

Praktikum 1

OPERATSIOONISÜSTEEM JA SELLE ÜLESEHITUS

Arvutisüsteem

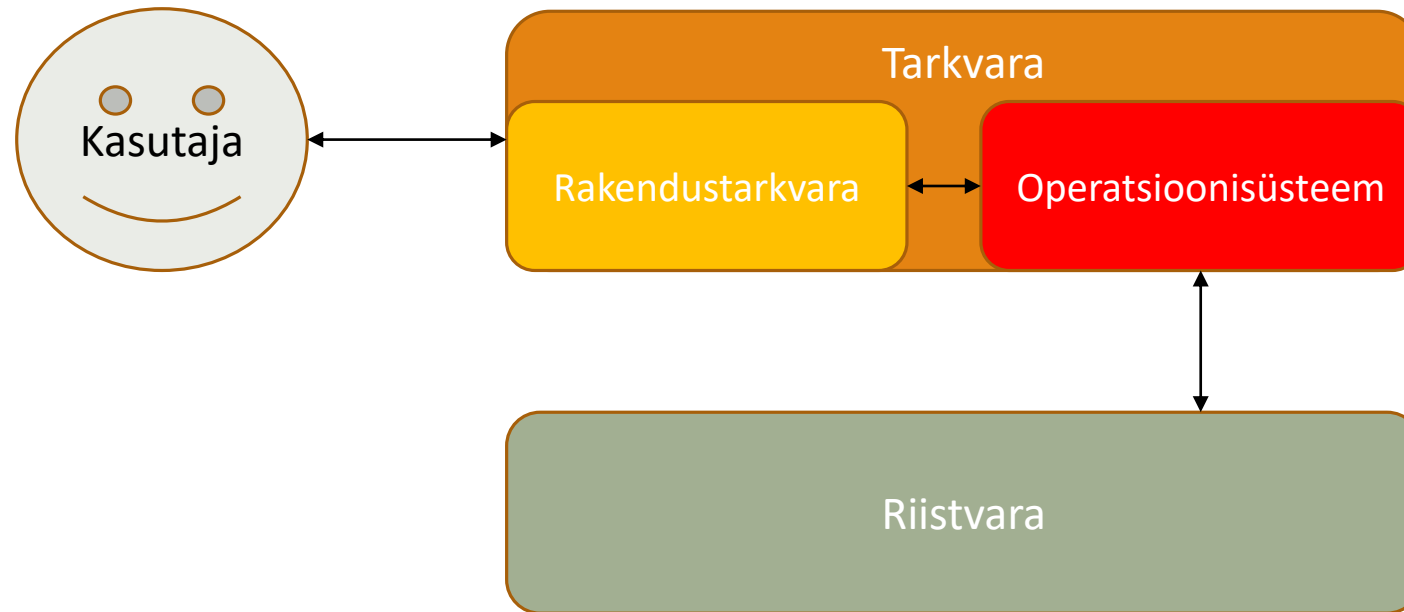
Mis moodustavad arvutisüsteemi?

- Riistvara
- Tarkvara
- Kasutaja(d)

Tarkvara saab laias laastus jagada kaheks

- Rakendustarkvara
- Operatsioonisüsteem

Arvutisüsteemi skeem



Operatsioonisüsteem

"Tarkvara, mis juhib programmide täitmist, nt Windows,,

- *ÕS 2006*

"Operatsioonisüsteem on tarkvara, mis töötab arvutisüsteemidel hallates riistvara ja pakkudes ühiseid teenuseid rakendustarkvara efektiivseks tööks.,,

- *Wikipedia*

"Tarkvara, mis kontrollib riistvararessursside jagamist ja kasutust"

-*Wiktionary*

Operatsioonisüsteemi ülesanded

- Protsessihaldus
- Mäluhaldus
- Katkestuste haldus
- Andmete haldus
- Süsteemi kaitse

Protsess

Mis on programm?

Kogum instruktsioone, mis täidavad mingi eesmärgi kui need arvutil käivitatakse. Programm on staatiline – on salvestatud faili (või kirjutatud paberile) ja ei tee iseenesest mitte midagi.

Kuna saab programmist protsess?

Programmist saab protsess, kui see käivitatakse. Protsess on aktiivne – on loetud mälu ja arvutisüsteem täidab järjest selle instruktsioone, mis võivad töödelda andmeid või juhtida seadmeid.

Sama programmi võib käivitada mitu korda – iga käivituse peale tekib eraldi uus protsess.

Protsessihaldus

Protsessihalduse ülesanded on:

- Protsessidele protsessoriaja jagamine
- Protsesside loomine ja kustutamine
- Protsesside peatamine ja taastamine
- Protsesside sünkroniseerimine
- Protsesside vahelise andmevahetuse tagamine

Mälu

Operatiivmälu

- Arvutisüsteemi põhimälu – tänapäeval mõõdetakse gigabaitides.
- Protsessor saab instruktsioone ja andmeid lugeda ja kirjutada ainult operatiivmälu vahendusel. Seega on väga oluline, et mällu oleks õigel ajal laetud õiged andmed.
- Operatiivmälu hulk on piiratud. Vajadus mälu järele on enamasti suurem kui süsteemis mälu on.
- Iga bait mälus on eraldi adresseeritav

Mäluhaldus

Mäluhalduse ülesanded on:

- Pidada arvet selle üle, millised osad mälust on parasjagu kasutuses ja kes neid kasutab
- Otsustada milline protsess ja/või andmed parasjagu mällu laadida või sealt hoopis välja visata.
- Protsessidele mälu eraldamine
- Protsessidele eraldatud mälu vabastamine

Katkestus

Katkestus on signaal arvutisüsteemis, mille peale on vaja protsessoril parasjagu töötav protsess peatada, et reageerida mingile sündmusele. Peatatud protsessi peab hiljem olema võimalik jätkata.

Katkestused jagunevad:

Riistvaralised - riistvara kontroller genereerib katkestuse andmete saatmiseks või teatamiseks andmete edukast vastuvõtmisest või hoopis mingi ajaperioodi möödumisest. Näiteks kasutaja liigutas hiirt, printer lõpetas printimise, süsteemi taktsageduse generaator genereeris uue impulsi.

Tarkvaralised – protsess genereerib süsteemikutse (*system call*) või lõksu (*trap*). Näiteks soovib protsess faili kirjutada (seda saab ta teha ainult operatsioonisüsteemi vahendusel) või toimub veasituatsioon (näiteks üritati jagada nulliga).

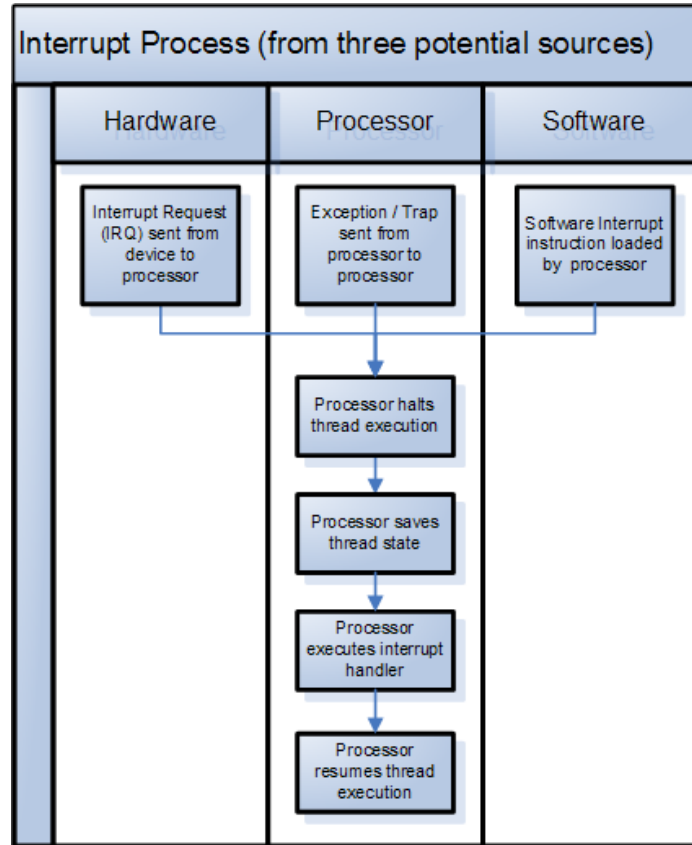
Katkestuste haldus

Katkestused on operatsioonisüsteemi töö oluline osa.

Operatsioonisüsteem peab oskama:

- katkestuse ilmnedes parajasti töös oleva protsessi peatada ja selle oleku salvestada
- leida katkestusele reageerimiseks sobiliku protseduuri
- käivitada katkestusele reageerimiseks sobiliku protseduuri.
- taastada katkestuse poolt peatatud protsessi oleku.

Katkestuse halduse näide



https://upload.wikimedia.org/wikipedia/commons/c/cf/Interrupt_Process.PNG

Andmed

Mis on andmed?

Andmed on kokkulepitud viisil vormindatud informatsiooniühikud.

Arvutisüsteemides salvestatakse andmeid kõige madalamal tasemel **kahendkoodis** nullide ja ühtedena.

Väikseim andmehulk on **bitt**, mis sisaldab ühte väärtust kahest võimalikust – nulli või ühte. Kaheksa bitti moodustab ühe **baidi**.

Peaaegu kõik operatsioonisüsteemid esitavad kasutajale andmeid failidena. **Fail** on ühe tervikuna käsitletav andmeüksus e. kogum ühtekuuluvaid andmeid, mis koos esitavad mingit tervikut – näiteks pilti.

See kuidas nullide ja ühtede jadast mingeid andmeid välja lugeda sõltub sellest, kuidas on kokkulepitud andmete salvestamine e. siis andmete salvestamise vormist.

Andmete haldus

Operatsioonisüsteemi ülesandeks oleva andmete halduse võib jagada kolmeks alamosaks:

- Failisüsteemi haldus (**Failisüsteem** – salvestusseadmel säilitatavate andmete kokkuleppeline korrastusviis.)
 - Failide loomine ja kustutamine
 - Failide organiseerimine (kataloogide struktuur)
 - Failide manipuleerimiseks vajaliku liidese pakkumine
- Salvestusseadmete haldus
 - Salvestusseadmete vaba ruumi haldus
 - Salvestusruumi eraldamine
 - Salvestusseadme andmevahetuse korraldamine
- Vahemälu haldus
 - Vahemälus olevate andmete sünkroniseerimine

Süsteemi kaitse

Operatsioonisüsteem peab hoolitsema, et protsessid saaksid kasutada ainult neile lubatud ressursse.

Näiteks:

- Mäluhaldur kontrollib, et protsess saaks adresseerida ainult talle määratud mäluosa (teatud eranditega).
- Süsteemi valvetaimer kontrollib, et protsess ei jääks lõpmatult protsessorit koormama.

Samuti on operatsioonisüsteemi ülesandeks kontrollida, et kasutajad saaks sooritada ainult neile lubatud tegevusi ning pääseks juurde vaid neile lubatud andmetele.

Näiteks:

- Arvutisüsteemi kasutamiseks peab kasutaja sisestama kasutajanime ja parooli.
- Operatsioonisüsteem kontrollib, kas sisseloginud kasutajal on õigus manipuleerida mingit faili või käivitada mingi programm.

Operatsioonisüsteemide tüübid

Operatsioonisüsteeme võib jagada mitmeti:

Vastavalt seadmele:

- Serverioperatsioonisüsteemid
- Töölauaoperatsioonisüsteemid
- Mobiilsete seadmete operatsioonisüsteemid

Vastavalt operatsioonisüsteemi ülesehitusele ja eripärale:

- Single- or multitasking – üksik- või mitmiktegunmtöötusega operatsioonisüsteem
- Single- or multiuser – üksikasutuslik või ühiskasutuslik operatsioonisüsteem
- Real Time Operating System – reaalaajaoperatsioonisüsteem
- Embedded – manusoperatsioonisüsteem

Serverioperatsioonisüsteemid

Tunnused:

- Optimeeritud teenustele
- Äritarbijale suunatud riistvara ja tehnoloogiate tugi
- Põhirõhk stabiilsusel ja usaldusväärsusel

Näiteid:

- Windows Server 2022
- Rocky Linux 9
- Suse Linux Enterprise Server 15
- RedHat Enterprise Linux 9

Töölauaoperatsioonisüsteemid

Tunnused:

- Optimeeritud rakendustarkvarale
- Suunatud tavatarbija riistvarale (võimalikult lai riistvara tugi)
- Põhirõhk kasutajaliidesel (kasutamiskihtsusel, atraktiivsusel)

Näited:

- Windows 11
- OS X
- Ubuntu

Mobiilsete seadmete operatsioonisüsteemid

Tunnused:

- Optimeeritud vähesele ressursinõudlikusele
- Spetsialiseeritud riistvara tugi
- Spetsialiseeritud kasutajaliides

Näited:

- Android
- iOS
- Fuchsia

Üksikasutuslikud vs. Ühiskasutuslikud operatsioonisüsteemid

Üksikasutuslik operatsioonisüsteem = operatsioonisüsteemi saab korraga kasutada ainult üks kasutaja.

Ühiskasutuslik operatsioonisüsteem = operatsioonisüsteemi saab korraga kasutada mitu kasutajat enamasti toimub see üle võrgu.

Üksikasutuslikke operatsioonisüsteeme:

- MS-DOS
- Palm OS

Ühiskasutuslikke operatsioonisüsteeme:

- Praktiliselt kõik UNIX-i laadsed operatsioonisüsteemid
- Kõik uuemad Windowsi perekonna operatsioonisüsteemid

Reaalaja-operatsioonisüsteem

Tunnused:

- Optimeeritud ajakriitiliste protsesside täitmiseks
- Sageli spetsialiseeritud kindla seadme jaoks

Näited:

- Wind River VxWorks
- FreeRTOS

Manusoperatsioonisüsteem

Tunnused:

- Täidavad kindlat spetsiifilist rolli, osana suuremast süsteemist
- Sageli väga piiratud funktsionaalsusega - võib toetada vaid ühe programmi jooksutamist.
- NB! Sageli võivad manusoperatsioonisüsteemid olla ka reaalaja-operatsioonisüsteemid

Näited:

- NetBSD
- watchOS

Üksiktegumtöötlus vs. Mitmiktegumtöötlus

Üksiktegumtöötlus = korraga töötab ainult üks protsess kuni oma töö lõpuni või veasituatsioonini.

Mitmiktegumtöötlus = korraga saavad töötada mitu protsessi samaaegselt (eeldab mitut protsessorituumat) või vaheldumisi.

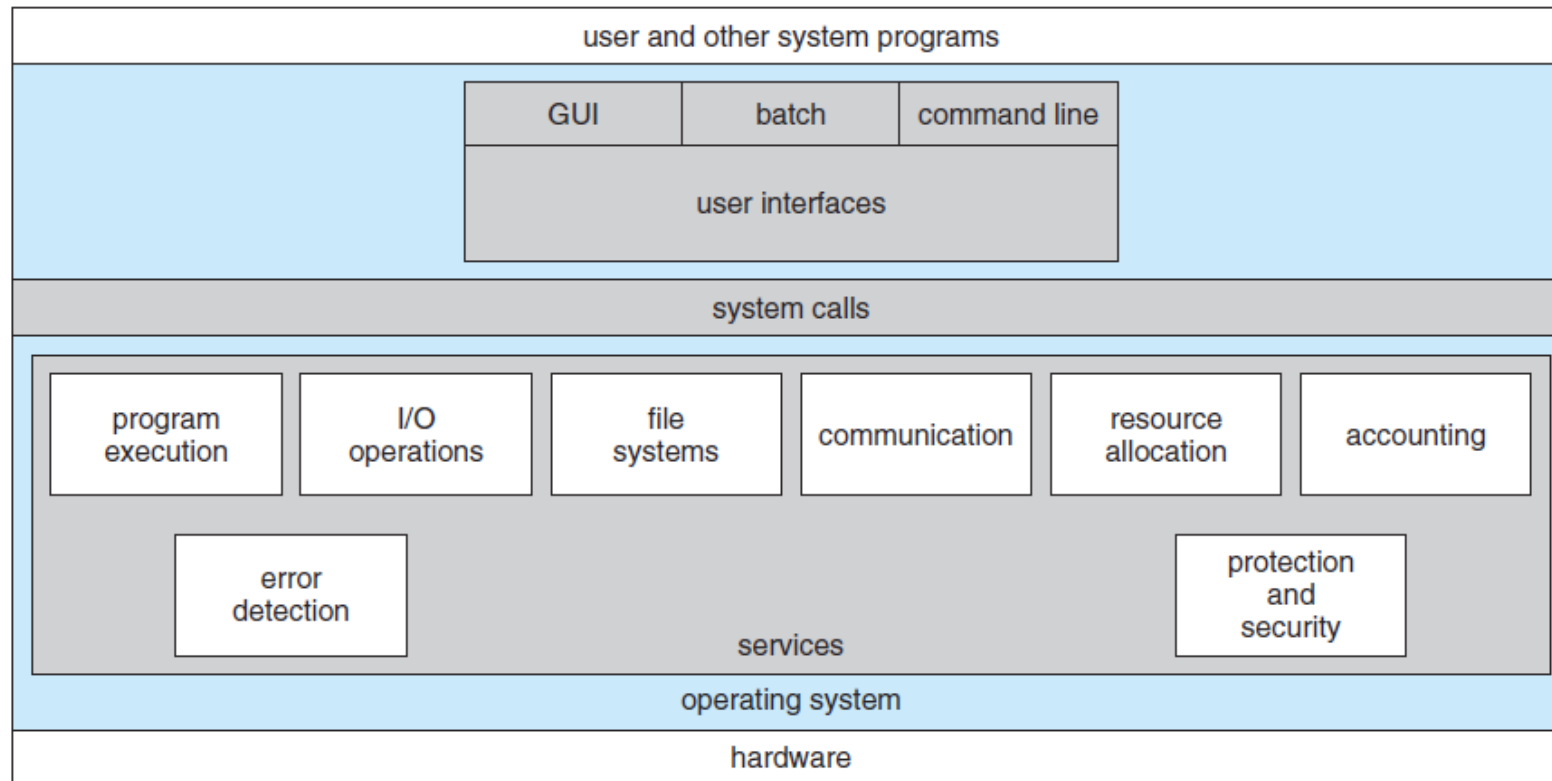
Üksiktegumtöötlusega operatsioonisüsteeme:

- Commodore BASIC 2.0
- MS-DOS (versioon 4.0 eelne)

Mitmiktegumtöötlusega operatsioonisüsteeme:

- Praktiliselt kõik tänapäevased operatsioonisüsteemid

Operatsioonisüsteemi ülesehitus pakutavate teenuste vaates



Allikas: Abraham Silberschatz, Greg Gagne, and Peter Baer Galvin, "Operating System Concepts, Ninth Edition"

OS teenused: kasutajaliides

Operatsioonisüsteem peab pakkuma kasutajale mingit viisi sellega suhtlemiseks. Kõigil operatsioonisüsteemidel on olemas üks või mitu järgnevaist kasutajaliidestest:

Graafiline kasutajaliides (GUI)

- Kasutaja suhtleb operatsioonisüsteemiga manipuleerides graafilisi objekte, mis esindavad arvutisüsteemi osasid (seadmeid, faile, käivitatud programme)
- Lihtne õppida, kuid teatud tegevused võivad olla aeganõudvad.

Käsurida (CLI)

- Kasutaja suhtleb operatsioonisüsteemiga andes käsked teksti kujul kindlaksmääratud süntaksi alusel.
- Kasutama õppimine võtab aega, kuid võimaldab lihtsalt tegevusi automatiseerida.

Pakktööde liides (batch)

- Kasutaja edastab käsud ja andmed operatsioonisüsteemile ettemääratud vormis ühe korraga (tavaliselt faili kujul - skriptifail).

OS teenused: programmide käivitamine

Operatsioonisüsteem peab olema võimeline programmi mällu laadima ja seda jooksutama. Programm peab lõppema, kas ettenähtud viisil või siis veateatega.

OS teenused: sisend/väljund operatsioonid

Töötav programm võib mingil hetkel nõuda sisend/väljund operatsiooni läbiviimist – näiteks faili lugemist kõvakettalt ja selle kirjutamist optilisele seadmele. Või hoopis mingi kujutise kuvamist ekraanile.

Sedalaadi tegevused toimuvad operatsioonisüsteemi vahendusel. Peamised põhjused selleks on:

Turvalisus – rakendusprogrammidel ei lubata üldjuhul sisend/väljundseadmeid otse kasutada, sest on võimatu kontrollida, kas seda tehakse korrektselt või mitte.

Efektisus – isegi kui rakendusprogramm oskaks seadet kasutada turvaliselt ei pruugi see tähendada, et seda tehakse efektiivselt. Operatsioonisüsteemi üks põhiülesandeid on aga olemasoleva riistvara võimalikult efektiivne kasutamine.

OS teenused: failisüsteemi operatsioonid

Erinevad operatsioonisüsteemid kasutavad erinevaid failisüsteeme – sageli mitmeid korraga. Kuna failidega tehtavad tegevused on sagedased, siis on tegemist süsteemi kiireks ja vigadeta toimimiseks väga oluliste operatsioonidega.

Failisüsteemi juures on oluliseks veel mõnede failisüsteemide poolt toetatud õiguste kontroll, mis võimaldab seadistada kasutajapõhist juurdepääsu.

OS teenused: protsesside vaheline kommunikatsioon

Kuigi üldjuhul on protsessid üksteisest eraldatud, leidub olukordi, kus on vajalik, et protsessid saaksid omavahel andmeid vahetada. Tegemist on keerulise probleemiga, kuna operatsioonisüsteem peab tagama protsessi puutumatus ja siiski edastama informatsiooni.

OS teenused: veatuvastus

Operatsioonisüsteem peab olema võimeline reageerima veasituatsioonidele.

Vead võivad olla **riistvaralised** (riistvarakontroller tagastab veakoodi) või **tarkvaralised/loogilised** (jagamine nulliga, protsess üritab lugeda mäluaadressilt, mis talle ei kuulu vms.).

Sellised olukorrad on juba oma olemuselt ettearvamatud ja seetõttu ei ole võimalik prognoosida, kuna viga aset leiab. Ainus, mida teha saab, on omada plaani, mida vea korral ette võtta.

Plaan peab olema iga veatüübi jaoks – paljudest vigadest suudab süsteem taastuda, kuid leidub ka vigu mille puhul on ainsaks lahenduseks taaskäivitus.

OS teenused: ressursside jaotus

Enamik tänapäevaseid operatsioonisüsteeme toetavad **mitmiktegunumtöötlust** ja **ühiskasutust**. See tähendab, et operatsioonisüsteem peab samaaegselt tegelema paljude protsessidega, mis võivad kuuluda erinevatele kasutajatele.

Selle eesmärgi saavutamiseks kuuluvad operatsioonisüsteemi põhikomponentide hulka mitmed alamsüsteemid, mis tegelevad ressursside jaotamise ja kasutuse planeerimisega.

Neid alamsüsteeme nimetatakse **plaanuriteks** (scheduler). Näiteks on olemas protsessori plaanur ja mäluplaanur samuti võib olenevalt operatsioonisüsteemist olla ühine plaanur sisend- ja väljundseadmete jaoks või igale seadmetüübile eraldi.

OS teenused: ressursside arvestus

Üks põhjus, miks operatsioonisüsteemid üldse tekkisid oli vajadus pidada arvet kasutatavate ressursside üle.

Alamsüsteem, mis peab arvet ressursside ja nende kasutuse üle on operatsioonisüsteemides olemas siiani, ning selle töö tulemust kasutatakse nii klientidele arvete esitamiseks (näiteks serverirendi teenus) kui süsteemis töötavate rakenduste töö optimeerimiseks.

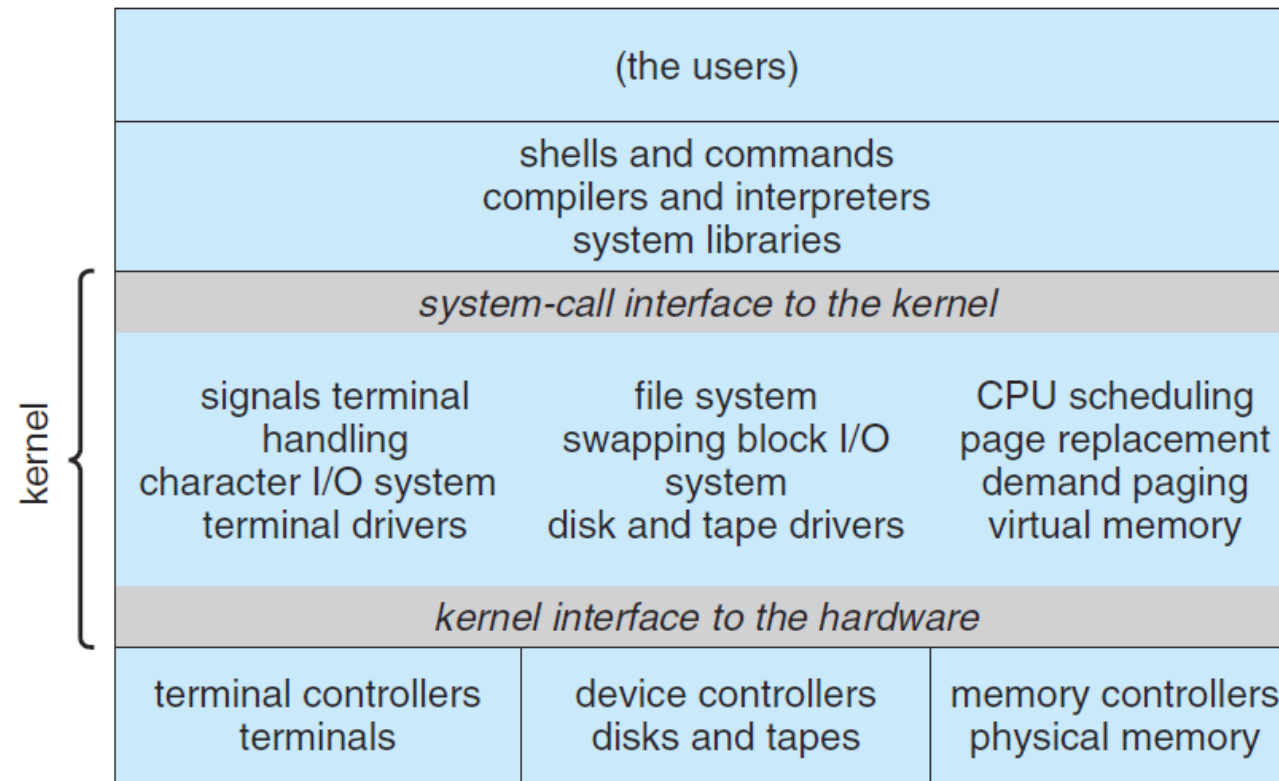
OS teenused: turvalisus ja kaitse

Operatsioonisüsteem peab tagama töötavate protsesside puutumatuse ning vahendid töödeldavate andmete kättesaadavaks tegemiseks ainult selleks volitatud isikutele.

Selle ülesande esimene pool on suuresti seotud ressursside kasutamise ja jagamise loogikaga – st. süsteem kontrollib oma protsesse, et need kasutaks ainult neile välja jagatud ressursse (mäluosa kaitse, sisend/väljund seadmete juurdepääs).

Ülesande teine pool eeldab, et süsteem toetab vahendeid (kasutajakontod, privilegieeritud tegevused, failisüsteemi juurdepääsuõigused), mille abil kasutaja saab seadistada ligipääsu süsteemile endale ja töödeldavatele andmetele.

Operatsioonisüsteemi struktuur (UNIX-i laadne)



Allikas: Abraham Silberschatz, Greg Gagne, and Peter Baer Galvin, "Operating System Concepts, Ninth Edition"

Tuum (kernel) ja süsteemikutse (system call)

Tuum on operatsioonisüsteemi keskne osa, mis sisaldab endas operatsioonisüsteemi põhifunktsioone. Tuum on operatsioonisüsteemi kõige kaitstum osa – töötab eraldi talle reserveeritud mälupiirkonnas ning suhtleb täiste operatsioonisüsteemi osadega ja rakendustarkvaraga ainult läbi rangelt reglementeeritud rakendusliidese.

Süsteemikutse on mehhanism, mille abil saavad rakendusprogrammid ja tuuma välised operatsioonisüsteemi osad kasutada operatsioonisüsteemi põhifunktsioone. Erinevatel operatsioonisüsteemidel on erinevad süsteemikutsed, kuid sageli täidavad nad väga sarnaseid funktsioone. Näiteks on igal operatsioonisüsteemil süsteemikutse failist andmete lugemiseks.

Süsteemikutsete tüüpe ja näiteid

Protsessi kontroll

- protsessi loomine (Windows: CreateProcess(); UNIX: fork())
- Protsessi lõpetamine (Windows: ExitProcess(); UNIX: exit())

Failihaldus

- Faili avamine või uue faili loomine (Windows: CreateFile(); UNIX: open())
- Faili sulgemine (Windows: CloseHandle(); UNIX: close())

Seadmete haldus

- Konsoolile kirjutamine (Windows: WriteConsole(); UNIX: write())

Informatsiooni haldus

- Protsessi ID küsimine (Windows: GetCurrentProcessID(); UNIX: getpid())

Kommunikatsioon

- Faili sisu laadimine põhimällu (Windows: MapViewOfFile(); UNIX: mmap())

Kaitse

- Faili juurdepääsuõiguste seadmine (Windows: SetFileSecurity(); UNIX: chmod())

Operatsioonisüsteemide ajalugu

Esimesed arvutid täitsid spetsiifilisi ülesandeid ega vajanud operatsioonisüsteemi.

Ka esimestel üldeesmärgilistel (*general purpose*) arvutitel puudus operatsioonisüsteem – kasutaja kirjutatud programm pidi ise kõigega hakkama saama.

Vajadus operatsioonisüsteemi järele tekkis sest:

- Arvutisüsteemid ja lisaseadmed muutusid järjest keerulikumaks
- Arvutisüsteemid olid kallid ja nende tööaega oli tarvis maksimaalselt ära kasutada
- Ressursside kasutuse üle oli tarvis arvet pidada

Operatsioonisüsteemide ajalugu

Seda, milline on päris esimene operatsioonisüsteem on väga raske öelda.

Sageli tuuakse välja MIT suurarvutil **Whirlwind** 1954.a. (mõnedel andmetel 1955) kasutusele võetud **Tape Director** süsteemi.

Üheks esimeseks levinud operatsioonisüsteemiks loetakse **General Motors'i GM-NAA I/O** süsteemi.

Loodud aastal 1956 (mõnedel andmetel 1955)

Töötas suurarvutil (main-frame) IBM 704

Peamiseks ülesandeks järgmise programmi käivitamine (pakktöötlus)

Pakkus sisend/väljund seadmetega suhtlemiseks liidest

Sellel põhinevad paljud hilisemad suurarvutite operatsioonisüsteemid.

Operatsioonisüsteemide ajalugu

Timesharing Director

- Loodud aastal 1964
- English Electric KDF9 jaoks
- Üks esimesi riistvara toega mitmiktegmistööstlust toetavaid süsteeme

Multics

- Arendatud vahemikus 1965 – 1969
- On mõjutanud praktiliselt kõiki kaasaegseid operatsioonisüsteeme
- Uued mäluhalduse põhimõtted: ühetasemeline andmehaldus (virtuaalmälu eellane), dünaamiline linkimine
- Oluliselt täiendatud operatsioonisüsteemi sisesed turvameetmed – prioriteetide süsteem.
- Esimene operatsioonisüsteem, millel oli hierarhiline failisüsteem.

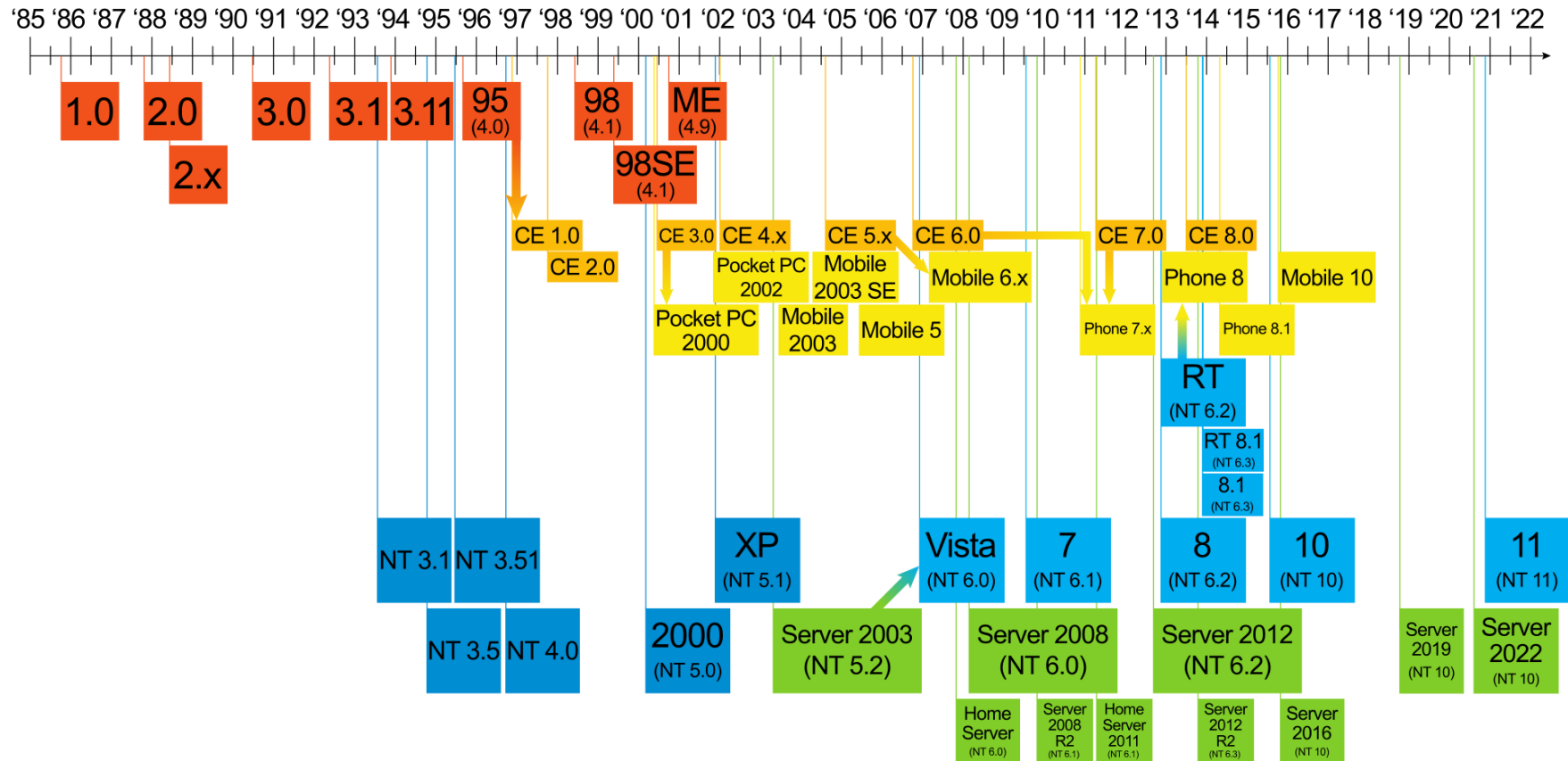
Unics (UNIX) 1971.a.

Oluline eelkõige tulevaste operatsioonisüsteemide mõjutajana ja standardite kujundajana.

Vt. skeemi: https://en.wikipedia.org/wiki/File:Unix_history-simple.svg

POSIX standardid – eesmärk määrata kuidas programmid ja operatsioonisüsteemid üksteisega ühilduvad.

Windows 1985.a.



https://upload.wikimedia.org/wikipedia/commons/6/6d/Windows_Updated_Family_Tree.png

Linux 1991.a

Loodud Linus Torvaldsi poolt

- Huvi operatsioonisüsteemide vastu
- MINIX-i piirav litsents

Linux muutus populaarseks esmalt serverite operatsioonisüsteemina kui suured organisatsioonid nagu näiteks NASA hakkasid oma suurarvuteid vahetama välja serveriklastrite vastu.

Linux toetab suurt hulka erinevaid riistvaraplatvorme ja on seetõttu tänapäeval kasutuses kõikjal pisiarvutitest kuni superarvutiteni.

OS X (uue nimega macOS) 2001.a.

