

Käyttöjärjestelmät I

Luento 4: KESKEISIÄ KÄYTTÖJÄRJESTELMIEN OSA-ALUEITA

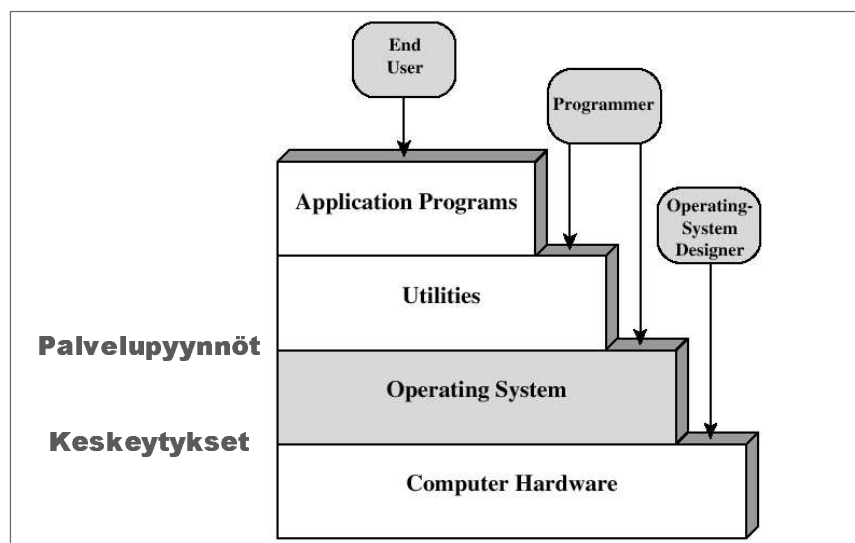
Stallings, Lukua 2

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 1

Järjestelmän tasot

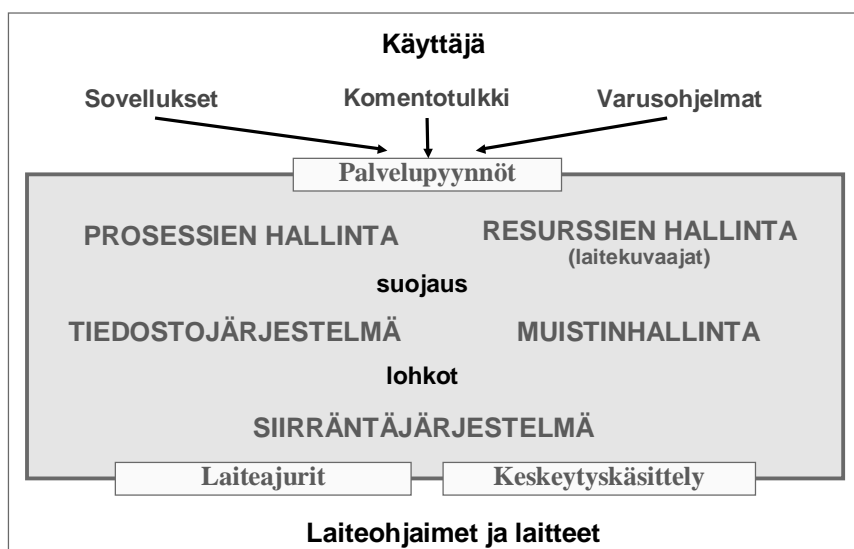
Kuva 2.1



KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 2

Keskeiset KJ:n osa-alueet



KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 3

Keskeisiä KJ:n osa-alueita

- **KJ eräs monimutkaisimmista ohjelmista**
- **Keskeisimmät osa-alueet (kirjan jaottelu)**
 - 1) Prosessin käsite
 - 2) Muistinhallinta
 - 3) Tietoturva ja suojaukset
 - 4) Vuorottaminen ja resurssien hallinta
 - 5) Järjestelmän hierarkkinen rakenne

virt. edellinen kuva

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 4

1) Prosessi

= Suoritettavaksi otettu ohjelma

- ◆ koodi muistissa (voi olla yhteiskäytössä)
- ◆ oma data-alue ja pino muistissa (muuttujat)
- ◆ kaikki ei välttämättä yhtäaikaan muistissa

+ KJ:n ylläpitämät hallinnolliset rakenteet

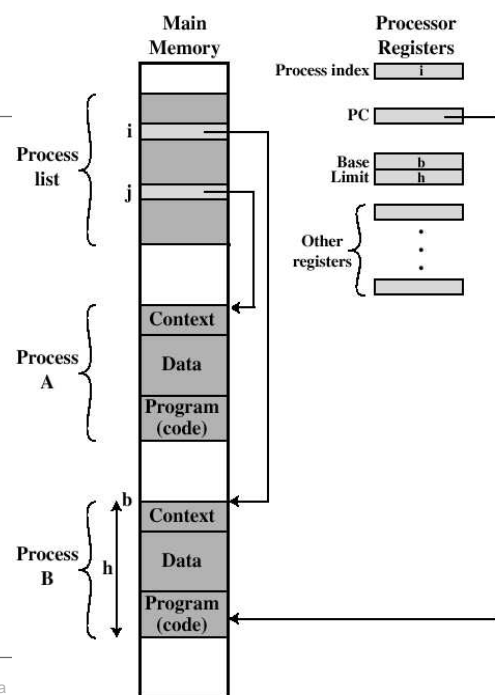
- ◆ prosessin kuvaaja PCB (Process Control Block)
 - CPU:n dataa prosessin suorittamisesta
 - prosessorin rekistereiden arvot
 - KJ:n dataa prosessin hallitsemiseksi
 - tunnistus, omistaja, prioriteetti ..
 - tietoja prosessin varaamasta muistista, tdstoista, ...
 - prosessin tila (esim. odottaa siirränän valmistumista)
 - ...

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 5

Eräs toteutus

- KJ:llä prosessilista, jossa viitteet prosessin kuvaajiin (context)
- CPU:ssa rekisteri, jossa suoritettavan prosessin numero (prosessin B numero on i)
- Prosessinvaihto: CPU A:lta B:lle
 - ◆ KJ talletti rekistereiden arvot A:n kuvaajaan
 - ◆ KJ latsi B:n kuvaajasta arvot rekistereihin



KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

2) Muistinhallinta

- **Suoritusaikainen tallennus**
 - ◆ prosessit (ohjelmat+data) keskusmuistissa (primary memory)
- **Pysyvä tallennus**
 - ◆ tiedostot (ohjelmat+data) 'tukimuistissa' (secondary memory)
- **Tilan allokointi**
 - ◆ KJ huolehtii automaattisesti
 - ◆ kirjanpito vapaista muistialueista / levylohkoista
 - ◆ kirjanpito varatuista muistialueista / levylohkoista
 - ◆ tarvittaessa KJ käyttää levyä muistin jatkeena (virtuaalimuisti)
- **Suojaus ja käyttöoikeudet**
 - ◆ prosessien eristäminen toisistaan
 - ◆ silti tuettava modulaarista ohjelmointia
 - ↳ koodin / datan yhteiskäyttö sallittua
 - ◆ muistinsuojaus, tdstojen käyttöoikeudet

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 7

Muistinhallinta

Avainkäsite virtuaalimuisti

- **Ei ota kantaa todellisen muistin määrään tai minne ohjelma muistissa sijoitetaan**
 - ◆ ohjelman muistiavaruus voi olla suurempi kuin fyysinen muistiavaruus (ts. todellinen muistin määrä)
- **Suoritusaikana muistissa tarvittava osa koodista/datasta, loput levyllä**
 - ◆ MMU huomaa puuttumisen
 - ◆ KJ lataa muistiin
- **Ei näy sovelluksen ohjelmoijalle**

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 8

Virtuaalimuisti

- **Ohjelma käyttää virtuaaliosoitteita, ts. osoitteita ohjelman alun suhteen**
- **Laitteisto käyttää aina fyysisiä (todellisia) osoitteita**
- **MMU tekee osoitemuunnoksen aina suoritusaikana muistiin viitattaessa**
- **Jos viitattu osa ei keskusmuistissa**
 - ◆ MMU aiheuttaa keskeytyksen
 - ◆ jos muistissa ei ole vapaata tilaa, KJ vapauttaa jonkin alueen
 - ◆ KJ hakee levyltä puuttuvan osan muistiin

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 11

3) Tietoturva ja suojaus

- **Käyttäjien tunnistus**
 - ◆ Vain tunnuksen & salasanan tuntevat käyttäjät saavat ottaa istunnon koneeseen
 - ◆ Prosessilla aina omistaja
- **Resurssien käyttöoikeus**
 - ◆ Prosessi käyttää resurssia vain omistajan luvalla
 - tdstoihin liittyy omistaja ja käyttöoikeudet
 - vain omistaja voi muuttaa käyttöoikeuksia
 - ◆ Ohjelmat ja data suojattava toisilta ohjelmilta
 - erityisen tärkeää on suojata KJ sovelluksilta
 - MMU ja ajonaikainen osoitemuunnos
 - ◆ Resurssien yhteiskäyttö silti sallittava

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 12

4) Resurssien hallinta

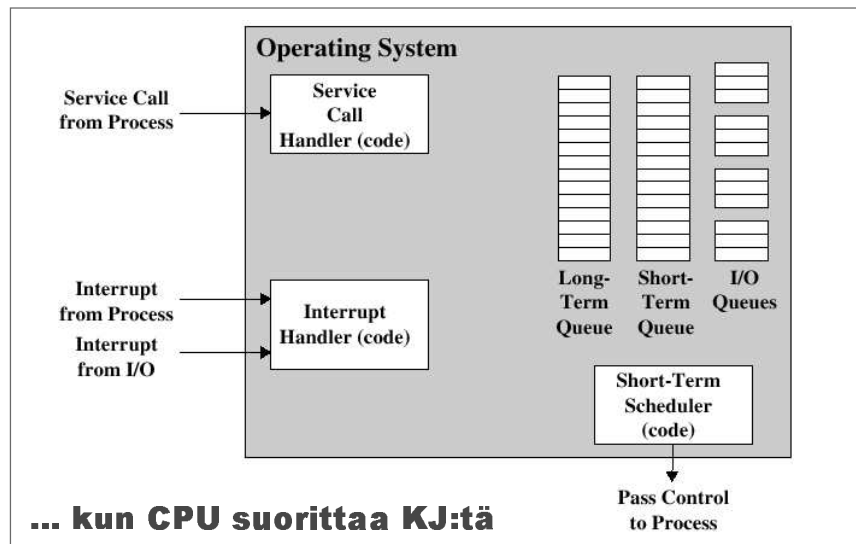
- **Resurssi?**
 - ◆ CPU, muisti, tdsto, I/O-laite ...
 - ◆ CPU:n allokointi = vuorottaminen (skedulointi)
- **Milloin?**
 - ◆ Prosessia käynnistettäessä, suoritettaessa, tapettaessa
 - ◆ KJ:n päätöksillä, prosessin (palvelu)pyyntöjen perusteella
- **Vastausaika**
 - ◆ Interaktiivisuus vs. tausta-ajo (eräajo)
 - ◆ KJ:n palveluprosessi vs. sovellus
- **Tasapuolisuus**
 - ◆ Samanlaisille prosesseille samantasoinen palvelu
- **Tehokkuus**
 - ◆ Maksimoi läpimenoaste, minimoi vastausajat
 - ◆ Palvele mahd. useaa käyttäjää (sovellusta)

Vuorottamisen perusideoita

- **Suoritukseen otetuille prosesseille**
READY-jono (short term queue)
 - ◆ *vuorottaja* valitsee seuraavaksi suoritettavan prosessin (jonon ensimmäinen)
 - ◆ Round-Robin: uusi työ jonon loppuun ja CPU:lta pois tuleva työ jonon loppuun
- **Suoritettavaksi ottamista odottaville prosesseille oma jono** (long term queue)
 - ◆ ei liian monta prosessia yhtäaikaan READY-jonoon (moniajoaste)
- **Kullakin tapahtumalla omat odotusjononsa**
 - ◆ I/O-laitteet, semaforit, ajastimet, ...

Milloin?

Kuva 2.11



KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 15

5) Hierarkkinen rakenne

- **KJ muodostuu useista erillisistä tasoista**
- **Hierarkian ansiosta kokonaisuus jakaantuu helpommin hallittaviin osiin**
 - ◆ suunnittelu, toteutus, testaus
- **Kullakin tasolla oma osajoukko tehtävistä**
 - ◆ ylemmillä tasoilla kehittyneimmät KJ:n palvelut
 - ◆ alemmilla tasoilla laiteläheisimmät toiminnot
- **Taso tarjoaa palvelunsa ylemmälle tasolle, käyttää alemman/alempien tasojen palveluja**
- **Rajapinnat hyvin määritellyjä**
 - ◆ tason toteutusta voi muuttaa koskematta muihin tasoihin

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 16

Laitetasot

Brown, Denning 1984

Level	Name	Objects	Example Operations
4	Interrupts	Interrupt-handling programs	Invoke, mask, unmask, retry
3	Procedures	Procedures, call stack	Mark stack, call, return
2	Instruction Set	Evaluation stack, micro-program interpreter, scalar and array data	Load, store, add, subtract, branch
1	Electronic circuits	Registers, gates, buses, etc.	Clear, transfer, activate, complement

Mitä laitteistopiirteitä tarvitaan KJ:n toteutuksessa?

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 17

KJ:n laiteriippuvat tasot

Level	Name	Objects	Example Operations
7	Virtual Memory	Segments, pages	Read, write, fetch
6	Local secondary store	Blocks of data, device channels	Read, write, allocate, free
5	Primitive processes	Primitive process, semaphores, ready list	Suspend, resume, wait, signal P / V

Muistinhallinta: MMU:n rakenteen huomiointi

Laitteajurit: ohjaimien ja esim. levyn rakenteen huomiointi

Vuorottaja: rekistereiden talletus/palautus, synkronointiprimitiivit

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 18

KJ:n laiteriippumattomat palvelut

Level	Name	Objects	Example Operations
13	Shell	User programming environment	Statements in shell language
12	User processes	User processes	Quit, kill, suspend, resume
11	Directories	Directories	Create, destroy, attach, detach, search, list
10	Devices	External devices, such as printer, displays and keyboards	Create, destroy, open, close, read, write
9	File system	Files	Create, destroy, open, close, read, write
8	Communications	Pipes	Create, destroy, open, close, read, write

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 19

KJ-toteutuksen vaikeat kohdat

- **Toimintojen synkronointi / ajoitus**
 - ◆ (seuraava kalvo)
- **Poissulkeminen** (Mutual exclusion)
 - ◆ eräitä resursseja voi käyttää vain yksi kerrallaan
 - ↳ esim. kirjoitin tai yhteinen tdsto / data
- **Lukkiutuminen / Nälkiintyminen** (Deadlock, Starvation)
 - ◆ vaikeaa havaita
 - ◆ odotettava resursseja, jotka ovat toisten hallussa
 - ◆ huono prioriteetti, ei saada ensinkään palvelua

ks. RIO-kurssi

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 20

KJ-toteutuksen vaikeat kohdat

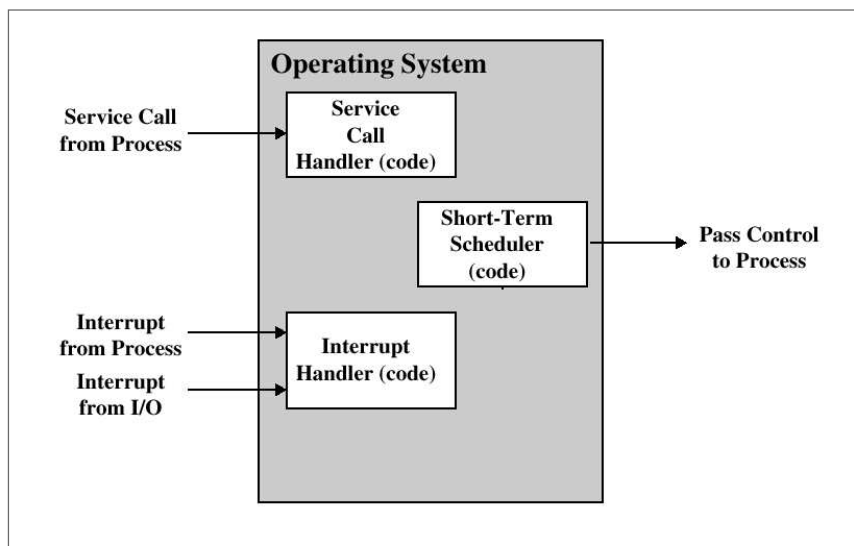
■ Toimintojen synkronointi / ajoitus

- ◆ joskus odotettava että jotain muuta tapahtuu ennenkuin voi jatkaa **ks. RIO-kurssi**
- ◆ keskeytysten oikea priorisointi
- ◆ laitteistosignaalit tai ohjelmien toisilleen välittämät tiedot eivät saa kadota tai kahdentua
- ◆ Epädeterministinen käyttäytyminen
 - Jaettu muisti ja erilainen vuorotus voi tuottaa yllätyksen
 - Tulos riippuu suoritusjärjestyksestä

Käyttöjärjestelmät I

PALVELUPYYNNÖT

Palvelupyyntö



KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 23

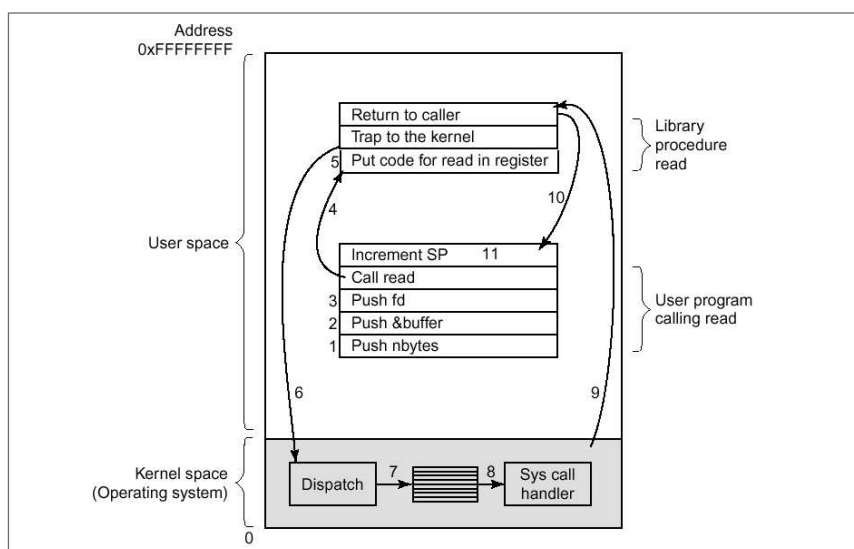
Palvelupyynnöt

- **Sovellus pyytää KJ:n palvelua käskykantaan kuuluvan käskyn välityksellä**
 - ◆ Prosessit ja niiden välinen kommunikointi
 - ◆ Muisti
 - ◆ Tiedostot ja tiedostojärjestelmä
 - ◆ Siirräntä
 - ◆ Ym.
- **Parametrit pinoon, sitten käsky esim. SVC**
- **Palvelupyyntö aiheuttaa keskeytyksen**
 - ◆ CPU etuoikeutettuun tilaan
 - ◆ CPU suorittamaan KJ:tä

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 24

read(fd, buffer, nbytes) Tan01 1-17



KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 25

POSIX palvelupyynnöt

Process management

Call	Description
<code>pid = fork()</code>	Create a child process identical to the parent
<code>pid = waitpid(pid, &statloc, options)</code>	Wait for a child to terminate
<code>s = execve(name, argv, environp)</code>	Replace a process' core image
<code>exit(status)</code>	Terminate process execution and return status

File management

Call	Description
<code>fd = open(file, how, ...)</code>	Open a file for reading, writing or both
<code>s = close(fd)</code>	Close an open file
<code>n = read(fd, buffer, nbytes)</code>	Read data from a file into a buffer
<code>n = write(fd, buffer, nbytes)</code>	Write data from a buffer into a file
<code>position = lseek(fd, offset, whence)</code>	Move the file pointer
<code>s = stat(name, &buf)</code>	Get a file's status information

Portable Operating System ("UNIX Style")

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 26

POSIX palvelupyynnöt

Tan01 1-18

Directory and file system management

Call	Description
s = mkdir(name, mode)	Create a new directory
s = rmdir(name)	Remove an empty directory
s = link(name1, name2)	Create a new entry, name2, pointing to name1
s = unlink(name)	Remove a directory entry
s = mount(special, name, flag)	Mount a file system
s = umount(special)	Unmount a file system

Miscellaneous

Call	Description
s = chdir(dirname)	Change the working directory
s = chmod(name, mode)	Change a file's protection bits
s = kill(pid, signal)	Send a signal to a process
seconds = time(&seconds)	Get the elapsed time since Jan. 1, 1970

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 27

Käyttöesimerkki

Tan01 1-19

```
#define TRUE 1

while (TRUE) {
    type_prompt( );          /* repeat forever */
    read_command(command, parameters); /* display prompt on screen */
                                /* read input from terminal */

    if (fork() != 0) {
        /* Parent code. */
        waitpid(-1, &status, 0); /* fork off child process */
    } else {
        /* Child code. */
        execve(command, parameters, 0); /* wait for child to exit */
        /* execute command */
    }
}
```

”Riisuttu” komentotulkki

KJ-I S2004 / Tiina Niklander Auvo Häkkisen kalvojen pohjalta

4 - 28

WIN32 API palvelupyynnöt

UNIX	Win32	Description
fork	CreateProcess	Create a new process
waitpid	WaitForSingleObject	Can wait for a process to exit
execve	(none)	CreateProcess = fork + execve
exit	ExitProcess	Terminate execution
open	CreateFile	Create a file or open an existing file
close	CloseHandle	Close a file
read	ReadFile	Read data from a file
write	WriteFile	Write data to a file
lseek	SetFilePointer	Move the file pointer
stat	GetFileAttributesEx	Get various file attributes
mkdir	CreateDirectory	Create a new directory
rmdir	RemoveDirectory	Remove an empty directory
link	(none)	Win32 does not support links
unlink	DeleteFile	Destroy an existing file
mount	(none)	Win32 does not support mount
umount	(none)	Win32 does not support mount
chdir	SetCurrentDirectory	Change the current working directory
chmod	(none)	Win32 does not support security (although NT does)
kill	(none)	Win32 does not support signals
time	GetLocalTime	Get the current time