Outline

1. Basic Concepts
2. Kernel Architecture
3. GNU/Linux
4. Summary
Abstract View of A Computer System
Concrete View of A Computer systems
Computer System Components

1. Hardware - provides basic computing resources (CPU, memory, I/O devices).
2. **Operating system** - controls and coordinates the use of the hardware among the various application programs for the various users.
3. Applications programs - define the ways in which the system resources are used to solve the computing problems of the users (compilers, database systems, video games, business programs).
4. Users (people, machines, other computers).
An operating system is the "permanent" software that controls/abstracts hardware.

User applications can thus be simpler and device-independent.
Importance of Operating Systems

- System API are the **only** interface between user applications and hardware
  - API are designed for general-purpose, not performance driven (e.g. network applications)
- OS code cannot allow any bug
  - Any break (e.g. invalid access) causes reboot
- The owner of OS technology **controls** the software & hardware industry
- Operating systems and computer architecture influence each other (Wintel)
Programmer Perspective

- User program
- Executable binary
- Compiler
- Linker
- System library
- Operating system
- Device drivers
- Architecture
- User
- Compiler
- OS interface
- OS
- Hardware
Outline

1. Basic Concepts
2. Kernel Architecture
3. GNU/Linux
4. Summary
OS Kernel Approaches

- Integrated Kernel Approach
  - "Monolithic kernel"
  - Flat function calling
  - UnixWare, Solaris, AIX, HP-UX, DG-UX, IRIX, Linux, etc

- Microkernel Approach
  - Serverization based on MK
  - Message passing
  - Cray UNICOS/mk, CMU Mach
  - Chorus, Digital UNIX, ETRI MISIX, Linux Mk, etc
2-layered OS Architecture
3-layered OS Architecture

Processes

Operating System

Kernel

Hardware
3-layered with Microkernel OS

Processes
- Applications
  - Graphics Subsystems
  - I/O Manager

Processes
- Network Drivers
- Device Drivers
- Graphics Drivers

Hardware
4-layered OS Architecture: android
Modern operating systems

- x86 platform
  - Linux
  - Windows
- Mac platform - Mac i-OS
- Oracle/Sun Solaris & Unix (Sun workstations)
- IBM AIX (mainframes)
- Embedded systems (special-purpose OS)
  - Palm (Handheld), PocketPC (Windows CE), etc
  - Xbox, etc
  - Android (Linux)
System Category

- Mainframe Systems
- Desktop Systems
- Parallel Systems
- Distributed Systems
- Real-Time Systems
- Handheld Systems
Migration of Operating-System Concepts and Features

source: OPERATING SYSTEM CONCEPTS Sixth Edition
Case Study: Unix Architecture

- Typical computer system consists of:
  - Hardware
  - Operating system
  - Applications and utilities
- Typical OS includes a user interface (and programmer interface) or command interpreter (the shell)
Case Study: Unix Architecture
Case Study: Unix Functional Layer Model
Dennis Ritchie

An American computer scientist who "helped shape the digital era." He created the C programming language and, with long-time colleague Ken Thompson, the Unix operating system. Ritchie and Thompson received the Turing Award from the ACM in 1983, the Hamming Medal from the IEEE in 1990 and the National Medal of Technology from President Clinton in 1999. Ritchie was the head of Lucent Technologies System Software Research Department when he retired in 2007. He was the 'R' in K&R C and commonly known by his username dmr....

http://en.wikipedia.org/wiki/Dennis_Ritchie

- September 9, 1941 – October 12, 2011
- Computer Scientist
- One of the greatest hackers
- Turing Award...
An American pioneer of computer science. Having worked at Bell Labs for most of his career, Thompson is notable for his work with the B programming language (basing it mainly on the BCPL language he had used to write Unix while in the MULTICS project), the C programming language, and as one of the creators and early developers of the Unix and Plan 9 operating systems. Other notable contributions included his work on regular expressions and early computer text editors QED and ed, his work on computer chess that included creation of endgame tablebases and the chess machine Belle, and most recently the co-creation of Google’s programming language Go...
http://en.wikipedia.org/wiki/Ken_Thompson

- Born February 4, 1943
- Computer Scientist
- One of the greatest hackers
- Turing Award...
What were real hackers doing at 1980s

"After 20 years, this is still the best exposition of the workings of a 'real' operating system."
Ken Thompson

Lions’ Commentary on UNIX®
6th Edition
with Source Code
John Lions

Foreword by Dennis Ritchie
Richard Stallman

In September 1983, he launched the GNU Project to create a free Unix-like operating system, and he has been the project’s lead architect and organizer. With the launch of the GNU Project, he initiated the free software movement; in October 1985 he founded the Free Software Foundation... http://en.wikipedia.org/wiki/Richard_Stallman

- Born March 16, 1953
- Software freedom activist and a programmer
- One of the greatest hackers
Andrew Stuart "Andy" Tanenbaum (sometimes referred to by the handle ast) (born March 16, 1944) is a professor of computer science at the Vrije Universiteit, Amsterdam in the Netherlands. He is best known as the author of MINIX, a free Unix-like operating system for teaching purposes, and for his computer science textbooks, regarded as standard texts in the field. He regards his teaching job as his most important work...http://en.wikipedia.org/wiki/Andrew_S._Tanenbaum

- Born March 16, 1944
- A Professor
- ACM/IEEE/Fellow,....
He is best known for having initiated the development of the open source Linux kernel. He later became the chief architect of the Linux kernel, and now acts as the project’s coordinator. He also created the revision control system Git as well as the Scuba diving log book software Subsurface...http://en.wikipedia.org/wiki/Linus_Torvalds

- Born December 28, 1969
- Software engineer
- One of the greatest hackers
The GNU Project

http://www.gnu.org/

GNU’s Not Unix
Birth of Linux
Linux is a Unix clone written from scratch by Linus Torvalds with assistance from a loosely-knit team of hackers across the Net. It aims towards POSIX compliance.

It has all the features you would expect in a modern fully-fledged Unix, including true multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables, proper memory management and TCP/IP networking.

It is distributed under the GNU General Public License.

Linux was first developed for 386/486-based PCs. These days it also runs on ARMs, DEC Alphas, SUN Sparcs, M68000 machines (like Atari and Amiga), MIPS and PowerPC, and others.
Linux Architecture
## Summary

### Operating System
- Core technique in Information Technology
- Windows/*Unix*

### Why GNU/Linux
- UNIX/INTERNET/GNU/MINIX/POSIX/
- A great opportunity to digging the kernel details
References

1. https://netfiles.uiuc.edu/rhasan/linux/