

Chapter XX

Issues to Consider when Choosing Open Source Content Management Systems (CMSs)

Beatrice A. Boateng

Ohio University, USA

Kwasi Boateng

University of Arkansas at Little Rock, USA

ABSTRACT

This chapter examines the main issues that have to be considered when selecting an open source content management system. It involves a discussion of literature and the experiences of the authors after installing and testing four widely used open source CMSs (Moodle, Drupal, Xoops, and Mambo) on a stand-alone desktop computer. It takes into consideration Arnold's (2003) and Han's (2004) suggestions for the development of CMSs, and identifies six criteria that need to be considered when selecting an open source CMS for use.

INTRODUCTION

Content management systems (CMSs) have gained prominence and are used for database-driven Web sites for all kinds of electronic communication activities.¹ A content management system is a nonstatic, dynamic, database-driven system that is used for electronic management and the publishing of information and resources in an organized manner. The features of a CMS-run Web site permit Web site administrators and authorized users to log into an electronic system

to author or approve posted materials. Similarly, they can facilitate access to archival, confidential, or password-protected materials that are hosted on the Internet.

The emergence of open source software (OSS) applications and the culture of making source code available for all to use is causing a stir in the software development industry and among software users. According to the Open Source Definition (OSD) Web site (<http://www.opensource.org/docs/definition.php>), an OSS ap-

plication generally complies with a set of criteria that include the following:

- Source code should be available to users.
- Software is available for free download and distribution.
- Users should be able to modify source code and possibly create new applications.
- Software is available under the “copyleft” licensing agreement (http://www.debian.org/social_contract).

In this chapter, we examine the main issues that have to be considered when selecting an open source content management system. We draw upon the literature and our experiences after the installation and testing of four widely used open source CMSs, namely, Moodle,² Drupal, Xoops, and Mambo, on a stand-alone desktop computer. Through our installation, we were able to verify the installation processes, and understand how the back end of the select CMSs work in order to address issues that a potential adopter of open source CMS should consider. We chose Mambo, Xoops, Drupal, and Moodle based on the fact that these CMSs come up often in discussions and the literature on open source CMSs. Also, from our observation, these CMSs have well-organized product software, support, documentation, training, integration, and professional services. These are standards considered necessary for determining the maturity of OSS as determined by Golden (2005). Also, information available on the CMSMatrix Web site, a site for the discussion, rating, and comparison of content management systems, prominently feature Mambo, Xoops, Drupal, and Moodle among the frequently used and efficient systems based on system requirements, security, support, ease of use, performance, management, interoperability, flexibility, built-in application, and commerce (<http://cmsmatrix.org>). Our examination of information on CMSMatrix on the four CMS candidates indicates that on the average, Drupal and Xoops are rated higher on all

the criteria. Similarly, Mambo, Xoops, Drupal, and Moodle are listed on the site <http://opensourcecms.com>, a Web site that displays a comprehensive list of open source CMSs, and provides a free administrative platform for real-time testing of various open source systems.

BACKGROUND: WHAT ARE CMSs AND HOW DID THEY EMERGE?

The need for online platforms ideal for the dissemination of information, document delivery, and electronic archiving of materials has necessitated the development of content management systems that support the publishing of materials in different and easily accessible electronic formats. Discussing CMSs as tools for such online organization of materials, Arnold (2003) opined that “when the needs and requirements [for the electronic delivery of materials] are understood, a system to manage the creation, approval, and dissemination of text, images, and even streaming video can make life in today’s fluid environment somewhat more orderly” (pp. 36-37). The development of CMSs is relatively new in the software industry (Arnold; Han, 2004). The origins of CMSs lie in (a) the records management field and (b) the need to move content from the desk of the creator to an organization’s Web site (Arnold). Performing these two tasks effectively in a 21st century online environment could be daunting. CMSs usually comprise of modules and/or components as add-ons of an application that allow a programmer or an adopter of such software to build an electronic system that could be used to perform various functions electronically. The functionality of a CMS is dependent on the structure of the CMS and the processes that occur within the CMS (Friedlein, 2003). For instance, the CMS could be configured to allow a system administrator and users to manage, compile, and publish resources, as well as to facilitate online interaction among users as illustrated in the process part of Figure 1. Figure

12 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/issues-consider-when-choosing-open/21193

Related Content

Project Management

Barbara Russo, Marco Scotto, Alberto Sillittiand Giancarlo Succi (2010). *Agile Technologies in Open Source Development* (pp. 287-300).

www.irma-international.org/chapter/project-management/36508

MEPES: Methodology for Evaluating the Performance of E-Mail Servers

Pedro Alexis Torres Calderónand Emigdio Antonio Alfaro Paredes (2018). *International Journal of Open Source Software and Processes* (pp. 47-64).

www.irma-international.org/article/mepes/221363

Dynamic Key-Based and Context-Aware Internet of Things Authentication Approach

Mihir Mehtaand Kajal Patel (2022). *International Journal of Open Source Software and Processes* (pp. 1-15).

www.irma-international.org/article/dynamic-key-based-and-context-aware-internet-of-things-authentication-approach/310939

Bridging the Gap between Agile and Free Software Approaches: The Impact of Sprinting

Paul J. Adamsand Andrea Capiluppi (2009). *International Journal of Open Source Software and Processes* (pp. 58-71).

www.irma-international.org/article/bridging-gap-between-agile-free/2771

A Model of Cultural Competence in Open Source Systems

Doris Wright Carroll (2013). *Open-Source Technologies for Maximizing the Creation, Deployment, and Use of Digital Resources and Information* (pp. 1-10).

www.irma-international.org/chapter/model-cultural-competence-open-source/70115