

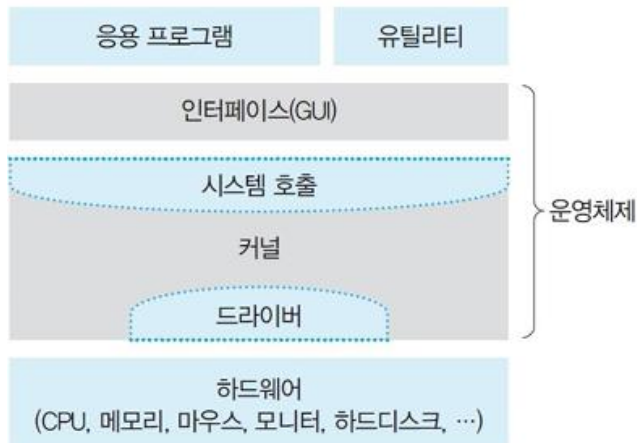
## 커널

: 프로세스 관리, 메모리 관리, 저장장치 관리와 같은 운영체제 핵심적인 기능 모아둠

## 인터페이스

: 커널에 사용자 명령을 전달하고 실행 결과를 사용자에게 알려주는 역할.

그래픽 사용 인터페이스 GUI(Graphical User Interface)

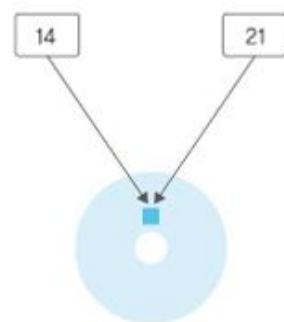


## 시스템 호출

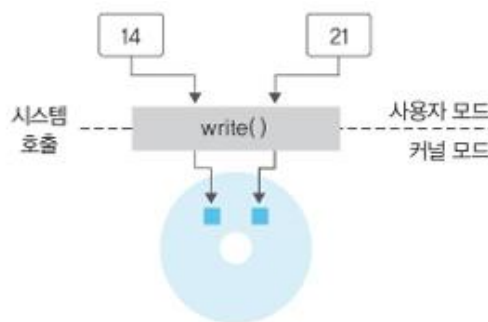
: 커널이 자신 보호하기 위해 만든 인터페이스. 커널은 사용자/응.프로부터 컴퓨터 자원을 보호하기 위해 자원에 직접 접근하는 것을 차단

직접 접근: 두 응.프가 자기 맘에 드는 위치에 데이터 저장하려 함 (다른 사람 데이터 지울 수 있고 내 데이터가 다른 사람에 의해 지워질 수도 있음)

시스템 호출을 통한 접근: 응.프가 직접 하드디스크에 데이터 저장하지 않고 커널이 제공하는 `write()` 함수 사용해서 데이터 저장해달라 요청 (커널이 데이터 가져오거나 저장하는 거 책임져서 컴퓨터 자원 관리 수월)



(a) 직접 접근



(b) 시스템 호출을 통한 접근

## 시스템 호출 정리

시스템 호출은 커널이 제공하는 시스템 자원의 사용과 연관된 함수

응.프가 hw자원에 접근하거나 운체가 제공하는 서비스 이용하려 할 때 시스템호출 사용

운체는 커널이 제공하는 서비스를 시스템 호출로 제한, 다른 방법으로 커널에 들어오지 못하게 막아서 컴퓨터 자원 보호

시스템 호출은 커널이 제공하는 서비스를 이용하기 위한 인터페이스! 사용자가 자발적으로 커널 영역에 진입할 수 있는 유일한 수단

시스템 호출과 유사한 용어로 응용 프로그램 인터페이스(Application Programming Interface; API), 시스템 개발자용 키트(System Developer's Kit; SDK)

디바이스 드라이버(device driver)

: 커널과 하드웨어의 인터페이스 담당. 마우스와 같이 간단한 제품은 드라이버를 커널이 갖고있는데 그래픽카드와 같은 복잡한 hw는 제작자가 드라이버 제공

표 1-2 커널이 하는 일

핵심 기능	설명
프로세스 관리	프로세스에 CPU를 배분하고 작업에 필요한 제반 환경을 제공한다.
메모리 관리	프로세스에 작업 공간을 배치하고 실제 메모리보다 큰 가상공간을 제공한다.
파일 시스템 관리	데이터를 저장하고 접근할 수 있는 인터페이스를 제공한다.
입출력 관리	필요한 입력과 출력 서비스를 제공한다.
프로세스 간 통신 관리	공동 작업을 위한 각 프로세스 간 통신 환경을 지원한다.

단일형 구조(monolithic architecture) 커널

: 초창기 운체 구조. 커널 핵심 기능 구현하는 모듈들이 구분 없이 하나로 구성



장점: 모듈 간 통신 비용 줄어 효율적인 운영 가능

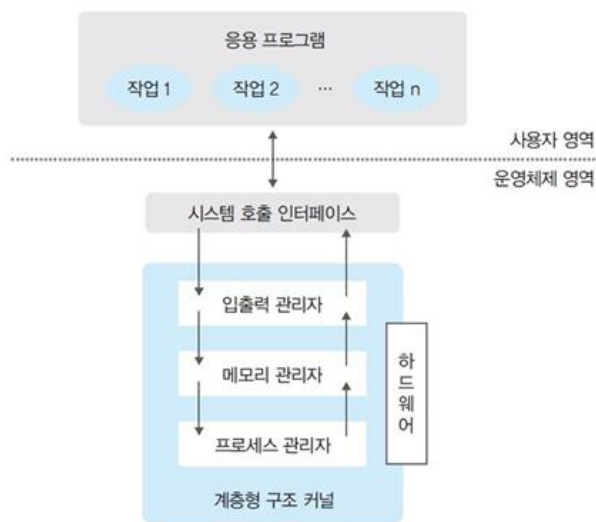
단점: 모든 모듈 하나로 묶여서 버그나 오류 처리하기 어려움

운체 여러 기능이 서로 연결돼있어 상호 의존성 높음 기능상 작은 결함이 시스템 전체로 확산됨

다양한 환경의 시스템에 적용 어려움. 현대 운체는 크고 복잡해서 완전 단일형 구조 운체 구현하기 어려움

계층형 구조(layered architecture) 커널

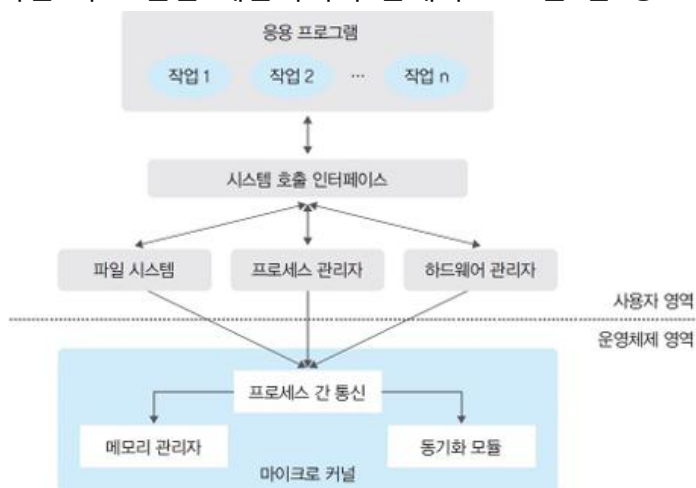
: 비슷한 기능 가진 모듈 묶어서 하나의 계층으로 만들고 계층 간 통신을 통해 운체 구현하는 방식



마이크로 구조(micro architecture) 커널

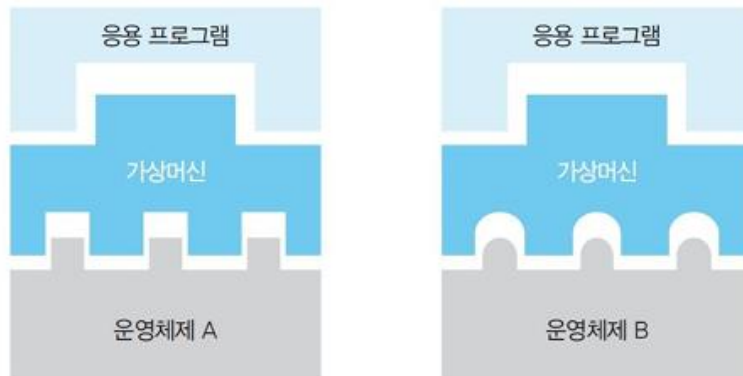
: 프로세스 관리, 메모리 관리, 프로세스 간 통신 관리 등 가장 기본적인 기능만 제공

커널 각 모듈은 세분화되어 존재하고 모듈 간 정보 교환은 프로세스 간 통신을 이용

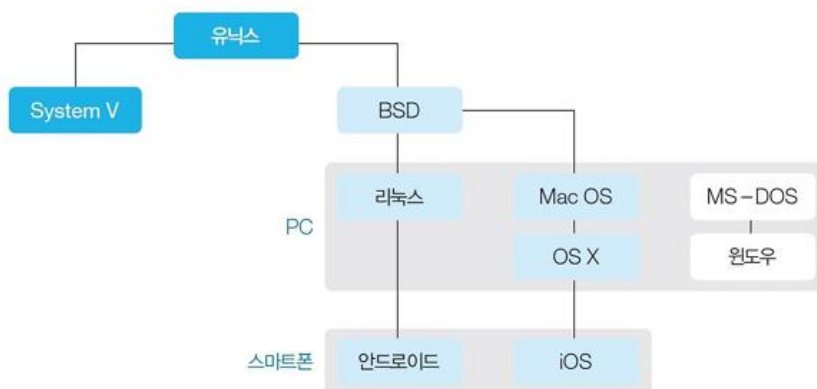


## 가상 머신

: 운체와 응프 사이에서 작동하는 프로그램. 가상머신 설치하면 응프가 모두 동일 환경에서 작동하는 것처럼 보임. 자바는 유닉스와 윈도우에서 작동하는 다양한 가상머신을 만들어 배포하는데 이를 자바 가상머신이라 함 (Java Virtual Machine; JVM)



다양한 유닉스 버전의 개발~



## 유닉스

: 유닉스는 켄 톰프슨, 데니스 리치, 피터 뉴만이 만든 운체로 단순하다는 의미 가짐. 이식하기 쉬워서 인기 얻음. 개발 후 소스코드 공개돼 계속 다른 기종 컴퓨터에 이식, 여러 기업과 대학에서 연구 진행되어 다양한 기능 추가

## GNU와 리눅스

: 1991 리누스 토르발스가 PC에서 동작하는 유닉스 호환 커널 작성해서 GPL로 배포, 소스코드 공개한게 리눅스 시작

## 애플II

: 1977 애플에서 만들. 개인용 컴퓨터 시초로 키보드, 메인보드, 전원장치가 하나로 합쳐

진 일체형 모델. 전문가만 사용하던 고가 컴퓨터를 일반 대중도 사용할 수 있게 되고  
hw sw 개발로 이어짐

Mac Os와 iOS

: 마하 커널 기반 Mac OS 사용 컴퓨터 등장. 강점은 마우스 이용하는 GUI 처음 도입

맥오에스 수정한 iOS가 적용된 스마트폰 아이폰 출시. 아이폰은 세계 최초로 멀티터치스  
크린 장착. (사용자 인페보다 더 사용자친화적+경험 중심 기술 UX(User eXperience))

윈도우 출시

: 마소는 애플 맥오에스 GUI에 자극받아 윈도우 운체 출시하고 계속 업글. 현재 버전11

스마트폰 등장

: 애플 아이폰이 스마트폰 시장 장악, 다른 회사에서도 개발. 아이폰은 자사 운체 iOS 사  
용, 그 외 회사는 자체 개발과 시간 비용 많이 들가서 구글 모바일 운체 안드로이드 사  
용

iOS

: 기존 Mac OS 커널이 지원 안 하는 코어 프레임워크(Core Framework)에서 지원

코.프는 GPS, 자이로스코프 센서, 가속도 센서, 나침반과 같은 스마트폰에 추가된 hw접근  
관리 담당

안드로이드(Android)

: 구글이 주도하는 Open Handset Alliance가 설계한 스마트폰 및 태블릿용 운체. iOS가  
애플에서 만들어지는 기기 전용으로 개발 + 소스코드 공개X와 달리 소스코드 공개. 빠르  
게 확산됨

리눅스 커널과 자바 사용해 호환성 매우 뛰어남