

GUI 部品の視覚的かつ対話的な設計と

9H-2

プログラムの自動生成

大嶋慶諾

電気通信大学大学院 情報工学専攻

小林光夫

電気通信大学 情報工学科

1. はじめに

計算機との視覚的な対話を司る GUI(Graphical User Interface) は、使い勝手の良いソフトウェアにとって必須である。しかし GUI のプログラムを一般の利用者が作成するのは容易ではない。

現在、GUI生成のための専用言語 [1] や GUI 部品を使用している GUI 生成ツール [2] などが提案あるいは実用化されているが、いずれも視覚的プログラミングが出来ない、GUI 部品の選択肢が少ない、GUI 部品の動作の記述の自由度が低いなどの問題点がある。

本稿では GUI 部品設計ツールを用いて望む GUI 部品を視覚的かつ対話的に容易に作成する方法を与える。

2. GUI 部品の視覚的な設計

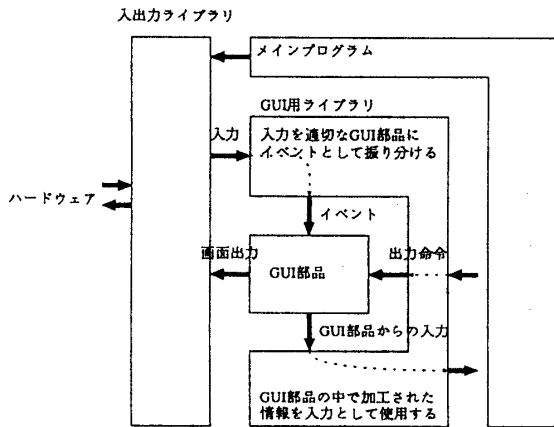


図1: GUI部品の使われ方

現在良く利用されている GUI 部品 (widget[3], control[4], control[5]) は、メインプログラムから直接使われるのではなく、図1に示すように GUI 用ライブラリを介して使われる。GUI 用ライ

Graphical and Interactive Design of GUI Module
OHSHIMA Keita, KOBAYASI Mituo
University of Electro-Communications

ブライは、メインプログラム、GUI 部品、および入出力ライブラリの間のインターフェースの役割を果たす。GUI 部品は、入出力ライブラリを通して伝わる入力操作を解釈し、操作のもつ意味すなわち利用者が入力したいと考える抽象的な情報をメインプログラムに返す。その際、画面出力の制御も行なう。

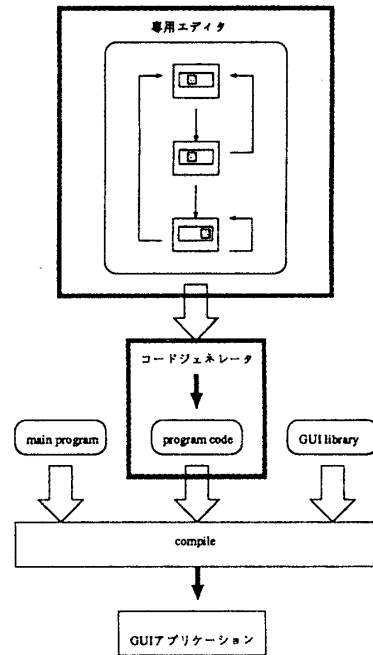


図2: GUI生成ツールの構成とそれを利用したアプリケーションの作成

われわれの方法によって生成される GUI 部品も、上記の GUI と同様の使い方を想定する。GUI 部品設計ツールは、専用エディタ (以後エディタと略) とコードジェネレータから成る (図2)。利用者は、まずエディタを用い、単純な基本図形と操作手順を組合わせて GUI 部品を設計する。そのとき、操作の推移も状態遷移図として与える。つぎに、コードジェネレータを用いて、目的とする GUI 部品のプログラムを生成する。

望む GUI 部品の仕様を視覚的に表現するにあたっては、その GUI 部品の意味が視覚的表現から容易に想像できるようになっていなければならない。そのためには、視覚的表現が以下の3つの条件を満たす必要がある。

1. 操作が画面変化に反映すること (状態変化).
2. 操作と描画図形の変化との対応がわかりやすいこと.
3. マウスポインタの位置に関する情報を表現できること.

エディタはこれらの条件を満たさなくてはならない。

1については、利用者の操作による遷移を表す状態遷移図を与えることによって解決できる。2については、位置関係のわかるいくつかの点(マウスポインタの位置と描画領域の原点を必ず含む)を定義し、その点にしたがって図形を描画することで解決できる。点の位置は、他の点から相対的に定めるようにすれば十分である。3については、2と同様に、図形を描画することによって解決できる。

3. GUI 部品設計ツールの実装

エディタとコードジェネレータを、X Window System 上で GUI ライブラリ Xt[3] を使用して作成した。エディタで描画出来る図形は、Xlib の提供している描画図形のみとした(実用上これで十分であろう)。

コードジェネレータはエディタの編集結果を受けとり widget(Xt の GUI 部品) のプログラムコード(C 言語) を生成するようにした。具体的には、表駆動により各種の機能を果たす汎用 widget のプログラムコードを用意しておき、コードジェネレータはその表およびそれを利用するための若干の変数、識別子を生成するようになっている。

4. おわりに

GUI 部品を視覚的かつ対話的に作成する方法を提案し、GUI 部品設計ツールを X Window System 上で実現した。実際に GUI 部品(widget) を作成しアプリケーションに使用してみた結果、このソフトウェアが有効であることがわかった。図3はエディタによる編集の様子である。

GUI には利用者に直観的な操作を可能にするという利点がある。しかし、少数の GUI 部品にあらゆる種類の入出力を任せるのでは、この利点の効果がない。新しい GUI 部品の必要性は常にある。その意味で、本稿で示した方法は GUI 作成に非常に有効であると考ええる。

最後に、いくつかの問題点を述べる。

現段階では、メインプログラムとのインタフェース部分については、利用者が自分でプログラムを書かなければならない部分もある(C 言語)。この部分についても視覚的プログラミングが出来るようにしたい。また、生成されたプログラムコードの効率が悪いので、改善する必要がある。

参考文献

- [1] Robert J. K. Jacob: A Specification Language for Direct-Manipulation User Interfaces, ACM Trans. Graphics, Oct, 1986, pp.283-317.
- [2] Andrew Harbert, William Lively and Sallie Sheppard : A Graphical Specification System for User-Interface Design, IEEE Software, July, 1990.
- [3] Adrian Nye and Tim O'Reilly: X Toolkit Intrinsic Programming Manual, O'Reilly & Associates, Inc, 1990.
- [4] Microsoft Corporation: Microsoft Windows Software Development Kit Guide to Programming,1990.
- [5] Apple Computer Japan : Apple Programmer's Introduction to the Machintosh Family(日本語版), トッパン,1990.

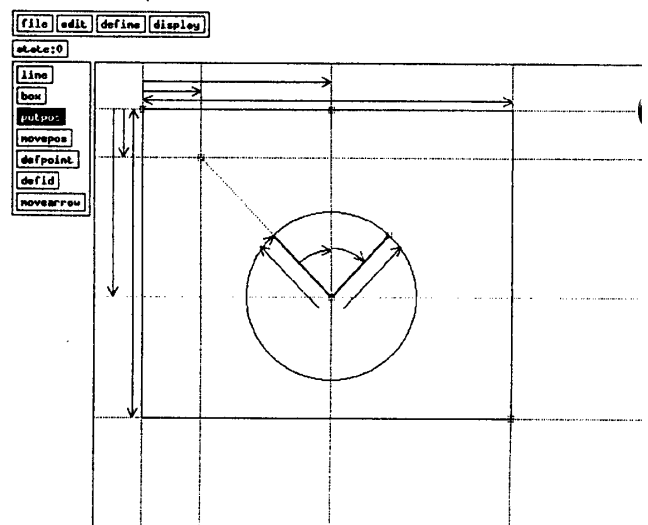


図3: エディタによる画面の編集