



IT KOLLEDŽ
TALLINNA TEHNIKAÜLIKOOL

Operatsioonisüsteem

Operatsioonisüsteemid ja nende haldamine ICA0001

Edmund Laugasson

edmund.laugasson@taltech.ee

https://wiki.itcollege.ee/index.php/User:Edmund#eesti_keeles

Käesoleva dokumendi paljundamine, edasiandmine ja/või muutmine on sätestatud ühega järgnevatest litsentsidest kasutaja valikul:

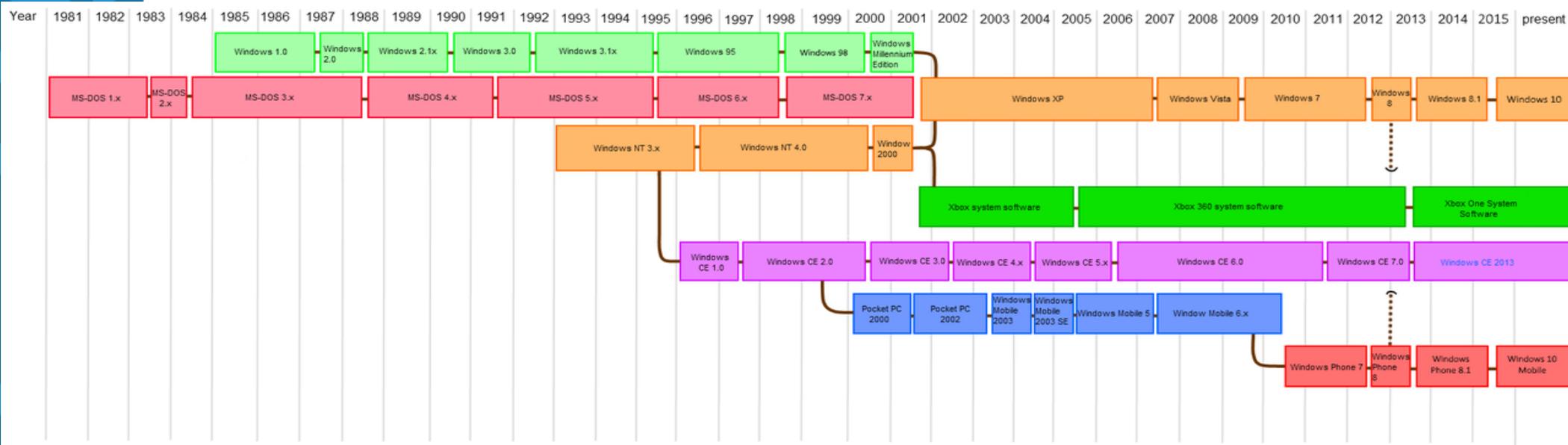
* GNU Vaba Dokumentatsiooni Litsentsi versioon 1.2 või uuem

* Creative Commons'i Autorile viitamine + Jagamine samadel tingimustel 4.0 litsents (CC BY-SA)

Operatsioonisüsteemide ajalugu

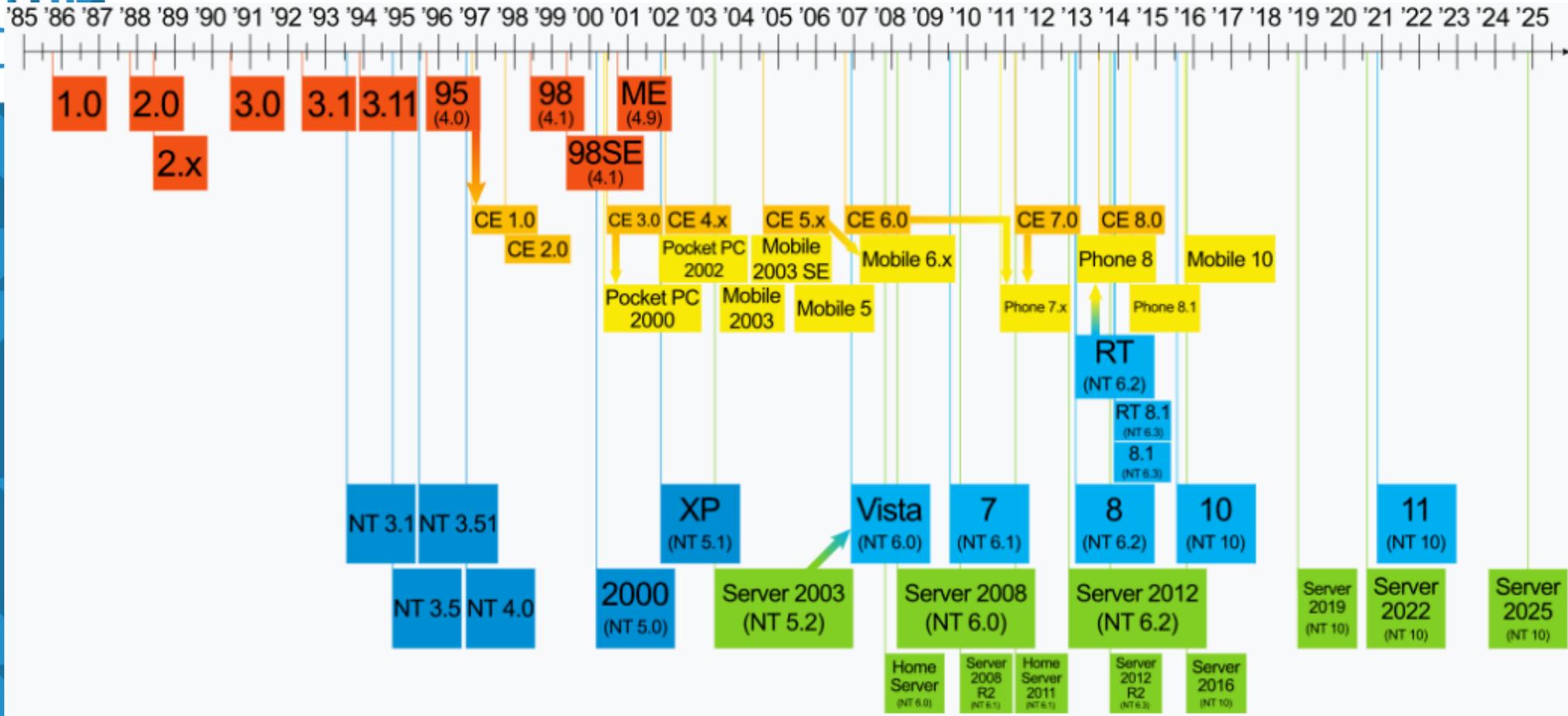
- prantsuse programmeerija Éric Lévénez <https://www.levenez.com/>
 - UNIXilaadsed OSid
 - Microsofti OSid
 - programmeerimiskeeled
- kokkuvõtlikumalt leiab muidugi ka Wikipediast

Microsofti OSid



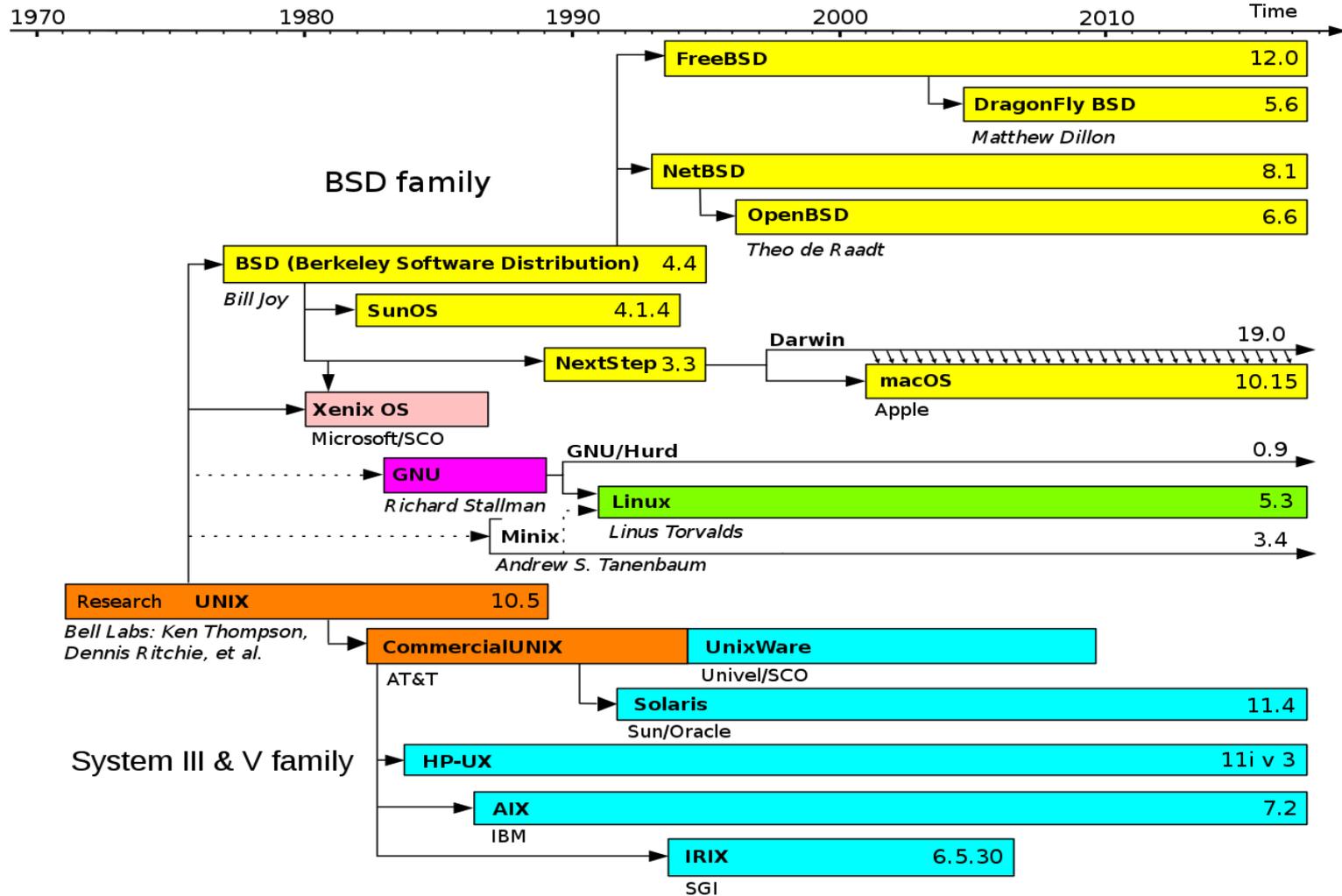
https://en.wikipedia.org/wiki/List_of_Microsoft_operating_systems

Microsoft Windows

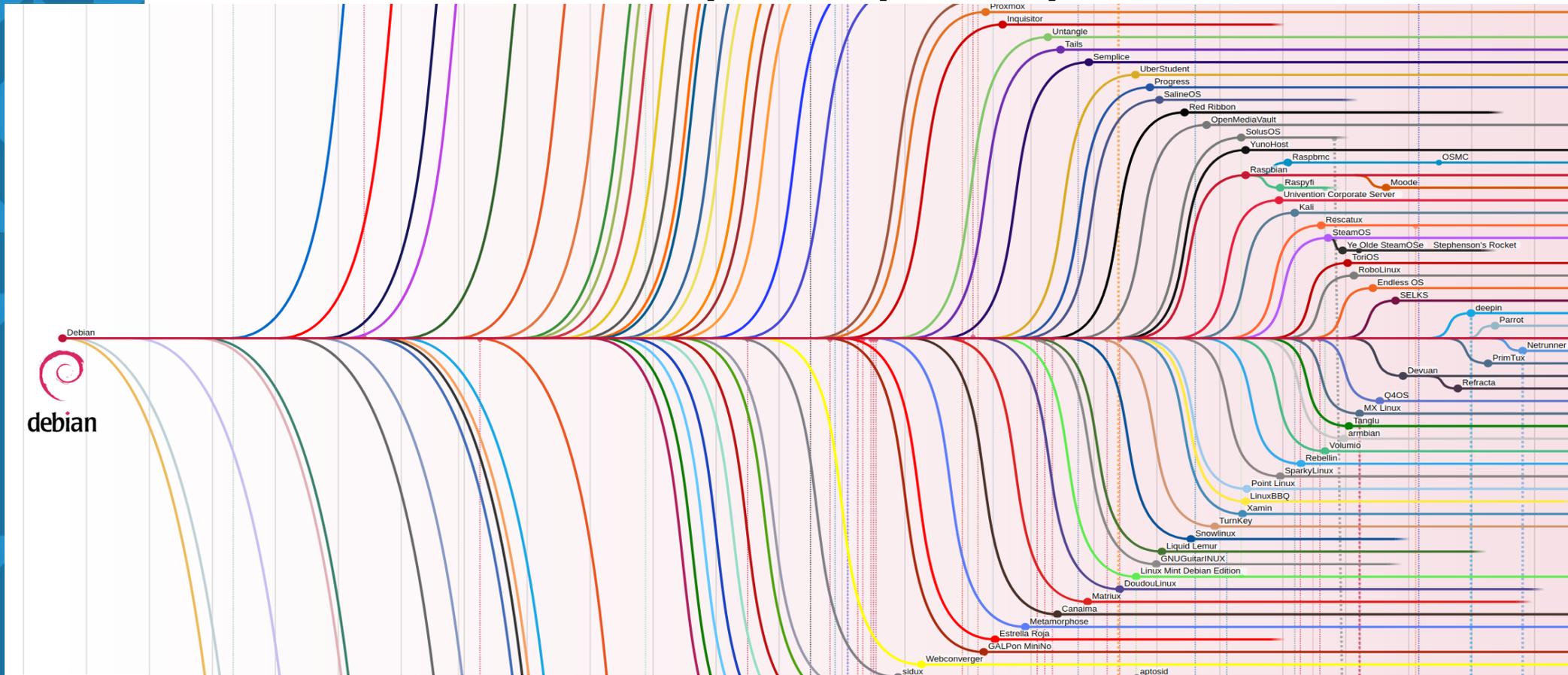


https://en.wikipedia.org/wiki/Microsoft_Windows

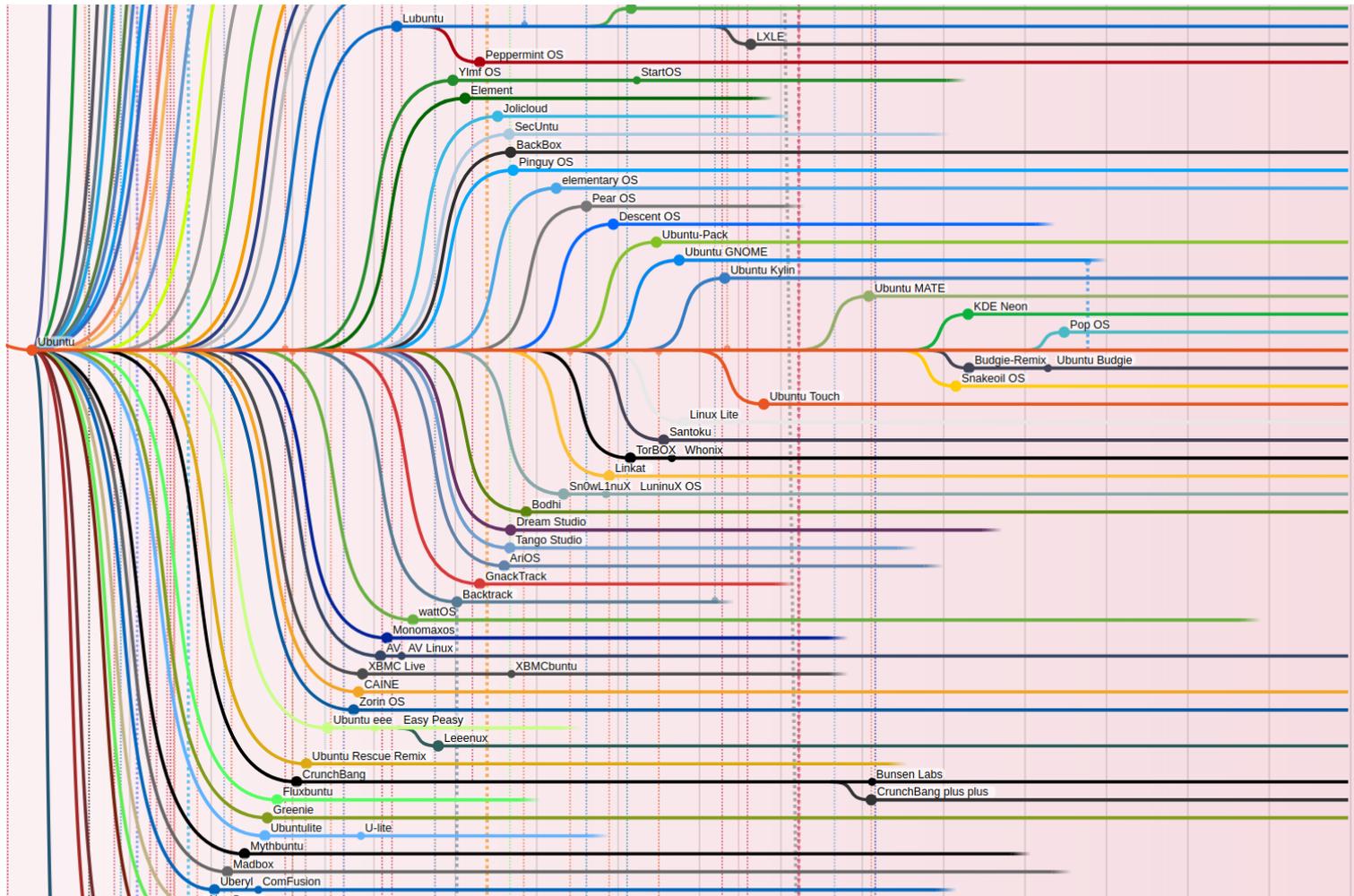
UNIXlaadsed OSid



GNU/Linux tarkvarakomplektid (distrod)



GNU/Linux tarkvarakomplektid (distrod)



https://en.wikipedia.org/wiki/Linux_distribution

<https://wiki.pingviin.org/Distributsioon>



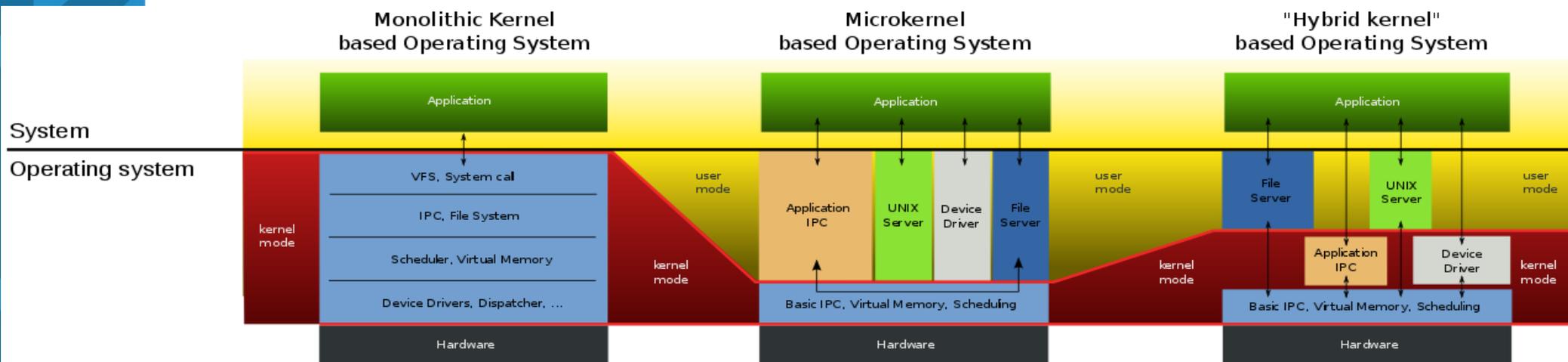
Arvutisüsteem



Ikoonid ja logod:
<https://www.iconfinder.com/>
<http://findicons.com/>
<http://logos.wikia.com/>
programmide kodulehed

Operatsioonisüsteem

- Operatsioonisüsteem on tähtsaim süsteemitarkvara hulka kuuluv programm, mis haldab arvutisüsteemi ressursse ja pakub kasutajale kasutajaliidese teenust ning programmeerijale rakendusprogrammide loomise liidest (API)



https://en.wikibooks.org/wiki/Operating_System_Design/Kernel_Architecture

https://en.wikipedia.org/wiki/User_space

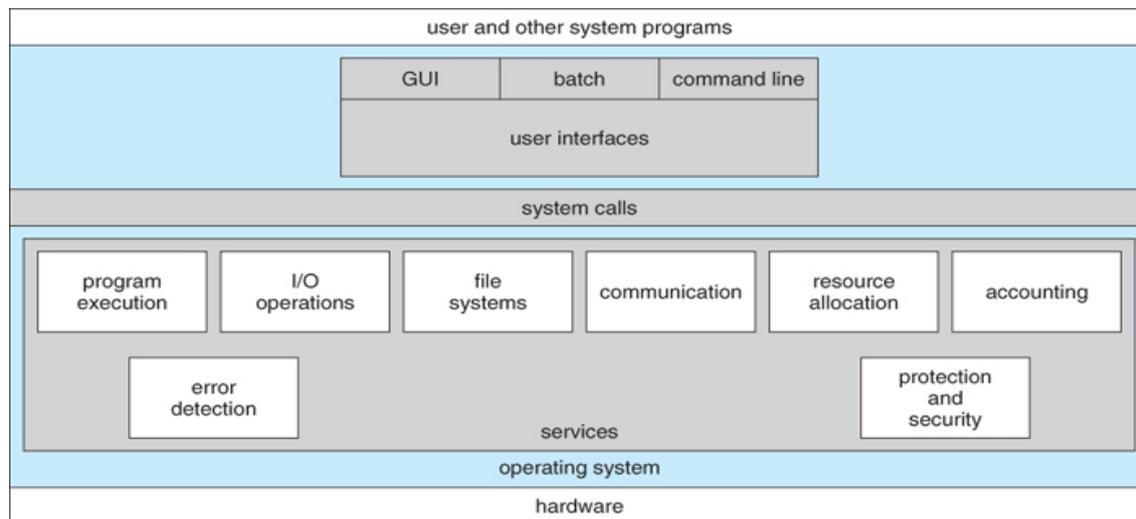
IPC Inter-process Communications

VFS Virtual File System

Operatsioonisüsteem

kasutajaliidesed:

- CLI: bash, sh, csh, jt
- GUI: Windows; KDE, GNOME jt (*UNIX-like*), 1970-ndatel Xerox, vt X11 (aknahaldus), töölauakeskkond



- *System Call* – kasutaja või rakendus kutsub välja süsteemse teenuse, enamasti kasutatud C/C++ (ka *assembler*). Võimalus kasutada näiteks *strace* nende jälgimiseks.
- I/O – sisend/väljundoperatsioonid seadmetega suhtlemisel, nt klaviatuur, hiir, salvestusseadmed, jne
- rakenduste käivitamine: laadimine RAMi, töötamine, sulgemine (viisakalt, jõuga)

Miks muutus OS vajalikuks?

- Esimesed arvutid operatsioonisüsteemi ei omanud;
- Programmid olid seotud konkreetse seadmega ja nende teisaldamine oli keerukas;
- Erinevate süsteemide integreerimine oli väga piiratud ja mahukas;
- Kui tekkis vajadus teenindada mitut programmi või mitut kasutajat, siis oli vaja ühte programmi, mis teiste üle arvet peaks.

Kaasaeg

- Operatsioonisüsteemid on kasutusel igal pool;
- Operatsioonisüsteemid töötavad nii pisiseadmetes kui pilvarvutite serveriruumides;
- Seadmete tootjatel on lihtsam kasutada seadmele sobivat operatsioonisüsteemi, kui luua kõik ise uuesti;
- Turvalisus on palju olulisem kui varem;
- Mitu operatsioonisüsteemi on sinu taskus/kotis?
 - nutiseade (telefon, aktiivsuspäevik, kell, ...)
 - süle-, tahvelarvuti
 - ...

Operatsioonisüsteemi ülesanded

- Arvuti riistvararessursside haldamine
 - CPU aja haldamine
 - Mälu haldamine (RAM, virtuaalmälu, välismälu)
 - Sisend/väljundseadmete haldamine
 - Võrgu haldamine
- Arvutisüsteemi juhtimine
 - Programmide juhtimine
 - Sisend/väljundseadmete juhtimine
 - Pääsukontroll ja kasutaja tuvastamine
 - Programmidevahelise andmevahetuse juhtimine (ka võrgus)

Operatsioonisüsteemi osad

- Tuum (**kernel**) on OSi kohustuslik osa
- Kasutajaliides võib olla OSi osa
 - *CLI* käsurea liides - *command line interface*
 - *GUI* graafiline kasutajaliides - *graphical user interface*

Käsuinterpretaator (CLI)

- CLI – *Command Line Interface* ehk käsuinterpretaator
 - *Bash shell, Bourne shell, cmd, PowerShell, zsh* jne
 - Süsteemiadministraator peab selles oskama:
 - Käske sisestada
 - Ühe käsu väljundit teise sisendisse suunata
 - Sisendit ja väljundit oskama suunata/võtta failist või teisest programmist
 - **Programmeerima** ja automatiseerima oma tegevusi

Operatsioonisüsteemi tuum

- Rakendusprogrammid pöörduvad arvutisüsteemi poole kasutades operatsioonisüsteemi tuuma (*kernel*) teenuseid



Tuum (*kernel*)

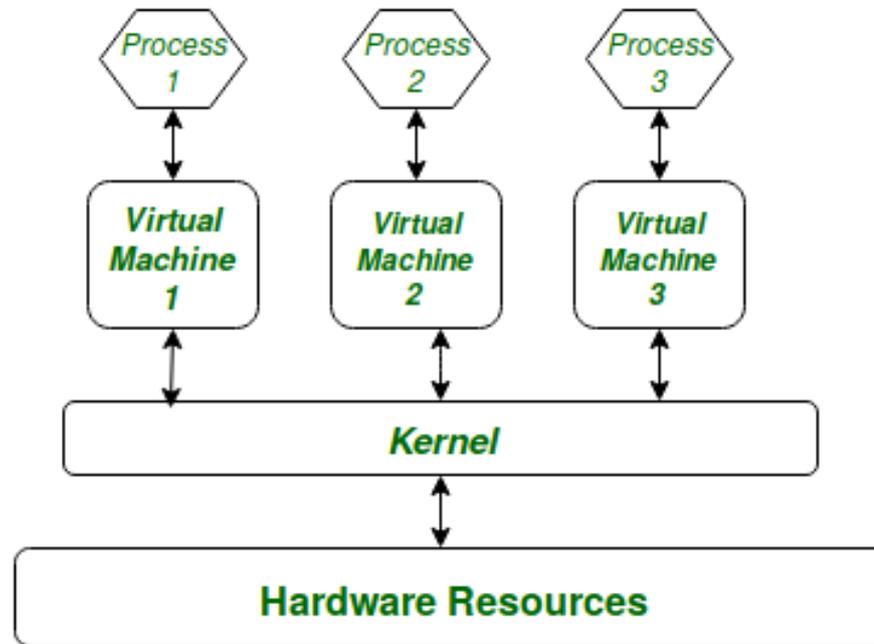
- CPU aja planeerimine
 - Reaalaja vs pakktöötlus
 - Sünkroniseerimine
 - Skaleeruvus mitme CPU-tuuma korral
- Mälu haldamine
 - MMU (*Memory Management Unit*), peamiselt virtuaalmälu haldus
 - NUMA (*Non-Uniform Memory Architecture*), mitme CPU tuuma korral andmete kättesaadavuse tagamine („nälgimise” vältimine) kuna RAM on aeglasem
 - Vahemälud
- I/O haldamine
 - Sünkroonne
 - Asünkroonne
- https://en.wikipedia.org/wiki/Comparison_of_operating_system_kernels
- <https://www.geeksforgeeks.org/kernel-in-operating-system/>

CPU aja planeerimine

- CPU aega planeeritakse, kuna on vaja tõsta:
 - CPU kasutamise efektiivsust
 - Vähendada kosteaega ehk *latency*
 - Kasutada multitegumi võimalusi (*multitasking*)
 - Vähendada voolutarvet
- Erinevaid näiteid aja planeerimisest
 - FCFS *First Come First Served*
 - SJN *Shortest Job Next*
 - Prioriteedi arvestamine
 - Nälgimise vältimine
 - *Round Robin*

Linuxi tuum

- GNU/Linux operatsioonisüsteemi süda, 1991 [Linus Torvalds](#)
- Tuum virtualiseerib riistvara, et pakkuda riistvara ressursse igale protsessile virtuaalselt eraldi.



Per-Process Hardware Virtualization

[EOL Linuxi tuum](#)

[Lühiülevaade Linuxi ajaloo](#)

[Linuxi tuuma versioonid \(Wikiwand\)](#)

lkml.org

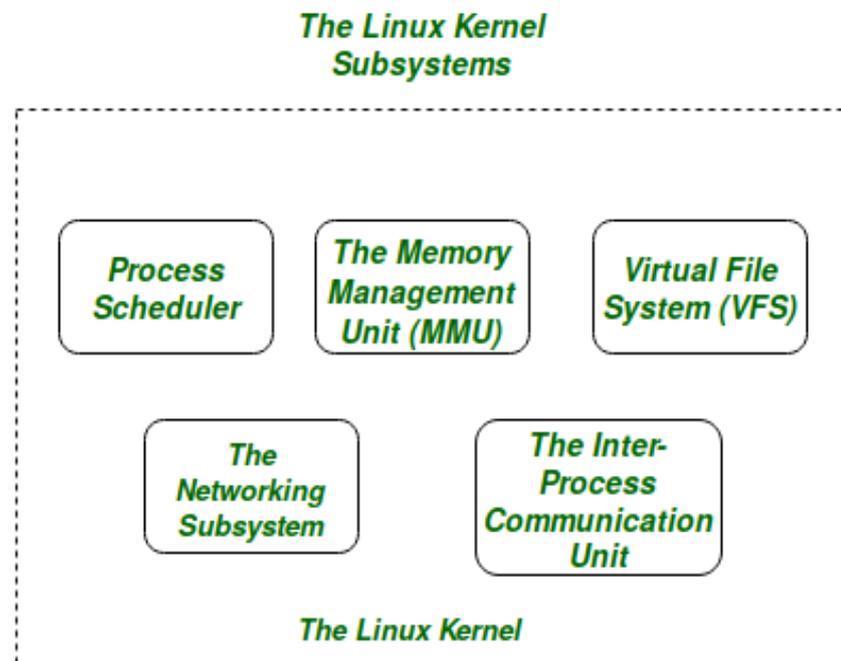
lwn.net/Kernel

[Linux tuuma ajalugu](#)

[Linux kernel evolution \(2024\)](#)

Linux tuuma alamsüsteemid

- Protsessi planeerija (*process scheduler*): see tuuma alamsüsteem vastutab protsessori aja õiglase jaotamise eest kõigi süsteemis samaaegselt töötavate protsesside vahel.
- Mäluhaldusüksus (*Memory Management Unit*): see tuuma alamüksus vastutab mäluressursside õige jaotamise eest erinevate süsteemis töötavate protsesside vahel. MMU teeb enamat kui lihtsalt iga protsessi jaoks eraldi virtuaalse aadressiruumi pakkumine.
- Virtuaalne failisüsteem (*Virtual File System*): see alamsüsteem vastutab ühtse liidese pakkumise eest, et pääseda juurde salvestatud andmetele erinevates failisüsteemides ja füüsilistes andmekandjates.



Linux'i arhitektuur

- GNU/Linux põhineb traditsioonilisel monoliitkerneli mudelil, kus kogu süsteemi põhifunktsioonid – nagu mäluhaldus, protsesside ajastus, riistvara haldus ja failisüsteemi tugi – on tihedalt integreeritud ühte koodi. Samas toetab Linux dünaamilist moodulite laadimist, mis võimaldab lisada või eemaldada funktsionaalsust jooksva süsteemi töös ilma kogu tuuma ümber kompileerimata.

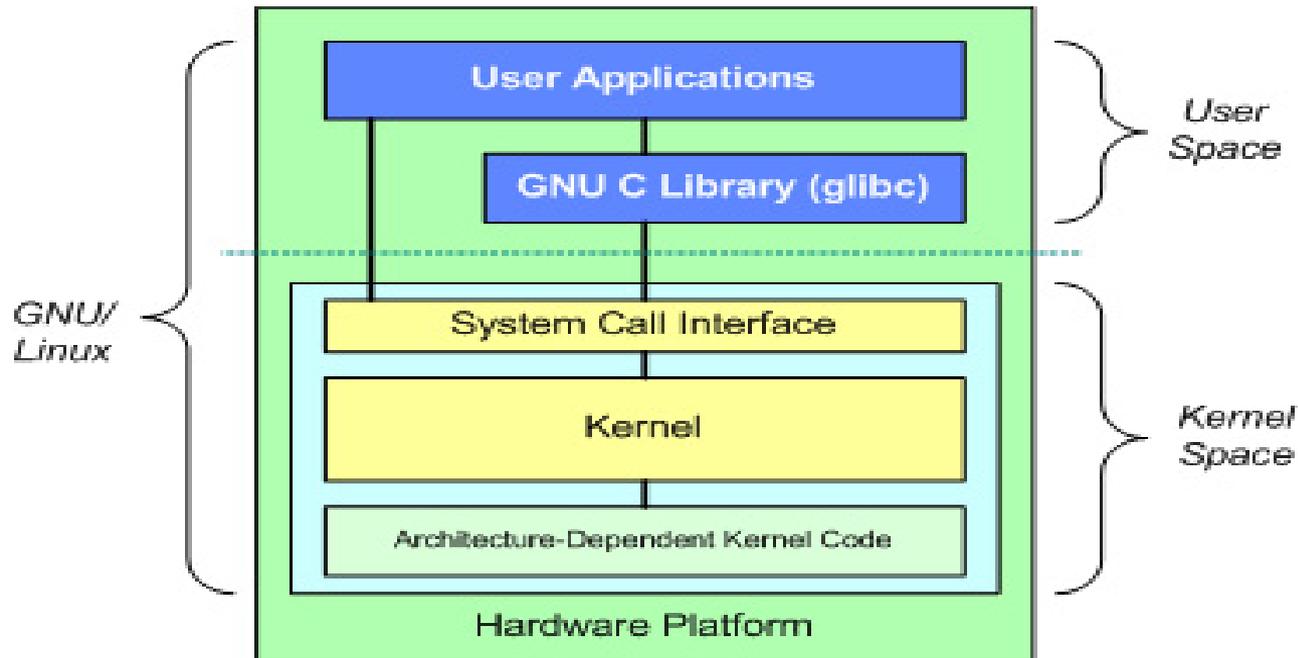
Kasutajapoolne osa koosneb mitmest kihist. Näiteks:

- Kest (*shell*) ja käsurealiidesed: Tõlgendavad kasutaja käsklusi.
- Graafilised keskkonnad (nt GNOME, KDE, XFCE): Pakuvad intuitiivset kasutajaliidest.
- Raamatukogud (*library*) ja süsteemikutsed (*system call*): Võimaldavad rakendustel suhelda tuuma ja riistvaraga abstraktsiooni kaudu.

Failisüsteemid:

- Linux toetab mitut failisüsteemi tüüpi (näiteks ext4, btrfs, ZFS) ning kasutab ka virtuaalseid failisüsteeme (nt procfs ja sysfs) süsteemiinfo esitamiseks ja haldamiseks.

Linux tuuma lihtsustatud skeem

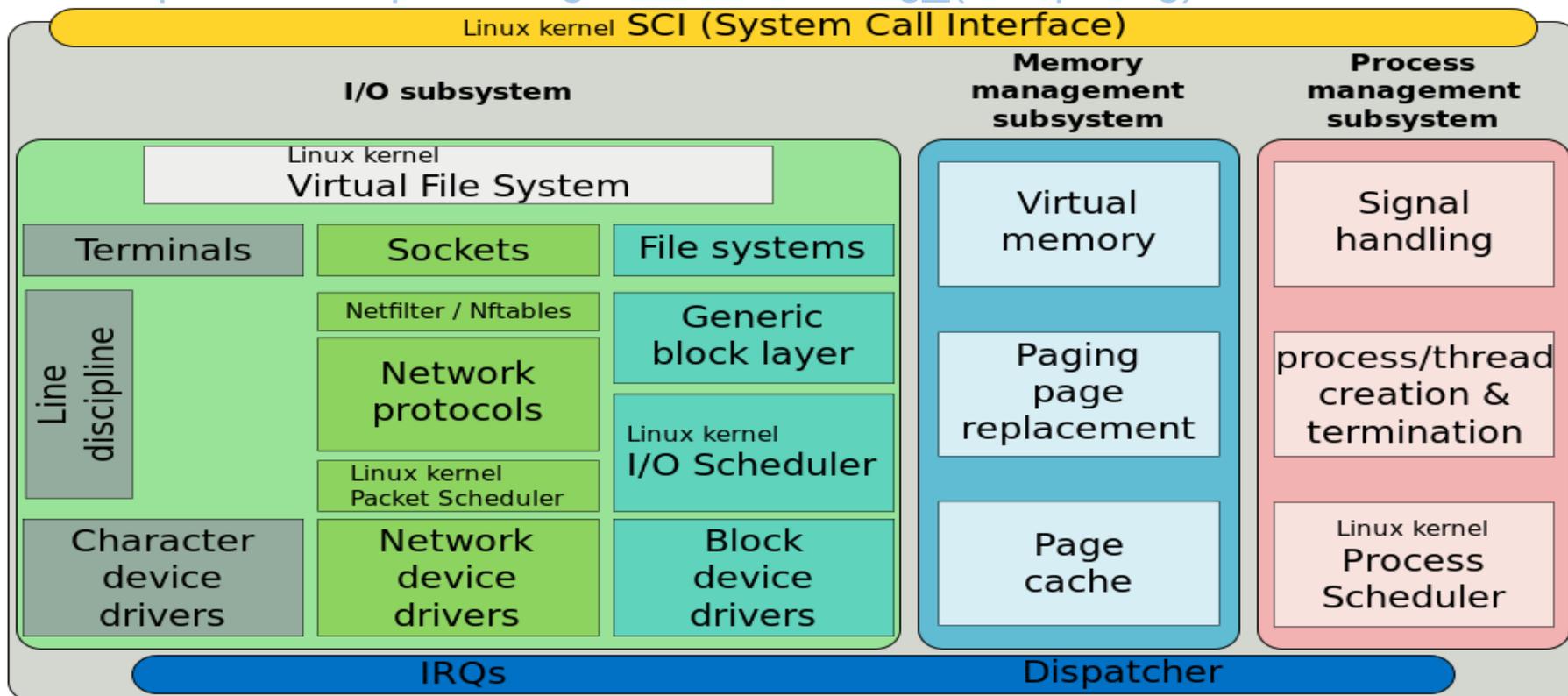


video: *How Linux is built?*

<https://www.youtube.com/watch?v=yVpbFMhOAwE>

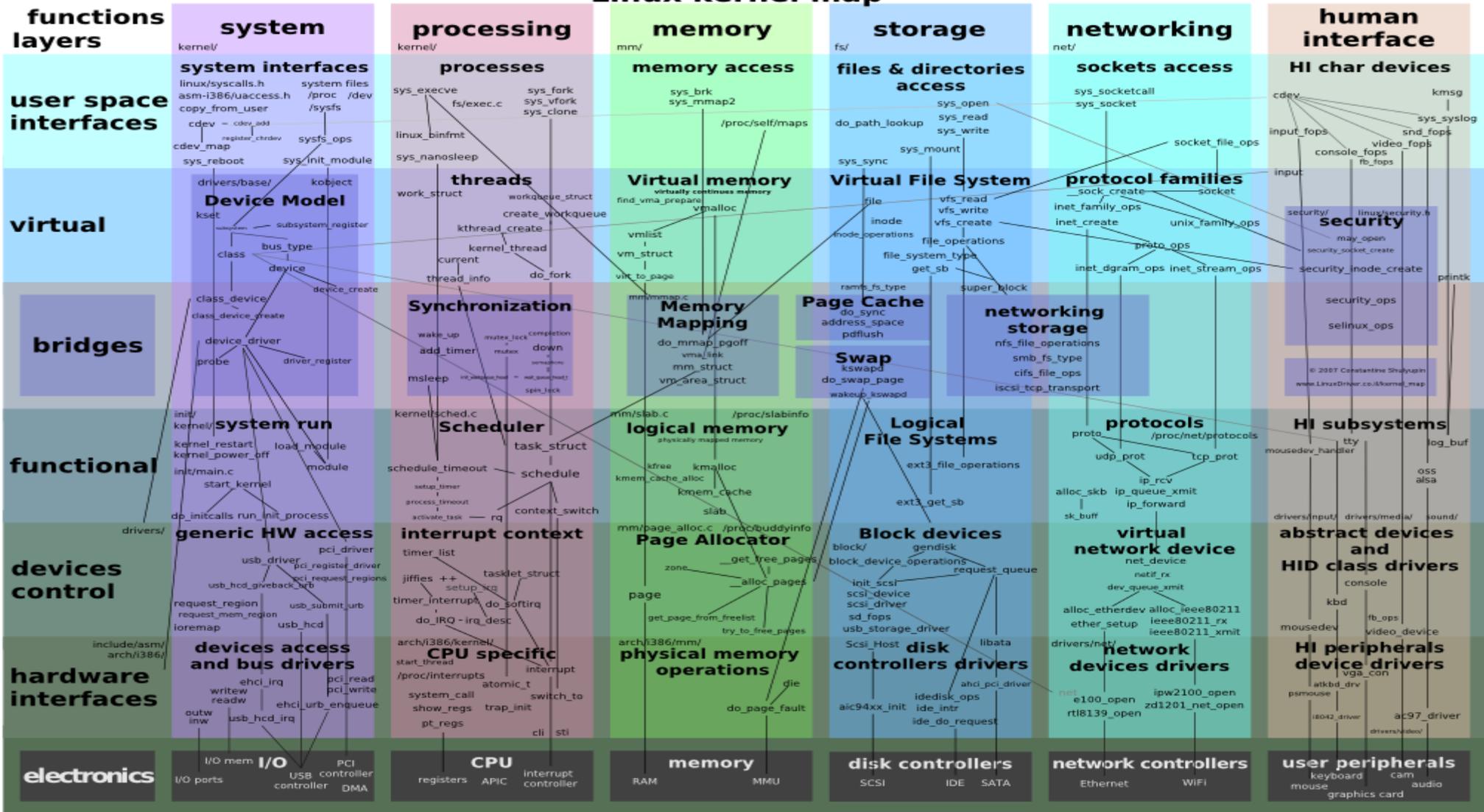
Linuxi tuum pisut detailsemalt

- [https://en.wikipedia.org/wiki/Scheduling_\(computing\)](https://en.wikipedia.org/wiki/Scheduling_(computing))



järgmisel slaidil: Linuxi tuum veelgi detailsemalt

Linux kernel map



Hübriidtuum:

- Windows NT-põhine süsteem kasutab hübriidtuuma arhitektuuri, mis ühendab nii monoliitsete kui ka mikrotuuma elemente. See lähenemine võimaldab süsteemil pakkuda stabiilsust ja turvalisust, säilitades samal ajal teatava modulaarse paindlikkuse.

Hardware Abstraction Layer (HAL):

- Windowsis on oluline kiht – HAL – mis varjab riistvaraspetsiifilisi detaile ning võimaldab süsteemil töötada erinevate riistvarasätetega ilma suurte ümberkorraldusteta.

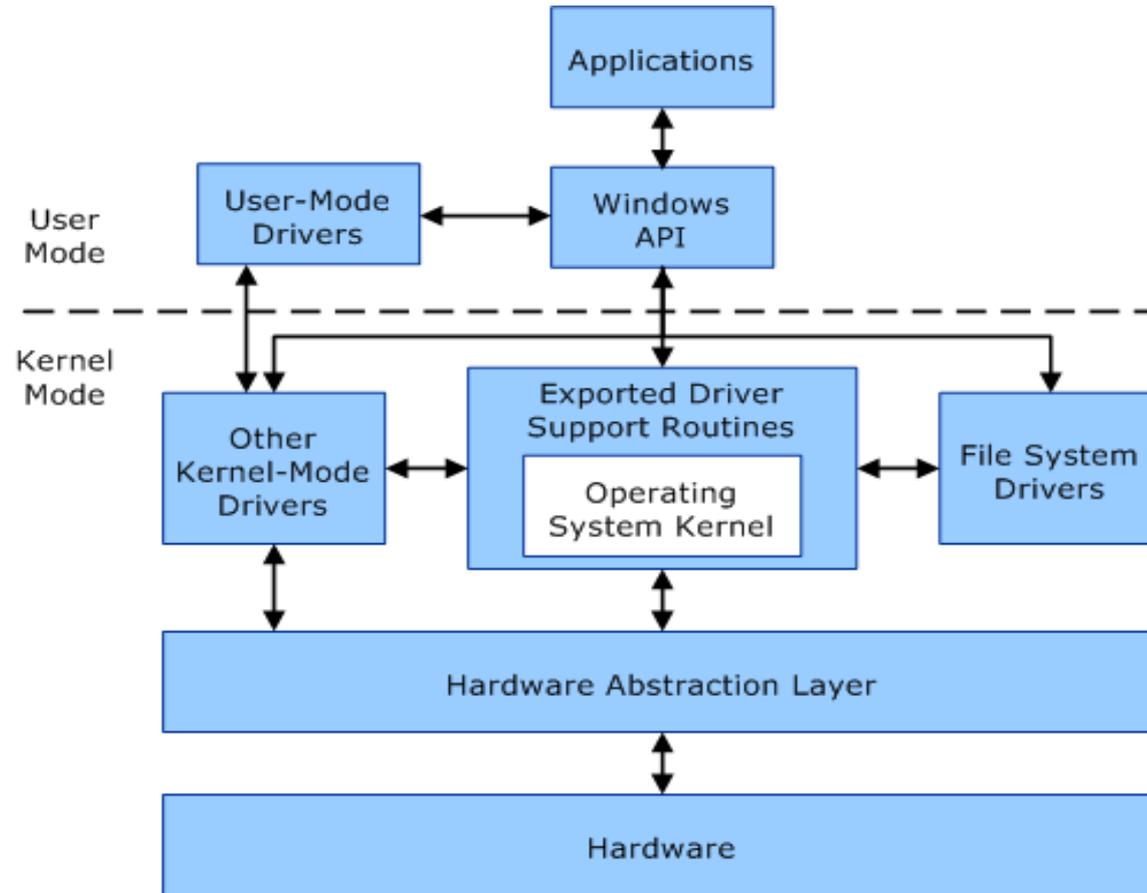
Kasutajaliidese ja rakenduste tugi:

- Kasutajapoolsed komponendid hõlmavad nii klassikalist Win32 API-d kui ka kaasaegsemaid raamistikke nagu .NET ja UWP, mis võimaldavad arendajatel luua mitmekülgseid ja turvalisi rakendusi.

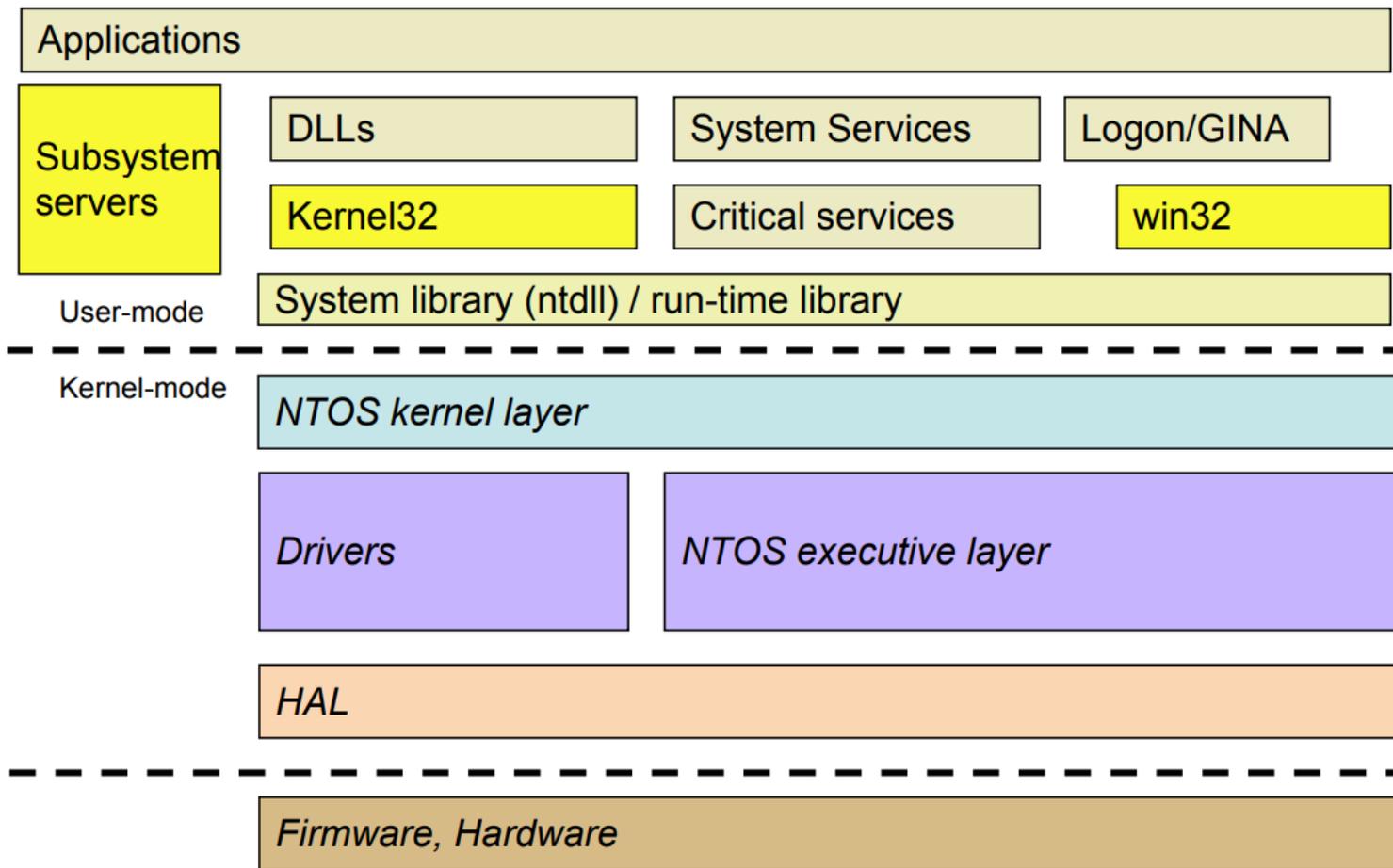
Failisüsteem:

- Peamiseks failisüsteemiks on NTFS, mis on loodud toetama suuri andmemahutusi, pakkuma turvalisust (näiteks failikodeerimine) ja säilitama süsteemi terviklikkust.

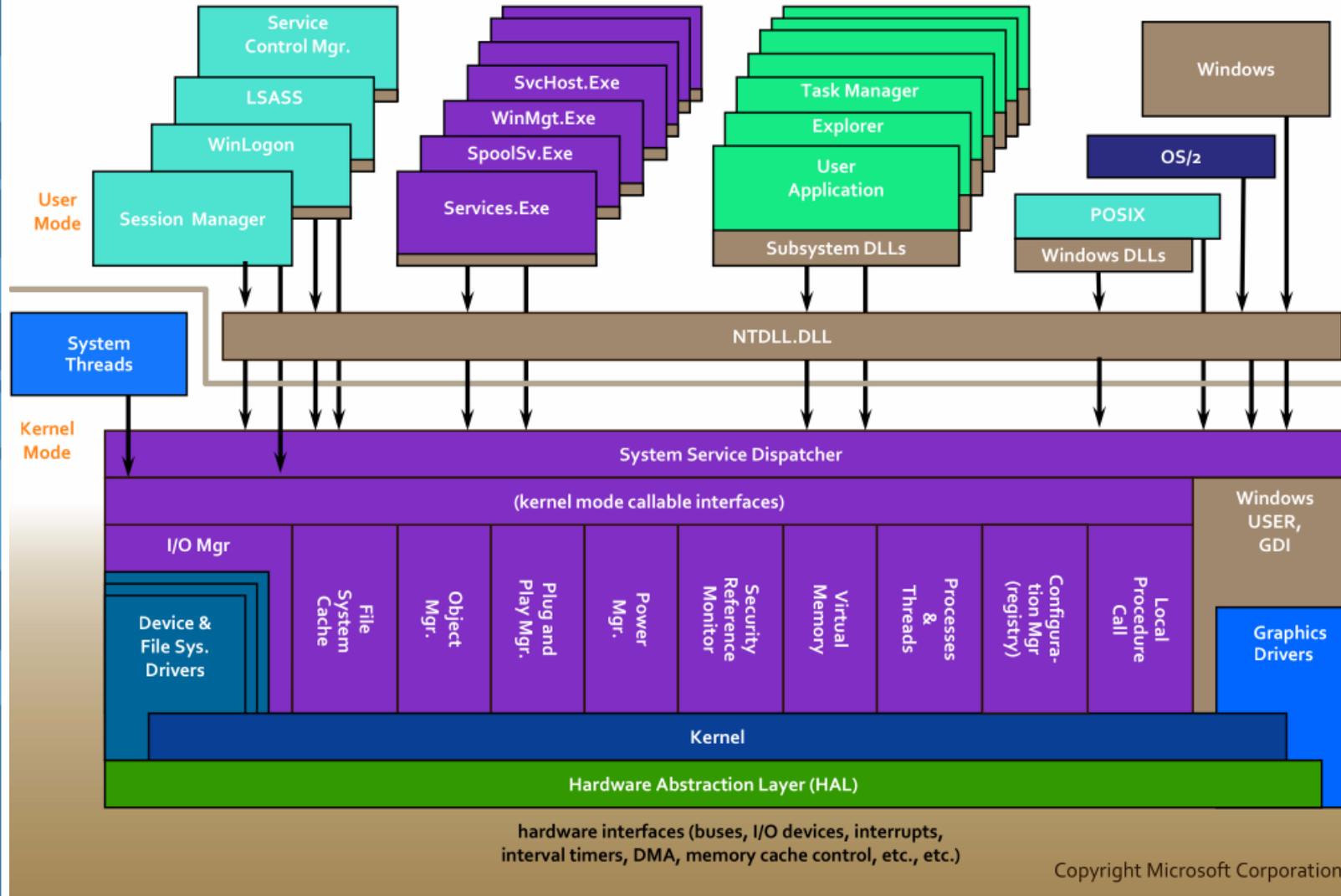
MS Windowsi arhitektuur lihtsustatult



MS Windowsi arhitektuur



MS Windowsi arhitektuur

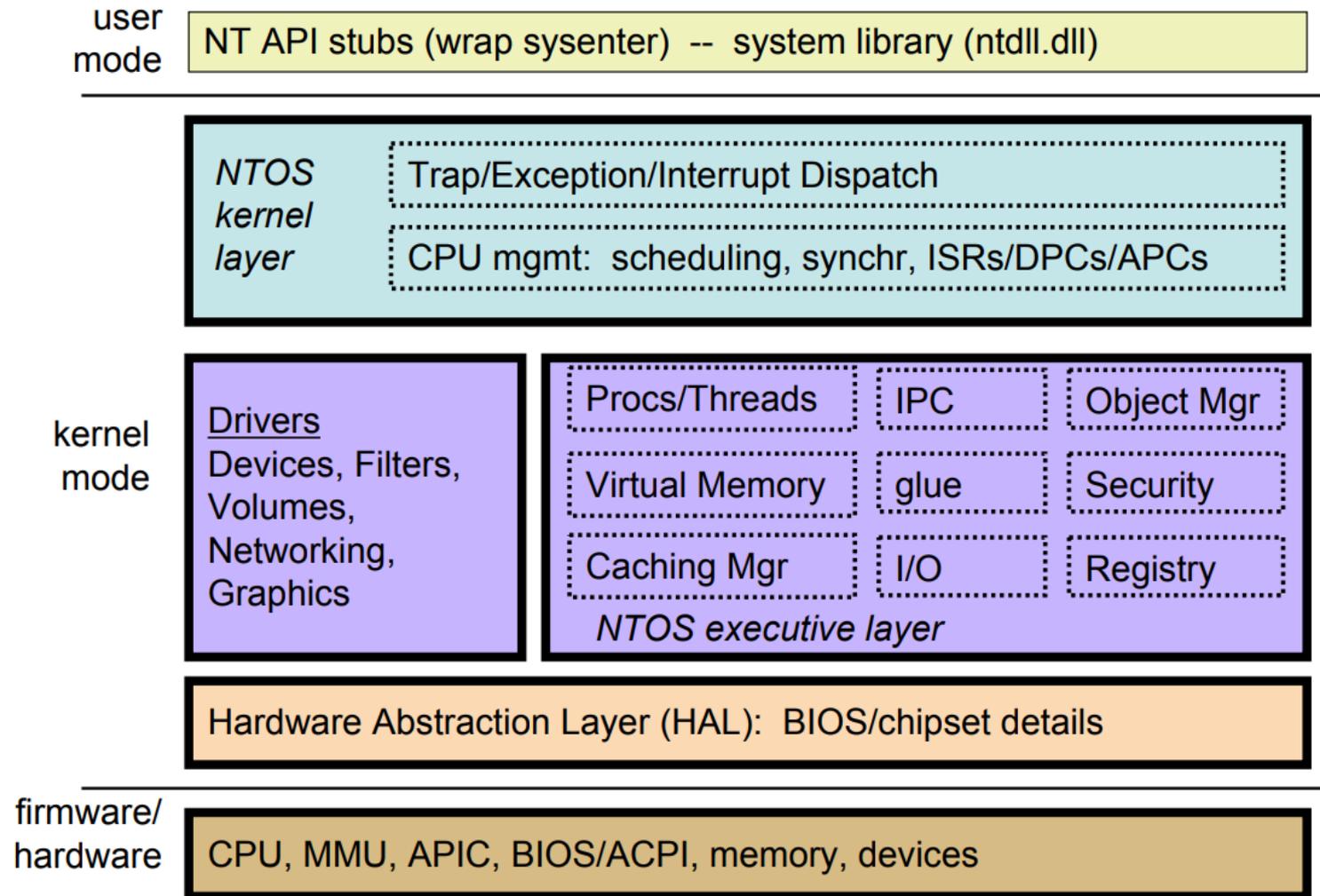


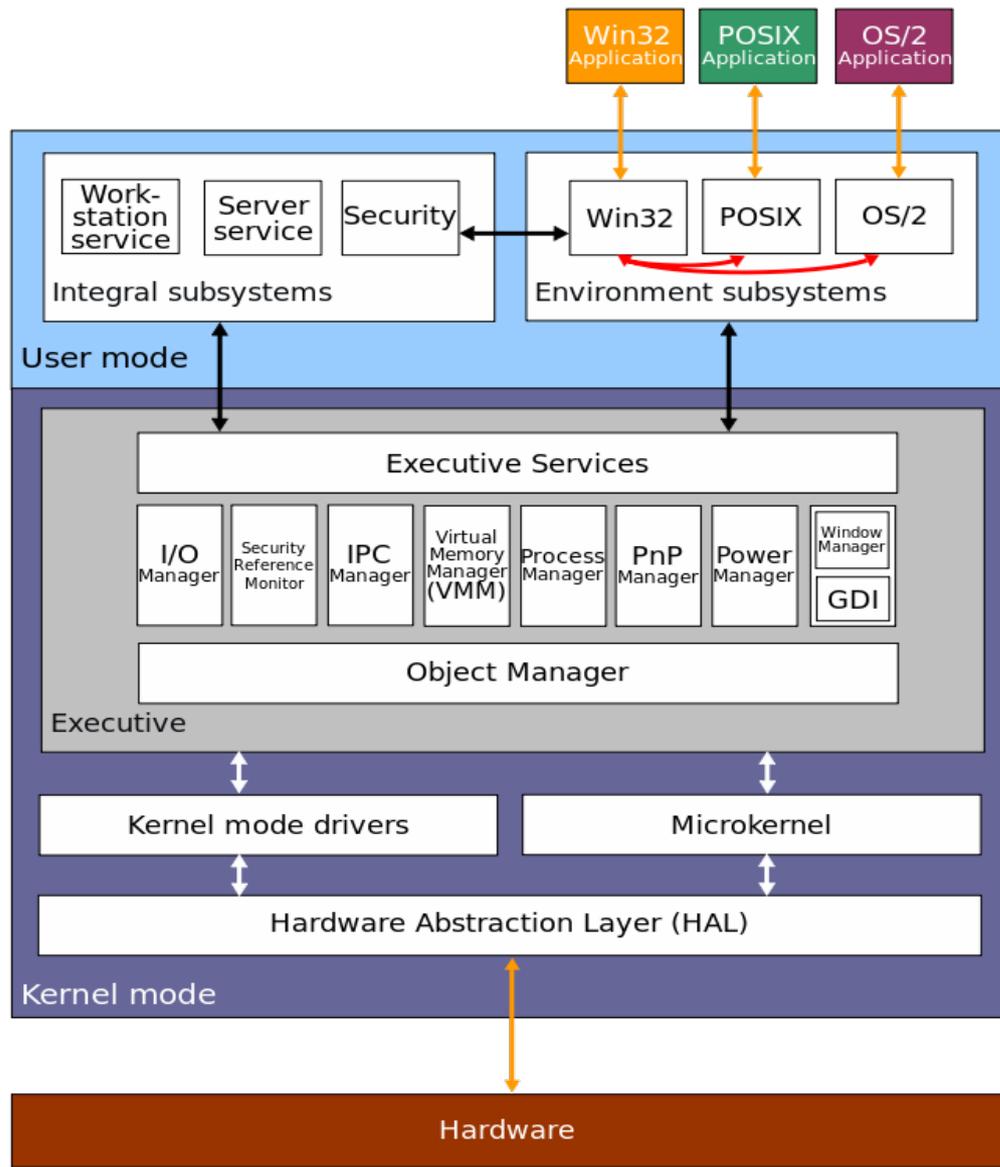
Sysinternals

Allikas

Copyright Microsoft Corporation

MS Windowsi tuuma režiimi arhitektuur





MS Windows NT perekonna arhitektuur

https://en.wikipedia.org/wiki/Architecture_of_Windows_NT
<https://blogs.technet.microsoft.com/askperf/2007/04/10/windows-architecture-the-basics/>

lisalugemist

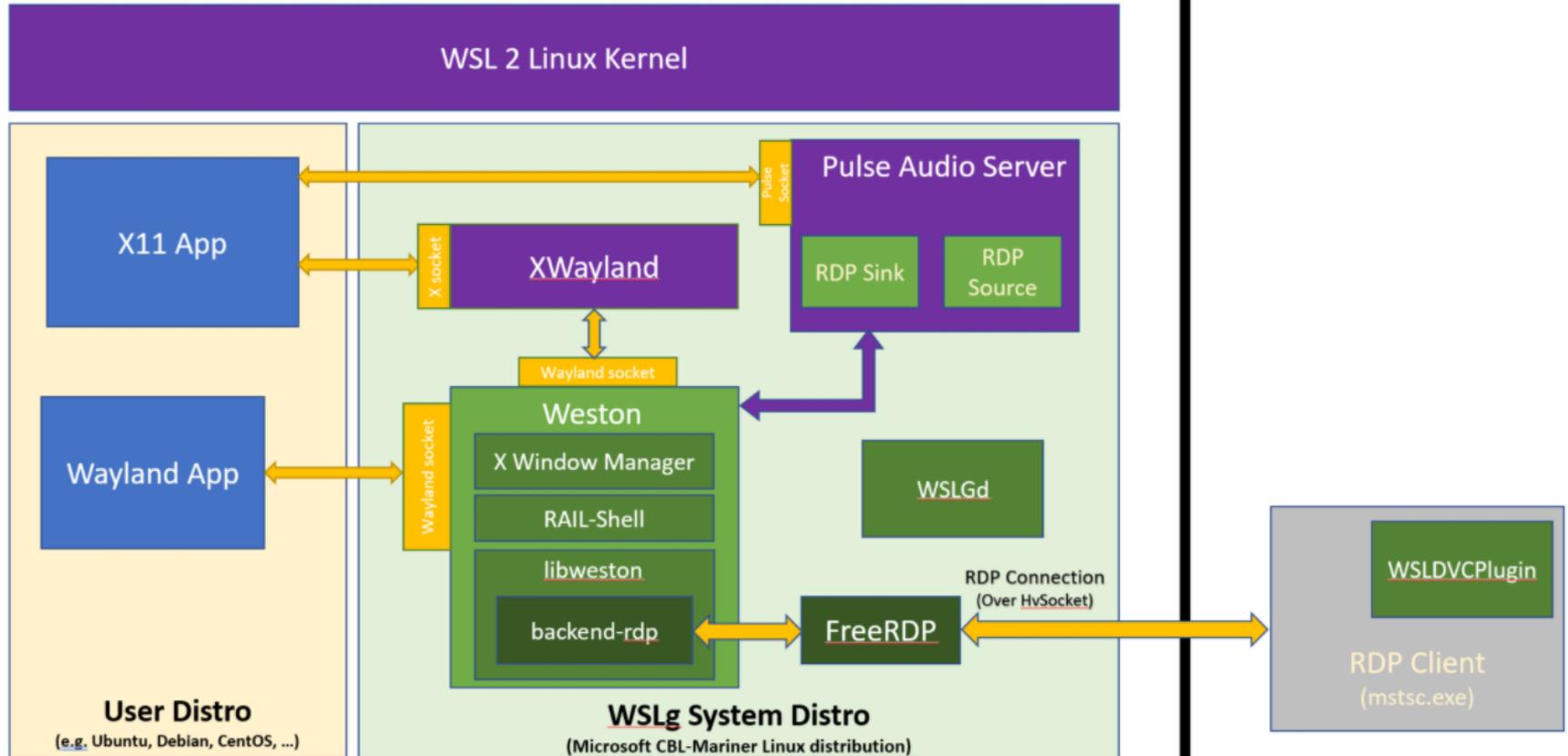
MS Windowsi tuum ja WSL

Windows Subsystem for Linux

WSLg arhitektuur

WSL Virtual Machine

Windows Host



XNU tuum:

- macOS ja iOS põhinevad XNU kernelil, mis on hübriidlahendus, ühendades Mach mikrotuuma ja BSD komponendid. See disain võimaldab süsteemil pakkuda head protsesside ja mälu haldust ning tugevaid turvamehhanisme.

Kasutajapoolne süsteem:

- macOS: Kombineerib UNIXi-põhiseid komponente, pakkudes graafilist kasutajaliidest, mis toetub Apple'i enda raamistikule (näiteks Cocoa ja Swift).
- iOS: Kuigi jagab sarnast tuuma, on iOS optimeeritud mobiilseadmetele – rõhuasetusega energiatõhususele, reageerimiskiirusele ja täiendavatele turvakihtidele, mis kaitsevad nii andmeid kui ka riistvara.

Integratsioon ja turvalisus:

- Apple'i süsteemid on tihedalt integreeritud riistvaraga, mis võimaldab saavutada kõrget jõudlust ning pakkuda innovaatilisi turvafunktsioone, sealhulgas sandboxing ja täiendavat autentimist.



Linux tuumal põhinev disain:

- Android kasutab kohandatud Linuxi tuuma, millel on sarnased põhimõtted nagu GNU/Linuxil, kuid see on optimeeritud mobiilseadmete spetsiifilisteks vajadusteks (nt energiatõhusus ja sensorite haldus).

Kihiline arhitektuur:

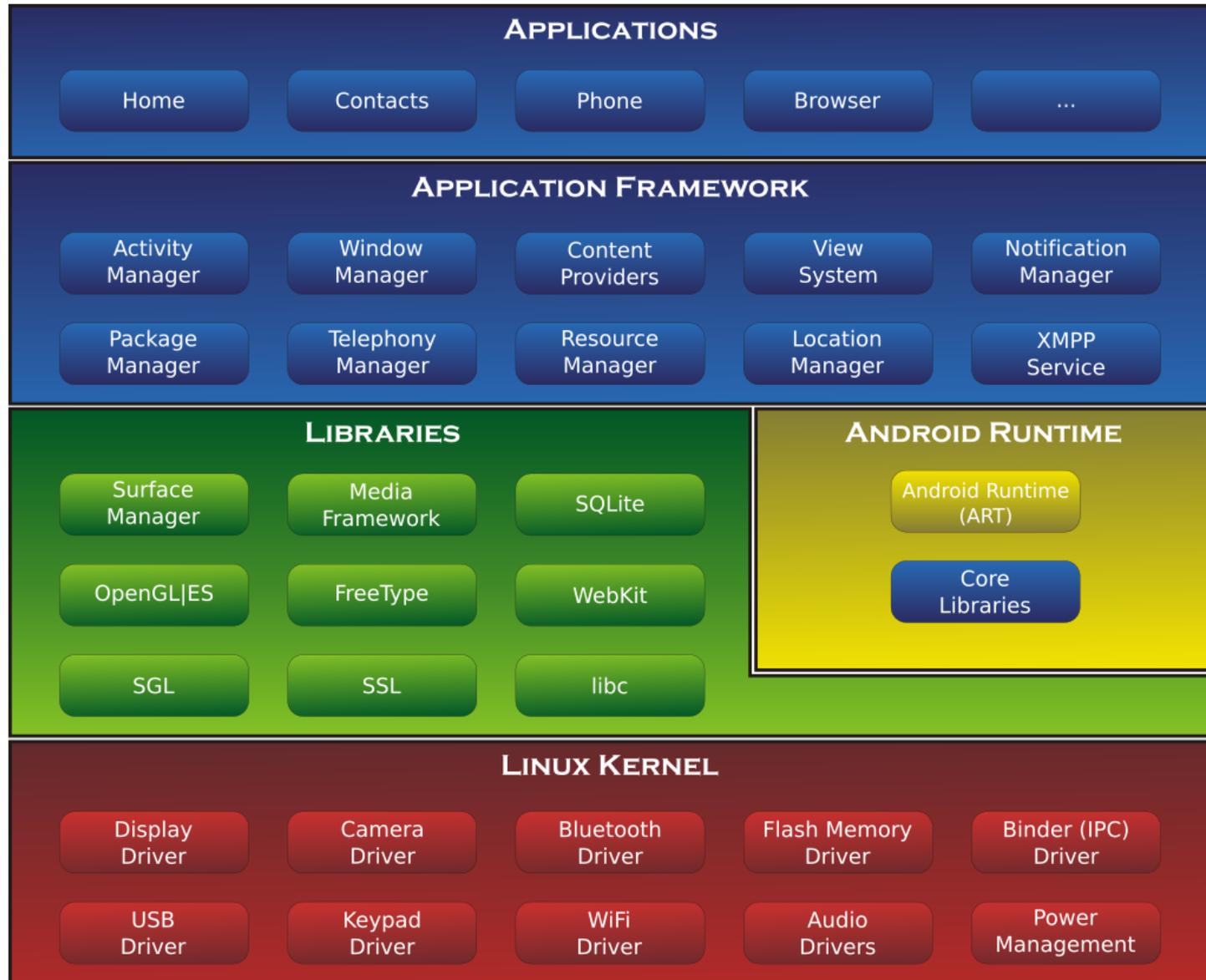
Androidi süsteemi ülesehitus jaguneb mitmeks kihiks:

- Linuxi tuum: tagab madala taseme riistvarajuhtimise ja süsteemi stabiilsuse.
- Loomulike teekide kiht: sisaldab C/C++ põhiseid raamatukogusid (näiteks [Bionic libc](#)), mis võimaldavad süsteemirakendustel suhelda kerneli ja riistvaraga.
- [Android Runtime](#) (ART): varasemalt kasutatud [Dalvik](#) virtuaalmasina asemel täidab ART rolli [Java](#) ja [Kotlin](#) keeles kirjutatud rakenduste käitamisel, pakkudes tõhusamat mälu- ja protsessihaldust.
- Rakenduste raamistik: Kõrgema taseme [API](#)-d ja teenused, mis lihtsustavad rakenduste arendamist ning kasutajaliidese loomist.

Optimeerimine mobiilseadmetele:

- Androidi arhitektuur on loodud toetama mitme rakenduse samaaegset käitamist, tagama energiatõhususe ja pakkuma tugevat turvalisust, mis on kriitilise tähtsusega nutitelefonide ja tahvelarvutite maailmas.

Androidi arhitektuur



Allikas

Erinevate operatsioonisüsteemide arhitektuurid

Erinevate operatsioonisüsteemide arhitektuurid on kujundatud vastavalt nende eesmärkidele ja kasutusjuhtudele:

- GNU/Linux rõhutab modulaarset ja avatud lähenemist, võimaldades dünaamilist kohandamist ning tugevat kogukonnapõhist arendust.
- MS Windows kasutab hübriidtuuma, mis tagab süsteemi stabiilsuse ja riistvarast sõltumatu toimimise läbi HAL'i.
- macOS ja iOS põhinevad XNU tuumal, pakkudes tugevat integratsiooni, jõudlust ja turvalisust, eriti mobiilseadmete puhul.
- Android omakorda kasutab Linuxi alust, aga kihilise arhitektuuri kaudu kohandab süsteemi mobiilseadmete unikaalsetele nõudmistele, pakkudes sujuvat ja energiatõhusat kasutuskogemust.

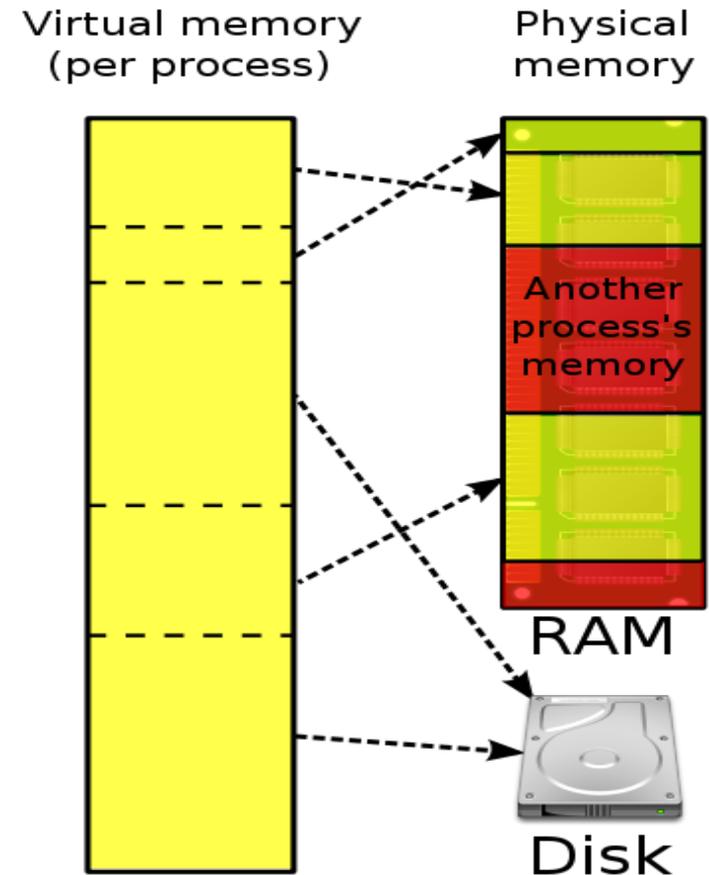
Need erinevad lähenemisviisid kajastavad mitte ainult tehnilisi otsuseid, vaid ka arenduskogukondade ja turu vajaduste mitmekesisust, mis on viinud kaasaegsete operatsioonisüsteemide pidevale arengule ja spetsialiseerumisele.

Virtuaalmälu ja mälu haldamine

- Iga protsess saab OS käest virtuaalse mälu.
- Protsess kasutab virtuaalseid aadresse, mis tõlgitakse MMU poolt füüsilisteks mäluaadressideks.
- Saalimine, *memory paging*, *swapping* – muutmälu pikendus kõvakettal (SSD, HDD)
 - Kui mälu ei kasutata, siis võidakse see välja saalida
 - Mäluleht laetakse saalealast mällu nõudmisel
 - Suurendab vaba mälu
- Ketta vahemälu – reeglina kasutatakse kogu vaba mälu ära failide puhverdamiseks

Virtuaalmälu

- Mälukaitse on OS osa
- Iga protsess näeb katkematut ja virtuaalset aadressiruumi
- MMU (mäluhaldusüksus) ja OS teisendavad virtuaalse aadressi reaalseks



Vahemälu

- Süsteemi kogujõudluse tõstmiseks kasutatakse vahemälu (*cache*)
- Protsessori vahemälu
- Ketta vahemälu
- Puhvermälu *versus* vahemälu

Puhvermälu vs vahemälu

- **Puhvermälu ((*data*) *buffer*)** – muutmäluosa ajutiseks andmesalvestuseks, mis võimaldab ajutise salvestuse abil vahetada andmeid kahe erinevate edastusparameetritega üksuse vahel. Enamiku puhvrite ülesanne on säilitada andmeid selleks, et protsessor saaks neid töödelda enne mingile seadmele (nt printerile või kõvakettale) saatmist.
- **Vahemälu (*cache*)** – tark- või riistvara komponent, mis hoiustab andmeid nende kiireks uuestikasutamiseks. Vahemälust andmete lugemine on kiirem kui lähteandmete lugemine muutmälust (RAM) või kõvakettalt.

Puhvermälu vs vahemälu 2

- Kui te töötate tekstiprotsessoriga mingi dokumendi kallal, siis hoitakse seda dokumenti kontoritarkvara puhvris ja alles siis, kui annate salvestamiskäsu, saadetakse dokumendifail puhvrist kettale (andmekandjale).
- Keskprotsessori vahemällu loetakse andmed arvuti põhimälust, kõvaketta vahemällu loetakse andmed kõvakettalt, veebilehitseja vahemällu salvestatakse andmed veebist jne
- **Puhver**
 - on midagi, mis tuleb kirjutada kettale (vms seadmele)
 - hoiab metaandmeid (õigused, asukoht jne)
 - hoiab järge, millise seadme mälu on loetud/kirjutatud
- **Vahemälu**
 - on midagi, mida tuleb lugeda kettalt ja säilitada hilisemaks kasutamiseks
 - hoiab reaalselt failisisu

Sisend/Väljund

- Sünkroonne
 - Andmed kirjutatakse andmekandjale, programmi töö läheb edasi
- Asünkroonne
 - Andmed edastatakse operatsioonisüsteemile ja programmi töö jätkub
 - Töökindlus kannatab
- Failisüsteem
 - Andmete salvestamise kord

Ühikud

- SI-mõõtühikute süsteem
 - kilo, mega, giga, tera, peta
- IEC/ISO binaarühikute süsteem
 - kibi, mebi, gibi, tebi, pebi

Faktor	Nimi	Sümbol	Kordsus	SI süsteemis
2^{10}	kibi	Ki	$(2^{10})^1 = 2^{10}$	kilo: $(10^3)^1 = 10^3$ - tuhat
2^{20}	mebi	Mi	$(2^{10})^2 = 2^{20}$	mega: $(10^3)^2 = 10^6$ - miljon
2^{30}	gibi	Gi	$(2^{10})^3 = 2^{30}$	giga: $(10^3)^3 = 10^9$ - miljard
2^{40}	tebi	Ti	$(2^{10})^4 = 2^{40}$	tera: $(10^3)^4 = 10^{12}$ - triljon
2^{50}	pebi	Pi	$(2^{10})^5 = 2^{50}$	peta: $(10^3)^5 = 10^{15}$ - kvadriljon
2^{60}	eksbi	Ei	$(2^{10})^6 = 2^{60}$	eksa: $(10^3)^6 = 10^{18}$ - kvintiljon

Failisüsteemid

- On erinevad
 - Võivad arvestada failiõigustega
 - Omavad erinevat arusaamist lubatud failinimest
 - Võivad olla erinevate piirangute ja eesmärkidega
- Näiteks:
 - UNIXilaadsed: [EXT{2,3,4}](#), [btrfs](#), [reiserfs](#), [XFS](#), [ZFS](#), jne
 - Microsoft: [NTFS](#), [ReFS](#) (salvestussüsteemid), [FAT{12,16,32}](#), [exFAT](#);
 - Apple: [HFS](#), [HFS+](#), [APFS](#)
- Võrgufailisüsteemid, hajutatud failisüsteemid

Lisalugemist:

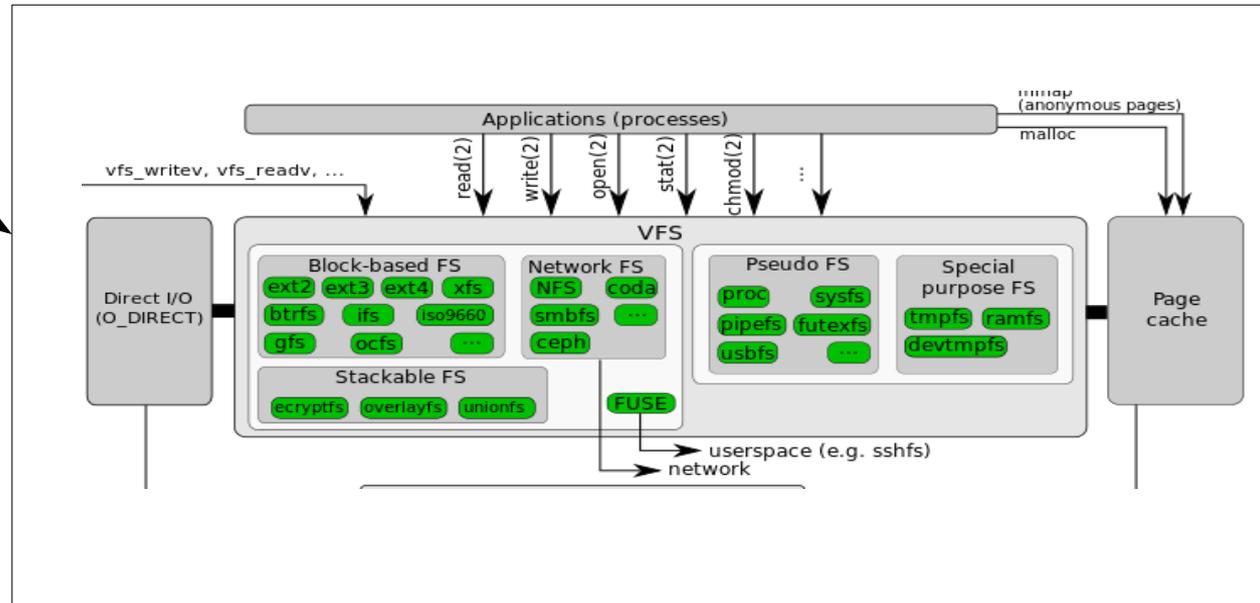
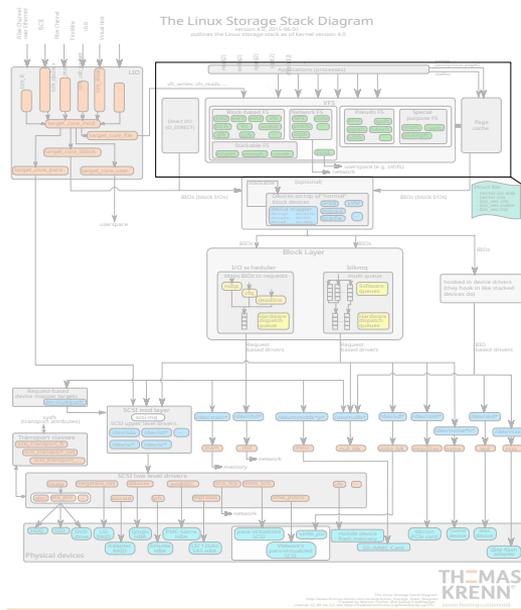
<https://wiki.itcollege.ee/index.php/Failis%C3%BCsteem>

https://en.wikipedia.org/wiki/Comparison_of_file_systems

Failisüsteem

- Failisüsteem ja selle hierarhia – OS'ide kaupa erinev. Vt Linuxis https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard
- Virtuaalne failisüsteem (VFS – *virtual file system*) – tarkvaraline kiht „päris” failisüsteemi peal, mis lubab paista erinevad failisüsteemid ühtsena rakenduste jaoks

Virtuaalne failisüsteem



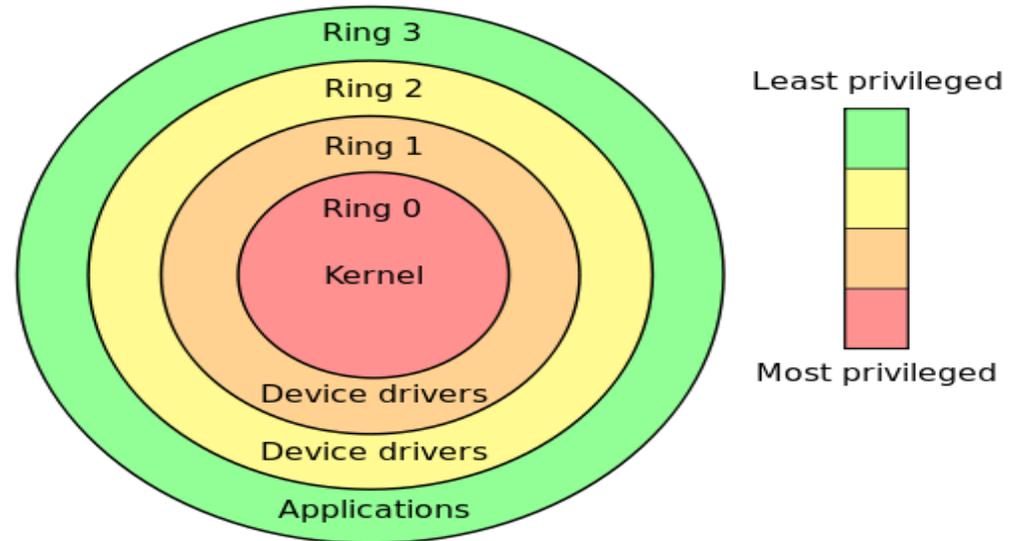
Baasvahetussüsteem, andmekandja jagude tabel

- **BIOS** (*Basic Input/Output System, baasvahetussüsteem*) arvutis võib andmekandjal olla kuni neli primaarset andmekandja jagu (*primary partition*), neist üks võib olla laiendatud (*extended partition*) koos vähemalt 1-128 loogilise jaoga (*logical volume*)
 - **MBR** (kuigi GPT teatud tingimustel toimib) – jagude tabel
 - MS Windowsi paigaldamine GPT peale BIOSi puhul takistatud (*UEFI-mode* vajalik kui BIOS toetab)
 - GNU/Linux: *BIOS Boot Partition* teha
- **UEFI** (*Unified Extensible Firmware Interface*) puhul on kõik primaarsed jaod. UEFI asendab BIOSi.
 - Windows: System (ESP EFI System Partition), 100 MiB + MSR (*Microsoft Reserved Partition*) 16 MiB (Win10), seadmetähist ei saa
 - Linux: EFI, 100 MiB
 - GPT (kuigi MBR teatud tingimustel toimib) – jagude tabel
 - üle 2 TiB, kuni 8 ZiB (8 mln TiB) andmekandja jaod
 - teoreetilist arvu piiri ei ole, praktikas üldjuhul 128

Kaitstud režiim

- 0 – OS
- 3 – rakendused
- Milleks?

ühidalt: süsteemi kaitseks



https://en.wikipedia.org/wiki/Operating_system#Modes

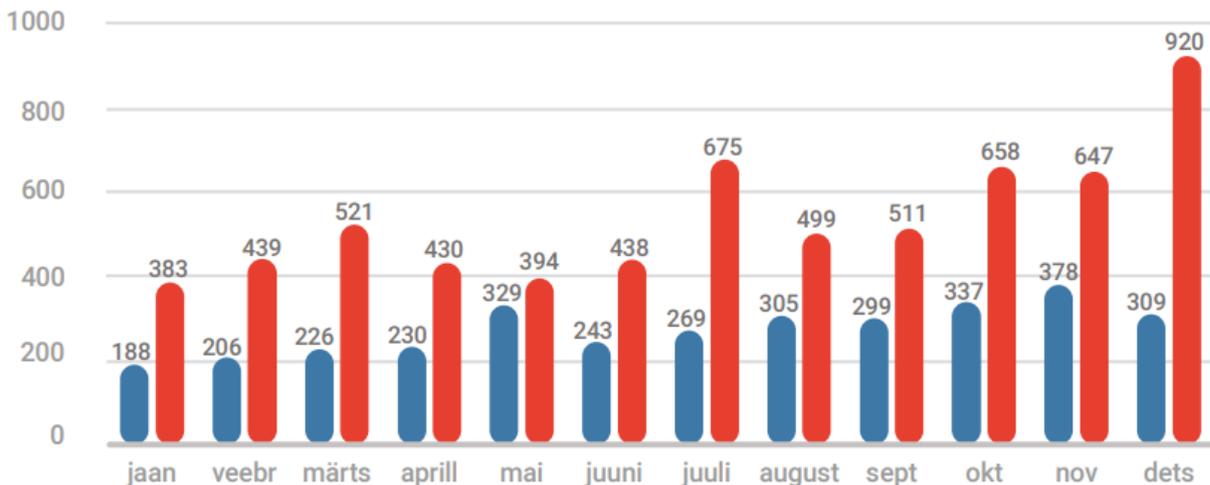
http://en.wikipedia.org/wiki/File:Priv_rings.svg

Süsteemi jõudlus

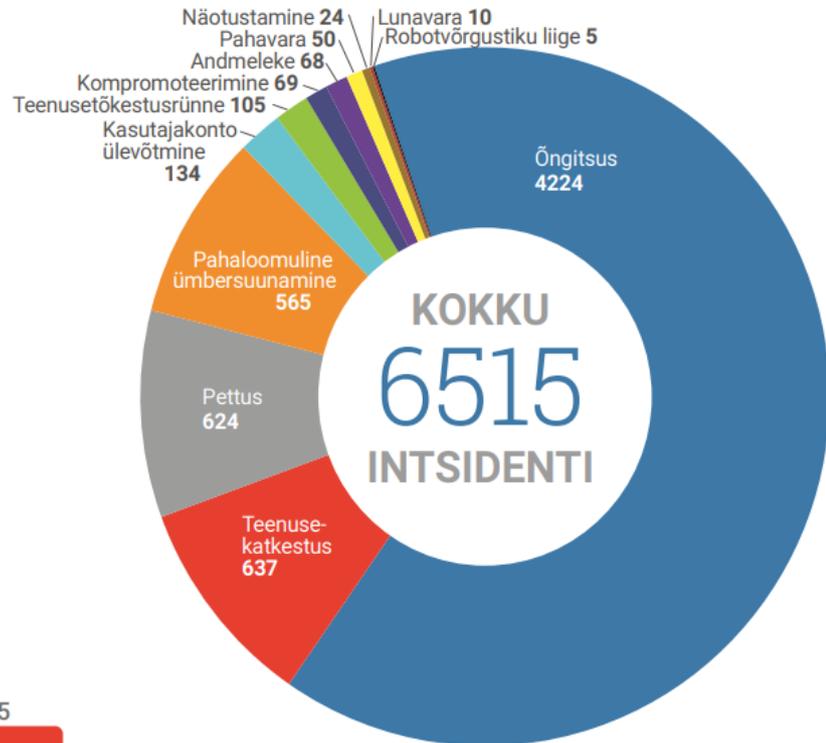
- Arvuti jõudlus (*computer performance*) näitab arvutisüsteemi kui terviku efektiivsust, sh andmetöötlemisvõimekust; mõõdetakse jõudlustestidega (*benchmark*); mõned iseloomustavad näitajad:
 - lühike reaktsiooniaeg (*response time*)
 - kõrge läbilaskevõime (*throughput*)
 - madal **ressursside** kasutus (*low utilization*)
 - kõrgkäideldavus (*high availability*)
 - kiire (või kõrge tihedusega) andmete pakkimine (*data compression*)
 - kõrge ribalaius (*high bandwidth*)
 - lühike andmete ülekandeaeg (*data transmission*),
 - kosteaeg (*latency*) – süsteemi ja selle eri komponentide viivitus pöördumise ja süsteemi reaalse vastamise vahel (vt ka *access time*)
 - **MIPS** – CPU'd iseloomustav näitaja, Linuxis ka **BogoMIPS**
 - failisüsteemi jõudlus, vt **Linuxi failisüsteemide jõudlustest**, NTFS'ist

Turvalisus on oluline!

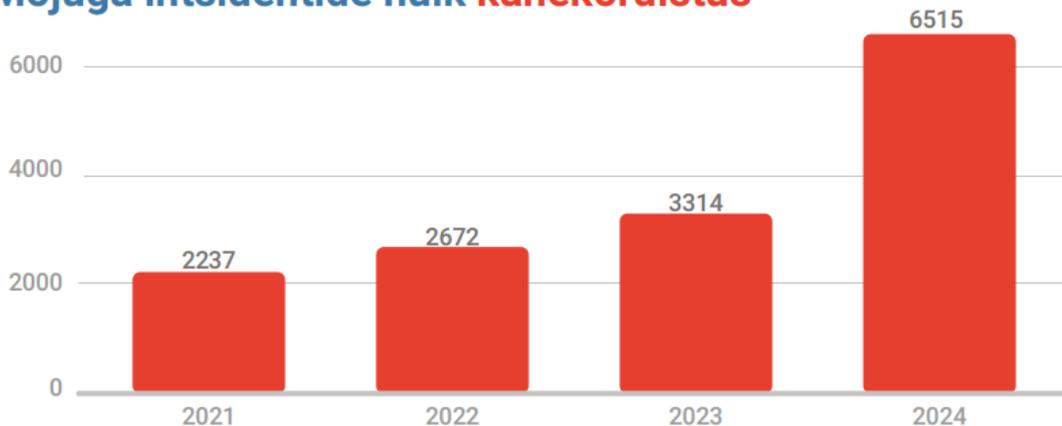
Mõjuga intsidentide hulk kuude lõikes



Mõjuga intsidendid 2024. aastal



Mõjuga intsidentide hulk kahekordistus



CERT-EE AVALIKUKS KASUTUSEKS MÕELDUD TEENUSED JA TÖÖRIISTAD

Veebipõhine viiruskontrolli tööriist

<https://irma.cert.ee/>

Tööriist riigiasutuste andmesidevõrgu kasutajatele ja erasektori koostööpartneritele, mõeldud e-kirjaga saabunud kahtlaste manuste ja teiste eba-kindla päritoluga failide kontrollimiseks. Tööriista eelis internetis leiduvate samalaadsete ees on, et sisestatud failid ei jää tundmatutesse kohtadesse ripakile, vaid paiknevad Eesti riigiasutuse failiserveris ja neid kustutatakse regulaarselt.

Infoturbeintsidentidest teavitamine

<https://raport.cert.ee>

Teavituskeskonna kaudu saab RIA-le edastada teate infoturbeintsidenti kohta. Mõeldud eeskätt asutustele ja teenuseosutajatele detailsema teabe edastamiseks, lihtsa intsidentiteavituse võib saata ka aadressil cert@cert.ee.

Failide edastuskeskkond

<https://paste.cert.ee>

Tööriist võimaldab saata kahtlased failid CERT-EE-le analüüsimiseks. Sobib õngitsuskirjade ja nendega saabunud manuste, pahavaranäidiste jms edastamiseks.

CERT-EE „liivakast“ (Sandbox)

<http://cuckoo.cert.ee>

IT-spetsialistidele mõeldud failide analüüsimise tööriist. Võimaldab turvalises keskkonnas järele kontrollida, kuidas erinevatel virtuaalsetel ja füüsilistel platvormidel töötavad operatsioonisüsteemid kahtlusaluse faili käivitumisel käituvad.

CERT-EE hoiatused ja teated

https://twitter.com/cert_ee

Kõige operatiivsem viis püsida kursis CERT-EE teadete ja hoiatustega.

Kübervaldkonna uudiskiri

<https://www.ria.ee/ee/cert-kontakt.html>

Iga päev ilmuv kokkuvõte avalikes allikates ilmunud küber- ja IT-uudistest. Listiga saab liituda ametliku e-posti aadressiga (ei sobi Gmail, Hotmail vms).

RIA blogi

<https://blog.ria.ee>

Sisaldab pikemaid analüüse ja kirjutisi aktuaalsetel teemadel, sh küberturbest.

<https://cert.ee/>

Administraator

- Haldur (*administraator*)
 - Süsteemihaldur
 - Süsteemide paigaldamine
 - Süsteemide haldamine
 - Süsteemide varundamine, taastamine
 - Süsteemi koormuse ja mahtude jälgimine
 - Süsteemi arhitektuuri arendamine
 - Rakendusehaldur
 - Rakenduse, teenuse spetsiifiline haldamine (näiteks majandustarkvara)

Süsteemide administreerimine

- Süsteemihalduril tuleb tihti tegeleda
 - Hangete läbiviimisega.
 - Süsteemi vajaduste jälgimise ja planeerimisega.
 - Testimisega ja dokumenteerimisega.
 - Arvutivõrgu seadmete paigaldamise ja haldamisega.
 - Süsteemi lihtsustamisega.
 - Tegevuste automatiseerimisega.
 - Töökoha arvutite haldamisega. (PC admin)
 - Rakenduste integreerimisega.

Süsteemide administreerimine 2

- Jätk...
 - programmeerimine
 - uute rakenduste otsimine ja valimine
 - kasutajate toetamine
 - turvaintsidentide lahendamine
 - jõudluse tõstmine
 - pidev enesetäiendamine ja uute tehnoloogiate uurimine
 - **suhtlemine!**

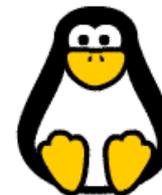
Viited

- https://en.wikipedia.org/wiki/Comparison_of_operating_system_kernels
- https://en.wikipedia.org/wiki/Comparison_of_operating_systems
- https://en.wikipedia.org/wiki/Category:Operating_systems_by_architecture
- https://en.wikipedia.org/wiki/Timeline_of_operating_systems
- https://en.wikipedia.org/wiki/Comparison_of_open-source_operating_systems
- https://en.wikipedia.org/wiki/Comparison_of_Linux_distributions
- https://en.wikipedia.org/wiki/Linux_kernel
- <https://opensource.com/article/17/8/linux-anniversary>
- https://en.wikipedia.org/wiki/Architecture_of_Windows_NT
- https://en.wikipedia.org/wiki/Architecture_of_Windows_9x
- https://en.wikipedia.org/wiki/Comparison_of_BSD_operating_systems
- https://en.wikipedia.org/wiki/Comparison_of_Microsoft_Windows_versions
- https://en.wikipedia.org/wiki/List_of_operating_systems
- https://en.wikipedia.org/wiki/Operating_system_advocacy
- https://en.wikipedia.org/wiki/Technology_evangelist
- https://en.wikipedia.org/wiki/Technology_adoption_life_cycle
- https://en.wikipedia.org/wiki/Hacker_culture

Küsimused? Tänan tähelepanu eest!



IT KOLLEDŽ
TALLINNA TEHNIKAÜLIKOOL



TALTECH IT KOLLEDŽ

Raja 4C, 12616 Tallinn

tel +372 628 5800

info@itcollege.ee

<https://taltech.ee/itcollege>