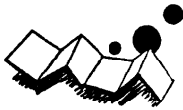


## 解説



### 事務用簡易言語の動向†

魚田勝臣\*\* 小碓暉雄††

#### 1. はじめに

最近のオフィスコンピュータやパーソナルコンピュータの急激な高性能化と普及に伴い、これらの計算機の事務データ処理を主体とした簡易言語の研究開発が盛んになっている。この背景には、高性能化したこれらの計算機を駆使するために、従来からの COBOL, FORTRAN, BASIC などの手続き型言語に頼っていたのでは、応用プログラムの生産能力が追いつかないという、メーカーや販売会社などの供給側および利用者側の深刻な悩みがある。オフィスコンピュータやパーソナルコンピュータの普及に伴い、応用プログラムの開発要員は増加しているものの、それ以上に計算機応用のためのプログラム開発作業は増えている。供給側ではプログラム要員やシステムエンジニア (SE) の増員が図られている。しかし、COBOL, BASIC などの言語の場合、一人前となるためには OJT も含めて3年程度以上の育成期間が必要であり、育成期間中に脱落する人達も多いのが現状である。オフィスコンピュータやパーソナルコンピュータの普及のためには、COBOL や BASIC は難しい。COBOL や BASIC が難しいのは、こうした古典的な手続き型言語が、自然言語に近い文法を持つものの基本的には計算機の処理の流れを忠実に表現する形をとっていることによる。売上げ伝票の発行、在庫台帳の修正、集計表の作成などといったオフィスの単純作業を機械化するためにも、相当なプログラム開発作業が必要となる。人間同志の情報の伝達と手作業でやれば誰にでも簡単にできることが、計算機に実施させようとするとなぜ簡単にできないのか、という素朴な疑問から事務用簡易言語の要求が芽生えたのである。

書類や口頭を通じた伝達と手作業による場合、入力データとその(演算などの)加工法および出力様式を端的に伝えれば、手作業する人の潜在知識や判断能力によって、速度や精度にばらつきはあるものの求める結果が得られる。手作業を機械化するのは、もちろん処理の速さと正確さを求めるなどのためである。そのための計算機への指示を人間同志の伝達に近い手法に近づけるために、人間の潜在知識や判断能力に相当する“内蔵機能”を計算機に備えさせる必要がある。事務用簡易言語の設計は、適用業務に合わせたマクロ命令的な内蔵機能によって利用者の指示を簡易化することが基本となる。この内蔵機能は計算機の規模や応用分野によって異なり、その結果、簡易言語の様式も違った様相を呈する。本文では、このうちのオフィスコンピュータから汎用中形計算機クラスの次のような条件に適合する事務用簡易言語に焦点を絞った。

(1) ハードディスクを媒体とする大容量ファイルを備え、大量の入出力データを反復的に処理するインライン又はローカルバッチ処理を主体とする。

(2) 入出力方式やデータ様式が多様で、特にプリント装置への出力では COBOL 的な編集機能が要求される。又、最近では漢字やグラフ処理も備えられている。

(3) 一般的に、一度作った応用プログラムの寿命が長く、改良が加えられながら反復利用される。

この条件設定の理由として、この種の計算機の簡易言語の歴史が比較的長くかつ幅広い応用範囲を持つことと、供給側と利用者の双方で、プログラム開発のための負荷抑制の要請が極めて強いためである。したがって、簡易言語のもう一方の分野を形成している、比較的少量のデータを専らエンドユーザー層で扱うパーソナルコンピュータ向けの簡易言語については除外している。

なお、簡易言語は次々に新しいものが発表されている(その中には寿命の極めて短いものも多数ある)。したがって、比較的知られているものでも本文中に洩

† Trends of Simplified Business Processing Languages by Katsuomi UOTA (Mitsubishi Electric Corp., Systems Engineering Department) and Teruo KOIKARI (Mitsubishi Electric Corp., Computer Works, Small Computer Development Division).

\*\* 三菱電機(株)電子システム部  
†† 三菱電機(株)計算機製作所小形電算機開発グループ

れている可能性もある。本文中での言語例の選択は、1981年度に情報処理学会機関誌編集委員会で収集した、簡易言語に関するアンケート調査の資料を中心にを行った<sup>1)</sup>。

## 2. 事務用簡易言語の歴史と分類

事務用簡易言語を説明する上で、IBMのRPG (Report Program Generator) の歴史と他に与えた影響を無視することはできないので、その歴史を最初に述べ、次にその分類について述べる。

### 2.1 RPGの発達とその影響

#### 2.1.1 RPGの創成

事務用簡易言語の草分けとして、その後の事務用簡易言語の代表的様式を築いたものがIBMシステム/360用のRPGである。RPGは、計算機による一般的なデータ処理が、入力、演算、出力の三つのプロセスからなることを前提としている。データファイルを読んで報告書を作成するために、データファイルの体裁と報告書の体裁を別々の仕様書で定義し、必要となるデータ項目の演算を演算仕様書で定義して、読み書きに伴う手続きの記述の必要をなくした。RPGでは見出し、明細、合計などからなる報告書作成のための処理サイクルと処理機能を内蔵しており、実行制御のために標識と呼ぶスイッチを使って、流れ図を書かなくても原始プログラムが作成できるようにした。

#### 2.1.2 RPG IIへの発展

RPGは、乱呼出しのためのディスクファイル応用や配列処理などを可能とするために言語仕様が拡張され、1969年のIBMシステム/3の発表とともにRPG II<sup>2)</sup>となった。RPGが固定された処理サイクルと機能を持つのにに対し、RPG IIは柔軟なロジックを組むことができるようなデータ定義と命令を追加し、より汎用性の高い言語へと発展した。

RPG IIは、システム/32、システム/34へと順次受け継がれ、ワークステーションシステムのための画面様式機能(DDF)やオンライン機能が強化されて、IBM小形事務用計算機の主言語となった。

RPG IIは、IBM計算機の使用者に広く普及しただけでなく、他のメーカーもRPG IIコンパイラを開発したので、それらの使用者の間にも広まって行った。一方、RPG II的な発想を採り入れながらRPG IIと異なる業務処理の最適化を計った三菱電機のプログレス II<sup>5)</sup>、UNIVACのIDLE 1100<sup>4)</sup>などの他の簡易言語の開発を誘発するに至った。

RPG IIプログラムは、図-1の例に見るようないくつかの仕様書からなり、各仕様書は行と列位置によって記入項目が定められている。RPG IIには、全部で8種類の仕様書があって、H, F, E, L, T, I, C, およびOの仕様書コードが決められている。実際の応用プログラムは、8種類の仕様書すべてを使うのではなく、必要なものだけ使えばよい。この仕様書の考え方は、事務用簡易言語で最も広く用いられている。RPG IIは、その処理サイクルや内蔵機能から、バッチ処理方式の事務データ処理にはコーディング労力の少ない最適化された言語と言える。一方、RPG IIへの不満としては、リアルタイム応答処理向きでないこととデータ入出力単位が常にレコード単位であることがあげられる。

#### 2.1.3 RPG IIIへの展開

RPG IIはIBMシステム/38の出現とともにRPG III<sup>3)</sup>へ発展している。RPG III仕様は、ファイル編成によってデータの呼出し法を制限してきた時代を過ぎ、適用業務に合ったデータの見方ができるデータベース化を実現した。また、ファイル仕様書の19桁目にEを指定することによって、プログラム外のデータ記述書DDS (Data Description Specification)<sup>16)</sup>でレコード定義や画面定義を行うことができ、データの独立を実現している。その他、構造化プログラミング的な命令の追加、複数端末での処理を1台の端末処理として記述するリエントラントなオブジェクト構造化などの仕様が付加された。いずれもプログラムの生産性及び保守性向上のための今後のプログラム言語の一つの方向を示すものである。

#### 2.1.4 RPGについてのまとめ

RPGの歴史が目指してきた方向は、COBOLよりプログラム開発効率のよい事務用簡易言語であって、その現実的効果は明白である。一方、

- (1) 効率の最適化がバッチ処理に片寄っていること、
- (2) 一定期間の教育による習得が必要なこと、および、
- (3) コーディングとコンパイルという準備作業が必要なこと、

などの問題点があり、これらを解決した各種の言語が誕生している。たとえば、日本電気のSMARTや三菱電機のDUETなどに代表されるワークステーションの画面を用いた会話型インタプリタ言語で、リアルタイム処理を容易にする簡易言語があげられる。

ファイル仕様書

記号	属性	ファイル名															フィールド名															フィールド属性															フィールド内容																												
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74
0:2	F	TAPE															IP															F 660 33															MT00																												
0:3	F	LIST															IP															F 132 132															LP00																												
0:4	F																																																																										

プログラム名: \_\_\_\_\_ 入力仕様書

プログラム: \_\_\_\_\_ 日付: 年 月 日

記号	属性	レコード識別コード															フィールド名															フィールド属性															フィールド内容																												
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74
0:1	I	TAPE															TT															10																																											
0:2	I																																														3 CHIEU L2																												
0:3	I																																														4 EIGHT L1																												
0:4	I																																														9 16 NR																												
0:5	I																																														17 230URAGE																												
0:6	I																																																																										

プログラム名: \_\_\_\_\_ 演算仕様書

プログラム: \_\_\_\_\_ 日付: 年 月 日

記号	属性	演算条件															演算結果															演算結果															演算結果																													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	
0:1	C	10															URAGE															ADD															TOTAL1															TOTAL1 30														
0:2	C	L1															TOTAL1															ADD															TOTAL2															TOTAL2 90														
0:3	C																																																																											

プログラム名: \_\_\_\_\_ 出力仕様書

プログラム: \_\_\_\_\_ 日付: 年 月 日

記号	属性	出力条件															出力結果															出力結果															出力結果																																																											
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74																															
0:1	O	LIST															H															03															IP															35															'7070-175-600'																													
0:2	O																																																																																																									
0:3	O																H															205															IP															10															'47'																													
0:4	O																																																																																																									
0:5	O																																														21															'エイト・エイト'																																												
0:6	O																																														35															'エイト・エイト・エイト'																																												
0:7	O																																														50															'7070-7070'																																												
0:8	O																2															10																																																																										
0:9	O																															L2															CHIEU															10															① ①は得意コードの最初1桁入力																													
1:0	O																															L1															EIGHT															21															② ②は得意コードの最初2桁入力																													
1:1	O																																																																																																									
1:2	O																																														35															NR																																												
1:3	O																																														50															URAGE1																																												
1:4	O																7															2															L1																														50															TOTAL11R														
1:5	O																7															2															L2																														50															TOTAL21														
1:6	O																																																																																																									

図-1 RPG II のプログラム例

表-1 プログラム言語の進化

番号	分類	言語例	コーディング
①	機械語		2進コード
②	記号化言語	アセンブラ	英数字記号
③	自然言語的 (第一世代)	FORTRAN COBOL ALGOL	一般に簡略英語的
④	自然言語的 (端末用途)	BASIC APL	簡略英語と記号の併用
⑤	自然言語的 (構造化言語)	PASCAL Ada	簡略英語と記号の併用
⑥	パラメータ言語	RPG II プログレス II TASKFORCE	空欄埋込方式の仕様書
⑦	画面会話型	SMART BRADS III DUET	メニュー方式による画面入力

2.2 事務用簡易言語の分類

事務用簡易言語は、プログラム言語の歴史的経過における産物といえる。そこで、事務用簡易言語のタイプを論ずる前に、プログラム言語の進化の面から表-1に代表的言語を分類した。

事務用簡易言語に対する一般に認められる明白な定義はないが、1章で記述したように、計算機に対する業務指示を人間同志の伝達様式に近づけたものとして考えると、表-1の⑥および⑦がほぼ該当すると言えよう。既存の簡易言語と呼べるものを、その言語様式、応用範囲、実行方式およびデータ構造の面から区分けて、それぞれの場合の特性を考察する。

I. 言語様式上の分類

言語様式上は、仕様書式と画面対話式に大別できる。仕様書式は、RPG II に代表されるように記法やオペレーティングシステムの背景に従って目的用紙の上に記述する。記述の時に計算機による支援が受けられないので習得までの教育が必要で初心者向けとは言い難いが、習得後のプログラムの生産性は高く、かつ汎用言語的な広い応用範囲をもつ言語も多い。これに対し、画面対話式は、画面の特性を生かして目視を通じて利用者を目的に誘導できるので、より初心者向きと言える。反面、このような手段によって初心者を

表-2 簡易言語の言語様式上の分類

	仕様書式	画面対話式
代表的な言語	RPG II/RPG III (IBM) プログレス II (三菱) TASKFORCE (富士通)	BRADS III (IBM) SMART (日電) DUET (三菱)
長所	応用範囲が広い (汎用性が高い)	初心者向けである (教育期間を要しない)
短所	初心者向けではない (教育期間を要する)	応用範囲に限られる (汎用性に欠ける)

表-3 簡易言語の応用範囲上の分類

	万能指向	限定業務指向
代表的な言語	RPG II/RPG III (IBM) プログレス II (三菱)	<多目的> SMART (日電) DUET (三菱) <バッチ処理> IDLE 1100 (ユニパック) <照会・検索> DRS (富士通)
長所	1. 習得すれば何でもできる 2. 利用率が高い	1. 習得が早い 2. 型に合えば開発が早い
短所	習得期間を要する	一言語ですべて処理できない

誘導できる事項には限界があり、複雑な指示が困難であることも否めない。一般には、パターン化された処理に適用範囲が絞られ比較的汎用性に欠けるのが現状である。表-2 に言語様式上の分類をまとめた。

II. 応用範囲上の分類

応用範囲上は、万能を目指したものと業務を限定したものとに大別できる。前者は後者に比較して習得期間は長いものの、習得後は一言語ですべての処理が賅えることになる。また、一般に適用機種での標準言語であることも多く、ユーザでの利用率も高い。一方、業務を限定したものは、約束ごとがそれだけ少なく、習得期間が短かくて済み初心者にも容易に使うことができる。型にはまった業務に対しては万能指向の言語より効果的であるが、この言語ですべての業務をカバーすることができず、COBOLなどの言語と併用される。表-3 に応用範囲上の分類をまとめた。

III. 実現方式上の分類

方式上は、コンパイラ型(翻訳後実行)とインタプリタまたはコーティリティ型(即実行)とに大別できる。コンパイラ型は、実行時の性能を最適化した形式で利用頻度の高いプログラムを保存しておくための言語が多い。また、例外処理に強く、IIの万能指向の言語であることが多い。インタプリタまたはユーティリティ型は、厳しい実行性能を求める処理には向かないが、概してコンパイラ型より文法が単純でコーディング量が少なく、初心者向けである。反面、例外処理に弱く、汎用性に欠ける。表-4 に実現方式上の分類を

表-4 簡易言語の実現方式上の分類

	コンパイラ型	インタプリタ型またはユーティリティ型
代表的な言語	RPG II/RPG III (IBM) プログレス II (三菱) TASKFORCE (富士通)	SMART (日電) DRS (富士通)
長所	1. 実行性能が最適化される 2. 例外処理に強い	1. 文法が易しい 2. 即座に実行できる
短所	1. 文法が複雑(教育要) 2. 即座に実行できない	1. 実行性能に劣る 2. 例外処理に弱い

まとめた。

IV. データ構造上の分類

プログラムの扱うデータ構造上は、物理ファイル(従来のファイル)を主体とする場合と論理ファイル(データベース)を主体とする場合とに分けられる。従来の事務用簡易言語は、物理ファイルを扱う場合が圧倒的に多かったが、物理ファイルではデータの物理的構造に処理が制約されるので、今後の傾向としては、RPG IIIのようなデータベース主体の簡易言語がより多く出てくるであろう。

総合的な評価として、万能でかつ利用者のあらゆる

条件を満足する簡易言語はあり得ないと言える。したがって、事務用簡易言語としても、利用者の目的や経験に合わせて、言語の様式、応用範囲、実現方式などに関して、最適化の焦点が絞られることになる。

3. 事務用簡易言語の例

1章で紹介した1981年のアンケート調査の中から、普及度が比較的高くタイプを異にする3種の事務用簡易言語について紹介する。ただし、これらが簡易言語の主流であると断定しているのではない。

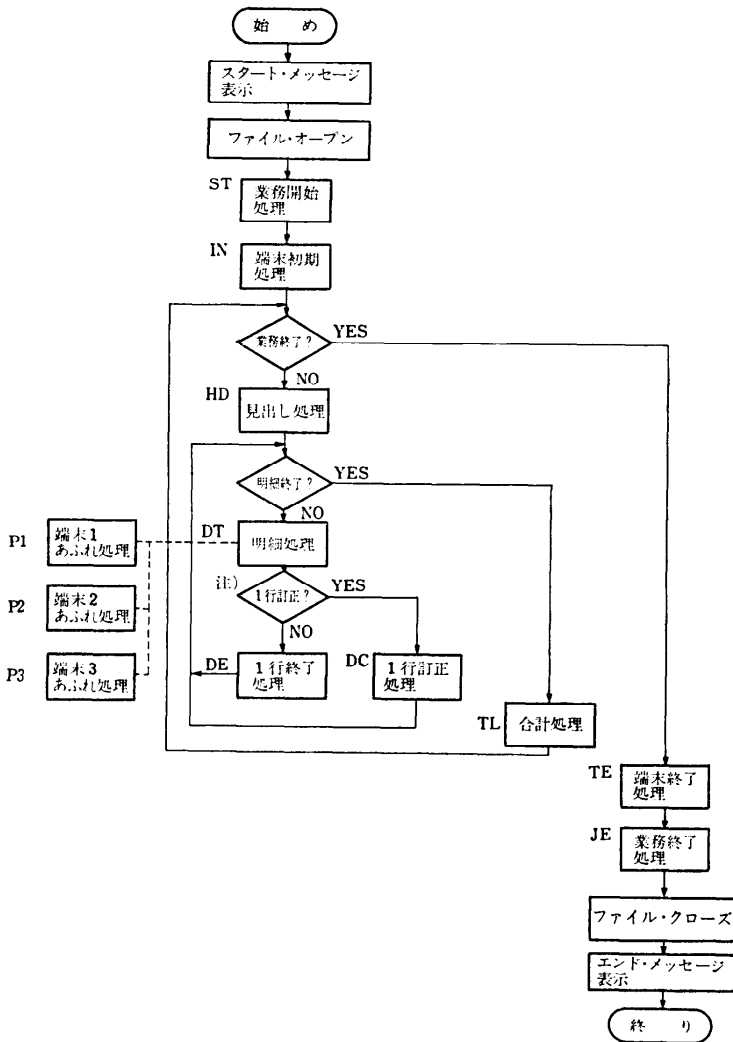


図-2 オンライン・プログラムの処理サイクル (プログレスII)  
 注) 各ブロックはすべてオプションであり、記入がなければその処理がパスされる。



### 3.1 プログレス II

三菱電機のプログレス II<sup>6)</sup> は、基本的には RPG II 又は RPG III と同じタイプの指示書 (RPG 系の仕様書に相当する) 形式によるコンパイラ型事務用簡易言語であるが、次の点で RPG 系と異なった進化をたどっている。

(1) RPG 的なバッチ処理サイクルとは別に、データ項目単位のリアルタイム応答処理のためのオンライン (又はインライン) 処理サイクルとその制御のための専用標識及び内蔵機能を備える<sup>17)</sup> (図-2 参照)。

(2) 漢字処理、図形処理などのためのデータ記述や特殊制御マクロ機能を備える。

(3) ワークステーション画面から会話型でプログラムの作成およびデバッグができる専用のエディタを備える。

RPG III の進化に先がけて、データベースの独立定義とデータベースアクセス、CALL 命令によるサブプログラムの呼出し、多端末処理に対するリエントラントなオブジェクト生成などを実現している。

プログレス II の指示書は、プログラム指示書、端末指示書、入力ファイル指示書、出力ファイル指示書、プリント指示書、端末入出力指示書、作業項目指示書及び処理指示書の 8 種類の指示書からなり、業務に応じて必要な指示書を選択する。中でも端末指示書、端末入出力指示書は、最もプログレス II の特長を表わすもので、データ項目単位の応答やそれに伴うフィールド訂正などきめ細かい処理を記述できる (図-3 参照)。

プログレス II は、オフィスコンピュータの代表的業務に対する記述の最適化をしている半面、COBOL と同じような命令で手続きを書くこともできるので事務処理のすべてをカバーできる言語仕様となっている。万能言語としての側面は、手続き的であり簡易言語と呼べる領域をはずれた部分だが、応用システムが一言語ですべて開発できることを意図した結果である。

### 3.2 SMART

SMART<sup>6)</sup> は、日本電気のオフィスコンピュータクラスのオペレーティングシステム ITOS のソフトウェア体系の中にあつて、報告書作成、データ入力、ファイル抽出、ファイル更新、問合せなどの定型業務を CRT 画面から会話型で実行できるユーティリティ型の事務用簡易言語である。

SMART は、図-4 のような処理形態で実行され、次のような特長を備える。

(1) ファイル、レコード及びレコードを構成する

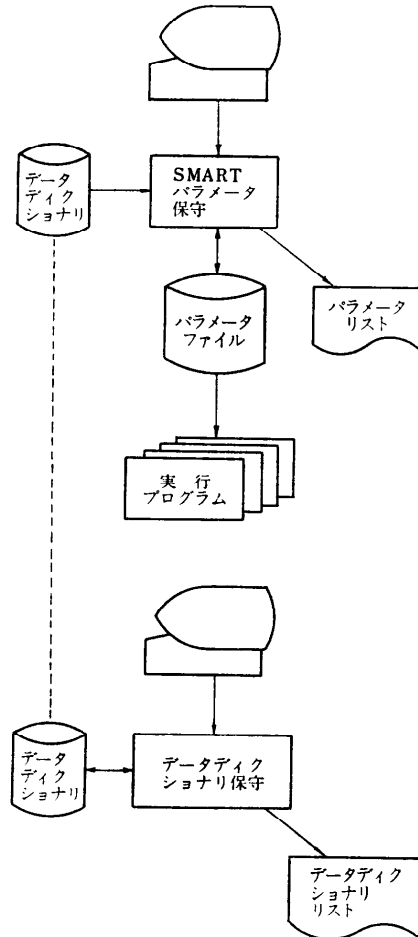


図-4 SMART の処理形態

データ項目に関する定義は、データディクショナリとして独立に定義、または修正できる。

(2) SMART 実行のパラメータは、パラメータファイルとして登録され、再利用や変更利用ができる。

(3) 報告書作成、データ入力、問合せなどの処理系は、単機能ユーティリティの集合であつて、CRT 画面の SMART ガイダンス表示に沿って要求処理に必要なパラメータを入力してゆくことにより、一連の業務が単機能の組合せとして実行できるようになっている (図-5 参照)。

SMART は、ユーティリティの集合であるので、適用範囲や利用条件がコンパイラ型の言語より、限定されるのは否めない。また、実行時の効率の面では必ずしも最適化されていないが、定型業務のインラ

インやバッチ処理については、大抵の業務に応じることができる。

SMART の秀れた点は、データディクショナリにファイル情報が登録されていれば、プログラム経験のないエンドユーザでも抵抗なく利用できる取りつき易さにある。目的に至る CRT 画面によるガイダンスの推移は、プログラムの経験者にとっては遠まわしな感是否めないが、計算機をエンドユーザ層により身近なものとするマンマシンインタフェースとして、効果を発揮している。今後、プログレス II のような、教育を受けたプログラマのための生産性向上に力点を置いた言語とは異なるエンドユーザ言語として、SMART 自身もその応用範囲を拡げられよう。

3.3 TASKFORCE

FACOM UNIOS の TASKFORCE<sup>7)-10)</sup> は、次の 4 種の処理パターンのコンパイラ言語を総称した事務用簡易言語である。

- バッチ処理用 (TASKAL)
- データエントリー用 (DES)
- ビリング処理用
- オンライン処理用

TASKFORCE は、RPG 系 やプログレス II と同様、数種類の、“記述”と呼ぶパラメータ記入用紙にコーディングする形式の言語である。TASKFORCE の記述は、一種類の記述の内容が極めて単純なのが特長である (図-6 参照)。これらの記述には、4 種の処理パターンに共通なアイデント記述、ファイル記述、レコード記述、リテラル記述などと、バッチ処理用のヘッド記述、ディテール記述など、処理パターンごとに固有なものがある。個々の記述は、/ID、/FILE、/RECORD などといったパラメータ見出しステートメントから始まる。

TASKFORCE において、説明書も合わせて四つの処理パターンに分け、1 行のパラメータ記述内容を単純にしているのは、利用者にとって、習得を楽にし、行単位ごとの記述誤りを少なくする効果が大い。また、処理パターンを拡張してゆくことにより、機能の拡張が容易にできる点も有利である。ただし、RPG II やプログレス II と較べて、1 プログラムのコーディングステップ数は多くなる傾向がある。

TASKFORCE は、処理パターン別に分かれていることから、どの処理パターンに属するか解りにくい業

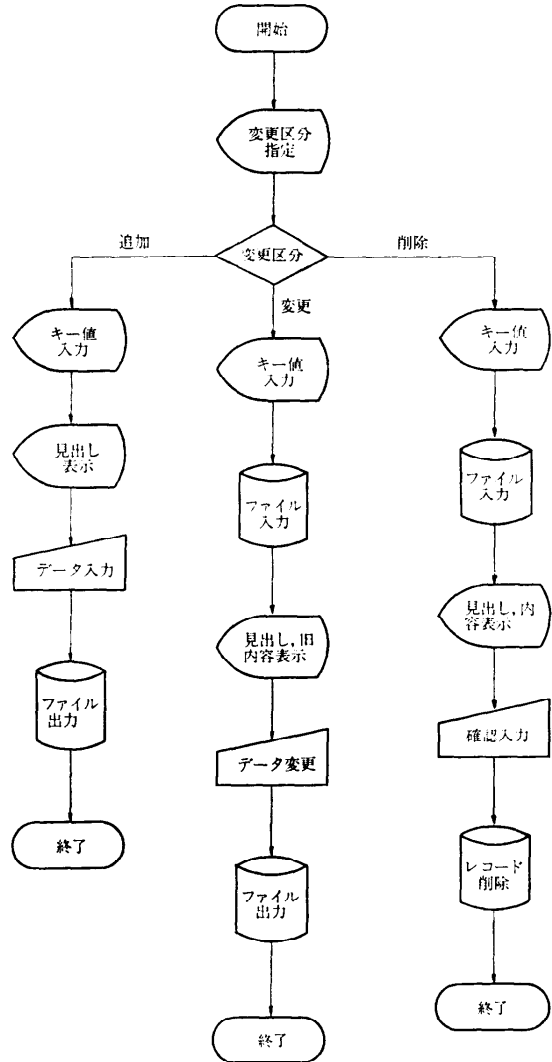


図-5 SMART のファイル変更の処理の流れ

務や複合的な業務には COBOL を用いることになっている。現実には、UNIOS ソフトウェア体系の位置づけの上でも、定型的な業務をとり扱う主力簡易言語とされており、RPG II やプログレス II のような統一言語としての役割を狙っているのではない。

3.4 まとめ

以上の三つの言語例に見る各メーカーの政策として、次の二つの方向が存在することが理解される。

- (1) 生産性向上を意図する簡易言語の側面に、汎用言語の要素を加えて、一言語での統一を図る。
- (2) 簡易言語は利用範囲を限定し、COBOL など



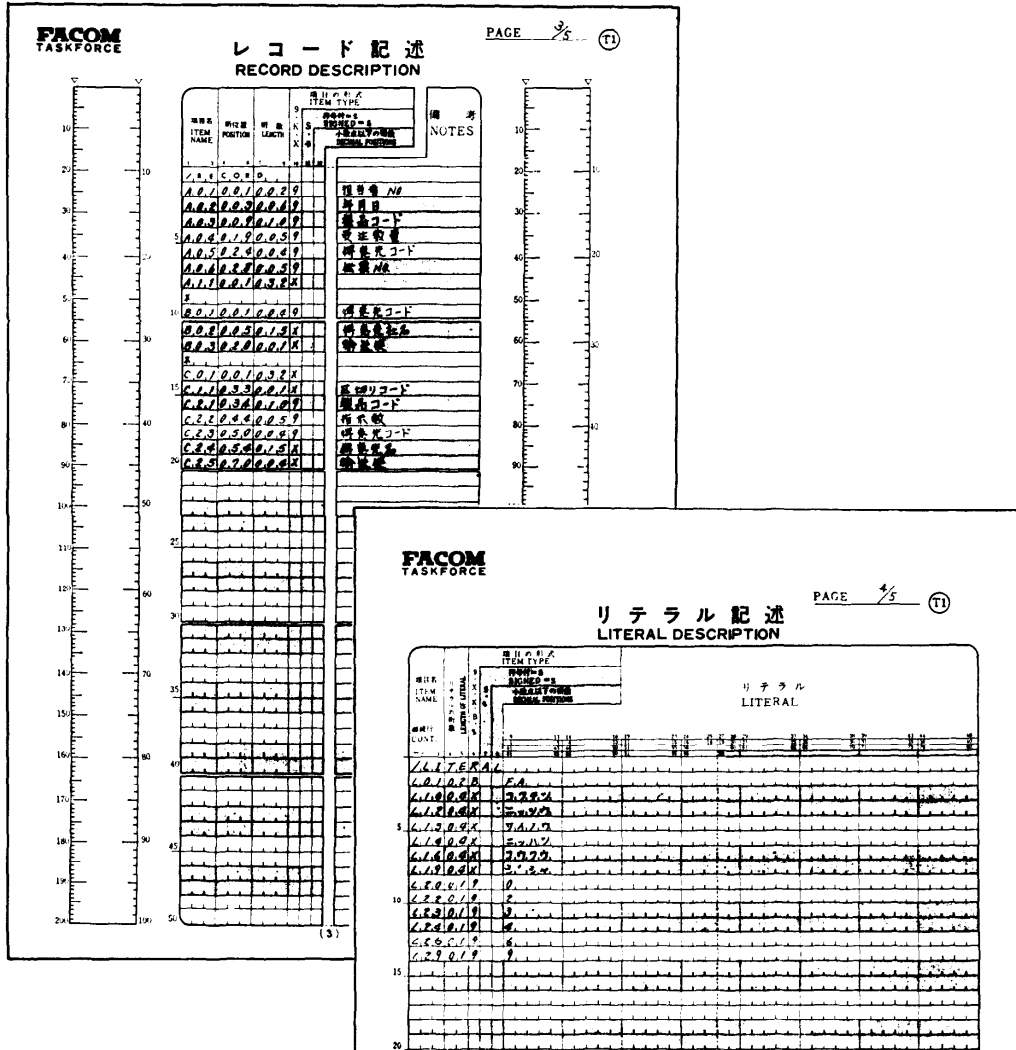


図-6 TASKFORCE の記述の例

の汎用言語との併用を図る。

なお、本論文では簡易言語の生産性の比較検討については触れない。各言語を紹介した論文や資料などには、“COBOL 比で 00 倍の生産性”のような表現が見られるが、客観性に乏しいためである。

4. 今後の動向

本節では、2章および3章に示した具体的簡易言語も含め、今後の簡易言語の機能要素として必ず考慮されなければならない二つの課題について説明する。

4.1 ファイルからデータベースへ

2.1 節で述べた RPG から RPG III への発展に見るように、データ処理方式の基本がデータのアクセスをファイル編成によって制限していた時代から、適用業務に合わせてデータベースをアクセスする時代に移ってきている。応用システムを構築する上で、データベースの方が、

- (1) データの一元管理及共同利用が計りやすい、
- (2) データ参照を多様化できる、および、
- (3) 応用システムのメンテナンスが容易である、

表-5 事務用簡易言語のデータベースモデル

言語名	提供会社	DBMS名	データベースの論理構造(モデル)
プログレスII	三菱電機	DMS-3 <sup>1)</sup>	ネットワークモデル(CODASTL型)
DRS <sup>1)</sup>	富士通	DBM	ネットワークモデル(TOTAL型)
RPGIII	IBM	CPF	関係モデル
SEQUEL 2 <sup>1)</sup>	RSI	ORACLE	関係モデル

などのメリットによりプログラム作成の生産性が高くなるであろう。一方では、データベース化する場合の論理構造化の手間や障害回復方法などのシステム設計上の問題もあるが、最近の計算機における著しい性能価格化の向上により、これらの難題を解決したデータベース管理システム(DBMS)が開発されてきている。これに伴い、事務用簡易言語でも今後データベース処理機能が標準的に提供されてゆくと思われる。

簡易言語でのデータベース処理機能の実現上、ファイルの場合と異なっていくつかの課題がある。

第1は、ファイルの場合、JIS COBOL 1980 (ANS COBOL 1974)に定められた順ファイル、相対ファイル及び索引ファイルといった規準がある。事務データ処理の世界では明に暗にこれをモデルとしているが、データベースの場合、その論理構造を決めるモデルに今のところ規準がない。ちなみに、データベース機能を提供している簡易言語の代表的な例についての採用モデルを表-5に示す。

表-5において、IBM システム/38 CPF で提供しているデータベースは、関係モデルの提唱者の Codd の定義した情報代数の結合(Join)を満足していない点で、関係モデルとは異なるという見解もある。なお、データベースの各種モデルの是非の議論は、本論文の主旨でないので割愛する。

第2は、言語やデータの互換性の点から、従来のファイルとの関係をどうするかという課題がある。三菱電機のDBMSであるDMS-3は、オフィスコンピュータのクラスでは最も早期にデータベース処理機能を提供した。そのため、大勢を占めていた従来からのファイル利用者に影響を与えないようにデータベースアクセスメソッドをファイルアクセスメソッドと独立に設計、言語インタフェースも独立にすることができた。データ構造の定義や変更は、プログレスIIの言語外の独立言語で行われる。データベースの物理構造は、ファイルとの共通性を保つために相対ファイルにデータベース辞書(ディレクトリ)を付加した形を採

っている。IBM システム/38 では、従来からのファイルを物理ファイルと称してそのユーザインタフェースを残し、データベースとしての見方を定めるための論理ファイルを定義して、論理ファイルアクセスルーチンが物理ファイルを投影するような実現方式を取っている。このように、プログレスIIもRPGIIIも従来からのファイルアクセスのためのインタフェースを残しつつ、データベース処理機能を実現している。今後は、ORACLEのSEQUEL2のようなデータベース専用の言語も普及して行くものと思われる。

以上に記述した通り、データベースにおいて、データの見方を物理的条件から解放できる機構は、言語インタフェースをよりエンド・ユーザ向けに簡潔にできる可能性を持っており、応用プログラムの開発作業の負荷の軽減をさらに推し進めるものと考えられる。

#### 4.2 オフィス・オートメーションへの対応

事務用簡易言語の動向を探る上で、今後のオフィス・オートメーション(以下OAと略記する)機能との関連を無視することはできない。OAの定義とその具体的機能については触れないが、OA機器としてその一役を荷う計算機として見ると、従来からの大量データを高速処理する役割に加えて、日常の人間行動により近いマンマシンインタフェースでデータの出入力や他のOA機器との通信を行う中枢となる必要がある。具体的には、最近オフィスコンピュータなどで常識となった漢字を含めた文書処理(ワードプロセッシング)およびグラフ処理に加えて、画像処理、音声入出力、手書きOCR入力、ファクシミリ伝送データのファイル化などを、従来からのデータ処理に融合させたマンマシンインタフェースの実現があげられる。このための基礎技術としては、大量データの高速伝送のためのローカルエリアネットワーク(LAN)、アナログデータ(イメージデータ)の必要に応じた圧縮を含めてのデジタル化などが求められてくる。これらに対処するために応用プログラムの負担は大きくなる。これを軽減するために簡易言語に対しても、新たな機能の表現が求められている。現在、これらの内の、ディスプレイ装置とハードコピー装置を媒体としたグラフ、図形などの処理のための表現手段が具体化されつつある。この場合、OA機器として重要なのは、従来からのデータ処理の世界と図形や文書処理の世界を遊離したものとしないことである。そこで、図形パターンの作成や編集とそのためのデータ入力のインタフェースがデータ処理や文章作成処理のための共通手段となる

べく、図形処理をパターン作成や編集のための独立言語として設計、図形パターンにデータを挿入して表示するインタフェースを共通マクロの形で用意することが一般的になってきている。今後は、データ、帳票、グラフなどのイメージパターンを広義に見たデータ定義として、処理系の言語とは別に設けるといことが一つの傾向となろう。

一方、従来のデータ参照は、簡易言語に限らずホストプロセッサに直結したファイルやデータベースを対象としてきた。OA化に伴うローカルエリアネットワーク技術と広域ネットワーク技術の発達に伴い、回線網に分散したプロセッサのどこにデータがあるかを意識せず、ローカルデータを扱うのと同じ言語インタフェースでデータを参照できるような分散型データベース指向の事務用簡易言語の追求も今後の課題となろう。ユニバックのMAPPER 1100<sup>14)</sup>,<sup>15)</sup>は、ホストコンピュータに回線接続された端末から、ホストのデータベースを端末ユーザのためのデータキャビネットと見立てて、データの参照、入力、編集、報告書作成、グラフ作成などを簡単なコマンドで操作できるもので、一つの方向を示すものである。

**謝辞** 本論文を執筆するにあたり、情報処理学会機関誌編集委員会で収集された、事務用簡易言語に関するアンケート調査資料を参考にさせて頂いた。このアンケート調査にご協力頂いた関係者に謝意を表す。また、本論文の査読者からも種々有益なご教示を賜わった。併せて感謝の意を表す。

## 参 考 文 献

- 1) 魚田勝臣：事務処理用簡易言語，情報処理 Vol. 22, No. 6, pp. 545-549 (1981).
- 2) IBM: IBM System/34 RPG II Reference Manual, SC-21-7667.
- 3) IBM: IBM System/38 RPG III 解説およびプログラマーの手引き, SC21-7725-3.
- 4) UNIVAC: UNIVAC 1100 シリーズ IDOL 11 解説書, 481205342-1.
- 5) 三菱電機: MELCOM 80 日本語 DPS III プログレス説明書 LM-SR 23-04A.
- 6) 日本電気: NEAC システム 100/150 ITOS アプリケーションユーティリティ SMART 説明書.
- 7) 富士通: FACOM UNIOS TASKFORCE 文法書バッチ処理編 20 SP 2500.
- 8) 富士通: FACOM UNIOS TASKFORCE 文法書ピリング処理編 20 SPO 160.
- 9) 富士通: FACOM UNIOS TASKFORCE 文法書オンライン処理編 20 SP 2700.
- 10) 富士通: FACOM UNIOS TASKFORCE 文法書 DES 編 20 SP 2800.
- 11) 富士通: FACOM UNIOS DRS 説明書 (データ検索・帳票作成ユーティリティ) 20 SP 5900.
- 12) RSI: ORACLE Users' Guide Version 3.0, RSI-100.
- 13) 三菱電機: MELCOM 80 DPS II データベース管理システム DMS-3 説明書 LM-SR24-16A.
- 14) UNIVAC: UNIVAC シリーズ 1100 MAPPER 1100 ベーシック・ファンクションガイド 481203401-0.
- 15) UNIVAC: UNIVAC シリーズ 1100 MAPPER 1100 解説書会話機能編 481205450-2.
- 16) IBM: IBM システム/38 制御プログラム機能 (CPF) 解説書—データ記述仕様 (DDS) SC21-7806.
- 17) 魚田勝臣, 溝口徹夫, 小碓暉雄, 富沢研三: オンライン処理志向のプログラミング言語, 情報処理学会論文誌, Vol. 21, No. 6, pp. 433-441 (1980).

1) 魚田勝臣：事務処理用簡易言語，情報処理 Vol.

(昭和58年6月15日受付)

