Chapter 5- Implementation and Maintenance

Purpose of System Implementation

- ✓ To convert final physical system specifications into working and reliable software
- ✓ To document work that has been done
- ✓ To provide help for current and future users

Six major activities:

(1) Coding
(2) Testing
(3) Installation
(4) Documentation
(5) Training
(6) Support

The Process of Coding, Testing, and Installation

Coding

Physical design specifications are turned into working computer code.

Testing

Tests are performed using various strategies.
Testing can be performed in parallel with coding.

Installation

The current system is replaced by the new system.

Deliverables for Coding, Testing, and Installation

1. Coding a. Code b. Program documentation 2. Testing a. Test scenarios (test plan) and test data b. Results of program and system testing	3. Installation a. User guides b. User training plan c. Installation and conversion plan i. Software and hardware installation schedule ii. Data conversion plan iii. Site and facility remodeling plan

The Process of Documenting the System, Training Users, and Supporting Users

Documentation

Information systems personnel who will maintain the system throughout its productive life People who will use the system as part of their daily lives

User Training
Application-specific
General: for operating system and off-the-shelf software

Deliverables for Documenting the System, Training, and Supporting Users

1. Documentation
   a. System documentation
   b. User documentation
2. User training plan
   a. Classes
   b. Tutorials
3. User training modules
   a. Training materials
   b. Computer-based training aids
4. User support plan
   a. Help desk
   b. Online help
   c. Bulletin boards and other support mechanisms

Software ApplicationTesting

- ✓ A master test plan is developed during the analysis phase.
- ✓ During the design phase, unit, system and integration test plans are developed.
- ✓ The actual testing is done during implementation.
- ✓ Test plans provide improved communication among all parties involved in testing.

Table of Contents of a Master Test Plan

1. Introduction
   a. Description of system to be tested
   b. Objectives of the test plan
   c. Method of testing
   d. Supporting documents
2. Overall Plan
   a. Milestones, schedule, and locations
   b. Test materials
      1. Test plans
      2. Test cases
      3. Test scenarios
      4. Test log
   c. Criteria for passing tests
3. Testing Requirements
   a. Hardware
   b. Software
   c. Personnel
4. Procedure Control
   a. Test initiation
   b. Test execution
   c. Test failure
   d. Access/change control
   e. Document control
5. Test-Specific or Component-Specific Test Plans
   a. Objectives
   b. Software description
   c. Method
   d. Milestones, schedule, progression, and locations
   e. Requirements
   f. Criteria for passing tests
   g. Resulting test materials
   h. Execution control
   i. Attachments

Test Classification

- ✓ Manual vs. Automated
- ✓ Static vs. Dynamic

A Categorization of Test Types

	Manual	*Automated*
Static	Inspections	Syntax checking
Dynamic	Walkthroughs	Unit test
	Desk checking	Integration test
		System test

Manual Testing Techniques

Inspection

A testing technique in which participants examine program code for predictable language- specific errors

Walkthrough

A peer group review of any product created during the systems development process; also called a structured walkthrough

Desk Checking

A testing technique in which the program code is sequentially executed manually by the reviewer

Steps in a typical walkthrough

GUIDELINES FOR CONDUCTING A CODE WALKTHROUGH

1. Have the review meeting chaired by the project manager or chief programmer, who is also responsible for scheduling the meeting, reserving a room, setting the agenda, inviting participants, and so on.
2. The programmer presents his or her work to the reviewers. Discussion should be general during the presentation.
3. Following the general discussion, the programmer walks through the code in detail, focusing on the logic of the code rather than on specific test cases.
4. Reviewers ask to walk through specific test cases.
5. The chair resolves disagreements if the review team cannot reach agreement among themselves and assigns duties, usually to the programmer, for making specific changes.
6. A second walkthrough is then scheduled if needed.

Automated Testing Techniques

Syntax Checking

The compiler is run against the source code to identify syntax errors.

Unit Testing

Each module is tested alone in an attempt to discover any errors in its code, also called module testing.

Integration Testing

The process of bringing together all of the modules that a program comprises for testing purposes. Modules are typically integrated in a top-down, incremental fashion.

System Testing

The process of bringing together of all the programs that a system comprises for testing purposes. Programs are typically integrated in a top-down, incremental fashion.

Stub Testing

A technique used in testing, especially where modules are written and tested in a top-down fashion, where a few lines of code are used to substitute for subordinate modules.

Test Case Description and Summary Form

Pine Valley Furniture Company
Test Case Description and Summary

Test Case Number: Date:
Test Case Description:

Program/Module Name:
Testing State:
Test Case Prepared By:
Test Administrator:
Description of Test Data:

Expected Results:

Actual Results:

Explanation of Differences between Actual and Expected Results:

Suggestions for Next Steps:

Test Cases

Test case is a scenario of transactions, queries or navigation paths. It can represent either:

- ✓ Typical system use
- ✓ Critical system use
- ✓ Abnormal system use

Test cases and results should be thoroughly documented so they can be repeated for each revision of an application. Test cases are usually developed by analysts. Test cases should not be created by the programmers. Separate people should program and test in order to ensure objectivity. Programmers use symbolic debuggers to isolate causes for errors.

User Acceptance Testing
Actual users test a completed information system. End result is the users' final acceptance of the system.

- ✓ Alpha testing: use simulated data
- ✓ Beta testing: use real data in real user environment

Types of Alpha Tests

- ✓ Recovery testing
  Forces software (or environment) to fail in order to verify that recovery is properly performed

- ✓ Security testing
  Verifies that protection mechanisms built into the system will protect it from improper penetration

- ✓ Stress testing
  Tries to break the system

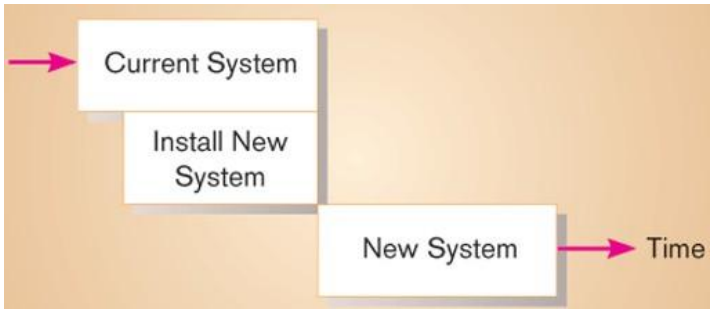
- ✓ Performance testing
  Determines how the system performs on the range of possible environments in which it may be used

Installation

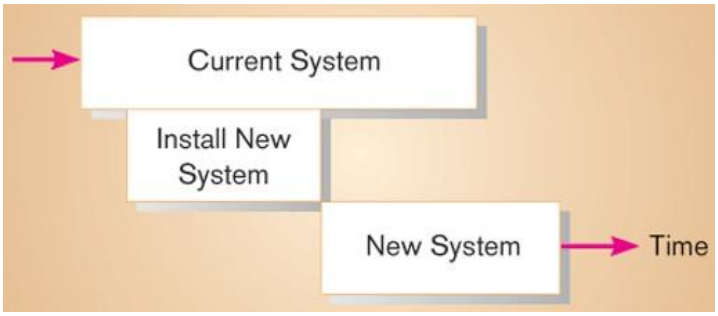
Installation is the organizational process of changing over from the current information system to a new one. Four installation strategies:

- ✓ Direct Installation
- ✓ Parallel Installation
- ✓ Single-location installation
- ✓ Phased Installation

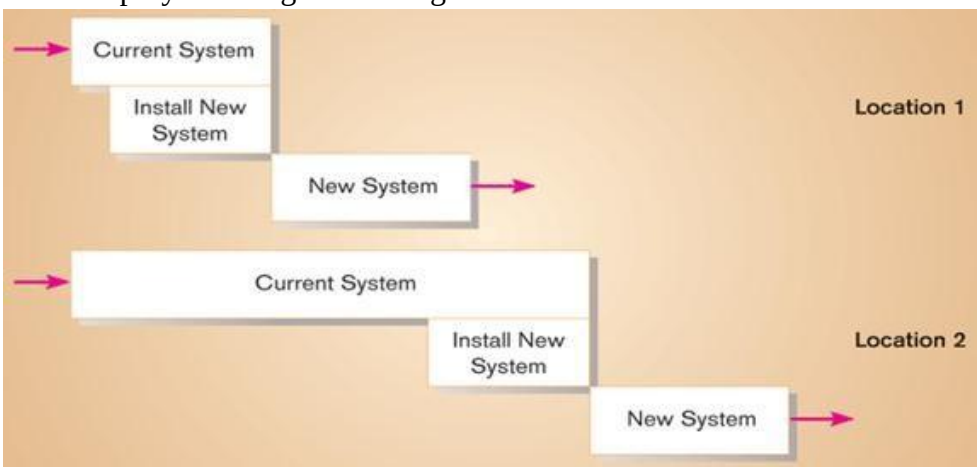
Direct Installation
Changing over from the old information system to a new one by turning off the old system when the new one is turned on



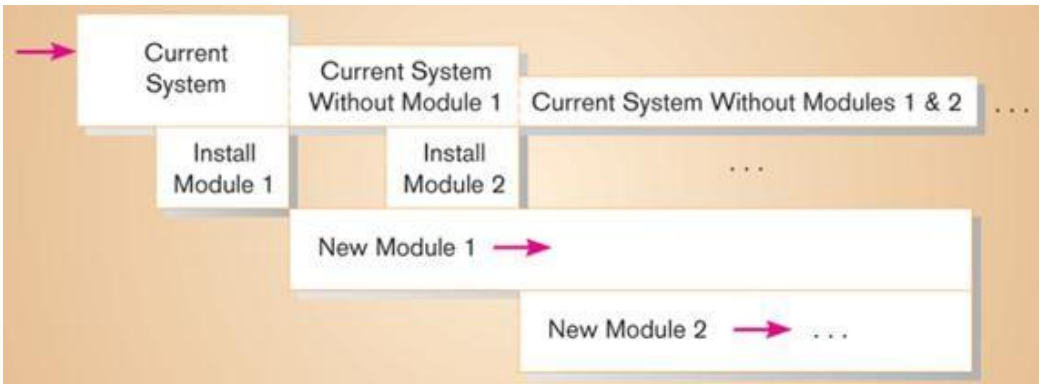
Parallel Installation
Running the old information system and the new one at the same time until management decides the old system can be turned off



Single location installation
Trying out an information system at one site and using the experience to decide if and how the new system should be deployed throughout the organization



Phased Installation
Changing from the old information system to the new one incrementally, starting with one or a few functional components and then gradually extending the installation to cover the whole new system



Planning Installation
Considerations

- ✓ Data conversion
  - Error correction
  - Loading from current system
- ✓ Planned system shutdown
- ✓ Business cycle of organization

Documenting the System

System documentation
It is detailed information about a system's design specifications, its internal workings and its functionality. The intended audience: maintenance programmers.

Internal documentation
System documentation that is part of the program source code or is generated at compile time

External documentation
System documentation that includes the outcome of structured diagramming techniques such as data-flow and entity-relationship diagrams

User Documentation
Written or other visual information about an application system, how it works, and how to use it

Preparing user documentation

- ✓ Traditional source has been information systems department
- ✓ Application-oriented documentation is now often supplied by vendors and users themselves

Outline of a Generic User's Guide

Preface
1. Introduction
   1.1 Configurations
   1.2 Function flow
2. User interface
   2.1 Display screens
   2.2 Command types
3. Getting started
   3.1 Login
   3.2 Logout
   3.3 Save
   3.4 Error recovery
   3.n [Basic procedure name]
n. [Task name]
Appendix A—Error Messages
([Appendix])
Glossary
   Terms
   Acronyms
Index

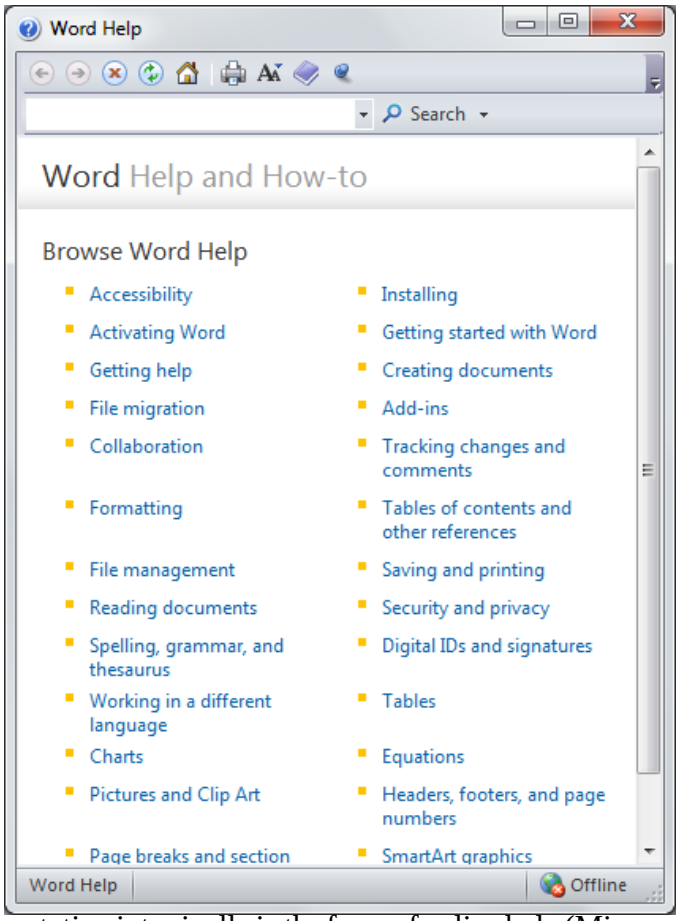


Figure: User documentation is typically in the form of online help (Microsoft Word 2010)

Training Information Systems Users

Potential training topics

- ✓ Use of the system
- ✓ General computer concepts
- ✓ Information system concepts
- ✓ Organizational concepts
- ✓ System management
- ✓ System installation

1. Tutorial—one person taught at a time
2. Course—several people taught at a time
3. Computer-aided instruction
4. Interactive training manuals—combination of tutorials and computer-aided instruction
5. Resident expert
6. Software help components
7. External sources, such as vendors

Figure: Seven common methods for computer training

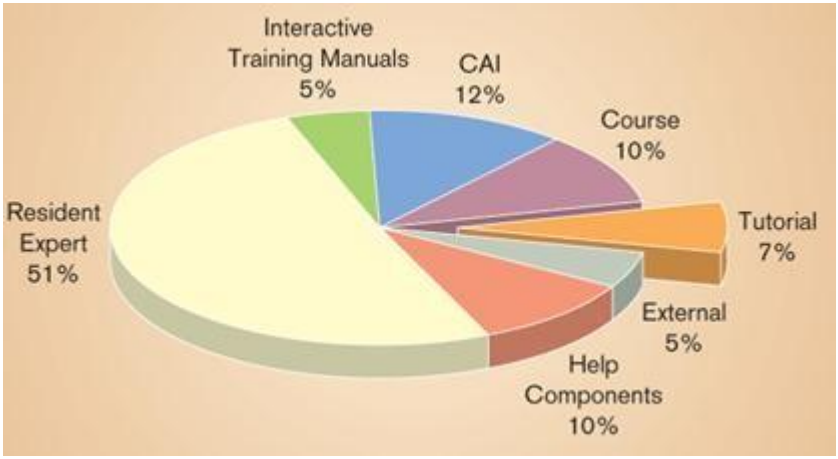
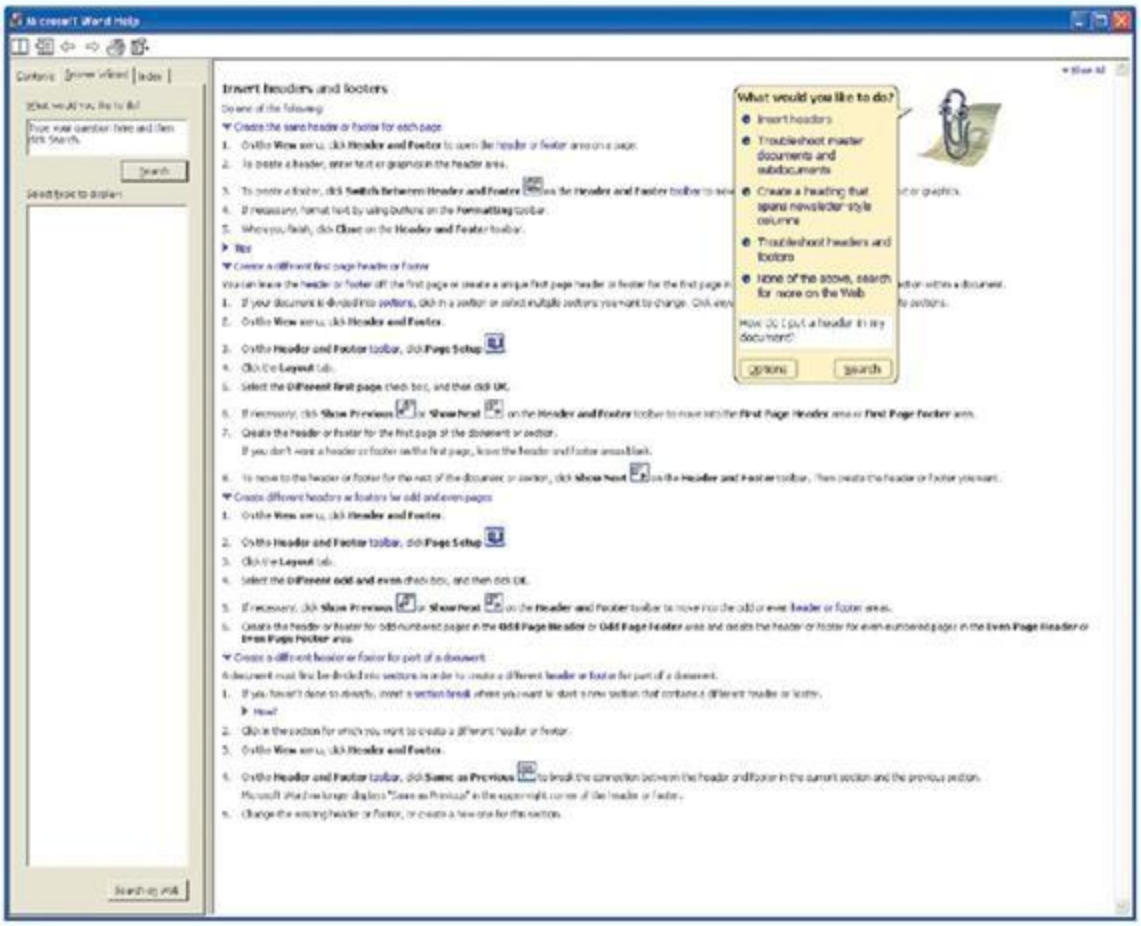


Figure: Frequency of use of computer training methods

By far the most common training method is informal, via interaction with an in-house expert on the software.

A Microsoft Office Assistant™ note



Electronic Performance Support Systems (EPSS), like Microsoft Office Assistant, are components of software applications that embed training and information for the user, in the form of tutorials, expert systems, and hyperlink jumps to reference topics.

Supporting Information Systems Users

- ✓ Support is extremely important to users
- ✓ Providing support can be expensive and time-consuming
- ✓ One approach is through automation
    - Internet-based online supportforums
    - On-demand fax
    - Voice response systems
    - Knowledge bases

Providing Support via Information Center

Information center is an organizational unit whose mission is to support users in exploiting information technology. Staff might perform the following tasks:

- ✓ Install new hardware or software and set up user accounts
- ✓ Consult with users writing programs in fourth-generation languages

- ✓ Extract data from organizational databases onto personal computers
- ✓ Answer basic on-demand questions
- ✓ Provide a demonstration site for viewing hardware and software
- ✓ Work with users to submit system change requests

Providing Support via Help Desk

Help desk is a single point of contact for all user inquiries and problems about a particular information system or for all users in a particular department. It requires:

Technical skills: extensive knowledge about how to use the system and typical problems that can be encountered

People skills: good listening and communication, dealing with complaints and frustrations Support Issues:

- ✓ User questions and problems
- ✓ Recovery and backup
- ✓ Disaster recovery
- ✓ PC maintenance
- ✓ Writing newsletters
- ✓ Setting up user groups

Why Implementation Sometimes Fails?

Two conditions necessary for a successful implementation

- ✓ Management support of the system under development
- ✓ Involvement of users in the development process

Insights about implementation process

- ✓ Risk
- ✓ Commitment to the project
- ✓ Commitment to change
- ✓ Extent of project definition and planning
- ✓ Realistic user expectations

Implementation Success Factors

Biggest measure of success: will it be used? Major factors influencing likelihood of use are:

- ✓ Personal stake of users
- ✓ System characteristics
- ✓ User demographics , such as age and degree of computer experience
- ✓ Organizational support
- ✓ Performance
- ✓ User's satisfaction with system
- ✓ System ease of use and reliability
- ✓ Extent to which system is used

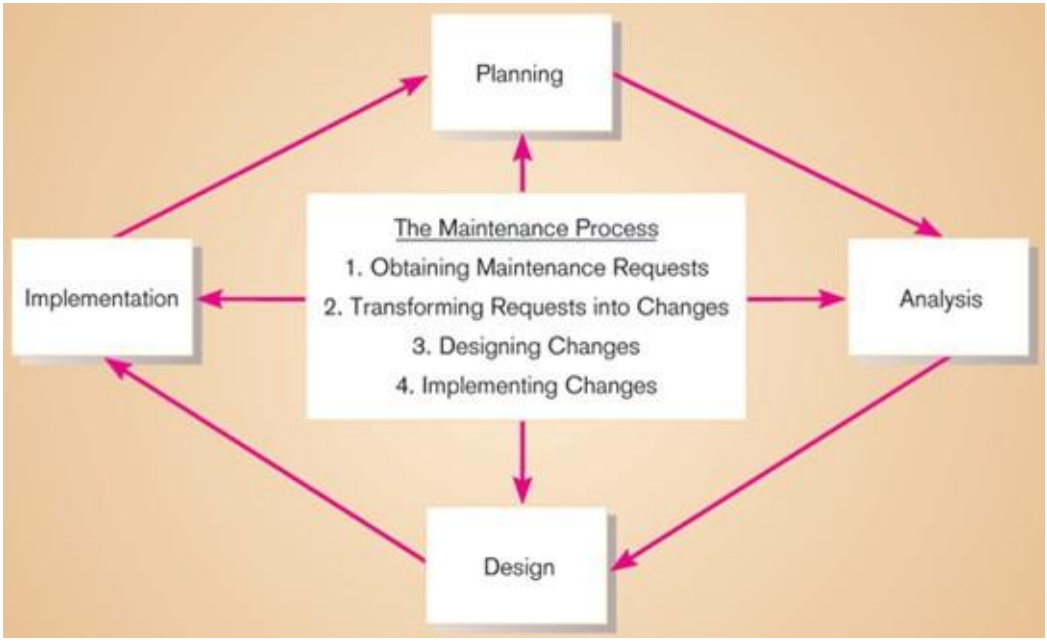
Project Close-Down

- ✓ Evaluate team
  - Reassign members to other projects
- ✓ Notify all affected parties that the development project is ending and that you are switching to operation and maintenance mode
- ✓ Conduct post project reviews
- ✓ Close out customer contract
  - Formal signoff

The Process of Maintaining Information Systems

It is the process of returning to the beginning of the SDLC and repeating development steps focusing on system change until the change is implemented. Maintenance is the longest phase in the SDLC. Maintenance is like a mini-SDLC. Four major activities are involved in maintenance:

(1) Obtaining maintenance requests
(2) Transforming requests into changes
(3) Designing changes
(4) Implementing changes



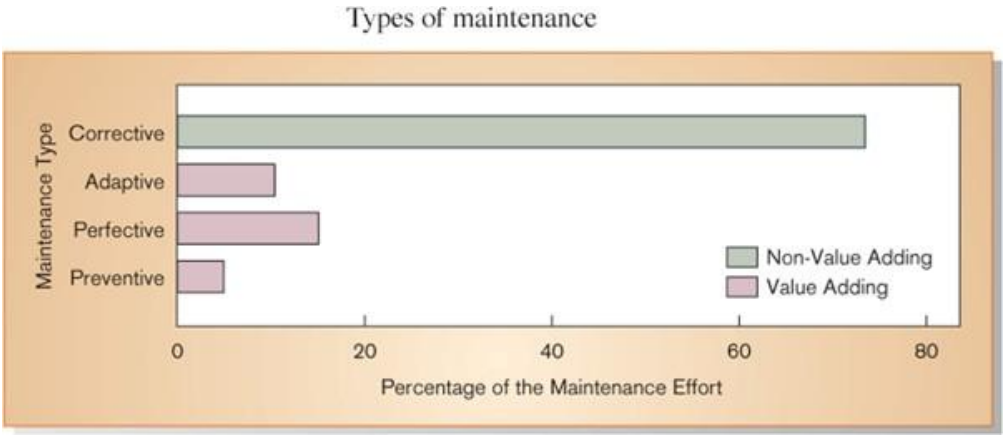
Types of System Maintenance

Corrective maintenance
Changes made to a system to repair flaws in its design, coding, or implementation

Adaptive maintenance
Changes made to a system to evolve its functionality to changing business needs or technologies

Perfective maintenance
Changes made to a system to add new features or to improve performance

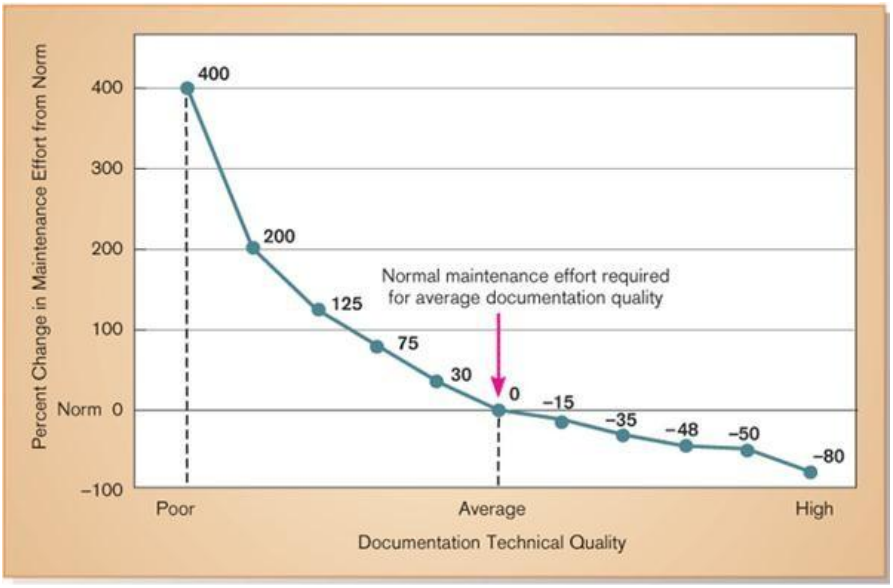
Preventive maintenance
Changes made to a system to avoid possible future problems



By far, most maintenance is corrective, and therefore urgent and non-value adding.

The Cost of Maintenance
Many organizations allocate 80% of information systems budget to maintenance. The factors that influence system maintainability are:

- ✓ Latent defects
- ✓ Number of customers for a given system
- ✓ Quality of system documentation
- ✓ Maintenance personnel
- ✓ Tools
- ✓ Well-structured programs



A well-documented system is easier to understand, and therefore easier to maintain.

Conducting SystemMaintenance

Managing Maintenance

Number of people working in maintenance has surpassed number working in development. Three possible organizational structures exist:

Separate
Maintenance group consists of different personnel than development group.

Combined
Developers also maintain systems.

Functional
Maintenance personnel work within the functional business unit.

Advantages and Disadvantages

Maintenance Organization Type	Advantages	Disadvantages
Separate	Improved system and documentation quality	Ignorance of critical undocumented information
Combined	Maintenance group knows all about system	Less emphasis on good documentation
Functional	Personnel have vested interest	Limited job mobility and human or technical resources

Assignment of Maintenance Personnel

- ✓ Maintenance work is often viewed negatively by IS personnel.
- ✓ Organizations have historically have rewarded people involved in new development better than maintenance personnel.
- ✓ Organizations often rotate personnel in and out of maintenance roles in order to lessen negative feelings about maintenance.

Measures of Effectiveness

Important factors to consider

- ✓ Number of failures
- ✓ Time between each failure
- ✓ Type of failure

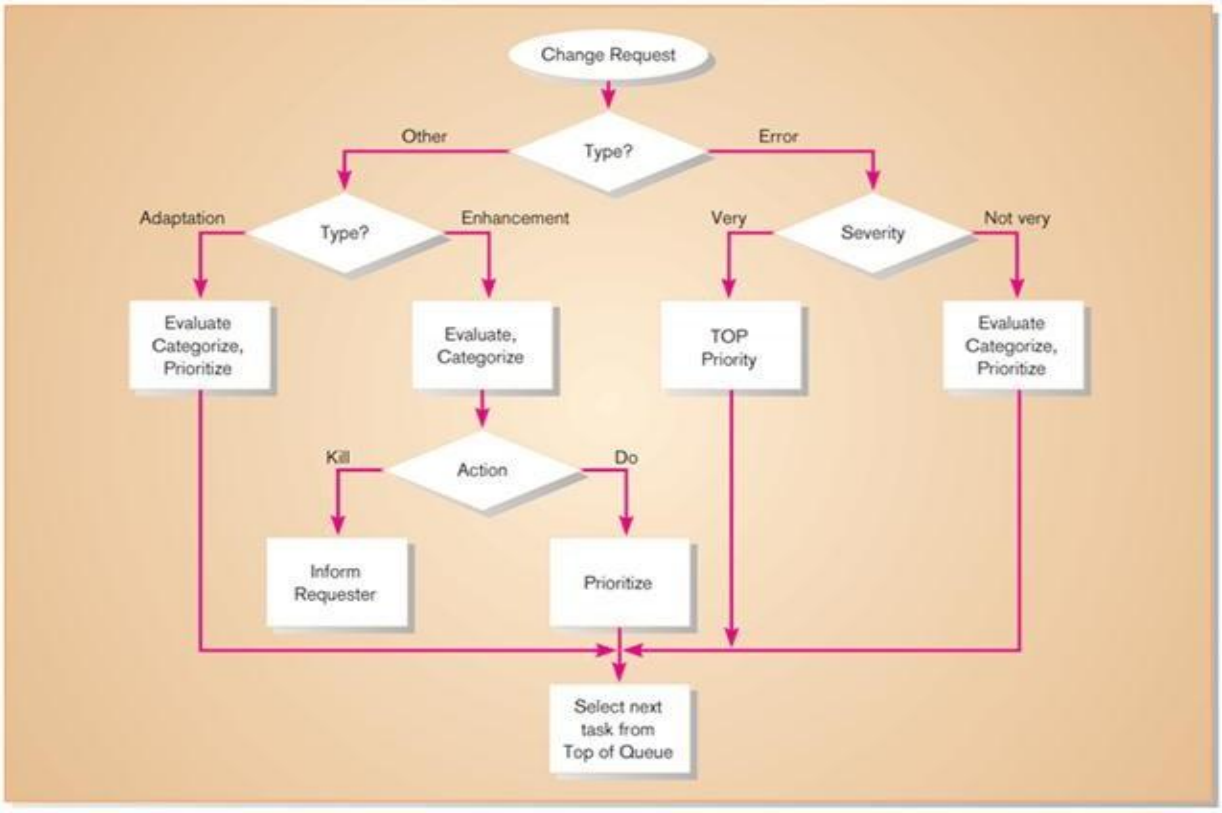
Mean time between failures (MTBF)
A measurement of error occurrences that can be tracked over time to indicate the quality of a system

How the mean time between failures should change over time

Controlling Maintenance Requests

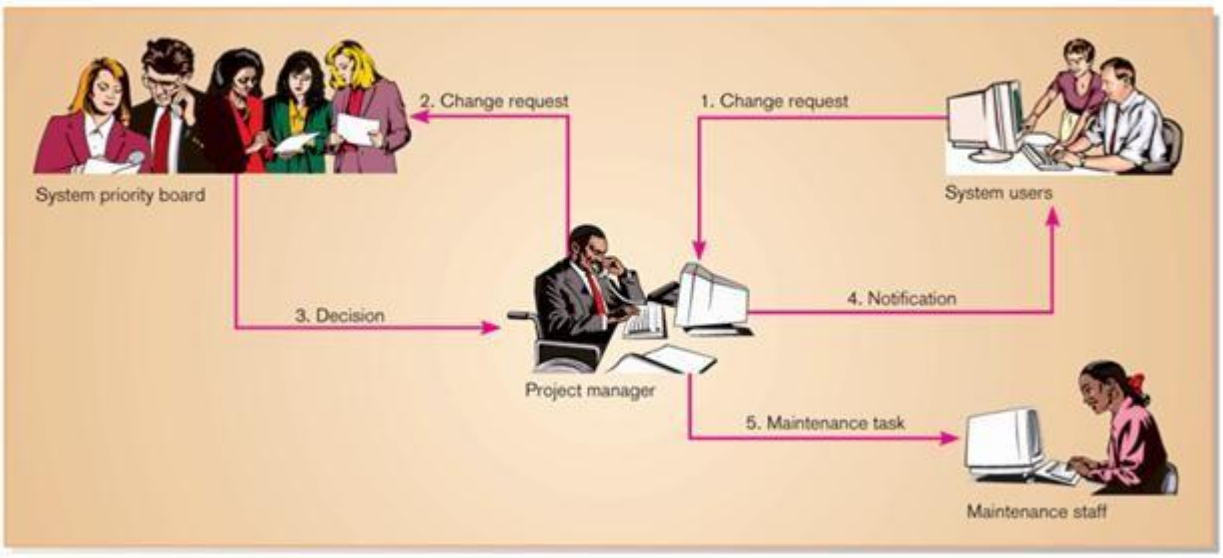
- ✓ Maintenance requests can be frequent
- ✓ Prioritize based on type and urgency of request

Flowchart of how to control maintenance requests



The Flow of a Maintenance Request

How a maintenance request moves through an organization



Configuration Management

It is the process of assuring that only authorized changes are made to the system.

- ✓ *Baseline modules*
  Software modules that have been tested, documented, and approved to be included in the most recently created version of a system

- ✓ *System librarian*
  A person responsible for controlling the checking out and checking in of baseline modules when a system is being developed or maintained

- ✓ *Build routines*
  Guidelines that list the instructions to construct an executable system from the baseline source code

Configuration Management Tools

They are the special software systems for managing system configuration. They control access to modules in the system library. Historical changes can be traced and previous versions can be reconstructed

Two types:
Revision control: modules labeled as frozen (unchangeable) or floating (checked out by programmer for modification)

Source code control: extend revision control to all interrelated modules

References:

- ✓ Hoffer, J.A., George, J.F. and Valacich J.S., "Modern Systems Analysis and Design", 3rd Edition, Pearson Education, 2003.
- ✓ K.E.Kendall and J.E.Kendall, "Systems Analysis and Design", 5th Edition, Pearson Eduation, 2003.
- ✓ V.Rajaraman, "Analysis and Design of Information Systems", 2nd Edition, Prentice Hall of India, New Delhi, 2002
- ✓ E.Yourdon "Modern Structured Analysis", Prentice Hall of India, 1996.
- ✓ Elias M Awad, "Systems Analysis and Design", Galgotia.
- ✓ Igor Hawryszkiewycz, "Systems Analysis and Design", PHI.
- ✓ R.Schultheis and Mary Summer, "Management Information Systems", Tata McGraw Hill, 1999.

Assignments:

(1) What do you understand by installation? Discuss the different approaches to installation.
(2) What are the different types of maintenance and how do they differ? What are the factors that influence the cost of maintenance?
(3) Explain the four different approaches to installation? Which is the most expensive? Which is the most risky?
(4) "Documentation starts right from the beginning of system development process." Is it true? Justify your answer with proper reasons.

A Gentle Advice:

Please go through your text books and reference books for detail study!!! Thank you all.