Computer Software

Professor Dr. Md. Ismail Jabiullah
Department of CSE
Daffodil International University

Learning Objectives

- **In this Lecture you will learn about:**

✓ Term "Software" and its relationship with "Hardware"

✓ Various types of software and their examples

✓ Relationship among hardware, system software, application software, and users of a computer system

✓ Different ways of acquiring software

✓ Various steps involved in software development

✓ Firmware

✓ Middleware

Software

- ***Hardware*** refers to the physical devices of a computer system.
- ***Software*** refers to a collection of programs
- ***Program*** is a sequence of instructions written in a language that can be understood by a computer
- ***Software package*** is a group of programs that solve a specific problem or perform a specific type of job

Relationship Between Hardware and Software

- Both hardware and software are necessary for a computer to do useful job. They are complementary to each other
- Same hardware can be loaded with different software to make a computer system perform different types of jobs
- Except for *upgrades*, hardware is normally a one-time expense, whereas software is a continuing expense
- Upgrades refer to renewing or changing components like increasing the main memory, or hard disk capacities, or adding speakers, modems, etc.

Types of Software

- Most software can be divided into two major categories:

✓ ***System software*** are designed to control the operation and extend the processing capability of a computer system

✓ ***Application software*** are designed to solve a specific problem or to do a specific task

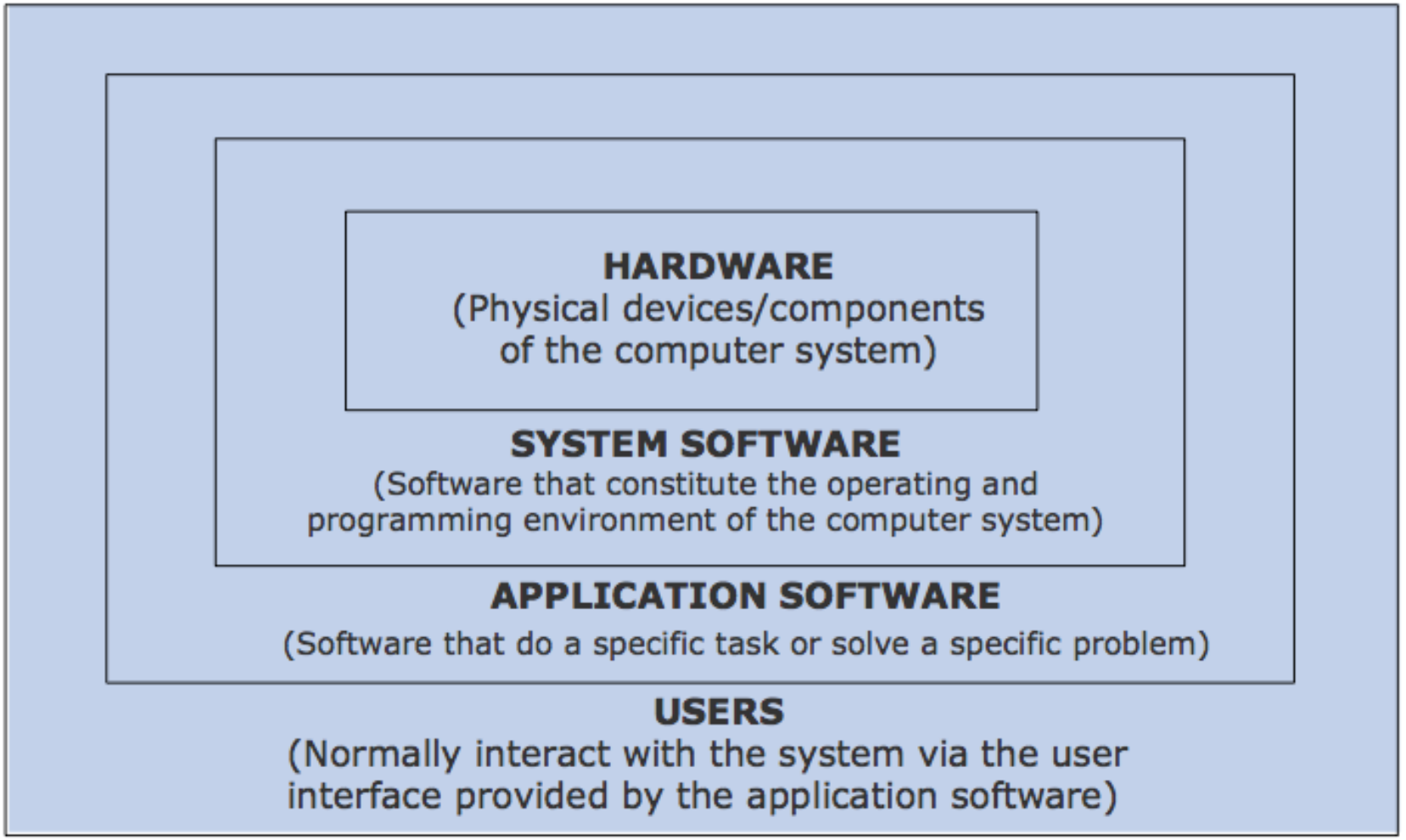
System Software

- Make the operation of a computer system more effective and efficient
- Help hardware components work together and provide support for the development and execution of application software
- Programs included in a system software package are called ***system programs*** and programmers who prepare them are called ***system programmers***
- Examples of system software are operating systems, programming language translators, utility programs, and communications software

Application Software

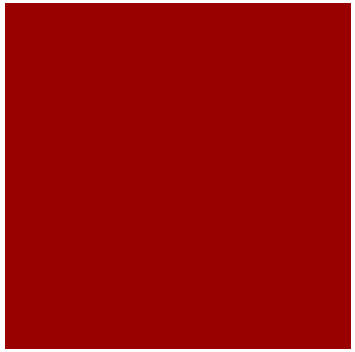
- Solve a specific problem or do a specific task
- Programs included in an application software package are called ***application programs*** and the programmers who prepare them are called ***application programmers***
- Examples of application software are word processing, inventory management, preparation of tax returns, banking, etc.

Logical System Architecture



Relationship among hardware, system software, application software, and users of a computer system.

Ways of Acquiring Software



- Buying pre-written software
- Ordering customized software
- Developing customized software
- Downloading public-domain software
- Each of these ways of acquiring software has its own advantages and limitations

Advantages and Limitations of Buying Pre-written Software

- Usually costs less
- Planned activity can be stared almost immediately
- Often, operating efficiency and the capability to meet specific needs of user more effectively in not as good for pre-written software packages as for in-house developed software packages

Advantages & Limitations of Ordering Customized Software

- User need not maintain its own software development team, which is an expensive affair
- User needs to always depend on the vendor for carrying out the changes and the vendor may separately charge for every request for change

Advantages & Limitations of Developing Customized Software

- Easier to carry out changes in the software, if it is developed in-house
- Developing software in-house means a major commitment of time, money, and resources
- In-house software development team needs to be maintained and managed

Advantage & Limitations of Downloading Public-domain Software

- Available for free or as shareware, and are usually accompanied with source code
- Usually community-supported as author does not support users directly
- Can be downloaded and used immediately
- They may not be properly tested before release
- Open Source Software (OSS) are becoming popular due to:
  - Allows any user to download, view, modify, and redistribute
  - User can fix bugs or change software to suit needs
  - Copyright is protected for both original and subsequent authors
- Not all open source software are free and vise-verse

Software Development Steps

- Developing a software and putting it to use is a complex process and involves following steps:
- Analyzing the problem at hand and planning the program(s) to solve the problem
- Coding the program(s)
- Testing, debugging, and documenting the program(s)
- Implementing the program(s)
- Evaluating and maintaining the program(s)

Firmware

- Firmware is software substituted for hardware and stored in read-only memory
- Firmware technology has enabled production of various types of smart machines having microprocessor chips with embedded software

Middleware

- Basic idea is to have a *separate software layer* to:
  - Act as "glue" between client and server parts of application
  - Provide programming abstraction
  - Mask heterogeneity of underlying network, hardware, and OS
- Encourages *three-tier* software architecture against two-tier popularized by Server-Client architecture

Key Words/Phrases

Application programmers
Application programs
Application software
Computer program
Customized software
Database
Education software
End-to-end solution
Entertainment software
Firmware
Graphics software
Hardware
Middleware
Open Source Software
Personal assistance software
Pre-written software
Public-domain software
Shareware
Software
Software package
Spreadsheet
System programmers
System programs
System software
Turnkey solution
User-supported software
Utilities
Word-processing