

## e ラーニングのための教材データベースの設計

庄司 成臣<sup>\*1</sup> 小山 嘉紀<sup>\*1</sup> 三宅 新二<sup>\*1</sup> 延原 哲也<sup>\*1</sup> 劉 渤江<sup>\*2</sup> 横田 一正<sup>\*3</sup>

<sup>\*1</sup> 岡山県立大学大学院 情報系工学研究科 電子情報通信工学専攻 〒719-1197 岡山県総社市窪木 111

<sup>\*2</sup> 岡山理科大学 総合情報学部 〒700-0005 岡山市理大町 1-1

<sup>\*3</sup> 岡山県立大学 情報工学部 情報通信工学科 〒719-1197 岡山県総社市窪木 111

shoji@c.oka-pu.ac.jp , koyama@rrr.gr.jp , shinji@ryobi.co.jp

tetsuya\_nobuhara@mail.benesse.co.jp , liu@mis.ous.ac.jp , yokota@c.oka-pu.ac.jp

**あらまし** 現在、WBT (Web Based Training) においては SCORM (Sharable Content Object Reference Model) を中心とした標準化が進められている。SCORM は、教材コンテンツの流通、再利用のための規格であり、SCORM をベースにした様々な拡張が可能となっている。本論文では、SCORM の教材コンテンツを集約するモデルであるコンテンツアグリゲーションモデルから、既存の教材表示システムのための教材を生成する方法について考察する。また、SCORM では定められていない教材の個人化について、必要な機能と、実現のためのアプローチを考察する。

**キーワード:** e ラーニング , SCORM , 個人化

## Design of a Teaching-Material Database for e-Learning

Nariomi SHOJI<sup>\*1</sup> Yoshinori KOYAMA<sup>\*1</sup> Shinji MIYAKE<sup>\*1</sup>  
Tetsuya NOBUHARA<sup>\*1</sup> Bojiang LIU<sup>\*2</sup> Kazumasa YOKOTA<sup>\*3</sup>

<sup>\*1</sup>Okayama Prefectural University Graduate School of Systems Engineering

111, Kuboki, Soja, Okayama, 719-1197 Japan

<sup>\*2</sup>Okayama University of Science, Faculty of Informatics

1-1, Ridai-cho, Okayama, 700-0005 Japan

<sup>\*3</sup>Okayama Prefectural University, Faculty of Computer Science and System Engineering

111, Kuboki, Soja, Okayama, 719-1197 Japan

shoji@c.oka-pu.ac.jp , koyama@rrr.gr.jp , shinji@ryobi.co.jp

tetsuya\_nobuhara@mail.benesse.co.jp , liu@mis.ous.ac.jp , yokota@c.oka-pu.ac.jp

**Abstract** The standardization of WBT (Web Based Training) has been developed based on SCORM (Sharable Content Object Reference Model) which is a standard for circulation and reuse of teaching-materials. As SCORM is a kind of standard protocol various extensions are possible. In this paper, we consider how to generate teaching-materials for existing systems from SCORM content aggregation model. Moreover, we consider the approach for personalization of the teaching-materials which are not defined in SCORM.

**Key words:** e-Learning , SCORM , Personalize

## 1 はじめに

近年、ネットワークの高速化などに伴い、Web を利用した教育に注目が集まっている。Web を利用した教育の特徴として、場所と時間の制約を受けないことがあり、そのようなメリットから資格の取得や社員教育などさまざまな分野で活用されている。

Web を利用した教育は同期型と非同期型の学習に分けることができる。同期型には、講義映像をネットワークを通じてリアルタイムに配信する形態や、テレビ会議システムなどを利用して教師と学習者がコミュニケーションをとる形態などがある。どちらの場合も、学習者と教師が授業を行う時間を設定する必要があり、時間的制約が強いと言えるだろう。

非同期型の教育は WBT (Web Based Training) と呼ばれ、教材提供者によって Web サーバ上に置かれた教材を利用して自主学習を行う形態である。同期型の教育と比較すると場所と時間は自由であるが、一方向の自主学習では学習者の疑問点を解消することが難しいため、メールや掲示板などを利用した学習者、教材作成者間でのコミュニケーションが行われることが多い。WBT では学習者側のクライアントとして一般的な Web ブラウザが用いられる。

WBT は現在、SCORM (Sharable Content Object Reference Model)[1][2] 規格を中心とした標準化が行われている。SCORM は教材再利用のための規格であり、AICC (Aviation Industry CBT Committee)[3]、IEEE LTSC (IEEE Learning Technology Standards Committee)[4] の CMI (Computer Managed Instruction)[5] 規格や、IEEE LTSC 及び IMS (IMS Global Learning Consortium, inc.)[6] によって策定された LOM (Learning Object Metadata)[7] 規格をベースに ADL (Advanced Distributed Learning Initiative)[8] によって立案され改良が続けられている。現在の最新バージョンは SCORM 2004 である。

SCORM の主な規格内容としては、LMS (Learning Management System の略で教材の配信や学習履歴の管理などを行う) サーバに読み込まれるコース構造のデータモデル及び XML バインディングの規定や、クライアント側で実行されるコンテンツである SCO (Sharable Content Object) と LMS との間の通信 API と、通信時のデータフォーマットの規定を行っている。

SCORM に準拠した LMS であれば、異なるコンテンツベンダによって作成された SCORM 準拠教材を実行することができる。このため、同じ内容の教材をそれぞれの LMS にあわせたフォーマットで作成するといった必要が無くなる。

また、SCORM では、LMS や SCO の仕様は定められていないので、ベンダーが自由に機能を追加する事ができるという特徴を持つ。

本論文では SCORM 準拠の教材から従来の教育システム用の教材を生成する手法と、個人化機能の実現方法について考察する。

SCORM については次章で概要を述べる。

## 2 SCORM の概要

本章では SCORM の概要と、本論文で扱う問題点を述べる。

### 2.1 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は、学習資源をどのような順序で表示するかを規定する方法を示している。コンテンツアグリゲーションモデルは、アセット、SCO、コンテンツアグリゲーションの3つの構成要素から成り立っている。図1にコンテンツアグリゲーションの例を示す。

**アセット** アセットは学習資源の最小単位であり、具体的にはテキスト、XML、画像、音声などである。SCORM 準拠の LMS では、アセットの起動は行われない。

**SCO** SCO はひとつ以上のアセットの集合から構成される起動可能な学習資源の最小単位である。SCO は SCORM 規格で定められた通信用 API を用いて、学習時間や得点などを LMS とやりとりする。再利用を可能とするために、SCO は他の SCO に依存せず、ひとつの完結した内容を持つ必要がある。

**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、SCO やアセットを組み合わせてコースや章などの学習ユニットとして集約する方法である。コンテンツアグリゲーションは任意の深さで入れ子にすることができる。

アセット、SCO、コンテンツアグリゲーションの属性や特徴はメタデータとして記述される。メタデータとそれぞれの学習資源は、コンテンツパッケージという仕組みにより結びつけられる。

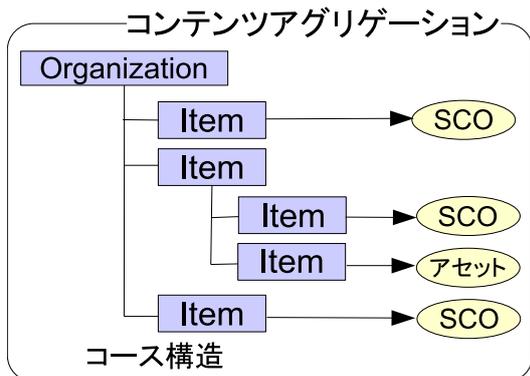


図 1: コンテンツアグリゲーションの例

## 2.2 シーケンシング

シーケンシングは SCORM のバージョン 1.3 で取り入れられた規格であり、学習アクティビティを提示する際の順序の制御や、学習の進捗状況やテスト結果によって次にどのアクティビティを提示するかといったルールを記述する方法を定めている。

シーケンシングで定められている用語について以下で概要を述べる。

### 学習アクティビティ

学習アクティビティは、ひとつの意味のある学習単位であり、SCO やアセット、もしくは複数の子アクティビティから構成される。

### シーケンシングコントロールモード

学習アクティビティ間の移動を制御する方法として、シーケンシングコントロールモードがある。シーケンシングコントロールモードでは前後のアクティビティ間の移動が許される Flow、次のアクティビティのみに移動できる Forward Only、制限をかけない Choice の 3 つのルールを指定することができる。

### シーケンシングルール

シーケンシングでは図 2 に示したように、ある条件とそれが満足された場合の動作というルールベースのモデルが採用されている。制御の条件となるのは、Objective (学習目標) と呼ばれるアクティビティが試行される際に一時的に保持される値と、学習の進捗状況である。子アクティビティの状態によって条件の判定を行うロールアップルールも記述することができる。

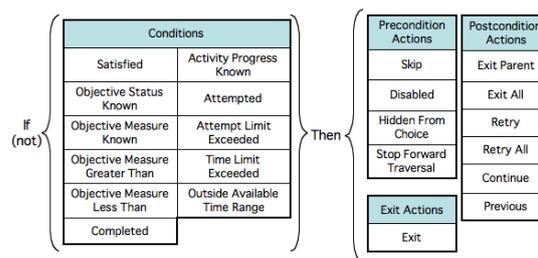


図 2: 条件とアクション

## 2.3 コンテンツパッケージ

SCORM では異なるシステムやツール間で学習資源を交換するための方法として、コンテンツパッケージが定義されている。図 3 にコンテンツパッケージの構造を示す。

具体的な内容として、コンテンツアグリゲーションの構造などをマニフェストと呼ばれるファイルに記述する方法や、学習資源を CD-ROM もしくは zip ファイルにパッケージングするための方法を定めている。

マニフェストは、XML で記述され次の情報を持つ。

- パッケージ自体のメタデータ
- コンテンツアグリゲーションとシーケンシングの構造 (Organizations)
- 学習資源への参照のリスト (Resources)

マニフェストでは、Resource で学習資源とメタデータの関連付けを行い、学習資源に一意的な ID を与える。Organization にコンテンツアグリゲーションの構造を記述し、Item の入れ子で章や節などの関係を表現する。学習資源の参照は、各 Item から Resources 内の ID を指定する事で行われる。シーケンシングは各 Item の子要素として記述されている。

また、マニフェストは複数のサブマニフェストを持つ事ができる。

## 2.4 SCORM の問題点

SCORM で定められているのはコース構造の記述や、SCO と LMS の通信の部分であり、それ以外の機能は、開発者が追加することができる。SCORM に追加する機能としては以下のものが挙げられる。

- 学習オブジェクトからの教材生成
- 教材の個人化機能
- 学習者の理解度の判定

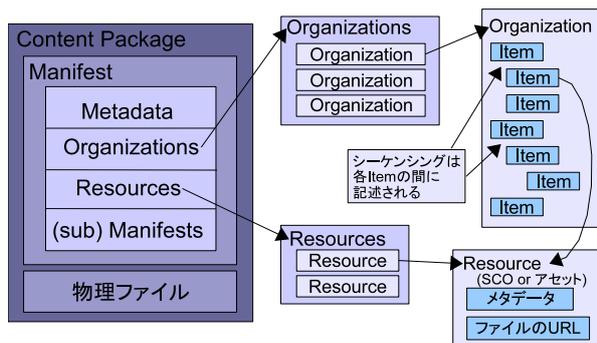


図 3: コンテンツパッケージの構造

このうち、学習者の理解度の判定は、[9]で考察している。本論文では学習オブジェクトからの教材生成と教材の個人化機能について考察する。

### 2.4.1 学習オブジェクトと教材生成

SCORM の仕様では、学習資源を提示する順番を指定する方法は定められているが、それぞれの学習資源の具体的な表示方法に関しては一切定められていない。このため、コンテンツアグリゲーションからの表示を行うルールを決めることができれば、SCORM 準拠のコンテンツをこれまで独自に開発されてきた教育システムで表示することができるだろう。SCORM は学習資源の流通、共通化のための規格であるので、SCORM 準拠教材を表示することができれば、提供できるコンテンツの選択肢が増えるというメリットがある。

教材の生成に関しては次章で詳しく述べる。

### 2.4.2 教材の個人化

学習者が教材を利用する上で、重要だと思われる箇所に線を引くといった個人化の機能が必要となるだろう。学習者へ個人化の機能を提供するためには、個人化の行われた箇所を個人化情報として記録し、それを教材へ反映する機構が必要となる。

教材の個人化は 4 章で詳しく述べる。

## 3 教材の生成

本章では、コンテンツアグリゲーションから既存の表示システムへの教材生成について考察する。

### 3.1 コンテンツアグリゲーションからの教材生成

コンテンツアグリゲーションからの教材の生成を行う場合、次に示す段階があると考えられる。

- シーケンシングを考慮しない教材生成

- シーケンシングの内容を表示用のシステムで解釈し反映する教材生成

シーケンシングを考慮しない場合、分岐を持たず教材内の移動は自由に行う事ができる教材が生成される。

シーケンシングを考慮する場合は、表示用のシステム側でシーケンシングの内容を解釈し、シーケンシングの内容が反映された教材を提示する。

本論文ではまずシーケンシングを考慮しない場合の教材の生成を考える。また、教材表示用のシステムの例として、著者の開発した Module BookViewer を用いて考察する。[10]

#### 3.1.1 Module BookViewer の概要

Module BookViewer の表示例を図 4 に示す。

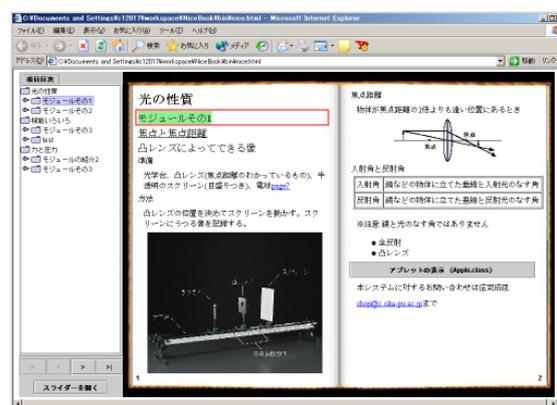


図 4: Module BookViewer の画面

Module BookViewer は、Web ブラウザ上で動作し、テキスト、静止画、動画、3D オブジェクト、Java アプレットを本の形式で表示することができるビューアである。SCO のように、項目ごとの部品化を行うことができ、それぞれの部品を XML で記述した木構造の目次から参照する。部品の記述は、文書記述の汎用 XML フォーマットである XML SmartDoc をベースにしたものとなっている。

学習者には、ナビゲーション用のインターフェースとして、前後のページに移動するためのボタンと、項目間の移動及び教材内の位置の表示を行う目次のツリーが提供される。

#### 3.1.2 目次の生成

コンテンツアグリゲーションは、organization 要素、item 要素からなる木構造になっているため、目次は文書の先頭から行きがけ順で作成すればよいだろう。シーケンシングの記述は、item 要素の子

要素として書かれているが、解釈しないため無視する。また、item 要素は入れ子になることができるので、子要素になっていくほど章、節と小さな項目にする。それぞれの要素は必須属性ではないが、title 属性を持っているので、それを目次の見出しとして利用することができる。

### 3.1.3 表示内容の生成

目次と同様の順番で各 SCO の内容を読み込みファイルの種類にあわせて SmartDoc のタグを生成する。ファイルの種類は、SCO やアセットのメタデータを見ることで判断することができる。具体的には、LOM 規格の format 要素に MIME タイプで記述されているのでそれを参照し、それぞれのファイルの種類にあわせたタグを生成する。複数のアセットから SCO が構成される場合は、アセットが記述されている順に並べる。

SCORM では、ひとつの SCO にどれだけの内容を含むかという教材の粒度に関しては主観的に小さな単位としか決められていない。通常 SCO は通信 API により開始と終了時の処理を行うので SCO ごとに改めて改ページを行い、ひとつのアクティビティの境界を表現する。

コンテンツからページのレイアウトを行う際の問題点として、現状の SCORM 準拠教材の多くが、HTML によって書かれているという事が挙げられる。HTML の問題点として、物理的なレイアウトを含むことや、あいまいな記述が許されているため文書が完全な木構造にならず解析が難しくなるということがある。このためリンクボタンを作成して Web ブラウザの別ウィンドウに表示する。

### 3.1.4 生成例

SCO から教材の部品を生成した例を示す。  
**SCO の例**

```
<resource identifier="S1" type="webcontent"
  adlcp:scormtype="sco" href="text.txt">
  <file href="text.txt">
    <metadata>
      <adlcp:location>
        text1.xml
      </adlcp:location>
    </metadata>
  </file>
  <file href="sample.jpg">
    <metadata>
      <adlcp:location>
        sample.xml
      </adlcp:location>
    </metadata>
```

```
</file>
</resource>
```

### 生成した教材

```
<?xml version="1.0" ?>
<doc>
  <p src="text.txt" />
  
</doc>
```

## 3.2 シーケンシングの表現

シーケンシングを解釈した場合、自由なページ移動だけでなく、提示する教材の制御や学習者の操作の制限が必要となる。本節では、シーケンシングを解釈した場合の実現方法について考察する。

### 3.2.1 Module BookViewer の拡張

現在の Module BookViewer では学習者の操作の制御を行うことは考えられておらず、シーケンシングに対応することができない。このため Module BookViewer を拡張する必要がある。

シーケンシングの内容の解釈の行うために、ビューアの外部にシーケンシング処理部を作成する。シーケンシングの制御は、ビューアと処理部のやりとりによって行う。

ビューアの自体の拡張としては、Objective と学習進捗状況の記録を行う機能が必要である。また、学習者によってページ移動操作が行われた場合に、シーケンシング処理部にページ移動のリクエストを出し、次に移動すべきページを受け取る機能が必要となる。

### 3.2.2 シーケンシング処理部

ビューアとシーケンシング処理部の関係を図5に示す。

シーケンシング処理部は、内部にアクティビティの木構造を持ち、それぞれのアクティビティとビューア内でのページ数との対応情報を持つ。シーケンシング処理部がビューアからのページ移動のリクエストを受けた場合、それがアクティビティ間の移動ならば、その移動に対応する Objective もしくは学習進捗状況によりシーケンシングルールの判定を行い、リクエスト結果をビューアに返す。

## 4 教材の個人化

本章では、教材の本論文で扱う個人化の定義を行い、個人化の機能として必要なものと、それらを実現するために必要なアーキテクチャについて述べる。

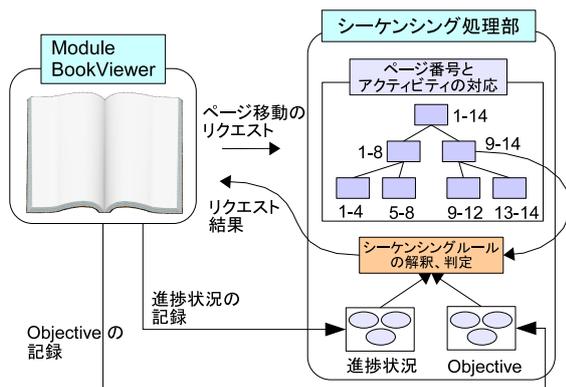


図 5: シーケンシングの処理

## 4.1 個人化の定義

教材の個人化とは、それぞれの学習者の意図や状態にあわせて教材の内容が変更される事であるといえるだろう。

教材の内容の変更といっても、学習者の操作によって行われる場合と、システムが行う場合に分けることができる。

### 4.1.1 学習者が行う個人化

学習者が行う個人化は、学習者の意図によって教材の内容を部分的に変更する行為である。紙の教科書に行う場合と同様にマーカーで線を引いたり、しおりを付けることで、教材を自分の使い易いものに変更する操作にあたる。紙の教科書に対する個人化と異なる点は、個人化を反映するかしないかを自由に切り替えることができるということである。

また、メモとしてのハイパーリンクの張り付けや、メモに対する検索、マーカーを引いた箇所の一覧を自動生成し、見直し学習に利用するといった事も紙の教科書では行えない操作であるといえるだろう。

### 4.1.2 システムが行う個人化

LMS が、学習者の力量にあわせて、表示する教材を変更するといった操作がこれに当てはまる。学習者の力量は、過去の学習履歴やテストの結果から学習者の能力を推定する。教材の内容の変更は、シーケンシングルールによる分岐で実現する場合と、メタデータにより学習者に適した SCO を検索し、学習者に提供する場合が考えられる。これらの操作は、学習者の状態に適した教材を提示するための個人化であるといえる。

利用者の力量を測定し、それにあわせて教材内容を変更する試みは [9] で行っている。本稿では学習

者の行う個人化を対象に考察する。

### 4.1.3 個人化とオーサリングの違い

学習者が行う個人化も、教材作成者が行うオーサリングも教材の内容を変更するという点では同じ行為である。個人化とオーサリングの最も異なる点として、オリジナルの教材の内容を変更を行うかどうかという事がある。個人化の場合は、変更が反映されるのは、自分に提示される教材のみであるので、オリジナルを変更してはいけない。

## 4.2 個人化の種類

本節では個人化の操作の種類を挙げ、必要な機能について述べる。

### 情報の追加

学習者によって、説明の不十分な箇所を補ったり、重要な箇所を強調することで、より理解を深めやすい教材に加工する操作である。

具体的にはメモを貼ることや、重要だと思ふ箇所に線を引くといった操作が当てはまる。

### 情報の削除

冗長な説明を削り、要約表示を行う場合などが当てはまる。削除した部分を再び表示するための操作を行えるようにする必要がある。

### 情報の選択

作成者側から予め、何種類かの選択肢が与えられており、その中から学習者が選ぶ場合である。例えば、用意されている説明の中から、最も理解できる説明を選ぶ操作などが考えられる。教材に自由度を与えることができ、尚且つ教材作成者の意図を反映することができる方法であるといえる。

### 難易度の調整

難易度に関する学習者の意図、例えばある部分の説明が難しいので、もう少し簡単な内容を提示して欲しいという要求をシステムに対して行う操作である。

これらの操作以外にも、コンテンツアグリゲーション内の任意の位置を記録するしおりの機能も必要となるだろう。

## 4.3 個人化実現のアプローチ

前節で述べた個人化機能を実現するためのアプローチについて考察する。

#### 4.3.1 個人化情報

教材の個人化を行う際には、オリジナルの教材の内容を変更してしまうのではなく、外部に個人化情報を持ち、表示する際にそれらが反映されることが望ましい。

また、個人化情報は、内容の更新やレイアウトの変更に対応するために、表示する際の座標といった物理情報に依存してはいけない。

個人化情報が持つべき情報としては以下のものが挙げられる。

- いつ付けられたものか
- どの学習者のものか
- どの部分(SCO、アセット)につけられたものか
- 個人化の種類
- 個人化の内容

#### 4.3.2 個人化情報の適用

XML やテキストファイルがアセットの場合、個人化情報の記述には本研究室で研究を行っている xTrics (XML TRee Information Control Script)[11] の使用を提案する。xTrics は XML を操作するスクリプト言語であり、属性や要素、要素内のテキストに対して変更、挿入、削除といった操作を行うことができる。テキストの一部分を削除する場合は、xTrics でテキストの削除を行う命令を記述し、教材生成の際にスクリプトを実行することで個人化を反映する。変数\$text で表されるテキストの 3 文字目から 5 文字目を削除するスクリプトは以下のように記述する。

```
$text=delete($text,#3:5#);
```

テキストに対してマーカーを引く場合やメモを貼る場合は、マーカーを表すタグを挿入する。しかし、XML にタグを挿入する場合、下記の例のように挿入した<marker>タグが複数のタグにまたがり整形 XML でなくなってしまう恐れがある。

```
<a>
  <b>い
    <marker color="red">
      ろ</b>は<c>に
    </marker>ほ
  </c>
</a>
```

この問題を解決するために、本研究室で提案した XML を意味的に拡張する方式 [12] を導入する。こ

の方式は、XML に特殊な属性を持たせることにより、構文的には XML の整形 XML のルールを満たしつつ、XML より表現能力の高いグラフモデルを表現する方式である。

先ほどの例の場合では、以下のように部分文字列を指定することで整形 XML を崩すことなくマーキングを行うことができる。

```
<a q:id="a"
  xmlns:q="http://alpha.c.oka-pu.ac.jp/Qpit">
  いろはにほ
  <b q:substring="a(0,2)" />
  <c q:substring="a(3,5)" />
  <marker q:substring="a(1,4)"
    color="red"/>
</a>
```

#### 4.4 個人化情報の共有

教材作成者にとって、学習者が行った個人化の位置と内容の情報は教材分析の資料になるといえる。例えば、ほとんどの学習者によってメモが付けられた部分を知る事で、説明の不十分な箇所の発見につながるといった事が考えられる。このことから、教材作成者には教材分析のための機能のひとつとして、個人化の行われている位置や内容を把握できる機能が提供されることが考えられる。

ただし、個人化情報として書かれている内容の全てが教材作成者に公開されてしまつては、学習者のプライバシーが損なわれる恐れがあるため、公開される情報の範囲を明確にする必要があるだろう。

## 5 おわりに

### 5.1 まとめ

本論文では、コンテンツアグリゲーションから既存の表示用システムへの教材の生成方法とシーケンシングを行う場合の教材提示の実現方法について考察を行った。

また、教材の個人化の機能について、必要な機能の分類と実現のためのアプローチについて考察した。

### 5.2 今後の課題

教材の生成に関しては、シーケンシングを考慮した場合においてシーケンシングルールの解釈方法について考察する必要があるだろう。また、今回は Module BookViewer を対象にしたが、他の教材表

示用システムの場合においても考えていく必要があるだろう。

教材の個人化に関しては、メタデータを含め個人化情報の具体的な記述方法と教材への反映方法について考察する必要があるだろう。また個人化情報の共有について共有範囲のポリシーを定める必要があるだろう。

それらの具体的な仕様が決まり次第、システムのプロトタイプの作成を行う。

## 参考文献

- [1] ADL, “SCORM Content Aggregation Model Version 1.3”, 2004.
- [2] ADL, “SCORM Sequencing and Navigation Version 1.3”, 2004.
- [3] AICC, <http://www.aicc.org/>
- [4] IEEE LTSC, <http://ltsc.ieee.org/>
- [5] CMI, <http://ltsc.ieee.org/wg11/>
- [6] IMS, <http://www.imsproject.org/>
- [7] LOM, <http://ltsc.ieee.org/wg12/>
- [8] ADL, <http://www.adlnet.org/>
- [9] 延原 哲也, 小山 嘉紀, 三宅 新二, 庄司 成臣, 劉 渤江, 横田 一正 “学習者の理解度に対応した適応型 e ラーニングシステムの考察”, DBWS2004.
- [10] 庄司 成臣 “Web 用モジュール型教材システムの実現”, 特別研究論文, 2004.  
<http://alpha.c.oka-pu.ac.jp/~shoji/h15/thesis.pdf>
- [11] 岡本 辰夫, 吉田 奈美子, 國島 丈生, 横田 一正, “XML 操作スクリプト言語による個人化手法の提案”, DEWS2002.
- [12] 國島 丈生, 貴志 聰一郎, 三戸 健二, 室山 恵美, 横田 一正, “意味的拡張 XML の処理系の実装と考察”, 電子情報通信学会技術研究報告, DE2001-14, 2001.
- [13] 先進学習基板協議会, e ラーニング白書 2003/2004 年版, オーム社, 2003.