

Florian Schneider

»Die Evolution des sichtbaren Bereichs

Die Bedienung und das Nutzungsverhalten im Internet haben sich in den letzten Jahren grundlegend gewandelt. Dies betrifft natürlich auch die Bedeutung des im Browserfenster sichtbaren Bereichs und den sorgsamsten Umgang mit dieser Thematik hinsichtlich des Webdesigns.

Der sichtbare Bereich ist – zumindest in der Theorie – die Fläche, die ein Nutzer auf jeden Fall sieht, ohne dass er horizontal oder vertikal scrollen muss. In der Praxis ist das ein wenig komplexer, denn die Nutzer bewegen sich im Internet auf unterschiedlichen Endgeräten mit verschiedenen Monitorgrößen und Auflösungen. Hinzu kommt: Hat ein Nutzer besonders viele Symboleleisten integriert, verkleinert sich der für ihn sichtbare Bereich weiter. Es handelt sich beim sichtbaren Bereich also nur um einen Durchschnittswert, was seine Bedeutung für die Entwicklung von Online-Angeboten seit den Anfangszeiten des Internets jedoch in gar keiner Weise schmälert. Mit der steigenden Anzahl von Inhalten auf Online-Präsenzen nahm auch die Bedeutung des sichtbaren Bereichs weiter zu. Dabei war der Umgang mit diesem Thema stets gleich: Alle relevanten Inhalte und alle Zugänge zu den jeweiligen Inhalten sollten stets möglichst direkt sichtbar platziert werden. Seit dem Einzug von Personalisierung, Customization und Social Media existieren jedoch weitere strategische Ansätze in diesem Bereich, die im Folgenden genauso wie der klassische Ansatz unter die Lupe genommen werden.

① Der klassische Umgang mit dem sichtbaren Bereich – Beispiel: web.de

Beim klassischen Ansatz wird versucht, möglichst alle wichtigen Themen innerhalb des sichtbaren Bereichs abzubilden. Dabei priorisiert der Website-Betreiber die Inhalte. Er definiert – gern auch mithilfe von Marktforschung und Usability-Tests –, was für seine Zielgruppe relevant sein könnte. Aufgrund der Vielzahl an relevanten Inhalten ist es allerdings oft unmöglich, diese alle unterzubringen. Deswegen gibt es einige Tools

und Funktionalitäten, mit denen sich noch mehr Content integrieren lässt. Dazu gehören beispielsweise Aufklappmodule oder kleine Tab-Navigationen innerhalb des Contents. Ein Vorteil der klassischen Strategie ist, dass ganz gezielt einzelne Inhalte nach vorne gebracht werden können. Darüber hinaus haben Nutzer dieses Prinzip gelernt und erkennen sofort, welche Themen oder Produkte für den Betreiber einen hohen Stellenwert besitzen. Eine kleine reine Image-Website mit wenigen Inhalten kann sehr gut komplett innerhalb des sichtbaren Bereichs funktionieren. Gerade bei umfangreichen Webangeboten jedoch überwiegen die Nachteile, denn der klassische Ansatz bricht mit einer sehr wichtigen Usability-Regel – der Begrenzung der Inhalte. Das menschliche Gehirn ist nicht in der Lage, eine unbegrenzte Anzahl einzelner Objekte wahrzunehmen. Ideal sind bis zu sieben sinnvoll zusammengefasste optische Blöcke. Wird der Nutzer visuell mit zu vielen Elementen konfrontiert, führt dies sehr oft direkt zum Abbruch eines Website-Besuchs.

Ein gutes Beispiel für diese Problematik ist das Portal web.de (Abb.1). Es existieren 76(!) klickbare Elemente, die alle für spezifische Inhalte stehen. Das ist die quantitative Seite. Es gibt allerdings auch noch eine qualitative Überforderung: Für den einzelnen Nutzer, der in einem bestimmten Use-Case vor seinem Computer sitzt, sind von den 76 Links vielleicht gerade einmal zehn interessant. Es ist nun seine Aufgabe, das ganze Angebot zu scannen. Er muss unzählige Texte anlesen und vielleicht noch auf den einen oder anderen Tab klicken, um zu den gewünschten Inhalten zu gelangen. Und genau dies steht im klaren Widerspruch zu der Bedie-



DER AUTOR



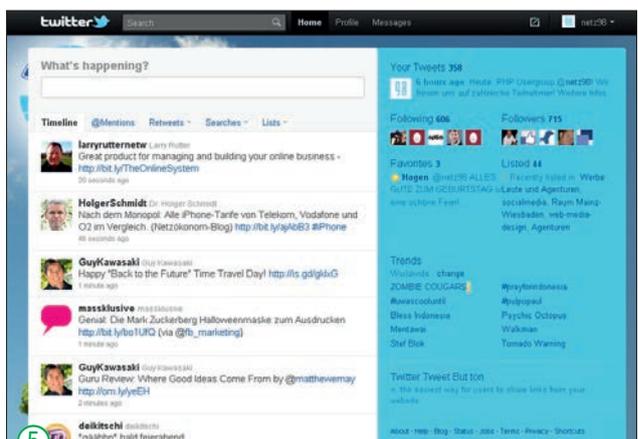
Florian Schneider
(Jg. 1975)
ist Creative Director und Managing Partner bei der netz98 new media GmbH. Bei netz98 leitet Schneider die Kreation und ist zudem in der Beratung und im Bereich Marketing und Sales tätig.



nung im Internet, wie sie von der Mehrzahl der Nutzer gewünscht wird: Nutzer möchten sehen und nicht lesen, sie möchten visuell und intuitiv geführt werden und nicht selber herausfinden müssen, wie sie zu den gewünschten Inhalten gelangen. Zudem führt dieser Overload an Elementen sehr oft zu einer selektiven Nutzung des Angebots. Es wird dann nur ein sehr kleiner und gelernter Teil des Leistungsspektrums in Anspruch genommen, beispielsweise die Free-Mail-Funktion bei web.de. Insgesamt gilt: Während der klassische Ansatz für eine kleine Website mit wenigen Inhalten noch funktionieren kann, stößt er bei komplexeren Angeboten klar an seine Grenzen und ist deswegen ein Auslaufmodell.

② Der zum Scrollen einladende sichtbare Bereich – Beispiel: Spiegel online

Einige Portale, wie zum Beispiel Spiegel online, setzen auf die Bereitschaft der Nutzer zum Scrollen (Abb. 2). Im sichtbaren Bereich ist nur das Top-Thema zu finden, alle weiteren Inhalte sind hauptsächlich über vertikales Scrollen erreichbar. Hierbei ist es essenziell, dass der Nutzer sehr gut visuell geführt wird. Er muss beispielsweise sofort erkennen können, dass er scrollen muss. Darüber hinaus sollte die grundsätzliche Anordnung der Bereiche immer gleich sein – das macht den Auftritt für den Nutzer sehr schnell erlernbar. Der Nutzer erkennt in diesem Fall rasch die für ihn irrelevanten Kategorien und überspringt sie nach einem gewissen Zeitraum so gut wie automatisch.



Für diesen Ansatz spricht, dass immer mehr Nutzer lieber scrollen als klicken. Das liegt unter anderem daran, dass jeder Klick mit einer entsprechenden Reload-Zeit verbunden wird. Zahlreiche Usability-Tests belegen diese Entwicklung hin zum Scrollen. Ein weiterer Vorteil liegt in der verbesserten visuellen Führung. Das lässt sich sehr gut am Beispiel von Bewertungen verdeutlichen: Im klassischen Modell sind solche Funktionalitäten aus Platzgründen oft über eine Tab-Navigation innerhalb des Contents oder über Links zugänglich – der Nutzer muss das Wort „Bewertungen“ wirklich

lesen. Besser ist es, wenn die Funktionalitäten so gestaltet und platziert sind, dass der Nutzer sie sofort als solche erkennt. Bei Bewertungen erreicht man dies beispielsweise durch die Verwendung der bekannten Sternchen.

③ Der „easy to use – hard to handle“-Ansatz – Beispiel: Amazon

Amazon hat einen weiteren Ansatz geprägt (Abb. 3). Dieser verfolgt das Ziel, auf den jeweiligen Use-Case eines Nutzers einzugehen. So sollen Nutzer, die genau wissen, was sie möchten, und den Fokus auf einen schnellen Vorgang legen, genauso adäquat bedient werden wie Nutzer, die noch sehr unentschlossen sind und sich erst sehr ausführlich informieren wollen. Dies ist bei Amazon auf der gleichen Seite möglich. Am Beispiel der Artikeldetailseite kann man dies sehr gut erkennen: Alle Informationen und Funktionalitäten, die zum Kauf eines Artikels benötigt werden, sind im sichtbaren Bereich zu finden. Je weiter der Nutzer scrollt, desto tiefgreifender, detaillierter und spezialisierter werden die Informationen. Der Vorteil liegt auf der Hand: Der Schnellentschlossene fühlt sich genauso verstanden wie der Informationsbegierige und vielleicht noch Unentschlossene. Wie bei Ansatz zwei wird auch hier darauf gebaut, dass gerade Nutzer, die sich etwas intensiver informieren möchten, lieber ein bisschen scrollen und entscheiden, welches Modul für sie interessant ist, als eine Vielzahl an Links durchzulesen, um dann jeweils auf neue Seiten zu gelangen.

④ Der personalisierte sichtbare Bereich – Beispiel: Facebook

Dank Social Media und Personalisierung gibt es immer mehr Angebote, die jedem einzelnen Nutzer auch sein eigenes, individualisiertes Portal bieten. Das bedeutet deutlich weniger Inhalte, dafür aber Inhalte mit einem sehr hohen Relevanzgrad. Am Beispiel von Facebook (Abb. 4) lässt sich dies sehr gut veranschaulichen: Im Friendfeed sehen Benutzer aktuelle Aktivitäten und Äußerungen des eigenen Netzwerks und in den Höhepunkten wird alles nach vorne gebracht, was im Netzwerk beliebt ist. Zudem kann man sich in der unten platzierten Toolbar unter anderem interne Bookmarks setzen und

**„ Fakt ist:
Der sichtbare
Bereich ist nach
wie vor ein sehr
wichtiger
Faktor bei der
Entwicklung
von Online-
Angeboten.“**

man sieht, wer von den Freunden ebenfalls online ist. Durch Funktionen wie beispielsweise „Person X verbergen“ oder „Applikation X verbergen“ hat der Nutzer direkten Einfluss darauf, welche Inhalte er angezeigt bekommt und welche nicht. Er gestaltet sein persönliches Portal, ohne dass er zuvor aufwendige Konfigurationen durchführen muss. Bemerkenswert ist bei diesem Ansatz, dass sich wieder sehr viele Inhalte im sichtbaren Bereich befinden. Im Endeffekt jedoch werden die Inhalte ganz gezielt für den einzelnen Nutzer und auch vom einzelnen Nutzer ausgewählt.

⑤ Der mehrdimensionale sichtbare Bereich – Beispiel: Twitter

Der folgende Ansatz des mehrdimensionalen sichtbaren Bereichs trifft auch in Teilen auf Social Networks wie Facebook zu. Bei Twitter ist das Ganze allerdings noch wesentlich konsequenter umgesetzt (Abb. 5). Hier wird einfach nur der Strom aus Meinungs-, Status- und Befindlichkeitsäußerungen der Nutzer, denen man folgt, angezeigt. Dabei kommt eine weitere Dimension ins Spiel: die Zeit. Ab einer bestimmten kritischen Menge an Leuten, denen man folgt, entsteht ein ständiger Strom an Tweets. Ein Reload nach kurzer Zeit reicht und alle Inhalte innerhalb des sichtbaren Bereichs sind ausgetauscht.

Fazit

Bis auf den klassischen Ansatz haben alle Strategien gute Chancen, noch beliebter zu werden – möglicherweise sogar in Kombination miteinander. Nur eben für den klassischen Ansatz könnte es etwas eng werden, denn die Nutzer hatten noch nie besonders große Lust, innerhalb eines riesigen Angebots selbst herausfiltern zu müssen, welche Inhalte für sie relevant sind. Und diese Bereitschaft sinkt weiter, seitdem viele Portale und Plattformen auf den einzelnen Nutzer und sogar auf dessen aktuellen Use-Case eingehen und das entsprechende Angebot maßschneidern. Fakt ist: Der sichtbare Bereich ist nach wie vor ein sehr wichtiger Faktor bei der Entwicklung von Online-Angeboten. Der Unterschied zu früher ist, dass immer mehr für den Nutzer und vom Nutzer bestimmt wird, was sich innerhalb des sichtbaren Bereichs befindet. ¶