

## Model Driven Architecture

Name: Anish Mehta

Year: 3

Lecturer: Dr. Wolfgang Emmerich

Supervisor: Dr. Graham Roberts

For many years architects have been designing buildings by looking at other architects' designs and work. Architectural styles have been present for many millenniums and have developed throughout these years. Architects extract the most elegant features of other people's design and implement it in theirs. Architects have had many challenging tasks, and still do, as they are demanded to design buildings for different areas of the world. Say for example a group of architects have been asked to design the same building in two different countries. The tectonics are different if you wish to construct a building in an area where there is a likely chance of an earthquake occurring (e.g. California, USA) than you would if you are constructing that same building in an area where earthquakes do not occur (e.g. Spain). Recent research from many universities suggests that elastomeric materials could be used in the foundations of buildings. This rubbery material can absorb a certain percentage of the vibrations from the earthquake and could stop the buildings from being demolished. As the details of how the building will be constructed in these two different countries are different, the main design details are the same. These details can be captured so that they can be conveyed to the stakeholders in a clear and concise way.

In similar ways IT architects perform the same task as they design an IT solution for the business requirements. There are so many different technologies that can be used to develop a system and different platforms on which the same system can run. Therefore it is necessary to model the business requirements using a technology independent way. This data can then be modeled on a specific platform from which you can implement the system. This is the concept of **Model Driven Architecture**. It allows you to record the requirements from the customer in a clear and concise way to develop a system using a platform independent model. Everyday new technologies are evolving and organizations have to keep up-to-date with them. Requirements from customers are changing and functionalities have to be added to the software already created, which may require new technologies. Therefore to enable the integration of the old system with the new system and any other system created in the future, model driven architecture has to be used. Model Driven Architecture (MDA) includes the iterations of deploying, combining and managing data as well as applications.

A model in general is a miniature representation of the system and is basically something that shows concepts, but the term 'model' in MDA means a "system design<sup>1</sup>". The design

---

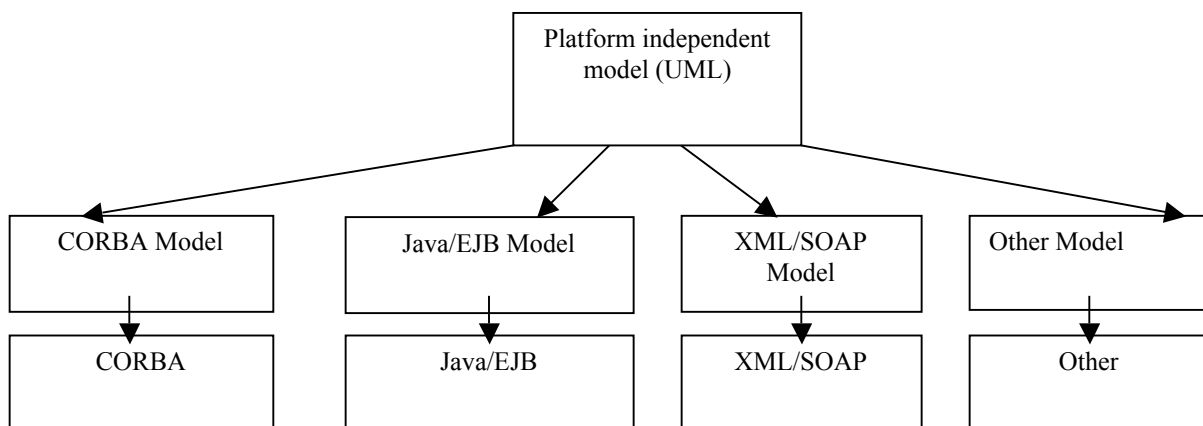
<sup>1</sup>["MDA and System Design"](http://ftp.omg.org/pub/docs/omg/00-11-05.pdf)  
by Makoto Oya, Hitachi, Ltd".  
<http://ftp.omg.org/pub/docs/omg/00-11-05.pdf>

from the MDA includes information that enables you to develop the actual system. The Model Driven Architecture extracts implementation details from the requirements analyzed, stated by the customer and is displayed in an open, platform independent way. This platform independent model can be obtained by using the following technologies<sup>2</sup>:

- (i) UML (Unified Modeling Language<sup>3</sup>) – This allows you to specify, visualize, develop and document software. E.g. Class diagrams, Use Cases, Activity diagrams, Collaboration diagrams and Sequence diagrams.
- (ii) Meta-Object Facility (MOF) – This lays down clearly the model required to design or implement the software required and also stores the PIM. The Meta-Object Facility achieves this by standardizing the meta-model.
- (iii) Common Warehouse Meta-model (CWM) – This allows easy exchange of information (metadata) between data warehousing tools and repositories (of metadata).

Normally a full Model Driven Architecture is made up of a platform-independent UML model and one or more platform-specific models. In addition, the platform independent model contains information about how the UML base model is implemented on the middleware platform.

The platform independent model created in UML from the customer requirements can be mapped to multiple platform specific models, which are then mapped to the platform on which the system is implemented. The PIM can be mapped to the platform specific model (PSM) by defining a set of common mapping rules and MDA tools can be used to achieve this. MDA allows most of the system to be generated using these tools and so the software developers do not have to hand code the model. The diagram below shows an example of the elements of Model Driven Architecture<sup>4</sup>.



<sup>2</sup> Definitions of the technologies have been based on: <http://www.rational.com/uml/index.jsp> and <http://www.omg.org/gettingstarted/>

<sup>4</sup> Diagram based on: [http://www.omg.org/mda/mda\\_files/Soley-MDA/mda-schema5.gif](http://www.omg.org/mda/mda_files/Soley-MDA/mda-schema5.gif)

A platform independent model created can be used to develop the software on many platforms like Java, CORBA, XMI/XML and Web Base platforms as it can be seen from the diagram above. Therefore the program can be created for mainframes, desktops, servers and the Internet.

The Pervasive services that the MDA supports are the directory, transactions, events and security services. The Pervasive Services model is created from the PIM and is implemented on the desired platform. Pervasive services are services provided by the MDA and these are called from the application, and these are only visible to the applications.

The best way to comprehend MDA would be by discussing a simple example<sup>5</sup>, like creating a server on the Java platform. Model Driven Architecture process would consist of the following procedures:

- (i) Create a platform independent model (PIM) using the Unified Modeling Language. This may include class, object, sequence and activity diagrams.
- (ii) The tool that is used for the MDA executes to create a Pervasive Services model and Domain Facilities model. Code is generated that calls the services of that particular platform.
- (iii) By using an automated mapping tool, you may select the desired platform (middleware), which in this case is Java, and obtain the platform specific model from the platform independent model. The platform independent model is also stored in the Model Object Facility.
- (iv) The MDA tool then creates all the required Java files for the server to run on the Java platform. If the application server on which the Java platform runs supports many programming languages the programming language has to be selected in the MDA tool. In this case Java has to be selected. Also the tool will look at the information from the UML model created in step (i) and create necessary files to set the application server to work as required.

Model Driven Architecture development initially concentrates “on the functionality and behavior of a distributed application or system<sup>6</sup>.” MDA extracts the implementation information from the customer requirements and therefore it is unnecessary to model the functionality and behavior of a system or application again when a new technology arises, unless the customer requirements are modified. MDA allows you to model the functionality and behavior only once. In addition it allows the team of developers to solve

---

<sup>5</sup> Example is based on the example given in the presentation “Developing in OMG’s Model Driven Architecture (MDA)” by Jon Siegel by Jon Siegel, Ph.D., and the OMG Staff Strategy Group (<ftp://ftp.omg.org/pub/docs/omg/01-12-01.pdf>).

<sup>6</sup> [http://www.omg.org/mda/faq\\_mda.htm](http://www.omg.org/mda/faq_mda.htm)

any problems that arise at an early stage that are due to the incorrect customer requirements gathered.

Model driven Architecture takes into account many of the problems that are faced by the IT industry and these are the main reasons why MDA is used by many organizations<sup>7</sup>:

- (i) **Portability** – This also allows the developers to re-use the components. Therefore the cost of the software development decreases and the complexity decreases. This is very beneficial to the organization as it could allocate its budget to other things for its project. Portability in MDA is mostly achieved by utilizing UML to create the PIM.
- (ii) Allows the software modeled to be developed on different platforms without losing the functionality of the software. This is called **Cross-Platform Interoperability**. Therefore the architecture is flexible if the infrastructure is changed. This is achieved in MDA by the platform independent model created in UML and this is why UML is a major technology in Model Driven Architecture.
- (iii) **Domain specific** – Model Driven Architecture can be molded to a particular industry, for example the Finance Industry. This is again due to the platform independent model, which is technology independent and captures the functionality and behavior of the requirements.
- (iv) **Productivity** – Again, it allows the system to be developed quickly. MDA provides a common form of communication between the whole IT team. In addition it allows them to interact more efficiently as things are much more clear and concise. A higher form of communication is presented to each member and therefore MDA endeavors to eliminate any ambiguities. “An architectural style expresses the language and design culture that helps stakeholders at all levels to communicate at a higher, more effective level.”<sup>8</sup> Therefore this could eliminate any unnecessary time wasted between the IT developers discussing any ambiguities that arise due to the language being used.

There’s no one single platform that may be ideal for a customer and so you must be able to design a model that will let you develop your software on any platform. This is main reason why MDA should be used when developing your software.

Before MDA was presented to the industry a lot of the large IT organizations had problems, as the models that were created were just platform dependent and the common diction used by team of developers was poor, causing huge failures. In addition this wasted a lot of their time as well. The technologies are always rapidly changing and new technologies are always emerging. New middleware may develop and the business may wish to use the latest technologies to be at the cutting-edge of the competition. For example the Carphone Warehouse Group wish to stop using Uniface for their stores and

---

<sup>7</sup> Developing in MDA – Object Management Group” – page 12:

<http://www.omg.org/mda/presentations.htm>

<sup>8</sup> Wiley Convergent Architecture – Building Model Driven J2EE systems with UML

instead just wish to use Java. Also currently they are in the process of outsourcing all their development to companies in India and this task is easy for them, as they have implemented MDA in their software development process.

Business requirements are always changing and so you require this change to be implemented into your system. To handle for this change you must use the Model Driven Architecture in your system development process. In the beginning stages of the prototype development, MDA makes it easy to detect if the flaws are due to requirements not being captured correctly in the platform independent model or the code generated from the platform specific model. The main reason why it is recommended to use MDA is because it allows you to produce programs for any middleware platform from the platform independent model. The MDA is used by many different industries including finance, e-commerce, telecomm, health-care, transportation, space and manufacturing industries.

## **Bibliography**

- Book: Wiley Convergent Architecture – Building Model Driven J2EE systems with UML
- Presentation “Developing in MDA – Object Management Group” – page 12  
Website: <http://www.omg.org/mda/presentations.htm>
- "MDA and System Design" by Makoto Oya, Hitachi, Ltd".  
<ftp://ftp.omg.org/pub/docs/omg/00-11-05.pdf>
- <http://www.rational.com/uml/index.jsp>
- "Developing in OMG's Model Driven Architecture (MDA)"  
by Jon Siegel, Ph.D., and the OMG Staff Strategy Group  
<ftp://ftp.omg.org/pub/docs/omg/01-12-01.pdf>
- <http://www.omg.org/gettingstarted/>
- [http://www.cwmforum.org/Model-Driven Architecture.pdf](http://www.cwmforum.org/Model-Driven%20Architecture.pdf):
- [http://www.omg.org/mda/faq\\_mda.htm](http://www.omg.org/mda/faq_mda.htm)
- <http://www.sciencenet.org.uk/slup/CuttingEdge/Jan00/earthquake.html>