

RDBをベースとした計画管理情報システム PLANNER

神田康敬, 岸本雄一, 小関伸夫, 甲田一也, 山下頼安
甘泉知雄, 春木尚美

(富士通株式会社)

1. はじめに

最近, エンドユーザ指向のシステムへの要請が高まり, 計算機に縁のなかった部門の人々が積極的に計算機を利用しようとしている。計算機利用の拡大の背景には, 氾濫した情報の一元管理, OA指向のオフィス部門の合理化といった利用者側のニーズと, 高速なCPU, 大容量のメモリ, ディスク装置, 日本語情報処理, データベース技術に見られるような技術面の発達がある。また, EDP部門のスタッフもできるだけ省力化できるような簡易な運用管理が可能な業務システムの構築を望んでいる。

計画管理業務という分野について考えてみると, 従来から, 統計解析, 帳表作成, グラフ作成などの機能単位に開発されたパッケージを組み合わせ, データの管理は, ネットワーク型のデータベース又は各パッケージの専用ファイルで行うというシステムが一般的である。

しかしながら, 非定型処理の多い業務をカバーし, かつデータベースに関する知識のほとんどない計画管理部門のユーザでも利用でき, 業務システムをエンドユーザのために開発し, 運用するEDP部門にも負担のかからないようなシステムへのニーズは大きく, 従来のアプローチだけでは, 不十分ではないかと思われる。

我々は, このような要請に答えて, リレーショナルデータベースを核としたエンドユーザ向きシステムPLANNER (PLANNing and management information system based on Easy RDB)を開発した。

PLANNERは, FACOM Mシリーズ OSIV/F4又はOSIV/X8のもとで稼動し, 1981年1月提供以来, 民間企業, 官公庁, 自治体をはじめとして, 多方面で利用されている。

本論文では, はじめに, PLANNER開発における基本的な考え方を述べ, その機能概要を紹介する。RDBは, 実務への適用により, その強力なデータ操作に対し, 高い評価が得られたが, 計画管理業務特有のデータ編集処理に対して, いくつか機能的に不十分な点も明らかになった。後半では, 具体的な例をまじえて考察する。

2. 開発における考え方

2.1 開発の背景

一口に計画管理業務といっても, 課単位の予算管理のような小規模なものから, 全社的な販売情報分析管理, 行政統計データ管理などのように幅広い範囲の業務があるが, これらの業務を支援する汎用ソフトウェアとしては, 次のような要件を満足すべきであろう。

(1) 多様で複雑な情報を容易に管理できるような理解し易いデータベース。

- (2) 非定型な検索処理を簡単に行える強力なデータ操作言語をもつこと。
- (3) データの検索のみならず、集計、加工分析、編集処理が一体化して行えること。
- (4) 検索、加工分析結果をユーザが理解しやすい形で表示する機能をもつこと。
- (5) データベースの創成、運用、管理が容易なこと。(EDP室の負担軽減)

リレーショナルデータベース(以後RDBと略す)を選択した理由は、単純なデータモデル(表形式)にもとづくため、データベースの設計・管理が容易であり、データの関連付けを得意とする簡便で強力なデータ操作言語をもっているのので、上述の要件によく適合するデータベースシステムであるからである。ただし、計画管理業務では、データの分析、分析結果のグラフ表示、レポート出力などが必要であり、RDBのデータ操作プラスαが必要である。従来のシステムは、ユーザがホスト言語インタフェース等を用いて個別にプログラムを組むか、機能別のパッケージをファイル経由で組み合わせているものが多い。そのためにEDP室にアプリケーションプログラム開発の負担がかかったり、エンドユーザが、複数のパッケージの使い方の習得に多くの時間を費やす必要があった。我々は、RDBの強力なデータ操作に加え、データの加工編集、統計処理、グラフ出力、地図出力などの一連の操作が統合化されたエンドユーザ言語 QAL (Query and Analysis Language) で対話的に処理が行えるシステムを目標に、PLANNERを開発した。

2.2 開発目標

PLANNERでは、次の事項を開発目標においている。

(1) 一体化したユーザ言語の開発

データベースのデータ操作言語とデータ加工編集、統計処理などのデータ処理言語を統合し、一体化したエンドユーザ言語を開発する。これによって、ユーザは、検索から分析結果の出力まで覚えやすいコマンドにより対話的に処理を進めることができ、より業務上の処理効率を高めることができる。

(2) アプリケーション属性の管理

時系列データの処理や地理的データの処理の機能向上、使い易さを高めるために、アプリケーション特有の属性を効果的に管理する方法を開発する。

(3) 日本語処理機能の開発

エンドユーザシステムとしては、当然のことであるが、日本語の入出力が容易に行えるための日本語処理機能を開発する。

(4) 運用のためのユーティリティ開発

データベース管理者の負担軽減のために、簡易に使用できるユーティリティを開発する。

(5) アプリケーションサブシステムの追加機能

多様なユーザニーズに応じて、次々と機能追加が容易に行えるように、オープンエンドなシステム構成とする。

(6) 最新の端末、出力機器への配慮

カラーグラフィックディスプレイ、日本語端末、日本語ラインプリンタなどの最新の端末、入出力機器とのインタフェースを十分考慮した開発を行う。

2.3 システム体系

PLANNERは、ユーザが端末より対話的に処理を進めるためのユーザ言語 QAL をサポートする対話型処理システムと、RDB の初期設定、大量データの格納、バックアップなどのユーティリティ機能をもつ CREATOR (CREATION utilities Of Relational database) に分けられる。図1は PLANNER のシステム構成を示したものである。

対話型システムは、RDB のデータ操作を行うリレーショナルサブシステム¹⁾ と目的に応じたデータ処理を行うアプリケーションサブシステム (APS) 群から成る。APS には、データの加工編集を行うデータ加工編集サブシステム、基本統計、クロス集計などの統計処理を行う統計処理サブシステム、データベースの内容照会や検索、コマンドの日本語編集などもウルスクリーン画面で行えるメニュー処理サブシステム、結果をカラーグラフィックディスプレイや日本語ラインプリンタなどに出カするグラフ・地図出カサブシステムがある。また、コマンドインタフェースは、入力されたコマンドがどのサブシステムに属するかを判断し、各サブシステムにコマンドを振り分ける。また、プログラムインタフェースにより、ユーザの応用プログラムから、RDB へのアクセスも可能である。

(1) リレーショナルサブシステム

対話型リレーショナル問合せサブシステム (ISRQ) とデータ記憶・制御サブシステム (DSCS) から成る。

ISRQ (Interactive Subsystem for Relational Query) は、RDB に対するデータ操作言語を解釈実行する。データ操作言語は、テーブル、インデックス、ビューの定義と削除、タプルの検索、更新、挿入、削除と表示などのコマンドからなる。

ISRQ の機能モジュールは、パーサー、セマンティクスチェッカー、スケジューラ及びインタプリタである。

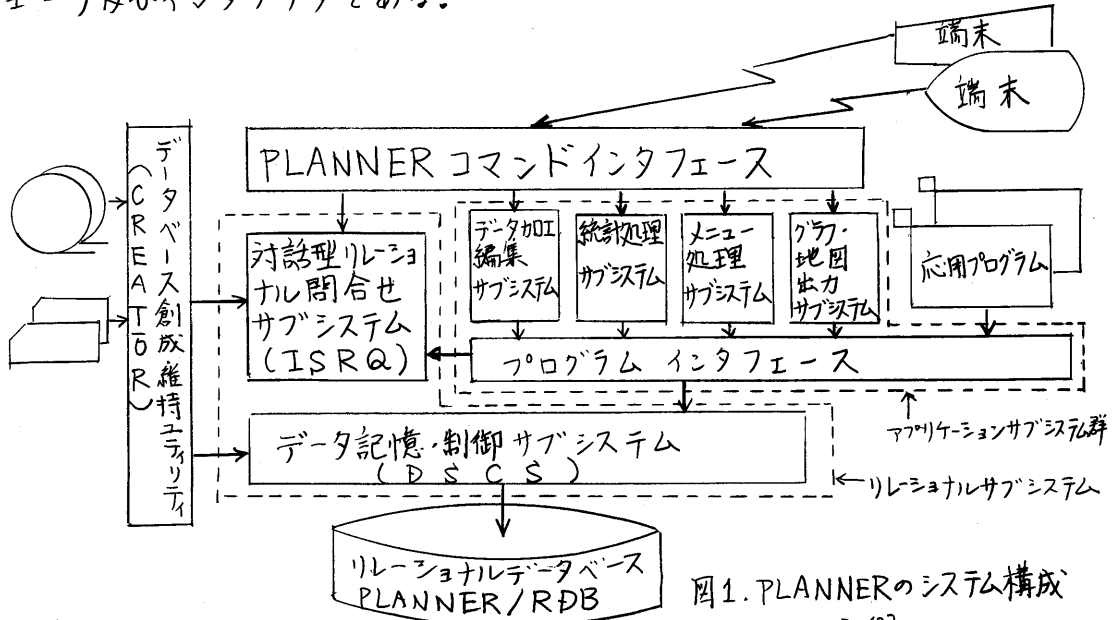


図1. PLANNER のシステム構成

1) リレーショナルサブシステムは、富士通研究所開発の RDB/V1 を母体^{(1), (3)}としている。

パーサーは各コマンドの構文解析を行い、構文情報をデータ構造に変換する。セマンティクスチェッカーは、データ辞書を参照して、コマンド中のテーブル、ビュー、フィールドに関する意味チェックを行う。スケジューラは、データ辞書を参照し、アクセスパス（インデックスなど）を選択して、最適な実行手順を決定する。インタプリタはその実行手順に従って、DSCSの提供するインタフェースを用いて、RDBをアクセスし、コマンドを実行する。

最適な評価手順を決定する方法は、他のデータベースシステムでも種々考察されているが、リレーショナルサブシステムでは、カスケード法、ミニマックス法などに代表される技法を用いている。⁽⁵⁾

DSCS (Data Storage and Control Subsystem) は、RDBのデータ編成法とアクセス法を提供する。データ編成法としては、4Kバイトのページに、順次タプルを格納する順編成がある。テーブルのアクセスパスとして任意のフィールドにインデックスを作ることができる。インデックスはB+tree構造であり、そのリーフページには、〈キー値、タプル識別子〉の組を格納する。

DSCSのインタフェースは、テーブルの創成、削除、インデックスの創成、削除、テーブル、インデックスのスキーマ及びタプルのフェッチ、挿入、削除、フィールド値の更新機能をサポートする。

上述の他にDSCSは、ページ単位のI/O管理、排他制御の機能を有している。また、リカバリのためのログの取得も近い将来サポート予定である。

(2) アプリケーションサブシステム (APS)

アプリケーションサブシステム群は、計画管理業務のために、リレーショナルサブシステムが提供するデータ操作だけでは不十分な機能を補うためのデータ処理機能を提供する。

APSは、時系列データ属性、地理的データ属性などのアプリケーション特有の属性を必要ならば参照し、質の高いデータ処理を行うように設計してある。時系列のテーブルを例にとると、そのテーブルがどんなタイプ（データの収集期間により、年次、半期、四半期、月次の4つのタイプがある）のテーブルで、そのキーフィールドはどれであるかというような意味的情報をシステムが参照し、コマンドの処理に役立つことにより、ユーザはコマンドで陽にキーフィールド名を指定することがなくなり、負担が軽減される。また、APSでは、この属性情報の参照以外に登録、更新が可能である。属性情報の参照を円滑に行うためにISRQはGETコマンドに属性の遺伝機能をもたせている。

(3) データベース創成・維持ユーティリティ (CREATOR)

データベース運用者のためのユーティリティで、通常は、バッチで動作する。RDBの初期設定、パスワードの設定、テーブルの創成、SAMファイルからのデータ入力、大量データの更新、RDBの退避・復元、地図情報の登録などの機能を有する。

3. PLANNERエンドユーザ言語 (QAL)

PLANNERが提供するエンドユーザ言語は、QALと呼ばれ、そのコマンドシンタックスは、データ操作言語もデータ処理言語も同一の言語体系になっており、覚えやすい言語である。日本語尚未使用時には、メッセージ、出力結果が

すべて日本語で出力される。

PLANNERの主なコマンドを表1に掲げる。表1の他にも、メニュー処理サブシステムを呼出すMENUコマンドやコマンドの編集を行うCATALOGコマンド、それを実行するRUNコマンドなどの制御コマンドがある。

3.1 データ操作言語

リレーショナルサブシステムの提供するデータ操作言語は、SQL2とQUELレベルの構文と意味を有し、一部、差のある機能を有する。ここでは、代表的なコマンドとして検索コマンドについて説明する。

検索コマンド (GET)

GETコマンドは、テーブルを条件検索し、条件を満たすタプルの集合を求める。結果は新たなテーブルであり、再度検索可能である。検索項目には、フィールド名の他に、集合関数(MAX, MIN, AVG, SUM, COUNT), 定数を含む任意の算術式を許す。また、2つのフィールド名を指定し、その間にあるすべてのフィールドを検索する区間指定やある特定の項目を除外する除外項目指定ができる。

複数テーブルを対象とするとき、*equi-join* 述語を指定すれば、通常のJoin演算を行う。*equi-join* 述語のないとき、Cartesian処理を行う。ただし、運用パラメタの選択により、Cartesian処理をエラーにする運用も可能にしている。

検索条件は、等式、不等式、メンバーシップ関係式を論理積(AND), 論理和(OR)で結合して表わす。また検索結果のタプルの集合を対象として、指定フィールド値でグループ演算を行い、条件を満たすグループのみを選択する機能がある。検索時にタプルをソートすることもできる。

一回のGETコマンドで別々に検索したタプルの集合に対し、集合演算が行える。集合演算子として、intersection, difference, Union, appendがある。

ビューに対する検索は、テーブルと全く同じようにできる。[4]ただし、ビューを通しての更新操作は行えない。

3.2 データ処理言語

データ操作以外で、計画管理業務に必要な機能を提供する言語を総称してデータ処理言語と呼ぶ。表1に示すように、加工編集、統計処理、グラフ・地図出力に分けられる。

(1) データ加工編集コマンド

計画管理業務では、しばしば時系列データのテーブルに対するデータ操作が行われ

表1. 主なQALコマンド一覧

| 機能分類 | コマンド名 | 機能概要 |
|--------------------------------|---------------------|--|
| データ操作言語 (データ表言語統) (30種類) | GET | 問い合わせ検索 |
| | PUT | データベースへのワークテーブル保存 |
| | CREATE TABLE | テーブル作成・定義 |
| | MOVEFILE | ファイルからのレコード挿入・追加 |
| | READ | カード形式データのレコード挿入・追加 |
| | INSERT | 端末からのレコード挿入・追加 |
| | UPDATE | テーブルのデータ更新 |
| | DELETE | テーブルのレコード削除 |
| | EXPAND TABLE | テーブルのフィールド追加定義 |
| | DEFINE VIEW | ビューの定義 |
| | CREATE INDEX | インデックスの作成 |
| | DROP TABLE | テーブルの削除 |
| | DROP VIEW | ビューの削除 |
| | DROP INDEX | インデックスの削除 |
| | RENAME TABLE | テーブル名の変更 |
| | RENAME FIELD | フィールド名の変更 |
| DISPLAY | テーブル内容の増表示 | |
| PRINT | テーブル内容のライプリンタ出力 | |
| SHOWDB | データベース上のテーブル名参照 | |
| データ処理言語 | 加工編集コマンド (15種類) | SORT ORDER RANK MERGE TRANSPROSE TRTS COMPTS1 など |
| | 統計処理コマンド (6種類) | STATIS CORR MULTIREG CROSSTB2 など |
| | グラフ・地図出力コマンド (23種類) | LINE BARGRAPH HISTOGRAM PIECHART LBARGRAPH SCATTER ZONEMAP MESHMAP など |
| | | フィールド値によるソーティング |
| | | フィールド値による順位付け |
| | | フィールド値によるランク付け |
| | 2テーブルの併合 | |
| | テーブルの転置 | |
| | 時系列データの編集 | |
| | 対前期比較 | |
| | 基本統計量の算出 | |
| | 相関係数の算出 | |
| | 重回帰分析 | |
| | クロス集計表 | |
| | 折れ線グラフ作成 | |
| | 棒グラフ作成 | |
| | ヒストグラム作成 | |
| | 円グラフ作成 | |
| | 層層グラフ作成 | |
| | 散布図作成 | |
| | ゾーンマップ作成 | |
| | メッシュマップ作成 | |

る。時系列テーブルは、年や月単位のタプルの集合である。時系列データの処理として、期、月単位のテーブルを年や期単位のテーブルに集計編集するTRTSコマンドがある。また、前月、前年同月、前年同期のデータ値との比較もよく行われるが、このためのコマンドとしてCOMPTRTSコマンドがある。これらの時系列編集コマンドは、年月を指定する単純な検索ではなく、時系列に関する属性情報があれば、それを参照し、時系列特有の概念にもとづく計算を行う。

データ操作の拡張の例として、RANK、MERGEなどがある。RANKは、グループ演算の拡張で、あるフィールド値の値の範囲でタプルをグループ分けし、各グループに同じランク値を与えるものである。一方、MERGEは、JOIN演算の拡張である。2つのテーブルをキーフィールドで結合することはJOIN演算と同じであるが、一方のテーブルにしかないキーをもつタプルも除外せずに、値のないフィールドにはユーザ指定の値(欠測値も可)を入れる。

(2) 統計処理コマンド

基本的な統計分析を行うためのコマンドとして、STATIS、CORR、MULTIREGの各コマンドがある。STATISコマンドは、最大、最小、標準偏差などの基本統計量を算出し、CORRコマンドは、フィールド相互の相関係数を求める。MULTIREGコマンドは、重回帰分析を行う。

CROSSTBコマンドは、テーブルの2つのフィールドを対象として、それぞれの値によって複数のレベルに分け、各レベルごとに該当するタプル数を求めたり、フィールド値を合計する。

(3) グラフ・地図出力コマンド

GETコマンドで求めた検索結果をユーザにとって見易いグラフや地図の形で出力する機能である。主なグラフとして、折れ線グラフ、散布図、棒グラフ、ヒストグラム、円グラフ、夕変量円グラフなどがあり、時系列の変化表示や営業店ごとの比較表示に適している。

地図は、日本語ラインプリンタ、一般のラインプリンタのどちらにも出力可能である。特に日本語ラインプリンタでは、地名や凡例を日本語で出力でき、鉄道などの線データも重ね合せて出力できる。

地図の境界線や道路などの線データの位置情報もテーブルで管理しており、これらと実データ(例えば地区別売上高)の入っているテーブルを関連付けて地図出力する。データの関連付けは、実データテーブルに定義された地理的属性情報を参照して行う。ユーザは、テーブルのキーフィールドである地区コードのフィールド名やそのテーブルを地図出力するための境界線データ格納テーブルなどを一度定義しておけば、コマンド入力時には、実データと地図の境界線データとの関連を意識することなしに結果が得られる。

4. 計画管理業務へのRDB適用の効果と課題

RDBは、強力なデータ操作言語をもち、計画管理業務のような非定型処理の多い分野では、その力を発揮する。事実、ユーザでの適用を通して、次のような評価を得ている。

(1) 表形式のデータ構造は理解し易く、設計が楽である。

(2) 短期間でデータベースセットアップが可能になり、エンドユーザへの迅

速なサービスが可能になった。

(3) GETには豊富な機能があり、従来プログラムを作成しなければならない処理が、コマンドあるいは複数コマンドの組合せで行える。

一方、計画管理業務には、特有のデータ編集処理があり、現在のRDBのデータ操作では、困難なものもいくつか明らかになった。これらは、次のような点に起因する。

- ・ 時系列データの処理には、時系列特有の概念にもとづいて編集するものがある。
- ・ 計画管理業務に携わるエンドユーザのデータモデルとリレーショナルデータモデルとの間にギャップがある。
- ・ グループ集計機能が機能的に不十分

PLANNERでは、これらの問題点に対し、加工編集コマンドを強化することにより、対処してきた。以下では、これらの問題点のいくつかを具体的な例をもとに考察してみる。

(1) 時系列データの編集

時系列データの編集に関して、GETコマンドでかなり場合は対処できる。はじめに、GETコマンドによる時系列データの編集の例を示したのち、GETコマンドでは、対処の困難な例をあげる。

例題として、ある企業の販売情報管理システムを考える。全国各地の営業所から、契約のあるたびに、その日付と売上がデータエントリされる。売上情報は一年間を一つのテーブルとして、データの発生順にタプルが挿入される。(図2) 通常、計画管理では、図2のような生データを月次、年次などに集約したデータを用いる場合が多い。

事例1. U82のテーブルに対し、各営業所の月次ごとの売上高を求め、テーブルS82とする。

```
GET INTO S82 CODE, MONTH, U = SUM(U)
FROM U82
GROUP BY CODE, MONTH;
```

時系列

U80

U81 (1981年売上高テーブル)

U82 (1982年売上高テーブル)

| 営業所コード CODE | 月 MONTH | 日 DAY | 売上高 U |
|----------------|------------|----------|----------|
| 100 | 1 | 27 | 1000 |
| 200 | 1 | 29 | 1200 |
| ... | ... | ... | ... |
| 100 | 2 | 8 | 1500 |

図2. 売上高テーブル

この例では、グループ化のためのキーは少ないが、商品の種類、地区別など、実際には、いろいろなキーでグループ化されるケースが多く、GROUP BY機能は、非常によく使われる機能の一つである。

以下では、事例1で求めた営業所別月次売上高テーブルS82(図3)を対象にした操作を考える。

S82は、営業所コード、月をキーとする時系列テーブルである。各月ごとのテーブルに変換するには、ビューを利用する。例えば、8月の各営業所の売上高をV8(CODE, U)とすると、以下のように定義できる。

```
DEFINE VIEW V8 AS CODE, U FROM S82
WHERE MONTH = 8;
```

S82

| 営業所コード CODE | 月 MONTH | 売上高 U |
|----------------|------------|----------|
| 100 | 1 | 3200 |
| 100 | 2 | 2700 |
| ... | ... | ... |
| 200 | 1 | 1800 |
| 200 | 2 | ... |

図3. 営業所別月次売上高テーブル

上記のビューを用いると、次のような検索が簡単に行える。

事例2. 1982年の7月と8月を比較し、売上高の減少した営業所を求める。

```
GET A.CODE
FROM A:V7, B:V8
WHERE A.CODE = B.CODE AND A.U > B.U ;
```

7月および8月の売上高を定義したビューV7, V8をJOINすることにより求める。S82のテーブルのみを用いても上記の検索は可能であるが、シンタックスが複雑になり、エンドユーザ向きではない。

事例3. S82と1981年の営業所別月次売上高テーブルS81を用いて、各営業所の売上高対前年同月比を求める。

```
GET X.CODE, X.MONTH, R=X.U/Y.U
FROM X:S82, Y:S81
WHERE X.CODE = Y.CODE AND X.MONTH = Y.MONTH ;
```

次に、GETコマンドでは、処理が困難な例を示す。これらは、時系列特有の計算が必要になる例である。

事例4. 1980~1982年の3年間の営業所別月次売上高テーブルS80, S81, S82を用いて、3年間のA営業所(CODE=100)の売上高対前月比を求める。

S80, S81, S82のテーブルから集合演算子appendにより一つのテーブルを作り(図4)、順次タプルどうしの比を求めるのであるが、1980年12月と1981年1月のように年(YEAR)が変わるタプルどうしの売上高の比を求めることは、GETコマンドでは困難である。PLANNERでは、COMPTS コマンドを用いれば、簡単に求められる。

```
GET INTO S3(YEAR, MONTH, U)
1980, MONTH, U FROM S80 WHERE CODE=100
(+)
1981, MONTH, U FROM S81 WHERE CODE=100
(+)
1982, MONTH, U FROM S82 WHERE CODE=100;
COMPTS U FROM S3 ;
```

S3

| 年 | 月 | 売上高 |
|------|-------|------|
| YEAR | MONTH | U |
| 1980 | 1 | 800 |
| | 2 | 900 |
| | ... | ... |
| 1980 | 12 | 1100 |
| 1981 | 1 | 1200 |
| | 2 | 900 |
| | ... | ... |
| 1981 | 12 | 1400 |
| 1982 | 1 | 1500 |
| | 2 | 1400 |
| | ... | ... |

図4. 3年間のA営業所売上テーブル(S3)

事例5. 事例4で求めたS3テーブルの売上高を四半期ごとに集計する。

キーフィールド(YEARとMONTH)の値がある値の範囲(例えば1980年1月~3月)にあるタプルを同一グループと見なして小計を求めていくような機能がGETコマンドがあれば処理可能となる。時系列テーブルの編集用コマンドTRTSがこの機能を提供する。

```
TRTS U FROM S3 WITH TTYPE=Q ;
```

↑ 四半期での編集を意味する。

(2) エンドユーザの考えるデータモデルとリレーショナルデータモデルとのギャップ

時系列テーブルの設計の例として、図5のような形のテーブルを用いて、議論を進めてきたが、計画管理業務では、時系列の表は、図6のような形でものが一般的である。これは、売上高という実データが2次元的な形で表現されており、理解しやすく、他の月との比較や半期、四半期などの集計もしやすい形になっているからである。一方、新規データの発生に対しては、図5は、タプルの挿入で済むのに対し、図6では、新しくフィールドを追加し(EXPANDコマンド)、追加されたフィールド

にデータを入れる(更新操作となる)必要がある。従って、データの挿入という面からは図5の方が好ましい。

そのため、テーブルは、図5の形でもち、必要に応じて、図6の形のテーブルに変換するような2つのデータモデルのギャップを埋める機能が必要になる。

| 営業所 | 月 | 売上高 |
|-----|-----|------|
| 100 | 1 | 1000 |
| 100 | 2 | 1200 |
| ... | ... | ... |
| 100 | 7 | 1300 |
| 200 | 1 | 1400 |
| 200 | 2 | 1500 |
| ... | ... | ... |
| 100 | 8 | 1800 |
| 200 | 8 | 2000 |

図5. 時系列テーブルの例1

| 営業所 | 売上高 | | 7月売上高 | 8月売上高 |
|-----|------|------|-------|-------|
| | 1月 | 2月 | | |
| 100 | 1000 | 1200 | 1300 | 1800 |
| 200 | 1400 | 1500 | ... | 2000 |
| ... | ... | ... | ... | ... |

図6. 時系列テーブルの例2

PLANNERでは、そのためにTRANSPPOSEコマンドを用意している。

5. 結論

本論文では、リレーショナルデータベースシステムを中核にすることにより、リレーショナルデータモデルの特長を十分に生かした高度な対話型計画管理情報システムが構成できることを示した。また、リレーショナルデータベースのデータ操作だけでは不十分な機能を補うデータ処理言語をデータ操作言語と一体化して提供することは、使い易いシステム構築という面から、非常に重要であると考えられる。

PLANNERは、現在、企業におけるマーケティング情報管理、経営情報管理、自治体における行政統計データバンクシステム、地域情報管理など多方面に渡って利用されている。性能に関しても、リレーショナルデータベースの最適化処理とインデックスを使ったチューニングによって、処理効率上問題ないことを確認している。

また、時系列編集をはじめとする編集コマンドもエンドユーザから好評であるが、機能的にさらに豊富にすることを目指している。

より高度な統計分析やモデリング、シミュレーションなどを可能にするための他のパッケージとのインタフェース、OA分野への適用のための文書情報処理やレポート作成機能の充実、より使い易いマンマシンインタフェースの開発などが今後の課題である。

終わりに、PLANNERの開発において、その中核であるリレーショナルデー

データベースに関して、終始、技術的な協力をしていただいた、富士通研究所、牧之内顕文氏をはじめとするRDB/V1グループの諸氏に感謝する次第である。

6. 参考文献

- [1] 牧之内, 手塚ほか : リレーショナルデータベース管理システムRDB/V1, データベース管理システム研究会資料 19-3, (1980)
- [2] 神田, 小関ほか : 関係データベースを核とした地域情報システムについて, 昭和55年度情報処理学会, 第21回全国大会, (1980)
- [3] 牧之内, 手塚ほか : リレーショナルデータベース管理システム RDB/V1, 昭和56年度情報処理学会, 第23回全国大会, (1981)
- [4] 甲田, 手塚ほか : RDB/V1におけるビューの実現方法, *ibid*
- [5] Makinouchi, A., Tezuka, M., Kitakami, H., and Adachi, S.
: The optimization strategy for query evaluation in RDB/V1,
Proc. 7th International Conference on VLDB, pp.518-529, (1981)