

マルチメディア ワークステーション"Yougao"のユーザインタフェース

阿部 聡 中村 公治 前川 守

東京大学理学部情報科学科

高品質マルチメディア・データを扱うためのワークステーション"Yougao"に実装されたグラフィカル・ユーザインタフェースの基本設計について述べる。Yougaoでは、ビューとシーンと呼ばれる概念を導入して、仮想作業空間の表示と操作を一般的にモデル化している。これは、従来のシステムのウインドウとその表示内容を表示オブジェクトとして統一、抽象化したもので、配置空間をマッピングすることで階層をつくり、ユーザ操作空間を構成する。各表示オブジェクトはフルカラー、任意の半透明度が持てるほか、背景によって表示内容を変える機能的な表示オブジェクトが定義できる。データの可搬性を高めるため、メディア表現はデバイス独立に記述する。

An Introduction of the User Interface System of the "Yougao" Multimedia Workstation

Satoshi Abe Koji Nakamura Mamoru Maekawa

Department of Information Science, Faculty of Science,
University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113 Japan

The design of the graphical user interface system implemented on the "Yougao" multimedia workstation is presented. Yougao has an object-oriented graphical human interface. Many user operations are defined as a manipulation of virtual work spaces. They are modeled by objects called "views" and "scenes". Views represent all visible objects which include both windows and contents in conventional window systems. Virtual work spaces are hierarchically structured by mapping spaces (scenes) to views.

High level graphic abilities such as half transparency and anti-aliasing are supported at the system level. Media representation is device-independent.

1.はじめに

Yougaoは、高品質メディアを扱うことを目的としたマルチメディア・ワークステーションである。Yougaoは、フルカラー画像／グラフィクス、高精度文字、音声、動画を扱うが、ここでは主に2次元のグラフィカルユーザインタフェースと表示モデルについて述べる。

Yougaoでは、従来のマルチウィンドウ環境でのウィンドウ、アイコン、パネル、ボタンなどを、すべてビューと呼ばれるオブジェクトとして一般化して扱う。操作環境は、簡単なCADのようになっている。とくに制限を設けないかぎり、ビューは自由に配置できる。このとき、拡大、縮小、回転などのアフィン変換を行うことができる。ユーザの操作する空間は仮想化されており、デバイス独立なグラフィクス表現を持つ。これによりスクリーンサイズやピクセル数、プリンタ解像度などを意識せずにビューを記述できる。

ビューは、任意形状、フルカラーであるだけでなく、局所的に半透明度が持てる。ビューはシーンと呼ばれる仮想空間に配置され、階層的に管理される。こうした仮想空間の階層構造は、“ビュー・シーン構造”と呼ばれ、Yougaoの基本的な表示モデルとなっている。ビュー・シーン構造を用いて、従来のウィンドウシステムを実現することもできる。しかしビューの表現の自由度や空間の参照関係の自由度を用いると、より強力な表示機構を作ることができる。

あらかじめ、ビューとして、2次元グラフィクス表現とテキストがリポートされている。マウスのオペレーション、キーボードからの入力などは調整の上、各ビューへ送られる。システムレベルで表示／操作だけでなく、仮想空間管理や変形、半透明度といった機構を備えたことにより、アプリケーションソフトウェアのグラフィカルユーザインタフェースの負担は大幅に軽くなる。

2.ビュー、シーンモデル

2.1 ビューとシーン

Yougaoで扱うメディアは、すべてビューとシーンと呼ばれるオブジェクトで階層化して扱われる。ここでいう“メディア”には、画像グラフィクス、テキストのほか、音声や動画も含まれるが、こ

ではまず、画像、グラフィクス、テキストの3つの静止表示型のオブジェクトについて述べることにする。

ビューは、機能的には、“描画できるオブジェクト”の総称である。モデルとして見ると、ビューは、ものの見えかたを規定するオブジェクトということもできる。ビューは、ローカルに座標系を持ち、それ自身描画可能である。ビューを複数個表示するには、複数のビューを空間に“配置”しなければならない。ビューを配置するための仮想空間オブジェクトは、シーンと呼ばれる。シーンは仮想空間であり、シーンそのものは不可視である。シーンが“見える”ようにするためには、シーンの上に、シーンの見えかたを規定するビューをつけなくてはならない。

あるシーンの見えかたを決めるビューを、別のシーンに組みこむことによって、シーンとビューは階層構造を持つことができる。図1は、ビュー・シーン構造の階層が浅い例を図示したものである。screen1は、ビデオスクリーンなどの表示デバイスを示す。view1は、スクリーンの絵を最終的に決める基本的なビューで、とくにルートビュー (root view) と呼ぶ。同様にscene1は、ルートシーン (root scene) と呼ばれる。

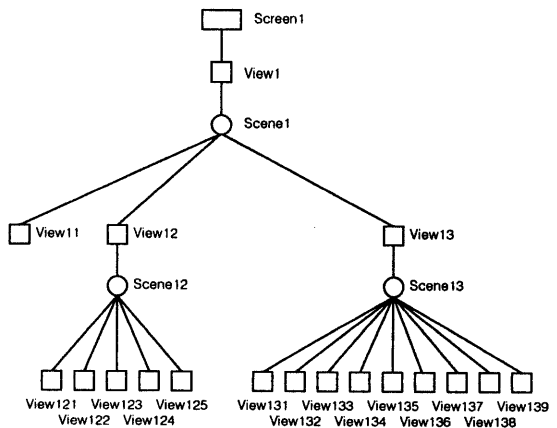


図1 ビュー・シーン構造

2.2 ビュー

2.2.1 リーフビュー

図1のview11は、枝の終端にあり下に何もつけていない。これは、ビューが何も参照しておらず、ビューそのものが表示実体となっていることを示す。このようなビューを、リーフビュー (leaf view) と呼ぶ。リーフビューは、任意の形状、色、半透明度を持つことができ、単純なアイコンやボタン、しおり、仮想空間の装飾などに使われる。

2.2.2 シーンビュー

図1のview12とview13は、それぞれ下にscene12とscene13を持っている。このように、シーンの見えかたを規定するビューを、とくにシーンビューと呼ぶ。もちろんルートビューview1は、ルートシーンscene1を可視化するシーンビューの1つである。

シーンビューは従来のウィンドウシステムのウィンドウに相当するオブジェクトだが、任意の形状、色、半透明度を持つことができる。このうち形状は、シーンを見るためのクリッピングウィンドウになる。色は、そのシーンビューの、いわばバックグラウンド模様で、何もビューがないときに表示される図柄である。半透明度は、その背景も含めてそのシーンビュー全体にかかる半透明度のことで、色と同様、局所的に変化させることができる。これによって、たとえば周囲を半透明にぼかしたシーンビューを作ることできる。

シーンビューは、下についたシーンとともに、ビュー・シーン構造の階層の基礎をなし、極めて重要な役割を演じる。従来のウィンドウシステムにおける階層ウィンドウとしての機能はこのまま果たせるほかに、よりアプリケーションよりの働きを持たせることができる。たとえば、文書編集アプリケーションを考えると、文書のページ全体を1つのシーンと考え、その中に、テキストや図などのビューが配置されていると考えることができる。また、ボタンの並んだ操作パネルや、文字盤と針から構成されるアナログ時計なども、シーンビューとシーンで表現できる。

2.2.3 レンダリング・ビュー

レンダリング・ビューとは、モデルを可視化するビューのことをいう。リーフビューと違い、ビュー自体が実体ではない。たとえば、一般的な音楽表現を楽譜として可視化するビューや、スプレッドシートの数値をグラフにして表示するビューなどがこの例である。Yougaoでは、標準的にはテキストも、そのものは不可視として扱われるので、テキストをレンダリングするビューも定義されている。図2は、表とテキストのレンダリングビューの例を示している。view12はTextを可視化していて、view121はTableを円グラフとして可視化している。円グラフview121は、六角形のウィンドウview12の中に表示されている。

レンダリング・ビューには、この他にも、3次元のモデルを2次元の画像としてレンダリングするようなビューなどが考えられる。シーン・ビューも、シーンをモデルとしたレンダリング・ビューの特殊な場合と考えることもできる。

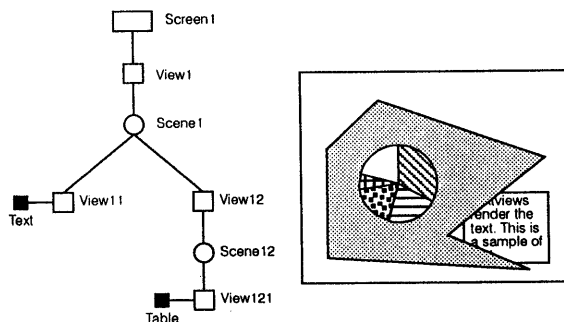


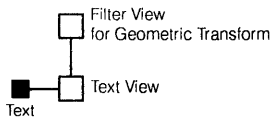
図2 レンダリング・ビュー

2.2.4 フィルター・ビュー

ビューそのものに、さまざまな変形や特殊な表示効果などをつける場合、ビューの上に、表示変換用のビューを重ねてつけることがある。こういったビューは、「フィルター・ビュー」と呼ばれる。フィルター・ビューは、主に変形、クリッピング、歪曲、ぼかしなどのさまざまな視覚効果に用いられる。図3は、フィルター・ビューの働きを図示したものである。



Pour le jour des
Hyacinthies, il m'a
donné une syrxinx faite
de roseaux bien
taillés, unis avec la
blanche cire qui est
douce à mes lèvres
comme du miel.



Pour le jour des
Hyacinthies, il m'a
donné une syrxinx faite
de roseaux bien
taillés, unis avec la
blanche cire qui est
douce à mes lèvres
comme du miel.

図3 フィルター・ビュー

フィルター・ビューは、ビューに直接の影響を与えずに付けたり、はずしたりできるとことに意味がある。一時的な変形、たとえば絵の編集集中に、一部を拡大してみたいときや、原画像を壊さずに絵に効果をかけたいときなどに有効である。システム標準のツールボックスには、よく使われるフィルター・ビューを生み出すオペレータが装備されている。このオペレータを使うと、ユーザはアプリケーション・プログラムとは独立に、さまざまなビューにフィルター・ビューを適用することができるようになる。

フィルター・ビューも、広い意味では、ビューを再可視化するレンダリング・ビューの一種と考えられる。実際レンダリング・ビューが一般的に持っている機能はすべて、フィルター・ビューも持っているので、そのように考えて利用することもできる。ただし、實質上は、フィルター・ビュ

ーは他のレンダリング・ビューとは違った扱われかたをする。これは、表示を制御する“スペース・マネージャ”と呼ばれるモジュールが、ビューを特別に扱っているためである。

2.3 マルチプル・ビュー

1つのモデルを、複数のビューが参照することがある。このような構造を、“マルチプル・ビュー” (multiple view) と呼ぶ。

例として3次元のモデルを考える。このモデルをさまざまな角度から見た透視画像を描くレンダリング・ビューを複数用意して、すべて1つの3次元モデルを参照するようにする。このようなマルチプル・ビューを定義すると、3次元モデルの変化が、それを参照するすべてのビューに自動的に伝達されて、それぞれのレンダリング・ビューの絵も変化する。つまり、マルチプル・ビューでの参照には、各ビューとモデルの間に一貫性が保たれるのである。

こういった多重参照の考えかたは、データベースによく用いられている。ビュー・シーン構造でのマルチプル・ビューは、ビューが異なるメディア表現にわたっても良いところに特徴がある。もう1つの例として、スプレッドシートをモデルにした場合を考える。図4に、この構造を示す。2つのビュー、view11とview12が、1つのモデルTableを参照するレンダリング・ビューである。view11は、Tableを最も自然な形、すなわち表で可視化している。これに対してview12は、この表を棒グラフの形で可視化している。ユーザの操作などによって、Tableの内容が変化するとき、Tableが変化したことは、view11とview12の両方に伝えられ、それぞれのビューは、表示内容を変える。マルチ

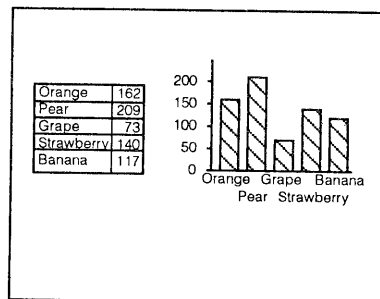
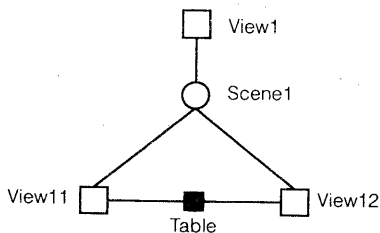


図4 マルチプル・ビュー

ブル・ビューは、表示だけでなく、ユーザの操作にも有効である。たとえば、図4の例で、ユーザはTableの内容を、view11を通して、つまりキーボードからコマンドや数値を打ち込んで変更することができるだけでなく、view12を通して、つまりマウスでメニューを選んだり、グラフの端をドラッグしたりすることによっても変更することができる。こうして行った変更は、Tableを参照するもう一方のviewに伝達され、そのビューの表示も変更される。

3.メディア表現

Yougaoのメディアには、2次元グラフィクス、テキスト、サウンド、動画があるが、ここではユーザインタフェースの基本システムに関係の深い表示オブジェクトについて述べる。

3.1 2次元グラフィクス

3.1.1 グラフィック・オブジェクト

Yougaoでは、標準的に、shapeとimageの2つのタイプのグラフィクス・オブジェクトをサポートしている。shapeは、パスとグラフによる2次元グラフィクス表現で、直線とBezier曲線によって構成される幾何学的な平面図形である。座標系、色彩、半透明度は、デバイス独立になっており、線形変換を繰り返し適用することができる。グラフィック・レンダラー“reng”は、デバイスドライバにディスプレイやフレームメモリの構成を問い合わせ、デバイスに適した方法で図形をレンダリングする。

imageは、サンプリングされた画素の配列である。マスクプレーンを指定することによって、画素単位に任意の形状が持てる（ただし、表示だけを考えるなら、半透明度プレーンを使っても同じことができる）。shapeと同様、色彩、半透明度はデバイス独立だが、ピクセルもデバイス独立である。つまり、imageに記述されているピクセルは、仮想ピクセルであり、直交座標によるものであれば、任意のピクセルアスペクト比を持つことができ、原画像の情報を失わずに線形変換も自由に適用できる。イメージ・レンダラー“gento”は、補間やアンチエイリアシングなどの処理をしながら、デバイスに適した方法でイメージをレンダリングする。

3.1.2 半透明度

Yougaoは、フルカラーのグラフィクスをサポートするだけでなく、グラフィック・オブジェクトやテキストに、任意の半透明度を局部的に持たせることができる。半透明度の記述には、通常 α （アルファ）値と呼ばれる値を用いる。 α 値は、背景と前景を線形に補間する値で、次の式で定義される。

$$I_p = I_b + \alpha (I_f - I_b), 0 \leq \alpha \leq 1$$

ここで、 I_b 、 I_f 、 I_p はそれぞれオブジェクトの置かれる前の背景の画素、オブジェクトの色（前景）の画素、オブジェクトが置かれた後の画素の光の強度を表す。

α 値は、通常はアンチエイリアシングに使われる値で、色とは独立に単一の値を持たせることが多い。しかしYougaoでは、視覚効果に利用できるように、赤、緑、青の各発色プレーンに対してそれぞれが α 値を持てるようにしている。ただし、フレームメモリに書き込まれた画素値と発光強度の関係は線形ではないので、補間の前後に変換が必要になる。図5に、Yougaoの画素値の決めかたを示す。

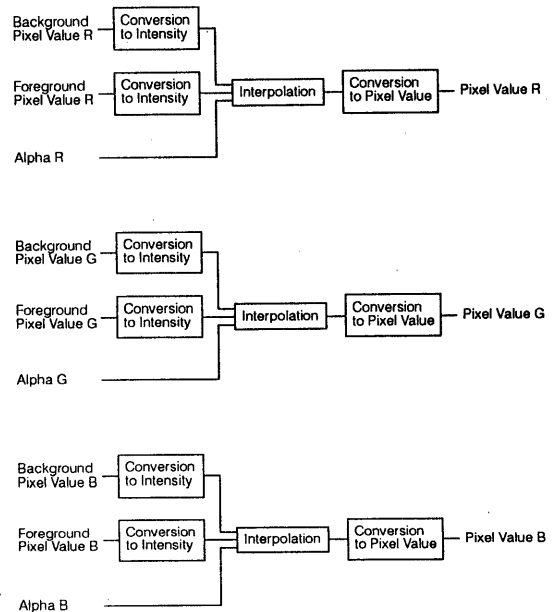


図5 α 値

3.1.3 抽象グラフィクス・プレーン

Yougaoでは、フルカラーと半透明度を表現するのに、抽象グラフィクス・プレーンという概念を用いる。抽象色プレーンは色や透明度を、物理的な3原色のフレームメモリに制限されずに、複数のプレーンで表現することを目的としている。

抽象グラフィクス・プレーンは、表示用フレームメモリとは別にシステム内にとられた複数枚の画像メモリである。グラフィック・オブジェクトは、最終的には、R (赤)、G (緑)、B (青)、 α_R (赤の α 値)、 α_G (緑の α 値)、 α_B (青の α 値)とshape (形状を表す1bitのマスキ)の7つの要素で表現されるが、レンダリング中は、そのビューを描くのに都合の良い使いかたで抽象グラフィクス・プレーンを使う。画像を表示用フレームメモリにコピーするとき、抽象グラフィクス・プレーンと7つの色要素との関係を表示マネージャ"showgi"に渡す。

図6に、抽象グラフィクス・プレーンを用いた色表現の例を示す。Aは、 α 値固定でフルカラーの絵を描く場合を示す。Bは、 α 値が変化する白

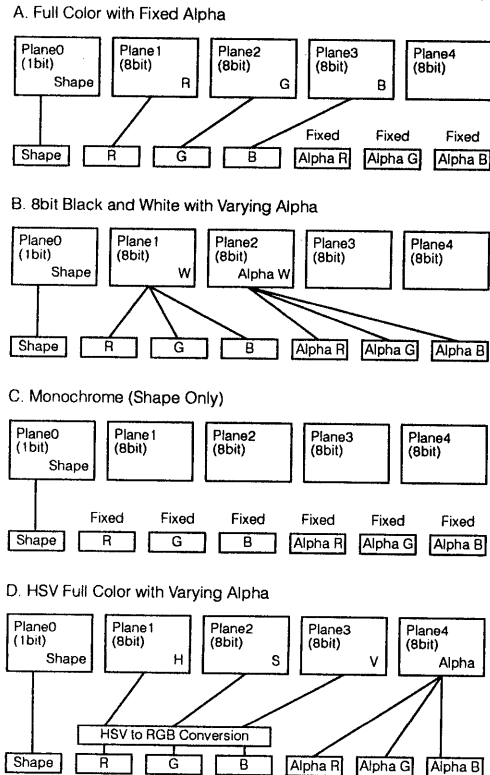


図6 抽象グラフィック・プレーン

黒画像を描く場合である。8bitの抽象グラフィクス・プレーンを2枚用いれば良い。Cは、モノクロームの形だけ (アイコンなどに多く使われる) を描く場合の例である。この場合、1bitのshapeプレーンだけで描画できる。最後にDは、RGBではなく、HSV色モデルで、可変 α 値フルカラーの絵を描く場合を示す。図6で示される通り、多くの場合、1bitのプレーン1枚と8bitのプレーン4枚があれば十分に絵が描ける。

3.1.4 半透明度の概念の拡張

Yougaoでは、 α 値による半透明度だけではなく、背景に依存して画素の色が決まるような部分を、すべて「半透明」と呼んでいる。こうした拡張された半透明ビューの例としては、背景の絵を拡大する「虫めがね」ビューや、背景の輪郭を抽出するビュー、背景の色調を変換するビューなどが考えられる。ビューの表示と操作を管理するスペースマネージャと呼ばれるプログラムは、各ビューの半透明部分と不透明部分を分けて管理している。半透明な部分については、原則として背景を先に描画する。

このように、拡張された半透明ビューは、フィルター・ビューのような役割を果たすことができる。フィルター・ビューとの違いは、半透明ビューは、ビュー・シーン構造には影響を与えない点にある。半透明ビューは、特定のビューの表示方法を変換するものではなく、属するシーンの中に自由に配置され、どのような部分であろうと背景を変換して表示する、手軽な道具としての機能を果たす。

3.2 テキスト

Yougaoでは、テキストは単なる文字列ではなく、構造が持てるよう設計されている。テキスト中には、多重のブロックが定義でき、各ブロックは名前を持つ。これによって、たとえば章や節という文書の構造が表現できる。

テキストのフォーマット情報には2種類ある。1つはハードフォーマット情報と呼ばれるもので、表示モードや文字の書体と大きさ、字送り、行送り、段幅など物理的な体裁を規定する。もう1つは、ソフトフォーマット情報

と呼ばれるもので、ルビや割注、上つき／下つき文字、脚注など、文書の表現構造を記述する。通常テキストオブジェクトは、ソフトフォーマッティング情報だけを持つ。そしてそのテキストを参照するレンダリングビューが、ハードフォーマッティング情報を持つ。このようなフォーマッティング情報の分離は、テキストオブジェクトのポータビリティを高めるのに有効である。

文字もグラフィクス・オブジェクトと同様、フルカラーで α 値を持つことができ、大きさに関してもデバイス独立である。文字フォントは、アウトラインリプレゼンテーションを使用しており、線形幾何変換を施しても高品質の出力を得ることができる。また、アンチエイリアシングをかけることにより、たとえば7ドット角程度の小さい文字でも日本語が判読できる。

マルチプル・ビューを用いると、WYSIWYG出力を見ながら、テキストを構造的に記述／編集することができる。テキストの編集操作は、テキスト操作マネージャtemaliが行う。

4. オペレーション

Yougaoの基本画面は、従来のウィンドウシステムの“デスクトップ”ではなく、“ウォールフェイス”という壁を模した構造になっている。ウォールフェイス上での基本操作の大半は、2ボタンのマウスから行う。マウスは、画面上のポイントを移動させるだけでなく、プレス（ボタンを押す）、リリース（ボタンを離す）、クリック（ボタンを押してすぐ離す）、ドラッグ（プレスしたままマウスを動かす）などの使いかたがあり、これらを組み合わせてウォールトップを操作する。従来のウィンドウシステムでは、マウスにさまざまな機能を持たせるために、キーボードのファンクションキーやシフトキーと併用するマウス操作が多いが、両手が必要なこともあって、直感的な操作の障害になる。このためYougaoでは、極力キーボードとマウスの併用を避けるため、2つのボタンを組み合わせた操作を導入している。

マウスを使った操作のほとんどはプログラマブルだが、標準して次の5つのウォールフェイス操作が用意されている。

1. メニュー

マウスの右ボタンをクリックすると、ポッ

プアップメニューが表示される。メニューの中には、現在示されているビューに対するさまざまなコマンドが並んでいる。

2. メタモルフォーゼ(オープン/クローズ)

従来のウィンドウシステムでは、オブジェクトにアイコンの状態とオープンされたウィンドウの状態の2種類を持つ場合が多い。

Yougaoでは、アイコンとウィンドウはどちらも1種のビューであり、表示構造上の違いはないので、2種類に限らず、1種類以上任意の状態をとることができる。この n 個の状態をサイクリックに変えていくのが、メタモルフォーゼと呼ばれる操作である。これには、目的のビューの上で左ボタンを押しながら右ボタンをクリックする。

3. 移動

マウスの右ボタンを押すと、選択されたビューの輪郭が表示され、とくにビューに制限がないかぎりシーンの中を自由にドラッグして移動できる。

4. ポスト/適用

移動中、右ボタンを離す前に、左ボタンをクリックすると、移動中のビューを他のシーンに移動させることができる。この操作をポストと呼ぶ。

ポストはこの他に、2つ以上のビューを結びつける重要な機能を持つ。たとえばアプリケーションのアイコンビューをドキュメントのアイコンビューにポストすると、そのアプリケーションがそのドキュメントに適用され、アプリケーションが立ちあがるとともにドキュメントがオープンする。この操作は、ビューの大きさを変えたり、回転したりするときにも用いる。標準ウォールトップには、ビューの大きさを変えたり、回転したりするためのビュー・オペレータと呼ばれるアイコン・ビューがある。これを目的のビューにポストすると、ビューを変形することができる。

5. 選択

マウスの左ボタンを押すと、ビューを撰択することができる。ビューは選択されると、それぞれのビューに特有な行動をとるが、最も典型的な例は、ドキュメントのビューが選択されると、編集可能になることである。オ

ペレータが選択されると、通常マウスポインタの形状が変わり、他のビューを直接操作できるようにする。

5. 結論

最近、マルチメディアを指向したシステムが数多く開発されている。しかし、各種のメディアを統合的に扱う手法に関しては、まだ十分に研究されていない。Yougaoでは、階層的にマッピング可能な仮想空間を利用することによって、空間/時間的な意味でメディアの統合を試みている。本論文では、主に空間的統合を取り上げたが、現在、ビューにタイムの概念を持ち込むことによって、時間的な意味でのメディア統合システムを試作中である。私たちは、メディアの統合は、メディア表現そのものの可能性を十分考えた上で行うことを方針としている。それは、1つには、現在のグラフィクス、テキスト、サウンド、アニメーションなどメディア一般を扱う環境が、品質、抽象度などから見てまだメディア統合の効果を発揮するまでに発展していない部分があるように見受けられるからである。多少冒険的なシステムであるために、現在ではハードウェアのパフォーマンスによって受ける制約も大きいですが、この実験が少しでもマルチメディア環境の新たな可能性を広げることになれば幸いである。

参考文献

- [1] Maekawa M, Sakamura K, Ishikawa C, Shimizu T (1983) Multimedia machine. Information processing 83 (IFIP). Elsevier Science Publishers B.V., North-Holland, pp 71-77
- [2] Ohta M, Maekawa M, Arano T, Kawachiya K (1986) Multimedia information processing based on a general media model. Information processing 86 (IFIP). Elsevier Science Publishers B.V., North-Holland, pp 957-962
- [3] Foley JD, van Dam A (1982) Fundamentals of interactive computer graphics. Addison-Wesley, Massachusetts
- [4] Foley JD, van Dam A, Feiner SK, Hughes JF (1990) Computer graphics: principles and practice. Addison-Wesley, Massachusetts
- [5] Rubinstein R (1988) Digital typography. Addison-Wesley, Massachusetts
- [6] Abe S, Maekawa M, Shimizu K, Kato T (1989) Fast display of high-quality anti-aliased Japanese characters. Raster imaging and digital typography (RIDT89). Cambridge University Press, Cambridge
- [7] Moore FR (1990) Elements of computer music. Prentice Hall, Englewood Cliffs
- [8] Adobe Systems Incorporated (1985) PostScript language reference manual. Addison-Wesley, Massachusetts