Computer Aided Software Engineering (CASE Tools)

Er. Gurpreet Singh, Er. Manjit Thapa, Er. Sarbjeet Singh, Er. Sukhveer Singh and Er. Pankaj Sharma
Department of Computer Science, Sri Sai College of Engg. & Tech. Badhani,
chohan87@gmail.com , manjit.thapa@yahoo.co.in , sarbaish@gmail.com , ss.sukhbeer@gmail.com , pankajuppal22@gmail.com ,

ABSTRACT: CASE (Computer Aided Software Engineering) technologies are tools that provide the assistance to the developer in the development of software. Main purpose of the CASE tools is to decrease the development time and increase the quality of software. Even the presence of these qualities CASE tools are not being used most often or freely. These tools are not used freely as they should be; there are some points that need improvement, so that the use of CASE tools can be increased. And finally the hurdles in the promotion of the CASE tools as a standard.

Keywords: CASE tools, Building blocks, Implication of CASE, Factors, Structural model of CASE.

1. INTRODUCTION

In this paper we will discuss computer aided software engineering (CASE) and how use of CASE tools help in making the software development process. CASE tools reduce the time and cost of software development and ensure the quality of software. The objective of introducing Computer Aided Software Engineering (CASE) tools was the reduction of the time, cost of software development and for the enhancement of the quality of the systems.

Case tools can be used to some activity associated with software development and shown in figure 1.

1. Upper Case
   a. Requirements
   b. Specifications
   c. Planning
   d. Designing

2. Lower case
   a. Implementation
   b. Integration
   c. Maintenance

Upper CASE tools are used to capture, analyze and organize the models of system. These models help designers to focus on the systems linear behavior. Lower CASE tools are used for development and software maintenance phases. Using these tools developers classify the scope and boundaries, describe current system, model requirements, prototyping, prepare design, etc. Integrated CASE tools provide support for Upper CASE and Lower CASE tool activities. Integrated CASE tools helps specifically in Analysis & Design, Maintenance and system planning. Integrated CASE tools provide the support throughout whole development lifecycle.

The primary objectives of deploying CASE tool are:

a) To increase productivity.
b) To produce better quality software at lower cost.

2. **Building Blocks For CASE**

Building blocks can be implementing by CASE environment and used by CASE tools, integration frame work, portability service, operating system, hardware platform & environment architecture as the software development processes as shown in figure2. Tools vendors can be contacted by the web.

![Building blocks diagram](image)

Figure2. Shown as building blocks of CASE tools.

These tools are used for software computer aided design & followed by basic points are:

a) Case tools are available that help developers to analyze, design and for the documentation of the software. Even CASE tools are available for deployment and maintenance of the software. Tools like Erwin, Rational Rose has create there importance in the market due to their qualities and reliability.

b) Integration frame work specialized programs allowing CASE tools to communicate with one another

c) Portability allows CASE tools and their integration framework to migrate across different operating systems and hardware platforms without significant adaptive maintenance.

d) Operating systems are due to database & object management service.

e) Hardware platform is a development system & choice of your hardware, pc or workstation.

Hardware platform depending upon a factor as:

a) Cost depending.

b) Performance & stability.

Case Environment supports for programming language, prototyping, structured analysis, project management system, hardware & software etc.

3. **CASE Environment Model**

CASE Environment models are realized only set of integrated into a common frame work. In CASE model, a programming environment is an integrated collection of tools to support only the coding phase of software development. The tools commonly integrated in a programming environment are a text editor, a compiler, and a debugger. These tools are integrated. If Environment models are not integrated and data generated by one tool would be input to the other tools. As shown in figure3.

![CASE Environment Model diagram](image)
This is another name of central repository. Central repository has the following function:

a) Data integrity – include function to validate entry to the repository and ensure consistency related object.
b) Information sharing – provide mechanism for sharing information among multiple developers and tools, control modification of information.
c) Data tool integration – establishes share data item and perform configuration management system.
d) Data – data integration – data base management system allowing access to related objects so functions can be achieved.
e) Document standardization – definition of object of database leads directly to a standard approach for creation of engineering document.
f) Project management - contain information about the software application and the organization general process for software development phases, task & deliverable.
g) Data Dictionary – should provide view and update access to data items and their relations stored in it. It should have print facility to obtain hard copy of the viewed screens. It should provide analysis reports like cross – referencing Impact analysis. It should support a query language to view its contents.

4. Benefits of CASE

Benefit case used for CASE environment or event from the use of isolated CASE tools:

a) CASE tools help produce high quality and consistent documents; redundancy in the stored data is reduced.
b) Use of case tools leads to considerable improvement to quality.
c) CASE tools reduce the drudgery in a software engineer’s work.

5. Need of CASE

Need of CASE developed by software engineering and has the followed by points:

a) Software developers always looking for such CASE tools that help them in many different ways during the different development stages of software, so that they can understand the software and prepare a good end product that efficiently fulfill the user requirements.
b) CASE tools provide the ways that can fulfill this requirement of software developers. These tools provide computerized setting to software developers to analyze a problem and then design its system model.
c) Computer-aided software engineering (CASE) encompasses computer-based procedures, techniques, and tools which can be used to develop, maintain, and reengineer software. CASE is to the software engineer as computer-aided design/computer-aided manufacturing (CAD/CAM) (q.v.) is to the mechanical engineer and computer-aided electrical engineering (CAEE) is to the electrical engineer.
d) Software engineering refers to the systematic approach to the development of computer software. This systematic approach defines various stages (requirements engineering, Software design, Implementation, Installation, Maintenance and retirement) that software has to undergo in a paradigm
known as the Software Development Life Cycle.

e) CASE tools encapsulate the software tools for the project management, software development, management and maintenance, business and functional analysis for the technology implementation, system design, code storage, compilers, translation tools, and test software.

f) CASE tools are need for deployment and maintenance of the software. Tools like Erwin, Rational Rose has create there importance in the market due to their qualities and reliability. The support provided by a toolkit is limited to programming, configuration management and project management. They are still in the process of improvement like the other fields of the world, because every human created thing in this world has always room for improvement.

6. Factor of CASE

CASE tools facing many troubles in creating a suitable place in market e.g.

a) Even after the completion of the design it is not necessary that it will fulfill the requirements.

b) Though CASE tools are helpful for the developer but do not assure that the design is according to the requirements.

c) Good quality CASE tools are very expensive and prove costly for the development.

d) CASE tools also required training for the user that increase the overall cost of development.

e) Almost every tool has its limitation that decreases its use and popularity.

f) Some tools may have very limited functionality and may not address all possible domain activities.

g) Every tool has a specific methodology for designing and modeling of the system. Due to this you’re bound to follow them that decrease the flexibility which decrease the use of CASE tools.

h) Frequent users get used to it and afterward developers try to use the same approach and tool for other projects whether the tool addresses the target projects problems or not.

7. Implication of CASE

Need for comprehensive CASE plan framework that considers estimated requirements for people, time, and money in the course of CASE adoption (awareness, commitment selection, and implementation). Case is used of requirement as cost, time limitation, lack of concern and technical limitations. As shown by figure 4.

![Implication of CASE](image-url)

Figure 4. Implication of CASE.

a. **Cost** - Many tools are costly and most companies are unwilling to implement these CASE tools by the fact that it increases the overall cost of the project. And second thing if a company does not get any extra by using CASE tools then
why should they use CASE tools, which are expensive for it in the end.

b. Time limitation - There is always a time limit for every software project for development. Most of developers do not adopt CASE because it requires a lot of time to train developers and perform all CASE activities and there are always deadlines to complete the project.

c. Lack of concern - Most of the developers who use CASE tools are not fully satisfied from the CASE tools, which are why they are quite neutral about the usefulness of the CASE tools. The developer does not fully enjoy using CASE tools and if some of them do so, they use only limited functionality of the CASE tool. Many developers use CASE tools as a requirement of organization only.

d. Technical location - Every CASE tool follows a methodology for the model the system. People who use any specific CASE tool for a longer period of time get used with the methodology of that tool and they try to apply the same methodology for other projects.

Most of the tools have their hardware and software requirements. These requirements should meet to use that tool.

8. Conclusion
In this paper, a CASE tool plays an important role in Software Development and getting its appreciation in software development industry slowly but surely. But CASE is still in growth and there are different perceptions about CASE. CASE tools are not being used as they are expected to.

9. References


