

# STUDY UNIT THIRTEEN

## INFORMATION TECHNOLOGY III

13.1	Operating Systems .....	1
13.2	Security .....	2
13.3	Types of Data Files .....	5
13.4	Nature of Binary Data Storage .....	6
13.5	File Organization and Access Methods .....	8

This study unit is the third of five covering information technology (IT).

### 13.1 OPERATING SYSTEMS

1. Every computer requires an **operating system**. The operating system negotiates the conversation between the computer's hardware, the application the user is running, and the data that the application is working with.
  - a. With early computers, one application had to be loaded, run to completion, then unloaded before another one could be run.
  - b. The first refinement to this limitation was **multiprogramming**, in which a second program could begin running while the first program was waiting for a command from the operator, or for input from a slower device such as a card reader.
  - c. An important feature of the current generation of operating systems is **multitasking**, in which the operating system rapidly switches the computer's attention back and forth between programs, sometimes in a fraction of a second, giving the appearance to users of jobs running simultaneously.
  - d. Multitasking should be contrasted with **multiprocessing**, in which the computer has multiple CPUs, permitting a single application to be broken up and have its parts run in parallel on the various processors, greatly speeding up completion times.
2. **z/OS** is the dominant operating system for IBM-compatible mainframes. It is the culmination of decades of mainframe operating system development by IBM.
3. For servers, popular operating systems include the following:
  - a. **UNIX** was developed by programmers at Bell Labs in the 1960s and 1970s.
    - 1) Their motivation was to create an operating system that was portable (i.e., could be used on many brands of computer), multi-user (allow more than one person at a time to use the computer), and multi-tasking (see above). Over the years, UNIX has been greatly expanded and refined, and is considered a very robust operating system for servers.
    - 2) Many companies offer their own customized versions of UNIX. Two well-known variants are:
      - a) Linux, which, unlike most UNIX distributions, is free (although versions of Linux with proprietary add-ons and technical support do involve some cost).
      - b) Solaris, a proprietary UNIX-like operating system from Sun Microsystems. It is primarily used on high-end Sun servers and Sun workstations, though a version is available for PCs.
  - b. **Windows Server** is the networking version of Microsoft's wildly popular Windows operating system for the desktop.
  - c. **Novell Open Enterprise Server** is the successor to that company's once-dominant NetWare network operating system.

4. For desktop and laptop computers, three operating systems predominate:
  - a. **Microsoft Windows** in its many variants (Windows XP, Windows ME, Windows Vista, etc.) owns the largest market share, particularly among large organizations.
  - b. **Mac OS X** (Roman numeral ten) is designed to run on desktop computers built by Apple. Apple computers are heavily favored by those in the graphics and desktop publishing fields.
  - c. **Linux** and other variants of UNIX are used for desktop computers and powerful workstations devoted to scientific and engineering functions.
5. Early operating systems of necessity required the user to type in commands from the keyboard stroke by stroke.
  - a. An important feature of any modern desktop operating system such as Windows or OS X is the **graphical user interface (GUI)**. The essence of GUI is “point-and-click,” the ability to use a mouse or touchpad to issue commands to the computer by manipulating pictorial icons on the screen.
  - b. Another characteristic of GUI is **windowing**, the ability of a computer to display more than one program on the screen at the same time. Each program has its own section of the screen, called a window.

## 13.2 SECURITY

1. **Information security** encompasses not only computer hardware and software but all of an organization's information, no matter what medium it resides on. It involves far more than just user IDs and passwords.
  - a. The importance of a broad definition of information security becomes clear in light of recent incidents of firms accidentally disposing of documents containing confidential customer information with their regular trash.
  - b. Organizations have three principal **goals** for their information security programs: data confidentiality, data availability, and data integrity.
    - 1) **Confidentiality** is protecting data from disclosure to unauthorized persons.
    - 2) **Availability** is assuring that the organization's information systems are up and running so that employees and customers are able to access the data they need (this topic is addressed in depth in Subunit 14.4).
    - 3) **Integrity** is assuring that data accurately reflect the business events underlying them and are not subject to tampering or destruction.
2. The organization accomplishes these goals by performing the following steps:
  - a. **Identify the threats** to the organization's information, i.e., events that can potentially compromise an organization's information infrastructure.
    - 1) Threats to confidentiality include the above-mentioned improper disposal of customer records; threats to availability include viruses and denial-of-service attacks; and threats to integrity include employee errors and disgruntled employee sabotage.
  - b. **Identify the risks** that these threats entail.
    - 1) Risk analysis has two phases: determining the likelihood of the identified threats and the level of damage that could potentially be done should the threats materialize.
    - 2) For example, an organization may conclude that, while the potential damage from sabotage is very high, its likelihood may be quite low.

- c. **Design the controls** that will compensate for the risks.
    - 1) Controls are designed based on the combination of likelihood and potential damage determined in the risk analysis.
    - 2) Controls are of three major types: physical, logical, and policy.
  - d. **Incorporate the controls** into a coherent **enterprise-wide information security plan**.
    - 1) The plan lists the controls that will be put in place and how they will be enforced.
  - e. **Policies** set forth expectations of all persons, both employees and external users, with access to the organization's systems.
    - 1) The single most important policy is that which governs the information resources to which individuals have access and how the level of access will be tied to their job duties.
      - a) Carrying out such a policy requires the organization's systems to be able to tie data and program access to individual system IDs.
      - b) One provision of the policy must be for the immediate removal of access to the system by the IDs of terminated employees.
3. The classic division of controls in information systems is between general controls and application controls.
- a. **General controls** relate to the organization's information systems environment as a whole. They include:
    - 1) **IT administration**
      - a) A modern organization should recognize information technology as a separate function with its own set of management and technical skills. An organization that allows every functional area to acquire and administer its own systems in isolation is not serious about proper control.
      - b) Treating IT as a separate functional area of the organization involves the designation of a chief information officer (CIO) or chief technology officer (CTO) and the establishment of an information systems steering committee to set a coherent direction for the organization's systems and prioritize information technology projects.
    - 2) **Separation of duties** within the IT function. See Subunit 11.3.
    - 3) Controls over **systems development**. See Subunit 11.4.
    - 4) **Hardware controls**
      - a) Hardware controls are built into the equipment by the manufacturer. They assure the proper internal handling of data as they are moved and stored.
      - b) They include parity checks, echo checks, read-after-write checks, and any other procedure built into the equipment to assure data integrity.
    - 5) **Physical controls** limit physical access and environmental damage to computer equipment and important documents. They include:
      - a) **Access controls.** No persons except operators should be allowed unmonitored access to the computer center. This can be accomplished through the use of a guard desk, a keypad, or a magnetic card reader.
        - i) The distribution of printed reports must be controlled so that unauthorized persons are not able to view data that are not connected with their job duties. This encompasses the proper disposal of documents in such a way that the disclosure of confidential customer or company data is prevented (e.g., shredding).

- b) **Environmental controls.** The computer center should be equipped with a cooling and heating system to maintain a year-round constant level of temperature and humidity, and a fire-suppression system.
- 6) **Logical controls** are established to limit access in accordance with the principle that all persons should have access only to those elements of the organization's information systems that are necessary to perform their job duties. Logical controls have a double focus, authentication and authorization.
  - a) **Authentication** is the act of assuring that the person attempting to access the system is in fact who (s)he says (s)he is. The most widespread means of achieving this is through the use of **IDs and passwords**. The elements of user account management are:
    - i) Anyone attempting access to one of the organization's systems must supply a unique identifier (e.g., the person's name or other series of characters) and a password that is known only to that person and is not stored anywhere in the system in unencrypted format.
      - Not even information security personnel should be able to view unencrypted passwords. Security personnel can change passwords, but the policy should require that the user immediately changes it to something secret.
    - ii) The organization's systems should force users to change their passwords periodically, e.g., every 90 days.
    - iii) The policy should prohibit employees from leaving their IDs and passwords written down in plain view.
  - b) **Authorization** is the practice of assuring that, once in the system, the user can only access those programs and data elements necessary to his/her job duties.
    - i) In many cases, users should be able to view the contents of some data fields but not be able to change them.
    - ii) An example is an accounts receivable clerk who can view customers' credit limits but cannot change them. This same clerk can, however, change a customer's outstanding balance by entering or adjusting an invoice.
    - iii) To extend the example, only the head of the accounts receivable department should be able to execute the program that updates the accounts receivable master balance file. An individual clerk should have no such power.
  - c) A **firewall** is a combination of hardware and software that separates an internal network from an external network (e.g., the Internet) and prevents passage of specific types of traffic (see Subunit 15.6).
    - i) Firewall systems ordinarily produce reports on organization-wide Internet use, exception reports for unusual usage patterns, and system penetration-attempt reports. These reports are very helpful as a method of continuous monitoring, or logging, of the system.
    - ii) A firewall alone is not an adequate defense against computer viruses. Specialized anti-virus software is a must (see Subunit 15.6).
- 7) Backup and contingency planning. See Subunit 14.4.

- b. **Application controls** relate to specific tasks performed by each system. They should provide reasonable assurance that the recording, processing, and reporting of data are properly performed. Application controls relate to individual computerized accounting applications, for example, programmed edit controls for verifying customers' account numbers and credit limits.
- 1) **Input controls** provide reasonable assurance that data received for processing have been properly authorized, converted into machine-sensible form, and identified.
    - a) They also provide reasonable assurance that data (including data transmitted over communication lines) have not been lost, suppressed, added, duplicated, or otherwise improperly changed. Moreover, input controls relate to rejection, correction, and resubmission of data that were initially incorrect.
    - b) An extensive list of input controls can be found in item 2. in Subunit 14.3.
  - 2) **Processing controls** provide reasonable assurance that processing has been performed as intended for the particular application.
    - a) All transactions should be processed as authorized, no authorized transactions should be omitted, and no unauthorized transactions should be added.
  - 3) **Output controls** provide assurance that the processing result (such as account listings or displays, reports, files, invoices, or disbursement checks) is accurate and that only authorized personnel receive the output.
    - a) Examples of output controls can be found in item 3. in Subunit 14.3.

### 13.3 TYPES OF DATA FILES

1. Data files can be classified as one of two main types.
  - a. A **master file** comes in two subtypes:
    - 1) One type contains records that do not change very often. An example is a vendor file, containing each vendor's number, name, and address.
    - a) **EXAMPLE:** Vendor master file

vendor_num	vendor_name	address_1	city	state	zip	credit_limit	last_updated
0187634	Neyland's Nuts	101 Dandridge Av	Knoxville	TN	37915	\$10,000	07/19/2002
1264428	Basic Barbecue	2224 Blossom St	Columbia	SC	29201	\$50,000	06/25/2005
4552170	Bayou Bakery	10118 Florida St	Baton Rouge	LA	70801	\$15,000	03/04/2006
5006321	Bulldog Barcoding	9085 Old West Point Rd	Starkville	MS	39759	\$5,000	10/01/2006
8981463	Razorback Restaurant Supply	3510 West Maple St	Fayetteville	AR	72701	\$20,000	07/01/2004

- 2) The other type of master file is one that is regularly updated to reflect ongoing activity. An example is a general ledger file, which at any given moment holds the balances of all accounts in the ledger.

a) EXAMPLE: General ledger file

account_num	account_name	balance	last_transaction_posted
A1209	Cash	\$89,580.22	01/10/2008
G6573	Accounts Receivable	\$72,024.57	01/10/2008
J0226	Accounts Payable	\$(15,156.89)	01/10/2008
K4411	Sales	\$(100,558.60)	01/10/2008
M2020	Cost of Goods Sold	\$70,005.64	01/10/2008
Y3577	Administrative Expenses	\$21,110.33	01/10/2008

- 3) A master file's **volatility** is the relative frequency with which records are added, deleted, or changed during a period.

b. A **transaction file** contains the data that reflect ongoing business activity, such as individual purchases from vendors or general journal entries.

1) EXAMPLE: Daily general journal file

transaction	transaction_date	account_num	debit	credit
GL5261904	01/10/2008	G6573	\$1,001.56	\$0.00
GL5261905	01/10/2008	J0226	\$0.00	\$(659.48)
GL5261906	01/10/2008	A1209	\$898.15	\$0.00
GL5261907	01/10/2008	K4411	\$0.00	\$(4,500.12)
GL5261908	01/10/2008	M2020	\$660.48	\$0.00
GL5261909	01/10/2008	Y3577	\$150.75	\$0.00
GL5261910	01/10/2008	R2112	\$0.00	\$(770.10)
GL5261911	01/10/2008	H8810	\$800.80	\$0.00
GL5261912	01/10/2008	Q4851	\$1,378.44	\$0.00

2. Transaction files and master files are constantly interacting.

- Before an invoice can be paid, the payables transaction file must be matched against the vendor master file to see whether the vendor really exists.
- The general ledger balance file must be updated every day by posting from the general journal transaction file.

## 13.4 NATURE OF BINARY DATA STORAGE

- The digital computers in common use today store all information in **binary** format, that is, as a pattern of ones and zeros. This makes arithmetic operations and true/false decisions on the lowest level extremely straightforward.
  - A **bit** (sometimes thought of as a contraction of "binary digit") is either 0 or 1 (off or on) in binary code. Bits can be strung together to form a binary (i.e., base 2) number.

EXAMPLE of a bit:

0

- b. A **byte** is a group of bits. Each byte is used to signify a character (a number, letter of the alphabet, or symbol, such as a question mark or asterisk).
- 1) The dominant coding systems for mapping the values of binary numbers to characters are the following:
    - a) Extended Binary Coded Decimal Interchange Code (EBCDIC), which was developed by IBM for its mainframe computers and uses 8 bits to a byte.
    - b) American Standard Code for Information Interchange (ASCII), which was developed by the American National Standards Institute, is employed by most personal computers and servers, and uses 7 bits to a byte (often padded to 8).

EXAMPLE of a 7-bit ASCII byte representing the letter P:

1010000
---------

- c) Unicode, sponsored by the International Organization for Standards, which can use multiple bytes to represent each character, thereby enabling the deployment of special characters and all the world's alphabets.
- 2) Quantities of bytes are measured with the following units:
 

$1,024 (2^{10}) \text{ bytes} = 1 \text{ kilobyte} = 1 \text{ KB}$   
 $1,048,576 (2^{20}) \text{ bytes} = 1 \text{ megabyte} = 1 \text{ MB}$   
 $1,073,741,824 (2^{30}) \text{ bytes} = 1 \text{ gigabyte} = 1 \text{ GB}$   
 $1,099,511,627,776 (2^{40}) \text{ bytes} = 1 \text{ terabyte} = 1 \text{ TB}$

- c. A **field** is a group of bytes. The field contains a unit of data about some entity, e.g., a composer's name.

EXAMPLE of a field:

Paul Hindemith
----------------

- d. A **record** is a group of fields. All the fields contain information pertaining to an entity, e.g., an orchestral work.

EXAMPLE of a record:

Paul Hindemith	Violin Concerto	Chicago Symphony	Claudio Abbado	Josef Suk
----------------	-----------------	------------------	----------------	-----------

- 1) Some field or combination of fields on each record is designated as the key. The essence of a key is that it contains enough information to uniquely identify each record, i.e., there can be no two records with the same key.
  - a) The designation of a key allows records to be sorted and managed with much greater efficiency. If all the records are sorted in the order of the key, searching for a particular one becomes much easier.
  - b) In the above example, the key is the combination of the first two fields.
    - i) The first field alone is not enough because there could be several works by each composer. The second field alone is likewise not enough since there could be many pieces with the same title.
    - ii) The combination of composer's name and title uniquely identify each piece of music.

- e. A **file** is a group of records. All the records in the file contain the same pieces of information about different occurrences, e.g., performances of several orchestral works.

EXAMPLE of a file:

Paul Hindemith	Violin Concerto	Chicago Symphony	Claudio Abbado	Josef Suk
Gustav Mahler	Das Lied von der Erde	New York Philharmonic	Leonard Bernstein	Dietrich Fischer-Dieskau
Bela Bartok	Piano Concerto No. 2	Chicago Symphony	Sir Georg Solti	Etsko Tazaki
Arnold Schoenberg	Gurrelieder	Boston Symphony	Seiji Ozawa	James McCracken
Leos Janacek	Sinfonietta	Los Angeles Philharmonic	Simon Rattle	None
Dmitri Shostakovich	Symphony No. 6	San Francisco Symphony	Kazuhiro Koizumi	None
Carl Orff	Carmina Burana	Berlin Radio Symphony	Eugen Jochum	Gundula Janowitz

### 13.5 FILE ORGANIZATION AND ACCESS METHODS

- To understand the vast improvement in performance brought about by database technology, it is helpful to review the development of file structures.
- The oldest file structure is the **flat file**, meaning that every record in the file has an identical layout; thus, the records can be conceived of as forming a two-dimensional pattern of rows and columns, like the table above. A telephone directory is a commonly encountered flat file.
  - The **linked list** was the earliest means of associating the records of a flat file with each other. Each record had a pointer tacked on the end that “pointed” to the next record.
- Variable-length records** represented a space-saving improvement. In the example below, a customer orders two different items on one occasion and only one item on another occasion. With variable-length records, valuable space is not taken up for the blank second item on the second order.

EXAMPLE of two variable-length records:

Record	Customer	Street	City	Order_Nbr	Part_Nbr_1	Qty_1	Price_1	Ext_1	Part_Nbr_2	Qty_2	Price_2	Ext_2
116385	Zeno's Paradox Hardware	10515 Prince Avenue	Athens, GA	19742133	A316	3	\$0.35	\$1.05	G457	12	\$1.15	\$13.80

(Many intervening records)

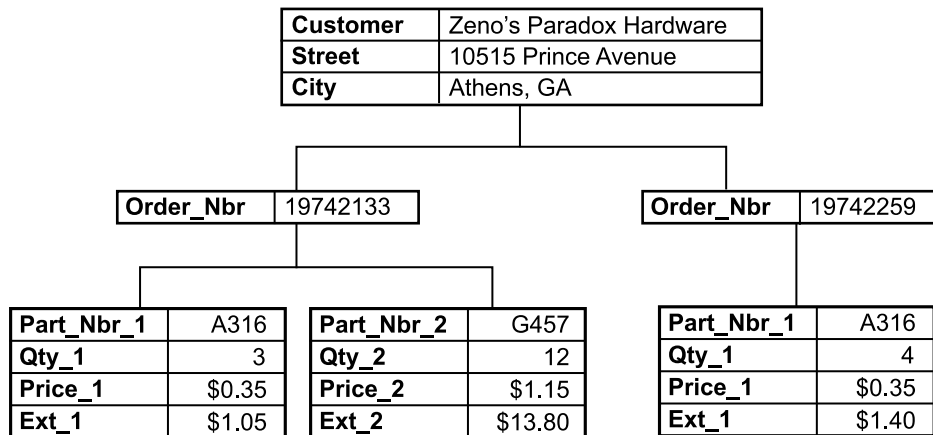
Record	Customer	Street	City	Order_Nbr	Part_Nbr_1	Qty_1	Price_1	Ext_1
122406	Zeno's Paradox Hardware	10515 Prince Avenue	Athens, GA	19742259	A316	4	\$0.35	\$1.40

- Some space is saved by not having empty fields representing Part Number 2 on the second order, but data redundancy has not been entirely eliminated: the customer's address is stored with both orders.
- While variable-length records were an improvement in terms of space, reading such files still involved the inefficient process known as **sequential access**. To find a particular record, every intervening record had to be examined and bypassed.
    - The analogy is a cassette tape; all the intervening songs must be identified and skipped in order to find the desired song. This analogy is apt because much early data storage was on large reels of magnetic tape.



- b. This inefficiency was overcome with the development of the **indexed** sequential access method (ISAM) by IBM.
    - 1) Under this method, each file contains an extra table holding the storage location of every record (every record is said to be “indexed”). When a certain record is desired, the system consults the index table to find where the record is stored. The record can then be retrieved directly without having to examine a lot of unwanted records.
    - 2) ISAM is a very powerful technique, and made the development of the relational database possible (see item 7. on the next page).
  - c. Another major improvement in efficiency came with the advent of disk drives, which can quickly seek out a given storage address. This technique is known as **direct** or **random access**.
    - 1) The analogy is a phonograph record: with a cassette tape, all unwanted songs must be physically bypassed, while with a phonograph, the user can place the needle anywhere (s)he wants.
    - 2) Random access is a necessity for real-time systems.
5. The **hierarchical, or tree, database model** was the next development in file organization. Instead of the records being strung out one after the other, they form “branches” and “leaves” extending from a “root.” Note that the customer’s address is now stored only once.
- a. Another feature of the tree file structure is that every “parent” record can have multiple “child” records, but each child can have only one parent.

EXAMPLE of a tree data structure:



- b. One customer has many orders, but each order can only be assigned to one customer.
    - 1) The tree structure improves speed and storage efficiency for related data; for example, a parent record consisting of a customer may directly index the child records containing the customer’s orders.
    - 2) However, adding new records is much more difficult than with a flat file. In a flat file, a new record is simply inserted whole in the proper place. In a tree structure, the relationships between the parent and child records must be maintained.
6. The **network database model** allowed child records to have multiple parents.
- a. This was an attempt to make queries more efficient, but the huge number of cross-references inherent in this structure made maintenance far too complex.

7. In the **relational database model**, the elements of data “relate” to one another in a highly flexible way.
- What were called tables in earlier data structures are technically referred to as “relations.” Likewise, a table’s columns are called “attributes” and the rows are called “tuples.”
  - Each data element is stored as few times as necessary. This reduction in data redundancy is accomplished through a process called normalization.

EXAMPLE of a relational data structure:

**Customer Table**

Customer_Nbr	Customer	Street	City
X1	Xylophones To Go	3846 N Lamar Blvd	Oxford, MS
Y1	Yellow Dog Software	1012 E Tennessee St	Tallahassee, FL
Z1	Zeno’s Paradox Hardware	10515 Prince Avenue	Athens, GA

**Order Table**

Order_Nbr	Customer_Nbr	Part_Nbr_1	Qty_1	Part_Nbr_2	Qty_2
19742133	Z1	A316	3	G547	12
19742259	Z1	A316	4		

**Parts Table**

Part_Nbr_1	Price
A316	\$0.35
G457	\$1.15

- Two features that make the relational data structure stand out are cardinality and referential integrity.
  - Cardinality** refers to the boundaries of the relationship between certain data elements.
    - For example, the Order Table above cannot contain a record where the quantity ordered has a value of 0 or less nor have a value greater than 500.
  - Referential integrity** means that for a record to be entered in a given table, there must already be a record in some other table(s).
    - For example, the Order Table above cannot contain a record where the part number is not already present in the Parts Table.
- The tremendous advantage of a relational data structure is that searching for records is greatly facilitated.
  - For example, a user can specify a customer and see all the parts that customer has ordered, or the user can specify a part and see all the customers who have ordered it. Such queries were extremely resource-intensive, if not impossible, under older data structures.
- A group of tables built following the principles of relational data structures is referred to as a **relational database**.
  - If the rules of cardinality, referential integrity, etc., are not enforced, a database will no longer be relational. To aid in the exceedingly challenging task of enforcing these rules, database management systems have been developed.