View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by AIS Electronic Library (AISeL)

Communications of the Association for Information Systems

Volume 9 Article 21

11-16-2002

AMCIS 2002 Panels and Workshops II: Spreadsheet-Based DSS Curriculum Issues

Cliff T. Ragsdale
Virginia Tech

Daniel J. Power
University of Northern Iowa, poer@uni.edu

Paul K. Bergey
North Carolina State University, Paul_Bergey@ncsu.edu

Follow this and additional works at: https://aisel.aisnet.org/cais

Recommended Citation
Ragsdale, Cliff T.; Power, Daniel J.; and Bergey, Paul K. (2002) "AMCIS 2002 Panels and Workshops II: Spreadsheet-Based DSS Curriculum Issues," *Communications of the Association for Information Systems*: Vol. 9 , Article 21.
DOI: 10.17705/1CAIS.00921
Available at: https://aisel.aisnet.org/cais/vol9/iss1/21

This material is brought to you by the AIS Journals at AIS Electronic Library (AISeL). It has been accepted for inclusion in Communications of the Association for Information Systems by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Communications of the Association for Information Systems

AMCIS 2002 PANELS AND WORKSHOPS II: SPREADSHEET-BASED DSS CURRICULUM ISSUES

Cliff T. Ragsdale
Department of Business Information Technology
Virginia Tech
crags@vt.edu

Daniel J. Power
Department of Management
University of Northern Iowa

Paul K. Bergey
Department of Information Technology
North Carolina State University

ABSTRACT

When challenged to justify the value of information systems (IS) research, decision support systems (DSS) is usually cited as one the most compelling examples of where IS research made the transition successfully from theoretical academic journals into the "real-world". In light of this assessment, it is somewhat surprising that offerings of DSS courses waned over the years. This paper identifies several possible reasons for the decline in DSS course offerings and suggests innovative approaches using spreadsheets for breathing new-life into this cornerstone of the IS field.

KEYWORDS: DSS, spreadsheets, DSS education

I. INTRODUCTION

Over the years, various hot-topics arose in IS to capture our research attention and compete for the limited "shelf space" in the course offerings colleges and universities are able to bring to their students. Some of the topics proved to be passing fads that never quite live up to their promise. Others made it from the drawing board of academic research into widespread acceptance and use in the business world. DSS is often cited as an example of the latter [Koch *et al.*, 2002].

From its inception in the 1970s, DSS captured the imagination of both academic researchers and industry with its promise to help business managers make better decisions to improve efficiency, increase profitability, and enhance competitiveness in an increasingly global marketplace. DSS was and remains a good idea that, when properly applied and implemented, makes significant business sense.

The topic of DSS provided a steady stream of academic research and led to the development of thousands of systems that undoubtedly saved (or made) billions of dollars for corporations around

the word. In research and practice, DSS continues to flourish. Ironically, in collegiate-level education, DSS seems to be on somewhat shakier ground.

While no objective data is readily available, informal IS curriculum analyses and information from textbook publishers suggest that over the past decade, DSS courses at many schools were sacrificed to make way for hotter topics (e.g., networking, Internet, and E-Business). Such progress in IS is both inevitable and good. However, we believe new opportunities exist for teaching DSS in ways that are better than ever and were, heretofore, nearly impossible. This paper discusses the authors' experiences at three universities in designing and teaching a spreadsheet-based DSS curriculum at the undergraduate and MBA levels.

II. THE VOID IN DSS EDUCATION

By its very nature, DSS is a broad topic, drawing on and integrating a number of technologies, disciplines, and IT skills. As a result, DSS can be a challenging course to teach. This challenge can be approached in different ways. It has been said that basically two different types of textbooks can be written for any subject:

- "All about" books, and
- "How to" books.

Books in the "All about" category tend to be more strategic in nature and provide surveys or an overview of many topics connected with a particular field. Books in the "How to" category tend to be more tactical in nature, focusing on the nitty-gritty details of how things are engineered and actually work.

Traditionally, the majority of textbooks dedicated to DSS fell into the "All about" category. Given the breadth of topics within DSS, this outcome is neither surprising or inappropriate. However, we believe that another reason for the "All about" approach in DSS texts and education is the lack of good DSS generation software for use in education.

In our opinion, "good" DSS generation software would, at a minimum, exhibit the following six characteristics:

1. *Ubiquity*. For educational purposes, it should be readily available both in the classroom and the workplace so that the skills students learn in class can also be used in their jobs.
2. *Easy to learn and use*. Obviously, software that is difficult to learn or requires knowledge of an obscure programming language is less desirable than one that is easy to learn and uses a common programming language.
3. *Highly Functional***.** It should provide easy access to a rich set of modeling and analytical tools and should integrate easily with external data and communication sources.
4. *Extendible***.** It should anticipate and provide for as much functionality as possible; however, it should also let students extend the tool's native capabilities when the inevitable happens and the user wants to do something that is not built into the tool.
5. *Integrate.* It should integrate easily with other software packages to allow students to develop custom applications using different components.
6. *Scalable***.** It should be capable of handling problems that vary from "toy sized" classroom examples to the much larger problems often encountered in the "real world".

Without the appropriate software tool, it is difficult to make the transition from "All about" to "How to" in a DSS course or textbook. DSS authors and educators faced this difficulty from the 1970s through much of the 1990s. During this time a number of DSS generation tools existed (e.g., SAS, IFPS, SPSS). However, from an educational vantage point, all of these tools (and their competitors) fell fatally short on one or more of the six criteria listed above. For example, while

SAS provides excellent functionality and scalability, it is not readily available in all educational and business organizations, and it requires knowledge of a fairly obscure, product-specific programming language.

In contrast, spreadsheets are now one of the most popular and ubiquitous software packages on the planet [Savage, 1997]. Indeed, Chan and Storey [1996] noted that managers often become so comfortable with spreadsheets that they are reluctant to adopt other software packages; even if the other packages are more suitable for specific applications. Although the basic spreadsheet row and column paradigm did not change much over the past two decades, the power and functionality underlying this simple interface grew enormously. Spreadsheets now offer a powerful modeling, analytical, and programming environment that IS educators could only dream about twenty years ago [cf., Power 2000].

III. FILLING THE VOID WITH SPREADSHEETS

In the mid-1980s, Bill Gates, founder of Microsoft Corporation, laid out his futuristic vision for "application programmability" that would leverage a PC user's knowledge of a common programming language and allow them to automate, extend, and integrate spreadsheet, database, word-processing, and other applications [Gates, 1987]. Ten years later, this vision was finally realized with the release of Office 97 which incorporated Visual Basic for Applications (VBA) as the macro language in Excel, Word, Access, and PowerPoint.

To understand the benefits of VBA, consider the characteristics of "application programmability":

- *Leverage.* Leverage refers to the ability to use a single programming language within a consistent development environment across multiple applications. As Microsoft's sole macro language, VBA provides this leverage. Microsoft also licensed VBA to hundreds of independent software vendors who integrate it as the macro language for their own products.
- *Automate*. Many of the spreadsheet tasks managers perform occur periodically or on a routine basis. VBA's ability to automate routine tasks frees managers from having to repeat routine tasks over and over and encourages them to look at 'what-if' scenarios that often lead to better decisions.
- *Extend*. In spite of Excel's many abilities, it does not do everything we might want it to do. However, with VBA it is possible to extend and customize the functionality of applications like Excel. Numerous add-ins are available that extend Excel's native capabilities to include advanced analytic and modeling techniques (e.g., optimization, simulation, artificial intelligence).
- *Integrate*. More than 500 programmable component objects in Microsoft Office allow creating custom business applications. With VBA, students can use these objects to create true decision support systems (DSS) that were previously either too costly or technically too challenging. In addition, using Microsoft's ActiveX Data Object, developers can gain access to virtually any data source outside Excel.

With the introduction of VBA as the programming language for all Microsoft Office products, Excel now supports virtually all GUI, database, modeling, artificial intelligence, data analysis, and programming tools required for creating extremely powerful and useful DSS. Given its ubiquity, we believe that Excel best embodies the six characteristics for "good" DSS generation software described in Section II. As a result, we believe that modern spreadsheets (particularly Excel) "fills the void" for those who want to transition from an "All about" to a "How to" orientation in a DSS course. The remainder of this paper discusses the authors' experiences teaching spreadsheet-based DSS courses.

IV. THE VIRGINIA TECH EXPERIENCE

For the past five years, faculty in the Department of Business Information Technology at Virginia Tech taught a spreadsheet-based DSS course that is taken by most MBA students. The course is delivered in numerous formats (e.g., the traditional classroom, on television, and on-line) and is well-received by students [Ragsdale, 2000, Zobel et al., 2000].

COURSE DESIGN

The timeline of the Virginia Tech course is shown in Figure 1.

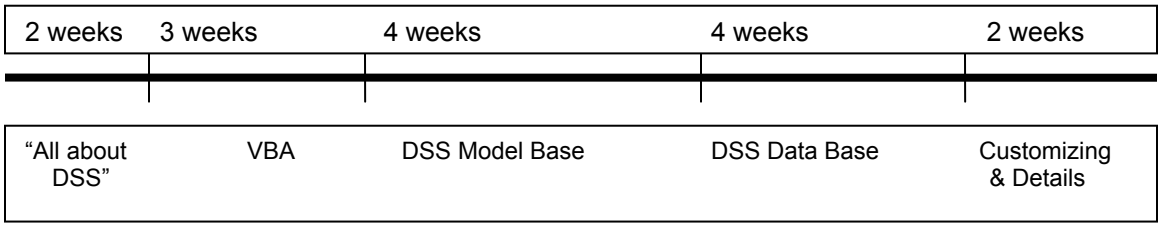

Figure 1 Course Timeline

"All about DSS"(2 weeks). The first two weeks of this course begin with a highly compressed synopsis of highlights of the "All about" material found in most traditional DSS texts. We then quickly move to a discussion of "How to" use VBA, Excel, Access and other applications to create integrated DSS. We also do a quick review of using the front-end of Excel and give an assignment to make sure everyone is up to speed with the basics of Excel.

VBA (3 weeks). Focus on issues related to VBA syntax (e.g., variables, data types, arrays, looping techniques, control structures, functions and subroutines) as well as how to record macros and use the VBA editor. To help students become used to using the editor and entering code, we give assignments that involve linking some GUI controls to some simple VBA macros (with code provided). During this time we also focus on developing an understanding of Excel's object model, particularly the Range object.

DSS Model Base (4 weeks). We illustrate how optimization, simulation, and forecasting tools can be controlled using VBA. The use of neural networks and genetic algorithms within Excel, using various add-in packages, are introduced. These weeks are a bit less VBA intensive and give those with weaker Excel and programming backgrounds a chance to digest what they have seen.

DSS Data Base (4 weeks). We spend about two weeks on the basics of relational database design, connecting to external databases from within Excel using the ActiveX Data Object (ADO), and using Structured Query Language (SQL) for simple data retrieval. Another week is devoted to more advanced features of SQL for modifying database entries and manipulating Recordset objects in ADO. We also spend a week or so discussing on-line analytical processing (OLAP) showing how to use pivot tables and pivot charts in Excel (both manually and programmatically using VBA) to display, explore, and analyze data from external databases. Various weekly assignments are provided to give students hands-on experiences with these topics.

Customizing (2 weeks). The final sessions cover issues related to "polishing" a DSS (e.g., creating custom command bars, automatically eliminating and restoring Excel's default command bars, disabling standard worksheet features). We also touch on how Excel can instantiate and control Word for use as a report generating tool using VBA concepts that are familiar to students by this point.

STUDENT PROJECTS

One of the hardest, yet fun and rewarding parts of teaching this course is helping students design, develop, and debug a required DSS development project. Many of the working students find they can apply the things we talk about in class directly to decision problems they face in the work place. In many cases, the results far exceed our hopes of what students would be able to accomplish coming out of this course. Three of these projects are described briefly below.

The Norfolk Southern Corporation (NSC). This railroad owns and maintains a fleet of over 2200 locomotives, with each engine on a six to eight year overhaul cycle. As a result, NSC overhauls approximately 450 engines a year at their maintenance shops in Roanoke, VA and Altoona, PA. One Virginia Tech MBA student was responsible for choosing NSC locomotive overhauls and spent several hours each week looking at various spreadsheets, mainframe screens, and loose pieces of paper to make this decision. This student developed a DSS that consolidates the information required for making good decisions on scheduling overhauls and displays the results in a series of tables and graphs. The user can either pick manually from a list of 100 of the best candidates or let the DSS pick the best two or three of each different engine type. The DSS interfaces to mainframe DB2 tables and allows the information to be filtered and/or sorted in a variety of ways within Excel. The DSS cut the overhaul selection time to about 15 minutes per week, freeing up his time for other tasks and allowing NSC to spend its maintenance dollars more effectively.

GE's Industrial Drives Division. Another student was a Senior Project Manager for GE's Industrial Drives Division. He developed a DSS to assist in creating consistent reports on the financial status of projects. The key critical success factors for projects include:

- target cost vs. actual cost incurred,
- the current margin on the project, and
- the areas of margin erosion which can be corrected by the business to improve the profits on the job.

The first step in this DSS was to use SQL and ODBC to link the DSS to one of the GE systems to download the baseline sales and cost data from the original contract. A variety of data entry sheets then allow project managers to enter additional sales and cost data (such as any items added to the job since the original contract) and notes about the project. Various graphs can then be generated with the click of a button to show target vs. actual costs, project contribution margins, and a margin erosion graph (or pricing waterfall). This DSS is now in use by GE Industrial Control Systems project managers on all projects in excess of $3 million dollars. Several "cost reduction" projects were started as a result of the price waterfall data graphs. This student's supervisor also nominated this project to receive a "best practices" award within GE.

Carilion Health Systems (CHS). This not-for-profit healthcare system provides services to approximately one million people in western Virginia. CHS continually evaluates physician supply and demand within their service areas to identify communities where there is an inadequate supply of physician specialties and target those areas for initiatives to improve access. This student developed a DSS to access various physician and population databases to calculate the "net need" for various medical specialties by county, city, or zip level in the regions CHS serves. The program also includes a feature that allows the user to determine quickly the supply of any specialty in any location by clicking a command button that constructs a pivot table based on the data in the physician table. This information provides support in decisions about new programs, program expansions, and physician recruiting. Hospitals that use "income guarantees" as a tool to attract new physicians must be able to demonstrate that they are legally defensible under Medicare fraud and abuse statutes and IRS laws. This DSS provides such evidence for CHS.

CHALLENGES

In an MBA program, the disparity in students' experience with Excel and/or programming can be a concern. The instructor really needs to start from the ground up and cover the "basics" for those with little or no experience while, at the same time, not boring those with greater experience. However, as Don Plane (a pioneer in management education using spreadsheets) is fond of saying, "Most students who report having 5 years of experience with Excel actually have the equivalent of six months of experience repeated ten times." We found Plane's analysis to be absolutely true. Furthermore, regardless of their previous experience with Excel, most students have little experience with VBA and are easily amazed by its capabilities. We also discovered that most of those with significant programming experience had little or no experience using VBA to manipulate objects within Excel. As a result, the "ground up" presentation proved to be interesting enough to ward off boredom.

V. THE UNIVERSITY OF NORTHERN IOWA EXPERIENCE

For many years the Decision Support Systems course for undergraduate MIS majors at the University of Northern Iowa (UNI) in Cedar Falls, IA was a survey course that included using demonstration versions of some PC-based software packages like IFPS and EXSYS. The course design included a group meeting session using GroupSystems, a decision analysis project using Expert Choice, and a small scale development project. The development project was always a problem because students tried to use a variety of software development environments and it was difficult to provide support, instruction, and assistance for all of the development packages. Some of the development projects were a failure from the perspective of the student developers. This course design was not a success. In an effort to create a course that provided students with a more uniform set of skills, the software focus began shifting to using a spreadsheet package as a development environment in the late 1990s.

Our MIS major includes Decision Support Systems as an elective. Because of the small number of majors it is only offered once a year. Some business students also take a second course in MIS that is sometimes structured as a DSS course. For a few years the MBA MIS course was also taught as a DSS course. The same course design didn't work for any of the three audiences.

ALTERNATIVE COURSE DESIGNS

Gradually three introductory DSS courses evolved.

1. The most technical, skills-oriented course is for undergraduate MIS majors who may want to work as DSS developers and analysts. These students need to learn about a variety of DSS applications and development tools. This course surveys DSS concepts and resources and helps students learn Microsoft Excel and the VBA development environment. Students complete a spreadsheet-based DSS project. This course is designed as about 40% concepts and 60% skills. Every week the students do exercises to develop Excel and VBA skills. Student teams propose a spreadsheet-based DSS project, prepare a feasibility study, and implement the project.

2. In the DSS course for non-MIS majors the emphasis shifts away from advanced skills and DSS development to using DSS for analysis. These students may work as decision support or business analysts in staff support roles. The course focuses on using DSS and spreadsheets to prepare special studies and reports. Students also complete a structured exercise to create a small-scale spreadsheet-based DSS. The course is about 50% content and 50% skills, but the students don't develop the same skill level with Excel. The students begin with more limited knowledge of Excel, programming, and systems development and that constrains what can be expected of the students.

3. The third type of DSS course provides a DSS orientation and DSS user perspective for future managers. This course is taught at the MBA level. This DSS course needs more "managerial content" but it also incorporates Excel exercises to help students understand data and model-

driven DSS better. Some students develop small-scale spreadsheet-based DSS, but students can choose to prepare a feasibility study as an alternative to developing an actual application.

COURSE OBJECTIVES

All three course designs share common course objectives. A major shared objective is to review and clarify the fundamental terms, concepts, and theories associated with Decision Support Systems, including Communications-Driven and Group Decision Support Systems (GDSS), Data-Driven DSS, Model-Driven DSS, Document-Driven DSS, and Knowledge-Driven DSS. A second shared objective is to discuss and develop skills in the analysis, design and implementation of computerized Decision Support Systems. A third shared objective is to examine user interface design issues and evaluate the user interfaces and capabilities of Decision Support Systems. Fourth, all of the course designs emphasize developing and improving hands-on skills. Finally in all of the courses it is important to discuss organizational and social implications of Decision Support Systems.

In the course for MIS majors, hands-on skills using Microsoft Excel is particularly important. In the course for MBA students it is especially important to examine examples and case studies documenting computer support for organizational decision-making. In all three courses the focus in on spreadsheet-based DSS built using Microsoft Excel and Web-based DSS examples. All three courses emphasize hands-on computer labs. The DSSResources.COM website includes materials used in the courses. Also, the Resources for Professors page includes links to example syllabuses and course materials.

CHALLENGES

A major challenge is course materials. A book *Decision Support Systems: Concepts and Resources for Managers* [Power, 2002] evolved from trying to organize the content for teaching a broad survey of DSS. The exercises and laboratories are an on-going challenge. The laboratories should teach students skills associated with DSS and skills associated with Excel and VBA. Some of the laboratory exercises used include: a DSS user interface design exercise, a data cleaning and loading exercise, a pivot table exercise, a solver DSS exercise, a break-even analysis exercise, and a data analysis exercise. A number of Excel books were used for instruction and reference.

A second major challenge is grading and testing. With undergraduates, especially non-MIS majors, it can be difficult to interest them in the concepts of DSS. The technical terminology is often difficult for them. Multiple choice and short answer tests are not very satisfactory in capturing how well the students understand the DSS materials. Adding essay or mini-case example questions provides more insight about the students' understanding.

A final challenge with the MIS majors is guiding and shaping the spreadsheet-based DSS projects they select. In general, students should be discouraged from choosing broad scope projects. They need encouragement to use the many capabilities of Excel and VBA.

VI. THE NC STATE EXPERIENCE

A spreadsheet-based DSS course was recently introduced at North Carolina State University in the College of Management as an elective in the MBA curriculum. It is designed for those seeking a concentration in MIS. The original design of the course contained excerpts of both the "All About" and "How to" flavors described previously. Because the course scheduling required meeting for 2.5 hours on a single evening during the week, the environment of the course consisted of an initial lecture containing mostly the "All About DSS" type of material, followed by a laboratory session where students practiced hands on development using VBA in Excel. As the course progressed (and as the students began to recognize the value of their newfound programming skills), the general mood of the student body clearly leaned towards furthering their

technical thirst for knowledge (via programming skills) over their curiosity for knowing additional "All About" (or more about) content. Thus, driven by popular demand the second half of our debut DSS course shifted to focusing full time on development programming skills using VBA in Excel.

The greatest challenge experienced in the NC State DSS course thus far is dealing with the tremendous variance in skill sets of the students enrolled in the course. Since NC State is located in Raleigh, NC, home to many Fortune 500 companies, the background of the typical MBA student can range from professional programmer to a novice who never wrote a line of code. For this reason, a semester project is performed individually by each student where the technical level of difficulty is established by a proposal that must be submitted and approved. The proposal includes a description of the students' background skills and establishes clear objectives that are consistent with their unique skill set. A debriefing survey of the students enrolled in the DSS course indicate that "updating their technological skill" is a primary objective in returning to graduate school. They also strongly indicated that learning VBA in the DSS course satisfied their objective of updating their technological skills.

VII.CONCLUSIONS

Spreadsheets are the standard lens through which most managers view analytical information about their organizations. Although Panko [1998] (and others) voice concerns about error rates in spreadsheet applications, recent developments in spreadsheet technology (e.g., error checking smart tags, automated data validation checking, and formula auditing utilities) provide new tools to help reduce or eliminate these errors. While spreadsheets may not be the most powerful or appropriate tool for all DSS applications, we cannot confidently assume students will have access to any other software package in the workplace. As a result, spreadsheets (in particular, Microsoft Excel) offers a premier platform for teaching the "How to" aspect of DSS. The outstanding results we (and others) witnessed using this approach suggests that it is a pedagogical direction that other educators in the IS community should consider seriously.

Editor's Note: This article is based on the authors' panel at AMCIS 2002 in Dallas, Texas. It was received on September 13, 2002 and was published on November 16, 2002 in a special issue of CAIS together with other articles from the panels and workshops. The special issue was under the editorship of Leslie D. Ball.

REFERENCES

Chan, Y.E. and V.C. Storey (1996), "The Use of Spreadsheets in Organizations: Determinants and Consequences," *Information & Management*, (31) pp. 119-134.

Gates, B.(1987) "Beyond Macro Processing: A Strategy for Customizing Applications Software," *BYTE*, (12)7, pp 11, 1987.

Kock, N., P. Gray, R., Hoving, H. Klein, M.D. Myers and J. Rockart (2002), "IS Research Relevance Revisited: Subtle Accomplishment, Unfulfilled Promise, or Serial Hypocrisy?," *Communications of the Association for Information Systems*, (8)23, March.

Panko, Raymond R. (1998), "What We Know About Spreadsheet Errors," *Journal of End User Computing*, (10)2, pp.15-21, Spring. (Also available at http://panko.cba.hawaii.edu/ssr/Mypapers/whatknow.htm.)

Power, D.J. "A History of Microcomputer Spreadsheets" *Communications of the Association for Information Systems* (4)9 pp. 154-162.

Power, D.J. (2002) *Decision Support Systems: Concepts and Resources for Managers*. Westport, CT: Greenwood/Quorum Books.

Ragsdale, C. (2000), "Teaching Management Science Using Spreadsheets: From Decision Models to Decision Support," *INFORMS Transactions on Education*, (1)2, pp. 68-76.

Savage, S.(1997), "Weighing the Pros and Cons of Decision Technology in Spreadsheets," *OR/MS Today*, (24)1, pp. 42-45.

Zobel, C., C. Ragsdale, and L. Rees (2000), "Teaching Decision Support Systems Using VBA," *OR/MS Today*, (27)6, pp. 28-31.

APPENDIX. DIRECTORY OF SOFTWARE PRODUCTS MENTIONED IN THIS ARTICLE

Product	Description
Excel	Electronic spreadsheet software. Web site: www.microsoft.com/excel
SAS	Originally conceived as a statistical analysis system (SAS), this product evolved to provide comprehensive data access, data analysis, data management, and data presentation to support decision making. Web site: www.sas.com
IFPS	Interactive Financial Planning System (IFPS) is a business modeling language that uses English-like or natural language formulas to describe the finance and accounting logic of a financial plan. It was discontinued by its vendor several years ago.
SPSS	Originally conceived as a statistical package for the social sciences (SPSS), this product now offers comprehensive data mining technology and analytic solutions to enhance decision making. Web site: www.spss.com
EXSYS	A system for the development and deployment of expert systems. Web site: www.exsys.com.
GroupSystems	A software product originally designed at the University of Arizona to support group idea generation, collaboration, voting, progress tracking and decision making. Web site: www.groupsystems.com.
Expert Choice	Based on the Analytic Hierarchy Process (AHP), Expert Choice helps groups to structure their objectives into a decision model, prioritize using pairwise comparisons, and justify decisions using graphical reports and sensitivity analyses. Web site: www.expertchoice.com.

ABOUT THE AUTHORS

Cliff T. Ragsdale is Professor and Director of the Dominion Center for Energy Modeling and Optimization in the Department of Business Information Technology at Virginia Tech. He received his Ph.D. in Management Science and Information Technology from the University of Georgia. Dr. Ragsdale's primary areas of research interest include microcomputer systems and applications, artificial intelligence, mathematical programming and applied statistics. His research appears in *Decision Sciences, Decision Support Systems, Naval Research Logistics, OMEGA, Computers & Operations Research, Operations Research Letters, Personal Financial Planning, Engineering Applications of Artificial Intelligence*, and other publications. He currently serves on the Advisory Boards of Computers and Operations Research, INFORMS Transactions on Education, and the International Journal of Information Technology & Decision Making. He is also author of the textbook *Spreadsheet Modeling and Decision Analysis* published by South-Western.

Daniel J. Power is Professor of Information Systems and Management at the College of Business Administration at the University of Northern Iowa, Cedar Falls, Iowa. He holds a Ph.D.

in Business Administration from the University of Wisconsin-Madison. His research interests include the design and development of Decision Support Systems and how DSS impact individual and organizational decision behavior. His research has appeared in *MIS Quarterly, Decision Sciences, Journal of Decision Systems, Academy of Management Review, and Information and Management*. He is also author of the book *Decision Support Systems: Concepts and Resources for Managers* published by Quorum in 2002. Professor Power is Editor of DSSResources.com and the bi-weekly DSS News e-Newsletter. He is also the Section Editor of the ISWorld pages on Decision Support Systems and Chair of the Association for Information Systems Special Interest Group on Decision Support, Knowledge, and Data Management Systems (SIG DSS).

Paul K. Bergey is Assistant Professor in the College of Management at North Carolina State University. He received is Ph.D. in Management Science and Information Technology from Virginia Tech. His research interests center on the use of mathematical models and computers to support decision making. He published in *Decision Support Systems*.

Copyright © 2002 by the Association for Information Systems. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation on the first page. Copyright for components of this work owned by others than the Association for Information Systems must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or fee. Request permission to publish from: AIS Administrative Office, P.O. Box 2712 Atlanta, GA, 30301-2712 Attn: Reprints or via e-mail from ais@gsu.edu .

Communications of the Association for Information Systems

ISSN: 1529-3181

EDITOR-IN-CHIEF
Paul Gray
Claremont Graduate University

AIS SENIOR EDITORIAL BOARD

Cynthia Beath Vice President Publications University of Texas at Austin	Paul Gray Editor, CAIS Claremont Graduate University	Sirkka Jarvenpaa Editor, JAIS University of Texas at Austin
Edward A. Stohr Editor-at-Large Stevens Inst. of Technology	Blake Ives Editor, Electronic Publications University of Houston	Reagan Ramsower Editor, ISWorld Net Baylor University

CAIS ADVISORY BOARD

Gordon Davis University of Minnesota	Ken Kraemer Univ. of California at Irvine	Richard Mason Southern Methodist University
Jay Nunamaker University of Arizona	Henk Sol Delft University	Ralph Sprague University of Hawaii

CAIS SENIOR EDITORS

Steve Alter U. of San Francisco	Chris Holland Manchester Business School, UK	Jaak Jurison Fordham University	Jerry Luftman Stevens Institute of Technology

CAIS EDITORIAL BOARD

Tung Bui University of Hawaii	H. Michael Chung California State Univ.	Candace Deans University of Richmond	Donna Dufner U.of Nebraska -Omaha
Omar El Sawy University of Southern California	Ali Farhoomand The University of Hong Kong, China	Jane Fedorowicz Bentley College	Brent Gallupe Queens University, Canada
Robert L. Glass Computing Trends	Sy Goodman Georgia Institute of Technology	Joze Gricar University of Maribor Slovenia	Ruth Guthrie California State Univ.
Juhani Iivari University of Oulu Finland	Munir Mandviwalla Temple University	M.Lynne Markus Bentley College	Don McCubbrey University of Denver
Michael Myers University of Auckland, New Zealand	Seev Neumann Tel Aviv University, Israel	Hung Kook Park Sangmyung University, Korea	Dan Power University of Northern Iowa
Nicolau Reinhardt University of Sao Paulo, Brazil	Maung Sein Agder University College, Norway	Carol Saunders University of Central Florida	Peter Seddon University of Melbourne Australia
Doug Vogel City University of Hong Kong, China	Hugh Watson University of Georgia	Rolf Wigand University of Arkansas	Peter Woolcott University of Nebraska-Omaha

ADMINISTRATIVE PERSONNEL

Eph McLean AIS, Executive Director Georgia State University	Samantha Spears Subscriptions Manager Georgia State University	Reagan Ramsower Publisher, CAIS Baylor University