

ユーザ指向のデータベース管理システム COOD——設計とデータベース用言語†

松田孝子^{††} 田中信行^{†††}

COOD は、PRF (Private Researcher File) あるいは UDL (User Data Library) レベルの学術情報データベースの作成・利用に適する汎用のデータベース管理システムとして設計、開発した。その基本的設計思想は、“ユーザ指向”の立場で、ユーザのデータベース操作全般にわたっての言語機能を提供することである。従来の DBMS は、その使用者にオペレーティングシステムの広範な機能の理解を強いることが多いが、本システムは、計算機システムからユーザを解放し、COOD のみを習得するだけで容易にデータベースを作成、利用することを可能にした。

COOD データベース用言語は、データベース定義用の DDL と FDL、データ操作用の CML と DML およびデータベース操作用のコマンドからなる。データモデルは表形式の単純なものを採用し、数値、文字データを扱う。データ操作 (格納・追加・変更・削除) を自立方式および FORTRAN 親言語方式の言語で行うことを可能にした。また、データベース操作 (生成、追加、変更、表示、削除、再構成) を Q-A 方式に行うようにし、データベースの作成・管理・利用をすべてユーザの手許の端末から行えるようにした。

本システムを ACOS-6 上に実現し、東北大学大型計算機センターのプログラムライブラリに登録し、現在、種々の分野の研究情報のデータベース作成に利用され、その有用性を実証している。

1. 緒言

近年、学術研究情報のより有効・広範な活用を目指して、データベースを作成し利用する機運が高まっている^{1),2)}。それに伴い、これらのユーザとデータベース管理システム (Database Management System; DBMS) の適合性が問題となってきた。

従来、DBMS のタイプを分類するためにデータモデルが用いられ、その主なものとして階層モデル、ネットワークモデルおよびリレーショナルモデルがあげられる^{3),4)}。東北大学大型計算機センターを例にとれば、CODASYL のデータベース仕様^{5),6)}に影響を与えた IDS⁷⁾ とその改版である ADBS⁸⁾ および擬似的なリレーショナルモデルの INQ⁹⁾ が利用可能である。しかし、コンピュータを専門外とするユーザ自身がこれらを使用してデータベースを作成した例はまだみられない。その第一の理由は、データベースを作成し管理するために高度な技術的能力が要求されることであり、第二は、操作環境、特にデータベースを作成する操作がバッチ処理に偏向していることである。

ところで、学術情報データベースにおいては、その

データの性格上、データベースを管理するのはコンピュータには非専門の各研究分野の研究者である場合が多く、それらの各分野ではデータベース要員としてのコンピュータスペシャリストを個々に確保することが困難である。また、学術情報の高次性、多様性から、そのデータベースの形態は、PRF (Private Researcher File) UDL (User Data Library), CDL (Center Data Library) の3つに分けられるが¹⁾、これらの形態は定常的に存在すると考えられるので、その各レベル、特に PRF や UDL に適合する DBMS が強く要求される。

以上を考慮し、本研究においては、各分野の研究者が容易にデータベースを作成しかつ管理、利用できるような DBMS として、会話型データベース管理システム COOD (Con conversationally Organized Database Management System) を開発したので¹⁰⁾⁻¹³⁾、ここにその設計とデータベース用言語を報告する。

2. COOD の設計

2.1 ユーザ指向の DBMS に要求される条件

学術情報のデータベース化が各研究分野で行われ、システム化されるためには、多様な情報を自由にかつ容易に表現でき、また容易かつ柔軟に操作できることが要求される。

学術文献情報の検索システムについては、我が国でも TOOL-IR¹⁴⁾ や JICST¹⁵⁾ などによる CDL レベルの実用システムがあるが、学術情報全般についてみ

† User-oriented Database Management System COOD——its Design and Database Languages by TAKAKO MATSUDA (Computer Center, Tohoku University) and NOBUYUKI TANAKA (Department of Chemistry, Faculty of Science, Tohoku University).

†† 東北大学大型計算機センター

††† 東北大学理学部化学科

れば、データに依存した検索システムとして個々に実現されていることが多い。これらの学術情報の蓄積をさらに促進し流通しうるデータベースにシステム化するためには、その目的に合う DBMS が必要である。

まず、データモデルについては Codd のリレーショナルモデルが^{16), 17)}、単純さ、柔軟さの点で有力視される。Thalmann は化学データを例に3つのデータモデルを比較考察の結果、リレーショナルモデルが適することを示している¹⁸⁾。しかし、これを PRF や UDL レベルの DBMS に採用した場合、データ管理側の意図とデータ利用側の期待の間に生じる不整合を解決し、データ誤用を防ぐ手段が必要である。

データベース用言語については、SEQUEL¹⁹⁾、QB E²⁰⁾、INGRES²¹⁾、ヤチマタ²²⁾ などデータ操作を主体にしたものにユーザ指向の配慮がみられるが、データベースを作成・管理する操作全般についてユーザ指向の立場で設計されたものはみられない。それは、従来の DBMS がデータベース管理者を暗黙にコンピュータスペシャリストと想定していたことに起因する。

ここで、学術情報用の DBMS が満たす条件についてまとめると、従来からあげられているデータ独立、機密保護、データ保全、データ操作機能の達成に加えて、つぎの点があげられる。

(1) システムの習得が容易で、かつその知識だけ持っていてデータベースの作成・利用ができること。

(2) 少なくとも数値情報と文字情報が扱えること。また、図形情報の取扱いができることが望ましい。

(3) 容易に実際の情報をモデル化でき、かつデータ管理側の意図がデータ利用側へ正しく伝達できること。

(4) 必要なときに容易にデータの格納、追加、変更および検索ができ、データ操作に柔軟性があること。

(5) データベースの定義、変更、追加、分割、統合などのデータベース操作が容易なこと。

(6) 過去に蓄積したソフトウェアと結合でき、そのアプリケーションが計算機システムの中で軽快に動くこと。

2.2 COOD の設計基準

COOD はつぎの設計基準にそって開発した。

(1) 習得容易性を実現するために、ユーザインタフェースであるデータベース用言語にデータベースに関連する諸操作をできるだけ反映させ、かつその仕様

ができるだけ対称性を持ちコンパクトになるように設計する。また、言語は極端に記号化せず、多少冗長であっても自然語に近い言葉とする。

(2) 数値および文字データを扱う。数値データとしては、FORTRAN で扱うデータ型に加えて、有効桁数の保存、連続量の表現、空値の表現ができるようにする。文字データは語、句単位の取扱いができるようにする。

(3) データモデルは、表形式の単純なものとし、表の中に列項目を持てるようにする。また、項目などに説明を付加して定義できるようにする。

(4) 共同利用の計算機で即時性を保障するために TSS の下で稼動するシステムとする。また、研究室に端末が設置され計算機利用がオンライン指向に移行している利用環境を反映して、オンラインでデータベースを作成したり利用したりできるようにする。端末としてはユーザの現況を反映してキャラクタ型のものをまず対象とする。

(5) 自立型のデータ操作機能および FORTRAN を親言語とするデータ操作機能を用意する。FORTRAN はデータ操作の親言語として最適とはいえないが、長年にわたり蓄積されてきた計算処理プログラムと結合できるという点で、学術情報用の DBMS の親言語としての条件を満たしている。

(6) データベースの定義・生成のみならず、定義の追加・変更、データベースの再構成、移行などを容易に行えるようにする。このような操作は、研究の進展につれて初期には予測できない問題が発生する場合に、データベースをその変化に柔軟に対応させるためおよびシステムの変更に対処するための機能として重要である。

(7) データの機密保護および同時利用の制御は表単位に行う。また、特別なアクセス方式を用いず、計算機システムが一般に提供する標準的なアクセス法を使用する。これによって、DBMS をコンパクトに作成でき、計算機システムに対する負荷を軽減できる。

(8) 機種間の変換性を高めるために、COOD システムを FORTRAN で記述する。FORTRAN はシステム記述用言語として最適な言語ではないが、オンラインで利用可能な言語として汎用性がある。

(9) DBMS の開発と併行して、それをを用いたデータベースを試作し、その使用経験を DBMS に反映させ設計の改良を行う。このことによって、真にユーザを指向し実際に結びついたシステムの開発を行うこ

```

DDL;
DATABASE REFEEK : Researcher's File of Etsuko Kikkawa;
TABLE LITS : Literatures on Metal Complex Reactions;
NO          (I4) UNIQUE : Serial number;
AUTHOR(5)  (A20) : Author(s);
TITLE(4)   (A70) : Title;
JNL        (A70) : Journal;
YEAR       (I4)  : Year of Publication;
KEY(5)     (A50) : Keyword(s);
TABLE REMECS : Reactions of Metal Complexes in Solutions;
NO          (I3) UNIQUE : Serial number;
DATE       (I6)      : Date;
INVESTOR   (A30)     : Investigator;
TYPSTUD    (A40)     : Type of study;
REACSYS(2) (A70)     : Reaction system;

K          (J10)      : Equilibrium constant;
NOTE(3)   (A70)     : Notes;
TABLE GINT : Gauss Integral;
N          (I2)      : Order of Legendre polynomials;
P          (I2)      : Precision of Data;
X(10)     (J30)     : Abscissas +x(i), -x(i);
W(10)     (J30)     : Weight Factors w(i);
C(2)      (A70)     : Comment of Data;
END-DDL;

```

図 1 データ定義例 (データベース REFEEK)
Fig. 1 An example of the data description for the database REFEEK.

とができる。

3. COOD データベース用言語とその機能

3.1 データベース用言語

COOD はその開発目的からユーザ用言語に重点をおいた。COOD データベース用言語をつぎのものとした。

- DDL (Data Description Language): データモデルを記述するための言語
- FDL (File Description Language): データベースの容量や共用範囲を記述する言語
- CML (Conversational Data Manipulation Language): 自立型のデータ操作言語で、会話式にデータ格納・追加・変更・削除・検索を行う機能を提供する。
- DML (Data Manipulation Language): FORTRAN を親言語とするデータ操作言語で、プログラムの中からデータの格納・追加・変更・検索を行う機能

```

FDL;
DATABASE REFEEK;
TABLE LITS; MAX 100;
PERMISSION WRITE/user3,user2/
TABLE REMECS; MAX 30;
PERMISSION WRITE/user2/,READ/user3/
TABLE GINT; MAX 50;
END-FDL;

```

図 2 ファイル定義例 (データベース REFEEK)
Fig. 2 An example of the file description for the database REFEEK.

を提供する。

- Database manipulation commands: データベースの生成・追加・変更・削除・出力などの諸操作を行うコマンドで、会話式にユーザアクションを誘導する。

3.2 データベースの定義と生成機能

COOD データベースをテーブルの集まり、テーブルをアイテムの集まり、アイテムを単一アイテムか列アイテムとした。

データベースの定義は、DDL によるデータ定義 (図 1) と FDL によるファイル定義 (図 2) で行う。DDL では、データベース、テーブルおよびアイテムの名前、アイテムの寸法、書式、値の一意性を定義する。アイテムのデータ型は書式から解釈している。また、名前は 8

文字以内の英数字としたので多くの名前が略号化されることとなり、そのデータ操作時に間違いを生じやすい。これを防ぐために名前の説明を定義できるようにし、データ操作時に説明の出力ができるようにした。書式はほぼ FORTRAN に準じているが、有効桁数を保存する J 変換を追加した。また、列アイテムの寸法に "RANGE" を加え、アイテムの定義を

```
XR (RANGE) (E10.3);
```

のように行い、上下限值だけを格納してその区間に含まれる連続量を表現できるようにした。

FDL では、格納データ件数、共用するユーザと許可アクセスを定義する。物理的なファイルスペースの計算は COOD システムの中で行うようにした。これはオペレーティングシステム固有のファイル量の単位

```
SYSTEM ?COOD
```

```
COOD-PROCESS ... ?DDL
SOURCE FILE ?REFEKDDL
```

```
COOD-PROCESS ... ?FDL
SOURCE FILE ?REFEKFDL
```

```
COOD-PROCESS ... ?DEC
DATABASE NAME ?REFEK
DATABASE FILE CREATED.
```

```
COOD-PROCESS ... ?
```

図 3 データベース生成例 (データベース REFEEK)
下線部分がユーザ入力

Fig. 3 An example of the database generation for the database REFEEK (Underlined characters are typed by the user).

表 1 CML の主なコマンド
Table 1 Principal CML commands.

<code>.USE [<umc>/<database name>/<table name>[=<tname>] [({<item name>[=<iname>]},{...})],...; ALL-ITEMS</code>
<code>.SELECT[*n] <tname>,... [({<iname>},{...})] ALL-ITEMS [TO <tname> [TERMINAL-]<file name>] [WHEN(<condition>)];</code>
<code>.ASK[*n] <tname>,... [({<iname>},{...})] ALL-ITEMS [WHEN(<condition>)];</code>
<code>.STORE[*n] {NEW} <tname> [({<iname>},{...})] {OLD} ALL-ITEMS [FROM <file name>];</code>
<code>.CHANGE[*n] <tname> [({<iname>},{...})] ALL-ITEMS [FROM <file name>] [WHEN(<condition>)];</code>

からユーザを解放するために行ったものである。共用範囲の指定はデータベース全体あるいはテーブルごとに設定でき、ユーザを限定することもできる。

図1と図2で定義したデータベース²³⁾を生成するには、COOD システムを図3のように操作する。

3.3 データ操作機能

特別に養成されたデータベース要員を持たずにデータベースを作成するには、データの蓄積が容易でなければならない。COOD では、つぎの3つの方法でデータ格納・追加・変更を行うことができるようにし

た。

(1) CML を用いる会話形式の応答によって端末から項目値をタイプインする。この場合、入力促進メッセージとして項目名あるいはその説明を用いる。

(2) あらかじめテキストエディタなどで入力データを順編成ファイルに格納しておき、CML を用いてデータベースへ移送する。

(3) FORTRAN-DML プログラムの処理結果をデータベースへ格納する。

表1に CML の主なコマンドを示す。USE はユーザ視野を決定するためのもので、名前によってデータベースとユーザアプリケーションの結合を行う。たとえば、図1に示したテー

ブル LITS について3つのアイテムに視野を限定して
USE REFEK/LITS (TITLE, AUTHOR, YEAR);
SELECT LITS WHEN (YEAR >= 1979);
で検索すると、表題、著者名、発行年の順で出力する。USE において名前の代替名を宣言できるようにし、重複した名前の識別や簡便な入力のために用いることができるようにした。

STORE と CHANGE はデータの格納・追加・変更用いる。図4は方法(1)によるデータ格納例である。入力はアイテム単位(列アイテムの場合は要素単

```
COOD-PROCESS ... ?CML
?USE REFEK/LITS;
EXPLAIN ITEMS OF LITS, YES OR NO ?NO
?STORE NEW LITS;
DISPLAY, NAME(N) OR EXPLANATION(E) ?N

NO
=1
AUTHOR
='Kimura M.', 'Shirai J.'
=7
TITLE
='Reaction of ethylenediaminetetraacetate with aquated chromium(III)'
='ions in mixed solvents of water with methanol and with ethanol'
=7
JNL
='J. Inorg. Nucl. Chem., vol.40, p.1085'
YEAR
=1978
KEY
='Kinetics, Cr(III); EDTA, Solvent, Fe'

NO
=2

NO
=7
***      2 DATA STORED.
?
```

図 4 データ格納例 (端末入力の場合)

Fig. 4 An example of the data storage from a terminal.

```

COOD-PROCESS ... ?CML
?USE user4/REFAY/KIPERIS=AY;
EXPLAIN ITEMS OF KIPERIS, YES OR NO ?→
?USE user5/REFMS/REMECS=MS;
EXPLAIN ITEMS OF REMECS, YES OR NO ?→
?USE user1/REFEK/REMECS=EK;
EXPLAIN ITEMS OF REMECS, YES OR NO ?→
?USE user6/REFTY/DICISIS=TY;
EXPLAIN ITEMS OF DICISIS, YES OR NO ?→
?SELECT AY,MS,TY,EK(NO)
MORE?WHEN((CONHYD>=1.26E-5 & CONHYD<=2E-5)
, (PH>=4.7 & PH<=4.9));
MORE?
*** END OF TABLE
*** ON DATABASE user4/REFAY /KIPERIS
*** END OF TABLE
*** ON DATABASE user5/REFMS /REMECS
*** END OF TABLE
*** ON DATABASE user6/REFTY /DICISIS
*** END OF TABLE
*** ON DATABASE user1/REFEK /REMECS

*** 5 DATA FOUND.
OUTPUT DATA, YES OR NO ?Y

*TABLE KIPERIS IN REFAY
DISPLAY, NAME(N) OR EXPLANATION(E) ?→

NO : 1
NO : 2
NO : 3

*TABLE REMECS IN REFMS
DISPLAY, NAME(N) OR EXPLANATION(E) ?→

NO : 1

*TABLE REMECS IN REFEK
DISPLAY, NAME(N) OR EXPLANATION(E) ?→

NO : 7

?SELECT*1 AY WHEN(NO=3);
DISPLAY, NAME(N) OR EXPLANATION(E) ?→

NO : 3
DATE : 790206
INVESTOR : Tadatsugu Yoshikuni
TYPSTUD : Kinetic parameters of electrode reaction
METHOD : TAST Polarography
TEMP : 25.0

?USE REFEK/GINT;
EXPLAIN ITEMS OF GINT, YES OR NO ?→
?SELECT*1 GINT(N,X,W) WHEN(N>=7 & P>=8);
DISPLAY, NAME(N) OR EXPLANATION(E) ?E

Order of Legendre polynomials: 7
Abcissas +x(i), -x(i):
0 0.405845151377397
0.741531185599394 0.949107912342759
Weight Factors w(i):
0.417959183673469 0.381830050505119
0.279705391489277 0.129484966168870
?

```

図5 異なるデータベースの複数テーブルの検索および説明を付した数値データの検索表示例

Fig. 5 Examples of the data retrieval of several tables of different databases and the numerical data output with its explanation.

位)で行う。字句単位の訂正は、たとえば、
SELECT REMECS (REACSYS) TO F1

WHEN (NO>10 & NO<=20);

で訂正部分のみを順編成ファイルF1に積下し、テキストエディタで訂正後、つぎのコマンドで積込む。

CHANGE REMECS (REACSYS) FROM F1

WHEN (NO>10 & NO<=20);

SELECT と ASK はデータの検索、種類の表示に用いる。検索結果の出力は、端末、順編成ファイルあるいはほかのテーブル（異なるデータベースのものでもよい）への追加のいずれもが可能である。ファイルの形式に端末出力形式と積下しファイル形式を設けた。後者のファイルは STORE, CHANGE の入力ファイルとして逆に指定可能にした。テーブル間の移送は、たとえば 1970 年以前の文献をテーブル OLDLITS に移す（追加）場合は、つぎのようにする。

SELECT LITS TO OLDLITS

WHEN (YEAR<=1970);

この方法を用いてテーブルの統合・分割が可能である。

CML の検索条件は、演算子=, ^=(not equal to), <, <=, >, >=, &, ,(or), ^ (not) および括弧を用いる一般の論理式とした。また、部分文字列関数 LEFT, RIGHT, PART, 絶対値 ABS, 最大・最小値 MAX, MIN を設けた。列アイテムの比較は条件を満たす要素が存在すればよいものとし、RANGE 属性を持つアイテムの比較は、たとえば XR に下限値 3.1 と上限値 8.7 を持つデータは、WHEN (XR=6.1) や WHEN (XR<5) の条件を満足する。英字については端末の実情および検索キーの入力を考慮し、大文字と小文字を比較時だけ同一視するようにしている。

図5に検索例を示す。前半では4つの異なるデータベースの中のテーブルを同時に検索し、条件を満たすデータの NO を出力している。NO は4つのテーブルに共通に含まれている必要があるが、条件に現われるアイテムは必ずしも含まれる必要はない。後半はアイテムの説明を付して数値データを検索表示した例である。図6に FORT-RAN-DML プログラムを示す。ピリオドではじまるのが DML ステートメントで、そのほとんどが CML でも共通に使用できる。

3.4 データベース操作機能

データベースの生成（図3）のほかに、つぎのよう

```

SUBROUTINE GAUSS(FC,A,B,N,S,IND)
  .USE user1/REFEK/GINT(N=IODR,X,W);
  .OPEN GINT;
  IND=0; S=0.0
1  .FIND GINT;
  .IF END(GINT), GOTO 9;
  .GET IODR;
  IF(IODR.NE.N) GOTO 1
  .GET X,W;
  NN=(N+1)/2
  E=(B+A)/2.0; F=(B-A)/2.0
  DO 10 I=1,NN
  IF(X(I).EQ.0.0) GOTO 11
  G=F*X(I)
  S=S+W(I)*(FC(E+G)+FC(E-G))
  GOTO 10
11 S=S+W(I)*FC(E)
10 CONTINUE
  S=F*S
  GOTO 90
  9 IND=1
  90 .CLOSE GINT;
  RETURN
END

```

図 6 FORTRAN-DML プログラム例 (テーブル GINT を用いるガウス積分)
Fig. 6 An example FORTRAN-DML program using the table GINT (Gaussian integral).

```

COOD-PROCESS ... ?DDL
SOURCE FILE ?LDDL

COOD-PROCESS ... ?FDL
SOURCE FILE ?LFDL

COOD-PROCESS ... ?DFC
DATABASE NAME ?REFEK/NEWLITS
DATABASE FILE CREATED.

COOD-PROCESS ... ?CML
?USE REFEK/LITS,NEWLITS;
EXPLAIN ITEMS OF LITS, YES OR NO ?→
EXPLAIN ITEMS OF NEWLITS, YES OR NO ?→
?SELECT *ALL LITS TO NEWLITS;
*** END OF TABLE
*** ON DATABASE /REFEK /LITS
?→

COOD-PROCESS ... ?SVR
?RELEASE REFEK;
TABLE OR ALL ?LITS
ERASE CONTENTS ... YES OR NO ?→
TABLE ?→

```

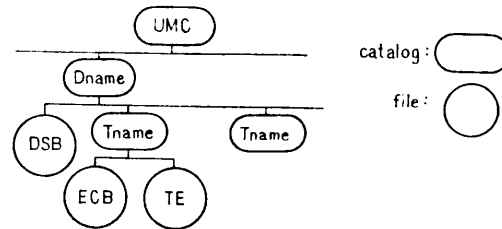
図 7 データベース REFEK へのテーブル追加, テーブル間データ転送およびテーブル削除の例

Fig. 7 An example of inserting a table into the database REFEK, data transfer between two tables, and table release.

なデータベース操作を会話的に行う機能を設けた。

- (1) 名前やその説明の変更, 共用範囲の変更
- (2) テーブルの追加, 消去, 削除
- (3) データベースの定義 (データ定義, 容量, 共用範囲など) の表示
- (4) データベースの抹消

* 文献 13) でそのソースプログラムを公開している。



UMC : User master catalog
Dname : Database name
Tname : Table name
DSB : Database Specification Block (DDL and FDL stored)
ECB : Table Element Control Block (Data link and Unique stored)
TE : Value of Table Element

図 8 COOD データベースファイルの構造

Fig. 8 The structure of the COOD database file.

これらの操作はシステムとの質問・応答で行うようにした。図 7 はその一例で、データベースへ新しいテーブルの挿入, 新旧テーブル間のデータ転送そして古いテーブルの削除を行ったものである。追加するテーブルの DDL や FDL はつぎの句ではじめる。

DDL; INSERT DATABASE REFEK;

データ量やアイテムの書式の変更, アイテムの削除・追加などデータベースの定義の変更も同様の操作で行うことができる。

4. システムの実現

COOD システムは ACOS-6 上にインプリメントした¹³⁾。システムプログラムはすべて FORTRAN で記述しており, 約 6,000 ステップである*。COOD データベースは ACOS-6 ファイルシステムを用い図 8 のように実現している。木構造のカタログ・ファイルの生成, 削除, 変更などは FORTRAN の中から TSS のコマンドを実行するための基本外部サブルーチンを用いており, ファイルの動的な参照や一時ファイルの確保などにも同様の基本外部サブルーチンを用いている。空値として, 整数型 (40000000000000), 実数型と倍精度実数型 (377000.....000), 文字型 (空白) を用いている。COOD の延べ実行メモリサイズは 60 kW 程度, DML の実行時ルーチン DME は 3.5kW 程度である。

5. 結 言

COOD は初版を完成した 54 年春から平行してデータベースの試作に利用してきた。操作を計算機の知識を持たない人に依頼したが, 短期間に COOD を習得

し作業を開始することができた。また、COOD の改版やデータベースの変更によりデータベースの再構成を数回行い、この手順の容易なことを実証した。

本システムは 54 年秋に東北大学大型計算機センターのプログラムライブラリとして公開し、理工学、医学、文学、経済学などの研究者に利用されている^{13), 23), 24)}。

本研究は以上のように初期の目的をほぼ達成したが、今後より広範なデータへの適用のために検討を進めていく予定である。

謝辞 本研究を行うに当りご指導、協力をいただいた東北大学工学部木村正行教授と研究室の方々、東北大学理学部無機化学研究室の方々および東北大学大型計算機センター高橋理助教授と研究開発室の方々に感謝する。なお、研究費の一部は文部省科学研究費補助金試験研究(2)「オンラインデータ蓄積を目的としたデータベース用言語の開発とデータベースの試作」(389010) および同特定研究(1)「トレース・キャラクター化に関する情報の処理、収録、管理、流通に関する研究」(411701) より支弁されたものである。

参 考 文 献

- 1) 学術情報システム計画, 文部省科学研究費による特定研究「情報システムの形成過程と学術情報の組織化」総括班報告 17 (1979).
- 2) 情報システムの形成過程と学術情報の組織化総合報告, 文部省科学研究費による特定研究「情報システムの形成過程と学術情報の組織化」総括班報告 16 (1979).
- 3) Date, C. J.: An Introduction to Database System (Second Edition), Addison-Wesley (1977).
- 4) 西野博二: データベース・システムの研究開発動向, 情報処理, Vol. 17, No. 10 (1976).
- 5) 植村俊亮: CODASYL 方式のデータベース・システム, 情報処理, Vol. 17, No. 10 (1976).
- 6) CODASYL Data Description Language, JOD, June 1973. CODASYL データベース用データ記述言語, 情報処理学会データベース言語研究委員会訳 (1977).
- 7) 統合データベース (IDS) 関係のマニュアル (FFE 02, FFE 03, FEG 01, FEG 02), 日本電気.
- 8) ADBS 関係のマニュアル (FFH 01, FFH 03, FFH 04, FFH 05), 日本電気.
- 9) INQ 関係のマニュアル (FFK 01, FFK 02, FFK 04, FFK 03), 日本電気.

- 10) 松田孝子, 田中信行: 研究者用オンラインデータベースシステム COOD の言語構成, 情報処理学会第 20 回全国大会論文集, p. 763 (1979).
- 11) 会話型データベース管理システム COOD 文法説明書 (第 2.0 版), 文部省科学研究費補助金試験研究(2)「オンラインデータ蓄積を目的としたデータベース用言語の開発とデータベースの試作」(1979).
- 12) 会話型データベース管理システム COOD 使用説明書 (第 2.0 版), 文部省科学研究費補助金試験研究(2)「オンラインデータ蓄積を目的としたデータベース用言語の開発とデータベースの試作」(1979).
- 13) 文部省科学研究費による試験研究(2)「オンラインデータ蓄積を目的としたデータベース用言語の開発とデータベースの試作」研究成果報告書 (1980).
- 14) 根岸正光, 山本毅雄: オンライン文献情報システム・TOOL-IR におけるマン・マシン・インターフェース, 情報処理, Vol. 17, No. 5 (1976).
- 15) 長谷川昇: 日本科学技術情報センターの文献検索システム, 情報処理学会データベース研究会資料 DB 28-3 (1976).
- 16) Codd, E. F.: A relational model of data for large shared data banks, Comm. ACM, Vol. 13, No. 6 (1970).
- 17) 穂鷹良介, 渋谷政昭: データベースの関係形式, 情報処理, Vol. 17, No. 10 (1976).
- 18) Thalman, N. and Thalman, D.: A Problem-Oriented Analysis of Database Models, J. Chem. Inf. Comput. Sci., Vol. 19, No. 2 (1979).
- 19) Astrahan, M. M. et al.: System R: A Relational Approach to Data Base Management, ACM TODS, Vol. 1, No. 2 (1976).
- 20) Zloof, M. M.: Query by Example, Proc. NCC 44 (1975).
- 21) Held, G. D., Stonebraker, M. R. and Wong E.: INGRES-A relational data base system, Proc. NCC 44 (1975).
- 22) 藤崎哲之助, 間下浩之, 諸橋正幸, 渋谷政昭, 鷹尾洋一: データベース照会システム「ヤチマタ」と名詞句データ模型, 情報処理学会論文誌, Vol. 20, No. 1 (1979).
- 23) 吉川悦子: COOD を利用する研究者ファイルの作成, 東北大学大型計算機センター広報, SENAC, Vol. 13, No. 1 (1980).
- 24) 文部省科学研究費特定研究(1)「トレース・キャラクター化」研究報告書(1980).

(昭和 55 年 3 月 4 日受付)

(昭和 55 年 5 月 15 日採録)