

WEBSITE-BOOSTING: ALLZWECKWAFFE VARNISH

Für den Erfolg eines Online-Shops, einer Website oder eines Online-Mediums spielen bereits Sekundenbruchteile eine Rolle – dann nämlich, wenn es um die Performance geht. Der Performance-Booster Varnish hilft, den letzten Rest an Optimierungspotential aus jeder Site heraus zu kitzeln.

TEXT **DOMINIC BRANDER UND ANDRI STEINER**

Je mehr Bandbreite zur Verfügung steht und je größer die Konkurrenz ist, desto entscheidender ist die Performance von Websites und Shops für die gute User-Experience (UX). Damit ist sie oftmals das Zünglein an der Waage, schlägt sie sich doch direkt in den Verkaufszahlen nieder: Bereits eine Sekunde Verzögerung beim Laden eines Shops bringt bis zu elf Prozent weniger Besucher, 16 Prozent geringere Kundenzufriedenheit und sieben Prozent weniger Verkäufe, wie Forscher der Aberdeen Group herausfanden [1].

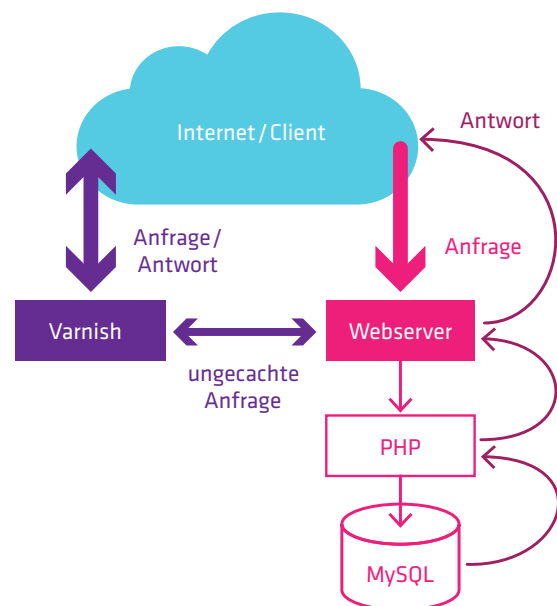
User erwarten schnelle und zuverlässige Websites, Shops und Applikationen – die Alternativen sind gerade mal einen Mausklick entfernt. „User besuchen eine Website einfach nicht, wenn ihre Ladezeit 250 Millisekunden langsamer ist als die ihrer Konkurrenz. Das ist eine Viertelsekunde, also weniger Zeit als man für ein Blinzeln braucht“, meinte Microsofts Corporate Vice President Harry Sum in der New York Times. Dazu kommt, dass Google seit 2010 die Performance einer Site in seinen Search Rankings berücksichtigt [2] und sie damit zu einem weiteren, wichtigen Kriterium der Suchmaschinen-Optimierung macht. Wer schnell viele Daten ausliefern muss, kommt um einen Performance-Optimierer wie Varnish kaum vorbei.

VARNISH – DER WEBSITE BOOSTER

Varnish ist eine Caching-Reverse-Proxy-Server-Software – also ein Web-Applikations-Beschleuniger, der sich vor jede beliebige Site oder jeden beliebigen Applikationsserver schalten lässt und HTTP nutzt. Der dänische Entwickler Poul-Henning Kamp initiierte Varnish 2005 aus Frust über die damals bestehenden Lösungen. Varnish ist Open-Source-Software und steht unter der BSD-Lizenz. Neben frei verfügbaren Modulen stehen auch kostenpflichtige Zusatzmodule, wie die Varnish Administration Console (VAC), die Varnish Custom Statistics (VCS) oder die Varnish-Paywall zur Verfügung. Für bekannte Systeme wie Drupal, Magento, TYPO3, WordPress und viele andere Systeme gibt es bereits bestehende Integrationen.

Varnish nutzt topaktuelle Features des Linux-Kernels und somit modernste Server-Ressourcen besonders effektiv. Der Einsatz der Software vermeidet es in der Regel, dass das Betriebssystem Daten im Cache und gleichzeitig auch auf der Festplatte des Servers

speichert. Der Cache legt Datenpakete im virtuellen Speicher ab und das System entscheidet hingegen autonom, welche Daten im Hauptspeicher bleiben und welche es auf die Festplatte auslagert. Inhalte lassen sich somit direkt aus dem superschnellen Arbeitsspeicher des Servers ausliefern. Langsame Datenbankzugriffe sind nur dann nötig, wenn die Site den Cache neu aufbauen muss.



Datenströme mit und ohne Varnish: Langsame Datenströme sind nur notwendig, wenn sich der Cache neu aufbauen muss.

Die Konfiguration geschieht mithilfe der Varnish Configuration Language (VCL), einer domänenspezifischen Sprache. Dank der hohen Flexibilität von VCL können Entwickler auf beliebige Situationen eingehen. Sie können ganze Business-Logiken und die damit verbunden rechenintensiven Abfragen komplett in VCL ausgelagern. Ein Beispiel dafür ist die Device-Detection, also die Geräteabhängige Ausgabe von Inhalten.

Klassischerweise fangen Sites dies über das Content-Management-System (CMS) mühselig und auf Kosten der Performance ab. Mit Varnish können Entwickler dies ohne spürbaren Performance-Verlust implementieren. Denn VCL wird in C übersetzt, in Binär-code kompiliert und als Shared Object direkt in den Varnish Cache gelinkt. Somit lässt sich die Konfiguration als nativer Code ausführen und ist selbst bei Tausenden Zeilen Konfigurationsanweisungen noch immer blitzschnell.

Doch Varnish setzt Geschwindigkeit und Stabilität nicht nur bei der eigentlichen Auslieferung der Datenströme schlüssig um. Auch das Kompilieren, Laden und Aktivieren neuer Konfigurationen erfolgt ganz ohne Neustart. Varnish Cache 2.0 führte im Oktober 2008 zusätzlich Loadbalancer-Funktionen ein, deren Logiken sich über die VCL-Policies ohne Performance-Verluste abhandeln lassen. Dabei können die Webserver identische Inhalte ausliefern oder Varnish leitet die Requests an festgelegte Server weiter. So kann Varnish selbst dann noch Daten ausliefern, wenn der eigentliche Webserver nicht mehr erreichbar ist.

Auch das Logging der Varnish-Aktionen setzt konsequent auf Geschwindigkeit. Varnish schreibt das Log nicht auf die Festplatte, sondern in den Arbeitsspeicher, denn der ist rund 4.000 mal schneller als selbst die schnellsten Solid State Disks. Das hat drei Vorteile:

- Der Log-Prozess konkurriert nie mit dem blitzschnellen Ausliefern der Daten – der Hauptaufgabe von Varnish.
- Varnish kann alle Aktionen loggen, ohne die Festplatte zu überfüllen.
- Der schnelle Log-Mechanismus erlaubt Echtzeitanalysen von Datenströmen und damit eine sofortige Reaktion auf kritische Situationen.

SACKGASSE KLASSISCHE PERFORMANCE-OPTIMIERUNG

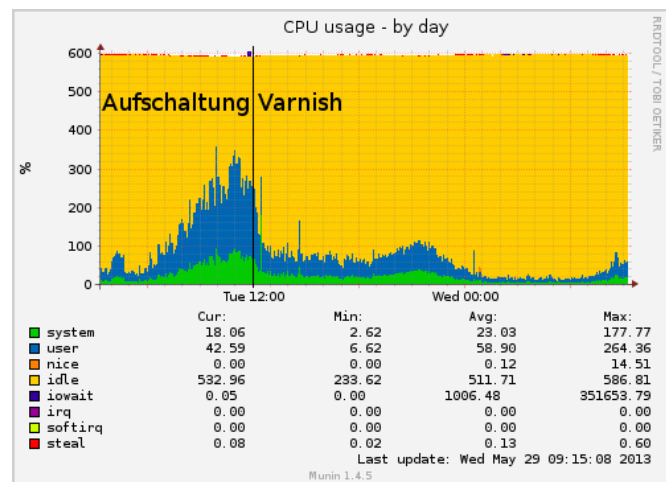
Mit der wachsenden Komplexität und Dynamik einer Weblösung steigen auch die Anforderungen an die Server-Hardware und die Caching-Mechanismen. Viele Performance-Probleme lassen sich bereits mit mehr CPU und RAM lösen. Auch eine schnelle Solid State Disk anstatt der klassischen Festplatten ist inzwischen erschwinglich geworden. Zudem fangen die mittlerweile recht ausgereiften Caching-Mechanismen Zugriffsspitzen besser ab. TYPO3 bietet gar eine Palette an Optimierungsmöglichkeiten wie serverseitiges Caching in der Datenbank, im File-System oder mit memcached [3] im Memory. Doch trotz sinkender Preise ist die Server-Hardware nach wie vor ein wesentlicher Kostenfaktor beim Hosting. Und je mehr Rechenleistung der Anbieter benötigt, desto mehr Strom und damit laufende Betriebskosten fallen an.

Eine ähnliche Sackgasse zeichnet sich bei applikationsspezifischen Caching-Mechanismen und Optimierungen ab. Je nach gewählter Lösung muss noch immer auf langsame Festplatten zugegriffen werden. Selbst beim Einsatz von schnellen Festplatten sind die Zugriffszeiten um ein Vielfaches höher als beim virtuellen Speicher. Noch schlimmer wird es, wenn die Applikation bei der Auslieferung des Caches beteiligt ist. Denn das stößt unnötig viele Prozesse an und kostet viel Rechenzeit. Das bereits genannte System memcached ist dabei sicherlich am vielversprechendsten – es kann bei situativen Anforderungen aber nicht steuernd eingreifen.

Auch das Optimierungspotenzial auf Applikationsebene – also eine effiziente Programmierung, eine CSS- und JavaScript-Kompromierung oder die Zusammenfassung von Icons zu Sprites und vieles mehr [4] – lässt sich ab einem gewissen Grad kostenmäßig nicht mehr rechtfertigen: Irgendwann halten sich Aufwand und Geschwindigkeitszuwachs einfach nicht mehr die Waage.

WARUM VARNISH WEITER GEHT

Der Website-Beschleuniger Varnish bietet den optimalen Ausweg aus diesen Sackgassen. Ohne zusätzlichen Hardware-Bedarf und dank flexibler Policies lässt sich auf individuelle Bedürfnisse eingehen. Viele dieser Performance-Optimierungen auf Caching-, Applikations- und Hardware-Ebene sind im Vergleich dazu „Peanuts“. Während Optimierungen auf Code-Ebene im unten angeführten Beispiel immerhin einige Hundert Millisekunden Geschwindigkeitszuwachs brachten, optimierte Varnish die Performance von zum Teil fünf Sekunden auf 0,05 Sekunden. Die Belastung der CPU sank von vier Prozessoren auf nur noch eine Einheit. Varnish hat damit zwei positive Effekte: Es senkt die Hosting-Kosten und steigert signifikant die Conversion Rate.



Nach dem Einsatz von Varnish reduziert sich die Belastung der CPU erheblich, die Performance steigt signifikant.

VARNISH IM EINSATZ

Die Integration von Varnish in eine bestehende Web-Lösung ist im Grundsatz recht einfach. Als Reverse-Proxy schaltet sich der Cache quasi als Schild vor den eigentlichen Webserver und organisiert die eingehenden Anfragen. Ist eine Anfrage noch nicht im Cache, sendet das System automatisch einen Request an den Webserver und nimmt ihn in den Cache auf. Ändert sich eine Seite im CMS oder Shop, muss das System eine Mitteilung an Varnish senden und so den Cache für diese spezifische Seite leeren. Diesen Mechanismus kann man beinahe bei allen modernen CMS- und Shop-Lösungen mithilfe einer Erweiterung nachrüsten.

Neben dieser Cache-Grundfunktion bietet Varnish aber noch weit mehr. Mithilfe von VCL liefert Varnish einen Teil der Systemlogik und Sicherheitsaspekte aus. Neben der bereits erwähnten Device-Detection gibt es zahlreiche frei verfügbare Module und Erweiterungen (VMODs). Für den professionellen Einsatz bieten sich zudem kommerzielle Module an [5]. Die Varnish Administration Console (VAC) ist zum Beispiel beim Einsatz mehrerer Varnish-Server sinnvoll und wenn Echtzeit-Informationen Engpässe beseitigen helfen sollen. Die Varnish Custom Statistics visualisieren die gesammelten Daten und ermöglichen so individuelle Auswertungen sowie die gezielte Analyse von Problemfällen.

DIE VARNISH-PAYWALL

Besondere Erwähnung verdient die Varnish-Paywall. Für viele Medienhäuser ist die Monetarisierung ihrer Online-Inhalte ein großes Thema und bei der schwindenden Akzeptanz von Online-Werbung rückt die Bezahlschranke als Option zunehmend in ihren Fokus. Die

erfolgreiche Einführung einer Paywall bei der Online-Ausgabe der New York Times im Jahre 2011 hatte Signalwirkung in der Medienwelt.

Derzeit lassen sich Paywalls zum Beispiel über JavaScript, SaaS oder direkt im CMS integrieren. Doch das Problem dabei ist, dass die Paywall pro User die Zugriffsrechte abfragen muss. Das hat massive Performance-Verluste und hohe Server-Investitionen zur Folge. Doch gerade zahlende Kunden sind gegenüber schlechter Performance besonders sensibel.

Mit der Varnish-Paywall ist dies nicht der Fall, da die Businesslogik direkt in Varnish integriert ist und sich äußerst schlank halten lässt. Sie unterstützt dabei die verschiedenen Zugriffskontrollen, wie Abonnement, Anzahl aufgerufener Artikel, Micro-Payment und vieles mehr. Zugriffskontrollen auf Applikationsebene sind damit überflüssig. Zahlreiche Zeitungen und Zeitschriften in Skandinavien verwenden bereits die Varnish-Paywall. Und auch die New York Times ist gerade dabei, ihre bestehende Paywall-Lösung durch die von Varnish zu ersetzen.

FAZIT

Wer die üblichen Optimierungen seiner Applikation auf Code-Ebene bereits umgesetzt hat und noch mehr Geschwindigkeit aus seiner Lösung herausholen möchte, kommt um einen Performance-Optimierer wie Varnish nicht herum. Varnish beschleunigt als Web-Proxy die Auslieferung von Websites um den Faktor zehn und mehr – und spart dabei sogar noch Rechenleistung. Richtig spannend ist das Caching System dank der flexiblen und schlanken VCL und den Zusatzmodulen für beinahe jede Fragestellung. ☺

LINKS ↗ Artikel diskutieren und alle Links auf t3n.de/3355

- [1] Aberdeen Group: <http://t3n.me/13My9Zk>
- [2] Performance im Google-Ranking: <http://t3n.me/12tX39R>
- [3] memcached: <http://memcached.org/>
- [4] Klassische Performance-Tipps: <http://t3n.me/14WejI9>
- [5] Varnish Module: <http://t3n.me/15Kdt2f>



DOMINIC BRANDER ist Mitgründer der Full-Service-Agentur snowflake productions GmbH. Er ist Open-Source-Begeisterter, Open-Street-Fanboy, todayu-Enthusiast, Mitgründer der TYPO3 Association und Teamleiter der TYPO3 Zertifizierung.



ANDRI STEINER ist Hostingverantwortlicher bei snowflake. Als offizieller Varnish Integrations- und Hostingpartner sorgt snowflake für schnelle Weblösungen. Andri Steiner liebt performante Websites, Höhenflüge mit dem Gleitschirm und gemütliche Fahrten mit seinem VW-Bus.
