

Käyttöjärjestelmät I

Luento 7: VIRTUAALIMUISTI

Stallings, Luku 8.1

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 1

Sisältö

- Ohjelman suoritus virtuaalimuistissa
- Sivutus
- Osoitemuunnospuskuri TLB
- Lisää sivutauluista

Luento 8:

- Segmentointi
- Segmentointi ja sivutus yhdistettynä
- Yhteiskäytöstä

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 2

Käyttöjärjestelmät I

OHJELMAN SUORITUS VIRTUAALIMUISTISSA

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 3

Suoritus virtuaalimuistissa

- **Ohjelman loogiset osoitteet muutetaan fyysisiksi osoitteiksi vasta ajonaikana**
 - ◆ prosessin paikka muistissa vaihtelee, sillä ei vaikutusta osoitemuunnokseen
 - ◆ MMU
- **KJ käsittelee ohjelmaa sivuina tai kääntäjä jakaa ohjelman segmentteihin, jotka KJ voi sijoitella vapaasti muistiin**
 - ◆ KJ:n kirjanpito osien sijainnista prosessin sivutaulussa tai segmenttitaulussa

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 4

Suoritus virtuaalimuistissa

- **Kaikkien sivujen / segmenttien ei tarvitse olla muistissa yhtäaikaan**
 - ◆ riittää, että suoritettava osa ja sen data muistissa
 - ◆ paikallisuus, suoritus viihtyy samoilla alueilla
 - ◆ ks. kuva 8.1
- **Laitteisto (MMU) ja KJ huolehtivat, että tarvittavat osat oikeaan aikaan muistissa**
 - ◆ MMU huomaa puuttumisen
 - ◆ KJ noutaa muistiin

= **Virtuaalimuisti**

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 5

Suoritus virtuaalimuistissa

- **Sivu- / segmenttitaulun alkiossa läsnäolobitti, josta käy ilmi onko sivu / segmentti muistissa**
- **Lataaja tuo aluksi muistiin vain muutaman sivun / segmentin (ennaltanouto) tai ei yhtään (tarvenouto)**
- **Prosessi CPU:lle ⇔**
 - MMU:hun sivu- / segmenttitaulun fyysinen muistiosoite**
 - TLB:n sisällön mitätöinti**

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 6

Puutoskeskeytys (memory / page fault)

- **Jos viitattu osoite ei ole muistissa, MMU aiheuttaa keskeytyksen**
- **KJ siirtää keskeytyksen aiheuttaneen prosessin (A) Blocked-tilaan**
- **KJ etsii sivulle / segmentille vapaan paikan muistista**
- **KJ käynnistää ohjaimen siirtämään puuttuvaa sivua / segmenttia ko. paikkaan**
- **Siirron aikana CPU suorittaa muita prosesseja**

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 7

Puutoskeskeytys

- **Kun siirto valmis, ohjain keskeyttää suorituksessa olevan prosessin B**
- **KJ päivittää prosessin A sivu/segmenttitaulun ja siirtää prosessin A Ready-tilaan**
- **Suoritus palaa takaisin prosessiin B**
- **Kun prosessi A taas aikanaan suoritukseen, se viittaa uudestaan äskeiseen osoitteeseen**
 - ◆ nyt viitatus mp:n sisältö muistissa

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 8

Virtuaalimuistin etuja

- **Muistia käytetään tehokkaammin hyväksi**
 - ◆ kustakin prosessista vain tarvittava osa muistissa
 - ◆ montako sivua/prosessi pidetään muistissa yhtäaikaan?
 - ☞ lokaalit ja globaalit algoritmit
- **Prosessoria käytetään (kenties) tehokkaammin**
 - ◆ moniajoastetta voi nostaa
 - ☞ muistiin mahtuu paremmin, ei ruuhkautumista

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 9

Virtuaalimuistin etuja

- **Voi ajaa fyysistä muistia suurempia ohjelmia**
 - ◆ ohjelmoijan ei tarvitse huolehtia kerrostuksesta
- **Osoiteavaruus voi olla valtaisa verrattuna todelliseen muistin määrään**
 - ◆ esim. 32 bittiä => 4GB:n osoiteavaruus
 - ◆ hyöty?
- **Looginen osoiteavaruus saa sisältää 'reikiäkin'**
 - ◆ vain tarvittavat osat kuvataan fyysiseen muistiin

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 10

Käyttöjärjestelmät I

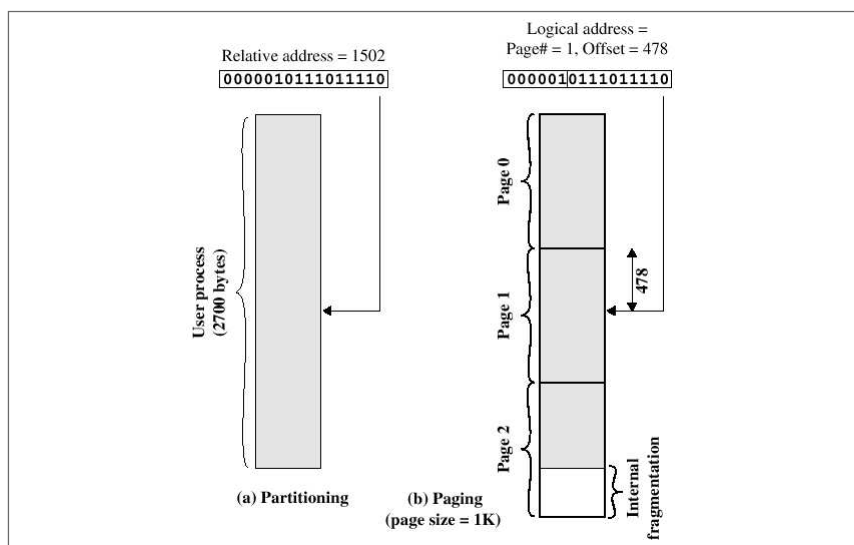
SIVUTUS

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 11

Sivutus

Kuva 7.11

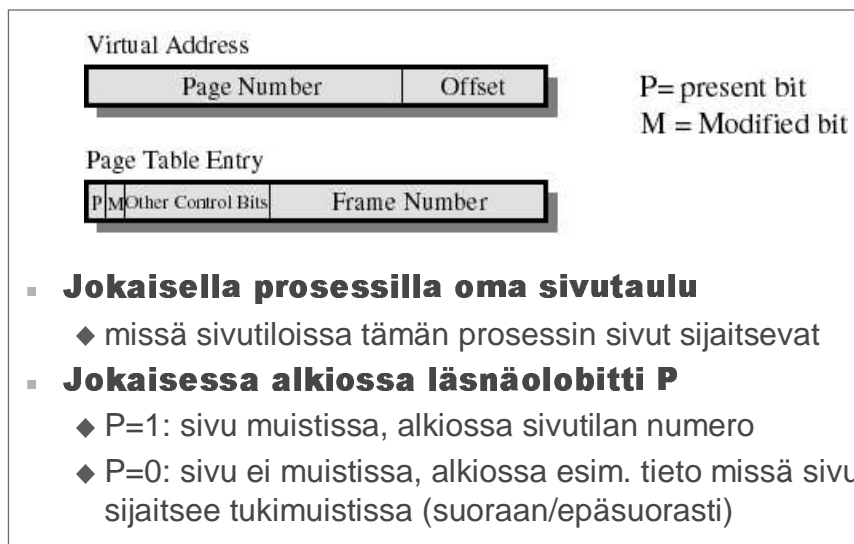


KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 12

Sivutaulu

Kuva 8.2a



KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 13

Sivutaulu

- **Jokaisessa sivutaulun alkiossa muutettu-bitti M (modified)**
 - ◆ M=1: sivun sisältö muuttunut muistissa, sivu kirjoitettava levyille, jos varaus vapautetaan
 - ◆ M=0: sivua ei muutettu, ei tarvitse kirjoittaa levyille sivutilaa vapautettaessa
- **Sivutaulun alkiossa mahd. myös muuta tietoa**
 - ◆ käyttötapabitti: R / RW
 - ◆ suojaustasobitti/bitit: KJ:n sivu / tav. prosessin sivu
 - ◆ milloin sivuun viitattu viimeeksi tai viitelaskuri
 - ☞ poistoalgoritmit tarvitsevat näitä

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 14

Osoitemuunnos

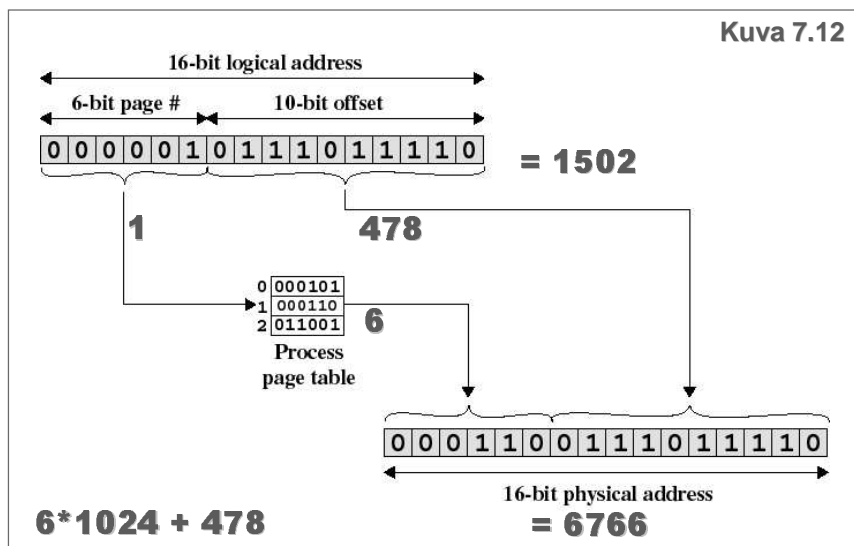
- **Prosessien sivutaulut tavallisesti eri kokoisia ja voivat olla suuria**
 - ◆ koko riippuu sivukoosta ja ohjelman koosta
 - ◆ koko taululle ei voi varata tilaa MMU:sta
 - ☞ sivutaulu muistissa ja osa jopa levyllä
 - ☞ sivutaulun fyysinen osoite PCB:ssä
- **MMU:ssa sivutaulurekisteri PTR, jossa suoritettavan prosessin sivutaulun fyysinen alkuosoite**

Osoitemuunnos

- **MMU jakaa loogisen osoitteen pariaksi (sivunro,siirtymä)**
 - ◆ Esim. kun sivukoko 1024 B (= 2^{10})
 - ☞ 10 viimeistä bittiä siirtymä
 - ☞ alkuosa sivunumero
- **MMU korvaa sivunumero-bitit sivutaulusta löytyvillä sivutilannumero-biteillä**
 - ◆ ts. MMU katenoi sivutilanumeron ja siirtymän bitit
- **Helppo laitetoiminto**

Looginen vs Fyysinen osoite

Kuva 7.12

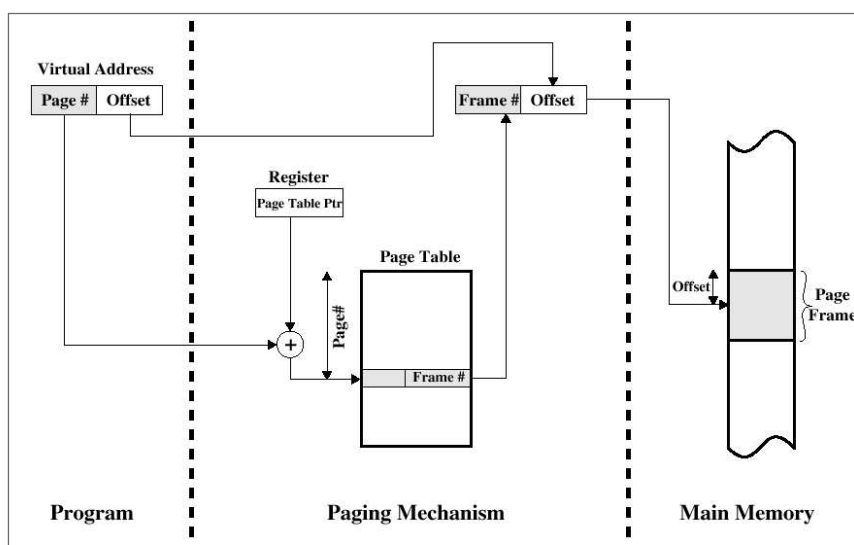


KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 17

Osoitemuunnos

Kuva 8.3



KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 18

Käyttöjärjestelmät I

Osoitemuunnospuskuri TLB

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 19

Osoitemuunnospuskuri TLB

- **Koska sivutaulu muistissa, jokaiseen muunnokseen näyttäisi liittyvän yksi ylimääräinen muistinouto**
 - ◆ hae sivutaulun alkio MMU:hun
 - ◆ tee osoitemuunnos
 - ◆ nouda / talleta ko. fyysiseen osoitteeseen
- **Ratkaisu: osoitemuunnospuskuri TLB**
 - ◆ Translation Lookaside Buffer
- **MMU pitää tallessa edellisissä muunnoksissa tarvittavat sivutaulun tiedot**
 - ◆ paikallisuus: sitä tarvitaan het'kohta uudestaan
 - ◆ sivutaulun alkioita ei tarvitse noutaa joka kerta

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 20

Etsintä TLB:stä

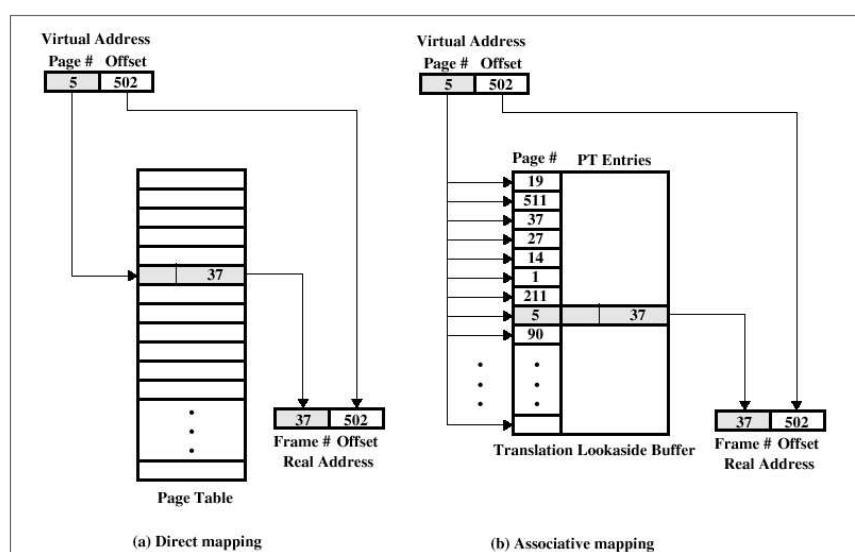
- **TLB nopeiden rekistereiden joukko**
 - ◆ esim. 32 rekisteriä
- **Käyttää assosiatiivista hakua**
 - ◆ vertailu kohdistuu kaikkiin alkioihin yhtäaikaan
 - ◆ etsintä nopeaa
 - ◆ laitetoiminto!
- **Kaikilla prosesseilla sivut 0, 1, 2, ..., mutta eri sivutiloissa**
 - TLB tyhjennettävä prosessin vaihdossa
- **TLB:n alkiossa (ainakin)**
sivunro, sivutaulun alkio, validibitti V

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 21

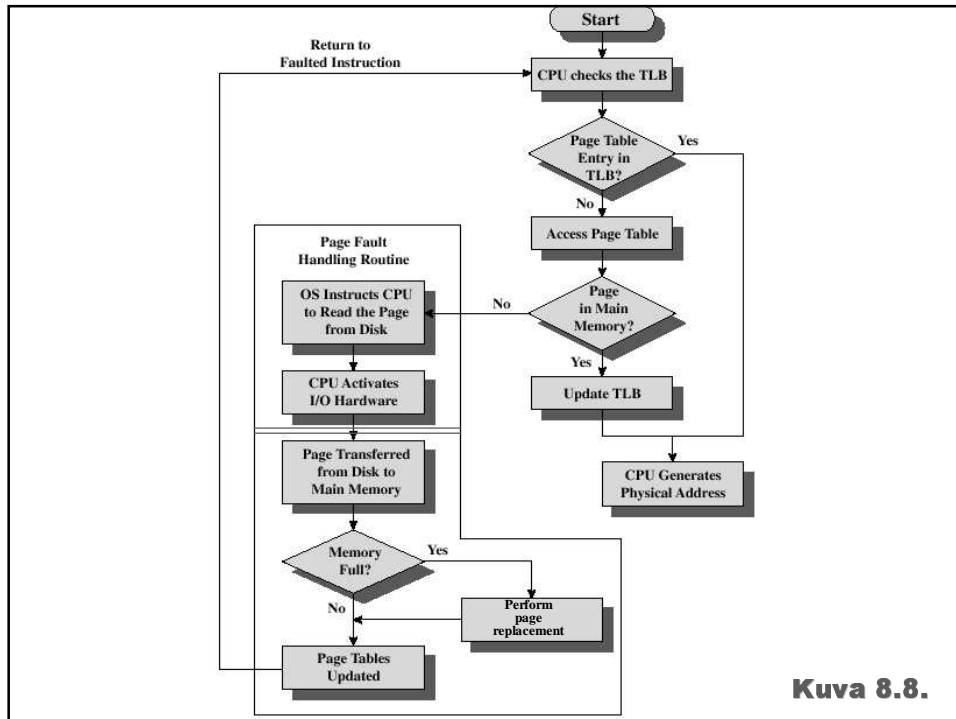
Etsintä TLB:stä

Kuva 8.9



KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 22



TLB ja osoitemuunnos

sivunro = loog. osoitteen alkubitit
 siirtymä = loog. osoitteen loppubitit

Jos sivun tiedot ei TLB:ssä tai V=0,
 nouda TLB:hen sivutaulun alkio
 osoitteesta PTR + sivunro

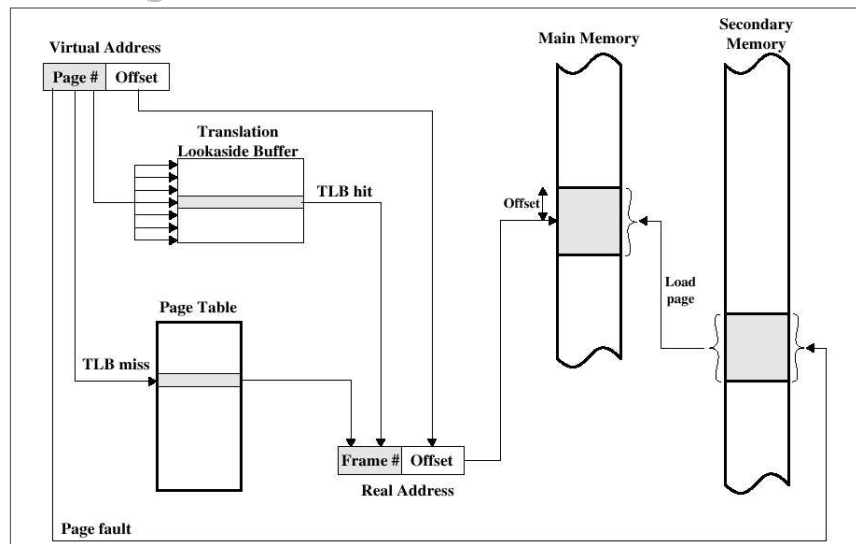
Jos P=0, aiheuta sivunpuutoskeskeytys

Fyys.os = Katenoi(Sivutilanro, Siirtymä)

- **Kun keskeytys käsitelty, sama osoite tulee viitattavaksi uudelleen**
 - ◆ Esim. PC:n kasvatus vasta osoitemuunnoksen jälkeen

TLB ja osoitemuunnos

Kuva 8.7



KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 25

TLB:n alustus

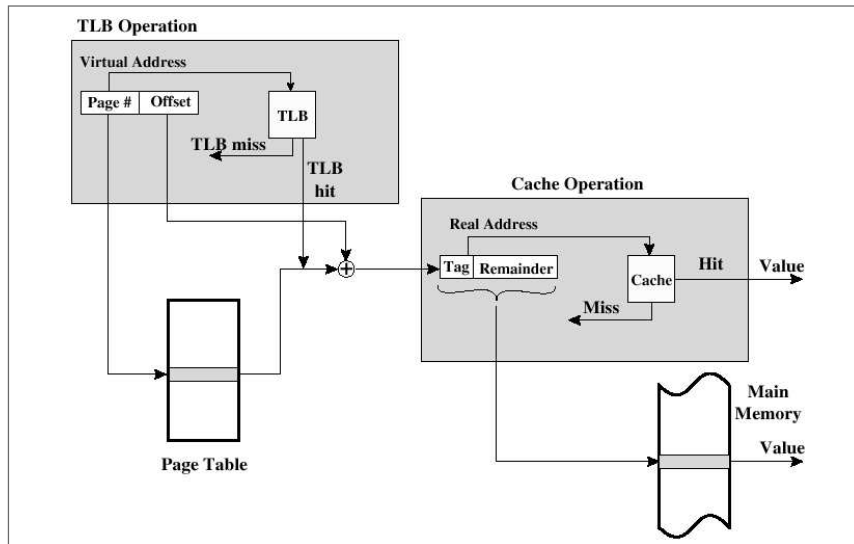
- **Kun suoritettava prosessi vaihtuu, TLB:n vanha sisältö mitätöitävä**
 - ◆ PTR osoittamaan uuden prosessin sivutauluun
 - ◆ nollattava TLB:n alkioiden validibittit $V=0$
- **Koska TLB suhteellisen pieni, tarvitaan sopiva laitetason algoritmi, jonka perusteella valitaan korvattava alkio**
 - ◆ TLB:n alkiossa mahd. myös laitetason viitelaskureita: poista se, jota ei ole aikoihin käytetty

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 26

TLB ja välimuisti

Kuva 8.10



KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 27

Käyttöjärjestelmät I

Lisää sivutauluista

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 28

Lisää sivutauluista

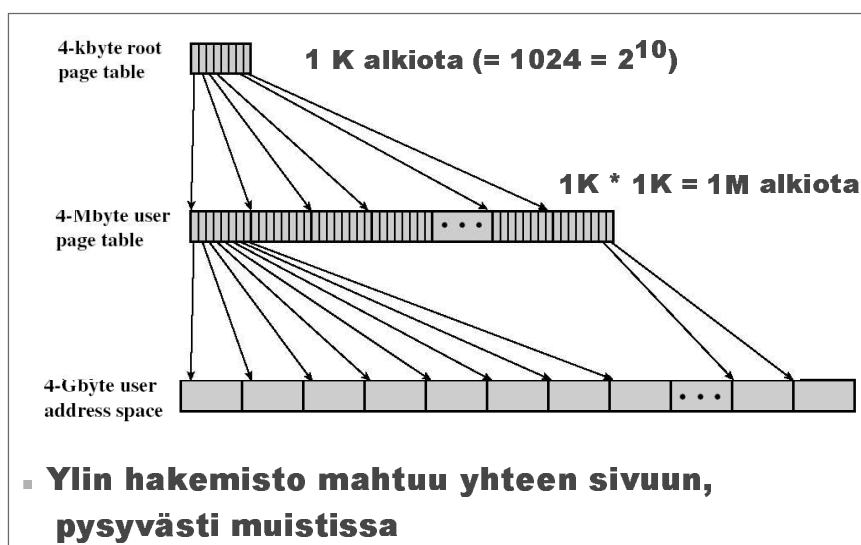
- **Monet järjestelmät sallivat suuren virtuaaliosoiteavaruuden**
 - ◆ looginen osoite esim. 32 tai 64 bittiä
- **Jokaisella prosessilla suuri sivutaulu**
 - ◆ jos 32-bittinen osoite ja sivukoko 4KB (12 bittiä), niin sivuja $2^{20} = 1\text{M}$ kappaletta
 - ◆ jokainen alkio useita tavuja, esim. 4 B, joten sivutaulu 4 MB
- **Myös sivutaulu jaetaan sivuihin ja myös sivutaulun osia voidaan pitää levyllä**
 - ◆ riittää, kun suorituksessa olevaan osaan liittyvät sivut muistissa

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 29

2-tasoinen sivutaulu

Kuva 8.4

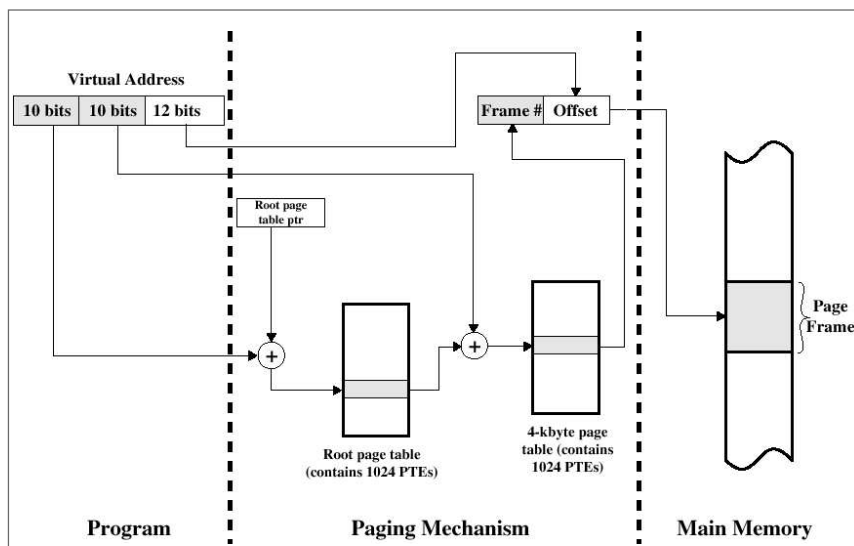


KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 30

2-tasoinen sivutaulu

Kuva 8.5



KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 31

Sopiva sivukoko?

- **Laitteisto (MMU) määrää mitä sivukokoa KJ:n käytettävä**
 - ◆ sivukoko aina 2:sen potenssi
 - ☞ nopea osoitemuunnos
 - ☞ katkaisu ja katenointi helppoa
- **Mitä isommat sivut, sitä sivuja/prosessi**
 - ◆ pienempi sivutaulu vie vähemmän tilaa
 - ◆ ison sivutaulun osia useammin levyllä
 - ☞ enemmän keskeytyksiä

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

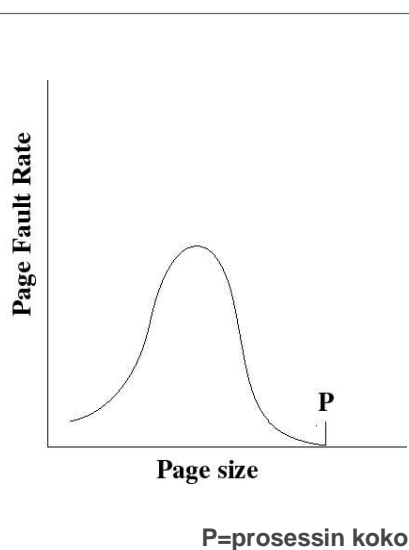
7 - 32

Sopiva sivukoko?

- **Pieni:** aiheuttaa väh. sisäistä pirstoutumista
- **Iso:** sisältää paljon esim. tarpeetonta koodia
- **Iso:** saattaa sopia paremmin yhteen levysierrojen kanssa (lohkokoko)
 - ◆ hakuvarren siirto ja pyörähdysviive syö paljon aikaa
- **Iso:** viittaus useammin samalle sivulle
 - ◆ TLB:n osumatodennäköisyys hyvä

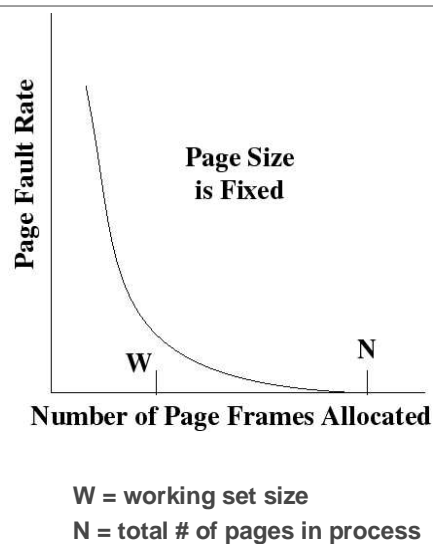
Sivupuutosten määrä

- **Prosessille varatulle alueelle sopii enemmän pieniä sivuja kuin suuria**
- **Pieneltä sivulta viite usein muille sivuille, muistiin valikoituu pian ne sivut joita käytetään paljon**
 - vähän sivupuutoksia
- **Tn. että isolta sivulta viitattu sivu muistissa pienempi**
 - paljon sivupuutoksia
- **Kun sivukoko lähestyy prosessin kokoa,**
 - vähän sivupuutoksia



Sivupuutosten määrä

- **Paljonko sivutiloja per prosessi?**
- **Jos vähän sivutiloja, KJ heittää helposti pois sivun, jota tarvitaan pian uudelleen**
 - paljon sivupuutoksia
- **Jos saa paljon, niin lähes kaikki sivut mahtuvat muistiin**
 - vähän sivupuutoksia
- **lokaalit vs. globaalit algoritmit**



KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 35

Sopiva sivukoko?

- **Yleisimmin käytetty sivukoko 4KB**
 - ◆ ks. myös taulukko 8.2
- **Tämä ei sovellu kaikkiin tarpeisiin, eräät prosessorit sallivat useita sivukokoja**
 - ◆ Pentium sallii 2 kokoa: 4KB tai 4MB
 - ◆ MIPS peräti 7 eri koko: 4KB ..16MB
- *Miten MMU tietää mitä kokoa käyttää?*

KJ-I S2004 / Tiina Niklander; kalvot Auvo Häkkinen

7 - 36

Table 8.2 Example Page Sizes

Computer	Page Size
Atlas	512 48-bit words
Honeywell-Multics	1024 36-bit word
IBM 370/XA and 370/ESA	4 Kbytes
VAX family	512 bytes
IBM AS/400	512 bytes
DEC Alpha	8 Kbytes
MIPS	4 kbytes to 16 Mbytes
UltraSPARC	8 Kbytes to 4 Mbytes
Pentium	4 Kbytes or 4 Mbytes
PowerPc	4 Kbytes
Itanium	4 Kbytes to 256 Mbytes

KJ-I S2004 / Tiina

7 - 37