

技 報

UNIVAC TECHNOLOGY REVIEW

1987年8月 第14号

特集：MAPPER システム

巻頭言

特集「MAPPER システム」の発刊によせて西原良 1

論 文

- MAPPER 概論川本光 3
- 情報化社会のシステム開発と MAPPER の有効性日置伸一 38
- MAPPERKIT におけるデータ辞書井上比佐乃 52
- 中規模製造業における MAPPER 5 の活用事例A. E. Shelton 69
- NASA における MAPPER 活用事例
 ——MAPPER と DMS データベースを用いた
 資源アカウンティングおよび利用管理システム
F. Lovato, M. W. Mickelborough 95
- トランザクション処理システムへの MAPPER の適用小西徳雄 111
- MAPPER によるユーザ・リアルタイム・システム事例浦 伸 120
- MAPPER における排他制御とリカバリ方法の考察岡島伸介 132
- 人工知能と MAPPER
 ——Northwest 航空における事例 藤隆司 148
- MAPPER と光ディスク
 ——花王(株)における事例岩佐 一夫 166

- 図書紹介176
- MAPPER パッケージ/ユーティリティ紹介181
- 掲載論文梗概表 2, 3

川本光一の MAPPER 概論は、本特集の理解に必要な MAPPER の基礎知識の提供を目的とするもので、MAPPER 登場の背景、仕組、関連ソフトウェア、利用状況、評価、将来計画等について紹介している。

システム開発部門における保守や開発のバックログは増加の一途をたどっている。この問題の真の原因は、システムやプログラムを工業製品とみなして、工業化社会の価値観で評価することにある。むしろ、人間が創る知的生産物であることに注目して、情報化社会の価値観で評価すべきである。MAPPER 導入および利用に当たっては、この新しい価値観で臨むべきである。日置伸一の情報化社会のシステム開発と MAPPER の有効性は、上記の観点に立ち、スクラップ&ビルドの意義、データベース設計の重視、プロトタイピングの有用性、MAPPERKIT の EDP 要員の教育への利用等について述べている。

MAPPER を使用したシステム開発では、開発工数が少なくなる反面、設計をキチンとしないでも目的が達成できるため、無計画なレポートがたくさん生み出される傾向がある。このため、MAPPER プログラムの作成を統一的技法で支援するツールとして、MAPPERKIT が用意されている。MAPPERKIT は、①設計技法とそのドキュメンテーション・ツール、②データベースの設計を容易にするデータ辞書システムで構成されている。井上比佐乃の MAPPERKIT におけるデータ辞書は、MAPPERKIT 1100 の概要とデータ辞書の一般論について紹介した後、MAPPERKIT 1100 のデータ辞書について述べる。

クレオール食品は、米国の Louisiana 州にある中規模の製造業であり、調味料、料理材料、料理本等を商品としている。同社は、MAPPER 専用機 MAPPER 5*を用いて、オーダー・エントリ、在庫

管理、生産管理、請求管理、購買管理、財務業務をシステム化しており、中規模製造業ながら MAPPER を十分に活用しているユーザである。A. E. Shelton の中規模製造業における MAPPER 5 の活用事例は、上記システムのデータベース設計、MAPPER ランの設計技術、効率に対する配慮等について述べている。

* 日本ユニパックの製品では、MAPPER 6 に相当する。

Slidell Computer Complex (SCC) は、米国航空宇宙局 (NASA) に属する機関で、NASA のスカイラプ計画などの支援を業務としている。SCC の業務の一つに、各部門から提出される計算資源の要求の調整があったが、従来うまくゆかず、各部門に正確な見積り用情報を提供できなかった。そこで、SCC は 1982 年に従来のシステムの評価分析を行い、MAPPER を利用しスクラップ&ビルドを行った。F. Lovato らの NASA における MAPPER 活用事例——MAPPER と DMS データベースを用いた資源アカウンティングおよび利用管理システムは、SCC のシステム資源アカウンティング・システムの開発の背景、システムの概要、設計上の考慮点について述べている。

中規模および大規模のトランザクション処理システム (リアルタイム・システム) は、処理速度の制約が厳しく、かつバッチ・システムや他のシステムとの連動が求められる。このため、これらのシステムの構築は、DB/DC を中心とした伝統的なソフトウェア体系での開発が多く、MAPPER の適用は少なかった。小西徳雄のトランザクション処理システムへの MAPPER の適用は、業務の分析によって得られた特性を利用したレポート設計を行い、MAPPER によるトランザクション処理システムを構築した事例について述べている。なお、レポート設計では、検索効率を向上させるために、レポートの複数 RID への分割等の工夫をしている。

特集「MAPPER システム」の発刊によせて

西原 良一

企業におけるシステム化要求と、それに応えるシステム開発力の不均衡は、ますます拡大の一途をたどっております。それゆえ、システム開発の生産性の向上は解決すべき最大の急務と言えましょう。最近、この需供ギャップを埋めるものとして、第4世代言語(4GL: 4th Generation Language)が脚光を浴びています。

第4世代言語の主な特徴は、①エンド・ユーザでも簡単に習得できる、②開発工数が大幅に削減でき生産性が高い、③プロトタイピングが容易である、④データベースとオンライン機能をその体系の中に有している、などであります。

本特集で取り扱う MAPPER (MAintaining, Preparing and Producing Executive Report) は、第4世代言語の概念の基礎を与えたソフトウェアで、その前身の RPS (Report Processing System) の開発は 1968 年にさかのぼります。

RPS は、Sperry 社 (現 Unisys 社) の Roseville 工場の製造部門のスタッフによって開発され、社内で使用されていたシステムです。その生い立ちが示すように、RPS ではエンド・ユーザ向きということが強く意識されていました。この RPS に大幅に機能改良を加え、1974 年に開発されたのが現在の MAPPER です。

MAPPER は、代表的第4世代言語として、全世界で約 3000 社で使用されており、わが国では、400 社のユーザに出荷されています。また、MAPPER の使用されている業種は金融、証券、流通、公共事業、製造業等、あらゆる分野にわたっております。

現在の MAPPER の 1 社当たりの平均使用端末数は 15 台強であり、小規模なシステムでは数台、大規模なシステムでは 400~500 台に及んでいます。アプリケーションとしては、非定型業務から定型的業務へと展開され、トランザクション処理量の面から見ると、簡易リアルタイム・システム、さらに中規模から大規模リアルタイム・システムの構築へと、年々その規模が拡大されております。

弊社は、このように急速に拡大しつつある顧客のニーズに応えるために、MAPPER システムのパフォーマンス、安定性、運用、機密保護の強化に鋭意取り組んでおります。

本号は、このような努力の一端をご紹介して、顧客での MAPPER の有効利用をさらに促進するために、内外より 10 編の論文を集め、特集と致しました。

掲載した内容は、MAPPER の紹介論文に始まり、MAPPER の導入およびそのシステム開発の考え方を述べたもの、MAPPER プログラムの開発・保守の自動化を支援するデータ辞書、2 編のシステム構築事例 (MAPPER と DMS 1100 を組み合わせた例、等)、MAPPER を利用したトランザクション処理システムに関するもの、MAPPER データベースの排他制御とリカバ

りについてなどであります。このほか、エキスパート・システム構築ツール KEE と MAPPER との接続、MAPPER と光ファイル・ディスクとの接続等の新しい MAPPER 応用事例も紹介しています。

また、本号では、愛知工業大学の日置先生にも論文を執筆頂き、本誌の内容を充実させることができました。

本特集が、MAPPER の導入・利用に携わっておられる各位に何等かのお役に立てれば幸いです。なお、今後とも一層、MAPPER システムの機能拡充に取り組んでゆく所存でありますので、ご指導ご鞭達のほどをお願い申し上げます。

(日本ユニバック(株) システム第2本部 システム統括3部長)

MAPPER 概論

Introduction to MAPPER

川 本 光 一

要 約 第4世代言語 MAPPER は、1974年に米国 Sperry 社 (現 Unisys 社) の Roseville 工場が開発され、社内で活用されていたもので、1981年に商品として登場して以来、幾多の機能拡充が実施され今日に至っている。

本稿では、まず MAPPER1100 の特質、エンド・ユーザ・コンピューティング思想出現の経緯から説き起こし、特徴、機能、仕組み (構造) について説明している。これにより、このプロダクトの持つ本質的なポイントについて読者の理解を得た上で、MAPPER1100 利用による効果を、利用形態別分類、COBOL との比較、第4世代言語としての評価などを交えながら解説している。さらに、今後拡充される MAPPER の将来動向についても言及している。

Abstract The fourth generation language MAPPER is a product which was originally developed and used in Roseville plant of Sperry Corp. (presently Unisys) in 1974.

MAPPER has been extensively improved since its debut as a commercial product in 1981.

The paper gives an introduction of MAPPER which starts topics in its nature by birth and advent of end user computing, and explains its characteristics, functions, and mechanism and structure.

Supposing some comprehension of a reader as to essence of MAPPER, the paper manifests its merit by showing the pattern of its usage, performance comparison with COBOL, and evaluation result as the 4th generation language.

Finally the future of MAPPER will be shown.

1. はじめに

MAPPER の前身である RPS (Report Processing System) が、米国 Sperry 社、Roseville 工場で生まれたのは、今からおよそ 20 年も前の 1968 年のことである。当時この工場では初期のシリーズ 4 × ×、シリーズ 1100 を主力商品として製造・出荷していたが、急増する需要に対応し効率化を図るために、旧来のシステムからの脱皮を目指して苦勞していた。当然、DP 部門に対しても種々改善要求が提出されたが、ご多分にもれず、部門要求に対し速やかに応えてもらえる環境になかったこともあり、何とか自分達の手で効率的オペレーションが展開できないか、今後の変化に柔軟に対応する道はないか、等について工場内で激論を戦わせた。この結果、導き出されたのが新しい発想に基づくエンド・ユーザ・コンピューティングの考え方であり、それを実現するためのツールとして 418 システムを使用した RPS が開発された。

工場内には製品の検査工程で使用するプログラムや、保守用の診断プログラムなどを開発するソフトウェア要員が多く存在したため、このようなアプローチが可能であったと言える。

その後、1974 年には、大幅な機能アップ要求に応えながら業務拡大に合わせた大規模システムとしても適用するために、シリーズ 1100 上で再開発され、この時に MAP-

PER (MAintaining, Preparing, and Producing Executive Report) と命令された。以来 Sperry 社内での普及は凄まじく、現在では 43 台の 1100 システムに 58 台の DCP* を経由して、約 30,000 台の端末をネットワークしている UNIDATS** の中核ソフトウェアとして位置づけられている。今や電話と同じ程度の自然さで MAPPER 端末が設置され、あらゆる部門の業務処理と密着した形で全社員に使用されている。

商品として世の中に出たのも、ユーザの工場見学を契機とした偶然の産物であった。

MAPPER が持つ特質のかなりの部分は、この誕生から発展・普及までの歴史の過程に由来する。MAPPER は、まだ第 4 世代言語(4GL)という言葉のない時代に生まれ、常に時の流れに合わせた実務への適用を通じて検証されながら改良が続けられてきた、今後とも進化し続けるプロダクトである。

初めから 4GL を意識し机上で仕様を決めて開発したものではなく、あくまで結果として評判となりユーザにも広く普及している事実は重要で興味深い。

時代は移り、日本 OA 協会 '87 OA 実態調査報告書⁶⁾によると、現在「企業は、社会の情報化、国際化、戦略化（知的戦略）といった波に洗われており、しかもその変化は激しい。このような社会において、企業はあらゆる活動で情報とのかかわり合いを持たざるを得なくなっている」という状況にある。一方では情報化戦略を進める上で、ソフトウェア技術者の絶対数不足、いわゆる“ソフトウェア危機”が叫ばれて久しい。コンピュータを核として深く広く進めようとするほど、要求と開発力の不均衡（バックログ）は拡大してゆく。対応策としては、通産省が中心となってソフトウェア円滑供給対策として進めているように

- 1) ソフトウェア技術者の養成……開発要員を短期的・効率的に増やす
- 2) 流通（汎用）ソフトウェアの活用……オーダーメードからレディ・メード、もしくはイージーオーダーへ
- 3) 開発の生産性の向上……部品化・自動化等を進める。Σ計画もその一環、

以上の 3 点が言われている。これらは、そのまま各企業にあっても社内の諸事情を勘案した上での努力目標とはなろう。

以上に加えて、J. Martin はその著書“プログラマなしのアプリケーション開発”の中でエンド・ユーザ・コンピューティングの重要性を強調している。これは DP 部門とエンド・ユーザ間の壁を取りはらい、業務を熟知しているエンド・ユーザにシステム開発の一部を担ってもらおうとする考え方である。彼は言う、

「DP 部門が考えるよりエンド・ユーザは、はるかに有能である」。

不確実性の時代の情報化戦略を考える鍵は、知的活動の面から柔軟で持続力のある企業情報基盤をいかに確立するか、であろう。環境の変化が大きいほどソフトウェアの短命化は進み、既存システムの延長線上での“応用動作”が求められる。そのためには、一人一人が企業活動の中でコンピュータを有効利用できる力を持つことが礎となろう。今日の企業は、まさに彼、Martin の主張に注意深く耳を傾けるべき状況にあると言える。

MAPPER は前述のように生まれ出た過程から、自然にエンド・ユーザ・コンピュー

* DCP : Distributed Communication Processor (通信制御機器)

** UNIDATS: UNified DATa Transmission System

ティング環境をオール・イン・ワンの形で提供する。既存システム (DP) と OA の融合を始めとして、さらに広がる MAPPER の世界をホスト・システムにまで至る“深まり”をも考慮したうえで、いかに企業情報システムの中に位置づけるかが差別化への道であろう。

2. MAPPER 1100 機能概要

MAPPER 1100 の特徴は、エンド・ユーザが業務知識を持ってシステム開発に参画し作業を進めてゆく上で必要とされる諸機能に加え、DP 部門によるシステム開発時に考慮されるデータ保全(リカバリ)、システム運営上の各種管理情報のロギング、暗証や使用者識別による機密保護等の諸機能をも、オール・イン・ワンの形で提供している点にある。

MAPPER は、このような特徴を持っているため、パソコンの持つリレーショナル・データベース・ソフトウェアや簡易言語を利用したシステムと比べてデータの保全性は高く、利用部門におけるディスク管理の複雑さからも解放される。さらに機密性や企業全体での統一性などの点で、ホスト管理であるゆえのキメ細かい配慮がなされている。

MAPPER1100 の機能概要をファイル・キャビネット型のデータベースと処理形態に大別して以下に説明する。

2.1 MAPPER1100 のデータベース

MAPPER1100 は独自のデータベースを持ち、図1に示すファイル・キャビネット群の概念で表すことができる。

図1にもあるように、使用者はファイル・ホルダに入ったレポートを後述する操作コマンドにより、検索・加工・編集する。レポートは、それぞれのキャビネット、引出し、ファイルに対応したモード、タイプ、レポートにより特定する。各レポートの中には複数の行 (ライン) で構成されている。

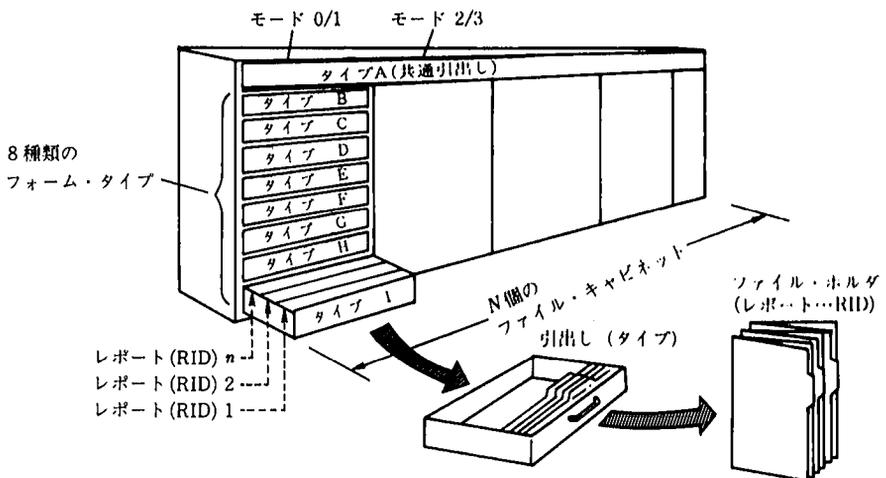


図1 MAPPER データベース概念図

Fig.1 MAPPER database concept

- 1) モード……各々のファイル・キャビネットに対応する番号で、偶数、奇数をペアで使用する。対になっているモード番号は、同一キャビネットを示しており偶数番号の時は読み取り・書き込みとも可能だが、奇数番号でモードを参照する時は読み取りのみ許される。通常、モードは部所単位に割り当てられ、使用者がMAPPER 開始操作(サインオンという)をすると、割り当てられたモードが使用できるようになる。
- 2) タイプ……各モード(キャビネット)は、各々8個の引出しを持ち英字のB~Iで特定する。これらをフォーム・タイプと呼ぶ。タイプAは特別で全モード共通のフォーム・タイプとして割り当てられ、どのモードからでもアクセス可能であり、部門を超えた情報の共有化などに利用される。
- 3) レポート……引出しの中にファイル・ホルダにより整理・格納されている帳票で、レポート識別番号(RID)により各タイプ(引出し)ごとに分類保管されている。通常、使用者は参照したいレポートをRIDにより指示する。
- 4) ライン……レポートを構成する個々の行を意味し、1行は最大132桁である。

2.2 MAPPER1100の代表的機能

MAPPER1100でのデータ処理形態には、使用者が対話形式で処理を実行する方法(マニュアル・ファンクション*)と、あらかじめ一定の手順を登録しておき、そ

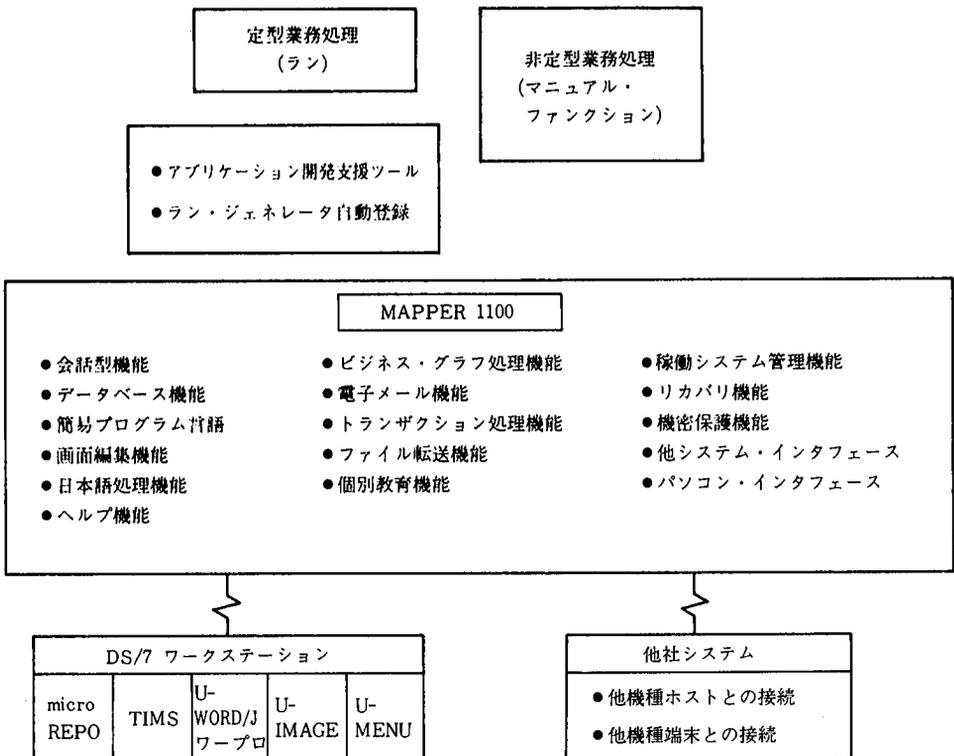


図2 MAPPER 機能概要

Fig.2 Overview of MAPPER functions

* マニュアル・ファンクションとは、会話型処理機能とも言う。

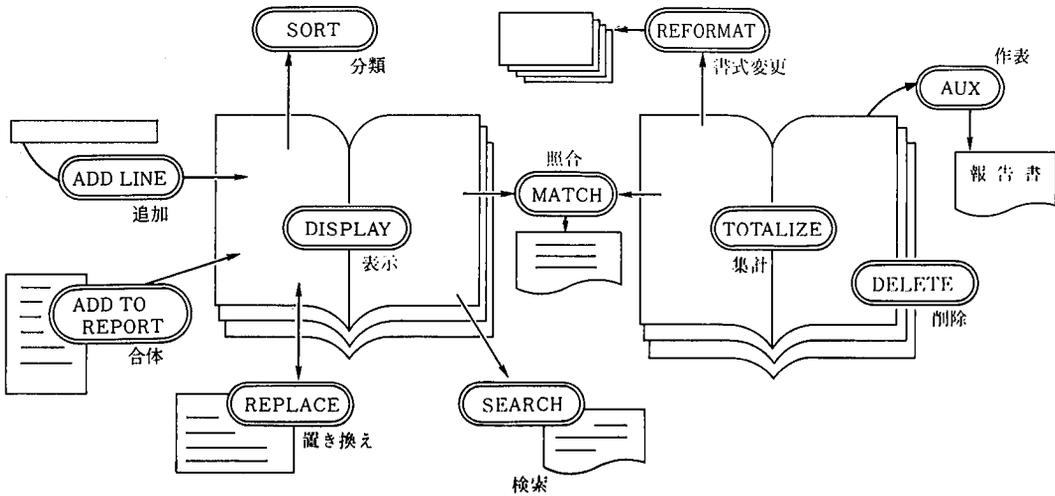


図3 MAPPERの機能(1) 指示コマンド
Fig.3 MAPPER functions(1)

れを使用し実行する方法（ラン）の2通りある。

これらの処理形態をサポートするコマンドの豊富さと相俟って、運用管理上のリカバリ、ロギング、機密保護等、エンド・ユーザ・コンピューティング環境をすべて一つのシステムで揃えており、それらを整理すると図2のようになる。

図2に示すように MAPPER1100 の持つ機能は多種多様であるが、それらの多くはコマンド形式で指示する。

コマンドの主なものを整理すると、以下のようなになる。

- 1) 制御コマンド……MAPPER の使用開始と使用終了の指示(サインオン、サインオフ)、画面コントロール(上下・左右のスクロールなど)
- 2) 検索コマンド……データの範囲指定、複数項目の AND と OR の組み合わせ、見つけた後の表示方法など指定条件に該当するデータをレポートから検索し表示するコマンド群。
- 3) 編集コマンド……ソート(分類)機能、マッチ(照合)機能、レポート中の文字列の置換機能等のレポート編集用コマンド群。レポートの表示・更新・作成・印書指示機能などを含む。
- 4) 演算コマンド……レポートの各項目間に対して、縦計・横計について小計・合計・平均・件数カウント・日付計算のような基本的な計算の他、内部関数の利用や条件による演算指示をするコマンド群。演算結果に対する編集機能も含む(右詰・左詰・四捨五入・コンマ挿入・先頭または末尾の不要な0の削除等)。

これまでのところを整理すると、図3 MAPPER の機能(1)になる。

- 5) ビジネス・グラフ……加工・編集したレポートから各種ビジネス・グラフを作成するためのコマンド群。(図4 MAPPER の機能(2)参照)
- 6) メーリング……端末(ステーション)と端末間でレポートや一時的に作成したメッセージを送受信するためのコマンド群。受信側は、メッセージの到着をブザーで知らされる。

7) 他システムとのインタフェース機能

- ① シリーズ 1100 上のプログラム・ファイル・エレメントや、データ・ファイルと MAPPER データベース間で相互にデータ転送する機能
- ② MAPPER レポート上に作られた JCL*をバッチランとして起動したり、バッチランから MAPPER ランの起動指示をする機能
- ③ トランザクション処理プログラムに MAPPER レポート上のトランザクションを投入したり、トランザクション・プログラムから MAPPER ランの起動指示をする機能
- ④ 通信回線により接続されている他ホスト上にある MAPPER1100 システムの MAPPER ランを実行し、結果を受け取る機能

8) 補助操作……ヘルプ・コマンドによる簡易操作案内の表示

他システムとのインタフェースや補助操作をまとめると、図 5 MAPPER の機能(3)のようになる。

さらに、MAPPER1100 システムを安全に運用してゆくための機能として、機密保護機能、リカバリ機能、稼動システム管理機能等も用意されている。

9) 機密保護機能……MAPPER システム自体が持つ機密保護として、次の八つの

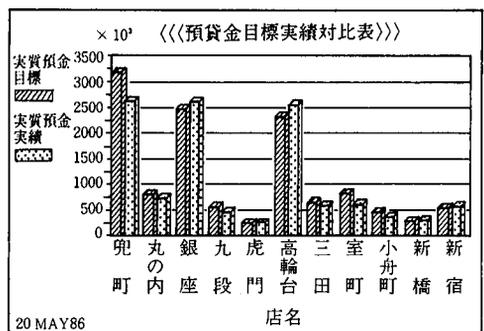
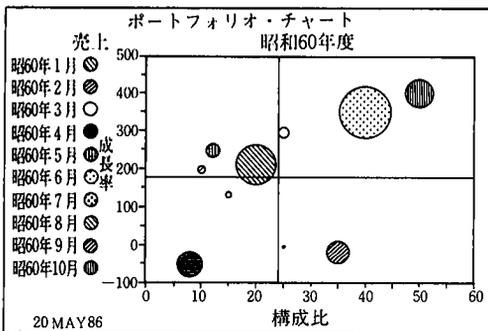
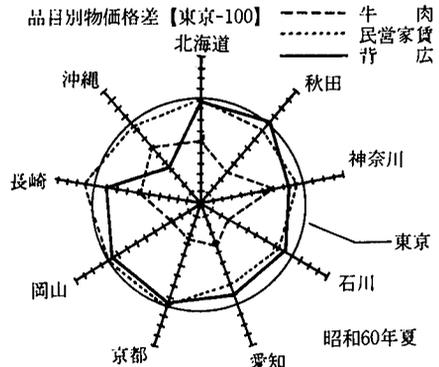
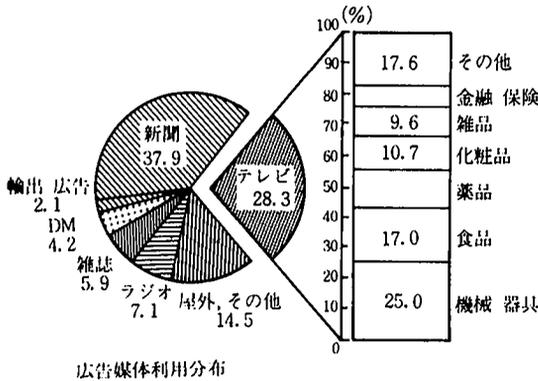


図 4 MAPPER の機能(2) 使用できるグラフ

Fig. 4 MAPPER functions (2)

* JCL: Job Control Language

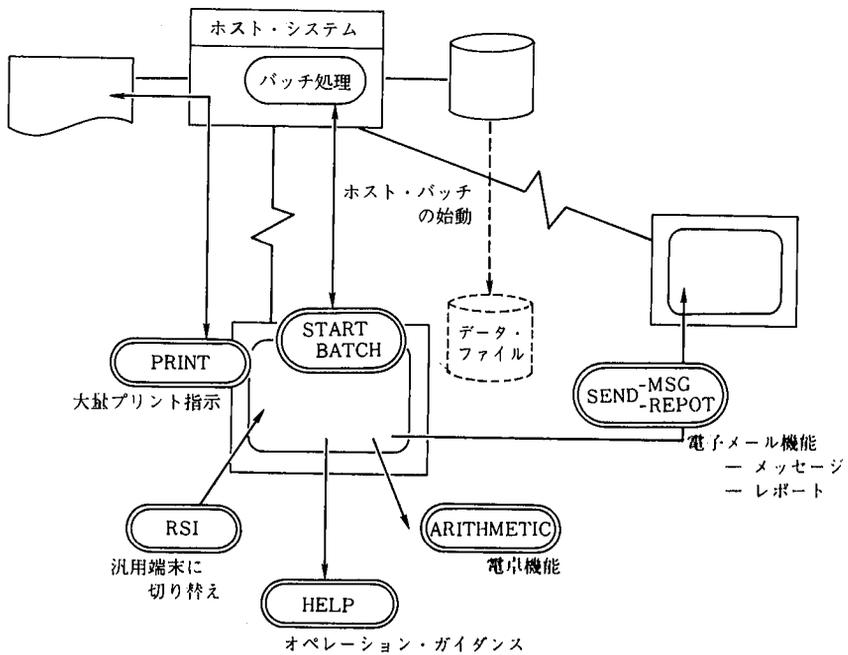


図5 MAPPERの機能(3) 他システムとのインタフェース機能

Fig.5 MAPPER functions(3)

方法（レベル）がある。

- ① サインオン時の暗証および非表示
 - ② 暗証の符号化
 - ③ 奇数モード番号の使用（リード・オンリー）
 - ④ レポートの暗証（リード，ライト）
 - ⑤ 使用者識別名による不正置き換え防止
 - ⑥ モードの暗証（モード・チェンジに対して）
 - ⑦ コマンドの使用者識別による制限
 - ⑧ ラン・ファンクションに対する規制（使用時間帯，使用端末等）
- 10) リカバリ機能……MAPPER データベースに対するデータ・リカバリの方法として，以下の三つの方法が用意されており，障害の程度に応じて使い分ける。
- ① ディスク・ファイル（MAPER 0）からのクイック・リカバリ機能
 - ② テープ・ファイル（RECVR 0）からのロング・リカバリ機能
クイック・リカバリで済むか，ロング・リカバリが必要かは MAPPER が自動的に判断する。
 - ③ 特定の RID のみのリカバリ機能
RID を指定すれば，その RID だけを最新ページ時に戻す。
- 11) 稼動システム管理機能……MAPPER1100 は自分自身の稼動情報をファイルに記録し保存する。稼動情報は以下の3種類の情報である。
- ① 統計情報
MAPPER として使用したシステム資源（メモリ，ディスク・ファイルなど）

について過去 5 分間のデータと、MAPPER が稼動してから現在までの 5 分間隔のデータが記録されており、稼動状況評価やシステム・チューニングの情報として使用する。

② MAPPER 使用状況

使用者識別名、時刻、使用した機能名、実行経過時間、参照したレポート番号、入出力レポートの行数、入出力回数等が記録されており、使用状況の詳細を時系列として把握すると共に、各使用者別の課金情報としても使用する。

③ 回線エラー情報

回線識別名、端末番号、使用者識別名、発生時刻、エラーの種類、エラー発生までに正常に入出力できた文字数等が記録されており、回線系の障害対応に加え、エラーの発生状況を分析するためにも使用する。

これらの稼動情報は、端末からの操作により運用管理者が参照し、分析してバランスのとれた、より良い利用環境を提供してゆく上での基礎情報として活用する。

ここに述べたことは、簡易習語ソフトや BASIC のプログラムの個別運用を行うパソコンの環境とは、機密保護、データ管理、情報の一元化などの面で大きく異なる点である。

3. MAPPER1100 のしくみ

この章は MAPPER1100 の内部構造を説明するもので、とくに興味のない読者は読み飛ばされたい。

ここでは、「ある営業所の売上目標金額を 10 パーセント増加する」という具体的な作業を六つのステップに分け示しながら、MAPPER1100 の内部でどのような処理をしているかをデータベースを中心に説明する。以下では、図中の(a)は使用者が行う操作であり、(b)は MAPPER1100 が表示する結果である。

- 1) レポートの表示……MAPPER のサインオン・ロゴが表示されている画面のトップ・ラインに表示するレポートの番号 11 E (タイプ E RID 11) が入力されてから (図 6(a))、そのレポートが端末に表示される (図 6(b)) までの処理とデータの流れを簡単に示すと、図 7 のようになる。

以下の(a)～(f)は、図 7 中の各ステップに対応している。

- ① 端末入力処理は、回線入力バッファに入力されたデータから回線制御文字、パリティ・ビット等を取り除き、それを入力データ・バッファに詰め直す。
- ② 命令語解析処理は入力データ・バッファを参照し、その内容がレポートの更新なのか、命令語なのか、またはランの実行なのかを検査する。その結果、それぞれの処理に対応したルーチンをファンクション・コントローラを介して呼び出す。
- ③ ファンクション・コントローラは、呼び出しを依頼されたルーチンが常駐ルーチンなのか非常駐ルーチンなのかを調べる。非常駐ルーチンの場合には、そのルーチンをロードするために自分自身が持っているバッファ (メモリ・プール) 上に領域を確保する。そしてルーチンをロードし、制御をわたす (この例の場合はレポート表示機能)。

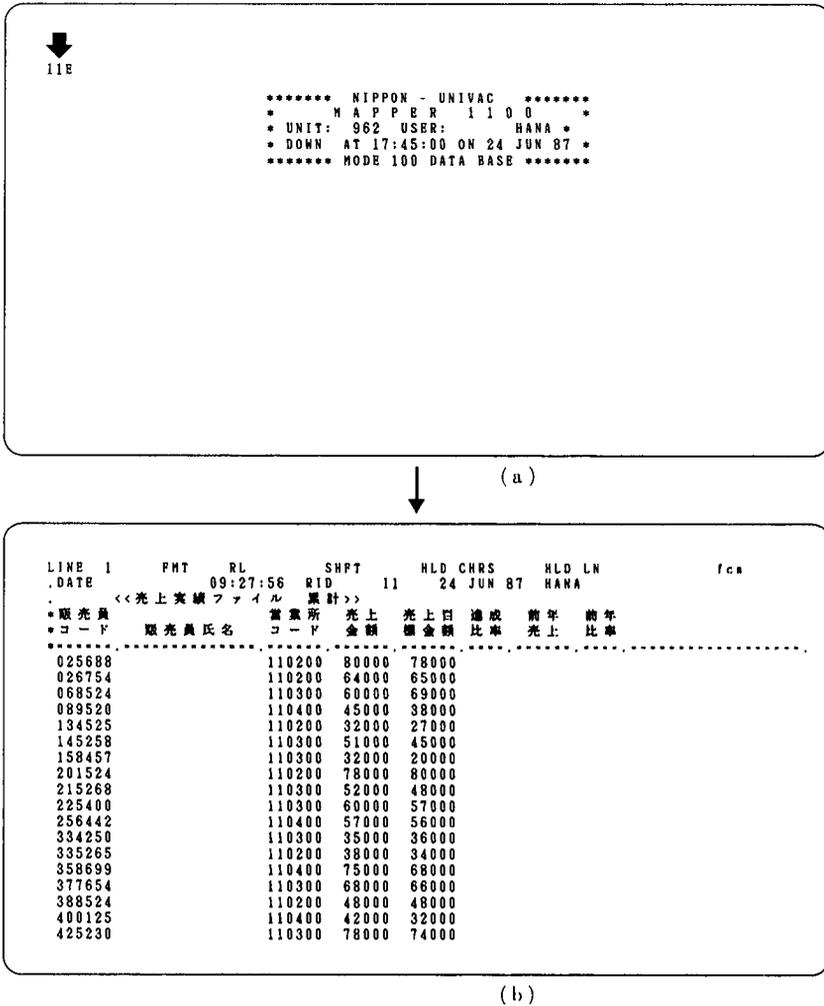


図6 レポートの表示
Fig.6 Display of the report

- ④ 制御を受け取ったレポート表示機能は、指定されたレポート（この例の場合は 11 E）をデータベースからメモリ上に画面表示分だけ読み込む。データベースの読み込みは、レポート入出力共通ルーチンを使用して行う。この共通ルーチンを使用することにより、レポート更新時の排他処理も自動的に行われる。
 - ⑤ 読み込まれたレポートは、画面制御文字などを付加し1画面分の出力データとして編集され、画面データ・バッファに詰められる。次に端末出力処理にデータ出力を依頼する。
 - ⑥ 端末出力処理では、回線制御文字、パリティ・ビット等を付加し、端末にデータを出力する。この結果、レポート（この例の場合は 11 E）が表示される。ここで、MAPPER1100 のデータベースについて図8を用いて説明する。
- MAPPER1100 のデータベース・ファイル（MAPER *n* という名前が付けられている）の形式は、OS1100 の標準プログラム・ファイルである。プログラム・ファイルは、

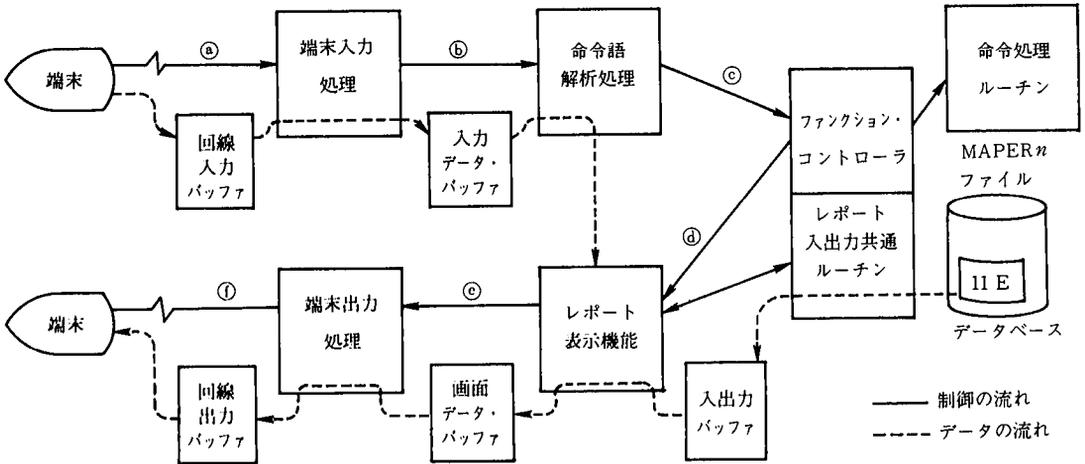


図7 MAPPER 1100 制御/データの流れ
 Fig. 7 MAPPER 1100 control/data flow

エレメントから構成されている。MAPPER1100の場合、レポートが一つのエレメントに対応している。通常プログラム・ファイル内のエレメントを参照するには、TOC (Table Of Contents—エレメントの目次) の情報からエレメントの番地を求めて入出力を行うが、MAPPER1100の場合は、初期化時にすべてのデータベース・ファイル (MAPER1~MAPER n) の TOC を読み、タイプ・テーブル、RID テーブルと呼ばれる二つのテーブルを作成する。二つのテーブルの役割は次の通りである。

- ① タイプ・テーブル……モードとタイプの情報 (データベース・ファイルの番号つまり MAPER n の n, タイプの最大レポート (RID) 番号, RID テーブルへのポインタなど) を持つテーブルである。このテーブルはメモリ上に常駐するとともに、そのコピーは、MAPPER1100 のスクラッチ・ファイル (MAPER0) 上に保存される。
- ② RID テーブル……各レポートの情報 (データベース・ファイルの番号, レポートの行数, レポートのマス・ストレージ番号など) を持つテーブルである。レポートの情報は、レポートの番号順に並べられている。このテーブルは、MAPER0 ファイル上に保存され、必要に応じてメモリ・プールに読み込まれる。一度メモリ・プールに読み込まれた RID テーブルは、メモリ・プールが不足状態にならない限り、メモリ・プールに保持される。

レポートを参照する時には、はじめにレポートのモードとタイプからタイプ・テーブルの場所を求める。タイプ・テーブルには、レポートの情報を含む RID テーブルのポインタがあるので、今度は RID テーブルを読み込む。RID テーブルには、レポートが存在するファイル番号があるので、ファイル名 (MAPER n) がわかる。また、RID テーブルにはレポートのマス・ストレージ番号があるので、ファイル名とこの番地によりレポートの参照ができる。

- 2) 特定営業所の選択 (その 1) ……特定営業所を選択するため 1 行目 (ライン 0) から SU (サーチ・アップデート・コマンド) を入力すると (図 9(a)), 図 7(b)まで処理が進められ命令語解析が行われる。SU は命令語なので、ファンクション・

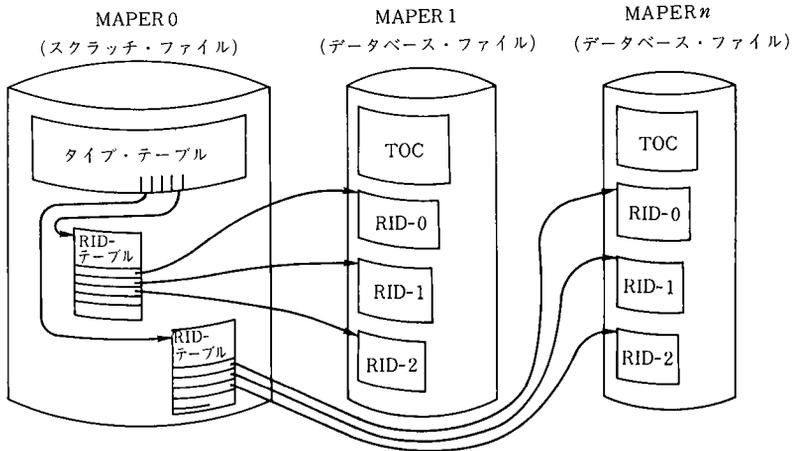


図8 MAPPER 1100 データベース関連図 (1)

Fig. 8 MAPPER 1100 database relationship (1)

コントローラを經由してサーチ・アップデート機能が呼び出される。サーチ・アップデート機能はレポートのタイプと同一タイプのRID 0 (この例の場合は0E)を1行ずつ読み込み、現在の表示方式に従ってマスク画面*を編集して表示する(図9(b))。

- 3) 特定営業所の選択(その2)……マスク画面に営業所コードとして110400が入力されると(図10(a)), MAPPER1100は、オプション行(2行目)から1文字ずつ解析を始める。80文字検査が終了したり、復帰文字(CR文字)を見つけると行の終りなので、これにより今何行目を処理しているかを知ることができる。CR文字は、80桁目に入力文字がないとき、最後の有効文字のうしろに付加される(オプション行を例にすると'D'のうしろにCR文字が付加される)。解析の目的は、(a) 指定されたオプションを得る、(b) マスクからレポートのサーチを行う位置を得る、(c) サーチするデータを得る、の三つである。

マスク画面の解析が終ると、指定されたレポートを1行ずつ読み、サーチを行う位置に指定されたデータがあるかを調べる。指定されたデータを見つけた時には、その行をアップデート・リザルトとしてMAPER 0ファイルに保存する。リザルトは、処理の中間結果として作られるレポートであり、アップデート機能で作られるリザルトをとくにアップデート・リザルトと呼ぶ。アップデート・リザルトの各行は、それが元のレポートの何行目のイメージかを記憶している。レポートの読み込みとアップデート・リザルトの書き出しは、図7のレポート入出力共通ルーチン呼び出して行う。レポート入出力共通ルーチンは、読み込みの時には数行分まとめて読み、それを1行ずつに分割して要求者に渡すようになっている。逆に書き出しの時には、バッファリングを行い、数行分まとめて書き出すようになっている。レポートのすべての行をサーチすると、MAPPER1100は作った

* マスク画面：マニュアル・ファンクションで命令指示をしたとき、最初に表示される画面で、レポートの見出し部とパラメータを入れるエリアとで構成されている画面をマスク画面という。パラメータを入れるエリアにはソートのキー、演算機能の演算子等を指定する。

↓

LINE	SU	PMT	RL	SHPT	HLD	CHRS	HLD	LN	fcs
DATE			09:27:56	RID	11	24 JUN 87	HANA		
<<売上実績ファイル 累計>>									
販売員 コード	販売員氏名	営業所 コード	売上 金額	売上目 標金額	達成 比率	前年 売上	前年 比率		
025688		110200	80000	78000					
026754		110200	64000	65000					
068524		110300	60000	69000					
089520		110400	45000	38000					
134525		110200	32000	27000					
145258		110300	51000	45000					
158457		110300	32000	20000					
201524		110200	78000	80000					
215268		110300	52000	48000					
225400									

↓ (a)

↓

<<売上実績ファイル 累計>>									
販売員 コード	販売員氏名	営業所 コード	売上 金額	売上目 標金額	達成 比率	前年 売上	前年 比率		
.....									

(b)

図9 特定営業所の選択(1)

Fig.9 Selection of a sales branch(1)

アップデート・リザルトを表示する (図 10(b)).

- 4) 売上目標金額の計算 (その 1) ……売上目標金額の計算をするため、ライン 0 から TOT (トータライズ・コマンド) を入力すると (図 11(a)), 2) 特定営業所の選択の時と同様にマスク画面を表示する (図 11(b)).
- 5) 売上目標金額の計算 (その 2) ……マスク画面に * 1.1 (1.1倍するため) が入力されると (図 12(a)), 3) 特定営業所の選択の時と同様にマスク画面を解析する。トータライズ機能の場合、解析の目的は、(a) 指定されたオプションを得る、(b) マスクからレポートの計算を行う位置を得る、(c) 計算の演算を得る、(d) 計算に使う定数を得る、(e) 計算結果を格納する欄を得る、の五つである。

マスク画面の解析が終ると、アップデート・リザルトを 1 行ずつ読み込み、指定された計算を行い新しくアップデート・リザルトを作り直す。レポートのすべての行で計算したあと、MAPPER1100 はアップデート・リザルトを表示する (図 12(b)).

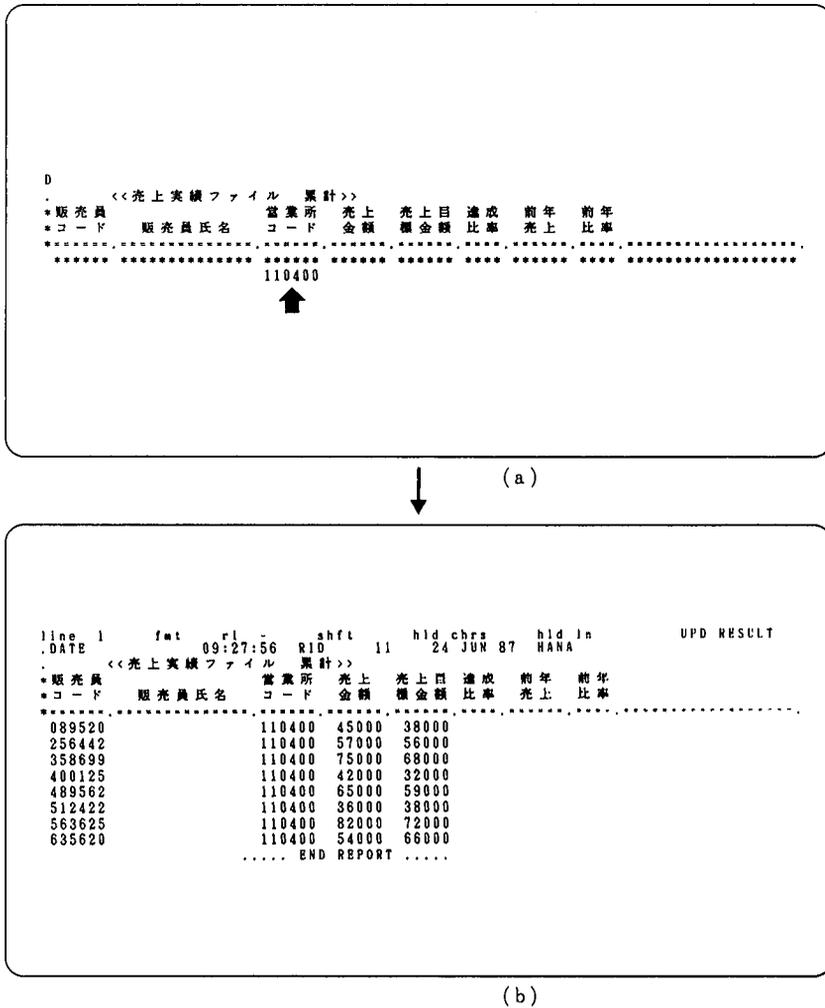


図 10 特定営業所の選択(2)

Fig. 10 Selection of a sales branch(2)

6) レポートの更新……UPD (アップデート・コマンド)を受け取ると(図 13(a)), MAPPER1100 は元のレポートのコピーを MAPER 0 ファイル上に作る。レポートは、一度更新されると MAPER 0 ファイルに移される。そして、RID テーブル内のデータベース・ファイルの番号、レポートのマスストレージの番地なども MAPER 0 ファイル上のレポートをポイントするように変更される。レポートの参照は、RID テーブルをもとにして行われるので、次回からは MAPER 0 ファイル上のレポートが参照されることになる。MAPER 0 ファイル上のレポートをデータベース・ファイルに反映させるための処理としてページ処理が用意されている。これらの関連を簡単に示すと図 14 のようになる。

元のレポートを MAPER 0 ファイルへコピー後、アップデート・リザルトを 1 行ずつ読む。アップデート・リザルトの各行は、元のレポートの何行目のイメージかわかるようになっているので、MAPER 0 ファイル上でその行を更新する。ア


```

R1
.<<売上実績ファイル 累計>>
*販売員      売上  売上  達成  前年  前年
*コード      氏名  所  金額  目標  比率  売上  比率
-----
*1.1
    
```

(a)

```

line 1      fmt      rl      shft      hld chrs      hld ln      UPD RESULT
.DATE      09:27:56  RID      11      24 JUN 87  HANA
.<<売上実績ファイル 累計>>
*販売員      売上  売上  達成  前年  前年
*コード      氏名  所  金額  目標  比率  売上  比率
-----
089520      110400  45000  41800
256442      110400  57000  61600
358699      110400  75000  74800
400125      110400  42000  35200
489562      110400  65000  64900
512422      110400  36000  41800
563625      110400  82000  79200
635620      110400  54000  72600
          ..... END REPORT .....
    
```

(b)

図12 売上目標金額の計算(2)

Fig. 12 Totalization of the targeted sales(2)

することにより、一つのディスク・ユニットへの入出力の集中が避けられ、効率の面からも有効である。

4. MAPPER1100 関連ソフトウェア群

MAPPER1100 には、本体とは別に種々の周辺ソフトウェア群が用意されている。これらの周辺ソフトウェア群を機能別に大きく分類すると、統合 OA 実現のためのソフトウェア群、アプリケーション開発支援のソフトウェア群、グラフ関連ソフトウェア群、ユーティリティ・ソフトウェア群、管理者支援ソフトウェア群、他社機器接続のためソフトウェア群、に分けることができる。さらに将来的には、他のソフトウェアとのインタフェースを容易にするためのソフトウェア群が提供される予定である。MAPPER1100 の本体と周辺ソフトウェア群の関連図は、図 15 の通りである。

次に、これらの周辺ソフトウェア群を簡単に説明する。

↓

line	UPD	fmt	rl	shft	hld	chrs	hld	ln	UPD	RESULT
.DATE			09:27:56	RID	11	24 JUN 87	HANA			
<<売上実績ファイル 累計>>										
*販売員				営業所	売上	売上目	達成	前年	前年	
*コード				コード	金額	標金額	比率	売上	比率	

089520				110400	45000	41800				
256442				110400	57000	61600				
358699				110400	75000	74800				
400125				110400	42000	35200				
489562				110400	65000	64900				
512422				110400	36000	41800				
563625				110400	82000	79200				
635620				110400	54000	72600				
..... END REPORT										

↓ (a)

LINE	1	PMT	RL	SHFT	HLD	CHRS	HLD	LN	fcs
.DATE	24 JUN 87	09:57:12	RID	11	24 JUN 87	HANA			
<<売上実績ファイル 累計>>									
*販売員				営業所	売上	売上目	達成	前年	前年
*コード				コード	金額	標金額	比率	売上	比率

025688				110200	80000	78000			
026754				110200	64000	65000			
068524				110300	60000	69000			
089520				110400	45000	41800			
134525				110200	32000	27000			
145258				110300	51000	45000			
158457				110300	32000	20000			
201524				110200	78000	80000			
215268				110300	52000	48000			
225400				110300	60000	57000			
256442				110400	57000	61600			
334250				110300	35000	36000			
335265				110200	38000	34000			
358699				110400	75000	74800			
377654				110300	68000	66000			
388524				110200	48000	48000			
400125				110400	42000	35200			
425230				110300	78000	74000			

(b)

図13 レポートの更新

Fig. 13 Updation of the report

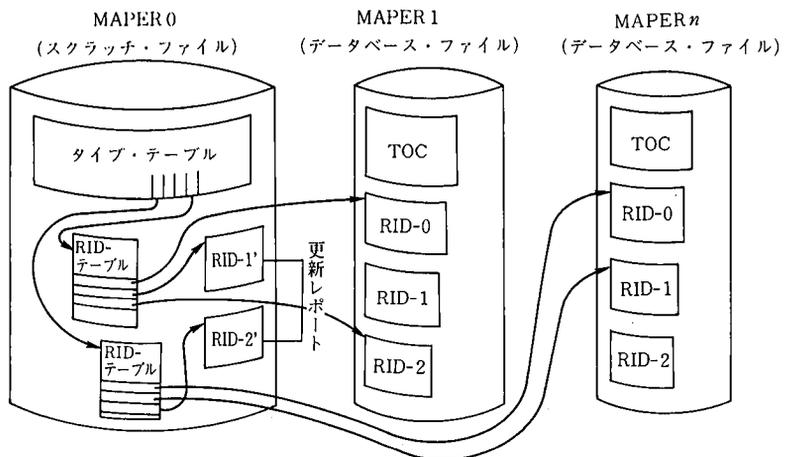


図14 MAPPER 1100 データベース関連図(2)

Fig. 14 MAPPER 1100 database relationship (2)

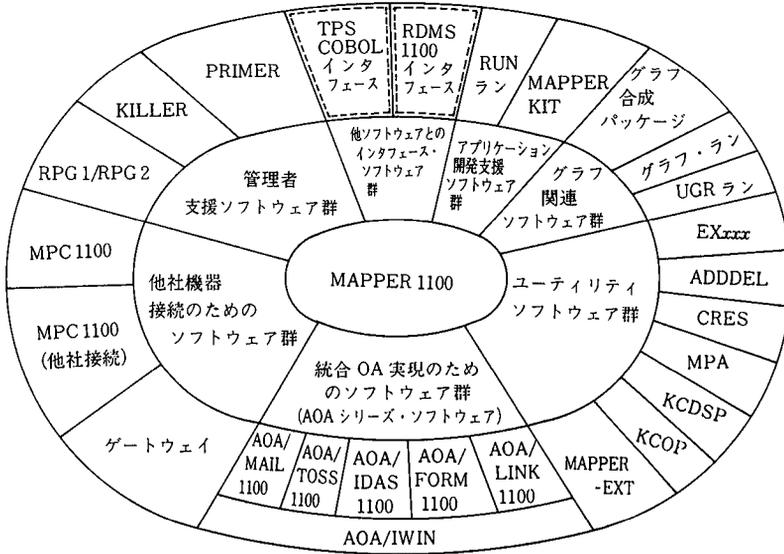


図 15 MAPPER 1100 関連ソフトウェア群
 Fig. 15 Software related with MAPPER 1100

1) 統合 OA 実現のためのソフトウェア群 (AOA シリーズ・ソフトウェア) ……
 MAPPER を中核としながらシリーズ 1100 または 2200 上で統合 OA を実現するためのツールとして、AOA (Automated Office Assistants) シリーズ・ソフトウェアがある。AOA シリーズ・ソフトウェアは、次の六つのソフトウェアから成り立っていて、全体で使いやすい OA 環境を構築するようになっている。使用者は、AOA シリーズ・ソフトウェアの入口である AOA/IWIN (ワークステーション・ソフトウェア) が表示する簡単なメニューを選択することによって、個々の AOA ソフトウェアを意識せずに必要な機能を使用することができる。AOA ソフトウェアのそれぞれの役割りは、図 16 のようになっている。

以下では、六つの AOA ソフトウェアの概要について述べる。

- ① AOA/IWIN : AOA を使用するための“情報の窓”で、ホストとかワークステーションの意識を持たずに使用できるよう、一般オフィス・ワーカーに親しみやすいユーザ・インタフェースを提供する。
- ② AOA/MAIL1100 (電子メール) : 文字通り社内メール・システムを電子化したもので、メールの作成、配布、受け取りから電子ファイリング、くずかごまで一貫して提供するソフトウェアで親展、往復、同報、回覧などの機能を持つ。(MAPPER 自体の電子メールがステーション間のメールで素朴なシステムであるのに対し、AOA/MAIL1100 は本格的なメール・ボックス思想に基づいたユーザや組織間のメール・システムである)
- ③ AOA/TOSS1100 (オフィス・セクレタリ) : 電子秘書ともいうべきもので、予定表の登録・管理、電子掲示板、電子メモ、ファシリティ予約などに加え、会議開催の支援機能も持つ。
- ④ AOA/IDAS1100 (統合文書管理) : オフィス内のさまざまな文書 (MAPPER

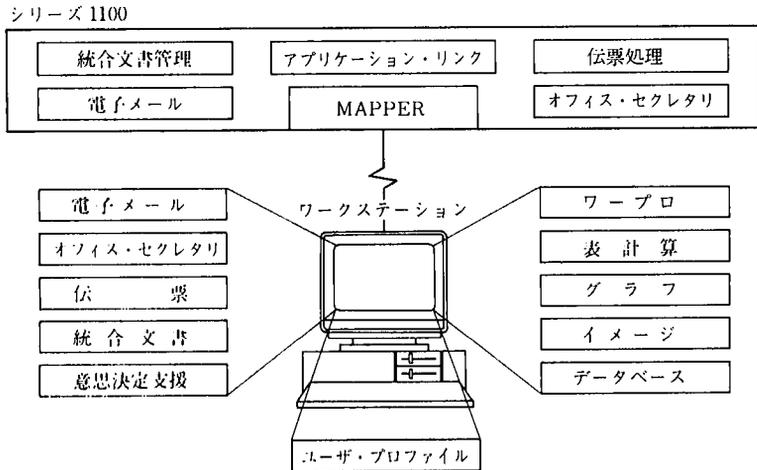


図 16 AOA ソフトウェアの構成

Fig. 16 AOA software components

のレポートから光ディスク上の文書、書籍などの外部文書などを一元管理し、目的に応じて高度な検索ができるように汎用性を持たせたもの。検索結果を電子メールと連結し配布することもできる。

- ⑤ AOA/FORM1100 (伝票処理)：電子伝票処理で、伝票の様式作成から起票、電子メールを使つての照査・承認に至るまで幅広い機能を持つ。入力チェックや各種挿入機能を使用することにより、少いキータッチでクリーン・データが得られ、指定の MAPPER レポートに格納される。
- ⑥ AOA/LINK1100 (アプリケーション・リンク)：MAPPER/AOA と外部ソフトウェアを接続するためのもので、電子メールのファクシミリへの出力や意思決定支援ソフトウェア SUFICS1100 とのインタフェースなどを提供する。ユーザの TIP トランザクション・プログラムとの連結もできるので、幅広いシステム構築に有効である。
- 2) 他社機器接続のためのソフトウェア群……MAPPER1100 は、UTS (Universal Terminal System) 仕様の端末を標準として使用するよう設計されている。このため、UTS 仕様でない端末を MAPPER1100 に接続する場合、特別な接続用ソフトウェアが必要となる。この接続用ソフトウェアは、'ゲートウェイ' と 'MPC1100' (MAPPER Protocol Converter1100) とに大別される。MAPPER1100 と他社機器を接続する形態としては次の二つが可能である (図 17)。

MAPPER と他社機器接続を可能にするソフトウェアの働きは、次のようになっている。

- ① MPC1100 と MPC1100 (他社接続)：CRT 画面や印字装置等を制御するための端末固有のデータ・ストリーム (ハードウェアへの命令群) と、MAPPER1100 が前提としている UTS 仕様のデータ・ストリームとを相互に変換するソフトウェアである。
- ② ゲートウェイ：回線、ネットワークなどを制御するソフトウェアで、DCP/

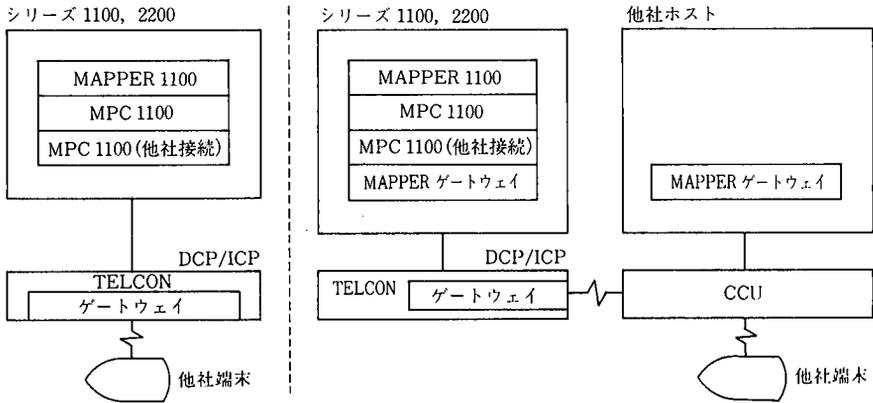


図 17 MAPPER 1100 の他社機器接続形態

Fig. 17 Connection pattern with foreign computer systems

ICP 上で動くもの、シリーズ 1100 と 2200 上で動くもの、他社システム上で動くものの 3 種類がある。

3) 管理者支援ソフトウェア群……MAPPER1100 の管理者には種々の役割があるが、その中で重要な役割の一つにファイルとレポートの管理がある。MAPPER1100 では、エンド・ユーザが簡単にレポートを作ることができる。このため不要なレポートが大量にファイルに存在し、無駄な領域を占めてしまうことが起こり得る。このようなファイルやレポートの管理を支援するために、次のランが用意されている。

- ① PRIMER：レポートの更新記録（ページ・インデックス）を参照し、指定された日付から現在までに一度も更新されていないレポートの一覧を MAPPER のレポートとして作成する。
- ② KILLER：PRIMER によって作られた、一定期間更新されなかったレポートの一覧を参照し、不要レポートを削除する。
- ③ RPG1/RPG2：ページ・インデックスを分析し、その結果（ファイルにあるタイプの総数、ファイルにあるレポートの総数と総行数、更新されたレポート数と行数等）を出力する。管理者は、その情報によってファイルを分割すべきかどうかを知ったり、更新ファイル MAPER 0 の大きさを見積ったりすることができる。

4) アプリケーション開発支援ソフトウェア群……アプリケーションの開発を支援するために、次のソフトウェアが用意されている。

- ① MAPPERKIT：MAPPER1100 システム上に構築するアプリケーションの開発支援を行うソフトウェアである。開発技法 SUM (Sperry Universal Model) のシステム化手順にそって設計・開発を支援するソフトウェア・ツール Kit Build と、使用・保守を支援するソフトウェア・ツール Kit Aid から成り立っている。

Kit Build はデータ・ディレクトリの機能を持っていて、それを利用し入出力画面や MAPPER ランのスケルトンを自動的に作り出す。また、開発技法 SUM

の手順に従って仕事を定義すると、それがそのままドキュメントになる。

Kit Aid は、データ・ディレクトリを利用し、項目または属性が変更されると、それに伴う入出力画面と MAPPER ランを自動的に変更する。

詳細については、本誌の MAPPER パッケージ/ユーティリティ紹介の 1.1 MAPPER KIT の項を参照されたい。

- ② RUN ラン：RUN ランは、エンド・ユーザが会話機能で行った処理をそのまま MAPPER ランに置き換えるものである。ラン命令をまったく知らないエンド・ユーザも RUN ランを使用することにより、日常行う作業をランにすることができる。
- 5) グラフ関連ソフトウェア群……グラフ関連のソフトウェアとしては、次のものが用意されている。
 - ① グラフ・ラン：MAPPER1100 では簡単な操作で描画できるグラフは、円グラフ、棒グラフ、積み上げグラフ、立体グラフ、折れ線グラフ、混合グラフ（棒と折れ線）、散布図、タイム・チャート、ターゲット・チャートおよびレーダ・チャートなどである。
 - ② グラフ合成パッケージ：グラフ合成パッケージは、一つのグラフ画面を最大 4 個に分割し、そのおのおのに指定されたグラフまたはグラフ化したレポートを合成して描画するランである。画面分割のパターンは 7 種類ある。
 - ③ UGR ラン：端末側ソフトウェア(U-G/CHART) を起動するとともに、指示された MAPPER のレポートを U-G/CHART に渡すランである。この後、使用者は、端末に受信した MAPPER レポートを U-G/CHART により描画する。
- 6) ユーティリティ・ソフトウェア群……汎用性のあるラン、またはプログラムをユーティリティ・ソフトウェアとして用意している。これらのソフトウェアの主なものは次の通りである。
 - ① EX_{xxx}：会話機能でファンクションを実行すると、80 桁のマスクしか表示されない。これに対して EX_{xxx} を使用すると 132 桁のマスクが 2 行に分かれて表示され、そのあとは通常の会話機能と同じ操作で処理を行うことができる。EX_{xxx} で使用できる機能は、カルキュレート機能 (EXCAL)、マッチ機能 (EXMA)、リフォーマット機能 (EXRF)、サーチ機能 (EXS)、ソート機能 (EXSORT)、トータライズ機能 (EXTOT)、グラフ・ラン機能 (EXBAR-棒グラフ、EXLIN-折れ線グラフ、EXPIE-円グラフ、EXSCAT-散布図) である。
 - ② ADDDEL：モード、タイプ、RID の範囲を指定して、複数のレポートの追加・削除をするランである。
 - ③ CRES：表示されているレポートまたは指定したレポートの全部、または一部をリザルトにするランである。
 - ④ MPA：移動型ディスク・パック上にファイルを分割して、カタログするプログラムである。更新用ファイル MAPER 0 を分割すると、I/O の集中が避けられ効率の向上が図れる。
 - ⑤ KC DSP：罫線/カラー画面を作るためのランで、端末画面より、罫線を引く

場所、文字、入力フィールドの属性、カラーを指定することにより、@OUT 命令で画面表示できるリザルトを作る。さらにカラー、入力フィールド属性などを変更することができる。

- ⑥ KCOP：端末画面から横罫線、縦罫線の情報を受け取り、これをもとに罫線文字のみのレポートを作る。また、このようにして作られた罫線文字のレポートと、データ・レポートを合成し印書するランである。
- ⑦ MAPPER-EXT：COBOL プログラムから CALL 命令で呼ばれるサブ・ルーチン群で、MAPPER1100 を使用することなく、MAPPER1100 レポートを作ることができる。

5. MAPPER1100 利用状況と利用形態

図 18 はユーザの業種別導入状況と、1 社当たりの平均使用端末数を 4 年前と比較したものである。業種別の比率は 4 年前（昭和 58 年）の調査の時と同様ほとんど変わらず、すべての業種で幅広く使われていることがわかる（ユーザ数は 3.2 倍）。

使用ユーザ 400 社の適用業務の代表的なものを列挙してみると、次のようになる。

- 1) 人事情報
- 2) 営業店管理
- 3) 経営管理（融資、資金、経理等）
- 4) 販売管理
- 5) 在庫管理
- 6) 本部情報システム（OA, DSS）
- 7) 生産管理
- 8) DP 部門での開発管理と各種ログ管理
- 9) 業績管理
- 10) 統計業務

傾向をみると、他のトランザクション処理専門のシステム等と結合し、基幹業務で使うユーザが増えてきている。各ユーザの分類の方法に違いがあるので多少の誤差はあるが、利用の多い業務は 4 年前とそれほど変わらず、むしろその業務の中での処理の拡張、掘り下げが相当なされてきている。

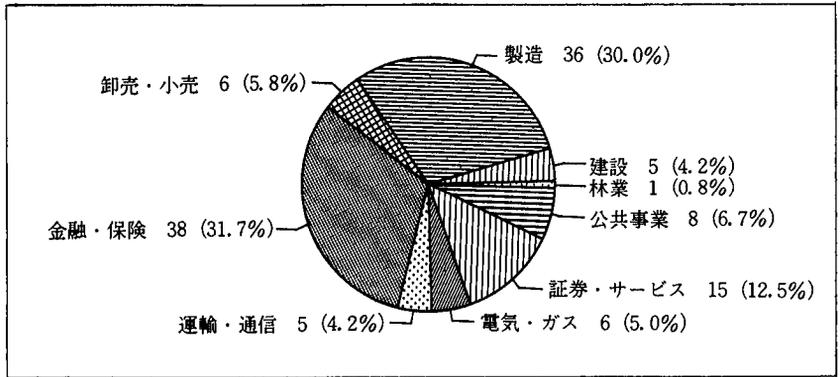
その理由は集計的な処理から、先へ先へと手を打ってゆく戦略的処理への移行が必要となってきたためで、MAPPER がユーザの要求に応じてきたことを反映している。

その一例をあげると、『あるメーカーの営業部で、各営業店の売上を把握し、売上を伸ばすために 58 年 3 月に店別売上実績表を作成した。しかし現状は把握できたが、売上不振の店の原因はなかなか掴みきれなかった。

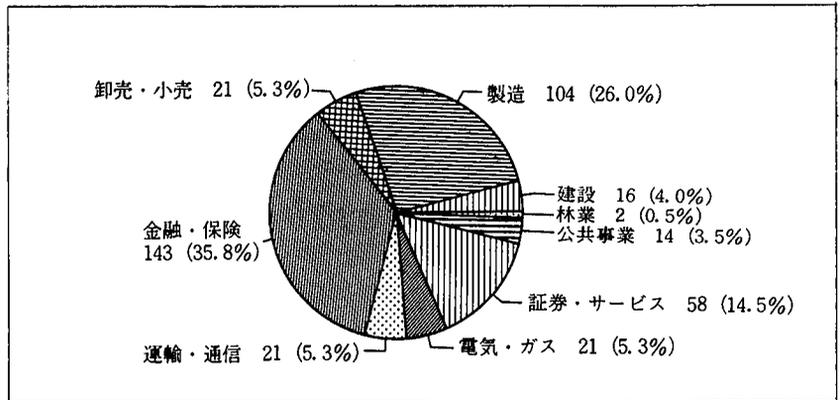
その後、月別商品別の売上実績と、従来ペーパー情報であった競合他社の地域別月別商品別の売上情報を、MAPPER データベースに入力できるシステムを構築し、それらの相関関係を見つけられるようにした。

その結果、地域性と時期と商品との関係がわかり、いつ・どの店に・どの商品を・どのくらい供給すればよいかを把握できるようになった』という。

これはまさしく、集計的処理から戦略的処理への展開の一例といえる。



(a) 昭和58年調査 120 ユーザ 平均使用端末数 7.6 台



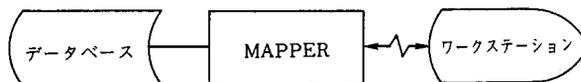
(b) 昭和62年調査 400 ユーザ 平均使用端末数 15.2 台

図 18 業種別 MAPPER 導入状況 (昭和58年, 昭和62年)

Fig. 18 MAPPER installations by market segments (1983, 1987)

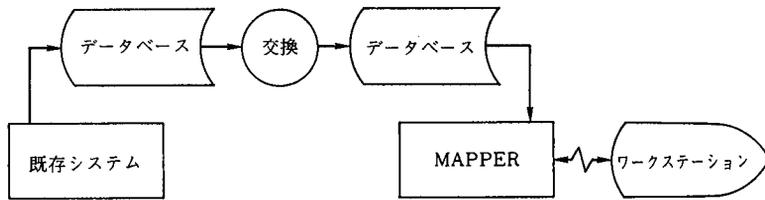
MAPPER の応用例には、大規模なもの、コンパクトなもの、ビジュアル感覚に訴えるもの、簡素なもの等、多種多様であるが、ここでは視点を変えて、それらをその利用形態から分類してみる。

1) MAPPER 単独個別利用型



MAPPER のみによる処理利用形態であり、これまでコンピュータ化がまったく行われていない分野での利用、個別事務処理の効率化、OA 化に利用される。また機密性の高いもの (人事システム)、パーソナル・ユースのものにも向いている。米国では、この形態で大規模なオンライン・システムにも利用されている。

2) 現行システム連結利用型

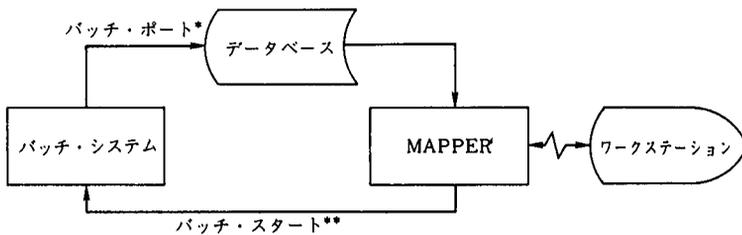


定型大量処理を行っているバッチ・システムのデータベースを MAPPER 側に引き込み、抽出・合成してさまざまな方法で加工をしたり、場合によってはデータを MAPPER 端末から追加して目的の帳票を作成する使用形態である。

営業情報システム等、日計表をベースにして作るシステムは、この形態をとる。日本語化されていないオリジナル・データを日本語化したり、グラフ化することにより見やすいデータを効率よく作成することができる。

また、還元帳票を別の切り口からみて、意思決定する必要がある業務ではよく使われる形態である。

3) MAPPER-バッチ結合型



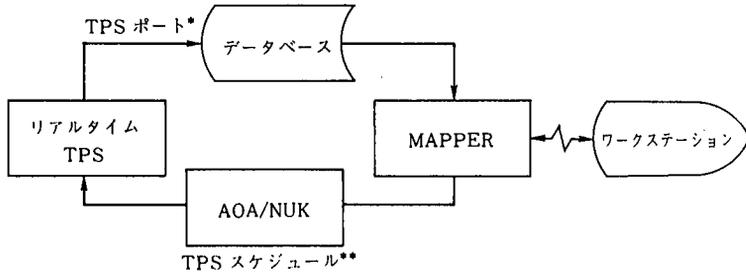
*バッチ・ポート：バッチ処理の結果を特定の出力持ち行列ファイルにキューイングすることにより、自動的に MAPPER 側のランを動かしデータを渡す。

**バッチ・スタート：レポート上にあるバッチ・ジョブ制御文 (JCL) を MAPPER 側からバッチ・ジョブとして起動させる。

定型大量処理等、本来バッチ・システム向きの処理や、データベースがバッチ側にありバッチで処理した方が便利な場合に、この使用形態がとられる。メインの処理はバッチ処理で行われ、そのバッチ処理の結果を MAPPER で利用するというもの。

この場合の起動は、バッチ側でも MAPPER 側でもよい。またバッチ処理と MAPPER 処理の比率は、対象業務の性格によって違ってくる。重要なことは各々の処理形態での持ち味を生かし、組み合わせとして最大効果を発揮させることである。

4) MAPPER-リアルタイム結合



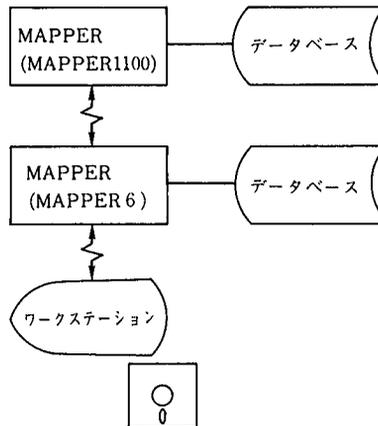
*TPS ポート：リアルタイムのトランザクション処理を行うプログラムである TPS (Transaction Processing System) の処理結果を MAPPER 側のランに渡す。

**TPS スケジュール：TPS にデータを渡し実行させる。

大量かつ高速のトランザクション処理が必要な業務、あるいはデータベースが既存リアルタイム処理側にあり、リアルタイム処理で処理した方が便利な場合にこの使用形態がとられる。

メインの処理はリアルタイム処理で行われ、そのリアルタイム処理の結果を MAPPER で利用するというもの。金融関係の業務に比較的多い形態である。

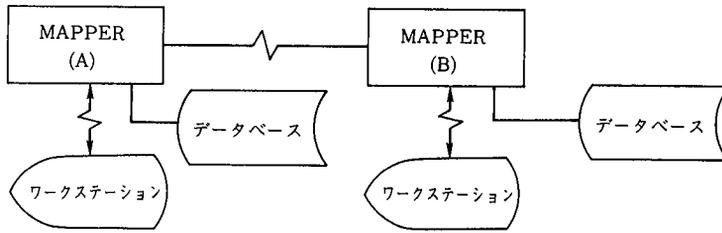
5) MAPPER TO MAPPER (垂直型)



大量かつ高速性を要求される処理は、ホスト側 MAPPER システムで行い、ローカル側オリエントドな処理や、機密性の高い処理はローカル側で行おうという使用形態である。

この形態をとれば、処理形態と業務内容によって、ホスト・システムとローカル・システムの運用形態、時間帯等の違いを調整し、適正な分散が計れる。

6) MAPPER TO MAPPER (水平型)



A のシステムに接続されている端末から、B のシステムの処理を行いたい場合この形態がとられる。

この応用例で、A, B, C の各システムをトライアングルで結べば、A システムに接続されている端末から、B システムと C システム間に対するコマンド指示ができる (たとえば、B システムのファイルを C システムへ転送させる指示等)。

以上で六つの利用形態を説明してきたが、どの利用形態を採用するにせよ最終的に使用するのはエンド・ユーザなので、その時々目的や状況の変化に柔軟に対応できるようなシステムになっていることが望まれる。

そのためには、MAPPER の原点であるところのマニュアル・ファンクションをうまく活用することが良いといってもいいのかもしれない。MAPPER はマニュアル・ファンクションで始まり、マニュアル・ファンクションで終わるといっても過言ではない。

6. 評価

システムを評価する際の基準はさまざまであり、一つの基準で判断することは時として誤った評価につながる可能性がある。そのため、ここでは MAPPER の評価を次の三つの面から行ってみることにする。

- 1) 第 4 世代言語としての機能から見た評価
- 2) ユーザの有効活用と利用から見た評価
- 3) MAPPER のもつプロダクトとしての特徴から見た評価

6.1 第 4 世代言語としての機能から見た評価

第 4 世代言語の定義は、それを語る人によって異なり明確にされてはいないが、いくつかの特性の共通点をまとめると以下ようになる。

- 1) プログラミングの経験のない人でも簡単に使い方が習得でき、すぐに利用可能である。
- 2) 第 3 世代言語と比較して 10 分の 1 程度の開発工数で目的の仕事ができる。
- 3) プロトタイプ手法でパイロット・システムが構築できる。
- 4) ビジネス・グラフィック・インタフェースを持っている。
- 5) クウェリー・レポート・ライタ機能がある。
- 6) データベース管理機能を持っている。
- 7) オンライン処理用に設計されている。
- 8) 対話的な操作を基本としている。
- 9) できる限り非手続き型である。

10) 処理の方法を指示するのではなく、何をして欲しいかだけ指示する (HOW → WHAT).

11) デバックが簡単である.

第4世代言語として紹介されているものに表1がある。これらの言語は、上記の要件をすべて完全には満足しているものではないが、何等かの手段によってそれらの要件をカバーしているものと言えよう。

表1 おもなホスト第4世代言語
(コンピュータワールド1986.10.15, 日経コンピュータ別冊1986.9.10号より)
Table 1 Major 4GLs

商品名	企業	販売実績		動作環境
		国内	海外	
MAPPER	ユニバック	400 (260)	2500 (600)	1100シリーズ OS 1100 (80シリーズ OS 3, MAPPER 6)
MANTIS	シンコム・システム	75	2000	IBM, 30 xx, 43 xx, FACOM, HITACH, DEC, MVS, VOS 3, VMS, DOS
LINC II	パロース	320	1600	Aシリーズ, Bシリーズ MCP, MCP/AS
FOCUS	インフォメーション ビルダーズ	90	1800	IBM MVS/TSO, VM/CMS, VOS 3/TSS
NATURAL	ソフトウェア エージェンシー	129	2000	IBM, FACOM, HITACH MVS, VM, DOS, OSIV, VOS 1, VOS 3
AS	IBM	—	—	IBM MVS, VM (専用OS)
EXCEED	日立	—	—	HITACH VOS 3
INTERACT -II	富士通	—	—	FACOM OSIV

さて、以上の各項目について MAPPER の評価をしてみると、表2の結果となる。

以上から MAPPER を第4世代言語として見た場合の評価は、その定義そのものに多少の異論はあるにせよ、総合的にみて高いものであると判断できよう。

6.2 ユーザの有効活用と利用から見た評価

ここでは、少なくとも提供者側の自己満足にならないためにも、実際に利用しているユーザ側からの評価をまとめてみる。

ユーザ研究会の論文や事例紹介、あるいは社内や社外での講演や報告での MAPPER 評価をまとめると以下のようなになる。

- 1) 生産性向上が図れる。
 - ・プログラミング・レベルの生産性が向上できる (MAPPER の1ステートメントは COBOL の5~10ステートメントに相当。表3参照)。
 - ・システム設計から開発までの期間が短縮される。
- 2) 要求したものと完成品との差がなくなった。
 - ・パイロット・システムを容易に構築できる。

表 2 MAPPER の評価
Table 2 Evaluation of MAPPER

機能項目	MAPPER
① プログラミングの経験のない人でも簡単に使い方が習得でき、すぐに利用可能である。	MAPPER のマニュアル・ファンクションは 1~2 日で習得でき、自由に使いこなせる。
② 第 3 世代言語と比較して 10 分の 1 程度の開発工数で目的の仕事ができる。	MAPPER のラン・ファンクションの生産性は COBOL の約 5~10 倍であるが、マニュアル・ファンクションとの組み合わせ等によりさらに大きくなる。
③ プロトタイプ手法でパイロット・システムが構築できる。	要求者と作成者はマニュアル・ファンクションで、画面設計等に関してお互いの意思の疎通をはかる。その後、テスト・データを作りラン・ファンクションによるパイロット・システムを作成する。これが MAPPER によるシステム開発の基本となる。
④ ビジネス・グラフィック・インタフェースを持っている。	MAPPER 本体に組み込まれているものとユーザ・レベルのものがあり、新しい種類のグラフもユーザ・レベルで作成可能である。
⑤ クウェリー・レポート・ライク機能がある。	レポート形式のデータベースおよび表示が基本で、検索機能が豊富である。
⑥ データベース管理機能を持っている。	リレーショナル型データベース管理機能を持つ（ファイル・キャビネット思想）。
⑦ オンライン処理用に設計されている。	オンライン・システム・パッケージとしても実績が多い（実行優先度もリアルタイム・モードである）。
⑧ 対話的操作を基本としている。	5 章で述べたように、MAPPER の原点はマニュアル・ファンクション（会話型）のレポート操作（非手続き型）である。
⑨ できる限り非手続き型である。	
⑩ 処理の方法を指示するのではなく、何をしたいかだけ指示する。	
⑪ デバッグが簡単である。	MAPPER ランはインタプリタで動くため、即座に文法チェックだけでなく、実行レベルのデバッグまで行える（手法も完備）。

- 3) 現場に即応した資料が作れる。
 - ・異なる視点からデータ加工ができる。
 - ・少量データでもコンピュータ化の対象にできる。
 - ・帳票だけでなくグラフ化も簡単である。
- 4) バックログが解消できる。
 - ・エンド・ユーザにも使用できる。
 - ・EDP 部門の経験の浅い要員の戦力化が図れる。
- 5) 利用部門のモラル向上が図れる。
- 6) システムの日本語化が容易である。
- 7) データの一元管理ができるので整合性がとれる。
- 8) 教育ツールがあるので好きな時に学習できる。
- 9) 機能アップとシステム・サービスの対応が早い。
- 10) 営業戦略ツールとして価値がある。
 - ・いろいろな視点で、誰でも、好きな時に、ホットな情報を得られる。
- 11) レス・ペーパー化が図れる。
- 12) データ保全機能が完備しているので、安心してオペレーションができる。

表 3 MAPPER と COBOL の生産性比較 (営業情報システムの例)
Table 3 Comparison of MAPPER productivity with COBOL productivity

ソフトウェア名 処理内容	MAPPER	COBOL	差
サーチ	1×2= 2	20×2= 40	-38
分類	1×1= 1	5×1= 5	-4
演算	1×1= 1	25×1= 25	-24
突き合わせ	1×3= 3	25×3= 75	-72
ステップ数 画面設計☆	2 or 3×7=19	100×7=700	-681
グラフ☆	1×1= 1	250×1=250	-248
ファイル定義	0	50×1= 50	-50
その他	45	45	0
合計	72	1,190	-1,118
開発工数 (人/日)	0.9	8	7.1
金額換算 (円)	44,100	392,000	347,900

・COBOL の場合、ほかにエディタ・コンパイル、ジョブ制御が必要であり、さらに画面設計サポート・プログラム、およびグラフサポート・プログラムを用意しておく必要がある。

・☆印に関しては、COBOL のインタフェースによって多少の差がある。

・プログラム能力は、1人/日で MAPPER の場合 80 ステップ、COBOL の場合 150 ステップとして計算している。

・人件費は、1人/日で 49,000 円として計算している。

13) 導入や運用が簡単である。

14) 管理者機能が有効である。

逆に問題点としては以下がある。

1) 使い方によっては CPU の負荷がかなり高くなる。

2) 大量データの場合、データベースの設計が悪いと、レスポンス・タイムに影響する。

3) 放っておくと、データベースがどんどん大きくなってしまう。

4) バッチ処理でやるべきものも、MAPPER 化してしまう。

これらの評価はユーザのコンピュータの規模・利用形態・環境によって多少違ってくるかもしれないが、最終的には MAPPER が提供する効果と、効率等の問題点とを相殺して個々のユーザが自分達が持っていた問題をいかに解決できたか、またどれだけの利益を生むことができたかが、その基準になろう。

さて、本節の最後に MAPPER 導入の効果を示すデータを見てみよう。

これは、エンド・ユーザにも積極的に開発に当たらせ、比較的部門単位に使用している T 社の例であり、端末台数 10 台、システム数約 50、端末操作人数約 30 人のシステムである。本例の開発期間と導入前後の処理時間の差から、MAPPER 導入の効果を読み取っていただけると思う(表 4)。なお、この種の業務は、多種少量データを扱うため、伝統的な DP 処理では効果が上げにくい分野であり、MAPPER の有用性が顕著に現れる例となっている。

表 4 MAPPER 導入効果のユーザ事例
Table 4 Case data of a MAPPER installation

利用部門	利用業務	開発期間 (月)	効果(分/月)	
			導入前	導入後
営業部	出荷実績	0.5	500	40
	重点商品達成率表	0.5	300	20
	業界地域別出荷統計	0.3	300	20
	業種用途別売上実績	0.3	400	10
	年度末上位売上分析	0.5	1390	55
	計	2.1	2890	145
経理部	営業所別費用売上対比	0.3	60	10
	営業所別販売費実績	0.3	180	10
	加工費予算関係	1.0	1080	150
	役員会資料	0.2	120	10
	販売予算	2.0	1800*	120*
	課出張所別損益検討表	2.0	1800	60
	事業所別人員表	0.1	120	5
	計	5.9	3510	255
業務部	棚卸差異一覧	0.3	120	5
	製品振替表	0.2	180	10
	営業所別棚卸集計	0.5	120	3
	仕入商品納入台数実績	0.2	210	5
	工場別販売予測	0.2	445*	20*
	配送計画	0.2	665	240
	各種赤残リスト	0.5	1200	46
	その他リスト	0.5	210	6
	計	2.6	2742	317
人事部	一時金考課	6.0	374	29
	賞与計算			
	昇給考課			
	年末調整			
	賞金分析 その他人事・給与システム			
計	6.0	374	29	

*の付されているデータの単位は分/年である。

6.3 プロダクトが持つ特徴から見た評価

MAPPER1100 の第 1 の目的は、エンド・ユーザ・コンピューティング環境を提供することで、基本思想はエンド・ユーザ主導型のシステムである。しかし、他方 DP 部門においても、データ保全、運用、セキュリティの各面で評価されており、機能の分散と集中をシステム内部で調整できる総合的なプロダクトとなっている。

まず、エンド・ユーザ主導型のシステムとしての特徴は次の通りである。

- 1) マスク画面に対するパラメータ入力方式（スプレッド・シートに類似）
- 2) インタプリタ方式
- 3) マニュアル・ファンクションからラン・ファンクションへの自動生成機能（手順の自動化）
- 4) 豊富なセキュリティ機能
- 5) 使用者レベルのリカバリ機能
- 6) ヘルプ機能
- 7) 大型機からマイクロ・コンピュータまでの、すべて同じ操作による処理等が挙げられるが、とくに7)は特筆すべき特徴と言えよう。

また、各種ログの採取と管理、データ保全、運用、管理に関する特徴は次の通りである。

- 1) 各種ログの装備
 - ① リソース管理ログ……リソースをどれだけ使用しているかを、時系列形式とモニタ形式で見ることができる。
 - ② 課金ログ……使用者ログを編集することにより、課金処理ができる。
 - ③ 使用者ログ……使用者が何時・どの端末から・何の処理をしたかがわかる。
 - ④ コミュニケーション・エラー・ログ……回線上でエラーが発生した場合、その原因と回数がわかる。
- 2) データ保全方法
 - ① パージ……指定した時間に自動的にデータベースを保存する。テープ・パージと無人化のためのディスク・パージがある。
 - ② サイクル/マージ……MAPPER の使用中に、データベースを保存する。
 - ③ ダイナミック・パック……万一に備え、リカバリ用に更新データをテープに引き上げる（MAPPER 稼働中）。
 - ④ 外部ファイル……読み込みのみ可能なファイル等をパージの対象外として登録できる。
- 3) 管理者モードの機能
 - ① 使用者に対する制限……使用コマンド等
 - ② ラン・ファンクションの実行に対する制限……実行時間帯、使用者、データベース、実行ステップ、入出力回数等
 - ③ その他……使用者の管理、セキュリティに関すること
- 4) 導入の容易さ
 - ① システム生成……基本的には不要
 - ② 実行時パラメータ……きめ細かいチューニングアップとカスタマイズ等が挙げられる。

最後に MAPPER の管理機能とパフォーマンスの問題について、やや詳しく論じ本章のまとめとする。

- 1) 管理機能……冒頭に記述したように使用形態から見た場合、エンド・ユーザ主導型であるが、システムの管理という面から見た場合はホスト主導型（集中型）

であるといえる。

MAPPER では、その管理を行う人間を MAPPER 管理者（コーディネータ）と呼んでいるが、この管理者に対するさまざまな機能、および権限（たとえば、使用者および使用端末の登録、使用リソースの制限および監視、使用者に対する機能の制限と監視等）の提供が、使用者と管理者の間に意思の疎通をもたらし、両者のモラルを向上させたといえよう。この管理者に対するさまざまな機能は、他の簡易言語には見ることができない MAPPER の大きな特長である。

システムをその利用面から見た場合、集中化が良いか分散化が良いかに関する議論は、種々の要因により結論は左右される。しかしながら、管理という面では集中化がよいことは異論のないところであろう。

MAPPER は利用面では集中化と分散化の両方を同一アーキテクチャのもとに実現させているのと同時に、管理の面では集中化を推し進めていることは特筆すべき点である。さらに管理という考え方に留まらず、それを監査という意識を持った機能にまで拡張していることは、現在の情報処理分野の要求にマッチしているといえる。

MAPPER は“より高度な技術をより使いやすく”を目指しエンド・ユーザ向きの機能拡張を長年行ってきているが、同時に、7章の図 19(b)に見られるように、地味で目立たない（しかしながら非常に重要である）管理者向けの機能も、一步一步確実に機能拡張を計ってきている。

- 2) パフォーマンスの評価……MAPPER を使ったシステムはオンライン専用のシステムと比べれば、パフォーマンスは悪くなる要素を持っている。一つはインタプリタ形式であることであり、もう一つはデータベースがリレーショナル型であることである。

しかし、この条件を持つソフトウェアの中ではその水準はどうであろうか。

これは、各メーカ（含ソフトウェア会社）の提供している 4 GL を使用しているユーザの講演会でのことであるが、MAPPER ユーザ事例の処理内容・規模・応答時間を聞いて、他の発表者のほとんどが、MAPPER がこの条件のもとで稼働しているソフトウェアとは思えないという所感を述べたことから察せられる。

これだけですべてを判断するわけではないが、世の中一般に出回っているソフトウェアのユーザ事例を考察してみても上の二つの条件を持つソフトウェアの中では、そのパフォーマンスの良さは、世の中の水準を上回っているものと思われる。

なお、MAPPER は実行時の効率を高めるために、次の三つの解決方法をユーザに提供している。

第一は、以下に挙げる MAPPER 本体の改良で対処したことである。

- 1) ランのラベル定義機能
- 2) 入出力バッファ拡張機能
- 3) 実行プライオリティ設定機能
- 4) MAPER 0 分割考慮機能
- 5) パス・レングスの縮小化

第二は開発支援総合パッケージである MAPPERKIT の提供である。これによりユ

ーザは、効率の良いランとデータベースを構築することが可能となった。

そして、第三は James Martin が彼の著書「データベース環境の実現と管理」で述べているエスケープ機能の提供である。彼は 4 GL は、従来のプログラミング言語で作成した論理モジュールと関連づける機能を持つ必要があると述べており、それをエスケープ機能と呼んでいる。ここではこれをもう少し広義に考えて、有機的結合機能として位置付ける。

たとえば 5 章で挙げたが、トランザクション処理形態の効率を上げるための TPS インタフェース機能、バッチ処理形態の効率を上げるためのバッチ・インタフェース機能がその典型例である。ここで、MAPPER の有機的結合機能をまとめたものを表 5 に示す。なお、効率を考える場合、実行効率のほか、生産（開発）効率も考慮する必要がある。

本稿の第 1 章で述べたようにソフトウェアの短命化が進み、今やその使い捨ての時代になってきている。要求されたものを、いかに迅速にアウトプットできるかが、今の情報システムに要求されている課題である。

それを考えた場合、開発の生産効率は個々の実行効率よりも、はるかに影響が大きい。今後の情報システムにおいては、激しい変化に対応していくために開発の生産性を上げることが最重要事項であろう。

表 5 有機的結合機能一覧（内部処理も含む）

Table 5 Advanced linking functions
with external systems

<ul style="list-style-type: none"> • TPS インタフェース機能 • バッチ・インタフェース機能 • FAX インタフェース機能 • SUFICS 1100 インタフェース機能* • QLP 1100 インタフェース機能** • TELIDON インタフェース機能 • 光ファイル・インタフェース機能 • 関数インタフェース機能
<ul style="list-style-type: none"> • SUFICS 1100 (Sperry Univac Financial Integrated Control System) : 意思決定支援ツールである SUFICS は、統合 OA (AOA) の中で MAPPER とインタフェースを持つ。 • QLP 1100 (Query Language Processor) : DMS に対する Query Language.

以上から総合的にみると、MAPPER は有効性と効率性と信頼性をバランスよく兼ね備えた一つのプロダクトとして評価されよう。そして、その利用はユーザの情報システムに対して大きな影響を与えてきたと思われる。

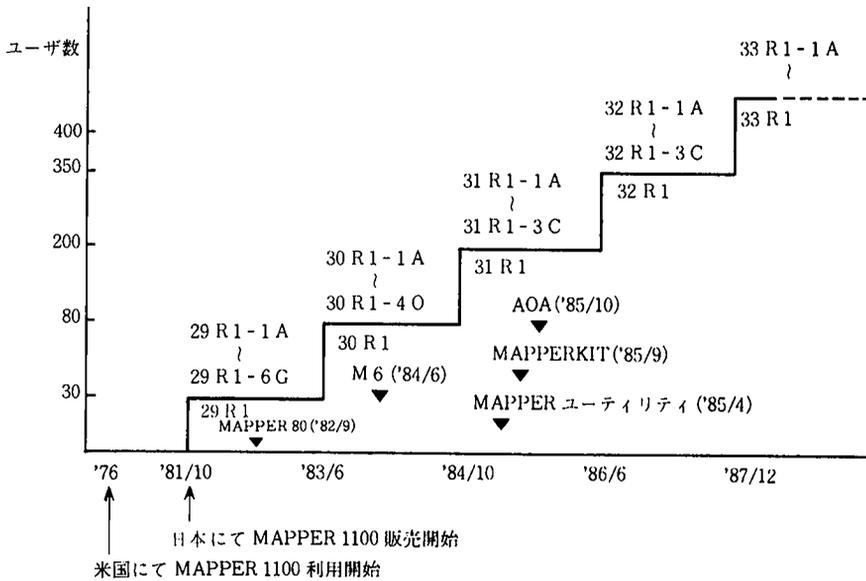
なかでも、エンド・ユーザ自身に自分達の要求をコンピュータ処理で解決しようという意識を持たせたことと、実際にそれを実現させてきたことが特徴的である。

その結果、MAPPER1100 は情報処理システムにおける戦略的処理を支えるツールとしての地位を確立し、経営とのかかわり合いをさらに深める役割を果たしつつあるといえよう。

7. お わ り に

ここでは、MAPPER の将来像を紹介する。その前に、MAPPER の機能改良の歴史を振り返ってみよう。

現在 MAPPER のバージョンは 32R 1-2B であるが、過去の各バージョンでの機能とユーザ数の推移を図 19 に示す。ユーザ数の増大とともに、開発支援ツールやパッケージ、さらには他システムとのインタフェース等が追加されてきている様子がうかがえる。



(a) 各バージョンとユーザ数

バージョン	リリース年月(国内)	主 な 新 機 能
29 R 1-1 A	'81/10	漢字対応
29 R 1-2 B	'82/ 1	0780 日本語プリンタ対応, サブルーチン機能, BF 機能
29 R 1-3 C	'82/ 4	グラフ機能
29 R 1-3 D	'82/ 7	MAPPER CMS インタフ.フェース機能
29 R 1-4 E	'82/ 8	新トータライズ機能, レポート・符号化パスワード, MSR 機能
29 R 1-5 G	'82/10	UTS 4000 クラスタサポート, ダイナミック・バック機能
29 R 1-6 A	'83/ 1	日本語エラー, メッセージ, バッチ・レベルのラン実行
30 R 1-1 A	'83/ 6	CMS1100 インタフ.フェース, UTS 50, CAL 機能, リード・パスワード, エラー・ルーチン登録
30 R 1-2 B	'83/ 8	日本語ヘルプ, グラフ・ラン, 横項目グラフ, MPC 1100 インタフ.フェース
30 R 1-3 C	'83/12	仮想端末でのラン実行, CONS-MSG 機能, プリント戻し/送り機能
30 R 1-4 D	'84/ 5	罫線・拡大文字印書, MAPPER-TO-MAPPER, リカバリ・ディスク, マイクロ・レボ・インタフ.フェース
31 R 1-1 A	'84/10	システム生成の不要化, 会話/ラン機能拡張, ランの分類登録
31 R 1-2 B	'85/ 4	日本語プリンタ・オーバーレイ, ディスク・バージ
31 R 1-3 C	'85/10	DS 7/DS 3, イメージ処理, レジデント・ファンクション, MAPPER 0 分放
32 R 1-1 A	'86/ 6	手順登録機能, 新会話/ラン機能, 画面合成, DS 7-WP インタフ.フェース
32 R 1-2 B	'86/11	リザルト戻し機能, CAL 計算式登録, サイクル・マージの時刻指定
32 R 1-3 C	'87/ 5	TOT 機能強化, レポート・パスワード機能強化, パーシャル・バージ
33 R 1-1 A	'87/12(予定)	ランの時刻指定実行, RDB インタフ.フェース, バッチとのダイレクト・インタフ.フェース

(b) 各バージョンとその追加機能

図 19 MAPPER 1100 の機能改良の歴史およびユーザ数の推移

Fig. 19 History of MAPPER 1100 releases and growth in user population

さて、MAPPERの将来であるが、まず第1にMAPPERのマイクロ・プロダクト化がさらに進むであろう。

その結果MAPPERプロダクトのライン・アップがよりきめ細かくなり、大型汎用機による全社的・戦略的な利用形態、オフィス用小型機またはミニコンによる部門的・戦術的な利用形態、オフィス・ワークステーションによる個人的でパーソナルな利用形態の各システムを同一のソフトウェアで構築し、有機的に結合することが今以上に容易となろう。

具体的には、現在、大型汎用機にはMAPPER1100、オフィス用の小型機にはMAPPER11、ミニコン用およびパーソナル・ユース用としてはMAPPER6がすでに提供されているが、近い将来UNIXマシン等のさらにマイクロ化されたコンピュータへのMAPPER搭載が予定されている。

第2に、評価のところで述べたような他プロダクトとの有機的結合機能がさらに強化されることになろう。

考えられるものとしては、

- 1) 他のリレーショナル・データベースとの結合
- 2) COBOL等のプログラム言語で書かれたモジュールとのスケジューラを通さない結合
- 3) ニュー・メディアとの結合
- 4) AIとの結合

等が挙げられる。

最後にパフォーマンス・アップについては、いずれハードウェアの高速化により解決される一面もあるが、ソフトウェアにおける追求は今後も続けられる。

このため、MAPPER内部処理の高速化の追求はもちろんであるが、ランのパフォーマンス予測機能も必要となろう。具体的にはランのレスポンス・タイムを予測し、パフォーマンス低下の原因となるロジックやデータベース処理等を指摘する機能となろうが、AIの発展に負うところが大きく課題も多い。

また、MAPPERのデータベースは本質的には関係形式なので、関係データベースの最適構成法および各構成法を評価する尺度の研究の発展も大きな鍵となろう。

いずれにせよ、エンド・ユーザ・コンピューティングが、ますます盛んになると予想される情報処理分野において、MAPPERは自らの広がりや結合とで今後も大きく成長してゆくことにまちがいはない。

そしてその結果、エンド・ユーザ自らが自分達のニーズをコンピュータ処理で解決してゆくという革新的進歩をさらに推し進め、利用ユーザに利益をもたらすために大きな役割を果たすものと確信している。

-
- 参考文献 [1] James Martin 著, 山中義昭監修, プログラマーなしのアプリケーション開発, (株)アシスト, 1984.
 [2] 栗田昭平, “第4世代言語MAPPERに期待する”, SYSTEMS, 9・10月号, No. 177, 1984.
 [3] 山岸史明, “プログラムレス・システムの動き. 利用者指向システム構築のために——プログラムレスを実現するMAPPER1100”, SYSTEMS, 7・8月号, No. 158, 1981.

- [4] H. Katzan, Jr., *Invitation to MAPPER (I) A Pragmatic Approach to End-user Computing and Report Preparation*, Petrocelli Books, 1983.
- [5] “特集：ソフトウェア・プロダクト”，週間コンピュータワールド，1986.10.1.
- [6] “最新技術が組み込まれるエンド・ユーザ向けアプリケーション開発システム”，日経コンピュータ別冊，1986.9.10.
- [7] James Martin 著，坂本広ほか訳，データベース環境の実現と管理(上巻)，日経マグローウヒル社，1987.
- [8] '87 オフィス・オートメーション実態調査報告書，日本オフィス・オートメーション協会，1987年3月.

執筆者紹介

川 本 光 一 (Kouich Kawamoto)

武蔵工業大学卒業，1970年日本ユニパック(株)入社，製造工業のユーザ・アプリケーション・システムのサービスに従事。現在，商品開発本部商品企画部所属。



論文

情報化社会のシステム開発と MAPPER の有効性

MAPPER Usability in System Development
in Informationalized Society Era

日 置 伸 一

要 約 システム開発部門は、20 余年間に急速に拡大してきたが、後半になってシステムの開発効率は停滞し、メンテナンスや開発のバックログが山積みされてしまった。

システムやプログラムは、機械が作る工業製品と違って人間が創る知的生産物であるから、情報化社会の価値観（変化・多様化・人間性等）で評価しなければならない。これまで、均質や量という工業化社会の価値観で対処してきたことに誤りの源がある。運用を始めてから、エンド・ユーザの要求仕様が次々と変化するのは、そのシステムが有効であったからこそ組織に変化を及ぼした好ましい結果であると積極的に評価すべきである。

MAPPER 1100 が情報化社会のシステム開発に有効な事例は沢山あるが、重要なのは MAPPER を導入してでも新しい価値観で対応しようとする意欲である。

ここでは、ワンパターンからの脱却、スクラップ&ビルト、データベース設計の重視、MAPPER 会話機能、プロトタイピング等の価値観と、MAPPER 1100 および MAPPER KIT 1100 による教育での人間性重視について述べる。さらに、MAPPER を活用する MAPPER の戦略的使用法について考える。

Abstract A System or program is not an industrial product which is produced by machinery but an intellectual product which is created by human craft.

Therefore its value should not be measured by the industrial society criteria but the informationalized society criteria. The former is characterized with uniformity, large quantity, and efficiency while the latter with variety, small quantity, and humanity.

The paper delineates a guideline of EDP department which is based on new value criteria in the informationalized society, namely acceptance of variety, scrap and build mechanism, and revolution in mind of middle management in EDP department.

The paper also describes MAPPER usability in system development in the informationalized society era which includes database design, interactive function, prototyping, design/development support system MAPPERKIT, and system education using MAPPER.

Finally the paper proposes the strategic use of MAPPER as a tool for EDP department and a uniform integrated user interface under TSS environment.

1. はじめに

情報化社会への進展が始まった今日では、コンピュータ・システム開発の優劣は企業活動の成否を左右するほどに重要になってきた。激増する需要のためシステム部門の規模はどんどん大きくなってきたが、生産性あるいは品質の向上に問題が現われ深刻化している。

この原因の一つは、システム部門が今だに過去の工業化社会の価値観で物事を見て

いることによると考えられる。「システムやプログラムの生産性」という言葉は、これを象徴している。

これに対処するには、まず情報化社会のまっただ中にいることを正しく認識し、知的生産がそれまでの工業生産とはまったく異なる事実を理解する必要がある。均質・量産から脱却し、多様性や人間性を重視する新しい価値観に変化していく必要がある。

MAPPER 1100 は、情報化社会へ向けてのシステム開発ツールとして「極めて鋭い、諸刃の剣」である。桁違いの生産性をあげて喜んだり、期待に反したオーバーヘッドに悩んだり、結果は両極端である。「生命を救うメス」か、「人を殺す凶器」かは、まさに使い方次第である。

システム開発環境の枠の中で、新しい価値観について、また MAPPER 1100 環境について考えてみた。

2. システム開発環境の変化 = 病原体の誕生 =

コンピュータ・システムは、企業を始めさまざまな組織（以後、代表して単に企業と表現する）の活動に有益であると認識され、需要は一方的に増加した。加えてコンピュータ・システムが企業戦略の対象となると、一気に爆発的な量のシステム開発が要請されることになる。

1970年代から1980年代には、

- 1) メインテナンス
- 2) バグログ
- 3) 生産性向上
- 4) 標準化
- 5) ドキュメンテーション
- 6) OA
- 7) 人材教育
- 8) 通信
- 9) 分散処理
- 10) 統合化ソフト

等々の話題が、次から次へと出現し、それぞれ特効的な処方箋を見い出せないまま今日を迎えている。これにより多くのシステム開発部門にはストレスが蓄積したり、体調の変化をきたしたり、場合によっては自覚症状が現われ始めている。

これには、さまざまな原因（これをシステムの病原体と呼ぼう）が、かなり複雑に絡んでいると考えられるので、そのうちのいくつかを取り上げてみたい。

2.1 病原体の目覚め 一効果的なシステムほど、多くの発ガン物質を含む一

システム開発部門の責任者を悩ませる現象は、最初のシステムが開発されて運用を始めて暫くして起こる。

企業でシステムを開発するという最初意思決定がなされる場合は対象アプリケーション分野で、合理化／スピード化／戦略……その他の目的のために強い意欲が働いている。その分野に投入されたシステムが企業にとって真に有効であるほど、その企業に大きな“変化（効果）”を与えることになる。ところが、その変化はただちにその

システムに対する要求仕様の変化となって現われる(メンテナンス要求)。コンピュータ・システムによって多くの利益を受けた企業は、さらに新規のシステム開発要求を出す結果となる(バックログの出現)。

このことは、優れたシステムほど大きな変化を企業に与え、ガン細胞(病原体)は全身に転移し始める宿命にある。

2.2 延命剤が命取り —システムの寿命を長くしようとする努力が、短命化を招く—

創設期には最も革命的であったシステム部門も2.1節の発ガン物質が目覚めると、数年後には最も保守的な体質に変身してしまう。

この結果、システム部門としてはシステムの寿命を伸ばすことができれば、最も幸せと考えるようになってしまう。

延命剤になりそうな事柄、

- 1) しっかりした基本設計
- 2) あいまいさのない詳細設計
- 3) 標準化されたシステム開発工程
- 4) ドキュメンテーションの整備

等に努力を払う。これらのすべてを十分にこなしたシステムは、およそ優れたシステムとなるので、優れているほど2.1節でメンテナンスやバックログの対象となり、結果的には短命に終る。逆に、十分こなしてないシステムは少ないメンテナンスながらも長期間生き伸びてしまうことがあるので、その間病原菌をばらまく結果になりかねない。

ここで前者の短命は決して悲しむべきことではなく、二世誕生と同じように本来喜ばしいことである。一方、後者の病原菌はかなり悪性である可能性があるので、よく診断し、場合によっては手術をして摘出するのがよい。

2.3 消化剤で中毒に —合理化/省力化/効率化…の夢は、無限大—

コンピュータ・システムは、手作業ではとても大変で不可能な作業も何なく消化してしまうのが常である。エンド・ユーザにしてみると、コンピュータ・システムとは、Aというアプリケーションではaという機能を提供し、Bアプリケーションではbという機能を提供する道具として見ることになる。同様にして、c, d, e, …という機能(消化剤)を入手することになる。

このような体験をしたエンド・ユーザは、当然Aアプリケーション用にもb消化剤やc消化剤がほしくなる(バックログの増大/爆発)。他のアプリケーションでも同様である。

この症状が強くなると、投資費用対効果という評価基準を無視して、麻酔を打つ快感を優先してしまう(中毒症状)。

2.4 栄養過多も不健康 —詰め込み式の人材教育では、消化不良を起こすだけ—

システム部門の人材開発(要員育成)の環境は急速に整備されようとしているが、人材不足はますます深刻になってきている。

もちろん、量的に不足している面も大きいですが、質的にはさらに難病化している。

システム部門の創設期の担当者は、あらゆる問題に立ち向かって試行錯誤の繰り返しの中で、解決の道を切り開いてきた。

それに比べて、現在の新任システム開発要員は、事前により多くの教育課程を経てきているのに、成長の度合は相対的にかなり低いと評価されるケースが多い。

現在のシステム部門のキーマンは、自分がコンピュータについて学んだ時は企業内の教育課程の中で、プログラミング技術はもちろんのこと、システム開発技術については何も知識として与えられなかったと思い込んでいる（しかし無限大のチャンスが与えられていた）。新任要員には、それらの技術研修も相当量与えられているのだから、当然に自分と同じくらいには身長・体重共に成長してよいと思ってしまう。

同じ距離の道のりでも、パイオニア達は汗を流して草を刈って道を創りながら進んできた。その後、その道は舗装され草も生えなくなった。その道を観光バスに乗って、ガイドさんの解説を聞いても、昔の草の色も香りも知る由がない。

人の成長は与えた食物がすべてではなく、与えた環境（とくに試行錯誤の繰り返し）により多く影響されるものであることは、衆知の事実である。

これに気付いて、人材教育をすれば問題解決の方向に進むが、往々にして現実のバックログ／メンテナンスのために新任要員を費してしまう。キーマンは、メンテナンスも実践的な意義深い教育と考えるが、新任要員はプログラミング・テクニックのみを学ぶことになる。そればかりか、時間が経過すると新任要員は単純繰り返しの飽き、モラル低下に陥りプログラミングの生産性まで停滞してしまう。

上記4項のうち、本質的な問題点は2.1節の病原体のみが宿命的で完全には避けがたい。この病原体に対しては、その活動を少しでも押さえたり、活動を始める前に対抗する仕組みで牽制することを考えねばならない。

一方、2.2節の延命剤や2.4節の栄養過多は明らかにシステム開発部門が“誤った対応”をした結果といえる。この誤った対応をしてしまった理由の一つとして、情報化社会での価値観でシステム開発を見ることができなかつたためと考えられる。

次に、このような観点から二、三の対応策を考えてみたい。

3. システム部門の変身 =抗体づくり=

前述のシステム開発環境の変化に対応するのに、一つの方法は新しい価値観の導入であり、一方は攻撃により守りを固めるという考え方がある。

3.1 ワン・パターンからの脱皮 一仮性近視は、視力回復できる一

工業化社会の特徴の一つは、ワン・パターン／大量生産方式を支える価値観が主流となっていることである。この価値観は情報化社会の新しい波に反する性格が強い。

たとえば、『システム開発の生産性向上』という命題に対して、①標準化、②構造化プログラミング、③ドキュメンテーション、④詳細な設計、⑤要求仕様の凍結、等はいずれも工業化社会の大量生産用尺度での方策である。これらの一つ一つは劇的（たとえば、100%以上）に生産性を高めるのには役立たないことは、多くのシステム開発部門が体験した通りである。

まず、システム開発による真の成果物は、ソース・プログラムでもプログラム仕様書でもない。物理的な形が見えない知的生産物である。

次に、元来「生産性」という言葉自身が工業化社会での代表用語の一つである。すなわち、要した労力量に対してどの程度の製品を生産したかという価値観の中には、

生産物（物理的な製品）や労力は常に均質であるという大前提が存在している。システムの実体の一つであるプログラムの優劣は、決してその大きさに測定できないことは明らかである。さらに、知的生産の過程では、労力を時間で評価するのは大いに問題である（労力の均質が前提）。

現実には、プログラミングの生産性は個人差（能力差）によって桁違い（10倍以上）な値になることはシステム部門の管理者によく知られている事柄である。

情報化社会で通用する価値観とは、「均質化／同一化」でなく「多様化」の方向であると言える。桁違いの差が現われる原因には、製造主体の違いが考えられる（図1）。

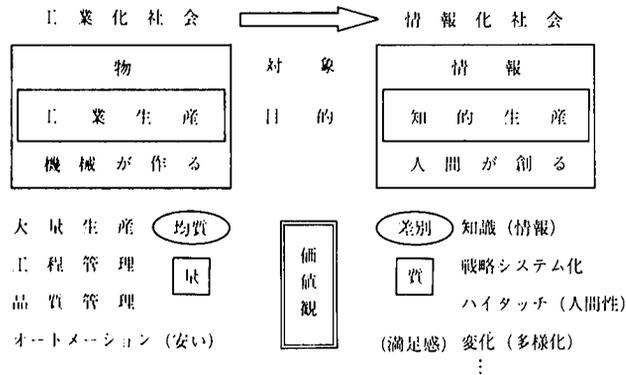


図1 情報化社会の価値観

Fig. 1 Value criteria in informationalized society

工業化社会では「機械」が単純な繰り返しによって製品を作り出す。知的生産物であるプログラムは、「人間」が創り出すもので、一般的には単純な繰り返しだけではない。よってプログラムの価値は、これを作成するのに要した時間やその大きさには必ずしも比例しない。そのプログラムの「有益度」という尺度の方がより適切である。さらに、そのプログラムをどのような目的で使用するか（使い方）で、その価値は大幅に異なる。すなわち、同じ実体に対して視点の違いにより大幅に価値が左右されるものである。また、時間が経過することによって、その価値が激変するのも一つの特徴である。工業製品では、この変化はずっと緩やかである。

このように情報化社会で重要な価値観の一つは、「多様化への対応」であり、これを達成するのに重要なのは「自由な視点」や「分別能力」である。いわば、“知的視力”がポイントとなる。物事のより本質に近い領域を見透す“視力”がワン・パターン価値観からの脱出には肝要である。

次に、より具体的な例としてプログラムの多様化について考えてみる。

- 1) 使用頻度
 - 毎日使う : 日次
 - 月に1回使う : 月次
 - 年に1回使う : 年次
 - 1度だけ使う : 単発

開発されたシステムが運用に入ってから、プログラム単位でその使用頻度を見ると、上のように単純に分けても毎日のように使用するものから一度だけ使う単発の

関連プログラムはすべて変更しなくてはならない。これに対処しようと、プログラム中のデータ記述をプログラム本体から独立させる例は少なくないが、該当項目がどのプログラムで使用されているかを管理する必要が生じてしまう。

上記の例のように、多様化はあらゆる角度から発生し、今後どの方向から出てくるかは予測しにくい。よって、多様化にどれだけ柔軟に対応できるかは、世の中の変化を大局的に見分ける視力が鍵をにぎる。

3.2 スクラップ&ビルド 一体質改善—

2.1節の論理により、システムのメンテナンスをなくすることは不可能である。このガン細胞を抗ガン剤で迎え撃つ方法もあろうが、多種多様なガン細胞のすべてに有効な抗ガン剤の発明はまだ実現していないし、副作用も心配である。

健康を取り戻すには、まず早期発見であり、病巣が局所的なうちに摘出手術をするのが現実的である。その後、健康管理に気配りをすると同時に、体質改善のための努力を継続すべきである。

これをシステム開発に当てはめて考えると、図2のようになる。

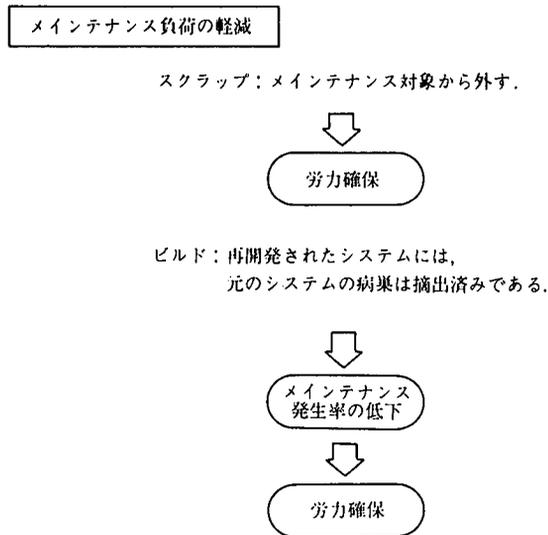


図2 メインテナンス負荷の軽減

Fig.2 Alleviation of maintenance load

この論理では、スクラップによる労力確保とビルドによる後のメンテナンス率の低下によって得られる労力の量は、再開発に必要な労力量を上回る場合には直接的に有効である。これが実現しない場合でも、泥縄的メンテナンスによって強い精神的圧迫感を受けているシステム開発部門の精神衛生管理上の効果は無視できない。放置すればモラル低下を招くが、スクラップしてでも体質改善するという姿勢を示せば管理者の強い意志が部下に伝わり、悪循環に歯止めをかけることも可能となる(図3)。

病原体の中で、最も悪質な「要求仕様の凍結」こそガン細胞の代表である。このガン細胞を叩き潰せばスクラップ&ビルド体質に変身しやすくなるし、スクラップ&ビルド体質になれば要求仕様の凍結にこだわる必要がなくなり、良い方向(快方)に回

スクラップ&ビルド体質による変化

- ・要求仕様の凍結にこだわらない。
- ・最適化（CPU 使用率、ファイル・フォーマット、オペレーション等）を施さない。



より変化しやすいプログラム：プログラムの体質改善

図3 スクラップ&ビルド体質

Fig. 3 System development based on "scrap and build"

り始める。

3.3 戦略意識 一病いは気から一

元来、パイオニア精神の持ち主であったシステム部門の中堅管理者が「守りの体制」になろうとしても、不慣れな姿勢を強いられストレスが蓄積される。守りの技術を知る必要のなかった人が、急に守りの姿勢をとろうとしても無理であろう。

これから脱却するには、「攻撃は最大の防御なり」式の意識改革を行うのが、その人に最もフィットしている。まず、他企業の同じ立場の人と会見し、自分だけが孤立していないことを知り、一応の安堵感を得ることである。これによって、事態は何も好転しないが、中堅管理者の「気持ちの切替え」のきっかけになる。この切替えができれば、ただちに攻撃にまわるべきである。

最初には、自らをシステム部門から距離を置いて、別の自分（今の自分の代理役）を作り上げ責任を彼に転嫁してみる。そして、自らを代理役の参謀にしてみる。参謀は事態打開のために種々の戦略を練らねばならない。①問題点の選別、②作戦の企画、③作戦の評価、その他を考える。このような攻撃型の発想は、中堅管理者の得意とするところである。ここまでくれば、あとは歯車は良い方向に回り始める。

たとえば、人材開発を考えてみる。

システム開発過程を生産ラインの工程管理と見做してきた時には、作業を標準化して、要員を「均質な機械」と見做して管理してきた。ここに、誤りがあった。まず目先の戦力を犠牲にしても、開発要員の一部をニュー・パイオニア人材に位置付ける。そして集中的に教育を行う。このとき、自らが部門の創設期に体験した試行錯誤の内、100パーセント無駄であった部分は除いて、必要な部分（必要な無駄を含む）だけを体験させるようにすれば極めて効率的にニュー・パイオニアを育て上げられる。選ばれた人材のモラルが向上するのは間違いない。当然、次にニュー・パイオニア候補のローテーションを考える。ハイ・レベル技術者はロー・レベル技術者の10倍以上の能力を発揮するのが、知的生産活動の特徴である。よって、1人のロー・レベル技術者を増すことに努力するより、1人のハイ・レベル技術者を養成するためにこそ10倍以上に頑張るべきである。

次に、新規開発を要請されたシステムを「戦略システム化」することを考える。仮りに、企業トップがそのシステムを企業戦略とまで考えていなくても、システム開発を決断した事柄であるから、競合企業も似た状態の場合が多い。システム部門が参謀

的な立場で要求システムの論理的分析を行えば、競合企業との差別化が可能になるほどに機能分析ができるものである。ここで多大な労力を要するが、基本設計の質は飛躍的な向上が期待できる。さらに、将来の変化に対応しやすい構造になる。透視能力向上につながり、次期戦略作成にも役立つ。

4. MAPPER 1100 =健康維持=

4.1 最も重視すべきはデータベース設計 —自然体—

MAPPER 1100 には、第四世代言語としての多くの機能がひと通り盛り込まれていて、その使いやすさ・機能の統合・シンプルなデータベース（以下、DB と略す）等を始め多様な特長をもっている。

MAPPER のレポートを、DB と見るかファイルと見るかも利用者の自由で、その都度アプリケーションに合わせて考えて良い。単発的なアプリケーションが独特なデータを扱う場合には、DB として考えないのは当然である。しかし、一般的には DB とした時に MAPPER の特長が最大限に発揮できることは言うまでもない。

DBMS（データベース・マネージメント・システム）製品が登場するまでプログラムとデータの関係は、およそデータはプログラムの付属品の存在であった。よって、データ項目の定義はもちろん、データ・フォーマットもそのプログラムに都合の良い（最適化された）ものであった。しかし、データの一元化がなされないと、処理結果間に不整合が生じ、これをプログラムで補正するのは不可能であることを経験して、DB 思想が生まれ重要さが認識された。ここで、プログラムとデータの関係は主従逆転し、データ中心的発想が生れた。

初期には、DB はそれ以前の「しがらみ」を取り払うまでの説得力にはならず、新アプリケーションの付属品としてアプリケーションごとの DB として位置付けられることが多く、データはプログラムに従属的であった。局所的なデータの一元化であった。最近、銀行の第3次オンライン開発で、いよいよ従来の「しがらみ」を切り取り本格的なスクラップ&ビルドが始まった。

もともと、プログラムは機械（コンピュータ）のためのものであり、データこそが人間にとって役立つものである（役立つデータこそが情報である）。データを中心に考えると、DB 設計の重要性はシステムの生命を決すると考えてよい。

可能な DB 設計の一つの指針は、「データの本質に近いデータ構造」を採用することであろう。その結果は、「どのアプリケーションにも、最適でない DB」になることは明らかであるが、①無理がなく、②多様化するアプリケーションに対応しやすい性格を期待できる。

最適化とは、「ある最適性を評価する基準」に対してのみ最適解を与えるもので、異なる評価基準に対してはおおよそ最適解とはならない。

このように考えると、MAPPER の物理データベースについては、1 行 132 桁の制限に対して、一般的にはタイプ・モードの複数化で対応すべきで、レポート中に 1 件当たり複数行で対応すべきではない。これは、おおよそ何かある最適化意識が働いている。

4.2 会話機能こそ、MAPPER の生命 —過保護の排除—

MAPPER 1100 は、エンドユーザ・コンピューティングに必要な機能を、ほぼ包括

的に提供している。

- 1) データ入力／ファイル作成
- 2) データ検索／帳表出力
- 3) 計算／データ操作
- 4) 日本語／グラフ
- 5) 既存ファイルとのインタフェース
- 6) ワープロ
- 7) 通信
- 8) OA
- 9) アプリケーション作成
- 10) DB 管理機能
- 11) マイクロ・メインフレーム結合機能

これ等の機能は、あらゆる組み合わせが可能であることに価値がある（統合化）。

システム部門は、「エンド・ユーザはコンピュータについてまったく無知」であろうという立場や「貴重なデータの安全性」という理由で、アプリケーションを用意する場合、“完全に準備されつくした環境”を作るべきと考えてしまう。しかし、このことにエンド・ユーザが満足感を覚えるのは、運用前か運用当初の一瞬で、しばらくすれば次の興味がエンド・ユーザの心の中に芽生えるものである。

世の親が、幼い我が子におもちゃを与えるとき、高級で完成度が高い（高価な）物を与えると最初は大変に喜び熱中しているが、すぐ飽きてしまう。逆に、素朴で単純なおもちゃ（素材）を与えれば、かなり長い期間に渡っていろいろと工夫して自分の遊びを創り出している。これが、我が子の成長に大いに役立つことは誰でも知っている。

このような意味で、エンド・ユーザに奇麗で重装備なシステムを与えてしまうより、どのような組み合わせも可能な素材的システムを与える方がずっと効果的である。エンド・ユーザはアプリケーションに対しては専門家であるので、システム部門とはまったく別の見方でコンピュータ・システムを見ることができる。また、より自由な発想でさまざまな工夫をして、システム部門を驚かすものである。

これを実現させるのに、MAPPER 1100 は、これまでの他のシステムに比べ圧倒的に有効なシステムといえる。

エンド・ユーザ・コンピューティングは、エンド・ユーザに簡易プログラミング言語を与えて、プログラム作成を分散化させることではない。それでは、いずれシステム部門が犯した道のりを繰り返すだけで、むしろマイナス面が多い。エンド・ユーザの意志でデータを加工して情報を創り出すのに都合の良い環境とは、適当に素材的な機能があって、素材の集まりをひとまとまりにする機能を加えることから始まる。MAPPER 1100 の会話コマンドはそのまま素材機能に当たり、同 32 R 版の RUN 作成コマンド（ラン自動作成）は素材の集まりを自動的にひとまとまりにする機能そのものである。これが、エンド・ユーザによるプログラミングの最初の姿として望ましい形態である。この段階での徹底的な学習が、データ処理に対する「正しい理解」に役立つ。これに比べれば、ループ構造や条件分岐等のプログラミング・テクニックは

MAPPER にとって枝葉に過ぎない。

この意味からも、エンド・ユーザ・コンピューティングにとって、MAPPER 1100 の価値のほとんどは会話機能である。エンド・ユーザがデータ処理について「正しい理解」を持つことは、エンド・ユーザのみならずシステム部門にとって極めて好ましいことで、システム部門の健康維持に多大な貢献をすることは明白である。この効果をより確実に得る目的で、システム部門としてはエンド・ユーザ・コンピューティングを推進するインフォメーション・センタ機能を提供しても余りある。インフォメーション・センタ機能がエンド・ユーザの精神衛生に好ましいことも自明であるので、両者の歯車が良い方向に回転し双方の健康に役立つ。

4.3 プロトタイピング —遊び心—

MAPPER 1100 のラン機能（プログラミング機能）は、システム部門がプロトタイピングを行うことを可能にしている。システム開発で、プロトタイピングは従来型開発のガンである要求仕様の凍結を否定することができる点で極めて有効である。

すなわち、システム部門が病原体に犯されにくい体質をつくるのに、プロトタイピングは一つの有力な方法である。まず、病原体の活動を事前に知ることができる点で有効であり、本番後の活動開始までの時間を伸ばすのにも役立つ。次に、基本設計も従来と違って、体験に基づく試行錯誤の結果を反映できるので質的に向上し、そのフィードバックのかかった安定したシステムとなる。さらに、本質的に第2、第3段のフィードバックをかけやすい体質になるので、より効果を期待できる。

システム部門の開発要員にとって、プログラム仕様書に基づく COBOL プログラムの作成とプロトタイプの実行とは大いに異なる。前者の工業的生産に比べ、後者のプロトタイプ作成者は子供がおもちゃをひねくり回すように「遊び心」を取り入れて試行錯誤を行うことになり、知的生産の特徴がはっきり現われる。

この遊び心は、知的生産をするためには必要不可欠で最も重要な要素と考えられるので、第一義的に大切である。ランの品質向上はもちろんのこと、システム部門の健康維持に意義がある。より健康であれば、次の知的生産に良い影響となり、システム部門の人間性回復、すなわち情報化体質となる。

4.4 MAPPER KIT 1100

MAPPER 1100 に対して、設計支援・開発支援を行う MAPPER KIT 1100 は、いくつかの効能を持っている。

設計段階での MAPPER KIT 1100 の特徴は、

- 1) トップダウン・アプローチ
- 2) 設計内容の図形表現
- 3) ドキュメンテーション生成支援
- 4) 標準化設計

等である。

ここで、トップダウン・アプローチはとくに重要である。いくつかの文献でトップダウン設計とボトムアップ開発を説くが、この二つが途中のどこかでぴったり結合することは考えにくい(図4)。とくに、ボトムアップ開発で同じパターンが出現すると、サブルーチン化(共通化)してプログラムの作成工程を最適化してしまう。しかし、

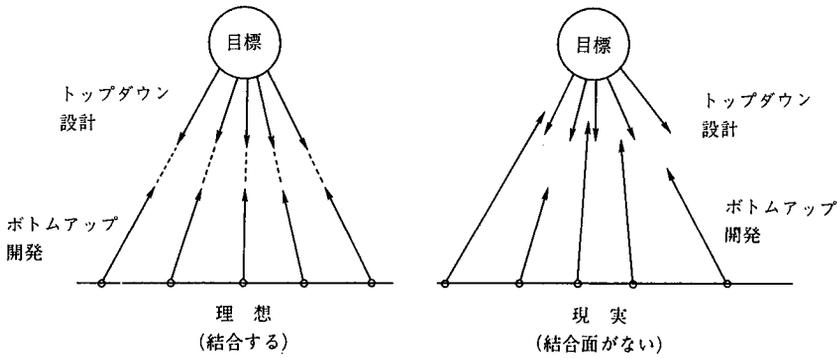


図4 トップダウン設計とボトムアップ開発

Fig. 4 Top-down design and bottom-up development

これがシステムの成長にとって障害となり、ガン細胞化しかねない。

2)の図形表現は、設計思想のような知的生産物を表現するのには、文章に比べて圧倒的に優れている。ある程度以上に時間が経過した後で、記憶を取り戻すのに図形はより連想力を助ける。

3)ドキュメンテーションの作成はもちろんのこと、ドキュメンテーションのメンテナンスは人間にとって著しく苦痛である。これを支援するソフトウェアの存在は直接的に効果があるし、ドキュメンテーションを確実に整備しようとする気持ち(心)を起こさせる効果はさらに大きい。

4.5 MAPPERによる教育/MAPPER KITによる教育

4.2節で述べたとおり、エンド・ユーザ教育にMAPPER 1100は、とくに優れた性格を持っている。さらに、自ら教育用の教材をデータベースに蓄えており好都合である。加えて、音声付CAIシステムLearn-upも利用可能なので、新任者教育の第一弾として十分に役立つ。

一方、4.4節のようにシステム部門の要員教育には、MAPPER KIT 1100は極めて優れている。一般に、システム部門の中堅管理者といえども、後継者を養成する手順を論理的に整えることは大変困難である。MAPPER KIT 1100の標準化された設計手順は、設計(知的生産活動)の教育過程を標準化することになり、設計教育の管理が可能となる点に意義がある。ただし、教育内容の標準化とはまったく異なる。

5. MAPPER 戦略 =体力増強=

5.1 システム部門のツール

MAPPER 1100は、第四世代言語と呼ばれ、エンド・ユーザ向けというのが、一般的で表面的な評価である。しかし、非システム部門向けということではないはずである。エンド・ユーザにとって使いやすい環境であれば、システム部門にとっては極めて使い勝手が良い環境と考える方が素直である。高級なもの・著しく使い勝手が良いものは、それを維持するためのオーバーヘッドが大きく、システム部門から見ると非効率的であるという評価が通用するのは、多様化の進行する中ではもはや一部に過ぎない。システム部門の「紺屋の白袴」的な性格を代表している。ミクロ的に見てオーバ

ヘッドが気になっても、少しマクロ的な立場になれば役立つ事柄は多くある。これを見分ける能力こそ、システム部門が一番強いはずである。

たとえば、システムの維持管理を行う場合にシステム部門は、いくつもユーティリティや自己開発したシステムを使って作業を進めるが、多くはデマンド処理環境で行うのが一般的である。確かに、それらの起動やパラメータを与えるにはデマンド環境で十分である。しかし、もう一段複雑な作業を行いたい場合には、通常の TSS 環境 (CTS や IPF 1100 等) では不足する機能がある。ユーティリティや自己システムの印刷出力イメージに対して、2 次加工して次の作業を進めたいことがしばしば起こる。このとき、この 2 次加工のために、TSS 環境ではほとんどプログラムを必要とするが、MAPPER 1100 のバッチポート機能によって取り込めば、MAPPER の会話機能が使え、大半はこれで決着するはずである。もし、ランが必要になっても、第三世代言語のプログラミングに比べれば圧倒的に容易になる。

システム部門が、MAPPER 1100 をツールとして使い始めていろいろな体験をして、ノウハウを蓄積することが、エンド・ユーザをサポートするインフォメーション・センタ機能の充実に役立つ。何よりも、システム部門の効率化を促し、意識改革を可能にする。むしろシステム部門の情報化を戦略的に進めて、企業の戦略の足を引張ることのないようにすべきである。

5.2 MAPPER インターフェイス

コンピュータと人間のマンマシン・インタフェースを考えると、5.1 節の例でも MAPPER 1100 の持つユーザ・インタフェースは、他の TSS 環境に比べてもいくつかの優れた特性がある。もちろん、MAPPER の弱点もあるので、すべてを MAPPER に任せるのは論外である。TSS/デマンド処理の優れている点、MAPPER の優れている点のそれぞれを抽出し、TSS の弱点や MAPPER の弱点は可能な限り使わないようにすること、すなわち多様化への対応が肝要である。

とくに、MAPPER 1100 を“高速のフル・スクリーン・エディタ”として見ると、他のシステム (たとえば IPF 1100) と比べて差別化ができる。フル・スクリーン・エディタは使用者に対して、使い勝手感の大半を占める。さらに、高速性は最も重要である (MAPPER 1100 の英文ワープロ機能は、これらの点で最強と考えられる)。

次に、MAPPER 1100 の日本語対応も効果的である。漢字シフトコードを持たないことは、使用者に日本語と ASCII コードとの区別の必要性を強制しなくて済む点でとくに勝っている。また、MAPPER 1100 からの印刷出力がとくに何の指示をしなくても、漢字対 ASCII が 2:1 の関係のままになっているのは画面イメージとの共通性から都合がよい。ただし、最終アウトプットとして適切かどうかは、異なる評価がある。これも多様化の中からの選択であり、使い分けこそ最も重要となる。

6. おわりに

情報化社会を、一言で代表する真の価値観が何であるかは、まだわかっていないと思われる。工業化社会とは大いに違っているのは確かであるから、多様化とか人間性が重要であるのは、これまでの反省であるかのように思える。

システム開発環境は、この 20 余年の間に一気に情報化社会へ移行してしまった。こ

の急激な変化速度がシステム部門に過去をゆっくりかえり見ることを許さず、変化を見る目を失って、不透明感を増幅している。

このような中で、MAPPER 1100 が切れ味の良い刃物で有力な武器になることはすでに多くの事例で報告されている。しかし、これは MAPPER を使えば解決するというのではなく、MAPPER という新しい刃物を使ってでも新しい道を開こうとする意気込みこそが、MAPPER システムの成功につながっていることは間違いない。

有力な武器は、いうまでもなくその使い方を誤れば自らの命を縮める結果になりかねない。MAPPER によって、より健康的なシステム開発環境を作るには、「細心にして、大担」な MAPPER 戦略（知恵）を採らねばならない。

執筆者紹介 日置伸一 (Shin-ichi Hioki)

昭和24年生、47年愛知工業大学電気工学科卒業、同4月愛知工業大学電子工学科・助手、60年同計算センター講師。最近10年間は、主に大学事務処理システム的设计・開発に従事。愛知工業大学・計算センター講師、情報処理学会、オフィスオートメーション学会会員。



論文 MAPPERKIT におけるデータ辞書

Data Dictionary in MAPPERKIT

井上比佐乃

要約 システムの大規模化・複雑化につれてシステム開発のあり方が改めて問われている。ソフトウェア工学をはじめとして、さまざまな形で研究が進められてきた解答の中からデータ辞書と MAPPER を取り上げ、MAPPER 上のシステムである MAPPERKIT におけるデータ辞書の扱い方について評価した結果を述べる。

具体的には、システムでのデータ辞書の位置付けと機能、および意義について一般的に説明した後に、MAPPER 上にデータ辞書を置いた場合の機能・効果と比較する。以上を前提として、MAPPERKIT でデータ辞書を活用することによって行われたシステム開発の合理化と自動化について論じている。

Abstract As system development has become complicate and large, its studies have brought into various methods, especially software engineering into existence. MAPPER and Data Dictionary are some results of these studies. MAPPERKIT is a tool using Data Dictionary which supports application system development and maintenance based on MAPPER.

This paper discusses Data Dictionary in general and evaluates the Data Dictionary in MAPPERKIT.

1. はじめに

コンピュータの歴史の中で、ハードウェアの発達は一躍的で目を見はらせるものがある。これに対するソフトウェアの進歩は遅々としたものである。これは後者が、人間の能力に大きく依存していることがその原因である。加えて、

- 1) コンピュータの普及によるソフトウェアの需要増加
- 2) 個々のソフトウェアの大規模・複雑化
- 3) 技術者の不足と人件費の高騰

などの環境の変化からくるソフトウェアの量的質的不足は、目に見えて拡大している。「ソフトウェアの危機」が叫ばれて久しいが、実状は一般に考えられているよりも深刻である。単にシステムを開発するだけで良しとする時代は終わった。

より効率的なシステム開発を目指す方策として、

- 1) 開発・保守の効率向上のための開発支援ツールの利用
- 2) ソフトウェア・パッケージの利用
- 3) 利用部門による開発

が一般的に提示されている。その中の開発支援システムの一つとして MAPPERKIT 1100 がある。MAPPERKIT 1100 は、開発工程を合理化し、開発・保守作業のエンド・ユーザ・レベルまでの自動化に成功したシステムである。なかでも、データ辞書とディレクトリの内蔵による自動化は、これまでにない効果を上げている。本稿ではこの点に注目し、データ辞書の機能と意義について述べると共に、MAPPERKIT 1100 におけるデータ辞書の効果を解析する。

2. 背景

ソフトウェア開発にたずさわる者にとって、ソフトウェアの生産性・信頼性の向上は重要な問題となっている。しかしながら、これまでのような「人間主体」のソフトウェア開発を脱け出さない限り、コスト、生産性、信頼性の向上は望めない。

「人間主体」からの解放策の研究は、いろいろな形で進められてきた。とくにソフトウェア工学の名は、ソフトウェアの危機の言葉と共に広く知られるようになってきた。ジャクソン法や SADT などのソフトウェア工学の成果が、情報処理業界から続々と出されている開発支援システムに反映されている。また、別方向からの試みもある。第4世代言葉は、開発作業をエンド・ユーザの手にゆだねることで、生産性、品質の向上を図っている。

データ辞書は、ファイル、データ、プログラムなどの相互関係を一括管理することで、プログラム数の増加や保守作業によるシステムの複雑化や信頼性の低下を防ぐとするものである。これによって設計ミスなどによる手戻り作業を減少させ、生産性および品質を向上させるようにしている。通産省のΣ計画もこれに答えるものである。

2.1 MAPPER の出現

第4世代言語の一つとして出現した MAPPER は、プログラミングの知識がなくても使用者が自らの業務を容易に EDP 化できることが特徴であり、EDP 処理にコンピュータの専門家でない人々を参加させることに成功した。

MAPPER が出現するまでは、利用部門が EDP 化を希望する業務と EDP 部門で EDP 化を可能とする業務のギャップが大きく、しかもギャップは拡大しつつあった。このギャップを埋めるためには、

- 1) プログラムレス処理
- 2) リアルタイム処理
- 3) エンド・ユーザに理解しやすい簡易言語による処理

が必要不可欠であった。

MAPPER は、上の機能をすべてそなえた簡易リアルタイム・パッケージであり、報告書の作成、管理、検索、更新、編集などが容易に行えるリレーショナル型データベース・システムである。

MAPPER を使うことによって、

- 1) 簡単な作業指示で大型コンピュータを直接的に活用できる。
- 2) すでに EDP 化されているデータは、そのまま利用できる。
- 3) 非定型業務および少量のデータ処理業務の EDP 化が可能である。
- 4) 漢字、グラフ、カラー機能、電子メール・システムなども容易に有効利用できる。
- 5) 定型業務はランとして登録することで簡単に実行できる。

などの利点がある。とくに非定型業務と少量データ処理業務が容易に EDP 化できることにより、ソフトウェア開発作業の生産性が大幅に向上した。

2.2 MAPPER によるシステム開発の問題点

MAPPER を導入することで、ソフトウェア開発のコード化などの作業負荷は大き

く減少し、小規模なシステムであれば、設計の段階も無視して開発できるようになった。しかし、逆に無計画なランやレポートが増加し、混乱を招いている面もある。無計画なシステムは、当然のことながらドキュメントも作成されず、開発した人間以外には利用できないことから、短命なシステムに終わってしまいがちである。

一方、大規模なシステムや複雑なシステムの開発には、これまでと同じように高度の専門知識が必要とされている。また、MAPPER データベースの特異性を考えたとき、MAPPER のシステム開発に合った設計技法が是非とも必要になる。総合的なソフトウェア開発を、MAPPER を取り入れて行おうとしたとき、

- 1) 設計技法
- 2) ドキュメンテーション技法
- 3) 設計関連情報の管理

の指針ともなる開発支援ツールがあれば、生産性、信頼性はさらに高まるはずである。以上のもとに生まれたのが MAPPERKIT 1100 である。

MAPPERKIT 1100 は、MAPPER に加えて、より多くの利用者がより高度のシステム構築を可能にすることを目的として開発された、利用者のためのアプリケーション開発支援システムである。

3. MAPPERKIT 1100 の概要

MAPPERKIT 1100 の全般について簡単に説明しておく。

3.1 MAPPERKIT 1100 の構成

MAPPERKIT 1100 は、

- 1) システム設計、およびデータベース設計を容易に行うための利用者向け実践的開発技法と、そのドキュメント・システム
- 2) 技法に基づき、分析・設計・開発・使用・保守を支援するデータ辞書をベースとした開発支援ソフトウェア群

から構成されている (図 1)。

開発技法では、業務分析からシステム作成までを 5 段階に分け、さらに細かいステップに分割している。各ステップでは、設計方法を手順的に定めることによって、設計について知識の浅い者でも、一つ一つのステップを踏んで高品質なシステム開発を可能とした。とくにデータベース設計においては、ジャクソン法を利用しており、処理をデータ構造図で表すことにより、各処理から必要となるデータ同志の関係を容易に導き出している。

また、各段階で作成するドキュメントは、アプリケーション内の MAPPER レポートに入力することによってシステムと一体化し、これまでにありがちだったドキュメントと実システムとのギャップを削減することに成功した。それに加えて、ドキュメントは単なるドキュメントに止まらず、システム稼働後は各種の運用マニュアルとして有効利用される。

開発支援ツールは、以下の 2 種類に大きく分けられる。

- 1) 設計支援および開発支援ツール (KITBUILD)
- 2) 保守支援ツール (KITAID)

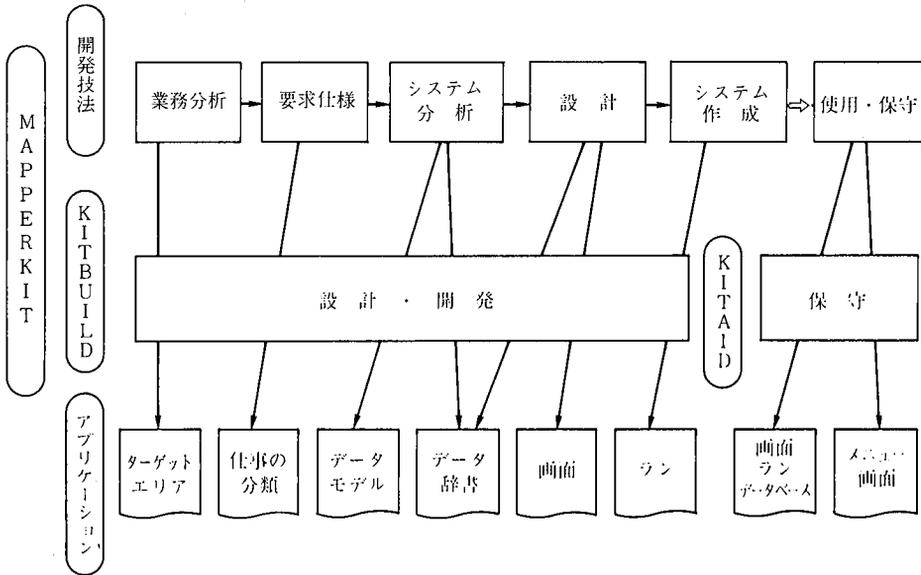


図1 MAPPERKIT 1100 の構成

Fig.1 MAPPERKIT 1100 system

設計支援ツールは、システム設計時のドキュメント作成を支援する。標準開発技法におけるドキュメント作成はいくつかの段階に分けられているが、前段階に作成したドキュメントから次段階に必要なドキュメントを作成する。内容も必要な部分を前段階のドキュメントから自動的に選択しコピーする。また、各種のドキュメントからダイアグラムを作成し、各設計段階での確認作業を容易にしている。

開発支援ツールは、これまで面倒といわれていたレポート、画面、ラン・スケルトンを自動作成する。この部分の自動化によって、システム部門の人間はもちろん、エンド・ユーザにも標準化されたシステムが容易にできるようになった。

保守支援ツールは、システム作成段階で項目、レポート、ランなどがデータ辞書およびディレクトリに登録されていることを利用して、自動的に保守作業を行う。保守の自動化によって、システムの信頼性は飛躍的に向上した。

3.2 MAPPERKIT 1100 の特徴

MAPPERKIT 1100 の特徴をまとめると以下ようになる。

- 1) システムのライフ・サイクル全般（設計・開発・保守）の支援
- 2) データ辞書を中心としたシステムの開発と保守
- 3) 容易なドキュメント作成
- 4) データ辞書/ディレクトリによる充実した管理
- 5) 自動生成による開発の生産性向上
- 6) 自動変更による保守の生産性向上と信頼性向上

MAPPERKIT 1100 を利用してシステム開発することにより、全体的な生産性の向上が期待できるだけでなく、

- 1) 効果的なデータベース設計と処理設計、および安全な保守による品質の向上
- 2) 開発手順、ドキュメント、使用者インターフェースにおける標準化の推進

3) 標準的なマン・マシン・インタフェース、ガイド機能による使いやすさの向上が望める。

4. データ辞書

データ辞書とはデータ資源の登録簿であり、各種データの定義情報を1か所に収めることによって一元管理を可能とするものである。本章ではデータ辞書について整理し、その目的と機能について記述する。

4.1 データ辞書の目的

システムの大規模化・複雑化に従って、扱うデータの名前や属性、レコード、ファイル、プログラムなどの情報が膨大になり、それぞれの関係が複雑になってきた。従来のシステムにおいては、これらはユーザやSEの記憶とドキュメントの形で一応管理されていたが、システムの相互関係が複雑化するに従い、人手による業務別のデータの把握・管理ではもはや追いつかなくなってきた。しかも、システムの改善や保守作業の比重は年々増加の一途をたどっている。それに従って、システムの保守や拡張の際に、関連情報の変更もれや情報間の整合性に矛盾が生じ、システムに混乱をきたす危険性も増加している。このような問題を解決するためには、前述のような情報を一元管理できる機能が必要である。

データ辞書は、データ、レコード、ファイル、プログラムのソース定義ライブラリとして、データ、レコード他の名前、属性、正当性、編集仕様、説明の記述などを保存するものである。これらの情報をもとに、以下のような目的をもって構築される。

- 1) 各種情報の定義を明確にし、各情報間の整合性を保つ
- 2) データ、レコード、ファイル、プログラムの情報を登録することで、冗長なデータなどの発生を防ぐ。
- 3) 設計の初期段階からデータその他の定義をサポートし、使用法などの標準化を助け、品質向上を図る。
- 4) 修正時の影響範囲を把握し、クロスリファレンスを作成することにより設計の変更時間を短縮し、修正もれをなくす。
- 5) 3)、4)に従ってドキュメント作成と修正を支援し、ドキュメントとシステムの間で起りがちな矛盾を防ぐ。
- 6) 登録した設計情報をもとにコピー・ライブラリを作成し、コーディング作業内の定義部分の作成をサポートする。これによって、システム作成作業の生産性の向上と信頼性を保障する。
- 7) ドキュメントと実際のシステムとの矛盾をなくし、ドキュメントの信頼性を高める。

4.2 データ辞書の機能

データ辞書として備えたい機能を大きく分けると、次の三つになる。図2にその概念を示す。

- 1) 登録機能
- 2) 検索・更新機能
- 3) 自動生成機能

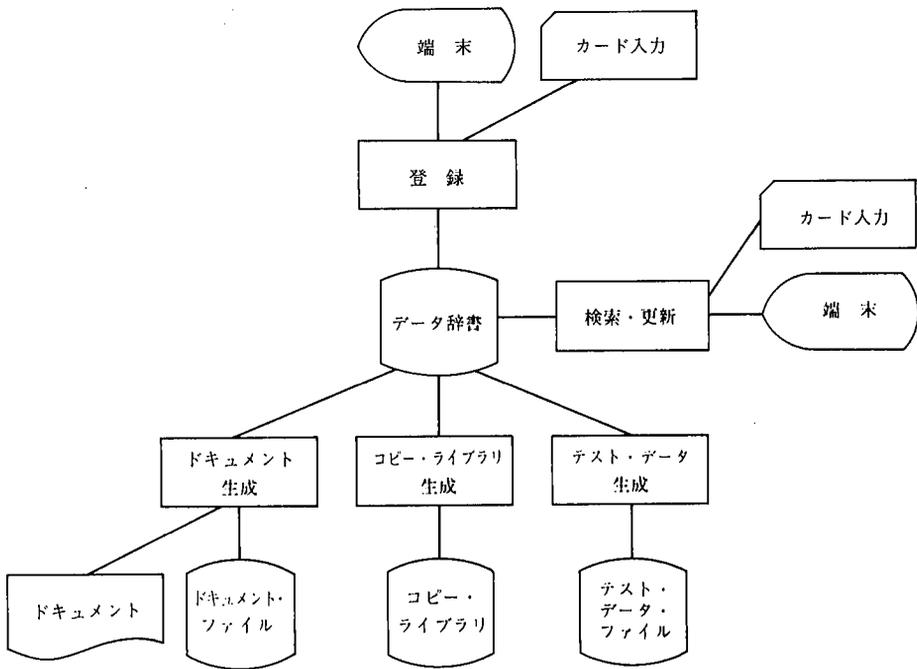


図2 データ辞書の機能概念図

Fig.2 Functions of data dictionary

4.2.1 登録機能

データ辞書の信頼性と生産性の向上のために、情報の自動登録機能が必要である。しかし、もう一步進めて考えると、データ辞書の信頼性を左右するのは、登録されている情報の正当性である。

情報の正当性は、分析および設計の段階でデータやプログラムの定義をいかにして行うかによる。つまり、システム開発の方法論を確立し、データ管理体制を整えた上で、データ辞書への登録を行わなければならないのである。また、膨大な情報を管理するためのデータ辞書であるがゆえに、情報の登録作業も膨大なものとなる。したがって、分析・設計作業の中で、情報のデータ辞書への登録が徐々に行われることが望ましい。当然データ辞書は、設計作業の初めから係わらねばならない。

以上のことから、データ辞書は開発支援ツールと有機的に結びついてこそ有効活用ができるということになる。したがって、登録機能は設計の各段階で必要になる。ドキュメントの作成をシステム化し、その中から登録すべき情報の冗長性、矛盾性をチェックした上で自動登録する方法と、設計情報をもとに人間が登録し、設計ドキュメントをデータ辞書から自動生成する方法が考えられる。

登録する内容としては、データ、レコード、ファイル、プログラムのいずれかによって細部で異なってくるが、

- 1) 識別子
- 2) 名前
- 3) 属性 (タイプまたは種別, 大きさ, 形式, 正当性の判断基準)
- 4) 親 (使用されている先)

- 5) 子 (使用しているもの)
- 6) セキュリティ
- 7) 詳細説明

のようなものが必要と考えられる。

各情報を識別子を与えることにより一意性が明確になって、同義語、同音異義語の取り扱いが確実に行える。識別子によって名前も属性として扱えるのである。使っているもの (ファイルで言えばレコード、プログラムでは副プログラムやコピー・ライブラリなど) 使われているもの (副プログラムにとっての主プログラムなど) の記述からクロスリファレンスを管理することができ、更新・自動生成の要となる。余裕があれば、更新情報も組み込めるとよい。

4.2.2 検索・更新機能

検索機能としては、

- 1) 情報間のクロスリファレンス出力
- 2) 変更要求の認識と変更内容の分析に応じた影響関係の検索
- 3) 個々の情報の内容による検索

が考えられる。現在のデータ辞書の多くは、この検索情報に基づいて人手で更新を行うものが多いが、品質と生産性の向上を考えると自動更新が必要となる。自動更新は、クロスリファレンスをもとに影響範囲を確認しながら行えなくてはならない。たとえば、1件のデータの属性を変更することによって、以下の変更がすべて行われる必要がある。

- 1) レコードの変更
- 2) ファイルの変更
- 3) データを使用しているプログラムの変更

当然、新しい情報に対しては、冗長性・矛盾性・正当性のチェックが必要となる。

4.2.3 各種自動生成機能

データ辞書を利用して生成できるものとしては、次のものが考えられる。

- 1) ドキュメント
- 2) コピー・ライブラリ
- 3) テスト・データ

開発工程の各段階で、蓄積した情報をもとに自動生成することによって、ドキュメントの標準化が可能となり、矛盾の発生も防げる。コピー・ライブラリとしては、プログラム、ファイル、レコード、データそれぞれの親子関係から、データ定義部分や入出力レイアウトの生成が可能であり、標準化とミス防止に有効である。

テスト・データの生成を扱っているデータ辞書は少ないが、正当性の条件などからデータを発生させられると、テスト、デバッグの効率を向上させることができる。また、プログラムのシミュレータを作成することによって、テストの効率を上げられる。

4.3 既存のデータ辞書

ここでは、データ辞書の例として富士通の ADMS と、日立の EAGLE2 を取り上げ紹介する。

4.3.1 ADAMS

ADAMS は、基本設計からプログラム作成・テストおよび保守までを支援している。ADAMS に組み込まれている各種支援ツールは共通のシステム・ディクショナリを使用しており、これによって作業を効率的に行えるようになっている。

登録機能としては、

- 1) コンパイラによる収集登録
- 2) 会計登録更新機能による直接登録
- 3) 既存ソフトウェア資産からの一括登録

がある^[5]。具体的には、以下の形で利用されている。

- 1) 設計工程で定義した基本項目と集団項目に関する情報（項目名、和名、桁数など）の登録
- 2) 画面/帳票とデータ項目の関係の登録
- 3) レコードとデータ項目の関係の登録
- 4) データベースとデータ項目の関係の登録
- 5) プログラムと他の要素の関係の登録

すでにデータ辞書に登録されている内容については、各所で自動参照できるようになっており、情報の一貫性を保つと同時に作業の生産性を上げている。たとえば、画面/帳票およびレコードの定義において、使用するデータ項目を指定することにより、属性や桁数などの必要な部分がデータ辞書から自動参照され、COBOL や PL/I のデータ部などが生成される。

検索機能は、大きく以下の3種類に分けられる。

- 1) 一覧検索……データ辞書が管理している情報の一覧を表示する。
- 2) 相互参照検索……特定の情報に着目し、それらと関連する情報の一覧を表示する。
- 3) 詳細検索……指定した情報の詳細を表示する^[6]。

以上の検索情報をもとに更新作業を行うことができ、保守の生産性および信頼性が向上する。

さらに、ADAMS では、データ辞書の内容をもとにプログラム機能概説図、フローチャートなどのドキュメントを生成する。テスト支援についても各種ツール（テスト用データベースの自動生成、対話型デバッグなど）が用意されていて、プログラム作成効率を上げるよう考慮されている。

4.3.2 EAGLE 2

EAGLE 2 は、システム開発方法論 HIPACE の開発標準手順および開発技法と結びついた支援ツールである。HIPACE では、システム開発標準手順にデータ中心アプローチを採用していて、データ辞書には、このアプローチの中で標準化されたデータを登録している。

同時にライブラリ格納データ（設計仕様書、プログラム仕様書、ソース・プログラム）の関連情報も登録されている^[4]。

データ辞書に登録するデータは、業務システムのデータから体系的に整理し、標準化されたものであるため、作業者の誤解や混乱を防ぐことができる。データ辞書を作

成する場合、登録されているデータの信頼性がデータ辞書そのものの信頼性を左右するため、そのデータの標準化は重要な作業である。EAGLE 2 では、HIPACE を利用した開発作業の中でこの標準化を行っている。

登録する内容は、

- 1) 正式のデータ名、略称、通称、属性など
- 2) ユーザの独自項目（値の範囲、テスト時の標準値など）
- 3) ライブラリ格納データとの関連情報

が主なものである。

このうちライブラリ関連情報については仕様書およびプログラムを解析し自動登録する。その他は仕様画面を使用しての入力となっている。

検索およびドキュメントの出力として、

- 1) データ項目仕様書
- 2) 設計仕様書
- 3) プログラム仕様書
- 4) クロスリファレンス

などを出力する。クロスリファレンスは、任意の組み合わせで生成することができ、これをもとに更新作業を行う。

情報検索においては、「利用者が自由にデータ辞書を利用できるように、アクセス・ルーチンを公開しており、利用者が独自に管理情報を設定し利用することができる^[4]」ようになっている。

データ辞書作成後の利用を活性化することで、より効率的なツールになり得るのではないかと思われる。

5. MAPPERKIT 1100 におけるデータ辞書

さて、ここでは実際に MAPPERKIT 1100 がどのようにデータ辞書を実現しているかについて記述する。

5.1 位置付けと機能

一般的なシステムでのデータ、レコード、ファイル、プログラムなどは、MAPPERKIT 1100 では、項目、レポート、ランに集約される。MAPPER 上では、データ辞書とそれを取りまく環境が単純な形になっているので、データ辞書の構成も他のシステムに比べて単純化され、エンド・ユーザにも扱いやすくできている。

5.1.1 情報定義

MAPPER 上のデータベースは表形式になっているため、データ間の関係を構造化しにくい部分があり、冗長な項目やレポートを作成してしまいがちである。

まず、システム全体を通して項目を標準化し、レポートの設計をしなくてはならない。

MAPPERKIT 1100 では、標準開発技法の中で、要求仕様工程からの分析にしたがってデータ・モデルを作成することによって、データの正規化および標準化を可能にしている。データは、データ・モデルを作成した時点でデータ辞書に登録する。

5.1.2 登 録

登録する情報としては以下のものが考えられる。

- 1) 項目……識別子とその属性 (名前, 桁, タイプ, 項目の内容, 意味, 項目を含むレポートの識別子, 項目を扱うランの識別子など)
- 2) レポート……識別子 (モード, タイプ, RID 番号) とその属性 (レポート名, 桁数, レポートに関するコメント, 含んでいる項目の識別子など)
- 3) ラン……識別子 (ラン名), 格納場所 (モード, タイプ, RID 番号) および属性 (使用しているサブルーチン, 使用するレポートの識別子, 使用する項目の識別子, ランの内容に関する説明など)

MAPPER レポートで上記の内容を管理することを考えると, クロスリファレンス

```

.DATE 15 JUN 87 15:52:17 RID 13 10 JUN 87 KIT-1
* MADS 800 データ辞書 教育システム 506 G 13
*データ:
*項目:名前および説明 :T:桁:A:D:変数:
-----
9A0001 コース・コード 8 B V100
*
* コースを一意に定めるコードで、以下の形式になっています。
* (例: BK01A101)
* B : 教育分野 (A: 新人教育, B: 初級プログラマー, ...)
* K : コース主催 (K: 教育部, J: 人事部, G: 外部講習会, ...)
* 01: 内部コード
* A : コース形態 (A: プロダクト紹介, B: 言語, ...)
* 1 : 開催地区 (1: 本社, 2: 横浜支店, ...)
* 01: 開催回数
9A0002 コース名 20 A V110
*
* コースの名前
9A0003 コース評価 8 A V120
*
* コース受講者によるコース評価の平均 A B C D E による5段階評価です。
* 1: コースレベル 2: 理解度
* 3: 期間 4: テキスト
* 5: 今後の仕事に役立つか 6: 講師の説明
* 7: 上級コースの受講希望 8: 総合評価
9A0004 コースまとめ 10 A V130
*
* 講師およびコース担当者によるコースのまとめ。
* 詳細はコース終了報告書ファイルをご覧下さい。
9A0005 インストラクタ・コード 8 B V145
*
* インストラクタの社員コード
*
* コース名 16 A V155
9AA002 コース終了日
*
* コースの終了日。開始日と同様
9B0001 社員コード 6 B V140
*
* 社員の登録コード
9B0002 社員名 16 A V150
*
* 社員の氏名
9B0003 所属 6 A V160
*
* 社員の所属コード
9B0004 請求チェック 2 A V170
*
* 外部講習会の場合の、受講料支払いのチェック
* 支払い済の場合は○。
9B0005 理解度 6 A V180
*
* テスト等の成績。
* A B C D の4段階評価です。
9B0006 所感 20 A
*
* コースの受講所感
    
```

図3 MAPPERKIT 1100 のデータ辞書

Fig. 3 Data dictionary in MAPPERKIT 1100

情報（項目を含むレポート、使用する項目など）は、親または子のどちらか一方であれば良いことになる（SEARCH 命令の利用）。

MAPPERKIT 1100 ではデータ辞書とディレクトリを用意していて、ディレクトリにレポートおよびランを、データ辞書にはデータ項目を登録している。

データ辞書には以下の情報を登録する（図 3）。

- 1) 識別子
- 2) 項目名
- 3) タイプ
- 4) 桁
- 5) 小数点位置
- 6) ランで使用する変数
- 7) 説明など

MAPPER のランでは、変数として V××× の形式しか使用できないため読みづらく、混乱を生じやすい。データ辞書で項目に対応する変数を定めることによって、統一性が生まれ混乱を防ぐことができる。

ディレクトリは、その情報として次のものを含む。

- 1) アプリケーションの識別子（KIT 名）
- 2) レポート・タイプ
- 3) レポート名
- 4) モード、タイプ、RID 番号

ここでレポート名と共にモード、タイプ、RID 番号を登録しているので、利用者はレポートの管理を無機的な数字でなく、意味あるレポート名を使って行うことができる。

クロスリファレンス情報は、レポートの種類（レポート・タイプ）ごとに作成するレポート説明と定義、画面を定義する入(出)力項目定義、MAPPER ランなどに含まれている。このように情報を分散することにより、データ辞書および他の登録レポートをエンド・ユーザにも理解しやすいものになっている。

項目は前述したように標準化した上で一括して登録する。画面作成などの段階で新たに出てきたデータは、DDFILL などの命令を使って自動登録できる。また、すでにデータ辞書に登録してある項目をレポート作成時などに使用する時には、属性を自動参照できる。レポート/ランの登録は、作成する時に自動的に行われる。

5.1.3 レポート/ランの自動生成機能とクロスリファレンス情報

レポートの作成作業では、直接レポートに書き込まず、RID 0 も作成しない。「レポート説明と定義」の文書に形式（使用する項目）を定義することで、EXP レポート（RID 0 に相当）と各レポートを MAPPERKIT が自動的に作成する。つまり、「レポート説明と定義」がデータ辞書のレポート情報となる。使用する項目が定義の一部に含まれているので、ここからクロスリファレンスを作成することができる。「レポート説明と定義」とそれをもとにしたレポートは、レポート・タイプの名称で関係づけられる。どちらもディレクトリに登録されているので、図 4 のように、データ辞書のレポート情報は複数レポートから構成されることになる。

ランについては、MAPPERKIT 1100 は特殊な扱いをしている。ラン内の画面作成部分をサブルーチン化し、ランから独立させ、定義も別々に行う。画面で使用する項目を「入(出)力項目定義」の文書で定義することによって画面を自動作成する。「入(出)力項目定義」の内容は、そのままクロスリファレンス情報になる。

画面を除くランは、さらに MAPPER ラン (ソース・ラン) と実行ランに段階的分割している。

実行ランはこれまでのランと同じで、ソース・ランは MAPPERKIT 1100 独自のものである。ランで使用するレポート、項目などをソース・ランに記述してから実行ランに変換することによって、データ辞書を有効利用している。ソース・ラン作成画面の中で使用する項目やレポート、画面を指定すると、クロスリファレンスおよび初期入力などを含むラン・スケルトン (ソース・ランの一部) を作成する。

以上のようなレポート、画面、ラン・スケルトンの自動生成により、ランの標準化が可能となる。

さらに、ランのコーディングにおいても有効な機能が用意されている。

MAPPER ランでは、各命令で使用使用するレポートのモード、タイプ、RID 番号を指定するが、MAPPERKIT 1100 では、MODE #, TYPE #, RID # という予約語を用いて、レポート・タイプの番号とレポート名を記述する。実行ランに変換すると、この記述が図 5 のように、それぞれのモード、タイプ、RID 番号となる。

```

@2:LDV V101A3=MODE#,V102A1=TYPE#,V103A5=RID# . *** 910,COURSE-M
@3:SRH,V101,V102,V103.6,,5 'B' 9A0001$,V100 . ***
@LDV V104A3=MODE#,V105A1=TYPE#,V106A5=RID# . *** 920,COURSE-JUKO
@4:SRH,V104,V105,V106.6,,6 'B' 9A0001$,9B0001$,V100,V140 .

@2:LDV V101A3=506,V102A1=D,V103A5=20 . *** 910,COURSE-M
@3:SRH,V101,V102,V103.6,,5 'B' 2-8 ,V100 . ***
@LDV V104A3=506,V105A1=D,V106A5=21 . *** 920,COURSE-JUKO
@4:SRH,V104,V105,V106.6,,6 'B' 2-8,11-10 ,V100,V140 .

```

図 5 モード、タイプ、RID 番号の交換

Fig. 5 Transformation of mode, type, and RID

また、ランの構文内で使う項目の文字位置と文字数は、その項目の識別子の後に \$ をつけて記述すると、その項目の文字位置と文字数に変換される (図 6)。

以上の機能によって、ランは再配置可能なものとなり、ラン作成時のミスも軽減される。そして何よりも、保守の容易なランができ上がる。

```

@2:LDV V101A3=MODE#,V102A1=TYPE#,V103A5=RID# . *** 910,COURSE-M
@3:SRH,V101,V102,V103.6,,5 'B' 9A0001$,V100 . ***
@LDV V104A3=MODE#,V105A1=TYPE#,V106A5=RID# . *** 920,COURSE-JUKO
@4:SRH,V104,V105,V106.6,,6 'B' 9A0001$,9B0001$,V100,V140 .
@LDV V11=2,V12=4 GTO 192 .

@2:LDV V101A3=506,V102A1=D,V103A5=20 . *** 910,COURSE-M
@3:SRH,V101,V102,V103.6,,5 'B' 2-8 ,V100 . ***
@LDV V104A3=506,V105A1=D,V106A5=21 . *** 920,COURSE-JUKO
@4:SRH,V104,V105,V106.6,,6 'B' 2-8,11-10 ,V100,V140 .
@LDV V11=2,V12=4 GTO 192 .

```

図 6 項目の文字位置と文字数への交換

Fig. 6 Calculation of position and characters number of field

5.1.4 検索・更新機能

MAPPERKIT 1100 では、データ辞書の検索機能として、DDID (識別子による検索)、DDNAME (項目名による検索)、DDLOC (属性による検索) 命令がある。この機能を使って、項目の属性変更の影響度を調べることができる。MAPPERKIT 1100 は他のデータ辞書システム、開発支援ツールと異なり、自動保守機能(ADJUST 命令)をもっている。ADJUST 命令は更新を総合的に行い、保守を自動化する。項目の追加、削除、属性の変更と、それに伴うレポート、ラン、画面の修正は、ADJUST 命令が表示する画面に命令と必要項目を入力するだけで自動的に行われる (図7)。

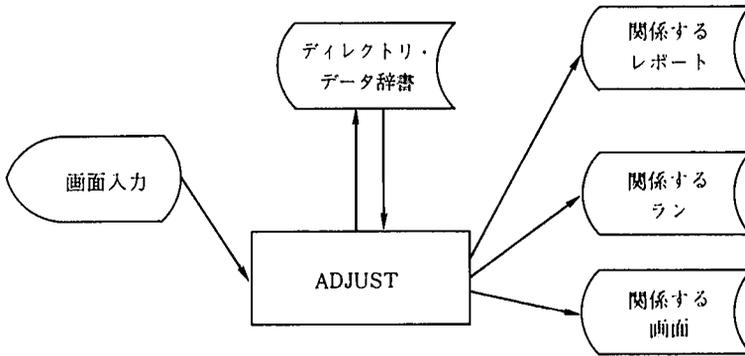


図7 ADJUST 機能

Fig. 7 ADJUST function of MAPPERKIT 1100

5.2 データ辞書の有効性

MAPPERKIT 1100 におけるデータ辞書の効果の主なものを以下に上げる (図8)。

- 1) 情報の冗長性・矛盾性のチェック……MAPPER は、エンド・ユーザでも簡単にソフトウェア開発ができるのが特徴であり大きな利点でもあるが、この特徴ゆえに不要なレポート、冗長な項目が作られ、放置される可能性も高い。ランやタイプの登録はコーディネータの管理のもとにチェックできるが、それ以外の部分での細かいチェックは困難である。データ辞書にそれぞれを登録し、その段階でのチェック機能を利用することで、MAPPER の利用がより効率的に行える。不要なレポートの検査も、データ辞書およびディレクトリを使うことで容易になる。
- 2) レポート/ラン作成効率の向上……データ辞書を使った自動参照、自動生成機能により生産性が向上し、標準化が行える。とくに、ランについては標準化がむずかしいとされていたが、スケルトンの自動生成と他の機能により、誰もが自然に標準化されたランを組むことができる。
- 3) 自動保守による生産性・信頼性の向上……MAPPERKIT 1100 における一番大きな効果は自動保守であろう。自動保守により修正ミス、修正洩れがなくなり、システムの信頼性は大幅に向上する。データ辞書を持たずには成り立たない機能であり、システムをデータ辞書/ディレクトリで管理する最大の目的が保守の生産性向上にある。他の支援ツールがクロスリファレンスの検索表示に止まっているのに比べても、重要かつ効果的な機能である。

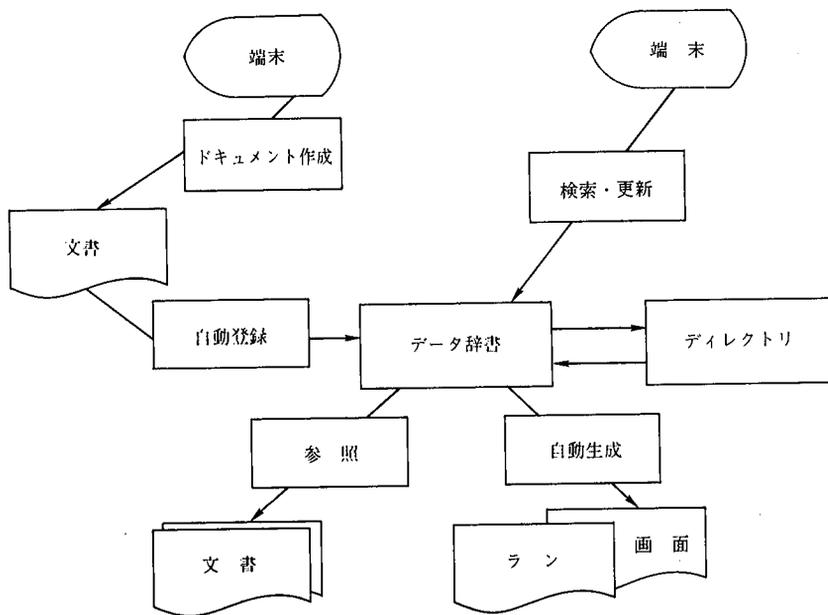


図8 MAPPERKITにおけるデータ辞書の機能

Fig. 8 Data dictionary functions of MAPPERKIT 1100

6. 今後の課題

ここでは、MAPPERKIT 1100のデータ辞書を評価すると共に、汎用性をさらに高めるために、私見ではあるが今後加えていきたい機能に関して述べる。

6.1 データ辞書

現在データ辞書で扱っている項目の属性に加えるべきものとして、項目間の関係がある。

現在データ辞書が扱っているのは項目固有の情報であって、他項目との関連については扱っていないが、データの階層構造を表現できると利用範囲も広がる。

レポートの情報に関しては、以下のものを加えるとより有効である。

- 1) レポート属性（たとえば時系列の日付、営業所/支店コード）
- 2) 登録年月日
- 3) 索引レポートとしての扱い

MAPPERのデータベース構造は2次元の表形式となっており、時系列を扱うためには、2次元から3次元への変換として

- 1) 時系列とRIDの対応表
- 2) 時系列を項目として並べる
- 3) 時系列を行として並べる

の3種のレポートが必要である。したがって、この3種のレポート間の関係もデータ辞書で持つ必要が出てくる。

索引レポートをMAPPERKIT 1100で管理し、索引レポートの下にデータ・レポートが複数ある形で使用できると、さらに大量のデータを効率良く処理できる。

6.2 新規ソフトウェア群

MAPPERKIT 1100 に加えるべきソフトウェアとして以下のものがある。

- 1) 大量データベース用 MAPPER ISAM 開発……ISAM 型のレポート構成を MAPPER 上に組み立てることによって、大量のデータ処理が可能になる。個人ベースで索引レポートとデータ・レポートを管理して使用するのは、システムの混乱を招く元となるが、データ辞書およびディレクトリで管理し、データ辞書に ISAM 情報を登録する機能と、その ISAM 情報に従ってレポートを生成する機能を合わせることで、レポートの標準化、信頼性および効率の向上を目指す。
- 2) グローバル・データ辞書……システムは個別に独立しているのではなく、相互にマスタ・ファイルなどの何らかの関連を持ちながら存在している。現在の MAPPERKIT 1100 では、データ辞書およびマスタ・ファイルはシステムに完全に含まれており、システム間のデータの取り扱いについては考慮されていない。グローバルなデータ辞書を作成し、これを管理できれば、MAPPERKIT 1100 により作成したシステムの拡張、他システムとの結合も容易に行える。
- 3) 実データについての ADJUST 機能……現在の MAPPERKIT 1100 では、データ項目とそれに関連するソフトウェアの自動保守が行えるが、この機能を実際のデータにまで広げて更新を行いたい。

以上の機能は、レポート、項目、ランを管理するデータ辞書機能の存在を前提としており、これがなくては、開発は不可能に近い。データ辞書をシステム上に構築することによって、広範囲にわたる処理が効率化され、既存のデータ構成にとらわれないツールも開発可能になる。データ辞書を基にした自動登録機能と自動生成機能、自動保守機能は、システムが大規模化・複雑化するほどその効果を大きく表す。

7. おわりに

MAPPERKIT 1100 は、業務分析から保守までを支援するシステムとして開発された。普通ならば、かなり複雑な構造になり、専門家でなければ扱えないものになってしまうはずであるが、MAPPER 上に構築して、データ辞書を大いに活用することによってエンド・ユーザでもシステム開発が行えるものになり得た。レポートやトレーラ行の扱い方などをうまく処理しているほかデータ辞書の構成も単純である。

MAPPERKIT 1100 は、まだ改良すべき点が多く残されているが、今後のニーズに応えてゆけるものになると確信している。

-
- 参考文献 [1] 米田英一, ソフトウェア・エンジニアリングに関する調査, (社)日本電子工業振興協会, 1982, pp. 50~63.
- [2] 木村淳美他, データ・ディクショナリによるデータベース拡張の影響解析シミュレータの開発, 情報処理学会論文誌, Vol. 23, No. 6.
- [3] 椿正明, データ・ディクショナリ/ディレクトリ, 情報処理, Vol. 23, No. 10.
- [4] 角谷一郎他, システム・フローを自動生成する開発支援ソフト EAGLE 2, 日経コンピュータ, 1986.7.7.
- [5] 牧村信之他, 一貫したアプリケーション開発支援システム ADAMS, 富士通, Vol. 34, No. 6.

[6] 原田実, ソフトウェア生産性向上ツール(下)ーオートメーション方式編, 日経コンピュータ, 1984.3.19.

執筆者紹介 井上比佐乃 (Hisano Inoue)

昭和31年生, 54年津田塾大学学芸学部数学科卒業, 同年日本ユニパック(株)入社, 主として客先教育に従事, 現在, システム第2本部システム企画開発2部システム企画室に所属.



論文 中規模製造業における MAPPER 5 の活用事例

MAPPER 5—A Solution for a Medium Size Manufacturing Business

A. E. Shelton

要約 本稿は、クレオール食品 (Creole Foods, Inc.) における MAPPER 5* システムの使用事例の紹介である。クレオール食品では MAPPER システムを、

- 1) オーダ・エントリ・システム (価格決定の自動化, 値引価格, 販促キャンペーン用の価格決定機能も完備している)
- 2) 在庫管理システム (原材料と最終製品の両方を対象とする)
- 3) 生産管理システム (製造工程の管理, 原料・原材料から製品に至るまでの全工程のスケジューリングおよび報告を含む)
- 4) 請求管理システム
- 5) 購買管理システム
- 6) 財務システム (売掛金, 買掛金, 総勘定元帳システムを含む)

の実現のために利用している。

本稿では、システム全体のモジュラーな段階的実現を中心に、データベース設計、MAPPER ランの設計技術、効率に対する考慮について述べる。

Abstract This paper will explain how the MAPPER 5 System was used to automate Creole Foods, Inc. The MAPPER System was used to implement an Order Entry System, complete with automatic price calculations, discounts and special campaign pricing; an Inventory Control System, for both raw materials and finished products; a Production Control System, to control manufacturing, schedule and report the conversion of raw ingredients and materials into finished products; a Billing and Invoicing System; a Purchasing System; a Financial System which includes account receivable, account payable, and general ledger system. All were totally integrated in the MAPPER 5 System.

The modular, step-by-step implementation of the complete system will be presented, together with details on the database design, run design techniques, and overall efficiency considerations.

1. はじめに

クレオール食品は Louisiana 州 Opelousas に本社を置く中規模の製造業である。同社は多種多様な調味料と料理材料および著名な料理本を主力製品として販売しており、各種製品を組み合わせた贈答品も人気商品になっている。

同社はこれらの製品を製造するに当たり、原材料として食塩、黒こしょう、とうがらし、ガーリックなどを購入しなくてはならない。また各種サイズの容器、ビン、壺の類を購入し、粉末や液体の調味料を入れて製品化している。さらに製品に貼る各種ラベルも調達している。製品は箱単位にパッケージし、出荷する。料理本は印刷会社から発刊している。贈答品セットはパッケージ化の専門業者が請け負い、特別なデザインの体裁に仕上げている。

© 1986, USE Inc., Proceedings of USE Spring '86, pp. 1265~1301.

* 日本ユニバックの製品では MAPPER 6 に相当する。

同社の各種製品は、米国内の各地の卸問屋と小売り店から注文がある。さらに南部の州では多数の小さな食料品店からも注文を受けている。受注はケース単位で行い、レストランや食品加工会社からは大量注文がある。また料理本には不特定多数の購読者がいる。レストランからは、以上の三つの商品すべてについて注文のあることも珍しくない。たとえば食品製造用として調味料の大量注文と、店頭小売り用のケース単位注文、それに料理本の注文がある。

クレオール食品のシステム化要求事項は、比較的単純である。ここ数年の急成長によって、製造、出荷、および請求業務のシステム化が遅れていたためである。在庫関係の状況が的確に把握できていないために、原材料と製品在庫の過不足で大慌てをすることが再三、発生していた。そのため大量の原材料を見込みで発注することになり、資金繰りにも影響を及ぼすありさまであった。

2. MAPPER 5 システムの登場

クレオール食品では、さまざまなシステムを評価検討した結果、MAPPER 5 を購入することにした。MAPPER 5 による同社のシステム構成は、次のとおりである。

- ・MAPPER 5 システム, 8481 マスストレージ装置×1
- ・8481 マスストレージ拡張装置 (60 MB)
- ・モデル 35 システム・プリンタ
- ・Unisys 社製パソコン, Step ソフトウェア付き
- ・マルチプレクサ
- ・Epson 132 キャラクタ補助プリンタ
- ・Epson 80 キャラクタ・プリンタ

3. システムの概要

ハードウェア・システムに加え、次のソフトウェアを開発することになった(図1)。

- 1) 顧客管理システム……顧客名および住所、請求に関する情報、出荷情報、データ・エントリの内容などを検索する。
- 2) オーダ・エントリ・システム……受注処理、受注量/特別値引き/販売促進活動による価格決定を自動化する。パッキング指示書の発行、出荷指図書との発行と確認、在庫管理システムへの情報入力を行う。
- 3) 請求管理システム……請求書発行業務を行う。売掛金システムへ情報を提供し、総勘定元帳を更新する。
- 4) 生産管理システム……最終製品の製造計画と実績記録、品目別の製造原価を把握し、原材料および製品の在庫量を更新する。あわせて総勘定元帳も更新する。
- 5) 購買管理システム……発注指示書を発行し、発注書用紙を用いた発注書発行業務を行う。買掛金システムおよび総勘定元帳システムへ情報を提供する。
- 6) 売掛金システム……顧客別の売掛債権管理を行う。売掛金の支払情報を総勘定元帳システムへ提供する。
- 7) 買掛金システム……購入物品を受領した後の納品書と、取引会社への支払を管理する。取引業者の情報と発注仕様ファイルを管理する。

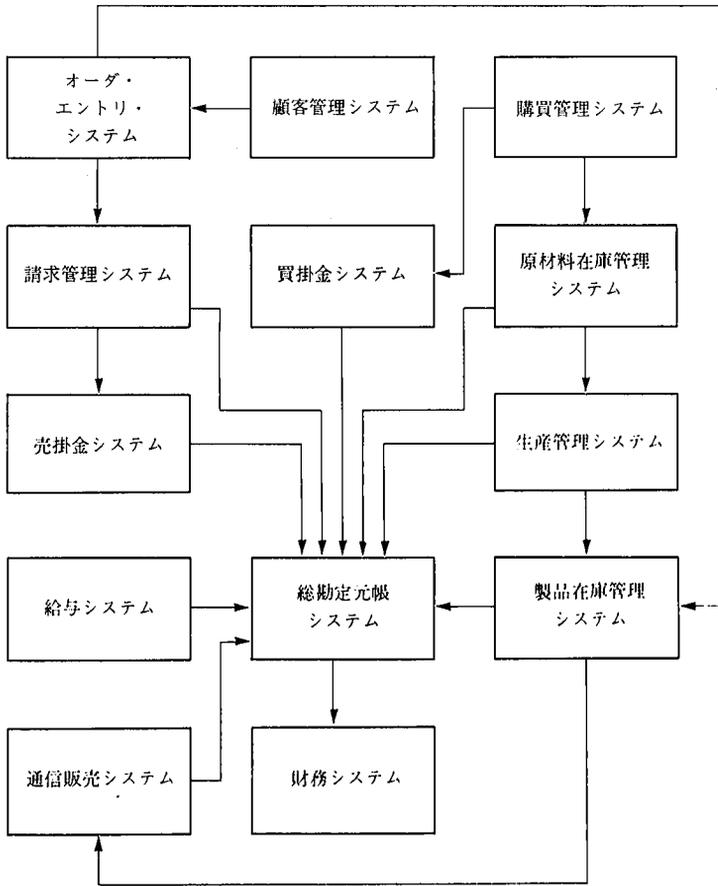


図 1 クレオール食品のシステム概要
Fig. 1 Creole foods, Inc. system overview

- 8) 財務システム……総勘定元帳の記帳処理と、各勘定の残高をリアルタイムで管理する。損益計算書、貸借対照表、およびその他の財務諸表を作成する。
- 9) 給与システム……従業員レコードを管理し、給与に充てる支払小切手を発行し、総勘定元帳を更新する。
- 10) 通信販売システム……通信販売用の顧客ファイルを管理する（約 60,000 件）。受注した商品のパッキング・リスト、クレジット・カードの写しなどを管理する。受注サマリを総勘定元帳へ転記する。

以上、説明した各システムは、すべて MAPPER システムで統一化している(図 1)。総勘定元帳に記入すべき取引は受注関係を除き、すべて自動的に仕分けと転記を行う。原材料および製品在庫はリアルタイムで管理する。在庫量は受注、出荷、製品の製造、購入品の荷受けにより変動する。

4. システム・デザイン上の考慮

4.1 MAPPER 5 システム

MAPPER 5 システムのリリース・レベル 1 R 3 は、MAPPER 1100 システムのサブ

セットである。機能性は、ほぼ MAPPER 1100 と同じだが、扱えるデータベースの大きさに基本的な相違がある。MAPPER 5 の制約条件は次の通りである。

最大モード数：30 モード (0/1 から 58/59 まで)

最大 RID 数：100 RID/タイプ+RID

最大行数：450 行/132 キャラクタ 700 行/80 キャラクタ

リザルトの最大サイズ：タイプ当たり最大行数のおよそ 4 倍まで。

4.2 MAPPER データベースのデザイン

MAPPER アプリケーションを開発するガイドラインの第 1 は、MAPPER のマニュアル・ファンクションだけを用いてシステムを開発することである。これが実現すると、後は MAPPER ランで簡単に自動化アプリケーションができて上がる。

第 2 のガイドラインは、用途別に MAPPER のタイプをデザインすることである。本システムでは次の用途を洗い出した。

- 1) 参照用タイプ……システムで利用する情報を収集するためのタイプで、更新の頻度は極めて少ないもの(たとえば顧客名と住所など)。
- 2) データ収集用タイプ(1 行更新)……トランザクション・データを集めるタイプで、オーダや購入伝票など、単票データを入力するためのもの。
- 3) データ収集タイプ(複数行更新)……トランザクションごとに複数レコードを作成する種類のデータを集めるタイプ。オーダに関連する詳細項目などがその例。
- 4) 更新データ・タイプ……更新の頻度は高いが、レコード数はあまり変化しないデータを格納するタイプ。更新は行を単位にして行う。たとえば在庫 RID などで、オーダ、出荷、製造などにより常に更新される。
- 5) 計算用タイプ……めったに更新することはないが、トランザクションの計算処理では頻繁に使用する。製品材料の数量やコストを求めるための“構成表”などがその一例である。
- 6) テーブル……時折り更新が必要だが、主として計算時の参照用に用いるタイプ。給与計算時の控除額表などがその例である。

以上の考え方を組み合わせると、“MATCH”や“TOT”、“CAL”などの命令の使い方を考えて空フィールドを持った計算用タイプがデザインできる。また在庫タイプなども考えられる。たとえば在庫品の使用済み量フィールドを設けて、これを“手持ち在庫量”や“予約在庫量”などで更新するような使い方ができる。

第 3 の考え方は、データベースを複数の RID へ展開する方法をどうするかについてである。MAPPER ランのデザイナーに依存するところが大きい。目的のシステムをどのように利用するか、またデータベースに格納するデータ件数と、その更新の頻度を明らかにすることが重要である。

最後に MAPPER のリレーショナル・データベース機能を活用したい。データの一部をインデックスに用いて、実際のデータ・レコードをポイントする手法は最少限に止めたい。

以上の基本的考え方を念頭におき、クレオール食品のシステム開発に取りかかった。

4.3 システムのサインオン

各システムは MAPPER 部門内で管理するため、ユーザが特定システムを簡単にア

クセスできる手段を提供することになった。各システムを管理する部門番号を覚えなくても使えるシステムを考えたのである。

この機能は次の方法で実現した。

- 1) ユーザは必ず部門-1 でサインオンする。
- 2) 最初に MAPPER ランの“GO”あるいは“MENU”を実行する(同じランを別々の名称でラン登録しておく。“GO”はどこでも利用できるようにしてある)。
- 3) スタートしたランはメニューを表示する。このメニューにはユーザが利用可能なすべてのシステムが示してあるので、タブでシステムを選定し XMIT を押すだけでよい。
- 4) スタートしたランがそのユーザ用に自動的にサインオンをする。
- 5) これでユーザは目的のシステムへサインオンしているので、業務を始められる。

ここで MAPPER システム固有のセキュリティ機能も活用している。ランによる自動サインオンは、そのユーザが特定の部門ナンバで登録済みでないかぎり、エラーになる。また、ユーザが他のシステムへアクセスしたい場合は、どの部門でサインオンしていてもラン“GO”を実行することで、最初のメニューへ入ることができる。したがってユーザは仕事の流れに応じて随時、任意のシステムへアクセスできる。各システムの部門ナンバを記憶しておく必要がない。MENU および GO のランは、図 2 のようになっている。

5. システム開発

5.1 顧客管理システム

最初に開発を開始したのは顧客管理システムであった。このシステムは同社の全顧客について、請求や出荷なども含めた一切の関連情報を管理する。

当時の顧客数はおよそ 300 社あり、最大 1200 社程度まで拡大することが見込まれている。顧客ナンバリング方式は設定しておらず、今回初めて採用した。顧客レコードごとに、複数出荷先アドレス(必要時)、指定運送会社、値引き率、その他の事項を設定した。

顧客データベースはモード 10 を使用した。タイプ B に基本顧客データを格納し、タイプ C に出荷先アドレスを格納した。タイプ I には MAPPER ランを入れた。タイプ B と C は 132 キャラクタ長で生成し、顧客ナンバはコモン・フィールドとした。顧客ナンバはデータ入力時に新たに発番することにした。RID の使い方は、A から始まる顧客名から B、C の順に RID 1, 2, 3, …と配置し、RID 1 に最初に入力した顧客名からナンバリングを開始した。したがって顧客ナンバは 5 桁となる。最初の 2 桁は 01 から 26 までで RID ナンバに対応し、残り 3 桁がその RID 内の順番を表す。こうして設定したモード 10 は図 3 のとおりである。

MAPPER 5 システムでは @LDV, N の命令が使用できないので、顧客名の最初の文字から RID ナンバ 01~26 を生成する方法を採用した(図 4 のラン参照)。

LOCATE 命令を用いると、MAPPER は指定した英字のあるカラム位置を変数に返す。この値は第 2 カラムのとき 1 となり、指定英字が探せないときは 0 となる。この方法により RID ナンバ算出のためのロジックと入出力の時間を最小化できる。

```

.DATE 09 APR 86 15:50:19 RID      50    31 JAN 86 TONY
.  RUN NAME: MENU/GO              BY:
*-----*
@BRK CHG V1H1 SOE# .

                CREOLE FOODS OF OPELOUSAS, INC
                ----- SYSTEM ENTRY MENU -----
                V1 , CUSTOMER INFORMATION SYSTEM
                V1 , INVENTORY / PRODUCTION SYSTEM
                V1 , ORDER ENTRY / INVOICING
                V1 , PURCHASING SYSTEM
                V1 , EQUIPMENT INVENTORY
                V1 , GENERAL LEDGER / FINANCIAL SYSTEM
                V1 , ACCOUNTS PAYABLE / RECEIVABLE
                V1 , SALES INFORMATION REPORTS
                V1 , (FUTURE EXPANSION)

@BRK OUT,2,I,-0,2,23,1,1,Y,,,P .
@CHG V512 SOEV# .
@IF V5 = 6 LDV V3I1=3 GTO 10 .
@IF V5 = 8 LDV V3I1=4 GTO 10 .
@IF V5 = 10 LDV V3I1=5 GTO 10 .
@IF V5 = 12 LDV V3I1=6 GTO 10 .
@IF V5 = 14 LDV V3I1=9 GTO 10 .
@IF V5 = 16 LDV V3I1=8 GTO 10 .
@IF V5 = 18 LDV V3I2=10 GTO 10 .
@IF V5 = 20 LDV V3I2=12 GTO 10 .
@ . IF V5 = 22 LDV V3I2=13 GTO 10 .
@10:CHG V1H1 SOE# CHG V2H11 USER# LDV,P V2=V2 .
V1J2,V3
@BRK OUT.2,I,-0,2,1,0,1,Y,,,Y .

```

図 2 ラン“MENU”/“GO”のリスト
Fig. 2 Listing for run “MENU”/“GO”

```

----- FILE CABINET - MODE 10/11 DRAWER - TYPE -----
.F.      FORM TYPE      .FORM.F.      FORM TYPE      .FORM.
.D.      DESCRIPTION    .TYPE.D.     DESCRIPTION    .TYPE.
-----
B CUSTOMER LIST          0122 F SALESMAN/BROKER LIST  0132
C SHIPPING ADDRESSES    0124 G OPEN          0134
D PRICING INDEX         0126 H OPEN          0136
E OPEN                  0130 I RUN FUCTIONS  0140

```

図 3 モード 10 のタイプ構成
Fig. 3 Type layout for mode 10

```
.DATE 18 FEB 86 10:55:56 RID      2      31 JAN 86 TONY
.RUN NAME: CUST-INPUT              BY:
*=====
ABCDEF GHIJKL MNOPQRST UVWXYZ
@BRK
@CHG INVAR$ V10H5,V11I3,V12I3,V13I4,V14S30,V15S26,V16S26,V17H16,V18H2,\
V19H9,V20H1,V21H1,V22H1,V23I2,V99H1
@
@OUT,10,I,7,4,23,1,1,Y,,,P
@IF V99 = X RUN MENU
@LOC,10,I,2,4 AFT$ 2-27 V14(1-1) V5I2 . GET RID NUMBER FROM FIRST LETTER
@LOK,10,B,V5 RDL,10,B,V5,2 2-3 V6I3 LZR,10,B,V5 V7I3 CHG V60I3 V6 +1
@IF V60 > V7 LN+,10,B,V5,V7,20 . ADD MORE BLANK LINES IF RID IS FILLED
@CHG V9I3 V6 -5 LDV,ZR V30H2=V5,V31H3=V9,V40H5=V30V31
@WRL,10,B,V5,V6 2-5,8-30,39-26,66-16,83-2,86-9,96-5,102-13,120-2,\
123-1,125-1,127-1,129-1 ' ',V40,V14,V15,V17,V18,V19,V10,'('V11')'V12'-'V13,\
V23,V22,'1',V20,V21 CHG V6 V6 +1
@DEF V2I1 V16 IF V2 = 0 GTO 20
@WRL,10,B,V5,V6 2-5,39-26 *,V40,V16 CHG V6 V6 +1
@20:WRL,10,B,V5,2 2-3 .,V6 ULK
@IF V23 > 0 GTO 50
@IF V99 = L RUN MENU
@RUN CUST-INPUT
@50:LDV V150I2=V23,V151I2=0
@55:LDV V153H4='S00'V151(2-1)
      CUSTOMER SHIPPING INFORMATION DATA ENTRY

2      Customer No: V40,2 Telephone No:( , ) , -
2      Customer Name: V14,
2      Address:
2      :
2      City: ,2 State: ,2 Zip Code:
2      Attention:
2      Address Code: V153,

      Transmit from here> ,2(Use "X" to EXIT,"L" if last entry)
@CHG INVAR$ V10H5,V11I3,V12I3,V13I4,V14S30,V15S26,V16S26,V17H16,V18H2,\
V19H9,V120S17,V21H4,V99H1
@
@BRK OUT,10,I,-0,2,23,1,1,Y,,,P
@IF V99 = X RUN MENU
@LOC,10,I,2,4 AFT$ 2-27 V14(1-1) V5I2 . GET RID NUMBER FROM FIRST LETTER
@LOK,10,C,V5 RDL,10,C,V5,2 2-3 V6I3 LZR,10,C,V5 V7I3 CHG V60I3 V6 +1
@IF V60 > V7 LN+,10,C,V5,V7,20 . ADD MORE BLANK LINES IF RID IS FILLED
@
@WRL,10,C,V5,V6 2-5,8-30,39-26,66-16,83-2,86-9,96-13,110-17,128-4 \
',V10,V14,V15,V17,V18,V19,'('V11')'V12'-'V13,V120,V21 CHG V6 V6 +1
@DEF V2I1 V16 IF V2 = 0 GTO 20
@WRL,10,C,V5,V6 2-5,39-26 *,V10,V16 CHG V6 V6 +1
@20:WRL,10,C,V5,2 2-3 .,V6 ULK
@CHG V15I V15I +1 IF V15I < V150 GTO 55
@
@IF V99 = L RUN MENU
@RUN CUST-INPUT
```

図 4 顧客登録ランのリスト

Fig. 4 Customer input run listing

新たに顧客レコードを入力するための空白行を探し出すため、別の方法も考案した。タイプ B と C にある RID(図 5)は、ピリオド・ライン 2 にライン・カウンタを付けて生成してある。このライン・カウンタは最初 6 になっている。これはヘッダの直後にある最初のタブ行の番号を意味する。タイプ B と C にある 26 個の RID のライン・カウンタ値を合計することで、更新すべき RID を探すロジックが不要になった(図 4)。

例示したランでは RID を最初にロックして、複数のデータ・エントリ・ターミナルから複数回更新するのを防いでいる。次にライン・カウンタを変数 V 6 I 3 ヘリードし、LZR 命令で RID の最終ライン番号を得て V 7 I 3 へ入れる。この二つの変数値を比較して、その RID に空白行があるかを調べる。空白行がなければ RID の終わりに 20 行

を追加して、“END REPORT”を越えてRIDを更新することのないようにしている。新たにレコードをRIDに書き込んだあと、ライン・カウントに1を加算してライン2へ書き戻す。最後にロックを解く。

上記の方法はクレオール食品のシステムのうち、RIDに1行加えてデータを更新するトランザクション処理が必要なものについては、全面的に採用した。

顧客管理システムが完成した後、同社では顧客レコードの入力を開始した。入力データの検索、ソート、確認訂正はマニュアル・ファンクションを用いて行った(図5)。

5.2 オーダ・エントリ・システム

次に開発したのが、オーダ・エントリ・システムである。このシステムの基本的な要求条件は次のとおりである。

- 1) 請求書番号を自動発番する。
- 2) 価格は顧客種別を基に適正な価格表を選定して自動的に算出する。
- 3) 量販割引はオーダの全量を基にして算出する。
- 4) 無償製品が設定できること(10個買えば一つおまけなどへの対応)。
- 5) 特別キャンペーン用割引へ対応できること。
- 6) 顧客ナンバまたは顧客名を用いてデータ・エントリできること。
- 7) 出荷先アドレスは複数持てること。
- 8) オーダを受け付けたら、出荷準備として“パッキング・リスト”を作成する。
- 9) オーダにより在庫更新をすること。“予約在庫数”フィールドを設けること。

デザイン当時のオーダ数は季節変動が大きいが、毎日平均して50オーダ程度が見込まれた。

オーダ・エントリ・システムはモード14にあり、図6に示すようなタイプを生成してある。

図6のとおり、モード14は各タイプを次の用途に設定してある。

タイプB：オーダを入力するごとに1レコード作成する。格納するRIDは、8桁の請求書ナンバの最後の2桁で決める。請求書ナンバが“00”のときはRID100に格納する。レコードのレイアウトは図7のようになる。

タイプC：オーダごとに複数レコードを作成する。各レコードにはオーダごとの製品詳細が入る。基本は、通常価格レコード、割引価格レコード、無償製品レコードの3種類である。タイプBと同じく、レコードは請求書ナンバ順に各RIDへ格納する。レコード・レイアウトは図7を参照のこと。

タイプD：このタイプうちのRIDは三つのセットにまとめてある。各セット内の最初のRIDには価格表と量販割引価格がある。2番目のRIDは特別キャンペーン用の割引価格表がある。3番目のRIDには出荷日を保存する。第1のRIDセットはケース単位のオーダ用、第2のRIDセットは大量注文用、第3のRIDセットは贈答用と料理本のオーダ用である。その他のRIDは価格表の拡張用使用する。

タイプF：処理可能な請求書のリストを保存する。この請求書リストは、タイプBから受付済みで出荷済みオーダをすべて探し出して作成する。“請求日”(INVOICE DATE)のないレコードが対象となる。

DATE	TIME	RID	MODE	CUSTOMER MASTER LIST	CITY	ST.	ZIP CODE	SLSM NO.	TELEPHONE NUMBER	SH. P. T. S. T.		
*CUST	*NO	*NAME	*ADDRESS	*CITY	*ST.	*ZIP	*SLSM	*NO.	*NUMBER	*FL. T. H. V. Y.		
04 FEB 86	16:00:05	1	04 DEC 85 MONA									
01001	ALBERTSONS SOUTHCO	P. O. BOX 7100	ORLANDO	FL	32854	H007	(504)275-8116	4	3	1	4	2
01002	AWARD FOODS	860 WANDALIA ST.	ST. PAUL	MN	55114	E003	(214)298-2957	1	3	1	1	1
01003	AQUA FRESH SEAFOOD	10823 MAC ARTHUR	OAKLAND	CA	94605	H008	(415)568-3382	1	3	1	5	2
01004	ASSOCIATED GROCERS	P. O. BOX 3763	SEATTLE	WA	98124	E006	(206)762-2100	1	3	1	3	1
01005	AVOYELLES WHSE. GROCER	P. O. BOX 5725	ALEXANDRIA	LA	71307	E001	(318)442-8865	0	3	1	6	1
01006	AFFILIATED OF FLORIDA	1102 N. 28TH ST.	TAMPA	FL	33630	C001	(813)248-5781	0	3	1	2	1
01007	AFFILIATED FOODS	P. O. BOX 91910	LAFAYETTE	LA	70501	()	0-0	0	2	1	6	1
01008	ARROW FOODS	1405 JEFFERSON HWY.	NEW ORLEANS	LA	70121	()	0-0	0	0	1	1	6
01009	ASSOCIATED GROCERS	P. O. BOX 1748	BATON ROUGE	LA	70821	()	0-0	0	0	3	1	6
01010	ASTOR FOODS	2065 LIDDEL DRIVE, N. E.	ATLANTA	GA	30324	()	0-0	0	0	1	1	1

DATE	TIME	RID	MODE	CUSTOMER MASTER SHIPPING ADDRESSES	CITY	ST.	ZIP CODE	TELEPHONE NUMBER	ATTENTION	SHIP. CODE.
*CUST	*NO	*NAME	*ADDRESS	*CITY	*ST.	*ZIP	*SLSM	*NO.	*NUMBER	*CODE
06 FEB 86	10:58:05	1	03 DEC 85 TONY							
01001	ALBERTSONS SOUTHCO	11321 FLORIDA BLVD.	BATON ROUGE	LA		()	0-0	0		S001
01001	ALBERTSONS SOUTHCO	7515 PERKINS RD.	BATON ROUGE	LA		()	0-0	0		S002
01001	ALBERTSONS SOUTHCO	9650 AIRLINE HWY.	BATON ROUGE	LA		()	0-0	0		S003
01002	AWARD FOODS	1711 WEST THOMAS ST.	HAMMOND	LA		()	0-0	0		S004
01003	AQUA FRESH SEAFOOD	4202 DAN MORTON DRIVE	DALLAS	TX	75236	()	0-0	0		S001
01004	ASSOCIATED GROCERS	10823 MAC ARTHUR	OAKLAND	CA	94605	(415)568-3382				S001
		7890 S. 188TH STREET	KENT	WA	98032	(206)762-2100				S001

図 5 モード 10, タイプ B および C の RID の配置. タイプ B は顧客マスタ・リストで, タイプ C は各顧客のアドレスであり, 出荷先“SHIP-TO”アドレスと請求先“BILL-TO”アドレスは別々になっている.

Fig. 5 RID layout for types B and C in mode 10.

```

----- FILE CABINET - MODE 14/15 DRAWER - TYPE -----
.F.      FORM TYPE      .FORM.F.      FORM TYPE      .FORM.
.D.      DESCRIPTION    .TYPE.D.      DESCRIPTION    .TYPE.
-----
B ORDER SUMMARY                0162 F INVOICE LIST                0172
C ORDER DETAIL                 0164 G INVOICE HOLDING AREA        0174
D OPEN                         0166 H OPEN                        0176
E PRODUCT & PRICE TABLES     0170 I RUN FUNCTIONS                0200

```

図 6 モード 14 の TOC

Fig. 6 Table of contents for mode 14

```

.DATE 09 APR 86 09:37:05 RID 1 06 FEB 86 DIANA
. 7 CUSTOMER ORDERS - SUMMARY INFORMATION B0162
*INVOICE .INVOICE . ORDER . ORDER .CUST .SHIP. TOTAL .S. FREIGHT .
* NO . DATE . NO . DATE . NO .CODE. INVOICE .H. CHARGES .
-----
4301 30401008 04/09/86 07003 0000 526.50
4302 04/09/86 20003 0000 1414.74
4303 04/09/86 20003 0000 534.38
4304 04/09/86 120404 04/09/86 06003 S001 1125.00 .00 B60409
..... END REPORT .....

```

```

.DATE 09:37:03 RID 1 09 APR 86 DIANA
. ORDER DETAIL C0164
*INVOICE . ORD . SHP . UPC . UNIT . TOTAL .
* NO . QTY . QTY . NO . DESCRIPTION .UN. PRICE . PRICE .
-----
4301 40 00001 CREOLE SEASONING, 8 OZ. EA 9.85 394.00
4301 10 00006 ROUX & GRAVY MIX, 10 OZ. EA 13.25 132.50
4302 80 00001 CREOLE SEASONING, 8 OZ. EA 8.75 700.00
4302 8 00001 CREOLE SEASONING, 8 OZ. EA .00 .00
4302 30 00005 CREOLE SEASONING, 17 OZ. EA 16.00 480.00
4302 3 00005 CREOLE SEASONING, 17 OZ. EA .00 .00
4302 10 00006 ROUX & GRAVY MIX, 10 OZ. EA 11.72 117.20
4302 1 00006 ROUX & GRAVY MIX, 10 OZ. EA .00 .00
4302 10 00012 BARBECUE SAUCE, 18 OZ. EA 10.20 102.00
4302 1 00012 BARBECUE SAUCE, 18 OZ. EA .00 .00
4302 10 00016 CRAB BOIL, 9 OZ. EA 9.00 90.00
4302 1 00016 CRAB BOIL, 9 OZ. EA .00 .00
4302 PICK-UP ALLOWANCE -74.46
4303 50 00003 CAJUN COUNTRY COOKBOOK EA 4.50 225.00
4303 25 00017 MICROWAVE COUNTRY COOKBOOK EA 4.50 112.50
4303 50 00040 LA ORIG CREOLE SEAFOOD RECP EA 4.50 225.00
4303 PICK-UP ALLOWANCE -28.13
4304 80 80 00001 CREOLE SEASONING, 8 OZ. EA 9.85 788.00
4304 15 15 00005 CREOLE SEASONING, 17 OZ. EA 18.05 270.75
4304 5 5 00006 ROUX & GRAVY MIX, 10 OZ. EA 13.25 66.25
..... END REPORT .....

```

図 7 モード 14, タイプ B および C. タイプ B は各オーダーのマスタ・レコードが入る。タイプ C はタイプ B にある各オーダーごとの全詳細レコードが入る。この二つのタイプは請求書ナンバーでリンクしてある。

Fig. 7 Layout for types B and C in mode 14.

タイプ G: フリー・フォーム, 90 キャラクタ長のタイプで, 発行済み請求書を保存する。請求書を発行しそのチェックが終わったら, このタイプから定型印刷フォームにプリント出力する。

タイプ D (モード 10): 遠隔地の顧客数が拡大するにつれて, 価格設定の複雑化が一層進むことが明らかとなった。そこでモード 10 に新たなタイプを生成し, 価格管理インデックスとして用いた。このタイプの RID は, 価格設定 RID と “Customer Type” (顧客区分), “Price Table Indicator” (価格表インジ

ータ：モード 10, タイプ B のレコードより), およびオーダーの合計数の各フィールドについて, カラム位置を決めたものである(図 8 a).

オーダー・エントリ・システムは次のように機能する.

- 1) MENU から“ENTER CUSTOMER ORDER”を選択する.
- 2) 最初の入力スクリーンが現れる. そこから“CUSTOMER NUMBER”(顧客ナンバ) または“CUSTOMER NAME”(顧客名), 顧客のオーダー・ナンバ, “CAMPAIGN NUMBER”(必要時) を入力する. 名前の最初の部分だけを入力するだけでもよい.
その結果, 複数エントリが見つかった時は全部, スクリーン上に表示するので, オペレータは所定の名前にタブキーでカーソルを合わせ, XMIT キーを押す. あとはランが顧客ナンバを取り上げて処理が進む.
- 3) ランはモード 10, タイプ B 内から, 顧客ナンバの最初の 2 桁をキーとして顧客レコードを探す. 顧客名とアドレス, 出荷先アドレスの数, 価格表インジケータ, 顧客種別コード, “SIP-VIA”(配送ルート) コードをそれぞれ変数に入れる.
- 4) 出荷先アドレス・コードが 0 の時は“SHIP-TO”アドレスと“BILL-TO”アドレスは同じになる. 1 の時はタイプ C 内の対応する RID を探して(FIND), 各種フィールドを変数へ格納する. 出荷先アドレス・コードが 1 より大きい時は, タイプ C から複数アドレスを探して(SEARCH)スクリーンに表示し, ステップ 2 と同様にオペレータに選択してもらう.
- 5) “BILL-TO”と“SHIP-TO”の各情報をスクリーンに確認のため表示する. オペレータはその内容が正しいことを確認した後, 次のステップへ進み, オーダーの種別を CASES, BULK, MISC のうちから選定する. 一つでも全部でもよい.
- 6) ステップ 5 で選択した種別により, 最大三つまでのスクリーンが表示される. 各スクリーンでは, 各製品ごとに注文数を入力するだけでよい. 左カラムは通常価格を示しており, 右カラムは無償製品にだけ用いる.
- 7) 各入力スクリーンにオーダーを入力したら, 次に“@CHG INPUT\$”命令でデータを得る. 出力エリアに RID を作成する. 価格インデックス RID(モード 10, タイプ D) を調べて使用する価格表を決める. 作成した RID でトータライズ命令(@TOT)を用いて, 受注量の合計を求める. 空白ランを探し出して削除し(SEARCH), 製品コードを比較して製品価格と説明を適当な表から移動する(MATCH). 同じ方法を繰り返して無償製品と特別割引も処理する.
- 8) 全部の入力を終了すると, “ORDER SUMMARY”レコードをモード 14, タイプ B 内の請求書ナンバの下 2 桁に対応する RID へ書き込む(@WRL). “ORDER DETAIL”レコードをモード 14, タイプ C の対応する RID へ追加する(@ADD と@REP).
- 9) 次に, “Packing List”を作成してプリントへ出力する. このリストは内容をチェックした後, 倉庫へ送付する.
- 10) 最後にオーダーの製品コードをキーとして小計を求め, 最終製品在庫を更新する(予約済み数量フィールド).

オーダーが処理されたことを示すパッケージ・リストが戻されたら, オペレータは

“VERIFY ORDER SHIPPED”の処理へ進む。このステップは次のようになる。

- 1) 最初の入力スクリーンで請求書ナンバを入力する。
- 2) モード 14, タイプ B と C にあるレコードをアクセスして“ORDER COMPLETE?”に Y または N を入力し, 使用するパレット数, および贈答品と料理本のオーダがあれば, その梱包用ボックス数を指示するスクリーンを作成する。また郵送する時は, その費用をここで入力する(図 9)。
- 3) オペレータが XMIT キーを押すと梱包指図書をプリンタから出力する(図 10)。
- 4) 最終製品在庫を更新し, “手持ち在庫量”と“予約在庫量”を出荷数量だけ減らす(@MAU, @CAL, @UPD)(図 8)。

その他, オーダ照会, オーダ取消, パッケージング・リスト再作成用のランをそれぞれ開発した。

```
.DATE 10 APR 86 07:57:16 RID 11 03 DEC 85 TONY
.RUN NAME: SUBR TO UPDATE INVENTORY I02
=====
01 . LABEL TO UPDATE QTY-ON-ORDER FIELD.
@FND,10,D,1,6,98 ' 2-1,4-1 ,V93,V91 ,V713 .
@RDL,10,D,1,V7 28-2 V9712 .
04:MAU,14,C,-3,12,F,V97 DM 11-5,23-5 ,A,1 2-5,84-5 ,1,A . MATCH ORDERED QTY
@CAL,12,F,-0 R1 45-8,54-8,81-8 ,A,B,C A=A+C;B=B-C . ADJUST QTY
@UPD . FINISHED
@ESR .
02 . LABEL TO UPDATE QTY-ON-HAND FIELD.
@FND,10,D,1,6,98 ' 2-1,4-1 ,V93,V95 ,V713 .
@RDL,10,D,1,V7 28-2 V9712 .
05:MAU,14,C,-1,12,F,V97 DM 11-5,23-5 ,A,1 2-5,84-5 ,1,A . MATCH ORDERED QTY
@CAL,12,F,-0 R1 36-8,45-8,81-8 ,A,B,C A=A-C;B=B-C . ADJUST QTY
@UPD . FINISHED
@ESR .
03 . LABEL TO UPDATE QTY-ON-ORDER FIELD. **ORDER DELETE**
@MAU,14,C,-4,12,F,1 DM 11-5,23-5 ,A,1 2-5,84-5 ,1,A . MATCH ORDERED QTY
@CAL,12,F,-0 R1 45-8,54-8,81-8 ,A,B,C A=A-C;B=B+C . BACK-OUT ORDER
@UPD . FINISHED
@ESR .
..... END REPORT .....
```

図 8 在庫更新用ランのサブルーチン

Fig. 8 Run subroutine used to update inventory

```
.DATE 07 APR 86 RID 1 TONY
. PRICE TABLE INDEX - CASE ORDERS
*T.P. QUANTITY . E.C. .INV. .
*Y.T. FROM. TD .RID.N.CC.RID. .
*.....*.....*.....*.....*.....*.....*
1 2 1 99999 1 4 63 1
1 3 1 200 1 2 45 1
1 3 201 750 1 3 54 1
1 3 751 99999 1 4 63 1
1 6 1 200 11 1 36 1
1 6 201 750 11 2 45 1
1 6 751 99999 11 3 54 1
2 3 1 99999 1 1 36 1
3 3 1 99999 1 1 36 1
2 6 1 99999 1 1 36 1
A 4 1 99999 1 1 36 10
B 4 1 99999 1 1 36 11
C 4 1 99999 1 1 36 12
D 4 1 99999 1 1 36 20
5 5 1 99999 1 5 72 1
..... END REPORT .....
```

図 8a オーダの価格決め用価格表インデックス

Fig. 8a Price table index used in order pricing

```

CUSTOMER ORDER SHIPPING VERIFICATION

CUSTOMER ORDER COMPLETE? Y ("Y" OR "N")  NUMBER OF PALLETS? 0
NUMBER OF PACKAGES FOR- COOKBOOKS: 4  GIFT SETS:

*-----*
*          ENTER U.P.S. FREIGHT HERE>  █
*-----*
*  CUSTOMER NAME:TROY SAVAGE
*  P.O. NUMBER:
*  INVOICE NUMBER: 4303          ORDER DATE: 04/09/86
*                               INVOICE DATE:
*-----*
INVOICE . ORD . SHP . UPC .          . UNIT . TOTAL
NO . QTY . QTY . NO .          DESCRIPTION .UN. PRICE . PRICE
=====
4303  50   50   00003 CAJUN COUNTRY COOKBOOK EA   4.50  225.00
4303  25   25   00017 MICROWAVE COUNTRY COOKBOOK EA   4.50  112.50
4303  50   50   00040 LA DRIG CREOLE SEAFOOD RECP EA   4.50  225.00
=====
    
```

図 9 出荷指示の入力スクリーン
Fig.9 Shipping order input screen

```

*****
*
*          P A C K I N G   L I S T
*
*****
*  BILL TO:          ***      SHIP TO:          *
*  -----          ***      -----          *
*  CHARLES MATTHIS, JR.      ***      CHARLES MATTHIS, JR.      *
*  1616 HARING ROAD          ***      1616 HARING ROAD          *
*  METAIRIE, LA 70001        ***      METAIRIE, LA 70001        *
*                               ***      *
*                               ***      *
*                               ***      *
*                               ***      SHIP VIA: SAIA MOTOR FREIGHT *
*                               ***      *
*                               ***      *
*****
*  INVOICE NO: 4309  P.O. NUMBER:          ORDER DATE: 04/09/86 *
*-----*
*
*  QTY   QTY
*  ORD   SHP
*  ----  ----
*  8     ----  CREOLE SEASONING, 8 OZ.
*  6     ----  CREOLE SEASONING, 17 OZ.
*  5     ----  SALT FREE SEASONING, 8 OZ.
*  4     ----  LITE SALT SEASONING, 8 OZ.
*  6     ----  ROUX & GRAVY MIX, 10 OZ.
*  30    ----  CRAB BOIL -POLY BAGS, 16 OZ
*  3     ----  GUMBO FILE, 1 1/4 OZ.
*  10    ----  SEAFOOD SAUCE, 10 OZ.
*  7     ----  BARBECUE SAUCE, 18 OZ.
*  2     ----  HOT SAUCE, 6 OZ.
*  4     ----  LIQUID CRAB BOIL, 4 OZ.
*  1     ----  LIQUID CRAB BOIL, 8 OZ.
*  120   ----  THE CRAWFISH COOKBOOK
*  120   ----  BASICS OF CREOLE COOKING
*
*****
*  ORDER APPROVED:  _____
*
*  ORDER FILLED BY:  _____
*
*  ENTER TOTAL NUMBER OF PALLETS IN SHIPMENT...[ ]
*
*  ENTER NUMBER OF PACKAGES FOR COOKBOOKS.....[ ]
*
*  ENTER NUMBER OF PACKAGES FOR GIFT SETS.....[ ]
*
*****
    
```

図 10 オーダ・エントリ・ランで作成したパッキング・リスト。オーダを入力し出荷用に作成した後、このリストはオーダ処理を終了するための入力ドキュメントとなる。

Fig. 10 Packing list generated by order entry run

5.3 購買管理システム

このシステムはモード 16 にあり、図 11 のようにタイプを使用する。

```

----- FILE CABINET - MODE 16/17 DRAWER - TYPE -----
.F.      FORM TYPE      .FORM.F.      FORM TYPE      .FORM.
.D.      DESCRIPTION    .TYPE.D.      DESCRIPTION    .TYPE.
-----
B PURCHASE ORDER SUMMARY      0202 F VENDOR MASTER FILES      0212
C PURCHASE ORDER DETAIL       0204 G P.O. ITEM SPECIFICATIONS    0214
D P.O. RECEIVING DETAIL       0206 H P.O. PRINT HOLDING AREA     0216
E OPEN                          0210 I RUN FUNCTIONS                0220

```

図 11 モード 16 の TOC

Fig. 11 Table of contents for mode 16

購買管理システムの主な機能は次の通りである。

- 1) 発注書の発行……製品材料およびパッケージ用/出荷用品など原材料を購入するための発注書を作成する。
- 2) 発注品の受入れ……原材料を受け取り、原料在庫量、買掛金、総勘定元帳を更新する。料理本を受け入れた時も同様の処理をして、最終製品在庫を更新する。
- 3) 仕様書……発注書に添付する仕様書の管理と更新を行う。
- 4) 購入記録……レビューおよび統計分析用に、購買品目の明細を記録する（リードタイム、価格変動など）。

オーダ・エントリ・システムと同様に、タイプ B と C をデザインした。購入発注書のマスタ・レコードをタイプ B に、また購入発注書の詳細レコードをタイプ C にそれぞれ割り当てた。

購入業者についての情報、たとえば業者名、住所と製品仕様は、それぞれタイプ F と G に格納する。購入発注書は、専用ランを用いてシステムへ入力する。図 12 に入力用スクリーンを示す。入力時のエラーを最少限に止めるため、入力手順のステップ別に適切な指示をスクリーン上に表示するなどの工夫をしている。

図 12 のスクリーン例で、ユーザは“VENDOR CODE”(業者コード) までの情報をすべて入力する。次にランがそのコードと“SHIP-TO”コードを確認してスクリーン

```

PURCHASE ORDER GENERATION
P.O. NUMBER: 4598
DATE ORDERED: 24 APR 86      DATE WANTED: 30 APR 86
TERMS:                      F.O.B.:
SHIP VIA: BEST WAY          SHIP TO: CF001
VENDOR CODE: B0005
VENDOR NAME & ADDRESS: BALTIMORE SPICE COMPANY
                        208 E. DURHAM
                        SLIDELL      LA
SHIP TO NAME & ADDRESS: CREOLE FOODS, INC.
                        533 LOMBARD ST.
                        OPELOUSAS    LA
-----
ITEM CODE: R1001
ITEM DESCRIPTION: RED PEPPER 35000 HU
                  BALTIMORE SP, #RED 2535, 50#CONT--.96/LB
-----
QTY ORDERED: 200      UNIT PRICE: 21.88      PER: 60      UOM: SACKS
INVENTORY MULTIPLIER: 60      TRANSMIT>      ("X" TO RE-ENTER MATL CODE)

```

図 12 購入発注書の入力スクリーン例

Fig. 12 Example screen for P.O. input

へ表示する。表示内容が正しければ、次に発注する製品コードを入力する。ここでもその製品の説明と仕様をスクリーン上に表示する。最後に残りの情報を入力する。“INVENTORY MULTIPLIER:”は、たとえばオーダ数量がケース単位であったら、それを適当な在庫単位に変換するファクタとして用いる。

こうして発注書を作成したら、連続用紙にプリント出力する。出力が終わると各マスタは発行済みのフラグが立てられる。マスタ・レコードはそのままシステム内に残されて、タイプ B と C の両データの分析に用いている。

発注した物品が到着すると、受入数量と返品量(必要時)をタイプ D に記録し、在庫量を適宜更新する。最後に総勘定元帳の勘定を更新して、買掛金発生エントリを作成、買掛金システムで後日行う支払いに備える。

5.4 在庫管理と生産管理システム

このシステムはモード 12 にあり、図 13 のようにタイプを配置している。

```

----- FILE CABINET - MODE 12/13 DRAWER - TYPE -----
.F.      FORM TYPE      .FORM.F.    FORM TYPE    .FORM.
.D.      DESCRIPTION    .TYPE.D.    DESCRIPTION   .TYPE.
-----
B RAW MATERIALS INVENTORY  0142 F FINISHED GOODS INVENTORY  0152
C PRODUCTION RUN RECIPE FILE  0144 G LABOR REPORT  0154
D PRODUCTION RUN PACKAGING  0146 H PROD RUN RID INDEX  0156
E PRODUCTION TRACKING FILE  0150 I RUN FUNCTIONS  0160
    
```

図 13 モード 12, 生産管理および在庫管理システムのタイプ・レイアウト

Fig. 13 Mode layout for mode 12, inventory and production control

モード 12 の各タイプは次のように使用している。

タイプ B：メイン RID が一つと、バックアップ用 RID を一つ使用している。正常在庫は RID 1 B にあり、2B を万一の入力エラーに備えたバックアップとしている。原材料あるいは製品材料は、各品目ごとに 5 キャラクタのコードと 27 キャラクタ長の内容説明、手持ち在庫数量、単価、最少在庫量、計算用空白フィールド、単位、総勘定元帳の番号がそれぞれ割り当ててある。

タイプ F：タイプ B と同じ形式であるが、2 番目の数量フィールドが予約量フィールドになっている点が異なる。その他の RID では配送車両に積載した在庫、アラバマの第 2 倉庫の在庫、通信販売の在庫をそれぞれ把握している。在庫管理システムの一部で、主要在庫 RID とその他の在庫 RID 間で製品移動ができるようになっている。図 14 に最終製品在庫 RID の例を示す。

タイプ C & D：最終製品の生産を管理する目的で、タイプ C と D を各製品の製法を持つようにデザインした。また各種製品に必要なパッケージング情報もこのタイプに持つ。タイプ C は原材料を混合して求める総量のバッチ数によって決まる。タイプ D は製造する製品の箱数によって決まる。タイプ C と D は、製品の“原材料構成表”RID と考えてよい。

タイプ E：各本番ランのサマリ情報、たとえば製造コスト合計、平均コストなどを保存するタイプ。保存 RID は各月ごとに RID 1 から 12 を使用する。

タイプ H：タイプ C と D のインデックス。製造品目の製品コードに基づいて、タイ

. DATE		13:36:54 RID 1 14 APR 86 TONY			FINISHED GOODS INVENTORY		F0152 CALC		
. UPC		ON-HAND.		RSVR	AVAIL	UNIT	MINIMUM.	RESULT	G/L
. CODE	DESCRIPTION	QTY	QTY	QTY	QTY	COST	QTY	FIELD	NO.
00001	CREOLE SEASONING, 8 OZ.	718	190	528	3.59	5000	90	1303	
00002	CREOLE SEASONING, 50 LB BOX	95.00	1	95.00	17.62	55	2.00	1306	
00003	CAJUN COUNTRY COOK BOOK	3160	2	3158	1.70000	2000	25	1207	
00004	CREOLE SEASONING, 7 LB/GAL	32.00	0	32.00	12.44	120	2.00	1305	
00005	CREOLE SEASONING, 17 OZ.	3023	73	3211	6.52	2500	60	1304	
00006	ROUX & GRAVY MIX, 10 OZ.	2217	32	2285	4.31	700	14	1320	
00007	ROUX & GRAVY MIX, 4 LB/GAL	24.00	0	24.00	7.53	15	1.00	1321	
00008	STEAK SAUCE, 10 OZ.	342	10	332	4.06	266	1	1335	
00009	THREE PIECE GIFT SET	141.00	0	141	2.96	60	3.00	1322	
00010	TWO PIECE GIFT SET	120.00	0	120	1.30	40	7	1323	
00012	BARBECUE	266	211	55	6.56	500		1357	
00013	BARBECUE		160	307	13.15			1343	
00250	CRAB BOIL, 8 OZ.	200	0	200	15.50	20	6.00	1344	
00250	CRAB BOIL, 1 GAL	-9.00	0			20	1.00	1339	
00250	SALT FREE SEASONING, 50 LB	39	0	40	36.51	2	1.00	1310	
00260	ROUX & GRAVY MIX, 50 LB BOX	16	0	16	15.77	20	10	1322	
00270	SEAFOOD SAUCE, 1 GAL	2.00	0	2.00	13.25	2	1.00	1334	
00280	STEAK SAUCE, 1 GAL	14	0	14	10.85	2	4	1336	
00290	HC SAUCE, 1 GAL	9	0	9	18.39	2	3	1333	
00300	CRAB BOIL, 50 LB BOX	70	0	71	17.54	60	4	1342	
17254	8OZ SEASONING-SHRIMP WRAPPED	100.00	0	100.00	7.94			1312	
17256	9OZ CRAB BOIL-SHRIMP WRAPPED	0	0	0	4.42			1315	
17255	10OZ ROUX SHRIMP WRAPPED	50	0	50	4.98			1314	
17257	17OZ SEASONING SHRIMP WRAPPED	0	0	0				1313	

図 14 最終製品在庫レポート
Fig. 14 Finished goods inventory report

プ C と D に適当な RID をポイントするインデックス。

タイプ G : 各製品の製造費のうち、労務費を計算する時に使用するタイプ。ランが作成したリザルトはスクリーンに表示され、SOE とカーソルで囲んで更新する。実行を再開すると、ランは各従業員ごとの作業時間と賃率を基にして労務費を算出する。

5.4.1 生産管理システム

このシステムの主な機能は、①製品材料と原材料から最終製品を製造する過程の管理、②各製品の製造コストの把握、③必要に応じて製品材料/原材料および製品の各在庫を管理することである。

タイプ C と D は各製品の製造に用いる材料の数量、および最終製品の製造単価を算出するのに用いる。図 15 は各タイプの RID の例である。

製品材料のバッチ混合数、製造する箱数、労務費が明らかになったら、製造レポート・ランを実行する。このランは製品材料のバッチ数を合計してタイプ C へ格納し(図 15 参照)、箱数の合計はタイプ D へ格納する。次に各タイプの“TOTAL QTY”(総量)フィールドを計算する。原材料在庫から単価を取り出す時は“MATCH”命令を用い、最後に“TOT”命令で各原材料別の総コストを算出している。各タイプ別の総コストも“TOT”命令を用いて算出する。

タイプ C および D は、マニュアル・ファンクションで利用することを念頭に置いてデザインした。そのためラン開発でも、ランとデータを文字通り独立させることができた。

図 16 はタイプ E のレポート例で、生産管理ランの実行が終わる都度、各タイプごとに 1 レコードを作成している。適正な数量、コストなどを記録することで、製造担当管理者は製造費、労務費の製造コストに及ぼす影響などを把握できる。

```

.DATE 07 APR 86 16:19:45 RID 1 27 NOV 85 LT
PRODUCTION RUN RECIPE FILE SEASONING C0144
*MATL . AMOUNT . NO OF . TOTAL . UNIT . TOTAL . CALC . G/L .
*CODE . /BATCH . BATCHES . QTY . COST . COST . FIELD . UOM. NO. .
-----
R1001 RED PEPPER 35000 HU 34 .96
R1002 BLACK PEPPER 34 MESH 28 2.30
R1003 CHILI POWDER 17 .89
R1004 GARLIC 17 1.46
R1005 MICRO CEL E 4 .72
R1006 SALT 400 .03750
..... END REPORT .....
    
```

```

.DATE 11 MAR 86 11:39:14 RID 1 29 NOV 85 LT
PRODUCTION RUN PACKAGING FILE 8 OZ SEASONING D0146
*MATL . AMOUNT . NO OF . TOTAL . UNIT . TOTAL . CALC . G/L .
*CODE . /CASE . CASES . QTY . COST . COST . FIELD . UOM. NO. .
-----
CC001 CORR. CASES 8 OZ 1 .1119300
FC001 FIBER CANS, 8 OZ 12 .0876000
AB001 ALUMINUM BOTTOMS 12 .01680
LA001 LABELS 8 OZ 12 .0059600
SS001 TAPE 2" * 1000 YD 2.5 .0046
..... END REPORT .....
    
```

図 15 生産管理に用いる“原材料構成表”(↑)

(↓) 図 16 製造レコード・レポート(モード 12, タイプ E)

DATE	PROD . UPC .										
02/27/86	00001	02/27/86	00001	02/27/86	00001	02/27/86	00001	02/27/86	00001		
02/20/86	00002	02/20/86	00002	02/20/86	00002	02/20/86	00002	02/20/86	00002		
02/07/86	00004	02/07/86	00004	02/07/86	00004	02/07/86	00004	02/07/86	00004		
02/10/86	00004	02/10/86	00004	02/10/86	00004	02/10/86	00004	02/10/86	00004		
02/19/86	00004	02/19/86	00004	02/19/86	00004	02/19/86	00004	02/19/86	00004		
02/20/86	00004	02/20/86	00004	02/20/86	00004	02/20/86	00004	02/20/86	00004		
02/25/86	00004	02/25/86	00004	02/25/86	00004	02/25/86	00004	02/25/86	00004		
02/27/86	00004	02/27/86	00004	02/27/86	00004	02/27/86	00004	02/27/86	00004		
02/28/86	00005	02/28/86	00005	02/28/86	00005	02/28/86	00005	02/28/86	00005		
02/28/86	00016	02/28/86	00016	02/28/86	00016	02/28/86	00016	02/28/86	00016		
02/28/86	00220	02/28/86	00220	02/28/86	00220	02/28/86	00220	02/28/86	00220		
02/28/86	00230	02/28/86	00230	02/28/86	00230	02/28/86	00230	02/28/86	00230		
02/28/86	00002	02/28/86	00002	02/28/86	00002	02/28/86	00002	02/28/86	00002		
02/28/86	00240	02/28/86	00240	02/28/86	00240	02/28/86	00240	02/28/86	00240		
145	309.74	218.34	22.78	550.86	3.80	145	309.74	218.34	22.78	550.86	3.80
30	490.04	21.12	.00	511.16	17.04	30	490.04	21.12	.00	511.16	17.04
50	479.97	110.87	32.42	623.26	12.47	50	479.97	110.87	32.42	623.26	12.47
82	769.14	184.03	34.21	987.38	11.90	82	769.14	184.03	34.21	987.38	11.90
34	653.38	75.37	26.91	755.66	22.23	34	653.38	75.37	26.91	755.66	22.23
23	326.69	51.01	18.23	395.93	17.21	23	326.69	51.01	18.23	395.93	17.21
41	326.69	90.95	18.68	436.32	10.14	41	326.69	90.95	18.68	436.32	10.14
567	2477.92	1120.80	129.28	3728.00	10.14	567	2477.92	1120.80	129.28	3728.00	10.14
223	595.66	430.39	54.68	1086.05	3.68	223	595.66	430.39	54.68	1086.05	3.68
490	1250.89	945.69	64.74	2196.58	6.28	490	1250.89	945.69	64.74	2196.58	6.28
274	1007.07	441.62	60.33	1448.69	4.15	274	1007.07	441.62	60.33	1448.69	4.15
248	592.84	386.94	36.45	1016.23	4.10	248	592.84	386.94	36.45	1016.23	4.10
171	671.76	341.98	51.71	1065.45	6.23	171	671.76	341.98	51.71	1065.45	6.23
230	659.32	528.01	51.71	1239.04	5.39	230	659.32	528.01	51.71	1239.04	5.39
28	464.61	19.79	.00	484.40	17.30	28	464.61	19.79	.00	484.40	17.30
29	534.92	56.57	20.55	612.04	21.10	29	534.92	56.57	20.55	612.04	21.10

図 15 “Bill of materials” RIDs for production control

Fig. 16 Production tracking report(mode 12, type E)

MATERIALS REQUIREMENTS FOR PRODUCTION PLANS (MRP)							
* UPC .	CASE	CASES	BATCH	RECIP.	PACKG.	CASES	
* CODE.	DESCRIPTION	QUANTITY	ON-HAND	REQMNT	RID	RID	/BATCH
00001	CREDLE SEASONING, 8 OZ.	10000.0	6045.0	49.4	1	1	80
00005	CREDLE SEASONING, 17 OZ.	5000.0	1402.0	94.7	1	2	38
00012	BARBECUE SAUCE, 18 OZ.	800.0	342.0	1.6	6	14	295
RAW MATERIALS REQUIREMENTS FOR PRODUCTION							
*MATL .	ON-HAND	ON-ORDER	AVAIL	REQ'D	TO ORDER.		
*CODE .	DESCRIPTION	QTY	QTY	QTY	QTY	QTY	
FC001	FIBER CANS, 8 OZ	31680.0	90000.0	121680.0	960.0		.0
FC002	FIBER CANS, 17 OZ	77940.0	0	77940	456.0		.0
GC001	GLASS CONTAINERS, 16 OZ DZ	6120.0	33600.0	39720.0	3540.0		.0
CC001	CORR. CASES 8 OZ	13475.0	0	13475	80.0		.0
CC002	CORR. CASES 17 OZ	9600.0	0	9600	38.0		.0
AB001	ALUMINUM BOTTOMS 2 1/2	127125.0	130000.0	257125.0	960.0		.0
TB001	TINPLATE BOTTOMS 3	126700.0	0	126700	456.0		.0
MC003	METAL	13800.0	33000.0	46800.0	3540.0		.0
LA001	LUM BENZOATE	0	0	287224	91.0		.0
RI023	TOMATO PASTE DRUMS	12840.0	0	12840	334.4		.0
RI023	WHITE DISTILLED VINEGAR	160.0	330.0	490.0	33.6		.0
RI024	COTTONSEED OIL	412.0	0	412	124.8		.0
RI027	METHOCEL	200.0	0	200	19.2		.0
SS001	TAPE 2"* 1000 YD	117000.0	0	117000	625.8		.0
SS011	GLUE, COLE TITE 47#/5GL	481.75	0	481.7500	14.8		.0

図 17 所定のケース数を製造するために必要な所要量を示す“MRP”レポート

Fig. 17 “MRP” report generated for required case production

図 17 は“MRP”レポートの例である。各製品製造についての要求を前提条件とし、タイプ C と D を逆に使用して、製造の所要原材料を算出することができる。製造担当管理者は、販売予測に基づいて必要な条件を指示する。するとランが製造計画ごとの総原材料を算出して、どの品目を発注すべきかがわかるレポートを作成する。

5.4.2 在庫管理システム

最終製品在庫を把握することが、このシステムの主目的である。必要に応じて最終製品は貨物車に積んで Louisiana 州の南部に所在する多数の顧客先を訪問、配送する。さらに通信販売の在庫は、倉庫からメールオーダー室へ移動し、そこから通信販売システムにより送付する。

タイプ F の別 RID を用いて、必要時に独立した在庫管理ができるようになっていく。そのため在庫間で製品を移動するためのランを開発し使用している。月末時の在庫総量は、やはり別のランで算出する。通信販売在庫もやはりタイプ F 内の RID を用いている。この在庫はケース単位でなく一品ごとに管理している。通販でケース単位はまれなためである。

5.5 売掛金/買掛金システム

両システムはモード 24 を使用している。このモードの内容は図 18 のようになっている。

FILE CABINET - MODE 24/25 DRAWER - TYPE				
.F.	FORM TYPE	.FORM.F.	FORM TYPE	.FORM.
.D.	DESCRIPTION	.TYPE.D.	DESCRIPTION	.TYPE.
B	ACCOUNTS RECEIVABLE	0302	F ACCOUNTS PAYABLE	0312
C	PAYMENTS - ACCT. RECEIV.	0304	G PAYMENTS - ACCT. PAYABLE	0314
D	OPEN	0306	H OPEN	0316
E	OPEN	0310	I RUN FUNCTIONS	0320

図 18 モード 24 の TOC

Fig. 18 Table of contents for mode 24

モード 24 のタイプは次の用途に用いている。

タイプ B：売掛金システム用。未回収の請求書レコードを格納する。このレコードは請求書発行プロセスで入力し、集金が完了するまでこのタイプ内に置かれる。格納するレコードは、請求書ナンバの下 1 桁を用いて、10 RID に分散してある。各売掛先別、支払期日数別に、RID のカラムを別けて分類してある(図 19)。

タイプ C：売掛金の回収データ用。回収した売掛金を記録する。顧客からの支払いを受領するつど、このタイプ内にレコードを作成してその支払い明細を記録しておく。図 19 下段に一例を示す。保存するレコードは顧客ナンバの下 2 桁を用いて、26 RID に分散してある。

タイプ F：買掛金システム用。仕入先から請求書を受け取ると、このタイプへ格納する。請求書の受領日、物品の納品日を基にして支払い期日を算出する。同時に割引価格を適用できる時は、それも計算しておく。図 20 はタイプ F のレポート例である。データ・レコードは仕入先の請求ナンバ下 1 桁に相当する 10 RID へ分散して保存している。請求ナンバがない場合は、RID 10 へ保存する。

```
.DATE 21 APR 86 10:14:29 RID 1 03 APR 86 LISA
. 36 AGED ACCOUNTING FILE BO302
*CLUST . TRAN . . 1-10 . 11-20 . 21-30 . 31-40 . OVER . INVOICE.
* NO. . DATE . TOTAL . CURRENT . DAYS . DAYS . DAYS . 40 DAYS . DAYS. NO .
*-----*-----*-----*-----*-----*-----*-----*-----*-----*
20003 860311 754.35 .00 .00 .00 754.35 .00 37 4041
06004 860311 531.00 .00 .00 .00 531.00 .00 37 4051
20003 860311 3775.49 .00 .00 .00 3775.49 .00 37 4061
04005 860311 881.46 .00 .00 .00 881.46 .00 37 4071
07003 860311 492.50 .00 .00 .00 492.50 .00 37 4081
16010 860311 325.00 .00 .00 .00 325.00 .00 37 4091
19013 860317 2155.00 .00 .00 .00 2155.00 .00 31 4121
19012 860317 97.20 .00 .00 .00 97.20 .00 31 4131
08003 860318 41.04 .00 .00 41.04 .00 .00 30 4151
16006 860319 36.00 .00 .00 36.00 .00 .00 29 4171
19013 860325 578.75 .00 .00 578.75 .00 .00 23 4201
20017 860326 190.40 .00 .00 190.40 .00 .00 22 4231
```

```
.DATE 18 APR 86 09:35:27 RID 1 30 JAN 86 TONY
. 18 ACCOUNTS RECEIVABLE - PAYMENT RECORDS CO304
*CLUST . PMNT . INVOICE. INVOICE . AMOUNT . DISC. . PROM . RET/ITM . OTHER .
* NO . DATE . NO. . AMOUNT . PAID . APPLIED . ALLOW. . DEDUC . DISC. .
*-----*-----*-----*-----*-----*-----*-----*-----*-----*
01001 860404 4070 76.04 74.52 1.52 .00 .00 .00
01001 860404 4161 170.54 167.13 3.41 .00 .00 .00
01001 860404 4162 38.60 37.83 .77 .00 .00 .00
01001 860404 4159 105.40 103.29 2.11 .00 .00 .00
01001 860404 4160 128.25 125.68 2.57 .00 .00 .00
01004 860404 4055 528.00 517.36 10.64 .00 .00 .00
01007 860404 4158 2676.05 2622.52 53.53 .00 .00 .00
01008 860404 4040 609.90 597.71 12.19 .00 .00 .00
01002 860418 4063 2266.05 2266.05 .00 .00 .00 .00
01002 860418 4085 528.50 528.50 .00 .00 .00 .00
01002 860418 4056 1763.10 1763.10 .00 .00 .00 .00
01002 860418 4164 4289.73 4242.35 .00 .00 .00 .00
```

図 19 売掛金用タイプ。タイプ B は未回収売掛金を経過日数別に保存している。またタイプ C は売掛金の支払い状況を記録している。

Fig. 19 Accounts receivable type

```

LINE 1  FMT 3 RL  SHFT  HLD CHRS  HLD LN
.DATE 09:22:58 RID 1 08 MAY 86 DIANA
. 11 ACCOUNT PAYABLE - VENDOR INVOICE DETAIL RECORDS F0312
* P/O .VENDR. INVOICE .INV RCVD.MER RCVD. INVOICE .DISC END. DISCOUNT . G/L
* * * * * . DATE . DATE . AMOUNT . DATE . AMOUNT . #
-----
0 TV042 1795 03/14/86 03/14/86 330.00 03/14/86 .00 2020
0 TV049 1797 03/18/86 03/18/86 37.50 03/18/86 .00 6130
0 TV041 1798 04/01/86 04/01/86 150.00 04/01/86 .00 2020
0 TV041 1799 04/01/86 04/01/86 17.40 04/01/86 .00 2020
0005 APR55741 04/14/86 04/14/86 8.74 2020
3026 TV022 20075 05/01/86 05/01/86 .00 6180
2996 TV043 B79718100 04/21/86 04/21/86 230.45 05/01/86 .00 2020
2867 TV044 75992 04/21/86 03/04/86 117.48 05/01/86 .00 2020
    
```

図 20 買掛金用タイプ。支払い期日の到来順に請求書が並べてあり、割引価格の適用の有無も一覧できるようにしてある。

Fig. 20 Accounts payable type

```

LINE 1  FMT  RL -  SHFT  HLD CHRS  HLD LN
.DATE 03 APR 86 09:17:57 RID 8 31 JAN 86 TONY
.RUN NAME: RUNB BY:
=====
@BRK LDV V9F9.2=.00,V1I3=1 .
@10:SRU,24,B,V1,6,,90 D 2-5 ,00000/R,99999 .
@TOT,24,B,-0 ' 25-9,35-9,45-9,55-9,65-9 ,=V9,=V9,=V9,=V9 .
@DAT,24,B,-0 ' 8-6,75-4 ,-,= .
@CAL,24,B,-0 R.01 15-9,25-9,35-9,45-9,55-9,65-9,75-4 ,A,B,C,D,E,F,G \ .
IF:G>10&G<21; THEN: C=A; IF: G>20&G<31; THEN: D=A; IF: G<11; THEN: B=A; \ .
IF: G>30&G<41; THEN: E=A; IF: G>40; THEN: F=A; .
@UPD .
@DSP,24,B,V1,,,Y .
@90:CHG V1 V1 +1 IF V1 < 11 GTD 10 .
@WAT 5000
3 *****
3 *****
3 ***** ACCOUNTS AGING PROCESS COMPLETE. *****
3 *****
3 *****
@BRK OUT,24,I,-0,2,10,1,1,Y,Y,,P .
@WAT 5000 .
@RUN MENU .

..... END REPORT .....
    
```

図 21 売掛金エージ分析ラン

Fig. 21 "Aging account receivable" run listing

タイプG：買掛金の支払いデータ用。買掛金を支払うため小切手を発行すると、その記録はこのタイプに保存される。この記録には支払い小切手の番号、金額などがあり、チェック・レジスタとして機能している。

5.5.1 売掛金のエージ分析

すべての売掛金レコードについて、その支払い期日が到来するまでの日数(エージ)を分析する MAPPER ランを作成した(図 21)。タイプ B もマニュアル・ファンクションでそのまま利用できるように RID のヘッダを設計してあるため、MAPPER ランも極めて簡単なもので十分役立っている。

5.5.2 買掛金システム

図 22 は、支払い期日の到来する買掛金リストを作成する MAPPER ランである。これも極めて簡単なランで、実用性の高いレポートが作成できる一例である。

5.5.3 総勘定元帳とのインタフェース

総勘定元帳の勘定を更新するトランザクションがあれば、自動的にその勘定を更新するようにしてある。総勘定元帳の項で、各勘定の借方/貸方記入を行うランで使用す

のサブルーチンのリストを紹介する。

ここでは、会計システムをリアルタイムで維持管理できることが大きな課題である。リアルタイムで貸借対照表のほか、全勘定残高をレポートすることが求められていた。

5.6 総勘定元帳システム

このシステムとインデックス（勘定一覧）はモード 20 に作成した。タイプ B から H まで、すべて同一形式で生成してある。各タイプに勘定ナンバ、取引発生日、取引内容の記述、参照ナンバ、借方金額、勘定残高が入っている。モード 20 の TOC は図 23 の通りである。

クレオール食品では、すべての勘定に 4 桁の番号を付けているが、データ・フィールドは 5 桁で設定してある。勘定ナンバは 1000 から 8999 までである。そこで、タイプ I のインデックス RID を勘定ナンバの最初の 1 桁で分割している。つまり、1XXX の勘定ナンバのインデックスは RID 1I を使用するのである。タイプ H だけが例外で、7XXX と 8XXX の勘定が入る。インデックスはそれぞれ 7I と 8I を用いている。

図 24 は勘定ナンバ 5XXX のインデックス RID を示している。合わせて勘定 RID の例も紹介してある。

特定の勘定を更新するため、サブルーチン・ランを作成した(図 25)。このサブルーチンはすべてのランで共用することにしてある。サブルーチン・ランを呼び出すランは、あらかじめ決めてある変数に勘定ナンバ、金額、借方/貸方コード、取引内容の説明、参照番号(必要時)、および取引日を入れて渡す。サブルーチンは現在の借方残、あるいは貸方残などの勘定残高を読み出して、その取引を書き込む(@WRL)。そのとき前述したラインカウンタ方式を用いている。

```

LINE 1      FMT      RL -      SHFT      HLD CHRS      HLD LN
.DATE 22 APR 86 09:37:58 RID      6      30 JAN 86      TONY
.RUN NAME: RUN6      BY:
*****
@BRK . DISPLAY ACTIVE ACCOUNTS PAYABLE

DISPLAY ALL ACTIVE ACCOUNTS PAYABLE

SORT REPORT BY: ,2("V"=VENDOR, "D"=DATE)

TRANSMIT> ,2("X" TO RETURN TO A/P MENU)
@BRK OUT,24,I,-0,2,23,1,1,Y,,P .
@CHG INPUT$ V10H1,V99H1 IF V99 = X RUN MENU,10 .
@SRH,24,F,,6,,91 DHR1-1000 8-5 ' ,@@@@ .
@IF V10 = V SDR,24,F,-0 ' 8-5 ' ,1 GTO 10 .
@IF V10 = D SDR,24,F,-0 ' 61-2,64-2,67-2 ' ,2,3,1 GTO 10 .
@10:DSP,24,F,-0,,,3 .
@RUN MENU,10 .

..... END REPORT .....
    
```

図 22 買掛金レポート作成ラン

Fig. 22 Accounts payable report run listing

----- FILE CABINET - MODE 20/21 DRAWER - TYPE -----				
.F.	FORM TYPE	.FORM.F.	FORM TYPE	.FORM.
.D.	DESCRIPTION	.TYPE.D.	DESCRIPTION	.TYPE.

B G/L	- ASSETS	0242 F G/L	- SELLING EXPENSES	0252
C G/L	- LIABILITIES	0244 G G/L	- OPERATING EXPENSES	0254
D G/L	- SALES	0246 H G/L	- MISCELLANEOUS	0256
E G/L	- COST OF SALES	0250 I	GENERAL LEDGER INDEX	0260

図 23 モード 20 の TOC

Fig. 23 Table of contents for mode 20

```

LINE 1  FMT  RL  -  SHFT  HLD CHRS  HLD LN
.DATE 07 FEB 86 13:44:39 RID 5 03 JAN 86 TONY
GENERAL LEDGER ACCOUNT INDEX IO260
* G/L . . . . .
* NO. . . . . DESCRIPTION . . . . . T.R.I.
* . . . . . Y.NO.
=====
5001 BROKER ADV.-METZDORF F 1
5002 ADVERTISING-COUPONS F 2
5003 ADVERTISING-BOOTH SPACE F 3
5004 ADVERTISING-CONSULTANT F 15
5005 IN-HOME PROMO MATERIAL F 4
5006 ADS - PROMO - SLOTTING ALLOW F 16
5010 ADVERTISING F 5
5020 AMORTIZATION F 6
5030 AUTO EXPENSE F 7
    
```

```

LINE 1  FMT 2 RL  SHFT  HLD CHRS  HLD LN
.DATE 21 APR 86 09:28:39 RID 11 03 APR 86 TONY
* 23 GENERAL LEDGER ACCOUNT: 3303 - SEASONING-8 0Z.
* G/L . TRAN . . . . .
* NO . DATE . DESCRIPTION . DEBITS . CREDITS . BALANCE .
=====
3303 04/03/86 STARTING BALANCE .00 .00 .00
3303 04/08/86 REVENUE SALES - 04/08/86 .00 9163.15 9163.15
3303 04/08/86 REVENUE SALES - 04/08/86 .00 3360.30 12523.45
3303 04/09/86 REVENUE SALES - 04/09/86 .00 10703.60 23227.05
3303 04/09/86 REVENUE SALES - 04/09/86 .00 4389.50 27616.55
3303 04/10/86 REVENUE SALES - 04/10/86 .00 1014.30 28630.85
3303 04/11/86 REVENUE SALES - 04/11/86 .00 8823.65 37454.50
3303 04/11/86 REVENUE SALES - 04/11/86 .00 885.55 38340.05
3303 04/11/86 MAIL ORD SALES-04/11/86 .00 64.75 38404.80
3303 04/14/86 MAIL ORD SALES-04/14/86 .00 28.00 38432.80
    
```

図 24 総勘定元帳の勘定グループを指すインデックス RID. このインデックスは取引の詳細レコードを格納しているタイプと RID をポイントしている。

Fig. 24 Index RID to a general ledger account group

```

.DATE 04 APR 86 10:43:43 RID 20 08 JAN 86 TONY
.RUN NAME: SUBROUTINE (UPDATES TO GENERAL LEDGER ACCOUNTS)
*****
@1:BRK . . V110=G/L NO, V111=DATE, V112=DESC(24), V113=AMT, V114=TYPE
@LDV V100I1=V110(1-1) . V100 IS INDEX RID IN MODE 20, TYPE 1 .
@FND,20,I,V100,6,91 ' 2-5 ,V110 ,V105I3 .
@RDL,20,I,V100,V105 76-1,78-2 V102H1,V103I2 .
@LDV,R V147F11.2=.00 .
@LDR,20,V102,V103 RDL,20,V102,V103,2 2-4 V107I4 LZR,20,V102,V103 V108I4 .
@IF V107 > V108 LN+,20,V102,V103,V108,20 . ADD LINES IF NOT ENOUGH ROOM
@CHG V109I4 V107 -1 RDL,20,V102,V103,V109 77-11 V140F11.2 .
@IF V114 = C,(50),D,(60) .
@50:CHG V140 V140 +V113 . ADD TO PREV BALANCE (CREDIT)
@WRL,20,V102,V103,V107 2-5,B-8,28-24,53-11,65-11,77-11 \ .
,V110,V111,V112,V147,V113,V140 GTD 65 .
@60:CHG V140 V140 -V113 . SUBTRACT FROM PREV BALANCE (DEBIT)
@WRL,20,V102,V103,V107 2-5,B-8,28-24,53-11,65-11,77-11 \ .
,V110,V111,V112,V113,V147,V140 .
@65:CHG V107 V107 +1 WRL,20,V102,V103,2 2-4 .,V107 ULK .
@ESR .
@91 . ERROR U
ERROR UPDATING GENERAL LEDGER.....
ACCOUNT NUMBER= V110 INDEX RID= V100
RID NUMBER= V103 IN TYPE= V102
..... END REPORT .....
    
```

図 25 総勘定元帳を更新するシステム共用のサブルーチン・ラン

Fig. 25 General ledger updating run subroutine used to update G/L accounts throughout the system

総勘定元帳のランはモード 22 に入っている。このモードには財務レポート・システムも入っている。ランとして次のものを用意している。

- 1) 勘定の初期設定ラン……勘定をクリアし、期首残高を設定する。
- 2) 勘定の調整ラン……指定した勘定の借方または貸方金額を調整する。
- 3) 勘定表示ラン……指定した勘定の詳細な取引レポートを表示する。

その他のランとして、新会計年度時に全勘定を初期設定するラン、支払い給与を総勘定元帳に転記するランがある。後者は MAPPER による給与システムが完成するまで、臨時に使用するものである。

5.7 財務レポート・システム

5.7.1 財務レポート

MAPPER で財務レポートを作成することについては、若干の懸念があった。MAPPER はカラムを主としたデータはうまく扱える。しかし財務レポートは、レポート上の位置(項目)によって加算や減算をしなくてはならない。

次のレポート例を見れば、項目の位置の重要性がどんなものであるかがわかるであろう。

Description	First Quarter	Second Quarter	Third Quarter	Fourth Quarter
Sales				
Sales - Item 1	1,000.00	2,000.00	3,000.00	4,000.00
Sales - Item 2	1,500.00	3,000.00	4,000.00	3,000.00
Total Sales	2,500.00	5,000.00	7,000.00	7,000.00
Expenses				
Expenses 1	800.00	1,500.00	2,000.00	2,500.00
Expenses 2	900.00	1,000.00	2,300.00	1,800.00
Total Expenses	1,700.00	2,500.00	4,300.00	4,300.00
Gross Profit	800.00	2,500.00	2,700.00	2,700.00

このレポートでは、“Total Sales” および “Total Expenses” は、トータライズ命令(@TOT)に S オプションを指定すれば簡単に算出できる。問題は“Gross Profit”行である。このレポートは簡単な例であるが、一般の財務レポートは小計のレベルも多く、各レベルで加算と減算を行う必要がある。

MAPPER を用いてこの種のレポートを効率よく作成するには、行だけを意識したランではまずい。また標準命令である SEARCH, TOTALIZE, CALCULATE, MATCH などを使用したランにすべきである。@RDL, @CHG, @WRL だけを用いたランも使用したことがあるが、処理すべきレポートの行が追加/削除された場合の使い方がむずかしい。

5.7.2 MAPPER での財務レポートの作成法

以下、説明する手法は本稿作成時点では、必ずしも目新しいものではなくなった。しかしクレオール食品のシステムを開発する時点では、まだこの方法を紹介した文献はなかったと思う。

前に例示した財務レポートを素材にして、“MAPPER 流”で作成する方法を述べる。まず MAPPER の RID を次のように作成する。

```

*          .First .Second .Third .Fourth . . .
* Description .Quarter .Quarter .Quarter .Quarter .1.2.ML
*=====,=====,=====,=====,=====,=,=,=,=
* Sales
  Sales - Item 1                               1
  Sales - Item 2                               1
* -----
* Total Sales                                 1 1 +1
*
* Expenses
  Expenses 1                                   2
  Expenses 2                                   2
* -----
* Total Expenses                             2 1 -1
*
* -----
* Gross Profit                               1
*

```

この RID を表示して“FCC”ランを実行すると、次のようなりザルトが得られる。

```

*          .First .Second .Third .Fourth .L.L. .
* Description .Quarter .Quarter .Quarter .Quarter .1.2.ML.
*=====,=====,=====,=====,=,=,=,=
  2-16        19-9        29-9        39-9        49-9        50-1
                                           52-1
                                           54-2

```

続いて次の手順でレポートを作成する。

- 1) レポートのカラム 50 を SEARCH 命令でレンジ・サーチする。レンジは 1 から 9 までとし、オプション D と L(*) を指定する。
- 2) リザルト・レポートを TOT 命令でトータライズする。オプションとして S と O を指定し、編集したい時は J(C) と R.01 も指定する。四つの Quarter (四半期) カラムの合計を求める。そのときカラム 50 に S を指定する。その結果、“Total Sales” と “Total Expenses” に算出した小計が付いた 2 行のリザルト・レポートが得られる。
- 3) このリザルトと元の RID とをマッチ・アップデートする。元の RID はスクリーン上に表示してあるので、オプション I と D を指定してマッチして得たりザルトのヘッダを削除する。次に “Description Field” をマッチして四つの “Quarter Columns” を移動する。最後に UPD 命令でリザルトを元の RID へ書き戻して更新する。これで第 1 段階は終わる。
- 4) 次の処理へ進むには、前述のステップ 1 と同じように SEARCH 命令を実行するが、その時はカラム 52 をサーチする。
- 5) ステップ 2) と同じくトータライズをするが、その前に CALCULATE 命令を使用する。まず J(C) と R.01 オプションを必要に応じて指定して、四つある四半期カラムのそれぞれのフィールドにカラム 54 を掛け算する。この計算で減算フィールドを負の値に設定し、加算フィールドを正の値にする。
- 6) ステップ 2) を繰り返し、小計のパラメータは第 2 レベルのカラム 52 に入れる。
- 7) ステップ 3) のマッチ・アップデートを繰り返す。

これで第 2 段階の小計が算出でき、例題のレポートは完成する。上記ステップを繰り返すことで、さらにカラムの数を増やしたレポートも作成できる。

以上のレポート作成は、次頁(上)の MAPPER ランを実行しても可能である。

このランでは、処理対象レベルのカラムしか、データを変更しない。変数を設定すれば、第 2 レベルのステップを必要な回数だけ繰り返し実行できる。

```
*RUN FUNCTION: FIN-REP
*****
@SRH,30,B,2,6,90 DL(*) 50-1 ' ',1/R,9 . SEARCH 1ST LEVEL
@TOT,30,B,-0 SOR.01J(C) 19-9,29-9,39-9,49-9,50-1 ' ',+,+,+,+,S .
@MCH,30,B,-0,30,B,2 D 2-16,19-9,29-9,39-9,49-9 ' ',1,A,B,C,D \ .
2-16,19-9,29-9,39-9,49-9 ' ',1,A,B,C,D RNM -1 .
@ . LEVEL 1 DONE - SAVED RESULT AS -1 INSTEAD OF @MAU & @UPD
@SRH,30,B,-1,6,92 DL(*) 52-1 ' ',1/R,9 . SEARCH LEVEL 2
@CAL,30,B,-0 R.01J(C) 19-9,29-9,39-9,49-9,54-2 ' ',A,A,A,A,B \ .
A=A*B; .
@TOT,30,B,-0 SOR.01J(C) 19-9,29-9,39-9,49-9,52-1 ' ',+,+,+,+,S .
@MCH,30,B,-0,30,B,-1 2-16,19-9,29-9,39-9,49-9 ' ',1,A,B,C,D \ .
2-16,19-9,29-9,39-9,49-9 ' ',1,A,B,C,D .
@DSP,30,B,-0 . DISPLAY COMPLETED FINANCIAL REPORT
```

図 26 は、貸借対照表の資産の部を設定する方法を示している。

```
.DATE 07 FEB 86 14:42:30 RID 1 06 JAN 86 TONY
. BALANCE SHEET - ASSETS
*
* DESCRIPTION : AMOUNT . G/L . SUB LVL .
* : NO. .1.2.3.4.5.6.
*-----*-----*-----*-----*-----*
*
* CURRENT ASSETS
AMERICAN BANK - OPERATING 1030 0
AMERICAN BANK - PAYROLL 1040 0
FIRST NATIONAL BANK-SAVINGS 1050 0
FIRST NATIONAL BANK-PETTY CASH 1059 0
FIRST NATIONAL BANK-REGULAR 1060 0
FIRST NATIONAL BANK-PAYROLL 1065 0
CASH IN BANK-MONEY MARKET 1066 0
CASH IN BANK-MM II 1067 0
CASH IN BANK-FIRST FEDERAL 1068 0
CASH IN BANK-FNB SAVINGS 1069 0
PETTY CASH 1070 0
* ----- 1088 0
* CASH 1089 0 1
*
*
* ACCOUNTS RECEIVABLE 1090 1
* PREPAID INSURANCE 1100 1
* PREPAID ADVERTISING 1105 1
* PREPAID INCOME TAX 1110 1
* EMPLOYEE ADVANCES 1115 1
* EMPLOYEE GARNISHMENTS 1116 1
* EMPLOYEE PENALTY 1117 1
* NSF RECEIVABLE 1120 1
* INVESTMENT-SHEARSON 1130 1
* INVESTMENT-THOMSON MCKINNON 1135 1
* INVENTORY-BOOKS 1249 1
* INVENTORY-RAW MATERIALS 1259 1
* INVENTORY-FINISHED GOODS 1379 1
* -----
* TOTAL CURRENT ASSETS 1389 1 1 1
*
* PLANT & EQUIPMENT
WAREHOUSE LABOR CLEARING 1390 2
OFFICE FURNITURE & EQUIPMENT 1505 2
MACHINERY & EQUIPMENT 1510 2
TRANSPORTATION EQUIPMENT 1520 2
BUILDING IMPROVEMENTS 1530 2
COMMERCIAL TV STAGE 1550 2
ACCUMULATED DEPRECIATION 1680 2
* -----
* NET PLANT & EQUIPMENTS 1699 2 1
*
* OTHER ASSETS
COPYRIGHT 1710 3
ACC AMORT-COPYRIGHT 1720 3
* -----
* TOTAL OTHER ASSETS 1799 3 1
*
* TOTAL ASSETS 1899 1
```

図 26 貸借対照表、資産の部に用いる財務レポート RID の例。総勘定元帳の勘定ナンバを用いて現残高とマッチさせ、小計を求める。

Fig. 26 Example financial report RID for resets/balance sheet report

6. おわりに

本稿では当社における顧客管理，オーダ・エントリ，購買管理，生産管理，在庫管理，売掛金，買掛金，総勘定元帳，財務システムなどについてデータベースの設計とRUNの開発技術を中心に述べてきた。MAPPER 5の採用によって，このような企業の基幹とも言うべき業務を短期間に比較的容易，かつ統一のとれた形でシステム化できた。

本稿が，中規模製造業のシステム化に何らかのお役に立てれば幸いである。

(統合 OA システム部 横山 正敏 訳)

論文

NASA における MAPPER 活用事例

— MAPPER と DMS データベースを用いた 資源アカウンティングおよび利用管理システム

A Resource Accounting and Utilization System Using MAPPER and DMS Database at the NASA Slidell Computer Complex

F. Lovato, M. W. Mickelborough

要約 システム資源アカウンティングは、長期計画、利用報告、ユーザへのコスト配賦のために必要な機能である。本稿は、NASA のスライデル・コンピュータ・センタにおけるコンピュータおよび、その他の資源用の統合アカウンティング・システムの開発の背景について述べたものである。本システムは、MAPPER と DMS データベースを併用したものであり、本稿ではその概要および設計上の考慮点について紹介する。

Abstract Resource accounting is a necessity for long-range planning, reporting and user billing. This report gives the rationale for developing the NASA Slidell Computer Complex integrated accounting system for computer and non-computer resources. Design considerations are presented along with an overview of the system which uses both MAPPER and DMS databases.

1. はじめに

スライデル・コンピュータ・コンプレックス (SCC, Slidell Computer Complex) は米国航空宇宙局 (NASA) の一部門であるジョージ C. マーシャル飛行センターに所属する機関で、Louisiana 州 Slidell に本拠地がある。SCC は 1962 年にサタン S-1 B および S-1 C アポロ宇宙船のブースタ研究開発を支援する目的で設立された。その後 SCC は NASA のスカイラプ計画やアポロ-ソユーズ計画などや、その他の米国政府機関のサポートを業務としている。

発足以来、SCC はプロジェクト別に、特定ユーザを対象とした支援業務を実施している。コンピュータ関連の業務に加えて、テレメータ・データの解析、コンピュータによる出力のマイクロフィルム (COM) 化、およびテープ・ライブラリの一元管理を実施しているが、いずれもシステム資源の利用について課金する必要がある。ユーザ部門は計画済みプロジェクトの各工程情報を SCC へ提出する。そのとき各要求に支障を与えないように SCC の所有資源が調整できなくてはならない。この過程で SCC は、ユーザ部門で正確な計画見積りを行うための正確な情報が提供できないという問題を抱えていた。

1982 年、従来までの MIS を評価して 80 年代のシステム・ニーズを満たせるかどうかを判定することになった。当初、この作業は簡単だと考えられていたが、評価作業が進むにつれてさまざまな問題が浮かび上がってきた。まず第 1 に、この MIS についての要求事項が文書化されていなかったことである。システムは長年の間にただ“膨脹”し続けたのである。第 2 は MIS のユーザが誰で、システムを利用する理由は何な

のかが不明確であることだ。多様な帳表が作成され配付されてはいたが、一体どのよう
に利用されているのか、はたして実際に受け取っているのか、ほんとうに必要なの
かが不明であった。

MIS システムの評価作業は、まず帳表・報告書類の配付先をすべて洗い出し、受取
人に面談して調べることから始めた。その結果、対象帳表・報告書のおよそ半数は不
要であることが判明、ただちに作成と配付を中止した。

評価分析作業が進むにつれて、この MIS システムを定義するものがまったくない
ことが明らかとなった。一体、何が求められてこのシステムはできあがったのか。こ
の検討の結果、システムに欠落している事柄が次のように明らかになった。

- 1) データ・レコード形式に依存性が高いこと……オペレーティング・システムが
定義するアカウントのレコード形式がリリース・レベルによって変わると、
システムの主要部分のほとんどは書き直しが必要である。
- 2) 拡張性がないこと……新たに設けられたシステム・パフォーマンス・データを
モニタしようとする、データベースのほとんどを作り直すような作業が必要で
ある。将来の新たなデータ・ニーズによっては、別の独立したデータベースとシ
ステムが必要となることが考えられる。
- 3) 制約条件が多すぎる……データベース上に存在するデータでも、臨時に検索し
ようとするプログラムとプログラムの全面変更とプログラム要求手順を踏まなくてはなら
ない。仮りにそのプログラムができ上がったとして、その後プログラムを修正した
くても元の開発担当者しかわからないために対応できない。

そこで原点に戻り、システム本来の目的を次の通り設定した。

- 1) 可能な限り入力データ形式とは独立したシステムであること。
- 2) 拡張が容易で要求条件に変更が生じたとき適切に変更できること。
- 3) ユーザが必要に応じてデータベースを随時検索し、レポートや分析に使用でき
る機能を備えること。
- 4) 可変長のデータ形式と全 ASCII データが扱える設計であること。
- 5) SCC システムへの機能追加あるいは再編成ができること。
- 6) システムの制約条件は可能なかぎり少なくし、既存のプログラミング言語の他、
当時新たに開発中であった MAPPER や Query ランゲージ・プロセッサ
(QLP 1100) などの対話方式ソフトウェアを活用した柔軟なものにすること。

こうしてシステムに求められる事柄を明らかにした後、その要求事項を充足する市
販システムを探したが適当なものは見あたらなかった。そこで 1983 年 11 月、統合化
SCC 情報システム（略称 ISIS）の開発に着手したのである。

さて、ISIS の技術的詳細を紹介する前に、ISIS の稼働環境を理解する上で重要な
SCC の負荷予測システムについて触れておきたい。

SCC 負荷予測システムはユーザ部門と SCC のマネジメント間で完結したクローズ
ド・システムである。ISIS データベースには次の各項について、ユーザ部門の将来 5
年間にわたる予測データと、実績データが累積保管してある。

- 1100 SUP 時間
- マスストレージのトラック数

- マイクロフィッシュ/マイクロフィルム・フレーム (FR-80)
- コンピュータ出力のマイクロフィルム・ページ (FR-80)
- データ解析ラボラトリの処理時間
- テープ・ライブラリのリール数
- 保管テープおよび購入テープ
- プログラミングにかかる人・時間
- コピーしたページ数
- 有償ソフトウェアの CPU 利用時間
- ファシリティ・サービス
- 用品
- 専有面積
- その他ハードウェアの要求事項

負荷予測システムは、以上の各項について過去の使用実績、当年実績、過去の予測値をユーザ別に抽出しレポートを作成する。ユーザ部門はこのレポートを受け取り、最新の要求条件を加味して必要な変更を施す。その結果のレポートを再び SCC へ戻し、ISIS データベースを更新する。更新後のデータを基にしてマネジメント・レポートを作成、このレポートをベースとして SCC の計画立案サイクルが再び始まる。システム資源利用の予実対比情報は月次の SCC マネジメント・レビューに提出される。このクローズド・マネジメント・サイクルは、ハードウェアおよび人員を含めた長期システム資源計画に求められる情報を提供している。

2. ISIS の概要

コンピュータ・システムにおいてシステム資源アカウンティングは、システムの長期計画、利用報告、ユーザへのコスト配賦上、必要である。長期計画はコンピュータおよびそれ以外のシステム資源、ならびにサービスの所要量を、当年度および次年度以降について見積もる作業である。ユーザへのコスト配賦、つまり使用料の課金と利用状況の報告を行うには、コンピュータ自体およびそれを取り巻く資源とサービスのすべてについて、実績値を把握、保存、検索できなくてはならない。

こうして得た実績値を予測値と比較すれば、システムのキャパシティ・プランニングおよび課金方法に採用しているシステム利用率の妥当性がわかる。以上の各機能は米国航空宇宙局の ISIS および SCC で稼働している。

ISIS で実現している機能を挙げると、次のようになる。

- 1) 当年度から 5 年先までの全所要システム資源の利用状況を予測する。
- 2) システム資源利用実績への課金。
- 3) システム資源利用の見積もりと実績を比較し (予実対比)、その分散状況をレポートする。
- 4) システム資源利用の予実対比を各種の方法でレポートする。
- 5) SCC のユーザ部門が利用したシステム資源について課金を行う。
- 6) デバイス別のアイドル時間を記録する。
- 7) リモート・ターミナル別の利用状況を記録する。

- 8) QLP 1100, および MAPPER ランとマニュアル・ファンクションを利用して臨時、随時のレポート作成機能を提供する。
- 9) 無許可アクセスからデータを保護する。
- 10) エンド・ユーザが直接データをアクセスする手段を提供する。
- 11) ユーザにデータ処理とレポート作成の教育を実施する。
- 12) MAPPER についてはメニュー方式を用い、対話形式のスクリーンを通して利用させる。

3. デザイン上の考慮

デザイン全体として柔軟性のあるシステムを構築し、エンド・ユーザが直接、システム資源の利用実績データおよび予測データにアクセスできる手段を提供することが求められた。またエンド・ユーザが必要なデータを抽出し、レポートを作成できること、そのデータをダウンライン・ロードし、処理をした上で、目的に最も適した方法で保存できなくてはならない。しかもプログラミング・スタッフの手を借りず、エンド・ユーザ自身ができることが条件となる。加えて、特定形式のレポートが作成できなくてはならない。このレポートにはユーザに対する課金情報も含められるようにしたい。以上、デザイン上の要点を考慮すると、次のようなシステムとなる。

- 1) システム資源利用データについて、予測値および実績値を完全に統合化する。
- 2) エンド・ユーザおよびプログラマの双方にとり、フレンドリ（親しみやすい）なシステムにする。
- 3) 1100 オペレーティング・システムが採用するアカウンティング・ログ・ファイル、あるいは MAPPER アカウンティング・ログ・ファイルの各データ入力形式とは独立したデータ形式を持たせる。
- 4) データベースは簡単に拡張できるものにする。
- 5) 使用方法が簡単で、維持管理が楽なシステムにする。
- 6) 柔軟なシステムであること。
- 7) オンラインでデータ入力と検索ができること。
- 8) ユーザが直接、データをアクセスできること。
- 9) 最新技術に基づくソフトウェアを採用すること。

4. システム開発

デザインしたシステムを実現する第1ステップは、データベースに含めるべきデータの種類を SCC のユーザの立場から明確に洗い出すことであった。そこで一般ユーザを対象としてアンケート調査を実施し、その一方でレポート作成などの諸要求条件を検討するミーティングを開催した。こうして収集したユーザの回答と希望する仕様を基に、MAPPER および DMS データベースを設計した。

プログラミング・スタッフは MAPPER の知識を持っていなかったため、全員が MAPPER のマニュアル・ファンクションとランのデザインについての講習会に参加した。講習会は SCC のファシリティ・マネジメント受託会社 SDC が実施した。同社は、また ISIS の MAPPER データベースの設計を全面的にサポートした。

プログラムは COBOL, MAPPER, DML COBOL を用いて開発し、できあがったプログラムはマネジメント・インフォメーション・システムの本番稼働システム上で並行してテストを行った。この並行テスト本番の結果についてユーザ部門の照査を得た後、ISIS の本番システムとして導入され、稼働を開始した。その後、若干の調整を経てから、旧システムは廃止された (図 1)。

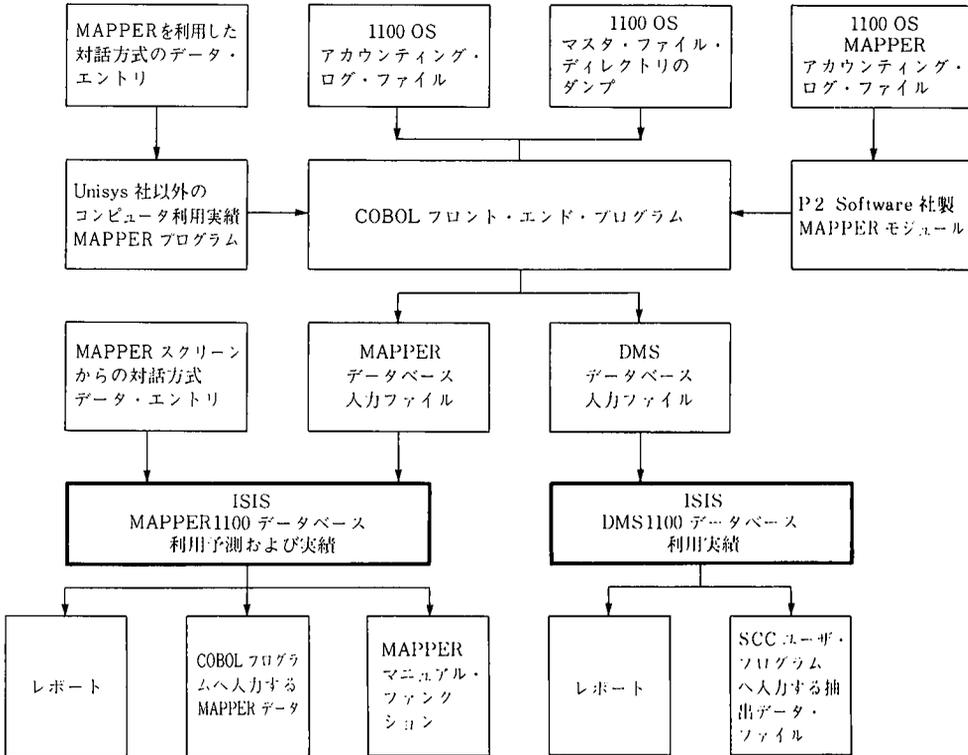


図 1 ISIS システム・フローチャート

Fig.1 ISIS system flowchart

5. ISIS システム

ISIS システムの処理は、UNIVAC シリーズ 1100 のオペレーティング・システムの アカウンティング・ログ・ファイル、MAPPER アカウンティング・ログ・ファイルの 日次処理、および真夜中に採取したマスタ・ファイル・ディレクトリ (MFD) のダンプ処理から始まる。MAPPER アカウンティング・ログ・ファイルは、BILLER/1100 ソフトウェア・パッケージの MAPPER モジュールへ入力する。

同パッケージは P2 Software 社が販売している製品である。このソフトウェアは SUP 消費量と各 MAPPER セッションでプリントしたライン数を計測する。BILLER/1100 MAPPER モジュールの処理結果、OS 1100 アカウンティング・ログ・ファイル、MFD ダンプ、および MAPPER が生成する Unisys 社製以外のコンピュータ利用情報を、それぞれ一連の COBOL プログラムに入力してファイルを作成する。これらのファイルは、DMS 1100 データベースおよび MAPPER データベースと一緒に更新される。この過程で Unisys 社製の全システム、およびそれ以外のシステムについて、その利用実績を処理する。

システム資源利用の全予測値は MAPPER データベースだけに入力する。同じく実績値についても MAPPER データベースに入力する。Unisys 社以外のシステムとしては、データ解析用コンピュータ、コンピュータ・アウトプット・マイクロフィッシュ (COM)、およびグラフィックスと非グラフィックス COM 機器などが含まれている。そのデータは MAPPER ランを利用してオンライン対話方式で入力する。

5.1 ISIS の DMS 1100 データベース

ISIS システムは全 Unisys 社製のシステムおよびそれ以外のシステムに関する利用実績データだけを、DMS 1100 データベースに格納している。資源利用の予測値は含まれていない。このデータベースは日次処理で更新し、磁気テープにバックアップする。DMS のリカバリ・ルーチンはいらず、このバックアップ・コピーからリカバリする。このデータベースは、パスワードの所有を認められた全 SCC ユーザがアクセスできる。データは当月分および 4 か月以前までが保存してあり、2 か月ごとに再編成が必要である。

5.1.1 DMS 1100 データベース・エリアとレコード

ISIS の DMS 1100 は、インデックス・エリアと二つのダイナミック・エリア・レコード・プレースメント (DARP) の合計三つのエリアで構成されている。DARP エリアの一つには日付、ユーザ、資源レコードが入る (図 2)。

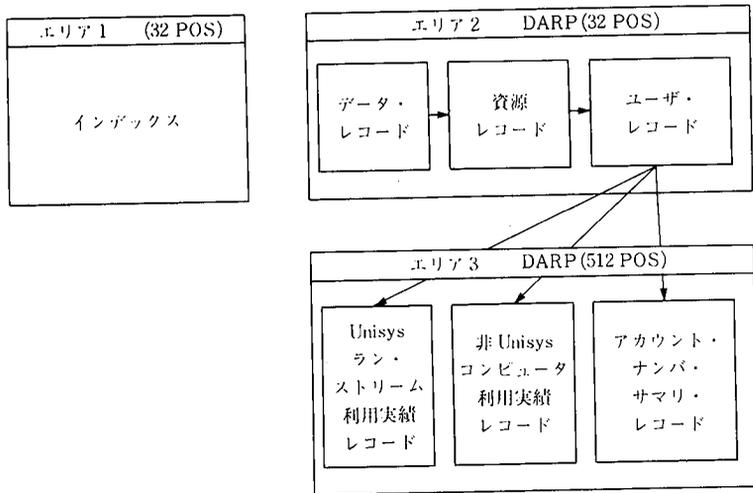


図 2 DMS 1100 データベースのレイアウト

Fig. 2 DMS 1100 database layout

その他の DARP エリアは Unisys 社製コンピュータのランストリーム使用レコード、Unisys 社以外のコンピュータの使用レコード、ユーザ・アカウント・ナンバのサマリ・レコードがそれぞれ入る。Unisys 社向けランストリーム・レコードにはユーザのアカウント・ナンバ、クォータ、サイト、プロジェクト、ランの各種識別子、開始/終了日時、SUP 時間、CPU 時間、コア・ブロック SUP 時間、ER 時間、コントロール・カード時間が入っている。また TIP トランザクション件数、実行プログラムの最大メモリ・サイズと名前、任意遅延時間、スループット時間、経過時間、見積もり実

行時間も含まれている。その他、マストレージ装置時間、通常シフトの識別コード、曜日、プライムまたはノンプライム・シフトの別 (5.2.2 で後述)、SPERRYLINK ラン、TIP ラン、MAPPER ランをデータ・レコードとして格納している。

Unisys 社以外の使用レコード (ASCII キャラクタで 112 文字長) には、コンピュータの識別子、ユーザのアカウント・ナンバ、シフト、開始/終了日時、経過時間、フレームまたはページ数が入る。

アカウント・ナンバ・サマリには ASCII キャラクタ 224 文字長で、アカウント・ナンバ別、コンピュータ資源別、日別の使用状況がまとめてある。このレコードを基にして特定アカウント・ナンバの使用量合計および、1 日のコンピュータ使用量合計が算出できる。このレコードにはアカウント・ナンバ、資源コード、開始/終了日、使用した資源の課金単位合計などの情報がある。実行ラン、デマンド・ラン、TIP ラン、MAPPER ランの合計実行回数も含まれている。さらに SUP 時間合計、デマンド SUP 時間、TIP SUP 時間、MAPPER SUP 時間、TIP トランザクション件数、カタログ済みマストレージ・トラック、可動型マストレージ・トラック数も入っている。

5.1.2 DMS 1100 データベース検索プログラム IFETCH

SCC のユーザは IFETCH と呼ぶプログラムを用いて、ISIS の DMS 1100 データベースをアクセスできる。このプログラムには四つの入力パラメータがある。まず日付パラメータで抽出したいデータの開始日と終了日を指定する。Unisys 社製コンピュータのランストリーム使用レコード、Unisys 社以外のコンピュータ使用レコード、あるいは資源レコード別アカウント・ナンバ・サマリのいずれか、または全部を抽出する指示を与えるタイプ・パラメータがある。Unisys の 1100/70 か 1100/90 か (あるいは両方か) も、マシン・パラメータで指定する。さらに 2, 3, 6 桁のアカウント・ナンバおよび 9 桁のパスワードからなるユーザ・パスワードをパスワード・パラメータによって指示する。

IFETCH の入力ファイルはすべて内部的に割り当てる。プログラムのマップ時に "Z" オプションを指定して、アクセス情報のダンプは作成しない。DMS 1100 データベースから IFETCH が抽出したデータは、USER-FILE という名前のファイルへ出力する。このファイルは、ユーザが独自のクォリファイアを指定して割り当てたものである。この USER-FILE は PCIOS ファイルで、COBOL, FORTRAN, QLP プログラムの入力ファイルとなる。レコードは、テキスト・エディタを用いて簡単に調べることができる。

5.1.3 DMS 1100 データベースの再編成

IFETCH プログラムおよび USER-FILE は、ISIS の DMS-1100 データベースの再編成作業でも利用する。まず保存すべきデータは IFETCH ランで 10 日間隔で抽出する。データベースはランを用いて、あるいは初期データベースを磁気テープからマストレージにコピーして再初期設定を行う。特別な更新プログラムを用いて、抽出してある USER-FILE をリードし、データベースを更新する。

5.2 ISIS の MAPPER 1100 データベース

このデータベースには SCC の全システム資源について、そのターミナル利用、デバイス別非稼働時間、および資源、ユーザ、ターミナルについての一般的利用情報の見

積もりデータと実績データが格納してある。このデータベースは SCC の一般ユーザはアクセスできない。偶数モード（リード/ライト可能）は、プログラミング要員だけがアクセスできるように設定してある。また奇数モード（リード専用）は、特定の SDC および NASA ユーザだけがアクセス可能である。

ユーザはマニュアル・ファンクションと MAPPER ランを用いて必要なデータを管理する。MAPPER ランはメニュー方式で対話形式の処理を可能にしている。一部の MAPPER ランはユーザの指示する情報を用いて OS 1100 のバッチ・ランをスタートさせる。このシステムには内部監査機能が組み込んであり、日次更新ランが正確に稼働しているかどうかを確認できる。

データベースは月末および年末ランを実行して自動再編成を行う。本番業務で使用するモードは二重化してあり、その一方をテスト・モードとしてランの開発やチェック、改造などに役立てている。データベース本体は六つのモードだけで構成してあり、各モードのうち、数個のタイプだけを使用している（図 3）。

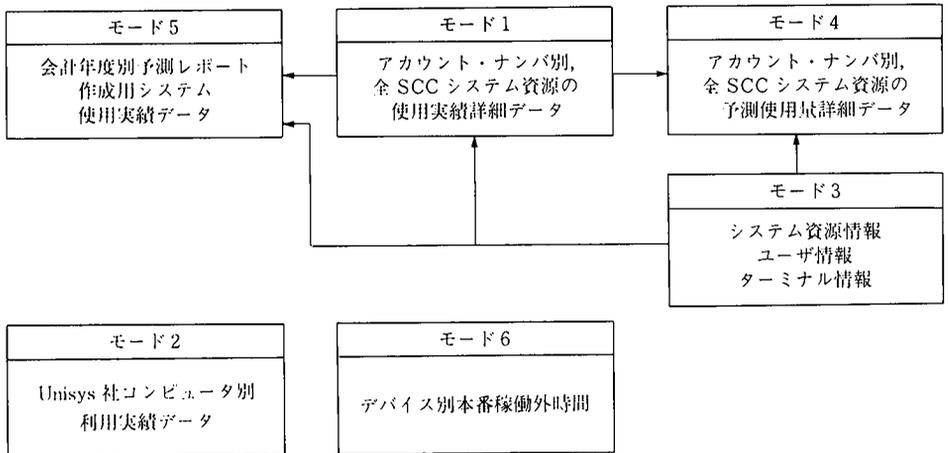


図 3 MAPPER データベース・レイアウト

Fig. 3 MAPPER database layout

5.2.1 MAPPER 1100 データベース モード 1

モード 1 には全 SCC システム資源の詳細な使用実績データが入る。一つのタイプにサイト識別子別、シフト別（プライム・シフトかノンプライム・シフトか）のターミナル利用実績を格納、その二つのタイプにはシステム資源別アカウント・ナンバごとの利用実績が入る。このターミナル利用実績には、過去 4 か月間の日別ターミナル利用実績と、過去 24 か月間の日次利用実績をまとめて月次データにしたものが格納してある。さらに RID 単位に日付、コンピュータ識別子、ターミナル・サイト識別子、シリアル・ナンバ、シフト、曜日、合計ラン、合計 SUP 時間、コネクト時間、合計 TIP 時間、合計 TIP トランザクション数、合計 MAPPER ラン数をそれぞれ格納している。照会ランを用いればサイト識別子別または回線番号別に使用実績レポートを得ることができる。また MAPPER ランでターミナル利用実績レポートを作成し、NASA および SCC ユーザ向けに配付している（図 4）。

DATE	MACH ID	TERML SITEID	SERIAL NUMBR	S D H W	TOT RUNS	TOTAL SUP HOURS	ELAPSED OR CONNECT TIME	TOTAL TIP RUNS	TOTAL TIP TRANSACTIONS	MAPPER RUNS
860201	110070	A131FA	63756	4 6	2	0.00157	0.12190	0	0	2
860201	110070	A160AS	39603	4 6	1	0.00426	0.00330	1	7	0
860201	110070	A163HS	39301	4 6	3	0.00482	0.00240	3	13	0
860201	110070	A164EA	33406	4 6	2	0.01469	1.37600	0	0	2
860201	110070	A164FA	39300	5 6	1	0.26329	1.42160	0	0	1
860201	110070	A23DBA	39541	5 6	1	0.00002	0.00000	1	1	0
860201	110070	A266GS	61910	5 6	1	0.01026	4.09550	1	1	0
860201	110070	A338EA	63765	5 6	3	0.04784	0.02420	3	117	0
860201	110090	ADM403	25140472	4 6	2	0.02954	0.12700	0	0	0
860201	110090	AHY101	CAB0002	4 6	2	0.19719	1.37970	0	0	0
860201	110090	A13CYA	RIB0001	4 6	4	0.41863	6.59770	0	0	0
860201	110090	A424AA	7117	5 6	1	0.01628	0.45390	0	0	0
860201	110090	A533AB	821	5 6	1	0.00282	0.01470	0	0	0

図4 プライム/ノンプライム・シフト別日次ターミナル利用実績サマリ RID 例
 Fig. 4 Daily summary of terminal usage by prime and non-prime shifts sample RID

システム資源の種類別アカウント・ナンバ別利用実績として、日次詳細データを4か月間、さらに暦年月および会計年月の両方で13か月間、それぞれ保存する。このデータはユーザ向け課金処理やその他の利用レポート処理へ入力する。この実績データは別モードへコピーして、システム資源利用の予実対比レポートを作成するのに用いる。臨時に必要なレポートは、MAPPERのマニュアル・ファンクションを用いて随時作成できる。

あるタイプには資源コード、ユーザ区分コード、ユーザ識別子、ユーザ・アカウント・ナンバ、会計年月、会計四半期、日付、合計使用単位、合計ラン実行数、会計年月末のレートなどがある。その他のタイプにはTIPランの数、MAPPERの利用回数、デマンド・ランの回数、デマンドの利用単位が保存してある。

5.2.2 MAPPER 1100 データベース モード2

Unisys社製コンピュータの使用実績データは、モード2の七つのタイプに格納している。この実績データはコンピュータ別、シフト別にまとめてあり、さらにバッチ、デマンド、TIP、MAPPER、およびユーザ別にランとSUPの使用状況を分類している。日次の詳細データは4か月、月次のサマリ・データは13か月、それぞれ保存するようになっている。MAPPERで作成した対話による照会プログラムMORNINGを用いて、ユーザが各タイプにアクセスできるようにしてある。また、ユーザはマニュアル・ファンクションによって実績データを利用している。

実績データは六つのシフト区分により分類されている。シフトは通常3シフトと仮3シフトがある。通常の第1シフトは00:00から08:00まで、第2シフトは08:00から16:00まで、第3シフトは16:00から24:00までとなっている。以降の第4から第6シフトは仮シフトである。第4シフトは07:00から19:00まででプライム・シフトと呼ぶ。また第5シフトは非プライム・シフトで00:00から07:00までと19:00から24:

00 までとなっている。第 6 シフトは終日シフトで、00:00 から 24:00 までである。

MAPPER の各タイプにあるレポートは、データ収集日、コンピュータ区分 (1100/70 または 1100/90)、シフト、および曜日別になっている。

最初のタイプにはコンピュータ別、シフト別の日次統計データ・サマリが格納してある。ここではラン数、SUP 時間、経過時間の各総合計があり、バッチ・ラン合計、バッチ SUP 時間、バッチ経過時間、デマンド・ラン、デマンド SUP 時間、およびデマンド経過時間に分類集計されている (図 5)。

第 2 のタイプには日次 TIP 使用実績がコンピュータ別、シフト別にまとめてある。特別なデータとしては、TIP ラン合計数、TIP SUP 時間、TIP 経過時間、TIP トランザクション数がある。また全部のラン、SUP 時間、経過時間の総合計も比較の基準とするため保存してある (図 6)。

DATE YRMODE	MACH ID	S.WEEK H DAY	TOTAL RUNS	.TOTAL .SUP HOURS	.TOTAL .ELAPSED HOURS	.BATCH .RUNS	.BATCH .SUP HOURS	.BATCH .ELAPSED HOURS	.DEMAND .RUNS	.DEMAND .SUP HOURS	.DEMAND .ELAPSED HOURS
860201	110070	1 6SAT	131	38.12026	99.4473	76	37.79197	89.9383	55	0.32829	9.5090
860201	110070	2 6SAT	287	25.78781	76.6108	37	21.41967	45.2460	250	4.36814	31.3648
860201	110070	3 6SAT	101	29.84795	92.2597	32	26.04760	62.2376	69	3.80035	30.0221
860201	110070	4 6SAT	365	34.59225	118.1965	60	27.48538	71.7968	305	7.10687	46.3997
860201	110070	5 6SAT	154	59.16377	150.1213	85	57.77386	125.6251	69	1.38991	24.4962
860201	110070	6 6SAT	519	93.75602	268.3178	145	85.25924	197.4219	374	8.49678	70.8959
860201	110090	1 6SAT	97	45.77467	18.8468	59	44.68589	12.0845	38	1.08878	6.7623
860201	110090	2 6SAT	270	37.36150	98.1678	129	24.98430	34.2002	141	12.37720	63.9676
860201	110090	3 6SAT	138	33.05330	77.9124	59	26.19789	45.1996	79	6.85541	32.7128
860201	110090	4 6SAT	353	57.47669	133.0892	153	40.67675	44.1143	200	16.79994	88.9749
860201	110090	5 6SAT	152	58.71278	61.8378	94	55.19133	47.3700	58	3.52145	14.4678
860201	110090	6 6SAT	505	116.18947	194.9270	247	95.86808	91.4843	258	20.32139	103.4427

図 5 コンピュータ別、シフト別、日次統計サマリ・データ RID 例
Fig. 5 Summary daily statistical data by machine by shift sample RID

*DATE *YRMODE	MACH ID	S.WEEK H DAY	TIP RUNS	.TIP .SUP HOURS	.TIP .ELAPSED HOURS	.TIP .TRANSACTIONS	.TOTAL .RUNS	.TOTAL .SUP HOURS	.TOTAL .ELAPSED HOURS
860201	110070	1 6SAT	42	0.10031	4.1461	214	131	38.12026	99.4473
860201	110070	2 6SAT	195	3.15649	2.8254	6074	287	25.78781	76.6108
860201	110070	3 6SAT	41	0.47693	9.2724	1906	101	29.84795	92.2597
860201	110070	4 6SAT	225	3.29037	2.9018	6435	365	34.59225	118.1965
860201	110070	5 6SAT	53	0.44336	13.3421	1759	154	59.16377	150.1213
860201	110070	6 6SAT	278	3.73373	16.2439	3194	519	93.75602	268.3178
860201	110090	1 6SAT	0	0.00000	0.0000	0	97	45.77467	18.8468
860201	110090	2 6SAT	22	0.06745	0.0261	169	270	37.36150	98.1678
860201	110090	3 6SAT	9	0.22911	0.1215	3481	138	33.05330	77.9124
860201	110090	4 6SAT	27	0.08537	0.0738	207	353	57.47669	133.0892
860201	110090	5 6SAT	4	0.21119	0.0738	3443	152	58.71278	61.8378
860201	110090	6 6SAT	31	0.29656	0.1476	3650	505	116.18947	194.9270

図 6 コンピュータ別、シフト別、日次 TIP 利用実績サマリ RID 例
Fig. 6 Summary daily TIP usage by machine by shift sample RID

DATE	MACHINE	S	DAY	AVG	AVG	AVG	H-O-L-D-F-L-A-G-S													SUPS									
							YR	MODA	ID	H-OF	EXPOOL	USER	USER	J-B	T-M	D-X	O-S	T-RUNS	BATC	DEMN	UN-	TIP	TIP	TIP	ACCUMULATED	BATC	DEMN	TIP	
				F-WEEK	AVAIL	CORE	D-A	E-S	E-P	P-E	P-NOT	RUNS	RUNS	OPEN	PROC	AC-	WAIT	SINCE				SPED	SPED	SPED					
				T	AVAIL	IN USE	L-T	R-T	M-L	R-Q	E-CORE	OPEN	OPEN	RUNS	GOSE	TIVE	QUE	LAST	BOOT										
860201	110070	1	6SAT	42.2	1527.7	1379.3	0	0	0	1	0	0	1	0	0	1	0	1	0	12	1	13	3	1	1	36.4153	4.9	0.0	0.1
860201	110070	2	6SAT	44.2	1002.6	1224.3	0	0	0	0	0	0	0	1	0	0	0	0	10	1	12	11	0	0	25.4492	3.0	0.0	0.4	
860201	110070	3	6SAT	40.2	1740.3	1059.3	0	0	0	1	0	0	1	0	0	0	0	0	8	-0	7	1	1	1	22.0311	2.9	0.1	0.0	
860201	110070	4	6SAT	42.9	1187.0	1195.0	0	0	0	1	0	0	1	0	0	0	0	0	10	0	11	8	0	0	39.2297	3.2	0.1	0.3	
860201	110070	5	6SAT	41.5	1660.1	1247.5	0	0	0	1	0	0	1	0	1	0	1	0	10	0	11	2	1	1	83.9414	4.0	0.0	0.0	
860201	110070	6	6SAT	42.2	1423.4	1221.2	0	0	0	1	0	0	1	0	1	0	1	0	10	0	11	5	1	1	84.0219	3.6	0.0	0.2	
860201	110090	1	6SAT	82.5	6325.7	1742.4	0	1	0	1	0	0	1	0	1	0	1	0	8	1	14	1	1	0	48.5786	6.4	0.1	0.0	
860201	110090	2	6SAT	85.1	6084.7	1983.8	0	0	0	1	0	0	1	0	1	0	1	0	10	9	23	4	2	0	37.6900	3.2	1.7	0.0	
860201	110090	3	6SAT	83.6	6304.5	1765.2	0	1	0	0	0	0	1	0	1	0	1	0	8	4	10	2	2	0	30.0108	3.2	0.8	0.0	
860201	110090	4	6SAT	84.6	6115.9	1952.9	0	1	0	1	0	0	1	0	1	0	1	0	10	8	21	4	2	0	59.5044	3.7	1.5	0.0	
860201	110090	5	6SAT	82.8	6360.7	1708.0	0	1	0	1	0	0	1	0	1	0	1	0	8	2	11	1	1	0	116.4658	4.9	0.2	0.0	
860201	110090	6	6SAT	83.7	6238.3	1830.4	0	1	0	1	0	0	1	0	1	0	1	0	9	5	16	3	2	0	116.4764	4.3	0.9	0.0	

図7 コンピュータ別、シフト別、タイプ40データ・サマリRID例
Fig. 7 Summary type 40 data by machine by shift sample RID

深夜に作成するマスタ・ファイル・ディレクトリのダンプにある，固定マストレージおよび可動ストレージ上のファイルのトラック総合計数を，第3のタイプへ記録している。

第4のタイプにはOS 1100のアカウントティング・ログ・ファイル，タイプ40のレコードを独自に収集し統計データとしたものを保存している。このレコードはコンピュータのアクティビティを65ないし66秒間隔でサンプリングしたデータであり，これをシフト別に平均化してある。たとえば，使用可能なEXPOOLサイズ，ユーザ用に割り当て可能なメモリ・サイズ，使用中のユーザ・メモリ，メモリ中不在ランの数，オープンされているバッチ・ランとデマンド・ランの数，未オープンのラン数などが，この情報から得られる。またTIPの処理情報として，最近60秒間に処理したTIPトランザクション数，アクティブなTIPトランザクション数，ウェイト・キュー上にあるTIPトランザクションなどを知ることができる。さらにレポートとしては最新ブート以後のSUP合計値，バッチ/デマンド/TIPの各処理スピードをまとめることができる。加えてアイドル状態，バッチ，ターミナル，マストレージ，デマンド，EXPOOL，操作状態，実行順序，テープ・ホールドなどの各種フラグも保存されている(図7)。

第5のタイプは，アカウントティング・ログ・ファイルから得たタイプ40データの続きが入っており，それにテープ・ドライブ数，およびマストレージの使用可能トラック数をデバイス別に分類して保存してある。

第6のタイプにはUnisys社製コンピュータの使用実績状況を，シフト1, 2, 3, 6で分類し，さらにユーザ名別にして保存している。データとしては日付，曜日，ユーザ名，合計ラン数，および合計SUP時間がシフト別に分類してある。またオーバーヘッドとなったユーザの利用状況も，利用した機能別に分類してある。

MAPPERの使用実績は第7のタイプに保存している。一般的な情報に加えて，MAPPERランの合計数，MAPPER SUP合計時間，MAPPER経過時間の合計が得

られる。また比較の基準データとして、全ランの合計数、SUP 時間数、経過時間も保存されている (図 8)。

DATE YRMODA	MACH ID	S H	WEEK DAY	MAPPER RUNS	MAPPER SUP HOURS	MAPPER ELAPSED HOURS	TOTAL RUNS	TOTAL SUP HOURS	TOTAL ELAPSED HOURS
860201	110070	1	6SAT	4	0.06316	0.9651	131	38.12026	99.4473
860201	110070	2	6SAT	39	0.94694	24.1411	287	25.78781	76.6108
860201	110070	3	6SAT	21	2.76923	19.1812	101	29.84795	92.2597
860201	110070	4	6SAT	57	2.99760	37.5311	365	34.59225	118.1965
860201	110070	5	6SAT	7	0.78173	6.7563	154	59.16377	150.1213
860201	110070	6	6SAT	64	3.77933	44.2874	519	93.75602	268.3178
860201	110090	1	6SAT	5	0.42473	0.3743	97	45.77467	18.8468
860201	110090	2	6SAT	0	0.00000	0.0000	270	37.36150	98.1678
860201	110090	3	6SAT	1	0.00721	0.0000	138	33.05330	77.9124
860201	110090	4	6SAT	1	0.01824	0.0000	353	57.47669	133.0892
860201	110090	5	6SAT	5	0.41370	0.3743	152	58.71278	61.8378
860201	110090	6	6SAT	6	0.43194	0.3743	505	116.18947	194.9270

図 8 コンピュータ別、シフト別、日次 MAPPER 利用実績サマリ RID 例

Fig. 8 Summary MAPPER usage by machine by shift sample RID

5.2.3 MAPPER 1100 データベース モード 3

モード 3 は、SCC における利用可能なシステム資源についての一般情報と、SCC のユーザ、および各ターミナル・デバイスに関する情報が保存してある。このモードにもアカウント・ログ・ファイルから収集したタイプ 40 の詳細データが入っている。以上のデータは MAPPER のマニュアル・ファンクション、および対話型ランでアクセスすることができる。このモードでは六つのタイプを使用している。システム資源とユーザに関する情報は、ユーザへの課金、各種資源の見積もり、および利用状況に関するレポート作成に徹底的に使用している。またターミナル情報は、ユーザ別ターミナル使用状況のレポートに使用するとともに、ターミナルの機器管理にも用いている。

第 1 のタイプはシステム資源の全体についての情報を永久保存するために用いられている。新たな資源の追加や評価レートの変更は随時、可能になっている。ここにあるシステム資源が削除できるのは、会計年度の開始時期に限定されている。システム資源コード、識別名、説明、課金単位、当年度および今後 5 年間の評価レートがこの RID に格納してある (図 9)。

第 2 のタイプにはユーザ情報が入っており、アカウント・ナンバの 2 桁目、3 桁目、および 6 桁目で区別するユーザ識別名、プロジェクト、NASA アカウント・ナンバ、分類コード、分類名が保存されている。新規ユーザは随時追加登録できるが、ユーザの削除は会計年度の開始時期だけに限られている。

ターミナル情報は二つのタイプに保存してある。新ターミナル・サイト ID は必要に応じて追加している。削除したターミナル情報は履歴 RID へ入れて、レポート作成に使用する。このタイプには、サイト ID、ターミナル所在地、ビル番号、部屋番号、シ

RES CDE	RESOURCE ID	DESCRIPTION	RESOURCE UNITS	*** FISCAL YEAR BILLING RATES **					
				CURRENT FY	CURRENT FY+1	CURRENT FY2	CURRENT FY+3	CURRENT FY+4	CURRENT FY+5
1	110070	SPERRY 1100/70	SUP HOUR	0.000	0.000	0.000	0.000	0.000	0.000
8	KDAK40	NON-GRAPHICS COM KOMSTAR	P FRAME	0.000	0.000	0.000	0.000	0.000	0.000
9	FR8037	GRAPHICS COM FR-80	P FRAME	0.000	0.000	0.000	0.000	0.000	0.000
10	XERO36	COM COPY	PER COPY	0.000	0.000	0.000	0.000	0.000	0.000
12	ADCS98	ADCS DATA REDUCTION	HOUR	10.000	15.000	20.000	25.000	30.000	35.000
13	ADRS97 0	ADRS DATA REDUCTION	HOUR	10.000	15.000	20.000	25.000	30.000	35.000
14	ADRS97 1	ADRS FREQUENCY ANALYSIS	HOUR	10.000	15.000	20.000	25.000	30.000	35.000
15	1100	110070 CATALOGED MASS ST	MAX TRKS	0.000	0.000	0.000	0.000	0.000	0.000
16	1100	110070 REMOVABLE MASS ST	MAX TRKS	0.000	0.000	0.000	0.000	0.000	0.000
22	110090	SPERRY 1100/90	SUP HOUR	10.000	15.000	20.000	25.000	30.000	35.000
25	1100	110090 CATALOGED MASS ST	MAX TRKS	0.000	0.000	0.000	0.000	0.000	0.000
26	1100	110090 REMOVABLE MASS ST	MAX TRKS	0.000	0.000	0.000	0.000	0.000	0.000
30	1100	NASTRAN	CPU HOUR	0.000	0.000	0.000	0.000	0.000	0.000
31				0.000	0.000	0.000	0.000	0.000	0.000
32		DATA REDUCTION	HOUR	0.000	0.000	0.000	0.000	0.000	0.000
36		ACTIVE TAPE STORAGE	REEL DAY	0.010	0.015	0.020	0.025	0.030	0.035
37		INACTIVE TAPE STORAGE	REEL DAY	0.010	0.015	0.020	0.025	0.030	0.035
38		TAPE STAGING	EACH	1.000	1.000	1.000	1.000	1.000	1.000
39		TAPE PURCHASE	EACH	1.000	1.000	1.000	1.000	1.000	1.000
40		OFFLINE LASER COPY	PER COPY	0.010	0.015	0.020	0.025	0.030	0.035

図9 システム資源情報 RID 例

Fig.9 Resource information sample RID

リアル・ナンバ、回線速度 (Kbit/秒)、セット・タイプ、回線番号の各情報が入る。また NASA の資産タグ番号、設置済みまたは予備コード、PID ナンバ、旧サイト ID (変更時)、および Unisys 社製ホスト指示子を保存することもできるように考慮してある。またターミナルの用途がデマンド、MAPPER、バッチ、あるいはオフィス・オートメーション用かがわかるフラグも保存してある。各ターミナルについて、コメントを記述するスペースも用意してある (図 10)。

5 番目と 6 番目のタイプには、アカウント・ログファイルから収集した詳細なタイプ 40 データを 38 日間保存している。同種データはモード 2 に、シフト別平均値として保存してある。このタイプでは、各エントリをログに採取した時刻を真夜中からの経過秒数で記録している点が違っている。

5.2.4 MAPPER 1100 データベース モード 4

全 SCC のシステム資源に関する予測使用量の詳細情報を格納するモードである。データは MAPPER の対話型ランで更新し、また MAPPER ランとマニュアル・ファンクションを併用してレポートを作成している。前年度のユーザ別使用実績データは、システム資源コードとアカウント・ナンバ別に一つのタイプ内に保存してある。ユーザ別の当年度予想システム使用量は、システム資源コード別、アカウント・ナンバ別に別のタイプに記録されている。向こう 5 年間についてのユーザ別、システム資源コード別、アカウント・ナンバ別予想使用量は、さらに別タイプを用意して保存してある。各ユーザには各タイプに専用 RID を割り当ててあり、その RID にシステム資源コ

SITE ID	TERMINAL LOCATION	BUILDING NUMBER	ROOM NUMBER	PRINCIPAL USER	TELEPHONE NO	DEVICE	SERIAL NUMBER	RATE (KBS)	SET TYPE	CIRCUIT NUMBER	S						
											TAG NO.	HOST	D	T	M	B	O
XDM101	BFC	1001	D/F7	USER A	-888-2222	ADM-22	FFF40570	9.6	GDC	PP001/2	997660	I					
*XDM101																	1170
*XDM101																	1190 X
XDM102	BFC	1002	D/F7	USER B	-888-3333	ADM-22	FFF40146	9600	2096	PP001/02	997665	I					
*XDM102																	1170
*XDM102																	1190 X
XDM103	BFC	1002	R/S9	USER B	-888-4444	ADM-22	FFF40346	9600	2096	PP001/02	997662	I					
*XDM103																	1170
*XDM103																	1190 X
XDM104	BCF	1002	P/R9	USER B	-888-5555	ADM-22	FFF40568	9600	GDC	PP001/02	997705	I					
*XDM104																	1170
*XDM104																	1190 X
XDM105	BCF	1001	H7	USER A	-888-6666	ADM-22	FFF40275	9600	2096	PP001/02	997673	I					
*XDM105																	1170
*XDM105																	1190 X

図 10 ターミナル情報データベース RID 例
 Fig. 10 Terminal information database sample RID

ード, アカウント・ナンバ, 第 1 四半期から第 4 四半期までの資源使用予想対実績値, システム資源の説明, 課金単位がそれぞれ含まれている。

5.2.5 MAPPER 1100 データベース モード 5

各ユーザごとのシステム使用実績データを格納するモードで, 会計年度別の予測レポート作成に使用する。この実績データは暦年月および会計年月の両方について, 各月別に分類してある。実績データは使用量と使用金額でシステム資源コード別, アカウント・ナンバまたはユーザ ID 別で把握できるようになっている。各ユーザごとに 8 タイプのそれぞれについて RID を一つ割り当ててある。四つのタイプで暦年月または会計年月別, アカウント・ナンバまたはユーザ ID 別の使用実績データを保存している。その他の 4 タイプに各区分の使用金額が格納してある。

5.2.6 MAPPER 1100 データベース モード 6

本番稼働以外の時間に関するデータを保存するモードである。デバイス別の本番稼働外時間は 24 か月間保存し, MAPPER ランおよびマニュアル・ファンクションの両方でアクセス可能になっている。本番稼働外時間コードは一つのタイプにあり, デバイス・シリアル・ナンバは別タイプに保存してある。3 番目のタイプには, デバイス名, 本番稼働外時間コード, 開始日時, 終了日時, 合計時間, メインフレーム・コード, 障害の説明, ダンプ採取リール番号, シリアル・ナンバ, ダウン時間区分, コンポーネントの説明がそれぞれ保存してある。このデータはデバイスのパフォーマンスをモニタし, ダウン時間実績からコンピュータ・メーカーの信頼度を把握するために用いる。また障害コールの対応時間を評価するためにも使用している。

5.2.7 MAPPER 1100 データベースのメニュー

MAPPER の本番稼働ランおよびそこで使用中のモードは, 対話メニュー・ランを用いてアクセスできる。このメニューはユーザが MAPPER ランを作成したりデータの

一覧ができるようになるための教育機能を内蔵している。このメニューを一覧するだけで、ISIS の全機能がわかるようになっている。複数の人々がラン開発に参加した場合は、そのランを実行した日時、および誰が実行したかを記録し、メニューで表示できるようにしてある。

```

YOU ARE ENTERING THE SCC ISIS MENU OF PRODUCTION RUNS PROCESS
NASA    SLIDELL    COMPUTER    COMPLEX
        (TAB TO THE DESIRED ACTION AND XMIT.)
MORNING -- MENU OF MORNING REPORTS (TIP,MAPPER,STATISTICAL,ETC.)
MAPR-UPD -- MENU OF RUNS TO UPDATE UTILIZATION MAPPER DATA BASE
2G-INPUT -- RUN TO PROVIDE SCREEN ENTRY OF NON-SPERRY DATA
ISIS-MONTHLY-- MENU OF END-OF-MONTH UTILIZATION RUNS
USER-BILLING-- MENU OF RUNS ASSOCIATED WITH USER BILLING
PROJECTIONS -- MENU OF RUNS ASSOCIATED WITH SCC WORKLOAD PROJECTIONS
TIDE     -- MENU OF RUNS TO MAINTAIN TERMINAL INFORMATION DATA BASE
TERMINAL -- MENU OF RUNS TO PRODUCE TERMINAL UTILIZATION REPORTS
TYPE-40  -- MENU OF RUNS TO PRODUCE TYPE 40 STATISTICAL RECORD REPORTS
NONPRD   -- MENU OF NONPRODUCTION TIME DATA AND RUNS
DEV-WK-PRFM -- RUN TO PRODUCE TECHNIQUES MEETING OPERATING STATISTICS
XMIT HERE FOR TUTORIAL SCREEN FOR ISIS RUNS
XMIT HERE TO END RUN
    
```

図 11 MAPPER データベースのメイン・メニュー・スクリーン

Fig. 11 MAPPER database main menu screen

メインとなるメニュー・ランは ISIS と呼び、その他の全メニューへアクセスする入口となる。MORNING メニューは一日の朝に使用するもので、日次統計データ、TIP、MAPPER、タイプ 40 に関するレポートなどが得られる。MAPR-UPD メニューは MAPPER データベースを対象に毎日実行する更新ランを表示する。また毎月末に実行するランは ISIS-MONTHLY メニューでアクセスできる。ユーザの課金処理をする上で必要なランは、すべて USER-BILLING メニューからアクセスする。さらに、PROJECTIONS メニューから予測データに関するランを実行することができる。

一般ターミナル情報を管理するランは、TIDE メニューから選択する。ターミナル使用状況レポートは、TERMINAL メニューから作成している。TYPE-40 メニューは、タイプ 40 の統計レポートを作成するランをアクセスする時に使用する。最後に NONPRD メニューを用いて、本番稼働外時間のデータをアクセスしレポート作成ができる。

6. おわりに

本システムは本番稼働を開始してから、ほぼ 19 か月が経過している。システム資源の予測業務は、使用実績に関する業務開始後 6 か月を経るまで完全には稼働しなかった。また、あらゆるシステムがそうであるように、本システムも新たな要求により変更が加えられる。本システムはシステム資源管理、資源計画、ユーザへの課金、Unisys 社製コンピュータに関する統計値の収集、ターミナル使用状況の把握、およびエンド・

ユーザが直接、本番稼働外時間について情報アクセスできることを基本目標としている。この当初の目標は完全に達成した。このシステムは COBOL プログラムをフロント・エンドとし、DMS 1100 データベースに DML COBOL プログラムを併用、さらに MAPPER データベースに MAPPER ランとマニュアル・ファンクションを利用して、初めて実現が可能になったものである。

(統合 OA システム部 横山 正敏 訳)

執筆者紹介 Martha Washington Mickelborough

Lousiana 州の Natchitoches の Northwestern 州立大学の理学部卒業。1962 年以來 NASA の宇宙計画にシステム・アナリストおよびプログラマとして参加。

Chrysler Corp. の宇宙部門を経て、現在は Unisys に勤務 (旧社名は Burroughs Corp. の子会社である System Development Corp.)。



論文

トランザクション処理システムへの MAPPER の適用

Transaction Processing System Using MAPPER 1100

小 西 徳 雄

要 約 MAPPER 1100 は、発表以来、適用業務を非定形の分野から定形の分野へと拡大してきている。しかしながら、MAPPER は第4世代言語の特長の一つであるシステム開発、および保守の容易性にもかかわらず、トランザクション処理システムへの適用には今一步ユーザがちゅうちょしているのが現状である。

トランザクション処理システムは、利用者から見ると一定時間内の応答が求められ、またシステム部門からは、円滑なバッチ業務および他のシステム・インタフェース業務の実行が求められる。この2点から、従来、トランザクション処理システムは、DB/DC を中心としたソフトウェア体系のもとで開発が行われてきた。

このたび、トランザクション処理システムに MAPPER 1100 を適用する機会が得られたので紹介する。適用に当たって留意した点は、業務の分析にもとづいたレポート設計である。具体的には、レポート行数を一定数以内に設定するために、① RID 分割の手法の適用、② データの生成から消滅に至る過程での使用目的（検索/更新）に応じたレポート間のデータの移動、を実施した。これにより初期の目的を果たすことができた。

Abstract MAPPER 1100 has been expanded in its use from nonroutine application to routine application since its announcement. But in spite of its merit in ease of system development and system maintenance, use of MAPPER is considerably limited in implementation of a transaction processing system.

As a transaction processing system is required a response time within designated time, and smooth data transfer to batch programs, it has been developed under traditional database management and data communication software, not using MAPPER.

The author had an opportunity to implement a transaction processing application using MAPPER. The paper introduces implementation techniques focusing the report design resulted from an analysis of application process. The author adopts RID partitioning for conforming the limit of report lines, and data moving between reports to improve the performance of updation.

1. はじめに

MAPPER 1100 は、昭和56年発売以来、非定形業務を中心とした分野に適用され現在に至っている。MAPPER は開発効率の高さにもかかわらず、トランザクション処理を中心とした定形業務への適用には、処理効率の面で制約がある。適用システムの特徴を把握し、システム設計の中心となるデータベース(DB)設計に工夫を加えることで、MAPPER のトランザクション・システムへの適用が可能であることを実証する機会を得たので紹介する。

2. システム概要

今回開発したシステムは、工事工程を管理するシステム（以下、工事システム）であり、顧客からの工事申し込みに始まり、工事を施行し、精算を行うまでに発生する

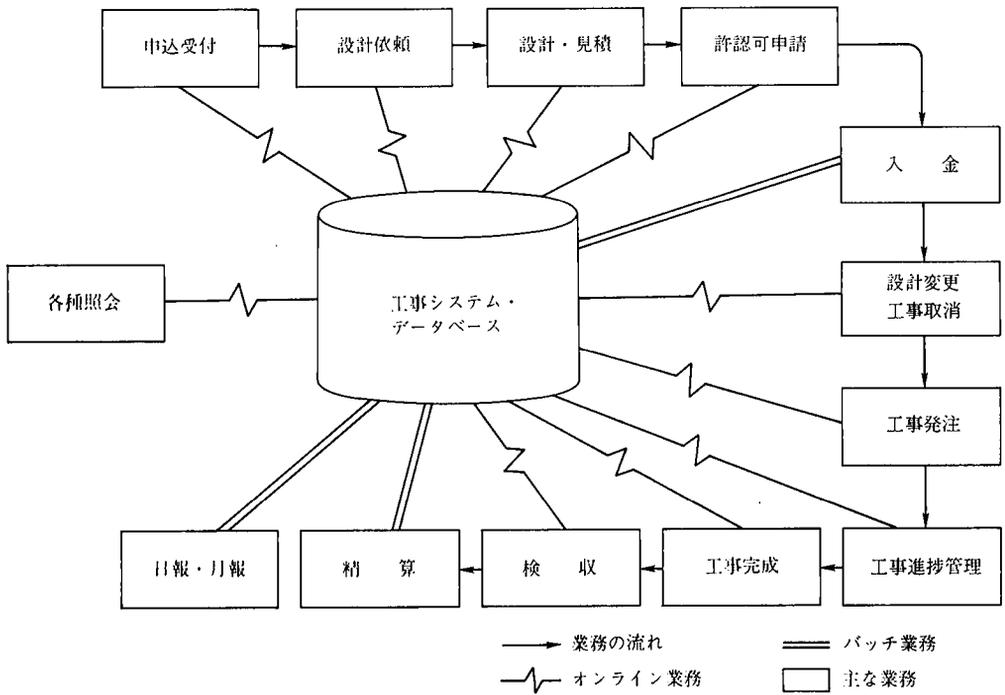


図1 システム概略

Fig.1 System structure

表1 システム規模

Table 1 System's scale

分類	単位	
平均申込件数	2,000 件/月	
平均工事施工件数	1,700 件/月	
平均滞留期間	9 か月/件	
管理項目数	200 項目/件	
システム開発	MAPPER ラン	100 本 (2 万 1 千ステップ)
	COBOL 新規	84 本 (6 万 7 千ステップ)
	COBOL 変更	30 本 (3 万 1 千ステップ)

トランザクションを処理するシステムである。

図1にシステムの概略を示す。工事データは、申込受付よりシステムに投入され、工事完成し精算処理後、一定期間（6か月）経過すると削除される。システムの規模を表1に示す。

システムの特徴は、次の通りである。

- 1) 工事の進捗に応じてデータを投入していく、トランザクション処理型のシステムである。
- 2) 入金処理と精算処理を中心としたバッチ業務と密接に連動したシステムである。トランザクションは、オンライン終了後にバッチ・システムに渡され、その

処理結果は翌日のオンラインに返される。

- 3) 利用者は、工事関連の営業部門および工事会社であり、不特定多数の顧客は直接利用しない。また、利用端末数も 20 数台と限定されている。
- 4) 取り扱うデータの処理形態は即時であるが、データは工事の結果または予定であり、予約システムのような即時データではない。

また、どのようなシステムでも言えるが、DB 管理ソフトウェアを決定する要件がポイントになる。

本システムの DB で求められるのは、次の事柄である。

- 1) 即答型の処理に耐えられる……トランザクションの処理で一定時間の応答が保証されること。
- 2) システム開発が容易である……同一機能を実現できるならば、安く、高品質でありかつ短納期のものが望ましい。
- 3) システムの保守が容易である……工程管理システムとして求められる機能は、端末数が少なくても同じため、本事例では利用者当たりのコストが割高となる。このことは、保守コストが全コストの 8 割を占める時代では、保守がより一層容易でなければならないことを意味する。
- 4) 拡張性があること……不特定多数の利用者のシステムではなく、職務に通じた人達が利用するシステムである。したがって、将来発生するニーズに利用部門自からが対応可能なシステムであること、

などであるが、とくに、1) の即答型の処理に耐えられることは、ある面で 2)～4) と相反する要望といえる。

3. なぜ MAPPER DB か

MAPPER は、COBOL、FORTRAN で代表される第 3 世代言語ではなく、第 4 世代言語 (4th Generation Language ; 4GL) と呼ばれている。4GL の定義は、James Martin によれば、次に示す機能を満足している言語と言える。

- 1) ユーザ・フレンドリである。
- 2) 専門家でないプログラマでも結果を得られる。
- 3) データベース管理システムを直接使用する。
- 4) COBOL の 10 分の 1 の命令数で、手続き型の場合は結果が得られる。
- 5) できる限り非手続き型コードを使用している。
- 6) レポート形式の選択などを自動的に行うことができる。
- 7) オンライン・オペレーション用として設計されている。
- 8) 構造化プログラミングが容易にできる。
- 9) 他人の書いたプログラムを容易に理解し、修正できる。
- 10) 2 日間のコースで、一般的ユーザが言語のサブセットを習得できる。
- 11) 容易にデバッグできるように設計されている。
- 12) COBOL や PL/I の場合の 10 分の 1 以下の時間で結果が得られる。

MAPPER は一部を満足していないが(たとえば、構造化プログラミングが容易にできる)、4GL にふさわしい言語と言える。MAPPER は、その特質から見て、エンド・

表2 オンライン・リアルタイム・システム構築の比較

Table 2 Comparison between DB/DC and MAPPER in system construction

	DB/DCによるシステム構築	MAPPERによるシステム構築
適 用 業 務	① 定型大量処理業務 ② 応答時間は数秒を保証	① 非定型小量処理業務 ② 応答時間は厳しくない。
設 計	① 長期間をかけて設計	① プロトタイプング手法による設計が可能
開 発	① 長期間必要 ② 工数大 ③ 開発ツールが必要 ④ デバッグの工数大 ⑤ 幅広い知識が必要	① 比較的短期で可能 ② 工数小 ③ 開発ツールは不要 ④ デバッグは容易 ⑤ MAPPER の知識で十分
DB	① 他システムのDB (DMS, COBOL) との接続が容易 ② 保守・運用コスト大 ③ 大容量 DB が構築できる	① 他システム DB とのインタフェースはソフトウェアを介在させる。 ② 保守・運用コスト小 ③ 大容量 DB には比較的向き
DC	① ソフトウェアが独立しており、知識が必要	① MAPPER の機能に含まれている。
保 守	① 変更しにくい。 ② コスト大	① 容易に変更可能 ② コスト小

ユーザ指向であり、エンド・ユーザによるエンド・ユーザのためのシステム構築用言語である。その点、オンライン・リアルタイム・システムで代表される従来のDB/DCシステムは、システム部門によるエンド・ユーザのためのシステムと言える。

MAPPERの持つ別の側面、つまりシステム開発と保守の容易性に着眼した適用もまた、その重大な使命と言える。

以上の観点で、従来のDB/DCによるオンライン・リアルタイム・システム構築とMAPPERによる構築を比較すると表2のように整理できる。表2では、利用者側から見れば応答の即時性が保証できるかが、またシステム部門から見ればDBが孤立しないかがポイントとなる。

4. MAPPER DB適用の方針

トランザクション処理システムのDB管理ソフトウェアにMAPPERを適用するに当たって、以下の方針で臨んだ。

- 1) 効率に最大限の配慮をすること。
- 2) 企業内の他システムとのインタフェースを持ち、かつ容易にデータ交換ができること。

4.1 効率に関する考慮

MAPPERは4GLの特長を備えているが、DB(レポート)作成には、以下の制約および特質がある。

- 1) レポートの桁数は132文字以内である。
- 2) レポート行数が増えれば、効率は悪くなる。
- 3) 同時更新可能レポート数は最大5レポートまでである。
- 4) ロックの単位はレポート単位である。
- 5) レポート行数を変更する命令は、レポート全体の更新となる。

これらの制約は、いずれも処理効率に多大の影響を与える。

この影響を最小限にするために、次のようなレポート設計を実施した。

- 1) RID 分割によるレポート行数の適性化
- 2) レポート行の拡大ファンクションの実施をすべてのトランザクションで均等に負う。
- 3) 工事工程に応じた情報のグループ化によるレポート設計
- 4) 検索専用レポートの設定

4.1.1 RID 分割

本システムの MAPPER DB では、1 レポート内の行数を 500 以内におさえるため、および同時に同一レポートを更新する際に発生するロックの待ちを極力なくするために、RID 分割の手法によって各レポートを分割している。

RID 分割のキーは支社コードと工事番号であり、マスタ・レポート・インデックスを支社コードで検索し、インデックス内に設定したモジュラス値で工事番号を割り算し、その余りで当該工事物件の RID 番号を得る方式を採っている (図 2)。

この RID 番号を得るロジックは、すべてのトランザクション処理の最初でサブルーチンをコールすることで行われており、このためテスト DB や教育用 DB の構築はプログラムを変更することなく可能である。

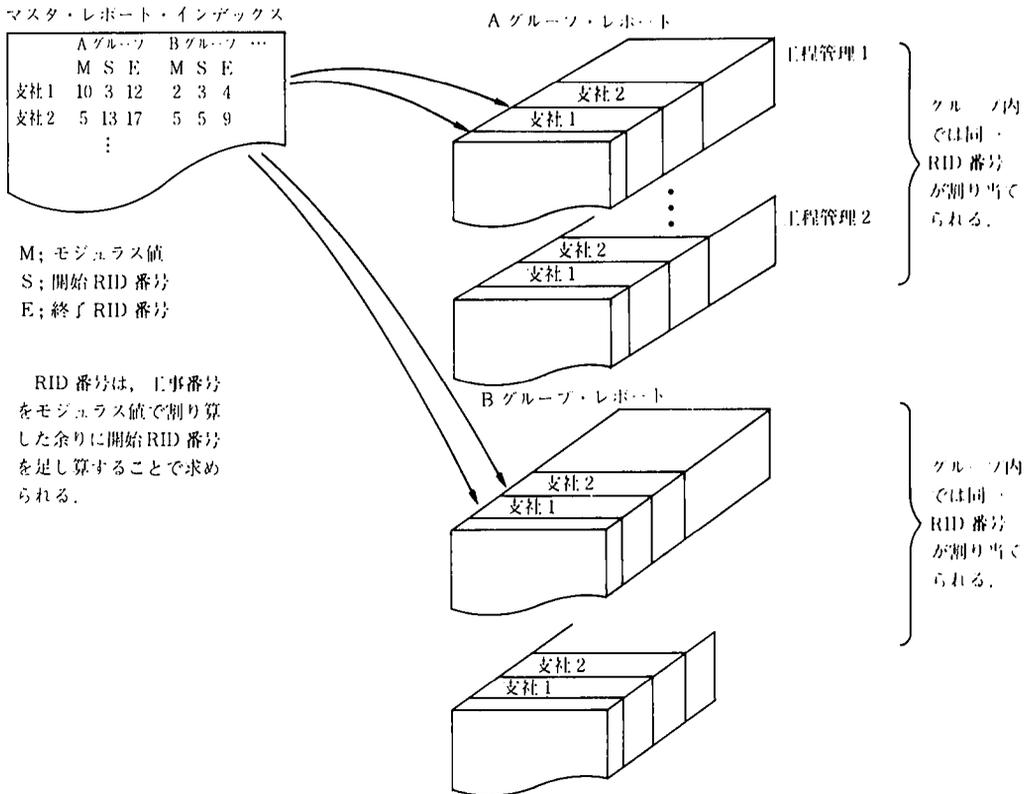


図 2 マスタ・レポート・インデックスと RID 分割

Fig. 2 Relation between master report index and RID

工事番号が連番であるため、あらかじめ必要な行数をオンライン開始前に生成して、オンライン中のトランザクション更新では、行のリライト (@WRL) のみ行う方式も考えられるが、

- 1) 会話機能で不要なデータが検索対象となる、
- 2) 帳表作成プログラムなどのレポート処理では効率が悪くなる、
- 3) 空き管理を日々実施しなければならない、

との理由で採用していない。

4.1.2 負荷の均等化

システムに投入されるトランザクションが、参照および更新するレポート数を平準化する必要がある。このために、各レポートのキー項目のみで構成される工程管理レ

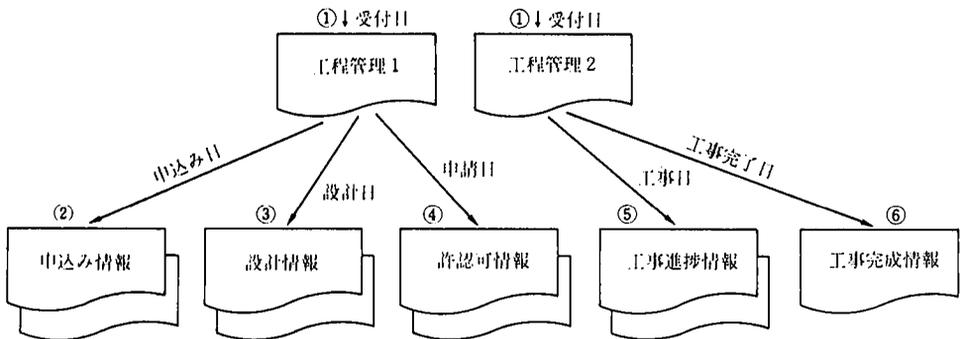


図3 工程管理レポート

Fig. 3 Construction process control report

ポートを2種類設定した(図3)。なお、図中の矢印と番号は、レポート生成のタイミングと順序を表している。このため、たとえば工程管理1では、申込み日、設計日、申請日を各レポートと重複して持つことになる。

これにより、工程管理レポートを参照することで、工事物件が現在どの工程にあるか、またどのレポートが作成されているかが明らかになった。また、工事の全容を把握するのに、二つのレポートのみの検索ですませることができた。

4.1.3 検索専用レポートとレポート行数の削減

工事物件は、工事工程に応じて、種々の状態でレポート内に存在しうる。

- 1) 申込みされても工事を実施しない物件
- 2) 工事を実施している物件
- 3) 工事が完了している物件

このうち、1)は6か月経過後レポートから削除され、3)は照会時のみ使用される。トランザクション処理で更新対象となるのは、2)の物件である。このため、月次にオンライン休止日に、レポート中の不要となった物件の削除および検索専用レポートへの物件の移動を実施し、更新対象レポートの行数が一定数以上にならないようにしている (DB のガーベージ・コレクション)。

また、更新対象レポートは、RID 分割されているが、検索専用のレポートは RID 分割されていない。これは、検索時にバイナリサーチ (@BFN) により検索するために、

行数が効率に与える影響が少ないためであるとともに、一覧表処理の効率を上げるためである。

4.2 他システムとのインタフェース

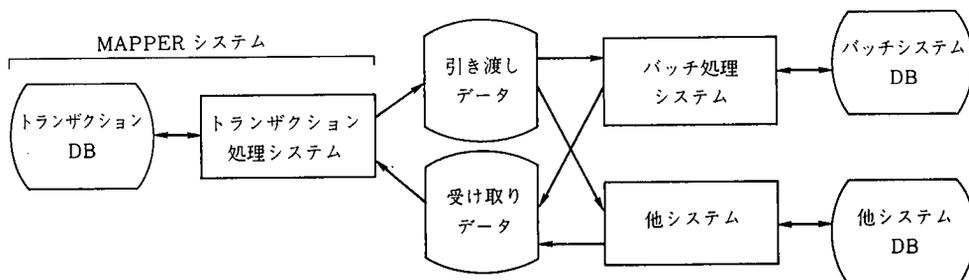


図4 他システム・インタフェース

Fig. 4 Interface between MAPPER and other systems

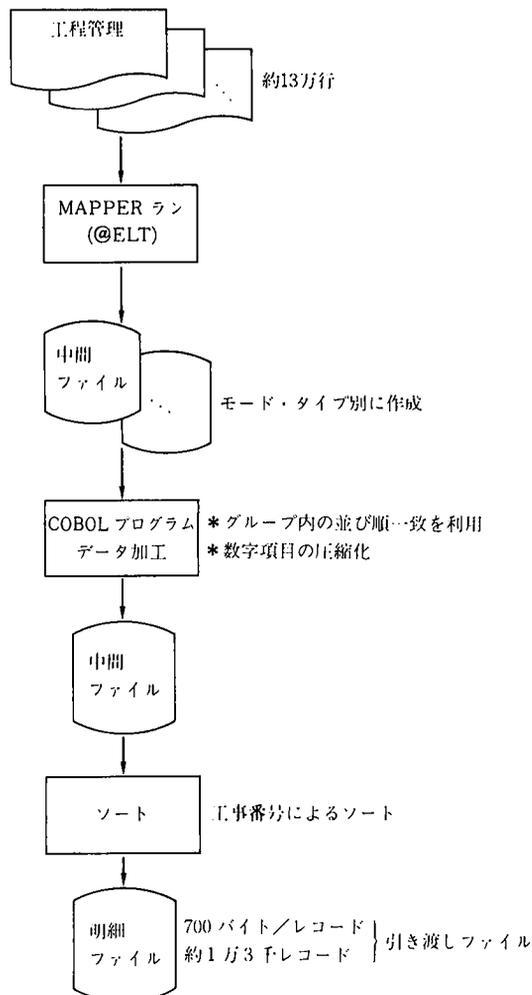


図5 バッチシステム引き渡しレコードの作成

Fig. 5 Production of a record which is transferred to the batch system

トランザクション処理システムは、通常オンラインによるトランザクション処理と、締切り処理等に代表されるオフラインによるバッチ処理がペアとなって運用される。したがって、トランザクション処理を MAPPER で行う場合、オフラインによるバッチ処理または、他システム・インタフェースと整合性を保持しつつ、短時間に実施できなければならない。

本システムの場合、工程管理を始めとするレポートは、トランザクション・データであり、日次締切処理後のバッチ処理（入金処理、日報月報作成）に必要なデータである。引き渡しデータの作成および受け取りデータのレポートへの展開を短時間に実施する必要がある。このために、以下の考慮を行っている。

- 1) グループ内の並び順とレポート行数を一致させ、ソート時間の短縮を計る。
- 2) 大量に RID 分割されているレポート（250 RID、約 24 万行）は、当日更新したレポートのみ引き渡す。

1) は、図 5 に示すように、約 13 万行のレポートを 700 バイト/レコードの明細ファイル（トランザクション・ファイル）に加工するが、不要なソートを排除している。

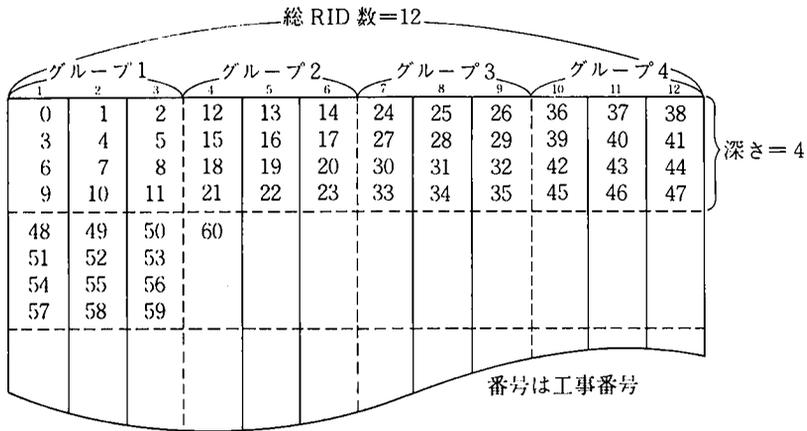


図 6 大量 RID を持つレポートの RID 番号決定

Fig. 6 RID number calculation in a report which has a large number of RIDs

2) は、RID 分割の手法としてマスタ・レポート・インデックス以外の決定ロジックを採用し、更新されるレポートがグループ内に集中するよう考慮するとともに、更新ロックの待ちを極力少なくするように設計している（図 6）。

5. おわりに

トランザクション処理システムの DB として MAPPER DB を採用した場合の考慮点を事例を示して報告したが、適用の最大のポイントは、

- 1) トランザクション量
- 2) DB 容量

の見極めである。RID 分割の手法により、トランザクション処理の効率化は、ある程度可能であるが限界が存在する。一方、利用部門の関心はトランザクション処理を中心とした効率に限定されるが、システム部門は業務処理や他システムとのインタフ

ケースを含めた効率を考慮する必要がある。本事例は、システムの特性を分析し両部門の要求を許容範囲内に納めた例であり、この手法がすべての場合に適用できるものではない。システム開発の効率化が求められる昨今、MAPPER の適用技術が蓄積され、MAPPER によるシステム開発の効率化と既存 DB/DC を中心とする処理の効率化が融合することを望む次第である。

執筆者紹介 小西 徳雄 (Norio Konishi)

昭和 27 年生，49 年大阪府立大学工学部経営工学科卒業。同年日本ユニバック(株)入社。製造工業および社会公共事業関連のオンライン・リアルタイム・システムの設計開発に従事。現在に至る。情報処理学会会員。



MAPPER によるユーザ・リアルタイム・システム事例

Development of A Realtime System using MAPPER1100

浦 伸 一

要 約 従来の MAPPER の事例の多くは、特定の基幹システムやサブシステムの一部への適用であるが、本稿で紹介する事例は、業務全体を MAPPER 1100 を主体として運用し、利用者とのインタフェースをすべて MAPPER に集約させたことに特徴がある。その結果、リアルタイム・プログラムやバッチ・プログラムを実行する場合にも、利用者は業務の処理形態を意識することなく、すべて MAPPER 操作で遂行できる。とくに、エンド・ユーザ志向である MAPPER 1100 をユーザ・インタフェース部に用いることにより、種々のシステムの変更要求や拡張にフレキシブルに対応可能であり、MAPPER の良さを十分に活かすことができる。

Abstract The paper reports a system which facilitates to bring both realtime programs (TPS) and batch programs into the framework of MAPPER.

The system realizes:

- 1) uniform user interface which is provided by MAPPER
- 2) massive realtime processing which is provided by TPS.

1. はじめに

近年ユニバック・ユーザでは、MAPPER の導入が浸透し、その活用のされ方も多岐に渡ってきている。とくに、基幹システムへの適用やサブシステムへの適用では、MAPPER 自身でクローズしたシステムと他の処理形態の長所の両方を生かすために、MAPPER と他の処理形態とを組み合わせることが増えてきた。

本稿では、MAPPER ラン、リアルタイム・プログラム (TIP 1100 使用プログラム) およびバッチ・プログラムで構成された業務システムの利用者インタフェースを、処理形態にかかわらず、すべて MAPPER インタフェースに統合したシステムの構築事例を紹介する。

2. システム概要

本システムは、MAPPER 1100 を中核として MAPPER の使用者ランとリアルタイム・プログラム (以後、TPS と呼ぶ) およびバッチ・プログラムからなる (図1)。



図1 システムの構成

Fig.1 System architecture

利用者は、MAPPER 端末からサイン・オン操作を行い、MAPPER で作成した業務メニュー画面を呼び出し、処理する業務を選択入力する。MAPPER の使用者ラン内では、選択された業務を処理する別の MAPPER ラン、TPS またはバッチ・プログラムをスケジュールし実行する。このように本システムでは、MAPPER が業務の中心としてスケジューラの役目を果たしている。このため、バック・グラウンドとして実行されるものがいかなる処理形態であっても、利用者にとっては、あたかも MAPPER 独自で処理しているかのように実行される。

システム全体の特徴としては、以下の項目があげられる。

- 1) ユーザ・インタフェース……利用者は、いかなる処理形態であっても、MAPPER の操作によりシステムの全業務処理を行う。
 - ・MAPPER 内にすべてのメニュー画面をもつことにより、全業務の開始をメニューより選択する。
 - ・リアル処理とバッチ処理の時間帯の運用区分がない。
- 2) TPS 処理インタフェース……TPS での処理業務が選択された場合、MAPPER 使用者ランから TPS のスケジュール要求を行う（MAPPER を使用しない場合、一般には入力電文の一部を用いて TPS のスケジュールの振り分けが必要）。
 - ・MAPPER からの TPS スケジュールは、@TPS 機能を使用している。
 - ・MAPPER-TPS 間のインタフェース・ファイルは、MAPPER の外部レポートを用いている。
 - ・TPS で生成する帳票（伝票等）は、MAPPER のレポートに生成し、MAPPER の操作により端末に印書（@AUX 機能）する。
 - ・TPS スケジュール方式として、応答待ち型（同期型）あるいは一方向スケジュール型（非同期型）を目的別に利用（MAPPER 使用者ランで管理）している。
- 3) バッチ処理インタフェース……バッチ・ランは、MAPPER のメニュー画面からバッチ処理用のパラメタ（日付、処理条件等）を入力し、@STR 機能により MAPPER から起動する。

3. 業務処理の流れ

3.1 業務処理手順

利用者の業務処理手順は、次のようになる。すべての操作は、MAPPER を通して行われる。

- 1) MAPPER のサイン・オンによる開始
- 2) 業務メニューの表示
- 3) 業務の選択
- 4) 入力画面の表示
- 5) 入力画面へのデータ入力……入力後、処理すべき TPS をスケジュール。
- 6) TPS 処理結果の画面への表示……スケジュールされた TPS 内でデータ・チェックエラーがあれば 4) へ戻る。
- 7) 次業務の選択……6) の処理結果表示後、ファイル更新を必要とする業務を選択したならば、ファイル更新のための TPS がスケジュールされる。

3.2 業務内部処理

MAPPER ランは、全業務共通のメニュー処理ラン(1個)と TPS ごとの処理ラン (TPS の数だけ存在する) で構成されており、処理の振り分け (スケジュール) はすべて MAPPER ラン内で行っている (図 2)。

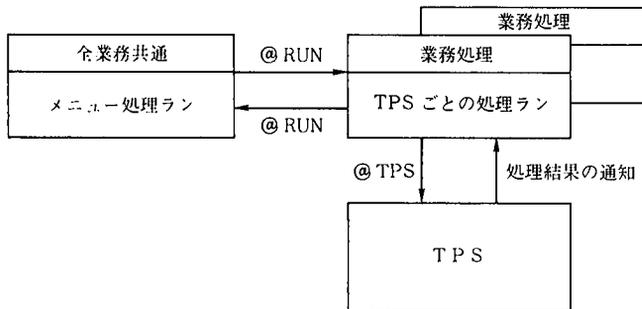


図 2 業務内部処理

Fig. 2 Scheduling mechanism

これらのやり取りを図示したのが図 3 である。

4. MAPPER スケジュール機能

このシステムで使用する MAPPER からの TPS およびバッチ・プログラムのスケジュールは、次の MAPPER 1100 の持つ機能を用いている。

4.1 MAPPER-TIP インタフェース機能

MAPPER-TIP インタフェースの機能は、大きく次の 3 点からなる。

- 1) TIP スケジュール機能……会話機能 (“TPS”) または、ラン・ファンクション (“@TPS”) で、MAPPER から TPS をスケジュールする。
- 2) TIP ポート機能……TPS から MAPPER ランをスケジュールする。
- 3) MAPPER/TPS 間のデータ受授……COMPOOL または FCSS ファイル経由で TPS で作成したデータを、MAPPER レポートとして引き渡すことができる (その逆も可能)。

これらを図示したのが図 4 および図 5 である。

MAPPER-TIP インタフェース機能の使用に当たっては、MAPPER 管理者として次の作業を行うだけでよい。

- 1) MAPPER 1100 実行時パラメータの設定
- 2) MAPPER 使用者ランの登録
- 3) MAPPER TPS レポートの登録

上記以外に、TPS 稼動環境設定のための TIP 1100 の準備作業が必要となる。

4.2 MAPPER-バッチ・インタフェース機能

MAPPER-バッチ・インタフェースの機能として、TPS と同様に次の 3 点からなる。

- 1) バッチ・スケジュール機能……会話機能 (“START”) または、ラン・ファンク

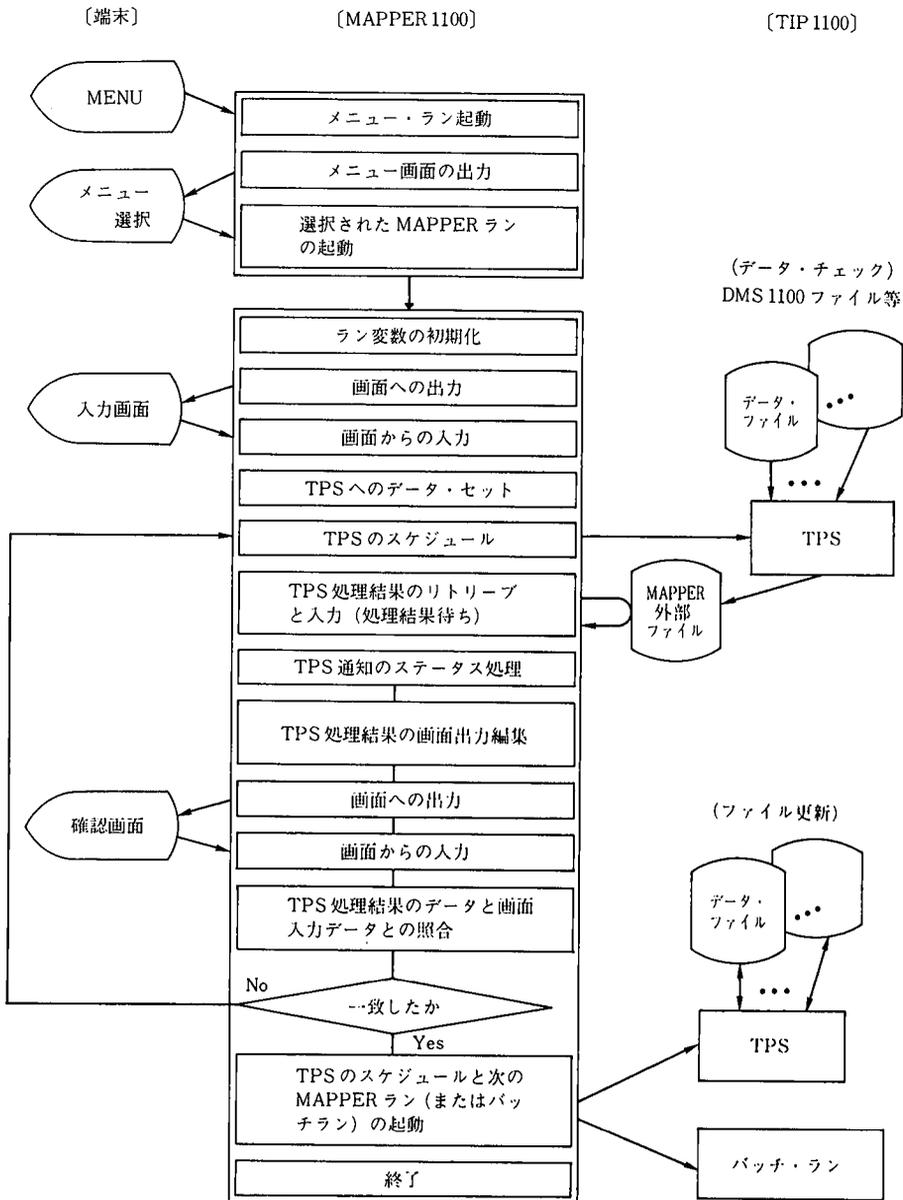
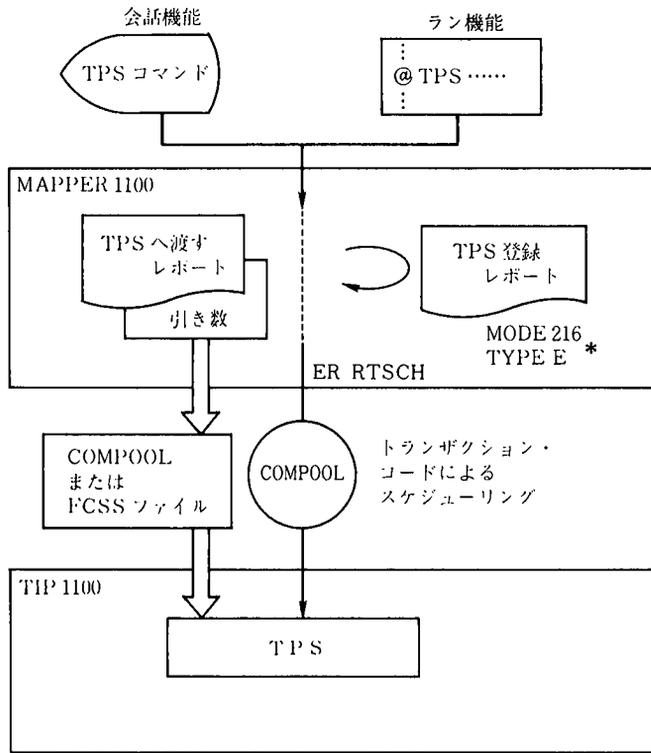


図3 MAPPER と TPS のやりとり

Fig. 3 Data transfer between MAPPER and TPS

ション (“@STR”) で、MAPPER からバッチランをスケジュールする。

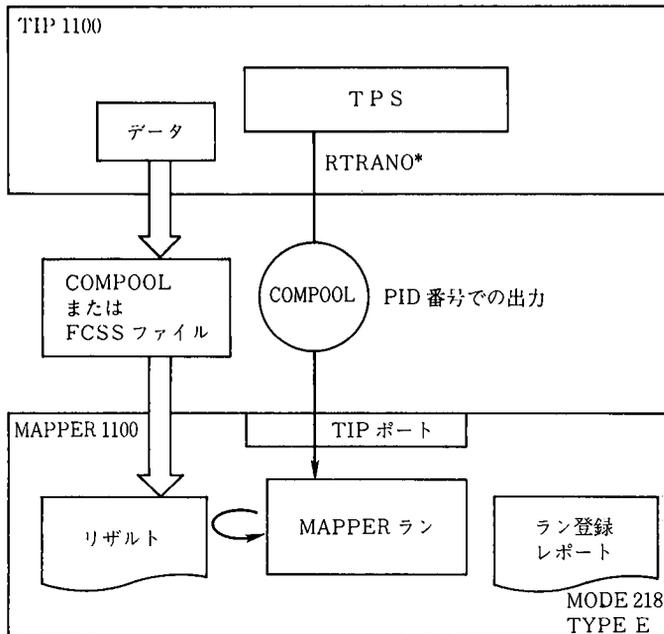
- 2) バッチ・ポート機能……バッチ・プログラムから MAPPER ランをスケジュールする。
- 3) MAPPER/バッチ間のデータ授受……SDF ファイル経由でバッチ・プログラムで作成したデータを MAPPER レポートとして引き渡すことができる (その逆も可能)。



* MAPPER レポートのモードとタイプを意味する。

図4 TPS スケジュール機能概略図

Fig. 4 TPS scheduling function



* TIP プリミティブの一つ。

図5 TIP ポート機能概略図

Fig. 5 TIP porting function

5. システム構築における考慮点

5.1 TPS 処理の考慮点

このシステムの業務処理 TPS は、図 3 に示したように、データ・チェック処理とファイル更新処理に分割されているが、実際には二つの処理を一つの TPS モジュールとして構成し処理する。

このため、MAPPER ランから TPS をスケジュールする際に、業務処理に必要な入力データ以外にコントロール・テーブルと称する処理制御情報を与え、MAPPER-TPS 間の制御を行っている。

5.1.1 コントロール・テーブル

80 文字のカードイメージで、次の形式である。

TRAN -ID	STATUS -FLG		NEXT -TASK	UNIT -NO	TIME -STUMP	STATUS -1
X(4)	9(5)		X(4)	X(4)	9(6)	X(1)

表記項目以外は、MAPPER のラン同士のデータ授受として利用される。各項目の意味と使用法は次の通りである。

- 1) TRAN-ID……MAPPER 側で TPS の名前を設定し、スケジュールされた TPS 側で検査する。また、ファイル更新処理時のエラー発生に対するエラー表示として使用する。
- 2) STATUS-FLG……TPS 側の処理結果を MAPPER 側に通知するフラグ。初期値“0”で TPS 側に渡され、TPS 側の入力データ検査時、論理エラーが発見された場合“1”を設定し、正常な場合“2”を設定する。したがって、MAPPER ランは、このフラグにより後続処理を決定する。
- 3) NEXT-TASK……この項目にラン名が設定されていると、MAPPER では TPS 処理の終了を検知して利用者の介在なしに、この項目に設定された MAPPER ランを起動する。
- 4) UNIT-NO……データ入力している MAPPER の端末番号を MAPPER ランで設定し、スケジュールされた TPS 側で MAPPER 外部ファイルへの書き込み時のレポート番号 (RID 番号) として使用する (後述の 5.3 節)。また、ファイル更新処理を行う TPS のエラー発生時の通知先として使用する。
- 5) TIME-STUMP……MAPPER ランで時刻を設定し、TPS の処理結果を取り出す際に要求した出力データか否かを検査するために使用する。
- 6) STATUS-1……TPS 側で入力データの論理的なエラー以外のエラー (たとえばファイルの入出力エラー) が発生した場合に設定され、MAPPER 側に返される。MAPPER 側では、これを受け取りその旨を端末に表示する。

5.1.2 TPS のエラー処理

MAPPER よりデータ・チェック処理 TPS をスケジュールした場合、図 3 で示されている通り MAPPER と TPS は、同期をとった処理を行う。したがって、TPS 側で入力データのエラー等の論理的な異常を発見すれば、MAPPER 側に通知可能であり、また、TPS の処理結果が一定監視時間内に戻らない場合にも MAPPER 側で検知可

能であり、共に利用者にその旨を通知することができる。

しかし、ファイル更新処理 TPS をスケジュール (非同期型) した場合は、スケジュール要求した MAPPER 側は、TPS の処理結果を待たず次の業務処理を続行する方式を採用している (これが、このシステムのメリットでもあり弱点でもある。すなわち、トランザクション更新処理とデータ入力処理が非同期に同時併行可能な反面、システム・ダウン発生時の特別な考慮が必要である)。

ファイル更新処理 TPS に論理的なエラーが発生した場合には、処理を要求した端末へエラー・メッセージを通知する。メッセージ通知の手段として、MAPPER のバッチ・ポート機能を使用している (図 6)。

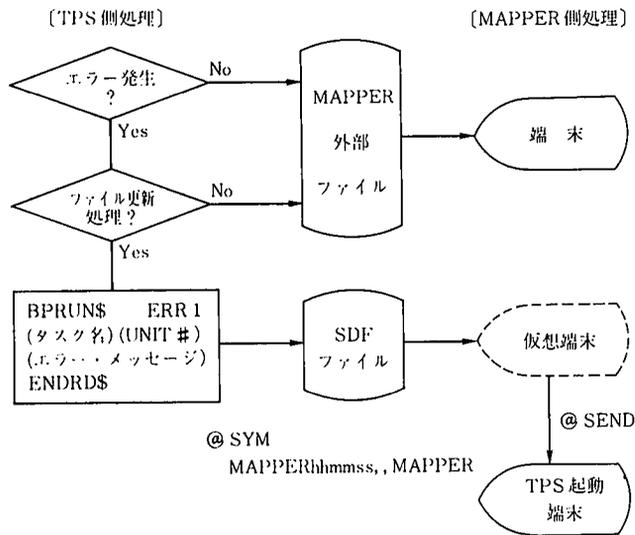


図 6 TPS からのエラー通知方法
Fig. 6 Noticing errors from TPS

この手法は、端末利用者の注意力に委ねられているため、端末のベル鳴動によってのみエラー通知が判明する。

5.2 バッチ処理の考慮点

バッチ・ランの処理は、TPS 処理と同様すべて MAPPER メニュー画面を通して行われる。前述の図 3 のメニュー選択にてバッチ処理がスケジュールされた場合である。バッチ処理における考慮点としては、スケジュールするバッチ処理に渡す処理条件の受け渡し方法と、バッチ処理完了後の端末への通知である。

5.2.1 バッチ処理受け渡しパラメタ

バッチ・プログラムでは、MAPPER からの処理条件のパラメタ入力として代替シンピオント・ファイルを使用し、MAPPER ランとの受け渡しを行っている。MAPPER ランでは、画面から入力された処理条件を付加し、指定されたバッチ・ランの JCL を生成し、バッチ・ランをスケジュールする。以下は、MAPPER ランで生成する JCL の例である。JCL 内のファイル名 “PARM” が MAPPER とバッチ間のインタフェース用のファイルであり、処理条件等が設定される。

```

@BRK
'@' RUN      V 160, V 161, V 162
'@' ASG, T   PARM., F
'@' ED, IQ   PARM.
V100V160V170.....(変数の内容)
                        V100：端末番号
                        V160：タスク名
                        V170：画面入力された処理条件

'@' EOF
'@' ADD      REALJCL . V160
'@' XQT      PROG . MSGSEND
V100V160
'@' ERRJMP :
@BRK
@STR...
    
```

5.2.2 バッチ処理完了の通知

この処理は TPS のエラー処理と同様バッチ・ポートの機能を使用している。正常に処理が終了した場合には、全バッチ・ラン共通のタスクである 'MSGSEND' (使用者プログラム) で処理を要求した端末へ結果を通知する。

エラーがあった場合には、ラン状況語に値をセットしそのタスクの実行をスキップする。

```

@ASG, A  ~
      :
@XQT  PROG . XXX
@TEST TE/0/T3
@JUMP  ERRJMP
@XQT  PROG . XXX
@TEST TE/0/T3
@XQT  PROG . MSGSEND
@ERRJMP :
    
```

エラーの場合にはエラーの種別を MAPPER に渡し、MAPPER 側で種別を判読して端末にエラー・メッセージを表示する。

これらの処理を図示すると図7のようになる。

5.3 効率面の考慮点

MAPPER-TPS 間のデータ授受には、次の2通りの方法がある。

- 1) バッチ・ポートの機能を使用し、TPS で出力した SDF ファイルを MAPPER で取り込む。
- 2) TIP ポートの機能を使用し、TPS の処理結果を MAPPER の外部ファイルのレポートに出力する。

当初、MAPPER-TPS 間のデータ授受には、すべてバッチ・ポート方式を採用した

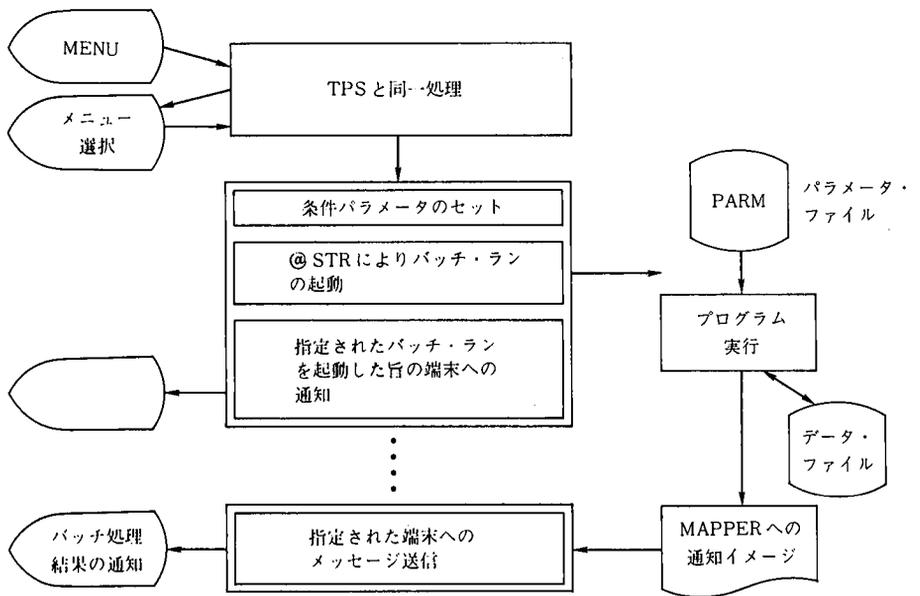


図7 MAPPER-バッチのやり取り

Fig. 7 Data transfer between MAPPER and batch program

が、レスポンスの悪さと MAPPER がシステム全体に与える負荷との両面を考慮した結果、TIP ポート機能を使用した (図 8)。

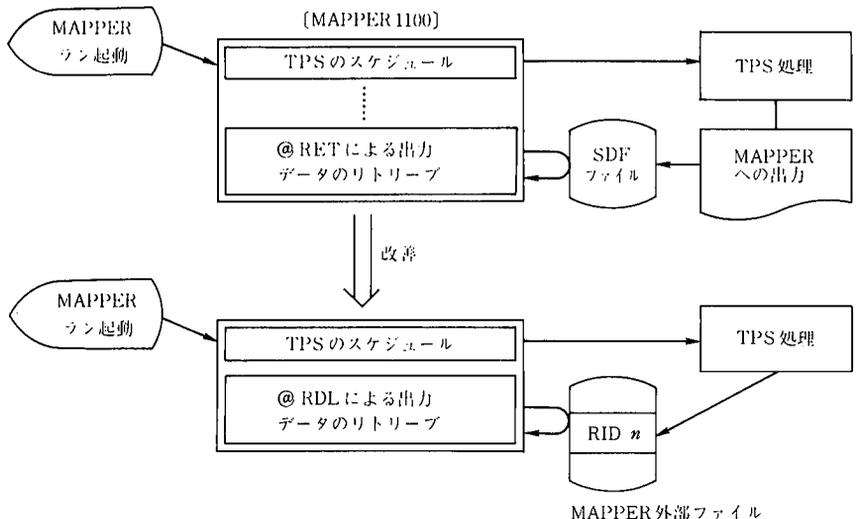


図8 MAPPER-TPS間データ授受

Fig. 8 Data transfer between MAPPER and TPS

その結果、応答時間は約半分に改善され、また MAPPER の命令の使用法を変更し (バッチ・ポートでは@RET, TIP ポートでは@RDL) したことにより、システム全体の負荷も減少した。レポートの性格上、保存の必要性がないため MAPPER の外部

ファイルを採用しているが、外部ファイルは MAPPER のページ処理の対象外となるため、ページ処理時間の短縮を図ることができる。

このシステムの MAPPER データベース構成を示したのが表 1 である。モード 300、タイプ I が MAPPER-TPS 授受用のデータ格納用レポートである。

表 1 MAPPER データベースの構成
Table 1 MAPPER database

ファイル名	モード	タイプ	RID	行長	
MAPER 3	28	I	1~200	80	ラン・ファンクション
		H	1~100	132	シンビオント・ファイル印書
MAPER 0 (外部ファイル)	300	I	1~100	80	MAPPER-TPS のデータ受け渡し
		H	1~100	132	伝票発行

格納レポートは、端末ごとに一意に決まるレポート (RID) が割り当てられるようにしている。簡単な方法として、MAPPER ランを実行した端末番号(ステーション番号)の下 2 桁を RID 番号として使用している。

例) ステーション番号 1010 → RID 番号 10

この方法により、99 台までの端末を稼動することができる。

5.4 伝票発行の考慮点

MAPPER 業務メニュー画面より印書指示を選択した場合、必ず起動端末に印書データを出力する考慮を行っている。起動端末の画面-1 からの印書指示であれば画面-2 を経由し、画面-2 からの印書指示であれば画面-1 を経由し印書する。また、印書装置の故障時の代行印書として、任意の印書先を指示することもできる。

図 9 は、TPS で印書データを作成し、バッチ・ポート機能を用いて印書する方法である。バッチ・ポート実行後は、印書指示と非同期に MAPPER から任意に取り上げられ、指定してある MAPPER 使用者ランの実行により印書装置に出力される。

しかし、出力イメージの多いバッチ・ランの処理結果等はこの方法で十分であるが、

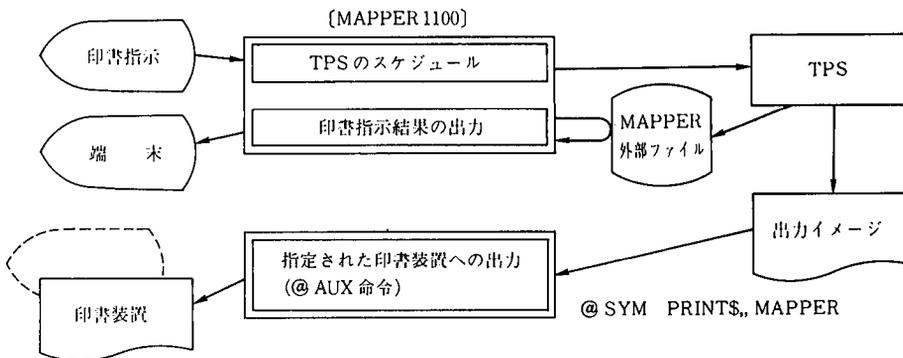


図 9 バッチ・ポート機能を利用した印書方法 (非同期出力)

Fig. 9 Print out via batch port (asynchronized output)

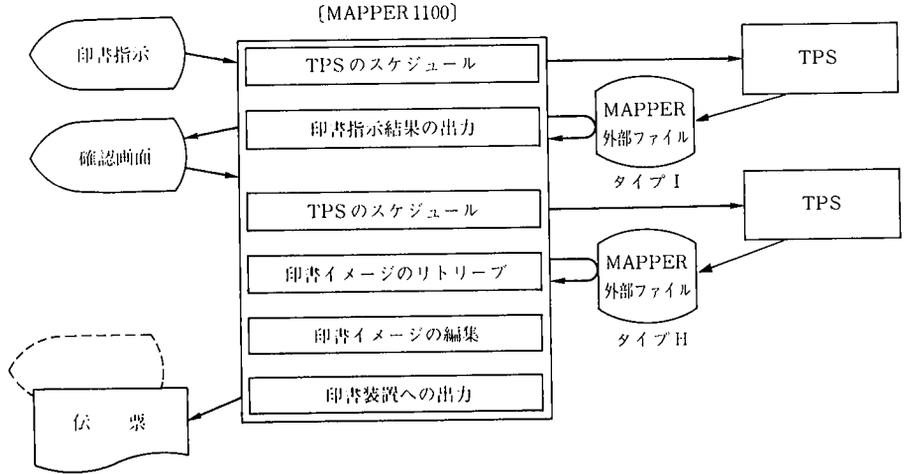


図 10 TIP ポート機能を利用した印書方法 (同期出力)
 Fig. 10 Print out via TIP port (synchronized output)

印書指示後、実際に印書装置に出力されるまで時間がかかる上に、印書指示と実際の印書が非同期のために指示順に印書されない欠点がある。とくに伝票発行処理等は、即時性と発行順の遵守が必要である。このため伝票発行処理では、前述した MAPPER-TPS の同期管理を行い、さらに伝票の印書データを MAPPER の外部ファイルに格納する方式を採用している (図 10)。

表 1 中のモード 300, タイプ H 内の RID 番号 1~100 までが、伝票の印書データの格納用レポートであり、5.3 節で述べたと同様の方法で管理使用される。

6. 当システムのメリット

6.1 ユーザ・インタフェース部への MAPPER 利用効果

第 1 に、ユーザ・インタフェースをすべて MAPPER に集約したことにより、利用者は MAPPER 操作のみ習得していればよい。端末の開閉局から始まり業務処理操作全般を MAPPER の機能命令 (または MAPPER の入出力画面) だけで行うことができる。

第 2 に、抽出したデータの 2 次加工が容易である。バック・グラウンド処理 (TPS やバッチ処理) で作成されたデータを、必要であればさらに MAPPER の会話型機能等を使用し簡単に加工することができる。TPS だけで構成されたシステムでは、必ず加工プログラムが必要となる。

第 3 に、変更要求の多いユーザ・インタフェース部分に MAPPER を採用することで、フレキシブルに対応できる。とくに画面の追加や変更の場合、直接該当する画面だけを対象とすればよく、インタラクティブに画面を見ながら追加や変更できる。

6.2 TPS 利用による効果

第 1 に、既存の基幹システムの大規模なデータベース (DMS 1100 ファイル等) を MAPPER のデータベースに格納する必要がなく、使用者側から見れば、あたかも MAPPER のデータベースに存在するかのように処理できる。この結果、既存データベ

ース情報の有効利用を計ることができる。

第2に、ユーザ・インタフェース部を MAPPER 側で処理することにより、TPS はトランザクション処理に専念できる。このことは、TPS 部分のインタフェース部の変更を極力避けることができると同時に、開発時においても、MAPPER ランと TPS の役割分担とその間のインタフェースを取り決めておくことにより、互いに独立した開発とテストが可能である。

第3に、MAPPER で処理すると効率の面から負荷が大きく、かつ即時性を要求されるものについては TPS 処理化を計り、システム負荷を軽減させることができる。

第4に、TPS 単独でトランザクション・システムを構築する場合に比べ、次の作業が軽減される。

- 1) 特別な TPS スケジュール機能が不要……MAPPER 使用者ラン内で、起動する TPS を決定し、ラン内から TPS のスケジュール化を要求し実行する。
- 2) 端末増による変更への対処……MAPPER のコンフィギュレーション用レポート(管理者用レポートの1種)を変更するだけで、業務処理プログラム(MAPPER ランや TPS 等)の変更が不要。
- 3) 業務処理 TPS の新規追加や変更……特別なリアル環境を必要とせず開発が可能(ただし TIP 1100 の TPS 登録は必要)。

7. おわりに

本システムは、MAPPER 1100 バージョン 32 R 1 で提供されている機能を利用した MAPPER-TPS、および MAPPER-バッチ・インタフェースを主体にした構築事例である。本稿が MAPPER を利用する上での一助になれば幸である。MAPPER の TPS 機能が MAPPER 本来の特徴と相まって有効に利用され、ユーザ・システム構築の強力なツールとなることを期待している。

執筆者紹介 浦 伸一 (Shin-ichi Ura)

昭和 23 年生、47 年大阪府立大学経済学部卒業。同年日本ユニバック(株)入社、入社以来、製造工業関連担当として SE サービスに従事。現在、大阪支店システム二部に所属。



MAPPER における排他制御とリカバリ方法の考察

Study of the Exclusive Control of the Report Updation
and the Recovery Mechanism in MAPPER

岡島伸介

要約 MAPPER によるアプリケーション開発に際しての重要な点は、同一レポートへの多重更新や異常終了発生時におけるリカバリをいかに考慮するかである。MAPPER には、LOK 命令と DFU 命令が存在し通常の場合は問題ないが、6 レポート以上の一括更新やロック中の禁止命令実行などを行う場合には機能的に十分とは言いがたく、各処理ランの中に排他制御やリカバリ機能を組み込まなければならない。

第1の考慮はランの起動制御である。ラン起動可否テーブルによってラン間の同時実行関係を管理し、ランの開始時にそのランが起動してよいか否かをチェックする。

第2の考慮はレポートの排他制御である。ロックレポート管理テーブルによって更新中のレポートを管理し、更新前にレポートを更新してよいか否かをチェックする。

第3の考慮はリカバリ機能である。更新時にはレポートの更新前イメージを退避しておき、ランの開始時にリカバリの必要性を判断して自動的にリカバリする。

以上3点の考慮を外部サブルーチンとして各ランに組み込むことにより、排他制御とリカバリにおいて必要十分な機能を持たせることができる。

Abstract Some difficult problems in developing MAPPER applications are:

- 1) how to prevent two or more runs from updating one report at the same time.
- 2) how to design the recovery mechanism for database in case of a contingency error and power failure.

Though MAPPER system has some run commands to solve these problems, namely, @LOK and @DFU, they are not sufficient.

Therefore the inclusion of following devices in each run function is required.

- 1) the control of run execution: a table to administrate the system condition, and judge whether or not a run may start.
- 2) the control of report updation: a table to administrate reports, and judge whether or not a run may update the report.
- 3) the recovery mechanism: each run copys the report before updation, and judges whether or not a run may recover automatically when restarting.

By including the above-mentioned considerations in each run as external subroutines, each run can have enough functions to provide the report updation control and the recovery mechanism.

The paper discusses the problem of the report updation control and recovery mechanism, and shows a solution for it.

1. はじめに

MAPPER によるアプリケーション開発は、さまざまな分野に及んでいる。システムの規模としても、小さなものから大きなものまで多様であり、MAPPER に取り込めない業務を捜す方が困難である。

しかしながらシステム設計上、開発担当者を悩ます共通の問題がある。それは、同一レポートへの多重更新や多数レポートの一括更新をどう制御するかという問題であり、DMS 1100 等におけるランユニットの概念が MAPPER では不十分であることにより発生する。これらの問題を解決しなければシステム構築はなし得ないのだが、その解決手法が確立されているとは言いがたい。

本稿では MAPPER における排他制御とリカバリ方法について考察し、一つの汎用的な解決手法を示す。

2. 問題点

2.1 レポート更新における問題点

MAPPER システムを設計するに当たって、上述の問題をどう扱うかは大きな課題である。レポートへのデータ登録、変更、削除等において、何も対策を講じなければ、

- 1) 同一レポートを複数の端末により同時更新すると、内容に矛盾が生じる(図 1)
- 2) 多数レポートの一括更新中にシステムダウン、回線エラー、ランエラー等が発

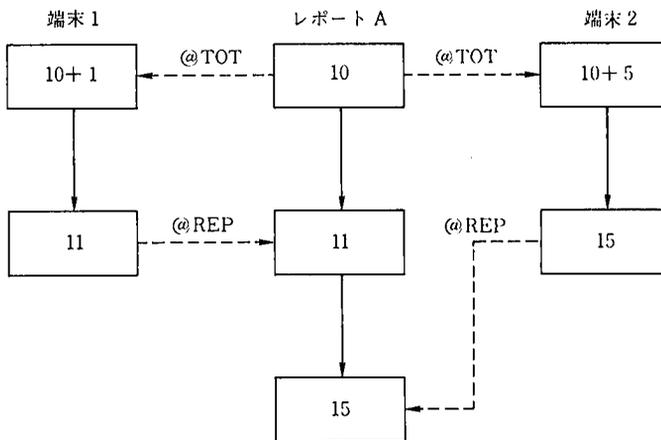


図 1 複数端末による同時更新例 (その 1)

Fig. 1 The example of report updations at the same time by multiple terminals(1)

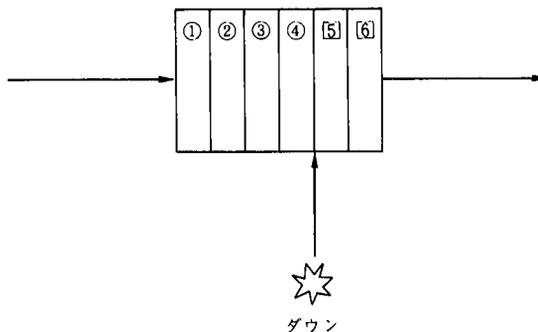


図 2 多レポート更新中のダウン

Fig. 2 The abort during multiple report updations

生するとレポート間に矛盾が生じる(図2)
という現象が発生する。

これらの問題は、DMS 1100 等におけるランユニット単位の排他制御とリカバリ機能を実現すれば解決する。ランユニットとは、プログラム実行中のある範囲のことであるが、ランユニットの考え方を以下に示す。

- 1) ランユニットの範囲内においては、他プログラムの影響を受けない。つまり、同一レポートを複数端末にて同時更新しようとしても、図3のようにどちらかが待たされる。
- 2) ランユニットの範囲内においては、どのような更新がなされても、万一異常が

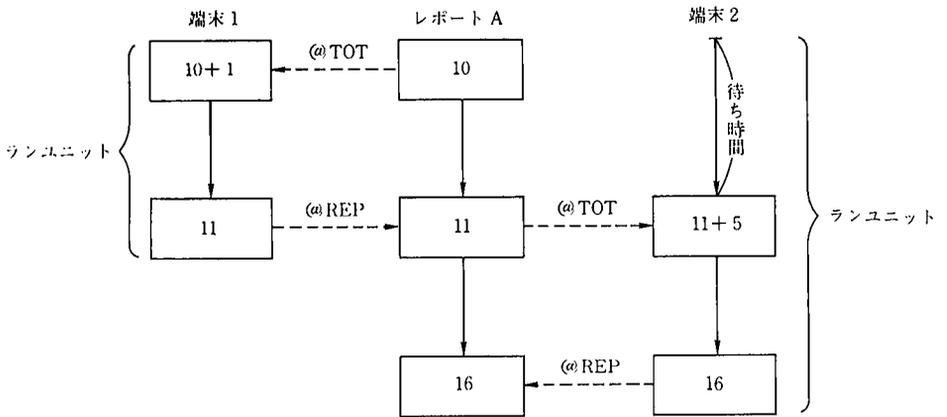


図 3 複数端末による同時更新例 (その 2)

Fig.3 The example of report updations at the same time by multiple terminals(2)

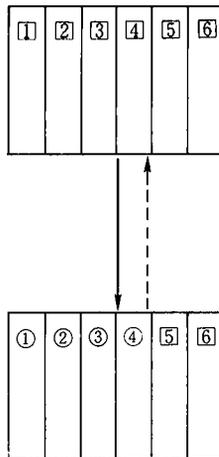


図 4 6 レポートの更新例

Fig.4 The example of 6 reports updation

発生した場合はその異常がどこで発生しても、まったく更新されていないか、すべて正しく更新されているかのどちらかである。たとえば、図4のように6レポートを更新する処理の場合、更新途中でダウンすると(図2)、すべてもとの状態に復元される(リカバリ機能)。

2.2 LOK 命令と DFU 命令の問題点

ランユニットに似た考え方は MAPPER でも存在する。LOK 命令と DFU 命令である。しかし、どちらにも制限がある。

2.2.1 LOK 命令

LOK 命令とは、ランでレポートを更新しようとするとき、他の使用者による同一レポートの更新を一時的にロックする命令である。もし、他の使用者が先にこのレポートの更新を開始していると、その更新が終了しロックが解除されるまで、そのランの実行は待たされるか、ラベルへ分岐する。

LOK 命令の制限は以下の通りである。

- 1) LOK 命令は一つのレポートしかロックできない。
- 2) LOK 命令は次の命令で解除される。
 - ULK 命令 (UNLOCK: ロック解除)
 - UPD 命令 (UPDATE: 更新)
 - DEL 命令 (DELETE LINES: 行削除)
 - ADR 命令 (ADD REPORT: 新設)
 - REP 命令 (REPLACE: 置き換え)
 - DUP 命令 (DUPLICATE REPORT: 複製)
 - LOK 命令 (UPDATE LOCK: ロック)
 - AUX 命令 (AUXILIARY DEVICE: 端末印書)
 - SEN 命令 (SEND: 送信)
 - DFU 命令 (DEFERRED UPDATES: 遅延更新)
 - SOR 命令 (SORT: 分類)
 - ランの終了

しかも上記命令中、SOR 命令以外は LOK 対象外レポートへの実行でも解除される。

2.2.2 DFU 命令

DFU 命令は CMU 命令および DCU 命令とともに使用し、複数のレポートの更新を制御するための命令である。DFU 命令を使用すると、ランが途中でエラーになったとき、DFU 命令上に指定したレポートは、たとえ更新が行われていたとしても DFU 命令を実行する前の状態にもどされる。更新が実際にレポートに反映されるのは、CMU 命令を実行したときである。また DCU 命令が CMU 命令のかわりに実行されると、更新は取り消され、レポートは DFU 命令を実行する前の状態にもどされる。

DFU 命令の制限は以下の通りである。

- 1) DFU 命令は最大5個のレポートまで同時にロックできる。
- 2) DFU 命令と CMU 命令、または DCU 命令の間には次の命令を置くことはできない。
 - DFU 命令

- ・DSP 命令 (DISPLAY REPORT：レポート表示)
- ・LOK 命令
- ・ISR 命令 (IMAGINARY STATION RUN：仮想端末)
- ・OUM 命令 (OUTPUT MASK：マスク表示)
- ・OUT 命令 (TERMINAL OUTPUT：画面表示)
- ・REL 命令 (RELEASE DISPLAY：ラン終了)
- ・RTN 命令 (RETURN REMOTE：遠隔ラン復帰)
- ・RUN 命令 (START RUN：別ラン始動)
- ・WAT 命令 (WAIT：休止)
- ・XIT 命令 (SIGN-OFF：サインオフ)

2.2.3 LOK 命令と DFU 命令で実現できない処理

LOK 命令と DFU 命令には前述のような制限がある。そのため、システム設計者の要求を満たすことができないことがある。それは、

- 1) 6 レポート以上の一括更新
- 2) ロック中の禁止命令 (前述) の実行

を行う場合である。これらの場合は、プログラミングの工夫によりランユニット的な考え方を採用しなければならない。

3. 問題解決のポイント

LOK 命令と DFU 命令で対処できない処理が含まれているシステムを設計する場合、同一レポートへの多重更新や多数のレポートの一括更新を制御するためには、

- 1) 論理的に同時実行すべきでないラン間の関係をチェックし、ランの多重起動を制限する (起動制御)、
- 2) レポートの更新時、他のランによる更新を禁止する (排他制御)、
- 3) 多数のレポートの一括更新において、各レポート間の関連を保証する (リカバリ機能)、

という3点の実現が必須である。

4. ランの起動制御

MAPPER ランによるシステム設計においては、複数ランの多重処理に注意しなければならない。同時実行すべきでないラン間の関係をチェックし、ランの多重起動を制限すべきである。そのためには、

- 1) ラン間の同時実行可否関係はどうなっているか、
- 2) 現在どんなランが実行されているか、

を管理する必要がある。これらを管理するために、ラン起動可否テーブル、端末管理テーブルを使用する。

4.1 ラン起動可否テーブルの設定

ランの種類別にラン ID を割り当て、同時実行が許されるかどうかラン間の関係をテーブルによって管理する。図 5 において、0 であれば同時実行可、1 であれば同時実行不可とする。

レポート・レイアウト上に「桁」を持っているのは、チェック・ルーチンをサブルーチン化しやすくするためである。すなわち、以下の手順でサブルーチン化できる。

- 1) 自分のラン ID のラインを求める。
- 2) 現在実行中のラン ID にて検索し、「桁」を求める。

```
.DATE
.  《 《 ラン起動可否テーブル 》 》
*          .ラン. .3.3.3.
*          ラン名称 .ID.桁.5.7.9.
*=====
ラン A      11 35 1 0 0
ラン B      12 37 0 0 1
ラン C      13 39 1 1 1
..... END REPORT .....
```

図 5 ラン起動可否テーブル

Fig. 5 Run execute decision table

```
.DATE
.  《 《 端末管理テーブル 》 》
* 端末.ラン.
* 番号.ID.
*=====
1580 00
1581 00
1582 00
1583 00
..... END REPORT .....
```

図 6 端末管理テーブル

Fig. 6 Terminal control table



図 7 ラン開始時の処理 (その1)

Fig. 7 The process at the time of run-starting(1)



図 8 ラン終了時の処理

Fig. 8 The process at the time of run-terminating

```

.DATE
. 《 《 端末管理テーブル 》 》
* 端末 .ラン.
* 番号 .ID.
*==== .==.
1580 00
1581 13
1582 00
1583 00
..... END REPORT .....

```

図 9 ラン実行中の端末管理テーブル

Fig. 9 Terminal control table during run-executing

```

.DATE
. 《 《 端末管理テーブル 》 》
* 端末 .ラン.
* 番号 .ID.
*==== .==.
1580 00
1581 00
1582 00
1583 00
..... END REPORT .....

```

図 10 ラン終了後の端末管理テーブル

Fig. 10 Terminal control table after run-terminating

- 3) 自分のラン ID のライン上, 2) で求めた桁を読み, 0/1 を判定する。
- 4) 2), 3) を繰り返し, 1 が現われたら同時実行不可である。

4.2 端末管理テーブルによる実行中ラン管理

現在どの端末でどんなランが実行中であるかを管理するために端末管理テーブル(図 6)を持つ。

端末管理テーブルには当該システムが実行されるすべての使用端末番号を設定し, 各端末で実行中のラン ID を書き込む。図 7, 図 8 はラン ID の書き込みと消し込みのタイミングである。原則としてランの開始時にラン ID を書き込み(図 7, 図 9), ランの終了によってラン ID をクリアする(図 8, 図 10)。

4.3 ラン開始時のチェック

ラン起動可否テーブル(図 5)と端末管理テーブル(図 6)とを組み合わせ使用することによって, 同時実行すべきでないラン間の関係を論理的にチェックし, ラン開始を制限することができる。そこで, 図 7 の流れに工夫を加えたものが図 11 である。

図 11 においてはラン起動不可の場合, 使用者に対して,

- 1) ラン起動を制限しているラン名
- 2) ラン起動を制限している端末名

を画面に表示する。すなわち, どのランによってラン起動が制限されているか, どの端末でそのランが実行されているのかを明らかにすることにより, オペレーションの円滑な運用を図ることができるからである。

図 11 におけるラン開始時のラン起動チェックも, 図 8 におけるラン終了時のラン ID クリアも, そのままラン内にコーディングするのではなく, 外部サブルーチン化し

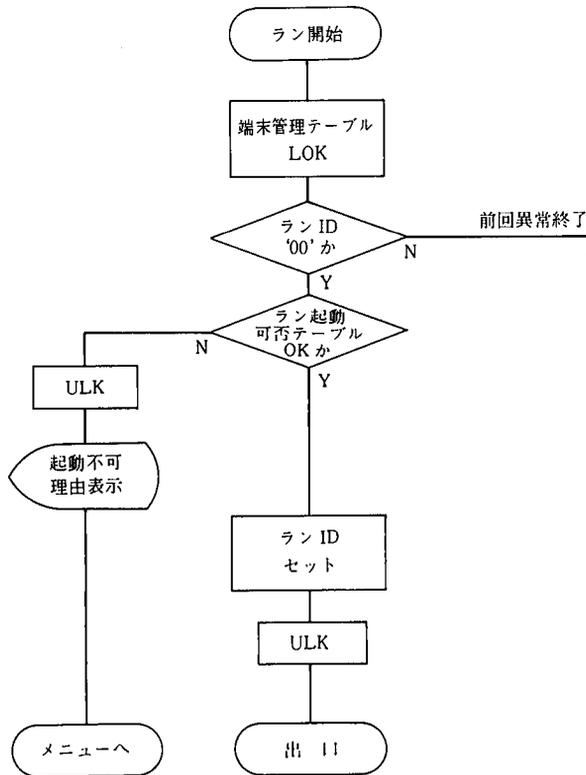


図 11 ラン開始時の処理 (その 2)

Fig. 11 The process at the time of run-starting(2)

てどのラン内でも使用できるよう汎用的にした。

以上述べてきたことをまとめると、ラン間の起動可否チェックをするためには

- 1) ラン起動可否テーブルにてラン間の同時実行可否を管理する。
- 2) ランの開始時にラン起動可否チェックをする外部サブルーチンを呼び、端末管理テーブルに実行中ラン ID を書き込む。
- 3) ランの終了時に端末管理テーブルのラン ID を、クリアするサブルーチンを呼ぶ。

の 3 点が必要である。

5. レポートの排他制御

4 章ではラン単位で同時実行を制限した。しかし、処理によってはラン単位ではなくランユニット単位で排他制御しなければならない場合も多い。なぜならデータ入力プログラムにおいて、更新レポートは一つであるが、入力項目が多く入力件数も多い場合があるからである。この場合、4 章で述べたようにラン単位で同時実行を制限すると、あとから実行しようとしたランが開始されず、最初に行われているランのみとなり効率が悪い(図 12)。しかし、更新時(ランユニット内)のみ排他制御するのであれば同時実行でき、図 13 のように効率的な運用ができる。ではこの排他制御を実現するにはどうしたらよいであろうか。

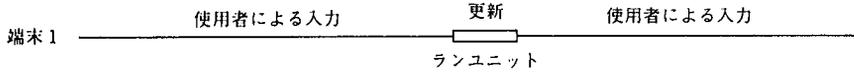


図 12 単独実行のみ可のプログラム

Fig. 12 The program which is executable by a single terminal

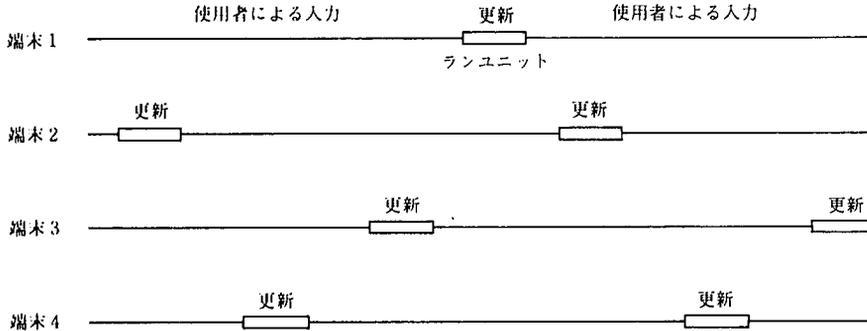


図 13 複数実行可のプログラム

Fig. 13 The program which is executable by multiple terminals

```
.DATE
. 《 《 ロックレポート管理テーブル 》 》
*MOD.T.RID.
*E .P. .
*===.=.===.
180 C 25
324 G 14
 92 B 1
 92 B 2
 92 B 3
126 F 33
..... END REPORT .....
```

図 14 ロックレポート管理テーブル (その1)

Fig. 14 Report-locking control table(1)

5.1 ロックレポート管理テーブル

汎用的な排他制御システムにするためには、更新中レポートをすべて把握する必要がある。すなわち、更新処理開始時にレポート名(モード(M), タイプ(T), リッド(R))を共通テーブルに登録しておき、更新処理終了時にレポート名を消去する。更新ランは、すべてこのテーブルによって更新可否をチェックするのである。この共通テーブルをロックレポート管理テーブルと呼ぶ(図 14)。

図 15 は、更新処理時におけるロックレポート管理テーブルの使用法である。

このロックレポート管理テーブルを使用して、複数端末による更新の排他制御が実現された例が図 16 である。

端末 1 はレポート A, B, C の更新, 端末 2 はレポート B, D, E の更新を行う。どちらもほとんど同時に実行されたが、端末 1 がわずかに早くロックレポート管理テー

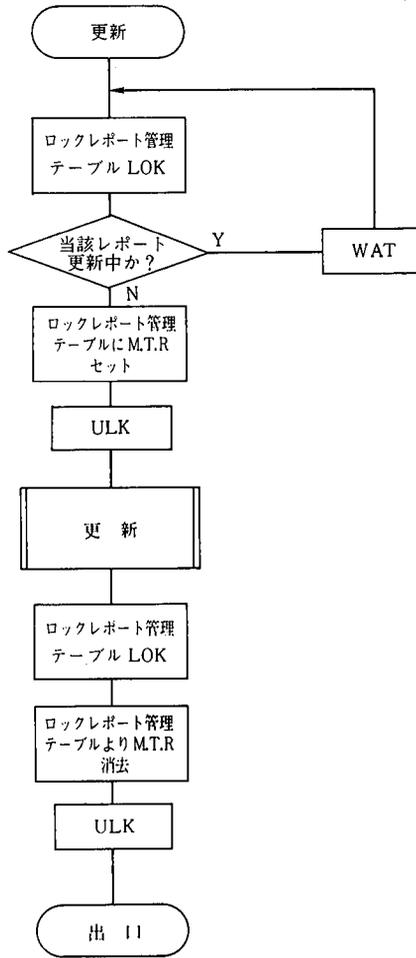


図 15 更新処理 (その 1)

Fig. 15 The process of updation(1)

ブルをロックしたため、端末 1 がロックレポート管理テーブルに A, B, C を書き込んでしまった。ロック解除後、端末 2 がロックレポート管理テーブルをロックし、ロックレポートのチェックをしたところ、レポート B はすでに登録されているため、その解除まで待たねばならない。その間に端末 1 はレポート A, B, C の更新を終了し、ロックレポート管理テーブルをクリアする。端末 2 は、クリアされたロックレポート管理テーブルを見て初めて自分のためにレポート B, D, E を登録し、更新が開始できるのである。

図 16 をよく見ると図 3 と同じであることに気付くであろう。つまり、あるレポートへの複数個の更新ランが同時に実行されることは決してありえず、矛盾した結果にならないのである。

6. リカバリ機能

4 章, 5 章において複数端末による同一レポート更新をどう制御すべきかを述べてき

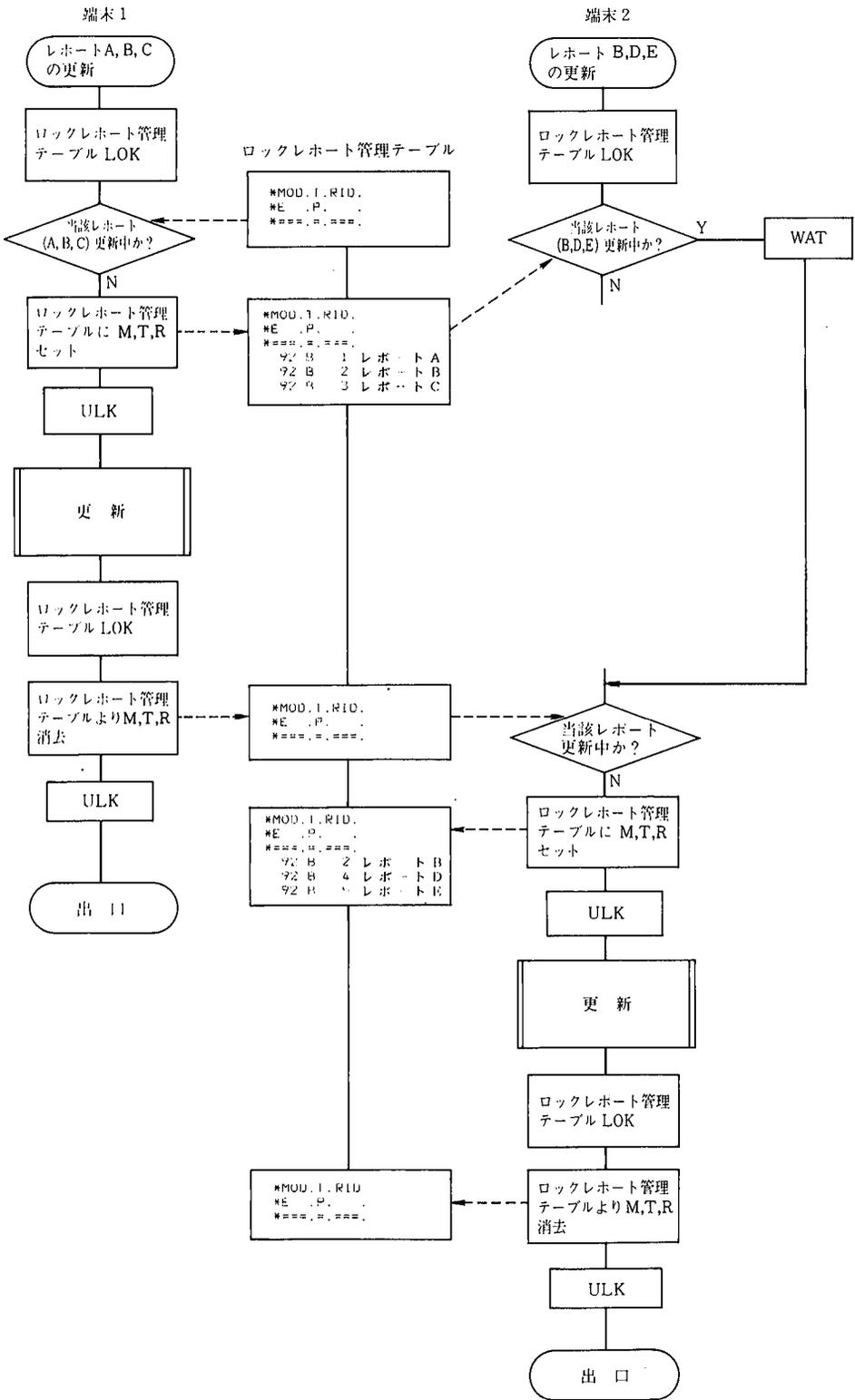


図 16 複数端末による更新の排他制御

Fig. 16 The control of report updation by multiple terminal

た。しかし、プログラム内でいかに考慮したとしても、プログラム外の要因による異常終了が発生すると、データ・レポートの内容に矛盾が生じる場合もありうる。プログラム外の要因とは、システムダウン、回線エラー、使用者による割り込みキーの入力等を指し発生頻度は高い。

レポート内容の矛盾と言っても、単一レポート内でみれば、MAPPER 1100 がリカバリしてくれるし、5レポート以内であればDFU命令で対処できる。それで、この章で問題とするのは6レポート以上の一括更新、またはロック中の禁止命令実行が要求される場合におけるレポート間の矛盾である。

たとえ異常終了が発生しても、レポート間の関係を保証するためにはレポートをリカバリ（更新前レポートへの復元）すればよい。そのリカバリ方法の一つに、異常終了発生後リカバリ・ランを実行することが考えられる。しかし、リカバリ・ランを実行する必要があるのかどうか、あるいはリカバリが必要な端末はどれか、リカバリ・ランの実行を失念することはないのか等、考慮すべき要素は多い。そこで、リカバリ・ランを独立して実行するのではなく、異常終了後の各ランの再実行時にラン自身がリカバリの必要性を判定して、必要に応じて自動リカバリするロジックを組み込むことを考える。その実行のためには更新時に、

- 1) 更新中の端末であることを識別できるようにしておく、
- 2) これから更新しようとするレポートの他端末からの使用禁止を明確にする。
- 3) 更新前レポートをすべて退避しておく、

という3点の考慮が必要である。

6.1 更新中端末の識別

端末で単にランが実行中であるだけでなく、ランユニットの範囲内、つまり更新中であることを識別できるようにする。これは、異常終了が発生した時にその端末でリカバリする必要があるかどうかの判断材料となる。この識別は、ロックレポート管理テーブルに自分のロックしたレポートごとに端末番号を書き込めばよい(図17)。

```
.DATE
.   《 《   ロックレポート管理テーブル   》 》
*MOD.T.RID.  端末.
 *E   .P.    .番号.
 *=== . =. ===. =====.
180 C   25 1581
324 G   14 1580
   92 B    1 1583
   92 B    2 1583
   92 B    3 1583
126 F   33 1582
      . . . . . END REPORT . . . . .
```

図 17 ロックレポート管理テーブル (その2)

Fig.17 Report-locking control table(2)

6.2 更新レポートの使用禁止

ランユニットとして独立させ、他のランからの影響を防ぐために、これから更新しようとするレポートをロックする必要がある。この手法は5.1節のロックレポート管

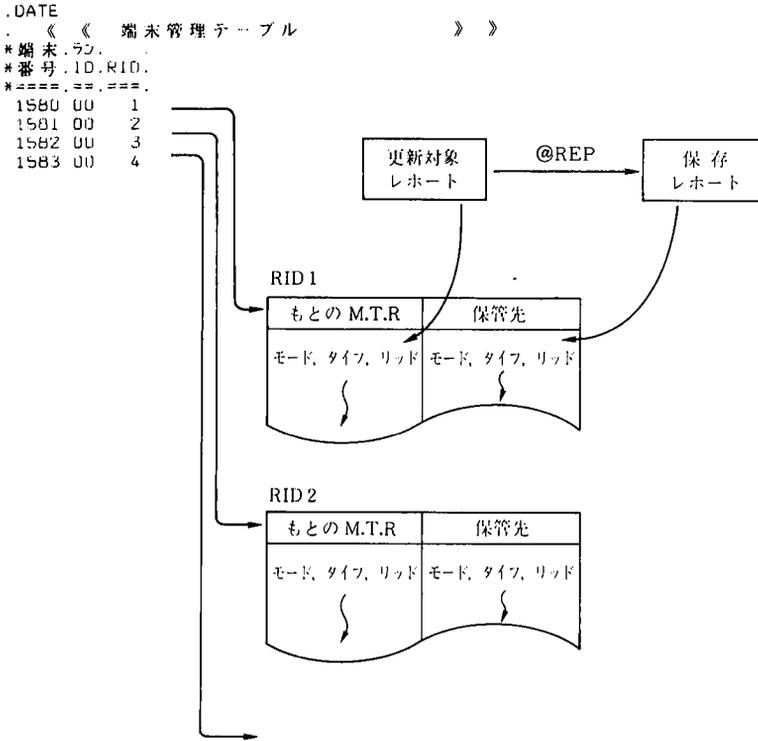


図 18 保存レポート管理関連図
Fig. 18 Report-saving control mechanism

理テーブルで述べた。

6.3 更新前レポートの退避

リカバリ機能を実現するために更新前レポートはすべて退避しておく必要がある。その際、どの端末がどのレポートをどこへ退避したかという情報を把握しておき、それが後述する自動リカバリにて使用される。

更新レポート数に制限を設けない汎用性のあるシステム、ということで考えた保存レポートの管理方法が図 18 である。

つまり、ランでは更新対象レポートの更新前イメージを任意のモード、タイプ、リッドに保存し、その情報を各端末専用のリッドに書き込んでおく。図 18 では、どの端末がどのレポートをどこへ退避したかを示している。

以上をもとに図 15 に工夫を加える。すなわち、

- 1) 更新処理前に更新前レポートを退避し、その退避先を登録する。
- 2) 更新処理後に、不要となった 1) の情報をすべてクリアする。

図 19 においてはどの時点で異常終了しても、自動リカバリ機能が実行されればレポートの正当性は保証される。

更新前後の処理は、外部サブルーチン化してラン作成の負荷を避けるべきであろう。そしてこのルーチンは、更新するすべてのランに組み込まなければシステム全体とし

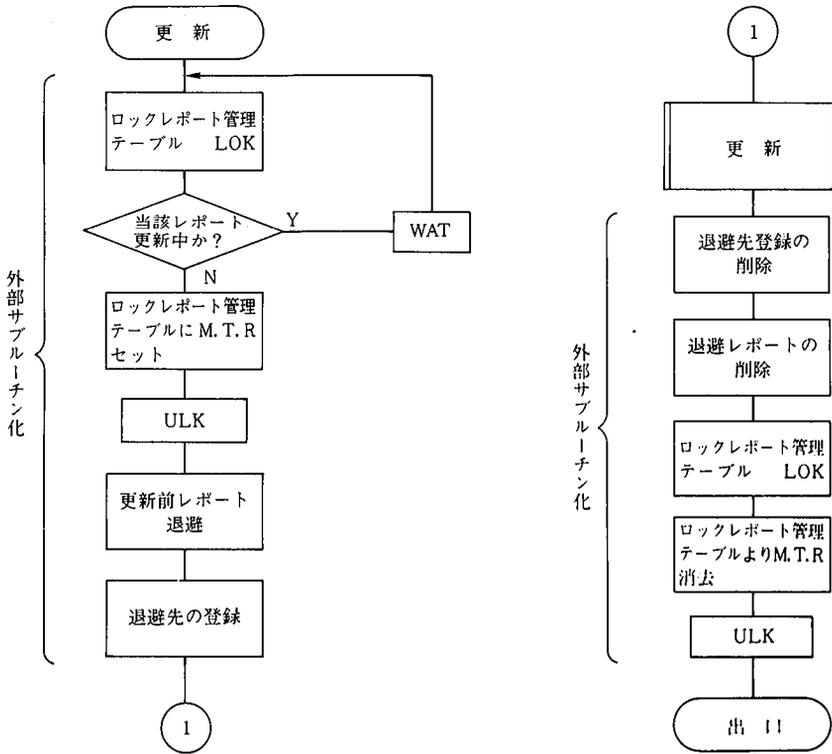


図 19 更新処理 (その2)

Fig. 19 The process of updatation (2)

ては意味をなさない。

6.4 リカバリ手順

たとえ異常終了が発生してもレポート間の関係を保証するためには、

- 1) 自動リカバリするための考慮をラン開始時に入れる、
 - 2) 自動リカバリさせるのは、更新中に異常終了が発生した場合のみである、
- という2点が必要である。

以上を前提に図 11 に工夫を加える。ラン開始時に自分の端末番号のラン ID が '00' でないということは、その端末で前回実行時に異常終了が発生していることを示す。しかし、ロックレポート管理テーブルを検索して自分の端末番号が存在しなければ、更新中でなかったことを意味するのでリカバリの必要はない。更新中であった場合のみ、ロック中レポートを復元すればよいのである (図 20)。

7. おわりに

以上述べてきたのは、

- 1) ランの起動制御
- 2) レポートの排他制御
- 3) リカバリ機能

を実現させるための考え方である。この3点を実現させるための最終的な環境とプロセスは以下の各図に示されている。

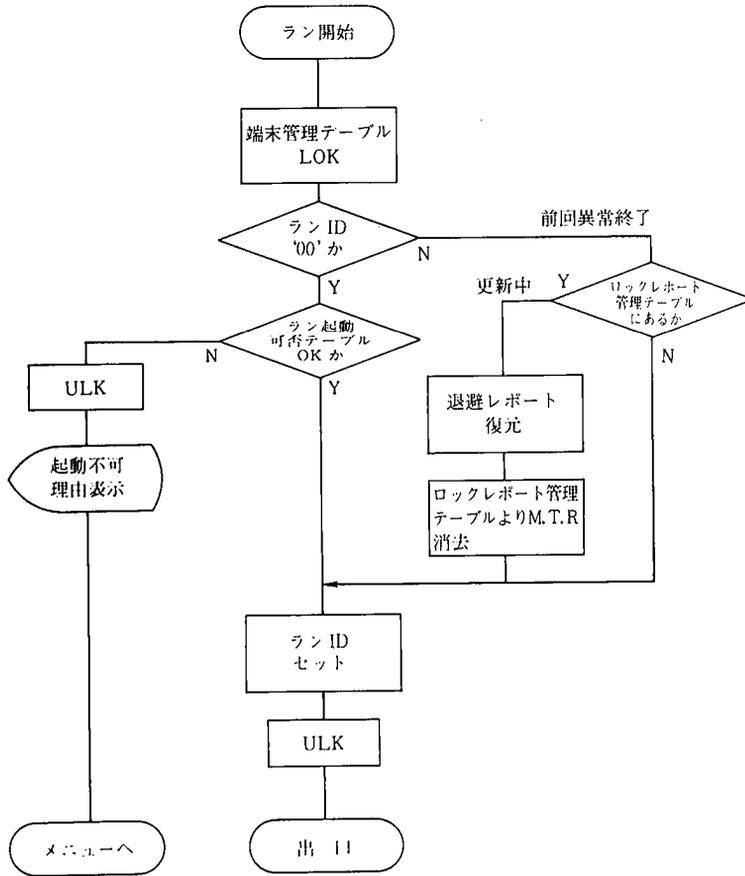


図 20 ラン開始時の処理 (その 3)

Fig. 20 The process at the time of run-starting (3)

- 1) 必要なテーブル類
 - ・ラン起動可否テーブル (図 5)
 - ・ロックレポート管理テーブル (その 2) (図 17)
 - ・保存レポート管理関連図 (図 18)
- 2) 必要な外部サブルーチン群
 - ・ラン終了時の処理 (図 8)
 - ・更新処理 (その 2) (図 19)
 - ・ラン開始時の処理 (その 3) (図 20)

本稿では、できるだけ汎用的に適用できるように話を進めてきたが、更新レポート名が特定されている場合や更新レポート数が限定されている場合等は、DFU 命令と組み合わせ、もっと簡単なロジックを用いシステム構築できるとも言える。

また、本稿では前提として、なぜ異常の発生した端末のみにリカバリを許している。そのため、ラン実行中あるいはレポート更新中のフラグが設定されたまま、万一その端末が故障等で物理的に使用できなくなった場合は、各テーブルを直接会話機能によって修正しなければシステム全体が動かない。異常の発生した端末でのみリカバリを

許しているのかは次の理由による。

- 1) 他端末からは、当該端末がラン実行中なのか、異常終了して停止しているのかを判別できない。
- 2) 原則として、その端末の操作者が自分で異常終了を認識し、自分で端末の処理を再実行すべきである。他端末より知らぬ間にリカバリされているのは問題がある。

しかし、端末の使用不能という事態には何らかの対策を準備しておくべきである。レポートの排他制御、あるいはレポートのリカバリに対する考え方は種々あるが、操作者がとくにリカバリを意識することなく、自動リカバリされる手法は大いに有効であり、本稿で述べた方法が多くのシステムで活用できるものと考える。

執筆者紹介 岡 島 伸 介 (Shinsuke Okajima)

昭和 55 年成蹊大学法学部法律学科卒業。60 年日本ユニバック(株)入社。金融機関向き経営情報システム FIQS 1100 等、主として MAPPER アプリケーションの設計・開発に従事。現在、名古屋支店システム 1 部に所属。



論文

人工知能と MAPPER

—Northwest 航空における事例

Artificial Intelligence and MAPPER

—The Case of Northwest Airlines: KEE + MAPPER

工藤 隆 司

要 約 MAPPER システム上の情報を, エキスパート・システムによる解析に用いた事例として, 『航空運賃割引管理エキスパート・システム (Airline Seat Advisor: SEATS)』がある。

SEATS は, 割り引き航空座席を管理して, 収益を増大するためのエキスパート・システムであり, Sperry (Unisys 社の前身), Northwest 航空, IntelliCorp の 3 社で開発された。

SEATS は, IntelliCorp 社の KEE を用いて作成されており, MAPPER 上の航空収益増加システム (Airline Revenue Enhancement: ARE) から, 割り引き座席を設定するかどうかを決定するために必要ないくつかの情報を受け取って, 推論し割り引き座席の設定に関し助言を行う。

KEE と MAPPER のインタフェースのソフトウェアを KEEMAP と呼ぶ。KEEMAP ソフトウェアは, KEE の上の KEEMAP 知識ベースと, Sperry PC (パーソナル・コンピュータ) 上のプログラム群, および MAPPER のランからなる。

本稿では, SEATS の事例をもとに, KEEMAP の概要について述べる。

Abstract Airline Seat Advisor (SEATS) is an expert system which uses information on MAPPER system. SEATS was developed by Sperry, Northwest Airlines, and IntelliCorp, and used for revenue enhancement by managing discount seats.

SEATS, implemented on KEE, gets some data to decide whether discount seats should be set up or not, from Airline Revenue Enhancement (ARE) system which is implemented on MAPPER, and gives us advice by using its inference engine.

The interface software between KEE and MAPPER is called KEEMAP. KEEMAP consists of KEEMAP knowledge base on KEE, some programs on Sperry PC and MAPPER runs on MAPPER system.

This paper describes the outline of KEEMAP using an example of SEATS.

1. はじめに

1.1 知識処理とエキスパート・システム

近年, 知識処理が急速に注目をあびている。この分野は, 先端的な研究の進行と同時に一部では実用的な技術として商用化も始まっている。知識処理は, 人工知能を基礎として登場してきた。

人工知能は人間の知能のあり方や思考過程を分析し, その活動を計算機上に実現しようとする試みである。人間の思考の研究は古くから心理学の研究テーマになっていて, 認知心理学という形で発展してきているし, 思考と言語の結びつきは強く, 言語学の側面から研究が進み, 自然言語理解等の形で発展してきている。一方, 情報処理の分野でも人工知能の歴史は古く, 計算機が世に出たのと時を同じくして始まった。

計算機科学者の最初の人工知能の興味はパズルを解くという形で現れた。この時代の課題は、効率的な探索機構の追求であり、評価関数を用いたり、最適ではないが多くの場合うまくゆくという発見的手法を用いて問題を解く試みであった。その他、人工知能分野で取り上げられているものに文字認識、形状認識、音声認識やパターン認識があり、人工知能はいろいろな分野と密接な係わりを持って発展している。

エキスパート・システムは人工知能の一分野を形成しており、知識工学を基礎としている。計算機を用いた初期の人工知能の問題が、ゲームなどの仮想化された問題であったのに対し、知識工学では個別的な狭い領域の現実の問題を対象としている。

歴史的には医療システムがその研究対象の一つであった。エキスパート・システムは特定の分野、たとえば医療のある領域に適用し、その領域のエキスパートとされる人から得た専門知識を利用し、推論することによって、エキスパートと同等の問題解決の能力を持つシステムである。

研究対象として、いくつかのエキスパート・システムが作成されたが、その試みによってわかったことは、狭い領域であっても、膨大な量の知識が必要であったということである。エキスパート・システムの基礎である知識工学は、「知識の表現」、「知識の利用（推論機構）」、「知識の獲得」が主な研究対象である。

エキスパート・システムの構成は、エキスパートの知識（知識とは、「事実」と、事実間の関係を表現するための「ルール」、アルゴリズムを表現するための「手続き」を総称したものである）を表現するための知識ベース、その知識をもとに推論を行う推論機構、使用者とのユーザ・インタフェースからなる。1985年頃から KEE 等の商用化されたエキスパート・システム・ツールが出現しており、各分野の専門知識を表現すれば、その領域のエキスパート・システムが容易に実現できる。

ところで、エキスパート・システムで用いられる知識は、人間にわかりやすい形で“記号”として表現される。したがって、この記号を直接扱うことのできる言語を利用できるのが望ましい。記号処理のための一つの記述言語として Lisp があり、Massachusetts 工科大学 (MIT) 等の米国の人工知能の研究は Lisp を中心に発展してきている。商用化されたエキスパート・システム・ツールも Lisp をベースにしているものが多い。しかし、Lisp はメモリを多く消費するという特徴を持っており、通常の計算機アーキテクチャでは扱いにくいいため、専ら人工知能関連の研究用の言語であった。このとき、研究用に開発使用された Lisp の専用機が Explorer*等の商用 Lisp 専用ワークステーションという形態で現在発展している。

1.2 汎用計算機とワークステーション

従来行われているソフトウェア開発/運用の形態は、一般的に汎用大型計算機という、ソフトウェア開発/運用に必要なほとんどの機能を集中した“資源”を用いて行われている。この資源はソフトウェア開発者や利用者の“共有の資源”であり、開発者や利用者は、端末機を利用してこの共有資源を利用する。ソフトウェア開発に必要なプログラムの編集機能でさえ汎用大型計算機に委ねられているために、端末機自体の機能は少なくすむ。

一方、ワークステーションは研究を主目的に発展したために、スタンドアローンで

* 日本ユニバックでは、KS-301 という名前で販売している。

利用できる機能を持ち、この意味で利用者“固有の資源”と言える。利用者固有の資源であるワークステーションは、プログラムの開発に必要な編集やコンパイルといったもの以外に、ファイル管理機能やネットワーク機能も備えた多機能で、比較的高価格といった特徴を持っている。しかし、大容量のデータを扱ったり、TSSでの利用には不向きである。

今までに研究開発されたエキスパート・システムの多くは、ワークステーション上で作成されている。他のワークステーションとの通信は、イーサネット等のローカルエリア・ネットワークで実現しているため、一般に汎用大型計算機との親和性は良くない。

1.3 従来のプログラムやデータベースとエキスパート・システムとの違い

従来のプログラミングは、データ構造の定義とアルゴリズムの記述で行っていた。従来のプログラムがエキスパート・システムと比べ、知的でないかという決してそうではなく、十分知的なのであるが、知識がプログラム中に散りばめられていて、知識とそれを利用するアルゴリズムとのモジュール性が良くない。したがって知識が変化すると、プログラムを書き直す必要が生じ、ソフトウェアの保守上好ましくない。これを避けるには、多少知識の違う多くのプログラムを作らなくてはならないということに成りかねない。しかし、処理効率は一般に良い。

一方、エキスパート・システムを眺めてみると、知識とその知識を扱う推論機構が明確に分離されていて、処理効率は悪いが、知識とそれを利用する推論機構とのモジュール性が良い。

したがって知識の変更や追加、削除が容易であり、変化しやすい知識を扱うのに向いている。さらにエキスパート・システム・ツールでの開発を考えると、推論機構はすでに備っているのだから、知識の記述をすることが、すなわちエキスパート・システムの開発をすることと等しい。

一般的なデータベースは、知識のうちの事実の集合と、その検索機構という構成からなり、一般に大容量のデータ（同じ種類の事実の繰り返し）を対象として効率の良い検索が行える。

さて、エキスパート・システムが受け入れられにくい理由の一つに、汎用計算機上の知識が扱いにくかったという点あげられる。もし、汎用計算機上のデータベースのデータをエキスパート・システムの推論で用いることが可能となれば、その応用範囲は広がるであろう。

この一つの事例として、Northwest 航空の「航空運賃割引管理エキスパート・システム」がある。航空運賃割引管理エキスパート・システムは Explorer 上に KEE を用いて作成されており、推論時に必要となるデータを MAPPER システムから得て実行する。

本稿では、この事例をもとに、

- 1) コンピュータ・システムの中における、エキスパート・システムのありかた
- 2) エキスパート・システムと MAPPER との接続事例

について述べる。

2. 航空運賃割引管理エキスパート・システム

このシステムは Airline Seat Advisor (SEATS) と呼ばれ、航空運賃の割り引き管理を行って、収益を増加させるためのエキスパート・システムである。SEATS は、Sperry, Northwest 航空, IntelliCorp の 3 社によって開発された。このシステムが開発された背景としては、開発当時 Sperry 社が、IntelliCorp 社の KEE を用いたエキスパート・システムの技術と能力の調査と、実際に役立つ良い問題を探していたことや、KEE 使用における技術開発に興味を持っていたことがあげられる。このシステムで扱う問題は次のとおりである。

2.1 問題の性質

1978 年の航空事業の規制解除に伴って、航空料金の自由設定ができるようになった。割り引き料金を設定することにより、競合他社と対抗でき、休暇旅行者などの任意旅行者を増やすことができる。

航空料金の割り引きの率やその種類は、最大の利益を上げるように管理される必要がある。割り引き座席が多すぎると、ビジネスなどで使用する強制旅行者に不必要な割り引きをしてしまうし、割り引き座席が少ないと、競合他社に客をとられたり任意旅行者が減少し、いずれの場合も収益減となる。割り引き座席の管理は、航空会社にとって比較的新しく、かつむずかしく徐々に変化していく分野である。

割り引き座席数に影響する要因は、次のとおりである。

- 1) 出発までの日数
- 2) 競合（競合の有無、料金、出発時間、サービス、等）
- 3) 地域間あるいは、都市間の市場の性質（ビジネス路線か観光路線か）
- 4) 出発時間（時刻、曜日、季節、季節休暇、等）
- 5) 過去の予約パターン
- 6) 現在の予約状況

これらの要因どうしの関連や要因間の関係は変りやすい。さらに客や競合他社の行動も明らかでない。これらの要因を考慮して割り引き座席を管理する専門家のアプローチもそれぞれ異なり、その方法も経験の積み重ねとともに変化する。

このように監視すべき項目が多数にのぼり、それが増加傾向にある。ちょっと大きな航空会社では、1 日当たり 1500 便くらいの出発便があり、出発日前 15 日間について管理し、各便ごとに二つの割り引きクラスを設定すると、割り引き座席を用意すべきかどうかという判断を毎日 45000 回下さなければならない。いくつかの航空会社では、200 人以上の専門家がこの分野に携わっている。

この問題の特徴は上に述べたように、たくさんの要因および要因間の関係があり、簡単な解析的手法で解を求めるのはむずかしいということである。さらに現在の専門知識が変化しやすい状況にある。したがって、この分野の専門家の知識の変化に耐える必要がある。現在のエキスパート・システムの技術は、変化に対応して素早くプロトタイピングをしたり、複雑な知識を表現する機能を持っている。

この分野は的確な判断に負うところが大きいのが、判断条件が多大なことや判断に必要な要因が多いことから、問題の解決にはエキスパート・システムだけでは不十分であり、従来のデータ処理システムとの連携による解決が必要である。

2.2 SEATS の概要

SEATS は、現在ある知識を利用して、特定の便に対して割り引き座席を「増加」、「減少」あるいは「現状維持」させるかどうかを決定する。座席の増加あるいは減少といった結論を指示するような要因が弱まりつつある時は「増減をほどほどに」という結論を出す。このような提案を行うことによって、収益の増加を計るのである。

2.3 SEATS の構造

SEATS と他システムの概念的な関係は次のとおりである (図1)。

航空座席予約システムは大容量データベースで実現されていて、Northwest 航空の座席予約の情報だけでなく、競合他社の座席予約の情報も入っている。

航空収益増加システム (Airline Revenue Enhancement : ARE) は、MAPPER のアプリケーション・プログラムである。このシステムは、各航空便の予約パターンを分析して、専門家に対して注視すべき点を指示するシステムである。

SEATS は ARE からの例外ケースを処理し、割り引き座席数の増減を提案する。SEATS と ARE の間のデータの受け渡しは、KEE と MAPPER のインタフェースである KEEMAP が行う。

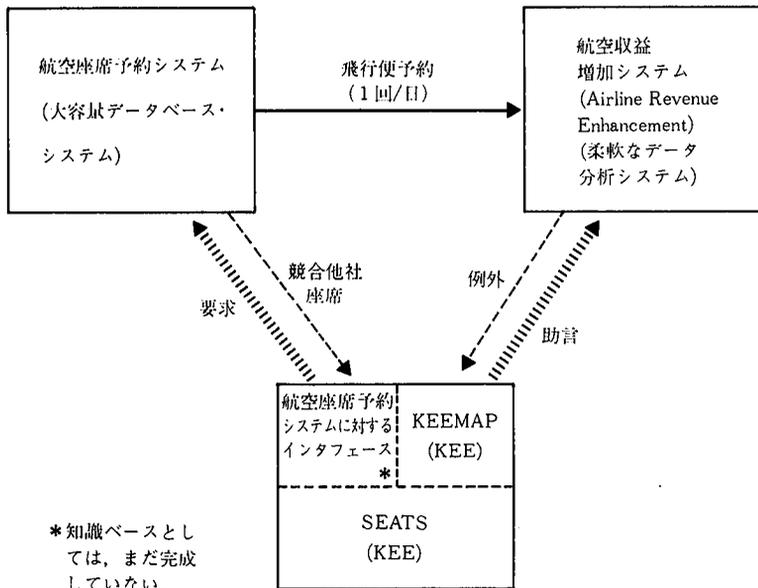


図1 他システムとの接続

Fig.1 Connecting to other systems

SEATS と KEEMAP のインタフェースは、MAPPER システム内のどのレポート (ファイル) のどの行 (レコード) をアクセスするかという問い合わせに関する情報と、その行で、どのデータ項目が欲しいかという、データ項目に関する情報の2種類である。KEEMAP 自身は MAPPER からの例外ケースのデータを受け取り、それを SEATS に渡すが、項目としては、

- ①便番号, ②出発空港, ③行き先, ④出発日, ⑤出発時間, ⑥収容乗客数, ⑦全予約数, ⑧割り引きクラスの予約数, ⑨追加された割り引き座席数, ⑩便全体と割り引

きクラスに対する過去の予約状況データ、などがある。

SEATS はこれらのデータをもとに、専門家の知識を用いて割り引き座席の増減を提案する。

3. KEE と MAPPER のインタフェース : KEEMAP

KEE と MAPPER のインタフェース・ソフトウェアを、KEEMAP ソフトウェアと呼ぶ。

3.1 KEEMAP ソフトウェア

KEEMAP ソフトウェアは、KEE 上の KEEMAP 知識ベース、Sperry PC 上のプログラム群、および MAPPER 上の MAPPER ランからなる。KEEMAP ソフトウェアについて述べる前に、KEE の概要について述べる。

3.1.1 KEE の概要

KEE はエキスパート・システム構築のための開発支援ツールである。エキスパート・システムの 3 要素である、知識ベース、推論機構、ユーザ・インタフェースのうち推論機構はすでに備っているため、使用者は知識ベースとユーザ・インタフェースの構築を行うことでエキスパート・システムのプロトタイプピングができる。

KEE で用いている知識表現法はフレーム型の知識表現と、プロダクション・ルール型の知識表現である。フレームのことを KEE の世界ではユニットと呼び、抽象的な“対象”や、具体的な“対象”を表現するために用いる (図 2)。

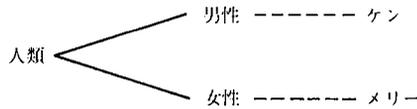


図 2 「人類」のフレーム表現

Fig. 2 Frame representation of "Mankind"

図 2 の例は、人類に関する簡単なフレーム表現であり、「人類」や「男性」、「メリー」はそれぞれ“対象”を表現している。さらに、このグラフは概念の包含関係を表している。左側の「人類」が最も抽象度の高い“対象”を表現しており、「ケン」や「メリー」は「人類」よりも抽象度が低く、この図の場合は、具体的な対象 (インスタンスという) を表現している。

- 1) グラフでの実線の先は抽象度の低い対象を表現し、KEE の世界ではサブクラスと呼ばれる。
- 2) このグラフでの破線の先は具体的な対象を表現し、KEE の世界ではメンバ (インスタンスと同意) と呼ばれる。

各対象には属性を定義でき、この属性を KEE の世界ではスロットと呼ぶ。たとえば、対象に、身長や体重、血圧といった属性を定義できる。フレームの一つの機能に属性の継承 (インヘリタンス) というものがあり、抽象度の高い対象に属性を定義しておけば、その属性が継承される。したがって「人類」の所に、身長や体重、血圧といった属性を定義しておくことによって、「ケン」と「メリー」まで継承することができる。しかし、継承はメンバのところまでストップする。これらのフレームとスロット

は、知識のうちの「事実」に相当する。ルールは、KEEではプロダクション・ルールで表現される。たとえば、「最高血圧が140を越えたら、高血圧である」という知識は、プロダクション・ルールでは次のように表現できる。

```
(IF (> (THE 最高血圧 OF ?被験者)
```

```
140)
```

```
THEN (THE 症状 OF ?被験者 IS 高血圧))
```

KEEでは?で始まる名前は変数を意味し、ユーザは任意の名前を使用できる。たとえば、上記のルールが推論で用いられると、?被験者は最高血圧というスロットを持つ任意のユニットと結合され、条件部が真であれば、そのユニットの症状という名前のスロットに高血圧という値が代入される。

その他の知識である「手続き」は、Lispプログラムのコードとして表現される。手続きのことを、KEEではメソッドと呼ぶ。メソッドの名前は、スロット名として表現され、プログラムのコードは、スロット値として保存される。メソッドの起動は、メソッドが定義されているユニットに対して、任意のメソッド名(スロット名)を指定する(メッセージを送る)ことによって行われる。具体的には次の型式である。

```
(UNITMSG <ユニット名> <スロット名> <引数1> … <引数n>)
```

この方法はユニットで表現された対象、すなわちオブジェクトに対して、メッセージ、すなわちメソッド名を送ることによって手続きを実行する形態であり、この方法によるプログラミングを、オブジェクト指向プログラミングという。概念的には次の型式となる。

```
(UNITMSG <オブジェクト> <メッセージ> <引数1> … <引数n>)
```

3.1.2 KEEMAP ソフトウェア概要

KEEMAP ソフトウェアは、Sperry と IntelliCorp により開発されたソフトウェア・パッケージのプロトタイプである。このパッケージの目的は、KEEのユーザにMAPPER レポートの検索を許し、このデータをもとにKEEでの分析を可能とするものである。

基本的な機能としては次のものがある。

- 1) MAPPER と KEE との間の会話型処理
- 2) MAPPER から KEE へのレポートの転送
- 3) KEE からの MAPPER レポートの作成

ただし、現在のところ KEEMAP ソフトウェアはプロトタイプ・システムであり、完成されたシステムではない。

KEEMAP ソフトウェアの現在のものは Explorer 上の KEE バージョン 2.1.64 と、MAPPER 5*リリース 1 R 2 で動作する。

Explorer と MAPPER の実際の接続は、Sperry PC を介して行われる (図 3)。

Explorer と PC の接続は、非同期方式で RS 232 C を用いて行われるが、PC にはスペリー・ターミナル・エミュレーション・プログラム (Sperry Terminal Emulation Program: STEP) と、それに必要な STEP ボードを備えなくてはならない。一方、PC と MAPPER 5 は、同期方式の Uniscope プロトコルで接続する。

* 日本ユニパックの製品では MAPPER 6 に相当する。

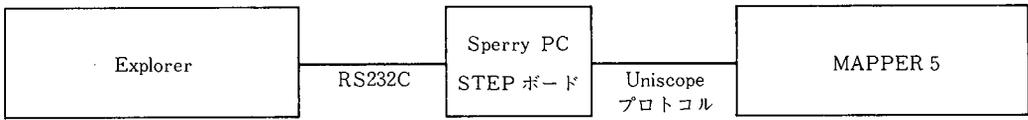


図3 ハードウェア構成
Fig.3 Physical connection

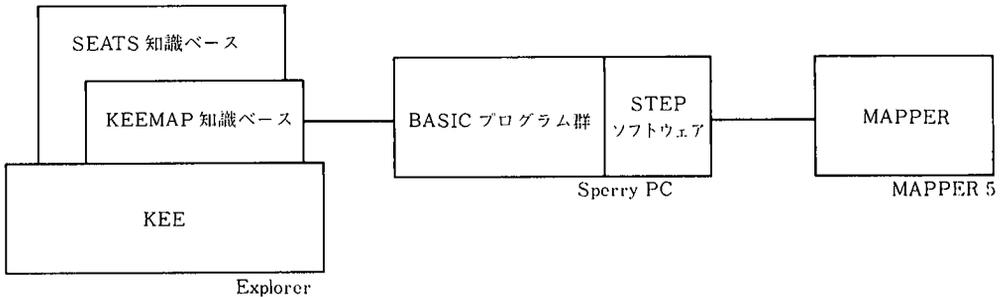


図4 KEEMAP のソフトウェア構成
Fig.4 Software components of KEEMAP

KEEMAP ソフトウェアを用いた SEATS のソフトウェア構成は、図4 の通りである。

Explorer 上には KEE を用いて KEEMAP 知識ベースが構築されている。さらに SEATS は、KEEMAP 知識ベースの上に構築される。PC 上には MAPPER と通信するために作成された STEP プログラムと、STEP プログラムと KEEMAP とのデータの受け渡しを行う BASIC プログラムが必要である。さらに、MAPPER 上には MAPPER ランが必要である。

KEEMAP 知識ベースと、PC 上のプログラム群、および、MAPPER 上の MAPPER ランを総称して、KEEMAP ソフトウェアと呼ぶ（以下、単に KEEMAP と書いた際には、KEEMAP ソフトウェアを指すものとする）。

KEEMAP の通信形態には、バッチ型のコミュニケータ (BATCOMM) と、会話型のコミュニケータ (INTCOMM) の2種類がある。BATCOMM は PC 上で動作する STEP プログラムのファイル転送機能を用いている。Explorer から送られてきた MAPPER のコマンドの一群は、PC の MS-DOS 上の BASIC プログラムに転送され、PC 上のローカルな DOS ファイルとして保持される。

ファイルはその時点で、STEP プログラムを通じてバッチ・ジョブとして MAPPER に引用され、リクエストが終了したとき、MAPPER からのアウトプットは他の DOS ファイルに書かれる。最後にこのファイルは、BASIC プログラムを通じて Explorer 上の KEE に転送される。

BATCOMM の特徴は、STEP プログラムとそれをサポートする BASIC プログラムが、同時には動作しないということである。すなわち、KEE と PC 上の BASIC プログラムの間、および PC 上の BASIC プログラムと STEP プログラムの間の情報の受け渡しは、DOS ファイルという“バッファ”を用いているということである。

一方、INTCOMM は STEP プログラムの User Program Interface (UPIF) 機能を通じて、KEE と MAPPER を接続する。KEE は、MAPPER と交信する際に BASIC プログラムへ直接コマンドを発行する。会話形態は、通常の MAPPER の使用方法とほぼ一致する。

SEATS で用いている KEEMAP はバッチ型の BATCOMM であり、本稿では、BATCOMM を中心に述べる。

3.1.3 KEEMAP 知識ベース

図 5 が KEEMAP 知識ベースである。

KEEMAP 知識ベース上には、必要な MAPPER のモード、RID、タイプの定義(レ

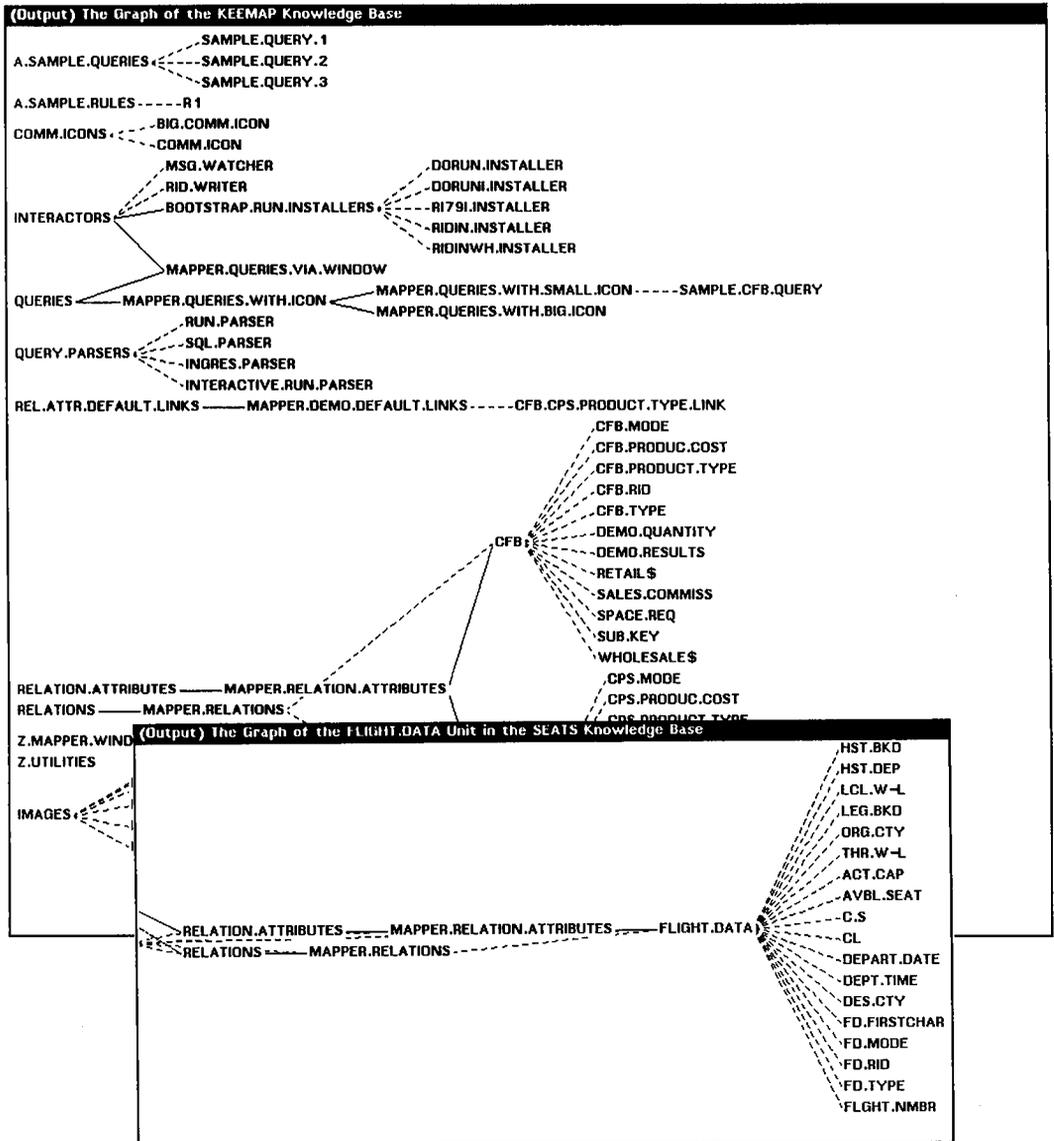


図 5 KEEMAP 知識ベース
Fig. 5 KEEMAP knowledge base

ポートの指定)と、必要なデータの項目の定義(レコードの中の項目の位置の指定)を行う。この定義は、MAPPER.RELATIONSのメンバ・ユニットであり、かつMAPPER.RELATION.ATTRIBUTESのサブクラス・ユニットである使用者の定義するユニットで行う。このユニットのことをクエリ・ユニット(後述)と言い、このユニットのメンバ・ユニットに、MAPPERから取り出す情報を定義する。図5中のFLIGHT.DATAユニットは、SEATSで必要とするクエリ・ユニットである。このFLIGHT.DATAユニットのメンバ・ユニットであるFD.MODEや、FD.RID、FD.TYPEがそれぞれSEATSで必要となる、モード、RID、タイプに該当する。その他、SEATSで必要なデータ項目に、便番号、出発空港、行き先等があり、それぞれ、FLIGHT.NMBR、ORG.CTY、DES.CTYといったユニットで定義される。これらのユニットの中に、データの存在場所を定義するためのスロット(必要な項目が、そのレコードの何かラムから始まるのかを指定する)が用意されている。

BATCOMMでは、MAPPERから受け取ったデータを保存する場所として、MAPPER.QUERIES.WITH.SMALL.ICON、またはMAPPER.QUERIES.WITH.BIG.ICONのメンバ・ユニットが必要で、SEATSではこのメンバ・ユニットとして、FLIGHT.DATA.CACHEDと、FLIGHT.DATA.CACHED.DRY.6の二つのユニットが存在する。このユニットの中に、CACHEというスロットがあり、MAPPERから得られたレポート・ファイルが入れられる。このスロットの働きは、レポート・ファイルを保持するだけではない。他のMAPPERのコマンドが、同一レポートを検索しにいった場合には、再びMAPPERをアクセスせずに、このスロットからレポートをアクセスし、効率を高めている。すなわち、キャッシュ・メモリの役割を演じているのである。

INTCOMMでは、MAPPER.QUERIES.VIA.WINDOWのメンバ・ユニットが必要である。この他、レポートの取り扱いに関するMAPPERのランを定義しているユニットがある。BOOTSTRAP.RUN.INSTALLERSの五つのメンバ・ユニットがそれである。

DORUN.INSTALLER、RIDIN.INSTALLER、RIDINWH.INSTALLERはBATCOMMに必要なMAPPERランを定義しているし、DORUNI.INSTALLERとRI79I.INSTALLERの二つはINTCOMMに必要なランを定義している。

DORUNで定義されるMAPPERランは、MAPPERからのレポートをMAPPER.QUERIES.WITH.ICONのサブクラス・ユニットのメンバ・ユニットの中の、CACHEスロットに代入する。

RIDINとRIDINWHは、KEEからMAPPERへのレポートの挿入をするために必要である。この二つの違いはレポートのヘッダの取り扱いにある。RIDINはそのレポート・タイプのRID0からヘッダの情報を得て、新しいRIDのヘッダとするが、RIDINWHは、それ自身のファイルのヘッダを用いる。

DORUNIは実際には、BATCOMMとINTCOMMの両方で使用されていて、得られたMAPPERレポートをExplorer上のファイルに書き込む。このファイルにはヘッダとEND REPORTラインを含んでいる。

RI79I(これはRID IN 79 CHARACTERS INTERACTIVEの頭文字をとった)

は、プロンプトに対して、モード、RID、タイプを答えることによって、そのレポートを KEE 上のウィンドウに表示するためのものである。

SEATS では DORUN を用いた。

3.1.4 PC 上のファイル

PC 上には、まず STEP プログラムと、RS 232 C 用の STEP ボードを備える必要がある。さらに、環境設定に、ACONF ファイルが必要となる。

1) ACONF ファイル

BATCOMM を行う時に STEP プログラムによって利用されるコンフィギュレーション用のファイルが ACONF である。このファイルは、通信の形態を定義するために必要であり、次のようなコンフィギュレーション用のパラメタを含んでいる。

- ① クラスタ・リモートアイデンティファイア (CLUSTER RID)
- ② Explorer のウィンドウ用のサイト・アイデンティファイア (SID)
- ③ スクリーン・バイパス、MAPPER とのコネクションに必要なタイプと、接続ポー (baud) レート

PC プログラムには SCONF という、ACONF の元になるファイルがすでに存在していて、STEP ソフトウェアで供給される PC のプログラム CONFIG を実行することで ACONF を作成できる。

この他、STEP プログラムと KEEMAP 知識ベース間の情報のやりとりに行くつかのバッチ・ファイル(.BAT のついたファイル)と、BASIC プログラム(.BAS のついたファイル)が必要である。

2) バッチ・ファイル

- ① BATCOMM.BAT: BATCOMM を起動するためのバッチ・ファイルであり、モードのセットをした後、制御を RUNLOOP.BAT に渡す。
- ② RUNLOOP.BAT: BASIC プログラムの RCVRUN.BAS を起動し、RCVRUN.BAS によって書かれたバッチ・ファイル RUNLINK.BAT に制御を移す。
- ③ INTCOMM.BAT: INTCOMM を起動するためのバッチ・ファイルであり、STEP プログラムと COUPLE.BAS を起動する。

3) BASIC プログラム

- ① RCVRUN.BAS: BATCOMM のためのメイン・コントロール・プログラムであり、MAPPER ランを受け取って CMDFILE というローカルな DOS ファイルに書き込む。CMDFILE は、ACONF のコンフィギュレーションの時に MAPPER によって利用されるコマンドが入る。さらに RUNLINK.BAT というバッチ・ファイルを生成する。RUNLINK.BAT には、いくつかのコマンドが入っている。

RUNLINK.BAT は STEP プログラムを起動して MAPPER を実行し、結果を RCVFILE という DOS ファイルに書き込む。

RCVRUN.BAS は最後に SNDRSLT.BAS を起動する。

- ② SNDRSLT.BAS: BATCOMM のための BASIC プログラムであり、

RCVFILE に書かれた MAPPER レポートを RS 232 C ポートに返す。

- ③ COUPLE.BAS: INTCOMM のための BASIC プログラムであり, KEE と MAPPER のコミュニケーションを行う。

3.1.5 MAPPER ランのインストール

MAPPER ランのインストールは, KEEMAP 知識ベースの BOOTSTRAP.RUN, INSTALLERS の五つのメンバにメッセージを送ることによって行われる。たとえば, DORUN をインストールしたければ, DORUN.INSTALLER ユニットに, メッセージ“TRANSMIT. BOOT. RUN”を送ればよい。

3.2 KEEMAP の動作

KEEMAP 知識ベースの構築と, PC 上のいくつかのプログラムやファイルの準備, 並びに MAPPER ランの準備が完了すれば KEE から MAPPER を操作することができる。ここでは BATCOMM の形態で, DORUN の MAPPER ランについて図を用いて動作を説明する (図 6)。

使用者が PC 上で BATCOMM.BAT のバッチ・ファイルに制御を渡すと, モード

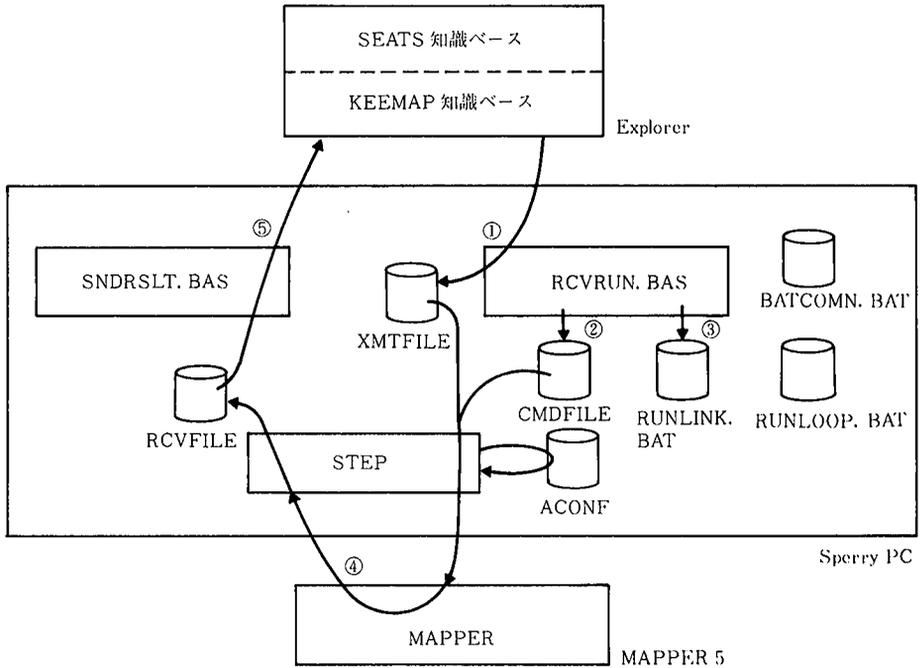


図 6 KEEMAP の動作

Fig. 6 Behavior of KEEMAP

をセットした後 RUNLOOP.BAT に制御が移される。RUNLOOP.BAT には RCVRUN.BAS の起動と, RCVRUN.BAS により書かれる RUNLINK.BAT の起動のコマンドが入っている。

RCVRUN.BAS は RS 232 C ポートからのログイン・ストリングや, RID, タイプと一連の MAPPER コマンド群を受け取り, XMTFILE という DOS ファイルに書き込む①。

RCVRUN.BASは、DORUNのMAPPERランを開始するコマンドと、XMT-FILEを転送するコマンド、さらにMAPPERの実行を終了させて、その結果をRCVFILEというDOSファイルに書き込むコマンドを、CMDFILEに書き込む(②)。

CMDFILEは後にSTEPプログラムにより参照される。その次にRCVRUN.BASは、STEPをACONFの環境で起動するコマンドと、結果のRCVFILEをRS 232 Cポートに返すためのSNDRSLT.BASを起動するコマンドをRUNLINK.BATに書く(③)。

次にRUNLINK.BATに制御が移され、STEPの実行が開始される。STEPプログラムは、MAPPERでの実行に必要な情報をCMDFILEより参照して、実際にMAPPERの検索を行い、MAPPERのレポートをRCVFILEというDOSファイルに書き込む(④)。

RUNLINK.BATの次のコマンドは、SNDRSLT.BASを起動する。SNDRSLT.BASはRCVFILEにあるMAPPERレポートをRS 232 Cポートに返し、KEEMAP知識ベースはその結果をMAPPER.QUERIES.WITH.ICONユニットの子供のメンバ・ユニットのCACHEというスロットに書き込む(⑤)。この実行のサイクルは、KEEMAP知識ベースからMAPPERの実行を終了するコマンドがくるまで続けられる。図7の中の、FLIGHT.DATA.CACHEDユニットのCACHEスロットには、SEATSがMAPPERから検索してきた航空便の情報が入っている(図7)。

3.3 MAPPER レポート検索用の拡張された KEE の TellAndAsk 言語

KEEでは知識ベースの検索に、関数による参照方法と、自然言語形式の表現によって検索を行うことのできるTellAndAsk言語の二つの方法が提供されている。

このTellAndAskを拡張することで、MAPPERのレコードを検索することが可能となる。KEEMAPにはTellAndAsk知識ベースを拡張するユニットが入っており、DB.OPERATORS、DBRETRIEVE、FROM.RELATION、ATTRIBUTEの四つの新たなユニットをTellAndAsk知識ベースに含めることで、MAPPERの検索ができる(図8)。

このようにして、三つの新しい述語が使用可能となる。その一つがDBRETRIEVEであり、データベースの問い合わせを行う。論理的にはDBRETRIEVEは、論理演算子のANDと同等の働きをする。DBRETRIEVEの中で使用できるS式(Lispで用いることのできる式を、とくにS式と言う)は以下に述べる三つであり、一般形は次の通りである。

(DBRETRIEVE

<型式1>

<型式2> … <型式2>)

<型式1>は、以下に述べる1)の型式のS式である。

<型式2>は、以下に述べる2)または、3)の型式のS式である。

1) (?X IS FROM. RELATION <DD. UNIT. OR. REF>)

このS式は、<DD. UNIT. OR. REF>ユニットによって定義されたMAPPERレポートの行を取り扱う。すなわちレポートの行の検索を行い、?Xは現在の行を意味する。ただし、<DD. UNIT. OR. REF>のユニットはMAPPER.

```

The FLIGHT.DATA.CACHED Unit in SEATS Knowledge Base
Unit: FLIGHT.DATA.CACHED in knowledge base SEATS
Created by carhart on 14-Aug-85 3:44:55
Modified by carhart on 4-Feb-1986 13:17:41
Member Of: FLIGHT.DATA.QUERIES.CACHED

Own slot: CACHE from FLIGHT.DATA.CACHED
Inheritance: OVERRIDE.VALUES
ValueClass: (UNION (LIST in kb KEEDATATYPES) (ONE.OF NIL))
Comment: "A list of tuples of retrieved values. When the CACHE.FLAG is ON (default), retrieved records are added to the cache by RETRI
EVE.RECORD so that they may be 'regurtiated' later if this query is reinvoiced. Each element in the cache list has same form as the
value of the slot RETRIEVED.VALUE.SET. Note that there is a distinction between a NIL value and an unknown value in this slot - NIL
means the cache is empty whereas unknown means there is no cache."
Values: ((0 0 138 28 146 T CLE 710 BOS 850801 34 22 1 409) (0 0 6 4 10 V CLE 710 BOS 850801 6 1 1 409)
(1 0 78 86 146 T CLE 955 BOS 850801 80 84 1 87) (1 0 0 25 25 V CLE 955 BOS 850801 23 18 0 87)
(1 0 114 52 146 T CLE 1405 BOS 850801 58 56 1 411) (1 0 -1 21 20 V CLE 1405 BOS 850801 19 14 0 411)
(5 1 12 153 144 T CLE 1730 BOS 850801 151 133 1 53) (5 1 -2 42 40 Q CLE 1730 BOS 850801 41 33 0 53)
(0 0 140 26 146 T CLE 710 BOS 850802 33 34 1 409) (0 0 7 8 15 V CLE 710 BOS 850802 10 7 1 409)
(4 0 85 79 146 T CLE 955 BOS 850802 103 115 1 87) (4 0 0 25 25 V CLE 955 BOS 850802 23 19 0 87)
(11 4 81 85 146 T CLE 1405 BOS 850802 80 82 1 411) (11 4 0 25 25 V CLE 1405 BOS 850802 21 17 0 411)
(6 0 22 146 144 T CLE 1730 BOS 850802 159 171 1 53) (6 0 -2 42 40 Q CLE 1730 BOS 850802 42 33 0 53)
(0 0 123 43 146 T CLE 710 BOS 850803 48 37 1 409) (0 0 0 25 25 V CLE 710 BOS 850803 20 14 0 409)
(28 14 57 107 146 T CLE 955 BOS 850803 103 101 1 87) (28 14 -12 37 25 V CLE 955 BOS 850803 31 26 0 87)
(16 10 97 69 146 T CLE 1405 BOS 850803 83 72 1 411) (16 10 -9 39 30 V CLE 1405 BOS 850803 40 30 0 411)
(1 0 34 134 144 T CLE 1730 BOS 850803 139 141 1 53) (1 0 -2 42 40 Q CLE 1730 BOS 850803 41 35 0 53)
(2 1 63 101 146 T CLE 955 BOS 850804 104 100 1 87) (1 0 0 30 30 V CLE 955 BOS 850804 29 25 0 87)
(10 5 86 80 146 T CLE 1405 BOS 850804 105 91 1 411) (10 5 -4 24 20 V CLE 1405 BOS 850804 26 36 0 411)
(6 0 14 154 144 T CLE 1730 BOS 850804 173 93 1 53) (6 0 -1 41 40 Q CLE 1730 BOS 850804 43 29 0 53)
(0 0 134 32 146 T CLE 710 BOS 850805 42 42 1 409) (0 0 7 8 15 V CLE 710 BOS 850805 9 9 0 409)
(2 1 72 89 146 T CLE 955 BOS 850805 95 97 1 87) (1 1 0 25 25 V CLE 955 BOS 850805 24 23 0 87)
(0 0 86 80 146 T CLE 1405 BOS 850805 65 50 1 411) (0 0 -2 22 20 V CLE 1405 BOS 850805 18 11 0 411)
(1 0 42 123 144 T CLE 1730 BOS 850805 134 136 1 53) (1 0 -1 41 40 Q CLE 1730 BOS 850805 41 38 0 53)
(0 0 134 32 146 T CLE 710 BOS 850806 23 19 1 409) (0 0 2 8 10 V CLE 710 BOS 850806 6 3 1 409)
(2 1 89 75 146 T CLE 955 BOS 850806 76 92 1 87) (0 0 0 25 25 V CLE 955 BOS 850806 19 19 1 87)
(1 0 118 48 146 T CLE 1405 BOS 850806 59 69 1 411) (1 0 5 15 20 V CLE 1405 BOS 850806 17 15 0 411)
(0 0 73 95 144 T CLE 1730 BOS 850806 107 130 1 53) (0 0 4 36 40 Q CLE 1730 BOS 850806 39 39 0 53)
(0 0 138 28 146 T CLE 710 BOS 850807 26 31 1 409) (0 0 8 2 10 V CLE 710 BOS 850807 3 5 1 409)
(0 0 96 68 146 T CLE 955 BOS 850807 77 85 1 87) (0 0 5 20 25 V CLE 955 BOS 850807 22 16 0 87)
(0 0 113 53 146 T CLE 1405 BOS 850807 64 62 1 411) (0 0 4 16 20 V CLE 1405 BOS 850807 20 12 0 411)
(3 1 15 148 144 T CLE 1730 BOS 850807 136 151 1 53) (3 1 -7 47 40 Q CLE 1730 BOS 850807 43 40 0 53)
(0 0 151 15 146 T CLE 710 BOS 850808 17 22 1 409) (0 0 7 3 10 V CLE 710 BOS 850808 5 1 1 409)
(0 0 91 73 146 T CLE 955 BOS 850808 76 84 1 87) (0 0 1 24 25 V CLE 955 BOS 850808 25 18 0 87)
(1 0 104 62 146 T CLE 1405 BOS 850808 56 56 1 411) (1 0 0 20 20 V CLE 1405 BOS 850808 19 14 0 411)
(7 4 44 123 144 T CLE 1730 BOS 850808 134 133 1 53) (7 4 -3 43 40 Q CLE 1730 BOS 850808 43 33 0 53)
(0 0 132 34 146 T CLE 710 BOS 850809 28 34 1 409) (0 0 -1 16 15 V CLE 710 BOS 850809 11 7 1 409)
(0 0 90 74 146 T CLE 955 BOS 850809 76 115 1 87) (0 0 2 23 25 V CLE 955 BOS 850809 24 19 0 87)
(3 0 75 91 146 T CLE 1405 BOS 850809 81 82 1 411) (3 0 -1 21 20 V CLE 1405 BOS 850809 23 17 0 411)
(13 3 23 145 144 T CLE 1730 BOS 850809 138 171 1 53) (13 3 -18 58 40 Q CLE 1730 BOS 850809 48 33 0 53)
(1 1 105 61 146 T CLE 710 BOS 850810 48 37 1 409) (0 0 2 23 25 V CLE 710 BOS 850810 19 14 0 409)
    
```

図 7 FLIGHT.DATA.CACHED ユニットの CACHE スロットの中味
Fig. 7 Contents of CACHE slot in unit FLIGHT.DATA.CACHED

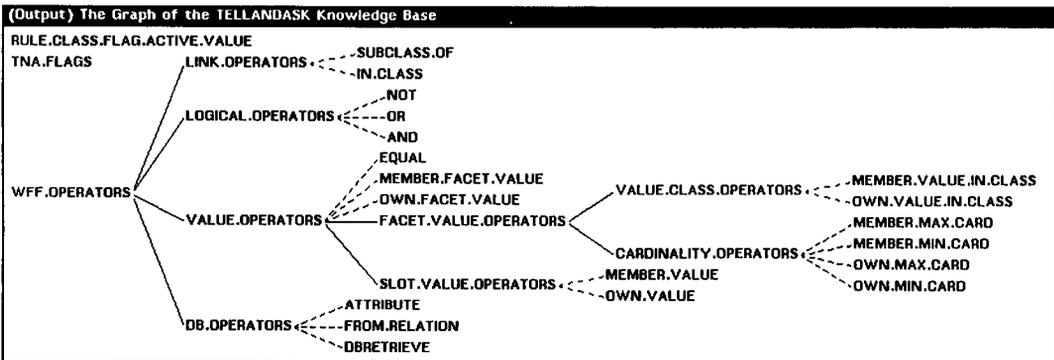


図 8 拡張された TellAndAsk 知識ベース
Fig. 8 Augmented TellAndAsk knowledge base

RELATIONS ユニットのメンバ・ユニットで, MAPPER . RELATION . ATTRIBUTES のサブクラス・ユニットである必要がある。

2) (ATTRIBUTE <ATTR . NAME> OF ?X <RELOP> <VALUE>)

ここで <ATTR . NAME> は <DD . UNIT . OR . REF> で定義されているメンバ・ユニット, すなわち属性名を意味する。<RELOP> は関係演算子であり, 次のうちの一つである。

```
IS
EQUALS
EQUAL
=
> (大なり)
>= (大なりイコール)
< (小なり)
<= (小なりイコール)
<> (等しくない)
```

オペレータのうち IS, EQUALS, EQUAL, = はすべて, 同じ意味である。<VALUE> は, 定数または変数であり, 変数の場合には, この S 式を評価する時に値が決まっている必要がある。したがって, この S 式は属性名の値の制約を表現できる。

```
(ATTRIBUTE SELL . COST OF ?X > 100)
```

この S 式は, 「SELL . COST が 100 を越えるレコード」という意味になる。これらの S 式は値を返すので, 次のような書き方もできる。

```
(ATTRIBUTE SELL . COST OF ?X < (ATTRIBUTE
PRODUCE . COST OF ?X))
```

3) (<RELOP> <VALUE> <VALUE>)

<RELOP> は上で示したとおり, 関係演算子である。各 <VALUE> は定数または, (ATTRIBUTE . . .) である。<VALUE> は変数でも良いが, この S 式が評価される時には値が決まっている必要がある。

DBRETRIEVE は, TellAndAsk の問い合わせの文脈で用いることができる。次の例がそうである。

```
(IF (AND (A HOLIDAY OF SPECIAL . DATES IS ?H)
(EQUALS ?DD . UNIT (A CURRENT . DD . UNIT OF
CURRENT . STUDY)))
(DBRETRIEVE
(?TUPLE IS FROM RELATION ?DD . UNIT)
(ATTRIBUTE SHIP . DATE OF ?TUPLE IS ?H)
(ATTRIBUTE COST . CODE OF ?TUPLE IS ?CC)))
THEN (A SPECIAL . CUSTOMER OF CURRENT . STUDY IS ?CC))
```

3.4 クエリ・ユニット: オブジェクト指向インタフェース

前述した TellAndAsk インタフェースは, 使用者に MAPPER アクセスに関してオ

プロジェクト指向の使用法を提供している。ここで、MAPPER のアクセスに必要なユニットについて詳しく述べる。

BATCOMM 用には、MAPPER . QUERIES . WITH . xxx . ICON ユニットのメンバ・ユニットが、INTCOMM 用には、MAPPER . QUERIES . VIA . WINDOW ユニットのメンバ・ユニットが必要であることは述べた。このメンバ・ユニットをクエリ・ユニットという。クエリ・ユニットには、いくつかのスロットが継承されてくる。

1) RELATION . VARIABLES

このスロットには、検索すべき MAPPER の RID のための情報が入る。形式は次の通りである。

((<TUPLE . VARIABLE . NAME> <DD . UNIT . OR . REF>))

この式は、TellAndAsk では次のものと同等である。

(?<TUPLE . VARIABLE . NAME> IS FROM RELATION <DD . UNIT . OR . REF>)

SEATS では次の値を用いた。

((X FLIGHT . DATA))

こうすることで、MAPPER のレポートを検索するのに必要な情報を持つユニットの指定ができる。上記の FLIGHT . DATA は、図 5 中の FLIGHT . DATA ユニットの意味する。

2) RETRIEVE . ATTRIBUTES

検索してきたレコードの中の抽出すべき属性値を指定する。形式は次の通り。

(<ATTRIBUTE . NAME><TUPLE . VARIABLE . NAME>)

SEATS では次の値を用いた。

((THR . W-L) (LCL . W-L) (AVBL . SEAT X) (LEG . BKG X)
(ACT . CAP X) (CL X)
(DES . CTY X) (DEPT . TIME X)
(ORG . CTY X) (DEPART . DATE X) (HST . BKG X)
(HST . DEP X)
(C . S X) (FLIGHT . NMBR X))

3) PREDICATE

MAPPER レコードの満たすべき条件を指定する。形式は次の通りである。

(<PRED-OP><SUB-PRED>…<SUB-PRED>)

<PRED-OP>は、>, >=, <, <=, <>, =, IS, EQUAL, EQUALS, AND, OR, NOT である。

<SUB-PRED>は定数または、他の PREDICATE か次のものである。

(<ATTRIBUTE . NAME><TUPLE . VARIABLE . NAME>)

SEATS では次の値を用いた。

(AND (> (DEPART . DATE X) 850800)
(= (FIRSTCHAR X) " "))

この式は「DEPART . DATE (出発日) が 1985 年 8 月以降で、レコードの先頭がスペースのもの」という意味になる。MAPPER のレポートは、先頭がスペース

のものがレコードである。

検索の制御に、次の二つのスロットがあり、メソッドが記述してある。

1) COMMUNICATE . QUERY

このメソッドは、上に示した三つのスロットの情報を分析して MAPPER のランを組み立てて PC に渡す。検索されたレポートは、CACHE . FLAG が ON のとき CACHE スロットに入れられる。

2) RETRIEVE . RECORD

このメソッドは、検索されたレポート中の次のレコードを検索してきて、RETRIEVE . ATTRIBUTES スロットに示されている形式の値を返す。

RETRIEVE . ATTRIBUTES スロットの値が、

((COST . CODE X) (SHIP . DATE X))

であればメソッドの返す値は、

(ARCO 850805)

という形式をとる。これは、COST . CODE が ARCO で SHIP . DATE が 850805 のレコードである。SEATS では、たとえば次のようなレコードが返る。

(0 0 138 28 146 T CLE 710 BOS 850801 34 22 1 409)

このレコードは、出発日時が 1985 年 8 月 1 日 7 時 10 分の 409 便に関するものである。

このように、クエリ・ユニットを用意し、このユニットに対しメッセージ“COMMUNICATE . QUERY”や“RETRIEVE . RECORD”を送ることで、MAPPER を検索できる。

4. お わ り に

本稿のねらいは、人工知能と MAPPER との係わり合いについて述べるだけでなく、コンピュータ・システムの中におけるエキスパート・システムのあり方についても言及することにあった。

エキスパート・システムの特徴は、その知識の取り扱いにある。したがって、その特徴を生かすような分野と問題に適用すべきである。近年の、ブームとも言えるほどの人工知能への期待には多大なものがある。このブームのおかげで、従来のコンピュータ・システムでは解決できそうもないと考えられ、あきらめられていた多くの問題が発掘されつつあり、その意味での意義は大きい。

しかし、人工知能に対する期待のあまり、人工知能が何でもできる（これは、将来技術的にできる可能性があるが、現状ではまだ完成していないものも含め）という誤解も多く存在する。ここで、人工知能、とりわけ、エキスパート・システムに対する正しい認識を持ちたいものである。

論文執筆にあたり、貴重な助言を下された、熊倉、西、柳沢の各氏に感謝します。

- 参考文献 [1] R. Wilson, “航空運賃割引管理エキスパート・システム”, 技報, 日本ユニバック, No. 12, 1987, pp. 13-26.
- [2] R. E. Carhart, “Installing and Using the KEE/MAPPER INTERFACE (KEEMAP)”, IntelliCorp Report.
- [3] 大須賀 節雄編著, “知識ベース入門”, オーム社, 1986.
- [4] MS-DOS User's Guide, SPERRY Personal Computer Library, UP-11701 Rev. 2.
- [5] BASIC User's Guide, SPERRY Personal Computer Library, UP-11702 Rev. 2.
- [6] Terminal Emulation User Guide, SPERRY Personal Computer Library, UP-10144 Rev. 2.2.
- [7] KEE User's Manual Level 2.1, UP-11451 Rev.1.
- [8] KEE Reference Manual Level 2.1, UP-11452 Rev. 1.
- [9] KEE RULESYSTEM2 Reference Manual, UP-11454 Rev. 1.
- [10] KEE Active Images User Manual, UP-11455 Rev. 1.
- [11] MAPPER1100 解説書 会話機能編, 日本ユニバック, 481205450-4.
- [12] MAPPER1100 解説書 フォーム/ラン生成機能編, 日本ユニバック, 481205451-5.

執筆者紹介 工藤 隆 司 (Ryuji Kudo)

昭和 29 年生. 55 年中央大学経済学部卒業, 同年日本ユニバック(株)入社. 社内システム教育インストラクタを経た後, 59 年通産省工業技術院電子技術総合研究所へ国内留学, 61 年 Sperry にて人工知能のトレーニングを受ける. 現在, 知識システム開発部に所属し, エクスパート・システムの開発に従事. 情報処理学会, 日本ソフトウェア科学会会員



論文

MAPPER と光ディスク ——花王(株)における事例

MAPPER Coupled with Optical Disk System ——The Case of KAO Corp

岩 佐 一 夫

要 約 光ディスク・ファイル・システムは、手書き文書、図形、スケッチ等の生の情報を大量に蓄積するには有効であり、数多く導入されている。しかし、スタンド・アローン型のシステムでは、蓄積された情報の検索を考えると使い勝手に制約が多い。

そこで、光ディスク・ファイル・システムをホスト・コンピュータと接続するとともに、ホスト側の MAPPER 検索システムを光ディスク・ファイル中の文書イメージの2次情報の検索に用い、効率のよい検索システムを開発した。本稿はその事例報告である。

Abstract An optical disk file system is very effective for a storage to store a large volume of raw data such as handwritten letters, graphics, and sketches.

Its full power is not drawn, however, without a host computer.

The paper reports the document image retrieval system which combines an optical disk file and the secondary information retrieval system using MAPPER1100 on a host computer.

This case illustrates one of synergistic effects of MAPPER and optical disk combination.

1. はじめに

レーザ光を用いた記録媒体としての光ディスク・ファイル・システムは、大容量情報を生のままで蓄積するには有効であり、スタンド・アローン型が数多く導入されている。スタンド・アローン型が適しているのは、蓄積量が多くなく検索方法が明解かつ、特定部門内で利用する場合である。しかし検索条件が複雑であり、蓄積文書の利用部門が広範囲な場合には、スタンド・アローン型の利用方法では使い勝手に制約が多く行きづまってしまう。

花王(株)では、光ディスク・ファイル・システムとホスト・コンピュータとを接続することにより、今までの EDP システムで行ってきた検索システムと、生の文書情報の検索を融合化させた。つまり、ホスト・コンピュータ上の MAPPER および文献検索システム CIR-J を利用することにより、光ディスク・ファイル・システムの2次情報検索および利用部門に対する情報提供指示を、スタンド・アローン型の光ディスク・ファイルによる検索システムの一機能として付加した。

本稿では、このシステムを事例として紹介する。

2. 光ディスク・システムの概要

花王(株)では、EDP 化が強力に推進されており、ほとんどの業務にコンピュータが利用されている。情報検索システムでは、たとえば販売実績のデータのように数値そ

のものをコード情報に変換蓄積し、その内容に解析を加えて利用者に提供するものが大半を占めている。これらの業務は主として、大量データを一括保存し取り出すことにより、加工時間の短縮と正確さを狙ったものであり、取り扱えるデータはコード化されたものに限定されているというのが現実である。

ところが、実際の企業活動を支援するために必要な情報としては、手書き文書、図形、スケッチ等コード化しにくい生の形が多々ある。これらの情報を従来の EDP システムで取り扱おうとすると、次の理由で困難である。

- 1) 既存の入出力装置では、イメージが扱えない。一部出力装置にはイメージ処理機能を有するものもあるが、入力装置が必要な場所にもれなく配置されていない。
- 2) A4 版書類つまり、一枚 4 M ビット相当のデータ量を圧縮して保存したとしても (G4 規格で圧縮した場合で、435 K ビット)、通常の媒体 (磁気ディスクやテープ) を当てるにはコストがかかり過ぎる。
- 3) コンピュータ・ネットワークは、コード情報の伝送を主目的に設計されており、通信一回当たりの伝送量は、20~30 倍に達し、利用頻度も従来の見積り基準では推測できない。

そこで、これらの問題に対処するために、イメージの取り扱いができる機器として、光ディスク・システムの採用を決定した。具体的な理由は以下の通りである。

- 1) 蓄積媒体として安価であり、長期保存に耐えられる。
- 2) 入出力装置が標準機構として完備されている。
- 3) 入力方法が簡単である。

しかし、光ディスクは、高密度記録かつ長期保存に対して強力なメディアであるが、情報を検索するには問題があった。

第 1 に、検索機能が利用者の要求を満足していないことである。大量に蓄積された情報群から、欲しい情報を選択するためにキーを指定することになるが、一文書当たりの検索キー数の最大条件が現実の要求を下回っている。また、曖昧な条件でも検索できるキーのマスク機能がないため、不都合な場合があった。

第 2 に、検索指示をする機構がシステム 1 台に 1 個しかなく、光ディスク・システムが設置されている近辺の人しか情報を検索できないという難点があった。この難点をカバーするには利用者のいる場所ごとに装置を配さねばならず、費用が膨大になってしまう。かりに費用を度外視して考えたとしても、すべての装置上に同じ情報を複写しなければならず利用目的には沿えないことになる。さらに、コンピュータを使った業務処理と合わせて考えると、光ディスク・システムと TSS 端末は分離されており、データベース上の情報と光ディスク情報を融合させようとするのは困難であった。

3. オンライン・光ディスク・ファイル・システム

光ディスクに手書き情報を保存しておけば、紙のまま保存する方法に比べ便利な点が多い。このため、さらに便利な仕組みが当然のごとく要求され始めた。新たな適用領域は、営業支援システムであり要望は以下の通りである。

- 1) 光文書情報の検索指示はすべて TSS 端末から行えること。

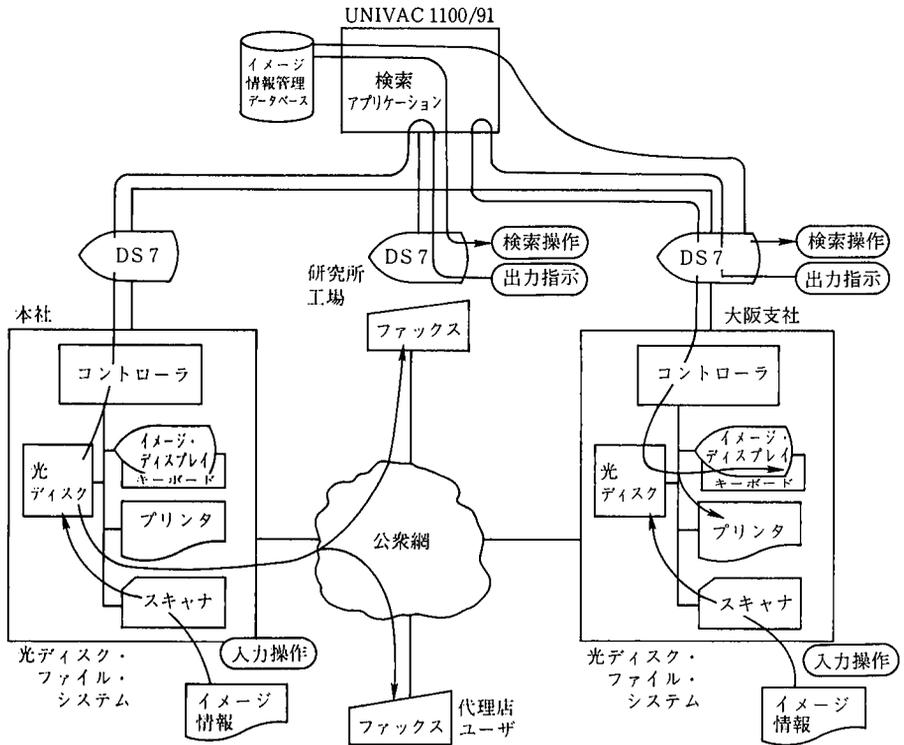


図1 オンライン・光ディスク・システム概念図

Fig.1 Online optical disk system

2) 遠隔地の TSS 端末利用者にも検索結果の出力ができること。

以上の要求に対して、次の方法を採用した。

1) 光ディスク・ファイル・システムを DS 7 と接続し、DS 7 上にゲートウェイを構築する。

2) 遠隔地の利用者に対しては、既設のファックスに出力を行う。

今回のシステムでは、文書情報の発生場所が東京本社、大阪支社に集中しており、東西に分散設置された光ディスク・ファイル・システムにそれぞれ異なる情報を蓄積し、ホスト・コンピュータ上の検索アプリケーションにより一元化された情報として利用者に提供するものである。これら検索アプリケーションは、MAPPER と CIR-J を使用し構築しており、光ディスク・ファイル・システムには、PANAFILE を使用している。オンライン・システムの概要を図 1 に示す。

4. UNIFILE システム

システムの特徴は、コントロール端末(DS 7)がゲートウェイの役割を果たし、TSS 端末としての機能と、光ディスク・ファイル・システムの制御機能を持ったものである。PANAFILE とコントロール端末を総称して UNIFILE システムと呼んでいる。構成は図 2 の通りである。

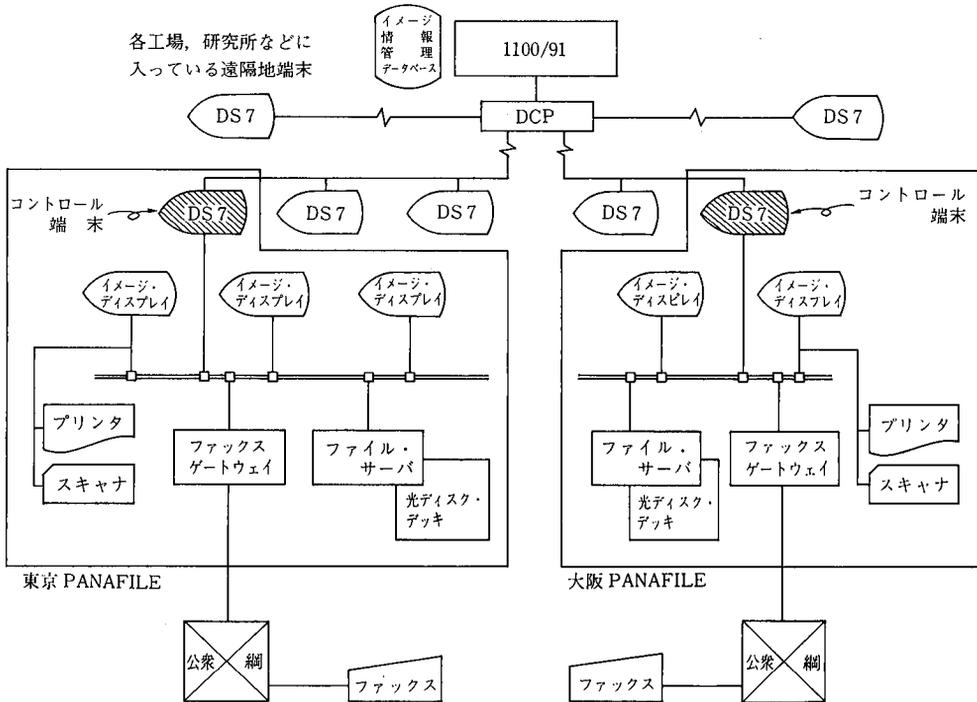


図2 UNIFILE システム構成図

Fig.2 UNIFILE system

4.1 UNIFILE システムの概要

PANAFILE の制御は、MAPPER のユーザ・アプリケーションで行っており、光サービス・ランと呼ぶ。光サービス・ランは、コントロール端末から常駐ランとして稼働している。主要機能は、以下の通りである。

- 1) 情報の登録……スキャナからのイメージ情報の読み込みは、MAPPER アプリケーションにて指示され、コントロール端末を経由し、光ディスクに登録される。登録された結果の光アドレス（光ディスク No.、文書 No.、総ページ数）は、PANAFILE よりコントロール端末に返され、MAPPER アプリケーションに渡される。光アドレスは、利用者が入力した 2 次検索情報と共に MAPPER データベース、CIR-J データベースに自動的に登録される。
- 2) 情報の検索……利用者は、TSS 端末から検索アプリケーション（MAPPER、CIR-J）を実行し、検索したい手書き情報の 2 次検索を行う。検索された情報は、利用者の出力要求指示によりイメージ・ディスプレイ、レーザ・プリンタ、ファックスへ出力される。
- 3) ファクシミリ送信……遠隔地からの利用者は、TSS 端末から必要情報を検索し、利用者の近くのファクシミリに出力指示を行うことができる。また、代理店などに対して、ファクシミリを利用した同報的な情報提供も可能である。

以上のような PANAFILE に対する指示は、MAPPER アプリケーションより下記方法にてコマンドを送って行う。

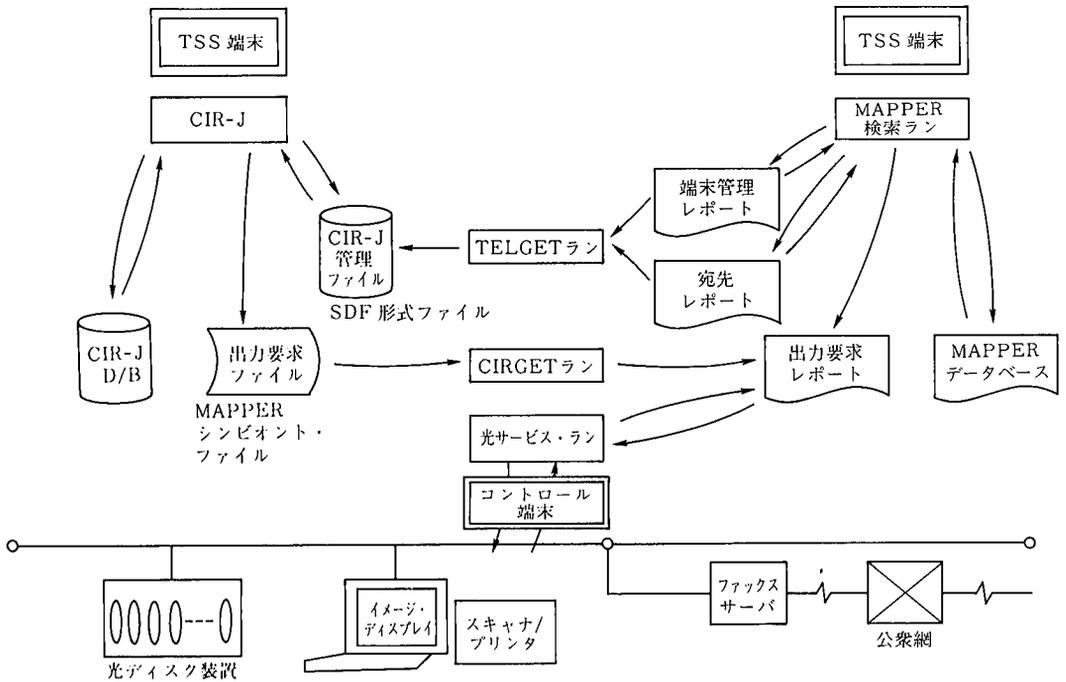


図3 UNIFILE システムの処理の流れ
Fig.3 Control/data flow in UNIFILE

```

    :
    @BRK
    /#/#/# PANAFILE コマンド } コマンド送信
    @BRK
    @OUT
    @CHG INPUT$ Vn          } ステータス受信
    :
  
```

ここで、/#/#/#付きの出力伝文は、コントロール端末上の PANAFILE コントロール・プログラムにて、PANAFILE に対するコマンド指示と判断されている。

4.2 UNIFILE システムの処理

ここでは、UNIFILE システムでの処理について、ユーザ・データベースと光サービス・ランを中心に述べる。

図3は、UNIFILE システムの処理の流れを示す。

検索時のセキュリティ管理、ファックス出力時の宛先管理、光ディスク・ファイル・システムに対する出力指示は、MAPPER レポート上にて集中管理されており、それぞれ端末管理レポート、宛先レポート、出力要求レポートに該当する。ここで、TELGET ランと CIRJET ランは、次の役割を持つ。TELGET ランは、端末管理、宛先レポートを、CIR-J 管理ファイルに変換するものであり、CIRJET ランは、CIR-J からの出力要求指示を MAPPER 出力要求レポートに登録するための、バッチ・ポート・ランである。

4.2.1 ユーザ・データベース

光ディスクに蓄積されている情報には、自社製品情報、他社製品情報、営業コンタクト・レポート等がある。これらの情報は、ホスト・コンピュータ側のデータベースに2次検索情報として管理されており、TSS 端末からの検索対象となる。

データベース中の1レコードは、光ディスクの1情報に対応しており、以下の項目からなる。

- 1) 2次検索情報としての検索用キーワード（カナ製品名等）
- 2) 手書き文書が蓄積されている場所（東西区分）
- 3) 蓄積されている文書の物理的光ディスク・アドレス（光ディスク No., 文書 No., 総ページ数）
- 4) その他属性（漢字名称, 目的等）

4.2.2 データベースの検索

データベースの検索方法は、蓄積されている情報により異なる。製品情報のように、カナ製品名称の固定項目による定型的な検索（マスク指定可）では、MAPPER による検索アプリケーションを作成している。また、すべての項目（キーワード）を対象とした非定型項目による複合検索では、CIR-J を利用している。

具体的な検索方法と出力指示方法は、添付資料の通りである。

4.2.3 光サービス・ライン

利用者からの PANAFILE への出力指示は、TSS 端末により MAPPER, CIR-J とともに出力要求レポートへ登録される。また、文書情報の登録指示についても同様である。コントロール端末から稼働されている光サービス・ランは、出力要求レポート中の出力指示を読み取り、コマンド/ステータスの受授を行っている。

以下にその概要を説明する。

- 1) 光サービス・ランは MAPPER ランであり、コントロール端末から常駐ランとして稼働している。
- 2) 出力要求レポートに未処理の要求指示が有るか否かを、0.5 秒間隔で監視する。
- 3) 未処理の要求指示は、各要求区分（ファックス, 表示, 印書, 登録）に合った PANAFILE へのコマンドに変換し送信する。
- 4) PANAFILE からのレスポンスを受信し、処理ステータス終了時刻等を出力要求レポートへ書き込む。

5. 効 果

今までのコード情報だけの EDP システムに、イメージ情報の検索を融合させることにより、花王(株)では以下の効果が得られた。

- 1) 報告書の類が、再活用し参照されることを前提に記述されるようになり、結果として統一され要旨とキーワードが明確になった。
- 2) 全社レベルでの文書ファイリング・システムへの展開が望まれ始めた。
- 3) 特定の代理店や顧客、または営業マンの訪問先でも要求される情報を直接ファックスに出力し、正確なデータを迅速、かつ必要な部分だけを提供でき、サービス向上に寄与した。

6. おわりに

イメージ情報を EDP システムで扱おうとするアプローチは、今後一層強い要求として発生すると思われる。なかでも、イメージ情報を直接ホスト側で取り扱えるようにするのが最近の課題となっている。

今回のシステムは、光ディスク・ファイルに蓄積された情報のキーワード検索と選択されたイメージ情報の出力指示を、MAPPER あるいは CIR-J を核に構築したものであり、イメージ情報の操作は光ディスク・ファイル・システムに依存しており、ホスト側の機能は使用されていない。

しかし最近、既設 TSS 端末へのイメージ情報の表示等のように、イメージ情報をホスト・コンピュータ側に取り込む（コンピュータ・ネットワークにイメージ情報を流す）課題も現実の要求として出てきている。これらの要求に対しては MAPPER は有効である。MAPPER の機能には、DS 7 上のイメージ処理プログラム (U-IMAGE) との連動、また、ファクシミリとのイメージ情報の授受 (FSP1100 経由) があり、イメージ情報システムの構築用ツールとしても使用できる。しかし、イメージ処理に固有な大量ビットの操作のためのホストのオーバーヘッドを考えると、適用に当たっては十分な検討が必要であろう。

今後は、この分野に対する技術動向の方向性を考慮しつつ、イメージ情報の効果的な利用形態を提案していきたいと考えている。

本システムの開発に当たって、ご指導いただいた花王(株)化学品管理部の半田進部長、およびシステム開発部の方々に篤くお礼申し上げます。

添付資料 検索画面紹介

(1) 商品情報検索

商品名の選択。付図 1 の検索画面に「ルナ」と入力すると、登録されている商品名の中から、ルナの名のつくすべての商品が表示される (付図 2)。「ル」だけを入力しても、表示される品目数は多くなるものの、検索を行うことができる。

検索したい商品名 (ルナック TD-5) をカーソルで指定する。

それから調べたい項目 (特性値) を指定する (付図 3)。すると、検索された原紙情報が、イメージ・ディスプレイに表示される (付図省略)。

なお、各商品情報は 4 頁目までは統一様式で構成され、5 頁以降は社外秘の関連情報が続く。

花 王 製 品 情 報 検 索

商 品 名

ルナ

付図1

花 王 製 品 情 報 検 索

* ルナック T D - 5

* ルナック T S H - 2

* ルナック T B A

* ルナック S - 3 5

* ルナック S - 3 0 5

* ルナック P - 9 5

* ルナック S - 2 0

* ルナック S - 4 0 0

* ルナック S - 5 0 0

* ルナック S - 9 0 0

* ルナック S - 9 8

* ルナック T D - 2

* ルナック M y - 9 8

* ルナック L - 9 8

* ルナック L - 7 0

* ルナック L - 8 5

次 面 面 前 面 面 次 処 理

付図2

花 王 製 品 情 報 検 索

商 品 名

ルナック T D - 5

終 了

特 性 値

法 関 連

安 全 衛 生

生 産 秘

社 外 秘

次 の 商 品 へ

付図3

(2) ファックス送信指示

プリント出力，ファックス出力の指定を行う（付図4）。

前画面でファックス送信が選択されると，付図5の画面が表示される。

次に「カオウ」と入力して，ファックス送信すべき宛先のマスク検索を行う。

すると，「花王」と名のつくすべての宛先が画面に表示されるので，送信したい宛先にカーソルを位置付ける（付図6）。

送信指示を行う（付図7）。

光ディスクNO	00002
文書NO	00224
ページNO	00001

処理を選択して下さい。

- プリンター出力
- FAX送信
- 次のページへ
- 次の商品へ
- 次の検索へ

付図4

花王製品情報 FAX送信処理	
送信相手先名	
カオウ	
<input type="checkbox"/> 終了	

付図5



付図6



付図7

- 参考文献 [1] “パソコン端末から光ファイルのイメージ情報を検索できる三菱重工の文書イメージ統合システム”, 日経コミュニケーション, 1985, 11-18.
- [2] “普及型と多機能型, 2極分化進む光ディスク・ファイリング・システム”, 日経コミュニケーション, 1986, 5-26.
- [3] NTT 業務管理部電報電話課編, ファクシミリ通信網, フェア(株).

執筆者紹介 岩佐一夫 (Kazuo Iwasa)

昭和23年生, 46年芝浦工業大学電気通信科卒, 46年日本ユニパック(株)入社, 大型計算機の利用技術に関する客先サービスに従事。



Harry Katzan, Jr.

“Invitation to MAPPER (I)”
—A Pragmatic Approach to
End-User Computing and Report
Preparation—

Petrocelli Books, Inc., A5 判, xi+ 143 pp.,
1983.

本書は、“MAPPER システムの紹介”として書かれたものであり、主として“エンド・ユーザ・システム”の設計や業務設計に携わる人、および“エンド・ユーザ・システム”とか“非手続き型言語”について、その背景となる知識を必要としている人を対象にして、まとめられた著書である。

著者自身の MAPPER システムでのシステム構築経験を踏まえ、エンド・ユーザ・システム開発ツールとして MAPPER を位置づけ、MAPPER システム利用のメリットおよび基本機能について解説している。

本書の構成は 8 章に分かれており、第 1 章では“エンド・ユーザ向けシステム”の必要性と備えるべき条件について、第 2 章から第 7 章までは、MAPPER の基本機能を中心に解説を加えている。随所に例題が挿入されており読みやすい。

第 8 章では、エンド・ユーザ・システムの導入により、組織的にみた情報処理システムがどう変わるか、どうあったら良いかを論じている。

本書を読むに当たって、エンド・ユーザ向けシステムとはどのような要件を備えたものか、MAPPER システムはどんな基本機能を持っているのか等を理解するためには、第 1 章から読まれるようお勧めするが、手取り早く MAPPER の機能を理解されたいのであれば、第 2 章から第 7 章を読まれることをお勧めする。

以下に本書の内容の一部を紹介してみよう。

著者は、**エンド・ユーザ自身による情報処理の必要性**について、次のように述べている。

『今日情報処理の世界では、適用業務の機械化のバック・ログが山積みし、その解決の見通しが立たないことが最大の課題であり、このため“エンド・ユーザ・システム”と総称されているさまざまなソフトウェアやシステム言語が開発されて

きている。MAPPER システムは、この範疇に属するシステムとして非常に優れたシステムであると言える。エンド・ユーザ・システムにとって最も肝要なことは、情報が迅速に検索できること、しかも情報の利用が広範囲で、かつ欲しい形で、いつでも利用できることである。』

また、“**エンド・ユーザ向け**”という概念と、それのもたらすメリットについては、次のように述べている。

『“**エンド・ユーザ向け**”の概念は、シンプルな言い方をすれば、コンピュータの専門家でない人々が伝統的な手法とは別のやり方でコンピュータを利用することを意味する。

少々詳しく述べると、“**エンド・ユーザ向け**”とは以下の三つの条件を満たすものと言える。

- 1) エンド・ユーザのためのシステムと言語があること。
- 2) エンド・ユーザ自身によるアプリケーション開発が容易にできること。
- 3) エンド・ユーザ自身でのコンピュータの自由な利用ができること。

これらの条件が満足されたとき、エンド・ユーザはコンピュータの能力を自分自身の手で駆使して情報のタイムリな活用や、仕事の生産性の大幅な向上が期待でき、さらにコンピュータの専門家の手をわずらわせることなく、自分自身の手で情報処理が行えることになる。これによる最大のメリットは、経済性の向上であり、企業にとって確実に利益を産み出してくれるであろう。』

さらに“**エンド・ユーザ・システム**” (MAPPER システムをベースとして) について、次のように述べている。

『**エンド・ユーザ・システム**は単なる会話型処理システムでも、リアルタイム処理システムでもなく、コンピュータのハード、ソフトおよびデータ通信設備をインテグレーションした新しい機能である。その利用にあたっては、たとえば従前からあるバッチシステム、オンライン・システム、あるいは外部データから情報が抽出され、加工されて、業務または目的に応じたデータベースとして蓄えられる。ひとたびデータが**エンド・ユーザ**にとって準備されれば、それは、**エンド・ユーザ**

自身のものとして自由に使えるようになり、業務処理のプログラムはエンド・ユーザ自身の手で準備され、必要に応じてその処理の実行もエンド・ユーザ自身で行われることになる。個別業務や、特定部門のための業務処理プログラムを集め、業務境界や組織範囲を越えた全社規模の処理結果を得ることも可能である。

これらの処理は、“エンド・ユーザ・システム”によってのみ実現できるものと言える。]

本書は、米国での企業への MAPPER 導入の経験をもとに、著者のとらえたエンド・ユーザ・システム (MAPPER) の実践的なあり方について述べており、今後のエンド・ユーザ・システム導入についての一助となると思われる。一読されることをお勧めしたい。

(ネットワーク本部営業開発部 中根紀夫)

山中義昭監修

プログラマーなしの
アプリケーション開発

アシスト出版局, B5判, xiv+419 pp.,
1984年, 4,800円。

本書は、情報システムの第一人者 James Martin による、第4世代言語がらみの著書である。

この第4世代言語 (4GL) という言葉は、とくに明確な定義はないが、一般的には、エンド・ユーザ向け高水準言語や対話型処理システム、データベース・アクセス言語、意思決定支援システムなどを指し、“従来型言語の問題点を解決するための機能を持った言語”と考えることができる。近年においてはその普及は著しく、また簡易言語的な使用から、システム構築のメイン言語へと役割が変わりつつあるようである。

本書は21章からなり、各章ごとの説明は繁雑となるのでここでは割愛させていただくが、全体的話の筋としては、『ソフトウェア・クライシス (例の、このペースでソフトを開発していかなければならないとすると、××年後には全世界の総労働力がコンピュータ・プログラミング作業に当てられなければならない、というやつ) から始まって、じゃあどうすればいいのか、という話になり、そのためには、「エンド・ユーザ自身が第4世代言語を使ってアプリケーション開発や、システムのメンテナンスをする」という方法、つまり“プログ

ラマーなしのプログラム開発”がありますよ』という流れになっている。

さらに、具体的な第4世代言語の説明が、その機能から使用方法まで実例を挙げて (この部分は大変詳しく述べてあり、ほぼマニュアルとして使えるくらいである。それゆえ非常に退屈でもある) 述べてある。とくに、MAPPER に関しては、ほぼ1章を費してその機能や特徴、使用例、実績まで詳しく説明してあり、かつ、「MAPPER はエンド・ユーザの間に独自の空気をもたらした」と評価されている。

確かに、著者の述べているように (当然、アメリカの状況ではあるが、日本でもほぼ同様であろう) システム開発の現状は、ユーザのシステム化ニーズが増大の一途をたどっており、ソフトウェア開発能力が需要に追いつかないという深刻な状況を呈するに至っている。誰もが、この状況を打開するためには第3世代言語に取って代わる“新しい言語”が必要であることに気づいており、この“ニュータイプ”の出現を待ち望んでいる。それこそ“第3世代言語に代わるもの”として“第4世代言語”が期待されている。

そこで、著者によると、このようなユーザのシステム化ニーズの増大への主な対応策としては、“ソフトウェア開発の生産性向上”と“エンド・ユーザ・コンピューティング”があると述べている。

前者は限られた人員の DP 部門で効率よくソフトウェアを構築することによってバック・ログを減らすことを目的とする。これはなにも第4世代言語とは限らないのだが、従来の言語による改善 (たとえば構造化や対話方式の採用) では25パーセントが限界である、とも言っている。劇的な生産性向上を図るのであれば、“ニュータイプ”の高水準言語を採用するか、プログラム開発に代わりプログラム購入を行うしかないそうである。

また、後者はシステム化ニーズをエンド・ユーザ自らが消化することによって、DP 部門の負荷そのものを軽減するという試みである。本書の表題にもなっている「プログラマーなしのアプリケーション開発」というやつである。第4世代言語は、この試みに対する言語面での解決策になりうると著者は主張している。ただし、このようにアプリケーション開発をエンド・ユーザ自身が行う場合、あるいは DP 専門家の手を借りる場合のいずれにおいても、この作業は適正な管理体制の下

で行わなければならない、とも言っている。

著者はこの管理体制の具体的手段をインフォメーション・センターと表現している。著者の言うインフォメーション・センターとは、直接、迅速にエンド・ユーザを支援するために設置されたDP部門のグループのことであり、エンド・ユーザに対するコンサルタントとして位置づけられる。

このように、本書は現在DP部門やシステム開発が抱える問題を大命題にし、それを第4世代言語という手段で解こうとしている。あたかも、第4世代言語が唯一無二の解決手段でもあるかのように感じられることも多かった。これは、原著の書かれた時期が数年前であり、第4世代言語のはしりのころであって、著者の第4世代言語に対する過大な期待ともいえるものがあつたのであろうか。

しかし、現実には、国内で今までに100本以上販売された第4世代言語はいくつかあるが、企業の基幹業務は相変わらずCOBOLやPL/Iで構築されたものが大部分である。なぜか、第4世代言語はまだ補助的な役割が多く、主力言語にはなっていないのである。

これは本書によると、ユーザが第4世代言語を導入する際、最も大きな障害となるのは膨大な量の既存のプログラムとプログラマーの抵抗である、といっている。この著者の懸念が当たつたためであろうか。確かに、どこのコンピュータ部門でもCOBOL、PL/I、アセンブラなどで組まれたプログラムが蓄積され、毎日の業務に欠かすことのできないものとなっている。これらのプログラムの保守および関連プログラム開発の作業は、コンピュータ部門の業務の中で非常に大きな割合を占めている。既存プログラムに関連した作業が多い以上、既存のプログラム言語を捨てるわけにはいかないであろう。

また、実際にプログラミング言語を使うプログラマーやシステム・エンジニアが既存言語を捨てきれないため、第4世代言語導入が見送られるケースも多いようである。既存言語での経験を積んだ熟練プログラマーほど、この傾向は顕著だそうである。なるほど、ベテランであればあるほど既存言語に精通し、ノウハウも蓄積している。まったく体系の違った新しい言語になれば、そのノウハウはほとんど生かせなくなってしまうであろう。それどころか、苦勞して習得した知識に時代

遅れのレッテルを貼られるかもしれない。

さらにプログラマーの中には、そのプログラマーとしての特性ゆえ第4世代言語は「単純すぎて物足りない」という者さえいるであろう。まあどの分野でも、基本的にまったく新しい考えが受け入れられるには長い時間を必要とするものではあるが、

何事も、「革新」には不安や反対がつきものであり、リスクも伴うが、DP部門の抱える諸問題(膨大なメンテナンス量、増え続けるバックログ、アプリケーション開発の生産性を向上させるには…等々)の解決のための一手段としての第4世代言語導入は、なかなか有効な手段であろうと思える。この機会を逃したら、将来迎えるであろう危機的状況により、もっと大きな不安やダメージを受けることになるであろう。

本書の活用のしかたとしては、確かに、著者の述べるところのエンド・ユーザ言語の活用状況、種々発表されているそれらの言語(4GL)の紹介、さらには、インフォメーション・センターについての具体的な環境設定・体制・運営方法は、企業のDP部門担当者にとっては、すぐにも役立つであろう。システム部門の担当者が4GLの導入を考えたとき、自社に合った効果的な4GLの選択が重要なテーマとなりうるが、その選択のために本書を一読するのも大変参考になるであろう。が、それよりも、現状の開発方法やアプリケーションに対する考え方自体を変える一助として本書が役立つことのほうが著者の本意と思える。

最後に、著者の“システム開発の現状に対する考え方”を端的に表していると思えた本文中のDaniel McCrackenのたとえ話を紹介しておく。多くのプログラマーの自分達の未知の技術に対する態度を揶揄したもので、——古い車でニューヨークからカリフォルニアを1日で走破しようとした男は、エンジンの馬力を増し、トランスミッションを変え、思いつく限り速度を増すための改良をした。警察のスピード・レーダを探知する機械をも取り付けた。それらの試みは、すでにそれを始める以前に失敗に終わった。彼がまったく別の技術を使って、たとえばジェット機を使ったらという提案を聞くと、彼は「そんなものは信用できない！」と言った——。

[James Martin, “Application Development without Programmer”]

(汎用システム部 生井和代)

成田光彰訳

管理職のための情報戦略
—データ処理から情報資源管理へ—

日経マグローヒル社, A 5判, 338 pp.,
1986年, 4,100円.

本書は1984年に原題「An Information Systems Manifesto」として出版された著書の訳本で、原著は「リアルタイム・システム」や「プログラマーなしのアプリケーション開発」などで有名なJames Martinによって書かれたものである。

原題にもManifestoとある通り、トップ・マネジメント、ユーザなど、さまざまな立場の人に対しての提言書かつ、宣言書である。まえがきにも書かれている通り、現に進行している情報システムに関する変革を、多くの企業は無視するか、無視しないまでも、これが自らに与える衝撃を正しく認識していない。これらの変化を望ましい方向に導くための情報提供を意図して本書は書かれている。

全体は二部構成で、第I部は「現在」と題されており、今起こっている変革と、ツールや、技法の活用によるコンピュータ利用改善措置などについて解説している。

第II部は「将来」と題されており、今後、起きると予想されるコンピュータ技術の激しい変化について展望している。

ここで、一応本書の章立てについて見ておこう。

第I部 現 在

- 第1章 データ処理の危機
- 第2章 自動化の自動化
- 第3章 ユーザを運転席に
- 第4章 仕様書に関する諸問題
- 第5章 仕様言語
- 第6章 インフォメーション・エンジニアリング
- 第7章 インフォメーション・センター
- 第8章 理想的なDP開発環境
- 第9章 数学的に証明可能なシステム設計
- 第10章 コンピュータを援用したソフトウェア設計の影響
- 第11章 開発ライフ・サイクルの変化
- 第12章 戦略計画
- 第13章 変革を管理する

第II部 将 来

- 第14章 富を築くベンチャー企業家
- 第15章 第5世代の主役は日本か
- 第16章 ベンチャー企業家と政府
- 第17章 連鎖反応

付録 I DP生産性を向上させる方法

付録 II 提 言

- A. トップ・マネジメントに対する提言
- B. エンドユーザに対する提言
- C. DP部門管理者に対する提言
- D. システム・アナリストに対する提言
- E. プログラマに対する提言
- F. コンピュータ学者に対する提言
- G. ソフトウェア・ハウスに対する提言
- H. コンピュータ・メーカーに対する提言
- I. ベンチャー企業家に対する提言

このように各章のタイトルを一覧しただけで、著者の言わんとしていることが伝わってくる。

著者は、今までの伝統的なソフトウェア開発の方法での限界を指摘し、これからの開発基盤の整備のためのキー・テクノロジーとして以下のものを挙げている。すなわち、

- 1) 第4世代言語
- 2) インフォメーション・センター
- 3) 柔軟性に富むデータベース
- 4) DP機能の最大限の自動化の活用
- 5) インフォメーション・エンジニアリング
 - ・情報資源のトップダウン管理
 - ・徹底的なデータ・モデリング
 - ・戦略的なネットワーク計画

などである。

次に、著者の思考の一端を探るべく本書の一部を引用してみよう。

第1章 「データ処理の危機」の章から……今世紀の初頭に発明された蒸気機関は、非常に複雑な機械であった。技術者たちは、1パーセント未満でも効率を上げるために精巧な機構を発明した。やがて電気モーターが使われるようになったが、それでも、彼らは愛情を込めて蒸気機関を調整し技術検討を続けた。COBOLを使ったプログラム開発とアプリケーション・ジェネレータを使ったアプリケーション開発との差はもっと劇的である。今日、ソフトウェアを製作したり、アプリケーションを開発したりする新

しい強力な技法は多数存在している。しかし、世界中の何百万のプログラマーやアナリストは、ほとんどそのような話を聞いたことがない。驚いたことに、欧州では蒸気機関の技術者は、電気モーターが開発されてから50年たっても蒸気機関の改良を続けていたという。

第3章 「ユーザを運転席に」から……ヘンリー・フォードは大量生産ラインを作っただけではなく、T型フォードの操作を簡単にして、誰もが運転できるようにした。コンピュータ産業もマイクロエレクトロニクスを応用して、T型フォードに当たるようなコンピュータを大量生産するだろう。専門のプログラマーがいなくてもうまく動かせるようにできて初めて、コンピュータは大量に売れるだろう。現在、これが起こりつつある。コンピュータ業界における、プログラマー抜きのアプリケーション開発は、トランジスタの発明以来、最も重要な革命である。

第14章 「富を築くベンチャー企業家」から……創造的な組織は、すべて2種類の“がん”と戦わなければならない。それは非現実主義と官僚主義である。少しでも機会があれば、両者は自己の活動を繰り広げる。不幸にして、コンピュータとOAはともに、これらの活動の強力な増幅器として働く。

新しい言語は、かつての電気モーターと同様、体制的DP部門側からはまず出現しない。

このように非常にわかりやすい例を引きながら随所に、注目すべき指摘や提言を行っている。

もちろん、MAPPERについても、代表的第4世代言語として取り上げているのに加えて、「DP革命の事例研究」(第13章9項)の中ではサンタフェ鉄道のMAPPERによるシステム構築事例を「世界で最も輝かしいDP成功談の皮切り」として紹介している。本書によれば『このMAPPERによるシステム構築が、同社の生産性上昇率を13パーセントから28パーセントに上昇させ、貨物取扱量が2倍になったにもかかわらず管理スタッフの数は増えていない。また、ある証券アナリストの報告によると、このシステムは株価を1.5ドル引き上げる働きをした』とのことである。

本書では先に述べたように、さまざまな立場の人に対する示唆に富む提言がなされており、単にコンピュータの技術を取り上げているものではない。

したがって、トップ・マネジメントからコンピュータの知識がほとんどないエンド・ユーザ(利用部門)の人まで広い層に読んでほしい本である。また、情報システムに従事する人も従来の常識にこだわることなく、変革の進展を掴みとり、正しい方向へ読者自身とエンド・ユーザを導くためにも、指針として是非一読することをお薦めする。
(James Martin, “An Information Systems Manifesto”) (統合OAシステム部 金子 崇)

1. 代表的な MAPPER パッケージ

1.1 アプリケーション開発支援システム
MAPPERKIT 1100

MAPPERKIT 1100 は、利用者向けの標準開発モデルと、その考え方に基づいて構築されたソフトウェア KITBUILD と KITAID から構成されている。KITBUILD は、システム的设计・開発を、KITAID はシステムの使用・保守を支援する。作業はすべてメニュー方式で進められ、コンピュータの知識を持たない人でも容易に利用できる。MAPPERKIT によって開発されたシステムは、データベース、MAPPER ランはもとより、データ辞書、仕様書、使用解説書などもすべて MAPPER に内蔵される。

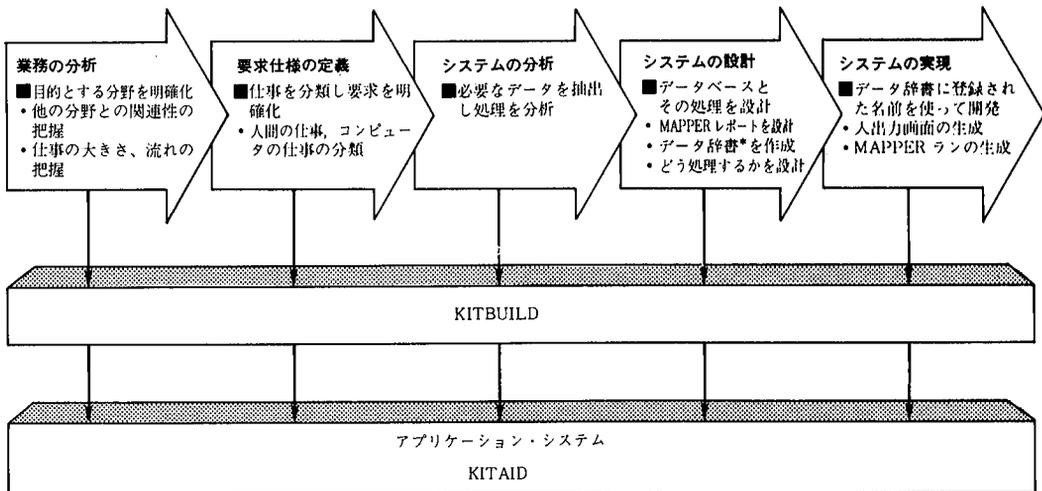
1) 特 長

- ① システム開発の生産性の向上のために、自動生成機能が用意されている。
- ② システム保守の生産性の向上のために、自動変更機能が用意されている。
- ③ 設計の各段階で必要なドキュメントが作成される。このドキュメントは、アプリケーションを使用する際の操作/使用解説書となる。
- ④ 開発方法や手順がわからなくなったとき、随時内蔵されているマニュアルを参照できる。

- ⑤ 利用者の実際の業務内容にそった作業メニューが作成できるので、開発されたシステムは一段と使いやすくなる。
- ⑥ システムの品質が向上する。
 - ・デバックが容易である。
 - ・開発手順、ドキュメントおよび MAPPER ランが標準化される。
 - ・入出力画面、ランの骨組みなどが自動生成されるため、バグの発生率が減少する。
 - ・保守のミスがなくなる。

2) 機 能

- ① KITBUILD……システム開発の各工程を支援し、標準化されたドキュメントをもとにアプリケーション・システムを構築する。
 - ・ドキュメントの作成・検索・表示・印書・削除・更新および管理
 - ・ダイアグラムの作成
 - ・ディレクトリの表示
 - ・データ辞書の作成および検索
 - ・レポート形式の作成
 - ・ガイド
 - ・入出力画面の作成
 - ・MAPPER ランの骨組み作成
 - ・MAPPER ランのクロス・リファレンス作成



※データ辞書は、システムに必要なすべてのデータ項目の名前(ID)、意味、長さ、タイプなどの属性が登録されたもの。

図1 MAPPERKITの標準開発モデル

- ・ソフトウェア部品の作成、引用
 - ・デバッグ支援
 - ・アプリケーション・システムのセーブ/ロード
 - ・セキュリティ
- ② KITAID……KITBUILD で開発されたアプリケーション・システムの使用・保守を支援する。
- ・使用者データベースの作成・検索・表示・印書・削除・更新および管理
 - ・メニューの作成
 - ・ガイド
 - ・データ項目の属性変更・追加・削除に伴うデータベース等の自動変更
 - ・汎用ユーティリティ

1.2 ダイレクト・マーケティング総合情報システム DM 1100

DM 1100 (Direct Marketing System for Series 1100) は、日々のフルフィルメントからマーケティング統計・分析、そして自動仕訳データの作成による財務会計へのデータ提供までの一連の管理業務を、効率的に処理する総合ダイレクト・マーケティング・システムのパッケージである。

このシステムを導入することにより、受注・出荷・配送・請求・入金といった一連の業務処理を迅速かつ効率的に行い、また、日々発注した取引をベースに顧客分析・商品分析等により、マーケティング、マーチャダイジングに対し有益な情報を得ることができ、効率良いダイレクト・マーケティング活動が遂行できる。

1) 特 長

- ① データベース項目が豊富（効果的なデータベース活用がはかれる）
 - ② 顧客のデータは、個人としても家族としても利用できる。
 - ③ 取引データを顧客別に集約して保有（必要に応じていつでも取り出せる）
 - ④ コンパクトで、しかもどのようなコードにも対応できる住所処理機能を装備。
 - ⑤ 購入の確認情報をシステムに与えられる。
 - ⑥ キメ細かなコード体系（媒体別につけた商品コード、コンピュータで使う商品コード、取引先のコードなど、豊富なコード体系を保有している）
 - ⑦ 商品をセットとしてオファーできる。
 - ⑧ 代引、クレジット、銀行振込など、決済体系も豊富である。
 - ⑨ さまざまな通信機能を保有している。
 - ⑩ 複数の物流センタを対象に、ビジネスを展開できる。
 - ⑪ 受注から入金にいたるまでの、各段階でのデータを保存しておく。
 - ⑫ 顧客の購入履歴を記録しておく。
- などの豊富な機能を搭載している。

2) 構 成

DM 1100 は、図 2 に示すサブシステムから成り立っている。

1.3 採用情報パッケージ

採用情報パッケージは、企業の採用業務におけ

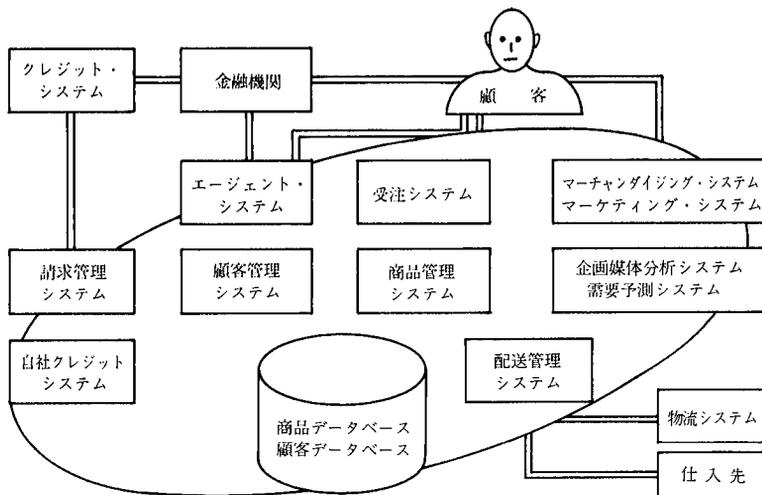


図2 DM1100の概要図

表 1 採用情報パッケージ概要図

採用情報パッケージ										
採用活動	会社訪問	応募	筆記試験	一次面接	本面接	健康診断	内定通知	内定者フォロー	入社	その他
採用業務	訪問者カード起票(訪問者)	訪問者カード追記	筆記試験日決定	一次面接日決定	本面接日決定	健康診断日決定	内定通知日決定	内定者呼出	内定者ファイル更新(入社月日入力)	学校ラベル印書(求人票, 礼状用)
		応募者ファイル更新(登録)	選考日入力	選考日入力	選考日入力	選考日入力	選考日入力	選考日入力	応募者ラベル印書(書類発送用)	応募者ファイル更新(変更・内定辞退・削除)
		応募者ラベル印書	受験者リスト印書(受付用)	受験者リスト印書(受付用)	受験者リスト印書(受付用)	受験者リスト印書(受付用)	受験者リスト印書(受付用)	受験者リスト印書(受付用)	内定者ファイル更新(入社予定日入力)	不合格者処理(不合格者データの削除)
		応募者ラベル印書(呼出用)	応募者ラベル印書(呼出用)	応募者ラベル印書(呼出用)	応募者ラベル印書(呼出用)	応募者ラベル印書(呼出用)	応募者ラベル印書(呼出用)	応募者ラベル印書(呼出用)		統計表
		筆記試験	面接票印書	面接票印書	健康診断	内定通知				学校マスタ更新(照会, 変更, 追加)
		選考結果入力	一次面接	本面接	選考結果入力	内定者処理				辞退者リスト印書
		受験者リスト印書(結果リスト)	選考結果入力	選考結果入力	受験者リスト印書(結果リスト)					OB情報登録
		応募者ラベル印書(結果通知用)	受験者リスト印書(結果リスト)	受験者リスト印書(結果リスト)	応募者ラベル印書(結果通知用)					前職者ファイル登録
		欠席者再呼出	応募者ラベル印書(結果通知用)	応募者ラベル印書(結果通知用)	欠席者再呼出					管理者機能 応募者ラベル・ ファイル更新
			欠席者再呼出	欠席者再呼出						各ファイル印書

る応募者のさまざまな情報を管理し、事務処理の省力化・正確化・迅速化をはかり、戦略的な採用活動を支援するシステムである。

1) 特 長

- ① 採用活動に関する事務処理全般をカバーしている。
- ② 本店一括型の採用方式だけでなく、本支店分散型にも対応できる。
- ③ 応募者や紹介者などからの選考状況の問い合わせに即座に対応できる。
- ④ 応募者や学校に対して通知等を郵送するための宛先ラベルが印書できる。
- ⑤ 各種統計データを簡単に作成し、次年度の採用活動戦略を立てることができる。
- ⑥ 全国の大学・短大・高校を網羅した学校マスタを用意している。

2) 機 能

- ① 応募者の登録
- ② 選考日と選考結果の入力
- ③ 受験者リストと受験結果リストの印書
- ④ 面接票印書
- ⑤ 各種ラベル印書(応募者・求人票・礼状用)
- ⑥ 内定者処理

- ⑦ 不合格者処理
- ⑧ 学校マスタ更新
- ⑨ 統計表
- ⑩ 補助機能

1.4 総合会計システム UNIMACS

UNIMACS は、OA 時代の会計システムとして従来のアプリケーション・パッケージとは異なった、新しい設計思想の下に開発された付加価値の高い総合会計システムである。

1) 特 長

- ① ペーパーレス指向……OA 時代の経理事務合理化に應えるため、ノン伝票、ノン帳表に対応可能となるように設計されている。データベースが MAPPER で作られているため、ノン帳表でも、経理システムで求められている即時見読可能性は保証されている。
- ② データは発生部門における直接入力……データ入力は原則として、データ発生部門において原始証憑より直接入力する方式であり、そのために多彩な専用入力画面が提供され、経理担当者でなくても容易に入力が可能となっている。
- ③ 非定型処理への対応が容易……MAP

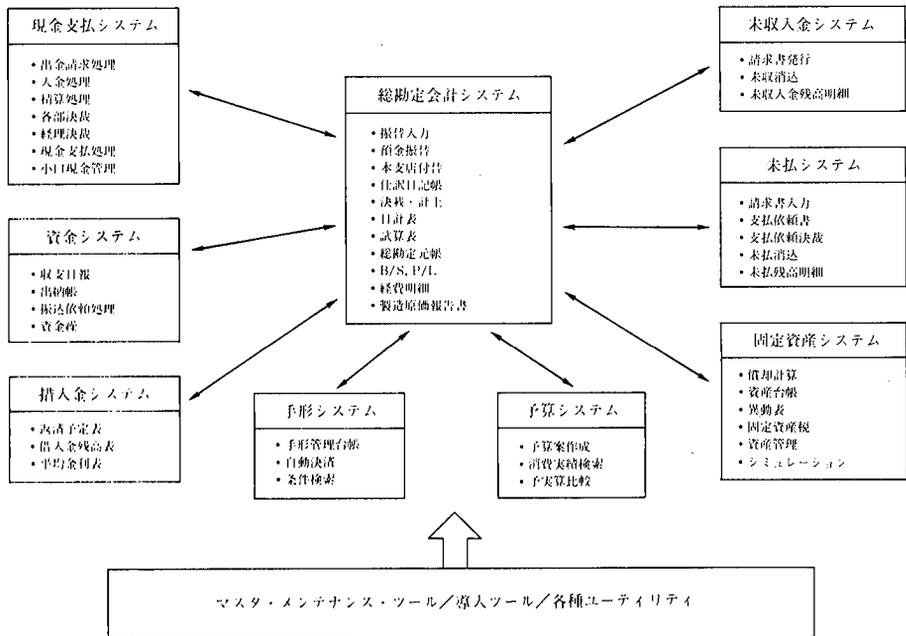


図3 UNIMACSのシステム概要図

PERの会話機能の利用により、UNIMACSで作成された経理データベースに対し検索・加工やグラフ化が容易にできるため、従来のアプリケーション・パッケージになかった非定型的処理が可能である。

- ④ データ入力の合理化……勘定科目コードを入力しなくても、勘定仕訳が可能となるよう各所で考慮が払われている。入力、画面ガイダンス、摘要コード、文言、これらの組み合わせが可能である。また、定型的に発生するデータ用に自動生成マスタ、自動振替用に前払いマスタ、本支店自動振替等が行える。
- ⑤ 複数会計単位の処理が可能……本社および、支店、工場等の事業所別の決算処理、全社合併処理も可能である。小さな事業所では小口現金部所としての処理ができる。
- ⑥ 内部牽制上の考慮……ノン伝票等の新しい運用に対応して、システム上・運用上の考慮が払われている。

2) 機能

九つのサブシステムからなる総合会計システムであり、各サブシステムは互いに有機的に結び付き、データの重複入力は極力排除されている。

- ① 総勘定会計システム……UNIMACSの中核をなすシステムであり、各サブシステムより作成された自動仕訳データ、およびこのサブシステム内で作成される仕訳データ、自動生成データをもとに、決算資料作成までの処理を行う。
- ② 資金システム……他サブシステム、およびこのサブシステム内で作成される資金関連データをもとに出納処理、振込依頼処理、資金繰処理を行う。
- ③ 現金支払システム……毎日発生する取引情報のうち現金の支払い、入金に伴う入力部所の諸管理、および出納部門の事務処理の支援を行う。つまり、交通費、出張仮払い金、経費立替え等の請求精算処理、各部より原始証憑をもとに直接入力処理するノン伝票化の中心となるシステムである。
- ④ 未入金システム……請求書発行から入金による消込み処理、および未取金の残高管理を行う。
- ⑤ 未払システム……受取請求書入力より未払い計上し、支払指示、未払消込み等の未払い管理に伴う各種処理を行う。
- ⑥ 手形システム……受取手形の取得、支払い手形の払い出しから手形の各種移動、お

および残高内容の把握等、手形に関する諸管理が行えるだけでなく、MAPPER データベースより簡単な条件を与えるだけで、割引き、裏書きのために最適な情報を得ることができる。

- ⑦ 固定資産システム……固定資産に関する会計情報から主管部所に対する固定資産情報の管理を行う。また、次年度以降の予想償却費や償却方法変更による償却費算出等のシュミレーションが行える。
- ⑧ 予算システム……予算案作成支援、予算統制のための各種機能、および予実績管理を行う。
- ⑨ 借入金システム……借入金単位に借入時から返済完了時までの利息計算と返済額の算出を行い、月決算に必要な利息情報および借入金管理に関する資料を提供する。

1.5 金融機関向け経営情報システム FIQS 1100

FIQS 1100 は、金融機関が総合的な情報管理システムを構築し、運用する際に必要となる各種ソフトウェアを提供するパッケージ・システムであり、適用業務としては、営業店情報管理・主要顧客取引情報管理・他行店情報管理・地区情報管理等を対象としている。

1) 特 長

- ① 情報の検索や加工・保管が、簡単な画面操作で行える（エンド・ユーザ・ファシリティ機能）。
- ② データベースの管理と検索機能
- ③ データベースの構築とデータの蓄積・管理を行うバッチ機能
- ④ 保有すべき基本的な情報項目と定型帳表の事例

等の機能を用意しており、少人数・短期間での情報系システムの構築を実現させるものである。

また、コンピュータ知識の少ない利用者でも、データの検索や帳表の作成、加工および目的別データベース (MAPPER データベース) への保管等が、画面からの統一された操作で行えるので、利用部門主体のデータ活用が可能となる。

2) 機 能

情報系システム全体と FIQS 1100 の提供する機能の位置付けは図 4 の通りである。

1.6 資産運用総合管理システム FASSET 1100

このシステムは、特定金銭信託および金銭信託以外の信託の全業務を総合管理することを目的に、中央信託銀行(株)と日本ユニバックとで共同開発しパッケージ化したものである。利殖を目的とし、有価証券を主とした運用管理・収益管理が可能のため、信託業務以外に次のような業務にも

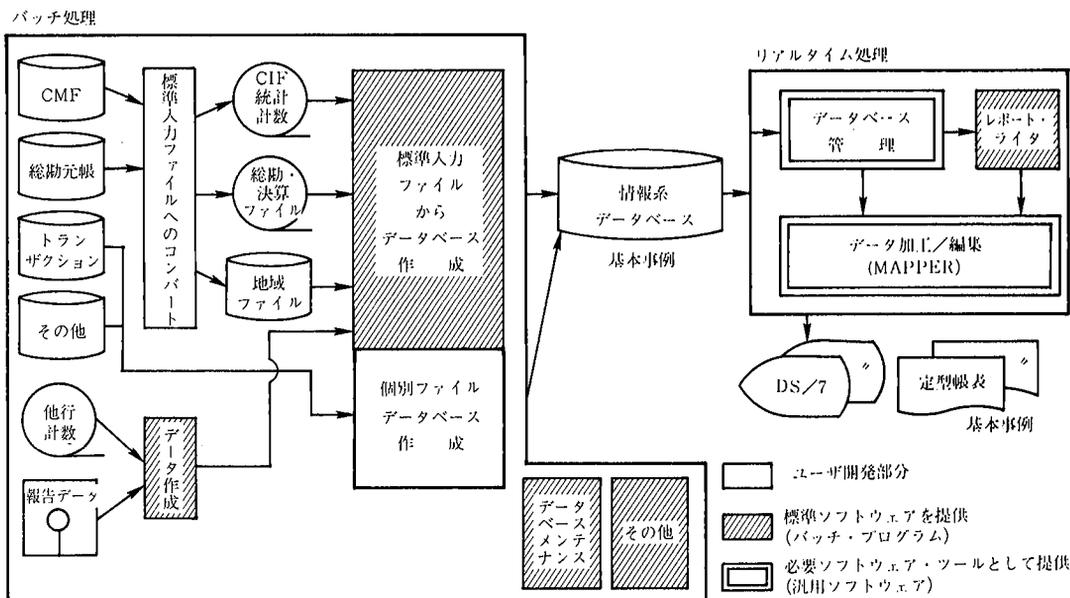


図 4 FIQS 1100 概要

適用可能である。生損保の変額保険用資産運用管理業務、投資顧問業務(以上実績あり)、財テクを指向する一般企業の余資産運用管理等。さらに信託業務では、投資信託・年金信託(運用管理)・常任代理人業務等に適用できる(図5)。

1) 特 長

- ① 資産運用管理(分別管理)に必要な機能を持ち、本システムのみで業務処理が可能である。
- ② 機能別にモジュール化・サブシステム化が計られているため、業務の要求に合わせて機能の追加・変更が容易である。
- ③ このパッケージは MAPPER 1100 で開発されており、データ量に応じて汎用小機から超大型機まで、そのまま運用することが可能である。

2) 機 能

- ① 契約管理……契約情報・元本等の設定・変更
- ② 運 用
 - ・株式(売買約定・配当等の異動/残高管理)

- ・債券(売買約定・利払い・償還等の異動/残高管理)
 - ・短期(コール・ローン, 手形等の受払い・利息管理)
 - ・帳表(残高/異動明細等の出力)
- ③ 会 計
- ・日計(ファンド毎日計表, 資金繰り, 伝票の作成)
 - ・決算(ファンド決算, 銀行決算等)
 - ・精査(運用対象資産別の集計・照合)
- ④ サポート……処理の記録作成, リカバリ等
- ⑤ 約定ベース・未収管理……発生ベースでの残高・運用状況および未収収益等を考慮した運用パフォーマンスの適正評価
- ⑥ 内部照会
- ・報告書(月次・決算時の報告書の内容)
 - ・約定(株式・債券・現先・外国証券等の約定取引状況)
 - ・資産評価(前日終値による時価評価をした純資産額, 簿価・時価対比)
 - ・その他(資金繰り状況(1か月先まで),

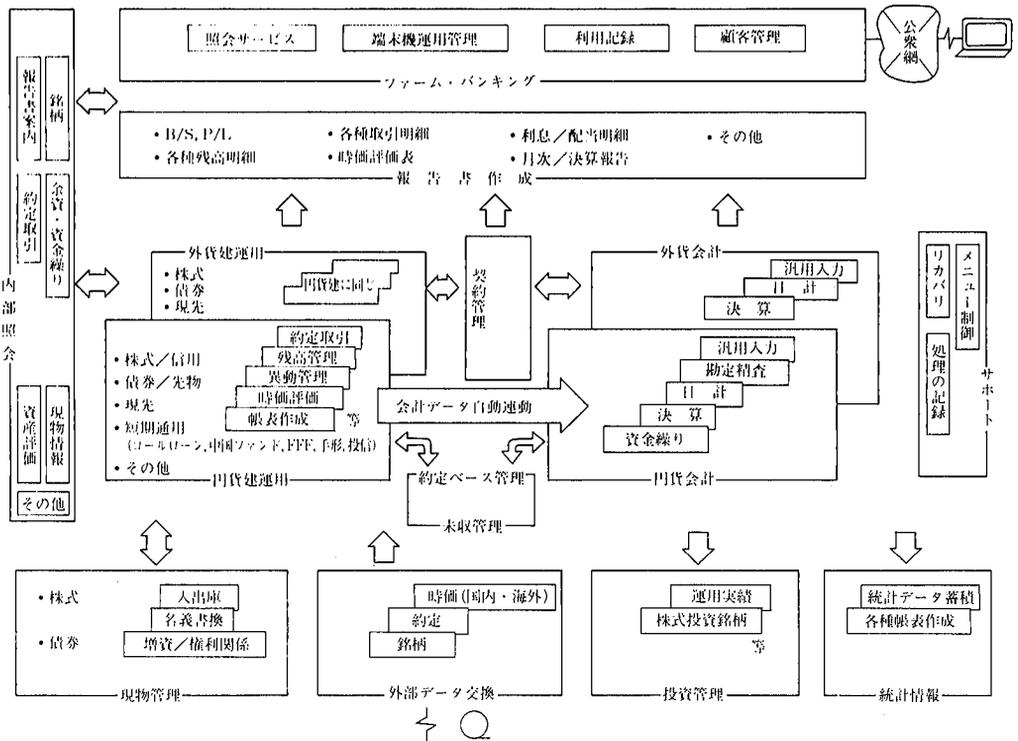


図5 FASSET1100 概要図

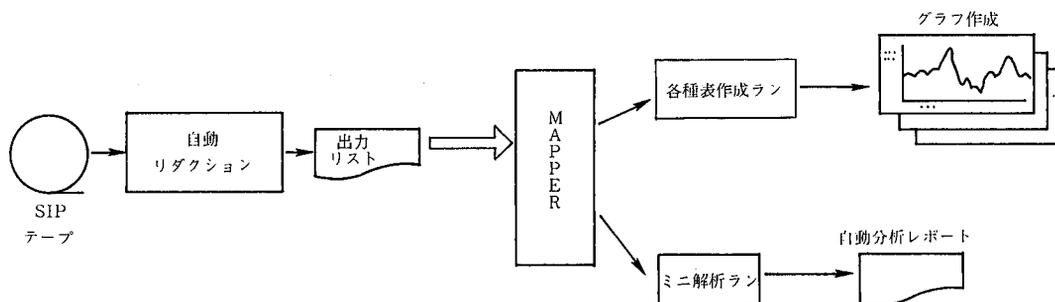


図6 SIP-Gによる作業の流れ

余資運用状況(預金, コール, 現先, TB等の残高), 利配収入状況(6か月間の収入金)

⑦ 報告書作成

- ・月次報告書(貸借対照表, 損益計算書, 利回り表, 残高明細表, 異動明細書等)
- ・決算報告書(月次報告書と同じ, 名寄せ帳表等)

⑧ 現物管理

- ・株式(本券・預り証の入出庫処理, 名義書換, 権利関係処理)
- ・債券(本券(券種・枚数)・登録等の入出庫処理)

⑨ 外部データ交換……時価情報の取込み, 株式・債券約定データの取込み処理

⑩ 投資管理……運用状況表, 資産構成表, 保有銘柄一覧表, 利払・償還表等の出力

⑪ 統計情報……各種データの集計・蓄積処理, 各種帳表の出力

⑫ ファーム・バンキング……顧客および他部門へ端末機を設置し, リアルタイムで各種照会が可能. なお厳重なセキュリティ・チェック機能を持っている.

1.7 SIP 採集データ分析ソフトウェア SIP-G

ソフトウェア・モニタリング(SIP)でシステム稼働情報を採集し, リダクションから分析まで行う作業では経験だけでなく労力が求められる. SIP-Gはこれらの全作業を機械化し, 稼働状況を自動分析して, レポートやグラフを作成するソフトウェアである(図6). その特長は次の通り.

① 自動リダクション……SIP 採集情報を調べ, 最適リダクションを行わせる. MAPPER上の処理効率を考慮し, オート, マニュアル, チェックの3種類のモードがあり, リダクション方式を選択できる.

② MAPPER上の処理……各種時系列表の作成(選択も可能), グラフ表示と印書(選択, 自動出力が可能), ミニ解析レポート(分析情報, 異常診断情報), 自動分析レポート(ミニ解析レポートを元に各コンポーネントを機械分析し, 適切な文章を挿入したレポート, 異常診断結果も含まれる)の作成.

なお, SIP-Gを利用すれば, 中規模システムでも30分程度で分析情報が得られる.

将来的には, 蓄積機能, マルチホスト機能, 見積り機能等の機能拡張をめざしている.

2. その他のパッケージ

1) システム関係

ソフトウェア名	機能
グラフ合成パッケージ	一つのグラフ画面を最大4個に分割し, その各々に指定されたグラフやレポートを合成して描画する.
カナ漢字変換パッケージII	レポート内にあるカナ文字列を漢字に変換する.
DSSM 1100	既存データベースのデータをMAPPERレポートに変換する.
MPC 1100	MAPPERプロトコル・コンバータ(シリーズ8, M 345, UTS 5000をサポートするためのインタフェース)
MPC 1100 (3270)	MAPPERによる他社端末およびパソコンのサポート
MAPPER 1100 LEARN UP コースウェア	MAPPERの会話機能とラン機能の使い方をテープを聞きながら端末を使い, 受講者のペースに合わせて学習する.

2) アプリケーション関係

ソフトウェア名	機能
一般会計処理システム	先日付け仕訳データ入力によるピーク日入力負荷の平準化, 他システムからの自動仕訳データの受け入れ等による経理業務の効率改善と質的向上を目的とする。
予算管理システム	毎月の予算と実績を把握することにより, 予算統制をサポートする。現時点での経費予算の残高や売上予算の達成率を正確に把握できる。
借入金システム	長期短期借入金の登録, 条件変更や借入金台帳, 未払利息明細書の出力等を行う。
固定資産システム	個々の資産を, 取得から処分まで効果的に管理するとともに, 減価償却費の算出により正確な財産価値を把握する。また, 固定資産税の申告も行う。
給与計算システム	毎月の給与計算から年末調整まで, 給与計算業務処理を一貫してサポートする。
企業財務分析システム	簡単な入力作業で, 利用度・加工度が高く, 使いやすい財務データをリクエストに応じ随時出力する。正確な顧客の財務分析を支援する。
顧客管理システム	65万社に及ぶ帝国データバンクの企業情報データベースの中より, 自社に必要なデータだけを MAPPER データベースに取り込み, 検索, 更新, DM 作成等の顧客管理を行う。
個人信用情報システム	取引関連企業や個人の信用情報を一元管理し, 必要に応じて即座に検索, 照会, 各種統計資料作成を行う。
抵当証券販売管理システム	証券の顧客への割付から証書発行, 利息支払, 償還中途解約等販売にかかわる一連の処理を行う。
人事情報システム	社員の個人情報を一元管理し, 必要に応じて容易に検索, 照会, 住所録作成, 各種統計資料作成等を行う。戦略的な人材配置やきめ細かな人材育成に有用である。
特許管理システム	特許・実用新案の出願から権利消滅まで, 複雑な特許管理業務の精度・効率を高める。
秘書室システム	秘書室の仕事の質と生産性を高めるためのツールで, 多忙な役員が抱えているさまざまな用件を効率良く管理し, 容易に検索できるようにする。
QC 総合支援システム	QC 活動の全域を総合的に支援し, QC 活動の効率と効果を大幅に高める。

3. MAPPER1100 ユーティリティ

MAPPER 1100 システムを使用する際に, 便利な汎用的なユーティリティをリスト・アップする。

ソフトウェア名	機能
ACTRID	モード, タイプごとのレポート数・ライン数の集計
ADDDEL	レポートの追加・削除
CROSS	文字列の検索
FMTGEN	RID 0 の構築
KSUP, GREPO	必要項目の収集およびレポートの合成
KCDSP	罫線/カラー画面作成
KCOP	レポート罫線印番
LINS	行の挿入
EX.xxx	132 桁レポートの会話型処理
NFCC	複数レポートの FCC の作成
SSETUP	MAPPER レポート/SDF ファイル間のレポート転送
TCOPY	モード, タイプの移動
MAPPER-EXT	MAPPER レポート作成サブルーチン (ACOB)
MPA	ファイル分割アサイン

浦伸一の **MAPPER** によるユーザ・リアルタイム・システム事例は、通常の実時間・システムおよびバッチ・システムを **MAPPER 1100** のフレームワークに取り込み、利用者とのインタフェースをすべて **MAPPER** に集約させた事例について述べている。これによって、利用者は、業務の処理形態を意識せずに、**MAPPER** の操作だけで、すべてのプログラムを実行できる。

MAPPER によるアプリケーション開発では、同一レポートへの多重更新と異常終了発生時におけるリカバリを考慮する必要がある。通常の場合は、システムで用意されている命令で対処できるが、6 レポート以上の一括更新とロック中の禁止命令の実行などを行う時は、特別な配慮が必要である。岡島伸介の **MAPPER** における排他制御とリカバリ方法の考察は、上述の問題の解決方法について紹介している。

航空運賃割引管理エキスパート・システム **SEATS** (Airline Seat Advisor) は、Sperry (現 Unisys)、Northwest 航空、IntelliCorp の 3 社によって開発された、割引引き座席の管理による収益調整システムである。**SEATS** は、エキスパート・システム構築ツール **KEE** を用いて開発されており、**MAPPER** 上のシステムから情報を受け取って推論し、割引引き座席の設定について助言する。工藤隆司の人工知能と **MAPPER** —— **Northwest** 航空における事例は、**KEE** と **MAPPER** とのインタフェース・ソフトウェア (**KEEMAP**) の概要について述べている。

岩佐一夫の **MAPPER** と光ディスク——花王 (株) における事例は、スタンド・アローン型の光ディスク・ファイル・システムをホスト・システムと接続して、**MAPPER** (および文献検索システム **CIR-J**) による 2 次情報の検索と光ディスク・ファイル・システムの原始文書イメージの検索を統合した事例について紹介している。

▶ 技報編集委員会

委員長 柳生孝昭
副委員長 米口 肇
委員 新野清嗣, 田中 博, 中村 脩, 永田利地, 西原良一, 野本雄一, 藤田康範, 前田英次郎, 榎 元治, 村井啓一, 山田達也

▶ 編集担当

技術研究部 朝倉文敏, 高橋 肇

▶ 制作担当

技術情報サービス部 青柳幸久, 丹野敬子

● Editorial Board

T. Yagi (Chairman)
H. Yoneguchi (Vice Chairman)
K. Shinno, H. Tanaka, O. Nakamura,
T. Nagata, R. Nishihara, Y. Nomoto,
Y. Fujita, E. Maeda, M. Maki,
K. Murai, T. Yamada

● Editorial Staff

F. Asakura, H. Takahashi
(Technical Research Dept.)

● Publications Staff

Y. Aoyagi, K. Tanno
(Technical Information Services Dept.)

ISSN 0289-6257

技 報

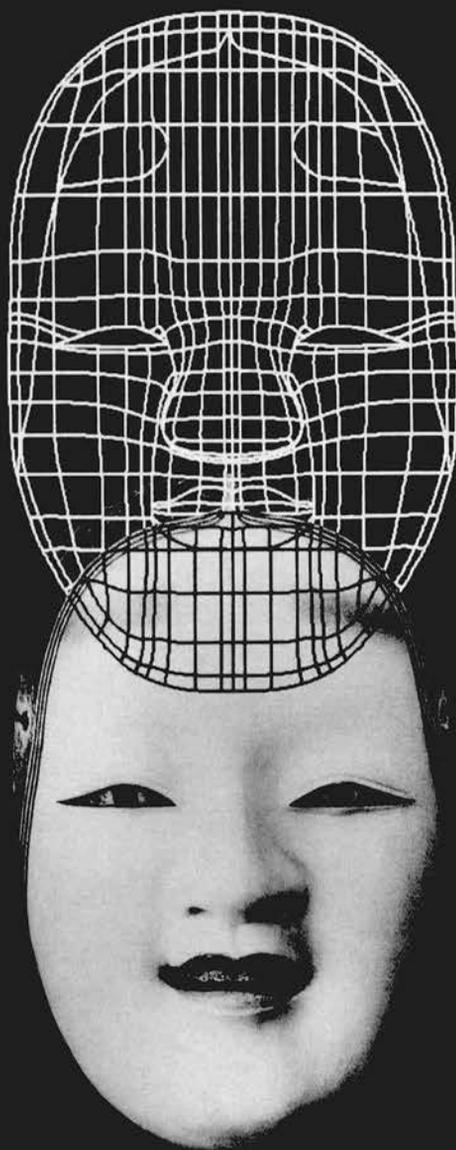
UNIVAC TECHNOLOGY REVIEW

No. 14

発 行 日	昭和 62 年 8 月 31 日
編 集 人	柳 生 孝 昭
発 行 人	富 田 和 夫
発 行 所	日本ユニバック株式会社 東京都港区赤坂 2-17-51 〒 107 TEL (03) 585-4111 (大代表)
頒 布 価 格	1,500 円
印 刷 所	三美印刷株式会社

禁無断複製転載

美曲面師。



企業の個性をシステム化する

UNIVAC

日本ユニバック 東京都港区赤坂2-17-51 03(567)4111
日本ユニバック情報システム 東京都港区赤坂2-17-22赤坂フインタワー本館 03(567)8111

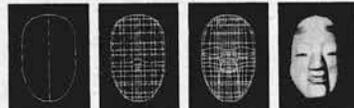
微妙な曲面のニュアンスをリアルに表現する美曲面師。設計デザインの新時代をひらく UNICAD®/SOLIDの世界。

自由曲面をリアルに、スピーディに表現。プレゼンや解析にも威力を発揮します。

多彩なシェイディング機能に加え●アンチエイリアシング●透明表示●陰線処理など豊富な表示機能を搭載。また設計情報を3次元データで記憶しているため、表示角度も自由自在。構造解析も行なえます。

彫刻のように、自由な感覚で設計を推進。設計データの製造ライン送出もOKです。

手作業感覚で設計が行なえ、デザイナーは思いのままにアイデアを試していけます。しかもUNICAD/SOLIDはユニバック独自のFAコンセプト“ALPFA”に基づいて作られており、CAMとの連動も万全です。



●UNICADの基本モジュールは、トヨタ車体株式会社とユニバックが共同開発したCAPES (Computer Aided Product Engineering System)の基本部分をもとに構成。●ALPFA (アルファ): All-around Linked Productive Factory Automation

複合自由曲面のリアリズムCAD。新世代ソリッド・モデリング・システム、誕生。

UNICAD/SOLID