21   UML: A Complex Technology Embedded in Complex Organizational Issues

Tor J. Larsen
Norwegian School of Management
Oslo, Norway

Fred Niederman
Saint Louis University
St. Louis, MO U.S.A.

Moez Limayem
Lausanne University
Lausanne, Switzerland

Joyce Chan
City University of Hong Kong
Hong Kong, China

Abstract   *Much computer science literature addresses the mechanics of UML and requirements modeling, but little research has addressed the role of UML in the broader organizational and project development context. This study uses a socio-technical approach to consider the interaction between UML as a technology embedded in a social environment. In this study, project developers were interviewed in detail about their use of UML along with influences on their decisions to use this tool and the results of using it. Data were analyzed using a causal mapping approach. Major findings included (1) the unanticipated observation that project success was only one of several distinct and important development outcomes; (2) a very large number of variables impacting project success were reported; (3) a number of important variables that exist in complex (nonlinear) relationships with project success; and (4) the majority of interviewees linked use of UML to project success.*

Keywords   Unified modeling language (UML), social environment, causal mapping approach

1   INTRODUCTION

In the past few decades, advancements existed in different areas of computing, including hardware, networking, and software—only software development productivity seems to be progressing slowly, if at all (Cline and Guynes 2001; Chatzoglou 1997; Sauer 1999). To increase software development productivity, many efforts have sought methods for creating better documentation and requirements structuring. For example, data flow diagrams (De Marco 1978), data modeling (Chen 1976), and object-oriented modeling (Brown 2002) have all been added to the repertoire of system designers. The underlying concept is that visual representation, accuracy, and a fairly straightforward nomenclature in modeling system characteristics can serve to help bridge understanding between system users and developers. Such understandings should allow for reducing the number of systems that are technically valid but don't resolve business problems while also providing clarity for technical designers and coders to more efficiently translate requirements into artefacts.

This study is focused on one particular approach to system representation, the unified modeling language (UML). UML is a language for the specification, visualization, and documentation of software systems (Object Management Group 2004). It also supports business modeling, including structure and design (Booch et al. 1999; Jacobson et al. 1999; Rumbaugh et al. 1999). UML consists of a set of diagrams for representation of system models in diagrammatic notation. These diagrams fall into three categories: structural diagrams (including class diagram, object diagram, component diagram, and deployment diagram), behavior diagrams (including use case diagram, sequence diagram, activity diagram, collaboration diagram, and state chart diagram), and model management diagrams (including packages, subsystems, and models) (Object Management Group 2004). Although UML is relatively new, it is the most complete approach spanning from user-information processes to implementation concerns. It is also widely held to be the future approach to modeling information systems (Edwards 2003).

Research and practice reports on UML are diverse and scattered (Cho and Kim 2002; Johnson 2002; Sim and Wright 2001). We find documentation of negative as well as positive effects of UML deployment. Typically, a small subset of UML-related issues is addressed within a very narrow area of the project effort (such as Chabrol and Sarramia 2000; Rástocný et al. 2004; Saleh and El-Morr 2004). However, reflecting upon how UML is used in the organizational context, it quickly becomes apparent that decisions and implementation interacting at multiple levels must be involved. For example, managers may or may not purchase CASE tools supporting UML modeling or make them available to workers; workers may or may not be interested in using the tools or in using them faithfully to their initial design. Because personnel are likely to vary in their attitudes toward using CASE tools and UML and because tasks will vary in their requirements for documentation, not all projects can be expected to use the UML approach, unless the technique is mandatory in the organization.

To develop an understanding of the use of UML in context, this study considers influences on the use of UML and results from the use of UML at the organizational, project, group, and individual levels. We believe this is consistent with a socio-technical perspective where the UML and CASE tools that support it present a new "technology" and the various social contexts interact with it to produce its ultimate

results (Mumford and Ward 1968). This is consistent with the discussion on principles of sociotechnical theory in viewing the domain as a set of human and technical subsystems that combine toward achieving overall goals of the whole system (Hirschheim et al. 1995; Lyytinen 1987).

In the remainder of this paper, we first review recent studies about UML and discuss where UML may fit into existing literature on MIS project success. Next, we discuss the research method based on extensive interviews and causal analysis of response. Then, we begin with a description of our observations regarding the nature of success, the dependent variables in development projects, and continue with a discussion of the results obtained through interviews and causal analysis regarding the research questions of this study. After that, we discuss implications of these findings for practice and for research. Finally, we conclude with observations about limits of the study and directions for future research.

2 LITERATURE REVIEW

A study of UML in the organizational context must take as its baseline that UML is a highly complex phenomenon (Siau and Cao 2001, 2003). This complexity shows itself in many ways. For example, content is important as well as formatting—elaborate and nice diagrams do not, in their own right, represent good design (Eichelberger 2003; Kim et al. 2000). On the other hand, without clear formatting, the content may not emerge accurately and clearly. When moving from the perspective of individuals to commonly expressed representation, such as UML, the end result may be colored by preferences among those making the representation as well as by observable aspects of business processes (Kjellman 2002). Moreover, UML having many rules governing each of several mapping techniques applied to different views of an application is generally not easy to learn (Sim and Wright 2001). In the spirit of requisite variety (Van de Ven 1986), a "technology" that has significant internal complexity, as we see from the computer science literature, should require some complex capabilities for assimilation in its organizational environment.

Because the use (or nonuse) of UML exists within a context of organizational transformation, change is looked upon as endemic to the practice of organizing (Iivari and Huisman 2001; Orlikowski 1996). UML in its totality might be looked upon as an information systems development (ISD) method and each of its maps as an example of an ISD technique (Iivari et al. 2004). Yet, relationships between organizational matters and ISD realities may coincide with or expand beyond well established views on ISD paradigms (Iivari et al. 1998, 2000-2001) or systematization of ISD issues into groupings such as organization, project, or supplier. The use of UML most likely involves a process of organizational change over time (Fichman and Kemerer 1997; Orlikowski 1993).

One purpose of this study, therefore, is to extend from the intricacies of UML rules and its role in technical development to additionally consider how its use or nonuse is manifested within the organizational environment. We further take the view that, from the perspective of the practitioner, such outcomes generally are not neutral in value. As a result, organizational research typically involves a search for factors that influence

outcomes toward those which are preferred. Because the specific preferred outcomes will differ in light of differing goals, the umbrella term for preferred outcomes without regard to the particular outcome is labeled *success*. So, this research aims to understand how UML is manifested in organizations but more particularly how it may influence organizational success. It is also clear that the success of UML cannot be tested if UML is not used or available for use in a development environment. Therefore, in this research we also consider what organizational factors may affect the decision to make UML available for use and that may affect decisions regarding whether, when, and how to use it if it is available. To this end the following general research questions are posed:

> **RQ1**: Within the organizational context, does the use of UML lead to project success, and if so, to which particular aspects of project success?

> **RQ2**: What organizational factors influence the use of UML in system development?

3   METHOD

In order to address our research questions, a grounded theory approach was chosen. First proposed by Glaser and Strauss (1967), this paradigm line has been expanded and enriched by Information Systems researchers (such as Bryant 2002; Galal 2001; Hughes and Jones 2002; O'Connor et al. 2003). Although grounded theory and case study overlap in terms of data collection and analysis, they are reciprocal in the sequence of theory generation. In case study, data are collected for hypotheses testing, while in grounded theory, data serve as the foundation of theory building. In other words, grounded theory examines and describes empirical observations in the language of abstracted theoretical concepts. Moreover, it emphasizes the practical contribution of the theories generated by providing practitioners with a thorough understanding of the phenomena they encounter everyday (Glaser and Strauss 1967) and thus fits the inter-pretative orientation of this study. The intent is to get close to UML-related phenomena as the stakeholders themselves would perceive them but simultaneously hoping to avoid drowning in data (Nandhakumar and Jones 1997).

We conducted in-depth interviews with individuals involved with IT development who logically would be involved in the selection of development approaches and tools or users of these approaches and tools (see Appendix A for respondent demographics). This approach allowed targeting of the research to issues of interest as suggested by the literature without predetermining the content as viewed by the interviewees. The interview questions were generally semi-structured in that they provided a general topic of interest, but allowed interviewees to interpret the question in terms of their own experience and present responses that frequently varied from the specific topic area. An example of such a question is: Thinking back over the last 5 years or so, what are the most drastic changes with regard to how you work on projects? Research based on semi-structured interviews represents an intermediate level of engagement between

researcher and research "subject" and is reportedly used in 8 percent of "positivist" research and 32 percent of "interpretive" research (Nandhakumar and Jones 1997).

Causal mapping serves as an analysis technique for observing relationships and providing insights into UML's role and its use in the organizational context (Larsen and Niederman 2005). Causal mapping provides a method for organization of qualitative data emphasizing the range of observations and the relationships among them. Our use of causal mapping can be understood in the context of the basic research method challenge of the personal pronouns of we, I, and you (Bohman 2000). "We" represents the objective approach to research—as in a deductive approach. "I" represents the individualized approach to research—as in an inductive approach. "You" represents the real world subjects and their views, attitudes or behaviors. Obviously, researchers using the voice of "we" or "I" attempt to express salient aspects of "you." Additionally, researchers employ a method within which aspects of "you" are studied and analyzed.

The particular concern in this research was that causal mapping strongly emphasizes a clear definition of the two elements of construct/variable and causal relationship between constructs (see Appendix B for this kind of statements extracted from interview transcriptions). Used in a straightforward and narrow manner, causal mapping represents the voice of "we"—a deductive and objective approach to research, as in Nelson et al. (2000). The present researchers argue that subjects—the "you" in research—may possess clear views of constructs/variables and causal relationships. However, the presumption, in Nelson et al.'s research is that these will be parsimonious on the individual level and across individual roles. The authors believe, however, that our domain of interest, UML within the organizational context, does not have the preexisting background of prior investigation to allow for identification of those constructs/variables and causal relationships before examining data; but rather the study accumulates data as the source for discovery of these.

Therefore, this data collection method was not intended to define a set of constructs/variables and causal relationships, as in Fahey and Narayanan (1989). Doing so would have the danger of leading respondents to fill in the suggested topics without considering whether the set of topics is complete. That is, since very many constructs/variables and relationships exist in the real world, directing subjects to talk about a preselected subset of these phenomena may lead to definitions that reflect the preselection rather than represent the real world. In this scenario, the outcome might appear clean, but actually hide the fact of missing elements (Kjellman 2002). Therefore, given the lack of prior research guidance, in this research subjects were encouraged to talk freely while guided toward the present research focus. Our approach should and must be closer to the spirit of having "you" talk as the "you" finds appropriate. Data collected in this manner would, hopefully, be a good source for defining constructs/variables and causal relationships with a high degree of validity.

Given the relatively larger numbers of identified variables affecting project success both directly and through linked chains of influence (e.g., OO design influencing scalability and scalability influencing project success), it became apparent that lessons could be learned about the organizational context of project success in addition to those regarding the role of UML per se. As an example, the final consolidated causal map for interview 3 is shown in Figure 1.
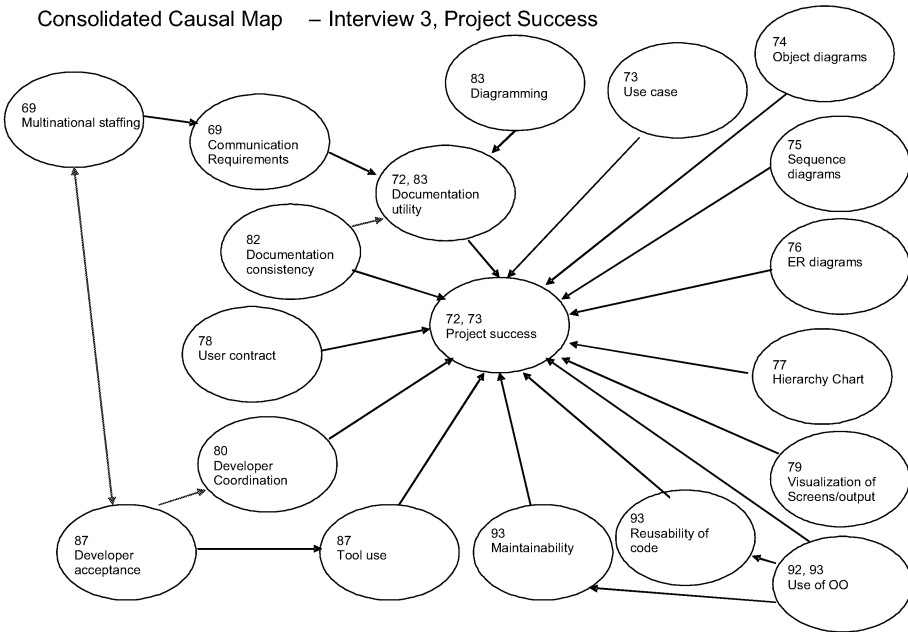
Consolidated Causal Map   – Interview 3, Project Success

Figure 1. Consolidated Causal Map for Interview Number 3

We see from this sample consolidated causal map that even a single interview can yield a large number of variables that impact project success. Indirect relationships are also observed. For example, "Use of OO" to the right and bottom of Figure 1 impacts "Maintainability" and "Reusability of code," both of which have an impact on "Project success." The set of variables impacting project success and indirect relationships varied to a large degree among our 11 interviews. This is illustrated by contrasting the sample shown in Figure 1 with the consolidated causal map for interview number 8 in Figure 2.

Although some degree of similarity exists between these two sample consolidated causal maps, to be discussed in the results section, we conclude that our raw data clearly documents that the use and value of UML is a highly complex phenomenon. It can be observed that the range of variables is quite broad, that the nature of the relationships is not at all simple and straightforward, and that observation of particular variables seems to shift significantly with varied circumstances.

By contrasting the consolidated causal map with the interview transcripts and observation memos, checking and revision can be made. We consider the extent to which the wide range of variables may affect project success by first grouping them under a more abstract heading called *category* (Corbin and Strauss 1990). This grouping is done by developing a categorization of these variables (see Appendix C). As discussed in the next section, these variables are of different types and occur at different levels of analysis in the overall picture of project success.
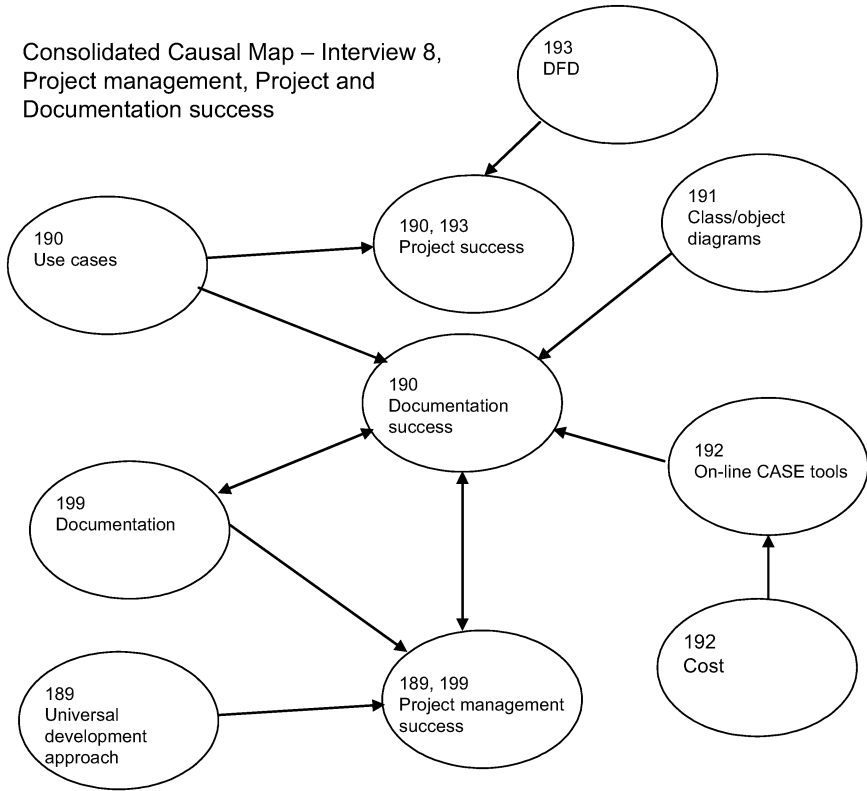
Figure 2. Consolidated Causal Map for Interview Number 8

4 RESULTS

RQ1: Within the organizational context, does the use of UML lead to project success, and if so, to which particular aspects of project success?

Most interviewees in the study clearly linked the use of UML with overall project success. Of the eleven interviewees, eight indicated a direct link of one sort or another. Commonly the source of that benefit was from providing a common understanding from which multiple staff members could work independently yet remain coordinated.

However, the interviewees also mentioned several variables linking UML with its influences and effects. These variables are presented by respondents in Appendix D. More than one respondent mentioned that UML could lead to documentation success independent of application or project success. They also mentioned UML having an impact on successful testing, and also on creation of more reusability. Each of these comments indicates more specifically how UML can be expected to influence overall project success. In some cases, however, the relationship between UML and applica-

tion/project success was stated in negative terms—that the lack of use of UML resulted in poorer application and project results.

Emergent from the data were several potential outcomes of interest to participants that one might want to optimize or improve. Four distinct outcomes could be identified from the data. These can be categorized as project management success, project success, application success, and documentation success. Project management success refers to the approaches and the activities that an organization implements to improve the effectiveness of new development projects and, ultimately, of its IT portfolio. Project success refers to the outcomes generated for each project. Application success refers to the technical outcomes for a particular development project. Finally, documentation success refers to the outcomes for creating, storing, retrieving, and using the meta-knowledge pertaining to the requirements, design, and implementation of new systems. The views held on these issues are not singular or one-dimensional. Project stakeholders' evaluation of each success category (and how important each category is held to be) may vary considerably (Glass 2003; Klein and Jiang 2001).

It is worth focusing some additional attention on documentation success because it was discussed in various ways by the study respondents and it relates closely to the use of UML. The documentation success variables include documentation quality, consistency, utility, standardization, thoroughness, requirements/technical models linkage, and use of a document management tool. These variables describe aspects of successful project documentation in the sense that they are goals and measures for the documentation itself. Respondents mentioned quality of documentation as a cause of more successful projects, as an end in itself, and also observed that the lack of quality documentation can cause project failure or increase project cost or time to completion. Two comments illustrate the complexity of this variable.

> You know, you can document everything in the world and if you've got people that either (a) don't read it or (b) read it and don't care about it—it doesn't matter. You've just wasted a lot of time. So, you know, I've been on teams where they've spent a great deal of time doing everything by the book and being very formal about it. And doing a great job at it. It's not like they didn't do a good job at it but the problem was that the team didn't just follow it.

> Uh…yes, we had both of them [object and class diagrams]….those were very advantageous because you could clearly see, you know, at a glance, what all was involved with an object and what the procedures and methods were.

Documentation success, in its various aspects appears to be influential relative to project success, but probably not in a linear relationship at all levels for all projects. Granted that different projects will suggest different amounts of documentation, for a given project, too little documentation holds the potential to raise costs if there is need for rework or reorganization. However, as the creation of documentation has inherent costs, producing more than is needed or used will decrease project success by increasing costs that are not recoverable, particularly if those who could potentially profit from their use choose not to do so.

4.1 What Are the Relationships among the Four Different Types of Outcome Success?

As stated above, documentation success would generally influence application success. Application success should influence, though not determine, project success. Clearly an application that won't compile isn't going to be of much practical value; however, one that performs a task flawlessly that no one needs is not going to be viewed as highly valued either. The relationship between documentation, application, and project success is complicated by the potential for documentation success to influence not only application success but also to directly influence project success. Particularly as quality of information requirements documentation rises, effects should be felt on user acceptance as well as technical quality of the applications produced. All of these, however, focus on measures by application or project. Project management success, in contrast, considers portfolios of applications and approaches to development that can influence documentation, applications, and project success project by project. This presents a potential feedback loop in that project management success is rather hollow if individual projects are not successful. It is possible that an organization will have strong normative commitment to generally successful project management approaches, yet fail to implement or execute these methods in some or all particular projects. It is also possible that an organization applies appropriate methods skillfully on a project by project basis with little or no self-conscious project management organization. It would be expected that in most cases project success and project management success should move together, yet these scenarios show circumstances where they might move in opposing directions, therefore they require separate conceptualization and measurement.

> **RQ2**: What organizational factors influence the use of UML in system development?

Although most of the respondents noted the potential of UML to influence project success, a very large number of additional variables were also observed influencing project success and the other dependent variables. It would be simplistic to expect that UML alone without an alignment of positive values on these other measures would lead to positive outcomes. Thus, we cannot say that a specific culture carries influence on the use of UML or on project success (Iivari and Huisman 2001). Our data suggest that UML is partially used, at best. In our sample, consistency in managerial focus on development method issues is missing.

The wide range of variables and how they may affect success outcomes are addressed first by developing a categorization of these variables (see Appendix C). It will be noted in the discussion below that these variables are of different types and occur at different levels of analysis in the overall picture of antecedents to dependent variable values. The eight categories observed are environmental factors, organizational factors, staffing issues, coordination methods/process, OO and CASE tool use, specific modeling tools, and mixed direction factors (see Figure 3). After considering the range and nature of these variables and categories, we discuss the relationships among these factors influencing project success.

The **environment** category of variables includes package use/customization, user contract, physical arrangement, task complexity/project difficulty, and user characteristics. These variables share two distinctions from the perspective of the development project. First, these variables are relatively fixed from the perspective of the project. Ideally decisions such as whether or not to go with installation of a package or self-built system as well as negotiation regarding the user contract are initial elements of the project itself. Frequently, though, these elements are settled upon prior to the beginning of the development of the application. The task complexity or project difficulty may be extremely fixed given the nature of the desired functionality, or may be amenable to phasing, scope management, decomposition, or other tactics for managing complexity. Physical arrangement pertained to the actual working space of the developers which, in the example described, led to communication and coordination difficulties.

The **organizational factors** category of variables includes organizational culture, change management, leadership, and matrix organization. This category has fewer variables than we expected, but three of them would seem quite important in many circumstances and probably to be broad enough to include a variety of more detailed elements. Organizational culture and leadership (as suggested by Westerveld 2003) would seem both important and closely related, as expressed by one respondent,

> because the real issue is not technical, it's cultural. And what I keep…at least in our environment, I'm finding, it is a bottom-up trying to get this stuff in. And I'm becoming more and more convinced that bottom-up culture changes don't work. Unless you have a top-down commitment to doing it, the bottom-up…there [are] just too many people. People trying to advocate one way are actually adding to the chaos rather than actually producing a solution.

Change management was mentioned by four respondents, three times in the sense of managing behavioral change of developers and system users to take advantage of new capabilities, but it was also mentioned in terms of managing changes or revisions in the project requirements. The idea of a matrix organization was mentioned once by an individual in a larger company in the context of suggesting that IT managers need clout in dealing with business managers from some IT organization that cuts across various business departments and can, therefore, influence adherence to standards, even when they do not optimize development of individual projects.

The **staffing** variables include multinational teams, skills and knowledge, developer skills and staffing, staff learning curve, project staffing, staffing patterns, and internal/external staffing. Project success is viewed as negatively related to turnover and vendor turnover. The conjecture that staff issues should be related to project success is quite logical. Developing new systems must be the responsibility of employees (or consultants) and the quality of their work will clearly be reflected in the outcomes of the project. Of more interest are the variations on this theme expressed by the study respondents. Overall skills and knowledge was observed in the context of the degree to which UML methods and CASE tools could be implemented and in terms of the cost of training or learning to use these methods and tools for a particular project. Emphasis is placed on the extra communication requirements when project teams are blended from personnel of multiple countries and where there is a mixture of consultants, contractors,

and full-time employees. Staffing patterns were viewed as important. Not only are the skills of the staff over the lifetime of the project important, but also the way that staff enter and exit from their portion of work on a project. Where turnover was mentioned, it was viewed as having a significant negative affect on the project in terms of requiring rework, venturing into new directions, and changing the culture of the group. Variables related to developers and developer teams were clustered together in this category. Skills and knowledge of the user was viewed as an environmental issue. Another respondent focused on the appropriate levels of technical and business-related skills of the project manager, which were viewed as critical for development team leadership and, ultimately, for project success.

The **coordination** variables are communication coordination and coordination between phases. Six of the respondents mentioned coordination and communication. Communication among developers and communication between developers and users were both mentioned. This particular element is similar to but slightly different from analysis-requirement specification-requirements gathering in that communication is a base for conducting those activities; communication also involves more open-ended exchange of ideas, which is a key to project success (Belout and Gauvreau 2004). In examining the flow of multiple causal statements, coordination seemed to moderate some of the environment, project management, and organizational factors as they in turn influenced UML and tool use as well as ultimate project success.

The **methods/process** category of variables includes OO project management, early testing, phased development, analysis requirements/specification requirements gathering, scope creep, task accounting, metrics, master scheduling, and quality assurance. These variables share the commonality of pertaining to the approach and methods used in project management that shaped the activities of the particular project. This category includes some specific activities, the inclusion of which contributed to project success. These are early testing, task accounting, metrics (as suggested by Avritzer and Weyuker 1999), master scheduling, and quality assurance. OO project management refers to the decision to move to UML for the documentation of the project and is closely related to the use of OO per se. Analysis, requirements specification, and requirements gathering are grouped into one factor as they all represent aspects of taking the time to document the desired functionality of the new application. These variables are derived from separate interviews, but each seems to be focused on taking the time to gather knowledge from eventual users/sponsors and to take the time to record this knowledge in a set of specifications. These variables would seem to relate in their instantiations to project success, but in general form to project management success. This topic area is a primary message of systems analysis courses since they have been offered, but, at the same time, remained a source of frustration as its absence or insufficient use of it is a cause of lack of project success according to our respondents.

The **OO and CASE tool use** variables include quality of use and standardization of use. The amount of use of either the OO approach, UML specifically, or CASE tools was not raised as an issue by the respondents. Rather how well the approach and tools were used was viewed as impacting ultimate project outcomes. The standardization of use might be viewed as a subset of quality of use, but we saw this as slightly different. Quality of use seemed to refer to how well the tools provided a mapping between user requirements and technical specification; standardization seemed to refer to how

consistently the various development team members applied the tools. Logically, these are likely related; consistency might be a characteristic, among others, of quality across a group of developers, and higher quality users, if measured by other means, are likely to be more consistent. Several respondents mentioning the use of the OO approach noted that such an approach was only feasible with the appropriate use of CASE tools. Other respondents did not mention CASE tools and in some cases performed UML modeling with standard office tools such as MS-Word for describing use case scenarios and PowerPoint or Visio for graphically showing relationships. A number of respondents indicated that high market share CASE tools, particularly Rational Rose, can be expensive, particularly for distribution across the set of developers, and were not extensively used. Even when used, a number of issues with CASE tools were raised including difficulty with the size of documents relative to viewing screen size. On the other hand, multiple respondents indicated that significant benefits from the use of OO approach with CASE tools was largely due to improved communication allowing for smoother transition from analysis/design to coding and implementation. In the words of one respondent,

> It was beneficial because there were programmers, contract programmers, who had certain expectations. They expected to be given a specification from which to work, not, you know, go on figure out how to do this cart blanche. So there was "here's a use case," "here's a sequence diagram," "these are the classes/methods you need to implement." So creating a specification that can be given to someone is the benefit…in a format that they're going to familiar with.

Specific modeling tools included reference to tools in the OO/UML category, sequence diagrams, activity diagrams, use cases, and class/object diagrams as well as some more traditional tools including prototyping, entity-relationship diagraming, hierarchy charts, and data flow diagrams. None of the respondents cited all of these tools as helpful relative to their particular projects and a number of them pointed out that in some cases the use of a particular tool was not helpful for a particular project. However, each of these techniques in the right circumstances is held to be of value. The most frequently used technique recommended by the respondents is use cases. The two interviewees who referenced prototyping individual projects referred to this technique as highly influential in creating successful projects. In both cases, the prototyping was used to document specification and particularly to demonstrate both user interface appearance and characteristics.

The **mixed direction factors** represent something of a miscellaneous group of variables that were frequently mentioned, but where each contains some complexity. These variables include reusability/reuse, maintainability, simplicity, user satisfaction, production speed, scalability, and cost. This complexity derives from the fact that they can be measures of project success as well as inputs to it. For example, a successful project may be one where the code is more maintainable, simpler, produced more quickly and at lower cost. On the other hand, documentation techniques, project management approaches, and staffing activities that produce documentation that is more reusable, with tools that are cost effective, and which generate user satisfaction will lead

to better project outcomes. Each of these variables appears in the text in multiple locations and represents a complex relationship with project success.

Upon examination of these findings, we believe in the range and potential power of the full list of variables; however, we hold out this categorization more tentatively. Our grouping of these variables tends to be influenced by consideration of variables in other MIS domains and traditional research that has proposed constructs influencing project success. It is quite conceivable that practitioners would have other bases for categorization, perhaps by cost, perhaps by clusters of variables that are found together in practice, or perhaps by the degree to which they have had control over the variables within their particular organizational history. Some of the variables clearly reside at the periphery of the categories to which they have been assigned. Physical arrangement, for example, could be housed in "environment" or "coordination." Whichever category one would place it, however, would not detract from the contention that under some circumstances each can influence project outcomes. The purpose of categorization, therefore, is to extract themes rather than to create solid constructs with categorical existence of which the variables serve as components.

4.2 Relationships among Variables and Categories

Among the elemental categories that we have provisionally populated with specific variables, we can identify some beginnings of logical relationships (see Figure 3). From the perspective of the project, the categories of environmental factors, organizational factors, staffing issues, and coordination were viewed as forming an outer ring. In many cases, the variables in these categories will be outside the direct and immediate influence of the project team participants. Much of the environment and organization may be relatively fixed for the duration of the particular project, yet the specific values of these variables may have significant effect on the ultimate project outcomes. These categories are likely to have direct effects on project success as well as effects moderated by the additional variables that are more likely to be under direct project control.

Methods, processes, and staffing variables are likely to share this characteristic of being outside the immediate and direct control of the project team, but perhaps to a lesser extent than the environmental and organizational variables. For example, the observed use of methods and processes by development team members in the actual circumstances of project work may have significant levels of choice by individual members—even though parameters of which, if any, tools are made available, and the degree to which such use is mandated may well relate to overall company policies and practices. Similarly, staffing decisions may be made solely outside of the project team in terms of organizational hiring and training and assignment to projects and teams. Decisions regarding the use of contract workers, consultants, and offshore outsourcing will rarely be part of the project team mission, although in some circumstances, teams or team members may influence these decisions. However, some staffing decisions, such as turnover, may be influenced by outside forces but also result from individual decisions and opportunities such as unexpected job offers, family crises, etc. In spite of these differences, for the most part these factors will be largely outside the control of the project team. Finally, coordination appeared to refer to a subset of the interaction
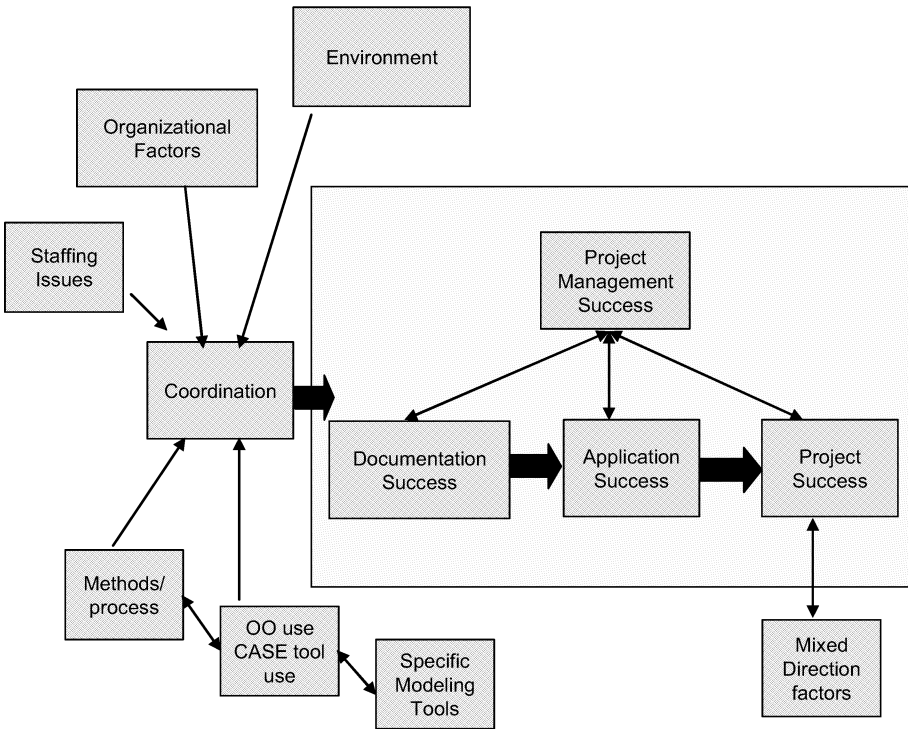
Figure 3. Combined Model of Project Success Precursors

of these other four elements—how well the organizational, environmental, process/
method approaches, and staffing blended together. It would appear that coordination
was influenced by these other "outer ring" variables. We would suggest that these
categories and the variables of which they are comprised are likely to have direct
influences on dependent variables, but that in many cases, differing values of these
variables will have counterbalanced effects given different actions and reactions by the
project team members. Thus, we place these variables in something of an outer ring
relative to the remaining categories.

     The remaining categories, OO and CASE tool use and specific modeling tools,
would seem generally in the hands of the project developers themselves based on our
respondents' comments. At least we would see it this way among the array of CASE
tools that are available at the time of the project. Truly it isn't likely that individual
teams or team members are going to buy new CASE tools beyond those that are
relatively inexpensive. But the quality, amount, and consistency of use of these tools
would seem something that the team should be able to determine. We would see that
the overall approach to OO/CASE tools, the selection of specific modeling approaches,
and the approach to documentation are highly interrelated. We would expect the sum
of these variables to be influenced by the "outer ring" variables; we would expect that

for different teams working on different projects, the detailed quantity of impact on the variables comprising these categories would vary. By the same token, we would expect that these variables would act together in influencing project outcomes and that individual values of these variables may have additional influence on project success.

The final category, mixed direction factors, is included in the overall model for influences of dependent variables primarily as a reminder that these variables exist to the respondents and may be causes, results, or measures related to development activities. Our emergent model is shown in Figure 3.

5 CONCLUSION

This study has documented a number of observations about the role of UML, the object-oriented approach to analysis and design, and CASE tool availability and use as part of a more comprehensive look at factors influencing development project outcomes.

5.1 Implications for Research

Observations in this study suggest that project success is only one of four discernable outcomes of interest in the evaluation of development activities; application success, documentation success, and project management success are also key out-comes. The correlation among these should be strong, but at times there may be tradeoffs between them as well. This study provides a basis for differentiating different types of success. Future research would do well to quantify the strength of relationships among these different dependent variables as well as catalogue the contingencies in which each merits predominance in the business environment. Additionally, this study identified variables that are themselves complex. Most notably the variables clustered in the "mixed direction factors" category may be viewed in some ways as influencing project success while also being measures of success.

5.2 Implications for Practice

In this study, a multitude of variables were identified as relevant to development outcomes. Although, in general, the use of UML and CASE tools was viewed as influencing project success and other outcomes, practitioners will do well to consider the array of potential influences on project success before committing to using these tools. The authors have extrapolated from the overall interviews and specific observed relationships to suggest a clustering of reasonably similar variables, although it is not clear that these would move together from one project to another. The authors have suggested a model that would encompass meta-relationships among categories of variables at an overview level. In particular, our data suggests strongly that project size should be accounted for in decisions regarding use of UML simply because its employment in small projects is likely to result in overkill. We would also point out that in project settings where the use of UML appears relevant, its implementation and use requires the integration of policies as well as practices across organizational levels.

Finally, we suggest that practitioners keep firmly in mind that policies designed to optimize success at one level may have unintended implications on other important success indicators. For example, investment in tools to optimize application success will only contribute to greater project success if their costs are outweighed by resulting benefits. However, if these costs are distributed across projects, they may result in both project and project management success over time.

5.3 Future Research

These observations and extrapolations suggest an array of detailed research questions. For example, are there key combinations of variables that influence project success either constantly or across a usefully high probability of occasions? Are there environmental and organizational contingencies that define sets of relatively constant variable relationships? Are there rules of thumb leading to more precise measurement of the tradeoffs among dependent variables deriving from decisions on particular independent variables? For example, would some amount of use of UML signal a transition from more benefits (resulting from improved documentation quality) to less net benefits (resulting from higher levels of overall project cost)?

On the other hand, as this study was carried out in a Western country, it is not clear whether the result will be different in other cultural settings. The factors contributing to project success identified in this study include staffing factors such as multinational team and staffing patterns, as well as organizational factors such as organizational culture, leadership, and matrix organization. The contribution of a multinational team to project success comes from a combination of the merits of different cultures. Besides, people with different cultural background may have different leadership styles, which ultimately affect the level of system engineering project success. Therefore, a cross-cultural study on the use of UML could enrich these findings.

5.4 Study Limitations

This study aimed at considering the interaction of what turned out to be many variables. Such a study tends to be ambitious, and as such, cannot help having some limitations. This study is based on 11 interviews, which yielded a large volume of transcript and many elemental relationships. Nevertheless, it is possible that conducting additional interviews would have yielded even more variables and relationships and that some of those might have changed some of the subsequent categorization and projection of meta-relationships. Labeling and categorizing variables requires extrapolation from the words themselves, the context, and also the tone of the interviewee. It is possible that some measure of inferred meaning (or failure to recognize subtle nonverbal signals) from participation in the interview events influenced the ultimate labeling and categorization of underlying comments. To partly address this issue, all interviews were conducted by two individuals from two cultural contexts (one European, the other North American). Interviewers sought to clarify terms used within the interview setting; for example, interviewees discussing UML were often asked more specifically to which models they referred, or in discussing success, were often asked to clarify in what sense

they were using the term. Nevertheless, some alternative meaning for common terms may have escaped the observation of interviewers. Along these lines, Nandhakumar and Jones (1997) point to the additional dangers in engaged research stemming from distorted accounts of behavior and deliberate misrepresentation. Although the researchers can never be certain that these behaviors did not occur, it was clear that interviews included a wide range of both critical and complementary commentary regarding organizational development policies and practices. However, the authors were careful not to predesignate labels or categories that could have led to preestablished conclusions.

5.5 Contribution

In the end, the observation of such a large array of variables, the interactions among multiple levels of organization, project, and individuals, and the multifaceted aspects of varied types of success indicators leads to the conclusion that UML is indeed a complex innovation that becomes embedded in complex organizational issues. It is our hope that presenting the structured observations of our interviewees helps initiate a process of ordering the complexity and diversity of influences in the relationship between social and technical influences in the development process.

References

Avritzer, A., and Weyuker, E. J. "Metrics to Assess the Likelihood of Project Success Based on Architecture Reviews," *Empirical Software Engineering* (4:3), September 1999, pp. 199-215.

Belout, A., and Gauvreau, C. "Factors Influencing Project Success: The Impact of Human Resource Management," *International Journal of Project Management* (22:1), January 2004, pp. 1-11.

Bohman, J. "The Importance of the Second Person: Interpretation, Practical Knowledge, and Normative Attitudes," in H. H. Kögler and K. R. Stueber (eds.), *Empathy and Agency: The Problem of Understanding in the Human Sciences*, Boulder, CO: Westview Press, 2000, pp. 222-242.

Booch, G., Rumbaugh, J., and Jacobson, I. *The Unified Modeling Language User Guide*, Boston: Addison-Wesley, 1999.

Brown, D. W. *An Introduction to Object-Oriented Analysis: Objects and UML in Plain English*, New York: John Wiley & Sons, 2002.

Bryant, A. "Re-grounding Grounded Theory," *Journal of Information Technology Theory and Application* (4:1), 2002, pp. 25-42.

Chabrol, M., and Sarramia, D. "Object Oriented Methodology Based on UML for Urban Traffic System Modeling," in *Proceedings of the Third International Conference on the Unified Modeling Language (UML 2000)*, York, UK, October 2-6, 2000, pp. 425-439.

Chatzoglou, P. D. "Factors Affecting Completion of the Requirements Capture Stage of Projects with Different Characteristics," *Information and Software Technology* (39:9), 1997, pp. 627-640.

Chen, P. P. S. "The Entity-Relationship Model—Toward a Unified View of Data," *ACM Transactions on Database Systems* (1:1), May 1976, pp. 9-36.

Cho, I., and Kim, Y.-G. "Critical Factors for Assimilation of Object-oriented Programming Languages," *Journal of Management Information Systems* (18:3), Winter 2002, pp. 125-156.

Cline, M. K., and Guynes, S. "The Impact of Information Technology Investment on Enterprise Performance: A Case Study," *Information Systems Management* (18:4), Fall 2001, pp. 70-76.

Corbin, J., and Strauss, A. "Grounded Theory Research: Procedures, Canons, and Evaluative Criteria," *Qualitative Sociology* (13:1), 1990, pp. 3-21.

De Marco, T. *Structured Analysis and System Specification*, New York: Yourdon Inc., 1978.

Edwards, C. "Modeling Standard Gets Ready for Second Round," *Electronic Systems and Software* (1:5), October-November 2003, , pp. 36-39.

Eichelberger, H. "Nice Class Diagrams Admit Good Design?," in *Proceedings of the 2003 ACM Symposium on Software Visualization*, San Diego, CA, June 11-13, 2003, pp. 159-216.

Fahey, L., and Narayanan, V. K. "Linking Changes in Revealed Causal Maps and Environment: An Empirical Study," *Journal of Management Studies* (16:4), 1989, pp. 361-378.

Fichman, R. G., and Kemerer, C. F. "The Assimilation of Software Process Innovations: An Organizational Learning Perspective," *Management Science* (43:10), October 1997, , pp. 1345-1363.

Galal, G. H. "From Contexts to Constructs: The Use of Grounded Theory in Operationalising Contingent Process Models," *European Journal of Information Systems* (10:1), March 2001, pp. 2-14.

Glaser, B. G., and Strauss, A. L. *The Discovery of Grounded Theory: Strategies for Qualitative Research*, New York: Aldine Publishing Company, 1967.

Glass, R. L. *Facts and Fallacies of Software Engineering*, Boston: Addison-Wesley, 2003.

Hirschheim, R., Klein, H. K, and Lyytinen, K. *Information Systems Development and Data Modeling: Conceptual and Philosophical Foundations*, Cambridge, England: Cambridge University Press, 1995.

Hughes, J., and Jones, S. "Reflections on the Use of Grounded Theory in Interpretive Information Systems Research," *Electronic Journal of Information Systems Evaluation* (6:1), 2002 (available online at http://www.ejise.com/volume-6/issue1-art2.htm).

Iivari. J., Hirschheim, R., and Klein, H. K. "A Dynamic Framework for Classifying Information Systems Development Methodologies and Approaches," *Journal of Management Information Systems* (17:3), Winter 2001-2002, Winter, pp. 179-218.

Iivari, J., Hirschheim, R., and Klein, H. K. "A Paradigmatic Analysis Contrasting Information Systems Development Approaches and Methodologies," *Information Systems Research* (9:2), June 1998, pp. 164-193.

Iivari, J., Hirschheim, R., and Klein, H. K. "Towards a Distinctive Body of Knowledge for Information Systems Experts: Coding ISD Process Knowledge in Two IS Journals," *Information Systems Journal* (14:4), October 2004, pp. 313-342.

Iivari, J, and Huisman, M. "The Relationship between Organizational Culture and the Deployment of Systems Development Methodologies," in K. R. Dittrich, A., Geppert, and M. C. Norrie (eds.), *Proceedings of the 13th International Conference of CAiSE 2001,* Berlin: Springer-Verlag, 2001, pp. 234-250.

Jacobson, I., Booch, G., and Rumbaugh, J. *The Unified Software Development Process*, Reading, MA: Addison Wesley, 1999.

Johnson, R. A. "Object-Oriented Analysis and Design: What Does the Research Say?" *Journal of Computer Information Systems* (42:3), Spring 2002, pp. 11-15.

Kim, J., Hahn, J., and Hahn, H. "How Do We Understand a System with (So) Many Diagrams? Cognitive Integration Processes in Diagrammatic Reasoning," *Information Systems Research* (11:3), September 2000, pp. 284-303.

Kjellman, A. "The Subject-Oriented Approach to Knowledge and the Role of Human Consciousness," *International Review of Sociology* (12:2), July 2002, pp. 223-247.

Klein, G., and Jiang, J. J. "Seeking Consonance in Information Systems," *The Journal of Systems and Software* (56:2), March 2001, pp. 195-202.

Larsen, T. J., and Niederman, F. "Causal Mapping for the Investigation of the Adoption of UML in Information Technology Project Development," in V. K. Narayanan and D. J. Armstrong (eds.), *Causal Mapping for Research in Information Technology*, Harrisburg, PA: Idea Group Publishing, 2005, pp. 233-262.

Lyytinen, K. "Different Perspectives on Information systems: Problems and Solutions," *ACM Computing Surveys* (19:1), March 1987, pp. 5-46.

Mumford, E., and Ward, T. B. *Computers: Planning for People*, London: B. T. Batsford Ltd., 1968.

Nandhakumar, J., and Jones, M. "Too Close For Comfort? Distance and Engagement in Interpretive Information Systems Research," *Information Systems Journal* (7:2), April 1997, pp. 109-131.

Nelson, K. M., Nadkarni, S., Narayanan, V. K., and Ghods, M. "Understanding Software Operations Support Expertise: A Revealed Causal Mapping Approach," *MIS Quarterly* (24:3), September 2000, pp. 192-222.

Object Management Group. "Introduction to OMG's Unified Modeling Language™ (UML®)," 2004 (available online at http://www.omg.org/gettingstarted/what_is_uml.htm).

O'Connor, G. C., Rice, M. P., Peters, L., and Veryzer, R. W. "Managing Interdisciplinary, Longitudinal Research Teams: Extending Grounded Theory-Building Methodologies," *Organization Science* (14:4), July-August 2003, pp. 353-373.

Orlikowski, W. J. "CASE Tools as Organizational Change: Investigating Incremental and Radical Changes in Systems Development," *MIS Quarterly* (17:3), September 1993, pp. 309-340.

Orlikowski, W. J. "Improvising Organizational Transformation Over Time: A Situated Change Perspective," *Information Systems Research* (7:1), March 1996, pp. 63-92.

Rástocný, K., Janota, A., and Zahradník, J. "The Use of UML for Development of a Railway Interlocking System," in H. Ehrig, W. Damm, J. Desel, M. Große-Rhode, W. Reif, E. Schnieder, and E. Westkämper (eds.), *Integration of Software Specification Techniques for Applications in Engineering, Priority Program SoftSpez of the German Research Foundation (DFG),* Final Report, Volume 3147 of Lecture Notes in Computer Science, Heidelberg, Germany: Springer, 2004, pp.174-198.

Rumbaugh, J., Jacobson, I., and Booch, G. *The Unified Modeling Language Reference Manual*, Reading, MA: Addison Wesley, 1999.

Saleh, K., and El-Morr, C. "M-UML: An Extension to UML for the Modeling of Mobile Agent-Based Software Systems," *Information and Software Technology* (45:4), March 2004, pp. 219-227.

Sauer, C. "Deciding the Future for IS Failures: Not the Choice Your Might Think," in W. Currie and B. Galliers (eds.), *Rethinking Management Information Systems: An Interdisciplinary Perspective*, Oxford, England: Oxford University Press, 1999, pp. 279-309.

Siau, K., and Cao, Q. "How Complex is the Unified Modeling Language?," in K. Siau (ed.), *Advanced Topics in Database Research*, Hershey, PA: Idea Group Publishing, 2003, pp. 294-306.

Siau, K., and Cao, Q. "Unified Modeling Language (UML)—A Complexity Analysis," *Journal of Database Management* (12:1), January-March 2001, pp. 26-34.

Sim, E. R., and Wright, G. "The Difficulties of Learning Object-Oriented Analysis and Design: An Exploratory Study," *Journal of Computer Information Systems* (42:4), Winter 2001, pp. 95-100.

Van de Ven, A. "Central Problems in the Management of Innovation," *Management Science* (32:5), May 1986, pp. 590-607.

Westerveld, E. "The Project Excellence Model®: Linking Success Criteria and Critical Success Factors," *International Journal of Project Management* (21:6), August 2003, pp. 411-418.

About the Authors

Tor J. Larsen earned his Ph.D. in Management Information Systems from the University of Minnesota in 1989. Since then, he has worked as associate professor at the Norwegian School of Management, Department of Leadership and Organizational Management, and has acted as its Head of Department since 2003. During 2001-2002, he was a visiting professor at the John Cook School of Business, Saint Louis University . He has served as an associate editor for *MIS Quarterly* and is currently a member of the editorial board for *Information & Management*. Tor's publications include articles in *Information & Management*, *Journal of MIS*, and *Information Systems Journal*. In 1999, he coedited a book on innovation and diffusion theory for Idea Group Publishing. His professional memberships include AIS, IFIP WG8.2, and WG8.6. He is actively involved in the roles of reviewer and conference program chair. Tor's research interests are in the areas of innovation, diffusion, innovation outcome specification, MIS, and technology mediated learning. He can be reached at Tor.J.Larsen@BI.NO.

Fred Niederman serves as the Shaughnessy Endowed Professor of MIS at Saint Louis University. He received his Ph.D. from the University of Minnesota in 1990. His primary research areas pertain to using information technology to support teams and groups; global information technology; and information technology personnel. He has published more than 30 articles in refereed journal including *MIS Quarterly*, *Communications of the ACM*, and *Decision Sciences*; has presented papers at several major conferences; and serves as associate editor of the *Journal of Global Information Systems*. Fred can be reached at niederfa@slu.edu.

Moez Limayem is a professor at HEC Lausanne. Until recently he was a professor and the BBA Electronic Commerce program coordinator at the Information Systems department of the City University of Hong Kong. Before joining City University of HK, he was the chair of the Management Information Systems department at Laval University in Canada. He holds an MBA and a Ph.D. in MIS from the University of Minnesota. His current research interests include IT adoption and usage, CRM, knowledge management, and electronic commerce. He has had several articles published in journals such as *Management Science*, *Information Systems Research*, *Communications of the ACM*, *IEEE Transactions*, *Accounting, Management & Information Technologies*, *Group Decision and Negotiation*, and *Small Group Research*. He has been invited to present his research in many countries in North America, Europe, Africa, Asia, and in the Middle East. He won the best MIS paper award at the ASAC conference in 1998 and the ICIS conference in 2003. Moez also acts as a consultant for the UNESCO and several private and public companies. He can be reached at moez.limayem@unil.ch.

Joyce Chan earned a Master of Philosophy in Information System from the City University of Hong Kong in 2002. Her primary research areas include knowledge management, electronic business, and small- to medium-sized enterprises. She has presented papers at several major conferences including the Hawaii International Conference on System Sciences. She can be reached at isjoyce@cityu.edu.hk.

Appendix A. Respondent Demographics

R#	Employer	Background	Age	G	Management Responsibility	Title
1	Large accounting/ consulting firm Reports to senior manager or VP Organized by industry sectors	Bachelors in Engineering 1994, MBA96 With firm since 1998	29	M	In projects 2 ½ yrs Focus on success and risk, analyzing requirements	Project manager
2	Medium sized custom manufacturing Reports to IT director New role/division –IT not quite understood Reporting track –president is engineer –CFO –director of IS	BS Computer Science	40	F	Manages two teams of business consultants Supports sales and e-commerce Helpdesk since 09/2001	Business Integration Manager
3	Medium sized custom manufacturing Reports to IT director –president is engineer –CFO –director of IS	Bachelors in Information systems	40	F	Two years tenure Manages 13 developers	Application systems manager
4	Large accounting/ consulting firm	Bachelors in Philosophy Business certificate MBA Employed since 1999 in current role	31	M	Responsibility is in the middle of the hierarchy Last project, 8 people reporting directly, four more reporting in-directly	Development manager
5	Medium sized financial brokerage IT department Has 20-year-old IT in a transition process	Bachelor of Science in electrical engineering	47	F	Mentor for OO/JAVA (80%) Acts as liaison between IT department and business line IT users	Lead developer
6	Medium sized financial brokerage Not critically dependent upon IT, makes money! Outsource accounting (new) Move toward standard packages IT department MM level 0 but not formally measured Data management group	Bachelors in theology and philosophy Masters of divinity 12 years with firm	50	M	Soft supervision, no hiring, etc. Technical rather than management track, more consulting than directing	Senior technical data consultant

R#	Employer	Background	Age	G	Management Responsibility	Title
7	Medium sized health care facility Heavy into IT development Clinical team within IT department	Bachelors of science in nursing1999 MS nursing, emphasis in informatics, 2002	42	F	In charge of project implementation, but no reporting employees	Clinical business analyst
8	Small biotech firm IT department	College 1year, 1980, computer science Was employee; rehired as individual consultant	41	M	None	Contract programmer/ analyst
9	Small biotech firm IT department	BS, Computer Science, 1983 Worked in field since that time	41	M	In charge of project and one other employee	Software engineering manager
10	Large food product company Flattened out budgets IT department 1200 employees	Electrical engineering, 1968	50	M	None Technical consulting	Corporate level software architecture
11	Large food product company Management System Group–Organizational Technical Development	Master's in aerospace engineering	42	M	15 people recently, down to 10 today (08/2002	MIS consultant

Appendix B. An Example of the Analysis Table

Comment #	R	Page #	Statement	Cause	Effect
31	2	4	We're still trying to get people to understand the IS model and our roles because people look at us when we say we need to be part of their planning units and their teams and so far, they look at us "We're not ready for IT yet." We say, "No, no, you don't understand. We're a part of your team. We need to understand prior to when you think you need technology so that we can best help you. It also helps us in our planning." So we've got varying degrees of people understanding our role but it is starting to grasp on. People are starting to understand that role and I can't seem to have enough people to fulfil those roles.	Early involvement	User understanding
32	2	4	Also, recently, in the business integration team, we had taken on process mapping, which makes sense because if you don't understand the process, it's difficult to start mapping out a project and mapping out technology. So we are also facilitating and training process mapping so that's also new.	Process mapping	Project organization

Com-ment #	R	Page #	Statement	Cause	Effect
33	2	6	We had a team, a fairly large team, that pretty much encompassed a good portion of our IS department internally as well as we brought in external consultants because we didn't feel we had the expertise and knowledge in doing every-thing that we were going to be embarking upon.	Addition of consultants	Enhance knowledge base for project work
34	2	7	We had some knowledge but it wasn't practical. It was textbook knowledge and we wanted some practical experience so we hired, actually, multiple consultants. We didn't have one firm. We brought in multiple people. We project managed it but we had a project manager from the consulting firm work with us hand-in-hand.	Addition of consultants	Enhance knowledge base for project work
35	2	7	I thought I could because my background seems like I could do that but I very quickly learned that if I got caught too much in the technical details, I couldn't manage the overall scope of the project.	Focus on overview	Project management success
36	2	9	And we went out and personally trained all 100 distributors when we rolled out phase one. They liked that. They thought that was great.	Personaliza-tion of training	User satisfaction

Appendix C. Categories and Variables Observed in the Study

Category	Variables	
Environment	• Package use/customization • User contract • Physical arrangement	• Task complexity/project difficulty • User characteristics • Project size
Project Management	• OO project management • Early testing • Phased development • Analysis/requirements specification • Requirements gathering	• Scope creep • Task accounting • Metrics • Master scheduling • Quality assurance
Organization	• Organizational culture • Matrix organization	• Change management • Leadership
Staffing	• Multinational teams • Skills and knowledge • Turnover • Developer skills and staffing • Staff learning curve	• Project staffing • Vendor turnover • Staffing patterns • Internal/external staffing
Coordination	• Communication coordination	• Coordination between phases
Mixed Direction Factors	• Reusability/reuse • Maintainability • Simplicity • User satisfaction	• Production speed • Scalability • Cost
Documentation Success	• Documentation quality • Consistency • Utility • Standardization	• Thoroughness • Requirements/technical models linked • Document Management Tool

Category	Variables	
Specific Modeling Tools	*OO/UML* • Sequence diagrams • Activity diagrams • Use Case • Class/Object diagrams	*Traditional* • Prototyping • ER • Hierarchy Charts • DFD
UML/CASE Tool Use	• Quality of use	• Standardization of use

Appendix D. Direct Linkages between OO/UML with Influences and Effects

Respondent	Effect of UML/CASE tools	
1	• Project success • System consistency • Design quality • Use in testing • Extra overhead	• Project size • OO language • Developer preferences • Self-training • Rapid evolution
2	• Project success • Testing effectiveness	• Project size • Hardware capacity • Faith and trust • Training methods
3	• Project success • Reusability • Maintainability • Application success	• Developer acceptance • Hardware/software capacity • Training
4	• CASE tool use • Cost • Standardized use (leads to project success)	• CASE tool use • Cost • Staff skills and knowledge
5	• Project success	• Developer skills • Standardization • Task complexity • Training • Staff skills and knowledge
6	• Project success	• Organizational culture
7	(none observed)	(none observed)
8	• Project success • Documentation success • Skills • Sense of closure on project	• Cost
9	• Project success	• Reusability
10	• Project management success • Development across environments • Component oriented production environment	• Unified platform approach • Training
11	• Project success • Scalability • Reuse • Documentation success	