

디스크 파티션 관리

안양대학교 컴퓨터공학과 하은용

내용

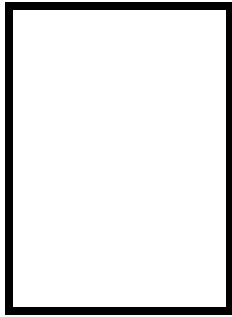
2

□ 디스크 파티션 관리

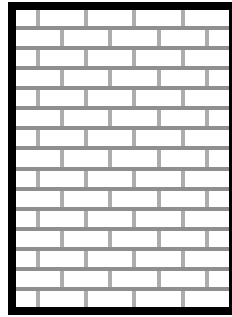
- https://docs.fedoraproject.org/en-US/fedora/f29/install-guide/appendixes/Disk_Partitions/#figu-partitions-different-file-system

디스크 드라이브

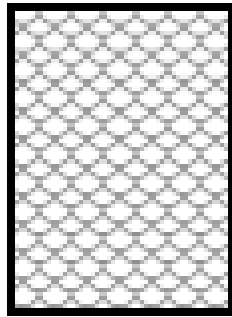
3



초기 디스크



파일 시스템 A

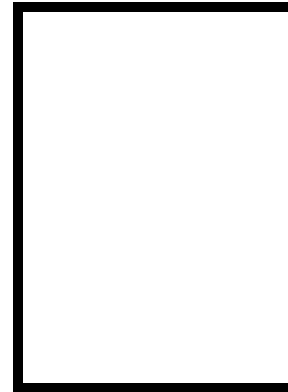
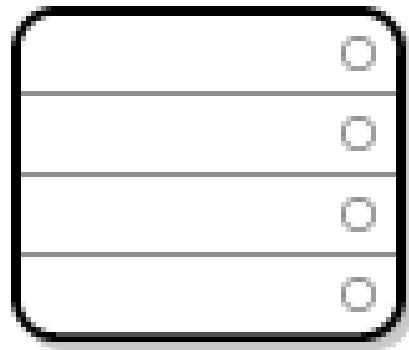


파일 시스템 B

파티션 테이블

4

□ 파티션 테이블



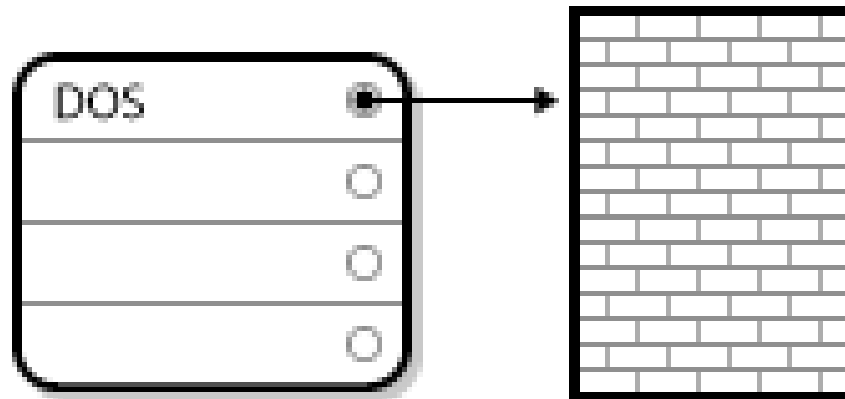
□ 표준 파티션 테이블

- ▣ **MBR** (Master Boot Record) : old layout, BIOS
- ▣ **GPT** (GUID Partition Table) : new layout, UEFI

파티션 테이블

5

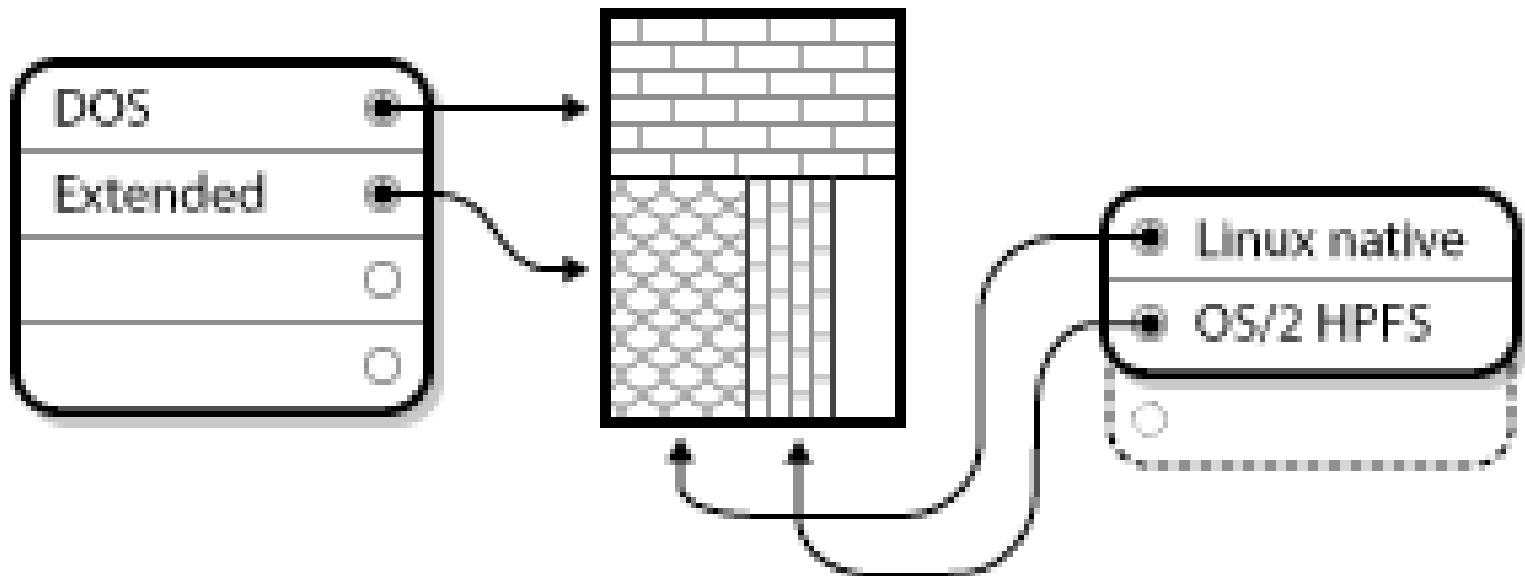
- 4 primary partitions
- primary partition : 1 logical drive
- PT entry
 - ▣ size : start and end points
 - ▣ status : active, inactive
 - ▣ type : number (DOS, Linux, ...)



확장 파티션 (extended partition)

6

- 4개 이상 파티션 구성 필요
- 확장 파티션 도입
- 여러 논리 파티션(logical partitions) 구성



논리 파티션 (*logical partition*)

7

- 이론적으로 논리 파티션 갯수 제한 없음
- 리눅스에서 12개까지 논리 파티션 지원

GPT (GUID PT)

https://en.wikipedia.org/wiki/GUID_Partition_Table

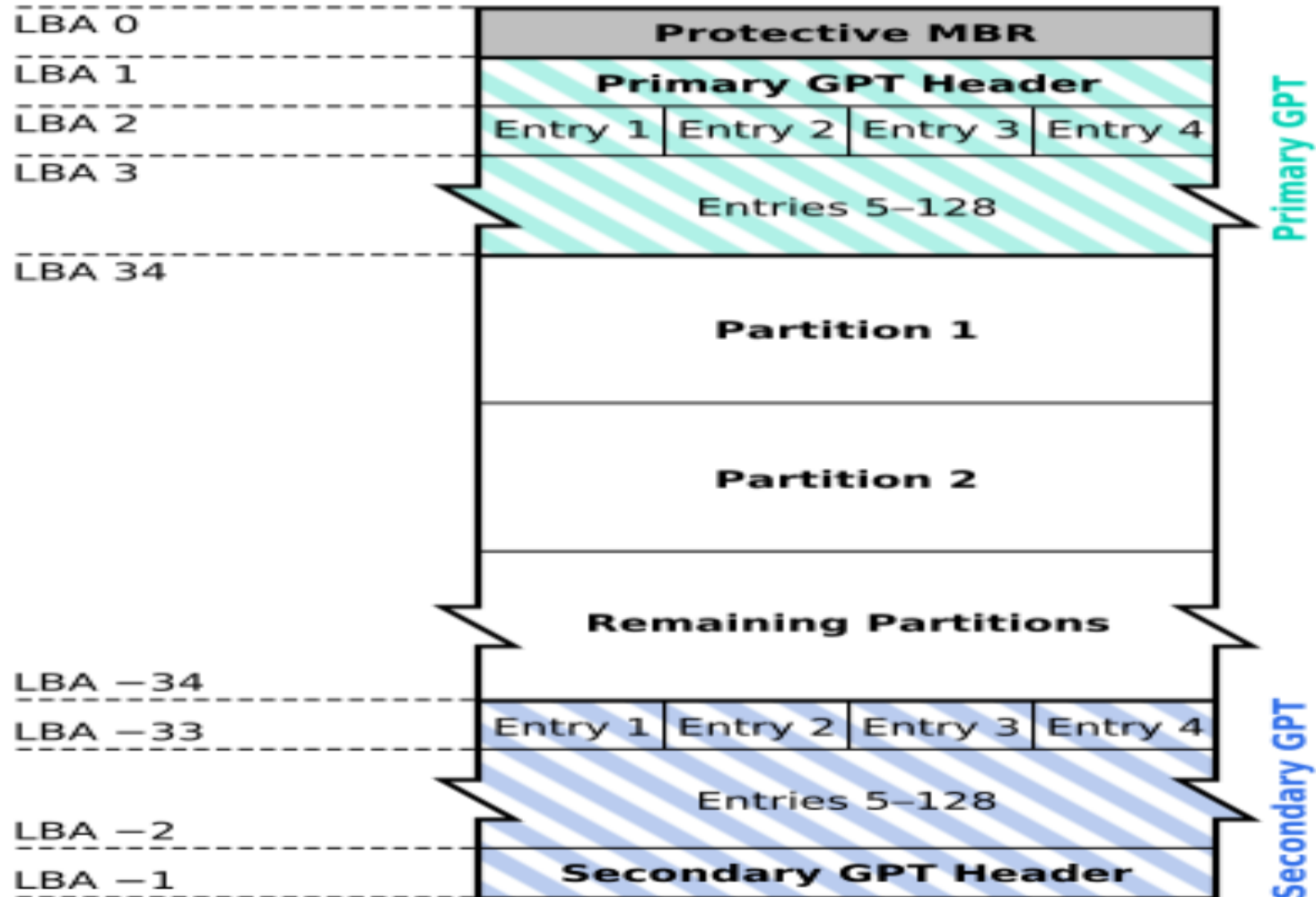
8

- **MBR : 2.2 TB 까지 가능**
 - ▣ 2^{32} sectors \times 512 bytes per sector = 2.2×1000^4 bytes
- **GPT : 9.4 ZB(zettabyte) 까지 가능**
 - ▣ 2^{64} sectors \times 2⁹bytes per sector = 9.4×1000^7 bytes
- **default 128 개 primary partition 지원**
- **파티션 테이블을 늘려 확장 가능**
- **LBA (logical block addressing) 사용**
- **LBA0 : MBR 지원**
- **LBA1 : primary GPT header**

GUID 파티션 테이블 구조

9

GUID Partition Table Scheme



primary GPT header

10

- disk GUID
- primary partition table 위치
- secondary GPT header 위치
- CRC32 checksums
- 테이블의 partition entries 개수

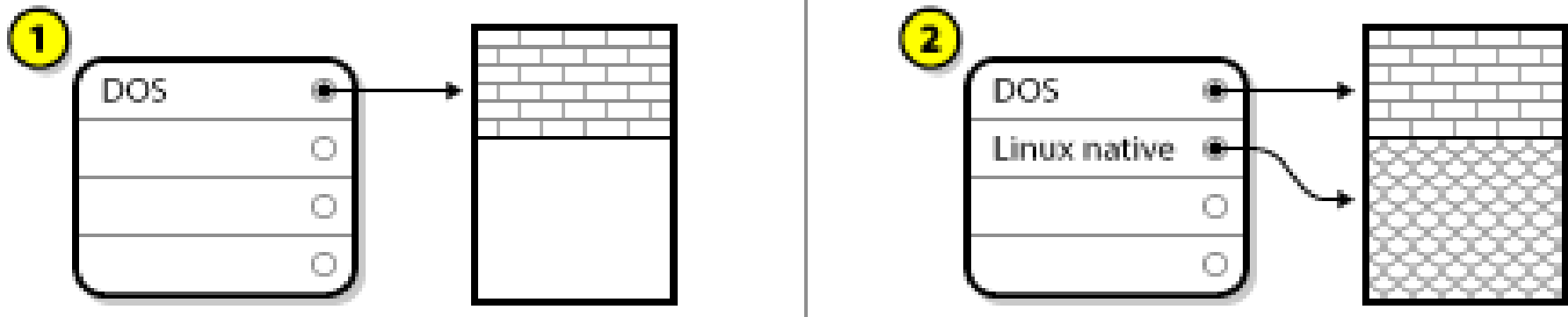
*secondary **GPT** table*

11

- **primary GPT table 동일**
- **복구용**

파티션 안된 Free Space 사용

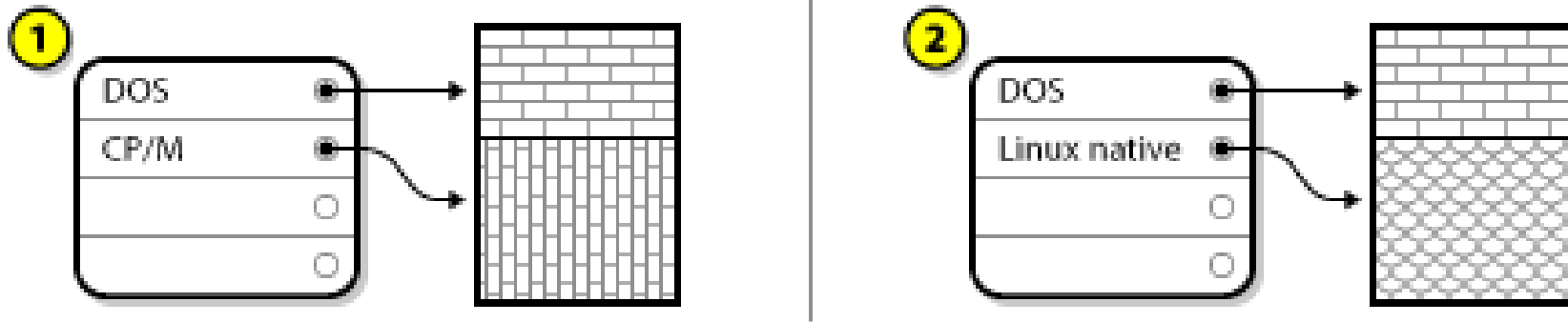
12



□ 할당 안된 자유 공간을 사용

미사용 파티션 Space 사용

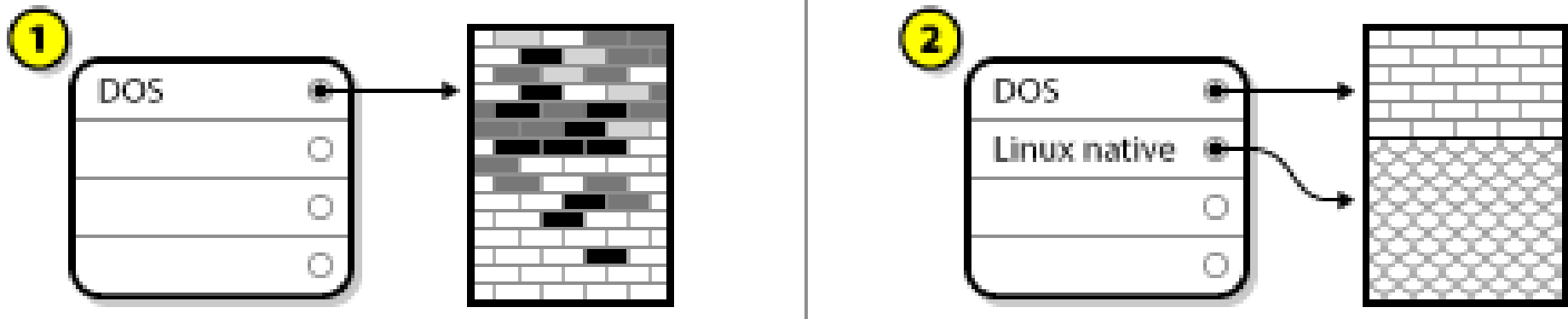
13



- CP/M 파티션을 재할당해서 사용

활성화된 파티션의 공간 사용

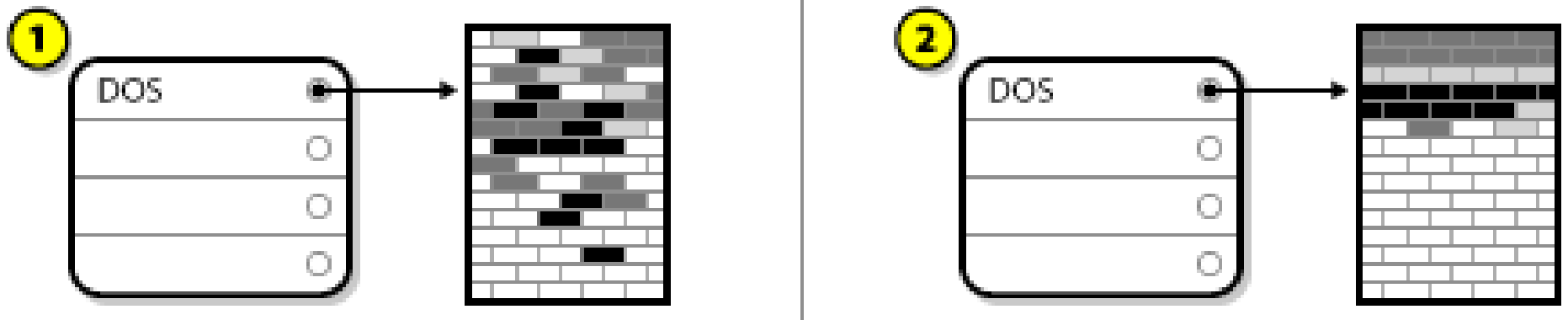
14



- 파괴적 방법으로 다시 파티션 구성
- 모든 데이터 사라짐

비파괴적 파티션 재구성

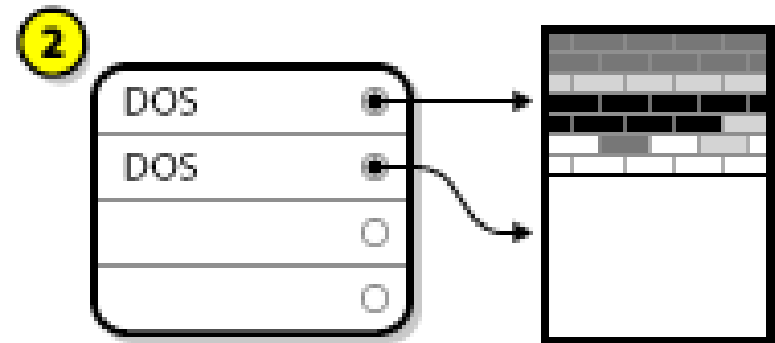
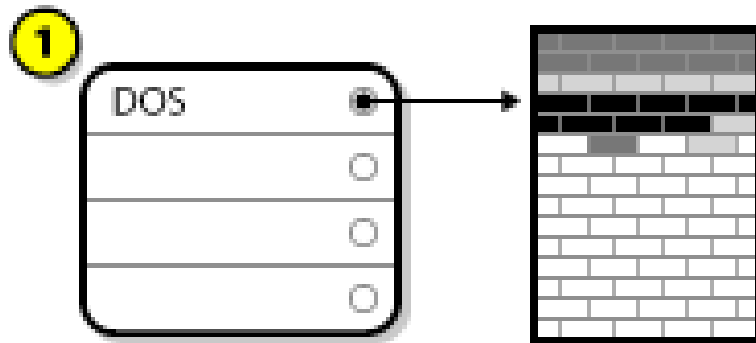
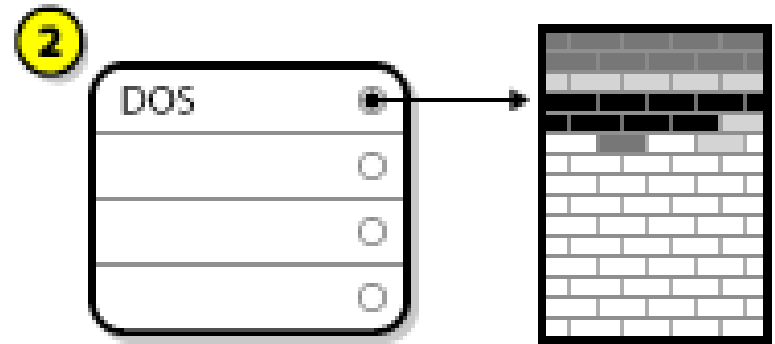
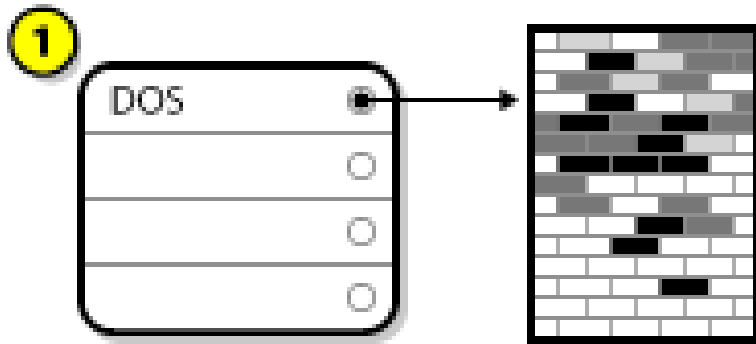
15



- 1) 파티션 정리 압축 후 데이터 백업
- 2) 파티션 크기 조정
- 3) 새 파티션 생성

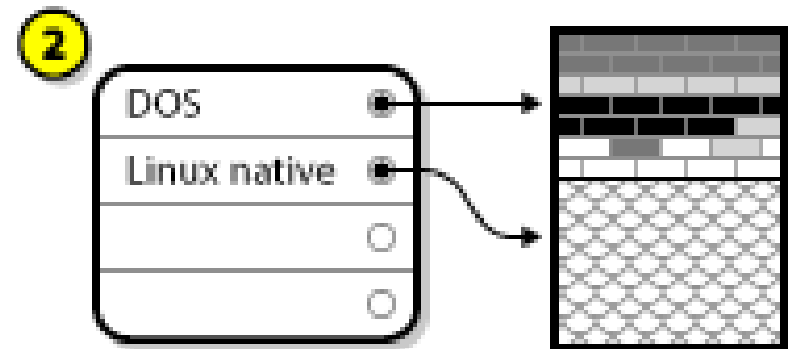
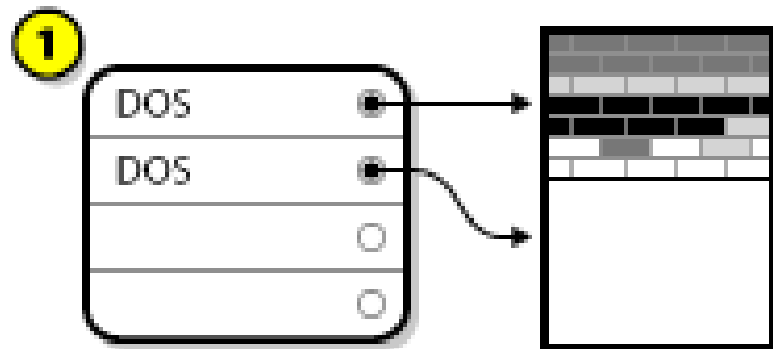
파티션 압축, 크기 조정

16



새 파티션 생성

17



파티션 이름

18

- 파티션 이름 : /dev/xyN
- xx : 디바이스 종류, 디스크는 sd
- y : 디바이스,
 - /dev/sda : 첫번째 디스크 드라이브
 - /dev/adb : 두번째 디스크 드라이브
 - ...
- N : 파티션 번호
 - primary 또는 extended : 1~4
 - logical : 5 부터 매겨짐
- /dev/sda3 : 1번 디스크의 3번째 주(확장)파티션
- /dev/sdb6 : 2번 디스크의 2번째 논리파티션

마운트 포인트 (*mount point*)

19

- 파티션을 디렉터리에 연관시키는 작업
- 마운트 포인트 : 파티션을 접근하는 디렉터리
- 파티션 `/dev/sda1` : 마운트 포인트 `/boot`

```
eyha@network:~$ lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	149.1G	0	disk	
├─sda1	8:1	0	243M	0	part	/boot
├─sda2	8:2	0	1K	0	part	
└─sda5	8:5	0	148.8G	0	part	
├─localhost--vg-root (dm-0)	252:0	0	147.3G	0	lvm	/
└─localhost--vg-swap_1 (dm-1)	252:1	0	1.5G	0	lvm	[SWAP]
sr0	11:0	1	1024M	0	rom	

권고 파티션 크기

20

- **/boot : 500 MB, 시스템 커널**
- **/ : 최소 10 GB, 보통 20 GB**
- **/home : 최소 10 GB, 보통 50 GB 이상**
- **swap : (보통, 하이버네이션 고려)**
 - ▣ RAM : 2GB 이하는 메모리 (2배, 3배)
 - ▣ RAM : 2~8 GB는 메모리 (1배, 2배)
 - ▣ RAM : 8~64 GB는 메모리 (0.5배, 1.5배)
 - ▣ RAM : 64~ GB는 메모리 (워크로드에 따라 다름, 고려 안함)

파티션 생성 관리 실습

안양대학교 컴퓨터공학과 하은용

디스크 파티션 관리

현재 파티션 상태 (fdisk)

22

```
$ sudo fdisk -l /dev/sda
```

```
Disk /dev/sda: 57.9 GiB, 62176362496 bytes, 121438208 sectors
```

```
Units: sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disklabel type: dos
```

```
Disk identifier: 0x00000000
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1		21952	121438207	121416256	57.9G	c	W95 FAT32 (LBA)

파티션 생성 실습 (*fdisk /dev/sda*)

23

```
$ su  
암호:  
# fdisk /dev/sda
```

```
Welcome to fdisk (util-linux 2.29.2).  
Changes will remain in memory only, until you decide to write  
them.  
Be careful before using the write command.
```

```
Command (m for help): m
```

파티션 생성 실습 (*fdisk /dev/sda*)

24

□ fdisk 내부 명령 'w' 실행 후 메시지

Command (m for help): **w**

The partition table has been altered.

Calling ioctl() to re-read partition table.

Re-reading the partition table failed.: 장치나 자원이 동작 중

The kernel still uses the old table. The new table will be used at the next **reboot** or after you run **partprobe(8)** or **kpartx(8)**.

파티션 생성 실습 (*fdisk /dev/sda*)

25

```
# fdisk -l /dev/sda
```

```
Disk /dev/sda: 57.9 GiB, 62176362496 bytes, 121438208 sectors
```

```
Units: sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disklabel type: dos
```

```
Disk identifier: 0x5564a0c9
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1		2048	33556479	33554432	16G	83	Linux
/dev/sda2		33556480	67110911	33554432	16G	83	Linux
/dev/sda3		67110912	100665343	33554432	16G	83	Linux
/dev/sda4		100665344	121438207	20772864	9.9G	5	Extended
/dev/sda5		100667392	109055999	8388608	4G	83	Linux
/dev/sda6		109058048	121438207	12380160	5.9G	83	Linux

파티션 상태 (df -h)

26

□ 디스크 파티션 마운트 상태 확인

```
# df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/root	29G	3.6G	25G	13%	/
devtmpfs	460M	0	460M	0%	/dev
tmpfs	464M	0	464M	0%	/dev/shm
tmpfs	464M	13M	452M	3%	/run
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	464M	0	464M	0%	/sys/fs/cgroup
/dev/mmcblk0p1	44M	22M	22M	51%	/boot
tmpfs	93M	0	93M	0%	/run/user/1000
/dev/sda1	58G	32K	58G	1%	/media/pi/3467-D863

파티션 테이블 검사 (*partprobe*)

27

□ 디스크 파티션 마운트 상태 확인

```
# partprobe
```

```
Error: Partition(s) 1, 2, 3, 4, 5, 6 on /dev/sda have been  
written, but we have been unable to inform the kernel of the  
change, probably because it/they are in use. As a result,  
the old partition(s) will remain in use. You should reboot  
now before making further changes.
```

```
# df -h
```

```
/dev/sda1          58G   32K   58G   1% /media/pi/3467-D863
```

기존 파티션 언마운트 (*umount*)

28

□ 파티션 언마운트 후 상태

```
# umount /dev/sda1  
# partprobe  
# df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/root	29G	3.6G	25G	13%	/
devtmpfs	460M	0	460M	0%	/dev
tmpfs	464M	0	464M	0%	/dev/shm
tmpfs	464M	13M	452M	3%	/run
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	464M	0	464M	0%	/sys/fs/cgroup
/dev/mmcblk0p1	44M	22M	22M	51%	/boot
tmpfs	93M	0	93M	0%	/run/user/1000

새 파티션 마운트 시도(*mount*)

29

□ 파티션 마운트 시도 에러 상태

```
# mkdir myPart1
```

```
# mount /dev/sda1 myPart1
```

```
mount: wrong fs type, bad option, bad superblock on /dev/sda1,  
       missing codepage or helper program, or other error
```

In some cases useful info is found in syslog - try
dmesg | tail or so.

파티션에 파일시스템 생성 (mkfs)

30

□ 파일 시스템 만들기

```
# mkfs -t ext4 /dev/sda1
mke2fs 1.43.4 (31-Jan-2017)
Creating filesystem with 4194304 4k blocks and 1048576 inodes
Filesystem UUID: 6c78e0b1-ab0e-4793-b27a-d09299a5ce3c
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736,
    1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information:
done
```

파일시스템 생성 후 마운트(mount)

31

□ 다시 파티션 마운트

```
# mount /dev/sda1 myPart1
# mount -t ext4
/dev/mmcblk0p2 on / type ext4 (rw,noatime,data=ordered)
/dev/sda1 on /home/pi/myPart1 type ext4
(rw,relatime,data=ordered)
# df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/root	29G	3.6G	25G	13%	/
.....					
/dev/mmcblk0p1	44M	22M	22M	51%	/boot
tmpfs	93M	0	93M	0%	/run/user/1000
/dev/sda1	16G	45M	15G	1%	/home/pi/myPart1

마운트된 파티션 사용

32

□ 일반적인 작업

```
# cd myPart1
```

```
# ls -l
```

```
합계 16
```

```
drwx----- 2 root root 16384 11월 25 17:14 lost+found
```

```
# mkdir work shell mp4
```

```
# ls -l
```

```
합계 28
```

```
drwx----- 2 root root 16384 11월 25 17:14 lost+found
```

```
drwxr-xr-x 2 root root 4096 11월 25 17:24 mp4
```

```
drwxr-xr-x 2 root root 4096 11월 25 17:24 shell
```

```
drwxr-xr-x 2 root root 4096 11월 25 17:24 work
```

```
#
```


사용 완료 후 다시 언마운트(umount)

33

□ 파티션 사용 완료 후 해제

```
# umount /dev/sda1
```

```
# ls -l myPart1
```

```
합계 0
```

```
# df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/root	29G	3.6G	25G	13%	/
devtmpfs	460M	0	460M	0%	/dev
tmpfs	464M	0	464M	0%	/dev/shm
tmpfs	464M	13M	452M	3%	/run
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
tmpfs	464M	0	464M	0%	/sys/fs/cgroup
/dev/mmcblk0p1	44M	22M	22M	51%	/boot
tmpfs	93M	0	93M	0%	/run/user/1000

요약

34

- 디스크 파티션 테이블 : MBR, GPT
- 파티션 생성 : fdisk, df
- 파티션 사용 : mkfs, mount
- 파티션 해제 : umount