

J2EE Clustering

Pojam klastera

- grupa međusobno povezanih računara koji funkcionišu tako da se mogu posmatrati kao jedan sistem koji pruža neki servis
- klaster se koristi kao
 - sredstvo za unapređenje performansi
 - sredstvo za unapređenje pouzdanosti
 - jeftinije rešenje u odnosu na jedan računar ekvivalentnih mogućnosti

Pojam klastera

- povezivanje čvorova u klasteru
 - najčešće u lokalnoj mreži
 - može i distribuirano

Vrste klastera

- high-availability (failover) cluster
 - redundantni hardver: veća pouzdanost
 - minimum 2 računara
 - eliminiše SPoF (single point of failure)

Vrste klastera

- load balancing
 - raspodela opterećenja na više čvorova
 - različiti algoritmi raspodele

Vrste klastera

- high performance clusters
 - namenjeni za specifične poslove masovne paralelne obrade podataka
 - sprežanje čvorova
 - tightly coupled
 - loosely coupled
 - grid computing

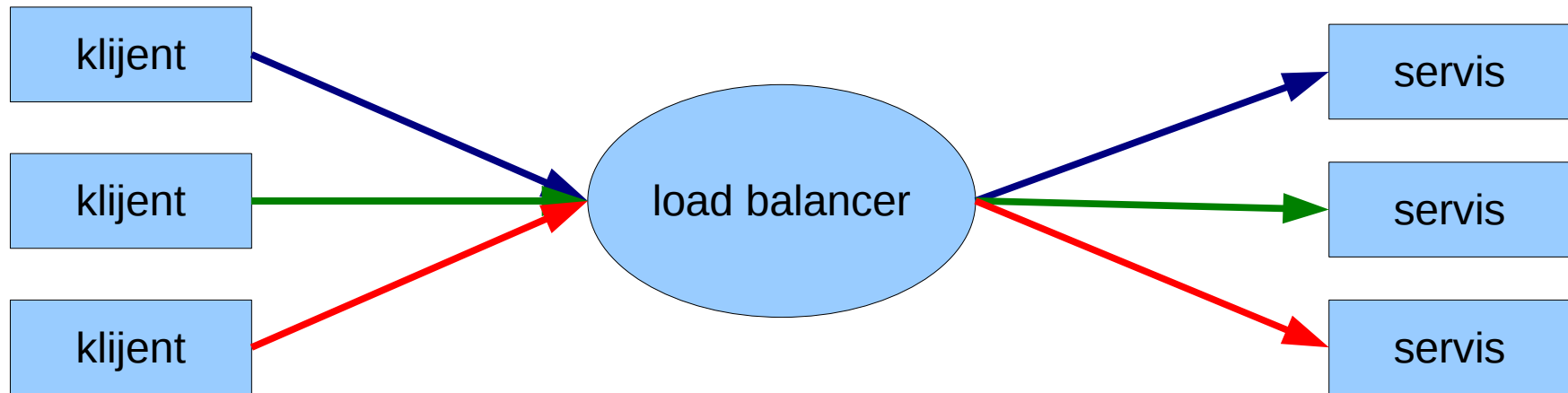


Vrste klastera

- granica između high-availability i scalability klastera nije čvrsta
 - dodavanjem novih čvorova u klaster se, u principu, povećava i dostupnost i skalabilnost

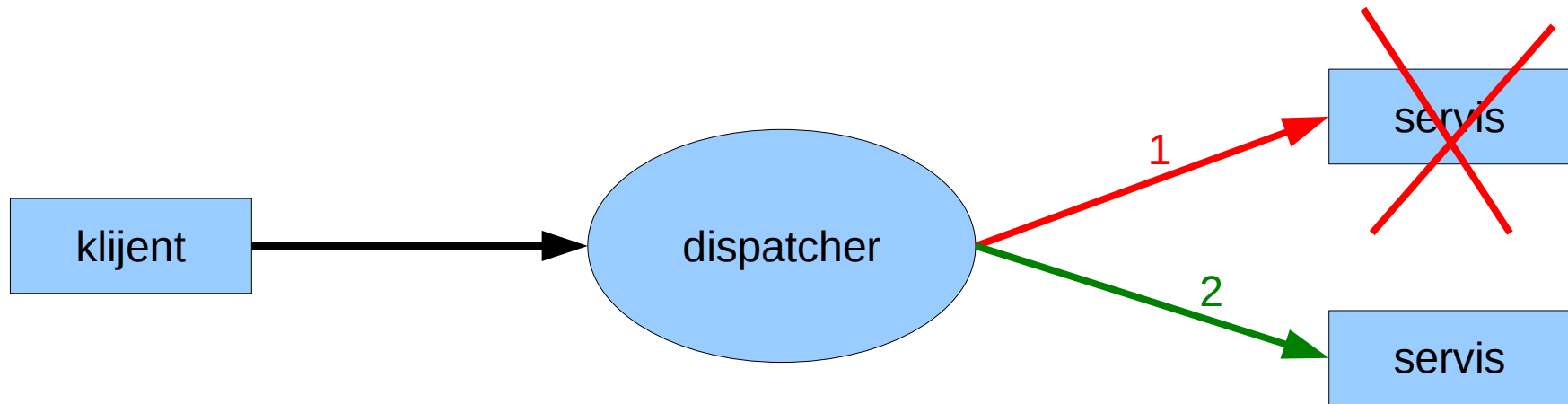
Principi rada u klasteru

- load balancing



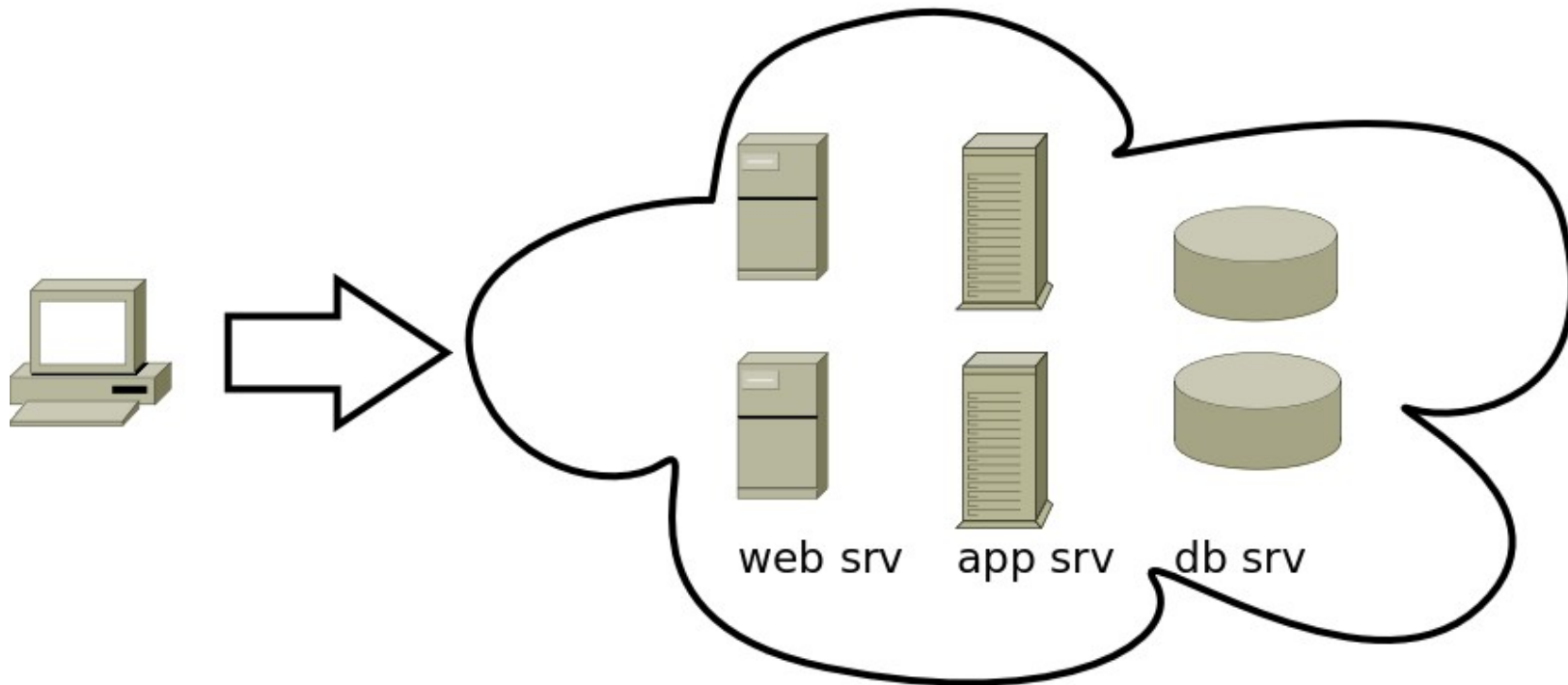
Principi rada u klasteru

- failover



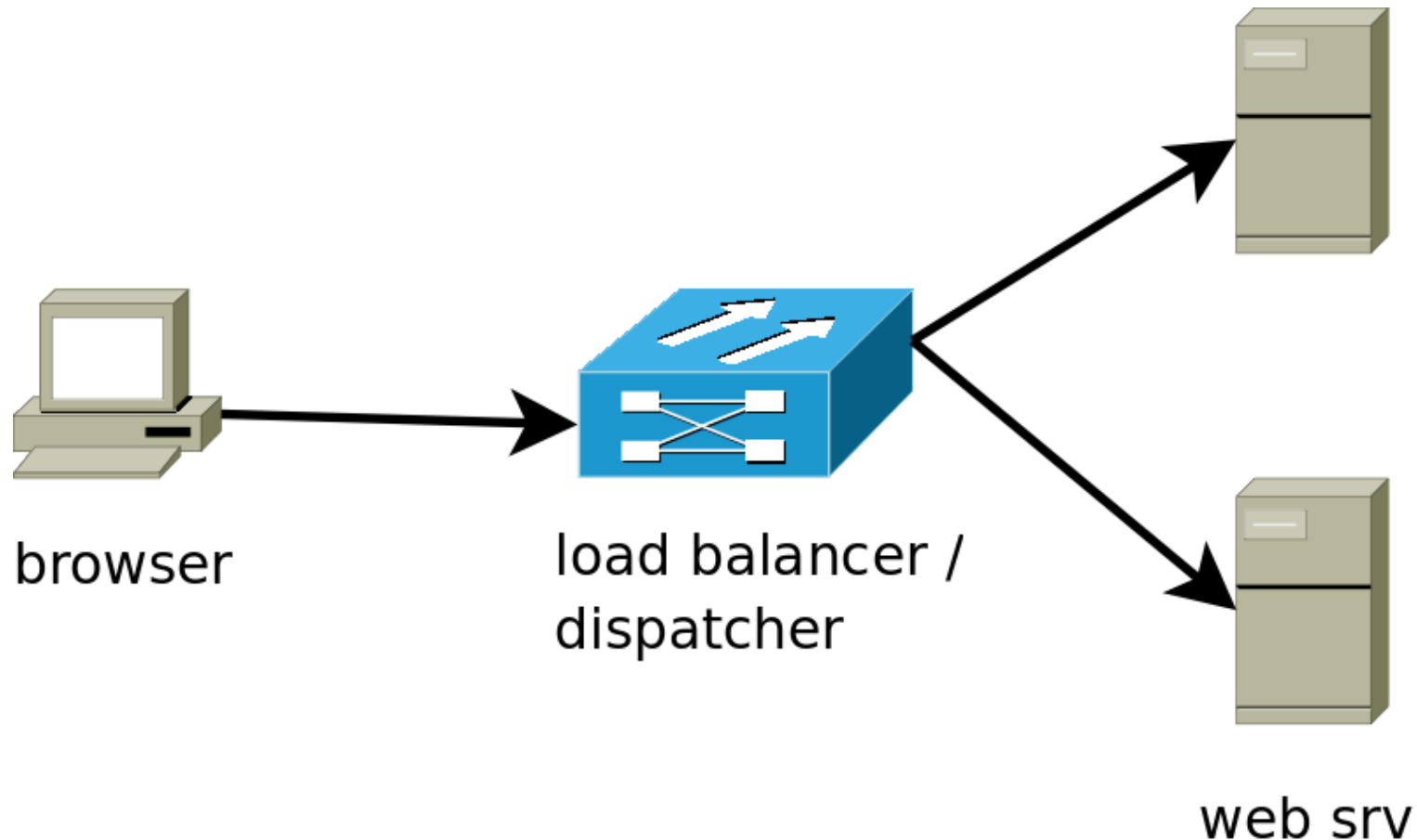
Klasteri za web aplikacije

- svaki sloj se može nezavisno klasterovati



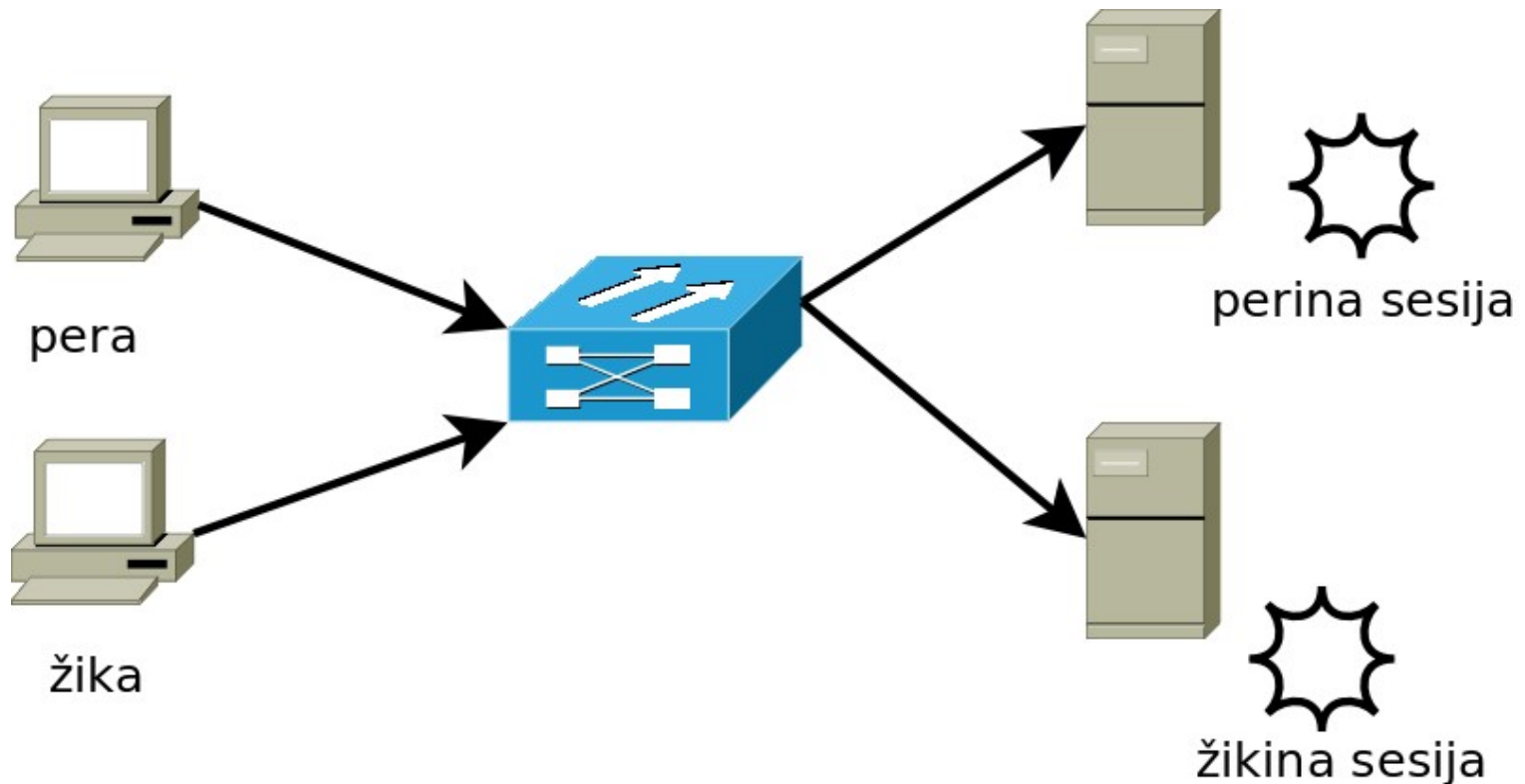
Klaster web servera

- serverski računari sa web serverom (npr. Tomcat)
- komunikacija između čvorova u klasteru
 - zavisi od načina korišćenja HTTP sesije



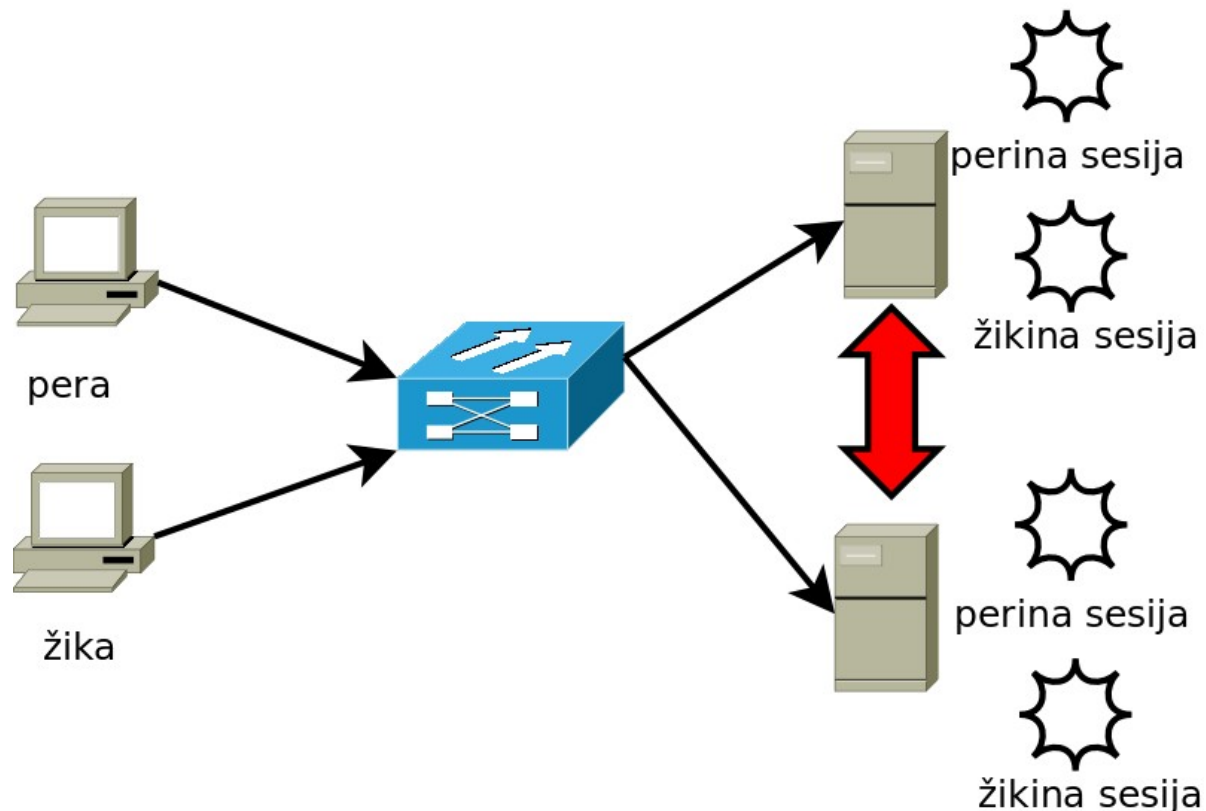
Klaster web servera

- replikacija HTTP sesije
 - varijanta 1: nema replikacije
 - load balancer: “sticky sessions” režim rada
 - zahtev jednog klijenta se uvek upućuje na isti server u klasteru
 - jednostavno, ali nema failover



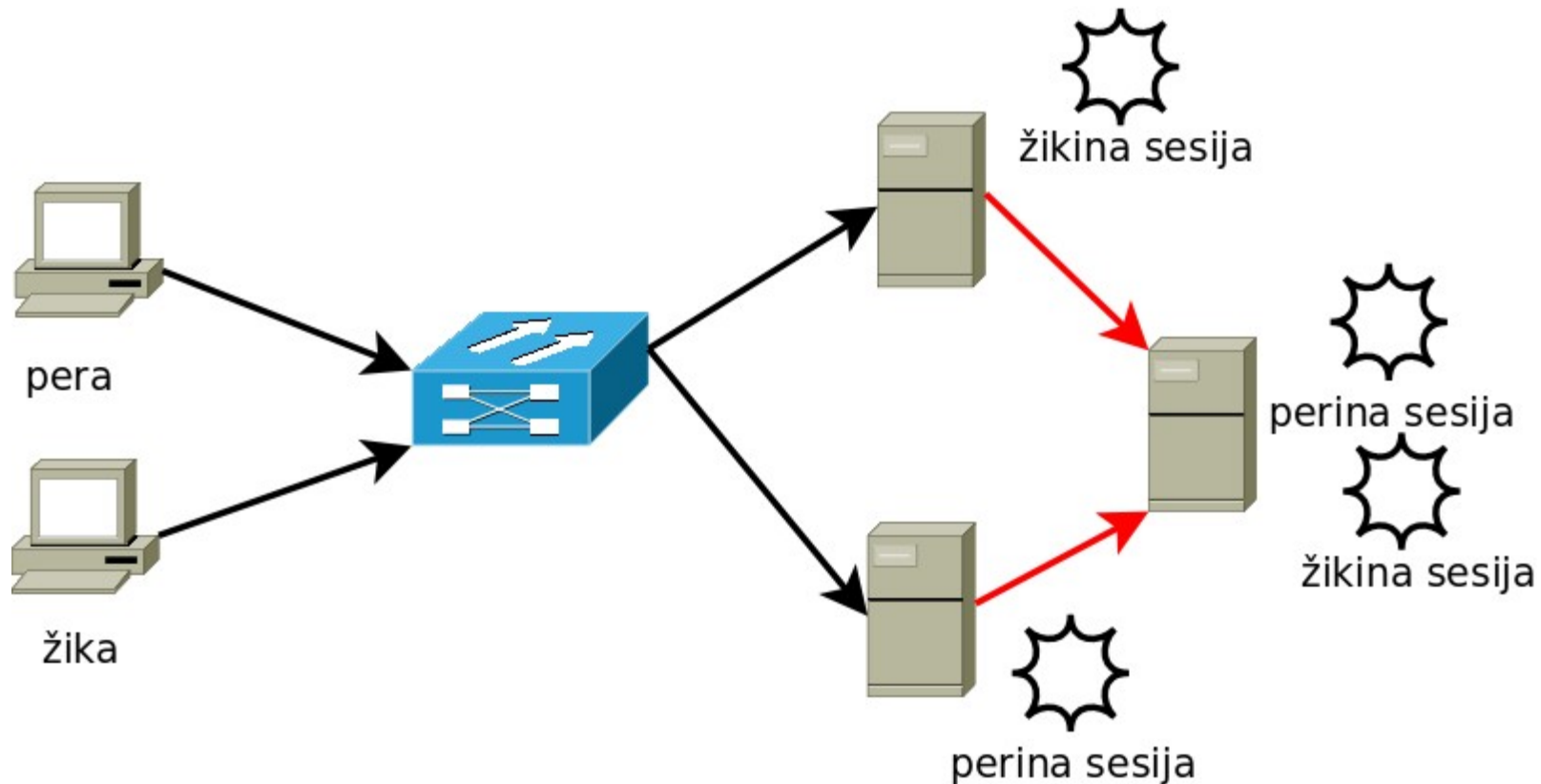
Klaster web servera

- replikacija HTTP sesije
 - varijanta 2: sve sesije na svim serverima (Tomcat)
 - load balancer: “request counting” ili “weighted traffic counting” režim
 - ima balancing, ima failover
 - replikacija sesija: veliki saobraćaj, nije za velike klastere ili velike sesije



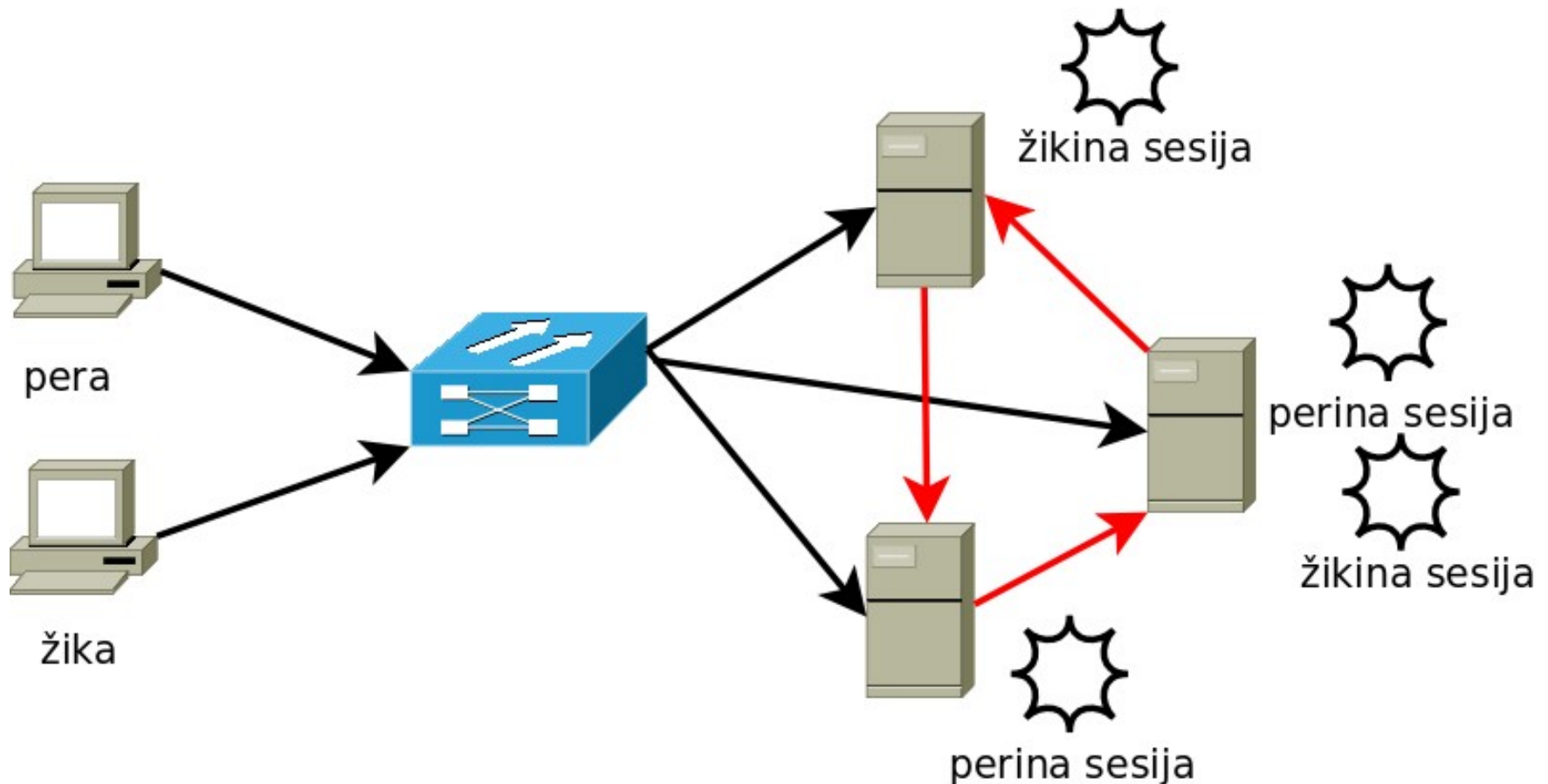
Klaster web servera

- replikacija HTTP sesije
 - varijanta 3: sesija se replicira samo na poseban čvor (Terracotta, IBM)
 - sesija je “slabo vezana” za čvor
 - load balancer radi sticky sessions dok je sve u redu
 - SPoF?



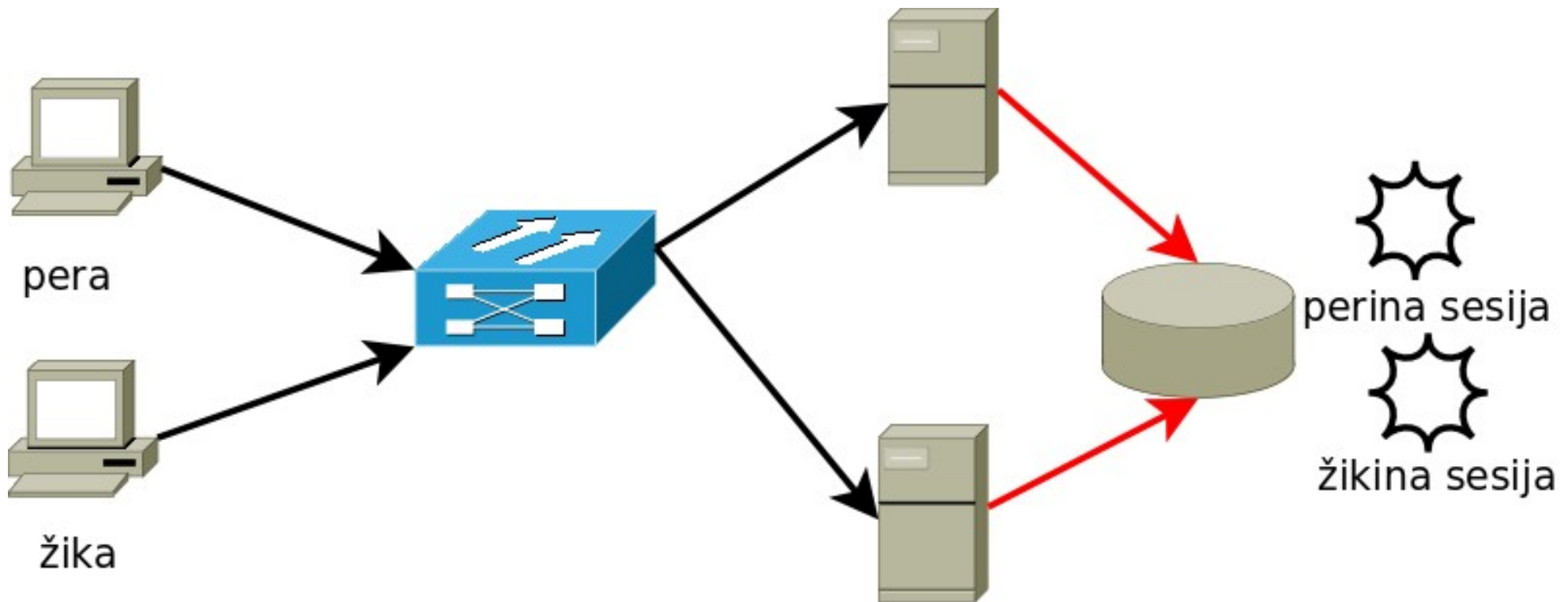
Klaster web servera

- replikacija HTTP sesije
 - varijanta 4: sesija se replicira na još jedan server (JBoss, Weblogic)
 - svaka sesija je na dva servera (primarni i backup)
 - dodavanje servera u klaster ne opterećuje saobraćaj



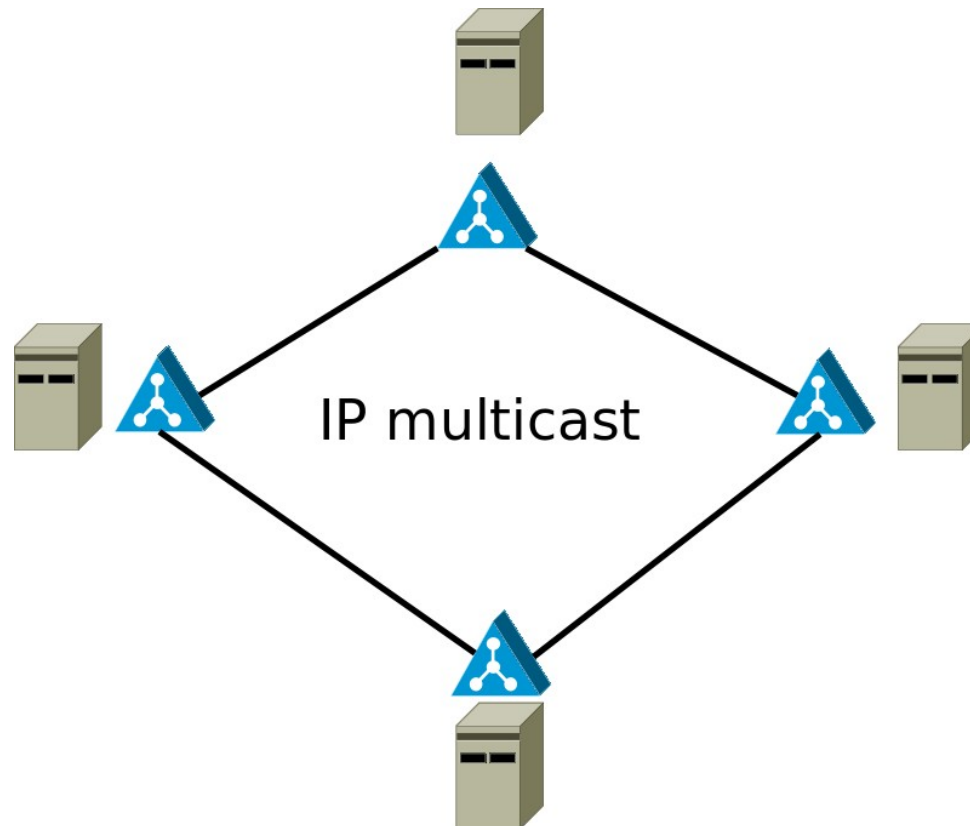
Klaster web servera

- replikacija HTTP sesije
 - varijanta 5: sesija se snima u bazu (Sun)
 - web serveri su praktično stateless
 - potencijalno veliki saobraćaj prema bazi podataka



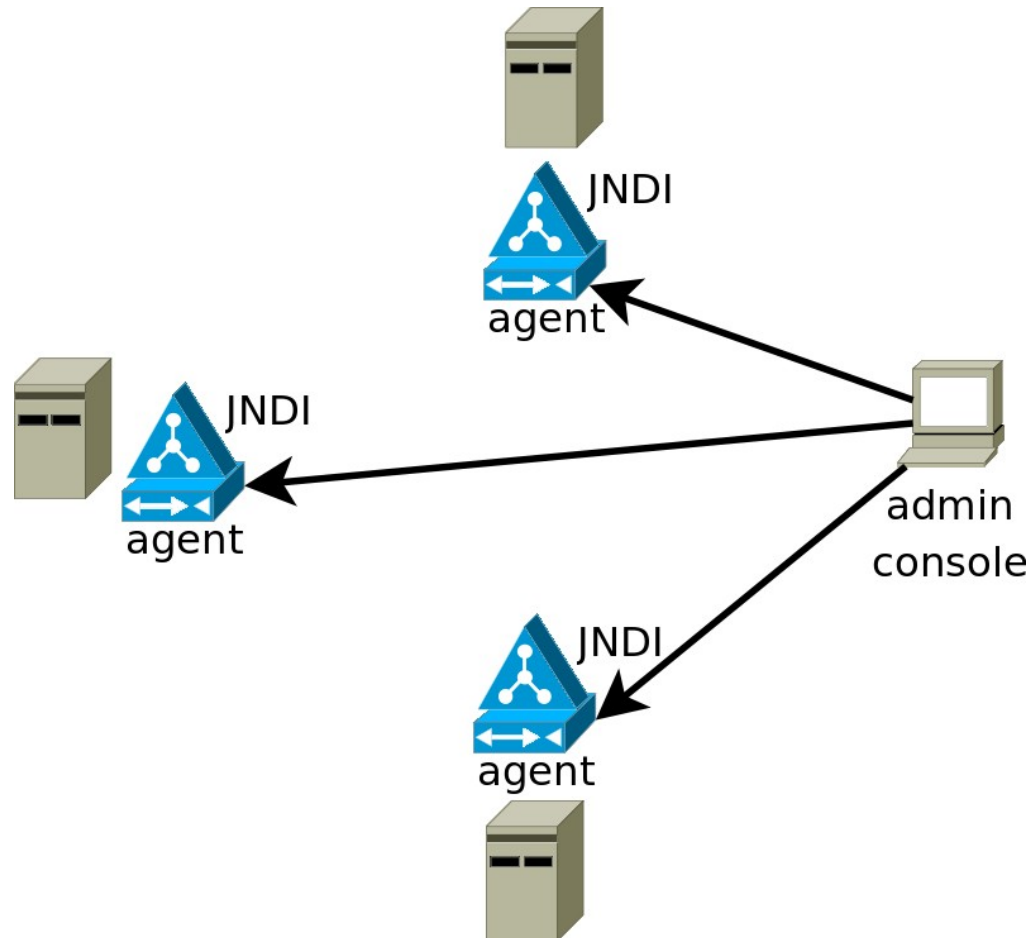
Klaster app servera

- JNDI stablo
 - varijanta 1: deljeno globalno JNDI stablo (JBoss, Weblogic)
 - stablo se replicira na sve čvorove pomoću IP multicasta
 - replikacija uključuje i objekte vezane za ime u stablu!



Klaster app servera

- JNDI stablo
 - varijanta 2: nezavisna JNDI stabla + agenti (Sun, IBM)
 - agenti zaduženi za automatizovano pronalaženje objekata u drugim stablima
 - ne repliciraju se objekti registrovani u stablu



Klaster app servera

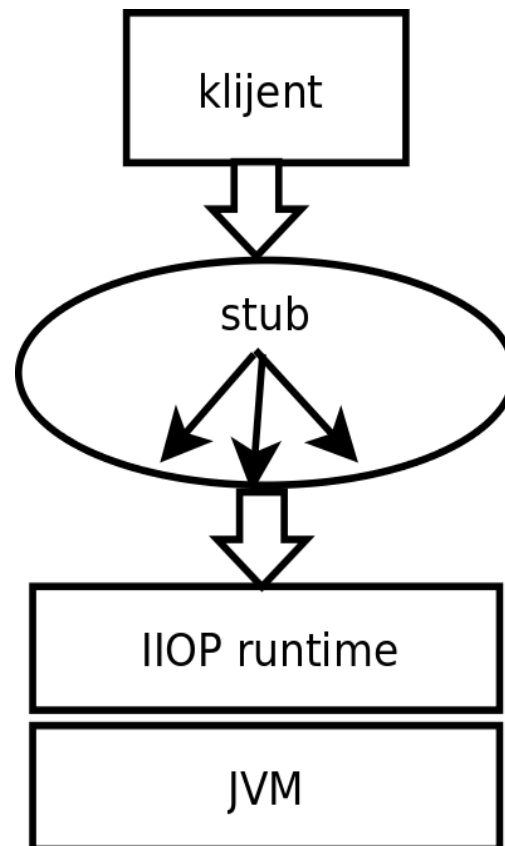
- JNDI stablo
 - varijanta 3: centralizovano JNDI stablo
 - poseban čvor
 - složena instalacija i administracija

Klaster app servera

- JNDI stablo
 - inicijalni pristup stablu - koja je adresa JNDI servera?
 - load balancer između klijenta i grupe JNDI servera
 - navođenje više comma-separated adresa prilikom inicijalizacije (Sun, JBoss)
 - `java.naming.provider.url=server1,server2,server3`
 - automatsko pronalaženje JNDI servera pomoću multicasta (JBoss)
 - `java.naming.provider.url=`

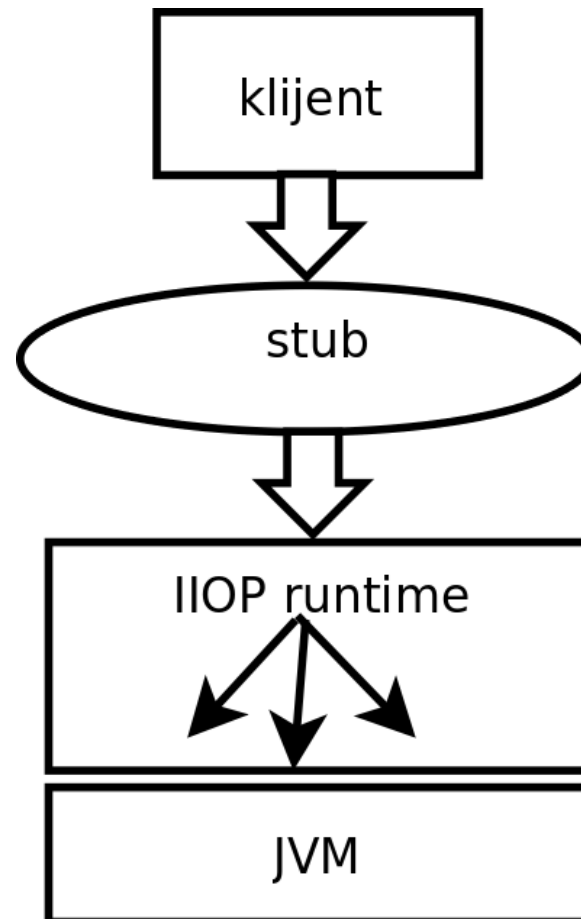
Klaster app servera

- pristup EJB komponentama
 - varijanta 1: smart stub (JBoss, Weblogic)
 - logika za pristup klasteru je ugrađena u stub



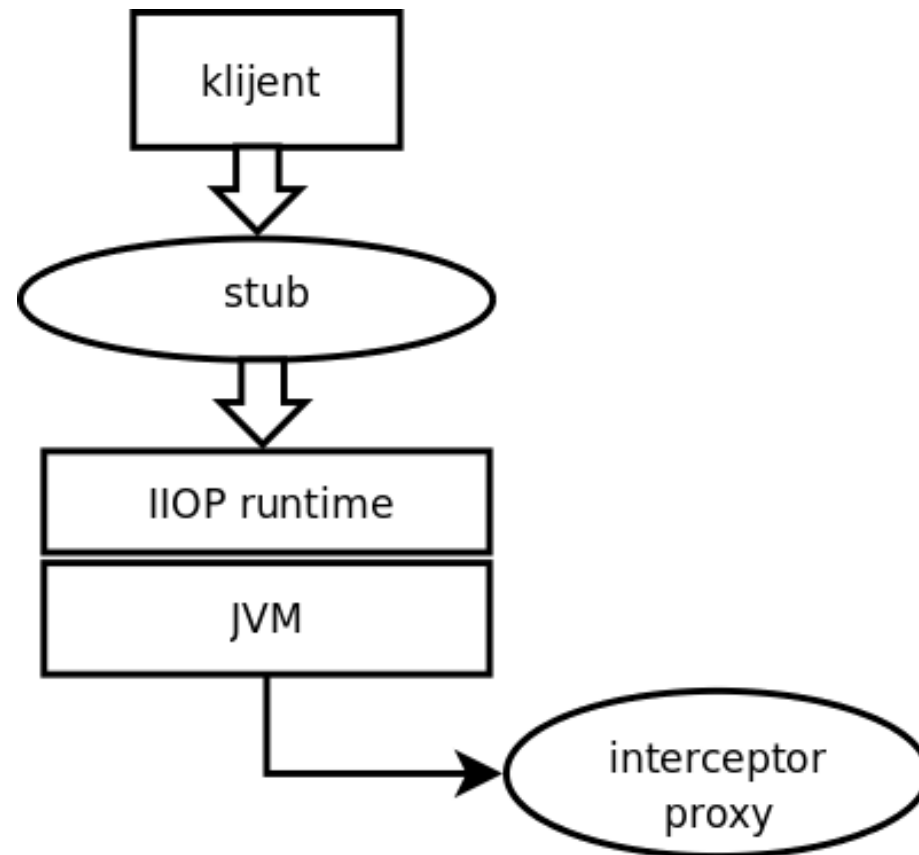
Klaster app servera

- pristup EJB komponentama
 - varijanta 2: smart IIOP runtime (Sun)
 - logika za pristup klasteru je ugrađena u IIOP runtime



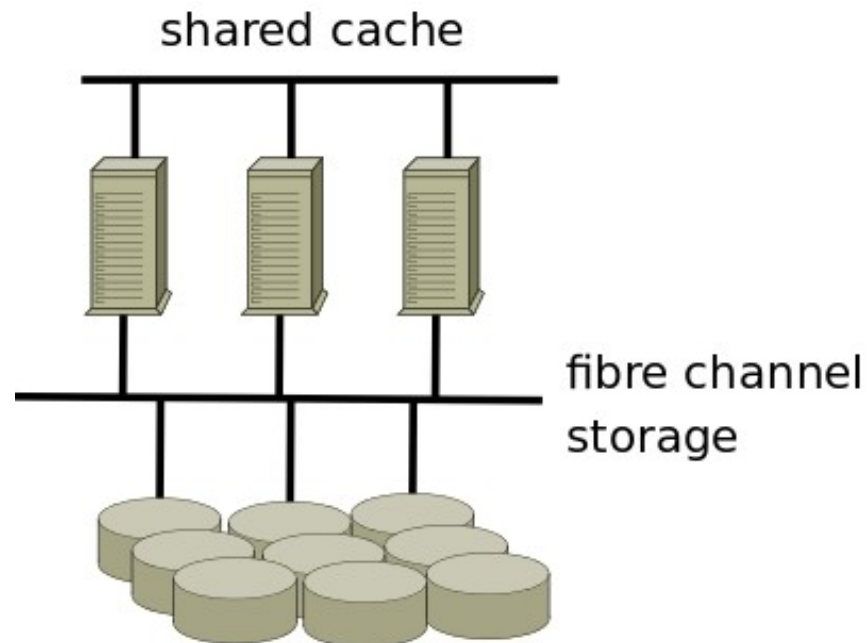
Klaster app servera

- pristup EJB komponentama
 - varijanta 3: interceptor proxy (IBM)
 - logika za pristup klasteru je ugrađena u poseban “presretački” objekat na serveru



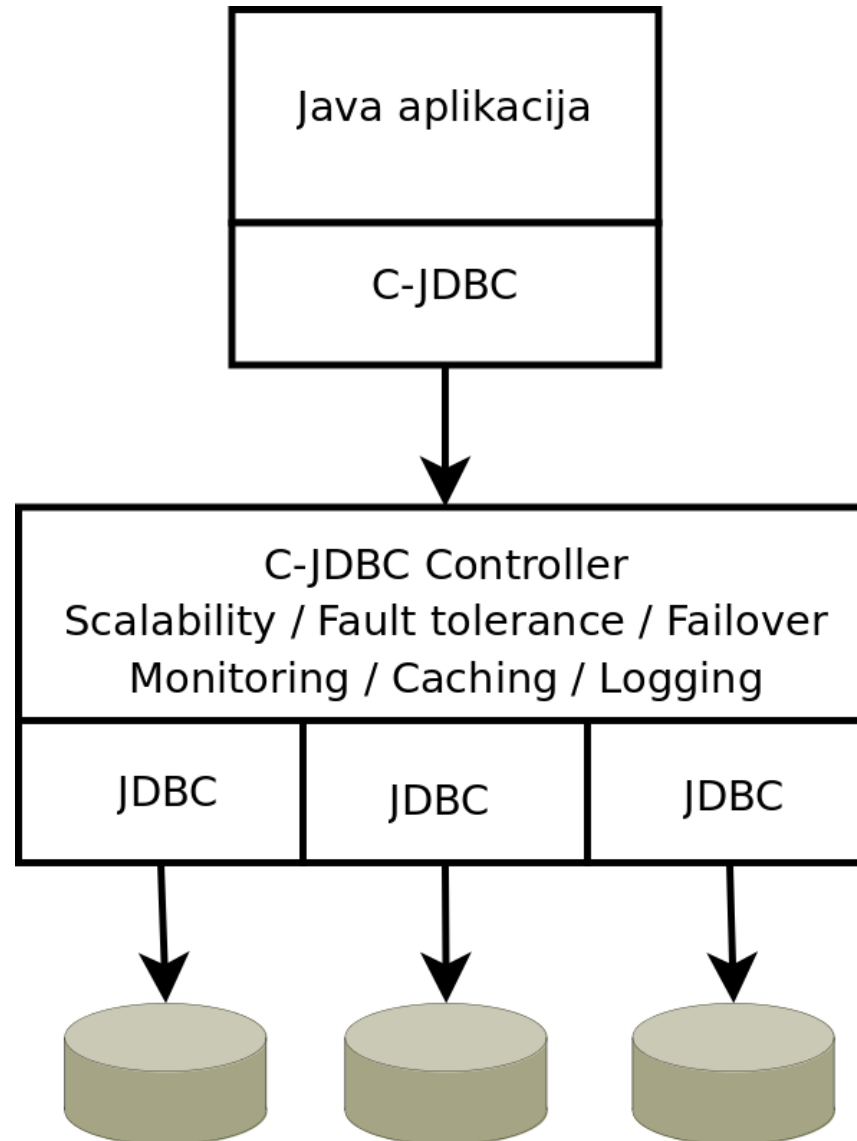
Klaster baze podataka

- najviše zavisi od konkretnog SUBP
 - Oracle u starijim verzijama: shared-nothing arhitektura
 - master/slave režim
 - promene se šalju sa master servera na 1 ili više slave servera
 - multiple master režim
 - više ravnopravnih master servera repliciraju promene u bazi
 - Oracle u novijim verzijama: shared-cache



Klaster baze podataka

- C-JDBC: klaster pomoću JDBC drajvera
 - cross-database: može povezati različite SUBP



Klasteri i pouzdanost

- da li je failover potpun i garantovan?
 - failover radi kada se otkaz desi između poziva EJB metoda
 - ako se otkaz desi u toku izvršavanja metode, nema oporavka!
 - ako je metoda idempotentna, server može pokušati da je ponovo pozove