Australian Flexible Learning Framework

Supporting Flexible Learning Opportunities

Potential Benefits and Challenges in the use of SCORM in VET

VET Learning Object Repository Project

flexiblelearning.net.au

An initiative within the Australian Flexible Learning Framework for the National Vocational Education and Training System 2000-2004

Managed by the Flexible Learning Advisory Group on behalf of the Commonwealth, all States and Territories in Conjunction with ANTA

Contents

Introduction

Much conjecture, discussion and interest has stemmed from Advanced Distributed Learning's (ADL) Shareable Content Object Reference Model (SCORM). Ignoring early scepticism and reluctance from educational circles, SCORM has been a model that governments; content and learning management systems (LMS) vendors and the learning industry around the world have voluntarily adopted.

There is no ignoring the fact that SCORM offers attractive solutions to addressing what Mark Oehlert, Advanced Distributed Learning's (ADL) communications officer, calls "the ability to reuse, locate and provide for interoperability of learning content to learning management systems (LMSs)" (Kraan 2002). On face value SCORM promises greater efficiency for learning objects, a prospect that has excited many in Government and Industry, including the Australian VET sector. However, voices from the e-Learning arena express concern that the adoption of SCORM may lead to inappropriate design and use of learning objects (Wiley 2000, 2003, Koper 2003).

This paper aims to open up discussion of key issues, challenges and implications for instructional design and instructive techniques faced by creators of VET learning/training material who adopt SCORM. It asks: what is at stake when learning content is, for want of a better word, SCORMified, either by designing SCORM conformant[1] material from existing instructional resources, or by designing new resources that conform to SCORM specifications.

An overview of SCORM

The Sharable Courseware Object Reference Model (SCORM), also more recently referred to as the Shareable Content Object Reference Model Initiative, was released by the Advanced Distributed Learning Network (ADL) in January 2000.

SCORM prescribes the development, packaging and delivery of education and training materials whenever and wherever they are needed. SCORM-compliant courses leverage course development investments by ensuring that compliant courses are "RAID":

- **Re-usable:** easily modified and used by different development tools
- **Accessible:** can be searched and made available as needed by both learners and content developers
- **Interoperable:** operates across a wide variety of hardware, operating systems and web browsers
- **Durable:** does not require significant modifications with new versions of system software.

SCORM is not a standard in itself, but rather a reference model that serves to test the effectiveness and real-life application of a collection of individual specifications and standards to create a "unified content model" to enable the re-use of learning materials across a range of products and platforms (Advanced Distributed Learning 2003). It includes:

- specifications relating to the run-time environment
- specifications for representing course structures or content aggregation

[1] Oehlert (2002) asserts that, as SCORM is a specification, not a standard, there is no such thing as SCORM "compliance" but rather the appropriate term is "conformance", that the product conforms to the rules within SCORM.

- specifications for creating metadata records for courses, content, and raw media elements.

A more detailed discussion of the SCORM specifications is available in the VET Learning Objects Repository Project's "Introduction to Standards and Specifications for Learning Objects and Repositories".

SCORM Runtime Environment

The Runtime Environment specifies how the content should behave once it has been launched by the LMS. A SCORM compliant LMS is required to implement an API (Application Program Interface) consisting of a range of functions that content may access to communicate with the LMS (Advanced Distributed Learning 2003).

This API is implemented by what the SCORM calls an API Adapter. The API Adapter must reside in a window that is a parent window or a parent frame of the window that contains the content. This means that the LMS may launch the content either in a new window or in a frameset. The API Adapter must be an ECMAScript (JavaScript) object named "API".

All communication between the SCO and the LMS is handled by the runtime environment API. The standard API consists of eight functions as described in the table below:

Function	Description
[LMSInitialize()]	This must be called before any other function and alerts the LMS that a content component is going to communicate with the LMS.
[LMSFinish()]	If the [LMSInitialize()] function has been called then this function must be used to terminate communication with the LMS. No other function can be called after [LMSFinish()] except LMSGetLastError.
[LMSGetValue()]	Returns a value from the LMS (eg 'student_ID' or 'student_name')
[LMSSetValue()]	Sends a value to the LMS (eg 'student_response' or 'student_score')
[LMSCommit()]	Tells the LMS to commit new values.
[LMSGetLastError()]	Returns the error code should an API call be unsuccessful
[LMSGetErrorString()]	Returns a textual description of the error represented by the error code number
[LMSGetDiagnostic()]	This function enables vendor-specific error descriptions to be developed and accessed by the lesson

Of these eight only the Initialise [LMSInitialize()] and the Finish [LMSFinish()] function are compulsory to communicate with the LMS. The other six functions are optional but are recommended to allow the transfer of variables between the content and the LMS.

Implementing this API in the LMS is a little more involved; it has to implement all of the API functions and support most of the SCORM data model. The tricky issue involved with implementing a SCORM compliant LMS is how to handle the browser-to-server communication. Most people choose to do this with a Java applet, but others have been successful using Flash, ActiveX controls and pure JavaScript (Rustici Software 2003).

SCORM metadata specification

The SCORM Metadata Application Profile directly references the IEEE Learning Object Metadata (LOM) standard and the IEEE Draft Standard for Extensible Markup Language (XML) Binding for Learning Object Metadata Data Model. The metadata specification defines a very rich data model of approximately 64 metadata elements: however, only a small subset of the data elements is required to achieve SCORM compliance.

While the SCORM fully adheres to the IEEE standard, ADL strongly recommends the use of the IEEE LOM for describing the SCORM content model components. However, other metadata schemas may be used. These metadata schemas may or may not be recognised by systems (MASIE Center 2003).

SCORM Content Packaging

SCORM uses the IMS Content Packaging Specification developed by the (IMS) Global Learning Consortium. The content packaging specification defines how training should be packaged digitally to facilitate sharing learning resources.

Basically, the content packaging model dictates that the data from the content model should be stored in an XML file named imsmanifest.xml at the root of your package. Metadata elements should be stored in arbitrarily named XML files (referenced by imsmanifest.xml). Content packaging also describes how to package data physically onto disk and how to compress the data into what is called a PIF file (basically a zip) (Rustici Software 2003).

SCORM compliant learning management systems are able to interpret the instructions in the manifest file, unpack the package and present the resources to the user.

Background: The SCORM Debate

The SCORM debate tends to be polarised in two vehemently opposed camps with little middle ground at present. This debate is an extension of the learning objects debate that can be traced to Wiley (2000). SCORM claims to be “pedagogically neutral”, a description which has since been associated with Wiley’s criticism of a learning object approach to learning product design:

> “Software vendors and standards bodies describe their learning object related work as being ‘instructional theory neutral’. Were this the case, all would be well in learning object land. Problematically, a more accurate description of their products is ‘instructional theory agnostic,’ or in other words, ‘we don’t know if you’re employing an instructional theory or not, and we don’t care’.”

Much like traditional conceptions of learning objects, SCORM specifies a basic-level aggregate, the SCO, as a representation of a single unit of re-usable learning content. In this regard, the same concerns about instructional value being lost when the learning structure is broken into objects also apply to SCOs. In referring to SCORM 1.2, for instance, Click2Learn’s Claude Ostyn notes that “[SCORM 1.2] is known to be deficient in the ability to capture pedagogical logic between learning objects.” (Welsch 2002.)

More recent critics of SCORM have expressed similar concerns about this claim. Dan Rehak[2] cited in Kraan and Wilson (2002), emphasises that SCORM is not the right approach for higher and primary education, that it is "essentially about a single-learner, self-paced and self-directed. It has a limited pedagogical model unsuited for some environments". Rehak further comments that SCORM is used "mainly in the area of training for specific systems and situations by people who are not generally in full time education."

Implied in these assertions is a belief that SCORM, far from being "pedagogically neutral," encourages a pedagogy that is behaviourist, didactic and instructive. What room is there, then, in SCORM for learning designs that adopt more constructive, problem or activity-based learning strategies?

In response, Oehlert offers alternative methods of thinking about SCORM. SCORM, as a set of specifications, is evolving "to ensure that it is as useful as possible to the ADL community". It is also a "minimum set of standards, not the max. It describes a way in which content from one system can interoperate with another system. It says nothing about the 'scope' or 'flexibility' of content once over that minimum line". (Oehlert 2002.) Oehlert goes on to say that SCORM is a tool, much like HTML which, while governed by internal standards, has allowed "for an incredibly diverse array of Web sites." He also points out that, "The Net started out to serve Department of Defence researchers. The Web started out to serve physics researchers. They have both grown wildly beyond those domains. ADL/SCORM began life within the learning/training/performance support arenas but I do not see it as being learning specific." Following Oehlert's logic, we should then be able to create a "diverse array" of teaching and learning resources that are SCORM conformant.

Contrary to Rehak's concerns, some instructional designers and content developers have shown that they can create learning material that is collaborative, interactive, engaging ***and*** SCORM conformant. In "Supporting Collaborative Learning Activities with SCORM," Ip and Canale (2003) demonstrate that a SCO "based on a collaborative learning activity (an activity which involves more than one learner... communicating with each other either as peer-to-peer or within assumed roles) can still be [packaged and used] as a learning object." Ip and Canale describe the creation of SCOs that make use of collaborative tools hosted by third-party servers to engender collaborative/discursive activities within the SCO. In one example, learners could read a poem and then, using a third-party annotation tool, add their thoughts about the poem for other learners to read. There are technical limitations to this approach, however. In this case, the collaborative learning activity (object) is re-usable as long as the required collaborative tool is available and correctly configured for each instantiation of the SCO.

The question of context

Application of SCORM conformance to the Australian VET sector needs to consider the issue of context, particularly in its dual signification (Anderson and Mah 2002):

- "Learning context" – the role of the object within the framework of the lesson; users may need to be oriented to this context via a lesson overview and conclusion.

[2] Described as one the "chief architects" of SCORM.

- "Workplace context" – the context within which training occurs and the value of the object in representing educational elements related to the workplace environment.

The *VET Learning Object Repository – Green Paper for Discussion* notes that "arguments for context neutral learning objects do not appear to be 'holding water' in the educational sector, with a number of authors and organisations maintaining that part of the uniqueness of [the VET sector] is the provision of workplace examples and scenarios." (2003) Much has been written about the importance of context in learning, and not only in the VET sector (Higgs, Meredith & Hand 2003, Wiley 2003, Anderson and Mah 2002). It is not within the scope of this paper to rehash or extend on this discussion. However, the importance of workplace/industry contexts will affect the way SCORM is adopted (and adapted) in the development of learning objects for the VET sector.

In complying with the competency-based, workplace-assessed requirements of AQTF Training Packages, Australian Flexible Learning (Framework) Framework Toolboxes[3] use contextualised, authentic tasks to frame learning activities, which are supported by information represented in the form of authentic workplace resources (such as workplace manuals, supervisorial characters, colleagues and magazines). Learners work towards completion of these tasks; success depends on the learner accessing a range of resources, often in non-linear fashion, located within the Toolbox and from the learner's training environment, workplace or community, to complete the tasks. Completion often requires participation in a workplace-based task to demonstrate competence. Given Oehlert's assertions cited above, learning designs such as those figured in Toolboxes, should be deliverable within a SCORM conformant framework. The delivery style, however, will be determined by the SCORM aggregation model.

The SCORM aggregation model and instructional design

In brief, the SCORM aggregation model defines content across three levels: assets, SCOs and aggregations. The specification provides "rules" that need to be observed in structuring content across these levels to allow for interoperability. Chief of these is "The Golden Rule", which states that a SCO, representing the smallest unit of potentially re-usable tracking-enabled learning content, cannot call or access another SCO without going through the LMS.

In traditional e-Learning course design, branching in the learning path allows learners to seamlessly move between content areas such as modules, units, lessons, topics and so on. Context related material such as overviews, summaries and conclusions could be used to mediate between content areas. The Golden Rule, however, prevents the free-flowing navigation between objects. If a SCO is to be potentially re-usable, then it needs to be decoupled from contextual material and cannot be dependent on, or affect, content from other SCOs; essentially this requires each SCO to be completely self-contained. It was this that gave rise to Ostyn's initial concerns about the absent pedagogic logic between SCOs.

SCORM 1.3, however, includes IMS Simple Sequencing, a specification that allows for inter-SCO navigation that is managed by the LMS. This is not too different from current LMS-managed learning material, where navigation between learning content is enabled by the LMS interface and where users essentially navigate between

[3] http://flexiblelearning.net.au/toolbox

"chunks" or "aggregates" of content via the LMS interface. IMS Simple Sequencing allows for a more flexible sequencing of SCOs, including those which are adaptive to specific outcomes. For example, a content aggregate containing two SCOs can be set up so that the LMS makes the second SCO available only if the learner passes a test embedded in the first SCO. The Carnegie Mellon *SCORM Best Practices Guide for Content Developers* happily declares that this provides designers with the "ability to 'equate' SCORM structures to the traditional instructional design components [they] are accustomed to [lets them] essentially create limitless structures of [their] own." But does this cater for the competency demonstration requirement of the VET sector?

The challenge for instructional designers is to conceive of ways in which sound learning strategies can be deployed within this framework and its rules. There are issues that SCORM 1.3 has yet to address, notably the position of contextual material; the Carnegie Mellon *SCORM Best Practices Guide for Content Developers* suggests that contextual material such as an introduction or overview should be created as individual SCOs. This may be a real issue for educators who regard context material as tightly coupled with learning content.

The role of Learning Management Systems

A learning management system (LMS) essentially is a tool capable of managing a range of organisational learning activities. These activities could vary from instructor-led classroom-based training to educational seminars or web-based online training. From an end-user point of view, an LMS provides a learning environment supporting a range of learning content with integrated access to various self-assessment and communication tools. Administratively, an LMS makes it easy to enter, track, manage, and report on learning activities and competencies within the organisation.

Commercial LMSs such as WebCT and Blackboard are already in common use in the VET sector. The types of courseware supported by these LMSs is wide and varied, but typically includes a number of HTML pages and supporting media that are linked together in such a way as to meet a particular learning outcome. This is what is known as System Directed Learning.

SCORM assumes System Directed Learning. It is intended for, and requires, implementation and management within an LMS. As such individual SCOs are independent of each other and only communicate via the learning management system (LMS). The SCOs themselves do not contain internal navigation systems and rely on the LMS to control their presentation and sequencing to the learner (Rae & Taylor 2002). An SCO without an LMS is simply a collection of loosely related digital resources.

Possibly one of the most important features of a SCORM compliant LMS is the SCORM API (Application Program Interface). The API is a set of eight base functions embedded within the LMS that interacts with the SCORM run-time component that is embedded within the SCORM content aggregation model.

Newman notes, "That in a perfect SCORM world, every piece of courseware works with every LMS" (Newman 2002). However, in a series of practical interoperability trials conducted by the Centre for Educational Technology Interoperability Standards (CETIS) in 2002, it was revealed that this was not the case, with only two out of the six tools tested performing satisfactorily (Wilson 2002). As the Arizona Learning Technology Partnership (ALTP) notes, "LMS vendors say they are compliant but they

are not. The trouble is in the lack of agreement on what compliance means". (ALTP 2001).

This potentially adds a further complicating feature to the design and development of SCORM compliant learning resources. Not only does the developer need to follow the SCORM specifications they also need to develop the learning materials to allow for the peculiarities of the LMS.

Case Study: Designing SCOs from existing Toolbox material

Following guidelines provided in the Carnegie Mellon University's *SCORM Best Practices Guide for Content Developers*, WestOne Services trialled the conversion of one unit of competency from an existing Series 3 Australian Flexible Learning Framework (Framework) Toolbox into Shareable Content Objects (SCOs) to seek out and document the practical hurdles presented to a design and development team when attempting to repurpose and package existing content for SCORM compliance.

This particular toolbox was chosen as its design fell mid range on the continuum between a closed, content presentation model and a completely open-ended problem solving approach. Whilst the toolbox utilised contextualised resources located outside of the activity it also presented the activity itself in a sequential order. Resources were shared between activities in the same and other units of competency. The original unit trialled was RUH HRT 208a Prune Shrubs and Small Trees. The original toolbox can be viewed at www.westone.wa.gov.au/toolboxes/horticulture. A quick review of this original toolbox unit will help in understanding the issues faced when attempting to "SCORM" it.

Following the model provided in the Carnegie Mellon (CM) *Best Practices Guide for Content Developers*, The unit Prune Shrubs and Small Trees can be aligned as a module with each of the four activities being a lesson (see Figure 1 below).

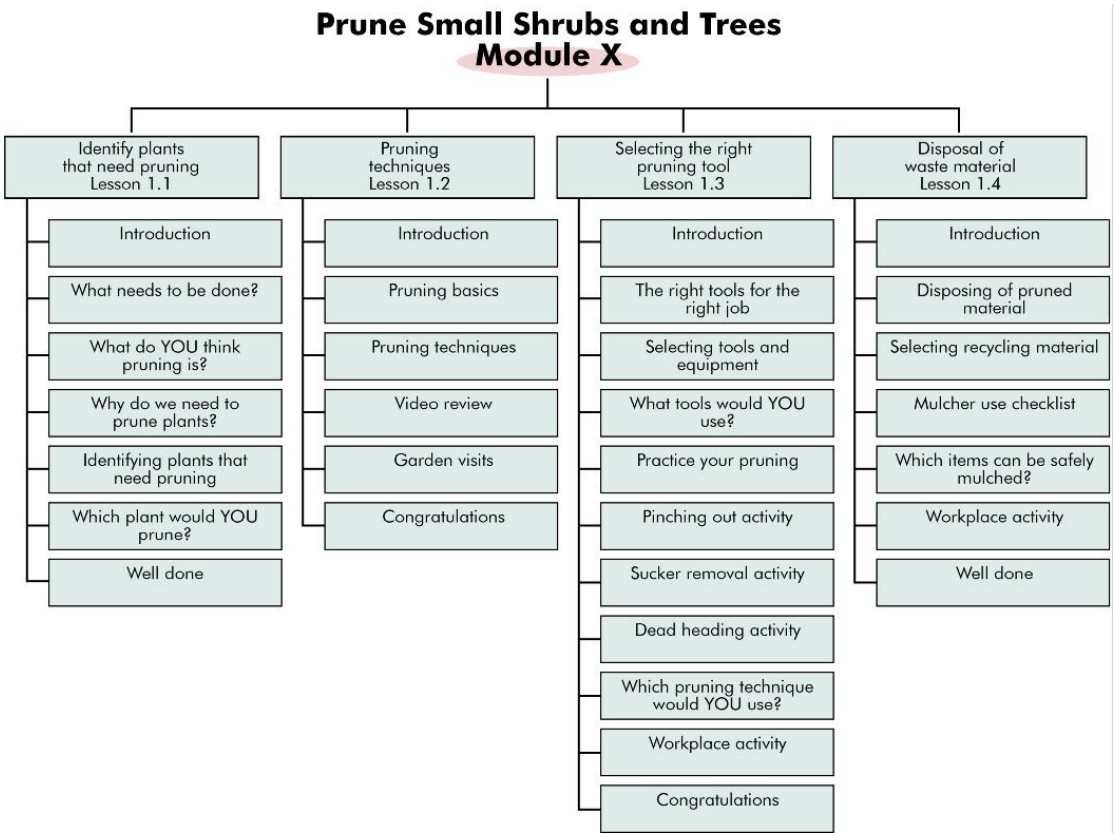

Figure 1

Following the CM model, it would appear that, at a cursory glance, the eleven titles under Lesson 1.3 Selecting the right pruning tool for example, would be all the assets associated with that lesson and would need to be incorporated into an SCO in order for that lesson to stand alone and work properly (see Figure 2 below).

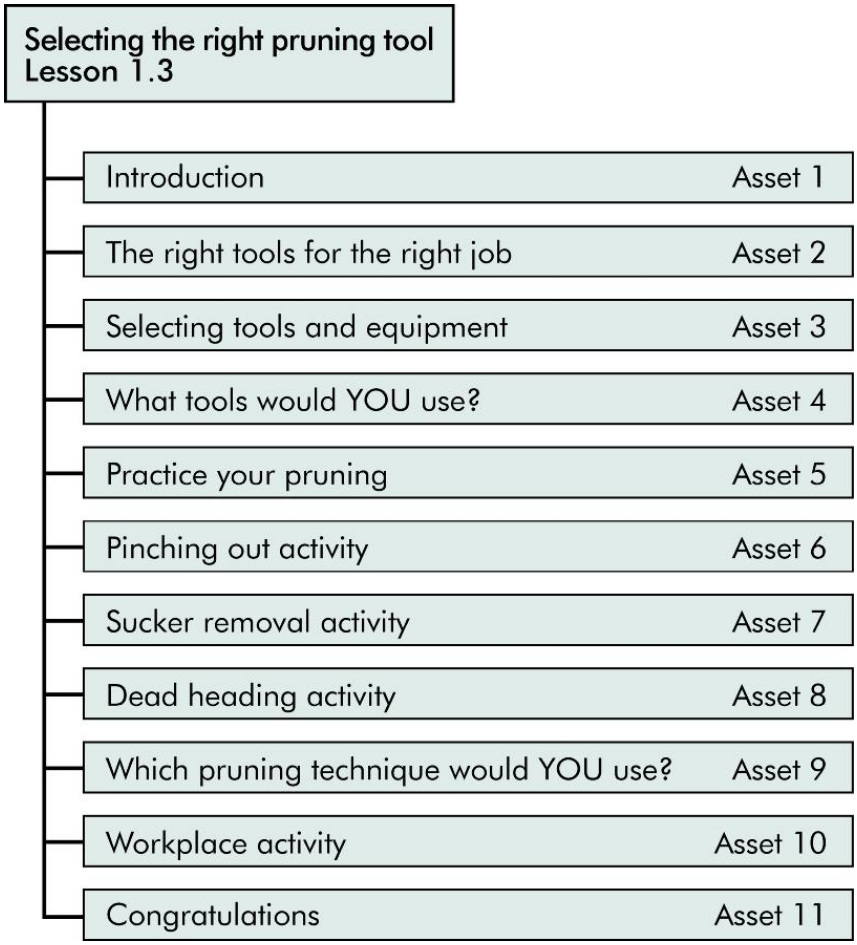

Figure 2

As can be seen in Figure 2, each activity within the Toolbox follows a fairly linear sequence of instructions, events and exercises; however, these activities also referred learners to a range of resources that were located outside of the activity and which were shared between activities in the same and in other units.

For example, in completing an activity on “selecting the right pruning tool” learners had to read three articles in the Horticultural Thymes magazine, view videos from the Bookcase and download two logsheets to guide their workplace activities.

Moreover, some terms that appear in the activity are also linked to glossary definitions. Hence, to deliver the activity as an SCO, these resources needed to be packaged as assets of that SCO.

So a map of the actual assets required for the lesson to work properly and stand alone would look more like that shown in Figure 3 below.

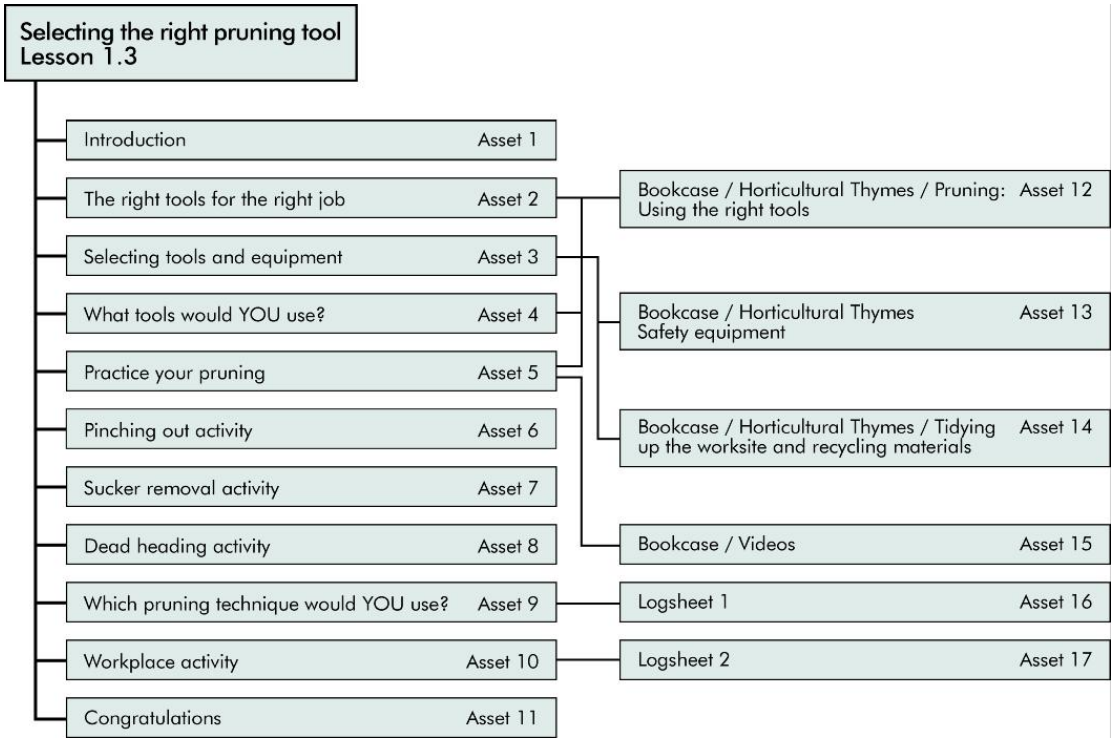

Figure 3

Therefore, if the lesson on Selecting the right pruning tool is to be packaged as a self contained SCO it must have all 17 assets included in that SCO for it to function properly, as shown in Figure 4 below.

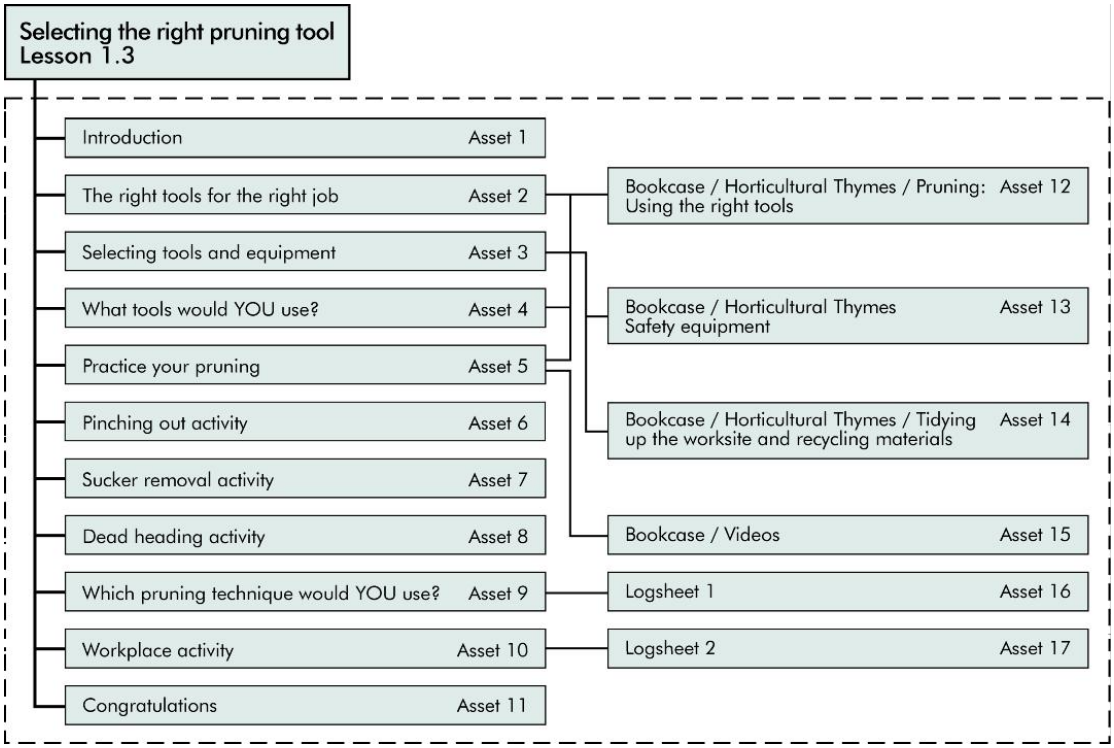

Figure 4

In the original Toolbox a number of resources (such as articles appearing in the Horticultural Thymes) are common to a number of activities. Within the Toolbox

learning model this is not a problem, and in fact makes for good design as each lesson would draw on the same resource. However, in acknowledgement of the SCORM golden rule, each activity (or SCO) can not link to or reference another activity (or SCO). This requires these common resources to be included in each SCO.

Packaging the same resource within each SCO would result in duplication, which raises challenges in relation to file size and maintenance of content if the SCOs are made available through a repository (see Figure 5 below).

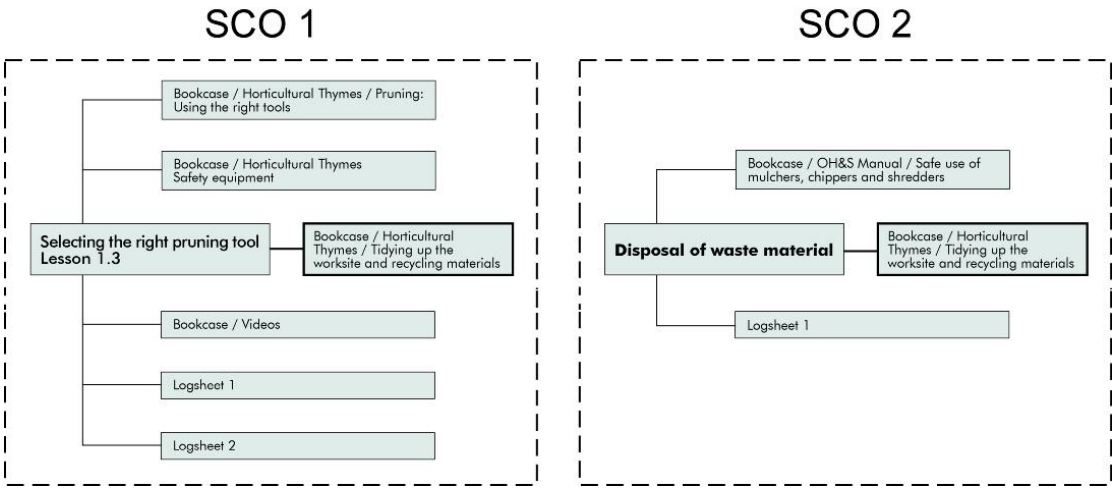

Figure 5

On a development level, the issue of duplication also occurs with lower level assets, such as images (*.jpg, *.gif), stylesheets (*.css) and other similar files. Typically, these common lower level assets exist in a centralised area and are then shared across the product. Before a lesson can be SCORMed, these shared component assets need to be harvested from the product and restructured. For example, Figure 6 and Figure 7 below:

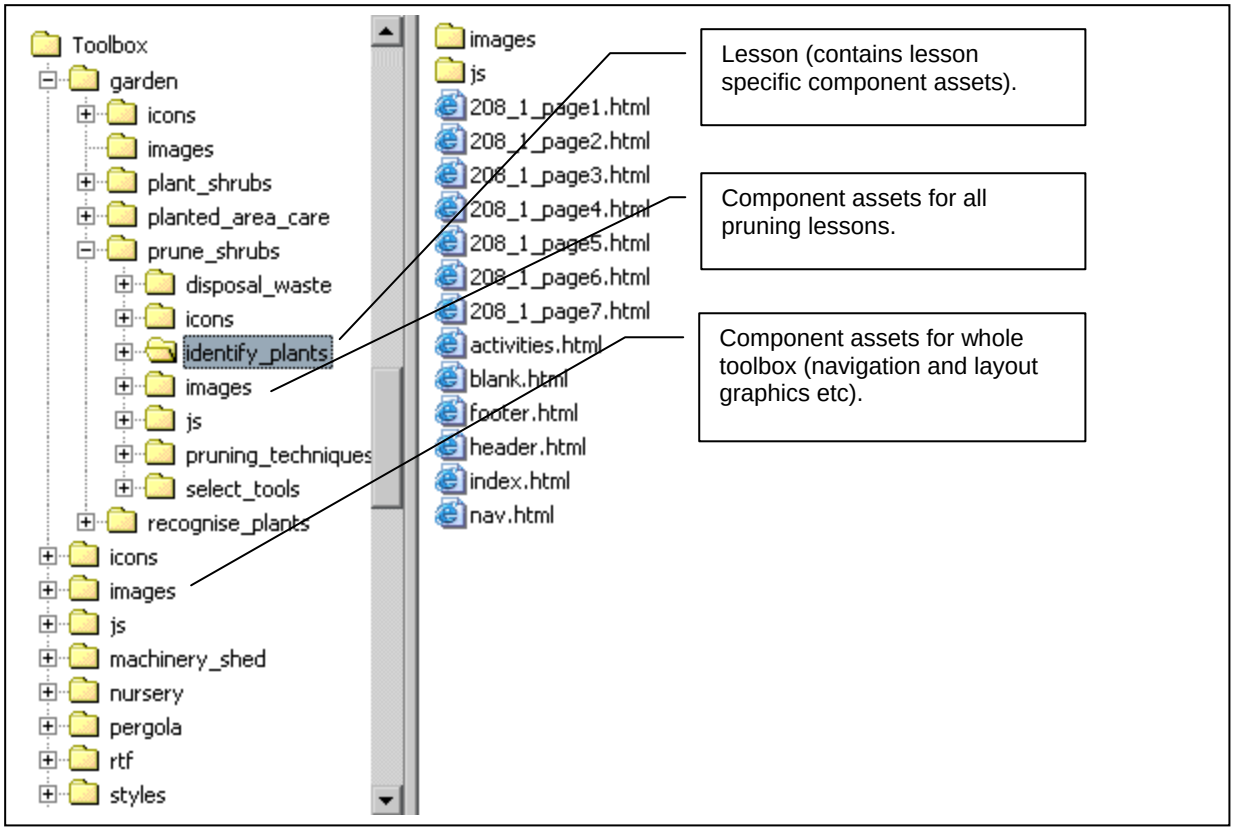

Figure 6 Original Toolbox Structure

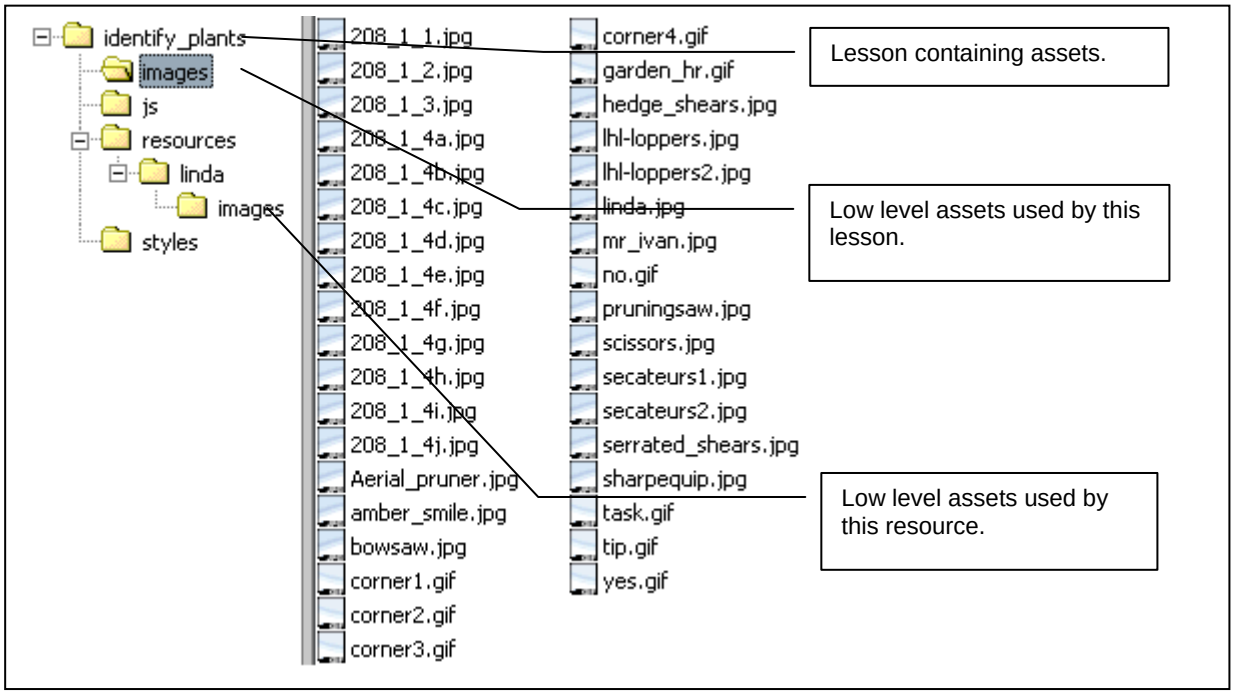

Figure 7 SCORMed Toolbox Structure

A solution was to package only those resource pages and corresponding low level assets that were directly relevant to the exercises in an SCO and refer to these specifically within the activity. For instance, if exercises in an activity require the learner to access information about pruning techniques from the Bookcase videos

and safety tips from the Horticultural Thymes, then only these pages would be packaged as assets in that SCO.

While this may appear to be an elegant solution, there are still some significant problems on both a pedagogical and developmental level:

- Does not make full use of Toolbox content.
  This approach ignores the content richness of Toolboxes and limits the opportunities for incidental learning that can occur when a learner is able to engage with such rich content at will.
- Larger file size for combined assets than original toolbox.
  For smaller assets or resources this is not a major problem, but when large resources are repeated across multiple SCOs, this can cause a significant increase in file size from the original Toolbox.
- Maintenance and customisation issues.
  With component assets and resources being unique to each object, if changes are made in one object then the changes are not passed on to other objects using the same component assets or resources.

Creating a SCORM package

Once the object had been mapped and its assets removed from the original Toolbox and formed into a discrete object, the next process was to access a SCORM packager. The case study used a freely available tool called Reload Editor (available from http://www.reload.ac.uk) to create the object. Figure 8 shows the process of creating a new ADL compliant SCORM package using the Reload Editor.

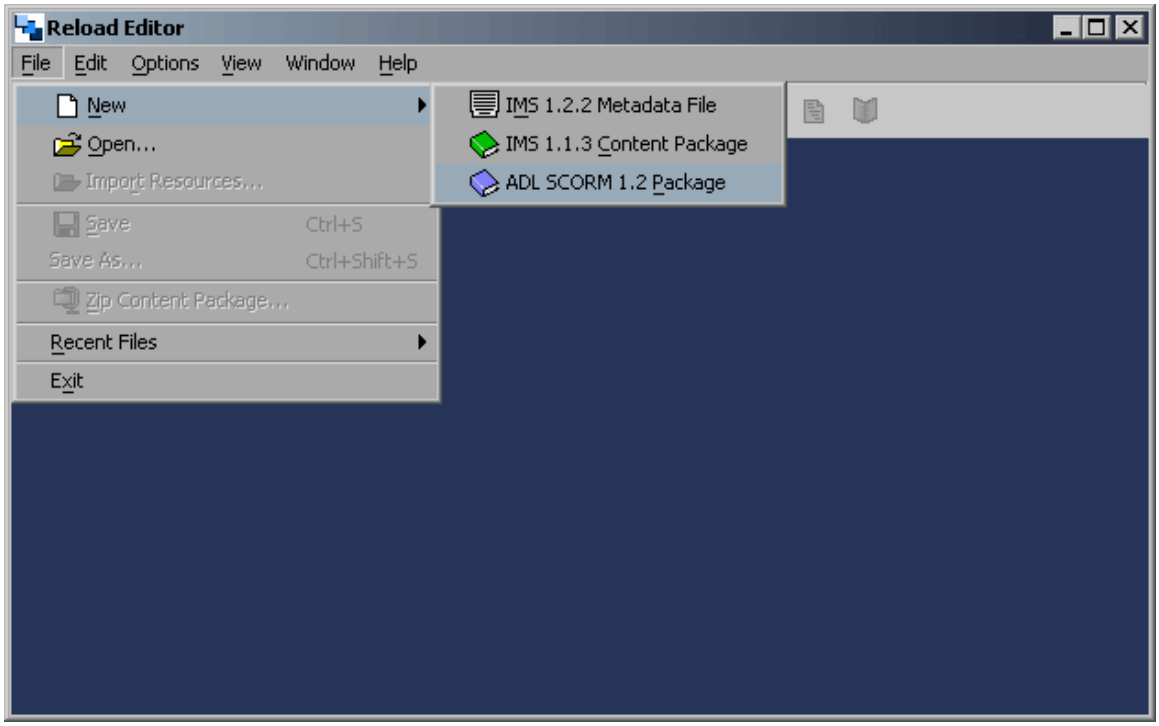

Figure 8 The Reload Editor in action

Once the content has been selected, Reload is able to automatically create an IMS manifest file for your object as shown in Figure 9.

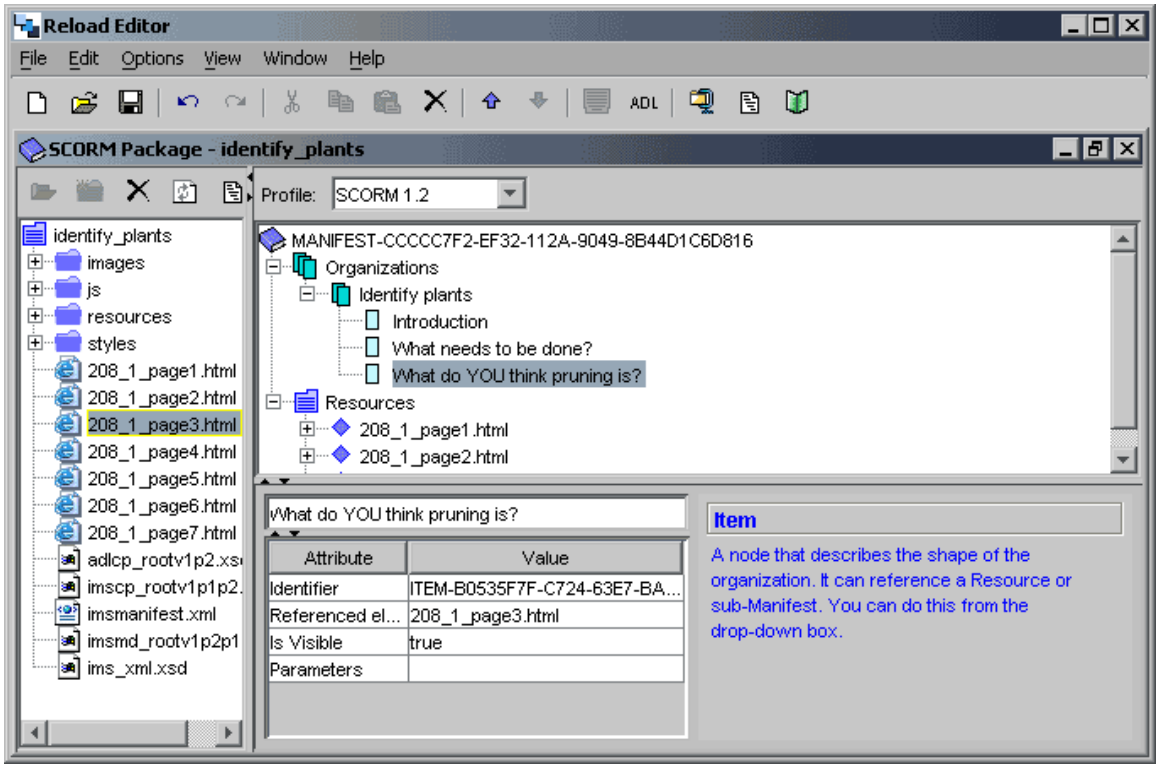

Figure 9 Creating a SCORM package

As each page is added, the corresponding low level assets are automatically added to the resources section of the manifest. At this stage, it is possible to attach metadata to the SCO or any of its assets that appear in the manifest. If the object uses assets that are not part of the organisation it is possible to add them just to the resources. This allows metadata to be attached to any asset inside the SCO.

How will it look in an LMS?

To trial the object as it would appear in an LMS, the case study used the Reload SCORM Player (also available from http://www.reload.ac.uk) shown in figure 10.

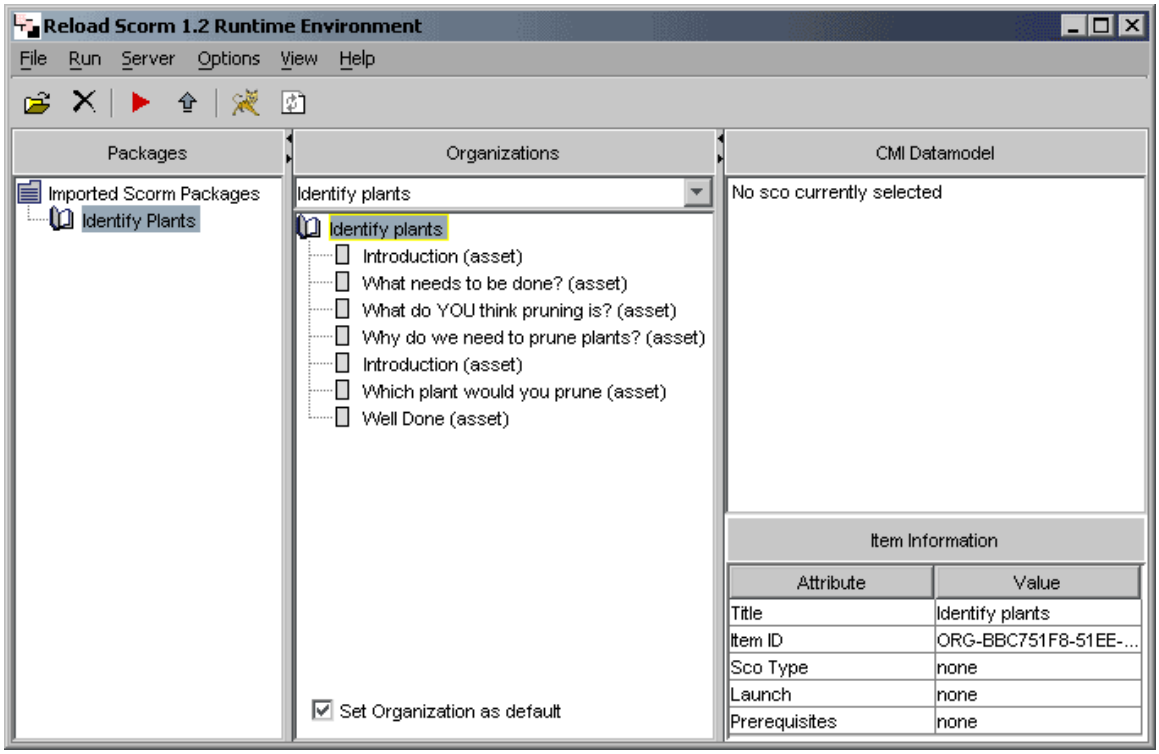

Figure 10 The Reload Player

By simply importing the package and running it in the player, it delivers the object inside its own navigational structure as shown in Figure 11.

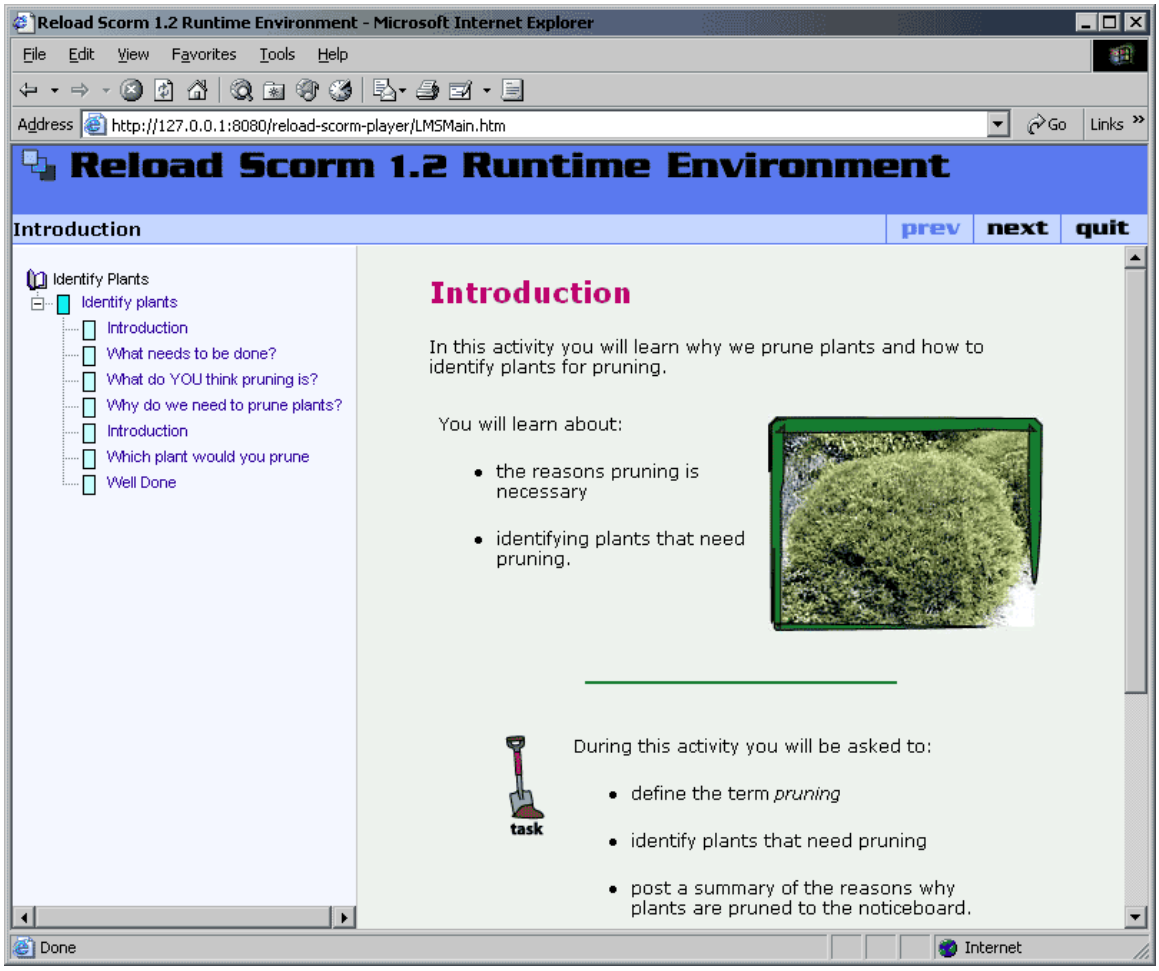

Figure 11 The completed activity using Reload navigation

Figure 12 shows the original Toolbox navigation.

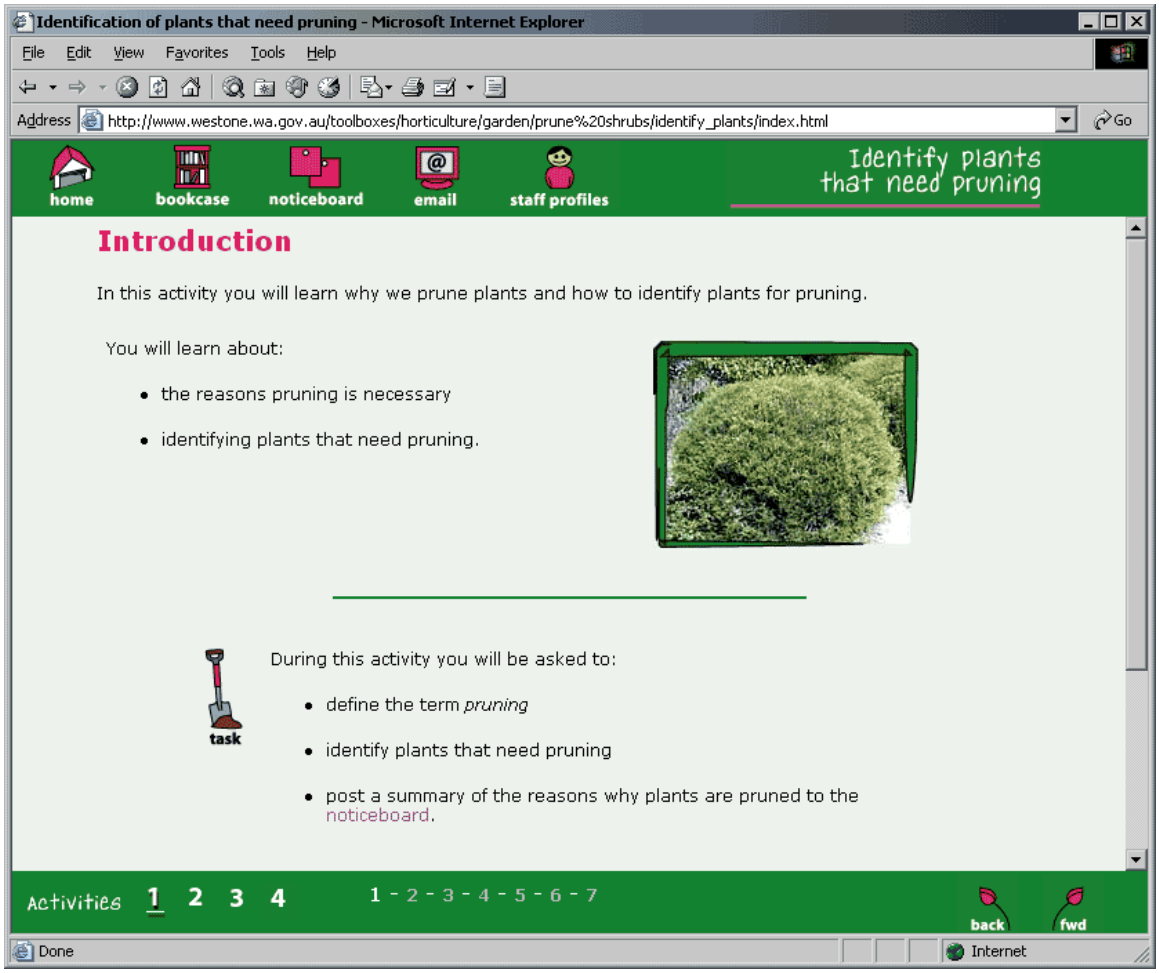

Figure 12 Navigation and layout as per the original Toolbox

As can be seen when comparing Figures 11 and 12 it was necessary to remove any navigational references from the content. In addition, the original Toolbox included links for communication tools (such as the noticeboard in Figure 12). While the Reload SCORM player does not feature any online communication tools, these links were retained: however, this does raise the point that such references would need to be tailored to the LMS or removed from the SCO altogether.

After getting this package to work inside the Reload SCORM player the development team attempted to import the package into WebCT 4.0 using its IMS tools. The package, although compliant with Reload Player, was not compatible with WebCT's importing tools. This highlights the differences that can be faced in attempting to prepare universally compatible SCOs.

Findings of the case study

Developers and instructional designers found that the deeper they went into the resources the more assets they found. Many of the external resources had links to glossary items or cross-referenced other resources within the bookcase or staff profiles. So a conscious decision had to be made about what level to stop including assets in the SCO. Consequently, the SCO will "work" to a predetermined level but some very deep links will not find their target.

Asset inclusion also needed to be selective. For example, rather than including all the resources from the Bookcase in the SCO, only those referred to by the lesson screens were included. On the positive side, this helps decrease file size of the SCO; however, the downside of this selectivity is that it ignores the content richness of individual Toolboxes and limits the opportunities for incidental learning that can occur when a learner is able to explore the rich content provided at will.

The context specificity of communication tools and on page references to navigation will also impact on any customisation that may be required. For example the instruction “go to the ‘Noticeboard’” is specific to the horticulture toolbox scenario. As the intention is to be able to plug an SCO into a variety of learning management systems with a variety of names for their communication tools, this would need to be changed to something more generic.

As noted in the Carnegie Mellon *SCORM Best Practices Guide for Content Developers*, the adoption of SCORM will ultimately affect the way instructional resources are designed suggesting that, “the instructional techniques you traditionally employ may have to change slightly as you create SCORM-compliant instruction.” The team of instructional designers working on the redesign of the Horticulture Toolbox would have to agree.

Summary

This paper has presented an overview of SCORM and has discussed a range of issues regarding its implementation and use within the Australian VET sector. To further inform the debate, this section will summarise these issues and present them in terms of them being perceived either as an advantage or disadvantage to the Australian VET sector.

Advantages of SCORM

SCORM is an international standard

One of the major advantages of SCORM is its recognition and significant adoption, particularly by the US military and international industry as an e-Learning standard. The standard is backed by the Advanced Distributed Learning Network (ADL) and has strong support of both IMS and the International Electrical and Electronic Engineering Associations (IEE) who are committed to ensuring its ongoing evolution and support.

As such, SCORM is currently the defacto standard for e-Learning and is supported by several of the major LMS[4] developers.

Sharing and re-use of learning materials

This promises to be one of the greatest advantages of implementing SCORM. SCORM promotes the use and re-use of learning resources in different formats and coming from different sources. This is achieved through a common API and a common distribution format that allows courses to register and retrieve data in a relatively generic way, making them auto-configurable and extensible.

[4] Such as WebCT and Blackboard.

A standardised approach to content packaging

SCORM uses the IMS content packaging format which allows for the simple distribution of courses since they can be created and stored in the way their creator needs. A standardised manifest file describes their structures and makes their manipulation easy.

The IMS content packaging is well supported by the majority of LMS developers.

Flexibility in content presentation

SCORM has a very sophisticated way of packaging content via the manifest that makes it easy to reorganise the content or even to have several alternative organisations within one package for different purposes. This can greatly reduce the cost of creating learning resources for specific purposes.

Interoperability across systems

SCORM-compliant content can be used on any SCORM compliant learning management system. This means that content from many different sources can be used on any SCORM compliant LMS.

Promoting discovery and re-usability

The SCORM Metadata Application Profile directly references the IEEE Learning Object Metadata (LOM) standard and provides roughly 64 metadata elements across nine categories. While the standard element set may not be suitable for all purposes, the use of even a minimal metadata set will greatly enhance discoverability.

There is scope within SCORM to develop a VET specific metadata application profile.

Disadvantages of SCORM

SCORM is in the early stages of evolution

SCORM is not a standard itself, but rather a reference model that serves to test the effectiveness and real-life application of a collection of individual specifications and standards. As such SCORM is still evolving and changes in future versions have the potential to invalidate earlier implementations.

There are also inconsistencies in how different learning management system vendors and content developers have implemented SCORM, so interoperability is less then perfect.

Reliance on an LMS

SCORM uses System Directed Learning. It is intended for, and assumes, implementation and management within an LMS. As such individual SCOs do

not contain internal navigation systems and require third party software to supply this level of functionality.

Implementation of SCORM within LMSs is inconsistent

Because the SCORM standard is open to interpretation, packages that work one way in one LMS are not guaranteed to necessarily work the same in other LMSs (Newman 2002, ATLP 2001). This could mean that packages will need to be tailored to suit specific LMSs.

Duplication of common or shared resources

"The Golden Rule" of SCORM is that an SCO cannot call or access another SCO without first going through the LMS. Essentially this requires each SCO to be completely self-contained. This means that if an asset is referred to by more than one SCO then that asset would need to be duplicated across multiple SCOs, creating challenges in relation to file size and maintenance.

Pedagogy and context

SCORM claims to be "pedagogically neutral", in addition, an SCO by itself should be as independent of learning context as possible so that it may be re-used in different learning experiences to fulfil different learning objectives. Both these issues have been the basis for considerable debate as to their impact on the instructional quality of learning materials and the inherent need for contextualisation of learning experiences in the VET sector.

Context is regarded by many as the key to a meaningful educational experience in VET. Attempts to remove context to enable greater re-usability or SCORM compliance may damage the vast connections learners make from a realistic example or a concept placed in context.

It is however, early days for SCORM. Ongoing research and pragmatic testing and experimentation may be able to overcome some of these contentious issues.

Implications for the VET Sector

As an international standard, the adoption of SCORM holds many potential advantages for the sector. These include the ease of integration with an LMS, and the benefits of being interoperable across a range of platforms promoting sharing and re-use of learning content. However, as highlighted in this paper, there are also some concerns about the SCORM pedagogical model and the impact it may have of VET learners.

Through the work of the Flexible Learning Framework, the Australian VET sector currently has access to a large range of digital learning resources such as Toolboxes. While these resources have received much acclaim as quality teaching and learning materials (Oliver & Herrington 2001), they are not[5] SCORM compliant. In addition to the case study described in this paper trials are currently underway to investigate the possibility of "SCORMifying" these legacy resources. However, as

[5] They were never intended to be.

indicated in the case study presented in this paper, the early indications are that there are some limitations in reverse-engineering SCOs from existing material.

It should be noted that SCORM is a reference model, for the development, packaging and delivery of education and training materials. So rather than being a single specification, SCORM is actually a collection of specifications to create a “unified content model” to enable the re-use of learning materials across a range of products and platforms (Advanced Distributed Learning 2003).

It is not necessary for an organisation to adopt SCORM in its entirety, but allows organisations to apply those components of SCORM that they believe will be of benefit to the organisation and its learners. The three main components that need to be considered are:

- the run-time environment
- content packaging specifications
- metadata specifications.

As all communication between a SCORM compliant LMS and the content is handled by the runtime environment, there is limited scope for debating the necessity of this component. However, as noted earlier, the standard API consists of eight functions, of which only the Initialise [LMSInitialize()] and the Finish [LMSFinish()] function are compulsory to meet SCORM compliance. The other six functions are optional but are recommended.

However, the adoption and implementation of content packaging and metadata specifications command significant interest to the ongoing development of learning objects and learning object repositories within the VET sector.

Content packaging specifications

There is a need to adopt an appropriate content packaging specification that suits the need of both lecturers and students within the Australian VET sector. The IMS Content Packaging specification as implemented by SCORM is currently the defacto standard for content packaging internationally and is currently supported by a range of LMS.

The case study presented in this paper has indicated that it may be possible to adopt the SCORM IMS content packaging specification for use with VET sector learning resources. However, it should be noted that there are also a series of practical trials currently being conducted by ANTA and the Flexible Learning Framework examining the suitability of SCORM content packaging for the VET sector.

While it is likely that the future of content packaging for the VET sector will be based on the SCORM IMS content packaging model, it is important that the decision regarding the adoption of a content packaging model for VET are informed by the findings of these trials.

Metadata specifications

There is a need within the Australian VET sector for metadata that allows for access, search, selection, use, trade and management of learning resources within a digital repository. The current SCORM metadata application profile directly references the IEEE Learning Object Metadata (LOM) standard consisting of roughly 64 metadata elements across nine categories. It is recognised that this profile may not be suitable

for all purposes, and that there is a need to develop a VET specific metadata application profile.

This application profile needs to be based on existing international metadata standards such as (Dublin Core, EdNA and IEEE's LOM) and be scalable to include both VET specific and organisation specific elements. The resulting metadata application profile should support the IMS content packaging specification as implemented by SCORM.

The development of such a metadata application profile should be informed by and contribute to other relevant national initiatives including other Framework projects[6], projects conducted by ANTA and cross-sectoral work occurring through the AICTEC Standards subcommittee, the Framework's Interoperability Committee and education.au.

[6] Such as the Flexible Learning Framework Collaborative Interoperability Project.

References

Advanced Distributed Learning (2002) *The SCORM Implementation Guide: A Step by Step Approach,* Advanced Distributed Learning available online at http://www.adlnet.org

Advanced Distributed Learning (2003) *Sharable Content Object Reference Model (SCORM) Version 1.2,* Advanced Distributed Learning Inc available online at www.adlnet.org

Advanced Distributed Learning (2003) *Sharable Content Object Reference Model (SCORM) Version 1.2,* Advanced Distributed Learning Inc. Available online at www.adlnet.org

Advanced Distributed Learning (2003) *Sharable Content Object Reference Model (SCORM™) Version 1.2 Conformance Requirements,* Advanced Distributed Learning Inc http://adlnet.org/index.cfm?fuseaction=rcdetails&libid=285&bc=false

ALTP 2001, ALTP News June 16 2001, ALTP. Available from http://www.altp.org/news/061601.htm

Anderson, A and Mah, S. (2002) *Keeping it Relevant: An Object Approach to Training Package Content Development that Facilitates Workplace Contextualisation.* HERDSA Conference Proceedings July 2002, HERDSA. Available from http://www.westone.wa.gov.au/portaladmin/PDF/westoneupload/Keeping%20it%20Relevant%20-%20Learning%20Objects%20-%20WestOne.pdf

Carnegie Mellon University (2003) *SCORM Best Practices Guide for Content Developers,* Carnegie Mellon University, Pittsburgh, Pennsylvania, USA. Available from http://www.lsal.cmu.edu/lsal/expertise/projects

Higgs, P, Meredith, S and Hand, T. (2003) *Technology for Sharing – A research project to inform VET Australia about Learning Objects and Digital Rights Management including systems and metadata to support them.* Flexible Learning Leader 2002 Report, Australian National Training Authority (ANTA).

Ip, Allbert and Canale, Ric, "Supporting Collaborative Learning Activities with SCORM" *Educause in Australia* 2003 conference paper .pp 669 – 678. Available: http://users.tpg.com.au/adslfrcf/scorm/ED031016.PDF

Koper, R. (2003) *Combining Re-Usable Learning Resources to Pedagogical Purposeful Units of Learning.* Journal of Interactive Media in Education, JIME.

Kraan, W and Wilson, S. (2002) "Dan Rehak: SCORM is not for everyone", CETIS. Available from: http://www.cetis.ac.uk/content/20021002000737

Kraan, W. (2002) "SCORM is not for everyone: ADL responds", CETIS. Available from http://www.cetis.ac.uk/content/20021017153823

Newman T., (2002) SCORM in a teacup? e-learning age,. Available on line at http://www.informationtransfer.com/elearning/download/InformationTransfer_SCORM.pdf

Oehlert, M. (2002) Discussion post for a thread in response to an article by Welsch, E. *SCORM: Clarity or Calamity*. Available from: http://instructionaltechnology.editthispage.com/discuss/msgReader$3094?y=2003&m=12&d=2

OIliver R., & Herrington J. (2001) *Teaching and learning online: a beginner's guide to e-learning in higher education.* Centre for Research in Information Technology and Communications, Edith Cowan University, Western Australia.

VET Learning Object Repository Green Paper for Discussion (2003) Australian Flexible Learning Framework.

Welsch, E. (2002) *SCORM: Clarity or Calamity.* Online Learning Magazine. Available from: http://www.onlinelearningmag.com/onlinelearning/magazine/article_display.jsp?vnu_content_id=1526769

Wiley, D. (2000) *Connecting Learning Objects to Instructional Design Theory: A Definition, a Metaphor, and a Taxonomy*, reusability.org Available from http://www.reusability.org/read

Wiley, D. (2003) *Introduction to Part 2 of Reusing Online Resources: Design Perspectives*, Journal of Interactive Media in Education, (1) Special Issue on Reusing Online Resources. ISSN: 1365-893X. Available from http://www.jime.open.ac.uk

Wilson, S. (2002) *Content Packaging Interoperability tests reveal room for improvement* CETIS. Available from: http://www.cetis.ac.uk/content/20020307103412

Acknowledgments

The VET Learning Objects Repository Project is managed by WestOne Services.

WestOne would like to acknowledge the following contributors to this paper:

Linda Arnold	Resource Generator
Vivienne Blanksby	Office of Training and Tertiary Education
Maxine Brodie	Macquarie University
Graeme Brownfield	Office of Training and Tertiary Education
Michael Coghlan	Douglas Mawson Institute of Technology
Les Comley	Resource Generator
Carol Fripp	AEShareNet
Gerry Green	TAFE Queensland
Tim Hand	Open Training & Education Network
Peter Higgs	TAFE Tasmania
Louise Housden	West Coast College of TAFE
Jill Jamieson	Challenger TAFE
Dennis McNamara	AEShareNet
Kerry Munston	TAFE Queensland
Dan Nicholas	EdNA Online Project
Gary Putland	education.au.edu.au
Simon Sinnot	The Learning Federation
Ralph Wirski	Edith Cowan University

This report has been researched and written by WestOne

For more information on the Australian Flexible Learning Framework contact:

Framework Communications Team
Phone: (07) 3225 3544
Fax: (07) 3237 0419
Email: aaron.snell@det.qld.gov.au