



Diplomski studij

Informacijska i  
komunikacijska tehnologija  
Telekomunikacije i  
informatika

Računarstvo  
Računarska znanost  
Programsko inženjerstvo i  
informacijski sustavi

# Raspodijeljeni sustavi

Pitanja za provjeru znanja s odgovorima  
1. blok predavanja

Ak.g. 2010./2011.

**Napomena:**

*Preporučena literatura su bilješke s predavanja.*

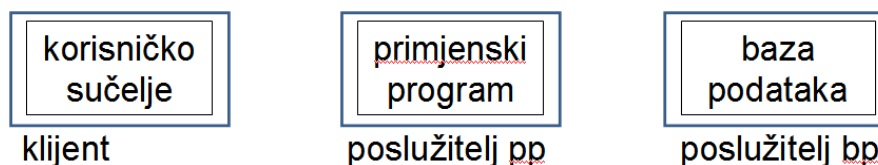
**Zadatak 1** Objasnite pojam skalabilnosti raspodijeljenog sustava.

Raspodijeljeni sustav je skalabilan ukoliko posjeduje sposobnost prilagodbe povećanom broju korisnika i sredstava, njihovoj rasprostranjenosti te načinu upravljanja sustavom.

**Zadatak 2** Objasnite pojam migracijske transparentnosti raspodijeljenog sustava.

Za raspodijeljeni sustav kažemo da posjeduje migracijsku transparentnost ukoliko on prikriva promjenu lokacije nekog sredstva na način da ta promjena ne utječe na način pristupa tom sredstvu.

**Zadatak 3** Skicirajte trorednu arhitekturu klijent-poslužitelj te na proizvoljnom primjeru aplikacije objasnite ulogu svake razine u cjelokupnoj arhitekturi.



Primjer su aplikacije weba, gdje klijentski program koji se izvodi na klijentskom računalu nikada ne pristupa direktno bazi podataka, već posredno preko aplikacije weba. Klijentski program prikazuje korisničko sučelje i komunicira s aplikacijom weba koja obavlja cjelokupnu logiku usluge i pristupa potrebnim podacima.

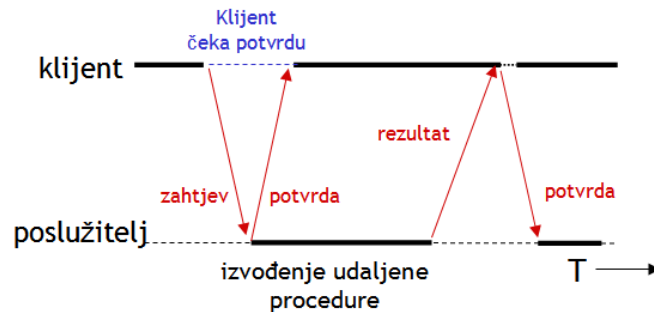
**Zadatak 4** Objasnite razliku između sinkrone i asinkrone komunikacije.

Dok je kod sinkrone komunikacije pošiljalatelj blokiran nakon slanja poruke sve do primitka potvrde o isporuci, kod asinkrone komunikacije pošiljalatelj nije blokiran te nastavlja procesiranje odmah nakon slanja.

**Zadatak 5** Navedite obilježja komunikacije *socketom* UDP.

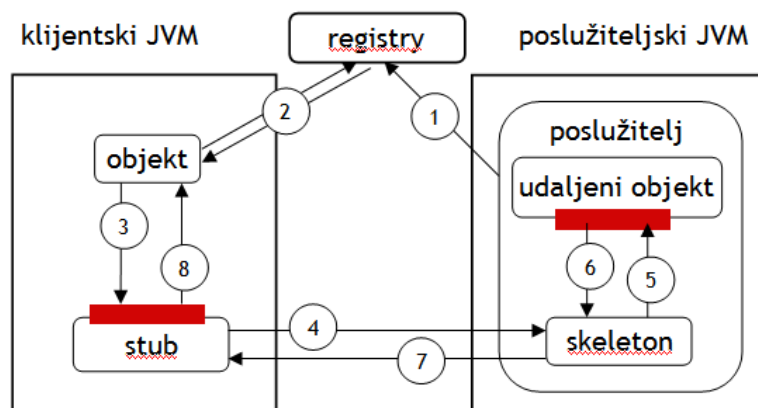
Ova komunikacija se temelji na modelu klijent-poslužitelj, gdje oba moraju istovremeno biti aktivna da bi se komunikacija mogla ostvariti. Komunikacije je tranzijentna i asinkrona, a može se koristiti za implementaciju komunikacije na načelu *pull* ili *push*.

- Zadatak 6** Skicirajte tijek komunikacije između klijenta i poslužitelja te objasnite odgođeni sinkroni poziv udaljene procedure RPC (*Remote Procedure Call*).



Kod odgođenog sinkronog poziva udaljene procedure, klijent nije blokiran dok čeka rezultat izvođenja, već nastavlja s radom nakon uspješnog primitka potvrde. Kasnije mu poslužitelj šalje rezultat koristeći drugi asinkroni poziv udaljene procedure.

- Zadatak 7** Skicirajte model pozivanja udaljene metode Java RMI (*Remote Method Invocation*). Navedite korake u komunikaciji potrebne da bi klijent pozvao metodu dostupnu na poslužitelju, uz pretpostavku da je klasa stub već instalirana na klijentskoj strani.



Koraci u komunikaciji su sljedeći:

1. Poslužitelj registrira udaljeni objekt pod odabranim imenom.
2. Klijent od *registry*a traži referencu na udaljeni objekt koristeći registrirano ime.
3. Klijent poziva metodu *stuba* dostupnu na klijentskom računalu.
4. *Stub* serijalizira parametre i šalje ih *skeletonu*.
5. *Skeleton* deserijalizira parametre i poziva metodu udaljenog objekta.
6. Udaljeni objekt vraća rezultat izvođenja metode *skeletonu*.
7. *Skeleton* serijalizira rezultat i šalje ga *stubu*.
8. *Stub* deserijalizira rezultat i dostavlja ga klijentu.

- Zadatak 8** Skicirajte i objasnite primjer komunikacije porukama između dva procesa/objekta (primatelja i pošiljatelja). Kakva je komunikacija porukama s obzirom na vremensku ovisnost primatelja i pošiljatelja?



U komunikaciji između pošiljatelja i primatelja rep sudjeluje kao posrednik. Pošiljatelju se u načelu garantira isporuka poruke u primateljev rep, ali ne i isporuka poruke primatelju. Primatelj može pročitati poruku iz repa u bilo kojem budućem trenutku. Stoga su pošiljatelj i primatelj poruke vremenski neovisni.

- Zadatak 9** Objasnite razliku u obilježjima komunikacije između dva komunikacijska modela podržana s JMS (*Java Messaging Service*)?

JMS podržava komunikaciju porukama i model objavi-pretplati. Obje vrste komunikacije su vremenski neovisne zato što pošiljatelj i primatelj ne moraju istovremeno biti dostupni. Kod komunikacije porukama pošiljatelj mora znati identifikator odredišta, dok je kod modela objavi-pretplati komunikacija anonimna. Komunikacija je perzistentna i asinkrona u oba slučaja. Komunikacija se pokreće na načelu *pull* kod komunikacije porukama, a na načelu *push* kod modela objavi-pretplati.

- Zadatak 10** Objasnite opći format poruka protokola HTTP. Navedite kako glasi potpun i apsolutan URI koji identificira resurs zatražen u zahtjevu, ako prva 2 retka HTTP zahtjeva sadrže sljedeće podatke:

```
GET /predmet/rassus HTTP/1.1
Host: www.fer.hr
```

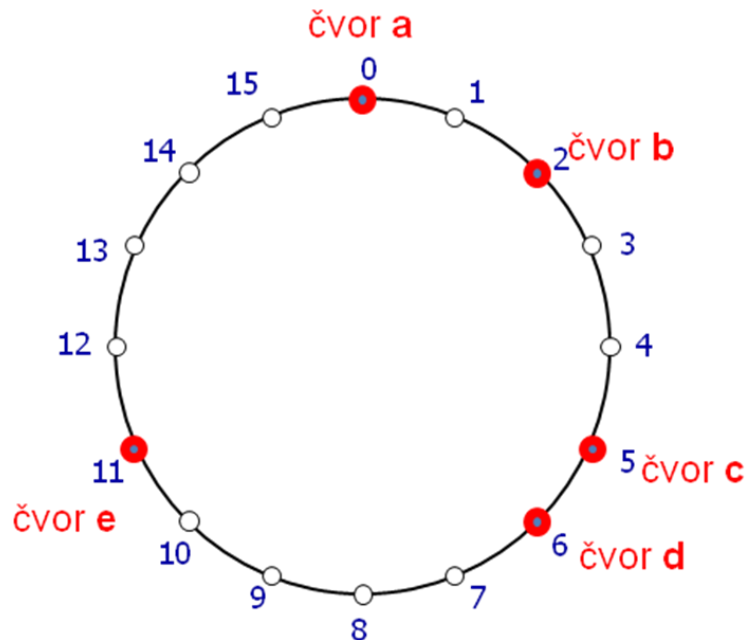
Opći format poruka protokola HTTP sastoji se od početnog retka, polja zaglavlja te tijela poruke. Potpun i apsolutan URI je `http://www.fer.hr/predmet/rassus`.

- Zadatak 11** Korisnik nakon ispunjavanja obrasca na Web-u odabire opciju Submit, čime pošalje podatke Web-poslužitelju na adresu `www.tel.fer.hr/obrazac/accept` korištenjem protokola HTTP verzije 1.1. Kojim se HTTP zahtjevom šalju podaci poslužitelju i kako je definiran prvi redak zahtjeva?

Podaci se šalju zahtjevom POST. Prvi redak je definiran na sljedeći način:  
POST /obrazac/accept HTTP 1.1 .

**Zadatak 12** U strukturiranom P2P sustavu *hash* tablica ključeva koji omogućuju pronalaženje podataka raspodijeljena je na 5 čvorova (a, b, c, d, e) u prostoru identifikatora veličine  $N = 16$ , kojima su dodijeljeni identifikatori prema funkciji  $H_1$  kako slijedi:

Čvor a: 0, Čvor b: 2, Čvor c: 5, Čvor d: 6 i Čvor e: 11.



Ukoliko se neki podatak pridijeli funkcijom  $H_2$  identifikatoru 8, objasnite na koji će se čvor taj podatak pohraniti?

Kako ne postoji čvor s identifikatorom 8, podatak će se pohraniti na prvi sljedeći čvor, a to je čvor e s identifikatorom 11.

**Zadatak 13** Definirajte što su to nestrukturirana imena i navedite barem 2 načina pronalaženja pristupnih točaka pomoću ovakvih imena.

Nestrukturirana imena su najčešće slučajni niz bitova. Ona sama ne sadrže informacije o tome kako pronaći pristupnu točku pa se zbog toga koriste sljedeći načini pronalaženja pristupnih točaka s nestrukturiranim imenima<sup>1</sup>:

1. jednostavni načini (samo u lokalnim mrežama)
  - a. opće razasijlanje (*broadcast*) i
  - b. višeodredišno razasijlanje (*multicast*),
2. pokazivači prema naprijed (*forwarding pointers*),
3. domaća lokacija (*home location*),
4. raspodijeljene *hash* tablice (*distributed hash tables*) i
5. hijerarhijski pristup.

<sup>1</sup> U odgovoru je potrebno navesti samo dva od pet mogućih načina pronalaženja pristupnih točaka s nestrukturiranim imenima.

**Zadatak 14**      Objasnite što je to razlučivanje imena na primjeru strukturnog imenovanja.

Razlučivanje imena je proces pronalaženja čvora iz zadanog puta. Primjer strukturnog imenovanja je datotečni sustav koji je zapravo usmjereni graf kod kojega se do svakog čvora može doći preko puta koji je predstavljen nizom oznaka odijeljenih specijalnim znakom („\“ ili „/“). Ovisno o početnom čvoru put može biti apsolutni (počinje specijalnim znakom) ili relativni (ne počinje specijalnim znakom). U datotečnom sustavu postoje alternativna imena (*aliasi*) tako da se do nekih čvorova može doći preko 2 ili više različitih puteva.