

パソコンのウィンドウシステム上のGUI構築ツール

2U-9

の実現方式について

岩井 仁史、荻田 茂、渡辺 実、渡辺 沙織

(株) 東芝 青梅工場

1. はじめに

グラフィックユーザインターフェイス（以下、GUIと略する。）を使った、ウィンドウシステム上でアプリケーションを開発する場合、ユーザインターフェイスの作成に開発工程のほとんどが費されてしまうことは、周知の事実となっている。そのため、GUIを簡単に作成するための開発ツールは数多く発表されている。

このたび、パソコン上でもっとも広く普及しているウィンドウシステムの一つである MS[®] Windows 上の GUI 開発支援ツール FineView を開発したので、その概要と特徴について報告する。

2. FineViewの構成

FineView は、FineView の中核であり基本的なデータの管理を行うFineViewマネージャと、そのうえに置かれている、いくつかのサブシステムから構成されている。このサブシステムをアタッチメントと呼んでいる。

マネージャをサーバ、個々のアタッチメントをクライアントとして考えると、全体の構成をつかみやすい。FinwView にデフォルトで組み込まれているのは、次の三つのアタッチメントである。

•オブジェクトマネージャ

FineView で扱うデータ（オブジェクト）を、実際のウィンドウやコントロール（ウィジット）として表現し、それらをマウスでドラッグ&ドロップすることでGUIの開発をすることができる。

•データマネージャ

データを、木構造を使った階層構造として表現し、編集作業を行うエディタである。GUIの構成要素の配置だけでなく、それぞれの処理についても ANSI-C のサブセットを使って処理を記述し編集することができる。

•Windows 基本アタッチメント

GUIの構成要素である、ウィンドウやコントロール（ウィジット）の管理を行うアタッチメントである。個々のコントロールの属性の設定を、GUIを使って簡単に行うことができる。

このうち、オブジェクトマネージャとデータマネージャは、FinwViewマネージャの管理するデータを二つのことなる表現で、編集するエディタであるといえる。アタッチメントは、アタッチメント間でのデータのやりとりはせず、FineViewマネージャに対してデータの更新、取得などの要求を行う構成となっている。このため、情報の同期処理は、個々のアタッチメントではなく、FineViewマネージャの機能

A implimentation of a GUI builder on a Personal Computer

Hitoshi Iwai, Shigeru Kanda, Minoru Watanabe, Saori Watanabe

TOSHIBA CORPORATION OME WORKS

として実現している。

また、ウィンドウシステムに依存する部分のほとんどは、アタッチメントとして実現されているので、FineViewマネージャはウィンドウシステムに依存せず、他のプラットフォームへの移植が比較的容易になっている。

3. FineViewの拡張性

3.1 コントロールのカスタマイズ

FineView の操作の特徴の一つは、『コピーする。』という操作を基本に置いていることにある。GUIを作成するときは、コピーボックスと呼ばれる、あらかじめ提供されている部品箱から、自分の作成しているウィンドウに部品（コントロールなど）をコピーする。

従来のGUI開発ツールでは、コピーボックスに相当するものは、プルダウンメニューやアイコンメニューである。これに対して、FineViewではコピーボックスは普通のウィンドウであり、ユーザが作成しているアプリケーションのウィンドウと、特に区別がない。このため、FineViewでは、ユーザがカスタマイズした部品を、コピーボックスに『コピーする。』ことで戻すことができる。

従来のGUIツールでカスタマイズコントロールを追加しようとするれば、特別なキットやユーティリティが必要であったのに比べると、FineViewのユーザカスタマイズの方法は、非常に単純で、簡単である。

3.2 アタッチメントの追加

FineViewのもう一つの拡張性に関する特徴の一つは、アタッチメントを動的に後から追加できることである。前述のようにコントロールを管理しているWindows 基本アタッチメントは、アタッチメントの一つであるから、別のアタッチメントに置き換えたり、新しいコントロールを管理するアタッチメントを追加することができる。

さらに、FineViewでは、プログラムの開発をGUIベースで行うオブジェクトマネージャや、データマネージャといったエディタもアタッチメントであるので、別のものに置き換えたり、新しいエディタを追加したりすることが可能である。

たとえば、クライアント/サーバ型のシステムを作成する場合に、通信のパラメタの設定などをGUIを使って簡単に行うアタッチメントを追加すれば、FineViewはGUIを作成するだけでなく、クライアント/サーバシステムの作成を支援するツールにもなることができる。データベースシステムのクライアント側の作成についても、まったく同じようにアタッチメントで対応できる。

4. 今後の課題

上記のような拡張をおこなう基本的な機構は、すでにFineViewの中に組み込まれており、現在は実験的にいろいろなアタッチメントを追加してみている。

今後の課題としては、アタッチメントのインターフェイスを一般化し、公開していくことである。現在のように、客先のニーズが多用化している場合、全てのシステムを構築できるようなツールを作成するだけの専門的な知識を持つことは、ほとんど不可能である。FineViewは、専門知識を持つ人に、アタッチメントを作成してもらえる場を提供することで、ツールとしての拡張をめざしていきたい。