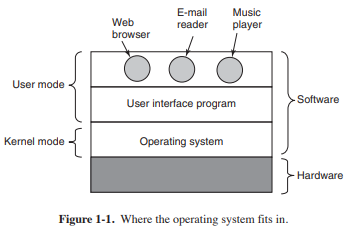
**SISTEMI II**

* Sodoben računalnik je sestavljen iz enega ali več procesorjev, nekaj glavnega pomnilnika, diskov, tiskalnikov, tipkovnice, miške, zaslona, omrežnih vmesnikov in različnih drugih vhodno/izhodnih naprav.
* Zaradi tega so računalniki opremljeni s plastjo programske opreme, imenovano operacijski sistem, katerega naloga je zagotoviti uporabniškim programom boljši, enostavnejši, čistejši model računalnika in upravljati z vsemi pravkar omenjenimi viri.
* Program, s katerim uporabniki komunicirajo, običajno se imenuje lupina, ko temelji na besedilu, in GUI (grafični uporabniški vmesnik) – ki se izgovori kot »gooey« – ko uporablja ikone, pravzaprav ni del operacijskega sistema, čeprav uporablja operacijski sistem za opravljanje svojega dela.



* Tukaj vidimo strojno opremo na dnu. Strojna oprema je sestavljena iz čipov, plošč, diskov, tipkovnice, monitorja in podobnih fizičnih predmetov. Na vrhu strojne opreme je programska oprema.
* Večina računalnikov ima dva načina delovanja: način jedra in uporabniški način.
* Operacijski sistem, najbolj temeljni del programske opreme, deluje v načinu jedra (imenovan tudi nadzorniški način).
* V tem načinu ima popoln dostop do vse strojne opreme in lahko izvede katero koli navodilo, ki ga je stroj sposoben izvesti. Preostala programska oprema teče v uporabniškem načinu, v katerem je na voljo samo podnabor strojnih navodil. Programom v uporabniškem načinu so zlasti prepovedana tista navodila, ki vplivajo na nadzor stroja ali izvajajo V/I )Input/Output«.
* Program uporabniškega vmesnika, lupina ali GUI, je najnižja raven programske opreme za uporabniški način in omogoča uporabniku, da zažene druge programe, kot so spletni brskalnik, bralnik e-pošte ali predvajalnik glasbe. Tudi ti programi močno uporabljajo operacijski sistem.
* Postavitev operacijskega sistema je prikazana na sliki 1-1. Deluje na goli strojni opremi in zagotavlja osnovo za vso ostalo programsko opremo. Pomembna razlika med operacijskim sistemom in običajno programsko opremo (uporabniški način) je v tem, da če uporabniku ni všeč določen bralnik e-pošte, si lahko pridobi drugega ali napiše svojega, če se tako odloči; ni svoboden pisati lastnega upravljalnika prekinitev ure, ki je del operacijskega sistema in je s strojno opremo zaščiten pred poskusi uporabnikov, da ga spremenijo.

**KAJ JE OPERACIJSKI SISTEM?**

* Težko je določiti, kaj je operacijski sistem, razen če rečemo, da gre za programsko opremo, ki deluje v načinu jedra - in tudi to ni vedno res.
* Del težave je v tem, da operacijski sistemi opravljajo dve v bistvu nepovezani funkciji: zagotavljajo programerjem aplikacij (in programom seveda) čist abstrakten nabor virov namesto neurejenih strojnih virov in upravljanje teh virov strojne opreme.

**Operacijski sistem kot razširjen stroj**

* Arhitektura (nabor ukazov, organizacija pomnilnika, V/I in struktura vodila) večine računalnikov na ravni strojnega jezika je primitivna in nerodna za programiranje, zlasti za vhod/izhod. Če želite to ugotoviti bolj konkretno, razmislite o sodobnih trdih diskih SATA (Serial ATA), ki se uporabljajo v večini računalnikov.
* Del programske opreme, imenovan diskovni gonilnik, ukvarja s strojno opremo in zagotavlja vmesnik za branje in pisanje diskovnih blokov, ne da bi se spuščali v podrobnosti.
* Operacijski sistemi vsebujejo veliko gonilnikov za nadzor V/I naprav.
* Toda tudi ta raven je za večino aplikacij veliko prenizka. Zaradi tega vsi operacijski sistemi nudijo še eno plast abstrakcije za uporabo diskov: datoteke. Z uporabo te abstrakcije lahko programi ustvarjajo, pišejo in berejo datoteke, ne da bi se morali ukvarjati z neurejenimi podrobnostmi o tem, kako strojna oprema dejansko deluje.
* Dobre abstrakcije spremenijo skoraj nemogočo nalogo v dve obvladljivi. Prvi je definiranje in izvajanje abstrakcij. Drugi je uporaba teh abstrakcij za rešitev obravnavanega problema. Ena abstrakcija, ki jo razume skoraj vsak uporabnik računalnika, je datoteka, kot je omenjeno zgoraj. To je koristen del informacij, kot je digitalna fotografija, shranjeno e-poštno sporočilo, pesem ali spletna stran. Veliko lažje je delati s fotografijami, e-pošto, pesmimi in spletnimi stranmi kot s podrobnostmi SATA (ali drugih) diskov.
* Ena glavnih nalog operacijskega sistema je skriti strojno opremo in predstaviti programe (in njihove programerje) z lepimi, čistimi, elegantnimi, doslednimi abstrakcijami za delo. Operacijski sistemi spremenijo grdo v lepo.

**Operacijski sistem kot upravitelj virov**

* Koncept operacijskega sistema, ki predvsem zagotavlja abstrakcije aplikacijskim programom, je pogled od zgoraj navzdol. Alternativni pogled od spodaj navzgor pravi, da je operacijski sistem tam, da upravlja vse dele kompleksnega sistema. Sodobni računalniki so sestavljeni iz procesorjev, pomnilnikov, časovnikov, diskov, miši, omrežnih vmesnikov, tiskalnikov in številnih drugih naprav. V pogledu od spodaj navzgor je naloga operacijskega sistema zagotoviti urejeno in nadzorovano razporeditev procesorjev, pomnilnikov in V/I naprav med različnimi programi, ki jih želijo.
* Skratka, ta pogled na operacijski sistem meni, da je njegova primarna naloga spremljati, kateri programi uporabljajo kateri vir, odobriti zahteve po virih, upoštevati uporabo in posredovati nasprotujoče si zahteve različnih programov in uporabnikov.
* Upravljanje virov vključuje multipleksiranje (skupno rabo) virov na dva različna načina: v času in v prostoru. Ko je vir časovno multipleksiran, ga izmenično uporabljajo različni programi ali uporabniki. Najprej lahko eden od njih uporabi vir, nato drugi itd. Določitev, kako je vir časovno multipleksiran – kdo gre naprej in kako dolgo – je naloga operacijskega sistema. Drug primer časovnega multipleksiranja je souporaba tiskalnika. Ko je več tiskalnih opravil v čakalni vrsti za tiskanje na enem tiskalniku, se je treba odločiti, kateri bo naslednji natisnjen.
* Druga vrsta multipleksiranja je vesoljsko multipleksiranje. Namesto da se stranke izmenjujejo, vsak dobi del vira. Na primer, glavni pomnilnik je običajno razdeljen med več izvajajočih se programov, tako da je lahko vsak rezidenčni hkrati (na primer, da se izmenično uporablja CPE). Drug vir, ki je prostorsko multipleksiran, je disk. V mnogih sistemih lahko en sam disk hkrati hrani datoteke več uporabnikov. Dodeljevanje prostora na disku in spremljanje, kdo uporablja katere diskovne bloke, je tipična naloga operacijskega sistema.

**ZGODOVINA OPERATIVNIH SISTEMOV**

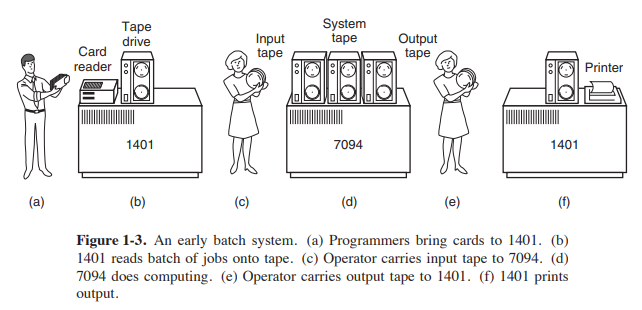
* Prvi pravi digitalni računalnik je zasnoval angleški matematik Charles Babbage (1792–1871). Čeprav je Babbage večino svojega življenja in bogastva poskušal zgraditi svoj "analitični motor", mu nikoli ni uspelo, da bi pravilno deloval, ker je bil izključno mehanski, tehnologija njegovega časa pa ni mogla proizvesti potrebnih koles, zobnikov in zobnikov. na visoko natančnost, ki jo potrebuje. Ni treba posebej poudarjati, da analitični motor ni imel operacijskega sistema. Kot zanimivo zgodovinsko stran je Babbage spoznal, da bo potreboval programsko opremo za svoj analitični stroj, zato je kot prvo programerko na svetu najel mlado žensko po imenu Ada Lovelace, ki je bila hči slavnega britanskega pesnika Lorda Byrona. Po njej je poimenovan programski jezik Ada®.

**Prva generacija (1945–55): vakuumske cevi**

* Po Babbageovih neuspešnih prizadevanjih je bil pri izdelavi digitalnih računalnikov dosežen le majhen napredek do obdobja druge svetovne vojne, kar je spodbudilo eksplozijo dejavnosti. Profesor John Atanasoff in njegov podiplomski študent Clifford Berry sta zgradila to, kar danes velja za prvi delujoči digitalni računalnik na državni univerzi Iowa. Uporabil je 300 vakuumskih cevi. Približno v istem času je Konrad Zuse v Berlinu izdelal računalnik Z3 iz elektromehanskih relejev.

**Druga generacija (1955–65): tranzistorji in šaržni sistemi**

* Uvedba tranzistorja sredi petdesetih let prejšnjega stoletja je korenito spremenila sliko. Računalniki so postali dovolj zanesljivi, da so jih lahko izdelovali in prodajali plačilnim strankam s pričakovanjem, da bodo še naprej delovali dovolj dolgo, da bodo lahko opravili nekaj koristnega dela.
* Glede na visoke stroške opreme ni presenetljivo, da so ljudje hitro iskali načine, kako zmanjšati izgubljen čas. Splošno sprejeta rešitev je bil paketni sistem. Zamisel je bila zbrati pladenj, poln opravil v vhodni sobi in jih nato prebrati na magnetni trak z majhnim (relativno) poceni računalnikom, kot je IBM 1401, ki je bil zelo dober pri branju kartic, kopiranju trakov, in tiskanje, vendar sploh ni dober pri številčnih izračunih. Veliki računalniki druge generacije so se večinoma uporabljali za znanstvene in inženirske izračune, kot je reševanje delnih diferencialnih enačb, ki se pogosto pojavljajo v fiziki in tehniki. Večinoma so bili programirani v FORTRANU in zbirnem jeziku. Tipična operacijska sistema sta bila FMS (Fortran Monitor System) in IBSYS, IBM-ov operacijski sistem za 7094.



**Tretja generacija (1965–1980): IC in večprogramiranje**

* Do zgodnjih šestdesetih let prejšnjega stoletja je imela večina proizvajalcev računalnikov dve različni, nezdružljivi liniji izdelkov. Na eni strani so bili besedno usmerjeni obsežni znanstveni računalniki, kot je 7094, ki so se uporabljali za numerične izračune industrijske moči v znanosti in tehniki. Po drugi strani pa so bili komercialni računalniki, usmerjeni v znake, kot je 1401, ki so jih banke in zavarovalnice pogosto uporabljale za sortiranje in tiskanje trakov. Razvoj in vzdrževanje dveh popolnoma različnih linij izdelkov je bila za proizvajalce draga ponudba.
* IBM je obe težavi poskušal rešiti naenkrat z uvedbo System/360. 360 je bila serija programsko združljivih strojev, ki segajo od modelov velikosti 1401 do veliko večjih, zmogljivejših od mogočnega 7094. Stroji so se razlikovali le po ceni in zmogljivosti (maksimalni pomnilnik, hitrost procesorja, število dovoljenih V/I naprav, in tako naprej). Ker so imeli vsi enako arhitekturo in nabor navodil, so se programi, napisani za en stroj, lahko izvajali na vseh drugih – vsaj v teoriji.
* IBM 360 je bila prva večja računalniška linija, ki je uporabljala (majhne) IC (integrirana vezja), s čimer je zagotovila veliko prednost glede cene in zmogljivosti pred stroji druge generacije, ki so bili zgrajeni iz posameznih tranzistorjev. To je bil takojšen uspeh in idejo o družini združljivih računalnikov so kmalu sprejeli vsi drugi večji proizvajalci.
* Druga pomembna značilnost operacijskih sistemov tretje generacije je bila zmožnost branja opravil s kartic na disk, takoj ko so jih prinesli v računalniško sobo. Potem, ko je bilo delo končano, lahko operacijski sistem naloži novo opravilo z diska na zdaj prazno particijo in ga zažene. Ta tehnika se imenuje spooling (iz Simultaneous Peripheral Operation On Line) in je bila uporabljena tudi za izhod.
* Timesharing, različica večprogramiranja, pri kateri ima vsak uporabnik spletni terminal.
* Prvi splošni sistem časovne delitve, CTSS (Compatible Time Sharing System), je bil razvit pri M.I.T. na posebej modificiranem 7094. Vendar časovna delitev ni postala priljubljena, dokler se potrebna zaščitna strojna oprema ni razširila v tretji generaciji.
* Po uspehu sistema CTSS so se M.I.T., Bell Labs in General Electric (takrat velik proizvajalec računalnikov) odločili, da se lotijo razvoja "računalniškega pripomočka", to je stroja, ki bi podpiral nekaj sto hkratnih uporabnikov časovne delitve. Njihov model je bil električni sistem – ko potrebujete električno energijo, samo vtaknete vtič v steno in v razumnem obsegu bo tam toliko energije, kot jo potrebujete. Oblikovalci tega sistema, znanega kot MULTICS (MULTIPlexed Information and Computing Service), so si zamislili en ogromen stroj, ki bo zagotavljal računalniško moč za vse na območju Bostona.
* MULTICS je bil mešan uspeh. Zasnovan je bil za podporo na stotine uporabnikov na stroju, ki je le nekoliko zmogljivejši od osebnega računalnika s procesorjem Intel 386, čeprav je imel veliko večjo V/I zmogljivosti. To ni tako noro, kot se sliši, saj so v tistih časih ljudje znali pisati majhne, učinkovite programe, veščina, ki je bila pozneje popolnoma izgubljena.
* Kljub pomanjkanju komercialnega uspeha je imel MULTICS velik vpliv na poznejše operacijske sisteme (zlasti UNIX in njegove izpeljanke, FreeBSD, Linux, iOS in Android).
* Drug pomemben razvoj v tretji generaciji je bila fenomenalna rast miniračunalnikov, začenši z DEC PDP-1 leta 1961. PDP-1 je imel le 4K 18-bitnih besed, vendar je znašal 120.000 $ na stroj (manj kot 5 % cene). od 7094).
* Za nekatere vrste neštevilčnega dela je bil skoraj tako hiter kot 7094 in je rodil popolnoma novo industrijo. Hitro mu je sledila vrsta drugih PDP-jev (za razliko od IBM-ove družine, vsi nezdružljivi), ki so dosegli vrhunec v PDP-11. Eden od računalniških znanstvenikov pri Bell Labsu, ki je delal na projektu MULTICS, Ken Thompson, je nato našel majhen miniračunalnik PDP-7, ki ga nihče ni uporabljal, in se odločil napisati zmanjšano različico MULTICS za enega uporabnika. To delo se je pozneje razvilo v operacijski sistem UNIX, ki je postal priljubljen v akademskem svetu, pri vladnih agencijah in številnih podjetjih.

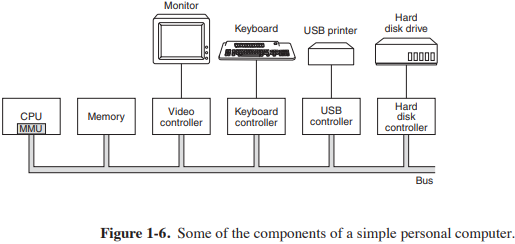
**Četrta generacija (1980 – danes): osebni računalniki**

* Z razvojem vezij LSI (Large Scale Integration) – čipov, ki vsebujejo na tisoče tranzistorjev na kvadratnem centimetru silikona – se je začela doba osebnega računalnika. Leta 1974, ko je Intel izdal 8080, prvi splošnonamenski 8-bitni CPU, je želel operacijski sistem za 8080, delno, da bi ga lahko preizkusil. Intel je enega od svojih svetovalcev Garyja Kildalla prosil, naj ga napiše. Kildall in prijatelj sta najprej izdelala krmilnik za novo izdano 8-palčno disketo Shugart Associates in priklopila disketo na 8080, s čimer sta izdelala prvi mikroračunalnik z diskom. Kildall je nato zanj napisal operacijski sistem, ki temelji na disku, imenovan CP/M (Nadzorni program za mikroračunalnike). Kildall je nato ustanovil podjetje Digital Research.
* V zgodnjih osemdesetih letih je IBM zasnoval IBM PC in iskal programsko opremo, ki bi se na njem izvajala.
* Doug Engelbart je izumil grafični uporabniški vmesnik, skupaj z okni, ikonami, meniji in miško.
* Mac OS X je operacijski sistem, ki temelji na UNIX, čeprav ima zelo značilen vmesnik.
* Ko se je Microsoft odločil zgraditi naslednika MS-DOS, je nanj močno vplival uspeh Macintosha. Proizvedel je sistem, ki temelji na GUI, imenovan Windows, ki je prvotno deloval na vrhu MS-DOS (tj. bil je bolj kot lupina kot pravi operacijski sistem). Približno 10 let, od 1985 do 1995, je bil Windows le grafično okolje na vrhu MS-DOS. Od leta 1995 pa je bila izdana prostostoječa različica Windows 95, ki je vključevala številne funkcije operacijskega sistema, pri čemer je osnovni sistem MS-DOS uporabljal samo za zagon in zagon starih programov MS-DOS. Leta 1998 je bila izdana nekoliko spremenjena različica tega sistema, imenovana Windows 98. Kljub temu sta tako Windows 95 kot Windows 98 še vedno vsebovala veliko količino 16-bitnega Intelovega montažnega jezika. Drugi Microsoftov operacijski sistem, Windows NT (kjer NT pomeni Nova tehnologija), ki je bil na določeni ravni združljiv z Windows 95, vendar je bil interno popolnoma prepisan iz nič. Bil je popoln 32-bitni sistem.
* Leta 2001 je izšla nekoliko nadgrajena različica sistema Windows 2000, imenovana Windows XP. Nato je januarja 2007 Microsoft končno izdal naslednika operacijskega sistema Windows XP, imenovanega Vista. S prihodom sistema Windows 7, nove in veliko manj zahtevne različice operacijskega sistema, se je veliko ljudi odločilo, da v celoti preskočijo Vista. Leta 2012 je Microsoft predstavil svojega naslednika Windows 8, operacijski sistem s popolnoma novim videzom in občutkom, namenjen zaslonom na dotik.
* Drugi glavni tekmec v svetu osebnih računalnikov je UNIX (in njegovi različni derivati). UNIX je najmočnejši na omrežnih in podjetniških strežnikih, vendar je pogosto prisoten tudi na namiznih računalnikih, prenosnih računalnikih, tablicah in pametnih telefonih. Na računalnikih, ki temeljijo na x86, Linux postaja priljubljena alternativa operacijskemu sistemu Windows za študente in vse več poslovnih uporabnikov.
* Procesorji so lahko 32-bitni ali 64-bitni z nekaj ali več jedri in cevovodom, ki so lahko globoki ali plitvi itd. Kjer je razlika pomembna, se bomo namesto tega sklicevali na eksplicitne modele – in uporabili x86-32 in x86-64 za označevanje 32-bitnih in 64-bitnih različic.
* FreeBSD je tudi priljubljena izpeljanka UNIX, ki izvira iz projekta BSD na Berkeleyju. Vsi sodobni računalniki Macintosh uporabljajo spremenjeno različico FreeBSD (OS X).
* Zanimiv razvoj, ki se je začel dogajati sredi osemdesetih let prejšnjega stoletja, je rast omrežij osebnih računalnikov z omrežnimi operacijskimi sistemi in porazdeljenimi operacijskimi sistemi (Tanenbaum in Van Steen, 2007).
* V omrežnem operacijskem sistemu se uporabniki zavedajo obstoja več računalnikov in se lahko prijavijo v oddaljene računalnike in kopirajo datoteke z enega računalnika na drugega. Vsak stroj poganja svoj lokalni operacijski sistem in ima svojega lokalnega uporabnika (ali uporabnike). Omrežni operacijski sistemi se bistveno ne razlikujejo od operacijskih sistemov z enim procesorjem.

**Peta generacija (1990 – danes): Mobilni računalniki**

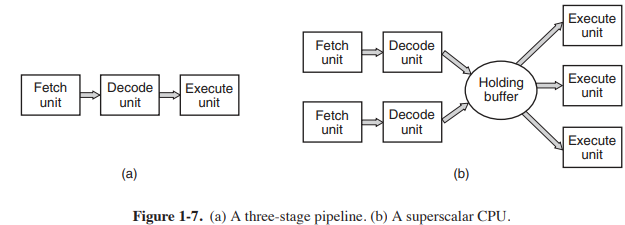
* Odkar se je detektiv Dick Tracy v stripu iz 40. let prejšnjega stoletja začel pogovarjati s svojo "dvosmerno radijsko zapestno uro", so ljudje hrepeneli po komunikacijski napravi, ki bi jo lahko nosili s seboj, kamor koli gredo. Prvi pravi mobilni telefon se je pojavil leta 1946 in je tehtal približno 40 kilogramov. Prvi pravi ročni telefon se je pojavil v sedemdesetih letih prejšnjega stoletja in je bil s približno enim kilogramom pozitivno lahek. Z ljubeznijo je bil znan kot "opeka". Prvi pravi pametni telefon se je pojavil šele sredi devetdesetih let prejšnjega stoletja, ko je Nokia izdala N9000, ki je dobesedno združila dve, večinoma ločeni napravi: telefon in dlančnik (Personal Digital Assistant).

PREGLED RAČUNALNIŠKE STROJNE OPREME



**Procesorji**

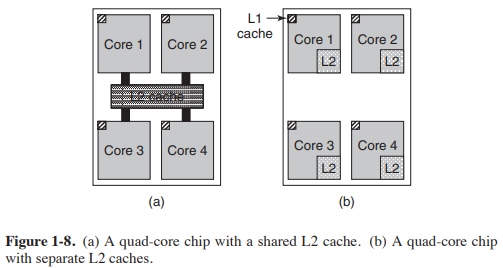
* "Možgani" računalnika so CPU. Pridobi navodila iz pomnilnika in jih izvede. Osnovni cikel vsakega CPE-ja je, da prvo navodilo pridobi iz pomnilnika, ga dekodira, da določi njegov tip in operande, ga izvede, nato pa pridobi, dekodira in izvede naslednja navodila. Cikel se ponavlja, dokler se program ne zaključi. Na ta način se izvajajo programi. Vsak CPU ima določen nabor navodil, ki jih lahko izvede. Ker dostopanje do pomnilnika za pridobitev ukaza ali podatkovne besede traja veliko dlje kot izvajanje ukaza, vsi CPE vsebujejo nekaj registrov za shranjevanje ključnih spremenljivk in začasnih rezultatov. Tako nabor ukazov na splošno vsebuje navodila za nalaganje besede iz pomnilnika v register in shranjevanje besede iz registra v pomnilnik. Druga navodila združujejo dva operanda iz registrov, pomnilnika ali obojega v rezultat, na primer dodajanje dveh besed in shranjevanje rezultata v register ali v pomnilnik.
* Poleg splošnih registrov, ki se uporabljajo za shranjevanje spremenljivk in začasnih rezultatov, ima večina računalnikov več posebnih registrov, ki so vidni programerju. Eden od teh je programski števec, ki vsebuje pomnilniški naslov naslednjega ukaza, ki ga je treba pridobiti. Po pridobitvi tega navodila se programski števec posodobi tako, da kaže na njegovega naslednika. Drug register je kazalec sklada, ki kaže na vrh trenutnega sklada v pomnilniku. Sklad vsebuje en okvir za vsak postopek, ki je bil vnesen, vendar še ni zapustil. Okvir sklada procedure vsebuje tiste vhodne parametre, lokalne spremenljivke in začasne spremenljivke, ki niso shranjene v registrih. Še en register je PSW (Program Status Word). Ta register vsebuje bite pogojne kode, ki so nastavljeni z navodili za primerjavo, prioriteto CPE, načinom (uporabnik ali jedro) in različnimi drugimi krmilnimi biti.
* Cevovod je prikazan na sliki 1-7(a) za cevovod s tremi stopnjami. Pogosti so daljši cevovodi. V večini zasnov cevovoda, ko je bilo navodilo pridobljeno v cevovod, ga je treba izvesti, tudi če je bilo prejšnje navodilo pogojna veja, ki je bila prevzeta.



* Še naprednejši od cevovodne zasnove je superskalarni CPU, prikazan na sliki 1-7(b). V tej zasnovi je prisotnih več izvršilnih enot, na primer ena za celoštevilsko aritmetiko, ena za aritmetiko s plavajočo vejico in ena za logične operacije. Dve ali več navodil se pridobijo naenkrat, dekodirajo in odložijo v zadrževalni medpomnilnik, dokler jih ni mogoče izvesti. Takoj, ko je izvršilna enota na voljo, poišče v zadrževalnem medpomnilniku, ali obstaja navodilo, ki ga lahko obravnava, in če je tako, odstrani ukaz iz medpomnilnika in ga izvede.
* Večina procesorjev, razen zelo preprostih, ki se uporabljajo v vgrajenih sistemih, ima dva načina, način jedra in uporabniški način, kot je bilo že omenjeno. Ponavadi, bit v PSW nadzoruje način. Ko deluje v načinu jedra, lahko CPE izvede vsako navodilo v svojem nizu navodil in uporabi vse funkcije strojne opreme. Na namiznih in strežniških računalnikih operacijski sistem običajno deluje v načinu jedra, kar mu omogoča dostop do celotne strojne opreme.
* Navodilo TRAP preklopi iz uporabniškega načina v način jedra in zažene operacijski sistem. Ko je delo končano, se nadzor vrne uporabniškemu programu po navodilu, ki sledi sistemskemu klicu.

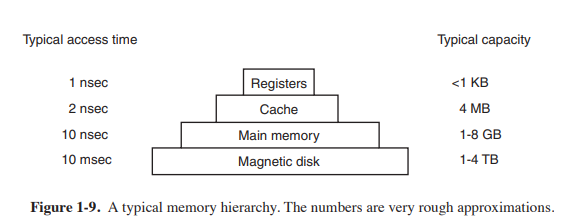
Večnitni in večjedrni čipi (Multithreaded and Multicore Chips)

* Moorov zakon pravi, da se število tranzistorjev na čipu podvoji vsakih 18 mesecev. Ta 'zakon' ni nekakšen zakon fizike, kot je ohranjanje zagona, ampak je opazovanje soustanovitelja Intel Gordona Moora o tem, kako hitro lahko procesni inženirji v polprevodniških podjetjih skrčijo svoje tranzistorje. Moorov zakon velja že več kot tri desetletja in naj bi veljal še vsaj eno. Po tem bo število atomov na tranzistor postalo premajhno in kvantna mehanika bo začela igrati veliko vlogo, ki bo preprečila nadaljnje krčenje velikosti tranzistorja.
* Večnitnost ne ponuja pravega vzporednosti. Naenkrat se izvaja samo en proces, vendar se čas preklopa niti zmanjša na vrstni red nanosekunde. Večnitno delovanje ima posledice za operacijski sistem, ker se vsaka nit operacijskemu sistemu zdi kot ločen CPE. C
* Poleg večnitnosti ima veliko čipov CPU zdaj štiri, osem ali več popolnih procesorjev ali jeder. Večjedrni čipi na sliki 1 8 dejansko nosijo štiri mini čipe, vsak s svojim neodvisnim CPU. Mimogrede, v smislu čistega števila nič ni boljše od sodobne GPU (grafične procesne enote). GPU je procesor z dobesedno na tisoče drobnih jeder. Zelo so dobri za številne majhne izračune, ki se izvajajo vzporedno, kot je upodabljanje poligonov v grafičnih aplikacijah.



**Pomnilnik**

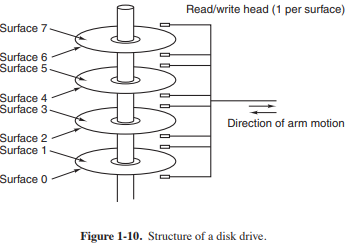
* Druga pomembna komponenta vsakega računalnika je pomnilnik. V idealnem primeru bi moral biti pomnilnik izjemno hiter, obilno velik in poceni. Pomnilniški sistem je zgrajen kot hierarhija plasti, kot je prikazano na sliki 1-9. Zgornje plasti imajo večjo hitrost, manjšo zmogljivost in večjo ceno na bit kot nižje, pogosto za milijardo ali več faktorjev. Zgornji sloj je sestavljen iz notranjih registrov CPE. Narejeni so iz enakega materiala kot CPE in so tako hitri kot CPE. Zato pri dostopu do njih ni zamude. Zmogljivost pomnilnika, ki je na voljo v njih, je običajno 32 × 32 bitov na 32-bitnem CPU in 64 × 64 bitov na 64-bitnem CPU. Manj kot 1 KB v obeh primerih. Programi morajo registre upravljati sami, v programski opremi.



* Sledi predpomnilnik, ki ga večinoma nadzoruje strojna oprema. Glavni pomnilnik je razdeljen na vrstice predpomnilnika, običajno 64 bajtov, z naslovi od 0 do 63 v vrstici predpomnilnika 0, od 64 do 127 v vrstici predpomnilnika 1 itd. Največkrat uporabljene vrstice predpomnilnika so shranjene v hitrem predpomnilniku, ki se nahaja znotraj ali zelo blizu CPE.
* Pri večjedrnih čipih se morajo oblikovalci odločiti, kam bodo namestili predpomnilnike. Na sliki 1-8(a) si vsa jedra delijo en sam predpomnilnik L2. Ta pristop se uporablja v Intelovih večjedrnih čipih. Nasprotno pa ima na sliki 18(b) vsako jedro svoj predpomnilnik L2.
* Glavni pomnilnik je naslednji v hierarhiji na sliki 1-9. To je delovni konj pomnilniškega sistema. Glavni pomnilnik se običajno imenuje RAM (Random Access Memory). Stari ljudje ga včasih imenujejo jedrni pomnilnik, ker so računalniki v 1950-ih in 1960-ih uporabljali drobna feritna jedra, ki jih je mogoče magnetizirati za glavni pomnilnik.
* Poleg glavnega pomnilnika ima veliko računalnikov majhno količino nehlapnega pomnilnika z naključnim dostopom. Za razliko od RAM-a, trajni pomnilnik ne izgubi svoje vsebine, ko je napajanje izklopljeno. ROM (pomnilnik samo za branje) je tovarniško programiran in ga pozneje ni mogoče spremeniti. Je hiter in poceni.
* EEPROM (Electrally Erasable PROM) in bliskovni pomnilnik sta prav tako nehlapna, vendar ju je v nasprotju s ROM-om mogoče izbrisati in prepisati. Vendar pa njihovo pisanje traja veliko več časa kot pisanje RAM-a, zato se uporabljajo na enak način kot ROM, le z dodatno funkcijo, da je zdaj mogoče popraviti napake v programih, ki jih imajo, tako da jih prepišejo na terenu.
* Yet another kind of memory is CMOS, which is volatile. Many computers use CMOS memory to hold the current time and date.. The CMOS memory can also hold the configuration parameters, such as which disk to boot from. CMOS is used because it draws so little power that the original factory installed battery often lasts for several years.

Diski

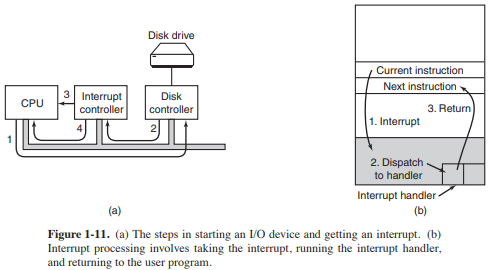
* Naslednji v hierarhiji je magnetni disk (trdi disk). Pomnilnik na disku je dva reda velikosti cenejši od RAM-a na bit in pogosto tudi dva reda velikosti večji. Edina težava je, da je čas naključnega dostopa do podatkov na njem skoraj tri rede velikosti počasnejši. Razlog je v tem, da je disk mehanska naprava, kot je prikazano na sliki 1-10.



* Disk je sestavljen iz ene ali več kovinskih plošč, ki se vrtijo pri 5400, 7200, 10.800 RPM ali več. Mehanska roka se vrti nad ploščami iz vogala, podobno kot dvigalka na starem fonografu s 33 RPM za predvajanje vinilnih plošč. Informacije so zapisane na disk v nizu koncentričnih krogov. V katerem koli položaju roke lahko vsaka od glav prebere obročasto območje, imenovano steza. Vsi tiri za določen položaj roke skupaj tvorijo cilinder. Vsaka skladba je razdeljena na določeno število sektorjev, običajno 512 bajtov na sektor.
* (Solid State Disks)- SSD diski nimajo gibljivih delov, ne vsebujejo plošč v obliki diskov in shranjujejo podatke v (Flash) pomnilniku. Edini način, po katerem so podobni diskom, je, da shranijo tudi veliko podatkov, ki se ne izgubijo, ko je napajanje izklopljeno.

**V/I naprave**

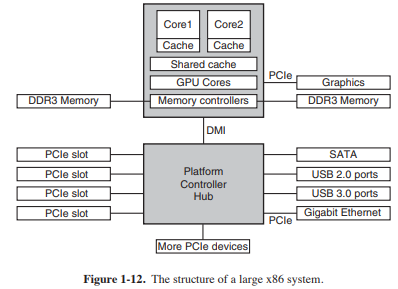
* CPE in pomnilnik nista edina sredstva, ki jih mora upravljati operacijski sistem. V/I naprave prav tako močno sodelujejo z operacijskim sistemom. V/I naprave so običajno sestavljene iz dveh delov: krmilnika in same naprave. Krmilnik je čip ali niz čipov, ki fizično upravljajo napravo. Sprejema ukaze operacijskega sistema, na primer za branje podatkov iz naprave, in jih izvaja. Za opravljanje vsega tega dela krmilniki pogosto vsebujejo majhne vgrajene računalnike, ki so programirani za opravljanje svojega dela.



* Na sliki 1-11(a) vidimo tristopenjski postopek za V/I. V 1. koraku gonilnik pove krmilniku, kaj naj naredi, tako da vpiše v registre naprav. Krmilnik nato zažene napravo. Ko krmilnik konča z branjem ali pisanjem števila bajtov, ki mu je bilo rečeno, da prenese, signalizira čip krmilnika prekinitve z uporabo določenih vodil vodila v koraku 2. Če je krmilnik prekinitve pripravljen sprejeti prekinitev (kar morda ne bo, če je zaposlen z obdelavo tistega z višjo prioriteto), v 3. koraku uveljavlja pin na čipu CPE, ki mu to sporoči. V 4. koraku krmilnik prekinitev postavi številko naprave na vodilo, tako da jo lahko CPE prebere in ve katera naprava je pravkar končala (več naprav se lahko izvaja hkrati).
* Ko se CPU odloči za prekinitev, se števec programov in PSW običajno potisneta v trenutni sklad in CPU preklopi v način jedra. Številka naprave se lahko uporabi kot indeks v delu pomnilnika, da se poišče naslov prekinitvenega obdelovalca za to napravo. Ta del pomnilnika se imenuje prekinitveni vektor.
* Tretja metoda za izvajanje V/I uporablja posebno strojno opremo: DMA (Direct Memory Access) čip, ki lahko nadzoruje pretok bitov med pomnilnikom in nekaterim krmilnikom brez nenehnega posredovanja CPE. CPU nastavi čip DMA in mu pove, koliko bajtov naj prenese, zadevne naslove naprave in pomnilnika ter smer ter ga spusti. Ko je DMA čip končan, povzroči prekinitev, ki se obravnava, kot je opisano zgoraj.

**Vodila (Buses)**

* Organizacija na sliki 1-6 je bila leta uporabljena na miniračunalnikih in tudi na originalnem IBM PC-ju. Ker pa so postajali procesorji in pomnilniki hitrejši, je bila zmožnost enega vodila (in zagotovo vodila IBM PC), da obvladuje ves promet, obremenjena do točke preloma. Nekaj je bilo treba dati. Posledično so bila dodana dodatna vodila, tako za hitrejše V/I naprave kot za promet med CPE in pomnilnikom. Kot posledica te evolucije je velik sistem x86 trenutno videti podobno kot na sliki 1-12.



* Ta sistem ima veliko vodil (npr. predpomnilnik, pomnilnik, PCIe, PCI, USB, SATA in DMI), vsako z drugačno hitrostjo prenosa in funkcijo. Glavno vodilo je vodilo PCIe (Peripheral Component Interconnect Express). PCIe je sposoben prenašati desetine gigabitov na sekundo, zato je veliko hitrejši od svojih predhodnikov.
* USB (Universal Serial Bus) je bil izumljen za priključitev vseh počasnih V/I naprav, kot sta tipkovnica in miška, na računalnik. USB uporablja majhen konektor s štirimi do enajstimi žicami (odvisno od različice), od katerih nekatere napajajo naprave USB ali se povezujejo z ozemljitvijo.
* Vodilo SCSI (Small Computer System Interface) je visoko zmogljivo vodilo, namenjeno hitrim diskom, optičnim bralnikom in drugim napravam, ki potrebujejo veliko pasovno širino. Lahko delujejo s hitrostjo do 640 MB/s.

**Zagon računalnika**

* Zelo na kratko je postopek zagona naslednji. Vsak računalnik vsebuje starševsko ploščo. Na starševski plošči je program, imenovan sistemski BIOS (Basic Input Output System). BIOS vsebuje programsko opremo za V/I nizke ravni, vključno s postopki za branje tipkovnice, zapisovanje na zaslon in izvajanje diskovnega V/I med drugim.
* Ko se računalnik zažene, se zažene BIOS. Najprej preveri, koliko RAM-a je nameščenega in ali so tipkovnica in druge osnovne naprave nameščene in se pravilno odzivajo. Začne se s skeniranjem vodil PCIe in PCI, da zazna vse naprave, ki so nanje priključene. Če se prisotne naprave razlikujejo od tistih ob zadnjem zagonu sistema, so nove naprave konfigurirane.
* BIOS nato določi zagonsko napravo tako, da preizkusi seznam naprav, shranjenih v pomnilniku CMOS. Običajno se poskuša zagnati s pogona CD-ROM (ali včasih USB), če je prisoten. Če to ne uspe, se sistem zažene s trdega diska. Prvi sektor iz zagonske naprave se prebere v pomnilnik in izvede.
* Ta sektor vsebuje program, ki običajno pregleda tabelo particij na koncu zagonskega sektorja, da ugotovi, katera particija je aktivna. Nato se s te particije prebere sekundarni zagonski nalagalnik. Ta nalagalnik bere v operacijski sistem z aktivne particije in ga zažene.
* Operacijski sistem nato poizveduje v BIOS-u, da dobi informacije o konfiguraciji. Za vsako napravo preveri, ali ima gonilnik naprave. Če ne, uporabnika zahteva, da vstavi CD-ROM z gonilnikom (ki ga priskrbi proizvajalec naprave) ali da ga prenese z interneta. Ko ima vse gonilnike naprav, jih operacijski sistem naloži v jedro. Nato inicializira svoje tabele, ustvari vse potrebne procese v ozadju in zažene program za prijavo ali GUI.

**ZOO OPERACIJSKI SISTEM**

**Operacijski sistemi velikih računalnikov (Mainframe Operating Systems)**

* Na vrhuncu so operacijski sistemi za velike računalnike, računalnike velikosti prostora, ki jih še vedno najdemo v večjih korporativnih podatkovnih centrih. Ti računalniki se od osebnih računalnikov razlikujejo po zmogljivosti V/I.
* Operacijski sistemi za velike računalnike so močno usmerjeni v obdelavo številnih opravil hkrati, od katerih večina potrebuje ogromne količine V/I. Običajno ponujajo tri vrste storitev: paketno obdelavo, obdelavo transakcij in časovno delitev.
* Sistemi za obdelavo transakcij obravnavajo veliko število majhnih zahtev, na primer obdelava čekov pri banki ali rezervacijah letalske družbe. Vsaka enota dela je majhna, vendar mora sistem obdelati stotine ali tisoče na sekundo. Sistemi časovne delitve omogočajo več oddaljenim uporabnikom, da hkrati izvajajo opravila v računalniku, na primer poizvedovanje po veliki bazi podatkov. Te funkcije so tesno povezane; veliki operacijski sistemi pogosto izvajajo vse. Primer operacijskega sistema velikega računalnika je OS/390, potomec OS/360. Vendar pa glavne operacijske sisteme postopoma nadomeščajo različice UNIX, kot je Linux.

**Strežniški operacijski sistemi (Server Operating Systems)**

* Eno raven nižje so strežniški operacijski sistemi. Delujejo na strežnikih, ki so bodisi zelo veliki osebni računalniki, delovne postaje ali celo veliki računalniki. Služijo več uporabnikom hkrati prek omrežja in uporabnikom omogočajo souporabo virov strojne in programske opreme.
* Strežniki lahko nudijo storitev tiskanja, datotečno storitev ali spletno storitev. Internetni ponudniki uporabljajo številne strežniške stroje za podporo svojim strankam in
* Spletna mesta uporabljajo strežnike za shranjevanje spletnih strani in obdelavo dohodnih zahtev. Tipični strežniški operacijski sistemi so Solaris, FreeBSD, Linux in Windows Server 201x.

**Večprocesorski operacijski sistemi (Multiprocessor Operating Systems)**

* Vse pogostejši način za pridobitev računalniške moči v višji ligi je povezovanje več CPE-jev v en sam sistem. Glede na to, kako so povezani in kaj je v skupni rabi, se ti sistemi imenujejo vzporedni računalniki, večračunalniki ali večprocesorji.
* Potrebujejo posebne operacijske sisteme, pogosto pa so to različice strežniških operacijskih sistemov s posebnimi funkcijami za komunikacijo, povezljivost in doslednost. Z nedavnim prihodom večjedrnih čipov za osebne računalnike se tudi običajni namizni in prenosni operacijski sistemi začenjajo ukvarjati z vsaj manjšimi večprocesorji in število jeder se bo verjetno sčasoma povečalo. Številni priljubljeni operacijski sistemi, vključno z Windows in Linux, delujejo na večprocesorjih.

**Operacijski sistemi za osebni računalnik (Personal Computer Operating Systems)**

* Naslednja kategorija je operacijski sistem osebnega računalnika. Sodobni vsi podpirajo večprogramiranje, pogosto z več desetimi programi, ki se zaženejo ob zagonu.
* Njihova naloga je zagotoviti dobro podporo posameznemu uporabniku. Pogosto se uporabljajo za obdelavo besedil, preglednice, igre in dostop do interneta.
* Pogosti primeri so Linux, FreeBSD, Windows 7, Windows 8 in Appleov OS X.

**Operacijski sistemi za ročni računalnik (Handheld Computer Operating Systems)**

* Če nadaljujemo do vse manjših sistemov, pridemo do tablic, pametnih telefonov in drugih ročnih računalnikov. Ročni računalnik, prvotno znan kot PDA (Personal Digital Assistant), je majhen računalnik, ki ga lahko med delovanjem držite v roki.
* Najbolj znani primeri so pametni telefoni in tablični računalniki. Večina teh naprav se ponaša z večjedrnimi procesorji, GPS-om, kamerami in drugimi senzorji, obilno količino pomnilnika in izpopolnjenimi operacijskimi sistemi.

**Vgrajeni operacijski sistemi (Embedded Operating Systems)**

* Vgrajeni sistemi delujejo na računalnikih, ki nadzorujejo naprave, ki se na splošno ne štejejo za računalnike in ki ne sprejemajo uporabniško nameščene programske opreme.
* Tipični primeri so mikrovalovne pečice, TV sprejemniki, avtomobili, DVD snemalniki, tradicionalni telefoni in MP3 predvajalniki. Glavna lastnost, po kateri se vgrajeni sistemi razlikujejo od prenosnikov, je gotovost, da se na njem ne bo nikoli izvajala nobena nezaupanja vredna programska oprema.

**Operacijski sistemi senzorskega vozlišča(Sensor-Node Operating Systems)**

* Omrežja majhnih senzorskih vozlišč se uporabljajo za številne namene. Ta vozlišča so majhni računalniki, ki komunicirajo med seboj in z bazno postajo z uporabo brezžične komunikacije.
* Senzorska omrežja se uporabljajo za zaščito oboda stavb, varovanje državnih meja, odkrivanje požarov v gozdovih, merjenje temperature in padavin za vremensko napoved, zbiranje informacij o sovražnikovih premikih na bojiščih in še veliko več.
* Vsako senzorsko vozlišče je pravi računalnik s CPU, RAM, ROM in enim ali več okoljskimi senzorji. Poganja majhen, a resničen operacijski sistem, običajno tisti, ki ga poganjajo dogodki, ki se odziva na zunanje dogodke ali občasno izvaja meritve na podlagi notranje ure. Operacijski sistem mora biti majhen in preprost, ker imajo vozlišča malo RAM-a, življenjska doba baterije pa je velika težava. TinyOS je dobro znan operacijski sistem za senzorsko vozlišče.

**Operacijski sistemi v realnem času (Real-Time Operating Systems)**

* Druga vrsta operacijskega sistema je sistem v realnem času. Za te sisteme je značilno, da imajo čas kot ključni parameter. Pogosto obstajajo strogi roki, ki jih je treba izpolniti. Če, na primer, varilni robot vari prezgodaj ali prepozno, bo avto uničen. Veliko teh najdemo v industrijskih procesih, letalski elektroniki, vojaški in podobnih področjih uporabe. Ti sistemi morajo zagotavljati absolutna jamstva, da se bo določeno dejanje zgodilo do določenega časa.
* Mehki sistem v realnem času je tisti, pri katerem je zamuda občasnega roka, čeprav ni zaželena, sprejemljiva in ne povzroča trajne škode. V to kategorijo spadajo digitalni avdio ali multimedijski sistemi. Pametni telefoni so tudi mehki sistemi v realnem času.
* Ker je v (trdih) sistemih v realnem času izpolnjevanje rokov ključnega pomena, je včasih operacijski sistem preprosto knjižnica, ki je povezana z aplikacijskimi programi, kjer je vse tesno povezano in brez zaščite med deli sistema. Primer te vrste sistema v realnem času je eCos.

**Operacijski sistemi s pametnimi karticami (Smart Card Operating Systems)**

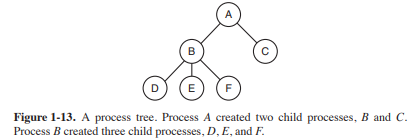
* Najmanjši operacijski sistemi delujejo na pametnih karticah, ki so naprave velikosti kreditne kartice, ki vsebujejo čip CPE. Imajo zelo hude procesorske moči in pomnilnika. Nekateri od njih lahko opravljajo samo eno funkcijo, na primer elektronska plačila, drugi pa lahko opravljajo več funkcij. Pogosto so to lastniški sistemi.

**KONCEPTI OPERATIVNEGA SISTEMA**

* Večina operacijskih sistemov ponuja določene osnovne koncepte in abstrakcije, kot so procesi, naslovni prostori in datoteke, ki so osrednjega pomena za njihovo razumevanje.

**Procesi**

* Ključni koncept v vseh operacijskih sistemih je proces. Proces je v bistvu program v izvajanju. Proces je v bistvu vsebnik, ki vsebuje vse informacije, potrebne za zagon programa. Z vsakim procesom je povezan njegov naslovni prostor, seznam pomnilniških lokacij od 0 do nekega maksimuma, ki jih proces lahko bere in zapiše.
* Naslovni prostor vsebuje izvedljivi program, podatke programa in njegov sklad.
* V mnogih operacijskih sistemih so vse informacije o vsakem procesu, razen vsebine njegovega lastnega naslovnega prostora, shranjene v tabeli operacijskega sistema, imenovani tabela procesov, ki je niz struktur, ena za vsak proces, ki trenutno obstaja.
* Tako je (suspendovan) proces sestavljen iz naslovnega prostora, ki se običajno imenuje jedrna slika (v čast spominov na magnetno jedro, ki so se uporabljali v preteklih dneh), in vnosa v tabelo procesov, ki vsebuje vsebino njegovih registrov in številne druge elemente. potreben za kasnejši ponovni zagon postopka.
* Proces, imenovan tolmač ukazov ali lupina, bere ukaze s terminala.



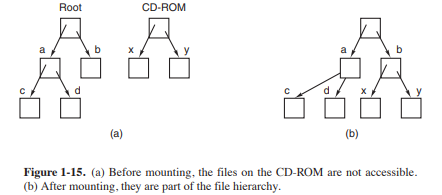
* Signali so programski analog prekinitev strojne opreme in jih poleg izteka časovnikov lahko ustvarijo različni vzroki.
* Vsaki osebi, pooblaščeni za uporabo sistema, administrator sistema dodeli UID (User IDentification). Vsak začet proces ima UID osebe, ki ga je začela. Uporabniki so lahko člani skupin, od katerih ima vsaka GID (Group IDentification).

**Naslovni prostori**

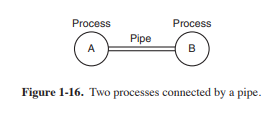
* V bistvu operacijski sistem ustvari abstrakcijo naslovnega prostora kot nabor naslovov, na katere se lahko sklicuje proces. Naslovni prostor je ločen od fizičnega pomnilnika naprave in je lahko večji ali manjši od fizičnega pomnilnika.

**Datoteke**

* Drug ključni koncept, ki ga podpirajo skoraj vsi operacijski sistemi, je datotečni sistem. Sistemski klici so očitno potrebni za ustvarjanje datotek, odstranjevanje datotek, branje datotek in pisanje datotek. Preden datoteko lahko preberete, jo je treba nahajati na disku in odpreti, po branju pa jo zapreti, zato so za te stvari zagotovljeni klici.
* Da bi zagotovili prostor za shranjevanje datotek, ima večina operacijskih sistemov za osebne računalnike koncept imenika kot načina združevanja datotek v skupine.
* Vsako datoteko znotraj hierarhije imenika lahko podate tako, da navedete njeno ime poti z vrha hierarhije imenika, korenskega imenika.
* Preden datoteko lahko preberete ali zapišete, jo je treba odpreti, pri čemer se preverijo dovoljenja. Če je dostop dovoljen, sistem vrne majhno celo število, imenovano deskriptor datoteke, za uporabo v naslednjih operacijah. Če je dostop prepovedan, se vrne koda napake.
* Drug pomemben koncept v UNIX-u je nameščen datotečni sistem.



* Drug pomemben koncept v UNIX-u je posebna datoteka. Na voljo so posebne datoteke, da bi V/I naprave izgledale kot datoteke. Na ta način jih je mogoče brati in pisati z istimi sistemskimi klici, kot se uporabljajo za branje in pisanje datotek. Obstajata dve vrsti posebnih datotek: posebne datoteke blokov in datoteke s posebnimi znaki.
* Posebne datoteke blokov se uporabljajo za modeliranje naprav, ki so sestavljene iz zbirke naključno naslovljivih blokov, kot so diski.
* Datoteke posebnih znakov se uporabljajo za modeliranje tiskalnikov, modemov in drugih naprav, ki sprejemajo ali oddajajo tok znakov.
* Po dogovoru se posebne datoteke hranijo v imeniku /dev.
* Cev je neke vrste psevdofile, ki se lahko uporablja za povezavo dveh procesov.
* Če želita procesa A in B govoriti z uporabo cevi, jo morata nastaviti vnaprej. Ko želi proces A poslati podatke procesu B, piše na cev, kot da bi bila izhodna datoteka.



**Vhod/Izhod**

* Vsi računalniki imajo fizične naprave za pridobivanje vhoda in proizvodnjo izhodnih podatkov.
* Obstaja veliko vrst vhodnih in izhodnih naprav, vključno s tipkovnicami, monitorji, tiskalniki itd.
* Vsak operacijski sistem ima V/I podsistem za upravljanje svojih V/I naprav.

**Zaščita (Protection)**

* Računalniki vsebujejo velike količine informacij, ki jih uporabniki pogosto želijo zaščititi in ohraniti zaupne. Operacijski sistem mora upravljati varnost sistema, tako da so datoteke, na primer, dostopne samo pooblaščenim uporabnikom.
* Datoteke v UNIX so zaščitene tako, da se vsaki dodeli 9-bitna binarna zaščitna koda. Zaščitna koda je sestavljena iz treh 3-bitnih polj, enega za lastnika, enega za ostale člane lastnikove skupine (uporabnike razdeli v skupine skrbnik sistema) in enega za vse ostale. Vsako polje ima bit za dostop za branje, bit za dostop za pisanje in bit za dostop za izvajanje. Ti 3 biti so znani kot rwx bit.

The Shell - Lupina

* V tem razdelku si bomo v tem razdelku na kratko ogledali tolmač ukazov UNIX, lupino. Čeprav ni del operacijskega sistema, močno uporablja številne funkcije operacijskega sistema in tako služi kot dober primer uporabe sistemskih klicev. Je tudi glavni vmesnik med uporabnikom, ki sedi na svojem terminalu, in operacijskim sistemom, razen če uporabnik uporablja grafični uporabniški vmesnik. Obstaja veliko lupin, vključno s sh, csh, ksh in bash.
* Ko se kateri koli uporabnik prijavi, se zažene lupina. Lupina ima terminal kot standardni vhod in standardni izhod. Začne se tako, da vnesete poziv, znak, kot je znak za dolar, ki uporabniku pove, da lupina čaka, da sprejme ukaz.

**Zaščitna strojna oprema (Protection Hardware)**

* Z uvedbo IBM 360 je postala na voljo primitivna oblika zaščite strojne opreme. Ti stroji bi lahko nato shranili več programov v pomnilniku hkrati in jim omogočili, da se izmenično izvajajo (večprogramiranje).

**Diski – Disks**

* IBM je prvi trdi disk – RAMAC (RAndoM ACcess) predstavil leta 1956. Zavzemal je približno 4 kvadratne metre talne površine in je lahko shranil 5 milijonov 7-bitnih znakov, kar je dovolj za eno digitalno fotografijo srednje ločljivosti.
* Ko so prišli v uporabo miniračunalniki, so sčasoma imeli tudi trde diske. Standardni disk na PDP-11, ko je bil predstavljen leta 1970, je bil disk RK05 s kapaciteto 2,5 MB, približno polovico IBM-ovega RAMAC-a, vendar je bil le okoli 40 cm v premeru in 5 cm v višino. Toda tudi sprva je imel enonivojski imenik.
* Ko so se pojavili mikroračunalniki, je bil sprva prevladujoči operacijski sistem CP/M, ki je tudi podpiral samo en imenik na (disketi) disku.

**Navidezni pomnilnik – Virtual Memory**

* Navidezni pomnilnik omogoča zagon programov, večjih od fizičnega pomnilnika naprave, s hitrim premikanjem kosov naprej in nazaj med RAM-om in diskom. Doživel je podoben razvoj, najprej se je pojavil na velikih računalnikih, nato pa se je preselil na mini in mikro.