
Witt • Göbe: webEdition



Copyright (C) Open Source Press

Andreas Witt ▪ Thomas Göbe

webEdition

CMS ▪ eCommerce ▪ Online-Marketing

Alle in diesem Buch enthaltenen Programme, Darstellungen und Informationen wurden nach bestem Wissen erstellt. Dennoch sind Fehler nicht ganz auszuschließen. Aus diesem Grunde sind die in dem vorliegenden Buch enthaltenen Informationen mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Autor(en), Herausgeber, Übersetzer und Verlag übernehmen infolgedessen keine Verantwortung und werden keine daraus folgende Haftung übernehmen, die auf irgendeine Art aus der Benutzung dieser Informationen – oder Teilen davon – entsteht, auch nicht für die Verletzung von Patentrechten, die daraus resultieren können. Ebenso wenig übernehmen Autor(en) und Verlag die Gewähr dafür, dass die beschriebenen Verfahren usw. frei von Schutzrechten Dritter sind.

Die in diesem Werk wiedergegebenen Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. werden ohne Gewährleistung der freien Verwendbarkeit benutzt und können auch ohne besondere Kennzeichnung eingetragene Marken oder Warenzeichen sein und als solche den gesetzlichen Bestimmungen unterliegen.

Dieses Werk ist urheberrechtlich geschützt. Alle Rechte, auch die der Übersetzung, des Nachdrucks und der Vervielfältigung des Buches – oder Teilen daraus – vorbehalten. Kein Teil des Werkes darf ohne schriftliche Genehmigung des Verlags in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren), auch nicht für Zwecke der Unterrichtsgestaltung, reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Bibliografische Information Der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Copyright © 2012 Open Source Press, München
Gesamtlektorat: Dr. Markus Wirtz
Lektorat: Franz Mayer
Satz: Open Source Press (L^AT_EX)
Umschlaggestaltung: Olga Saborov
Gesamtherstellung: Kösel, Krugzell

ISBN 978-3-941841-79-6

<http://www.opensourcepress.de>

2 Kapitel

Konzeption

2.1 Vom Konzept zum Betrieb

Bei unserer Arbeit als Web-Entwickler und Projektmanager erleben wir immer wieder, dass Dienstleister zwar von agiler Software-Entwicklung und agilem Projektmanagement (z. B. nach Scrum¹) reden, jedoch gerade bei Web-Projekten die Frage nach dem einzusetzenden (Content Management) System bereits zu Beginn des Projektes beantwortet ist – ohne die genaue Zielstellung und somit die Anforderungen des Projektes zu kennen.

Berücksichtigen wir die schier unüberschaubare Zahl an Content Management Systemen, ist dies auch nicht verwunderlich. Für keinen Dienstleister wird es möglich sein, Know-how für alle Systeme und in allen Bereichen aufzubauen. Dennoch sollte sich die Beratung an den Grundsätzen

¹ Scrum ist ein agiles Management-Framework zur Entwicklung von Software unter stetiger Überprüfung der Arbeitsweise und des zu entwickelnden Produktes (Software). Vgl. Pichler, R. (2009): „Scrum – Agiles Projektmanagement erfolgreich einsetzen“, dpunkt.verlag, Heidelberg, S. 1–2.

des Web-Projekt-Managements orientieren, d.h. die Systemauswahl kann gezielt erst in der Phase der Grobkonzeption erfolgen. Zum besseren Verständnis befassen wir uns kurz mit den Stufen bzw. Phasen der *Website-Konzeption*. Hier gilt es zwei Herausforderungen zu bewältigen:

1. Es muss ein Projekt erfolgreich realisiert werden.
2. Es wird eine Software entwickelt.

Oftmals wird gerade der zweite Punkt unterschätzt bzw. vernachlässigt, dabei gilt: Eine Website ist auch eine Software, und das Content Management System ist das Framework, mit dem die Website realisiert wird. Diese Betrachtungsweise führt dazu, dass ein Projekt mit all seinen Eigenschaften² in den Prozess der (agilen) Softwareentwicklung integriert werden muss. Wie auch beim klassischen Projektmanagement gliedert sich die Website-Konzeption in verschiedene Phasen. Dabei kann die Anzahl und Benennung der Phasen je nach Modell variieren.

Abbildung 2.1:
Phasen der
Website-Konzeption

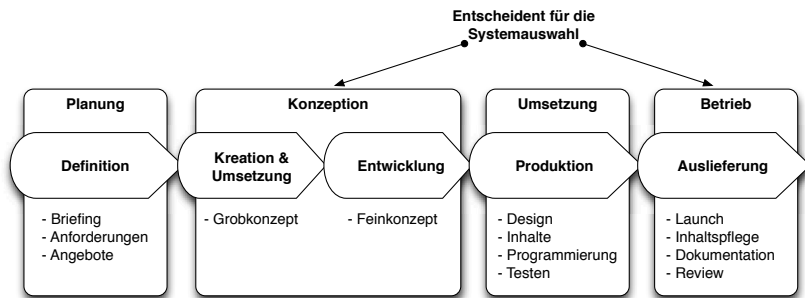


Abbildung 2.1 zeigt mögliche Phasen der Website-Konzeption. Unabhängig davon, ob man sich für vier Phasen³ oder fünf Phasen⁴ entscheidet, gibt es zwei entscheidende Punkte, die die Auswahl des passenden CMS beeinflussen – Konzeption und Betrieb. Bei der Erstellung des Grobkonzeptes werden Rahmenbedingungen, Zielstellungen und detailliertere technische Anforderungen erfasst, die unter anderem für die Auswahl des benötigten CMS von Bedeutung sind. Dabei werden jedoch (redaktionelle) Anforderungen für den laufenden Betrieb meist nicht berücksichtigt. Dies führt oft dazu, dass sich Probleme bei der Nutzung des CMS ergeben, die erst zum Ende des Projektes ersichtlich werden:

- ² Typische Projekteigenschaften sind: Zielvorgabe, Einmaligkeit, Zeitbegrenzung, Risiken und Komplexität.
- ³ Vgl. Jacobsen, J. (2011): „Website Konzeption – Erfolgreiche Websites planen, umsetzen und betreiben“, 6. Auflage, Addison-Wesley, München, S. 4.
- ⁴ Vgl. Grotenhoff, M.; Stylianakis, A. (2002): „Website-Konzeption – Von der Idee zum Storyboard“, Galileo Press GmbH, Bonn, S. 10.

- Die Benutzeroberfläche des CMS funktioniert nur in bestimmten Web-Browsern.
- Struktur und Funktionsweise von Eingabe-Dialogen sind für Redakteure nicht verständlich bzw. nicht dokumentiert.
- Es gibt keine Vorschauansichten vor der Veröffentlichung.
- Handbücher und Dokumentationen sind gar nicht oder nur unzureichend vorhanden.
- Redaktionelle Freigabeprozesse lassen sich nicht abbilden.
- Erstellte Inhalte (z. B. News) sind nach dem Veröffentlichenden nicht sofort auf der Website sichtbar (statisches vs. dynamisches CMS).
- Links (URLs) werden vom CMS vorgegeben und können nicht beeinflusst werden (Stichwort Suchmaschinenoptimierung).

Solche Probleme lassen sich teilweise umgehen, indem das oder die in Betracht kommenden Content Management Systeme mithilfe einer Demo-Installation getestet werden. Dabei ist es wichtig, auch mit Demo-Daten – sog. Demo-Sites, wie es sie beispielsweise zu diesem Buch gibt⁵ – zu arbeiten. Dadurch lassen sich gerade redaktionelle Probleme, aber auch das Potenzial eines CMS frühzeitig erkennen.

Eine weitere Möglichkeit, die oft bei größeren Projekten genutzt wird, ist das Hinzuziehen eines externen Beraters. Unsere Erfahrungen zeigen immer wieder, dass viele Unternehmen bereits kurz nach der Einführung eines Content Management Systems unzufrieden sind, da die o. g. Punkte nicht beachtet wurden. Als externe Berater unterstützen wir Unternehmen bei der Auswahl des für sie passenden CMS und des passenden Dienstleisters, indem wir den gesamten Prozess der Website-Konzeption begleiten und gemeinsam mit dem Kunden dessen Anforderungen und Ziele durchgehend berücksichtigen. Evaluieren wir ein IT-System, geht es nicht nur um die Frage, ob eine bestimmte Anforderung erfüllt wird, sondern auch, wie diese erfüllt wird – oft ist das *Wie* entscheidender als das *Ob*!

Anhand der Tabelle 2.1 wollen wir exemplarisch und stark vereinfacht am Beispiel des Content Management Systems webEdition zeigen, welche zehn Fragen bei der CMS-Auswahl am häufigsten gestellt werden und wie webEdition diese beantwortet.

Obwohl sich viele der Fragen pauschal mit Ja beantworten lassen und insgesamt 90% der Anforderungen erfüllt sind, ist diese Evaluierung nicht ausreichend aussagekräftig. Wie bereits erwähnt, ist die Frage nach dem *Wie* zu klären, um vollständig abschätzen zu können, inwieweit z. B. das CMS webEdition für ein Projekt geeignet ist.

⁵ <http://www.webedition-buch.de>

Tabelle 2.1:
Zehn Fragen an das
CMS webEdition

Anforderungen	erfüllt
Lässt sich unser Content-Modell mit dem CMS abbilden?	Ja
Können bestehende IT-Systeme an das CMS angebunden werden?	Ja
Ist das CMS skalierbar?	Nein
Können wir unseren Workflow mit dem CMS abbilden?	Ja
Lassen sich mit dem CMS mehrsprachige Inhalte verwalten?	Ja
Bietet das CMS Import- und Exportmöglichkeiten?	Ja
Entstehen durch das CMS Lizenzkosten?	evtl. webEdition Apps
Welche Deployment-Varianten gibt es?	statisch und dynamisch
Mit welchen Clients kann man auf das CMS zugreifen?	Web-Browser
Wird das CMS am Markt bestehen können?	Ja

Betrachten wir einige Fragen genauer:

Können bestehende IT-Systeme an das CMS angebunden werden?

webEdition bietet verschiedene Möglichkeiten, Daten und Assets aus Drittsystemen zu verarbeiten. Neben Import-Schnittstellen (XML und CSV) kann das webEdition SDK⁶ genutzt werden, um eigene Anwendungen (siehe Kapitel 6) zu entwickeln und somit auf externe Datenbestände komfortabel zuzugreifen. Oft bieten die zu integrierenden Drittsysteme Schnittstellen für den Datenaustausch, die direkt in webEdition Templates genutzt werden können. Bis auf die Import-Schnittstelle haben die genannten Lösungen den „Nachteil“, dass sie erst programmiert werden müssen – Out-of-the-Box-Lösungen stehen nicht zur Verfügung. Ausnahmen bilden hier die Bezahlösungen Paypal und Saferpay des Shop-Moduls (siehe Abschnitt 5.8) und die Integration der Web Shop Controlling Software ECONDA (siehe Abschnitt 5.8.5 auf Seite 280).

Ist das CMS skalierbar?

Unter skalierbar verstehen wir die Installation eines Software-Systems auf verschiedene Server wie z. B. mehrere Datenbank-Server oder Server Load Balancing-Systeme. webEdition bietet die Unterstützung solcher Lastverteilungssysteme aktuell nicht. Diese Tatsache ist unter

⁶ <http://documentation.webedition.org/wiki/de/webedition/developer-information/software-development-kit-sdk/start>

anderem auf die früheren Lizenz-Strategien – pro Domain eine Lizenz – zurückzuführen.

Können wir unseren Workflow mit dem CMS abbilden?

Das Content Management System webEdition bietet für die Abbildung von redaktionellen Arbeitsabläufen das Workflow-Modul (siehe Abschnitt 5.11). Damit lassen sich verschiedene Freigabeprozesse manuell anstoßen und abarbeiten. Werden vom Nutzer erzeugte Inhalte (z. B. Beiträge, Kommentare etc.) erstellt, kann vor der Veröffentlichung ein definierter Workflow automatisch gestartet werden.

Entstehen durch das CMS Lizenzkosten?

Grundsätzlich ist webEdition eine Open Source Software, und sowohl die Basisversion als auch die Module sind kostenfrei uneingeschränkt nutzbar. Etwas anders verhält es sich bei den webEdition Apps, die von Entwicklern sowohl kostenlos als auch kostenpflichtig angeboten werden können.

Wird das CMS am Markt bestehen können?

Bei der Auswahl eines IT-Systems handelt es sich oft um eine strategische und somit langfristige Entscheidung. Daher ist die bisherige und auch zukünftige Entwicklung des Systems zu betrachten. Im Anhang A.1 habe wir die lange Historie des CMS webEdition für Sie zusammengestellt. Die bisherigen Entwicklungen zeigen, dass der Übergang von der kommerziellen Lösung hin zur Open Source Software mit der Gründung des Vereins und der stetig wachsenden Anzahl an Partnern und Vereinsmitgliedern mehr als positiv verlaufen ist. Die gesamte Community engagiert sich stark bei der Weiterentwicklung des CMS – dazu gehört auch dieses Buch. Bei all den positiven Entwicklungen darf natürlich nicht vergessen werden, dass webEdition hauptsächlich von ehrenamtlichen Programmierern abhängig ist, die an der Weiterführung des Projektes arbeiten. Wir können zwar nicht in die Zukunft schauen, sind uns jedoch sicher, dass webEdition eine haben wird.

Nachdem wir uns mit einigen der in Tabelle 2.1 genannten Fragen genauer befasst haben, wird deutlich, welche verschiedenen Ausprägungen ein *Ja* haben kann. In den nachfolgenden Abschnitten wollen wir uns daher detaillierter mit den wichtigsten webEdition-Systemkonzepten befassen. Dadurch wird deutlich, welche Möglichkeiten, aber auch systembedingte Einschränkungen es gibt.

An dieser Stelle möchten wir noch darauf hinweisen, dass webEdition im Vergleich zu anderen CMS wie TYPO3 oder Joomla! keine Plugins, Extensions oder Themes benötigt, um ein breites Spektrum an Funktionalität zu bieten. Obwohl webEdition modular aufgebaut ist, werden alle Funktionsanforderungen von Websites durch sog. *we-Tags* und generische Template-

Entwicklung realisiert. Dadurch ist webEdition eher mit kommerziellen Systemen wie z. B. CMS Fiona,⁷ SixCMS⁸ etc. vergleichbar. Der individuelle Charakter von webEdition lässt sich am besten mit einem Framework vergleichen, das zwar zu Beginn eine höhere Lernkurve als *Click & Go-Systeme* erwartet, jedoch deutlich flexibler im Umgang mit speziellen Anforderungen agiert.

2.2 Datei- und Datenbank-basiert

webEdition ist nach unserer Definition (siehe Abschnitt 1.1) ein hybrides Web Content Management System. Hybrid bezieht sich dabei auf die Art der Seitenauslieferung (Deployment), die bei webEdition sowohl statisch als auch dynamisch erfolgen kann. Daher unterscheidet webEdition zwei Arten von Assets: Dokumente und Objekte.

Dokumente können Bilder, PDFs, Office-Dokumente, HTML-, CSS- oder JavaScript-Dateien etc. sein. webEdition-Dokumente werden als Dateien in Verzeichnissen organisiert und auf dem Webserver abgelegt. Dokumente, die auf Vorlagen basieren (z. B. HTML- und PHP-Dateien, aber auch CSS, JavaScript etc.), werden zusätzlich in der Datenbank gespeichert. Dadurch können Dokumente sowohl statisch als auch dynamisch ausgeliefert werden.

Objekte in webEdition lassen sich wie Objekte in einer Programmiersprache beschreiben. Wird das webEdition-Datenbank/Objekt-Modul genutzt, besteht die Möglichkeit, sog. *Klassen* anzulegen. Diese Klassen definieren die Feld-Struktur eines Assets (z. B. Titel, Beschreibung, Inhalt, Link, Grafiken etc.). Wird nun ein Objekt (Instanz einer Klasse) vom Redakteur angelegt, können die zuvor in der Klasse definierten Felder mit Inhalten frei von Layout-Informationen versehen werden. Im Gegensatz zu Dokumenten sollten Objekte als rein dynamische Assets verstanden werden. Zudem erfolgt die Inhaltserfassung bei Objekten nicht Vorlagen-basiert, sondern als Auflistung von Feldern (siehe Abbildung 2.2).

Objekte eignen sich besonders dann, wenn externe strukturierte Daten automatisiert in webEdition eingespielt (z. B. Artikel aus Warenwirtschaftssystemen, Stellenangebote, Kataloge über unterstützte Hardware etc.) oder Daten mehrfach in unterschiedlicher Darstellungstiefe angezeigt werden. Außerdem eignen sich Objekte, wenn Inhalte erfasst werden, die in Beziehung mit anderen Dokumenten, Kundendaten usw. stehen. Das können etwa Nutzerkommentare, Rezensionen und Veranstaltungen sein.

Bei den meisten Website-Projekten sind Dokumente die optimale Wahl. Bei Dokumenten erfolgt die Inhaltspflege Template-basiert, wodurch die Re-

⁷ <http://www.infopark.de/cms>

⁸ <http://www.six.de/sixcms/detail.php/14960>

Redakteure bereits während der Eingabe eine Art Vorschau erhalten. Gleichzeitig sind durch Vorlagen die Eingabemasken sprach- und sogar Domain-spezifisch steuerbar (siehe Abschnitte 2.3 und 2.4). Zusätzlich lassen sich Dokumente mit der Sidebar (siehe Abschnitt 4.5.4) verknüpfen und ermöglichen so zusätzliche Hilfestellungen bei der Inhaltspflege für Redakteure.

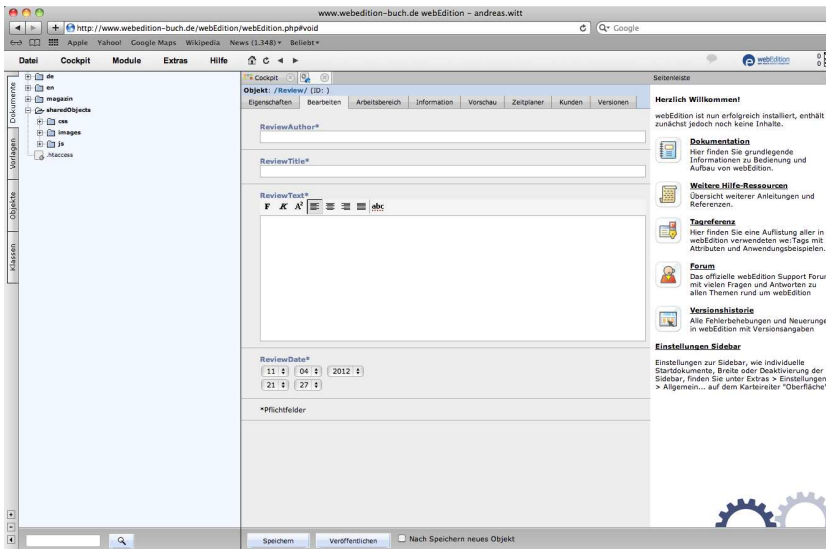


Abbildung 2.2:
Eingabemaske eines
webEdition-Objekts,
z. B. Rezension

Unabhängig von unseren Empfehlungen können Websites sowohl rein auf Dokumenten als auch ausschließlich auf Objekten basieren. Durch die Flexibilität beider Asset-Typen sind den Programmierern, Redakteuren und Website-Verantwortlichen keine Grenzen gesetzt.

2.3 Internationalisierung und mehrsprachige Websites

Obwohl wir Internationalisierung und mehrsprachige Websites in einem Abschnitt besprechen, handelt es sich hierbei um zwei eigenständige Ansätze. Als mehrsprachig bezeichnen wir Websites, die ihre Inhalte in mindestens zwei verschiedenen Sprachen anbieten.

Internationale Websites dagegen richten sich explizit an Nutzer in verschiedenen Ländern und sind mehr als eine reine Übersetzung der Inhalte. Häufig werden bei internationalen Websites eigene Top Level Domains mit der entsprechenden Länderkennung genutzt und sowohl inhaltlich als auch

strukturell unterschiedlich aufgebaut. Dabei ist es nicht unüblich, dass international ausgerichtete Websites auch mehrsprachig sind.

Bei der Realisierung mehrsprachiger und/oder internationaler Websites gilt es verschiedene Anforderungen zu berücksichtigen und Herausforderungen zu lösen, die je nach Projektumfang und Größe der Redaktionsteams variieren. Neben den organisatorischen und technischen Anforderungen bedarf es eines inhaltlichen Konzeptes, das unter anderem folgende Punkte berücksichtigen sollte:

- Welche Sprachen sollen angeboten werden?
- Ist der Umfang (Navigation, Texte, Grafiken etc.) für jede Sprachversion annähernd gleich?
- Wie sollten die verschiedenen Sprachen strukturell (URL-Struktur) aufgebaut werden?

Ein Punkt, der besonders viel Diskussionspotenzial bietet, ist die URL-Struktur, da sie entscheidenden Einfluss auf die Suchmaschinenoptimierung haben kann. Daher geben selbst Suchmaschinenbetreiber wie z. B. Google Hinweise⁹ auf mögliche Umsetzungsstrategien. Dabei werden meist die folgenden vier Vorgehensweisen genannt:

1. Für jede Sprachversion eine eigene länderspezifische Top Level Domain (für internationale Projekte geeignet).
2. Sprachversionen werden in sog. *Folders* abgelegt, z. B. `webedition.org/de/` (für mehrsprachige Projekte geeignet).
3. Sprachversionen werden als Subdomains umgesetzt, z. B. `de.webedition.org` (für mehrsprachige Projekte geeignet).
4. Sprachversionen werden durch URL-Parameter aufgerufen, z. B. `webedition.org/?lang=de` (nicht geeignet).

Soll eine Website ausschließlich mehrsprachig sein und sich an keine spezielle Nutzergruppe eines Landes richten, empfiehlt es sich, die Lösungsvariante 2 oder 3 zu wählen. Zeitgleich sollte eine generische Top Level Domain wie z. B. `.com`, `.org` etc. genutzt werden. Ausnahmen wären hier mehrsprachige Länder wie z. B. die Schweiz, bei der sich die verschiedenen Sprachen an die Nutzer in einem Land richten. Obwohl Variante 4 technisch ebenfalls umsetzbar wäre, ist diese Lösung mit Nachteilen (z. B. unterschiedliche Sprachversionen sind nicht über eine eindeutige URL erreichbar etc.) behaftet.

⁹ <http://support.google.com/webmasters/bin/answer.py?hl=de&answer=182192>

Nachdem das inhaltliche Konzept einer mehrsprachigen und/oder internationalen Website festgelegt wurde, müssen organisatorische und technische Anforderungen ermittelt werden. Bei größeren mehrsprachigen und bei internationalen Websites arbeiten oft unterschiedliche Redakteure an einer jeweiligen Sprachversion. Somit muss sichergestellt werden, dass der Zugriff (Arbeitsbereich – siehe Abschnitt 5.2) von Redakteuren eingeschränkt werden kann. Gleichzeitig ist es notwendig, dass das Backend (Arbeitsoberfläche) eines CMS selbst in verschiedenen Sprachen vorliegt oder zumindest Englisch als Backendsprache anbietet. webEdition bietet aktuell die Sprachen¹⁰ Deutsch, Englisch, Niederländisch, Finnisch, Polnisch, Russisch und Spanisch an. In Hinblick auf große Redaktionsteams oder extern arbeitende Redakteure ist die Abbildung von Workflows ebenfalls von Bedeutung. webEdition bietet hierfür ein eigenes Modul (weitere Informationen im Abschnitt 5.11) an, über das verschiedene Freigabeprozesse umgesetzt werden können und das zusätzlich mit einem internen Messaging-System mit Aufgabenverwaltung verknüpft ist.

Bei der Umsetzung mehrsprachiger Websites unterstützt webEdition die zweite Strategie (Sprachversionen werden in *Folders* verwaltet) optimal. Unabhängig davon, ob Inhalte innerhalb von webEdition als Dokumente oder Objekte verwaltet werden, sieht das CMS vor, dass es für jede Sprachversion ein eigenes Dokument und/oder Objekt gibt. Die einzelnen Sprachversionen werden dann über die Dokumenteneigenschaften (siehe Abbildung 2.3, Bereich *Meta-Infos*) miteinander verknüpft.

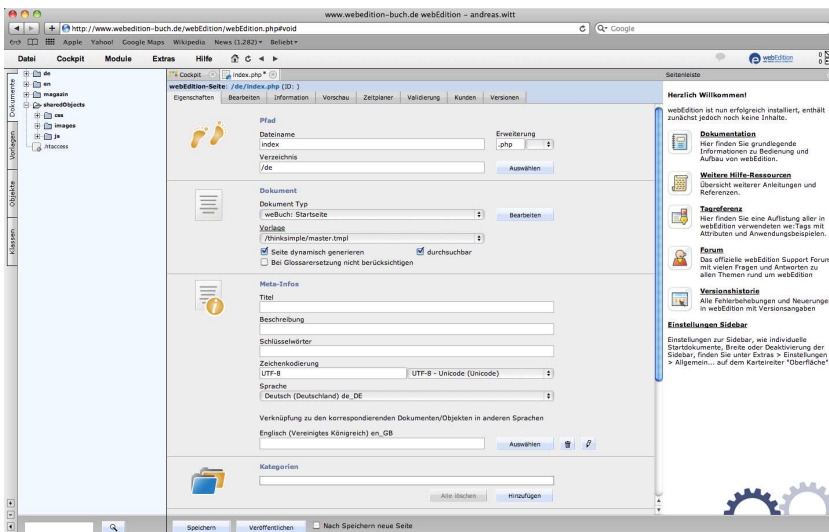


Abbildung 2.3:
Verknüpfung von
Sprachversionen in
webEdition

¹⁰ Nicht alle der genannten Sprachen sind schon vollständig übersetzt. Hier bedarf es der Unterstützung der Community. Gleiches gilt für ein Angebot weiterer Sprachen.

Basierend auf dieser Verknüpfung und der Zuweisung einer Sprachversion, werden unter Verwendung von webEdition Tags automatisch Sprachumschaltungen (<we:listview type="languageLink">), Formularübersetzungen (<we:ifPageLanguage>), Datumsformatierungen (<we:date format="d.m.Y" />) usw. für die jeweilige Sprache vorgenommen.

Formulareingaben wie z. B. Land oder Sprache werden mit webEdition ISO-kodiert gespeichert und auf der Website ebenfalls automatisch übersetzt ausgegeben. Dadurch wird ermöglicht, dass je nach Sprache die korrekte Übersetzung – auch im webEdition Backend für Redakteure – angezeigt wird. Da webEdition auch sprachabhängige (Bestätigungs-)E-Mails versenden kann, empfiehlt es sich, Kundendatensätze ebenfalls mit einer Sprache zu verknüpfen. Dadurch ist es möglich, einem Kunden, der auf einer englischen Website eine Bestellung im Online-Shop aufgegeben hat, auch eine englische Bestellbestätigungs-E-Mail zuzusenden (weitere Informationen im Abschnitt 5.8).

Bei der Programmierung mehrsprachiger Websites empfiehlt es sich, für alle Sprachversionen dasselbe Template zu nutzen. Durch Verwendung der entsprechenden webEdition Tags ist dies auch problemlos möglich und vereinfacht zudem die spätere Wartung und Anpassung. Gleichzeitig können die webEdition Tags auch für den Bearbeitungsmodus für Redakteure genutzt werden. Dies ermöglicht, dass Beschriftungen von Feldern und Hinweistexten ebenfalls mehrsprachig angeboten werden können (dies ist aktuell nur bei Dokumenten und nicht bei Objekten möglich). Zudem lassen sich auch Eingabefelder für bestimmte Sprachversionen ausblenden, falls diese Informationen für eine Sprachversion nicht benötigt werden.

Problematisch bei der redaktionellen Pflege von mehrsprachigen Websites ist die Übersetzung bzw. Benennung von Formularfeldern. Diese werden oftmals vom Programmieren „fest“ ins Template geschrieben. Um dieses Problem zu umgehen, nutzen wir bei webEdition-Projekten aktuell eine PHP-Datei `languageString.php`, die ein Übersetzungs-Array enthält und vom Redakteur bearbeitet werden kann.

```
<?php
    $GLOBALS['translationWorkspaceID'] = (int) 1; // ID des \
Verzeichnisses "/de"
    $GLOBALS['translationNavigationParentID'] = (int) 1; // ID des \
Ordners "/Deutsch" im Navigations-Tool
    $GLOBALS['translationRegistrationLabelForename'] = (string) 'Vorname\
';
    $GLOBALS['translationRegistrationLabelSurname'] = (string) 'Nachname\
';
    $GLOBALS['translationRegistrationLabelStreet'] = (string) 'Straße';
    $GLOBALS['translationRegistrationLabelNumber'] = (string) 'Haus-Nr.';
    $GLOBALS['translationRegistrationLabelZipcode'] = (string) 'PLZ';
    $GLOBALS['translationRegistrationLabelCity'] = (string) 'Ort';
    $GLOBALS['translationRegistrationLabelCountry'] = (string) 'Land';
    $GLOBALS['translationRegistrationLabelPhone'] = (string) 'Telefon';
```

```

$GLOBALS['translationRegistrationLabelFax'] = (string) 'Fax';
$GLOBALS['translationRegistrationLabelEmail'] = (string) 'E-Mail-\
Adresse';
?>

```

Diese Dateien werden pro Sprachversion angelegt und direkt zu Beginn eines Templates eingebunden (siehe nachfolgende Code-Schnipsel). Wir nutzen für das Übersetzungs-Array die PHP-Variable `$GLOBALS`, um direkt mit dem `webEdition` Tag `<we:var type="global" name=" />` auf die Inhalte des Arrays zugreifen zu können. Das ermöglicht die Ausgabe und Manipulation der Variablen ohne weitere PHP-Programmierung. Eine etwas elegantere Alternative wäre ein Übersetzungs-Tool auf Basis einer `webEdition` App, die es aktuell leider noch nicht gibt.

```

<we:ifWorkspace path="/en/" doc="top">
  <we:include type="document" path="/en/languageString.php" />
<we:else/>
  <we:include type="document" path="/de/languageString.php" />
</we:ifWorkspace>

```

Wie der Ausschnitt aus der `languageString.php` zeigt, kann diese Datei auch für die korrekte Einbindung der Navigation genutzt werden. Dazu kann im `webEdition` Navigations-Tool, das wir in Abschnitt 4.4 noch detailliert vorstellen werden, die Navigation für die unterschiedlichen Sprachen in Verzeichnissen (siehe Abbildung 2.5) gegliedert werden. Das jeweilige Root-Verzeichnis (z. B. Deutsch) wird dann über die ID ermittelt und im Template ausgegeben.

Aktuell gibt es bei der Realisierung mehrsprachiger Websites mit `webEdition` kaum Einschränkungen. Eine Ausnahme bildet hier das `webEdition` Shop-Modul. Obwohl Artikel bzw. Produkte auf Dokumenten und/oder Objekten basieren und auch Übersetzungen der Artikel über die Sprachzuweisung verknüpft werden können, ist eine inhaltliche Übersetzung von Artikeln, die sich im Warenkorb befinden, aktuell nicht möglich. Außerhalb des Warenkorbs funktioniert auch hier die Übersetzung und Sprachumschaltung problemlos.

2.4 Multi-Domains – Möglichkeiten und Einschränkungen

Für die Verwendung zusätzlicher Domains sprechen gleich mehrere Gründe. Für das Online-Marketing und den Vertrieb kann es sinnvoll sein, bestimmte Produkte und/oder Dienstleistungen auf eigenen Domains beispielsweise als Microsites zu betreiben. Neben der lokalen Ausrichtung der Produkte (z. B. Hotels von Hotelketten in verschiedenen Orten) kann zudem die unterschiedliche Zielgruppenansprache von Bedeutung sein. Auch die

Expansion auf den internationalen Markt kann für den Einsatz von sog. Multi-Domains sprechen. So verschieden die Gründe sind, sind auch die potenziellen Lösungen. Neben der technischen Realisierung (mehrere Server, unterschiedliche IP-Adressen etc.) von Multi-Domain-Ansätzen sind noch weitere Aspekte wie die Suchmaschinenoptimierung, der redaktionelle Aufwand, das Social Media Marketing usw. zu berücksichtigen.

Im Bereich der Mehrsprachigkeit von Multi-Domain-Ansätzen ist auch oft von *SEO-Domain-Konzepten* die Rede. Dabei werden für die Realisierung von mehrsprachigen bzw. international ausgerichteten Web-Projekten die drei verschiedenen Konzepte *Multiple Domains*, *Subdomains* und *Folders* diskutiert. Unabhängig davon sind für den Einsatz von Multi-Domains generell verschiedene Aspekte zu berücksichtigen, die in der Tabelle 2.2 beispielhaft aufgelistet sind.¹¹

Tabelle 2.2:
Multi-Domain-
Ansätze

	Multiple Domains	Subdomains	Folders
Möglichkeiten GEO Targeting	hoch	mittel	niedrig
Anpassungen Design & Site- Struktur	hoch	mittel	niedrig
Website- Steuerung	sehr schwierig	schwierig	einfach
SEO Link- Struktur	mehrere Domains	mehrere Subdomains	intern
Beschränkungen bei SERPs ¹²	keine	fast keine	Anzahl begrenzt
Domain Au- thority, -Trust und -Strength	keine Vererbung	teilweise Vererbung	Vererbung
Sitelink- Unterstützung	Nein	Ja	Ja

Wie der Tabelle 2.2 zu entnehmen ist, bieten die unterschiedlichen Strategien verschiedene Vorteile. Im Bereich der Website-Gestaltung und des strukturellen Aufbaus bieten Multiple und Subdomains die größte Flexibilität. Bei der Vererbung von SEO-Indikatoren wie Authority, Trust und Strength sind es die Folders. Neben den genannten spielen jedoch noch weitere, oft unterschätzte Faktoren eine wichtige Rolle. So ist der redaktionelle Aufwand für Multiple Domains größer als bei Folders. Zudem ist das

¹¹ Quelle: <http://www.webseoanalytics.com/blog/multiple-domains-vs-subdomains-vs-folders-in-seo/>

¹² Search Engine Result Page (Ergebnisseite von Suchmaschinen)

benötigte (Spezial-)Wissen für bestimmte Konzepte, gerade im Bereich der Suchmaschinenoptimierung, nicht zu unterschätzen – ein Multi-Domain-Projekt besteht nicht nur in weiteren Top Level Domains für eine Hauptdomain, sondern es ist eine eigenständige Strategie mit speziellen (technischen) Anforderungen!

Unabhängig von der technischen Basis, mit der Multi-Domain-Strategien umgesetzt werden, hat es sich bei verschiedenen Projekten bewährt, Domains unabhängig vom Server-Hosting zu verwalten. Neben spezialisierten Anbietern, die sich auf die Domainverwaltung konzentriert haben, bieten auch bekannte Hoster Pakete, die eine reine Domainverwaltung ohne Server und Webpakete ermöglichen. Neben dem Vorteil der zentralen Verwaltung können dadurch Domains gezielt auf unterschiedliche physische Server verteilt werden.

Als technische Basis können mit webEdition nur eingeschränkt Multi-Domain-Strategien umgesetzt werden. Dieser Umstand ist der kommerziellen Historie von webEdition geschuldet, bei der das CMS ehemals pro Domain (pro Domain eine webEdition-Installation) lizenziert wurde. Dennoch lassen sich unter den nachfolgenden Voraussetzungen mit webEdition Multi-Domain-Strategien realisieren:

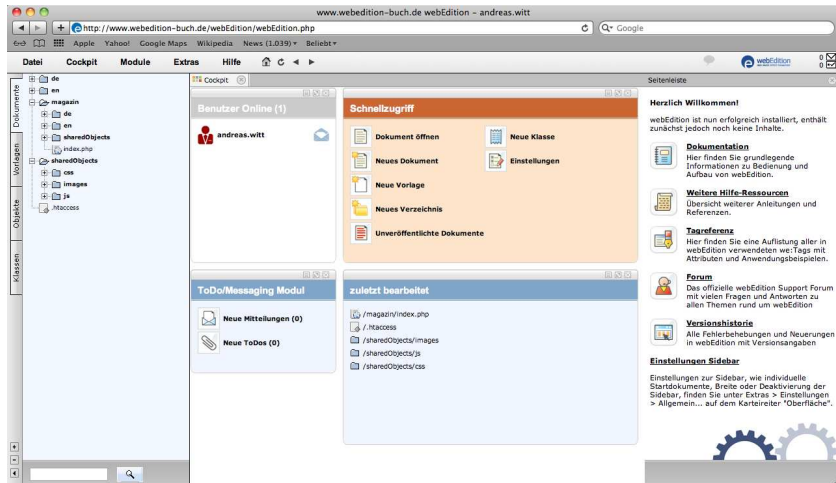
- Idealerweise nutzen alle Domains denselben Server.
- Es gibt eine Hauptdomain, über die das CMS bedient wird (z. B. `webedition-buch.de/webEdition`).
- webEdition-Module (z. B. Kundenverwaltung, Shop, Glossar etc.) können nur Domain-übergreifend genutzt werden.
- Alle Domains nutzen das gleiche Thumbnail-Verzeichnis.

Trotz der genannten Einschränkungen bietet webEdition etwa mit dem Tag `<we:ifIsDomain>`¹³ Unterstützung bei der Umsetzung von Multi-Domain-Strategien. In Verbindung mit einer entsprechenden Verzeichnisstruktur und der Verwendung von `.htaccess`-Dateien lassen sich verschiedene Projekte mit einer webEdition-Installation umsetzen. Abbildung 2.4 zeigt am Beispiel von `webedition-buch.de` und `webedition-magazin.de`, wie eine entsprechende Verzeichnisstruktur aussehen kann.

Bei diesem Beispiel haben wir uns für `webedition-buch.de` als Haupt- und Verwaltungs-Domain entschieden. Somit ist das webEdition Backend für beide Websites über `webedition-buch.de/webEdition/` erreichbar. Alle Inhalte für die Website `webedition-buch.de` werden direkt im Root-Verzeichnis mehrsprachig (getrennt in die Verzeichnisse `/de` und `/en`) abgelegt. Alle Inhalte für die Website `webedition-magazin.de` befinden sich im Verzeichnis `/magazin`.

¹³ <http://www.webedition.org/de/webedition-cms/dokumentation/tag-referenz/ifisdomain>

Abbildung 2.4:
Verzeichnisstruktur
im CMS webEdition
für Multi-Domain-
Projekte



Auf Grundlage der Verzeichnisstruktur kann durch `.htaccess`-Dateien die Weiterleitung der unterschiedlichen Domains in die entsprechenden Verzeichnisse erfolgen.

```
Options +FollowSymlinks
RewriteEngine on
```

```
# Magazin ohne www auf www.webedition-magazin.de/magazin/ umleiten
RewriteCond %{HTTP_HOST} ^webedition-magazin\.de$ [NC]
RewriteRule ^(.*)$ http://www.webedition-magazin.de/magazin/$1 [R=301,L\
]
```

```
# Magazin mit www auf /magazin/ weiterleiten
RewriteBase /
RewriteCond %{HTTP_HOST} ^www\.webedition-magazin\.de$ [NC]
RewriteRule ^$ /magazin/$1 [R=301,L]
```

```
# Buch ohne www auf www umleiten
RewriteCond %{HTTP_HOST} ^webedition-buch\.de$ [NC]
RewriteRule ^(.*)$ http://www.webedition-buch.de/$1 [R=301,L]
```

In Verbindung mit dem bereits genannten `<we:ifIsDomain>` lässt sich die Ausgabe von Grafiken, Inhalten und der Navigation in Templates steuern. Dazu wird mit dem webEdition Navigations-Tool die Navigation der einzelnen Websites ähnlich strukturiert wie die Dokumentenverzeichnisse zuvor (siehe Abbildung 2.5).

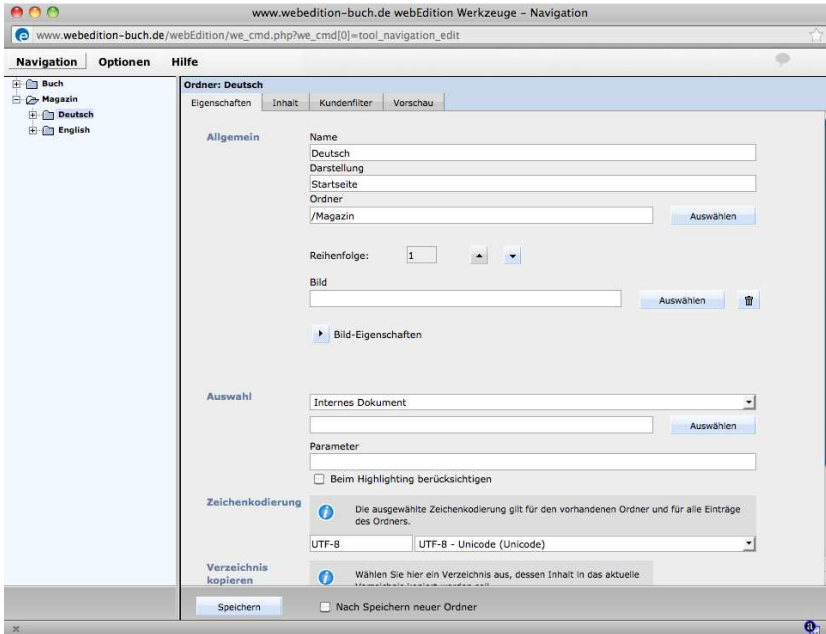


Abbildung 2.5:
Navigationsstruktur
im CMS webEdition
für Multi-Domain-
Projekte

Das Rechte- und Zugriffsmanagement (siehe Abschnitt 5.2) von webEdition erlaubt es, den Zugriff für Redakteure auf einzelne Verzeichnisse zu beschränken. Somit kann sichergestellt werden, dass Redakteure der Website `www.webedition-magazin.de` nur auf das Verzeichnis `/magazin` Zugriff haben. Dieses Verhalten lässt sich jedoch nicht auf die webEdition-Module (weitere Informationen zu webEdition-Modulen in Kapitel 5) anwenden. Sofern ein Redakteur Rechte für ein Modul besitzt, kann er dort auch alle Inhalte sehen und bearbeiten. Neben dem Navigations-Tool könnte dies auch gerade für die Kundenverwaltung und das Shop-Modul problematisch werden. Hier zeigt sich die fehlende Mandantenfähigkeit von webEdition.

Zusammenfassend lässt sich festhalten, dass webEdition teilweise Multi-Domain-Projekte unterstützt. Unter Verwendung von Multi-Domain-Zertifikaten lassen sich auch SSL-geschützte Projekte realisieren. Neben den vorgeschlagenen Lösungsansätzen können Multi-Domains durch Einsatz von statischen Seiten (siehe Abschnitt 2.2) automatisiert auf verschiedene Server verteilt werden, indem eine Replikation¹⁴ einzelner Verzeichnisse (z. B. `/magazin`) per Cronjob vom Quell- auf den Zielserver erfolgt. Wobei

¹⁴ Eine Replikation in der einfachsten Form ist die Erstellung einer Kopie der Quelldaten auf ein entferntes Zielsystem.

dies nur bei sehr kleinen und einfachen Webseiten sinnvoll ist und eine Manipulation der webEdition Link-Struktur erfordert.

2.5 Suchmaschinenoptimierung

Suchmaschinenmarketing (*Search Engine Marketing*, kurz: SEM) ist bereits seit einigen Jahren fester Bestandteil des Online-Marketings und unverzichtbar für eine erfolgreiche Online-Strategie. Das Suchmaschinenmarketing gliedert sich in die Bereiche *Suchmaschinenoptimierung* (*Search Engine Optimization*, SEO) und *Suchmaschinenwerbung* (*Search Engine Advertising*, SEA). Mit dem SEM wird versucht, die Sichtbarkeit der eigenen Webseite in den Ergebnislisten (*Search Engine Result Pages*, SERP) der Suchmaschinen zu verbessern, um somit mehr Aufmerksamkeit und Traffic zu erhalten. Die Suchmaschinenoptimierung versucht diese Ziele mit einem verbesserten Listing in den sog. organischen Suchergebnissen zu erreichen.

Obwohl alle großen Suchmaschinenbetreiber Richtlinien für die Suchmaschinenoptimierung anbieten, wird oft versucht, mit nicht zulässigen Maßnahmen das organische Ranking zu beeinflussen, was im schlimmsten Fall mit einer Abstrafung (und eventuell Ausschluss aus dem Index) durch den Suchmaschinenbetreiber enden kann. Anhand der Google-Richtlinien¹⁵ werden wir nachfolgend zeigen, wie webEdition Maßnahmen zur Suchmaschinenoptimierung unterstützt.

Bei der Suchmaschinenoptimierung unterscheiden wir zwischen der Indexierung einer Website und der Darstellung der Website in der Ergebnisliste der Suchmaschine (Content-Optimierung). Zur besseren Indexierung von Websites gibt es verschiedene Maßnahmen, die abhängig von der Website genutzt werden können. Ein sehr wichtiges Werkzeug dabei sind die Google Webmaster Tools.¹⁶ Deren Nutzung setzt voraus, dass sich der Nutzer als Inhaber der Website identifiziert. Dies kann durch einen Meta-Tag auf der Startseite erfolgen und lässt sich mit jedem CMS einfach einbauen.

```
<meta name="google-site-verification" content="YXF1xMmQ2j2i7A..." />
```

2.5.1 Optimale Indexierung für webEdition-Websites

Um die Indexierung von (neuen) Websites zu unterstützen, können sog. XML-Sitemaps¹⁷ über die Google Webmaster Tools eingereicht werden. Dies können einfache Textdateien mit Links zu den einzelnen Seiten, aber auch

¹⁵ <http://support.google.com/webmasters/bin/answer.py?hl=de&answer=35769>

¹⁶ <https://www.google.com/webmasters/tools/>

¹⁷ <https://support.google.com/webmasters/bin/answer.py?hl=de&answer=156184>

XML-Dateien mit zusätzlichen Meta-Informationen sein. Das nachfolgende Code-Beispiel zeigt, wie sich mit webEdition sehr einfach Listen über die Dokumente und Objekte erzeugen lassen, die dann als Sitemap eingereicht werden können.

```
<we:comment>Textdatei mit Linkliste ueber alle webEdition Dokumente \
ohne die Ausgabe von index.php</we:comment>
<we:listview type="document" order="we_id" languages="de_DE" \
hidedirindex="true">
  <we:repeat>
    <we:field type="text" name="WE_PATH" to="global" nameto="\
docPath"/>
    <?php echo "http://www.webedition-buch.de".$GLOBALS['docPath\
']."\n"; ?>
  </we:repeat>
</we:listview>

<we:comment>Textdatei mit Linkliste ueber webEdition Objekte der Klasse\
"Rezension" (ID: 1) mit SEO-URLs</we:comment>
<we:listview type="object" classid="1" languages="de_DE" objectseourls\
="true" hidedirindex="true">
  <we:repeat>
    <we:field type="text" name="WE_PATH" to="global" nameto="\
objPath"/>
    <?php echo "http://www.webedition-buch.de".$GLOBALS['objPath\
']."\n"; ?>
  </we:repeat>
</we:listview>
```

Bei der Verwendung von Kategorien (siehe Abschnitt 4.5.2) und Dokumententypen (siehe Abschnitt 4.5.1) werden die in der Auflistung enthaltenen Links gefiltert. Gleichzeitig lassen sich dadurch für unterschiedliche Kriterien wie Kategorien oder Dokumententypen verschiedene Sitemaps (z. B. Text oder XML) in den Webmaster Tools einreichen.

Bevor jedoch die ersten Sitemaps eingereicht werden, sollte eine Strategie für die URL-Struktur vorhanden sein. Dass es sich hierbei um ein komplexes Thema handelt, haben wir bereits im Abschnitt 2.4 über Multi-Domain-Projekte gezeigt. Obwohl es für den generellen Aufbau von internationalen und mehrsprachigen Websites einige Empfehlungen gibt, ist ein URL-Schema für jedes Projekt individuell zu entwickeln. Dieser Individualität wird auch webEdition gerecht, unabhängig davon, ob Inhalte als Dokumente oder Objekte erfasst werden. Basis für die Umsetzung von sog. „suchmaschinenfreundlichen URLs“ oder auch „sprechenden URLs“ mit webEdition sind die Einstellungen für SEO-Links, wie sie in Abbildung 2.6 zu sehen sind. Diese allgemeinen Einstellungen filtern z. B. ein `index.php` oder `index.html` aus dem Link heraus. Dieses Verhalten lässt sich für bestimmte webEdition Tags auch mit dem Attribut `hidedirindex="true"` erzwingen, wie wir es im obigen Code-Beispiel bereits genutzt haben.

Eine weitere erwähnenswerte Einstellung ist die Auswahl eines Fehler-Dokuments, das angezeigt wird, falls eine URL nicht aufgelöst werden kann. Das verbessert zum einen die Usability für den Nutzer und verhindert zum anderen eine 404-Fehlermeldung des Servers.

Abbildung 2.6:
Einstellungen für
SEO-Links

www.webedition-buch.de webEdition

Oberfläche | Dateiweiterungen | Editor | Proxy Server | Erweitert | System | SEO-Links
 Modulaktivierung | Sprachen | Länder | Fehlerbehandlung | Backup | Validierung | E-Mail
 Benachrichtigungen | Formmail | Versionierung

Versteckte DirectoryIndex-Dateinamen in der Ausgabe

Nach einer Änderung muss ein Rebuild durchgeführt werden (z.B. Navigation Cache, Objekte usw.)

der Navigation true

von Links aus dem WYSIWYG-Editor true

DirectoryIndex-Dateinamen (Komma-separiert, einschl. Datei-Extensions, z.B. 'index.php,index.html')

Für die Tags <we:a>, <we:href>, <we:link>, <we:linklist>, <we:listview> und <we:url> kann das Attribut 'hidedirindex' verwendet werden.

Voreinstellung für Tags true

Erzeuge Objekt SEO-Urls

in der Navigation true

bei Links aus dem WYSIWYG-Editor true

Für die Tags <we:link>, <we:linklist>, <we:listview>, <we:object> kann das Attribut 'objectseourls' verwendet, oder die folgende Voreinstellung vorgenommen werden:

Voreinstellung für Tags true

URLEncode die SEO-Links false

Darstellung innerhalb von webEdition

Werden DirectoryIndex-Dateinamen innerhalb von webEdition versteckt bzw. Objekt SEO-Urls verwendet, so kann webEdition diese nicht als interne Links erkennen und ein Klick auf den Link öffnet nicht mehr das Bearbeitungsfenster. Mit den folgenden Optionen kann das Erzeugen solcher Links (bis auf die Navigation) im Editmode und der Vorschau unterdrückt werden.

Im Editmode unterdrücken false

In der Vorschau unterdrücken false

Fehlerseite für nicht existierende Objekte

unterdrücke 404 not found true

Dokumente in webEdition werden wie Dateien auf einem Computer in Verzeichnissen organisiert, die der Redakteur nach Belieben benennen kann. Dadurch können individuelle URL-Strukturen (z. B. `/de/unternehmen/anfahrt.php` oder `/de/unternehmen/anfahrt/index.php`) abgebildet werden. Gleichzeitig können beim Anlegen von webEdition-Dokumenten Verzeichnisse vorgegeben werden, in denen das Dokument zu speichern ist. Auch das Erstellen von dynamischen Verzeichnissen, wie z. B. das aktuelle Jahr, ist möglich. Das kann besonders bei Newsarchiven oder Veranstaltungen sinnvoll sein (z. B. `/de/unternehmen/news/2012/`). Jedoch ist bei webEdition-Dokumenten zu berücksichtigen, dass zwar Verzeichnisse vorgegeben werden können, der Name eines Dokuments jedoch immer vom Redakteur festgelegt wird.

Etwas anders verhält es sich bei webEdition-Objekten (siehe Abschnitt 5.3). Hier kann der Administrator/Entwickler nicht nur die URL-Struktur granular bestimmen, sondern auch die Objektnamen automatisch vom CMS erzeugen lassen. Für die automatische Objektnamenvergabe stehen unter anderem folgende Werte zur Verfügung:

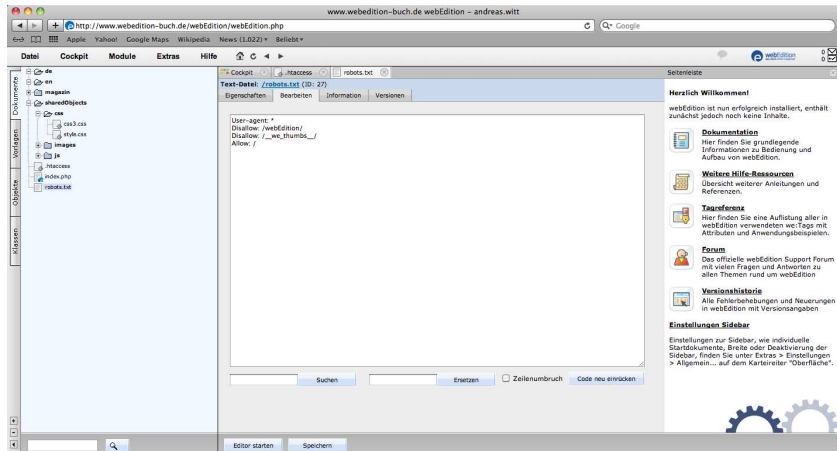
- eindeutiger Name (String mit definierter Länge)
- Datumsangaben (z. B. Jahr, Tag, Stunden etc.)
- eine ID
- frei wählbarer Text

Im Unterschied zu webEdition-Dokumenten hat der Name bei webEdition-Objekten aktuell keinen Einfluss auf die URL-Struktur. Hierfür stehen dem Administrator/Entwickler neben den bereits genannten Werten für die Objektnamen noch weitere zur Verfügung:

- dem Objekt zugewiesenes Land
- dem Objekt zugewiesene Sprache
- Locale des Objektes
- bis zu drei frei wählbare Objektfelder (URL-Feld-1, URL-Feld-2 und URL-Feld-3)

Somit bietet webEdition verschiedene Möglichkeiten, URL-Strukturen mit unterschiedlichen Freiheitsgraden für Redakteure umzusetzen. Daneben ist auch das Erstellen und Verwalten von weiteren, für die Indexierung von Suchmaschinen wichtigen Dateien wie `robots.txt` und `.htaccess` möglich. Dazu bietet das CMS eigenständige Dokumententypen, die im Bearbeiten-Modus ein unformatiertes Texteingabefeld (siehe Abbildung 2.7) anbieten, um die Konfigurationen komfortabel zu bearbeiten.

Abbildung 2.7:
Verwalten von
robots.txt und
.htaccess in
webEdition



Kehren wir zu mehrsprachigen Websites zurück, ist das relativ neue HTML-Attribut `hreflang` von Bedeutung. Mit diesem Attribut lassen sich die zu einer Seite existierenden Sprachversionen auszeichnen. Die Auszeichnung kann dabei auf Sprachebene

```
<link rel="alternate" hreflang="en" href="www.webedition-buch.de/en/ />
```

oder auf Sprach- und Gebietsebene

```
<link rel="alternate" hreflang="de-CH" href="ch.webedition-buch.de/de/" \
/>
```

erfolgen. Somit ist es den Suchmaschinen möglich, die für den Suchenden optimale Sprachversion anzubieten. Die Auflistung der vorhandenen Sprachversionen erfolgt auf Seitenebene und wird im `<head>`-Bereich jeder Seite eingefügt. Mit webEdition lässt sich dies sehr einfach anhand des nachfolgenden Code-Beispiels umsetzen. Dabei werden alle mit einem Dokument verlinkten Sprachversionen aufgelistet und entweder auf Sprachebene oder auf Sprach- und Gebietsebene ausgegeben. `<we:ifFound>` stellt sicher, dass keine Ausgabe erfolgt, wenn nur eine Sprachversion vorhanden ist.

```
<we:listview type="language" order="Locale" pagelanguage="top" \
showself="false" objectseourls="true" hidendirindex="true">
  <we:repeat>
    <we:ifFound>
      <we:comment>Fuer die Verlinkung auf Sprachebene</we:comment\
    >
      <link rel="alternate" hreflang="<we:field name="
WE_TARGETLANGUAGE"/>" href="<we:field name="we_path"/>" />
```

```

        <we:comment>Fuer die Verlinkung auf Sprach- und \
Gebietsebene</we:comment>
        <link rel="alternate" hreflang="<we:field name="\
WE_TARGETLOCALE"/>" href="<we:field name="we_path"/>" />
    </we:ifFound>
</we:repeat>
</we:listview>

```

Dadurch kann zudem das Problem von Duplicate Content vermieden werden. Gleichzeitig können identische Sprachversionen für verschiedene Gebiete (Länder) automatisiert an die regionalen Besonderheiten angepasst werden. So kann es für den Betreiber eines Online-Shops, der auch in Englisch angeboten wird, von Vorteil sein, die Preise für US-Amerikaner in US-Dollar und für Briten in britischen Pfund auszuzeichnen. Dieses Vorgehen lässt sich z. B. auch bei der Datumsformatierung anwenden, wie das nachfolgende Code-Beispiel zeigt.

```

<we:ifPageLanguage match="de_DE" doc="top">
    <we:comment>Deutsche Datumsformatierung Wochentag, Tag. Monat Jahr\
</we:comment>
    <we:date type="php" format="l, d. M Y" />
<we:else/>
    <we:ifPageLanguage match="en_GB" doc="top">
        <we:comment>Britische Datumsformatierung Tag Monat, Jahr</we:\
comment>
        <we:ifIsDomain gb.webediton-buch.de>
            <we:date type="php" format="d M, Y" />
        <we:ifIsDomain>

        <we:comment>US-amerikanische Datumsformatierung Monat, Tag Jahr\
</we:comment>
        <we:ifIsDomain us.webediton-buch.de>
            <we:date type="php" format="M, d Y" />
        <we:ifIsDomain>
    </we:ifPageLanguage>
</we:ifPageLanguage>

```

Die beiden gezeigten Beispiele lassen sich auch beliebig erweitern und miteinander kombinieren. Neben dem Attribut `hreflang` wäre in diesem Zusammenhang noch `rel="canonical"` zu erwähnen, mit dem sich für identische Inhalte unter verschiedenen URLs die bevorzugte Variante auszeichnen ließe. Gerade bei großen Websites und Online-Shops sind Produkt-Detailseiten aufgrund von Such- und Filterfunktionen unter verschiedenen URLs erreichbar. Durch den Einsatz von `rel="canonical"` ließe sich einer derzeitigen Besonderheit des webEdition Shop-Moduls begegnen, die dazu führt, dass Produkt-Detailseiten aus dem Warenkorb heraus über eine andere URL erreichbar sind als außerhalb des Warenkorbs. Dies ist dem Umstand geschuldet, dass das Shop-Modul aktuell keine SEO-URLs unterstützt.

2.5.2 Content-Optimierung

Content-Optimierung – oder besser gesagt, das Bestreben, dem Nutzer relevante Inhalte anzubieten – ist nicht erst mit Erscheinen von Google Panda-Update von großer Bedeutung. Im Gegensatz zu manch anderen, auf Themes und Plugins basierenden Content Management Systemen bietet webEdition den Vorteil, dass es keine automatisierten Quellcode-Ausgaben erzeugt. Der Entwickler allein entscheidet darüber, wie der fertige Code einer Website aussieht, und kann somit die Qualität, Struktur und Usability zu 100 Prozent beeinflussen.

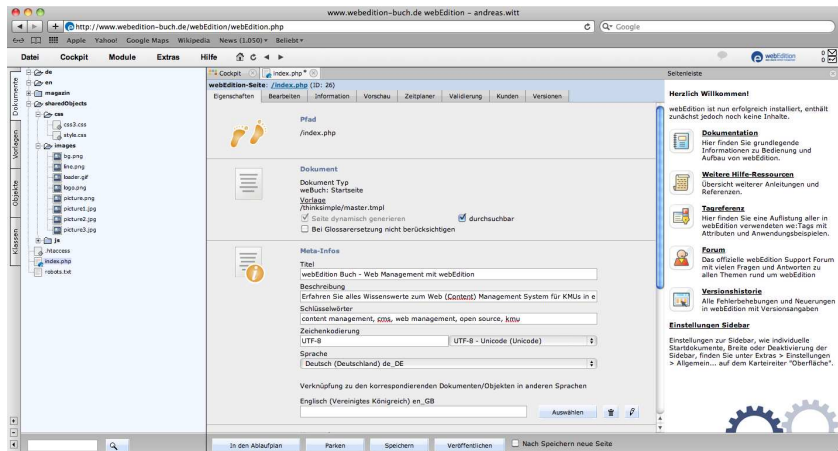
Im Bereich der Suchmaschinenoptimierung befasst sich die Content-Optimierung mit der optimalen Darstellung einer Website in den Suchergebnislisten von Suchmaschinen. In diesem Zusammenhang hat der Marketing-Guru Bryan Eisberg den Begriff *Conversion Trinity* geprägt. Dahinter verbirgt sich ein Vorgehensmodell zur optimalen Gestaltung des Seitentitels einer Webseite, die Bestandteil des sog. Google Snippet (Abbildung 2.8) ist.

Abbildung 2.8:
Google Snippet am
Beispiel von
webedition.org (nicht
optimal)

webEdition CMS | webEdition e.V. - webEdition CMS
www.webedition.org/
 Der Umzug der Domain **webedition.org** sowie sämtlicher damit verbundener Sites und Dienste auf einen eigenen, dedizierten Server wurde ohne größere ...

Optimale Gestaltung umfasst dabei die drei Aspekte *Relevanz*, *Mehrwert* und *Handlungsanreiz*, d. h.: Gestalte den Titel so, dass er für den Suchenden die nötige Relevanz besitzt, die Website einen Mehrwert bietet und er eine Aktion ausführen soll.

Abbildung 2.9:
Dokumenten-
Eigenschaften für
Google Snippet



Neben dem Seitentitel gehört noch ein Beschreibungstext (*Description*) zum Google Snippet. Insgesamt stehen dem Website-Betreiber bzw. dem Redakteur 229 Zeichen (69 für den Titel und 160 für die Beschreibung) zur Verfügung, um den Suchenden davon zu überzeugen, dass er seine Website besuchen soll. Für seine Dokumente bietet webEdition die Dokumenteneigenschaften (siehe Abbildung 2.9), die die Angaben Titel, Beschreibung und Schlüsselwörter enthalten.

Der geschulte Redakteur kann die Felder **Titel** und **Beschreibung** nutzen, um einen optimalen Spannungsbogen nach dem *Conversion-Trinity*-Konzept aufzubauen, und der Entwickler kann auf die Dokumenteneigenschaften zugreifen und sie durch weitere Informationen anreichern. Gleichzeitig gibt es die Möglichkeit, die Werte für diese Felder mit Default-Werten im Voraus auszufüllen, was den Redakteur bei der Arbeit unterstützt.

```
<head>
  <we:charset defined="UTF-8,ISO-8859-1" xml="true"></we:charset>

  <we:comment>neben suffix sind auch noch prefix moeglich</we:comment>
>
  <we:title suffix="Das webEdition Buch" delimiter=" | ">Hier steht \
der Default-Titel</we:title>

  <we:keywords xml="true">Hier stehen die Default-Keywords</we:\
keywords>
  <we:description xml="true">Hier steht die Default-Beschreibung</we:\
description>
  [...]
</head>
```

Bei webEdition-Objekten wird dasselbe Verhalten erzeugt, indem die benutzerdefinierten Felder als Titel, Beschreibung und Schlüsselwörter in den Feldeigenschaften der Klasse einmalig definiert werden.