

Martin Günther

»Suchmaschinenoptimierung für TYPO3 – die Basics

TYPO3 erfreut sich aufgrund seiner vielfältigen Erweiterbarkeit und hohen Anpassungsfähigkeit zunehmender Beliebtheit. Suchmaschinenexperte Martin Günther erläutert die wesentlichen Möglichkeiten zur Suchmaschinenoptimierung mit TYPO3 und gibt konkrete Praxistipps zur Umsetzung an die Hand.

Von Haus aus produziert [TYPO3*](#) bereits einen relativ suchmaschinenfreundlichen und crawlbaren Quellcode und stellt somit keine Barriere für Suchmaschinen dar. Dennoch gibt es vielfältige Möglichkeiten, die Ausgabe für Suchmaschinen zu optimieren und die Auffindbarkeit der eigenen Website somit noch weiter zu verbessern. Dieser Artikel basiert auf der aktuellen TYPO3-Version 4.4.2 (sämtliche Einstellungen sind abwärtskompatibel und funktionieren damit auch in den meisten älteren Versionen) und zeigt, wie man die wesentlichen Optimierungsvorschläge aus dem Artikel „Suchmaschinenoptimierung Basics I“ aus der letzten Ausgabe von Website Boosting mit TYPO3 umsetzen kann.

«Title»

Der „Title“ einzelner Webseiten wird von TYPO3 auch ohne Anpassungen bereits mit Inhalten belegt, allerdings ist die Voreinstellung aus Suchmaschinensicht nicht optimal. Standardmäßig wird der in den Seiteneigenschaften eingegebene Seitentitel von TYPO3 für den Title verwendet. Idealerweise beinhaltet der Title vier bis sechs Wörter, die in der Reihenfolge der Bedeutung gesetzt werden. Problematisch an der Arbeitsweise von TYPO3 ist, dass genau dieser Seitentitel auch im TYPO3-Seitenbaum dargestellt wird, der bei Verwendung von vier bis sechs Wörtern schnell unübersichtlich wird, wie in Abbildung 1 zu sehen ist (die Seitentitel wurden hier aus Gründen der besseren Verständlichkeit bewusst so gewählt). Um die Übersicht im [Backend*](#)

zu bewahren, ist es daher hilfreich, den Seitentitel zur Erkennung möglichst kurz und aussagekräftig zu halten und stattdessen ein anderes, optionales Feld für die gezielte Optimierung des eigentlichen <title> für den Headerbereich der Seite zu verwenden. Hierfür gibt es grundsätzlich zwei Möglichkeiten. Die erste Möglichkeit besteht darin, dass man ein bereits vorhandenes Feld (z. B. den Untertitel) verwendet, um den Wert des <title>-Tags zu setzen bzw. zu überschreiben. Dazu muss folgender [TypoScript-Code*](#) in das (Haupt-)TypoScript-Template eingefügt werden:

```
# TYPO3 internes <title> Tag ausschalten
config.noPageTitle = 2
page.headerData.10 = TEXT
# Untertitel überschreibt das Feld Seitentitel
page.headerData.10.field = subtitle // title
# <title> Tag formatieren
page.headerData.10.wrap = <title>| &nbsp; - domain.de</title>
```

Damit wird der Untertitel für das <title>-Tag verwendet, sofern hier Inhalte hinterlegt werden. Bleibt das Feld Untertitel leer, verwendet TYPO3 weiterhin den Seitentitel. Hinweis: Die Zeichenfolge „ - meinedomain.de“, nach dem „|“ ist optional. Hier kann beispielweise die Domain oder auch der Firmenname an das Ende des <title>-Tag aller Seiten automatisch hinzugefügt werden – man beachte hierbei aber die ideale Länge (von ca. 65 Zeichen) des <title>-Tags. Vorteil dieser Methode ist, dass es sich schnell und ohne Installation eines Plugins umsetzen lässt. Dadurch erwirbt man sich aber den Nachteil, dass der Unter-

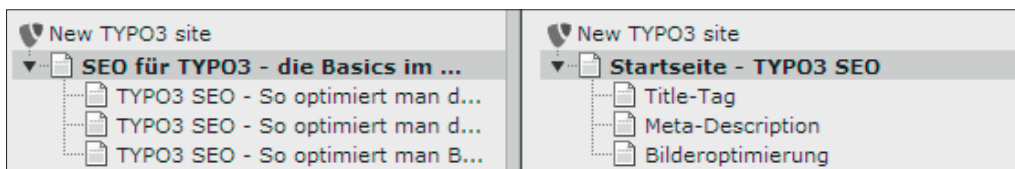


Abbildung 1: Durch zu lange Seitentitel (links) verliert man schnell den Überblick im TYPO3-Seitenbaum. Kurze, aussagekräftige Seitentitel helfen dagegen, sich schnell zu orientieren.

* siehe Glossar Seite 96-98

DER AUTOR



Martin Günther als Website-Boosting Engineer

für die tms development GmbH in Nürnberg und beschäftigt sich seit mehreren Jahren professionell mit den Themen Online-Marketing und Web-Programmierung.

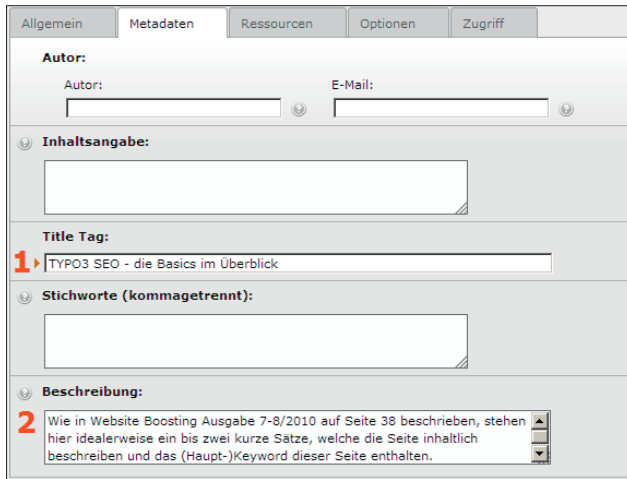


Abbildung 2: 1. Zusätzliches Feld für das <title>-Tag nach Installation von Basic SEO Features. 2. Feld für die Meta Description.

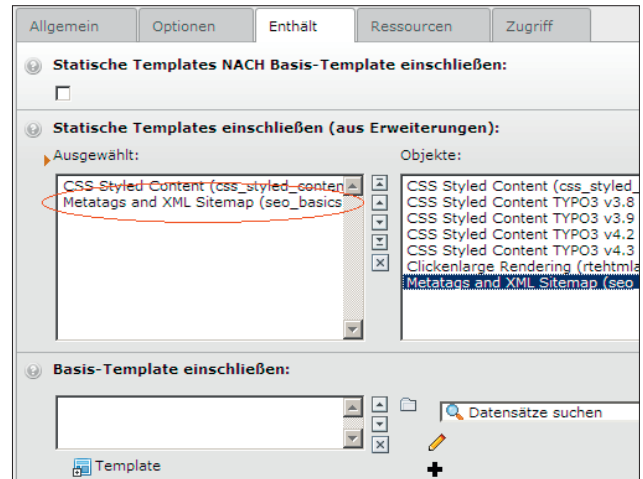


Abbildung 3: Das TypoScript Template „Metatags and XML Sitemap“ muss hinzugefügt werden, damit das zusätzliche Feld für das <title>-Tag im Frontend geladen wird.

titel damit blockiert wird und es durch- aus Anwendungsfälle gibt, in denen man das Feld zur Darstellung eines Untertitels auf der Seite selbst benötigt. Deshalb kann sich durchaus in den meisten Fällen die zweite Möglichkeit lohnen, die Installation eines zusätzlichen Plugins. Empfehlenswert ist für diese Aufgabe: Basic SEO Features (seo_basics) – die aktuelle Version findet man im TYPO3 Extension Repository unter <http://bit.ly/wsb591>. Nach erfolgreicher Installation über den Extension Manager findet man unter den Metadaten in den Seiteneigenschaften ein zusätzliches Feld „Title Tag“ das den Seitentitel überschreibt, sobald etwas eingegeben wird (siehe Abbildung 2, Punkt 1). Damit das <title>-Tag auf der Seite im Frontend tatsächlich geladen wird, muss einmalig noch die mitgelieferte TypoScript-Extension-Datei im Haupt-TypoScript-Template der Seite eingebunden werden, wie in Abbildung 3 zu sehen ist.

<Description>

Beim Einsatz des Basic SEO Features Plugin braucht man sich in Sachen Meta Description um nichts mehr zu kümmern, außer natürlich relevante und individuelle Inhalte zu hinterlegen. Das entsprechende Feld findet man in den Seiteneigenschaften, unter dem Reiter „Meta-Daten“, und es trägt die Überschrift „Beschreibung“ (siehe Abbildung 2, Punkt 2). Ohne Einsatz des Plugins gibt es auch

hier die Möglichkeit, das Meta-Description-Tag direkt über TypoScript einzubinden – in diesem Fall mit nur einer Zeile Code:

```
page.meta.description.field = description
```

<Keywords>

Der Vollständigkeit halber sei die Einstellung für das (unnötige) Meta-Keywords-Tag hier auch noch erwähnt. Es verhält sich analog zur Meta Description. Mit Einsatz des Plugins muss nur noch das Feld „Stichworte“ ausgefüllt werden. Die alternative TypoScript-Codezeile lautet:

```
page.meta.keywords.field = keywords
```

Alle (wichtigen) Tags im Überblick

Das beschriebene Plugin bietet noch eine weitere nützliche Funktion, die allerdings leider recht gut versteckt im Info-Modul untergebracht wurde. In der Drop-Down-Auswahl des Info-Moduls ist die Funktion „SEO-Management“ (siehe Abbildung 4, Punkt 1) hinzugekommen, die dem Nutzer eine Übersicht aller Unterseiten (je nach eingestellter Anzeigeebene) und den dazugehörigen bzw. hinterlegten Title-, Description- und Keywords-Tags in Tabellenform anzeigt. Über den Button „Edit SEO Fields“ hat der Nutzer die Möglichkeit, alle angezeigten Tags direkt und innerhalb der Übersicht

zu bearbeiten. Bei den hinterlegten Farben, wie in Abbildung 4, Punkt 2 zu sehen, handelt es sich um eine weitere Funktion. Sie signalisieren dem Nutzer die optimale (Zeichen-)Länge des jeweiligen Tags. Hier gilt: gelb = zu kurz (Title: unter 50 Zeichen, Description: unter 115 Zeichen), grün = passt (Title: zw. 50 u. 65 Zeichen, Description: zw. 115 u. 145), orange = zu lang (Title: über 65 Zeichen, Description: über 145 Zeichen). Diese Werte sind nicht absolut zu sehen, geben dem Nutzer aber einen guten Überblick darüber, an welchen Stellen man gegebenenfalls nachjustieren sollte.

Überschriften

Hinsichtlich der Überschriften liefert TYPO3 – im Vergleich zu früheren Versionen – bereits relativ gute Ergebnisse. Einziger Kritikpunkt: Im Einsatz mit dem CSS-styled-content-Modul, welches zum TYPO3-Standard gehört, werden Überschriften, um deren Formatierung zu erleichtern, in unnötige <div>-Tags eingeschlossen.

Wesentlich eleganter und vor allem platzsparender – Geschwindigkeit spielt, wie in der ersten Ausgabe von Website Boosting ja bereits erwähnt, ebenfalls eine Rolle – wäre stattdessen einfach nur <h1>Überschrift 1</h1>. Durch Entfernung des unnötigen HTML-Codes wird es jedoch schwieriger, einzelne <hx> Überschriften gleicher Ebene unterschiedlich zu formatieren (Beispiel: erste <h2> Über-

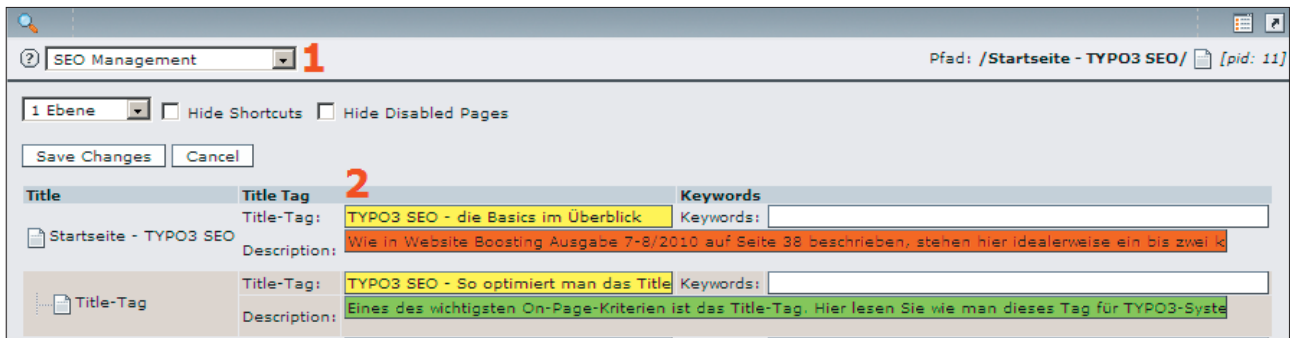


Abbildung 4: 1. Funktion „SEO Management“ im Drop-Down-Menü des Info-Moduls auswählen. 2. Direkte Bearbeitung der Title-, Description- und Keywords-Tags.

schrift größer als die zweite) – dieser Versuchung sollte man den Nutzern zuliebe aber ohnehin widerstehen. Fazit: Am besten entfernen! Den unnötigen Ballast wird man mit den folgenden Zeilen Typo-ScriptCode los:

```
lib.stdheader.3.headerClass =
lib.stdheader.3.headerClass.noTrimwrap = |
lib.stdheader.stdwrap.datawrap = |
```

Quellcode optimieren

Auch der von TYPO3 verwendete [Rich Text Editor \(RTE\)*](#) produziert stellenweise zusätzlichen Code, um die Formatierung einzelner Elemente zu erleichtern. Aber damit bläht er den Quellcode unnötigerweise auf. Das Layout lässt sich mithilfe von CSS und etwas Übung auch ohne „Formatierungs“-Code relativ problemlos anpassen. Deshalb gilt auch hier die Empfehlung: Am besten entfernen! Die nötigen Anpassungen erfolgen ebenfalls über TypoScript.

```
lib.stdheader.3.headerClass =
lib.stdheader.3.headerClass.noTrimwrap = |
lib.stdheader.stdwrap.datawrap = |
```

Im nächsten Schritt werden die sogenannten Prefix Comments, die das Lesen des Quellcodes erleichtern (Beispiel: <!-- Header: [begin] --> hier steht HTML-Code <!-- Header: [end] -->) entfernt. Diese Kommentare sind während der Entwicklungsphase oder zur Fehlersuche nützlich, haben aber im Produktivbetrieb nichts mehr verloren. Die folgende Zeile TypoScript sorgt für deren einfache Entfernung:

```
config.disablePrefixComment = 1
```

Für ältere TYPO3-Versionen sollten nachfolgende Einstellungen durchgeführt werden, die dafür sorgen, dass CSS und JavaScript in externe Dateien ausgelagert werden – sicher ist sicher.

```
# Javascripts in externe Datei auslagern
config.removeDefaultJS = external
# CSS in externe Datei auslagern
config.inlineStyle2TempFile = 1
```

Den letzten Schliff bei der Optimierung des Quellcodes erhält man mit dem Plugin SourceOptimization (sourceopt). Dieses Plugin ist relativ leicht zu konfigurieren und verkürzt den Quellcode, je nach gewählter Einstellung, nochmals deutlich.

Die einzelnen Einstellungen sind übersichtlich in der Dokumentation der aktuellen Version des Plugins zu finden (<http://bit.ly/wsb312>). Daher wird auf eine detaillierte Darstellung hier aus Platzgründen verzichtet. Zudem verrichtet das Plugin bereits in der Standardkonfiguration sehr gute Dienste. Dennoch gibt es zwei Einstellungen, die besonders wichtig – und daher erwähnenswert – sind. Da wäre zum einen die Einstellung `enable_utf-8_support`, die man in jedem Fall abschalten sollte, falls das gesamte System nicht bereits vollständig auf UTF8 umgestellt wurde. Andernfalls kommt es hier sehr wahrscheinlich zu Problemen, wie beispielsweise leere Zeichenfolgen und leere HTML-Tags. Die Einstellungen lassen sich sehr bequem über den TYPO3-Konstanten-Editor bearbeiten (siehe Abbildung 5) oder alternativ auch direkt über TypoScript.

```
# UTF8 ausschalten (0=aus, 1=ein)
config.sourceopt.enable_utf-8_support = 0
# HTML-Code-Reduzierung (0=aus, 1-4=ein)
config.sourceopt.formatHtml = 1
```

Mit der zweiten Codezeile wird festgelegt, wie stark der Quellcode reduziert werden soll. Der Wert 0 schaltet die Funktion aus, und es werden nur noch HTML-Kommentare entfernt. Bei einem Wert von 1 werden sämtliche Leerzeile und Umbrüche entfernt, und der gesamte Quellcode wird in einer Zeile ausgegeben (entspricht der Empfehlung „Minify HTML“ des Firefox Plugins PageSpeed, siehe hierzu auch WSB Ausgabe 5-6/2010 „Schneller Server – gutes Ranking?“). Bei den Werten 2, 3 oder 4 wird der Quellcode strukturiert und mit Einrückungen ausgegeben. Dabei gilt: Je höher der Wert, desto stärker wird strukturiert und desto mehr Umbrüche und Leerzeichen werden der Übersichtlichkeit halber eingefügt. Daher sollte man den Wert von `formatHtml` im Produktivbetrieb auf jeden Fall auf 1 setzen, während sich in der Entwicklungsphase oder gezielt bei der Fehlersuche 4 als hilfreich erweist.

Mit diesen Einstellungen zur Optimierung des Quellcodes werden die Seiten bereits deutlich schlanker und somit auch schneller – und man schont die Nerven von Mensch und die Ressourcen der Maschine.

Lesen Sie in der nächsten Ausgabe, wie man mit TYPO3 saubere, statische URLs erstellt und noch einiges mehr an Geschwindigkeit aus diesem System herausholen kann. ¶

* siehe Glossar Seite 96-98