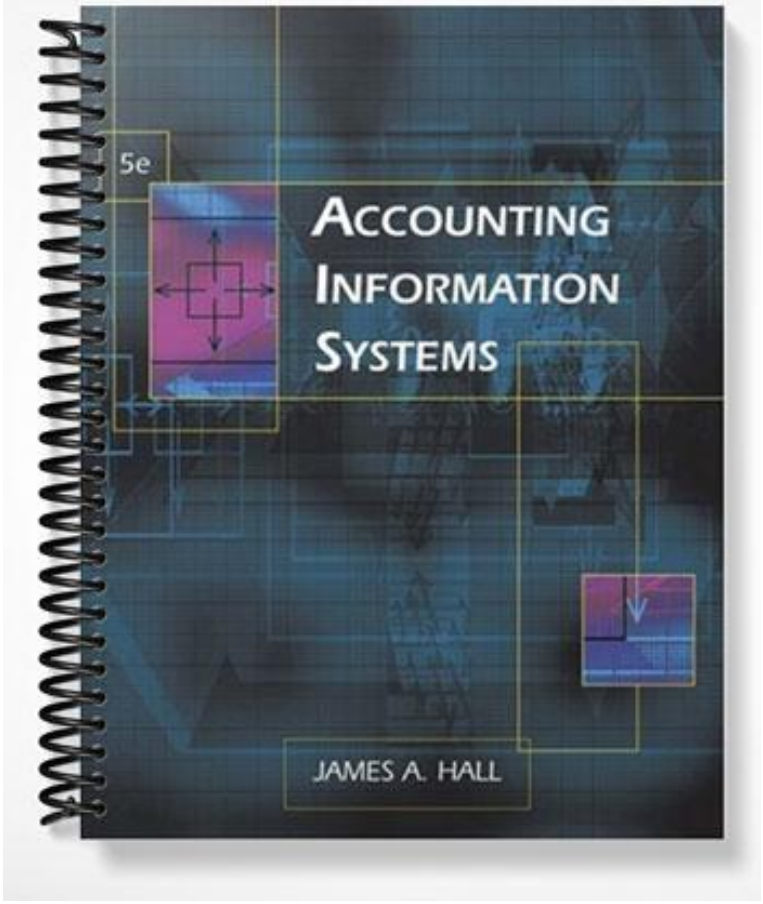
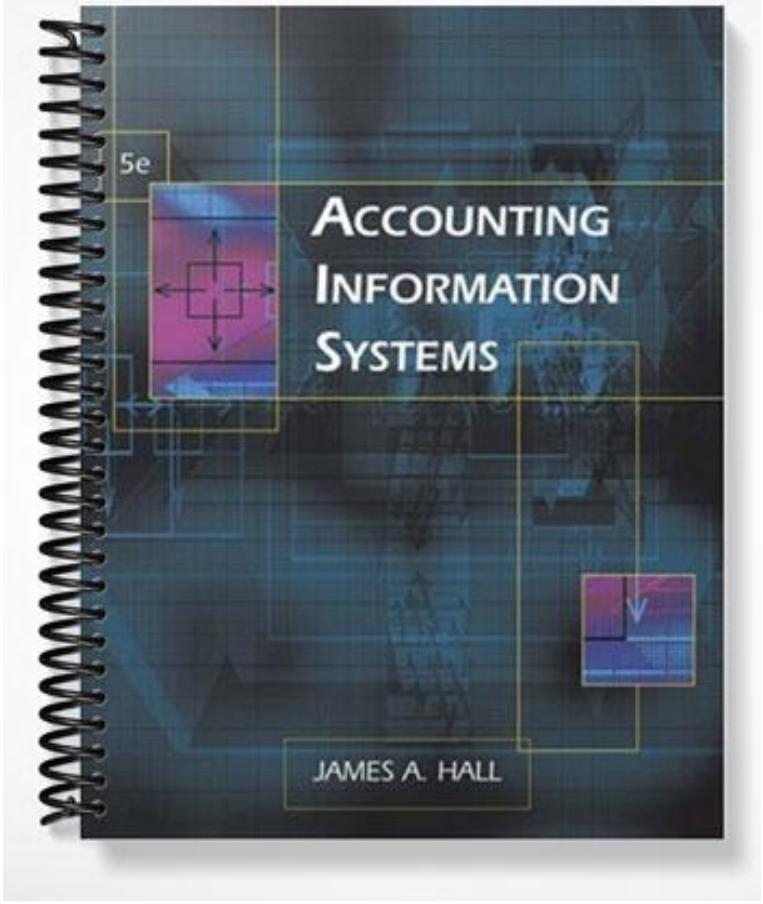
SOLUTIONS MANUAL



SOLUTIONS MANUAL



CHAPTER 2

INTRODUCTION TO TRANSACTION PROCESSING

OUTLINE OF CHAPTER 2

- Learning Objectives
- An Overview of Transaction Processing
  - Transaction Cycles
  - The Expenditure Cycle
  - The Conversion Cycle
  - The Revenue Cycle
- Accounting Records
  - Manual Systems
    - Documents
    - Journals
    - Ledgers
  - The Audit Trail
  - Computer-Based Systems
    - Types of Files
    - The Digital Audit Trail
- Documentation Techniques
  - Data Flow Diagrams and Entity Relationship Diagrams
    - Data Flow Diagram
    - Entity Relationship Diagram
    - Relationship Between ER Diagrams and Data Flow Diagrams
  - Flowcharts
    - Document Flowcharts
    - Batch Processing
    - System Flowcharts
    - Program Flowcharts
  - Record Layout Diagrams
- Computer-Based Accounting Systems
  - Differences Between Batch and Real-Time Systems
    - Time Lag
    - Resources
    - Operational Efficiency
    - Efficiency Versus Effectiveness
  - Alternative Data Processing Approaches
    - Legacy Systems versus Modern Systems
    - Updating master Files from Transactions
    - Database Backup Procedures
  - Batch Processing Using Real-Time Data Collection
  - Real-Time Processing
- Chapter Summary

Appendix

Section A: Secondary Storage
- Magnetic Tape
- Magnetic Disks
- Optical Disks

Section B: Legacy Systems

Data Structures
- Sequential Structure
- Direct Access Structures
- Indexed Structure
- VSAM Structure
- Hashing Structure
- Pointer Structure

Batch Processing Using Sequential Files
- Keystroke
- Edit Run
- Sort Runs
- Update Runs
- Sequential File Backup Procedures

Batch Processing Using Direct Access Files
- Direct Access File Update and Backup Procedures

General Logic for File Update
- Sequential File Update
- Start-Up
- Update Loop
- End Procedures
- Direct Access File Update

LECTURE NOTES

Learning Objectives

- To understand the broad objectives of transaction cycles.
- To recognize the types of transactions processed by each of the transaction cycles.
- To know the basic accounting records used in transaction processing systems.
- To understand the relationship between traditional accounting records and their magnetic equivalents in computer-based systems.
- To be familiar with the documentation techniques used for representing manual and computer-based systems.
- To understand the characteristic differences between batch and real-time processing and their impact on transaction processing.

Purpose of Chapter 2:

- To understand the different purposes and common characteristics of the revenue, expenditure and conversion cycles,
- To understand the role of the audit trail,
- To understand the documentation techniques utilized in business, and

- To understand the fundamental features of batch and real-time technologies and their implications for transaction processing.

An Overview of Transaction Processing

A financial transaction is an economic event that affects the assets and equities of the firm, is reflected in its accounts, and is measured in monetary terms. Financial transactions are common, repetitive events that involve external parties (customers or suppliers) or internal events (depreciation, use of materials, etc.). Similar transactions are grouped into cycles.

The Transaction Cycles

Figure 2-1 describes the three primary transaction cycles (expenditure, conversion, and revenue cycles) and their typical subsystems.

The Expenditure Cycle involves the acquisition of materials, property, and labor in exchange for cash. The actual disbursement of cash may occur at a different point after the receipt of the goods or services, based on a credit relationship. These financial transactions have a physical component and a financial component. The subsystems of the expenditure cycle include:

- Purchases/accounts payable system
- Cash disbursement system
- Payroll system
- Fixed asset system

The Conversion Cycle involves tracing the cost flows in the transformation of raw materials into finished goods, a primary concern of cost accounting courses. The conversion cycle (for manufacturing companies) has two typical subsystems:

- Production system (materials planning, scheduling, releasing, controlling)
- Cost accounting system (flow of costs for inventory valuation, budgeting, decisions)

The Revenue Cycle involves tracing a customers order through delivery (physical component) and payment (financial component), which may occur at two points in time. The subsystems of the revenue cycle include:

- Sales order processing (order, credit, shipping, billing)
- Cash Receipts (collecting, depositing, recording)

Accounting Records

Manual Systems

Manual Systems utilize documents to provide evidence of an economic event. The document may either initiate the transaction processing, or be a result of processing. There are three types of documents discussed in this chapter:

- Source documents: triggers created at beginning of the transaction to capture data. [Figure 2-2.]
- Product documents: are the results of transaction processing (invoices, paychecks). [Figure 2-3.]
- Turnaround documents: are product documents that become source documents for another system. (e.g., the part of the bill that gets returned with your check). [Figure 2-4.]

Journals are records of chronological entry. Documents are the primary source of data for journals. Each transaction requires a separate journal entry. There may be a time lag between when the source document is prepared and when the journal entry is made. There are two types of journals:

- Special Journals: group like types of transactions, and are more efficient. [Figures 2.5 and 2-6.] Periodic posting to general ledger and subsidiary ledgers is required from the special journals. Common special journals: sales, purchasing, cash receipts, cash disbursements.
- Registers can be a type of special journals (payroll register).
- General Journal: nonrecurring, infrequent, and dissimilar transactions. [Figure 2-7.] These events appear in chronological order. Journal voucher systems help keep these transactions controlled.

Ledgers: by account type, show the account balances, or the financial effects of the transactions on the related account(s). [Figure 2-8 shows flow from source documents to ledgers.] Two types of ledgers:

- General ledger: [Figure 2-9. Note the totals of all activities (beginning, current activity, ending balance) and control accounts for subsidiary ledger totals.]
- Subsidiary ledger: Groups of similar individual accounts (customers for accounts receivable, vendors for accounts payable, specific inventory items for inventory, specific machines for fixed assets, etc.). [Figure 2-10 illustrates the relationship between subsidiary ledger and general ledger.]

The Audit Trail

It is important that managers, accountants, and auditors be able to follow the audit trails of data through the accounting system. If records are kept properly, an accountant should be able to trace forward from source documents to the journals to the ledger(s) to the financial statements, as well as to vouch back from the financial statements to the ledger(s) to the journals to source documents. By tracing forward from source documents, auditors can verify the completeness of the recording of assets, and by vouching back from the recorded entry to the source documents, auditors can better verify the existence of assets claimed to be owned. Confirmation refers to the auditing technique of asking individual customers or suppliers to agree with the recorded balance for their account.

Computer-Based Systems

Types of Files in computer-based accounting systems include:

- Master files: contain account data that are updated from transaction data: ledgers.
- Transaction files: hold records from events that will change master files: journals.
- Reference files: hold the transaction processing standards or rules: vendors, prices.
- Archive files: are historical files of past journals and ledgers.

<u>The Digital Audit Trail</u>

Figure 2-11 illustrates the flow of accounting information in computer based systems. This figure explains the magnetic audit trail as well as introduces the flowcharting symbols. The numbers within circles refer to the auditor's steps in following the trail backwards (which is called vouching).

1. Compare financial statement A/R balance with general ledger control account.
2. Compare A/R control account balance to subsidiary ledger.
3. Select a sample of subsidiary ledger accounts and trace to sales journal.
4. Select a sample of sales journal transactions and trace to source documents and contact the customer.

Documentation Techniques

These are visual or graphic descriptions can convey accounting system components and flows more effectively and efficiently than words. This chapter explains six documentation techniques:

- Data flow diagrams,
- Entity relationship diagrams,
- Document flowcharts,
- System flowcharts
- Program flowcharts
- Record layout diagrams

Data Flow Diagrams and Entity Relationships Diagrams

Data Flow Diagrams (DFDs)

DFDs represent logical representations of entities, processes, data flows and stores that pertain to a system. Rather than showing how, where, or by whom processing is accomplished, DFDs show what processing needs to be performed. Figure 2-12 illustrates the symbols' meaning: squares (entities), ovals (processing verbs), rectangles (data storage), and arrows (data flows with direction indicated). Figure 2-13 is a DFD for a sales order processing system.

DFDs are built in hierarchical levels from broad system scope (level zero) to narrow subsystem scope (levels 1, 2, 3, and higher). For example, a broad scope level 0 DFD for sales process would describe approving credit sales, shipping product, billing customers, and preparing the accounts receivable. Then, a higher level DFD would take one of these subsystems (shipping product) and explain the substeps within that subsystem.

Entity Relationship (ER) Diagrams

Depict the established relationships between entities (resources, events, agents).
Rectangles represent entities (nouns). Entities can be:

- Resources: automobile, cash, inventory, a college major.
- Events: ordering inventory, receiving cash, payment, shipment, selecting a major.
- Agents: salesperson, vendor, customer, student.

Diamonds represent relationships (verbs). Relationships have inherent cardinalities representing the maximum number of records in one file that relate to one record in another file.

- 1:1 (one to one): one salesperson is assigned to one automobile.
- 1:M (one to many): one customer may owe on many invoices.
- M:M (many to many): many inventory items are purchased from many vendors.
- [Figure 2-14 illustrates three relationships.]

DFDs model the processes of a system. ER diagrams model the data that is used or affected by the system. Each data store in a DFD represents a corresponding data entity in the ER diagram.
Figure 2-15 illustrates the ER diagram for the DFD illustrated in Figure 2-13.

Flowcharts

Flowcharts represent the physical relationship between key entities. They explain how, where and by whom processes are accomplished:

- Document flowcharts: manual system components and processes.
- Systems flowcharts: contain both computerized system components and processes.
- Program flowcharts: describe the logic of an application programming.

Document flowcharts are typically created by following the hard copy of business forms utilized by a company (purchase order, customer order, etc.) from beginning to end. [Figures 2-16 and 2-17 illustrate the structure and symbols utilized in document flowcharting..]

- The internal entities (entities that perform data processing of some sort) are presented as columns in a document flowchart.
- Other document flowcharting symbols include: start/end of processing oval, source document, manual process, hardcopy file, accounting journals, registers, logs and ledgers, batch totals, on- and off-page connectors, comments, and data flow arrows.
- Figures 2-18, 2-19, and 2-20 illustrate step by step the document flowchart components for a typical sales process: prepare sales order, check credit, pick and ship goods to the customer. Note the source data, reference data, storage data, and data flows.

Batch processing refers to grouping large volumes of similar transactions to increase efficiency and to control the data processing. Batch processing occurs in both manual and computerized systems. Basically, several sums are calculated at the initial transaction grouping (e.g., total sales, total customer number, total number of items, etc.). Then, throughout the process, these totals can be checked and verified against the initial totals to ensure that no transactions have been lost, were added, or were included in error. Processing in batches adds control over this process.

- Batch size is determined as a function of the volume of transactions, the competitiveness of the industry, the normal frequency of errors, the financial implications of an undetected error, and the costs of processing.

System flowcharts contain both manual and computerized processes. Use Figure 2-16 (for the columns) and Figures 2-21, 2-22, and 2-23 (for the symbols) to walk through a sales order process.

- Compare the system flowchart to the Manual/Document flowcharting: the systems flowchart has a Computer Operations Department as a column and does not have a Credit Department.
- Note that the Data Flow Diagram for both of these would be identical. While there are physical differences, the same processing (the "what") is performed. Note that the flowcharts (the how, where, by whom) are different from each other, both in terms of columns and symbols utilized.
- Note the flow lines for some files only go one way (read or write) while others go both ways (read and write).

Program Flowcharts explain what a computer application program does to the data.
Figure 2-24 illustrates the program flowchart symbols: logical processes (rectangles), decisions (diamonds), starts/stops (ovals), input/output operations (parallelograms), and arrows (data flows).

- Use Figures 2-25 and 2-26 to illustrate a typical accounting transaction program flowchart. Program flowcharts are typically tested by auditors to ensure the correct processing of transactions.

Record Layout Diagrams

Figure 2-27 illustrates the purpose of record layout diagrams: to depict the fields or data attributes within a particular record, or instance of an entity.

Computer-Based Accounting Systems

Differences Between Batch and Real-Time Systems

The two types of computer-based accounting systems with respect to transaction processing classes are: batch systems and real-time systems. Table 2-1 illustrates the three characteristic differences between these types of systems:

- Time Lag refers to the period of time between the economic event and when it is recorded. With batch processing, data is collected over a set period of time (a day or certain number of days, a week, etc.) and then is processed all at once, or in a "batch." This choice of procedure creates a time lag between when a transaction takes place and when it is recorded in the accounting system, and only works in industries that do not need up-to-the-minute currency in their databases to be effective in the marketplace.
- Resources: the hardware, programming, and training required to perform the processing.
- Operational Efficiency refers to the amount of resources consumed per unit of output. Batch processing is typically more efficient when there are large volumes of similar transactions. Efficiency (minimizing the consumption of resources) needs to be evaluated with respect to effectiveness (accomplishing your goals). If up-to-the-minute information is required to properly conduct business effectively, then real-time processing must be used (e.g., an Internet retailer, such as Land's End or J.Crew, or an airline reservation system).

Alternative Data Processing Approaches

Legacy Sustems versus Modern Systems

Legacy systems typically use a large mainframe computer, are batch oriented, and can have highly structured and inflexible storage systems. The appendix to this chapter discusses the details of legacy systems. Modern systems are more likely to be client-server networks of distributed data processing in real-time, where transactions are stored in databases in real-time. The rest of Chapter 2 describes these modern systems.

Updating Master Files from Transactions

Figure 2-28 illustrates record structures and primary key fields that link the sales order/invoice transaction file with accounts receivable and inventory master files. Recording the sales order will cause domino effects in the accounts receivable file (changing the balance owed) and the inventory file (changing the number of units on-hand and available for sale to others).

Database Backup Procedures

Each record in a database needs a unique identifier (field or combination of fields), so that the record can be updated accurately with confidence. Destructive updating is when the prior field contents are replaced with the new field contents [Figure 2-29] Therefore, prior to each update, a backup is typically created [Figure 2-30].

Batch Processing using Real-Time Data Collection
Real time transaction file creation and subsidiary ledger update with delayed batch mode of sales journal and general ledger master file updating for efficiency purposes. Figure 2-31 illustrates this approach.

Real-Time Processing
Excellent for systems that process lower transaction volumes (reducing the efficiency reason for batches) and do not share common records (eliminating the need to "lock" master files). Figure 2-32 illustrates this kind of system, typically used when the product is ordered, filled and shipped in the same day.

Summary

- Transaction processing systems (TPS) support financial reporting, internal management reporting, and day-to-day operations. The most common are the revenue cycle, the expenditure cycle, and the conversion cycle.
- Batching of large volumes of similar transactions improves efficiency of data processing.
- Six documentation formats are described: data flow diagrams, entity relationship diagrams, flowcharts (document, systems, and programs), and record layout diagrams.
- Batch processing and real-time processing are described for both transaction file creation and master file update.

Appendix

Section A: Secondary Storage
The media may be magnetic tape, magnetic disk, or optical disks.

Magnetic Tapes: hold lots of data, are cheap and are reusable. However, they require sequential structures. A tape's density refers to the amount of characters that can be recorded on one inch. A tape drive reads/writes blocks of data at a time, separated by interblock gaps. Figure 2-33 illustrates a magnetic tape.

Magnetic Disks: hard or floppy are considered nonvolatile: the data will remain on the magnetic surface until replaced by different data or erased. However, the surface must be pre-formatted. A disk drive (DASD: direct access storage device) utilizes software to go directly to the record of interest on the disk. Access time refers to the time it takes for the operating system to request a piece of data to when it is read back into the computer. It is a function of seek time, switching time, and data transfer time. File allocation tables explain what data is stored where on the disk. Figure 2-34 illustrates a magnetic disk.

Disk address refers to the small blocks of storage space on individual tracks of the disk. Cylinder number, surface number, and record block number are described in the address.

Every disk has two surfaces and the same number of tracks on each surface.

Disk-Packs refer to stacked disks with identical tracks on the top and bottom surfaces of each disk. Eleven disks would have 22 surfaces, however typically the top and bottom surfaces are not used, so there would only be 20 usable tracks. Fogire 2-35 illustrates a disk pack.

Optical Disks can store much more data than magnetic technologies.

- CD ROM is a compact disk read-only memory.
- WORM is a compact disk where the user can write once and read many times.
- Erasable optical disk: is "write-many and read-many" times.

Section B: Legacy Systems

Data Structures

Data structure refers to the physical and logical organization and method of access of data in files and databases. The method of access affects the data processing. Table 2-2 lists the typical file processing operations (retrieve, insert, update, read, find next, search, delete). The two fundamental components are organization and method of access.

- Organization refers to the physical storage on secondary storage devices and involves either sequential or random storage.
    - Sequential refers to the physical method of storage and the sequential access. Figure 2-36 illustrates a sequential storage and access method. This typically requires the transaction data to be first sorted in the order of the file to be updated.
    - Random structures place the new data wherever there is room on the storage device, and then create an index that explains where the physical storage has occurred for each record (as the physical location of the record stored can be scattered throughout the storage device). The disk address has a unique numeric value that represents the cylinder, surface and block location for that data on the disk. This method requires no pre-sorting of data records. Typically there is an index structure that organizes each data record by logical sequence, and contains its numeric value of the physical disk storage location, for quick look-up.
- Method of access refers to the technique utilized to locate records in a data file or database in order to update, delete, or add records.

The "best" data structure depends on many variables, such as size of file, number of changes to be made to the file, etc.

Sequential Structure indicates that the data records are physically stored one after another in contiguous spaces in a specified sequence arranged by their primary key field. The typical technology utilized is either magnetic tapes or magnetic disks.

- This structure works best for reading a complete file of records, finding the next record in a file, or when a significant amount of the file will be updated (high activity ratio).
- Sequential structure does not work well if searching for only a few records in a large file.

An example would be payroll files at payday, when most records in the file refer to current employees that need to be paid. Figure 2-36 illustrates this structure and access method. Or a music cassette tape.

Direct Access Structures have physical location addresses that refer to cylinder, surface, and block location on magnetic disks (hard or floppy). Direct access systems physically store incoming data where ever there is physical space for that input (optimizing the storage utilization) and then stores this physical location address in an index table with the primary key of the data record. This allows a user to go immediately to a particular record without going through all of the prior records before it (which is what is required in a sequential structure). Think of a music CD.

- Index Structure is another term for files arranged with direct access structures. Figure 2-37 illustrates this structure and access method. Indexes may be either sequentially or randomly organized.

VSAM Structure refers to Virtual Storage Access Method, and is typically recommended for very large batch processing files that require a significant amount of individual record retrieval and updating processing. Figure 2-38 illustrates the VSAM structure. These are sequentially stored files with hierarchical indexes to allow direct access. VSAM files have three components: the indexes, the prime data storage, and the overflow area.

- The response time of these systems are slower than indexed sequential or indexed random structures because these files may cross different cylinders of a disk track and therefore must go through several indexes. Direct access speed has been compromised in order to receive the efficiencies needed for large batch processing.
- The greatest disadvantage of VSAM systems are that they do not perform record insertion operations efficiently and therefore should only be used with low volatility (low number of record changes) files. When an old record needs to be changed, the VSAM inserts a pointer into the correct insertion location, which leads the computer to the location in the overflow area where the new data has been stored. Periodically, the data in the overflow area is updated into the prime area, and the affected indexes updated. See Figure 2-39 for an example of this.

Hashing Structure utilizes an algorithm that converts the primary key of each record into a storage address, thereby eliminating the need for a storage/key index. Figure 2-40 illustrates hashing.

- The principal advantage of hashing is access speed.
- Principal disadvantages: storage space is not optimized and up to one-third of the storage space may be wasted, and there may be collisions because two records end up with the same key after the algorithm is applied. When this happens a pointer is added to the first record to direct the computer to the location of the second record with the same hash total. Collisions are risky because if the first record is deleted, the pointer to the second record could also be deleted. Some software move the second record to the first record's place, and others keep the pointer even if the first record is deleted.

Pointer Structures create a linked-list file, where a field of one record contains the address of a related record (either the next sequential record in a random file or the next record in the same type of product or sales region). This field is termed the "pointer." Figure 2-41 illustrates a pointer system used to create a linked-list file. There are several types of pointers:

- Physical address pointers: good for fast access and for disk controllers, risky if disks are reorganized.
- Relative address pointers: contain the position of a record in a file that must be further manipulated to convert it to the actual physical address.
- Logical key pointer: contains the primary key of the related record, which is then converted into the record's physical address by a hashing algorithm. Figure 2-42 illustrates each of these types of pointers.

Batch Processing Using Sequential Files

Figure 2-43 illustrates the typical batch system using sequential files. Consider walking the students through each of the steps of this batch system, explaining the backup method of parent and child for master files (on the right hand side of the figure). Each program execution in a batch system is called a run. The following are the sequential steps of this process:

- **Keystroke** is the first step: recording the transaction in machine-readable form. This creates the transaction file. Batch control totals are calculated and stored.
- **Edit Run** is the first run, identifying erroneous records and removing them to an error file, recalculating batch totals.
- **Sort Runs** are the runs where the transaction file is reordered to match the sequence of the primary key of the master file to be updated. Figure 2-44 illustrates the interdependencies between the record structures.
- **Update Runs** actually change the records of the master file.
- **Sequential File Backup Procedures** refer to the grandfather-father-son approach.

Batch Processing Using Direct Access Files

Direct access eliminates the need for sorting activities and eliminates the automatic file backup, requiring a separate backup procedure before the run. Figure 2-45 illustrates batch processing with direct access files. Updating Files is typically termed "destructive updates" because the prior balance is replaced.

General Logic for File Update

Sequential File Update

Assumptions: the transaction file is smaller than the master file, there may be more than one change to a single master file record, both transaction file and master file are in sequential order on the same key field, and the master file is correct. [Use Figure 2-46.]

- Start-Up: reading the first record from each of the transaction and master files.
- Update Loop: Compare the fields of both records and determine:
    - If the keys are identical, the master file record is updated, and the next transaction file record is read.
    - If the keys do not match, a new master file record is read and the update loop is restarted.
    - If the transaction key is lower than the master file key, an error results.
- End Procedures: When transaction EOF (end of file) is reached, then the remaining records in the master file must be read to the new master file.

Figure 2-47 illustrates the issues with sequential file update.

Direct Access File Update

- Record retrieval is performed by the computer's operating system, not the application software, no new master is created. Logic is much simpler than sequential file updating, and there is no need to compare the transaction and master file key fields. Figure 2-48 illustrates the direct access file update method.

Sequential files are efficient when a large portion of the records in the file require updating. Direct access processing is efficient when the variety of operations that need to be performed is high.