

# Luento 9

## Järjestelmän ulkoinen muisti

Muistihierarkia  
 Kiintolevyt  
 Muut pyörivät levyt  
 I/O:n toteutus

19/11/2001

Teemu Kerola, Copyright 2001

1

## Muistihierarkia <sup>(4)</sup>

ks. Fig. 4-1 [Stal99]

- Ulkoinen muisti on halvempaa toteuttaa per tavu
- Ulkoinen muisti on hyvin paljon hitaampaa kuin sisäinen muisti
- Aika/tila optimointi
  - suuret tietomäärät täytyy (kannattaa) kustannussyistä pitää ulkoisessa muistissa
  - pienet tietomäärät täytyy (kannattaa) tehokkuussyistä pitää sisäisessä muistissa
- Kaiken viitatus tiedot tulee suoritusajana olla sisäisessä muistissa!



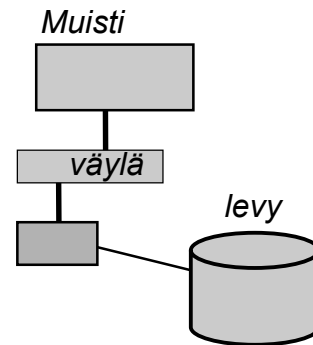
19/11/2001

Teemu Kerola, Copyright 2001

2

## Virtuaalimuisti (3)

- Osa muistihierarkiaa
- Vastaus ongelmaan
  - miten tehdä suoritusajaisesta muistista yhtä suuri kuin levymuisti ja yhtä nopea kuin keskusmuisti?
- Kaksitasoinen:
  - keskusmuistissa kulloinkin käytössä olevat alueet
  - levyllä kaikki tiedot
  - kopiointi tarvittaessa



19/11/2001

Teemu Kerola, Copyright 2001

3

## Virtuaalimuistin toteutus (4)

- Toteutustavat
  - kanta- ja rajarekisterit
  - sivutus
  - (segmentointi ja sivuttava segmentointi)
- Pääosa toteutuksesta ohjelmistotasolla
- Laitteistotuki
  - MMU – muistinhallintayksikkö
  - nopeuttaa viitatus muistipaikan todellisen osoitteen laskentaa
  - rakenne ja toiminta vaihtelee virtuaalimuistin toteutustavan mukaan

Lisää tietoa?



Tietokoneen rakenne

Käyttöjärj.- I ja II

19/11/2001

Teemu Kerola, Copyright 2001

4

19/11/2001

Teemu Kerola, Copyright 2001

5

## Tiedostojärjestelmä <sup>(5)</sup>

- KJ:n osa, hallitsee kaikkia tiedostoja
- Valvoo oikeuksia tiedostoa avattaessa
- Muuntaa tiedostonimet fyysisiksi osoitteiksi
- Ylläpitää taulukoita, joista näkee mitä kohtaa mistäkin tiedostosta kukin prosessi on käsittelemässä
- Tiedostojärjestelmä lukee ja kirjoittaa tiedostoja suurina kerralla käsiteltävinä lohkoina (0.5-8 KB?)
  - käyttäjätason prosessit käsittelevät tiedostoja tavuittain eikä niiden tarvitse tietää tiedoston todellista fyysistä rakennetta (laiteajuri huolehtii siitä)

19/11/2001

Teemu Kerola, Copyright 2001

6

## Levymuisti <sup>(9)</sup> ks. Fig 5.3 [Stal99]

- Levykkö 150 MB-181GB/levykkö
  - pyörii nopeasti (koko ajan?) ~ 3600-10800 rpm
  - luku/kirjoituspäät liikkuvat kaikki yhtä aikaa?
  - monta levyä ~ 1-16 levyä/levykkö
- Levypinta ks. Figs 5.3, 5.4 [Stal99]
  - 2 per levy (tai 1) ~ 2000-3000 uraa/pinta
  - ura ~ 20-100 sektoria/ura
  - sektori: pienin kerralla osoitettavissa oleva alue ~ 0.5-8KB/sektori
  - sylinteri: päällekkäin olevat urat, luku/kirj. pää samalla kohtaa ~ 1-32 uraa/sylinteri

19/11/2001 Teemu Kerola, Copyright 2001 7

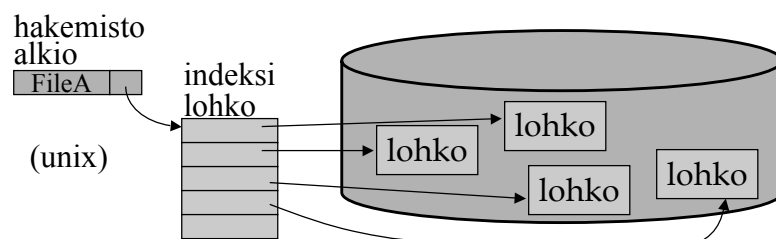
## Levymuistin saantiaika <sup>(5)</sup>

- Tiedon osoite: levypinta + ura + sektori
  - laiteajuri etsii KJ-taulukoista loogisen osoitteen perusteella ks. Fig 5.4 [Stal99]
- Saantiaika:
  - hakumarren siirtoaika (seek time)  
Esim: aver 6.3 ms, min-max 2-15 ms?
  - odota kunnes sektori kohdalla (rotational delay)  
Esim: pyörähdysviive kun 3600 rpm: 8.33 ms  
(keskim. puolen kierroksen aika)
  - siirrä sektorin verran tietoa (data transfer time)  
Esim: pyör.aika / sekt. lkm = 0.42 ms

19/11/2001 Teemu Kerola, Copyright 2001 8

## Tiedoston talletus levyille (2)

- Tiedosto koostuu useista lohkoista
  - lohko per sektori (lohko per usea sektori?)
- Levyn hakemistossa on tieto kunkin tiedoston käyttämistä lohkoista
  - luetaan lohkot annetussa järjestyksessä



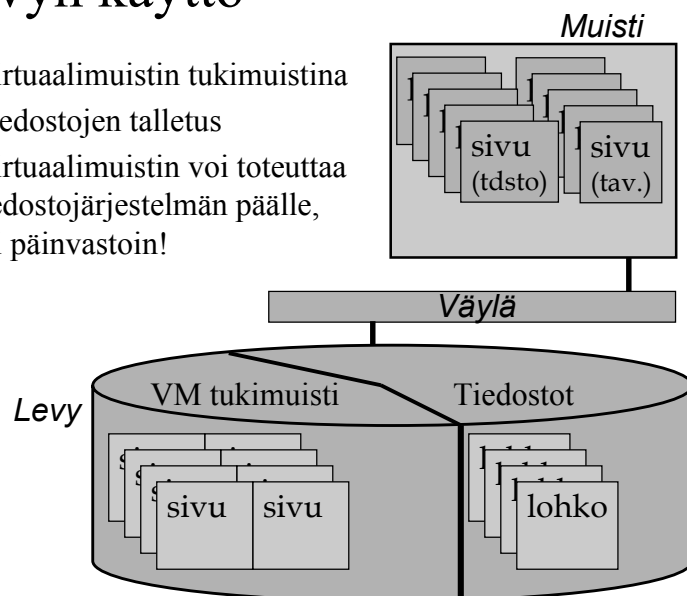
19/11/2001

Teemu Kerola, Copyright 2001

9

## Levyn käyttö

- Virtuaalimuistin tukimuistina
- Tiedostojen talletus
- Virtuaalimuistin voi toteuttaa tiedostojärjestelmän päälle, tai päinvastoin!



19/11/2001

Teemu Kerola, Copyright 2001

10

## DOS-levykkeen rakenne <sup>(10)</sup>

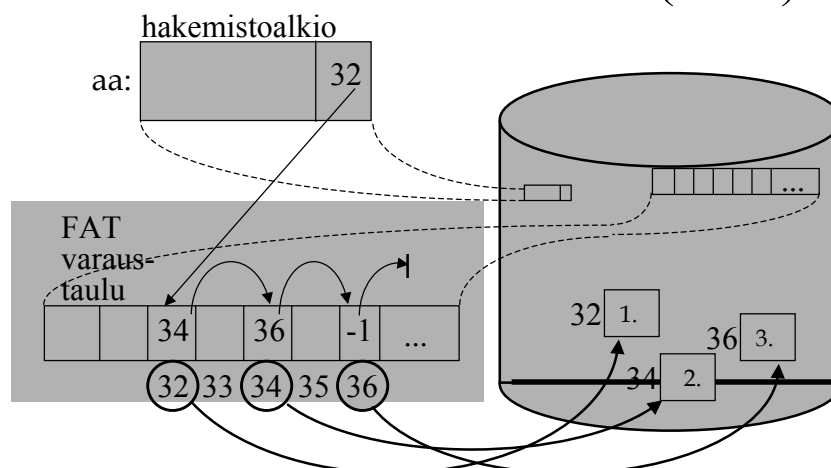
- FAT – File Allocation Table
  - kertoo, mitkä sektorit ovat vapaana
  - kertoo, mitkä sektorit ovat käytössä millekin tiedostolle
  - kiinteä paikka levykkeellä, 2 kopiota
- Hakemisto
  - erikoistyyppinen tiedosto
  - sisältää hakemistoalkion joka tiedostolle
    - nimi, tyyppi, koko, muutos pvm ja kellonaika
    - attribuutit (invisible, read-only, ...)
    - linkki ensimmäiseen sektoriin FAT:ssä

19/11/2001

Teemu Kerola, Copyright 2001

11

## FAT - File Allocation Table (DOS)



DOS levykkeet: 1.44MB, lohko 512 B, 2.9K lohkoa  
 entä: 1 GB, lohko 64 KB, 64K lohkoa OK?

19/11/2001

Teemu Kerola, Copyright 2001

12

## Mikä on hyvä levylohkon koko?

- Mihin käyttöön?
- Videokuvan talletus/playback?

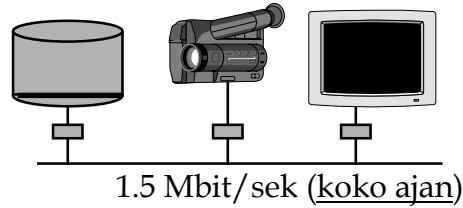
- 1 minuutti
- 12.3 MB

- lohko 4KB?  
osoitetaulu?

- epäsuorat viittaukset?

- riittääkö levyn/väylän/ohjaimien nopeus?

- entä jos 1 tunti? riittääkö levyn kapasiteetti?



19/11/2001

Teemu Kerola, Copyright 2001

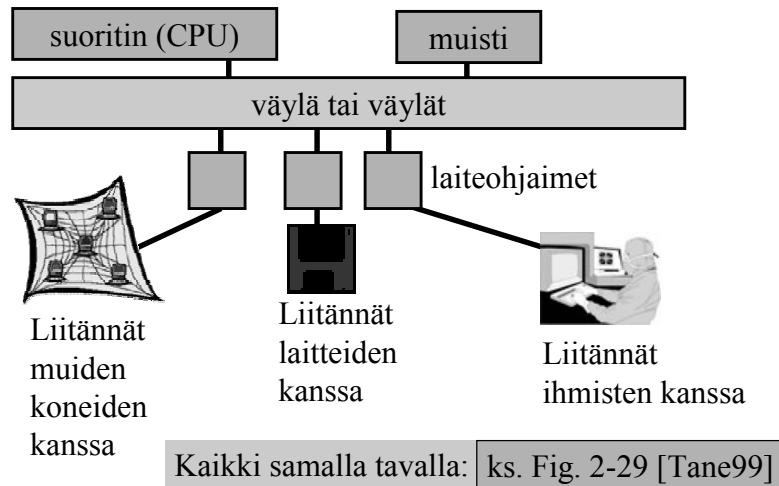
13

19/11/2001

Teemu Kerola, Copyright 2001

14

## Laitteiden liittäminen järjestelmään (4)

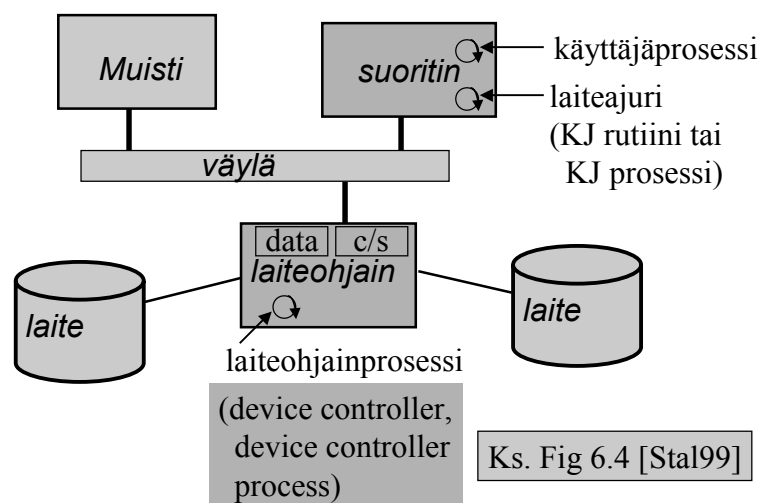


19/11/2001

Teemu Kerola, Copyright 2001

15

## Laiteohjain (I/O Moduuli)



19/11/2001

Teemu Kerola, Copyright 2001

16



## Laitteiden käytön toteutus <sup>(5)</sup>

ks. laiteohjainkuva (ed. kalvo)

- Käyttäjäohjelma kutsuu käyttöjärjestelmän laiteajuria tekemään I/O:n. Laiteajuri suoritetaan samalla suorittimella kuin käyttöohjelmakin.
- Laiteajuri ohjaa laitteen toimintaa laitteen laiteohjaimella olevien kontrollirekisterien (muistialue) avulla
- Laiteajuri voi lukea laitteen tilatietoa laiteohjaimella olevien statusrekisterien (muistialue) avulla
- Laiteajuri voi lukea (kirjoittaa) laitteen lukemaa (laitteelle kirjoitettavaa) tietoa laiteohjaimella olevien datarekistereiden (muistialue) avulla
- Kontrolli-, status- ja datarekisteri kolmikko muodostaa ”I/O portin” suorittimen näkökulmasta

19/11/2001

Teemu Kerola, Copyright 2001

17

## Laiteohjaimen rekistereihin viittaaminen <sup>(5)</sup>

- Ongelma: miten suorittimella ks. laiteohjainkuva suorittavan laiteajuri viittaa eri kortilla oleviin rekistereihin?
- Ratkaisu 1: omat I/O-konekäskyt tätä tarkoitusta varten
  - käskyssä annetaan laiteohjaimen identifikaatio ja rekisterin nro (I/O osoiteavaruus)
  - vaikea laajentaa käyttöä uusiin laitteisiin, joilla ”laiterekisterit” voivat olla hyvinkin erilaisia

x86: IN, OUT  
INS, OUTS

KOKSI:  
IN, OUT

ks. Minix esimerkin  
port\_out [Tane87]

19/11/2001

Teemu Kerola, Copyright 2001

18

## Ratkaisu 2: muistiinkuvattu I/O <sup>(5)</sup>

ks. laiteohjainkuva

- Laiteajuri lukee/kirjoittaa laiteohjaimella olevia rekistereitä (data, status/kontrolli) tavallisilla muistin luku/kirjoitus käskyillä
  - ei tarvita erillisiä I/O-konekäskyjä! `load R1,=DiskRd`  
`store R2, DiskCtr`
  - laiteohjaimella olevat ”laiterekisterit” ovat samanlaista viitattavaa muistia kuin ”normaali muisti”
  - muistisoitteen ensimmäiset bitit valitsevat, mille laitteelle (vai tavallisen muistiin) viittaus kohdistuu
  - voidaan käyttää rinnan I/O käskyjen kanssa (laiterekistereihin voi siis viitata sekä I/O-käskyillä että muistiinkuvatun I/O:n avulla)

esim. Intelin arkkitehtuurit

19/11/2001

Teemu Kerola, Copyright 2001

19

## I/O tyypit <sup>(2)</sup>

ks. laiteohjainkuva

- Suora I/O: laiteajuri odottaa tiukassa silmukassa, kunnes laiteohjaimen statusrekisteri ilmoittaa I/O-pyyntön valmistuneen (direct I/O)
  - laiteajuri siirtää tietoa muistin ja datarekisterin välillä
- Epäsuora I/O: I/O:n odotusaikana suorittimella suoritetaan jotain muuta ohjelmaa (indirect I/O interrupt driven I/O)
  - Kun I/O-pyyntö valmistuu, laiteohjain antaa keskeytyksen (laitekeskeytys, I/O interrupt) suorittimelle, joka (jonkin ajan kuluttua) jatkaa kesken jäänyttä I/O-pyyntön esittänyttä ohjelmaa.
  - laiteajuri siirtää tietoa muistin ja datarekisterin välillä

19/11/2001

Teemu Kerola, Copyright 2001

20

## I/O tyypit (jatkoa) <sup>(4)</sup>

ks. laiteohjainkuva

- DMA - Direct Memory Access
  - älykkäämpi laiteohjain
  - laiteohjain voi suoraan kopioida tiedot keskusmuistiin (laiteajurin ei tarvitse siirtää tietoa muistin ja datarekisterin välillä)
  - laiteohjain tekee paljon suuremman määrän työtä itsenäisesti (kuin epäsuorassa I/O:ssa) ennen suorittimelle annettavaa laitekeskeytystä

19/11/2001

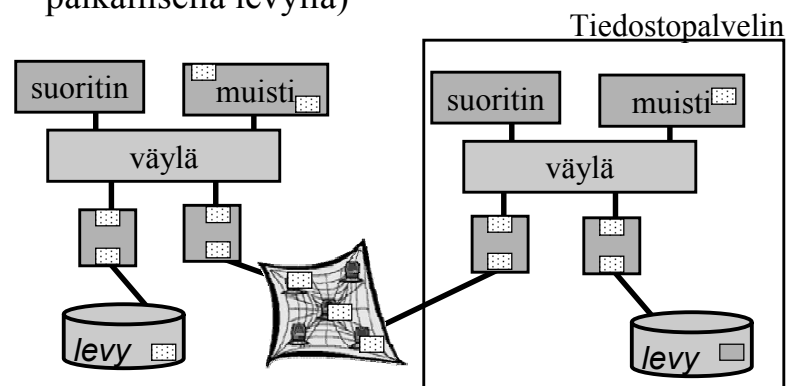
Teemu Kerola, Copyright 2001

21

## Tiedostopalvelin

- (Lähi)verkossa oleva palvelin
- Käytettäessä tiedoston (osien) kopio on muistissa (ja ehkä myös paikallisella levyllä)

□ orig. tiedsto  
 ▤ tiedston kopio?



19/11/2001

Teemu Kerola, Copyright 2001

22

## Tiedostopalvelin <sup>(4)</sup>

- Käytetään kuten paikallista levyä oman systeemin KJ:n liitospalikan avulla
- Paljon hitaampi kuin paikallinen levy
- Tiedostovälimuistit (file cache)  
(muistipuskurit tai levypuskurit)  
nopeuttavat toimintaa käytännössä
  - omassa järjestelmässä
  - palvelimella
  - 50% oman järjestelmän keskusmuistista voi olla varattu tiedostovälimuistille

Lisää  
tietoa?



KJ kurssit

19/11/2001

Teemu Kerola, Copyright 2001

23

19/11/2001

Teemu Kerola, Copyright 2001

24

## Esimerkki: kirjoittimen laiteajuri ttk-91 koneelle <sup>(6)</sup>

- Laitteella voi tulostaa kokonaislukuja yksi kerrallaan
- Muistiinkuvattu I/O, suora I/O
- Laiteportti
  - kontrollirekisteri                   muistipaikka 1048576 = 0x80000
  - tilarekisteri                         muistipaikka 1048577 = 0x80001
  - datarekisteri                         muistipaikka 1048578 = 0x80002
- Laiteajuri toimii etuoikeutetussa tilassa
- Kutsu:
 

```
PUSH SP, =0 ; space for return value
PUSH SP, X ; parameter to print
SVC SP, =Print ; returns Success/Failure
POP SP, R1
JNZER R1, TakeCareOfTrouble
```

19/11/2001

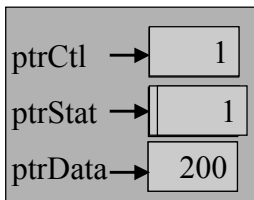
Teemu Kerola, Copyright 2001

25

## Esim: laiteajurin toteutus <sup>(12)</sup>

```
ptrCtr DC 1048576 ; control register address
ptrStat DC 1048577 ; status register address
ptrData DC 1048578 ; Print
retVal EQU -3
parData EQU -2
```

Oleta: SVC:n ja IRET:n  
toteutus samalla tavalla  
kuin CALL ja EXIT



See: driver.k91

```
Solution with no timeout
PUSHR SP ;save regs
LOAD R1, parData(FP)
STORE R1, @ptrData ; data to print
LOAD R1, =0
STORE R1, @ptrStat ; init state register
LOAD R1, =1
STORE R1, @ptrCtr ; give command to print

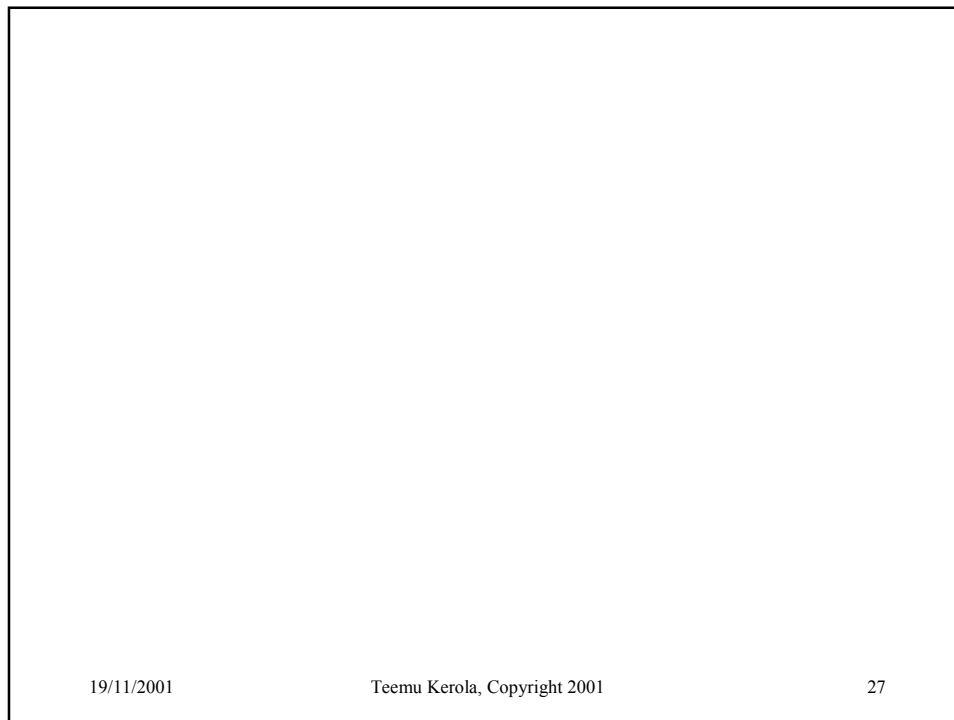
Wait LOAD R1, @PtrStat ; check state register
JNZER R1, Done
JUMP Wait ; wait until I/O done

Done LOAD R1, =0 ; return "Success"
STORE R1, retVal(FP)
POPR SP ; recover regs
IRET SP, =1
```

19/11/2001

Teemu Kerola, Copyright 2001

26



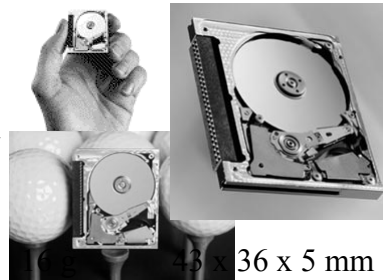
## Erilaisia levyjä

- Kiintolevy
- ZIP levyke
- Levyke
- CD-ROM, CD-R, CD-RW
- DVD
- ...?...

19/11/2001 Teemu Kerola, Copyright 2001 28

## Kiintolevy

- Kiinteä, ei vaihdettavissa oleva levy
- Tila: 0.150-181 GB (v. 2000)
- Haku aika: 5-15 ms
- 1-10 levyä
- Pyörimisnopeus: 4500-10800 rpm
- Siirtonopeus: 5-50 MB/sec



19/11/2001

Teemu Kerola, Copyright 2001

29

## Zip & Jaz levykeasemat

- Vaihdettava levyke
- Tila: 0.1-2 GB
- Haku aika: 10-30 ms
- Pyörimisnopeus: 3000-5400 rpm
- Siirtonopeus: 1-6 MB/sec



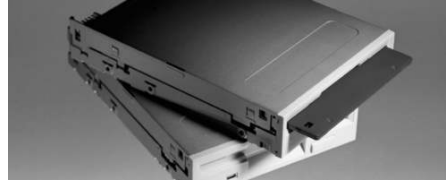
19/11/2001

Teemu Kerola, Copyright 2001

30

## Levykeasema

- Vaihdettava levyke
- Tila: 1.44 MB
- Haku aika 90 ms
- Pyörimisnopeus 300 rpm
- Siirtonopeus 0.05 MB/sec



19/11/2001

Teemu Kerola, Copyright 2001

31

## CD - Compact Disc

- Vaihdettava levyke
- CD-R (Recordable)
- CD-RW (Rewritable)
- Yksi pitkä spiraalimainen ”ura”
- Tila: 650 MB
- Haku aika 90 ms
- Pyörimisnopeus 200-9000 rpm
- Siirtonopeus 0.1-2 MB/sec



19/11/2001

Teemu Kerola, Copyright 2001

32



## DVD - Digital Versatile Disk

- Vaihdeettava levyke
- DVD-ROM
- DVD-R (Recordable)
- DVD-RAM (kuten tavallinen kovalevy)
- Tila: 4.7-17 GB
- Haku aika 100-180 ms
- Pyörimisnopeus 2000-8000 rpm
- Siirtonopeus 2-8 MB/sec



19/11/2001

Teemu Kerola, Copyright 2001

33

