SCSD2613
System Analysis and Design

PART VI
Systems Implementation

www.utm.my
innovative • entrepreneurial • global

univteknologimalaysia utm_my utmofficial

update: August 2019 (sharinhh)

OBJECTIVES

- Describe the process of coding, testing, and converting an organizational information system and outline the deliverables and outcomes of the process.
- Apply four installation strategies: direct, parallel, single-location, and phased installation.
- List the deliverables for documenting the system and for training and supporting users.
- Explain why system implementation sometimes fails.
- Describe the threats to system security and remedies that can be applied.

MAJOR TOPICS

SYSTEM IMPLEMENTATION	• System implementation process
SYSTEM CONSTRUCTION (CODING) & TESTING	• System construction • Types of system testing
INSTALLATION	• Types of system installation • Planning the installation
DOCUMENTING THE SYSTEM	• Generic guidelines
TRAINING AND SUPPORT	• Types of training • Support
ORGANIZATIONAL ISSUES	• Factors • Security issues • Project closure

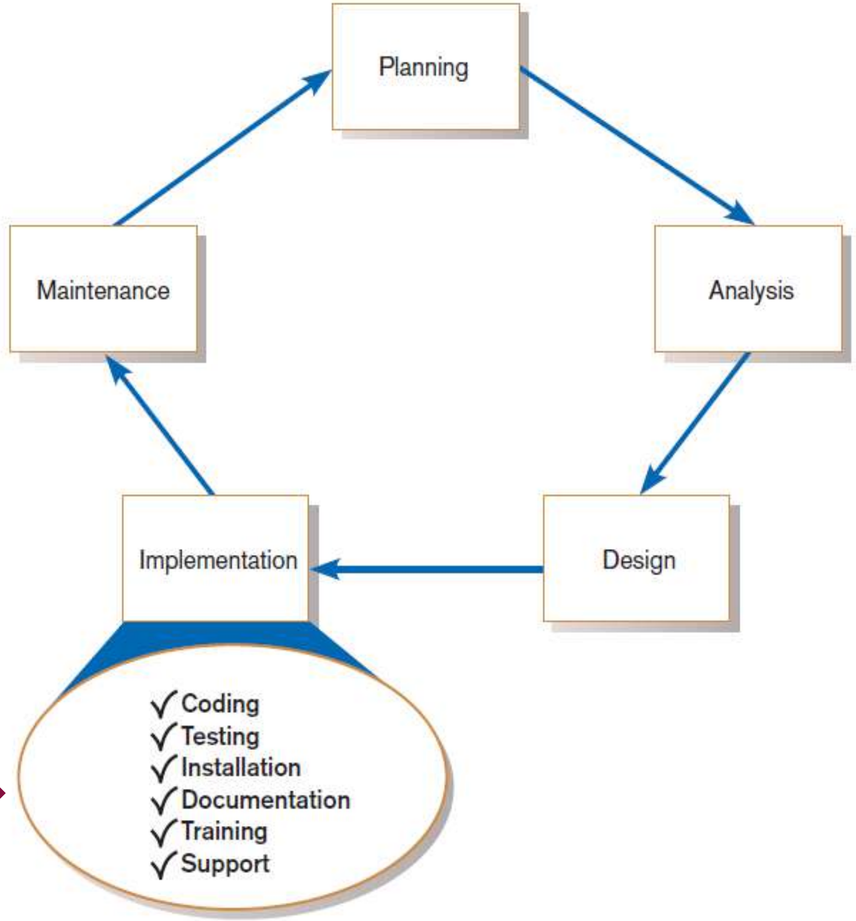
MAJOR TOPICS

SYSTEM IMPLEMENTATION

- System implementation process

RECAP: SDLC

Systems development life cycle with the implementation phase highlighted

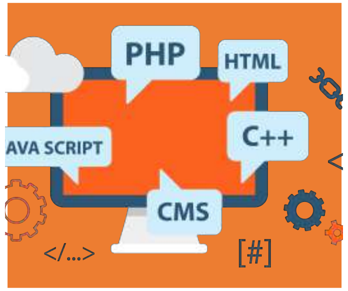


SYSTEM IMPLEMENTATION

- Purpose:
  - 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

SYSTEM IMPLEMENTATION – MAJOR ACTIVITIES

CODING

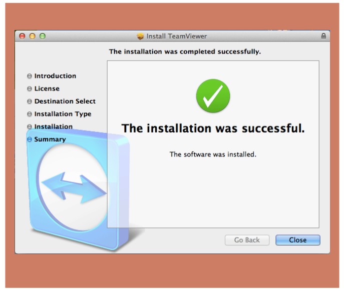


Physical design specifications are turned into working computer code.

TESTING

- Tests are performed using various strategies.
- Testing is performed in parallel with coding.

INSTALLATION



Install TeamViewer

The installation was completed successfully.

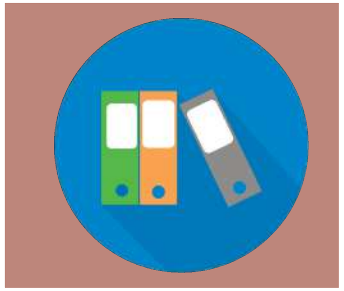
- Introduction
- License
- Destination Select
- Installation Type
- Installation
- Summary

The installation was successful.

The software was installed.

The current system is replaced by a new system.

DOCUMENTATION



Two audiences for final 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 (**system user**)

TRAINING

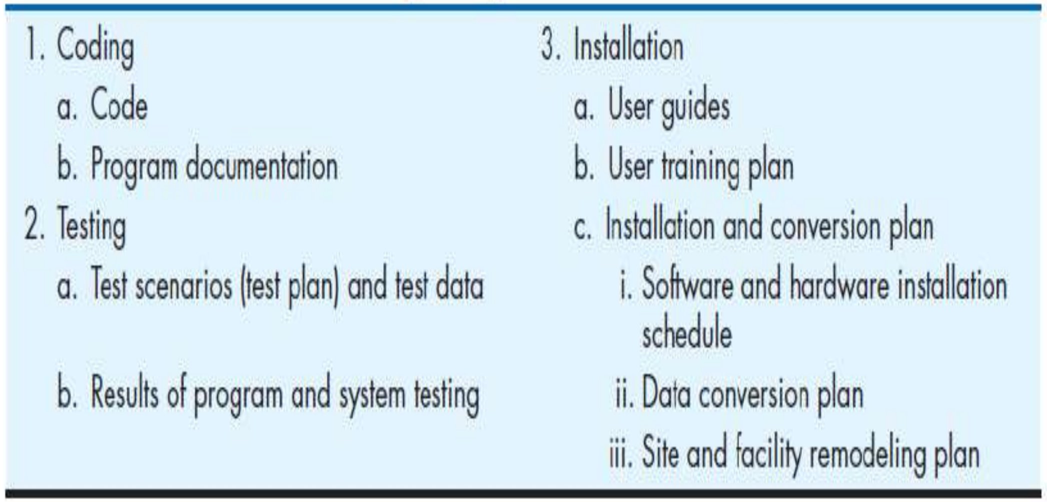
- Application-specific
- General for operating system and off-the-shelf software

SUPPORT

- Automating support (chatbot)
- Help desk

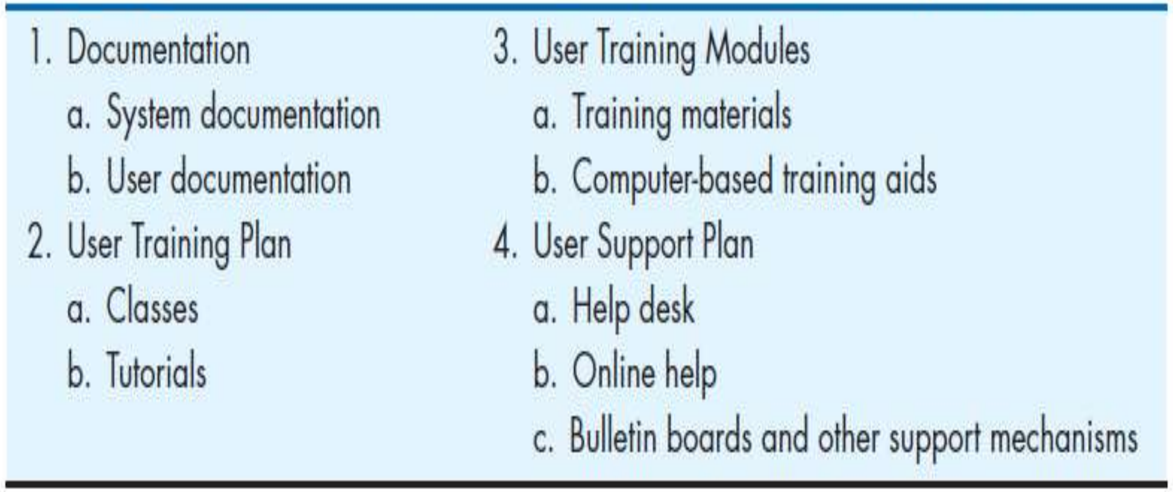
SYSTEM IMPLEMENTATION DELIVERABLES

TABLE 13-1 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

TABLE 13-2 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

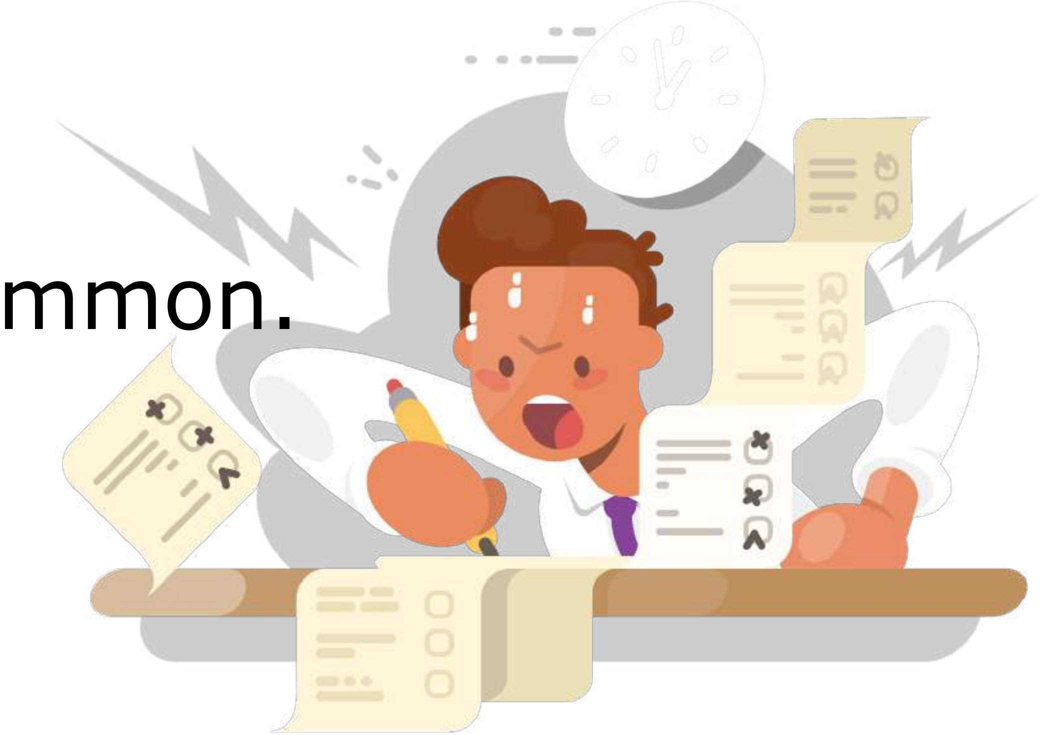
MAJOR TOPICS

SYSTEM CONSTRUCTION (CODING) & TESTING

- System construction
- Types of system testing

SYSTEM CONSTRUCTION AND TESTING

- Labor intensive activity.
- Tools are required.
- Blueprints are essential.
- Mistakes can be deadly.
- Weariness and Stress are common.



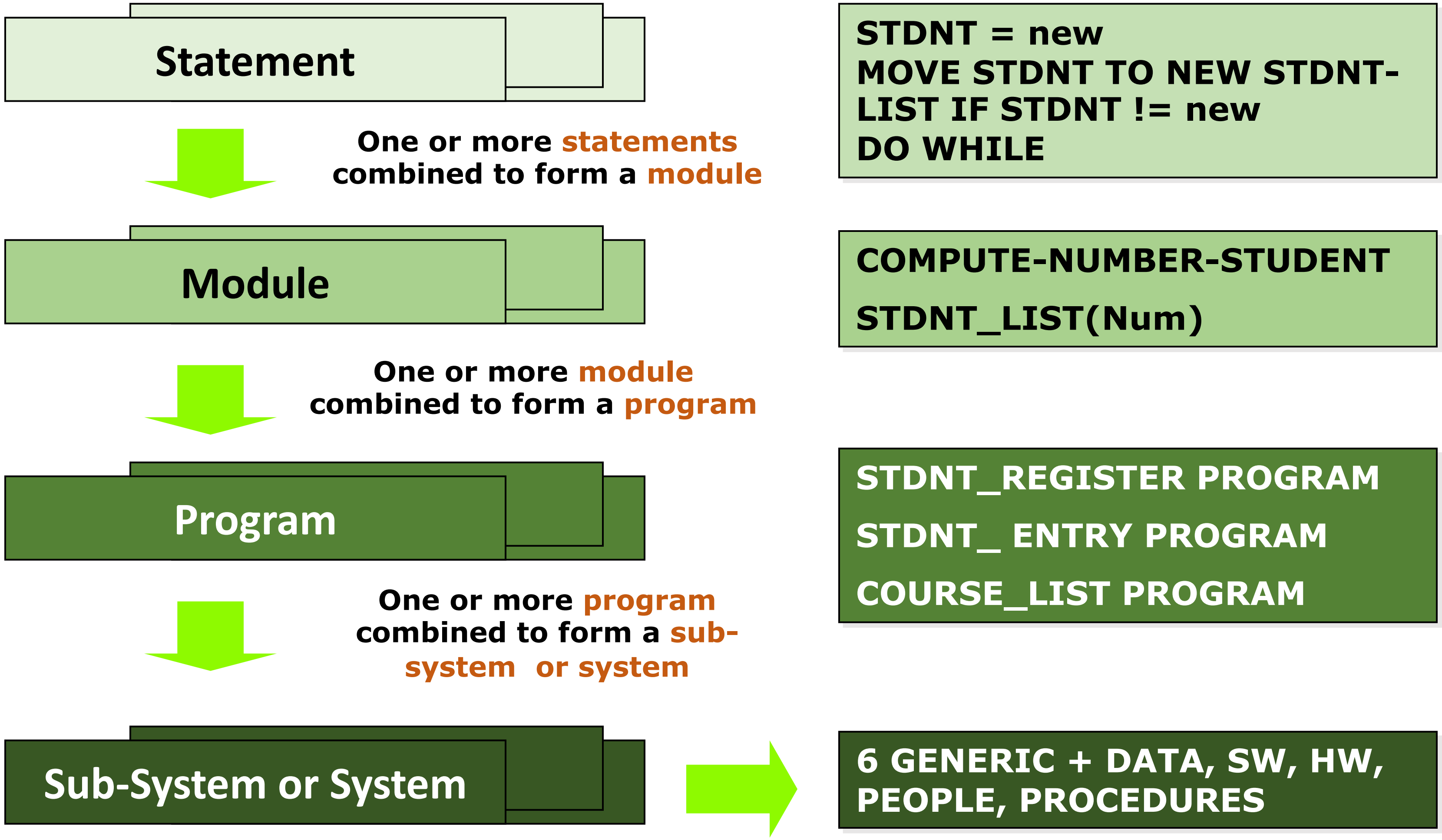
SYSTEM CONSTRUCTION

- System construction is the development of all parts of the system, including the software itself, documentation, and new operating procedures.
- **Programming** is often seen as the focal point of system development.
- Programming and testing are **tightly coupled.**

SYSTEM TESTING

- System testing is the process to ensure that application programs written and tested in isolation properly when they are integrated into the total system.
- Testing starts with the tester's developing a **test plan** that defines a series of tests that will be conducted.

SYSTEM TESTING HIERARCHICAL VIEW



Statement

One or more statements combined to form a module

Module

One or more module combined to form a program

Program

One or more program combined to form a sub-system or system

Sub-System or System

STDNT = new
MOVE STDNT TO NEW STDNT-LIST IF STDNT != new
DO WHILE

COMPUTE-NUMBER-STUDENT

STDNT_LIST(Num)

STDNT_REGISTER PROGRAM

STDNT_ ENTRY PROGRAM

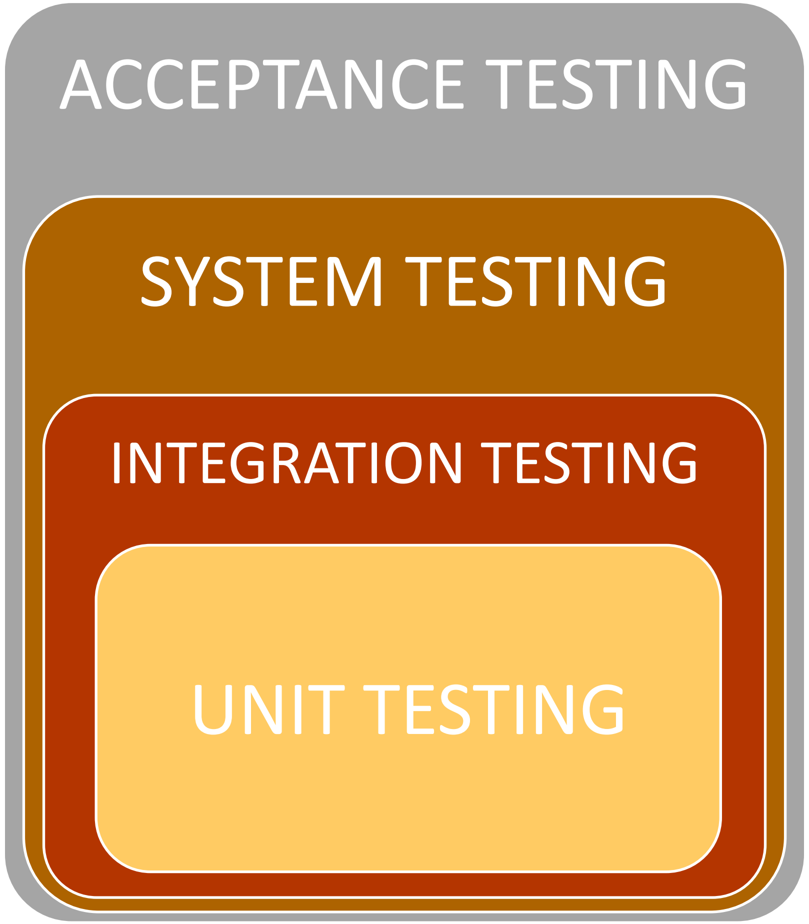
COURSE_LIST PROGRAM

6 GENERIC + DATA, SW, HW, PEOPLE, PROCEDURES

SYSTEM TESTING PRINCIPLES

- User-accepted test plan.
- Testing strategies
  - Top-down
  - Bottom-up
  - Middle-out
  - Hybrid
  - Black box
  - White box
  - Alpha
  - Beta

- Generally, there are 4 testing stages:
  - Unit Testing.
  - Integration Testing.
  - System Testing.
  - Acceptance Testing.



SYSTEM TESTING STAGES (1)

Testing Stage	Type of Test Strategies	Notes
Unit Testing	Black-box, White-box	Focus on whether the unit meets the req stated in the **program spec.**
Integration Testing	User interface testing	Testing is done by moving through each and every menu item in the interface either in a top-down or bottom-up.
	Use scenario testing	Testing is done by moving through each **use scenario** to ensure it works correctly. Use scenario testing is usually combined with user interface testing because it does not test all interfaces.
	Data flow testing	Testing is done when involve data processing. **Physical DFDs** as primary resources.
	System interface testing	When the system exhanges data with other system. Also, **physical DFDs** as primary resources.

SYSTEM TESTING STAGES (2)

Testing Stage	Type of Test Strategies	Notes
System Testing	Requirement testing	Test the system whether it fulfill business needs.
	Usability testing	Test the system whether it convenient. **GUI design** as primary resources.
	Security testing	To test disaster recovery and invalid access. **Architecture design** as primary resources.
	Performance testing	To test system robust. (e.g system overloading, etc)
	Documentation testing	To test whether the documentation is written correctly(e.g procedure, tutorial, etc

SOFTWARE APPLICATION TESTING

- 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.
- Written test plans provide improved communication among all parties involved in testing.
- **Static or dynamic techniques**
    - Static testing means that the code being tested is not executed.
    - Dynamic testing involves execution of the code.
- **Test is automated or manual**
    - Automated means computer conducts the test.
    - Manual means that people complete the test.

SOFTWARE APPLICATION TESTING

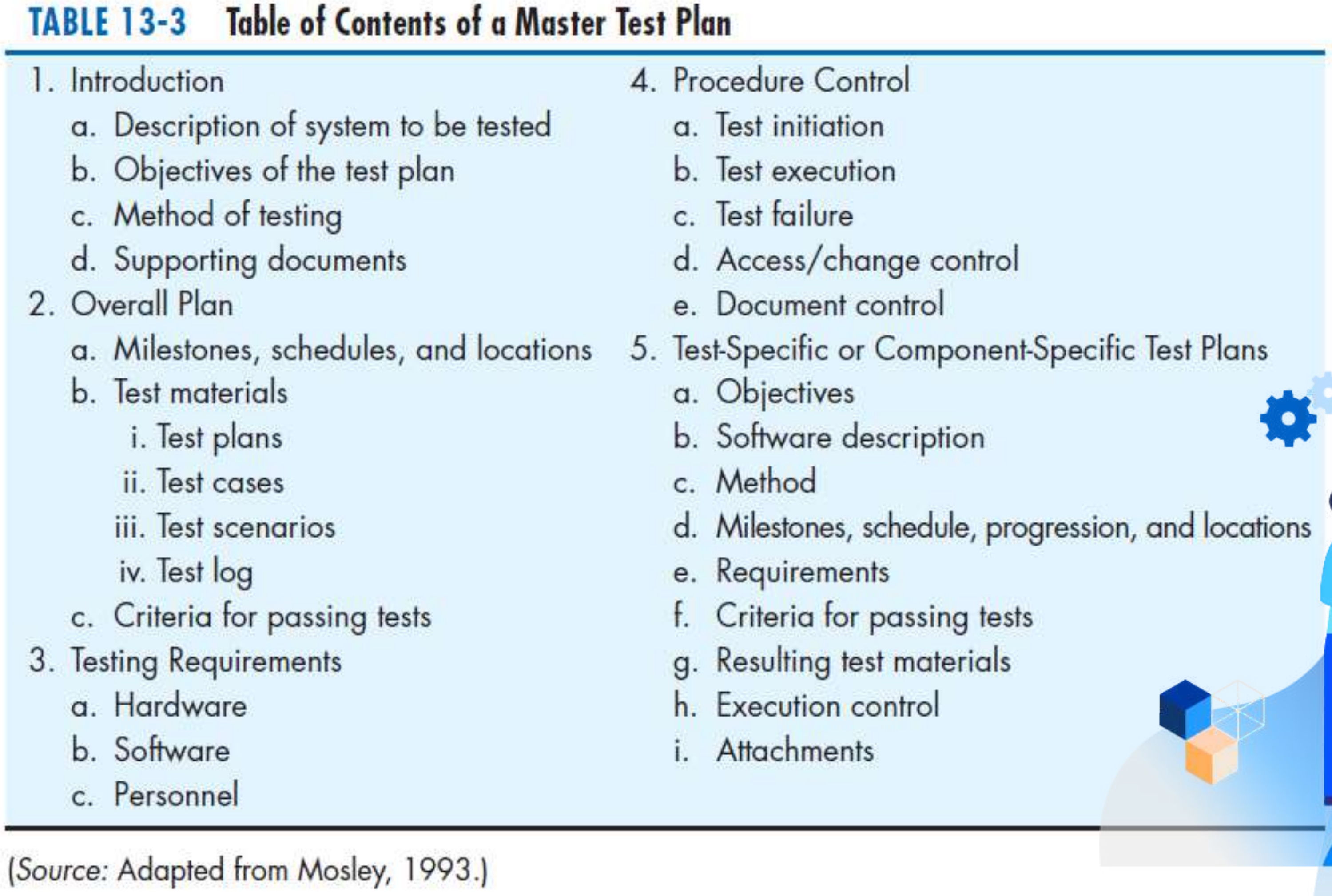
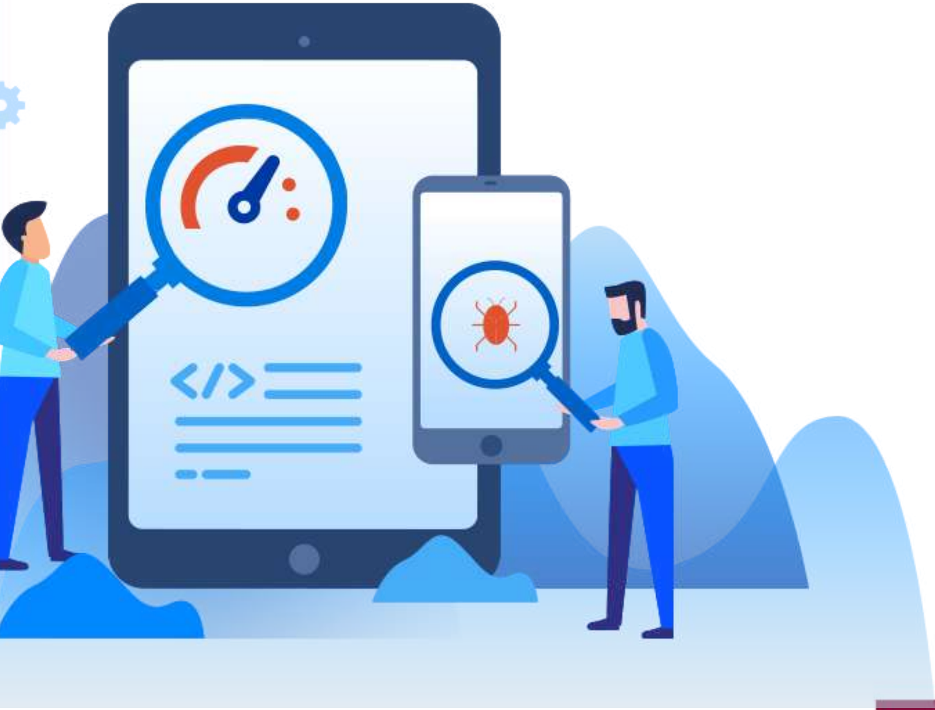


TABLE 13-3 **Table of Contents of a Master Test Plan**

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, schedules, and locations
   b. Test materials
      i. Test plans
      ii. Test cases
      iii. Test scenarios
      iv. 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

(*Source:* Adapted from Mosley, 1993.)



TYPES OF TEST

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, including code

DESK CHECKING

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

UNIT TESTING

- each module is tested alone in an attempt to discover any errors in its code

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 bringing together of all of 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 modules, 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

ACCEPTANCE TESTING BY USERS

- **Acceptance testing**: the process whereby actual users test a completed information system, the end result of which is the users' acceptance of it.
- Two types of acceptance test:

ALPHA TESTING

- user testing of a completed information system using simulated data
- Types of Alpha Testing:
  1. ***Recovery testing*** — forces software (or environment) to fail in order to verify that recovery is properly performed
  2. ***Security testing*** — verifies that protection mechanisms built into the system will protect it from improper penetration
  3. ***Stress testing*** — tries to break the system
  4. ***Performance testing*** — determines how the system performs on the range of possible environments in which it may be used

BETA TESTING

- user testing of a completed information system using real data in the real user environment
- A subset of intended users runs the system in the user's own environments using their own data.
- The intent is to determine whether the software, documentation, technical support and training activities work as planned.
- As rehearsal of the installation phase.

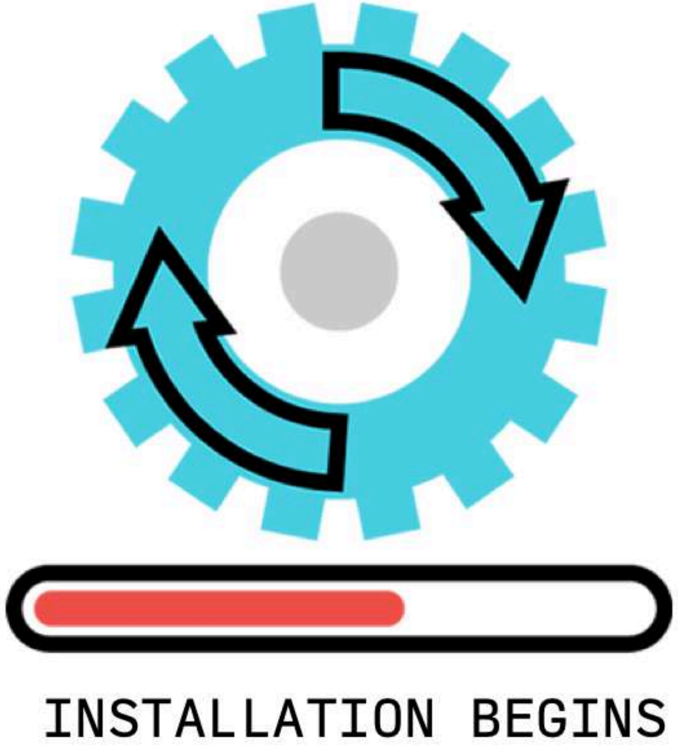
MAJOR TOPICS

INSTALLATION

- Types of system installation
- Planning the installation

INSTALLATION

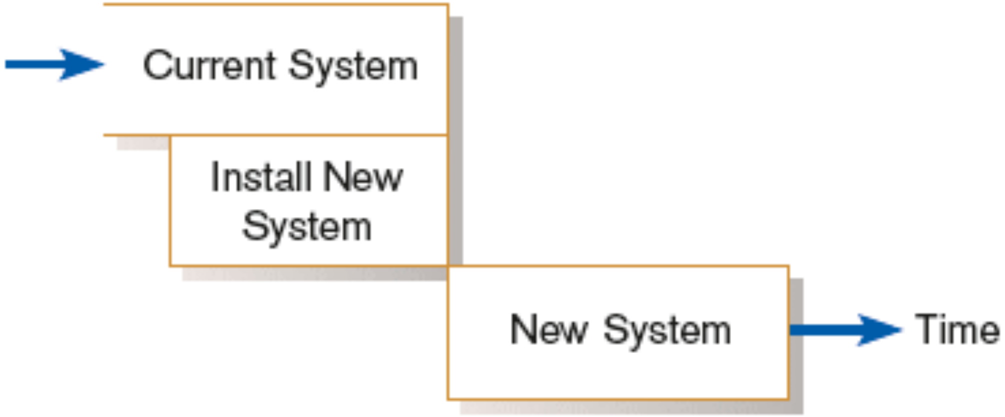
- Installation: the organizational process of changing over from the current information system to a new one.
- It can be involved three important steps:
  - Software installation.
  - Hardware installation.
  - Data conversion.
- Four installation strategies:
  - Direct Installation
  - Parallel Installation
  - Single-location installation
  - Phased Installation



INSTALLATION STRATEGIES

DIRECT INSTALLATION

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



PROS

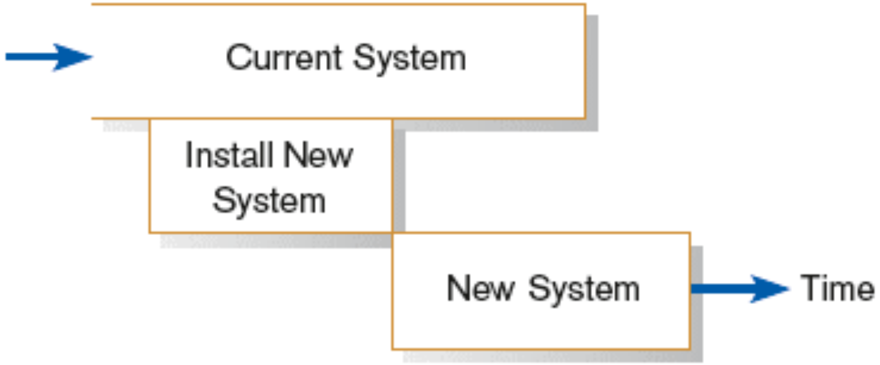
- No duplication of effort for the users.
- No transition costs.
- Learning advantage.
- Immediately benefits.

CONS

- High risk.
- Cannot compare results with the old.
- Sense of user insecurity

PARALEL INSTALLATION

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



PROS

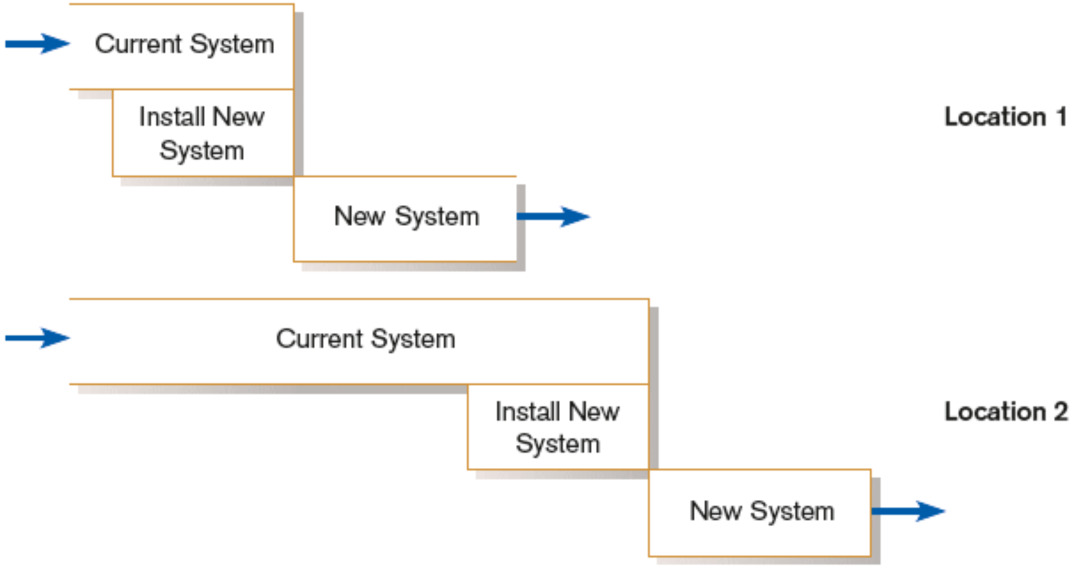
- Low risk.
- Sense of user insecurity.
- Ability to compare results with the old

CONS

- Duplication of effort for the users
- Transition costs.
- Additional computer processing.

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
- Also known as location or pilot installation



✓ PROS

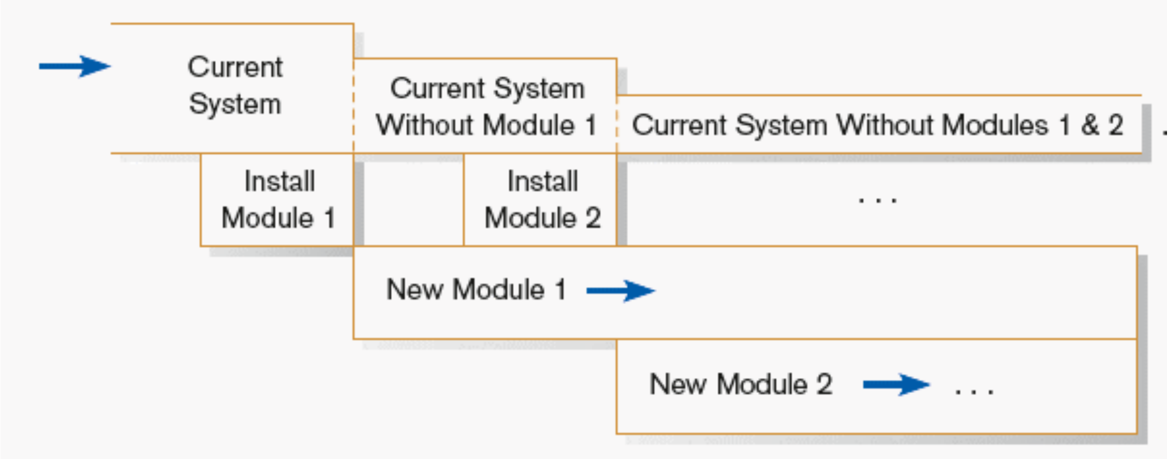
- At the location, same with direct installation.
- Low risk for different location.

✗ CONS

- At the location, same with direct installation.
- High risk at the same location.

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



PROS

- User learning advantage.
- Low risk.
- Easier Installation - incremental

CONS

- Long period of change.
- Careful version control.

PLANNING INSTALLATION

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

MAJOR TOPICS

DOCUMENTING THE SYSTEM

- Generic guideline

DOCUMENTING THE SYSTEM

- **System documentation**: detailed information about a system's design specifications, its internal workings, and its functionality
- **User documentation**: written or other visual information about an application system, how it works, and how to use it
- **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 E-R diagrams

DOCUMENTING THE SYSTEM

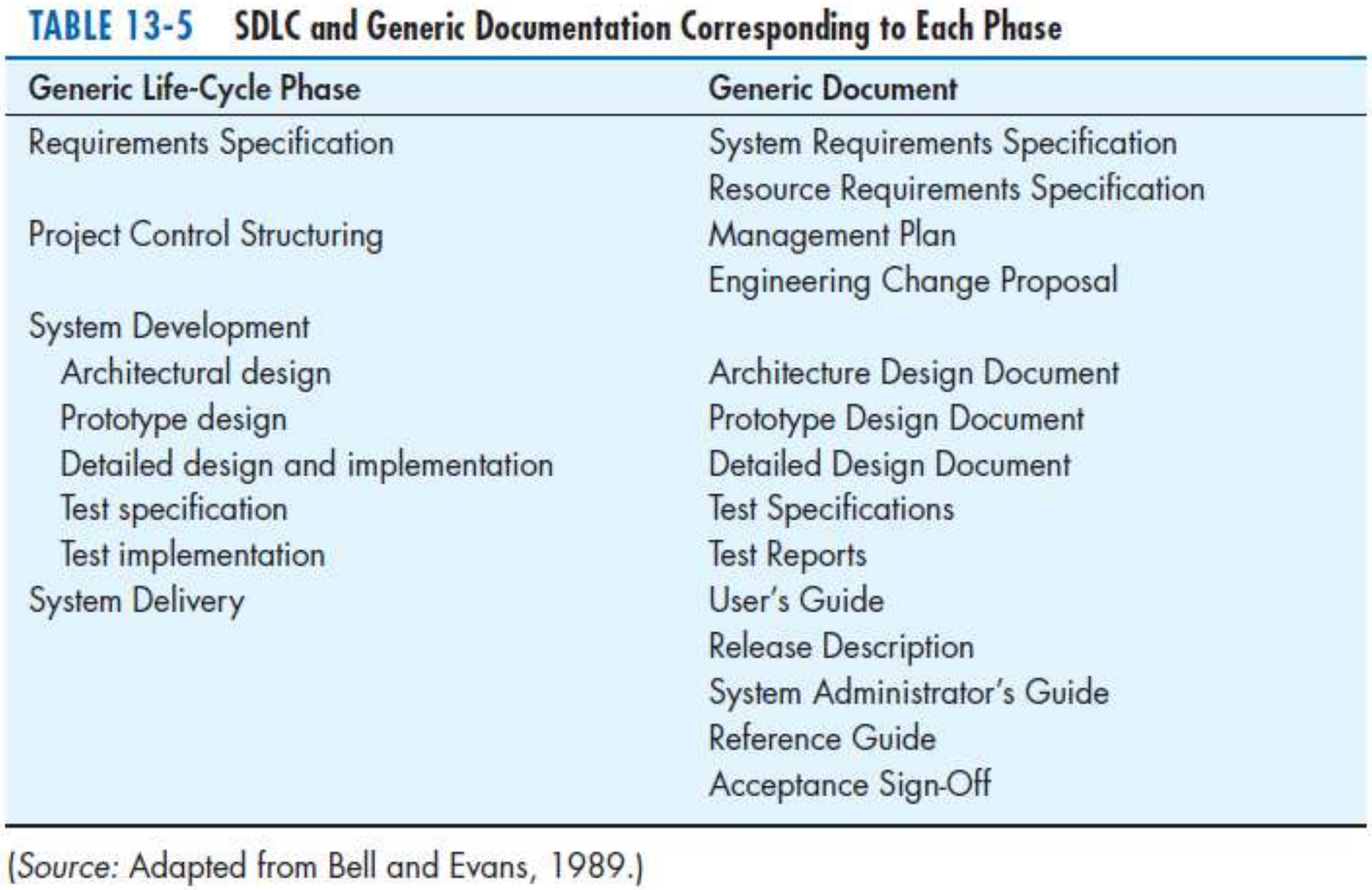


TABLE 13-5 SDLC and Generic Documentation Corresponding to Each Phase

Generic Life-Cycle Phase	Generic Document
Requirements Specification	System Requirements Specification Resource Requirements Specification
Project Control Structuring	Management Plan Engineering Change Proposal
System Development	
Architectural design	Architecture Design Document
Prototype design	Prototype Design Document
Detailed design and implementation	Detailed Design Document
Test specification	Test Specifications
Test implementation	Test Reports
System Delivery	User's Guide Release Description System Administrator's Guide Reference Guide Acceptance Sign-Off

(*Source:* Adapted from Bell and Evans, 1989.)

GENERIC USER'S GUIDE OUTLINE

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]
        - 3.n.n [Task name]

…1

Appendix A—Error Messages
([Appendix])

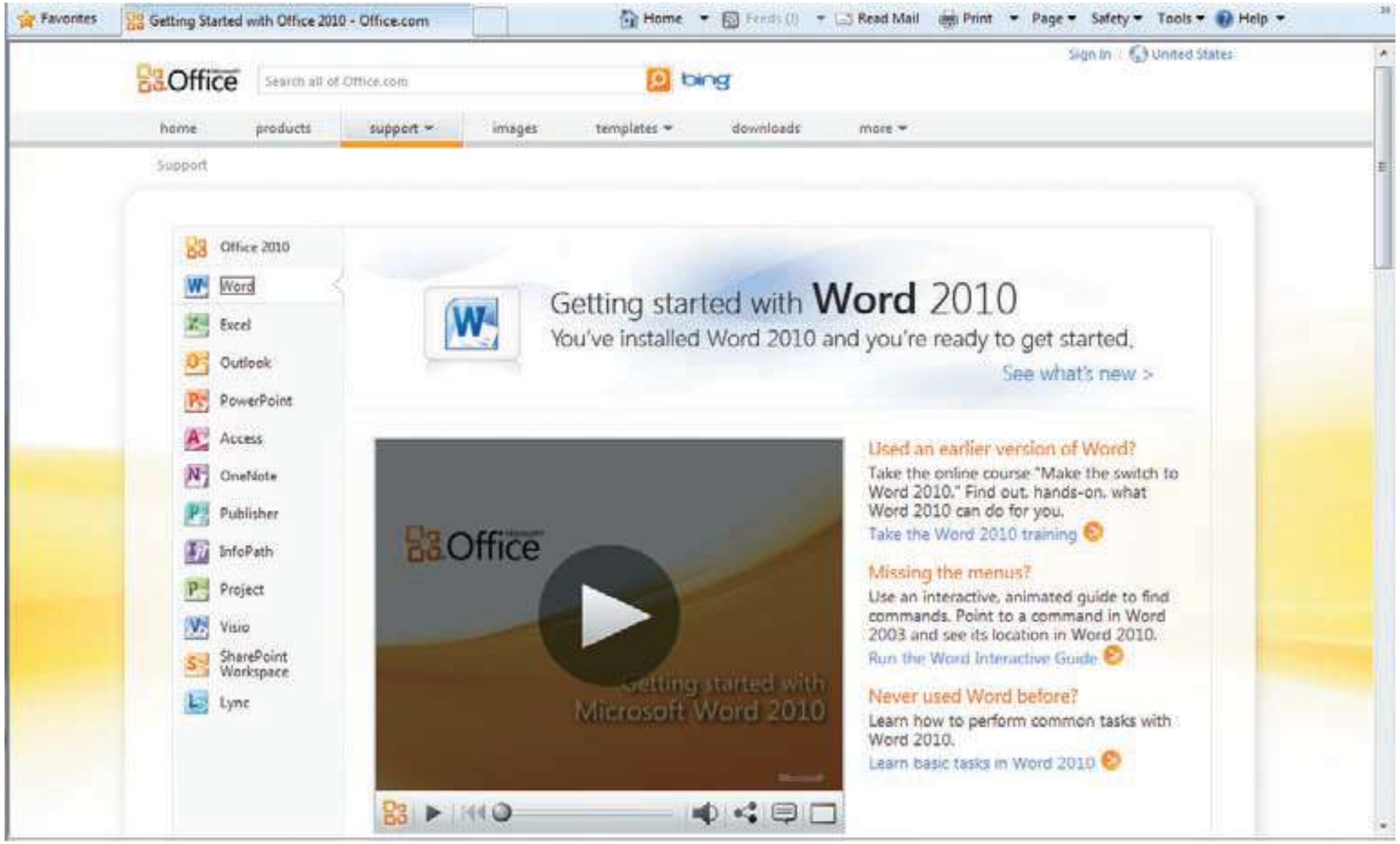
Glossary
- Terms
- Acronyms

Index

…2

Source: Adapted from Bell and Evans, 1989.)

DOCUMENTING THE SYSTEM



Getting started with **Word** 2010

You've installed Word 2010 and you're ready to get started.

See what's new >

Used an earlier version of Word?

Take the online course "Make the switch to Word 2010." Find out, hands-on, what Word 2010 can do for you.

Take the Word 2010 training

Missing the menus?

Use an interactive, animated guide to find commands. Point to a command in Word 2003 and see its location in Word 2010.

Run the Word Interactive Guide

Never used Word before?

Learn how to perform common tasks with Word 2010.

Learn basic tasks in Word 2010

Figure 13-7

Example of online user documentation

(*Source:* Microsoft Corporation.)

MAJOR TOPICS

TRAINING AND SUPPORT

- Types of training
- Support

TRAINING AND SUPPORTING USERS

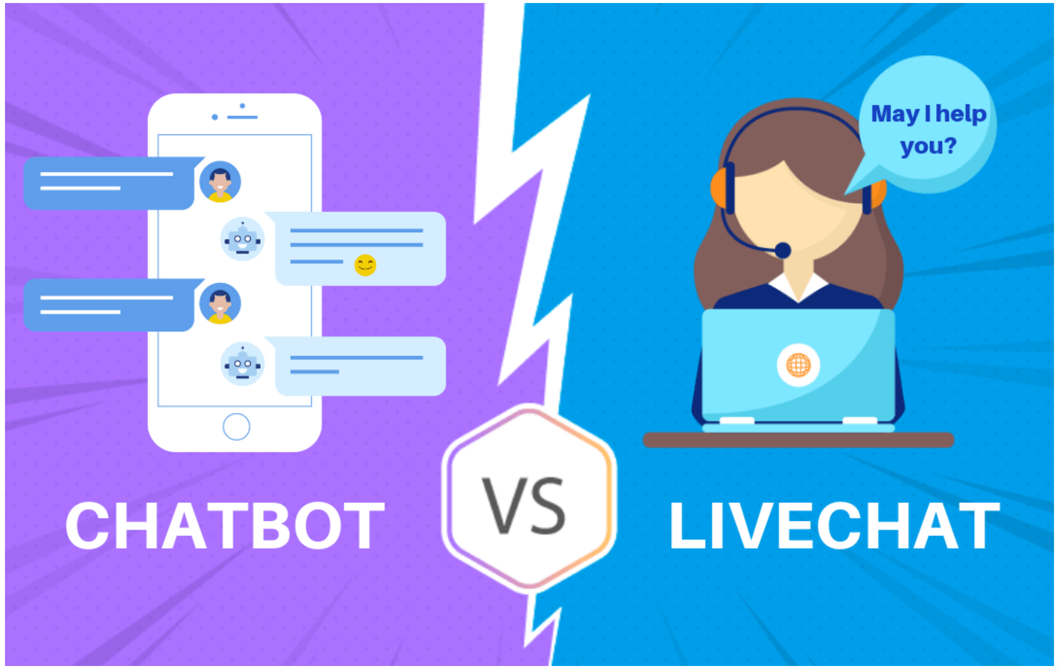
- **Training users**
  - Potential training topics
    1. Use of the system
    2. General computer concepts
    3. Information system concepts
    4. Organizational concepts
    5. System management
    6. System installation
- **Support**
  - providing ongoing educational and problem-solving assistance to information system users
  - For in-house developed systems, support materials and jobs will have to be prepared or designed as part of the implementation process.

TYPES OF TRAINING METHODS

- Resident expert
- Traditional instructor-led classroom training
- E-learning, distance learning
- Blended learning (instructor plus e-learning)
- Software help components
- External sources (e.g. vendors)

SUPPORTING USERS

- Support is important to users, but has often been inadequate.
- Providing support can be expensive and time-consuming.
- Vendors usually charge for their support, using 900- numbers, or charge a fee for unlimited or monthly support.
- One approach is through **automation**.
  - Internet-based online support forums
  - On-demand fax
  - Voice response systems
  - Knowledge bases



PROVIDING SUPPORT THROUGH HELPDESK

- **Help desk**: a single point of contact for all user inquiries and problems about a particular information system or for all users in a particular department
- 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 TO BE CONSIDER

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

MAJOR TOPICS

ORGANIZATIONAL ISSUES

- Factors
- Security issues
- Project closure

ORGANIZATIONAL ISSUES IN SYSTEM IMPLEMENTATION

- Why does implementation sometimes fail?
- Traditional wisdom of primary success factors:
  - Management support
  - User involvement
  - But these are not enough
- Other important factors
  - Commitment to project
  - Commitment to change
  - Extent of project definition and planning

FACTORS INFLUENCING SYSTEM USE

- Personal stake of users
- System characteristics
- User demographics
- Organizational support
- Performance
- Satisfaction

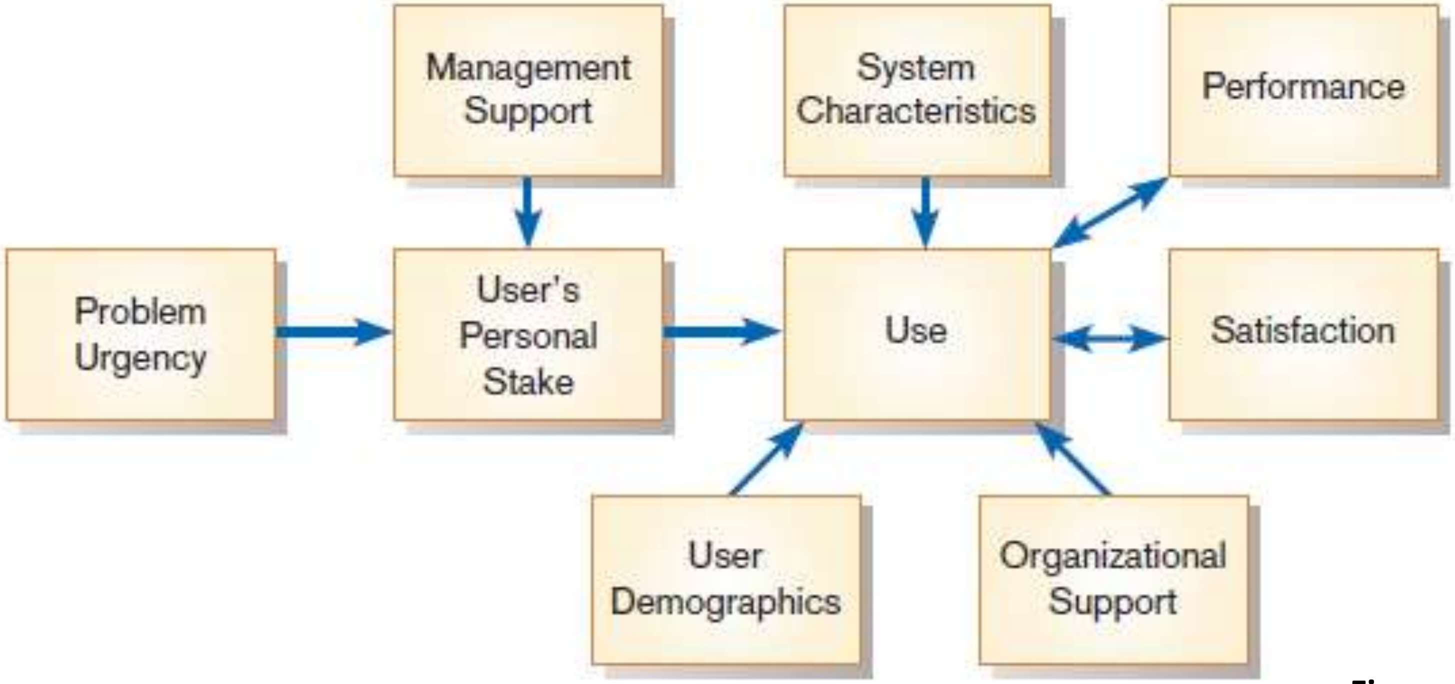


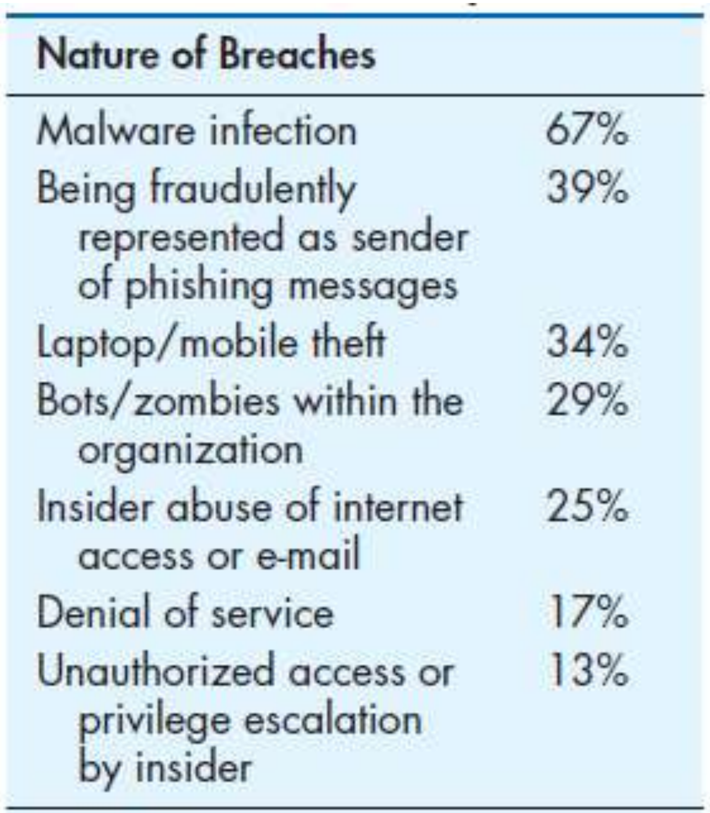
Figure 13-10

Implementation success

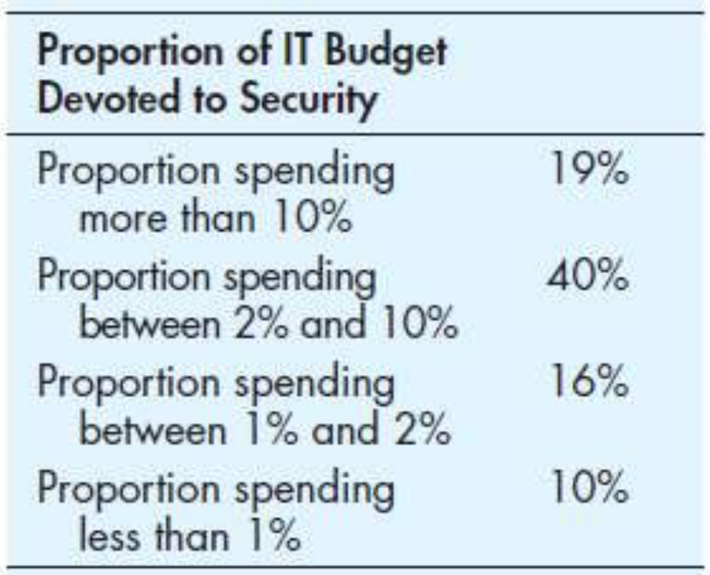
(*Source:* Adapted from Lucas, H. C. 1997.*Information Technology for Management*. New York: McGraw-Hill, with the permission of the McGraw-Hill Companies. All rights reserved.)

SECURITY ISSUES

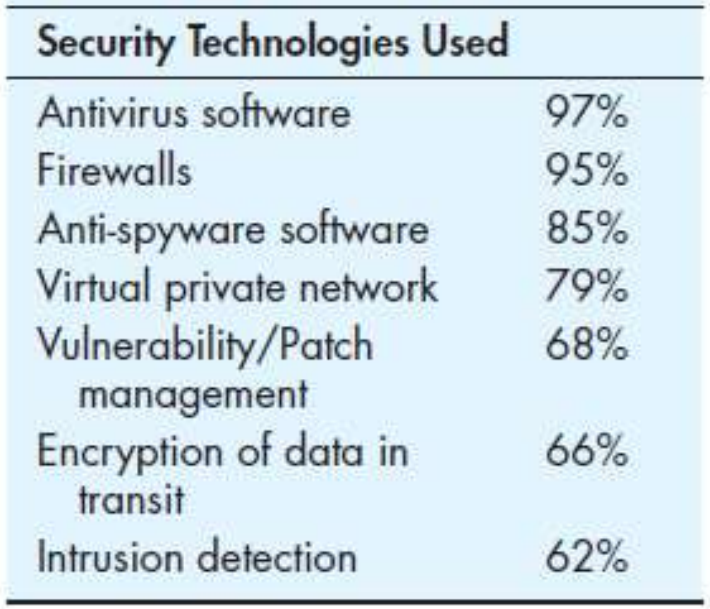
- Increasingly important issue for organizations and their management
- **Mal**icious soft**ware** (*malware*): includes Trojan horses, worms, viruses, and other kinds
- External sources of threats include laptop theft, system penetration, and denial of service.



Nature of Breaches	
Malware infection	67%
Being fraudulently represented as sender of phishing messages	39%
Laptop/mobile theft	34%
Bots/zombies within the organization	29%
Insider abuse of internet access or e-mail	25%
Denial of service	17%
Unauthorized access or privilege escalation by insider	13%



Proportion of IT Budget Devoted to Security	
Proportion spending more than 10%	19%
Proportion spending between 2% and 10%	40%
Proportion spending between 1% and 2%	16%
Proportion spending less than 1%	10%



Security Technologies Used	
Antivirus software	97%
Firewalls	95%
Anti-spyware software	85%
Virtual private network	79%
Vulnerability/Patch management	68%
Encryption of data in transit	66%
Intrusion detection	62%

(*Source:* Data from Computer Security Institute [Richardson], 2011.)

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

SUMMARY

- In this chapter you learned how to:

✓Describe the process of coding, testing, and converting an organizational information system and outline the deliverables and outcomes of the process.

✓Apply four installation strategies: direct, parallel, single-location, and phased installation.

✓List the deliverables for documenting the system and for training and supporting users.

✓Explain why system implementation sometimes fails.

✓Describe the threats to system security and remedies that can be applied.

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

univteknologimalaysia

utm_my

utmofficial

Thank You

update: August 2019 (sharinhh)

www.utm.my

innovative ● entrepreneurial ● global