

# Wissens- und Kommunikations-Management mit Drupal

Carsten Logemann

paratio.com e.K.  
Gr. Eschenheimer Str. 41a  
60313 Frankfurt am Main  
cl@paratio.com

**Abstract:** Der Vortrag vermittelt einen Gesamt-Überblick über die Möglichkeiten zur Realisation individueller Konzepte für Wissens- und Kommunikations-Management mit Drupal. Insbesondere wird auf diverse sehr flexible Module verwiesen, die es ermöglichen, selbst komplexe Webanwendungen ohne Programmier-Kenntnisse zu erstellen oder zumindest einfacher erweitern und pflegen zu können.

## 1 Grundlagen

### 1.1 Kontext

Wenn Verweise in Hypertexten einen bestimmten Kontext benötigen, werden diese oftmals entsprechend beschrieben. In komplexeren Konzepten zum Wissensmanagement würde dies die Lesbarkeit beeinträchtigen. Andere Möglichkeiten der Kennzeichnung oder der „Auslagerung“ aus dem Fließtext wie z. B. mit Fußnoten sind sehr sinnvoll und mit Drupal leicht zu realisieren. Innerhalb von Texten können „**Verweise zu Wortbedeutungen**“ eindeutigen Kontext herstellen, wie es in Enzyklopädien schon lange vor Wiki-Systemen angewendet wurde. Eine „**Assoziative Verknüpfung**“ z. B. durch Schlagworte (Tags) können ebenfalls Kontexte herstellen. Auch Referenz- und Bookmark-Listen ermöglichen es, größere Zusammenhänge darstellen und vermitteln zu können. Dieser Ansatz wurde von Vannvar Bush schon 1945 im *Memex*-Konzept [Bu45] beschrieben. Das *Resource Description Framework (RDF)*<sup>1</sup> als Teil des *Semantic Web*-Konzepts ermöglicht Meta-Kontexte für Text-Einheiten und andere Medien-Inhalte. Um der Mehrdeutigkeit natürlich-sprachlicher Texte entgegen zu wirken, könnte Kontext noch umfangreicher hergestellt werden, indem Texte selbst stärker organisiert werden z. B. mit künstliche Sprachen oder dem *Ikorbo*-Konzept<sup>2</sup>.

<sup>1</sup> Siehe RDF - Standard des W3-Consotiums: <http://www.w3.org/RDF/>

<sup>2</sup> Ausführlich in der Diplomarbeit vom Autor [Lo03] beschrieben.

## 1.2 Statische Informationen

Am Anfang des WWW gab es ausschließlich statische Informationen auf der Basis von HTML-Dateien oder damit verknüpfter Medien wie z. B. Bilder, die mit Hilfe einer Webserver-Software an Browser ausgeliefert werden. Insbesondere bei Medien-Dateien und die inzwischen vom Text-Content getrennten CSS-Gestaltungsinformationen ist dieses Datei-Konzept aus Performance-Gründen noch sinnvoll. Content-Management nur auf HTML-Dateiebene ist dagegen nicht sehr flexibel. Dafür kann dieses – auch aus Sicherheitsgründen – ferngesteuert von anderen Rechnern erfolgen.

## 1.3 Dynamische Webanwendungen

Die dynamische Verwaltung und Zusammenstellung von Inhalten kann nur mit Webanwendungen funktionieren. Neben einer gewissen Vereinfachung der HTML-Erzeugung erfordert z. B. eine Kommunikations-Plattform zwingend die dynamische Generierung und Modifikation von Inhalten. Ausgefeilte Caching-Techniken (die für Drupal verfügbar sind) helfen insbesondere die nötigen Datenbanken zu entlasten. Da Informationen ständig verändert werden können – vor allem in einer gewollten Mehr-Benutzer-Konfiguration, wie z. B. in Foren – bekommt Sicherheit hier eine große Bedeutung. Auch die Verwaltung statischer Medien-Inhalte kann mit Webanwendungen wie Drupal erfolgen.

## 1.4 Zeit und Kosten im Rahmen

Frameworks sparen Zeit sowohl bei der Erstellung als auch bei der späteren Pflege einer Webanwendung. Je nach Einsatzzweck sollte bedacht werden, dass Frameworks zu unflexibel und zu langsam sein können. Drupal bietet neben flexiblen Programmier-Schnittstellen ein umfangreiches Sortiment an fertigen Modulen. Auch zur *Usability*-Verbesserung mit Javascript gibt es diverse Frameworks, von denen *jQuery* schon direkt in Drupal integriert ist. Für die schnelle Programmierung von Templates gibt es (X)HTML/CSS-Frameworks, von denen z. B. Zen und YAML bereits für den Drupal-Einsatz vorbereitet sind.

# 2 Kommunikation von Menschen und Maschinen

## 2.1 Individualität und Privatheit

Im WWW ging es zunächst ausschließlich um die Veröffentlichung von Informationen. Innerhalb von Kommunikations-Systemen ist die Zugriffssteuerung sehr bedeutsam. Insbesondere auch die individuelle Verarbeitung von Wissen wie z. B. Persönliche Notizen und Bookmarks finden zunehmend im Internet statt. Zudem gibt es auch in Drupal diverse Aufzeichnungen über das Verhalten von Benutzern.

Diese können zum Teil technisch notwendig sein oder auch Statistiken dienen, die sich oft deaktivieren lassen. Auch Konzeptbedingt können besondere Informationen über Benutzer gesammelt werden. In einem Lern-System können dies z. B. Lernfortschritte und Test-Ergebnisse sein. Obwohl Drupal es ermöglicht, den Zugriff auf Informationen sehr fein einstellen zu können, ist zu bedenken, dass private Informationen doch öffentlich werden können. Neben Fehlbedienungen kann es auch Fehlfunktionen geben oder ein Angriff auf eine Web-Anwendung könnte mal erfolgreich sein. Selbst ein als *Intranet* angelegtes System kann ebenso angegriffen werden, wie ein privater Computer.

## 2.2 Soziale Organisation

Drupal bietet sehr viele Möglichkeiten zur sozialen Organisation von Benutzern. Angefangen bei einem Rollen- und Rechte-Management gibt es diverse Möglichkeiten, soziale Beziehungen (z.B. Freundeslisten) darzustellen. Die Möglichkeit offene Gruppen oder geschlossene Teams zu bilden bietet z. B. das Modul *Organic Groups*. Diese Gruppen lassen sich wiederum mit Lese- und Schreibrechten zu verknüpfen. So ist z. B. Redaktions-Workflows oder eine kooperative Inhalts-Bearbeitung möglich.

## 2.3 Zugänglichkeit der Maschine

Durch die Organisation und Präsentation von Wissen im Internet bekommt die technische Zugänglichkeit dazu eine massive Bedeutung, insbesondere, wenn man ein Recht auf Bildung als das „*Recht auf die totale Beteiligung an dem erreichten Weltwissen*“ [Ba93, S. 5] versteht. Zur **Aufrechterhaltung des Server-Betriebs** und der Webanwendung – nebst Anbindung an das Internet – ist zunächst die **Zugänglichkeit der Inhalte** (Barrierefreiheit) von Interesse, für die es zum Teil auch rechtliche Vorschriften gibt<sup>3</sup>. Bezüglich interaktiver Web-Anwendungen und dem Trend zu Social Media wird auf den zunehmend auch die **Zugänglichkeit der Funktionen** bedeutsam<sup>4</sup>.

## 2.4 Aktion und Reaktion

Drupal ist so aufgebaut, dass an sehr vielen Abläufen mit dem so genannten Hook-System programmiertechnisch angeknüpft werden kann. Schon das Kern-System bietet einfach konfigurierbare Möglichkeiten, wie z. B. Kontakt-Formulare oder einen Moderations-Workflow. Auch gibt es die Möglichkeit Aktionen auszuführen, wenn bestimmte Ereignisse eintreten z. B. E-Mail-Versand, User sperren oder ähnliches. Neben vielen Modulen für bestimmte Aufgaben bietet das Rules-Modul ein komplexes Aktion-Reaktion-Framework. Da Rules wenn nötig per Programmierung erweitert werden kann, sind eigene Module für eine individuelle Automatisierung oft nicht notwendig.

Für zeitbasierte Aktionen benötigt Drupal einen Taktgeber der vom Betriebssystem kommen oder auch durch den Aufruf der Datei „cron.php“ erfolgen kann.

<sup>3</sup> In Deutschland z.B. im BITV geregelt: <http://www.gesetze-im-internet.de/bitv/index.html>

<sup>4</sup> Siehe z.B. „BIENE nimmt Forschungssemester“: <http://www.einfach-fuer-alle.de/biene-2008/#biene08-pm01>

## 3 Ordnen und Erweitern mit Daten-Strukturen

### 3.1 Inhalts-IDs und Node-Typen

Sämtliche Inhalte inkl. Benutzer und Ereignis-Protokolle werden mit eindeutiger ID-Nummer in Drupal verwaltet. Das zentrale Inhalts-Konzept sind so genannte *Nodes* (Knoten) mit denen diverse Meta-Informationen verknüpft werden. Aufgrund der Komplexität von Nodes – die sich zudem durch diverse Module erhöht – sind Nodes aus Performancegründen nicht in jedem Fall eine sinnvolle Wahl zur Speicherung von Informationen, wie z. B. Bei einfachen Ereignis-Protokollen.

Das Kern-System von Drupal erlaubt die Definition eines Node-Typs als **elementare Inhalts-Unterscheidung**. Dies ermöglicht es z. B. auch Wissens-Arten voneinander zu unterscheiden, wie z. B. Erfahrungsberichte, FAQ, Definitionen oder konkrete Anleitungen.

Viele Module erzeugen spezielle Node-Typen mit bestimmten Funktionen. Die vom Kern-System bereitgestellten Basis-Felder Textkörper und Titel lassen sich insbesondere mit dem Content Construction Kit (CCK) – das inzwischen Teil des Drupal 7 Core ist – leicht erweitern. Neben vielen funktionalen Feldern sind oftmals auch weitere Text-Felder sinnvoll, um z. B. auch Text-Arten innerhalb eines Nodes zu unterscheiden, wie z. B. weiterführende Informationen zu den eigentlichen Aussagen in einem Node.

### 3.2 Schlagwörter, Lesezeichen und Inhalts-Gruppen

Drupals Taxonomie-System ermöglicht es, Nodes mit Tags (Schlagworten) zu versehen. Diese werden in Vokabularen organisiert und können z. B. mit Nutzer-Rollen und Node-Typen verknüpft und **hierarchisch oder flach** organisiert werden. Es können **redaktionell** definierte neben **offenen Vokabularen** verwendet werden, die ein „free tagging“ erlauben. Synonyme innerhalb einer Sprache oder die komplette Übersetzung von Tags (Terme) ist ebenfalls möglich. Auch CCK erlaubt ein Schlagwort-Konzept. Die Verwendung mehrerer Vokabulare und/oder CCK-Feldern ermöglicht eine **Facettenklassifikation** und die entsprechende Suche danach (siehe 4.4).

Für individuelle Lesezeichen oder evtl. auch nur temporäre Kennzeichnungen gibt es z. B. das Flag-Modul. Man kann Nodes oder Benutzer in Listen zusammen zu führen oder direkt auf besondere Weise aufeinander verweisen lassen. Letzteres ermöglicht hierarchische Eltern-Kind-Beziehungen oder auch Verbindungen in einer „Ebene“.

### **3.3 Semantik und Meta-Informationen**

In Texten können Überschriften, Listen und andere sinnvollen HTML-Auszeichnungen mit WYSIWYG-Editoren eingefügt werden. Für die Verwaltung von Meta-Tags (Autor, Schlüsselwörter, Beschreibung usw.) und RDF-Informationen in Nodes gibt es ebenso Module zur Verwaltung. Für die automatische Ergänzung bzw. Veränderung von Texten mit Meta-Informationen z. B. auch für eine Verknüpfung mit einem Glossar oder Verknüpfung von Akronymen mit ihren „Übersetzungen“ gibt es fertige Module. Das Filter-System (siehe 4.1) ermöglicht eine einfache Realisation einer individuellen, automatisierten Verwaltung von Meta-Informationen.

### **3.4 Elementare Annotation**

Eine elementare Annotation von Texten z. B. wie im Ikorbo-Konzept des Autors beschrieben [Lo03] würde eine komplette, granulare Veränderung von Texten mit sich bringen. Auch hierzu ließe sich Drupal nutzen, da Inhalte sowohl vor dem Abspeichern in der Datenbank als auch vor der Ausgabe komplett per Programmierung verändert werden können.

## **4 Suchen, Ersetzen und Anzeigen**

### **4.1 Dynamischer Inhalt**

Innerhalb von Text-Feldern können Benutzer unterschiedliche Eingabe-Formate nutzen, die mit unterschiedlichen Editoren und Filtern verbunden sind. Das Filter-System ermöglicht es, Inhalte vor der Ausgabe zu modifizieren ohne die Nodes zu verändern. Neben der Ergänzung von Texten mit Zusatz-Informationen (siehe auch 3.3) können auf diesem Weg vor allem problematische HTML-Befehle ausgefiltert und z. B. E-Mail-Adressen maskiert werden. Individuelle Filter – auch zum Ersetzen von Text-Fragmenten – werden mit entsprechenden Modulen ohne Programmierkenntnisse ermöglicht.

### **4.2 Volltext-Suche**

Der Drupal-Kern ermöglicht eine Indexierung und Gewichtung von Inhalten für die Volltext-Suche. Da diese nicht sehr schnell ist, lassen sich externe Volltext-Such-Technologien wie z. B. *Apache Solr* einbinden. Ein globales Suchen und dauerhaftes Ersetzen von Texten in der Datenbank ist mit dem Modul *Search and Replace Scanner* möglich.

### 4.3 Datenbank-Ansichten und Massen-Bearbeitung

Ein einfacher Weg zum Suchen, Laden, Filtern, Sortieren und Anzeigen von Informationen aus der Datenbank bietet das Views-Modul ganz ohne Programmierung. Auf diesem Weg lassen sich vor allem Listen unterschiedlicher Art erzeugen, wie z. B. „neueste Kommentare“. Das Modul *Bulk-Operations* zum Bearbeiten der angezeigten Informationen, erlaubt z. B. das schnelle Erstellen von Moderations-Werkzeugen.

### 4.4 Finde-Maschine

Die Unterscheidung von Inhaltstypen ermöglicht entsprechende Suchfilter auch in der internen Volltext-Suchmaschine, die das Kern-System bereit stellt, wie auch die Suche nach Autoren und Tags.

Das Views-Modul ermöglicht es zudem, diverse Filter auch als Anzeige-Modifikation der Datenbank-Abfrage zu definieren. Diese beschriebenen Möglichkeiten sind vor allem effizient, wenn die/der Suchende schon weiß, was er/sie sucht. Das Konzept der Facettensuche, bei dem eine Volltext-Suche zunächst alle Ergebnisse präsentiert, aber dazu diverse Facetten wie Tags, Autoren Inhaltstypen oder CCK-Felder als mögliche Filter selbständig präsentiert, ist auch mit Drupal möglich. Die Facettensuche ist bei erfolgreichen Webshops schon lange im Einsatz<sup>5</sup>.

## 5 Meta-Vernetzung von Websites, Web-Diensten und Desktops

Wenn Web-Dienste eine externe Kommunikation ermöglichen, gibt es häufig Module, um diese mit Drupal zu verbinden. Auf der anderen Seite gibt es viele Daten-Schnittstellen, die Drupal selbst bereit stellen kann, um Informationen auf unterschiedliche Weise auszugeben oder entgegen nehmen zu können. Insbesondere RSS- und iCal-Feeds eignen sich zum einfachen und plattformübergreifenden Daten-Austausch. Drupal kann z. B. auch sehr flexibel mit dem Views-Modul diese Daten bereit stellen und z. B. mit dem Feeds-Modul sehr flexibel automatisch einlesen und weiter verarbeiten.

## Literaturverzeichnis

- [Ba93] Ballauff, Theodor: „Über die Unerläßlichkeit der Bildung“. In: Borelli, Michele (Hrsg.): „Deutsche Gegenwartspädagogik“, Seite 1-19, Baltmannsweiler, 1993.
- [Bu45] Bush, Vannevar: As We May Think, The Atlantic, July 1945. URL [2010/07/09]: <http://www.theatlantic.com/past/docs/unbound/flashbks/computer/bushf.htm>
- [Lo03] Logemann, Carsten: „Wissensorganisation mittels Bedeutungsreferenzen und Beziehungsformeln“ (Diplomarbeit), 2003. PDF-Version: <http://embia.org/liath/liber/scio>
- [VD09] van Dyk, J. K.: Das Drupal-Entwicklerhandbuch. Addison-Wesley, München, 2009.

<sup>5</sup> Siehe z.B. <http://www.amazon.de/> oder <http://www.ebay.de/>