1987

A Prototyping Methodology for Developing an Online Distributed Banking System

Mohammed H. Omar

University of Richmond

Follow this and additional works at: https://scholarship.richmond.edu/robins-white-papers

Part of the Business Commons

Recommended Citation

This White Paper is brought to you for free and open access by the Robins School of Business at UR Scholarship Repository. It has been accepted for inclusion in Robins School of Business White Paper Series, 1980-2011 by an authorized administrator of UR Scholarship Repository. For more information, please contact scholarshiprepository@richmond.edu.
A PROTOTYPING METHODOLOGY FOR DEVELOPING
AN ONLINE DISTRIBUTED BANKING SYSTEM

Mohammed H. Omar

1987-4
A Prototyping Methodology for Developing an Online Distributed Banking System

by

Mohammed H. Omar, Phd.
School of Business - Management Systems
University of Richmond
Richmond, VA 23173
Tel: (804) 289-8577
ABSTRACT

A prototyping development method is presented here which was used to develop an online distributed banking system. The author's experiences indicate that a prototyping process can assist in the efficient development of complex application systems by breaking the problem into several parts. A prototyping model based on combining the main concepts of the traditional prototyping model, the state-transition model, and the overlapping phases model, is presented and discussed. A four prototyping method implemented over five years is presented in the context of this model.

INTRODUCTION

Prototyping as a method for system development has received a great deal of attention in the last few years. Prototyping has developed due to several factors including:

1. The availability of new tools like Data Base Management Systems, screen management software, and report generators.
2. The awareness to the need and advantage of user participation in the development process.
3. The benefits demonstrated in early prototyping efforts in engineering and business applications.
4. Complaints about the traditional life cycle methodology due to its inability to handle today's complex problems, especially when accompanied by a high degree of uncertainty.
The disappointment of users from the increasing application backlogs and the inability of MIS departments to satisfy the demands and requirements of the users, led to putting more emphasis on new development methodologies like prototyping.

The purpose of this article is to provide an understanding of prototyping concept and methodologies; illustrate this method with a case study derived from my experience in supervising the development of an online interactive banking system through prototyping; and report on the implementation of this system.

Prototyping - Definition

Naumann - Jenkins [10] define prototyping as a system that captures the essential features of a later system. A prototype system is intentionally incomplete, is to be modified, expanded, supplemented or supplanted.

Lantz [7] defines prototyping as building a model of an information system and using it for designing, testing, and installing the system.

Prototyping - A Methodology

Prototyping has been used extensively over a long period of time
for electrical and mechanical engineering systems design. Janson [5] defines prototyping in this sense as the process of building and testing models of the target system. A computer manufacturer, for example, always starts by designing and building a prototype which will be tested and revised till it meets the specifications and achieve the objectives of the target system. Information systems are similar to engineering systems in that they transform objects (data) that are undergoing a change in state. Hence the concept of prototyping became appealing for use in developing information systems as well as engineering systems.

Mason and Gary [9] describe a methodology for interactive information systems named the architecture-based methodology. The name is derived from the analogy with the architecture approach to a building. The system design consists of four elements: (1) developing a view of the system based on its external description or appearance embodied in some form of specification that is capable of complete and unambiguous interpretation by the user; (2) the adoption of a common view or design concept for the underlying structure of all applications using a screen dialogue perspective; (3) the concept of project management and determining the role of the user and the developer; and (4) the view of the development process as an iterative process.

Harrison [4], Cerveny-Garito-Sanders [2] classify prototyping into three categories: mockup, working model, and evolutionary.

Naumann and Jenkins [10] consider prototyping as one methodology for development versus the traditional life cycle methodology.

When To Use Prototyping?

Although prototyping as a methodology can be used to develop all kinds of applications, yet there are certain factors that make it a more attractive option, and in some cases the only methodology that guarantees the production of a working system that meets the user requirements. Some of the factors are:

1. The user environment is unstable or extremely dynamic.
2. The user is not able to specify a complete set of requirements.
3. The system developer does not have enough experience in the nature of the application.
4. The system is complex and serves a large number of users.
5. The uncertainty of successful implementation.

Advantages Of Prototyping

The purpose of installing a prototype can be any or combination of the following:
1. Improve the likelihood of developing systems that meet user needs.

2. Reduce the management risk in developing complex systems.

3. Shorten the overall development process.

4. Serve as realistic specifications for further system developments.

5. Allow the users to test all major system components that are critical to the success within the organization.

6. Provide training for staff members who will be using the system.

Prototyping Models

Naumann and Jenkins [10] present the prototype model of an information system as a four-step process (Figure 1). The first step is to identify the user's basic information requirements. The initial prototype includes the essential outputs of the proposed systems, in terms of data and process. The second step is to develop a working prototype. It represents the essential elements desired by the user in a simplified and modular form to enable future additions and enhancements. This prototype should be presented to the users to determine if the basic information requirements, developed in step one, have been incorporated. The third step
in this model is to implement and use the prototype system. As an operational system, it forces managers and users to participate more effectively in providing feedback about any shortcomings in the system and through the ongoing learning enable them to complete the definition of the information requirements for the total system. Step four is to revise and enhance the prototype system by incorporating the user feedback and improving the operational effectiveness of the system. Accordingly step three will be repeated and iterations between steps three and four will continue until the user accepts the system as a final operational system.
Step One

Identify Basic User Requirements

Step Two

Develop Working Prototype

Step Three

Implement and Use

User Satisfied

Yes

Operational System

No

Modify and Enhance

Step Four

Figure 1

Traditional Prototyping Model
Kraushaar and Shirland [6] presented a variation to the traditional prototyping model to what they called a state-transition model. The authors presented a three transition - two prototype model. The first transition is building an initial prototype. The steps involved in this transition include: (1) needs assessment to identify and prioritize the user needs, (2) design of the initial prototype, (3) implementing the initial prototype, (4) testing the initial prototype, (5) experimenting with the initial prototype and project, (6) evaluating the initial prototype and project. Based on the final evaluation of the initial prototype, a proposal for a second prototype is developed.

The second transition is building the second prototype. The overall process is similar to building the initial prototype, however, the design modified and extended the initial prototype to develop a more realistic model of the desired operational system. The initial sample of users was expanded to insure that user groups having an interest in the project were exposed to the prototype. Suggested changes at this stage were not implemented unless they were critical to continued successful experimentation of the prototype.

Transition three is building the operational system. Many components of the second prototype were transferred to the operational system making the development effort much shorter.
Lantz [8] presented another variation to the traditional prototyping model. The approach consists of three overlapping phases. The first phase defines the prototype through an initial feasibility study and an evaluation of the present system. The second phase begins in the latter stages of phase one by building the prototype. The third phase focuses on experimenting with the prototype in a way similar to the iterations between steps three and four in the traditional model in Figure 1. While phase three is underway, final plans for conversion and installation of the system are developed.

Boar [1] proposes a prototyping life cycle approach to prototyping. This results in a structured prototyping life cycle as a tool in the definition phase. Thus the preliminary design phase can take the finished prototype from the structured prototype life cycle and use it to specify the final system. From this step on, the traditional life cycle methodology can be applied.

THE BANKING SYSTEM CASE

Background

Late 1981 I joined Arab National Bank (ANB) to start a project for design, development, and implementation of a computer based banking system. The bank was using a manual system, thus my function included establishing an MIS department, determine the bank requirements, issue a request for proposal, procure the
equipment and software, select a development strategy, and supervise
the development and implementation of the system. This section
will explain the steps taken to achieve the bank objectives.
Historically the bank was a non Saudi bank owned by Arab Bank
Limited of Jordan operating through five branches in major cities.
In 1979, through a government plan for regulating banking in
Saudi Arabia, the bank became a share holding corporation with
sixty percent of the shares owned by Saudi public. The new board
adopted a plan of expansion to open fifteen new branches and
expansion was going according to plan. New employees were massively
hired, trained and put to work in the new branches. Experienced
employees in the old branches were transferred to leading positions
in the new ones. Top corporate management was being staffed and
new departments created to coordinate and control the process.

THE INITIAL PLAN

With the traditional life cycle development in mind, I started by
preparing an initial plan for development which included: (1)
preliminary study, (2) determine information requirements, (3)
prepare and issue the RFP, evaluate and select the hardware and
software suppliers, (4) system design, (5) system development,
(6) system testing and conversion, (7) system implementation, and
(8) post implementation review.
The Preliminary Study

This stage included a study of general characteristics of automated banking systems through survey of literature; arranged visits to banks in the United States, Europe, and the Middle East; and interviews with bank employees at the level of the branches and top management. Meetings with the board of directors clarified the goals of the bank in general and set the guidelines for the development of the computer based banking system. The main decisions included: (1) using a distributed approach to establish three regional computer centers, to install minicomputers in branches, and link the branches and regions through a communication network; (2) the computer system should be able to handle bilingual input and output (Arabic/English); (3) establish a steering committee to oversee the planning, development and implementation of the system.

Requirements Specifications

As a result of the preliminary study and further interviews with key personnel at the operational and managerial levels; and after studying the current manual system procedures and the available documentation the following outcomes and findings were identified:

- branch level systems include cash and check withdrawal and deposit system, branch accounting, letters of credit, loans and facilities, foreign exchange, transfers, and bills.
- general management systems include credit system, corporate accounting, investment, foreign exchange, and administrative
systems.
- data flow diagrams were prepared for each activity.
- user environment is going through continuous change due to expansion.
- users have no previous experience in computing and no past experience with automated banking systems.
- there is a great deal of uncertainty as to expectations from computer based system.
- the system is complex and will serve more than one hundred branches and a large number of users.
- large number of systems are to be implemented and integrated at the branch, region, and general management levels.
- the interfaces between the systems and between the branches and the regions were not clear.

Selection Of Computer Hardware and Software
Based on the information collected from the preliminary study and the general requirements that were identified so far, a request for proposal was prepared and submitted for bidding to major international computer hardware and software suppliers through their local agents in the country. The criteria for evaluation included the availability of bilingual banking software applications or the readiness of the supplier to custom tailor a prepackaged application or participate in developing a special application to the bank specifications, the ability of the local agent to install and support the system, the capability of the local agent to
maintain the hardware and software, and the cost of the complete system.

It was difficult to find a supplier to highly qualify for all of the above criteria. Finally, a compromise was made and the contract was awarded to the distributor of Digital Equipment Corporation to supply a turnkey system with close coordination with the bank MIS department. The system builder (local DEC distributor) had good experience in arabizing input/output devices and in implementing conventional bilingual systems in the country, but has no experience in banking applications. The bank MIS and operations departments were given the responsibility of assisting in defining the bank requirements and coordinating the interface between the system builder and the functional units in the bank.

The contract was divided into stages of implementation. the successful implementation of a stage triggers the start of the second stage with the option of terminating the remainder of the contract if the supplier cannot deliver according to a plan of implementation for each stage.

THE PROTOTYPING METHODOLOGY

Based on the previously mentioned characteristics of the user and system builder, and due to the dynamic nature of the organization and the uncertainty of successful implementation of such a complex system in a bilingual manner, the traditional life cycle for
development was found inappropriate in this case, and a prototyping methodology was adopted to go hand in hand with the spirit of the contract.

THE RESOURCE REQUIREMENTS

The total system was supported by the following components:

The Hardware Configuration

A distributed approach was adopted to follow the management structure which called for establishing three levels of management: branch management, regional management, and general management. Three regional computer centers were to be established with one of them also acting as the central computer system. The center hosts three VAX 11/780s linked through Ethernet, the two other regions were to host two VAX 11/780s. The branches were divided into three categories depending on the volume of transactions. The first type hosts a minicomputer PDP 11/44, the second type hosts a PDP 11/24, and the third type is a satellite branch with terminals linked to a minicomputer or another branch through a multiplexer. Terminals and printers were modified to accept bilingual input and output.

The System Software

The system software in the regions and the branches include:

* a relational Data Base Management System (TOTAL) and a query language
* a communication software (DECNET)
* an operating system (VAX/VMS, PDP RSX)
* a high level language compiler (COBOL)

Generalized Input and Output Software

A special software package for developing screen layouts and report layouts (TMX) was installed. In addition to I/O capabilities the software also supports terminal identification and monitoring functions and control of transfer of screens from one terminal to another.

The Applications

Due to the large number of systems, the complexity of the interfaces between the systems, and the uncertainties mentioned earlier; the applications were classified according to priorities and operational considerations into five stages to be implemented in succession.

Stage 1: includes the development and implementation of a customer information system at the branch level to include: the functions of opening and maintenance of customer accounts (current accounts, saving accounts, overdraft accounts, and time deposit accounts); the functions of withdrawal, deposit and check clearing; administrative and maintenance functions for addition, deletion, and update of customer information; query capabilities; and production of operational reports and customer statements.

Stage 2: includes the development and implementation of the branch accounting system and interface it with the customer
Stage 3: includes the development of the regional information system to enable interbranch banking between the branches in different regions and provide online backup for the files and operations of the branches in each region and produce consolidated reports about the operations and accounting in each region.

Stage 4: includes the development and implementation of the rest of the branch systems: the letter of credit, loans and facilities, foreign exchange, transfers, bills and collections; and integrate the new systems with the customer information and accounting systems.

Stage 5: includes the development and implementation of the general management systems: the credit, investment and foreign exchange; and provide consolidated reports and query capabilities from the corporate data base.

THE PROTOTYPE MODEL

Due to the unique nature of the project and the requirement for bilingual operation of the system, two types of prototyping models were used.

The Illustrative (Mockup) Prototype:

The objectives of this prototype include: (1) testing the terminal capability of displaying bilingual characters, (2) testing the capability of the printers to display bilingual characters in an acceptable form, (3) testing the generalized input/output software
(TMX) and its ability to format screens and reports in bilingual form, (4) presenting the user with examples of actual screen and report layouts to demonstrate the ease of use of the input screens and the movement of the cursor in the correct direction from right to left for Arabic data and from left to right for English data.

The prototype was developed over a two month period. After testing the prototype, the MIS department and the technical committee in the bank were satisfied with the results and gave the vendor the green light to proceed in developing the first stage in the bank applications. The prototype was dropped after it served its purpose.

The APPLICATION PROTOTYPES

The prototyping model used for developing the banking system draws on several prototyping methodologies and models.

1. The concept of iterations is based on the same concept found in the traditional prototyping models.

2. The steps involved in defining, designing, building, testing, and implementing the prototype are similar to the steps used in the state-transition model.

3. The steps used to transit from one state to another are not sequential. They overlap depending on the ease of communication and the degree of cooperation between the users and the system builder.
4. The sensitivity of banking applications and the unfamiliarity of the user with automated applications forced the system builder to incorporate more structure into the methodology and prepare system documentation and user manuals for every prototype and keep updating them as the process continues.

5. Looking at the steps included in building each prototype, we find it similar to the steps in the traditional life cycle methodology with the exception that the scope of each prototype was much less that the operational system making it easier to develop and implement quickly.

The strategy used in developing the banking system required the development of four prototypes. The first prototype provided the users/system builder with an application framework to explore the user needs for an operational system. The second prototype was developed as an expansion to the first prototype to further include the needs of the users and be a working prototype and distributed to all branches. The third prototype was to incorporate the branch accounting and the interfaces to the other system and to the regional centers. The fourth prototype was to include interbranch banking and complete the networking between the branches, regions and centers.
The following steps were included in developing the first working prototype. Figure [2] represents the model used for this purpose; the overlapping of the steps and the iterations are shown in the model and in figure [4] as well.

1. Needs Assessment

The needs assessment for the first prototype was based on the feasibility study conducted for the purpose of preparing the RFB and the contract with the supplier. Further meetings with bank operations personnel and the branch managers, detailed study of the existing manual system and implemented procedures and the decision of the bank management on prioritizing the functions to be automated were the basis for defining the basic user requirements. The objective of this prototype was to develop a working system to perform the basic functions of the branch deposit withdrawal system as implemented in the manual system.

2. Design of the first Prototype

Our experience and findings of previous research indicate that the user interface is the most critical factor in determining the success of a prototype. In our case, the interface consisted of a series of linked screen menus. In addition, the major functions (menus) were callable by pressing programmed function keys thus bypassing some menus to arrive to the desired function. The design stage also included the indentification of the data items needed in this prototype and the design of the reports.
expected to be produced by the prototype. The number of screens
and reports are shown in table [1]. Accordingly the database was
created and the relationships between the data items were identified.
3. Building the prototype

For the purpose of building and later testing of the prototype,
a development branch machine, and a testing machine were installed
at the builder premises. A machine was also installed in the
bank MIS center for the bank testing. Using the screen formatting
and report writing features of the TMX software, the TOTAL DBMS
and the COBOL compiler, the prototype application program was
created.
4. Testing the prototype

The correctness of the application program was determined by
three levels of testing: (1) the system builder test data, (2)
the bank MIS test using copies of actual transactions obtained
from a branch bank, (3) system efficiency test performed by
external team from Society General Bank in Brussels - who developed
a similar system using same hardware and system software - on
behalf of the bank management. The prototype was modified to
include the many fixes and recommendations.
5. User experimentation

The modified prototype was installed on the bank machine. A
group of ten supervisors from the branches deposit sections were
given introductory training and were given the chance to experiment
with the system and record their observations. Another group of
branch managers went through the same experience. A walkthrough
was also done by the EDP manager of the partner bank. A lengthy list of errors identified, more user requirements, and recommendations for future prototypes were compiled. The errors and recommendations that were critical to the working of the system were incorporated in a new version of the prototype. The rest of user requirements and recommendations were identified to be included in the next prototype.

6. Installation of the prototype:

The new modified version of the prototype was installed in a small branch to run parallel to the existing system. At this stage, the prototype design was documented and an operator manual as well as user manual were prepared. Branch personnel were trained on using the system. Reconciliations were made daily and differences were identified and erroneous routines were corrected on the development machine. New requirements were identified to be included in the next prototype.

7. Live implementation

The prototype was modified and tested to the satisfaction of the branch personnel and became the working system.

BUILDING THE OTHER PROTOTYPES

User requirements identified from each previous prototype experience were incorporated in the design of the next prototype. The main steps were repeated until a working prototype is in place. The following are the major features and changes in each
of the remaining three prototypes.

Prototype 2:

The first prototype produced an acceptable performance and proved the reliability of the system software. Major changes were in the area of interest calculations where the users realized that utilizing the computer capabilities enabled them to do this function in a different way than it used to be done in the manual system. The proposed printers were dropped and a more reliable and faster type of bilingual printers were placed instead. The computer operation procedures were simplified to make it operator independent. The backup procedures were improved and the media was changed to fixed disk rather than a limited capacity removable disk that was proposed. During this period the computers proposed for the regional centers were changed from one DEC 20/60 to two VAX 780's due to the fact that DEC was now supporting the VAX line more than the other line.

Prototype 3:

Due to the inclusion of the accounting function, and the large number of screens and reports that became part of the system, and due to the large number of changes in the design, a complete redesign of the system took place using the experience gained from the first two prototypes. This explains the lengthy period of time it took to develop and implement this prototype.

Prototype 4:

The last stage of the project has been developed along the same line of the previous prototypes. Development has been
complete and tested and is now under implementation.

CONCLUSIONS

This experience indicates that complex real life systems take a long time to implement. During that time, users and their environment change creating the risk that a system when implemented might not fit the users needs. In cases similar to this experience, where there is dynamic change in the environment, user needs are difficult to define for the total system, similar applications are not available and the users are eager to get an operational system as soon as possible; the prototyping approach proved to be the most practical methodology.

User involvement in the needs assessment, the testing, experimenting, and implementing of the system led to reasonable system expectations with fewer surprises and ended up with the user ownership of the operational system and his ability to maintain and implement the system in the various branches. The user involvement was a great asset in speeding the training, the cutover and implementation of the system.

The main problem that arose from our experience includes: (1) user demand for full documentation of every prototype which lengthened the time of development, (2) as the users get more involved in the system, the requirements tend to increase rapidly.
(3) the high turnover of the user personnel requires continuous retraining of the users interfacing with system builder, (4) and finally the most critical problem is the tendency from the user side to keep the system open for modifications as long as possible especially in prototypes 3 and 4 where the user was satisfied with the working of prototype 2 and does not have the qualified manpower to distribute that prototype to other branches and at the same time get involved in the development stages of the rest of the prototypes.

Overall, the prototyping methodology was the only way this kind of system could have been implemented, given the limitations and characteristics of the user and the system builder.
Needs Assessment
Design Prototype
Build Prototype
Test Prototype

Modify

Critical

Yes

Satisfied

Yes

Install, Operate
In Parallel Mode

Satisfied

Yes

Implement
Distribute

Critical

No

Modify

No

NEXT Prototype

NEXT Prototype

NEXT Prototype

Prepare User Manuals
Prepare Conversion Plan

User Experimentation with prototype

Document System
FIGURE [3] PROTOTYPE 2, 3, 4 MODELS
Figure (4)
<table>
<thead>
<tr>
<th>PROTOTYPE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO. of Screens</td>
<td>19</td>
<td>25</td>
<td>30</td>
<td>156</td>
</tr>
<tr>
<td>No. of Batch reports</td>
<td>6</td>
<td>28</td>
<td>50</td>
<td>60</td>
</tr>
<tr>
<td>No. of online reports</td>
<td>-</td>
<td>12</td>
<td>15</td>
<td>20</td>
</tr>
<tr>
<td>No. of Inquiry screens</td>
<td>3</td>
<td>8</td>
<td>11</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 1
Number of screens and reports in each prototype
REFERENCES


