



LINUX

MM4220 게임서버 프로그래밍
정내훈

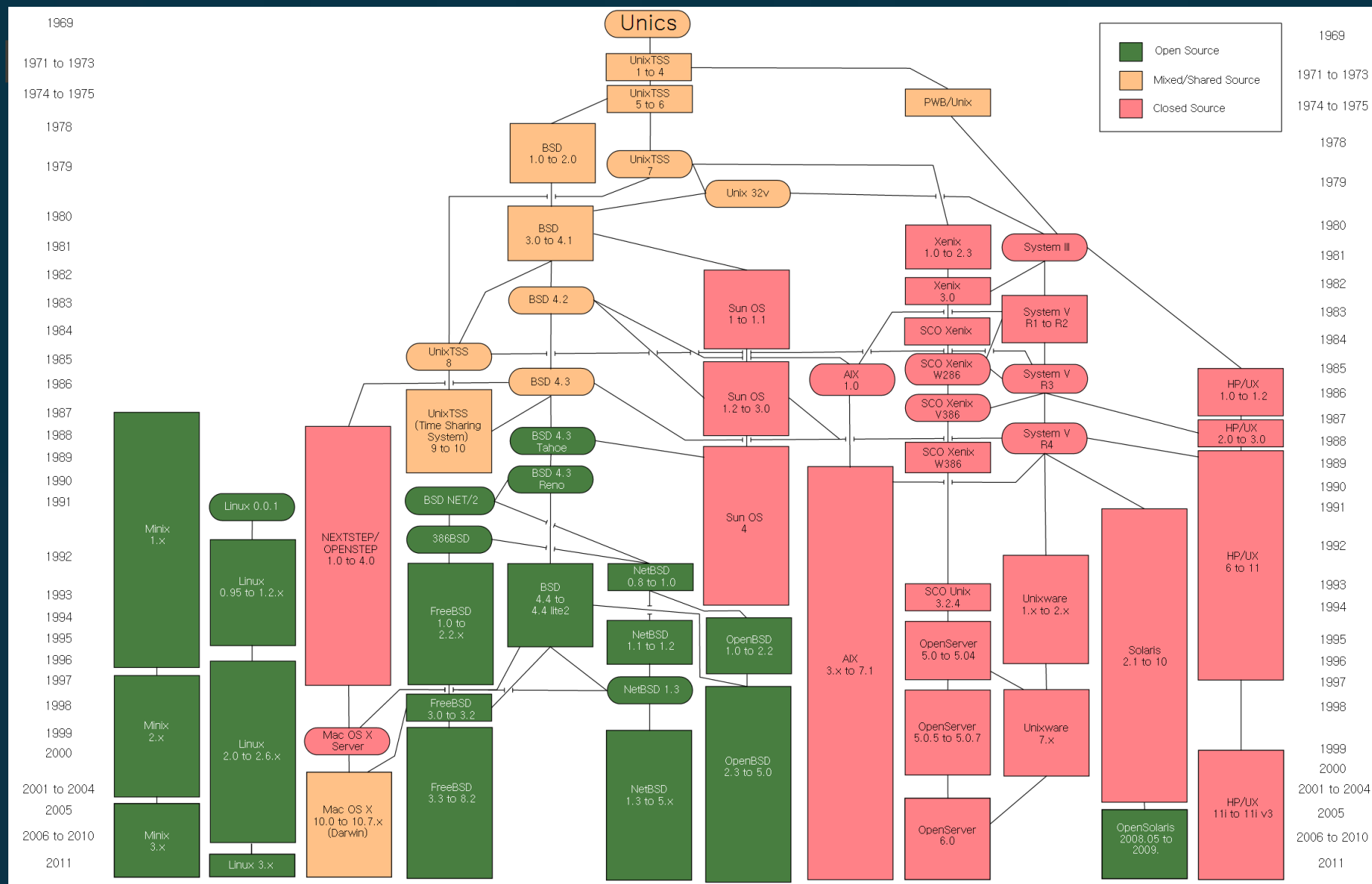
내용

- LINUX
- LINUX & Game
- 실습

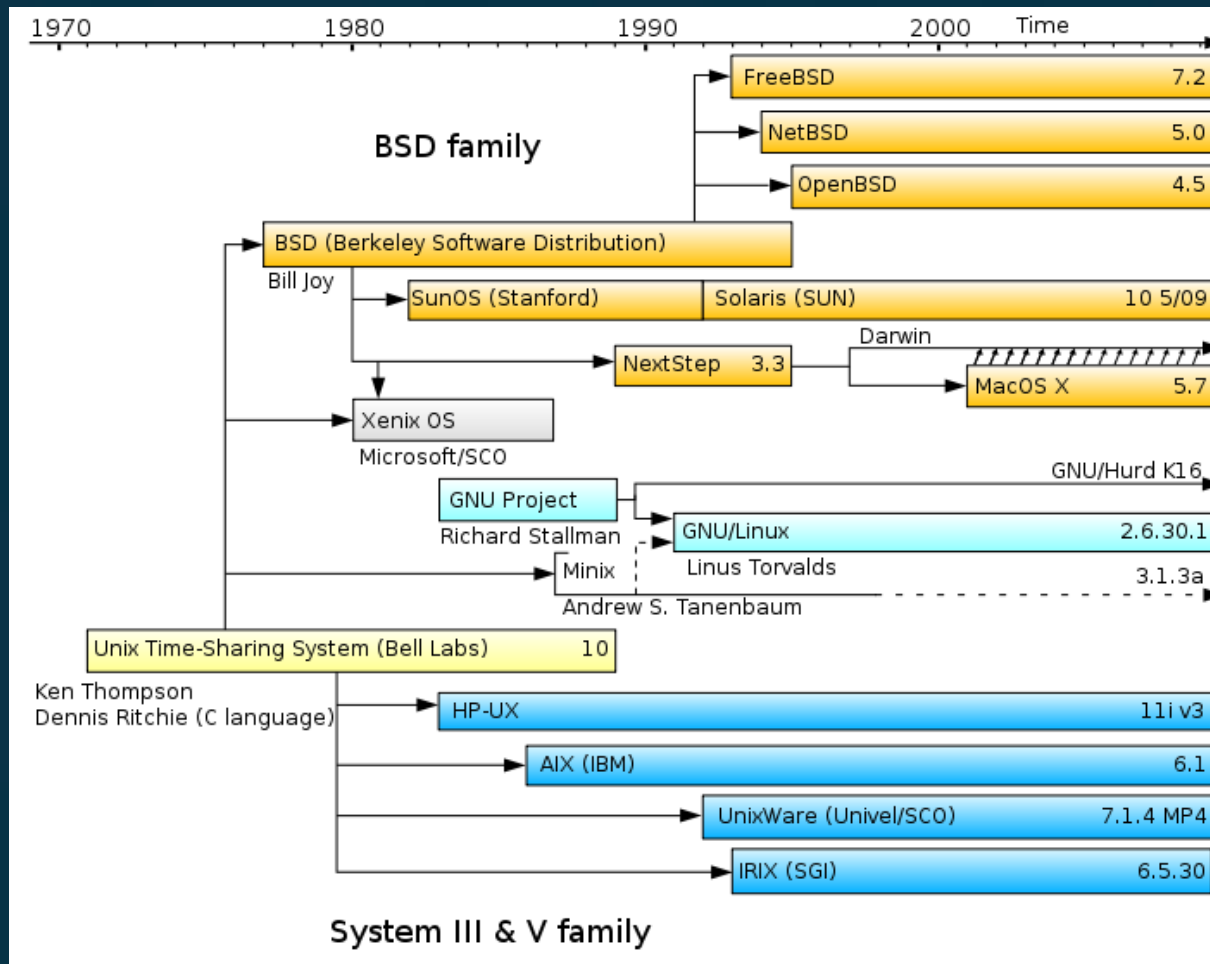
LINUX

- UNIX

- 1969년 개발
- Multi Process Operating System
- Text기반의 UI
 - X-Window로 window환경 진화 (1980년대 후반)
- 현대 OS의 모태
 - 현대OS에 추가된 개념들
 - Multi thread, network programming, windows system...
- <https://m.facebook.com/engineertoont/photos/pcb.512251568961703/512249205628606/?type=3&source=48>
 - (<http://www.ddanzi.com/ddanziNews/92939697>)



LINUX



- 인터넷 서버 점유율

Note

Note

W3Cook checks the top **one** million web servers monthly, taken from the Alexa ranking, using HTTP headers, DNS records, and WHOIS data, among other sources.^[118]

LINUX

- 게임서버에서의 리눅스 사용비율

- <http://www.gpgstudy.com/forum/viewtopic.php?p=114119>
- 클라우드와 중국의 영향으로 상승 중

온라인 서버 제작자 모임(<http://cafe.naver.com/ongameserver>)에서 2009.01.19 ~ 2009. 01.31일까지 현재 온라인 게임 서버의 개발 및 서비스를 할 때 사용하는 OS에 대한 설문 조사를 했습니다.

총 85명이 참여 해 주셨습니다.

아주 많은 수의 개발자가 설문에 참여한 것은 아니지만 각 OS 사용 비율은 한국 게임업계의 상황과 거의 비슷하리라 생각합니다.

* 개발은 현재 개발 중인 게임의 게임 서버를 실행할 OS이고, 운영은 서비스 중인 게임의 게임 서버를 실행하고 있는 OS를 말합니다.

32비트 윈도우 65명
64비트 윈도우 10명
32비트 유닉스 계열 5명
64비트 유닉스 계열 5명

LINUX

- 배포판
 - Open Source로 나와 있는 것은 Kernel만
 - Device driver와 UI, utility, 개발툴을 묶어서 배포판 형태로 배포
- 유명
 - UBUNTU : desk top에서 많이사용
 - RED HAT : 서버 전용, 유료, 버그수정해 줌
 - Cent OS : Red Hat의 공짜, 아무런 지원이 없음

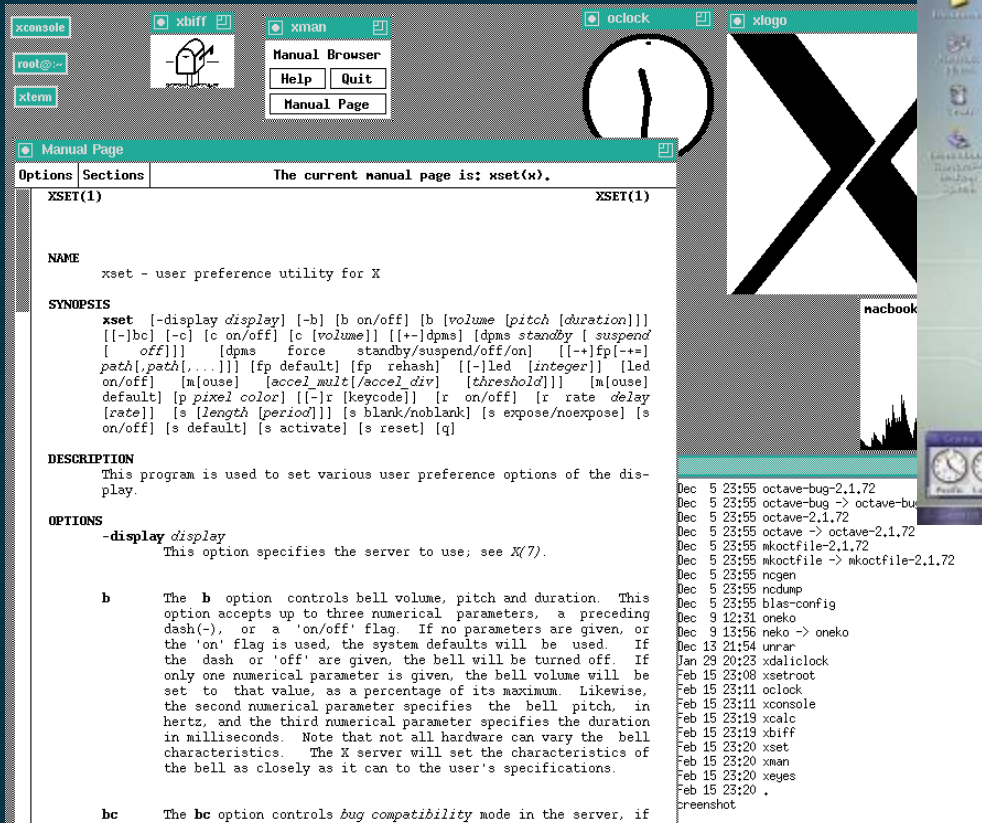
LINUX

- UI

```
drwxr-xr-x  2 root root  4096 Oct 16  2007 opt
dr-xr-xr-x 153 root root      0 May 27 11:36 proc
drwxr-xr-x 15 root root  4096 Feb  7 10:56 root
drwxr-xr-x  2 root root  4096 Apr 20 17:57/sbin
drwxr-xr-x  2 root root  4096 Mar  7  2009 selinux
-rw-----  1 root root 31903 Dec 10  2008 sqlc3xs0V
drwxr-xr-x  2 root root  4096 Oct 16  2007 srv
drwxr-xr-x 12 root root      0 May 27 11:36 sys
drwxrwxrwt  8 root root 12288 Jun  3 14:57 tmp
drwxr-xr-x 14 root root  4096 Nov 10  2009 usr
drwxr-xr-x 17 root root  4096 Oct 24  2008 var
lrwxrwxrwx  1 root root      30 Mar 20 11:24 vmlinuz -> boot/vmlinuz-2.6.31-23-generic
lrwxrwxrwx  1 root root      30 Jun  8  2010 vmlinuz.old -> boot/vmlinuz-2.6.31-23-generic
-rw-r--r--  1 root root  2533 May 14  2009 xorg.conf.new
guest@taon:~$ who
ken      pts/1      May 27 11:39 (192.168.10.148:S.0)
ken      pts/2      May 27 11:39 (192.168.10.148:S.1)
ken      pts/3      May 27 11:39 (192.168.10.148:S.2)
ken      pts/4      May 28 00:11 (192.168.10.148:S.3)
ken      pts/5      May 30 11:30 (192.168.10.148:S.4)
guest    pts/6      Jun  3 14:56 (210.93.61.41)
guest@taon:~$
```

LINUX

• UI의 진화



<http://en.wikipedia.org/wiki/Unix>

LINUX

- 장점
 - 가격이 싸다
 - 가볍다.
 - Open Source이다.
- 단점
 - 익숙해 지기가 어렵다.

LINUX

- 개발환경
 - 컴파일러
 - GCC, G++
 - 디버거
 - GDB
 - 통합개발환경
 - QT Creator, 이클립스 등등.. VS-CODE
 - 매뉴얼
 - man
 - DataBase
 - MySQL, NoSQL

LINUX(2019-수목)

- Windows프로그래밍과의 차이
 - 언어
 - C++ 사용, C++11을 사용할 경우 큰 차이 없음
 - UI
 - Windows프로그래밍 vs X Window프로그래밍
 - Multithread programming
 - POSIX thread library 사용 => C++11 사용
 - Network
 - IOCP없음
 - epoll이나 kqueue를 사용 => boost/asio사용

LINUX

- Windows 프로그래밍과의 차이
 - Databse
 - Mysql을 많이 사용
 - www.mysql.org
 - ODBC사용 가능
 - 직접 **socket** 프로그래밍으로 접속하기도 함.
 - REDIS같은 NoSQL은 리눅스가 대세

LINUX

- EPoll

epoll 은 변경기준감지(Edge Triggered) 또는 상태기준감지(Level Triggered) 인터페이스 양쪽 모두를 사용할 수 있도록 하여 많은 수의 fds를 감시할 수 있도록 확장시킨 poll(2) 의 변형이다. epoll 기능에 대한 설정 및 제어를 위해 다음과 같은 3가지 시스템 함수가 제공된다: `epoll_create(2)`, `epoll_ctl(2)`, `epoll_wait(2)`.

파일 지시자와 연결하기위한 epoll 집합은 `epoll_create(2)` 로 생성할 수 있다. 그런다음 특정 파일 지시자를 `epoll_ctl(2)` 을 통하여 등록할 수 있다. 마지막으로 `epoll_wait(2)` 을 사용하여 (등록한 파일 지시자의 이벤트에 대한) 대기처리를 시작할 수 있다.

LINUX 실습

- SSH을 통한 접속
 - Windows에서 Linux 서버로 접속
 - SSH 클라이언트 필요
 - Putty.exe
 - <http://www.putty.org/>
- 주소 및 계정
 - 210.93.61.41 (SSH로 접속)
 - gameserver/게임서버

LINUX 실습

- 간단한 명령어들
 - 디렉토리 탐색 : `ls`
 - 디렉토리 : `cd`, `mkdir`, `rmdir`
 - 파일 조작 : `mv`, `cp`
 - 파일 내용 보기 : `cat`

LINUX 실습

- 에디팅

- VI (Visual Editor)

- Insert mode, cursor mode
 - ‘i’ + text + ‘esc’
 - ‘hjkl’ for cursor key
 - ‘x’ for delete, ‘dd’ for delete line

- ‘Hello World’ 작성 (각자 자기 디렉토리 작성 후 거기서)

```
#include <iostream>
int main()
{
    std::cout << "hello world!\n";
}
```

LINUX 실습

- 컴파일링
 - G++
 - ANSI C++ 표준
 - 'g++ hello.cc'
 - 실행
 - './a.out'

LINUX 실습

- 디버깅
 - 컴파일
 - `g++ -o hello -g hello.cc`
 - 디버깅
 - `gdb hello`
 - 커맨드
 - `list, break, step, start`

질문

- ???