AGILE SOFTWARE DEVELOPMENT: A SURVEY OF EARLY ADOPTERS

LEO R. VIJAYASARATHY
COLORADO STATE UNIVERSITY
Leo.Vijayasarathy@ColoState.edu

DAN TURK
COLORADO STATE UNIVERSITY
Dan.Turk@ColoState.edu

ABSTRACT

In the past few years, anecdotal evidence of mostly positive experiences with the implementation of agile development methods has emerged from case studies. We used a survey strategy, which enables data gathering from a larger sample of organizations, to learn about the factors driving the adoption and use of agile practices, and their benefits and challenges as percieved by early adopters of this software development methodology. Our survey reveals that personal interest is the driving force behind agile adoption, and conversely, organizational resistance and managerial apathy are key inhibitors. The ability to meet client needs and the delivery of quality software products on time are significant benefits of agile development, while a steep learning curve and its unsuitability for projects characterized by distributed environments and large development teams are identified as minor concerns. This article reports these and other findings accompanied by relevant interpretive analysis.

Keywords: agile development, early adoption, survey

INTRODUCTION

The Agile Manifesto [1] stresses the importance of a) people and interactions over processes and tools, b) working software instead of detailed documentation, c) active customer participation and involvement rather than time and effort expended on negotiating contracts, and d) willingness and ability to take on changes over steadfast commitment to a static plan. Agile software development methods including eXtreme Programming (XP), Scrum, Adaptive Software Development and Feature-Driven Development are based on the principles of the Agile Manifesto and geared towards realizing its goals and objectives.

In general, the feedback from organizations that have implemented agile development is positive. Some of the benefits attributed to agile development are increased productivity, expanded test coverage, improved quality/fewer defects, reduced time and costs, understandable, maintainable and extensible code, improved morale, better collaboration, and higher customer satisfaction. The adoption of agile development has also revealed some challenges such as slow participant buy-in, opposition to pair-programming, lack of detailed cost evaluation, scope creep, reduced focus on code base's technical infrastructure and maintainability, difficulty evaluating and rewarding individual performance, and the need for significant on-site customer involvement, management support, competent managers and developers, and extensive training.

These findings gleaned from case studies and experimental work such as those by Drobka, Noftz and Raghu [3] (Motorola's experiences with XP), Schatz and Abdelshafi [7] (Primavera Systems' adoption of Scrum), and

and Williams, Kessler, Cunningham and Jeffries [10] (comparison of individual and pair programming assignments completed by students at the University of Utah) provide detailed insights into the application of agile software development for specific projects. However, there is a paucity of survey research that can offer more generalizable results on the state of agile development.

Rogers' [6] diffusion of innovations theory categorizes adopters of innovations into innovators, early adopters, early majority, late majority and laggards. While innovators embrace an innovation for its own sake and early adopters have the vision to seize on the innovation as an opportunity to address a pressing need/problem, it is the early majority, that avoids the risks of pioneering but pragmatically weighs the costs and benefits experienced by the early adopters, that determines whether or not the innovation will be widely diffused.

The objective of our study is to tap into a sample of early adopters of agile development to shed light on the factors influencing their adoption decision, the agile methods that they commonly use, and the consequences of their adoption including benefits attained and challenges faced. Our study, with data collected from a relatively large group of respondents, aims to contribute to the bridging of the chasm [5] between early adopters of agile development and the critical mass of pragmatists who may follow them.

EARLIER SURVEY RESULTS

In 2003, Shine Technologies [9], an Australian information technology (IT) consulting firm, conducted a web-based survey to ascertain organizations' interest in agile methods. They received 131 responses from around the world, the majority of whom (84.7%) indicated that they were knowledgeable about agile development. This survey's findings show that XP was the most popular agile method with 59% of respondents using it. An overwhelming majority of the respondents (80% or above) reported that agile processes had improved team productivity, the quality of the applications, and business satisfaction. Further, about half the respondents believed that costs were reduced with the introduction of agile methods. While responsiveness to change and the emphasis on people over processes were identified as positive features of agile development, the lack of structure, planning and documentation were noted as drawbacks. Finally, there was almost unanimous intent among the respondents to continue using agile development or adopt it in the near future.

Digital Focus [2], another IT consulting firm, completed a comparable online survey in 2005 eliciting responses from 136 individuals representing 128 organizations from 17 different countries. About 90% of the respondents of this survey had a basic understanding of agile development practices and 81% were either using or planning to use agile methods in their organizations. Popular motivations for adopting agile included the need to tackle projects with ambiguous and/or evolving requirements, infuse stability to the development process, and to speed-up software delivery. A majority of the participants singled out the ability to cater to change as a key value offered by agile development, and the lack of organizational knowledge and skill as the most pressing challenge to its implementation.

Two other surveys conducted in 2005 provide information about the rate of adoption of agile development. The first, an online poll conducted by MethodsAndTools.com [4], indicate about 40% of the 232 participants' organizations had adopted agile methods and another 20% were evaluating them in pilot projects. The second study, conducted by Schwaber and Fichera [8] for Forrester Research, states that about 14% of North American and European companies were using agile approaches and another 19% were planning to adopt them in the immediate future. This study also concludes that while the early adopters were typically smaller firms generating high-tech products, the recent adopters tended to be information technology groups within larger organizations.

DATA COLLECTION STRATEGY

Data for this study were collected through an anonymous online survey of software development professionals who are most likely to be early adopters of agile development. We identified fifteen Yahoo online discussion groups that focused on agile software development and after obtaining permission from the groups' moderators, posted a solicitation message inviting members who had experience using agile development approaches to complete an anonymous web-based survey.

Respondent Profile

We received 98 responses from software professionals who have an average of 15.5 (median = 15.0) years of experience with software development and 3.9 (median = 3.0) years of agile experience. The majority of the respondents (81) identified their country of residence as the United States of America followed by Canada, India and the United Kingdom with 4 respondents each and Australia, Botswana, Colombia, Mexico and New Zealand with one respondent each. While the respondents had a variety of job titles, including some unique titles such as Agile/XP Coach and Chief Agilist, the most popular cate-

gories are software developer/architect/engineer/programmer (48%), director/manager/leader (34%), and analysts/consultants (11%).

The respondents' organizations cover all the major industry sectors, but most are from the IT (31%) and Finance/Insurance/Real Estate (16%) sectors. The median annual organizational revenues category is US$10 million to US$49 million; the median number of employees in the organizations is 100 to 199; and the median number of software professionals in the organizations is 20 to 49 (Note: respondents were asked to specify a range rather than provide a specific number for each of these three measures).

Our study solicited responses from individuals who are knowledgeable and/or interested in agile development. Therefore, our sample pool of respondents is more representative of early adopters and not of all developers.

Processes and Methods

As might be expected, test-first and XP are reported to be used the most extensively, ranking 5.40 and 5.04, respectively, on a 7-point scale (see Figure 1). Pair Programming is used slightly less extensively, with a ranking of 4.20, followed by Scrum and Agile Modeling with rankings of 3.50 and 3.41, respectively. AUP (the Agile Unified Process) came in last with a ranking of 1.94.

Some of the other methods mentioned by multiple survey respondents include continuous integration (n=4, mean=6.75), Crystal Clear (n=3, mean=6.33), Feature-Driven Development (n=3, mean=6.00), frequent releases/iterative-incremental development (n=4, mean=7.00), and refactoring (n=2, mean=6.50).
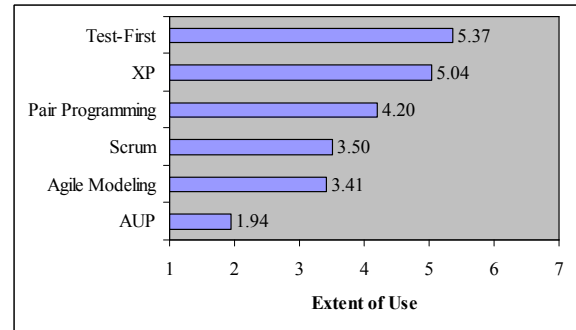

Figure 1: Use of Agile Processes and Methods

Type of Projects

Our respondents specified that agile techniques are mostly used for Internet-based software development projects (70%) and are sparingly used for systems applications (operating systems, compilers, languages), real-time/control systems, and simulators (see Figure 2). In addition, about half the respondents employ agile approaches for front-end (data-entry, GUI-oriented) and back-end (data-processing, batch, non-GUI) systems and about a third utilize them in developing ERP and "stand-alone" systems, such as word-processing or spreadsheet applications.

These results suggest that while agile development is not confined to a particular type of software project, its inherent flexibility and responsiveness may be best suited for Web applications that face rapid changes in both requirements and the facilitating technologies.
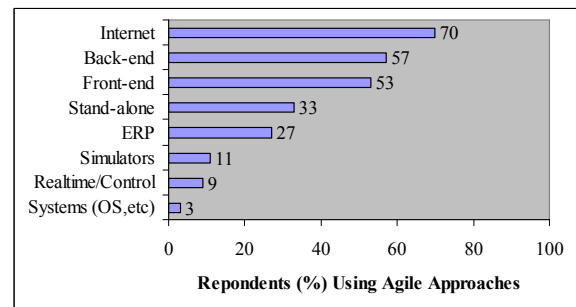

Figure 2: Project Types Supported by Agile Processes and Methods

Agile Use

A majority of the respondents (75%) indicated that they use agile development approaches in half or more of all their projects (see Figure 3). In contrast, only 8% specified that they rely on agile techniques for one out of four projects or less. Since our sample is drawn from members of agile-related discussion groups, this result may not be surprising. However, the finding that a sizable number (60%) employ agile techniques in 75% or more of their projects suggests that in some organizations agile approaches are used almost exclusively for developing software.
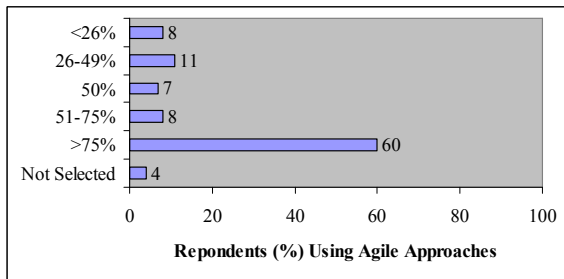

Figure 3: Percentage of Software Development Supported by Agile Processes and Methods

Factors Influencing Agile Adoption

It is intriguing that personal interest is, by far, the most important factor influencing the agile adoption decision, with a mean score of 6.32 on the 7-point survey scale (see Figure 4). The emergence of personal initiative as the primary driver of agile use may be a reflection of the lack of organizational awareness about agile approaches and/or the unwillingness of managers to embrace these newer (and possibly unproven) methods. Among the rest of the factors, peer influence appears to be moderately important with a score of 4.68, underlining the role of a critical mass in the acceptance of innovations.

In addition to rating our list of factors, respondents had the opportunity to mention and rate any other factors that they believed are relevant in agile adoption. Notable among the list generated by the respondents are a) books and seminars by leading proponents of agile including Kent Beck, Alistair Cockburn, Joshua Kerievsky, and Ken Schwaber, b) client demand/interest c) quest for productivity, value, and success, and d) past experience with agile.
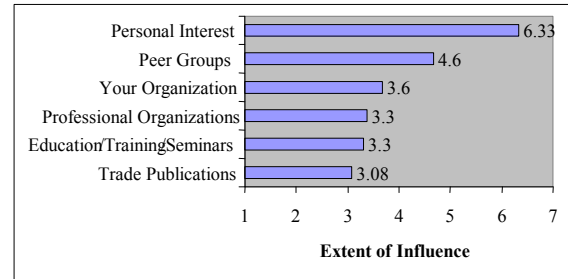

Figure 4: Factors Influencing Adoption of Agile Processes and Methods

Factors Influencing Agile Use

Agile approaches are intended to produce software systems faster while simultaneously (or, more precisely, ambidextrously) anticipating and catering to changes in their requirements. Therefore, it is understandable that project turn-around time, software complexity and the stability of requirements emerge as the most influential factors in organizational decisions to use agile approaches (see Figure 5). In addition, the relatively low importance given to the criticality of the project is probably an indication of the respondents' confidence in the capabilities of agile methods to take on any type of project, including mission-critical ones.

Other factors mentioned and rated highly by respondents include a) corporate/team culture, b) improved communication and collaboration, c) availability of required skill-sets, and d) the failure of traditional development methods such as the waterfall model.
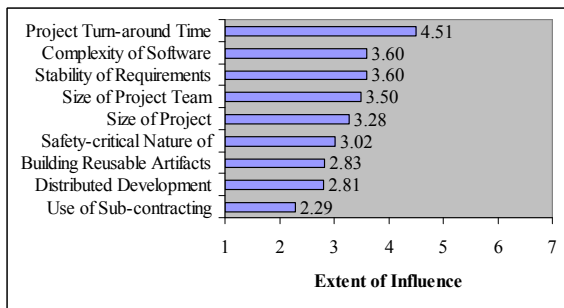
Figure 5: Factors Influencing the Use of Agile Processes and Methods

Problems/Challenges in Adopting and Using Agile

It is apparent that our respondents do not believe that any perceived limitations with agile approaches such as the lack of formal guidelines, inadequate rewards or increased risk of project failure, are inhibitors of their adoption and use (see Figure 6). Rather, they attribute problems with the acceptance of agile development to organizational resistance and managerial disinterest. Lack of training and peer support are also recognized as challenges compounding the view that organizational actions (or the lack thereof) are probably the biggest roadblocks to the adoption and diffusion of agile practices.

Our respondents offer more insight by identifying a) ignorance of agile, b) lack of facilities for pair programming, c) individual resistance and d) the exclusive reliance on economic evaluation criteria as additional barriers to the acceptance of agile methods.
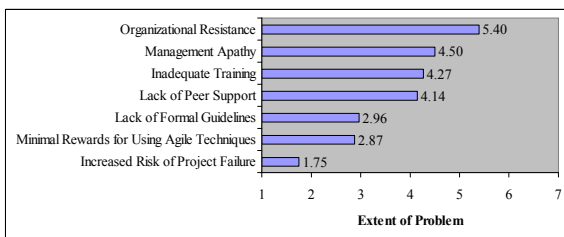

Figure 6: Problems/Challenges in the Adoption and Use of Agile Processes and Methods

Benefits Realized From Agile Use

As an affirmation of its guiding principles, the ability to be flexible and to deliver quality software that meets customer needs faster are recognized as key benefits of using agile development (see Figure 7). The relatively lower rating for reduction in development costs and the production of reusable code is an indication that costs and reusability may not be primary motivations for embracing agile methods and techniques.

Some of the other benefits generated by our respondents are a) increased productivity, b) greater team morale, job satisfaction, and fun, c) improved predictability of schedule/costs/quality, d) knowledge transfer and sharing and e) lower risk of project failure.
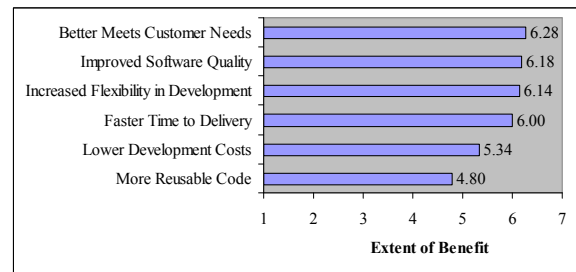

Figure 7: Benefits of Using Agile Processes and Methods

Limitations of Agile Development

In general, our respondents appear to be more upbeat about agile; more convinced about its benefits with few, if any, misgivings about its shortcomings (see Figure 8). Relatively, the limited support for development involving distributed environments and large teams and a steep learning curve may be the only worrisome issues.

Aside from our custom list – few of which are considered serious drawbacks – respondents named the following as limitations to agile development: a) inability to work in CMMI environments, b) incompatibility with development culture, and c) lack of predictive control.
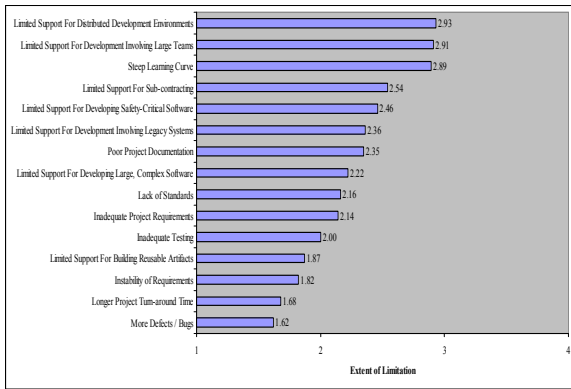
Figure 8: Limitations of Agile Processes and Methods

other key evaluation criteria, with the latter measure emphasizing the iterative nature of agile techniques.
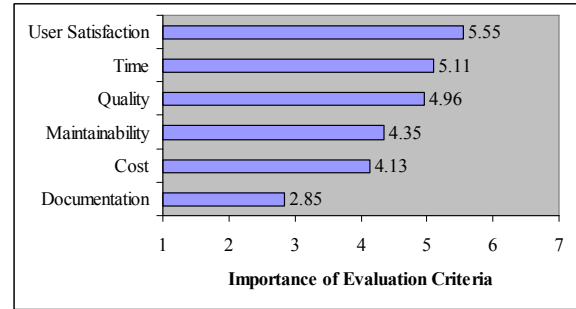

Figure 9: Evaluation Criteria for Agile Projects

Type and Length of Training

A majority of our respondents (90 to 95%) indicated that they have received peer mentoring or taken self-paced or professional development courses in agile development. The average length of peer mentoring is about a month and the time spent on course-based training is typically one to two weeks.

While it is encouraging that almost all respondents appear to have received some training, the relatively small amount of time devoted to learning a new software development approach (that is generally believed to be radically different from traditional methods) is rather puzzling. On the one hand, it may be an indication that agile approaches are easy to learn, and thus don't require a lot of up-front or formal training in order for developers to successfully apply them. On the other, it may be a reflection of the low level of managerial championing that is required for the successful adoption of innovations in organizations.

Project Evaluation Criteria

Congruent with the principle of being responsive to changing client needs, user satisfaction is rated as the dominant criterion for evaluating agile-driven software development projects (see Figure 9). Meeting delivery schedules and producing quality software also emerge as important criteria followed by maintainability and cost. Not surprisingly, documentation, whose role is downplayed in agile development, is rated as least important for assessing agile-assisted projects. Our respondents added delivery of business value and frequency of delivery as

In Their Own Words

At the end of our survey, which consisted mostly of structured questions, we invited our respondents to share additional comments about agile processes and methods and their experiences with using them. More than a third of the survey participants took us up on this offer and provided their valuable observations. While some of these comments are insightful and others memorable, almost all of them reflect the contributors' deep and sincere passion for agile development practices. We reproduce some of our respondents' most perceptive and quotable observations as a qualitative supplement to highlight our quantitative results.

The following quotes offer descriptions of agile development with some arguing that its practice is nothing new:

> "Agile is a set of values and principles and many powerful practices, but all of them have to be adapted to fit to each development situation. There is no recipe, rather a set of cooking techniques that must be adapted based on many variables -- all of which must be understood by the chefs in order to produce a pleasant tasting stew."

> "Agile Processes are not "new" they are trying to strike a balance between common sense, the "urge" to code, and "analysis paralysis"."

> "Agile development is a development strategy where the most important issue is to produce a

software solution that fits exactly what the client needs with the least possible resources and in a very short period of time."

Here are some notable testimonials about the benefits derived from agile development including specifics measures to support their claim:

"It is a joy to develop our software. We use 2 week iterations and release every 2 months. We get so much more done, with higher quality, and team satisfaction. The old way is really the bad way. Never retreat, never go back!"

"Several non-XP teams are considering adopting some form of Agile for future work. Our productivity has increased; we tend to launch major initiatives on time and within or under budget, at a quality level the company has not accomplished in previous major releases."

"We developed a J2EE content management system with 80k lines of code and .075 defects per KLOC (including bugs found in "smoketests"). During the first two weeks of production operation with a global audience we had only 4 trouble tickets -- none related to a defect or system failure. We used XP and Scrum, tailored to our needs."

"After one year doing XP, I don't ever want to go back to wasting my time and my employer's money doing waterfall development."

There are also quite a number of observations about the challenges faced in implementing agile methods. Some of these cautionary notes, which tend to lampoon managerial practices, and lament over unyielding and unsupportive organizational culture, are:

"We are attempting to employ Agile with our state government department, one of the first we are aware of. A lot of resistance from our peers, just another fad is the normal response."

"It is way too easy for management to slip in to the command-and-control regardless of how much they like or buy into agile methods."

"I'm a strong believer in the potential of Agile Methodologies, yet have not found a place where management or fellow programmers are really

interested in trying it. Reading "Men are from Mars, and Women are from Venus", I wonder if Martians' propensity to like doing things on their own is interfering with the team-effort essential to Agile Methodologies - for that matter teamwork anywhere."

"There are 2 kinds of development processes: those that help deliver software successfully, and those that cover your ass. (Okay, most do some of both.) Agile methods are all about the former, while the sad fact is that management gets more mileage out of the latter. Also, software architects may feel threatened by the collaborative egalitarian agile culture, which shifts gradually toward meritocracy (Shudder). Since Agile methods are not much help to people in political power in most organizations, I expect their adoption to happen in little pockets where customers have the most power. Elsewhere, it always falls back to the same Dilbertesque cubicled landscape, with traditional cover-your-ass processes (or no defined process at all)."

"Our largest problems have been lack of interest from other developers. They have little desire to read/learn about agile development techniques."

Finally, at least one of our survey participants expressed skepticism about agile, likening it to yet another software development fad. Here is his/her observation:

"In my own experience at Lockheed, IBM, XOL (a travel agency oriented start up), and FedEx 'processes' are like fad diets - any one of them can work if the personnel are prepared to do them; none of them propose anything essentially new but reaffirm core values of personal cooperation and coordinated application of professional experience/training - the quality of the personnel is the only critical factor."

CONCLUSION

While informed and thoughtful skepticism may be necessary for the proper vetting and adoption of agile practices, the persistent problems of managerial apathy and organizational resistance to change, that are identified in earlier studies and echoed in ours, should be of concern to agile proponents.

Within a short span of time, the theoretical principles of agile development have spawned a number of

practical methods and techniques that appear to be increasingly accepted in organizations, thanks in large part, to the passionate advocacy of individual champions. Nevertheless, in the long run, the sustained realization of agile development's promised benefits will be the key to winning over skeptics and resistors alike, and facilitating its wider dispersion and use.

REFERENCES

[1] Agile Manifesto, "Manifesto for Agile Software Development," http://agilemanifesto.org/, December 2007.

[2] Digital Focus, "Agile 2006 Survey: Results and Analysis," Digital Focus was bought by Command Information in January of 2006.

[3] Drobka, J.; Noftz, D.; & Raghu, R. "Piloting XP on Four Mission-Critical Projects," IEEE Software, Vol. 21, No. 6, 2004, pp. 70-75.

[4] Methods and Tools, "Adoption of Agile Methods," http://www.methodsandtools.com//dynpoll/oldpoll.php?Agile, December 2007.

[5] Moore, G.A. Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers, New York: Harper-Collins, 1991.

[6] Rogers, E.M. Diffusion of Innovations. New York: The Free Press, 1983.

[7] Schatz, B.; & Abdelshafi, I. "Primavera Gets Agile: A Successful Transition to Agile Development," IEEE Software, Vol. 22, No. 3, 2005, pp. 36-42.

[8] Schwaber, C.; & Fichera, R. "Corporate IT Leads the Second Wave of Agile Adoption," http://www.forrester.com/Research/Document/Excerpt/0,7211,38334,00.html, December 2007.

[9] Shine Technologies, "Agile Methodologies Survey Results," http://www.shinetech.com/download/attachments/98/ShineTechAgileSurvey2003-01-17.pdf, December 2007.

[10] Williams, L.; Kessler, R.R.; Cunningham, W.; & Jeffries, R. "Strengthening the Case for Pair Programming," IEEE Software, Vol. 17, No. 4, 2000, pp. 19-25.

ACKNOWLEDGEMENTS

AUTHOR BIOGRAPHIES

Leo R. Vijayasarathy is Associate Professor of Computer Information Systems and First Community Bank Faculty Fellow in the College of Business at Colorado State University. His research has been published in *Electronic Markets, European Journal of Information Systems, IEEE Transactions on Professional Communications, Information & Management, Internet Research, and Journal of Management Information Systems*. He serves on the editorial advisory/review boards of the *International Journal of Cases on Electronic Commerce*, *Internet Research, and Journal of E-Business Research*. He earned an MBA from Marquette University and a Ph.D. from Florida International University.

Dan Turk is Associate Professor of Computer Information Systems in the College of Business at Colorado State University. His research interests are in the areas of computer networking, object-oriented systems, software engineering, business- and system-level modeling, software development process modeling, the value of modeling, process improvement, and the empirical study of these areas. He is a member of the IEEE, the ACM, and the AIS, and has published papers in *IEEE Transactions on Software Engineering*, *Journal of the Association for Information Systems*, *IBM Systems Journal*, *Journal of Database Management*, *L'Objet*, *The Journal of Systems and Software*, *Information Technology & Management*, and the *International Journal of Human Computer Studies*. He earned an M.S. degree in computer science from Andrews University and a Ph.D. degree in business administration (computer information systems) from Georgia State University.