What about SCORM 1.3?

The SCORM 1.3 specification looms on the horizon. The content created according to the guidelines in this document will not automatically be rendered obsolescent by that or other future specifications because:

- Since SCORM 1.2 is much simpler than SCORM 1.3, it is expected that support for SCORM 1.2 for general content delivery, migration and archival will continue for a long time, particularly since there are entire classes of content that do not require the new features introduced in SCORM 1.3.
- It is expected that the implementation of SCORM 1.3 conformance in commercial may take anywhere from 3 to 24 months. While some vendors were quick to ship products that complied with SCORM 1.2, many other vendors are still working on their first SCORM 1.2 compliant implementation long after that specification was released.
- Systems that support SCORM 1.3 content are expected to also support SCORM 1.2 content, either directly or through the use of a generic adapter.
- This documents describes a layered approach that allows wholesale upgrades of SCORM content object from 1.2 to 1.3 compliance by just upgrading one or two generic script files.

An update of this document is planned for release after the SCORM 1.3 specification is released. The plan includes the inclusion of a section to explain how to update content created according to this document to be fully SCORM 1.3 compliant.

About authoring tools

The authoring tool used to develop the examples in this document is a simple text editor. However, these examples are not populated with the rich content and interactions that a real authoring tool can facilitate. Tools such as Click2learn's Aspen web-based enterprise LCMS, or desktop authoring tools such as Click2learn's ToolBook 8.5, do much of what is described in this document automatically: You just specify that you want to export the content as a SCORM package and they do it for you. Those tools are designed specifically to isolate you from the technical minutiae described in the SCORM specification and in this document, so that you can focus on what your content shows, tells and teaches.

Note that if when you use ToolBook, Aspen LCMS or another SCORM-conformant authoring tool or content generator to create the content, the actual code and web document structure those tools generate will be quite different from what is described in this document. There are many ways to achieve SCORM conformance, and that is good, because it leaves a lot of room for creativity.

About the examples

The examples in this document illustrate how content that conforms to the SCORM specification *can* be implemented. You may use any of the code included in the samples described in this document but, if you do, proper credit should be given to Click2learn, Inc, including a reference to this document. If you choose to reuse any of the sample code, it is entirely at your own risk. The sample listings and the code in the companion files in working examples are not intended to be actual production code and do not represent that Click2learn is actually implementing such code in its products. Some code and features that could be useful in an actual product were deliberately not included because they are considered proprietary, or trade secrets of Click2learn, Inc. In particular, error handling is minimal in order to keep the sample listings brief, and exhaustive testing using all the browsers in common use has not been done on most of these samples. The examples have been tested with Microsoft Internet Explorer 6.0 and should work with other browsers and versions.

Your own implementation, coding style and policies may be completely different from what is shown here, but still conform to SCORM and interoperate predictably with other SCORM conformant systems. That is the beauty of a standard. Interoperability is what matters, not how you implemented it.