

Dexterモデルに基づいたCAI教材データベースの検討

5Q-1

山北隆典* 谷川 健* 星原健二郎* 富士 隆* 三枝武男**

*学習情報通信システム研究所

**北海道情報大学

1. はじめに

高度個別型CAIの要素技術の一つとして、教材の部品化を可能とする教材のデータモデル、「ハイパーフレーム」¹⁾を提案してきた。ハイパーフレームは教材部品の単位であり、教える内容をマルチメディア表現で分かり易く提示する表現実体オブジェクトと学習者の質問に答えたり学習結果を評価する教材知識オブジェクトを保持しているオブジェクトである。現在、ハイパーフレームの管理を目的としたCAI教材データベースの検討を進めている。

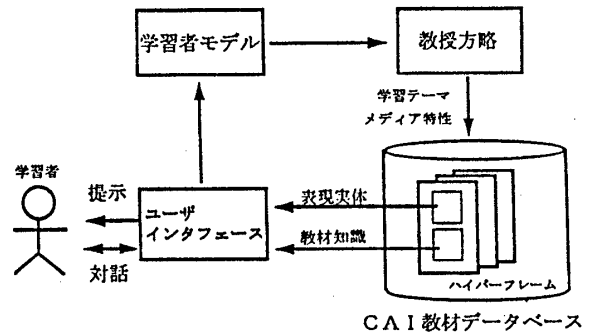


図1 高度個別型CAIの概念図

2. 高度個別型CAIの教材データベース

高度個別型CAIでは、学習者の学習目標をもとに、教材と分離されている教授方略オブジェクトがこのCAI教材データベースにアクセスし、学習テーマの体系に添ったハイパーフレームを選択し、学習者のメディア特性を勘案して適切な表現実体で教える内容を学習者に提示する。また、学習者はそのハイパーフレームの持つ教材知識と対話（質問や回答）したり、他のハイパーフレームを参照したりできる。高度個別型CAIの概念図を図1に示す。図2にはCAI教材データベースのオブジェクト図を示す。

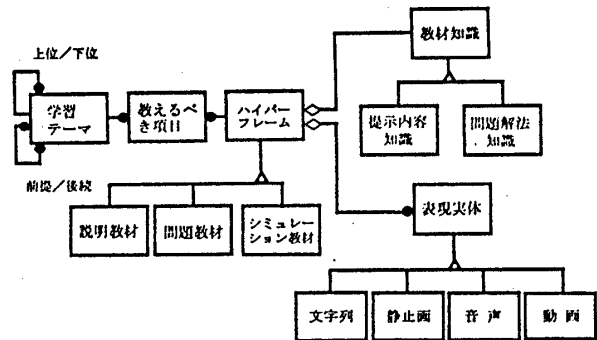


図2 CAI教材データベース

3. Dexterモデルの適用

CAI教材データベースを実現するためには、オブジェクトを個別に扱い、その個々の関連を管理できるハイパーテキスト構造の利用が有効である。そこで、ハイパーテキストの参照モデルであるDexterモデル²⁾を基本モデルとして高度個別型CAIを捉え、CAI教材データベースの検討をはじめた。

3.1 Dexterモデルとの対応

Dexterモデルと高度個別型CAIとの対応はおおよそ図3のようになる。

すなわち、ユーザーインタフェースや教授方略、学習者モデル等はDexterモデルにおけるRuntime Layerに位置付けることができる。CAI教材データベースで管理するオブジェクトはStorage LayerとPresentation Specification、Anchoringの2つのインタフェースにおいてモデル化できる。表現実体のコンテンツである文字列、静止画、音声、動画といったオブジェクトはWithin-Component Layerに位置付けることができる。教材知識は提示する教材に関する知識を意味ネットワークで表現し対象領域の問題解決のための知識をルール表現で保持しておりこれらの知識表現もWithin-Component Layerでモデル化できる。

A CAI Teaching Material Database based on the Dexter Hypertext Reference Model
 Takanori Yamakita* Takeshi Tanigawa* Kenjiro Hoshihara* Takashi Fuji* Takeo Saegusa**
 *Software Research Laboratory
 **Hokkaido Information University

	Dexterモデル	C A I教材データベース
Run-time Layer	アプリケーション (Session, History)	ユーザインタフェース、教授方針、学習者モデル等
Presentation Specification	Componentの視覚的表現指定	表現実体の提示仕様 教材知識との対話仕様
Storage Layer	Atomic Component Composite Component Link Component	ハイパーフレーム、表現実体、教材知識、学習テーマ 上位/下位、前提/後続リンク、意味的包含リンク、参照リンク
Anchoring	Component内の領域指定	表現メディア毎の領域指定
Within-Component Layer	文字列、図形、画像、アニメーション、...	表現実体のコンテンツ 文字列、静止画、音声、動画 教材知識のコンテンツ 意味ネットワーク、プロダクションルール

図3 Dexterモデルと高度個別型C A Iとの対応

3.2 Dexterモデルの問題点

マルチメディアデータをデータベースで管理する場合は、入出力環境に依存しない形式が望ましい。DexterモデルではComponentのコンテンツの提示仕様を提示の実現メカニズムと分離することの重要性をPresentation Specificationというインタフェースで指摘しており、属性としてPresentation Specificationを保持している。C A I教材データベースにおいても表現実体のコンテンツとしてマルチメディアデータを扱うためのAtomic Componentを定義する。ところで、C A I教材データベースでは更に教材知識をコンテンツとするAtomic Componentも定義する必要がある。すなわち、教材知識の対話仕様と対話の実現メカニズムとを分離した形でComponentを管理する必要もあり、この点の拡張が不可欠となる。

3.3 Presentation Specificationの拡張

(1) Atomic Component

表現実体のコンテンツ（マルチメディアデータ）についてはWithin-Component Layerのオブジェクトのデータ型情報(media_type)をPresentation Specificationとして保持する（コンテンツの加工は行わない）。

教材知識との対話形態(dialog_type)として選択式と記述式を扱う。学習者からのリクエストの受付仕様(accept_form)として、選択式の場合は選択肢ラベルとそれを選択した時の戻り値を、記述式の場合は許容文字数や入力文字種別等を質問項目毎や課題毎に保持することでRun-time Layerでの入力画面の提示が可能となる。また、学習者の入力情報をWithin-Component Layerの

教材知識のコンテンツへ伝えるためのデータフォーマット(trans_form)も指定する。

従って、Presentation Specification を

```
presentSpec = < media_type,
                 dialog_type,
                 accept_form,
                 trans_form >
```

とする。

(2) Composite Component

内部に含んでいるComponentの持つコンテンツの提示順番(order)、スクリーン上での提示場所(space)、透過表示か上書き表示かを指定する表示モード(mode)、といった大域的な情報を指定する。

従って、Presentation Specification を

```
presentSpec = < order, space, mode >
```

とする。

4. 実装にむけて

実装に於いては汎用のオブジェクト指向データベースを利用し、データベースサーバ上にStorage LayerとWithin-Component Layerを、クライアントにRun-time Layerを割り当てる。クライアント側にはSession(トランザクション)を管理するオブジェクトとPresentation Specificationを解釈するオブジェクトを実装する。

5. まとめと課題

Dexterモデルを基本モデルとしてC A I教材データベースを検討した。特に教材知識との対話を実現するためのPresentation Specificationの拡張に焦点を当てて報告した。課題として、モデルの実装と評価によってモデルを洗練すること、またC A I教材データベースが分散した環境に於いても機能するためのLink方式やAnchoringの機構に付いての検討がある。

参考文献

- 1) 富士隆他；ハイパーフレームを用いた知的マルチメディアC A Iの開発、1994年情報学シンポジウム講演論文集、pp.133-141 (1994)
- 2) Halasz, F. and Schwartz, M. ; The Dexter Hypertext Reference Model, Comm. of the ACM, Vol. 37, No. 2, pp. 30-39 (1994)