‘PENCasting’ – a standards-based approach to LMS / LCMS integration

Version 1.0, 30th June 2008
Ian Leader, Director of Consulting (ian.leader@line.co.uk)
Paul Nebel, Technical Consultant (paul.nebel@line.co.uk)

1. Abstract

Efficient and open interoperability between Learning Management Systems (LMSs) and content authoring and management platforms (authoring tools, LCMSs) would bring many benefits, to content authors, those who procure content, and learners alike. While there are already a number of approaches proposed to support this (PENS[1], OLSA[2]), we believe that the best approach would be one based on existing Internet standards, and propose ‘PENCasting’ which leverages elements of PENS alongside de facto Internet standards, RSS and PodCasting.

2. Problem definition

There are many tools for creating e-learning but most learners interact with e-learning through a Learning Management System (LMS), which delivers the content and tracks the learner's progress. We are seeing organizations increasingly adopt Learning Content Management Systems (LCMS) to author their e-learning, or manage content authored in other tools. There are numerous benefits to using an LCMS, including version management and better work flow, but their use does raise a problem – how to get new and updated content from the LCMS into the LMS.

3. Use cases

The primary use case we are considering is as follows:

- A content author or system administrator wishes to automatically publish new learning content from a content authoring or management environment to an LMS

In addition, we consider a number of secondary use cases:

- As for primary use case, but with the restriction that the LMS is located behind a corporate firewall, whereas the authoring environment is on the internet
- An administrator wishes to be alerted personally when new content is published
- An LMS should have a subset of all published content made available to it by a content authoring or management environment.

[1] See: http://pens.lmstesting.com

[2] See: http://www.skillsoft.com/about/press_room/press_releases/may_3_05_olsa.asp

4. Proposed solution

4.1. Existing Solutions and Issues

One proposed solution to this problem is PENS (Package Exchange Notification Services) – which has already been implemented in a number of LMSs and LCMSs. However, we have observed a number of practical problems with PENS:

1. The architecture of PENS means that it does not work well with an LMS behind a corporate firewall and an authoring tool outside that firewall

2. PENS uses, a 'push' model in which configuration of which systems should receive notifications of content publication must be done centrally in the PENS host. This puts a burden on the administrator of that system.

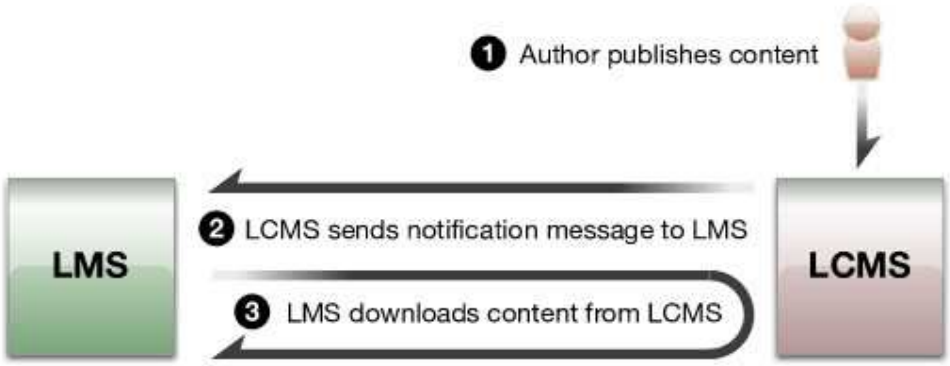

Figure 1 – PENS Push Architecture

4.2. Proposed Solution

We are suggesting an alternative implementation of LCMS to LMS publishing based on RSS/Podcasting, which we call 'PENCasting'[3], using a 'publish and subscribe' model to make central configuration of each recipient redundant.

Really Simple Syndication (RSS) is a ubiquitous de-facto standard for syndicating (or pushing out) information on the Internet. Virtually every blog, news site, or other web site that is regularly updated has an 'RSS feed'. Using an RSS reader, users can subscribe to these feeds, and get a notification when there is a new blog post, breaking news, or an update to a site.

RSS is also used to deliver Podcasts and Video Podcasts - the format is extensible, and additional fields were added by Apple so that it could be used to push audio and video files to iPods (and most other audio player devices).

The proposed architecture for PENCasting is shown in the diagram below.

[3] Neither LINE Communications nor this paper are affiliated with the AICC or the authors of PENS in any way.

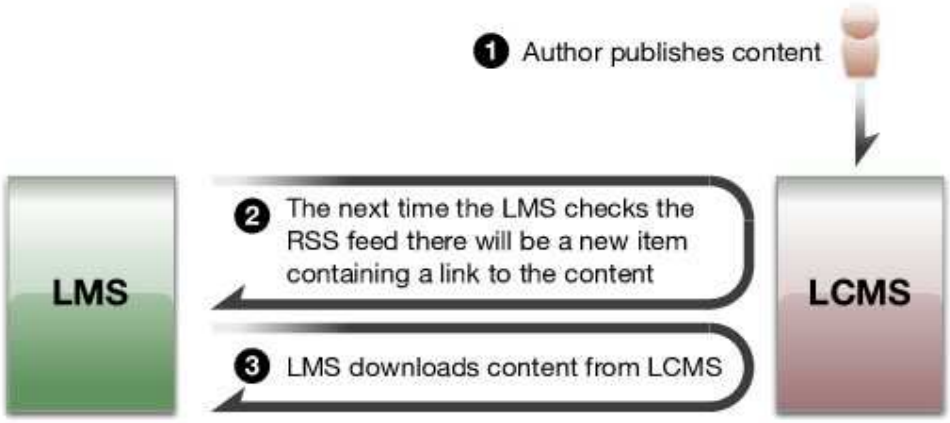

Figure 2 – PENCasting Publish and Subscribe Architecture

4.3. Solution Detail

The PENCasting data model allows LCMSs or other content publishing systems to announce the description and location of content packages that are available for transfer.

The PENCasting data model does not define which servers are notified, nor does it define what happens to any package that has been retrieved.

RSS formats are specified using XML. In the table below we summarize the tags comprising the model. The tag elements map to a selected sub-set of the fields in the PENS data model.

Element	Value	Description
<pens:package/>	Enclosing tag.	Specifies a content package. It has 4 required sub-elements: • **type** • **version** • **id** • **url** It has one optional sub-element: • **url-expiry**
<pens:type/>	string (required)	Allowable types of content packages: AICC assign-able unit (aicc-au), SCORM package (scorm-pif) or IMS package (ims-qti)
<pens:version/>	string (required)	Identifies the version of the packaging specification relevant for the package to be processed, e.g., for ADL SCORM “scorm-pif” packages, a system might use “1.2” or “2004”.
<pens:id/>	string (required)	Unique identifier required for package. id is be a Uniform Resource Identifier (URI) conforming to RFC standard 2396 and consisting of two parts; a globally unique namespace taken from the URL

		associated with the LCMS generating the ID plus an ID unique within the LCMS itself.
<pens:url/>	string (required)	Location of package archive ready for download (this will be a URI)
<pens:url-expiry/>	string	The content package is expected to be available for processing until at least the date and time specified (this will be a Coordinated Universal Time - UTC - conforming to ISO standard 8601)

An example of a PENCasting notification to be published via RSS is shown below (note the PENSCasting namespace in the RSS tag is hypothetical):

```
<?xml version="1.0" encoding="UTF-8"?>
<rssversion="2.0" xmlns:pens"http://www.pens.org/v1.2/">
        <channel>
                <title>Making Soup</title>
                <link>http://www.goodsoup.com</link>
                <description>Learn how to make great soup from professional
chefs.</description>
                <pubDate>Fri, 25 Apr 2008 13:00:00 +0100</pubDate>
                <item>
                        <title>Soup making techniques explained</title>
                        <description>Learn how to make any kind of soup using the
techniques employes by top soup makers.</description>
                        <pubDate>Fri, 25 Apr 2008 13:00:00 +0100</pubDate>
                        <enclosure
url="http://learning.goodsoup.com/scorm/cooking/techniques/makingsoup.zip"
length="5259231" type="pens/zip">
                                <pens:package>
                                <pens:id>http%3A%2F%2F \
learning.goodsoup.com%3A2631e419-1573-4720-b4c6-8701f960</pens:id>
                                        <pens:type>scorm-pif</pens:type>
                                        <pens:version>1.0</pens:version>
                                        <pens:url>http%3A%2F%2F \
myauthoringtool%2Fmakingsoup.zip</pens:url>
                                        <pens:url-expiry>2008-07-
22T06%3A51%3A29</pens:url-expiry>
                                </pens:package>
                        </enclosure>
                </item>
        </channel>
</rss>
```

5. Integration and other technical issues

The main technical issues with PENSCasting are actually issues with RSS itself. These can be summarised as follows:

- There are several different versions of RSS, falling into two major branches. This can lead to interoperability problems
- There can be a compatibility problem with HTML mark-up. Publishers have adopted the habit of placing HTML mark-up into the titles and descriptions of items in their RSS feeds. This behaviour has become expected of readers, to the point of

becoming a de facto standard, though there is still some inconsistency in how reading software handles this mark-up, particularly in titles

6. Summary and recommendations

We believe that there are several advantages to using PENCasting rather than the existing content distribution architectures:

1. Because RSS and Podcasting are so widely used, and a de-facto standard, they are already well understood by developers and users. Moreover, there are existing (free and open source) code-libraries which make implementation and testing quicker and easier.

2. PENCasting is scalable. There is no requirement for a central administrator of the LCMS to set up and maintain a list of web addresses to which notifications are sent. The subscription model of PENCasting lets recipients subscribe to the feed themselves just like any other RSS feed. What's more, this approach makes it easy for LMSs to choose what kind of information they wish to subscribe to: an LCMS could publish several different PENCasting feeds (e.g. sales training, after sales training and technical training). By just subscribing to the relevant feed, an LMS would only pick up content that was relevant to the audience it served.

3. RSS allows you to retrieve historical publications as easily as new ones. What happens if you deploy a new content consuming system, or a system is down when a content announcement is made? RSS feeds typically provide some history; so, even if a system is down for a whole day, it can catch up once it's back on line. You can also provide two RSS feeds: one which has, say, a week's worth of announcements, the other with a full history that can be used to 'seed' a new system.

4. RSS works transparently through firewalls. An LMS behind an organization's firewall can receive notifications from a hosted LCMS outside.

5. You can monitor new content publications using standard tools (the ubiquitous RSS reader) – so anyone can stay informed of new publications and download them if they desire. This also paves the way for PENCasting / PodCasting as a way of delivering e-learning content directly to mobile devices such as the iPhone or Google's Android initiative.

Therefore we recommend that SCORM 2.0 expands the scope of SCORM to cover content publishing from content producers to content consumers, and adopts PENCasting for this standard.

7. LINE Communication

LINE Communications is a learning and communications specialist. We advise our clients on every aspect of learning, communications, technology and related business issues. For almost 20 years, we have been working with clients in government, multinationals and the MoD, helping them develop and train their employees and partners and communicate with internal and external audiences. LINE has offices in London, Sheffield and Zurich. For more information see http://www.line.co.uk.