

PGS (PLAN Graphic Support) について*

栗生晴夫** 山本貞昭**

計算機の今後の発展は、適用業務に密着した、いわゆる、問題向き言語 (Problem-Oriented Language) の普及・進歩に負うところが大きい。

従来、応用プログラムの開発には、GPSS, PERT/COST, LP などのように、それぞれの適用分野特有の言語が定義され使用されてきた。応用プログラムごとに、言語仕様の決定、翻訳プログラムの作成、保守・拡張等を行なわなければならなかった。また、翻訳プログラム自身、ハードウェア、プログラミング・システムに密着しているため、機械が変わるごとに作り直さなくてはならなかった。

今後、計算機の適用分野が広がるにつれて、ますます適用分野ごとにわかりやすく、かつ、効率の良い問題向き言語が要求されようし、また一方、計算機 (ハードウェア/ソフトウェア) 自体はますます複雑になってくる。

これからの分野であるグラフィック・プロセッシングは、その最たるもののひとつであろう。

以下、応用プログラム作成の汎用的なアプローチともいうべき PLAN (Problem Language Analyzer) と、そのグラフィック・プロセッシングにおける適用である PGS (PLAN Graphic Support) について簡単に紹介する。

PLAN では、応用プログラムが機能的

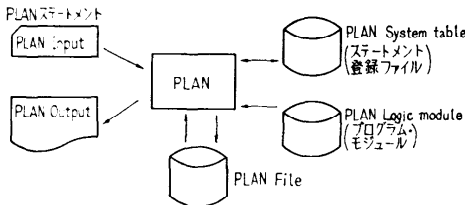


Fig. 1 PLAN System Diagram

にモジュール化されている。各モジュールは、ASA FORTRAN で書かれ、コンパル化され、実行可能な形で Fig. 1 のようにライブラリにたくわえられている。応用プログラムの各機能と、それを果たすプログラム・モジュール、データとの結びつけは、ステートメントによって行なわれる。このステートメントには、それぞれ Fig. 2 の例のように、GRAPHIC REPORT GENERATOR, PARAMETER, GRAPH などの名前がつけられ、ステートメント登録ファイル (Fig. 1) に登録される (定義時)。ADD PHRASE がこの目的で使われる。実行時には、すでに登録されているステートメント名を用い、パラメータ、データ等に初期値をセットしてインプットすると、順番に所定のプログラムをライブラリよりロードし実行す

```

定義時 {
ADD PHRASE : GRAPHIC REPORT GENERATOR, LEVEL1, PPO'Mφ7φ1';
ADD PHRASE : PARAMETER, (4)ILC'N', P+2 (25)YGL11,
I(2) DDV 1, PRO'Mφ7φ2';
ADD PHRASE : GRAPH, P+2 (26)XGS8.5, P+2 (27)YGS 11,
(8)BPC'Y', I (1φ)ADCφ, PRO'Mφ7φ3';
}

実行時 {
:
:
GRAPHIC REPORT GENERATOR;
PARAMETER, ILC'Y', YGL3φ;
GRAPH, XGS 1φ 15 ADC1 BPC'N';
:
:
}
モデル 1.
  
```

Fig. 2 PLAN Statement Sample

る。一度登録したステートメントの削除・変更・追加もちろん可能である。Fig. 2 は、IBM 113φ における PLAN の例である。

PLAN の利点は

○ **拡張性** 応用プログラムに新しい機能をつけ加えたいときには、FORTRAN でモジュールをつくり、ADD PHRASE を使って登録してやれば、以後ステートメントの名前だけで使える。

○ **汎用性** 応用プログラムごとに翻訳プログラムを作る必要がない。翻訳プログラムは PLAN Interpreter のみである!!

* Introduction to PGS (PLAN Graphic Support), by Haruo Kuryu and Sadaaki Yamamoto (IBM, JAPAN, Ltd.)

** 日本アイ・ビー・エム株式会社

◦独立性 計算機が変わっても応用プログラムを書き直す必要はない (ASA BASIC FORTRAN 使用).

紙面の制約のため、PLAN の説明がきわめて不十分であるが、PGS に話しを進める。

PGS では、PLAN によって基本的な図形処理の機能が、ステートメントとしてすでに登録されている。もちろん、その機能に対応するプログラム・モジュールも FORTRAN で書かれ、実行可能な形でライブラリに入れられている。したがって、ユーザーは IBM 2250 のハードウェア/ソフトウェアを知らなくても、簡単に図形を表示したり、オペレーターよりのインプット (ライトペン (Light Pen: LP), ファンクションキーボード (Programmed Function Keyboard: PFKB), キーボード (KB)) を処理するプログラムを書くことができるわけである。

PGS の主要構成は、以下のようになっている。

1. Panel Specification

次の各項目についての条件が与えられているディスプレイ情報を Panel という。

- 表示すべき図形
- 図形構成要素間の区別
- PFKB のランプのオン/オフ
- 割込みの種類
- 割込みの結果とるべき処置 (プログラム・レスポンス)

これらの条件は、Fig. 3 のように 1 Panel につき何枚かのカードで与えられる。

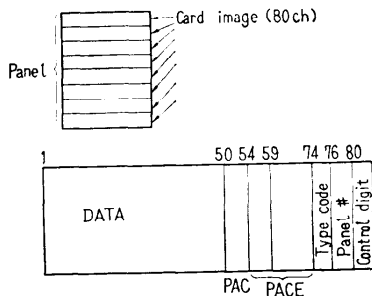


Fig. 3 GPS Panel Information Flow

Type Code (2けた):

01 }
 }
35 }

36 文字表示

- 37 文字表示(大)
- 38 文字によるグラフ
- 39 " (大)

45 円ないし円弧

5φ PFKB

Panel # (4けた):

何枚かのカードが 1 Panel に属することを示す。

Control Digit (1けた):

- 9 : ラスト・カード
- 8 : コメント・カード

PAC (Program Action Code) (4けた):

このカードによって図形が表示されたとき、または表示された図形を LP がさしたときに、プログラムがとるべき処置を 4 けたのコードによって指定する。

たとえば

- 次の Panel を表示する。
- プログラム・モジュールをロードし実行する。
- スクリーン上の X, Y 座標を Common Area にセットする。
- PGS を終わる。
- その他

PACE (PAC Extention) (20 けた):

PAC で指示した処置に関するパラメータやデータを与える。

2. Display Management Files

(Fig. 4 参照)

① Specification File

Panel に関する情報の集まり。

② Panel File

Specification File 中の各 Panel について、実行可能な IBM 2250φ のバッファ・コードに変換された

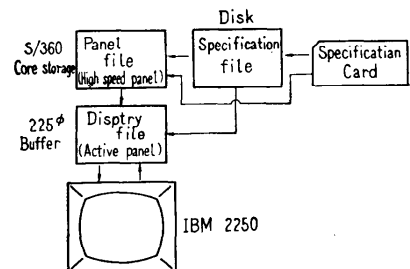


Fig. 4 PGS Panel Information

ものおよびテーブル類。

③ Display File

現在 IBM 225 ϕ 上に表示されている Panel File.

3. Display Management Routines

PGS の各機能を果たすためのプログラム・モジュール。

もちろん、ユーザーは PLAN の約束に従って FORTRAN で書くことによって、独自のプログラム・モジュールを加えることができる。

4. Panel Language

図形処理に便利のように定義された PLAN ステートメントの集まり。3. の Display Management Routines に対応する。

① File および PGS の Initialize/Terminate.

② 各 Specification Record の生成、取扱い。

③ 各 File の作成。

④ Specification File への追加/削除

(ADD/DELETE).

Fig. 5 の簡単な例について説明する。

ここでは、Fig. 5 のように A が LP でセンスされると、カーソルをこの右側に移し、KB より A の値がセットされる。END-KEY が押されると、この値は CPU のコアの中に入れられる。同じことを B, C について繰り返す。LP が COMPUTE をセンスすると、 z の値が計算され、 z を LP でセンスすると計算の結果が表示される。CONTINUE をセンスするとこのパネルに関する処理を終わり、次の

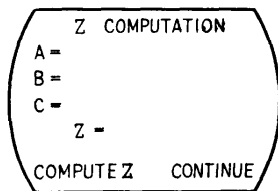


Fig. 5 PGS Sample (Screen)

C	ZCOMP -- Z COMPUTATION COMMON L(625), LS(15), MC(45 ϕ), D(1 ϕ) EQUIVALENCE (A,D(51)), (B,D(52)), (C,D(53)) (Z, D(6 ϕ)) Z=A + B + C RETURN END
---	---

Fig. 6 PGS Sample Case (Program Module)

PLAN ステートメントの処理に移る。

以上のうち z の値を計算する機能以外は PGS がすでに PLAN ステートメントの形で定義し持っている。 z を計算するモジュールは Fig. 6 のようになる。

ここで L(625), LS(15) は PLAN プログラムが駐在する場所であり、MC(45 ϕ) は PLAN によって管理されている Common Area で、PLAN が必要に応じて Save, Restore を行ない、また Reset もする。D(1 ϕ) は通常各モジュール間でデータの受け渡しをするのに使われる (大きさ任意)。

Panel Record として Fig. 7 のように 8 枚のカードが使われる。

Fig. 7 のカードについて簡単に説明する。

(1) Type Code 37 は、スクリーンを 49 \times 35 の格子に分割し、 z を 18(X), 32(Y) の位置から 13 けたディスプレイすることを表わしている。

(2) PAC Code 37 は(1)と同様。

PAC Code 4731 は文字がディスプレイされたとき、COMMON Area の D(51) に ϕ をセットし、この文字が LP でさされたとき、キイ・インしたデータを D(51) にセットすることを表わしている。

(3) PAC Code 4631 は LP が文字を LP がさしたとき、D(6 ϕ) にあるデータを Fixed Point の形でディスプレイすることを表わしている。

(4) PAC Code ϕ 313 は、文字 z を LP がさしたとき、 z COMP というプログラム・モジュールを Load し実行することを表わしている。

(5) PAC Code ϕ 318 は、文字 CONTINUE を LP がさしたとき、次の PLAN ステートメントを実行するため、PLAN モニターへ Return することを表わしている。

この例を PLAN のもとで実行するためには、Fig. 8 で示した PLAN ステートメントを用意し、Fig. 7 のカードを所定のところにはさみインプットすればよい。

それぞれのステートメントの意味は

- (1) PLAN システムの開始。
- (2) PLAN のもとでの PGS Job の開始。
- (3) PGS で使用する File の指定。
- (4) 以上のデータを Panel Number = 8 として

(1)	18	32	13	Z COMPUTATION				37	0801
(2)	7	26	4	A=	4731	51	φ.	37	0802
	7	22	4	B=	4731	52	φ.	37	0803
	7	18	4	C=	4731	53	φ.	37	0804
(3)	12	13	4	Z=	4631	6φ		37	0805
(4)	5	4	9	COMPUTE Z	φ313		Z COMP	37	0806
(5)	34	4	8	CONTINUE	φ318			37	0807 9

Appearance data
PAC
Pointer Data
Type code
#
Control Digit

PACE

Fig. 7 PGS Sample Case (Panel Specification Cards)

- (1) PLAN JOB ;
- (2) PGS JOB ;
- (3) PGS
- (4) READ SPECIFICATION CARDS FOR PANEL 8 ;
- ⋮
- } Fig 7 のカード
- (5) DISPLY , PANEL 8 ;

Fig. 8 PGS Sample Case (PLAN Statement)

Specification File に Store する.

- (5) Specification File から Panel Number=8 をとりだして PGM のプログラム・モジュールによって IBM 225φ にディスプレイする.

参 考 文 献

IBM Application Program, Problem Language Analyzer (PLAN) Application Description, H20 -0490-0 (1968)

(昭和44年2月4日受付)