2. マルチメディアシステムのプラットフォームとアーキテクチャ

下 條 真 司[†]

1. ま え が き

現在のマルチメディア時代は映像、音声など、これまでのアナログメディアをコンピュータの扱えるデジタルの世界に取り込むことから始まっている。その昔「コンピュータは0/1しか扱えない、数字しか扱えない」といわれた。コンピュータで画像を扱えるといっても、モノリザを「@/.」などの記号で打ち出したものや、なんとなくギザギザの角張った絵を想像していた。しかし、コンピュータの数字の処理能力が上がるにつれ、このギザギザをどんどんどんどん細かくすることができ、「デジタルを限りなくアナログに近づける」ことができるようになってきた。デジタルで映像を扱うHDTVは映画の品質に迫りつつあるし、CDはあっという間にレコードを駆逐してしまっただけで、いまや、コンピュータでもこれらの映像や音声を取り込み、自由自在に扱うことができるようになってきている。

マルチメディアとは、一言でいって、このさまざまなメディアをすべてコンピュータで処理し、表示するということである。それによって映像や音声の、これまでとは違った利用法が生まれてくる。本稿では、マルチメディアを扱うためのコンピュータとその周辺、つまりマルチメディアプラットフォームとマルチメディアシステムのアーキテクチャについて述べることにする。

2. マルチメディアプラットフォームの変遷

マルチメディアのためのハードウェアはワークステーションから始まった。当初は映像や音声を取り込むために特殊なハードウェアを開発しなければならず、

ワークステーションはこの意味で都合が良かった。つまり、ワークステーションの持つ拡張性の良さ、標準的なバスアーキテクチャを持ち、OSとしてUNIXを採用していることが、専用のハードウェアを開発する基盤として役に立ったわけである。このような特殊なハードウェアを用いたマルチメディアシステムは主として実験的に構築された。MITのMuse¹⁾、Brown大学のIntermedia²⁾などのマルチメディアオーサリングシステムは、ワークステーションをベースにした高度な教育を行うために、マルチメディア教材の作成支援・提示のために作られた。大学のようにワークステーションが大多数であり、どこにでもある場合には、実用システムとしても構築された。しかし、専用システムとなるため、価格が高くなり、広まるには限界があった。

やがて、パソコンの時代がやってきた。ワークステーションで培った技術を高性能、低価格のパソコンに置き換えて専用ハードウェアが続々と発売され、価格も安くなった。マルチメディアパソコンとして真っ先に名をあげたのが、Apple社のMacintoshである。もともと、早くからウィンドウやマウスを採用し、モトローラ社の680XXのチップをCPUとしていたせいで、あまりパソコンらしくないパソコンであった。マルチメディアを行うためにはユーザインタフェースが命であり、とくにウィンドウを採用しているかどうか別かれ目になる。マルチメディアシステムでは複数のメディアの情報を混在して表示させることになり、そのためにはウィンドウ環境が必要になるからである。また、ウィンドウ環境とともに整えられる高度なグラフィックスライブラリーも、マルチメディアシステム実現には欠かせない。

これらの条件を備えていたMacintoshは、いち早くマルチメディアの世界に突入し、パソコンマルチメディアの花形となった。まず、音声や映像を取り込んでウィンドウに表示するボードが登場し、それにと

[†] 大阪大学 大型計算機センター

“2. A Platform and Architecture for Multimedia System”
by Shinji Shimojo (Computer Center, Osaka University, Osaka)

なってそれらを取り込んだオーサリングツールやDTV(Desktop Video)と呼ばれる、卓上ビデオ編集ツールのようなアプリケーションが続々と登場した。

映像や音声を取り込むアプリケーションが多数登場してくると、アプリケーション毎にこれらの取り扱いが異なることが問題になる。例えば、あるアプリケーションで作った映像を別のワープロに張り付けたりといった、カット & ペーストが行える必要がある。そのためには、アプリケーション間で統一したメディアの取り扱いができる必要がある。このため、Apple 社はいち早く QuickTime というアプリケーションで映像を扱うための枠組を提供した。これによって、マルチメディアアプリケーション開発にますます拍車がかかったのはいうまでもない。

パソコンのもう一方の雄である Microsoft/IBM/PC 連合軍は、当初出遅れたものの Windows というウィンドウ環境で着実に巻き返しをはかりつつある。QuickTime と同様な仕掛けとして“Video For Windows”を開発し、QuickTime との映像素材の相互乗り入れまではかっている。市場規模で圧倒しているだけに、本気を出すと思しいところである。

さて、ワークステーションとはいえば、出だしは早かったものの、その後伸び悩んでいる。ワークステーションが今ひとつ伸び悩んでいる原因は、パソコンのような統一したメディアの扱いがまだできないことである。これはひとえにワークステーションが、同じ UNIX とはいえマルチベンダーが参入しており、統一がとりにくいことに依存している。パソコンの世界は、かたや Apple 社、もう一方は Microsoft 社の寡占状態であり、思い通りの手が打てるわけである。

これまで、ワークステーションの世界では、できるだけ異なるベンダーの間でも共通の環境を提供することで、利用者が分散を意識しないことを実現してきた。これをオープンシステムという。X Window System はこの一例である。しかし、オープンシステムの欠点は複数のベンダーが合意に達するまでに時間がかかることである。X Window System の場合のように、大学という中立の機関が主導権を握っている場合はいいが、企業間で調整しようすると、それぞれの企業のエゴが出てうまくまとまらない。いま、ワークステーションにおけるマルチメディア基盤は、それぞれのベンダーによる独自の開発が始まったばかりで、とてもベンダー間の標準的なものを調整できる段階にはないのである。

こんな中で、ワークステーションでありながら、マルチメディアワークステーションとして他を引き離し

たのが、NeXT Computer であった。Apple 社を飛び出した Steve Jobs が作ったことで知られるこのコンピュータは、時代の技術をできるだけ低価格で提供するというコンセプトに基づき、当時は珍しかった大容量(640 M バイト)の光磁気ディスクを標準で装備するなど、それまでのワークステーションとは一線を画していた。より品質の良い音声を幅広い応用で扱うための DSP を備え、優れた描画能力を持つ Display PostScript を備えるなど、マルチメディアを扱うに適したワークステーションであるといえる。また、680 XX という比較的一般的な CPU のコンピュータを他と区別させるのは、そのソフトウェアである。マルチメディアを扱う複雑なアプリケーションのための環境として提供される NeXTStep は、アプリケーション間でマルチメディア素材を扱うための共通のインタフェースを与え、その開発環境である Interface Builder は質の高いウィンドウアプリケーションの開発を容易にする。その NeXT Computer が、はやばやとハードウェアの生産を中止し、プラットフォームを IBM/PC 系に変えたことは、今後のマルチメディアプラットフォームの行方を占う上で重要な出来事も知れない。

3. マルチメディアプラットフォーム

マルチメディアシステムは、マルチメディアプラットフォームの上に構築されている。マルチメディアとは、すべてのメディアをコンピュータで扱うことであるから、コンピュータがなければ話しが始まらない。しかし、コンピュータだけではマルチメディアシステムを作り上げることができない。ここでは、マルチメディアシステムのためのプラットフォームについて述べる。

3.1 マルチメディアプラットフォームの要件

マルチメディアのもうひとつの側面が「コンピュータの家電化」である。文字や数字しか扱えなかったコンピュータは、そういうものだけを必要とする一部の専門家のみにしか使われなかった。しかし、映像や音声を扱えるようになって、テレビやビデオ、ラジカセ、ゲーム機など、日常我々が目に触れる家電製品とあまり違わなくなってきた。よく、マルチメディアプレゼンテーションがテレビ番組とどこが違うのと比較されるのも、コンピュータがテレビに近づいてきたことに他ならない。逆にいうと、これまでコンピュータに触れることのなかった多くの人々が、マルチメディアを通じてコンピュータを使うことになる。こう考えてみると、マルチメディアプラットフォームの持つ本

質的な要求は「低価格、使いやすい、丈夫」といったことであろう。

さらに、システム構築に当っては以下のような点に注意する必要がある。

① 時間的なメディアを効果的に扱えること

マルチメディアシステムを従来のシステムと決定的に区別しているのが、映像や音楽など時間的なメディアを含んでいることである。このため、これらのメディア間の時間的な同期関係を表現し、実現しなければならない。

② 拡張性を有すること

マルチメディアシステムは音声、映像、アニメーションなどを含んだ複雑なシステムである。この複雑性を押えながら効率的なシステムにすることは、アーキテクチャの最大の課題である。また、新しいメディアが加わったり、メディアの変更に対して拡張性を持つ必要がある。このため、異なるメディアをシステムの中で統一的に取り扱えることも重要である。

3.2 システムアーキテクチャ

ここでは、マルチメディアシステムのアーキテクチャを大雑把に図1のようにとらえることにする。

(1) 外部表現モデル(シナリオモデル)

オーサリングシステムの場合、マルチメディアを組合せプレゼンテーションのストーリーを表現するためのモデルが必要である。電子メール・ニュースシステムの場合には、さまざまなメディアを含んだマルチメディア文書に対する表現モデルが必要である。このようなモデルを外部表現モデルと呼ぶ。ここには、どのようなメディアを使うかの指定や、用いられるメディア間の時間的な関係、空間での位置関係、1つのメ

ディアの表示から別のメディアの表示に移る時の効果(例えば、テレビで用いられるワイプなど)が記述される。したがって、このような記述は通常“シナリオ”と呼ばれることもある。また、この外部表現モデルはシナリオを保存し、配布したり、交換したりするためにも用いられる。

外部表現モデルは異なるプラットフォームでの相互交換にも用いられるため、目的に応じた標準化も進められている。CD-ROMによる出版を前提としたMHEG³⁾や音楽情報の表現から発展したHyTime⁴⁾などがある。また、AppleはIBMと共同でScript Xというハイパーメディア記述言語の開発を進めている。マルチメディア文書としては、通常のオフィス文書標準であるODAの拡張が考えられている。一方、Internet上の電子メールやニュースなどでは、既存のニュース配送システムとの整合性を考慮しながら、マルチメディアを含んだ通常の文字以外を送ることができるようにMIME(後述)が規定されている。

(2) 内部表現モデル(オブジェクトモデル)

外部表現モデルは、利用者が理解するためであったり、他のプラットフォームとの互換性のために決められている。また、あくまでもメディアをデータとして表現したものであり、処理が含まれていない。外部表現モデルを実現するために、処理も含めてモデル化したものが内部表現モデルである。

内部表現モデルとして“オブジェクト指向”を用いることで、データとそれに付随する処理を含めてオブジェクトとしてモデル化することができる。オブジェクト指向でモデル化する場合には、何をオブジェクトにするかと、オブジェクトどうしの関連をモデル化する必要がある。マルチメディアシステムの場合、目に見えるメディアをそれぞれオブジェクトとしてモデル化(メディアオブジェクト)し、オブジェクト間の関連を別のオブジェクト(シナリオオブジェクト)で管理するという方法が考えられる。また、オブジェクト指向の継承メカニズムを用いることで、上位に抽象的なメディアを表現するクラスを置き、個々のメディアをそのサブクラスにしてメディアの統一的な扱いが可能である。

(3) データベース

必ずしも必要ではないが、オーサリングシステムにおけるシナリオやメディア情報そのものを蓄積する場合には、ファイルシステムやデータベースが用いられる。通常、外部表現モデルをテキストなどの形でファイルシステムに蓄積するのが一般的である。内部表現モデルを蓄積する場合には、ファイルシステムやデー

図1 マルチメディアシステムのアーキテクチャ

データベースが取り扱える形にモデルを変形する必要がある、また、内部表現モデルが持つデータに付随する処理は管理できない。一方、オブジェクト指向データベースを利用することで、内部表現モデルをそのままデータベースに管理させることができる。この場合、データとその処理をまとめてデータベースに管理させることができる。

メディア情報そのものをデータベースに蓄積するのは、いくつかの問題を含んでいる。音声や映像データの場合、そのままでも大量の情報になる。また、映像データをそのままハードウェアなどに蓄積した場合、ディスクから取り出す速度が追いつかず、リアルタイムに再生できない可能性があり、何らかの圧縮が必要になる。このような長大データをリアルタイムにアクセスすることができなければ、マルチメディア情報そのものをデータベースに管理させることができない。

例えば、我々が開発しているマルチメディアプレゼンテーションシステムである Harmony の場合、マルチメディア情報そのものはファイルとしてファイルシステムに管理させ、ファイル名のみをデータベースで管理している。現状のオブジェクト指向データベースは、メモリー上に大きなオブジェクトのキャッシュをとることにより、長大データの効率的なアクセスを可能にしており、マルチメディア情報そのものもデータベースで管理させる可能性が出てきた。しかし、キャッシュの管理をうまくしないと莫大なメモリーが必要になってくる。また、メディア情報はファイルシステム、データベース、ウィンドウシステムなど、さまざまな関連する環境の中で別々にキャッシュされており、効率が悪い。もう一度、OS そのもののアーキテクチャを見直す必要がある。

(4) ツールキット/ウィンドウシステム

マルチメディアシステムは汎用性を高め、他のアプリケーションとの相互運用性を保つためにも、何らかのウィンドウ環境の下で作られる。IBM 系では Windows, Macintosh は独自のものを持っており、ワークステーションは X Window System, NeXT には Display PostScript がある。しかし、アプリケーションが複雑化するにつれて、ウィンドウシステムのすぐ上に直接アプリケーションを開発することは労力を必要とするため、あまり行われなくなってきた。通常は、ウィンドウシステムの上に“ツールキット”と呼ばれる高度な GUI ライブラリが用意されている。Macintosh では MacApp と呼ばれるライブラリがあり、X Window System の上には Motif がある。NeXT の NeXTStep も同様である。また、UNIX ワ

ークステーションでは Interviews⁵⁾, ET++⁶⁾といったツールキットもある。ツールキットには、ボタンやスクロールバーといった部品がライブラリとして用意されており、ツールキットを用いることで、ウィンドウを用いたアプリケーションを容易に作成することができる。

ただし、マルチメディアシステムにツールキットを用いる場合は、さまざまなメディアをツールキットの中で扱うため、ツールキットを拡張する必要がある。たとえば、映像をウィンドウの中で表示する場合、映像表示用の部品をツールキットの中で提供する必要がある。ワークステーションに接続される映像表示用のハードウェアの場合、X Window System の widget と呼ばれるツールキット上の部品がハードウェアベンダーによって提供されている。つまり、ハードウェア毎に提供されている部品が異なっている。パソコンの場合には、Macintosh には QuickTime, Windows には Video for Windows と呼ばれる枠組が OS ベンダーによって提供されており、アプリケーション間での統一的なメディアの扱いが可能である。

(5) OS/ハードウェア

マルチメディア、特に音声と映像を扱うためには通常特殊なハードウェアが必要であり、コンピュータとしてはハードウェアを接続するバスの拡張性に優れていることが要件となる。OS も、これらのハードウェアを簡単にとり込めることが重要であり、UNIX ではドライバと呼ばれるソフトウェアを組み込むことで、簡単に新しいハードウェアを接続することができる。

また、時間的なメディアを扱うためには、リアルタイム性も重要であり、複数メディアの時間的同期等を扱うためにはパソコンのようなシングルトask OS が都合が良く、UNIX のようなマルチタスク OS は不利である。ただ、パソコンもネットワークを本格的に扱うようになれば、いつまでもシングルトaskのうまみを享受しているわけにもいなくなる。

4. マルチメディアシステムのアーキテクチャの実例

ここでは、筆者がこれまで手掛けた2つのマルチメディアシステムを例にとり、そのアーキテクチャを見ていくことにする。一つは、UMIME⁷⁾というマルチメディア電子メールシステムであり、マルチメディア文書を扱うためのアーキテクチャを示している。もう一つは Harmony⁸⁾と呼ばれるマルチメディアプレゼ

ンテーションシステムである。

4.1 UMIME

ワークステーションやパソコンで画像、音声などが扱えるようになり、電子メールやニュースでこれらの情報を交換したいという要求が高まってきた。それにとともに、最近、Internet 上でやりとりされるマルチメディア電子メール・ニュースのための規格として MIME (Multipurpose Internet Mail Extension)⁹⁾ が制定されている。UMIME は X Window System 上で動作し、MIME フォーマットの電子メールやニュースを読み、マルチメディア情報を表示する MIME リーダーのひとつである。UMIME は従来の MIME リーダーにはない以下のような特徴を備えている⁷⁾。

- 1つのウィンドウ中でのテキストと画像・音声の混在した表示が行える。
- マルチメディア文書の自由なスクロール、マルチフォントでの表示が行える。
- さまざまなプラットフォームへの移植性があり、メディアに対する拡張性を有している。

以下では、まず MIME について説明し、UMIME のアーキテクチャについて述べる。

(1) MIME とは

MIME とは、RFC 1341 で規定されている Internet 上でマルチメディア電子メール・電子ニュースを扱うための規格であり、既存の電子メール配送システムに対する互換性を保つため、従来の電子メール・電子ニュースの規格である RFC 822 をヘッダを用いて拡張している。また、電子メール・電子ニュースの内容に構造を持たせることができ(図2参照)、非 ASCII 文字は ASCII 文字に符号化することによって、バイナリデータ、テキストや画像、音声が含まれるメッセージの取り扱いを行うことができる。また、フォント属性を持つテキストを表現するフォーマットとして、**richtext** を用意し、マルチフォントテキストを扱うことができる。

(2) UMIME のアーキテクチャ

図3に UMIME のアーキテクチャを示す。UMIME は、MIME フォーマットのマルチメディア文書を入力とし、それを画面上に表示する。したがって、外部表現としては MIME を用いていることになる。システムは MIME を読み込み、表示するために VObjectText と呼ばれるマルチメディア文書を表現するオブジェクトに変換する。これが内部表現モデルに相当する。VObjectText は通常の文書をマルチフォントで表現し、自由にスクロール等が行えるのに加えて、VObject と呼ばれるマルチメディアの個々の

```
From: wakabaya@ise.osaka-u.ac.jp
Subject: test
Mime-Version: 1.0
Content-Type: multipart/mixed;
boundary="Boundary"
```

```
--Boundary--メディアの境界
Content-Type: text/richtext

<center>
<bigger><bigger><bigger><bigger>
<bigger><bigger><bigger><bigger>
A Sample of <italic><bld> MIME
</bld></italic> Format
(略)

--Boundary--
Content-type: image/gif
Content-Transfer-Encoding: x-uue

begin 644 kiri-h.gif
MlTE&.#=A0 'P .< $^G;D_I_=GD$80AW]_-^
(略)
&>OGO>_SV
end
(略)
--Boundary--

--Boundary--
```

図2 MIMEによりマルチメディア化された電子メールの例

図3 UMIME のシステムアーキテクチャ

メディアを表現するオブジェクトを混在させることができる。UMIME ではこの機能を利用し、MIME より読込んだ文書以外のメディア、音声と画像をそれぞれ VObject のサブオブジェクトである LipIcon、Bitmap クラスのオブジェクトに変換し、VObjectText 中に混在させてマルチメディア文書の表示を行っている(図4)。各メディアを VObject のサブクラスとして独立に作ることでメディア毎に拡張が可能であり、また新しいメディアも容易に組み込める。また、テキストから VObject の変換のために Converter というクラスを用意することで、gif、JPEG など、さまざまなフォーマットの画像を表示できる(口絵①)。

図 4 'UMIME' の設計の概要

最終的な表示は、ツールキットとして ET++ と呼ばれるスイス銀行で開発された C++ のクラスライブラリーを用いている⁹⁾。ET++ は X Window System 上に実現されており、通常の Xtoolkit などのツールキットに比べて以下のような特徴を持っている。

- アプリケーションフレームワークと呼ばれるオブジェクト指向に基づいた設計法を用いている。このため、利用者は半完成のアプリケーションの足りない分を補っていくという形でプログラミングを行い、開発期間が短縮するとともに、すっきりとした設計が行える。
- 基本的には Macintosh の MacApp をベースに作られており、結果として出来上がったアプリケーションのユーザインタフェースが素晴らしい。
- ウィンドウシステムや OS から独立に作られており、SunView, X, NeWS の上で動いている。
- Sniff と呼ばれるプロジェクト管理、クラスブラウザなどのプログラミング環境を別に用意している。

UMIME でも、アプリケーションフレームワークと VObjectText, VObject, Converter といった ET++ のクラスを用いて短期間で作られている。ハードウェアプラットフォームとしては、SUN sparc IPX を用いており、音声の再生のために内蔵音声チップ利用し、特別なハードウェアを使用していない。このため、ET++ を拡張し、音声を扱うためのクラスを加えている。

4.2 マルチメディアプレゼンテーションシステム Harmony

次に、マルチメディアプレゼンテーションシステムである Harmony のアーキテクチャを取り上げる。Harmony は、ビデオ、音声、アニメーションなどの

図 5 Harmony のシステムアーキテクチャ

時間的なメディアを容易に取り扱い、システムの拡張性を保つことを目標としてオブジェクト指向に基づいて設計されたマルチメディアプレゼンテーションシステムであり、以下のような特徴を持っている。

- シナリオモデルとしてオブジェクト指向モデルとハイパーテキストモデルを融合した“ハイパーオブジェクトモデル”を用いている。
- オブジェクト指向データベースを利用したマルチメディアシステムである。
- 各メディアはそれを扱う独立したプロセスとして作られており、拡張性に優れている。

(1) Harmony のアーキテクチャ

Harmony のシステムアーキテクチャを図 5 に示す。Harmony は外部表現モデルとしてハイパーオブジェクトモデルというものを使用している。ここでは、素

図 6 ハイパーオブジェクトモデル

図 7 Harmony のアーキテクチャ

材となる各メディアをオブジェクト(メディアオブジェクト)として表現し、ノードとしてハイパーテキスト上に配置し、それらをリンクによって関連づけることでメディア間の時間的関係等を表現する(図6)。利用者はハイパーオブジェクトモデルに基づきシナリオを表現する。

ハイパーオブジェクトモデルは同時に内部表現モデルでもある。ただし、実現は複数のプロセスで行われており、ユーザインタフェースを司る Harmony/UI、ハイパーオブジェクトモデルにしたがって、リンクの航行等を行う Harmony/LM から構成される“Harmony サーバ”と、個々のメディアの表示等の処理を行うメディアプロセスから構成されている。

Harmony はシナリオや各メディア情報の蓄積・管理のためにオブジェクト指向データベース(OODB)を用いており、Harmony サーバ、メディアプロセスをクライアントとしている(図7)。

先に述べたように、OODB を利用することにより、内部表現モデルをそのままデータベースに蓄積するこ

とができ、関係データベースやファイルシステムのように特別な処理を用意する必要がない。また、Harmony のような複数プロセスのアプリケーションを構築する際には、プロセス間で共有するデータをデータベースに管理させるとにより、プロセス間でデータをやりとりする必要がなくなる。

Harmony は映像表示装置である Parallax 社の X Video を備えた SUN4/IPX 上で開発され、XVideo 上の JPEG 圧縮ハードウェアを用いることで映像をハードディスクに蓄積し、リアルタイムで再生を行っている。このため、プレゼンテーションを行う際にレーザーディスクなど、外部の映像装置を必要としないポータブルなプレゼンテーションが可能である。ツールキットとしては UMIME と同じ ET++ 用いているが、映像を扱うために特別なクラスを用意している。Harmony は JPEG による映像を初め、アニメーション、テキスト、音声、画像を扱うことのできるプレゼンテーションシステムである(口絵②)。

5. む す び

本稿では、マルチメディアプラットフォームについて述べるとともに、マルチメディアシステムのアーキテクチャについて述べた。マルチメディアという言葉だけが先行する時期は終りを告げ、ゲームや教育、出版など、身近なところからマルチメディアが着実に始まっている。その中で最も注目すべきは、光ファイバによる 500 チャンネルにも及ぶ巨大な CATV である。この 500 チャンネルを利用する鍵となる技術がマルチメディアであると筆者はにらんでいるのだが、いかがだろうか。

(1993 年 7 月 12 日受付)

〔参 考 文 献〕

- 1) M. E. Hodges, R. M. Sasnett and M. S. Ackerman: "A Construction Set for Multimedia Applications", IEEE Software Magazine, 22, 1, pp. 37-43 (Jan. 1989)
- 2) B. J. Haan, P. Kahn, V. A. Riley, J. H. Coombs and N. K. Meyrowitz: "IRIS Hypermedia Services", Communications of The ACM, 35, 1, pp. 36-51 (Jan. 1992)
- 3) ISO/IEC JTC1/SC2/WG12, "Information Preprocessing-Coded Representation of Multimedia and Hypermedia Information Objects", Presented at Working Document S. 4 (Sep. 1991)
- 4) ISO/IEC JTC1/SC18/WG8, "Information Technology-Hypermedia/Time-based Structuring Language (Hy Time)", Draft International Standard, ISO/IEC DIS 10744 (1991)
- 5) M. A. Linton and P. R. Calder: "The Design and Implementation of Interviews", USENIX Proceedings (1987)
- 6) A. Weinand, E. Gamma and R. Marty: "Design and Implementation of ET++, a Seamless Object-Oriented Application Framework", Structured Programming, 10, 2 (June 1989)

- 7) 若林進, 下條真司, 西尾章治郎: “マルチメディアメール・ニュースの表示システム設計および実装”, 情報処理学会マルチメディアと分散処理研究会 (July 1993)
- 8) S. Shimojo, K. Fujikawa, T. Matsuura, S. Nishio and H. Miyahara: “Architectural Issues on Multimedia Presentation System Harmony”, in The Object Oriented Approach in Information Systems (Oct. 1991)
- 9) N. S. Borenstein and N. Freed: “MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the Format of Internet Message Bodies”, RFC1341 (June 1992)

しもじょう しんじ
下條 真司 昭和 61 年, 大阪大学基礎工学部大学院博士課程後期課程修了。昭和 61 年, 大阪大学助手, 平成元年, 同大学大型計算機センター講師, 平成 3 年より, 同助教授となり, 現在に至る。分散処理システムの設計開発, CSCW の研究に従事。近年, オブジェクト指向データベースのアーキテクチャ, マルチメディアシステムへのオブジェクト指向の応用などに興味を持つ。工学博士。

1993 年 12 月号「最近の音声処理技術」小特集予定目次

【ふ お ー か す】		パイオニア	山本武夫
【小 特 集】	1. 展 望.....	早 大	白井克彦
	2. 音声の生成と知覚.....	ATR	河原英紀
	3. 基礎技術		
	3-1 概念からの音声合成.....	KDD	山本誠一
	3-2 連続音声認識.....	NEC	渡辺隆夫
	4. 応用技術		
	4-1 音声処理と翻訳.....	ATR	浦谷則好, NHK 江原暉将
	4-2 テキスト合成.....	リコー	北川博雄
	4-3 声質変換.....	NHK	都木 徹
	4-4 話者認識.....	NTT	古井貞照
【研 究 動 向】	最近のコントロールルームの音場設計.....	日東紡音響エンジニアリング	福満英章・小林照二
	情報機器の障害者インタフェース.....		宇都宮大 鎌田一雄
【講 座】	カラー画像工学の基礎と応用 (最終回):		
	あとがきに代えて—視覚と画像のインタフェース—.....	大日本印刷	日下秀夫
【技術解説 (規格特集) ④】	CS 音声 PCM 放送.....	郵政省	篠原聡兵衛
【論 文】	Bézier 曲線の高速生成とアンチエイリアシング.....	大阪市大	浜 裕光・奥本孝昭
	奥行き方向への視線移動時の眼球運動特性の加齢効果.....	ATR	魚森謙也・山田光徳
	景観シミュレーションのための簡易情景モデル構築と動画生成の一手法		
		中部電力	曾山 豊, 名 大 安田孝美・横井茂樹・鳥脇純一郎
	高感度 CdTe 光導電膜を用いたビジコン型 X 線イメージセンサ		
		浜松ホトニクス	富田康弘・高林敏雄・永田幹夫・河合敏昭, 静岡大 畑中義式
【これすばんでんす】	カラービデオ補正の色再現への影響について.....	千葉大	阿部昌人・池田宏明
	ハイビジョン静止画システムを用いたプレゼンテーションシステム		
		三洋電機	岡本 浩・杉浦洋治・原田茂樹・森本庸一・渡辺金雄
	光ファイバネットワークを利用した日米間デジタル映像伝送.....	東京放送	荒井清実
【研 究 速 報】	高効率符号化を用いたホログラフィック 3 次元画像の圧縮法		
		日 大	吉川 浩・丹治英一郎
【会議レポート】	PCSJ '93.....	東 大	相澤清晴