

Reverse Engineering

Suyash Painuly

Department of Computer Science Engineering, Dronacharya College of Engineering, Farrukhnagar, Haryana, India

Abstract- Reverse Engineering also known as back engineering is nothing but a process of reexamining the design or architecture of an object to extract knowledge from the object. The purpose to apply reverse engineering to an object or any software is to gather maximum and detailed knowledge about that object so that later on it could be re-engineered in a better way by using latest and affordable technology to build it.

I. INTRODUCTION

Engineering is basically a branch of science and technology where scientific principles are used to design, manufacture, construct, and maintain the products, systems, and structures. At a deeper level, there are two types of engineering: forward engineering and reverse engineering.

Forward Engineering is a step by step approach to build a fully functional device or software using raw materials or raw code which itself is of no use. Each and every material is composed with each other in such a way that they could perform certain operations and fulfil the needs of the user and serve the purpose for which it is built. For forward engineering to be implemented there are several steps through which the builder has to go through, that are: Requirement analysis, Design, Implementation, Testing, Deployment and Maintenance.

Reverse Engineering helps one to gain more knowledge and understanding about a man-made object or a software. In this process the object is dismantled step by step and its structure is thoroughly examined in order to gain better knowledge about that object or software. Reverse Engineering can be applied to the fields of Mechanical Engineering, Electronic Engineering, Software Engineering, Chemical Engineering and Systems biology.

II. USES OF REVERSE ENGINEERING

A. Product Security Analysis

Reverse Engineering is useful in acquiring sensitive data by breaking up and analysing the structure and design of system component. While disassembling the object major flaws could also come up which were missed or ignored earlier and that object can later on be re-engineered without these flaws.

B. Obsolescence

Integrated Circuits after some time become obsolete and cannot be used further due to the advancement in technology because of which these parts can no longer be maintained, so the only way to transform these circuits into a better version of themselves is to reverse engineer the chip and redesign it using new and better technology.

C. Repurposing

The obsolete objects after being reused and applying various upgrades are proposed again to the users which are ready to be used.

III. APPLICATIONS OF REVERSE ENGINEERING

There are several types of reverse engineering which is implemented in several areas -:

A. Reverse engineering of machines

As computer-aided design (CAD) is in demand and is desirable, reverse engineering has become a practical method to create a virtual model of an existing physical part for use in CAD or any other software. The reverse-engineering process involves measuring an object and then redesigning and reconstructing it as a 3D model. The measured data alone is represented as a point cloud which lacks topological information and design intent.

B. Reverse engineering of software

The two basic components of reverse engineering are redocumentation and design recovery. Redocumentation is the process of creating new representation of the computer code so that it'd be much easier to understand. While, design recovery is the reasoning from general knowledge or personal experience of the product in order to get the knowledge about product functionality. In Waterfall model, the output of the implementation phase is reverse-engineered back to the analysis phase. In Reverse engineering only examination of the software or an object is done and no modification is done to that software. Reverse engineering can be performed at any stage of the product cycle, it is not necessary to reverse engineer the functional end product.

C. Binary Software

Whenever the source code of a software is not available, Binary reverse engineering is performed. This process is also known as reverse code engineering. In the process the software is decomposed into several smaller modules using various software like Java decompiler (JAD) is used to decompile binaries for java. Reverse engineering of software is protected in the United States by the fair use exception in copyright law.

Some of the software reverse engineering techniques are:

- Analysis through observation of information exchange is most common protocol in reverse engineering, which involves bus analysers and packet sniffers. Sometimes, reverse engineering on embedded systems is aided by tools introduced by the manufacturer. In Windows Operating System, debuggers like Soft ICE are popular.
- Disassembly of a program is done using a disassembler which means, the raw machine language of the program is read and understood in its own terms, only with the aid of machine-language also known as mnemonics. This works on any computer program but is a bit slow and take quite some time. The Interactive Disassembler is a particularly popular tool.
- Decompilation is a process that tries to recreate the source code using different outputs/results from that program in some high-level language

for a program available only in machine code or bytecode.

IV. REASONS FOR REVERSE ENGINEERING

There are several reasons to do reverse engineering, some of them are listed below:

- The primary manufacturer of a product no longer produces the product.
- Sufficient documents of the original design are not available.
- The original manufacturer no longer exists, but there's a demand for that product in the market.
- The original design documentation is no longer available or never existed.
- There are some flaws in the product which need to be designed out.
- To brace the features of a product based on long-term usage of the product.
- To analyze all the features (good and bad) of the product.
- To explore new course of action to improve the performance and features of the product.
- To gain guiding principles to understand competitor's products and develop better products.
- The original model is not sufficient to support modifications or current manufacturing methods.
- The original manufacturer cannot supply or unwilling to provide additional parts.
- The original equipment manufacturers either don't want to or unable to supply replacement parts, or they may either be demanding inflated costs for sole-source parts.
- To update obsolete material with more current and less-expensive technologies

IV. CONCLUSION

Reverse engineering is an important aspect to build an advanced and better version of any software or an object as it allows one to understand the object in a detailed manner and also about each and every component or module used to build that thing. Because of reverse engineering one could know how different things act when combined with different other things and how can different combinations of materials or products can lead to manufacture

different useful technologies and stuff. Reverse engineering can also help to re-engineer various products which can no longer be used or maybe outdated or have some defects and flaws which needs to be taken care of.

REFERENCES

- [1] <https://www.geeksforgeeks.org/software-engineering-reverse-engineering/>
- [2] https://en.wikipedia.org/wiki/Reverse_engineering
- [3] <https://www.informit.com/articles/article.aspx?p=23692&seqNum=6>
- [4] <https://www.npd-solutions.com/reverse-engineering.html>
- [5] <https://www.youtube.com>
- [6] <https://searchsoftwarequality.techtarget.com/definition/reverse-engineering>
- [7] <https://dictionary.cambridge.org/dictionary/english/reverse-engineering>
- [8] <https://www.techopedia.com/definition/3868/reverse-engineering>
- [9] https://www.researchgate.net/publication/313212489_A_Software_Reuse_Approach_and_Its_Effect_On_Software_Quality_An_Empirical_Study_for_The_Software_Industry/fulltext/5892a33092851cda256a36db/313212489_A_Software_Reuse_Approach_and_Its_Effect_On_Software_Quality_An_Empirical_Study_for_The_Software_Industry.pdf?origin=publication_detail