

Ernesto Baschny

TYPO3 4.0: XHTML

Vor fünf Jahren beschloss das W3C-Konsortium, HTML 4 durch XHTML abzulösen. HTML 5 sollte es nie geben. Doch selbst heute sind noch viele Seiten in HTML 4.01 aufgebaut. Ein Grund: Viele Tools und Mentalitäten stammen noch aus der Zeit von HTML (wie auch TYPO3) und arbeiten daher nicht XHTML-konform. Mit TYPO3 ist bereits heute zum Teil eine XHTML-konforme Ausgabe möglich, mit TYPO3 4.0 wird dies noch weiter verbessert.

XHTML ist im Wesentlichen eine Umformulierung von HTML als XML-Dialekt. So gelten für XHTML die gleichen, strikteren syntaktischen Regeln wie für XML. XHTML 1.0 Transitional ist im Jahr 2000 als direkter Nachfolger von HTML 4.01 entstanden: Die Semantik wurde beibehalten, die Syntax wurde auf XML umgestellt. Durch eine Modularisierung der Auszeichnungssprache ist es dem W3C möglich, XHTML zu erweitern, ohne dass eine komplett neue Spezifikation notwendig ist: Alte Anwendungen arbeiten ohne die Erweiterungen weiter. Durch so genannte Namespaces (Namensräume) ist es möglich, XHTML in anderen XML-Dokumenten nahtlos zu integrieren. Somit ist die Speicherung der Webauszeichnung in XML-Datenbanken ohne Umwandlungen möglich. Umgekehrt ist es auch möglich, andere XML-Dialekte (z.B. SVG für vektorbasierte Grafiken) innerhalb von XHTML-Dokumenten direkt einzubinden. Die Verarbeitung von XHTML-konformen Dokumenten ist auch aus Software-Sicht einfacher. So können über vorhandene XML-Schnittstellen auch XHTML-Dokumente verarbeitet werden. Dies ist gerade bei Geräten mit geringer Speicherkapazität (Handys, PDAs etc.) von Vorteil. Außerdem können Browser einfacher, schneller und robuster programmiert werden. Durch XHTML wird die Zugänglichkeit erhöht – somit spielt die XHTML-Konformität auch im Hinblick auf Barrierefreiheit eine wesentliche Rolle. Barrierefreie Umsetzungen setzen in der Regel auf XHTML für die Strukturierung der Informationen und CSS für die gestalterischen Vorgaben (Layout, Positionierung, Farben, Schriften). Mehr Infos zu den Unterschieden zwischen HTML und XHTML liefert die Website von SELFHTML [1].

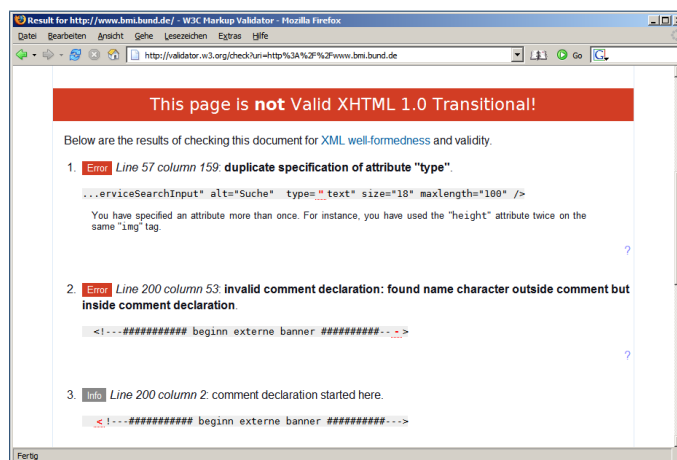
XHTML mit TYPO3 bisher

TYPO3 entstand zu einer Zeit, in der es XHTML noch nicht gab. Auch die „alte“ Browser-Generation konnte nur bedingt CSS-Anweisungen umsetzen, sodass man auf so genannte „inline“-Formatierungen angewiesen war (z.B.). Das ist nicht nur „unschön“ und nicht XHTML-konform, sondern bläht den Quelltext auch mit viel Redundanz unnötig auf. So wurde die Frontend-Ausgabe durchgängig nach der „HTML 4.01 Transitional“-Spezifikation realisiert. Wegen der ständig wachsenden Nachfrage nach XHTML, der Trennung von Content und Layout, aber auch wegen der Barrierefreiheit wurde in TYPO3 Version 3.6 schließlich die Möglichkeit hinzugefügt, TYPO3 für die strikte XHTML-Ausgabe einzustellen. Es gab die neue Konfigurationsmöglichkeit „config.doctype“. Bei Angabe einer XHTML-Variante wird TYPO3 angewiesen im Page-Header die passende XHTML-Präambel zu erzeugen. Auch weitere Stellen im TYPO3-Kern nutzen diese Information gezielt, um Inhalte entsprechend der Spezifikation auszugeben. Um von TYPO3 eine Ausgabe in XML-Syntax zu bekommen, ist folgende Zeile notwendig:

```
TYPOSCRIPT
```

```
config.xhtml_cleaning = all
```

Listing 1



Der Albtraum aller Webentwickler: Der W3C-Validator meldet „nicht valide“.

Im Einzelnen bewirkt dies folgendes:

- leere Elemente werden mit „/>“ abgeschlossen (z.B. „
“, „<img.../>“ usw.)
- die Tag- und Attributnamen werden stets in Kleinbuchstaben ausgegeben
- alle Attribute werden in Anführungszeichen gesetzt

Diese Transformationen geschehen in sämtlichen Ausgaben, die durch TYPO3 erzeugt werden. Allerdings wird die Wohlgeformtheit der Ausgabe nicht überprüft. Somit ist eine unkonforme Ausgabe trotz „cleanings“ möglich, z.B. bei Attributen oder Verschachtelungen, die in XHTML nicht erlaubt sind. Die „xhtml_cleaning“-Transformationen lassen sich im Übrigen auch für den Rich Text Editor (RTE) einschalten. Somit ist auch die redaktionelle Eingabe über den RTE zumindest syntaktisch XHTML-konform:

```
TYPOSCRIPT - PAGE SETUP
```

```
RTE.default.proc.entryHTMLparser_db.xhtml_cleaning = 1
```

Listing 2

Mit der Version 3.8 wurden im TYPO3-Kern die meisten Ausgaben bereits für XHTML angepasst. Es ist daher möglich, mit dieser Version zum großen Teil XHTML-konforme Webseiten zu gestalten. Die Erweiterung „qcom_htmlcleaner“ schafft hier zusätzlich Abhilfe, indem von der entsprechenden XHTML-Spezifikation nur noch gültige Tags und Attribute zugelassen werden. Die Ausgabe ist dann weitgehend konform, allerdings kann an einigen Stellen die Funktionalität verloren gehen.

Was die Zukunft bringt: XHTML mit TYPO3 4.0

In TYPO3 4.0 sollen die noch offenen Problemstellen korrigiert werden. Ultimatives Ziel ist die Konformität des TYPO3-Kerns mit XHTML 1.1. Im Einzelnen: Wer XHTML-konforme Seiten entwickelt,

sollte mit „css_styled_content“ statt „content (default)“ arbeiten. Eine strikte XHTML-Konformität ist bisher bei der Ausgabe der Inhaltselemente „Text mit Bild“ nicht möglich. Hier werden „align“-Attribute in Tabellen verwendet, was in XHTML 1.1 nicht mehr erlaubt ist. Durch die geplante Integration von „cron_cssstylingtext“ in „css_styled_content“ sind Tabellen für das Ausrichten der Bilder in diesen Elementen nicht mehr notwendig: Die Positionierung basiert vollständig auf CSS. Somit werden Elemente nicht nur XHTML-konform, sondern sogar barrierefrei.

Ein weiteres Problem verursacht die Erweiterung „indexed_search“, die als TYPO3-Systemerweiterung mitgeliefert wird. Bisher war eine XHTML-konforme Ausgabe nur durch Ändern des Quelltextes möglich, da dort HTML direkt eingebettet war. Die Ausgabe war daher nicht XHTML-konform. Ab TYPO3 4.0 wird auch dieses Problem behoben sein. Die Erweiterung wird mittels HTML-Templates gesteuert und ein XHTML 1.1 Template ist bereits enthalten.

In der XHTML 1.1 Spezifikation wurde bei den Elementen „a“, „applet“, „form“, „frame“, „iframe“, „img“ und „map“ das Attribut „name“ durch das generische, XML-konforme „id“ ersetzt. Ein Identifier muss dabei mit einem Buchstaben beginnen. Standardmäßig generiert TYPO3 vor jedem Inhaltselement einen Anker „“, wobei „{pid}“ die ID der Seite in Form eines Integer-Werts ist. Also gilt es hier gleich zwei Probleme zu lösen. In der nächsten TYPO3-Version gibt es eine Konfigurationsmöglichkeit, in der festgelegt werden kann, ob dem Anker ein Buchstabenpräfix vorangestellt werden soll. Auch wird der Anker dann nur noch mit dem „id“-Attribut generiert.

Ähnliche Probleme hatte man bisher bei Formularen. Hier wurde das Attribut „name“ im FORM-Element verwendet. Auch die Konfiguration mittels „accessibility=1“ war noch nicht ausgereift. Dies wird in der neuen Version besser umgesetzt, sodass auch Formulare komplett barrierefrei gestaltet werden können: Dazu gehören Fieldsets für Radio-Buttons mit Legend-Elementen und korrekt ausgezeichnete Label-Elemente für sämtliche Eingabefelder - natürlich alles XHTML-konform.

Ausblick

Selbst wenn TYPO3 4.0 einen großen Schritt in Richtung vollständiger XHTML-Konformität erzielen wird und auch wesentliche Möglichkeiten der Barrierefreiheit berücksichtigt werden, steht noch ein langer Weg bevor. Viele TypoScript-Konfigurationen und zahlreiche Erweiterungen müssen noch von der Konformität „überzeugt“ werden. Selbst heute werben viele Websites mit angeblicher Barrierefreiheit und XHTML-Konformität, was zum großen Teil nicht der Realität entspricht. Aufklärung ist in jedem Fall notwendig. TYPO3 und die Content Rendering Gemeinschaft können dabei eine wesentliche Rolle spielen.

Links und Literatur

- [1] SELFHTML, Unterschiede zwischen XHTML und HTML: <http://de.selfhtml.org/html/xhtml/unterschiede.htm>

DER AUTOR



Ernesto Baschny ist Mitglied der TYPO3 „Content Rendering Group“ und dadurch mitverantwortlich für die Umsetzung von Barrierefreiheit und XHTML-Konformität in TYPO3. Als IT-Entwickler der Firma cron IT GmbH verfügt er über langjährige und hervorragende Kenntnisse in den Bereichen Webtechnologien und OpenSource.