
Das Offline Content Management System EBCMS

Inhaltsverzeichnis

Technisches Handbuch EBCMS	1
1. Überblick	1
Einstieg	1
Komponenten	1
Eingabedaten	1
Programm	1
Templates	1
Build-Skript	1
CSS	1
2. Logische Struktur	2
Website	2
Kategorie	2
Artikel	2
3. Layout	4
Überlegungen	4
Artikel-View	4
Kategorie-View	5
Anwenderhandbuch EBCMS	1
1. Artikel	1
Einführung	1
Gliederung	1
Titel	1
Kurzübersicht	1
Inhalt	1
Kategorien	1
URL	2
Seitenumbruch	2
Zeitliche Bezüge	2
Erzeugungsdatum	2
Publikationsdatum	3
Publikation verzögern	3
Spezielles	3
Unterdrücken	3
Screenshots	3
Link	4
Anchor	4
Logo	5
CSS	5
JavaScript	5
Open-Graph-Tag Bild	6
Aktualisierungen	6
Fußnoten	7
Include	7
Metadescription	7
Linksammlung	8
Mathematische Formeln	8
Graphen mit PlantUML	8
Alternative Templates	9
Nützliche HTML-Konstrukte	9
Bilder links, umflossen	9
Bilder rechts, umflossen	9

Bilder zentriert	10
List von DownloadLinks	10
2. Kategorien	11
Fachliche Kategorien	11
3. Formatierung	12
Inline Markdown	12
Inline HTML Tags	12

Technisches Handbuch EBCMS

Jürgen Key

Technisches Handbuch EBCMS

Jürgen Key

Inhaltsverzeichnis

1. Überblick	1
Einstieg	1
Komponenten	1
Eingabedaten	1
Programm	1
Templates	1
Build-Skript	1
CSS	1
2. Logische Struktur	2
Website	2
Kategorie	2
Artikel	2
3. Layout	4
Überlegungen	4
Artikel-View	4
Kategorie-View	5

Kapitel 1. Überblick

Einstieg

Das Ziel bei der Erstellung dieser Anwendung war, Möglichkeiten zu schaffen, Artikel im Web zu publizieren, ohne an ein bestimmtes Content Management System oder spezielle Voraussetzungen im Server gebunden zu sein.

Das heißt, ich wollte ein System schaffen, das es erlaubt, am Ende eine statische Webseite zu haben, die auf jeden HTTP-Server ausgerollt werden kann - ohne Skripte oder Datenbank.

Daraus entstand der Gedanke eines Offline Content Management Systems: Eine Sammlung von Softwarekomponenten, die es gestattet, aus Eingabedaten eine Sammlung von HTML-Seiten und zugehörigen Ressourcen zu erstellen, die in ihrer Gesamtheit direkt auf einen beliebigen HTTP-Server kopiert werden können.

Die Eingabedaten sollten so einfach wie möglich gestaltet werden - im Speziellen sollten zum Erstellen der Inhalte keine Kenntnisse in HTML benötigt werden. Daher wurde ein sehr zurückhaltendes eigenes Markup entwickelt, das beim kreativen Prozeß so wenig wie möglich als störend empfunden werden sollte: Die Anzahl der Direktiven wurde sehr gering gehalten, Direktiven stehen für sich allein auf einer Zeile und nicht im Fließtext.

Komponenten

Eingabedaten

Die Eingabedaten sind unerlässlicher Input in den Erstellungsprozeß der Website. Sie sind als Textdateien in einem gemeinsamen Verzeichnis abgelegt. Aus jeder dieser Textdateien entsteht ein Artikel der fertigen Website.

Programm

Das Programm realisiert die Transformation der Eingabedaten mittels Templates in HTML-Dokumente. Dazu baut es aus den Eingabedaten interne Datenstrukturen auf, die dann als Grundlage für den Transformationsprozeß dienen.

Templates

Die Templates sind Velocity-Templates, die die Struktur der einzelnen Bestandteile der Website definieren und durch das Programm mit Daten gefüllt werden. Daraus entstehen dann die HTML-Seiten der Website.

Build-Skript

Das Build-Skript ist der zentrale Dreh- und Angelpunkt des Systems: Es startet das Programm, welches unter Zuhilfenahme der Templates die Eingabedaten in HTML-Dokumente transformiert und kopiert diese zusammen mit den benötigten Ressourcen in ein Zielverzeichnis, so daß dessen Inhalt ohne weitere Anpassungen auf einen beliebigen HTTP-Server ausgerollt werden kann.

CSS

Die Cascading StyleSheets (CSS) geben die visuelle Erscheinung der Website im Browser beim Leser vor. Sie sind Ressourcen und werden vom Build-Skript in das Zielverzeichnis kopiert.

Kapitel 2. Logische Struktur

Website

Eine Website ist eine Sammlung von Artikeln. Diese Artikel sind verschiedenen Kategorien zugeordnet. Jede Kategorie kann mehrere Artikel beinhalten. Jeder Artikel kann zu einer oder mehreren Kategorien gehören.

Kategorie

Eine Kategorie ist eine Klassifizierung von Artikeln anhand eines bestimmten Merkmals: Stimmt dieses Merkmal in zwei Artikeln überein, gehören sie zur selben Kategorie.

Fachliche Kategorien (Tags)

Fachliche Kategorien sind Kategorien, bei denen die Merkmale vom Inhalt der Artikel abhängen: Existieren zwei Artikel, die sich mit dem Betriebssystem Linux beschäftigen, gehören sie zur fachlichen Kategorie Linux. Damit sind fachliche Kategorien das, was heutzutage auch gerne als Tags bezeichnet wird.

Spezialkategorien

Spezialkategorien sind Kategorien, in die Artikel wegen bestimmter technischer Sachverhalte eingeordnet werden. Folgende Spezialkategorien existieren:

Publiziert	Zu dieser Kategorie gehören alle Artikel der Website, deren Veröffentlichungsdatum in der Vergangenheit liegt. Diese Kategorie ist die Grundlage für den Kategorie-View der Startseite.
Wochen	Hier existieren mehrere Kategorien: Artikel gehören dann zur gleichen Wochen-Kategorie, wenn sie in der gleichen Woche veröffentlicht wurden. Diese Kategorien sind die Grundlage der Wochenübersicht.
Monate	Hier existieren mehrere Kategorien: Artikel gehören dann zur gleichen Monats-Kategorie, wenn sie im gleichen Monat veröffentlicht wurden. Diese Kategorien sind die Grundlage der Monatsübersicht.
Zukunft	Zu dieser Kategorie gehören alle Artikel der Website, deren Veröffentlichungsdatum in der Zukunft liegt.

Artikel

Artikel sind durch verschiedene Eigenschaften definiert, die im Nachfolgenden genauer beschrieben werden:

Erstellungsdatum	Wenn nicht explizit angegeben, wird das Modifikationsdatum der zugehörigen Textdatei benutzt. Kann auch in der Zukunft liegen.
Abstract	Kurze Inhaltsangabe - wird zur Darstellung des Anrisses in Kategorie-Views benutzt. Darf gültige HTML-Konstrukte anhalten - allerdings keine <a>-Tags! Obligatorisch.

Content	Inhalt des Artikels. Darf gültige HTML-Konstrukte enthalten. Fakultativ.
Tags	Liste von durch Kommata getrennten Kategoriebezeichnern. Fakultativ.
Frontpage-Flag	Falls gesetzt, wird dieser Artikel immer als jüngster - an der Spitze aller Kategorien, zu denen er gehört - einsortiert. Fakultativ.
URL	Name der resultierenden HTML-Seite (ohne abschließendes .html). Falls nicht vorhanden, wird ein Name vom System generiert.
Screenshot-Gallery	Wenn ein Artikel viele Illustrationen benötigt, kann der Verfasser diese entweder mittels HTML-Code selbst bauen, oder diese spezielle Eigenschaft eines Artikels nutzen: werden hier die Namen der Bilder hinterlegt, erstellt das System automatisch eine entsprechende Galerie im erzeugten HTML-Dokument.
Fußnoten	Fußnoten für den Artikel. Fakultativ.
Linksammlungen	Sammlungen von Ressourcen, die durch einen kurzen beschreibenden Text erklärt werden und einen Link zur entsprechende Ressource mitbringen. Fakultativ.
Seitenumbrüche	Ein langer Artikel kann durch Seitenumbrüche in mehrere Seiten aufgeteilt werden. Die einzelnen Seiten stehen durch Querverweise miteinander in Verbindung und können dadurch vom Leser erreicht werden.

Kapitel 3. Layout

Überlegungen

Vor Erstellung der Templates und der CSS muß zunächst klar sein, welchen Vorgaben die inhaltliche Struktur der Website folgen soll. Anhand dessen kann man ein Layout und schließlich das CSS und die Templates entwickeln.

Dazu wird zunächst die logische Struktur der Website festgelegt. In diesem Fall ist es so, daß die Website zwei verschiedene Views auf die Daten anbieten soll:

Ein View soll der Präsentation eines Artikels dienen: ein Artikel wird vollständig angezeigt, so daß ein Leser in der Lage ist, die Informationen darin aufzunehmen. Dieser View soll darüber hinaus noch die Möglichkeit bieten, schnell zu den neuesten Artikeln der Website zu wechseln. Weiterhin liefert er einen schnellen Überblick über alle Kategorien oder Tags der Website, wie die, zu denen der aktuell angezeigte Artikel gehört.

Ein zweiter View dient der übersichtlichen Darstellung aller Artikel in einer Kategorie. Dabei wird für jeden Artikel der Kategorie ein Anriss angezeigt, der den Titel und das Abstract dieses Artikels angezeigt. Möchte ein Leser weitere Informationen zu dem Thema erlangen, muß er auf diesen Anriss klicken.

Artikel-View

Für den Artikel-View wurde ein dreispaltiges Layout gewählt: in der Mitte wird der Inhalt des Artikels selbst angezeigt, links daneben die neuesten Artikel und rechts die Liste der Kategorien. Oberhalb der Kategorien sind Links angeordnet, die zu speziellen Abschnitten der Website führen:

RSS	Link auf das RSS-Dokument für die Website
Wochenübersicht	Link zu einem Kategorie-View, die von den fachlichen Kategorien oder Tags abweicht: hier werden als Kategorien die einzelnen Kalenderwochen benutzt, so daß man schnell und gezielt einen Überblick über alle in einem bestimmten Zeitraum veröffentlichten Artikel bekommt.
Monatsübersicht	Link zu einem Kategorie-View, die von den fachlichen Kategorien oder Tags abweicht: hier werden als Kategorien die einzelnen Monate benutzt, so daß man schnell und gezielt einen Überblick über alle in einem bestimmten Zeitraum veröffentlichten Artikel bekommt.
Impressum	Artikel-View mit Angaben zum Autor
Startseite	Kategorie-View mit allen Artikeln der Seite sortiert nach Veröffentlichungsdatum: jüngste zuerst. Auf der Startseite wird dieser Link wegelassen.

Die Farbe der Kategorienamen ist dabei etwas heller dargestellt: Nur die Namen der Kategorien, zu denen der aktuell dargestellte Artikel gehört, werden schwarz dargestellt. Bewegt man die Maus über eine der Kategorien, wird ihre - bis dahin unsichtbare - Kurzbeschreibung angezeigt.

Die Liste der neuesten Artikel links wird auf eine Maximalzahl beschränkt. Gehört der aktuell angezeigte Artikel zu den neuesten Artikeln, wird er links in der Liste nicht mit aufgeführt, statt dessen rückt ein älterer nach. In der Liste der neuesten Artikel wird jeweils der Anriss angezeigt, der den Titel und das

Abstract dieses Artikels enthält. Möchte ein Leser weitere Informationen zu dem Thema erlangen, muß er auf diesen Anriss klicken.

Anmerkung

Bei Geräten unter 1000 Pixeln horizontaler Auflösung fällt im Artikel-View die Liste der neuesten Artikel links weg.

Kategorie-View

Kategorie-Views haben ein zweispaltiges Layout: Im größeren linken Bereich werden Anrisse der Artikel, die zu der jeweiligen Kategorie gehören dargestellt. Anrisse bestehen aus Titel und Abstract dieses Artikels. Möchte ein Leser weitere Informationen zu dem Thema erlangen, muß er auf diesen Anriss klicken. Diese Anrisse werden in zwei Spalten angeordnet.

Maximal 6 Anrisse werden gleichzeitig dargestellt, sin dmehr Artikel für eine Kategorie enthalten, ist der letzte Anriss unten rechts ein Link zur nächsten Seite mit den darauf folgenden Artikeln dieser Kategorie. Artikel innerhalb einer Kategorie werden nach ihrem Veröffentlichungsdatum sortiert - der jüngste zuerst.

Rechts daneben ist die Liste der Kategorien angeordnet: der Name der jeweils aktuell angezeigten ist schwarz dargestellt, die Namen der anderen etwas heller. Oberhalb der Kategorien sind Links angeordnet, die zu speziellen Abschnitten der Website führen:

RSS	Link auf das RSS-Dokument für die Website
Wochenübersicht	Link zu einem Kategorie-View, die von den fachlichen Kategorien oder Tags abweicht: hier werden als Kategorien die einzelnen Kalenderwochen benutzt, so daß man schnell und gezieht einen Überblick über alle in einem bestimmten Zeitraum veröffentlichten Artikel bekommt.
Monatsübersicht	Link zu einem Kategorie-View, die von den fachlichen Kategorien oder Tags abweicht: hier werden als Kategorien die einzelnen Monate benutzt, so daß man schnell und gezieht einen Überblick über alle in einem bestimmten Zeitraum veröffentlichten Artikel bekommt.
Impressum	Artikel-View mit Angaben zum Autor
Startseite	Kategorie-View mit allen Artikeln der Seite sortiert nach Veröffentlichungsdatum: jüngste zuerst. Auf der Startseite wird dieser Link wegelassen.

Anwenderhandbuch EBCMS

Jürgen Key

Anwenderhandbuch EBCMS

Jürgen Key

Inhaltsverzeichnis

1. Artikel	1
Einführung	1
Gliederung	1
Titel	1
Kurzübersicht	1
Inhalt	1
Kategorien	1
URL	2
Seitenumbruch	2
Zeitliche Bezüge	2
Erzeugungsdatum	2
Publikationsdatum	3
Publikation verzögern	3
Spezielles	3
Unterdrücken	3
Screenshots	3
Link	4
Anchor	4
Logo	5
CSS	5
JavaScript	5
Open-Graph-Tag Bild	6
Aktualisierungen	6
Fußnoten	7
Include	7
Metadescription	7
Linksammlung	8
Mathematische Formeln	8
Graphen mit PlantUML	8
Alternative Templates	9
Nützliche HTML-Konstrukte	9
Bilder links, umflossen	9
Bilder rechts, umflossen	9
Bilder zentriert	10
List von DownloadLinks	10
2. Kategorien	11
Fachliche Kategorien	11
3. Formatierung	12
Inline Markdown	12
Inline HTML Tags	12

Liste der Beispiele

1.1.	2
1.2.	3
1.3.	4
1.4.	4
1.5.	4
1.6.	5
1.7.	5
1.8.	5
1.9.	6
1.10.	6
1.11.	6
1.12.	7
1.13.	7
1.14.	8
1.15.	8
1.16. Beispiel für einen custom Server für die Konvertierung eines PlantUML-Diagramms in ein Bild	9

Kapitel 1. Artikel

Einführung

Artikel werden als Normaler Text geschrieben. Das Encoding der Textdatei muß dabei UTF-8 sein. Im Artikel wechseln sich Direktiven und Inhalt ab. Direktiven sind dabei Zeilen, die mit einem Punkt beginnen. Inhalt, der nach einer Direktive fortgesetzt werden soll, muß auf einer eigenen Zeile beginnen und darf nicht in derselben Zeile einer Direktive auftauchen.

Kommentare werden durch ein # am Zeilenanfang eingeleitet. Alles, was in einer Zeile steht, die mit diesem Zeichen beginnt, wird ignoriert: Auch Direktiven sind dann wirkungslos.

Abschnitte im Artikel werden explizit durch eine Direktive eingeleitet und implizit durch die nächste Direktive beendet: Es existiert zum Beispiel eine Direktive, die den Beginn der Kurzzusammenfassung anzeigt, jedoch keine, die ausdrücklich deren Ende anzeigt: dieses Ende ist automatisch dann erreicht, wenn eine Zeile mit einer Direktive auftaucht.

Gliederung

Titel

Der Titel des Artikels - der nachfolgende Inhalt sollte kurz und prägnant sein. Innerhalb des Inhaltes dieses Abschnittes ist kein HTML erlaubt.

`Obligatorisch!`

Name der Direktive:

`.HEADING`

Kurzübersicht

Die Kurzübersicht über den Inhalt des Artikels. Der nachfolgende Text sollte das Thema des Artikels in Kurzform vorstellen. Darf gültige HTML-Konstrukte enthalten - allerdings keine <a>-Tags!

`Obligatorisch!`

Name der Direktive:

`.ABSTRACT`

Inhalt

Der Inhalt des Artikels. Darf gültige HTML-Konstrukte enthalten. Absätze werden durch Leerzeilen gekennzeichnet.

`Fakultativ.`

Name der Direktive:

`.CONTENT`

Kategorien

Die Namen der fachlichen Kategorien, zu denen der Artikel gehört. Beliebig viele Namen fachlicher Kategorien durch Komma getrennt.

`Obligatorisch!`

Name der Direktive:

`.TAGS`

URL

Die URL dieses Artikels.

`Fakultativ.`

Normalerweise werden Titel für die aus den Artikeln generierten Webseiten automatisch generiert. Durch Verwendung dieser Direktive kann der Automatismus ausgesetzt und ein eigener Name für die generierte Webseite des Artikels vorgegeben werden Name der Direktive:

`.URL`

Der Name selbst steht in der unmittelbar auf diese Direktive folgenden Zeile ohne die Endung .html.

Seitenumbruch

Diese Direktive erlaubt es, einen Artikel in mehrere Seiten aufzuteilen. Das Template muß dafür sorgen, daß die einzelnen Seiten durch den Leser zu erreichen sind. Das kann zum Beispiel durch die Einrichtung eines Inhaltsverzeichnisses auf der ersten Seite geschehen, dessen Einträge Links zu den einzelnen Seiten darstellen:

`Fakultativ.`

Name der Direktive:

`.PAGE`

Zwischen den einzelnen Direktiven für Seitenumbrüche dürfen alle Direktiven stehen, die in den Artikeln auch sonst erlaubt sind. Folgende Direktiven bilden eine Ausnahme - sie sind nur einmal im Dokument erlaubt und beziehen sich immer auf den Artikel als Ganzes:

1. .TAGS
2. .CREATED
3. .PUBLISHED
4. .FUTURE
5. .HIDDEN
6. .URL

Zeitliche Bezüge

Erzeugungsdatum

Das Erzeugungsdatum dieses Artikels.

`Fakultativ`

- Falls nicht angegeben, wird das Veröffentlichungsdatum (siehe unten) des Artikels herangezogen. Falls es noch kein Veröffentlichungsdatum gibt und diese Direktive nicht gesetzt wurde, wird das aktuelle Datum beim Erzeugen der HTML-Seite eingesetzt. Name der Direktive:

`.CREATED`

Format der Direktive: Der Name wird durch Leerzeichen getrennt gefolgt vom Erzeugungsdatum des Artikels in folgendem Format (Beispiel):

Beispiel 1.1.

Mi, 01 Mai 2013 13:10:43

Publikationsdatum

Das Datum, an dem dieser Artikel zum erstenmal öffentlich zugänglich ist/sein wird.

`Fakultativ`

- Falls nicht angegeben, wird der Artikel als in der Zukunft zu veröffentlichen (siehe unten) gekennzeichnet. Name der Direktive:

`.PUBLISHED`

Format der Direktive: Der Name wird durch Leerzeichen getrennt gefolgt vom Publikationsdatum des Artikels in folgendem Format (Beispiel):

Beispiel 1.2.

Mi, 01 Mai 2013 13:10:43

Publikation verzögern

Zeigt an, daß dieser Artikel irgendwann in der Zukunft erscheinen wird. Damit wird der Artikel in die Kategorie Zukunft eingeordnet - Leser sehen daher zunächst nur Titel und Kurzübersicht; der Inhalt bleibt zunächst noch verborgen

`Fakultativ.`

Name der Direktive:

`.FUTURE`

Spezielles

Unterdrücken

Diese Direktive unterdrückt die Publikation eines Artikels, auch wenn ein gültiges Publikationsdatum gesetzt wurde, das in der Vergangenheit liegt.

`Fakultativ.`

Name der Direktive:

`.HIDDEN`

Screenshots

Diese Direktive eröffnet einen speziellen Abschnitt im Artikel: Sie dient der Darstellung einer Galerie von Screenshots.

`Fakultativ.`

Name der Direktive:

`.SCREENSHOTGALLERY`

Jeder Screenshot besteht aus einem Paar von Abbildungen: einer Miniaturansicht (Thumbnail) und der eigentlichen Abbildung, sowie einem Titel und einer (optionalen) Beschreibung. Nach Beginn dieses Abschnittes werden die folgenden Zeilen wie folgt interpretiert: Am Beginn einer Zeile steht ein Paar Dateinamen (mit Pfad relativ zu dem Verzeichnis, das auch das Verzeichnis mit den Artikeln enthält). Dieses Paar wird durch = getrennt - dabei steht links der Name der Miniaturansicht und rechts der der Abbildung. Daraus wird ein Link erzeugt, der die Miniaturansicht zeigt und auf die Abbildung verweist. Wird die rechte Seite und das = weggelassen, wird nur die Miniaturansicht eingefügt und kein Link erzeugt. Nach dem Paar von Dateinamen kann ein § folgen. Der darauf folgende Text wird als Bildunterschrift der

Miniaturansicht benutzt. Folgt darauf nochmals ein §, wird der Text, der danach folgt, als Bildbeschreibung der Abbildung eingefügt.

Screenshots, die auf diese Weise definiert werden, werden nicht im Fließtext der erzeugten Seite integriert, sondern stehen zum Zeitpunkt der Expansion des Templates für Artikel als Collection (Liste) zur Verfügung, so daß sie gemeinschaftlich formatiert werden können.

Beispiel 1.3.

```
images/mini.png=images/abb.png§Kurztitel§Ausführliche Beschreibung
```

Eine Leerzeile nach der Definition solcher Screenshots schließt die Definition ab. Weiterer Inhalt, der nach dieser Leerzeile steht, wird wieder als normaler Inhalt des Artikels behandelt.

Link

Lokal

Erzeugt einen Link auf einen anderen Artikel.

`Fakultativ.`

Name der Direktive:

`.LINK`

Format ist `ziel#anchor=text`. Dabei ist Text der Text des Links und Ziel der Name der Datei des Artikels, auf den verlinkt werden soll (ohne Pfad) und Anchor stellt den Ankerpunkt dar (- siehe „Anchor“; dieser ist optional, ist keiner angegeben, darf das Zeichen # ebenfalls fehlen) - also zum Beispiel

Beispiel 1.4.

```
.LINK abc.txt#anchor=Auf mich kann man klicken!
```

Global

Erzeugt einen Link ins Internet.

`Fakultativ.`

Name der Direktive:

`.LINK`

Format ist `ziel=text`. Dabei ist Text der Text des Links und Ziel die absolute URL, auf die verlinkt werden soll - zum Beispiel

Beispiel 1.5.

```
.LINK https://elbosso.github.io/index.html=EL BOSSOs Site
```

Anchor

Erzeugt einen Ankerpunkt in einem Artikel.

`Fakultativ.`

Name der Direktive:

`.ANCHOR`

Format ist name. Dabei ist name der Ankerpunkt (kann zum Beispiel mit der Direktive `.LINK` benutzt werden - siehe „Link“) - also zum Beispiel

Beispiel 1.6.

```
.ANCHOR anchor
```

Logo

Definiert eine URL, die auf eine Bilddatei verweist, die als Logo für den Artikel dient. Das Bild kann sowohl im Anriss wie im eigentlichen Artikel Verwendung finden

`Fakultativ.`

Name der Direktive:

`.SMALLLOGOURL`

Kann sowohl absolut wie auch relativ angegeben werden. Der Wert der Direktive ist der Inhalt der unmittelbar auf die Direktive folgende Zeile.

Beispiel 1.7.

```
.smalllogourl  
small/aviator_logo.png
```

CSS

Definiert eine URL, die auf eine CSS-Ressource verweist, die in die resultierende HTML-Seite eingebunden werden soll.

`Fakultativ.`

Name der Direktive:

`.CSS`

Kann sowohl absolut wie auch relativ angegeben werden. Der Wert der Direktive ist der unmittelbar auf die Direktive folgende Text.

Beispiel 1.8.

```
.css resources/ascinema/ascinema-player.css
```

JavaScript

Definiert eine URL, die auf eine JavaScript-Ressource verweist, die in die resultierende HTML-Seite eingebunden werden soll.

`Fakultativ.`

Name der Direktive:

`.JS`

Kann sowohl absolut wie auch relativ angegeben werden. Der Wert der Direktive ist der unmittelbar auf die Direktive folgende Text.

Beispiel 1.9.

```
.js resources/asciinema/asciinema-player.js
```

Open-Graph-Tag Bild

Teilt ein Nutzer eine Website in den sozialen Medien, muss diese auf der Plattform irgendwie dargestellt werden. Ein Vorschaubild und der begleitende Text sollen die Website möglichst gut widerspiegeln und andere User zum Besuch anregen. Damit aber nicht einfach irgendein Inhalt auf der Website in der Vorschau gezeigt wird, nutzen Webdesigner Open Graph. Webdesigner können in den Quelltext der Website Open-Graph-Tags einfügen und dadurch beeinflussen, wie die Vorschau in den Netzwerken aussieht (Quelle: IONOS [<https://www.ionos.de/digitalguide/online-marketing/social-media/open-graph/>]). EBCMS nimmt das dem Autor ab: aus den Metadaten des jeweiligen Artikels werden die Informationen für die Open-Graph-Tags extrahiert und die Tags automatisch in die Seite eingefügt. Das anzuzeigende Bild kann leider nicht abgeleitet werden und muss - falls gewünscht - mittels dieser Direktive spezifiziert werden. Falls diese Direktive nicht benutzt wird, aber ein Logo mittels „Logo“ angegeben wurde, wird dieses dafür verwendet.

Fakultativ.

Name der Direktive:

.OPENGRAPHIMAGEURL

Kann sowohl absolut wie auch relativ angegeben werden. Der Wert der Direktive ist der Inhalt der unmittelbar auf die Direktive folgende Zeile.

Beispiel 1.10.

```
.opengraphimageurl  
small/aviator_logo.png
```

Aktualisierungen

Diese Direktive hat eine entsprechende Direktive zur expliziten Beendigung. Der Abschnitt zwischen beiden Direktiven dient der Aufnahme von später zum Artikel hinzugefügten Inhalten oder Aktualisierungen.

Fakultativ.

Name der Direktive, die den Beginn eines solchen Abschnittes anzeigt:

.UPDATE

Format der Direktive: Der Name wird durch Leerzeichen getrennt gefolgt vom Erzeugungsdatum des Artikels in folgendem Format (Beispiel):

Beispiel 1.11.

```
Mi, 01 Mai 2013 13:10:43
```

Man kann statt des konkreten Datums auch den Platzhalter

.\$today\$

einsetzen, der durch das System durch das aktuelle Datum ersetzt wird. Ab der darauf folgenden Zeile gehört sämtlicher folgender Text zur Aktualisierung bis die Direktive

`.UPDATEEND`

erkannt wird.

Weitere Platzhalter für das Datum der Aktualisierung sind

`$plusoneweek$`

und

`$plustendays$`

Diese werden entsprechend durch ein Datum ersetzt, das eine Woche beziehungsweise 10 Tage nach dem Erstellungsdatum liegt. Die Besonderheit dieser beiden Platzhalter ist, daß sie so lange nicht ersetzt werden, wie die Direktive `.future` noch im Artikel steht. Logisch lässt sich das damit begründen, daß ja das Erstellungsdatum noch unbekannt ist und daher auch der Zeitpunkt eine Woche oder 10 Tage danach unbekannt sein muß.

Fußnoten

Erzeugt eine Fußnote im Artikel.

`Fakultativ.`

Name der Direktive:

`.FOOTNOTE`

Der Rest der Zeile ist der Text der Fußnote - also zum Beispiel

Beispiel 1.12.

```
.FOOTNOTE Das ist ein wenig Hintergrundwissen zum Thema Fußnoten
```

Include

Erlaubt, Artikel in mehrere Dokumente aufzuteilen.

`Fakultativ.`

Name der Direktive:

`.INCLUDE`

Der Rest der Zeile ist der Pfad der Datei, deren Inhalt an dieser Stelle eingefügt werden soll relativ zum Verzeichnis des Artikels.

Beispiel 1.13.

```
.INCLUDE includedir/snippet
```

Metadescription

Erlaubt, eine kurze Beschreibung des Artikels hinzuzufügen. Kann zum Beispiel bei der Erzeugung von HTML mittels Velocity-Templates wie folgt benutzt werden: `<meta name="Description" content="{article.metaDescription}" />`. Falls keine Metadescription explizit gesetzt wird, wird statt dessen der mittels „Kurzübersicht“ angegebene Text benutzt.

`Fakultativ.`

Name der Direktive:

`.`

Der auf die Direktive folgende Text bis zur nächsten Direktive (gegebenenfalls auch mehrere Zeilen) ist der Wert der Direktive.

Beispiel 1.14.

```
.metadescription
Java Mangement Extensions oder JMX erlauben es auf
einfache Art und Weise, eine Serveranwendung mit
Sensoren zur Überwachung zu instrumentieren
```

Linksammlung

Diese Direktive eröffnet einen speziellen Abschnitt im Artikel: Sie dient der Darstellung einer Sammlung von Links:

```
<strong>Fakultativ.</strong>
```

Name der Direktive:

```
<strong>.COLLECTEDLINK</strong>
```

Jeder Link besteht aus einem erklärenden Text und einem Ziel oder Querverweis. Nach Beginn dieses Abschnittes werden die folgenden Zeilen als Teil der Beschreibung interpretiert. Der Inhalt der ersten Zeile, die nach Beginn dieses Abschnittes mit `http` beginnt, wird als Ziel des Links interpretiert. Damit ist die Definition eines Links abgeschlossen.

Links, die auf diese Weise definiert werden, werden nicht im Fließtext der erzeugten Seite integriert, sondern stehen zum Zeitpunkt der Expansion des Templates für Artikel als Collection (Liste) zur Verfügung, so daß sie gemeinschaftlich formatiert werden können.

Beispiel 1.15.

```
.collectedLink
unter anderem findet man hier Informationen zum Produkt EBCMS
http://elbosso.github.io
```

Eine Leerzeile nach der Definition eines solchen Links schließt die Definition ab. Weiterer Inhalt, der nach dieser Leerzeile steht, wird wieder als normaler Inhalt des Artikels behandelt.

Mathematische Formeln

Diese Direktiven schließen einen syntaktisch korrekten mathematischen TeX-Ausdruck ein. Während der Generierung der Web-Site wird dieser Ausdruck in eine PNG-Graphik umgewandelt und ein Verweis auf diese Graphik an die entsprechende Stelle im erzeugten HTML-Code eingefügt. Die Formel erscheint dann zentriert im Browser.

```
<strong>Fakultativ.</strong>
```

Name der Direktiven:

```
<strong>.FORMULA</strong>
```

und

```
<strong>.FORMULAEND</strong>
```

Graphen mit PlantUML

Diese Direktiven schließen ein syntaktisch korrektes Plant-UML-Dokument ein. Während der Generierung der Web-Site wird dieser Ausdruck in eine PNG-Graphik umgewandelt und ein Verweis auf diese Graphik an die entsprechende Stelle im erzeugten HTML-Code eingefügt. Die Graphik erscheint dann zentriert im Browser.

```
<strong>Fakultativ.</strong>  
Name der Direktiven:  
<strong>.PLANTUML</strong>  
und  
<strong>.PLANTUML</strong>
```

Es ist möglich, dass der Direktive

```
<strong>.PLANTUML</strong>
```

durch ein oder mehrere Whitespace (keine Zeilenumbrüche!) getrennt ein optionaler Parameter folgt. Dieser beschreibt dann die URL eines alternativen Servers, der zum Rendering herangezogen wird. Wird dieser optionale Parameter nicht angegeben, wird der Server benutzt, der in der System-Property `de.elbosso.tools.cms.MainClass.plantumlurl` angegeben ist. Ein Beispiel der Benutzung ist in Beispiel 1.16, „Beispiel für einen custom Server für die Konvertierung eines PlantUML-Diagramms in ein Bild“ in „Graphen mit PlantUML“ zu finden:

Beispiel 1.16. Beispiel für einen custom Server für die Konvertierung eines PlantUML-Diagramms in ein Bild

```
.PLANTUML https://www.plantuml.com/plantuml/png/  
Object <|-- ArrayList  
  
Object : equals()  
ArrayList : Object[] elementData  
ArrayList : size()  
.PLANTUML</strong>
```

Alternative Templates

Diese Direktive spezifiziert ein abweichendes Template für den Artikel, in dem sie angegeben wird. Der Wert der Direktive folgt in der selben Zeile direkt hinter dem Namen der Direktive.

```
<strong>Fakultativ.</strong>  
Name der Direktive:  
<strong>.TEMPLATE</strong>
```

Nützliche HTML-Konstrukte

Es existieren einige nützliche HTML-Konstrukte zur Verwendung im Inhaltsabschnitt eines Artikels.

Bilder links, umflossen

```

```

Bilder rechts, umflossen

```
<img src="images/image.png"
```



```
alt="Text?"  
class="imgright">
```

Bilder zentriert

```
  
<span class="imgdesc">  
  Bildbeschreibung  
</span>
```

List von DownloadLinks

```
<dl>  
<a href="resources/license.txt">  
<dt></dt>  
<dd>Lizenz</dd>  
</a>  
<a href="resources/sentMails.sh">  
<dt></dt>  
<dd>sentMails</dd>  
</a>  
</dl>
```

Kapitel 2. Kategorien

Fachliche Kategorien

Die fachlichen Kategorien werden in den Artikeln an der rechten Seite dargestellt - die, zu denen der aktuell angezeigte gehört, werden dabei hervorgehoben.

Prinzipiell ist es nicht nötig, diese Kategorien speziell zu konfigurieren: Alle in den Artikeln referenzierten Tags tauchen automatisch in dieser Liste auf.

Allerdings kann man die Tags noch ein wenig aufwerten, indem man dafür sorgt, daß kurze Erläuterungen erscheinen, wenn man den Muszeiger darüber ruhen läßt. Dazu muß man eine Datei namens tags.properties pflegen, die im selben Verzeichnis liegt, wie das Verzeichnis, das die Artikel enthält.

Diese Datei hat einen denkbar einfachen Aufbau: Sie ist eine normale Properties-Datei, bei der die Tags die Property-Namen darstellen und die Kurzbeschreibung den Property-Wert.

Kapitel 3. Formatierung

Inline Markdown

Inline Markdown

Inline Markdown

Um flexibler bei der Formatierung von Fließtext agieren zu können, wurden einige Features implementiert, die aus Wikis und anderen Markup-Systemen wie etwa Markdown bekannt sein dürften:

So kann man durch Einschließen des Textes in entsprechende Marker erreichen, dass der eingeschlossene Text ***fett***, *_kursiv_* oder ``mit fester Zeichenbreite`` dargestellt wird. Das sind die voreingestellten Stile - erreicht wird diese Wirkung durch die Definition von drei Klassen im CSS, die der Autor beliebig anpassen kann - die hier gezeigten Auswirkungen sind die Voreinstellung. Die Klassen und ihre Standardeinstellungen heißen

```
mdbold      {font-weight: bold;}
mdcode      {font-family: monospace;}
mditalics    {font-style: italic;}
```

Diese Auszeichnungen können sowohl im Artikeltext wie auch in der Kurzzusammenfassung verwendet werden.

Es ist möglich, Aufzählungslisten in Markdown-Syntax in Artikeln zu verwenden - dabei ist es möglich, ungeordnete Listen und geordnete Listen zu verwenden. Ungeordnete Listen werden durch Zeilen identifiziert, die mit `*` beginnen. Links vom `*` stehende Leerzeichen dienen dabei der Spezifikation der Schachtelungstiefe. Geordnete Listen werden durch Zeilen gebildet, die mit einer Zahl, gefolgt von einem Punkt (`.`), gefolgt von einem Leerzeichen eingeleitet werden. Auch hier bestimmt die Anzahl von Leerzeichen links der Zahl die Schachtelungstiefe. Geordnete und ungeordnete Listen können beliebig ineinander geschachtelt werden.

Es ist möglich, Bilder in Markdown-Syntax in Artikeln zu verwenden. Die Syntax ist die übliche: `! [image info](images/application.png)`

Inline HTML Tags

HTML-Tags können überall uneingeschränkt eingesetzt werden - allerdings kann es bei zu sorglosem Einsatz dazu kommen, dass das resultierende Template zerstört wird - daher gilt hier: unbedingt ausführlich testen!