

シートによる NET/C ビジュアルプログラミングの汎用化*

2L-1

岩田 竜一 田中 義一 上原 稔 森 秀樹†
東洋大学工学部情報工学科‡

1 はじめに

現在、計算機のほとんどの操作が GUI を利用した環境によって行われている。これは対象物の可視化と、その直接操作が多くの人に理解されやすい方法であり、また高速、高解像度のグラフィックス表示の技術の進歩によって、実用的なシステムが開発されたためである。更に近年では、このような手法を取り入れたプログラミングについて研究が行われ、様々なビジュアルプログラミング（以下 VP）が提案されている [1][2][6]。

我々は並行プログラミング言語 NET/C[3][5] の VP 環境 [4] を開発中である。NET/C はアクターモデルに基づいた並行プログラミング言語である。VP 環境では NET/C プログラムの計算モデルを図形によって表し、またビジュアルなデバッグ環境を提供している。

しかし本システムでは、NET/C のモデル以外のプログラムを表現する手段がないこと、また図形で表現するよりテキストのプログラムの方が簡潔に記述できる場合もあるが、その方法がない、という問題点がある。

そこでフローチャートやテキストのプログラムなど、今まで扱うことのできなかつた他の図形表現法を扱えるように VP 環境を拡張する。本稿では、NET/C のビジュアルプログラムを拡張するための図形要素シートと、シートを定義するためのフレームについて述べる。シートの機能を拡張し、NET/C 以外の言語に対応することで、VP 環境をより汎用的なシステムにすることが可能である。

2 NET/C と VP

NET/C はアクターモデルに基づいた並行プログラミング言語である。アクターは自立的に動作し、複数のアクターが並行に処理を進めることができる。またアクターは互いに、ストリームにより通信をする。ストリームは単方向の通信路である。

アクターはより小さなアクターの組合せによって構成されている。それ以上分割することのできない最小単位のアクターをプリミティブアクターと呼ぶ。

NET/C プログラムの記述は、どのアクターを使用し、通信相手が誰かを明確にすることである。よってアクターとストリームを、ノードとエッジに対応させて描いた有向グラフで表現することができ、VP ではこのグラフをプログラムの表現法として利用している。

3 VP の汎用化

NET/C の VP では有向グラフを表現法として使っているため、データフローは自然に表現することができるが、手続的な記述はうまく表現することができない。

そこで表現法を拡張し、様々なプログラムの表現を可能にするためにシートを取り入れ、VP の汎用化を図る。シートはその上に他の図形を重ねることが可能であり、重ねられた図形の解釈はシートごとに決められた方法で行われる。

4 シート定義のためのフレーム

VP の対象となる言語やモデルに応じて、次々に新しいシートが追加され、種類が増加すると考えられる。そこでユーザがシートを柔軟に、簡単に拡張することができるように、シートを定義するためのフレームを導入する（図 1 参照）。

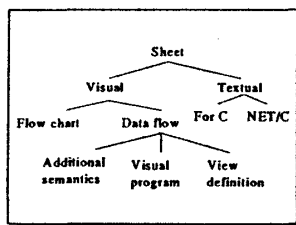


図 1: フレームの階層構造

フレームを細分化すると複数のフレームに共通した属性が表れるので、その属性を持つフレームをグループ化する。そして共通属性を持ったスーパークラスのフレームと、その

他の属性を持ったサブクラスのフレームに分け、スーパークラスから継承された属性値とサブクラスの属性によって一つのフレームを構成する。こうして属性ごとにフレームをまとめることで、ユーザはシートを定義するためのフレームを素早く探し出すことができる。

*A Generalization of NET/C Visual Programming by Sheets

†Ryuichi IWATA, Yoshikazu TANAKA, Minoru UEHARA, Hideki MORI

‡Department of Information and Computer Sciences, Toyo University

5 プログラムの例

プログラムをフローチャート表現で記述するシートをフレームによって生成する。この様子を図2に示す。図2の上部はフレーム階層を表し、下部は選択されたフレームによって得られるシートを表している。木構造の各ノードはボタンであり、クリックによって属性のリストが表示される。属性に合わせてシートの様子も変化する。

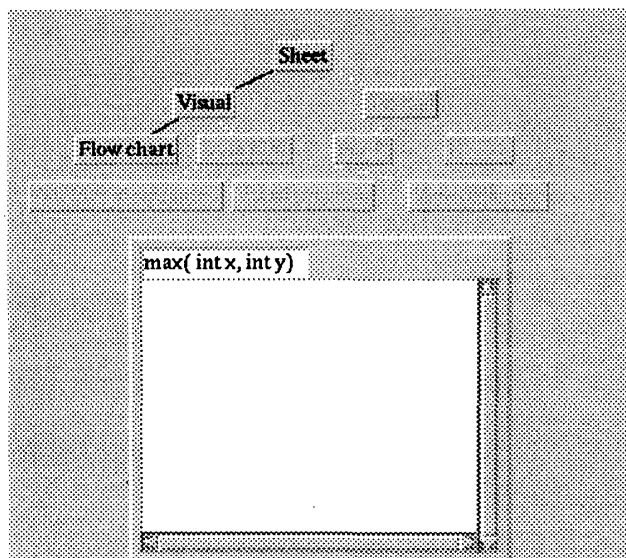


図2: フレームによってフローチャート表現のシートを定義

フローチャートによるプログラムの例を図3に示す。この例では2つの入力のうち、大きい数を入力するmax関数を定義している。フローチャートの表現規則として処理を表す四角形の中にはC言語の関数、またはNET/Cのアクターを記述するが、この例ではC言語によって表現している。内部で使われる変数名が関数の引数と同じならば、入力ストリームから入力を得る。また出力ストリームに書き出すためには標準出力へ出力する。

また同じmax関数をC言語のテキストプログラムで表現したものを図4に示す。

6 まとめ

本稿ではNET/CのVPを拡張するために、シートとフレームを導入した。シートによって様々なプログラムの表現が可能となり、本システム汎用化の基礎を作った。今後シートを多くの言語と表現法に対応させることが課題である。

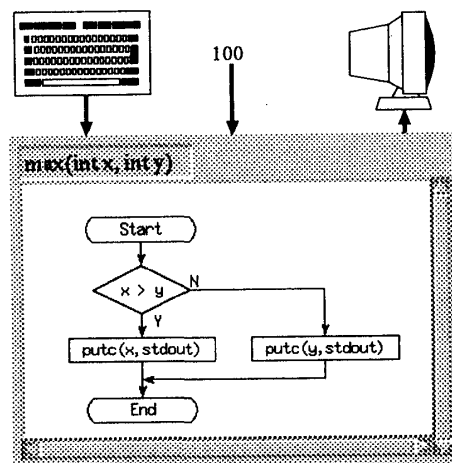


図3: フローチャートによるmax

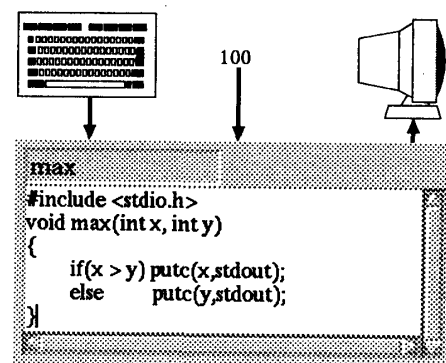


図4: C言語によるmax

参考文献

- [1] Dan Ingalls, Scott Wallace, et al. Fabric (A Visual Programming Environment). In *OOPSLA '88 Conference Proceedings*, pp. 176-189. ACM SIGPLAN, September 1988.
- [2] P.T.Cox, F.R.Giles, and T.Pietrzykowski. Prograph. In Adele Goldberg, Margaret Burnett, and Ted Lewis, editors, *VISUAL OBJECT-ORIENTED PROGRAMMING*. Manning Publications, 1995.
- [3] Minoru Uehara. NET/C: Toward the Fine Grained UNIX-like OS. In *OOS'94*, pp. 453-456. Springer-Verlag, December 1994.
- [4] 岩田 竜一. NET/Cのための図形による統合プログラミング環境. 第50回全国大会講演論文集(分冊5), pp. 23-24. 情報処理学会, Mar 1995.
- [5] 中林 嘉徳. ストリーム通信に基づく並行プログラミング言語NET/C. 第50回全国大会講演論文集(分冊5), pp. 21-22. 情報処理学会, Mar 1995.
- [6] 長崎 祥, 田中 謙. シンセティック・メディアシステム: Intelligent Pad. コンピュータソフトウェア, Jan 1994.