Overview of System Programming

By: Prof. Prakash Patel

IT Dept, GIT

Introduction

- Computers are a balanced mix of software and hardware.
- Hardware is just a piece of mechanical device and its functions are being controlled by a compatible software.
- Hardware understands instructions in the form of electronic charge, which is the counterpart of binary language in software programming.
- System programming requires a great degree of hardware awareness. Its goal is to achieve efficient use of available resources.

System Software

- **Computer software**, or simply **software**, refers to the non-tangible components of computers, known as computer programs.
- The term is used to contrast with computer hardware, which denotes the physical tangible components of computers.
- It forms a s/w layer which acts as a intermediary between user and the computer.

Software classification

- Software can be classified into
  - System software:
  - **System software** (or **systems software**) is computer software designed to operate and control the computer hardware and to provide a platform for running application software.

    - System software is collection of software program that perform a variety of functions like IO management, storage management, generation and execution of programs etc.
      - Operating Systems
      - Compiler / Assembler (utility software)
      - Device Drivers

- Application software:
  - Application software is kind of software which is designed for fulfillment specialized user requirement.
    - MS Office
    - Adobe Photoshop

System Software

- Each program in the system software is called a system program.
- It performs various tasks such as editing a program, compiling it and arranging for its execution.
- They also performs other tasks that user can unaware of it like, linking of program, functions libraries, protecting program against interference.

- One characteristic in which most system software differ from application software is machine dependency
  - e.g. assembler translate mnemonic instructions into machine code
  - e.g. compilers must generate machine language code
  - e.g. operating systems are directly concerned with the management of nearly all of the resources of a computing system

The difference between different machine

- Machine code
- Instruction formats
- Addressing mode
- Registers

Goals of System Software

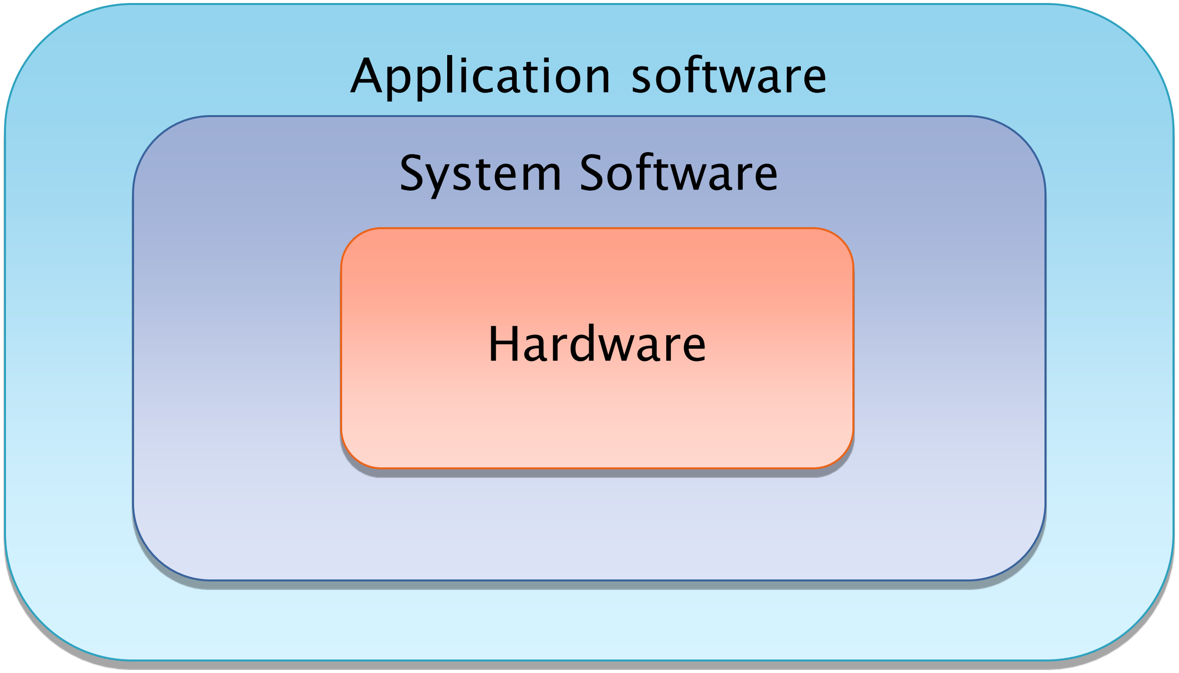
- The fundamental goal of SS are as follows:
  - User convenience: Provide convenient methods of using a computer system
  - Efficient use: Ensure efficient use of computer resources.
  - Non-interference: Prevent interference in the activities of its users.
  - It should be portable
  - It should be evolving and adapting new technologies

System Software

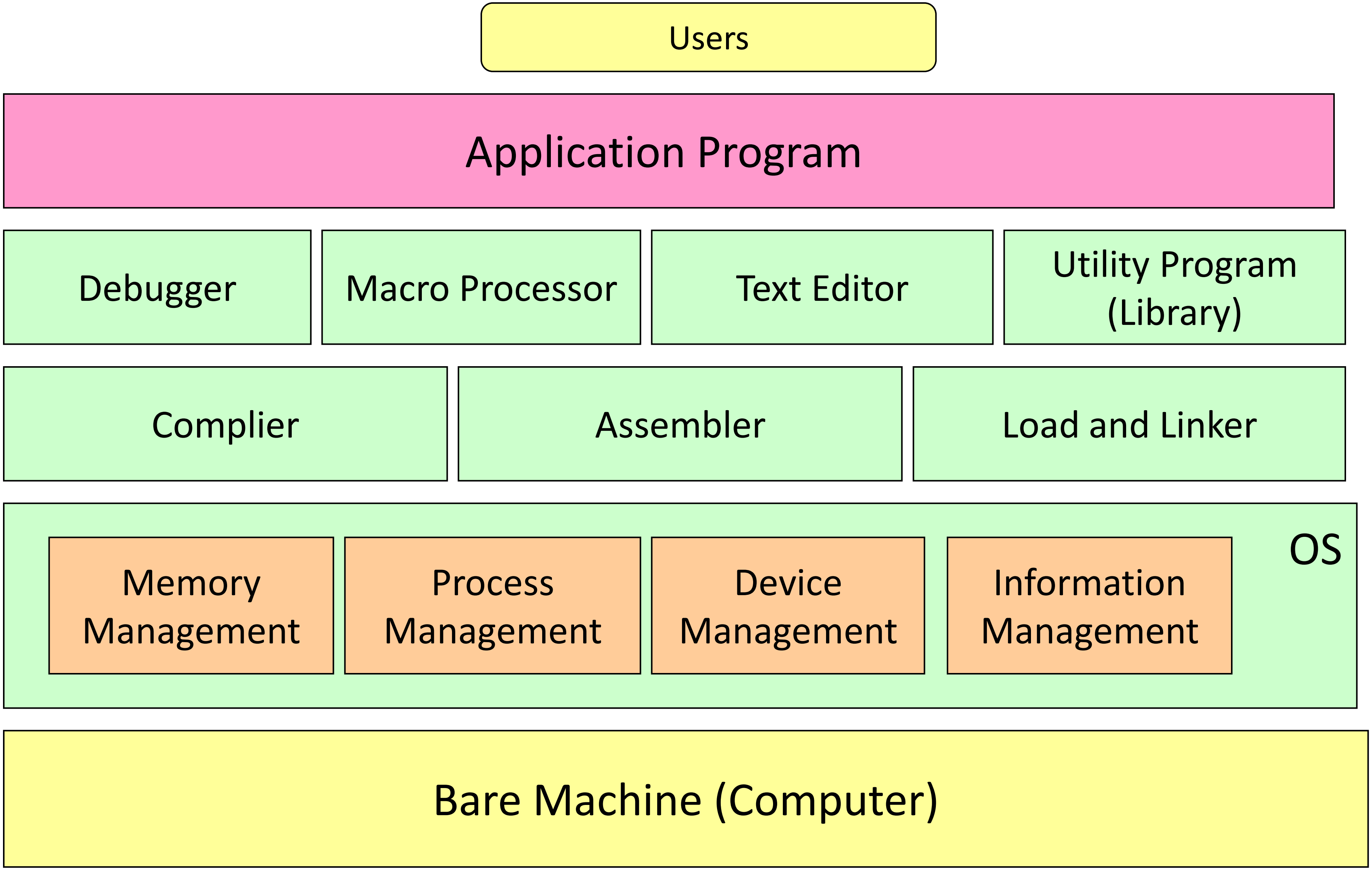
- The system software includes
  - Assembler
  - Linker
  - Loader
  - Macro processor
  - Text editor
  - Compiler
  - Operating system
  - Debugging system
  - Source Code Control System
  - (optional) Database Management System

Cont.

- The system software work as middleware between application software and hardware.



System Software Concept



Users

Application Program

Debugger

Macro Processor

Text Editor

Utility Program (Library)

Complier

Assembler

Load and Linker

OS

Memory Management

Process Management

Device Management

Information Management

Bare Machine (Computer)

Traditional (CISC) Machines

- Complex Instruction Set Computers (CISC)
  - complicated instruction set
  - different instruction formats and lengths
  - many different addressing modes
  - e.g. VAX or PDP-11 from DEC
  - e.g. Intel x86 family
- Reduced Instruction Set Computer (RISC)

RISC Machines

- RISC system
  - instruction
    - standard, fixed instruction format
    - single-cycle execution of most instructions
    - memory access is available only for load and store instruction
    - other instructions are register-to-register operations
    - a small number of machine instructions, and instruction format
  - a large number of general-purpose registers
  - a small number of addressing modes
  - Three RISC machines
    - SPARC family
    - PowerPC family
    - Cray T3E

System Programming

- **System programming** is the activity of programming computer system software.
- The primary distinguishing characteristic of systems programming when compared to application programming is that application programming aims to produce software which provides services to the user directly (e.g. word processor).

Whereas it aims to produce software and software platforms which provide services to other software, are performance constrained, or both (e.g. operating systems, computational science applications, game engines , industrial automation, and software as a service applications).