

2T-9

図形処理副言語 F X の設計と実現

石原 亘

京都芸術短期大学CG研究室

パスカル語プログラムの中で使える図形処理用の定数、関数、命令の集合を設計し実現した。その概要を報告する。

1 F X の機能

新しく開発した図形処理語彙は F X (Figure eXpress) と名付けられている。これを、親言語とするパスカル語と組み合わせて用いる。

F X の機能の中心は図形を定数と関数とからなる式として記述することにある。式の中では平行移動量や色などを指定するためにパスカル語で許される形式の数式を用いることができる。

F X ではおもに立体図形を扱う。稜線や表面の属性には色と肌目が含まれる。F X では 2 次線形多面体(頂点と直線と平面とからできた図形)だけを扱う。したがって、内部の詰まった図形や曲面図形は 2 次線形多面体で近似してから記述する。なお、視点の正面に板状の図形を立てれば平面図形と全く同じになるので、この手法によって平面図形も扱える。

F X で書いた譜は、記述された図形を線画か面画で表示することができる。面画の表示ではあまりリアルな映像は得られない。しかし、面の肌目に応じて陰影を与えるくらいのことはできる。

```

:
enscreen;
expose(
  head(roll(cube,pi/6.0),pi/4.0),
  -10.0,1.0,flash,sound,
  0.0,0.0,0.0,10.0);
descreen;
:
    
```

譜 1 F X による作譜

2 F X の構成

F X の語彙のほとんどは図形からより複雑な新しい図形を導き出す関数である。このほかいくらかの定数と五つの命令とが含まれる。これらはパスカル語から

は関数/手続きとして扱われる。

F X の語彙は目的によって次の三つに大きく分けることができる。

1 図形記述用 立方体や円などの基本形態を表す定数、平行移動や着彩の結果を導出するための関数、およびいくつかの図形を結合して新しい図形を導出する関数の 3 種の語彙がある。

これらのうち最後の型の関数は、そのままではパスカル語プログラムにうまく馴染まない。結合させたい図形の個数が一定しないからである。そこで、バインダ(binder)と呼ぶキューに似たタイプを準備し、その上に図形を並べてから関数に渡すようにした。

move	平行移動
expand	拡大/縮小
pitch	X軸に対する回転
head	Y軸に対する回転
roll	Z軸に対する回転
opaque	塗り潰し
stain	塗り重ね
retex	磨き出し
unite	結合
bridge	開いた掃引体
ring	閉じた掃引体
cone	錐体
cylinder	柱体

図 2 opaque(cube,T,1.0-T,0.0)
 ⇨ <緑から黄色を経て赤に変わって行く立方体>

図 2 図形記述関数

2 表示用 図形を表示する命令と少しの補助命令が含まれる。また、図形およびそれを表示してできた映像をファイルに記録する命令がある。

3 リアリズムフィルタ 図形を面画で表示する場合にはリアリズムフィルタ(realism filter)と呼ばれる関数をいくつか組み合わせて使う。これらを通してから

図形を表示するとよりリアルな映像が得られる。

sort	遠い順に並び替え
front	裏向き面を除く
devide	面の細分
lighten	照明を当てる
fade	レンズでの調色

図3 リアリズムフィルタ

リアリズム機能は表示命令の中に含ませることもできたが、FXではこのように独立させている。目的とする映像の傾向によってはリアリズム処理のうちいくつかは効果がないことがある。このような場合にプログラマの判断でそれらを切り落とせば、あまり速くない機械でもそこそこの効率を得られるからである。

4) バインダ関係 バインダを扱うための関数と命令が含まれる。バインダはキューに似ているが、読み終えた項目を削除しない。また、FXとは関係ないが現在位置を動かさないで別の場所を乱読みできる。

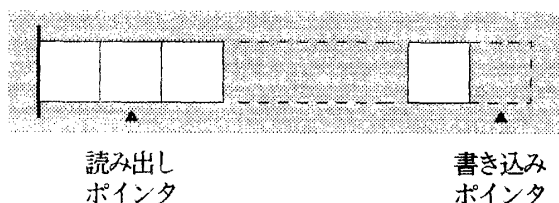


図4 バインダ

3 設計における留意点

1) 変数のいらぬ作譜 FXでは、第1に譜とそれが生成する映像との関係が明確になることを目指した。なぜなら、作譜を通じて図形をデザインする場合、譜はコンピュータだけが使うものではなく、作家が作品の内容を検討する記述法としても機能しなければならないからである。

このため、FXでは図形の導出を文(statement)ではなく式によって行う。文で図形を導出する方式では、導出の結果を受け取って次の処理に回すために図形の内容とは関係のない変数が必要になるからである。ただ、可変数個オペランド演算のためにバインダ変数を使うことだけは避けられなかった。

2) 運動量を数式で指定 FXはアニメーションの制作にも利用される。この場合、図形の運動量は時刻を含む式で指定できなければならない。これはまた反復

的な空間図形を制作する場合にも重要である。

FXは親言語への追加語彙なので、この要請も容易に実現できた。

3) 変形の記述 変形が記述できるようにするには、それぞれが独立して運動している図形をいくつか繋いで掃引体が生成できればいい。この点でフィグス(FIGS)などのU演算は複雑な図形を構成するには十分でない。

FXでは各種の掃引操作を準備して変形の記述に備えている。

```

:
create(S);
for K:=-N to N do
begin
D:=float(K)/float(N);
append(S,move(circuit,
0.0,
sin((D+TIME)*2.0*pi,
D))
end;
F:=unite(S);
:

```

(TIMEは当該フレームの時刻)

図5 bridgeによる波打ち

4 実現

FXはパソコン環境とEWS(Engineering Work Station)環境とで利用できるようになっている。ファイルをインクルードするためのコンパイラ命令を除いて全く同じソースが両方の環境で使える。

パソコン環境ではPC100、PC98XA、PC98XL(日本電気)のそれぞれの上でターボパスカル(ポーランド)と合せて使うことができる。

EWSでは、DN660とDN3000(アポロコンピュータ)上のイージスパスカルから利用できる。

FXはFXキットと呼ばれるさらに下の層の語彙によって記述されている。FXキットでは図形を頂点、稜線、表面それぞれのバインダを集積したものとして扱う。プログラマは機能や効率におけるFXの不足をFXキットで補うことができる。