Load Balancing

**Einleitung**

Load Balancing ist kein neues Konzept im Server- und Netzwerkbereich. Viele Produkte, wie z.B. Router die Netzwerkverkehr über verschiedene Netzwerk Ressourcen zum gleichen Ziel verteilen, können unterschiedliche Arten des Load Balancing durchführen. Im Gegensatz zum Router wird bei Server Load Balancing der Netzwerkverkehr über verschiedene Server Ressourcen verteilt. Anfangs wurden Load Balancer als reine Lastenverteiler verwendet, doch heute sind Load Balancer schon so weit entwickelt, dass sie zusätzliche Funktionen wie Healthchecks, Optimierung von Datenflüssen, etc. zur Verfügung stellen.

Im Webhostingbereich wird Load Balancing typischerweise für die Verteilung von http-Verkehr auf mehrere Server (Nodes), die als Web Front-End agieren, eingesetzt. Der Load Balancer beurteilt die Antwortzeiten und Auslastung der einzelnen Server und verteilt basierend auf dieser Beurteilung die Anfragen. Durch die Verteilung des Dienstes auf mehrere Server schützt man sich zusätzlich vor Hardwareausfall, da bei Ausfall eines Server Nodes der Netzwerkverkehr einfach über andere Nodes erfolgen kann. Das primäre Ziel ist es, den Netzwerkverkehr bei hoher Nachfrage auf alle vorhandenen Nodes aufzuteilen, um die bestmögliche Performance zu gewährleisten. Der User weiß normalerweise nichts über vorhandene Backend- bzw. Backupserver, für ihn scheint es so als ob nur ein Server hinter einem Dienst steht.

**Wann und wozu verwendet man Load Balancing?**

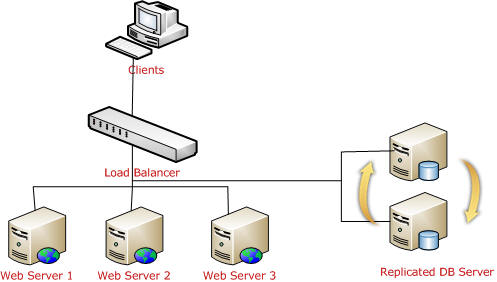
* Failover und Redundanz

Ein wesentliches Einsatzgebiet des Load Balancing ist die Ausfallsicherheit und die daraus resultierenden erhöhte Uptime (Betriebszeit). Durch Verwendung von mehreren identischen Nodes kann der Netzwerkverkehr im Falle eines Hardware oder Softwarefehlers umverteilt werden und die Website, Dienst, etc. bleibt verfügbar.

* Wachsende Nachfrage und Verfügbarkeit

Falls die Website, Dienst, etc. so beliebt wird, dass ein einziger Server die Masse an Anfragen nicht in absehbarer Zeit abarbeiten kann (was zu z.B. lange Wartezeit bei Seitenaufbau führt), sollte man ebenfalls Load Balancing einsetzten. Dadurch können die Anfragen auf mehrere Server verteilt werden und so ist die Verfügbarkeit wieder gegeben.

**Beispiel Firmenwebsite**

****

<http://www.codeproject.com/Articles/32545/Exploring-Session-in-ASP-Net#38>

Anhand dieses kleinen Beispiels einer Firmenwebsite möchte ich die zwei oben genannten Haupteinsatzgründe für Load Balancing veranschaulichen.

Mögliche Probleme bei einer Website: zu hohe Anzahl an Anfragen für Webserver (falls die Seite nur von einem Webserver gehostet wird), Datenbank- bzw. Webserver fällt aus

Lösung durch Load Balancing: Einsatz von mehreren Servern. Um für eine hohe Anzahl an Anfragen gerüstet zu sein, muss die Website über mehrere Webserver verfügbar sein. Da eine Domain nur einem physikalischen Server zugewiesen werden kann, wird der Load Balancer zwischen dem Internet und den Webservern geschaltet und bekommt die öffentliche Domain zugewiesen. Wenn mehrere User die Website abrufen, beurteilt der Load Balancer die Antwortzeiten und Auslastung der einzelnen Webserver und verteilt die Anfragen so, dass die bestmögliche Performance erreicht wird. Außerdem wird so auch die Ausfallsicherheit der Website gewährleistet, da die Website jetzt über mehrere Webserver gehostet wird. Zusätzlich wird in diesem Beispiel ein Backupserver der Datenbank erstellt. Falls der Datenbankserver ausfällt oder ausgetauscht werden muss, springt einfach der Backupserver ein.

**Notwendigkeit von Load Balancing**

Das primäre Ziel von Load Balancing ist es den Netzwerkverkehr bei hoher Nachfrage auf alle vorhandenen Nodes aufzuteilen, um die bestmögliche Performance zu gewährleisten. Netzwerke und Server sind die Hauptgründe, warum Load Balancing notwendig ist. In der heutigen Zeit, ist das Internet ein wichtiger Bestandteil für jedes Unternehmens ist (um Netzwerk zwischen Firma, Lieferanten, Kunden, etc. zu erzeugen), bei dem es auch um viel Geld geht. Firmen, vor allem im e-commerce Bereich (elektronischer Handel), können es sich nicht leisten, wenn ihr Netzwerk entweder sehr langsam ist bzw. ausfällt.

Um z.B. eine Website für elektronischen Handel zu erstellen, muss man Faktoren wie Server, Switches, Firewalls, etc. berücksichtigen. Durch das Verwenden von mehreren Servern für so eine Website entstehen aber Herausforderungen in den Bereichen Skalierbarkeit, Verwaltbarkeit und Verfügbarkeit. Load Balancing löst zusätzlich viele dieser Probleme.

**Skalierbarkeit**

Skalierbarkeit ist kein neues Problem. Früher ist eine Applikation auf einem Server gelaufen. Falls dieser Server für die Applikation nicht mehr ausreichend war, wurde dieser entweder verbessert oder durch einen performanteren Server ersetzt. Durch Load Balancing ist es jedoch möglich, die Applikation auf mehreren Servern zu verteilen. Durch die mehreren Server ist es möglich, sämtliche Anfragen zu verteilen um bestmögliche Performance zu gewährleisten

**Verwaltbarkeit**

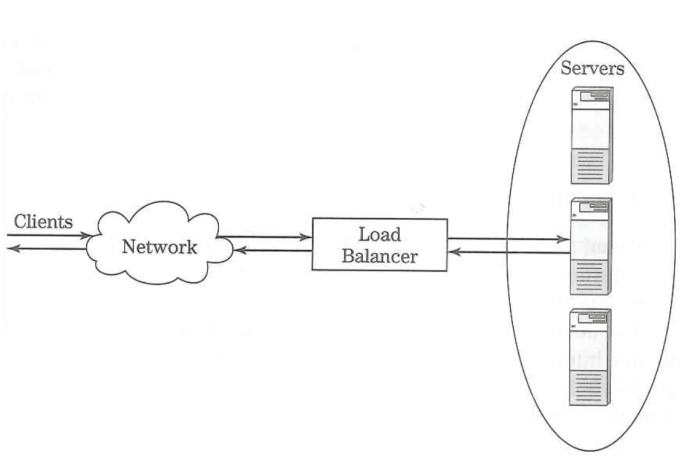
Wenn die Serverhardware verbessert werden muss oder das Betriebssystem aktualisiertet wird, muss der Server abgeschaltet werden. Man kann das natürlich in der Zeit machen, in der am wenigsten Bedarf an der Website/Dienst besteht (vermutlich in der Nacht), jedoch hat man trotzdem eine Ausfallzeit (Downtime). Doch manche Firmen können sich überhaupt keine Ausfallzeit leisten oder haben konstanten Bedarf an ihrem Service. Ein Load Balancer kann Server, die für Wartungszwecke offline gestellt werden müssen, ohne Downtime ausschalten. Das wird gewährleistet, in dem der Load Balancer keine weiteren Anfragen an diesen Server weiterleitet und wartet bis sämtliche bestehende Anfragen abgearbeitet wurden. Dann kann der Server offline gestellt werden. Sämtliche Anfragen werden auf die anderen Server weitergeleitet. Zusätzlich helfen Load Balancer beim Verwalten von Content Managementsystem, da es bei solchen Systemen ebenfalls sein kann, das ein Server nicht ausreicht.

**Verfügbarkeit**

Der Load Balancer kontrolliert kontinuierlich die Verfügbarkeit aller Server und laufenden Applikationen. Wenn der Healthcheck einer dieser Server oder Applikationen scheitert, sendet der Load Balancer keine Anfragen mehr zu diesem Server.

**Load Balancing Applikationen**

„With the advent of the Internet, the network now occupies center stage. As the Internet connects the world and the intranet becomes the operational backbone for businesses, the IT infrastructure can be thought of as two types of equipment: computers that function as a client and/or a server, and switches/routers that connect the The Network Environment computers. Conceptually, load balancers are the bridge between the servers and the network, as shown in the graphic below. On one hand, load balancers understand many higher−layer protocols, so they can communicate with servers intelligently. On the other, load balancers understand networking protocols, so they can integrate with networks effectively“



Load Balancer haben zumindest vier Applikationen:

* Server Load Balancing
* Global Server Load Balancing
* Firewall Load Balancing
* Transparent Cache Switching

Server Load Balancing: Verteilung der Anfragen auf mehrere Server um die Auslastung gleichmäßig zu verteilen und um sich vor Serverausfall zu schützen

Global Server Load Balancing: Verteilung der User zu verschiedenen Data Center, die aus Serverfarmen bestehen für schnelleren Rückmeldung und als Schutz vor Data Center Ausfällen

Firewall Load Balancing: Verteilung des Netzwerkverkehrs auf mehrere Firewalls um vor Firewallausfall geschützt zu sein

Transparent Cache Switching: lenkt den Netzwerkverkehr zu Caches um den Antwortzeit zu verringern

**Load Balancing Produkte**

Load Balancing Produkte können in 3 Kategorien unterteilt werden: Softwareprodukte, Switches und Geräte

Software Load Balancing Produkte laufen direkt auf den Load-Balancer Servern. Die Software führen die Algorithmen aus, welche für die Lastverteilung zuständig sind. Beispiele für Software Load Balancing Produkte sind z.B. Incapsula, NGINIX und BalanceNG

http://www.capterra.com/load-balancing-software/

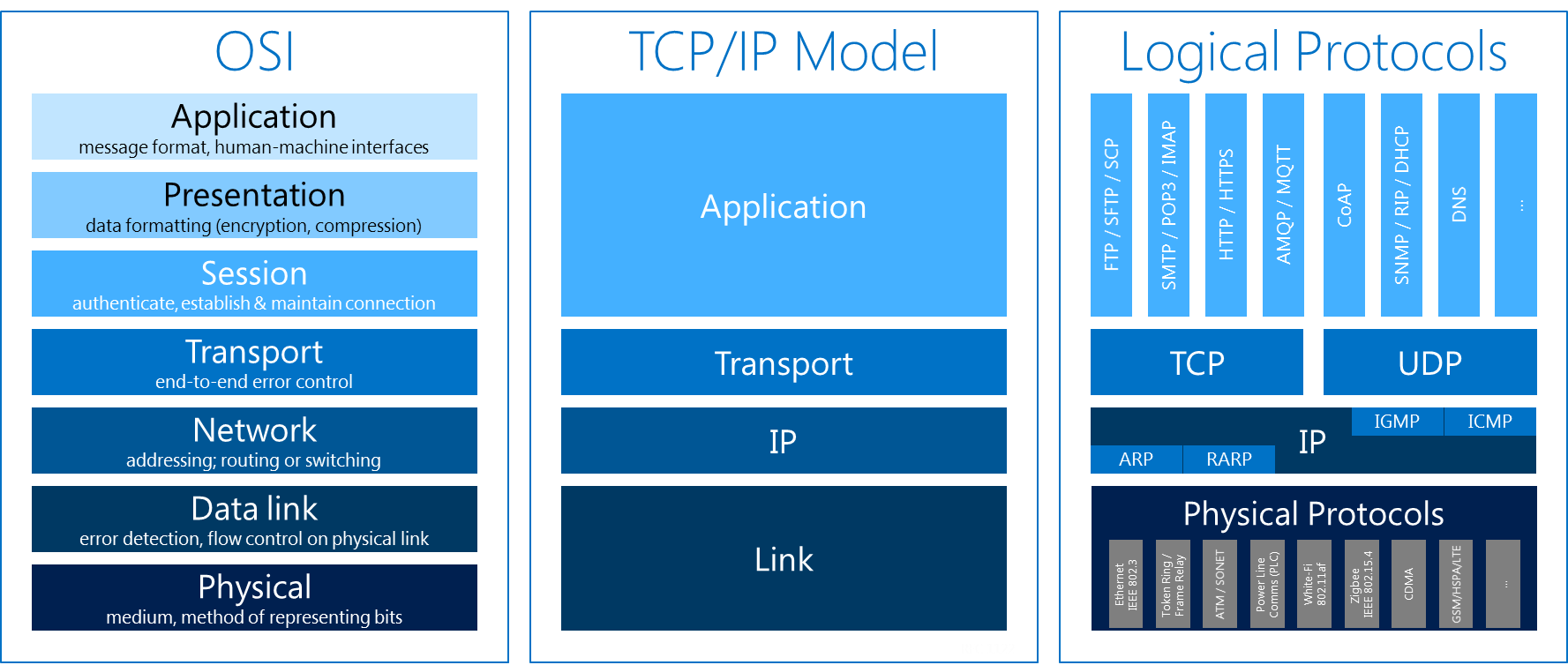
Switches haben ihre normalen Funktionalitäten auf OSI Layer 2/3, doch können zusätzlich Load Balancen auf Layer 4-7. Das wird durch zusätzliche Hardware oder Software erreicht.

Geräte sind Blackbox Produkte, welche die notwendige Hardware und Software für Web Switching besitzen. Das könnte z.B. ein einfacher Computer oder Server mit speziellem Betriebssystem und spezieller Software sein

**Networking Grundlagen**

**OSI Schichtenmodell**

Das OSI Schichtenmodell ist der Standard, der definiert wie unterschiedliche Geräte oder Computer miteinander kommunizieren können. Jeder Layer kann mit demselben Layer (Peer) eines anderen Rechners kommunizieren. Zusätzlich kann der Layer mit den Layern direkt über und unter ihm Informationen austauschen.



<http://jimwilsonblog.com/?tag=osi>

Layer-2 Load Balancing (auch als Link- oder Portaggregation bezeichnet) verbindet zwei oder mehrere Links zu einer leistungsfähigeren logischen Verknüpfung. Aggregierte Verknüpfungen stellen zusätzlich Redundanz und Fehlertoleranz, wenn jeder der einzelnen aggregierten Links einen unterschiedlichen physikalischen Pfad folgt.

Layer 4 Load Balancing ist für die Verteilung der Anfragen zu den Servern auf Transport Layer Ebene zuständig, dazu gehören die Transportprotokolle TCP, UDP und SCTP. Z.B. ein normaler Router sendet eintreffende Pakete einfach an die geeignete IP Adresse, doch ein Layer 4 Router verteilt die Anfragen der Clients, die nur die IP-Adresse eines Services wissen, an alle Server, auf denen der Service läuft, bestmöglich.

Layer 7 Load Balancing (auch als Application-Level Load Balancing bezeichnet) ist die Lastenverteilung bezogen auf den Content-Typ (z.B. Scripts wie HTML, CSS, etc.) der Client-Anfrage. Geräte, die Layer 7 Load Balancing betreiben können, heißen Application Delivery Controllers (ADC). Da der Load Balancing Server weiß, was der Client für einen Content-Typ haben will, kann er abwägen, welcher der vorhanden Server am besten mit dieser Anfrage umgehen kann.

