

Software Evolution

LECTURE 1

Software Change

Change is inevitable - to remain useful, software will change after delivery due to: - new requirements, business environment changes, bugs, or poor performance - the majority of the software budget in large companies is devoted to software maintenance or evolution.

Evolution processes

plan driven

- a change request prompts impact analysis, release planning, change implementation, and release
- the change implementation activity may devote time to understanding previous requirements and making the appropriate specification updates
- urgent change requests may omit impact analysis, release planning, and documentation updates

agile

- evolution is implemented just like another iteration in the development process

If different approaches are used by the development and evolution teams, detailed documentation or automated tests and refactoring may be needed.

LECTURE 2

Software Maintenance

The general process of modifying a program after its delivery

Costs

- often greater than development costs
- can be reduced through precise specifications and or configuration management
- factors include varying development teams, poor development, or increasing program structure degradation

maintenance prediction

- allows for planning and cost analysis when change requests arise for a given component
- if a system is tightly coupled with its environment, it is more prone to maintenance
- the more complete a component, the more prone it is to maintenance
- may be reassured in maintenance requests, assessment time, and implementation time

System Re-Engineering

- re-writing components without changing their functionality
- appropriate for improving maintainability or avoiding system replacement
- costs are often lower than alternatives

Refactoring

- a form of preventative maintenance that focuses on structural improvements, reduced complexity and increased understandability
- can reduce long-term maintenance costs
- often occurs during development unlike re-engineering and evolution

Legacy systems

- evolution may involve long-term maintenance, re-engineering, replacement, or elimination