Extensibility Interaction Flow Modeling Language Metamodels to Develop New Web Application Concerns

Karzan Wakil
Sulaimani Polytechnic University-Iraq
University of Human Development-Iraq
University Technology Malaysia-Malaysia
karzanwakil@gmail.com

Dayang N.A. Jawawi
Software Engineering Department
Faculty of Computing, University Technology Malaysia
Skudai, 81310, Johor, Malaysia
dayang@utm.my

Abstract: *Web engineering is a systematic approach to develop web applications, and numerous web engineering methods have been proposed. These methods were extended through defining new models by using different mechanisms to capture the web application concepts. Due to the complexity rising of web applications, the web engineering methods cannot provide web solutions anymore. Even though Interaction Flow Modeling Language (IFML) is recently proposed as a new method for developing web applications, it has limitations. Therefore these methods need to be improved. In this paper, we present the ability of IFML extensibility to support new concerns from web applications. Moreover, we extend IFML through UML mechanisms to support new concerns from the context to the user interface. The new IFML solves the lack of context web application through defining a new model and becomes a new direction to develop concerns modern web applications.*

Keywords: Web Engineering, IFML, Extensibility, Metamodel.

1. INTRODUCTION

The model-driven development (MDD) approach of software development utilizes three key elements in the development process. These are; models, model transformation, and metamodels. In the recent past, the world has witnessed an evolution in most of the model-driven approaches to web engineering owing to the emerging challenges in web systems design. These challenges are attributable to new requirements in the web domain together with changes in implementation technologies. Here, evolution refers to the adoption of new models and processes that take into consideration new concerns and aspects. Such changes in a methodology are a risky and error-prone process.

Over the years, web hypermedia applications have improved in scope, becoming some of the best approaches for ensuring proper information access to web users. However, in the development of these applications, the different hypermedia modeling activities face several design issues [1]. A large number of researchers have experienced design issues when such as modeling complex business processes, navigation access structures, activities and transactional workflows, user dependent processes [2-3]. To deal with these design issues, a number of hypermedia methods that utilize different modeling concepts and alternative

design features are proposed. They include UML-based Web Engineering (UWE) [4-7], Object-oriented Hypermedia Design Method (OOHDM) [8], Object Oriented Hypermedia (OOH) [6-7], and the Website Design Method (WDM) [9]. Lastly, Interaction Flow Modeling Language (IFML) [10] is considered a standard approach to solving current issues associated with the development of web applications. For example, it helps solve issues relating to mobile applications and the interaction user-interface.

Many researchers made attempts to define "common" meta-models, yielding very interesting results. The Web Engineering Interoperability (WEI) initiative was the most thorough systematic approach applied by these researchers with the aim of providing an easy exchange of models and addressing the new concerns that had been incorporated into the existing methods [11]. The best practices of defining terms are utilized to define the IFML metamodel. These include reuse, abstraction, extensibility, and modularization. The IFML metamodel has various characteristics. For instance, it works by reusing the basic data forms derived from the UML metamodel. It forms IFML metaclasses by specializing several UML databases, and assumes that either a UML class diagram or an appropriate notation is used to represent a domain model [10].

However IFML is a standard method and designed after ten year experience of WebML, but it needs more extensions and improvements especially for new concern and features of web applications. The problem is that most previous works have extended IFML to support user interface. We extend IFML metamodels furthermore to support new concerns from content to user interface through UML mechanisms.

The paper is organized as follows: Section 2 explains the related work for the web engineering metamodels especially IFML metamodel. Section 3 analyzes of IFML metamodels and capability IFML for extensibility. Section 4 defines a framework to further extend IFML. In section 5 we implement framework with case study and evaluation the framework. In Section 6 we present some concluding remarks and points to future works.

2. RELATED WORK

Several existing web engineering methods that address the systematic development of web applications through extend models and metamodels. Based on[12], 5.5%

publication on web engineering worked on metamodels. In this section, we reviewed some related work about extension metamodels in web engineering methods, extension mechanisms, IFML metamodels, and IFML extension metamodels.

Today, many of the available works have defined new models and metamodels using the extension technique of UML. The definitions are provided for a variety of purposes. The UML extension provided in [13] enables modeling system audits by supporting an event-based modeling approach. This extension makes it possible to integrate different audit properties for a wide range of UML models. Reference [14] documents the details of a rood traffic management system that is based on stereotypes, while [15] offers a Use Case metamodel. The proposed metamodel models the structural view of the use cases as well as the behavioral view. A new element notation is proposed in another work [16]. It is defined by an expanded form of the unified modeling language through the creation of new UML concepts. An extension of the UML metaclasses "Association" and "Class" yields some stereotypes presented in this work.

Source [17] utilizes scientific techniques to suggest a new set of notations that is an extension of the UML state charts notation. An industrial survey was conducted online to assess the perceptions of the proposed notational set with regards to its coverage of state-based security aspects and its semantic transparency. Reference [18] explored the process of extending UML in Web information systems on the basis of a context-based navigation framework. The framework encompasses various navigation phases and models including analysis, design, and realization. An android application used in modeling specifically described by the extended UML metamodel. This extended UML metamodel also helps improve communication between developers and other parties because it adheres to the concepts and notation of the UML metamodel [19]. In addition, UML profile is presented as an aspect-oriented form of modeling founded on aspect language [20]. Earlier on, the UWE navigation model was extended in support of new web application models, while the UML extension technique was applied in defining new elements. Comparisons were also made among capability metamodels [21-23].

Laaz and Mbarki provide a new approach MDE that assembles two essential abstract specifications, described by the OMG, for deriving UIs Rich Internet Applications (RIAs). The proposed approach assumes that it is possible to induce UIs for RIA from ontology and IFML. IFML allows for an abstract representation of the organization of user interfaces. It also shows how different elements interact as demonstrated in Figure 1 [24]. Further, other researchers [24] provide an alternative approach for the generation of RIA. This model-driven approach uses the OMG standard IFML. They also define a RIA metamodel that adheres to the Model View Presenter pattern, and develop a

transformation engine that automatically generates the output model.

Figure 1 IFML extension [24]

In [25] the authors designed and implemented a customized MDUID process that integrates Graphic User Interface (GUI) patterns. Then presented their GUI pattern catalogue and its formalization based on the general UI language IFML. The feasibility of their approach given by a tool support which extends the current IFML editor by combined GUI patterns. The implementation of the customized MDUID process and the practical usage of the tool support were presented in the context of creating RIAs. Gotti and Mbarki defined an IFVM Virtual Machine for the effective performance of IFML models, which can be performed on many platforms due to java virtual machine advantages [26-27].

M. Brambilla et al. suggested an extended form of the IFML language used in modeling. The extension was specially created for mobile applications. The researchers describe their experiences during the implementation process, which includes the creation of automatic code generators applicable in cross-platform mobile applications. The code generators were built on CSS, HTML5, and JavaScript, modified to suit the Apache Cordova framework [28].

3. ANALYZING EXTENSIBILITY FOR IFML METAMODELS

IFML [10] has been completed to improve the platform-independent definition of GUI with applications installed on systems like desktop, laptops, smart phones, tablets, and PDAs. The key focus is on the application's behavior and structure as observed by the end user. In this section, we analyze IFML metamodels in the process development mobile applications and web applications. Then we extract previous extensions.

3.1 IFML-Artifacts

IFML officially defined by OMG, the technical artifacts explained by [10] we explained most important artifacts below:

- IFML metamodel, specify the structure and associations between the elements;

- The IFML is UML profile, meaning used UML concepts for designing and extending class diagram, state machine and elements.
- The IFML visual syntax, it has graphical notations for representing elements and models.
- The IFML model entertainment in installments and switch format, for tool portability.

Altogether, these artifacts compose the IFML language specification. Each of them is specified according to the OMG standards:
- The metamodel is defined through the MOF metamodeling language.
- The UML profile is defined consistent with UML 2.4 profile policy.
- The visual syntax is defined through Diagram Definition (DD) and Diagram Interchange (DI) OMG-standards.
- The model serialization and exchange format is defined founded on XMI.

3.2 IFML Metamodels

Definition of IFML metamodel is completed as the best approach of the language description. Incorporate abstraction, modularization, recycle as a best extensibility. It is composed of three packages that are; "Core Package," "Extension package," and "Data type Package." The first package is core package that contains all concepts for creating interaction infrastructure of the method such as "Interaction Flow Elements, Interaction Flows, and Parameters.", The second package is Extension package when extended the ideas that defined by Core Package, the third package is Data- Types package, it has data types that determined by metamodel of UML, and specializes some UML metaclasses as the origin for IFML meta-classes, and presume that the IFML domain model is represented in UML [10], as explained in Figure 2.

Figure 2 IFML metamodels Packages[10]

IFML model is known as the top-level component of the other model components. It involves a domain model, an Interaction Flow Model, as well as View Points. Interaction Flow Model offers the application view of the user, by reference to the "Interaction Flow Model Elements" sets, together defining a wholly functional portion of the system.

3.3 Extensibility IFML Metamodels

In this section, we have been a discussion about extension metamodels in web engineering methods, extensibility IFML metamodels; then we discuss extension mechanisms that used for IFML and UML metamodels. Moreover, at the end of this section, we analyze existing work about IFML extension from previous work.

For addressing the new concern in web engineering methods three ways defined; 1- combining current original methods with additional models, 2- merging two or three methods, 3- defining new models or new method through extension or generating [29]. Furthermore, UML allows extending package in meta model[30]. In another hand UML extensibility mechanism allows designers customize and extend the UML by adding new building blocks that consist of stereotypes, tagged values, and constraints[31].

Extension classified for extension mechanism and extension purpose by [32]. The aspect "extension purpose" covers the objective that is related to the aimed extension and reflects the purpose that needs to be fulfilled. An extension mechanism is understood as either an explicit mechanism of an EML for the extension of this language or a more general approach for extension. We have evolved the following mechanisms based on the review of both the literature [33-34] and existing EML specifications as shown in Figure 3 and Figure 4.
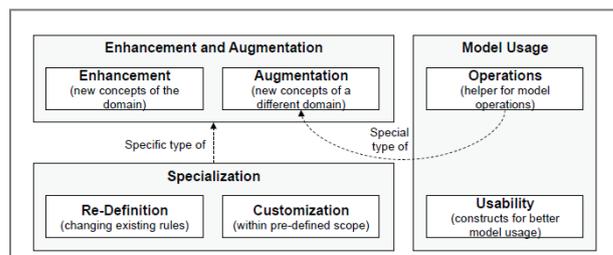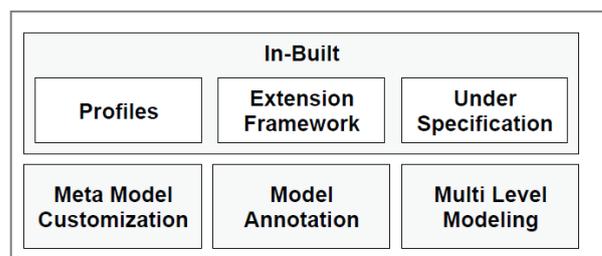

Figure 3 Consolidated Extension Purpose [32]

Figure 4 Consolidated Extension Mechanisms [32]

3.4 IFML Extension from Existing Work

IFML as new method from 2014 can support most interaction aspects but also need to extension, because web applications and mobile applications growth day by

day, the following table we collected all papers that extended IFML models and metamodels.

Table 1: IFML extension from Existing Work

No	Ref.	Mechanism	Aspect	Contribution
1	[24]	Defining UML metamodel and UML meta class for modify Core package	For Defining GUI Ontology	Represent complete RIA interfaces.
2	[35]	WebRatio	extensions feature primitives	Web and Mobile applications
3	[28, 36]	HTML5, CSS and JavaScript for platform mobile applications	to mobile application development	the development of automatic code generators
4	[37]	Integration with WSDM	Smantic web	To develop web semantic design

According to the philosophy of the language, not all possible extensions are allowed. Valid extensions should refine or adapt the core concepts to specific cases, specializing their semantics without altering them. The IFML specification explicitly mentions that only the following concepts (and their specializations) can be extended while retaining compliance with the standard:

- View Container (for defining specific screens or interface containers),
- View Component (for describing specific widgets or controls),
- View Component Part (for specifying particular properties of existing or new View Components),
- Event (for covering platform-specific events),
- Domain Concept and Feature Concept (for covering additional content sources), and Behavior Concept and Behavioral Feature Concept (for covering integration with additional behavioral models or modeling languages).

Extensions of other elements are disallowed by the standard. Any other extended concept will be considered proprietary and outside the IFML notation

IFML is UML profile, IFML uses UML extension mechanism, and four common mechanisms for extension UML are: Specifications, Common Divisions, Adornments, and Extensibility mechanisms.

4. METHODOLOGY

In this section, we define a new framework for extending IFML metamodels to support a new concern for developing web applications. Preparing our new model focused on Enhancement and augmentation as Extension purpose, and focused on Metamodel Customization as Extension mechanism. Figure 5 explains our framework

that consists of five steps; the step1 is IFML before extension when explained in previous sections, step2 showed IFML metamodel is three packages that are; Core package, Extension Packages and Data type packages, in step3 we explained the metamodel packages, this packages analyzed in section 3, the detail of packages very important because guide us how we can extend it, in the step4 we use UML extension mechanism for extending IFML to support the new concerns of web applications, final step is new IFML after extension.
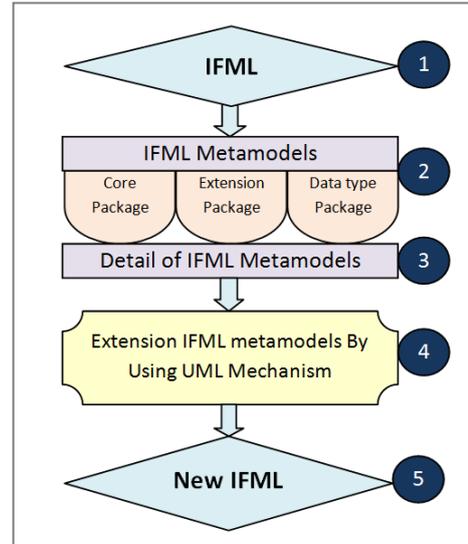

Figure 5 A New Framework for Extension IFML metamodels

Based on process development IFML to cover lifecycle [38], our framework extends all steps, in the extended metamodels from content to user interface, we extend Interaction Flow Elements, Interaction Flow, and Parameters. Then by using UML mechanism, we define the new elements. This demonstrated that the added features in the metamodels lead to more usability in concluding web applications. Extension mechanism helps the researchers to improve the web engineering methods for the web applications development. In the following, we present the steps of defining new elements from a mechanism that prepared by [39] as shown in Figure 6.
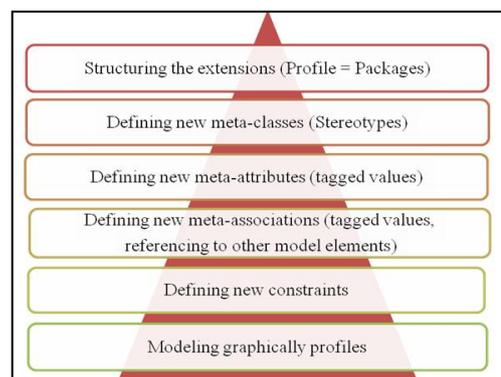

Figure 6 UML Extension Mechanism [40]

5. DESIGN CASE STUDY

In this section, we extend payment execution that designed by IFML for RIA as new concept, however IFML can support RIA, but here for proving our framework we extend IFML model front-end, the payment execution previously we designed in [38]. As shown in Figure 7.
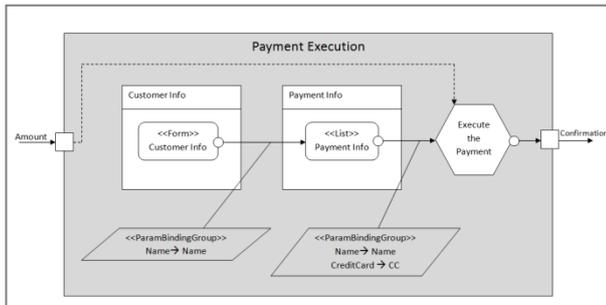

Figure 7 Inner Process of the Module Payment Execution [38]

RIA features focused on Client (C) and Server (S), in Figure 8 we extend View component, View container, and parameters, and moreover we define an event for checking client or server side.
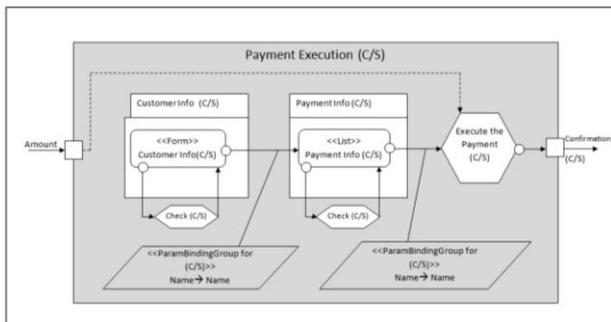
Figure 8 Inner Process of the Module Payment Execution (Extended for RIA)

After extension we define RIA element based on UML extension mechanism as explained the steps:
1. Name : RIA
2. Stereotype: ClientServer
3. Attribute {Title ClintServer=" ClintServer", Number Elements=n, name of elements="string"}
4. List of attributes (Name, Number, Elements)
5. Context: ClientServer inv: name.size()<=40 inv: number >= 0 && number <= 100 inv: elements.size()<=4.
6. Graphical Icon is RIA .

Our extension successfully completed for supporting new cancers that is RIA features. The researcher can follow same steps and mechanism to extend IFML for the new concerns. We can evaluate our framework as a success framework for extending IFML metamodels for supporting new concerns.

6. CONCLUSION AND FUTURE WORK

In this work, we proposed a new framework for extending IFML metamodels, and then we extend IFML through UML mechanisms to support new concerns from content to user interface of the modern web application. The extension IFML solves the lack of context, and become to a new direction to develop modern web application features. Our future plan is to improve our framework for defining an adaptive model for IFML to develop multi web applications.

7. REFERENCES

[1] A. H. Jantan, *et al.*, "Design Processes For Web-Based Hypermedia Engineering," 2007.

[2] H. A. Schmid and G. Rossi, "Modeling and designing processes in e-commerce applications," *IEEE Internet Computing,* vol. 8, pp. 19-27, 2004.

[3] S. S. Selmi, *et al.*, "Toward a comprehension view of web engineering," in *International Conference on Web Engineering*, 2005, pp. 19-29.

[4] N. Koch and A. Kraus, "The expressive power of uml-based web engineering," in *Second International Workshop on Web-oriented Software Technology (IWWOST02)*, 2002.

[5] P. Dolog and M. Bieliková, "Hypermedia systems modelling framework," in *Computing and Informatics*, 2002.

[6] J. Gómez and C. Cachero, "OO-H Method: extending UML to model web interfaces," *Information modeling for internet applications,* pp. 144-173, 2003.

[7] N. Koch, *et al.*, "Modeling web business processes with OO-H and UWE," in *Third International Workshop on Web-oriented Software Technology (IWWOST03). Schwabe, D., Pastor, O., Rossi, G., Olsina, L.(eds.)*, 2003, pp. 27-50.

[8] D. Schwabe and G. Rossi, "Developing hypermedia applications using OOHDM," in *Workshop on Hypermedia Development Process, Methods and Models, Hypertext*, 1998.

[9] O. De Troyer and S. Casteleyn, "Modeling complex processes for web applications using wsdm," in *Proceedings of the 3rd International Workshop on Web-Oriented Software Technologies*, 2003, pp. 27-50.

[10] M. Brambilla and P. Fraternali, *Interaction flow modeling language: Model-driven UI engineering of web and mobile apps with IFML*: Morgan Kaufmann, 2014.

[11] N. Moreno, *et al.*, "An Overview Of Model-Driven Web Engineering and the Mda," ed: Springer, 2008.

[12] K. Wakil and D. N. Jawawi, "Model driven web engineering: A systematic mapping study," *e-Informatica Software Engineering Journal,* vol. 9, pp. 107--142, 2015.

[13] B. Hoisl and M. Strembeck, "A UML extension for the model-driven specification of audit rules," in *International Conference on Advanced Information Systems Engineering*, 2012, pp. 16-30.

[14] M. Singh and R. Saxena, "An Application of UML for Road Traffic Management System by Implementing Extensive Mechanism: Stereotypes," *International Journal of Computer Science Issues (IJCSI),* vol. 11, p. 83, 2014.

[15] M. Misbhauddin and M. Alshayeb, "Extending the UML use case metamodel with behavioral information to facilitate model analysis and interchange," *Software & Systems Modeling,* vol. 14, pp. 813-838, 2015.

[16] M. S. Benselim and H. Seridi-Bouchelaghem, "Extended UML for the development of context-aware applications," in *International Conference on Networked Digital Technologies*, 2012, pp. 33-43.

[17] M. El-Attar*, et al.*, "Extending the UML statecharts notation to model security aspects," *IEEE Transactions on Software Engineering,* vol. 41, pp. 661-690, 2015.

[18] J. Hong*, et al.*, "Extending UML for a context-based navigation modeling framework of web information systems," in *International Conference on Software Engineering Research and Applications*, 2004, pp. 108-122.

[19] M. Ko*, et al.*, "Extending UML Meta-model for Android Application," in *Computer and Information Science (ICIS), 2012 IEEE/ACIS 11th International Conference on*, 2012, pp. 669-674.

[20] M. Chibani*, et al.*, "Towards a UML Meta Model Extension for Aspect-Oriented Modeling," 2013.

[21] K. Wakil*, et al.*, "Enhancement of UWE navigation model: Homepage development case study," *International Journal of Software Engineering & Its Applications,* vol. 8, 2014.

[22] K. Wakil and D. N. Jawawi, "Metamodels Evaluation Of Web Engineering Methodologies To Develop Web Applications," *International Journal of Software Engineering & Applications,* vol. 5, p. 47, 2014.

[23] K. Wakil*, et al.*, "A Comparison of Navigation Model between UWE and WebML: Homepage Development Case Study," *International Journal of Information and Education Technology,* vol. 5, p. 650, 2015.

[24] N. Laaz and S. Mbarki, "Combining Ontologies and IFML Models Regarding the GUIs of Rich Internet Applications," in *International Conference on Artificial Intelligence: Methodology, Systems, and Applications*, 2016, pp. 226-236.

[25] E. Yigitbas*, et al.*, "Model-driven UI Development integrating HCI Patterns," *Large-scale and Model-based Interactive Systems,* p. 42, 2015.

[26] S. Gotti and S. Mbarki, "Toward IFVM Virtual Machine: A Model Driven IFML Interpretation," 2016.

[27] E. Yigitbas and S. Sauer, "Customized UI Development Through Context-Sensitive GUI Patterns," *Mensch und Computer 2016–Workshopband,* 2016.

[28] M. Brambilla*, et al.*, "Extending the interaction flow modeling language (IFML) for model driven development of mobile applications front end," in *International Conference on Mobile Web and Information Systems*, 2014, pp. 176-191.

[29] N. Moreno*, et al.*, "Addressing new concerns in model-driven web engineering approaches," in *International Conference on Web Information Systems Engineering*, 2008, pp. 426-442.

[30] D. D'Souza*, et al.*, "First-class extensibility for UML—Packaging of profiles, stereotypes, patterns," in *International Conference on the Unified Modeling Language*, 1999, pp. 265-277.

[31] J. Farhad, "The UML Extension Mechanisms," *Department of Computer Science, University College London,* 2002.

[32] R. Braun, "Towards the state of the art of extending enterprise modeling languages," in *Model-Driven Engineering and Software Development (MODELSWARD), 2015 3rd International Conference on*, 2015, pp. 1-9.

[33] C. Atkinson*, et al.*, "Modeling language extension in the enterprise systems domain," in *Enterprise Distributed Object Computing Conference (EDOC), 2013 17th IEEE International*, 2013, pp. 49-58.

[34] R. Braun and W. Esswein, "Classification of domain-specific bpmn extensions," in *IFIP Working Conference on The Practice of Enterprise Modeling*, 2014, pp. 42-57.

[35] R. Acerbis*, et al.*, "Model-Driven Development Based on OMG's IFML with WebRatio Web and Mobile Platform," in *International Conference on Web Engineering*, 2015, pp. 605-608.

[36] R. Acerbis*, et al.*, "Model-Driven Development of Cross-Platform Mobile Applications with Web Ratio and IFML," in *Mobile Software Engineering and Systems (MOBILESoft), 2015 2nd ACM International Conference on*, 2015, pp. 170-171.

[37] J. Blanckaert, "Integrating the Interaction Flow Modelling Language (IFML) into the Web Semantics Design Method (WSDM)," 2015.

[38] K. Wakil and D. N. Jawawi, "Analyzing Interaction Flow Modeling Language in Web Development Lifecycle," *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS,* vol. 8, pp. 286-293, 2017.

[39] P. Desfray, "UML Profiles versus Metamodel extensions: An ongoing debate," in *OMG's UML Workshops: UML in the. com Enterprise: Modeling CORBA, Components, XML/XMI and Metadata Workshop*, 2000, pp. 6-9.

[40] D. Philippe, "UML Profiles versus Metamodel extensions: An ongoing debate," in *OMG's UML Workshops: UML in the. com Enterprise: Modeling CORBA, Components, XML/XMI and Metadata Workshop*, 2000, pp. 6-9.