

## 3J-7

鈴木十三男、高橋快昇、錦織正明  
富士通株式会社

### 1. はじめに

今日パソコンやワークステーション(両者合わせてWSという)のソフトウェアの持つ使い勝手の良さと適用業務の範囲の広さにより、オフィスでのWSの利用が盛んになっている。さらに、スタンドアロンだけでなく、ホストコンピュータ(以下ホストという)との連携によるWSの利用が急速に広まっている。

オフィスでの作業環境の延長にWSを位置付けて、文書の作成・配布・保管といった文書処理を各人が円滑に行うことのできる統合文書処理システムを開発した。本システムでは流通OSであるMS-DOSのソフトウェアを利用できるFMRシリーズをWSとし、より効率的かつ効果的な作業環境を実現するために、WSのディスプレイ画面上に文書処理の操作環境を用意している。

本稿では、本システムの目標、実現方法及びプログラム構造について述べる。

### 2. システムの目標

MS-DOSの豊富なアプリケーション・ソフトウェアを利用できる作業環境をWS上に作り、個人の日常作業を手軽に支援できるシステムとして、次の目標設定によるシステム設計を行った。

- ・文書の取り出し、保管、配布などの作業をオフィスでの作業の流れのイメージで操作できる環境
- ・個人ごとのWS上の作業環境の容易な構築
- ・煩雑な操作を必要としないWS機種間での文書交換

### 3. 実現方法

文書処理のフレームワークを①基本資料の電子化、②情報検索(参照)、③文書作成、④文書作成支援(図形作成、定型文書作成)、⑤情報蓄積、⑥文書出力・印刷、⑦文書配布・広報・収集としてまとめ、本フレームワーク全てに対応できるよう、システムの拡張性を考慮したシステム設計・開発を行った。

MS-DOS上で動作するために、本システムのプログラムのメモリサイズを小さくするようなプログラムの構造、機能分割を行っている。

#### (1) アイコン、ウィンドウによる統一した操作ビュー

オフィスの机やキャビネットなどをWSのディスプレイ画面上(デスクトップ画面)に仮想的に配置した環境を用意した。すなわち、オフィスで使う文書、封筒、キャビネット、アプリケーション・ソフトウェアなど(オブジェクトという)に対する基本的な操作(開く、閉じる、入れる、出す、捨てるなど)で目的とする処理を実現している。

このために、デスクトップ画面での操作の制御部と個々の処理部とにモジュール分割を行い、コマンド化し、メモリの占有量の細分化を図っている。

個々の処理部では、属性(入れ物、デバイス、アプリケーション・ソフトウェアなど)により分類したオブジェクトを表現する絵文字(アイコン)に対応した固有の処理を行う。

制御部については、アイコン属性に応

じた処理（ソフトウェア起動、ウィンドウ表示など）をコマンド化し、必要なコマンドだけをメモリ上にロードさせるようにした。

また、グラフィック・VRAMの一部をビットマップに使用し、重なり合ったウィンドウの表示範囲の管理することにより、表示ウィンドウの個数の増加に伴うプログラム・メモリサイズの圧迫を回避している。

(2) ホストと連携したメーリング、共用キャビネット

ホストの文書処理基盤システムと連携してキャビネット、メールサービスを行うために、ホストとの通信処理、上記サービスの要求処理及びファイリングやメーリングの操作をそれぞれ独立したモジュールに階層化して制御している。

デスクトップ画面から利用者の選択した機能に対応して、デスクトップ制御部が空きメモリ量を管理しながら、不要モジュールだけのメモリからのアンロードと必要モジュールのロードの制御を行う方式をとっている。

この方式により、メモリ容量の許容範囲内で、別文書の表示ウィンドウを開いたままでのメーリングやホストキャビネット登録を可能となった。

(3) 各種アプリケーション・ソフトウェアの登録・起動環境

使用したいアプリケーション・ソフトウェアを自分のデスクトップ画面に登録・利用する方法を簡易にできる手段を用意した。

OAツールを文書処理や部門内のデータ処理などの機能的に分類した道具とみなして、それを格納する道具箱にすでに用意してある未定義アイコンにそのソフトウェアを割りあてることにより、デスクトップ画面からのソフトウェア起動を可能としている。

具体的には、簡易エディタを組み込んだアイコンの環境設定モジュールを呼出

し、アプリケーション・ソフトウェアに対応するアイコンの表題・図柄パターン番号・起動コマンド列を入力して設定を行う。

4. プログラム構造

プログラム単体のサイズと表示ウィンドウ数などの操作状態によるメモリサイズへの影響を可能な限り少なくするように、モジュール化を図った構造とした。

特に、制御部は常駐する部分を少なくし、レスポンスを保証する範囲で、それぞれの制御処理もコマンド化して必要な時点のみロードするように設計・開発した。

また、本システムで使用するWS、ホストの各ソフトウェアで共通に使用する情報の管理・運用を柔軟に処理できるように、各ソフトウェアの動作環境をプログラムと独立させたプロフィール情報として、システム又は利用者単位で定義できるようにした。

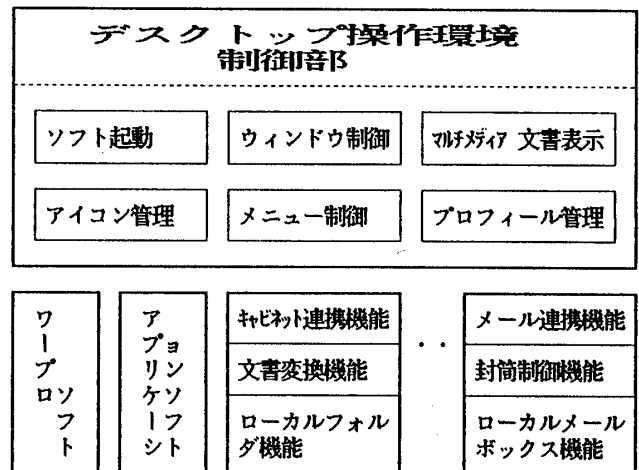


図 文書処理システムの構造

5. おわりに

本システムは当初の目標を達成したが、文書処理のフレームワークの基本部分だけの実現であり、今後フレームワークにそった展開を進める予定である。

また、各種WSへのデスクトップ環境の実現も併せて行っていく。