Enterprise knowledge based UML timing model generation process

Ilona Veitaitė
Department of Informatics
Vilnius University, Kaunas Faculty
Kaunas, Lithuania
e-mail: Ilona.Veitaite@knf.vu.lt

Audrius Lopata
Department of Informatics
Vilnius University, Kaunas Faculty
Kaunas, Lithuania
e-mail: Audrius.Lopata@knf.vu.lt

***Abstract*–The main scope is to present UML dynamic models generation from Enterprise model (EM) method with an example of UML Timing model. As enterprises develop and information systems (IS) become bigger it is needed to create precise and complex analyses for developing systems. The quality of analyses like these is crucial for the success of IS development process, because the later an error or mistake is found the more expensive is to solve it. IS designers daily face new challenges when they need to face and understand models made by other designers and analytics. This process creates additional problems and difficulties. Automatization of IS engineering process lets create superior and more qualified models with less errors. To achieve this goal there is analyzed ISO standards and UML models integration with knowledge-based Enterprise model, MOF architecture's role in IS engineering process, opportunities to upgrade its structure with new knowledge-based layer and its improved usage in UML dynamic models generation from Enterprise model process.**

Keywords–Enterprise Model; Knowledge-based; IS engineering; UML, ISO, MOF, CASE

I. INTRODUCTION

Business and IT alignment has kept as a meaningful management concern for over two decades because it remains a significant goal. Superior strategic alignment between business and IT strategy should guide to superior execution in comparison with reduced number of stages of strategic alignment. There are many debates about the business-IT alignment terminology. Some would controvert that alignment is a fragile goal that can be achieved only by a minor number of enterprises, where everyone prefers the help desk and the IT budget is enough to finance any proposed project. And some would accept that business and IT alignment insures suitable function of entire organization. Professionals offer that organizations need to reach strategic business and IT alignment to be competitive. Strategic business and IT alignment influence business performance and IT effectiveness [4, 18, 19].

In order to help induce communication and understanding, many enterprises are generating business-facing roles that have major responsibility for creating and sustaining relationships between business and IT business fields. The balance of IT and business vision, the capacity to explain business and IT challenges with same clarity should be ensured [1, 9].

The present situation and the relevance of the research was evaluated by analysing the scientific literature related with the IS engineering, IS life cycle phases, enterprise modelling, UML, ISO standards, MOF architecture, model driven development, and other areas. The method of UML dynamic models generation from Enterprise model was created to insure more effective and qualified generation process and lower number of mistakes in the final IS development stage.

II. KNOWLEDGE-BASED IS ENGINEERING

A knowledge-based IS engineering proposed methods and tools for system modelling and decision-making, which help to develop more specific and detailed subject area relative to the project [1, 5]. IS project participants (like developer, programmer and etc.) can use not only the knowledge of the project, which is stored in traditional CASE tool repository, but also the knowledge repository, where subject area knowledge tested by formal criteria is stored (Fig. 1) [6, 7, 10].

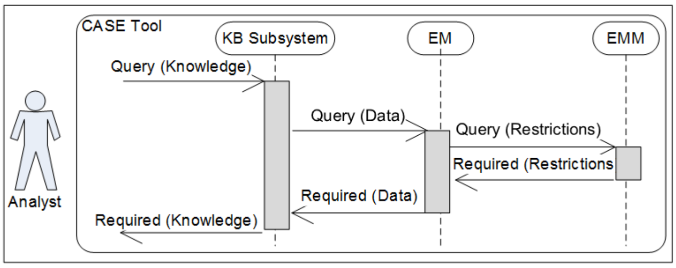

Fig. 1. *Knowledge-based subsystem connection to the Enterprise model and Enterprise meta–model inside CASE tool [10]*

UML is one of the most common software specification – standards. It is a universal IS modelling language applied to a number of methodologists and used in the most popular modelling tools, such as Enterprise Architect, System Architect, MagicDraw and etc. The method of UML models generation from Enterprise model implements a knowledge-based design phase in the IS development cycle [10,11,12,15,16, 20].

Knowledge-based subsystem as CASE tool component with Enterprise meta-model and Enterprise model inside can solve this question. Enterprise meta-model is a formal structure which ensures more qualified project development process and knowledge base data collection [21, 22].

A. *Strategic alignment model and Knowledge-based CASE Tool subsystem*

The strategic alignment model (SAM) proposed by Henderson and Venkatraman is one of the most cited strategic alignment models [8, 17, 18, 19]. SAM is composed of two main dimensions: strategic fit and functional integration. Strategic fit refers to the concordance between internal and external domains. Functional integration refers to two type of integration between business and IT domains. The first type is termed strategic integration and reflects the link between business strategy and IT strategy. The second type is termed operational integration and deals with the link between organizational infrastructure and process, and IT infrastructure and [8, 17]. SAM is a conceptual model that it has been used to understand strategic alignment from the perspective of four components. J.C. Henderson and N.Venkatraman described the interrelationship between business and IT strategies. Strategic Alignment model (SAM) was created in order to describe these relations. Model consists of four domains: Business Strategy Domain, Business Infrastructure Domain, IT Strategy Domain, IT Infrastructure Domain and relationships between them. Model is based on two parts: strategic integration and functional integration [8, 18]. Strategic integration recognizes that the IT strategy should be expressed in terms of an external domain (how the organization is positioned in the IT environment) and an internal domain (how the IT infrastructure should be configured and managed). There are two types of functional integration: strategic and operational. First one is the relation between the business strategy and IT strategy. Second one covers the internal domain and deals with the relation between business infrastructure and IT infrastructure. Functional integration considers how choices made in the IT domain impact those, which are made in the business domain and vice versa. Strategic alignment model describes four basic alignment perspectives: Strategy Execution, Technology Transformation, Competitive Potential, and Service Level [8].

First proposed business and IT strategy alignment model is conceptual. It does not provide a practical framework to implement this kind alignment, despite that, there are alignment mechanisms developed and used in organizations to achieve the business and IT synthesis, but these mechanisms mostly are oriented to business, not to IT [10].

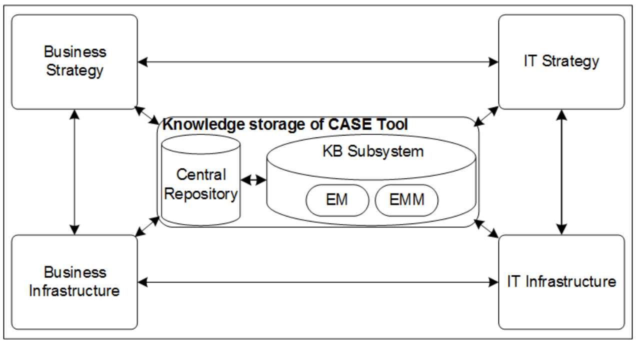

Fig. 2. *Business and IT alignment model and Knowledge storage of CASE tool [10]*

Knowledge-based CASE Tool subsystem in business and IT management can be used as the major source for certain frameworks [13,14]. These frameworks specify all relevant structures within the organization, including business, applications technology, data and their relationships to perform business. For business and IT alignment process some specific data is necessary and it is used in Knowledge-Based Subsystem. Business strategy domain provides business goals to knowledge base (Fig. 2). Business goals are described by the business managers and are received from business environment and IT goals are described by the IT managers. Business infrastructure domain delivers business rules, constraints, processes, functions and other related data and IT infrastructure domain delivers information about IT infrastructure, where describes current software, hardware. All this information is stored in knowledge base and can be used through enterprise model, where is validated according to the enterprise meta-model [13, 14].

III. ENTERPRISE MODELLING

Enterprise meta-model is formally defined enterprise model structure, which consists of a formalized enterprise model in line with the general principles of control theory. Enterprise model is the main source of the necessary knowledge of the particular problem domain for IS engineering and IS re-engineering processes [14].

Enterprise meta-model manages Enterprise model composition. Enterprise model stores knowledge that is necessary for IS development process only and will be used during all phases of IS development life cycle [14, 21].

There is given formalized Enterprise meta-model description, which is needed to define UML Timing model generation process algorithm. Enterprise model can be described as Malcev algebra based algebra system (Fig. 3) [21]:

$$M1 = \langle K, R \rangle \qquad (1)$$

where M1 – Enterprise model as algebra system; K – elements set of M1 system; K={K1, K2,…, K21}, where K1,....K21 EM meta-classes; R – set of relationships between elements, where R={r1, r2, r3}.

For each set of K element Kn composition is defined as: Kn=<{an1, an2,…,ank}, {mn1, mn2,…,mnl}>, where {an1, an2,…,ank} – attributes of Kn element, {mn1, mn2,…,mnl}– methods of Kn element.

Enterprise model M1 composition is as follows:

$$M1 = \langle \{K1, K2, ..., K21\}, \{r1, r2, r3\} \rangle \qquad (2)$$

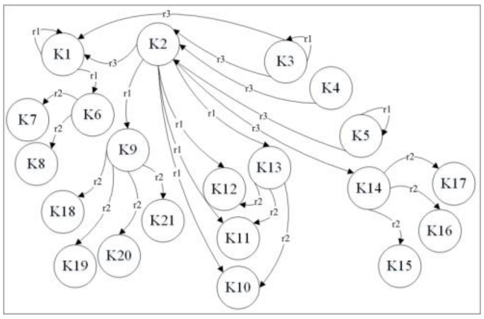

Fig. 3. *En Enterprise meta-model graphical schema based on Malcev algebra [21]*

where: K1 – meta-class Process, K2 – meta-class Function, K3 – meta-class Actor, K4 – meta-class Event, K5 – meta-class Goal, K6 – meta-class Material Flows, K7 – meta-class Input Material Flow, K8 – meta-class Output Material Flow, K9 – meta-class Information Flow, K10 – meta-class Interpretation, K11 – meta-class Data Processing and Solution Making, K12 – meta-class Realization, K13 – meta-class Information Activity, K14 – meta-class Business Rules, K15 – meta-class Interpretation Business Rules, K16 – meta-class Data Processing and Solution Making Business Rules, K17 – meta-class Realization Business Rules, K18 – meta-class Process Output, K19 – meta-class Information processing Input, K20 – meta-class Information processing Output, K21 – meta-class Process Input, r1 – Aggregation, r2 – Generalization, r3 – Association [22].

A. *MOF and UML relationship*

The Meta-Object Facility (MOF) is an Object Management Group (OMG) standard for model-driven engineering. MOF provides an open and platform-independent metadata management framework and associated set of metadata services to enable the development and interoperability of model and metadata driven systems. Examples of systems that use MOF include modeling and development tools, data warehouse systems, metadata repositories. MOF has contributed significantly to the core principles of the OMG Model Driven Architecture. Building on the modeling foundation established by UML, MOF introduced the concept of formal meta-models and Platform Independent Models of metadata as well as mappings from PIMs to specific platforms. MOF only provides a means to define the structure, or abstract syntax of a language or of data [20].

Enterprise modelling has become an integral part of information system development process. Lately, the organization's business modelling has become an important phase of modelling design processes. MOF architecture supplemented with enterprise meta-model assures appropriate information system development process [21].

The Meta-Object Facility (MOF) is an Object Management Group (OMG) standard is designed as a four-layered architecture. It provides a meta-meta model at the top layer, called the M3 layer. This M3-model is the language used by MOF to build meta-models, called M2-models. The most prominent example of a Layer 2 MOF model is the UML meta-model, the model that describes the UML itself. These M2-models describe elements of the M1-layer, and thus M1-models, for example, models written in UML. The last layer is the M0-layer or data layer. It is used to describe real-world objects. MOF is a closed meta-modelling architecture; it defines an M3-model, which conforms to it-self [21].

As it is described above M3 is meta-meta model, the base for a Meta-modelling Architecture, which defines the language to describe meta-models. And M2 meta-model is an instance of a meta-meta model, which defines the language to describe models.

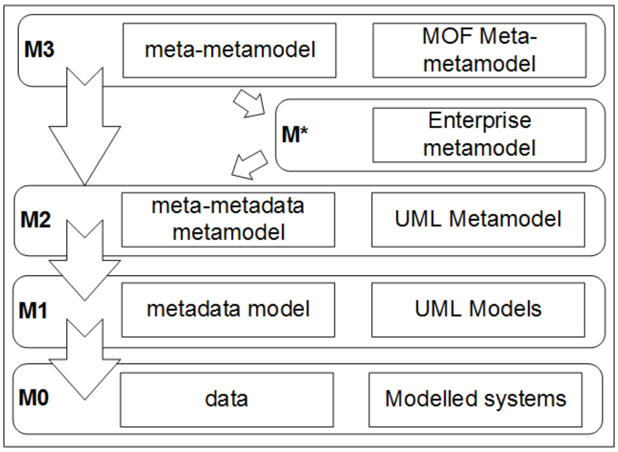

Fig. 4. *MOF architecture with additional EM layer [21]*

Among M3 and M2 layers one more layer is needed to assure more accurate usage of MOF architecture [21]. This additional layer consists of enterprise meta-model (Fig. 4). Enterprise model and enterprise meta–model makes information system needed UML project models generation process more efficient and eligible and ensure reduced number of mistakes in the final information system development stage [21].

B. *ISO standards in requirements specification stage*

ISO standards make a positive contribution to the world. They provide solutions and achieve benefits for almost all sectors of activity. ISO Standards are documented agreements containing technical specifications or other precise criteria to be used consistently as rules, guidelines, or definitions of characteristics, to ensure that materials, products, processes and services are fit for their purpose [2, 3, 22].

International standards in software and system engineering are an excellent indication on what is considered good practice by the international community of professionals that work in these areas [3, 22].

Subcommittee 7 (SC7) is responsible for IT services and software and systems engineering standardization [2]. SC7 standards are constantly updated by developing and improving on standards. One of the main scopes is to integrate IT and business system definition and provide the software and system engineering tools to implement enterprise information systems. SC7 standards collection consists of several blocks, where one of the most relevant is process implementation and assessment [2, 3].

IV. UML DYNAMIC MODELS GENERATION PROCESS FROM EM

UML model is a partial graphical view of a model of a system under design, implementation, or already in presence. UML model contains graphical elements – UML nodes connected with flows – that represent elements in the UML model of the designed system [9, 10].

A. UML Timing model generation example

One of the UML behaviour models is Timing model. Timing models are UML interaction models used to show interactions when a primary purpose of the model is to reason about time. Timing model focus on conditions changing within and among lifelines along a linear time axis. Timing models describe behaviour of both individual classifiers and interactions of classifiers, focusing attention on time of events causing changes in the modeled conditions of the lifelines [16].

There is given formalized UML Timing model description (Fig. 5). UML Timing model also can be described as Malcev algebra based algebra system [21]:

$$M3=<K,R> \quad (3)$$

where M2 – UML Timing model as algebra system; K – elements set of M2 system; K={K28, K29,…, K33}, where K28,....K32 UML Timing model meta-classes; R – set of relationships between elements, where R={r1, r2, r3}.

UML Timing model M2 composition is as follows:

$$M3=\{K28, K29,\dots, K33\},\{r1\}> \quad (4)$$

where: K28– meta-class Lifeline, K29 – meta-class State or Condition Timeline, K30 – meta-class Interval Constraint, K31 – meta-class Duration Constraint, K32 – meta-class Time Constraint, K33 – meta-class Destruction Occurrence, r1 – Aggregation [21].

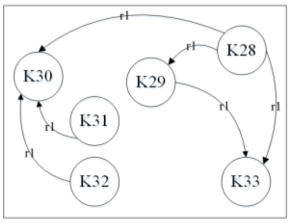

Fig. 5. *UML Timing model graphical schema based on Malcev algebra*

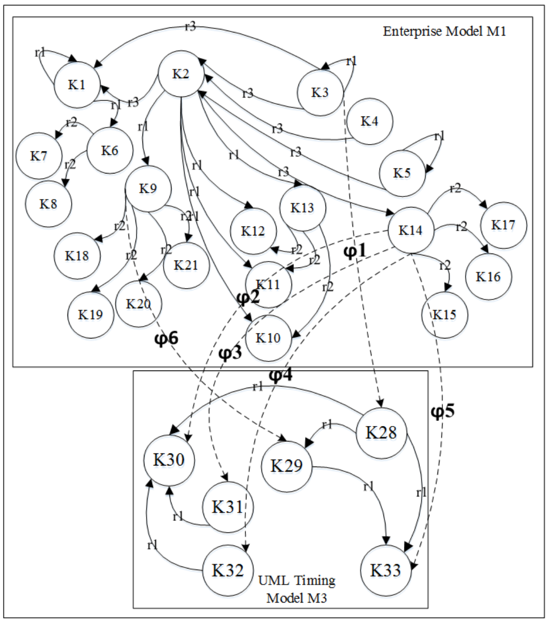

Fig. 6. *Graphical scheme of intersection between Enterprise Model and UML Timing Model elements based on Malcev algebra*

According to the figure (Fig. 6) it is clear that enterprise model elements: Actor, Function and Business rules can be generated as UML Timing model elements: Lifeline, Destruction Occurrence, Intervals Constrains, Durations Constraint, Time Constraint and State or Condition Timeline.

TABLE I. INTERSECTION BETWEEN ENTERPRISE MODEL AND UML TIMING MODEL ELEMENTS

Enterprise model set element	UML Timing model set element	Formal description
Actor (K3)	Lifeline (K28)	φ1: K3→K28
Business Rules (K14)	Interval Constraint (K30)	φ2: K14→K30
Business Rules (K14)	Duration Constraint (K31)	φ3: K14→K31
Business Rules (K14)	Time Constraint (K32)	φ4: K14→K32
Business Rules (K14)	Destruction Occurrence (K33)	φ5: K14→K33
Information Flow (K9)	State or Condition Timeline (K29)	φ6: K9→K29

Table (Table 1) presents intersection between Enterprise model and UML Timing model elements, where formal description of enterprise model elements generated to UML Timing model elements according to Malcev algebra can be found.

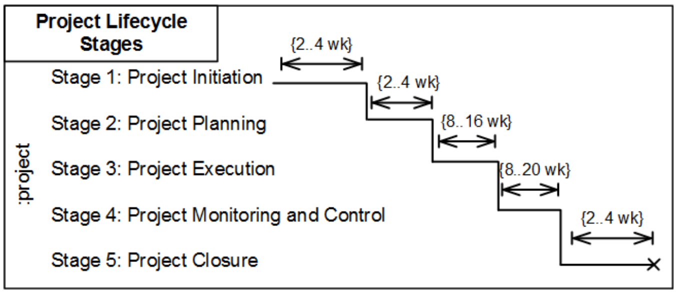

Fig. 7. *UML Timing Model example, Project Lifecycle Stages*

In the example (Fig. 7) of UML Timing model, which presents project lifecycle stages and their duration, certain elements generated from Enterprise model can be found: lifeline – project, state or condition – project stages, timelines, duration and timing constraints and destruction occurrence, where destruction event is depicted by a cross in the form of an X at the end of a timeline.

CONCLUSIONS

Qualitatively realized information system development lifecycle phases are very important for the success of whole IS development process and mistakes made in that phases will cause huge problems and cost a lot of time and expenses to solve it.

Enterprise meta-model as a supplementary Meta-Object Facility layer insures that enterprise model includes business management information process fundamental attributes and provides a knowledge base as a CASE Tool subsystem, which insures quality and verified knowledge in specific scenarios.

Each element of UML models can be generated from the EM using knowledge based enterprise model.

Method of UML models generation process from enterprise model could implement whole knowledge-based IS development cycle design phase. This is partially proved by the example with UML Timing model elements generation process.

The future work is to approve the use of described method of generation with more detailed UML models examples and to create partly implemented prototype.

REFERENCES

[1] Butleris, R., Lopata, A., Ambraziūnas, M., Veitaitė, I., Masteika S., “SysML and UML models usage in knowledge based MDA process,” Elektronika ir elektrotechnika, vol. 21(2), pp. 50-57, 2015. Print ISSN: 1392-1215, Online ISSN: 2029-5731.

[2] IEEE Computer Society, “Guide to the Software Engineering Body of Knowledge SWEBOK,” 2014. Version 3.0. Paperback ISBN-13: 978-0-7695-5166-1.

[3] ISO/IEC JTC 1, Information technology standards, 2016, http://www.jtc1-sc7.org/

[4] Gudas S, “Enterprise knowledge modelling: Domains and aspects. Technological and economic development of Economy,” Baltic Journal on Sustainability, pp. 281–293, 2009.

[5] Gudas S., “Architecture of Knowledge-Based Enterprise Management Systems: a Control View,” Proceedings of the 13th world multiconference on systemics, cybernetics and informatics (WMSCI2009),) July 10-13, 2009, Orlando, Florida, USA, vol. III, pp.161-266 ISBN-10: 1-9934272-61-2 (Volume III). ISBN-13: 978-1-9934272-61-9 (Volume III).

[6] Gudas S., “Informacijos sistemų inžinerijos teorijos pagrindai.” Vilniaus universiteto leidykla, 2012. ISBN 978-609-459-075-7.

[7] Gudas s., Lopata A., “Meta-Model Based Development Of Use Case Model For Business Function,” Information Technology And Control, vol. 36(3), pp. 302-309, 2007. ISSN 1392 – 124X 2007.

[8] Henderson, J., Venkatraman, N., “Strategic Alignment: Leveraging Information Technology for Transforming Organizations,” IBM Systems Journal, vol. 38(2-3), pp.472-484, 1999.

[9] Lopata A., “Veiklos modeliu grindžiamas kompiuterizuotas funkcinių vartotojo reikalavimų specifikavimo metodas,” Dissertation, 2004.

[10] Lopata, A., Veitaitė, I., Gudas, S., Butleris, R., “CASE Tool Component – Knowledge-based Subsystem. UML Diagrams Generation Process,” Transformations in Business & Economics, vol. 13(2B(32B)) pp. 676-696, 2014. ISSN: 1648 – 4460.

[11] Lopata, A., Veitaitė, I., “UML Diagrams Generation Process by Using Knowledge-Based Subsystem,” International conference “15th International Conference on Business Information Systems,” BIS2013 (5th Workshop on Applications of Knowledge-Based Technologies in Business (AKTB 2013)), 2013.

[12] Lopata A., Ambraziūnas M., Gudas S., Butleris R., “The Main Principles of Knowledge-Based Information Systems Engineering,” Electronics and Electrical Engineering, vol. 11(1(25)), pp. 99-102, 2012. ISSN 2029-5731.

[13] Lopata A., Ambraziūnas M., Gudas S., “Knowledge Based MDA Requirements Specification and Validation Technique,” Transformations in Business & Economics, vol. 11(1(25)), pp. 248-261, 2012.

[14] Lopata, A., Ambraziūnas, M., Gudas, S., “Knowledge-based MDA requirements specification and validation technique,” Transformations in Business & Economics , 11(1(25)), pp. 248-260, 2012. ISSN 1648-4460.

[15] Morkevicius A., Gudas S., “Enterprise Knowledge Based Software Requirements Elicitation,” Information Technology and Control, vol. 40(3), pp. 181-190, 2011. 1392 – 124X.

[16] OMG UML, Unified Modelling Language version 2.5. Unified Modelling, 2017, // http://www.omg.org/spec/UML/2.5

[17] Perjons, E., “Model-Driven Process Design. Aligning Value Networks, Enterprise Goals, Services and IT Systems,” Department of Computer and Systems Sciences, Stockholm University. Sweden by US-AB, Stockholm, 2011. ISBN 978-91-7447-249-3.

[18] Plazaola, L., Flores, J., Vargas, N., Ekstedt, M., “Strategic Business and IT Alignment Assessment: A Case Study Applying an Enterprise Architecture-based Metamodel,” Proceedings of the 41st Hawaii International Conference on System Sciences, 2008. 1530-1605/08.

[19] Vargas N., “A Unified Strategic Business and IT Alignment Model: A Study in the public universities of Nicaragua,” Licentiate Thesis in Computer and Systems Sciences Royal Institute of Technology, KTH Stockholm, Sweden, 2010. ISBN 978-91-7415-646-1.

[20] OMG MOF, Meta Object Facility, http://www.omg.org/spec/MOF/2.5.1

[21] Veitaitė I., Lopata A., “Additional Knowledge Based MOF Architecture Layer for UML Models Generation Process,” 18th International Conference on Business Information Systems, BIS2015 International Workshop, Series: Lecture Notes in Business Information Processing, 2015.

[22] Veitaitė I., Ambraziūnas M., Lopata A., “Enterprise Model and ISO Standards Based Informations System’s Development Process,” 16th International Conference on Business Information Systems, BIS2014 International Workshop, Larnaca, Cyprus, May 22-23, 2014, Series: Lecture Notes in Business Information Processing, 2014.