

Justus Blümer

Löwenstarke Websites: Die richtige Vorbereitung auf Trafficspitzen

Vox hat mit „Die Höhle der Löwen“ den großen Wurf gelandet und mit teilweise über drei Millionen Zuschauern die quotenstärkste Primetime-Eigenproduktion geschaffen. In der Sendung präsentieren sich Start-ups aus den unterschiedlichsten Branchen vor fünf finanzstarken potenziellen Investoren, um Geld für die Weiterentwicklung ihrer Unternehmen zu erhalten. Die TV-Reichweite ist für die Gründer extrem wertvolle Werbung, die jedoch ohne die richtige Vorbereitung auch wirkungslos verpuffen kann.

Hohe Trafficspitzen sind ein Phänomen, das aus unterschiedlichsten Gründen auftreten kann: Ein bei reddit.com erfolgreicher Beitrag, Mittelpunkt eines brandaktuellen Themas (Germanwings-Absturz) oder einfach nur das erste Google-Ergebnis für eine „Wer wird Millionär?“-Frage zu sein, können ähnliche Effekte wie „DHDL“ haben. Unabhängig von der Ursache muss eine Website in diesen kritischen Momenten einwandfrei funktionieren, ansonsten entgehen ihrem Unternehmen direkte Einnahmen und Markenbekanntheit. Bricht unter der Last des Traffics die Website so zusammen, dass nur noch eine Fehlermeldung angezeigt werden kann, entsteht sogar ein negativer Eindruck.

Die technische Optimierung

Um mit viel Traffic umgehen zu können, muss die Infrastruktur im Hintergrund darauf ausgelegt sein. Dabei kommt es vor allem auf die Lastspitzen an, also Traffic pro Zeit.

Welche Software im Hintergrund zum Einsatz kommt, ist dabei nicht entscheidend: Ein Wordpress-System mit WooCommerce-Erweiterung kann genau so gut oder schlecht performen wie eine umfangreiche Magento-Installation. Wichtig ist, dass die folgenden Tipps umgesetzt werden, und das ist (wenn auch mit unterschiedlichem Aufwand) in allen Systemen möglich:

Die Last abgeben

Die erste und einfachste Maßnahme sollte sein, so viel Last wie möglich vom eigenen Server abzulenken und auf externe Anbieter auszulagern. Mit statischen Ressourcen wie JavaScript-,

CSS- und Bilddateien geht das am besten. Universelle Funktionsbibliotheken wie jQuery können kostenlos z. B. von Google Hosted Libraries eingebunden werden.

Für eigene Dateien gibt es Content-Delivery-Network(CDN)-Dienste wie Amazon CloudFront, fastly oder MaxCDN, die für deutlich unter 0,10 € pro übertragenem GB ihre Bandbreite zur Verfügung stellen. Das ist in der einfachsten Variante oder unter Zeitdruck nicht kompliziert: Sobald die Dateien dort hochgeladen sind, können sie unter ihrer neuen URL auf dem CDN eingebunden werden:

```
<script src="https://mein-startup.cloudfront.net/script.js"></script>
```

Insbesondere für große Bilder ist das besonders wichtig, weil sie leicht die Bandbreite des Hauptservers belasten – dessen Hauptaufgabe sollte aber die Auslieferung des HTML bleiben.

Noch größer sind meist nur Videos, die deshalb auch auf keinen Fall selbst gehostet werden sollten. Das kostenlose bzw. sehr günstige Hosting & Einbetten von YouTube oder Vimeo bietet neben einer stabilen Infrastruktur auch noch bessere Kompatibilität auf den unterschiedlichsten Endgeräten. Ganz nebenbei schafft man damit v. a. auf YouTube passenden Inhalt, den Interessierte auch dort suchen und teilen.

Browsecaching aktivieren

Leider nicht selbstverständlich, aber genau so einfach ist die Aktivierung der Browsercaching-Einstellungen, die dafür sorgt, dass der Browser bereits heruntergeladene Dateien kein zweites Mal abrufen muss.

DER AUTOR



Justus Blümer ist Berater bei Wingmen Online Marketing in Hamburg und beschäftigt sich besonders intensiv mit technischem SEO und der Analyse und Optimierung sehr großer Websites.

Name	Domain	Type	Size
RHqh14DjcvqZGJknoCzKXnYhjbSpvc47ee6XR_80Hnw.woff2	fonts.gstatic.com	font	9.6KB
hqdefault.webp	i.ytimg.com	webp	9.2KB
picturefill--5doYF3f1gS.js	dpbcfko19qo1q.cloudfront.net	script	8.7KB
hqdefault.webp	i.ytimg.com	webp	7.9KB
owl.carousel.min--Qx2ymv2gM6R.js	dpbcfko19qo1q.cloudfront.net	script	7.8KB
www.bluffix.com	www.bluffix.com	document	7.1KB
logo--VRy7jN8StkM.png	dpbcfko19qo1q.cloudfront.net	png	5.6KB
jquery.rating--DoEUJYARSf.js	dpbcfko19qo1q.cloudfront.net	script	5.2KB
JcBbv8_JdKtlhZ9y6LjIdAD-EV-EzPAX1QHCBcGfXjk.js	www.google.com	script	4.6KB
homepage-slider-sprite--CB63Hz3F92m.png	dpbcfko19qo1q.cloudfront.net	png	4.5KB
jquery.socialshareprivacy.min--h8J5JQb5snr.js	dpbcfko19qo1q.cloudfront.net	script	4.3KB
fbevents.js	connect.facebook.net	script	4.0KB

77 requests | 1.7MB transferred | Finish: 3.19s | DOMContentLoaded: 930ms | Load: 3.13s

Abb.1: Auf der Startseite des Start-ups Bluffix (bluffix.com) werden nur 0,05 % der gesamten 1,7 MB vom eigenen Server geladen, der Rest wird von extern eingebunden (Quelle: Google Chrome Entwickler-Tools)

```

27 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/default/limberry/css/styles.css" media="all" />
28 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/base/default/css/widgets.css" media="all" />
29 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/base/default/css/smarty/amlabel/amlabel.css" media="all" />
30 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/base/default/css/magento/default.css" media="all" />
31 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/default/limberry/css/local.css" media="all" />
32 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/default/limberry/css/cloud-zoom.css" media="all" />
33 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/default/limberry/css/multiroom.css" media="all" />
34 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/default/limberry/css/tooltip/prototype.css" media="all" />
35 <link rel="stylesheet" type="text/css" href="https://www.limberry.de/skin/frontend/default/limberry/css/print.css" media="print" />
36 <script type="text/javascript" src="https://www.limberry.de/js/lib/core.js"></script>
37 <script type="text/javascript" src="https://www.limberry.de/js/prototype/validation.js"></script>
38 <script type="text/javascript" src="https://www.limberry.de/js/scriptaculous/builder.js"></script>
39 <script type="text/javascript" src="https://www.limberry.de/js/scriptaculous/effects.js"></script>
40 <script type="text/javascript" src="https://www.limberry.de/js/scriptaculous/dragdrop.js"></script>
41 <script type="text/javascript" src="https://www.limberry.de/js/scriptaculous/controls.js"></script>
42 <script type="text/javascript" src="https://www.limberry.de/js/scriptaculous/slider.js"></script>
43 <script type="text/javascript" src="https://www.limberry.de/js/variant.js"></script>
44 <script type="text/javascript" src="https://www.limberry.de/js/variant/form.js"></script>
45 <script type="text/javascript" src="https://www.limberry.de/js/variant/menu.js"></script>
46 <script type="text/javascript" src="https://www.limberry.de/js/mage/cookie.js"></script>
47 <script type="text/javascript" src="https://www.limberry.de/js/mage/cookies.js"></script>
48 <script type="text/javascript" src="https://www.limberry.de/js/limberry/localize/popup.js"></script>
49 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery-1.5.2.min.js"></script>
50 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery-noconflict.js"></script>
51 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery.sno.alias.pack.js"></script>
52 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery-ui-1.8.13.custom.min.js"></script>
53 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery.scrollTo.js"></script>
54 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery.scrollTo.custom.js"></script>
55 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery.DDWindow.js"></script>
56 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/jquery.lightbox-0.3.min.js"></script>
57 <script type="text/javascript" src="https://www.limberry.de/skin/frontend/default/limberry/js/cloud-zoom-1.0.2.min.js"></script>
    
```

Abb.2: Limberry (www.limberry.de) lädt allein in diesem Codeblock 36 Einzeldateien, die zu zweien zusammengefasst werden könnten

Die Aktivierung der Expires oder Cache-Control-Header findet normalerweise in der Serverkonfiguration statt. Aber Achtung: Wenn kurzfristig noch eine Datei geändert wird, haben wiederkehrende Besucher die Datei eventuell noch in ihrem alten Zustand zwischengespeichert. Um ein frisches Herunterladen der Aktualisierung zu erzwingen, kann einfach der Dateiname geändert werden.

Ob Browsercaching bereits richtig aktiviert ist, kann man besonders einfach mit dem kostenlosen Tool Pagespeed Insights von Google ermitteln.

Weniger Requests und weniger Datenvolumen

Auslagerung und Browsercaching sind umso effektiver, wenn gar nicht erst viele einzelne Dateien aufgerufen werden müssen, denn für jede dieser Dateien wird ein einzelner HTTP-Re-

quest (eine Anfrage vom Browser an den Server) gebraucht – das bedeutet unnötigen Overhead. Mehrere einzelne CSS- und JavaScript-Dateien sollten zu einigen wenigen zusammengefasst werden, sodass die gleichen Daten mit weniger Anfragen geladen werden können. Viele Browser sind außerdem in der Menge ihrer gleichzeitigen Verbindungen zu einem Hostnamen limitiert, sodass die vielen Einzeldateien nur nacheinander statt gleichzeitig geladen werden.

Diese Empfehlung gilt jedoch nur, solange der Server kein HTTP/2 unterstützt. Mit dieser neuen Version des Übertragungsprotokolls können alle Einzeldateien mit nur einem Request abgefragt werden. Die Einrichtung von HTTP/2 ist je nach Hostinganbieter nicht immer einfach oder überhaupt möglich. Langfristig ist es aber die bessere Lösung, die ganz unabhängig von Traffic-Spitzen deutliche Performanceverbesserung für den User mit sich bringt.

Serverseitiges Caching

Die meisten Inhalte einer Website verändern sich selten: Die Startseite, textlastige Contentseiten und Produktseiten müssen deswegen nicht jedes Mal neu vom Webserver zusammengesetzt werden. Das schont insbesondere die Datenbank, neben dem Webserver der zweite kritische und ressourcenintensive Part einer Website.

Ein solcher Cache kann selbst aufgebaut werden über eine eingebaute Funktion im CMS oder spezielle Software wie Varnish. Die Pflicht, für leistungsfähige Server zu sorgen, gibt man damit aber noch nicht ab. Das geht nur, wenn man auch für diese Aufgabe einen externen CDN-Anbieter nutzt.

Mit der Entscheidung für Server und den dazugehörigen Hostinganbieter könnte man einen eigenen Artikel füllen. Grundsätzlich kann man aber festhalten, dass ein einzelner Server auch mit der stärksten Hardware wahrscheinlich nicht ausreichen wird. Um einen Loadbalancer, d. h. ein vorgeschaltetes System, das den Traffic auf die unterschiedlichen Instanzen verteilt, kommt man also nicht herum.

Insbesondere für die Start-ups aus der „Höhle der Löwen“ hat sich die Amazon Elastic Compute Cloud (EC2) bewährt. Die Abrechnung erfolgt stundengenau und je nach geforderter Leistung können weitere Instanzen kurzfristig und sogar automatisch hinzugebucht werden, um die Last weiter zu verteilen.

Das Loadbalancing muss nicht zwangsläufig beim Hostinganbieter geschehen, Google stellt beispielsweise eine Standalone-Lösung im Rahmen der Google Cloud Platform zur Verfügung.

Andere Grenzen

Jede Website hat ihre Eigenheiten und Sonderlösungen, die bedacht werden müssen. Wer nur die Startseite optimiert, erlebt vielleicht im Kauf-

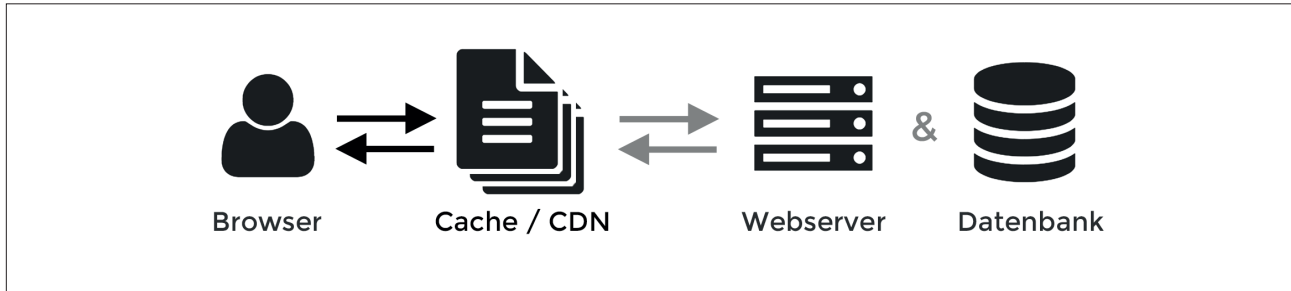


Abb.3: Der Browser fragt die meisten HTML-Dokumente aus dem Cache ab, sodass diese Aufrufe gar nicht erst bis zum Webserver und zur Datenbank gelangen; nur Informationen, die dort verarbeitet werden müssen, wie z. B. die Warenkorbverwaltung eines Shops, werden bis dorthin übertragen

schluss-Prozess sein blaues Wunder: Wo nutzerindividuelle Daten verarbeitet werden müssen, kann kein Cache zum Einsatz kommen und die Datenbank bricht unter der Last zusammen. Bei mehreren Start-ups aus „Die Höhle der Löwen“ ist genau das passiert: Die Zuschauer sind vom Produkt begeistert, legen es in den Warenkorb und bringen damit die ansonsten durchoptimierte Seite doch noch zu Fall.

Nicht vergessen sollte man beispielsweise:

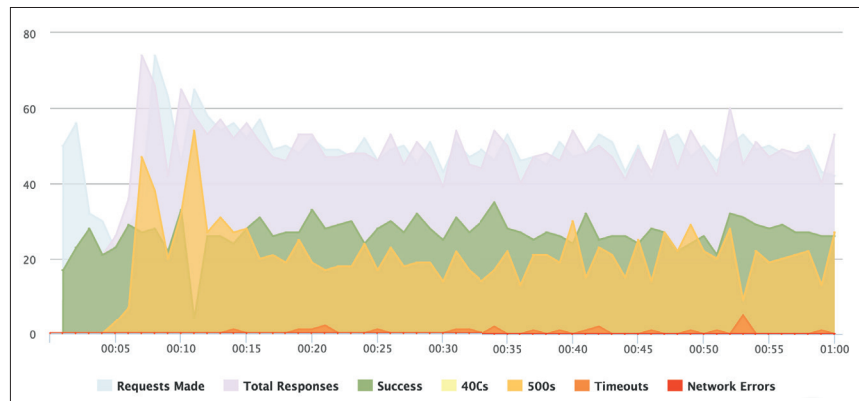


Abb.4: Schon bei unter 100 Anfragen pro Sekunde zeigt diese unoptimierte Website bei einem Standard-Sharedhoster Ausfallerscheinungen und gibt in rund 50 % der Fälle Fehler zurück

Ultraschnelles
High-Performance
SSD-Webhosting mit nginx

- » App-Server, wenn man eine eigene App betreibt.
- » Die interne Suchfunktion, die eventuell über ein eigenes System wie ElasticSearch funktioniert.
- » Auf der Website angebotene Tools wie z. B. Konfiguratoren, die besondere Serverlast erzeugen können.
- » E-Mail-Systeme, die Probleme machen, wenn 20.000 Bestellbestätigungen innerhalb weniger Minuten versendet werden müssen.
- » Selbst gehostete Analysesysteme wie Piwik sollten deaktiviert werden; es ist unrealistisch, sie mit vertretbarem Aufwand für solche Lastspitzen fit zu machen.

Auch externe Tools können eigene Grenzen mitbringen: Die kostenfreie Version von Google Analytics hat ein Limit von zehn Millionen „Hits“ pro Monat. Hat eine Website also schon relativ viel Traffic und hohe Hit-Zahlen durch intensives Ereignistracking, kann es passieren, dass diese Schwelle überschritten wird. Google garantiert dann nicht mehr, dass alle Daten korrekt erfasst werden.

Lasttests

Auch die beste Vorbereitung muss getestet werden. Um die Bedingungen von so viel Traffic nachstellen zu können, gibt es Tools wie loader.io, die in vordefinierter Zeit eine große Menge Anfragen an den Server stellen.

Bei einem entsprechend großen echten Trafficansturm können Tausende Requests pro Sekunde durchgeführt werden, dementsprechend sollten auch die Tests angelegt werden. Hundertprozentige Sicherheit können jedoch auch solche Testtools nicht bieten, weil sie keine echten Aufrufe in einem Browser durchführen und damit kein Surfverhalten im Bezug auf Ladereihenfolge oder Wechselwirkungen zwischen einzelnen Softwarekomponenten realistisch nachbilden können.

Die Reichweite langfristig nutzen

Wenn alles gut geht, kann man sich insbesondere bei einer so starken Trafficquelle wie „Die Höhle der Löwen“ über Unmengen Traffic freuen. „SugarShape“ (www.sugarshape.de) beispielsweise hat es geschafft, 125.000 Besucher gleichzeitig auf der Website zu bedienen. In den ersten 24 Stunden nach dem TV-Auftritt sorgte das für 16.000 Bestellungen.

Nicht alle Kunden sind so kaufbereit – vielleicht noch nicht. Mit Remarketing-Pixeln von Google und Facebook können sie aber erfasst und über AdWords oder Facebook-Ads wieder angesprochen werden, z. B. mit einem passenden „DHDL-Rabatt“ oder ähnlichen Aktionen. Dabei sollte man die Möglichkeiten zur weiteren Klassifizierung nutzen und beispielsweise Nutzer mit Artikeln im Warenkorb, aber ohne abgeschlossenen Kauf, zusätzlich identifizierbar machen. Diese Aussteuerung erfolgt im Browser und hat damit keinen negativen Einfluss auf die Serverperformance. Um die IT-Ressourcen nicht auch noch mit Trackingimplementierung zu belasten, kann das auch über ein externes Tool wie den Google-Tag-Manager stattfinden.

Langfristiger als die kurze Trafficspitze sind die Sekundäreffekte, die sich nach einem TV-Auftritt einstellen: Nicht ganz uneigennützig, weil sie selbst von der Nachfrage und dem Traffic profitieren, berichten im Anschluss viele Newsportale über die vorgestellten Start-ups. Das bringt kurzfristig eine weitere Runde Aufmerksamkeit, aber vor allem Links von den etablierten News-Seiten auf die Shops der Start-ups. Die sind für Google ein positives Signal und können das Ranking der Jungunternehmen in den Suchergebnissen deutlich verbessern.

Schadensbegrenzung

Auch mit der besten Vorbereitung und einer eingespielten IT kann es Probleme geben. Wenn es doch so weit

kommt, sollte es Maßnahmen geben, um die Reichweite nicht ganz ungenutzt zu lassen.

Es sollte niemals der Fall auftreten, dass eine kryptische Fehlerseite angezeigt wird, die den Nutzer hilflos allein lässt. Stattdessen sollte ein dem allgemeinen Branding angepasster, laienverständlicher Hinweis ausgespielt werden, der ehrlich kommuniziert, dass aufgrund der hohen Nachfrage gerade die Server heiß laufen.

Da kein Kauf stattfinden kann, bietet es sich an, den Nutzer um seine E-Mail-Adresse zu bitten, um ihn später (mit einem Rabattcode und Dank für seine Geduld) auf die wieder funktionierende Website zu holen. Das Formular darf natürlich nicht den eigenen Server zum Ziel haben, sondern muss die Daten direkt zu einem externen Anbieter wie MailChimp oder Aweber schicken. Für Nutzer, die dazu noch nicht bereit sind, bietet sich ein eingebetteter Facebook-Follow-Button an.

Um nicht alle Verkäufe zu verlieren, kann zusätzlich zum eigenen Shop ein Angebot bei Amazon hilfreich sein, auf das von eventuellen Fehlerseiten verlinkt wird. Um diese als Quelle der Amazon-Verkäufe nachzuvollziehen, kann man über das Amazon-Partnernet einen eigenen Affiliatelink dafür generieren.

Der Haken: Es werden Amazon-Verkäufergebühren fällig, die je nach Kategorie zwischen 7 und 20 % betragen. Außerdem hat man keinen Zugriff auf die E-Mail-Adresse des Käufers.

Fazit

Für junge und kleine Unternehmen kann eine so plötzliche Millionenreichweite der entscheidende Schlüssel zum Erfolg werden, wenn man es schafft, sie zu nutzen. Das klappt nur mit rechtzeitigem Planen und Handeln. ¶