

# Towards a Structure for the Computation Independent Model

Samir Kherraf<sup>1</sup>, Alexandre Moïse<sup>2</sup>, Éric Lefebvre<sup>1</sup> and Witold Suryn<sup>1</sup>

<sup>1</sup> École de Technologie Supérieure  
Information Technologies and Software Engineering Department  
100 Notre-Dame West street, Montreal, Quebec, H3C 1K3, Canada

<sup>2</sup> Almonix inc., 5524 Saint-Patrick street, office 565, Montreal, Quebec, H2C 1X7, Canada

**Abstract.** Model Driven Architecture (MDA) is composed of a hierarchy of three levels of models: computation independent model (CIM), platform independent model (PIM) and platform specific model (PSM). Currently, there exists no formal CIM definition describing its structure of elements to which transformation rules can be applied. It is therefore the goal of this paper to attempt such a structure. In our view, the CIM can be divided into a hierarchy of three interconnected models: the business motivation model (BMM), the business process model (BPM), and the requirements model (RM). This paper presents how these three models can account for the CIM and how they are related to each other.

## 1 Introduction

Model Driven Architecture (MDA) is an architecture proposed by the Object Management Group (OMG) which defines it as “an approach to using models in software development” [1]. MDA is composed of a hierarchy of three levels of models where each level represents a viewpoint, i.e. an abstraction, of a system or its requirements: computation independent model (CIM), platform independent model (PIM) and platform specific model (PSM). A model from one level is obtained through manual or, ideally, automatic transformations from a model of the above level. Thus, the CIM transforms into a PIM, the PIM transforms into a PSM, and the PSM transforms into source code.<sup>3</sup>

In defining MDA, the OMG limited itself to the “what?” rather than the “how?” leaving the task of defining the former to tool vendors or academics. While many efforts were invested in the transformation from a PIM to lower levels, there is still plenty to be done at the CIM level. However, before attempting to formalize a set of rules to transform a CIM into a PIM, it is mandatory to define precisely what the CIM is all about.

Currently, there exists no formal CIM definition, on the one hand, describing its structure of elements to which transformation rules can be applied and, on the other

<sup>3</sup> Source code is considered a model of the execution of the system, but is not part of the MDA specification.

hand, linking these elements to business intent. For instance, a set of CIM and PIM interpretation attempts are summarized in [2]. The authors, adopting an information systems perspective, have also proposed their own definition by dividing the CIM into a Human Intelligence Model and an Artificial Information Model. Nonetheless, the structure of these models and the relations of its elements to business intent are not addressed.

It is therefore the goal of this paper to attempt the development of such a structure. In our view, the CIM can be divided into a hierarchy of three interconnected models: the business motivation model (BMM), the business process model (BPM), and the requirements model (RM). Starting with the OMG definition of the CIM, the remainder of this paper defines the CIM structure from the perspective of these three models.

## 2 The CIM According to the OMG

The following statements summarize the CIM definition from the OMG [1]:

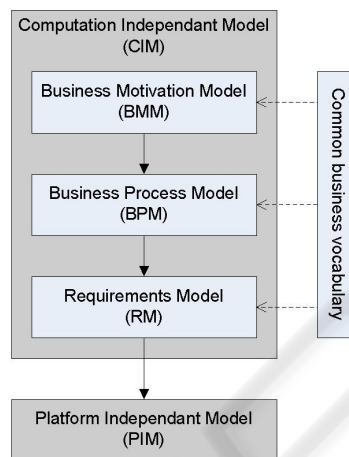
- 1 It is independent of any computation;
- 2 It does not show details of the structure of a system;
- 3 It is often called a domain model or a business model and uses a vocabulary that is familiar to the practitioners;
- 4 It bridges the gap between domain experts and technical experts;
- 5 It shows the system in the environment in which it will operate;
- 6 Its elements should be traceable to PIM elements.

This set of statements constitutes an overview of what a CIM is. However, it does not define precisely the structure of elements of this level of the MDA which, according to point 6, is needed to be transformed into a PIM. Some work has been done in that direction (e.g. [3]), but the lack of precision impedes on a consensual definition. The analyst is thus left to his/her experience and creativity when in charge of creating a CIM. In order to standardize this task, we propose a structure of elements divided into three models as presented in figure 1.

Moving upward from the PIM, the third model of the CIM stack is the RM. This model presents what the system must do conceptually. Use cases are generally employed to this end [4][5][6][7]. Requirements should treat the system as a "black box" in order to comply with points 1 and 2. Point 3 suggests that requirements are restricted to the application domain vocabulary, thus, avoiding technical details such as database concepts or mentioning user interface widgets. It is demonstrated in [4] how to transform the RM into a PIM. This transformation shows that the RM conforms to points 4 and 6.

The second model is the BPM. It describes the operations that need to be supported by the system described in the RM. More precisely, it describes the sequence of activities needed to achieve an organizational goal [8]. These activities generally transform information, either manually or by automated mechanisms, and are realized by different organizational units. Thus, the BPM is compliant to points 3 and 5. The BPM describe the environment in which the system will operate in a way that is relevant to practitioners.

The uppermost model of the CIM stack is the BMM. This model describes the organization for which the processes are created and the goals they aim to support. It represents a knowledge model of CIM. The BMM is a specification of the OMG [9]. Simply put, it consists of a meta-model of the main elements necessary to describe business plans: end, means, influencer and assessment. It complements the BPM in covering points 3 and 5.



**Fig. 1.** Overview of the proposed CIM structure.

When creating a CIM, the starting point then should be the BMM. It describes what the business is all about. From this description, a BPM can be created to support the means to the ends defined in the BMM. Based on the BPM, the RM can finally be created to support automation.

### 3 The Proposed CIM Definition

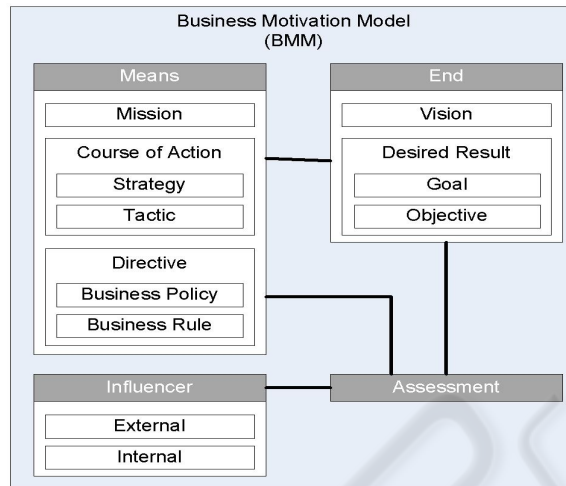
The previous section presented an overview of the proposed CIM structure. This section describes in more details the relationships between the three models composing the CIM in the order of highest to lowest level: BMM, BPM and RM.

#### 3.1 Business Motivation Model

Figure 2 presents the BMM according to [9]. There are four main elements: end, means, influencer and assessment.

**End.** An end describes what an organization wishes to achieve. For example, it can wish to be the leader in market shares or reduce financial risks. An end can be either a vision or a desired result, which in turn can be a goal or an objective. A vision is optional. It gives an overview of the end. On the other hand, a desired result, whether a goal or an objective, is more specific. An objective is a step in the attainment of a goal.

It can be used to measure progress towards a goal. While a goal is normally long-term and can be defined qualitatively, objectives have an end date and must be defined in such a way that it is possible to determine if it has been reached or not.



**Fig. 2.** Structure of the BMM [9].

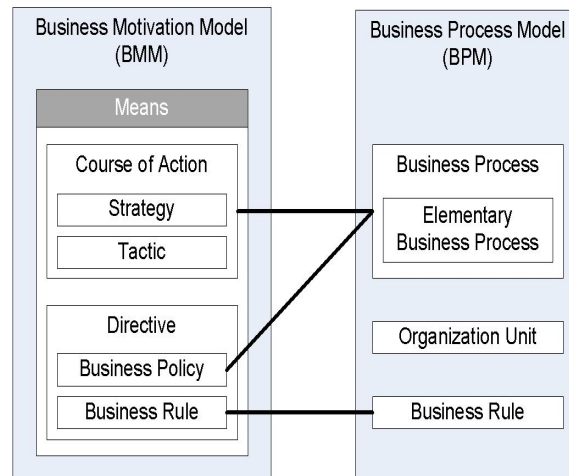
**Means.** Means describe how an organization decides to achieve an end. Means can be a mission, a course of action or a directive. Although optional, the mission broadly defines the activity of an organization. Course of actions defines what needs to be done, but not how to do it. It can be either a strategy or a tactic although [9] does not precisely distinguish the two. Course of actions are related to desired results. It is suggested that strategies are related to the attainment of goals and tactic are related to the attainment of objectives. A directive, which can be either a business policy or a business rule, governs courses of action. Business policies set limits on what can and cannot be done. Business rules are said to be actionable and are derived from business policies.

**Influencer.** An influencer is supposed to affect means or ends. It can be external or internal. External influencers can be of different types such as competitors, customers, environments, partners, regulations or technology. Internal influencers can also be of different types such as corporate value, habits or infrastructure.

**Assessment.** An assessment is a judgment of the impact of an influencer on means and ends. Influencers are said to be neutral until an assessment is made about them.

### 3.2 Business Process Model

In its BMM specification [9], the OMG relates elements of the BMM to the following external elements: business process, organization unit and business rule. These three elements compose the BPM which is defined in more details in [10]. Figure 3 presents the BPM and how it relates to the BMM.



**Fig. 3.** Structure of the BPM.

**Business Process.** A business process realizes a course of action, mainly in terms of sequences of steps. In other words, the justification for a business process is its related BMM course of action. Business processes are governed by business policies.

A business process can be decomposed into smaller business processes. It can be decomposed all the way to an elementary business process (EBP) which is defined by [11] as “a task performed by one person in one place at one time, in response to a business event, which adds measurable business value and leaves the data in a consistent state (...)”.

**Organization Unit.** In the BPM, business processes are under the responsibility of an organization unit. An organization unit can be any structural element of an organization, either internal or external, such as a business partner, a department, a team or a role.

**Business Rule.** BPM business rules come directly from the BMM business rules. They serve to guide business processes in that they represent an obligation or a necessity. According to [9], “they provide the basis for decisions that need to be made within business processes”.

### 3.3 Requirements Model

Systems development is based on defined requirements. Figure 4 presents the RM and how it relates to the BPM. In our view, the RM is composed of three elements: use case, actor and business rule.

**Use Case.** Simply put, a use case describes the interactions between an actor and the system through a sequence of steps called a scenario. A use case must have a main

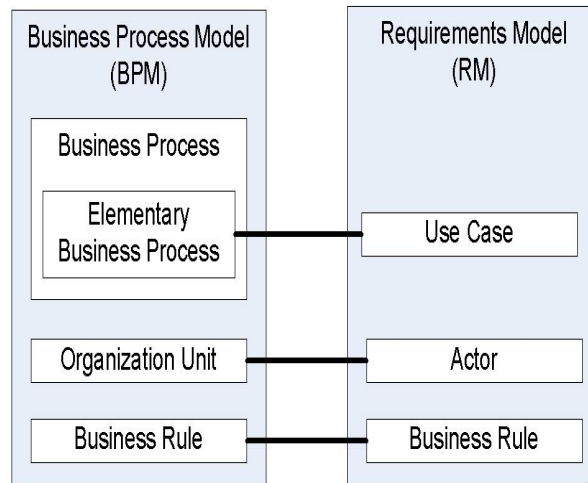


Fig. 4. Structure of the RM.

scenario that describes what should happen most of the time. It can also have alternate scenarios that describe what should happen in case of an exception.

A use case describes the details of an EBP. Thus, the RM defines only structural relationships between use cases, i.e. through «include» and «extend» associations. The RM does not define flow between use cases. It is defined in the BPM.

The transformation of an EBP into a use case has been demonstrated by [4]. Furthermore, it is demonstrated in [4] how to transform a use case into a PIM by using patterns and archetypes.

**Actor.** An actor is an entity, either physical or logical, which benefits from the use case. A physical actor is a person such as manager or a clerk, while a logical actor is a group of persons such as a department or a system such as a banking system. A use case has one primary actor, but can have a secondary actor. The latter does not benefit from the use case, but is indispensable in its completion. For instance, in a use case about credit card payment, the customer is the primary actor, but the system must rely on a secondary actor - a banking system to validate the payment - in order to achieve the goal. An actor is directly related to an organization unit in the BPM.

**Business Rule.** A RM business rule is related to a BPM business rule. In a use case, business rules trigger alternate scenarios.

## 4 Conclusions

To facilitate the creation of a CIM, it is important to define a process for transforming, respectively, a BMM into a BPM and a BPM into a RM. A set of rules and a process to transform a BPM into a RM is proposed in [4]. Improvements and implementation of these are in progress.

Using the BMM as the first model to be specified within the CIM provides a simple mean to manage high-level business requirements. It is sufficiently generic to capture requirements from any business domain. A software tool will provide a better understanding of the relationships between the BMM, the BPM and the RM.

In order to improve our inter-model transformation strategy and strengthen the specification of source and generated models, it is essential to define a UML profile to define each of the three models that constitute the CIM. The definition of a UML profile conforming to the MOF will greatly facilitate the inter-model transformations.

Traceability plays an important role in our proposal. Transformation rules that will be proposed must allow traceability between elements of each model as partially proposed in [4]. As such, relationships between these elements must be made explicit.

Finally, the transformation process will be supported by a tool that is yet to be implemented.

## References

1. Joaquin Miller and Jishnu Mukerji, MDA Guide Version 1.0.1.,OMG (2003)
2. Kirikova, M., A. Finke, and J. Grundspenkis, What Is CIM: An Information System Perspective, in *Advances in Databases and Information Systems*, (2010) p. 169-176
3. Osis, J., E. Asnina, and A. Grave, Formal Problem Domain Modeling within MDA, in *Software and Data Technologies*. Springer Berlin Heidelberg. (2009) p. 387-398
4. Kherraf, S., E. Lefebvre, and W. Suryan, Transformation from CIM to PIM Using Patterns and Archetypes, in *Proceedings of the 19th Australian Conference on Software Engineering*. IEEE Computer Society, (2008)
5. Roussev, B., Generating OCL specifications and class diagrams from use cases: a Newtonian approach, in *Proceedings of the 36th Annual Hawaii International Conference on Systems Sciences*, (2003)
6. Giganto, Reynaldo and Smith, Tony, Derivation of Classes from Use Cases Automatically Generated by a Three-Level Sentence Processing Algorithm, *ICONS '08: Proceedings of the Third International Conference on Systems*. IEEE Computer Society, Washington, DC, USA, (2008) 75-80
7. Anda, Bente and Sjberg, Dag I. K., Applying Use Cases to Design versus Validate Class Diagrams - A Controlled Experiment Using a Professional Modelling Tool, *ISESE '03: Proceedings of the 2003 International Symposium on Empirical Software Engineering*, IEEE Computer Society, Washington, DC, USA, (2003) 50
8. Wfmc, Workflow Management Coalition, Interface 1: Process Definition Inter-change Process Model Wfmc-TC-1016-P, (1999)
9. OMG, Business Motivation Model (BMM) Specification September (2007)
10. OMG, Business Process Definition MetaModel (BPDM), Process Definitions, dtc/2008-05-09 (2008)
11. Larman, C., Applying UML and Patterns. Third ed. Prentice Hall (2004)