Chapter 2
Documenting Accounting Information Systems

- Introduction
- Why Documentation Is Important
- Document and Systems Flowcharts
- Data Flow Diagrams
- Other Documentation Tools
- End-User Computing and Documentation

Documentation of Systems

- Documentation is a vital part of any AIS.
- Accountants use many different types of diagrams to trace the flow of accounting data through an AIS.
- A wide variety of software is available for documenting AISs.

Why Documentation Is Important

- Depicting how the system works
- Training users
- Designing new systems
- Controlling system development and maintenance costs
- Standardizing communications with others
- Auditing AISs
- Documenting business processes

Types of Documentation

- Document Flowcharts
- Systems Flowcharts
- Data Flow Diagrams

Document Flowcharts

- A document flowchart traces the physical flow of documents through an organization.
- Constructing a document flowchart begins by identifying the different departments or groups that handle the documents of a particular system.
- Auditors and accountants may use document flowcharts when analyzing a current system for weaknesses in controls and reports.
### Common Document Flowcharting Symbols

- Keying operation
- Manual Operation
- Document
- Connector between two points on a flowchart
- Multiple copies of a specific document
- Journal or ledger

Also on page 37 of text

### A Sample Document Flowchart

Requesting Department → Central Supplies Department

PRF 1 → 2

File

### Common System Flowchart Symbols

- Computer Processing
- Input/Output
- On-line keying
- Document
- Communication Link
- Magnetic Disk

### System Flowcharts

- **System flowcharts** depict the logical flows of data and processing steps in an AIS.
- They use symbols that are industry conventions standardized by the National Bureau of Standards.
- Each processing phase of a system flowchart usually involves preparing one or more control reports.
- These flowcharts depict an electronic job stream of data through processing phases of an AIS, and therefore illustrate audit trails.

### Systems Flowcharting Guidelines

- Should read from top to bottom and left to right.
- Use appropriate symbols.
- A process symbol should always be found between an input and an output symbol. This is called the sandwich rule.
- Use connectors to avoid crossed lines and cluttered flowcharts.
**Systems Flowcharting Guidelines**

- Sketch a flowchart before designing the final draft.
- Use annotated descriptions and comments in flowcharts for clarification.

**A Sample Systems Flowchart**

![Flowchart Diagram](image1)

Processing of Cash Remittances to Update AR

**Using Excel to Flowchart**

- Problem 2-12 (pg 60 of text)

**Using Excel to Flowchart**

- Problem 2-12 (pg 60 of text)
- Other Flowcharting tools
  - Visio ([www.microsoft.com](http://www.microsoft.com))
  - SmartDraw ([www.smartdraw.com](http://www.smartdraw.com))

**Data Flow Diagrams**

- A data flow diagram shows the logical flows of data through a transaction processing system of an organization.
- They are primarily used in the systems development process as a tool for analyzing an existing system.

**Symbols used in Data Flow Diagrams**

- A square represents an external data source or data destination.
- A circle indicates an internal entity that changes or transforms data.
- Two horizontal lines represent the storage of data. This is usually a file.
- A line with an arrow indicates the direction of the flow of data.
Parts of the DFD - Sources/ Destinations

- Data sources and destinations - represent an organization or individual that sends or receives data used or produced by the system.
- An entity can be both a source and a destination.
- Represented by squares.
- Examples: Supplier, customer, bank, cashier, credit manager.

Parts of the DFD: Data Flows

- Data flows - the flow of data between processes, data stores, and sources and destinations.
- Can be composed of more than one piece of data/elements.
- Different data flows cannot have the same name.
- Data flows moving in and out of data stores do not require names.
- A DFD does not indicate why a process began.
- Data flows can move in two directions.

Parts of the DFD: Data Processes

- Data processes - represent the transformation of data.
- Process names should include action verbs such as update, edit, prepare, reconcile and record (be as descriptive as possible).
- Each process is given a number to help readers move back and forth between levels of DFDs.

Parts of the DFD: Data Stores

- Data stores - temporary or permanent repository of data.
- Represented by horizontal, parallel lines with name inside.

Context Diagrams

- Data flow diagrams are usually drawn in levels that include increasing amounts of detail.
- A top level (or high-level) DFD that provides an overall picture of an application or system is called a context diagram.
- A context diagram is then decomposed, or exploded, into successively lower levels of detail.

Physical Data Flow Diagrams

- The first level of detail is commonly called a physical data flow diagram.
- It focuses on physical entities involved in the system under study, as well as the tangible documents, reports and other hard-copy inputs and outputs.
**Logical Data Flow Diagrams**

- Logical data flow diagrams address what participants do.
- Each bubble contains a verb that indicates a task the system performs.
- Logical data flow diagrams help designers decide what system resources to acquire, what activities employees must perform to run these systems, and how to protect and control these systems after installation.

**Decomposition**

- Decomposition is the act of exploding data flow diagrams to create more detail.
- Level 0 data flow diagrams may be exploded into successive levels of detail. The next level of detail would be a level 1 data flow diagram.
- The DFDs become linked together in a hierarchy, which would fully document the system.

**Guidelines for Drawing DFDs**

- Avoid detail in high level DFDs.
- Each logical DFD should contain between five and seven processes.
- Different data flows should have different names.
- All data stores should have data flows both into them and out of them.
- Temporary files are usually desirable to include in a DFD.

**Guidelines for Drawing DFDs**

- Classify most of the final recipients of system information as external entities.
- Classify personnel and departments that process the data of the current system as internal entities.
- Display only normal processing routines in high-level DFDs.
- If several system entities perform the same task, show only one to represent them all.

**A Sample Data Flow Diagram**

Data Flow Diagram for Payroll Processing

**Other Documentation Tools**

- Process maps
- Program flowcharts
- Decision tables
**Process Maps**

- Process maps document a business entity’s current business processes.
- Process maps use rectangles and arrows as its primary symbols.
- Consultants use process maps to assist in reengineering efforts.

**Process Map Example**

- Process Map for a restaurant
  - Teams of 4
  - Map a typical dining out at a restaurant

**Program Flowcharts**

- Organizations use **structured programming** techniques to create large computer programs in a hierarchical fashion.
- A **program flowchart** outlines the processing logic for each part of a computer program and indicates the order in which processing steps take place.
- The highest level program flowchart is called a **macro program flowchart** and serves as an overview of the data processing logic.

**Decision Tables**

- A **decision table** is a matrix of conditions and processing tasks that indicate what action to take for each possibility.
- They may be used when the computer program involves many conditions and subsequent courses of action.
- They may be used as alternatives to program flowcharts or in addition to the flowcharts.

**End-User Computing**

- **End-user computing** refers to the ability of non-computer employees to create their own computer applications.
- It is important for end-users to document applications they develop.
Controls for End-User Computing and Documentation

1. Formally evaluate large projects.
2. Develop formal end-user development policies.
3. Formalize documentation standards.
4. Limit the number of employees authorized to create end-user applications.
5. Audit new and existing systems.

CASE Tools

- CASE is an acronym for computer-assisted software engineering.
- CASE tools automate costly, inefficient, slow documentation tasks.
- CASE tools can reduce the time and cost to produce high-quality documentation for new systems, thus supporting rapid application development (RAD).

Copyright

Copyright 2005 John Wiley & Sons, Inc. All rights reserved. Reproduction or translation of this work beyond that permitted in Section 117 of the 1976 United States Copyright Act without the express written permission of the copyright owner is unlawful. Request for further information should be addressed to the Permissions Department, John Wiley & Sons, Inc. The purchaser may make backup copies for his/her own use only and not for distribution or resale. The Publisher assumes no responsibility for errors, omissions, or damages, caused by the use of these programs or from the use of the information contained therein.

Chapter 2