Study on Development Methodologies for Web Applications capable to Evolutions on Frameworks

Xiaolei Wu (1010207)

School of Information Science,
Japan Advanced Institute of Science and Technology

February, 2013

Keywords: Framework, Web Application, Development Methodology, Requirement Graph, Realization Graph.

1 Background and research aim

It is very common to develop Web applications by using framework. However, the demands which are required to each framework force them consistent change of their functions. In this thesis, we call it evolutions on frameworks. The Web application which is developed based on old version of framework cannot update against framework evolution. The developers have to maintain the Web application which is based on the new version of framework. The developers need to familiar with both versions of framework otherwise it is difficult to develop the new Web application.

In order to solve the problem we propose an approach that improve Web application when the framework evolutes. We focus on the structure of requirement elements, structure of realization elements, and relationship among both elements. We call this graph as Requirement －Realization－Relation　Graph(R3Graph).

2 Problem of framework evolution

When we develop Web application which is based on framework, the

common functions are provided by framework, such as user-interface transition, input and output functions. The developer only need to focus on implementation of user-interface and business logic process. We obtain an uniform implementation method according to the development rules which are defined by the framework. Therefore, it improves the development efficient.

During the evolution of frameworks, their structure might be changed by the improvement of their elements. For example, the structure and method for their configuration rule of logic process and rule of user interfaces such as fields and buttons. Therefore, the Web application which is developed based on old version of framework cannot update against framework evolution. But the developers need to familiar with both versions of framework otherwise it is difficult to develop the new Web application. These continuous change of application makes them complex and hard to maintain. Whenever frameworks change, the applications must be changed. Therefore, we need a systematic change method for Web applications capable to evolutions on frameworks.

3 Approach

In this research, we propose an approach that deals with the framework evolution problem. In this thesis, we propose a Requirement－Realization－Relation Graph. We define the following three graphs: (1) requirement graph whose vertexes are elements of requirements such as a unit of functions and whose edges are dependencies in requirements such as preconditions,(2) realization graph whose vertexes are elements of realization such as a dataset for DB accesses and whose edges are relationship among elements such as 'used-by' and (3) relationship graphs whose vertexes are all of requirement elements and realization elements and whose edges are specified relationship defined by the framework such as 'described-by'. We call (1) as requirement graph, (2) as realization graph and (3) as Requirement－Realization－Relationship graph (R3Graph).

We assume that the Web application is developed based on a framework. When the evolutes on frameworks, the parts of the application which associate to the framework are also changed. Therefore, if the requirement elements are not change, the R3Graph is also different according to different version framework. First, we decision the requirement factors and generate the requirement graph. Second, we

decision the implementation factors which are based on original framework and evolutionary framework. And then we generate the different realization graphs. By comparing the different graphs, we find the necessary change parts and define the change rules which are based on R3Graph. We can obtain the change of implementation throw the change of the graph. Finally, we can obtain the change of the application which reflects the evolutions of frameworks.

4 Case study

As for agenda of case studies, we use a small Web application in the field of hotel reservation management systems. There are two DBs in the system:(1) Reservation record and (2) Room status. And five functions: (a) register new user (b) authorize the user when login (c) submit a new reservation (d) retrieve old reservation and (e) cancel the reservation. Each reservation record contains a set of rooms. Two DBs must be updated with consistency when a reservation is inserted/deleted. We use our systematic change method to implement the change of application when the framework updates from Struts1 to Struts2. And then we evaluate our approach. First, we generate two Implementation Graphs based on Struts1 and Struts2 respectively. By comparing two graphs, we find the necessary change parts. Second, we define two change rules which are based on R3Graph: (1) Moving *model* to the inside of *Action*; (2) Unifying *ActionForm Bean* and *Action*.

When we add new functions to the system, we evaluate change rules by comparing two change costs. One use the graph and another doesn't use the graph. We generate Implementation Graphs associate with new functions for two different applications. We can obtain the new function's implementation factors which are based on Struts2 from the ones which are based on Struts1 according to the change rules. Thus, we can see that our approach and the change rules are effective.

5 Summary and future work

In this research, I studied the development methodologies for Web applications which capable to evolutions on frameworks. We focus on the structure of requirement elements, structure of realization elements, and relationship among both elements. We proposed a Requirement—

Realization－Relation Graph(R3Graph). By comparing two R3Graphs which are generated from original application and evolutionary application, we make up the change parts and define the change rules. We can see that our approach is effective through the case study.

In the future, our approach is needed to be improved in order to adapt to various frameworks. Implementing an automatic approach is another issue.