability to automatically restore the project in case of any error.
- It does not include flowcharts, data flow diagrams, relationship diagrams, entity ERD, requirements diagrams, and requirements tables.
- It does not support the ability to convert the class diagram to an entity relationship diagram.
- It is not considered a lightweight tool, meaning that it is not possible to share work and distribute work to several people and link them to each other.
- It does not deal with databases.

8-3 ArgoUML Tool [11]:

It was created by Jason Robbins and the rest of the team at the University of California. The first version of it was released in 1998 and the last version in 2001. In this research, we will deal with version: 0.32.1 ArgoUML version.

8-3-1 Main advantages:

- Supports UML diagrams.
- Import and export UML 1.3, 1.4 (UML V1.0) XMI, XMI V1.1, (XMI V1.2) models to and from other tools.
- Ability to import code as it continuously searches for the needs of object-oriented software designers to meet them and provide them with innovative features that increase productivity.

- Export diagrams in different formats (PNG, EPS, GIF, SVG, PS.)
- Supports widely open standards such as XMI, SVG, OCL, and others.
- Supports generating code for the following programming languages (SQL, C#, C++, Java, PHP4, PHP5)
- Supports reverse engineering from languages (C#, Java, C++, IDL).
- Ease of use.
- Open source and free.

8-3-2 Disadvantages:

- The generated code cannot be viewed within the tool.
- It does not support the ability to automatically restore the project in case of any error.
- It does not include flowcharts, data flow diagrams, relationship diagrams, entity ERD, requirements diagrams, and requirements tables.
- It does not support the ability to convert the class diagram to an entity relationship diagram.
- It is not considered a lightweight tool, meaning that it is not possible to share work and distribute work to several people and link them to each other.
- It does not deal with databases.
- It does not support project documentation.

8-4 Software Ideas Modeler Tool [12]

It is a lightweight, easy-to-learn tool that helps designers describe and design software and processes using flowcharts, UML 2.4, PMN 2.0, SysML 1.3, ERD and other diagrams. This tool is intended to serve managers and IT professionals. This research was conducted on version (Software Ideas Modeler 6.86.5175.) version 2 for the year 2009.

8-4-1 Main features:

- Supports all UML 2.4 diagrams, BPMN2.0 diagrams, SysML 1.3 diagrams, flowcharts, data flow diagrams, entity-relationship diagrams (ERD), user interfaces, work hierarchy analysis, concurrency diagrams, matrix diagrams (CRUD, RACI, RASCI, etc.), JSD diagrams and other diagrams.
- Export diagrams in different image formats (BMP, PNG, JPG, GIF, TIFF, SVG, WMF, (EMF) and in PDF format for all project models or parts of them.
- Generate documentation in different formats (PDF, RTF, HTML, ODT, and TXT).
- Generate code in (C#, VB.NET, Java, C++, Action Script, Java Script, PHP, Ruby, Python, SQL DDL, VB6, and XSD).
- Reverse engineering and .NET suite and database and programming in (C#, VB.NET, Java, PHP).
- Automatic project recovery in case of any error.
- Commercial tool but there are free trial versions of it.
- It is considered one of the light and easy-to-use tools.
- The diagrams that can be tested to convert are ERD and class diagrams.
- We can convert the class diagram to an entity relationship diagram where it appears in this diagram (after the transformation process) relations and PK and Fk keys automatically.

8-4-2 Disadvantages:

- The code that will be generated cannot be viewed within the tool.
- It does not support the ability to automatically restore the project in case of any error.

- It does not deal with databases.
- It does not integrate with other tools, import/export from XMI only within the same tool.
- We cannot benefit from the user guide if the Internet is not available.

8-5 Visual paradigm UML Enterprise Edition Tool [13]:

Produced by Apache Software, it has a long history in the field of developing modeling tools. This version is the eighth of the company's release, as the program belongs to what is known as commercial computer programs. This tool is not intended for personal use, but rather for use by analysts, designers, programmers, developers, maintenance personnel, and project managers. It is a comprehensive analysis and design tool. It covers all stages of system development, starting with defining requirements, analysis, model design, testing, and maintenance of the application, with the possibility of full transformation. This research will be done on the Enterprise Edition 8.0 (sp1-2) Visual Paradigm UML released in 2010.

8-5-1 Main features:

- Includes all UML diagrams.
- Integration with many other tools such as Rational Rose and Visio where models can be imported from them.
- Import and export models of XMI and XML type.
- Import and export diagrams as Excel files.
- Export diagrams as an image.
- This tool supports generating code for many programming languages (Action Script, C, C# (.NET 1.1 and .NET 2.0), C++, Delphi, Java, PHP, Python, Visual Basic, Visual Basic, NET).
- This tool supports database generation for many database systems (MS SQL, MySQL, HSQL, H2, FRONTBASE, DB2, Oracle, Sybase ASE, Sybase SQL Anywhere, PostgreSQL, Cloudscape/Dreby, Ingres, OpenEdge, Firebird, Cache, SQLite, Openbase.)
- This tool performs three forms of transformation:
 - Generate Code From Database.
 - Generate Code and Database from Class Diagrams. □ Generate Code and Database from ERD Diagram.
- Flexibility in documentation and reporting by generating detailed reports on diagrams in three formats (html.pdf,.doc,.) These reports include a table of contents, a table of partial paragraphs, detailed data about each component of the diagram, and a data dictionary.
- Speed in loading the form and speed in performance.
- This tool supports a single user developing models or multiple overlapping users where the work is distributed equally among them, i.e. it supports the Agile Method.
- The ability to control multiple previous versions and versions.
- Managing access control and user security, as it allows creating a password for each diagram independently.
- A commercial tool, but there are free trial versions of it for an unlimited period.
- Supports synchronization between the Class diagram and the ERD diagram by converting it to an ORM model and vice versa.
- Supports synchronization between the database and the ERD diagram.
- SQL instructions can be generated for the Class diagram by connecting to the database system and converting it to an ORM model.
- Strongly supports error messages.

8-5-2 Disadvantages:

- We cannot benefit from the user guide if the Internet is not available.
- When connecting to a specific database management system, its driver file must be downloaded from the company's website, as each database system has its file.
- It is difficult to change the default settings of the tool.
- The code that will be generated within the tool cannot be viewed.

8-6 Enterprise Architect Tool [14]:

This tool was introduced by Spark Systems, an Australian company with a history of development and innovation in the modeling tools market. The company is also a contributing member of the Object Management Group (OMG) and provides standards responsible for defining and improving UML and its associated specifications. This tool is based on the latest UML versions (UML 2.1) and this tool can support all stages of the system development life cycle and provides full traceability from the initial design stage to deployment, maintenance, and testing. Enterprise Architect 10.0.1007 version 10 for the year 2013 will be dealt with.

8-6-1 Main features:

- Supports several types of diagrams such as UML, MaidMapping, DFD, BPMN, and other diagrams.
- Supports the ability to import a file from Rational Software Architect (RSA) Rational Software Modeler (RSM) in addition to XML files exported by ArcGIS.
- Supports the ability to import a package from XMI in several formats.
- Supports the ability to preview the generated code, modify it and export it to an XMI file.
- Supports documentation in several formats (PDF, RTF, WORD, HTML, and Excel).
- Supports the ability to open any file in multiple formats (C, C++, C#, HTML, RTF) and if it contains an error, it identifies the location of the error.
- Supports Agile Method.
- Supports the ability to develop a model through reverse engineering, i.e. converting a model from (PSM) to a (PIM) model, using (MDA) transformations.
- Supports generating code for many programming languages (C#, C++, DDI, EJB Entity, EJB Session, Java, JUnit, NUnit, PHP, VBet, WSDL, XSD).
- Supports generating databases for many database systems (MS SQL, MySQL, HSQL, H2, FRONTBASE, DB2, Oracle, SybaseASE, Sybase SQL Anywhere, Ingres, OpenEdge Openbase, Firebird, Cache, PostgreSQL, Cloudscape/Derby, SQLite)
- Ability to display the database, which reflects to the engineer the extent of the ability to openly link databases, and generate DDL diagrams to create database structures.
- Ability to connect to database warehouses using Oracle Server - MS SQL - My SQL.
- Creates a database from an ERD schema, to generate a database model for different database formats.
- Manages access control and security of groups and users.
- Ease of modifying schemas.

8-6-2 Disadvantages:

- The commercial version cannot be downloaded as a free trial.

9. COMPARISON

Some criteria can be identified based on which the tools for Model Driven Architecture technology can be compared. The following table shows a comparison between the six tools that were identified through the criteria that were proposed.

Table No. (1) Comparison between MDA tools according to criteria determined by the researcher

Tool Name Comparison Criterion	Software Ideas Modeler	Astah Professional	StarUML	Argo UML	Visual Paradigm for UML	Enterprise Architect
Product	Dusan Rodina	Change Vision, Inc.	Plastic Software	Tigris.org	Visual Paradigm Int'l Ltd.	Sparx Systems
Working Platform	Windows (.NET), Linux (Mono)	Multi-platform	Windows	Cross-platform (Java)	Cross-platform (Java)	Linux & Mac installation)
Memory Size	18 MB	203 MB	MB150	unknown	unknown	186MB
Date of first release	2009-08-06	2006	2005-11-01	1998-04	2002-06-20	2000
Open source	no	no	yes	yes	no	no
Reliability	yes	no	no	no	yes	no
Date of last release	2014-01-19	2014-1-16	2007-07-24	2011-12-15	2013-01-21	2013-1-31
Trial period duration	30 days	50 days	unlimited	unlimited	unlimited	30 days
Documentation generation	yes	yes	yes	∟	yes	yes
Database generation	no	no	no	no	yes	yes
Support for distributed analysis and design	yes	yes	no	no	yes	yes
Special support for Operation	no	yes	yes	no	yes	yes
Integration with Eclipse	unknown		C#	unknown	Eclipse, NetBeans and IntelliJ	Eclipse, Visual Studio, TcSE
Support for system development life cycle stages	no	no	no	no	yes	yes
Synchronization	no	no	no	no	yes	yes
License Type	Commercial	Commercial-Free-Educational-For-Team	Experimental	Experimental	Commercial - Free - Educational Trial	Commercial
Agile Support	yes	yes	no	no	yes	yes
Trial Support for All Functions	yes	no	yes	no	yes	yes
Schemes from which Code can be Generated	Class diagram, ERD	Class diagram, ERD, Mind Map	Class diagram	Class diagram	Class diagram, ERD	Class diagram, ERD
Generated Languages	Java, C++, C#	Several programming languages	Java, C#, C++	Several programming languages	Several programming languages	Several programming languages
Reverse Engineering Languages	Java C++, C#	C#, VB.NET, Java, PHP	Java, C#, C++	Java	Java, C# (binary), C++, PHP	Several programming languages

10. CONCLUSIONS

- a) After presenting the tools and comparing them through the proposed framework (comparison criteria), we conclude the following:
- b) Both Visual Paradigm and Enterprise Architect were distinguished by their great capabilities compared to the rest of the tools, and thus these two tools are considered tools that can be used in developing many types of information systems due to the advantages of these two tools.
- c) Ease of work and simplicity of open source tools, which makes them good tools for teaching students to analyze small projects.
- d) Support for the Agile Method by some tools makes it helps in distributing work and increasing the interaction process between members of a single work team.

11. RECOMMENDATIONS

The researcher recommends familiarizing oneself with the various transformation tools that utilize the Model Driven Architecture (MDA) technique. It is essential to conduct a comparative study of these tools to understand their advantages and disadvantages, and to identify the most suitable tool for different types of information systems.

Highlighting the weaknesses of each tool, particularly open-source options, presents an important opportunity for researchers and developers. By addressing these weaknesses, they can create tools that reduce the costs associated with developing information systems. This effort can also lead to enhancements in features and the addition of new programming languages, ensuring that these tools continue to meet the needs of software developers and keep pace with rapid technological advancements and changes.

REFERENCES

- [1] Wang, W., 2005 - Evaluation of UML Model Transformation Tools, Technical report, Business Informatics Group, Vienna University of Technology, Nr 1-2(16).
- [2] Pietraszek K., 2012 - Comparative analysis of MDA tools, STUDIA INFORMATICA, Systems and information technology,
- [3] Cernickins A., 2010 - Clarifying a vision on certification of MDA tools, SCIENTIFIC PAPERS, UNIVERSITY OF LATVIA, 2010. Vol. 757
- [4] Palacios-Gonzalez E., Fernandez-Fernandez H., Garcia-Diaz V., Pelayo GBustelo B., Cueva Lovelle J., and Oscar Sanjuan Martinez O., 2008 - General purpose MDE tools, International Journal of Interactive Multimedia and Artificial Intelligence, Vol. 1, N° 1, ISSN 1989-1660
- [5] Brown A., 2004 - an introduction to model driven architecture part1: MDA and today's systems, Developer Work, IBM Corporation.
- [6] Booch G., Brown A., Lyengar S., Rumbaugh J., and Selic B., 2004 - An MDA manifesto, The MDA journal.
- [7] OMG - The Architecture of Choice for a Changing World, executive overview, Accessed: 11-06-2014 http://www.omg.org/mda/executive_overview.htm
- [8] <http://astah.net/edition/professional>
- [9] [http://staruml.sourceforge.net/docs/user-guide\(en\)/toc.html](http://staruml.sourceforge.net/docs/user-guide(en)/toc.html)
- [10] http://staruml.sourceforge.net/docs/StarUML_5.0_Developer_Guide.pdf
- [11] <http://argouml.tigris.org>
- [12] <http://softwareideas.net>
- [13] <http://www.visual-paradigm.com/installers/vpsuite5.0/vp-uml/vpuml-help.jar>
- [14] <http://www.sparxsystems.com/products/ea/index.html>