

USMJERAVANJE PROMETA KA WEBLOGIC INSTANCAMA NA JEDNOM ILI VIŠE POSLUŽITELJA

TRAFFIC DIRECTION TO MULTIPLE WEBLOGIC INSTANCES ON ONE OR MORE SERVERS

Dubravko Miljković

HEP-SIT, Vukovarska 37, Zagreb

Mob: 098 9825602

E-mail: dubravko.miljkovic@hep.hr

SAŽETAK

Pojedinačna instanca WebLogic aplikacijskog poslužitelja je ograničena memorijom na 2 GB RAM-a što postavlja ograničenja na broj korisnika koji instanca može opsluživati. Za veliki broj korisnika uobičajeno je korištenje više WebLogic instanci. Pri tome različite WebLogic instance, ako su instalirane na istom poslužitelju, koriste različite mrežne portove. Potrebno je usmjeriti korisnika ka više WebLogic instanci na istoj ili različitim IP adresama te na različitim portovima. Iznesena su tri rješenja: Oracle Web Cache kao load balancer, jednostavni redirektor u PERL-u i novi Oracle Traffic Director. Prve dvije varijante ako koriste više poslužitelja mogu se kombinirati s balansiranjem mrežnog prometa da se izbjegne jedinstvena točka prekida.

ABSTRACT

An individual instance of WebLogic application server has 2 GB RAM memory limit which sets limits on the number of users that can be served by an instance. To accommodate large number of users it is common to use multiple WebLogic instances. If installed on the same server these WebLogic instances use different network ports. Thus, it is necessary to direct users to multiple WebLogic instances with same or different IP addresses and different listening ports. Three solutions are presented: Oracle Web Cache configured as a load balancer, simple redirector written in Perl and newest Oracle Traffic Director. The first two variants when used with multiple servers could be combined with network load balancing to remove single point of failure.

1. UVOD

WebLogic je novi aplikacijski server 11g, Nasljednik iAS 10g. JVM na pojedinačnoj WebLogic instanci ograničena na 2 GB. Ovo ograničenje se javlja na 32-bitnom operacijskom sustavu. Na 64-bitnom OS-u ograničenje ne proizlazi iz OS-am ali JVM ne iskorištava učinkovito više od 2 GB memorije, pa se suočavamo sa istim problemom. Ovisno o složenosti aplikacije, broj korisnika za Java aplikacije po instanci je reda 50-100, rijetko 200 (za vrlo jednostavne aplikacije), [1]. Za Forms/Reports aplikacije broj korisnika ovisi o non-interactive desktop heap size i slobodnoj RAM memoriji. Stoga kod većeg broja korisnika brzo dolazimo do potrebe korištenja više WebLogic instanci.

2. JEDNA ILI VIŠE WEBLOGIC INSTANCI NA ISTOM SERVERU

Sve do prije nekoliko godina serveri u računskim centrima su imali dvije, rijetko četiri jezgre i do 4 GB RAM-a. Sada jedan moderan server ima 4-8 jezgri više od 4 GB RAM-a (slika 1). Upravo je šteta takve moćne server iskoristiti za instalaciju samo jedne WebLogic instance jer njegove mogućnosti uvelike nadmašuju potrebe jedne WebLogic instance.



Slika 1 Suvremeni server s 8 jezgri - sve tanji, sve moćniji (izvor HP)

2.1. Zašto više instanci na jedan fizički server

Oppluživanje velikog broja korisnika glavni je razlog korištenja više WebLogic instanci. Osim tog postizemo i redundanciju – sustav i dalje radi kod pada jedne instance. Sam WebLogic nije zahtjevan na CPU te to ne predstavlja problem. S druge strane moderni serveri imaju dovoljno RAM-a i omogućuju instalaciju više WebLogic instanci. Pri tome više WebLogic instanci na istom fizičkom serveru koriste isti OS pa uz smanjeni utrošak RAM memorije ostvaruju se niži troškovi zbog licenciranja samo jednog OS-a.

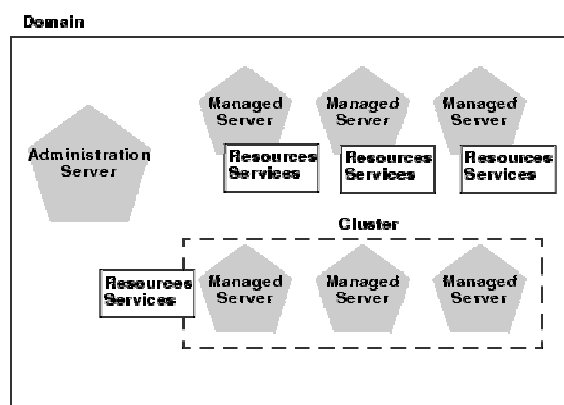
2.2. Zašto više instanci na jedan virtualni server server

S dolaskom virtualizacije i virtualnih servera problem opsluživanja velikog broja korisnika može se riješiti instaliranjem više WebLogic instanci na isti virtualni server. Sama virtualizacija donosi određene prednosti u odnosu na fizičke servere, prvenstveno u sferi veće fleksibilnosti, boljeg iskorištenja serverske infrastrukture, a zbog lakog seljenja virtualnih strojeva pogodni su za realizaciju Business Continuity i Disaster Recovery (BCDR) rješenja.

Moderni fizički serveri mogu imati ugrađene izrazito velike količine RAM memorije (16 GB+) što omogućuje instaliranje većeg broja virtualnih servera pri čemu i svaki virtualni server ima na raspolaganju dovoljno RAM-a da bez problema podrži instalaciju više WebLogic instanci. Pri tome kao i kod fizičkog servera više WebLogic instanci na istom virtualnom serveru koriste isti OS pa uz smanjeni utrošak RAM memorije ostvaruju se niži troškovi zbog licenciranja samo jednog OS-a po virtualnom serveru.

3. WEBLOGIC DOMAIN I CLUSTER

Prilikom instalacije WebLogic servera potrebno je imati jedan administrativni server i jedan ili više (upravljan) managed servera. Administrativni server omogućava centralizirano upravljanje i poželjno ga je instalirati na zaseban fizički ili virtualni server. Managed server je instanca na kojoj se izvršavaju aplikacije. Postoji i Node manager koji se instalira na svaki fizički ili virtualni server na kojem se nalaze managed serveri. On omogućava remotly start/stop i restart. Managed serveri mogu se staviti u domenu i u cluster (slika 2). Domenu čini jedna ili više WebLogic instance s kojima se upravlja s istog administrativnog servera, [2]. Domena je pogodna za distribuciju iste konfiguracija svim svojim članovima. Upravljeni (managed) nodovi mogu se staviti u cluster, [3]. Cluster čini grupa managed servera koji rade simultano i zajedno. WebLogic Cluster može se iskoristiti prilikom usmjeravanja prometa na instance za ADF, ali ne donosi prednosti kod balansiranja prometa Forms sesija.



Slika 2 WebLogic domena (izvor Oracle)

4. WEBLOGIC LOAD BALANCING I PORTOVI

WebLogic instanca kad je instalirana kao prva (ili možda čak jedina) WebLogic instance na serveru po default-u koristi sljedeće portove.

- WebLogic portovi nakon instalacije
 - AdminServer 7001
 - HTTP 8888
 - Forms 9001
 - Reports 9002
 - ADF 7003

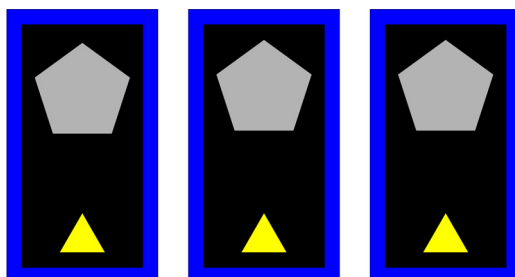
Dodatne instance na istom fizičkom ili virtualnom serveru automatski zauzimaju slijedeće slobodne portove. Vrijednosti za portove se mogu promijeniti izmjenama u /stage/Response/staticports.ini file.

Prilikom instaliranja jedne ili više WebLogic instanci na jedan ili više servera nailazimo na slijedeće kombinacije koje će biti razmotrene (slike 3-9):

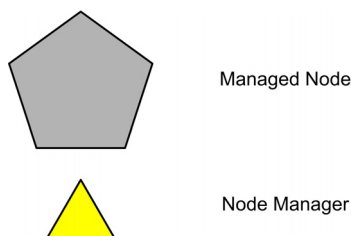
- jedan server - jedna WebLogic instanca
- jedan virtualni server - jedna WebLogic instanca
- jedan server - više WebLogic instanci
- jedan virtualni server - više WebLogic instanci
- više servera sa više WebLogic instanci
- više virtualnih servera više WebLogic instanci

4.1. Jedan server - jedna WebLogic instanca

- Svaka instanca na zasebnoj IP adresi
- WebLogic instance i linkovi (primjer)
 - <http://szg01weblogic01:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic02:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic03:9000/forms/frmservlet?config=gsm>
- Aplikacija koristi isti port na svim instancama
- Usmjeravanje prometa je jednostavno ostvarivo uobičajenim metodama



Slika 3 Jedan server - jedna WebLogic instanca

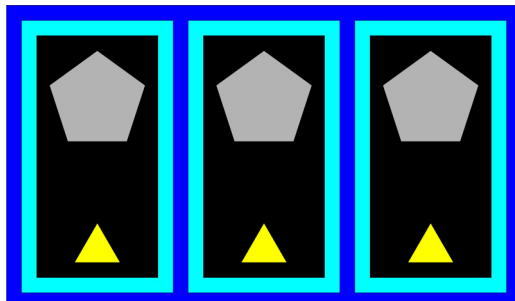


Slika 4 Legenda

4.2. Jedan virtualni server - jedna WebLogic instanca

- Slična situacija, ovaj put WebLogic instance na zasebnim virtualnim strojevima
- WebLogic instance i linkovi (primjer)

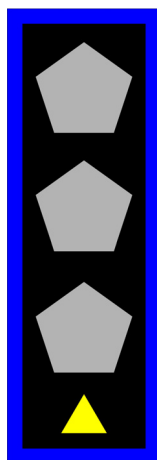
- <http://szg01weblogic01:9000/forms/frmservlet?config=gsm>
- <http://szg01weblogic02:9000/forms/frmservlet?config=gsm>
- <http://szg01weblogic03:9000/forms/frmservlet?config=gsm>
- Svaka instanca na zasebnoj IP adresi
- Aplikacija koristi isti port na svim instancama
- Usmjeravanje prometa je jednostavno ostvarivo uobičajenim metodama



Slika 5 Jedan virtualni server - jedna WebLogic instanca

4.3. Jedan server - više WebLogic instanci

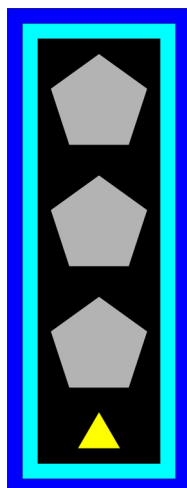
- WebLogic instance i linkovi (primjer)
 - <http://szg01weblogic01:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic01:9002/forms/frmservlet?config=gsm>
 - <http://szg01weblogic01:9004/forms/frmservlet?config=gsm>
- Svaka instanca istoj IP adresi
- Aplikacija koristi različiti port na svakoj instanci
- Potrebno usmjeravanje prometa ka više portova



Slika 6 Jedan server - više WebLogic instanci

4.4. Jedan virtualni server više WebLogic instanci

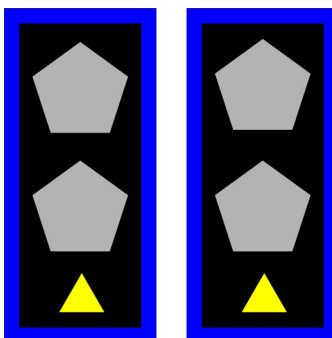
- Slična situacija, ovaj put više WebLogic instanci na zasebnim virtualnim strojevima
- WebLogic instance i linkovi (primjer)
 - <http://szg01weblogic01:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic01:9002/forms/frmservlet?config=gsm>
 - <http://szg01weblogic01:9004/forms/frmservlet?config=gsm>
- Svaka instanca istoj IP adresi
- Aplikacija koristi različiti port na svakoj instanci
- Potrebno usmjeravanje prometa ka više portova



Slika 7 Jedan virtualni server više WebLogic instanci

4.5. VIŠE SERVERA SA VIŠE WEBLOGIC INSTANCI

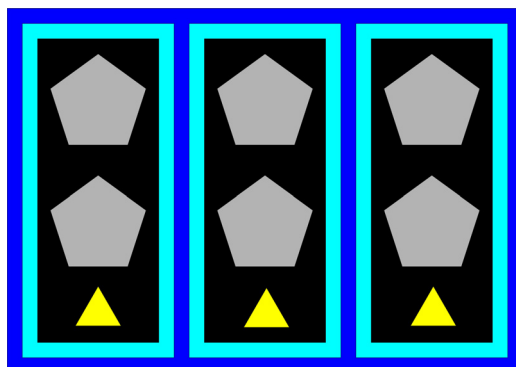
- WebLogic instance i linkovi (primjer)
 - <http://szg01weblogic01:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic01:9002/forms/frmservlet?config=gsm>
 - <http://szg01weblogic02:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic02:9002/forms/frmservlet?config=gsm>
- Aplikacija koristi različite IP adrese i različite portove



Slika 8 Više virtualnih servera - više weblogic instanci

4.6. Više virtualnih servera - više weblogic instanci

- Slična situacija, ovaj put više WebLogic instanci na više virtualnih strojeva
- WebLogic instance i linkovi (primjer)
 - <http://szg01weblogic01:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic01:9003/forms/frmservlet?config=gsm>
 - <http://szg01weblogic02:9000/forms/frmservlet?config=gsm>
 - <http://szg01weblogic02:9003/forms/frmservlet?config=gsm>
- Aplikacija koristi različite IP adrese i različite portove



Slika 9 Više virtualnih servera - više WebLogic instanci

5. RJEŠENJA BEZ DODATNOG HARDWARE-A I SOFTWARE-A

Ova rješenja uključuju menu stranice gdje su fiksno dodijeljeni url-ovi za svaku instancu (slika 10). Pogodna su kad imamo više aplikacija, pa ih raspodijelimo po raspoloživim instancama. Da bi koliko-toliko ravnomjerno rasporedili opterećenje po instancama potreban je bar okvirna procjena korištenja aplikacija (broj korisnika tijekom nekog perioda), a treba razmotriti i složenost aplikacija (da ne bi sve jednostavne aplikacije stavili na jednu instancu, a sve složene na drugu). U biti procjena bi trebala ponderirati korištenje aplikacije sa njenom složenošću (i time zahtjevom uglavnom na memoriju s kojom raspolaže instanca).

5.1. Menu links pages

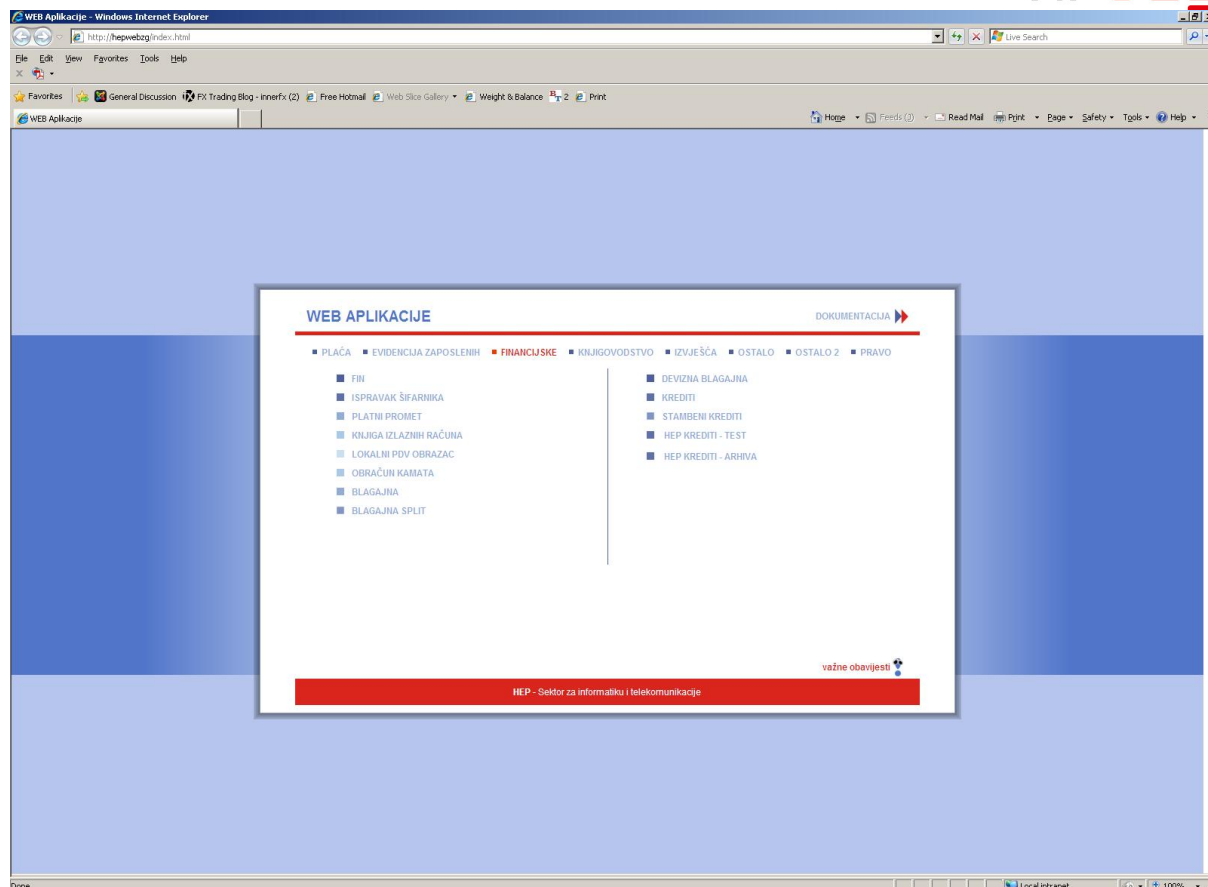
- Specificiranje URL-ova koji vode do aplikacija na pojedinim instancama
- Koristi HTTP port
 - Setirati port 80 (umjesto default-nog porta 8888) da bi korisnik došao na meu stranicu
- DocumentRoot
 - Specificiran u weblogic.properties datoteci
 - Pokazuje na weblogic/myserver/public_html direktorij
- weblogic.httpd.documentRoot=*path*weblogic/myserver/public_html

5.2. Grupe aplikacija na instancu

Umjesto balansiranja prometa aplikacije se podijele u grupe. Pri tome se grupe aplikacija približno jednakog ukupnog broja korisnika i složenosti vežu se za pojedinu instancu kako bi ih ravnomjerno opteretile.

Na taj način imamo npr. slijedeću podjelu:

- Instanca 1 (Grupa 1)
 - Aplikacija 1
 - Aplikacija 2
 - Aplikacija 3
- Instanca 2 (Grupa 2)
 - Aplikacija 4
 - Aplikacija 5
 - Aplikacija 6

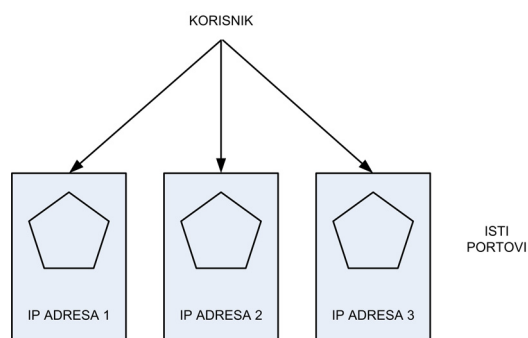


Slika 10 Menu links page (početna strana s linkovima na aplikacije)

6. RJEŠENJA KOJA PODRŽAVAJU SAMO RAZLIČITE ADRESE

Pretpostavlja se da su sve WebLogic instance na zasebnom serveru i koriste isti port (slika 11). Ova rješenja uključuju:

- Klasični load balancer
- Network Load Balancing (NLB)



Slika 11 Korisnik se usmjerava ka instancama na različitim adresama. Sve instance koriste isti port.

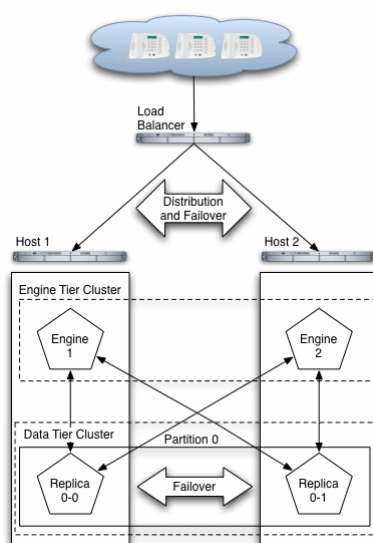
6.1. Klasični load balancer

Klasično rješenje koje je često integrirano u switch (slika 12), [4, 5]. Samostalno (bez kombiniranja s drugim metodama koje bi usmjeravale korisnika na različite portove unutar iste IP adrese) podržava jednu WebLogic instancu po IP adresi. Pogodni su za raspodjelu opterećenja kako između fizičkih tako i između virtualnih servera (slika 13). Affinity se obično ostvaruje uz pomoć

cookie-a. S Load balancerom ne moramo balansirati pune linkove koji vode do aplikacije već Usmjeriti korisnika na početnu menu stranicu instance, a sva daljnja komunikacija može se ostvarivati direktno s pripadnom instancom što uklanja i problem affinity-a. Najnoviji load balanceri imaju mogućnost balansiranje prometa i po IP adresama i po portovima, ali takav egzotičan zahtjev osoblje zaduženo za mreže obično ne zna konfigurirati (ali može tražiti pomoć tehničke podrške proizvođača, npr. CISCO podrška tvrdi da je to moguće konfigurirati).

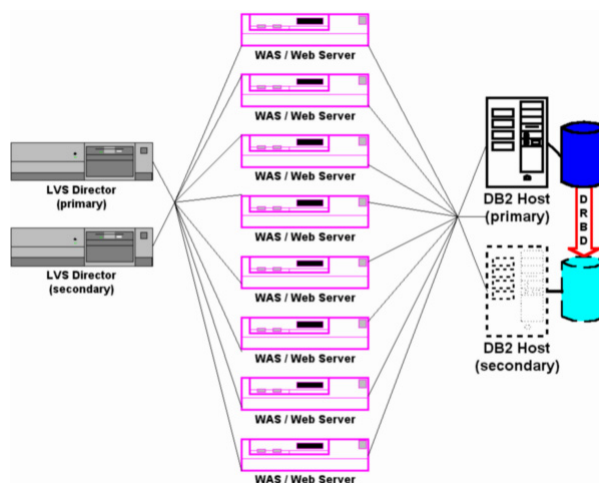


Slika 12 Klasični load balancer (izvor CoyotePoint)



Slika 13 Load balancer osigurava usmjeravanje prometa i failover u slučaju ispada instance (izvor Oracle)

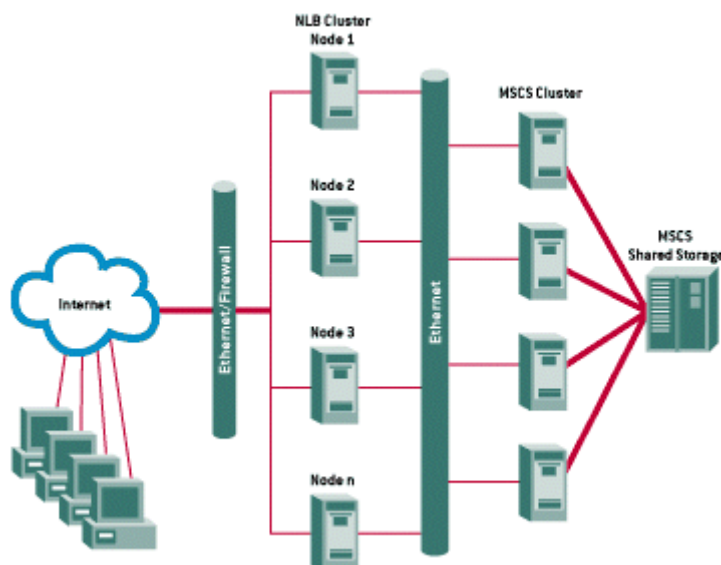
Ukoliko želimo izbjegnuti single point of failure (High Availability rješenje) potrebna su nam dva balancera (slika 14).



Slika 14 Dva load balancera za High Availability (izvor TechThoughts)

6.2. Network load balancing

Samostalno (bez kombiniranja s drugim metodama koje bi usmjeravale korisnika na različite portove unutar iste IP adrese) podržava jednu WebLogic instanca po IP adresi. Može se koristiti za raspodjelu opterećenja između fizičkih i između virtualnih servera (slika 15), [6, 7]. Kod virtualnih servera situacija može biti složenija (uporaba virtualnih mrežnih kartica), mogući su dodatni problemi (ponekad NLB radi samo između virtualnih strojeva na različitim fizičkim strojevima) iako je to danas s novim revizijama software-a za virtualizaciju otklonjeno.



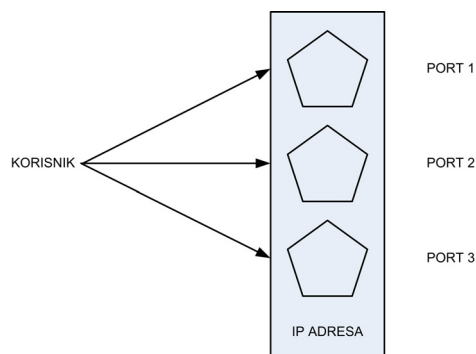
Slika 15 NLB load balancing u NLB clusteru (izvor Orion eSolutions)

7. RJEŠENJA KOJA PODRŽAVAJU RAZLIČITE ADRESE I PORTOVE

Postoje rješenja koja podržavaju WebLogic instance koje mogu biti na različitim IP adresama i različitim portovima (slika 16).

Među ova rješenja spadaju:

- Web Proxy Server
- Oracle HTTP Server (mod_wl_ohs)
- Web Cache
- Perl random HTTP Redirector
 - Mogu se kombinirati sa klasičnim load balancerima i NLB-om
- High availability loadbalancer for WebLogic cluster
- Oracle Traffic Director
 - Može se isto kombinirati s NLB-om (slično kao i web cache i Perl HTTP redirector, ali je predober samo za jedan nod



Slika 16 Usmjeravanje korisnika na WebLogic instance koje koriste različite portove

7.1. Load Balancing Domain Setup sa Proxy Plug-in

Kombinira cluster i load balancing domenu (A. Baranovskis, M. Negi), [8]. Glavni koraci su:

- Postaviti WebLogic Cluster
- Na kraju promijeniti Multicast messaging type u Unicast
- Load Balancing Domain Setup sa Proxy Plug-in
- Promijeniti HttpClusterServlet putem WebLogicCluster parametra web.xml i weblogic.xml u load-balancer.war

7.2. Load Balancing sa HTTP serverom

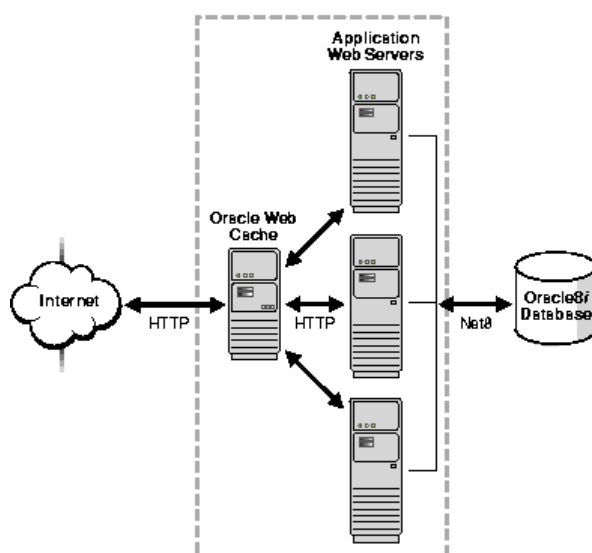
Konfiguriranje Oracle HTTP Servera ispred Oracle WebLogic Servera, [9]. Glavni koraci su:

- WebLogic instance staviti u cluster
- Instalirati Oracle HTTP Sever
- Modificirati mod_wl_ohs.conf

```
<Location /myServerURL>
  SetHandler weblogic-handler
  WebLogicCluster MS1:7010,MS2:7010
</Location>
```

7.3. Web cache

Osim cach-iranja statičkih sadržaja web cache, može se koristiti i kao load balancer (slika 17), [10]. Web cache u funkciji load balancera podržava više WebLogic instanci na istoj IP adresi koje mogu koristiti različite portove. Web cache može se aktivirati na na jednoj instanci. Sam web cache posjeduje jednostavni mehanizam ispitivanja ispravnosti destinacijskih nodova. Nedostatak rješenja sa web cache-om je taj što web cache predstavlja single point of failure (zbog jedinstvene pristupne točke i pouzdanosti samog web cache-a i servera na kojem je instaliran).

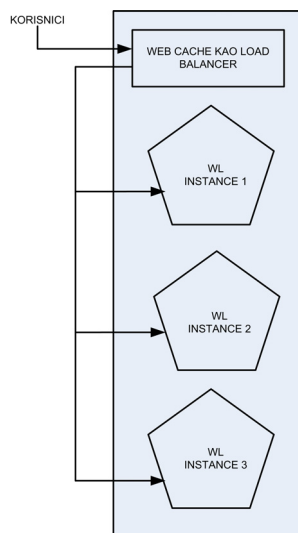


Slika 17 Usmjeravanje HTTP prometa uz pomoć web cache-a (izvor Oracle)

Web cache instalira se kao dodatna komponenta na serveru (slika 18). Kod konfiguriranja load balancinga prijavljuju se sve instance i koriste se slijedeći parametri):

- Sites
- Origin Servers
- Site-to-Server
- Session Binding

Za više aplikacija po instanci mogu se koristiti menu stranice instance pri čemu se url-ovi se na menu stranicama razlikuju samo po portovima i konfiguraciji.



Slika 18 Usmjeravanje HTTP prometa uz pomoć web cache-a

Kod web cache-a affinity se uglavnom ostvaruje primjenom sesija koje koriste Cookie.

Slika 19 ilustrira primjer prijavljivanja instanci prema kojima se balansira promet, [11].

WC1 | Logged in as abc | Host edrsr15p1.us.oracle.com
 Web Cache | Page Refreshed Feb 22, 2010 5:22:06 PM GMT+07:00

Information
 All changes made in this page require a server restart to take effect.

Origin Servers [Apply] [Revert]

Configure Web Cache with the application Web servers for internal sites and proxy servers for external sites outside a fire sends cache misses. These settings are required for load balancing, failover, and site-to-server mappings.

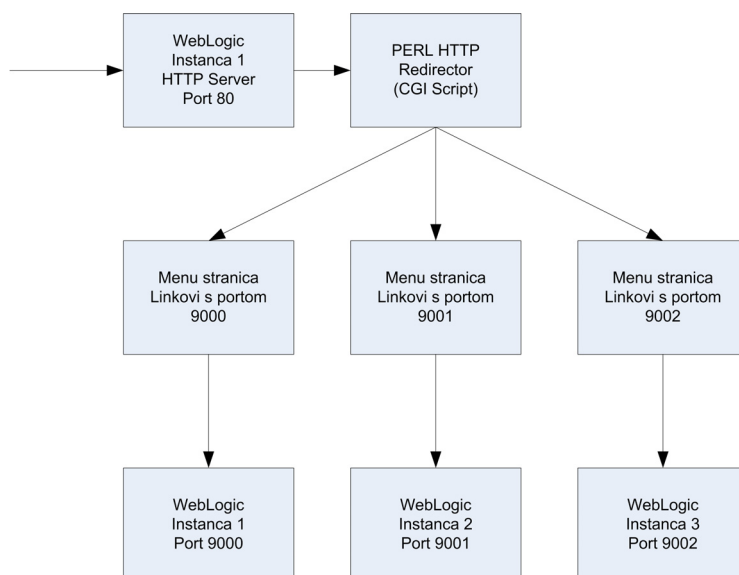
Create... Edit... Delete...

Host	Port	Protocol	Routing Enabled	Capacity
localhost	7780	HTTP	<input checked="" type="checkbox"/>	100
localhost	7781	HTTP	<input checked="" type="checkbox"/>	100

Slika 19 Primjer prijavljivanja instanci (izvor Oracle)

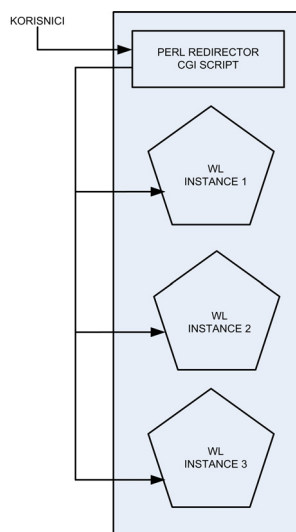
7.4. Perl random HTTP redirector

Kao alternativa web cache-u može se koristiti random HTTP redirector. Perl HTTP redirector svaki put slučajno usmjerava (HTTP redirect) korisnika na početnu (menu) stranicu jedne od instanci (slike 20 i 21). Pri tom ne ispituje stanje servera (za to je potreban dodatni Sentry software). Nakon redirekcije korisnik je vezan na dotičnu instancu i dalje s njom direktno komunicira (time je riješen problem affinity-ja). Sam redirector je jednostavno realizirati u PERL-u (kratak kod, jedna strana).



Slika 20 Usmjeravanje prometa na instance putem Perl HTTP redirektora

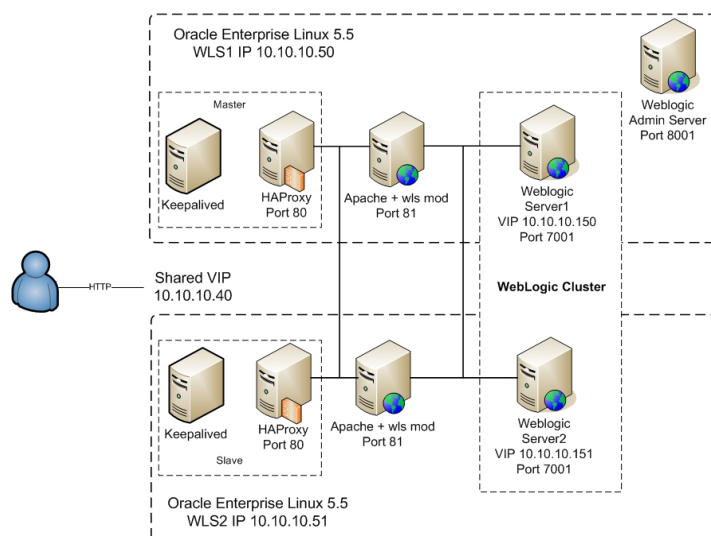
PERL random HTTP redirector vezan je na jedan HTTP server (za taj HTTP server koristi se port 80 - promijeniti default-nu vrijednost). Pri tome svaka instanca ima svoju menu stranicu s pripadnim portom specificiranim u url-ovima aplikacija. Url-ovi se na menu stranicama pojedinih WebLogic instanci razlikuju samo po portovima. Nedostaci rješenja: ukoliko padne HTTP server na koji je vezan PERL gubi se pristup svim instancama, a ukoliko padne pojedina instanca promet se i dalje usmjerava na nju.



Slika 21 Usmjeravanje prometa na instance putem Perl HTTP redirektora

7.5. High availability loadbalancer for WebLogic cluster

Rješenje kombinira dva programa: HAProxy i Keepalived u zanimljivo rješenje za usmjeravanje prometa (slika 22), [12]. HAProxy je besplatno, vrlo brzo i pouzdano rješenje koje nudi visoku dostupnost, funkcije balansiranje opterećenja i proxy poslužitelja za TCP i HTTP-temeljene aplikacije. Osobito je pogodan za web stranice pod vrlo visokim opterećenjima, koje zahtijevaju affinity ili obradu zahtjeva u Layer-u 7. Keepalived je besplatan Linux program koji može pratiti ispravnost rada HAProxy-ja i prebaciti zajedničku IP adresu, tako da u slučaju pada jednog poslužitelja zahtjevi budu obrađeni na drugom poslužitelju.



Slika 22 High availability load balancer for WebLogic cluster (izvor biemond.blogspot.com)

7.6. Oracle Traffic Director

Oracle Traffic Director trenutno je razvijen samo za UNIX. Predstavlja brzi, pouzdani i skalabilni layer-7 programski load balancer(reverse proxy) koji posjeduje velike mogućnosti konfiguriranja, [13]. Može se koristiti kao pouzdana ulazna točka za sav HTTP i HTTPS promet ka aplikacijskim i web serverima. Arhitektura Oracle Traffic Director omogućava rad s velikim volumenom aplikacijskog prometa s niskom latencijom. Može se instalirati na isti stroj s WebLogic instancama (zasebni path). Proizvod je optimiran za uporabu u Oracle Exalogic Elastic Cloud-u

Oracle Traffic Director podržava slijedeće oblike balansiranja prometa:

- Round robin
- Least connection count
- Weighted round robin
- Weighted least connection count

Može opslužiti veliki promet s niskom latencijom. Posjeduje High Availability (HA) opcije konfiguriranja na slijedeće načine: active-active instance i active-passive instance (failover). Ugrađena je provjera ispravnosti servera (Health check back end servera)

Od ostalih opcija vrijedi spomenuti:

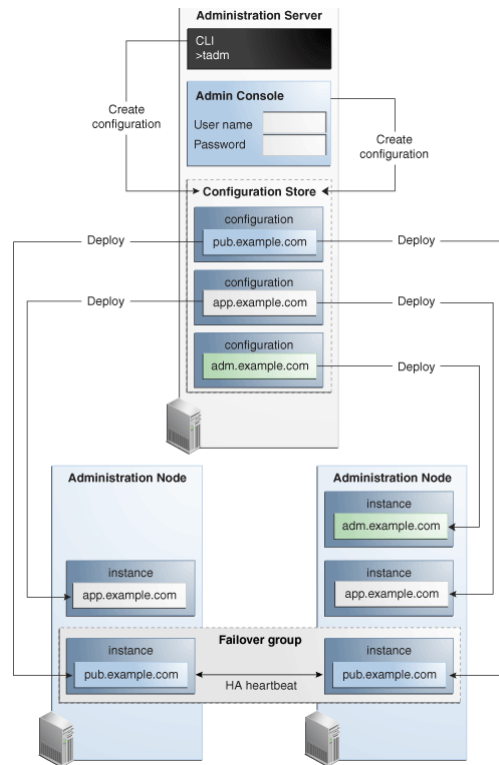
- Fleksibilno rutiranje
- Kontrola opterećenja (Load Control)
- Ubrzavanje opterećenja na novopridošlom nodu (Request rate acceleration)

Ova zanimljiva opcija omogućuje pojačano opterećivanje servera koji su tek uključeni (npr. nakon restarta WebLogic instance ili uključivanje nove instance u cluster.) Pri tome dolazni zahtjevi se prvo upućuju na novu neopterećenu instancu, a kad ona dostigne opterećenje ostalih instanci nastavlja se s ravnomjernim opterećivanjem svih instanci.

- Ograničavanje broja konekcija (Connection limiting)

Oracle Traffic Director se može instalirati na iste servere kao i same WebLogic instance (pri čemu koristi različite HOME direktoriji) pa ne zahtijeva dodatne servere. Sastoji se od dvije glavne komponente (slika 23):

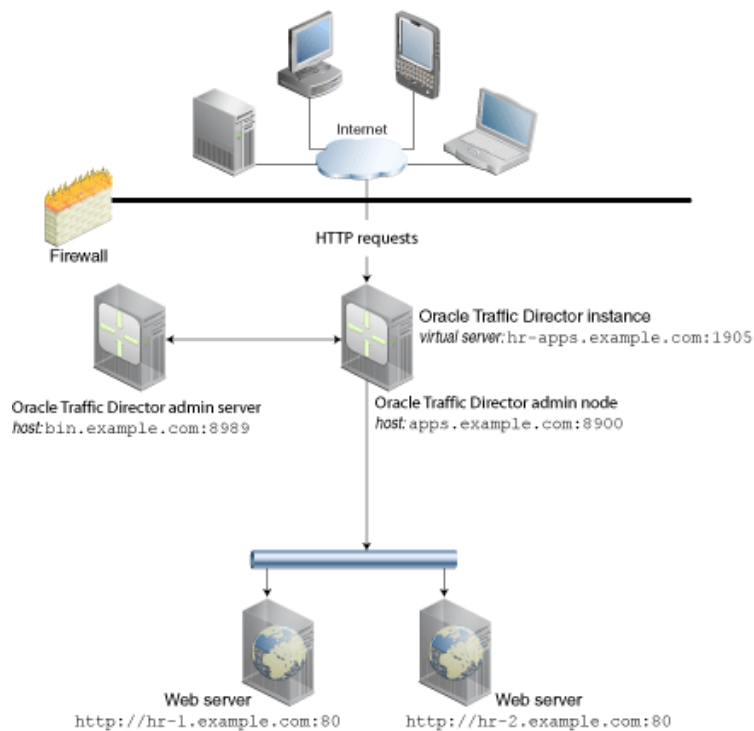
- Administration Server
- Administration Node (Oracle Traffic Director Node)



Slika 23 Oracle Traffic Director (izvor Oracle)

Za visoku raspoloživost (High Availability - HA) mogu se koristiti dvije instance Oracle Traffic Director-a u paru i to u slijedećim načinima rada kao što je prikazano na slici 24:

- active-passive failover
- active-active



Slika 24 Oracle Traffic Director konfiguriran za High Availability (izvor Oracle)

8. PREGLED RJEŠENJA

Rješenja koja su spomenuta u prethodnom dijelu rada prikazana su u tablici I.

Tablica I Pregled rješenja za usmjeravanje prometa ka WebLogic instancama

	Jedan WL na fizički	Jedan WL na virtualni	Više WL na fizički	Više WL na virtualni	Napomena
Menu links	+	+	+	+	Nije fleksibilno
Klasični LB	+	+	-	-	Isti port
NLB (Windows)	+	+	-	-	Isti port
LB Domain + Proxy Plugin	+	+	+	+	
HTTP Server (mod_wl_ohs)	+	+	+	+	
WEB Cache	+	+	+	+	SPOF*, zastarjelo
PERL HTTP redirektor	+	+	+	+	SPOF*, random
HAProxy + Keepalived	+	+	+	+	LINUX
Oracle Traffic Director	+	+	+	+	LINUX, složeno ali najbolje

SPOF – Single Point of Failure

9. ZAKLJUČAK

Izloženo je više metoda usmjeravanja korisnika na instance. Odabir rješenja ovisi o OS-u (Windows, Unix) kao i o potrebi za primjenom High Availability (HA) rješenja. Ukoliko ne želimo primijeniti nikakav dodatni hardware ni software preostaje nam rješenje s početnom menu stranicom i pripadnim od aplikacije (ili grupe aplikacija) specifičnih aplikacijskih linkova na pojedinačne instance. Za Windows i Linux okruženja s WebLogic instancama unutar istog fizičkog ili virtualnog servera moguće je primijeniti slijedeća rješenja: Web Proxy server, Oracle HTTP server (sa mod_wl_ohs), web cache (malo zastarjelo rješenje) konfiguriran kao load balancer i Perl random HTTP redirektor. Pri tome web cache i Perl random HTTP redirektor usmjeravaju zahtjeve po pojedinačnim instancama inicijalnim usmjeravanjem na početnu menu stranicu dotičnih instanci. Za usmjeravanje prometa između WebLogic instanci na više fizički ili više virtualnih servera moguće je odabrati između klasičnog load balancer-a (mrežni uređaj), network load balancing-a (Windows NLB ili Linux specifično rješenje) i web cache. Kod rješenja sa klasičnim load balancer-om ili NLB-om moguće je unutar pojedinačnih servera daljnje usmjeravanje po portovima ostvariti uz pomoć web cache-a ili Perl random HTTP redirektora. Za okruženja zasnovana na Linux-u najbolje je primijeniti novi Oracle Traffic Director sa izrazito brojnim naprednim funkcijama.

10. LITERATURA

1. B. Cho, Planning Number Of Instance And Thread In Web Application Server, <http://www.slideshare.net/Byungwook/planning-number-of-instance-and-thread-in-web-application-server>
2. Configuring and Managing WebLogic Server, Overview of WebLogic Server Domains, http://docs.oracle.com/cd/E13222_01/wls/docs81/adminguide/overview_domain.html
3. Using WebLogic Server Clusters, Introduction to WebLogic Server Clustering, http://docs.oracle.com/cd/E13222_01/wls/docs81/cluster/overview.html
4. T. Bourke, Server Load Balancing, O'Reilly & Associates, 2001
5. C. Koppurapu, Load Balancing Servers, Firewalls and Caches, John Wiley, 2002
6. Network Load Balancing, [http://technet.microsoft.com/en-us/library/cc732855\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc732855(v=ws.10).aspx)

7. Linux Network Load Balancing, <http://lnlb.sourceforge.net/>
8. Baranovskis, WebLogic Load Balancing for Oracle ADF Applications, <http://andrejusb.blogspot.com/2009/04/weblogic-load-balancing-for-oracle-adf.html>
9. M. Negi, Load Balancing in Weblogic Server, <http://weblogicserveradministration.blogspot.com/2010/10/load-balancing-in-weblogic-server.html>
10. Oracle 9iAS Web Cache, Administration and Deployment Guide, Release 2 (9.0.2), 2002
11. Configuring Oracle Web Cache for Session Binding and Load Balancing, http://www.oracle.com/webfolder/technetwork/tutorials/obe/fmw/web_cache/11g/r1/session_bind/session_bind.htm
12. E. Biemond, High Availability Load Balancer for Weblogic Cluster, <http://biemond.blogspot.com/2010/04/high-availability-load-balancer-for.html>
13. Oracle Traffic Director, <http://www.oracle.com/us/products/middleware/application-server/oracle-traffic-director-ds-1389582.pdf>