

Operativni sistem – definicija i funkcije

Operativni sistem (u daljem tekstu OS) objedinjuje raznorodne delove računara u skladnu celinu i sakriva od korisnika detalje funkcionisanja ovih delova koji nisu bitni za korišćenje računara.

OS radi sledeće:

- Upravlja programima, podacima i delovima od kojih se računar sastoji (procesor, kontroleri, radna memorija), s ciljem da oni budu što celishodnije upotrebljeni
- Obezbeđuje pristupačno radno okruženje za krajnjeg korisnika računara

U opštem smislu, operativni sistem se može definisati kao skup programa koji upravljaju resursima računarskog sistema i obezbeđuje interfejs ka korisniku.

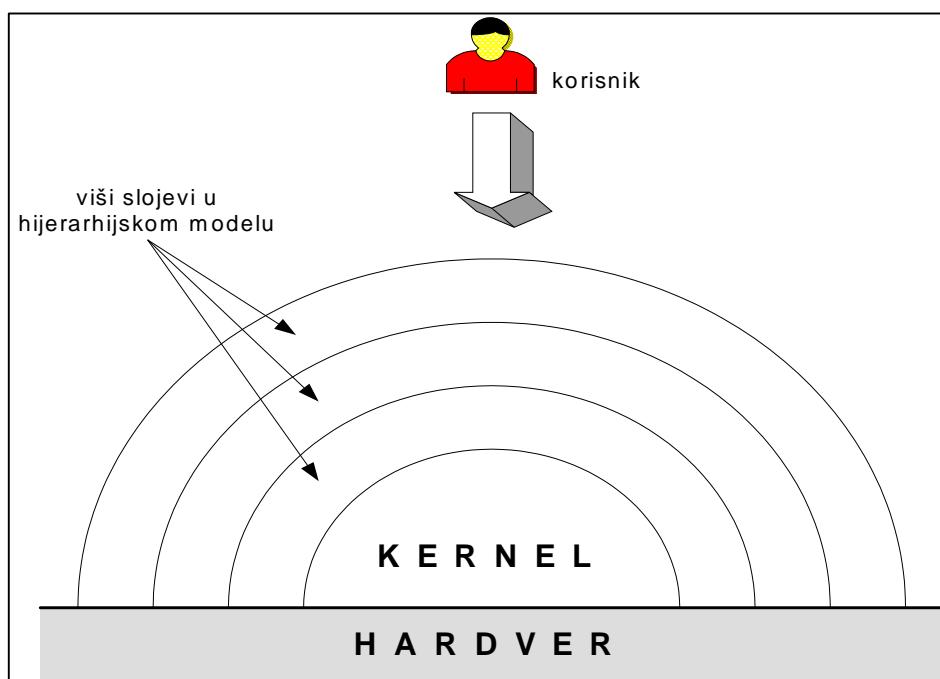
Prva funkcija OS je upravljanje resursima računara (*resource management*). Pod pojmom resurs podrazumevamo sve što je programu potrebno za rad. Resursi mogu biti hardverski (procesor, memorija, I/O uređaji) i softverski (programi, podaci datoteke). Zadatak OS je da vodi računa o resursima računara tj. da zadovolji potrebe programa, da prati koji program koristi koje resurse itd.

Ukratko rečeno, OS je skup sistemskih programa koji posreduju između korisnika računara i računarskog hardvera a cilj je da:

- Izvršava korisničke programe i olakšava rešavanje korisničkih problema
- Korišćenje računarskog sistema učini podesnjim za korisnika
- Omogući što efikasnije iskorišćenje računarskog hardvera

OS se sastoji od više relativno nezavisnih celina. Svaki proizvođač računara ima svoje operativne sisteme, pa je teško da ti opštu strukturu OS. Hijerarhijski model OS je sastavljen od nivoa. Hijerarhijski model ima sledeći smisao: na posmatranom nivou OS-a mogu se zahtevati usluge samo od njegovih nižih nivoa, a nikako od viših. Najniži sloj je poznat kao jezgro OS-a (*kernel*).

Zbog odnosa veličine OS-a i radne memorije, većina OS-a ne može da stane u radnu memoriju. Zato se u memoriji uvek nalaze samo najvažniji delovi OS-a, takozvani rezidentni delovi, koji aktiviraju i izvršavaju korišničke programe, dodeljuju memoriju i datoteke i obavljaju I/O operacije. Rezidentni deo OS-a mora da podržava mehanizam prekida, jer je on osnova višeprogramskog rada. Deo OS-a koji uvek mora da se nalazi u memoriji obično se naziva jezgro ili kernel. Funkcije koje koriste svi nivoi moraju se smestiti u jezgro OS-a. Ostali delovi se ubacuju u memoriju kada su potrebni i izbacuju kada više nisu potrebni.



Slika 1. Hijerarhijski model OS-a

Vrste operativnih sistema

Postoje brojne podele OS-a na osnovu različitih kriterijuma: prema broju korisnika i/ili procesa, prema načinu obrade poslova, prema distribuciji procesorske snage i ostalih resursa, prema nameni i funkcionalnim osobinama.

Klasifikacija prema broju korisnika i procesa

Prema broju korisnika, OS se dele na:

- jednokorisničke (singleuser)
- višekorisničke (multiuser)

Jednokorisnički sistemi obezbeđuju virtualnu mašinu za samo jednog korisnika. To su računarski sistemi prilagođeni za jednu funkciju ili je reč o slabijim i jeftinim konfiguracijama tipa mikroračunara. Uglavnom ih karakteriše jeftin hardver, solidna prateća programska podrška, jednostavan sistem datoteka, jednostavan U/I sistem.

Višekorisnički sistemi su kvalitetni OS koji zahtevaju jače hardverske konfiguracije. Tipičan višekorisnički OS je UNIX, koji obezbeđuje simultani pristup za više korisnika istovremeno, pri čemu korisnici pristupaju sistemu preko posebnih terminala.

Prema broju simultanih aktivnosti tj. prema broju procesa koji se mogu izvršavati paralelno ili kvaziparalelno, operativni sistemi se mogu podeliti na:

- jednoprocesne (*singletasking, singleprocess*)
- višeprocesne (*multitasking, multiprocess*)

Na osnovu kombinovanog kriterijuma mogu se izdvojiti tri vrste operativnih sistema:

- jednokorisnički jednoprocesni (*single-user, singletasking*) kao što je MS-DOS
- jednokorisnički višeprocesni (*single-user, multitasking*) kao što su OS/2 i MS Windows 3.1/9x/Me
- višekorisnički višeprocesni (*multiuser, multitasking*) kao što je UNIX, ali uslovno se mogu prihvati i MS Windows 2000/XP/2003 ukoliko obezbeđuju terminalske usluge

Klasifikacija prema nameni i načinu obrade poslova

Prema nameni, operativni sistemi se dele na:

- OS opšte namene
- OS specijalne namene (služe za upravljanje procesima)

Prema načinu obrade poslova OS se dele na:

- Sistemi sa grupnom obradom (*batch*)
- Interaktivni sistemi
- Kombinovani sistemi

Sistemi sa grupnom obradom (*batch*)

Grupna (serijska, paketna) obrada je takav način rada računara u kome korisnici predaju svoje poslove na izvršenje posredstvom ulaznih jedinica i koji se zatim odvijaju jedan za drugim u nizu, pri čemu korisnik nema mogućnost komuniciranja sa svojim posлом

Interaktivni sistem

Interaktivne sisteme (**time-sharing** sistemi) karakteriše postojanje terminala za svakog korisnika, preko kojih korisnici zadaju zadatke i poslove i komuniciraju sa svojim poslovima. Paralelnost u radu se postiže tako što se svakom korisničkom programu dodeljuje jedan kvantum vremena centralnog procesora, pa se na svaku poruku korisnika odaziva u roku od nekoliko sekundi. Po isteku vremenskog kvantuma dodeljenog jednom procesu, on se prekida, bilo da je završio sa radom ili nije, a procesor se dodeljuje sledećem procesu u redu čekanja.

Kombinovani sistem

Kombinovane sisteme karakteriše mogućnost istovremenog obavljanja interaktivnih poslova i paketne obrade. Npr korisnik može u pozadini pokrenuti nekoliko vremenski zahtevnih poslova koji ne zahtevaju interakciju sa njim, a ok čeka može čitati elektronsku poštu ili Web stranice.

Klasifikacija prema funkcionalnim osobinama računarskog sistema

Prema funkcionalnim osobinama računarskog sistema za koji su namenjeni, OS se dele u sledeće kategorije:

- OS za velike računarske sisteme (*mainframe systems*)
- OS za sisteme sa deljenim vremenom (*time-sharing systems*)
- OS za stolne računare (*desktop systems*)
- OS za višeprocesorske sisteme (*multiprocessor systems*)
- Mrežni operativni sistemi (*network OS*)
- Distribuirani sistemi (*distributed systems*)
- OS za upravljanje u realnom vremenu (*real-time systems*)

OS za velike računarske sisteme

Na velikim računarskim sistemima (*mainframe system*) prvi put su se pokretale mnoge komercijalne i naučne aplikacije. U svoje vreme bile su to fizički ogromne mašine kojima se upravljalo preko konzole. Zajednički ulazni uređaji su bili čitači bušenih kartica i magnetne trake, a izlazni takođe bušene kartice, trake ili štampači. Uvodi se tehnika multiprogramiranja, a sa njom i automatsko sekvenciranje poslova, koje bez operatera prebacuje kontrolu sa jednog posla na drugi. Da bi OS mogao da funkcioniše, njegov osnovni deo mora da bude uvek u memoriji. Taj deo bira posao koji će se trenutno izvršavati. Po završetku posla ili nakon blokade zbog čekanja na I/O operacije, OS opet preuzima upravljanje i dodeljuje kontrolu drugom poslu.

OS sistemi za sisteme sa deljenim vremenom

Sistemi sa deljenim vremenom nastali su kao specijalna klasa velikih sistema u kojoj je svakom korisniku omogućena *on-line* komunikacija sa svojim poslom i operativnim sistemom. Vreme odziva se uvodi kao kriterijum za ocenu efikasnosti ove klase sistema. Opet imamo tehniku multiprogramiranja - što znači više poslova koji se u memoriji nalaze istovremeno – i tehniku deljenja vremena, gde posle isteka vremenskog kvantuma svaki posao mora da preda kontrolu drugom poslu i sačeka svoje novo vreme. Ove sisteme delimo na dve posebne klase: **upitne sisteme (file interrogation systems)** i **transakcione sisteme (transaction processing)**

Kod upitnog sistema, kao što su razni informacioni sistemi koji rade sa velikim bazama podataka, veoma je bitno da vreme odziva za postavljeni upit bude veoma kratko. Transakcioni sistemi su relativno slični upitnim s tim što se kod njih često (npr nekoliko puta u sekundi) modifikuju baze podataka. Primer su sistemi za rezervaciju karata i bankarsko poslovanje. Ažurnost baze podataka je za transakcione sisteme veoma bitna.

OS za stone računare

Stoni računari (*desktop systems*) jesu mali računarski sistemi namenjeni jednom korisniku i dostupni po pristupačnim cenama. Ovu klasu karakteriše više vrsta operativnih sistema (DOS, MS Windows, Novell NetWare, UNIX, Linux)

OS za višeprocesorske sisteme

Višeprocesorski sistemi (*multiprocessor systems*) jesu sistemi sa više procesora koji su čvrsto povezani zatvorenim komunikacionim linijama a nalaze se u istom kućištu. Pominju se pod imenom paralelni sistemi. Procesori dele istu memoriju i sistemski časovnik. Komunikacija i sinhronizacija između procesora se obavlja preko deljive memorije. Prednosti višeprocesorskih sistema su povećanje brzine, ekonomičnosti i pouzdanosti. Što se tiče povećanja brzine, N procesora bi u idealnom slučaju radilo N puta brže, ali je zbog međuprocesorske sinhronizacije to povećanje manje od N . Ekonomičnost se postiže zato što se dele isti hardverski resursi, memorija, časovnik, DMA kanali, prekidni kontroleri i jedan izvor napajanja. Povećanje pouzdanosti se postiže tako što će sistem nastaviti da radi i ukoiko otkaže jedan ili više procesora.

Postoje 2 koncepta za realizaciju višeprocesorskih sistema: **simetrično multiprocesiranje (symmetric multiprocessing, SMP) i asimretično multiprocesiranje (asymmetric multiprocessing).**

U slučaju simetričnog multiprocesiranja svaki procesor izvršava istu kopiju operativnog sistema, pri čemu te kopije komuniciraju jedna sa drugom kad god je to potrebno. Svi procesori su ravnopravni, pa nema odnosa nadređeni/podređeni. U idealnom slučaju svakom procesoru se dodeljuje jedan proces i oni se izvršavaju nezavisno, bez slabljenja performansi. Većina modernih operativnih sistema podržava SMP. U slučaju asimetričnog multiprocesiranja svakom procesoru je dodeljen specifičan posao. Postoji jedan glavni procesor (*master*) koji potpuno kontroliše ceo sistem i koji dodeljuje poslove ostalim procesorima (*slave*).

Mrežni OS

Mrežni OS obezbeđuju okruženje u kome korisnici sa svojih lokalnih mašina mogu pristupiti resursima udaljenih mašina na dva načina: procedurom daljinskog prijavljivanja na sistem (*remote login*) ili razmenom datoteka sa u udaljenim sistemom (*remote file transfer*). Posebnu, mnogo kvalitetniju klasu mrežnih operativnih sistema predstavljaju distribuirani sistemi, kod kojih se gubi koncept lokalne i udaljene mreže.

Distribuirani OS

Distribuirani sistemi predstavljaju kolekciju procesora tj računara koji ne dele zajednički memoriju i sistemski časovnik. Umesto toga svaki procesor tj računar ima sopstvenu lokalnu memoriju, a međusobna komunikacija se ostvaruje putem mreže realizovane kao LAN ili WAN. Osim podatka, datoteka i štampača distribuiraju se i procesi. Četiri glavne prednosti distribuiranih sistema su **deljenje resursa, ubrzavanje izračunavanja, pouzdanost i komunikacije**.

Distribuirani sistemi zahtevaju mrežnu infrastrukturu i mogu biti realizovani kao klijent/server ili kao ravnopravni računarski sistemi koji dele resurse na mreži (*peer-to-peer systems*). U klijent/server arhitekturi postoje računari koji predstavljaju servere i računari koji koriste njihove usluge – klijenti.

OS za upravljanje u realnom vremenu

Sistemi za upravljanje u realnom vremenu (real-time systems) spadaju u klasu specijalnih operativnih sistema. Glavna karakteristika ovih sistema je davanje odziva u propisanom vremenskom intervalu i precizno definisana vremenska ograničenja u kojima se moraju dogoditi odzivi. Glavni izvori podataka su senzorski uređaji, razni prekidači, AD i DA konvertori. To su sistemi koji kontrolišu aplikacije specijalne namene, kao što su: specijalni grafički sistemi, sistemi za medicinsku grafiku,

sistemi za industrijsku kontrolu. Dele se na **čvrste** sisteme za rad u realnom vremenu i **meku** sisteme za rad u realnom vremenu.

Čvrsti realni sistemi garantuju da će se svi vaćni poslovi obaviti na vreme. Po pravilu, nemaju sekundarnu memoriju, tj diskove, jer najčešće rade u teškim uslovima, a kompletan kôd i operativnog sistema i aplikacije se čuva u ROM memoriji. Sam operativni sistem je ograničen, tako da mnoge funkcije operativnih sistema opšte namene ne postoje.

Meki realni sistemi su manje zahtevni i obezbeđuju 2 klase procesa, a to su obični procesi i procesi u realnom vremenu koji imaju apsolutni prioritet. To je mešavina sistema za rad u relanom vremenu i operativnih sistema opšte namene. Meki realni sistemi su sve vrste UNIX i Linux sistema, zatim Windows 2000/XP/2003.

Opšti pregled strukture operativnih sistema

Upravljanje osnovnim resursima računarskog sistema obezbeđuje više funkcionalnih grupa programa namenjenih za:

- Upravljanje procesorom
- Upravljenje memorijom
- Upravljanje ulazom i izlazom
- Upravljanje podacima
- Upravljanje sekundarnom memorijom
- Umrežavanje
- Zaštitu
- Korisnički interfejs

Koncepcije projektovanja operativnih sistema

U nastavku teksta opisane su koncepcije projektovanja OS, a to su monolitna organizacija, slojевита, organizacija i arhitektura mikrojezgra.

Monolitni sistemi

U prošlosti su monolitni sistemi predstavljali najčešću organizaciju OS-a. (velika zbka – big mess). Onje realizovan kao skup procedura koje se po potrebi mogu međusobno pozivati , bez ikakvih ograničenja. Korisnički programi upotrebljavaju service operativnog sistemana sledeći način: parametri sistemskog poziva se smeštaju na određena mesta (registri procesora, stek), nakon čega sledi pozivanje jezgra operativnog sistema. Ova operacija prebacuje procesor iz korisničkog režima u sistemski režim rada i kontrolu predaje operativnom sistemu. U sistemskom režimu rada dostupne su neke komande procesora kojima se ne može pristupiti iz korisničkog režima. Posle pozivanja jezgra, OS preuzima kontrolu i na osnovu parametara poziva određuje koju sistemsku proceduru treba pozvati. Nakon izvršenja procedure , kontrola se vraća korisničkom programu. Operativni sistem ima sledeću strukturu, sastavljenu od 3 osnovna skupa programa:

- Glavni program koji obrađuje sistemske pozive
- Skup sistemskih procedura koje se pozivaju prilikom sistemskih poziva
- Skup pomoćnih procedura koje koriste sistemske procedure

Slojevita (hijerarhijska) organizacija

U slojevitoj realizaciji (layered system), OS se deli na različite slojeve. Slojevi su organizovani hijerarhijski: svaki sloj može da poziva samo funkcije nižih slojeva. Prvi slojeviti operativni OS, THE (*Technische Hogeschool Eindhoven, E.W. Dijkstra*), sastojao se od 6 slojeva. Slojevi od 0 do 3 predstavljaju jezgro operativnog sistema i rade u sistemskom režimu:

- Multi sloj upravlja procesorom, tj dodeljuje procesor različitim procesima
- Prvi sloj upravlja memorijom, tj dodeljuje potrebnu memoriju procesima
- Drugi sloj upravlja komunikacijom između različitih procesa i komandnog interpretera
- Treći sloj obavlja ulazno-izlazne operacije

Na četvrtom sloju rade korisnički programi, koji se ne brinu o dodeli procesora, dodeljivanju memorije, komandnom interpreteru, I/O operacijama – sve to obavljaju niži slojevi. Znači:

- OS monolitne strukture sastoji se od skupa procedura bez ikakvog grupisanja ili hijerarhije
- OS slojevite strukture deli se na više slojeva od kojih svaki ima tačno određenu funkciju i oslanja se isključivo na funkcije nižih slojeva, kojima pristupa pomoću poziva sličnih sistemskom pozivu

Generalno, slojeviti OS deli se na određeni broj slojeva od kojih se svaki gradi na vrhu prethodnog sloja. Najniži sloj (*layer 0*) predstavlja hardver, dok je najviši (*layer N*) – korisnički interfejs. Sa ovakvim modularnim konceptom, slojevi koriste isključivo funkcije i usluge nižih slojeva.

Slojeviti OS dozvoljava pozivanje (korišćenje) operacija stvaranja i uništavanja procesa samo iz sloja koji se u hijerarhiji nalazi iznad svih slojeva operativnog sistema. To znači da je postojanje procesa isključivo vezano za korisnički sloj. Iako su procesi locirani u korisničkom sloju, oni su međusobno razdvojeni zahvaljujući činjenici da svaki od procesa poseduje zaseban adresni prostor – korisnički prostor (*user space*). Na istom principu se i zasniva i razdvajanje procesa i operativnog sistema. Ali, razdvajanje korisničkog prostora (*user space*) od adresnog prostora operativnog sistema tj sistemskog prostora (*kernel space*), sprečava da se pozivi operacija operativnog sistema zasnivaju na korišćenju poziva potprograma. Zato je neophodno uvođenje posebnog mehanizma sistemskih poziva koji omogućuje prelazak iz korisničkog prostora u sistemski prostor radi pozivanja operacija operativnog sistema. Sistemski pozivi zahtevaju korišćenje specifičnih asemblerских naredaba i zbog toga se sakrivaju unutar sistemskih potprograma. Sistemski potprogrami obrazuju sistemsku biblioteku. Prema tome, pozivanje sistemskih operacija svodi se na pozivanje potprograma iz sistemске biblioteke. Sistemska biblioteka se isporučuje uz operativni sistem kako bi se koristila u postupku povezivanja. U toku povezivanja, bibliotečki potprogrami se vezuju za objektni oblik korisničkog programa radi stvaranja izvršnog oblika korisničkog programa koji tako postaje spremna za saradnju sa OS.

Sistemski pozivi

Aplikacioni programi komuniciraju sa OS pomoću sistemskih poziva (*system call*) tj preko operacija (funkcija) koje definiše OS. Sistemski pozivi se realizuju pomoću sistema prekida: korisnički program postavlja parametre sistemskog poziva na određene memorijske lokacije ili registre procesora, inicira prekid, operativni sistem preuzima kontrolu, uzima parametre, izvršava tražene radnje, rezultat stavlja na određene memorijske lokacije ili u registre i vraća kontrolu programu.

Sistemski pozivi često podržava i hardver tj procesor, na taj način što razlikuje 2 režima rada: korisnički režim (*user mode*) i sistemski režim (*supervisor mode*). Korisnički programi mogu isključivo da rade u korisničkom režimu rada procesora, sistemski režim rada je predviđen za operativni sistem. Prilikom sistemskih poziva procesor prelazi iz korisničkog režima rada u sistemski, a vraća se u korisnički režim posle obrade poziva.

Sistemski pozivi obezbeđuju interfejs između programa koji se izvršava i operativnog sistema. Generalno, realizuju se pomoću asemblerorskog jezika, ali i C i C++ takođe omogućavaju realizaciju sistemskih poziva. Program koji se izvršava može proslediti parametre operativnom sistemu na tri načina:

- Prosleđivanjem parametara u registre procesora
- Postavljanjem parametara u memorijskoj tabeli, pri čemu se adresa tabele prosleđuje u registre procesora
- Postavljanjem parametara na vrh steka (push), koje operativni sistem „skida“ (pop)

Arhitektura mikrojezgra

Mikrojezgro (microkernel) predstavlja veoma savremen koncept u realizaciji savremenih operativnih sistema. Osnovna zamisao je napraviti minimalno i pouzdano jezgro visokih performansi, a sve ostale funkcije jezgra potisnuti u tzv korisnički prostor (*user space*). Korisnički moduli međusobno komuniciraju slanjem poruka (*message passing*). U dobre osobine arhitekture mikrojezgra spadaju:

- Jednostavno proširivanje i optimizacija jezgra
- Jednostavno dodavanje novih modula bez uticaja na osnovno jezgro
- Jednostavna prenosivost na drugu računarsku arhitekturu
- Veća pouzdanost (manje koda se izvršava u režimu jezgra)
- Veća sigurnost