Software Re-Engineering

Lecture: 04

Dr. Imran Ali
School of Computing- Software Engineering
FAST-National University of Computer & Emerging Sciences, Karachi

Sequence [Todays Agenda]

Content of Lecture

- Reverse engineering
- Restructuring
- Forward Engineering

Re-Engineering Techniques

- Software re-engineering encompasses several critical techniques designed to address the challenges posed by legacy systems, enhancing their performance, maintainability, and adaptability.
- These techniques include reverse engineering, restructuring, and forward engineering.

Reverse Engineering

- Software Reverse Engineering is a process of recovering the design, requirement specifications, and functions of a product from an analysis of its code.
- It builds a program database and generates information.

Objectives of Reverse Engineering

Reducing Costs: Reverse engineering can help cut costs in product development by finding replacements or cost-effective alternatives for systems or components.

Analysis of Security: Reverse engineering is used in cybersecurity to examine exploits, vulnerabilities, and malware. This helps in understanding of threat mechanisms and the development of practical defenses by security experts.

Integration and Customization: Through the process of reverse engineering, developers can incorporate or modify hardware or software components into pre-existing systems to improve their operation or tailor them to meet particular needs.

Recovering Lost Source Code: Reverse engineering can be used to recover the source code of a software application that has been lost or is inaccessible or at the very least, to produce a higher-level representation of it.

Fixing bugs and maintenance: Reverse engineering can help find and repair flaws or provide updates for systems for which the original source code is either unavailable or inadequately documented.

Reverse Engineering Goals

Cope with Complexity: Reverse engineering is a common tool used to understand and control system complexity. It gives engineers the ability to analyze complex systems and reveal details about their architecture, relationships and design patterns.

Recover lost information: Reverse engineering seeks to retrieve as much information as possible in situations where source code or documentation are lost or unavailable. Rebuilding source code, analyzing data structures and retrieving design details are a few examples of this.

Detect side effects: Understanding a system or component's behavior requires analyzing its side effects. Unintended implications, dependencies, and interactions that might not be obvious from the system's documentation or original source code can be found with the use of reverse engineering.

Facilitate Reuse: Reverse engineering can be used to find reusable parts or modules in systems that already exist. By understanding the functionality and architecture of a system, developers can extract and repurpose components for use in other projects, improving efficiency and decreasing development time.

Steps Involved in Reverse Engineering

Collection Information: This step focuses on collecting all possible information (i.e., source design documents, etc.) about the software.

Examining the Information: The information collected in step-1 is studied so as to get familiar with the system.

Extracting the Structure: This step concerns identifying program structure in the form of a structure chart where each node corresponds to some routine.

Recording the Functionality: During this step processing details of each module of the structure, charts are recorded using structured language like decision table, etc.

Recording Data Flow: From the information extracted in step-3 and step-4, a set of data flow diagrams is derived to show the flow of data among the processes.

Recording Control Flow: The high-level control structure of the software is recorded.

Review Extracted Design: The design document extracted is reviewed several times to ensure consistency and correctness. It also ensures that the design represents the program.

Generate Documentation: Finally, in this step, the complete documentation including SRS, design document, history, overview, etc. is recorded for future use.

Reverse Engineering Tools

Reverse engineering tools accept source code as input and produce a variety of structural, procedural, data, and behavioral design.

Reverse engineering if done manually would consume a lot of time and human labor and hence must be supported by automated tools. Some of the tools are: CIAO and CIA, Rigi, Bunch, GEN++, PBS.

Restructuring in Re-Engineering

- The restructuring phase focuses on redesigning and reorganizing the software system to enhance its performance, maintainability, and scalability. Here are the key aspects of the restructuring phase:

- Code Refactoring: Developers modify the existing codebase to improve its structure, readability, and maintainability. This may involve removing duplicate code, extracting reusable components, and applying design patterns to enhance the system's modularity.

- **Architecture Enhancements:** The system's architecture may be redesigned to address architectural flaws, improve scalability, or incorporate new technologies. This includes reevaluating component interactions, introducing layers or modules, and optimizing system performance.
- **Removal of Obsolete Code:** Outdated or unused code is identified and removed, reducing complexity and improving system performance. This declutters the codebase and makes it easier to understand and maintain.
- The restructuring phase aims to optimize the software system's structure, making it more efficient, flexible, and maintainable. It addresses identified weaknesses and prepares the system for future enhancements.

Forward Engineering

- The forward engineering phase involves using the knowledge gained from reverse engineering and the improvements made during the restructuring phase to develop a new and improved version of the software. Here are key aspects of forward engineering:
  - Implementation of Redesigned Components
  - Introduction of New Features
  - Testing and Quality Assurance


Thank You!