UNIT I

INTRODUCTION

Introduction

- **Introduction**
  - What Operating Systems Do
  - Computer-System Architecture
  - Operating-System Structure
  - Operating-System Operations
  - **Operating-System Services**
  - **User Operating System Interface**
  - System Calls
  - Types of System Calls
  - System Programs
  - System Boot
- **Process Concept**
  - Process Scheduling
  - Operations on Processes
  - Interprocess Communication

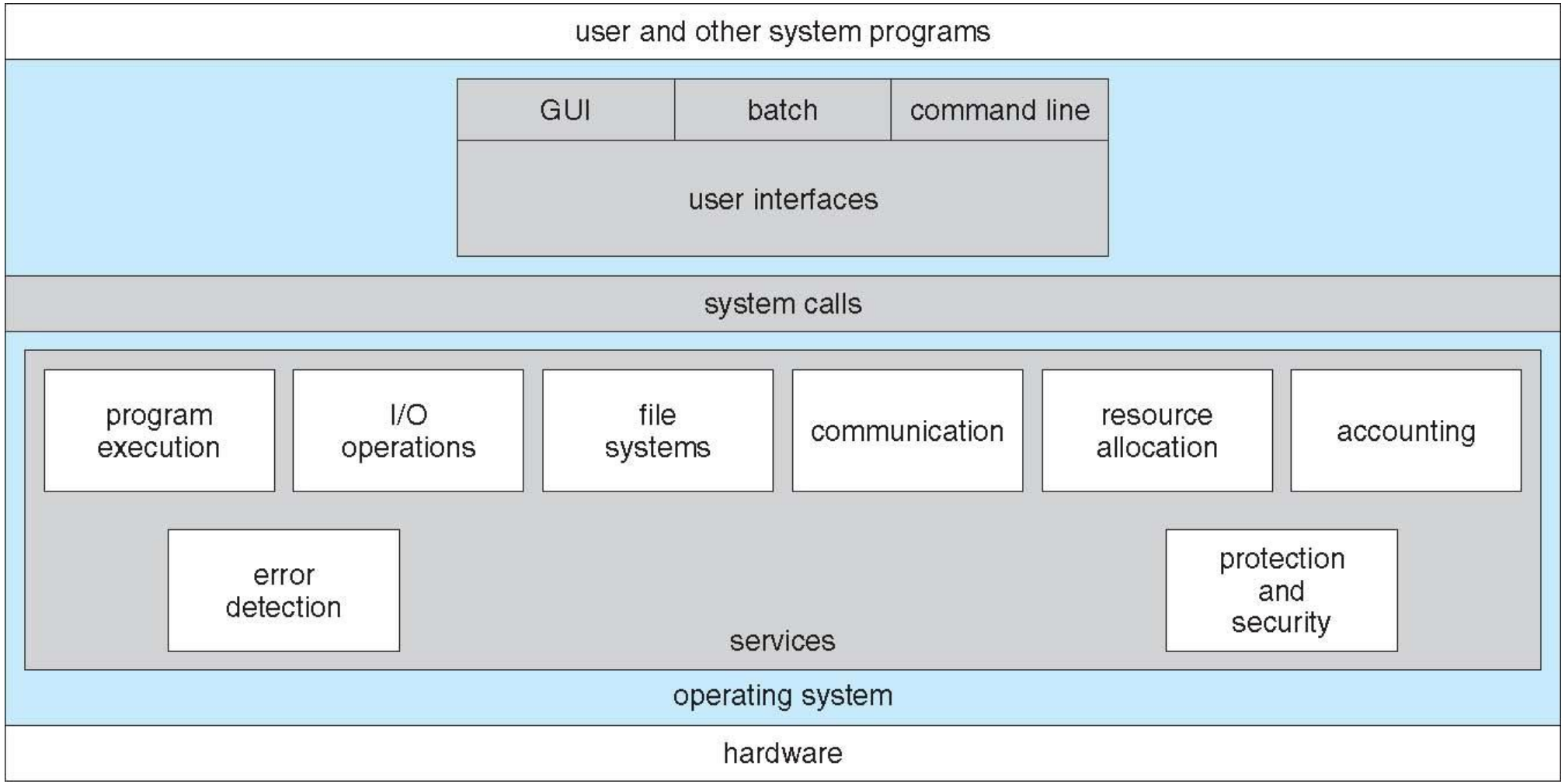
OPERATING SYSTEM SERVICES

- **User interface** - Almost all operating systems have a user interface (**UI**).
- Varies between **Command-Line (CLI)**, **Graphics User Interface (GUI)**, **Batch**
- **Program execution** - The system must be able to load a program into memory and to run that program, end execution, either normally or abnormally (indicating error)
- **I/O operations** - A running program may require I/O, which may involve a file or an I/O device
- **File-system manipulation** - Programs need to read and write files and directories, create and delete them, search them, list file Information, permission management.
- **Error detection** – OS needs to be constantly aware of possible errors

OPERATING-SYSTEM SERVICES

- **Communications** – Processes may exchange information, on the same computer or between computers over a network may be via shared memory or through message passing
- **Resource allocation -** When multiple users or multiple jobs running concurrently, resources must be allocated to each of them - CPU cycles, main memory, file storage, I/O devices.
- **Accounting -** To keep track of which users use how much and what kinds of computer resources
- **Protection and security -** The owners of information stored in a multiuser or networked computer system may want to control use of that information, concurrent processes should not interfere with each other

A View of Operating System Services

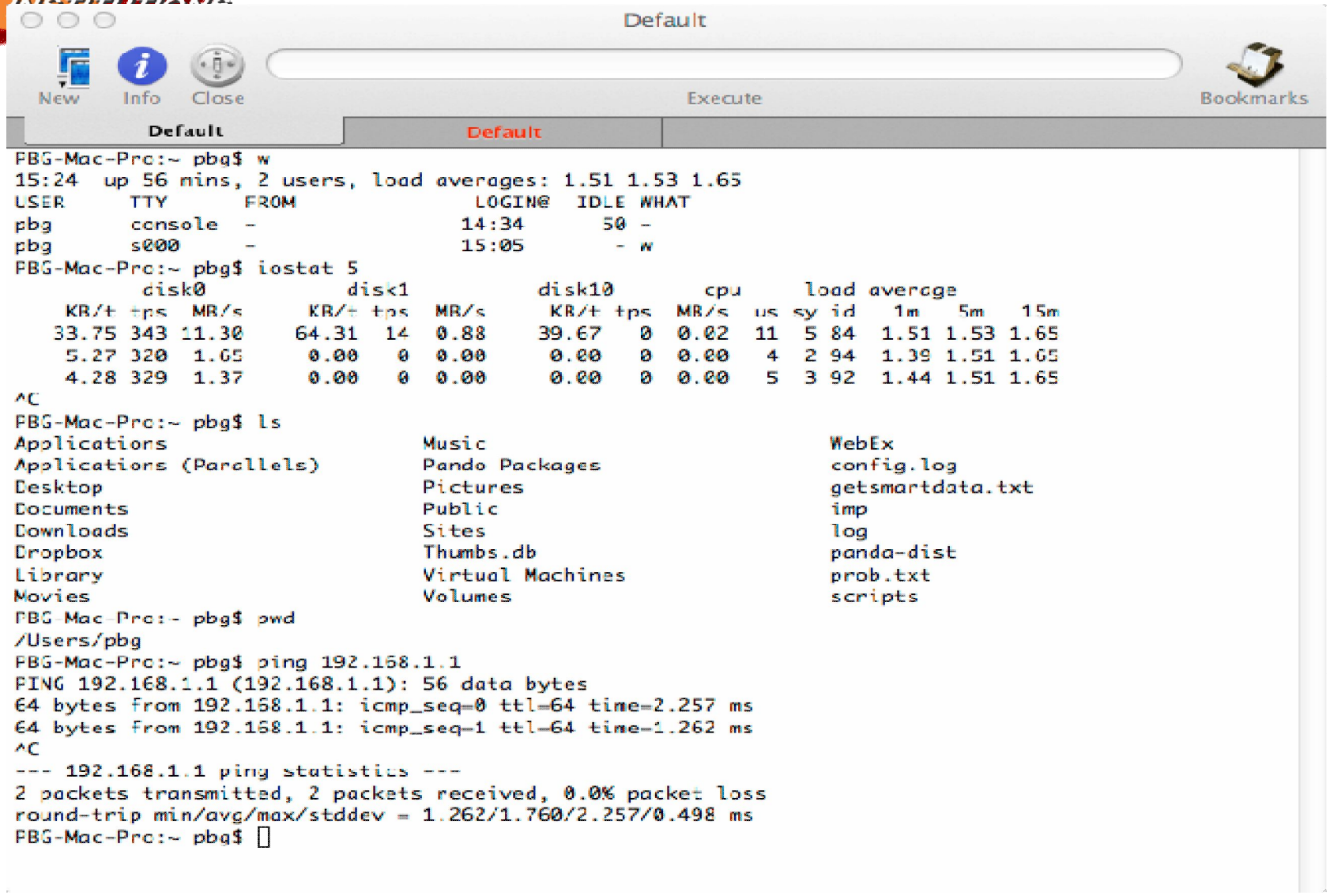


User Operating System Interface - CLI

CLI or **command interpreter** allows direct command entry

- Sometimes implemented in **kernel,** sometimes by **systems program**
- Sometimes multiple flavors implemented – **shells**
- Primarily fetches a command from user and executes it
- Sometimes commands built-in, sometimes just names of programs
  - If the latter, adding new features doesn't require shell modification

Bourne Shell Command Interpreter



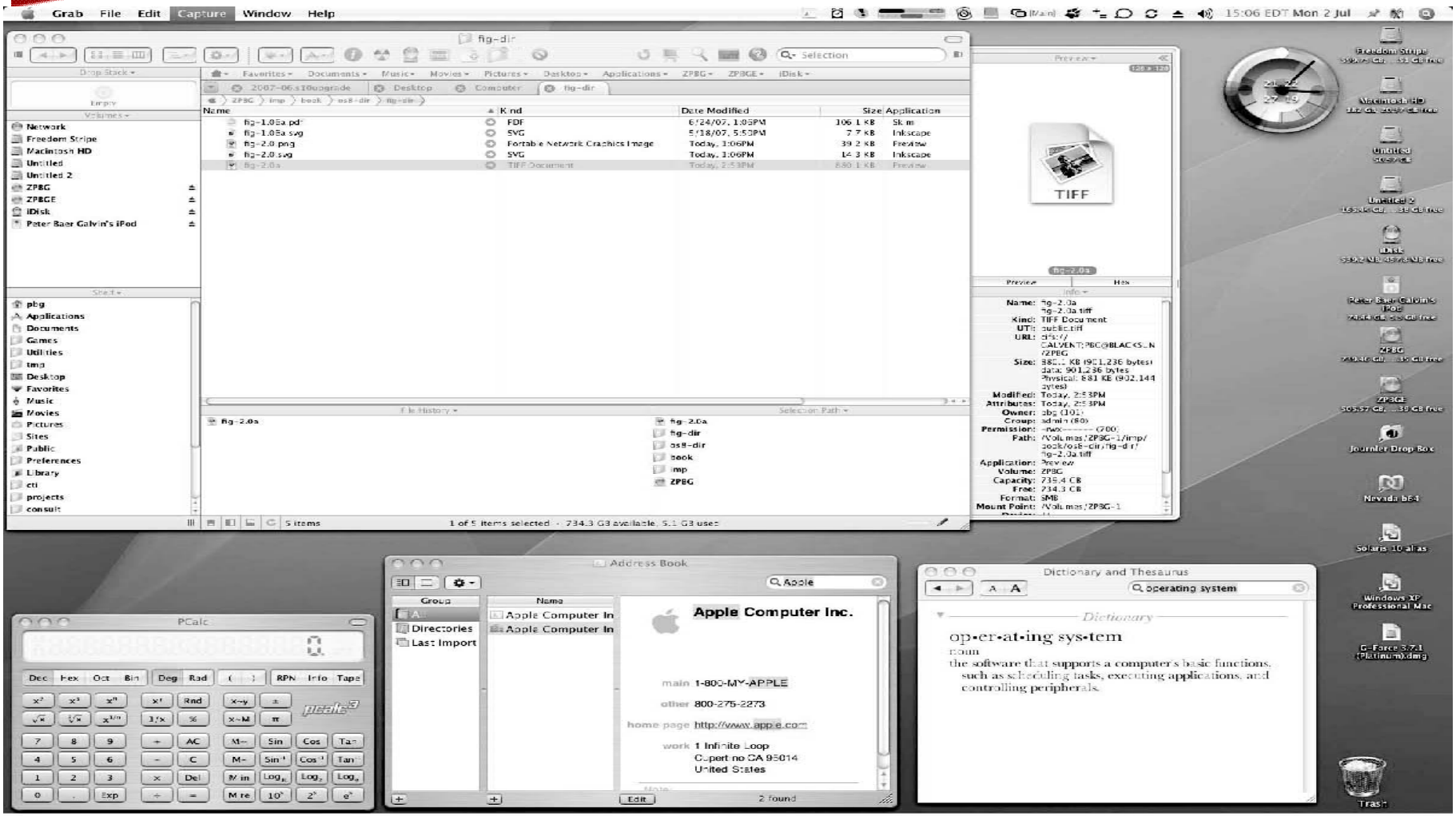
User Operating System Interface - GUI

- User-friendly **desktop** metaphor interface , Invented at Xerox PARC
  - Usually mouse, keyboard, and monitor
  - **Icons** represent files, programs, actions, etc
  - Various mouse buttons over objects in the interface cause various actions
- Many systems now include **both CLI and GUI interfaces**
  - Microsoft Windows is GUI with CLI “command” shell
  - Apple Mac OS X is “Aqua” GUI interface with UNIX kernel underneath and shells available
  - Unix and Linux have CLI with optional GUI interfaces (CDE, KDE, GNOME)

Touchscreen Interfaces

- Touchscreen devices require new interfaces
  - Mouse not possible or not desired
  - Actions and selection based on gestures
  - Virtual keyboard for text entry
- Voice commands.

The Mac OS X GUI



TEXT BOOK

1. Abraham Silberschatz, Peter B. Galvin, “Operating System Concepts”, 10th Edition, John Wiley & Sons, Inc., 2018.

2. Jane W. and S. Liu. “Real-Time Systems”. Prentice Hall of India 2018.

3. Andrew S Tanenbaum, Herbert Bos, Modern Operating Pearson , 2015.

REFERENCES

1. William Stallings, “Operating Systems: Internals and Design Principles”,9th Edition, Prentice Hall of India., 2018.

2. D.M.Dhamdhere, “Operating Systems: A Concept based Approach”, 3rdEdition, Tata McGraw hill 2016.

3. P.C.Bhatt, “An Introduction to Operating Systems–Concepts and Practice",4th Edition, Prentice Hall of India., 2013.

THANK YOU