

高機能ワークステーションの基本技術

棚橋 賢一 (東芝青梅工場)

1. はじめに

ハードウェアの高性能化・高密度化・低価格化に伴い、ワークステーションと呼ばれる多種の機器が登場してきている。この中でも、用途によりエンジニアリングワークステーションとオフィスワークステーションに大別されているが、いずれもビットマップディスプレイを中心として、マンマシンインタフェースを重視した設計となっている。

ここでは当社が実現している、よりユーザー層の広いオフィスワークステーション用途の高機能ワークステーションについての設計思想や基本方式技術について、ハードウェア・ソフトウェアの両面から紹介し、今後のシステム応用についてもまとめてみた。

2. 高機能ワークステーションのシステム方式

以下には、当社の分散処理コンピュータや新世代のオフィスコンピュータとしてのトータルOAプロセッサに接続された高機能ワークステーションのシステム方式について説明する

2.1 設計思想

高機能ワークステーションでは、高級なマンマシンインタフェースを実現するために、次の特徴を持っている。

- (1) ビットマップディスプレイによる画面上での文字図形イメージなどのマルチメディア表示
- (2) マルチウィンドウを利用した優れたマンマシンインタフェースとコンピュータ利用形態の革新
- (3) マルチメディア情報をベースとした複合文書の作成と統合化ソフトウェア
- (4) ドキュメント転送を含む本体とのプロトコルの拡張

2.2 ハードウェア方式

2.2.1 ハードウェア構成の概要

高機能ワークステーションのハードウェアアーキテクチ

ャーはシステムバスを中心とした各種の専用用途をもつ複数のマイクロプロセッサが効率良く処理できる方式をとっている。(図1参照)

ハードウェア構成図

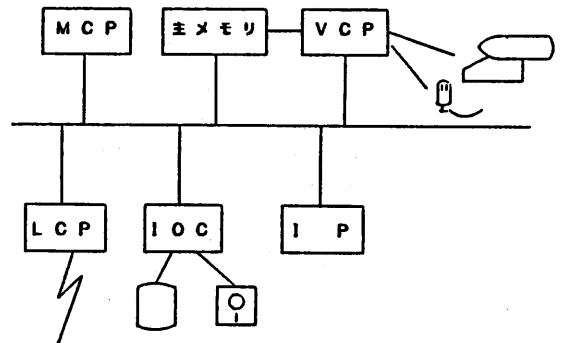


図1. 高機能ワークステーションのハードウェア構成

(1) MCP (メインコントロールプロセッサ)

ワークステーション制御の中心となるプロセッサであり、アドレス空間は16MBのCPUを使用している。メモリ方式はマルチウィンドウ制御のため、ダイナミックに生成・消滅するウィンドウデータを効率的に処理するために、ページング方式を採用している。そしてマッピングテーブルは、ディスパッチング時の効率化のため、複数装備している。

(2) VCP (ビデオコントロールプロセッサ)

ビットマップメモリを高速に制御するために、専用のコントロールプロセッサを装備している。本プロセッサでの処理機能は次のとおりである。

- ① 主メモリとビットマップ、フォントマスクROMとビットマップメモリ間の高速転送のためのビットムーバーの制御 (BITBLT)
- ② ジオメトリック (図形) 表示
- ③ リアルタイムカーソル制御を含むポインティングデバイス (マウス) 制御

主メモリは、MCPとVCPで共有できるように2ポートとなっており、いずれのプロセッサからも高速にアクセス可能となっている。

(3) IP (イメージプロセッサ)

イメージ処理の高速化のために、主として次の機能をもっている。

- ① イメージの圧縮・伸長 (MH, MR, MMR方式可)
- ② 密度変換
- ③ 拡大・縮小
- ④ 切り出し

(4) LCP (ラインコントロールプロセッサ)

本体との通信を制御するプロセッサであり、基本となるデータ処理系 (コード系) の転送と、マルチメディア転送を含めたプロトコルを処理している。

(5) その他

磁気ディスク、フロッピーディスクなどを制御する I/OC (I/Oコントロールプロセッサ) がある。

上記は本ワークステーションの基本モデルについて機能分担を示したものであり、CPUの高速化、用途の違いにより各プロセッサの機能を統合化したモデルもある。

2. 2. 2 ビットマップメモリ制御の概要

マルチウィンドウ/マルチビューポートを実現する基本的なハードウェアとして、ビットムーバーがある。ビットムーバーは語境界を意識しないで、ソース→デスティネーションのデータのムーブを行なう機構であり、転送処理中には、拡大・縮小・回転・反転などの操作を行なうこともできる。

ビットムーバーの転送は、以下の方向で可能である。

- ① 主メモリ ← → 主メモリ
- ② 主メモリ ← → ビットマップメモリ
- ③ 漢字パターンメモリ → 主メモリ
- ④ 漢字パターンメモリ → ビットマップメモリ

ビットマップメモリは、任意のビット位置からアクセス可能であり、物理画面上のビューポートを任意の位置に任意の大きさで定義可能としている。

2. 3 ソフトウェア方式の概要

マルチウィンドウ制御やマンマシンインタフェースの優れたソフトウェアを実現するためには、リアルタイム性を持つマルチタスク方式のカーネルが必要となる。

またメモリ管理として、動的に生成・消滅する論理的な画面情報やビューポート管理のために、ページング機構を利用して、使用者やアプリケーションソフトウェアから要求される各種のメモリ要求を効率よく処理可能としている。

さらに、複数のタスクによりこれらのマルチウィンドウ制御を実現するために、タスク間通信機能を用意している。

3. マルチウィンドウ機能及び特徴

本システムでのマルチウィンドウシステムは、次の概念をベースとして構築されている。

(1) 論理画面

論理画面は、従来物理画面に表示されていた画面を文字コード・ジオメトリックコマンド・イメージなどに分け、一つの論理画面情報として定義するものである。したがって、大きさやデータ量などは、物理画面の制御とはまったく無関係に設定可能である。

(2) ウィンドウ/ビューポート

論理画面上に設定された矩形のことをウィンドウと称している。一方、物理画面上に定義されている矩形のことをビューポートと称し、ウィンドウとビューポートは基本的には一対一に対応して存在している。

ウィンドウは、一つの論理画面上に複数個指定することができ、さらに重なりあったウィンドウ設定も可能としている。また、ビューポートの重なりは、プライオリティにより上下関係が設定される。

上記概念に基づいて、次の機能が実現できる。

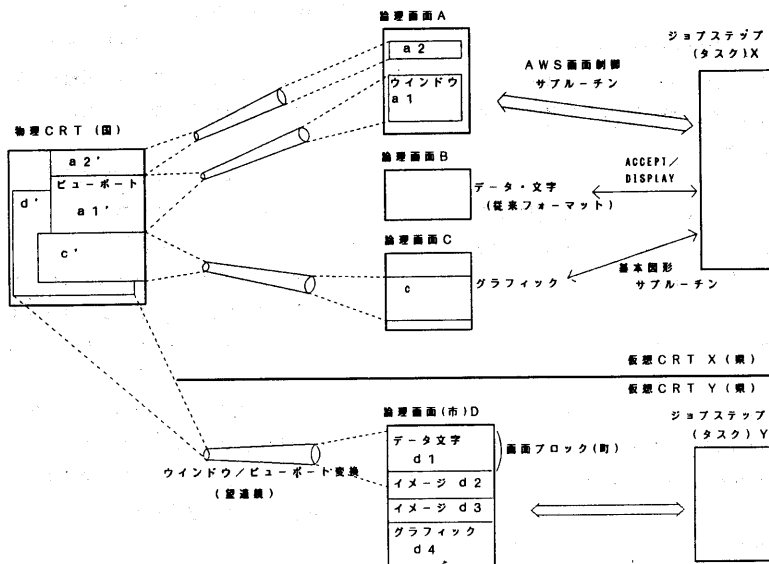


図2. マルチウィンドウの機能概念図

3. 1 仮想端末

仮想端末とは、従来一つの物理端末を意識して作成されたアプリケーションプログラムをまったく変更することなしに、同一の物理端末に複数プログラムがアクセスできる機能である。図2では、タスクX、Yは、まったく独立に作成されたプログラムであるが、同一の物理CRTを利用していることを示している。

3. 2 複数論理画面

一つのアプリケーションプログラムは、独立した複数の論理画面を設定し、アクセスすることができる。図2では、論理画面A、B、Cが設定でき、図2ではa1、a2、cの3つのウィンドウが設定されている。論理画面Bに対しては、ウィンドウが、未だ設定されていない状態を示している。

また各種論理画面は、文字・イメージ・ジオメトリックの3種の情報を表示できる。論理画面Dは、この状態を示している。

3. 3 ウィンドウ/ビューポート処理

ウィンドウ/ビューポートは、次のような設定により、各種の論理画面処理を実現できる。

(1) スクロール

ビューポートを一定としたとき、ウィンドウの位置を移動することにより、従来端末での上下スクロール・左右スクロールが実現できる。

(2) ビューポート移動

ウィンドウを一定として、ビューポートを移動することは、単に、物理的な表示位置を変更する処理となる。

(3) ビューポート優先度

ビューポートは重さなりが許されるので、複数のビューポート間の上下関係は優先度により決定する。優先度を変更することにより、表示の上下関係を逆転させることができる。

(4) 拡大・縮小

ビューポートとウィンドウはそれぞれ拡大・縮小できる。ビューポートとウィンドウを同時に同じ比率で拡大縮小した場合は、表示されている文字サイズは同じで、表示領域が拡張されることになる。

ビューポートを一定として、ウィンドウを大きくした場合は、画面上は縮小表示されることになる。逆にウィンドウを小さくしたときは、画面上は拡大表示されることになる。

(5) 回転

ウインドウに対応した物理画面上のビューポートの位置設定で、X軸、Y軸方向を逆に設定すれば、表示上は回転することになる。回転角度は90度単位で指定することが可能である。

上記のウインドウ/ビューポート処理は作成・削除を含めアプリケーションプログラムレベルからオペレータからの操作といずれでも可能としている。

3. 4 複合文書処理でのマルチウインドウ

今まで説明したビットマップディスプレイとマルチウインドウ機能をベースとした統合化されたソフトウェアの一例として複合文書処理機能について説明する。

(1) マルチメディアの融合

オフィスにおける文書は、文書、ビジネスグラフ、表、図形、イメージなどから構成される。これらの複合文書は、ISOにて標準化されつつあるが、本サブシステムでもこの方向にそった文書データ構造を採用している。

文書データ構造は、ページ・フレーム・ブロックという概念で、各メディアの境界を明確にしている。ブロックは矩形の領域であり、単一のメディアを含んでいる。フレームは関係するブロックを結び付けるものであり、階層構造をとることもできる。これらフレーム、ブロックの組み合わせによりどんな複雑な文書でも取り扱うことができる。

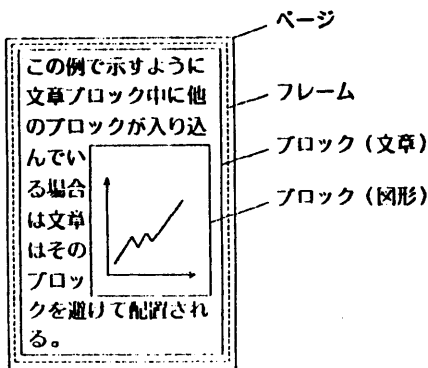


図3. フレーム、ブロック構造

(2) 優れたマンマシンインタフェース

ビットマップディスプレイをベースとしたマルチウインドウ機能により、以下の優れたマンマシンインタフェースが採用可能となった。

- ① デスクトップモデルの採用
- ② オブジェクト先行型の操作
- ③ アイコンによる表示選択方式
- ④ プロダウンメニューによる操作指示
- ⑤ プロパティシートによる属性変更
- ⑥ エラーメッセージ・ヘルプメッセージ

これらの機能は、オーバラップを可能としたマルチウインドウ機能とマウスによるビットマップディスプレイのペインティング機能により、実現できるのである。

図4では、その画面例を示す。

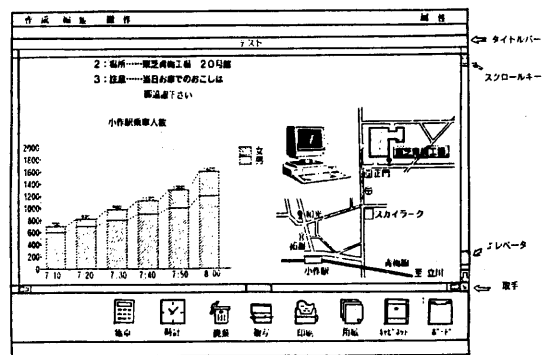


図4. 画面表示例

(3) 表示と印刷の一致

ビットマップディスプレイをベースとしているため、画面表示上もプリンタと同様に行ピッチ、文字ピッチを設定できる。さらに、ビットムバによる拡大・縮小機能により拡大文字やルビ・添字などの縮小文字も画面表示可能となる。

また、図形やイメージと文章との相対的位置関係で自由に設定できるため印刷結果と同じものを画面表示で確認しながら文書を作成できる。縦書きについても、回転機能を利用することにより、実現できる。

(4) リアルタイム制御

マンマシンインタフェースの中心は応答性のよいマウス制御である。リアルタイム性を要する以下の処理の実現が必須技術となっている。

① マウスカーソル制御

一般に一定のサンプリングタイム毎に位置を知ってカーソル表示するわけだが、人間工学的に二つの工夫がされている。

a. 近傍処理

文章中の文字列の選択や図形編集時の図形選択時などに必須の処理である。これらの処理はたんに物理位置の制御だけでなく、近傍処理が可能な効率的な論理的データ構造が必要である。

b. 移動速度調整処理

マウスカーソル移動には、2つの排反する要求を満たす必要がある。

- ・ 画面上を大きく素速く移動させたい。
- ・ 詳細に位置付けしたい。

この二つの要求を満たさせるためには、単純サンプリングではなく、マウスの移動速度を考慮し、マウスカーソルの移動距離を決定すればよい。即ち、速い移動では大きくカーソルを移動し、遅い移動では小さくカーソルを移動させる。また、アイコン移動などのためにカーソル形状の可変性は当然の技術である。

② リアルタイムリバース処理

プルダウンメニューや文書編集時などに選択しようとしている対象物を反転させる機能であり、これも対象となるデータ構造の設計が大きく影響してくる。

③ ラバーバンド機能

ビューポート作成、移動時や図形作成時にリアルタイムの連続するグラフィック表示を基本とするものであり、システム的には、かなり狭い範囲での高速処理が実現できればよい。但し、ビューポートをこえるかどうかのチェックは、必要な処理である。

4. システム技術

上記まではディスプレイ中心の狭義のワークステーション技術について説明してきたが、以下にはその応用システムについて説明する。

4.1 ドキュメント管理システム

日々増大する図面資料等を管理するドキュメント管理システムはオフィスワークステーションの一つの応用例である。これには二つのアプローチがある。

(1) イメージ処理中心

既存のドキュメントからの移行を重視した運用形態であり、以下の技術が必須となる。

- a. 蓄積デバイスとしての光ディスク制御
- b. イメージ編集表示を行なうワークステーション
- c. ユーザニーズに応じた各種検索及びドキュメント管理ソフトウェア
- d. 印刷装置やドキュメント蓄積装置とワークステーションとのネットワーク技術

本システムでは系内を大量のイメージデータが通過することになり、高速のLANを中心とするネットワーク技術とデータ量を抑えるための圧縮・伸長性能も重要なポイントとなる。

(2) マルチメディア中心

前記の複合文書処理をベースに新規にドキュメント管理システムを構築する運用形態である。以上の基本技術に加え、以下の技術が必要となる。

- e. マルチメディアを編集できる複合文書処理
- f. 複合文書データ構造を他のデータ処理システムと結合する技術

上記ドキュメント管理システムは、運用規模によってホスト中心の集中管理と完全な水平分散管理を両極として、階層的なドキュメント管理まで、各種の形態が今後展開されて行くであろう。

4.2 仮想端末の利用システム

仮想端末機能は、OSでサポートされているので、その下で実行されるアプリケーションプログラムやユーティリティプログラムなどは特にマルチウインドウを意識する必

要がない。次のような利用例が考えられる。

(1) プログラム開発者

- ・ ソースプログラムの編集プログラムとコンパイラのエラーリストの同時表示
- ・ 実行デバッグ時（特にシンボリックデバッガ）とプログラム編集

(2) ファイル管理者

- ・ ボリューム中のファイル使用状態の表示とファイル割付けや削除プログラムの同時操作

(3) ホスト接続とローカル処理

最近のコンピュータは、ホストの端末エミュレーション機能と、ローカル処理機能を合わせてもつものが多い。従って2種の機能を仮想端末機能によって同時に実行することが可能となる。

また、他のOSやハードウェアが可能ならば、複数のホストと同時に接続することも可能となろう。

(4) 事務データ処理

事務用のデータ処理パッケージとしては、各種の組み合わせ利用が考えられるが、以下に一例を示す。

- ・ 売上げ伝票発行中に、在庫の問い合わせをする。
- ・ 売上げ伝票発行中に、マスタ登録されていない商品に登録する。

これらは、一般に業務パッケージの複雑さを軽減し、各種機能のパッケージを運用により、効果的に関連付をもたせた利用形態を実現できるため、ソフトウェア生産性の問題にも一つの手助けとなることも考えられる。

4.3 複数論理画面の応用システム

複数論理画面とは、一つのアプリケーションプログラムがまったく独立に設計した画面を利用できる機能である。以下に応用例として考えられるものを示す。

- ・ 業務パッケージ、会話ユーティリティなどのヘルプガイダンス。
- ・ ソースプログラム編集では、データ記述と手続き部を同時に表示編集したり、関連するモジュールを同時に参照したり、変更したりする。
- ・ 印刷帳票と同じ画面形式でデータ処理を行なう業務パッケージ。これは帳票の一部分を複数ウィンドウとして設定することで可能となる。

5. おわりに

ビットマップディスプレイを中心としたマルチウィンドウ技術とその応用例を中心に高機能オフィスワークステーションの基本技術を述べてきたが、これらワークステーションの利用はまだ緒についたばかりであり、ワークステーションの応用技術やシステムは、今後のコンピュータ利用の大きな変革の柱であることを確信し、さらに強化に努めたい。

参考文献

- [1] 情報処理学会誌
「特集：高機能ワークステーション」(84.2)
- [2] 技術処理学会第30回全国大会
棚橋・中条「高機能ワークステーションの設計思想」
他9編 P.1809~1828
- [3] 第28回オペレーティングシステム研究会(85.9)
棚橋「オフコン/分散プロセッサとワークステーションの機能分散」
- [4] 第27回マルチメディア通信と分散処理研究会(85.9)
柳「複合文書処理における文書構造とプロトコル」