

Original Research Paper

An Automatic Transformation Method from the E3value Model to IFML Model: An MDA Approach

¹Nassim Kharmoum, ¹Soumia Ziti, ²Yassine Rhazali and ¹Fouzia Omary

¹Department of Computer Science, Intelligent Processing Systems and Security Team, Faculty of Sciences, Mohammed V University in Rabat, Morocco

²Department of Computer, Information and Communication Systems Engineering Research Group, Higher School of Technology, Moulay Ismail University in Meknes, Morocco

Article history

Received: 23-02-2019

Revised: 02-04-2019

Accepted: 10-06-2019

Corresponding Author:

Nassim Kharmoum
Department of Computer
Science, Intelligent Processing
Systems and Security Team,
Faculty of Sciences,
Mohammed V University in
Rabat, Morocco
Email: nkharoum@gmail.com

Abstract: Nowadays the software industry has known a significant growth, while the end-users have become very demanding. In this sense, the model transformation has become one of the essential solutions to ensure competitiveness in the field of the software industry domain. For that, the Object Management Group (OMG) proposes for the Model-Driven Architecture (MDA) approach three abstraction levels, namely Computation Independent Model (CIM), Platform Independent Model (PIM) and Platform Specific Model (PSM). Therefore, our contribution in this paper is to shed more light on the first MDA transformation, which is the transformation from CIM to PIM levels. We propose a disciplined Model-Driven approach for the e-business information system, which generates the IFML (Interaction Flow Modeling Language) model automatically in the PIM level from business value model in the CIM level, using the ATL transformation language. For this purpose, the business value model is illustrated by the E3value model, whereas, the IFML model exhibits the front-end applications content, interface composition, user interaction and control behavior for several kinds of applications, such as web applications, mobile applications and desktop applications.

Keywords: Model-Driven Engineering, Model-Driven Architecture, Model Transformation, Automatic Transformation, Computation Independent Model, Platform Independent Model, CIM to PIM Transformation, Meta-Model, E3value, Business Value Model, Interaction Flow Modeling Language

Introduction

Since the advent of the Model-Driven Architecture (MDA) approach, the model transformation has become one of the essential solutions to ensure competitiveness in the software's field industry, which grows exponentially, while the end-users have become very demanding. To respond to this demand, the Object Management Group (OMG) (Colomb *et al.*, 2006) proposed the Model-Driven Architecture (MDA) approach (OMG-MDA, 2014) at the beginning of the 21st-century. This approach put the models at the center of the development process for any software system (Maatougui *et al.*, 2016), based on the model's transformation task. In this context, the OMG proposes three abstraction levels for the MDA

approach, which are Computation Independent Model (CIM), Platform Independent Model (PIM) and Platform Specific Model (PSM).

The CIM level is considered as the higher abstraction level. At this level, we specify customer requirements without providing any technical considerations or system implementation. The PIM level presents the average abstraction level, at this level; we design the collected customer requirements. However, models at the PIM level do not contain any implementation and technical details. The PSM level is treated as the lowest abstraction level and it is related to the platform execution.

In practice, most researchers avoid the first transformation from CIM to PIM levels because these

levels contain higher abstraction model nature (Rhazali *et al.*, 2018), such as requirements models. In this contribution, we put much focus on the construction and the transformation of our proposed CIM to PIM levels; the CIM level is expressed by the E3value model, whereas the generated PIM level is represented by IFML model (OMG-IFML, 2015; Brambilla and Fraternali, 2014). So, our E3value source model presents the value-based requirements engineering models (Gordijn and Akkermans, 2003; Garrigo's *et al.* (2012) and unifies the e-business models notations from a value viewpoint, by showing how the economic value is created, exchanged and consumed within a network of actors (Gordijn *et al.*, 2001). Thus, the business value model makes models more comprehensible by <<Business Executives>> and <<Business Value Analysts>>. The target model that is generated automatically will be presented by IFML model and is considered as one of the OMG modeling standard (OMG-IFML, 2015; Brambilla and Fraternali, 2014). Thus the IFML is a visual modeling language that expresses front-end applications content, interface composition, user interaction and control behavior for several kinds of applications, such as web applications, mobile applications, desktop applications (Brambilla and Fraternali, 2014; Hamdani *et al.*, 2018; Laaz *et al.*, 2018). So our method facilitates bridging the gap between the business value model and the interaction flow models, by allowing stakeholders who do not know about IFML creation to generate their own IFML models.

We generate automatically our model utilizing ATLAS Transformation Language (ATL) (Jouault *et al.*, 2006) that employs the following OMG standards; such as Object Constraint Language (OCL) (OMG-OCL, 2014), Meta Object Facility (MOF) (OMG-MOF, 2015) and XMI Metadata Interchange (XML) (OMG-XMI, 2015), which speed up the development of our e-business information systems process by reducing time and effort, therefore reducing the cost to guarantee competitiveness in the software industry.

We structure the remainder of the paper as follows. Next section highlights our related work using the chosen models in our method. Later, we enlighten our proposal; it shows the source and the target meta-models and explains all transformation rules used in our method. After that, we present a case study illustrating our method. Then, we analyze and discuss all the obtained results. Finally, we conclude our contribution by proposing our future work.

Related Work

This section highlights different work was done over the previous decade which practices the chosen

models in this paper, particularly the E3value model at the CIM level and the generated front-end interfaces models at the PIM level while respecting the MDA approach. Up to now, few methods focus on our chosen models, but we achieved to detect some work done in this regard. We are starting with papers based on E3value model in their source model. For this purpose, we have Pijpers and Gordijn (2007) in their e3transition approach, they start with the E3value model that presents the business Value models of a value web, to arrive at the Process Models of the same value web, through the e3transition models.

Also, Schuster and Motal (2009) propose an enhancement of the Business Modeling Ontology (BMO) (Osterwalder and Pigneur, 2002), they translate the e3-value notation into an REA-stereotyped (McCarthy, 1982) UML (OMG-UML, 2017) class diagram, applying their own mapping rules.

Besides, De Castro *et al.* (2011) apply the CIM to PIM model transformation for the service-oriented development of information systems. They modeled their CIM model using both the e3vlaue model and the BPMN model (OMG-BPMN, 2011), to generate the UML Use case diagrams (use case diagram and extended use case diagram) and the UML Activity diagrams (process model Service and Service composition model) in their PIM level. The transformation is made in a semi-automatic transformation way basing on ATL language.

After highlighting different methods based on E3value model in their source model, now we survey various work that generates the front-end interfaces models. First, Brambilla and Fraternali (2011) generate the WebML diagram (Ceri *et al.*, 2000), but in this case, they based on BPMN diagram as source model and the transformation are made using their own WebRatio (Acerbis *et al.*, 2007) extension.

Furthermore, Grigera *et al.* (2012) transform the Interface Mockups automatically and WebSpec diagram (Luna *et al.*, 2011), which present the requirement artifacts, to WebML diagram that gives the Web Modeling Language, basing on their own WebSpec tool.

Rhazali *et al.* (2016a) generate the IFML model automatically from UML Class diagram basing on ATL transformation language in the PIM level. In other work, the same authors generate automatically basing on ATL transformation language in Rhazali *et al.* (2016b); the UML Class diagram structured through MVC (Krasner and Pope, 1988) from UML Class diagram at the PIM level. Finally, the authors Rhazali *et al.* (2018) generate a Web modeling diagram from UML diagrams precisely since UML Use Case diagram, UML State diagram and UML Class diagram at the PIM level. After highlighting the above work, we

notice for the CIM and PIM levels creation manner that all the work uses a graphical representation, which is recommended by the MDA approach. Besides we have some methods generate their models in a non-automatic way using human language, other methods generate their models dealing with their own tools. Also, until now no author has generated front-end interfaces models from the E3value model, whereas, most of generated front-end interfaces model methods are based on several transformation stages and start at the PIM level.

To do so, we propose in this paper a new method that automatically generates front-end interfaces model based on Interaction Flow Modeling Language in the PIM level, from the E3value model presenting the business value model in the CIM level; since all the highlighted methods do not meet our proposal. The following section will take a closer look at our proposed method.

Our Proposal

In this section, we will shed more light on our disciplined proposed method, which is intended to construct and transform the CIM level automatically to the PIM level for e-business information systems, respecting the MDA approach. We stress the fact that our proposal results from our last analytical studies (Kharmoum *et al.*, 2016; 2018; 2019).

Our objective in this method (Fig. 1) is to generate the IFML model automatically in the PIM level from the E3value model in the CIM level. The target IFML model gives the behavior and interaction model and offers front-end applications content, interface composition, user interaction and control behavior. Thus, IFML is a platform independent language, it has been designed with

an executable semantic and it mapped readily into executable applications for different platforms and devices (Gotti and Mbarki, 2016) for instance web applications, mobile applications and desktop applications. Also IFML is an OMG fully-recognized standard (Brambilla and Fraternali, 2014). Whereas, the source E3value model presents the business value model and the value-based requirements. We add that the transformation is done automatically via ATL transformation language. The purpose is to have an e-business system containing correct E3value model and IFML model. We aim to use the models of our method as a support for the system's understanding and aiding communication between technical and e-business stakeholders such as <<Developers>>, <<Systems Engineer>>, <<Business Executives>> and <<Business Analysts>>. And to achieve the development process of our e-business information system by generating other models in the lower abstraction level PSM.

To explain our method according to the MDA approach, Fig. 2 explains its transformation process. We start with the definition of meta-models; the purpose is to create and generate automatically correct models (Rodríguez *et al.*, 2010). For that, our E3value source meta-model describes the structure of our business value model on the higher abstraction level. Then, we determine the target IFML meta-model representing the structure of our generated IFML's model in the average abstraction level. Next, we specify the constraints during the source model construction. After that, we explain the set of transformation rules implemented via ATL transformation language to generate our target models automatically from the source model. Finally, we highlight the generated target IFML model.

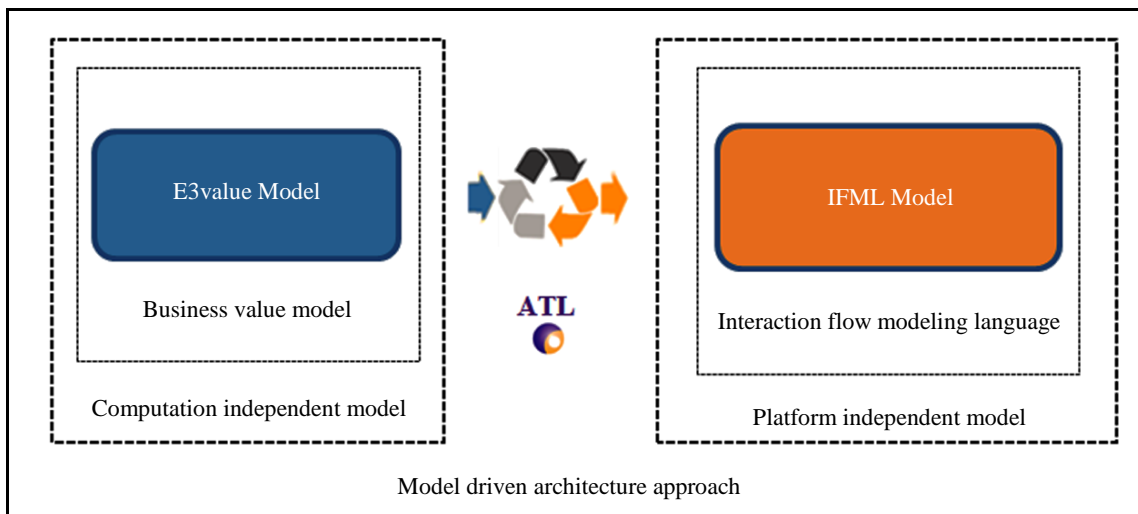


Fig. 1: Our proposed method

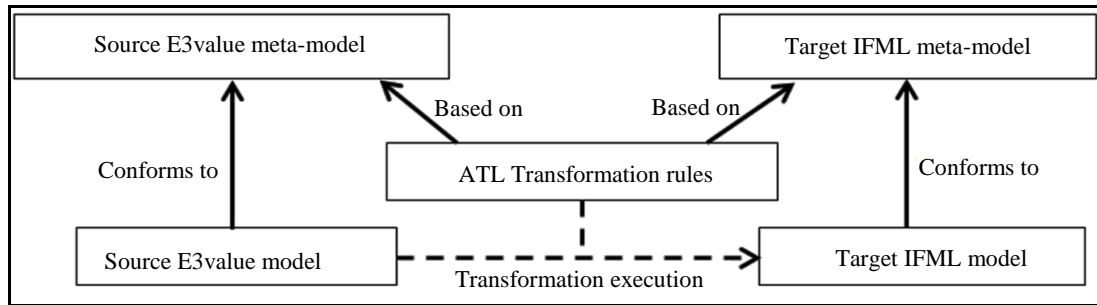


Fig. 2: The Transformation process of our proposed method

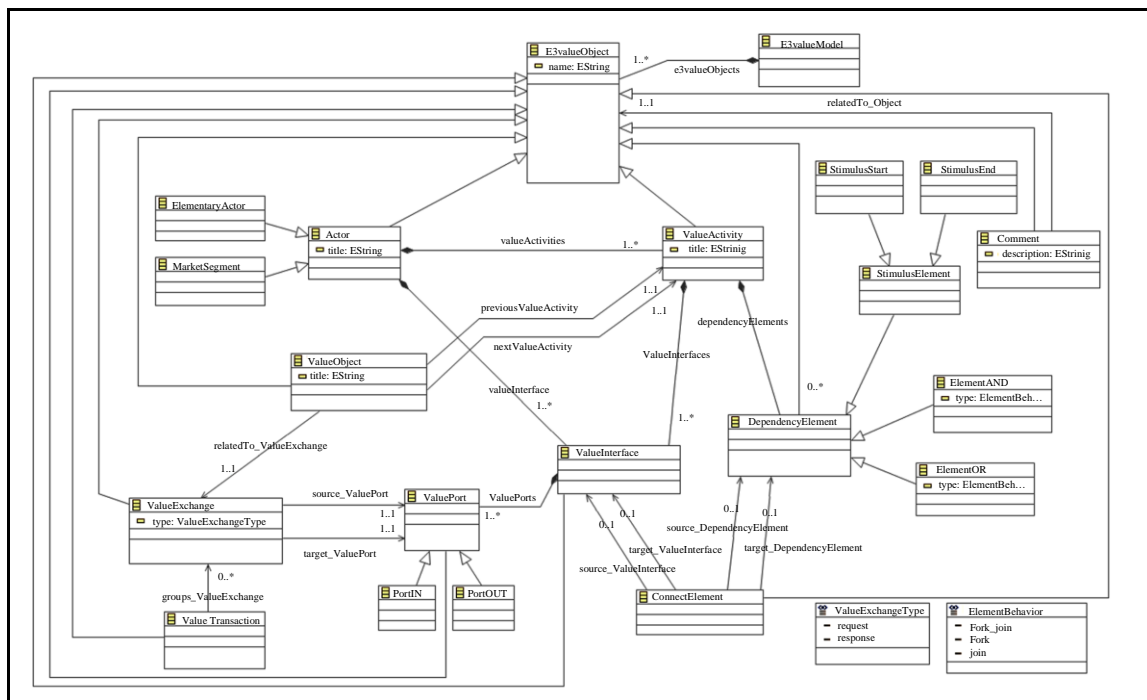


Fig. 3: Our E3value source meta-model

Source E3value Meta-Model

We describe in this part the construction of our E3value source meta-model (Fig. 3) that any created E3value model must conform to. The purpose is to produce a correct model explaining the manner that economic value is created, exchanged and consumed in a network of actors. Since, our E3value meta-model elements are taken from (Gordijn and Akkermans, 2003; Gordijn *et al.*, 2006; Gordijn and Akkermans, 2018).

Target IFML Meta-Model

We describe in this part the construction of our IFML target meta-model (Fig. 4) that any generated IFML model must conform to. The IFML model has generated

automatically from the E3value source model and express front-end applications content, interface composition, user interaction and control behavior. Our IFML meta-model elements are extracted from (OMG-IFML, 2015; Brambilla and Fraternali, 2014).

Source E3value Model Construction Rules

To create a correct source E3value model; the only constraint is to respect the previously explained and illustrated of the E3value meta-model in Fig. 3.

Transformation Rules from E3value Model to IFML Model

This section explains the E3value model transformation rules toward the IFML model based on its

meta-model and using the ATL transformation language (Fig. 5), to ensure the transformation automation:

- Rule 1: Presents the transformation from <<E3value Comment>> to <<Annotation>>. The generated <<Annotation>> has an <<id>>, which is inspired from the related <<E3value Comment>> element <<name>>. Whereas, <<text>> contains the <<E3value Comment description>>
- Rule 2: Allows transforming <<E3value Actor>> to <<Module Definition>> and <<View Container>> the <<Actor>> can be <<Elementary Actor>> or <<Market Segment>>. The generated <<Module Definition>> has as attributes <<id>> (is related to the <<E3value Actor's name>>), <<name>> (is also related to the <<E3value Actor's name>>), <<input Ports>> (presents the <<E3value Actor's Port IN>>), <<output Ports>> (presents the <<E3value Actor's Port OUT>>) and <<view Containers>> (presents the generated <<View Container>> that also contains the attributes <<id>>, <<name>>, <<title>>, <<is XOR>> (with always the value true), <<view Elements>> and <<annotations>>

- Rule 3: Transforms <<E3value Value Object>> to <<Action>>, <<Action Event>> and <<Navigation Flow>>. The generated <<Action>> acts as a sender action and catches one <<Action Event>>. The source and the target <<Action Event>> are made through <<Navigation Flow>>. All the generated elements have <<id>> and <<name>> which are inspired from the related <<E3value>> element <<name>>
- Rule 4: Generates the output <<Port Definition>> and his related <<Navigation Flow>> from <<E3value PortOUT>> to mention the interaction point between the related <<Module>> and its environment that are going out from the <<Module>>. All the generated elements have <<id>> and <<name>> which are inspired from the related <<E3value>> element <<name>>
- Rule 5: Generates the input <<Port Definition>> and his related <<Navigation Flow>> from <<E3value PortIN>> to mention the interaction point between the related <<Module>> and its environment that are arriving at the <<Module>>. All the generated elements have <<id>> and <<name>> which are inspired from the related <<E3value>> element <<name>>

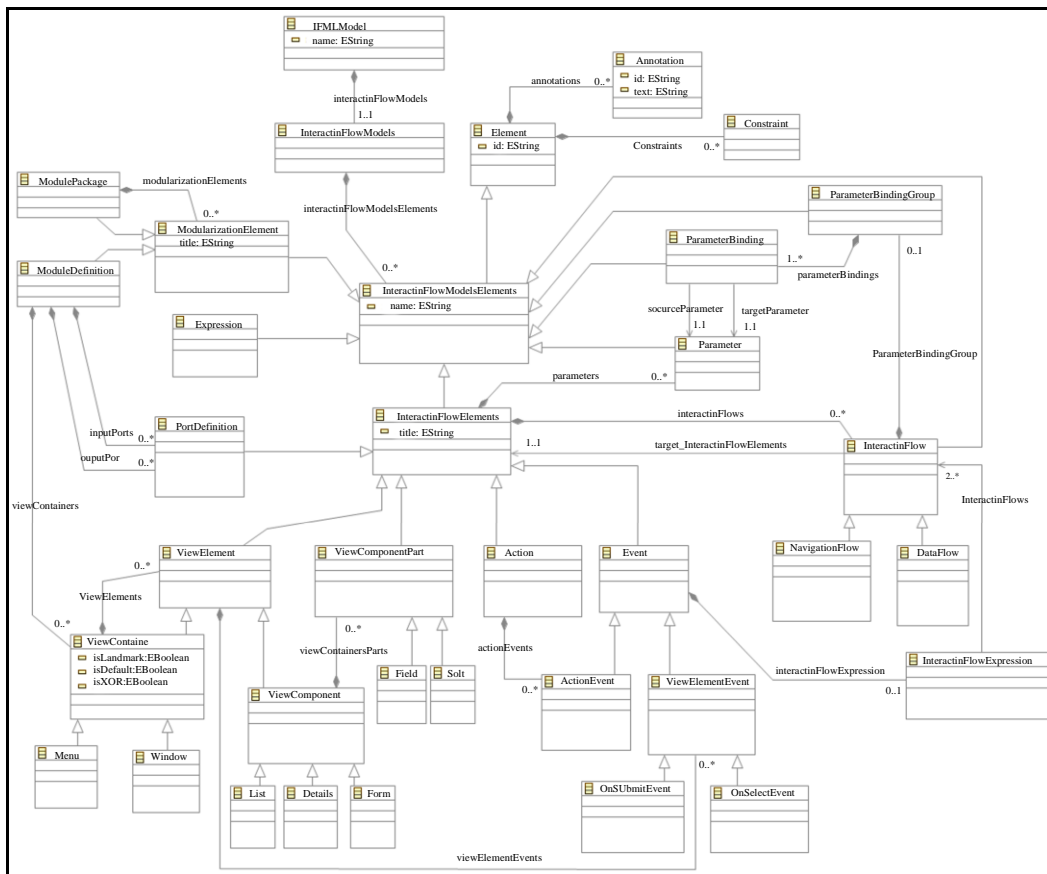


Fig. 4: Our IFML target meta-model (OMG-IFML, 2015)

- Rule 6: Transforms the <<E3value Value Activity>> to the <<View Container>> and <<form>>. The generated <<View Container>> has a true <<is Landmark>> attribute in order to be reachable from any other element of the user interface and contains the generated <<form>> as a <<view Elements>>. All the generated elements have <<id>>, <<name>> and <<title>> which are inspired from the related <<E3value>> element <<name>> and <<title>>
- Rule 7: Transforms <<E3value Value Exchange>>, which is related two <<E3value Value Activities' Ports>> or is related to the <<E3value Value Activity's Port>> and <<E3value Value Actor's Port>>, to the <<View Element Event>> and <<Navigation Flow>>. The generated <<View Element Event>> element catches the generated <<form>> (in Rule6) events. Whereas, the source and the target <<Action Event>> are made through <<Navigation Flow>>. All the generated elements have <<id>> and <<name>> which are inspired from the related <<E3value>> element <<name>>
- Rule 8: Generates the <<IFML Model>> from <<E3value Model>>. The generated model groups <<Interaction Flow Model>>, <<Module Package>> and all the above-generated elements since Rule1 until Rule7

Target IFML Model

The target IFML model presents the result of our previously defined ATL transformation rules (Fig. 5) from the source E3value model and respects its conformed meta-model (Fig. 4). Therefore, the target IFML model has generated automatically and presents the behavior and interaction model and offers front-end applications content, interface composition, user interaction and control behavior.

Case Study

To illustrate our method, this section shows the theoretical and practical aspect of our <<University Library management>> case study transformation within Eclipse editor (including its necessary plug-ins) to explain the automatic generation of the IFML model from the E3value model.

For our eclipse's plug-ins, we use the <<ATL Eclipse plugin>> as a standard tool for transformation (Rhazali *et al.*, 2015). We also focused on the <<IFML editor>> which is an Eclipse plugin based on <<EMF>> and <<Sirius>> (Laaz *et al.*, 2018), to draw our generated IFML model.

As a result, Fig. 6 illustrates our practical case study in three main folders; the first contains meta-models (has extensions <<.ecore>> and <<.ecore diagram>>), the second contains models (with <<.xmi>> extensions) while the third folder embraces our transformation rules (with extensions <<.asm>> and <<.atl>>). For more ergonomics, each folder contains two sub-folders; one for the source (E3value), while the other is for targets (IFML model).

To better position the Fig. 6 in our proposed method, this part allows us to link each file in Fig. 6 with our figures created and generated above:

- <<E3value_metaModel.ecore>> and <<E3value_metaModel.ecore diagram>> illustrate our E3value source meta-model (Fig. 3)
- <<IFML_metaModel.ecore>> and <<IFML_metaModel.ecore diagram>> illustrate our IFML target meta-model (OMG-IFML, 2015) (Fig. 4)
- <<E3value_model.xmi>> represents our proposed E3value-source model in XMI format (OMG-XMI, 2015), that we symbolize it graphically in Fig. 7a
- <<IFML_model.xmi>> represents our generated IFML model in XMI format, that we symbolize it graphically in Fig. 7b
- <<E3valueToIFMLTransformationRules.atl>> and <<E3valueToIFMLTransformationRules.asm>> represent our transformation rules from the E3value model to IFML model. (Fig. 5)

In our case study, we propose the following <<Actors>>: <<Patrons>>, <<Library>>, <<Publishing Companies>> and <<Research Databases>>. We have that every <<Patrons>> can access <<Library>> resources. For instance, we have <<Books>> and <<Online Research Articles>> by paying fees for the chosen access. From its part, the <<Library>> is always examining the <<Patron's>> requests; it lends <<Books>>, browses <<Research Articles>> if the requested need exists, if not, it manages its needs by delivering them to suppliers who are <<Publishing Companies>> to offer the requested <<Books>> and <<Research Databases>> to provide <<Databases>> craved access. The purchase of each <<Book>> or <<Research Database access>> requires fees.

The rest of the section illustrates the <<University Library Management>> case study source and target models.

E3value Model Proposal

We focus in this section on our source model creation, by proposing the business value model of our case study, as illustrated in Fig. 7a.

The <<Actors>> can be <<Elementary Actors>> such as <<library>> or can be <<Market Segments>> like <<patrons>>, <<publishing companies>> or <<research databases>>. Each <<Actor>> can possess at least one <<Value Activity>>. In our model the <<patrons>> can <<access library resources>>, the <<library>> has <<lend books>>, <<browsing research articles>> and <<managing library needs>>; the <<publishing companies>> have <<sell books>> and <<research databases>> have <<sell access to research databases>>. The <<Value Objects>> can be a request like <<access to books registration fees>>, <<online access to research articles registration fees>>, <<books needs>>, <<research articles need>>, <<purchasing books fees>> and <<research databases subscription fees>>. Moreover, <<Value Objects>> can be a response such as <<access to books>>, <<online access to research articles>>, <<access to available books>>, <<access to available research articles>>, <<books>> and <<access to research databases>>. In our case, the E3value model employs <<Value Exchange>> to define the dependency path direction. These dependency paths relate E3value dependency elements, which can be <<Start Stimulus>>, <<End Stimulus>>, <<And-Join>>, <<And-Fork>>, <<Or-Join>> or <<Or-Fork>>.

Generated IFML Model

In this section, we describe our IFML generated model, based on the set of previously defined transformations rule. The Fig. 7b presents our generated IFML model, which is composed of one <<Module Package>>. The <<Module Package>> encompasses all other generated IFML model elements. We have in the <<Module Package>> four <<Modules Definition>> which are <<Library>>, <<Patrons>>, <<Publishing companies>> and <<Research databases>>. For each generated <<Module Definition>> we have an <<XOR View Container>>, such as <<Manage Patrons>>, <<Manage Library>>, <<Manage Publishing companies>> and <<Manage Research databases>>, which allow comprising other child <<View Container>> elements. And as mentioned during the transformation rules: each <<E3value Value Activity>> is transformed to the <<Landmark View Container>> and <<form>>, we have as an example for the <<E3value Value Activity>>, the <<Landmark View Container: <<Manage Accessing library resources>> that contains the <<form: Manage Accessing library resources>>.

Each generated <<form>> can own <<View Element Event>> to catch the sending <<form>> event; for example, the <<form: Manage Accessing library resources>>, we have two <<send>> <<View Element

Event>> because the <<E3value Value Activity>> linked to the two <<E3value Value Exchange>>. In order to present <<actions>> between <<Modules Definition>> or <<View Containers>> (included in the same <<Modules Definition>>), we are based on <<Action>> and their <<Action Event>>; for example, we have <<Send Access to books registration fees>> and <<Send Access to books>> in order to present the request and response <<actions>> between the two <<Modules Definition>>: <<Patrons>> and <<Library>> via the <<Ports Definition>> <<Input Port>> and <<Output Port>>. Alternatively, we have <<Send Books needs>> and <<Send Access to available books>>, to present the request and response <<actions>> between the two <<View Containers>>: <<Manage Lending books>> and <<Manage library needs>> in the same <<Modules Definition>> <<Library>>. Last but not least, all generated <<Navigation Flows>> are used to link the generated IFML elements.

Analysis and Discussion Based on the Evaluation Criteria

In this section, we analyze and discuss our proposed method based on the evaluation criteria of all studied related work. To do so, we concentrate on the OMG recommendations (OMG-MDA, 2014) and following work (Rhazali *et al.*, 2015; Kriouile *et al.*, 2013; Yue *et al.*, 2011) to conclude our evaluation criteria.

So, we arrange our concluded criteria according to their source and target models construction, the model transformation manner and the use of assessment methodology:

- Model construction criteria: present the coverage of the graphical representation for each model. And the participation in one of the MDA approach levels
- Transformation criteria: verify transformation automation and meta-models based transformation to ensure the correctness of the transformation, which are one of the essential concepts in MDA (OMG-MDA, 2014; Cetinkaya and Verbraeck, 2011). Also, we check the existence of the mapping rules definition
- Evaluation approach criteria: Present the occurrence of the assessment methodology for the theoretical and practical ways

The rest of this section analyzes and discuss the studied methods with our proposed method based on our concluded evaluation criteria.

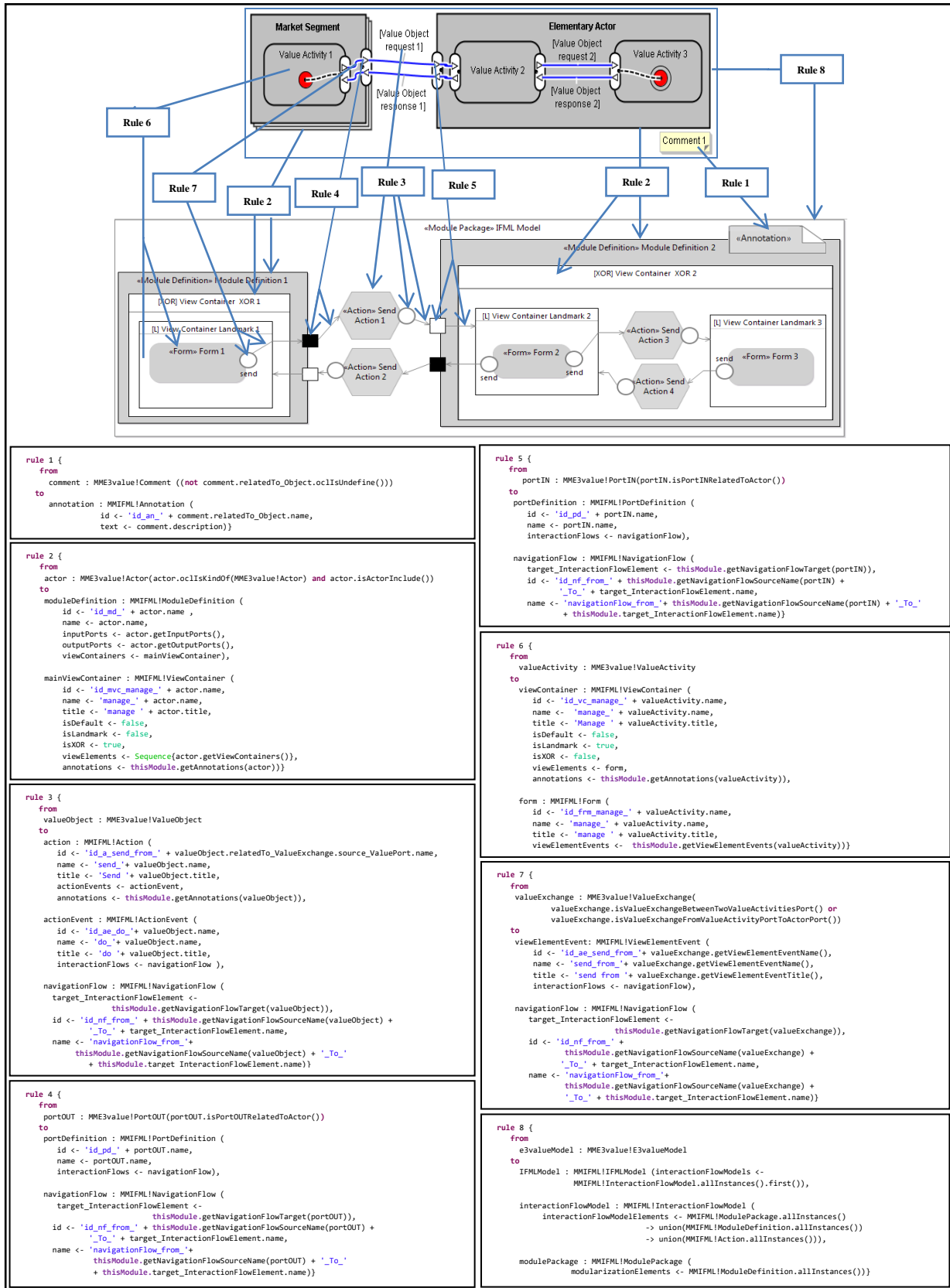


Fig. 5: Transformation rules from E3value model to IFML model

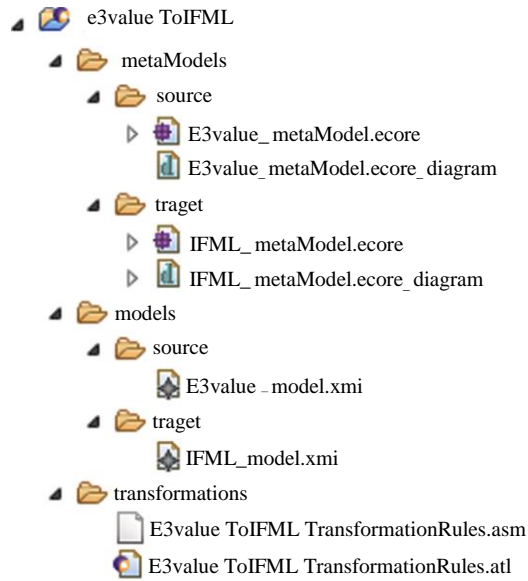


Fig. 6: Our practical case structure

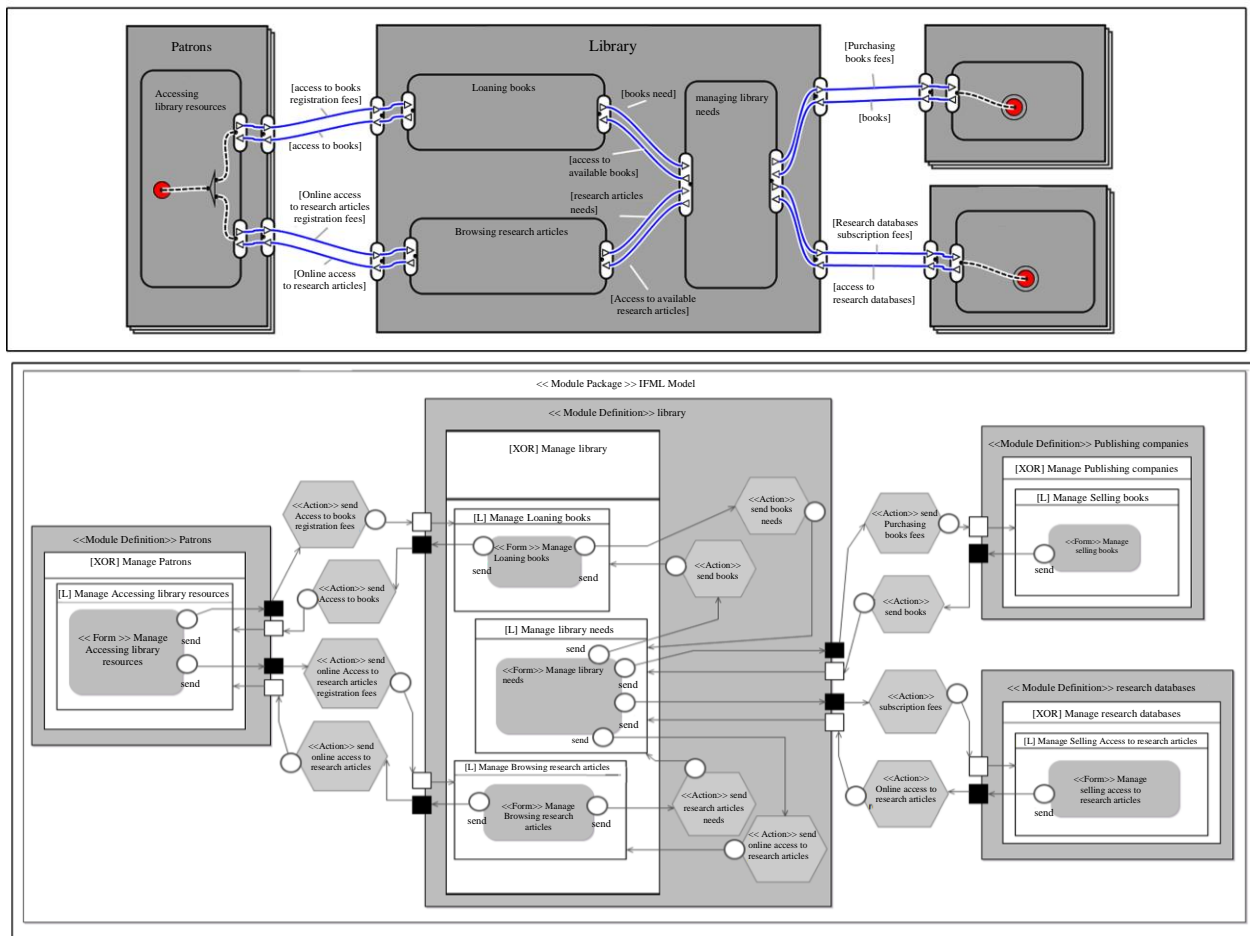


Fig. 7: Our models (a) E3value-source model (b) Generated IFML model

Studied papers	Source model			Target model			Transformation				Assessment Methodology		
	Graphical representation	MDA level	Representation	Graphical representation	MDA level	Representation	Based Meta-model	Is Automatic	Type	Mapping rules Definition	Theoretical	Practical	
Transformation methods From value based model	Pijpers and Gordijn (2007)	Yes		E3value models & E3transition models	Yes		UML activity diagram (High & Low Levels)			Human language	Yes	Case study	
	Schuster and Motal (2009)	Yes		E3value model	Yes		REA model	Yes		Human language	Yes	Case study	
	De Castro <i>et al.</i> (2011)	Yes	CIM	E3value model & BPMN process model	Yes	PIM	UML Use Case diagram & UML Activity diagram	Yes	Yes	Transformation language (ATL)	Yes	Case study	Standard tool (Eclipse)
Transformation methods generating the user front-end interface models	Brambilla and Fraternali (2011)	Yes	CIM	BPMN diagram	Yes	PIM	WebML diagram	Yes	Yes	Transformation Based on an extension of WebRatio tool	Yes		Custom Extension (Extension of WebRatio tool)
	Grigera <i>et al.</i> (2012)	Yes	PIM	Interface Mockups & WebSpec diagram	Yes	PIM	WebML diagram	Yes	Yes	Transformation Based WebSpec tool	Yes	Case study	Tools (UI Mockups & WebSpec)
	Rhazali <i>et al.</i> (2016a)	Yes	PIM	UML Class diagram	Yes	PIM	IFML diagram	Yes	Yes	Transformation language (ATL)	Yes	Case study	Standard tool (Eclipse)
	Rhazali <i>et al.</i> (2016b)	Yes	PIM	UML Class diagram	Yes	PIM	UML Class diagram structured through MVC	Yes	Yes	Transformation language (ATL)	Yes	Case study	Standard tool (Eclipse)
	Rhazali <i>et al.</i> (2018)	Yes	PIM	UML Use Case diagram & UML State diagram & UML Class diagram	Yes	PIM	Web modelling diagram	Yes		Human language	Yes	Case study	Standard tool (Eclipse)
Our proposal	Yes	CIM	E3value model	Yes	PIM	IFML model	Yes	Yes	Transformation language (ATL)	Yes	Case study	Standard tool (Eclipse)	

Fig. 8: Studied papers Comparison via Evaluation criteria

Analysis

We exhibit in Fig. 8 the result of our evaluation criteria analysis. The figure lines present studied papers, whereas, the figure columns list deduced criteria.

We start with model construction criteria; on the one hand, we have all studied methods based on a graphical representation for their source and target models. On the other hand, we have some methods base on the MDA approach making their transformation from CIM to PIM such as

(De Castro *et al.*, 2011; Rhazali *et al.* (2016b) and our proposal. We also have other methods base on the MDA approach, but doing their transformation from PIM to PIM like: (Grigera *et al.*, 2012; Rhazali *et al.*, 2016a-b; 2018). Nevertheless, the methods (Pijpers and Gordijn, 2007; Schuster and Motal, 2009) do not base in the MDA approach, but base on their approach.

Concerning the transformation criteria, we have all methods base on the meta-model in their transformation except the authors of the method (Pijpers and Gordijn, 2007). Thus, for the transformation automation, we notice some ways transform their models based on an ATL transformation language such as (De Castro *et al.*, 2011; Rhazali *et al.*, 2016a-2016b) and our proposal. Likewise, other authors are based on their methods for their tools, for instance, using <<WebRatio>> tool in (Brambilla and Fraternali, 2011) and using <<WebSpec>> tool for (Grigera *et al.*, 2012). But we have other methods that transform their models manually basing on Human language like (Schuster and Motal (2009; Pijpers and Gordijn, 2007) and (Rhazali *et al.*, 2018). Plus, for the mapping rules definition, all studied methods explain their transformation rules, for a better understanding.

Besides, for the evaluation approaches criterion; all methods based on a case study for their theoretical assessment methodology, except the authors of the method (Brambilla and Fraternali, 2011). But, for the practical assessment methodology, we have some methods base on their tool to evaluate method such as in (Brambilla and Fraternali, 2011) the authors use a custom <<WebRatio>> tool Extension and in (Grigera *et al.*, 2012) the authors use both tools: <<UI Mockups>> and <<WebSpec>>. Additionally, other methods deal with standard tool <<Eclipse>> (Rhazali *et al.*, 2015), for example methods (De Castro *et al.*, 2011), (Rhazali *et al.*, 2016a-2016b; 2018) and our proposal.

Discussion

After analyzing all the studied methods, the papers (De Castro *et al.*, 2011; Grigera *et al.*, 2012; Rhazali *et al.*, 2016a; 2016b) validate all our proposed evaluation criteria; they generate a graphical target models automatically from a graphical source model basing on their meta-models, also they explain the mapping rules during the transformation, respecting the MDA approaches.

Besides, on our proposal we validate all evaluation criteria also we are the unique method that generates

the IFML model in the PIM level starting with a straightforward model which is E3value model in CIM level. Further, the executions of our transformations rules are meta-model based; forasmuch these diagrams are generated automatically using ATL Transformation language and we use theoretical and practical assessment methodology based on a case study with a standard tool.

Finally, using our method for e-business systems will allow generating correct front-end applications content, interface composition, user interaction and control behavior models in the PIM level. Also, our method based on a simple source method which can be produced by non-technical stakeholders who do not know about computer modeling. As a result, our proposal will decrease the effort, time and the construction cost of e-business projects as all transformations is automated. Beside, our proposal allows increasing the odds of being more competitive in the software industry, then satisfied the end-users demands. Thus, basing on our generated IFML model; will facilitate the construction and the generation of different models for other lower levels such as PSM for web applications, mobile applications and desktop applications, which is one of MDA experiments.

Conclusion and Future Work

Our challenge in this paper was to propose a disciplined method for e-business information systems to follow the software industry growth and end-users demands. To do so, we transform the business value model automatically in the CIM level to the IFML model at the PIM level. The E3value serves to model the business value model, whereas, the IFML model presents the front-end applications content, interface composition, user interaction and control behavior models. For that, we started by defining the source and the target meta-models. Then, we specified the constraints during the source model construction. Besides, we explained the set of transformation rules implemented via ATL transformation language. After that, we deal with a case study to illustrate our proposed method. In the end, we analyzed and discussed all of our method obtained results.

In our future work, we will continue our e-business Information System construction, dealing with our generated IFML model by proposing a disciplined method that generates automatically new models in the lowest MDA level, plus to obtain the code for a different kind of application such as web applications, mobile applications and desktop applications.

Author's Contributions

Nassim Kharmoum: Worked on most of the parts, introduction, method idea, implemented the method, proposed and implemented the ATL transformations rules, designed the case study and participated in the analysis and the discussion of the results.

Soumia Ziti: Engaged in the literature review, clarified abstraction and planning, proposed the related work and participated in the analysis and discussion of the results.

Yassine Rhazali: Participated in the ATL transformation rules implementation, proposed the evaluation criteria, verified the proposed method and validated all the case study results.

Fouzia Omary: Proposed the research methodology and reviewed the final version.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all the authors have read and approved the manuscript and there are no ethical issues involved.

References

- Acerbis, R., A. Bongio, M. Brambilla and S. Butti, 2007. Webratio 5: An eclipse-based case tool for engineering web applications. Proceedings of the International Conference on Web Engineering, (CWE' 07), Springer, pp: 501-505.
- Brambilla, M. and P. Fraternali, 2014. Interaction flow modeling language: Model-driven UI engineering of web and mobile apps with IFML. Morgan Kaufmann.
- Brambilla, M. and P. Fraternali, 2011. Implementing the semantics of BPMN through model-driven web application generation. Proceedings of the International Workshop on Business Process Modeling Notation, (PMN' 11), Springer, pp: 124-129.
- Ceri, S., P. Fraternali and A. Bongio, 2000. Web modeling language (WeBML): A modeling language for designing web sites. *Comput. Netw.*, 33: 137-157.
DOI: 10.1016/S1389-1286(00)00040-2
- Cetinkaya, D. and A. Verbraeck, 2011. Metamodeling and model transformations in modeling and simulation. Proceedings of the Winter Simulation Conference, (WSC' 11), pp: 3048-3058.
- Colomb, R., K. Raymond, L. Hart, P. Emery and C. Welty *et al.*, 2006. The object management group ontology definition metaModel. Proceedings of the Ontologies for Software Engineering and Software Technology, (EST' 06), Springer, pp: 217-247.
- De Castro, V., E. Marcos and J.M. Vara, 2011. Applying CIM-to-PIM model transformations for the service-oriented development of information systems. *Inform. Software Technol.*, 53: 87-105.
DOI: 10.1016/j.infsof.2010.09.002
- Garrigo's, I., J.N. Mazon, N. Koch and M. Escalona, 2012. Web and requirements engineering. *IET Software*, 6: 83-84.
DOI: 10.1049/iet-sen.2012.0044
- Gordijn, J. and H. Akkermans, 2018. Value Webs: understanding e-business innovation. *Value Eng.*
- Gordijn, J. and J. Akkermans, 2003. Value-based requirements engineering: Exploring innovative ecommerce ideas. *Requirements Eng.*, 8: 114-134.
DOI: 10.1007/s00766-003-0169-x
- Gordijn, J., E. Yu and V.D.B. Raadt, 2006. Eservice design using i* and e/sup 3/value modeling. *IEEE Software*, 23: 26-33.
DOI: 10.1109/MS.2006.71
- Gordijn, J., H. Akkermans and V.J. Van, 2001. Designing and evaluating e-business models. *IEEE Intell. Syst.*, 16:11-17.
DOI: 10.1109/5254.941353
- Gotti, S. and S. Mbarki, 2016. Toward IFVM virtual machine: A model driven IFML interpretation. ICSoft-EA.
- Grigera, J., J.M. Rivero, E.R. Luna, F. Giacosa and G. Rossi, 2012. From requirements to web applications in an agile model-driven approach. Proceedings of the International Conference on Web Engineering, (CWE' 12), Springer, pp: 200-214.
- Hamdani, M., W.H. Butt, M.W. Anwar and F. Azam, 2018. A systematic literature review on Interaction Flow Modeling Language (IFML). Proceedings of the 2nd International Conference on Management Engineering, Software Engineering and Service Sciences, (ESS' 18), ACM, pp: 134-138.
- Jouault, F., F. Allilaire, J. Bézivin, I. Kurtev and P. Valduriez, 2006. ATL: A QVT-like transformation language. Proceedings of the Companion to the 21st ACM SIGPLAN Symposium on Object-Oriented Programming Systems, Languages and Applications, (SLA' 16), ACM, pp: 719-720.

- Kharmoum, N., K. Elbouchti, S. Ziti and F. Omary, 2019. Descriptive analysis of business value models' transformation in MDA approach. Proceedings of the International Conference Scientific Days in Applied Sciences, (DAS' 19), Larache, Morocco.
- Kharmoum, N., S. Ziti and F. Omary, 2016. An analytical study of the CIM to PIM transformation in MDA. Proceedings of the International Workshop on Computing Sciences (WCS'16), Kenitra, Morocco, pp: 14-19.
- Kharmoum, N., S. Ziti, Y. Rhazali, K. Elbouchti and W.A. Rhalem, 2018. Analytical study of requirements models construction and their transformations in MDA approach. Proceedings of the 5th Edition of the JDSIRT Conference on Information Systems, Networks and Telecommunications, (SNT' 18), Meknes, Morocco.
- Krasner, G.E. and S.T. Pope, 1988. A description of the model-view-controller user interface paradigm in the smalltalk-80 system. *J. Object Oriented Programm.*, 1: 26-49.
- Kriouile, A., T. Gadi and Y. Balouki, 2013. Cim to pim transformation: A criteria based evaluation. *Int. J. Comput. Technol. Applicat.*, 4: 616-616.
- Laaz, N., K. Wakil, S. Mbarki and D.N. Jawawi, 2018. Comparative analysis of interaction flow modeling language tools. *J. Comput. Sci.*, 14: 1267-1278.
DOI: 10.3844/jcssp.2018.1267.1278
- Luna, E.R., G. Rossi and I. Garrigós, 2011. Webspec: A visual language for specifying interaction and navigation requirements in web applications. *Requirement Eng.*, 16: 297-297.
DOI: 10.1007/s00766-011-0124-1
- Maatougui, E., C. Bouanaka and N. Zeghib, 2016. Towards a meta-model for quality-aware self-adaptive systems design. Proceedings of the ModComp@ MoDELS, (MCM' 16), pp: 12-18.
- McCarthy, W.E., 1982. The REA accounting model: A generalized framework for accounting systems in a shared data environment. *Account. Rev.*, 3: 554-578.
- OMG-BPMN, 2011. Business Process Model and Notation version 2.0. OMG.
- OMG-IFML, 2015. The Interaction Flow Modeling Language Specification version 1.0. OMG.
- OMG-MDA, 2014. MDA Guide version 2.0. OMG.
- OMG-MOF, 2015. Meta Object Facility version 2.5. OMG.
- OMG-OCL, 2014. Object Constraint Language version 2.4. OMG.
- OMG-UML, 2017. Unified Modeling Language version 2.5.1. OMG.
- OMG-XMI, 2015. XML Metadata Interchange version 2.5.1. OMG.
- Osterwalder, A. and Y. Pigneur, 2002. An ebusiness model ontology for modeling e-business. Proceedings of the 15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy, Jun. 17-19, Bled, Slovenia.
- Pijpers, V. and J. Gordijn, 2007. Bridging business value models and process models in aviation value webs via possession rights. Proceedings of the International Conference on System Sciences, HICSS 40th Annual Hawaii, Jan. 3-6, IEEE Xplore Press, Waikoloa, HI, USA, pp: 175-175.
DOI: 10.1109/HICSS.2007.111
- Rhazali, Y., Y. Hadi and A. Mouloudi, 2015. A methodology of model transformation in MDA: From CIM to PIM. *Int. Rev. Comput. Software*, 10: 1186-1201.
DOI: 10.15866/irecos.v10i12.8088
- Rhazali, Y., Y. Hadi and A. Mouloudi, 2016a. A model transformation in MDA from CIM to PIM represented by web models through SoAML and IFML. Proceedings of the 4th IEEE International Colloquium on Information Science and Technology, Oct. 24-26, IEEE Xplore Press, Tangier, Morocco pp: 116-121.
DOI: 10.1109/CIST.2016.7805027
- Rhazali, Y., Y. Hadi and A. Mouloudi, 2016b. Model transformation with ATL into MDA from CIM to PIM structured through mvc. *Procedia Comput. Sci.*, 83: 1096-1101.
DOI: 10.1016/j.procs.2016.04.229
- Rhazali, Y., Y. Hadi, I. Chana, M. Lahmer and A. Rhattoy, 2018. A model transformation in model driven architecture from business model to web model. *Int. J. Comput. Sci.*
- Rodríguez, A., I.G.R. De Guzmán, E. Fernández-Medina and M. Piattini, 2010. Semi-formal transformation of secure business processes into analysis class and use case models: An MDA approach. *Inform. Software Technol.*, 52: 945-971.
DOI: 10.1016/j.infsof.2010.03.015

Schuster, R. and T. Motal, 2009. From E3value to rea: Modeling multi-party e-business collaborations. Proceedings of the IEEE Conference on Commerce and Enterprise Computing, Jul. 20-23, IEEE Xplore Press, Vienna, Austria, pp: 202-208. DOI: 10.1109/CEC.2009.58

Yue, T., L.C. Briand and Y. Labiche, 2011. A systematic review of transformation approaches between user requirements and analysis models. Requirement Eng., 16: 75-99. DOI: 10.1007/s00766-010-0111-y