
 講座

データベースの実際 (4)

—人事管理システムからのアプローチ—

熊谷 東 愷*

1. はじめに

コンピュータの急激な発達と普及は金融機関では、営業店とのオンラインによる即時処理の全盛をみたが、反面集中された膨大なデータの加工技術が、それに伴って進歩しているとはいえない。バッチ処理の段階では、データは何らかの基準の下に厳選されてコンピュータ処理にかけられていたが、データ発生源からの直接インプットが進むにつれ、その処理技術は従来のもものと自ずから異なってくる必要がある。

データの新鮮さを損わず、有効な資料を入手する手段として、最近、いわゆるデータベース体制の試行が内外で数多く行なわれている。

データの無制限な増加と外的環境の変化のなかで、電算セクションには、そういった有形無形のインパクトがあるが、理論面でのデータベースを実際に企業内に導入するには多くの克服すべき問題点がある。特に最初の段階に於ては、問題点は予想外の摩擦係数を有し、その最大摩擦係数の突破こそ、データベース化への一番の難所であるとの認識から、我々はほぼ一年半をかけてプロトタイプともいべき一つのシステムを作り上げてきた。コンピュータ技術面に於ては、特筆すべき手法を用いた訳ではないが、データベース化の橋頭堡を築くまでのステップをここに要約してみた。

2. 導入の背景

2.1 パイロットモデルとしての人事システム

一般に企業内にコンピュータが導入されると、まず給与計算作業から着手するのが定石となっている。それには、それなりの幾多の理由が考えられるが、我々の EDP 部門も昭和 36 年、IBM 1401 機を導入とともに給与システム・人事システムを開発し、それ以

来、IBM 1410、IBM 360、IBM 370 とその改善を図ってきた。その間、プログラム、データは量的にも、内容種別的にも著しく増加したが、いずれも単純な集計とプリント作業が大半を占め、基本的には手作業のコンピュータによる代替の域を出ていない。

ほぼ十数年にわたっての人事セクションからの要請に基づき、新しいアウトプットを追加してきた関係で、無数のサブシステムが存在し、現在のダイナミックな経営にそれらを有機的に動かすことが非常に困難になってきた。これは偏に、人事システムだけに言える事ではなく、コンピュータのあらゆるシステムが体型を無視した、いわゆる積上げ方式で開発されてきた帰結であるが、ここで情報処理の新しい手段としてのデータベース検討は、時代の趨勢となってきた。

我々はデータベースの実際的な経験を積む事により、将来の本格的なデータベース開発のリスクの軽減を図るために、コンピュータ導入時に給与システムから始めたのと同じ理由で、人事システムのデータベース化を一つのパイロットモデルとして選択した。パイロット・システムの条件は種々考えられるが、次のような面で人事システムはモデル・システムとして適当である。

(a) データのインプット、アウトプット面とも、直接社外とのかかわりあいを持つ事がなく、顧客への影響が全くない。(CLOSED SYSTEM)

(b) データ量が適当である。(48 年 10 月現在約 6,000 人)(入力システムが単純)

(c) データの種類、アウトプットの計表が非常に多い、いわゆる少量多種システム。

(d) 開発期間が比較的短く、データベース化が完成すると直ちにその効果が期待できる。

(e) マネージメント・セクションの参加……企業の経営戦略と不可分の人事問題解決の助けとしてデータベースが機能すると、いわゆるミドル・マネジメ

* 住友信託銀行(株)事務管理部

ント、トップ・マネージメントとコンピュータの距離を著しく縮める。エグゼクティブの参加は、データベース・システム成否の不可欠な要素でもある。

2.2 従来の人事システム

前述のように、我々のコンピュータ利用の歴史は、ある意味では人事システムの歴史に象徴される。ハードウェアの変更、オペレーティング・システムの変更にできるだけ合わせてきたものの、開発のタイムラグから次第に一貫性を欠いてきている。そのため、一定の要請された資料作成の前に、一連の関連サブ・システムを動かしてデータの加工を行なう必要があり、多くの熟練した人員の介入とマシン・タイムを必要とする。たとえば、ファイル自体、多元管理の下に置かれているため、フィールドの重複部分が非常に多く、職員の転勤に伴うマスター・ファイルの変更で、実に十数本のテープ・メンテナンスが必要となっている。

2.3 データベース・システムを用いた統合化

将来的なトータル・システムへの指向からみて、システムの統合は必然の帰結であるが、人事管理システムのみをとり上げてみても次のような固有の特質を有して、プログラム面、ファイル面の一元化の必要性が考えられる。

(a) 人事政策

企業の新しい戦略がうちだされるとそれに伴い、ダイナミックな人事が行なわれる必要がある。政策の変

更は、現在の金融情報の下では極めて流動的であり、プログラム面、データ面、コード面の対処が可能でなければならない。特にデータ面においては、その価値観の変更さえも生じてくるので、めまぐるしい追加、削除がシステムマティックにスムーズに吸収される手段を講じておかねばならない。この面での詰めを怠ると、再びオールド・システムの轍を踏むことになりかねない。

(b) トップマネージメント

コンピュータ・アウトプットの利用者が直接、トップマネージメントにつながっている。そのため、アウトプットの見やすさと正確性が、システムへの信頼観をかちとるためには不可欠である。

(c) 緊急性

トップマネージメントの意思決定は非定型的なものであり、データ面、プログラム面でのアクションが直ちにとられる事が必要である。

(d) ファイルの多様性と重複

コンピュータによる人事管理の限界を少しでも広げるためには、一職員の多面的な分析が可能なシステムである必要があり、データの集積を急ぐあまり同一データの重複と半完成品ファイル(中間ファイル)の増加を起しやすい。

(e) 機密保持

ファイルを分散すればするほど、機密の保持は困難になる。機密保護の確立のためにも、一元化されたプロテクション・システムが要求される。

2.4 データベースの必然性

(a) データ処理に対する要求の高度化

ユーザー・セクションの情報処理技術の高度化に対応して、EDP 部門への要求も単なる集計表でなく、多種類のデータを組み合わせた分析資料やシミュレーションが求められてくる。

(b) 情報量の増加と高性能コンピュータの出現

更新のタイミングにより、新鮮なデータを陳腐化しないようにするためには、情報の整理統合を行なっておく必要がある。一方、高性能コンピュータの出現により記憶手段の改善が行なわれてきていて、従来と異なったファイル管理技術の下でない、それを有効に生かす事ができなくなってきた。

(c) 適用業務の統合化

個々の業務をいわゆる STAND・ALONE・SYSTEM をもって稼動させるには、今後ますます時間とコストが高んでくる。また、外部要因として企業自体

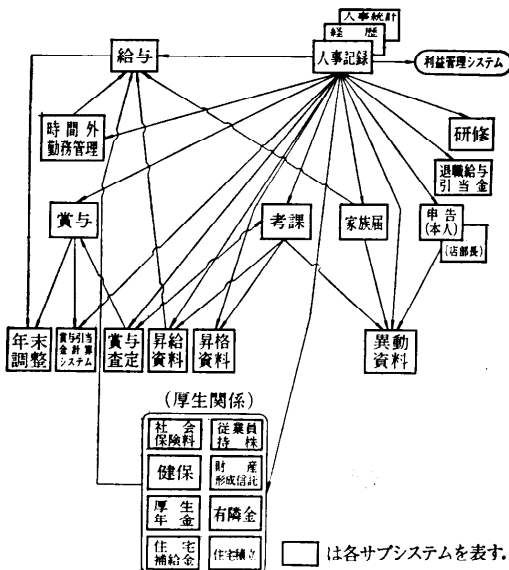


Fig. 1 従来の人事システム

各種のサービスを組み合わせたシステム商品の時代をむかえつつあり、従来のスタンドアローン・システムでのカバーが困難になってきている。

(d) プログラムコストの増大

ハードウェアの発達と比べて、プログラム専門家の養成は至難の事となりつつある。このままでは、専門化不足は解決のめどがなく、コスト面でもソフトウェアの開発コスト、メンテナンスコストが企業収益の重圧となってきている。

(e) 情報の即時性

問題の提起から解決までのレスポンスを短くするには、たんにオンライン網の確立だけでなく、日常言語でのファイル処理を可能に（問題提起者自身がファイル・アクセスできる）する必要がある、そのためには論理的な記述が自動的に物理的なアクセスを行なうような高水準言語が不可欠となる。

3. 導入と定着化

3.1 検討から決定まで

(a) マネージメント・セクションへのアプローチ
データベース・システムが必然の結果として新しいマネージメントの確立を究極の目標とするものである限り、全経営的な性格を有してくる。この事は、実際に個別問題を取上げて具体化してみると良く分る。特に EDP 部門が主導権をとってデータベース・システム（今回の人事システム）を推進していく場合、このような項目に対する人事を始めとする関係部門トップのコンセンサスを得る事が開発以前に大変重要な事となる。

① NEEDS

USER 部門
EDP 部門

② 開発

規模
期間

③ マンパワー

専任者の確保 (USER 部門, EDP 部門共)
教育

④ コスト

イニシャル・コスト
ランニング・コスト

⑤ システム

コンピュータ・リソース
移行

INPUT/OUTPUT

運営体制

⑥ 将来の EXTENSION

この面で発生する問題解決に当り、USER 部門(人事部)と EDP 部門の担当者レベルの検討と並行し、両部門のトップへの働きかけに初期の段階では特にウエイトをおいた。

(b) コストの検討

データベース・システムの導入に当っては、当社におけるシステムに対するコストの概念と異なる方法で判断していく必要があった。

① 従来のシステムのコスト概念

Fig. 2 のコスト見積書をベースとしてコスト計算を行ない、ほとんどの場合、それに見合う人件費削減のメリットとのバランスでコストの大小を計ってきた。

② データベースにおけるコストの概念

従来のシステムに比して合理化メリットによる経営削減よりも、経営の意志決定に必要とする資料をいかにカバーし、それからのメリットをどう評価するかという点で、単純な原価計算方式でのコスト概念の採用は不適當である。特に、事務産業ともいえる金融機関における人事管理システムを考える際、人員増加の抑制、事務コストの軽減、生産性の向上を推進するため機械化集中化の合理化も必要であるが、同時に財務のデパート指向のため、広範な営業活動を行なっている信託銀行の特殊性から、各専門セクションにどのように人を配置し、かつその能力を十分に活かしていくかが経営のポイントであり、データベース導入で得られるメリットを正しく評価する必要があった。

(c) ソフトウェアの検討と決定

① 検討の前提条件

○市販のソフトウェアの中から選定

○バッチ処理

開発期間とコスト及びパイロット・システムであるという点から、バッチ処理システムでも動かせる事

○リソース

○中型コンピュータで処理可能である事

○使いやすさ

USER 部門で容易に使える言語を有するシステムである事 (特に IR の容易性)

○データ、プログラムの互換性

通常のオペレーティング・システムの下でも処理可能なもの

○汎用性 発展性

機械処理コスト見積書				金額		
システム開発費・修正費						
調査分析	@	_____千円 × _____人 × _____日		_____千円		
システム、帳票設計	@	_____千円 × _____人 × _____日		_____千円		
プログラム作成	A	3,000 枚以上.....	_____本	} _____千円		
含む {	B	1,000 "	_____本			
	C	500 "	_____本			
	D	250 "	_____本			
	E	100 "	_____本			
機械費用						
パンチ	@	_____千円 × _____人 × _____日		_____千円		
総合チェック		(小計)		_____千円①		
フォーム	@	_____円 × _____枚		_____千円		
帳票 (OD)	@	_____円 × _____枚		_____千円		
パンチ	@	_____円 × _____枚		_____千円		
移行人件費用	@	_____千円 × _____人 × _____日		_____千円		
機械費用	@	60 千円 × _____時間		_____千円		
諸雑費		_____千円		_____千円		
		(小計)		_____千円②		
		イニシャルコスト合計①+②		_____千円		
定例作業 (6カ月間)						
機械費用	@	60 千円 × _____時間		} _____千円		
(プリントのみ)	@	20 千円 × _____ "				
パンチ	@	_____円 × _____枚		_____千円		
磁気テープ	@	5 千円 × _____本		_____千円		
フォーム	@	_____円 × _____枚		_____千円		
帳票 (OD)	@	_____円 × _____枚		_____千円		
オペレーター (チェッカー)	@	_____千円 × _____人 × _____時間		_____千円		
諸雑費		_____千円		_____千円		
		定例作業合計		_____千円		
(参)	(1月当り)	(1日当り)	(プログラム作成費算出基準)			
人件費	A—250 千円	10 千円	ソース枚数	テスト回数	必要日数	金額
	B—200 千円	8 千円	A 3,000 枚以上	20 回	60 日	80 万以上
	C—150 千円	6 千円	B 1,000 "	15 "	20 "	50 "
			C 500 "	10 "	20 "	30 "
			D 250 "	5 "	10 "	15 "
			E 100 "	2 "	3 "	5 "

Fig. 2

他システムの導入、データ量の増加に対応できるもの

②検討

DB/DC システムがコンピュータ界の一つの流れとなった今、この種のソフトウェアは無数に開発、発表されている。その中から、上記条件をほぼ満足し、かつ IBM 360 で使えるものとして次の三つを選び出し、これに的をしぼってその内容の詳細を検討してみた。

結論として、我々は GIS を採用した訳であるが、上記性能比較とともに次の点も評価してのことである。

- 日本での使用実績
- 将来のオンライン・リアルタイム処理へのスミ

ズな移行が可能

それ自体、オンライン機能を有するばかりでなく、IMS への移行も DL/I, IQF を介入すれば容易に行なえる。

使いやすさとコストの面から、我々は*印の FEATURE は当面導入せずに別の手段で不足部分を補う事にした。

3.2 開発

(a) 旧システムの見なおし

新しいシステムを作成するに当り、旧システムを徹底的に分析し、システム構成・ファイル構成使用頻度重要度をはかるとともに、新システムでの開発ポイントのカバー域を把握する必要がある。

シ ス テ ム		MARK IV		GIS		IMS		
会 社		Informatic, Inc.		IBM		IBM		
機	費 用	中 位		やや高い (レンタルのみ)		中 位		
	機 械	IBM 360/25 以上		IBM 360/40 以上		IBM 360/40 以上		
	オペレーティング・システム	DOS, OS		OS, VS		OS, VS		
	Enviroment (s)	PCP, MFT, MFT II, EMFT, MVT, HASP, ASP		PCP, MFT, HASP		MFT, MVT		
	必 要 な コ ア (最小)	DOS で 48K, OS で 128K		OS で 128K		OS で 128K		
	I/O 装 置	カード, テープ, ディスク, プリンター		カード, テープ, ディスク, プリンター, ディスク上のソース・ファイル以外すべて		カード, テープ, ディスク, プリンター, ディスク上のソース・ファイル以外すべて		
	インプット・ファイル数	更新あるいは作成に照会にコーディネートされて3本		更新1, 修正と照会に3本		256 パールート		
	アウトプット・ファイル数	1 マスター, 10サブ・ファイル		作成に1, 1 プロセデューアにつき6ファイル		256 パールート		
	レ コ ード 構 造	Fixed, repeating (9m レベル)		Fixed, repeating (16 レベル)		Fixed, repeating (16 レベル, 任意のレベルでブランチ可)		
	論 理 構 造	線 型	○		○		○	
木		○		○		○		
能	フ ァ イ ル 構 成	シーケンシャル, インデックス・シーケンシャル (マスター・ファイル処理のみ)		シーケンシャル, インデックス・シーケンシャル		ランダム		
	フ ァ イ ル 作 成	あり		あり		あり		
	フ ァ イ ル 変 更 の 検 査	あり		あり		あり		
	フ ァ イ ル の 保 護	あり (ファイルとフィールド)		あり (ファイルとフィールド: 照会, 更新)		あり (ファイルとフィールド: 照会)		
	ユ ー ザ ー 言 語	表形式		任意形式 (シンタックス・コントロール)		会話形式 他に (DL/I, IQF)		
	処 理 方 式	バ ッ チ	○		○		×	
		リアルタイム	○		○		○	
		TSS	×		×		○	

Fig. 3 データベースシステム比較表

Basic Retrieval System				
Format Report Feature	*Extended Multi File Support Feature	Control Statement Feature	Utility Feature	*File Update and Create Feature
File Modify Feature	Hierarchie File Support Feature	Arithmetic Statement Feature	*Edit and Encode Feature	
Processing Statement Feature	*Teleprocessing Support Feature			

* は必要としないもの, または GISに頼らず別的手段でカバーしたもの.

Fig. 4 GIS Modules

①アウトプット

現在 EDP 部門から関係各セクションに渡されている人事関係諸作表の流れをつかむとともに, アウトプットの利用側にそのすべての提示をもとめ, それらが今後とも本当に必要なものか否かのサーベイを行なう. (Fig. 5)

②データ

多くのサブシステムのそれぞれのマスターファイルの内容を列挙し, 重複しているデータの発見と管理コードの確認を行なう. また, この TABLE をもとにアウトプットのサーベイ結果と突合し, 不要なフィールドの削除と将来必要とみなされるデータ追加の検討

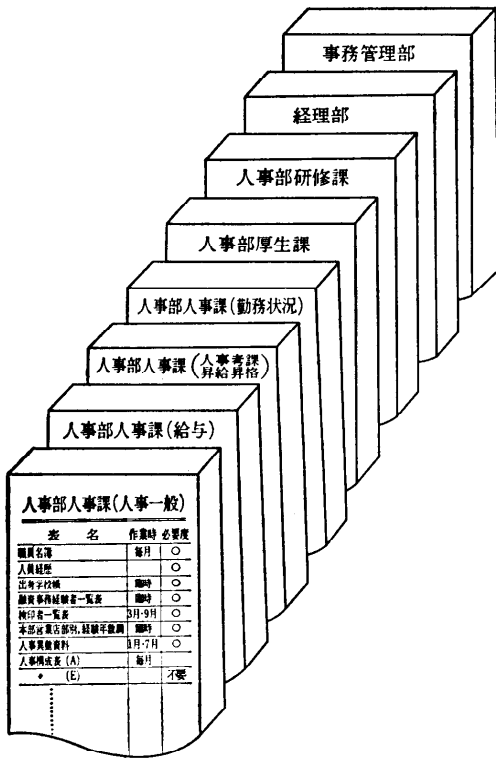


Fig. 5 アウトプット使用調査表

を行なう。(Fig. 6)

③プログラム

○オペレーショナルな作業分とそうでないものに分類

○GIS 言語に変換した方が良いものと、COBOL/ASSEMBLER で残した方が良いものに分類

④周辺システムへの影響

(b) スケジュールを決定するのに先だち、開発までの作業項目と順序をチェックした。(Fig. 7)

要員、プログラム本数、システムの必要性から、カット・オーバを 12 ヶ月後とみこみ、全体を 6 フェーズに分け、さらに細かい作業項目をチェックしスケジュールをたてた。スケジュールと各フェーズの作業項目は Fig. 8 のようになった。

①フェーズ(1) 導入についての妥当性検討

①STUDY

GIS, OS

②OUTPUT LIST & FIELD の検討

REQUIREMENT (SECURITY, OCCURRENCE 等)

D/B (どこまでのせられるか)

FILE 容量

現状の徹底的な分析 (PROGRAM・FILE・他システムとの関連)

②PLANNING

DATA/PROGRAM コンバージョン

新システム開発順位

①導入決定 (FEATURE の選別)

②フェーズ(2)

①REQUIREMENT の統合

②PRIORITY の決定

③MAN・POWER の確保, 教育

④FILE 設計

・FILE の分割

・HIERARCHY・OCCURENCE

・FILE CREATION 順位

⑤OS・STUDY

⑥PROGRAM・LANGUAGE の使い分けのルール化

③フェーズ(3)

①DDT (DATA DESCRIPTION TABLE) 作成
DDT コーディング

②OS SYSTEM GENERATION REL・21.0 MFT

③GIS SYSTEM GENERATION VERSION 2.0

④GIS プログラム コーディング/テスト

⑤人事データ コンバージョン

⑥一部試行

④フェーズ(4)

①GIS プログラム コーディング/テスト

②順次試行または並行処理

③OS SYSTEM RELEASE UP (REL. 21.6 MFT)

④GIS SYSTEM RELEASE UP (REL. 2.1 の試行 2.2 へ)

⑤ACCESS METHOD の変更

ISAM (DISK) ⇒ SAM (TAPE) へ

⑤フェーズ(5)

①FILE の再検討 (PERFORMANCE 面)

APPLICATION の範囲再検討も含めて⇒

(給与・賞与)のオペレーショナルなシステムの吸収

②S/370 導入

③VS/1 SYSTEM の下での GIS

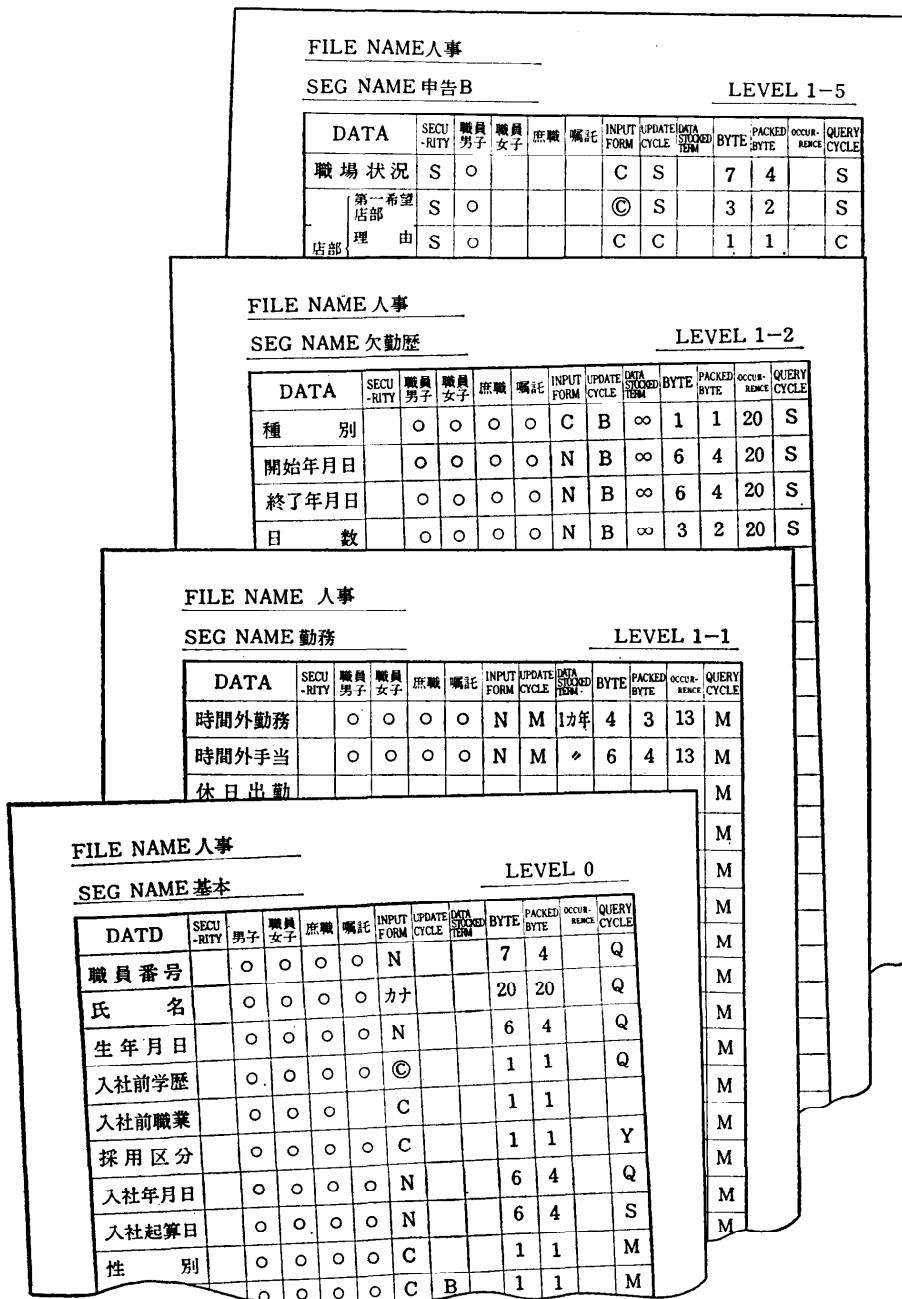


Fig. 6 データ分析表

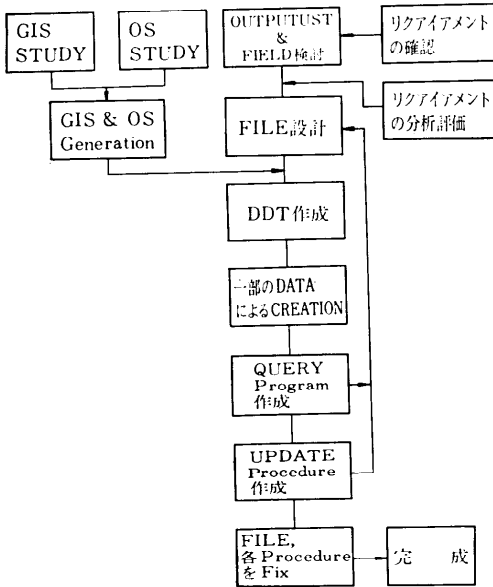


Fig. 7 INSTALLATION 作業項目

テストと VS/1 導入準備

- ④VS/1 SYSTEM GENERATION
- ⑤UPDATE PROCEDURE の確立
- ⑥フェーズ(6)
 - ①システムの基本部分開発終了
 - ②システムの改良
 - ③次期開発 APPLICATION の検討

4. GIS による人事システムの概要

4.1 概要

開発当初は、GIS による完結システムをめざし、相当高レベルなものをめざしたが、ランテストを進めていく過程で従来のシステムの完全な払拭は、実際の運営上必ずしも有効でなく、でき上がったものは後退を余儀なくされた面もある。しかしながら、以前のシステムに比べると非常に SIMPLE で使いやすいものとなり、BATCH という制約の下でも次第にその効果がでてきている。

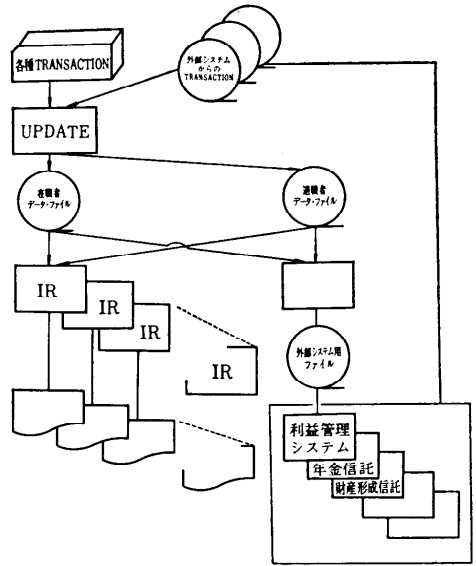


Fig. 9 システムフロー

4.2 人事システム

(a) システム

①システム・フロー (Fig. 9)

②インプット

キーパンチ方式

OCR 方式

周辺システムからの TAPE 方式

③データ

- ・在職者ファイルと退職者ファイルにわけて管理
- ・それぞれ HISTORICAL・DATA を有しそのファイルから任意のデータをとる事ができる。

④アウトプット

OPERATIONAL な OUTPUT (新システムにおいても) は主として COBOL/ASSEMBLER を使い、GIS 全面依存によるマシンパフォーマンスの低下に努めている。GIS では DDT のデコード機能により、カナ文字を大幅に採用し、EDP・OUTPUTを直接最終 USER で使用可能にした。(従来はトップマネー

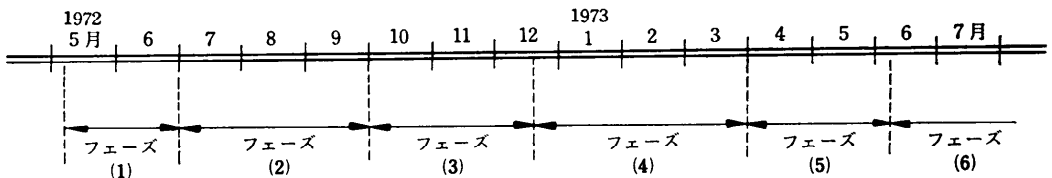


Fig. 8 全体スケジュール

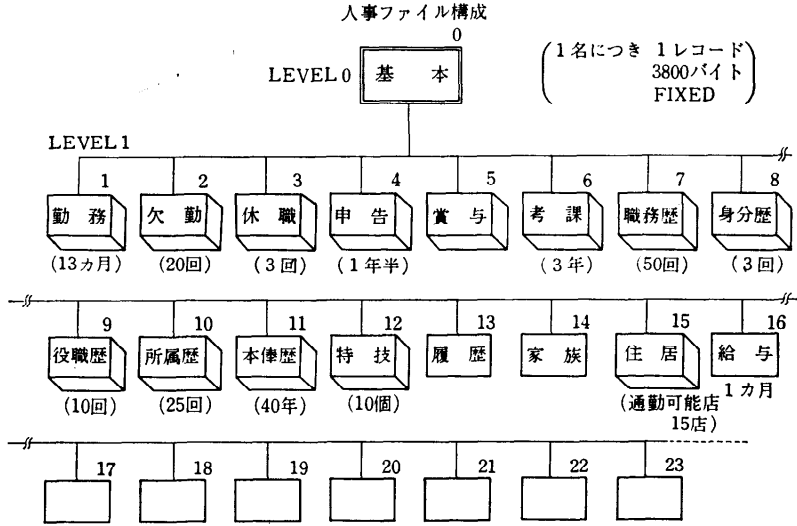


Fig. 10

ジメント向きには、HAND で転記して提出するものが多かった.)

⑤外部システムと GIS・FILE とのインターフェース

GIS・FILE を直接・他システムに連結するのは、SECURITY の問題があり、必要フィールドのみ抽出したインターフェースファイルで行なっている。

⑥プログラム

Table 1 プログラム

言語	GIS	COBOL	ASSEMBLER	摘要	開発セッション	
					EDP 部門	人事部
メンテナンス	0本	5本	10本		15本	0本
アウトプット	9	15	5	(定例的なもの)	29	0
アウトプット (IR)	52	0	3	(非定例的なもの)	9	47
インターフェース用	0	6	10		16	0
サブルーティン及び (Table)	0	3	10		13	0
DDT	5	0	0	GIS のデータ定義プログラム	5	0
ターンアラウンド作成	4	5	6		15	0
その他	0	81	123	周辺サブシステム及び未移行分	204	0

S. 48.10 現在

⑦UPDATE・PROCEDURE

GIS の UPDATE・PROCEDURE を使わずに、DATA・CHECK・FILE 作成の部分はすべて ASSEMBLER/COBOL で行なっている。データの UPDATE・PROCEDURE は基本ファイルのセグメント単位に組立てている。一例として、OCR・INPUT

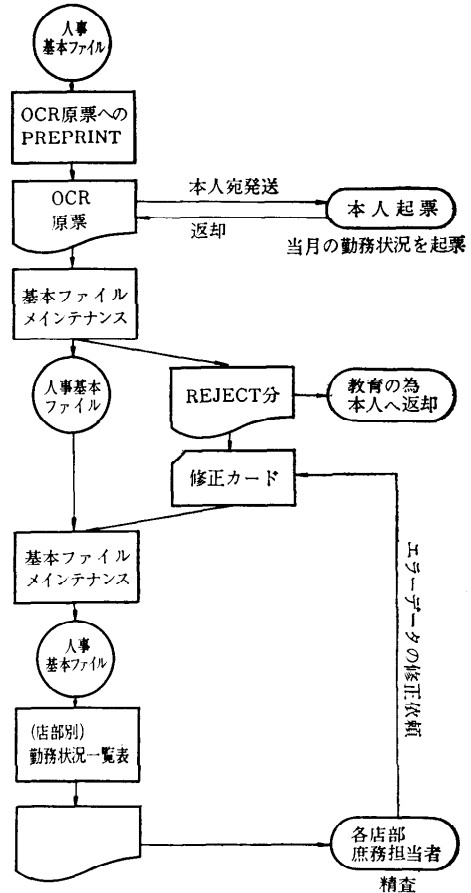


Fig. 11 勤務状況データのUPDATE PROCEDURE

をしている勤務状況データに関する UPDATE は、次のような FLOW となっている。

(b) ターン・アラウンド方式の採用

データベース・システムの設計に当って、EDP 部門の運営の簡素化のみでなく、関連部門における運営の簡素化を図るためにターンアラウンド方式を多く採用した。本方式は、単一データのターンアラウンド方式だけでなく、Fig. 12 のように多面的にその運営を行なっている。

(c) データベース化した人事ファイルの活用

- 新店に伴う人事シミュレーション
- 営業店定員管理
- 勤務状況管理
- セクション別モラルサーベイ
- オンライン効果の測定
- 教育計画
- 年金ファンドの財政計算、退職金予測
- スキルズ・インベントリー
- 人員構成予測
- 人件費予測

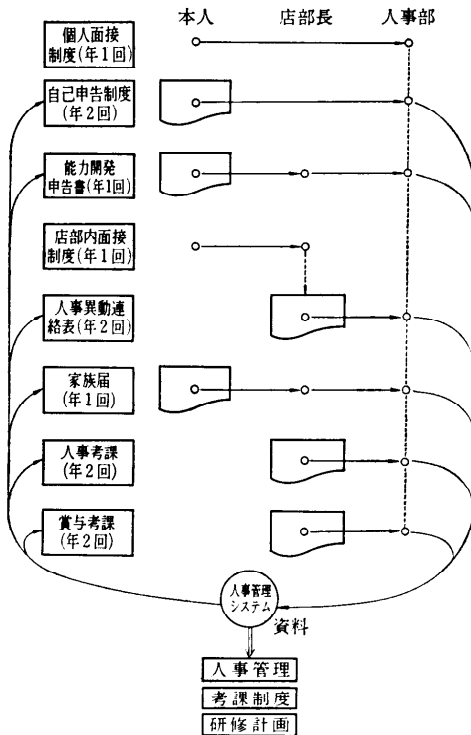


Fig. 12 ターンアラウンド方式

(例) 勤務状況管理の内、時間外統計表 (Fig. 13)

4.3 特色

(a) GIS 言語の使い方

GIS 言語を使用する者は、易しい命令で高度なロジックの組立や技術を使わなくてもよく、またオペレーティングシステムのこと等は全く知らなくても思い通りの結果を得ることができる。コーディングは単純な検索であれば5ステップ位でよく、複雑なものでも300ステップ程度である。当社における平均的ステップ数は40~50ステップと考えられる。

基本的なプログラムは必要フィールドの抜き出し (QUERY), データの並びかえ (SORT), アウトプット (REPORT) という順序でコーディングすればよく、処理時間もファイル全体を処理するのに比較すると相当早くなる。プログラム内での各ステップの区分、JCLおよび使用されるデータセットはシステムの方で準備されるため自由に使える。(Fig. 14)

Fig. 15 のプログラムは、大阪大学出身者を所属部課順にレポートする場合のコーディング例である。店部コードのカナ文字へのデコードや、プリントのエディット、フォームコントロール等は、GIS でカバーされていて、プログラマーは、必要とするデータへの検索条件の論理面に留意するだけでよい。

(b) GIS の限界と他の言語によるそのカバー

TREE 構造データセット処理において GIS では、その上位レベルセグメントのデータは容易にリトリブできるが、並列的なセグメント中のデータや、その中の各オカレンス間のデータを必要とする時は GIS プロパーでの処理はすこぶるめんどうなものとなる。レポートが複雑なものになってくると、その点で、せっかくの使いやすいという GIS 言語の特長がなくなりユーザー部門でのプログラム作成に大きな障害となって来る。我々はその面をカバーするものとして、アSEMBラーとコボルによるサブルティンパッケージを別途用意し、GIS のコーディング負担軽減を図った。またこのパッケージ内には、定例的な演算ルティン (たとえば期間計算等) も組込んでいて、使用者はあたかも GIS ステートメントの一部であるかのように、それらを使う事ができる。(パッケージ中のモジュールと GIS 言語は LINKAGE EDITOR で自動的に結合される。) Fig. 16 のレポートは GIS によるものであるが、データ構造との関係で他の言語でのカバー状態を図示してみた。

(c) 運営

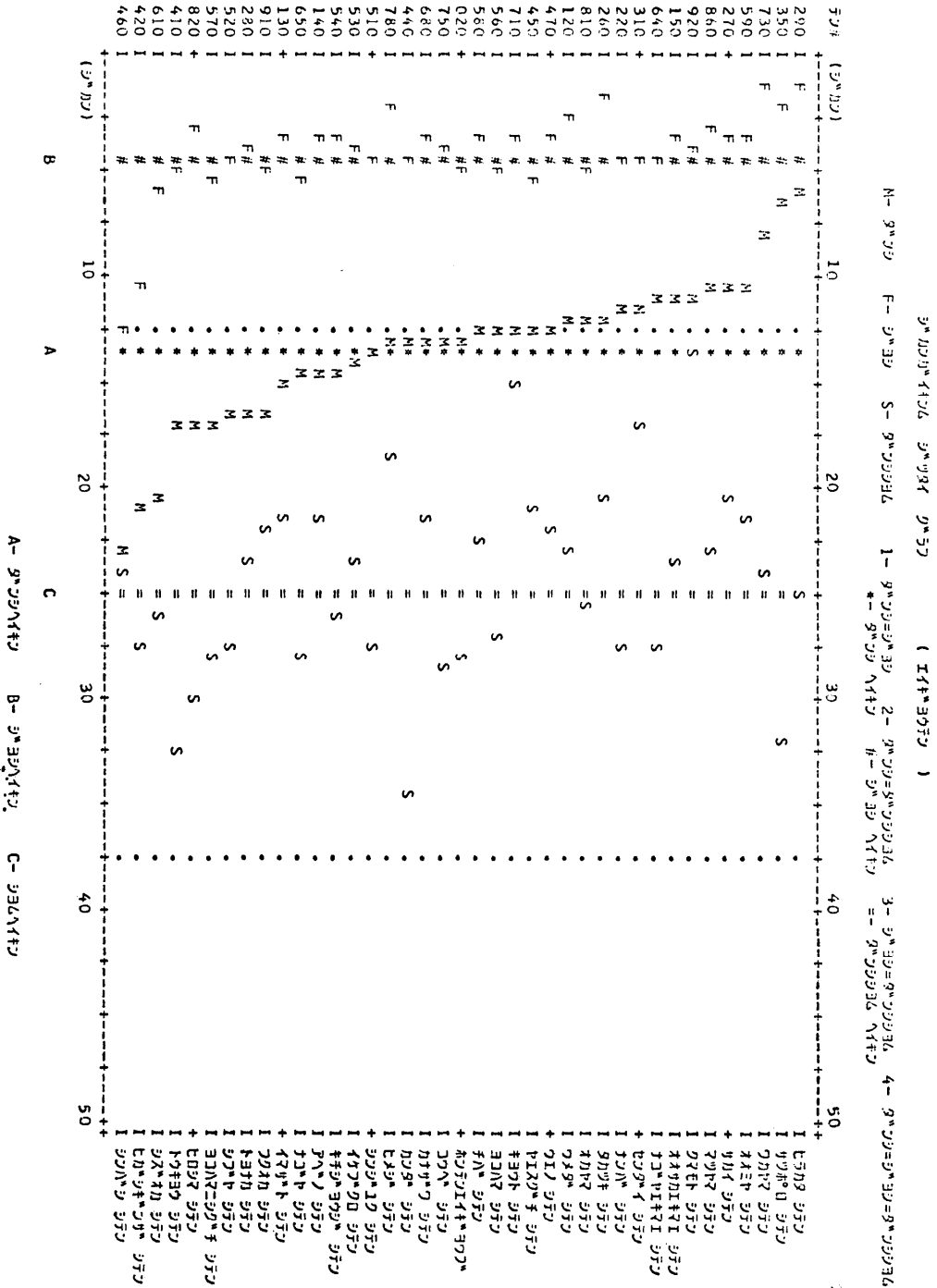


Fig. 13

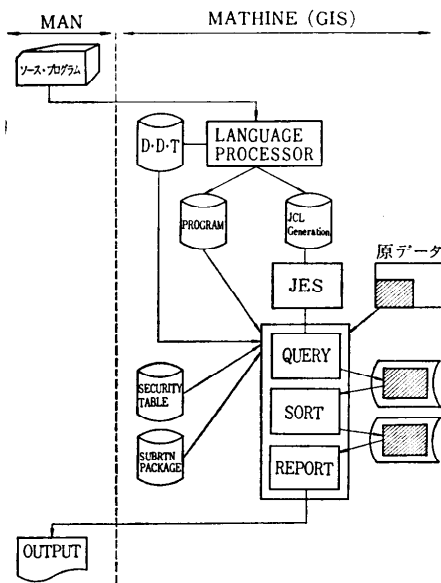


Fig. 14 データベース QUERY フロー

データベースシステムを確立する事によりファイルのメンテナンスがやり易くなった。ファイル容量が少なくなった(フィールドの重複がなくなった), 機密性が高まった等いろいろとメリットをもたらしたが、当社において最も目立って導入効果をもたらしたと考えるのは、問題が発生してからデータを手にいれるまでの時間が非常に短縮し、またプログラムを作るという意識なしに(つまり本棚から本を取りだし資料を手元にするという気易さと同じように)資料を手元できるようになった事だと考えられる。従来 EDP 部門に資料作成依頼があるとはほぼ一週間程度の作業工程が必要であったが、ユーザーの直接コーディングによりバッチ処理の下においても半日か1日でのスループットが可能になった。(なお GIS の EDP では処理時間は通常の IR であれば、コンパイルタイムに3分、データ検索時間に3分(データ数 6,000 件)程度を要する。)

5. おわりに

データベースのパイロット・システムとしては、このような形で一応の終了をみたが、今後この経験を生かし、全社的なものに拡大していく事が我々の使命であると考えます。

全体システムへ拡大していくためには、データ管理

```

* オオカ タイカク ショウジツト メイホウ カクビ
QUERY PERSON
WHEN SCHUJL EQ 'オオカ タイカク'
HOLD HOLDFL1 NAME, NYUSHA, TENBU
EXHAUST RECORD
SORT HOLDFL1 TENBU
LOCATE HOLDFL1
LIST RECORD
EXHAUST RECORD
END PROCEDURE
    
```

Fig. 15 GIS QUERY プログラムコーディング例

と SECURITY, オープンプログラマー運営, データ収集方法の改善等々の諸問題を解決していく必要がある。

一方、銀行の EDP 部門は、近年営業ベース事務処理のオンライン・リアルタイム・システムのネットワーク確立を急いでおり、当社のオンライン体制も着々とその効果を上げている。

このような背景の下に、データベースのバッチ処理の限界を将来に向けては、オンライン処理で解決を図っていくべき研究を重ねてゆきたいと考える。

このような、データベース・システムの企業への定着化は、単に EDP 部門の理論的・技術的な検討を重ねても、実際の場での成功は期しがたく、全社的な NEEDS の的確な発掘と把握が不可欠であることを確認させられた。

将来のシステムの拡張にあたり、当面 EDP 部門のなすべき事は、まわり道をしてあらゆるセクションで共通の基盤に立つての問題解決ができるような、人的底辺の拡大であり、結果的にはそれが最短距離にあるのではなかろうか。

我々は、そのような認識から PILOT・MODEL の完成時から当社の研修部門と共同で EDP 部門外に向けての継続的な研修を開始した。

この地道な小さな芽が、将来の当社の EDP 利用技

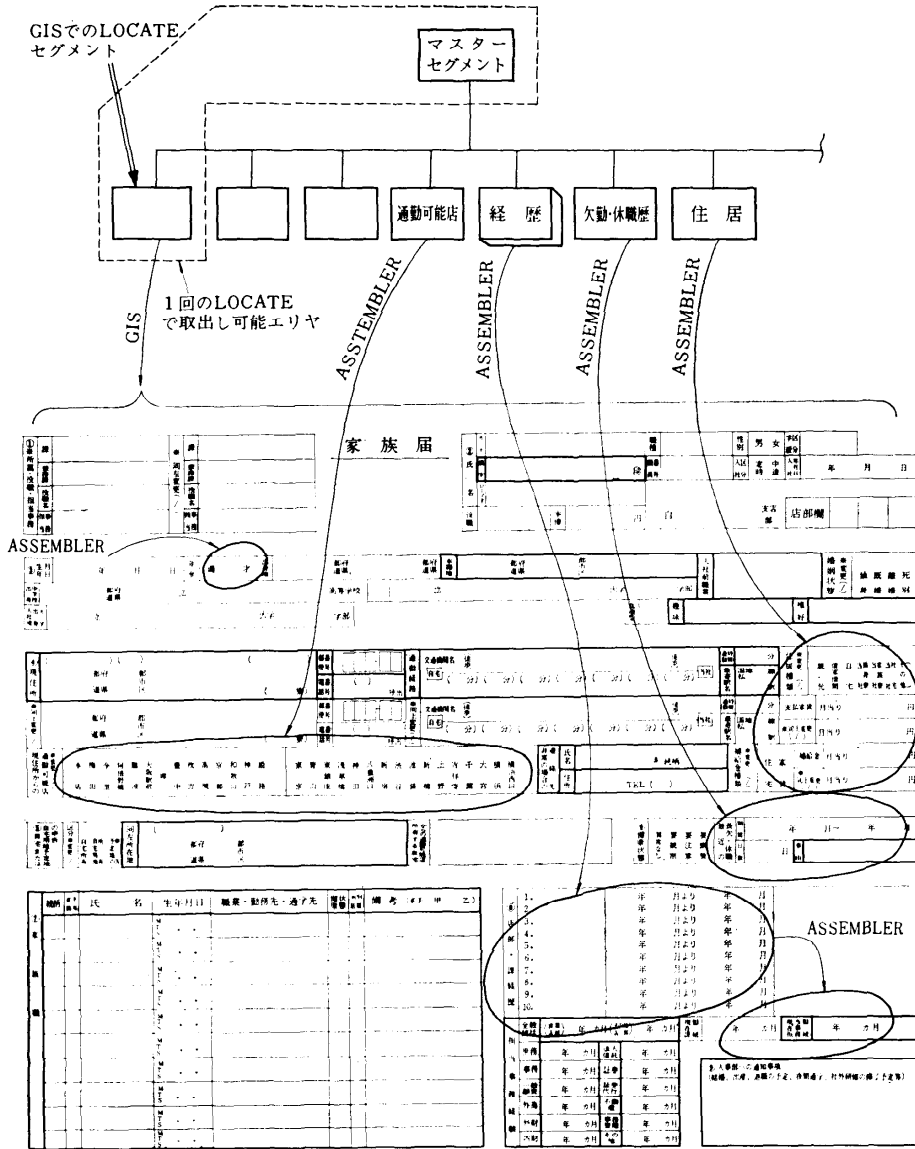


Fig. 16 データ、プログラム、アウトプットの関連

術発展に大きな原動力として返ってくる事を期待して いる。

(昭和48年10月31日受付)