計算機アーキテクチャ 49－2
（1983. 6. 3）

マルチメディア・マシンM³のアーキテクチャ

坂村 健、石川千秋、清水 徹、前川 守
（東京大学理学部情報科学科）

1．はじめに

別の小論[1]で述べたTOM³がマルチメディアの可能性を追求する実験機であるのに比べ、MMM（マルチ・メディア・マシン、M³とも言う）はワークステーション的色彩を強め、多少の制限（例えば動画は取り扱わない）をつけてもインプリメンテーションを重視したアーキテクチャとなっている。MMMはTOM³と同じくマルチメディア・マシン用のブロードバンド方式のローカル・ネットワークに接続されて使われるというイメージである。

MMMでは、静止画像や音声など種々のメディアのデータを取り扱うにあたって全てを（例えば磁気ディスクのような）単一の記憶媒体に格納することはしない。それぞれのメディアに最適な記憶媒体を用いることによりコストパフォーマンスの向上を図っていく。

MMMの設計上興味深いのは、種々のメディアの様々な形式のデータを、概念的に整理されたきれいな形で操作できるようにするためのユーザインタフェースのデザインである。MMMのユーザインタフェースは、種々のメディアを統一的に取り扱うため、様々なデータ形式に対して一様で統合された処理機能を実現しなくてはならない。

このような一様な、概念的に整理されたユーザインタフェースを作るには、マルチメディアのデータやその処理の概念モデルが必要である。小論では、このモデルを中心に述べたい。

2．マルチメディアデータ

ここではマルチメディアのデータがMMMでは情報オブジェクトという表現で表わされ、これを基本として操作されることを述べる。

2.1 情報オブジェクト

MMMは、多くの要素から成る情報のオブジェクトを取り扱う。例えば、文章は節から、節は 段落から成り、更に段落は文から、文は単語から成る。また、図は三角や四角などの基本形から構成され、更に基本形は直線や点から成る。さらに文章中に図がはいってもよい。なお、一般に情報オブジェクトの内部表現はトリーあるいはDAG(Directed Acyclic Graph)構造のリストの形状になっている。

2.2 情報の内容と表現形の分離

一様なインタフェースを実現するには、情報の内容と表現形を別々に分けて考える必要がある。そして個々の情報に関して、その内容を変える事なく種々の形式で表現したり、或は表現形によらず情報の内容は一様に取り扱うことも必要である。ある指定した形式における情報の外部表現を、出力インスタンスと呼ぶ。ユーザは、MMM 上の出力インスタンスを通じて、情報を取り扱うことができる。

2.3 レイアウト

レイアウトとは、出力メディア上での、情報オブジェクトの外部表現を整えてやることである。レイアウトの例として、CRTに文字画像出力を行なう場合がある。また音声出力の場合には、相対的な音量や音の流れの組み合わせなどがレイアウトにあたる。ユーザは、最終的な出力が人間にわかりやすいようにレイアウトを組み立てなければならない。

3．情報オブジェクトの構造

情報の外部的な側面は出力インスタンスとして、出力デバイスに表示される。この際、出力インスタンスは、ユーザの与えたフォーマット情報に従って構成される。一方、情報の内部表現は、ユーザのコマンドに対して、MMMが容易に操作でき、また容易に出力できるものでなければならない。

我々は、情報を表現する基本的なデータ構造として、"木構造"を選んだ。情報木（I-tree）は、情報の内容を維持する木構造である。I-treeは、フォーマット仕様に従って位置づけられ、出力インスタンスとなる。こうしてできた出力インスタンスは、他の出力インスタンスと組み合わせて、最終的なディスプレイ形式であるレイアウトとなる。図1はこの様子を表わしているがここではフォ

ーマット仕様を"割り付け原稿"、出力インスタンスを"版下"と呼んでいる。出力インスタンスは、I-tree全体についても、また部分についても生成できる。ある時点でユーザーに見える(あるいは感じる)レイアウトは、出力インスタンスを一時的に組み合わせたものである。そのレイアウトをつくりだすためのフォーマット仕様の組み合わせは、名前を付けて登録して、後でもう1度使う場合には、簡単に参照出来る。

3.1 I-tree

I-treeは、文章や表、グラフ、画像、音声などを表わすノードやリーフから成る。I-treeは、それぞれ特定の情報オブジェクトを表わし、後で述べるようなコマンドで操作できる。本に相当するI-treeを図2に示す。I-treeのノードは、ノード名とそのノードに対応した情報オブジェクトのタイプを示す。例えば、図2に示したI-treeのノードは、本、章、序文、段落、文章、図、表などを示している。また、これらのタイプのフォーマットは、次節で述べるフォーマット仕様で指示される。I-tree自体も、ノードで指定されるタイプである。これにより他のI-treeを基にI-treeを作ることができる。

I-treeのリーフは、情報オブジェクトの実際の内容であり、色々な形式のものがある。リーフの値は、次節で述べるコマンドで指定することができる。また他の情報オブジェクトと関連づけることもできる。例えば、図の番号である。文章中の図番号と図を関連させることにより、片方を変えた場合、もう片方も自動的に変わる。

3.2 フォーマット仕様

情報のタイプそれぞれに対してフォーマットが決められる。フォーマットは、そのタイプの情報オブジェクトがどのように表示されるかを定めるものである。フォーマットは、いろいろな方法で指示できるが例えばディスプレイ上で例示により、指示することができる。

我々は情報の内容とその表示フォーマットを分けてとらえることが、洗練されたフォーマッティングを行うためには重要だと考えている。

3.3 I-treeとフォーマット仕様におけるオペレーション

I-treeの入力にあたりI-treeの生成を簡単にし、ユーザ指向のインタフェースを与える為、我々は入力方式として例示方式を採用した。ユーザは、画面エディタにより、枠組みを作り出せばよい。それをもとに、MMMが必要な情報を自動的に抜き出す。図3に例を示す。I-tree はこの枠組みから生成される。

I-treeが表示される場合、MMMはリーフそれぞれに対してウィンドウを開き、これをフォーマット仕様に従って表示する。ただし、後から手でウィンドウの位置や大きさを変えることもできる。後からでてくるウィンドウムーバーの記述からもわかるようにウィンドウのあいだに親子関係があり、子供のウィンドウは親のウィンドウに完全に含まれる。

I-treeやフォーマット仕様を変更する機能がある。その中には部分木の付加、削除、交換、そしてリーフのデータや内部ノードの指定の変更などが含まれる。I-treeとして表現された情報オブジェクト上の検索は、キーと範囲を指定して行うことができる。このキーは、リーフ中の要素か、ノード中の指定である。例えば、画像上で検索する場合には、図の一部、多角形、線、点単位でこれを実行することができる。

検索の具体的な方法は、トリーのマッチングであり、リーフ同士のマッチングはそれぞれのデータに応じて定義される。

4. エディタとフォーマッタ

4.1 フォーマッティング

I-treeを生成したり、フォーマットを指示したりするのに、2次元のフォーマッタが用いられる。フォーマッタは、フォーマットのガイドラインを得て動作する。このガイドラインは、図は文中にはさむか、それとも節の終りに置くか、或は図を置くには頁の上下左右どちらによせたらいいか、といったことをフォーマッタに指示する。

ウィンドウの位置に関するガイドラインはスクリーン上の絶対位置ではなく、コンテキストに応じた位置として与える。例えば、図を、その解説のある段落のすぐ近くに置くように指示してやることができる。この場合、フォーマッタは、CRT上に映像を作る際に、その図のウィンドウをその段落のすぐ近くに置こうとする。ウィンドウの正確な位置は、与えられたガイドラインを考慮した上で定まる。

では段落のはじめといったコンテキスト情報はどうやって取り出すか。ユーザがI-treeを作る際に、段落、節、章といった指示を使う。これらの指示は、I-treeのリーフを作る際に用いられる。従ってフォーマッタは、I-treeの構造を調べることにより、コンテキスト情報を抜き出すことができる。

将来は、データ自身からコンテキスト情報を抜き出せるようになるかもしれないが、現状では、我々のアプローチが現実的かつ有効と考える。

4.2 ウィンドウムーバ

自動的に生成された画像に対して、人間が手を加えることは、非常に重要であるが、従来のエディタやフォーマッタではあまり考慮されていなかった。我々のシステムでは、エディタやフォーマッタと人間の共同作業が重視され、人間とのやりとりを通じて得られた知識が、さらに新たなフォーマットを作り出すのに用いられる。すなわち自動生成された画像が編集されて、さらに新しいフォーマットを作るため、フォーマッタにフィードバックされる。先に述べたように、フォーマッティングの大部分はフォーマッタが自動的に行う。しかし最終的フォーマットはウィンドウムーバにより、ユーザがディスプレイ上で直すことができる。ウィンドウムーバは、回りのフォーマットへの影響を最小限におさえつつ、ウィンドウを動かすことができるので、これによってユーザは、自動的に生成されたフォーマットに対して、最終的な変更を加えることができる。

通常、もとのデータに変更があった場合には、たとえわずかであっても、ウィンドウの位置が変わり、人間が自動生成された画像に加えた変更は無効になってしまう。従って、データに変更が加えられたあとで、新たに自動生成されたウィンドウをユーザが、またはじめから直してやらなければならない。

もとのデータに変更があるたびにユーザが最終フォーマットを直してやらなくてもいいようにウィンドウムーバは手動で行ったウィンドウの変更内容を、できる限り相対的な位置やコンテキスト情報の形で記録する。そしてこの情報を用いることによって、データに少々変更があっても、人間の介入をほとんど必要とせずにフォーマットを作り直すことができる。ここでは人間の変更の意図をフォーマット仕様よりも高いレヴェルで記憶するようにしていくことが有効であり、人工知能研究の成果が応用できよう。

4.3 ウィンドウのフォーマッティング

ウィンドウの大きさや相対的な位置は、絶対値でも相対値でも指定することができる。ウィンドウのフォーマットは、ウィンドウ外のデータフォーマットとは独立に指定することができる。従って、ウィンドウの外側を処理しているフォーマッタにとっては、ウィンドウは幅や高さのみが既知のブラックボックスである。

ウィンドウを入れ子にすることも可能である。この場合、外側のウィンドウの大きさを変えた時に、内側のウィンドウの大きさもそれに比例して変えるかどうか、指示してやることも可能である。

またウィンドウを一部重ね合わせることもできる。この場合、ユーザは、どちらのウィンドウが前にくるか、即ち後ろのウィンドウのブロック部分をフォーマッタに教えてやらなければならない。

4.4 ウィンドウ間の相互作用

ウィンドウ中のある情報が、別のウィンドウ中の情報と関係することがある。例えば、数値データの表があり、文章中でその要素を参照している場合などである。この場合、表とこれを参照している文章には全く同じデータがはいらなければならない。しかし、ユーザはこのデータ間の関係を忘れてしまう怖れがある。従って、エディタやフォーマッタがこの関係を保ってやるとよい。我々のエディタ、フォーマッタには、このような異なるウィンドウにあるデータの対応関係をサポートする機構があり、データの一貫性が破れたり、大きなデータの改訂によりその関係が保てなくなる場合には、警告を発する。

さらに、「番犬」となる手続きをI-TREEのリーフやノードに与えておいて、情報オブジェクトの一貫性を保つことが出来る。 またこれを用いて、VISICALCのような表操作を行うことも(速度の点は別にして)容易である。(図4)

4.5 音声データの取り扱い

視覚メディアと聴覚メディアの2つのメディアを1つのデバイス上で混ぜ合わせることは不可能である。しかし、情報の内容をさほどかえずに片方の出力インスタンスをもう1つの出力インスタンスに変換したりマップしたりすることができる。アプリケーションによっては、2つのメディアの出力インスタンスが、本質的には同じ情報を持っている

ことがある。例えば、楽譜と音楽、パワースペクトラムと記録音などの関係である。

MMMでは音声データを画像表現にする。例えばFOURIER変換を用い音声データを処理すれば、その数値データはディスプレイ上にグラフにして表示できる。この場合に音声データの編集に視覚モードの画像出力を使うことが出来る。音声エディタは画像の出力インスタンスに対して加えられる編集操作を内部の音声データベースに反映する。

このような画像表現を用いる利点として、既に開発した画像編集の機能が使える点がある。さらに画像表現は音の「きりばり」に適している。音声編集では「きりばり」は重要な操作である。テープの編集がよい例である。特に音メディアを画像メディア表現(時間-power spectrum)にすることにより「きりばり」に必要な2つの音の分かれ目を容易に見付けることが出来るようになる。これはテープレコーダなど編集作業に比べると著しく容易な作業であろう。

5. おわりに

MMMは現在、部分実験を踏まえ68000マイクロプロセッサベースのワークステーションとして構築中である。

参考文献

[1]坂村健:異種複合データ取り扱い可能計算機
ＴＯＭ³アーキテクチャ:
情報処理学会 計算機アーキテクチャ
研究会資料49－1 1983

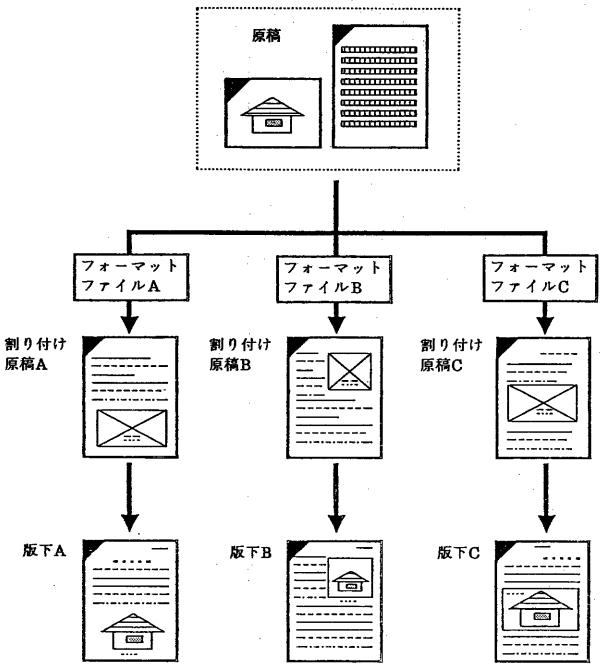

図１　ＭＭＭの原稿と版下の関係

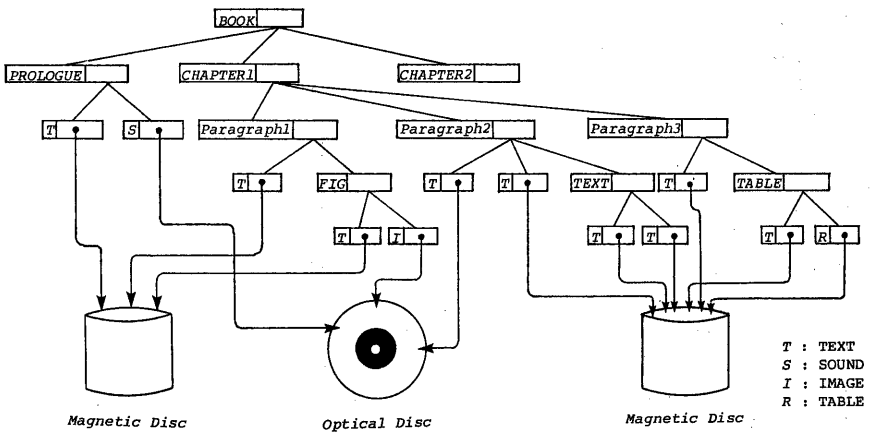

図２　Ｉ－ｔｒｅｅ

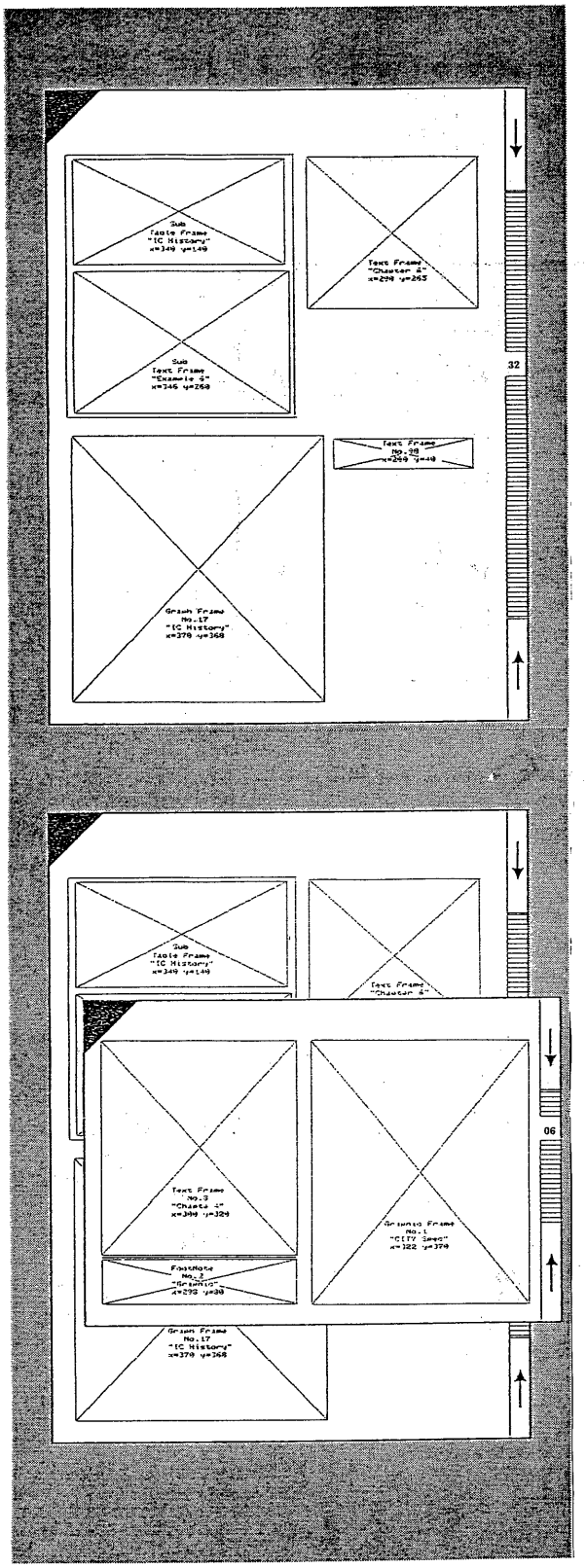

図3 例示方式によるI-treeの作成
（その結果の出力インスタンスは図4参照）

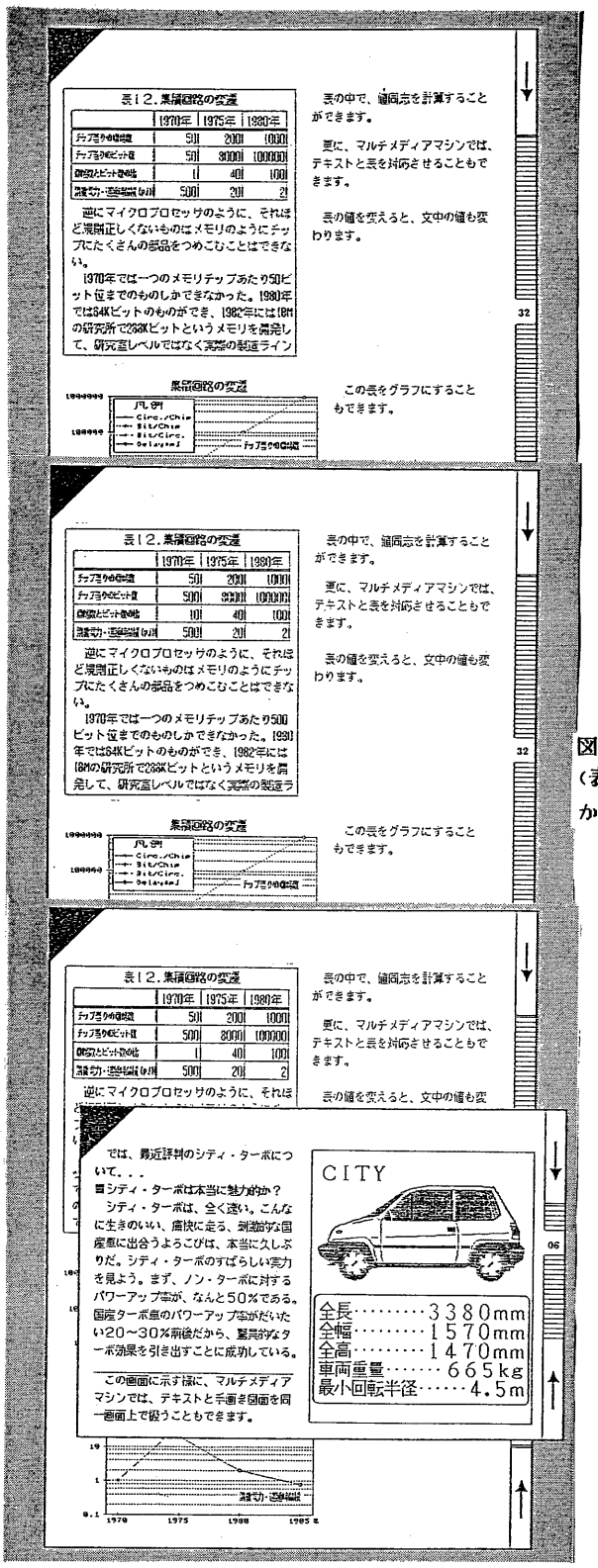

図4 相互作用例
（表中で1970年のﾁｯﾌﾟ当りのﾋﾞｯﾄ数を50から500に変えると文章中の値も変わる。）