

# An Applied DSS Course Using Excel and VBA: IS and/or MS?

William Wehrs

Department of Information Systems, University of Wisconsin - La Crosse  
La Crosse, WI 54601, USA

## Abstract

Instruction in Decision Support Systems within Information Systems curricula heretofore has not had a significant applied or hands-on emphasis. In particular, Information Systems faculty have not taken advantage of the availability of modern, Windows based software tools, such as spreadsheets, that can serve as a Decision Support Systems generator. As opposed to this, Management Science faculty have embraced Microsoft Excel as an instructional tool for quantitative modeling. Recently, this trend in Management Science teaching is being extended to include instruction in Visual Basic for Applications in a Decision Support Systems context. Over several years the author has been working to develop an applied Decision Support Systems class that employed Windows based software tools. Based on the rationale guiding efforts to incorporate Visual Basic for Applications on the Management Science side, the author developed and offered an applied Decision Support Systems class that included instruction and use of Excel Visual Basic for Applications. The structure of this class is discussed and compared with the Management Science approach in terms of the traditional components of a Decision Support System; models, data, and user interface. With respect to the issue of which discipline area, Information Systems or Management Science, should offer instruction in Decision Support Systems, the author suggests a joint effort in which the respective strengths of the students (and faculty) would complement one another.

**Keywords:** Decision Support Systems, Teaching, End User Modeling, Visual Basic for Applications, Excel, Management Science

## 1. INTRODUCTION AND BACKGROUND

Judging on the basis of textbook content (Mallach 2000; Marakas 1999; Sauter 1997; Turban and Aronson 1998), a traditional Decision Support Systems (DSS) class within an Information Systems (IS) curriculum is very survey-oriented. Students are exposed to the taxonomy of DSS with relatively little hands-on experience with the diverse software identified within the taxonomy. It is noteworthy that these circumstances prevail in spite of the widespread availability of spreadsheet software.

Spreadsheets are DSS generators with a special emphasis on the model management component of the system. A DSS generator is defined as a kit or environment for developing an application-specific DSS. In turn, a DSS has traditionally been characterized in terms of three components; a model management subsystem, a data management subsystem and a dialogue management subsystem or user interface (Sprague 1980).

In contrast to this, in Management Science (MS) modern spreadsheets have become an important instructional

vehicle for quantitative modeling. Microsoft Excel appears to be the spreadsheet of choice. The emergence of MS spreadsheet modeling is most pronounced in MBA classes. This was the focus of the first annual Summer workshop on the teaching of MS held at Dartmouth College in 1998 -- "How to Excel in MS Teaching." A summary of this workshop is available at Erhan Erkut's web site (<http://www.bus.ualberta.ca/erkut/TMSSdraft3.html>) at the University of Alberta. For further information on this trend in MS instruction see "Spreadsheet Analytics: Resources for Management Science and Operations Research in Spreadsheets", a Web site (<http://www.acs.ucalgary.ca/~grossman/resources.htm>) maintained by Tom Grossman. Among other resources, this site lists eleven MS textbooks emphasizing spreadsheet modeling.

Why has this trend evolved within MS? It has been driven by a perceived decline in the relevance or value-added for students of the MS course in MBA curricula. For further background, see the report (Available at <http://www.bus.ualberta.ca/informed/magnanti.html>) of an INFORMS sponsored task force that investigated this

issue. One outcome of this scrutiny has been the advocacy of a philosophical change with respect to the objective of the MS/OR course in business curricula. Grossman (1997) has termed this emerging philosophy "end-user Modeling." As the phrase suggests, the basic idea parallels the concept of end-user development in IS. The use of widely available spreadsheets as a modeling vehicle by the end users is a corollary of this philosophy.

A recent extension of the spreadsheet intensive approach to MS instruction has been an effort to further promote the spreadsheet capabilities of end-user modelers by providing them with instruction in Excel Visual Basic for Applications (Excel VBA). The intent of this extension is to prepare students for creating DSS using these tools. This was identified as one of three major trends coming out of the Dartmouth workshop.

Ragsdale (1998) has identified four benefits to be derived from an ability to employ VBA; leverage, automate, extend, and integrate. Leverage refers to the ability to employ a standard programming approach across the entire suite of Microsoft Office productivity tools (i.e. Excel, Access, & Word). Automate, as a benefit, is based on an efficiency criterion. VBA provides an opportunity to automate routine tasks that might otherwise be undertaken in a repetitive, interactive manner. The development of custom functions and procedures allows an extension of functionality beyond that directly provided by Excel. Lastly, integrate refers to an ability to develop a custom application (i.e. a specific DSS), including not only a model management capability, but also data management and a custom interface, within the context of the Office suite.

## 2. EXPERIMENTS WITH APPLIED DSS

Starting in 1994 I have taught the DSS course four times. From the beginning my intent has been to capitalize on the Windows operating system in a manner that captures aspects of the "integrate" benefit associated with the use of VBA. In this regard my instructional objective has been to "Employ windows data/information sharing features to integrate the use of software tools (windows applications) within a Windows DSS shell."

As this objective suggests, the course is an applied DSS class with a broad DSS survey on the front end. It is an elective course for undergraduate IS and Management majors. At the present time, the prerequisites are a sophomore level introduction to IS and a business research class that emphasizes statistics.

When I initially offered the course in 1994 the students employed Windows 3.1 on the desktop in conjunction with the Quattro Pro for Windows spreadsheet, the Paradox (for windows) database management system,

and WordPerfect for Windows. In 1996 and 1997 I employed the Microsoft Office Suite along with SPSS for windows. The emphasis in these early courses was integration based on data transfer between the respective tools. In the Summer of 1999 as I looked forward to offering the DSS class in the Fall semester, having come into contact with trends on the MS side, I was energized by the opportunity to incorporate VBA into the class from an IS perspective.

The syllabus for the course that I offered in the Fall of 1999 is available on my web site (<http://Perth.uwlax.edu/faculty/wehrs/MGT320Syllabus.html>). For a survey of DSS I had students employ the Mallach (1994) text. For review of the interactive tools of the Microsoft suite I made available a collection of modules employed in the prerequisite introduction to IS class. For Excel VBA programming I had the students use Walkenbach (1997). For my part, I did a good deal of reading in Wells and Harshbarger (1997). During the Fall I covered the following topics in the indicated order.

---

### Topics:

---

- Introduction to DSS
- Models & Modeling
- Introduction to VBA
- VBA Programming
- Spreadsheet DSS Design
- Spreadsheet DSS Construction
- Spreadsheet DSS Analysis
- Spreadsheet DSS Data Management
- Data Graphics
- Spreadsheet DSS User Interface

---

The discussion of modeling took place in terms of relatively simple algebra within a business context. There was no attempt to employ the optimization models that would be found in an MS course.

Excel VBA was introduced relatively early in the course. My approach was to encourage the students to employ the Excel macro recorder and then examine the resulting VBA code to learn how to use the objects and methods appropriate for the given spreadsheet activity. Throughout the course, exercises were employed in order to provide students hands on practice with DSS related spreadsheet functionality. VBA problems were included in these exercises.

In keeping with an IS emphasis, I spent quite a bit of time on spreadsheet DSS design. In this regard I employ the design techniques advocated by Nevison (1989) and by Amoroso(1994). These techniques are intended to mitigate some of the disadvantages of spreadsheets as modeling tools. In particular, Nevison encourages partition of the spreadsheet into separate areas, each of which has a particular purpose. The fundamental design principle in this approach is to segregate the data from the model(s) that employ it. In a

situation in which a spreadsheet contains more than one model, and in which the results of a given model enter another model as inputs, sub models are formed that are independently tested. The principle here is analogous to the modular decomposition and unit testing characteristic of structured programming. It is noteworthy that MS educators are undecided on design. At the Dartmouth workshop one unresolved question was whether to allow students to create "spaghetti spreadsheets."

The construction topic complemented the design principles. The goal of this topic was to enhance the likelihood that the spreadsheet model, when implemented, would produce computationally accurate results. Analytical activities included construction of an active area to focus attention on the relationship between model inputs and results, scenarios, sensitivity analysis, and goal-seeking.

Data management is an important DSS component, especially for data-oriented DSS. My treatment of this topic focused on the use of Microsoft Query. Microsoft Query is a data retrieval tool (i.e. ODBC browser) that can be employed within Excel 97. It allows a user to create and save queries on external relational databases for which an ODBC driver is available. In addition to Microsoft Query, Excel 97 has a powerful pivot table wizard that permits external data retrieval and flexible, summarized, tabular display of that data. Both these data management tools can be enhanced via VBA code. The pivot table has the characteristics of a DSS Online Analytical Processing (OLAP) tool. The data graphics topic serves to complement data retrieval, in that Excel charts may be used to analyze and display retrieved data or data in a pivot table.

The user interface topic is closely associated with VBA. By using VBA it is possible to customize a specific DSS to the point where no aspect of the Excel environment is visible to the user (Wells and Harshbarger 1997). In this regard, in their review of 11 non-spreadsheet DSS generators focused on decision analysis, Bhargava, Sridhar and Herrick (1999) emphasize their inadequacy with respect to the dialogue management component. The notable exception was a spreadsheet-coupled product that provided access to the spreadsheet's user interface design tools. For this course, the first time through, I emphasized the development of a simple user form and the data input validation capabilities of Excel.

### 3. OBSERVATIONS & SUGGESTIONS

With only one semester's experience employing VBA in this applied context, I have only scratched the surface of possibilities. However, there are some observations that can be made.

The IS and Management students were not strong with models or at modeling. Certainly MS students would

have a relative advantage in the model management subsystem. The infrastructure work could be done during the course at the expense of some other subject matter. Alternatively, a course with strong modeling content could be employed as a prerequisite. One possibility is to include a Production and Operations Management (POM) course that is already in the core of the business college. This alternative would be especially attractive if the POM class stressed end user modeling.

The growing popularity of OLAP, data warehousing, and supply chain management has led to increased interest in DSS development (Bhargava, Sridhar and Herrick 1999). OLAP and data warehousing tend to be associated with data-oriented DSS. Mallach's (2000) new text is indicative of this. In the preface Mallach states that the book is "intended as a textbook for a one-semester course in ... DSS, with data warehousing playing the same starring role as it does in today's decision support picture." The data retrieval tools in Excel can complement an emphasis on data-oriented DSS. Furthermore, IS students should be well prepared for this. In this area it would be very useful to have a functional data warehouse to employ for instructional purposes.

In terms of the user interface, more extensive use of VBA would be desirable. However, students need at least one course in computer programming. Even for the relatively simple VBA activities undertaken in my DSS class, the Management students had great difficulty and many of them dropped the class. It is noteworthy that over 60% of those Management students that dropped indicated some programming experience. On the other hand, the IS students should be capable of handling this.

Ragsdale (1999) advances an argument favoring DSS taught by MS faculty. The logic is as follows. First, the current demand by IT employers exceeds the capacity of IS departments to produce graduates. Second, MS techniques have been important in model management component of DSS and MS educators have a right to address this area. Third, recently IS departments have reduced DSS courses in favor of courses such as electronic commerce. His conclusion is that MS programs have the opportunity to help fill a void.

I suggest that a joint effort would be desirable. IS students need modeling work, but have expertise in data management, programming and user interface development. Based on my experience, the programming component in VBA will be a major impediment to use of this approach for non-IS students at the undergraduate level. This should be especially true with respect to the "leverage" benefit. The VBA objects and associated methods differ across the Office suite (e.g. Excel and Access). To successfully employ VBA across these tools one needs a solid grasp of the

object paradigm and some experience with control structures. Furthermore, data-oriented DSS, a form of DSS for which IS students are well prepared, is currently receiving a great deal of attention. The strengths of the IS student in terms of data management, design, programming and user interface could nicely complement the capabilities of a student who has acquired end-user modeling skills.

#### 4. ACKNOWLEDGEMENTS

The author would like to express his appreciation to Professor Cliff Ragsdale of Virginia Tech University for providing initial information that was extremely useful in implementing the DSS course used as an example in this paper.

#### 5. REFERENCES

- Amoroso, D. (1994). Decision Making Using Lotus 1-2-3 For Windows: Building Quality Applications. Mitchell McGraw-Hill.
- Bhargava, H., Sridhar, S. & Herrick, C. (1999). "Beyond Spreadsheets: Software for Building Decision Support Systems, ". Computer, March 1999, pp. 31-39. PDF at (URL <http://http://www.heinz.cmu.edu/~bhargava/Pub/Papers/DSS/dss-gen-computer.pdf>)
- Grossman, T. (1997). "End-User Modeling", OR/MS Today\_(October).
- Mallach, E. (1994). Understanding Decision Support Systems and Expert Systems. Irwin.
- Mallach, E. (2000). Decision Support and Data Warehouse Systems. Irwin McGraw-Hill.
- Marakas, G. (1999). Decision Support Systems in the Twenty-First Century. Prentice Hall.
- Nevison, J. (1989). 1-2-3 Spreadsheet Design. Brady.
- Ragsdale, C. (1998) "Should MBAs Know VBA?", OR/MS Today (December).
- Ragsdale, C. (1999). "Teaching Management Science with Spreadsheets: Where do we go from here?" Unpublished Manuscript, Virginia Tech University, March 1, 1999.
- Sauter, V. (1997). Decision Support Systems: An Applied Managerial Approach. John Wiley & Sons, Inc.
- Sprague, R. (1980). "A Framework for the Development of Decision Support Systems," MIS Quarterly, Vol. 4, No. 4.

Turban, E. & Aronson, J. (1998). Decision Support Systems and Intelligent Systems. (5<sup>th</sup> ed.) Prentice Hall.

Walkenbach, J. (1997), Excel 97 Programming for Windows for Dummies. IDG Books Worldwide, Inc.

Wells, E. & Harshbarger, S. (1997), Microsoft Excel 97 Developer's Handbook. Microsoft Press.