
THE ELECTRONIC STAFF RECORD PROJECT



NATIONAL HEALTH SERVICE

A-10000 NHS NATIONAL LMS PROJECT CONTENT GUIDANCE AND STANDARDS

Information Classification: COMMERCIAL IN CONFIDENCE

Author: Iain Harding
Owner: Stuart Fox
Creation Date: 10th March 2008
Last Updated: 11th March 2009
Document Ref: A-10000
Version: v 2.2

Approvals:

Name Lee Pacey

Title Head of Development (NHS)

Name Steve Thrussell

Title McKesson Solution Design

1. Document Control

1.1. Change Record

Date	Author	Version	Change Reference
04/02/2008	Iain Harding	0.1	Initial Draft
16/02/2008	Iain Harding	0.2	Redrafted after comments
20/02/2008	Iain Harding	0.3	Review of document
21/02/2008	Iain Harding	0.4	Second review of document
21/02/2008	Iain Harding	0.5	Awaiting approval and Sign Off
29/02/2008	Stuart Fox	0.51	Updated document following feedback
06/02/2008	Iain Harding & Keith Jackson	0.52	Updated following feedback from e-LfH
10/03/08	Stuart Fox	0.9	Amended to v 0.9 for official approvals process
11/03/08	Stuart Fox	1.0	v1.0 issued for sign-off
25/03/08	Stuart Fox	1.1	Classification changed to Commercial in Confidence
25/03/08	Stuart Fox	2.0	Issued for Sign-off
06/11/08	Nick Moseley	2.1	Addition of section covering packaging of content
11/03/09	James Haddon	2.2	Addition of section covering screen resolution and amendments to SCORM calls.

1.2. Reviewers

Name	Position
Nick Adcock	
Steve Thrussell	
Stuart Fox	
Keith Jackson (CfH)	
Michael Noel (Oracle)	

1.3. Distribution

Copy No.	Name	Location
1	Library Master	Project Library
2		Project Manager

2. Contents

1. Document Control.....	2
1.1. Change Record	2
1.2. Reviewers.....	2
1.3. Distribution.....	2
2. Contents.....	3
3. Purpose and Background	5
3.1. Client PC and Software specification	5
3.2. Content Standard	5
3.3. Accessibility Standard	6
4. The Oracle Learning Object Model	6
4.1. Learning Object Concepts	6
4.2. Topics.....	7
4.3. Topic Groups	7
4.4. Tests.....	7
5. Content Structure Options	7
5.1. Single Topic Course	7
5.2. Single Topic Course + Test or Survey.....	8
5.3. Outline with Auto Tracking.....	8
5.4. Outline with CMI or HACP Tracking	8
6. How OLM Plays Content.....	8
6.1. The OLM Player	9
6.2. Navigational Control	9
6.3. Player Options	9
7. Content Deployment Guidelines.....	9
7.1. Build the Learning Object Structure.....	9
7.2. Ensuring Accurate Tracking	10
7.3. Pre-requisites	10
7.4. Using Existing Media	11
7.5. Building HTML Content	11
7.6. Building Rich Media Content	11
7.7. Frame-Friendly Content.....	12
7.8. Screen Resolution	12
7.9. UNIX Servers and Case Sensitivity	13

7.10.	Adding CMI Tracking (JavaScript API)	13
8.	Tracking learner Progress in Content.....	13
9.1.	Auto Tracking	14
9.2.	CMI (SCORM) Tracking Overview	14
9.3.	OLM CMI Implementation.....	14
9.4.	SCORM Adapter.....	15
10.	Connectivity and hosting.....	16
11.	Access Security.....	16
12.	Packaging and Loading Content.....	17
12.1.	elearning Content Variance.....	17
12.2.	Using SCORM Packaging tools.....	17
12.3.	Installing elearning Content on hosts	18

3. Purpose and Background

The National Learning Management System (NLMS) provides the ability for NHS organisations to deliver elearning content to NHS staff in England who have a record on the National Electronic Staff Record (ESR) solution.

The NLMS builds on the existing ESR Learning Management module (which is based on the Oracle Learning Management (OLM) application) to provide elearning functionality. elearning content from both national providers (e.g. Connecting for Health, e-Learning for Healthcare and the Core Learning Unit) will be made available, as well as allowing locally produced elearning content to be delivered to staff.

This document intends to give elearning providers guidance and information to enable them to procure content, both internally through their own development teams and communicate effectively with external vendors of choice.

Related Documentation:

- Loading of course information into the NLMS / OLM course catalogue is covered in the **ESR User Guide** and the **NLMS0007 - Guide to Setting up OLM course catalogue - local elearning users**.
- Guidance for set up of the Content Hosting server can be found in the **M3960 - Content Hosting Guide**.
- Guidance on the configuration of the local network and PCs for elearning is available in the **NLMS0008 - Guide to Configuration of local networks and end user PCs to run elearning** document.
- Further guidance regarding IT infrastructure and set up for ESR as whole is to be found in the **M-0100 Trust Site IT, Printer and Network Infrastructure Readiness** document.
- A further document – **NLMS0004 - NLMS elearning Common Issues**, provides help in dealing with known problems.
- An implementation methodology for NLMS and elearning is given in the **Guide to Implementing OLM and Talent Management (including elearning)**.

All of the documents referred to above can be found on [KBase](#) (the ESR Knowledgebase system) for which you will need a user name and password (registration is via www.esrsolution.co.uk/kbase). The following path is to the NLMS page on KBase:

Functionality > Oracle Learning Management/Talent Management/National Learning Management System > National Learning Management System (elearning)

3.1. Client PC and Software specification

In order for users to be able to use OLM and Self Service they must have access to the appropriate level of PC and software resources. The minimum requirements for end user PCs are referenced in the **NLMS0008 - Guide to Configuration of local networks and end user PCs to run elearning** which is a supplement to the **ESR M-0100 Trust Site IT, Printer and Network Infrastructure Readiness** document.

Providers of elearning content will also be able to advise on PC and Software specifications required for learners to access and play their elearning offerings, if appropriate.

3.2. Content Standard

In order to ensure both a consistent approach to the design/operation/additional functionality of tracking progress and results of elearning content accessed through the NLMS, content provided will need to comply with SCORM 1.2. Providers can test SCORM compliance by accessing a free self test download from <http://www.adlnet.gov/scorm/>.

The SCORM specification was developed by the Advanced Distributed Learning (ADL) group. ADL collaborates with governments, academia, and the industry to promote effective online learning. SCORM is a set of interrelated specifications, designed to enable interoperability among content providers. SCORM combines features from other specifications such as IMS and AICC to enable the reuse of online learning content across multiple environments and products.

3.3. Accessibility Standard

Accessibility standards help ensure web based content is made accessible to as wider an audience as possible. They consider the differing requirements of people with disability as well as the varying levels of computer specifications.

Accessibility standards are constantly evolving. Against this background it is unsurprising that conflicts between differing standards and guidelines will occur. It is also clear that despite claims to the contrary, there are frequently conflicts between standards and software products, and between different software products that have to work together.

It is also important to acknowledge that generally prescribed inter/intranet standards for accessibility do not always translate well to the elearning environment, where user's interactions with their computer and the content are very different to normal web user.

Please note: NHS Connecting for Health (CfH) has developed a Standard in order to help comply with the Disability and Discrimination Act (DDA 1995) in the design, development & delivery of training materials. It should be noted however that the detailed interpretation of this act in the context of elearning is still largely unproven resulting in many different opinions and approaches. However, Project managers should make themselves aware of this Standard. For more information on these standards please go to: <http://www.connectingforhealth.nhs.uk/systemsandservices/etd/standards>

4. The Oracle Learning Object Model

elearning content deployed through OLM is based on common standard of learning objects. The Oracle learning object model is fully aligned with the ADL SCORM model for content.

The hierarchy of learning objects in OLM is described as:

Learning objects that represent the actual learning material. These objects provide a hierarchy to structure your content. This document uses the terms: -

1. "Topic" to refer to an individual section of learning that is launch-able, and
2. "Topic Group" to refer to a learning object that provides the structure or outline of the course and that is not launch-able.
3. Tests / Assessments are tracked differently than the other objects due to their scoring capabilities

This is common practice in most SCORM compliant content.

4.1. Learning Object Concepts

Learning objects have four key characteristics:

- **Granular Content**
Content is broken into many small, self-contained pieces, or objects. Each of these objects can be launched independently.
- **Metadata**
In addition to the media files that make up a course, there is data about each of the individual learning objects, and how they fit together, that is external to the media files. This metadata is read into a learning management system (LMS) and generally stored in a database.
- **Externalized Sequencing**
Sequencing between learning objects is specified in the metadata about the learning objects, not hard coded into the media files of the course. An LMS controls the launch of each individual learning object within the sequence defined in the metadata.
- **Tracking**
Learning objects are capable of communicating with an LMS about a learner's progress. For each learner, the LMS tracks a number of elements for each learning object, including status, time spent, and a score.

4.2. Topics

Topics are the lowest level of object in a course outline. A topic often corresponds to a single learning objective in a course.

A topic has media files associated with it. These can be any type of media that can be delivered to the learner's browser: a Microsoft PowerPoint¹® file, a set of HTML pages, a Microsoft Word²® document, a Real³® Audio or Video file, a Macromedia Flash⁴® movie, an Adobe Acrobat⁵® PDF document, etc. A topic may have multiple pages of content. OLM holds the information about the location of the first file to launch the topic. The media files are responsible for navigation to other pages of the topic.

4.3. Topic Groups

Topic groups are just collections of other learning objects. Generally, topic groups do not have playable media associated with them, but just serve to create the structure of an elearning course. Topic groups can be nested to any level, allowing course structures to be as detailed as necessary. Generally, a topic group will be marked completed for a learner whenever all the learning objects within the topic group are completed.

4.4. Tests

The facility to reinforce understanding and learning is provided with OLM by means of tests and assessments.

Tests are a special kind of learning object in OLM that can represent either a test that is scored or a survey for which there is no score. Rather than linking to external media, tests draw from questions stored in the OLM database. OLM provides an interface for creating banks of questions and the tests that use them. Tests can have a set sequence of questions or can draw randomized questions from a bank of reusable questions. OLM handles presentation of the test questions, scoring the test, and recording the learner's results.

Test and assessments can be imported providing they are compliant with IMS Question and Test Interoperability (QTI). This is a specification for a standard way of sharing assessment data. IMS QTI is designed to make it easier to transfer information such as questions, tests and results between different software applications.

5. Content Structure Options

There are many ways to structure content in OLM. The following are recommended practices that meet various needs.

5.1. Single Topic Course

Sometimes a course is best deployed as a single topic in OLM. This option has the advantage of being simple to set up, but provides very limited tracking information. The OLM auto-tracking capability, for non SCORM content, marks the course completed when the learners launch the single topic, and tracks the time they spend in the course.

This is the recommended option when there is existing content that cannot be broken into more granular content, such as Microsoft PowerPoint or Adobe Acrobat files.

One case of this would be off-the-shelf content that was not created as a structure of learning objects. Often this content can only be launched at the course level and handles all course navigation internally within the course media files. Older off-the-shelf content falls into this category.

A second case would be legacy content that needs to be deployed quickly. For example, a company that creates Microsoft PowerPoint presentations for classroom courses may post each of these presentations as a

¹ Microsoft PowerPoint is a registered trademark of Microsoft Corporation

² Microsoft Word is a registered trademark of Microsoft Corporation

³ Real is a registered trademark of Real Networks, Inc.

⁴ Flash is a registered trademark of Macromedia, Inc.

⁵ Acrobat is a registered trademark of Adobe Systems Inc.

single topic course on OLM. While this is certainly not an optimal learning experience, it does enable a large amount of content to be distributed very quickly.

If you are creating content from scratch, consider breaking your content into smaller pieces, and use the “Outline with Auto Tracking” option below if you do not plan to embed CMI tracking calls within the content.

5.2. Single Topic Course + Test or Survey

This option is recommended when you have any of the types of content listed in the option above, but need to have additional proof of completion or competency.

In this option, a topic group learning object would contain two child learning objects:

1. A topic for the main part of the course and
2. A test

The test can be a true assessment of the material in the course, and the test can be set up so that learners must pass the test for the course to be marked completed. Alternatively, the test can be an unscored survey indicating completion, but not necessarily demonstrating mastery.

5.3. Outline with Auto Tracking

This option is recommended for most custom content, whether developed in-house or by a professional content development vendor who chooses not to embed CMI tracking functions. Content built in this manner enables fairly detailed tracking of learner progress, yet still allows for quick deployment of existing simple media assets.

In this option, the course is broken into a series of topics, often grouped into an outline structure using topic groups. Each topic-level learning object is linked to the media files covering that material. The type of content for each topic can be different if necessary.

As each topic is launched, it is marked completed for the learner. Each higher level (topic group) in the outline is marked completed when all the learning objects (topics, tests, or topic groups) within it are completed.

Tests built in OLM can be used at the end of the course, at the end of each topic group (chapter or module), or throughout the course, to test progress.

5.4. Outline with CMI or HACP Tracking

This option is recommended if custom content is being built by a professional content development team that is familiar with SCORM tracking standards.

The content development team can create the learning object structure of the course directly in OLM or can create it externally and use SCORM metadata and packaging standards to bring the content into Oracle Learning Management.

It is highly recommended that content is tested in the OLM player throughout the development process.

There are a number of authoring tools on the market whose output contains the necessary SCORM metadata and tracking calls to successfully deploy and track within OLM.

HACP Tracking is not supported by OLM

6. How OLM Plays Content

When OLM plays content, it pulls together the content structure information from the OLM database and the media files from a content server, in the OLM player.

6.1. The OLM Player

The OLM player enables the learner to navigate and play content. There are several options for configuring the player, with the most common option having a toolbar across the top of the window, an outline frame to the left, and the actual course content played in the frame to the right.

OLM generates the toolbar and outline frames from data about the content and the learner stored in the database. Note that these frames are created and reside on the LMS server, which is a different server from where the content is stored. Each launch-able topic in the course outline has a content location stored in the database. When a learner selects a learning object in the outline, the corresponding content is loaded from a content server into the content frame on the right.

Topic groups, which assist in defining the structure of the course, should not have a content location assigned. These serve to provide structure (logical groupings of topics) to the course only. The completion status of these learning objects is calculated based on the completion status of the topics residing beneath the topic group.

The content server can be a simple web server behind a customer's firewall, or a server provided by a third party content provider with the appropriate SCORM adapters.

6.2. Navigational Control

All navigation between learning objects is handled by the OLM player, from either the outline or the left and right arrows in the toolbar. Content should not contain internal links that jump between learning objects, as this breaks the rules governing learning objects as pre-defined objects that can be launched independently from one another. Coding links into the content to switch between learning objects will confuse the OLM player and the content will not be tracked correctly. Content may have multiple pages within a learning object and may navigate across those pages in any manner as defined in the navigational controls within the content files.

6.3. Player Options

Each offering in OLM can have different player options set by an administrator, depending on the nature of the course. The toolbar and outline pane can be enabled or disabled. Consideration should be given to users when deciding to remove the outline pane as this is the main method of navigation between learning objects. Links for navigating between learning objects in the toolbar can be enabled or disabled as well.

Another option enables the player to launch in its own window or remain in the same window as the rest of the OLM learner interface. When the option "Open in New Window" is selected, this launches the OLM player in a separate browser window, but still maintains all the features of the player such as the toolbar at the top of the screen and the outline frame. This option does not open only the content files in a new window, it also includes the OLM player in order to track progress successfully.

In the interests of consistency there may potentially be a set of standards applied to the configuration of the player.

7. Content Deployment Guidelines

7.1. Build the Learning Object Structure

The content development team can create the learning object structure of the course directly in the OLM administration interface.

The learning object structure can be built in OLM in parallel to creating the actual media files. After the outline structure is built, media files can be linked in as they are completed.

Alternatively, content developers can create the structure externally and use SCORM metadata and packaging standards to bring the objects into OLM. This is only recommended if the team is quite familiar with these standards, and already has a suite of tools in place to support building SCORM compliant content.

For further information related to deploying and implementing SCORM compliant content, please send a request email to the point of contact in 'Further Information' section. Alternatively, if you have access, these can be located on Oracle's metalink website: <https://metalink.oracle.com/>

7.2. Ensuring Accurate Tracking

The key to accurate course tracking within OLM is to ensure that any learning object that contains child learning objects does not have a content location assigned. In the example below, the top level learning object “Content Assembly in Oracle Learning Management” has two child learning objects:

- Content Assembly in Oracle Learning Management
 - Proper Content Assembly Techniques
 - Post Test

The completion status and time spent for the top-level learning object is calculated by aggregating the completion status and time spent from the child learning objects. Once both child learning objects are completed, the top-level learning object is marked complete and reflects the total time spent in the course. These rules apply to more complex course structures, where there can be one or multiple topic groups structured within a course, as shown in the example below:

- Inland Rules of the Road (HTML Format)
- Conduct of Vessels in any Condition of Visibility
- Sailing Rules & Steering
- Conduct of Vessels in any Condition of Visibility
- Conduct of Vessels in Sight of One Another
- Conduct of Vessels in Restricted Visibility
- Lights & Shapes
- Inland Maritime Navigation Signals
- Inland Rules of the Road Examination

The overall completion status of the “Inland Rules of the Road (HTML Format)” learning object is calculated based on the completion status of the learning objects beneath it. The completion status of the “Sailing Rules&Steering” topic group learning object is calculated based on the completion status of the three topics beneath it.

7.3. Pre-requisites

Although a sequence may be implied by the outline structure of a course, by default learners can access the individual topics of a course in any order. If you want to force learners to progress through the content in a particular sequence, player prerequisite relationships can be created between learning objects. Learners are not able to start a later learning object until they have completed the earlier objects that are set as prerequisites.

Note: There is currently one limitation when assigning prerequisites to learning objects:

Prerequisites that are set on a topic group learning object do not filter down to the child learning objects in that group, which prevents the prerequisite functionality from working as you might expect. For example, let us assume you have a course with two topic groups. Each topic group contains two topics:

- Course with Prerequisites
 - Topic Group 1
 - Topic A
 - Topic B
 - Topic Group 2

- Topic C
- Topic D

In this example, you would like learners to complete all the topics in Topic Group 1 prior to beginning the topics in Topic Group 2. A natural inclination would be to set Topic Group 1 as a prerequisite to Topic Group 2. This however, does not work because Topic Group 2 is not a launch-able object, and the prerequisite relationship does not flow down to the topics within Topic Group 2.

In order to do this effectively in OLM, set prerequisite relationships on Topic C and Topic D, so that the Topic Group 1 learning object is the prerequisite. This prevents a learner from launching Topic C or Topic D until Topic Group 1 is considered complete. As described earlier in the document, Topic Group 1 is considered complete when Topic A and Topic B have been completed.

7.4. Using Existing Media

Most companies have many existing resources that can be re-purposed as e- Learning content. A learning object can point to files including a Microsoft PowerPoint file, a set of HTML pages, a Microsoft Word document, a PDF document, etc. A course can easily be constructed as a series of these kinds of resources.

While this content is not interactive and does not provide for the richest learner experience, leveraging these existing assets does allow a large amount of content to be deployed quickly and at low cost. While the quality is not the best, because the investment is so low, the ROI for this type of elearning can be quite high. Be aware that Microsoft Internet Explorer® (MSIE) often handles these kinds of documents differently from other browsers. The default in MSIE is to try to open these documents inline in the browser, whereas other browsers, by default, often try to download them to the user's computer. Also, the Adobe Acrobat plug-in often has difficulty displaying PDF files over 1MB in the browser. If you distribute PDF files as part of a course for viewing online, keep the size fairly small.

7.5. Building HTML Content

Since HTML is the native language of the web, HTML-based content is a natural fit for elearning.

A learning object may have multiple pages of HTML content. OLM holds the location of the first HTML file that is launched for the object. The HTML files are responsible for navigation to other pages of the object.

As stated in the "Navigational Control" section above, all navigation between learning objects is handled by the OLM player; either from the outline, or the left and right arrows in the toolbar. Content should not contain internal links that jump between learning objects, as this breaks the rules governing learning objects as pre-defined objects that can be launched independently from one another. Coding links into the content to switch between learning objects will confuse the OLM player and the content will not be tracked correctly.

7.6. Building Rich Media Content

OLM can play rich media content created in any tool that produces content that can be delivered to a browser.

OLM plays content requiring plug-ins just like any other browser-deliverable content. So, while plug-in based content is fine to use, OLM does not provide any special mechanism to detect or ensure that the plug-in is installed on the learner's browser. The content should have a mechanism to detect the plug-in if possible and should always provide a link for the learner to download and install the plug-in.

Most plug-in based formats require a MIME type to be configured on the content server from which the content is delivered.

Streaming media often has additional server-side requirements. For true streaming, a server-side component from the streaming technology vendor is required to manage the streaming process. This component must be installed on the customer or third-party content server.

Be aware that many forms of streaming content can be played as if it were static content, without providing true streaming and requiring a server-side component.

The bandwidth requirements of the media, and connectivity to and within the customer site, will determine whether this is an acceptable option.

Note: Local desktop build policy will determine the use of content requiring plug-ins.

7.7. Frame-Friendly Content

To play correctly in the OLM player, content must be frame-friendly.” The content may contain browser frames of its own and may manage those frames how it wishes, however, it must affect only those frames that it creates.

The safest way to accomplish this is for the content to create an explicit name for each frame. The content can also control its frames by using the parent/child frame hierarchy in the browser, but must not affect frames created by OLM above the content. In particular, the content should never make reference to the window.top JavaScript object or use `_top` as the target for a URL.

This is especially important for SCORM compliant content that employs CMI tracking calls where the content resides on a separate server from the LMS. All NLMS elearning content will reside on a separate server from the NLMS Therefore these types of calls will fail because the content resides on a different domain from the LMS hence the use of SCORM Adapters to overcome this.

Current browsers have strict security rules embedded that prevent JavaScript functions from calling operations across domains. This security extends beyond the domain name (such as “.oracle.com”). The host and domain values must be the same (and in some cases the port number if applicable) in order to be considered in the same domain. For example, if the LMS is running on “learning.mycompany.com” and the content resides on “learningcontent.mycompany.com”, the browser considers these to be different domains and does not allow cross-domain functions to execute.

If cross-domain errors occur, you can troubleshoot the offending code by looking for error messages produced in the browser.

- In MSIE, you see a warning icon in the bottom left hand corner of your browser window. Double-click on this icon to view the relevant JavaScript error that occurred. Typically, the JavaScript error will say “Permission Denied”, which indicates a cross-domain scripting error, along with the line number in the HTML document where the error occurred. In many cases, this will assist you in determining which line of code is attempting to cross domains. Note that MSIE only references the html file where the error occurred. If the lines of code causing the problem occur in another file that is referenced within the html page, this warning message does not identify the referenced file.
- In Netscape® based browsers, troubleshooting is somewhat easier. The JavaScript console references the exact file in which the offending code resides. Typically, you will see a JavaScript error message in the status bar at the bottom of the browser window. In some cases, you may not see this error but the content may not work correctly. Type “JavaScript:” into the Location bar of the browser to bring up the JavaScript console. This typically references the offending line of code in the file (including referenced JavaScript files) where the code is executed.

Content that is not frame-friendly may not work in these instances if JavaScript functions attempt to cross domains to perform certain operations.

7.8. Screen Resolution

Most users in the NHS have a screen resolution of no less than 1024x768 pixels. The size of content should be sympathetic to this as if content is too large, then users will have to use scroll bars to use content effectively.

The actual screen real-estate available for users to see will depend on the structure of the content. In OLM, there are two bars that are displayed – the ‘Outline’ and the ‘Toolbar’. The toolbar is situated at the top of the screen, and is approximately 45 pixels tall. The outline is situated to the left of the screen and is approximately 210 pixels wide.

The toolbar is mandatory and must be displayed on all content as this is the only method the user has to correctly exit content. The outline however is not mandatory and can be removed for content that comprises of a single module. The outline is used to navigate to different learning objects in the course – without it, the user would not be able to move to the next module.

In conclusion, the available space for use in OLM is:

With outline: 810(w) x 554(h)

Without outline: 1020(w) x 554(h)

Screenshots can be found in the document below:



7.9. UNIX Servers and Case Sensitivity

All URLs for UNIX web servers are case sensitive, but URLs for Microsoft Windows® web servers are not. Frequently, content is developed in a Windows environment and may not be tested on UNIX servers prior to its release. Because of this, case discrepancies in file names, directory names, and URLs often go unnoticed, even for professionally produced content! If content will be deployed on a UNIX server, it is very important that it be thoroughly tested in a UNIX environment before deployment.

7.10. Adding CMI Tracking (JavaScript API)

This is recommended only if custom content is being built by a professional content development team or in-house resources that are familiar with SCORM tracking standards. Details of CMI (Computer Managed Instruction) tracking are covered elsewhere in this document.

8. Tracking learner Progress in Content

The tracking of a learner's progress is directly tied to the learning objects within a course. When a learner launches an online course for the first time, the learner is taken to the first learning object within the course structure that has an associated starting URL. At this point, an attempt record for the learner is created in the database. This attempt record stores information such as the time spent, completion status, and score for that learning object. When the learner leaves the learning object to return to the home page, or to navigate to another learning object, the attempt record is finished and a performance record for the learning object is created. It is the data stored in the performance record that learners see in the learner interface, to indicate overall progress for the associated learning object. Performance records are also created at this time for parent learning objects that assist in defining and maintaining the course structure. However, because these learning objects are not launch-able, they do not have any attempt records.

Every time a learner launches a learning object, a new attempt record is created, regardless of the learner's completion status. Each learner may have many attempt records for the same learning object, but only one performance record. The performance is updated to reflect learner progress up to the point where an attempt is marked as completed or passed. At that point, the performance record completion status is updated with the latest status (and optionally a score) and cannot be modified by subsequent attempts on a learning object. The time spent on the performance record continues to be updated however, based on subsequent learner attempts on the learning object.

Completion status can occur at different levels depending on the course structure. This can range from a basic level of when a user launches and closes a single object, to the completion of many objects, i.e. a modular course.

OLM has three mechanisms for tracking a learner's progress through content: Auto tracking, CMI (SCORM) tracking. With each mechanism, a learning object can track, for each learner, a completion status (not attempted, incomplete, completed, passed, failed), the time spent in the content, and possibly a score.

OLM supports the CMI tracking standard in the SCORM specification from ADL.

9.1. Auto Tracking

OLM can track the launch of, and time spent in, any content automatically. When set to auto tracking, each topic is marked completed as soon as it is launched. If a course has multiple levels of structure, the parent learning object will be marked completed when all of its children are marked completed. This may be useful for playing and reporting on content that is not SCORM compliant.

Tests created in OLM also record the score for the test automatically.

9.2. CMI (SCORM) Tracking Overview

For more detailed tracking, OLM supports the CMI tracking model in the SCORM specification published by the ADL. (See <http://www.adlnet.gov/scorm/> for more information). SCORM CMI tracking uses JavaScript calls for the content to communicate with the LMS.

Please refer to the next section for Oracle's implementation of the CMI specification.

The CMI API allows statuses and scores to be set against each content object. The SCORM specification uses the term "SCO" (Shareable Content Object), which is equivalent to a topic-level learning object in Oracle Learning Management (OLM). The CMI API also allows the content to query OLM for data about the learners and their past interactions with the content.

As specified by the ADL for SCORM CMI tracking, the content must contain code to find the API, initialize the API, get and set values as appropriate, and finalize transaction upon exit. The specific technology used by the content to implement its tracking logic is not important, as long as it can make the correct calls via JavaScript to meet the SCORM specification.

9.3. OLM CMI Implementation

This section assumes some familiarity with details of the CMI API available in the SCORM Run-Time Environment specification at:

http://www.adlnet.org/library/documents/scorm/specifications/SCORM_1.2_RunTimeEnv.pdf .

OLM supports all mandatory SCORM function calls. The SCORM specification has an extensive list of data elements, some mandatory and many optional. Most LMS systems support some subset of the total. In the ADL terminology, Oracle Learning Management is "LMS RTE 2". This means that OLM supports the mandatory data elements of the SCORM specification. The data elements supported in OLM are listed in the following two tables (extracted from the AICC CMI Specification documentation).

9.3.1. SCORM Calls compatible with OLM

Core (exit, score, credit, lesson_location ,session_time ,total_time, lesson_mode, student_id, student_name (will only return initial), lesson_status, entry)
suspend_data
launch_data
objectives (Must be given an id)

9.3.2. SCORM Calls not compatible with OLM

comments
comments_from_lms
student_data (mastery_score, max_time_allowed, time_limit_action)
student_preferences (audio, language, speed, text)

Core Element Name	Definition
student_id	The username of the user.
student_name	The student's name. However, in OLM this will only return the student's initials (for security purposes) and should not be used.
lesson_location	This corresponds to the lesson's exit point passed to the CMI system the last time the learner experienced the lesson.
Credit	Indicates whether the content gives credit for completion.
lesson_status	The status of the lesson – i.e. Incomplete, Completed.
Entry	Indication of whether the student has been in the lesson before.
Exit	Indication of how or why the student left the session
score.raw	Numerical representation of student performance in lesson. May be unprocessed raw score.
Total_time	Accumulated time of all the student sessions in the lesson.
Lesson_mode	Identification of student related information that may be used to change the behaviour of the lesson.
Suspend_Data	Unique information generated by the lesson during previous uses that is needed for the current use.
launch_data	Unique information generated at the lesson's creation that is needed for every use.
session_time	Time spent in the lesson during the session that is ending.

Note that the CMI terms “lesson” and “AU” (assignable unit) refer to a learning object in OLM.

Objective Element Name	Definition
Id	A unique identifier for each objective. Note: if you are using objectives in your content, each objective MUST have an id, otherwise the communication may fail.
Score	Indication of the score obtained by the student after each attempt to master an objective.
Raw	This may be an unprocessed or processed indicator of how the student performed with the AU's interactions (related to the objective) experienced.
Max	This is the largest score the student could have with the AU's interactions (related to the objective) experienced.
Min	This is the smallest score that the student could have achieved with the AU's interactions (related to the objective) experienced.
Status	Indication of the status of an objective.

9.4. SCORM Adapter

The JavaScript security model does not allow scripts in frames coming from different Internet domains to pass information back and forth. Because of this security constraint, CMI calls between an LMS system and content hosted on a separate server will normally fail.

To overcome this limitation, Oracle has developed the SCORM Adapter. The adapter is a set of files that are stored on the remote server with the content files.

The files are static HTML. There is nothing to execute on the content server machine and no special permissions or configuration is necessary. The files need to be installed only once on the content server and then registered in the OLM administrative user interface.

When SCORM compliant content is launched, OLM scans its list of registered SCORM adapters. If the content comes from a location with a registered SCORM adapter, the URL is automatically re-written to redirect the learner's browser to the SCORM adapter files, which in turn load the content files.

The content sees a SCORM API provided on the frameset of the SCORM adapter, coming from the same server as the content, eliminating any security problems. As noted in the section above, the content must search for the SCORM API from the bottom up in order to find the API correctly.

To enable the connectivity the Oracle SCORM Adapter folder will also need to be loaded on the content server. **The OLM SCORM adapters are accessible via the McKesson Customer Portal or you may raise an SR on the McKesson Remedy system requesting them to be sent to you.**

In addition to this each elearning item will need to have the following information:

- Name Of Server
- Protocol
- Host
- Port
- Physical Directory
- Location URL of SCORM adapter
- URL Prefix

These details will be registered within ESR to ensure connectivity and enable the importing of setup details from the content server.

10. Connectivity and hosting

It is a requirement of the NLMS that elearning content will be hosted by the content provider or a contracted third party on an N3 facing server as the ESR NLMS does not include scope for storing elearning content.

- If organisations require remote user access set up they will need to ensure their content is held on internet facing servers (not N3 facing servers).
- If elearning content is not for public access, it will either need to be hosted on the N3 environment (and therefore not available remotely) or the url will need to be secure and the content not discoverable.
- Local content providers will be required to provide a single URL to each of their elearning courses.

For detailed information on content hosting please refer to the **M-3960 – Content Hosting Guide**, available on KBase.

11. Access Security

It is paramount to the success of the NLMS that appropriate security access is set up and administered correctly. Users will be required to hold personalised logons for both access paths to ensure only authorised users may access elearning.

In respect to users viewing appropriate elearning content OLM has the flexibility to restrict access to content by Organisation, Jobs, Positions, Specific learners or Learner Work Area. In addition, a user can be restricted from accessing a course by not completing "Mandatory" course or competency prerequisites.

12. Packaging and Loading Content

12.1. elearning Content Variance

Over the years across the NHS numerous learning solutions have been adopted. These vary from, for example:

- Standalone content (CDs running locally or with cookies enabled to track learning progress);
- Static websites with learner content (non-interactive);
- Interactive sites that produce reports on progress;
- Flash based sites;
- Adapted content for learning, for example: PowerPoint or Adobe Acrobat PDF navigated content;
- Learning management systems capable of registration, course allocation, book marking and both score and usage tracking.

In the case of Learning Management Systems, content is packaged or 'wrapped' using elearning standards, for example: SCORM 1.2. Such content can vary in terms of type, for example: Image, Audio, Video, Flash and Text. To assemble elearning content of this type, content packing tools are required.

It is important to consider packaging when designing a course, the following design concepts may influence packaging:

- Web Browser capability – to ensure compatibility of content behaviour;
- Javascript – to capture and pass on required information, for example: tracking, scores and completion statuses;
- Page framing – to assure original navigation guide-lines.

When assembling content for an elearning course it is recommended that attention is given to keywords and sources of information; for example, understanding key references and accreditation for content originators.

12.2. Using SCORM Packaging tools

In order to make an elearning course compatible (playable) by the NLMS / OLM, learning assets (for example, video, audio, text, images and exercises), also known as SCO⁶s (Shared Content Objects) must be wrapped or collated using the SCORM 1.2 standard. The term 'packing' is often used to 'e-enable' a course.

A number of commercial and open source packaging tools are available. Before selecting a packing tool it is recommended that consideration is given to licensing issues and permitted elearning course user limitations. The main reason for performing the 'packing' operation is to ensure that the content becomes SCORM compliant.

The following diagram shows how content is packaged to form a working SCORM learning site:

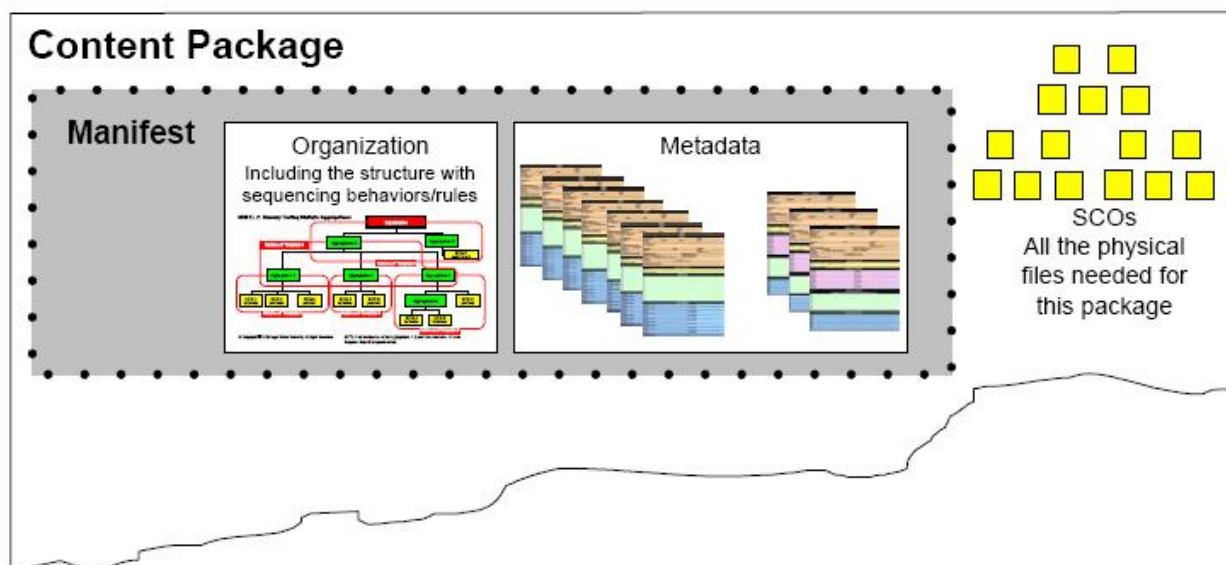
There are three main elements of the course in its native form (please see the grey box in the illustration below):

1. Course structure (how the course is designed in terms of flow and expected learner behaviour, for example: Part 1 is the Course Introduction, Part 2 is Key concepts etc.).
2. Metadata (data about data) – a list of all the files and content items that the course will use, for example: Image.jpg, a description of each course item, the author.
3. Learning Objects (also referred to as Shared Content Objects or SCOs) i.e. the actual content files.

All three must be packaged together using a SCORM packing tool.

⁶ The term SCO has a broader definition than the term learning object., i.e.: any content used or referenced during technology supported learning. Examples include instructional content, instructional software and persons, organisations, or events referenced during technology supported learning. It can be more than the content item itself as it has the flexibility to describe related events or activities.

The result is a set of files that once loaded onto a content server can be run as a SCORM 1.2 compliant course, with the ability to communicate key course behaviours, for example: test scores and tracking.



One of the files generated by this process is known as the IMS Manifest file. The IMS Manifest file (XML) can be considered as the binding or index element of the entire set of files. This file contains metadata (data or information about data). In other Learning Management Systems this file is used to populate course data and objectives. With the NLMS / OLM, however, this must be done manually using the Learning Administration URP, (see below).

To enable SCORM 1.2 compliance, all content is usually zipped by the packaging process.

12.3. Installing elearning Content on hosts

In the case of NLMS / OLM, the content must not be in zipped format. When the course is placed on a Content Server the zipped file will need to be unzipped with its files loaded into a separate folder that can be referenced with a unique starting URL. Alternatively, before loading the files onto the server do not zip the content as part of the packing process.

All course content must be loaded at the same level or below that of the SCORM adapters in the hierarchy on the content server as in the example below:

http://www.ourserver.com/ourcourses/Oracle_SCORM_Adapter/Oracle_SCORM_Adapter.html
<http://www.ourserver.com/ourcourses/course1/index.html>
<http://www.ourserver.com/ourcourses/course2/index.html>
<http://www.ourserver.com/ourcourses/course3/index.html>

For further explanation regarding the SCORM adapter see Section 5.3 of the **M-3960 - Content Hosting Guide**, available on KBase.

Courses must also be registered within the OLM course catalogue. It is not possible to import metadata at present from the manifest file into OLM, therefore the courses must be set up manually. For full instructions on how to do this refer to the **NLMS0007 - Guide to Setting up OLM course catalogue - local elearning users**.