

Zur SCORM-Fähigkeit konventioneller XML-unterstützender eLearning Developer Tools im Projekt I-can-EIB¹

Prof. Dr. Claus Möbus
Carl von Ossietzky Universität, Department für Informatik

**Jan Feindt, Claudia Janßen, Stefan Sölbrandt,
Heinz-Jürgen Thole, Jens Krefeldt, Holger de Vries**
Kuratorium OFFIS e. V.

Abstract. Mit diesem Beitrag stellen wir einen Workflow zur Entwicklung standardisierter Lerninhalte vor, der Adaptivität und Wiederverwendbarkeit, sowie Plattform- und Toolunabhängigkeit fördert. Standardisierung von eLearning-Content nach SCORM wird auch aus kommerziellen Gründen immer interessanter. Um SCORM-kompatiblen Content zu entwickeln, stehen zwei Wege offen. Erstens können bestehende Tools erworben und eingesetzt werden. Als großer Nachteil stellt sich dabei heraus, dass der bewährte Workflow nicht beibehalten werden kann. Zweitens können bereits genutzte Developer Tools weiterentwickelt werden. Im Projekt I-can-EIB haben wir uns für die Modifikation des genutzten, XML-basierten Drehbuch-Editors entschieden, um den gewohnten Arbeitsprozess der Content-Entwickler zu erhalten.

In diesem Beitrag stellen wir unsere übertragbare Lösung vor: nach einer einleitenden Beschreibung der Rahmenbedingungen und des Workflows bei der Content-Entwicklung geben wir einen Überblick zu Standardisierungsinitiativen. Kern des Beitrages bildet die Modifikation des Drehbuch-Editors zur Generierung SCORM-kompatibler Lerneinheiten.

1 Content-Entwicklung im Projekt I-can-EIB

Durch zunehmende Verbreitung des Internets finden eLearning und eTraining in der universitären Lehre und der beruflichen Aus- und Weiterbildung immer mehr Anwendung.

Diese Entwicklung bietet eine Reihe von Chancen (unter anderem Einsparpotentiale), gleichzeitig wird evident, dass fehlende Standardisierung die Kosten in die Höhe treiben kann, da Entwickler Bausteine anderer Systeme nicht ohne aufwendige Reimplementation wiederverwenden können. Um diesen Nachteil zu beheben, wurde eine Reihe Standardisierungsinitiativen gegründet. Auf dem Weg zum Standard befindet sich SCORM (Shareable Content Object Reference Model). SCORM setzt sich zurzeit international durch und bildet die Grundlage der hier vorgestellten Maßnahmen.

Die Entwicklung standardisierter Lerninhalte wird in diesem Beitrag in einen Workflow eingebettet, der Adaptivität und Wiederverwendbarkeit, sowie Plattform- und Toolunabhängigkeit fördert. Eine eingehenden Analyse des State of the Art hat ergeben, dass vorhandene Lösungen zur SCORM-Kompatibilisierung für unser konkretes Ziel nicht geeignet sind. Es gibt Konverter die bereits vorhandene Materialien, wie zum Beispiel Word-Dateien oder Powerpoint-Präsentationen, in SCORM-kompatible Objekte umwandeln. Diese Systeme sind zwar leicht zu bedienen, unterstützen aber keine

¹ Das vom BMWA geförderte LERNET-Projekt I-can-EIB (Innovative CBT-Architektur im Internet für den Europäischen Installationsbus) wird vom OFFIS in Zusammenarbeit mit dem Bundestechnologiezentrum für Elektro- und Informationstechnik e. V. (bfe) durchgeführt. URL: <http://www.i-can-eib.de>

Strukturierung durch Hierarchien. Andere proprietäre Multimedia-Werkzeuge (z. B. Macromedia Authorware, Director) sind sehr komplex, womit viele Fachautoren überfordert sind, da sie für ihre spezielle Aufgabe nur eine geringe Menge der Funktionalität benötigen. Unser Vorgehen dagegen verändert den Arbeitsprozess der Content-Entwickler und Fachautoren nur geringfügig, es werden lediglich einige zusätzliche Daten abgefragt (s. u.).

Anhand des im Projekt I-can-EIB entwickelten Drehbuch-Editors wird untersucht, welche Erweiterungen notwendig sind, um SCORM-kompatible Lernmaterialien zu erstellen und inwieweit dieser Prozess automatisiert werden kann.

Das Projektziel ist die Erstellung eines kombinierten Kommunikations-, Informations- und Lernsystems für eine neuartige Elektro-Installationstechnik, kurz gesagt ein *elektronischer Marktplatz für Wissen*.

Insbesondere wird dabei auf die verschiedenen Nutzergruppen des Systems, vom interessierten Laien über den Bauherren bis hin zum Architekten, mit ihren speziellen Anforderungen eingegangen. Diese Anforderungen betreffen dabei nicht nur den Content selbst, sondern auch dessen benutzergerechte Darstellung. Um die verschiedenen didaktischen Bedürfnisse [1,2] der Benutzer schon während der Entwicklung zu unterstützen, wurde das modulare „Windrosen“-Design [3] entwickelt, das je nach Anforderung der Benutzer auf deren Sichtweise abgestimmte Informationen in unterschiedlich komplexen Kursen bereitstellt.

2 Entwicklung eines Drehbuch-Editors zur Unterstützung von Autoren bei der Content-Entwicklung

Im Projekt I-can-EIB ist die Autorenunterstützung ein zentrales Thema. Die Autoren sollen zum einen in ihrer Konzeptionsphase unterstützt werden, zum anderen soll die Wiederverwendung der bereits entwickelten Lerninhalte gefördert werden. Bei Projektbeginn wurde deshalb der Ablauf der Content-Entwicklung beobachtet und festgestellt, dass es eine strikte Arbeitsteilung zwischen Drehbuch-Autoren und Multimedia-Entwicklern gibt. Für die Autoren bedeutet diese Arbeitsteilung, dass sie sich nicht in die umfangreichen Multimedia-Tools einzuarbeiten brauchen, sondern sich vorrangig mit didaktischen Aspekten beschäftigen können.

Die Autorenunterstützung durch einen Drehbuch-Editor sollte eine umfangreiche Reorganisation des Workflows vermeiden und doch Rahmenbedingungen in Form von Schablonen bzgl. des Aufbaus der Lernmodule vorgeben. Schablonen in Form von vordefinierten Eingabemasken fördern nicht nur die angeleitete konzeptionelle Umsetzung, sondern ermöglichen auch, die Entwicklungszeit und –kosten besser abzuschätzen. Weitere Anforderungen entstanden aus den Vorgaben, Drehbücher offline bearbeiten und sie in einer geeigneten Datenbank persistent halten zu können, sowie die Extraktion der Inhalte im PDF-Format zu ermöglichen. Außerdem sollten Konvertierungsmöglichkeiten bestehen, um auf zukünftige Standards reagieren zu können. Ein besonderes Augenmerk wurde auf das Content Sharing gerichtet, um Inhalte über das Internet (z. B. als Web Service [4]) auszutauschen und Information Retrieval-Prozesse zu unterstützen. Zu diesem Zweck soll die Erfassung von Metadaten möglich sein, um diese zur Identifikation von Ressourcen heranziehen zu können.

Aufgrund der Anforderungsanalyse wurde eine Toolrecherche durchgeführt. Die bekannten proprietären Drehbuch-Editoren (Autorensysteme) lassen sich grob in kartenbasierte, zeitachsenbasierte, strukturorientierte und buchorientierte Ansätze kategorisieren [5]. Jeder Ansatz verfolgt eine andere Philosophie und verlangt von den Autoren zur Umsetzung ihrer Ideen ein differenziertes Denkschema. Beispielsweise wird im zeitachsenbasierten Ansatz mit der Bühnen-Metapher gearbeitet, in der die Objekte als Darsteller bezeichnet werden und Rollen übernehmen. Ein Regieplan steuert hierbei die Vorgänge zeitorientiert. In diese Kategorie gehört z. B. das Tool Macromedia Director [6]. Als Vertreter des strukturorientierten Ansatzes, in dem die Aktivitäten auf dem Bildschirm mittels Flussdiagrammen gesteuert werden, lässt sich Authorware [6] nennen.

Die Untersuchung der Tools ergab, dass relativ lange Einarbeitungszeiten notwendig sind, um annähernd akzeptable Ergebnisse zu erzielen. Zudem stellt sich die Benutzung der Tools, besonders für die Fachautoren mit geringer PC-Erfahrung, als schwierig dar. Die Umstellung auf ein anderes Werkzeug während der Content-Entwicklung bedeutet zudem ein nur schwer kalkulierbares Zeitproblem. Schließlich unterstützen die betrachteten Tools die Trennung von inhaltlicher Konzeption und multimedialer Umsetzung nicht in dem von uns und den Autoren gewünschtem Maß.

Aufgrund dessen wurde die Entscheidung getroffen, einen Editor nach Vorgaben des Projektpartners bfe Oldenburg, ein langjährig erfolgreicher eLearning-Anbieter im Bereich E-/IT-Technik, zu entwickeln. Ein eigenes XML-Derivat wurde so konzipiert, dass Richtlinien für Autoren durch Schablonen manifestiert und Metadaten integriert werden können. Der Editor wurde mittels XML-Spy [7] entwickelt, eine Umstellung auf andere XML-Editoren ist jedoch ohne große Anpassungen möglich.

Die Drehbücher umfassen jeweils ein Kapitel eines Kurses und bestehen durchschnittlich aus zwei bis drei Bildschirmseiten, die zeitachsenbasiert beschrieben werden.

Die Eigenentwicklung ist plattformunabhängig, kann zwecks Information Retrieval leicht in Web Services integriert werden und bietet per XSLT [8] die Möglichkeit, andere (zukünftige) Standards zu berücksichtigen. Zur Speicherung und Verwaltung der XML-Dateien wird die XML-native Datenbank Xindice [9] eingesetzt, auch hier ist ein Umstieg auf andere Produkte möglich. Der Drehbuch-Editor konnte nahtlos in den Workflow der Autoren integriert werden. Nach einer intensiven Pre-Test- und Pilot-Phase konnte die Handhabung optimal an die Bedürfnisse angepasst werden.

3 Standardisierung des Contents

3.1 ADL-SCORM

Besonders im Hinblick auf das Content Sharing ist die standardisierte Gestaltung der Lerninhalte unverzichtbar. Das Ziel ist es, die Lerninhalte durch Rekombination, Reorganisation und Rekontextualisierung in verschiedenen Anwendungen und Systemen nutzen zu können. Hier ist die Beachtung von Standards in der Content-Herstellung im Hinblick auf die Kursgestaltung und den Datenaustausch hilfreich.

Die Advanced Distributed Learning Initiative (ADL) wurde 1997 vom amerikanischen Department of Defense und dem White House Office of Science and Technology Policy gegründet und integriert verschiedene Standardisierungsansätze, u. a. von AICC, LTSC-IEEE, IMS und ARIADNE. Als Referenzmodell für web-basierte Lerninhalte wurde das Shareable Content Object Reference Model (SCORM) entwickelt. SCORM setzt sich aus dem Content Aggregation Model (CAM) und einer Runtime-Environment (RTE) zusammen. Das CAM beschreibt die Zusammenstellung von Lernsequenzen aus einzelnen Lerneinheiten. Die RTE stellt eine Schnittstelle zwischen dem Learning Management System (LMS) und einzelnen Lerneinheiten zur Verfügung [10, 11, 12].

Die kleinste Einheit in SCORM, vergleichbar mit einer Stand-alone-Lesson, ist ein Sharable Content Object (SCO). SCOs setzen sich aus nicht weiter zerlegbaren Bestandteilen (Assets) zusammen, z. B. Bilder, Texte oder Soundfiles oder enthalten wiederum SCOs [10, 11, 12, 13]. Die SCOs und Assets werden durch Metadaten beschrieben, die zur Identifikation und Suche herangezogen werden können und somit die Wiederverwendung erleichtern. Derzeit findet hierfür die Learning Object Metadata (LOM)-Spezifikation Anwendung [14]. Diese umfasst neun Kategorien mit verschiedenen (optionalen) Attributen, ist aber grundsätzlich erweiterbar. Grundlage der Metadaten-Standardisierung ist der Metadata Element Set der Dublin Core Metadata Initiative [15].

Um die Integration des Contents (SCOs) in die jeweiligen LMS und den Austausch zu ermöglichen, wurde im IMS-Projekt eine Content Packaging-Spezifikation entwickelt. Ein Content Package (CP) repräsentiert z. B. einen Kurs und besteht aus Lernmaterialien, Metadaten-Dateien und einem Manifest.

Das Manifest ist ein XML-Dokument zur Beschreibung des CP und stellt dessen zentrales Dokument dar. Es enthält Metadaten zur Beschreibung des CP selbst, eine Organisationseinheit zur Beschreibung der SCO-Hierarchie und Referenzen auf die Ressourcen (SCOs und Assets) [10].

Um die unabhängige Nutzung der einzelnen SCOs in anderen Kontexten zu ermöglichen, werden die SCOs nicht direkt verknüpft. Die Navigation durch die Lerninhalte erfolgt ausschließlich durch das LMS auf Grundlage der SCO-Hierarchie aus den Manifest-Dateien.

3.2 Generierung SCORM-kompatibler Lerneinheiten

Die verteilte Entwicklung multimedialer Lerninhalte gliedert sich im Projekt I-can-EIB in mehrere Schritte. Im ersten Schritt werden Themen, Konzepte und Content der Lerneinheiten von Fachdidaktikern ausgearbeitet und die Drehbücher erstellt. Im Anschluss daran wird dieses Drehbuch von Multimedia-Entwicklern umgesetzt, indem z. B. Flash-Filme mit Audio-, Video- und Textelementen, sowie Animationen erstellt werden. Gegebenenfalls enthält der Flash-Film am Ende eine Lernkontrolle.

Die Autoren notieren in den Drehbüchern bereits einige Metadaten (z. B. Keywords, Beschreibung des Contents, Zielgruppe) und in einer gesonderten Datei den Ablauf des Kurses (Reihenfolge der Kapitel).

Das Ziel ist es, auf Basis der Drehbücher und Metadaten ein Content Package zu erstellen, das in einem beliebigen SCORM-konformen LMS lauffähig ist. Für Testzwecke haben wir die ADL-Sample-RTE v. 1.2.1 [16] verwendet.

Der Drehbuch-Editor soll im Zuge der Standardisierung strukturell nur minimal verändert werden. Es werden lediglich Eingabefelder für die Metadaten, die gemäß der LOM-Spezifikation verpflichtend sind, hinzugefügt. Zu diesem Zweck wird der Workflow um einen Arbeitsschritt erweitert, in dem die – für die SCORM-Kompatibilisierung notwendigen – Metadaten abgefragt werden.

Die anschließende Transformation der Drehbücher in SCORM-kompatible Lerneinheiten wird automatisch mittels XSLT (Extensible Stylesheet Language Transformations) [8], einem Standard für textbasierte Transformationen, vorgenommen, ohne dass weitere Schritte seitens der Entwickler erforderlich sind. Diese Transformation ist in der Abbildung 1 schematisch dargestellt.

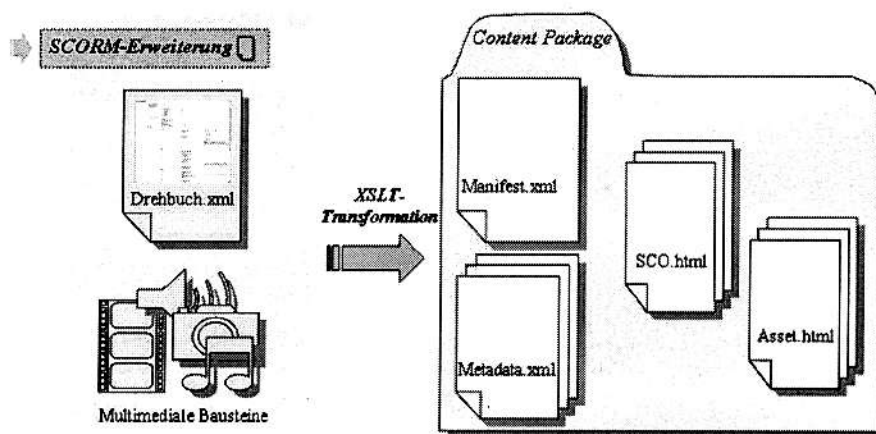


Abbildung 1: Generierung SCORM-kompatibler Lerneinheiten

Aus den Drehbuch.xml-Dateien und den multimedialen Bausteinen werden automatisch SCO.html und Asset.html generiert. Die benötigten Metadaten werden aus dem Drehbuch extrahiert, um zusätzlich erhobene Daten (SCORM-Erweiterung) ergänzt und in einzelne Metadata.xml-Dateien gespeichert. Die Kapitelhierarchie (Ablauf des Kurses) und die Referenzen auf die Ressourcen werden in der Manifest.xml-Datei festgehalten. Die dazu notwendigen Angaben notiert der Fachautor im Drehbuch. Diese Angaben nutzt das LMS zur Navigation durch den Kurs. Die SCO-interne Navigation bleibt davon unberührt.

Bedeutend war die Frage nach der Granularität der SCO-Generierung. Eine kritische Größe sollte nicht unterschritten werden, um den Aufwand der Metadatengenerierung bzw. Abfrage überschaubar zu halten. Zudem treten Schwierigkeiten mit der Verarbeitung der Link-Strukturen auf, da hierbei leicht essentielle Zusammenhänge auseinander gerissen werden können. Andererseits erschwert die Generierung zu großer SCOs das kontextkonforme Einbinden in andere Systeme.

4 Zusammenfassung und Ausblick

Wir haben eine Lösung vorgestellt, wie sich auf Basis eines xml-basierten Drehbuch-Editors und den zugehörigen multimedialen Bausteine automatisch SCORM-kompatible web-basierte Lerninhalte generieren lassen, ohne in den Workflow von Fachautoren und Multimedia-Entwicklern zu sehr einzugreifen. Diese Lerninhalte sind in hohem Maße wiederverwendbar, sowie plattform- und toolunabhängig.

Die generierten Metadaten können zur Identifikation von Ressourcen bei der Content-Entwicklung herangezogen werden. Diese können z. B. dazu genutzt werden, die Transformation zu parametrisieren und eine zielgruppenspezifische Gestaltung der Lerneinheiten zu erreichen.

Die vorgestellte Methode ist übertragbar und kann mit entsprechenden IT-Kenntnissen auf andere XML unterstützende Produkte angewendet werden.

Weitergehende Untersuchungen sind noch notwendig, um auch tiefer geschachtelte SCO-Hierarchien automatisch verarbeiten zu können. Des Weiteren ist eine automatische Analyse der internen und externen Verknüpfungen innerhalb bzw. zwischen den Lernmodulen nötig, um die geeignete Größe eines SCO zu ermitteln.

Autoren

Prof. Dr. Claus Möbus
Carl von Ossietzky Universität - Department für Informatik
26111 Oldenburg
E-mail: Claus.Moebus@informatik.uni-oldenburg.de

Dipl.-Inform. Jan Feindt
Dipl.-Inform. Claudia Janßen
Dipl.-Inform. Stefan Sölbrandt
Wiss. Mitarb. Heinz-Jürgen Thole
cand.-inform. Jens Krefeld
cand.-inform. Holger de Vries

Kuratorium OFFIS e.V.
Escherweg 2
26121 Oldenburg
E-mail: {Feindt, Claudia.Janssen, Soelbrandt, Thole}@offis.de

Literatur

- [1] Bruns, B., Gajewski, P.: Multimediales Lernen im Netz: Leitfaden für Entscheider und Planer, Berlin: Springer Verlag, 1999.
- [2] Jank, W., Meyer, H.: Didaktische Modelle, Frankfurt a. M. Cornelsen Scriptor, 1994.
- [3] Möbus, C., Albers, B., Garbe, H., Hartmann, St., Thole, H.J., Yakimchuk, V., Zurborg, J.: Towards an AI-Specification of Intelligent Distributed Learning Environments, KI - Zeitschrift Künstliche Intelligenz Heft 1/03 "Schwerpunkt:Lernen: Modellierung und Kommunikation", S. 19-24, Bremen, arendtap Verlag, 2003.
- [4] <http://www.w3.org/2002/ws/> Letzter Zugriff am 28.08.2003.
- [5] Dietrich Boles: Vergleich und Bewertung von Autorensystemen. Carl von Ossietzky Universität Oldenburg. <http://www-is.informatik.uni-oldenburg.de/~dibo/paper/ais/ais.html> Letzter Zugriff am 28.08.2003.
- [6] <http://www.macromedia.com> Letzter Zugriff am 28.08.2003.
- [7] <http://www.xmlspy.com> Letzter Zugriff am 28.08.2003.
- [8] <http://www.w3.org/TR/xslt> Letzter Zugriff am 28.08.2003.
- [9] <http://xml.apache.org/xindice/> Letzter Zugriff am 28.08.2003.
- [10] Dodds, P. (Hrsg.): Advanced Distributed Learning Initiative. Sharable Content Object Reference Model (SCORM). Version 1.2. The SCORM Overview. 2001. http://www.adlnet.org/screens/shares/dsp_displayfile.cfm?fileid=481 Letzter Zugriff am 10.06.2003.
- [11] Dodds, P. (Hrsg.): Advanced Distributed Learning Initiative. Sharable Content Object Reference Model (SCORM). Version 1.2. The SCORM Content Aggregation Model. 2001. http://www.adlnet.org/screens/shares/dsp_displayfile.cfm?fileid=477 Letzter Zugriff am 10.06.2003.
- [12] Dodds, P. (Hrsg.): Advanced Distributed Learning Initiative. Sharable Content Object Reference Model (SCORM). Version 1.2. The SCORM RTE. 2001. http://www.adlnet.org/screens/shares/dsp_displayfile.cfm?fileid=483 Letzter Zugriff am 10.06.2003.
- [13] Advanced Distributed Learning Initiative. The SCORM Implementation Guide: A Step-by-Step Approach. 2002. <http://www.adlnet.org/index.cfm?fuseaction=rcdetails&libid=493&cfid=339257&cftoken=69830896> Letzter Zugriff am 20.05.2003.
- [14] LTSC of IEEE 1484.12.1-2002 (2002): Draft Standard for Learning Object Metadata, Institute of Electrical and Electronics Engineers, Inc.15.07.2002. http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf Letzter Zugriff: 28.08.2003
- [15] <http://dublincore.org/> Letzter Zugriff am 28.08.2003.
- [16] <http://www.adlnet.org/index.cfm?fuseaction=rcdetails&libid=398&filterid=35&page=1&keywords=&applyto=libTitle,libAuthor,contentText> Letzter Zugriff am 28.08.2003.