

2K-7

X Window上の
ユーザインタフェースプログラムジェネレータ

杉本直樹 (CSK総合研究所) 浅野雅史 (三井物産) 松浦敏雄 (大阪大) 吉岡信夫 (大工大) 的場裕司 (甲南大)

1. まえがき

ワークステーションの普及に伴い、計算機とユーザとのインタフェースとしてマルチウィンドウシステムが注目されている。そのなかで、X Window^[1]は移植性の良さや機能の豊富な点から、ワークステーション上のウィンドウシステムの標準となりつつある。一般にウィンドウシステムを利用したプログラムの作成は容易ではない^[2]。そこでX Windowでは、ユーザとの入出力によく使用されるボタンやメニューなどの部品をパッケージ化したライブラリ(XToolkit^[3])を用意している。しかしながら、XToolkitを用いる場合でも、そのコンセプトを理解するのに多くの時間が必要である。また、個々の部品の大きさを決めて配置するのもかなり面倒であり、何度もコンパイルと実行を繰り返す必要がある。

本研究ではアプリケーションプログラムで必要となる個々の部品を画面上でインタラクティブに配置することにより、そのようなインタフェース画面を実現するためのX Window上のプログラムを自動生成するシステムWeissを作成した。WeissはX WindowやXToolkitに関する詳細な知識がなくても使用でき、また生成されたプログラムの実行時でも、部品の大きさ、配置、および個々の部品属性の変更が可能である。

2. アプリケーションプログラムの作成手順

アプリケーションプログラムのWeissによる作成手順を以下に説明する(図1参照)。

(1) 部品の配置

Weissではアプリケーションプログラムが利用できる部品として以下のものを予め用意している。

Button	スイッチボタン
Volume	スライドボリューム
Choice Button	複数の中から1つを選択するボタン
Menu	ポップアップメニュー
Key Input	文字列入力用の領域
Message box	文字列出力用の領域

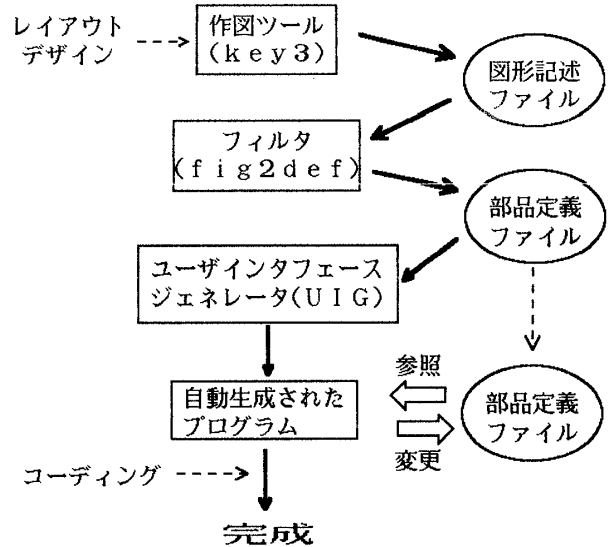


図1 Weissによるアプリケーションプログラムの開発課程

Layout Tool 配置変更用部品

部品の配置を行うには、作図ツールkey3^[3]を用いて、画面上で適当な位置に適当な大きさで並べれば良い。key3は部品の位置と大きさの情報を、図形定義ファイルとして出力する。

(2) 部品定義ファイルの生成

フィルタfig2defは図形定義ファイルを入力として、図2に示すような部品定義ファイルを出力する。この部品定義ファイルは各部品の名前、識別子、大きさ、

```
F:na="RootFrame":id#0:hn=NULL:
F:na="Frame1":id#1:fi#0:lb="Frame1":¥
:xp#10:yp#480:hi#430:wi#460:bw#2:
B:na="Button1":id#2:fi#1:lb="Button1":¥
:xp#140:yp#820:hi#30:wi#80:bw#2:¥
:ty="Push":lp#0:fn="8x13":
L:na="Layout":id#999:fi#3:lb="LayoutTool":¥
:xp#268:yp#405:wi#176:hi#82:bw#1:fn="8x13":
```

図2 部品定義ファイルの例

位置、線の太さなど、部品に関する全ての属性情報を保持している。この段階では各部品の位置と大きさ以外の項目の値は既定値が書き込まれている。

(3) ユーザインタフェース部の生成

フィルタUIGは部品定義ファイルを入力として、その部品群を画面に表示し取り扱うためのプログラムを生成する。生成されたプログラムはイベントのディスパッチループを関数mainの中に含み、個々の部品が必要とするCallback Procedureの枠組を含んでいる。このプログラムの実行時に、画面上の部品に対する操作（例えばメニューの選択など）が行われると、このディスパッチループから、予め用意されている部品ごとの処理ルーチンに制御が移行し、パラメータ（例えばメニューの項目番号など）とともに、必要に応じてCallback Procedureが呼び出される。

(4) 部品属性の変更

UIGにより生成されたプログラムは部品定義ファイルを読み込んで直ちに実行させることができ、部品定義ファイルの記述にしたがって画面上に部品が配置される（図3参照）。このとき、特別な部品であるLayout Toolを用いて、実際の画面を確認しながら画面上の部品の配置、大きさの変更ができる。さらに、各部品の細かい属性（境界線の太さやラベルなど）の変更も可能である。また、Layout Toolは、修正後の画面の部品を部品定義ファイルとして出力することができる。従って、次にこの新しい部品定義ファイルを用いて、プ

ログラムを稼働させると、修正後の画面が得られる。

(5) プログラミング

ユーザインタフェースの画面を作成した後、アプリケーションプログラムはUIGが生成したプログラムの内、Callback Procedureの中身をコーディングするだけでプログラムを完成させることができる。

3. ユーザプログラマブルなユーザインタフェース

Layout Toolはアプリケーションプログラムが実際の画面を確認しながらユーザインタフェースを作成するためのツールであるが、Layout Tool自身を完成された画面上に残しておくことで、画面の配置などの変更をエンドユーザに解放することができる。この機能によって、アプリケーションプログラムを変更することなく、エンドユーザの好みに応じたレイアウトを実現できる。

4. あとがき

画面上に部品を並べるだけで、アプリケーションプログラムのユーザインタフェース部のプログラムを自動生成するシステムWeissについて説明した。本システムは主にC言語で書かれ（一部awkを用いている）、ソースコードにして約8500行である。現在、予め用意している部品の種類および機能は充分とは言えない。これらの充実を図るとともに、部品自体の定義機能を用意する予定である。

参考文献

- [1] Scheifler, R.W. and Gettys, J.: "The X Window System", MIT, Laboratory for Computer Science (Jul. 1986).
- [2] Rosenthal, D.: "A Simple X11 Client Program - or - how hard can it really be to write 'Hello World?'", USENIX Winter Conference, Dallas, Texas (Feb. 1988).
- [3] "X Toolkit Intrinsics - C Language X Interface, X Window System, X Version 11, Release 2 (1988).
- [4] 北村他: "図形間の接続・包含関係を考慮した作図システム", 第31回情報処理学会全国大会, 7G10, pp.1507-1508(1985).
- [5] 酒匂: "Xme User's Guide(ver1.0)", 第3回SEA環境ワークショップ, 長野 (1988.1).

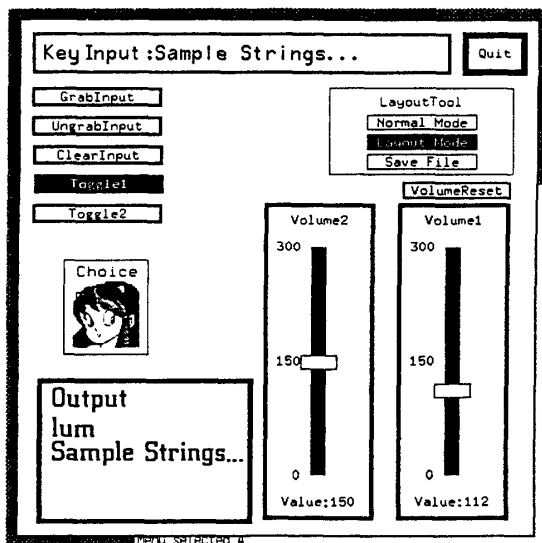


図3 画面表示例