Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2000 Proceedings

Americas Conference on Information Systems (AMCIS)

2000

An Exploration of the Diffusion of Object-Oriented Systems Development

Richard A. Johnson
Southwest Missouri State University, richardjohnson@mail.smsu.edu

John W. Satzinger
Southwest Missouri State University, jws086f@mail.smsu.edu

An Exploration of the Diffusion of Object-Oriented Systems Development

Richard A. Johnson, CIS Dept., Southwest Missouri State University, richardjohnson@mail.smsu.edu
John W. Satzinger, CIS Dept., Southwest Missouri State University, jws086f@mail.smsu.edu

Abstract

This study tests the applicability of Rogers' (1995) diffusion of innovation (DOI) theory to a particular information technology innovation: object-oriented systems development (OOSD). Survey data from 150 experienced developers are analyzed using LISREL structural equation modeling techniques to determine the factors associated with OOSD adoption. Results indicate that the developer's receptiveness to OOSD is the primary determinant of its adoption, followed by the perceived process benefits of easier programming, improved modularity, and flexible development. The influence of one's coworkers and supervisors is shown to have a small but significant effect on the rate of OOSD adoption.

Introduction

A software crisis (Pressman 1992), often characterized by systems that are grossly behind schedule and over budget, has plagued systems development for decades. Various methodologies, tools, and techniques (e.g., prototyping, CASE, and RAD) have been promoted over traditional methods during recent years to address the problem, with but limited success. The most recent software development innovation designed to address the crisis is object-oriented systems development (OOSD). Many proponents (Booch 1994, Coleman et al. 1994, Jacobson et al. 1995, Rumbaugh, et al. 1991) claim that OOSD is a radically different approach to software development that has the potential to improve systems quality and developer productivity by orders of magnitude (Taylor 1992). However, OOSD has yet to realize widespread acceptance in the software development community, especially in the field of business information systems (Pancake 1995).

While there is a plethora of anecdotal evidence concerning the benefits of OOSD, there is little convincing empirical research to support the claims made in the literature. The purpose of this paper is to provide empirical evidence to further an understanding of what makes OOSD appeal to some practitioners and not others. The research question guiding this study is, "What factors are most influential in the adoption of OOSD among individual software developers?"

The theoretical basis for this research is Rogers' (1995) diffusion of innovations (DOI) theory. The results of a survey of 150 systems developers from across the U.S. will be analyzed within the framework of DOI to determine the factors that are most influential in the adoption of OOSD. Such research is important to practitioners since it may assist organizations in the process of adoption of OOSD. This study should also be valuable to IS researchers as it will further an understanding of how DOI may apply to a complex innovation such as OOSD. This study is especially relevant since DOI has heretofore not been applied to a systems development methodology.

This paper will present a brief background on OOSD and DOI followed by an examination of the research methodology employed in the study. A fairly detailed application of LISREL structural equation modeling techniques will be made to the data collected from a survey of experienced software developers. Findings from the analysis will then be presented followed by a discussion of their implications. Finally, limitations of the study and directions for future research will be considered.

Background

Object-Oriented Systems Development (OOSD)

OOSD is considered to be radically different from conventional systems development approaches (Booch 1994, Fichman and Kemerer 1992, Hardgrave 1997, Hodgson 1994). Although many believe OOSD will eventually become the dominant method for all types of systems development (Smith and McKeen 1996), the adoption process has been slow. Recent estimates are that only 17% of organizations are actively using OOSD (Douglas and Massey 1996), although 80% usage is expected by 2001 (*Computerworld*, 1994).

There are many accounts of OOSD advantages, including easier and more effective modeling of problems (Garceau et al. 1993, Khoshafian and Abnous 1995), easier transition between development phases (Eliens 1995), improved communication among developers and users (Coleman et al. 1994, Garceau et al. 1993), improved forms of modularity (Coleman et al. 1994, De Champeaux et al.1993), improved model and code reuse (Coad 1992, Gamma 1995, Khoshafian and Abnous 1995, Tkach and Puttick 1994), improved quality of systems (Coleman et al. 1994), improved maintainability of systems (Chen and Chang 1994), and faster development in the long run (Rumbaugh et al. 1991).

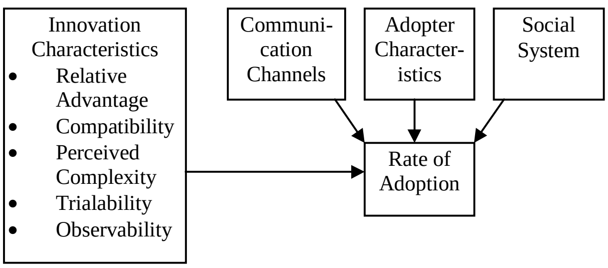
While the literature generally paints a very bright future for OOSD, object-orientation does have its critics. Some authors question the fundamental advantages of

OOSD (Armstrong and Mitchell 1994, Bryant and Evans 1994, Guthery 1989). Others question the feasibility of adopting OOSD given its radical departure from traditional methods (Fichman and Kemerer 1993). Many researchers and practitioners are expressing concerns about the apparent lack of acceptance of what they consider to be a breakthrough IS development methodology (Fayad and Tsai 1995, Fayad et al. 1996, Laubsch 1996, Pancake 1995). The reasons most often cited for the disappointing levels of OOSD acceptance are (1) the apparent difficulty in learning the technology (Douglas and Massey 1996, Fayad and Tsai 1995, Sheetz et. al 1997, Vessey and Conger 1994, Weinberg et al. 1990), (2) the relative immaturity of the technology (Pancake 1995, Smith and McKeen 1996, Weinberg et al. 1990), and (3) the inability to verify the advantages of OOSD (Hardgrave 1997). These factors tend to significantly increase the cost and risk of adopting OOSD. As some put it, object-oriented technology is "still long on hype and short on results" (Smith and McKeen 1996, p. 28).

Diffusion of Innovations Theory

Rogers' (1995) diffusion of innovation (DOI) theory is illustrated in Figure 1. Four primary factors are hypothesized to influence the rate of adoption of a particular innovation. In this study, the target innovation is OOSD. In applying DOI, the unit of adoption can be either the organizational or the individual. In this study, the unit of adoption will be the individual software developer since it is the developer who will ultimately decide the merits of a particular development approach. The individual developer should also have some degree of influence over eventual organizational adoption of a software development innovation.

Figure 1.Diffusion of Innovations Theory (Rogers 1995)



The Innovation Characteristics factor can be subdivided into Relative Advantage (the benefits of the innovation compared to the existing technology), Compatibility (the ability of the innovation to successfully blend with existing processes), Perceived Complexity (the perceived difficulty of learning and successfully implementing the innovation), Trialability (the ability to try the innovation without excessive cost or risk), and Observability (the ability to clearly observe or demonstrate the superiority of the innovation).

Communication Channels refers to how potential users would learn about the innovation. This could be through direct contact with peers or through various forms of media. Adopter Characteristics refers to those facets of the adopting unit that may be instrumental in acceptance of the innovation. If the unit of adoption were the individual, Adopter Characteristics would relate to the special capabilities of developers such as experience or skill. The Social System refers to those individuals who may have influence over the acceptance of the innovation by the potential adopter.

It is apparent that OOSD is an ideal candidate for the application of DOI. Much of the existing OOSD literature relates directly to the factors found in DOI. For example, much has been written about the advantages of OOSD over conventional methods of systems development (as previously cited). The complexity of OOSD, and consequential difficulty of learning OOSD, is likewise often viewed as a serious obstacle to adoption. Managers would especially be concerned about issues of compatibility, trialability, and observability of OOSD within organizations. Communication channels, such as training sessions and in-house mentors, are often presented as very positive influences for successful OOSD implementation. Adopter characteristics, such as openness to new technologies and skill, are frequently cited as important. The social system, including coworkers and managers, should have both direct and indirect influences over the individual's adoption of OOSD. For these reasons, DOI theory was selected as the theoretical framework for this study.

Methodology

Model Variables

For the purpose of this study, DOI will be operationalized by dividing its four major factors into several independent variables, each of which will be hypothesized to directly influence the dependent variable, Rate of Adoption of OOSD. These variables and the numbered survey items used to measure them are listed in Table 1. Note that the specific Innovation Characteristics of Compatibility, Observability, and Trialability have been combined in one construct called "Manageability" since these are issues primarily of concern to managers. Since the survey instrument was part of a wider research effort, the numbered survey items not relevant to this study have been excluded from Table 1. A pictorial representation of the proposed relationships among model variables is provided in Figure 2.

Table 1. Variables and Survey Items Used

DOI Categories	Latent Variables (Constructs)	Observed Variables (Survey Items)
Innovation Characteristics		
Relative Advantage	Modeling Benefits (MODBEN)	C01. Easier modeling w/ OOSD C04. OOSD models more understandable C06. Greater OOA/D model reuse
	Process Benefits (PROCBEN)	C02. Easier programming w/ OOP languages C03. Easier transition from OOA thru OOP C05. More effective code reuse C12. Decreased development time C13. Improved productivity C14. More flexible development
	Product Benefits (PRODBEN)	C07. Improved modularity w/ OOSD C08. Higher system quality w/ OOSD C09. Improved maintainability w/ OOSD C15. Greater stability of designs C16. Better run-time performance C17. Greater user satisfaction
	Communication Benefits (COMMBEN)	C10. Better communication w/ users C11. Better communication w/ developers
Perceived Complexity	Perceived Complexity (COMPLEX)	B11. Confusion of multiple OOA/D methods B12. Complexity of OOA/D B13. Complexity of OOPL
Manageability	Manageability (MANAGE)	B25. Compatibility of OOSD w/ processes B27. Demonstrability of OOSD benefits B28. Observability of OOSD benefits
Communication Channels	Communicability (COMMABILITY)	B17. Availability of OOSD training B18. Opportunity to network w/ OO dev B19. Affordability of OOSD training B23. Availability of OOSD mentor B24. Prominence of OOSD "champion"
Adopter Characteristics	OOSD Skill (OOSKILL)	B07. OOSD experience B08. Skill in OOA/D B09. Skill in OOP
	General SD Skill (GENSKILL)	B02. Exposure to many SD methods B03. Experience w/ conventional SD methods
	Difficulty in Learning OOSD (DIFF)	B05. Difficulty in learning OOA/D B06. Difficulty in learning OOP
	Receptiveness (RECEPTIVE)	B01. Interest in new technologies B04. Openness to OOSD B10. Drive to use OOSD
Social System	Extra-organizational influence (EXTRAORG)	A02. Influence of peers outside the organization A03. Influence of authors you read on OOSD A05. Influence of consultants/trainers in OOSD A06. Influence of vendors in OOSD A07. Influence of potential future employers
	Intra-organizational influence (INTRAORG)	A01. Influence of coworkers A04. Influence of supervisor/manager A08. Influence of your customers/users
Rate of Adoption	Rate of Individual Adoption (ADOPTION)	D09. My personal goal to use OOSD D18. My level of commitment to OOSD D22. My personal intention to use OOSD

Figure 2. Proposed DOI Model Applied to OOSD Adoption

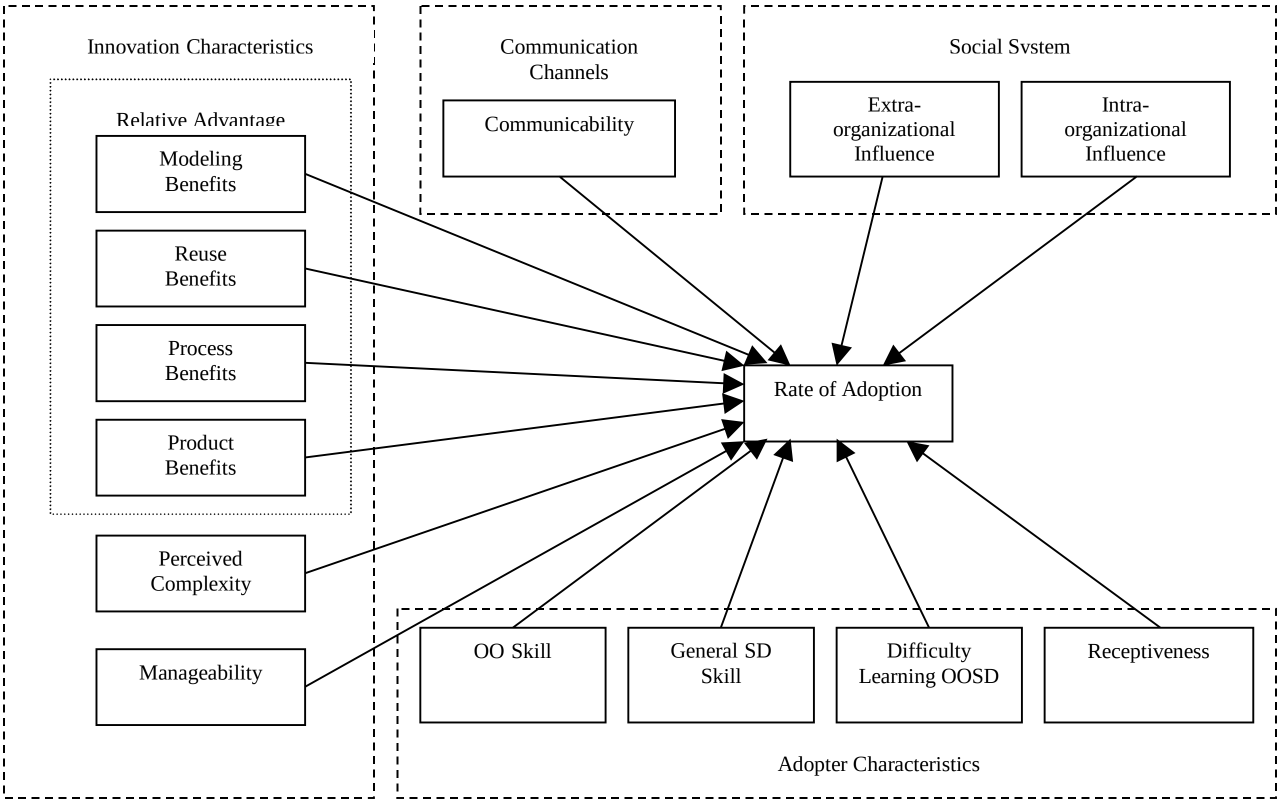
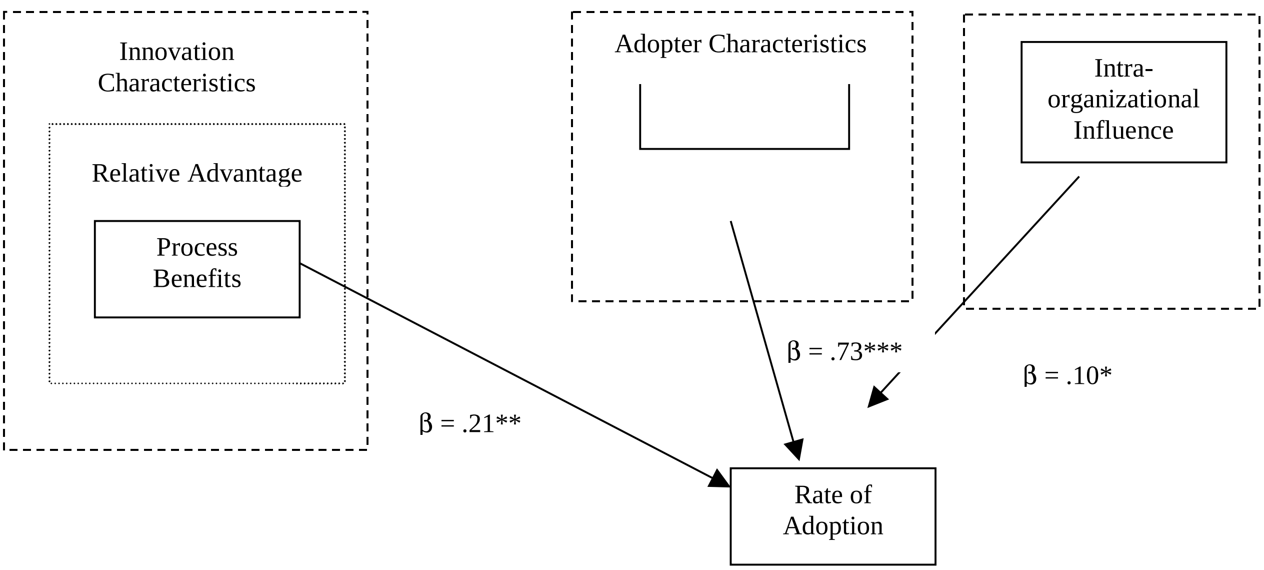


Figure 3. Final DOI Model Applied to OOSD Adoption



*Significant at α=.10
**Significant at α=.05
***Significant at α=.01

The Sample

The population of interest for this study includes all active systems developers including those with and without experience in OOSD. Both types of developers are included to capture a wider range of opinion concerning this emerging technology. Those developers who are not using OOSD can certainly gauge their intention to use it in the future while those who are using OOSD (perhaps at widely differing levels) may have varying intentions to continue using it. In either case, the rate of adoption (intention to use OOSD in the future) can take on a fairly wide range of values (as opposed to an overly simplified adopt-or-not decision).

To ensure a mix of experienced and inexperienced OO developers, the sample frame for this study was selected from two groups. First, a general group of developers was randomly selected from only those subscribers to *Communications of the ACM* who had expressed special interests in software engineering, programming, management information systems, or office information systems. A second group was randomly selected from subscribers to *OOPS Messenger* and registrants at recent OOPSLA conferences. This second group helped to ensure that a sizable percentage of respondents would at least have some experience in OOSD.

Proposed Data Collection and Analysis

A survey instrument was created so that participants could respond to each of the items in Table 1 using a 7-point scale ranging from very unlikely to very likely for the Relative Advantage items, or from very low to very high for all other items. The survey also included several demographic items to indicate characteristics such as gender, age, education, OO and non-OO development experience, job function, and company size.

Analysis of the collected data began by performing a chi-square goodness-of-fit test on the demographic variables of the population (obtained from ACM subscriber data) with those of the sample, and for early vs. late respondents. Such a test should indicate the representativeness of the sample relative to the population.

Further analysis of the data continued using LISREL (version 8.3) software (Joreskog and Sorbom, 1993) to evaluate the measurement model proposed in Table 1 and the structural equation model proposed in Figure 2. Before ordinal, discrete survey variables of this type are analyzed with LISREL, they should first be transformed into continuous variables, using the accompanying PRELIS software, creating an asymptotic covariance matrix (Joreskog and Sorbom, 1993). Failure to do so can seriously skew the results. One drawback, however, of using PRELIS is that the sample size dictates how many observed variables (survey items) can be examined at one time. There are 38 different survey items measuring independent variables in this study, but the sample size of n=150 allows only fifteen to twenty different items to be analyzed at once. This is not a serious limitation since different logical parts of the measurement model may be analyzed in succession.

The first stage of LISREL analysis will establish construct validity through a process of iteratively removing those indicators (observed variables) that do not adequately contribute to the construct (Hair et. al, 1992). For this purpose, LISREL provides many diagnostic statistics including modification indices (MI) for each observed variable that is initially assumed to contribute to a construct. A MI of ten or higher indicates that removing the offending estimate from the model should significantly improve the model's fit to the data.

Table 2 provides the goodness-of-fit (GOF) criteria that will be used in this study. It is usually recommended that a variety of indicator types, as opposed to only one, be used to help ensure overall fit (Hair et. al, 1992). The larger (or smaller) the values, the better the fit of the model to the data.

Table 2. Goodness-of-fit Criteria

GOF Indicators	Recommended Values for Adequate Fit
Root Mean Square Residual (RMR)	< .10
Goodness of Fit Index (GFI)	> .90
Adjusted Goodness of Fit Index (AGFI)	> .80
Parsimony Goodness of Fit Index (PGFI)	< .60

Once the final measurement model for the DOI constructs is established, the second stage of LISREL analysis can begin. The proposed causal links of Figure 2 will be iteratively evaluated to determine which constructs significantly influence the rate of OOSD adoption. Those constructs that do not significantly contribute to an explanation of OOSD adoption will be eliminated from the proposed model. The end result will be a final structural equation model for the adoption of OOSD. The process of eliminating items and constructs from the initial model should shed light on views of experienced developers toward OOSD.

RESULTS

The Sample

The survey and cover letter were mailed to a random selection of 1500 individuals from the original sample frame of 3000. A total of n=150 usable responses were returned representing a 10% response rate. This fairly low response rate was somewhat expected since a strongly stated qualification for participation in the survey was active status as a software developer. Many journal subscribers and conference attendees do not normally meet this qualification. However, demographic data were available for the *CACM* subscribers from which the sample was obtained. These demographic data were compared with the demographic data obtained from the respondents. A chi-square goodness-of-fit test at the $\alpha = .05$ level was conducted for each of the seven demographic variables included in the survey (gender, age, education, OO experience, non-OO experience, current job function, and company size) resulting in the failure to reject the null hypothesis that the sample is representative of the population. Early and late respondents were also compared using the chi-square test. No differences in these groups were detected. Thus, although the response rate is low, the sample appears to be highly representative of the population.

The Measurement Model

To begin to establish a valid measurement model, the seventeen items in Table 1 relating to Relative Advantage were analyzed as a group. Table 3 summarizes the results of ten iterations of a LISREL analysis applied to these items. The actions taken as a result of applying LISREL's diagnostic features are listed.

Table 3. Successive Versions of the Measurement Model for Relative Advantage

Version 1. Original measurement model for Relative Advantage (17 items)
Version 2. Remove "C03— Easier transition from OOA thru OOP"
Version 3. Remove "C17— Greater user satisfaction"
Version 4. Remove "C16— Better run-time performance"
Version 5. Combine "C05— More effective code reuse" and "C06— Greater OOA/D model reuse" into a new "reuse" construct
Version 6. Remove "C15— Greater stability of designs"
Version 7. Remove "C10— Better communication w/ users," make "C11— Better communication w/ developers" a process benefit
Version 8. Move "C07— Improved modularity w/ OOSD" to a process benefit
Version 9. Remove "C12— Decreased development time"
Version 10. Remove "C11— Better communication w/ developers"

The changes summarized in Table 3 indicate that the removed items may not be valid indicators of Relative Advantage. A total of seven of the original seventeen items were removed. It was not until Version 10 that the goodness-of-fit indicators were all in the acceptable range.

The next step in establishing a valid measurement model was to add the six items of Perceived Complexity and Manageability to the remaining ten items for Relative Advantage. This provided a valid measurement model for the entire Innovation Characteristics category of DOI. A LISREL analysis was performed on the combined sixteen items resulting in the additional actions taken as listed in Table 4.

Table 4. Successive Versions of the Measurement Model for Innovation Characteristics

Version 1. Measurement model for Relative Advantage, Perceived Complexity, and Manageability (16 items)
Version 2. Remove "C06-- Greater OOA/D model reuse" and move "C05-- More effective code reuse" to a process benefit
Version 3. Remove "C05-- More effective code reuse"
Version 4. Remove "B13— Complexity of OOPL"
Version 5. Remove "C12— Decreased development time"

As indicated by Table 4, a total of four of the remaining ten items for Relative Advantage were removed and one of the original three items for Perceived Complexity was removed. Version 5 of the measurement model for Innovation Characteristics was the first to yield goodness-of-fit indicators that were all in the acceptable range.

Next is the analysis of the measurement model for Communication Channels and Adopter Characteristics, which contains fifteen items. Table 5 displays the results of six successive refinements of this model.

Table 5. Successive Versions of the Measurement Model for Communication Channels and Adopter Characteristics

Version 1. Measurement model for Communication Channels and Adopter Characteristics (15 items)
Version 2. Remove “B23— Availability of OOSD mentor”
Version 3. Remove “B01— Interest in new technologies”
Version 4. Remove “B18— Opportunity to network w/ OO developers”
Version 5. Remove “B03- Experience w/ conventional SD methods,” move "B02-- Exposure to many SD methods " to OO Skill
Version 6. Remove “B02-- Exposure to many SD methods”

Of the fifteen original items within the Communication Channels and Adopter Characteristics constructs, a total of five were removed yielding an acceptable fit. The eight items for Social System were next examined for the validity of the measurement model resulting in the findings in Table 6.

Table 6. Successive Versions of the Measurement Model for Social System

Version 1. Original measurement model for Social System (10 items)
Version 2. Remove “A02— Influence of peers outside the organization”
Version 3. Remove “A08— Influence of your customers/users”

Of the eight original items for Social System, six survived yielding a measurement model satisfying all goodness-of-fit criteria. Next, these six Social System items were combined with the ten surviving Communication Channels and Adopter Characteristic items to form a larger measurement model for continued analysis. The results are shown in Table 7.

Table 7. Successive Versions of the Measurement Model for Communication Channels, Adopter Characteristics, and Social System

Version 1. Measurement model for Communication Channels, Adopter Characteristics and Social System (16 items)
Version 2. Remove “A07— Influence of potential future employers”
Version 3. Remove “A05— Influence of consultants/trainers in OOSD”

Only two items were removed from the original sixteen in order to achieve an acceptable fit. At this point, separate (acceptable) measurement models existed for Innovation Characteristics and for the combination of Communication Channels, Adopter Characteristics, and Social System. These DOI categories cannot be combined for further analysis due to the limitation of sample size in this study. However, one can be relatively assured of the validity of the constructs after such a lengthy and careful analysis. Table 8 presents the final values of item loadings (λ’s) for the factor pattern matrix (i.e., the loading of each survey item on its respective hypothesized construct). These λ’s are analogous to partial standardized regression coefficients in a multiple regression analysis. The larger the loading, the greater the influence that a particular observed variable has on its construct. Table 9 presents the final goodness-of-fit statistics for the two measurement models.

Table 8. Final λ Values for DOI Model Variables

DOI Categories, Latent Variables, and Observed Variables	**Loading (λ)**
Innovation Characteristics	
Relative Advantage	
Modeling Benefits	
C01. Easier modeling w/ OOSD	.90
C04. OOSD models more understandable	.76
Process Benefits	
C02. Easier programming w/ OOP languages	.66
C07. Improved modularity w/ OOSD	.88
C13. Improved productivity	.74
C14. More flexible development	.79
Product Benefits	
C08. Higher system quality w/ OOSD	.93
C09. Improved maintainability w/ OOSD	.91
Perceived Complexity	
B11. Confusion of multiple OOA/D methods	.68
B12. Complexity of OOA/D	.98
B13. Complexity of OOPL	.68
Manageability	
B25. Compatibility of OOSD w/ processes	.49
B27. Demonstrability of OOSD benefits	.69
B28. Observability of OOSD benefits	.51
Communicability	
B17. Availability of OOSD training	.55

B18. Opportunity to network w/ OO developers	.68
B19. Affordability of OOSD training	.55
Adopter Characteristics	
OOSD Skill	
B07. OOSD experience	.87
B08. Skill in OOA/D	.96
B09. Skill in OOP	.79
Difficulty Learning OOSD	
B05. Difficulty in learning OOA/D	.86
B06. Difficulty in learning OOP	1.01
Receptiveness	
B04. Openness to OOSD	.73
B10. Drive to use OOSD	.87
Social System	
Extra-organizational Influence	
A03. Influence of authors you read on OOSD	1.02
A06. Influence of vendors in OOSD	.57
Intra-organizational Influence	
A01. Influence of coworkers	1.03
A04. Influence of supervisor/manager	.69

Table 9. Goodness-of-fit Indicators for DOI Measurement Models

Indicator	*Measurement Model for Innovation Characteristics*	*Measurement Model for Communication Channels, Adopter Characteristics, and Social System*
RMR	.034	.063
GFI	.90	.91
AGFI	.82	.85
PGFI	.48	.54

The Structural Equation Model

Now that an acceptable measurement model has been established for the DOI variables, the structural equation model (SEM) for DOI will be investigated. The goal of the SEM analysis is to determine the statistical significance and strength of the theorized causal links between the independent variables and the dependent variable. It was hypothesized that each construct in the original model (Figure 2) would have a direct effect on the rate of OOSD adoption. As a result of establishing a valid measurement model for the DOI constructs, two of the original variables (Reuse Benefits and General SD Skill) were eliminated from consideration since their indicators were removed. The SEM analysis was performed in two parts, one for a combination of Innovation Characteristics and Rate of Adoption constructs, the other for a combination of Communication Channels, Adopter Characteristics, Social System, and Rate of Adoption constructs. Again, this partial analysis was necessitated by the limitation of including only fifteen to twenty observed variables in a model at a time.

Using the method of successive model refinement suggested by Joreskog and Sorbom (1993), constructs with standardized regression coefficients (β's) that are statistically insignificant are removed one at a time from the analysis until only those constructs with statistically significant β's remain. Any additional offending estimates of model construct indicators (survey items) that are uncovered in this analysis will be subject to removal as well. Table 10 summarizes the results of a SEM analysis for Innovation Characteristics and Rate of Adoption.

Table 10. Successive Versions of the SEM Model for Innovation Characteristics and Rate of Adoption

Version 1. SEM model for Innovation Characteristics and Rate of Adoption (see Figure 2)
Version 2. Remove Manageability construct
Version 3. Remove “C01— Easier modeling w/ OOSD” item
Version 4. Remove Modeling Benefits construct
Version 5. Remove Product Benefits construct
Version 6. Remove Perceived Complexity construct

The only variable shown to have a significant effect on Rate of Adoption is Process Benefits with a rather strong $\beta = .71$. A similar SEM analysis was then performed on the combination of Communication Channels, Adopter Characteristics, Social System, and Rate of Adoption. Table 11 details the results of this analysis.

Table 11. Successive Versions of the SEM Model for Communication Channels, Adopter Characteristics, and Social System

Version 1. SEM model for Communication Channels, Adopter Characteristics, Social System, and Rate of Adoption
Version 2. Remove Extra-organizational Influence construct
Version 3. Remove Difficulty in Learning OOSD construct
Version 4. Remove “B19— Affordability of OOSD training” item
Version 5. Remove Communicability construct

The final step in performing the SEM analysis is to combine all surviving constructs from Innovation

Characteristics, Communication Channels, Adopter Characteristics, and Social System into one model with Rate of Adoption as the dependent variable. Table 12 presents the step-by-step results of this LISREL analysis.

Table 12. Successive Versions of the SEM Model for All DOI Constructs

Version 1.	SEM model for Innovation Characteristics, Communication Channels, Adopter Characteristics, Social System, and Rate of Adoption
Version 2.	Remove OOSD Skill construct
Version 3.	Remove "C13—Improved productivity" item

Figure 3 presents the final results of the SEM analysis for the diffusion of OOSD. Table 13 provides the final indicators of goodness-of-fit for the two separate SEM models and the final combined SEM model.

Table 13. Goodness-of-fit Indicators for SEM Models

Indicator	Innovation Characteristics	Communication Channels, Adopter Characteristics, Social System	All Independent Variables
RMR	.033	.039	.042
GFI	.95	.93	.96
AGFI	.90	.84	.92
PGFI	.44	.49	.50

Discussion

Findings

Figure 3 presents the final model resulting from this investigation. Most importantly, DOI does appear to have fairly broad applicability to OOSD. Three of the four major categories of variables in the DOI were found to influence the rate of OOSD adoption. Only Communication Channels does not appear to have a significant influence. This finding may be due to the possibility that members of the systems development community are often fairly independently minded and formulate many of their beliefs about the credibility and applicability of a particular systems development approach on their own.

An equally important finding is that many of the variables within each major DOI category do not influence the rate of OOSD adoption as suggested by much of the OOSD literature. For example, many of the purported benefits of OOSD (modeling, reuse, and productivity), while possibly valid, do not appear to motivate adoption. Other factors, such as the perceived complexity and the difficulty of learning OOSD, likewise appear to have little influence on the rate of OOSD adoption. It is encouraging that perceived complexity and difficulty of learning do not exhibit a negative influence over OOSD adoption.

The most influential variable in the model is that of Receptiveness. This is particularly interesting since receptiveness is much more of a psychological human factor than elements such as reusability or maintainability. This finding indicates that if a developer is simply more open to OOSD and has a personal drive to use OOSD, he is much more likely to embrace the new technology. Such receptiveness could be enhanced through management practices.

To a much lesser extent, the Process Benefits of OOSD (easier programming, improved modularity, and flexible development) appear to influence the rate of OOSD adoption. These findings corroborate much of the OOSD literature that cites modularity and flexible development as some of the primary benefits of OOSD. However, one does not find much in the literature regarding the ease of OO programming. OO developers may feel that OO programming is at least easier than expected, tending to increase their acceptance of OOSD.

Finally, intra-organizational influences (coworkers and supervisors) have a very small, though statistically significant, influence over the rate of OOSD adoption. This finding makes sense, as the acceptance of OOSD among these important groups should have a positive affect on one's adoption.

Limitations

There are many limitations to this present study. First, the data were collected using the survey research method where respondents self-report their perceptions. The survey research method does, however, have wide applicability in IS research. Additionally, DOI theory is predicated on adopters' perceptions. The sample size of n=150 did constrain the analysis to be performed in a piecemeal fashion. A much larger sample size would have allowed all variables of the model to be analyzed concurrently. However, great care was taken to study logical parts of the model together and recombine them where possible.

Future Research

Future research should seek to increase the sample size to a more acceptable level and to possibly focus on different groups of developers, such as those with little OOSD experience vs. those with substantial OOSD experience. Of particular interest would an investigation of why some developers are more receptive to systems development innovations than others.

Conclusion

This study has been largely successful in demonstrating that Rogers' (1995) diffusion of innovations (DOI) theory is indeed applicable to a relatively radical and complex innovation in software development—OOSD. A group of 150 experienced developers evaluated a very wide range of issues potentially related to the adoption of OOSD. The findings suggest that innovation characteristics, adopter characteristics, and the social system associated with OOSD all have a significant impact on the rate of OOSD adoption. However, communication channels and many other factors suggested in the OOSD literature were found not to have a sizable influence on OOSD adoption. By far the most potent factor in OOSD adoption is simply the receptiveness of the developer to this new technology. Developers are also influenced slightly by some of the purported benefits of using OOSD and by the views of others within the organization.

References

Armstrong, J.M. and Mitchell, R.J. (1994). Uses and abuses of inheritance. Software Engineering Journal, 9:1 (January), 19-26.

Booch, G. (1994). Object-oriented analysis and design with applications, 2nd ed. Benjamin/Cummings (Redwood City, CA).

Bryant, T. and Evans, A. (1994). OO oversold: those objects of obscure desire. Information and Software Technology, 36:1, 35-42.

Chen, J. and Chang, S. (1994). An object-oriented method for software maintenance. Journal of Object Oriented Programming, 6:8 (January), 46-51.

Coad, P. (1992). Object-oriented patterns. Communications of the ACM, 35:9 (September), 152-159.

Coleman, D., Arnold, P., Bodoff, S., Dollin, C., Gilchrist, H., Hayes, F. and Jeremaes, P. (1994). Object-oriented development: the fusion method. Prentice-Hall (Englewood Cliffs, NJ).

Corporate IS considers object plans (1994). Computerworld, 28:4 (January 24), 61, 64.

De Champeaux, D., Lea, D., and Faure, P. (1993). Object-oriented system development. Addison-Wesley (Reading, MA).

Douglas, D. and Massey, P. (1996). Is industry embracing object-oriented technologies? Journal of Computer Information Systems, 36:3 (Spring), 65-72.

Eliens, A. (1995). Principles of object-oriented software development. Addison-Wesley (New York).

Fayad, M.E. and Tsai, W. (1995). Object-oriented experiences. Communications of the ACM, 38:10 (October), 51-53.

Fayad, M., Tsai, W. and Fulghum, M. (1996). Transition to object-oriented software development. Communications of the ACM, 39:2 (February), 108-121.

Fichman, R.G. and Kemerer, C.F. (1992). Object-oriented and conventional analysis and design methodologies: comparison and critique. IEEE Computer, 25:10 (October), 22-39.

Gamma, E. (1995). Design patterns: elements of reusable object-oriented software. Addison-Wesley (Reading, MA).

Garceau, L., Jancura, E., and Kneiss, J. (1993). Object-oriented analysis and design: a new approach to systems development. Journal of systems management, 44:1 (January), 25-33.

Guthery, S. (1989). Are the emperor's new clothes object-oriented? Dr. Dobb's Journal, 4:12, 80-86.

Hair, J.F., Jr., Anderson, R.E., Tatham, R.L. and Black, W.C. (1992). Multivariate data analysis with readings. Macmillan (New York).

Hardgrave, B.C. (1997). Adopting object-oriented technology: evolution or revolution? The Journal of Systems and Software, 37, 19-25.

Hodgson, R. (1994). Contemplating the universe of methods. In A. Carmichael (Ed.), Object Development Methods (pp. 111-132). SIGS Books (New York).

Jacobson, I., Christerson, M., Jonsson, P., and Overgaard, G. (1995). Object-oriented software engineering: a use case driven approach, 2nd ed. Addison-Wesley (Wokingham, England).

Joreskog, K. and Sorbom, D. (1993). LISREL 8: structural equation modeling with the SIMPLIS command language. Erlbaum (Hillsdale, NJ).

Khoshafian, S. and Abnous, R. (1995). Object orientation. Wiley (New York).

Laubsch, J. (1996). A beginner's guide to developing with the Taligent application frameworks. Journal of Object-Oriented Programming, 9:7 (Nov/Dec), 35-42.

Pancake, C.M. (1995). The promise and the cost of object technology: a five-year forecast. Communications of the ACM, 38:10 (October), 33-49.

Pressman, R.S. (1992). Software engineering--A practitioner's approach, 3rd ed. McGraw-Hill (New York).

Rogers, E.M. (1995). Diffusion of innovation. (4th Ed.). The Free Press (New York).

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., and Lorensen, W. (1991). Object-oriented modeling and design. Prentice Hall (Englewood Cliffs, NJ).

Sheetz, S.D., Irwin, G., Tegarden, D.P., Nelson, H.J., and Monarchi, D.E. (1997). Exploring the difficulties of learning object-oriented techniques. Journal of Management Information Systems, 14:2 (Fall), 103-131.

Smith, H.A. and McKeen, J.D. (1996). Object-oriented technology: getting beyond the hype. DATA BASE, 27:2 (Spring), 20-29.

Taylor, D. A. (1992). Object-oriented information systems: planning and implementation. Wiley & Sons (New York).

Tkach, D. and Puttick, R. (1994). Object technology in application development. Benjamin/Cummings (Redwood City, CA).

Vessey, I. and Conger, S. (1994). Requirements specification: Learning object, process, and data methodologies. Communications of the ACM, 37:5 (May), 102-113.

Weinberg, R., Guimaraes, T., and Heath, R. (1990). Object-oriented systems development. Journal of Information Systems Management, 7:4, 18-26.