WEB APLIKACIJE I KLASTERI

POJAM KLASTERA

ŠTA SU KLASTERI?

- Grupa međusobno povezanih računara koji funkcionišu tako da se mogu posmatrati kao jedan sistem koji pruža neki servis

UPOTREBA KLASTERA

- Sredstvo za unapređenje performansi
- Sredstvo za unapređenje pouzdanosti
- Jeftinije rešenje u odnosu na jedan računar ekvivalentnih mogućnosti

POJAM KLASTERA

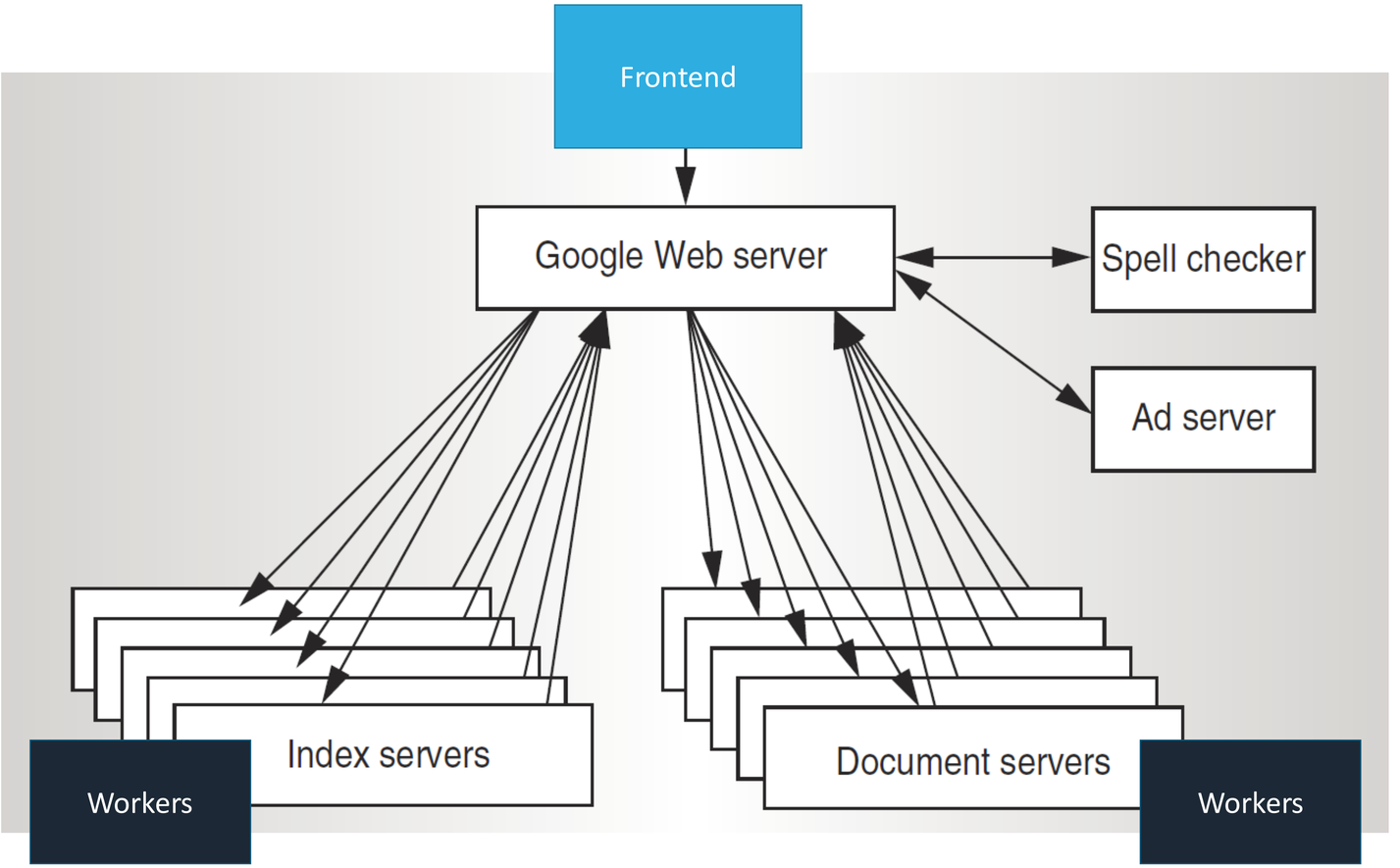
POVEZIVANJE ČVOROVA U KLASTERU

- Najčešće u lokalnoj mreži
- Može i distribuirano
- Dodavanjem novih čvorova u klaster povećavaju se i dostupnost i skalabilnost

GOOGLE KLASTER

WEB SEARCH FOR A PLANET: THE GOOGLE CLUSTER ARCHITECTURE

- Luiz Andre Barroso, Jeffrey Dean, Urs Holzle (2003)



SLA

SERVICE LEVEL AGREEMENT (SLA)

- Termin koji koriste pružaoci usluga (*service providers*)
- Dogovor između pružaoca servisa i klijenta koji formalno definiše nivo dostupnosti servisa (*uptime*)
- Obično se izražava u procentima do 100% (što više devetki to bolje)
- Amazon[1], Google[2] i Microsoft[3] definisali su u svojim SLA 99,9% i više dostupnost servisa

1 Amazon SLA example https://aws.amazon.com/compute/sla/

2 Google SLA example https://cloud.google.com/compute/sla

3 Microsoft Azure SLA example https://azure.microsoft.com/en-us/support/legal/sla/summary/

SLA

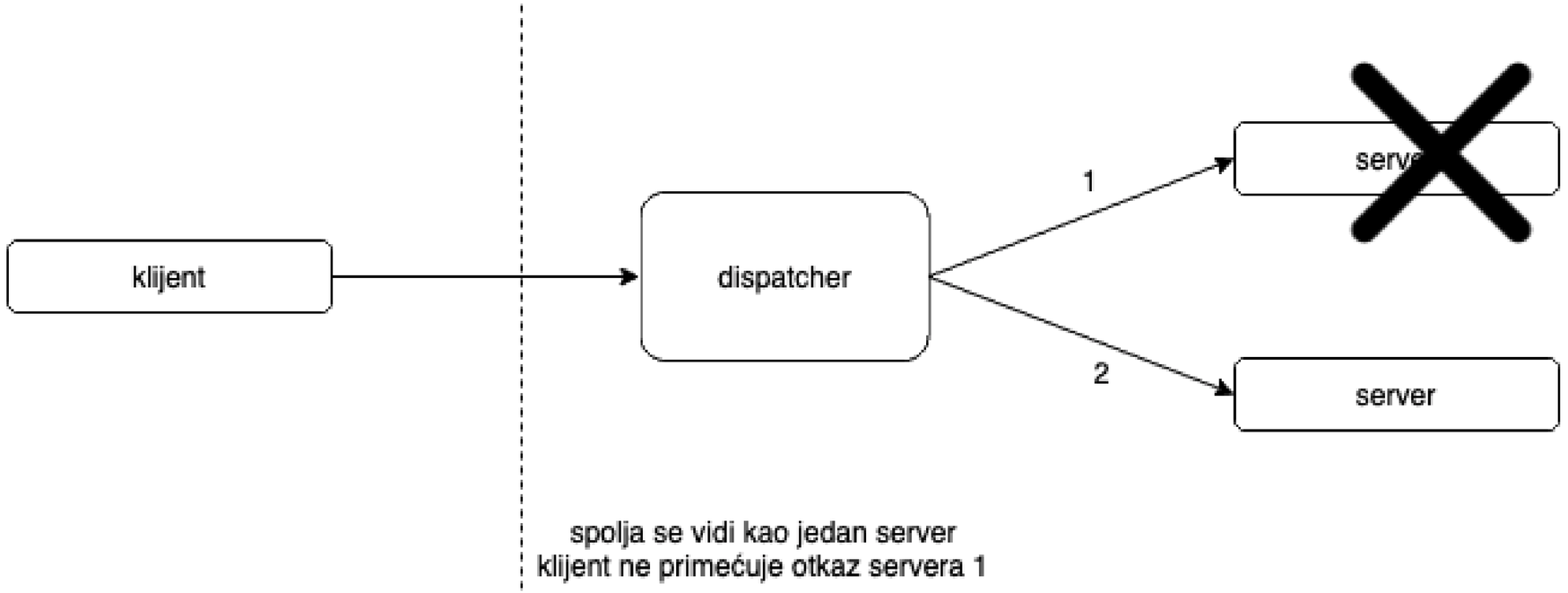
CIFRE OD ZNAČAJA ZA DOSTUPNOST SERVISA

Dostupnost %	Nedostupnost po danu	Nedostupnost po godini
99%	14,4 minuta	3,65 dana
99,9%	1,44 minuta	8,77 sati
99,99%	8,64 sekunde	52,6 minuta
99,999%	864 milisekunde	5,26 minuta
99,9999%	86,4 milisekunde	31,56 sekundi

VRSTE KLASTERA

KLASTER VISOKE DOSTUPNOSTI (HIGH-AVAILABILITY, FAILOVER)

- Redudantni hardver, veća pouzdanost
- Minimum dva računara
- Eliminiše SPoF (single point of failure)



KLASTER VISOKE DOSTUPNOSTI

ACTIVNI-PASIVNI KLASTER (ACTIVE-PASSIVE)

- U početku samo jedan čvor opslužuje korisnike i nastavlja da radi sam sve dok iz nekog razloga ne otkaže
- U tom trenutku, i nove i postojeće sesije se prenose na rezervni ili neaktivan čvor
- Preporuka je da se doda još jedna redundantna komponenta za svaki tip resursa (N + 1 redundantnost) da bi se osiguralo da postoji dovoljno resursa za postojeće opterećenje, dok se istovremeno pokriva potencijalni otkaz

KLASTER VISOKE DOSTUPNOSTI

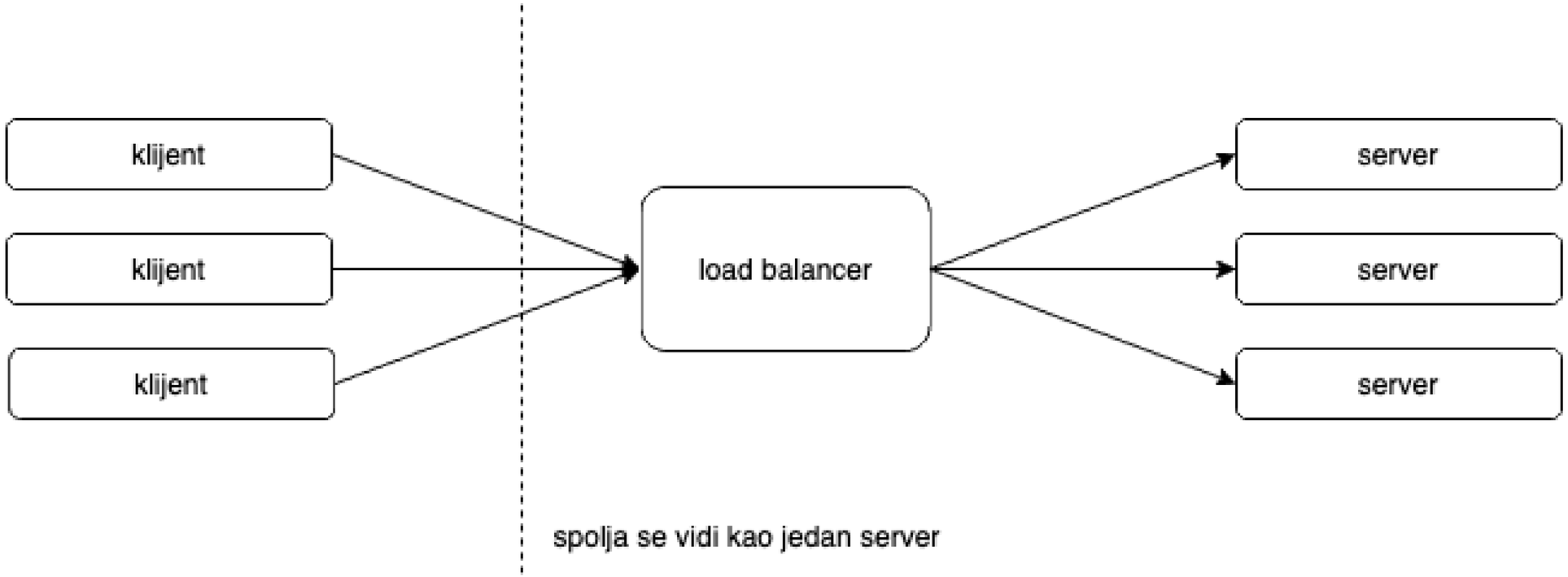
AKTIVNI-AKTIVNI KLASTER (ACTIVE-ACTIVE)

- Postoje dva ili više čvorova sa istom konfiguracijom, kojima klijenti direktno pristupaju
- Ako jedan čvor otkaže, klijenti se automatski preusmeravaju na drugi čvor i počinju da rade sa njim, sve dok ima dovoljno resursa (jer jedan čvor sada upravlja opterećenjem za dva čvora)
- Nakon oporavka ili zamene prvog čvora, klijenti se ponovo dele između dva originalna čvora
- Glavna prednost active-active klastera je ta što se može postići ravnoteža između čvorova u mreži

VRSTE KLASTERA

RASPOREĐIVANJE OPTEREĆENJA (LOAD BALANCING)

- Raspodela opterećenja na više čvorova
- Različiti algoritmi raspodele



VRSTE KLASTERA

LOAD BALANCING – TEHNIKE RASPODELE

- (Weighted) Round Robin
- Source IP Hash
- (Weighted) Least Connection
- Resource Based
- Weighted Response Time
- ...

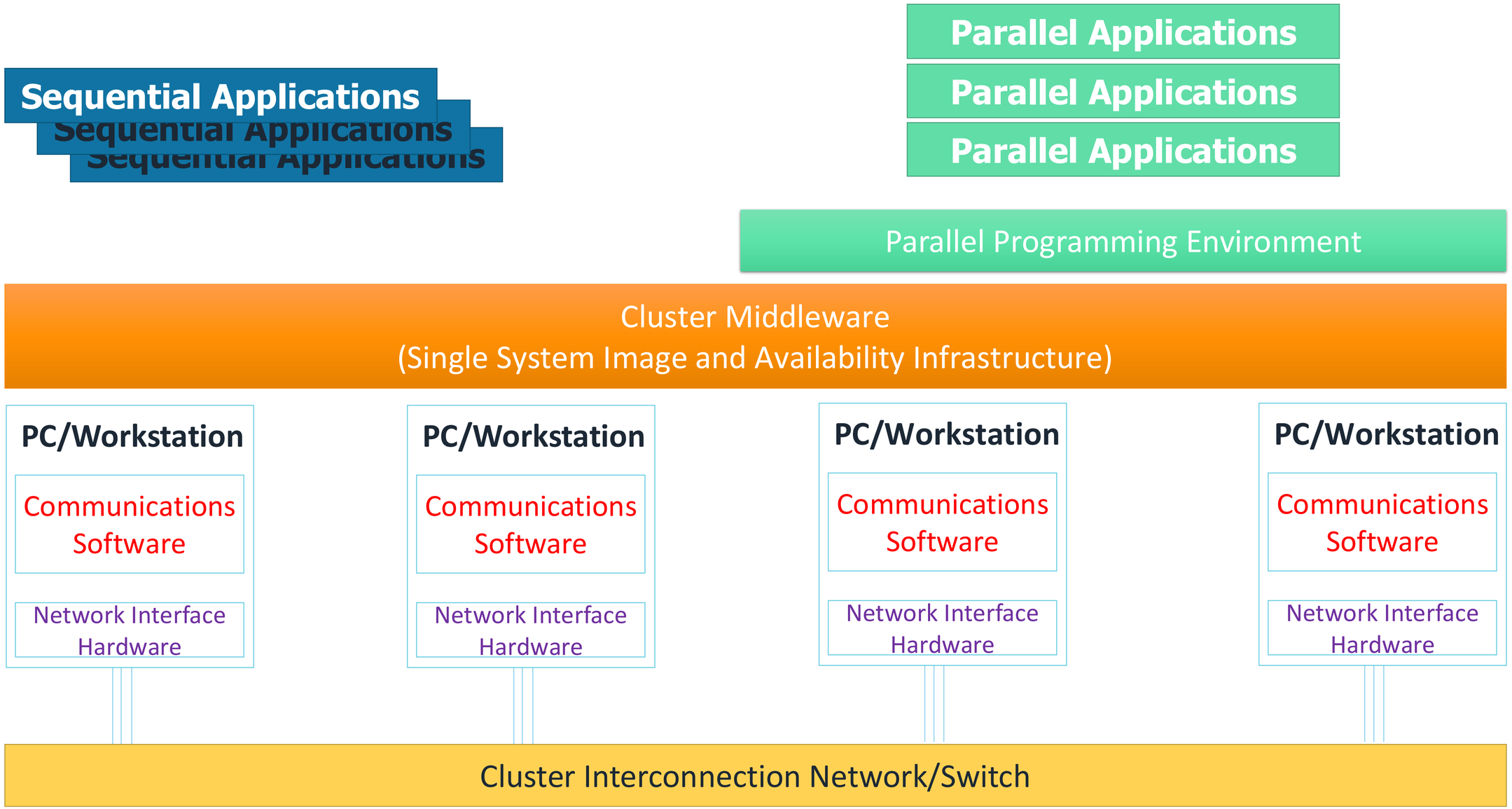
VRSTE KLASTERA

KLASTERI VISOKIH PERFORMANSI (HIGH PERFORMANCE COMPUTING)

- Namenjeni za specifične poslove masovne paralelne obrade podataka
- Različiti načini za sprezanje čvorova:
    - Tightly coupled
    - Loosely coupled
    - Grid computing[1]

1 Grid computing https://en.wikipedia.org/wiki/Grid_computing

KLASTERI VISOKIH PERFORMANSI



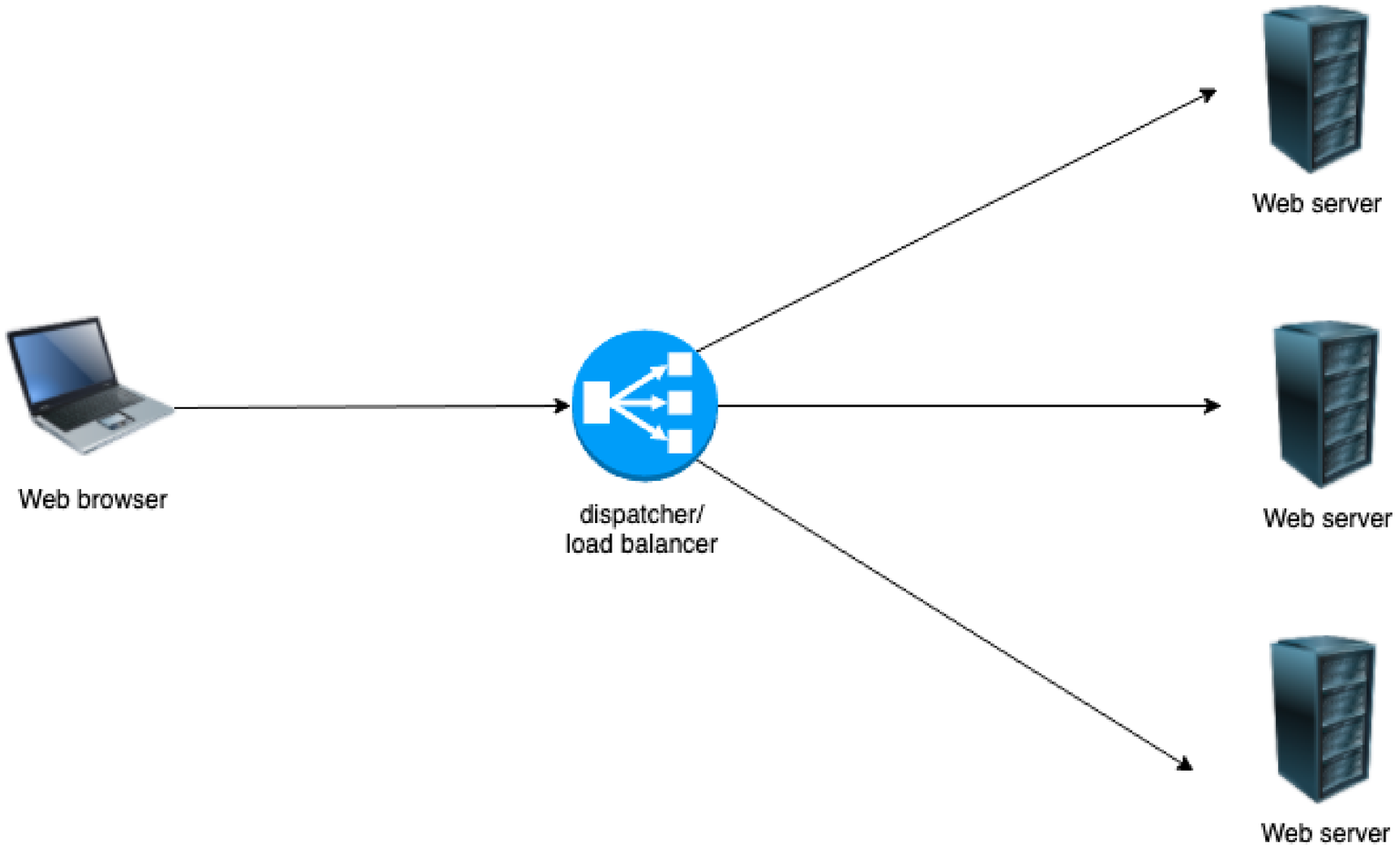
KLASTERI VISOKIH PERFORMANSI

BEOWULF KLASTER

- Dizajn koji koristi paralelnu obradu na više računara za kreiranje jeftinih i moćnih super-računara
- Beowulf klaster u praksi je obično kolekcija generičkih računara, bilo prekonfigurisanih ili sačinjenih iz delova koji se kupuju i sklapaju nezavisno
- Klaster ima dva tipa računara, glavni (master) računar i čvorove
- Kada se kompleksni problem ili veliki skup podataka prosledi Beowulf klasteru, glavni računar prvo pokreće program koji razbija problem na male diskretne delove
- Nakon toga šalje deo svakom čvoru na obradu
- Kako čvorovi završavaju svoje zadatke, glavni računar im neprestano šalje nove delove dok se ceo problem ne reši

KLASTER WEB SERVERA

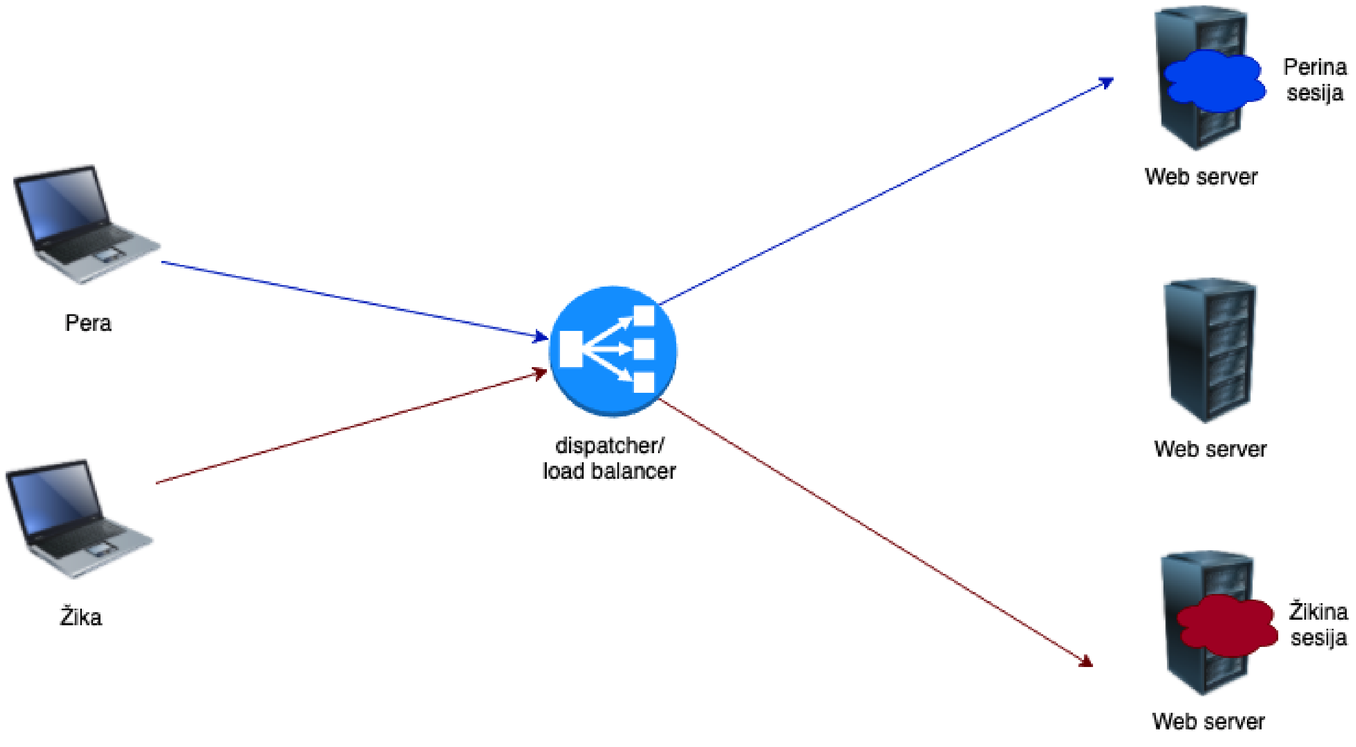
- **SERVERSKI RAČUNARI NA KOJIMA JE WEB SERVER**



KLASTER WEB SERVERA I REPLIKACIJA HTTP SESIJE

NEMA REPLIKACIJE

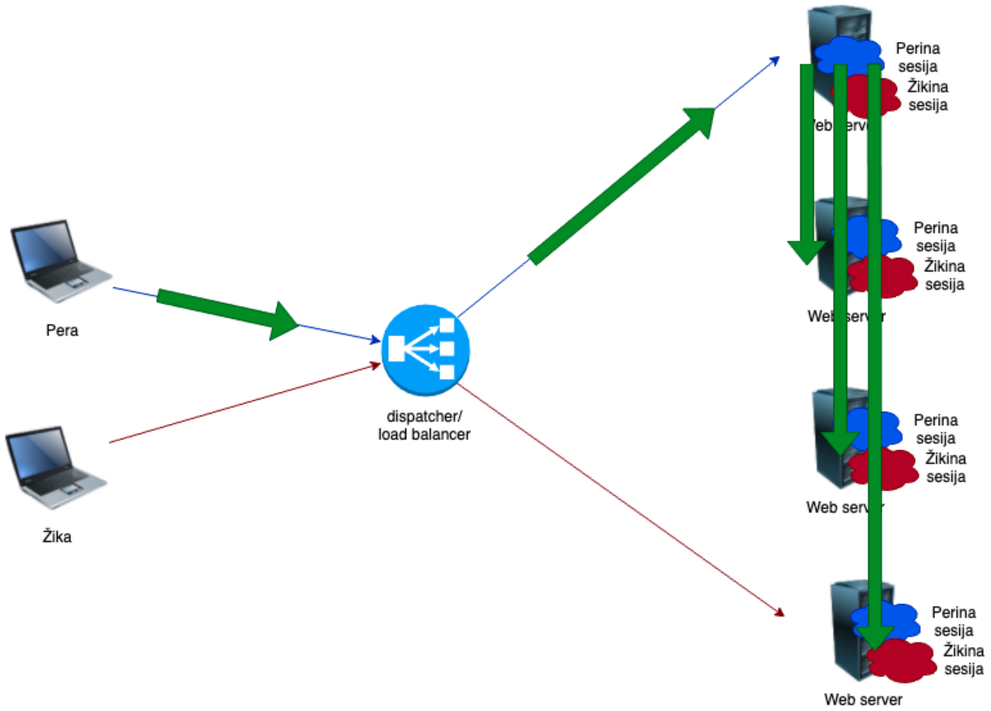
- *Sticky sessions* režim rada
- Zahtev jednog klijenta uvek se upućuje na isti server u klasteru
- Jednostavno, ali nema *failover*



KLASTER WEB SERVERA I REPLIKACIJA HTTP SESIJE

SVE SESIJE NA SVIM SERVERIMA (TOMCAT[1])

- Ima balansiranje, ima *failover*
- Replikacija sesija - veliki saobraćaj, nije za velike klastere ili velike sesije

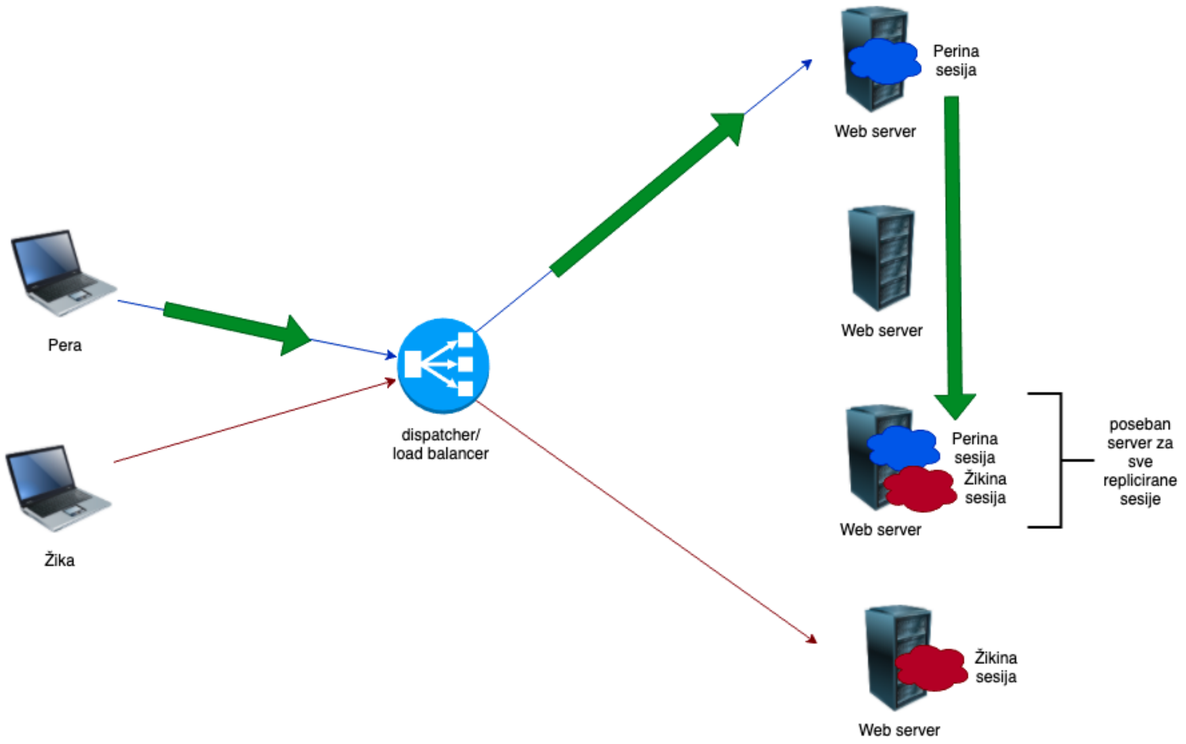


1 Tomcat session https://tomcat.apache.org/tomcat-8.0-doc/cluster-howto.html

KLASTER WEB SERVERA I REPLIKACIJA HTTP SESIJE

SESIJA SE REPLICIRA SAMO NA POSEBAN ČVOR (TERRACOTA, IBM)

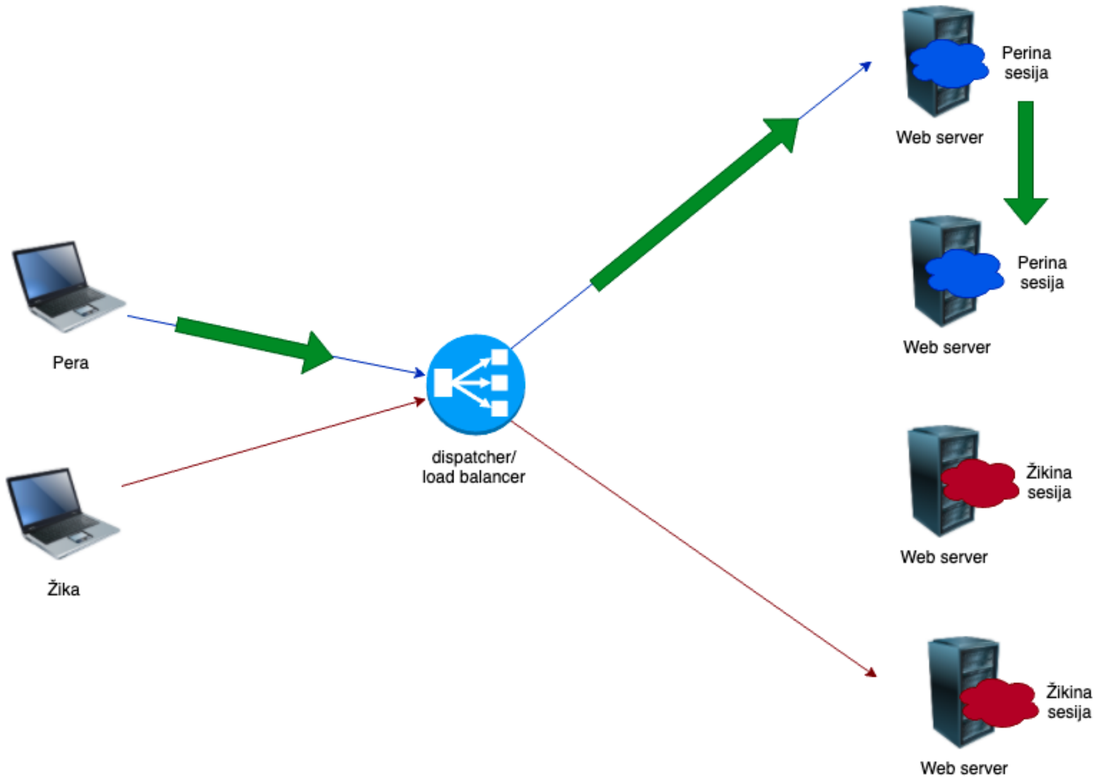
- Sesija je slabo vezana za čvor
- Load balancer radi *sticky sessions* dok je sve u redu
- SPoF?



KLASTER WEB SERVERA I REPLIKACIJA HTTP SESIJE

SESIJA SE REPLICIRA NA JOŠ JEDAN SERVER (JBOSS, WEBLOGIC)

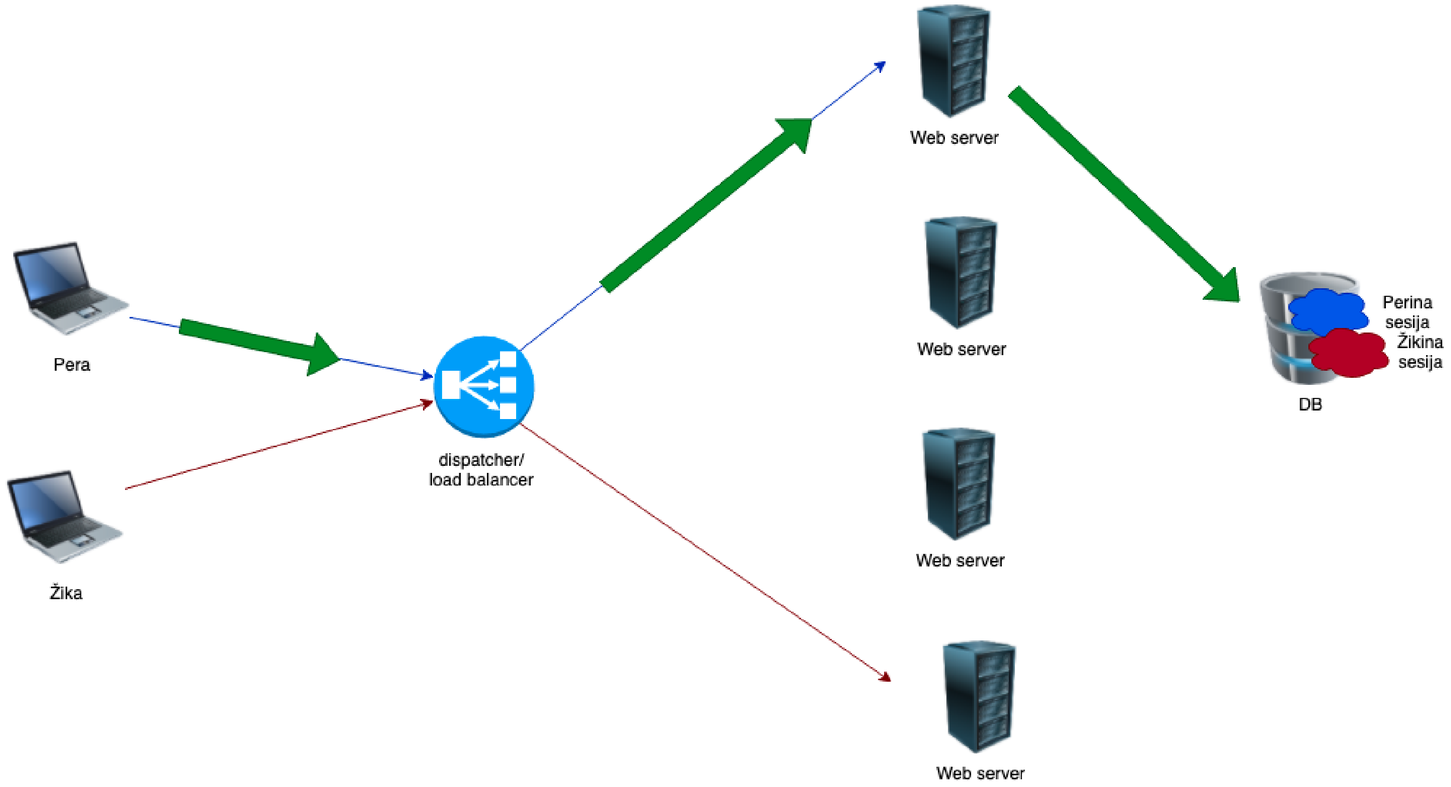
- Svaka sesija je na dva servera (primarni i backup)
- Dodavanje novih servera ne povećava saobraćaj



KLASTER WEB SERVERA I REPLIKACIJA HTTP SESIJE

SESIJA SE ČUVA U BAZI PODATAKA (SUN)

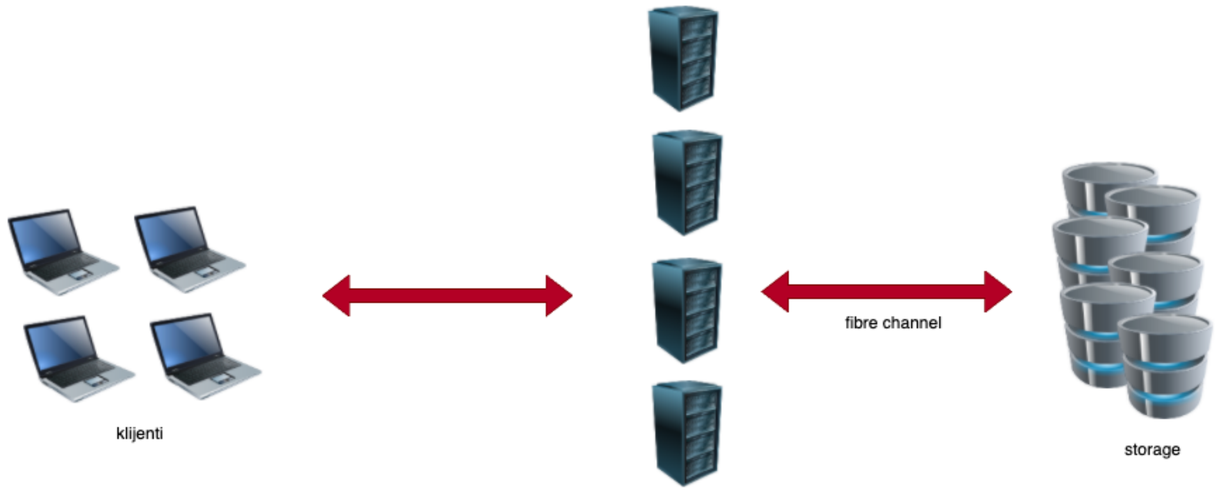
- Web serveri su *stateless*
- Potencijalno veliki saobraćaj prema bazi podataka



KLASTER BAZE PODATAKA

SPECIFIČAN ZA KONKRETAN SUBP

- *Shared-nothing* arhitektura
  - Podaci na disku (ili u memoriji) se ne dele između nodova u klasteru
  - Svaki zahtev obrađuje jedan čvor (CPU/memorija/disk)
- *Shared-everything (shared-disk)* arhitektura
  - Podaci na disku (ili u memoriji) se dele između čvorova u klasteru

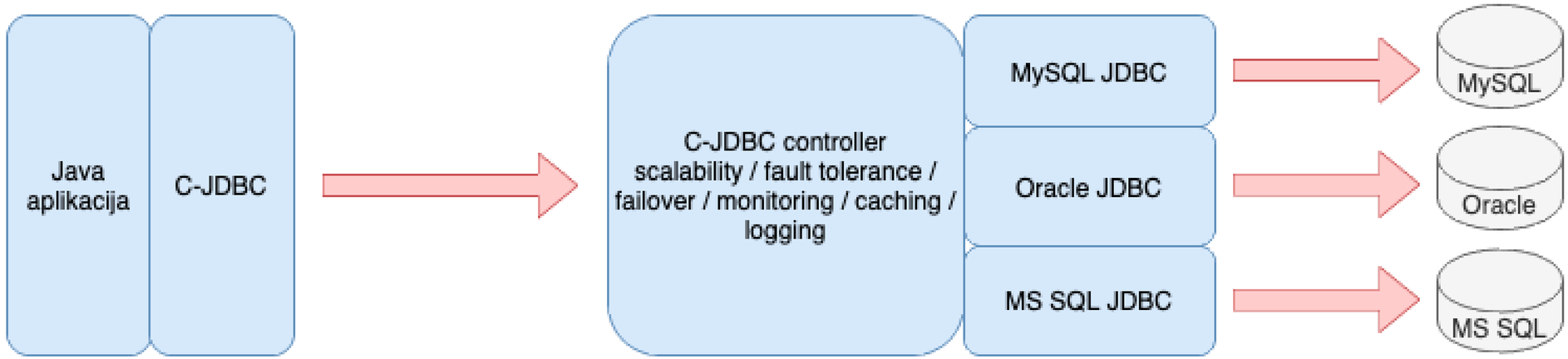


Spisak baza koje koriste shared-nothing i shared-everything arhitekture
https://en.wikipedia.org/wiki/Oracle_RAC

KLASTER BAZE PODATAKA

KLASTER POMOĆU JDBC DRAJVERA – C-JDBC

- *Cross-database* – može povezivati različite SUBP u jedan klaster



REFERENCE

- SLAJDOVI PO UZORU NA https://github.com/mbranko/isa19/blob/master/09-arch/clustering.pdf
- BARROSO ET AL. WEB SEARCH FOR A PLANET: THE GOOGLE CLUSTER ARCHITECTURE https://static.googleusercontent.com/media/research.google.com/en//archive/googlecluster-ieee.pdf
- BECKER ET AL. BEOWULF: A PARALLEL WORKSTATION FOR SCIENTIFIC COMPUTATION https://webhome.phy.duke.edu/~rgb/brahma/Resources/beowulf/papers/ICPP95/icpp95.html
- GOOGLE CLOUD. ARCHITECTURES FOR HIGH AVAILABILITY OF MYSQL CLUSTERS ON COMPUTE ENGINE https://cloud.google.com/architecture/architectures-high-availability-mysql-clusters-compute-engine
- C-JDBC USER'S GUIDE. https://c-jdbc.ow2.org/current/doc/userGuide/html/index.html
- STONEBRAKER M. THE CASE FOR SHARED NOTHING https://static.aminer.org/pdf/PDF/000/255/770/the_case_for_shared_nothing.pdf
- DEWITT D. ET AL. HOW TO BUILD A HIGH-PERFORMANCE DATA WAREHOUSE http://db.csail.mit.edu/madden/high_perf.pdf
- SCALEDB. SHARED-DISK VS. SHARED-NOTHING - COMPARING ARCHITECTURES FOR CLUSTERED DATABASES https://bit.ly/3BqL5Yk

KOJA SU VAŠA PITANJA?