USER SATISFACTION AND SYSTEM SUCCESS: CONSIDERING THE DEVELOPMENT TEAM

Julie Fisher
School of Information, Management and Systems
Monash University
Melbourne Australia
Julie.Fisher@sime.monash.edu.au

ABSTRACT

Two key indicators of system success are that the software performs as intended and the users are satisfied with the system. Frequently, however, the major and overwhelming focus of the development team is on building and refining the technical side of the system. The result is that many systems are delivered where the users are not satisfied. This paper explores the skills, other than technical skills, that are needed on development teams to produce systems that are successful from a user's perspective.

Keywords: design, human factors, information systems development, user expectations, system success

INTRODUCTION

Early computer systems were built mainly for scientific purposes and often built by those who would ultimately use them. There was little need to explain how the systems worked because those who used them had also programmed them (Avison and Fitzgerald 1995, 17). Grudin (1991) notes that as the costs of computers fell and user numbers increased, expectations of usability also grew. He argues that the users needs are not met because the information about the users' needs takes time to get through to the developers (Grudin 1991, 61).

With the growth of e-commerce systems comes a need for development teams to pay more attention to the usability aspects of these systems. The audience that may use an e-commerce system is frequently unknown. Further, e-commerce systems must cater for people who often have little knowledge or interest in the technology. System development teams must deliver systems that are innately intuitive. This paper will argue that there are two sides to producing an effective information system, the technical /functional side and the human side. Frequently systems development teams are lacking in people with skills that focus on the human factors aspects of systems and therefore the systems produced are not successful because the users are not satisfied with them.

SYSTEMS DEVELOPMENT METHODOLOGIES

Avison and Fitzgerald (1995) note that although there are many hundreds of systems development methodologies, many are very similar. Only a small number of basic and well understood methodologies are widely used in the IT development community. The most common methodology is the traditional approach (also known as the waterfall method or systems development life cycle (SDLC)). Many of the newer development methods are a variation on the traditional approach (Avison and Fitzgerald 1995; Rouse, Watson et al. 1995). Its acceptance as a widely employed methodology is reflected in the number of computing text books referring to it and the number of courses that teach the traditional methodology (Nicholas, 1990; Gibson and Huches, 1994). Figure 1 is a diagrammatic representation of the traditional approach to systems development and includes the skilled roles normally involved at the different stages.

The SDLC, because it is still the development methodology most widely used, has been selected to illustrate how and where current practice in developing systems is not meeting the needs of the users. The most commonly recognised roles in the development process include the client or project sponsor, system analyst, managers, developers (or programmers), and users (Nicholas 1990; Avison and Fitzgerald 1995).

SKILLS

Each member of a development team brings to the process a skill set relevant to the activity or role they will perform. Table 1 describes the key skills or knowledge we would expect each of the team members normally associated with systems development, to possess.
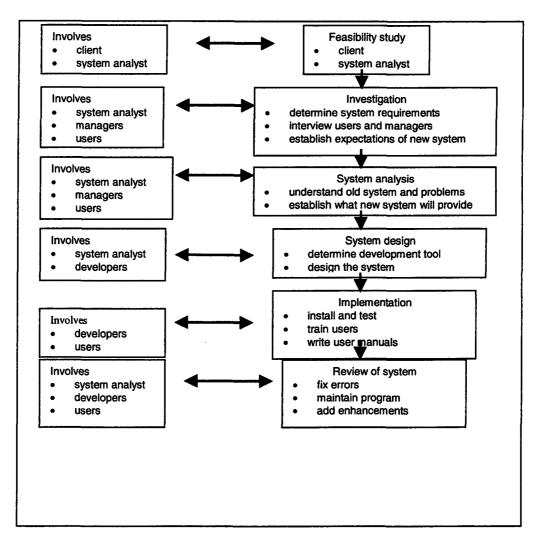
Figure 1 Traditional Systems Development Life Cycle

Team member	Skills /knowledge
Client (system sponsor)	Knowledge of the business and business process. Understanding of what the new system should provide
Project manager/ team leaders	Management (people and financial) skills Verbal and interpersonal communication and listening skills Technical competence
System analysts	Communication skills Analytical skills Technical skills Problem solving
Developers (programmers)	Programming Analytical skills Technical skills System testing Problem solving

Table 1 Skills of the development team

The basis of this information comes from a range of sources including (Nichols 1994, 174-180; Simon 1994; Misic 1996; Turner and Lowry 1999). There are of course many other skills that are needed however this list is restricted to those skills specific to development of an information system as distinct from management or other

skills. It should be noted that the skills listed in Table 1 are also the primary skills taught in Australian university information systems courses and are recognised by the Australian Computer Society (Underwood and Maynard 1996).

SYSTEM SUCCESS

There are numerous reasons proffered as to why systems are not successful, amongst these are:
- Inadequate management planning and control (McComb and Smith 1991; Weinberg 1991)
- Poor management of time and budget (McComb and Smith 1991)
- Problems with implementation (Myers 1993)
- Poor systems quality and performance (McComb and Smith 1991; Ballantine, Bonner et al. 1996)

In a detailed assessment, Delone and McLean identified six major categories for measuring system success, these are: system quality, information quality, use, user satisfaction, individual impact and organisational impact (DeLone and McLean 1992). This paper will focus on those less obvious factors that directly relate to the use of the system, that is: user satisfaction and individual impact of use. These factors are those that impact most on users and relate to the human factors aspects of systems.

Two key measures of success from the users' perspective are:
- Meeting user expectations. A system may be functionally correct but if it does not meet user requirements or expectations fails. Grudin made an important point when he said:

  There is a strong consensus that intuition and indirect approaches to understanding users and their work are usually insufficient. The design principles formulated by Gould and his colleagues at IBM. are: (a) focus early and continuously on users, (b) integrate consideration of all aspects of usability, (c) test versions with users early and continuously, and (d) iterate the design. Despite being widely cited, these principles are not often followed. (Grudin 1991)

- Effective human factors elements. Poor communication between users and developers often leads to deficient analysis and poor usability. The technologists are unable to communicate to the users what the system will look like and the users are unable to explain to the technologists what is they want (Grudin 1991; Bresko 1995). Bresko (1995) cites communication problems as a major cause of system failure and argues that development methods and management techniques are the cause of many of the problems.

Failing to meet the needs of the users and poor communication between system developers and users typically result in the failure of the human element or human factors aspects of the system; that is, the user is unable to effectively use the system. The majority of development methodologies - including the traditional method, describe in detail how and when the technological aspects of a system will be dealt with. They however, rarely mention how and when the human elements of a system are designed. Avison and Fitzgerald make the point that:

Although not simple, the technological aspects are less complex than the human aspects in an information system, because the former are predictable in nature. However, many information systems methodologies only stress the technological aspects. This may lead to a solution which is not ideal because the methodologies underestimate the importance and complexity of the human element. (Avison and Fitzgerald 1995, 41)

This is a significant and expensive problem for the IT industry. A contributing issue, is that those on the development team usually responsible for the usability aspects of systems and are not the people with the most appropriate skills.

DEFINING A SUCCESSFUL SYSTEM – THE USERS' PERSPECTIVE

From a technologist's perspective the key elements that define a successful system would be: a system that is functionally correct, provides accurate information, operates at an optimum speed relative to the technology it is run on and meets the needs of the organisation. This is the technical and functional side. From a user's perspective however, unless the user can use the system and use it effectively and fully, the system cannot be deemed to be a success. This is the human side. If the user cannot use the system then it must be regarded as a failure.

The question is then, what are the elements of an information system that make it successful from a user's perspective? The literature suggests a number of factors that contribute strongly to users rating a system as successful. The elements presented in Table 2 are those that relate just to the development process prior to implementation.

Factor	Authors / research
User expectations of a system	
Task–technology fit	(Hirschheim and Newman 1988; Lyytinen 1988; Goodhue and Thompson 1995)
Understanding the user's perspective. The system reflects the cognitive style and mental model of the users	(Geriach and Kuo 1991; Kubie, Melkus et al. 1996; Williams 1996)
Meeting user expectations or perceptions of the system.	(Lawrence and Low 1993; Szajna and Scamell 1993; Ballantine, Bonner et al. 1996; Jordan and Burn 1997)
System usability	
Quality and effectiveness of the interface design	(Doll and Torkzadeh 1988; Mantel and Teorey 1989; Kay 1990; Norman 1990 B; Geriach and Kuo 1991; Gould, Ukelson et al. 1996)
Quality and effectiveness of user documentation and information	(Doll and Torkzadeh 1987; Geriach and Kuo 1991; Torkzadeh and Doll 1993; Kekre, Krishnan et al. 1995; Coe 1996)
Ease of use	(Doll & Torkzadeh 1988; Kekre et al. 1995; Kubie et al. 1996;)
User acceptance and ownership of a system	
Level of user involvement in the development process	(Lawrence and Low 1993; Szajna and Scamell 1993; Goodhue and Thompson 1995; Robert 1997)
Participation in the development leading to greater commitment	(Hirschheim & Newman 1988; Lawrence & Low 1993; Robert 1997)
The extent of user involvement and participation	(Montazemi 1988; Barki and Hartwick 1989; Lawrence and Low 1993)
The quality of user–developer communication.	(Szajna and Scamell 1993; McKeen, Guimaraes et al. 1994; Robert 1997)
The quality of the system and system reliability	(Doll & Torkzadeh 1988; Kekre et al. 1995)

Table 2 User satisfaction factors

Figure 2 summaries these elements in diagrammatic form.
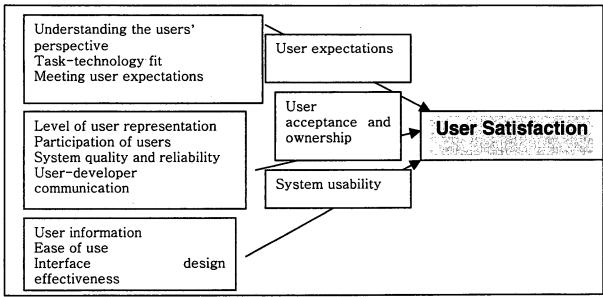

Figure 2 User satisfaction factors

Drawing on the factors identified in the literature as important measures of user satisfaction, a number of key elements that are within the control of the development team have been identified. These factors are:

- The design of the user interface

- The match between the users' work-flow and the design of the system.
- The provision of effective user information both online and offline
- The effectiveness and usability of the system and error messages
- The quality of the communication between users and developers that is, how well the developers understood the needs of the users and how well the users were able to communicate their needs to the developers.
- How well the users' perspective was understood.
- How effectively the users were able to participate in the development process.

Human factors skills

Following on from this: if the factors described above are important in determining the success of a system from a users' perspective, what skills are needed to design a system that has regard for these factors? Some of the key skills needed for developing the human elements of a system are:

- Written and oral communication skills
- Understanding of users and how they work
- Organisation of information
- Design and layout of information
- Graphic design and illustration skills
- Interface design skills
- Understanding of how people work
- Understanding of users' work-flow. (Shand 1994 B; Avison and Fitzgerald 1995; Bresko 1995)

Table 1, listed the skills generally recognised as those sought by a project manager forming a team. There is therefore a mismatch between the skills required to meet the needs of the users, as described above, and the skills required for developing the technical aspects of a system. The skills we teach to information systems students generally are the traditional, technical skills as described in Table 1.

Researchers such as Bresko (1995), Shand (1994) and Avison and Fitzgerald (1994) acknowledge a high rate of systems failure resulting from the failure of the human aspects of a system . Development teams however rarely include people with the skills necessary to effectively design the human factors elements of systems.

OTHERS WHO CAN CONTRIBUTE TO THE DEVELOPMENT PROCESS

To produce a system that is both technically correct and satisfies the needs of the users a wider range of experts need to be included on development teams and included early. Table 3 lists some of other experts who can and should make a contribution to the development of an information system.

Profession	Skills	Contributes to
Graphic designers	Illustration Graphic design	Interface design including icons
Technical communicators (professional writers)	Written and oral communication skills Organisation of information Understanding of user perspective Illustration Semiotics	User information including error messages. Determining user requirements Development of menus and other information structures Interface design
Cognitive psychologists / Human Factors experts Usability experts	Understanding of how users work User centred design skills	Interface design Work-flow Overall usability User task analysis Ergonomic design

Table 3 Other professionals and their skills

Whilst the role of these other experts is acknowledge in the literature as being of value to the development process they are rarely included in IT teams (Avison and Fitzgerald, Grudin (1991 A)., Shand, McComb and Smith (1991))

Graphic designers

Graphic designers bring to the development process an understanding of what works in terms of illustrations, icons, colours and fonts for example. They also understand the layout and design of graphics and aesthetic effectiveness. Selecting the most appropriate media for the task, photographs or illustrations, sound or no sound. Knowing how to place graphics on the screen to be most effective. Graphic designers understand topography, how text should look to be most accessible to the user.

Technical communicators

Technical communicators are professional writers. They bring to a development team specialist writing skills. Writing material for users is a specialist skill particularly when it comes to online help. Online help and error messages have to be brief, written in an accessible language and be effective, the se are the skills of a specialist writer. Technical communicators also understand about how to layout text so that it is easy to read and attracts the eye of the reader, they know about typography that is how the written text looks on the screen. Previous research by the author statistically demonstrated that:

- Users were more satisfied with systems where a technical communicator was involved.
- Users rated systems more successful when a technical communicator was involved
- Users used online help more frequently and rated the quality higher when it was written by a technical communicator. (Fisher 1999A; Fisher 1999B)

Human factors experts – Computer Human Interaction (CHI)

The field of human factors is well established and professionals in the field work on a range of every day items whose design we would not think twice about. Those working in the area of human factors are involved in designing for usability. Norman a leading expert in the field in 1988 wrote 'The Psychology of Everyday Things' this book continues to be one of the leading texts in the area. The field of Computer Human Interaction is an extension of the work of human factors people but specialises in the design of computer systems. Their interest and expertise is in usability, ensuring that users are able to interact effectively with a system.

DISCUSSION

Traditional methods for developing systems typically do not consider the involvement of other, non-technical professionals, in designing systems. Current practice has changed little and yet the number of non-technical people exposed to computer systems, is growing exponentially as a result of the world wide web. There is a need to include on to the development team, people who have the expertise and knowledge of how users work and think. Figure 3 defines an information system in terms of these two identified sides, the technical /functional side and the human factors side.
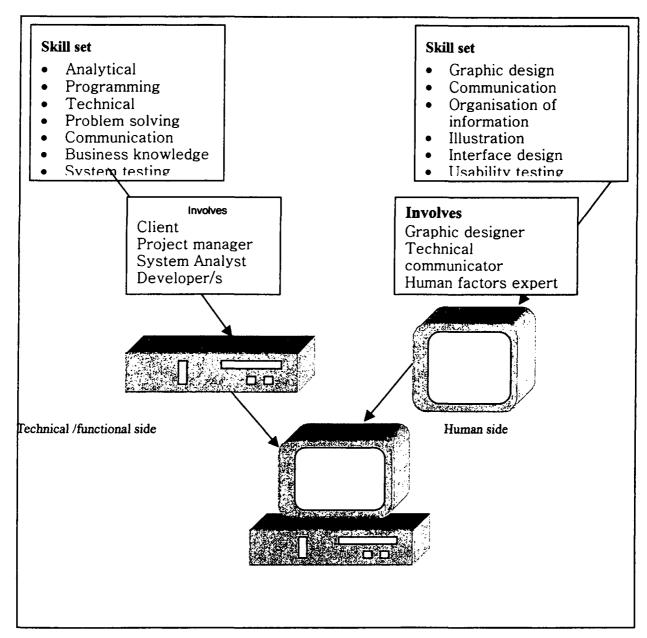
Skill set

- Analytical
- Programming
- Technical
- Problem solving
- Communication
- Business knowledge
- System testing

Skill set

- Graphic design
- Communication
- Organisation of
  information
- Illustration
- Interface design
- Usability testing

Involves

Client
Project manager
System Analyst
Developer/s

Involves

Graphic designer
Technical
communicator
Human factors expert

Technical /functional side						Human side

Figure 3 Skills need to produce a successful system from the user's perspective

CONCLUSION

What I have attempted to do in this paper is to argue that there is a gap between the skills of the people usually involved in the development of an information system and the skills that are needed to develop an effective usable system. If we want to deliver to users the most effective system then consideration has to be given to expanding development teams to include those with skills that fill the current gaps. What we have today on development teams are people who are technically competent but have limited knowledge, understanding or skills of how to develop a system for non-technical users. We should not expect that those with the technical skills also have skills in graphic design, technical communication or cognitive psychology. These are specialist fields in their own right. But if we want systems, and in particular e-commerce systems, that can be used by anyone then we need people with both the technical /functional skills as well as skills to ensure the system is effective and usable.

REFERENCES

Avison, D. E. and G. Fitzgerald (1995). **Information Systems Development: Methodologies, Techniques and Tools**. London, The McGraw-Hill Companies.

Ballantine, J., B. Bonner, et al. (1996). "The 3-D Model of Information Systems Success: the Search for the Dependent Variable Continues." **Information Resources Management Journal** (Fall): 5-14.

Barki, H. and J. Hartwick (1989). "Rethinking the Concept of User Involvement." **MIS Quarterly** 13(1) 59 - 82

Bresko, L. L. (1995). The Need for Technical Communicators on the Software Development Team. **Perspectives on the Profession of Technical Communication**. M. Flammia. Florida, Society for Technical Communication. **1**: 213-217.

Coe, L. (1996). "Five Small Secrets to Systems Success." **Information Resources Management Journal** (Fall): 29-38.

DeLone, W. and E. McLean (1992). "Information Systems Success: The Quest for the Dependent Variable." **Information Systems Research** 3(1): 60-95.

Doll, W. and G. Torkzadeh (1987). "The Quality of User Documentation." **Information and Management** 12: 73-78.

Doll, W. and G. Torkzadeh (1988). "The Measurement of End-User Computing Satisfaction." **MIS Quarterly** 12(2): 259-274.

Fisher, J. (1999A). "The Value of the Technical Communicator's Role in the Development of Information Systems." **IEEE Transactions on Professional Communication** 42(3): 145-155.

Fisher, J. (1999B). "Improving the usability of information systems: the role of the technical communicator." **European Journal of Information Systems** 8(4): 294-303.

Geriach, J. and F.-Y. Kuo (1991). "Understanding Human-Computer Interaction for Information Systems Design." **MIS Quarterly** 15(4): 527-549.

Gibson, M. L. and C. T. Huches (1994). **Systems Analysis and Design: A Comprehensive Methodology with Case**. United States, Boyd & Fraser publishing company.

Goodhue, D. and R. Thompson (1995). "Task-Technology Fit and Individual Performance." **MIS Quarterly** 19(2): 213-236.

Gould, J., J. Ukelson, et al. (1996). Improving User Interfaces and Application Productivity by Using the ITS Application Development Environment. **Human-Computer Interface Design**. M. Rudisill, C. Lewis, P. Polson and T. McKay. San Francisco, Morgan Kaufmann: 173-197.

Grudin, J. (1991A). "Interactive Systems: Bridging the Gaps Between Developers and Users." **IEEE Transactions on Computers** April: 59-69.

Grudin, J. (1991B). "Systematic Sources of Suboptimal Interface Design in Large Product Development Organizations." **Human Computer Interaction** 6(2): 147-196.

Hirschheim, R. and M. Newman (1988). "Information Systems and User Resistance: Theory and Practice." **The Computer Journal** 31(5): 398-408.

Jordan, E. and J. Burn (1997). Information Systems Acceptance: A Consequence of Cultures. PACIS '97, Brisbane, Information Systems Management Research Concentration, Queensland University of Technology, Australia 451-459

Kay, A. (1990). User Interface: A Personal View. **The Art of Human-Computer Interface Design,**. B. Laurel. USA, Addison Wesley Publishing Comp: 191-207.

Kekre, S., M. Krishnan, et al. (1995). "Drivers of Customer Satisfaction for Software Products: Implications for Design and Service Support." **Management Science** 41(9): 1456-1470.

Kubie, J. J., L. A. Melkus, et al. (1996). "User-Centered Design for Productive Systems." **Information Systems Management** (2): 38-48.

Lawrence, M. and G. Low (1993). "Exploring Individual User Satisfaction Within User-Led Development." **MIS Quarterly** 17(2): 195-208.

Lyytinen, K. (1988). "Expectation Failure Concept and Systems Analysts' View of Information System Failures: Results of an Exploratory Study." **Information and Management** 14: 45-56.

Mantel, M. and T. Teorey (1989). "Incorporating Behavioral Techniques Into the System Development Life Cycle." **MIS Quarterly** 13(3): 257-274.

McComb, D. and J. Y. Smith (1991). "System Project Failure: The Heuristics of Risk." **Journal of Information Systems Management** 8(1): 25-34.

McKeen, J., T. Guimaraes, et al. (1994). "The Relationship Between User Participation and User Satisfaction: An Investigation of Four Contingency Factors." **MIS Quarterly** 18(4): 427-451.

Misic, M. (1996). "The Skills Needed by Today's Systems Analysts." **Journal of Systems Management** 47(3): 34-37.

Montazemi, A. (1988). "Factors Affecting Information Satisfaction in the Context of the Small Business Environment." **MIS Quarterly** 12(2): 239-256.

Myers, M. D. (1993). A Disaster for Everyone to See: An Interpretive Analysis of a Failed IS Project. Proceedings 4th Australian Conference on Information Systems. **4th Australian Conference on Information Systems**, Brisbane, Department of Commerce, The University of Queensland. 595-614

Nicholas, J. M. (1990). **Managing Business & Engineering Projects**. New Jersey, Prentice-Hall Inc.

Nichols, M. C. (1994). "Using Style Guidelines to Create Consistent Online Information." **Journal of the Society for Technical Communication** 41(3): 432-438.

Norman, D. (1990 B). Why Interfaces Don't Work. **The Art of Human-Computer Interface Design**,. B. Laurel. USA, Addison Wesley Publishing Comp: 209-219.

Robert, D. (1997). "Creating an Environment for Project Success." **Information Systems Management** 14(1): 73-77.

Rouse, A., D. Watson, et al. (1995). System Development Practices: A Comparison of the Experience of Australian and United States' Organizations. **1995 Pan Pacific Conference on Information Systems**, Singapore, National University of Singapore. 46-57

Shand, R. M. (1994 B). "User Manuals as Project Management Tools: Part II - Practical Applications." **IEEE Transactions on Professional Communication** 37(3): 123-141.

Simon, H. (1994). The Executive as Decision-maker. **Readings in Decision Support Systems**. D. Arnott and P. O'Donnell. Melbourne, Department of Information Systems, Monash University: 59-64.

Szajna, B. and R. Scamell (1993). "The Effects of Information System User Expectations on Their Performance and Perceptions." **MIS Quarterly** 17(4): 493-516.

Torkzadeh, G. and W. Doll (1993). "The Place and Value of Documentation in End-user Computing." **Information and Management** 24: 147-158.

Turner, R. and G. Lowry (1999). Educating Information Systems Professionals: Towards a Rapprochment Between New Graduates and Employers. **Australasian Conference on Information Systems**, Wellington, New Zealand, School of Communications and Information Management, Victoria University of Wellington, New Zealand. 1049 - 1058

Underwood, A. and G. Maynard (1996). ACS Guidelines for the Accreditation of Courses in Universities at the Professional Level. **Australasian Conference on Information Systems** 1996, Hobart, Tasmania, University of Tasmania. 759-765

Weinberg, R. S. (1991). "Prototyping and the Systems Development Life Cycle." **Journal of Information Systems Management** 8(2): 35-43.

Williams, R. (1996). The Impact of Cognitive Style on User Satisfaction with an Information Systems: A Preliminary Study. **Australasian Conference on Information Systems** 1996, Hobart, Tasmania, University of Tasmania. 789-797

ONTOLOGICAL ANALYSIS OF INTEGRATED PROCESS MODELS: TESTING HYPOTHESES

Peter Green, Michael Rosemann
Dr Peter Green
Department of Commerce
University of Queensland
Ipswich, Australia
p.green@mailbox.uq.edu.au
+61 (0)7 3381 1121
+61 (0)7 3381 1029

ABSTRACT

Integrated process modeling is achieving prominence in helping to document and manage business administration and IT processes in organizations. The ARIS framework is a popular example for a framework of integrated process modeling not least because it underlies the 800 or more reference models embedded in the world's most popular ERP package, SAP R/3. This paper demonstrates the usefulness of the Bunge-Wand-Weber (BWW) representation model for evaluating modeling grammars such as those constituting ARIS. It reports some initial insights gained from pilot testing Green and Rosemann's (2000) evaluative propositions. Even when considering all five views of ARIS, modelers have problems representing business rules, the scope and boundary of systems, and decomposing models. However, even though it is completely ontologically redundant, users still find the function view useful in modeling.

INTRODUCTION

The aim of this paper is to report the results of some initial data gathering and testing of propositions that derived from an ontological evaluation of integrated process modeling (Green & Rosemann 2000). This evaluation used the ARIS framework popularized by Scheer (1999) for integrated process modeling. Over the last ten years, process management has received increased attention within the business administration and information systems communities. This increased attention derives from the fact that understanding and managing processes are tasks critical to achieving benefits from such new management philosophies as Total Quality Management, Activity-based Costing, and Business Process Re-engineering (Hammer 1990; Davenport 1993). Accordingly, integrated process modeling as a means of documenting, analyzing, and evaluating processes has also risen to prominence in organizations over the last decade.

Scheer's ARIS framework was selected as the example of integrated process modeling for the ontological evaluation for a number of reasons. First, it integrates five views of the process (data, function, organization, output, and process) to provide the user with a comprehensive modeling framework that can be used to generate relatively easy-to-understand process models. Second, its implementation through ARIS Toolset claims in excess of 20,000 licenses worldwide. Accordingly, there appears to be a large, mature user base against which analytical propositions can be tested. Finally, ARIS underlies the reference models provided in one of the world's most popular Enterprise Resource Planning (ERP) packages worldwide at the present time, SAP R/3.

This work is motivated by the desire to develop and begin testing the propositions generated by Green and Rosemann's (2000) ontological evaluation of Scheer's ARIS integrated process modeling framework. In doing so, we can develop and initially test a data-gathering instrument. We can use the instrument to collect some preliminary data on the propositions. From the results, we can refine the data-gathering instrument but, more importantly, we can begin providing some insights into the usefulness of the ontological evaluation for process modelers using ARIS. Accordingly, this paper reports the formulation of a testing instrument and essentially its initial pilot testing with a cohort of post-graduate information systems students studying process modeling.

The paper unfolds in the following manner. First, some further background is provided on what is ontology and how the ontological evaluation was performed. Next, Green and Rosemann's (2000) analytical results are summarized. Then, the research methodology used in this work is explained. Following that explanation, the results of this initial work are presented and discussed. Finally, we explain briefly how this work will be progressed.

BACKGROUND

As grammars for information systems analysis and design have proliferated over the years, researchers and practitioners alike have attempted to determine objective bases on which to compare, evaluate, and determine when to use these grammars (e.g., Batra, Hoffer, & Bostrom 1990; Karam & Casselman 1993). Throughout the 80's and 90's, and now into the new millennium however, it has become increasingly apparent to many researchers that without a theoretical foundation on which to base information systems analysis and design (ISAD) grammar specification, incomplete evaluative frameworks of factors, features, and facets would continue