1993

WHAT INFLUENCES REGULAR CASE USE IN ORGANIZATIONS? An Empirically Based Model

Carsten Sørensen

Riso National Laboratory, Denmark, carsten@risoe.dk

Follow this and additional works at: http://aisel.aisnet.org/sjis

Recommended Citation

Sørensen, Carsten (1993) "WHAT INFLUENCES REGULAR CASE USE IN ORGANIZATIONS? An Empirically Based Model," Scandinavian Journal of Information Systems: Vol. 5 : Iss. 1 , Article 3. Available at: http://aisel.aisnet.org/sjis/vol5/iss1/3
WHAT INFLUENCES REGULAR CASE USE IN ORGANIZATIONS?
An Empirically Based Model

CARSTEN SØRENSEN
Cognitive Systems Group
Riso National Laboratory
DK-4000 Roskilde, Denmark
carsten@risoe.dk

Abstract

The diffusion of a tool for Computer Aided Software Engineering (CASE) in an organization is affected by a multitude of technical and organizational factors. The aim of this paper is to build a statistical model expressing relationships between: The extent of CASE implementation, organizational diffusion activities, modification of the CASE tool, experience in using the tool, and the size of the information systems department. The analysis of dependencies are conducted by discrete graphical modeling of empirical data collected in a survey. CASE is not diffused in most of the organizations. In 61% of the organizations, a quarter or less of the developers use the tools. The analysis indicates that investing time and effort in introducing the CASE tool, and changing the work procedures can be rewarded by an increased extent of implementation. Modification of CASE tool functionality is not found to be an important issue in the diffusion process.

Keywords: Software engineering, CASE diffusion in organizations, CASE survey.
1 Introduction

The effects of a technological innovation can be ascribed to other aspects than just the technical properties of the innovation. As put by Rogers (1983): "One reason why there is so much interest in the diffusion of innovations is because getting a new idea adopted, even when it has obvious advantages, is often very difficult".

Computer Aided Software Engineering (CASE) tools offers a potential for software producing organizations. CASE tools could prove to be an important step towards enhancing both the software development process, as well as the products. The process of implementing a CASE tool into the software organization is, however, contingent on a mix of factors, e.g. the profitability of the tool, the scale of investment, technical characteristics, acceptability, change agents, and developer qualifications (Aaen & Sørensen 1991).

According to Aaen et al. (1992), CASE is only used by a minority of developers in CASE organizations. Therefore, the primary problem analyzed in this paper is organizational activities affecting the process of diffusing the CASE tool among developers once it is purchased. The paper focuses on organizational adaption of CASE tools, and analyzes relationships between the extent of CASE implementation, as the dependent variable, and the following characteristics of the implementation process, independent variables: (1) The activities applied in order to support developers in using the tool, (2) whether or not the functionality of the CASE tool has been modified, and (3) organizational characteristics. The variables describing the introduction process are: The number of activities applied when introducing the CASE tool to developers, the number of user support functions, whether or not a methodology expert is appointed, and whether or not work procedures change as a result of introducing the CASE tool in software projects. The organizational characteristics consists of: The number of CASE projects and the size of the information systems (IS) department. Because of the statistical technique used, the model also describes dependencies between independent variables.

CASE implementation has been studied in several projects using different research methods such as: a theoretical approach, case studies, interviewing developers in a small number of companies, experiments, and by questionnaire surveys analyzing a broad spectrum of experiences with CASE introduction. Despite the efforts in the CASE field, few empirically based models describing the organizational diffusion of CASE have been promoted (Wynekoop & Conger 1990, Wynekoop et al. 1992). This paper documents the process of building an empirically based model. The data used is from a survey among 64 Danish and 38 Finish CASE users conducted in 1991. A structured questionnaire containing 45 questions with a total of 200 attributes for each observation was issued. The survey is documented by Aaen et al. (1992) and further methodological aspects of the survey are documented by Sørensen (1992). The statistical method applied is discrete graphical modeling where an initial model assuming dependencies be-
tween all selected factors is iteratively reduced. The resulting model shows both dependent and conditionally independent factors.

In the following section, a conceptual framework is presented. Section 3 presents related CASE research. In Section 4 the initial model is argued and presented. Section 5 presents the use of graphical modeling in reducing the initial model. Section 6 presents and discusses the final model, and Section 7 concludes the paper.

2 Conceptual Framework

Information systems implementation can be viewed as an organizational effort to diffuse an information technology within a user community (Kwon & Zmud 1987). Taking this view, the organizational diffusion of CASE technology then becomes a process of diffusing information technology innovations aimed at software engineers as the user community (Wynekoop et al. 1992). This stance leads to focusing on the level of the software organization stressing both technical and organizational factors affecting the outcome of the implementation process.

Implementing CASE tools in software organizations imply changes in both the development process, i.e. necessary changes in work procedures according to the methodology supported by the CASE tool, as well as in the products such as specification documents (Aaen & Sørensen 1991, Aaen et al. 1992).

Theories on the diffusion of technological innovations describe the diffusion process as a sequence of phases or as a set of activities. They suggest which factors are important for the actual outcome of the diffusion process. As suggested by several authors, e.g. Aaen & Sørensen (1991) and Orlikowski (1989), the success of the CASE diffusion process depends on complex relationships between a multitude of both organizational and technical factors. Zmud (1982) points out that organizational innovation in general is influenced by a variety of factors and that different studies of similar factors have led to inconsistent results.

The framework suggested by Kwon & Zmud (1987) and developed further by Cooper & Zmud (1990) contains six stages describing the diffusion process. The stages can also be interpreted as activities with less strict assumptions about temporal dependencies (Cooper & Zmud 1990). For the purpose of this paper it is not crucial whether or not diffusion processes in general can be described as a set of consecutive stages, and the categories in the framework will here primarily be interpreted as activities. Discussions pro et con have been documented by, for example, Rogers (1983) and Peltz (1983). The six activities in Cooper & Zmud’s (1990) framework have been slightly modified for CASE as the technological innovation to be diffused, Table 1):

**Initiation:** marks the initial activities and can be characterized by organizational pull or technology push.

**Adoption:** the process of deciding to acquire and implement CASE.
<table>
<thead>
<tr>
<th>Process</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiation</td>
<td>Active and/or passive scanning of organizational problems/opportunities and CASE solutions are undertaken. Pressure to change evolves from either/or both: organizational need (pull), e.g., a need to have computer aided support for the methodologies applied; and technological innovation (push), e.g., the CASE tool as a remedy to enhance the methodological foundation.</td>
</tr>
<tr>
<td>Adoption</td>
<td>Rational and political negotiations to ensue to get organizational backing for implementation of the CASE tool.</td>
</tr>
<tr>
<td>Adaption</td>
<td>The CASE tool is purchased, installed, and maintained. Organizational procedures are revised and developed. Organizational members are trained both in the new procedures and in using the tool.</td>
</tr>
<tr>
<td>Acceptance</td>
<td>Organizational members are induced to commit to CASE usage.</td>
</tr>
<tr>
<td>Routinization</td>
<td>Usage of the CASE tool is encouraged as normal activity.</td>
</tr>
<tr>
<td>Infusion</td>
<td>Increased organizational effectiveness is obtained by using CASE in a more comprehensive and integrated manner to support higher level aspects of systems development.</td>
</tr>
</tbody>
</table>

Table 1: Application of Cooper & Zmud’s model to CASE introduction.

**Adaption:** buying the tool and training developers.

**Acceptance:** developers are induced to use the tool.

**Routinization:** CASE usage is considered a normal part of development projects.

**Infusion:** using CASE to its fullest potential.

Zmud (1982) describes the diffusion of innovations in terms of the three phases suggested by Thompson (1965): (1) Initiation, (2) adoption, and (3) implementation.
tation. This tripartite is the basic distinction in the innovation diffusion theory (Wynkoop & Senn 1992). When the term implementation is used in the following it covers the last four activities in Cooper & Zmud’s framework. The framework is, at least on an abstract level, fairly representative of several innovation process frameworks, e.g. frameworks suggested by Rogers (1983), Zaltman et al. (1973), and Scheier (1983). It is, however, important to note that one of the basic assumptions of Cooper & Zmud’s framework is that as the diffusion process progresses the organization will experience increased productivity and quality. The framework does not take into account that the whole organization or single individuals might experience a setback compared to before the innovation was introduced.

Cooper & Zmud’s framework is only used in order to characterize the totality of the CASE diffusion process. This article focuses on the diffusion of CASE tools among software developers. This implies that activities concerning with initiation and adoption activities are not considered.

3 Related Research

Several research projects have been studying different aspects of CASE. Experiments collecting quantitative data on the properties of CASE tools have been conducted by Vossey et al. (1992) and Norman & Nunamaker (1988, 1989a, 1989b). Orlikowski (1988, 1989) uses observation as research method when studying the organizational implications of CASE. Smolander et al. (1991) use interviews on methodology adaptation in a small number of organizations as empirical research methodology. CASE surveys have in USA been performed by Burkhard & Jenster (1989), Herbert & Browdy (1989), and Wynkoop (Wynkoop 1992a). In Holland Wijers & van Dort (1990) have conducted a CASE survey. Undocumented surveys are, or have been, performed in USA (Chan 1992) and in Great Britain (Majumdar 1992).

Most of the efforts in the CASE field do, however, not aim at building and testing empirically based CASE implementation models. If we look at studies of innovation diffusion in general, they mainly focus on innovation characteristics. Tornatzky & Klein (1982) forward a need to study other independent variables such as, for example, characteristics of the innovation process.

The analysis by Wynkoop et al. (1992) is one of the few exceptions from the rule. It investigates relationships between the extent of CASE implementation, as the dependent variable, and the five independent variables:

1. Perceived innovation characteristics: relative advantage and complexity.
2. Expectation gap: differences between expectations and experiences.
3. Organizational effort, communication.
4. Perceived management commitment.
5. Months the tool was used.

The dependent variable consists of both acceptance and level of use.

The focus in this paper is similar to the one chosen by Wynekoop et al. (1992). Both focuses on the extent of CASE implementation among developers in the organization as the dependent variable. There are, however, major differences between the two.

Firstly, Wynekoop et al. operationalizes the extent of implementation variable by measuring level of use and tool acceptance. As described in the next section, this article operationalizes extent of implementation by measuring the percentage of developers using CASE on a regular basis.

Secondly, this paper focuses on attributes of the CASE adaption process. Wynekoop et al. focus on factors describing preconditions for the implementation process. They focus on measuring differences in developers’ perceptions before and after CASE adoption. The only exception is the organizational communication measure, which characterizes the amount of communication between developers and managers during the implementation process.

Thirdly, the primary decision-making unit (Rogers 1983) is in this paper the software organization and in Wynekoop et al. the individual developer.

Fourthly, Wynekoop et al. aim at testing a model established based on a theoretical line of argumentation. This article aims at building a model using empirical data. Only the components in the model are based on a theoretical line of argumentation. The suggested model resulting from this analysis can then be tested using other data.

4 Initial CASE Adaption Model

The main focus of the analysis documented in this paper is after a CASE tool has been chosen, and the decisions have been made regarding the investment of resources in implementing it in the organization, i.e. in Cooper & Zmud’s (1990) concepts after initiation and adoption. The main idea is to establish a model explaining relations between the extent of CASE diffusion in the organization, CASE diffusion activities, and organizational characteristics. It is the goal to assess the importance of each of these elements.

The initial CASE adaption model is established in this section. The initial model contains two elements: a set of variables or factors, and a set of assumed dependencies between the factors. The statistical technique used in the analysis, discrete graphical modeling (presented in Section 5), allows the initial assumption that all selected factors are mutually dependent. Therefore this section results in the initial assumed model being saturated, i.e. each factor is assumed to be dependent on all the others.

The analysis done by Aaen et al. (1992) shows that, although there are differences between the responses from Danish and Finnish organization, these are not as numerous as the differences found between organizations with different levels
Table 2: Frequencies of: The number of people employed in the enterprise, the average size of projects in man-years, how long the tool has been used, and the total size in MB of all specifications produced by the tool (Aaen et al. 1992).

of CASE experience across the two countries.\(^1\) Hence, this analysis is performed under the assumption that variation in CASE experience is more significant than variation due to cultural differences. Consequently both Danish and Finnish organizations are analyzed as a whole.

In order to get a profile of the CASE experiences in the organizations, they were, among other things, asked about how long they had used the tool and how many mega-bytes of specifications they had made using the tool. Questions on the size of the enterprise and the estimated average size of projects were also included (Aaen et al. 1992). Table 2 shows the frequencies of these variables in order to provide a profile of the responding organizations’ demographical background.

CASE diffusion is dependent on a multitude of factors (Norman et al. 1989, Orlikowski 1989, Aaen & Sørensen 1991) and almost all of the variables in the survey could be of relevance. In order, however, to practically be able to perform the analysis, choices have to be made. Aaen et al. (1992) document the survey and present in an appendix the questionnaire used.

Of all the variables in the data set, a subset is chosen as eligible candidates for dependent and independent factors related to activities affecting the process of utilizing the CASE tool once it is purchased. The considered factors fall within the following four categories: Extent of CASE implementation as the dependent variable, and as independent variables: (1) CASE introduction activities, (2) tool modification, and (3) demographics. The following subsections present and discuss the operationalizations of dependent and independent variables considered.

### 4.1 Dependent Variable: Extent of CASE Implementation

Based on a survey of innovation research in general, Tornatzky & Klein (1982) concludes that almost all studies examined only adoption of the innovation, not adoption and implementation. They further state that: "The failure to use degree-of-implementation as a dependent variable probably yields misleading correlation of innovation characteristics with innovation behavior. Wynekoop et al.
(1992) argues that the extent of implementation is a more sensitive measure of adoption success compared to just measuring the CASE adoption process.

As the dependent variable, describing the extent of CASE implementation, two possible candidates are considered: (1) The percentage of developers in the organization using the CASE tool on a regular basis or (2) the respondents judgment of how far the organization are in the CASE introduction process measured on a seven point Likert-scale from *initial implementation* to *completely routine*. The last four stages in Cooper & Zmud’s model describe the extent of organizational implementation, therefore the percentage of developers using the tool is chosen as the dependent variable.

### 4.2 Independent Variables: Introduction, Modification & Demographics

**CASE Introduction Activities**

This independent variable covers organizational activities aiming at: installing and maintaining the CASE tool, changing work procedure, and introducing the tool to the developers.

Five variables containing data about how the CASE tool was introduced to the developers are summarized into one, i.e. a sum of whether or not the following types of introduction activities were conducted: tutorial/instruction, internal training/course, supplier course, self study, and other. In the same fashion, five variables are summarized on whether or not different types of user support for the tool were provided: maintenance contract; central help-desk, hot-line service by supplier, organized user group, and other. The method of summarizing five variables into one only counts the number of activities applied. Missing values and indication on not having applied a particular measure both results in 0.

CASE resulting in changes in work procedures has been promoted by several (Orlikowski 1989, Parkinson 1990, Smolander *et al.* 1991, Aaen & Sørensen 1991). Hence, a variable is included reflecting whether or not the introduction of the CASE tool led to new work procedures. In a subsequent question, not included as a factor here, the following types of changes in work procedures were indicated: access/authorization, standard reporting (mile stones), drawing conventions, naming conventions, and other.

Since the utilization of a CASE tool implies an extensive use of the structured methodology supported by the tool a variable is included measuring whether or not the introduction of the CASE tool lead to appointment of a methodology expert. The methodology expert could also be functioning as a change-agent in the diffusion process (Aaen & Sørensen 1991).

The introduction of CASE tools can lead to changes in the management of development projects as well as other organizational changes due to, for example, the formation of a ‘CASE task force’ (Parkinson 1990, Siltanen 1990, Aaen & Sørensen 1991). Two variables contain data on whether the introduction of the
tool were combined with management changes, or other organizational changes. In the first question 4 respondents indicated that the introduction of CASE led to management changes and 93 that it did not. In the second question 10 reported other organizational changes and 87 indicated no organizational changes. This indicates either that the introduction of CASE did not lead to major management or other organizational changes or that the two questions might have been too generally formulated and hence not interpreted correct. Although being highly relevant, the two variables are omitted from the model because of the very few indications on management or other organizational changes.

Modification of Tool

Since CASE is an information technology much like standard application packages (Aaen & Sørensen 1991) and since the utilization of CASE is contingent of a match between the tool and the methodology applied (Smolander et al. 1990), modification of the functionality is an important issue.

The issue of whether the functionality of the CASE tool adequately supports existing work procedures and systems development standards has been forwarded by, for example, Smolander et al. (1991) and Aaen & Sørensen (1991). Hence, the inclusion of a variable describing whether or not the functionality of the CASE tool has been changed regarding, for example: representation layout, consistency rules, report layouts, and other.

Organizational Demographics

In order to control for variations between organizations due to, for example, different size and different extent of CASE experience, demographic variables are included in the analysis.

In order to control whether or not the primary explanation for the extent of CASE diffusion among developers is the amount of CASE experience, two variables are considered: (1) How long the organization has used the CASE tool or (2) in how many projects the CASE tool has been used. In (Wynekoop et al. 1992) the length of CASE usage was chosen. Since 41% of the organizations have had the tool for more than 6 months, and still have only used it in 0-2 projects, the project variable is chosen. Kemerer (1992) argues that measuring the number of CASE projects instead of length of CASE experience more accurately reflects the learning process. This implies that only actual project experience is taken into account, and not experience from having 'installed' the tool on a shelf. The average size of projects in man-years is excluded, which could bias the result because of differences in project size.

DeLone (1981) examines how companies of different size manage their computer operations differently. Hence, CASE diffusion is assumed to have different conditions in organizations of different size. Two variables are considered: (1) The total number of people employed in the enterprise or (2) the total num-
ber of people employed in the IS department. Because the dependent variable is measuring the degree of CASE implementation as the percentage of developers using CASE on a regular basis, the number of people employed in the IS department is included in the analysis.

### 4.3 Model

Eight variables are, hence, included in the initial model. Extent of CASE implementation as the dependent variable and the following seven independent variables: (1) size of IS department, (2) number of CASE projects, (3) modification of tool functionality, (4) tool introduction activities, (5) new work procedures, (6) user support functions, and (7) appointment of methodology expert. The variables are presented in Table 3. Frequencies of the chosen variables are listed in Table 4, and Table 5.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Category</th>
<th>Answer Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dependent Variable</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extent of implementation</td>
<td>Ordinal</td>
<td>(0-25, 25-50, 50-75, &gt;75)% of developers</td>
</tr>
<tr>
<td><strong>Independent Variables</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Size of IS department</td>
<td>Ordinal</td>
<td>(0-25, 26-100, &gt;100) employees</td>
</tr>
<tr>
<td>Number of CASE projects</td>
<td>Ordinal</td>
<td>(0-2, 3-5, &gt;5) projects</td>
</tr>
<tr>
<td>Modification of tool</td>
<td>Binary</td>
<td>(Yes, No)</td>
</tr>
<tr>
<td>to developers</td>
<td>Ordinal</td>
<td>(1,2,3-4): Sum of Tutorial/Instruction, Internal training/course, Supplier course, Self study, and Other</td>
</tr>
<tr>
<td>New work procedures</td>
<td>Binary</td>
<td>(Yes, No)</td>
</tr>
<tr>
<td>User support for tool</td>
<td>Ordinal</td>
<td>(0,1,2,3-4): Sum of Maintenance contract, Central help-desk, Hot-line-service by supplier, Organized user group, and Other</td>
</tr>
<tr>
<td>Appointment of methodology</td>
<td>Nominal</td>
<td>(Yes, No, Not relevant)</td>
</tr>
<tr>
<td>expert</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Measures of dependent and independent variables

Despite of the logical distinction made between dependent and the independent variables, the initial CASE adaption model is based on the assumption that dependencies between all the eight variables exist, i.e. no a priori assumptions about the relative importance of variables are made. The model is represented as an undirected graph. In the graph, the vertices represent variables. If two variables are dependent this is represented by an edge between the two vertices (Section 5 presents the method applied in detail). The model consists of 8 factors and hence 28 assumed dependencies,³ as presented in Figure 1.
Table 4: Frequencies of: The percentage of developers using CASE, the number of developers in the IS department, the number of projects performed using the CASE tool, and whether or not the functionality of the CASE tool has been modified.

<table>
<thead>
<tr>
<th>Implementation</th>
<th>Size of IS</th>
<th>Projects</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-25%</td>
<td>59</td>
<td>0-25</td>
<td>42</td>
</tr>
<tr>
<td>25-50%</td>
<td>17</td>
<td>26-100</td>
<td>30</td>
</tr>
<tr>
<td>50-75%</td>
<td>12</td>
<td>&gt; 100</td>
<td>27</td>
</tr>
<tr>
<td>&gt; 75%</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>n=</td>
<td>96</td>
<td>99</td>
<td>99</td>
</tr>
</tbody>
</table>

Table 5: Frequencies of: Number of different means used in the introduction of the tool to developers, whether work procedures were changed as a result of the tool introduction, number of different types of user support, and whether the introduction of the CASE tool involved the appointment of a methodology expert.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>21</td>
<td>Yes 62</td>
<td>0 9</td>
</tr>
<tr>
<td>2</td>
<td>42</td>
<td>No 29</td>
<td>1 32</td>
</tr>
<tr>
<td>3-4</td>
<td>34</td>
<td>2 19</td>
<td>3-4 35</td>
</tr>
<tr>
<td>n=</td>
<td>97</td>
<td>91</td>
<td>95</td>
</tr>
</tbody>
</table>
5 Graphical Modeling

Since the variables chosen (see Table 3) are either nominal or ordinal, discrete statistical analysis methods is chosen instead of, for example, multiple regression analysis (Agresti 1984, Agresti 1990). The method chosen is discrete graphical model analysis where log-linear models for contingency tables are represented as undirected graphs (Darroch et al. 1980, Lauritzen 1989, Whittaker 1990). It is a conceptually simple, but very powerful technique to analyze and describe dependencies or conditional independencies between groups of variables (Kreiner 1989b). In the graph illustrating the model, the vertices represent variables. If two variables are dependent this is represented by an edge between the two vertices. Hence, the absence of an edge between two vertices implies that the corresponding variables are conditionally independent.

In this analysis the initial assumption is dependencies between all pairs of variables (Figure 1). The analysis then iteratively investigates the hypothesis of dependencies between variables, with the purpose of testing whether or not the 0-hypothesis can be rejected. A likelihood ratio test of conditionally independence between factors A and B implies testing whether or not information is discarded when collapsing the partitioning according to the two factors. If the
0-hypothesis of conditional independence can not be rejected the edge between the two variables is removed.

The modeling process of reducing the complete graph constitutes of several activities: (1) recursive automatic removal of edges, (2) manual adjustments to the model, (3) rank tests, (4) test of resulting model against the initial model, (5) approximate estimations of significance of non-decomposable dependencies, and (6) interpretation of results (Badsberg 1991, Badsberg 1992, Kreiner 1989a, Kreiner 1989b, Kreiner & Edwards 1989, Kreiner & Karpatchov 1989).

Graphical modeling is a heuristic technique which implies that a given initial model through different testing strategies may result in different models. If this occurs a selection between the models must be made according to the specific research problem investigated. (Kreiner 1989b).

Discrete graphical modeling is conducted using the CoCo^3 application (Badsberg 1991, Badsberg 1992).

Because the variables each have from 2 to 4 categories the analysis of any pair of the variables results in contingency tables with between 4 and 16 cells (Table 3). The only 102 observations implies a high probability of violating the criterion for performing likelihood ratio statistics using chi-square approximation. Chi-square approximation requires that at least 80% of the cells in the analyzed contingency tables have 5 or more observations (Agresti 1984). Because of the small sample, exact tests based on Monte Carlo approximation are used (Agresti 1990, Badsberg 1991). An exact test simulates the actual distribution by generating random tables with marginal values equal to the ones in the observed table (Agresti 1990, Badsberg 1991).

In the modeling process, a level of significance on p=0.05 is applied. Automatic removal of edges is done by backwards elimination, recursively removing the least significant edge when testing the model against its predecessor. Once an edge has been rejected, it is excluded in following tests. Tests are only performed on decomposable models, i.e. triangulated graphs which is graphs containing no cycles of lengths ≥ 4 without a cord (Lauritzen 1989). After the recursive elimination of least significant edges, the resulting model is scrutinized in order to check whether further edges can be removed or if some of the already rejected edges is significant. This is done by applying manual adjustments to the model performing forward inclusion and backward elimination of edges (Badsberg 1992).

The exact likelihood ratio statistics only uses the categories as discriminants in the computations. It does not support the detection of dependencies due to, e.g. ordering of the different levels (Lauritzen 1989). This implies that structures among ordinal valued variables might not be detected. In order to detect such dependencies rank tests between ordinal factors are applied. As suggested by Kreiner (1989a), a two-sided Goodman-Kruskal Gamma coefficient test is performed (Agresti 1990, Badsberg 1992).

If adding significant edges from the rank test results in a non-decomposable model, the graph is triangulated and the model is further manually adjusted. The purpose of testing the resulting model against the initial model is to decide
whether the parsimonious model can be rejected given the saturated model. If the exact \( p \)-value is less than, or equal to 5%, the parsimonious model is rejected. The reason for only analyzing decomposable models is that the final test can only be performed as an exact test if the model is decomposable (Badsberg 1992).

In order to obtain estimates of \( p \)-values for the non-decomposable edges, an approximated test is performed for each of these edges. The closest decomposable model without the edge is tested against the same model with the edge added. Interpretation of results implies analyzing the structure of the dependencies found.

6 Results

The initial graphical model of assumed dependencies between all variables (Figure 1) results in a decomposable parsimonious model. In an exact test of the parsimonious model against the initial model, the 0-hypothesis cannot be rejected \( (p=0.226) \). The approximation of \( p \)-values for non-decomposable edges shows a conditional independence in \( H_6 \), i.e. between the extent of CASE implementation and the user support for the tool \( (p=0.400) \). As a result of this, the edge is removed from the final decomposable model. The analysis process, hence, results in a model with 11 dependencies between pairs of factors, i.e. 17 of the initial 28 assumed dependencies turned out to be conditionally independent. The 28 hypotheses investigated are presented in Table 6 and 7. The hypotheses of dependencies between the dependent variable and the seven independent are presented first (H1–H7) and hypotheses concerning dependencies between independent variables are presented last. The supported hypotheses are indicated with a “•”.

The model is shown in Figure 2.\(^5\) Labels on the edges in Figure 2 refer to the dependencies found. Labels on factors refer to the conditional independencies found. For example, both factors Extent of implementation and Modification have the label \( H_3 \) which indicates that the two factors are conditionally independent and that the hypothesis can be found in Figure 6 or 7 as \( H_3 \).

Although different testing strategies can result in different models, the data analyzed seem to be relatively stable. In a preliminary analysis where the variable containing data on size of the IS department is excluded, the application of three different automatic search procedures result in the same model\(^6\) (Badsberg 1991, Badsberg 1992).

All dependencies found, except one, are generally direct relations, e.g. high degree of organizational implementation implies having used CASE in several projects. The only inversely related dependency found is between extent of implementation and size of the IS department (H1).

The following three sections explores the model with respect to: extent of CASE implementation, adaption activities, and modification of tool functionality.
DEPENDENT VARIABLE

The extent of CASE implementation is:

- $H_1$: inversely related to the size of the IS department. $\gamma = -0.49$, $p = 0.000$
- $H_2$: directly related to the number of projects performed using the CASE tool. $\gamma = 0.38$, $p = 0.041$
- $H_3$: conditionally independent of whether the functionality of the CASE tool has been modified. $p = 0.389$
- $H_4$: dependent on the number of different means used in the introduction of the tool to developers. $p = 0.003$
- $H_5$: dependent on whether work procedures has changed as a result of the introduction of the tool. Non-decomposable. $p \approx 0.003$
- $H_6$: conditionally independent of the number of different means of organizing user support in relation to daily use of the tool. Non-decomposable. $p \approx 0.40$
- $H_7$: conditionally independent of whether the introduction of the CASE tool involved the appointment of a methodology expert. $p = 0.226$

INDEPENDENT VARIABLES

The number of projects using the CASE tool are:

- $H_8$: dependent on whether work procedures has changed as a result of the introduction of the tool. $p = 0.009$
- $H_9$: conditionally independent of whether or not the functionality of the CASE tool has been modified. Non-decomposable
- $H_{10}$: conditionally independent of the number of different means used in the introduction of the tool to developers. $p = 0.400$
- $H_{11}$: conditionally independent of whether or not the introduction of the CASE tool involved the appointment of a methodology expert. Non-decomposable
- $H_{12}$: conditionally independent of the number of different means of organizing user support in relation to daily use of the tool. $p = 0.941$
- $H_{13}$: conditionally independent of the size of the IS department. Non-decomposable

Table 6: Hypotheses supported by graphical modeling - Part 1. Dependencies are marked with a "*"
INDEPENDENT VARIABLES (continued)

Work procedures being changed as a result of the introduction of the tool are:

- $H_{13}$ dependent on the number of different means of organizing user support in relation to daily use of the tool. Non-decomposable, $p=0.013$
- $H_{15}$ dependent on whether the introduction of the CASE tool involved the appointment of a methodology expert. $p=0.023$
- $H_{16}$ conditionally independent of the size of the IS department. $p=0.099$
- $H_{17}$ dependent on the number of different means used in the introduction of the tool to developers. $p=0.002$
- $H_{18}$ conditionally independent of whether the functionality of the CASE tool has been modified. $p=0.217$

The appointment of a methodology expert in relation to CASE introduction is:

- $H_{19}$ dependent on the number of different means of organizing user support in relation to daily use of the tool. $p=0.017$
- $H_{20}$ conditionally independent of the size of the IS department. Non-decomposable
- $H_{21}$ conditionally independent of the number of different means used in the introduction of the tool to developers. Non-decomposable
- $H_{22}$ conditionally independent of whether the functionality of the CASE tool has been modified. Non-decomposable

The number of different means of organizing user support is:

- $H_{23}$ directly related to the size of the IS department. $\gamma=0.39$, $p=0.006$
- $H_{24}$ conditionally independent of the number of different means used in the introduction of the tool to developers. $p=0.154$
- $H_{25}$ conditionally independent of whether the functionality of the CASE tool has been modified. Non-decomposable

The number of different means used in the introduction of the tool to developers is:

- $H_{26}$ dependent on whether the functionality of the CASE tool has been modified. $p=0.000$
- $H_{27}$ conditionally independent of the size of the IS department. Non-decomposable

The modification of CASE tool functionality is:

- $H_{28}$ conditionally independent of the size of the IS department. Non-decomposable

Table 7: Hypotheses supported by graphical modeling. Part 2: Dependencies are marked with a “●”

40
6.1 Factors Affecting the Extent of CASE Implementation

In this section the model is interpreted from the perspective of the dependent variable *extent of CASE implementation*.

In 61% of the organizations less than a quarter of the developers use CASE tools on a regular basis. An interesting question then is: What characterizes the organizations where a larger percentage of the developers use CASE?

The size of the IS department is clearly important for the extent of the implementation ($H_1: \gamma = -0.49$), which seems natural since small departments more easily can diffuse the tool among a relatively large percentage of the developers.

The size of the IS department does, however, not account for all the variation. Even given the size of the IS department, the extent of implementation depends on: the number of CASE projects, changes in work procedures, and the introduction strategy. The extent of CASE implementation is conditionally independent of: types of user support for the tool, whether a methodology expert has been appointed, and whether the functionality of the tool has been modified.

The major difference in extent of implementation, given the number of different types of introduction ($H_4$), is between those having applied only one and those having applied more than one introduction activity. 90% of those having introduced the tool using only one activity had only diffused the tool amongst less than a quarter of the developers. If more than one activity has been applied approximately a half of the respondents indicate having diffused CASE among more than 25% of the developers. A closer analysis of the data shows that rela-
tively more have used self study and internal training compared to using supplier courses and tutorials. This could imply that although there is a commitment to facilitate the tool introduction the more costly means are generally avoided.

The organizational implementation of the tool is dependent on the number of CASE projects (H₂). In general, the higher a percentage of developers that use the tool the more CASE projects have been performed. The only deviation from this is for organizations having diffused the tool among 25–50% of the developers where 19% have used the tool in 0–2 projects and 31% have used it in more than 5 projects. None of these organizations have used CASE in 3–5 projects. A possible explanation for this anomaly could be the relatively small sample.

The extent of CASE implementation are also dependent on whether the tool introduction led to changes in work procedures (H₃). 52% of the organizations having introduced new work procedures had diffused CASE among more than a quarter of the developers. This could indicate that the organizations adapting CASE without changing work procedures get stuck in the implementation process, as suggested by Humphrey (1989). Mismatches between the existing development process and the requirements of the tool leads to the tool only being scarcely utilized. Since a dependency is found between the extent of implementation and the number of CASE projects performed, the dependency between extent of implementation and changes in work procedures suggests that work procedures are not necessarily changed when organizations have performed a number of projects.

Using a CASE tool implies the possibility of better coordination of development activities because of the common repository (Curtis et al. 1988). If the introduction of CASE has not led to changes in work procedures this possibility has not been utilized.

If the allocation of resources for organizational adaption measures are interpreted as management commitment the results could indicate a relationship between degree of implementation and management commitment, as indicated by Wynekoop et al. (1992). The introduction of the tool and changes in work procedures seem to affect the diffusion rate. These two factors are perhaps, compared to appointment of a methodology expert and user support services, more resource demanding measures. This suggests that the resource demanding steps are not taken by the less committed organizations.

6.2 Dependencies Between Independent Variables

The previous section focused on the variables in the model directly explaining the degree of organizational diffusion of CASE. This section presents dependencies found between independent variables.
CASE Introduction Activities

A total of 68% indicate that introduction of the CASE tool led to changes in work procedures. Most of the organizations, 78%, have applied two or more introduction measures and 57% have two or more types of user support for the tool. In only a quarter of the organizations did the introduction of CASE lead to the appointment of a methodology expert. 27% deemed it not relevant.

Whether the introduction of CASE has led to changes in work procedures seems to play an important role. In the model, changes in work procedures is only conditionally independent of: size of the IS department and whether the functionality has been modified (H_{16}, H_{18}). This result is not surprising since changes in work procedures can be assumed to reflect whether the introduction of CASE has led to changes in the use of methodology. Zmud (1983, 1984) points out that the introduction of modern software practices can be viewed as an organizational process innovation and that such process innovations seem to be scarcely adopted. According to, for example, Yourdon (1986) and Humphrey (1989) most of the North American software organizations do not manage and perform development of software in a structured fashion. Hence, the indication of no changes in work procedures most probably indicate that the CASE tool is not utilized in an effective manner.

The appointment of a methodology expert increases the likelihood that new working procedures follows the introduction of CASE (H_{15}). Among the organizations that had appointed a methodology expert 81% indicated changes in work procedures. Because there are no indications in the data as to whether the organizations did not find the appointment relevant or whether they already had a methodology expert prior to the introduction of CASE it is problematic to interpret the consequences of those answering that they did not appoint a methodology expert and those answering that such an appointment was not relevant.

A dependency is found between having applied several introduction measures and the introduction of the tool leading to new work procedures (H_{17}). Among the organizations having applied one measure there is a 50–50 split between the CASE tool having and not having resulted in new work procedures. 68% of the organizations having applied two introduction measures report changes in work procedures. When three or four introduction measures have been applied 81% report changes in work procedures.

In a similar fashion, the more types of user support applied, the larger is the likelihood that the introduction of CASE lead to new work procedures (H_{14}). Among those having applied 2 and 3–4 types of user support 78% and 79%, respectively, indicated that new work procedures had occurred. Only 43% of the organizations indicating no user support had introduced new work procedures. This indicate that investing in a variety of tool and methodology maintenance measures can be an important step in facilitating the proper usage of the tool. The application of several means of both tool introduction and user support can
be interpreted as a measure of the organizational commitment to make organizational investments in CASE. Huff (1992), amongst others, puts emphasis on costs regarding skill development training and user support, and estimates these costs to approximately twice the value of the CASE software.

The introduction of new work procedures were also dependent on the number of CASE projects \( (H_3) \). Among the organizations having introduced new work procedures 60\% had used CASE in three or more projects. Among the organizations that did not introduce new work procedures only 34\% had done three or more CASE projects. If project managers and developers are to realize the possibilities a CASE tool offers, changes in work procedures most likely will have to take place (Humphrey 1989). Hence, doing several projects using the tool most likely will result in changes in work procedures. At the same time, the tool can be a leverage for changing the current development practice since the proper utilization of the tool requires a certain degree of strictness in the development process. This implies that changes in work procedures can be interpreted as both cause and effect.

The appointment of a methodology expert were most likely among organizations applying 3–4 types of user support \( (H_{19}) \). A total of 64\% of the organizations that appointed a methodology expert had applied 3–4 types of user support.

Due to the rank tests, a direct relationship is found between the number of user support measures and the size of the IS department \( (H_{23}) \). A larger IS department implies more different kinds of user support. This could be explained by small IS departments relying more on self-appointed product-champions, whereas larger IS departments often is characterized by more resources for this kind of task and a larger degree of specialization.

**Modification of the CASE Tool**

Only 30\% of the organizations have modified the functionality of the tool. Modification of the tool is only dependent on the number of introduction measures \( (H_{26}) \). The organizations that have not modified the functionality are almost equally distributed with respect to the number of introduction activities they have applied, i.e. 25\% have applied one, 36\% have applied two, and 39\% have applied three or more. The organizations that have changed the CASE tools have most often conducted two introduction activities, i.e. only 14\% have applied one, 59\% have applied two, and 27\% have applied three or more.

The fact that modification most likely is performed in organizations having applied two types of introduction measures could indicate that these organizations try to modify the tool to the existing way of developing systems and not committing to adapt the organization to using the tool. The modifications performed are, however, not fundamental changes in, for example, the methodological support. Tool modification can be interpreted as an additional introduction activity. As the model (Figure 2) shows, the extent of CASE implementation is conditionally independent of whether or not the tool was modified, i.e. given that
we know the types of introduction activities applied, we do not obtain further information about the extent of implementation by knowing whether or not the functionality of the CASE tool was modified.

The modification of tool functionality is only found to be dependent on the introduction of the tool, and conditionally independent of, for example, the number of CASE projects. This could suggest that the maturity in the use of the tools has not yet reached a level where modification is an important issue.

7 Conclusion

This paper establishes, through statistical analysis of survey data in Denmark and Finland, a discrete graphical model describing the relationship between the extent of CASE implementation, characteristics of the implementation process, and demographic variables. An initial saturated model containing dependencies between all of the following eight variables was analyzed: (1) Extent of CASE implementation, (2) size of Information Systems (IS) department, (3) number of CASE projects, (4) modification of tool functionality, (5) tool introduction activities, (6) new work procedures, (7) user support functions, and (8) appointment of methodology expert. The initial model assumed 28 dependencies. The graphical analysis reduced this to a parsimonious model containing 11 dependencies and 17 conditional independencies.

The extent of CASE implementation is found to be inversely related to the size of the IS department and is dependent on: the number of CASE projects, changes in work procedures, and the CASE introduction activities. Besides being dependent on the extent of CASE implementation, changes in work procedures are found to be dependent on: the number of CASE introduction activities, the number of CASE projects, the number of user support activities applied, and the appointment of a methodology expert. The user support applied is directly related to the size of the IS department, and dependent on the appointment of a methodology expert.

The analysis supports the hypothesis that the extent of CASE implementation is dependent on management commitment. Proper utilization of a CASE tool implies changes in development practices, at the same time as the tool can boost the use of structured methodologies.

The modification of the functionality of the CASE tool is only found to be dependent on the introduction of the tool. This suggests that modification of tool functionality was not an important issue in the organizations surveyed. Tool modification can be viewed as an element in the CASE introduction strategy, similar to changing work procedures to the CASE tool. It is, however, important to note, that the model does not contain any variables expressing the compatibility (Tornatzky & Klein 1982) of the CASE tool compared to current work practice. Further improvements of the model could, amongst others, incorporate a variable describing whether CASE was a radical, incremental or compatible technology.
(Wynekoop 1992b). The existing data could be used as a starting point, because they contain variables on the system development techniques used before and after CASE. The type of introduction strategy applied could also be included in the model, since the strategy can affect the perception of the compatibility of the technology (Wynekoop 1992b). The data set does, however, not contain variables on the type of strategy applied.

The analysis performed in this paper can serve as a starting point for further investigations. The survey data have been used to build a model of dependencies between the selected variables. Subsequent studies could aim at testing the model using other data samples. In order to probe deeper into the hypotheses established by in the model, further qualitative analysis of a small number of organizations could be performed in order to explain the dependencies and conditional independencies found. The model could serve as a basis for selecting the organizations in order to ensure diversity in the sample.

The graphical modeling technique proved to be an intuitively very suitable method for analyzing data and reducing the complexity of the saturated model.

Multiple regression analysis supports the test of models where variation in a dependent variable is explained by variation of a set of independent variables. Tornatzky & Klein (1982) point at the need for innovation diffusion studies to analyze dependencies between independent variables. Discrete graphical analysis, as it is used in this paper, is an experimental technique that supports a process of discovering dependencies and conditional independencies between both types of variables. The technique itself does not imply any distinction between dependent and independent variables, although applications of the technique can use the distinction. It is, however, still important to be thorough in the selection of variables, and to utilize knowledge about the domain investigated when performing graphical analysis.

Acknowledgments

Jens Henrik Badsberg convinced me that graphical modeling using CoCo was the most suitable technique and tool for the analysis I was planning to do. He has, furthermore, put a substantial amount of work and pride into helping when the theories and practicalities of the statistical method stopped me. His contributions are gratefully acknowledged. Also, a warm thanks to three anonymous reviewers and to Kristin Braa, Lars Mathiassen, Peter A. Nielsen, Erik Stolterman, John Venable, and Ivan Aaen who all provided very important comments. The data analyzed in this paper stems from the CASE Monitor Project, partly funded by the Technology Development Center of Finland and by the Danish Natural Science Research Council, Grant No. 11-8394. In the CASE Monitor Project I worked together with Ivan Aaen, Aalborg University, Denmark, and Aila Siltanen & Veli-Pekka Tahvanainen, University of Jyväskylä, Finland. All errors in this paper naturally remain the responsibility of the author.
Notes

1. Exact likelihood ratio tests show no differences between Denmark and Finland with respect to: Enterprise size ($p=1.000$); size of IS department ($p=1.000$); number of developers in project teams ($p=0.400$); size of projects ($p=0.400$); years of CASE experience ($p=0.195$); total size of models produced ($p=0.850$); evaluation of advancement in implementation process ($p=0.400$); and extent of implementation among developers ($p=0.340$).

2. Regarding the method used for summarizing five variables into one, it would perhaps give a better result if factor analysis was used to collapse the introduction- and user support variables, instead of just adding them, because of possible dependencies between the summarized variables (Rummel 1970).

3. The number of edges in a saturated model is calculated as: \[ \frac{n(n-1)}{2} \], with $n$ being the number of factors.

4. CoCo runs under Unix and DOS and can be obtained free of charge by anonymous ftp over internet from iess.auc.dk.

5. In order to avoid crossing edges and for aesthetic reasons the graph has been unfolded. This, unfortunately results in a slightly less readable figure because of the dependent variable being placed differently in the figure compared to the initial model.

6. The three strategies are: (1) recursive elimination of edges, i.e. same strategy as described in the previous section, (2) recursive elimination of edges with all edges from the extent of implementation factor fixed, i.e. dependencies between the dependent variable and the independent variables are only tested when performing manual adjustments to the model, and (3) non-deterministic recursive removal of edges, i.e. headlong backward search where in each iteration step a non-significant edge is randomly chosen for removal, instead of in each step removing the most non-significant edge.

References


