

# Meta-model to Support End-User Development of Web Based Business Information Systems

Buddhima De Silva and Athula Ginige

University of Western Sydney, Locked Bag 1797, Penrith South DC, 1719, NSW, Australia  
bdesilva@scm.uws.edu.au, a.ginige@uws.edu.au

**Abstract.** End-user development is proposed as a solution to the issues business people have when getting web applications developed. In this paper, we have presented a meta-model for web based information systems to support End-user Development. End-users can actively participate in web application development using tools to populate and instantiate the meta-model. The meta-model we created is based on three abstraction levels: Shell, Application, and Function. At Shell Level, we model aspects common to all business web applications such as navigation and access control. At Application Level, we model aspects common to specific web applications such as workflows. At Function Level, we model requirements specific to the identified use cases. Inheritance and Overriding properties of the meta-model provide a balance between ease and flexibility when developing business information systems. The key aspect that underpinned this research work is the view- “software is a medium to capture knowledge rather than a product”. Meta-model will help end-users to participate in web application development activities.

## 1 Introduction

The characteristics of the web such as ubiquity and simplicity make it a suitable platform to disseminate information and automate business processes. AeIMS research group at University of Western Sydney has been working with businesses in Western Sydney region to investigate how Information and Communication Technologies (ICT) can be used to enhance their business processes [1-3]. In this work, we have identified many issues Business users have to overcome when trying to implement web applications [3]. These issues vary from not being able to get web applications developed to meet needs of the business in a timely manner to development projects running over budget. The development approach should also reduce the gap between what the users actually wanted and what is being implemented in terms of functionality [4]. Researchers have proposed to empower end-users in web application development as a solution to these issues [3, 5-8].

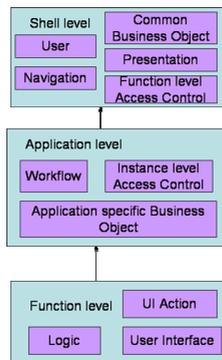
There are different approaches to empower end-users. One approach is to provide end-users with tools to develop any kind of a web application such as informational, search directory and directory look up, workflow and collaboration, e-commerce and web portal, etc. However, such a tool will become very complex because it has to cover a variety of features [3]. The other approach is to develop different tools to

support different types of web applications. Our approach falls in to the second category. We have analysed many business web applications. From this analysis, we identified, the business users need support to store, process and report information to effectively carry out various business processes. The meta-model for this class of business web applications at conceptual level is form being routed based on rules. Example instance of that meta-model is a leave processing system where employees can apply for leave. Meta-model elements are high level abstract concepts such as user, role, form User Interface, business object etc. Being able to develop web applications using these high level concepts help to address the critical concerns of end-user developer relating to details of creating databases, creating web forms, user authentication, etc..

The major finding presented in this paper is the hierarchical meta-model. In section 2 we present the hierarchical meta-model and section 3 discuss the properties of meta-model. In section 4 we review the related work and section 5 conclude the paper.

## 2 Hierarchical Meta-model of Business Web Applications

As mentioned previously we view Web based business applications as an instance of a meta-model. Theoretically by creating a meta-model and developing tools to populate the instance values we can generate business applications. In practice creating a meta-model to support end-user development is not easy because of the complexities of business applications. To manage the complexity we developed the meta-model at 3 levels of hierarchical abstraction called Shell, Application and Function as shown in figure 1.



**Fig. 1.** Hierarchy of Abstraction Levels of the Meta-Model

- Shell Level: The aspects common to all web applications such as user, navigation are modeled at shell level.
- Application Level: The aspects specific to a web application such as workflow, instance level access control are modeled at this level.
- Function level: The function specific aspects which are required to implement the function are modeled at function level.

### 2.1 Shell Level

We analysed many business applications to identify common functionality required in most of the web applications. These common functionalities are modeled at the shell level. User model, Access Control model, Navigation Model and Business Object Model are four models that we identified at this level. Shell level of meta-model is shown in figure 2.

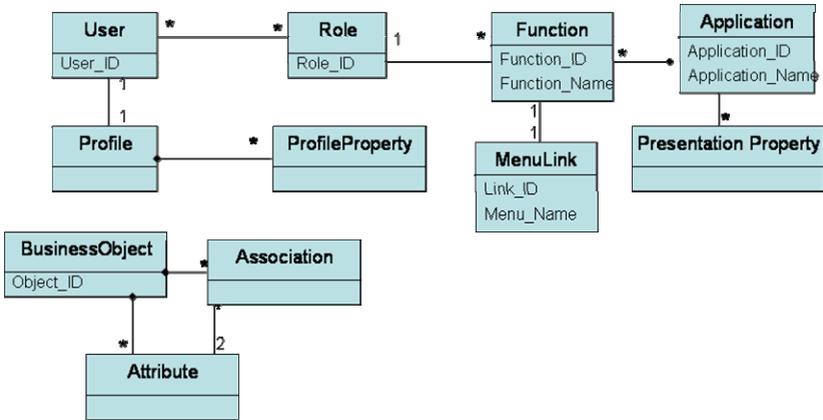


Fig. 2. Shell level of meta-model

Each user has a profile. Profile can have many properties such as name, address, e-mail address, etc. User can play many roles. A role can have one or more users. Each role has functions in an application assigned to it. Application has presentation properties. Each function has a menu link. Business Objects have many attributes and associations. Navigation model is the mechanism that authenticated users can use to access the authorised functions. We have identified two types of functions: State independent functions and State dependent functions. Once a user log in, user will be provided with a menu to access state independent functions which are available to authorized users to access based on the navigation model. On the other hand, the state dependant functions are available to users only if it is waiting for that particular user at that time. We have a menu link to access such functions in all applications at one place.

### 2.2 Application Level

As mentioned earlier an application consists of many functions. Therefore, application consists of models which support many functions. The Application level of the meta-model is shown in figure 3. Application inherits the function level access control, common business objects and navigation models from the shell level. It consists of workflow model, instance level access control model and application specific object models. Workflow model models the business rules that govern the flow of information. When the business objects are accessed through the state dependant

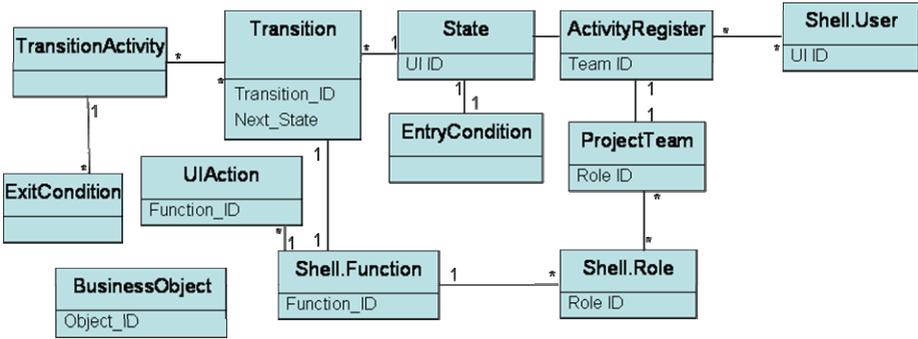


Fig. 3. Application Level of meta-model

functions there could be rules specifying who can access what instances of a Business object. By applying the instance level access rules we can identify the ‘project team’ that participates in actions in state dependant functions in the workflow.

### 2.3 Function Level

Functions are the way of performing the actions in an application. The User Interface is the mechanism users have to perform the functions.

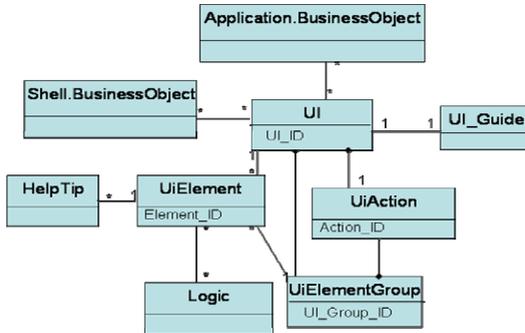


Fig. 4. Function level of meta-model

The function level meta-model is shown in figure 4. User Interface Model consists of UI guide, UIElementGroup, and UIElements, and UI Actions. UI elements can be in input mode or output mode. In a form we have UI elements in Input mode. In a report we have UI elements in output mode. UIGuide provide the guidelines to use the particular interface. At function level we model two types of business rules. One type is the business rules used to derive new object attribute values based on existing object attribute values. The other category is the validation rules applied over values of form field in a user interface.

### 3 Meta Model Properties

The hierarchical model has two properties inheritance and overriding. These two properties provide a balance between flexibility and ease of use. For example, if end-user wants he/she can develop a web application with default presentation style inherited from the shell. He/she doesn't have to bother about the presentation styles and templates at the application level. However, if an end-user wants a custom look and feel he/she can override the default presentation style provided from shell level at the application level.

### 4 Related Work

Several meta-models exist for web applications. One of them is UWE meta model [9, 10]. It is designed as an extension based on MOF 1.4. The objective of UWE meta-model is to provide a common meta-model for the web application domain, which will support all web design methodologies. UWE meta-model has similar model elements as ours. However, the main difference between the 2 meta-models is the 3 level, hierarchical abstraction that we used to support end user development. W2000 [11], a successor of HDM [12], has provided model semantics and transformation rules to achieve consistency between models. Muller et al. [13] present a model-driven design and development approach with the Netsilon tool. The tool is based on a meta-model specified with MOF 1.4 and the Xion action language. Recently another two meta models[14, 15] based on MOF and UML 2 profiles are presented for WebML design methodology with the objective of interoperability.

All these meta-models of web applications are towards the precise definition of the semantics of existing web models. Our work is complementary to existing web meta-models; in that we propose a hierarchical organization of meta-model with a different perspective- to help to effectively involve end users in development.

### 5 Conclusion

In this paper we have presented a hierarchical meta-model enabling business end-user to develop business web applications. The semantics in our meta-model helps end users to begin the development work with little knowledge in computer domain. For example to configure an application they only need the knowledge of the shell meta-model. To participate in development at application level they have to use their domain knowledge and logical thinking; thus should know the application meta-model. We limit our focus to form based web applications as these are the most required by business end users. However, in the future we are planning to expand the meta-model for development of other types of web applications. We are currently planning formal experiment in end user development based on this meta-model. This will help us to better understand the mental model of end users; thus help to refine the tools available for end users to instantiate the meta-model.

## Reference

1. Arunatileka, S., Ginige, A.: Applying Seven E's in eTransformation to Manufacturing Sector. in *eChallenges* (2004)
2. Ginige, A.: From eTransformation to eCollaboration: Issues and Solutions. In: *2nd International Conference on Information Management and Business (IMB 2006)* Sydney, Australia (2006)
3. Ginige, J., De Silva, B., Ginige, A.: Towards End User Development of Web Applications for SMEs Using a Component Based Approach. In: *Lowe, D.G., Gaedke, M. (eds.) ICWE 2005. LNCS, vol. 3579, Springer, Heidelberg* (2005)
4. Epner, M.: Poor Project Management Number-One Problem of Outsourced E-Projects, in *Research Briefs, Cutter Consortium* (2000)
5. Ginige, A., De Silva, B.: CBEADS: A framework to support Meta-Design Paradigm. In: *3rd International Conference on Universal Access in Human-Computer Interaction (UAHCI07), China. LNCS, vol. 4554, pp. 107–116, Springer, Heidelberg* (2007)
6. Costabile, M., F., et al.: A meta-design approach to End-User Development. In: *VL/HCC05* (2005)
7. Fischer, G., Giaccardi, E.: A framework for the future of end user development, in *End User Development*. In: *Lieberman, H., Paterno, F., Wulf, V. (eds.) Empowering People to flexibly Employ Advanced Information and Communication Technology, Kluwer Academic Publishers, Boston, MA* (2005)
8. Fischer, G., et al.: Meta Design: A Manifesto for End -User Development. *Communications of the ACM* 47(9), 33–37 (2004)
9. Kraus, A., Koch, N.: A Metamodel for UWE. 2003, *Ludwig-Maximilians-Universität München* (2003)
10. Koch, N., Kraus, A.: Towards a Common Metamodel for the Development of Web Applications. In: *Third international conference on Web engineering, Oviedo, Spain. Springer-Verlag, Heidelberg* (2003)
11. Luciano, B., Sebastiano, C., Luca, M.: First experiences on constraining consistency and adaptivity of W2000 models. In: *Proceedings of the 2005 ACM symposium on Applied computing. ACM Press, Santa Fe, New Mexico* (2005)
12. Garzotto, F., Paolini, P., Schwabe, D.: HDM — A Model-Based Approach to Hypertext Application Design. *ACM Transactions on Information Systems (TOIS)* 11(1), 1–26 (1993)
13. Muller, P., et al.: Platform independent Web application modeling and development with Netsilon. *Software & System Modeling* 4(4), 424–442 (2005)
14. Schauerhuber, A., Wimmer, M., Kapsammer, E.: Bridging existing Web modeling languages to model-driven engineering: a metamodel for WebML. In: *Workshop proceedings of the sixth international conference on Web engineering. ACM Press, Palo Alto, California* (2006)
15. Nathalie, M., Piero, F., Antonio, V.: A UML 2.0 profile for WebML modeling. In: *Workshop proceedings of the sixth international conference on Web engineering. ACM Press, Palo Alto, California* (2006)