

マルチメディアオブジェクト構成モデルとその実現方式

坂上 秀和 †

濱川 礼 †

暦本 純一 †

†NEC C&C システム研究所

‡NEC ソフトウェア生産技術開発研究所

マルチメディアデータを扱うための、オブジェクト指向に基づいたモデルについて提案する。構成モデルは、メディアオブジェクトを組み合わせて複合マルチメディアオブジェクト構成するときの要素となる全てのオブジェクトについて伸縮属性を持つことを特長とする。また、複数のマルチメディアオブジェクト間の同期関係を、従来のように単純に時間軸上に配置するのではなく、オブジェクト間の制約を基に定義していく。この方式により、柔軟な同期定義や、高い再利用性が達成される。本論文では、構成モデルの内容とこのモデルに基づいて構築したクラスライブラリおよびそれらの応用について述べる。

MULTIMEDIA OBJECT COMPOSITION MODEL
AND ITS IMPLEMENTATION

Hidekazu Sakagami †

Rei Hamakawa †

Jun Rekimoto †

†NEC C&C Systems Research Laboratories

‡NEC Software Engineering Laboratory

The authors propose an object-oriented model for handling multimedia data; it is called composition model. In the composition model, all the element objects of complex multimedia data is assumed to have spatial and temporal elasticity. By introducing "temporal glue" object and various types of composite multimedia objects, it is quite easy to construct and re-use complex multimedia data without specifying precise time-line locations. In this paper, the authors briefly discuss the composition model, as well as our experimental C++ multimedia class library and applications based on the model.

1 はじめに

従来のテキスト／静止画像等のメディアに加えて、オーディオ／ビデオなどのメディアを操作するために、マルチメディアシステムへの期待と重要性が高まっている。マルチメディアを取り扱う上で、定められたメディア間の同期をどのように達成するかという課題と共に、そもそも利用者がどのようにしてメディア情報を組み上げていくか、メディア間の同期をどのように定義していくか、といった問題も重要な研究課題である。従来の [1]、[2] のようなマルチメディアオーサリングシステムでは、メディアオブジェクトを時間軸上に複数のメディアオブジェクトを配置していくことでメディア情報を組み上げて行く方式である（これをタイムラインモデルと呼ぶ）。

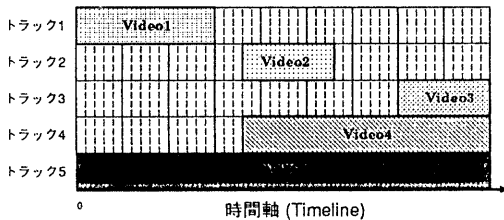


図 1: タイムラインモデル方式

しかし、このモデルは実際に使用してみると、いくつかの問題点があることに気づく。

- 柔軟性がない
組み合わせるメディアオブジェクトの時間管理処理を予めやっておかなければ、その後の組み合わせ作業が容易に行えない。例えば、あるシーンの間だけBGMを繰り返し流しつづける、という単純な複合マルチメディアでも、シーンの長さが変更されると、それに合わせてBGMの長さも変更しなければならない。
- 修正が困難
作成した複合マルチメディアオブジェクトに対し、若干の修正を加えるためにも、関連する全てのデータについて修正しなければならない。また、複数のトラック上にあるメディアデータで、どの位置が本当に同期すべき点なのか、どこはたまたま同じ時間に存在するだけなのか、が明示的に表現されていないため、一部分を修正した場合の修正が及ぶ範囲など

は、利用者の責任で把握していなければならない。

- 部品としての再利用性が低い
部分的に組み上げた複合マルチメディアオブジェクト情報を、部品として保存していくことが困難である。例えばナレーションと字幕をペアにして部品とし、映像と音響効果を組にして部品として、それらの組み合わせで日本語用、英語用のマルチメディアタイトルを構成していくようなことが案にはできない。

筆者らはこれらの問題点を解決するために、タイムラインモデルとは異なった、同期の設定や修正が容易に行えるマルチメディアオブジェクトモデル（これを構成モデルと呼ぶ）を提案する。この構成モデルは、取り扱うすべてのマルチメディアオブジェクトが伸縮属性を持つことを特長とする。また、各メディアオブジェクトを組み合わせるための方式についても、従来方式のようにタイムラインを基準とするのではなく、Boxなどの複合オブジェクトを用いてメディアオブジェクト間の相対的な位置関係を基準に構築する方法をとる。

この伸縮属性の考え方は、TEX[4]やInterViews[3]における2次元平面でのBox & Glueの概念を、時間を加えた3次元の時空間に拡張したものである。伸縮属性の導入によりメディアオブジェクトの(時間的)長さに柔軟性が与えられ、Boxなどの複合オブジェクトを用いて(非タイムラインモデル的に)複合マルチメディアオブジェクトを作成する方式であるため、データの再利用性が高められるなどの特長を持つ。

また構成モデルでは、ビデオやオーディオ等の時間軸を持つメディアオブジェクトのみならず、これらのメディアオブジェクトを複合したときの結合部分に対しても、伸縮属性を持つ仮想的なメディアオブジェクトとしてみなしており(これをtglueと呼ぶ)、複合マルチメディアオブジェクトを作成する際の柔軟性や、作成された複合マルチメディアオブジェクトを修正する際の柔軟性を高めている。

本構成モデルは、ビデオやオーディオ等の時間軸を持つメディアを、他のメディアと共通のクラスに属するオブジェクトとみなしたオブジェクト指向に基づくモデルであり、メディア間の親和性が高い。また、複合オブジェクトを生成するための編集オペレータや、生成した複合オブジェクトを再生するための再生オペレータもオブジェクトとして、本モデルでは保持している。

2 構成モデル

構成モデルは、オーディオやビデオや静止画などの基本メディアオブジェクトと、それらを組み合わせるための複合オブジェクトから構成される。構成モデルでは、オブジェクトの長さに柔軟性を与えるため、全てのメディアオブジェクト(基本・複合)に対して2次元の平面に時間軸を加えた3次元の時空間軸方向について伸縮属性を導入する。

この伸縮属性の基本形式は、

Requirement(natural,stretch,shrink)

であり、各パラメータは、naturalは何も外圧を加えない場合の長さ、stretchは最大伸び量、shrinkは最大縮み量を表している。すなわち、このメディアオブジェクトはnatural-shrinkからnatural+stretchの間の大きさをとりうることになる。また、伸縮不能なオブジェクトについては、stretchやshrinkなどのパラメータの値を0とすることで対応する。

2.1 基本メディアオブジェクト

基本メディアオブジェクトは、従来から計算機の実操作対象であったテキスト、静止画の他に、ビデオデータ、オーディオデータなどのように、時間という概念を持つメディアデータがある。また本構成モデルでは、これらの素材としてのメディアデータの他に、tglueと呼ばれるこれらのメディアデータを組み合わせるときの結合部分も基本メディアオブジェクトの一つとして扱う。現在、以下のような基本メディアオブジェクトがある。

Video 動画。Videoのソースとしては、VCRやLaserDiscなどの外部機器のほか、計算機の記憶装置上に蓄積されたビデオデータを再生する場合などもある。このとき、JPEGやMPEGなどの各種の圧縮技術を用いて蓄積されている場合もある。

Audio 音声データ。Videoと同様にソースとしてはCDなどの外部機器のほか、計算機の記憶装置上に蓄積されたオーディオデータを再生する場合がある。

Still 静止画。静止画などのように、時間軸という概念をもたないメディアオブジェクトについても、構成モデルでは、時間パラメータが0のメディアオブジェクトとして扱う。

tglue メディアオブジェクトを組み合わせる複合マルチメディアオブジェクトを構築するときの、メディアオブジェクト間の結合部分。実体としてはデータを持たないが、仮想的に伸縮属性のみを持つメディア

オブジェクトとして扱う。TeXにおけるglueを時間に関して拡張したものに相当する。

2.2 複合オブジェクト

構成モデルでは、上記の基本メディアオブジェクトを組み合わせる操作を行うオブジェクトとして、複合オブジェクトを用意する。

複合オブジェクトは、複数のオブジェクトのあるレイアウトポリシーに従って時空間上に配置するためのものと、元となるオブジェクトに別の属性を付加するためのものがある。具体的には、現在、以下のような複合オブジェクトがある。

Cut 元となるオブジェクトの指定した座標軸上でのある一部分だけを切り出す。ただし、実体をコピーするのではなく、元のオブジェクトへの参照情報と、範囲情報を持っている。範囲自体も伸縮属性を持っていない。

Loop 子供の内容が所望の長さになるまで何回も繰り返されるオブジェクト。

Box 指定された座標軸に従ってオブジェクト群を(サイズを伸縮属性から比例配分しながら)配置する。TeXのBoxを時間軸方向に拡張した概念に対応している。配置する方向により、TBBox、LRBox、SEBoxなどがあり、それぞれオブジェクトを上下、左右、時間的な前後に配置する。

Space 明示的に時空間上での位置を指定して子供オブジェクトを配置する。従来のタイムラインモデルを本モデルに取り込むためのものである。

Choice 複数の子供のうち、外部から与えられた長さに最も近いものを選び出す。例えば、長いテロップと短いテロップとを組みにしておき、状況に応じてどちらかが再生されるようなオブジェクトを組み合わせることができる。

これら以外にも、たとえばクロマキー処理などのビデオオブジェクトも複合オブジェクトの概念でモデル化することができる。

次に、これらの複合オブジェクトと伸縮属性がどのような働きをするのかを示す。図2は、SEBoxの中に2つのmessage(固定長)と、それらの間に挿入されているtglue(自由伸縮)の、合計3個のメディアオブジェクトが含まれている例を示している。SEBoxの大きさを変更した時に、内部に含まれるメディアオブジェクトの大きさは、自由伸縮可能なtglueの部分だけが変更されている。

また、複数のtglueが含まれる複合マルチメディアオブジェクトが伸縮する場合のtglueの大きさの計算は、図3で示されるように、全体としての

SEBoxとtglue

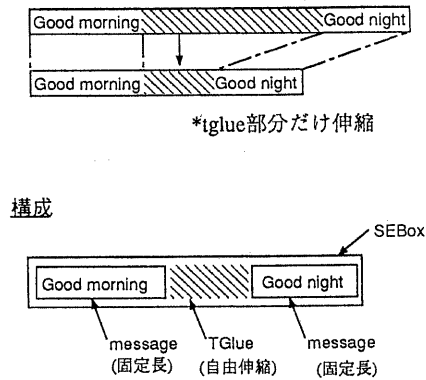


図 2: tglue の伸縮

伸縮量を、各 tglue の shrink あるいは stretch の量によって比例配分することで得られる。

3 複合マルチメディア情報の記録方式について

本モデルでは、作成したマルチメディアの複合オブジェクトのデータを記録する方式として

- I. オブジェクト間の論理的リンク情報のみ記録
- II. 上記情報と素材のデータを共に記録

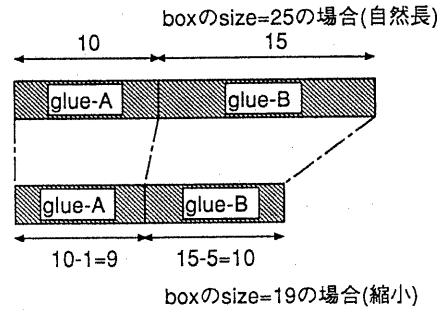
の2つを用意する。

I. の方式で記録された複合オブジェクトは、その修正の際に、論理的同期情報は生かしたまま素材データのみを交換するなどの編集作業を行ないたいときに、リンク先のオブジェクトを他のオブジェクトに置き換える操作のみで可能である。従って、この方式によって生成された複合マルチメディアオブジェクトは再利用性が高められている。

4 同期のための演算

同期のための演算として、Box などの複合オブジェクトのほかに、同期の鍵となる点同士を確実に同期させるための演算として、適当な制約条件を付加したグループ化演算 Group がある。Group は、具体的には複数のメディア間で同期をとるための操作であり、あるメディアの指定した位置と別のメディアの指定した位置を同時に再生するなどの制約をそのグループに付加するものである。制約とは同期をとるためのキーとなるもので、絶対時刻、フレー

A(10, 7, 2), B(15, 1, 10) の2つのt glueの伸縮例



全縮み量: $(10+15)-19=6$

shrinkによって比例配分

$$\begin{cases} \text{Aの縮み量: } 6 * 2 / (2+10) = 1 \\ \text{Bの縮み量: } 6 * 10 / (2+10) = 5 \end{cases}$$

図 3: tglue の伸縮量の計算

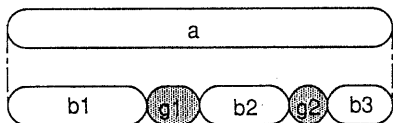
μ等で指定できるほか、論理的な mark によって指定することも可能である。mark には、

- | | | | |
|---|------------|---|--------------|
| 1 | start | → | 先頭 |
| 2 | end | → | 最後 |
| 3 | comp.start | → | グループの構成要素の先頭 |
| 4 | comp.end | → | グループの構成要素の最後 |

等が最初から定義されており、さらにユーザが新たに定義するために Addlabel(tm, loc, mark) を用いることができる。図 4に、これらの関数を用いた応用例を示す。

5 マルチメディアオブジェクトのためのクラス

本モデルを用いてマルチメディアオブジェクト作成システムを構築するために、C++ クラスライブラリとしてインプリメントした。マルチメディア用のサブクラスは、InterViews [3] のクラスライブラリに必要なクラスを追加する形で構築される。InterViews は、スタンフォード大学で開発されたオブジェクト指向 GUI 構築ツールキットで、C++ のクラスライブラリとして実現されているものであり、Xwindow 上にインプリメントされている。筆者らは、InterViews が TeX と同じ Box&Glue モデルを全面的に採用していること、フリーソフトであること、広い範囲で実用的に使われているター



```

g1 ← tglue(0,∞,0)
g2 ← tglue(0,∞,0)
b ← Append(b1,g1,b2,g2,b3)
d ← Group(a,b)
    with a.start==b.start
    && a.end==b.end
display()

```

図 4: 応用例

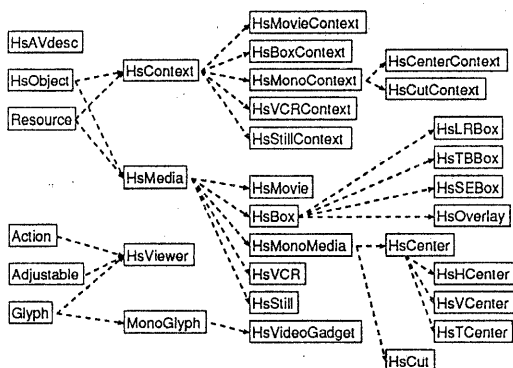


図 5: マルチメディアオブジェクト拡張クラス

ルキットであることなどから、本モデルを実現するための基盤として利用している。

図 5 に拡張したクラス階層を示す。ただし、Hs で始まる名前のクラスが本モデルでの拡張部分である。本クラス拡張では、(基本・複合)メディアオブジェクトを扱う Media オブジェクト、メディアオブジェクトを再生しているときの動的な情報を扱う Context オブジェクト、再生すべきマルチメディアオブジェクトをディスプレイなどの表示機器に結び付けるための Viewer オブジェクトの 3 部分に分け、各々について必要なサブクラスを追加することによって実現している。

6 インタフェースイメージ

構成モデルの考え方を取り入れた上で、ユーザが複合マルチメディアオブジェクトを簡単に構築する作業を支援するためのツールとして、構成モデル

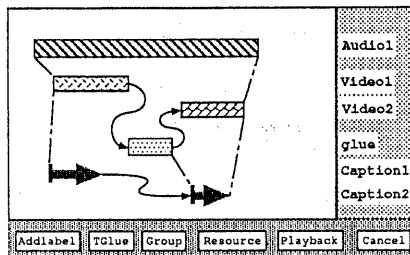


図 6: 複合オブジェクトの生成

の考え方をとり入れた、同期を簡単に指定するためのユーザインタフェースとして、図 6 のような論理同期エディタを用意することにした。このインタフェースには、同期のキーとなる mark の指定や mark の追加コマンド (Addlabel) が装備されており、ユーザが簡単にラベル定義ができる。また、mark を基準として複数のリソース間の同期を規定し、グループ化することができる。図 6 の例は、素材オブジェクトとして 2 つの映像 (Video1, Video2) とそれぞれに対する見出し (Caption1, Caption2)、および BGM としての音声 (Audio1) が与えられたときに、複合オブジェクトを生成するための同期指定を行なっている様子を表したものである。

図 6 で与える同期制約条件は、

- Audio1、Video1、Caption1 が同時に開始
- Caption1 の終了と同時に、Video2 と Caption2 が開始
- Audio1、Video2、Caption2 が同時に終了

の 3 つである。ただし、Video1 と Video2 との間は無映像状態である。ここで Video1 と Video2 の長さの和は、Audio1 の長さより短い。そこで、Video1 と Video2 の間に tglue を挿入することによって同期をとることができる。見出しについては、Static 操作により時間軸を持たせる、すなわち tglue を付加させる。

図 7 は、図 6 で指定した複合オブジェクトがどのように生成されるかを示したものである。図 7 において、Video1/Video2/Audio1 などの既存のリソースは図 6 と同じ長さであるが、tglue が伸びることによって全体の長さを調節している。

図 8 は、論理エディタから出力されるオブジェクト間のリンク情報の一例である。

また、この論理エディタを中心としたマルチメディアオブジェクトマネージャの構成について図 9 に

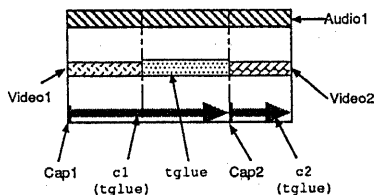


図 7: 複合オブジェクトの同期指定

```

a ← Append(Video1, TGlue(0,0,∞), Video2)
c1 ← Append(Static(caption1), TGlue(0,0,∞))
c2 ← Append(Static(caption2), TGlue(0,0,∞))
c ← Append(c1, c2)
d ← Group(Audio1, a, c)
  with Audio1.start==a.start==c.start
    && a.Video2.start==c2.start
    && Audio1.end==a.end==c.end
display()

```

図 8: 論理エディタから出力されるリンク情報

示す。図 9 において Composer 部分が論理エディタに相当し、オブジェクト間の論理的な同期、空間的な同期等を整理する。そこから出力される論理的制御情報を解釈して物理的制御情報に変換するのは、図 9 の Controller 部分である。Controller は Composer から情報をコンパイル処理し、Video、Audio 等の仮想デバイスに対して(実時間)制御情報を送る。LD、VTR、CD、MIDI 等の物理デバイスは、この仮想デバイスクラスを通して制御される。

7 まとめ

マルチメディアオブジェクトを扱うためのモデル(構成モデル)について、主にユーザインタフェースの観点からその構成、応用について考察した。本構成モデルは、取り扱うすべてのマルチメディアオブジェクトに対して伸縮属性を与えることを特長とし、主に同期について柔軟且つ容易にエンドユーザが指定することができ、簡単な操作の組合せによって複雑なオブジェクトも構成することができる。また複合マルチメディアを構築する際の、複数のメディアオブジェクトを組み合わせる方式を、従来のタイムラインモデル方式ではなくメディアオブジェクト間の相互の制約関係を基に記述していく方式であるため、作成されたオブジェクトの再利用性が高められており、現在評価中である。

今後の課題としては、空間的な制約条件につい

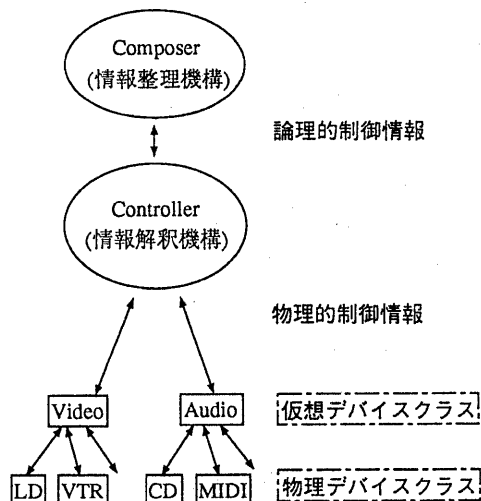


図 9: Multimedia Object Manager の構成

ても、柔軟且つ簡単に指定できるようなモデルおよびインタフェースを構築、本モデルおよびインタフェースとの融合をはかることがあげられる。

参考文献

- [1] MacroMind Inc. *Director User's Manual*, 1990.
- [2] Adobe Systems Inc. *Adobe Premiere User Guide*, 1991.
- [3] M. Linton, J. Vlissides, and P. Calder. *Composing user interfaces with InterViews*, Computer, Feb. 1989.
- [4] Donald E. Knuth. *The TeXbook*, Addison-Wesley, 1984.
- [5] 濱川 礼、暦本 純一、マルチメディアオブジェクトの構成手法、第 44 回情処全国大会, 1992.
- [6] 暦本 純一、坂上 秀和、濱川 礼、マルチメディアオブジェクトモデルとその実現、日本ソフトウェア科学会第 9 回全国大会, 1992.
- [7] 坂上 秀和、濱川 礼、暦本 純一、HyperStation: オブジェクト指向 GUI ツール InterViews の AV 拡張、第 45 回情処全国大会, 1992.
- [8] Rei Hamakawa, Hidekazu Sakagami, Jun Rekimoto, Audio and Video Extensions to Graphical User Interface Toolkits, Third International Workshop on Network and Operating System Support for Digital Audio and Video, 1992.