Software Reengineering: New Approach to Software Development

Harmandeep Singh

Assistant Professor

P.G.Deptt.C.sc & I.T

SGAD College,Khadur sahib-143117

Abstract: Programs are often reengineered for better maintainability in the way to make new softwares with the help of programs re-engineering..Program reengineering is not a simple task because code reengineer or code reengineer is such a difficult and a challenging task. In software reengineering the existing system is reengineer or refined to produce the new software. In this research paper we try to describe the activities that are helpful in reengineer the existing software or program to produce the new one.

1.INTRODUCTION: Re-engineering means "the examination or alteration of a subject system to reconstitute it in a new form and subsequent implementation of that form".[3] Reengineering   may occupy re-documentation, restructuring and reorganizing of the existing system. Existing system and translated into modern system with the help of modern programming languages. May modify the existing data values or updates the data values. Software reengineering is concerned with re-implementing legacy system to make th.bem more maintainable. The total functionality of the system is not changed, sometimes the architecture of the existing remains the same. Most popular used programming languages by the software engineers to reengineer existing systems are: Java,C++, PROLOG, ADA etc.

Factors which affect re-engineering costs are as follows:

- Availability of Experts
- Tools
- Quality Metrics
- Data conversion Methods:

Availability Of Experts: The engineers required   for maintaining and reengineer the system will increase development costs.

Tools: Requirement of tools such as CASE(computer aided software engineering) tools to mechanize most of the program changes.

Quality Metrics: If the quality check metrics are not used, then there may be lower quality product developed so we have to use Quality metrics for the   Development of software product.

Data Conversion Methods: If we have to convert large volume of data, this will increase the development costs.

2. Scope of Re-engineering

Reengineering means renovation, it is basically the examination and alteration of the existing system to reconstruct in a new manner and the subsequent implementation   in the new form[1]. Reengineering is the process of transforming the existing system into a new system with the addition of some new features. Re-engineering helps the firms and organizations to regain investment in software and preserve its corporate memory. Reengineering helps to reduce the organization's fruition risks. With the help of reengineering, changes are easily make to softwares and improves software reusability.
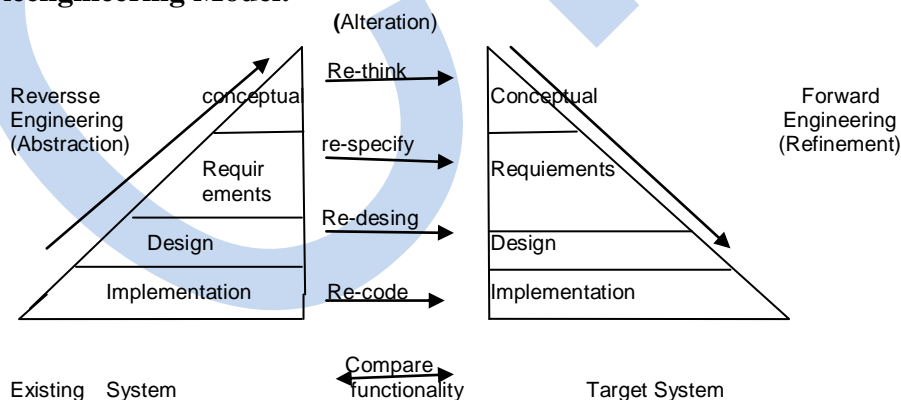
3. Software Reengineering Model:

Figure: Software Re-engineering Model[2]

Re-engineering starts with the source code of an existing legacy system and conclude with the source code of a target system(Proposed New System). In this phase the existing system(Traditional system) may convert into the new(Modern system) system.i.e, from language to other or from one operating system to another. The figure shows the software reengineering model, indicates the processes for all

the levels of reengineering based on abstraction used in the software development[2]

4. The model of Software reengineering above applies three principles of re-engineering :

4.1.Abstraction

4.2.Alteration

4.3.Refinement

Abstraction is a gradual increase in the abstract level of system. System representation is formed by the successive replacement of existing detailed information with the information that is more abstract.

4.1.Abstraction produces a   representation that emphasizes certain system characteristics by suppressing information about others. This upward movement is called Reverse Engineering [3]

4.2.Alteration is the making of one or more changes to a system representation without changing the degree of abstraction including addition, deletion, and modification of the information only not the functionality.[3]

4.3.Refinement: Refinement is the gradual decrease in the abstraction level of system representation and is caused by the successive replacement of existing system information into more details. This process is termed as Forward Engineering and resembles software development of new code, but with some refinements of the process.

5.Approaches to Software Reengineering:

The term structure refers to process activities and the relationships among them. To model the re -engineering process these factors must be identified and their effect on the process understood. There are several basic re-engineering approaches. Each approach embodies a different means of achieving the same goal: creation of the target system(Re-engineered system version)[4]

5.1. Lump-sum re-engineering

5.2. Incremental re-engineering

5.3. Partial re-engineering

5.1. Lump-sum re-engineering: This approach re-engineers an entire system in one sweep. Once the re-engineer effort begins it continues until all project objectives are satisfied and the target system is produced. Figure 5.1. shows this approach as a black-box process with the existing system as input and the target system as output. This approach results in monolithic projects and not always suitable.

Existing System   Target System

Figure5.1.: Lump –sum Re-engineering Approach[4]

5.2. Incremental re-engineering: This approach, the entire system is re-engineered gradually. One or more intermediate systems satisfies more project goals than preceding the intermediate system. The target system is produced when all project goals are satisfied. Figure 5.2. shows the incremental approach as a series of black boxes.
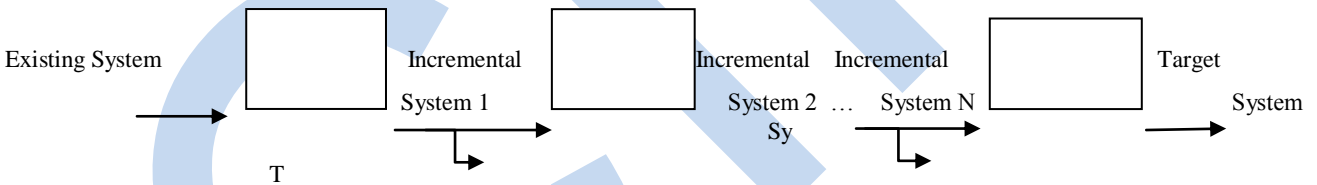
Existing System   Incremental System 1   Incremental System 2   Sy   Incremental System N   Target System

T

Figure5.2.:Incremental Re-engineering Approach[4]

5.3.Partial re-engineering: In partial re-engineering, only a part of system is re-engineered. The re-engineered system portion must integrate with the existing non re-engineered portion, which reduces the potential improvements to the system. First, the existing system must be divided into two portions: a to be re-engineered portion and a non-to-be-changed portion. The to-be- re-engineered portion can be part .

of the system that must be changed or a larger portion of the system that consists of the part that must be changed and another part whose inclusion simplifies the interface to the remainder of the system. Second, the re-engineering work must be done. Third, the portions of the system must be merged to produce the target system. Figure5.3. shows three steps of partial re-engineering approach
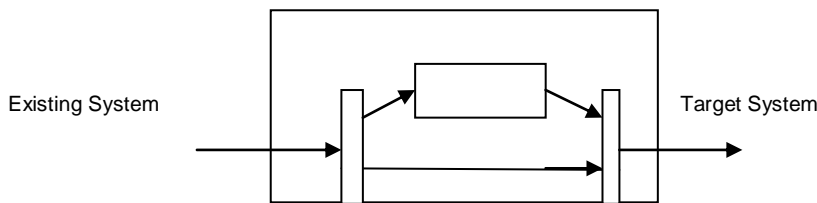
Existing System   Target System

Figure5.3. Partial Re-engineering Approach[4]

Conclusion: Re-engineering becomes very useful for software development with the help of existing system to produce the new and more featured software model .Re-engineering not completely change the existing system but with the help of existing system, produce the new and modified model take some functionality of the already working system and adds some functionality to produce the new software product. More systems will be produced from the existing system with the help of re-engineering without changing the functionality of the already working system.

References:

[1] Byme, Eric J,"Software Reverse Engineering: A Case study," Software practice & Experience,vol 21,no.12, Dec 1991,pp 1349-1364.

[2] Colbrook 901,A., Smythe,C., Darlison,A.,"Data Abstraction in a software reengineering Reference Model," Conference on Software Maintenance, November 26-29,1990.pp.2-11

[3] G.kaur,J.Kaur, P.Kaur, A.Kaur," Software Re-engineering-A state Of Art" in ETCIT,2011.

[4] Chilkofsky, Elliot J., Cross, James H..," Reverse Engineering and Desiugn Recovery: A Taxonomy," IEEE Software,Vol.7,No.1,jan 1990.pp 13-17