



Chapter III

Systems implementation and Maintenance

Chapter Outline

- *System implementation*
- *Systems Maintaining information systems*

Systems Implementation & Operation

- During *implementation and operation*, *physical design specifications* must be turned into **working computer code**.
- Then the *code is tested until most of the errors have been detected and corrected*, the system is installed, *user sites are prepared for the new system*, and users must come to rely on the new system rather than the existing one to get their work done.
- The *implementation and operation* phase of the SDLC is the **most expensive and time consuming phase** of the entire life cycle.
- This phase is *expensive* because so many people are involved in the process.
- It is time consuming because of all the work that has to be completed through the entire life of the system.
- Systems implementation and operation is made up of **seven major** activities. These are **coding, testing, installation, documentation, training, support, and maintenance**.

I. The Process of Coding, Testing, and Installation

- *Coding is the process where by the physical design specifications created by the design team are turned into working computer code by the programming team.*
- *Once coding has begun the testing process can begin and proceed in parallel.*
- *As each program module is produced, it can be tested individually then as part of a larger program, and then as part of a larger system.*
- *Installation is the process during which the current system is replaced by the new system.*
- *This includes conversion of existing data, software, documentation, and work procedures to those consistent with the new system.*
- *The most obvious outcomes of this process are the code itself, but just as important as the code is documentation of the code.*
- *The results of program and system testing are important deliverables from the testing process because they document the tests as well as the test results.*

Cntd...

- The next **two deliverables**, *user guides*, and the *user training plan*, result from the installation process.
- *User guides* provide information on *how to use the new system*, and the training plan is a strategy for training users so they can quickly learn the new system.
- *Coding, testing and installation* work may be done by *IS professionals* in your organization, contractors, hardware designers, and increasingly users.
- *Table 1* provides the main deliverables from this process.

Action	Deliverable
Coding	<ul style="list-style-type: none">- Code- Program documentation
Testing	<ul style="list-style-type: none">- Test Scenarios (test plan) and test data- Result of program and system testing
Installation	<ul style="list-style-type: none">- User guides- User training plan- Installation and conversion plan<ul style="list-style-type: none">. Hardware and software installation schedule. Data conversion plan. Site and facility remodeling plan

III. Documenting the system, Training users, and Support users

- *As the team is getting ready to move on to new projects, you and the other analysts need to prepare documents that reveal **all the important information** you have learned about this system during its development and implementation.*
- *There are two audiences for this final documentation:*
 - (1). *the information system personnel who will maintain the system throughout its productive life and*
 - (2). *the people who will use the system as part of their daily lives.*
- *Users documentation can be paper based, but it should also include computer-based modules.*
- *The training plan should be supplemented by actual training modules or at least outlines of such modules that at a minimum address questions like who should be trained? And so on*

Cntd...

- *Finally, the **development team** should also deliver a user support plan that address such issues how users will be able to find help once the information system has become integrated into the organization.*
- *In table 2 presents the deliverables from documenting the system, training users and supporting users .*

Documentation	- User training modules
System documentation	Training materials
User documentation	Computer-based training aids
User training plan	- User support plan
Classes	Help desk
Tutorials	Online help
	Bulletin boards and other support mechanisms

III. Maintaining Information System

- *The process of maintaining an information system is the process of returning to the beginning of the SDLC and repeating development steps, focusing on the needs for system **change**, until the change is implemented.*
- *Four major activities occur with in maintenance.*
 1. *Obtaining maintenance request.*
 2. *Transforming requests into changes.*
 3. *Designing changes.*
 4. *Implementing changes.*
- *Obtaining maintenance request requires that a formal process be established where by users can submit system change request.*

Cntd...

- *Once a request is received, analysis must be conducted to gain an understanding of the scope of the request. Next a change request can be transformed into a formal design change, which can then be fed into the maintenance implementation phase.*
- *Finally, the SDLC phase implementation and maintenance equates to implementing changes.*
- *Because maintenance is basically a subset of the activities of the entire development process.*
- *the deliverables and outcomes from the process are the development of a new version of the software and new version of all design documents and training materials developed or modified during the maintenance process.*
- *This means that all documents created or modified during the maintenance effort, including the system itself, represent the deliverables and outcomes of the process.*

Software Application Testing

- *Testing software* begins earlier in the SDLC, even though many of the *actual testing activities* are carried out during implementation.
- *During analysis*, you develop an *over all test plan*.
- *During design*, you develop a *unit test plan*, an *integration test plan* and a *system test plan*.
- *During implementation*, these various plans are put into effect and the *actual testing* is performed.
- *There are Seven Types of Different Tests*
- *Software application testing* is an umbrella term that covers several types of tests.
- Tests can be done with or without executing the code and they *may be manual* or *automated*.
- Using this framework, we categorize types of tests as shown in Table 3.

Cntd...

	Manual	Automated
Without code execution	Inspection	Syntax testing
Without code execution	Walkthrough Desk checking	Unit testing Integrated testing System testing Stub testing

1. Inspection: is a testing technique in which participants examine program code for predictable language specific errors.

- Exactly what the code does is not investigated in an inspection.
- The inspection process can also be used to ensure that design specifications are accomplished.

Cntd...

2. Walkthrough: using structured walkthrough is a very effective method of detecting errors in a code.

- Unlike inspection, *what the code does is an important question in a walkthrough.*
- The purpose of walkthrough is to *detect errors, not to correct them.*
- It is the *programmers job to correct the errors uncovered in a walkthrough.*

3. Desk Checking: a testing technique in which the program code is sequentially executed manually by the reviewer.

- In one sense, the reviewer acts as the computer, *mentally checking each step and its result for the entire set of computer instruction.*

4. Syntax Checking: is typically done by a *compiler.*

- Errors in syntax are uncovered but *the code is not executed.*

Cntd...

5. Unit Testing: *it is sometimes called **module testing**.*

➤ *In unit testing, each module (roughly a section of code that performs a single function) is **tested alone** in attempt to discover any errors that may exist in the modules code.*

6. Integrated Testing: *the process of bringing together all of the modules that a program comprises for testing purpose.*

➤ *Modules are typically integrated in a top-down, incremental fashion.*

7. System Testing: *the bringing together of all the programs that a system comprises for testing purpose.*

➤ *Programs are typically integrated in a top-down, incremental fashion.*

➤ *System testing is intended to demonstrate whether a system meets its objective.*

➤ **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 sub ordinate modules.*

Acceptance Testing by User

- Once the *system tests have been satisfactorily completed*, the system is ready for acceptance testing, which is testing the system in *the environment where it will eventually be used*.
- The purpose of acceptance testing is for users to *determine whether the system meets their requirements*.
- The most complete acceptance testing will include:
 - 1) **Alpha Testing:** user testing of a completed information system *using simulated (false) data*.

The types of tests performed during alpha testing include:

- **Recovery testing:** forces the 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.

Cntd...

- **Stress testing:** tries to break the system (e.g., what happens when a record is written to the database with **incomplete information** or what happens under extreme online **transaction loads** or with a large number of concurrent users).
- **Performance testing:** determines how the system performs on the range of possible environments in which it may be used.
 - often the goal is have the system perform with similar response time and other performance measures in each environment.
- 2) **Beta Testing:** user testing of a completed information system using real data in the real user environment.
- Problems uncovered in alpha and beta testing in any of these areas must be corrected before users can accept the system.

Installation

- *It is the organizational process of changing over from the current information system to the new one.*
- *Four different approaches to installation have emerged over years:*
 - *Direct*
 - *Parallel*
 - *Single location*
 - *Phased*
- *Each installation process involves getting workers to change the way they work.*
- *A such installation should be looked at not as simple installing a new computer system, but as an organizational change process.*

Documenting the System

- There are **two** basic types of documentation.
 1. **System Documentation:** detailed information about a systems design specifications, its internal workings, and its functionality.
- System documentation can be further divided into internal and external documentation.
 - I. **Internal Documentation** a system documentation that is **part of the program source code** or is generated at **compile time**.
 - II. **External Documentation** a system documentation that includes the outcome structured diagramming techniques such as data flow and entity-relation ship diagram.
- 2. **User Documentation:** written or other **visual** information about an application system, **how it works**, and **how to use it**.

The documentation lists the item necessary to perform the task the user inquired about

Training and Supporting Users

- Training and support are *critical for the success of an information system*.
- Training and support help people adequately *use computer systems to do their primary work*.
- *Without proper training* and the opportunity to ask questions and gain assistance/consultation when needed, users *will misuse, under use, or not use the information system you develop*.
- Support is providing ongoing *educational and problem-solving assistance* to information system users. Support material and jobs must be designed along with the associated information system.

Conducting System Maintenance

- It is the *process of making changes to a system to fix or enhance its functionality.*
- There are several types of maintenance that you can perform on an information system.

1. *Corrective Maintenance:*

- refers to *changes made to repair defects in the design, coding, or implementation of the system.*
- Corrective maintenance *adds little or no value to the organization.*
- it simply focuses *on removing defects from an existing system without adding new functionality.*

2. *Adaptive Maintenance:* : involves *making changes to an information system to evolve its functionality to changing business needs or to migrate it to a different operating environment.*

- Unlike to corrective maintenance, adaptive maintenance is generally a *small part of an organizations maintenance effort but does add value to the organization.*

Cntd...

3. *Perfective Maintenance:*

- involves making *enhancements* to *improve processing performance*, interface usability, or to add desired, *but not necessarily required system features*.
- Many system professionals feel that perfective maintenance *is not really maintenance but new development*.

4. *Preventive Maintenance:*

- involves changes made to a system to *reduce the chance of future system failure*.
- *Adaptive, perfective, and preventive maintenance activities can lead to corrective maintenance activities* if not carefully designed and implemented.

End of chapter Three

Any Question?