The Use Of CASE Tools

Diane Lending
University of Michigan-Dearborn
School of Management
4901 Evergreen Road
(313) 593-5008
Dlending@fob-f1.umd.umich.edu

Norman L. Chervany
University of Minnesota
Information and Decision Sciences, 271 19 th Avenue South
Minneapolis, MN 55455
(612) 624-1520
Nchervany@csom.umn.edu

1. ABSTRACT

Computer-Aided Software Engineering (CASE) technologies are tools that provide automated assistance for software development [3]. The goal of introducing CASE tools is the reduction of the time and cost of software development and the enhancement of the quality of the systems developed [3], [20]. This paper explores the use of CASE tools. We ask several questions. Are CASE tools being used? If yes, what features within the tool are being used? Next, we explore two potential reasons for the expected low use. Do CASE tools change the job of the systems developer in an unattractive way? And are the people who are expected to use CASE tools motivated to use them? 233 systems developers were surveyed to answer these questions. We found that CASE tools are being used but not in many companies. Within the companies that have adopted CASE tools, few people are actually using the tools. The systems developers who use CASE tools are using formal methodologies more often than systems developers who do not use CASE tools. Systems developers allocate their time differently depending on whether they are using a CASE tool or not. Those who use the tools are using few of the functions within the tools. Finally we found that people were basically neutral on whether they enjoyed using the tool and whether the tool was useful.

1.1 Keywords

CASE tools, information systems development, Technology Acceptance Model (TAM)

2. INTRODUCTION

Computer-Aided Software Engineering (CASE) technologies are tools that provide automated assistance for software development [18]. The goal of introducing CASE tools is the reduction of the time and cost of software development and the enhancement of the quality of the systems developed [3,20]. Prior research into CASE tool use has suggested that (1) few organizations use CASE tools (e.g., [8], [17]); (2) organizations abandon the use of the tools (e.g., [21], [24], [8]); and (3) organizations that do use CASE tools contain many systems developers who do not actually use the tool [14].

This paper adds to what we already know about the use of CASE tools. We first see if we get similar results to preceding studies on the low use of CASE tools. We then dissect this broader question to look at CASE usage in more depth. First in the companies that are using CASE tools, what features of the tools are being used? Second, does the use of CASE tools in some way change the job of the systems developers? Thirdly we look at the antecedents of intentions to use a computer tool, and explore how motivated systems developers are to actually use CASE tools. This study will add depth to what we already know about the usage of CASE tools.

3. DEVELOPMENT OF RESEARCH QUESTIONS

Four research questions are asked in this study. This section discusses the development of these questions based upon prior research and theory.

CPR 98 Boston MA USA
Copyright 1998 0-89791-959-9/98/3..$5.00

3.1 Research Question 1

Are CASE Tools being used?

Many prior studies have reported limited use of CASE tools. In a survey of 53 companies, [8] found that 39 (73.5%) had never used CASE. Of the14 companies who had tried CASE, five had subsequently abandoned use of the tools. People within these fourteen companies believed that use of CASE tools improved documentation quality, improved analysis, and resulted in systems that were easier to test and maintain. However, they also found use of CASE tools difficult and time consuming. [17] in another cross organization survey, found that only 24% of companies were using CASE tools. In a follow-up survey of thirteen managers who had been using CASE tools two years earlier, [24] reported that continued CASE use could only be verified for four managers. The reasons for abandonment included cost, lack of measurable returns, and unrealistic expectations. [11] looked within organizations that used CASE tools and found that large numbers of their systems developers were not using CASE tools. He reported that in 57% of the organizations surveyed that were using CASE tools, less than 25% of the systems developers used the tools.

In the first research question, we are replicating these prior studies to see if similar results are found.

3.2 Research Question 2

What features of CASE tools are being used?

The term Computer-Aided Software Engineering (CASE) encompasses many different products with different functionalities. In the International Workshop on Computer-Aided Software Engineering (IWCASE) definition of CASE very broad terms are used: "...tools and methods to support an engineering approach to systems development at all stages of the process" [9]. When the term CASE is used, it is important to clarify what is being discussed. Most classifications of CASE tools start by considering whether the tool is upper CASE, lower CASE, or integrated CASE [3]. An upper CASE tool (front end CASE) provides support for the early stages in the systems development life cycle such as requirements analysis and design. A lower CASE tool (back end CASE) provides support for the later stages in the life cycle such as code generation and testing. Integrated CASE tools support both the early and later stages. Further classifications (e.g., [19]) usually list which functionalities are supported by the tool, such as data flow diagrams, entity relationships data models, etc. [10] provides a different type of model of CASE functionality which helps organize CASE tools. This model includes three functional dimensions of CASE tools: production technology, coordination technology, and organizational technology.

Production technology is the functionality that directly affects the capability of an individual to generate planning of design decisions and subsequent artifacts or products. An example of production technology is support for drawing a data flow diagram. [10] further divides production technology into representation functionality, analysis functionality, and transformation functionality. Representation functionality is the functionality that allows a system developer to define or describe an object, relationship, or process such as a data flow diagram. Analysis functionality is the functionality that allows a systems developer to experiment with alternate representations, models, or relationships, (e.g., testing consistency between a process model and a data model.) Transformation functionality is the functionality that executes a significant planning or design task, replacing or substituting for a human analyst, (e.g., generating executable code.) CASE tools may have any combination of these functionalities. Representation functionality is always present; the other two may or may not be present. This classification of functionalities has been used by researchers such as [2].

What is important about the classification in [10] is that it goes beyond the usual specification that a function is present in the tool and identifies operationally what CASE tools add to the function. For example, data flow diagrams can be done manually or with a CASE tool. If the CASE tool provides automated support for construction of the diagram, the tool provides representation functionality. If the tool checks the data flow diagrams for consistency with an entity/relationship model and definitions, the tool provides analysis functionality. If the tool transforms a high-level data flow diagram into more detailed diagrams, it provides transformation functionality.

The model developed in [10] includes two functional dimensions beyond production technology: coordination technology and organizational technology. These dimensions support the group and organizational use of CASE tools. While conceptually possible, [10] found very limited existence of this technology. [25] reports similar lack of these functionalities. Thus, these dimensions are not examined in the present research.

With the second research question, we explore how many CASE tool features are actually being used.

3.3 Research Question 3

Do people who use a CASE tool use the same methodologies that systems developers who do not use a CASE tool use?

Do people who use a CASE tool perform the same activities that systems developers who do not use a CASE tool perform?

There has been little research on the effect of CASE technology on the jobs of CASE users. The existing literature mostly concentrates on user acceptance or resistance. The most relevant prior literature is summarized here.

Many of the prior studies focus on the adoption process; any examination of the system development job is peripheral at best. [20] looked at the introduction of CASE tools into a single organization and focused on the social relationships of the systems developers. She found that CASE tools had significant changes in the division of labor on systems development projects that led to shifts in power and dependency among the workers [20]. [26] looked at the implementation of CASE tools from an innovation diffusion perspective. They found that those users who perceived the relative advantages of the CASE tool in a realistic manner (rather than unduly optimistically or pessimistically) had higher acceptance of the tools. [23] found that one major obstacle to adoption of CASE was the resistance of systems developers.

Other CASE research focuses on outcomes of work in terms of productivity. [19] looked at the opinions of systems developers about which software engineering tasks were improved (in terms of productivity) by the use of a specific CASE tool. They found that, in general, using CASE tools increased users' perceptions of their own productivity. Another study focusing on which aspect of CASE tools improves productivity was [3]. This case study found that by focusing on managing software re-use, substantial gains in productivity were documented.

[11] looked at both the adoption process and productivity. His article reflects the lower than expected use within companies that had adopted CASE tools. Since respondents perceived increases in productivity and system quality with CASE use, he concluded that the low use is a loss in productivity and quality for the company. He also found that voluntariness of CASE use within a company is associated with less CASE use. He suggests that managers may want to strengthen their commitment to CASE adoption in light of this finding.

Does the use of CASE tools change the way systems developers develop systems? In the past, the systems development life cycle (SDLC) methodology was the most common methodology for building systems [7]. The SDLC methodology is less well suited for the types of systems that are being built in the 1990s [12]. While many organizations still use the SDLC methodology, it is often supplemented with other methods. Systems developers were asked to indicate which methodologies they used in their job. The methodologies examined in this research were:

Life Cycle

Object-oriented Approach

Rapid Applications Development (RAD)

Prototyping

Joint Applications Development (JAD)

In a pilot study, an "Other" category for methodology was included. Because, no single methodology was added more than once or twice, this list was used in the full study.

This research also explored the systems development activities that made up the systems development job. The job of a systems developer may contain requirements analysis, process design, data design, and programming among other activities [7]. But, not all systems developers do the same activities. One may spend most of his or her time on analysis; another, on design. This study captured the particular activities that each individual does. Systems developers were asked to indicate which activities were part of their job. The activities examined in this research were:

Systems Analysis (including feasibility studies and requirements definition)

Systems Design (including user interface, data, and process design)

Programming (or generating code)

Testing

Supervisory or other management tasks

Maintenance.

In a pilot study, an "Other" category for activities was included. Because no single activity was added more than once or twice, this list was used in the full study.

In the third research question, we see how the job of the systems developer changes with use of a CASE tool.

3.4 Research Question 4

Do systems developers who use CASE tools enjoy them?

Do they perceive CASE tools as being useful?

The best prediction of a behavior is the person's attitude about that behavior [1]. This explanation of behavior (the Theory of Planned Behavior) is widely used in social psychology and in IS research (e.g., [15], [16].) [5] introduced the Technology Acceptance Model, an adaptation of the Theory of Planned Behavior. The TAM model, which includes perceived usefulness and ease of use, has been found to predict intentions to use technology very well [15]. In an extension to TAM, [6] showed that there was an intrinsic motivation to use computers (enjoyment) as well as an extrinsic motivation (perceived usefulness). Enjoyment is defined as the degree to which an individual enjoys using the tool in its own right that is without regards to consequence [6]. Perceived usefulness is defined as the extent to which using the tool is perceived to improve performance of the job [6]. In [6], these two different types of motivation explained up to 75% of variance in intentions to use the tool. Using this model, if we find that CASE tools are fun to use and are perceived as useful, then we will find a group that is motivated to use CASE tools and intend to use the tools. Conversely, if people do not enjoy the tools and do not think they are

useful, they will not be motivated to use the tools and will not intend to use them.

In the fourth research question, we explore these important motivators to explain intentions to use CASE tools.

4. METHODOLOGY

The research involved a cross-sectional field study of several organizations. Organizations were selected based upon two criteria: 1) the company needed to have systems development projects underway and 2) the company needed to be willing to cooperate with the researchers. Several companies in a large midwestern metropolitan area were asked to participate in the research. The attempt was made to include organizations from different industries to increase external validity. Six companies and a public institution agreed to participate. Table 1 summarizes the sites.

Each company that did not use CASE was asked if all systems developers could be surveyed. Each company that did use CASE was asked if all systems developers using CASE could be surveyed. The companies either provided the researcher with a list of names of systems developers or handled distribution within the company themselves. Two questionnaires were used for data collection: one captured information about the systems development job and one captured information about CASE tool use. In Table 1, job response rate is the percentage of individuals that received the survey about systems development and returned it. The CASE response rate is the percentage of individuals who were sent the CASE survey and returned it. No CASE surveys were sent to companies that did not use CASE Tools.

The data collection method was a written questionnaire. The steps suggested by [22] were followed to ensure a reliable and valid instrument. The initial questionnaires were developed based upon theory and prior research. The instruments were pre-tested to address content validity using experts in the field. A pilot test was conducted to measure construct validity and reliability. Multitrait-multimethod (MTMM) and factor analysis techniques were used to assess construct validity. Reliability was measured using Cronbach's alpha.

To increase the statistical power, we investigated whether the data from the pilot study could be included in the full study. None of the data items used in this project were changed between the pilot and the full study. All procedures were identical to those used in the main study. There was no overlap between populations. The only differences found between the pilot and full study were related to differences in organizations. Based upon this analysis, the pilot data were included in the analysis.

Company	Company Type	Pilot/ Full	CASE Tool	n	Nonusers of CASE	Users of CASE	Job Response Rate	CASE Response Rate
A	Public Institution	Pilot	None	33	33	0	.78%[1]	NA
B	Retail	Pilot	IEF	35	13	22	55%	49%
C	Manufacturing	Full	Excelerator	14	6	8	78%	72%
D	Utility	Full	None	29	29	0	64%	NA
E	Financial	Full	ADW	25	10	15	35%	56%
F	Retail	Full	None	84	84	0	85%	NA
G	Financial	Full	ADW	13	6	7	72%	61%
Totals				233	181	52		

Table 1: Organizations participating in the study

[1] This organization handled the distribution process and did not provide an exact response rate. The organization estimates imply a response rate of approximately 78%.

5. MEASUREMENT OF CONSTRUCTS

5.1 Representation, Analysis or Transformation Functionalities Used

The functionality questions were developed based upon [10]'s theory and tested in interviews with CASE experts. Because, representation functionality is always present with CASE tools, questions assessing its usage were not included. Use of analysis functionality was measured as the sum of the respondent's relative use of various features that [10] had defined as being analysis functions. Transformation functionality was measured in a similar manner. See Tables 4 and 5 for questions asked.

5.2 Enjoyment and Perceived Usefulness

Measures for Enjoyment and Perceived Usefulness (see Table 2) were taken from the already tested instrument developed in [6].

The Cronbach's alpha calculated for the enjoyment questions was .950. The Cronbach's alpha calculated for the perceived usefulness questions was .965.

Data were analyzed using regression techniques. The SAS statistical package was used for analysis.

6. DESCRIPTIVE STATISTICS

This section describes the sample. 233 people returned questionnaires. For demographic data characteristics, the sample size is slightly less than that as some people did not complete these fields. Table 1 summarizes the sample by company and CASE use.

Table 3 shows demographic statistics for the data. It was not assumed that the two populations would be identical demographically; however, the means and distributions were compared. For age, years in organization, and years in systems development, the means are statistically equal. However, the distribution of gender is not statistically identical. Higher percentages of males use CASE tools than do females. Additionally, the distribution of education is not statistically identical. CASE users were more highly educated.

7. RESULTS

7.1 Research Question 1

Are CASE Tools being used?

Our study found the same low use of CASE tools as reported in earlier studies. It was difficult to find companies using CASE tools. Even in the companies that use CASE tools, the amount of use is very low. Often managers have no idea how few people are using the CASE tools. For example, in one company that was not included in the study, a vice-president estimated that about 60 people in the systems development group were using the tool. However, a lower level manager much closer to the tool said that the real number was less than ten. This same pattern occurred in all the companies surveyed. In the four companies that use CASE tools (B, C, E, and G), surveys were sent to systems developers whom the company had identified as using CASE tools. Of those that responded to the survey, slightly more than 40 percent (35 out of 87) said that they did not use a CASE tool. (That percentage may be inflated since to indicate that you did not use a CASE tool, you simply had to fill in one field on the survey and return it. This could have resulted in higher response rates for non-CASE users.)

Enjoyment	
1.	I find using this CASE tool to be enjoyable.
2.	I have fun using this CASE tool.
3.	The actual process of using this CASE tool is pleasant.
Perceived Usefulness	
1.	Using this CASE tool in my job enables me to accomplish tasks more quickly.
2.	Using this CASE tool improves my job performance.
3.	Using this CASE tool in my job increases my productivity.
4.	Using this CASE tool enhances my effectiveness on the job.
5.	Using this CASE tool makes it easier to do my job.
6.	I find this CASE tool useful in my job.

Table 2: Measurement of Enjoyment and Perceived Usefulness

Variable	Total Population[2]	Non Users of CASE	Users of CASE	Minimum	Maximum
Age	37.49 (8.21)	37.41 (8.50)	37.82 (6.96)	23	60
Years with Organization	5.58 (6.16)	5.35 (6.25)	6.53 (5.76)	0	31
Years in Systems Development	9.93 (6.52)	10.01 (6.77)	9.56 (5.45)	0	30

Variable	Total Population[3],[4]	Non Users of CASE	Users of CASE
Gender			
Male	142 (68.9%)	109 (65.7%)	33 (82.5%)
Female	64 (31.1%)	57 (34.3%)	7 (17.5%)
Education			
High School	12 (5.8%)	12 (7.2%)	0 (0%)
Associate	38 (18.4%)	34 (20.5%)	4 (10%)
Bachelors	131 (63.6%)	105 (63.3%)	26 (65%)
Masters	24 (11.7%)	14 (8.4%)	10 (25%)
Doctorate	1 (.5%)	1 (.6%)	0 (0%)

Table 3: Demographic Statistics

7.2 Research Question 2:

What features of CASE tools are being used?

This research looked at two types of features: those providing analysis functionality and those providing transformation functionality. Use of these features was low.

Very few respondents use the analysis functionality available in their tool. Analysis functionality was measured on a 1 to 5 scale, with 1 being "Never Used," 3 being "Sometimes Used," and 5 being "Always Used." The overall mean value for use of this functionality was 1.75. Less than 13% of the 48 CASE users who filled in this item (6 users total) said that they used Analysis functionality sometimes or more often (a level of 3 or more). Table 4 contains the details on the use of Analysis Functionality.

Table 4 contains each of the functionalities that was identified in [10] as allowing a systems developer to experiment with alternate representations, models or relationships. As can be seen in the table, only one of these functionalities had a mean use close to 3 or "sometimes used" and that was detecting inconsistencies. None of the items were indicated as on average being used "sometimes" or more often than that (a level of 3 or more.)

[2] Counts and percentages.

[3] Means and standard deviations.

[4] Numbers do not sum to population totals because of missing data. Twenty-seven respondents did not provide demographic statistics either because they did not return the systems development survey form or because they left the demographic page blank. Another 4 people left age blank.

Specific Functionality	Mean[5]	Standard Deviation
Tested for consistency between a process model and a data model?	1.98	1.60
Checked for the structural equivalence of objects or processes?	1.59	1.43
Checked for unnecessary or redundant model connections?	2.19	1.64
Detected inconsistencies in models, definitions, etc?	2.92	1.57
Identified the design impact of proposed changes in a design?	2.16	1.56
Searched the design for similar objects?	2.31	1.55
Used analytical decision aids to measure performance?	1.12	1.10
Detected and analyzed system errors from execution of a target system?	1.08	1.10
Searched design for complex relationships?	1.53	1.49
Suggested problem resolutions based on previously used solutions?	1.79	1.52
Estimated the process/performance characteristics of a design?	1.45	1.33
Searched design for objects with specified characteristics?	1.76	1.32
Simulated the production environment of the target system?	1.37	1.54
Identified where predefined criteria or rules have been violated?	1.53	1.36
Traced relationships between detailed specs and planning efforts?	1.26	1.19
Identified the differences between separate versions of an object?	1.96	1.63
Overall Mean	1.75	

Table 4: Summary of Analysis Functionality

Similarly, very few respondents take advantage of the transformation functionality available in their tool. Transformation functionality was measured on a 1 to 5 scale, with 1 being "Never Used," 3 being "Sometimes Used," and 5 being "Always Used." The mean value for use of this functionality was 1.76. Less than 10% of the 48 CASE users (4 users total) said that they used transformation functionality sometimes or more often than that (a level of 3 or more.) Table 5 contains details on the use of Transformation Functionality.

Specific Functionality	Mean[5]	Standard Deviation
Generated executable code from a screen mockup?	1.33	1.36
Generated executable code in several languages?	1.45	1.43
Generated code compatible with a variety of physical environments?	1.59	1.54
Generated standard code for generic programs?	1.22	1.31
Generated executable versions of a design for testing/evaluation?	1.67	1.64
Converted a logical specification into a physical one?	2.14	1.68
Transformed a high-level representation into a more detailed one?	2.28	1.68
Provided documentation as a by-product of design?	2.51	1.62
Performed reverse engineering?	1.22	1.05
Generated screen mockups?	1.80	1.66
Imported data from or exported data to external files or packages?	2.20	1.70
Overall Mean	1.76	

Table 5: Summary of Transformation Functionality

[5] 1= Never Used, 3 = Sometimes Used, 5 = Always Used

Table 5 contains each of the functionalities that were identified in [10] as executing a significant planning or design task. As can be seen in the table, only one of these functionalities had a mean use over 2.5 and that was providing documentation as a by-product of design. None of the items were indicated as on average being used "sometimes" or more often than that (a level of 3 or more).

7.3 Research Question 3

Do people who use a CASE tool use the same methodologies that systems developers who do not use a CASE tool use?

Do people who use a CASE tool perform the same activities that systems developers who do not use a CASE tool perform?

Table 6 summarizes the systems development methodologies used by the sample of systems developers. Higher percentages of the CASE users used all types of methodologies. The Life Cycle Methodology was the most popular methodology for all systems developers but more than 75% of those who used CASE tools used the life cycle as compared to 56% of non-CASE users. CASE and non-CASE users used object-oriented methodologies approximately as often. The other methodologies (Rapid Applications Development, Prototyping, and Joint Applications Development) were used by approximately twice as high a percentage of CASE users as non-CASE users.

Table 7 describes the percentage of time that the respondent indicated that they spent on each systems development activity. The most striking difference is that users of CASE tools indicated that they spent more than twice as much of their time doing systems analysis than did non-CASE users. They also spent considerably less time doing programming, testing, and maintenance than non-CASE users.

7.4 Research Question 4

Do systems developers who use CASE tools enjoy them?

Do they perceive CASE tools as being useful?

Systems developers who used CASE tools were relatively neutral about whether they were enjoyable or useful. Table 8 summarizes the results from the survey. The items were measured with the items in Table 2. The variables were measured on a scale of 1 to 7 where 4 was Neutral and 5 was Agree Slightly. The results of the items for each construct were averaged. As can be seen, the mean for both variables was between 4 and 5. This means that CASE users when asked their opinion of whether they agreed that using their CASE tool was enjoyable, on average they were between neutral and agree slightly. And perhaps even more telling, when asked whether they agreed that using the CASE tool was useful on the job, on average they were between neutral and agree slightly.

Methodologies Used[6]	Total Population[7]	Non Users of CASE	Users of CASE
Life Cycle	126 (60%)	95 (55.9%)	31 (77.5%)
Object-oriented Approach	61 (29%)	49 (28.8%)	12 (30%)
Rapid Applications Development (RAD)	57 (27.1%)	37 (21.1%)	20 (50%)
Prototyping	100 (47.6%)	75 (44.1%)	25 (62.5%)
Joint Applications Development (JAD)	79 (37.6%)	55 (32.4%)	24 (60%)

Table 6: Systems Development Methodology Used

[6] Respondents could indicate that they used more than one methodology.

[7] Counts and percentages

Time Spent on	Total Population[8]	Non Users of CASE	Users of CASE
Systems Analysis	14.72 (18.13)	11.54 (12.20)	28.00 (29.63)
Systems Design	17.13 (14.16)	16.78 (13.64)	18.60 (16.28)
Programming	19.82 (17.81)	21.82 (18.25)	11.45 (13.01)
Testing	15.32 (13.51)	16.68 (13.64)	9.62 (11.62)
Supervising	13.97 (24.53)	13.70 (24.36)	15.10 (15.28)
Maintenance	13.39 (18.02)	14.75 (18.39)	7.7 (15.28)

Table 7: Percentage of Time Spent Doing Various Tasks

Variable	Mean	Standard Deviation	N
Enjoyment	4.49	1.47	52
Perceived Usefulness	4.80	1.56	52

Table 8: Enjoyment and Perceived Usefulness of Case Tools

8. DISCUSSION

This study provides interesting additional insight into how CASE tools are actually being used in business. We replicated prior research that shows low use of CASE tools. It was difficult to find companies that use CASE tools. Within the companies that are using CASE, not many of the systems developers actually use the tool. Interestingly, this low use was not well communicated within the companies. High level managers greatly overestimated what percentage of their systems developers uses the tool. Even managers close to the tool overestimated the numbers of users. When these managers identified the people that they were sure were using CASE tools, about 40% of those who returned the surveys actually were not using the tools. It is easy to understand why systems developers and lower level managers are not sharing this low use with higher level managers. They are aware of the high cost of a CASE tool (estimated to be $22,000 per person including software, hardware, and training [21].) Publicizing this questionable investment might be unpleasant.

We further dissected CASE tool usage in this study. Few of the advanced features within the tool are being widely used. One systems developer said that her company carefully evaluated CASE tools to make sure their tool had all the features possible. Yet she commented that within the company, the major use was as a communication tool between systems developers: one person drew a diagram, others could see it. Most of the features were not used. "We only use what we need to meet the deliverables," she said [Personal interview, 1996.]

What explains this low use of the CASE tools? This study suggests two possible reasons. First, the job that systems developers do is not the same when a CASE tool is used. Systems developers using a CASE tool are more likely to be using a formal methodology then those who do not use a CASE tool. And the way systems developers spend their time when using CASE tools is different then the way systems developers who do not use CASE tools spend their time. More than twice as much time is spent in systems analysis. One CASE tool user commented that it seemed as if they never ended the requirements determination stage when they used CASE tools. This implies that the job is much more formal with CASE tools. Prior research has shown that systems developers prefer jobs that have high autonomy [13], [4]. This increased formality of the job may not be received well by systems developers who have a strong desire for autonomy.

Secondly, systems developers do not appear greatly motivated to use the tools. Neither intrinsic motivation (the tool is fun to use) nor extrinsic motivation (the tool is perceived to be useful) is high. [6] found that these two motivations explain up to 75% of the variance in intentions to use a tool. If systems developers are not sufficiently motivated to use the tool, either because it is not enjoyable or because it is not perceived to be useful, it is not surprising that use is low.

What then can an organization do to attain the expected high productivity benefits associated with CASE tools? Obviously these benefits cannot be achieved without increasing use of the tools. More usage may be attainable

[8] Means and standard deviations

through incentives. However, this study shows that systems developers are fairly neutral about the usefulness of CASE tools. A manager could increase the perception of usefulness by training and reinforcement. Additionally, careful selection of a CASE tool to make sure that it is enjoyable to use would increase usage.

9. REFERENCES

[1] Ajzen, I. and Fishbein, M. Understanding Attitudes and Predicting Social Behavior, Prentice-Hall, Englewood Cliffs, NJ, 1980.

[2] Alavi, M. "Making CASE an Organizational Reality: Strategy and New Capabilities Needed," Information Systems Management (10), Spring 1993, 15-20.

[3] Banker, R.D. and Kauffman, R.J. "Reuse and Productivity in Integrated Computer-Aided Software Engineering: An Empirical Study," MIS Quarterly (15:3), 1991, 375-401.

[4] Couger, J.D. and Zawacki, R.A. Motivating and Managing Computer Personnel, John Wiley, New York, 1980.

[5] Davis, F.D. "A Technology Acceptance Model for Empirically Testing New End-User Information Systems: Theory and Results," doctoral dissertation, Massachusetts Institute of Technology, 1986.

[6] Davis, F.D., Bagozzi, R.P., and Warshaw, P.R. "Extrinsic and Intrinsic Motivation to Use Computers in the Workplace," Journal of Applied Social Psychology (22:14), 1111-1132.

[7] Davis, G.B., and Olson, M.H. Management Information Systems: Conceptual Foundations, Structure and Development, 2nd edition, McGraw-Hill, NY, 1985.

[8] Elshazly, H. and Grover, V. "A Study on the Evaluation of CASE Technology," Journal of Information Technology Management (4:1), 1993.

[9] Forte, G. and Norman, R.J. "A Self-Assessment by the Software Engineering Community," Communications of the ACM (35:4), 1992, 28-32.

[10] Henderson, J.C. and Cooprider, J.G. "Dimensions of I/S Planning and Design Aids: A Functional Model of CASE Technology," Information Systems Research (1:3), 1990, 227-254.

[11] Iivari, J. "Why are CASE Tools Not Used?" Communications of the ACM (39:10), 1996, 94-103.

[12] Laudon, K.C. and Laudon, J.P. Management Information Systems: Organization and Technology, Prentice-Hall, New Jersey, 1996.

[13] Lending, D. and Chervany, N.L. "The Changing Systems Development Job: A Job Characteristics Approach," in Proceedings of the 1997 ACM SIGCPR Conference, F. Niederman, ACM Press, New York, 1997, 127-137.

[14] Martin, M.P. "The Case against CASE," Journal of Systems Management (46), Jan/Feb 1995, 54-57.

[15] Mathieson, K. "Predicting User Intentions: Comparing the Technology Acceptance Model with the Theory of Planned Behavior," Information Systems Research (2:3), 1991, 173-191.

[16] Moore, G.C. and Benbasat, I. "Development of an Instrument to Measure the Perceptions of Adopting an Information Technology Innovation," Information Systems Research (2:3), 1991, 192-222.

[17] Necco, C.R., Tsai, N.W., and Holgeson, K.W. "Current Usage of CASE Software," Journal of Systems Management, May 1989, 6-11.

[18] Norman, R.J. and Forte, G. "Automating the Software Development Process: CASE in the '90s," Communications of the ACM (35:4), 1992, p. 27.

[19] Norman, R.J. and Nunnamaker, J.F., Jr. "CASE Productivity Perceptions of Software Engineering Professionals," Communications of the ACM (32:9), 1989, 1102-1108.

[20] Orlikowski, W.J. "Division among the Ranks: The Social Implications of CASE Tools for System Developers," in Proceedings of the Tenth International Conference on Information, DeGross, Henderson and Konsynski, editors, 1989, 199-210.

[21] Senn, J.A. and Wynekoop, J.L. "The Other Side of CASE Implementation: Best Practices for Success," Information Systems Management, (12), 1995, 7-14.

[22] Straub, D.W. "Validating Instruments in MIS Research," MIS Quarterly (13:2), 1989, 147-169.

[23] Sumner, M. "The Impact of Computer-Assisted Software Engineering on Systems Development," in The Impact of Computer Supported Technologies on Information Systems Development, Kendall, Lyytinen, and DeGross, eds., Elsevier, Amsterdam, The Netherlands, 1992, 43-60.

[24] Sumner, M. "Making the Transition to Computer-Assisted Software Engineering," in Proceedings of the 1992 ACM SIGCPR Conference, A.L. Lederer, ed., ACM Press, New York, 1992, 81-92.

[25] Vessey, I. and Sravanapudi, A.P. "CASE Tools as Collaborative Support Technologies," Communications of the ACM (38:1), 1995, 83-95.

[26] Wynekoop, J.L., Senn, J.A. and Conger, S.A. "The Implementation of CASE Tools: An Innovation Diffusion Approach," in The Impact of Computer Supported Technologies on Information Systems Development, Kendall, Lyytinen, and DeGross, editors, Elsevier, Amsterdam, The Netherlands, 1992, 25-41.