

平成 18 年度
学士学位論文

SCORM コンテンツ作成支援システムの 構築

Development of SCORM Contents Making Support
System Focused on Content Aggregation Model

1070401 寒川 剛志

指導教員 妻鳥 貴彦

2007 年 2 月 16 日

高知工科大学 情報システム工学科

要 旨

SCORM コンテンツ作成支援システムの構築

寒川 剛志

近年，情報技術が発展し教育分野にも情報技術を利用するケースが増えている．情報技術を利用して教育を行うことを e-Learning とよぶが，企業や大学で用いられるのは WBT (Web Based Training) と呼ばれる Web ベースの e-Learning が主流である．しかし，従来の WBT のシステムでは，システム，教材同士に相互運用性がなく再利用性にも欠けていた．WBT システムにこれらの性質を与えるために，米国の ADL (Advanced Distributed Learning) はコンテンツに SCORM という標準規格を提唱している．SCORM を用いることによって WBT システムは再利用・相互運用可能となった．しかし，SCORM を使用したコンテンツは SCORM への適用を行うためにコンテンツアグリゲーションモデルとランタイム環境と呼ばれる SCORM の構成要素を記述する必要があり，従来のコンテンツに比べコンテンツ作成コストが増加している．そのため，コンテンツの内容に関してコンテンツ作成者が十分に専念できない可能性がある．

そこで本研究では SCORM に規定されている事項のうち，特にコンテンツアグリゲーションモデルに関する支援を行い，コンテンツアグリゲーションモデルの自動的な生成を通じてコンテンツ作成コストを軽減する SCORM コンテンツ作成支援システムを構築した．また，システムを使用して作成したコンテンツの SCORM 対応 LMS (Learning Management System) 上での正常な動作を確認した．

キーワード e-Learning, WBT, SCORM, コンテンツアグリゲーションモデル

Abstract

Development of SCORM Contents Making Support System Focused on Content Aggregation Model

SAMUKAWA Takeshi

In recent years, WBT(Web Based Training) has been common style in enterprise training and university training. However, existing WBT style has some problems about reusability and interoperability between systems and contents. ADL(Advanced Distributed Learning, U.S.) has been offering SCORM(Sharable Content Reference Model) as standard model in WBT. SCORM aims providing reusability and interoperability to WBT. However, SCORM contents authors need knowledge about SCORM and its structure to make SCORM contents. SCORM contents authors need to describe CAM(Content Aggregation Model) and RTE(Run-time Environment) as well as existing contents. These descriptions make SCORM contents difficult to make.

In this research, we focused on description of CAM in SCORM contents. We developed SCORM Contents Making Support System to support SCORM contents authors with no definite knowledge about SCORM. This system supports describing CAM in SCORM contents semi-automatically. By using this system, authors can easily make SCORM contents. And we confirmed contents made by this system on SCORM compliant LMS(Learning Management System). As a result, contents operated as SCORM contents well.

key words e-Learning, WBT, SCORM, Content Aggregation Model

目次

第 1 章	はじめに	1
第 2 章	研究の目的	2
2.1	背景	2
2.1.1	e-Learning の現状	3
2.2	研究の目的	4
第 3 章	SCORM の問題点	5
3.1	SCORM とは	5
3.2	コンテンツアグリゲーションモデル	7
3.2.1	コンテンツモデル	7
3.2.2	メタデータ	8
3.2.3	コンテンツパッケージング	10
3.3	ランタイム環境	12
3.4	SCORM コンテンツ作成にかかるコスト	12
3.5	SCORM コンテンツ作成の問題点	16
第 4 章	SCORM コンテンツ作成の支援	17
4.1	支援方法	17
4.1.1	CAM に関する支援	17
4.1.2	ランタイム環境に関する支援	18
4.2	システム的设计	19
第 5 章	システムの概要	23
5.1	支援方法の実現	23

目次

5.2	動作	25
第 6 章	まとめ	29
謝辞		30
参考文献		31

目次

3.1	コンテンツアグリゲーション SCORM 仕様書より	9
3.2	コンテンツ構造と資源リスト SCORM 仕様書より	11
3.3	アセットメタデータの例 (一部)	14
3.4	SCO メタデータの例 (一部)	14
3.5	コンテンツメタデータの例 (一部)	15
3.6	マニフェストの例 (一部)	15
3.7	SCORM コンテンツ作成によるコストの増加	16
4.1	CAM の XML ファイルの一部	18
4.2	支援方法	22
5.1	JAXB の概念図	24
5.2	システムの動作画面	25
5.3	教材情報入力後	26
5.4	画像の情報入力画面	27
5.5	詳細情報入力画面	27
5.6	作成終了時	27
5.7	完成したコンテンツ	28

表目次

3.1 SCORM メタデータの必須項目数	9
3.2 メタデータの情報の代表例	10
3.3 10 個の SCO からなるコンテンツの作成コスト	13
4.1 メタデータの必須項目	21

第 1 章

はじめに

近年，e-Learning の重要度は高まっており，多くの企業や大学が e-Learning を利用し教育を行っている．e-Learning とは，情報通信技術を利用して学習を行う手法である．そのなかでも，Web を用いて学習を行う WBT (Web Based Training) が主流になりつつある．WBT では，一般に学習全体の管理を行う LMS (Learning Management System) と学習コンテンツから構成されており，学習者は Web ブラウザなどを利用して学習コンテンツにアクセスすることによって，時間や場所にとらわれない学習が可能である．

しかし，従来の WBT システムは同一のベンダの開発した LMS とコンテンツをセットで使用する必要があり，異なったベンダの LMS とコンテンツは相互運用ができず，コンテンツの再利用が困難であった．この問題を解決するために ADL (Advanced Distributed Learning) [2][3] は e-Learning の標準規格である SCORM (Sharable Content Object Reference Model) [2][3] を提唱している．SCORM ではコンテンツに関する規定であるコンテンツアグリゲーションモデルと，LMS との通信を用いた学習管理に関する規定であるランタイム環境を規定し，従来の WBT システムになかったコンテンツの相互運用と再利用を可能にしている．しかし，SCORM コンテンツの作成には SCORM の知識が必要であり，従来のコンテンツ作成に加えてコンテンツアグリゲーションモデルやランタイム環境の作成が不可欠になる．そのためコンテンツ作成コストが増加すると考えられる．このコストの増加はコンテンツの内容に直接かかわるコストではなく，コンテンツの内容にコンテンツ作成者が専念できない可能性がある．

そこで本研究では，SCORM コンテンツ作成支援システムを構築し，SCORM コンテンツ作成にかかるコストを軽減する．

第 2 章

研究の目的

2.1 背景

近年，情報技術の発展と普及，特にインターネットに代表されるネットワーク技術の著しい発展と普及によって，教育分野にも情報技術が利用されている．従来の教育では学習者と講師が同じ場所にいる必要があり，加えて学習管理には大きなコストがかかった．そこで情報技術を利用し，マルチメディアを用いた学習教材や情報処理による学習管理や教材管理を行われるようになった．このような学習形態は e-Learning とよばれ，多くの企業や大学で利用されている．

e-ラーニング白書 2006 年版 [4] によれば e-Learning は次のように定義されている．

「e-Learning とは，情報技術によるコミュニケーション・ネットワークなどを活用した主体的な学習である．コンテンツは学習目的にしたがって編集され，学習者とコンテンツ提供者との間で必要に応じてインタラクティブ性が確保されている．このインタラクティブ性とは学習者が自らの意思で参加する機会が与えられ，人またはコンピュータから学習を進めていくうえでの適切なインストラクションが適時与えられることを指す．」

e-Learning の特徴としては学習の自由度とインタラクティブ性が挙げられる．学習の自由度とは，いつでもどこでも学習できることであり教材のデジタル化のレベルと関係がある．また，インタラクティブ性とは学習の受講側（学習者）と学習の提供側（講師など）が双方向でのコミュニケーションを行えることを指す．

近年では CBT（Computer Based Training）と呼ばれる CD-ROM などの学習教材を用いた学習が行われてきた．しかし，CBT では学習教材を作成や特に教材の更新コストが大

2.1 背景

きくなりやすく、またインタラクティブ性が低く、成績データや学習進捗状況が把握しづらく学習者とのコミュニケーションがしづらいという問題が存在した。そこで、インターネットなどのネットワークを用いて学習を行う WBT (Web Based Training) が主流になってきている。

2.1.1 e-Learning の現状

WBT はブラウザを用いてインターネットやイントラネットなどのネットワークを介して、学習コンテンツにアクセスし学習を行う e-Learning の手法のひとつである。WBT は CBT の問題点である作成・更新コストの大きさを Web システムを用いることで軽減し、ネットワークを使用することで CBT に比べインタラクティブ性が上昇している。加えて、コンテンツをサーバにアップロードすることで学習の自由度が向上し、学習コンテンツを多くの学習者に配信することを可能にした。WBT は LMS (Learning Management System: 学習管理システム) を利用してよりインタラクティブ性を高めた手法が主流である。LMS は学習者の成績・個人情報や学習の進捗状況など学習管理に必要な情報の記録や学習コンテンツの管理を行うシステムである。また、単純に学習コンテンツをストックするだけの機能のものあり、それらは CMS (Course Management System) と呼ばれることもある。学習コンテンツは学習を行う教材そのものであるが、WBT では LMS との通信を行うための手段や LMS 上でのコンテンツの検索や管理に使用する情報がコンテンツ自体に付与されることが多い。WBT では LMS と学習コンテンツの相互の通信によってインタラクティブな学習が可能となる。

しかし、従来の WBT システムはシステムのベンダによって通信方法や管理方法が独自の仕様で作成されており、同一のベンダが作成した LMS と学習コンテンツをセットで使用する必要があった。つまり、異なったベンダの LMS と学習コンテンツの相互運用、学習コンテンツの再利用が困難であった。そこで、教材本体と LMS と通信を行う機能やコンテンツの情報を分離し、教材本体の変更や切り出しが容易にするコンテンツの標準規格が必要となった。

2.2 研究の目的

現在注目されている標準規格は ADL(Advanced Distributed Learning) が提唱する SCORM(Sharable Content Object Reference Model) である。SCORM は ADL とその他の団体が SCORM 以前に行ってきた標準化活動の成果を統合した規格である。SCORM によって学習コンテンツに再利用性と相互運用性などが与えられ、学習の幅が広がったといえる。

SCORM コンテンツでは従来の学習コンテンツでコンテンツに統合されていた LMS との通信やコンテンツの管理情報を統一の様式で記述している。これは従来のコンテンツ作成に加えて、それらの情報を別途記述する必要が生じたことを示している。したがって、SCORM コンテンツの作成は従来のコンテンツ作成に比べて作成コストが増加している。これらの要素は学習の教育的な効果に関しては直接関係ないものであり、それらの作成にコストが大きくかかることはコンテンツの教育的効果の低下につながる可能性があり問題である。

2.2 研究の目的

本研究では、前節に述べた SCORM 作成時に発生する学習に直接関連のないコストを軽減し、コンテンツ作成者が教材の中身に集中して作成できる SCORM コンテンツ作成支援システムを構築する。コスト軽減の手法としては、SCORM の仕様から LMS との通信やコンテンツの情報にあたる規定を抽出し、それらの自動的な生成を行うことが考えられる。そこで本研究ではそれらの要素のひとつでコンテンツに関する規定であるコンテンツアグリゲーションモデルに関する支援を行う。

第 3 章

SCORM の問題点

SCORM によって WBT システムには再利用性や相互運用性を持つことができる。しかし、SCORM コンテンツ作成時に発生するコストに関しては問題がある。本章では SCORM 仕様書から SCORM の構成要素について述べ、SCORM の問題点を述べる。

3.1 SCORM とは

SCORM は ADL が提唱した WBT の標準規格である。WBT の規格は IMS (IMS Global Learning Consortium,) , AICC (the Aviation Industry CBT Committee) ARI-ADNE12(Alliance of Remote Instructional Authoring & Distribution Networks for Europe), IEEE LTSC2(Institute of Electrical and Electronics Engineers, Learning Technology Standards Committee) などの団体が策定した規格が存在しており、SCORM はそれらの規格を統合したものである。SCORM は SCORM1.2 が策定済みの仕様として存在しており、現在 SCORM2004 が策定中である。

SCORM1.2 の仕様書 [2] によると、SCORM の目的は e-Learning コンテンツに次にあげる性質を与えることである。

- 再利用性
学習コンテンツを複数のアプリケーションやその他のコンテンツに組み込める柔軟性
- アクセス可能性
遠隔から教材コンテンツを探してアクセスし、それを他の多くの場所に配信する機能
- 耐久性

3.1 SCORM とは

再デザイン，再構成，再コーディングすることなく，技術上の変更に対応できる機能

- 相互運用性

ある場所であるツールもしくはプラットフォームを用いて開発した学習コンテンツを取り出し，それを他の場所で異なるツールもしくはプラットフォーム利用する機能

これらのことは次のように言い換えることができる．

- Web ベース LMS が異なったベンダのツールで作成したコンテンツを起動し，そのコンテンツとデータをやりとりできる
- 異なるベンダの Web ベース LMS が同じコンテンツを起動し，実行中に内容のデータをそのコンテンツとデータをやりとりできる
- 複数の Web ベース LMS/環境が，実行可能なコンテンツが格納されている共通レポジトリにアクセスし，そのようなコンテンツを起動できる

これらの目的を達成するために SCORM ではコンテンツアグリゲーションモデル(CAM)とランタイム環境(RTE)を規定している．CAM はコンテンツに関する規定であり，コンテンツの発見，再利用，共有，相互運用可能な資源から教材を構成する方法を提供する．さらに，コンテンツの識別方法，記述方法，コースないしコースの一部への集約方法，LMS レポジトリなどのシステム間の移動方法を定義している．また，RTE は学習管理に関する規定であり，後述する SCO(Sharable Content Object) である学習コンテンツと LMS との通信の方法，コンテンツの起動方法，LMS-コンテンツ間でやり取りされるデータ要素を定義している．

これらによって，SCORM では従来の Web コンテンツで困難であったコンテンツの相互運用や再利用が可能になった．

3.2 コンテンツアグリゲーションモデル

3.2 コンテンツアグリゲーションモデル

コンテンツアグリゲーションモデル (CAM) はコンテンツに関する規定である。CAM は教材設計者やコンテンツの実装者が学習資源を集約して、適切な学習体験を提供するための、教育手法に依存しない手段である [2]。学習コンテンツを作る際には単純な画像や動画などのコンテンツ素材を作成し、それらを複雑なコンテンツとしてまとめあげる、あるいは、集約する過程が存在する。CAM はこの過程を支援するものである。CAM は以下の要素で構成される。

- コンテンツモデル
- メタデータ
- コンテンツパッケージング

3.2.1 コンテンツモデル

コンテンツモデルは低レベルな学習資源を集約し高レベルな学習ユニットを構成するための手法である。コンテンツモデルの構成要素は以下の三つである

- アセット

学習コンテンツの最も単純な形は動画、画像や音声などの単一のファイルである。SCORM ではこのような単一のファイルで構成される学習コンテンツをアセットと呼ぶ。アセットの属性や特徴はアセットメタデータに記述し、これにより学習コンテンツまたはそれらを集めたレポジトリから検索・発見が可能となる。

- SCO

SCO は RTE を使用して LMS と通信を行う機能を有したアセットの集合である。SCORM ではアセットを HTML 形式によって整形し SCO を作成する。また、実際の学習コンテンツは SCO の集まりとして整形されていることが多い。そのため、再利用性の観点から SCO は他の SCO となるべく独立で、小さな規模することが要求され

3.2 コンテンツアグリゲーションモデル

る。つまり、SCORM コンテンツを作成する場合は教育的効果や LMS との通信、再利用性をどの程度与えるかなどを考え SCO の規模を最小にすることが望ましい。加えて、SCO では RTE を使用する必要があり、最小限の API(Application Program Interface) 呼び出しを行う必要がある。それ以外の呼び出しは任意である。アセット同様に SCO の特徴や属性は SCO メタデータに記述される。SCO と SCO メタデータを結びつけるのは 3.2.3 節で述べるコンテンツパッケージングである。

● コンテンツアグリゲーション

コンテンツアグリゲーションはアセットを結合した SCO を集約して作成した学習コンテンツの分類や検索のために使用されるコンテンツ構造である。これらはコンテンツアグリゲーションメタデータに記述されコンテンツメタデータとの対応は、SCO と同じく 3.2.3 節で述べるコンテンツパッケージングで行われる。コンテンツアグリゲーションの例を図 3.1 に示す。

CAM におけるコンテンツモデルは教材本体や後述するメタデータ、コンテンツパッケージングで作成するファイルやコンテンツの概念となる。

3.2.2 メタデータ

メタデータはコンテンツモデル構成要素の属性や特徴を記述するための仕組みである。SCORM では、アセット、SCO、コンテンツアグリゲーションを LMS で検索し共有・再利用するために、これらにメタデータを付与している。また、メタデータでは 3.2.1 節で述べた SCO、アセット、コンテンツアグリゲーションの各要素についてメタデータに記述すべき必須項目と任意項目が存在する。コンテンツモデルの構成要素の必須項目と任意項目の個数を表 3.1 に示す。

メタデータに記述することができる情報の中で代表的なものを仕様書から抜粋して表 3.2 に示す。

これらの情報は実際に記述する際には XML を使用して記述する。SCORM1.2 仕様書

3.2 コンテンツアグリゲーションモデル

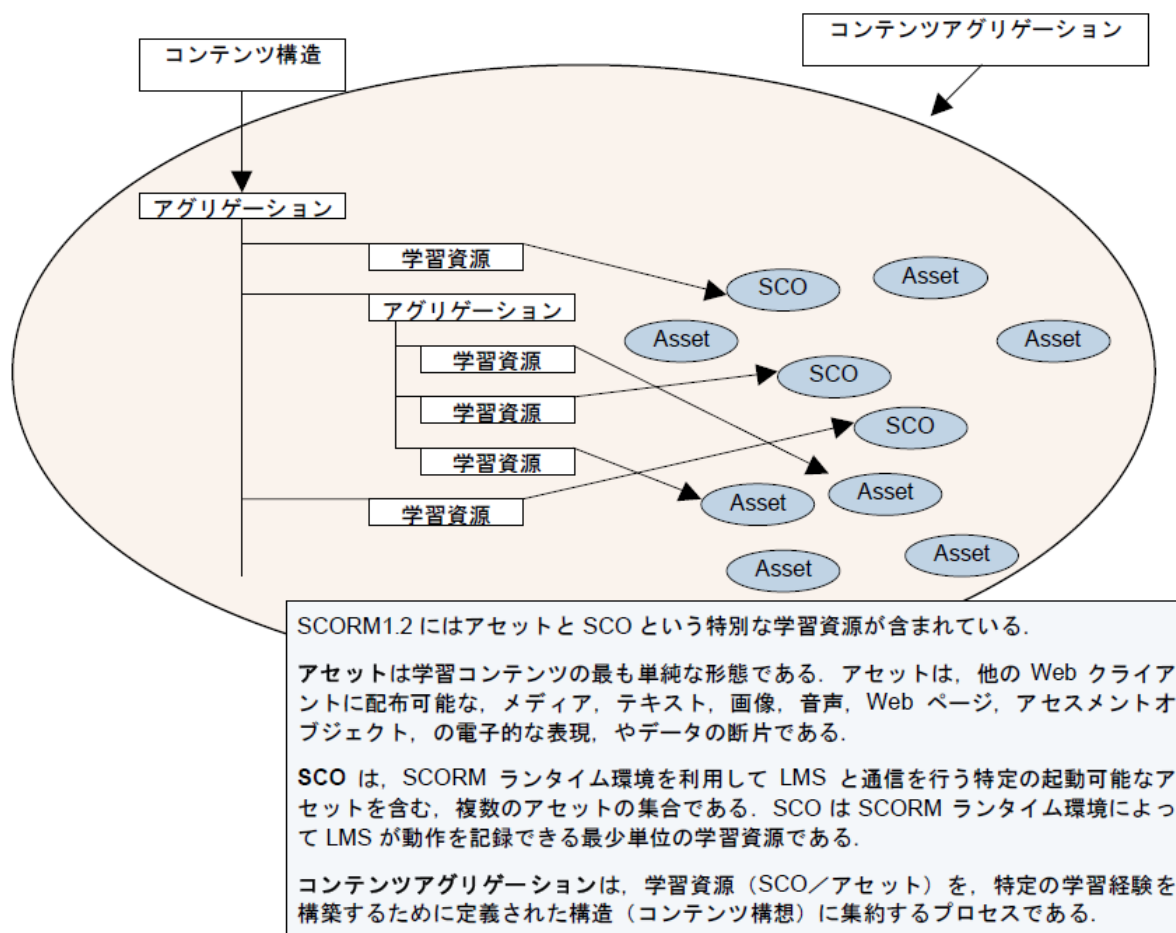


図 3.1 コンテンツアグリゲーション SCORM 仕様書より

表 3.1 SCORM メタデータの必須項目数

名前	必須項目数	任意項目数
アセット	7	49
SCO	15	42
コンテンツアグリゲーション	15	42

3.2 コンテンツアグリゲーションモデル

表 3.2 メタデータの情報の代表例

名称	説明	繰り返し数	データタイプ
Title タイトル	コンテンツに与えられた名前	1 回のみ	言語文字列
Description 内容説明	コンテンツの内容についての文章記述	1 以上	言語文字列
Keyword キーワード	コンテンツの属性や特徴を記述するためのキーワード	0 以上	言語文字列
Difficulty 難易度	コンテンツをユーザが実行するのがどの程度の難しさか	0 または 1	選択 very easy easy medium difficult very difficult

[2] , SCORM2004 仕様書 [3] にはこれらの情報と XML とのバインディングが規定されている .

3.2.3 コンテンツパッケージング

コンテンツパッケージングは学習コンテンツの意図した振る舞いの記述方法 (コンテンツ構造) , および , 異なる環境に移植するために学習資源を 1 つのパッケージにする方法である . コンテンツパッケージングは異なるシステム間でのコンテンツの交換する標準化手法といえる . コンテンツパッケージングはマニフェストファイルと呼ばれるファイルに記述される . マニフェストファイルには以下の内容を記述する .

- コンテンツパッケージそのものに対する記述
- パッケージに関するメタデータ

3.2 コンテンツアグリゲーションモデル

- コンテンツ構造と動作を定義する任意のオーガニゼーション
- パッケージに含まれる資源のリスト

コンテンツの構造は、コンテンツ作成者に対し学習資源の集合をひとつに統合した学習ユニットに編集する方法、および、構造とそれに伴う固有の動作を異なる LMS で再現可能にする方法を提供している。つまり、コンテンツ構造はコンテンツ作成者が提示したいコンテンツの提示順などの意図を LMS に伝える役割がある。

コンテンツ構造の例を図 3.2 に示す。コンテンツパッケージングでは図に示すように、コンテンツ構造と資源リストが関連性を持ち該当する SCO を指し示す。また、資源リストは個々の SCO に含まれるアセットなどの情報も持ちコンテンツに使用されているすべての要素を記述することが可能である。

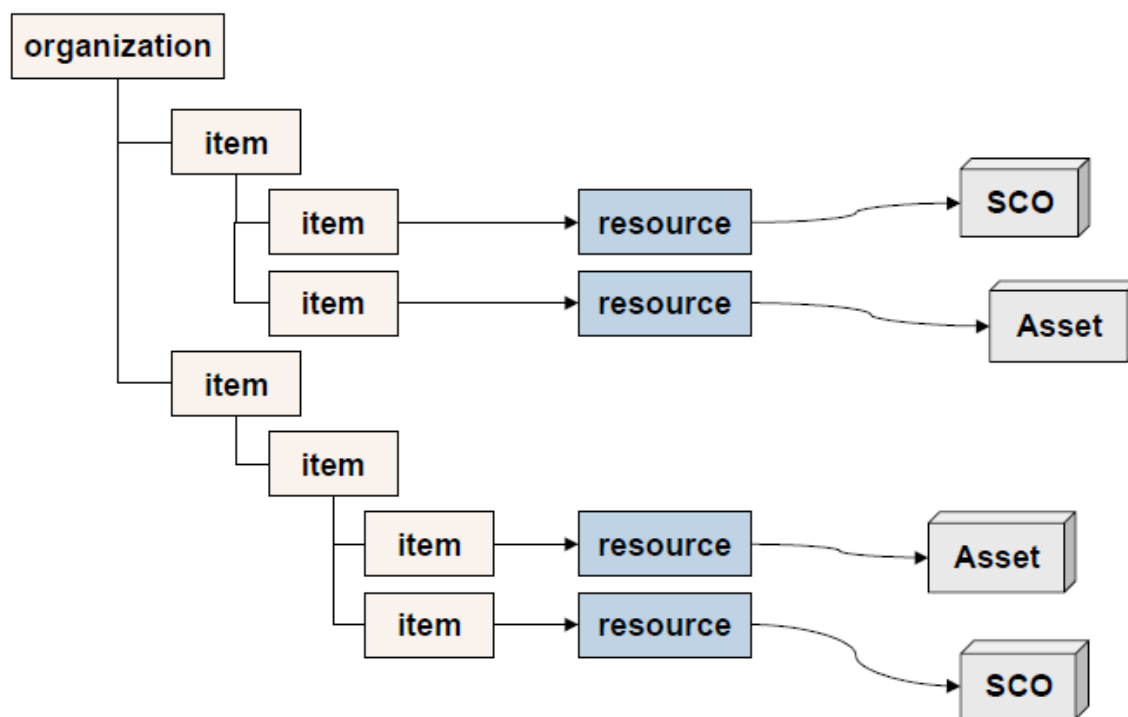


図 3.2 コンテンツ構造と資源リスト SCORM 仕様書より

3.3 ランタイム環境

RTE は学習管理に関する規定で、コンテンツと LMS との標準通信手法を定義するとともに、LMS においてデータを保持する方法やデータの送受信の方法を規定している。RTE を構成する要素は、API とデータモデルである。API には LMS にコンテンツが終了したのか中断したのかなどの進捗状況などのコンテンツの実行状態用関数、LMS-SCO 間で成績データをやり取りするためのデータ転送用関数、エラー用関数があり、LMS-SCO 間のデータ通信に用いられる。SCO では学習の開始と終了を LMS に通知する必要があるため、実行状態用関数を SCO 内で実行する必要がある。データモデルは API のデータ通信用関数で用いられる SCORM で規定された SCORM 対応 LMS に標準で用意されたデータ要素である。データモデルをデータ通信用関数に用いることで LMS から SCO が特定のデータを取得し学習コンテンツに使用したり、逆に SCO から LMS へデータを設定することができる。

また、RTE の API は JavaScript を用いて使用する。したがって、コンテンツに記述する RTE も JavaScript を用いて記述する必要がある。

3.4 SCORM コンテンツ作成にかかるコスト

SCORM コンテンツ作成は従来の WBT コンテンツに加えて以下のようなファイルを作成する必要がある。

1. アセットメタデータ (3.2.2 節)
2. SCO メタデータ (3.2.2 節)
3. 学習コンテンツ全体のメタデータ (3.2.2 節)
4. マニフェストファイル (3.2.3 節)
5. RTE を記述した JavaScript ファイル (3.3 節)

1,2,3 に挙げたメタデータは 3.2.2 節で述べた必須項目を最低限含んでいる必要がある。さらに、3.2.2 節で述べた必須項目を XML で記述する際の SCORM で規定された XML パイ

3.4 SCORM コンテンツ作成にかかるコスト

ンディングに準拠する必要がある。4 に挙げたマニフェストファイルは学習コンテンツの構造と使用したアセット、作成した SCO などの資源リストを含んでいる必要がある。5 に挙げた JavaScript ファイルは SCO の場合、最低でも学習の開始と終了を通知する API を呼び出す必要があり、高度なデータのやり取りを行う場合はデータ通信用関数を利用しコンテンツと RTE が連携する必要がある。

例えば 10 個の SCO (HTML ページ) を持ちそれぞれに画像ファイルが 1 個ある学習コンテンツを作成するために、作成するファイルは表 3.3 のようになる

また、アセットメタデータ、SCO メタデータ、コンテンツメタデータ、マニフェストの例を図 3.3 から図 3.6 に示す。

表 3.3 10 個の SCO からなるコンテンツの作成コスト

種類	個数	備考
アセットメタデータ	10	画像 1 つに対して 1 つ必要
SCO メタデータ	10	SCO であるページ 1 つに対して 1 つ必要
コンテンツメタデータ	1	コンテンツ 1 つに対して 1 つ必要
マニフェストファイル	1	すべてのファイルを資源リストへの記述が必要
JavaScript ファイル	1 以上	コンテンツに応じて可変
HTML ファイル	10	コンテンツの内容
計	33 以上	-

3.4 SCORM コンテンツ作成にかかるコスト

```
<?xml version="1.0"?>
<lom xmlns="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1 imsmd_rootv1p2p1.xsd">
  <general>
    <title>
      <langstring>Vessel Aground Lighting</langstring>
    </title>
    <description>
      <langstring>Picture showing lighting requirments for inland vessel less than 50 meters in
length in an aground condition.</langstring>
    </description>
    <keyword>
      <langstring>Vessel</langstring>
    </keyword>
    <keyword>
      <langstring>Lighting</langstring>
    </keyword>
    <keyword>
      <langstring>Aground</langstring>
    </keyword>
  </general>
```

図 3.3 アセットメタデータの例 (一部)

```
<?xml version="1.0"?>
<lom xmlns="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1 imsmd_rootv1p2p1.xsd">

  <general>
    <title>
      <langstring>Conduct of Vessels in any Condition of Visibility</langstring>
    </title>
    <catalogentry>
      <catalog>ADL Course Catalogue ID</catalog>
      <entry>
        <langstring>1000-01-01</langstring>
      </entry>
    </catalogentry>
    <language>en-US</language>
    <description>
      <langstring>Discusses general rules of operation for vessels on inland waters.
Topics discussed include: Look-out, Safe Speed, Collision, Channels, Traffic Separation.</langstring>
    </description>
    <keyword>
      <langstring>Vessel</langstring>
    </keyword>
```

図 3.4 SCO メタデータの例 (一部)

3.4 SCORM コンテンツ作成にかかるコスト

```
<?xml version="1.0"?>
<lom xmlns="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1 imsmd_rootv1p2p1.xsd">

  <general>
    <title>
      <langstring>Inland Rules of the Road</langstring>
    </title>
    <catalogentry>
      <catalog>ADL Course Catalogue ID</catalog>
      <entry>
        <langstring>1000</langstring>
      </entry>
    </catalogentry>
    <catalogentry>
      <catalog>U.S. Coast Guard, Commandant Instruction</catalog>
      <entry>
        <langstring>1000</langstring>
      </entry>
    </catalogentry>
    <catalogentry>
      <catalog>U.S. Coast Guard, Commandant Instruction</catalog>
      <entry>
```

図 3.5 コンテンツメタデータの例 (一部)

```
<?xml version="1.0"?>
<manifest identifier="SingleCourseManifest" version="1.1"
  xmlns="http://www.imsglobal.org/xsd/imscp_rootv1p1p2"
  xmlns:adlcp="http://www.adlnet.org/xsd/adlcp_rootv1p2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.imsglobal.org/xsd/imscp_rootv1p1p2 imscp_rootv1p1p2.xsd
    http://www.imsglobal.org/xsd/imsmd_rootv1p2p1 imsmd_rootv1p2p1.xsd
    http://www.adlnet.org/xsd/adlcp_rootv1p2 adlcp_rootv1p2.xsd">
  <organizations default="B0">
    <organization identifier="B0">
      <title>Maritime Navigation</title>
      <item identifier="B100" isvisible="true">
        <title>Inland Rules of the Road (HTML Format)</title>
        <item identifier="S100001" identifierref="R_S100001" isvisible="true">
          <title>References and Lesson Objective</title>
        </item>
        <item identifier="B110" isvisible="true">
          <title>Steering & Sailing Rules</title>
          <item identifier="S110001" identifierref="R_S110001">
            <title>Conduct of Vessels in any Condition of Visibility</title>
```

図 3.6 マニフェストの例 (一部)

3.5 SCORM コンテンツ作成の問題点

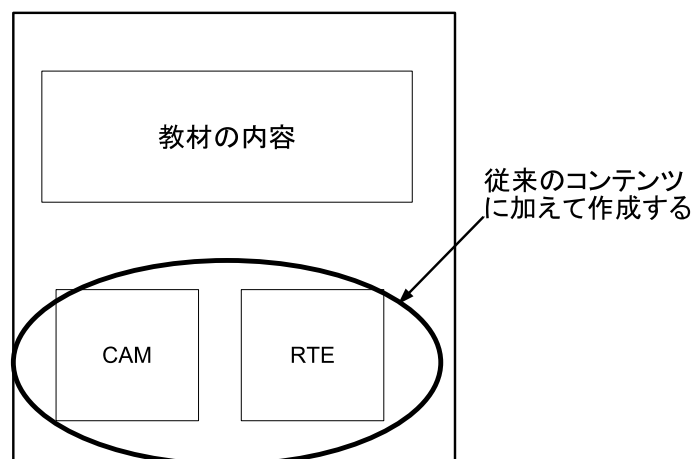


図 3.7 SCORM コンテンツ作成によるコストの増加

3.5 SCORM コンテンツ作成の問題点

一般的に、コンテンツ作成者は教育効果の高いコンテンツの作成に集中すべきであり、CAM や RTE の記述といったコストはできる限り抑えるべきである。しかし、すでに 3.4 節で述べたように SCORM コンテンツの作成は、従来のコンテンツ作成に加えて CAM と RTE を記述する必要があり、CAM と RTE の記述には XML と JavaScript の知識が必要である。SCORM についての知識を持たないコンテンツ作成者が SCORM コンテンツを作成する場合、これらの知識をコンテンツ作成者が持っているとは限らない。また、SCORM の仕様自体も複雑であり、XML バインディングやデータモデルなどを理解することは時間を要する。加えて、表 3.3 のように単純に 10 個の SCO を作成する際にページ以上のファイルを作成する必要が生じている。つまり、SCORM コンテンツの作成は従来のコンテンツ作成に比べて、コンテンツ作成コストが増加している。

このためコンテンツの中身に関してコンテンツ作成者が十分に専念できない可能性があり、SCORM コンテンツを作る際の問題となっている。

第 4 章

SCORM コンテンツ作成の支援

本研究では、3.5 節で挙げた問題点を解決し、SCORM について知識を持たないコンテンツ作成者が SCORM を特に意識せずに、SCORM コンテンツを作成できる SCORM コンテンツ作成支援システムを構築する。

4.1 支援方法

SCORM コンテンツ作成のコストを軽減する方法として、3.4 節で述べた作成すべきファイルの自動生成を行うことが考えられる。この節ではそれぞれのファイルに関して支援方法を述べる。

4.1.1 CAM に関する支援

コンテンツ作成者が CAM を作成する際に記述する情報と XML タグとの対応を把握し、XML を記述する必要がある。本研究の支援ではこれらの対応をシステムが把握し、コンテンツ作成者は情報の入力を行うのみで自動的にシステムが CAM を作成する。

実際の CAM の XML ファイルは図 4.1 のように記述される。この図よりメタデータの XML はタイトルなどのコンテンツ作成者が決定する情報と SCORM で定められた XML タグで構成されている。XML タグは SCORM の仕様で対応する情報ごとにどのタグを使うかは決定されている。

したがって、メタデータを自動的に生成する場合は、コンテンツ作成者から記述する情報に関する入力を行い、入力された情報に対応した XML タグを自動的に判断し XML を作

4.1 支援方法

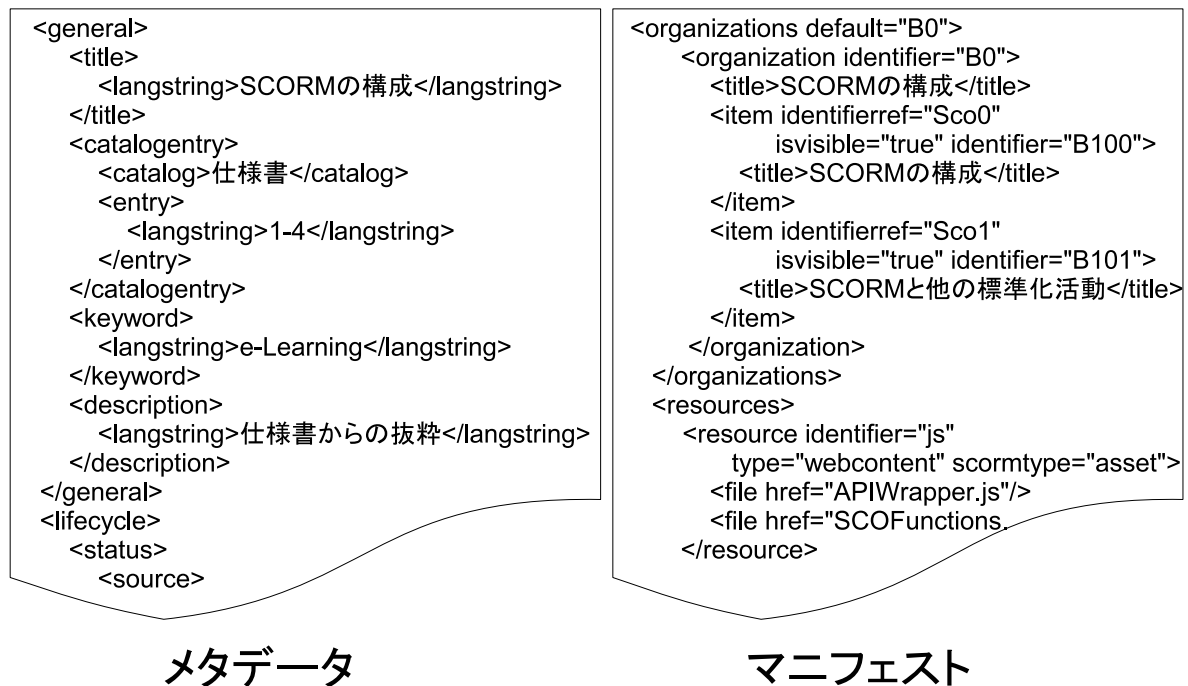


図 4.1 CAM の XML ファイルの一部

成することでメタデータの自動生成が可能である。これによって、コンテンツ作成者はタイトルなどの情報を考えるだけでメタデータを作成できる。

また、マニフェストはコンテンツの構造を表すものであり、コンテンツの作成をシステム上で行う際にコンテンツ構造を自動的に XML にすることが可能である。この支援によって、コンテンツ作成者はコンテンツページの作成を行う際に、マニフェストを意識する必要がなくなる。

4.1.2 ランタイム環境に関する支援

ランタイム環境のデータ通信用関数は性質上コンテンツの内容と密接なかかわりがある。例えば、コンテンツにクイズなどの試験が含まれる場合にそれらの成績データを LMS に送信し、LMS にデータを保存しておくべきである。このとき、どのようなデータモデルに

4.2 システムの設計

データを与えるかを適切に選択する必要がある。データ通信用関数を適切に利用することはコンテンツの内容を踏まえて考えるべきことであり、コンテンツの作成のコストを軽減する本研究では扱わない。

本研究でのランタイム環境に関する支援は、学習の開始時と終了時に進捗状況を通知する関数を使用するようにコンテンツを作成するのみとする。これは SCORM 仕様で定められた SCO の必須通信であり、本研究ではランタイム環境に関しては必要最低限の支援を行う。

具体的には、SCO 内での実行状態用関数の呼び出しを行うよう SCO に記述する。これらの関数は、SCO による学習の開始時に学習開始を LMS に送信する“LMSInitialize()”，学習の終了時に学習終了を通知する“LMSFinish()”である。

4.2 システムの設計

本研究では以下の 3 点の支援を行う。

- メタデータの自動生成

メタデータに入力する情報をコンテンツ作成者が入力するのみでメタデータを生成する

- マニフェストの自動生成

コンテンツの構造を自動的にマニフェスト化し、コンテンツ素材の資源リストを作成する。

- 簡易なコンテンツ作成機能

作成されたメタデータとマニフェストを含む、簡易なコンテンツ作成機能

メタデータ・マニフェスト自動生成では XML タグを入力された情報に対して適切にシステムが付与する。この際に、特にメタデータで記述する情報に必須項目と任意項目が存在しているため、どの程度の XML タグをシステムが扱えるかを考えなければならない。表 4.1 に必須項目を挙げる。ここに挙げた必須項目には少なくとも対応しなければならない。本システムでは、SCORM のメタデータの機能をすべて使用できるように SCORM の XML バインディングに完全に対応し、すべての XML タグの要素をシステムが保持する。また、そ

4.2 システムの設計

れにともない必須項目の入力のチェックを行う。これは、SCORM の準拠したコンテンツを作成するために必要である。任意項目の入力はコンテンツ作成者による選択可能にし、コンテンツ作成者が必要である判断した情報のみメタデータに反映する。

また、マニフェストの自動生成は作成しているコンテンツの構造をコンテンツの作成段階でシステムが完全に自動的に行うことが可能である。コンテンツの作成中にコンテンツ構造が変化し、SCO が追加された場合にコンテンツ構造を変更する。マニフェストではそれぞれの資源に同一のファイル内で重複のない ID を割り振る必要があるが、本システムでは重複のないよう自動で ID を決定する。

コンテンツの作成機能では、HTML を用いた SCO を作成する機能である。この機能は非常に簡易な HTML を出力し、2 つまでの画像ファイルをアセットとして SCO に挿入可能とする。

システムはこれらのファイルと学習の開始と終了などの簡易な機能の JavaScript を SCORM コンテンツとして出力する。

4.2 システムの設計

表 4.1 メタデータの必須項目

項目名	コンテンツアグリゲーション	SCO	アセット
一般			
タイトル	<u>必須</u>	<u>必須</u>	<u>必須</u>
カタログ	<u>必須</u>	<u>必須</u>	任意
エントリ	<u>必須</u>	<u>必須</u>	任意
記述	<u>必須</u>	<u>必須</u>	<u>必須</u>
キーワード	<u>必須</u>	<u>必須</u>	任意
ライフサイクル			
バージョン	<u>必須</u>	<u>必須</u>	任意
状態	<u>必須</u>	<u>必須</u>	任意
メタメタデータ			
メタデータスキーマ	<u>必須</u>	<u>必須</u>	<u>必須</u>
技術的事項			
フォーマット	<u>必須</u>	<u>必須</u>	<u>必須</u>
格納場所	<u>必須</u>	<u>必須</u>	<u>必須</u>
権利			
費用	<u>必須</u>	<u>必須</u>	<u>必須</u>
著作権	<u>必須</u>	<u>必須</u>	<u>必須</u>
分類体系			
目的	<u>必須</u>	<u>必須</u>	任意
記述	<u>必須</u>	<u>必須</u>	任意
キーワード	<u>必須</u>	<u>必須</u>	任意

4.2 システムの設計

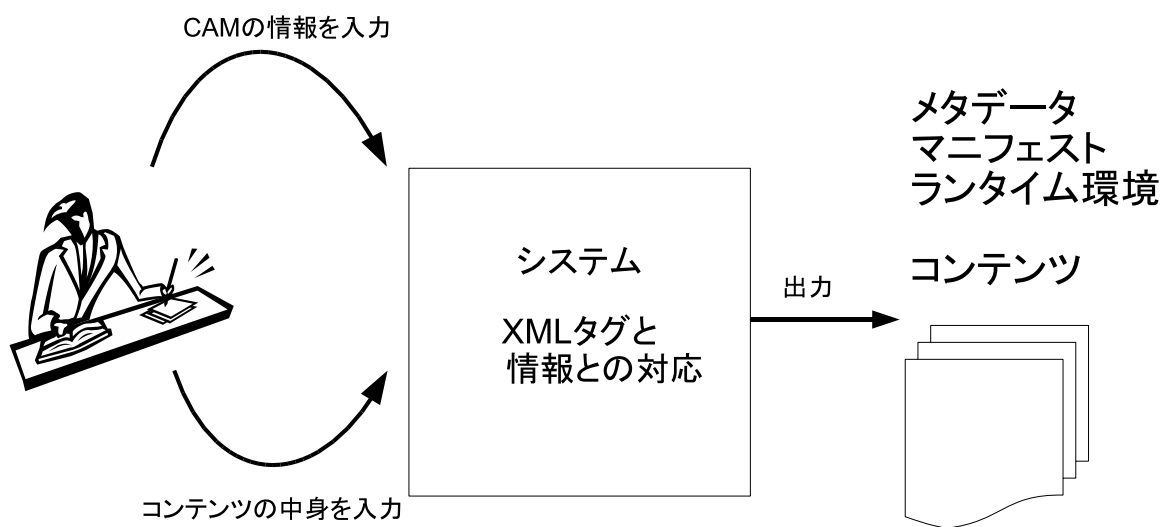


図 4.2 支援方法

第 5 章

システムの概要

5.1 支援方法の実現

本システムは Java を用いて実装したデスクトップアプリケーションである。4.2 節で述べた支援方法を実現するために行った実装について解説する。本システムの実装では JAXB (The Java Architecture for XML Binding) [5] を用いた。JAXB は Sun が開発した Java における XML モデルからオブジェクトモデルへのマッピングツールである。JAXB では図 5.1 に示すように XML のタグ構造を Java のオブジェクト構造に変換する。Java で XML を扱う際にタグ構造をオブジェクトに変換するため、別ファイルの XML テンプレートを用意する必要はない。使用には XML の定義書である DTD(Document Type Definition) または XML Schema を使用することでオブジェクトデータへマッピングできる。また、この際にオブジェクトから XML への出力、XML からオブジェクトへの解析を行う関数を JAXB は自動的に生成する。

本研究では、JAXB を用いて ADL より公開されている SCORM のメタデータ・マニフェスト双方の XML Schema から SCORM の XML タグ構造をオブジェクトデータへマッピングを行い、それを用いて実装を行っている。

JAXB を用いて生成したオブジェクトにデータを渡すために、必須項目、任意項目それぞれの情報入力フィールドを実装し、入力されたデータをオブジェクトに代入している。この際に、必須項目の入力は最初からコンテンツ作成者に表示し、任意項目の入力は別ウィンドウに表示する。

マニフェストの生成には、コンテンツ作成者がページ (SCO) の作成終了ボタンを押した

5.1 支援方法の実現

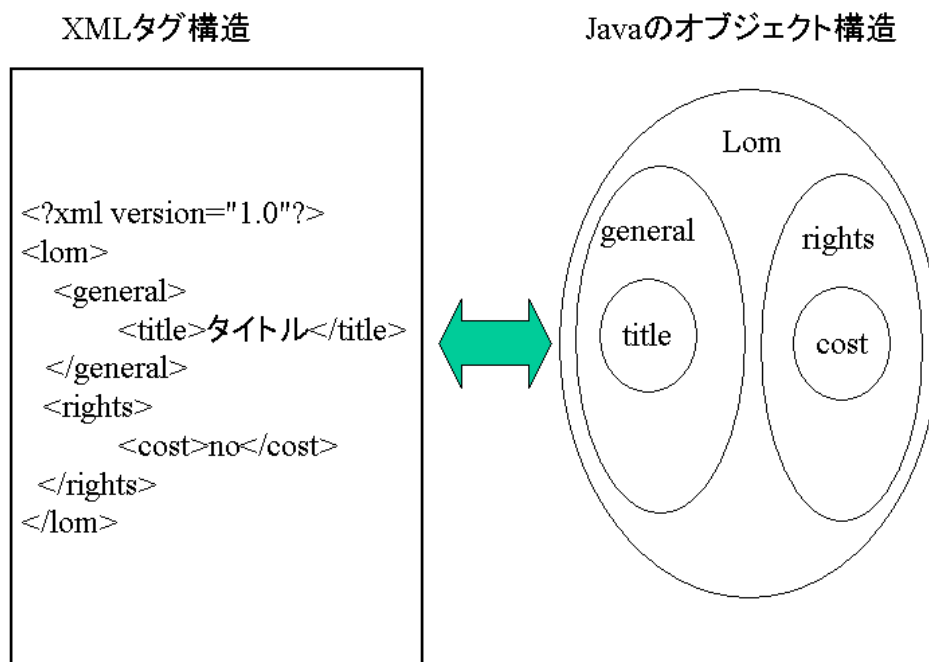


図 5.1 JAXB の概念図

際に JAXB を用いて生成したマニフェストに関するオブジェクトにデータを代入する。また、作成中のコンテンツの構造をコンテンツ作成者が把握しやすいよう、コンテンツ構造をツリー表示する。

簡易なコンテンツ作成機能では、コンテンツの本文とタイトル、教材名をコンテンツ作成者が入力し、それぞれ HTML の本文、メタデータのタイトル項目、コンテンツをまとめたフォルダ名に対応する。画像の挿入は画像のパスを入力することによってシステムがパスから画像のデータを画像用のフォルダにコピーする。

完成したコンテンツはメタデータ、マニフェスト、JavaScript ファイルと共に出力する。

5.2 動作

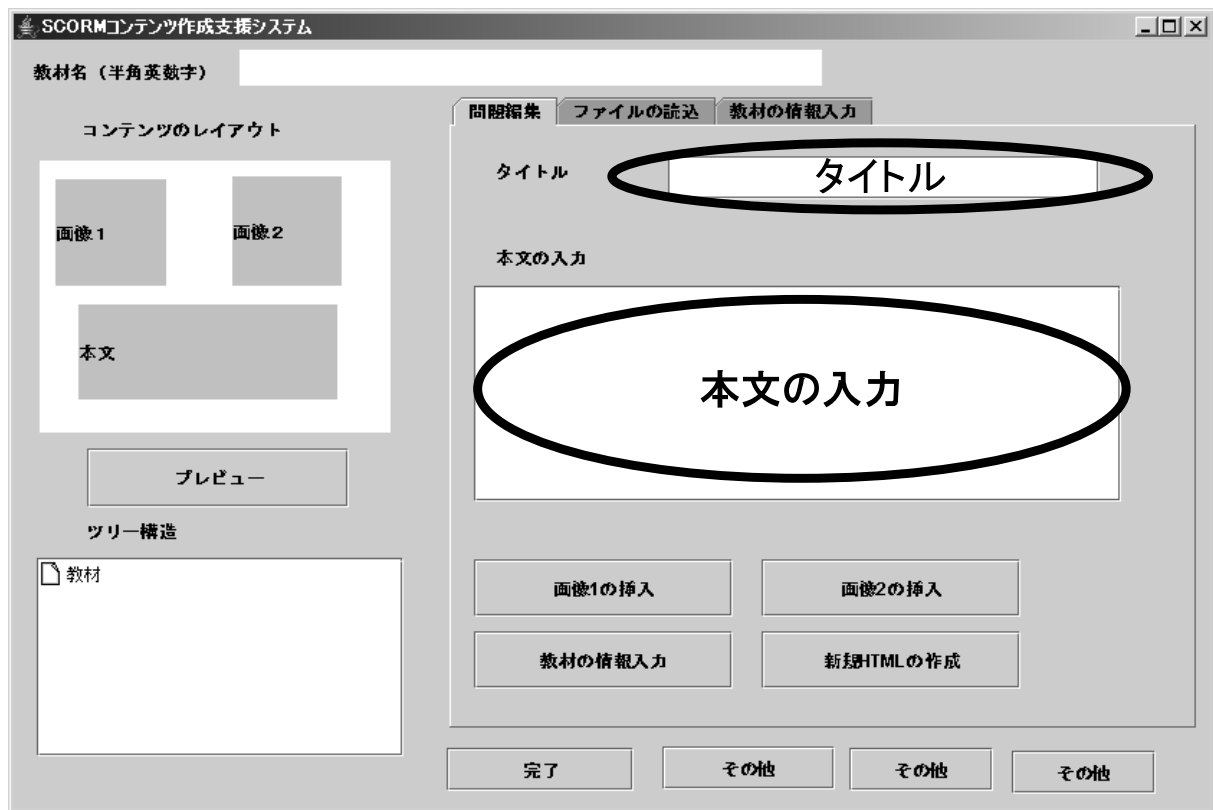


図 5.2 システムの動作画面

5.2 動作

システムを起動すると、図 5.2 のような画面が表示される。この画面には教材名のフィールド、SCO の本文の入力、SCO のタイトル、コンテンツ全体の名前を入力するフィールド、コンテンツのレイアウト表示、コンテンツのツリー構造表示がある。最初の画面にある教材名、タイトル、本文の入力は必須であり、システムが入力をチェックしている。

教材のメタデータを入力する情報は図 5.3 のウィンドウに入力する。ここに表示している項目はカタログ、カタログのエントリ、コンテンツのキーワード、コンテンツの記述などの情報を入力する。

画像の情報入力画面では、コンテンツに挿入する画像のファイルパス、アセットメタデータの入力を行う。この画面にある詳細な情報入力ボタンを押すことによってメタデータの任意項目を入力するウィンドウが表示される。

5.2 動作

The screenshot shows the 'SCORMコンテンツ作成支援システム' (SCORM Content Creation Support System) window. The title bar indicates the course name is 'SCORM1.2Overview'. The '教材の情報入力' (Enter Course Information) tab is active, displaying a form with the following fields and values:

Field	Value
カタログ	仕様書
エントリ	1-4
キーワード	e-Learning
記述	仕様書からの抜粋
費用	必要なし
著作権	あり
教材の状態	最終版
教材の目的	スキルレベル
分類に関する記述	ADL
分類に関するキーワード	ADL

Additional interface elements include a 'コンテンツのレイアウト' (Content Layout) section with a preview of '画面1', '画面2', and '本文', and a 'ツリー構造' (Tree Structure) section showing a folder named '教材'. Buttons for '完了' (Complete), 'その他' (Other), and '詳細な情報の入力' (Enter Detailed Information) are visible at the bottom.

図 5.3 教材情報入力後

詳細な情報の入力画面は図 5.5 のように表示される。ここに表示されている項目は任意入力であり、入力したフィールドをシステムが自動的に判別する。コンテンツ作成者はこのウィンドウの中からコンテンツに付与したい情報を選び入力する。すべてのメタデータの必須項目や任意項目、本文が入力されたら”新規 HTML の作成” ボタンを押し、次の SCO 作成に移行できる。このとき、先に述べたメタデータとマニフェストの XML タグ構造をマップしたオブジェクトに値を代入している。

すべての SCO を作成し、コンテンツが完成した場合、”完了” ボタンを押すことによってメタデータとマニフェストのオブジェクトをファイル化し、SCORM コンテンツを出力する。出力されるディレクトリはプログラムのディレクトリ以下の”work” フォルダ内に入力した教材名のフォルダが作成される。

完成したコンテンツを SCORM 対応 LMS 上で動作させると図 5.7 のようになる。左側と上部のフレームは LMS、中央がコンテンツである。

5.2 動作

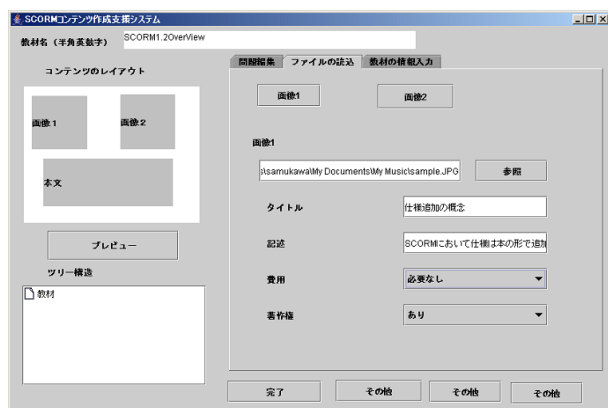


図 5.4 画像の情報入力画面

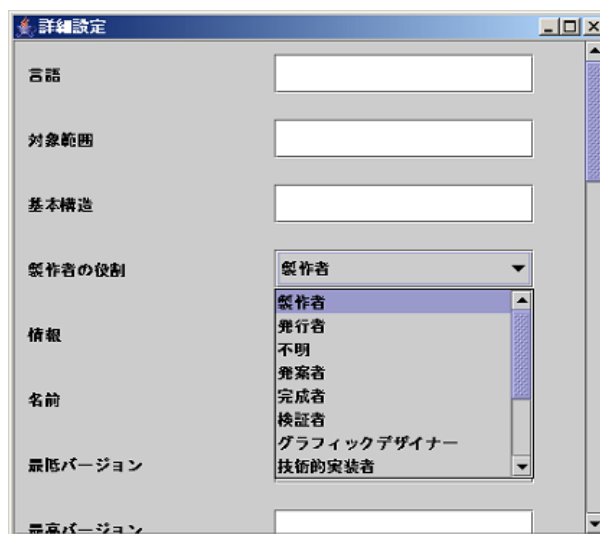


図 5.5 詳細情報入力画面



図 5.6 作成終了時

5.2 動作

図に示すように、SCORMでは、それぞれ個別に参照される技術仕様は「本」として取り扱われる。SCORMの将来バージョンでは新しい仕様の「本」が追加される見込みである。このバージョンでは、SCORMは以下に示す三冊の本に分かれていて、それぞれ前バージョン(SCORM1.1)の第1章、第2章、第3章に対応している。・本1(この文書)は、ADLイニシアチブの概観、SCORMの必然性、以降の章の技術仕様・ガイドラインの概要、である。・本2(SCORMコンテンツアグリゲーションモデル)は、学習資源を構造化された学習コンテンツに集約するためのガイドである。この本では、学習コンテンツに関する用語の説明、SCORMコンテンツパッケージングの説明が行われ、さらに、IMS学習資源メタデータ情報モデルが参照されている。この情報モデルは、IMSおよびARIADNEの共同作業によって開発されたIEEE LTSC学習オブジェクトメタデータ(LOM)仕様に基づいている。これらの仕様が合わさってSCORMコンテンツアグリゲーションモデルを構成している。これらは、図1.1.3aでは本2仕様となっている。・本3(SCORMランタイム環境)はWeb環境で、コンテンツを起動し、通信を行い、動作を記録するためのガイドである。この本はAICC CMI001 相互互換性ガイドラインで定義されたランタイム環境機能に基づくものである。ADLはAICCのメンバー参加者と共同で、起動およびAPI仕様を開発し、AICCデータモデルをWeb環境のデータ項目に適用した。これらは図1.1.3aでは本3仕様となっている。

図 5.7 完成したコンテンツ

第 6 章

まとめ

本研究では、SCORM コンテンツ作成時に発生するコストについて考察し、学習に直接関連のないコストを軽減し、コンテンツ作成者が教材の中身に集中して作成できる SCORM コンテンツ作成支援システムの構築を行った。システムは特にコンテンツアグリゲーションモデルに関する支援を行い、SCORM1.2 で規定される XML バインディングをすべて扱える Java を用いたデスクトップアプリケーションである。また、本システムを使用して作成したコンテンツの SCORM 対応 LMS 上での正常な動作を確認した。

本システムによりコンテンツ作成者が SCORM コンテンツを作成する際に、コンテンツアグリゲーションモデルの XML の記述をすることなく SCORM コンテンツが作成できるようになった。

今後の課題として、現状では CAM に特化した支援のみであるがコンテンツの内容との連携を考えた RTE に関する支援を行うことが考えられる。これは、コンテンツにおいて LMS とのデータ通信を行う状況を考え、どのようなデータを LMS に送信するかを考える必要がある。また、SCORM2004 から新たに登場した学習順序を柔軟に設定できるシーケンシング/ナビゲーションに関しても支援を行えるようシステムを改良する必要がある。

謝辞

本研究の遂行および本論文に関して，多大なるご指導，適切なお助言を頂きました高知工科大学情報システム工学科，妻鳥貴彦講師に心より御礼申し上げます．

ご多忙な中，本研究の副査をお引き受け頂き適切なお助言を頂いた同学科，島村和典教授に心より御礼申し上げます．同じく，ご多忙な中，本研究の副査をお引き受け頂き適切なお助言を頂いた同学科，篠森敬三教授に心より御礼申し上げます．

同研究室修士 1 回生の高木翔平氏，木下聡氏には研究の本質に関して熱く議論を交わし多大なるご助言を頂き，心より感謝致します．そして，SCORM/LMS グループとして共に研究を行った同研究室学部 4 回生大黒隆弘氏に心より感謝申し上げます．また，本研究に関して様々なお協力をしてくださった同研究室 4 回生ならびに 3 回生に心より感謝いたします．特に，同研究室 3 回生の福田将行氏，畠山博和氏にはシステム実装や SCORM の仕様分析において多大なるご協力を頂き，心より感謝いたします．

最後に，この論文を読んでいただいたすべての読者の皆様に感謝申し上げます．

参考文献

- [1] Advanced Distributed Learning ,
<http://www.adlnet.gov/>
- [2] Advanced Distributed Learning and e-Learning Consortium Japan , Sharable
Content Object Reference Model (SCORM) Version 1.2 日本語版 ,
<http://www.elc.or.jp/cgi-bin/csvmail/kigyouscormdownload.html> , 2002
- [3] Advanced Distributed Learning, “Sharable Content Object Reference Model 2004
3rd Edition”, <http://www.adlnet.gov/scorm/index.cfm> , 2006
- [4] 経済産業省監修 日本 e ラーニングコンソーシアム (ELC) 編著 ,
“e ラーニング白書 2006/2007 年版” , 東京電機大学出版局 , 2006 年
- [5] Sun Microsystems, “The Java Architecture for XML Binding(JAXB)”,
<http://java.sun.com/webservices/jaxb/>