

オブジェクト指向による OS/omicon 第4版の構築

加藤 泰志, 森永 智之, 早川 栄一, 並木 美太郎, 高橋 延匡

東京農工大学 工学部 電子情報工学科

1. はじめに

近年、表示一体型タブレットの登場により、手書きインタフェースが注目を集めている。しかし、既存のシステムでは手書きインタフェースが持つデータの多様性、多義性、応答性などに対応することが困難である。そのため OS/omicon 第4版[1]では、手書きインタフェースの柔軟性、多様性に対応するための枠組みとして、仮想的な紙である電紙を提供する。この電紙を実現するための機構として、OS/omicon 第4版では、2次元アドレス、ワンレベルストア、ダイナミックリンクなどの機能を使う。しかし、これらの機能は、手書きデータだけではなく、様々なデータを扱うための枠組みとなるものである。

2. 本研究の目的

OS/omicon 第4版では、仮想的な紙である電紙を提供しているが、電紙には次の機能が必要である。

- (1) 電紙の持つ属性（型）を応用プログラマが定義できること
- (2) 電紙同士の属性の対応関係の管理と、電紙の属性変換の機構を備えること
- (3) データの表現を可視化するための対応関係とその実行機構を備えること

このような機能を持つ電紙を一つのオブジェクトとすることにより、電紙の持つデータとそのデータを操作する手続きをカプセル化することができる。また、システム標準で用意されている電紙クラスを継承することにより、ユーザが新しい機能を持った

電紙を定義することができる。

このように、OS/omicon 第4版には、オブジェクト指向が有効であると考えた。そのため、本研究では OS/omicon 第4版にオブジェクト指向の概念を導入する。また、手書きデータをはじめとするデータの多様性、多義性に対応するため、システムの拡張性を確保することを目的とする。

3. 設計方針

このシステムの設計方針は次の通りである。

- (1) 言語C++を用いてマイクロカーネルより上の資源管理部を記述する。これにより、OS/omicon 第4版にオブジェクト指向の概念を導入する
- (2) OS/omicon 第4版の特徴である電紙をプログラミング言語で抽象化し、プログラマに提供する
- (3) ダイナミックリンクの機構を用いてシステムの拡張性を確保する

また、このシステムの記述に言語C++を用いることにした理由は、言語C++はシステムを記述できる、ポインタが使える、言語Cからの移行がしやすいの3点である。なお、ポインタは電紙間のリンクを張るために必要となる。

4. 設計

4.1 電紙とオブジェクトの関係

このシステムでは、電紙をクラス構造により定義し、一つの電紙は電紙クラスのインスタンスになる。これにより、電紙に対するインタフェースを統一することができ、電紙のデータとそれをアクセスするコードの一体化などのオブジェクト指向の利点を使える。また、OS/omicon 第4版では 電紙 = セグメント であるが、このシステムでは 電紙 = セグメン

An Object Oriented Method for the Construction of the OS/omicon Version 4.

Yasushi KATO, Tomoyuki MORINAGA, Eiichi HAYAKAWA, Mitarou NAMIKI and Nobumasa TAKAHASHI
Tokyo University of Agriculture and Technology

ト = オブジェクト, となる。

4. 1. 1 電紙のユーザ定義

応用プログラミングに必要となる電紙は, そのプログラムに依存する。そのため, ユーザが新しい電紙クラスを定義できる必要がある。このシステムでは, システムで標準で用意される電紙クラスを継承した新しい電紙クラスを定義することにより, ユーザが新しい電紙クラスを定義することができる。

4. 1. 2 電紙の型変換

手書きの文字が書かれている電紙を認識系にかけてコードに変換するなどの電紙の型変換はオブジェクトの型変換と見なせるから, 言語C++の構文でキャスト演算により行なうことができる。

4. 1. 3 電紙の共有

このシステムでは, 電紙はシステムの資源の一つとなり, アプリケーション間で電紙を共有することができる。このようにアプリケーション間で共有される電紙は, 永続的なオブジェクトとなる。また, 作業領域となる電紙は一時的なオブジェクトとなる。永続的なオブジェクトの指定には persistent という特別なキーワードを用いる。永続的なオブジェクト以外の電紙はすべて一時的なオブジェクトとなり, プログラム終了時に消去される。

4. 1. 4 電紙間のリンク

手書きの文字用電紙と認識された後のコード用電紙のように, 電紙間にリンクを張る必要がある。OS/omicon 第4版では, ワンレベルストアを用いているため電紙間のリンクはポインタで表すことができ, 永続的な電紙であればそのポインタはそのまま保存される。

4. 2 システムの拡張

C++では, 仮想関数のリンクにレイアウトバイディングが使われ, 仮想関数のリンクは実行時まで保留される。しかし, リンクできる範囲はコンパイル時に分かるものだけであり, システムの動的拡張までは考えられていない。そこで, このシステムでは仮想関数のリンクにダイナミックリンクを使うことにより, システムの動的拡張を可能にする。

そのために, システムをクラス構造で構築し, システムへのインタフェースをクラス構造により規定する。例えば, 電紙はシステムが用意するものであるから新たな電紙クラスを作ることシステム拡張である。また, ダイナミックリンクは, 仮想関数名とクラス名を用いて行なう。

5. C++言語処理系

すでに述べたようにこのシステムの記述には言語C++を用いる。しかし, システムでクラスやオブジェクトの管理を行ない仮想関数のリンクにダイナミックリンクを行なうために, 言語処理系で得られるクラスの情報をシステム側へ渡してやる必要がある。このため既存の言語処理系では対応できない。そこで, 我々の研究室で独自に開発した言語C処理系(CAT)を拡張することにより言語C++処理系を作成中である。この処理系では, 4. 1. 3で述べたように言語C++の文法を一部拡張している。

6. 今後の課題

現在C++処理系を開発中であるが, 今後はC++処理系を早急に完成させて, 実際にC++を用いてOS/omicon 第4版を構築し, 検証する。

7. おわりに

本稿では言語C++を用いてOS/omicon 第4版を実現することにより, OS/omicon 第4版にオブジェクト指向の概念を導入し, OS/omicon 第4版が提供する機能を言語レベルでサポートする方式について述べた。

参考文献

- [1] 早川, 他: “手書きインターフェースを支援する OS OS/omicon 第4版の構成”, 情報シンポジウム論文集, Vol.92, No.7, pp 35-42, 1992.
- [2] Y.Yokote, F.Teraoka, A.Mitsuzawa, N.Fujinami, and M.Tokoro: “The Muse Object Architecture: A New Operating System Structuring Concept”, ACM Operating Systems Review, Vol. 25, No.2, pp.22-46, 1991.