MODEL-DRIVEN QUANTITATIVE PERFORMANCE ANALYSIS OF UPDM-BASED ENTERPRISE ARCHITECTURE

Aurelijus Morkevičius¹, Saulius Gudas¹,³, Darius Šilingas²

¹Kaunas University of Technology, Faculty of Informatics, Information Systems Department, Studentu 50-313a, LT-51368 Kaunas, Lithuania, aurelijus.morkevicius@stud.ktu.lt, gudas@soften.ktu.lt
²Vytartus Magnus University, Faculty of Informatics, Department of Applied Informatics, Vileikos 8-409, LT-44404 Kaunas, Lithuania, darius.silingas@gmail.com
³Vilnius University, Kaunas Faculty of Humanities, Maistas 8, LT-44280 Kaunas, Lithuania, gudas@vukhf.lt

Abstract. The idea of the enterprise architecture (EA) has been active since 1980-ies. However, enterprise architecture performance attributes analysis still lacks a clear approach and tools for implementing it in practice. The paper presents an approach for the model-driven performance evaluation of the EA models. The suggested approach is based on the Unified profile for MODAF and DoDAF (UPDM), System Modeling Language (SysML) parametric diagram, and a bottom-up performance evaluation algorithm. The support for this method has been implemented in MagicDaw modeling product line. A real world example is presented to validate the suitability of the approach.

Keywords: UPDM, SysML, Enterprise Architecture, Performance Analysis, Model-Driven Architecture.

1 Introduction

Enterprise Architecture (EA) has been a hot topic since 1980-ies [22]. However, it was not very widely applied in practices due to lack of modeling languages and tools suitable for EA [2]. The EA movement was reinforced with the successful adoption of the Unified Modeling Language (UML) [18] and the Model-Driven Architecture (MDA) [15]. There have been multiple attempts to apply Unified Modeling Language (UML) for Enterprise Architecture (EA) modeling [4], but many modelers found it too complicated and non-natural for solving their domain-specific problems [20]. In 2005, the Unified profile for MODAF and DoDAF (UPDM) initiative has been started in OMG, but the first version of UPDM was released only in 2009, four years later [19]. As soon as the UPDM has been officially released, US Department of Defense mandated UPDM as Information Technology Standard and Profile Registry (DISR) standard. As UPDM is a profile of UML, it has been easily adopted by the majority of UML tool vendors. The versatility of UML and its compatibility with its profiles allows integrating UPDM with the other Object Management Group (OMG) standards based on UML, such as System Modeling Language (SysML), Service Oriented Architecture Modeling Language (SoaML), etc [21]. This enables creating large and versatile EA models, but does not provide a toolkit for analyzing them and making decisions such as choosing between alternative EA solutions or identifying performance problems in an existing EA solution. For achieving these goals, EA modelers need tools for quantitative performance analysis of the EA models from various domains. While OMG is not proposing any methods or tools in addition to modeling language, it is necessary to adopt the methods existing in industry or invent new ones suitable for EA models based on UPDM. The goal of this paper is to present an approach of adopting existing quantitative performance analysis algorithm using SysML Parametric Diagram for evaluating performance values for EA models following UPDM standard, its implementation in MagicDaw modeling product line, and its application to experimental system.

The rest of this paper is structured as follows: in section 2, the related works are analyzed; in section 3, the proposed approach for quantitative analysis of EA models based on UPDM is presented; in section 4, experimental evaluation of the proposed approach a small real world EA model is described; in section 5, the achieved results, conclusions, and future work directions are indicated.

2 Quantitative Performance Analysis of Enterprise Architectures

Considering EA, we generally believe that quality attributes (such as security, and integrity) of an enterprise system are primarily achieved through EA (same as software architecture [13]). 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 [1].

There is a common misconception that quantitative analysis is „too detailed“ to be performed at the architecture level [3]. 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 [14].
Quantitative analysis can serve several purposes. In the first place it is often used to optimization of, for example, processes or systems. Similarly, it can be used to obtain measures to support 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 [14].

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.

The techniques and example presented in this paper focus on performance measures.

2.1 Related work

Not much can be discussed about the related works, because of the novelty of UPDM standard, which first version was released a half-year ago. SysML is relatively new standard also. The approach of using SysML parametric diagrams for enterprise architecture quantitative analysis has never been applied before and none closely relative works has been published.

Model driven approach for the evaluation of EA have been suggested by Pontus Johnson, Robert Lagerstrom, Per Narman and Marten Simonsson [11]. Suggested extended influence diagrams for EA quantitative evaluation, uses Bayesian networks. The proposed extended influence diagrams differ from the conventional ones in their ability to cope with definitional uncertainty, i.e. the uncertainty associated with the use of language and in their ability to represent multiple levels of abstraction.

Ulrik Franke, Pontus Johnson, Evelina Ericsson, Waldo Rocha Flores, and Kun Zhu propose [6] an improvement and formalization of EA dependency analysis by methods from Fault Tree Analysis (FTA). FTA is a combinatorial model of systems dependability, widely used for safety and reliability evaluations [5]. The method translates the failure behavior of a physical system into events connected by arcs. A visual model portrays the relationships in an accessible way, while a corresponding logical model enables quantitative evaluation. However, this approach phases two major gaps: gap of abstraction and gap of expressive power [6].

Maria-Eugenia Iacob and Henk Jonkers [10] proposes Quantitative analysis approach on Archimate based EA. The approach became as a predefined one for Archimate tools, but in general the evaluation process itself is not model driven. The approach [10] shall be adapted to UPDM based enterprise architecture in fully model driven evaluation manner using SysML parametric diagrams.

3 Model-driven quantitative performance analysis of UPDM-based Enterprise Architecture

Standards, techniques and formulas for the quantitative analysis of UPDM based architecture shall be discussed in this chapter.

3.1 SysML and UPDM compliance

The Unified Profile for MODAF and DoDAF (UPDM) defines a set of UML and optional SysML stereotypes and model elements and associations. A set of stereotypes extending UML, UML model elements and set of SoaML stereotypes defines the L0 compliance level of UPDM that is mandatory. A set of optional SysML stereotypes for UPDM is called L1 compliance level [19].

In the model level L1 compliance results in an UML element with UPDM and SysML stereotypes applied. For example Operational Node UPDM stereotype applies for a class. SysML Block stereotype applies for a class also. By applying these both stereotypes for a class, as the result we get a class that possesses meta-properties form both stereotypes applied and in conceptual meaning defines both the Operational Node and the Block. For the rest of the mappings between UPDM and SysML stereotypes [19], see Table 1.
Table 1. Mappings between UPDM and SysML stereotypes.

<table>
<thead>
<tr>
<th>SysML stereotype</th>
<th>UPDM stereotype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value Type</td>
<td>Climate, Entity Item, Environment, Light Condition, Location, Measurement Set</td>
</tr>
<tr>
<td>Item Flow</td>
<td>Commands, Controls, Data Exchange, Energy Exchange, Information Exchange, Materiel Exchange, Organizational Exchange, Resource Interaction</td>
</tr>
<tr>
<td>Requirement</td>
<td>Enterprise Goal</td>
</tr>
<tr>
<td>Flow Port</td>
<td>Node Port, Resource Port</td>
</tr>
</tbody>
</table>

3.2 Applying SysML Parametric Diagram for UPDM based Enterprise Architecture

In SysML, Parametric diagrams are used to create systems of equations that can constrain the properties of blocks [17]. Each block may consist of value properties. Parametric diagram connects these values to the constraint block’s parameters using binding connectors [12], see Figure 2. Incoming values are constrained within the constraint block and the result is provided to one or more outgoing parameters. The result may be used for further calculations within the same context [7].

In order to apply SysML parametric diagrams to UPDM, UPDM based enterprise architecture should be compliant with SysML (the word compliant means that UPDM element may have SysML stereotype applied, see “3.1 SysML and UPDM compliance” section above) [19]. In case the condition is satisfied, block stereotypes shall be applied to the majority of UPDM entities such as nodes, resources and etc.

From SysML point of view, UPDM elements that have block stereotypes applied may have value types assigned. Value types in turn may be bound to constraint blocks using the binding connectors. A binding connector is a connector which specifies that the properties at both ends of the connector have equal values [17].

![Figure 2. SysML constraint blocks meta-model](image-url)

3.3 Bottom up performance calculation

Quantitative performance analysis technique shall be used to evaluate the effectiveness of the proposed approach. Technique has been suggested by Maria-Eugenia Iacob and Henk Jonkers [9].

In summary, proposed [10] analysis approach consists of the following two phases: a top-down calculation and propagation of the workloads imposed by the top layer; this provides input for a bottom-up calculation that are about to perform on UPDM-based architecture and propagation of performance measures.

In this paper we focus on the bottom-up propagation of performance measures. The following recursive expressions apply [9]:

1. The utilization of any resource \( r \) is:
   \[
   U_r = \frac{1}{c_r} \sum_{i=1}^{d_r} \lambda_{k_i} T_{k_i}
   \]  
   where \( d_r \) is the number of internal behavior elements \( k_i \) to which the resource is assigned.

2. The processing time and response time of an internal behavior element a is computed using the following recursive formulas:
\[ T_a = S_a + \sum_{i=1}^{d-a} n_{k_a} R_k \]

\[ R_a = F(a, r_a) \]

where \( d-a \) denotes the in-degree of node \( a \), \( k \) is a parent of \( a \) and \( r_a \) is the resource assigned to \( a \) and \( F \)

is the response time expressed as a function of attributes of \( a \) and \( r_a \).

3. For example, if we assume that the node can be modeled as an M/M/1 queue [8], this function is:

\[ F(a, r_a) = \frac{T_a}{1 - U_{r_a}} \]  

From a given set values and the values calculated using top-down workload calculation, we calculate utilization \((U)\) of a resource, response time \((R)\) and the processing time \((T)\) of a service for each resource and service within the EA scenario.

3.4 The process of Enterprise Architecture scenario performance attributes evaluation

As all the techniques required for EA scenario attributes quantitative evaluation have been discussed separately, it requires a clear workflow definition of how to associate these techniques together to achieve the desired results.

![Figure 3. The process of Enterprise Architecture scenario performance attributes evaluation](image)

The process of applying the proposed approach consists of seven steps:

1. Model UPDM based EA scenario. Selected operational or systems view scenario is modeled using UPDM internal structure diagrams such as OV-2, SV-1 or SV-2.
2. Model Constraint Blocks. This step consists of modeling SysML constraint blocks, their parameters and mathematical equations. Constraint blocks are usually modeled using SysML block definition diagram.
3. Add SysML value properties for UPDM entities. At this step UPDM entities to be evaluated are supplemented with value properties.
4. Bind value properties to constraint blocks parameters. Value properties of the UPDM entities should be bound to parameters of the constraint blocks using SysML binding connectors.
5. Instantiate UPDM entities. Instance specifications are created for all the UPDM entities with the empty slots for every value property of the UPDM entity.
6. Add given values. Values for the slots of Instance specifications shall be assigned.
7. Solve parametries. Simulation of SysML parametric diagram shall be performed to calculate the result values.

Summarizing the process of the proposed approach step by step, UML activity diagram is provided in Figure 3 that covers the process of applying SysML parametric diagrams for UPDM based Enterprise Architecture scenario quantitative analysis.

4 Experimental Evaluation

Let us define a simple systems view (SV) fragment of UPDM based Enterprise Architecture from [9] in order to demonstrate proposed model-driven EA analysis approach.

The given EA fragment consists of human resource, systems and services divided into three used configurations called organization, application and infrastructure. These three capability configurations are the internal parts of the workload capability configuration.

Organization capability configuration consists of the post role administrator that initiates the scenario by searching damage reports using application’s search component. From the service oriented viewpoint the administrator stands for the service receiver and the search component stands for the service provider [16]. The search component requests the query results from the database system and the data base system requests data access from the database server.
SysML parametric diagram shall be used in the context of Workload capability Configuration, to visualize our sample fragment of EA. Usages of constraint blocks for resource utilization, response time and service provision time shall be added and connected to the resources’ value properties, see Figure 4.

To perform the bottom up performance calculation on the given sample, initial values for the calculation are needed. To calculate utilization (U) for a resource, workload (L) and capacity (C) values are needed. To calculate time (T) for service provision, service execution time (S), multiplicity (n) and response time (R) values are needed. The values shall be taken from the research of Maria-Eugenia Iacob and Henk Jonkers [9].

In order to assign the initial values and mark the values to be calculated, the provided parametric diagram should be instantiated, see Figure 5. That is the next step before the sample can be solved quantitatively. The values that are given are marked as given and those to be calculated are marked as targets.

As you can see in the figure above (Figure 5), the calculation has been successful. All target values have been calculated.

Unfortunately the proposed approach allows evaluating only the structural constructs of EA. Current version of SysML does not allow to add value properties to SysML Behaviors [17]. Behavioral especially the Activity diagrams are important to the UPDM models. SysML parametric models that can use the information captured in behavior diagrams would simplify UPDM SysML integration [23].

5 Conclusions and Future works

We have presented an approach how to apply quantitative model-driven performance analysis of EA models based on a new UPDM standard. SysML Parametric Diagram is used to model EA parameters, and
bottom-up calculation algorithm is applied for deriving performance values. The proposed approach has been implemented in MagicDraw and has been evaluated on a small illustrative fragment of real world EA model.

Based on the experience on implementing and evaluating the proposed approach, we can make the following conclusions:

- A model-driven approach enables modeling all kind of calculations;
- The models fragments, such as constraint blocks, can be easily reused many context, which makes it very promising for constructing reusable libraries of EA model elements and evaluating various combinations for the best performance solution;
- While the proposed approach looks promising for complex EA, it may be over complex and too expensive for a small fragment of EA.
- The proposed approach allows evaluating only the structural constructs of EA.

The proposed approach shall be used as a starting point for the more detailed future works on performing quantitative evaluation of EA models based on UPDM.

6 Acknowledgement

The authors would like to thank No Magic, Inc, especially the MagicDraw product team for comprehensive support.

References

[23] Zwemer, D. Using ParaMagic and SysML in Combination with UPDM. Atlanta: InterCAX LLC.