

# Protsessidest

- Protsessid
- Lõimed

# Mis on protsess

- Protsess on täidetav programm mis sisaldab:
  - koodiosa (*text section*);
  - käsuloendur (*program counter*);
  - protsessori registrid;
  - pinu (*stack*) - ajutised andmed (funktsioonide muutujad);
  - andmesektsioon (*data section*) - globaalsed andmed.

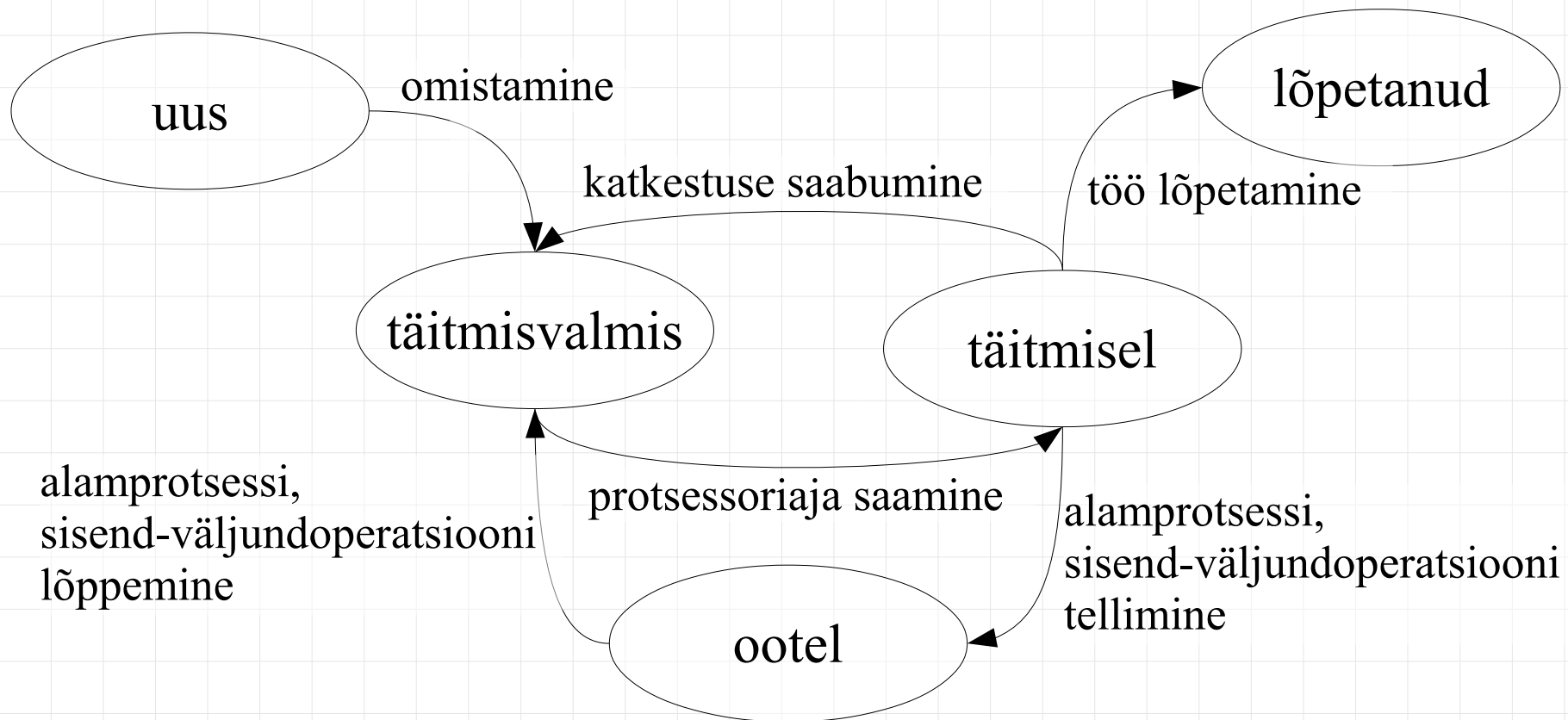
- Programm on passiivne olem
- Protsess on programmi töötav e. aktiivne koopia

# Protsessi olek

Protsess satub oma töötamise jooksul järgmistesse olekutesse:

- uus (*new*) luuakse protsess;
- töös (*running*) täidetakse antud protsessi käske;
- ootel (*waiting*) oodatakse mingi sündmuse toimumist;
- valmis (*ready*) oodatakse protsessoriaega;
- lõpetatud (*terminated*) programm on lõpetatud.

# Protsessi olekud



# Kontekstivahetus

- Ühelt protsessilt teisele lülitumisel tuleb salvestada vana protsessi olek ja laadida uue protsessi salvestatud olek
- See aeg on lisakulu, vähendab rakendustele jäävat protsessoriaega
- Kuluv aeg sõltub protsessori võimalustest

# Protsesside loomine

- Vanemprotsessid võivad luua lapsprotsesse
- Iga lapsprotsess võib olla teistele protsessidele omakorda vanemaks
- Sedasi tekib protsesside puu

# Protsesside loomine

- Ressursside jagamine
  - Vanem ja laps jagavad kõiki ressursse
  - Laps saab alamhulga vanema ressurssidest
  - Vanem ja laps ei jaga mingeid ressursse
- Töötamine
  - Vanem ja laps töötavad paralleelselt
  - Vanem ootab, kuni laps töö lõpetab





# Käsk *ps*

- UNIXis protsessitabelisse piilumiseks
  - PPID – vanemprotsessi number
  - TTY – terminal
  - PRI – prioriteet
  - NI – nice väärtus
  - SIZE – mälu kasutus
  - CMD – käsuriid
- ***ps -efl*** – kõik parajasti käimasolevad protsessid täielikult ja pikalt.

# Protsside jälgimine Windowsis

- Tegumihaldur (*Task Manager*) – OS'ga kaasas
- Process explorer (freeware) veebilehelt:  
<http://live.sysinternals.com/>

# Tegumihaldur (Task manager)

- MS Windows
- Rakendused – aknad
- Protsessid – tavakasutaja näeb enda ja süsteemiprotsesse
- Jõudlus – protsessori ja mälu hõivatus
- Võrgundus – võrgukaardi koormatus

# Sysinternals – process explorer

The image shows two windows from Sysinternals Process Explorer. On the left is the 'Color Selection' dialog, which allows users to customize the colors of various process categories. On the right is the main 'Process Explorer' window, displaying a list of running processes with columns for CPU usage and Private Bytes.

**Color Selection Dialog:**

Category	Color	Checked	Action
New Objects	Green	<input checked="" type="checkbox"/>	Change...
Deleted Objects	Red	<input checked="" type="checkbox"/>	Change...
Own Processes	Light Blue	<input checked="" type="checkbox"/>	Change...
Services	Pink	<input checked="" type="checkbox"/>	Change...
Suspended Processes	Grey	<input checked="" type="checkbox"/>	Change...
Packed Images	Purple	<input checked="" type="checkbox"/>	Change...
Relocated DLLs	Yellow	<input type="checkbox"/>	Change...
Jobs	Brown	<input type="checkbox"/>	Change...
.NET Processes	Light Yellow	<input type="checkbox"/>	Change...
Immersive Process	Cyan	<input checked="" type="checkbox"/>	Change...
Graph Background			Change...

**Process Explorer - Sysinternals:**

Process	CPU	Private Bytes
services.exe	0.01	2,412,000
svchost.exe	0.03	3,728,000
dllhost.exe		1,440,000
SkyDrive.exe	0.21	6,336,000
SettingSyncHost.exe		10,240,000
WShost.exe		6,816,000
FlashUtil_ActiveX.exe		2,608,000
WmiPrvSE.exe	2.15	4,128,000
livecomm.exe		15,744,000
RuntimeBroker.exe		3,328,000
svchost.exe	0.04	3,728,000
svchost.exe	0.05	19,200,000
audiodg.exe		5,504,000
svchost.exe	0.21	41,728,000
taskhost.exe		5,920,000

# Protsessi lõpetamine

- Protsess võib lõppeda vabatahtlikult (*exit*):
  - vanemale antakse tagasi väljundinfo (*wait*);
  - operatsioonisüsteem vabastab protsessiga seotud ressursid.
- Sunniviisiliselt – lõpetajaks võib olla vanemprotsess (*kill*) või operatsioonisüsteem:
  - vanemprotsess sureb ja opsüsteem tapab protsessi
  - Lapsprotsess on ületanud talle ressursside kasutamiseks seotud piirid
  - Lapsprotsess on oma ülesande täitnud

# Käsk *kill* (UNIX)

- Sellega saadetakse protsessile teatud signaal
  - Töö lõpetamine;
  - Töö ajutine peatamine;
  - Töö jätkamine.
- Signaalide tabelit näeb käsuga  
**kill -l**
- Signaale saab saata ainult enda protsessidele
  - Protsessi töö lõpetamine: `kill -15 1234`
  - Sama tulemus: `kill -TERM 1234`

## Käsk *kill* (2)

- Mõned protsessid ei reageeri TERM signaalile
  - Neile võib saata signaali HUP(1)
- Viimase võimalusena kasutada signaali KILL(9)
  - Käsk on blokeerumiskindel ja lõpetab koheselt töö
  - Avatud sisend-väljundseadmed, võrguühendused katkestatakse ilma salvestamata



# Protsessid - prioriteedid

- **nice** – käsk programmi teatud prioriteediga käivitamiseks
- Vahemik -20 (kõrgeim) kuni 19 (madalaim)
- Ilma parameetrita antakse vaikeprioriteet
- **nice -4 <programm>**
  - Käivitada programm prioriteediga -5
- **renice** – käimasoleva protsessi prioriteedi muutmiseks

# Protsesside koostöö

- Sõltumatud protsessid ei saa teisi protsesse mõjutada ja neid ei mõjuta teised protsessid
- Koostööd tegevad protsessid saavad teisi protsesse mõjutada ja/või neid mõjutavad teised protsessid
- Miks protsesside koostööd vaja on?
  - Info jagamine
  - Töö kiirendamine
  - Modulaarsus
  - Mugavus

# Eellaetud protsessid (Windows)

- Osa programme võivad olla arvuti käivitamise ajal osaliselt ette laetud
  - MS Office
  - OpenOffice.org
  - Adobe Reader jpm
- Plussid:
  - Programmi käivitamise aeg lüheneb
- Miinused:
  - Arvuti (OS laadimine) käivitumine on aeglasem,
  - kulutatakse rohkem ressursse

# Eellaetud protsessid – kuidas neid ette laetakse

- Tavaliselt on otseteed MS Windowsis  
Start->Programs->Startup
- Või MS Windowsi registris  
[HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run]  
[HKLM\Wow6432Node\SOFTWARE\Microsoft\Windows\CurrentVersion\Run]
- Registris on kohti veel kuhu kirjutada:
  - Kasutaja **Run** jne.
- Sysinternals autoruns: <http://live.sysinternals.com/>
- Windowsi services

# msconfig

- MS Windows
- Käivitada saab ainult administraator
- msconfig – (System Configuration Utility) – arvuti alglaadimise aegsete teenuste, programmide käivitamise lubamine/keelamine.
- Kasutatakse probleemide lahendamisel.

# services.msc (Windows)

- Võimaldab vaadata arvutis käimasolevaid teenuseid, samuti nende käivitustingimusi muuta.

# Protsessid, mis käivitatakse koos operatsioonisüsteemiga (Linux)

- Redhat Linuxis
  - /sbin/chkconfig –list
- Gentoo Linuxis
  - /etc/init.d kataloog
  - Käsk rc-status
  - <http://linuxreviews.org/man/rc-status/>

# Protsesside käivitamine teatud aegadel (UNIX)

- **at** – käivitab käsu ettenähtud ajal
- **atq** – kuvab ootel käske
- **atrm** – eemaldab käske ootenimekirjast
- **cron** – käskude käivitamine regulaarselt



# Protsesside käivitamine teatud aegadel (Windows)

- AT – võimaldab ühekordset käivitust ning perioodilist.
  - Siin saab kasutada pakkfaili.

# Protsess ja tegum (task)

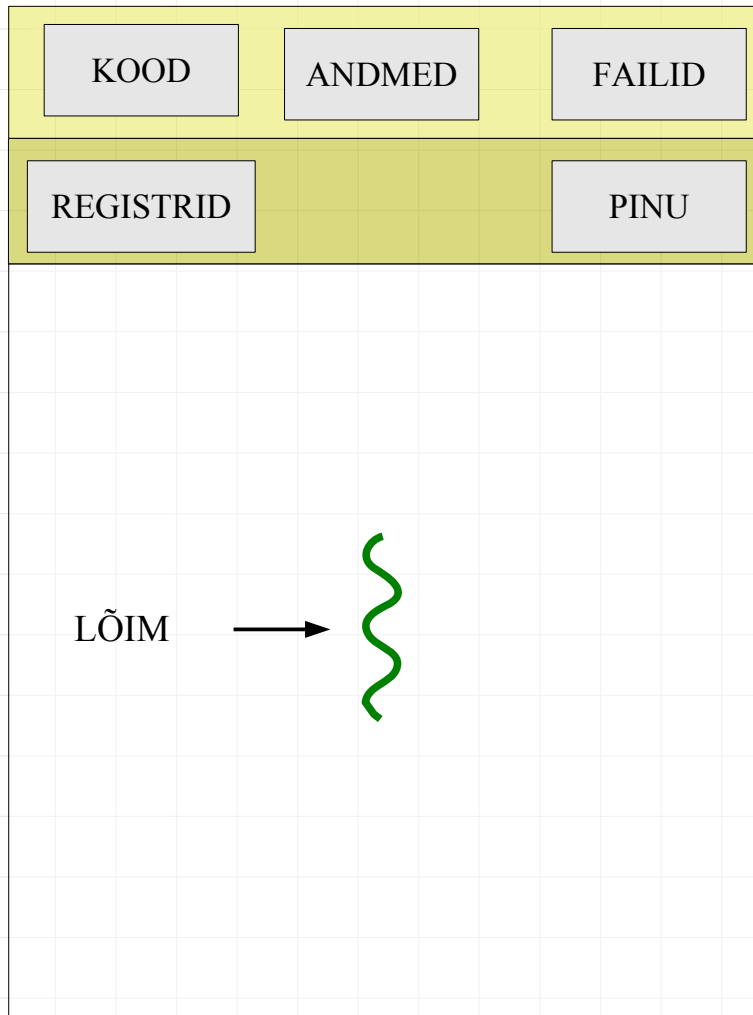
- Protsess ei pea olema seotud kindla aknaga.
- Nt MacOS operatsioonisüsteemis akna kinni panemine ei tähenda protsessi sulgemist.
- Protsess + tema lõimed = tegum (*task*)

# Lõimed

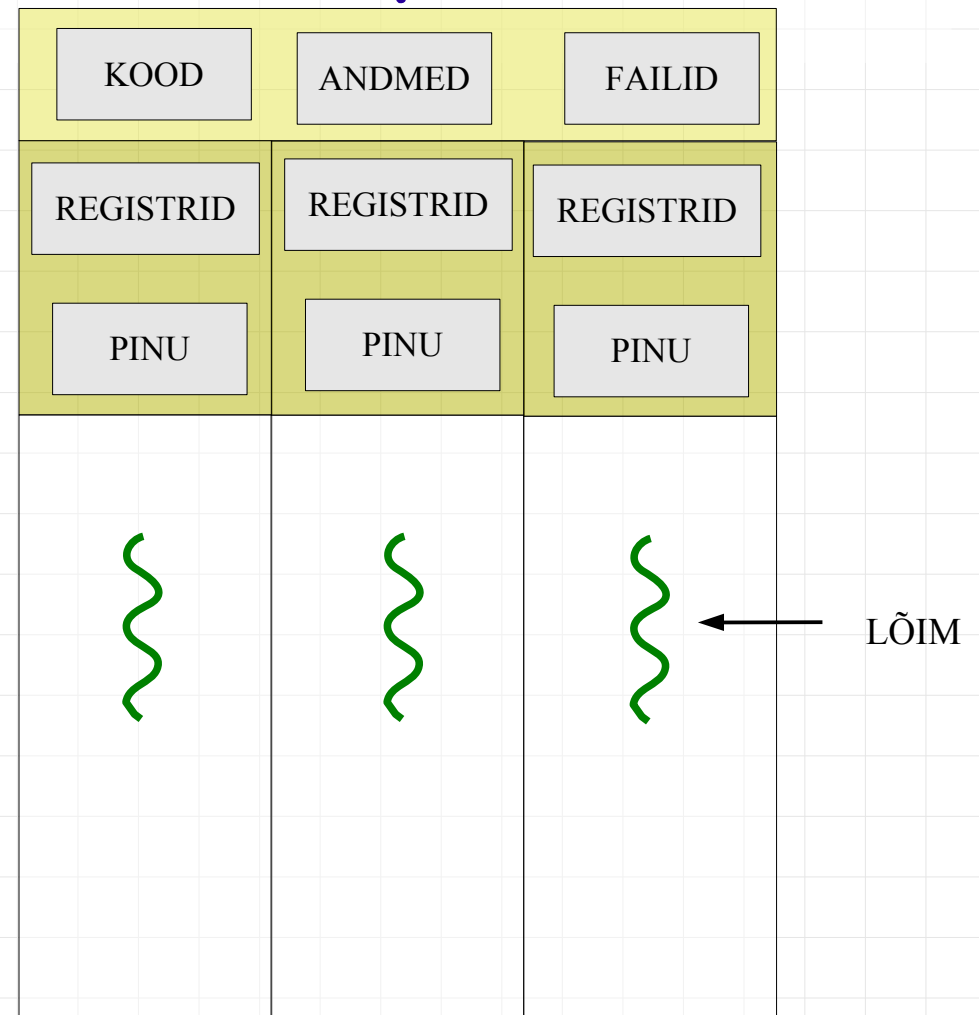
# Lõimed

- Kontekstivahetus on aeglane protseduur
- Lõim – kergekaaluline protsess
  - käsuloendur, registrite sisu, pinu
- Protsess + tema lõimed = tegum (*task*)

# Ühe- ja mitmelõimelised protsessid



ühelõimeline protsess



mitmelõimeline protsess

# Milleks lõimed?

- **Reageerimiskiirus** – parema interaktiivsuse saamiseks võib kasutajaliidese ja muu töö eraldi lõimedesse panna
- **Ressursside jagamine** – aadressipiirkond on programmi mitme täitja poolt kasutatav
- **Kokkuvõid** – hoiame kokku jagatud ressursside, uute protsesside loomise ning kontekstivahetuste pealt
- **Mitme protsessori ära kasutamine ühe programmi poolt**

# Kasutaja taseme lõimed

- Lõimede haldus on realiseeritud kasutaja tasemel teegina
- Tuum ei tea midagi, tema jaoks üks ühelõimeline protsess
- Haldus on kiire, kuna toimub kasutaja tasemel
- Näiteks
  - POSIXi Pthreads
  - Mach'i C-threads
  - Solarise lõimed (UI-threads)
  - FreeBSD oma lõimed

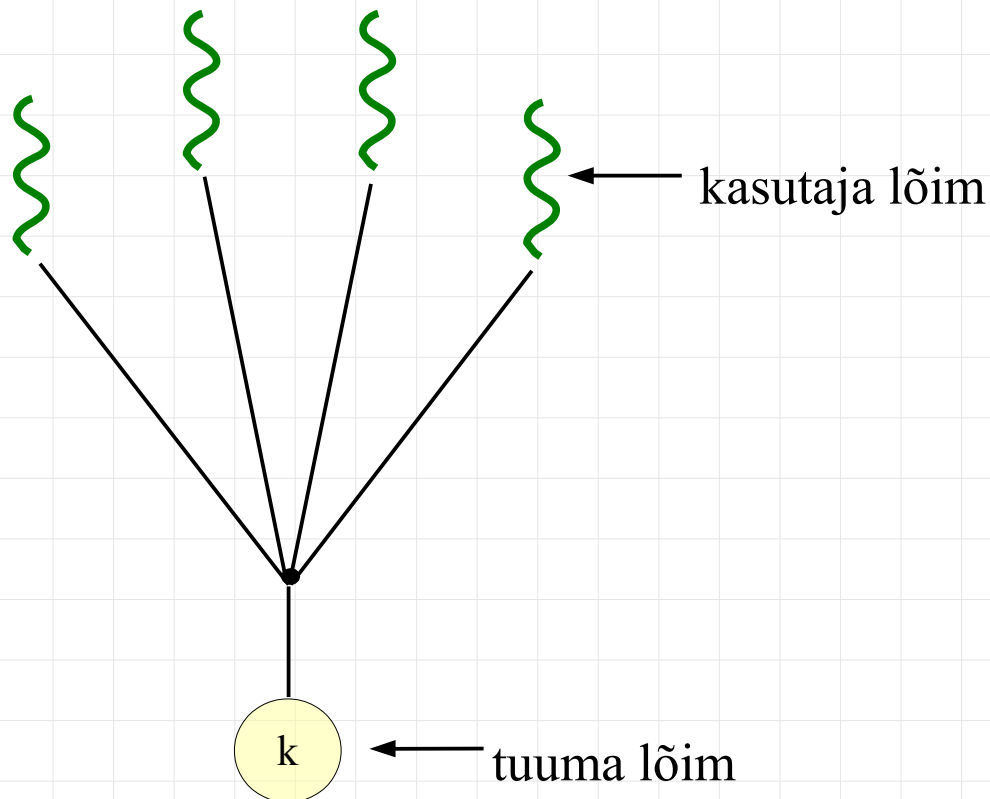
# Tuuma lõimed

- Toetatud tuuma poolt, tuum jagab nende lõimede vahel aega
- Haldus kulukam kui kasutaja tasemel
- Näiteks
  - Windows 95/98/NT/2000
  - Solaris
  - Tru64 UNIX
  - BeOS
  - Linux



# Lõimede mudelid

- Mitu-ühele — protsessi kõigi kasutaja tasemel lõimede jaoks on üks tuuma lõim.

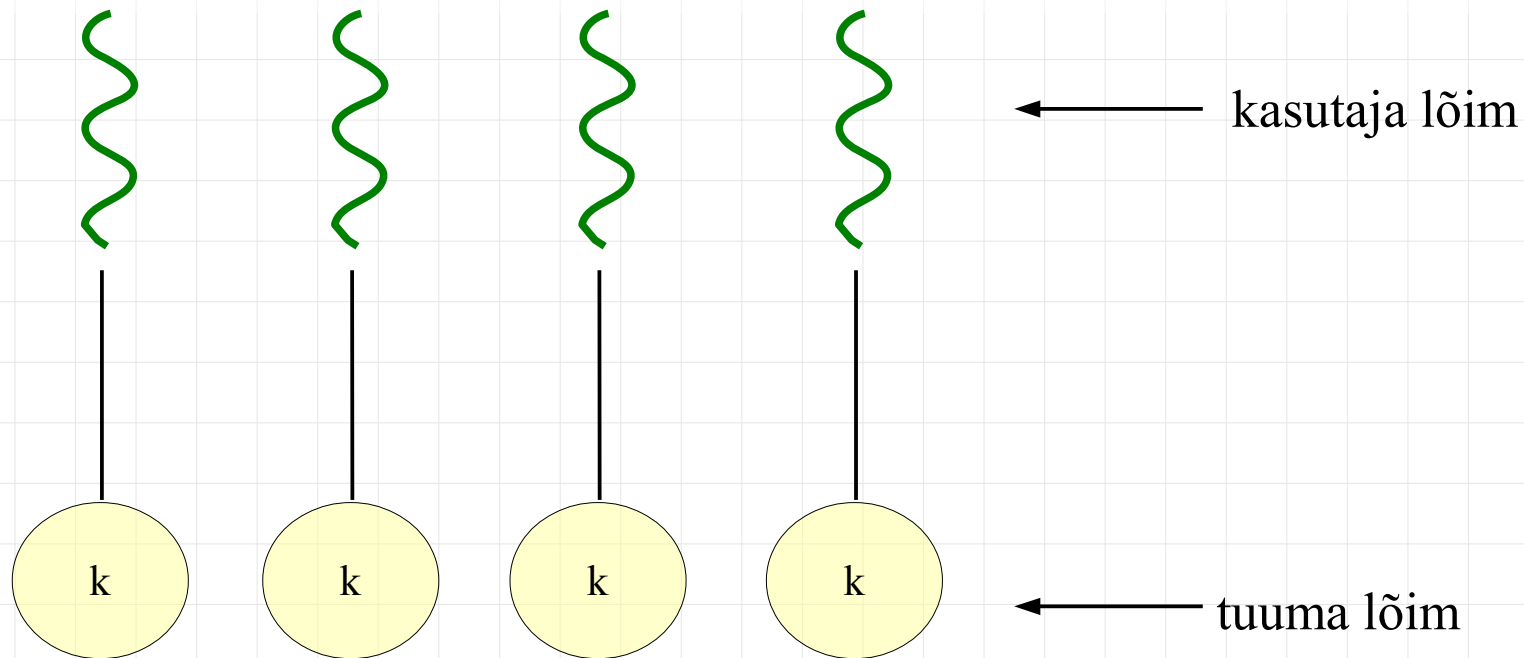


# Lõimed – mitu-ühele mudel

- Haldus kiire, kuna haldus toimub kasutaja tasemel.
- Töötada saab üks lõim korraga tuuma lõimega.
- Blokeeruva süsteemikutse kasutamisel blokeerub kogu protsess.

# Lõime mudelid

- Üks-ühele — iga kasutaja taseme lõimele vastab tuuma lõim

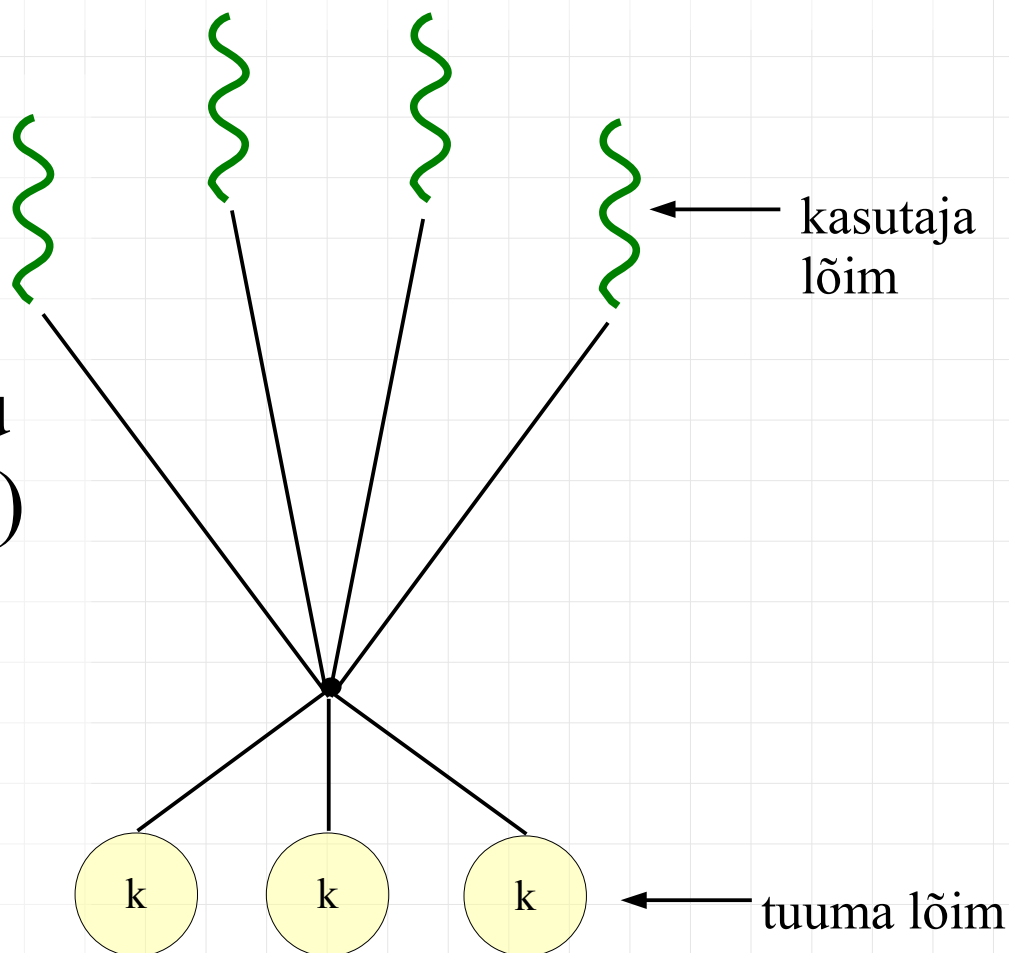


# Lõimed – üks-ühele mudel

- Haldus on kulukam
- Saab kasutada mitut protsessorit
- Saab korraga teha mitut blokeeruvat operatsiooni
- Ei maksa liiga palju lõimi teha
- Windows NT/2000 ja OS/2

# Lõimele mudelid

- Mitu-mitmele — protsessi paljudele kasutaja taseme lõimelele vastab palju (samapalju või vähem) tuuma lõimesid



# Lõimed – mitu-mitmele mudel

- Kombineerib eelmiste head küljed
- Keerukam
- Solaris 2
- Windows NT/2000 *ThreadFiberi* teegi abil

# Solarise lõimed

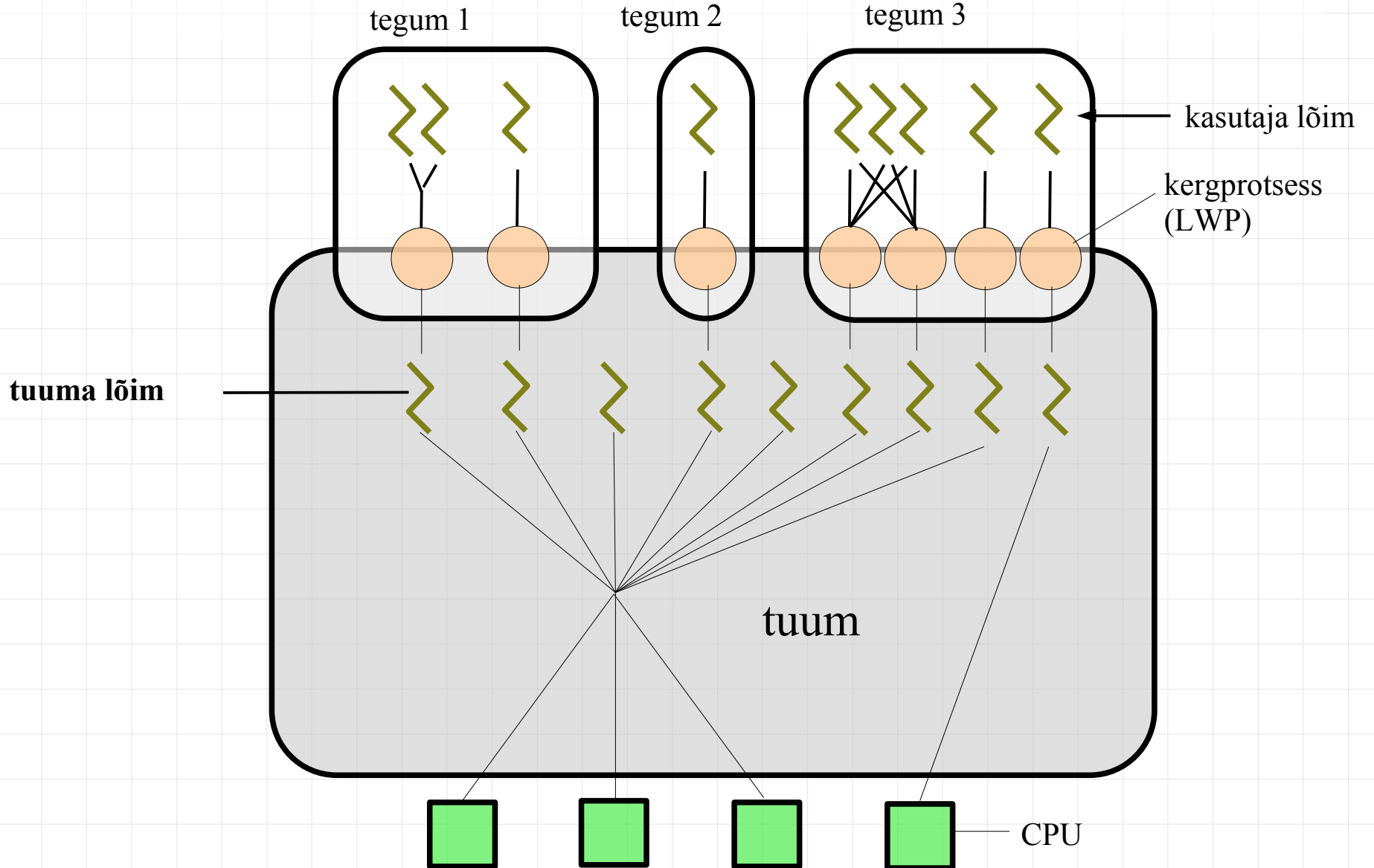
- on läbini lõimeline
- LWP (lightweight process) kergprotsess (lõimede liides kasutaja ja tuuma tasemete vahel)
- Igale LWP-le vastab tuuma lõim, tuumas võib lõimesid veel olla
- Kasutajataseme lõimesid saab vajadusel LWP-dega siduda (LWP-sid läheb vaja siis, kui on vaja tuuma teenuseid kasutada)

# Solarise lõimed

- Sidumata lõimed jagavad omavahel mingit hulka LWP-sid
- Protsessi poolt kasutatavate LWP-de arvu reguleerib lõimede teek
- automaatselt (saab ka käsitsi)



# Solarise lõimed



# Tänaseks kõik