

Distribuirani sistemi, klijent/serveri i klasteri

Operativni sistemi 2

Institut za matematiku i informatiku

Institut za matematiku i informatiku
Prirodno-matematički fakultet, Kragujevac

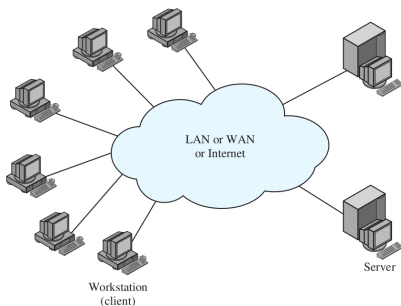
Novembar 2016. god.

O čemu će biti reči?

- 1 Klijent/server
- 2 Komunikacija u dist. sistemima
- 3 Klasteri

Klijent/server računarstvo

- **Klijent** računari su jednokorisnički PC-ji ili radne stanice sa GUI. *Windows, OS X, Android, iOS, Web browser...*
- **Serveri** obezbeđuju klijentima skup zajedničkih usluga
- **Mreža** je takođe element klijent/server okruženja



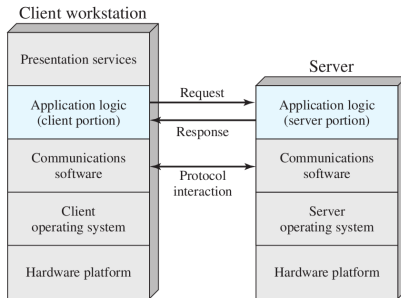
Klijent/server računarstvo

Karakteristike

- 1 Oslanja se na poznate aplikacije na klijentskoj strani
- 2 Centralizovane baze podataka. Veći deo firme oslobođen brige o skupim centralizovanim sistemima
- 3 Otvoreni sistemi koji se lako nadgrađuju. Mešanje opreme razlih *vendor*-a
- 4 Umrežavanje i bezbednost kao važni parametri

Klijent/server aplikacije

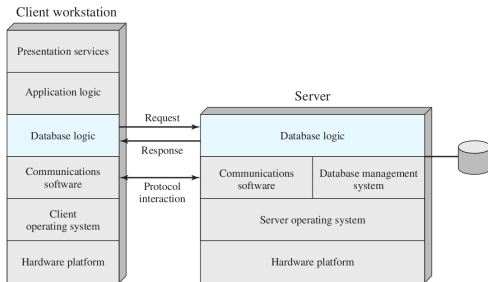
- Više tipova platformi koje rade u istom okruženju
- Sve je ok dok **klijent i server dele iste protokole**
- Glavno pitanje je **podela posla između klijenta i servera**



Klijent/server aplikacije

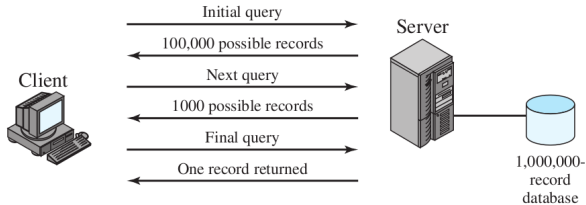
Aplikacije baze podataka

- Server je odgovoran za održavanje baze podataka
- Klijent ima aplikacije koje koriste bazu, recimo na nivou SQL upita
- **Primer:** svi intenzivni proračuni se odigravaju na klijentu, dok se server samo koristi za smeštaj podataka

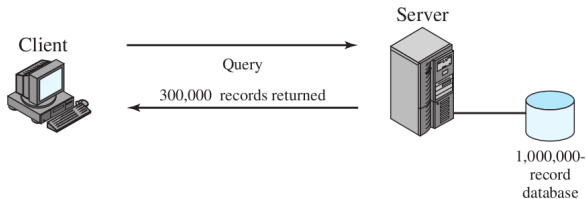


Klijent/server aplikacije

Primer ispravnog i pogrešnog korišćenja baze

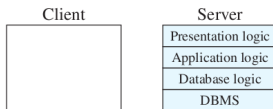


(a) Desirable client/server use

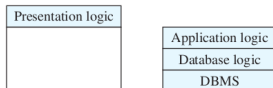


(b) Misused client/server

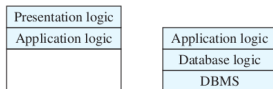
Klase klijent/server aplikacija



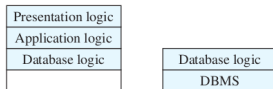
(a) Host-based processing



(b) Server-based processing



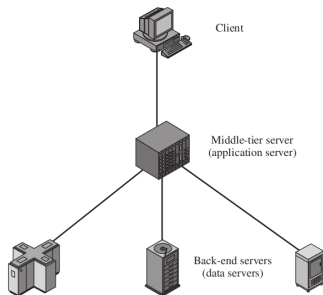
(c) Cooperative processing



(d) Client-based processing

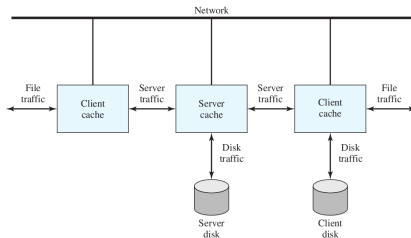
Troslojna klijent/server arhitektura

- Dodaje se **aplikacioni server**
- Mašine srednjeg sloja su mrežni prolaz između klijenata i različitih pozadinskih servera
- Prevođenje upita, objedinjavanje rezultata



Konzistentnost fajl keša

Zahtev se prvo prosleđuje kešu klijenta ("saobraćaj fajla"). Ako se tamo ne zadovolji, prosleđuje se lokalnom disku ("saobraćaj diska"). Ako ga ni tamo nema, onda se zahtev prosleđuje serveru ("saobraćaj servera") koji ima i svoj keš.

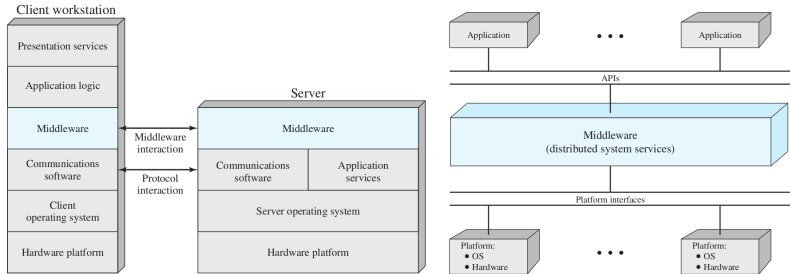


Rešenja za konzistentnost keša

- 1 Tehnika *lock*-ovanja fajlova, konzistentnost po cenu performansi
- 2 OS *Sprite* - Obaveštavanje procesa putem poruka o nedostatku konzistencije

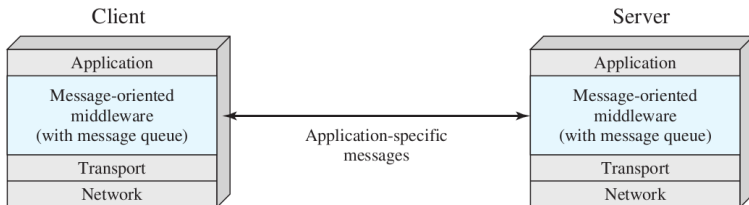
Arhitektura *middleware*-a

- Primer skrivanja nepodudranosti **različitih SQL dijalekata**
- Primer skrivanja nepodudranosti **klaster *resource manager*-a preko *grid servisa***



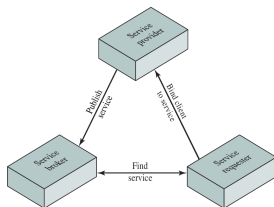
Distribuirano prosleđivanje poruka

- Primitive Send i Receive
- Pouzdano i nepouzdana prosleđivanje poruka
- Blokirajući/neblokirajući

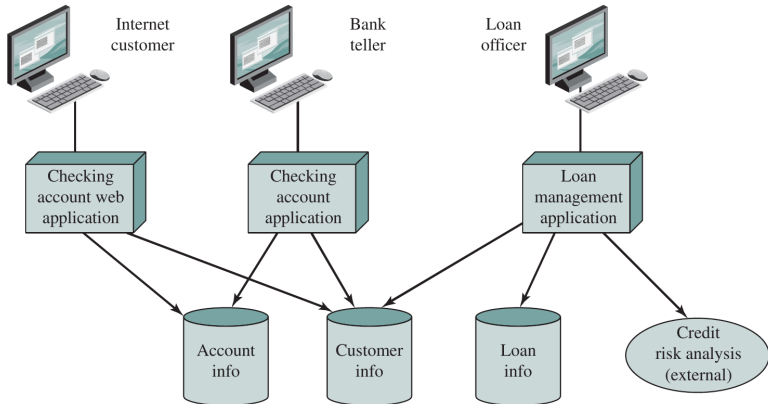


Service Oriented Architecture - SOA

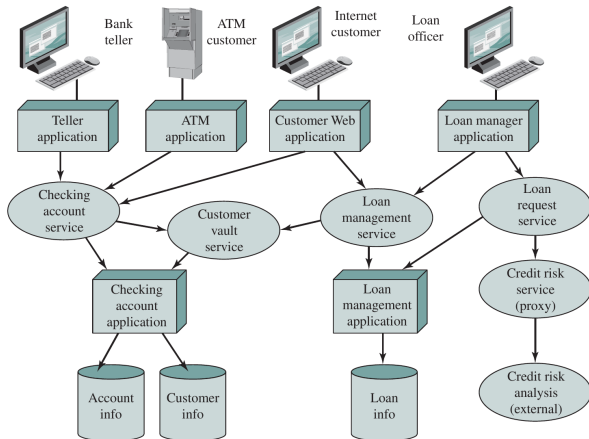
1. Provajder usluge
 2. Korisnik usluge
 3. Posrednik (*Service broker*)
- Gruba podela nadležnosti (*coarse-grained*)
 - Dizajn zasnovan na implementaciji interfejsa



Primer migracije na SOA arhitekturu - PRE



Primer migracije na SOA arhitekturu - POSLE



Distribuirano prosleđivanje poruka

- Zašto ne funkcionišu mehanizmi sa deljenom memorijom?
- Osnovne primitive (Send, Receive)
- Kako rešiti međusobno isključenje putem poruka?
- Pouzdanost ili nepouzdanost?
- Blokiran ili neblokiran poziv?

Remote Procedure Call (RPC)

- Pristupanje uslugama koje su stacionirane na drugim sistemima
- Korisnik tog poziva može da bude svestan ali i ne mora
- CALL $P(X, Y)$, X -argumenti, Y -povratne vrednosti

Pitanja projektovanja

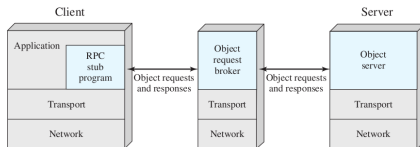
- 1 **Prosleđivanje parametara** - vrednost/referenca
- 2 **Predstavljanje parametara** - L/B endian, ASCII/UNICODE...
- 3 **Povezivanje** klijenta i servera - stalno ili nestalno
- 4 **Sinhrono/asinhrono**
- 5 **Objektno orijentisani mehanizmi** - COM, CORBA

Remote Procedure Call (RPC)

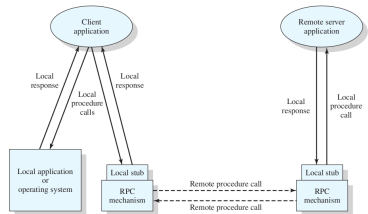
Slika: **Levo:** RPC i objektno model, **Desno:** Mehanizam RPC



(b) Remote procedure calls



(c) Object request broker



Klasteri

Prednosti klaster sistema nad SMP sistemima

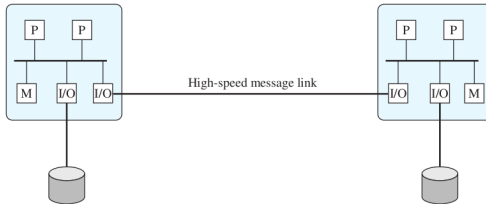
Definicija (Klaster)

Skup međusobno povezanih potpunih računara koji rade zajedno kao ujedinjeni resurs, stvarajući iluziju kao da su *jedan računar*.

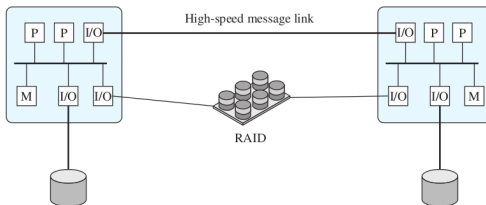
- 1 Apsolutna skalabilnost
- 2 Rastuća skalabilnost
- 3 Visoka raspoloživost
- 4 Bolji odnos cena/performance

Konfiguracije klastera

Bez deljenog diska i sa njim



(a) Standby server with no shared disk



(b) Shared disk

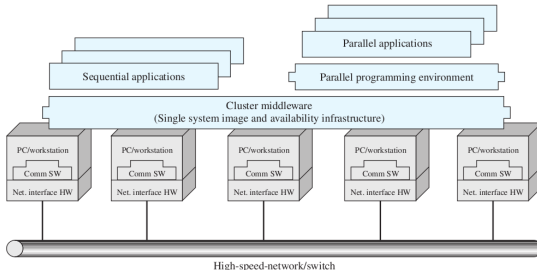
Konfiguracije klastera

HP-ova kategorizacija klastera

Clustering Method	Description	Benefits	Limitations
Passive Standby	A secondary server takes over in case of primary server failure.	Easy to implement.	High cost because the secondary server is unavailable for other processing tasks.
Active Secondary	The secondary server is also used for processing tasks.	Reduced cost because secondary servers can be used for processing.	Increased complexity.
Separate Servers	Separate servers have their own disks. Data is continuously copied from primary to secondary server.	High availability.	High network and server overhead due to copying operations.
Servers Connected to Disks	Servers are cabled to the same disks, but each server owns its disks. If one server fails, its disks are taken over by the other server.	Reduced network and server overhead due to elimination of copying operations.	Usually requires disk mirroring or RAID technology to compensate for risk of disk failure.
Servers Share Disks	Multiple servers simultaneously share access to disks.	Low network and server overhead. Reduced risk of downtime caused by disk failure.	Requires lock manager software. Usually used with disk mirroring or RAID technology.

Pitanja projektovanja OS-a

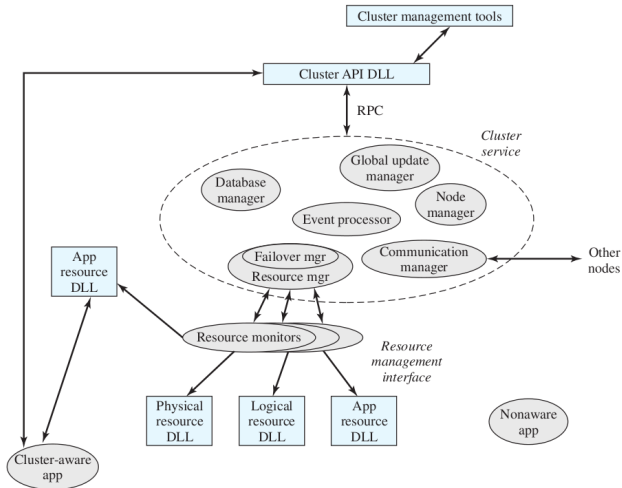
- 1 Upravljanje otkazima - predaja i vraćanje
- 2 Uravnotežavanje opterećenja
- 3 Paralelizam obrade
 - Paralelizovani kompajler ili jezik
 - Paralelizovane aplikacije
 - Parametarsko izračunavanje



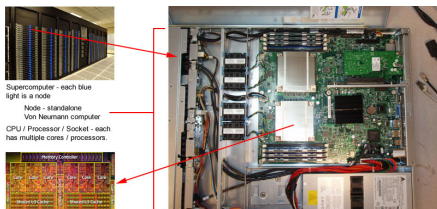
Karakteristike klastera

- 1 Jedinствена ulazna tačka
- 2 Jedinствена fajl hijerarhija
- 3 Jedinствено mesto upravljanja
- 4 Jedinствено virtuelno umrežavanje
- 5 Jedinствен *job manager*, npr. Torque
- 6 Jedinствeni UI prostor
- 7 Jedinствен prostor procesa (na nekim sistemima)
- 8 Preseljenje procesa

Windows klaster server



Linux Beowulf klaster



Prednosti

- 1 Nema posebno projektovanih komponenti
- 2 Moguće su komponente od različitih *vendor*-a
- 3 UI komponente se lako mogu proširivati
- 4 Osnovni softver je *besplatan* i *otvoren*
- 5 Besplatni alati za distribuirani kompjuting
- 6 Javno dostupni *patch*-evi