Comparison of Software and Hardware Design Tools (CASE vs. Simulators)

Lingaraj A. Hadimani, Dinesh Kumar Saini, Vaishali P Khoche and Sanad Al Maskari

***Abstract*— This paper discusses automated Tools for achieving productivity and efficiency of design. Efforts are made to quantify the usage, benefits and comparison of two tools used for design namely- Simulators and CASE tools. While simulators are used in many areas of engineering with specific purposes, CASE tools are used specifically in software development. The benefits obtained help engineers in analysis and design in terms of cost benefit analysis and productivity improvement.**

Index Terms: Software, Hardware, Simulators, CASE tools, Productivity.

I. INTRODUCTION

Today engineers are aided by design and analysis tools which increase productivity, decrease cost, minimize effort, achieve better productivity, better perception, better change management, better reliability and adoption of better process. There are field specific design and analysis tools in all fields of engineering. But there is a particular set of tool called simulators, which find their use in various fields of engineering. Simulators help in analyzing problems using models of system. They help in conducting experiment, which are otherwise costly, or time consuming. They help in training too [1].

There is a particular set of tools called Computer Aided Software Engineering (CASE), which are used in software development.[2] Their usage is analogous to similar tools in other engineering in the sense that they help in analyzing and designing the software.

First of all, we investigate simulators and CASE tools in detail to understand the morale of their usage and then present comparative studies. For simulators we understand what is simulation, the process of simulation, examples of simulation studies, simulator's usage, simulator cost justification and case studies of usage of simulators in industries.[3,4,5]

Manuscript received April, 17, 2011; revised April 17, 2011.

Lingaraj A. Hadimani is with Caledonian College of Engineering, Muscat, Sultanate of Oman.(lingaraj@caledonian.edu.om), Member IAENG.

Dinesh Kumar Saini is Sohar University, Sohar, Sultanate of Oman, (dinesh@soharuni.edu.om)

Vaishali P Khoche is with Brahma Valley College of Engineering and Research Institute, Nasik, Maharashtra, India (vaishu_khoche@yahoo.com)

Sanad Al Maskari is with Sohar University, Sohar, Sultanate of Oman, (sanad @soharuni.edu.om)

II. COMPARATIVE STUDY

A. Conceptual Differences

It may sound strange to compare two very distinct tools. So let us begin with some conceptual differences. In later section we will show how they are similar too.

- Case tools aid software engineer whereas simulators
- aid all other engineers.
- Simulator's chief goal is to verify a design through simulation experiment results. Case tool's chief goal is to assist a design.[6]
- Simulation is a supplementary activity. After simulation is over we go back to redesign or workout of design. Whereas case tools are a part of software life cycle. The design produced using case tools is used for next activity.
- Simulation is an old concept and is applied in various fields. Case tools came after we had systematic approach towards software engineering established.
- Single user can conduct simulation. Since software projects are generally team projects, team as a whole uses case tools [7].
- Simulators are used for verification and validation. Simulators are very useful in critical system verification. Case tools by themselves do not let us verify or validate our design. We need to do that through other techniques.
- CASE Tool helps in static verification of system while simulator help in dynamic verification of system. [8]

B. Impact on processes and quality

Comparison is made between CASE tools and Simulators by considering by considering the process/quality parameter and the same is shown in Table II.

III. SIMULATION IN SOFTWARE ENGINEERING

Simulation is a general methodology usually applied to computation intensive, complex, modeled situation. It can be applied in software engineering too [9].

Let us see how simulation is helping in the software industry.

A. Requirements gathering

We can simulate the software system that we are going to build. It will reveal many characteristics of its dynamic behavior that will help to gather requirements more precisely, especially the nonfunctional requirements that need quantified values.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

B. Model Building

Prior to software development or during analysis, if we do simulation then in that process we build a model of system.

Model built for the purpose of simulation is mathematically sounder and is useful at the time of verification of our system. Simulation gives us better understanding of our system.

C. Project Cost Assessment

Simulation can provide not only estimates of cost, but also estimates of cost uncertainty. Simulation is a powerful tool to aid activity-based costing, and can incrementally accumulate costs to a very fine degree of resolution. In addition, it can assess the uncertainty of costs based on the interacting uncertainties of independent variables

D. Project management

Simulation experiments can give project managers some predictions that will guide important project management decisions. Simulation can reveal what aspects will be computationally complex and will require extra effort.

E. Process Improvement

This refers to software process simulation. We can observe the behavior of newly adopted or yet to be adopted software process through simulation.

F. Testing

The model built during simulation and the results that predicted expected behavior of system helps in effective testing of the system. Simulation results were quantified. This gives us better verification of system and its expected behavior.

IV. What simulators can't do?

- Simulators do no help in Code Generation, Database Schema Generation
- Simulators do no help in debugging
- We cannot go for reverse engineering
- They does not provide any help in documentation, Text Editing, Graphical Editing, Publishing
- Simulator does little help in developing High Level DFD, State Transition diagrams, Object Oriented Models.
- Simulators have no Language Support (Compiler, Linker and Assembler)

V. WHAT CASE TOOLS CAN'T DO?

Case tools are the automated design tools and it is very useful in the software design and provides very useful supports to the designers but everything the case tools can't solve so some of the issues which can't be solved. These are some of the limitations which are listed

- CASE Tool cannot do numerical experiments in problem domain.
- CASE Tool cannot be used in training.
- Usage of CASE Tools is limited to software engineering.
- They do not emulate virtual environment.

VI. Characteristics for comparison or evaluation

Any automated software tool like simulator or CASE can be evaluated based on following criteria's which are very essential for consideration when comparison is made for the evaluation of performance to achieve the design proficiency. For the design point of view the comparison is not only done for performance parameters but is done for learners point of view , trainers point of view and like that the listed parameter are also compared which are as follow

- Ease of learning,
- Efficiency,
- Throughput,
- Response time
- Ease of Installation Portability

Ease of learning is one of the important characteristics, performance issues related to throughput of the tool, how much time the tool takes as response and portability issues for the new upcoming version of the updated tools are some of the parameters of the characteristics of the tools must be considered seriously.

Comparison of CASE tools and simulators on similar characteristics is given in Table I.

VII. CONCLUSION

An attempt is made to identify the similarities and difference between the design process in software and hardware industry. Software is designed with help of automated tools called as CASE tools and hardware is designed using simulators. Both types of automated platforms help designers to improve design efficiency. The impact on the process quality is studied for both CASE and simulators.

References

[1] "IEEE Recommended Practice for Evaluation and selection of CASE Tools". I*EEE Std. 1209-1992.*

[2] A. Javor, "Proposals on the Structure of Simulation Systems", Central Research Institute for Physics of the Hungarian Academy of Science, Budapest, Hungary, 1982 Alan M. Christie Simulation - An Enabling Technology in Software Engineering software Engineering Institute

[3] Banks, J., and Gibson, R. " 10 Rules for Determining When Simulation Is Not Appropriate." *IIE Solutions,* September 1997, pp. 30-32.

[4] David Raffo "Software Process Simulation Modeling: Workshop Report" School of Business Administration, Portland State University, Portland, Oregon 97207-0751

[5] Glenny, Neil E. and Gerald T. Mackulak. " Modeling and Simulation Provide key to CIM Implementation and Policy." *Industrial Engineering.* May 1985.

[6] Herald, Charles, Ghosh, Biman K. and Bowden, Royce O., *Simulation Using ProModel*, second edition McGraw Hill 2003.

[7] John F. Brock, Cynthia Jacobs, Richard Buchter, Milestone Group Arlington, Virginia "Design of a guidebook for the acquisition and use of driving simulator for training transit bus operators". *http://ppc.uiowa.edu*

[8] Jongmoon Baik "The Effects of case tools on software development effort", *http://sunset.usc.edu/publications/TECHRPTS/2000/usccse2000-504/usccse2000-504.pdf*

[9] Law, A.M. and McComas, M. G. "How Simulation Pays Off." *Manufacturing Engineering*, February 1988, pp. 37-39.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[10] Mott, Jack and Kerim Tumay. "Developing a Strategy for Justifying Simulation." *Industrial Engineering*, July 1992, pp. 38-42.
[11] R J Offen, "CASE Tools and Constraints", Technology and Management University of New South Wales Sydney 2052, Australia *http://www.caeser.unsw.edu.au/publications/pdf/Tech00-1.pdf*
[12] Ross Jeffery, Fredrik Utbult, Kevin Chung, and Sabine Bruynincx "The Effect Of Constraint Notification Within A Case Tool Environment On Design Productivity And Quality", Technology and Management University of New South Wales Sydney 2052, Australia *http://www.caeser.unsw.edu.au/publications/pdf/Tech00-11.pdf*
[13] Smith B, Gaba D. Lake C, Blitt C, Hines R, "Simulators Clinical Monitoring: Practical Application." New York, NY: WB Saunders Company. 2000

TABLE 1
COMPARISON OF CASE TOOLS AND SIMULATORS

Characteristic	Simulators	CASE tools
Training required	Simulator is employed in many generic fields. Engineers that will use simulator will possibly be unfamiliar to software tools and simulation itself. So initial training can be more	Software engineers quickly learn and appreciate the benefits of CASE tools. Tool adoption is easier than simulator
Hardware cost	Special purpose devices and high computing power computers are needed	Normal affordable computers will do. No special purpose hardware is required.
Software cost	Industrial simulators are very costly. Investment in them need to thoroughly discussed among decision makers	Many software companies build their own custom CASE tools that suit their needs and software practices. General-purpose CASE tools are costly but are welcomed by competent software companies for long-term benefits.
Skilled personnel cost	To handle the tool minimal skilled personnel will be required. But we need skilled engineers that can utilize the power of simulation in designing and analyzing problems	Skilled personnel are required initially because adoption of CASE Tool affects software processes. Team should get adapted to new process and the tool itself.
Computation cost	Simulation experiments are highly computationally expensive. Some of them require hours of computation.	No such need. They help in design do not do computations compared to simulators.
Scope in project	There scope is mainly in analysis and design of project	Their scope spans entire project.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE II
COMPARISION OF CASE TOOLS AND SIMULATOR

SN.	Process/Quality parameter	Simulators	CASE tools
1	Problem domain modeling	Simulation experiments cannot be conducted until a numerical model of problem domain is available. However a simulator software it self can help to build o model of system to be developed for the purpose of design and experimentation.	CASE tools extensively help in building the model of system. They aid through support of standard graphical representation technique of modeling. Model developed in a CASE tool is referred to in all subsequent steps.
2	Problem definition	No tool will fully automate this step however results of numerical experiments in simulation help to identify various problem aspects.	CASE tools do not directly help in this step. But they help in structuring and organizing requirements resulting from problem definition
3	Project planning	This may be additional feature. This is not an objective of simulator. However results of simulation experiments guide in better project planning because we gain better understanding of problem.	Modern CASE tools help in project planning through their added feature because it is an important part of software engineering project. This facility can be a separate feature or separate software from same CASE tool vendor (For example RequisitePro for Rational Rose)
4	Risk analysis	Simulation experiments help in itute assessment of risks. Such assessment depends on kind of experiment and kind of interpretation from that.	There may not be any direct support but with better understanding and process adoption risks are minimized.
5	Project cost	Depends. Initial investments in simulators are more but overall budget is less due to many reasons. Interested reader can refer to our section 1.5 on simulation cost	Project cost gets minimized for reasons similar to that of simulator.
6	Team efficiency	Team understands and experiment in simulated environment, which is quicker way as compare to physically doing experiments.	Definitely increases. After some initially training team performs better. All phases of product development gets efficient.
7	Product performance	Simulators help in better problem analysis and thus adoption of better process. With the usage of better process of building the product, it expected to be of better quality.	Product is better engineered and thus is of better quality.
8	Change management	By conducting experiments in model itself we can see the impact of a change and can thus better incorporate it.	Better organization of design helps in better change management. Additionally due to vertical integration change gets propagated well.
9	Product enhancement	Conducting simulation for the product at various scenarios to judge performance in various ways under various conditions.	It is easier to go back to previous design and do changes for new enhancements because of better organizing of design. CASE tools help in versioning of our design.
10	Process improvement	Used in building model, doing analysis by conducting experiments, designing using such automated tool which gives virtual environment is ultimately the adoption of better process.	CASE tools help in over all management of software. We adopt an engineered approach with standard notation and procedures.
11	Project time	By conducting simulation experiments we minimize late changes thus project finishes within time.	With better understanding of problem domain the subsequent implementation takes lesser time.
12	Manpower required	Initially we may need to hire some skilled people.	Initially we may need to hire some skilled people but later due to automation
13	Skills required	There is an initial learning period for any such tool. But in subsequent work efficiency improves.	CASE tools will not be new and hard to learn for software engineers. They will appreciate the benefits they achieve.
14	Problem analysis	Simulators extensively help in analysis of problem through the conduction of various experiments in problem domain.	Analysis is done by building various diagrams like state chart diagram sequence diagram, dataflow diagram etc.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)