Burrokeet, an Application for Creating and Publishing Content Packages with support for Multiple Input and Output Formats

Margaret Bernard
The University of the West Indies, Trinidad

Anil Ramnanan
The University of the West Indies, Trinidad

ABSTRACT

This paper describes the design and use of Burrokeet, an e-learning software platform which facilitates the creation of IMS/SCORM Content Packages. Burrokeet will accept input content documents in most of the common file formats; it provides the tools to sequence that content into a pedagogically sound 'unit of learning' and to generate the IMS Manifest file for the Content Package. Burrokeet also has a publishing engine which can publish Content Packages created within Burrokeet or SCORM-compliant Content Packages generated from external sources. The Publishing engine will generate the output in a unified format, regardless of the input file formats used. Several output formats are supported, including html for website publication, pdf for print and presentation formats for delivery during class time.

1.0 INTRODUCTION

Learning Content Management Systems (LCMS) usually deliver the e-learning content in the format in which it was created; for example, if the content resource is a .doc document then the document is presented to the user as a Microsoft Word file, requiring the user to have the appropriate application software on their desktop. When multiple input formats are used for developing a course, the user may see some content in HTML, some in Word, some in Open Office formats and the presentation of the material to the user is not uniform. This means that viewing the content will usually require the software that was used to originally create the document. Also, a content creator will have to create multiple versions of the content themselves for different purposes e.g. they would have to format the content as slides for presentations then do a separate document in HTML for online delivery and another in PDF for print [1].

This paper describes an Open Source LCMS, Burrokeet, which allows the user to upload resources in multiple formats, sequence those resources and create an XML SCORM-compliant Content Package from those resources. SCORM [2] is the industry standard for packaging content on the web so that it can be shareable and vendor-neutral; it allows interoperability amongst Learning Management Systems. Burrokeet includes a publication engine that allows the user to publish the content in a unified output format with a standard navigation system. This allows the user to use any content creation tool that they are comfortable with to create the content and then use that single content package to publish the content in a unified format. The course designer can even provide multiple unified formats for delivery of the course, all from the same input documents; for example, the input documents may be a combination of say, Word2007 (.docx) , Open Office (.sxw), HTML, Notepad (.txt) and PowerPoint2007 (.pptx) documents. The course may be delivered as a website displaying all the content in HTML. On the display page the designer

may have a print version link which will provide all of the same content in PDF format; the same Content Package may be published in presentation format (e.g. Power Point .pptx) for use within a classroom for courses delivered using a blended learning approach; all of this from the same input source documents [Fig 1].
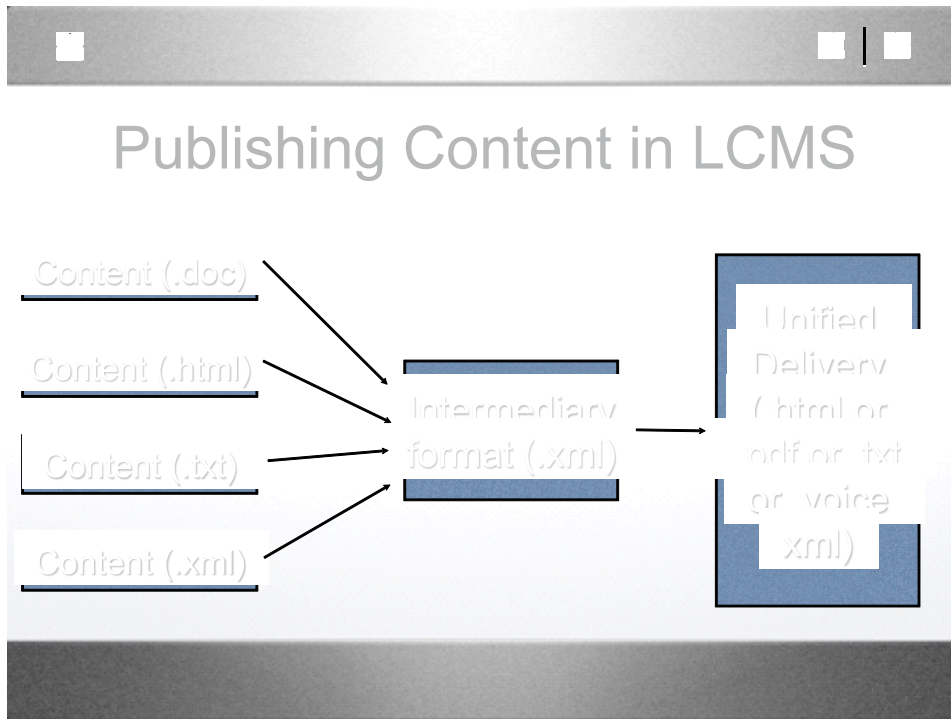

Figure 1: Framework for publishing content in Burrokeet

2. OVERVIEW OF BURROKEET

The Burrokeet LCMS application consists of two major components:

Content Package Creator

This system provides the tools necessary to create, store and reuse Learning Objects. The Learning Objects are digital content such as lecture notes, presentation slides, essays, case studies, lesson activities and examination materials within a pedagogical structure. The source documents for the Learning Objects may be in different input file formats; for example, source documents may be a combination of OpenOffice.org documents (.sxw), text documents (.txt), XML documents (.xml) or Word 2007 documents (.docx) or graphics or animation or even RSS Feed. Burrokeet provides the tools to sequence and structure these Learning Objects [3] and create a SCORM compliant Content Package (Figure 2). The software allows the instructional designer to add and remove resources to the Content Package and organize the sequence in which these resources would appear in the Content Package. The structure for the Content Package is stored in an XML file called, IMS Manifest, and the Content Package itself is based on the IMS/SCORM Content Packaging Information Model [4]. This is an international standard such that a Content Packages which conforms to the standard can be published in any Learning Management System (e.g. Moodle [5], Angel [6], ATutor [7], Blackboard/WebCT [8] and even Burrokeet [9] itself). The Content Package represents a unit of learning. It can be the entire course, or a course

module or even a single lesson (Figure 3)

Publication System

The publication system allows the user to present Content Packages in a user selected unified output format (Figure 4). Regardless of the original input formats of the source documents, the user can choose for the Content Package to be presented as a website (.html), or as a single PDF document for printing (.pdf), or as a presentation document for delivery in the classroom, or other unified output formats that may be added later (for example a format for a speech synthesizer for the visually impaired). The user decides which presentation output format is required in real time, and the Content Package is presented in that selected output format. The delivery of any output format can be over a network or on a CD-ROM for consumption where no network connection is available. Burrokeet's publication system will accept Content Packages not created in Burrokeet as long as they conform to SCORM standard. This means that the designer can search for suitable Content Packages off the web and import it into Burrokeet for reuse.

3. BURROKEET DESIGN

Burrokeet is an open source Java application developed using Eclipse Platform [10] for the base application and Apache Forrest [11] for the publication engine.

The Eclipse software is a software development platform comprising an Integrated Development Environment and a plug-in system to extend its functionality. The IDE was used to develop the Burrokeet application and several plug-in Java modules were written and integrated into one application. e.g. Editors for different document formats were included within the Burrokeet application by adding an editor plug-in for that application. Eclipse provides the runtime framework in which plug-ins are loaded, integrated, and executed.

The Burrokeet IMS editor was written from the ground up as a plug-in for the Eclipse Rich Client Platform. It is a graphical editor which allows the creation and sequencing of content into Content Packages. The Burrokeet IMS editor identifies the type of resource that is added to content package and adds it to the manifest e.g. an OpenOffice.org word processing file would have the type of "application/vnd-sun-xml-writer". This is important since this information is used by the publication engine to identify the source documents.

Apache Forrest is a publishing framework that transforms input from various sources into a unified presentation in one or more output formats. Apache Forrest has a robust plug-in feature that allows developers to create plug-ins to handle different types of input and output formats. There is an existing library of a large number of plug-ins that already handle different file formats and new plug-ins are being added on a regular basis. Most importantly, Apache Forrest had an IMS plug-in that would be able to allow Forrest projects to be mapped to IMS content packages. This meant that an editor could be built that can create a content package that can be read directly into Forrest for publication.

The Burrokeet publication system was written as an Apache Forrest Servlet engine. This plug-in creates an interface between the Eclipse Application and Apache Forrest. The Servlet engine allows the user to start and stop a Test Server (during development), build a site of static documents or build a Webapp for deployment on a Webserver.
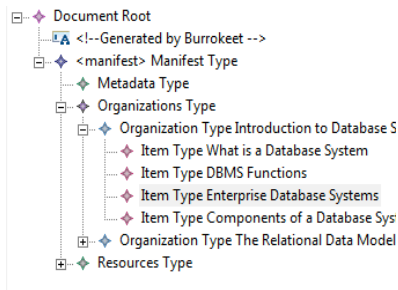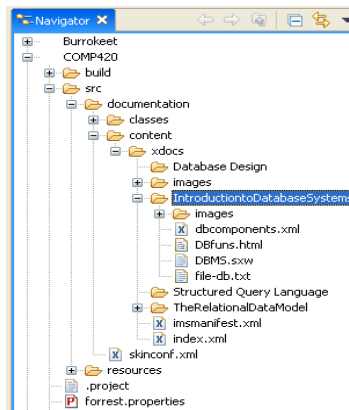
Figure 2:
Burrokeet Content Package View

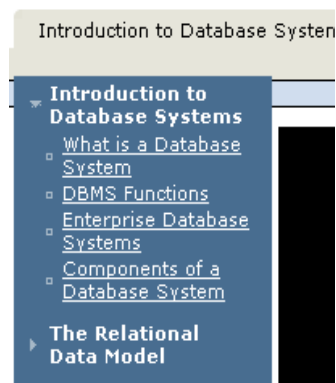*Figure 3:*
Physical Structure of Content Package

Figure 4:
Rendered View of Burrokeet Content Package

4.0 BURROKEET APPLICATIONS

Burrokeet was developed as a software system for creating and publishing learning content. It is a flexible tool which can be used for several other applications. It has been used a Website Development tool as well as a Document Management System.

4.1 Burrokeet as a Learning Content Management System

A University course in Computer Science was selected to test the capabilities of Burrokeet as a tool for developing and publishing learning content. Burrokeet was able to accept documents in most of the common file formats; the Burrokeet IMS Editor allowed the instructional designer to sequence these documents into a pedagogically sound course module and Burrokeet was able to successfully package them as a SCORM Content package with the built in sequencing mechanism. The Content Package was first tested using the course delivery features in Burrokeet but this was limited as Burrokeet did not include full User Management features. User management was handled externally using Apache's User Authentication system.

We also tested Burrokeet with other Learning Management Systems, namely ATutor, Moodle, WebCT, Angel (now Blackboard/WebCT/Angel). Burrokeet was able to import SCORM packages from these LMS as well as off the web. Content packages created in Burrokeet were successfully imported into ATutor (with some tweaking), Moodle and Angel.

4.2 Burrokeet as a Website Development System

Burrokeet's publication ability makes it a suitable tool for developing a website. The website content is built into the Content Package and Burrokeet uses 'Skins' from Apache Forrest to apply the design to the website. The skins describe the properties of the website and are the means used to add navigation and menu information to the content of a page as well as apply a consistent theme to the website for layout, colors etc. Forrest has a collection of default skins which are configurable and so can be adapted to a number of websites.

Burrokeet's source code was modified and extended to another open source project, EDGES, which is a tool for building websites for secondary schools. EDGES includes a

4

default Content Package with a structure that is customized for schools and includes skins that apply different themes and properties for the individual schools. EDGES allows the user to easily generate a school website which can be maintain by students by simply adding their own content. This was tested by using EDGES to build the website for St Augustine Girls High School. In November 2008, EDGES won an award under the Prime Minister's Awards for Innovation and Invention for its innovative use of a software platform to allow users with little knowledge of web development technologies to easily generate and maintain a website for schools.

4.3 Burrokeet as a Document Management System/ Print Publication Tool

Burrokeet can accept documents in several file formats, and allow the user to create a structure to those documents then publish the content in many output formats, including PDF. This ability makes it a useful tool for the generation of printed material. An interesting example of this use was the production of a postgraduate thesis. Over a period of about two years (when the research work was being done), the student uploaded content documents into Burrokeet as these documents were written. The Content Package was organized into the usual thesis format (Chapters, Sections etc). During the research period those documents were edited and re-edited (and re-sequenced) and could be viewed as a website (by the student and supervisor). Production of the final thesis did not require merging of long Word files (and all the associated problems of managing large text files). Rather, those same content documents that comprised the Content Package were simply publish as a PDF file and printed as the thesis.

5.0 CONCLUSION

Currently, Burrokeet has basic functionality but still requires refinement in terms of its usability and user management tools. However, the software has achieved its target in terms of functionality. It can accept input documents in many of the common formats and can create a Content Package that is compliant with industry standards. It can publish the Content Package in selected unified formats. The software design is extensible using the plug-in framework of Eclipse and thus can be extended to cater for other formats including voice.

Burrokeet is a software platform that manages content; it does this in an innovative manner but produces output that is standards based, making it interoperable with existing open-source and commercial LMS. It allows the content author to focus on the quality of the content and not have to worry about the delivery format.

REFERENCES

[1]   Robbins, Shelley R. *The Evolution of the Learning Content Management System*,   2002 http://www.learningcircuits.org/2002/apr2002/robbins.html

[2]   ADL. 2003. *Sharable Content Object Reference Model (SCORM) Version 1.3* http://adlnet.org

[3]   Becta 2005*. Packaging and Publishing Learning Objects: Best Practice Guidelines.*

http://www.becta.org.uk

[4]   IMS Content Packaging Information Model, Version 1.1.2

        http://www.imsproject.org/content/packaging/cpv1p1p2/imscp_infov1p1p2.html

[5]   Moodle Learning Management System

        http://www.moodle.org

[6]   Angel Learning Management System

        http://www.angellearning.com

[7]   ATutor Learning Content Management System

        http://www.atutor.ca

[8]   WebCT/Blackboard Learning Management System

        http://www.blackboard.com

[9]   Burrokeet Learning Content Management System

        http://burrokeet.sta.uwi.edu

[10]   Eclipse Development Platform

        http://www.eclipse.org

[11]   Apache Forrest Project

        http://forrest.apache.org