

Mäluhaldus

Ülevaade

- Mõisted;
- Pidevate mälualade hõivamine;
- Lehekülgede saalimine.

Mälu

- Mälu – adresseeritud baidimassiiv.
- Protsessor saab mälust käske ja andmeid
 - loe käsk;
 - loe esimene argument;
 - loe teine argument;
 - täida käsk;
 - salvesta tulemus mällu.



Aadresside sidumine

- Enamuse elust veedab programm kõvakettal.
- Tema täitmiseks tuleb programm tuua mällu.
- Programmi täitmiseks tuleb koodis sisalduvad sümbolaadressid seostada tegelike mäluaadressidega; see võib toimuda:
 - 1) **Kompileerimise ajal;**
 - 2) **Laadimise ajal;**
 - 3) **Täitmise ajal.**

Aadresside sidumine

1) Kompileerimise ajal:

- mälupaigutus on ette teada;
- genereeritakse kood absoluutaadressidega;
- paigutuse muutumisel on vaja ringi kompileerida;
- seda skeemi kasutasid MS DOSi .com programmid.

Aadresside sidumine

2) Laadimise ajal:

- genereeritakse mälus ümberpaigutatav kood;
- reaalsed mäluaadressid asendatakse programmi laadimisel.

Aadresside sidumine

3) Täitmise ajal:

- aadresside sidumine lükatakse edasi konkreetse aadressi poole pöördumiseni;
- programmi saab täitmise ajal mälus ringi liigutada;
- vajalik riistvaraline tugi.

Loogilised ja füüsilised aadressid

- **Loogiline aadress** - programmide poolt kasutatav aadressiruum, nimetatakse ka virtuaalseks aadressiruumiks
- **Füüsiline aadress** – loogilisele aadressile vastav mälupesade aadress, mida mäluseade tegelikult näeb ja kasutab
- Kompileerimise ja laadimise ajal seotavate aadresside puhul on loogiline ja füüsiline aadress samad, täitmise ajal seotavatel erinevad

Mäluhaldusplokk

- MMU (*Memory-Management Unit*) ehk mäluhaldusplokk - riistvaraline seade loogiliste aadresside füüsiliseks teisendamiseks.
- Kasutajaprogramm tegeleb oma loogiliste aadressidega (0..max) ega näe otseselt füüsilisi aadresse.

Mälukasutus

- **Dünaamiline laadimine** – protseduuri ei laeta mällu enne kui teda tegelikult vaja läheb.
- **Staatiline linkimine** – kasutatavad protseduurid kaasatakse käivitatavasse faili.
- **Dünaamiline linkimine** – käivitatavas failis hoitakse vaid infot protseduuri asukoha kohta.
- **Ülekatmine**

Dünaamiline laadimine

- Alamprogramme ei laadita enne, kui neid kord vaja läheb;
- Efektiivsem mälu kasutus;
- Kasulik näiteks siis, kui harva esinevate erijuhtude töötlemiseks on kokku palju koodi;
- Ei vaja operatsioonisüsteemi tuge, realiseeritav täiesti programmi enda tasemel;

Staatiline linkimine

- Teatud operatsioonisüsteemid lubavad vaid seda.
- Süsteemsed teegid loetakse samaväärseks muude objektimoodulitega ja seotakse programmi binaarkoodi.

Dünaamiline linkimine

- Linkimine lükatakse edasi täitmisajaks
- Tegelik alamprogrammi asemel on viit (*stub*), mis esimesel tema poole pöördumisel laadib päris alamprogrammi ja asendab ennast selle alamprogrammiga (enamasti viida vahetusega)
- Operatsioonisüsteemi tuge on vaja juhul, kui tahame neid alamprogramme jagada mitme protsessi vahel ja kasutame seejuures mälukaitset
- Eriti kasulik (süsteemsete) teekide kasutamisel - kettaruumi ja mälu kokkuhoid, mugav teekide

Ülekatmine (*overlay*)

- Rakendusprogramm hoiab mälus ainult seda osa programmist ja andmetest, mida hetkel vaja on.
- Kasutatakse, kui kogu programm kokku on suurem kui talle antud mälu.
- Operatsioonisüsteemilt ei vajata tuge, realiseeritav kasutaja tasemel.
- Näide: DOS ja suured programmid.

Saalimine (*swapping*)

- Protsessi võib ajutiselt mälust välja salvestusruumi kirjutada ning hiljem tagasi mällu laadida
- Salvestusruum - kiire ketas, mis mahutab kõigi kasutajaprogrammide mälukujutised, peab olema otsejuurdepääsuga suvalise mälukujutise juurde
- *Roll out, roll in* - saalimise variant prioriteedil põhinevate planeerimisalgoritmide juurde: madalama prioriteediga protsess saalitakse välja kõrgema prioriteediga protsessi töö ajaks

Saalimine (*swapping*)

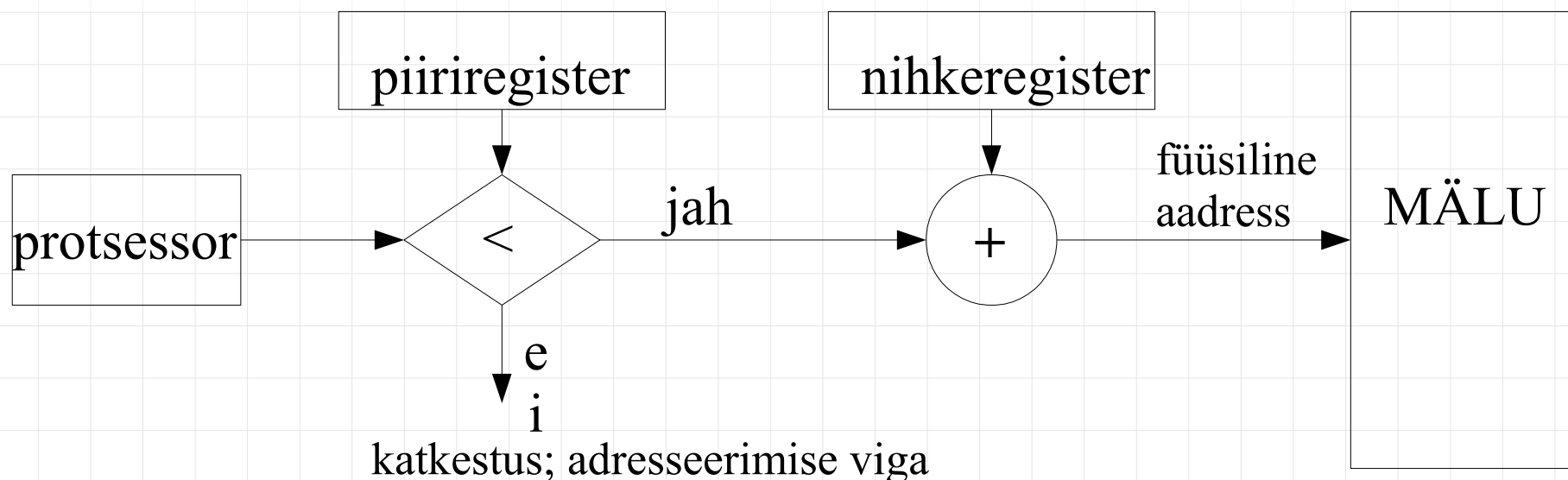
- Enamuse saalimise ajast võtab andmete kopeerimine välisseadmele, aeg on lineaarselt sõltuv mäluhulgast.
- Erineval kujul on saalimine kasutusel paljudes süsteemides, näiteks UNIX, Linux, Windows.

Pidevate mäluosalade hõivamine

- Põhimälu on üldjuhul jagatud kahte ossa:
 - operatsioonisüsteem (sageli mäluosalade eesosas katkestusvektorite asukoha tõttu);
 - kasutajaprotsessid.
- Kasutajaprotsessidele mõeldud mälu on omakorda jaotatud väiksemateks tükkideks
- Igale täidetavale protsessile eraldatakse üks sobiva suurusega terviklik mälu tükk
- Auk - vaba mäluosa

Pidevate mälu alade hõivamine

- Iga protsessiga on seotud nihke- ja piiriregister
 - Kasutatakse loogiliste aadresside teisendamisel
 - **Kaitseb** teisi protsesse ja opsüsteemi vigaste mälupöörduste eest



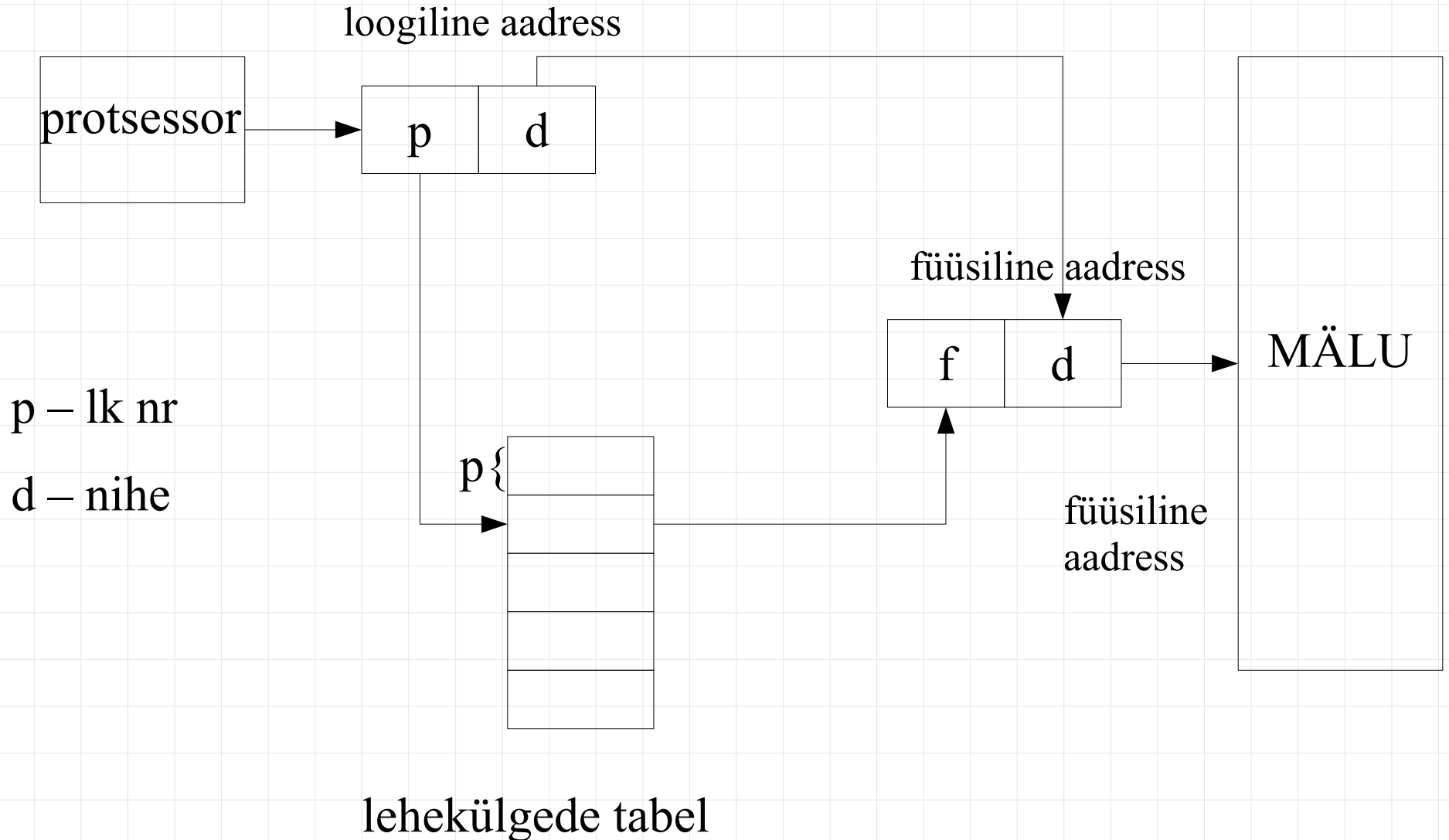
Fragmenteerumine

- Sisemine
 - Kui ruumi hõivamisel kasutatakse suuremaid blokke, siis ka väiksema ruumi vajadusel hõivatakse kogu blokk.
 - Vaba ruumi on, kuid ei saa kasutada.
- Välimine
 - Ruumi hõivamisel antakse sobival hulgal ruumi. Ruum ei pruugi asetseda järjestikku.
 - Vaba ruumi on, kuid see ei asu järjestikku ja kasutada ei saa

Lehekülgede saalimine (*paging*)

- Eeldus – protsessi hõivatud mälu alad ei pea asetsema füüsiliselt järjestikku.
- Kaader (*frame*) – pidev blokk füüsilist mälu suurus kahe aste (512, 4096, 8192)
- Lehekülg (*page*) – pidev blokk loogilist mälu
- Protsessi käivitamisel loetakse tema leheküljed vabadesse kaadritesse
- Iga aadress on protsessoris jagatud kaheks:
 - Lehekülje number (p) – indeks lehekülje tabelis
 - Nihe lehekülje sees (*page offset*) (d)

Lehekülgede tabel



Lehekülgede saalimine (*paging*)

- Peame arvet vabade kaadrite üle
- Suurusega n protsessi jaoks otsime lihtsalt n vaba kaadrit
- Iga protsessi kohta teeme lehekülgede tabeli loogiliste lehekülgede ja füüsiliste kaadrite vastavusse seadmiseks
- NB! Lehekülgede kaupa hõivamisel tekib paratamatult sisemine fragmenteerumine

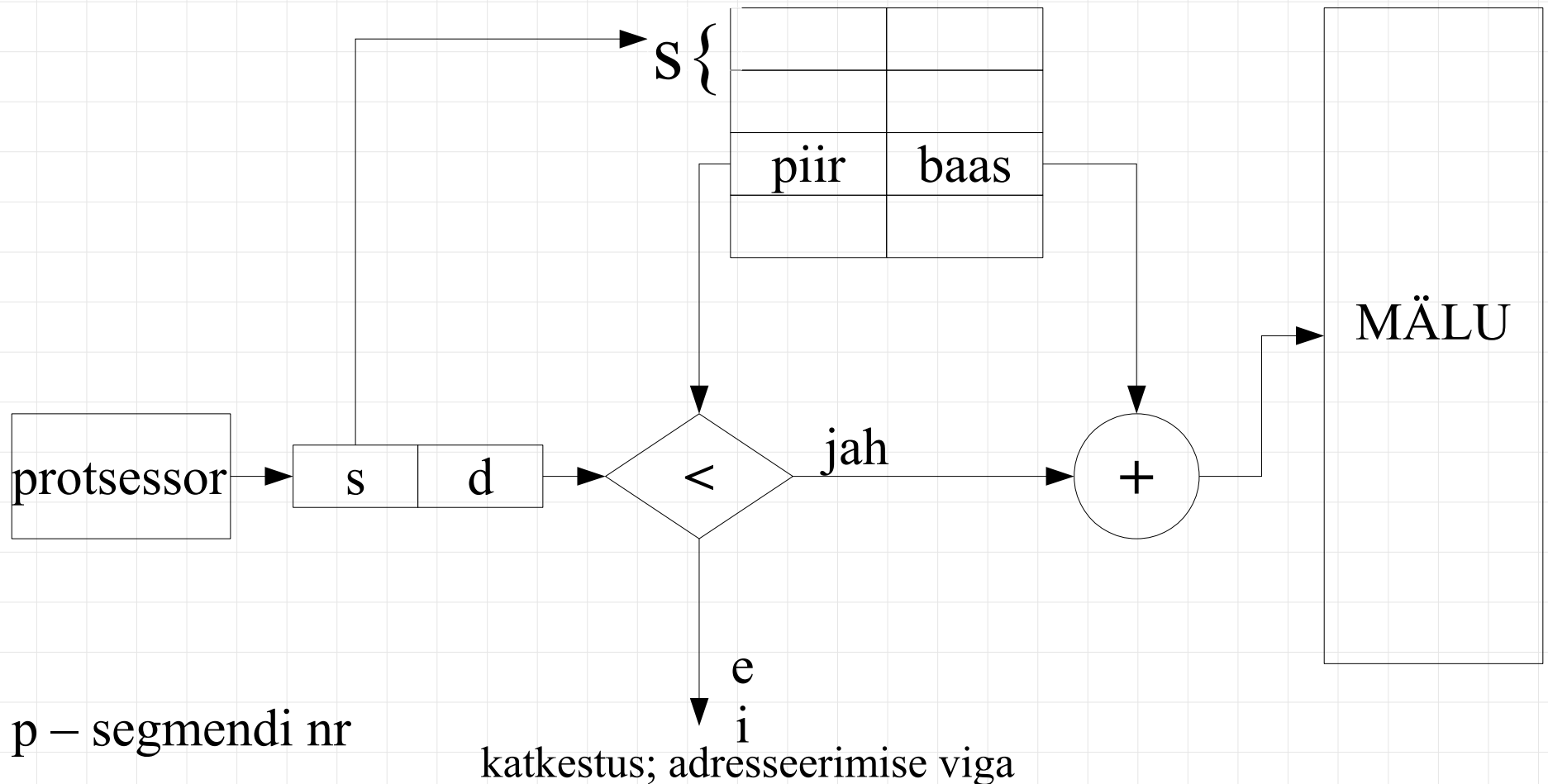
Segmenteerimine

- Segmenteerimine on mäluhalduse skeem, kus kasutatakse paljusid segmente, ühe pideva loogilise aadressruumi asemel
- Läheb paremini kokku paljude kasutajate mõttemaailmaga

Segmenteerimine

- Programm on komplekt segmente. Segment on loogiline ühik, näiteks
 - Põhiprogramm
 - Protseduur, funktsioon, meetod
 - Objekt
 - Lokaalsed muutujad, globaalsed muutujad
 - Magasin
 - ...

Aadressi moodustamine segmenteerimisega



p – segmendi nr

d – nihe

Virtuaalmälu

- Virtuaalmälu — kasutaja loogilise mälu füüsilisest mälust lahti sidumine:
 - ainult osa protsessi poolt vajatavast mälust on korruga füüsilises mälus;
 - loogiline aadressiruum saab seega olla oluliselt suurem kui füüsiline aadressiruum;
 - aadressirume (aadressiruumide osi) saavad protsessid omavahel jagada;
 - võimaldab efektiivsemalt protsesse luua .
- Virtuaalmälu saab realiseerida kahel viisil:
 - lehekülgede laadimine nõudmisel (*demand paging*);
 - segmentide laadimine nõudmisel (*demand segmentation*).

Virtuaalmälu praktikas

- Virtuaalmälu (VM) suurus
 - ~1,5 korda tegelik mälu (max 4GB)
 - NB! 32bit OS ei suuda adresseerida üle 4GB mälu!
 - SSD kõvaketastega arvutitel lülitada VM välja.
 - Kui arvutis on 3GB või rohkem aktiivmälu, lülita VM välja (kui ei kasutata).
- Linuxitel on selleks eraldi Swap partitsioon
- MS Windows
 - NT/2000/XP/2003 fail pagefile.sys
 - 9X Win386.swp

Virtuaalmälu seadistamine

- Suurus fikseeritud: max 1,5 korda tegelik mälu
 - Algsuurus=lõppsuurus
- Asukoht:
 - Eraldi kiire kõvaketas
 - Füüsiliselt kettal alguses oleval partitsioonil
- NTFS failisüsteem ja 4 kB klasteri suurus (olenevalt süsteemist)
- RAID0 sobib
 - Ei sobi: RAID1 ja RAID5 – virtuaalmälu ei vaja tõrkekindlust

NX bitt

- Riistvaraliselt realiseeritud
 - AMD 64-bitine protsessor
 - INTEL Itaniumi 64-bitine protsessor (*XD eXecute Disable*)
- NX – No eXecute – protsessor keelab programmil käivitada koodi väljaspool talle eraldatud mäluala.
- Tagab turvalisuse, takistab viiruste levikut (aitab puhvri ületäitumise vastu).

NX bitt

- Tarkvaraline lahendus:
 - Linux
 - Exec Shield
 - Pax
 - Windows
 - DEP (Data Execution Prevention)

Süsteemi vahemälu (*CPU cache*)

- Vahendajaks protsessori ja aeglase mälu vahel
 - NB! aeglane võrreldes protsessori kiirusega
 - Vanematel arvutitel 2 vahemälu nüüdisaegsetel rohkem

Kettapuhver (*disk cache*)

- Suurus keskmiselt 2MB kuni 8MB
- Seotud kindla seadmega (kõvaketas)
- Kasutusvaldkond:
 - ette/tahalugemine
 - Kiiruse tasakaalustamine
 - Kirjutamise kiirendamine
 - Käsujärjekorra hoidmine