

BALANSIRANJE PROMETA TOMCAT INSTANCI

Dubravko Miljković
HEP Zagreb



Uvod

- Ponekad je potrebno uslužiti veliki broj korisnika
- Nadmašuje mogućnosti pojedinačne Tomcat instance
 - 300 - 350 transakcija u sekundi (ovisno o transakciji)
 - 200 threads default
- Formiranje clustera Tomcat instanci
- Instalirane na jednom ili više fizičkih servera

Kapacitet i raspoloživost

- Kapacitet
 - Broj korisnika koji sustav može uslužiti
- Raspoloživost
 - Vjerojatnost da sustav ima sposobnost obavljanja definirane funkcije u pretpostavljenom trenutku



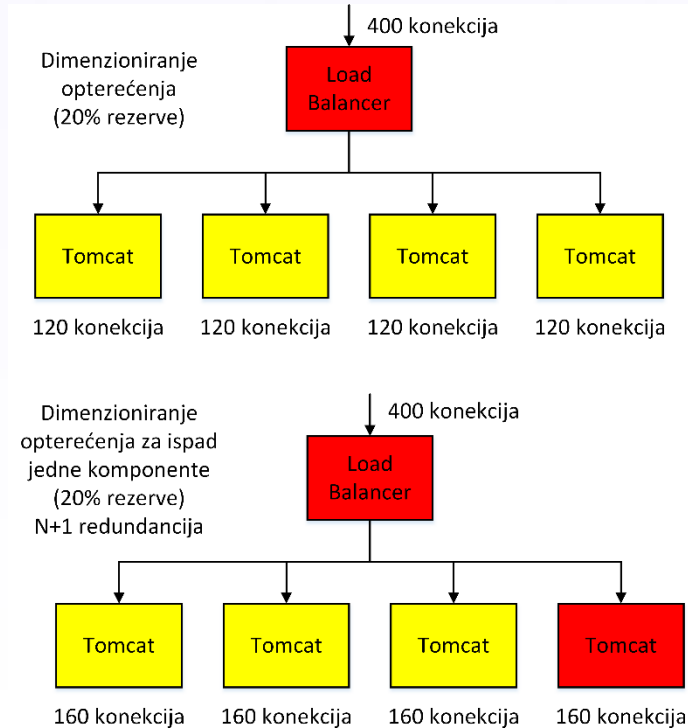
Kapacitet i raspoloživost

- Kapacitet se ostvaruje povezivanjem individualnih komponenti u klaster
 - Raspodjela opterećenja između više instanci
 - Instance predvidjeti za 20 % veće opterećenje od idealne raspodjele
 - Instance mogu biti na istom server ili na različitim serverima



Kapacitet i raspoloživost

Rezerve za greške
distribucije konekcija
od Load Balancer-a



Kapacitet i raspoloživost

- Na jedan server može se instalirati više Tomcat instanci
 - Svaka u svom direktoriju
 - Svaka sluša na svom port-u



Kapacitet i raspoloživost

- Visoka raspoloživost se ostvaruje redundantnim hardware-om
 - Uporaba više servera
 - Load balancing bez pojedinačne točke kvara
 - Maksimizira vrijeme u ispravnom stanju
 - Minimizira vrijeme u stanju kvara



Kapacitet instance i mrežni promet

- Korisnici toleriraju odzive do otprilike 0,5 s
- Kapacitet je ograničen instancom i mrežom
- Average Application Response Time (AART) ms
- Average Application Response Size (AARS) KBytes

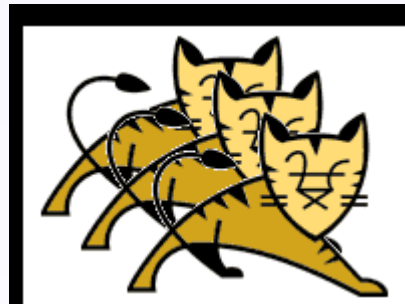
Formule za izračun (M. Turk):

$$\text{ConcurrentUsers} = (500 / \text{AART (ms)}) * N_{\text{CPU}} \quad (\text{max. } 200 \text{ per CPU})$$
$$\text{ConcurrentUsers} = \text{NetworkThroughput (KBytes/s)} / \text{AARS (KBytes)}$$

- brze mrežne kartice (1 Gbps) ili dodatne mrežne kartice



Balansiranje prometa

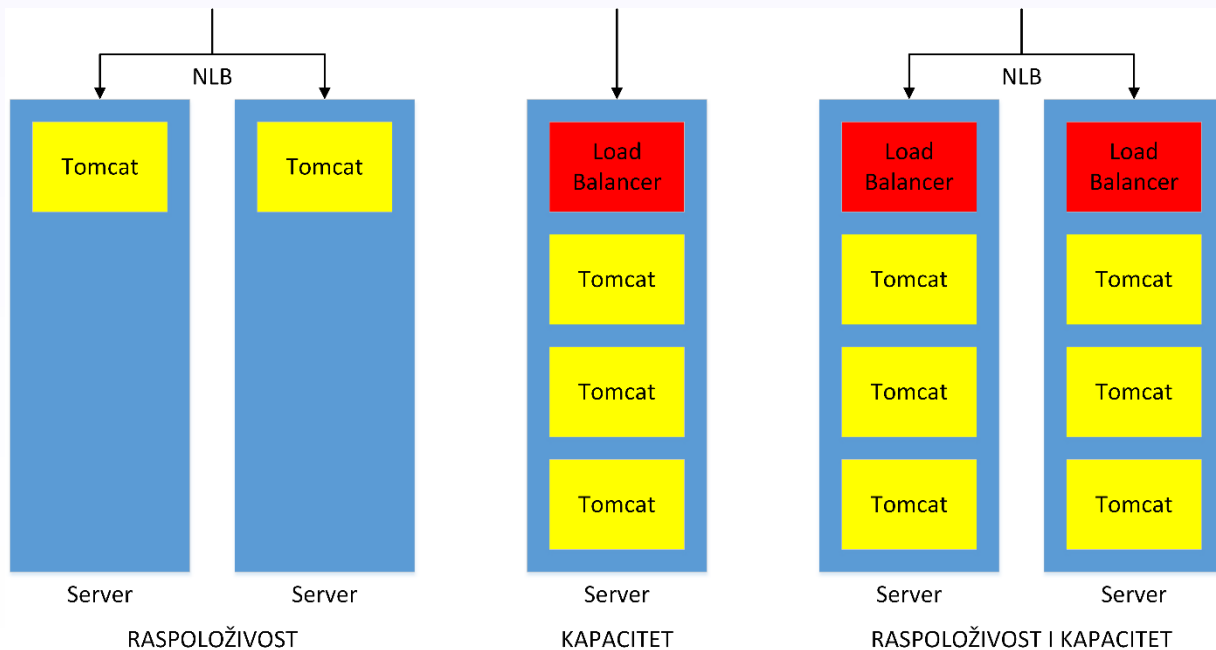


- Između
 - više servera
 - više instanci na istom serveru
 - više instanci na različitim serverima



Balansiranje prometa

Nekoliko pristupa, ovisno o cilju: kapacitetu i/ili raspoloživosti



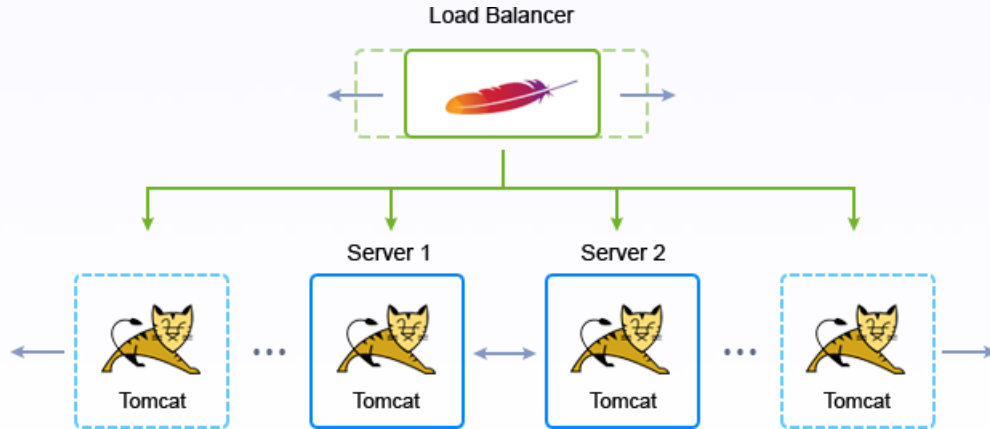
Balansiranje prometa

- Hardware
 - Load balancer kao zaseban uređaj
 - Za raspoloživost potrebna redundancija
 - active-standby ili active-active
- Software
 - Microsoft NLB
 - Apache load balancer
 - Nginx
 - ...



Balansiranje prometa

Load balancer: izdvojeni uređaj, NLB ili software (Apache, Nginx)



Balansiranje prometa



- Pristup sistemaša
 - NLB ili neki drugi Load Balancer
 - Fizički ili virtualni serveri
 - Tomcat instance na različitim IP adresama, ali na istom portu
- Pristup JAVA programera
 - Apache load balancer
 - mod_jk
 - mod_proxy
 - Nginx

(open source rješenja)

 - Tomcat instance na istoj IP adresi, ali na različitim portovima
- Kombinirani pristup
 - NLB između pojedinih servera
 - Apache load balancer unutar pojedinih servera

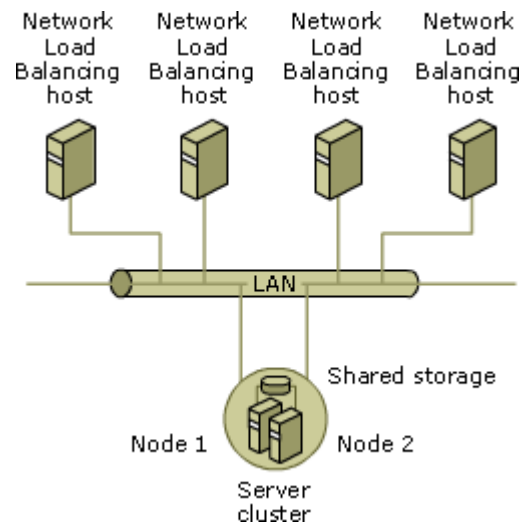
Balansiranje prometa - Microsoft NLB

- Pristup sistemaša ima smisla kad koristimo više fizičkih servera zbog redundancije
- Inače rasipa resurse
 - virtualni stroj za svaku Tomcat instancu, OS licence (ne uvijek, npr. Datacenterar licencing)
- Ne zahtijeva detaljna znanja o Tomcat instanci i Apache serveru



Balansiranje prometa - Microsoft NLB

- NLB – Network Load Balancing
 - Distribuirira promet uporabom TCP/IP protokola
 - Arhitektura sabirnice – bez pojedinačne točke kvara (Single Point of Failure)
 - Klaster fizičkih ili virtualnih servera
 - Visoka raspoloživost i skalabilnost
 - Centralizirano upravljanje klasterom

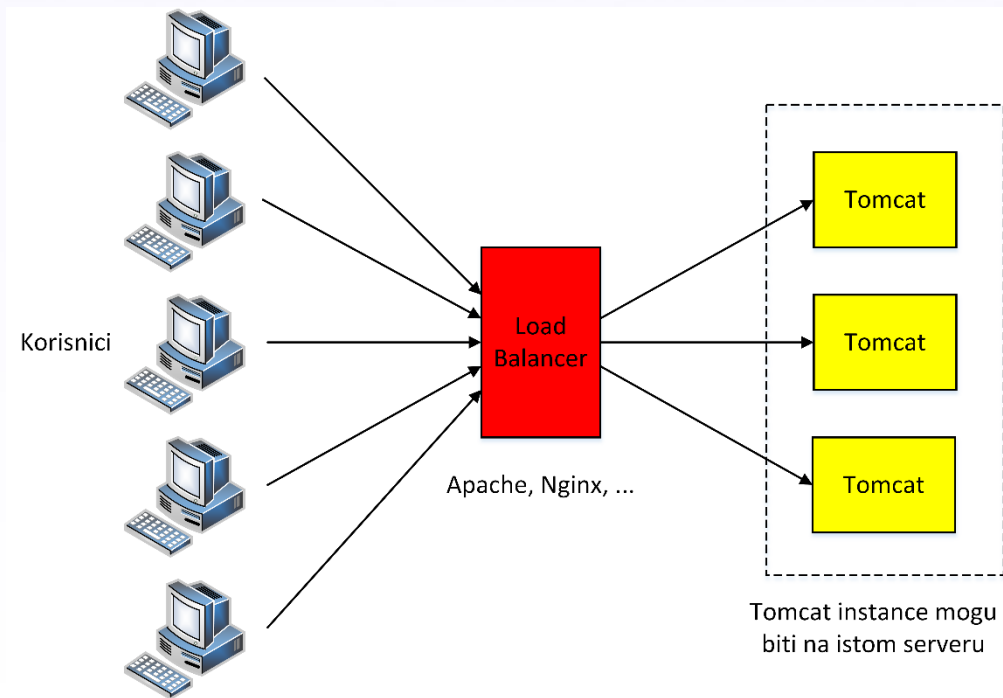


Balansiranje prometa - Microsoft NLB

- Provjera stanje komponente
 - Promet se usmjerava samo na ispravne komponente
 - IPSentry i slična rješenja (Active Server Watch)
 - Periodički ispituje stanje komponente



Balansiranje prometa - Apache i Nginx



Perzistentne sesije (sticky sessions)

- Single NLB affinity (Microsoft NLB)
- Cookies (Apache)
- IP hash (Nginx)



Više Tomcat instanci na istom serveru

- Svaka instalirana u svom direktoriju
- Svaka sluša na svom portu
- Instalacija
 - Instalacija prve instance Tomcat-a
 - Multipliciranje Tomcat direktorija
 - Konfiguriranje Tomcat instance
 - Instalacija Windows Servisa
 - Editiranje Windows Servisa
- Mehanizam za replikaciju sadržaja



Više Tomcat instanci na istom serveru



Sve instance na
Istoj IP adresi,
različiti portovi

- **Apache Tomcat 1 (Prva instalacija)**
- Connector Port: 8080 (Default)
- Shutdown Port: 8005 (Default)
- AJP Port: 8009 (Default)
- Redirect Port: 8100 (Default)

- **Apache Tomcat 2**
- Connector Port: 8081
- Shutdown Port: 8006
- AJP Port: 8010
- Redirect Port: 8101

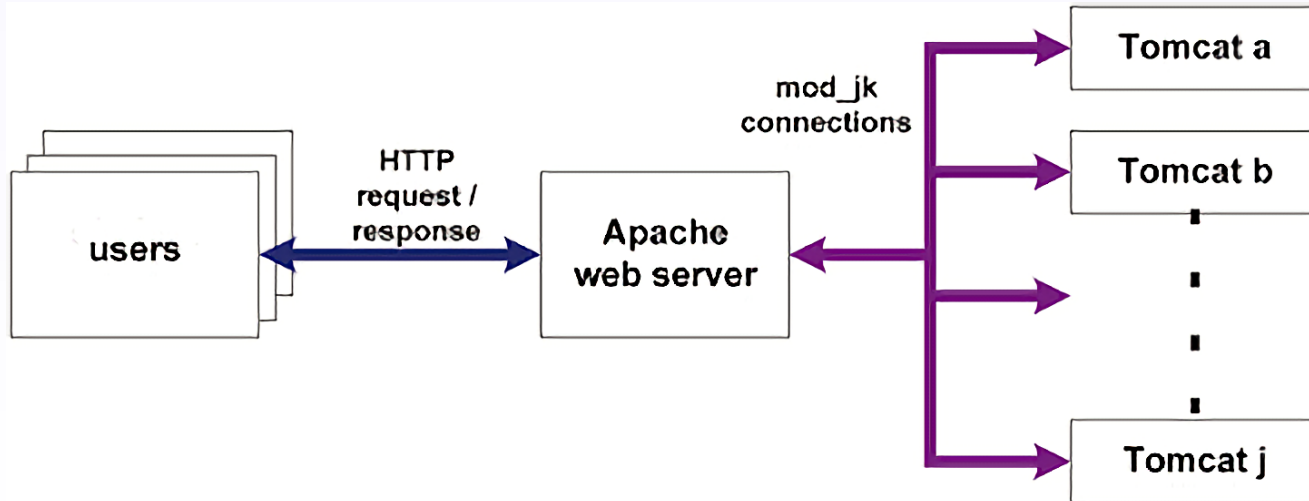
- **Apache Tomcat 3**
- Connector Port: 8082
- Shutdown Port: 8007
- AJP Port: 8011
- Redirect Port: 8102

Load Balancer - Apache

- Dva pristupa
 - Uporaba JK 1.2.x native connector-a
 - Jakarta Tomcat Connector
 - Uporaba Apache HTTP Server 2.x s mod_proxy



Load Balancer - Apache mod_jk



Load Balancer - Apache mod_jk

- Definiranje workers
- Mapiranje URLs na workers
- Konfiguracija Tomcat
 - set jvmRoute za perzistentne sesije
- Monitoring mod_jk



Load Balancer - Apache mod_jk

```
# The load balancer worker balance1 will distribute
# load to the members worker1, worker2 and worker3
worker.balance1.type=lb
worker.balance1.balance_workers=worker1, worker2, worker3
worker.worker1.type=ajp13
worker.worker1.host=myhost
worker.worker1.port=8001
worker.worker2.type=ajp13
worker.worker2.host=myhost
worker.worker2.port=8002
worker.worker3.type=ajp13
worker.worker3.host=myhost
worker.worker3.port=8003
```



Load Balancer - Apache mod_jk

- Perzistentne sesije
- Metode
 - In-memory registry
 - Cookie
 - Enkodirana instanca u session id



Load Balancer - Apache mod_jk

- Za:
 - Napredni load balancer
 - Napredna detekcija pada čvora
 - Podrška za velike AJP pakete
- Protiv:
 - Održavanje odvojenog modula

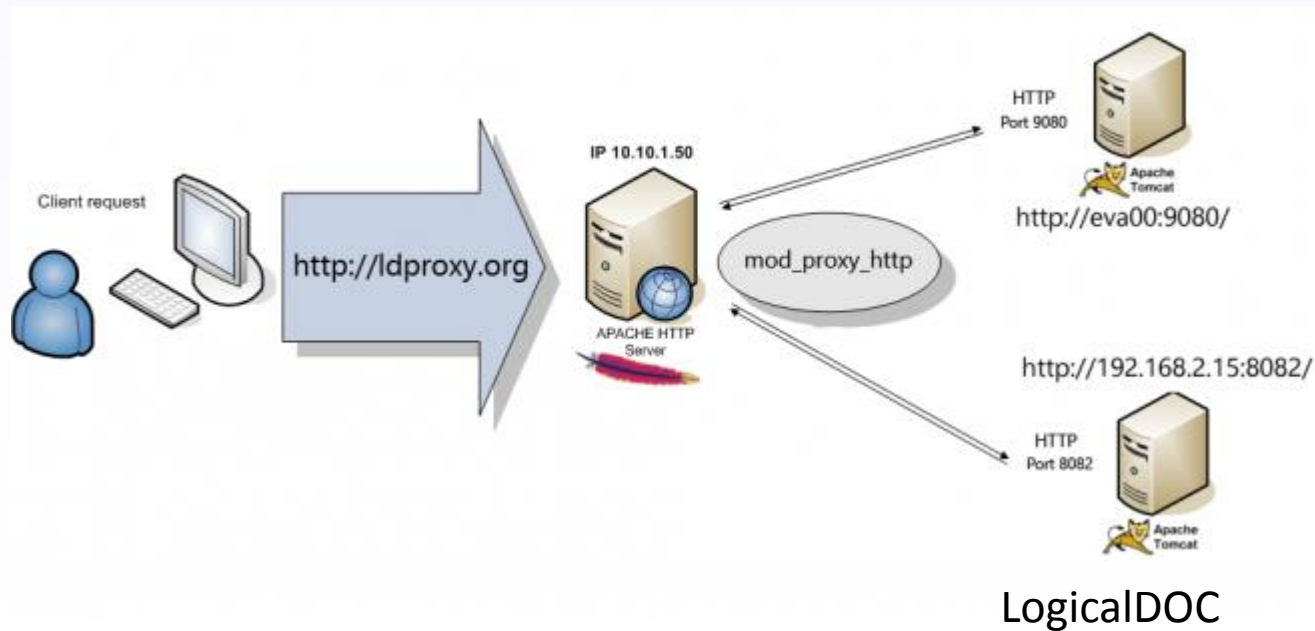


Load Balancer - Apache mod_proxy

- Uporaba Apache kao load balancera
- Round robin load balancer
 - mod_proxy
 - mod_proxy_balancer
- Peristentne sesije (sticky session – cookie)
 - mod_headers



Load Balancer - Apache mod_proxy



Load Balancer - Apache mod_proxy

- Konfiguracija

- /etc/httpd/conf/httpd.conf
- Unwanted requests
ProxyRequests off
- Balance web heads

```
<Proxy balancer://cluster>
BalancerMember http://10.x.x.x:8080
BalancerMember http://10.x.x.x:8081
BalancerMember http://10.x.x.x:8082
#security
Order Deny,Allow
Deny from none
Allow from all
#use round-robin balancing
ProxySet lbmethod=byrequests
</Proxy>
```



Load Balancer - Apache mod_proxy

- Konfiguracija

- Balance-manager

```
#balance-manager is a tool that lets you configure and tune Apache  
<Location /balancer-manager>  
SetHandler balancer-manager  
#lock this down tightly  
Order deny,allow  
Allow from all  
</Location>
```

- ProxyPass

```
#what to actually balance  
#in this case, balance everything except the manager  
ProxyPass /balancer-manager !  
ProxyPass / balancer://cluster/  
</VirtualHost>
```



Load Balancer - Apache mod_headers

- Perzistentne sesije
- Korištenje cookie-a

```
Header add Set-Cookie "ROUTEPATH=.%{BALANCER_WORKER_ROUTE}e; path=/"  
env=BALANCER_ROUTE_CHANGED
```

```
<Proxy balancer://cluster>
```

```
BalancerMember http://10.x.x.x:8080 route=1
```

```
BalancerMember http://10.x.x.x:8081 route=2
```

```
BalancerMember http://10.x.x.x:8082 route=3
```

```
ProxySet stickysession=ROUTEPATH
```

```
</Proxy>
```



Load Balancer - Apache mod_proxy

- Za:
 - Nema potrebe za kompajliranjem i održavanjem odvojenog modula mod_proxy
 - mod_proxy_http, mod_proxy_ajp i mod_proxy_balancer dolaze kao dio standardne Apache 2.2+ distribucije
 - Može koristiti http https i AJP protokole, čak unutar istog balancer-a
- Protiv:
 - mod_proxy_ajp ne podržava velike pakete 8K+
 - Jednostavan load balancer
 - Ne podržava Domain model clustering



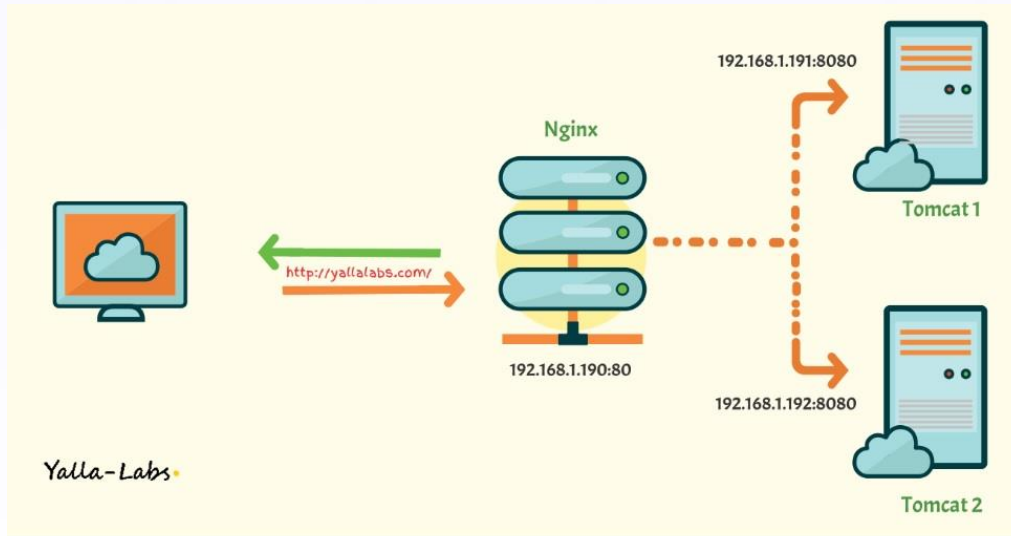
Load Balancer - Nginx

- Uporaba Nginx kao load balancera
- Nginx ('engine x') je HTTP i reverse proxy server
- IP hash, round robin, least-connected
- Perzistentne sesije (sticky sessions), implementirane sa IP hash load-balancing algoritmom

```
http {  
    upstream tomcat_servers{  
        ip_hash;  
        server 10.x.x.x:8080;  
        server 10.x.x.x:8081;  
        server 10.x.x.x:8082;  
    }  
}
```

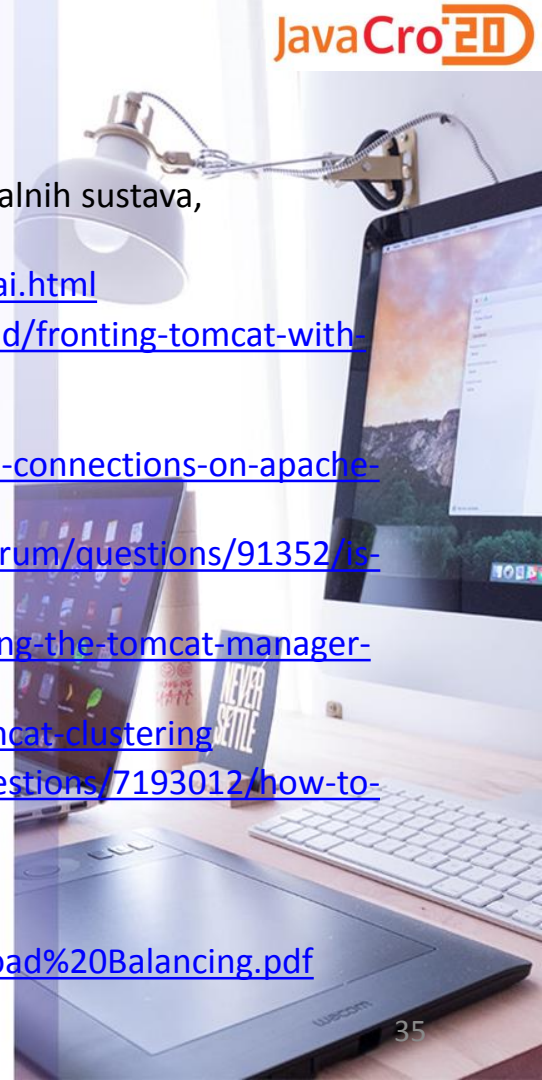


Load Balancer - Nginx



Reference

- A. Kraš, Jasminka Bonato i Biserka Draščić Ban, Pouzdanost i raspoloživost digitalnih sustava, <https://repository.pfri.uniri.hr/islandora/object/pfri%3A1371>
- M. Turk, Fronting Tomcat, <https://people.apache.org/~mturk/docs/article/ftwai.html>
- M. Turk, Fronting Tomcat with Apache Httpd, <https://documents.pub/download/fronting-tomcat-with-apache-httpd-mladen-turk-red-hat-inc-5687edc783321>
- M. Cropper, How to Increase the Maximum Connections on Apache Tomcat <https://www.contradodigital.com/2018/06/16/how-to-increase-the-maximum-connections-on-apache-tomcat/>
- Is there a limit to the maxThreads value for Tomcat server?, <https://jazz.net/forum/questions/91352/is-there-a-limit-to-the-maxthreads-value-for-tomcat-server>
- K. Morris, Configuring the Tomcat Manager Webapp, <http://kief.com/configuring-the-tomcat-manager-webapp.html>
- Tomcat Clustering - A Step By Step Guide, <https://www.mulesoft.com/tcat/tomcat-clustering>
- How to handle 2000+ requests/sec on tomcat?, <https://stackoverflow.com/questions/7193012/how-to-handle-2000-requests-sec-on-tomcat>
- Apache JMeter™, <https://jmeter.apache.org/>
- T. Bourke, Server Load Balancing, <http://www.cesarkallas.net/arquivos/livros/informatica/network/Server%20Load%20Balancing.pdf>



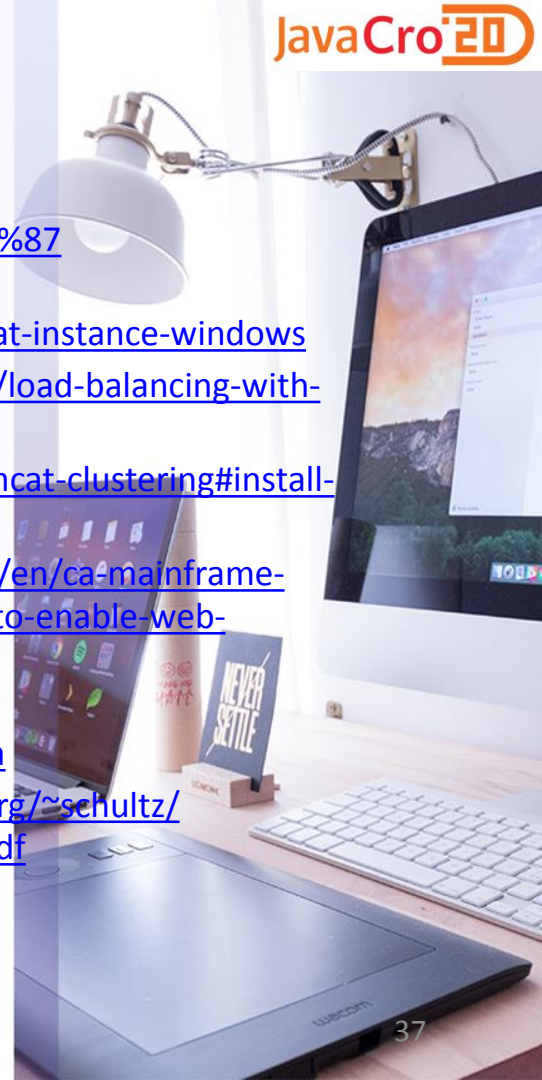
Reference

- Network Load Balancing, <https://docs.microsoft.com/en-us/windows-server/networking/technologies/network-load-balancing>
- D. Miljković, Metode balansiranja opterećenja (load balancinga) za aplikacijske servere, HROUG 2011, https://www.hroug.hr/content/download/4843/74156/file/601_Miljkovic.pdf
- How to use Apache reverse proxy as Load Balancer, <https://linuxtechlab.com/use-apache-reverse-proxy-as-load-balancer/>
- Apache Reverse Proxy Guide, https://httpd.apache.org/docs/trunk/howto/reverse_proxy.html
- What is a Reverse Proxy vs. Load Balancer?, <https://www.nginx.com/resources/glossary/reverse-proxy-vs-load-balancer/>
- J. Boadas, Apache Tomcat Load Balancing Tutorial, <https://examples.javacodegeeks.com/enterprise-java/tomcat/apache-tomcat-load-balancing-tutorial/>
- The Apache Tomcat Connectors - Common HowTo, http://tomcat.apache.org/connectors-doc/common_howto/loadbalancers.html
- Load balancing Tomcat with Apache, <https://devellopaper.com/load-balancing-tomcat-with-apache/>
- How can I reach load balancing using Tomcat clustering?, <https://www.quora.com/How-can-I-reach-load-balancing-using-Tomcat-clustering>
- Running multiple instances of Tomcat with single server installation, <https://howtodoinjava.com/tomcat/running-multiple-instances-of-tomcat-with-single-server-installation/>



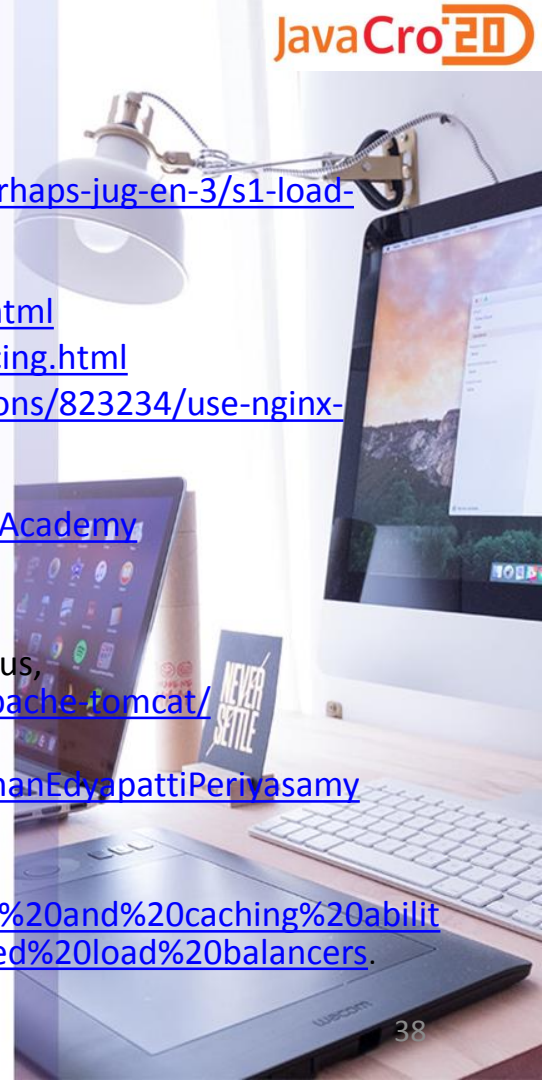
Reference

- G. Vasić, Tomcat - Multiple instances on Windows, https://www.youtube.com/watch?v=eJzD8tCwFI&ab_channel=GoranVasi%C4%87
- A. Cheung, 5 Steps install multiple Apache Tomcat instance on Windows, <http://www.ansoncheunghk.info/article/5-steps-install-multiple-apache-tomcat-instance-windows>
- Load Balancing With Apache, <https://www.beginlinux.com/server/web-server/load-balancing-with-apache>
- Tomcat Clustering - A Step By Step Guide, <https://www.mulesoft.com/tcat/tomcat-clustering#install-modjk>
- Configure Apache Tomcat Load-balancing, <https://techdocs.broadcom.com/us/en/ca-mainframe-software/devops/ca-endeavor-software-change-manager/18-0/installing/how-to-enable-web-services/configure-apache-tomcat-load-balancing.html>
- Load Balancing, clustering with Apache & Tomcat, https://www.youtube.com/watch?v=yNuuQLw0tA&ab_channel=abanibehera
- Intro to Load-Balancing Tomcat with httpd and mod_jk, http://home.apache.org/~schultz/ApacheCon%20NA%202015/Load-balancing%20Tomcat%20with%20mod_jk.pdf
- Load Balancer How-To, https://tomcat.apache.org/tomcat-8.5-doc/balancer-howto.html#Using_the_JK_1.2.x_native_connector



Reference

- Load Balancing at the Web Level with mod_jk, <https://aws.huihoo.com/jonas/rhaps-jug-en-3/s1-load-balancing.html>
- Java And Eelated, mod_jk, mod_proxy and mod_proxy_ajp, <http://javafatihk.blogspot.com/2014/11/modjk-modproxy-and-modproxyajp.html>
- Using nginx as HTTP load balancer, http://nginx.org/en/docs/http/load_balancing.html
- Use nginx upstream group with multiple ports, <https://serverfault.com/questions/823234/use-nginx-upstream-group-with-multiple-ports>
- Linux Training Academy, HTTP Load Balancing with Nginx, https://www.youtube.com/watch?v=SpL_hJNUNEI&ab_channel=LinuxTrainingAcademy
- How to configure Nginx as a load balancer for apache Tomcat servers, <https://yallalabs.com/linux/how-to-configure-nginx-load-balancer-tomcat/>
- Load Balancing Apache Tomcat Servers with NGINX Open Source and NGINX Plus, https://docs.nginx.com/nginx/deployment-guides/load-balance-third-party/apache_tomcat/
- R. E. Periyasamy, Tomcat Clustering Series Part 5 : NginX Load Balancer, https://www.youtube.com/watch?v=zRPJ_U2MrU&ab_channel=RamakrishnanEdyapattiPeriyasamy
- Nginx, <http://www.aosabook.org/en/nginx.html#fig.nginx.arch>
- Nginx vs Apache, <https://anturis.com/blog/nginx-vs-apache/#:~:text=Both%20Apache%20and%20Nginx%20can,load%20balancing%20and%20caching%20abilities.&text=Apache%20has%20a%20load%20balancer,are%20hardware%20based%20load%20balancers.>



Zaključak

- Veliki kapacitet uporabom više Tomcat instanci
- Više Tomcat instanci na istom serveru
- Razna rješenja za balansiranje prometa
 - Unutar istog servera programska rješenja
 - Apache mod_jk
 - Apache mod_proxy
 - Nginx
- Visoka raspoloživost uporabom više servera
 - Između server moguća uporaba NLB
 - Arhitektura sabirnice bez pojedinačne točke kvara



Hvala na pažnji!