IMPLEMENTING A MARKOV-BASED ACCOUNTS RECEIVABLE DSS: A PROTOTYPING APPROACH

Julie E. Kendall
Rutgers University

Marc J. Schniederjans
University of Nebraska-Lincoln

A microcomputer was used to develop a prototype DSS for a Markov-based accounts receivable problem. The DSS was developed for and used by decision makers in a U.S. distribution center for religious articles. Analysis and evaluation of the application reveals that the use of the microcomputer-based DSS helps end users understand the use and decision value of the information provided by Markov analysis, enables decision scientists to demonstrate the usefulness of their methodologies, and helps analysts successfully prototype models on a microcomputer before transferring them to larger systems.

When a decision scientist suggests a new decision support methodology, the management information systems (MIS) group is often left to implement it. Implementing a change in the way decisions are supported and made can be met with significant resistance by managers or end-users. Indeed, recent research has shown that in some organizations employees actively try to sabotage software innovations to resist their implementation (Barton, 1987). This resistance can pose problems when decision scientists or systems analysts try to convince an organization’s management of the benefits of incorporating new decision models into their current systems. Resistance comes about for many reasons. In some cases it occurs because the decision scientists do not do an adequate job of presenting the benefits of the new methodology (Tingley, 1987). In other situations management simply does not want to incur implementation costs such as:

1. manager’s time spent learning new software,
2. manager’s time spent understanding new output,
3. possible downtime of the mainframe to install the new package, and
4. manager’s time spent developing the new package.

Ideally users will participate actively in all stages of systems development. The reality is that many users have little time for helping consultants...
or specialists develop software systems. For instance, most U.S. managers “learn-while-doing” or learn how to use software while trying to make profits for their organization. A successful implementation strategy for software should not only save managers the implementation costs, but should also permit them to be active participants in the development of the software. Active participation in systems development helps convince management of the merits of new methodologies.

This paper focuses on an approach that can be used to bridge the gap between introducing a new decision science methodology and successfully incorporating it into daily practice. This paper illustrates how a prototyped microcomputer-based decision support system (DSS) was used to ensure adoption of a management science model. Specifically, this paper is a case study of how a prototype DSS was used to implement a Markovian decision science model in an accounts receivable department of a national clothing distribution operation. Benefits reported by end users and system developers are also presented.

Case Background

The North American Youth Ministries (NAYM) Distribution Center of Lincoln, Nebraska is a major international distribution operation for uniforms and ceremonial clothing for religious groups. They also distribute educational arts and crafts activities kits. During the mid 1980s NAYM merged with a California company, thereby changing from a regional to an international distributor rapidly. Dramatic changes were made in its operations location, staffing, and operating systems.

One of the problem areas that surfaced during this period of time was the organization’s accounts receivable which were tying up a considerable amount of the organization’s funds. The information collected and provided to management on accounts receivable could be characterized as “descriptive” at best. The accounts receivable were grouped into four overdue accounts payment categories of 30 days, 60 days, 90 days and 120 days overdue. Daily transactions (i.e. receipts and disbursements) were tracked via an accounts receivable module on the mainframe and a descriptive monthly report on the current status of customer accounts and dollar values that fell in each of the four categories was then provided. No projections or forecasts of the number of accounts or their dollar value amounts were given. The rapid growth which NAYM underwent in the distribution center during the mid 1980s placed considerable pressure on the financial aspects of the operation. As a result of this financial pressure, the management of NAYM sought to tighten their control of sources of funds, such as accounts receivable.

Accounts Receivable Planning

Using Markov analysis for accounts receivable planning is well documented in the literature (Barkman, 1981; Clec & Icerman, 1980) and basic management science textbooks (Lee, Moore & Taylor, 1990 and Rubin & Stinson, 1986). As described in Lee, et al, the analysis requires the construction of a transition matrix P where:

\[
P = \begin{bmatrix} I & 0 \\ R & Q \end{bmatrix}
\]

1

I = an identity matrix
0 = a matrix of zeros
R = a matrix containing the transition probabilities of being absorbed in the next period (i.e. being an account that is paid up or, alternatively, being recognized as a bad debt)
Q = a square matrix containing probabilities for movement among all nonabsorbing states (i.e. accounts that are 30, 60, 90 or 120 days overdue)
overdue)

The absorbing transition states are either “Account Paid” or “Bad Debt.” Once entered a customer’s account is permanently retained in that state. The nonabsorbing transition states include the four stages in which a customer’s account moves from 30 days overdue, to 60 days, to 90 days, and finally to 120 days overdue. Beyond 120 days overdue the account is moved into the “bad debt” state. The actual probabilities used in the transition matrix for the absorbing states are always set at a value of one. The probabilities for the nonabsorbing states are obtained (as they were at NAYM) by tracking a set of customer accounts and recording the number that remain unpaid during a four month period.

Once the transition probabilities are obtained they can be used to determine the expected time before the accounts become paid off or become a bad debt. These planning statistics are found by determining the fundamental matrix, F, where:

\[ F = (1 - Q)^{-1} \]  

(2)

In addition to the P and F matrices’ probabilities, status vectors C and D are necessary. The C vector is a listing of the current number of customer accounts for the four months (i.e., for stages) of accounts receivables. The D vector is a listing of the current total outstanding dollar values for the four months of accounts receivables. These dollar values are obtained by totaling the overdue customer accounts that were tracked to determine the (overdue) nonabsorbing state transition probabilities. The C and D vectors are used in combination with the P matrix to obtain the projected accounts and dollar amounts of accounts receivable that will either be paid or become a bad debt. By multiplying the fundamental matrix, F, by the R matrix we determine the H matrix, or where the H matrix probabilities, \( P_{i,j} \), denote the probability of transition from overdue A/R nonabsorbing states to paid or bad debt absorbing states:

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>30 Days Overdue</th>
<th>P1,1</th>
<th>P1,2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Bad Debt</td>
<td>P2,1</td>
<td>P2,2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>P3,1</td>
<td>P3,2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>P4,1</td>
<td>P4,2</td>
<td></td>
</tr>
</tbody>
</table>

The H matrix represents the probabilities of the accounts receivable becoming a state of either an “accounts paid” or a “bad debt.” To project the number of accounts receivable that will be paid or become a bad debt in each of the four next months, we multiply the H matrix by the C vector. To project the dollar amounts of accounts receivable that will be paid or become a bad debt in each of the four next months, we multiply the H matrix by the D vector.

In summary, the Markov analysis requires the development of a probability matrix, \( P \), based on historical performance of accounts receivable. The I and Q matrices within \( P \) are converted into a fundamental matrix \( F \) by the matrix mathematics in Equation (2). The \( F \) and \( R \) matrices are then converted into the \( H \) matrix to determine the probabilities of the accounts being paid or becoming a bad debt. Finally, the \( H \) matrix is multiplied by each of the \( C \) and \( D \) vectors to determine the actual projections of accounts and dollar amounts that the firm can expect to receive or write off during the next four months.

Although the Markov model seems basic to those schooled in Markov analysis, it is not intuitive to those without this background. Unskilled decision makers will not bother to try to accomplish this kind of analysis on a calculator, even if it is available. As such, the Markov analysis is much more useful and approachable to them encapsulated in the DSS which presents information in an understandable format and permits them to make informed decisions based on their interactions with it.

Development of the DSS Prototype

The NAYM distribution center did mainframe computing on a time-sharing basis from a local
The university’s mainframe. The distribution center also made some microcomputer support available to managers. The mainframe had ample capacity to handle most applications and was constructed of functional areas modules for accounting, sales-marketing, purchasing-inventory, and so on.

For any component of the operating system to be reprogrammed, the entire module had to be shut down during the change-over period. Markov analysis could easily be programmed as part of the accounting module. Although development and testing could be performed without any down-time of the accounting module, the estimated implementation time of the proposed Markov model for use in accounts receivable was two weeks. This is disruptive to normal work routines and delays can be costly.

The prototyped Markov-based transitional decision support system had the following characteristics: it was oriented to decision making, supporting end users in providing structure to what was previously an unstructured problem to them, and by organizing their data into useful, understandable information that propels them into decision situations.

Decision making has been said to encompass three dimensions; the degree of decision-making skill required, the degree of problem complexity, and the number of decision criteria required. A useful DSS supports any or all of these three dimensions, and in doing so it will eventually change the decision maker. Change occurs by permitting the decision maker to structure problems or portions of problems in a new and different way. By interacting with the DSS the user is able to reframe problems from a new perspective. Extended interaction with the new perspective will change the way a decision maker approaches the problem (Kendall & Kendall, 1988).

The prototype accounts receivable DSS implemented for the NAYM Distribution Center was developed to permit quality interaction, taking into consideration that the end users were unskilled in using any decision science models, including Markov. The model used in the DSS is a simple Markov model that permits novice end users to effectively classify and compute overdue accounts. Also, the DSS implemented at NAYM supports the end user in balancing multiple criteria, as suggested in the above definition, since the end users were able to, with the help of the DSS output, consider multiple criteria such as customer goodwill, the value of pursuing an overdue account, and amount of additional staff time to pursue each course of action.

Finally, the A/R DSS changed the decision makers by changing the way they viewed accounts receivable. It provided a forecast of overdue accounts, organizing the alternatives available to them, and provided a process by which they could make a better decision.

Decisions made by the end users interacting with the A/R DSS are accurately described as subjective. By supplying credible models of complex situations, and by processing data that are too tedious and voluminous for humans to accurately process manually, DSS are able to structure, organize, and clarify alternatives. In a useful DSS, the computers do what machines do best and humans do what they do best, i.e. bring judgment and creativity to the decision making process. To adequately grasp the usefulness of the DSS it is important to know more about the people who interact with it.

Characteristics of End Users

Understanding the characteristics of the end users of the accounts receivable system is critical to understanding how the DSS provided a workable solution for them. The end users of the DSS (also called decision makers throughout the paper) did not trust the computer since few of them had formal computer training and their access to computer services was limited. It was important then, that the system interface not serve as an obstacle for their use of the DSS, and that the data should be easy to enter and information meaningfully displayed.
End users’ grasp of the importance of mathematical modeling to help structure a problem was extremely limited. Their educational backgrounds and work experience afforded little formal training to this end. Users of the DSS evidenced an understandable reticence to tackle problems in anything but the most basic and straightforward form. Even use of a calculator was viewed negatively by some.

The forte of the personnel in accounts receivable was an accumulation of extensive knowledge of their customers, and an understanding of their relationships with the distribution center. This knowledge was deep and insightful. Because of it they were invaluable to decisions made in that department.

In light of these characteristics, which were observed in site visits and interviews, we felt prompted to create a prototype microcomputer-based DSS. We believed that this would be the most effective way to address several user characteristics as well as support accounts receivable decisions within one system.

The DSS, a prototype for the module eventually planned for the mainframe, helped overcome the staff’s general mistrust of computers. The screens were easy to understand and data entry was simple, since most users were already familiar with a keyboard layout. End users quickly learned that the DSS was very forgiving since there were few actions that could cause errors in their interactions with it.

Also, a DSS that used natural-language screens to effectively “hide” the mechanics of the Markov analysis helped to gain acceptance of the management science model. At the same time output from the DSS model served to structure and organize the overdue accounts problem for end users.

Use of a microcomputer-based DSS prototype supported unskilled decision makers in making complex decisions based on multiple criteria. The knowledge of customers and their specific and unique relationships with NAYM Distribution Center remained the sole, and subjective province of the decision makers. Using output from the DSS model decision makers were able to add their own insight and knowledge to decisions made on customer accounts. Results of these decisions then became input for the next interaction with the DSS.

A DSS is also defined here as an interactive, model-based computer system that allows the decision makers to learn and evolve in their decision making as they interact with the computer via a well-designed user interface (Turban, 1990). Microcomputer systems have been reported in the literature as a fast and friendly way of introducing management science models (Carlisle, et al., 1987; Kwak, et al., 1986). It was felt that the prototype microcomputer system represented an excellent proving ground for the Markov model. It permits users to assess for themselves whether the informational value of the system is worth the effort required to implement it on the mainframe.

In developing the model, the transition matrix, P, for the Markov process had to be developed first. A collection of two years of accounts receivable were tracked by month and reviewed to see if patterns might exist that would disrupt the modeling logic of the Markov process. A cursory review of the monthly transition patterns for this sample of 24 months did not discover any notable patterns in the accounts receivables that might violate the Markov assumptions for the limited purpose of the system that was being devised. (For a list of Markov assumptions see Lee et al, 1990, p. 446).

Data are collected by the procedures in place for the existing accounting module and must be entered manually into the micro A/R DSS. The NAYM report, generated by the mainframe A/R module, where the information is made available to the user (called SUMMARY A/R REPORT), is mentioned on the user’s screen. True to the DSS concepts of user friendliness and seamless incorporation of mathematical models, the prototype DSS required no user knowledge of Markov analysis.

**Alternative Actions Supported by the DSS**

After acquiring the model output, the human decision maker is free to take over. As can be seen
The first column shows what would occur if the current policy remained in force (actual data from NAYM). The second column predicts the costs for issuing stern reminders, and the third column predicts the costs (including the collection agency fee) for turning the accounts to a collection agency. In this example, the lowest total cost occurs when stern reminders are sent to overdue accounts.

When the decision maker examines each alternative, the inherent ramifications and tradeoffs can better be comprehended. Additionally, the decision process becomes more complex when it is recognized that many combinations of these actions is possible.

### Justification of the Markov Model

The anticipated downtime caused by the installation of the Markov model on the mainframe and...
the user inexperience resulting in a cost of learning time appeared to outweigh the informational benefits of the Markov A/R DSS system. To counter this perception, a Markov analysis was undertaken to demonstrate the model’s ability to project based on historic information.

Similar to the study by Kwak et al. (1986), chi-square tests were conducted to compare the actual and projected Markov values. Markov projected accounts receivables for the four stages of accounts were compared with the actual values during the four most recent months. Chi-square tests comparing the projected and the actual accounts receivables were found not to be significantly different at a 0.05 level of significance. The fact that the Markov analysis generated accounts receivable projections in each of the four stages of accounts over a four month planning period lent credibility to the model and appeared to help motivate users to use the system initially.

Users (accounting personnel) were asked to use the A/R prototype DSS for a trial period of four months. Users were asked to freely suggest improvements and make comments (as outlined by Avison & Wilson, 1991) on the use of the A/R DSS during the initial prototyping.

At the end of each of these four months the accounts receivables were tracked and tallied. The Markov-projected accounts receivable versus actual accounts receivable were then compared and presented to the users. The projected accounts receivable were so close to the actual values, a statistical comparison was deemed unnecessary. (The resulting chi-square values showed no significant difference between the actual and projected values at a level of significance less than 0.01). By providing highly accurate projections in an organized manner, the DSS freed decision makers to make fact-based decisions, in a confident manner, on how to proceed on overdue accounts. (See Schneiderjans, 1986 for a further discussion of the motivational power of accurate statistics). Decision makers had several options open to them as discussed earlier. Additionally, the prototype DSS supported planning decisions based on the projected availability of funds.

During the four-month prototyping period users offered formatting and other types of feedback that improved the input/output functions of the A/R DSS. After prototyping, the resulting A/R DSS was incorporated into the mainframe accounting module. The prototype DSS was not needed any longer, and was taken out of use.

### Observed Benefits of the DSS Prototype

During the trial period of the A/R DSS, and after its installation as a part of the mainframe accounting module, several benefits were observed and reported by end users and system developers:

1. **The microcomputer based A/R DSS permitted fine tuning of the accounts receivable module for transfer to the mainframe.** The microcomputer-based DSS was a prototype in which users and developers could quickly and inexpensively make changes in structure, formatting, and output of the system. The final version of

<table>
<thead>
<tr>
<th>Days Past Due</th>
<th>Amount Not Collectible (in Dollars)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Current Policy</td>
<td>Stern Reminder</td>
</tr>
<tr>
<td>30</td>
<td>11,350.29</td>
<td>10,132.25</td>
</tr>
<tr>
<td>60</td>
<td>12,607.40</td>
<td>9,567.81</td>
</tr>
<tr>
<td>90</td>
<td>9,334.22</td>
<td>8,123.42</td>
</tr>
<tr>
<td>120</td>
<td>5,977.44</td>
<td>4,739.75</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td><strong>39,269.35</strong></td>
<td><strong>32,563.23</strong></td>
</tr>
<tr>
<td><strong>Collection Agency</strong></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>Lost Sales</strong></td>
<td>0</td>
<td>1,307.09</td>
</tr>
<tr>
<td><strong>Total Costs</strong></td>
<td>39,269.35</td>
<td>33,870.32</td>
</tr>
</tbody>
</table>

**Table 2:** An example of the information that can be obtained from the Markov-based DSS
the A/R DSS was then converted into a mainframe module. The developers reported that the module required only two days to implement as opposed to the two weeks that had originally been estimated.

2. Users were ready to accept the module based on their satisfactory experience with the microcomputer-based A/R DSS prototype. Four months of interacting with the micro-based prototype facilitated the use of the Markov model on the mainframe. In practice, the mainframe module was even easier to use than the microcomputer system since the basic inputs (i.e., C and D vectors) were already part of the database and did not require additional manual input.

3. Users were motivated to make decisions based on output from the computerized Markov model. Prior research (Rivard, 1987) has shown that management’s directives are not enough to motivate the use of computer-based planning systems. Highly accurate projections and hands-on interaction during the four-month trial prototyping period were reported by the users as the chief motivating factors for their enthusiastic use of the Markov models.

Users viewing the prototype DSS as an experiment that did not demand commitment resulted in the perception of a less threatening requirement of change. Eventually, they felt it was worth the effort to implement the A/R Markov model on a permanent basis.

Interestingly, users found the accounts receivable projections to be useful during the prototyping period, and their interest was reflected by their request for additional information on the mathematical process of Markov analysis. The users actually requested the probability matrix, H, as they found new applications for its informational value in predicting the outcome of accounts receivables.

The only way to determine if DSS prototypes can become a major agent for implementing several kinds of decision models is to conduct further research, including other case studies. A prototype DSS on a microcomputer served here as a means of bridging the gap between the unskilled decision makers who initially resisted the idea of a model-based computer system for accounts receivable decisions and its successful incorporation into the accounting module of a mainframe.

References


We would like to thank the North American Youth Ministries (NAYM) for their assistance on this project.

Part of the work on this manuscript was completed while Julie E. Kendall was a visiting scholar at the Judge Institute of Management Studies, University of Cambridge, Cambridge CB2 1RX, U.K.

Julie E. Kendall is an assistant professor of MIS at Rutgers University. She has had her research published in MIS Quarterly, Decision Sciences, Information & Management and many other journals. In addition she has co-authored a college textbook titled Systems Analysis and Design. Currently, she is on sabbatical at the University of Cambridge.

Marc J. Schniederjans is a professor in the Department of Management at the University of Nebraska-Lincoln. He is the author of Case Studies in Decision Support Systems and Topics in Just-in-Time Management. His current research interest is in the development of management science based decision support systems.
Related Content

Studying the Translations of NHSnet
www.irma-international.org/chapter/studying-translations-nhsnet/4430/

Validating the End-User Computing Satisfaction Survey Instrument in Mexico
www.irma-international.org/chapter/validating-end-user-computing-satisfaction/18268/

A Semantically Adaptive Interface for Measuring Portal Quality in E-Government
www.irma-international.org/chapter/semantically-adaptive-interface-measuring-portal/24474/

www.irma-international.org/chapter/critical-review-online-consumer-behavior/163875/

Knowledge Super Corridors in Southeast Asia: Seeing and Doing from a Critical Lens
www.irma-international.org/article/knowledge-super-corridors-in-southeast-asia/94607/