

対象世界のモデルを利用した

データベース検索システム

泉田 義男 星合 忠 石川 博 吉野 利明 牧之内 顕文
(富 士 通 研 究 所)

1. はじめに

我々は現在データベースシステムの日本語インタフェースを開発している。本システムは対象分野の知識を表現した世界モデルを中心としたシステムであり、本稿では世界モデルを利用した文解析、文脈処理ならびにデータベースの問い合わせ言語の生成などについて報告する。

文解析において分野依存の意味を利用するパーサはあるが[1]、構文解析のルールと分野の知識が分離されず、対象分野からの独立性がなく移行性に乏しい。また TEAM [2] のように移行性を高めるために対象分野の知識を独立させたものもあるが、その知識がデータベースの構造を意識したものとなっている。我々は世界モデルを対象分野に対する知識と問い合わせに用いられる言語に関する知識を統合したものとして実現し、これを利用して解析を行うモデル指向パーサを開発している。このパーサは構文解析と意味解析とを同時に行うが、意味解析を重視し、日常会話に近い文法的な誤りを含んだ問い合わせ文でも理解できることが特徴である。またパーサの解析ルールはパケットを単位とする部分文法の集合として記述されているのでクラスの拡張性に富んでいる。

問い合わせコマンドの生成、変換においては、その手続きをルールセットというプロダクションルールのオブジェクトによって記述することによって対象とするデータベースシステムの変更に対しても柔軟に対処できるようにしている。

この種のシステムで最も重要なのは分野の移行性であり、これを支援するためにモデルエディタを用意した。モデルエディタは世界モデルを分野の知識と言語の知識の融合したものとして編集できるようにしたエディタであり、ユーザの能力に合せたインタフェース

を用意している。

以下ではまずシステム全体の構成と世界モデルの説明を行い、次に世界モデルを用いた文解析と文脈処理について述べる。さらにコマンドの生成および変換の方式とモデルエディタの概要について述べる。

2. システム構成

本システムのシステム構成を図1に示す。インタラクタはユーザとの会話を行うためのモジュールである。ユーザが作成した質問文は、インタラクタを経由してまず形態素解析に渡され、単語辞書を使って単語分割を行い、その結果を単語のリストとして作成する。構文・意味解析では、構文解析を行うと同時に、世界モデルで表現された対象分野の知識を使って意味解析を行い解析木を作成する。世界モデルはモデリングシステム REALM [3] で管理されている。文脈処理では、前文との関係を世界モデルを利用しながら解決して、今回の入力文に対する意味構造を作成する。言い替え生成では、この意味構造から日本語の言い替えを作成してユーザにシステムが理解した内容を表示する。コマンド生成では、REALM が持っているコマンド生成の知識を使って意味構造から REALM のコマンドを生成する。データ検索では、REALM のコマンドをコマンド変換の知識を使ってデータベースシステムの問い合わせ言語に変換し、これを実行する。応答生成では実行結果の表示の確認後、結果をディスプレイ画面上に表示する。

システムにはこの他にモデルエディタ WED (World Model Editor) がある。WED は世界モデルおよび単語辞書の編集を行うツールであり、単語の登録、知識の入力・編集に使用でき、問い合わせの実行中に未登録語があったり、知識が不足していた場合などにも利用

できる。

システムの入出力はすべてウィンドウシステムを介して行なわれ、マンマシンインタフェースの向上をはかっている。

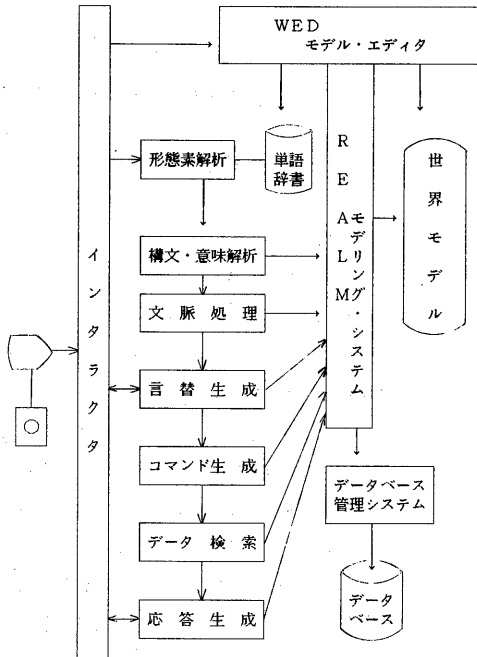


図1 システム構成図

3. 世界モデル

世界モデルはシステムがユーザに対して提供している統合化されたインタフェースであり、これを操作することによってユーザの所望する効果が得られるようにするのが我々の目的である。世界モデルには以下に示す各種の知識が格納されている。

(1)対象分野の知識

人間が対象分野に対して持っている知識を表現したものである。

(2)問い合わせ文に用いられる単語に関する知識

対象分野に対する問い合わせ文で用いられる単語

の知識である。

(3)データベースへの写像の知識

人間が対象分野に対して持っている知識とデータベースとの写像関係の知識である。

対象分野の知識は問い合わせ文を解析する場合に生ずる文節間の修飾関係の曖昧さを解決するために用いられる。また単語に関する知識もこのために必須である。また、写像に関する知識は問い合わせ文を解析した結果の意味構造をデータベースシステムの問い合わせ言語に変換するために必要である。

世界モデルはクラスとアークの2つの基本的な概念から構成される。クラスは対象分野における事物に対応するもので、アークは事物間の関係を表したものであり、世界モデルはクラスをアークで結んだネットワークとして表現されている。アークには属性関係と上位・下位関係がある。

世界モデルはモデリングシステム REALM で管理される。REALM は知識プログラミングシステム FOLK

[3] を用いて世界モデルの管理およびその知識の操作を行うために作成されている。FOLK は LOOPS [4] と同じように3レベルから成るオブジェクト指向の知識表現構造を採用しており、世界モデルはクラスレベルで表現されている。REALM では写像を記述するための REALM/DML コマンドとこれを解釈して問い合わせ文に対応した REALM/DML コマンドを生成する知識ならびにそれをデータベースの問い合わせコマンドに変換する手続きを提供している。

商品の販売情報に関する世界モデルの一部を図2に示す。商品クラスは商品コード、商品名、商品価格の各クラスを属性クラスとして持っている。ビールおよびウィスキークラスは商品クラスの下位クラスであり商品の属性を遺伝する。

単語の情報はクラスまたはアークに対応づけて定義を行う。例えば、「商品」という名詞は商品クラスに対応させる。また「売る」という動詞は販売クラスに対応させ、「存在する」という動詞は小売店クラスから小売店所在地をさすアークに対応させる。クラスに

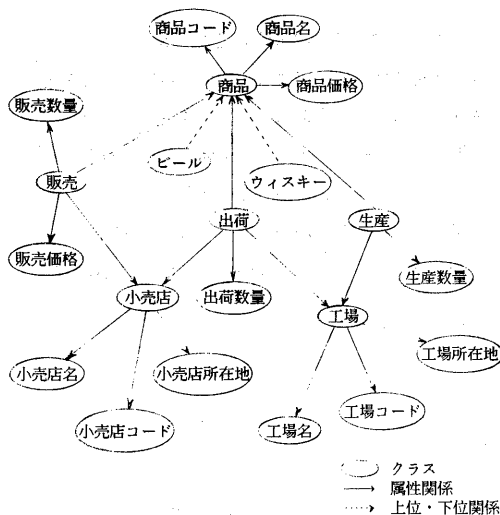


図2 販売世界モデルの一部

```

PRODUCTION (生産)
class   ¥value   REALM:NONPRIMITIVE
super   ¥value   REALM:ENTITY
level   ¥value   CLASS-LEVEL
factory ¥class   FACTORY
¥storage ¥class PRODUCTION-FACTORY-STORAGE
quantity ¥class PRODUCTION-QUANTITY
¥storage ¥class PRODUCTION-QUANTITY-STORAGE
merchandise ¥class MERCHANDISE
¥storage ¥class PRODUCTION-MERCHANDISE-STORAGE

```

```

PRODUCTION-FACTORY-STORAGE (生産→工場の写像)
class   ¥value   REALM:STORAGE
super   ¥value   @ENTITY@
level   ¥value   CLASS-LEVEL
get     ¥value   GET   factory-tbl : factory-code-flid
                                factory-tbl : factory-name-flid
                                WHERE factory-tbl : factory-code-flid
                                = produce-tbl : produce-fcode-flid

```

図3 生産クラスとその写像クラス

対応する動詞をクラス動詞、アークに対応する動詞をアーク動詞と呼ぶ。

図3に生産クラスとその写像クラスを示す。クラスには、メタクラスを示す CLASS スロット、上位クラスを示す SUPER スロット、レベルを示す LEVEL スロットなどのシステム定義スロットがある。factory, quantity, merchandise スロットはユーザー定義スロ

トでありその ¥class ファセットは属性関係を表わしている。また ¥storage ファセットは、アークに対応した写像情報を記述した写像クラスを記述している。PRODUCTION-FACTORY-STORAGE クラスは生産クラスから工場クラスを指すアークに対する写像クラスであり、その get スロットには工場テーブルの工場コードフィールドと生産テーブルの工場コードフィールドとをジョインして、工場コードと工場名を検索する REALM /DML コマンドが記述されている。

4. 形態素解析

本システムでは、かなで入力した問い合わせ文を端末側のかな漢字変換辞書によってかな漢字混じり文に変換し形態素解析の入力としている。ホスト側の単語辞書には単語のカナ読み、ローマ字読み、漢字辞書的な情報は不要となる。

形態素解析における曖昧性を解消するためには、

- ①形態素的情報
- ②構文的情報
- ③意味的情報
- ④文脈・知識情報
- ⑤その他の情報

などの情報が考えられるが、本システムでは構文・意味解析以降の処理と形態素の処理を分離したので、②～④の情報は利用していない。これによって発生する複数の単語分割の候補に対しては、文節数最小法 [5] によって選択された尤度の高い候補から順に構文・意味解析を成功するまで行う方式をとっている。

形態素解析で使用する品詞情報、接続情報は曖昧性を排除するためにできるだけ精密化する必要がある。一方、ユーザが登録する必要のある単語に対して精密な情報を要求するとユーザの負担が大きくなる。このため、単語を内容語(名詞、動詞、形容詞、形容動詞)と機能語(助動詞、助詞、副詞、接続詞など)とに大別し、ユーザの扱う内容語に対しては品詞分類を粗く、登録項目数を少なくし、機能語に対しては品詞分類を細かく、接続情報を精密にしている。

形態素解析の結果は、単語とそれに対する構文・意味解析用の特徴集合とをペアにしたリストの形で作られ、構文・意味解析に渡される。

5. 構文・意味解析

モデル指向パーサでは構文解析時に世界モデルに基づいた意味解析を同時に行うことによって文節間の修飾関係の曖昧さを解決している。本パーサでは各文節の意味的中心である単語に対応する世界モデル上のクラス間に上位・下位関係または属性関係が存在する場合に文節間の修飾関係が成立すると判断し、その修飾関係の強さは関係のパスの長さが短い程強く同一の強さなら左側を優先する。また、上位・下位関係による修飾は属性関係によるそれよりも強い。

文解析において世界モデルを使う場合には次の原理に基づいている。

(1)特殊化原理

上位クラスと下位クラスでは下位クラスの方の意味を重視する。

(2)中心化原理

属性関係で関係付けられたクラス間では、アークの始点側のクラスに意味の中心がある。

文解析では解析ツールとして TEC [6] を使用しており、解析ルールは TEC の文法記述言語である PIDGIN を用いてパケットを単位とする部分文法の集合として記述されている。解析ルールは上述の原理を使って記述されるが、その基となるルールは次のルールである。

(1)同一化ルール

上位・下位関係にあるクラスに対応する2つの文節を1つにまとめ、特殊化原理によって下位クラスの意味を全体の意味とする。

(2)結合ルール

属性関係にあるか、直接属性関係がなくても上位・下位関係を通じて属性関係がある場合はその2つのクラスに対応する文節を1つにまとめ、特殊化原理と中心化原理によって属性関係のアークの起点側のより下位のクラスの意味を全体の意味と

する。

パーサの処理の流れの概略を図4に示す。文節生成では、形態素解析から渡された特徴付単語列から文節を構成し世界モデル上の対応クラスを決定して以降の文節間の修飾関係の決定処理の準備を行う。並列処理では「と」、「か」、「または」などを挟んだ2つの文節に対応するクラスが同一の場合に、これらを1つの文節にまとめる。2つの文節が助詞なしで接続されて

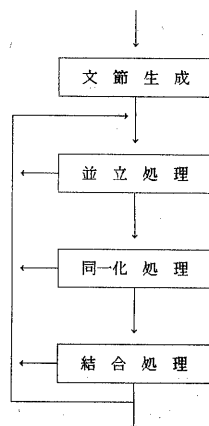


図4 構文・意味解析処理の流れ

いる場合もこの処理の対象となる。同一化処理は「の」同一化ルールと「が」同一化ルールから成り、この順に処理が行なわれる。「販売価格が2000円の」という文は、「が」同一化ルールによって1つの文節にまとめられ、全体の意味は販売価格となる。結合処理は「と」結合ルール、「の」結合ルールおよび述語結合ルールからなり、この順に処理が行なわれる。「小売店の名前」という文は「の」結合ルールを使って文節にまとめられ、全体の意味としては小売店が探られる。述語結合ルールでは、述語節とその述語のとり得る格を持った文節とを結合ルールでまとめて文を構成し、さらにこれと結合できる名詞節があればまとめて名詞節を作る。文節間の修飾関係を決定するこれらの処理は、図4に示すように1つのルールが実行される

と最初のルールに戻って再び適用を試み、与えられた特徴付単語列の全てが1つの文または名詞節にまとめられれば終了するが、未統合の文節があれば再び修飾関係の決定処理を繰り返す。最終的にパーサは出来上がった解析木とその全体の意味を文脈処理に渡す。本解析方式は、世界モデルを利用した明確なルールに基づいた処理を単純な制御構造で実行するので、記述が容易であると共に理解し易く、解析ルールの拡張も文法ルール全体の安定性を損なわずに行うことができる。

6. 文脈処理

文脈処理では、構文・意味解析から渡された解析木とその全体の意味から問い合わせ文の意図を表現する SPEECH-ACT を作成し、これを用いて前文との照応関係を解決している。SPEECH-ACT は問い合わせ文の基準概念のクラスを示す concept, 構文的に中心となるクラスを示す syntax, 検索対象項目のクラスを示す target, 検索条件のクラスを示す qua などの情報を保持している。

SPEECH-ACT を使って文脈処理を行う際に用いるルールは構文・意味解析の基本ルールである同一化ルール、結合ルールである。構文・意味解析では文節間の修飾関係を決定するためにこれらのルールを使用したのが、文脈処理では今回の文と前文の SPEECH-ACT の各情報間にこれらのルールを適用して照応関係を解決する。

文脈処理では同一化ルール、結合ルールの他に同族化ルールを使用する。同族化ルールは同一の上位クラスを持つ下位クラス間には文脈があると判定し前文のクラスを置換するルールである。これらのルールは同一化ルール、同族化ルール、結合ルールの順で適用され各ルール内では syntax, concept, target, qua の順にルールが適用される。また文脈の有無の判定や照応処理には、使用された代名詞の種類(その、それらの、その中等)や条件記述の数の大小などをヒューリスティクスとして用いている。

図5は文脈処理による SPEECH-ACT の変化を示した

ものである。この場合は「その中で」という単語の存在から文脈があると考え前文の SPEECH-ACT の照応処理を行い concept は前文の販売クラスを継承し、target にも販売クラスが付加された形となり販売→商品→商品コードというパスによって指定された商品コードを検索対象としている。qua も同様に販売クラスを中心とした条件の指定に変更されている。target-D および qua-D は前文までの累積された検索対象および条件であり、target と target-D, qua と qua-D がマージされたものが今回の問い合わせ文の意図を表わしている。

富士通屋が販売している商品の名前は？

concept	(販売)
syntax	(商品名)
target	(販売 商品 商品名)
qua	(販売 小売店 小売店名)

その中で定価が1000円以上かつ3000円以下の商品コードは？

concept	(商品)
syntax	(商品コード)
target	(商品 商品コード)
qua	(商品 商品価格)

照応処理後

concept	(販売)
syntax	(商品コード)
target	(販売 商品 商品コード)
target-D	(販売 商品 商品名)
qua	(販売 商品 商品価格)
qua-D	(販売 小売店 小売店名)

図5 SPEECH-ACT による文脈処理

7. コマンド生成とデータ検索

コマンド生成およびデータ検索では図6に示すような REALM システムのオブジェクトを使って処理を行う。DML オブジェクトは SPEECH-ACT から REALM/DML コマンドを生成する知識を持っており、QL オブジェクトは REALM/DML コマンドをデータベースの問い合わせ言語に変換する知識ならびにそれを実行する手続

きを持っている。REALM オブジェクトはこれら2つのオブジェクトの処理を制御するオブジェクトである。

コマンド生成部は、DML オブジェクトに対してmake-dml というメッセージを送る。DML オブジェクトは、SPBEBCH-ACT につながるオブジェクトの写像情報を参照して REALM/DML コマンドを作成する。

データ検索部が REALM オブジェクトに interpret というメッセージを送ると、REALM オブジェクトは QL オブジェクトにまず translate というメッセージを送って、REALM/DML コマンドをデータベースの問い合わせ言語に変換する。この変換の知識は、手続きを明確かつ柔軟にするために、図7に示すような複数のルールセットのオブジェクトで表現されている。各ルールセットは、問い合わせ言語の構文(図8参照)に応じてG-rule, Q-rule, GB-rule, H-rule などがあり、それぞれ GET 句、WHERE 句、GROUP BY 句、HAVING句に対応した変換ルールをプロダクション・ルールの形式で記述した変換のエキスパートである。QL オブジェクトが T-rule という変換を制御するルールセットに RUN メッセージを送ると T-rule は各句に対応したルールセットを順に RUN メッセージで起動して問い合わせコマンドを作成する。REALM オブジェクトは、次に evaluate メッセージを QL オブジェクトに送り、問い合わせコマンドを実行する。

8. モデルエディタ

システムを特定の対象分野向けに開発し、それを利用するには、単語辞書ならびに世界モデルの作成・編集の容易性が肝要である。モデル・エディタ WED は、このユーザの作成・編集を支援するためのツールである。

世界モデルを新規作成する場合には、対象分野の概念を抽出してクラスとしそれらの間の関係をアークで表現することになるが、最初からすべての概念がそれらの間の関係と共に決定できることは少なく、試行錯誤を伴いながら編集を繰り返すことが普通である。また、この過程の中で決定された概念に対して、その内

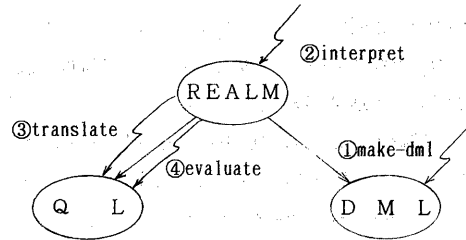


図6 コマンド生成とデータ検索処理の流れ

```

    GET フィールド名, ...
    FROM テーブル名, ...
    WHERE 条件式
    GROUP BY フィールド名, ...
    HAVING グループ選択条件
  
```

図8 問合せコマンド

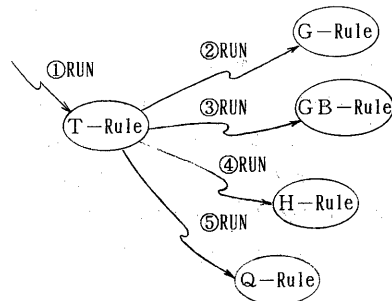


図7 コマンド変換用ルールセット

容を詳細に記述する作業が随時行なわれる。このような編集作業を円滑に行うために、モデルエディタに図9に示すような階層を設けている。

(1)構成記述レベル

構成記述レベルでは、クラスとアークの編集を行う。クラスに関する操作にはクラス生成、クラス編集、クラス削除およびクラス関係の表示がある。このレベルのクラス編集では、上位・下位関係などの他クラスとの関係を示す部分の編集のみを行

う。アークに関する操作にはアーク生成、アーク削除、アーク移動などがある。

(2) 単語割当レベル

単語割当レベルでは作成されたクラスまたはアークに対する単語の割当、単語の割当解除および割り当てられた単語の表示などを行う。このレベルで編集の対象となる単位は名詞、動詞などの内容語のみである。

(3) 内容記述レベル

内容記述レベルには、クラス内容記述レベルと単語内容記述レベルがある。このレベルではクラスおよび単語のデータ構造を意識した編集を行う。クラス内容記述レベルでは、クラスのスロット、ファセット、値という構造を直接編集するので、アークの内容編集はクラスのスロット情報の編集に帰着する。単語内容記述レベルでは、単語辞書の識別子、品詞、パーサ情報、セグメント情報などの各項目単位の編集を行う。

モデルエディタのユーザとしては、システムの利用者であるエンドユーザと管理者であるデータベースのエキスパートとが考えられる。エンドユーザは、構成記述レベルおよび単語割当レベルを使ってクラス・アークのみを意識した編集を行うことができる。またエキスパートは、内容記述レベルを使ってクラスとデータベースとの写像関係などの情報を入力・編集する。このように、ユーザのレベルによってモデルエディタのインタフェースを使い分けることができるので、各ユーザの世界モデルならびに単語辞書に対するビューに合った形での編集を行うことができる。未登録語の発生時などのようにエンドユーザが構成記述レベルと単語割当レベルを利用して対処できる場合とエキスパートによる内容記述レベルの編集が必要な場合とがある。従ってシステムはあるクラスまたは単語が使用できる状態にまで編集が行われているか否かを知る必要がある。このため、WED ではクラスおよび単語に未完成フラグを設け、システムが利用できる状態か否かを示すように管理している。

モデルエディタによる構成記述レベルの編集例を次に示す。クラス関係の表示では、図10に示すようなウィンドウ構成で編集を行う。表示を行う場合は、どのクラスを中心にして表示を行うかを指定し、次にそのクラスのまわりのどのような関係について表示するかを指示すると、表示ウィンドウに示された形式でクラス関係の表示が行なわれる。1階終点は中心のクラスを始点とするアークの終点のクラスであり、2階終点は1階終点のクラスを始点とするアークの終点のクラスである。1階始点は中心のクラスを終点とするアークの始点クラスである。アーク生成の画面を図11に示す。アーク生成ではアークの始点およびアークの終点のクラスを指定することによってその2つのクラス間の関係付けを行う。

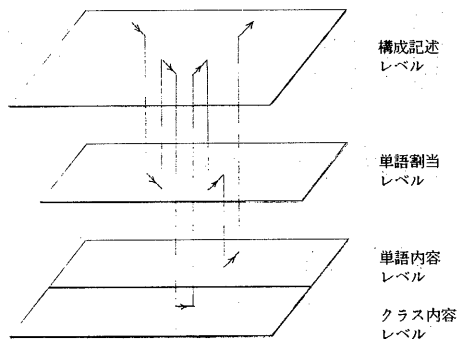


図9 モデルエディタの階層

9. おわりに

世界モデルを利用したデータベースの日本語インタフェースについて述べた。本システムは Utilisp を用いて開発した。現在のシステムサイズは約7200行であり、WED, FOLK, ウィンドウシステムなどのツールが約6500行である。文法ルールは PEDGIN で約1500行あり約50個のパケットから構成されている。

今後は、文脈処理の拡張・高度化およびモデルエディタのユーザインタフェースの改良ならびに機能拡張を進める予定である。

メニュー (近接クラス表示)	
1. 上位下位関係表示 2. 上層下層関係表示 3. アーク関係表示	4. 表示中心の指定 0. 終了
指示	
メニューから操作の番号を選択して下さい。	
入力	
選択番号=> 3	
表示	
2階終点 STORE-CODE STORE-NAME STORE-ADDR	1階終点 RETA?STORE SHIP?QUANT
中 心 SHIPMENT	1階始点 2階始点

図10 近接クラス表示の画面

メニュー (アーク生成)	
1. 別アーク生成 2. 近接クラス表示	0. 終了
指示	
アークの始点のクラス名を入力して下さい。 アークの終点のクラス名を入力して下さい。	
入力	
アーク始点=> SHIPMENT アーク終点=> FACTORY	
表示	
2階終点 STORE-CODE STORE-NAME STORE-ADDR	1階終点 RETA?STORE SHIP?QUANT
中 心 SHIPMENT	1階始点 2階始点
新アーク SHIPMENT => FACTORY を生成しました。	

図11 アーク生成の画面

参考文献

- [1] Waltz, D. L. : An English Language Question Answering System for a Large Relational Database, Comm. ACM, Vol. 27, No. 7, pp. 526-539 (1978).
- [2] Grosz, B. J. : TEAM: A Transportable Natural-Language Interface System, Proc. Conf. Applied Natural Language Processing, pp. 39-45 (1983).
- [3] 泉田: 日本語によるデータベース検索システム, 第25回プログラミングシンポジウム, pp. 181-190 (1984).
- [4] Stefik, M. and Bobrow, D. G. : The LOOPS Manual: A Data Oriented and Object Oriented Programming System for Interlisp, Xerox Knowledge-Based VLSI Design Group Memo KB-VLSI-81-13 (1981).
- [5] 吉村, 日高, 吉田: 文節数最小法を用いたべた書き日本語文の形態素解析, 情報処理学会論文誌, Vol. 24, No. 1, pp. 40-46 (1983).
- [6] 杉山, 谷内田, 牧之内: 自然言語解析向き解析木作成ツールTEC, 情報処理学会第25回全国大会講演論文集, pp. 1033-1034 (1982).
- [7] Shipman, D. W. : The Functional Data Model and the Data Language DAPLEX, ACM Trans. Database Syst., Vol. 6, No. 1, pp. 140-173 (1981).
- [8] Hammer, M. and Mcleod, D. : Database Description with SDM: A Semantic Database Model, ACM Trans. Database Syst., Vol. 6, No. 3, pp. 351-386 (1981).

謝辞

本研究の機会を与えて頂いた林達也ソフトウェア研究部長に感謝致します。