

운영체제의 기초 활용하기(2001020216_15v3.1)

□ 운영체제의 개념 ?

하드웨어를 제어하고 컴퓨터 자원을 관리하며, 컴퓨터 사용을 좀 더 편리하게 해주며, 응용 프로그램의 수행을 도와주고, 사용자와 하드웨어 사이의 매개체 역할

□ 운영체제의 역사

년도	처리방식	내용
1940년대	초기 전자식 디지털 컴퓨터	<ul style="list-style-type: none"> • 운영체제가 존재하지 않음 • 단지 기계적 스위치에 의해 작동
1950년대	단순 순차처리	<ul style="list-style-type: none"> • 한 번에 오직 하나의 작업만을 수행 • IBM 701용 운영체제개발(GM)
1960년대	다중 프로그래밍	<ul style="list-style-type: none"> • 1964년 IBM S/360 운영체제 개발 • 멀티프로그래밍, 멀티프로세싱, 시분할처리 개념 대두 • 멀티 대화식 사용자, 멀티모드 시스템의 보편화
1970년대	다중모드 시분할	<ul style="list-style-type: none"> • 멀티모드 시분할 시스템(알칼처리, 시분할처리, 실시간처리)의 보편화 • 근거리 지역 네트워크(이더넷(Ethernet))(표준에 의해 실용화) • 정보보호 및 보안문제에 종대에 암호화 중요성 대두
1980년대	분산네트워크	<ul style="list-style-type: none"> • 운영체제 기능들이 하드웨어에 포함된 펌웨어 개념의 대두 • 개인용 컴퓨터와 워크스테이션, 강력한 데스크톱 컴퓨터의 구축가능 • 각종 응용 프로그램의 개발 및 데이터베이스 활용 확대 • 네트워크를 기반으로 클라이언트/서버(Client/Server)모델의 확대
1990년대	병렬계산과 분산계산	<ul style="list-style-type: none"> • 중앙집중이 아닌 분산으로 발전 • 네트워크와 멀티미디어 처리 기술의 발달 • 인터넷 보급의 급속한 확산 • 그래픽 사용자 인터페이스(GUI)의 강화 • 선점형, 멀티태스킹, 멀티스레딩, 가상메모리의 보편화 • PC용 운영체제(Windows, Mac), 서버용 운영체제(Unix, Linux)의 보편화
2000년대	모바일 및 임베디드 운영체제	<ul style="list-style-type: none"> • 시스템을 초고속화, 고기능화, 초경량화 방향으로 발전 • 다양한 통신망의 확대와 개발형 시스템의 발달 • 다양한 기능, 확장성과 호환성의 극대화 • 네트워크 기반의 분산 및 병렬운영체제의 보편화 • PDP, PMP 등의 모바일 장치와 가전제품을 위한 모바일 및 임베디드 운영체제의 보편화 • 64비트 CPU에 호환되는 64비트용 운영체제

※ 일괄 처리 시스템(Batch Processing System) ?

일정량의 데이터를 모아서 한꺼번에 처리하는 방식

※ 다중 프로그래밍 시스템(Multi Programming System) ?

하나의 CPU와 주기억장치를 이용하여 여러 개의 프로그램을 동시에 처리하는 방식

※ 시분할 시스템(Time Sharing System) ?

- 여러 명의 사용자가 사용하는 시스템에서 컴퓨터가 사용자들의 프로그램을 번갈아 가며 처리해 줌으로써 각 사용자에게 독립된 컴퓨터를 사용하는 느낌을 받음
- 다중 프로그래밍 방식과 결합하여 모든 작업이 동시에 진행되는 것처럼 대화식 처리가 가능
- 시스템의 전체 효율은 좋아지나 개인별 사용자 입장에서는 반응 속도가 느려질 수 있음

※ 다중 처리 시스템(Multi Processing System)?

- 여러 개의 CPU와 하나의 주기억장치를 이용하여 여러 개의 프로그램을 동시에 처리하는 방식
- 여러 CPU는 하나의 메모리를 공유함으로 단일 운영체제에 의해 관리
- 프로그램의 처리 속도는 빠름
- 기억장치, 입/출력 장치 등의 자원 공유에 대한 문제 관리가 어려움

※ 실시간 처리 시스템(Real Time Processing System)?

데이터 발생 또는 데이터에 대한 처리 요구가 있는 즉시 처리하여 응답해주는 시스템

※ 분산 처리 시스템(Distributed Processing System)?

여러 개의 컴퓨터를 통신 회선으로 연결하여 하나의 작업을 처리하는 방식

□ 운영체제의 성능 평가

- ① 처리능력(Throughput) : 일정 시간 내에 시스템이 처리하는 일의 양
- ② 응답 시간 또는 반환시간(Turnaround time) : 시스템에 작업을 의뢰한 시간부터 처리가 완료될 때까지 걸리는 시간
- ③ 사용 가능도(Availability) : 시스템을 사용할 필요가 있을 때 즉시 사용 가능한 정도
- ④ 신뢰도(Reliability) : 시스템이 주어진 문제를 정확하게 해결하는 정도

□ 운영체제의 종류

- 유닉스, 윈도우즈, OS X, 안드로이드, iOS
- Dos
디스크 입출력을 포함한 OS, 시스템 소프트웨어를 말한다. 대표적인 DOS에

MS-DOS 또는 DR-DOS 등이 있다.

dir (디렉토리 내용보기명령어)

cd [이동하고자 하는 드라이브명\경로] (디렉토리 위치 변경명령어)

copy [복사하고자 하는 파일명] [복사할 장소의 드라이브명\경로] (복사 명령어)

데이터베이스 기초 활용하기(2001020216_15v3.2)

☐ 파일 시스템의 문제점

① 데이터 종속성(Data Dependency)

② 데이터의 중복성(Data Redundancy)

한 시스템 내 같은 내용의 데이터가 다수의 파일에 중복 저장 시 모든 파일을 수정해야 한다. 수정이 완벽하지 못할 경우 데이터 불일치성(Data Inconsistency)이라는 문제도 발생할 수 있다.

☐ 데이터베이스 시스템의 장점

- ① 데이터 중복 최소화
- ② 여러 사용자에게 의한 데이터 공유(데이터의 공유성)
- ③ 데이터의 일관성
- ④ 데이터의 무결성
- ⑤ 정확한 최신의 정보이용 가능
- ⑥ 데이터 처리속도 증가
- ⑦ 많은 양의 종이파일 간소화

☐ 데이터베이스 시스템 특징

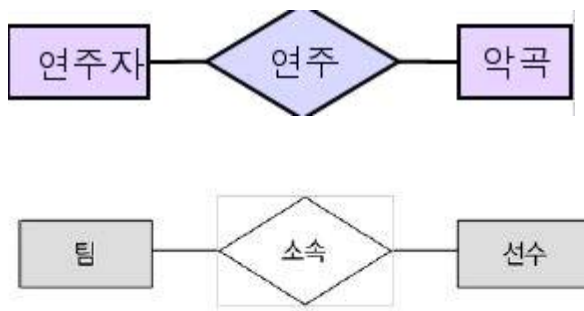
- ① 실시간 접근성: 사용자의 질의에 대하여 즉시 처리하여 응답
- ② 계속적인 진화: 삽입, 삭제, 갱신을 통하여 항상 최근의 정확한 데이터를 동적으로 유지
- ③ 동시 공유(concurrent sharing): 여러사용자들이 데이터를 사용
- ④ 내용에 의한 참조: 데이터베이스에 있는 데이터를 참조할 때 튜플(tuple)의 주소나 위치가 아닌 사용자가 요구하는 데이터 내용에 따라 참조
- ⑤ 데이터 논리적 독립성: 응용프로그램과 데이터베이스를 독립시킴으로써 데이터의 논리적 구조를 변경시키더라도 응용프로그램은 변경되지 않는 특징을 갖는다.

☐ 개체-관계 모델

"구조"화된 데이터를 저장하기 위해 데이터베이스를 쓴다. 이 데이터의 "구조" 및 그에 수반한 제약 조건들은 다양한 기법에 의해 설계될 수 있다. 그 기법 중 하나가 개체-관계 모델링(Entity-Relationship Modelling)이다. 줄여서 ERM이라고 한다.

□ 전통적인 개체-관계 다이어그램(Entity-Relationship Diagram)

두 개의 엔티티와 관계

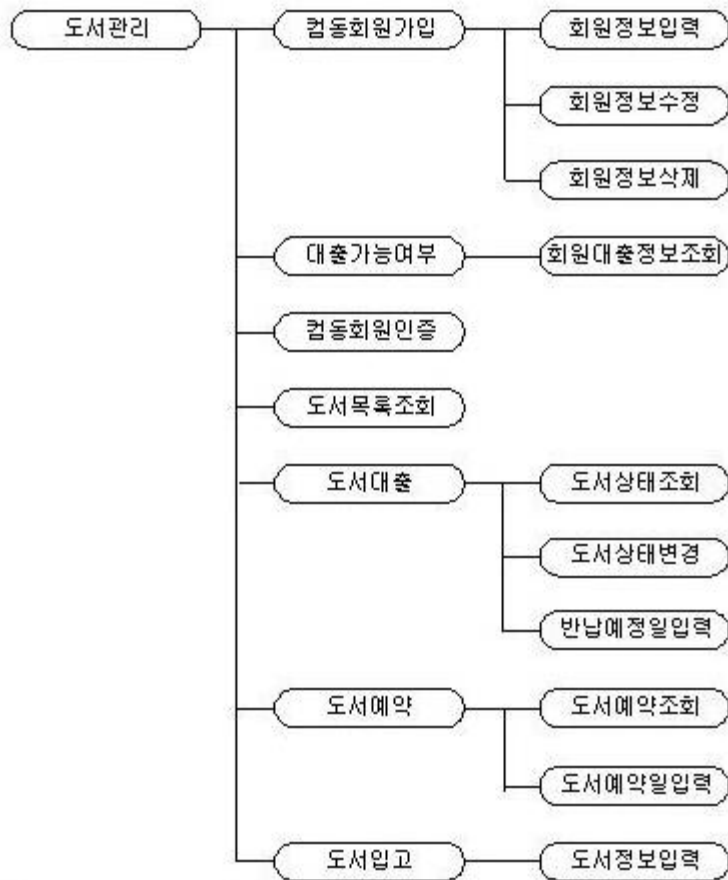


□ CRUD 매트릭스

CRUD 상관 모델링은 데이터 모델과 프로세스 모델과의 상관관계를 정의함으로써 정확한 데이터 모델이 도출되었는가를 검증하는 것이다. 상식적으로 생각해보아도 쓰지 않는 데이터가 저장되어 있다는 것은 잘못된 데이터 모델링을 하는 것이다. CRUD의 의미는 Create, Update, Read, Delete를 의미한다. SELECT, UPDATE, INSERT, DELETE 다. SELECT는 “R”에 해당되고, UPDATE는 당연히 “U”에 해당된다. INSERT는 “C”에 해당되고, DELETE는 “D”에 해당된다.

아래는 도서관리 기능 분해도이다.

기능분해도를 통해 CRUD 매트릭스 를 만들었다



엔티티집합 단위프로세스	도서	컴동회원	대출	예약
회원정보입력		C		
회원정보수정		U		
회원정보삭제		D		
회원대출정보조회	R	R	R	
컴동회원인증		R		
도서목록조회	R			
도서상태조회	R	R		
도서상태변경	U			
반납예정일입력			C	
도서예약조회	R	R		R
도서예약일입력				C
도서정보입력	C			

각 엔티티에는 create 또는 read 기능이 있어야 한다. 아래 매트릭스에서는 컴동회원 엔티티에 create 기능이 없는 것을 체크할 수 있다. 따라서 단위프로세서 중 하나가 빠진 것을 알 수 있다.

엔티티집합 단위프로세스	도서	컬동회원	대출	예약
회원대출정보조회	R	R	R	
컬동회원인증		R		
도서목록조회	R			
도서상대조회	R	R		
도서상대변경	U			
반납예정일입력			C	
도서예약조회	R	R		R
도서예약일입력				C
도서정보입력	C			

CRUD 매트릭스를 통해

- Create가 존재하지 않는 엔티티 집합이 있는가?
- Read가 존재하지 않는 엔티티 집합이 있는가?
- 매트릭스에 C, R, U, D 가 하나도 표시되지 않은 엔티티 집합이 있는가?
- 매트릭스에 C, R, U, D 가 하나도 표시되지 않은 프로세스가 있는가?

체크할 수 있다

네트워크 기초 활용하기(2001020216_15v3.3)

: 전자신호를 통해 통신하는 모든 기기가 서로 통신하기 위해 만든 하나의 망 (Computer Network 혹은 Data Network)

1 통신망의 종류

: 광역 통신망, 도시권 통신망, 근거리 통신망, 개인 통신망

2. 통신망 구축형식

: 링 형, 별형, 트리 형, 그물형, 버스형

3. 프로토콜 : 네트워크 통신규약

3.1 OSI(Open System Interconnection) 참조 모델 7계층

서로 다른 시스템 간의 원활한 통신을 위해 ISO(국제표준화기구)에서 제안한 통신 규약으로, 7단계로 표준화 하여 규정

⑦ 응용 계층 : 사용자(사람 또는 소프트웨어)가 네트워크에 접근할 수 있도록 함. 사용자 인터페이스를 제공하고, 전자우편, 원격 파일접근과 전송, 공유 데이터베이스 관리 및 여러 종류의 분산정보 서비스를 제공함

⑥ 프레젠테이션(표현) 계층: 두 시스템 사이에서 교환되는 정보의 구문과 의미에 관련되어 변환, 압축 및 암호화를 담당함

⑤ 세션 계층: 통신하는 시스템들 사이의 상호작용을 설정 · 유지하고 동기화

④ 트랜스포트(전송) 계층 : 전체 메시지의 프로세스-대-프로세스 전달을 함
네트워크 종단(end)시스템간의 데이터를 일관성 있게 전송(TCP, UDP)

③ 네트워크 계층 : 패킷을 발신지로부터 여러 네트워크(링크)를 통해 목적지까지 전달함
통신 트래픽의 흐름을 제어, 통신 시스템간의 경로 설정 및 네트워크 연결 관리

② 데이터 링크 계층 : 2개 인접된 호스트(Host) 간에 데이터의 전송을 행하고 전송에러를 제어한다. 신뢰성 있고 효율적인 프레임 데이터 전송

① 물리 계층 : 물리적인 장치와 인터페이스가 전송을 위해 필요한 기능과 처리절차를 규정함

장치와 전송매체 간의 인터페이스 특성 규정, 전송 매체의 유형 규정, 전송로의 연결 · 유지 및 해제를 담당

①, ②, ③ - 네트워크 지원계층

(한 장치에서 다른 장치로 데이터를 이동할 때 필요한 물리적인 면(즉, 전기적인 규격, 물리적인 연결, 물리주소, 전송시간과 신뢰도 등) 처리)

④ - 전송층

(종단-대-종단까지의 믿을 만한 데이터 전송 보장)

⑤, ⑥, ⑦ - 사용자 지원계층

(서로 상관없는 소프트웨어 시스템 사이의 상호연동을 가능하게 함)

3.2 TCP/IP (Transmission Control Protocol/Internet Protocol) : 4계층 레이어

①네트워크 접근 계층 : OSI 물리 계층과 데이터 링크 계층에 해당

사용 프로토콜 : EtherNet , token Ring , ppp

②인터넷 계층 : 네트워크 계층에 해당 , 주소 지정, 경로 설정 및 라우팅 기능

IP 주소(address) 체계이다.

- IPv4는 현재 인터넷 및 TCP/IP 네트워크에서 활용하는 IP 주소(address) 체계이며, IPv4의 주소 체계는 총 4바이트(32비트)로 표시한다.
- IPv6는 IPv4의 IP 주소 부족 및 보안 문제를 해결하기 위해 나온 차세대 IP 표준이다.
- 하나의 IP 주소는 크게 네트워크 주소와 컴퓨터 주소로 나뉘며, 네트워크의 크기나 호스트 컴퓨터의 수에 따라 클래스(Class) A, B, C, D, E 등급으로 나뉜다. 이중 클래스 A, B, C가 일반 사용자에게 부여된다.
- IPv4가 생성할 수 있는 주소 자원은 총 43억개이며, 이 가운데 우리나라는 약 4,100만 개의 주소 자원을 확보하고 있다.

사용 프로토콜 : IP(Internet Protocol) , ICMP(Internet Control Message Protocol) , IGMP(Internet Group Management Protocol), ARP(Address Resolution Protocol) , RARP(Reverse Address Resolution Protocol)

③전송 계층(Layer3) : 호스트들 간의 신뢰성 있는 통신 제공

사용 프로토콜 :

- TCP(Transmission Control Protocol) -데이터 전달의 신뢰성을 위해 연결성 방식을 사용
- IP 프로토콜 위에서 연결형 서비스를 지원하는 전송계층 프로토콜로, 인터넷 환경에서 기본으로 사용한다.
- 전이중(Full Duplex) 방식의 양방향 가상 회선을 제공한다.
- 신뢰성 있는 데이터 전송을 보장한다.
- 연결형 서비스를 제공한다.

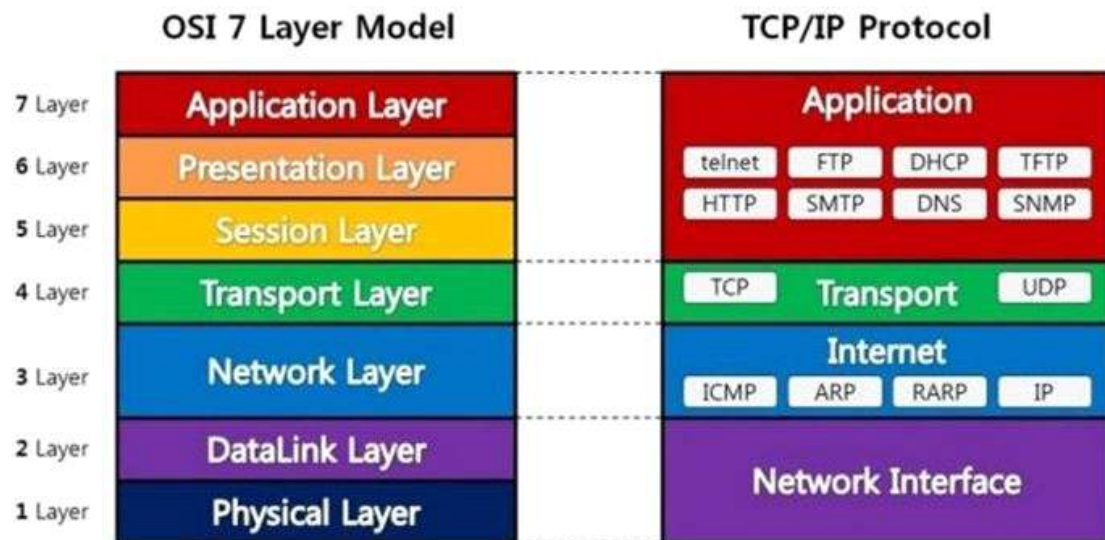
- UDP(User Datagram Protocol) -데이터의 전달을 위해 비연결성 방식을 사용

④응용 계층 : 응용 프로그램 간의 데이터 송·수신 제공

사용 프로토콜 : FTP(File Transfer Protocol) : 파일 전송 프로토콜, SMTP(Simple Mail Transfer Protocol) : 전자우편을 전송할 때 이용되는 표준 프로토콜, SNMP(Simple Network Management Protocol) : 간이 망 관리 프로토콜, TELNET(TELEcommunication NETWORK) : 원격지의 호스트 컴퓨터에 접속할 때 지원되는 표준

프로토콜

[OSI 7계층]



4. 인터넷

인터넷(Internet)은 인터넷 프로토콜 TCP/IP를 기반으로 하여 전 세계적으로 연결된 컴퓨터 네트워크를 일컫는 말이다

기본 개발환경 구축하기(2001020216_15v3.4)

윈도우 환경 하에서 자바 응용프로그램을 개발하기 위해 설치해야 할 설치 도구를 순서대로 기술하시오.

<http://www.oracle.com/technetwork/java/javase/downloads/index.html> 사이트에서 다운로드 하고

<https://www.eclipse.org/downloads/> 사이트에서 이클립스를 다운로드 한다.