

Meta-Kommandos in HTML-Seiten

Jobst-Hartmut Lüddecke

14. Dezember 2011

Zusammenfassung

Jedes verwaltete Dokument muss – um wiedergefunden zu werden – nach bestimmten Kriterien eingeordnet werden. Damit geht es einer Web-Seite im *Internet* nicht anders als einem Buch in einer Bibliothek. Diese Einordnung geschieht in beiden Fällen mit Hilfe einer *Klassifikation*.

Wird eine Web-Seite nicht *klassifiziert*, werten *Suchmaschinen* in der Regel nur die ersten 100 Zeichen der Seite aus. Kommt die *relevante Information* in der Seite erst nach diesen 100 Zeichen, dann findet sie in den Suchmaschinen keine Berücksichtigung mehr.

Um dies nun etwas *professioneller* zu machen gibt es zwei Arten der *Klassifikation*. Dies sind die in *html* definierten *Meta-Kommandos* mit festen *Schlüsselbegriffen* als *Standard* und den *Dublin Core*. Welche dieser Arten nun von welcher Suchmaschine wie ausgewertet wird, ist nicht so ganz klar und so liegt man auf der sicheren Seite einfach **beide** sinnvoll zu verwenden.

Inhaltsverzeichnis

1	Klassifikation mit Meta-Kommandos	1
1.1	Klassifikation (Standard)	2
1.2	Klassifikation nach dem Dublin-Core	4

1 Klassifikation mit Meta-Kommandos

Meta-Kommandos bieten eine Möglichkeit zusätzliche Informationen zur Seite innerhalb der Seite zu vermitteln. Dies geschieht immer im Kopf der Seite, also zwischen `<head>` und `</head>`.

Diese Klassifikation umfasst Angaben, wie den *Autor* der Seite, die *Sprache*, *Urheberrechte*, *Inhaltsangabe*, *Themenbereich* usw. Kurz gesagt, Angaben die einen *Bibliothekar glücklich machen können* und von vielen Suchmaschinen genutzt werden, um die Seite vernünftig zu indexieren (gelegentlich auch zu indizieren)¹ und damit wiederzufinden. Es ist also eine der *Basis-Kompetenzen* von Bibliothekaren.

Man spricht dabei auch von einer Zuordnung von *Deskriptoren* zu einem Dokument zur *Erschließung* im Rahmen eines *Information Retrieval*.

Zu diesem Zweck gibt es den Befehl `<meta>` mit mindestens einem Options-Paar. Die erste Option ist entweder *name*, oder *http-equiv*, gefolgt von einem Gleichheitszeichen und dem **Namen** der *Variablen* in Anführungsstrichen. Die zweite Option hat den Namen *content*, gefolgt von einem Gleichheitszeichen und den Angaben in Anführungsstrichen. Der *content* ist der **Inhalt** der *Variablen*.

Der Unterschied von *http-equiv* und *name* ist der, dass bei bekannten Werten von *http-equiv* gleichzeitig Anweisungen an den *http-Server*, dem *Browser*, oder dem *Proxy-Server* ergehen (nähere Angaben

¹Schauen Sie doch einfach einmal bei de.wikipedia.org nach, was der kleine Unterschied ist.

bei den Befehlen). Der Name der Variablen, bestimmt durch *name*, kann im Prinzip beliebig sein, allerdings werden von den Suchmaschinen und Browsern nur **bekannte** Variablen ausgewertet.

Die Reihenfolge der verschiedenen Variablen ist beliebig. Ein vorgefertigtes Schema (Template) in der immer gleichen Reihenfolge erleichtert die Arbeit aber wesentlich. Es braucht dann nur mit *copy and paste* eingefügt und sinnvoll ausgefüllt zu werden.

1.1 Klassifikation (Standard)

http-equiv="content-type" es geht gleich damit los, um was für eine Art von Dokument es sich handelt und welcher Zeichensatz Verwendung finden soll. Der *content* besteht also mindestens aus dem *MIME-Type*² und dem zu verwendenden Zeichensatz. Damit wird dem *Sender* und *Empfänger* mitgeteilt, welches Übertragungsprotokoll Verwendung finden soll (in diesem Fall *http*) und welcher Zeichensatz der Browser (also Empfänger) zur Darstellung der Seite verwenden soll (im Beispiel *iso8859-2*, also der Zeichensatz für Zentral-Europa. *Iso8859-1* wäre West-Europa und die USA und eine Alternative für neue Versionen der Browser aus den USA). Hier lohnt noch einmal ein Blick zurück in die Lektion *Codes*!

```
<meta http-equiv="content-type"
      content="text/html;CHARSET=iso8859-2">
```

http-equiv="expires" hiermit kann man selbst bestimmen, wie lange diese Seite maximal im *Puffer* des *Browsers*, bzw. im *Proxy-Server* gelagert werden soll, bevor die Seite als *Ungültig* gilt und bei Bedarf neu geladen werden soll. Ein Inhalt von *Null* bewirkt immer eine neue Anforderung und es liegt kein *Alter Schrott* längere Zeit auf einem *Proxy-Server*. Allerdings verliert der *Proxy-Server* auch damit seine Vorteile, da er ja *puffert* bzw. *zwischenspeichert* um gleiche Internet-Zugriffe zu reduzieren.

```
<meta http-equiv="expires" content="0">
```

name="AUTHOR" jetzt wird es einfacher, die Variable *AUTHOR* ist recht selbstredend und wird in großer Schreibweise von fast allen Suchmaschinen erkannt und ausgewertet.

```
<meta name="AUTHOR" content="Jobst-Hartmut Lueddecke">
```

name="description" kurze Inhaltsangabe der Seite, die gern von Suchmaschinen als Beschreibung und Inhaltsangabe der Seite verwendet werden. *Romane* werden gern nach wenigen Zeilen mit Punkten abgekürzt.

```
<meta name="description"
      content="Kurze Inhaltsangabe der Seite in wenigen Zeilen">
```

²*Multipurpose Internet Mail Extension*, die die Art der Übertragung im Internet bestimmt. Diese Angaben sind genormt und funktionieren nur, wenn der betreffende Mime-Type sowohl auf der Senderseite, in diesem Fall der *http-Server* als auch auf der Empfängerseite, in diesem Fall der *Browser* bekannt und definiert sind.

name="keywords" sind die selbst gewählten Schlüssel, bzw. Suchbegriffe mit denen die Seite in Suchmaschinen einsortiert wird. Anders gesagt, wenn ich nach dem Suchbegriff *Schlüsselwort1* suche, sollte mir die Suchmaschine (hoffentlich) meine Seite als *Fund* anbieten. Fülle ich meine Variable *keywords* nicht auf, werden von den Suchmaschinen beliebige Begriffe aus den ersten 100 Zeichen des Inhaltes der Seite gewählt und man erhält eher *Zufallstreffer*.

Auf der anderen Seite wird dieses Feld auch gern missbraucht, indem häufige, alltägliche Suchbegriffe auf recht *halbseidene* Web-Seiten führen. So kann man schon mal auf der Suche nach der *Tour de France* bei *Oralsex* landen.

```
<meta name="keywords"
      content="Schlüsselwort1 Schlüsselwort2 Schlüsselwort3">
```

name="language" gibt an in welcher Sprache die Seite hauptsächlich verfasst ist. Dies kann z.B. *de* für deutsch, *en* für englisch, *fr* für französisch usw. sein. Wer Spass daran hat, kann sich die entsprechenden Normen *ISO 639-1* und *ISO 639-2* unter

<http://www.loc.gov/standards/iso639-2/>

anschauen und danach eine Seite in *Plattdeutsch*, *Nordfriesisch* oder *Mittelhochdeutsch* verfassen und entsprechend klassifizieren. *Alt-Occitan*, *Alt-Provençal* oder *Baskisch* hätte auch was. ☺

```
<meta name="language" content="de">
```

name="copyright" wer hat die Rechte an dieser Seite. Selbstverständlich gelten *Copyright*, *Urheberrecht* usw. auch für Inhalte von Web-Seiten. Es ist eine irrige Annahme, dass man, was im Internet frei lesbar ist, auch einfach frei *verwursten* darf. Die Seiten sind genauso wie andere Quellen (Bücher, Artikel, Bilder usw.) nach geltenden Regeln für Zitat und Referenzen zu behandeln.

Verstöße können *teuer* werden. Nach aktueller Rechtsprechung kann dies nicht nur einen *Schadensersatz* kosten, sondern **zusätzlich** sogar *Schmerzensgeld*.

```
<meta name="copyright" content="Jobst-Hartmut Lueddecke">
```

name="page-topic" hier gibt man an, um welches Thema, bzw. Fachgebiet es in dieser Seite geht.

```
<meta name="page-topic" content="worum geht es in der Seite">
```

name="audience" wer ist die Zielgruppe dieser Seite.

```
<meta name="audience" content="all">
```

name="robots" die *Robots* sind Programme der Suchmaschinen, die die Bestände der Web-Server, sowie die Inhalte der Seiten *scannen* um sie indexiert in die Suchmaschine aufzunehmen. Mit diesem Eintrag kann man gewisse Spielregeln für die *Robots* vorgeben. Allerdings ist dies keine Garantie dafür, dass sich die Suchmaschinen auch wirklich daran halten. Folgende Werte sind möglich:

index diese Seite **soll** in den Index der Suchmaschine aufgenommen werden.

noindex diese Seite **darf nicht** im Index der Suchmaschine aufgenommen werden.

follow der *Robot* soll beim Scan den Links auf der Seite folgen und damit auch die Verweise *abgrasen*.

nofollow die Verweise auf der Seite gehen die Suchmaschine nichts an und der *Robot* soll den Links nicht folgen.

```
<meta name="robots" content="index, follow">
```

name="revisit-after" Einladung für den *Robot* zu einem neuen Besuch im angegebenen Zeitabstand. Dies ist aber auch keine Garantie, dass er dann auch wirklich kommt.

```
<meta name="revisit-after" content="10 days">
```

1.2 Klassifikation nach dem Dublin-Core

Der *Dublin-Core* ist eine Vereinbarung, die wie der Name schon sagt bei einer Konferenz in Dublin, getroffen wurde, um eine genauere Klassifikation zu ermöglichen. Es ist eine gute Idee einfach beide Möglichkeiten parallel zu verwenden.

name="DC.Title" der Titel unter dem die Seite aufgenommen werden soll. Am Besten man nimmt den gleichen, möglichst selbst redenden Wert, wie im `<title>`. Der Begriff *Homepage* ist keine gute Idee, davon gibt es schon ein *paar*.

```
<meta name="DC.Title" content="Titel der Seite">
```

name="DC.Creator" Urheber der html-Seite. und damit Inhalte-Anbieter (Content-Provider) im Sinne des *Informations- und Kommunikationsdienstegesetzes*.

```
<meta name="DC.Creator" content="Autor">
```

name="DC.Publisher" der Herausgeber bzw. Verleger dieser Seite. Dies kann der Autor selbst, ein Verlag, eine Hochschule usw. sein. An dieser Stelle die Stichworte *Nutzungsrecht* und *Veröffentlichungsrecht* die nicht unbedingt beim Autor liegen müssen.

```
<meta name="DC.Publisher" content="Verleger">
```

name="DC.Contributor" wichtige Personen, die für dieses Werk etwas beigetragen habe. Wer hat die Übersetzung, oder die Illustrationen gemacht usw.

```
<meta name="DC.Contributor" content="Beitragender">
```

name="DC.Rights" wer hat die Rechte an der Seite. Siehe auch Copyright. Es muss nicht immer der Autor sein, die Verwertungsrechte kann man auch verkaufen, vererben usw. Wem ist schon bewusst, dass die Verwertungsrechte einer Diplom- und Bachelor-Arbeit bei der Hochschule und nicht beim Autor liegen?

```
<meta name="DC.Rights" content="Autor">
```

name="DC.Subject" eine Art *Betreff* der Seite, also worum geht es in der Seite. Dies ist ähnlich einem *Betreff* in einem Formbrief zu sehen.

```
<meta name="DC.Subject" content="Betreff">
```

name="DC.Description" Kurzbeschreibung der Seite. Siehe auch *page-topic*.

```
<meta name="DC.Description" content="Kurzbeschreibung">
```

name="DC.Coverage" ergänzende Informationen zum Hintergrund. Wo und wann hat sich die auf der Seite beschriebene Begebenheit abgespielt. In welchen Kontext ist diese Seite zu sehen usw.

```
<meta name="DC.Coverage" content="Hintergrundinformation">
```

name="DC.Date" Stand der Seite, also wann ist diese Version der Seite das letzte Mal geändert worden.

```
<meta name="DC.Date" content="2009-11-13">
```

name="DC.Type" Was ist der Typ der Seite (Text, Bild, Ton usw.). Siehe auch *content-type*.

```
<meta name="DC.Type" content="text">
```

name="DC.Format" Format der Seite. Besonders interessant bei Bild- und Ton-Dokumenten. Siehe auch *content-type*.

```
<meta name="DC.Format" content="text/html">
```

name="DC.Identifier" die eigentliche *URL* der Seite. Da es vorkommen kann, dass Seiten in *Frames* *reingequetscht* werden, ist die *echte* Adresse der Seite schon wichtig.

```
<meta name="DC.Identifier"
content="http://www.bui.haw-hamburg.de/pers/jobst-hartmut.lueddecke/">
```

name="DC.Source" Quelle, auf die sich die Seite bezieht.

```
<meta name="DC.Source" content="Quelle">
```

name="DC.Language" siehe auch *language*.

```
<meta name="DC.Language" content="de">
```