HNDIT 2312- Principles of Software Engineering

Week 11: Software Maintenance

Topics Covered

- Introduction to Software maintenance
- Factors for maintenance
- Types of maintenance
- Cost of maintenance
- Re-engineering

Software Maintenance

- Software maintenance is widely accepted part of SDLC now a days. It stands for all the modifications and updates done after the delivery of software product.

Reasons for maintenance

- Market Conditions
  - Policies, which changes over the time, such as taxation
- Client Requirements
  - Over the time, customer may ask for new features or functions in the software.
- Host Modifications
  - If any of the hardware and/or platform (such as operating system) of the target host changes
- Organization Changes
  - any business level change at client end

Types of maintenance

- Corrective Maintenance
- Adaptive Maintenance
- Perfective Maintenance
- Preventive Maintenance

Corrective maintenance

- This includes modifications and updates done in order to correct or fix problems, which are either discovered by user or concluded by user error reports.

Adaptive Maintenance

- This includes modifications and updates applied to keep the software product up-to date and tuned to the ever changing world of technology and business environment.

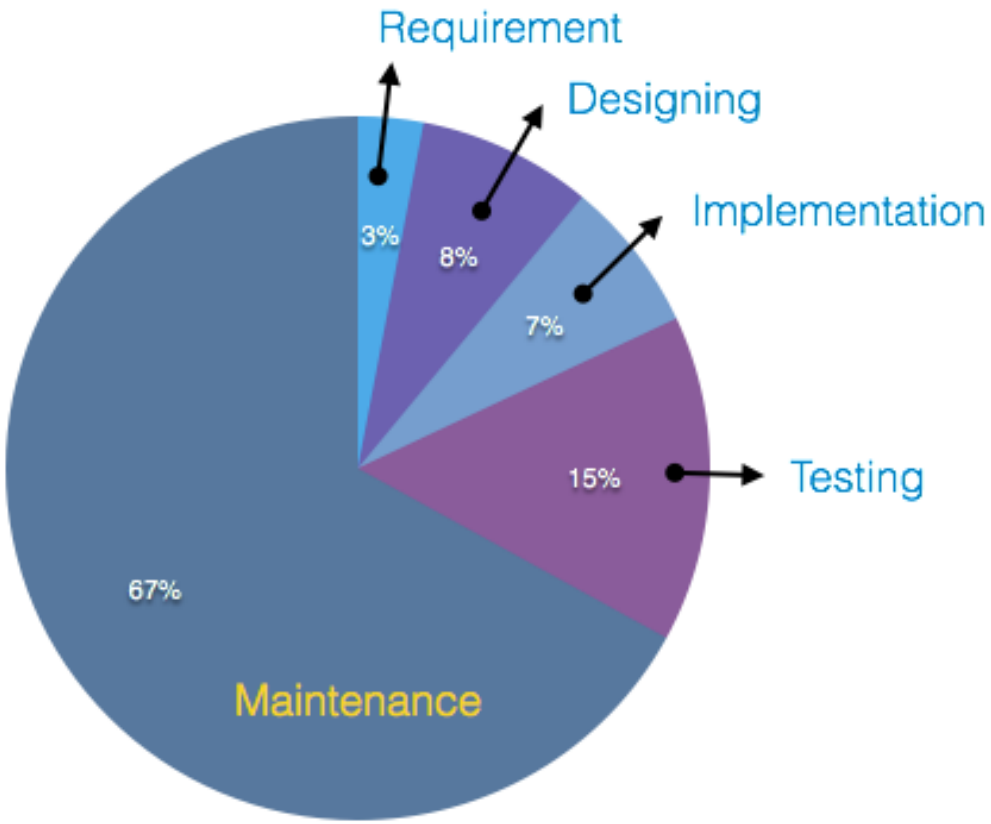
Perfective Maintenance

- This includes modifications and updates done in order to keep the software usable over long period of time.
- It includes new features, new user requirements for refining the software and improve its reliability and performance.

Preventive Maintenance

- This includes modifications and updates to prevent future problems of the software.
- It aims to attend problems, which are not significant at this moment but may cause serious issues in future.

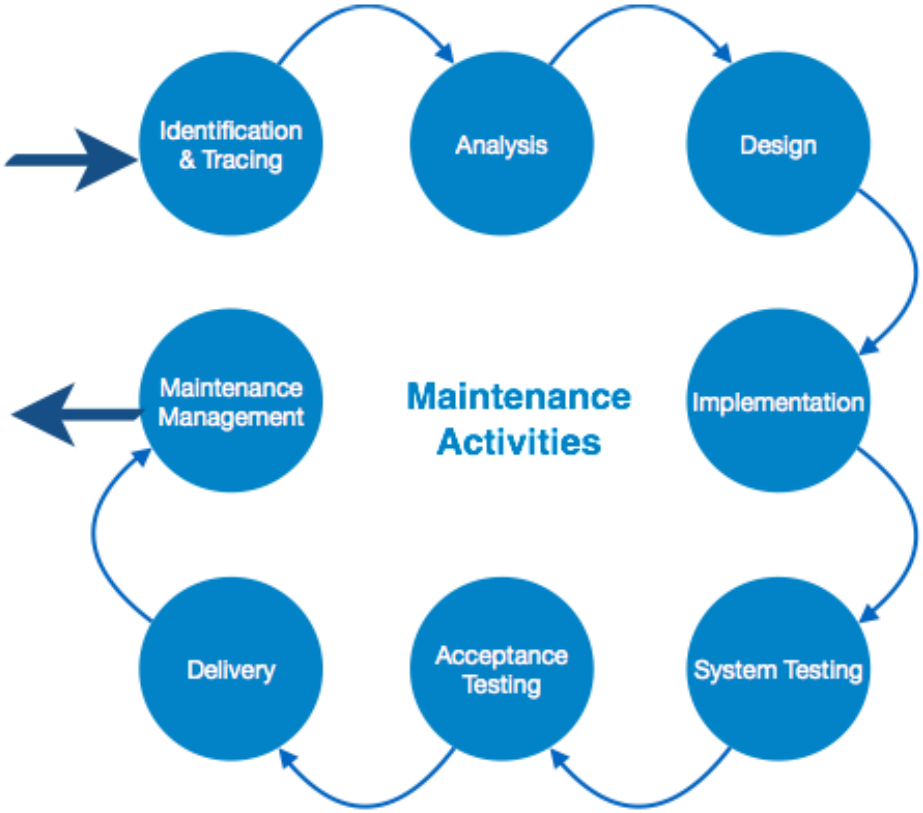
Cost of Maintenance



Maintenance Activities

- IEEE provides a framework for sequential maintenance process activities.
- It can be used in iterative manner and can be extended so that customized items and processes can be included.

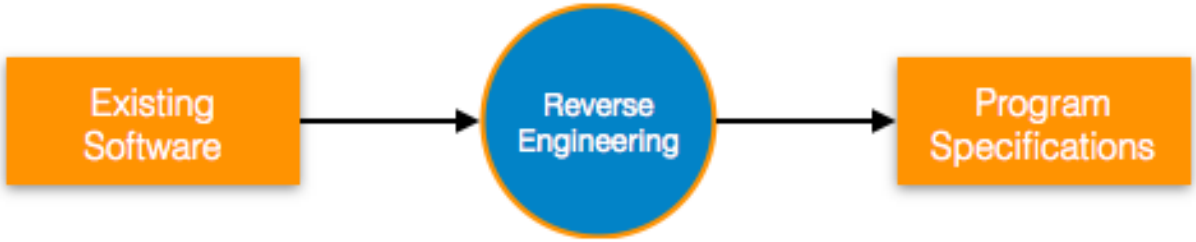
Maintenance Activities



Reverse Engineering

- It is a process to achieve system specification by thoroughly analyzing, understanding the existing system.
- Designers then do reverse engineering by looking at the code and try to get the design.

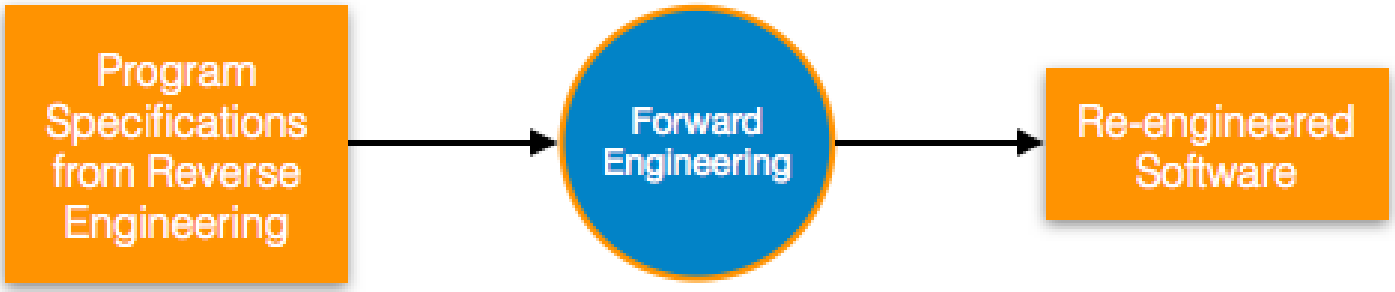
Reverse engineering



Forward Engineering

- Forward engineering is a process of obtaining desired software from the specifications in hand which were brought down by means of reverse engineering.

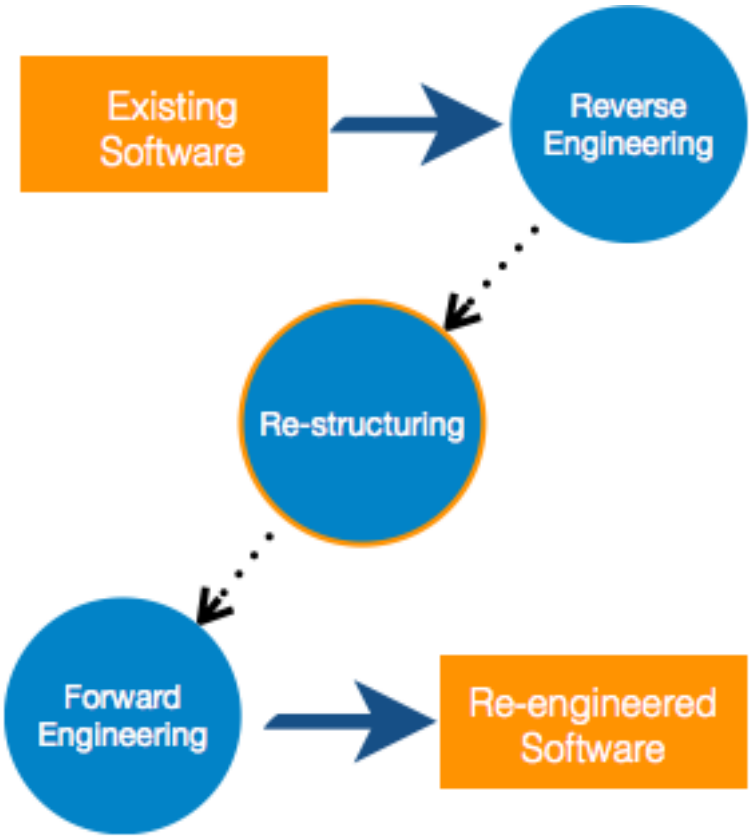
Forward Engineering



Software Re-engineering

- When we need to update the software to keep it to the current market, without impacting its functionality, it is called software re-engineering.
- Legacy software cannot keep tuning with the latest technology available in the market. As the hardware become obsolete, updating of software becomes a headache.

Re-Engineering Process



Program Restructuring

- It is a process to re-structure and re-construct the existing software.
- It is all about re-arranging the source code, either in same programming language or from one programming language to a different one.
- Re-structuring does not impact the functionality of the software but enhance reliability and maintainability.

Maintenance management

- Configuration management is an essential part of system maintenance.
- It is aided with version control tools to control versions, semi-version or patch management.

Key points

- A study on estimating software maintenance found that the cost of maintenance is as high as 67% of the cost of entire software process cycle.