

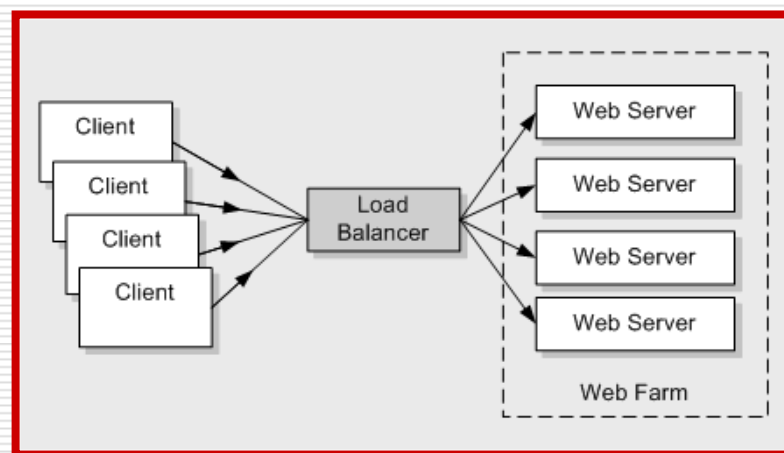
# **METODE BALANSIRANJA OPTEREĆENJA (LOAD BALANCINGA) ZA APLIKACIJSKE SERVERE**

---

**Dubravko Miljković**  
**Hrvatska elektroprivreda**  
**Zagreb, Vukovarska 37**

# ZAŠTO LOAD BALANCING

- ❑ Osiguranje potrebne raspoloživosti, kapaciteta i skalabilnosti
- ❑ Korištenje farmi aplikacijskih servera
- ❑ Load balancer presreće mrežni promet i usmjerava ga prema različitim serverima



# METODE BALANSIRANJA PROMETA

---

- Round Robin DNS
- Hardware Load Balancing
- Software Load Balancing
- Windows NLB – TCP/IP

# Round Robin DNS

- Najjednostavniji način distribucije opterećenja
- DNS server odgovara na zahtjev sa više IP adresa servera
  - ; zone file fragment
  - IN MX 10 hepweb
  - ....
  - hepweb IN A 192.168.0.4
  - IN A 192.168.0.5
  - IN A 192.168.0.6
- Na uzastopne upite DNS rotira IP adrese:
  - 192.168.0.4, 192.168.0.5, 192.168.0.6
  - 192.168.0.5, 192.168.0.6 ,192.168.0.4
  - 192.168.0.6 ,192.168.0.4, 192.168.0.5

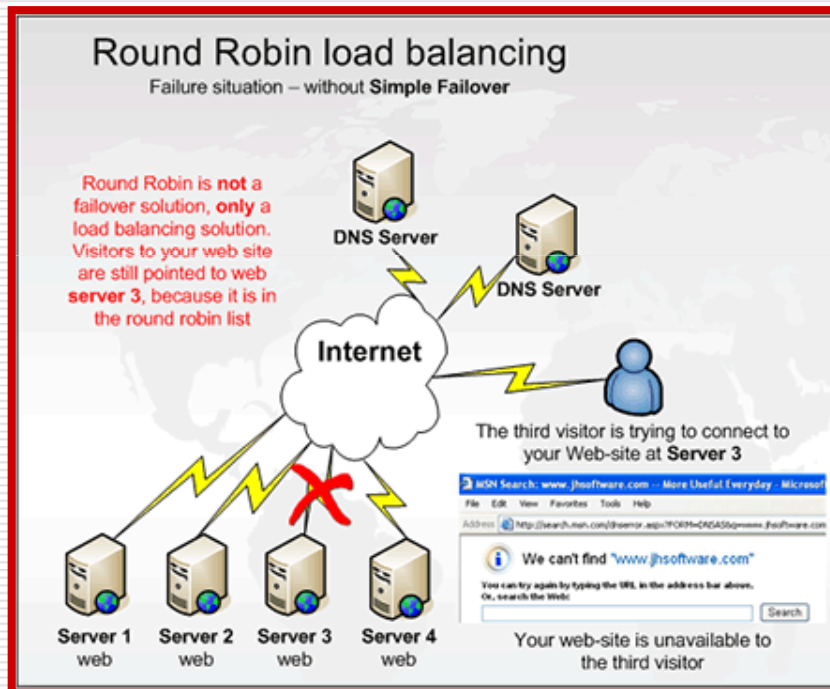
# Round Robin DNS

---

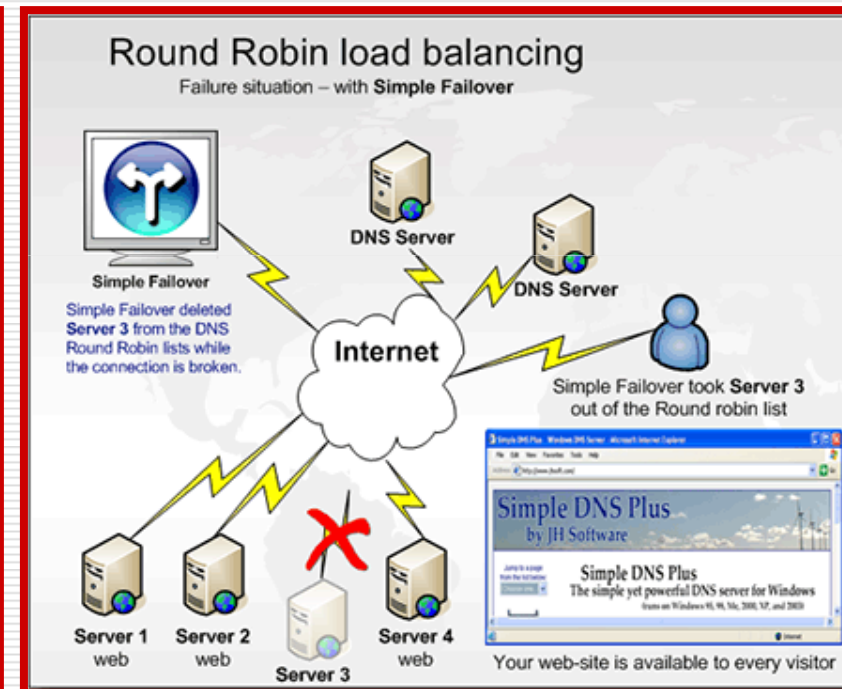
- Metoda pati od ozbiljnih nedostataka
  - Na DNS serveru i/ili klijentu obično je uključena subnet prioritizacija (subnet prioritization) koja u slučaju korištenja servera na različitim mrežama uređuje redoslijed IP adresa
  - DNS serveri i klijenti obično koriste cache - uzastopni zahtjevi ne moraju bit raspoređeni na različite servere
  - Round Robin DNS ne vodi računa o stvarnoj raspoloživosti servera koji osigurava uslugu klijenta (ne uključuje fault-tolerance)

# Round Robin DNS

## Simple Failover



Round robin i ispad servera



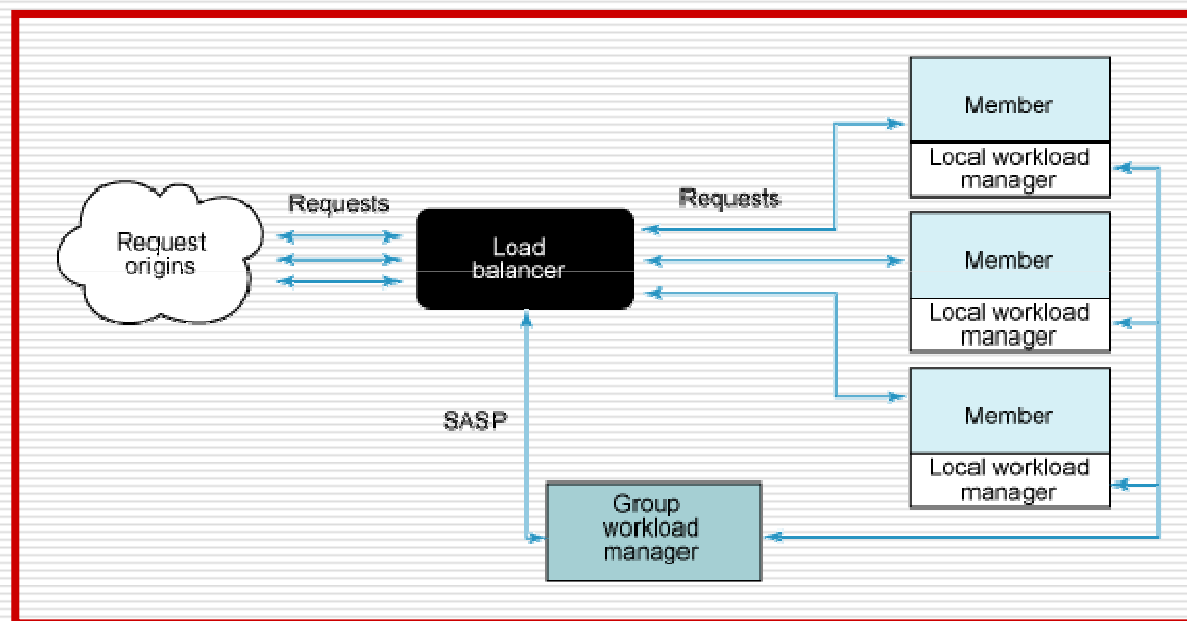
Round robin sa SimpleFailover

# Hardware Load Balancing

---

- ❑ Hardware load balanceri (Hardware Load-balancing Device HLD) obično su integrirani unutar modernih router-a (layer 4-7 router)
- ❑ Mogu opsluživati izuzetno velike promete
- ❑ Uobičajene izvedbe
  - programska rješenja na posebnom brzom hardware-u (Application-Specific Integrated Circuit - ASIC)
  - zasebne kombinacije custom hardware i software-a na nekoj varijanti LINUX-a ili custom OS-a
- ❑ DSR (Direct Server Return)
  - odgovor iz servera zaobilazi load balancer
- ❑ Postoje redundante (dual core) izvedbe čime se uklanja load balancer kao potencijalni single point of failure
- ❑ Mogućnost server feedbacka-a (agent na serveru)

# Hardware Load Balancing



**Hardware load balancer koji uzima u obzir opterećenje pojedinih servera (agenti su instalirani na serverima)**



# Software Load Balancing

---

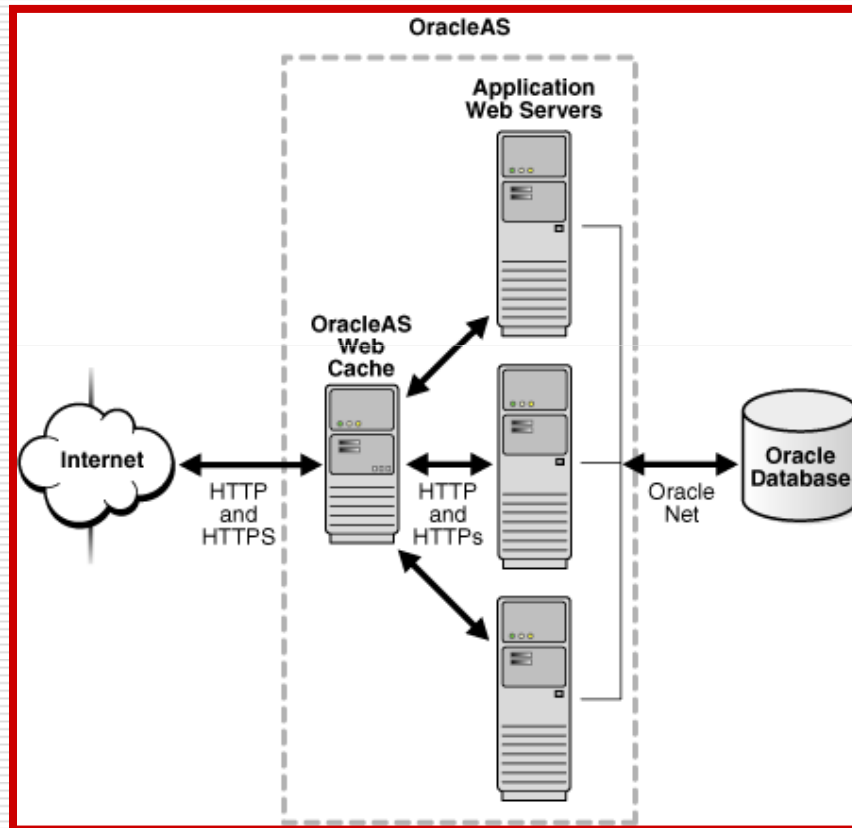
- Load balancing je tradicionalno poistovjećivan sa posebnim hardware-om
- Load balancing može se realizirati i kao programsko rješenja na konvencionalnim serverima
  - OracleAS Web Cache arhitektura
  - Programski load balancer (load dispatcher) na konvencionalnom serveru
  - HTTP redirector
- Windows NLB (Network Load Balancing)

# Software Load Balancing

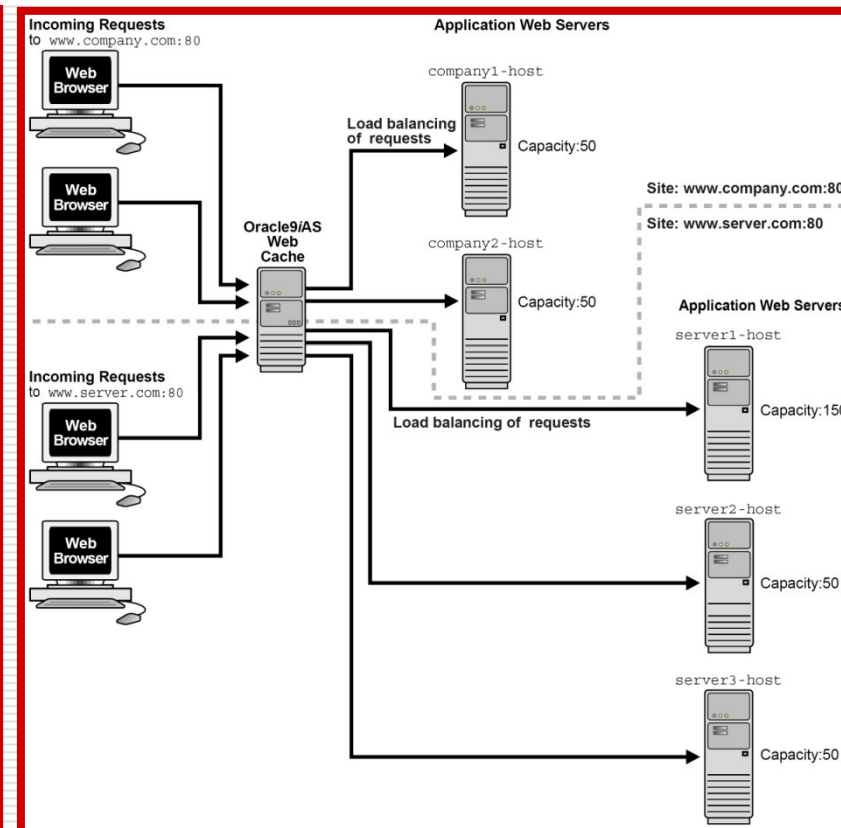
---

- OracleAS Web Cache arhitektura
  - omogućuje cache-ing statičkih web stranica, nažalost ne i servleta
  - omogućuje jednostavnu formu load balancinga
  - kod konfiguriranja navedu se svi serveri u farmi
  - ostaje problem jedne pristupne točke za korisnike (single point of failure)

# Software Load Balancing



Oracle web cache load balancer



Oracle web cache load balancer  
s primjenom težinskih faktora

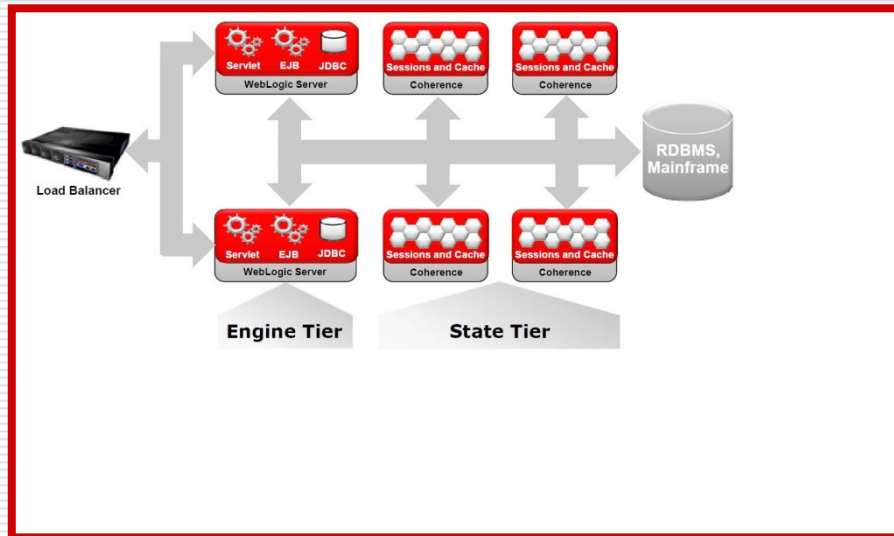
# Software Load Balancing

---

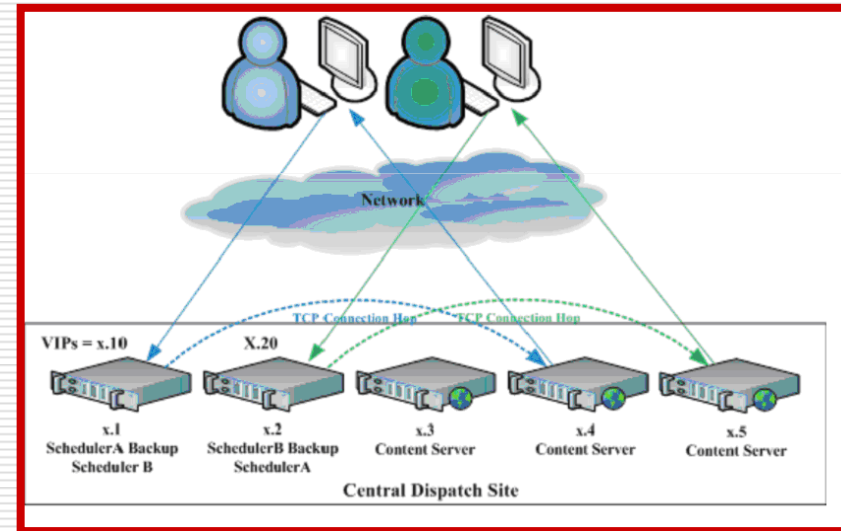
- Programski load balancer (load dispatcher) na konvencionalnom serveru
  - Programski paket koji se odvija na posebnom serveru
  - Izrazito velika fleksibilnost kod definiranja politike raspoređivanja prometa
  - Neka load balancing pravila:
    - round robin
    - weighted round robin
    - perceptive (predviđa najprikladniji nod kombinirajući broj konekcija i vrijeme odziva)
    - least connections (nod s najmanje konekcija)
    - weighted least connections
    - fastest response time (nod s najbržim vremenom odziva)
    - random node itd
  - Za visoku raspoloživost potrebno je koristiti dva balancera (u active-standby ili active-active konfiguraciji)

# Software Load Balancing

Programski paket na posebnom serveru



Software load balancer ispred Weblogic servera



Software load balancer za visoku raspoloživost (primarni balacer i backup)

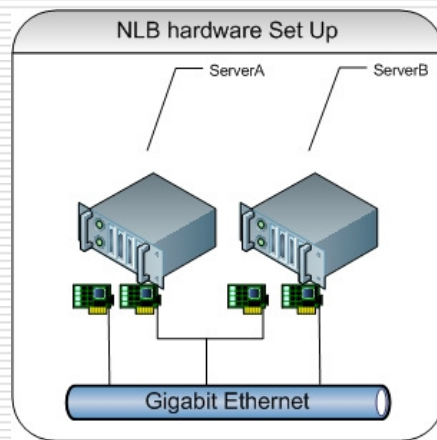
# Software Load Balancing

---

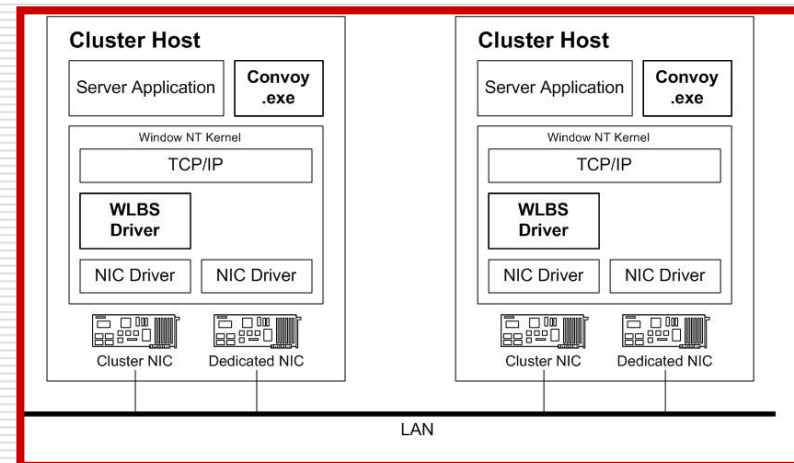
## □ HTTP redirektor

- Ograničen samo na HTTP promet
- HTTP zahtjev preusmjerava sa HTTP redirect komandom direktno na pojedini aplikacijski server
- Najjednostavnija rješenja u obliku Perl skripti (random node, round robin, IP address based - geotargeting)

# Windows NLB (Network Load Balancing)



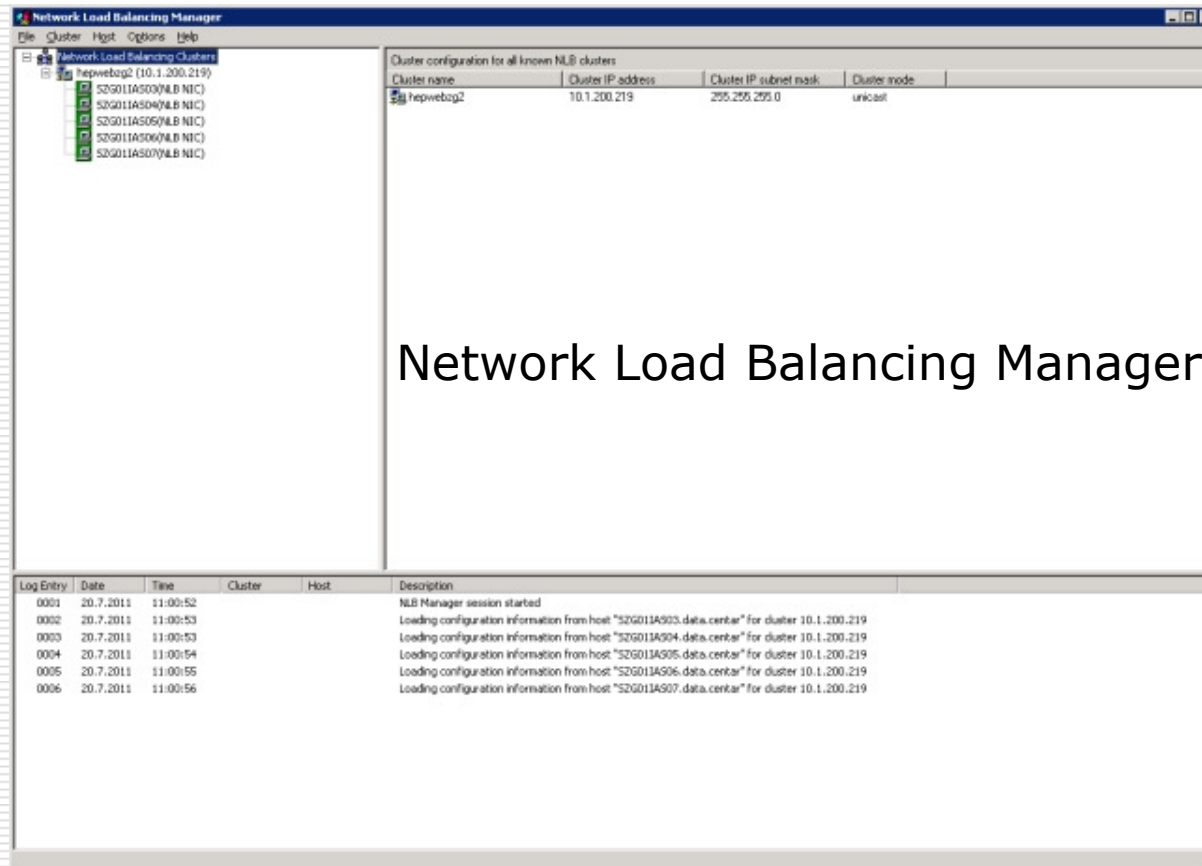
NLB Setup



Windows NLB Stack

- ❑ kartice su na mreži kao na paralelnom bus-u
- ❑ nodovi su ravnopravni
- ❑ ne postoji interconnect
- ❑ nema vanjskog load balancera
- ❑ eliminiran single point of failure

# Windows NLB (Network Load Balancing)

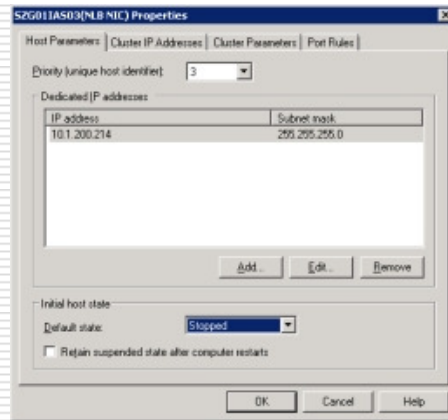


Network Load Balancing Manager

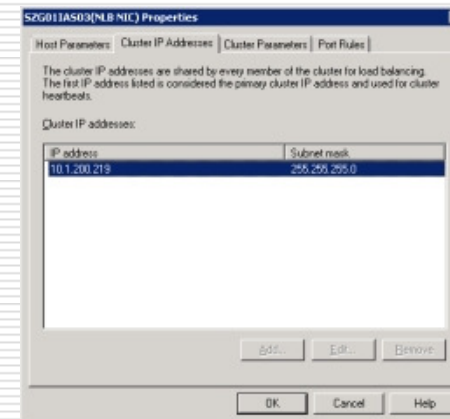
Dodavanje novih nodova u NLB cluster (Windows 2003 i 2008)



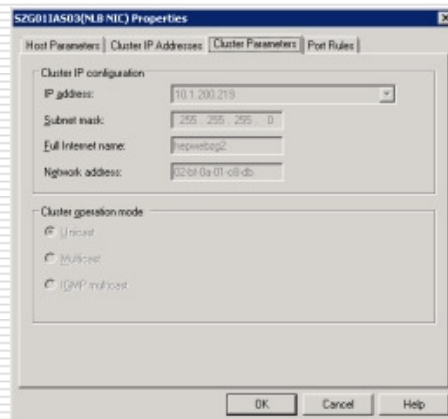
# Windows NLB (Network Load Balancing)



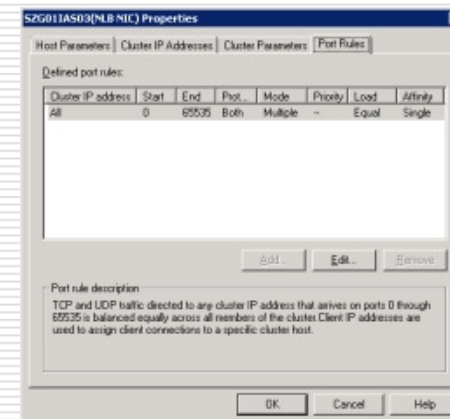
Host parameters – default state stopped



Cluster IP address (virtual address)



Cluster parameters - unicast



Port rules – Load Equal, Affinity Single

# Windows NLB (Network Load Balancing)

---

- ❑ U svojim počecima zamišljen da radi preko hub-a (ne preko switch-a)
- ❑ Danas direktno spajanjem NLB mrežnih kartica na switch
- ❑ Potrebno na mrežnoj NLB kartici osposobiti IP Forwarding
- ❑ Windows 2008
  - netsh interface ipv4 set int "[name of the NIC]" forwarding=enabled  
Napomena: [name of the NIC]=NLB
- ❑ Windows 2000, 2003
  - **HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters**
  - Set the following registry values:  
  
Value Name: **IPEnableRouter**  
Value type: **REG\_DWORD**  
Value Data: **1**

# PROVJERA ISPRAVNOSTI NODA (APLIKACIJSKOG SERVERA)

---

- ❑ Balansiranje prometa ima smisla samo ako je server ispravan
- ❑ Utvrđivanje ispravnosti noda i aplikacije (application health monitoring)
- ❑ Poželjna je dvorazinska provjera
  - Brza (osnovna) - jednostavnija provjera kritičnih komponenti (npr. Apache s HTTP GET)
  - Detaljna (application specific HTTP GET s content checks, reports servers, provjera funkcionalnost servisa – ne samo provjera da li je servis podignut)

# PROVJERA ISPRAVNOSTI NODA (APLIKACIJSKOG SERVERA)

---

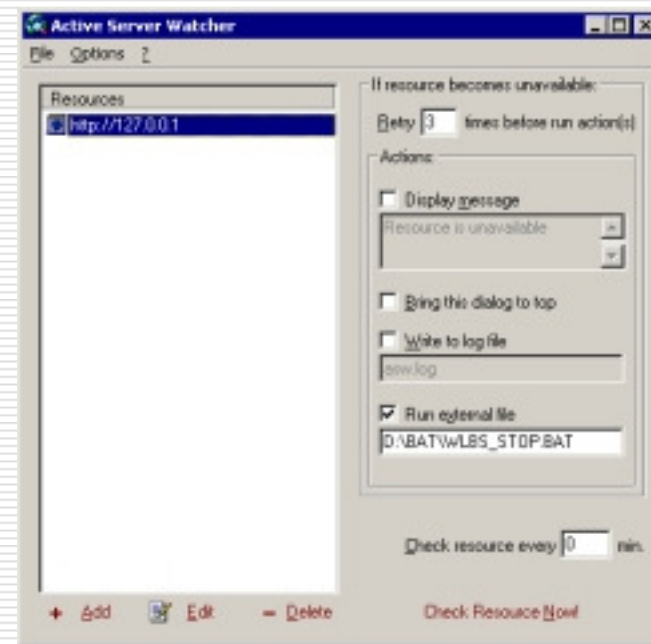
- ❑ Za postizanje visoke raspoloživosti load balancing treba detektirati neispravan server i automatski ga izostaviti iz raspoređivanja prometa
- ❑ Za ostvarivanje vrlo visoke raspoloživosti (99,999 %) potrebno reagirati jako brzo u slučaju kvara noda

Raspoloživost	Godišnje vrijeme van pogona	Rješenje (okvirno)
99%	87,6 sati	jedan server
99,9%	8,76 sati	jedan server, vrlo dobra konfiguracija i održavanje
99,99%	52,5 minuta	cluster
99,999%	5,25 minuta	cluster, geografski disperzirani cluster

# PROVJERA ISPRAVNOSTI NODA (APLIKACIJSKOG SERVERA)

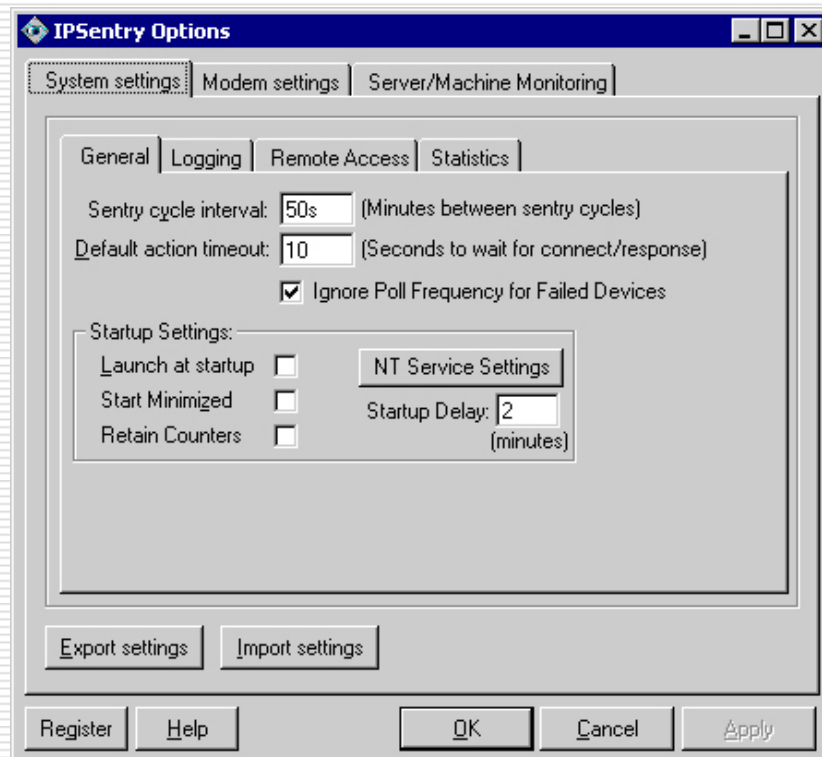
---

- ❑ Brza (osnovna) provjera ponavlja se svakih nekoliko sekundi
- ❑ Budući da je vrlo česta, ne smije puno opterećivati CPU
- ❑ Primjer: aplikacija Active Server Watcher

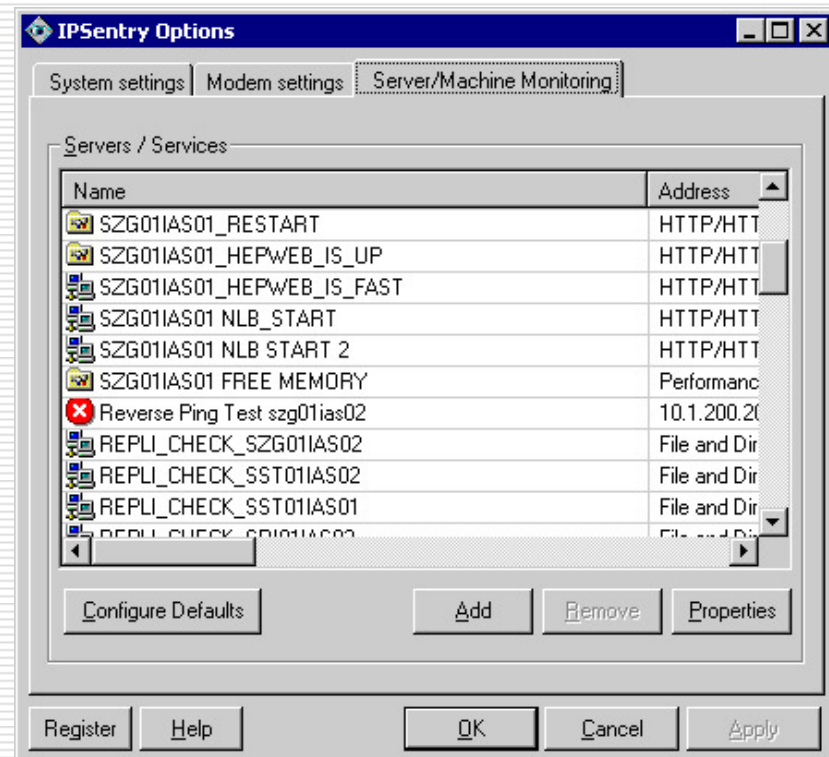


# PROVJERA ISPRAVNOSTI NODA (APLIKACIJSKOG SERVERA)

---



IP Sentry za detaljne provjere I



IP Sentry za detaljne provjere II

# PERZISTENCIJA, ASIMETRIČNA OPTEREĆENJA

---

- Perzistencija
- Asimetrična opterećenja
- Problem IP affinity-a  
(i perzistencije uopće)
- Ponašanje nakon restarta noda
- Opterećenje preostalih nodova

# Perzistencija

---

- Tijekom sesije korisnik komunicira uvijek s istim serverom u farmi
- Statefull application
- Sticky session
- Primjena cookie-a
  - cookie pohranjen u browseru klijenta
- IP zasnovana
  - Temelji se na IP adresi klijenta
  - Problem kad klijenti dolaze preko proxy servera



# Asimetrična opterećenja

---

- ❑ Serveri u farmi nemaju uvijek isti kapacitet
- ❑ Mogućnost rada sa serverima različitih kapaciteta
  - ❑ weighting faktor kod round robin
  - ❑ weighting faktor u hardware load balanceru
  - ❑ weighting faktor u software load balanceru
  - ❑ weighting faktor u Windows NLB

# Problem IP affinity-a (i perzistencije uopće)

---

- IP affinity je dobro rješenje za ostvarenje perzistentnih session-a, ali
  - koliko vremenski dugo affinity treba vrijediti (affinity window)
  - ne treba dovijeka zalijepiti korisnika za pojedini aplikacijski server
- Potrebno je ponekad nakratko resetirati affinity (npr. noću)

# Ponašanje nakon restarta noda

---

- ❑ Kod ispada noda obično se obavlja restart servera
- ❑ Za vrijeme podizanja noda korisnici će se spajati na preostale aktivne nodove
- ❑ Mnogi korisnici s kojima je bila prekinuta komunikacija već će započeti nove perzistentne sesije
- ❑ Restartani će nod biti inicijalno slabo opterećen (funkcija vremena oporavka i pristizanja korisnika) sve dok ne stignu potpuno novi korisnici

# Opterećenje preostalih nodova

---

- ❑ Ispad jednog noda može izazvati domino efekt i porušiti sve preostale nodove
- ❑ U slučaju ispada jednog noda, preostali nodovi trebaju imati dovoljan ukupan kapacitet da usluže korisnike
- ❑ Load balancing cluster treba dimenzionirati s jednim serverom viška (redundancija  $N+1$ )

# GEOGRAFSKI DISPERZIRANI SUSTAV

---

- Pogodan za veliku organizaciju s regionalnim centrima
- Osiguranje visoke raspoloživosti
- U normalnom radu korisnici se raspoređuju na najbliži (regionalni) cluster
- Raspodjela opterećenja nije ograničena samo na regionalni cluster – fleksibilnost
- U abnormalnim situacijama preostali aktivni regionalni clusteri omogućuju neprekidan rad

# GEOGRAFSKI DISPERZIRANI SUSTAV

---

- ❑ Primjer geografski disperziranog sustava koji se koristi u HEP-u
- ❑ Uključuje četiri iAS clustera u četiri regionalna centra (Zagreb, Rijeka, Split i Osijek)
- ❑ Regionalni centri posjeduju NLB cluster koji može raditi samostalno
- ❑ Sustavu se pristupa preko zajedničke web adrese hepweb, bez obzira na lokaciju korisnika

# GEOGRAFSKI DISPERZIRANI SUSTAV



- Koristi se geotargeting redirekcija
  - DNS subnet prioritizacija
  - Perl redirektori
- Za zajedničko krovno ime hepweb postoje višestruki (konkretno četverostruki) A rekordi u DNS-ima unutar HEP-a
- Perl redirektori na regionalnim clusterima:
  - hepwebos 10.193.64.116
  - hepwebri 10.65.64.206
  - hepwebst 10.129.64.116
  - hepwebzg 10.1.200.206

# GEOGRAFSKI DISPERZIRANI SUSTAV



---

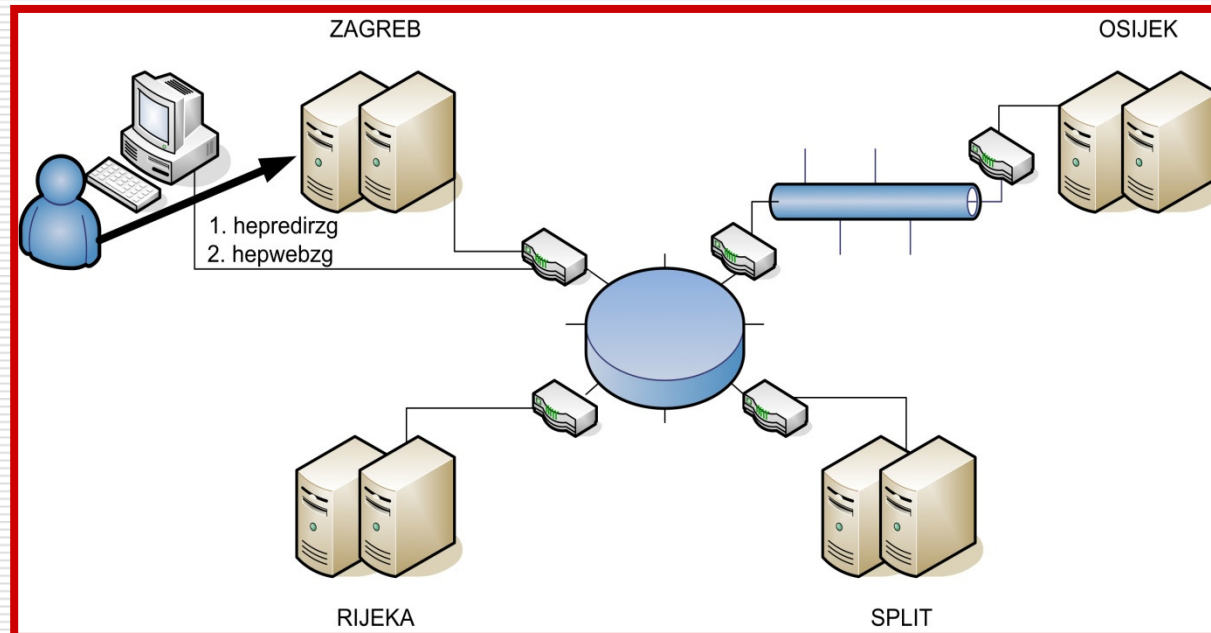
- Perl redirektor
  - Obavlja drugu razinu redirekcije (prva razina je DNS subnet prioritizacija)
  - Usmjerava klijenta na sam iAS cluster
  - Jednostavna skripta koja obavlja geotargeting redirekciju, temelji se na adresnim područjima korisnika iz pojedinih regija
  - Potvrđuje odluku donesenenu na prvoj razini
  - Vodi računa o raspoloživosti i opterećenjima svih regionalnih iAS clustera – clusteri oglašavaju svoje opterećenje



# GEOGRAFSKI DISPERZIRANI SUSTAV

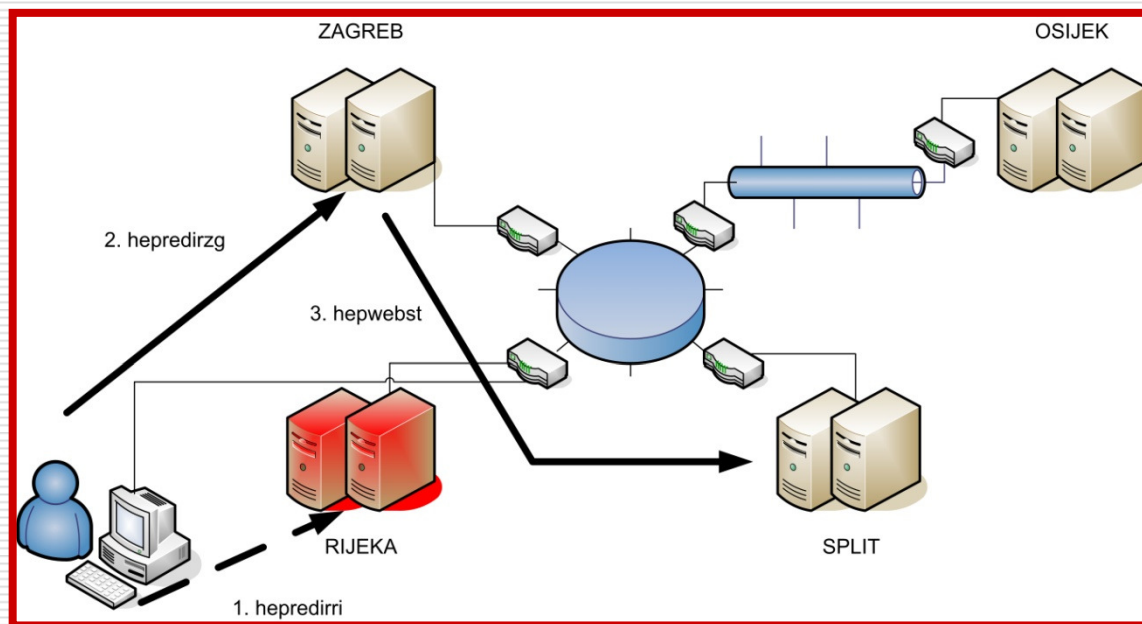
- ❑ Na upit hepweb DNS-ovi (uključena subnet prioritizacija) odgovaraju sortiranim adresama Perl redirektora
- ❑ Za korisnike iz raznih regija to izgleda ovako:
  - Regija Osijek  
10.193.64.116,      10.129.64.116,      10.65.64.206,      10.1.200.206
  - Regija Rijeka  
10.65.64.206,      10.1.200.206,      10.193.64.116,      10.129.64.116
  - Regija Split  
10.129.64.116,      10.193.64.116,      10.1.200.206,      10.65.64.206
  - Regija Zagreb  
10.1.200.206,      10.65.64.206,      10.129.64.116,      10.193.64.116
- ❑ Korisnici na prvom mjestu uvijek dobiju adresu svog regionalnog Perl redirektora
- ❑ Ukoliko je prva adresa nedostupna, klijentov IE browser prelazi nakon 20 sekundi na sljedeću adresu

# GEOGRAFSKI DISPERZIRANI SUSTAV



- ❑ Normalan rad geografski disperziranog sustava
- ❑ Korisnik se usmjerava na najbliži (regionalni) iAS NLB cluster
- ❑ U slučaju preopterećenja promet preuzimaju drugi iAS NLB clusteri

# GEOGRAFSKI DISPERZIRANI SUSTAV



- ❑ Ispad iAS NLB clustera Rijeka (cluster je nedostupan)
- ❑ Korisnik se automatski preusmjerava (nakon 20s) na drugi iAS NLB cluster (u ovom primjeru Zagreb)
- ❑ Pri izboru drugog iAS NLB clustera uzimaju se u obzir svi regionalni clusteri koji imaju višak kapaciteta (iAS NLB clusteri oglašavaju opterećenje)

# ZAKLJUČAK

---

- ❑ Većina metoda zahtjeva dodatni hardware
- ❑ Kod visoke raspoloživosti (high availability) hardware treba biti udvojen – inače imamo samo load balancing
- ❑ Hardware load balancing
  - često integrirani u router-e
  - izuzetne brzine (da li nam stvarno trebaju?)
- ❑ Software
  - programska rješenja na posebnom serveru
  - izrazita fleksibilnost, pravila balansiranja
- ❑ Microsoft NLB je jednostavan za konfiguriranje
  - ne zahtijeva dodatni hardware i nema single point of failure
  - po performansama je više nego dovoljan za većinu primjena
- ❑ Geografski disperzirani cluster može se realizirati dvorazinskim usmjeravanjem prometa
  - DNS subnet prioritization
  - PERL geotargeting skripte

# HVALA!

---