Structure within the SCO
A Strategy for Effective Content Reuse

Tyde Richards
Eduworks Corporation
trichards@eduworks.com
Overview

- Problem Area
  - Requirements for representing content below the SCO level
  - Implications for effective content reuse

- Background
  - Two prototype projects sponsored by the Joint ADL Co-Lab

  - 2004-5 SCORE Prototype (LSI, Eduworks, NUWC)
    - Legacy content conversion into SCOs
    - Pedagogical and user interface consistency

  - 2005-6 XML SCORM Studio (Eduworks, MSState, LSI, NUWC, JADL)
    - Improve SCORE technical approach
    - Evaluate benefits of ADL OPEL
    - Base content format on open specifications
    - Improve usability

ADL International Plugfest
January 18, 2006
Reuse is a major goal of
- SCORM Program
- “Learning Objects”

Reuse entails
- Starting with instructional context
- Finding appropriate content
- Cutting & Pasting
- Re-styling
- Packaging for SCORM Delivery
Problem

- Tools do not support this type of reuse today
  - Formats are proprietary
  - Content is locally held
  - SCORM tools operate on *content* not *learning experiences*
2004-5 SCORE Project Goal

- Develop tools, techniques, and procedures to
  - Effectively disaggregate existing courseware
  - Separate presentation from content
  - Mix and match content from different sources
  - Make SCOs this way
  - Produce instructionally sound learning experience
Motivated by…

- Quantity of non-SCORM legacy content
- SCORM treats the SCO as a “black box”
  - No support for pedagogical consistency across SCOs from different sources
  - No support for user interface consistency across SCOs from different sources
- Solving the content conversion problem entails
  - Opening the black box
  - Defining a common content model for SCO-level content
How we assumed it would work

1. Source Content is converted into an XML-based Exchange Format by a Conversion Utility.

2. The Exchange Format is imported into the SCORE Tool where converted content can be edited into a styled, conformant SCO.

3. An Aggregation Tool such as Reload is used to aggregate and sequence SCOs into a conformant SCORM Package.

ADL International Plugfest
January 18, 2006
SCORE Prototype

Splash Screen. Displays copyright statement and Version number.

Main Option Panel. Provides interface to open Converter Utility, Editor/Disaggregation Utility or launches Reload.

Converter Input

Legacy Courseware Files

New Converter XML

Create Template

SCORE Exchange XML Template

SCORE Exchange XML

Create Template XML provides a template SCORE Exchange document for courseware vendors to populate with data that represents their content.

Converter Output

SCORE Data Base

SCORE Courseware Files

Import SCO Files copies necessary files from the legacy courseware file location(s) into a SCORE file structure.

Import From XML transfers the SCORE Exchange document into a SCORE database.

January 18, 2006
SCOR E
SCO
Editing

Splash Screen. Displays copyright statement and Version number.

SCORE Database

SCORE Courseware Files

Converter Output

SCORE XSL

SCORE Content XML

HTML

SCORE JavaScript

Main Option Panel. Provides interface to open Converter Utility, Editor/Disaggregation Utility or launches Reload.

Menu Panel. Provides the interface for launching all editing utilities.

SCORE Disaggregator. Disaggregates and Reaggregates content below the SCO level. Web pages are the smallest reusable units.

SCORE Editor. Provides the interface to edit SCORE web pages. Controls are provided to manipulate web page elements.

Selected web pages may be saved as a new SCO or merged into any other SCO.
Splash Screen. Displays copyright statement and Version number.

Main Option Panel. Provides interface to open Converter Utility, Editor/Disaggregation Utility or launches Reload.

SCORM Data Base

Converter Output

SCORE Courseware Files

Asset Metadata Utility

SCO Metadata Utility

Content Aggregation and Manifest Generator

IMS Manifest

Manifest Database
Fidelity of Converted Content

Before Conversion

Authorware Source Content

Unstructured HTML “Blob” Source Content

After Conversion
SCORE Prototype Demonstration

Version 1.0.0

Warning illegal reproduction of this program violates copyright laws.

ADL International Plugfest
January 18, 2006
SCORE Project Results

- Created pedagogically and graphically consistent SCOs using rich media content
  - From different sources
  - In different content formats
- Learned that structure, not just format, are important for converting courseware
- Developed repeatable content conversion methodology
- Formed opinions about representing SCO content
Project Result
The basic SCORE Process

- Source Content
- SCORE Conversion Utility
- SCORE Exchange Format
- SCORE Tool
- SCORM Package

ADL International Plugfest
January 18, 2006
Project Result
A Content Conversion Methodology

intended users

necessary support materials?

technical constraints on delivery format

intended pedagogic use?

Planning SCORE
Content Conversion

source content in convertible format

user interface style conventions

pedagogy-to-SCORM mapping

ADL International Plugfest
January 18, 2006
The Central Importance of a Data Model

ADL International Plugfest
January 18, 2006
Project Result

The Success of the Data Model depends on the Card Deck Metaphor

Card Deck

shared state information across cards

ADL International Plugfest
January 18, 2006
The Role of Generic and Learning-specific Content Elements

Project Result

ADL International Plugfest
January 18, 2006
Project Result
Content Element need to Share State
Data across Locations

Shared State Variables

Location 1
Content Element X

Location 2
Content Element Y

ADL International Plugfest
January 18, 2006
Project Result
A Requirement for Linear and Relational Navigation within the SCO

Linear Navigation
next, previous, first, last

Linear
• location 1
• location 2
• location 3
• location 4

Relational Navigation
help, browse, related content

Relational

relation L1 L3
relation L2 L4
Project Result
Implications of State and Navigation?

LMS – SSN and maybe State Persistence

Current SCORM

Black Box SCO
Black Box SCO
Black Box SCO

navigation and state services at location level

special requirements at SCO-boundary locations

We’re seeing this

SCO
SCO
SCO

ADL International Plugfest
January 18, 2006
Prototype an end-to-end user-centric solution for the SCORM content lifecycle

Technically improve and base SCORE prototype capabilities on an open XML specification

Use the technical and design expertise of the MSU InSite Studio Team to improve usability and support for assessments

Evaluate the Joint ADL Co-Lab's Open Platform for e-Learning (OPEL) and use in prototype if feasible
Implementation Approach

Second Problem

SCO Aggregation and Sequencing

First Problem

SCO Conversion and Editing (SCORE Prototype)

1 SCO Delivery Format (SCORM PIF)

1 SCO XML Exchange Format

Multiple SCOs Delivery Format (SCORM PIF)
• MSU InSite Studio Team

- Brings expertise from InSite Studio
  - User-centric authoring tool
  - Validated by teacher authors
  - Support for SCORM content
  - Support for IMS QTI assessments
- Will provide
  - User interface design expertise
  - Interactions based on IMS QTI
A Few Words about OPEL (Open Platform for E-Learning)

- Under development by Joint ADL Co-Lab
- Technical lead is Chris Bray
- Intended to provide core functionality for ADL prototype projects so that the projects do not need to re-invent it
One Possible OPEL Use

OPEL is being evaluated as a framework to provide user interface support
The Role of an Open Specification

- **Goal**
  - a SCO content model based on a general purpose open specification

- **Why?**
  - Jump start critical mass adoption
  - Leverage platform adoption broader than learning
  - Provide a non-proprietary basis for SCORM content

- **Requirements include**
  - the “card deck” metaphor
  - linear and relational navigation
  - UI styling
  - event triggered actions
  - state variables
  - generic content elements
  - learning-specific content elements

ADL International Plugfest
January 18, 2006
Open Specification Candidates

- OASIS DITA
  - Readable help documentation
  - Type/specialization & relational browsing
- OASIS Open Document Format (ODF)
  - ISO standards track “office document” format
  - Includes several W3C content formats
- Rich Internet Application/UI languages
  - Mozilla XUL, Macromedia MXML, Microsoft XAML
  - Associated with particular platforms
- W3C Compound Document WG
  - Issues of integrating XML content specifications
- Resource Aggregation Specifications
  - MPEG 21.1
  - IMS Content Packaging

ADL International Plugfest
January 18, 2006
Decision to use OASIS ODF

- Open specification
- Incorporates W3C specifications
- On ISO/IEC standards path
- Significant platform adoption demonstrates viability (Open Office, IBM Workplace)
- ODF profile (with extensions) can meet project requirements
- Open source library (AODL) available
High Level Content Model

Each SCO “Black Box”

SCO

contains

1 or more Locations

Location

Location

Location

contains

1 or more Content Components

Component

Component

Component

Component

ODF “primitive”

Extended (e.g. “learning”)

ADL International Plugfest
January 18, 2006
All Components Have

**ODF Content Component**

- **Common Properties**
  - Height, Width
  - XY coordinate
  - Type

- **Common Events**
  - DOM event
  - JavaScript Handlers
  - Use optional

- **Delivery Format Generator**
  - XML to delivery format
  - Default dhtml

- **Editor**
  - means to visualize ODF component
  - means to edit an ODF component

The component approach also supports a modular approach to implementing content converters
Proposed Strategy for ODF Content Components

Location may contain:

- A location contains content components
- All content components share common capabilities and are cleanly encapsulated (can be inserted, cut and pasted as wholes)
- Primitive content components are directly taken from ODF.
- Extended content components may be largely based on ODF (e.g. a component for S1000D elements)
- Extended content components may use an ODF wrapper and be largely based on other schemas (e.g. others listed)
Conclusion

- Problem Area
  - Requirements for representing content below the SCO level
  - Implications for effective content reuse
- 2004-5 SCORE Prototype demonstrated …
  - Content conversion from multiple sources and formats into SCOs
  - The benefits of an XML-based approach to achieve pedagogical and user interface consistency across SCOs
  - A repeatable methodology for content conversion
- 2005-6 XML-based SCORM Studio (in progress) will add …
  - Expanded support for the SCORM content lifecycle
  - Improved usability
  - Improved implementation (leverage open source libraries)
  - XML content representation for the SCO based on OASIS ODF