

Käyttöjärjestelmät I

Luento 6: VIRTUAALIMUISTI Stallings, Luku 8.1

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 1

Sisältö

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

Luento 7:

- n Segmentointi
- n Segmentointi ja sivutus yhdistettynä
- n Yhteiskäytöstä

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 2

Käyttöjärjestelmät I

OHJELMAN SUORITUS VIRTUAALIMUISTISSA

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 3

Suoritus virtuaalimuistissa

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 4

Suoritus virtuaalimuistissa

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

= Virtuaalimuisti

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 5

Suoritus virtuaalimuistissa

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 6

Puutoskeskeytys (memory / page fault)

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 7

Puutoskeskeytys

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 8

Virtuaalimuistin etuja

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 9

Virtuaalimuistin etuja

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 10

Käyttöjärjestelmät I

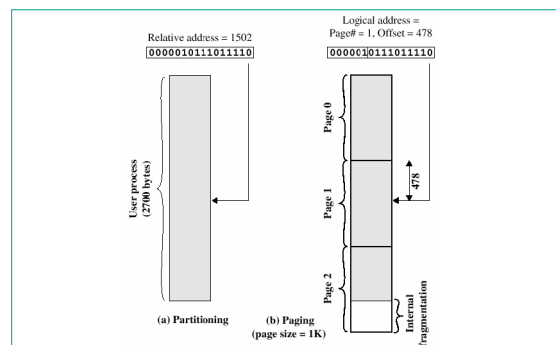
SIVUTUS

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 11

Sivutus

Kuva 7.11

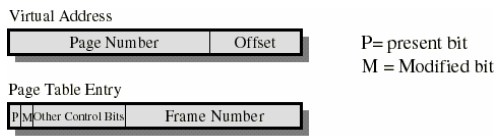


KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 12

Sivutaulu

Kuva 8.2a



- Jokaisella prosessilla oma sivutaulu
 - missä sivutiloissa tämän prosessin sivut sijaitsevat
- Jokaisessa alkiossa läsnäolobitti P
 - P=1: sivu muistissa, alkiossa sivutilan numero
 - P=0: sivu ei muistissa, alkiossa esim. tieto missä sivu sijaitsee tukimuistissa (suoraan/epäsuorasti)

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

Osoitemuunnos

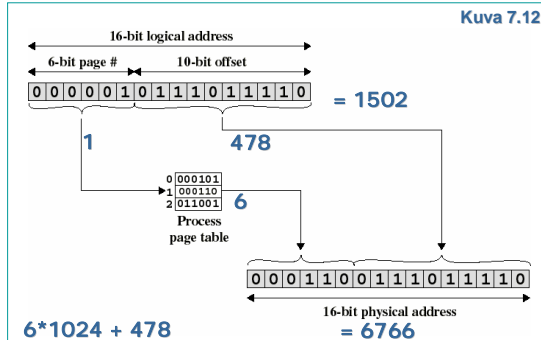
- Prosessien sivutaulut tavallisesti eri kokoisia ja voivat olla suuria
 - koko riippuu sivukooosta ja ohjelman koosta
 - koko taululle ei voi varata tilaa MMU:sta
 - sivutaulu muistissa ja osa jopa levyllä
 - sivutaulun fyysinen osoite PCB:ssä
- MMU:ssa sivutaulurekisteri PTR, jossa suoritettavan prosessin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

Osoitemuunnos

- MMU jakaa loogisen osoitteen pariaksi (sivunro,siirtymä)
 - Esim. kun sivukoko 1024 B (= 2¹⁰)
 - 10 viimeistä bittiä siirtymä
 - alkuosa sivunumero
- MMU korvaa sivunumero-bitit sivutaulusta löytyvillä sivutilanumero-biteillä
 - ts. MMU katenoi sivutilanumeron ja siirtymän bitit
- Helppo laitetointo

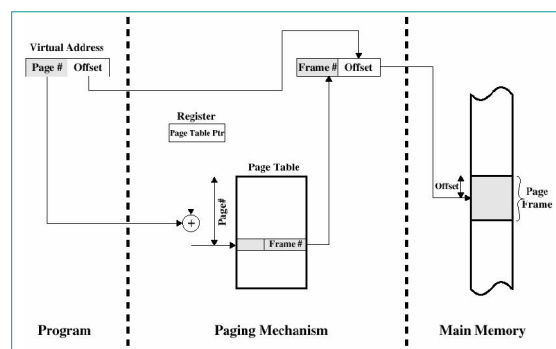
Looginen vs Fyysinen osoite

Kuva 7.12



Osoitemuunnos

Kuva 8.3



Käyttöjärjestelmät I

Osoitemuunnospuskuri TLB

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 19

Osoitemuunnospuskuri TLB

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 20

Etsintä TLB:stä

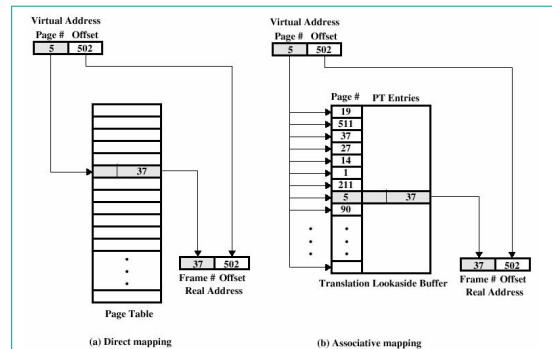
- n TLB nopeiden rekistereiden joukko
 - u esim. 32 rekisteriä
- n Käyttää assosiativista hakua
 - u vertailu kohdistuu kaikkiin alkioihin yhtäaikaan
 - u etsintä nopeaa
 - u laiteominto!
- n Kaikilla prosesseilla sivut 0, 1, 2, ..., mutta eri sivutilloissa
 - o TLB tyhjennettävä prosessin vaihdossa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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 21

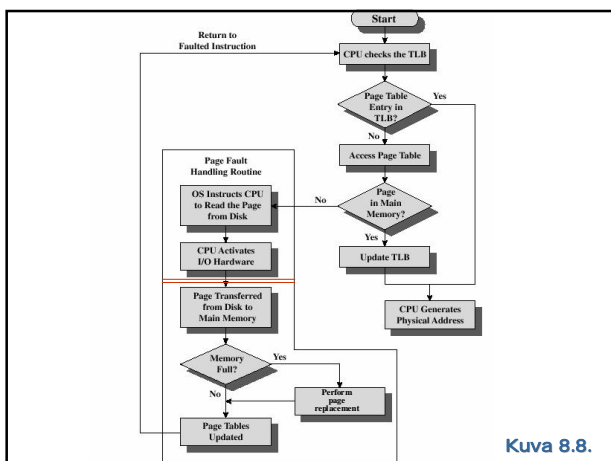
Etsintä TLB:stä

Kuva 8.9



KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 22



Kuva 8.8.

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)

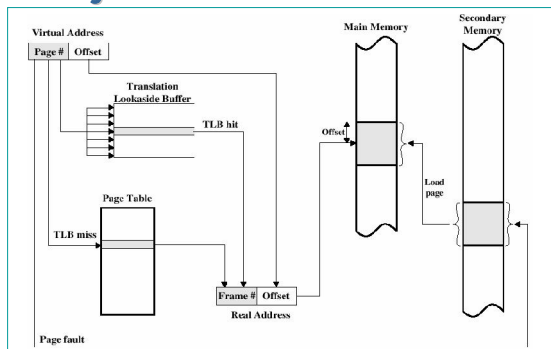
- n Kun keskeytys käsitelty, sama osoite tulee viltattavaksi uudelleen
 - u Esim. PC:n kasvatus vasta osoitemuunnoksen jälkeen

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 24

TLB ja osoitemuunnos

Kuva 8.7



KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 25

TLB:n alustus

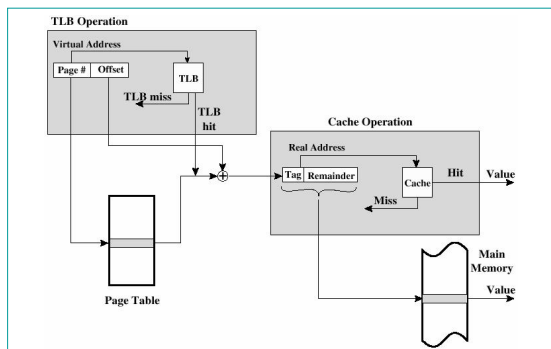
- n Kun suoritettava prosessi vaihtuu, TLB:n vanha sisältö mitätöitävä
 - u PTR osoittamaan uuden prosessin sivutauluun
 - u nollattava TLB:n alkioiden validiibitit V=0
- n Koska TLB suhteellisen pieni, tarvitaan sopiva laitetason algoritmi, jonka perusteella valitaan korvattava alkio
 - u TLB:n alkiossa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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 26

TLB ja välimuisti

Kuva 8.10



KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 27

Käyttöjärjestelmät I

Lisää sivutauluista

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 28

Lisää sivutauluista

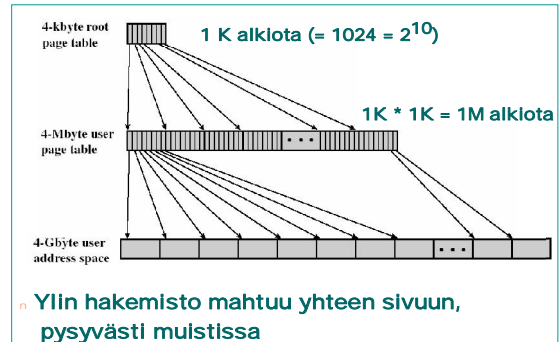
- n Monet järjestelmät sallivat suuren virtuaaliosoitteavaruuden
 - u looginen osoite esim. 32 tai 64 bittiä
- n Jokaisella prosessilla suuri sivutaulu
 - u jos 32-bittinen osoite ja sivukoko 4KB (12 bittiä), niin sivuja $2^{20} = 1M$ kappaletta
 - u jokainen alkio useita tavuja, esim. 4 B, joten sivutaulu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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 29

2-tasoinen sivutaulu

Kuva 8.4



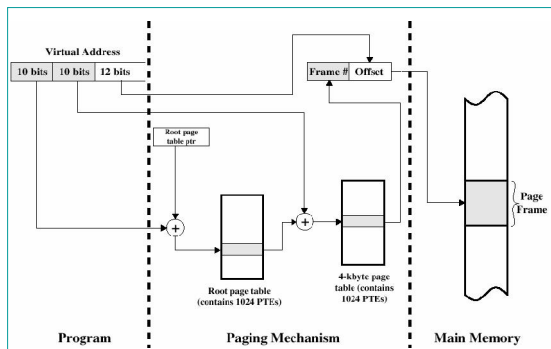
- n Ylin hakemisto mahtuu yhteen sivuun, pysyvästi muistissa

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 30

2-tasoinen sivutaulu

Kuva 8.5



KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 31

Sopiva sivukoko?

Laitteisto (MMU) määrää mitä sivukoko KJ:n käytettävä

- u sivukoko aina 2:sen potenssi
- F nopea osoitemuunnos
- F katkaisu ja katenointi helppoa

Mitä isommat sivut, sitä sivuja/prosessi

- u pienempi sivutaulu vie vähemmän tilaa
- u ison sivutaulun osia useammin levyllä
- F enemmän keskeytyksiä

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 32

Sopiva sivukoko?

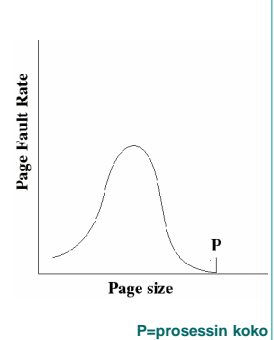
- n **Pieni:** aiheuttaa väh. sisäistä pirstoutumista
- n **Iso:** sisältää paljon esim. tarpeetonta koodia
- n **Iso:** saattaa sopia paremmin yhteen levysirtojen kanssa (lohkokoko)
 - u hakuvarren siirto ja pyörähdysviive syö paljon aikaa
- n **Iso:** viittaus useammin samalle sivulle
 - u TLB:n osumatodennäköisyys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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 33

Sivupuutosten määrä

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

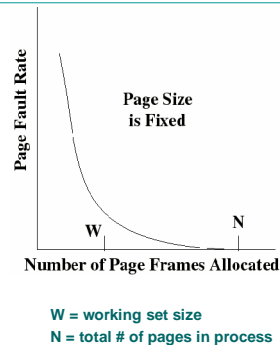


KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 34

Sivupuutosten määrä

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



KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 35

Sopiva sivukoko?

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

KJ-I S2005 / Tiina Niklander; kalvot Auvo Häkkinen

6 - 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 S2005 / Tiina

6 - 37