

O operacijskih sistemih

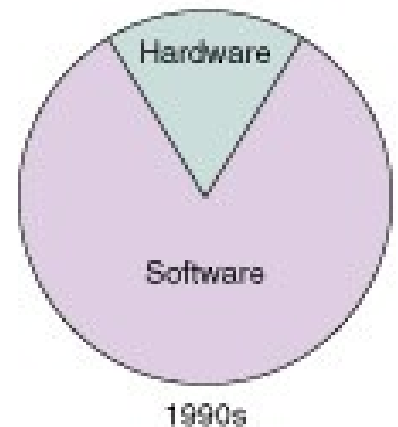
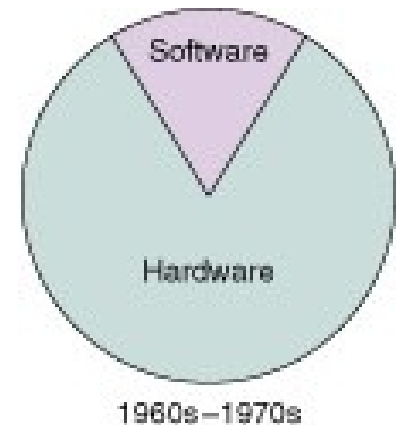


Komponente računalniškega sistema

1. **Aparaturna oprema** – Zagotavlja osnovne računalniške vire (CPE, pomnilnik, vhodno izhodne naprave).
2. **Operacijski sistem** – nadzira in koordinira uporabo aparaturne opreme med različnimi aplikacijskimi programi za različne uporabnike.
3. **Aplikacijski programi** – določajo na način, kako uporabiti sistemske vire za reševanje računalniških problemov za uporabnike (prevajalniki, databsistemi za delo s podatkovnimi bazami, video igrice, poslovni programi).
4. **Uporabniki** (ljudje, stroji, drugi računalniki).

Naraščanje pomena programske opreme

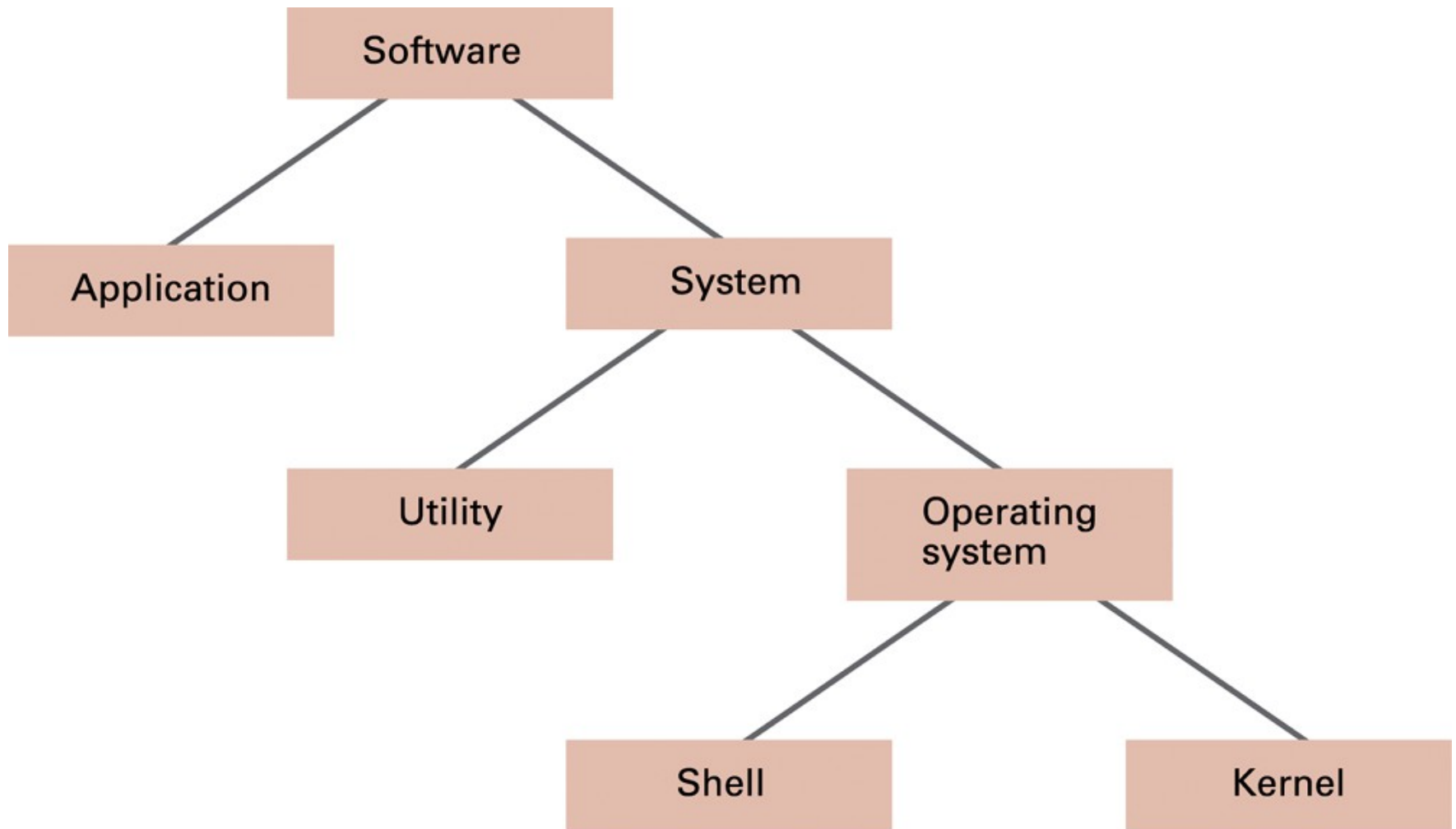
- Premiki v ceni tehnologije
 - Padanje cen aparaturene opreme
 - Naraščanje kompleksnosti programske opreme in rast njene cene
 - Rast cene programerskega dela
- Programski pripomočki za “reverzni inženiring” organizacij
- Programska oprema pomaga pri standardizaciji poslovnih postopkov



Ključne programske komponente informacijskih sistemov

- **Software**
 - Množica programov, ki nadzirajo delovanje aparaturene opreme računalnika
- **Program**
 - Množica kodiranih ukazov, ki jih bere in izvaja računalnik
- **Dokumentacija**
 - Množica navodil, ki pomagajo uporabniku pri uporabi računalniških programov

Klasifikacija programske opreme

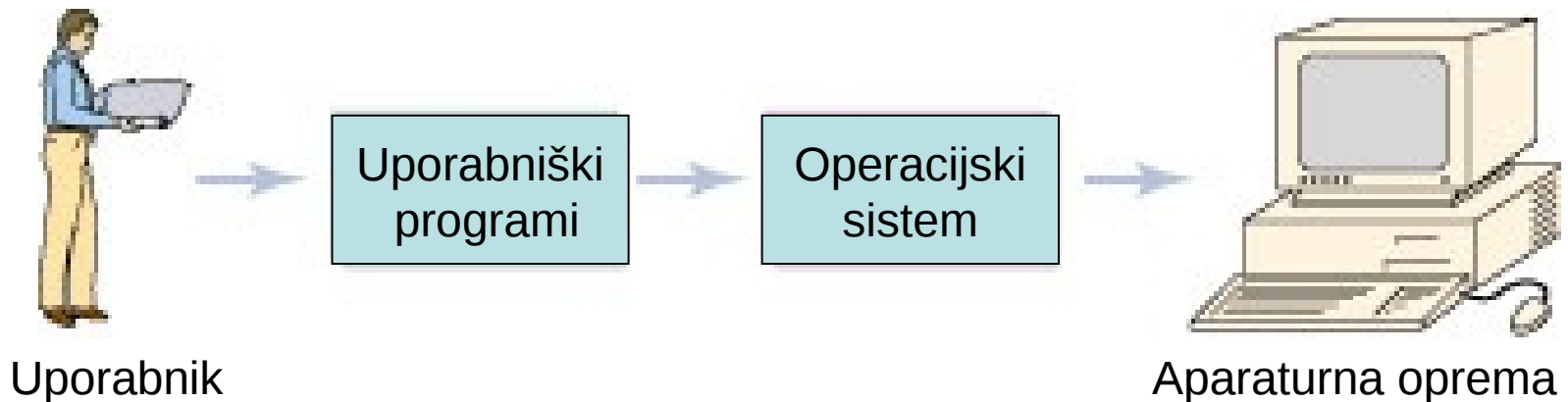


Uporabniški programi

- Omogočajo uporabniku izvedbo določene naloge
 - Pisanje nekega dokumenta
 - Izvedbo določenih računov
 - Uporabo elektronske pošte
 -
- Interaktirajo s sistemsko programsko opremo
- Dva osnovna tipa
 - Prilagojeni uporabniku (customized)
 - Neprilagojeni, “vzeti s police” (Off-the-shelf)

Sistemski programi

- Temu včasih pravijo tudi “operacijski sistem”
- Zbirka programov za krmiljenje osnovnih operacij računalniške aparature opreme
- Cilj – naj bo delovanje računalnika za uporabnike transparentno



Sistemske programi

- Sistemske programi nudijo okolje, primerno za razvoj in izvajanje programov. Delimo jih lahko na programe za:
 - Rokovanje z datotekami
 - Informiranje o statusu
 - Spreminjanje datotek
 - Podporo programskim jezikom
 - Nalaganje in izvajanje programov
 - Komunikacije
 - Druge uslužnostne programe
- Pogled **uporabnikov** na operacijski sistem je večinoma opredeljen s **sistemskimi programi** in ne s sistemskimi klici.

Funkcije sistemske program. opreme

- Izvedba sistemskih operacij
 - Zagon računalnika
 - Branje programov v pomnilnik
 - Upravljanje dodeljevanja pomnilnika programom
 - Vzdrževanje strukture direktorijev
 - Krmiljenje perifernih naprav
- Nudenje uporabniškega vmesnika
 - Vmesnik, temelječ na ukazih
 - Grafični vmesnik (Graphical User Interface: GUI)

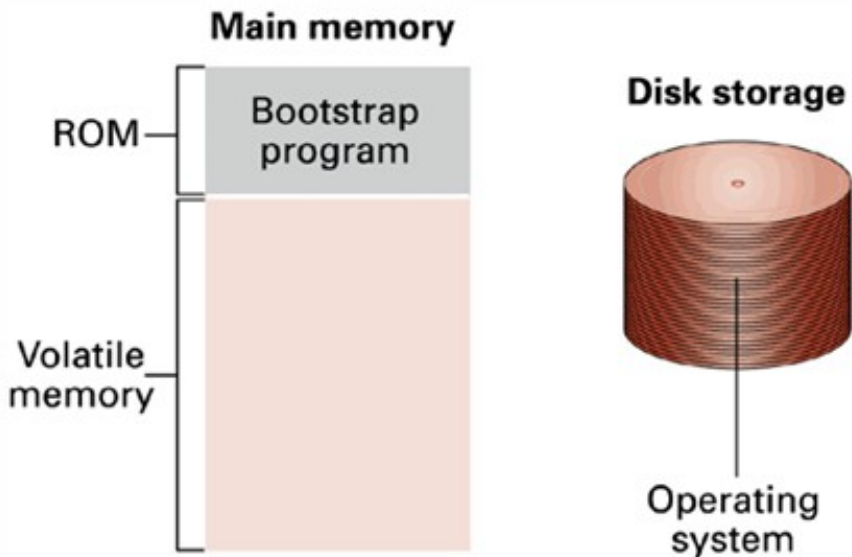
Kaj je torej operacijski sistem?

- Programje, ki deluje kot posrednik med uporabnikom računalnika in računalniško aparaturno opremo.
- Cilji operacijskega sistema:
 - Izvajanje uporabniških programov in lažje reševanje problemov uporabnika.
 - Narediti računalniški sistem sploh uporabljiv.
- Učinkovita uporaba računalniškega sistema.

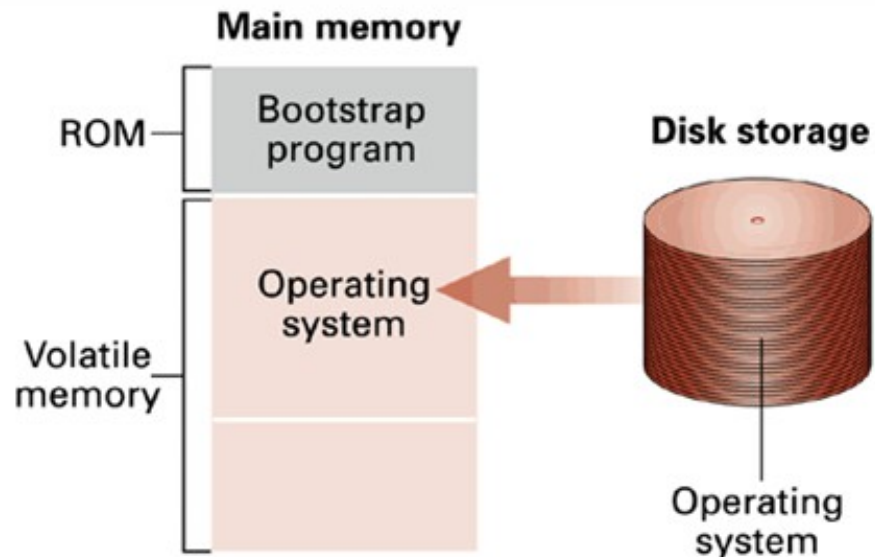
Kako se stvar začne? (bootstrapping)

- **Bootstrap**: program v pomnilniku ROM (read only memory)
 - Požene ga CPE ob vklopu računalnika
 - Prepiše operacijski sistem z diska v pomnilnik
 - Nato preskoči na operacijski sistem

Zagon operacijskega sistema

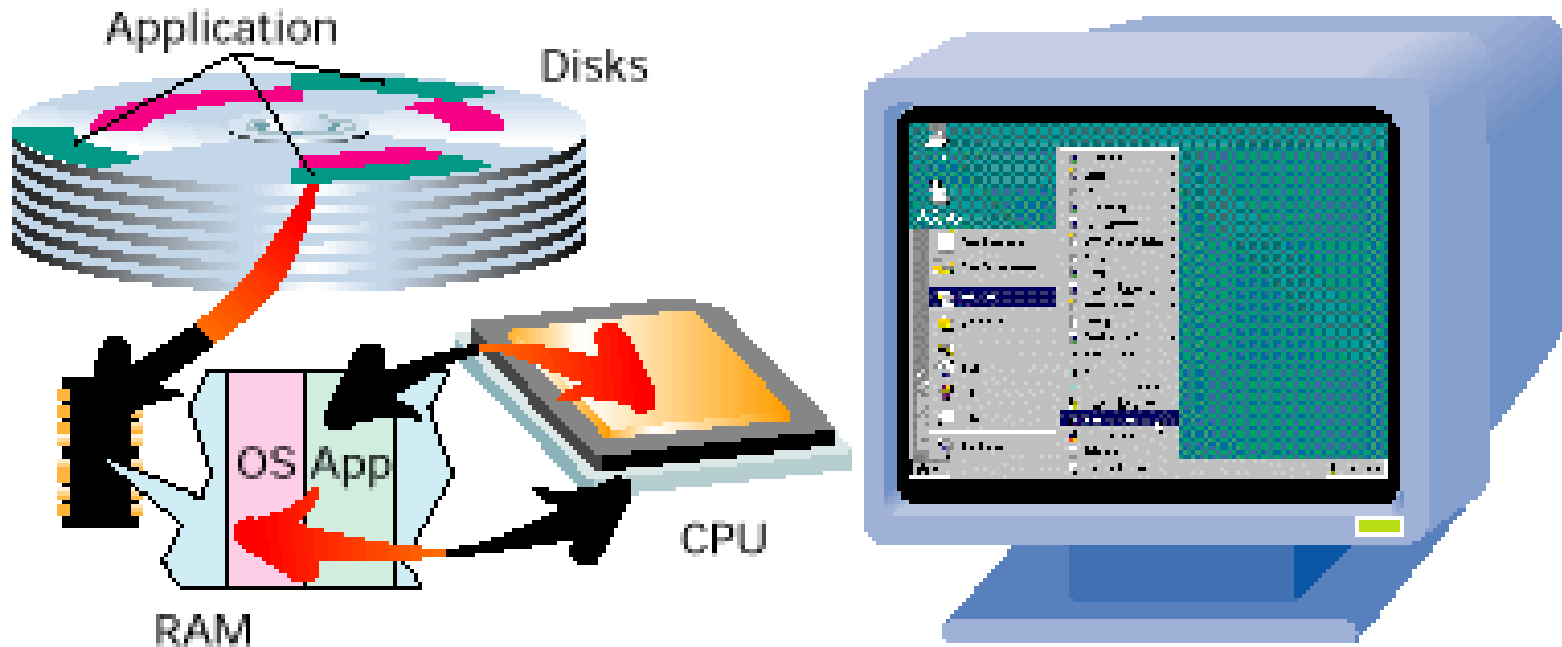


Korak 1: Ob vklopu računalnika se požene zagonski program (bootstrap program), ki je že v pomnilniku. Operacijski sistem pa se nahaja na disku.



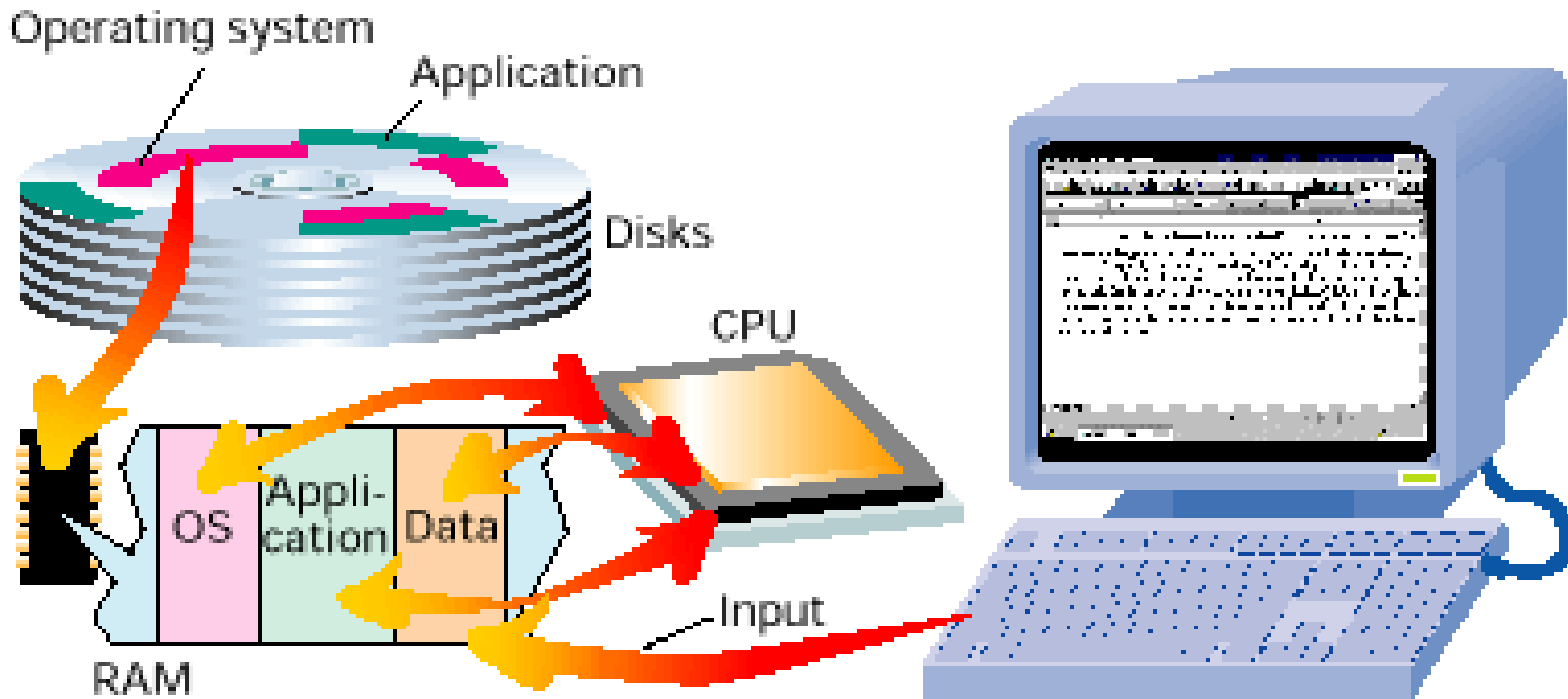
Korak 2: Zagonski program prepíše operacijski sistem v primarni pomnilnik in preda nadzor njemu.

Kaj dela operacijski sistem?



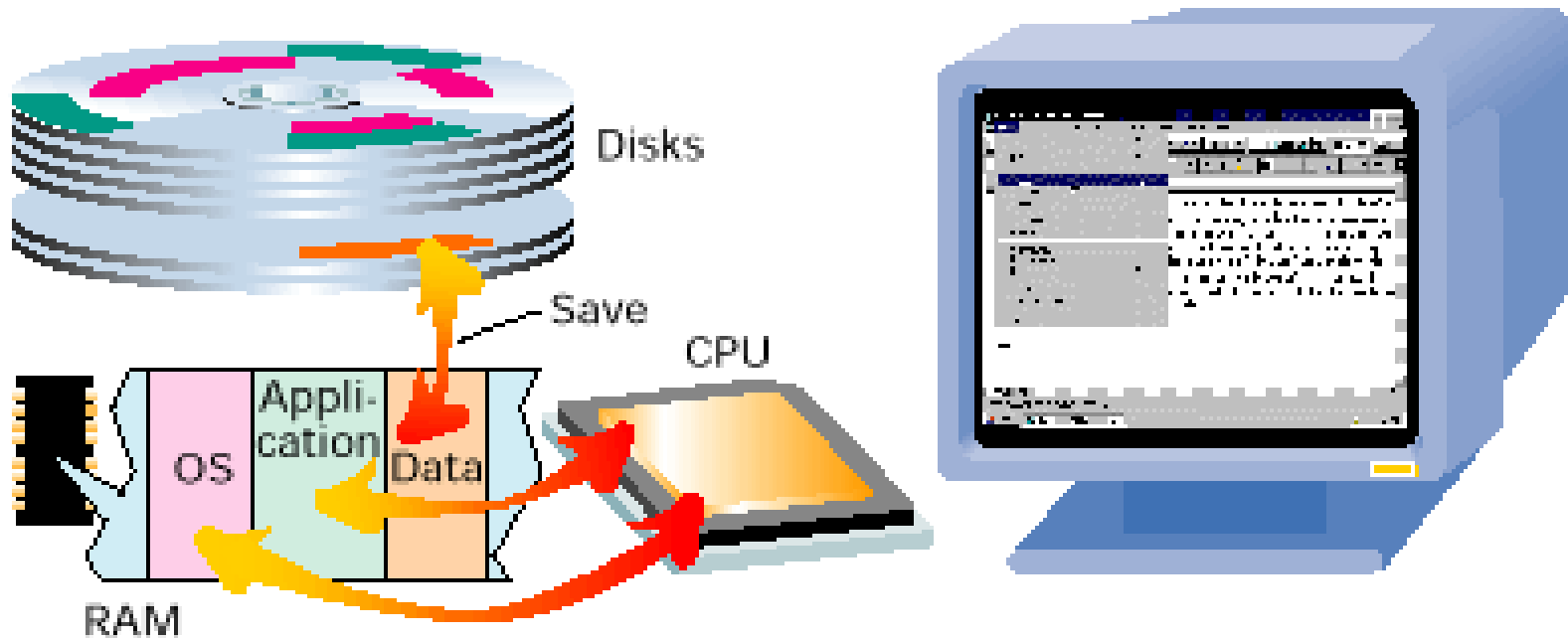
- Zahtevo za zagon neke aplikacije (na primer urejevalnika Word) pošljemo operacijskemu sistemu
- OS poišče program na disku, ga naloži v pomnilnik in ga požene

Kaj dela operacijski sistem?



- OS nudi aplikaciji različne servise
- OS, aplikacije in podatki tekmujejo za pomnilnik

Kaj dela operacijski sistem?



- Med potekom uporabniškega programa mu operacijski sistem (OS) po potrebi bere ali zapisuje podatke na disku

Pogledi na operacijski sistem

- Dva pogleda
 - Pogled uporabnika
 - OS nudi abstrakcijo virov za uporabnika, tako da je reševanje problemov lažje
 - Sistemski pogled
 - OS deluje kot alokator virov, upravnik procesov in nadzornik sistema

Še nekaj definicij operacijskega sistema

- **Resource allocator** – upravlja in alocira vire.
- **Control program** – nadzoruje izvajanje uporabniških programov in operacij vhodno-izhodnih naprav .
- **Kernel** – Program, ki te e ves as (kar v splošnem ne velja za uporabniške programe).

Še nekaj pojmov pri OS

- **Režimi izvajanja** (Running Modes)

- **Uporabniški režim** (User Mode)

- Omejena uporaba aparature opreme in omejen nabor strojnih ukazov

- V tem režimu te je uporabniški programi

- **Sistemski režim** (Kernel Mode, Trusted mode)

- Neomejen dostop do aparature opreme

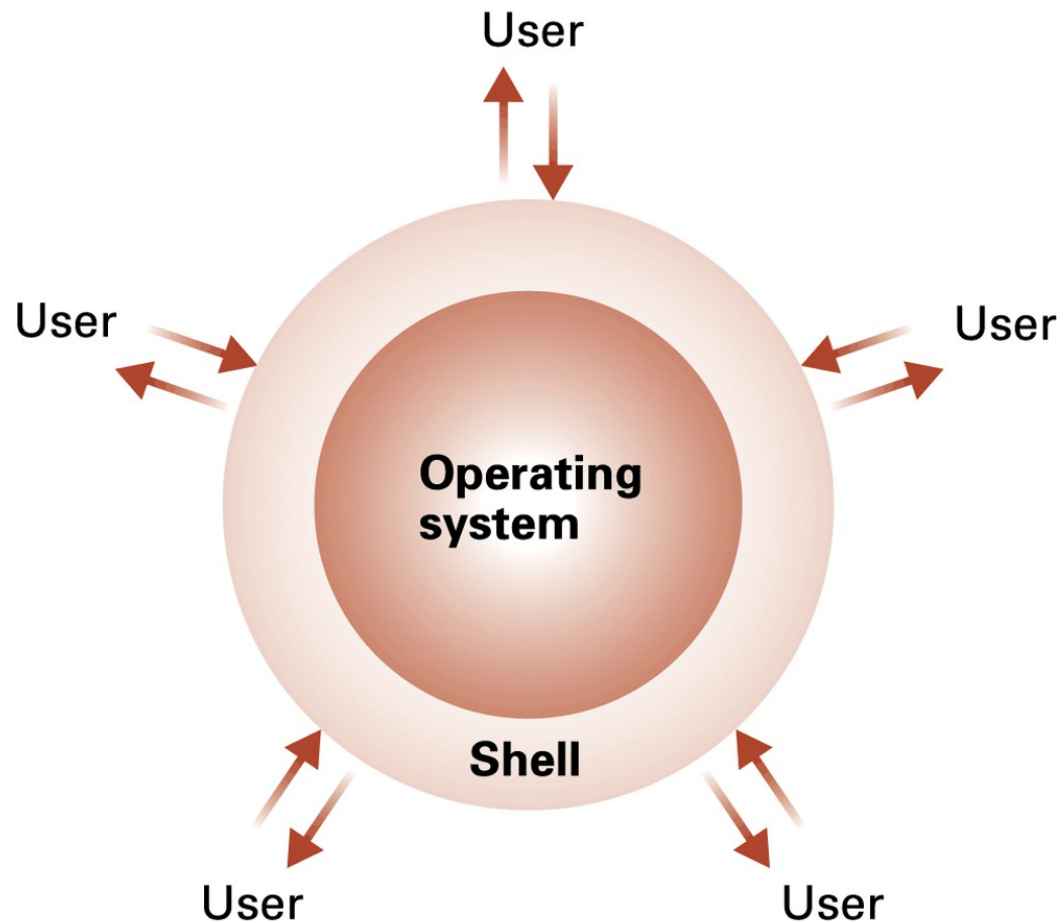
- Možna uporaba vseh strojnih ukazov

- V tem režimu v splošnem te je operacijski sistem

Komponente operacijskega sistema

- **Lupina** (shell): del, ki komunicira z uporabniki
 - Grafi ni uporabniški vmesnik (GUI)
Window manager
- **Jedro** (kernel): vsebuje komponente za opravljanje osnovnih funkcij
 - Upravljanje z datotekami (File manager)
 - Gonilniki naprav (Device drivers)
 - Upravljanje s pomnilnikom (Memory manager)
 - Razvrščanje nalog (Scheduler and dispatcher)

Lupina (shell) kot vmesnik med uporabniki in operacijskim sistemom



Upravljanje datotek (File Manager)

- **Direktorij ali mapa (Directory, folder)**: Skupina datotek, ki jo običajno ustvari uporabnik
- **Pot (Path)**: Položaj datoteke v hierarhiji direktorijev
- **Opisnik datoteke (File descriptor)**: Informacija, potrebna za dostop do odprte datoteke

Upravljanje s pomnilnikom (Memory manager)

- **Stran (Page)**: enota upravljanega pomnilnika (nekaj kilobytov)
- **Navidezni pomnilnik (Virtual memory)**: navidezni pomnilniški prostor
 - Tvorimo ga s premetavanjem strani med resni nim pomnilnikom in prostorom na disku

Procesi

- **Program** = statična skupina ukazov
- **Proces** = aktivnost (“življenje”) programa, ki teče
- **Stanje procesa** = trenutno stanje aktivnosti
 - Posnetek pomembnih delov stanja programa
 - Programski števec, drugi registri, dodeljeni pomnilnik

Administracija procesa

- Razvrš evalnik (Scheduler)
 - Stanje vseh procesov hrani v procesni tabeli
 - Pripravljene ali čakajoči procesi (Ready or waiting)
 - Prioriteta procesov
 - Dodatni podatki: strani v pomnilniku ipd.

Administracija procesa (nadaljevanje)

- Dispe er (Dispatcher)

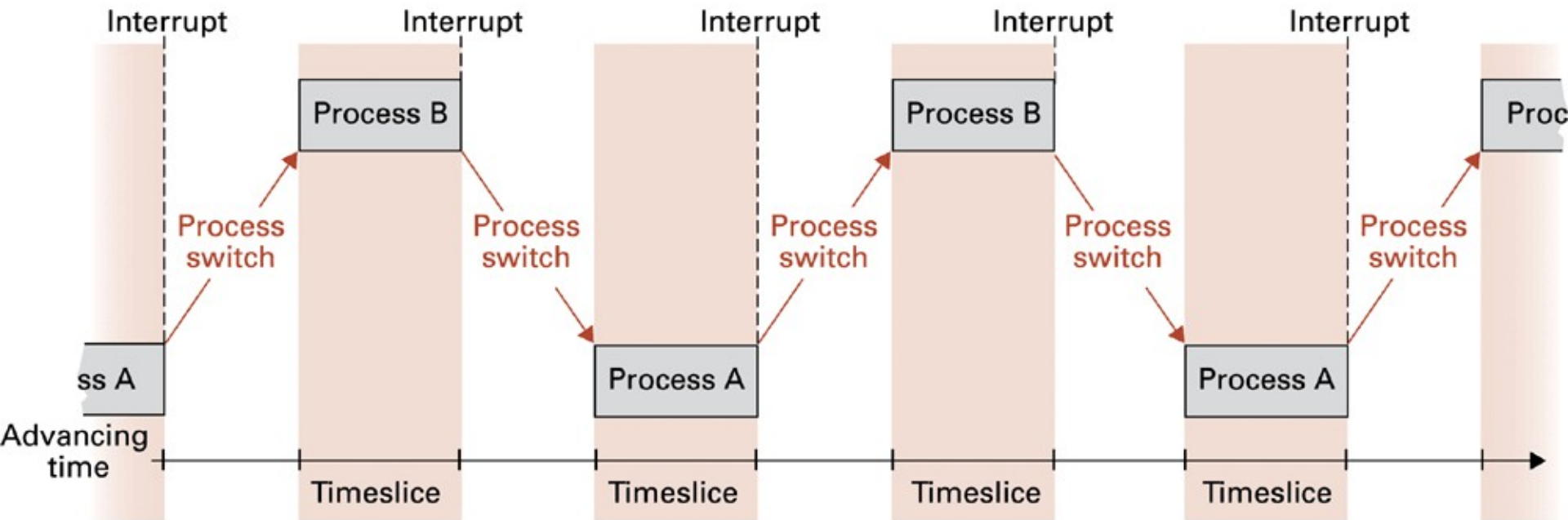
- Dodeli eno osnovno rezino (kvantum) procesu, ki je pripravljen

- Izvede preklop procesov (context switch), ko dodeljena osnovna rezina pote e

- Iztek osnovne rezine nakaže prekinitev (interrupt)

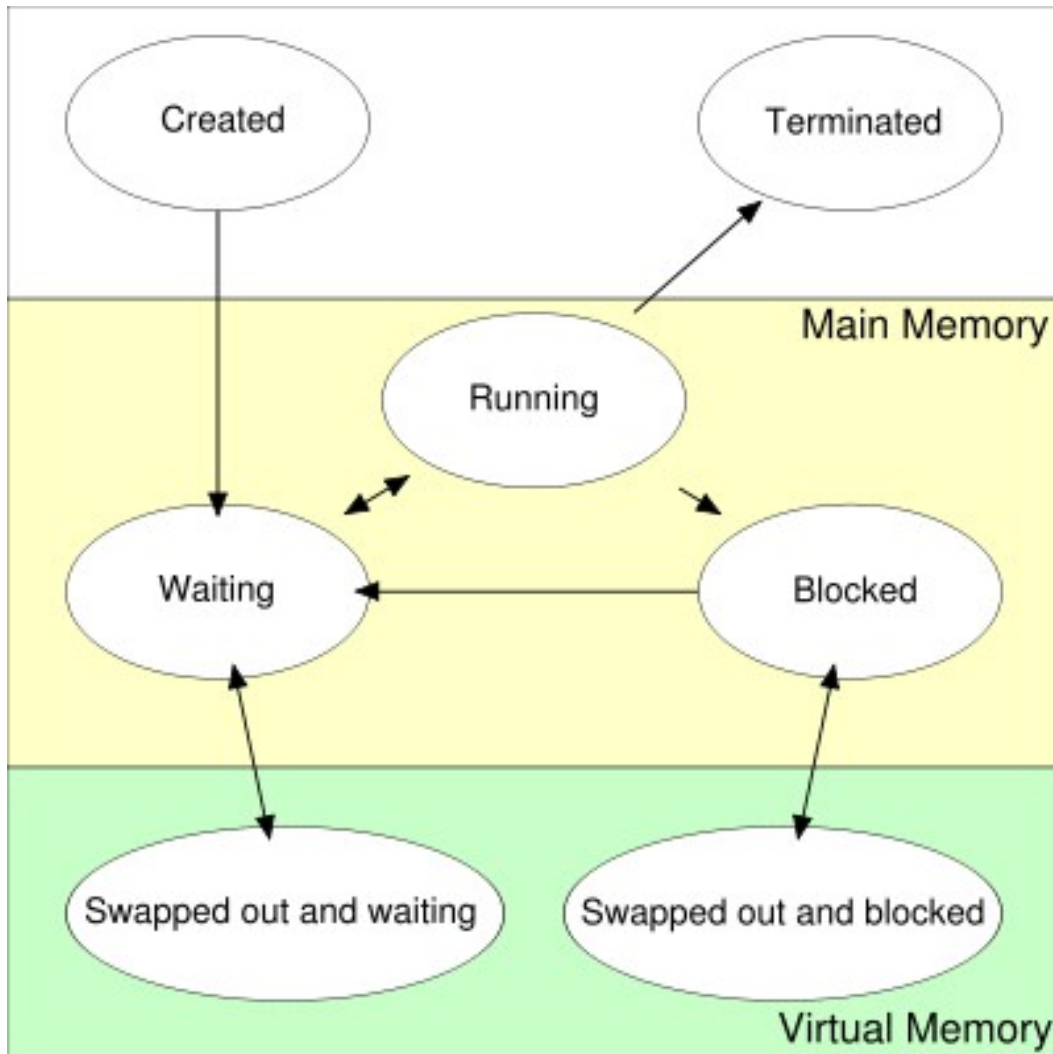
- Rokovalnik prekinitev (Interrupt handler): je del dispe erja

Dodeljevanje časa (Time-sharing)



Dodeljevanje časa procesoma A in B

Stanje izvajanja (state) procesa



- kreiran
- končan
- v stanju izvajanja
- čakajo (pripravljen)
- blokiran (npr. čakajo na zaseden vir)
- čakajo (na disku navideznega pomnilnika)
- blokiran (na disku navideznega pomnilnika)

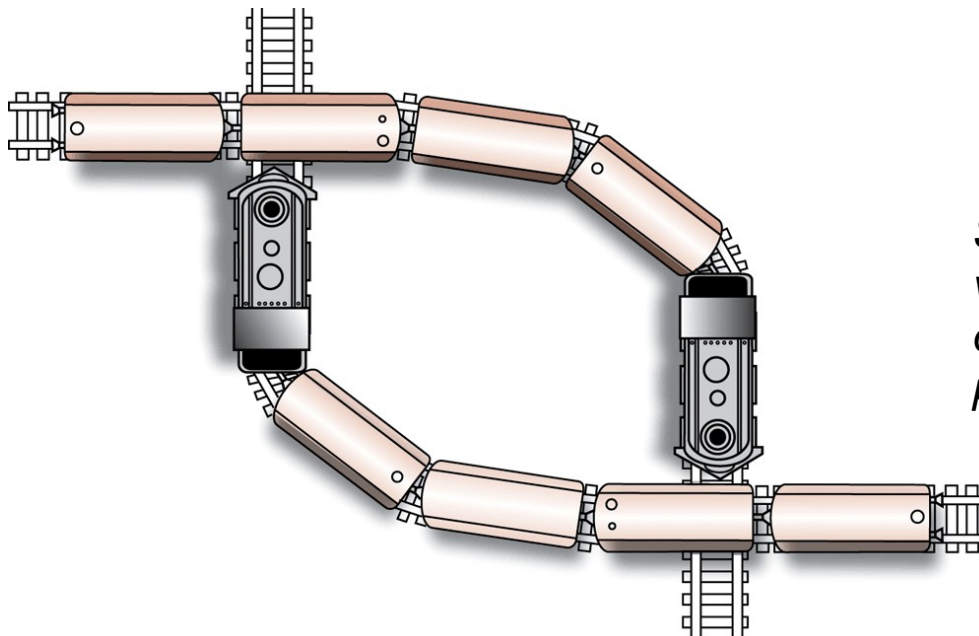
Kako rešujemo tekmovanje za vire?

- Določene vire lahko zaseda le en proces hkrati – procesi tekmujejo za vire
 - Če je vir zaseden, morajo ostali procesi počakati, da se vir sprosti (so blokirani)
- **Semafor** = kontrolna zastavica, ki pove, da je nek vir zaseden
 - Z ustrezno funkcijo moramo sposobno izvesti operaciji test in set
- **Kritična sekcija** = zaporedje ukazov, ki jih naenkrat lahko izvaja le en proces
 - Običajno ščitena s semaforjem
- Medsebojno izobčudenje (Mutual exclusion) = zahteva za pravilno implementacijo kritične sekcije



Smrtni objem (Deadlock)

- Dva procesa blokirata drug drugemu nadaljevanje
- Pogoji, ki vodijo v smrtni objem
 1. Tekmovanje za vire, ki niso deljivi
 2. Vsak proces potrebuje najmanj dva vira
 3. Alociranega vira ne moremo sprostiti



Smrtni objem dveh vlakov, ki tekmujeta za dve nedeljivi križiš i proge

Smrtni objem (primer)

viri so nedeljivi (jih lahko zaseže le en proces)

Proces 1:

zaseži vir 1

...

zaseži vir 2

...

sprosti vir 2

...

sprosti vir 1

Proces 2:

zaseži vir 2

...

zaseži vir 1

...

sprosti vir 1

...

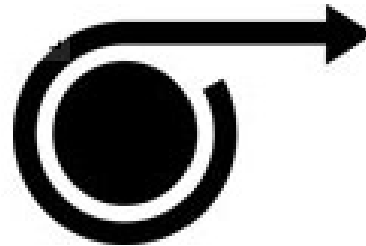
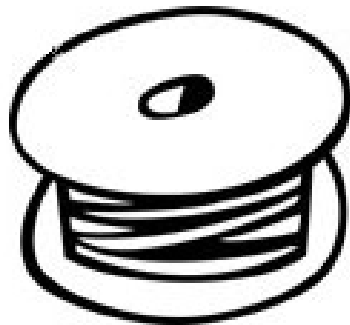
sprosti vir 2

Smrtni objem (kako se izogniti)

- izogibanje: vnaprej vemo kakšne vire zahteva proces
 - bankirjev algoritem (Dijkstra) za ugotavljanje varnih stanj in izdajo dovoljenj za zaseganje virov
- preprečevanje:
 - izogibanje ekskluzivnemu zaseganju virov in uporaba alternativ – spooling, “backoff” algoritmi...
 - zahteva po vseh resursih pred začetkom izvajanja
 - izogibanje krožnemu zasedanju na vire (npr. z upoštevanjem prioritete virov)
- odkrivanje:
 - ponoven zagon (“restart”) procesov

“Spooling” (namotavanje?)

- Zahtevano operacijo prestavimo na nek kasnejši čas
 - Tako izgledajo nedeljivi viri kot deljivi (shareable)
- Tehnika izogibanja smrtnemu objemu

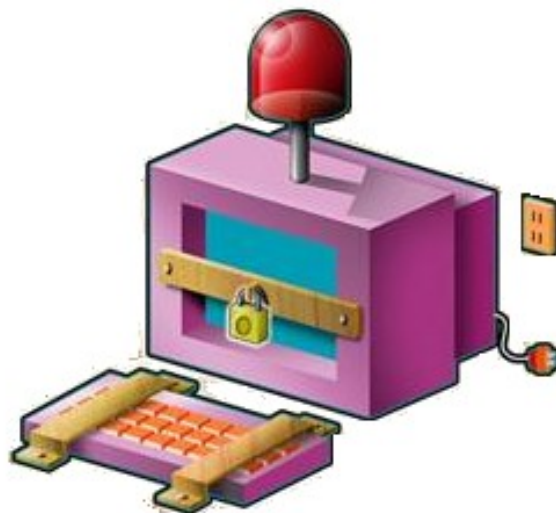


Procesi in varnost

- Operacijski sistem preprečuje nelegalen dostop do virov
 - Različen pomnilnik za različne procese
 - Privilegirani ukazi možni le v jedru
 - Vsi dostopi do datotek so možni le preko jedra
 - Dostop do drugih naprav le preko jedra

Varnost pred zunanjimi napadi

- Najbolj pogosta zaš ita: Uporabniško ime in geslo
 - Problem: kraja gesel
 - Problem: avtomatski uganjevalci gesel
 - Protiukrepi:
 - Uporabniku povemo, kdaj je bil zadnji prijavljen
 - Poro anje o ponavljanih napa nih ugibanjih
 - Uganjevalca prijavimo v poseben ra un (account) in gledamo, kaj po ne



Varnost pred notranjimi napadi

- Operacijski sistem preprečuje nelegalen dostop do virov
 - Različen pomnilnik za različne procese
 - Privilegirani ukazi možni le v jedru
 - Vsi dostopi do datotek so možni le preko jedra
 - Dostop do drugih naprav le preko jedra

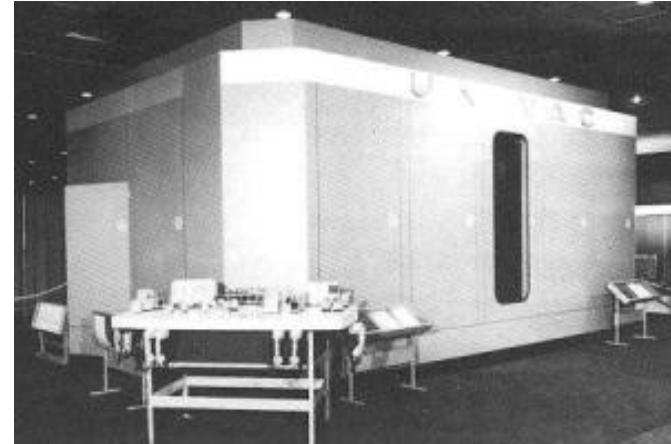


Vrste operacijskih sistemov

- Preprosti paketni sistemi (Batch Systems)
- Multiprogramski paketni sistemi
- Sistemi z dodeljevanjem časa (Time-Sharing Systems)
- Sistemi osebni računalnikov
- Paralelni sistemi
- Porazdeljeni sistemi
- Sistemi za delo v realnem času (Real -Time Systems)

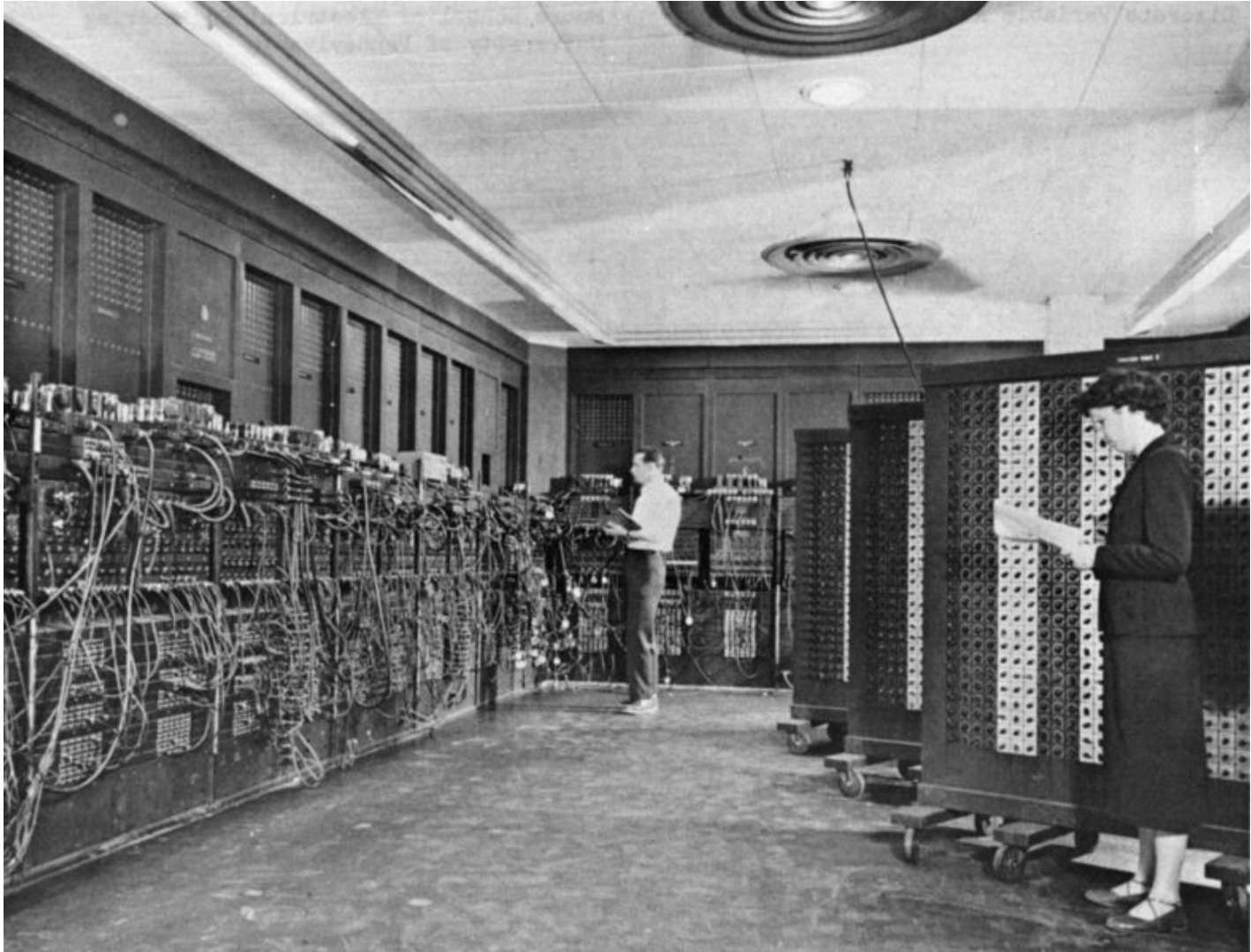
Računalniki 1. generacije

- 1940-1955:
 - elektronke za vezja
 - magnetne bobne za podatke (pomnilnik)
 - papirne kartice, tiskani izpisi za I/O
 - en program naenkrat
- brez pravega OS in višjenivojskih prog. jezikov
 - program pri npr. ENIACu določen s pretikanjem kablov in stikal, pozneje tudi s shranjenimim funkcijami v ROM
 - programi v binarni strojni kodi
 - podatki na karticah



ENIAC

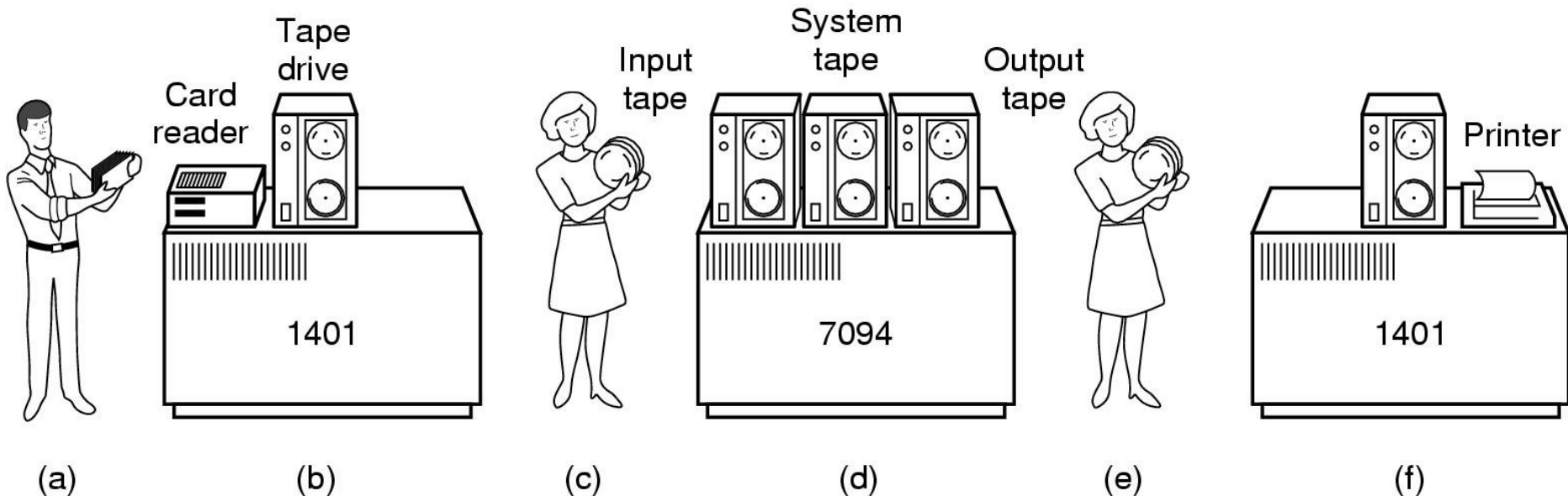
Electronic Numerical Integrator and Computer



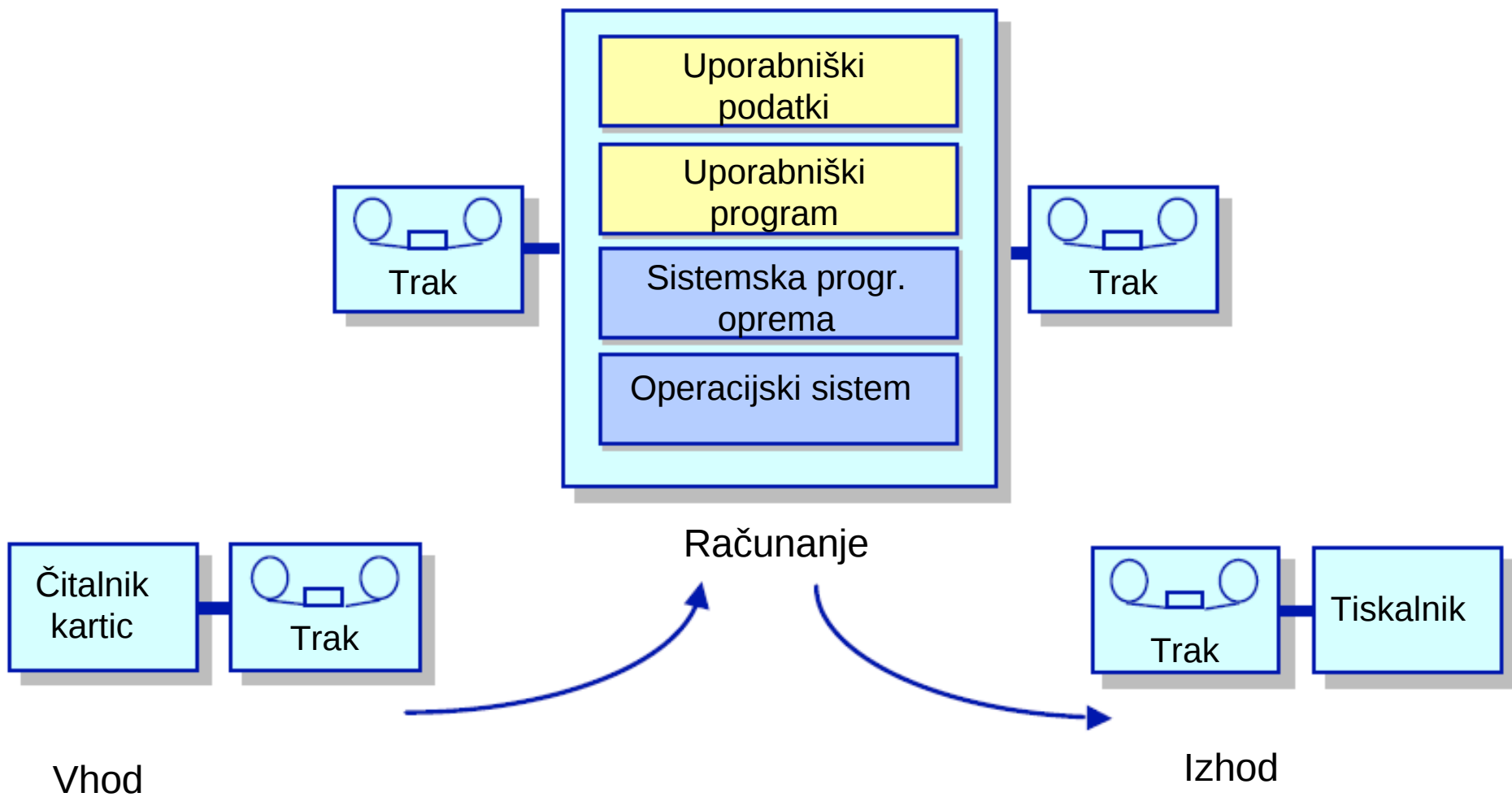
Raunalniki 2. generacije

(1955-1965)

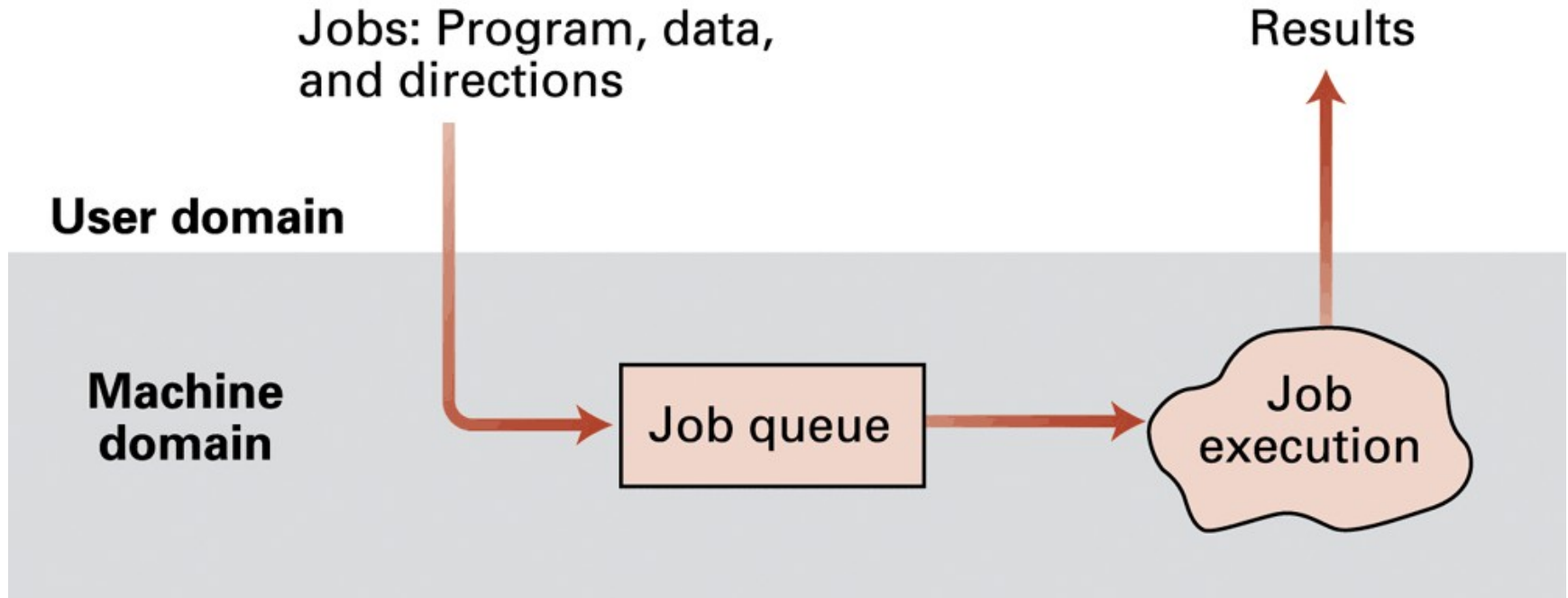
- Tranzistorji
- Programski jeziki
- Prvi OSi za paketno obdelavo



Preprosti paketni sistemi

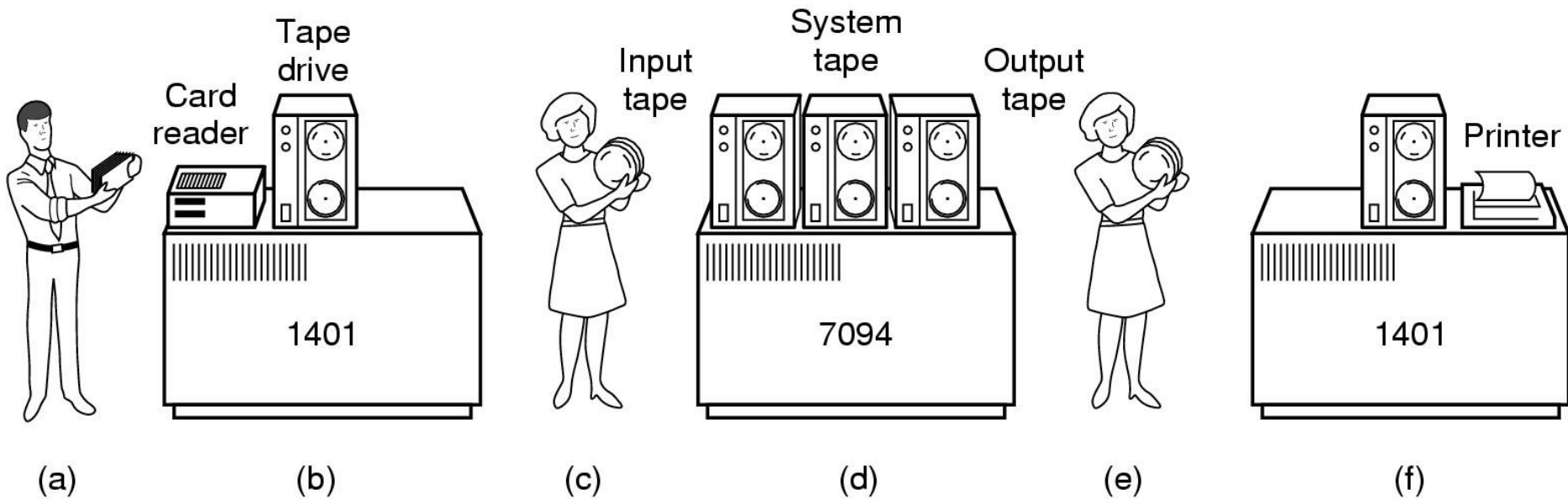


Paketno procesiranje (Batch processing)



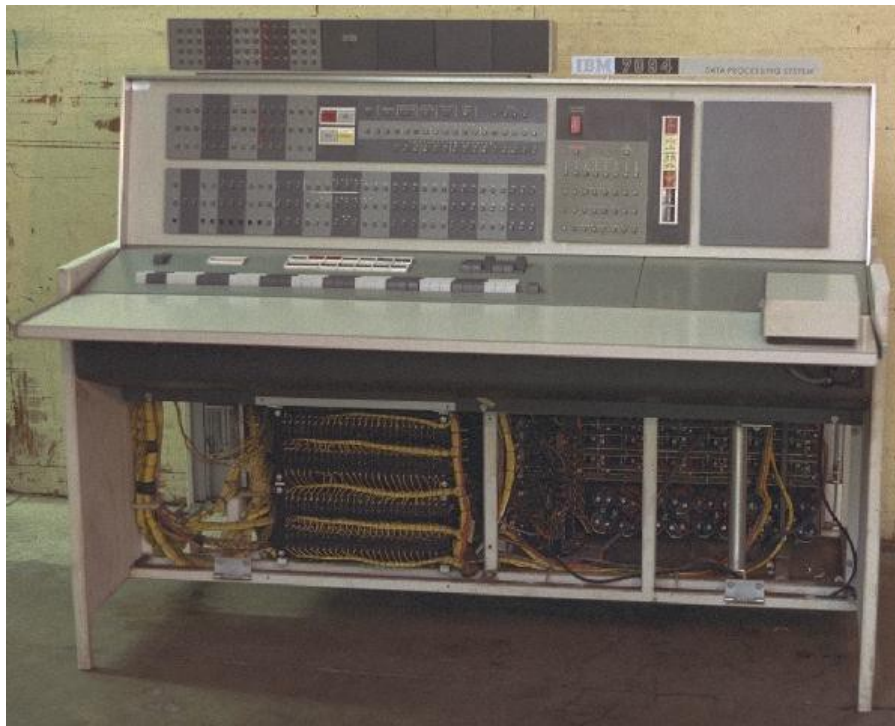
Raunalniki 2. generacije (1955-1965)

Prvi sistemi za paketno obdelavo



Raunalniki 2. generacije

(1955-1965)



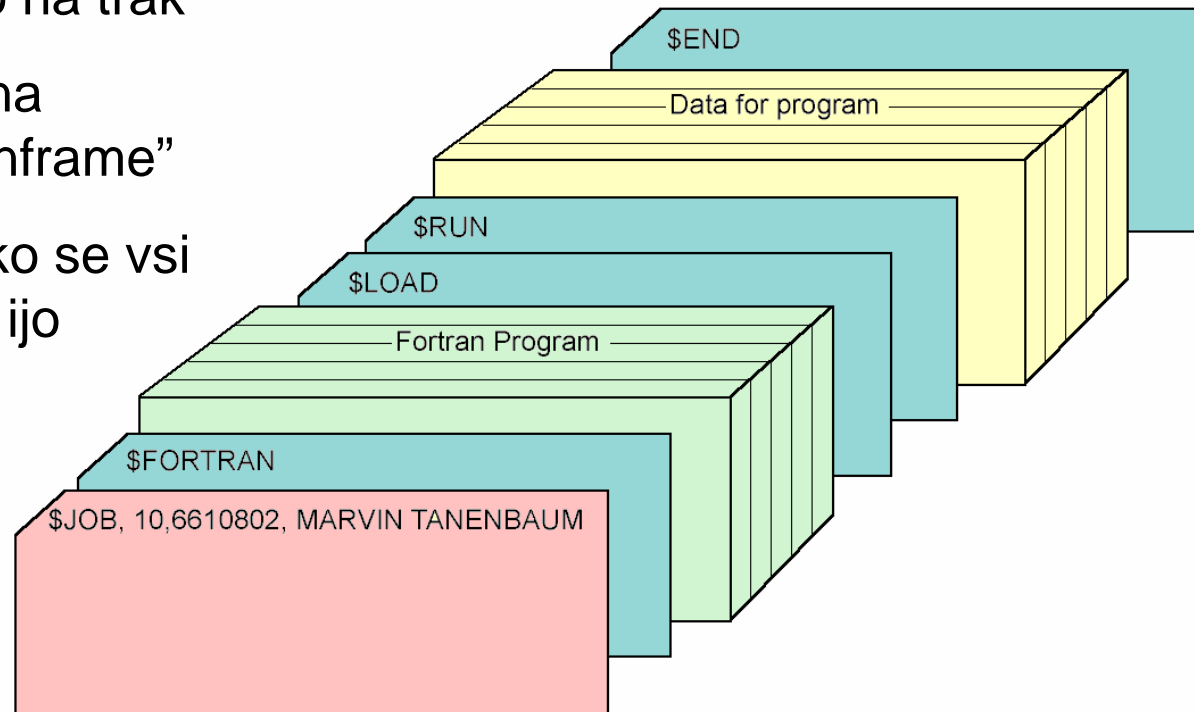
IBM 7094



IBM 1401

Raunalniki 2. generacije (1955-1965)

- Programer pripravi paket luknjanih kartic
- Več paketov luknjanih kartic prenesemo na trak
- Trak požemo na računalniku "mainframe"
- Izpis rezultatov, ko se vsi posli (job) zaključijo

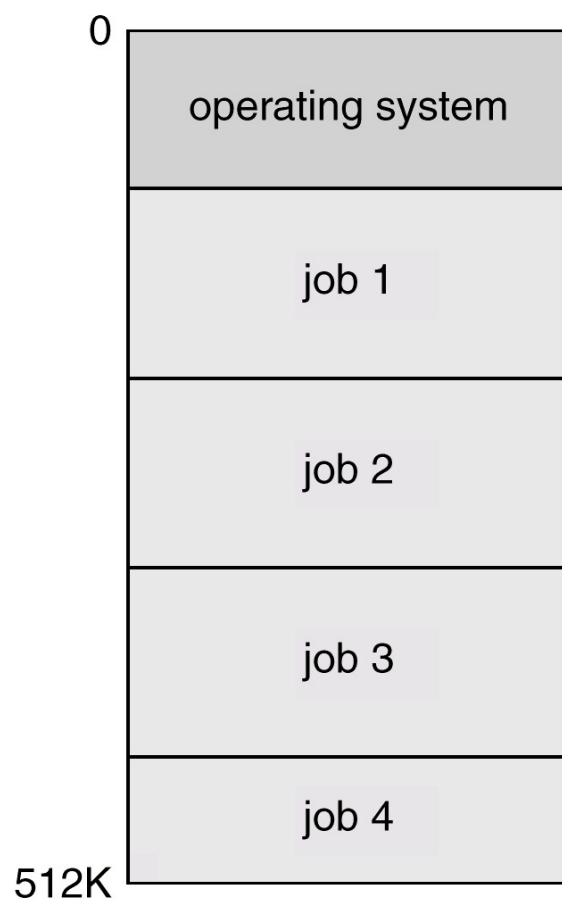


Preprosti paketni sistemi

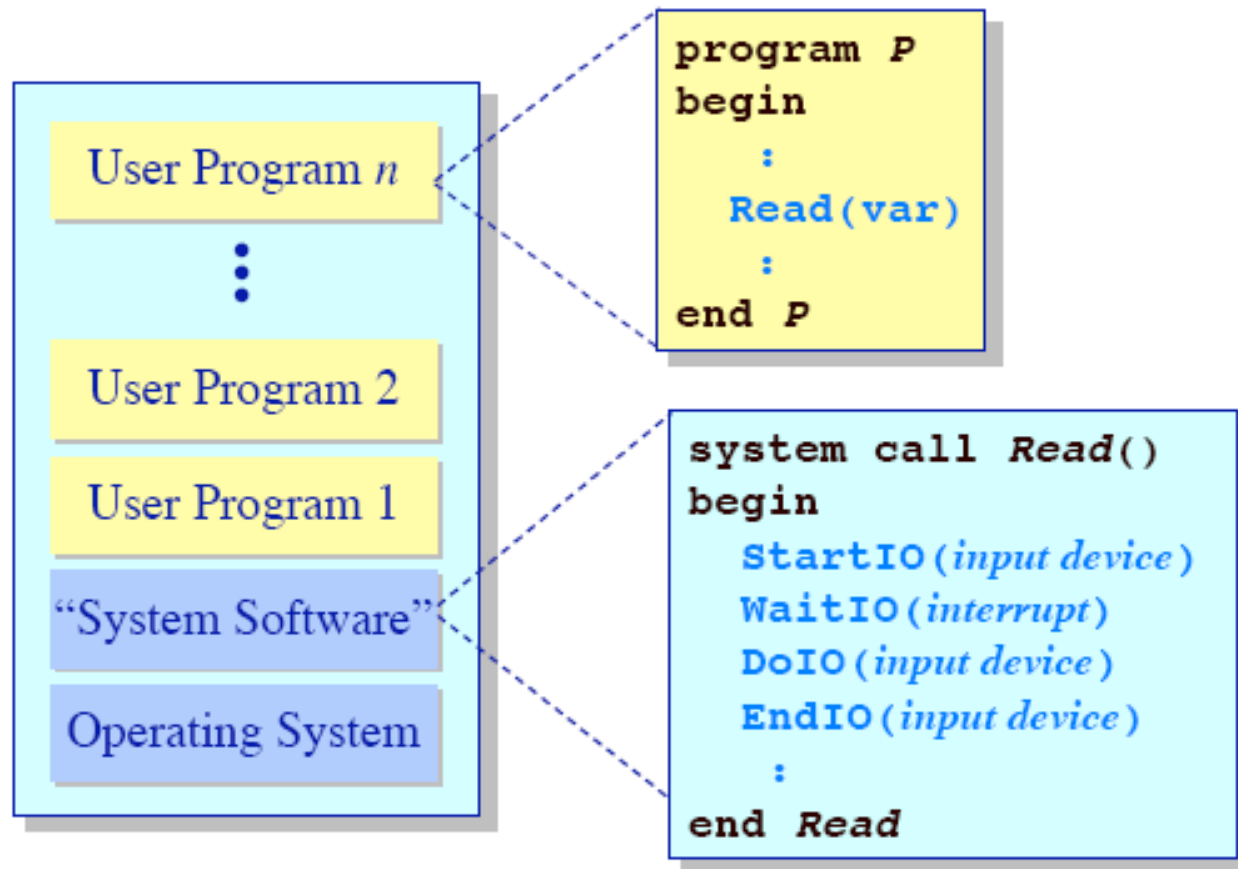
- Nadzoruje jih operater (administrator)
- **Uporabnik ≠ operater**
- Rokovanje s perifernimi napravami
- Krajšanje časa nameščanja (setup time) s primernim grupiranjem podobnih poslov v pakete
- Avtomatsko razporejanje poslov (job sequencing) – avtomatsko prenese nadzor iz enega posla (job) na drugega. To so bili prvi preprosti operacijski sistemi.
- **Residenčni monitor**
 - V začetku ima nadzor monitor
 - Nadzor se prenese na posel
 - Ko se posel zaključi, se nadzor spet prenese na monitor

Multiprogramski paketni sistemi

V primarnem pomnilniku je sočasno več poslovanj in uporaba CPE je med njimi multipleksirana.



Multiprogramski paketni sistemi

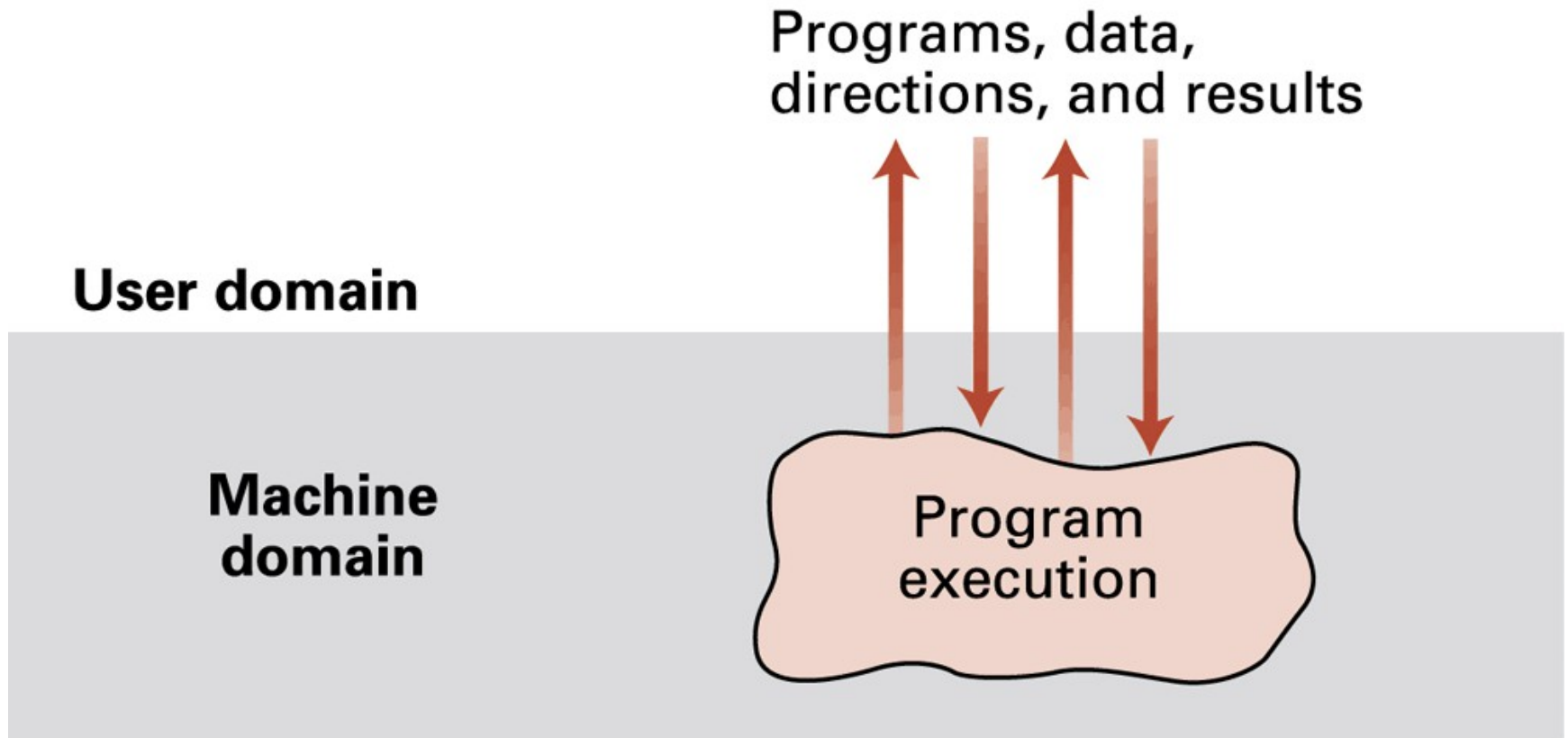


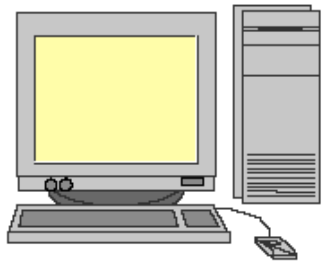
- Sistemska programska oprema nudi storitve računalniškega sistema glede na zahteve s strani uporabniških programov
- Do spremembe konteksta (programa) pride, ko določen paket čaka na zunanji dogodek (npr. V/I napravo)

Sistemi z dodeljevanjem časa (Time sharing)

- Podobno multiprogramskim sistemom, multipleksiranje pa je tako hitro, da navidezno programi tečejo sočasno. Zato lahko poteka uporaba računalnika **interaktivno**.
- **CPU je multipleksirana** med več posli, ki so sočasno v pomnilniku oziroma na disku (CPE je dodeljena poslu le, če se ta nahaja v pomnilniku).
- **Posel (job) je premeščan (swapped)** v pomnilnik ali iz njega na disk.
- Na voljo je **on-line komunikacija** med uporabnikom in sistemom; Ko operacijski sistem konča izvajanje enega ukaza, poišče naslednji “ukaz”, ki ga uporabnik vtipka preko tipkovnice.
- Uporabniki imajo dostop do podatkov in programov s pomočjo sprotnega sistema (**on-line system**).

Interaktivna obdelava

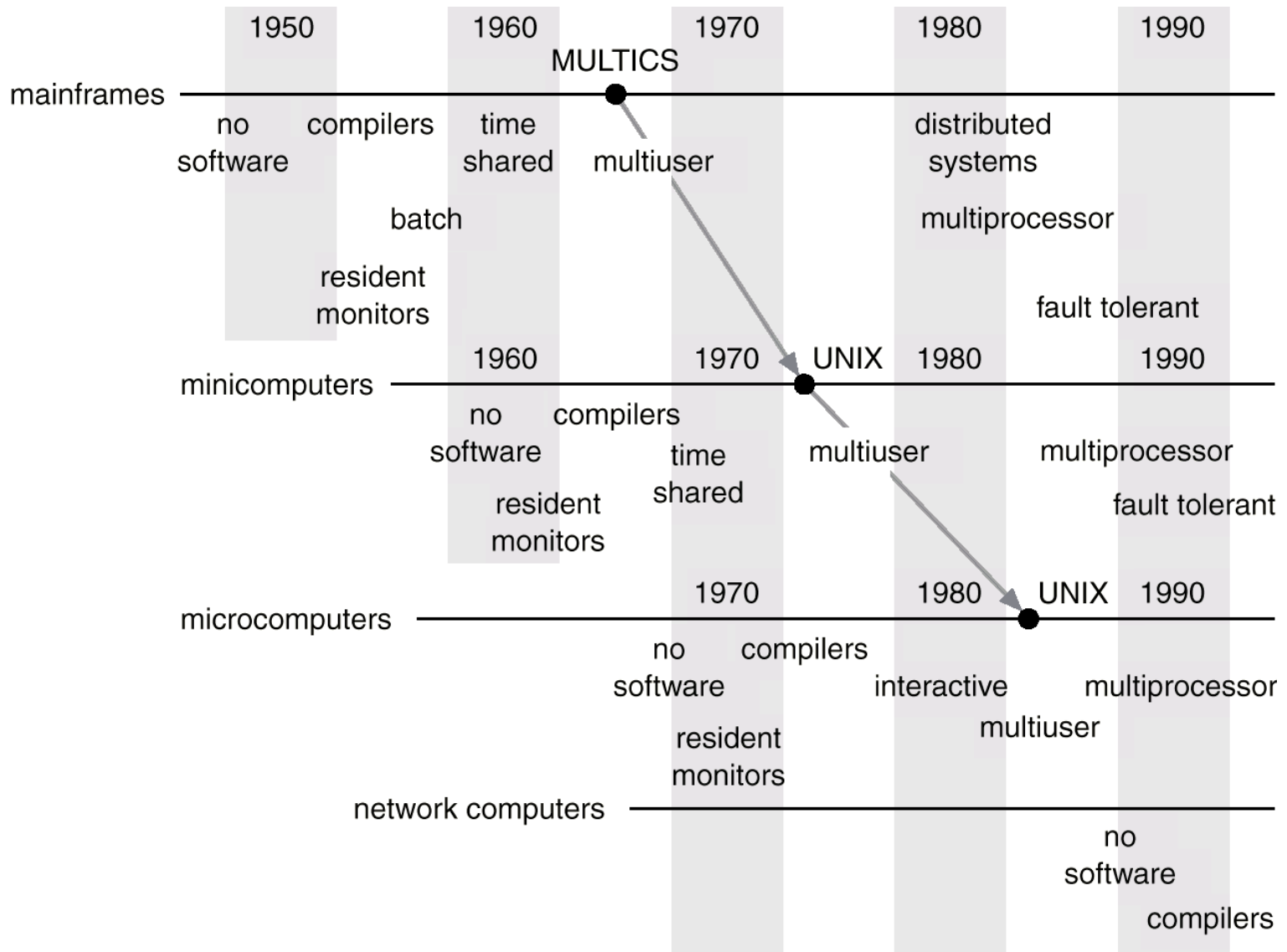




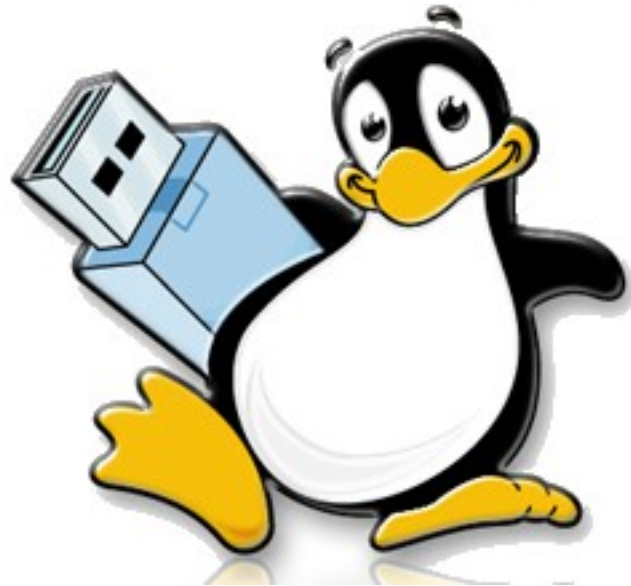
Osebni računalniški sistemi

- *Osebni računalniki* – Računalniški sistemi, namenjeni **enemu uporabniku**.
- **Vhodno izhodne naprave** – tipkovnica, miš, prikazovalnik, tiskalniki,...
- **Prikladnost** uporabniku in odzivnost.
- Možna uporaba **tehnologij**, razvitih za večje računalniške sisteme, pri čemer pa posameznik pogosto ne potrebuje vseh sistemskih zaščitnih mehanizmov, ki so pri večjih sistemih običajni in nujni.

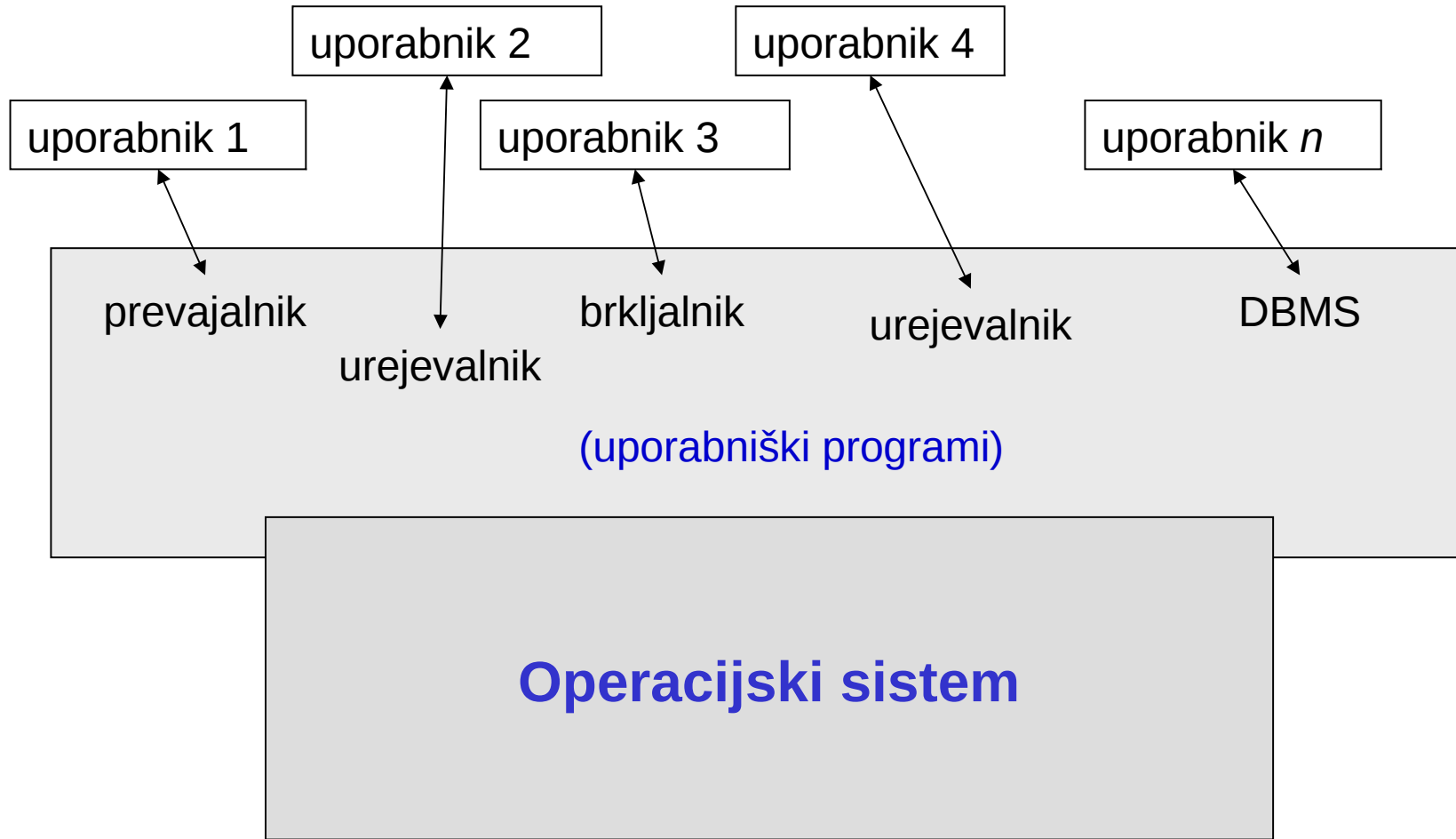
Razvoj pojmov in lastnosti OS



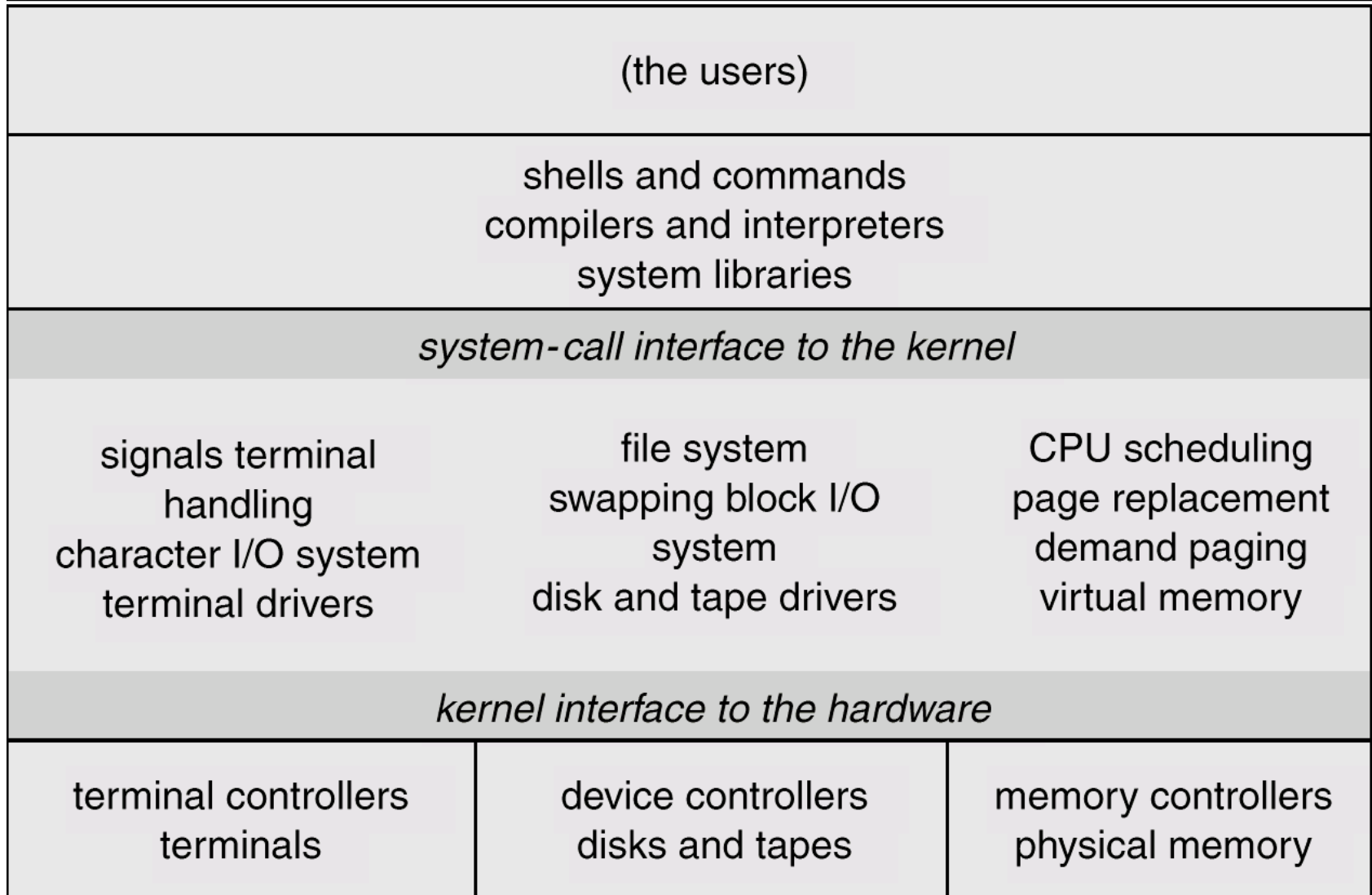
LINUX na USB ključku



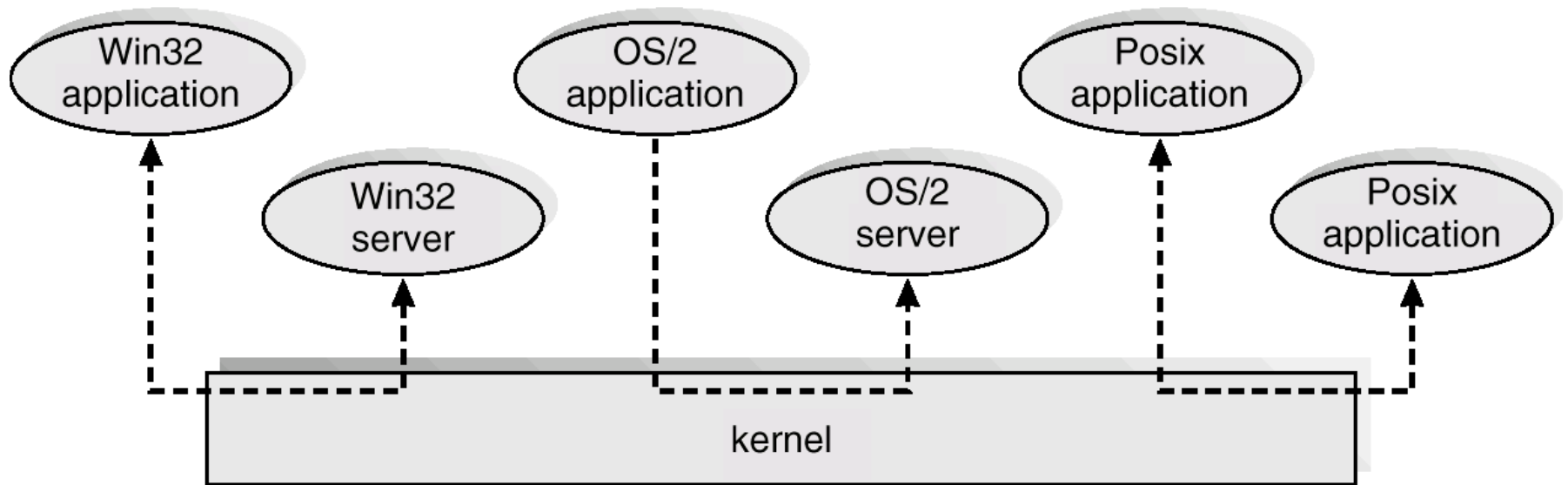
Abstraktni pogled na komponente sistema



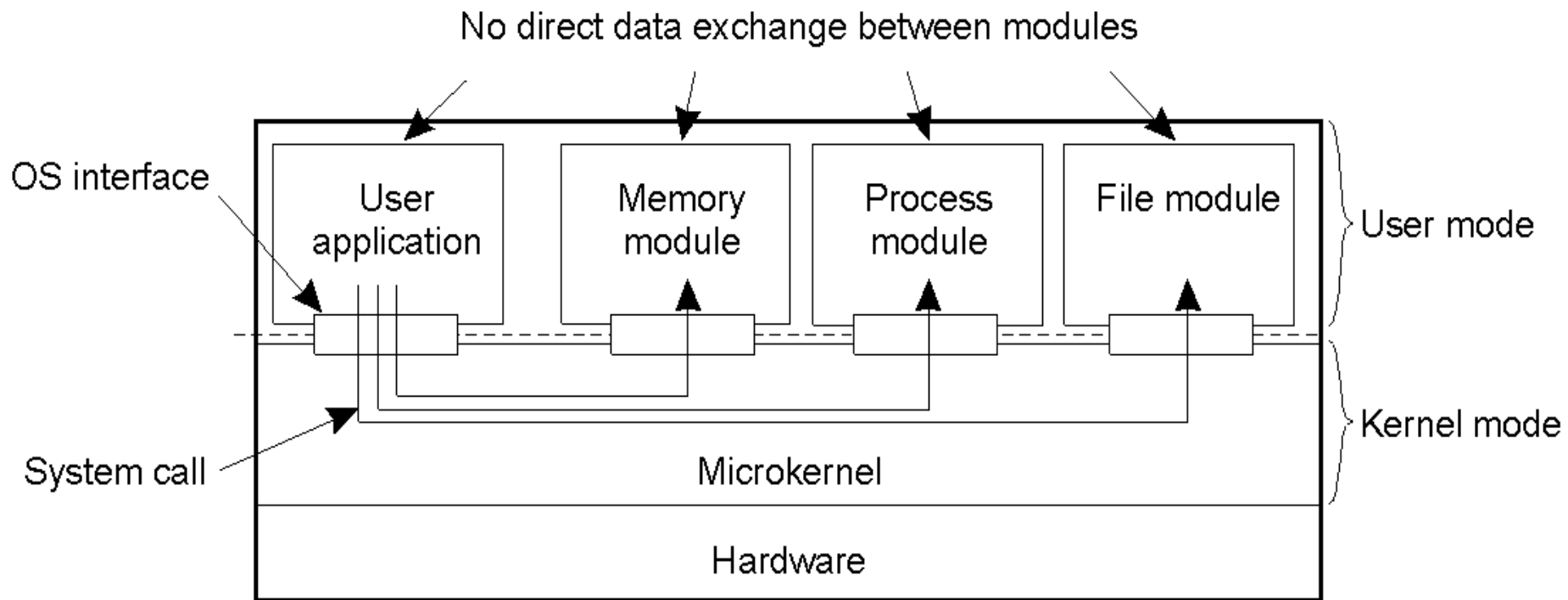
Struktura sistema UNIX



Struktura Windows NT (odjemalec- strežnik)



Enoprocesorski operacijski sistemi



Navidezni stroji

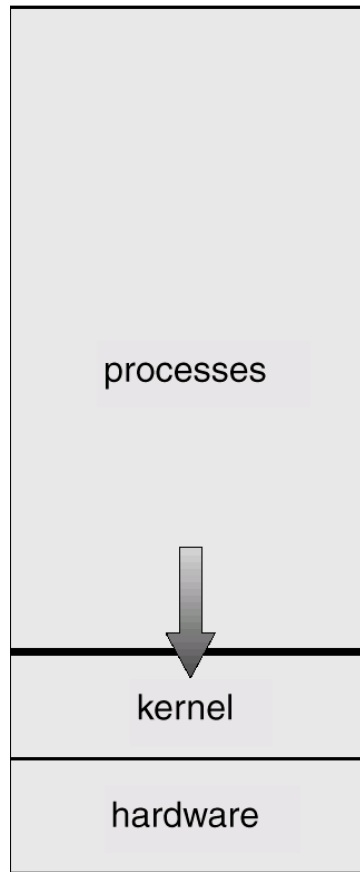
- Navidezni stroj (*virtual machine*) je pripeljal koncept slojne arhitekture do njenega logičnega zaključka. Aparaturno opremo in jedro operacijskega sistema obravnava, kot da je vse skupaj aparaturna oprema.
- Navidezni stroj nudi vmesnik, identičen aparaturni opremi pod njim.
- Tvorimo iluzijo več procesov, od katerih vsak simulira svoj lastni procesor s svojim lastnim (navideznim) pomnilnikom.

Navidezni stroji

- S souporabo virov fizi nega ra unalnika tvorimo navidezne ra unalnike.
 - Razvrš anje CPE daje uporabnikom vtis, kot da ima vsak svoj procesor.
 - “Spooling” in datote ni sistem lahko nudita navidezne (virtualne) periferne naprave.
 - Kot konzolo operaterja uporabljamo obstoje i uporabniški terminal (tipkovnico, zaslon, miško)

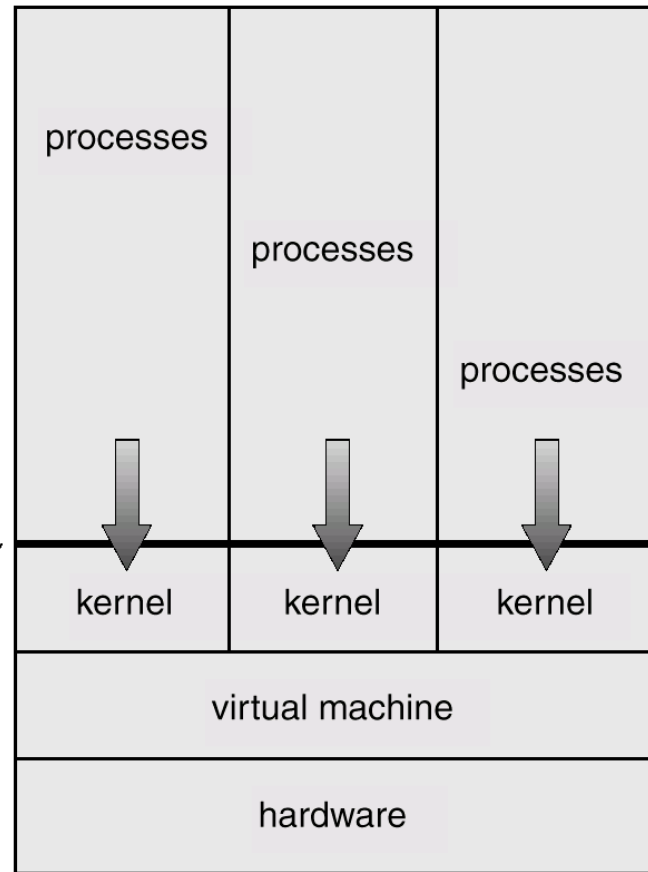
Modeli sistema

Non-virtual Machine



(a)

Virtual Machine



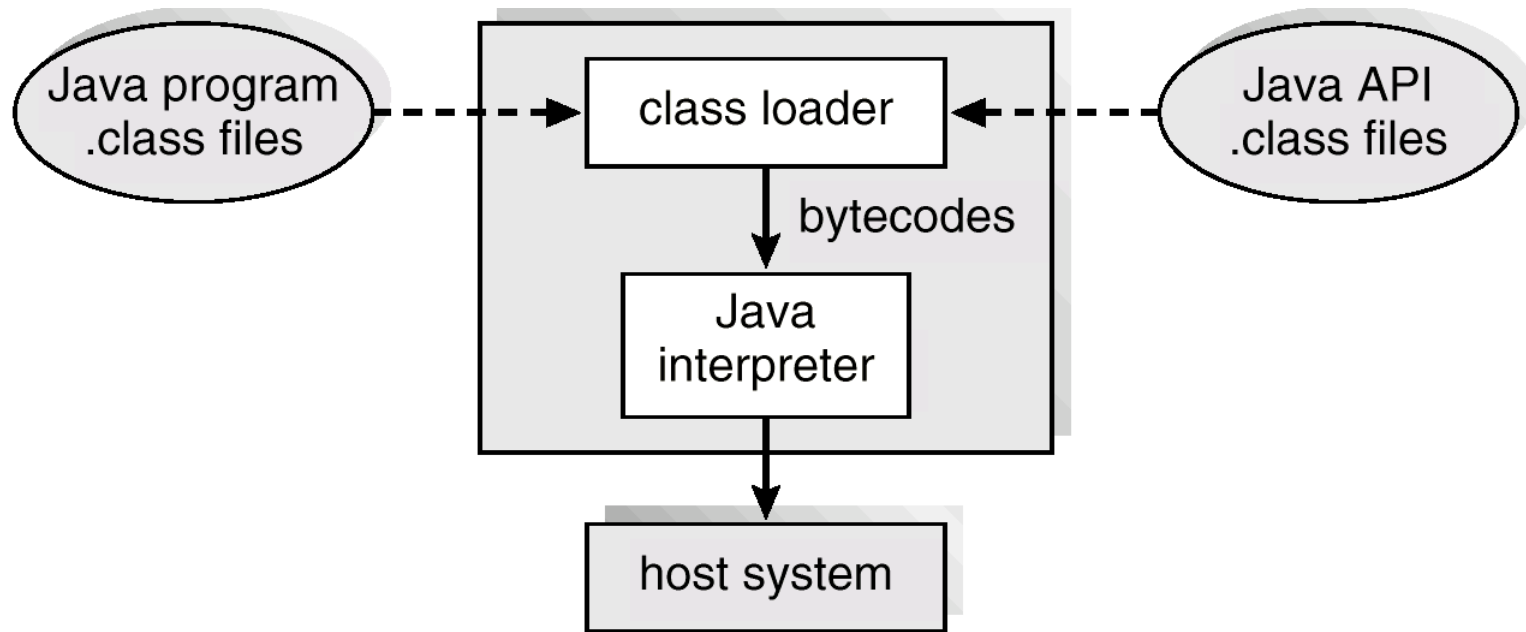
(b)

Prednosti in slabosti navideznih strojev

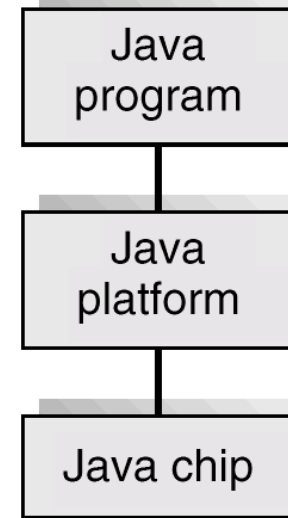
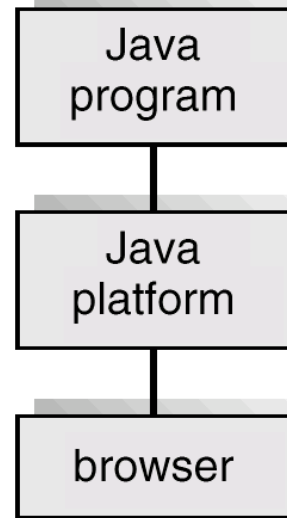
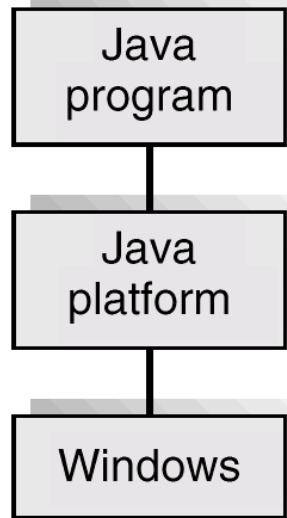
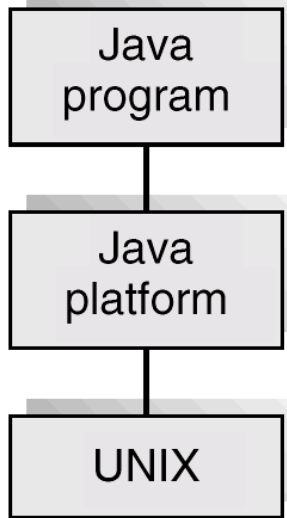
- Koncept navideznega stroja nudi popolno zaščito sistemskih virov, saj je vsak navidezni stroj izoliran od drugih navideznih strojev. Taka izolacija pa ne dovoljuje direktne souporabe virov.
- Sistem z navideznim strojem je odličen za učenje, razvoj in raziskave. Sistemski razvoj izvajamo na navideznih računalnikih namesto na fizičnem in tako ne motimo normalno delovanje računalnika.
- Implementacija navideznega stroja je težavna, saj moramo nuditi točno kopijo stroja na spodnji plasti.



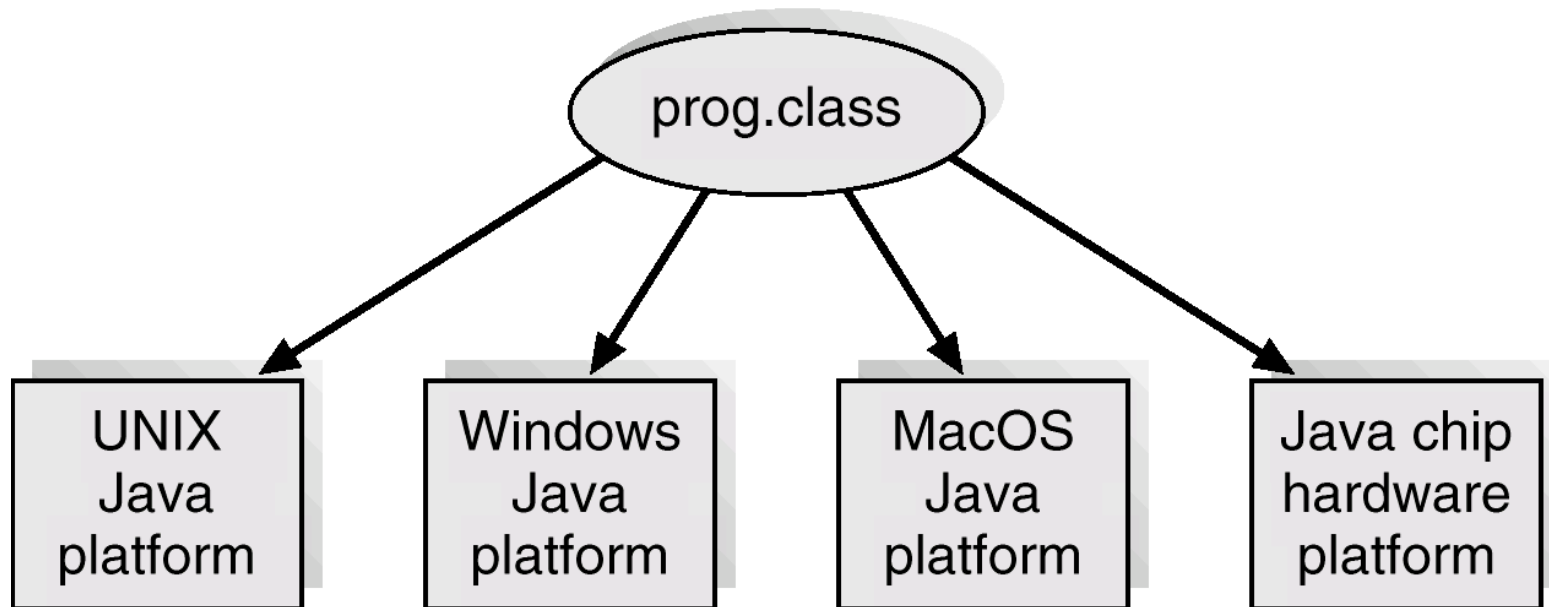
Javanski navidezni stroj



Platforma Jave

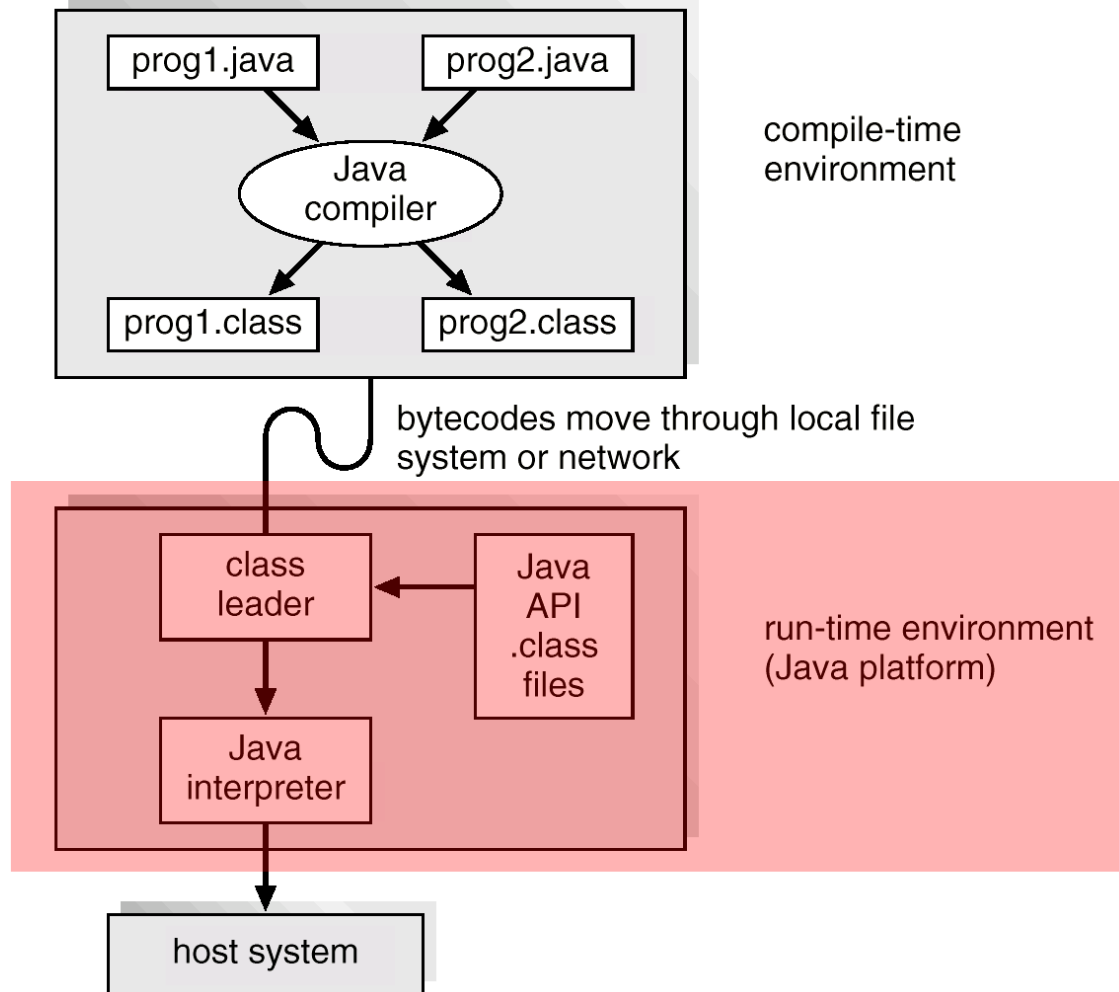


Java na različnih platformah



Javanske datoteke .class na različnih platformah

Koncept Jave – navidezni stroj



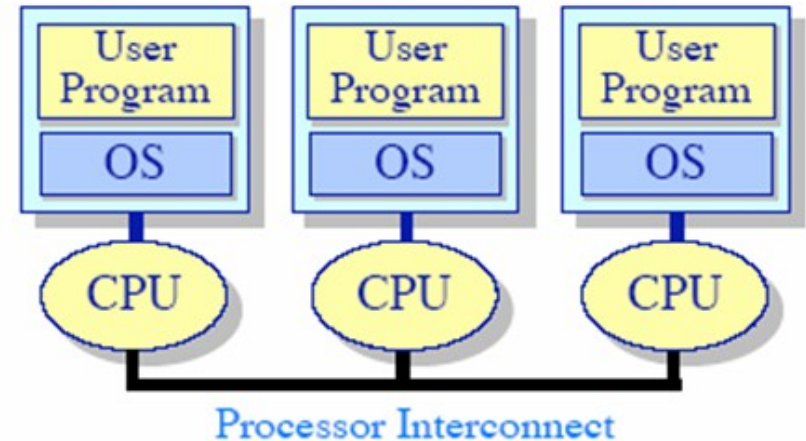
Sistemi za delo v realnem času

- Jih pogosto uporabljamo za krmiljenje naprav v posve enim aplikacijam, kot so na primer krmiljenje znanstvenih poskusov, medicinsko oslikovalni sistemi, industrijski nadzorni sistemi ipd.
- Dobro dolo ene, trdne časovne omejitve.
- *Hard real-time system.*
 - Sekundarni pomnilnik je manjši, lahko ga celo ni. Podatki so v hitrem pomnilniku ali ROM (read-only memory)
 - Prekora itev časa ima lahko usodne posledice za delovanje naprave (roboti, industrijske naprave,...)
 - Splošno namenski operacijski sistemi za to niso primerni.
- *Soft real-time system*
 - Prekora itev časa je mote a, ni pa usodna
 - Uporabno v aplikacijah, kot so navidezna resni nost, multimedijske aplikacije ipd. Zahteva napredne zna ilnosti operacijskega sistema.

Paralelni sistemi

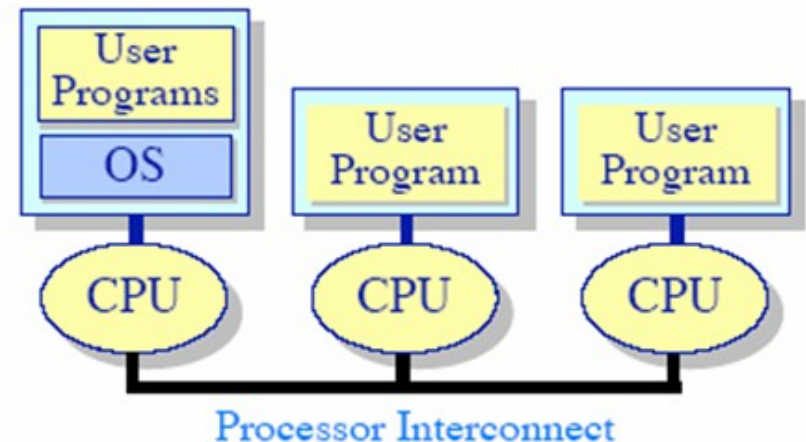
Simetrično multiprocesiranje

Vsak procesor ima svojo kopijo operacijskega sistema
Več poslov izvajamo sočasno



Asimetrično multiprocesiranje

Glavni procesor dodeljuje posle podrejenim procesorjem



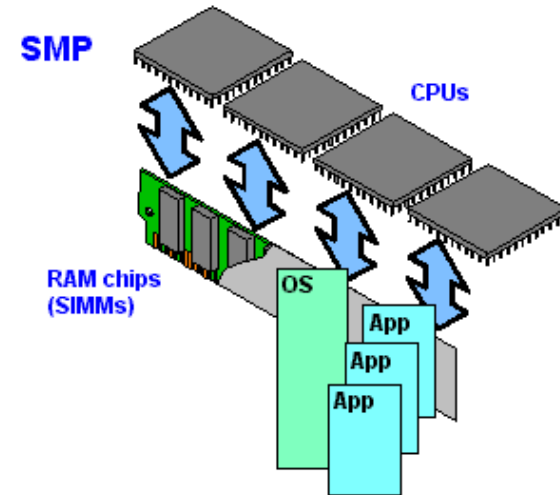
Paralelni sistemi

From Computer Desktop Encyclopedia
© 1998 The Computer Language Co. Inc.

Simetrično multiprocesiranje (SMP)

Več enakovrednih procesorjev dostopa do skupnega pomnilnika
1 OS, ki enakovredno distribuira procese procesorjem

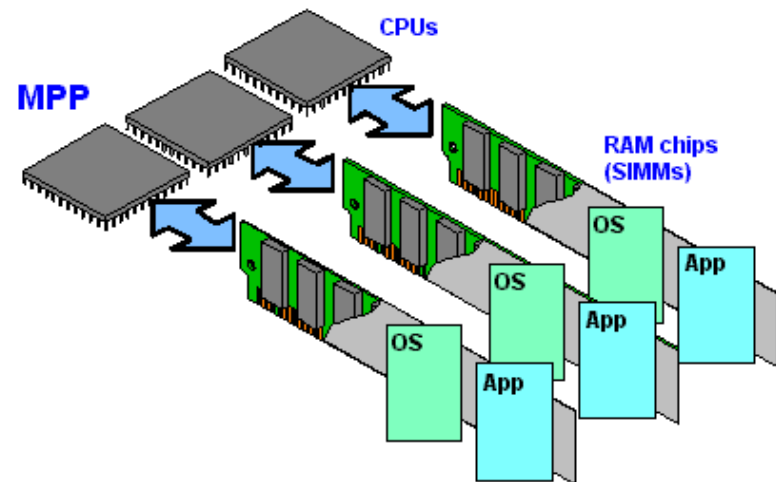
Podpira ga večina modernih OS
(Windows, Linux...)



MPP – Masivno paralelno procesiranje

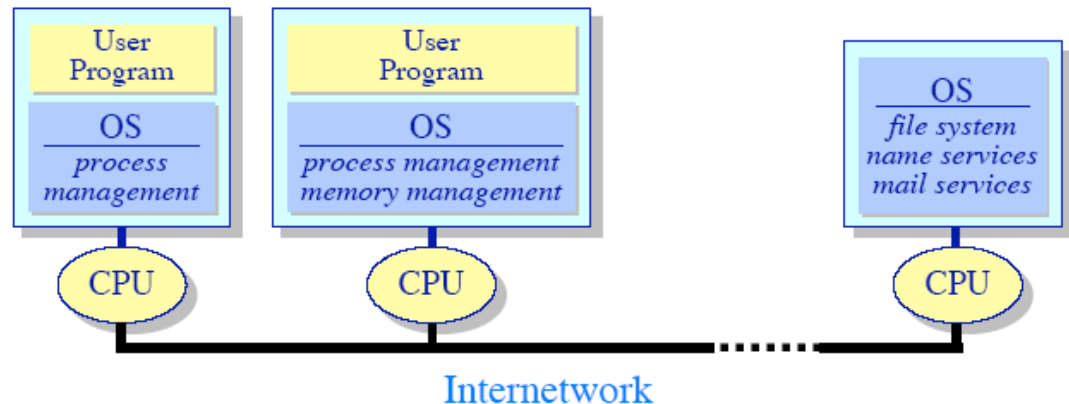
Vsaka CPE vsebuje svojo kopijo OS
in aplikacije, povezane so s hitrim
vodilom.

Problemi, ki jih rešujejo, morajo biti
primerni (možno razbitje na kose...)



Porazdeljeni sistemi

- Računanje je porazdeljeno med več fizičnih procesorjev.
- *Mehko sklopljeni sistemi* – vsak procesor ima lasten lokalni pomnilnik. Procesorji komunicirajo med seboj preko komunikacijskih linij.
- Prednosti porazdeljenih sistemov.
 - Souporaba virov
 - Hitrost računanja – sodelavna razdelitev premena
 - Zanesljivost
 - Komunikacije



Porazdeljeni sistemi

- Omrežni operacijski sistem (Network Operating System)
 - Nudi souporabo datotek
 - Nudi shemo komunikacije
 - Te je neodvisno od drugih računalnikov v omrežju
- Porazdeljeni operacijski sistem
 - Manj avtonomije med računalniki
 - Daje vtis, kot da en sam operacijski sistem nadzoruje omrežje.

Operacijski sistemi bodo nosti

- **Vhodne naprave**

- Vnos ukazov z glasom, zasloni, ob utljivi na dotik, nadzor z daljinskim upravljanjem (remote controls), mobilnimi telefoni ipd

- **Upravljanje s podatki**

- Shranjevanje podatkov lokalno in na oddaljenih strežnikih
- Ne skrbimo za podrobnosti o pomnilnih napravah

- **Fizi na identifikacija uporabnika ob vstopu**

- na primer s prstnimi odtisi, skeniranjem zenice ipd

