

IntelligentPadを用いたプログラムの グラフィック・トレーサの実現

7S-2

今滝 隆元 長崎 祥 田中 譲

北海道大学 工学部

はじめに

現在、我々は、IntelligentPad^{[1],[2],[3]}というシステムを開発している。このシステムは思考支援システムの構築ツールを目指した統合環境システムである。今回、その1つの特殊機能としてプログラムをグラフィカルにトレースするための特殊パッド(トレースパッドと呼ぶ)を開発したので、ここに報告する。

プログラムを作成する際に、作成中のプログラムの動作をグラフィカルに見たいと思うことがよくある。たとえば、ソートのプログラムを作ったとき、動作をグラフィカルに見ることができればこのプログラムが期待通りに動くかどうかを容易に試すことができる。しかし、そのためには、グラフィックインターフェースのプログラムを書かなければならない。この種のプログラムは元々のプログラムより大きくなることが多い。また、アニメーションのための特殊関数を用いねばならずプログラミングは一般に困難である。アニメーションのための特殊関数を用いたインターフェースプログラムを書くことなく、グラフィカルにプログラムのトレースができるようにすることが望まれる。そこで、IntelligentPadにおけるプリミティブパッドの1つとしてグラフィック・トレース用のパッドを新たに用意した。そのパッドの上のいろいろなパッドを貼ることによりグラフィック・トレースの機能が実現できるようにした。IntelligentPadはSmalltalk-80^[4]で開発しているのでグラフィック・トレースの対象となるプログラムはSmalltalk-80で書かれている場合に限定される。

トレース・パッドについて

トレース・パッドはプログラムソースの中の変数と、トレース・パッド上のパッドの内その変数名と等しい名前を持つパッドとを結び付け、変数値の変化を対応パッドの表示機能によりトレースすることを可能にする。プログラム中で表示パッドに対応付けられた変数に代入などが起こり、変数値の変化が生じると、この表示パッドに対してこの変数の値をパラメータとしてupdate:メッセージが転送される。もしその表示パッドの上にも別の表示パッドが貼られているとそれにもupdateが転送される。Smalltalk-80のようなオブジェクト指向言語ではプログラムをどのオブジェクトのものでコンパイルし実行するかによってプログラムの実行結果が違ふ。トレース・パッドではsendObject(Smalltalk-80におけるクラスInspectorのインスタンス変数objectのようなもの)というプロパティをもち、このプロパティで指定されるオブジェクトのもので、プログラムをコンパイルして実行する。図1では、

パッドabcがトレース・パッド、countがアナログメータのパッド、delta、degcountがデジタルメータのパッドである。図1の下部のプログラムをabcに送るとこのプログラムが実行され、各変数の値の変化がそれぞれに対応するパッドに表示される。ここでdegcountはcountがupdateを受けることによりupdateを受ける。

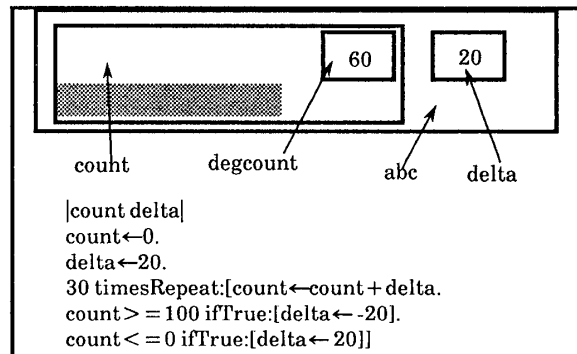


図1.変数のトレース

パッドの移動(図2参)

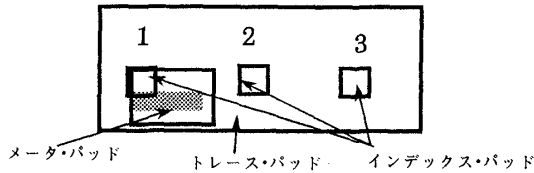
トレース・パッドの導入により、プログラム中の変数値のトレースがこの変数に対応付けた出力パッドを貼ることによりできるようになったが、ソートやハノイの塔などのプログラムのトレースは実際にパッドが移動しないとわかりにくい。そこで配列において要素の置き換えが生じると各要素に対応付けたパッドの移動が起きるようにした。これによりソートなどのプログラムに対して、簡単に動きのあるトレースを行うことが可能になった。ソートにかぎらずハノイの塔や土人と宣教師などのパズルのグラフィック・トレースも、プログラム中では配列に対する操作を行うだけで、各要素の動きを伴うアニメーションとして実行することが可能になる。この機能を実現するために、プリミティブパッドのひとつとして、プロパティとしてindexをもつパッド(インデックス・パッドと呼ぶ)を新たに定義した。ここでいうインデックスは配列のkey(索引)に当たる。インデックスは通常はクラスIntegerのインスタンスである。インデックス・パッドはトレース・パッド上で配列の各要素に対応するパッドを置く位置を指定する。これにより、配列の要素の置き換えが生じたとき、対応するパッドが移動する移動先の位置が定められる。インデックス・パッドはマウス操作でトレース・パッド上の任意の位置に貼ることができる。インデックス・パッドにパッドを貼ることにより配列のそのインデックスの要素に、上に貼ったパッドが保持される(1つのインデックス・パッドは複数のパッ

Graphical Program Tracer on IntelligentPad System

Takamoto IMATAKI, Akira NAGASAKI, Yuzuru TANAKA

Hokkaido Univ.

ドを保持できる)。インデックス・パッドのプロパティにはパッドの配置位置を示すplace、同じインデックス・パッドに保持されている複数のパッドの表示位置をずらすdeltaなどがある。



self at: 2 put:(self at:1).
 self at:1 put:nil.でメータパッドが1から2に動く。
 self at:1 put:4ではメータパッドの値が4になり移動はない

図2.パッドの移動

例.ハノイの塔(図3参)

ハノイの塔のグラフィカル・トレースをトレース・パッドを利用しておこなってみた。プログラムとしては、インスタンス変数resultをもつクラスHanoiを定義し、図3のプログラム3をHanoiのインスタンスメソッドとして定義するだけである。あとはトレース・パッドabcのsendObjectをHanoiのインスタンスにし、abcの上に図3のようにパッド配置する。そして“self hanoi:4 from:1 to:2 use:3”というプログラムをabcに送ればハノイの塔の動きがグラフィカルに見ることができる(図4のハードコピーの左下)。

おわりに

今回、IntelligentPadの新しい応用を目指して、プログラムのグラフィカル・トレースのためのトレース・パッドとインデックス・パッドを開発したので、これらのパッドの機構と利用法を例を用いて説明した。

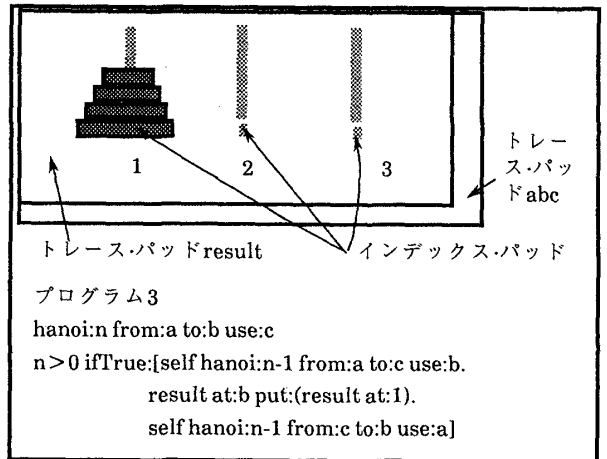


図3.ハノイの塔

【参考文献】

- [1]今滝隆元、田中譲: インテリジェント・パッドの開発、第37回情報処理学会全国大会講演論文集 pp. 707-708 (1988.9).
- [2]今滝隆元、田中譲: IntelligentPadにおけるパッドの実現機構、第38回情報処理学会全国大会講演論文集 pp. 1277-1278 (1989.3).
- [3]Y. Tanaka and T. Imataki: IntelligentPad: A HYPERMEDIA SYSTEM ALLOWING FUNCTIONAL COMPOSITIONS OF ACTIVE MEDIA OBJECTS THROUGH DIRECT MANIPULATIONS, Proc. of the IFIP 11th World Computer Congress, pp. 541-546, San Francisco (1989.8).
- [4]Adele Goldberg and Davis Robson.: "Smalltalk-80: The Language and its Implementation" (1983).

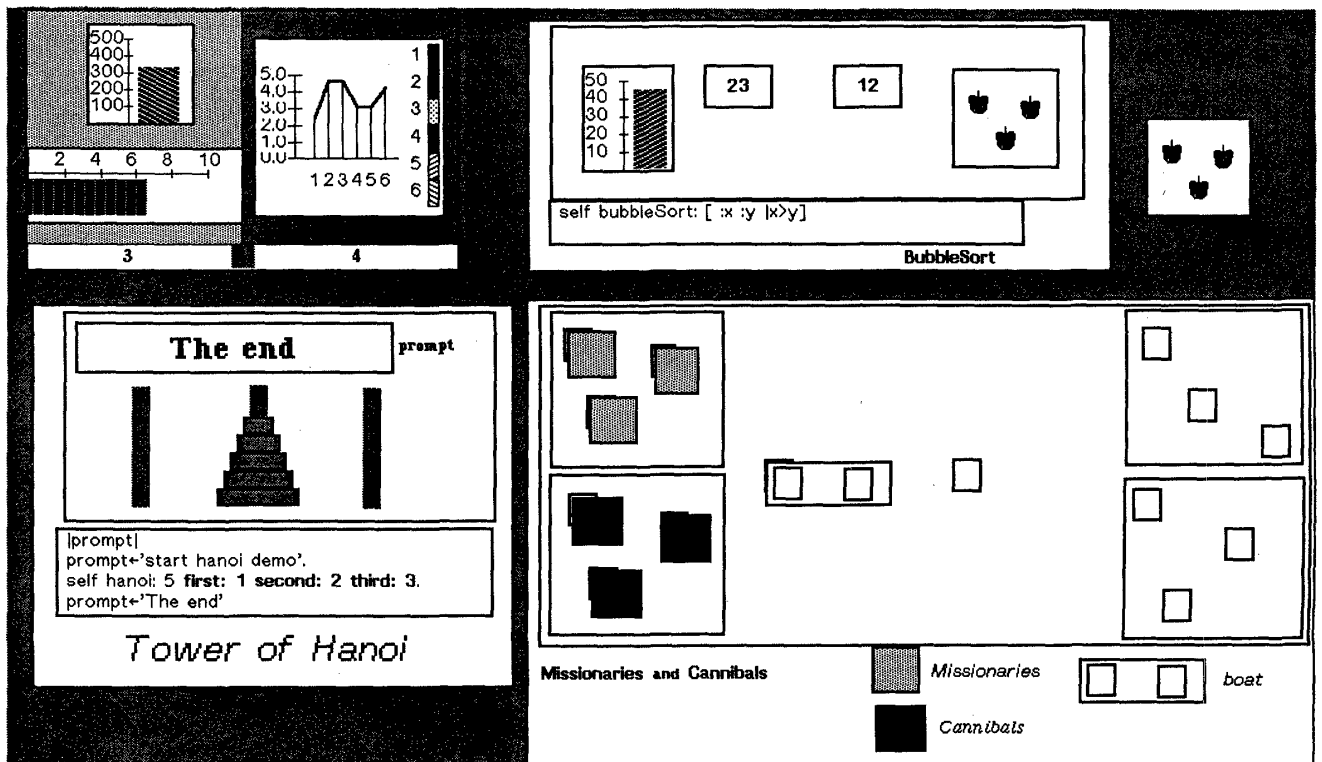


図4.画面のハードコピー