Jurnal Teknologi, bil. 23, Jun 1994 hlm. 7 – 14

END-USER INVOLVEMENT IN SOFTWARE PROTOTYPING

MOHD. HASSAN SELAMAT
MD. MAHBUBUR RAHIM
ABU TALIB OTHMAN
Department of Computer Science
Faculty of Science
Universiti Pertanian Malaysia
43400 UPM, Serdang, Selangor
Malaysia

Abstract. Software prototyping is an attractive alternative approach to systems development, within which end-users exercise with a series of prototypes in a working environment. Such interactions offer an excellent scope for meaningful participation of end-users in the system development process. This is crucial for the success of a project. However, despite its significance, relatively little attention has been paid towards the management of end-user involvement. Advocates of prototyping approach merely express a need for active user participation in the prototype development process. They fail to offer any practical guidelines. Moreover, few documented case studies concerning management of user involvement within a prototype project have been published. It is argued that managing user participation is not a trivial task and it requires considerable attention. In this paper, the authors propose a framework that relates degree of user involvement with the type of prototyping approach adopted and the stages of prototype development process.

Keywords: Software prototyping, user involvement, systems development, user management.

1 INTRODUCTION

Software prototyping is receiving recognition as a viable alternative systems development paradigm in which users participate heavily in the development process. In prototyping, users are not considered as passive observers (Vonk, [23]). Instead they play more active roles than is possible in traditional development methods (Neumann and Jenkins, [15]). Their participation provides a defiwtion of the delivered system (Scharer, [18]) and raises commitment to the system (Alavi, [1]). As such, enthusiastic user involvement is crucial for the success of a prototype project. However, despite its significance, relatively inadequate attention has been paid regarding the management of user involvement in prototyping. Existing literature shows that most of the efforts have been directed to devise development methodologies of prototyping such as rapid prototyping (Boar, [4]), ADISSA based Prototyping (Shoval and Pliskin, [22]), structured rapid prototyping (Connel and Shafer, [60]) and operational prototyping (Davis, [7]). The majority of the proponents of prototyping merely express a need for active user involvement in the prototype development process. They fail to offer any useful guidelines to manage, organise and control user involvement in a prototype project. Furthermore, (Kieback et al [12]), while assessing several industrial prototyping projects, observed that misconceptions exist relating to the question of how and to what extent users should be involved in a prototype development process. In many

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

cases, they reported that user management was reluctant to allow the actual end-users of an application system to participate in the evaluation of the prototypes. This indicates that there is a need to identify the extent of end-user involvement prior to undertaking any prototyping project. In this paper, the authors propose a framework to assist in selecting the appropriate degree of user involvement during the various stages of prototyping. This would be useful, as it provides a direction in managing user involvement in a prototype project.

2 USER INVOLVEMENT IN PROTOTYPING: PRIOR RESEARCH

Existing literature reveals that inadequate research efforts have been devoted to address issues associated with end-user participation in software prototyping. The majority of the proponents (Appleton, [2]; Boar, [4]; Burns and Dennis, [3] Park, Chae and Kang, [16]; Gavurin, [8]) of prototyping assert that active user participation is essential before undertaking a prototype project, but they do not specify to what extent and at what stages of prototyping such involvement would be required. However, some of the authors (Harker, [10]; Carey, [5]; Mayhew and Deamley, [14]) have focussed on several important aspects concerning user characteristics. These are: selection of users, number of users and responsibilities of users.

Table 1 Advantages obtained from single and multiple users

Advantages	No. of Users	
	Single Users	Multiple Users
Enables user debate		×
Generates feeling of ownership		×
Uses combined knowledge of many users		×
Easy to manage user expectations	×	
More prototype iterations are less likely	×	

2.1 Selection of users

Harker [10] advises recruiting a sample of users which is as representative as possible on critical dimensions like age, sex and grade of employment. She further suggests that the regional trends in the characteristics of the users must be taken into account, if the delivered system is to be spread across the whole country. Carey [5] identifies several variables that can help profile end-users. Some of them include previous exposure to computers, the nature of the task users are attempting to perform, level of training, level in the organisation and amount of dependency on the system. Mayhew and Deamley [14] state that the selection of users is the responsibility of the user management and depends on such criteria as status, experience, attitude and enthusiasm. They advocate choosing users from both management and end users. Vonk [23] states that prototyping is possible only when the working models are evaluated by users who have the authority of taking sensible decisions in respect of the system requirements definition.

2.2 Number of users

Authors are divided into two groups over the issue of number of users to participate in a prototype development project. For instance, (Boar [4]) cautions that in a 'stereotype' image of prototyping, the usual assumption is that prototyping takes place with a user and a developer. However, (Mayhew and Deamley [14]) highlight several disadvantages of including only one user in the prototype development process. They argue that involving multiple users provide a greater wealth of knowledge. On the contrary, (Jenkins [11]) prefers to develop a prototype with one typical user and then to use it as a pilot with all other users. The advantages obtained from these divergent views are summarised in Table 1.

2.3 Responsibility of users

Most of the proponents of prototyping advocate that in a prototype project, the key task of end-users is to exercise with working prototype in order to evaluate its usefulness and inadequacies. Connel and Shafer [6] suggest that users should be required to experiment with the prototypes to find its flaws and missing elements. Their responsibility is to be able to state clearly what the prototype is doing wrong, which functions are difficult to use and why. What additional functionality is needed to make the system more useful? They argue that finding these defects is not a trivial responsibility of the users. Vonk [23] states that it is the users' responsibility to check that the system is complete from a functional viewpoint. The users must also be able to arrive at an ergonomically sound user interface that appeals to all of them. The task characteristics as described by them are shown in Table 2.

Table 2 Task characteristics of users in a prototype project

Task	Viewpoint	
	Functionality	Ergonomics
Finding missing functionality	×	
Finding defective functionality	×	
Finding what functionalities are difficult to use	×	×
Ease of use of menu		×
Ease of use of help and error messages		×

However, (Pliskin and Shoval [22]) differ from the previous authors. They suggest that in addition to the prototype evaluation tasks, some of the simpler development tasks could be delegated to the users, depending on their degree of sophistication and previous computer experience.

2.4 Synthesis of prior research

It is thus obvious that existing literature mentions user involvement in a very superficial fashion. A majority of the authors just state the need for active user participation. However, several authors as mentioned in the preceding sections, attempted to focus some the important aspects of user management. Their works can be best described in Figure 1. In this paper, the authors introduce another aspect that is known as 'Degree of user involvement in prototyping', which has not been highlighted by other researchers.

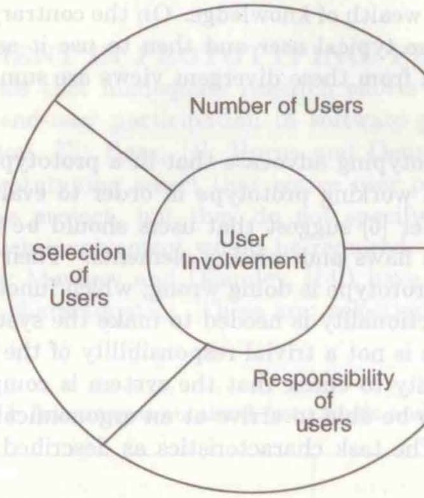

Fig. 1 Three dimensions of managing user involvement

3 DEGREE OF USER INVOLVEMENT: A FRAMEWORK

The proponents of prototyping propose to involve users during requirements gathering and the prototype review phases only (Appleton, [2]; Boar, [4]; Vonk, [23]) to highlight the deficiencies in the prototype. Very few authors advocate involving users in other tasks. The authors argue that it is important the participation of end users should not be limited to prototype review phase only, rather it could be effectively distributed to the entire life-cycle of the prototypes. Such distributed involvement would definitely enhance user commitment and facilitate user acceptance of the delivered system. In this context, an approach for selection of appropriate degree of user involvement in prototyping is proposed. The underlying philosophy of this approach is that the degree of user involvement is dependent on two factors, such as: (a) type of prototyping approach adopted and (b) stages of prototyping.

3.1 Types of prototyping approach adopted

In a prototype project, three different approaches can be adopted based on the degree of sophistication of end users. These are : a) traditional prototyping, b) shared prototyping, and c) end-user prototyping.

In traditional prototyping, users are allowed to participate in the more conventional way. They are interviewed during the requirements gathering stage and sometimes consulted for requirements clarification purpose in the analysis stage. However, their contribution is maximum when they are allowed to exercise with the working prototypes during evaluation stage. This exercise assists greatly in identifying missing or inconsistent features in the system. The majority of the popular prototyping methods such as those suggested by (Gomma [9], Young [24], Boar [4], Vonk [23]) fall within this category.

In shared prototyping, the developers can delegate some of the simple analytical, design and implementation tasks to the knowledgeable users. However, the majority of the development tasks are accomplished by the developers. In this approach, the users can render assistance in a more meaningful way. Shoval and Pliskin [22] cited case studies of responsibilities among users and developers in ADISSA based prototyping.

End user prototyping (EUP) can be considered as a product of both end user computing (EUC) and prototyping where sophisticated end users, after Teaming through EUC, are incorporated in information systems (IS) development (Kraushaar and Shirland, [13]; Pliskin and Shoval, [19]). End user prototyping is different from both traditional prototyping and shared prototyping since under either of these latter development strategies, users are assumed to have no data processing (DP) knowledge and take less participative users role (Neumann and Jenkins, [15]). In EUP, owing to the users' sophistication, development of applications can be initiated before the DP department is even aware of the need for an application (Selamat, [20]). Users are more active and are even involved in the design and actual building of the application, while the DP staff can assume supervisory role and emphasize discipline, planning and control.

3.2 Stages of Prototyping

The framework of prototyping can be best described by the 'State-Structured Transition Model' as developed by (Rahim [17]) which is shown in Figure 2. This model views every 'version' of a prototype as a 'state' and recommends to accomplish transitions from one 'state' to another through a series of six stages, which grow in size in successive iterations. Prototyping does not necessarily imply that users' involvement would be limited to the requirements gathering and prototype interactions stages only. Instead, user involvement could be expanded to cover all the six stages of structured transition. However, such involvement depends on the user competency and the type of prototyping approach adopted.

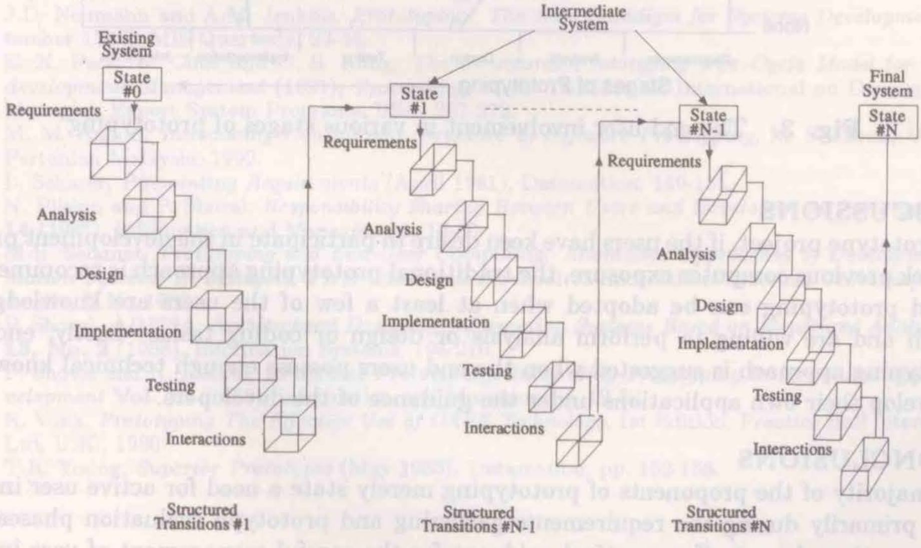

Fig. 2. The details of inner view of the state-structured transition model

It is argued that the extent of user involvement is not the same at all the six stages of structured transition. The proposed degree of user involvement during the six stages of prototyping is shown in Figure 3. This figure demonstrates that user involvement is related to the type of prototyping approach adopted and varies during the various stages of prototype development.

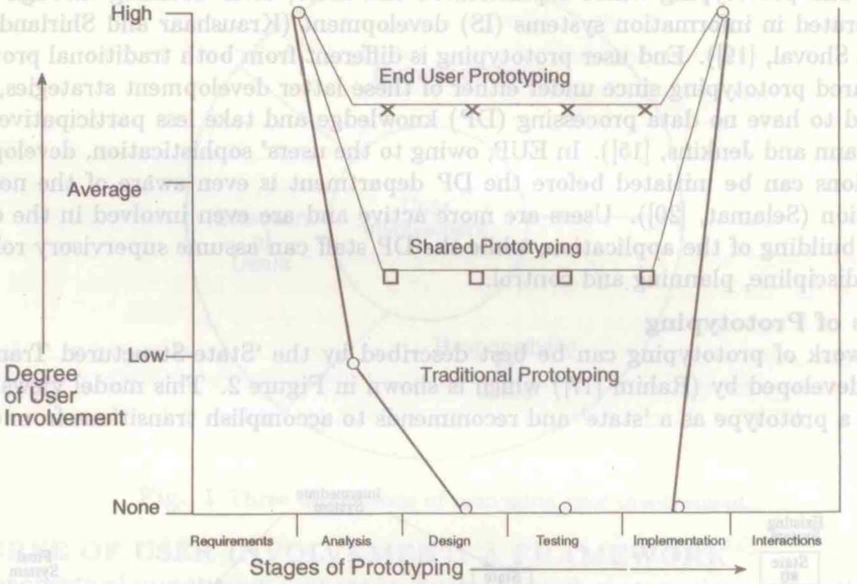

Fig. 3. The end user involvement in various stages of prototyping

4 DISCUSSIONS

In a prototype project, if the users have keen desire to participate in the development process but lack previous computer exposure, the traditional prototyping approach is recommended. Shared prototyping can be adopted when at least a few of the users are knowledgeable enough and are willing to perform analysis or design or coding tasks. Lastly, end user prototyping approach is suggested when the end users possess enough technical knowledge to develop their own applications under the guidance of the developers.

5 CONCLUSIONS

The majority of the proponents of prototyping merely state a need for active user involvement primarily during the requirements gathering and prototype evaluation phases only. However, they do not offer practical guidance for the careful management of user involvement. The participation of users could be made more meaningful if they could be involved at the stages of development. Keeping this in mind, the authors propose a conceptual framework to determine the extent of end-user involvement at the various stages of prototyping based on the types of prototyping approach adopted. This framework would enable IS practitioners to be more confident in obtaining better cooperation from the end users. This, in turn, would facilitate user acceptance of the system and enhance creditibility of the IS staff within an organisation.

REFERENCES

[1] M. Alavi, *An Assessment of the Prototyping Approach to Information Systems Development* **Vol. 27** (1984), Communications of the ACM, No. 6, 556-563.

[2] D.S. Appleton, *Data-Driven Prototyping,* (November 1983), Datamation, 259-268.

[3] R.N. Burns and A.R. Dennis, *Selecting the Appropriate Application Development Methodology* (1985), Data Base, Fall, 19-23.

[4] B.H. Boar, *Application Prototyping: A Requirements Definition Strategy for the 80,* 1st Edition, John Wiley & Sons, USA, 1984.

[5] J.M. Carey, *Prototyping: Alternative Systems Development Methodology* (1990), Information and Software Technology, 33:2, pp. 119-126.

[6] J. Connell and L. Shafer, *Structured Rapid Prototyping An Evolutionary Approach to Software Development,* 1st Edition, Yourdon Press, Englewood Cliffs, N.J. USA, 1989.

[7] A. M. Davis,, *Operational Prototyping: A New Development Approach* (September 1992), IEEE Software, 70-78.

[8] S. L. Gavurin, *Where Does Prototyping Fit In Is Development?,* Journal of Systems Management (February 1991), 13-17.

[9] H. Gomma, *The Impact of Rapid Prototyping on Specifyping User Requirements* (1983), ACM SIGSOFT Software Engineering Notes, 8:2,17-28.

[10] S. Harker, *The Use of Prototyping and Simulation in the Development of Large-Scale Applications* **Vol. 31, No. 5** (1988), The Computer Journal, 420-425.

[11] A.M. Jenkins, *Prototyping: A Methodolology for the Design and Development of Application Systems* (1983), Discussion Paper, No.227, Graduate School of business, Indiana University, USA.

[12] A.H. Kieback, H. Lichter, M.Schneider and H. Zullighoven, *Assessing Industrial Prototyping Projects* (January 1992), Proceedings of the 25th Hawaii International Conference on Systems Sciences, 564-565.

[13] J. M. Kraushaar and L. E. Shirland, *A Prototyping Method for Applications Development by End Users and Information Systems Specialist* (September 1985), MIS Quarterly, 189-197.

[14] P.J.Mayhew and P.A.Dearnley, *Organization And Management of Systems Prototyping* **Vol. 32, No. 4** (1990), Information and Software Technology, 245-252.

[15] J.D. Neumann and A.M. Jenkins, *Prototyping: The New Paradigm for Systems Development* (September 1992), MIS Quarterly, 29-44.

[16] E. K. Park, K. Chae and C. S. Kang, *The Structured Prototyping Life Cycle Model for Systems development Management* (1991), Proceedings of the IEEEIACM International on Developing and Managing Expert System Programs, USA, 267-272.

[17] M. M. Rahim, *Introducing Control and Structure in Software Prototyping,* M. S Thesis, Universiti Pertanian Malaysia, 1992.

[18] L. Scharer, *Pinpointing Requirements* (April 1981), Datamation, 149-151.

[19] N. Pliskin and P. Shoval, *Responsibility Sharing Between Users and Developers in Prototyping* **Vol. 14** (1987), Information and Management, 19-30.

[20] M.H. Selamat, *Prototyping and End User Computing: Altemative Approaches of Developing Information Systems in Malaysia,* Ph.D Dissertation School of Information Systems. University of East Anglia, U.K., 1988.

[21] P. Shoval, *ADISSA: Architectural Design of Information Systems Based on Structured Analysis* **Vol. 13, No. 2** (1988), Information Systems, 193-210.

[22] P. Shoval and N. Pliskin, *Structured Prototyping: Integrating Prototyping in Structured System Development* **Vol. 14** (1988), Information and Management, 19-30.

[23] R. Vonk, *Prototyping The Effective Use of CASE Technology,* 1st Edition, Prentice Hall International Ltd, U.K., 1990.

[24] T.R. Young, *Superior Prototypes* (May 1983), Datamation, pp. 152-158.