# A MODEL OF CASE IMPLEMENTATION AND AN EXPERIENCE OF UNSUCCESSFUL CASE ADOPTION

***M. H. Selamat***
Dept. of Computer Science, Univ. Pertanian Malaysia, 43400 UPM, Serdang, Selangor, Malaysia

***M. M. Rahim***
Dept. of Computing & Information Systems, Institut Teknologi Brunei, Jalan Muara, PO Box: 2909, BSB 1929, Negara Brunei Darussalam, Fax: 673-2-330776

## ABSTRACT

Computer-Aided Software Engineering (CASE) is a revolutionary technology that seeks to automate the software development process. Current literature reports on many successes of CASE adoption. Despite of such successes, many organisations are experiencing difficulties in implementing CASE technology. Even many powerful CASE tools have become 'shelfware'. It is argued that the benefits of CASE are unlikely to be realised, unless it is introduced based on a sound implementation plan. This article presents a conceptual model to introduce CASE technology within a MIS department. Such a plan facilitates change in software culture and management practice and enables management to deal with confusion and skepticism prior to its occurrence. Furthermore, a case study is described that focuses on a CASE implementation that failed to achieve its purpose. The reasons for the failure are critically examined and explained in the light of the model.

**Keywords** : *Software Engineering, Computer-Aided Software Engineering, Implementation Plan, CASE Failure*

## 1.0 INTRODUCTION

Computer-Aided Software Engineering (CASE) is the automation of software engineering methods and practices (Stobart, Thompson and Smith, 1991). Such automation is a revolutionary progress in software industry that promises to alleviate many of the problems associated with the systems development and maintenance tasks (Laplante, 1991). Several survey studies (Eaves, 1990; Stobart et al ,1991; Kusters, 1993) report that the usage of CASE is growing and developers are citing substantial improvement in quality and productivity through software automation. As such, CASE is often regarded as a miracle solution to confront 'software crisis' (Menendez, 1991). However, it is hard to gain these benefits, unless CASE technology is introduced in a systematic manner. It is because successful automation of information systems requires a good deal more than the purchase of CASE tools (Stone, 1993).

Furthermore, in every MIS department there exists a set of traditions - a culture - that is deeply entrenched. Adopting CASE technology causes disruptive changes in management practice and culture. This is because CASE is often associated with skepticism and fear among both managers and developers. As for instance, senior managers become suspicious about the 'return on investment' from this new technology, while developers are concerned that the 'learning curve' associated with CASE may cause projects to fall behind schedule. Some developers even feel that their hard earned experience would be challenged by the introduction of CASE technology. As such, changes caused by CASE are often met with resistance (Pressman, 1993). A sound implementation plan is required to introduce CASE technology in order to surmount these problems. Without such a plan it is unlikely to achieve the proclaimed benefits of CASE technology. This is supported by McClure (1989), who reported that many powerful CASE tools became "shelfware", because the organisations acquired them without a plan. This is in the line of other authors (Stobart, Thompson and Smith, 1991; Humphrey, 1991; Low and Jeffery, 1991) who emphasised the significance of planning before the introduction of CASE technology. Reid (1993) complained that a common problem with organisations is that they rush into a new technology like CASE without proper planning. As a consequence, CASE technology had failed to achieve its purposes in many enterprises.

This paper presents a conceptual model that seeks to facilitate implementation of CASE technology within a MIS department. It was argued that adequate preparation, understanding, efforts and commitment were required to adopt such an implementation model. Without strong commitment, the model alone cannot guarantee success. Furthermore, a case study was presented to disclose the experience of a Malaysian enterprise that attempted to introduce a CASE tool. Unfortunately, their efforts ended up with failure. The reasons for such failure were critically examined and explained in the light of the proposed CASE implementation model.

## 2.0 CASE IMPLEMENTATION MODEL

A model as shown in Figure 1, is proposed to implement CASE technology within a MIS department. This model proceeds through a series of 5 steps. These steps form the foundation of the CASE implementation plan. This model acknowledges that implementing CASE technology is not a simple straightforward task, and adequate thinking and planning are required for smooth introduction of this technology within the existing culture of a MIS department. Problems may arise from many unanticipated sources. This model helps to generate awareness among IS personnel related to possible problems, and provides some guidelines as how to overcome such problems. The model further enables managers to effectively deal with confusion and skepticism prior to its occurrence.

### 2.1 Process Assessment

The first step is the assessment of the existing process through which software is developed within a MIS department. Such an assessment is important because it brings into light the strengths and weaknesses of the software development and maintenance process. It has been found that many organisations do not have any established process, while developers in some other organisations practice their own way of software development. Even if an organisation has a standard software development process, its managers and technical staff are often too anxious to purchase CASE tools (Stone, 1993). However, most of them have a weak understanding of the procedures, methods, techniques and tools that are currently being applied in their organisations (Pressman, 1993). It is thus suggested that CASE technology should be adopted only once a good understanding of existing software process is gained. Such an understanding forms the basis of the selection of an appropriate CASE tool that would be required to improve the software development and maintenance process. A number of methods like Software Engineering Institute's (SEI) Software Process Maturity Capability Model (Humphreys, 1989; Humphrey, 1991) and Process Adviser (Pressman, 1992, Pressman, 1993) are available to software conduct process assessment within an organisation.

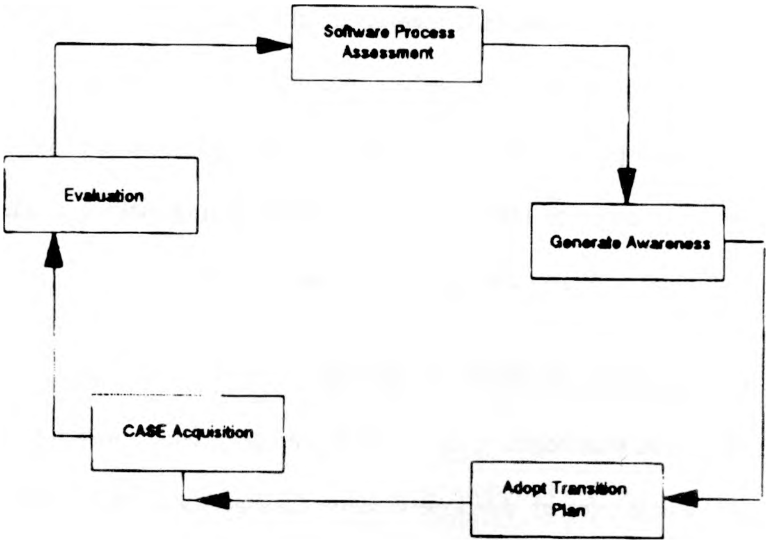

Figure 1 : CASE Implementation Model

### 2.2 Generate Awareness

Once the process assessment is completed, the strengths and weaknesses of the existing software development practices employed by the organisation are known. At this stage, a suitable education strategy needs to be adopted in order to create awareness. Such an awareness improves understanding of both managers and developers concerning the software engineering concepts, procedures, methods and tools and thereby reduces staff resistance (Pressman, 1993). It is argued that without a solid understanding of the weaknesses and strengths of existing process and how technology can improve the existing process and the resulting product, the benefits of CASE technology are less likely to be achieved.

### 2.3 Formulate and Adopt Transition Plan

Once satisfactory level of awareness and knowledge has been raised, a transition plan should be formulated to select, acquire and introduce a suitable CASE tool for the organisation. A committee should be established to perform these tasks. The transition plan should consider several important aspects including training requirements, participants in training, selection of a suitable pilot project, and establishment of metrics to measure of effectiveness of selected CASE tool. A suitable transition plan has been proposed by the authors (Selamat, Othman, Rahim and Khalil, 1993) and is shown in Figure 2. This plan consists of concentric circles. The outermost circle denotes several important aspects that should be taken into consideration during making a transition to CASE technology. Each of these are elaborated in the following sections.

#### 2.3.1 Formation of selection committee

The transition plan begins with the formation of a selection committee. Ideally, such a committee should include selected developers, senior managers as well as some key users. Such participation would assist to overcome the resistance of those who would actually use them once the tool is acquired. The most critical task of the selection committee is to select an appropriate CASE tool that is suitable for the organisation. A number of factors should be taken into consideration by the committee in selecting a suitable tool. These are shown in Table 1.

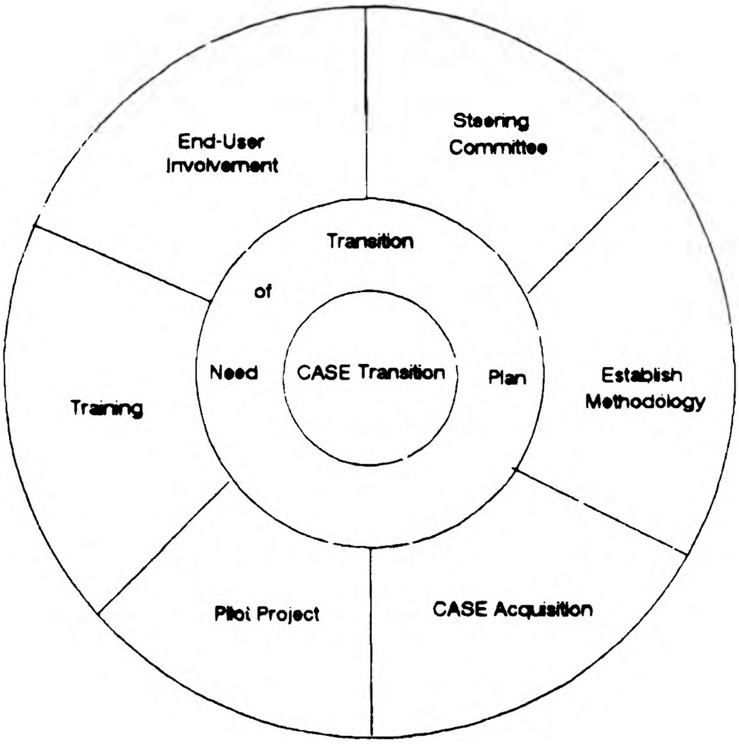

Figure 2 A Cyclic CASE Transition Plan

Table 1 Factors affecting choice of a CASE tool

| [illegible] |
|---|
| Budget |
| Hardware platform |
| Use of existing software development practice |
| Use of existing software maintenance practice |
| Degree of vendor support |
| IS culture of the organisation |
| Consulting services |

An important task of the selection committee is to prepare a realistic budget for the selected CASE tool. While preparing budget, the committee should consider that their organisation is expected to incur substantial cost in training and consulting. It has been reported that many organisations are making a mistake to think of the transition to CASE as a matter of purchasing products (Hamilton, 1990). Actually, the product is only a small fraction of the total cost of implementation. Like the visible tip of an ice-berg, vendors price of a CASE tool actually represents only a small portion of true adoption cost of CASE (Huff, 1992). In fact, construction of a realistic budget is one critical step towards successful CASE adoption.

### 2.3.2 Establishment of a standard methodology

Organisations intending to use CASE tool should adopt a formal methodology for applications development. Menendez (1991) warned that it is no longer feasible to build a system without an engineering like approach. It is unfortunate that many organisations encounter difficulties in choosing an appropriate methodology for developing software systems. To overcome this problem, Knight (1992) provided some guidelines to chose a suitable methodology. These guidelines are listed in Table 2.

**Table 2**: Guidelines to chose a method

| Factors |
|---|
| Method must be scaleable |
| Method must be modular |
| Method must support forward and reverse engineering |
| Method must lend itself to automation |

The establishment of a standard method has been emphasised by Duncan (1989), who suggested that the success of implementation of CASE is facilitated when CASE is married to a methodology in a compatible environment. Without a sound methodology, CASE tools result in misdirected projects (Smith, 1992).

### 2.3.3 Adequate training

Training is an essential accompaniment to the successful implementation of CASE (Duncan, 1989). Therefore, it cannot be ignored (Burkhard, 1989). CASE imposes an engineering discipline on the development of application software. Therefore, training is required at two levels: concepts and tools. However, training would be different based on the types of audience. Managers will be looking for quick courses that give them an overview of the concepts. While, experienced system analysts will be looking for advanced courses that explain the operating details of the CASE tools. The system analysts who form the highly qualified group requires training before, during and after implementation of CASE (Philpson, 1990). However, mere class and workshops are not sufficient for this group. There is a need for 'hands-on' training with CASE tools for them (Bailer et al, 1993). Moreover, each member of this group should receive at least one and as many as three weeks of methods training every year (Pressman, 1993). Such a training duration is important because lack of training has been identified as one principal reason for CASE failure (McClure, 1989).

In addition to training, consultation is important as well in order to ensure success (Reid, 1993). Without experiencing in the CASE technology, people may feel easy to learn only a part of a discipline and revert back to 'old habits', when training is over.

### 2.3.4 Selection of a pilot project

Many CASE consultants agree that the best way to introduce CASE is through a pilot project (Philipson, 1990). A pilot project offers an excellent opportunity to gain experience on how to integrate CASE tool with the existing software development practice within a MIS department. Furthermore, it provides scope to evaluate the effectiveness of the selected CASE tool. Drotos and Burgetz (1990) suggested several useful criteria for choosing a pilot project. These are listed in Table 3.

**Table 3**: Factors affecting choice of a pilot project

| Factors |
|---|
| Size of the project should not exceed six months |
| Project should be important but not too critical |
| Project team should not work under extreme deadlines |
| Project members should be well trained in the use of the selected CASE tool |

However, it is understandable that mistakes may occur and the pilot project may actually take longer time to be completed with CASE than without it. Due to high learning curve, refinement of the tools and the methodology, productivity may decrease. Therefore, pilot projects should be regarded as an extension to formal training (Duncan, 1989).

### 2.3.5 End user involvement

It is vital to recognise that a CASE-aided software development project requires more active user participation. Therefore, efforts should be made to devise means to increase user involvement. It is because joint application design and prototyping requires more commitment and efforts on the part of the end-users. This is supported by Duncan (1989), who emphasised that activities such as joint application design, prototyping and design walkthroughs require users to become more savvy. Smith (1992) reported that in one of the most comprehensive surveys of CASE users to date, 97% of sites that were happy with CASE, involved end-users with CASE output.

### 2.3.6 CASE Acquisition and Implementation

Once approved by the management, selected CASE tool should be acquired and introduced in the light of the systematic transition plan.

### 2.3.7 Evaluation

Some organisations introduce CASE tools with the intention to improve their system development process but dedicates little time to evaluate their impact and effectiveness. As such, it is important to develop ways to measure success before the first CASE-aided project is over (Drotos and Burgetz, 1990). Quantitative or qualitative terms should be devised to determine the increase or decrease in productivity and quality. Productivity can be measured by quantifying efforts and system size (Low and Jeffery, 1991). Function point technique can be used to determine size of a system, while efforts can be computed as suggested by Jones (1986). Smith (1992) reported that in one of the most comprehensive surveys of CASE users todate, 97% of sites that were happy with CASE, made greater efforts to quantify the benefits obtained from the CASE products. The evaluation step actually initiates an on-going assessment of the CASE implementation model.

## 2.4 CASE IMPLEMENTATION MODEL : SOME NOTES

Adopting the proposed CASE implementation model requires exhaustive commitment from all levels of management and MIS personnel of an enterprise. It is argued that management must actively implement such a plan, because a new technology like CASE would rarely succeed without a support from all levels.

## 3.0 A CASE STUDY

This section briefly describes the background of an enterprise where a CASE tool was abandoned soon after it was acquired. Due to the sensitive nature of CASE failure, the names of the enterprise as well as the CASE tool are not divulged. This enterprise is from the transport industry that started operations as early as 1947 in Malaysia. The key objective of this organisation is to provide an efficient and profitable transport system in line with the policies of the government.

### MIS Department: Task Environment

The Management Information Services (MIS) department of this enterprise is one of the largest single computer installations in Malaysia. This department has been developing application systems for the past 20 years. The MIS department uses more than 1 IBM mainframes for applications development and data processing tasks. The total staff in this department is around 350 with 80 to 100 systems analysts and over 100 programmers. The main areas of applications include accounting, payroll, personnel, inventory control, job costing, assets management, and reporting. The department utilises PL/1 for code development. The developers also use COBOL and C languages to a limited extent.·The DeMarco technique is employed as the preferred method for system specification and design. In general, top-down and modular programming techniques are practiced during code development.

**Respondent's Profile**

A senior MIS manager having more than 10 years of experience in applications development agreed to participate in this study. The respondent possesses a Masters Degree in Computer Science.

**CASE Tool: Background**

The MIS department acquired an upper CASE tool in late 1988. This tool supports information engineering methodology (IEM) and operates on the DOS platform. The MIS department was primarily interested to automate the design and analysis stages. Code generation facilities of the CASE product was not important to them. It is due to the availability of plenty of programmers in their department. Data flow diagram feature of the CASE tool was cited as the most frequently used facility.

**CASE Usage: Current Status**

The selected CASE tool was used only once by the developers in developing a low priority small size project. However, during the project development, difficulties were encountered by the developers and the project was abandoned in the middle of the project schedule. Currently, it is not being used by any MIS personnel.

**Purchase of CASE : Reasons**

The responding manager appraised that a senior MIS manager noticed the product was being used in another organisation and found it quite interesting. Moreover, that manager came to know that another government organisation also purchased the same CASE tool. This motivated the manager to purchase the CASE product. Five modules (2 design modules and 3 analysis modules) costing RM20,000.00 per module were acquired.

### CASE Implementation : Actions Taken

The respondent informed that no committee was formed to evaluate and select the CASE tool. Only a senior MIS manager was involved in the selection process. Some briefings and seminars on CASE technology, were arranged in order to generate awareness concerning the strengths of CASE tools before the introduction of the tool into the department. Furthermore, a short training on the product was conducted at the vendor's office. A group of system analysts attended that training programme. After CASE installation, another training was conducted by an external consultant hired by the vendor at the premise of the enterprise. The duration of the training was one week. However, instead of experienced system analysts twenty fresh graduates attended this training programme.

A suitable small scale pilot project was undertaken by several system analysts who attended earlier short training. These group of developers also attended the briefing that aimed to generate awareness about automation. They were highly motivated to use the CASE tool in the pilot project. Unfortunately, the pilot project was unsuccessful. The developers encountered difficulties in applying the tool for developing the application based on the information engineering methodology. Eventually, they gave up their efforts. The message of unsuccessful attempt to use the CASE tool spread very fast across the MIS department. The MIS manager who initiated the CASE tool was reluctant to take any remedial measures to improve the situation. As such, the potential of the tool was not fully exploited and the developers never attempted to use the tool again.

### CASE Failure: Reported Reasons

The respondent reported that developers were not adequately trained in the use of the CASE product. Furthermore, there was a lack of knowledge concerning information engineering methodology (IEM). Developers were not familiar with IEM concept and they never applied information engineering philosophy in developing any system. In general, developers were dissatisfied with the level of training received and the support provided by the management. These two reasons according to the respondent facilitated the failure of the pilot project.

The respondent further complained that the vendor who supplied the CASE tool had limited technical expertise on the IEM as well as on the product. As such, they failed to offer any meaningful support to the developers. The vendor hired an external consultant from overseas for one week who imparted training to the developers. Once, the consultant left, the vendors could not provide useful assistance to the developers. It was stressed by the respondent that the vendors had no personnel who had practical experience in building applications using that CASE product. The low support from the vendor was identified as another major hindrance in successful implementation of the CASE tool.

The only technical limitation that was pinpointed by the respondent was the lack of multi-user facilities offered by the CASE tool. The developers preferred a LAN-version CASE tool. Unfortunately, the tool did not provide that facility at that time.

## 4.0 CASE FAILURE : DISCUSSIONS

It is evident that inadequate attention was paid by the organisation in adopting a systematic transition plan. The organisation failed to establish a standard methodology for software development and maintenance. Once the methodology is in place, the organisation should have formed a committee to evaluate a number of available CASE tools and to recommend a suitable one that best fits priorly established standard methodology. The MIS department was using DeMarco specification and design technique, but the selected CASE tool supported only the information engineering methodology. Unfortunately, the developers were not familiar with the IEM. Furthermore, the training conducted by the vendor did not focus on the information engineering methodology. As a consequence, developers were forced to learn the information engineering methodology as well as the tool at the same time. This learning requirement demanded additional efforts on the part of the developers. The burden of additional efforts together with the lack of knowledge of the information engineering methodology frustrated the developers.

In spite of qualified system analysts, the organisation trained twenty fresh graduates, who were not eventually involved in any CASE related project. This selection of inappropriate participants for the CASE training can be considered as a major flaw on the part of the management.

Furthermore, the consultant who was hired by the vendor did not provide his expert services during the development of the pilot project. It is essential to recognise that mere training on the methodology and the tool is not sufficient to implement CASE technology successfully. In addition to training, consulting is important in order to ensure success (Reid, 1993). Without people experienced in the CASE technology, it is expected that developers would experience difficulties. Therefore, the consultant should have continued his services during the pilot project. Unfortunately, this aspect has been ignored by the management.

Inadequate funding for the engagement of consultants and training can be regarded as a serious lack of management commitment. A technology like CASE requires exhaustive patronage from all levels of management. Unfortunately, management support was perceived to be very low in this organisation. Management failed to train their developers who eventually were not knowledgeable enough to exploit the strengths of automation technology.

The lack of management's commitment in ensuring the continued use of CASE was prevalent because the developers perceived inadequate backing and encouragement from the upper hierarchy. The senior managers did not allocate enough finance for in-depth training on the IE method as well as on the product. The developers felt that the external consultant should have worked with them during the entire period of the pilot project. Unfortunately, management was not keen to bear the fees of the consultant. A summary of the reasons that contributed to the failure of CASE implementation in this organisation is presented in Table 4.

**Table 4** : Key reasons for CASE failure

| Factors | Status |
|---|---|
| Sound CASE transition plan | Not adopted |
| Management commitment | Very low |
| Use of a standard methodology | Not established |
| Vendor support | Unsatisfactory |
| CASE training | Inadequate |
| Developers knowledge of methods supported by CASE tool | Very low |
| Consulting services | Limited |

In summary, it can be argued that the actual reasons for abandoning the CASE tool are different from those cited by the respondent. The organisation failed to recognise the full complexity of their software problems and therefore, did not fully comprehend how to use CASE technology as a solution to their problems. The person responsible for introducing CASE was prompted to acquire CASE technology by knowing that other government organisations are using that particular variety of the CASE tool. This can be regarded as a 'fashion approach' (Mazzucchell, 1992). This implies that the organisation did not examine actual needs for acquiring CASE tools. The adoption of CASE by others facilitated purchase of the CASE tool, even if they did not know how to use the tool. Furthermore, this organisation concentrated on the product and neglected the overall approach that is required to introduce CASE tool within the organisation. Actually, successful application of CASE requires a good deal more than the purchase and deployment of CASE tools (Stone, 1993).

## 5.0 LESSONS LEARNED

The key lesson that can be learned is that introduction of a new technology like CASE in a organisation requires sound transition plan. Furthermore, adoption of CASE requires substantial commitment from all levels of management and MIS personnel. Management should not look upon CASE tool as a solution to their software problems. The tool alone is not a solution to the software problem of an organisation. Wrong tool at the hands of the ill-prepared MIS personnel would produce negative results and would tarnish the image of automation.

## 6.0 CONCLUSIONS

Adoption of CASE technology requires a well-researched planning. Organisations are advised not to rush into this technology without understanding how their system development tasks would be benefited through CASE. Furthermore, organisations should be prepared to commit substantial resources in its implementation. Without a sound plan and commitment from management, it is unlikely to realise the proclaimed benefits of CASE. To support this claim, an experience of CASE failure in a large Malaysian organisation has been described. The reasons for such failure are critically examined and interpreted in the light of the CASE implementation model. The conclusion is that adopting a new technology like CASE without prior planning would rarely succeed. The effectiveness of CASE depends as much on how it is introduced as on the tool itself.

### References

Bailer, B., Bauknecht, K. and Schatzmann, C. (1993) "Ways to CASE: An Analysis of Selected CASE Experiences", *In Proceedings of the Sixth International Workshop on CASE*, Singapore, July 19-23, pp.36-43.

Burkhard, D.L. (1989) "Implementing CASE tools", *Journal of Systems Management*, May, pp: 20-25.

Drotos,D. and Burgetz, B. (1990) "CASE eight starting steps", *Canadian Datasystems*, June, pp: 32-34.

Duncan, M. (1989) "Training is CASE leading edge", *Computerworld*, Vol: 23, No. 4, January 30, pp.81.

Hamilton, R. (1990) "CASE veterans say: look before you leap", *Computerworld*, July 2.

Huff, C.C. (1992) "Elements of a realistic CASE tool adoption budget", *Communications of the ACM*, Vol: 35, No: 4, April, pp. 45-54.

Humphrey, W.S. (1989) *Managing the Software Process*, Software Engineering Institute, Addison-Wesley Publishing Company, UK.

Humphrey, W.S. (1991) "CASE Planning and The Software Process". *Journal of Systems Integration*, November, pp.321-337.

Jones, T.C. (1986) *Programming Productivity*, McGraw-Hill.

Knight, R. (1992) "Methodologies a Must: CASE will Fail Without Proper Use of Tools", *Software Magazine*, January, pp.20-21.

Laplante, A. (1991) "Making a CASE for better training", *Computerworld*, Vol: 25, No. 8, February 25, pp. 100.

Low, G.C. and Jeffery, D.R. (1991) "Software development productivity and back-end CASE tools", *Information and Software Technology*, Vol. 33, No. 9, pp.616-621.

McClure, C. (1989) *CASE is Software Automation*, Prentice Hall International, Inc., U.S.A.

Menendez, D.A. (1991) "The impact of CASE", *Internal Auditor*, Vol. 48, No. 6, December, pp.48-53.

Philipson, G. (1990) *Implementing CASE Technology*, Computer Technology Research Corporation, USA.

Pressman, R.S. (1992) Process Advisor: A Self-Directed System for Improving Software Engineering Practice, R.S. Pressman & Associates, Inc., USA.

Pressman, R.S. (1993) *A Manager's Guide to Software Engineering*, McGraw-Hill, Inc., USA.

Reid, T.F. (1993) "Technology Transfer is not Just Training", *In Proceedings of the Sixth International Workshop on CASE*, Singapore, July 19-23, pp.75-80.

Selamat, M.H., Othman, A.T., Rahim, M.M. and Khalil, I. (1993) "A Model for CASE Implementation: A Malaysian Experience", *In Proceedings of the Sixth International Workshop on CASE*, Singapore, July 19-23, pp.11-17.

Smith, W.M. (1992) "CASE works, but not alone", *Computerworld*, Vol: 26, Issue: 15, April 13, pp.74.

Stobart, S.C., J.B. Thompson and P. Smith (1991) "Use, Problems, Benefits and Future Direction of Computer-Aided Software Engineering in United Kingdom", *Information and Software Technology*, Vol: 33, No:9, pp: 629-636.

Stone, J.A. (1993) *Inside ADW and IEF, The Promise and Reality of CASE*, McGraw-Hill, Inc., USA.