

Smart Tools to Support Meta-design Paradigm for Developing Web Based Business Applications

Athula Ginige, Xufeng Liang, Makis Marmaridis, Anupama Ginige,
and Buddhima De Silva

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

Abstract. Many Small and Medium Enterprises (SMEs) tend to gradually adopt Web based business applications to enhance their business processes. To support this gradual adoption we need a framework that supports iterative development. Further processes that have been supported by web based business applications can change and evolve requiring applications to be changed accordingly. To support these needs we have extended the **Component Based E Application Development and Deployment Shell; CBEADS[®]**. We analysed many business applications and derived a meta-model. We implemented this meta-model with in CBEADS[®] and developed a set of Smart Tools to take the instance values of the meta-model and generate the web based business applications. When a new business application is required, a business analyst can create a new instance of the meta-model. To change an implemented business application the appropriate values of the meta-model instance that corresponds to the particular application can be changed.

Keywords: Meta-Design, Web Engineering, Smart Tools, rapid development, model driven software development.

1 Introduction

The AeIMS research group at the University of Western Sydney has been working with Small to Medium Enterprises (SMEs) in the Western Sydney region to investigate how Web based business applications can be used to enhance their business processes to become competitive in a global economy [1, 2]. In this research one challenge was to find a way to develop web based business applications rapidly and in a cost-effective manner [3]. Also it was necessary to have the ability to change these applications with evolving business needs [4]. The development approach should also reduce the gap between what the users actually wanted and what is being implemented in terms of functionality [5].

In a Business organisation there are many software applications to support its business processes such as ordering, inventory management, leave processing, invoicing, production planning, customer relationship management (CRM) etc. Today most of these applications tend to be web based. Some of the data used in these processes such as employee details, customer details, product or service details etc will be common to many business processes. Thus organisations that have invested in separate systems to

support their business applications are finding it hard to keep the common data stored in separate systems in sync. Unlike large organisations, SMEs have not got the money or the time to invest in obtaining a single ERP system to support its business processes as a one-off project. We have observed often in SMEs one process at a time being enhanced by the use of web technologies based on some priority criteria. Also these web applications used to enhance the business processes need to evolve when business processes change with time.

To meet the above requirements we adopted the concept that software is a medium to capture knowledge rather than a product [6]. This led us to change our thinking from looking at methodologies to develop web based business applications to developing a framework within which web based business applications can evolve. We adapted the **Component Based E-Application Development and Deployment Shell (CBEADS[®])** as our framework [3, 7]. CBEADS[®] has the ability to create new functions within its own framework so that it can evolve. These functions can be grouped together to form various business applications.

This resulted in a hierarchical model; an organisation can have a shell, the shell will have many applications, applications consist of many functions (use cases). We analysed many business applications and found that most of these can be modeled as a form being routed based on some rules and different actors can have access to different views of the underlying data objects. Thus rather than developing “the application” we developed a meta-model of the application and a set of tools within which the Business Analyst can create the applications that they want [8].

2 Meta-model for Web Based Business Applications

Our meta-model consists of three levels of hierarchical abstraction called Shell, Application and Function as shown in figure 1. The Shell provides the common

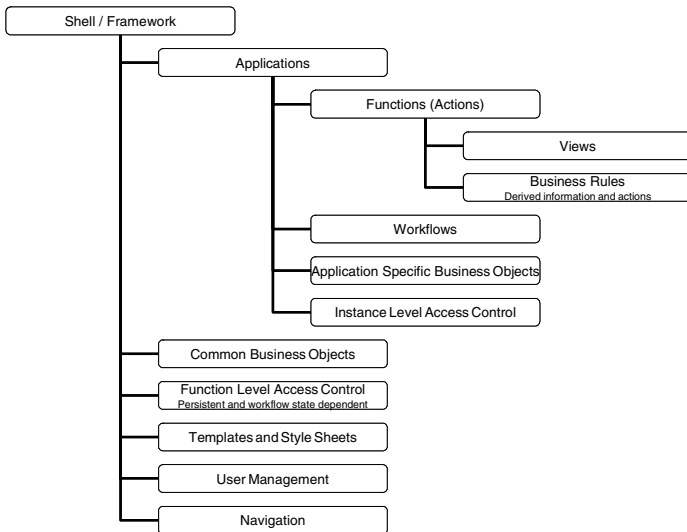


Fig. 1. Hierarchy of Abstraction Levels in Business Application Meta-Model

functionality required for any application such as user management, authentication, access control and overall navigation. The Application provides a set of functions required to perform a business process and sequencing of functions as required to create the necessary workflows. Functions provide views or user interfaces required to perform actions in a business process.

To implement this meta-model within CBEADS[®] framework we had to first develop a way to specify Business Objects, Views, Business Rules, Function level and Instance level access control, Workflows, Over all layout and, Look and feel. Then we had to create a way to generate the physical objects (i.e. databases), functions and workflows required for the application based on instance values of the meta-model. For this we developed a set of Smart Tools. We incorporated some computer domain knowledge into these tools so that a Business Analyst can use these tools to develop parts of the application without needing to develop detail code as well as databases.

3 Smart Tools

We developed an architecture for Smart Business Objects and two tools; Builder and UI Generator to create the objects and to generate the required views incorporating various business rules [9]. We developed a state machine based approach to model business processes that can evolve with changing business needs and a workflow engine to enact these business process models [10]. We identified that navigation and access control are tightly coupled and implemented these two together.

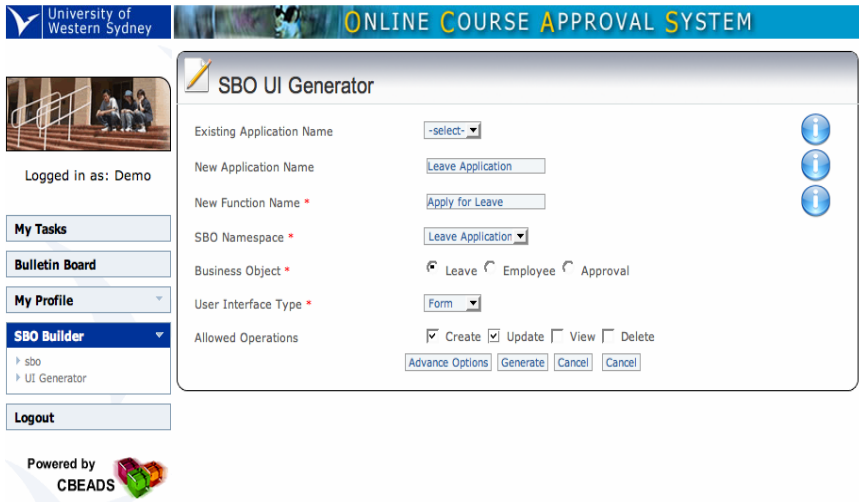


Fig. 2. Generating UIs for leave processing application

3.1 Smart Business Objects (SBO)

Using the SBO builder tool in CBEADS[®] we can specify business objects using the SBOML language [9]. Example of a business object used in a leave processing

application is shown below. The SBO builder then generates the Business Objects for the application.

SBOML Specification of the Leave Object

In leavesystem, leave has date, applicant, from (date), to (date), type (which could be sick or annual or no pay), status (which could be approve or reject), many approval (has date, approved by, comment)

UI Generator tool allows us to select the business object and necessary attributes to create different views required for the leave processing application. This tool is shown in figure 2.

3.2 Workflow Engine

Next we need to sequence the views. This we do using a state table. The workflow specification for the leave processing application is shown in Table 1.

Table 1. State Table for Leave processing workflow

Current State	Actor	Function	Buttons	Do Action	Next State
1	Employee	Apply_Leave	Submit	Email Division_Manager	2
2	Division_Manager	Approve_Leave	Approve	Email HR_Manager, Employee	3
			Reject	Email Employee	4
3	HR_Manager	Process_Leave	Processed	Email Employee, Division_Manager	4
4	HR_Manager	View_Applications	-		4

3.3 Navigation and Access Control

We have identified the need to provide 2 types of access to various functions; functions that can be accessed any time (workflow state independent access) and functions that can be accessed based on workflow state such as a manager getting a link to access approve leave function only when there is a pending leave application. We also need to manage instance level access control.

We used role based access control mechanism in CBEADS[®] to provide state independent navigation to functions at shell level. We developed an access control mechanism based on setting and revoking capabilities to provide access to workflow state dependent functions. We implemented the notion of a “Project Team” that owns an object instance to achieve instance level access control.

4 Conclusion

We have developed a set of Smart Tools to rapidly develop Web based Business Applications based on meta-design paradigm. This approach is well suited to develop

web based business applications for SMEs as it supports iterative development. We are now in the process of developing another set of tools to empower end users to develop their own business applications.

References

1. Ginige, A.: Collaborating to Win - Creating an Effective Virtual Organisation. In: International Workshop on Business and Information, Taipei, Taiwan: Shih Chien University and National Taipei University (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, A.: Re Engineering Software Development Process for eBusiness Application Development. In: Fifteenth International Conference on Software Engineering and Knowledge Engineering. San Francisco Bay, USA (2003)
4. Ginige, A.: New Paradigm for Developing Software for E-Business. In: IEEE Symposia on Human-Centric Computing Languages and Environments, Stresa, Italy, IEEE, New York (2001)
5. Epper, M.: Poor Project Management Number-One Problem of Outsourced E-Projects, in Research Briefs, Cutter Consortium (2000)
6. Armour, P.G.: The Case for a New Business Model: Is software a product or a medium. In: Communication of the ACM (2000)
7. Ginige, A.: New Paradigm for Developing Evolutionary Software to Support E-Business. In: Chang, S.K. (ed.) Handbook of Software Engineering and Knowledge Engineering, pp. 711–725. World Scientific, Singapore (2002)
8. Ginige, A., De Silva, B.: CBEADS©: A framework to support Meta-Design Paradigm. In: (HCII 2007). 12th International Conference on Human-Computer Interaction, Beijing, P.R. China. LNCS, vol. 4554, pp. 107–116, Springer, Heidelberg (2007)
9. Liang, X., Ginige, A.: Smart Business Objects: A new Approach to Model Business Objects for Web Applications. In: 1st International Conference on Software and Data Technologies. Setubal, Portugal (2006)
10. Ginige, J.A., Uma, S., Ginige, A.: A Mechanism for Efficient Management of Changes in BPEL based Business Processes: An Algebraic Methodology. In: ICEBE 2006. IEEE International Conference on e-Business Engineering, Shanghai China, IEEE, Los Alamitos (2006)