

## Chapter 2.8

# Building an LMS with Ubiquitous Software

**Michael Rees**

*Bond University, Australia*

**Charles Herring**

*G-Netech Pty Ltd, Australia*

### ABSTRACT

Teaching institutions around the world are using large, unwieldy, and expensive learning management system (LMS) packages that are beginning to have profound effects on their whole organizations. Such LMS packages in turn go to great lengths to interoperate with the desktop information productivity software that almost all institutions use, Microsoft Office System. Since a very large part of the instructional content is generated in Office, it seems sensible to investigate whether straightforward extensions of the Office System could become an LMS in their own right. This chapter describes research and development that integrated Microsoft Office System, SharePoint Windows Services, and SharePoint Portal server (SPS) as the heart of an off-the-shelf LMS. Already designed to work closely with Office, SPS features are compared against the list of features expected of an ideal LMS. Where gaps

in the LMS features were discovered in SPS, a number of small extensions of standard Office applications were proposed to fill these gaps and create a credible LMS. These Microsoft tools and custom extensions were put to use in teaching and administration during a 2-semester (8-month) trial at Bond University. The SPS installation was hosted in partnership with G-Netech Pty Ltd. This Bond University/G-Netech SharePoint Alliance project (BUGSA or SharePoint Alliance) was able to call upon combined research, development, and teaching expertise provided by the partners. The outcomes of the short trial support the concept of the Office System as a viable LMS.

### INTRODUCTION

When one surveys the choice of learning management system (LMS) software, one finds a very lop-sided situation. What were the two leading examples of a LMS deployed across the higher edu-

cation institutions of the world, WebCT (WebCT, 2006) and Blackboard (Blackboard Academic Suite, 2006), have now merged into one company although for the time being the two products are still differentiated. Together these products account for 70-80% of the total market. Significant in the context of this chapter is that with a decade of development both products are available in enterprise versions comprising collections of major components, which include content management systems, portals, and communication tools. Such all-encompassing feature sets inevitably lead to high costs and significant training needs.

Despite such dominance, there is a long tail of alternative software for the primary LMS function. At the tail's end are the local in-house systems that each institution has developed with their own technical and teaching staff. Of late, there has been a significant effort by a group of 80 institutions worldwide to work together on producing a common learning environment called Sakai (2006). The Sakai Project uses an interesting variant of the open source development model called the community source model, which involves some financial inputs. What remains are open source systems with their growing group of users and several less significant commercial LMS packages that typically target the smaller institutions.

From their own product descriptions, the supplier companies extol the fact that an integrated LMS is an enterprise level system. Such systems by implication therefore require very significant support in terms of technical design, administration, and staff training in addition to content design and creation teams for the support of the teaching staff. It is not surprising that the founder of the WebCT LMS, Goldberg (2004), in his keynote presentation at AusWeb 2004 indicated that, in his experience for many institutions the LMS, has become the most important information system on a university campus after the payroll system! By implication, however, this entails a large cost in software acquisition, technical support,

ongoing maintenance, and integration with other institutional information systems.

It is the authors' contention that an important avenue of LMS development has largely been ignored, that of adapting existing off-the-shelf content management and interactive communication systems intended for use by organisations in general across all industry and government sectors. One major example of this class of systems is the Microsoft SharePoint technologies (2006) aimed at generic Web portal and information repository use. As will be described in detail in this chapter, a very great advantage of SharePoint is its tight integration with the ubiquitous Microsoft Office applications. These are the main information content creation tools used by educators across the world and are licensed by very many institutions.

In the pages that follow, a case is made for adopting SharePoint as the foundation of an LMS. Examples and case studies show how this can be achieved. Some new developments that are happening now, and planned changes to Microsoft Office over the next months, are discussed. This shows that future trends will lend credence to the use of off-the-shelf software to compete with the highly specific and less flexible LMS packages currently in use. Leveraging existing software in this way helps in reducing software acquisition costs and training needs.

## **BACKGROUND**

The authors have worked together for a number of years in a research centre environment where software development of new leading-edge solutions was the norm. Often these software packages were built from scratch unnecessarily, and the authors soon became proponents of the intelligent deployment of straightforward off-the-shelf software that can often be extended to provide support for new solutions. This comment applies to apparently mundane applications such

14 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/building-lms-ubiquitous-software/29414](http://www.igi-global.com/chapter/building-lms-ubiquitous-software/29414)

## Related Content

---

### Comparing Four-Selected Data Mining Software

Richard S. Segall and Qingyu Zhang (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 1750-1759).

[www.irma-international.org/chapter/comparing-four-selected-data-mining/29475](http://www.irma-international.org/chapter/comparing-four-selected-data-mining/29475)

### Handling of Software Quality Defects in Agile Software Development

J. Rech (2007). *Agile Software Development Quality Assurance* (pp. 90-113).

[www.irma-international.org/chapter/handling-software-quality-defects-agile/5070](http://www.irma-international.org/chapter/handling-software-quality-defects-agile/5070)

### A Comparative Analysis of Reliability Assessment Methods for Web-based Software

Jinhee Park, Yeong-Seok Seo and Jongmoon Baik (2013). *International Journal of Software Innovation* (pp. 34-47).

[www.irma-international.org/article/a-comparative-analysis-of-reliability-assessment-methods-for-web-based-software/103280](http://www.irma-international.org/article/a-comparative-analysis-of-reliability-assessment-methods-for-web-based-software/103280)

### Data Mining Techniques for Software Quality Prediction

Bharavi Mishra and K. K. Shukla (2013). *Designing, Engineering, and Analyzing Reliable and Efficient Software* (pp. 112-139).

[www.irma-international.org/chapter/data-mining-techniques-software-quality/74877](http://www.irma-international.org/chapter/data-mining-techniques-software-quality/74877)

### Simulation Modelling within Collaborative Spatial Decision Support Systems Using "Cause-Effect" Models and Software Agents

Raja Sengupta (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 1434-1445).

[www.irma-international.org/chapter/simulation-modelling-within-collaborative-spatial/29455](http://www.irma-international.org/chapter/simulation-modelling-within-collaborative-spatial/29455)