CHAPTER 8

SUMMARY AND CONCLUSIONS

This chapter presents summary and conclusions of the research work done in this thesis. It also gives future directions of the work that can be carried out further.

8.1 CONCLUSIONS

This thesis first proposed seven multiple views for MoDSE. For each view, view points are identified. Each proposed view is derived by using these viewpoints. Different concerns of the stakeholders are satisfied by each view, providing sufficient knowledge regarding MoDSE. Proposed views are validated analytically by using eight Lehman’s laws of software evolution. From this validation it is observed that the proposed views support seven laws. These proposed views are sufficient for the stakeholder to capture the information about models during evolution in context of MoDSE. Stakeholder can only understand evolution of models in different perspectives by using proposed views. But these proposed views cannot address the stakeholder’s concerns in detail and also the tools which are required for those concerns. To overcome this limitation a framework is proposed. GQM paradigm is used for derivation of framework. In this framework Goal refers to key area, Question refers to key feature and Metric is a quantitative measure for the key features which makes application of GQM Paradigm straightforward. Above proposed views are introduced as key areas in the framework. For each key area, key features are identified. These Key features are possible questions that can be posed by the stakeholders to understand their concerns. Intended stakeholders are identified and various evolution activities are considered as concerns in MoDSE. To measure the key features quantitatively 6-level likert scale is used and each measure has intrinsic meaning. Proposed framework is evaluated in two ways – Tools assessment and Stakeholders assessment. Tools assessment is required in MoDSE to measure the impact of tools to which extent the use of a tool
facilitate the execution of a particular evolution activity and how does one compare the performance of different tools to carry out the same evolution activity. Stakeholder’s assessment is also essential to check whether the key features identified are sufficient or not. From these assessments it is observed that comparison and assessment of various tools is manual and tedious work when there are huge numbers of tools. Framework can be used as a uniform platform to evaluate the tools performance. Stakeholder should have minimum knowledge about the tools to assess their performance. Software practitioners participated and claimed that concerns identified are sufficient to understand the evolution of models in MoDSE.

To overcome the tedious and manual work of tools assessment in the proposed framework, a recommendation system is implemented in this thesis. To date there is no such type of system to recommend tools in MoDSE. Proposed recommendation system is automated to compare the performance of different tools to carry out the same evolution activity. Three step approach is proposed to generate recommendations for selecting appropriate tool for right concern in MoDSE. ‘MapReduce’ technique is used to compute pair wise similarity between various tools. Pair wise tool similarity matrix is also computed for generating comparison summary. Proposed recommendation system is implemented and named as ‘mROSE’. Four major components of mROSE are Basic Mode, Expert Mode, Tool Comparison and Knowledge Base. User can interact with mROSE through either Basic Mode or Expert Mode. All possible stakeholder interests and/or concerns, tools used in MoDSE are considered in the repository known as Knowledge Base. Tool comparison component implicitly works as recommendation engine which generates timely and useful recommendations. Evaluation of mROSE consists of performance evaluation, longitudinal user study and laboratory user study. Participants from various organizations have participated in both the user studies. Performance evaluation is done by using available performance metrics such as response time, storage requirements and computation complexity. From these results it is proved that mROSE is performing well. During both the user studies participants are asked to answer nine tasks and a questionnaire. Participants claimed that answering tasks is time consuming and tedious without using mROSE. So, mROSE is useful to select right tool for the right concern and also to understand evolution of models
in MoDSE. Participants have rated the mROSE as more informative, user friendly, not often frustrating, learning about tools and MoDSE made easy. Evaluating mROSE by longitudinal and laboratory user studies conforms the desire for a system like mROSE.

8.2 FUTURE DIRECTIONS

Proposed multiple views are used to understand the evolution of models in Model-Driven Software Evolution. These views are not intended to discuss about managerial decisions which are required to maintain and control the evolution activities. These management decisions about system growth and evolution not only depends on the stakeholder’s feedback inputs and controls but also on external forces like trade unions or the availability of other resources in the organization. Therefore, in this aspect proposed multiple views can be extended to discuss the resources and local corporate management activities. To construct the managerial or organizational views, viewpoints will be identified in the stakeholder’s perspective. Stakeholder’s role like project manager, team leader etc., may be added to understand the influences of management in organizational levels. In this aspect new concerns will be identified. However, newly constructed views and concerns can be introduced as key areas and key features in the proposed framework.

Evaluation of the proposed recommendation system reveals many future directions like change in user preferences, prediction generation, and ranking mechanism. Increase in number of concerns and tools rises the scalability which is the major challenge of many existing recommendation systems is subject of future work. Investigations which were conducted involving participants like academicians, software practitioners, research students suggest that trust is a critical factor in the acceptance of recommendation systems and is not surprising. To borrow trust from a trusted peer, this might be possible through rephrasing of presented recommendations by preserving user interactions, predictions and ranking mechanisms.