

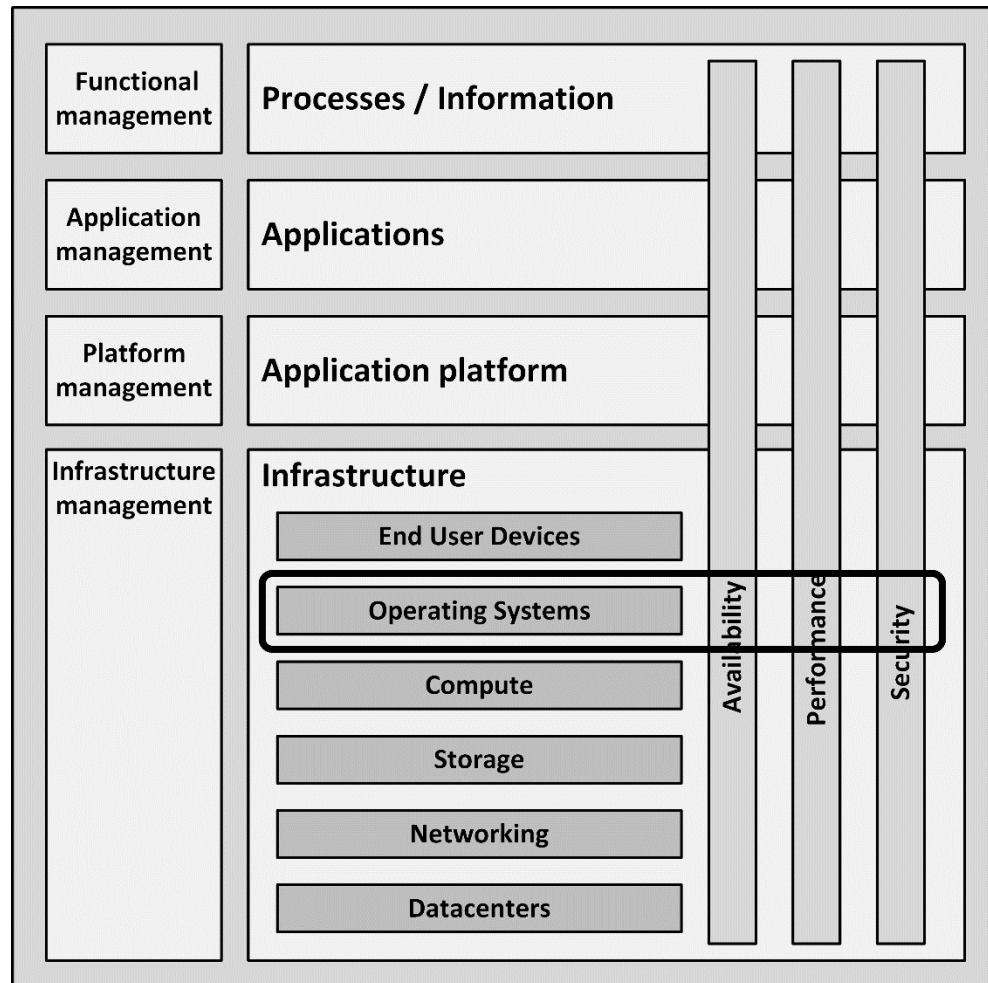
IT Infrastructure Architecture

Infrastructure Building Blocks
and Concepts

Operating Systems

Introduction

- An operating system is the set of programs that, after being initially loaded into a computer by a boot program, controls all the other programs in a computer



Introduction

- Operating systems manage a computer's internal workings
 - Memory
 - Processors
 - Internal and peripheral devices
 - File system
- Operating systems are used on for instance::
 - PCs and laptops
 - Virtual machines
 - Servers
 - Tablets and mobile phones
 - Network routers
 - Storage arrays
 - Cars
 - Televisions

Introduction

- Operating systems provide an abstraction layer between (virtualized or physical) hardware and software applications
- As a service to applications, low level hardware management is handled by the operating system
 - Process management
 - Memory management
 - Interrupt handling
 - Multi user management
 - File locking and file sharing

Introduction

- Operating systems provide services to applications in the form of Application Programming Interfaces (APIs)
- For example for:
 - File management
 - I/O interfaces (like video and keyboard)
 - Hardware drivers (like printer drivers)

History

- Early operating systems could execute one program at a time
- At any given time, one user had sole use of the computer
- Through the 1950s, many major features were pioneered in the field of operating systems, including multitasking
- During the 1960s, IBM introduced a single operating system (OS/360) for all of its mainframes
- In 1969, UNIX was created
 - UNIX used a file system with directories
 - Standard POSIX system calls

History

- An early operating systems for personal computers was CP/M
 - CP/M introduced drive letters (c:), but no directories
- Parts of CP/M and UNIX were imitated in MS-DOS
 - Drive letters and directory structures
 - System calls
 - MS-DOS became extremely popular when chosen as the default operating system for the IBM PC
- The most popular operating systems running on servers today are Microsoft Windows, Linux, and UNIX

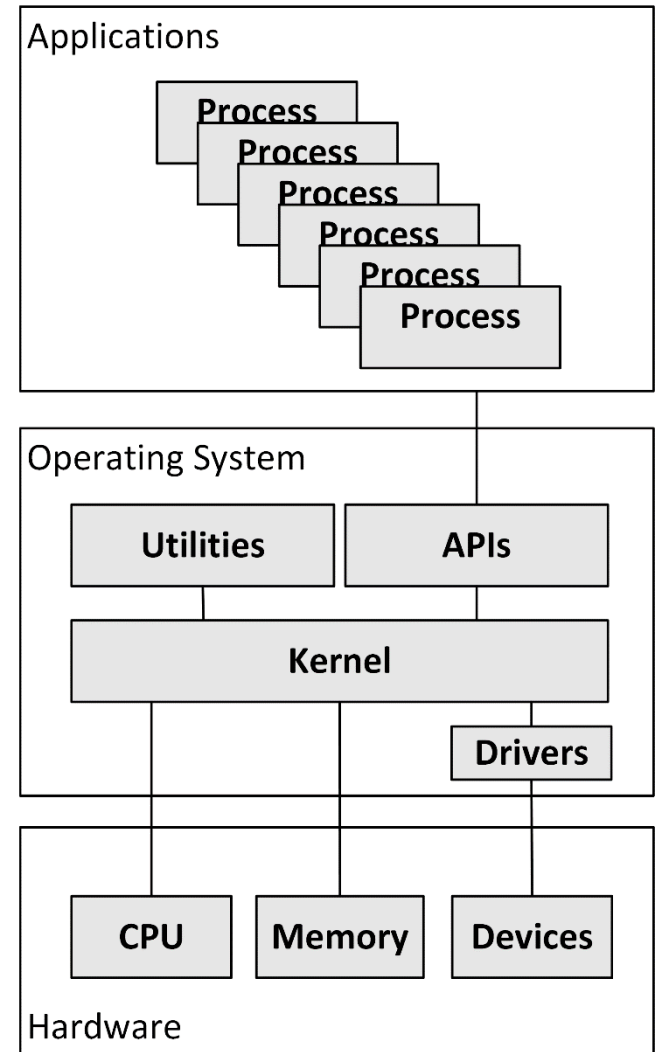
Operating system building blocks

Operating System building blocks

- An operating system basically performs two basic operations:
 - It enables multiple users, multiple processes, and multiple applications to run together on a single piece of hardware
 - It hides the technical complexities of the underlying hardware from the applications running on top of the operating system

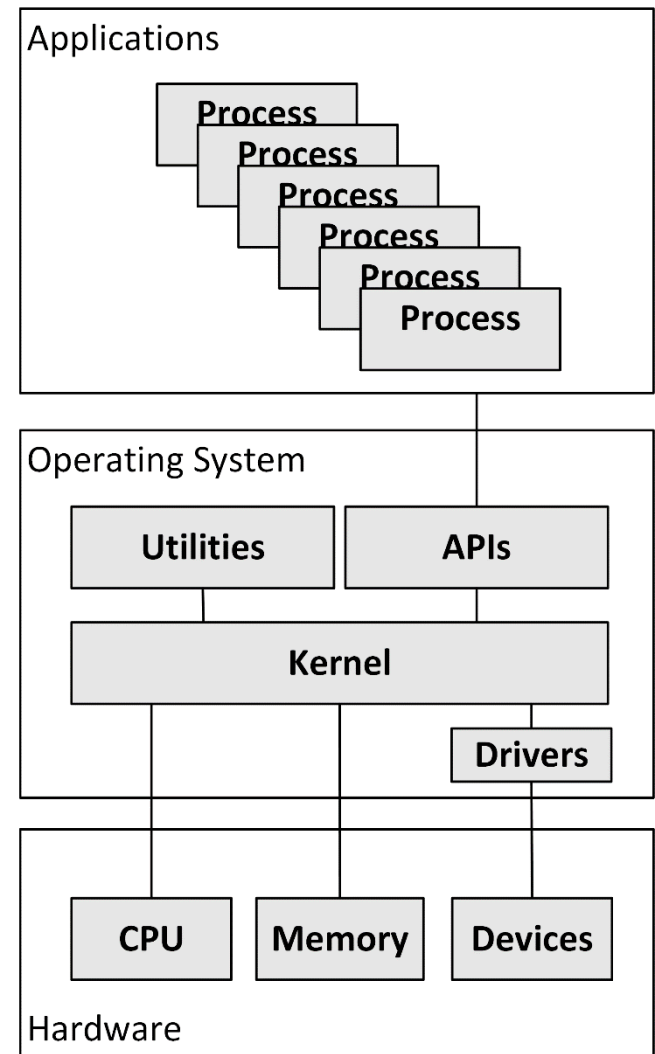
Operating System building blocks

- The kernel is the heart of an operating system
 - Starts and stops programs
 - Manages the file system
 - Performs low level tasks that most programs need
 - The kernel schedules access to hardware to avoid conflicts if two programs try to access the same resource or device simultaneously



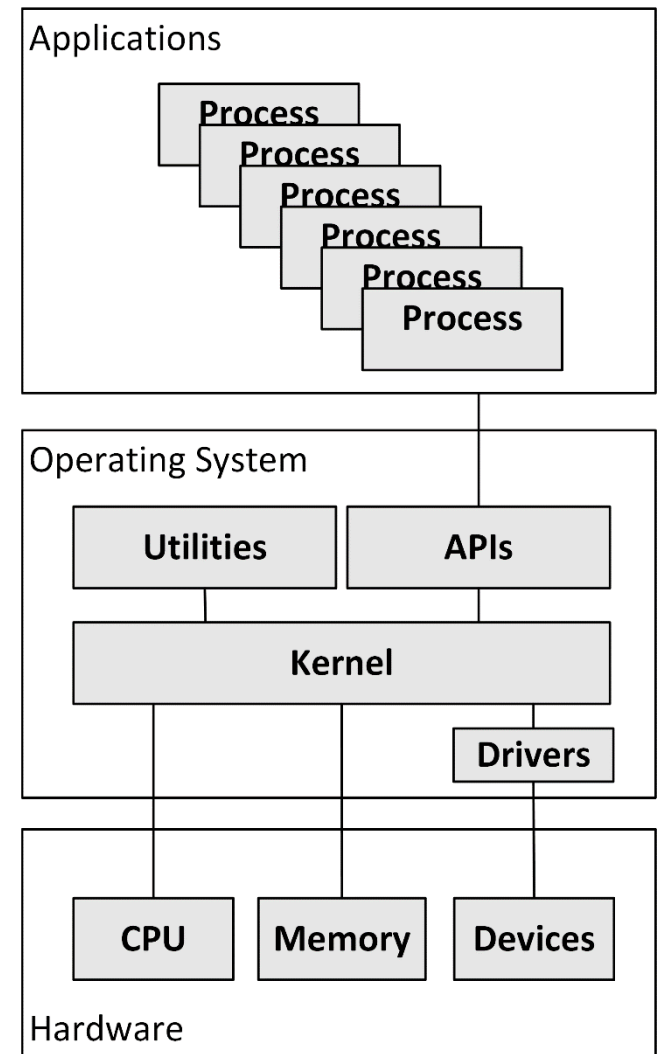
Operating System building blocks

- Drivers are small applications that connect specific hardware devices to the kernel
 - Printers
 - Network cards
 - Keyboard and mouse
 - Video screens
- Utilities are applications that are considered part of the operating system
 - User interfaces
 - Logging tools
 - Editors
 - System update processes



Operating System building blocks

- Applications consist of one or more processes that communicate with the operating system using system calls that are invoked through Application Programming Interfaces (APIs)



Process scheduling

- Operating systems create the illusion of multiple running processes in parallel by scheduling each process to run only during a short time frame
 - This principle is also known as preemptive multitasking
 - Periodically, the operating system decides if a running process is to be suspended in favor of another process, or if the running process can keep on running for a while
- Process scheduling is fairly complex
 - Must be well-balanced
 - Switching processes introduces some overhead
 - The scheduling algorithm guarantees each process gets its fair share of CPU time
 - Because operating systems have evolved over decades, scheduling algorithms are very sophisticated

File systems

- The operating system provides a file system to applications
 - File systems usually consist of directories (also known as folders) with files or other directories
- The operating system hides complexity
 - Handling individual disk blocks or communication with a SAN or NAS
 - Managing the files and the directory structure
 - Security – permission to read, write, create, and delete files and directories

File systems

- Most operating systems can handle multiple types of file systems on multiple disks at the same time
- Some popular file systems are:
 - FAT (File Allocation Table), vFAT, and FAT32, used in MS-DOS, older versions of Windows, and removable storage devices like USB memory sticks
 - NTFS (New Technology File System) used in Windows
 - UFS (Universal File System) and VxFS (Veritas File System) used in most UNIX flavors
 - Ext (and Ext2, Ext3, Ext4) - used in Linux

File systems

- Journaling file systems keep track of changes made to files in a journal log before committing them to the main file system
 - Higher availability
 - Fast recovery in case of a malfunction
- File systems must be mounted before they can be used by the operating system
 - A disk and the file system on it must be recognized by the operating system and attached to it
- After mounting, the file system is typically given either:
 - A drive letter (Windows)
 - A drive name (OpenVMS)
 - A mount point in the global directory tree (UNIX and Linux)

File systems

- Most operating systems provide file sharing functionality
 - File sharing enables files on one system to be accessed by (users on) other systems
 - File sharing protocols:
 - NFS: originates from UNIX
 - SMB/CIFS: originates from Windows

APIs and system calls

- System calls are programming functions
 - Provide a hardware-independent interface to tasks the operating system can perform for applications
- Example:

```
int read(int handle, void *buffer, int  
nbyte) ;
```

translates into the system call

```
READ(FILEHANDLE, DESTINATION DATA  
POINTER, NUMBER_OF_BYTES)
```

APIs and system calls

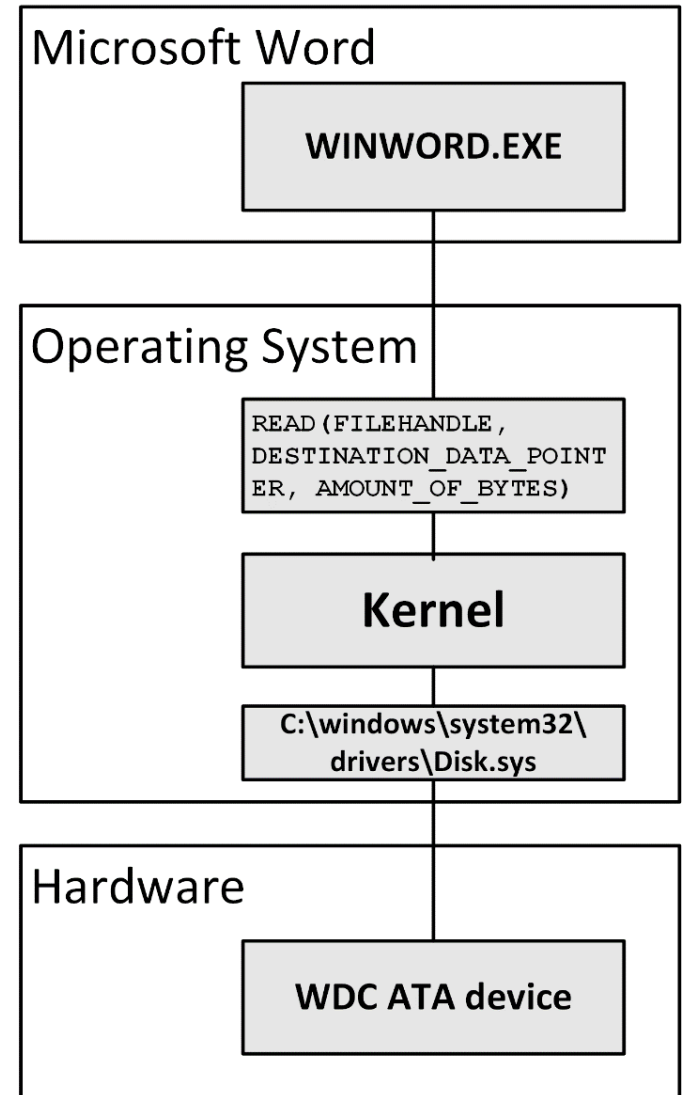
- The operating system takes care of:
 - Looking-up the file in a file allocation table
 - Looking up the disk blocks on disk
 - Instructing the disk controller to fetch the needed disk blocks
 - Copy the disk blocks to memory
 - Providing a pointer to the disk blocks in memory

APIs and system calls

- System calls are grouped and presented to application processes as Application Programming Interfaces (APIs)
- APIs describe the available system calls in an operating system and how they can be used by programmers in their applications
- Each operating system has its own API
 - UNIX and Linux use the POSIX standard
 - Windows has its own API

Device drivers

- The operating system manages all hardware
- I/O devices are controlled using device drivers
 - Pieces of software that interact with the device's hardware (like an Ethernet network adapter or a SAS disk adapter)
 - They provide an Application Programming Interface (API) to the operating system



Memory management

- The operating system:
 - Allocates and de-allocates memory on behalf of applications
 - Manages what happens when the amount of requested memory exceeds the physical amount of memory
- Memory management includes:
 - Cache management
 - Paging
 - High volume data transfers
 - Memory management units (MMUs)
 - Thin memory provisioning (memory overcommitting)
 - Direct Memory Access (DMA)
- The operating system takes care of all of this and just provides chunks of memory to applications

Shells, CLIs and GUIs

- A shell provides a user interface to the operating system
 - The primary purpose of shells is to launch other programs, started by end users or scripts
- Two types of shells:
 - Command-Line Interfaces (CLIs)
 - The user types commands on a keyboard on a command-prompt
 - Examples: UNIX shells (bash, sh, csh) and Windows' cmd.exe (also known as a DOS box)
 - Graphical User Interfaces (GUIs)
 - The user uses a mouse to click on icons or buttons
 - Examples Microsoft Windows and X Windows (UNIX and Linux)

Operating system configuration

- The configuration of an operating system is stored in an operating system specific database or in text files
- Examples:
 - Windows registry
 - Files in the Linux `/etc` directory
 - AIX Object Data Manager (ODM) database
- For most used configuration parameters, user-friendly tools are provided
 - These tools still edit the text files, but that is hidden from the user

Popular operating systems

IBM z/OS

- The most used mainframe operating system
- Extreme backward compatibility is one of z/OS's main design philosophies
 - Programs written for MVS in 1974 can still run on today's z/OS without modification
- Typical use of z/OS:
 - Batch processing: reading and writing large amounts of data and performing relatively simple calculations on it
 - Interactive users: supports thousands of interactive users

IBM i (OS/400)

- IBM i is an operating system only used on IBM's Power Systems midrange systems
 - The operating system was previously known as OS/400
 - The midrange system was previously known as AS/400
- One of the biggest advantages of IBM i is its completeness, it includes:
 - Communications
 - Transaction processing
 - Relational database manager
 - Features for the implementation and maintenance of data security
- The latest version is "IBM i 7.2", released in 2014

OpenVMS

- OpenVMS is an operating system developed by DEC
 - VMS means Virtual Memory System
 - OpenVMS is now maintained by VMS Software, Inc.
- OpenVMS is a multi-user operating system designed for:
 - Time sharing
 - Batch processing
 - Real-time processing
 - Transaction processing
- OpenVMS is not open source software
 - The source listings are available for purchase

OpenVMS

- OpenVMS is typically used in environments where system uptime and data access is critical
- It is used for various applications, including:
 - Mail servers
 - Network services
 - Manufacturing
 - Transportation control and monitoring
 - Critical applications
 - Databases
- OpenVMS for HP Integrity servers based on Intel Itanium 9500 processors was released in 2015

UNIX - History

- UNIX is a multitasking, multi-user operating system
- Created by Bell Labs (now AT&T) in 1969
- In 1973, UNIX was rewritten in the new C programming language
 - C was created by the same people that created UNIX
 - This made UNIX portable to multiple types of computer hardware
- In 1982, UNIX was licensed to a number of computer manufacturers
 - Most marketed their own UNIX versions based on the original UNIX source code
 - They adapted the code to meet their own hardware and software requirements

UNIX – vendor versions

- Vendors came up with different names for their UNIX flavors
- These versions are 90% the same, but have some minor differences
 - Wording of error messages
 - The order of commands used to start-up the machine
 - The location of certain files

Vendor	UNIX flavor
IBM	AIX
Oracle/Sun	Solaris
HP	HP-UX
Apple	Mac OS X (built on FreeBSD, discussed in the next section)

UNIX – vendor versions

- Each flavor needs specific hardware
 - HP-UX only runs on HP Integrity systems
 - HP systems cannot run for example IBM's AIX
- Applications running on a particular flavor of UNIX cannot run on another flavor without (at least) recompiling
 - Software vendors must provide separate versions of their applications for each flavor of UNIX

UNIX – vendor versions

- UNIX has been ported to a wider variety of machine architectures than any other operating system
 - UNIX is written almost entirely in the C programming language
 - Source code is published
- Many business software today is released on Linux before being released on the various flavors of UNIX
 - Linux runs on many hardware platforms without recompiling
 - Linux is now more widely used than UNIX

UNIX - filesystem

- UNIX popularized the hierarchical file system with nested subdirectories – the directory tree
- All files and directories appear under the so-called root directory "/"
 - Even if they are stored on different physical disks
- UNIX has no concept of drive letters, like Windows or DOS
 - Drives are mounted on a branch in the directory tree, providing disk space for that particular branch
 - Also removable drives must be mounted in the tree

UNIX – system tools

- The UNIX philosophy is to use a large set of small tools that do only one thing, and do it very well
- To perform complicated tasks, commands can be combined using a system called pipes
 - Pipes feed the output of one command to the input of another command, without storing the intermediate result
 - For instance, the UNIX command: `ls | sort` prints a sorted list of files on the screen
 - The pipe sign “|” sends the output of the “ls” command as input to the “sort” command
- In practice, these chains of piped commands can get very long and complex

Linux - history

- Linux is a free UNIX-like operating system for the x86 platform
- It is not derived from the UNIX source code
- In 1991, Linus Torvalds wanted to explore the multitasking possibilities of the new Intel 80386 CPU in his PC
- He decided to create a small multitasking, multi-user operating system with the help of the internet community
- Because of the open source nature of Linux many developers contributed
 - Kernel patches
 - Device drivers
 - Multilingual keyboards
 - Disk drivers
 - Support for video card devices

Linux

- Linux commands and scripts are almost similar to those of UNIX
- Linux also uses the same commands, file structure, scripting language, pipes, etc. as UNIX
- Today Linux is a very mature operating system, used in:
 - Servers
 - Workstations
 - Mobile devices
 - All Android smartphones
 - Appliances like set-top boxes, firewalls and NAS devices
- Ninety-five per cent of the supercomputers listed in the top 500 list of the fastest computers in the world are running Linux
- Almost all internet services run on Linux

Linux – GNU/Linux

- The GNU project was launched in 1984 by Richard Stallman
 - Goal was to develop a free UNIX-like operating system
 - GNU is a recursive acronym for “GNU's Not UNIX!”
 - By 1990, the GNU project had recreated all the major components of the UNIX-like system except one – the kernel
- Combining Linux with the GNU system resulted in a complete operating system: the GNU/Linux system
 - It is important to understand that Linux is actually only an operating system *kernel*
 - Wat we call Linux, is actually GNU/Linux
- Linux and the GNU tools are licensed under the GNU General Public License
 - Ensures all source code will be free for all to copy, study, and to change

Linux – distributions

- Vendors compiled the Linux source code, added some tools and configurations of their own, and releasing it in a distributable format
- Some of the best-known Linux distributions:
 - Red Hat
 - SuSe
 - Ubuntu
 - Debian

Linux - Support

- Linux can be downloaded from the internet for free
- Most organizations demand professional support for their software
- Professional support is not free
- Most Linux distribution vendors, like Red Hat and SuSe, and some independent vendors, offer support contracts for Linux

Linux on mainframes

- Some Linux distributions can be used on IBM mainframes, running in virtual machines
- Linux uses X-Windows emulators on PCs to connect to the mainframe
- The emulated LAN on the mainframe can be used to connect multiple Linux virtual machines and to provide an external LAN route

BSD

- Berkeley Software Distribution (BSD) is a UNIX operating system derivative
 - Developed and distributed by the University of California, Berkeley
- BSD was the basis for three open source development projects:
 - FreeBSD
 - Most widely used
 - FreeBSD is a complete operating system (Linux is only a kernel!)
 - NetBSD
 - Ported to 57 hardware platforms across 15 different processor architectures
 - Often used in embedded systems

BSD

– OpenBSD

- Most secure BSD version
 - Has a tradition in which developers audit the source code for software bugs and security problems
 - In the 10+ years of its existence, only three security bugs have been found in OpenBSD
- Darwin, the system on which Apple's Mac OS X is built, is a derivative of FreeBSD