

## 7F-5 ウィンドウのための ユーザインタフェース定義ツール

神原 英子 福岡 久雄 坂下 善彦  
三菱電機(株) 情報電子研究所

### 1. はじめに

より良いユーザインタフェース(U I F)の実現を目指して、ワークステーション(WS)へマルチウィンドウシステム(MWS)が取り入れられてきた。通常、MWSには、ウィンドウの表示、ウィンドウ上への描画、ポインティングデバイス(PD)やキーボード(KB)からの入力などの基本機能が用意されている。しかし、アプリケーション(AP)の開発者にとって、これらの基本機能を駆使してU I Fを設計することは容易ではない[1]。

本報告では、AP開発者のU I F設計に伴う負担を軽減することを目的としたU I F設計ツールについて述べる。

### 2. U I F設計上の問題点

マルチウィンドウ環境では、ユーザとの対話手段に・APの処理結果を出力するためのウィンドウ・コマンドやデータを入力するためのメニュー、及びフォーム

が用いられることが多い。このような形態のU I Fを持つAPの開発においては、ウィンドウ、メニュー、及びフォームを定義するデータ(総称してウィンドウデータと呼ぶ)を、ウィンドウシステムが扱える形で用意する必要がある。従来は、これらのデータ定義をAP本体で定義したため、次のような問題があった。

- (1) ウィンドウデータはウィンドウシステムが提供する基本機能を組み合わせて記述されるため、一般に記述レベルが低く、細部に渡る定義まで記述する必要がある。従って、データ記述量が多く、また、記述内容が複雑となり、AP開発者の負担となる。
- (2) ウィンドウデータの定義をAP本体で記述しているために、ウィンドウデータの修正の結果をAPに反映するには、AP本体の再コンパイルが必要となる。従って、結果を反映させるまでのターンアラウンドタイムが大きくなり、AP開発の生産性が低下する。
- (3) 操作性の観点からは、ウィンドウなどのレイアウト(位置、サイズ)が重要な問題である。従来のA

P開発では、APが完成に近付くまでは、それらを画面上で確認することは困難であった。

### 3. 問題解決のためのアプローチ

2. の問題に対して以下のような方法を取る。

- (1) データの記述量を少なくする。  
マクロな指定から細かい値をツールで計算する。(文字列のセンタリングなど)  
デフォルト値を示し、必要な場合のみ変更させる。  
必要なデータのみガイドして入力させる。
- (2) データの定義をAPの外で行う。  
ウィンドウデータはAP外で定義する。  
ウィンドウにはウィンドウ識別子が付加される。  
APはこの識別子を用いてウィンドウを表示する。
- (3) ウィンドウデータからウィンドウを表示する。  
APとは関係無く、ウィンドウの表示を確かめる。  
複数ウィンドウを表示して、画面レイアウトを確かめる。

### 4. U I F定義ツール

AP設計者のウィンドウ設計にかかる負担を軽減するために、3. の方法を取り入れたU I F定義ツールを検討し一部試作した。

U I F定義ツールの構成を図1に示す。

- (1) ウィンドウデータ定義部  
デフォルト値の提示などにより、データ定義を少なくする。  
既定定義データを再利用、または、修正する。
  - (2) ウィンドウ表示部  
ライブラリをコールして(1)のデータからウィンドウを表示する。
  - (3) 複数表示指定部
    - (1)のウィンドウの表示順序を決める。
    - (2)を使用して表示順序に沿ってウィンドウを画面上に表示する。
- (1)、(2)の過程を繰り返すことで、ウィンドウのデータ定義、表示確認をインタラクティブに行える。

また、生成されたウィンドウデータはAPがコールするライブラリによってAPに参照される。

4.1 ウィンドウの種類

ウィンドウには以下に示す種類がある。

- (1) ワークウィンドウ：  
APの処理結果を表示する。
- (2) ポップアップメニュー：  
メニューを表示し、コマンドを選択させる。
- (3) フォーム：  
PD入力エリアとKB入力エリアを設定し、データを入力させる。  
各ウィンドウによって、必要なデータの項目は異なる。

4.2 ライブラリ

UIF定義ツールがウィンドウを表示したり、メニュー操作やフォーム操作を行うために、APと共通の以下のようなライブラリが用意されている。

- (1) ウィンドウ表示
- (2) ウィンドウ消去
- (3) メニュー処理：  
メニューを表示して選択された項目番号を返す。
- (4) フォーム処理：  
フォームを表示し、ユーザ操作に対して、各エリアに対応した視覚反応を行い（エリアの反転など）ユーザ操作の結果を返す。

4.3 UIF設計手順

UIF設計手順を図2に示す。

- (1) ウィンドウデータ定義部  
ウィンドウデータの新規作成、修正、再利用を行う。データ値の有効性チェックも行われる。
- (2) ウィンドウの表示  
(1)のデータから、ライブラリを使用してウィンドウを表示する。
- (3) (1)、(2)の過程を繰り返す。
- (4) 仕様を満たすウィンドウデータのファイル化。
- (5) (1)-(4)で定義したウィンドウの表示順序を指定する。
- (6) 複数ウィンドウを表示しレイアウトを確かめる。

5. おわりに

ウィンドウ表示のためのUIF設計ツールについて述べた。このツールによって、ウィンドウ、ポップアップメニュー、フォームの設計、レイアウトの決定をインタラクティブに行う。AP設計者はライブラリを使用して、これらのウィンドウを即時に表示して確認する。それによって、AP設計者のウィンドウ設計における負荷が軽減した。

ウィンドウ内の細かい描画データの設計もUIFツールで行い、更に、ウィンドウ表示順序の指定情報もAPへ自動的に反映させることが今後の課題である。

参考文献

- [1]Henry Lieberman: There's More to Menu Systems Than Meets the Screen, SIGGRAPH '85 pp.181-189

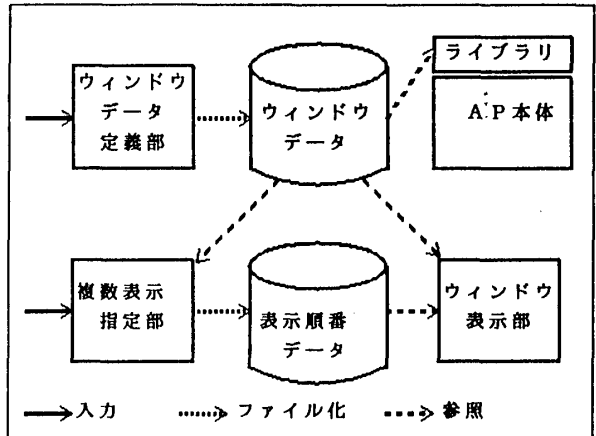


図1 UIF定義ツールの構成

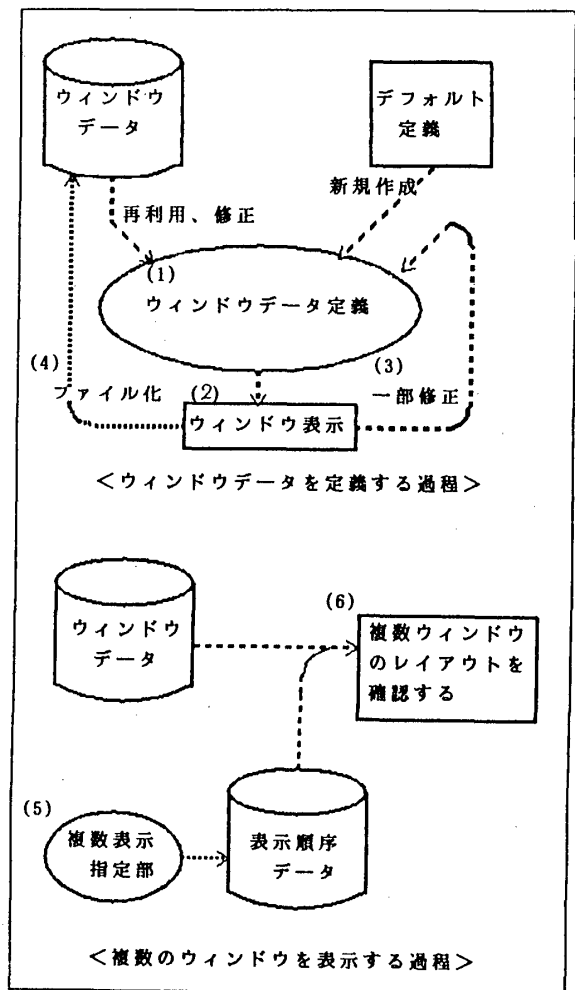


図2 UIFの設計手順