

解説

対話型アプリケーションにおけるリレーショナル・モデルの応用例*

宇野 栄** 宇土正浩**

1. はじめに

データ・ベースのリレーショナル・アプローチ（リレーショナル・モデル）は、高度のデータ独立性と簡明なデータ・モデルを提示し、理論的背景に支えられた新しい考え方である。この適用分野は広汎多岐にわたり、今後ますます普及していくものと考えられる。ここでは、これまであまり議論のない分野、即ち製造業における設計支援（CAD）システムと地域整備等のための計画支援システムにおける応用例をとりあげる。前者は複雑なデータ構造をいかに整理するかという表現力の観点から、後者は大規模データに対する非定型的照会の処理という観点から述べる。どちらも、日本アイ・ビー・エム・サイエンティフィック・センターで研究開発され、実験研究用として稼働中のシステムを基盤としている。

2. 設計支援システムとリレーショナル・モデル

2.1 CAD システム

設計支援（CAD）システムにおいてデータ構造の問題は重要な課題の1つである。データ構造は、コンピュータ・グラフィックスの立場から対話的処理に対する柔軟性と効率上の要請だけでなく、データ解析の立場からも十分考慮されていなければならない。すなわち、設計された対象物に対する構造計算などの解析プログラムが容易に実行できることや、設計者やプログラマにとって理解しまた使用しやすい形をしていることが必要である。これまでいろいろなモデルが提唱されそれに対する議論も行われている¹⁾が、ポイント連鎖の技法を用いたものが多く、複雑な構造になると熟

練した専門家でないで全体を把握するのが難しくなる。このような意味でリレーショナル・モデルはCADシステムのデータ構造として有力な表現法の1つである²⁾。ここではCADにおけるリレーショナル・モデルの例を事例研究をふまえて述べる。

2.2 データ・モデル

設計情報特に形状情報は、各エンティティの属性とエンティティ間の関連とから成っている。これらをデータ構造として表現したものをそれぞれ属性モデル、リンク・モデルと呼ぶ。リンクは一般にグラフとしてアークとノードで表現される。

例として図-1のグラフを考える。このリンク・モデルは表-1で表わせるであろう。次にアークに方向性を入れた有向グラフを考える。図-1で $a_1: n_2 \rightarrow n_3$, $a_2: n_3 \rightarrow n_1$, $a_3: n_2 \rightarrow n_1$ の方向を加えたものは表-2（次頁参照）となろう。ここで a) は表-1の表現に方向性を加えたもの（負符号は定義域 A の方につけてもよい）、b) はアークとその始点、終点を1行（nヶ組）

表-1 グラフの表現

A-N	Arc	Node
	a_1	n_2
	a_1	n_3
	a_2	n_3
	a_2	n_1
	a_3	n_1
	a_3	n_2

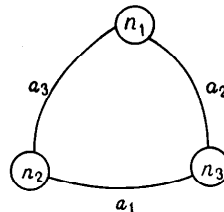


図-1 グラフの例

* Some Relational Models in Interactive Applications by Sakae UNO and Masahiro UDO (IBM Japan, Ltd., Scientific Center)

** 日本アイ・ビー・エム(株)サイエンティフィック・センター

表-2 有向グラフの表現

a)			b)			
A-N	Arc	Node	A-NN	Arc	Start Node	End Node
	a_1	n_2		a_1	n_2	n_3
	a_1	$-n_3$		a_2	n_3	n_1
	a_2	n_3		a_3	n_2	n_1
	a_2	$-n_1$				
	a_3	$-n_1$				
	a_3	n_2				

tuple) で表わしたものである。表現力は同等であるが記憶域の大きさや、a)では2つの n ケ組が対になっていなければならないというリレーション内の整合性の問題があるため、b)を標準形としこれをアーク中心型表現と呼ぶことにする。無向グラフも形式的にこの表現を用い、方向性を無視するものとする。

形状情報にはあるノードに接合するアークの順番が必要になることがある。たとえば交差点における右折左折を区別する場合である。前に述べた標準形からはこの情報は得られない。このために、想定される接合アークの数(ノードの次数)の最大が格納できるだけの列(column)をもつ表(relation)の中に、ノード名と接合アーク名を順番に入れる方法が考えられる。しかしこのような表では、例外的にアークが集中するノードのために多くの記憶域が浪費される。そこで1つの n ケ組ではノード名と接合アークの1つ及び「その次」の接合アークを表わすことにし、これをノードの次数だけ繰返す方法を採用(リレーショナル・モデルではデータ中に明示しない限り n ケ組の順番はついていない)。順番は絶対番号ではなく相対的かつ巡回的にとられている。このようにすると、どのアークからでも一巡できるし、変更による維持が簡単である。図-2に例を示す。ここでノード n_i におけるアーク a_j の「次の」アークとは、 n_i を正の方向にまわったとき a_j の次にあるものと定義する。これを表現したものが表-3である。この表現をノード中心型表現と呼ぶ。なお、アークは n_i を始点とするとき正、終点

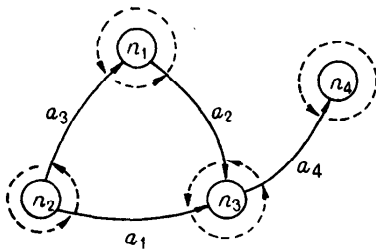


図-2 有向グラフの例

表-3 ノード中心型表現

N-AA	Node	Arc	Next Arc
	n_1	a_2	$-a_3$
	n_1	$-a_3$	a_2
	n_2	a_1	a_3
	n_2	a_3	a_1
	n_3	$-a_1$	a_4
	n_3	a_4	$-a_2$
	n_3	$-a_2$	$-a_1$
	n_4	$-a_4$	$-a_4$

とするとき負としている。

一方、属性モデルは「アーク」と「ノード」の2つの表から成る。これらの表は、エンティティ名と適用業務によって指定される任意個の定義域をもつ(形状情報には座標値を格納する場所が必須である)。

2.3 建築計画への応用

前節で述べたデータ・モデルを用いたCADシステムIDAS^{*)}を紹介する。IDASはIBM 2250映像表示装置により、形状を直接対話的に入出力するよう設計された実験研究システムである。ここでは建築計画における構造計画と平面計画についてふれる。

2.3.1 形状モデル

構造計画では、梁をアーク、柱をノードとするアーク中心型表現を用いる。吹抜にある柱などある階の平面図をみると梁のついていない柱がある。これらはリンク・モデルには現われず属性モデルにのみ存在する。

一方平面計画では各部屋の形状即ち平面グラフの面情報が重要である。このため部屋をノード、壁をアークとする双対グラフで考える。しかも部屋を構成する壁の順番がついている方が都合がよいのでノード中心型表現を用いる。これと壁自体の形状を表わすために、壁とその端点をアーク中心型で表現し組み合わせる。このようにして平面を表現したものを平面形状モデルと呼びこの例を表-4(次頁参照)に示す。これは図-3(次頁参照)に対応しており、ある部屋の外側の境界壁は正の方向にまわった順に、内側の境界壁は負の方向にまわった順に順番づけられている^{**}。図-4(次頁参照)に事務所ビルの部屋割を行った例を示す。この図にみられるように、壁で仕切られていないとき

* Integrated Designer's Activity Support system

** [F-AA]における負符号は「A-NN」で定義されている壁の方向と、平面に対する壁の順番づけの方向が異なっていることを示す。双対グラフにおけるアークの方向を、壁の方向に対して、左の部屋から右の部屋に向かうものと定義すれば、[F-AA]は、ノードに対して始点アークを正、終点アークを負とするノード中心型表現の約束に矛盾しない。

表-4 平面形状の表現

F-AA	Face	Arc	Next Arc	A-NN	Arc	Start Node	End Node
	f_0	$-a_1$	$-a_6$	a_1	n_1	n_2	
	f_0	$-a_6$	$-a_5$	a_2	n_2	n_3	
	f_0	$-a_5$	$-a_4$	a_3	n_3	n_4	
	f_0	$-a_4$	$-a_3$	a_4	n_4	n_5	
	f_0	$-a_3$	$-a_2$	a_5	n_5	n_6	
	f_0	$-a_2$	$-a_1$	a_6	n_6	n_1	
	f_1	a_1	a_2	a_7	n_6	n_7	
	f_1	a_2	$-a_7$	a_8	n_7	n_8	
	f_1	$-a_7$	a_8	a_9	n_8	n_9	
	f_1	a_8	a_1	a_{10}	n_9	n_{10}	
	f_2	a_3	a_4	a_{11}	n_9	n_{10}	
	f_2	a_4	a_5				
	f_2	a_5	a_7				
	f_2	a_7	a_8				
	f_2	$-a_8$	$-a_{10}$				
	f_2	$-a_{10}$	$-a_9$				
	f_2	$-a_9$	$-a_8$				
	f_3	a_9	a_9				
	f_3	a_9	a_{11}				
	f_3	a_{11}	$-a_{11}$				
	f_3	$-a_{11}$	a_{10}				
	f_3	a_{10}	a_8				

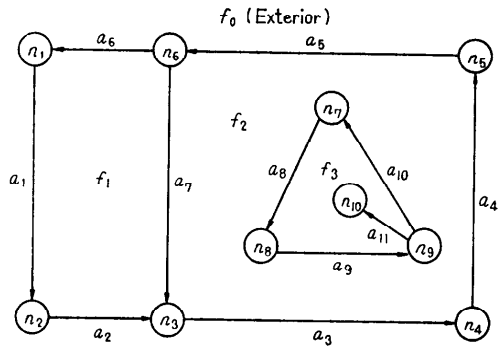


図-3 平面形状の例

でも機能上別の部屋とした方がよいことがある。この場合は、ダミーの壁を導入し属性モデルにその旨記してある。平面形状モデルの属性モデルは「部屋」、「壁」、「壁の端点」の3つの表から成る。

IDAS では、CAD システムを制御管理するシステム部と、適用業務固有の処理をするアプリケーション部を分離している。リンク・モデルはモデル操作言語

にもとづいてシステム部が管理する。また属性モデルは、エンティティ名の追加/削除と座標値についてはシステム部、その他の定義域についてはアプリケーション部が管理する。なお、モデル操作については別の報告にゆずる⁵⁾。

2.3.2 代替案モデル

設計過程の進行とともに生まれる数多くの代替案の関連もグラフとしてとらえることができる。代替案は図-5のように階層構造をしており各案がノードに対応する。アークに対応するのは案間の相連即ちその間の加工手順である。この手順をアクション・レコード(以後ARと略記)と呼ぶ。ARは問題向け言語(POL)に類似した形式をしており、映像表示装置による形状指示はARに翻訳される。これにより、タイプライタ型端末やパッチによる言語を用いた設計も容易に包含される。

アークに対応させることができる最小単位は

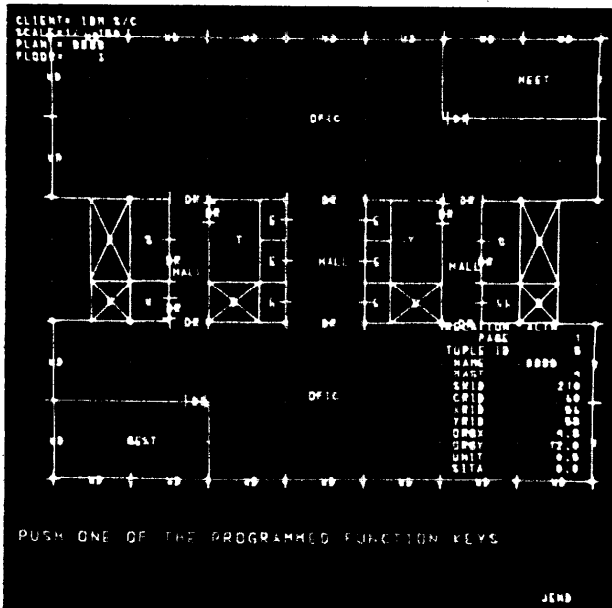


図-4 平面計画の実例

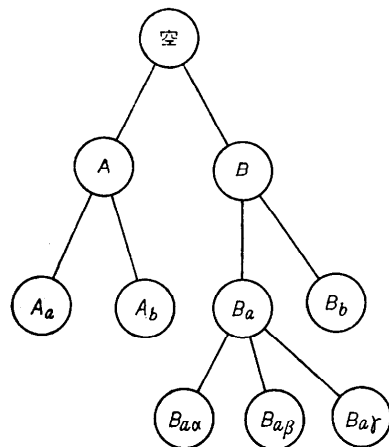


図-5 代替案の構造

1つの AR であるが、実用面からはいくつかの連続した AR をまとめるのが賢明であろう。代替案モデルにより、過去の任意時点の形状モデルを復元することができる。即ち最初のノード（設計開始時点：空集合）から指定された時点に至るアークをたどって AR を追跡してゆけばよい。しかし大規模かつ複雑な設計をしている場合には、いつも最初のノードからたどるのでは効率上の問題がある（この問題は対話型システムでは特に重要な要因である）。このため主要ノードでは、その時点での形状モデルを保存し、これを AR 追跡の起点にとれるようにした。このようにすると、データの重複の問題があるが、本来設計が試行錯誤的にいろいろな代替案を併存しながら進められるという事情を考えると、案ごとにデータを持っている方がかえって便利とも思われる。代替案（ノード）属性表には、案を構成する形状モデル（構造計画と平面計画）の表の内部名が格納される。どの案でも外部名としては「F-AA」、「A-NN」、「部屋」、「壁」…であり、これらを現在対象としている案名に従って内部名に変換するためである。これらは明らかに（第1）正規形⁶⁾をしていないが、制御管理のための表はこの方が都合がよいことが多い。

3. 要素操作型データ準言語

ここでは IDAS のデータ・モデルを物理的レベルで管理する BARTH^{*7)} について述べる。BARTH は要素操作型データ準言語を提供するリレーショナル・データ・ベース管理システムであり、自己完結した汎用モジュールである。

3.1 BARTH の設計目標

BARTH の設計目標は次の通りである。

- a. リレーショナル・データ・ベースの基本的管理と操作
- b. 対話的使用のための効率を重視
- c. 設計分野での使用が容易

このため外部仕様には、表や n ケ組の内部名などの物理的構成に依存する部分がある。したがって一般ユーザのためには、この間にもう一段インタフェースを介するのがよい（IDAS でもこのようにした）。

表は実行時に作成／削除できる。作成時に表の内部名（RID と呼ぶ）が割り当てられる。各表の表頭には特別の n ケ組が2つはいる。1つはデータ型を示し他の1つは定義域名を示す。データ型は、整数型、実数

型、文字型でありいずれも4バイトの固定長である。

一般の n ケ組は自由に追加／除去できる。追加時に n ケ組の内部名（TID と呼ぶ）が割り当てられる。TID は n ケ組のデータ・アドレスに関連しており、これを指定するとアクセスが速い。

3.2 BARTH の機能

3.2.1 表に関する操作

表に関する操作には、作成、削除、活性化、非活性化及び複写がある。

BARTH では、ユーザの要求によって表が主記憶域上に移される。これを活性化という。次に述べる n ケ組に関する操作は活性化されている表に対してのみ有効である。逆に主記憶域から表を排除することを非活性化という。このとき現在の表を補助記憶域に保管するかどうかを指定できる。保管するとは、今まで行った作業は有効であるとしてデータ・ベースを変更することであり、保管しないとは、データ・ベースは変更されず、その表は活性化されたときの状態に戻される。この機能は設計分野では重要である。即ち、設計データの変更の有効性はそれに対する評価、検討等の解析の後に決定される。したがってデータの変更はあくまで仮の変更であり、それがデータ・ベースの変更に直接つながるとかえって都合が悪い。表の複写も設計固有の問題で、代替案として形状データを保存するときに便利ように加えられた機能である。

3.2.2 n ケ組に関する操作

n ケ組に関する操作には、追加、除去、読込、変更、及び条件付検索があり、いずれも活性化されている表の RID を指定する。これらの操作の単位は n ケ組であり、一部の定義域だけに限る操作はできない。したがって、追加のときは全データが揃っているときのみなされるし、一部の定義域の値の変更は、前もって読込または条件付検索によりすべての定義域の値をとり出しておかなければならない。

条件付検索では、ある指定した定義域の値と与えられた値とを比較して、指定された条件を満たす n ケ組が得られる。条件は、=, ≠, >, ≥, <, ≤ または無条件のいずれかである。無条件ではすべての n ケ組が条件を満たすとみなされる（逐次読込）。条件付検索ではまず検索条件を BARTH に登録する。このとき比較すべき定義域名、比較値及び条件を指定する。これに対して検索識別名（SID と呼ぶ）が割り当てられる。次に SID を指定して n ケ組の検索をする。1回の操作では1つの n ケ組が得られる。一般に、条件を

* BAsic Relational Table Handler

満たす n ケ組は複数個あり、検索要求を繰り返すことにより、別の n ケ組が次々に得られる。SID の導入により、条件付検索のネストも混乱なく処理される。

条件付検索で最もよく用いられるものは、等号条件でしかも条件を満たす n ケ組は 1 つしかない場合である。このときには、効率上及び使用上の便宜のために簡略要求ができる。これは条件登録の操作を省略し、検索要求のときに比較すべき定義域名と比較値を指定するものである。これはキイ検索のときに特に有効と思われる。

4. 計画支援システムと表形式データ

4.1 計画支援システム

計画支援システムという名称はまだ一般的でない。ここでは「計画」を地域整備事業など地域にかかわる計画業務に限定する。これらの計画業務で取扱われる情報は多方面にまたがり、しばしば膨大な量になる。そのなかで時系列表、地域別表、原単位表など「表」として取扱われるものが大きな割合を占めている。というのは情報整理・情報伝達の手段として表が多く活用され、有効であるからであろう。実際、資料データとしての表にもとづき立案・計画された結果が表として整理され、他部門での立案・計画に利用されるというサイクルが計画業務のなかでそれが持っている重要性を示している。ここでは計画業務の情報の流れをこのサイクルとして捉えることにより、計画支援システムのデータ構造の表現として、リレーショナル・モデルが有効であることを事例研究にもとづいて述べる。

4.2 表形式データ

地域にかかわる計画業務に必要な情報を地域情報とよぶと、地域情報は「地域属性情報」、「地図情報」という 2 つのカテゴリーに分けられる。4.1 で述べた「表」とは前者の具体的表現である。しかし後者についても、地域の境界などの図形情報に対するリレーショナル・アプローチが有効であることは、2.2 から理解できる。ここでは前者についてだけ述べ、後者についてはその概念を述べた別の報告にゆずる⁹⁾。

地域属性情報は地域、属性、時間という 3 軸でその特徴が表現できる。すなわちある時点での地域属性情報とは、地域をエンティティとし人口、世帯数、事業所数などの地域の属性を必要とするだけ列挙したものである。これは地域名を表側とし個々の地域属性を表

頭とした 1 枚の表である。更に次の 2 点が指摘できる。すなわち、①地域は行政区分などの地域区分によって生ずる階層構造をもつ、②地域の属性の間には互いに関連するものがある、ということである。属性間の関連は一般に関連図として表現される。図-6 はその 1 例である。関連図のなかで特に関連式が定義されるものもある。図-6 では教員数=小学校教員数+中学校教員数+高校教員数がその例である。

さて、地域属性情報の特徴をこのように捉えるとき、データ構造としては、①時間軸にそって地域の属性値を要素としてもつ表、②地域間の階層構造、③地域の属性間の関連、を表現していなければならない。このなかで特に①を「表形式データ」と称し、プログラマや計画担当者が使用するデータはすべてこの表形式データとして取り扱う。表形式データの内容は表の名前、表頭名*、表側名**の組合せで参照し、参照されるものを要素とよぶ。

表側名が地域か、時系列か、その他かでそれぞれ地域別表、時系列表、一般表として区別する。

②③の問題について例を中心に述べよう。いま図-7(a)(次頁参照)にある $A1 \dots A5$ なる地域名を表側とし、表頭名 $D1, D2, D3$ (それぞれ小学校、中学校、高校教員数を意味するとする) を持つ表形式データを考えよう。このとき地域の階層構造としては図-7(b)を仮定する。地域属性の関連は $D\phi$ を教員数合計とすれば $D\phi = D1 + D2 + D3$ となる(図-7(c))。データ構造として、①を表形式データで表現したように、②の地域階層構造も表を用いて表現できる。いまの例で、図-7(b)の階層構造は表-5(次頁参照)としてあらわされる。③地域属性の関連については、現在のところユーザの責任で表形式データの表頭名を管理する

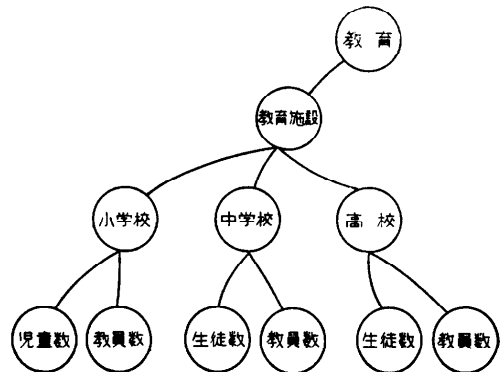
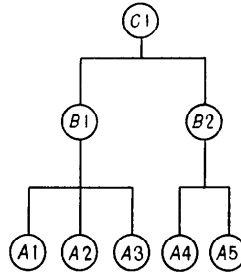


図-6 地域属性の関連図

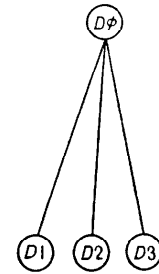
* 表頭名としては定義域名をもちいる。
** 表側名は表の第 1 列目にあり n ケ組のキイとなるものである。

TCHR	AREA	D1	D2	D3
	A1	120	60	55
	A2	80	40	40
	A3	75	30	35
	A4	110	55	50
	A5	90	35	30

(a) 表形式データ



(b) 地域の階層構造



(c) 地域属性の関連

図-7 地域属性情報

表-5 地域の階層構造の表現

地域区分	親	子	世代
	C1	B1	1
	C1	B2	1
	B1	A1	0
	B1	A2	0
	B1	A3	0
	B2	A4	0
	B2	A5	0

ことによって可能である。

結局、地域属性情報のデータ構造は表形式データを中心に考えることができ、表形式データから表形式データへの変換の補助手段として地域間の階層構造、地域属性間の関連を位置づけることができる。図-7の例をもとに変換の例を図-8に示す。

4.3 適用例

この節では4.2で述べた表形式データを用いた対話型の計画支援システム CARPS^{*)}を紹介する。これは兵庫県企画部と行った地域整備総合管理システム¹⁰⁾についての共同研究の一環として開発された実験研究システムで、IBM 3270 情報表示装置を使用している。

4.3.1 概要

CARPS は、①表形式データの対話型操作、②モデルの対話型運用管理、③アプリケーション・サブシステム、という主要モジュールより構成されており、次の3点を設計目的としている。すなわち「表形式データを中心としたデータ/モデル/アプリケーションの総合的運用支援」、「非定型的利用を可能にする汎用の対話型システム」、「定型のアプリケーションの組み込みを容易にするシステムの拡張性とユーザ・インタフェースの提供」である。エンド・ユーザである計画担当者の利用形態は一律ではなく、各担当者によって使用するデータも、加工・編集方法も、更に出力方法も異

TCHR	AREA	D1	D2	D3
	A1	120	60	55
	A2	80	40	40
	A3	75	30	35
	A4	110	55	50
	A5	90	35	30

属性間の関連による変換
 $D\phi = D1 + D2 + D3$

TCHR	AREA	Dφ
	A1	235
	A2	160
	A3	140
	A4	215
	A5	155

地域構造による変換

$$B1 = A1 + A2 + A3$$

$$B2 = A4 + A5$$

TCHR	AREA	D1	D2	D3
	B1	275	130	130
	B2	200	90	80

TCHR	AREA	Dφ
	B1	535
	B2	370

図-8 表形式データの変換

なっている。この様な利用形態を非定型的利用とよび、すでに手続化された一定の利用形態を定型的利用とよぶと、①②のモジュールは計画担当者の非定型的利用を支援し、③が定型的利用をシステム化したものである。③の例として、表形式データの蓄積・検索を行う地域資料提供システム、人口の基礎変動予測を行う地域基礎人口変動情報提供システム、教育施設の需要予測を行う生活環境施設予測情報提供システムがある。②のモデルの対話型運用管理¹¹⁾では、予測モデルを対話的に実行させ、予測結果を時系列表として出力する。その際、入力データとして時系列表を使用することも可能である。

4.3.2 表形式データの対話型操作

CARPS で使用されるデータは表形式データに統一されており予測モデル、アプリケーションの間でこのデータを媒介としたデータの相互利用ができる。これ

* Computer-Assisted Regional development Planning System

表-6 表形式データの対話型操作

1. 操作機能群

機能	説明
縮小	必要な表頭名とその列を取り出し縮小された表を作成する。
結合	2枚の表の表頭名を結合し1枚の表にまとめる。
検索	与えられた条件式を満たす行を取り出し、それらの行だけからなる表を作成する。
演算	列と定数、列と列のあいだでの四則演算を行う。
時系列表への交換	時間(年)を含む表頭を表側へ交換する。
集計	地域区分ごとの集計を行う。

2. 管理機能群

機能	説明
削除	データ・ベースにある表を削除する。
廃棄	主記憶域にある表を排除する。
格納	表をデータ・ベースに保管する。
名前変更	表の名前を変更する。
属性変更	主記憶域に作成された表(作業用の表)をデータ・ベースに保管可能とさせる。
数値変更	表の要素、表頭名を変更および表側名を追加・削除する。

ら表形式データを管理し、対話的に加工・編集する機能を、表形式データの対話型操作モジュールが持っている¹²⁾。表-6に機能一覧表を示す。計画担当者はこれらの機能を、情報表示装置上のメニューを選択する

ことによって実行できる。機能はメニュー体系にのって、操作機能群と管理機能群にわかれ、エンド・ユーザとしての計画担当者が使用する機能は、おもに前者に分類されている。検索機能の条件として=,キ,>, ≥, <, ≤の6条件を表のある列に与えることができ、演算機能が実行できる演算式は term 1=term 2 {+|-|*|/} term 3 で表現され、term 1 は演算結果をいれる表頭名であり、term 2,3 は表頭名あるいは定数(整数型、実数型)である。更に表側が地域名である地域別表については、地域区分で示される階層構造にもとづく集計を行うことができる。図-9に集計機能の結果表示画面を示す。この機能は4.2で述べた地域間の階層構造にもとづく表形式データの変換を行うものである。

CARPS においてこれらの機能が果たす役割は、①計画担当者が持つ「テーマ」を機能及びその組み合わせに分解し実行していくことによって必要な結果を得ることができる、例えば「人口密度が高い市町」を求めるには「人口」、「土地面積」をもつ表形式データを取り出し(縮小)、1枚の表にし(結合)、その表で「人口密度」を演算により求め(演算)、人口密度が基準値以上の市町を求める(検索)わけである。②計画担当者

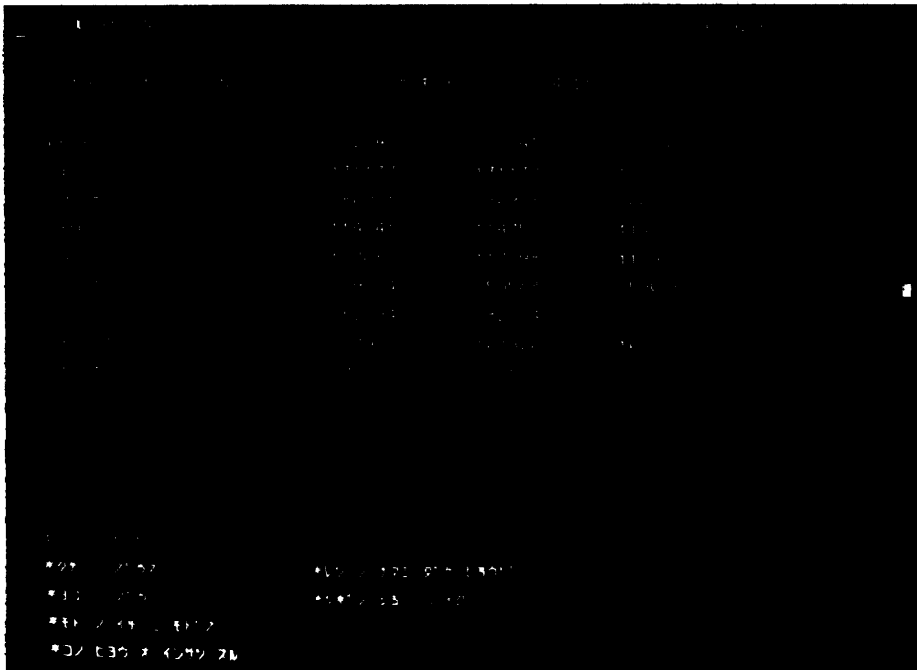


図-9 集計機能による結果表示画面

は予測モデル、アプリケーション・サブシステムを使用する際、要求に応じて表形式データを前処理、後処理することができる。

4.3.3 今後の課題

4.3.2 で「人口密度」を演算によって求めたが、このとき、 $\text{人口密度} = \text{人口} / \text{土地面積}$ なる式で新しい地域属性が定義され、属性値としての人口密度の値が、表形式データの指定された表頭名をもつ列に挿入されるわけである。しかし表形式データの表頭名は、ユーザが管理するもので、システムは挿入された表頭名が人口密度を意味することを関知していない。大規模なデータ・ベースを表形式データから生成するとき、4.2 で述べたところの地域属性についての記述データ、属性間の関連（関連図）を管理することの必要性が、CARPS 開発を通じて強く感じられた。

5. おわりに

リレーショナル・モデルをそのデータ構造として用いた IDAS, CARPS について述べてきた。これら実験研究システムの開発経験からデータ・モデルの単純さ、即ち表という単純な形でデータを取り扱っているため、プログラマやエンド・ユーザにとって非常に理解しやすくプログラムの作成、虫取り、保守が容易に行えたこと、またデータの独立性によるアプリケーション開発の容易さ、システム拡張の容易さが検証され、リレーショナル・モデルの有効性が示されたと考える。

参 考 文 献

- 1) 古川康一：コンピュータ・グラフィクスにおけるデータ構造の問題，情報処理，Vol. 11, No. 9, pp. 523~532 (1970).
- 2) T. Kawai: Comparison of Data Structures for

CAD Systems, p. 43, IBM Scientific Center, GE18-1824, Tokyo (1975).

- 3) 松家, 川井, 宇野: 設計活動援助統合化システム (IDAS) の概念, p. 35, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1813, 東京 (1975).
- 4) 松家, 川井, 宇野: IDAS-A (建築計画用) の概念, p. 51, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1818, 東京 (1975).
- 5) 川井敏弘: IDAS におけるモデル管理, p. 39, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1817, 東京 (1975).
- 6) E. F. Codd: A Relational Model of Data for Large Shared Data Banks, Comm. ACM, Vol. 13, No. 6, pp. 377~387 (1970).
- 7) S. Uno: Basic Relational Table Handler (BARTH), p. 31, IBM Scientific Center, GE 18-1816, Tokyo (1975).
- 8) 宇土正浩: 地図情報の表現と処理, p. 39, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1814, 東京 (1975).
- 9) 大河内, 宇土: 対話型計画策定支援システム (CARPS), p. 35, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1837, 東京 (1976).
- 10) 兵庫県企画部, 日本アイ・ビー・エム・サイエンティフィック・センター: 地域整備総合管理システム共同研究報告書一, p. 169, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1840, 東京 (1976).
- 11) 大河内正明: 対話型計画策定支援システム (CARPS) におけるモデルの対話型運用管理, p. 46, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1839, 東京 (1976).
- 12) 宇土正浩: 対話型計画策定支援システム (CARPS) における表形式データの対話型操作, p. 45, 日本アイ・ビー・エム・サイエンティフィック・センター, N: GE 18-1838, 東京 (1976).

(昭和 51 年 5 月 28 日受付)

(昭和 51 年 7 月 22 日再受付)