Making the Most of an Enterprise Architecture Modeling Tool

Aurelijus Morkevicius
Product Manager for the UPDM plugin for MagicDraw and Cameo Enterprise Architecture
aurelijus.morkevicius@nomagic.com
Contents

Introduction .................................................................................................................................................. 3
A Process: Making the Most of an EA Modeling Tool ............................................................................... 4
  Set Up Requirements ............................................................................................................................... 5
  Pick the Right Tool .................................................................................................................................... 5
  Make the Tool Follow Your Rules ............................................................................................................. 9
  Visualize .................................................................................................................................................. 11
  Share ....................................................................................................................................................... 14
  Analyze ..................................................................................................................................................... 15
  Measure Gain .......................................................................................................................................... 17
Conclusion .................................................................................................................................................. 18
Introduction

Modern enterprises are based on complex business systems that are supported by even more complicated IT systems integrating multiple technologies and standards. Modeling helps to capture and communicate the multilayered architecture of the enterprise and cause it to evolve for better enterprise performance.

The ability to capture, integrate, and maintain integrated enterprise architecture (EA) models strongly depends on the modeling tool features that make enterprise architects’ daily work more productive. In this paper, we will be analyzing the modeling tool features which are critical for integrated enterprise architecture (EA) success.

First of all, a good EA modeling tool should support multiple standards, such as UPDM, SysML, BPMN, UML, and others. It also should enable integration of these models into a single repository while linking elements from other models conforming to different standards. It is critical that the tool is compliant with standards, as this enables interoperability in a tool chain and data migration between tools. Second, the EA modeling tool should provide good collaboration-oriented features, which include a shared remote repository, simultaneous model editing, version management including model comparison and merge, easy publishing, and appropriate review capabilities. Third, the EA modeling tool should support the Model Driven Architecture paradigm, which not only includes multiple abstraction levels and model to model and model to code transformations, but also the ability to validate the model for completeness and correctness, and easily correct any issues identified in the model. Fourth, the EA modeling tool should support linking various model elements and provide easy traceability, which enables impact analysis when evolving the EA. Last but not least, the tool should be customizable – it should support multiple user roles with different sets of features displayed/hidden and customizable report templates, and provide an Open API-enabling enhanced productivity through plugins and integrations with a tool chain, and other customization capabilities that can boost modeling productivity.

We will examine these features, going step by step through a process of making the most of enterprise architecture modeling tools, and we will shortly demonstrate how some of these features are implemented in MagicDraw, one of the leading integrated EA modeling solutions.
A Process: Making the Most of an EA Modeling Tool

Based on our experience in enterprise systems and software modeling, we have developed the cycle of how to make the most of a modeling tool. We have discovered that some activities bring more value for the EA modeler than others. For example, model creation using diagrams is a capability provided by all modeling tools. In contrast to this, model visualizations in tabular or time-based charts are valuable rarities. In the context of EA, we are usually talking about multiple users with different backgrounds – such as the business analyst, CIO, system engineer, etc. The model is a medium between them. However, some users are not aware of BPMN or UML, and others are not familiar with other modeling notations. Multiple visualizations of a model is one of the ways for people in organizations to communicate successfully.

We have determined that there are seven steps to using a modeling tool efficiently:

1. Set up requirements
2. Pick the right tool
3. Make the tool follow your rules
4. Visualize
5. Share
6. Analyze
7. Measure gain

To make the most of your modeling tool, set up requirements and write questions. This will help in the selection of the tool which meets the project requirements. When you select the tool, use it to model, visualize, share, and analyze model data. Finally, measure gain. Repeat the cycle again if you feel the choice was wrong. If that is not an option, try to get the most value from the tool you selected during the first iteration.
Set Up Requirements

Before selecting the tool, set up requirements and constraints. Think of what standards you are planning to use to implement EA. Will it be TOGAF, DoDAF, or a custom standard? Decide what modeling language to use. Think of the interoperability of the models. Are you planning to exchange models with other companies or between different tools? There are many issues to be evaluated before making the decision. Here are the main points you must consider before choosing the tool:

1. What architecture framework am I planning to use?
2. Do I need models to be interoperable?
3. How many users will need to work in parallel with the same project?
4. What collaborative standards do I plan to use; how do they fit?
5. Do I need professional services; training, consulting, support?
6. What is my budget?

As soon as your requirements and constraints are determined and you have an answer for each of the six questions, make a list of tools meeting your requirements. Then raise another set of questions of what exactly you need from a modeling tool to pick the right one.

Pick the Right Tool

We have raised 10 common questions to answer when choosing a modeling tool:

1. Is it a standards-based integrated architecture repository?
2. How many enterprise architecture frameworks does it support?
3. How many ways to visualize model data does it provide?
4. Is time aspect management supported?
5. Is any modeling automation provided?
6. Does it support traceability?
7. Is there an integrated solution for parallel development and configuration management?
8. Is it adjustable and customizable?
9. Does it provide a model validator?
10. Is it secure?

Is it a standards-based integrated architecture repository?

A standard is a set of the best practices. Compliance with the standards saves resources, reduces maintenance, enables reuse, etc. However, even if standards-compliance is not a primary requirement, it is treated as a quality mark.
A market survey by Scott W. Ambler [1] unveiled that the majority of unsuccessful EA programs were based on their own architecture frameworks. See Figure 1.

Multiple standards support, however, is not as efficient without an integrated environment (Figure 2). An integrated environment or, as we call it, an integrated architecture repository (IAR) allows you to have it all in one single repository. The IAR:

- Helps to maintain information integrity
- Allows establishing and maintaining traceability between different abstract layers of modeling
- Gives you a broad choice of modeling languages for any kind of domain
- Ensures interoperability
- Brings people from different domains together
- Saves time
- Saves resources

Try to ask yourself if any of the points listed above meet your requirements. If not, maybe the requirements need to be updated.

**TIP!** Some standards such as UPDM, SysML, and SoaML are extensions of UML. Non-UML implementations of these standards are not standards-compliant.

**How many enterprise architecture frameworks does it support?**

Support of enterprise architecture frameworks gives you a structure to organize artifacts in a standardized manner. DoDAF, MODAF, TOGAF, and Zachman architecture frameworks are the most popular standardized solutions in the market (Figure 3). A modeling tool should provide a user environment corresponding to its preferred architecture framework, but not the modeling language, such as UPDM, providing a common background for MODAF, DODAF, and NAF frameworks.

Besides predefined solutions, EA tools should offer powerful extensibility solutions such as the domain-specific language (DSL) engine in MagicDraw, which allows you to define your own framework:

- Custom diagrams
- Custom elements, and their appearance and connectivity
- Custom role-based modeling environment
- Custom reports

At this point we can add one more question: Does the tool allow you to craft your own EAF?
How many ways to visualize the model does it provide?

Flexible support for EAFs provides many different model visualizations capturing different aspects of the architecture for users from various domains. Besides diagrams, model data can be represented in the form of a table, matrix (simplified tabular view), report, time-based chart, and many others. For instance, our observable tool, MagicDraw, provides seven ways to represent model data: two types of diagrams, behavioral and structural, in addition to tables, matrices, timeline charts, relation maps, and reports.

To acquire value from visualizations, all of them should reflect the model, and be editable, customizable, printable, and exportable to an image, HTML or other formats. There should be predefined visualizations provided to quickly build EAF artifacts; for instance, an OV-3 table is required by DoDAF and MODAF, and a PV-2 Gantt chart is required by DoDAF and many others.

Is time aspect management supported?

Enterprise Architecture is an iterative process by its nature, as business and IT are constantly changing. Tracking of capability realization and deployment, changing resource configurations, and project statuses at a particular time are the time-dependent activities encountered in almost every enterprise. By not having the ability to visualize it in a time-based chart, time aspect management is barely possible. This results in ineffective capabilities and project management, resource configurations versioning, and overall transition between iterations of the enterprise modeling life cycles.

Is there any modeling automation provided?

Plain modeling usually takes a lot of time, especially when instantiating an architecture or establishing traces between elements. Modeling automation allows users to save time in achieving specific, usually continuously performed tasks such as defining interactions between two performers, instantiating architecture, and managing the time aspect of the architecture. Automated guides provide an intuitive way to perform daily tasks, so before choosing the tool for EA modeling, make sure there is modeling automation functionality implemented in it, and it best fits your needs.

TIP! UPDM is not a framework. It is a modeling language for building MODAF, DODAF and NAF architectures. So do not get confused by the tools supporting UPDM, for instance, if you expect to get DoDAF.
Does it support traceability?

Traceability between different layers of architecture allows for specification and analysis of traces. The tool should provide predefined traceability rules and allow creation of your own transitive traces, perform impact analysis according to defined traces, and allow easy navigation between traced elements in the model.

Is there an integrated solution for parallel development and configuration management?

Do not even think of choosing a tool without an integrated parallel development environment when modeling EA. This is a must-have capability.

For example, MagicDraw provides Teamwork Server, software which allows more than one developer to work with the same model. If you are connected to the Internet, you may work with the model that is stored in a central repository. If you are not, you may save the teamwork project locally and work without a constant connection with the Teamwork Server. Changed project files can be saved to the Teamwork Server later. A project manager may provide read-only or full access to the repository for interested parties. Supported branching ensures separated areas of development, tests, and trials for the same project.

Is it adjustable and tailorable?

Making the tool look like you want, the ability to build your own custom diagrams and edit the existing ones, the ability to record macros and build your own model validity rules, and many more. This is what adjustability and tailorability are all about.

Usually, if the tool provides many different environments for the users from different domains and supports a number of modeling languages and domain specific languages (DSLs), it is a tool that can easily be adjusted to your needs. Additionally simple things such as look and feel themes or so called tool skins may give you a pleasure and enjoyment while working with the tool.

Does it provide a model validator?

How does the user know if the model is correct and complete? At the least, correction of the model is a duty of the tool. If the tool is standards compliant, it should track and let you know if your model is correct, and that it meets the specifications of the standard. Critical violations should be displayed instantly, and in comparison minor issues should be tracked and displayed on demand in order to avoid interrupting your work process.
Is a model complete? This is another question which in comparison to correctness is much harder to answer. However, the tool should at least provide rules checking – for example, if business processes have allocated resources or capabilities are satisfied by capability configurations.

Is it Secure?

If you are from the government sector, this section is probably one of the most important. However, if you are from the industrial sector you should not skip it either! Information security has held confidentiality, integrity, and availability (known as the CIA triad) to be the core principles. These principles also set requirements for the modeling tool. Questions that should be asked at this point:

- Are SSL connections to the server repository supported?
- May different people in the project have different access levels to the projects stored in the server repository?
- May users have different permissions on the data that is located in the server repository?
- Are administrative rights supported?
- Is native user support in the server repository integrated with the Subversion, ClearCase, and/or LDAP?

Make the Tool Follow Your Rules

Let’s assume you have chosen the tool. From this point, we will observe the rest of the steps for a successful EA modeling by using MagicDraw.

To make sure your tool follows your rules:
- Set up project and environment options.
- Set up a GUI.
- Add your own executable model validity rules.
- Add your own traceability rules.
- Record macros.

Set up project and environment options
Before beginning and during any time of modeling, you may control the project’s look and behavior with a set of project options common to all project users.

For instance, if you choose a DoDAF project in MagicDraw you can define various compliance modes, such as BPMN compliance, which allows using BPMN for some of the DoDAF views modeling.

Another set of options is mostly related to user interface and common to all projects under the one machine.
MagicDraw has a set of environment options such as look and feel theme, enabled plugins, and others.

**Set up a GUI**

Set up a user interface to best fit your needs. Choose an appearance theme, windows layout, and role-based modeling environment. Control the experience level of the GUI.

MagicDraw provides a set of look and feels, such as Windows 7, Eclipse, etc. It also allows choosing the modeling environment based on the role and experience level of the user. For instance, a DoDAF architect can select DoDAF 1.5 architect or DoDAF 2.0 architect modeling environments. This hides tools that are not expected to be used by a DoDAF architect and emphasizes tools such as diagrams, action buttons, elements, and menu items specific to the DoDAF framework. Some of the features are controlled by expert and non-expert experience levels.

**Add your own executable model validity rules**

Are you not happy with the behavior of the predefined rule sets? That is fine; you can create and execute your own validity rules. OCL and JAVA are supported languages in MagicDraw for executable rules development. You can select one of five severity levels. Each rule can be marked as an activity and act instantly if it is violated. Additionally, automatic solvers can be added to perform particular actions.

Our experience shows that validity rules are not necessarily used for validity checks. You may, for instance, set up low level severity rules to highlight elements on the diagrams according to a specific criterion such as version, type, etc.

**Add your own traceability rules**

If you feel you are missing transitive traces between elements, you are welcome to set up your own traceability rules. Based on the rules, you will get the ability to track, visualize, and navigate through traces. There are multiple ways provided to set up a new rule in MagicDraw which may be based on meta-chain, OCL, or JAVA.

**Record macros**

Macro recording is another helpful feature of the modeling tool. A recorded set of actions may be performed in one click, thus saving the modeler’s time.

The MagicDraw macro engine is actually a scripting engine allowing coding, recording, and execution of scripts in various scripting languages such as BeanShell, JavaScript, JRuby, Groovy, and Jython. You may access the MagicDraw Open API and even use macros for learning the Open API by analyzing recorded scripts.
Visualize

Use as much visualization as possible:

- Gantt chart for planning.
- Capability time chart for gap analysis.
- Tables for textual data specification.
- Matrices and relation maps for traceability analysis and specification.
- Spreadsheet reports for analysis of measurements.

Gantt chart for planning

Plan your projects using a time-based chart, thus improving the clarity, efficiency, and flexibility of project portfolio management.

![Figure 4 PV-2 Project Timelines](image)

MagicDraw provides a full-featured Gantt chart for EA modelers reflecting the UPDM model. Using the Gantt chart, you can more easily model a timeline perspective on your projects due to the following enhancements:

- Both grouping and elementary projects can be displayed as bars on the chart. A bar illustrates the duration of a project from start to finish. You can expand, shrink, and move bars directly on the chart.
- There are several types of dependencies that can be defined between projects (for example, “finish-to-end”, “end-to-end”).
- Milestones can be created for defining and tracking project status at any given time. Milestones are displayed as pie charts along a project bar. You can also create your own model-based templates for defining project statuses.
- The chart can be used to show the current schedule status using percent-complete shadings and a vertical TODAY line.
- The chart view can be easily manipulated using the shortcut menu features, as well as some popular shortcut keys (for
example, holding down CTRL and scrolling the mouse wheel down to zoom out).

**Capability time chart for gap analysis**

Plan capabilities and perform a capability gap analysis. Use a time-based chart to visualize capability increments and deployments to organizations, and to efficiently discover capability gaps and duplications.

**Figure 5 CV-3 Capability Phasing**

MagicDraw provides a time-based chart for capabilities management reflecting the UPDM model. Using the time chart, you can more easily model a timeline perspective on your capabilities because of the following enhancements:

- Both grouping and elementary resources can be displayed as bars on the chart. A bar illustrates the increment of a capability by a particular resource or a set of resources. You can move, merge, stretch, or shrink bars directly on the chart.
- The chart can be used to display the planned achievement of capability at different points in time or during specific periods of time. Gaps or duplication can be identified more efficiently in the capability provision.
- The Capability Provision Creation wizard makes creation of capability increments easy.
- Capability increments without an end date defined will be displayed on the chart with a pointer on the right side of the bar.
- A vertical TODAY line can be added.
- The chart view can be easily manipulated using the shortcut menu features, as well as some popular shortcut keys (for example, holding down CTRL and scrolling the mouse wheel down to zoom out).
- The chart look can be easily managed by changing colors, bar display style, labels, visible period tiers, and other style properties.

**Tables for textual data specification**

Tabular representation of model data allows you to easily manage data of the same kind. For instance, information exchange between performers can be easily managed and is recommended to be managed in a tabular format by DoDAF and MODAF AFs. Moreover,
tabular format can usually be exported to Excel, CSV, HTML, and other popular formats.

Figure 6 CV-5 Capability to Organizational Development Mapping
Our observable tool, MagicDraw, provides a set of predefined tables for DoDAF and MODAF and the ability to create your own custom table for displaying the desired data. Tables allow quick spreadsheet report generation, and the ability to export to CSV and HTML formats.

Matrices and relation maps for traceability analysis and specification
A matrix is a simpler form of a table displaying relationships between two types of elements. Matrices allow the user to quickly analyze data, create or delete relationships, and track traces based on the predefined or custom traceability rules.

MagicDraw suggests using a number of predefined matrices for DoDAF and MODAF and provides an ability to create your own matrices based on the simple relationships, element properties, or predefined and custom traceability rules.

Figure 7 SV-3 Systems-Systems Matrix and a fragment of radial relation map
Relation maps or so-called contextual maps, mind maps, etc. are the best tools for traceability analysis, discovery of suspect links, and an overview of the architecture.

Relation map functionality provided by MagicDraw allows you to perform analyses based on the simple relationships, element properties, or predefined and custom traceability rules.

Spreadsheet reports for analysis of measurements
Use an Excel spreadsheet or Open Document spreadsheet report to analyze data using various statistical and mathematical functions.
that are out of scope for a modeling tool. This is especially important if analyzing measurements of systems, business processes, and capabilities.

MagicDraw allows the user to generate spreadsheet reports directly from predefined DoDAF and MODAF tables or from the whole model. A versatile WYSIWYG reports engine allows you to create your own report or edit the predefined ones according to your needs. Besides the spreadsheet report, it is definitely important for EA architects to generate Word documents, PowerPoint presentations, PDFs and many other less popular formats such as open document format.

MagicDraw’s versatile report engine is built on top of the Velocity Engine (Open Source Templates engine). The type of template files that the Report Wizard supports are normal text, RTF, HTML, Spreadsheet template (need to be saved as HTML format), XML template (DocBook or FO), and Microsoft Word and Excel 2007 files.

Share

Establish a collaborative environment:
- Share the project with different architecture development teams.
- Use branching and version control.
- Publish the project online.

Share the project with different architecture development teams
Develop a single project in parallel, by multiple EA teams. Access the project from the network wherever you are. Review, comment, and make decisions based on the up-to-date model data. By adopting a collaborative environment, you will no longer need to send files using e-mail, merge, and differentiate files constantly.

MagicDraw provides an integrated teamwork server where every user may instantly obtain the newest version of the model; everybody is allowed to work in parallel on their own part and all designs are stored in a single place so there are no more files that
are scattered around in the network. Changes can be introduced in an orderly fashion and without conflicts.

Use version control
Version control brings the ability to track changes, promotes accountability, and makes it easier to find the right person to solve problems in the materials maintained. Usually modeling tools provide several options for versioning. For instance, the MagicDraw Teamwork Server supports project versioning and branching. However, if you already use Subversion or ClearCase, MagicDraw provides an option to integrate.

Publish the project online
Online publishing is another important feature you will probably need.

Figure 9 MagicDraw project published online
Using the MagicDraw reports engine, you may publish your model in a Java Doc–like report with clickable navigation and image maps for diagrams and elements from the model. The model will be accessible from anywhere to review and even comment.

Analyze

Measure the performance of the model:
- Analyze parametrics.
- Execute behavioral model.

Analyze parametrics
Considering EA, we generally believe that the quality attributes (such as security and integrity) of an enterprise system are primarily achieved through EA (same as software architecture). In other words, most of the design decisions within the EA are strongly influenced by the need to achieve quality attributes. In software
engineering, the aim of analyzing the architecture is to predict the quality of a system before it has been built and not to establish precise estimates but the principal effects of the architecture [2]. There is a common misconception that quantitative analysis is “too detailed” to be performed at the architecture level. Performance engineering practitioners argue that next to functional aspects, non-functional aspects of systems should also be taken into account at all stages of the design of a system.

Quantitative analysis can serve several purposes. In the first place it is often used to optimize, for example, processes or systems. Similarly, it can be used to obtain measures to support the impact of change analysis. A third application of quantitative analysis is capacity planning, e.g., how many people should fulfill a certain role to finish a process in time.

EA models can be quantified in several ways. Measures of interest may include:
- Performance measures, i.e., response time, utilization, workload.
- Reliability measures such as availability and dependability.
- Cost measures.

MagicDraw provides a solution for quantitative architecture analysis based on SysML parametric model simulation. It is a separate product called the Cameo Simulation toolkit. Based on the experience of implementing and evaluating architectures, we can make the following propositions:
- A model-driven approach enables modeling all kind of calculations.
- The model fragments, such as constraint blocks, can be easily reused in many contexts, which makes it very promising for constructing reusable libraries of EA model elements and evaluating various combinations for the best performance solution.

**Execute behavioral model**

UPDM in combination with SysML, BPMN, SoaML, and UML provides a standards-based enterprise modeling powerhouse for various domains. Besides parametric analysis, architecture behavioral models can be executed using various techniques such as fUML, SCXML, JSR223 action language, XPDL, BPEL, and many others.
The Cameo Simulation toolkit is a fully standards-based solution provided for activity and state machine execution. It supports an extendable model execution framework based on OMG fUML and W3C SCXML standards. The simulation toolkit extends MagicDraw to validate system behavior by executing, animating, and debugging UPDM Statemachine (OV-6b, SV-10b), UML Statemachine, UPDM Activity (OV-5, SV-4), UML Activity and SysML parametric models in the context of realistic mock-ups of the intended user interface.

**Measure Gain**

Let’s answer the final question: How do I know I made the most of the modeling tool? There are multiple articles with lists of reasons why EA programs fail, and the tool is one of them. The simplest way to evaluate the tool is a staff questionnaire. However, this may reveal only the quality in use measures of the software. In order to make sure you are making the most of it, it is not enough to read this paper. The guarantees for a successful tooling are professional services such as training, consultancies, effective use of a support system, and tool documentation. Also, please remember that even if the primary intention for implementing EA may not be cost reduction, it can in fact provide some stunning returns on investment and the right tooling is likely to be a key to it.
Conclusion

Today, enterprise modeling tools are experiencing two problems: one is how to select the tool and the other is how to successfully use it. It is becoming obvious that just having the tool does not necessarily mean it is being used in the most effective way; we have seen multiple instances when the tool has been used incorrectly. For this reason, we highly recommend the use of professional services such as training and consulting, and we also advise users to take advantage of the tool’s support structure and to carefully read the documentation and other material provided with the tool.

In this white paper we have introduced you to the steps we consider to be the most important in choosing today’s modern enterprise modeling tool. However, just reading will not help you to succeed. We hope reading the paper has provided food for thought and will serve you in the future in making the most of your enterprise modeling tool.

References

About the Author

**Aurelijus Morkevicius**

Product Manager, UPDM Plugin for MagicDraw and Cameo Enterprise Architecture

aurelijus.morkevicius@nomagic.com

Aurelijus has been with No Magic, Inc. since 2008. He has worked as a System Analyst for Enterprise Architecture solutions such as the UPDM Plugin for MagicDraw. Currently Aurelijus is the Product Manager for the UPDM plugin for MagicDraw and Cameo Enterprise Architecture. He also participates actively in various OMG activities. He is one of the leading architects for the current OMG UPDM standards development group.

Aurelijus is also actively involved in educational activities. He earned a master’s degree in computer science from Vilnius University in 2009 and currently is undertaking doctoral studies in Information Engineering. He also teaches an enterprise modeling course at Kaunas University of Technology.

Aurelijus is an OMG Certified UML and BPM Professional.

Contacts

No Magic Europe, UAB
Savanoriu ave. 363, Kaunas
LT - 49425, Lithuania
Phone:+370-37-324032,
Fax:+370-37-320670
E-mail: sales@nomagic.com

No Magic, Inc.
One Allen Center, 700 Central Expressway South,
Suite 110 Allen, TX 75013
Phone: +1-214-291-9100,
Fax: +1-214-291-9099
E-mail: sales@nomagic.com

No Magic Asia
719 KPN Tower, 22nd floor,
Rama IX Road, Bangkapi, Huaykwang,
Bangkok 10310, Thailand
Phone: +66 2717 0250
Fax: +66 2717 0251
E-mail: nomagicth@nomagicasia.com