

# 시스템 보안

## #2 운영체제의 이해 - 1



- **목차**

- 운영체제 개요
- 운영체제의 종류
- 운영체제의 유형
- 운영체제의 기능
- 운영체제의 구성

# 운영체제 개요

- 시스템 보안

- 시스템 보안에서의 시스템 = 컴퓨터 시스템

- 시스템을 보호하려면 보호 대상인 컴퓨터 시스템(호스트 컴퓨터나 단말기 등)을 알아야 함

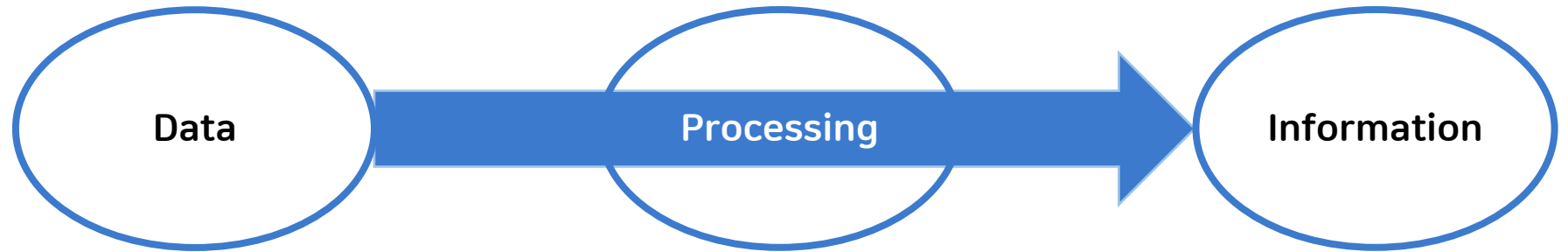
- 시스템에 대한 불법 침투, 부정 접속, 신분 위장, 자료 삭제 및 유출 등의 공격에 대한 방어를 목적으로 함
  - 각종 해킹으로부터 시스템을 보호하기 위해 자원에 대한 불법적인 수정이나 참조를 방지해야 함

- 해킹이나 바이러스와 같은 공격에 대해 보안 서비스를 제공하며, 주로 운영체제 보안과 관련되어 있음

# 운영체제 개요

- 컴퓨터 시스템이란?

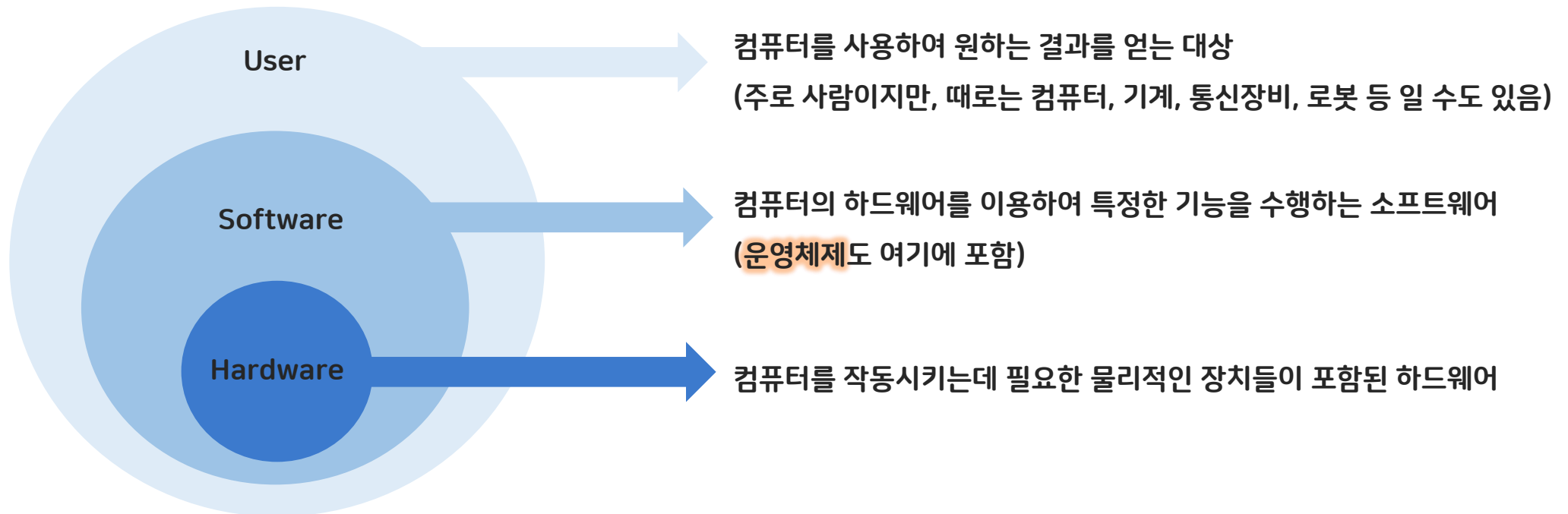
- 자료(Data)를 입력하고 프로그램에 의한 자료의 가공 처리(Processing)로 정보를 생성하고 정보(Information)를 출력하는 시스템



# 운영체제 개요

- 컴퓨터 시스템이란?

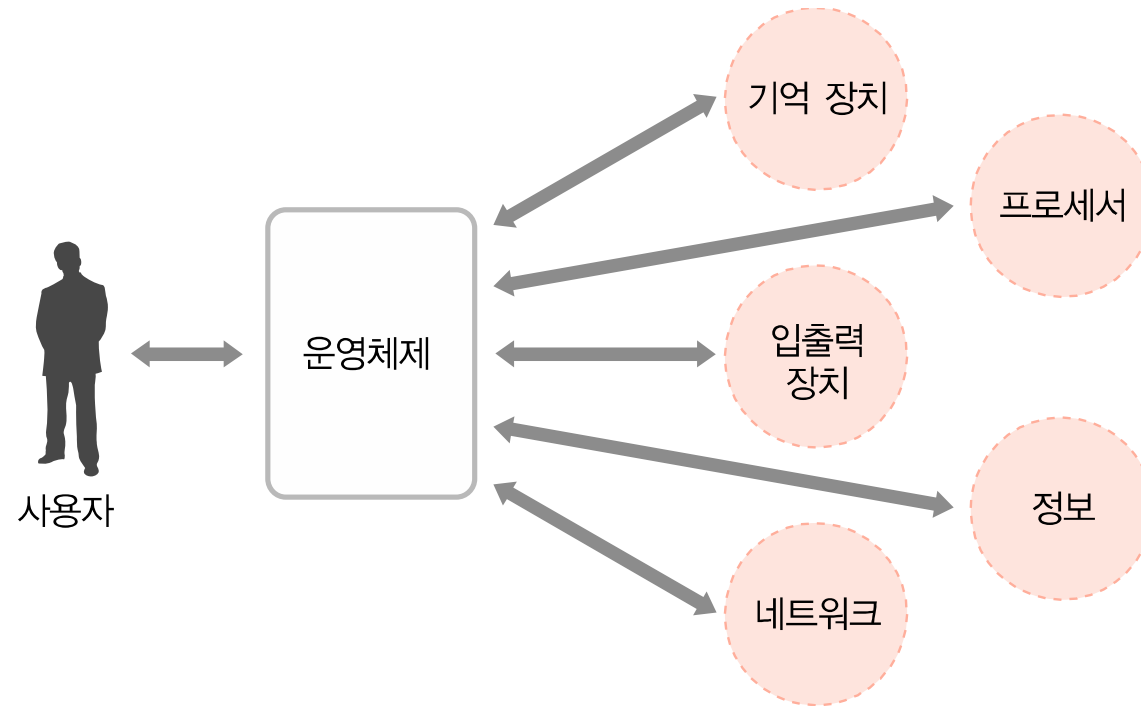
- 컴퓨터 시스템은 하드웨어(Hardware), 소프트웨어(Software), 사용자(User)의 3가지 기본 요소로 구성되어 있으며, 하드웨어와 사용자를 연결시켜 주는 운영체제가 컴퓨터 시스템에서 가장 중요하게 살펴봐야 될 요소



# 운영체제 개요

- 운영체제 = 오퍼레이팅 시스템(Operating System, OS)

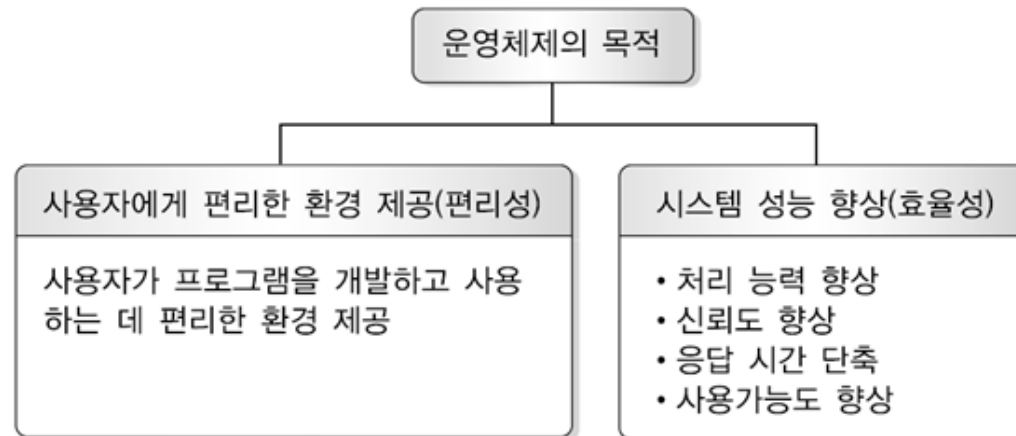
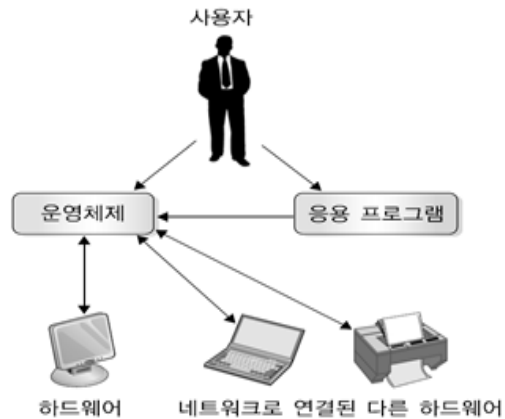
- 사용자가 컴퓨터 시스템을 손쉽게 사용하도록 하고, 시스템 자원(기억 장치, 프로세서, 입출력 장치, 정보, 네트워크 등)을 효율적으로 관리할 수 있도록 하는 소프트웨어의 집합



# 운영체제 개요

- 운영체제 = 오퍼레이팅 시스템(Operating System, OS)

- 운영체제는 사용자에게 편리한 환경을 제공하고 시스템의 성능을 향상시키는 목적을 가지고 시스템의 모든 영역에 관여
- 사용자와 시스템 모두에 영향을 주기 때문에 운영체제가 공격을 받으면 시스템과 사용자 모두 위험, 공격자 입장에서는 가장 공격하고 싶은 대상



# 운영체제의 종류



# 운영체제의 종류



다양한 운영체제



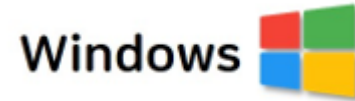
# 운영체제의 종류

The UNIX logo is displayed in a bold, black, sans-serif font. The word "UNIX" is underlined with a series of short horizontal dashes.

대부분의 현대적인  
운영체제의 원형

The Mac OS logo features a blue square icon with a white stylized face and the text "Mac OS" in a blue, serif font.

Linux

The Windows logo consists of the word "Windows" in a black, sans-serif font, followed by the four-pane colored square icon (red, green, blue, yellow).

가장 많이 사용되는  
데스크톱 운영체제



가장 많이 사용되는  
모바일 운영체제

# 운영체제의 종류

유닉스 계열	Windows	Microsoft Windows (Windows 9x, Windows NT)
	데비안	데비안 · 우분투 · 리눅스 민트 · 칼리 리눅스 · 크런치뱅 · Lubuntu 하모니카 · 구름 OS · 기린 · Damn Small Linux · elementary OS · 스팀 OS · Slax · T maxOS · 라즈베리 파이 OS
	레드햇	레드햇 엔터프라이즈 리눅스 · 페도라 · MeeGo · 붉은별 · CentOS · 아시아눅스 타이젠 · Open Mandriva Lx · 마제야 · PCLinuxOS
	슬랙웨어	슬랙웨어 리눅스 · openSUSE
	아치	아치 리눅스 · Manjaro Linux · Antergos · ArchBang Linux · Chakra Linux
	안드로이드	안드로이드 · Remix OS · Polaris OS · Phoenix OS
	젠투	젠투 리눅스 · 편투 리눅스 · 크롬 OS · 웨인 OS
	기타	OpenWRT · Endless OS · NixOS · Solus
	BSD	BSD · FreeBSD · NetBSD · OpenBSD · TrueOS · GhostBSD · DragonFly BSD
	XNU	Darwin · iOS · macOS · iPadOS · tvOS · watchOS
	System V	솔라리스 (SmartOS) · IBM AIX
	DOS	DR-DOS · MS-DOS · FreeDOS · K-DOS
	기타	실시간 운영 체제 · TempleOS · ReactOS · BeOS · 하이쿠 · Mac OS · OS/2(eCS, Arc aOS) · Redox

- 운영체제의 경우 크게 임베디드, 서버, 모바일, 데스크톱의 4개의 시장
- 시장별로 가장 높은 점유율의 운영체제는
  - 임베디드의 경우 리눅스 계열
  - 서버는 유닉스 계열(리눅스 점유율 상승 중)
  - 모바일은 안드로이드
  - 데스크톱은 Windows

# 운영체제의 종류

# UNIX

유닉스는 1969년 미국 벨(Bell) 연구소의 켄 톰슨(Ken Thompson)에 의해 PDP-7 시스템의 운영 체제로 처음 개발



켄 톰슨(Ken Thompson) 과  
PDP-7 시스템

- Unix 계열 운영체제



# 운영체제의 종류



리눅스는 1991년 핀란드 헬싱키 대학의 학생 리누스 토발즈(Linus Torvalds)에 의해 유닉스 호환 운영 체제로 개발



리누스 토발즈(Linus Torvalds) 와  
우분투 리눅스 화면



리눅스는 UNIX가 아니며 별개 계통에 속하기 때문에 Unix-like 운영체제



# 운영체제의 종류

- Linux 계열 운영체제



- 주요 Linux 배포판

- Debian Linux 계열

Ubuntu Linux

- Red Hat Linux 계열

Fedora Linux

CentOS Linux(CentOS Stream)

Red Hat Enterprise Linux(RHEL) : 유료(기업용)

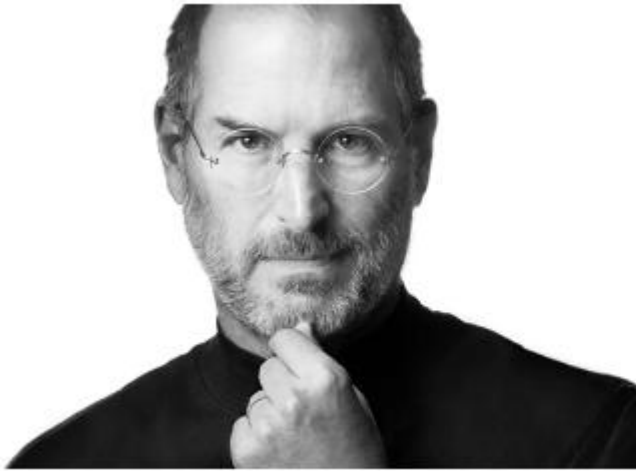
Rocky Linux : RHEL과 1:1로 대응되는 배포판

Oracle Linux : RHEL과 호환

# 운영체제의 종류



Mac OS는 1984년 애플의 개인용 컴퓨터인 매킨토시용으로 개발된 GUI 운영 체제로 초기에 “시스템”으로 불리다가 2006년 Mac OS로 이름 변경 이후 OS X(텐)을 거쳐 2016년 WWDC에서 macOS Sierra를 발표하며 현재의 macOS로 이름을 변경



스티브 잡스(Steve Jobs) 와  
Mac OS 화면



macOS는 현존하는 유일한 PC용 유닉스 운영체제



# 운영체제의 종류



윈도우는 1985년 마이크로소프트가 애플의 그래픽 사용자 인터페이스(GUI) 운영 체제인 맥 OS에 대항하기 위해 개발한 운영 체제



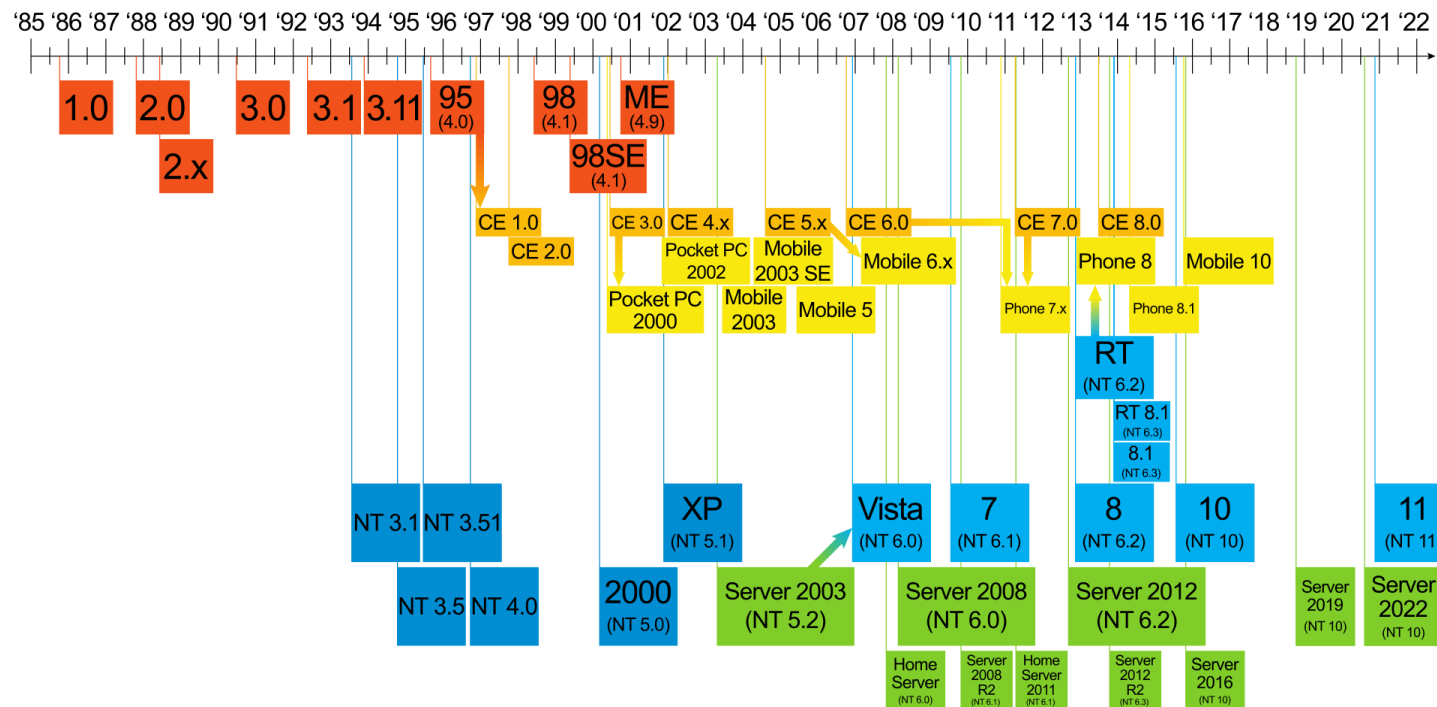
빌 게이츠(Bill Gates) 와  
윈도우11 화면

# 운영체제의 종류

## • Microsoft Windows 운영체제

- NT 계열부터 Unix의 POSIX 표준을 한 때 지원하기도 했고 Windows 10에서 WSL을 지원하고 있으나 Unix 기반의 운영체제로 보기는 어려움
- WSL(Windows Subsystem for Linux)
- POSIX(Portable Operating System Interface , Portable Operating System Interface based on UNIX)

다른 운영체제들 사이의 호환성을 위해 IEEE에서 만든 표준

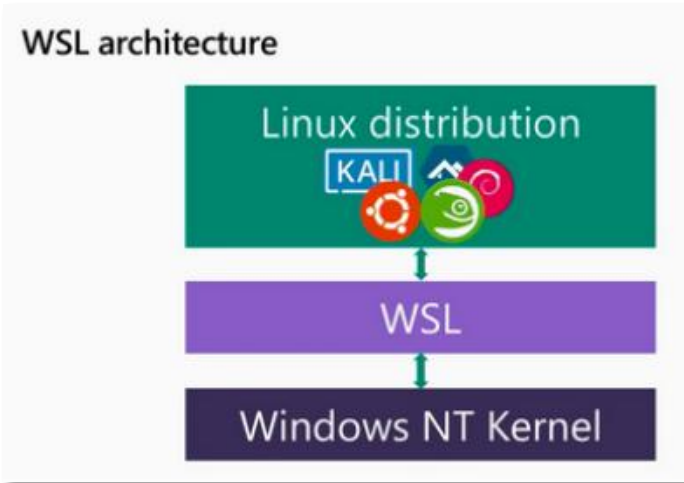


# 운영체제의 종류

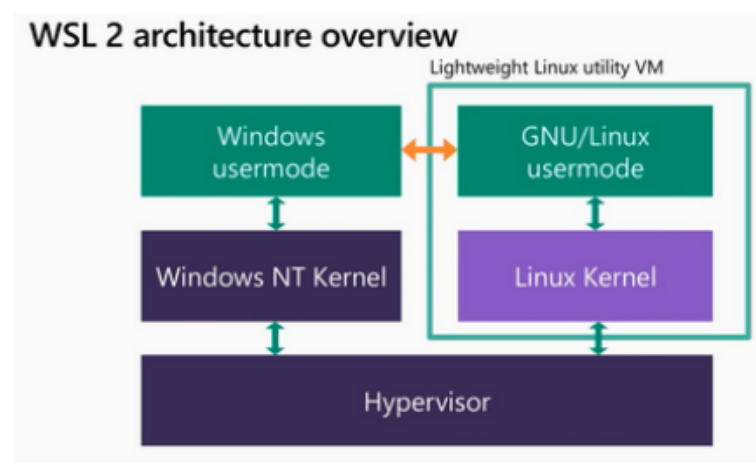
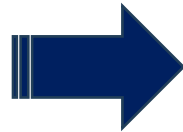
- 리눅스용 윈도우 하위 시스템(Windows Subsystem for Linux, WSL)

- Hyper-V 기반의 경량화 된 가상화 기술을 사용하여 실제 리눅스 커널을 탑재해 Linux 운영체제를 구동할 수 있도록 해주는 기술
- Linux의 호환과 Bash도 사용 가능하고 바이너리를 컴파일 없이 설치하고 실행

WSL은 Windows에서 GNU/Linux 계열의 실행 파일 표준인 ELF(Executable and Linkable Format) binary 파일들을 실행



Linux의 System Call을 Windows API로 변환  
(속도가 느리고 일부 API는 변환 불가능)



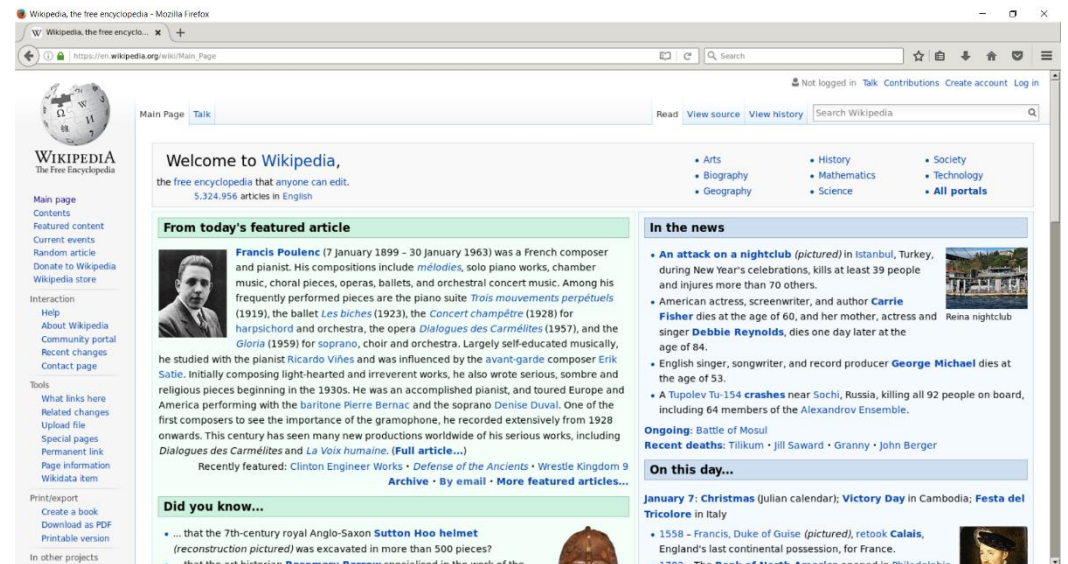
Windows Hyper-V 위에 Linux Kernel을 직접 탑재  
(Linux의 모든 API 사용 가능)

# 운영체제의 종류

- 리눅스용 윈도우 하위 시스템(Windows Subsystem for Linux, WSL)

```
neven@BYTE-PC: /mnt/c/Users/Byte
neven@BYTE-PC:~$ cd /mnt/c/Users/Byte
neven@BYTE-PC:/mnt/c/Users/Byte$ uname -a
Linux BYTE-PC 4.4.0-19041-Microsoft #1-Microsoft Fri Dec 06 14:06:00 PST 2019 x86_64 x86_64 x86_64 GNU/Linux
neven@BYTE-PC:/mnt/c/Users/Byte$ cat /etc/os-release
NAME="Ubuntu"
VERSION="18.04.2 LTS (Bionic Beaver)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 18.04.2 LTS"
VERSION_ID="18.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=bionic
UBUNTU_CODENAME=bionic
neven@BYTE-PC:/mnt/c/Users/Byte$
```

WSL에서 실행 중인 배시 셸

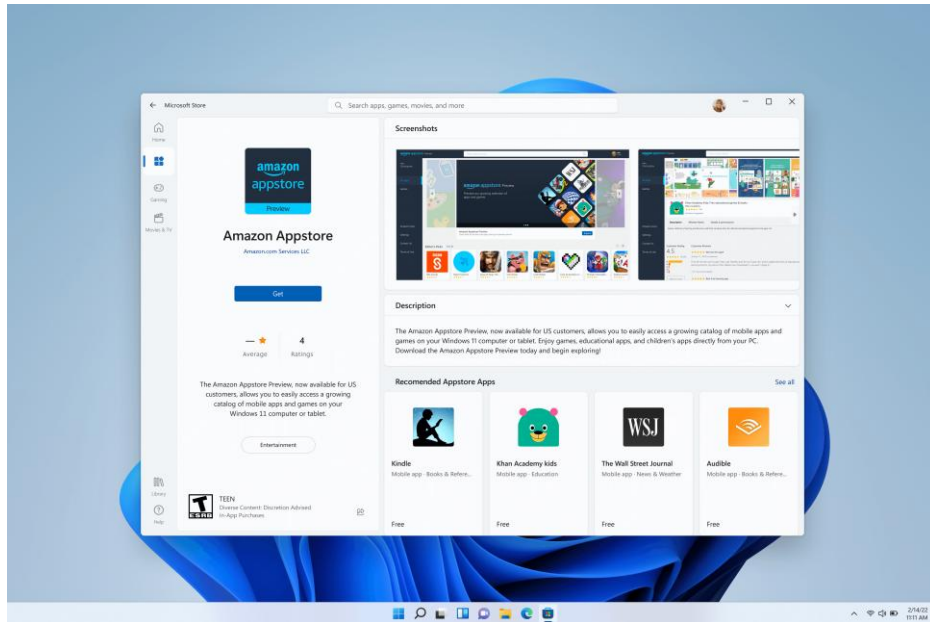


WSL에서 구동 중인 리눅스용 모질라 파이어폭스

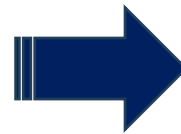
# 운영체제의 종류

- 안드로이드용 윈도우 하위 시스템(Windows Subsystem for Android, WSA)

- WSL(Windows Subsystem for Linux)과 마찬가지로 WSA는 운영체제 네이티브처럼 Windows 11 PC에서 Android 앱 실행 가능  
WSL과 마찬가지로 Hyper-V(Hypervisor) 위에 Android OS가 구동되는 방식
- Android용 Windows 하위 시스템은 Windows 11에서 사용 가능하지만,  
Windows 기반 Amazon Appstore(Windows 11에서 Android 앱을 실행하기 위한 요구 사항) 만 사용 가능



Windows 기반 Amazon Appstore



2025년 3월 지원 종료

# 운영체제의 유형

# 운영체제의 유형

- 운영체제의 유형

- 운영체제는 데이터를 처리하는 방식에 따라서 6가지의 유형으로 구분할 수 있음

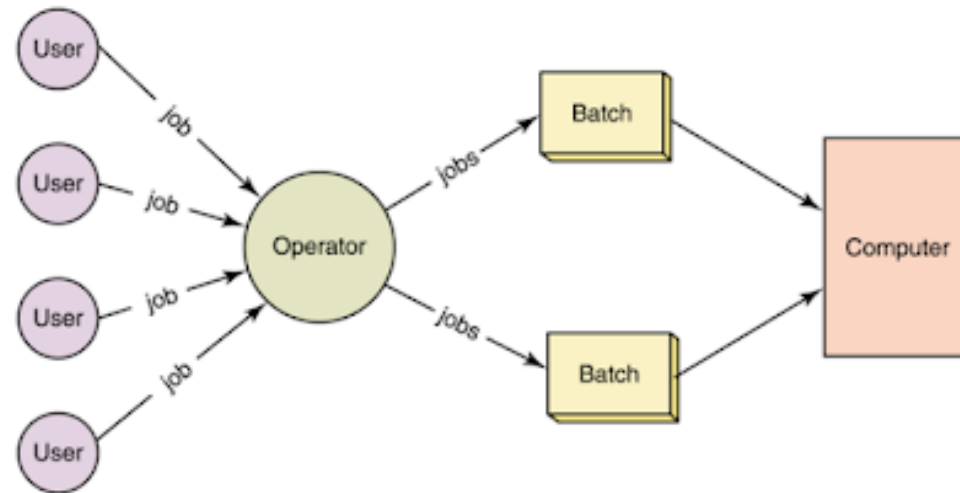
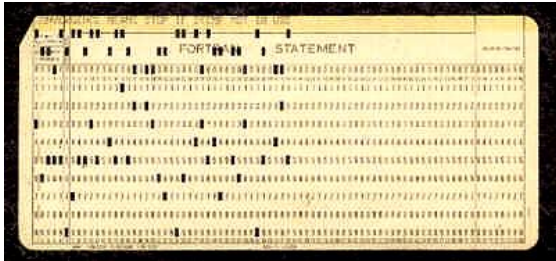




# 운영체제의 유형

- 일괄 처리 시스템(Batch processing system)

- OMR 카드와 같은 펀치 카드에 작업을 저장하고 처리 시스템인 오퍼레이터에 전달하면 오퍼레이터는 비슷한 작업을 한데 모아 일괄 처리하는 방식
- 초기의 컴퓨터 시스템에서 사용된 형태로, 데이터를 바로 처리하는 것이 아니라 일정량 또는 일정 기간 동안 데이터를 모아서 한번에 처리하는 방식
- 프로세스가 다른 작업 수행 시 입출력 작업이 불가능하여 프로세서와 메인 메모리의 활용도가 떨어짐





# 운영체제의 유형

- 다중 프로그래밍 시스템(Multiprogramming system)

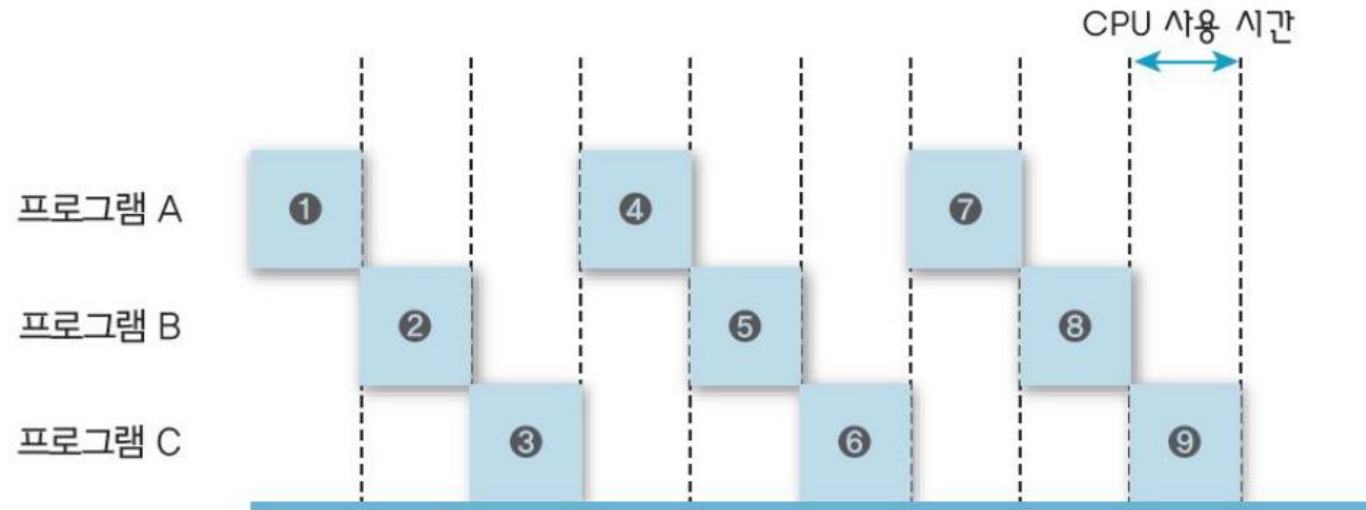
- 일괄 처리 시스템의 문제점을 다중 프로그래밍을 도입하여 해결 (프로세서 사용 최대화)
- 프로그램 A를 실행(1), 프로세서를 사용하지 않는 유휴 시간이 생기면 프로그램 B를 실행(2), 다시 유휴 시간이 생기면 프로그램 A를 실행 (3)
- 효율적으로 프로세서를 사용 (거의 동시에 프로세서를 할당 받는 듯한 느낌)
- 여러 작업을 준비 상태로 두려면 이를 메모리에 보관, 일정 형태의 메모리를 관리해야 함



# 운영체제의 유형

- 시 분할 시스템(Time sharing system)

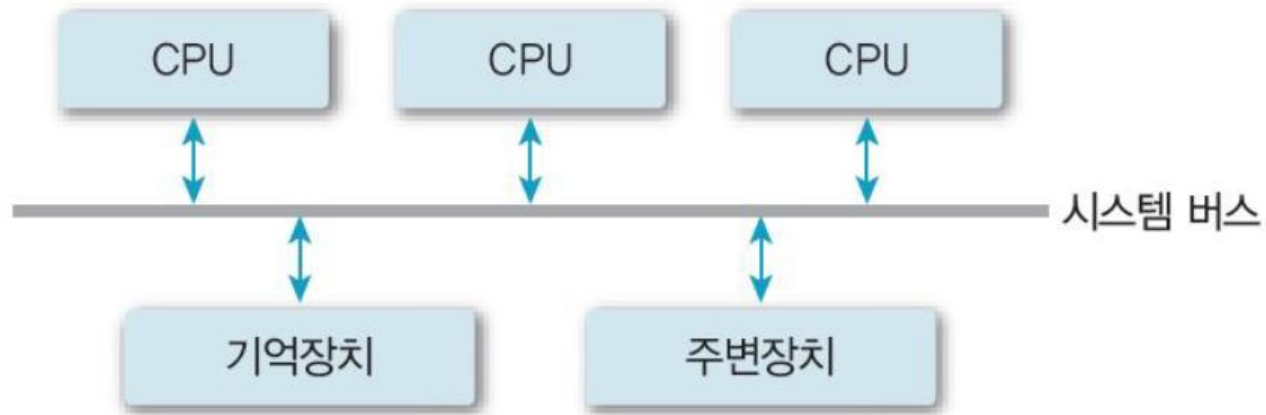
- 다중 프로그래밍의 확장된 개념, CPU 스케줄링과 다중 프로그래밍을 이용해서 각 사용자들에게 컴퓨터 자원을 시간적으로 분할하여 배분
- 시스템은 한 사용자에서 다음 사용자로 빠르게 전환함으로써 각 사용자에게 자신만이 컴퓨터를 사용하고 있는 것과 같은 착각을 주지만, 실제로는 여러 사용자가 하나의 컴퓨터를 공유하여 사용
- 어떤 프로그램을 먼저 실행할지 결정하는 스케줄링 개념 필요 (오늘날 대부분의 시스템은 시분할 시스템)



# 운영체제의 유형

- 다중 처리 시스템(Multiprocessing system)

- 단일 컴퓨터 시스템 내에서 둘 이상의 프로세서 사용, 동시에 둘 이상의 프로세스 지원
- 여러 프로세서와 시스템 버스, 클럭, 메모리와 주변장치 등을 공유
- 빠르고, 프로세서 하나가 고장 나도 다른 프로세서를 사용하여 작업을 계속할 수 있음 >> 신뢰성 향상
- 프로세서 간의 연결, 상호작용, 역할 분담 등을 고려해야 함
- 다중 프로그래밍은 CPU 1개, 프로그램 여러 개, 다중 처리 시스템은 CPU 여러 개



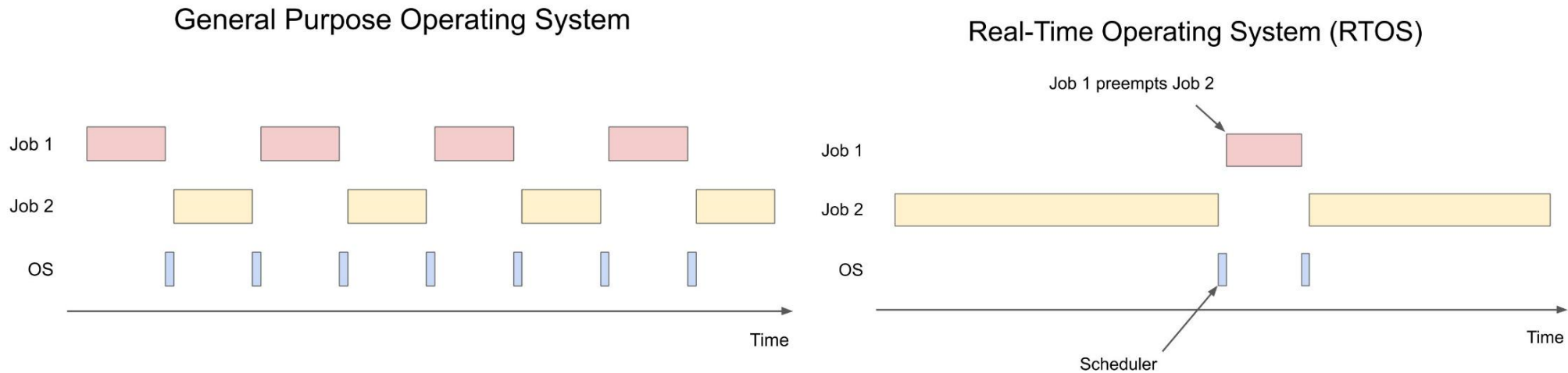
# 운영체제의 유형

- 실시간 처리 시스템(Real time processing system)

- 처리를 요구하는 자료가 발생할 때마다 즉시 처리하여 그 결과를 출력하거나 요구에 대하여 응답하는 방식
- 정해진 시간 안에 어떠한 일이 반드시 종료됨이 보장되어야 하는 시스템을 위한 운영체제
- 작업의 종류

Hard real-time task : 각 task들을 무조건 deadline에 맞춰서 끝내주는 시스템 (발전소/공장 제어, 미사일 제어, 반도체 제어, 은행, 공장, 기차나 비행기 좌석 예약 등)

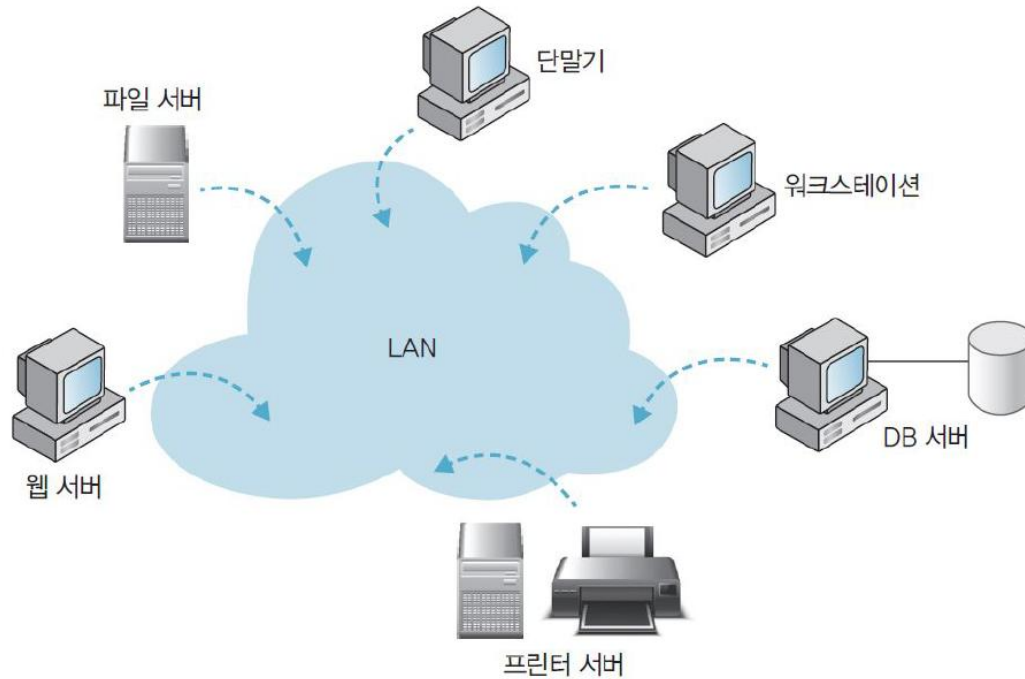
Soft real-time task : 시스템이 real-time task에 우선 순위를 더 두지만, 무조건 deadline에 맞춰서 끝낼 것을 보장하지는 않음 (동영상 재생 등)



# 운영체제의 유형

- 분산 처리 시스템(Distributed processing system)

- 주로 네트워크를 기반으로 구축된 병렬 처리 시스템
- 시스템마다 독립적인 운영체제와 메모리로 운영, 필요시 통신하는 시스템
- 데이터를 여러 위치에서 처리, 저장, 여러 사용자가 공유
- 하나의 프로그램을 여러 시스템에서 동시에 실행

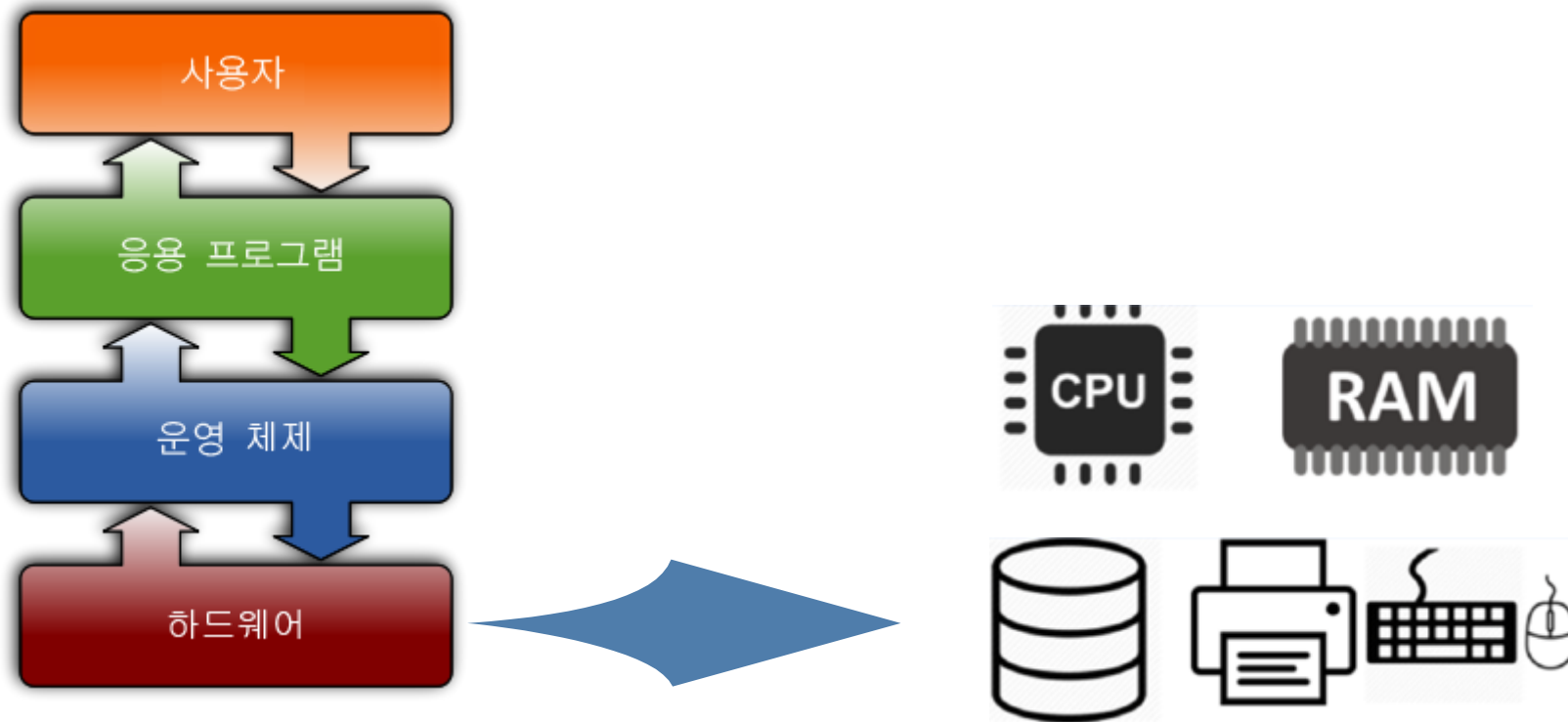


# 운영체제의 기능

# 운영체제의 기능

- 컴퓨터 시스템 : 하드웨어(Hardware, H/W)

- 하드웨어는 중앙 처리 장치(CPU), 메모리(RAM), 디스크(Disk), 키보드, 마우스, 그래픽 카드, 사운드 카드, 메인보드, 프린터, 모니터와 같은 컴퓨터의 물리적 부품을 의미



시스템의 하드웨어

# 운영체제의 기능

## • 컴퓨터 시스템 : 응용 프로그램

- 응용 소프트웨어 또는 애플리케이션으로도 불리며, 운영 체제에서 실행되는 모든 프로그램을 의미  
워드프로세서나 스프레드시트, 웹브라우저 뿐만 아니라 컴파일러나 링커 등도 응용 프로그램
- 좁은 의미에서는 운영체제 위에서 사용자가 직접 사용하게 되는 프로그램을 말함

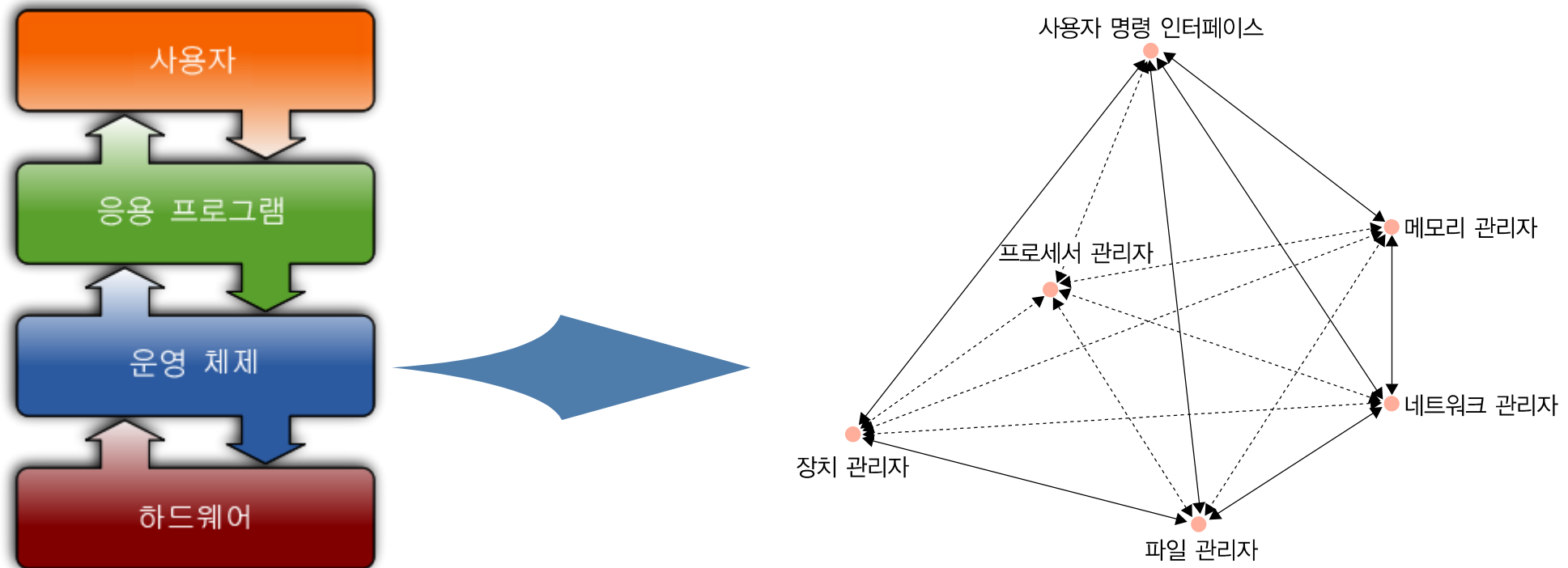




# 운영체제의 기능

- 컴퓨터 시스템 : 운영체제 = Operation System(OS)

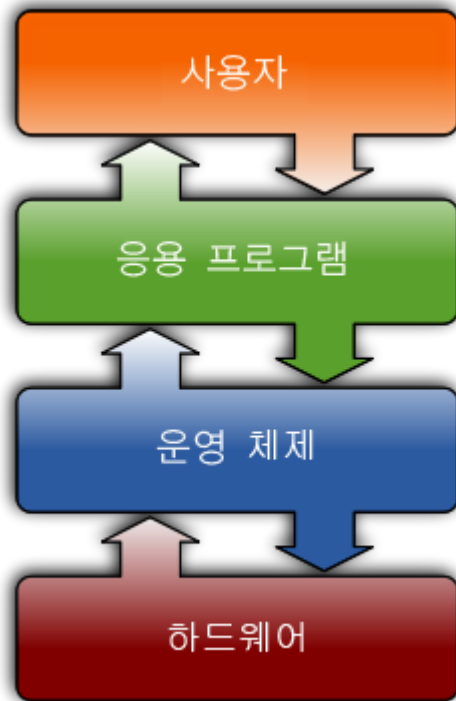
- 사용자의 하드웨어, 시스템 리소스를 제어하고 프로그램에 대한 일반적 서비스를 지원하는 시스템 프로그램
- 일반 PC는 단일 사용자 운영체제 구성 모델, 즉 사용자 명령 인터페이스를 기반으로 하여 메모리 관리자, 프로세서 관리자, 장치 관리자, 파일 관리자 등 4가지 서브 시스템 관리자로 구성 (네트워크 지원 운영체제에서는 네트워크 관리자 추가)



# 운영체제의 기능

- 컴퓨터 시스템 : 운영체제 - 서브시스템 관리자

- 운영체제를 구성하는 요소는 다양하지만 컴퓨터 시스템의 자원의 성격에 따라 크게 네 가지의 서브시스템으로 구성  
(네트워크 지원 운영체제에는 네트워크 관리자 추가)



- 메모리 관리자**

프로그램의 메모리 요청의 적합성 점검, 적합한 경우 할당된 메모리를 다른 프로그램이 접근하지 못하게 관리, 보호하고 프로그램 종료 시 할당된 메모리를 회수

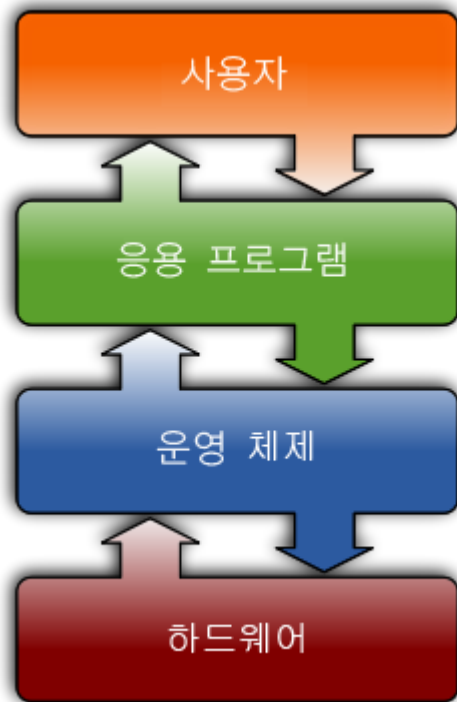
- 프로세서 관리자**

명령어들이 체계적이고 효율적으로 실행되도록 작업 스케줄링하여 사용자의 작업 요청에 대해 수용 또는 거부

# 운영체제의 기능

- 컴퓨터 시스템 : 운영체제 - 서브시스템 관리자

- 운영체제를 구성하는 요소는 다양하지만 컴퓨터 시스템의 자원의 성격에 따라 크게 네 가지의 서브시스템으로 구성  
(네트워크 지원 운영체제에는 네트워크 관리자 추가)

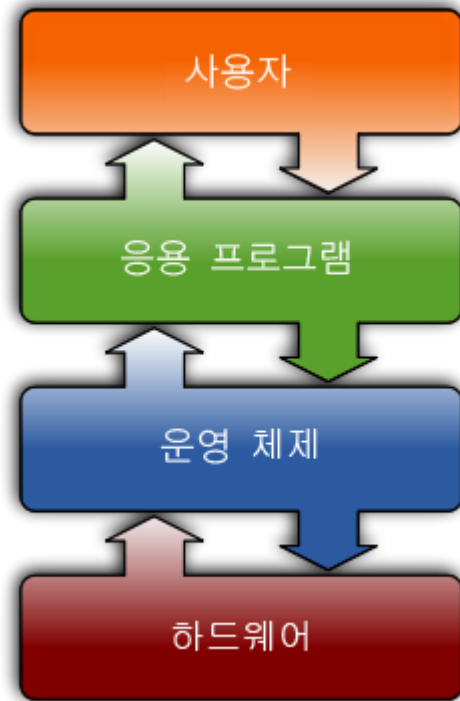


- **장치 관리자**  
프린터, 디스크 드라이버, 모뎀, 모니터 등 시스템 내의 장치(Device)를 프로그램에 할당 또는 회수
- **파일 관리자**  
사용자 별로 파일 접근(Access) 권한을 부여하고, 권한에 따라 파일을 할당(Open)하고 해제(Close)
- **네트워크 관리자**  
네트워크에서 접근 가능한 CPU, 메모리, 프린터, 디스크 드라이버, 모뎀, 모니터 같은 자원을 관리

# 운영체제의 기능

- 컴퓨터 시스템 : 응용 프로그램(셸, Shell) = 사용자 명령 인터페이스

- 운영 체제 상에서 다양한 운영 체제 기능과 서비스를 구현하는 인터페이스를 제공하는 프로그램



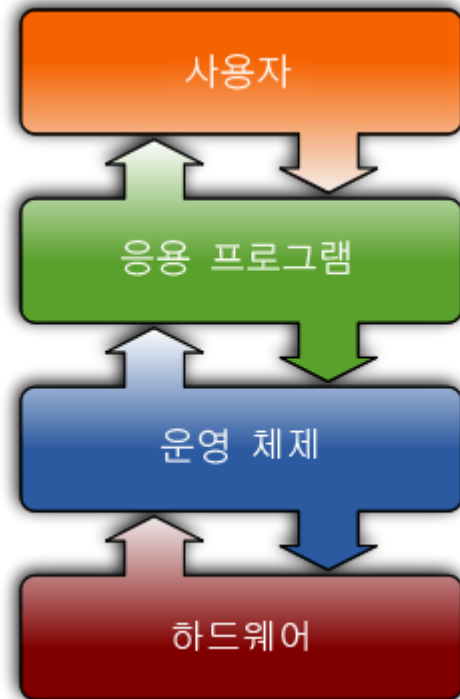
셸



- 셸이라는 용어는 사용자와 운영 체제의 내부(커널) 사이의 인터페이스를 감싸는 층이기 때문에 그러한 이름이 붙음
- 셸은 응용 프로그램의 하나이지만 운영체제의 일부로 제공
- 셸은 일반적으로 명령 줄과 그래픽 형의 두 종류로 분류
  - 명령 줄 셸은 명령 줄 인터페이스(CLI)를 제공
  - 그래픽 셸은 그래픽 사용자 인터페이스(GUI)를 제공

# 운영체제의 기능

- 컴퓨터 시스템 : 응용 프로그램(셸, Shell) = 사용자 명령 인터페이스
  - 명령 줄 인터페이스(CLI, Command-line interface)



셸 (CLI)



```
C:\Temp> dir
Volume in drive C is C
Volume Serial Number is 74F5-B93C

Directory of C:\Temp

2009-08-25  11:59    <DIR>          .
2009-08-25  11:59    <DIR>          ..
2007-03-01  11:37             2,321,600 AdobeUpdater12345.exe
2009-04-03  10:01             27,988 dd_depcheckdotnetfx30.txt
2009-04-03  10:01              764 dd_dotnetfx3error.txt
2009-04-03  10:01             32,572 dd_dotnetfx3install.txt
2009-06-09  13:46             35,145 GenProfile.log
2009-08-05  12:11              155 KB969856.log
2009-04-20  08:37              402 MSI29e0b.LOG
2009-04-09  16:34             38,895 offcln11.log
2009-04-03  16:02    <DIR>        OfficePatches
2009-07-14  14:30    <DIR>        OHotfix
```

cmd (윈도우 CLI 셸)

```
chealer@vinci:/usr/share/doc/bash$ export LC_ALL=C
chealer@vinci:/usr/share/doc/bash$ cd ~chealer/
chealer@vinci:~$ ls
Cloutier  Ido      Musique  logs     skolo    sources
Desktop  Mes images boston   ncix.png smb4k    vieux
```

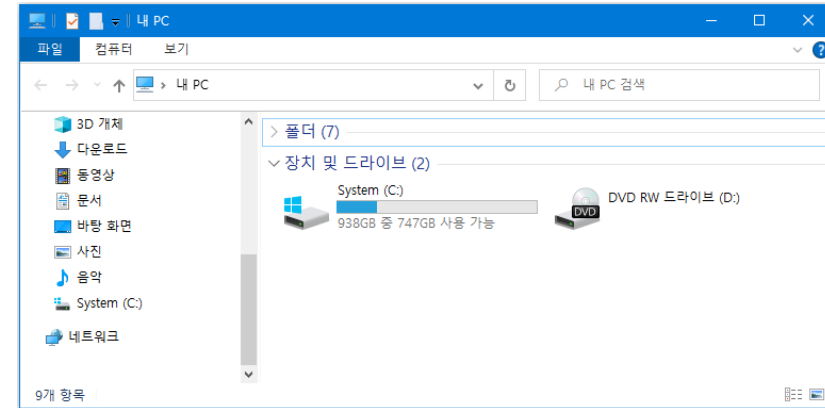
bash (리눅스 CLI 셸)

# 운영체제의 기능

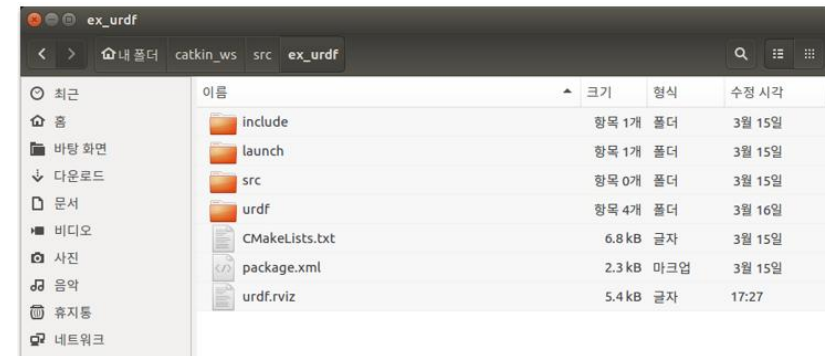
- 컴퓨터 시스템 : 응용 프로그램(셸, Shell) = 사용자 명령 인터페이스
  - 그래픽 사용자 인터페이스(GUI, Graphic user interface)



셸 (GUI)



윈도우 탐색기(윈도우 GUI 셸)

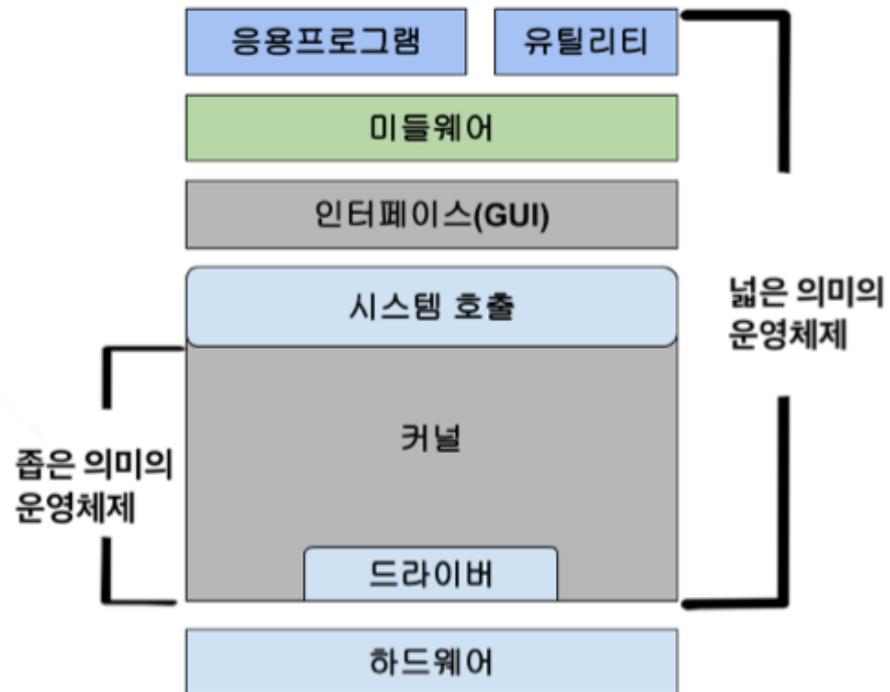


우분투 탐색기 (리눅스 GUI 셸)

# 운영체제의 구성

# 운영체제의 구성

- 운영체제 = 커널(kernel) + 시스템 프로그램(시스템 서비스, 시스템 유틸리티 등)



- 좁은 의미에서의 운영체제는 커널을 의미하지만
- 넓은 의미에서는 운영체제(윈도우, 우분투 등)와 함께 제공되는 시스템 프로그램(시스템 서비스, 시스템 유틸리티 등)도 포함



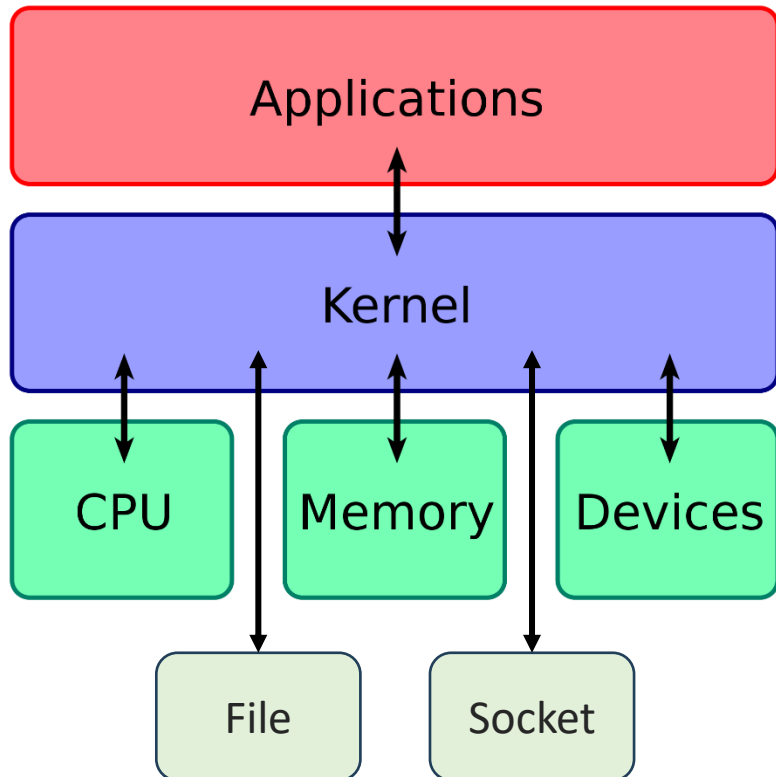
# 운영체제의 구성

- 운영체제의 시스템 프로그램(시스템 서비스, 시스템 유틸리티 등)의 예
  - 파일 관리 : 파일과 디렉터리를 생성, 삭제, 복사, 이름변경(rename), 프린트, 목록(list) 보기와 같은 기능 제공
  - 상태 정보 : 날짜, 시간, 사용 가능한 메모리, 디스크 공간의 양, 사용자 수 등의 상태 정보를 제공
  - 파일 변경 : 디스크나 다른 저장 장치에 저장된 파일의 내용을 생성하고 변경하기 위해 다수의 텍스트 편집기 제공
  - 프로그래밍언어 지원 : 프로그래밍 언어들( ex: C, C++, Java, python등 )에 대한 컴파일러, 어셈블러, 디버거 및 해석기 제공
  - 프로그램 적재와 실행 : 프로그램의 실행을 위해서 프로그램을 메모리에 적재하는 로더(loader) 제공
  - 통신 : 프로세스, 사용자, 그리고 다른 컴퓨터 시스템들 사이에 통신을 위한 기능을 제공
  - 백그라운드 서비스 : 항상 백그라운드에서 계속 실행되는 시스템 프로그램(서비스, 서브시스템, 또는 데몬으로 불림) 제공

# 운영체제의 구성

- 운영체제와 커널

- 운영체제의 핵심(core)인 커널(kernel)
- 커널의 가장 큰 목표는 컴퓨터의 물리적(하드웨어) 자원과 가상화 된 자원을 관리 하는 것



- 커널은 항상 메모리에 올라가 있는 운영체제의 핵심 부분으로서 하드웨어와 응용 프로그램 사이에서 인터페이스를 제공하는 역할을 하며 컴퓨터 자원들을 관리하는 역할
- 즉, 커널은 인터페이스로서 응용 프로그램 수행에 필요한 다양한 서비스를 제공하고, 여러가지 하드웨어(CPU, 메모리) 등의 리소스를 관리하는 역할

# 운영체제의 구성

- 운영체제와 커널 - 가상화(Virtualization)

- 시스템의 물리적 자원을 가상화 하는 이유는 물리적으로 하나뿐인 하드웨어를 여러 사용자들이 번갈아 사용할 수 있도록 하기 위한 것으로, 각 사용자들을 커널이 관리하는 가상화 된 자원을 사용함으로써 실제 물리적 자원인 특정 하드웨어를 독점하는 것처럼 느낄 수 있도록 함

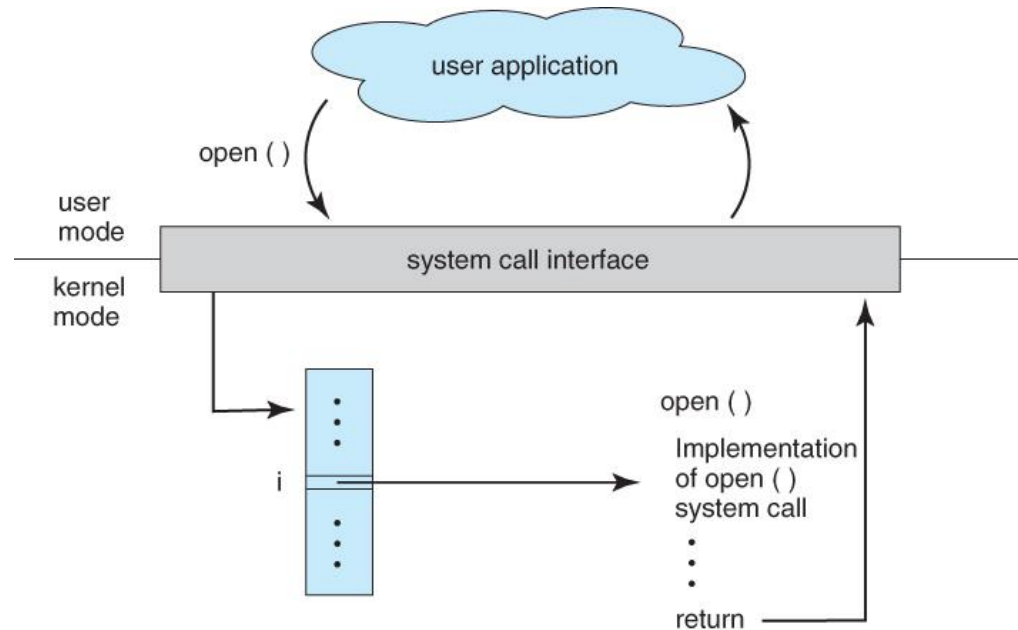
- 추상화(Abstraction)와 가상화(Virtualization)

- 추상화란 복잡한 것을 단순화 시켜서 표현하는 것을 말함  
사용자가 컴퓨터를 사용한다 라고 할 때, 내부적으로 하드웨어들이 어떻게 동작하는지 알 필요 없음
- 가상화는 물리적인 시스템 자원을 추상화 시켜서 물리적인 자원을 논리적인 단위로 나누고 통합하여 자원을 활용할 수 있게 해주는 기술  
디스크와 같은 물리적인 저장 장치를 "File"로 표현 한다거나 물리적인 네트워크 장치를 "Socket"으로 표현)

# 운영체제의 구성

- 운영체제와 커널 - 운영체제 동작 방식

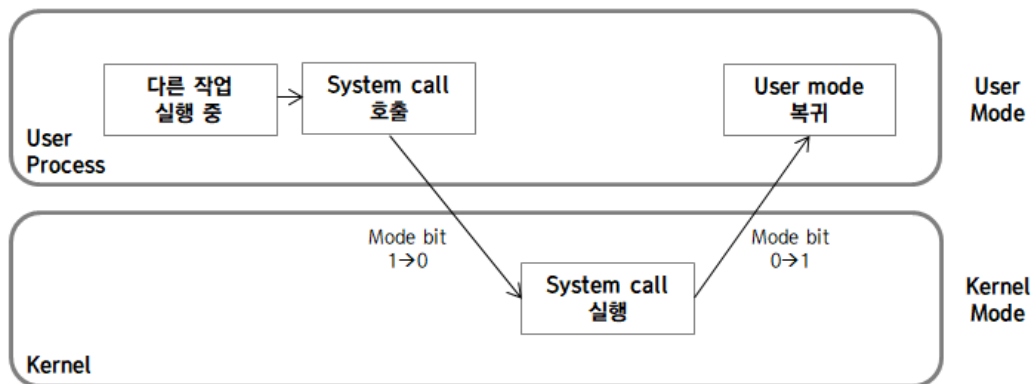
- 운영체제는 보안상의 이유로 사용자 모드(user mode)와 커널 모드(kernel mode, supervisor mode, privileged mode)로 구분되어 동작
- 유저 모드의 프로그램이 커널 모드에서 동작해야 하는 기능을 호출하게 되면  
운영체제는 커널 모드로 전환하고 커널에서 요청을 처리한 다음 유저 모드로 제어권을 돌려 주게 됨



# 운영체제의 구성

## • 운영체제와 커널 - 시스템 호출(System Call)

- 사용자 프로그램이 운영체제가 제공하는 여러 서비스들을 사용하기 위해서는 커널(kernel)에 요청을 해야 하는데, 이때 커널에 보내는 요청을 시스템 호출이라고 함



### EXAMPLES OF WINDOWS AND UNIX SYSTEM CALLS

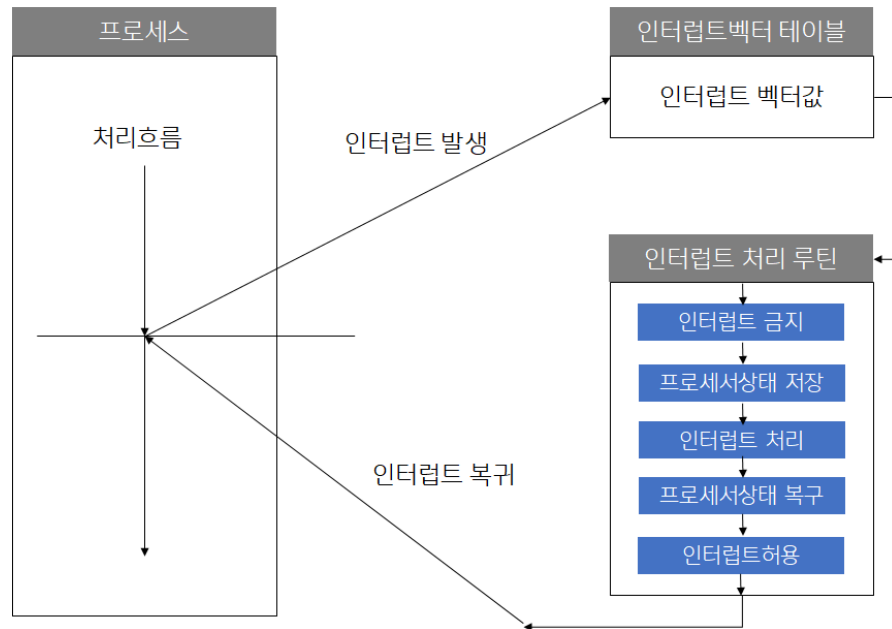
The following illustrates various equivalent system calls for Windows and UNIX operating systems.

	Windows	Unix
Process control	CreateProcess() ExitProcess() WaitForSingleObject()	fork() exit() wait()
File management	CreateFile() ReadFile() WriteFile() CloseHandle()	open() read() write() close()
Device management	SetConsoleMode() ReadConsole() WriteConsole()	ioctl() read() write()
Information maintenance	GetCurrentProcessID() SetTimer() Sleep()	getpid() alarm() sleep()
Communications	CreatePipe() CreateFileMapping() MapViewOfFile()	pipe() shm_open() mmap()
Protection	SetFileSecurity() InitializeSecurityDescriptor() SetSecurityDescriptorGroup()	chmod() umask() chown()

# 윈도우 시스템

## • 운영체제와 커널 - 인터럽트(Interrupt)

- CPU는 여러 가지 일을 처리해야만 하고 동시에 처리할 수 없는 일들을 순차적으로 하나씩 처리해야 함  
그러나 지금 하고 있는 일보다 긴급한 일(또는 중요한 일)이 발생하면 그 일을 먼저 처리해주고 나서 하던 일을 계속하게 되는데  
이러한 긴급한 일을 인터럽트라고 하며 하드웨어나 소프트웨어 에러, 시스템 콜 등에 의해서 발생
- 인터럽트 기능이 없다면 CPU는 어떤 특정한 일을 할 시기를 알기 위해 계속 체크하는 폴링(polling)에 자원을 낭비하게 됨

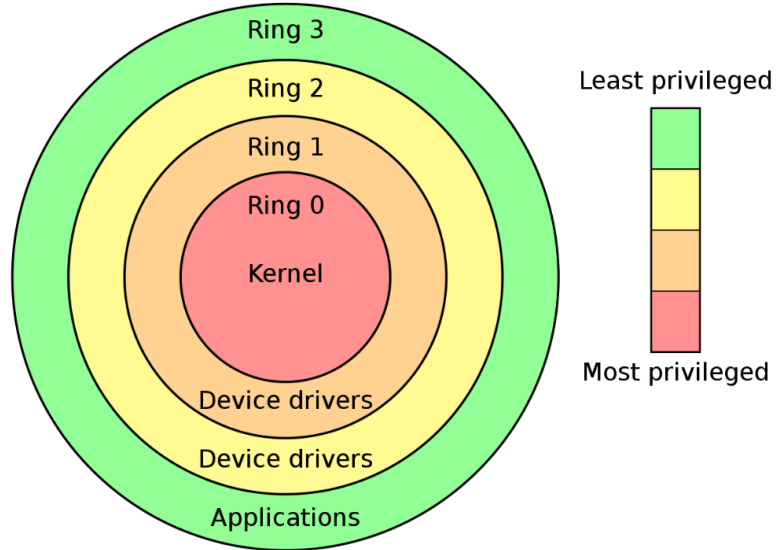


- 인터럽트가 발생하면 현재 수행 중인 프로그램을 멈추고 상태 레지스터와 PC(Program Counter)등을 스택에 잠시 저장한 후 인터럽트 서비스 루틴으로 점프해서 처리 후 원래의 작업으로 복귀
- 작업을 저장하는 이유는 인터럽트의 서비스 루틴이 끝나고 나서 다시 원래의 해당 작업으로 돌아와야 하기 때문

# 윈도우 시스템

- 운영체제와 커널 - CPU의 링(Ring)

- CPU 차원에서도 보안을 위해서 링(Ring) 개념을 사용하고 있음
- CPU는 악성 소프트웨어나 버그로 인한 충돌로 부터 커널을 보호하기 위해서 가장 높은 권한을 가지는 커널이 Ring 0로 실행되며 일반적인 응용 프로그램은 가장 낮은 권한을 가지는 Ring 3에서 실행됨
- Ring 1 과 Ring 2는 장치 드라이버를 위해 있지만 윈도우나 리눅스와 같은 운영체제에서는 사용되지 않음

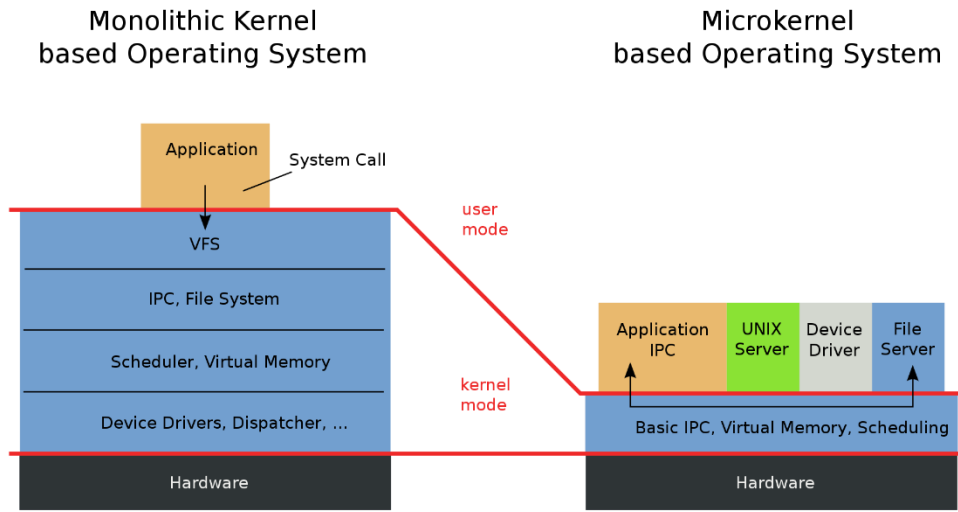


- Ring 0 : 커널 모드(kernel mode)  
운영체제가 커널을 사용할 때 쓰는 모드
- Ring 3 : 유저 모드(user mode)  
응용 프로그램이 사용하는 모드

# 운영체제의 구성

- 운영체제와 커널 - 커널의 종류

- 커널은 크게 두 가지로 구분할 때는 단일형 커널(monolithic kernel), 마이크로 커널(microkernel)로 구분하고 두 가지 특징을 모두 가지게 되면 혼합형 커널(hybrid kernel)로 구분



- 단일형 커널과 마이크로 커널 기반 운영 체제의 구조

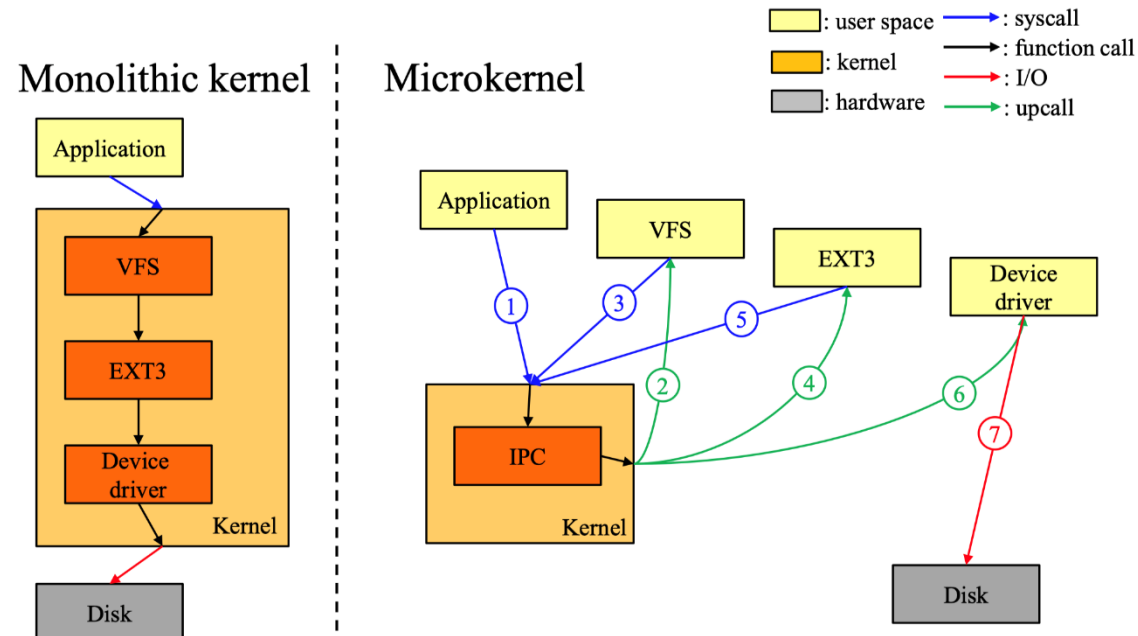
- 단일형 커널은 사용자 프로그램을 제외한 모든 시스템 서비스들을 커널이 직접 처리하는 방식 (Unix, Linux 계열)
- 마이크로 커널은 커널에서는 핵심 서비스만 담당하고 다른 서비스들은 유저 모드에서 처리하는 방식 (Windows NT, Mac OS 계열)



# 운영체제의 구성

- 운영체제와 커널 - 커널의 종류

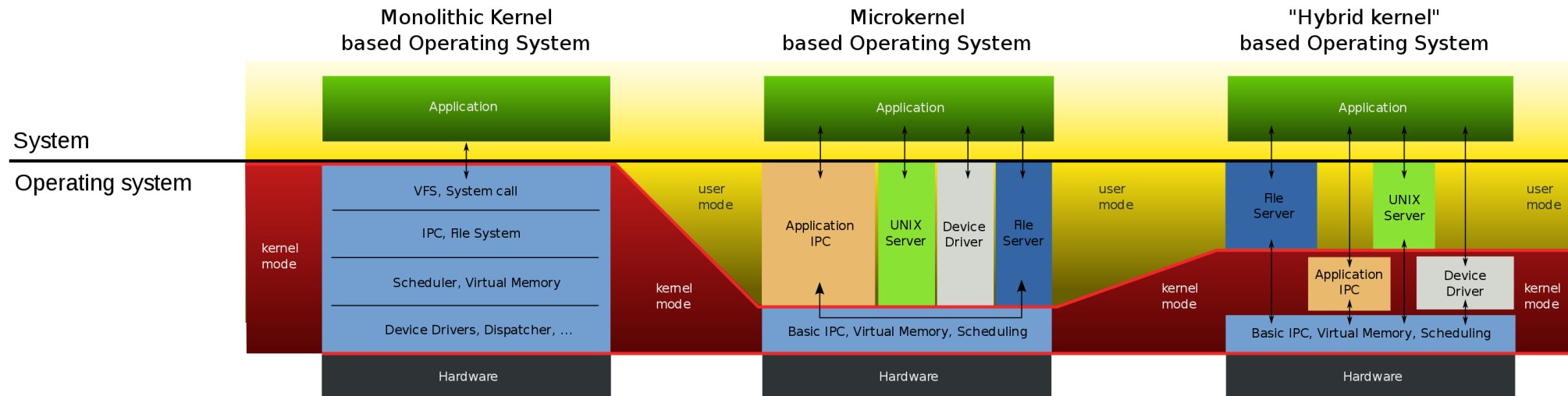
- Block I/O에 대해 단일형 커널(monolithic kernel), 마이크로 커널(microkernel)의 차이



# 운영체제의 구성

## • 운영체제와 커널 - 커널의 종류

- 혼합형 커널(hybrid kernel)은 단일형 커널과 마이크로 커널의 이점을 모두 취하기 위한 방식으로  
현재의 Linux, Mac OS, Windows는 혼합형 커널(hybrid kernel) 구분하기도 함
- 단일형 커널과 마이크로 커널로만 구분하는 경우에는 Linux는 단일형 커널, Mac OS와 Windows는 마이크로 커널로 분류)



# QA

