

## TYPO3: DESIGNVORLAGEN UND CSS

von Jörg Drzycimski

**Moderne grafische Designs umzusetzen erfordert bei den grossen CMS häufig sehr viel Arbeit und Programmier-Know-How. Das Open Source CMS Typo3 bietet hierfür einen einfachen und schnellen Weg: externe Designvorlagen.**

Gerade in grossen Agenturen, wo ein Screendesigner die Vorlagen liefert, sind die Anforderungen an eine Website sehr komplex, und die Umsetzung in einem CMS bereitet dem Webdesigner häufig Probleme. Der einfachste und schnellste Weg, eine Site unter Typo3 aufzubauen, führt über Designvorlagen, sogenannte **Templates**. Hierbei handelt es sich um nichts anderes als eine mittels Photoshop und eines HTML-Editors erstellte **HTML-Datei**. Header-Grafiken, Farb-Schemata und Content Container lassen sich dabei beliebig mit Typo3 austauschen. Die Teile der Website, die sich von Seite zu Seite ändern sollen - wie beispielsweise **Texte oder Untermenüs** - werden in der Designvorlage durch sogenannte **Marker oder Subparts** ersetzt. Dieses geschieht direkt im HTML-Quellcode. Der Trick ist denkbar einfach: der HTML-Parser von Typo ersetzt die Marker und Subparts in der Designvorlage durch den Teil, der nachher direkt im CMS-Interface eingegeben wird, und sendet ihn an den Browser.



EINFACHES DESIGN MIT LOGO, NAVIGATION, FARBSHEMA UND CONTENT

Marker werden meistens für einfache Anweisungen benutzt, wie zum Beispiel das dynamische Einbinden von Grafiken oder Hintergrundfarben. In diesem Fall bietet sich die Hintergrundfarbe der Tabellenzellen an. Marker werden in TypoScript mit drei Rauten am Anfang und Ende markiert: **###COLOR###**. Mit den Markern können sowohl **einzelne HTML-Attribute** in den Tags, als auch komplette Tags dynamisch generiert werden. Typo3 ersetzt diese dann mit Plain Text. Das ist auch interessant, um Tags **CSS-Klassen** zuzuweisen, die in einem externen Stylesheet definiert sind. Subparts hingegen werden dazu verwendet, **dynamischen Content** zu liefern, wie zum Beispiel Text oder Menüs. Der Subpart wird im HTML-Quellcode mit HTML-Kommentaren und drei Rauten eingebettet, also auf beiden Seiten der Stelle, an der der Subpart erscheinen soll: **<!--start of subpart: ###TEXT### -->Text<!-- end ###TEXT### -->**. Um Typo jetzt noch mitzuteilen, welchen Teil der Designvorlage es bearbeiten soll, muss am Anfang und Ende der Datei eine Anweisung dafür eingefügt werden: **<!-- ###DOCUMENT\_BODY### -->**.

```
<body>
<!--###DOCUMENT_BODY### -->
<table width="100%" border="0" cellpadding="4">
<tr>
<td colspan="3" bgcolor="#000000">###LOGO###</td>
</tr>
<tr>
<td height="10" colspan="3" bgcolor="###COLOR###" > </td>
</tr>
<tr>
<td width="200" valign="top" bgcolor="###COLOR###">
<!-- ###MENU### -->Menu<!-- ###MENU### -->
</td>
<td valign="top" bgcolor="#FFFFFF">
<!-- ###TEXT### -->Text<!-- ###TEXT### -->
</td>
<td width="150" valign="top" bgcolor="###COLOR###">
</td>
</tr>
</table>
<!--###DOCUMENT_BODY### -->
</body>
```

*Oben: Der HTML-Quelltext der Designvorlage*

Nach dem Anlegen der Seiten und der Inhalte muss nur noch das **Typo-Template** angelegt werden, um den Seiteninhalt an die richtigen Positionen im HTML-Template zu

setzen. Der in Typo3 verwendete Begriff "Template" ist etwas verwirrend, beim Typo-Template handelt es sich - im Gegensatz zum bisher gezeigten HTML-Template - um ein **Grundgerüst für den HTML-Parser**, das den Seitenaufbau steuert. Das Haupt-Template, das für alle Seiten samt Unterseiten gilt, wird auf der Rootebene des Pagetrees eingefügt. Um beispielsweise abweichende Farbschemata oder Grafiken für Unterseiten zu definieren, verwendet man neue Typo-Templates auf der jeweiligen Ebene im Pagetree, die nach dem Prinzip der Vererbung gezielt nur einzelne Werte überschreiben.

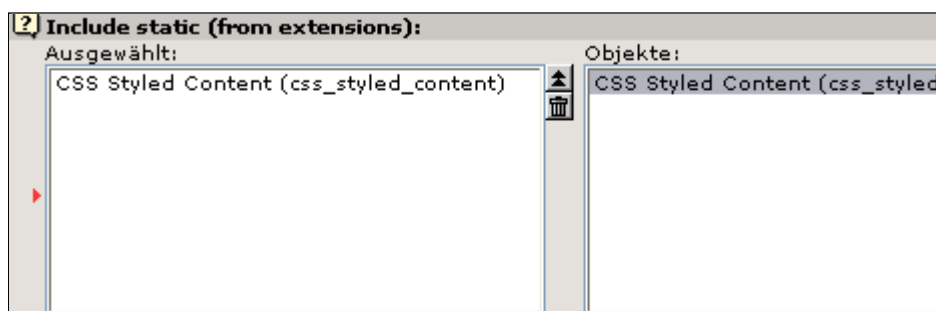
Die Templates benutzen TypoScript. Im Prinzip ist es keine echte Script-Sprache, sondern Anweisungen, die der Typo-Parser ausführt. Der folgende Code muss im Setup-Feld des Templates eingesetzt werden:

```
# Default PAGE object:
page = PAGE
page.typeNum = 0
# Ext. Stylesheet
page.stylesheet = fileadmin/templates/styles.css
# Menü
temp.menu = HMENU
temp.menu.1 = TMENU
temp.menu.1 {
    NO {
        ATagParams = style="font-family: Arial; text-decoration: none;"
        allWrap = |<br>
    }
}
page.10 = TEMPLATE
page.10 {
    #Haupt-Template einbinden
    template = FILE
    template.file = fileadmin/templates/template.html
    workOnSubpart = DOCUMENT_BODY
    # MARKER
    # Logo Grafik
    marks.LOGO = IMAGE
    marks.LOGO.file = fileadmin/templates/bulo-neg-plus.gif
    # Tabellenfarbe
    marks.COLOR = TEXT
    marks.COLOR.value = #FFFF00
    # SUBPARTS
    #Hauptnavigation
    subparts.MENU < temp.menu
    #Inhalt
    subparts.TEXT < styles.content.get
}
```

*Oben: Der TypoScript-Code im Setup des Templates*

Die Namen vor dem Gleichheitszeichen sind feste Variablennamen, die Typo benutzt, um bestimmte Elemente zu identifizieren. So werden im Code zum Beispiel Marker mit "marks", Subparts mit "subparts" und Templates mit "template" angesprochen.

Um in Typo3 **Cascading Style Sheets** zu verwenden, gibt es zwei verschiedene Ansätze, die am besten in Kombination arbeiten. Zum einen führt der Weg über eine sogenannte **Extension**, die man über den Extension Manager installieren muss. Die Extension heisst "CSS Styled Content", und wird im Template in der Box "Include static (from extensions)" integriert.



EINBINDUNG DES "CSS STYLED CONTENT"

Die zweite Variante führt über **externe Stylesheets**. Diese werden, wie im TS-Code oben

gezeigt, über "page.stylesheet = [Pfad/Datei]" eingebunden. Die Extension sorgt dafür, dass Typo3 seinen Content mit CSS formatiert, die in einem Stylesheet in dem Extension Directory abgelegt sind. Ideal nimmt man sich die benötigten Klassen aus diesem Stylesheet, und fügt diese in dem eigenen Stylesheet mit neuen Attributen ein. So wird zum Beispiel das Content Element Text "Normal" mit der Klasse "bodytext" definiert, und Image Text mit "csc-caption".

```
p.bodytext {
font-family: Arial, Helvetica, sans-serif;
font-size: 90%;
}
p.csc-caption {
font-family: Arial, Helvetica, sans-serif;
font-size: 80%;
}
h1 {
font-family: Verdana, Geneva, sans-serif;
font-size: 110%;
font-variant: small-caps;
font-weight: bold;
}
```

*Oben: Teil des externen CSS*

Weitere Möglichkeiten, Content-Elemente zu formatieren, sind die TypoScript-Anweisungen "**wrap**" oder "**params**" im Setup, die bestimmten Tags **Parameter** (zum Beispiel CSS-Klassen) übergeben, oder die Content Blöcke mit Tags umschliessen (zum Beispiel P-Tags mit CSS-Klassen). Auch **Style-Attribute** für Tags lassen sich so mühelos integrieren.



**PLUS**

- >Inhaltsverzeichnis
- >Webdesign
- >Marketing
- >Suchmaschinen
- >CMS

**Typo3: Designvorlagen und CSS**  
von Jörg Drzycimski

**Moderne Websites kommen heute nicht komplexes Design aus, und modernes H über Cascading Style Sheets. Das Open Typo3 bietet hierfür einen einfachen und Weg: externe Designvorlagen und Style**

Gerade in grossen Agenturen, wo ein Screen Vorlagen liefert, sind die Anforderungen an e

DIE FERTIGE WEBSITE

### Fazit

Typo3 ist ein CMS der Enterprise-Klasse, und trotzdem geeignet für jegliche Site-Grösse und -Anforderung. Selbst kleinere Sites können von den Eigenschaften profitieren, und die Umsetzung ist schnell erledigt. Bei grösseren Sites oder komplexen Designvorlagen ist etwas Übung und Dokumentation notwendig, der oben beschriebene Ansatz über Templates, Marker und Subparts ist aber exakt gleich.