

## 解説

# ヒューマンインタフェースをつかさどる BTRON 仕様の実現と評価†



楠木好明†† 清水正博††

## 1. はじめに

パーソナルコンピュータやワークステーションの OS は、①コンピュータ資源の有効利用（マルチプロセスなど）、②応用共通機能の共有（グラフィックスなど）、③操作環境の向上（マルチウィンドウなど）などを実現する方向に発展してきた。

①についてはシングルタスク OS (MS-DOS) からマルチタスク OS (OS/2, UNIX) への流れがある。②については、CGI, GKS などのグラフィックスに関する機能や、かな漢字変換と辞書からなる日本語処理機能を OS に取り込み、③については、UNIX における X window や OS/2 における PM (Presentation Manager) などの操作環境を実現するものがある。

しかし、これらの OS 機能のベースになる情報管理モデル、すなわち、操作モデル/データモデル/実行モデルは、統一的概念で設計されたものではない。よく普及したもの（するであろうもの）を OS の機能として組み込んで発達してきたものと考えられる。

BTRON (Business TRON) 仕様では、OS 全体のコンセプトに実身/仮身モデルという統一モデルをもち込んだ。この実身/仮身モデルを、ファイルシステムやデータ交換フォーマットにおけるデータモデルだけでなく、操作環境やプログラムを実行するための実行環境にも統一モデルとして適用した。このことにより、ユーザからみた操作モデル、アプリケーションで用いるデータモ

デル、アプリケーションを実行するときの実行モデルを効率的かつ明快に具現化している。本文では、BTRON 仕様における特長的な概念とその実現について述べる。

## 2. BTRON 仕様の情報管理モデル

BTRON 仕様の情報管理モデルである実身/仮身モデルでは、情報の本体を「実身 (Real-Object)」と呼び、この実身は「仮身 (Virtual-Object)」と呼ばれるタグによって参照される<sup>1)</sup> (図-1)。

パーソナルコンピュータやワークステーションの情報管理モデルは、操作モデル/データモデル/実行モデルから成り立つと考えられる。

操作モデルとは、ユーザの指示の方法と、コンピュータ側からの応答による対話系の抽象化モデルのことで、この部分の洗練度がそのまま「使いやすさ」の指標になる。BTRON 仕様では、ユーザは画面に表示されている仮身を選び操作を行うが、その操作は実身を指している仮身に対する操作であり、仮身を操作することによりネットワーク構造のファイルを自由に操作できる。データモデルは、データの関係構造モデルであり、BTRON 仕様では、実身/仮身モデルと呼ばれ

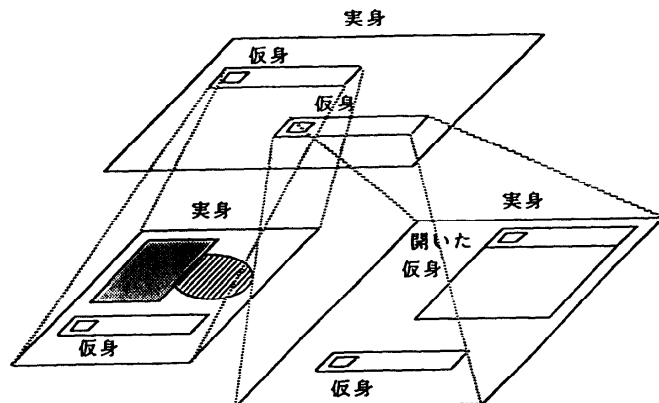


図-1 実身/仮身モデル

† The Realization and the Evaluation of Operating System Based on BTRON Specification which offers the Unified Human Interface by Yoshiaki KUSHIKI and Masahiro SHIMIZU (Information Systems Research Laboratory, Matsushita Electric Industrial Corp.Ltd.).

†† 松下電器産業(株)情報システム研究所

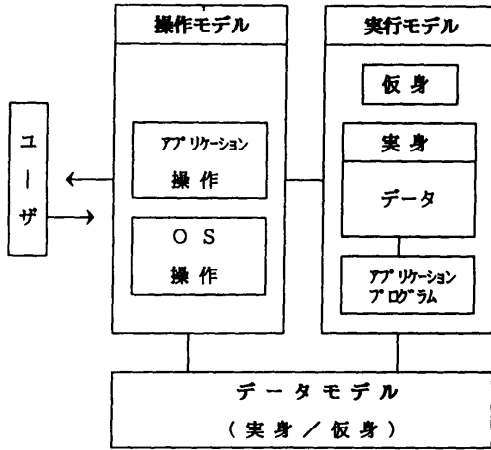


図-2 実身/仮身モデルによる BTRON 仕様情報管理モデル

るネットワーク構造になっている。実行モデルとは、アプリケーションプログラム（機能）とそれが取り扱うデータの関係を示す抽象化モデルである。BTRON 仕様では、仮身に対して起動操作がなされるが、起動されるプログラムは仮身が指している実身に結合したアプリケーションプログラムである（図-2）。

実身には、実行機能付箋と呼ぶアプリケーションプログラムを指す付箋を貼りつけることにより実身とアプリケーションプログラムの対応をとっており、データとプログラムを分離し、複数のプログラムが同一のデータを共有する環境を提供している。従来はこの関係は「コマンド+ファイル名」という命令で機能するものであり、このコマンドとファイル名の組合せをユーザが記憶していなければならなかった。

BTRON 仕様では、この三つの操作モデル（特に仮身の操作）/データモデル/実行モデルに共通のコンセプトとして実身/仮身モデルを採用している。唯一のモデルで統一されることにより、データ構造を理解し、機能を実行する操作手順を憶えるとき、わかりやすく、明快なものになっている。

### 3. 操作環境における実身/仮身モデル

BTRON 仕様の操作環境は、実身/仮身モデルを基本思想として統一されている。すなわち、画面に表示された仮身をポインティングデバイスで「2度押し」か、また「実行メニュー」の中から該当するプログラムを選択することにより実身が実行される。実身の実行とは、実身のデータを処理するアプリケーションの実行を意味し、この実行により、実身データの表示・

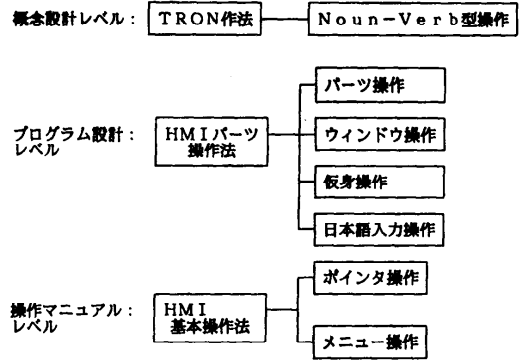


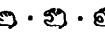
図-3 BTRON 仕様操作体系

処理・転送などが行われたりする。実身の実行は仮身操作により実現され、①異なるアプリケーション間、②アプリケーションと OS 間において、仮身操作の手続きが統一されている。

BTRON 仕様の操作環境は階層的に三つのレベル「概念設計レベル」、「プログラム設計レベル」、「操作マニュアルレベル」に分けることができる。

「概念設計レベル」においては、TRON 作法、すなわち、BTRON 仕様における操作原則、設計指針を規定としている。TRON 作法とは、「Noun-Verb 型の操作指示」、「可能なかぎりコンピュータで判断すること」、「即時応答」などの操作方式設計の原則を規定したものである。

「プログラム設計レベル」では、吟味された操作手法を提供し、操作統一するため HMI (Human Machine Interface) 機能を標準化し、BTRON 仕様組み込んだ。これが HMI パーツでウィンドウ操作、仮身操作、日本語入力操作のほかに、ボリュームやスイッチなどのパーツがある。約 20 種類のパーツでかなりのアプリケーションの操作がカバーでき、アプリケーションプログラム作成の容易性に加えて、プログラムサイズの削減や開発効率の向上などの効果がある<sup>3)</sup>。

「操作マニュアルレベル」では HMI 基本操作法を提供する。HMI 基本操作とは、すべての操作の基本となるポインティングデバイス (PD) の操作のことである。PD の操作には、1 度押し、2 度押し、押し続けなど、5 種類ある。また、PD の位置や操作によって画面上のポインタの形状が自動的に変更される。この基本形状は 8 種類(  )あり、この形状によって操作可能な動作が分かる。これにメニューの選択方法を加えると HMI 基本操作のすべてである。この HMI 基本操作さえ憶えれば、

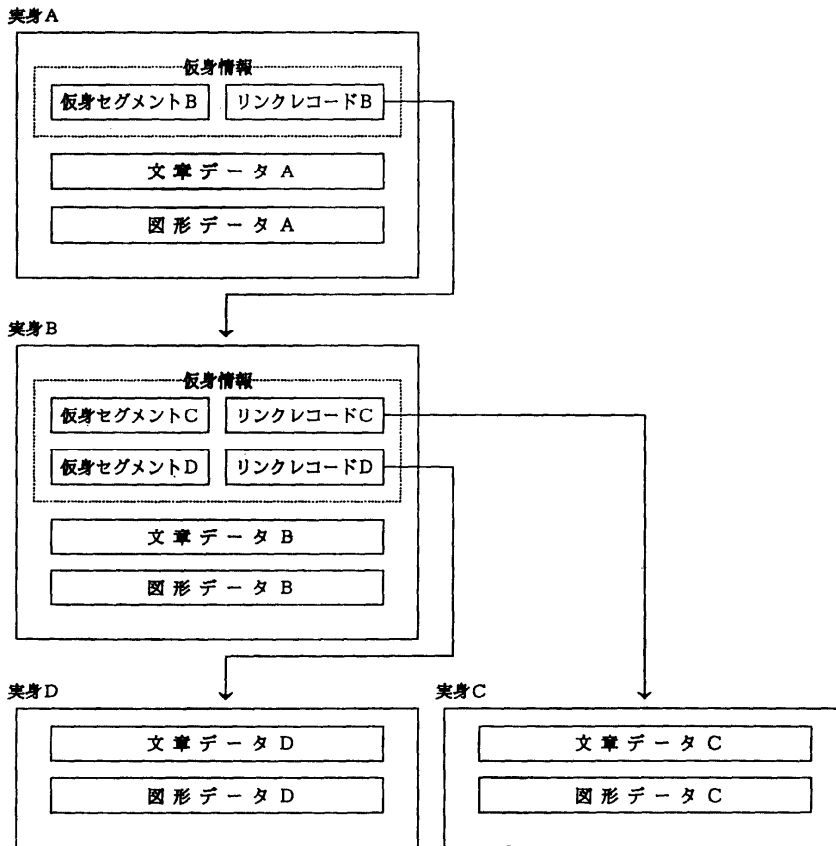


図-4 実身/仮身データモデル

BTRON 仕様ではマニュアルはなくとも、基本的な操作を行うことができる。

#### 4. 実身/仮身データモデル

実身/仮身データモデルでは実身の中に埋め込まれた仮身が他の実身を指し、また、複数の仮身が同一の実身を指すことができるような、ネットワーク型のファイル構造となっている<sup>4)</sup>。

実身/仮身データモデルにおいて、実身は「文章/図形等データ」、参照しているアドレス情報を格納する「リンクレコード」、仮身の位置、サイズ、色などの表示情報を格納する「仮身セグメント」などからなるマルチレコードで構成されている。

たとえば、図-4 において、実身 A に含まれている仮身 B は、実身 A が実身 B を参照していることを表しているタグであり、リンクレコード B により実身 B と

リンクをとっている。実身 B においても仮身を含むことにより、実身/仮身ネットワークを構成できる。

実身/仮身データモデルにおいて、実身/仮身の複写・削除は次のように実現できる。

##### (1) 仮身の複写

あるアプリケーションで参照している実身を異なるアプリケーションで参照したい場合、仮身情報のみを複写すればよく、実身は複写する必要はない (図-5 (a))。

##### (2) 新版作成

仮身と、仮身が指している実身から構成されているネットワークシステム全体を複写することを新版作成といい、この操作により実身/仮身ネットワークの複写が可能となる (図-5 (b))。

##### (3) 実身の複写

実身は仮身を開くことにより表示でき、その実身を

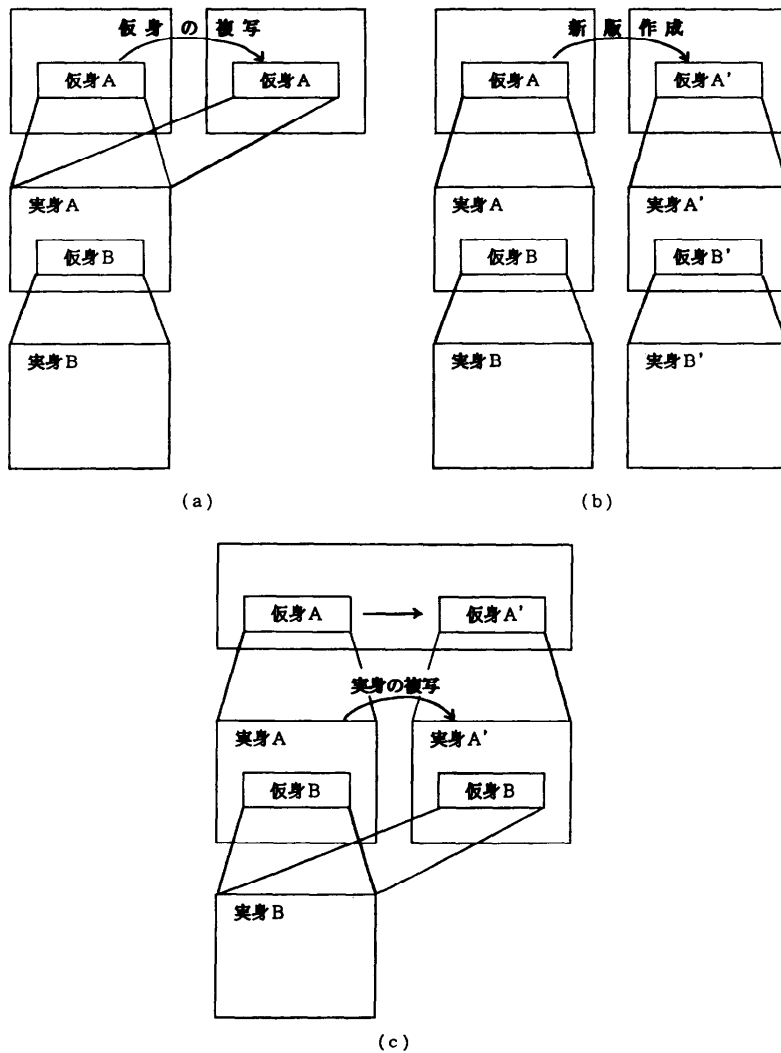


図-5 実身/仮身の複写

異なる名前で保存することにより、実身の複写が行える。実身の複写にともない、仮身が新規に作成される(図-5(c))。

(4) 実身/仮身の削除

従来の木構造のファイルシステムでは、ファイルの削除は単純であった。

実身/仮身モデルでは、実身は複数の仮身から参照されることが可能であり、仮身を消去した場合にすぐには実身を消去できない(図-6)。削除可能な実身はどこからも参照されていない実身であり、参照されなくなった時点で、自動的に削除される。

5. 実行環境における実身/仮身モデル

アプリケーションプログラムを提供する場合、従来のコマンド指向型 OS では実行コードファイルのみを提供するのが一般的であるが、BTRON 仕様では「実行環境」を提供する。たとえば、「原稿用紙」と「エディタ」、「郵便受け」と「通信プログラム」などの「データ」と「プログラム」の組合せが「実行環境」に相当する。

BTRON 仕様では情報管理モデルとして実身/仮身モデルを用いており、仮身(たとえば「郵便受け」)

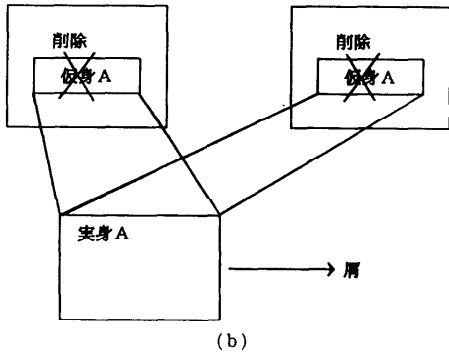
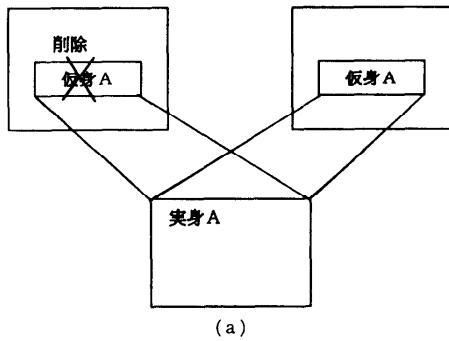


図-6 実身/仮身の削除

に対して起動操作がなされると、参照する「実身」であるデータに結合しているアプリケーションプログラム（たとえば「通信プログラム」）が起動される。このアプリケーションプログラムは、実行機能付箋でリンクがとられ、実身に貼ることができる。

実行環境における実身/仮身モデルは次のように実現されることになる(図-7)。

- ① ディスプレイ上のウィンドウ1に表示されている仮身Aに対して起動操作を行う。
- ② 仮身Aにリンクされている実身Aに付与されている実行機能付箋Xに対するプログラムXが起動される。
- ③ プログラムXは実身Aの仮身情報、データをウィンドウ2に表示し、処理を行う。

また、アプリケーションプログラム内の変更可能な静的データ（たとえば

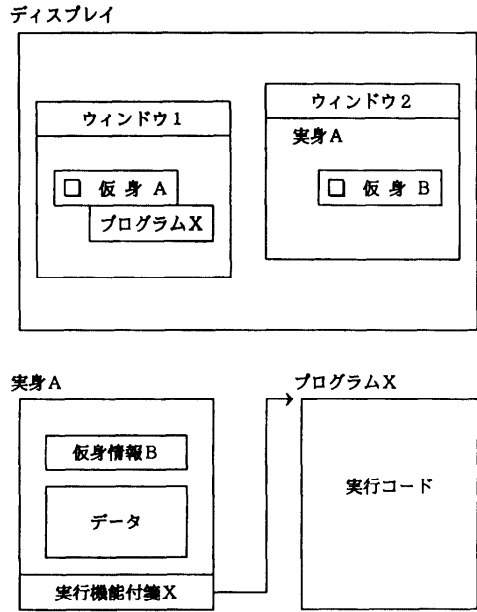
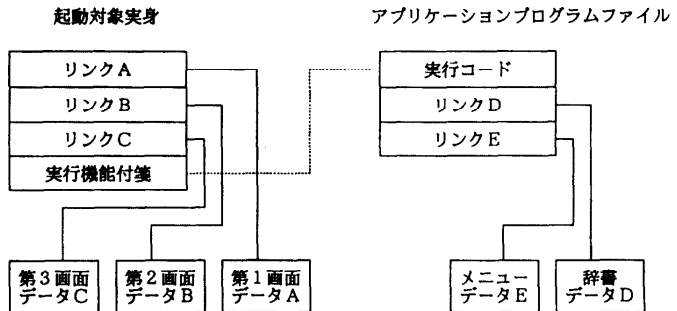
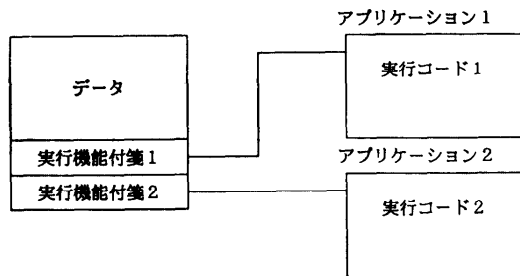


図-7 実行環境モデル



(a) リンクレコード機能を活用したアプリケーションの設計例



(b) 複数の実行機能付箋

図-8

メニューデータや辞書データ)を実行コード部分から分離して独立したデータとしてもたせることが可能である。このことにより、データとプログラムのリンクを実現しながら、同時に独立な関係を維持することができ、ダイナミックにメニューデータや辞書データを交換できる(図-8(a))。

また、BTRON仕様では起動対象実身に対して、実行機能付箋を自由に付与したり削除でき、また複数の実行機能付箋も付与できる(図-8(b))。これにより、一つのデータを複数のアプリケーションで共有し、処理の範囲を広げることが可能となる<sup>6)</sup>。

ユーザは起動操作時にメニューに表示された複数の実行機能付箋の中から、任意の実行機能付箋を選ぶことができる。

## 6. データ交換フォーマット TAD

BTRON仕様では、アプリケーション間のデータ交換を実現する目的で、TAD (TRON Application Data Bus)と呼ぶデータフォーマットを規格化している<sup>6)</sup>。

TADデータは大きく文章データと図形データに分かれる。注目すべき点は、文章データを構成する文章要素として図形データが存在し、逆に図形データを構成する図形要素として文章データが存在している点である。すなわち、文章と図形は互いにネスティングしながら混在することが可能な構造になっている。また、画像セグメント(画像データ)も文章・図形データの要素となる。

このような文章/図形/画像などのマルチメディアを効率良く扱うため、TADでは1ファイルに複数レコードを格納するマルチレコードファイルを用いている。従来パソコン用OSにおいてシングルレコードファイルが主体であったのと比較すると、マルチメディアを非常に効率良く扱える。

文章データには文字コード、図形データ、画像データのほかに、データのレイアウト処理を制御する付箋を含みうる。データの受け手はこの付箋情報を解釈することにより一次的な順序関係しかもたないデータ例から、二次元的なレイアウトイメージを形成できる。

図形データには、すでに二次元的な位置関係がデータ中に記述されている。たとえば、ページ内にレイアウトされている文章・図形などのデータを印刷イメージで蓄積交換するような場合には図形データとして交換する。

TADは、基本的なデータ交換を保証する一方で、属性や制御情報を付箋という可変長のセグメント形式で表現することにより、アプリケーションの処理の容易性、アプリケーション依存データの保持を実現している。また、TADでは、ODAが定める文書構造を用いた高度なデータ制御はできないが、ODAにない特長として従来のワードプロセッサとの整合性が良い点や、変換処理系がコンパクトな点が優れている。

一方、TADでは文字、図形、画像のほかに、動画、音楽も扱いやすくなっている。動画データ形式を表現するAV-TAD、動画を実身として扱いウィンドウ表示するビデオマネージャ、音楽のデータ形式を表現する音楽TAD、音楽TADを「演奏」するサウンドマネージャが試作検討されている。

TADにおける動画拡張部の、BNF表現は、次のようになっている<sup>7),8)</sup>。

[TAD データ]=《管理情報セグメント》

[TAD 本体]

[TAD 本体]=[文章データ][図形データ]

[動画データ]

[動画データ]=[動画要素並び]

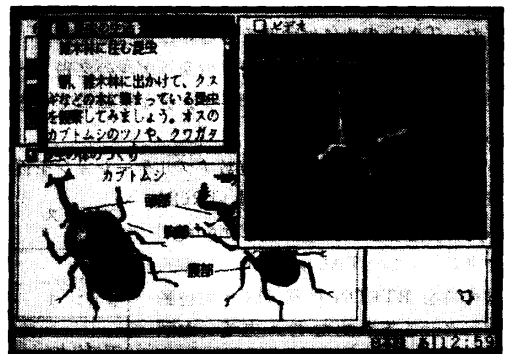


図-9 動画を扱う BTRON 仕様実験機とその画面例

[動画要素並び]=《空》[動画要素]

[動画要素並び]

[動画要素]=《動画開始セグメント》《動画セグメント》《動画終了セグメント》

動画データを含む TAD を、画面に表示すると、文章や図形に動画の仮身が置かれ、この仮身を開くと動画が現れる。

動画の周囲に存在する文章や図形は、エディタやワープロソフトで、自由に編集することができるし、動画を表示するウィンドウも文章や図形の中で自由に移動することができるようになる。図-9 に動画を扱う BTRON 仕様 OS 搭載実験機とその画面例を示す。

### 7. アプリケーションの融合

BTRON 仕様では複数のアプリケーションが密接に関連をもち「融合」する。操作の統一に加え、データ互換、「お返し」を用いた自動実行により「融合」が実現される。

BTRON 仕様では、TAD (TRON Application Data Bus) によってアプリケーション間のデータ互換を維持する。あらゆるアプリケーションが、TAD を共通バスとして、システムの一時的なデータ保存領域であるトレイや実身 (ファイル) を介して対話する。また、TAD の中には仮身を含むことが可能である。このことは TAD 自身が実身/仮身モデルの文書データ構造 (動画、音楽を含むマルチメディア構造でもある) であることを意味する。

BTRON 仕様 OS 上の複数アプリケーションを逐次的に実行させるのが TACL (TRON Application Control-flow Language) である<sup>9)</sup>。この TACL により、定型的な作業を BTRON 仕様 OS 上で自動実行させる基本的な考え方を「お返し」と呼ぶ。このように、新規に追加されたアプリケーションは、最初から統合された考えの中に組み込まれることになる。

以上により、BTRON 仕様 OS 上の新しいアプリケーションは「融合」し、多様性と拡張性に富んだ機能を増殖させることができる。

### 8. インプリメンテーション

BTRON 仕様の基本コンセプトの一

つであるリアルタイム処理は、ユーザの操作に対するコンピュータの高速応答を実現することである<sup>10),11)</sup>。このため、BTRON 仕様ではイベント駆動型の応答体系を基本としている。つまり、ポインティングデバイスによる仮身に対する実行指示操作イベントにより、実身データに対してアプリケーションが自動的に起動され、次のイベントの待ち状態になる。ここでなんらかのイベントを得ると、すぐにイベントに応じた処理が実行され、再びイベント待ち状態になる。以上の動作の繰り返し実行をすべてのアプリケーションが容易に実現するため、次のような工夫をしている。

(1) ユーザの操作 (PD 操作やキー入力など) はすべてイベントとしてシステムのイベントキューに蓄えられる。(イベント管理)

(2) どのアプリケーションがイベントをイベントキューから取り出すかを OS が制御する。(ウィンドウマネージャ)

(3) 取り出したイベントがパーツやメニューなどの標準的な操作を指示するものであれば、その処理を OS にまかせることができる。(メニューマネージャなど)

(4) 各イベントに対応する非定型的な処理を記述するだけで、アプリケーションを構成できるライブラリを用意している。(アプリケーションライブラリ)

図-10 に、ユーザがメニュー操作を用いた場合のイベントの流れを示す。

以上のような機能をインプリメントした BTRON/286 仕様 OS のソフトウェア機能一覧を表-1 に、試

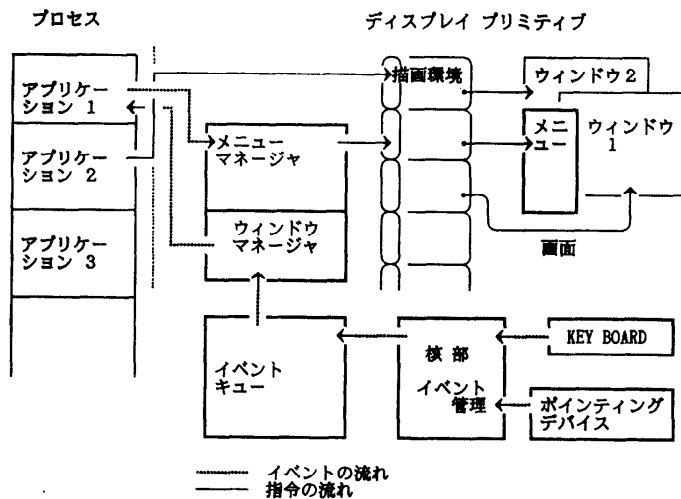


図-10 イベントに対するリアルタイム応答

表-1 BTRON/286 仕様 OS の機能一覧

核	プロセス管理機能	シングルユーザ/マルチプロセスの管理、プロセス間のメッセージ通信、セマフォによる同期/排他制御、プロセス間のグローバルデータ管理などの機能をもつ
	メモリ管理機能	プロセスのローカルメモリ領域の管理、全プロセスの共有メモリ領域の管理、プロセス領域のスワッピング処理を行う
	ファイル管理機能	ネットワーク構造のファイルシステムをサポート、ファイルのアクセス管理を行う
	イベント管理機能	キーボード/ポインティング・デバイスとのインタラクティブ操作のためのイベント駆動機能、ソフトウェアで発生するイベントの管理
	デバイス管理機能	入出力機器を統一的に管理する
	時計管理機能	実時間管理を行う
	システム管理機能	例外処理登録、共通プログラム管理、外核実現のためのシステムコール拡張を行う
	描画機能 ディスプレイ ミティブ	ディスプレイ上に図形や文字を表示するための基本的な機能の集合
	かな漢字変換機能	日本語処理機能
	外核	マルチウィンドウ管理
メニュー管理		メニューの登録・表示・操作(選択機能)の管理機能
コントロールパーツ管理		CRT 上に表示される機能部品(スイッチなど)を用いてユーザとシステムとの会話を行う機能
パネル管理		機能部品により構成された矩形の一時的な表示領域を用いて、ユーザとシステムとの会話を行う機能
トレイ管理機能		ウィンドウ間のデータの移動・複写を実現するための共有データ保管管理機能
データ管理機能		HMI 機能などで使用する固定的なデータをファイルとして定義し、そのファイル内のデータを統一的に取り扱う機能
テキスト入力 ミティブ		文章の入力、編集をサポートする機能
フォント管理		文字フォントの生成および管理機能
実身/仮身管理機能		実身/仮身の処理をサポートする機能

作機を図-9 に示す。BTRON/286 仕様は BTRON 仕様のサブセットで、TAQL<sup>9)</sup>、多国語処理<sup>12)</sup>、分散処理などは含まれていない。ハードウェア性能の向上と歩調を合わせながら、今後機能拡張することがで

きる。

BTRON/286 仕様 OS は、大きく分類して、OS 核、外核 (HMI 機能)、システムアプリケーションの 3 層から構成される<sup>10)</sup>。システムコール数は OS 核部、外核部があわせて約 400 である。BTRON/286 仕様 OS の実現に際して工夫した点は、

(1) HMI パーツなどのライブラリの OS への動的組み込み機能をもつ。

(2) リアルタイム応答の実現のために、キーボードやポインティングデバイスからの入力をイベントという概念にまとめ、イベント駆動型のプログラミング方式を採用している。

(3) 高速に大容量ファイル (画像データなど) が扱えるよう、ディスク上で連続な領域 (最大 2GB) が確保できるファイル管理機能 (最大 128 TB) をもつ。

(4) デバイスドライバの動的拡張機能を有する。

(5) イベント処理を中心とした HMI システムと、それに連動して稼動するアプリケーション用プログラムテンプレートをもつ。

などであり、操作体系統一や実身/仮身ファイルシステムなどの豊富な機能を実現しながらも、リアルタイム応答を実現している。

## 9. 評 価

BTRON 仕様の特長であるネットワーク構造の実身/仮身ファイルシステムでは、木構造のファイルシステムに比べてソフトウェア構造が複雑になる。したがって、BTRON 仕様では、ファイルアクセスのオーバ・ヘッドがどの程度になるかが問題になる。

ここでは、BTRON/286 仕様の試作 OS と、木構造ファイルシステムをもつ MS-DOS をインテル 80286 を CPU とする同一ハードウェア、フロッピディスクシステム上で動作させた場合の評価結果の概略について述べる。

ファイルのオープン・クローズ時間について、BTRON/286 仕様で通常使用されるリンクレコード (ファイルの連結番号) によるオープン・クローズ時間と、MS-DOS のファイル名によるオープン・クローズ時間について比較すると、BTRON/286 仕様 OS は MS-DOS の約 1.5 倍の時間を要している。これは、シングルタスク、木構造の MS-DOS に対して、マルチタスク、ネットワーク構造の BTRON/286 仕様 OS のほうがファイルシステムの負荷が大きい



めと考えられる。

また、ファイルのリード・ライト時間は、標準入出力関数を用いて 1k バイト単位で行うと、BTRON/286 仕様 OS と MS-DOS ではほぼ同じ時間であった。また、BTRON 仕様の特長であるマルチレコードファイルを用いると、一連の文字ストリームからなるシングルレコードファイルである MS-DOS の場合に較べて、0.5 倍の時間で済んだ。以上の結果から、実身/仮身ファイルシステムは、木構造ファイルに対して、従来の 1 ファイル 1 レコードの形態のファイル処理はやや劣るものの、1 ファイルマルチレコードの形態のファイル処理は十分な性能が得られ、現在普及している木構造ファイルシステム相当の実用性が確認できた。

## 10. まとめ

実身/仮身モデルに基づく BTRON 仕様のアプリケーション思想を中心にインプリメンテーションの体験をふまえて解説した。BTRON 仕様はこれ以外に、ハードウェア仕様、キーボード仕様、LAN 仕様などがあるが今回は割愛した。

BTRON/286 仕様 OS については、現在開発・評価が続けられており、操作性やデータ互換機能の重要度に対する認識が深まるにつれ、実用化の期待も高まっている。BTRON 仕様は広大な概念をもち、多様性に富むだけに、高性能 CPU と、ビットマップ高速転送が可能な高解像度ディスプレイを有する高機能ワークステーションにおいて、さらに威力を発揮するものと考えられる。

終わりに、日頃から BTRON 仕様についてご指導いただいている東京大学坂村健助教授、BTRON 専門委員会各位ならびに当社関係各位に深謝します。

## 参考文献

- 1) 坂村: BTRON のファイル管理システム, 第 1 回リアルタイム OS-TRON 電子情報通信学会, 研究会資料, pp. 2-21 (1987).
- 2) 坂村: BTRON における統一的操作モデルの提案, 情報処理, Vol. 26, No. 11 (1985).
- 3) 小林, 竹之内, 榑木: BTRON 仕様による HMI のソフトウェア構造, 第 1 回リアルタイムアーキテクチャ電子情報通信学会, TRON 研究会資料, pp. 31-39 (1987).
- 4) 岩村, 石田, 今井, 榑木: 実身/仮身モデルのインプリメンテーション, 第 3 回リアルタイムアーキテクチャ, 電子情報通信学会, TRON 研究会資料, pp. 42-55 (1987).
- 5) 三村, 増田: 実身/仮身モデルに基づくアプリケーションプログラム設計手法, トロン協会, トロン技術研究会, Vol. 1, No. 2 (1988).
- 6) 松瀬, 坂村: アプリケーション用データ交換用規格 TAD の特徴について, トロン協会, TRON 技術研究会資料, Vol. 1, No. 2 (1988).
- 7) Kajimoto, K. Shimizu, M. and Kushiki, Y.: A Study on Video Manager of the BTRON Specification, TRON Project 1988, Springer-Verlag (1988).
- 8) Kajimoto, K. et al.: New-Media Document (New Doc) and Dynamic Navigation on the BTRON Specification, Proc. of 34th IEEE COMPCON (1989).
- 9) Sakamura, K.: TACL: TRON Application Control-Flow Language, TRON Project 1988, Springer-Verlag (1988).
- 10) 安藤, 今井, 榑木: BTRON 仕様 OS 核部の 80286 へのインプリメンテーション, 第 1 回リアルタイムアーキテクチャ, 電子情報通信学会, TRON 研究会資料, pp. 22-30 (1987).
- 11) 榑木, 坂村: BTRON 仕様による 80286 オペレーティングシステム, 情報処理学会, マイクロコンピュータ研究会資料 45-4 (1987).
- 12) 久保田, 榑木, 榑木: 多言語環境における日本語処理の実現, 情報処理学会, マイクロコンピュータ研究会資料 49-3 (1988).

(平成元年 3 月 13 日受付)