

TR-I-0154

ADD検索システム利用解説書

A Retrieval System for The ADD (ATR's Linguistic Database)
User's Manual

橋本一男

Kazuo Hashimoto

1990.3

概要

本稿は、言語データベースADD検索システムの利用解説書である。内容は次のように4部構成になっている。

- 第1部 チュートリアル
- 第2部 ADD検索システム操作説明書
- 第3部 パターンエディタ操作説明書
- 第4部 DEFSEARCH文法説明書

ATR 自動翻訳電話研究所
ATR Interpreting Telephony Research Laboratories

© ATR 自動翻訳電話研究所
© ATR Interpreting Telephony Research Laboratories

関連文献

ATRテクニカルレポートTR-I-0150

自動翻訳電話研究用言語データベースの構成

ATRテクニカルレポートTR-I-0153

言語データベースADDの検索システム

第1部

チュートリアル

第1.0版

目次

1	はじめに	1
2	実行環境の設定	2
3	検索システムの起動	2
4	検索範囲の指定	3
5	検索パターンの作成	4
	STEP1 パターンエディタの起動	4
	STEP2 検索条件の記述	4
	STEP3 リターン形式の記述	10
	STEP4 検索パターン名の記述	10
	STEP5 コメントの記述	12
	STEP6 検索パターンのセーブ	12
6	検索実行	13

1 はじめに

- 本書では検索システムの実際の操作例を示します。実際に端末に向かって、本書の通りに操作をすれば、ADD検索システムの検索を体験することができます。
- 検索の内容は、『「固有名詞を含む文節および文」を求める』というものです。
- システムからの応答はゴシック体で、利用者が入力するコマンドはボールド体(**boldface**)で示してあります。また次のような記号を使っています。

<ctrl> + <文字> コントロールキーを押しながら、<文字>キーを押します。<ctrl>+vの場合、コントロールキーを押しながらvの文字を押してください。

<esc> + <文字> エスケープキーを押してから、<文字>キーを押します。<ctrl>+vの場合、コントロールキーを押しながらvの文字を押してください。

2 実行環境の設定

1) 次の操作は必須です。

```
% /usr/jsy/bin/stty/dec dumb  
% set term=vt282
```

2) 日本語が入力できるように、適当な(好みの)日本語FEPを用意しておいてください(たとえば、ATOKなど)。日本語FEPがない場合は、次のようにすれば、Ultrix提供のローマ字漢字変換モジュールが使えるようになります。

```
% /usr/jsy/bin/stty -even -odd  
% setenv JSYKOJIN ~/jsykojn.dic  
% /usr/jsy/bin/stty jin
```

この場合、日本語入力の方法は次の通りです。

無変換(小文字)	テンキーの0を押す
無変換(大文字)	テンキーの3を押す
ひらがな変換	テンキーの1を押す
カタカナ変換	テンキーの2を押す
漢字変換	テンキーの7を押す

3 検索システムの起動

1) 起動

```
% ldbsh  
ldbsh:
```

2) 終了

```
ldbsh:end  
%
```

4 検索範囲の指定

1) targetコマンド入力と検索範囲指定方法の選択

検索に先立ち、検索範囲(検索対象)を指定するためtargetコマンドを入力します。その結果、「範囲指定方法を選択して下さい。」というメッセージとメニューが表示されます。範囲指定方法には、会話属性指定と会話番号指定の2通りがありますが、ここでは2の会話番号指定(初期化)で行います。

```
ldbsh: target
```

範囲指定方法を選択して下さい。

0. 検索対象表示

1. 会話属性指定

2. 会話番号指定(初期化)

3. 会話番号指定(前回の継続)

4. 会話番号指定(属性指定の結果を使用)

= => 2

2) 会話番号の選択

会話番号選択画面が現れたら、会話IDが83の行が現れるまで、<ctrl>+vを使って画面送りをします(画面戻しは<esc>+v)。現れたら、矢印キー(↓と↑)で、83の行までカーソル(アンダーライン)を移動して、リターンキーを押します。これで、会話番号83が選択されました(反転表示される)。選択を解除するときは、もう一度リターンキーを押します。

<< 会話番号選択 >>			
会話ID	発話数	文数	単語数
81	14	29	662
82	15	47	1204
83	10	28	575
84	17	48	1191
85	21	47	1259
86	9	26	727
87	17	96	3689
88	11	64	2443
89	12	55	2163

3) 選択終了

qキーを押すと(カーソル位置はどこでもよい)、選択が終了し、対象会話が確定します。

```
対象会話: 83
```

```
ldbsh:
```

5 検索パターンの作成

STEP 1 パターンエディタの起動

1) パターンエディタの起動

検索要求を作成するため、editpコマンドを入力してパターンエディタを開きます。パターンエディタは、利用者が表に記入する形式で、検索要求を作成できる簡易ツールです。

ここでは、「固有名詞を含む文節および文」を求めるという問合せを考えます。まず、editpコマンドを入力し、「ベースのテーブル名を入れて下さい[デフォルト:単語:]」というメッセージが表示されたら、リターンキーを押します(デフォルトを用いる)。- 利用者は上記問合せに対する答えが、単語テーブルを参照して求められることを知っているとする -。

```
ldbsh: editp
ベースのテーブル名を入れて下さい[デフォルト:単語]:
```

STEP 2 検索条件の記述

2) 単語条件の記述

次の画面が現れたら、矢印キー(↓と↑)でカーソルを最下段の\$<単語1>行に移動して、eキーを押します。

```
<<メインウィンドウ>>
パターン名      :
コメント       :
リターン形式    : [レベル] 文
-----
単語           : $<単語1>
```


3) 品詞条件の記述

次の画面が現れたら、矢印キー(↓と↑)でカーソルを品詞の行に移動して、eキーを押します。

<<オブジェクトウインドウ>>	
パターン名	:
コメント	:
リターン形式	: [レベル]文

単語	: \$<単語i>
AFTER	:
BEFORE	:
文字数	:
PART-OF	:
PREV-IS	:
NEXT-IS	:
出現単語	:
読み	:
標準表現	:
音便	:
品詞	:
活用形	:
活用型	:
係受け	:
単語対応	:
ランダム対応	:
日本語格	:

4) 固有名詞の選択

画面左上に次のようなポップアップメニューが現れたら、矢印キー(↓、↑、→、←)でカーソルを「固有名詞」に移動して、リターンキーを押します。これで、品詞として固有名詞が選択されました(反転表示になる)。選択を解除するときは、もう一度リターンキーを押します。

品詞表	
EXIT	終助詞
記号	接尾語
形容詞	接頭語
形容動詞タ	補助動詞
形容動詞タ	自動詞
普通名詞	他動詞
サ変名詞	固有名詞
代名詞	形容名詞
数詞	本動詞
副詞	間投詞
連体詞	準体助詞
接続詞	並立助詞
感動詞	係助詞
助動詞	慣用句
副助詞	その他
接続助詞	NIL
格助詞	

5) PART_OF条件の記述

qキーを押すと(カーソル位置は任意)、選択が終了し、品詞欄に固有名詞が記入されます。次に、カーソルをPART_OFの行に移動し、eキーを押します。

<<オブジェクトウインドウ>>	
パターン名	:
コメント	:
リターン形式	:[レベル]文

単語	:\$<単語1>
AFTER	:
BEFORE	:
文字数	:
PART-OF	:
PREV-IS	:
NEXT-IS	:
出現単語	:
読み	:
標準表現	:
音便	:
品詞	:固有名詞
活用形	:
活用型	:
係受け	:
単語対応	:
ランダム対応	:
日本語格	:

6) 文節条件の新規作成

画面左上に次のようなポップアップメニューが現れたら、矢印キー(↓、↑)でカーソルを「展開」に移動して、リターンキーを押します。

エディット方法
EXIT
展開
パターン指定

7) 単語の順序関係の記述その1

次の画面が現れたら、矢印キー(↓と↑)でカーソルをCOMPOSED_OFの行に移動して、eキーを押します。

<<オブジェクトウインドウ>>	
パターン名	:
コメント	:
リターン形式	:[レベル]文

文節	:\$<文節1>
AFTER	:
BEFORE	:
単語数	:
PART-OF	:
COMPOSED-OF	:\$<単語1>
PREV-IS	:
NEXT-IS	:
文節対応	:

8) ワイルドカード挿入

画面左上に次のようなポップアップメニューが現れたら、矢印キー(↓、↑)でカーソルを「*挿入」に移動して、リターンキーを押します。

※ *は0個以上の単語を表す。

エディット方法
EXIT
展開
パターン指定
*挿入
?挿入
擬似マルチ
削除
全部削除

9) 挿入位置の選択

画面左上に次のようなポップアップメニューが現れたら、矢印キー(↓、↑)でカーソルを上の方の「★」に移動して、リターンキーを押します。

挿入位置
EXIT
★
単語1
★

10) 単語の順序関係の記述その2

さきほどの画面のCOMPOSED_OF行の\$<単語1>の前に、ワイルドカード*が挿入されます。次に、7)、8)、9)と同様の操作で\$<単語1>の後にもワイルドカード*を挿入します。9)のポップアップメニューの内容が少し変わりますが、一番下の★を選んで下さい。

<<オブジェクトウインドウ>>	
パターン名	:
コメント	:
リターン形式	:[レベル]文

文節	:\$<文節1>
AFTER	:
BEFORE	:
単語数	:
PART-OF	:
COMPOSED-OF	:\$<単語1>
PREV-IS	:
NEXT-IS	:
文節対応	:

挿入位置
EXIT
★
*
★
単語1
★

11) 文条件の記述

COMPOSED_OF行の\$<単語1>の後に、ワイルドカード*が挿入されました。次に、カーソルをPART_OFの行に移動して、eキーを押します。

<<オブジェクトウインドウ>>	
パターン名	:
コメント	:
リターン形式	:[レベル]文

文節	:\$<文節1>
AFTER	:
BEFORE	:
単語数	:
PART-OF	:
COMPOSED-OF	:\$<単語1>*
PREV-IS	:
NEXT-IS	:
文節対応	:

12) 文節の順序関係の記述

次の画面が現れたら、カーソルをCOMPOSED_OFの行に移動して、eキーを押します。以降、8)、9)、10)と同様の操作をします。

<<オブジェクトウィンドウ>>	
パターン名	:
コメント	:
リターン形式	:[レベル]文

文節	:\$<文1>
AFTER	:
BEFORE	:
文節数	:
PART-OF	:
COMPOSED-OF	:\$<文節1>
PREV-IS	:
NEXT-IS	:
文対応	:

13) メインウィンドウへの移動

COMPOSED_OF行の\$<文節1>の前後に、ワイルドカードが記入されていることを確認したら、qキーを押します(カーソル位置は任意)。

<<オブジェクトウィンドウ>>	
パターン名	:
コメント	:
リターン形式	:[レベル]文

文節	:\$<文1>
AFTER	:
BEFORE	:
文節数	:
PART-OF	:
COMPOSED-OF	:\$<文節1>*
PREV-IS	:
NEXT-IS	:
文対応	:

STEP 3 リターン形式の記述

14) リターン形式の記述

メインウィンドウが現れます。2)のときと比べて、\$<文節1>と\$<文1>の行が新しく加わっていることを確認して下さい。

※ カーソルをいずれかの行に移動し、eキーを押せば、再びオブジェクトウィンドウに入って編集できます。オブジェクトウィンドウからメインウィンドウへは、qキーで移ります。

カーソルをリターン形式行に移動し、eキーを押して下さい。

<<メインウィンドウ>>	
パターン名	:
コメント	:
リターン形式	:[レベル] 文

単語	:\$<単語1>
文節	:\$<文節1>
文	:\$<文1>

15) リターン形式の変更

画面左上に、次のようなポップアップメニューが現れたら、矢印キー(↓、↑)でカーソルを「リターン形式変更」に移動して、リターンキーを押します。

選択
EXIT
リターン形式変更
パラメータ編集

16) 変数指定の選択

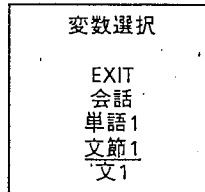
画面左上に、次のようなポップアップメニューが現れたら、矢印キー(↓、↑)でカーソルを「変数指定」に移動して、リターンキーを押します。

リターン形式
EXIT
変数指定
レベル指定
統計

17) 文節文字列の選択

画面左上に、次のようなポップアップメニューが現れたら、矢印キー(↓、↑)でカーソルを「文節1」に移動して、リターンキーを押します。画面下に「文字列に変換しますか? y/n?」というメッセージが表示されるので、yと入力する。

※ yならば文節文字列を、nならば文節IDを出力として指定することになる。



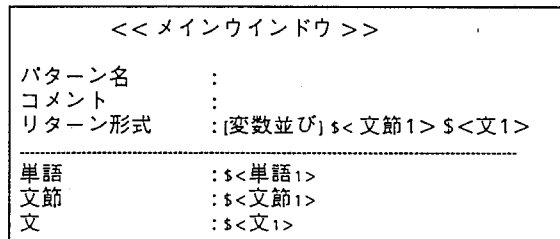
18) 文文字列の選択

引続きポップアップメニューが表示されているので、17)と同様の操作を、「文1」についても行う。

STEP 4 検索パターン名の記述

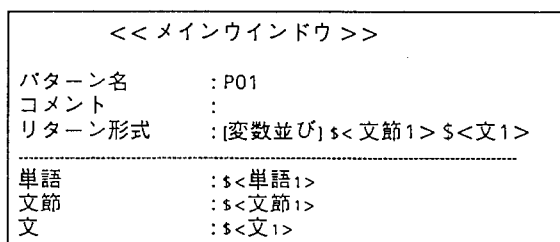
19) 検索パターン名の記述1

リターン形式行の値が、次のように変更されているのを確認したら、パターン名の行にカーソルを移動して下さい。次に、eキーを押して、カーソルがコロン(:)の右側に移動したら、<ctrl>+cを入力します。



20) 検索パターン名の記述2

画面右下に「漢字入力」と表示されたら、適当な名前(たとえば、q01)を入力します(日本語も可)。次に2度、リターンキーを押すとパターン名入力が終了し、カーソルが左端に移動します。名前を変えたいときは、19)からもう一度繰り返します。



STEP 5 コメントの記述

21) コメントの記述

コメント行への記入も、19)、20)と同様に行います。ここでは、「固有名詞を含む文節および文」と記入します。ただし、コメントを記入する、しないは利用者の自由です。

<<メインウインドウ>>	
パターン名	: P01
コメント	: 固有名詞を含む文節および文
リターン形式	: [変数並び] \$<文節1> \$<文1>

単語	: \$<単語1>
文節	: \$<文節1>
文	: \$<文1>

STEP 6 検索パターンのセーブ

22) 検索パターンのセーブ

以上で、目的の検索パターンが作成できました。sキーを押して(カーソル位置は任意)、これをファイルにセーブします。画面左下に「パターンをセーブして終了します。 Y/N? 」と表示されるので、yと答える。

<<メインウインドウ>>	
パターン名	: P01
コメント	:
リターン形式	: [変数並び] \$<文節1> \$<文1>

単語	: \$<単語1>
文節	: \$<文節1>
文	: \$<文1>
パターンをセーブして終了します。 Y/N? y	

23) セーブファイルの指定

検索パターンをセーブするファイル名を聞いてくるので、適当な名前を入力します。ここではtestにします。

パターン Q01 をセーブするファイル名を入れて下さい[デフォルト]: test
ldbsh:

6 検索実行

1) 検索範囲の確認

```
ldbsh: target -l
対象会話:0083
```

2) 検索実行

psearchコマンドを用いて検索を実行します。このとき、検索パターンをセーブしたファイル名を指定します。そのファイル内のどのパターンを使うか聞いてくるので、番号で答えます。ここでは、1と入力します。

※ 所要時間は、ホスト計算機の利用状況にもよりますが、本例では約10秒かかります。

```
ldbsh: psearch test
0:キャンセル
1:Q01
番号選択してください: 1
  1. 文節1:パイパー先生の、|文1:[あの一]パイパー先生の、研究室でいらっしゃいますでしょうか。
  2. 文節1:パイパーです。|文1:はい、私がパイパーです。
  3. 文節1:国際コンピューター会議の|文1:[あ、あの一]こちら国際コンピューター会議の事務局ですけれども、いつもお世話になっております。
検索 "固有名詞を含む文および文節()" が終了しました。
```


第2部

ADD検索システム操作説明書

第1.0版

目次

	頁
1. 概要	1
2. 主な機能	3
3. 実行環境	4
4. コマンド	5
4. 1 検索範囲指定 (target)	8
4. 2 検索	11
4. 2. 1 パターン指定検索コマンド (psearch)	11
4. 2. 2 単語条件列検索コマンド (wsearch)	13
4. 2. 3 S Q L (sql)	15
4. 2. 4 その他検索サポートコマンド	16
4. 3 結果処理 (mode)	24
4. 4 ヘルプ (help)	25
4. 5 その他	27
4. 5. 1 U N I Xコマンドの実行	27
4. 5. 2 終了 (end)	27
5. 単語条件列検索文法	28

1. 概要

本書では、ADD検索システムのMMI（マンマシンインタフェース）となるLDBSHの操作について説明する。

LDBSHはコマンドシェルである。用意されているコマンドは全部で13種類あり、それらの機能と使用方法を解説する。またLDBSH上からのUNIXコマンドの実行や検索結果のリダイレクション、検索のバックグラウンド処理の指定などについても解説を行う。

LDBSHを利用するための端末としては、VT282を想定しているが、エミュレーションできれば特に機種は問わない（たとえばPC-9801でも使える）。

検索コマンドとして、次に述べるパターン指定検索と単語条件列の2つが用意されているが、それ以外に対話型SQLも起動できる。

A. パターン指定検索コマンド (psearchコマンド)

検索パターンファイル(、パターン名)を指定して検索実行する。

パターンファイルというのは検索パターン(以後、DEFSEARCH文と呼ぶこともある)が1つ以上定義されているファイルである。

検索パターンファイルは、パターンエディタ、または通常のエディタであらかじめ作成しておく。

検索終了後、検索結果は標準出力(デフォルトは端末)に出力される。

パターンエディタについては、「パターンエディタ操作説明書」を参照、検索パターンについては、「DEFSEARCH文文法説明書」を参照。

(例)

```
% ldbsh
ldbsh: psearch patterns1 P1

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

ldbsh: end
%
```

B. 単語条件列検索コマンド (wsearchコマンド)

検索条件を単語レベルに限定し、単語条件の並びを検索条件とするコマンドである。

文法については4.2.2で説明する。

(例)

```
% ldbsh
ldbsh: wsearch は/品詞:格助詞 + * + する

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

ldbsh: end
%
```

以後、この条件列を"検索文"と呼ぶ。

2. 主な機能

○ 検索範囲の指定

次の2つの指定方法が用意されている。

1. 会話属性による検索範囲指定

会話の属性指定を行い、対象の会話を選択する。

2. 会話番号指定による検索範囲指定

会話番号の一覧を画面表示し、対象の会話を選択する。

○ 検索実行

1. に示した2つの検索方法とSQLが用意されている。

○ 検索結果処理

検索結果について、次のような処理を行える。

なお、これらは「検索モード」の概念であり、検索実行前に指定しておく。

- ・ 検索の結果の日本語表示
- ・ 対訳表示
- ・ ユニーク処理
- ・ 結果文字列の統計処理（カウント）
- ・ 結果のソート（ユニーク、統計指定時は強制的にソートされる）

○ セーブ機能

検索パターン、検索結果をファイルにセーブする機能が用意されている。セーブした検索パターンを、検索実行に使用する事も可能である。

- ・ 検索パターン（文字列）のセーブ
- ・ 検索パターン（DEFSEARCH文展開後）のセーブ
- ・ 検索結果のセーブ（リダイレクション）

○ パターンエディタ

検索パターンを作成するための支援ツールとしてパターンエディタが用意されている。なおこれについては「パターンエディタ操作説明書」を参照。

○ その他

- ・ ヘルプ機能（コマンド、テーブル名、フィールド名などの一覧表示）
- ・ UNIXコマンドの実行
- ・ バックグラウンドでの検索実行
- ・ 実行環境セーブ機能

3. 実行環境

ldbshを実行するには、以下の様な制限事項、注意事項がある。

1.

ldbshは、あらかじめデータベース検索デーモンが起動されていないと動作しない。このデーモンはデータベース管理者しか起動できないので、必要な場合、データベース管理者に依頼しなければならない。

2.

言語データベース管理者と同じグループIDを持つユーザしか実行できない。

3.

環境変数 TERM の値が正しく、かつ、その端末に対する termcap ファイルが正しくなければならない。

4.

漢字環境が正しくなければならない。これは、sttyの端末タイプの設定、及び、かな漢字変換のフロントエンドモジュールが実行されていることを指す。

5.

リダイレクションで実行した場合、検索結果ファイルはデーモンが（作成）書き込みを行うので、そのディレクトリ、ファイルには適当なパーミッションが与えられていなければならない。

また、ファイル名は英数字、“_”、“.”で構成されるものでなければならない。

4. コマンド

用意されているコマンドの一覧を示し、その仕様について説明する。

分 類	コマンド	機 能
範囲指定	target	検索範囲の指定を行う
検索	psearch	パターンファイルを指定して検索を行う
	wsearch	単語条件列を指定して検索を行う
	sql	sqlコマンドに入る
	editp	検索パターンのエディットを行う
	editw	直前の検索文をエディットする
	restartw	直前（またはeditw直後）の検索文を再実行する
	savew	直前の検索文をファイルへセーブする
	startw	検索文ファイルを指定して実行する
	status	検索実行の状態を知る
結果処理	mode	検索結果処理方法の変更を行う
ヘルプ	help	コマンド、テーブル、フィールド名等の一覧表示
終了	end	実行環境のセーブを行い、ldbshを終了する

・その他

- UNIXコマンドは"|"でエスケープして実行する。
- 実行結果のセーブは、Cシェルとほぼ同様である。すなわち、コマンド、または、もしあれば全ての引数の後に
 > ファイル名
をつける。
 >> ファイル名
にすれば、ファイルへの追加となる。

ファイル名は英数字、"_"、"."で構成される名前でないといけない。

リダイレクションで実行した場合、検索結果ファイルはデーモンが(作成)書き込みを行うので、そのディレクトリ、ファイルには適当なパーミッションが与えられていなければならない。

バックグラウンドでの実行は、さらにその後に
 &
をつける。(バックグラウンドでの実行結果は必ずファイルへセーブされるので、"> ファイル名 &"の形となる)

そのほかの、"<"、"<<"等のリダイレクションはサポートしていない。
また、パイプ(|)もサポートしていない。

従って、許されるコマンド形式は、

 コマンド [引数] [>|>>] 出力ファイル名 [&]

となる。但し、引数は、コマンドにより、必須、オプション、なしの3通りのケースがある。

典型的な検索の流れは次のようになる。

1. targetコマンドで検索範囲を指定
2. modeコマンドで検索モードを設定
- 3-1. editpで検索パターンを作成、psearchコマンドで検索実行
または
- 3-2. wsearchで検索実行
4. (UNIXコマンドで)検索結果の評価
5. end

但し、これはあくまでも典型的な例であり、この順序にこだわる必要はない。

4. 1 検索範囲指定 (target)

<コマンド形式>

```
target [-a/-n/-l]
```

<オプション>

- a: 会話属性指定 (次の1を選択した場合と同じ)
- n: 会話番号指定 (次の2を選択した場合と同じ)
- l: 検索対象表示 (次の0を選択した場合と同じ)

検索範囲の指定は、targetコマンドにより行う。

1度検索範囲の設定を行うと、再びtargetコマンドで再設定するまで、その検索範囲が使用される。

このコマンドでは、まず、指定方法を選択する。

(例)

```
% ldbsh
ldbsh: target
範囲指定方法を選択して下さい。
0. 検索対象表示
1. 会話属性指定
2. 会話番号指定 (初期化)
3. 会話番号指定 (前回の継続)
4. 会話番号指定 (属性指定の結果を使用)
=> 1
```

検索対象表示 (0) を選択すると、1~4によって選択された、現在の検索対象の会話IDが表示される。

(例)

```
% ldbsh
ldbsh: target
範囲指定方法を選択して下さい。
0. 検索対象表示
1. 会話属性指定
2. 会話番号指定 (初期化)
3. 会話番号指定 (前回の継続)
4. 会話番号指定 (属性指定の結果を使用)
=> 0
選択会話:0081 0082 3045 3046
ldbsh:
```

会話属性指定 (1) を選択した場合、次のような画面が表示されるので、画面から対象会話を選択する。

(例)

<< 会話属性選択 >>	
対象会話数: 15	
<属性名>	<属性>
メディア	: キーボード <input checked="" type="checkbox"/> 電話
領域	: <input checked="" type="checkbox"/> 国際会議
内容	: <input checked="" type="checkbox"/> 問合せ <input type="checkbox"/> 依頼・説得 <input type="checkbox"/> 相談・交渉
言語パターン	: <input checked="" type="checkbox"/> 英・日 <input type="checkbox"/> 日 <input type="checkbox"/> 英 英・日 (同)

↑
項目

_____ がカレントな項目である。

各項目はトグルになっており、リターンキーを押せば、選択/非選択が切り替わる。

選択中の項目が反転している。

選択中の対象会話数が右上に表示される。

カレントな項目は、矢印キー (→←↑↓) で移動する。(矢印キーで移動しない場合、システムのtermcapファイルが正しくないのでシステム管理者に修正を依頼して下さい)

任意の項目上で"q"キー入力すれば選択終了。

会話番号指定（2～4の違いは後で説明する）を選択した場合、次のような画面が表示されるので、画面から対象会話を選択する。

（例）

<< 会話番号選択 >>			
会話 I D	発話数	文数	単語数
0081	50	87	421
0082	37	62	312
0083	43	71	375
3021	50	87	421
3022	37	62	312
.			
.			

← 項目

_____ がカレントな項目（行）である。

各項目はトグルになっており、リターンキーを押せば、選択／非選択が切り替わる。

選択中の項目（会話）が反転している。

カレントな項目は、矢印キー（↑↓）で移動する。（矢印キーで移動しない場合、システムのtermcapファイルが正しくないのでシステム管理者に修正を依頼して下さい）

CTRL+vで次の頁へ、ESC vで前の頁へ移動する。

（CTRL+vはCTRLキーを押しながらvキーを押す。ESC vはESCキーを押してからvキーを押す）

任意の項目上で”q”キー入力すれば選択終了。

2から4の違いは起動時の初期値の違いである。

2は初期化し、起動時に選択されているものはない。

3は前回の「会話番号指定」の結果を初期値とする。

4は「会話属性指定」の結果を初期値とする。

4. 2 検索

検索コマンドとしてSQL以外に2種類のコマンドが用意されている。

1つは検索パターンを指定して、検索実行をする「パターン指定検索コマンド」、他の1つは、単語条件の並びを検索条件とする「単語条件列検索コマンド」である。

また、これらをサポートするためのコマンドがいくつか用意されている。

4. 2. 1 パターン指定検索コマンド (psearch)

<コマンド形式>

psearch 検索パターンファイル名 [パターン名]

検索パターンファイル (パターン名) を指定して検索実行する。

パターンファイルというのは検索パターンが1つ以上定義されているファイルである。

パターン名を指定しない場合、そのパターンファイル中で定義されているパターン名の一覧が表示されるので、その中から選択、実行する。

検索パターンファイルは、パターンエディタ (editp)、または通常のエディタであらかじめ作成しておく。

検索終了後、検索結果は標準出力 (デフォルトは端末) に出力される。

パターンエディタについては、editpコマンド及び「パターンエディタ操作説明書」を参照、検索パターンについては、「DEFSEARCH文法説明書」を参照。

(例)

```
% cat patterns1
(DEFSEARCH P1 "は+++する"
 (OBJECT-PATTERN (VAR 単語1)
  ((INSTANCE-OF 単語)
   (出現単語 は)
   (AFTER (VAR 単語2))))
 (OBJECT-PATTERN (VAR 単語2)
  ((INSTANCE-OF 単語)
   (出現単語 する)))
 (RETURN-FORM 文 0 0))
```

```
% ldbsh
```

```
ldbsh: psearch patterns1 P1
```

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

```
ldbsh: psearch patterns1
```

0. キャンセル

1. P1 は+++する
番号選択して下さい=>1

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

```
ldbsh: end
```

```
%
```

4. 2. 2 単語条件列検索コマンド (wsearch)

<コマンド形式>

wsearch 検索文

検索条件を単語レベルに限定し、単語条件の並びを検索条件とするコマンドである。

検索文は単語条件の並びを記述したものであり、“+”で接続される。
(+ は NEXT-IS と考える。)

検索文中のブランクコードは無視される。

1つの単語の条件は“:”で条件を記述し、“/”で条件の区切りとする。

(例) 品詞が本動詞で、活用型が五段活用、活用形が終止形の単語
品詞:本動詞/活用型:五段/活用形:終止

”出現単語:”は省略してもよい。

(例)

は/品詞:格助詞 <=> 出現単語:は/品詞:格助詞
等価

or 条件は次のように記述する。

(例)

活用形:(or (終止 連体))

単語の並びを記述するため、以下のような記号がある。

? . . . 任意の1単語

* . . . 任意の単語 (0単語以上)

(検索文例)

は/品詞:格助詞 + * + する

品詞:(or (本動詞 助動詞))/活用型:五段/活用形:終止 + * + 読み:か

(注)

区切り子 (:, /, ?, *, (, or など) はすべて半角 (ASCII) コードでなければならない。

(例)

```
% ldbsh
```

```
ldbsh: wsearch は/品詞:格助詞 + * + する
```

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

```
ldbsh: wsearch は/品詞:格助詞 + * に + * + する
```

1. 太郎はその会議に参加する。

```
ldbsh: end
```

```
%
```

4. 2. 3 SQL (sql)

<コマンド形式>

sql [SQLスクリプトファイル名]

UNIFYのSQLコマンドを実行する。
SQLスクリプトファイルを指定して実行することもできる。

(例)

```
% ldbsh
ldbsh: sql

UNIFY TURBO/SQL
Copyright .....

sql>

[SQLの実行]

sql> end
ldbsh: sql SQLスクリプトファイル名
問い合わせを認識しました。

会話 I D | 単語 I D | ...
      3021 |      100 | ...
      .
      .

ldbsh:
ldbsh: end
%
```

4. 2. 4 その他検索サポートコマンド

(1) editp

<コマンド形式>

editp [パターンファイル名 [パターン名]]

psearchコマンドで使用する検索パターンを作成するための支援モジュールである。

パターンファイルは検索パターンが1つ以上定義されているファイルである。(検索パターンについては「DEFSEARCH文文法説明書」参照)

パターンファイル名を指定した場合、そのパターンファイルの中で定義されているパターン名の一覧が表示されるので、その中から選択しエディットする。

パターン名まで指定した場合、そのパターンをエディットする。

詳細については「パターンエディタ操作説明書」を参照。

(2) editw

<コマンド形式>

editw

直前に実行した検索文のエディットを行う。

これは、直前に実行した検索文がテンポラリファイルへ出力されるので、これをエディタで修正する。

エディタで修正した後、restartw で再実行する。

(参照、restartw、UNIFY、SQLのedit)

(例)

```
% ldbsh
ldbsh: wsearch は/品詞:格助詞 ++ する
文法エラーです。
ldbsh: editw

[ (通常の) エディタで修正 ]

ldbsh: restartw
は/品詞:格助詞 + * + する

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

ldbsh: end
%
```

(注)

どの様なエディタを使用するかは、環境変数 EDIT で設定可能。デフォルトはjemacsエディタである。

(3) restartw

<コマンド形式>

restartw

直前に実行した検索文、または、editw後の検索文を再実行する。

(参照、editw、UNIFYのSQLのrestart)

(4) savew

<コマンド形式>

savew [-p] ファイル名

直前の検索文をファイルへセーブする。

このセーブした検索文ファイルをstartwコマンドの引数として使用する。

-p オプションをつけた場合、直前の検索文を検索パターン (DEFSEARCH文) に展開してファイルへセーブする。

セーブした検索パターンをパターンエディタなどで修正し、それをpsearchコマンドの引数として使用する場合などに使用する。

(参照、startw、psearch)

(5) startw

<コマンド形式>

startw ファイル名

検索文をあらかじめファイルに作成(セーブ)しておき、そのファイル名を指定して、「単語条件列検索」を行う。

ファイル中には1文(1行)の検索文しか記述できない。

(参照、savew、UNIFY、SQLのstart)

(例)

```
% ldbsh
ldbsh: wsearch は/品詞:格助詞 + * + する

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

ldbsh: savew pattern1
ldbsh:

          [その他のコマンド実行]

ldbsh: startw pattern1

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

ldbsh: end
%
```

(6) 結果のセーブ

検索結果をファイルへセーブするときは、リダイレクション記号">"を使用する。追加する場合は、">>"とする。

リダイレクションで実行した場合、検索結果ファイルはデーモンが(作成)書き込みを行うので、そのディレクトリ、ファイルには適切なパーミッションが与えられていなければならない。

(例)

```
% ldbsh  
ldbsh: wsearch は/品詞:格助詞 + * + する > kakkal  
ldbsh: end  
% cat kakkal
```

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

```
%
```

(7) バックグラウンドでの検索実行

バックグラウンドで検索実行するときは、記号"&"を使用する。
バックグラウンドで実行したものは、endコマンドでldbshを終了しても（さらにlogoutしても）実行される。

(例)

```
% ldbsh
ldbsh: wsearch は/品詞:格助詞 + * + する > kakka2 &
ldbsh:
      .
      . [その他のコマンド]
      .
      .
検索 "は/品詞:格助詞+++する(kekka2)" が終了しました。
ldbsh: !cat kekka2

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

ldbsh: end
%
```

(注)

"&"は、ファイルへの出力">"または、">>"とペアでしか使用できない。（バックグラウンドで実行し、結果を直接画面に出力することはできない）

(8) status

<コマンド形式>

status

バックグラウンドで実行中のプロセスの状態を知る。

但し、これでわかるのはldbshに入ったあと、自分がバックグラウンドで実行したものだけである。他人のプロセスはもちろん、自分の実行したものであっても一度endコマンドでldbshを終了すると、その状態を知ることができない。

また、ここでいう状態とは「EXECUTING」か「WAITING」のいずれかである。

(例)

```
% ldbsh
ldbsh: psearch patterns P1 > kekka1 &
ldbsh: wsearch は/品詞:格助詞 + * + する > kekka2 &
ldbsh: status
SEQ_ID  COMMENT                OUTPUT  STATUS
00004   是/品詞:格助詞+*+する kekka1  EXECUTING
00006   は/品詞:格助詞+*+する kekka2  WAITING

ldbsh: end
%
```

4. 3 結果処理 (mode)

<コマンド形式>

mode

結果の処理については、“検索モード”の概念でセットする。
即ち、1度セットすると、modeコマンドで再びセットするまで、その値が有効である。

modeコマンドを実行すると、次のような画面が表示される。

<< モード設定 >>			
言語	: 日本語	英語	
ユニーク処理	: する	しない	
統計処理	: する	しない	
ソート処理	: する	しない	
リターンレベル	: 発話	文	文節 単語

↑
項目

■がカレントな項目である。

各項目はトグルになっており、カレント項目を矢印キー(→←)で移動すれば選択項目が切り替わる。

選択中の項目が反転している。

各行間は、矢印キー(↑↓)で移動。その時反転している項目がその行のモードとなる。

任意の項目上で“q”キー入力すればモード設定終了。そのとき反転しているモードが以降の検索で使用される。

(注)

ユニーク、統計指定時(「する」とした時)はソートのモード指定は無視され、強制的にソート「する」となる。

また、このモードが有効なのは、DEFSEARCH 文で“RETURN-FORM”を指定した場合のみである。

4.4 ヘルプ (help)

<コマンド形式>

help

または

help DB [テーブル名 [フィールド名]]

引数がない場合、コマンドの一覧が表示される。

引数 DB をつけた場合、データベースに定義されているテーブル名、フィールド名、データとして許される値の一覧が表示される。

(例)

```
% ldbsh
ldbsh: help [コマンドの一覧]

editp      パターンエディット
editw      単語検索文字列エディット
.
.
.
wsearch    単語検索

ldbsh: help DB [テーブル名の一覧]

会話      発話      文      文節
単語
.
.

ldbsh: help DB 単語 [フィールド名の一覧]

* 会話 I D      * 単語 I D      * NEXT_IS      * PREV-IS
  出現単語      読み          標準表現      $ 音便
$ 品詞          $ 活用型      $ 活用形
.
.

ldbsh: help DB 単語 活用形 [データ値の一覧]

未然      連用      終止

ldbsh: end
%
```

(注)

フィールド名一覧で"*"はポインター型データ、"\$"はメンバーデータを持つフィールドである。

フィールド名まで指定して意味があるのは、その値があるメンバーの1つである場合のみ、つまり、"\$"がつくフィールドのみである。

help 単語 出現単語

などは意味がないので、実行しても何も出力されない。

4. 5 その他

4. 5. 1 UNIXコマンドの実行

UNIXコマンドは、“!”でエスケープして実行する。

(例)

```
% ldbsh
ldbsh: wsearch は/品詞:格助詞 + * + する > kakkai
ldbsh: !cat kekka1

1. 太郎はその会議に参加する。
2. 次郎は読書をする。

ldbsh: !lpr kekka1
ldbsh: end
%
```

4. 5. 2 終了 (end)

<コマンド形式>

end

実行環境のセーブを行い、ldbshを終了する。

ここでいう実行環境とは、

検索範囲 (番号と属性)
各種モード

である。

5. 単語条件列検索文法

単語条件文の文法をBNF的記述法で示す。

```
<<検索文>> ::= <<単語条件>>{<<接続子>><<単語条件>>}*
<<接続子>> ::= <<スハ°-ス>>* + <<スハ°-ス>>*
<<単語条件>> ::= <<フィールド条件列>> | * | ?
<<フィールド条件列>> ::= <<フィールド条件>>{/<<フィールド条件>>}*
<<フィールド条件>> ::= フィールド名:<<値条件>>
<<値条件>> ::= リテラル | (or (リテラル*))
<<スハ°-ス>> ::= スハ°-ス
```

第3部

パターンエディタ操作説明書

第1.0版

目 次

	頁
1. 概要	1
2. ウィンドウ構成	4
2. 1 メインウィンドウ	5
2. 2 オブジェクトパターンエディタウィンドウ	8
2. 3 ポップアップメニューウィンドウ	11
3. コマンドの起動、終了	12
3. 1 起動	12
3. 2 終了	14
4. 各ウィンドウにおける操作について	16
4. 1 メインウィンドウ	16
4. 2 オブジェクトウィンドウ	25
5. スロット条件のエディット	28
5. 1 即値データを持つスロット	29
5. 2 メンバーデータを持つスロット	30
5. 3 シングルポインター型のデータを持つスロット	31
5. 4 疑似マルチポインター型のデータを持つスロット	36
6. リターン形式のエディット	40
6. 1 変数指定	41
6. 2 レベル指定	42
6. 3 統計	42

1. 概要

本書では、検索要求記述支援ツール（パターンエディタ）の操作について説明する。

パターンエディタは、問合せ言語DEFSEARCHを使って検索要求を書くための簡易ツールであり、LDBSH上で利用する。その基本的な仕様は、利用者がオブジェクトウィンドウと呼ばれる表の適当な場所に検索条件を書き込むことによって、DEFSEARCH検索パターンを生成するというものである。パターンエディタを用いれば、DEFSEARCH文を直接書くことなく、必要最小限の記述で検索パターンを作成することができる。

[コマンド]

説明の際、いくつかのコマンドが出てくるので、コマンドとキーボードオペレーションの関係を先にまとめておく。

コマンド名称	キー	備考
エディット	e	
デリート	d	
セーブ	s	
クイット	q	
変数入力	v	
再表示	CTRL+l	画面再表示
ヘルプ	?	カレント行のヘルプ
カレント行移動	↓	カレント行を次の行へ
カレント行移動	↑	カレント行を前の行へ
カレント行移動	t	カレント行を最上行へ
カレント行移動	b	カレント行を最上行へ

[制限]

DEFSEARCH文で許されていて、パターンエディタで使用できないものがある。

- (1) RETURN-FORMにおける範囲指定パラメータ
- (2) スロット条件の NOT 文

これらを含むパターンをエディットした場合の結果については保証しない。

主なオペレーションフローを図1-1に示す。

また、この操作説明書で使用する典型的な検索パターン（DEFSEARCH文）を図1-2に示す。図1-2以外にも用いるが、それについてはその都度そこで示す。

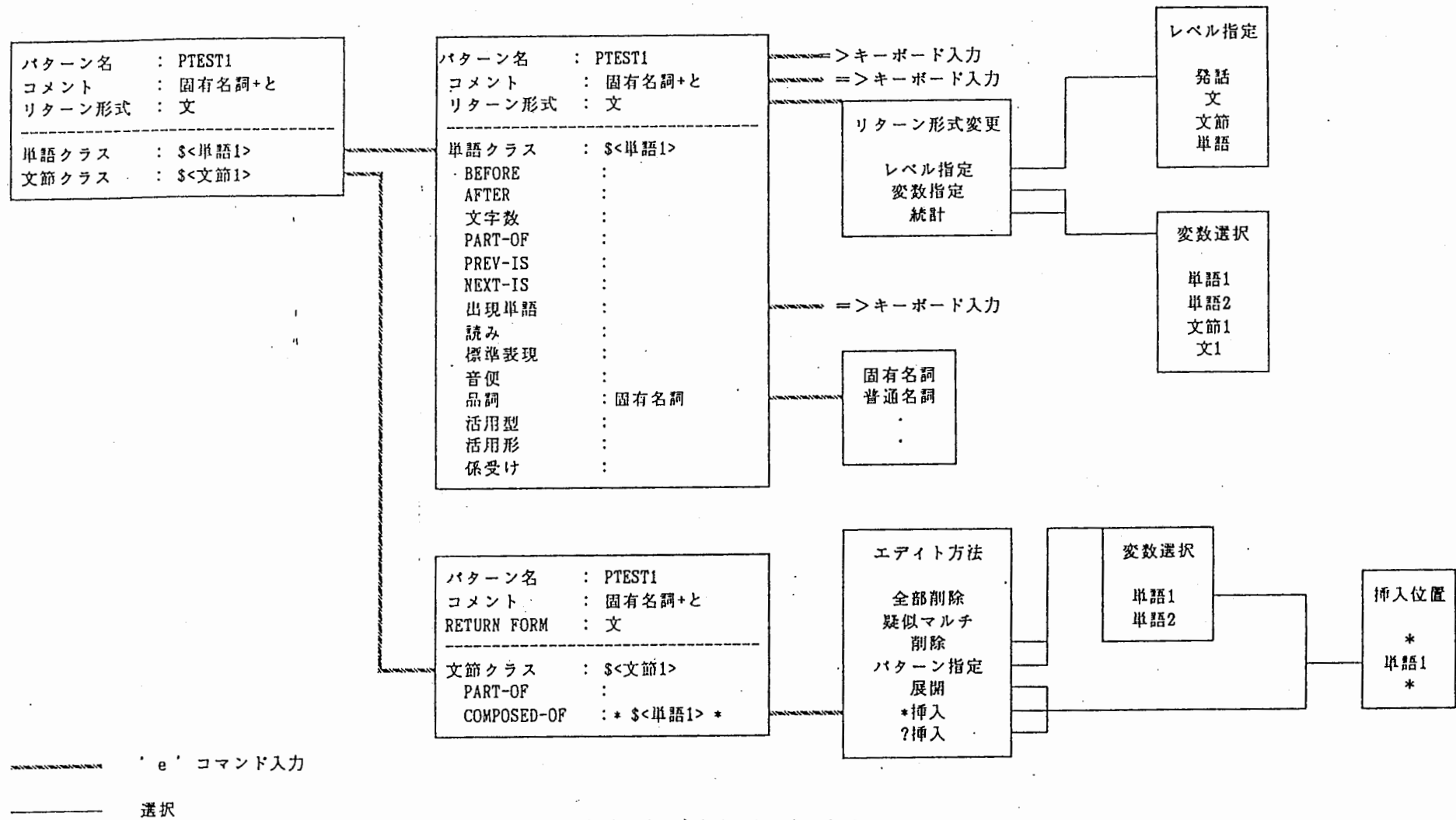


図1-1 主なオペレーションフロー

(注) エディットのオペレーションフロー (デリートなどは除く)
便宜上、メニュー項目の並びの順序を、実際のもの入れ換えたものもある
また変数などは仮のものである。


```

(DEFSEARCH PTEST1 "固有名詞+と"
(OBJECT-PATTERN (VAR 単語1)
((INSTANCE-OF 単語)
(PART-OF (VAR 文節1))
(NEXT-IS (VAR 単語2))
(品詞 固有名詞)))
(OBJECT-PATTERN (VAR 単語2)
((INSTANCE-OF 単語)
(PART-OF (VAR 文節1))
(PREV-IS (VAR 単語1))
(出現単語 と)))
(OBJECT-PATTERN (VAR 文節1)
((INSTANCE-OF 文節)
(COMPOSED-OF * (VAR 単語1) (VAR 単語2) *)))
(RETURN-FORM 文))

```

図 1 - 2 説明で使用する DEFSEARCH 文

2. ウィンドウ構成

パターンエディタは大きく分類して、次の3つのウィンドウ（群）で構成される。

- (1) メインウィンドウ
- (2) オブジェクトパターンエディタウィンドウ
- (3) ポップアップメニューウィンドウ

ここでは、これらのウィンドウについて説明する。

2. 1 メインウインドウ

メインウインドウはパターンエディタのベースになるウインドウであり、その全体図は、次のようなものである。

<< メインウインドウ >>	
パターン名	: PTEST1 (1)
コメント	: 固有名詞+と (2)
リターン形式	: [レベル] 文 (3)

<u>単語</u>	: \$<単語1> (4)
単語	: \$<単語2>
文節	: \$<文節1>

図 2 - 1 メインウインドウ

メインウインドウには

- 検索パターン名 (1)
- コメント (2)
- リターン形式 (3)
- 検索パターンに含まれるオブジェクト変数名 (4)

が表示される。

- 各行はタグ、データで構成される。
- コマンド入力可能な行 (カレント行) を下線表示している。
- カレント行の先頭に仮想的なコマンド入力フィールドがある。
- コマンドは1バイト文字か矢印キー (↑↓) である。
- カレント行の移動は矢印キーで行う。

画面上の変数の表示、リターン形式の表示と、DEFSEARCH文での変数、リターン形式の対応は次のとおりである。

変数

DEFSEARCH文	画面
(VAR 単語1) (MVAR 単語1) (OR A B)	\$<単語1> M\$<単語1> A B

リターン形式

DEFSEARCH文	画面
RETURN RETURN-FORM RETURN-STAT	変数指定 レベル指定 統計

以下、便宜上、検索パターン名、コメント、リターン形式の3行を「ヘッダ部」、(4)以降の行を「ボディ部」総称することがある。(2.2参照)

また、オブジェクト変数名の行(4)をオブジェクト変数行と呼ぶこともある。

メインウインドウで使用可能なコマンドは次の通りである。

なお、実行するコマンドによっては画面の最下部に確認メッセージ、警告メッセージがでることもある。

確認メッセージは必ず "Y/N?" で終わっており、ユーザの入力を待つ。

警告メッセージは、ユーザへの警告(及び、これから行うオペレーションのガイダンス)であり、しばらく(約1秒)表示した後、自動的に消える。

○ メインウィンドウ共通コマンド

キー	動作	備考
q	パターンをセーブしないで終了	確認後終了
s	パターンをセーブして終了	確認後終了
↓	カレント行を次の行へ	
↑	カレント行を前の行へ	
t	カレント行を最上行へ	
b	カレント行を最下行へ	
CTRL+l	画面再表示	
?	カレント行のヘルプ	

○ ヘッダ部コマンド

キー	動作	備考
e	検索パターン名などのエディット	

○ ボディ部コマンド

キー	動作	備考
e	オブジェクト条件のエディット	オブジェクト条件の編集をする
d	オブジェクト条件の削除	確認後削除

○ 行データエディット時

キー	動作	備考
CTRL+c	文字入力モードへ	CRで確定
CR	行エディット終了	
←	1文字左へ	
→	1文字右へ	
DEL	1文字削除	
BS	前文字削除	
CTRL+u	行データ削除	カーソルはデータ部先頭位置へ

(注) 検索パターン名、コメントのエディット時のみ
(リターン形式はメニューオペレーション)

各コマンドの詳細は4. 1、及び6. で説明する。

2.2 オブジェクトパターンエディタウインドウ

オブジェクトパターンエディタウインドウは、メインウインドウ中でエディトコマンド、またはデリートコマンドを実行することにより作成されるウインドウで、その全体図は次のようなものとなる。

<< オブジェクトウインドウ >>	
パターン名	: PTEST1 (1)
コメント	: 固有名詞+と (2)
リターン形式	: [レベル] 文 (3)

単語	: \$<単語1> (5)
BEFORE	:
AFTER	:
文字数	:
PART-OF	: \$<文節1>
PREV-IS	:
NEXT-IS	: \$<単語2>
出現単語	:
読み	:
標準表現	:
音便	:
品詞	: 固有名詞 (6)
活用型	:
活用形	:
係受け	:
単語対応	:
ランダム対応	:
日本語格対応	:

図2-2 オブジェクトパターンエディタウインドウ

オブジェクトパターンエディタウインドウには、

- 検索パターン名 (1)
- コメント (2)
- リターン形式 (3)
- オブジェクト変数 (5)
- オブジェクト条件を展開表示したもの (6)

が表示され、オブジェクト条件をエディット可能にしたものである。

このウインドウはメインウインドウを消去 (clear) した後、作成される。通常、1つのオブジェクト条件のエディット終了後、メインウインドウが再表示される。

各行はタグ、データで構成される。
コマンド入力可能な行（カレント行）を下線表示している。
カレント行の先頭に仮想的なコマンド入力フィールドがある。
コマンドは1バイト文字、または矢印キー（↑↓）である。
カレント行の移動は矢印キーで行う。

以下、便宜上、検索パターン名、コメント、リターン形式の3行を「ヘッダ部」、オブジェクト変数の行（5）を「オブジェクト変数行」、それ以降の行を「ボディ部」総称することがある。（2.1参照）

また、オブジェクトパターンエディタウインドウを「オブジェクトウインドウ」と略称する。

なお、説明で用いる図は、説明に直接関係のない一部のスロット名を省略することがある。

オブジェクトウインドウで使用可能なコマンドは次の通りである。

実行するコマンドによっては画面の最下部に確認メッセージ、警告メッセージがでることもある。

○ オブジェクトウィンドウ共通コマンド

キー	動作	備考
q	オブジェクト条件エディットを終了	メインウィンドウへ戻る
↓	カレント行を次の行へ	
↑	カレント行を前の行へ	
t	カレント行を最上行へ	
b	カレント行を最下行へ	
CTRL+l ?	画面再表示 カレント行のヘルプ	

○ ヘッダ部コマンド

キー	動作	備考
e	検索パターン名などのエディット	

○ オブジェクト変数行コマンド

キー	動作	備考
d	オブジェクト条件の削除	確認後削除

○ ボディ部コマンド

キー	動作	備考
e	スロット条件エディット	行データエディット (注)
d	スロット条件の削除	
v	変数入力	行データエディット

○ 行データエディット時

キー	動作	備考
CTRL+c	文字入力モードへ	CRで確定
CR	行エディット終了	
←	1文字左へ	
→	1文字右へ	
DEL	1文字削除	
BS	前文字削除	
CTRL+u	行データ削除	

(注) 検索パターン名、コメント、及び即値データを持つスロットの場合のみ
(リターン形式、ポインターデータのスロットはメニューオペレーション)

各コマンドの詳細は4. 2.、5.、及び6. で説明する。

2. 3 ポップアップメニューウインドウ

ポップアップメニューウインドウはメインウインドウ、オブジェクトウインドウの左上に一時的にオーバーレイするものであり、いくつかの選択枝から、1つ、または複数個の項目を選択するためのウインドウである。

通常、選択終了後、メインウインドウ、または、オブジェクトウインドウが再表示される。

以下、便宜上ポップアップメニューウインドウを「メニュー」と略称する。

メニューには1つ選択する”単一選択メニュー”と、複数選択可能な”複数選択メニュー”があるが、特に断わらない限り”単一選択メニュー”である。

メニュー選択時のコマンドは次の通りである。

○ メニュー選択時

キー	動作	備考
CR	選択	単一の時は同時にメニュー終了
↓	下の項目へ移動	
↑	上の項目へ移動	
←	左の項目へ移動	
→	右の項目へ移動	
q	メニュー選択終了	メニューオペレーションキャンセルとなる
CTRL+u	キャンセル	

(注)

メニューには”EXIT”が必ず含まれている。
これを選択すると、

単一選択のメニューはキャンセル(CTRL+u)
複数選択のメニューは選択終了(CR)

となる。

3. コマンドの起動、終了

3.1 起動

パターンエディタは検索シェル `ldbsh`、または通常のシェル環境から起動できる。

起動方法はどちらも同じで次のように入力する。

```
editp [パターンファイル名 [検索パターン名]]
```

ここで、パターンファイル名、検索パターン名はオプションである。(オプションを指定する場合、“[]”は不要)

パターンファイルは検索パターン(DEFSEACRH文)が1つ以上定義されているファイルで、通常、このパターンエディタにより作成される。

(1) オプションがない場合

パターンエディタは新規に検索パターン(パターンファイル)を作成するものとして実行を開始する。

この場合、次のようなメッセージがでるので、検索パターンのベースとなる(とりあえずの)テーブル名(クラス名)を入力しなければならない。

```
ベースのテーブル名を入れて下さい [デフォルト: 単語]
```

単にリターンキーを押した場合、単語クラスがベースとなる。

(2) パターンファイル名のみを指定した場合

まず、次の例のように既存のパターンファイルに定義されている検索パターン名、コメントの一覧を表示し、その中から検索パターンを選択しエディットする。

(例)

```
% editp patterns
0. 新規作成
1. PTEST1      固有名詞+と
2. PTEST2      を+++する
3. PTEST3      本動詞+たい
番号選択してください: 1      PTEST1のエディット
```

ここで、0(新規作成)を選択すると、セーブする際のデフォルトファイル名がセットされることを除けば、オプションを指定しなかった場合と同じである。

(3) パターンファイル名、検索パターン名を指定した場合

既存のパターンファイルから指定された検索パターンを検索しエディットする。

(4) 起動時のエラー

パターンファイルを指定して、そのファイルが存在しなかった場合、次のようなメッセージが出力される。

```
ファイル yyy y が存在しません。  
新規作成で継続しますか? (Y/N)
```

"Y" と入力すると、セーブする際のデフォルトファイル名がセットされることを除けば、オプションを指定しなかった場合と同じである。"N" と入力するとパターンエディタは終了する。

パターンファイル、検索パターン名を指定し、指定された検索パターンがそのファイル中になかった場合、次のようなメッセージが出力される。

```
パターン xxxx はファイル yyy y の中に定義されていません。  
新規作成で継続しますか? (Y/N)
```

この場合も、"Y" を入力すると、パターン名、セーブする際のデフォルトファイル名がセットされることを除けば、オプションを指定しなかった場合と同じである。"N" を入力するとパターンエディタは終了する。

また、パターンファイルを指定し、ファイルの内容が正しくない場合、なんらかのエラーメッセージ（これはケースにより数多くある）に続き次のようなメッセージがでることがある。（例えばパターンファイルでなく一般のファイルだった場合や、パターンファイルを通常のエディタで作成、修正した場合などに起こり得る）

```
ファイル読み込み中にエラーが発生しました。継続しますか? (Y/N)
```

このような場合、"Y" と入力すれば、そのファイル中のエラー発生までに読み込めた（パースできた）検索パターンのみが有効となり、処理を続行することが可能である。"N" を入力するとパターンエディタは終了する。

3. 2 終了

パターンエディタはメインウィンドウでセーブコマンド (s)、またはクイットコマンド (q) を入力することで終了する。

どちらのコマンドも終了する前に、次のような確認メッセージを出力する。

```
パターンをセーブして (/しないで) 終了します。 Y/N ?
```

ここで、"Y" を入力すれば実際に終了プロセスにはいる。

クイットコマンドで終了すると、editpコマンドを全く実行しなかったのと同じである。ファイル、その他、実行環境に影響を及ぼさない。

ここではセーブの際の手順を説明する。

セーブでは、上のメッセージを出す前に、まず、エディット中の検索パターンがセーブできる状態かチェックし、セーブできない場合、メッセージが表示される。これについては4. 1 (4) で説明する。

セーブ可能な場合、上のメッセージがでる。

さらに、"Y" を入力すると、次のようなメッセージがでる。

```
パターン xxxx をセーブするファイル名を入れて下さい
```

```
[ファイル名: yyyy]
```

ここで xxxx はメインウィンドウ (オブジェクトウィンドウ) で定義した検索パターン名、 yyyy は起動時に指定したファイル名 (オプションなしで起動した場合ヌル) である。

起動時と別のファイルにセーブしたいときは、ここでファイル名を入力すればよい。但し、有効なファイル名はアルファニューメリック (a-z, A-Z, 0-9) とアンダーバー (_)、ピリオド (.) で構成されるものである。ディレクトリ付きのファイル名指定も可能である。

次に、もし指定したファイル中に同じ検索パターン名があった場合、次のようなメッセージがでる。

```
ファイル yyyy には同じ名前のパターンがあります。
```

```
その内容が置き換えられますがいいですね? (Y/N)
```

"Y" を入力すると、その検索パターンの内容が置き換えられる (ファイル中の他のパターン定義は残る) 。"N" を入力すると、もう一度ファイル名入力を要求する。

以上の手順によりエディットした検索パターンが指定したパターンファイルへセーブされる。

なお、検索パターン名が既存のものでない場合、パターンファイルの最後に追加される。

(1) セーブ時のエラー (ファイル指定の誤り)

セーブするパターンファイルは次のいずれかの条件を満足するものでなければならない。

- 既存ファイルの場合、読み書き可能 (パーミッション) であり、かつ、パターンファイルとしてファイルの内容が正しい。
- 新規ファイルの場合、有効なファイル名であり、作成可能 (パーミッション) であること。

4. 各ウインドウにおける操作について

ここでは、各ウインドウにおける操作について説明する。

4. 1 メインウインドウ

メインウインドウはパターンエディタのベースとなるウインドウである。

パターンエディタの起動処理の後、新規パターンの場合はベースのオブジェクト条件の変数が1つ(図4-1)、既存の検索パターンを指定した場合は、その中で定義されているオブジェクト条件の変数(図4-2)がメインウインドウが表示される。

新規で作成した場合、リターン形式は"レベル指定"の"文"がデフォルトとなる。

<< メインウインドウ >>	
パターン名	:
コメント	:
リターン形式	: [レベル] 文

単語	: \$<単語1>

図4-1 メインウインドウ (新規: ベース単語)

<< メインウインドウ >>	
パターン名	: PTEST1
コメント	: 固有名詞+と
リターン形式	: [レベル] 文

単語	: \$<単語1>
単語	: \$<単語2>
文節	: \$<文節1>

図4-2 メインウインドウ (既存)

次に、メインウインドウで使用可能なコマンドについて説明する。

(1) ~ (4) は、カレント行がどこにあっても使用できる。

(5) はパターン名行、(6) はコメント行、(7) はリターン形式行のエディット、(8) (9) はオブジェクト変数行でのコマンドである。

(1) カレント行の移動

メインウィンドウが開いた時にはカレント行はオブジェクト条件リストの先頭になっている。

カレント行の移動は次の4つのコマンドで行える。

↓	カレント行を次の行へ
↑	カレント行を前の行へ
t	カレント行を最上行へ (検索パターン名の行)
b	カレント行を最下行へ

(2) 画面の再表示

パターンエディタを使用中に、なんらかの理由で (例えば Mail が飛び込んで来た場合など) 画面が乱れることがある。このような場合、次のコマンドで画面を再表示できる。

CTRL+l	画面再表示
--------	-------

(3) ヘルプ

カレント行で実行できるコマンドはヘルプコマンドで確認できる。

?	ヘルプコマンド
---	---------

ヘルプコマンドを実行するとメインウィンドウが消え、ヘルプウィンドウが開く。(図4-3)

<< ヘルプウィンドウ >>	
e:	コメントエディット
q:	パターンをセーブしないで終了
s:	パターンをセーブして終了
t:	カレント行を最上行へ
b:	カレント行を最下行へ
↑:	カレント行を上の方へ
↓:	カレント行を下の方へ
CTRL-L:	再表示
任意のキーを押して下さい	

図4-3 ヘルプウィンドウ (コメント行の例)

任意のキーを押せば元のウィンドウ (メインウィンドウ) が再表示される。

(4) クイットとセーブ

クイットコマンドかセーブコマンドを実行することにより、パターンエディタから抜けることができる。

q	パターンをセーブしないで終了
s	パターンをセーブして終了

どちらのコマンドも実際に終了する前に図4-4のように確認メッセージを表示する。

```
<< メインウインドウ >>

パターン名      : PTEST1
コメント        : 固有名詞+と
リターン形式    : [レベル] 文
-----
単語            : $<単語1>
単語            : $<単語2>
文節            : $<文節1>

パターンをセーブしないで終了します Y/N ?
```

図4-4 確認メッセージ (クイットの例)

ここで、“Y”を入力すれば実際に終了プロセスにはいる。“N”であればメインウインドウでの作業を継続できる。

クイットコマンドを実行すると、パターンエディタでの全ての作業は無視され、パターンエディタを終了。

セーブコマンドは、確認メッセージを出す前に、セーブ可能かどうかチェックを行う。

セーブできない状態として次の2つの状態がある。

1. 検索パターン名が未定義
2. リターン形式が、変数指定、統計の場合で、リターン変数のリスト中にオブジェクト条件リストで定義されないものが参照されている。

このような場合、それぞれ次のようなメッセージがでる。

パターン名が未定義です。

リターン変数の指定に正しくないところがあります。

どちらのケースも正しい状態、つまり

検索パターン名を入力
リターン変数の修正

を行った後、もう一度セーブコマンドを実行しなおさなければならない。検索パターン名の入力（５）、リターン変数の修正（７）で説明する。

(5) 検索パターン名のエディット

カレント行がパターン名の位置にある時、エディットコマンドにより、検索パターン名の変更ができる。

e	エディットコマンド
---	-----------

検索パターン名は英数字、漢字コードで構成される任意の文字列であるが、アルファベットについては小文字を入力しても大文字となる。(小文字、大文字は区別できない)

検索パターン名のエディットに限らず、行データエディットは、そのデータの表示位置でデータ入力が行える。以後、特に明記しない限り同様である。

エディットコマンドを実行するとカーソルがデータ部へ移動する。(図4-5)

<< メインウインドウ >>	
パターン名	: █
コメント	:
リターン形式	: [レベル] 文

単語	: \$<単語1>

図4-5 行データエディット (パターン名の例)

行データエディット中に使用できるコマンドは次の通りである。

キー	動作	備考
CTRL+c CR ← → DEL BS CTRL+u	文字入力モードへ 行エディット終了 1文字左へ 1文字右へ 1文字削除 前文字削除 行データ削除	CRで確定 カーソルはデータ部先頭位置へ

矢印キー（←→）で文字を挿入したい位置に移動し移動し、CTRL-Cで入力モードに入り、文字列を入力後 CR（キャリッジリターン）で確定させ、もう一度 CR を押して行エディットを終了する。（入力モードでの漢字変換については V A X のマニュアルを参照）

文字の削除は BS（バックスペース）、DEL（デリートキー）、CTRL+uで行う。BS はカーソルの直前の文字、DEL はカーソル上の文字、CTRL+u はデータ部の文字列全てを削除する。

なおここでいう BS、DEL は端末が発生するコードである。従って、sty の設定と合わないこともある。

(6) コメントのエディット

カレント行がコメントの位置にある時、エディットコマンドにより、コメントのエディットができる。

コメントのデータ部には1行に納まる範囲で任意のコメントを記述できる。検索パターンの内容を端的に表すようなコメントを入れればよい。コメントを入れなくても DEFSEARCH 文としては正しい。

エディット方法は検索パターン名と同じである。(5) 参照。

(7) リターン形式のエディット

便宜上、6. で説明する。

(8) オブジェクト条件のエディット

オブジェクト変数行でエディットコマンドを実行すると、オブジェクトウィンドウが開き、個々のオブジェクト条件のエディットが可能となる。(4.2、及び5.参照)

(9) オブジェクト条件のデリート

オブジェクト変数行でデリートコマンドを実行すると、そのオブジェクト条件が削除される。

d	デリートコマンド
---	----------

但し、確認のため、オブジェクト条件の内容をオブジェクトウィンドウを表示し、ユーザに確認を求めた後、削除する。図4-6参照。

<< 削除 >>	
パターン名	: PTEST1
コメント	: 固有名詞+と
リターン形式	: [レベル] 文

単語	: \$<単語1>
BEFORE	:
AFTER	:
文字数	:
PART-OF	: \$<文節1>
PREV-IS	:
NEXT-IS	: \$<単語2>
出現単語	:
読み	:
標準表現	:
音便	:
品詞	: 固有名詞
活用型	:
活用形	:
係受け	:
削除します	Y/N ?

図4-6 オブジェクト条件の削除(確認)

削除を実行すると、オブジェクト条件のリストから対象オブジェクト条件を削除するのみでなく、そのオブジェクト条件の中で使用されていた変数が他のオブジェクト条件で参照されていた場合、それをどうするかユーザに確認する。(参照、4.2(9))

例えば図4-6のオブジェクト条件の場合、図4-7のような確認メッセージがでる。(DEFSEARCH文は図1-2参照)

<< 削除 >>

パターン名 : PTEST1
コメント : 固有名詞+と
リターン形式 : [レベル] 文

単語 : \$<単語1>
BEFORE :
AFTER :
文字数 :
PART-OF : \$<文節1>
PREV-IS :
NEXT-IS : \$<単語2>
出現単語 :
読み :
標準表現 :
音便 :
品詞 : 固有名詞
活用型 :
活用形 :
係受け :

単語2のPREV-ISに単語1が定義されています。削除しますか Y/N ?

図 4 - 7 オブジェクト変数の削除 (確認)

ここで "Y" を入力すると、単語2のPREV-ISの条件から単語1が削除される。
"N" を入力すると、単語2のPREV-ISの条件には単語1が残る。

文節1のCOMPOSED-OFについても確認を行う。

4. 2 オブジェクトウインドウ

オブジェクトウインドウは、主として、個々のオブジェクト条件のエディットを行うためのウインドウである。(オブジェクト条件削除の確認でも使用する。4. 1 (9) 参照)

メインウインドウのオブジェクト変数行でエディットコマンドを実行すると、そのオブジェクト条件が展開され、オブジェクトウインドウに表示される。(図4-8)

.<< オブジェクトウインドウ >>	
パターン名	: PTEST1
コメント	: 固有名詞+と
リターン形式	: [レベル] 文

単語	: \$<単語1>
BEFORE	:
AFTER	:
文字数	:
PART-OF	: \$<文節1>
PREV-IS	:
NEXT-IS	: \$<単語2>
出現単語	:
読み	:
標準表現	:
音便	:
品詞	: 固有名詞
活用型	:
活用形	:
係受け	:

図4-8 オブジェクトウインドウ

次に、オブジェクトウインドウで使用可能なコマンドについて説明する。但し、便宜上、スロット条件のエディットコマンドは5. で説明する。

(1) ~ (4) は、カレント行がどこにあっても使用できる。

(5) はパターン名行、(6) はコメント行、(7) はリターン形式行のエディットコマンド、(8) はオブジェクト条件の、(9) はスロット条件のデリートコマンドである。

- (1) カレント行の移動
- (2) 画面の再表示
- (3) ヘルプ

これらは、メインウインドウと同じである。4. 1 参照

- (4) クイット

オブジェクト条件のエディットを終了し、メインウインドウへ戻る。

- (5) 検索パターン名のエディット
- (6) コメントのエディット

オブジェクトウインドウでもこれらを実行できる。
操作はメインウインドウと同じである。4. 1 参照

- (7) リターン形式のエディット

便宜上、6. で説明する。

- (8) オブジェクト条件の削除

デリートコマンドにより、そのオブジェクト条件が削除される。但し、実際に削除する前に、ユーザに確認を求める。機能は4. 1 参照。

(9) スロット条件の削除

便宜上、まず、削除の説明を行う。

スロット条件をすべて削除するには、アリートコマンドを実行すればよい。

アリートコマンドを実行すると、スロットの条件部だけでなく、もし逆スロット条件が設定されていたとき、それをどうするのかユーザに確認する。

例えば図5-1はNEXT-ISスロットを削除した場合の例である。

```
<< オブジェクトウインドウ >>

パターン名      : PTEST1
コメント        : 固有名詞+と
リターン形式    : [レベル] 文
-----
単語            : $<単語1>
BEFORE          :
AFTER           :
文字数          :
PART-OF         : $<文節1>
PREV-IS         :
NEXT-IS         : $<単語2>
出現単語        :
読み            :
標準表現        :
音便            :
品詞            : 固有名詞
活用型          :
活用形          :
係受け          :
単語2のPREV-ISに単語1 が定義されています。削除しますか Y/N.??
```

図5-1 スロット条件の削除(確認)

ここで"Y"を入力すると、単語2のPREV-ISの条件から単語1が削除される。

"N"を入力すると、単語2のPREV-ISの条件には単語1が残る。

5. スロット条件のエディット

オブジェクトウインドウで各スロットのスロット条件のエディット、削除が行える。

スロットのタイプにより、エディット方法が異なるので、

- 一般の即値条件を持つスロット (単語の出現単語など)
- スロットのデータの値が制限されているもの (単語の品詞など)
- シングルポインター型 (PART-OFなど)
- 疑似マルチポインター型 (COMPOSED-OFなど)

に分けて説明する。

5. 1 即値データを持つスロット

エディットコマンドによりスロット条件のエディットができる。
エディット方法はパターン名などと同じように行エディットする。

ここで"\$"で始まる文字列が入力されると、それは変数として認識される。

変数は任意の文字列であるが、他で使用するオブジェクト変数("単語1"など)は使用できない。

(例)

「単語の"読み"と"出現単語"が同じもの」というオブジェクト条件を定義するとき、読み、出現単語において、それぞれ

```
読み      : $X1  
出現単語 : $X1
```

と、入力すればよい。

ここで入力した変数も、他の変数と同じように"\$<X X>"の形で表示される。(図5-2)

OR 条件はスペースコードで区切って入力する。(図5-2、標準表現)

<< オブジェクトウインドウ >>	
パターン名	: XXXX
コメント	: 変数記述
リターン形式	: [レベル] 文

単語	: \$<単語1>
BEFORE	:
AFTER	:
文字数	:
PART-OF	:
PREV-IS	:
NEXT-IS	:
出現単語	: \$<X>
読み	: \$<X>
標準表現	: する 為る
音便	:
品詞	:
活用型	:
活用形	:
係受け	:

図 5 - 2 変数の記述

5. 2 メンバーデータを持つスロット

データの値が制限されているもの（品詞、音便等）も、エディットコマンドによりスロット条件のエディットができるが、直接入力するのではなく、メニューを使用する。

エディットコマンドを実行すると、図5-3に示すように、画面左上にそのスロットの条件となり得るメンバーの一覧メニューが表示される。

このメニューは複数選択可能メニューで、各項目はEXITを除きトグルになっている。

項目間は矢印キーで移動でき、リターンキーで選択/非選択が切り替わる。（選択中の項目は反転表示される。）

選択終了後、EXITを選択する（またはqをキーインする）と、その時選択されていた項目のOR条件となる。当然であるが、1つだけなら通常の即値条件となる。

音便表	ーンエディタ（スロットエディット） >>
EXIT	: XXXX
NIL	: 変数記述
撥音便	: [レベル] 文
促音便	-----
イ音便	: \$<単語1>
ウ音便	:
	:
	:
PART-OF	:
PREV-IS	:
NEXT-IS	:
出現単語	:
読み	:
標準表現	:
音便	:
品詞	:
活用型	:
活用形	:
係受け	:

図5-3 メンバーからの選択（音便の例）

メンバーデータを持つスロット条件として変数を入力したいこともある。その時には変数入力コマンド（v）を使用する。

変数入力の方法は5.1の即値データの場合と同じである。

なお変数入力コマンドは、他のスロット（即値データを持つスロットなど）で使用した場合、エディットコマンドと同じとなる。

5. 3 シングルポインター型のデータを持つスロット

PART-OF、BEFORE、PREV-IS、NEXT-IS、AFTER等、シングルポインター型スロット値を持つスロット条件のエディットもメニューを使用する。

条件として既に何か値が入っている場合、次のような警告メッセージが表示され、何も実行せずにコマンド入力モードに戻る。

~~シングル条件しか記述できません。~~

また、何も表示されていなくても、逆スロット条件により暗黙のうちに設定されていることがある。例えば次のようなDEFSEARCH文である。

```
(DEFSEARCH PTEST2 "固有名詞+と"  
  (OBJECT-PATTERN (VAR 単語1)  
    ((INSTANCE-OF 単語)  
      (品詞 固有名詞)))  
  (OBJECT-PATTERN (VAR 単語2)  
    ((INSTANCE-OF 単語)  
      (PREV-IS (VAR 単語1))  
      (出現単語 と)))  
  (RETURN-FORM 文))
```

このDEFSEARCH文では、単語1のオブジェクト条件にはNEXT-ISが明示されていないが、単語2のPREV-ISに単語1が記述されている。つまり暗黙のうちに単語1のNEXT-ISは単語2となっている。

このような場合、図5-4のようなメッセージが出た後、自動的にその暗黙の条件が設定され、コマンド入力モードに戻る。

<< オブジェクトウインドウ >>

パターン名 : PTEST2
コメント : 固有名詞+と
リターン形式 : [レベル] 文

単語 : \$<単語1>
BEFORE :
AFTER :
文字数 :
PART-OF :
PREV-IS :
NEXT-IS :
出現単語 :
読み :
標準表現 :
音便 :
品詞 : 固有名詞
活用型 :
活用形 :
係受け :

単語2のPREV-ISに単語1が定義されています。

図5-4 暗黙のスロット条件のセット

以上のような事を確認した後、スロット条件の設定が可能ならば、画面左上に次のメニューが現れる。

エディット方法
EXIT
展開
パターン指定

EXITはオペレーションのキャンセルとなる。(以下、特に断わらない限り EXIT は同様である)

(1) “展開”

ポインター型のデータを持つスロットは、必ず自分以外のオブジェクトを指す。

展開というのは、そのオブジェクトのためのオブジェクト条件を新たに作成することになる。

表示されていたオブジェクト条件が画面から消え、展開された新しいオブジェクト条件が表示され、エディット可能となる。この時、この新しいオブジェクト条件には、逆スロット条件が設定されている。

先ほどのDEFSEARCH文 PTEST2 において、単語1のPART-OFでこれを実行したとすると図5-5に示すオブジェクト条件が表示される。

<< オブジェクトウインドウ >>	
パターン名	: PTEST2
コメント	: 固有名詞+と
リターン形式	: [レベル] 文

文節	: \$<文節1>
BEFORE	:
AFTER	:
単語数	:
PART-OF	:
COMPOSED-OF	: \$<単語1>
PREV-IS	:
NEXT-IS	:
文節対応	:

図5-5 展開例

(2) "パターン指定"

このメニュー項目名の"パターン"は「オブジェクトパターン」を指す。これは、そのスロット条件を既に定義されているオブジェクト変数とした場合に使用する。

これを選択すると、そのスロットの条件として選択可能なオブジェクト変数の一覧がメニューが表示されるので、設定したい変数を選択する。但し、候補の一覧は必ずしも完全なものではない(余分なものを含むことがある)。あくまで文法的に許される"候補"であり、その中からの選択はユーザの責任である。

(1)の"展開"とは違って、スロット条件が変更されるだけで、表示されるオブジェクト条件はそのままである。

例えば作成していく過程で、概念として次のようなDEFSEARCH文となることがある。


```

(DEFSEARCH PTEST3 "固有名詞+と"
(OBJECT-PATTERN (VAR 単語1)
((INSTANCE-OF 単語)
(品詞 固有名詞)))
(OBJECT-PATTERN (VAR 単語2)
((INSTANCE-OF 単語)
(PART-OF (VAR 文節1))
(出現単語 と)))
(OBJECT-PATTERN (VAR 文節1)
((INSTANCE-OF 文節)
(COMPOSED-OF (VAR 単語2))))
(RETURN-FORM 文))

```

ここで、単語1のPART-OFに文節1をセットしたい時、この機能を用いる。
この例の場合、PART-OFの候補となり得るのは文節1だけであるから、次のようなメニューが表示される。

パターン選択

```

EXIT
文節1

```

ここで、文節1を選択すると単語1のPART-OFに文節1がセットされる。

それと同時に、逆スロット条件の設定も行う。

NEXT-IS (PREV-IS) のように逆スロットもシングルポインター型であればシステムは内部的に自動設定するだけであるが、この例のように逆スロットがマルチポインター型で、且つ、既になんらかの条件が設定されている場合、挿入する位置を決めなければならない。

この例であれば、文節1のCOMPOSED-OFのどこに単語1を挿入するか決めなければならない。

そのために、次のような"位置決めメニュー"が表示される。

挿入位置

```

EXIT
*
単語2
*

```

ここで、挿入したい位置の"*"を選択すればよい。(画面の都合上、上下に表示されるが、メニューの上がCOMPOSED-OFの並びの前である。)

5. 4 疑似マルチポインター型のデータを持つスロット

COMPOSED-OF、各種対応、係受け等、疑似マルチポインター型スロット値を持つスロット条件のエディットもメニューを使用する。

但し、疑似マルチポインター型スロットはCOMPOSED-OFを除いて、その中のデータの並び（スロット値）は意味がないので、マッチング条件に複数個の変数を並べることは無意味である。（文法的には許される）

まず、逆スロット条件により暗黙のうちに設定されていることがあるので、システムはそのようなものがないか調べる。そして、もし逆スロット条件で何か設定されていた場合（複数個存在することがある得る）、その条件を正しく並べようと試みる。

例えば、次のようなDEFSEARCH文を考える。

```
(DEFSEARCH PTEST4 "固有名詞+と"  
  (OBJECT-PATTERN (VAR 単語1)  
    ((INSTANCE-OF 単語)  
     (PART-OF (VAR 文節1))  
     (品詞 固有名詞)))  
  (OBJECT-PATTERN (VAR 単語2)  
    ((INSTANCE-OF 単語)  
     (PART-OF (VAR 文節1))  
     (出現単語 と)))  
  (OBJECT-PATTERN (VAR 文節1)  
    ((INSTANCE-OF 文節)))  
  (RETURN-FORM 文))
```

このDEFSEARCH文で文節1のオブジェクト条件を開くと、スロット条件には何も表示されない。そしてCOMPOSED-OFスロットのエディットをしようとする、まず次のようなメッセージが表示される。

逆条件があります。逆条件からスロット条件を設定しますか Y/N?

ここで"Y"と入力すると、逆スロット条件より図5-6に示すように暗黙の条件が表示される。

暗黙の条件が複数個存在する場合、必ずしも図5-6のように意味として正しい順になるとは限らない。（むしろ正しくないことが多い）

その順序をユーザが設定したいときは、先ほどのメッセージに対し"N"と入力し、"パターン指定"により並べるか、または、とりあえず設定して、"削除"と"パターン指定"を用いて並べ直すとよい。（"パターン指定"、"削除"は次に説明する）

<< オブジェクトウインドウ >>

パターン名 : PTEST4
コメント : 固有名詞+と
リターン形式 : [レベル] 文

文節 : \$<文節1>
BEFORE :
AFTER :
単語数 :
PART-OF :
COMPOSED-OF : \$<単語1> \$<単語2>
PREV-IS :
NEXT-IS :
文節対応 :

図 5 - 6 暗黙の条件の展開

暗黙の条件のチェック、展開を行った後、次のようなメニューが表示される。

エディット方法
EXIT
展開
パターン指定
*挿入
?挿入
疑似マルチ
削除
全部削除

(1) "展開"

これは5.3のシングルポインター型の(1)"展開"とほぼ同様である。但し、展開する前に"位置決めメニュー"を使用して、自分自身の条件列での挿入位置を決定する。展開後、新しいオブジェクト条件が表示される。

(2) "パターン指定"

これは5.3のシングルポインター型の(2)"パターン指定"とほぼ同様である。

但し、展開する前に"位置決めメニュー"を使用して、自分自身の条件列での挿入位置を決定する。

また、選択されたオブジェクト条件の逆スロットが疑似マルチポインター型の場合(対応テーブルのいくつかはこのような関係にある)、選択されたオブジェクト条件での逆スロット条件の挿入位置も同じように"位置決めメニュー"を使用して決定する。

(3) "*挿入"、"?挿入"

これは、疑似マルチスロットのマッチングにおけるメタキャラクター"*"、"?".を挿入する時に使用する。

"位置決めメニュー"を使用して挿入位置を決定する。

(4) "疑似マルチ"

リターン形式が変数指定、統計の場合、そのスロット値に並ぶ疑似マルチを一括して処理できるように、並びを1変数(疑似マルチ変数)にする。

この変数はリターン形式以外の所(他のスロット条件)では使用できない。

また、他の一般変数と区別するため、疑似マルチ変数は画面上、図5-8のように"M"を前につけて表示している。

<< オブジェクトウインドウ >>

パターン名 : PTEST4
コメント : 固有名詞+と
リターン形式 : [レベル] 文

文節 : \$<文節1>
BEFORE :
AFTER :
単語数 :
PART-OF :
COMPOSED-OF : M\$<単語3>
PREV-IS :
NEXT-IS :
文節対応 :

図 5 - 7 疑似マルチ変数

(5) "削除"、"全部削除"

"削除"を選択すると、メニューが現れ、削除可能な変数(*、?のメタキ
ャラクタ等も含む)を列挙する。ここで選択した変数がスロット条件から削
除される。

"全部削除"はスロット条件に並ぶ変数(メタキョラクタを含む)をすべ
て削除する。

なおこれは、コマンド入力モードにおいてデリートコマンドを実行するの
と全く同じである。

6. リターン形式のエディット

メインウインドウ、オブジェクトウインドウにおいて、カレント行がリターン形式の位置にある時、エディットコマンドにより、リターン形式のエディットができる。

リターン形式変更には、リターン形式そのものを変更する場合と、パラメータのみを変更する場合があるので、まず次のようなメニューが現れる。

選択
EXIT
リターン形式変更
パラメータ編集

「パラメータ編集」を選択した場合、リターン形式のリターンパラメータのエディットが可能である。パラメータのエディット方法は次に説明する「リターン形式変更」の中で行うのと同じ手順であるので、ここでは「リターン形式変更」のみ説明する。

「リターン形式変更」を選択すると、リターン形式を変更できる。これを選択すると、続いて次のメニューが現れる。

リターン形式変更
EXIT
変数指定
レベル指定
統計

6. 1 変数指定

次の例のような、検索パターン中に定義されている変数の一覧のメニューを使用しながら、リターン変数を設定していく。

“会話”は会話番号(会話ID)を返したい時に使用する為のものであり、必ず変数一覧に含まれている。

変数選択
EXIT
会話
単語1
単語2
文節1

変数指定のパラメータには複数個の変数を指定できるので、メニューで変数を選択すると、選択した変数がパラメータリストに並んでいく。

但し、文字列に変換可能な変数の場合、次のようなメッセージがでる。

文字列に変換しますか Y/N?

ここで“Y”と入力すれば、その変数については文字列に変換される。

“N”であればデータIDをそのまま返す。

IDを返すものは、図6-1に示すように、パラメータリストの変数の前に“#”をつけて表示されている。

<< メインウインドウ >>	
パターン名	: PTEST1
コメント	: 固有名詞+と
リターン形式	: [変数指定] #<単語3>

単語	: \$<単語1>
単語	: \$<単語2>
文節	: \$<文節1>

図6-1 IDのリターン

6. 2 レベル指定

次のようなメニューが表示されるので、リターンレベルを選択し、変更する。

DEFSEARCH文で許されている範囲指定パラメータは、パターンエディタではサポートしない。どうしても必要な場合、検索パターンをセーブした後、通常のエディタでDEFSEARCH文を修正する。

レベル指定
EXIT
発話
文
文節
単語

6. 3 統計

“変数指定”とはほぼ同様である。

但し、IDのリターンは統計では許されていないので、文字列に変換できない変数は選択メニューに現れない。

第4部

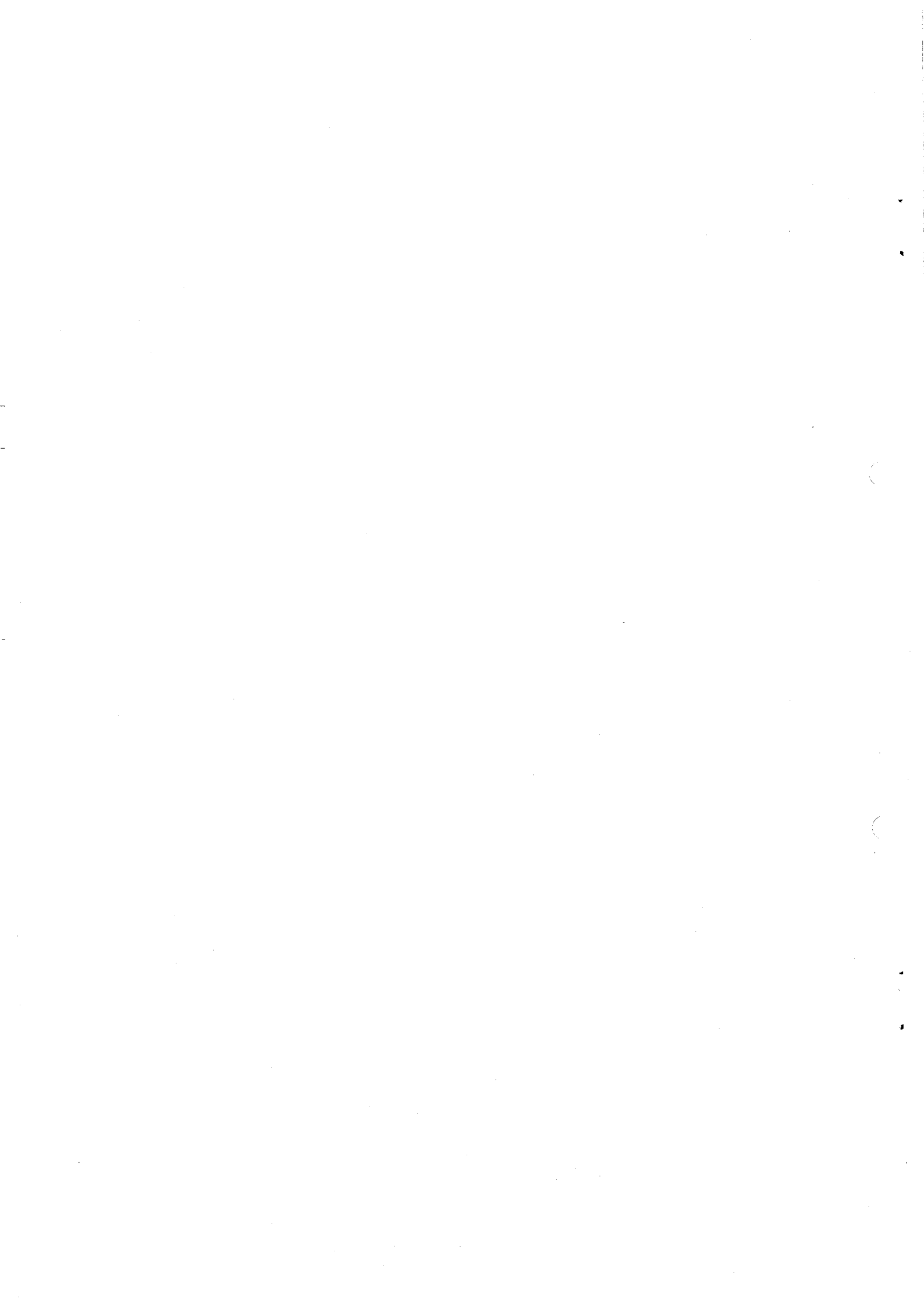
DEFSEARCH文法説明書

第1.0版



目次

	頁
1. 概要	1
2. 文法説明	4
2. 1 検索パターンのBNF的記述法による文法定義	4
2. 2 検索パターン	5
2. 3 オブジェクト条件	6
2. 4 スロット条件	7
2. 5 スロット値条件	8
2. 6 リターン形式	9
3. 検索パターンのコーディングと結果の例	12
例1 格助詞「は」の含まれる文	12
例2 ”わかりました”が含まれる文	13
例3 ”に+++ (標準表現)する”が含まれる文	14
例4 ”わかりました”が含まれる文節 (変数指定)	15
例5 標準表現の度数 (統計)	16



1. 概要

本書では、問合せ言語DEFSEARCHについて説明する。

なお、この文法説明書を読むに先だって、データベースの構造について理解していることが望ましい。ATRテクニカルレポートTR-I-0150「自動翻訳電話研究用言語データベースADDの構成」を参照のこと。

DEFSEARCHは、ADD検索の多様性に対応するために開発された言語である。利用者はDEFSEARCHを用いて、自分が得ようとする言語パターン（以下、これを検索パターンと呼ぶ）を記述することにより検索を行う。

検索パターンの作成にはパターンエディタ（「パターンエディタ操作説明書」参照）を用いるが、emacsなどのエディタを用いてもかまわない。

検索パターンは、オブジェクト条件列とリターン形式、モードで記述する。

```
(DEFSEARCH パターン名 "コメント"  
          << OBJECT条件+ >>  
          << リターン形式 >>  
          << モード文 >>)
```

この最初の予約語"DEFSEARCH"より、検索パターンをDEFSEARCH文と呼ぶ。

また、DEFSEARCH文を記述したファイルをパターンファイル、あるいは、DEFSEARCHファイルと呼ぶ。

「オブジェクト条件」と言うのは、そのオブジェクトの各スロットが満足すべき条件を記述するものである。

例えば、「単語オブジェクトで、品詞が格助詞」というのは、

```
(OBJECT-PATTERN (VAR 単語1)  
                 ((INSTANCE-OF 単語)  
                  (品詞 格助詞)))
```

と記述する。

オブジェクト条件を代表する変数（上の例では単語1）を「オブジェクト変数」（正確に言うならばオブジェクト条件変数）と呼ぶ。

スロット名（上の例では品詞）の右に記述するのが（上の例では格助詞）「スロット値条件」である。この1行を「スロット条件」と呼ぶ。但し、混乱しないと思われる場合、スロット値条件のことをスロット条件と呼ぶこともある。

検索が成功したとき「結果をどの様に返すか」というのを記述する部分が「リターン形式」である。

リターン形式として、

(1) レベル指定

結果をあるレベルに統一して、テキストの形に変換して返す。

(2) 変数指定

どのオブジェクトを返すか、直接指定する。

(3) 統計

オブジェクトを文字列に変換し、文字列毎の統計（カウント）を返す。

の3種類がある。

「モード文」というには、検索シェル（ldbsh）でシステムが内部的に使用するものであり、ユーザが定義しても、それは無視される。（現時点では、パターンファイルのモードを直接使用する検索はできない）

モード文は、リターン形式が「レベル指定」の場合、対応する英語、あるいは、ユニーク処理を行うためのものである。（詳細は「ADD検索システム操作説明書」を参照のこと）

このページ空白

2. 文法説明

2. 1 検索パターンのBNF的記述法による文法定義

検索パターンのBNF的記述法、以下のようになる。

```
(DEFSEARCH パターン名 "コメント"  
  << OBJECT条件+ >>  
  << リターン形式 >>)  
  << モード文 >>)
```

```
<< OBJECT条件 >> ::= (OBJECT-PATTERN <<オブジェクト変数>>  
  ((INSTANCE-OF クラス名)  
  << スロット条件+ >>  
  ))
```

```
<< オブジェクト変数 >> ::= 変数  
<< スロット条件 >> ::= (スロット名 << スロット値条件 >>+ )  
<< スロット値条件 >> ::= リテラル|?|*|(NOT リテラル)|(OR (リテラル+ ))|  
  <<変数>>|<<疑似マルチ変数>>  
変数 ::= (VAR リテラル)  
疑似マルチ変数 ::= ($VAR リテラル)
```

```
リターン形式 ::= << レベル指定 >>|<< 変数指定 >>|<< 統計 >>  
<< レベル指定 >> ::= (RETURN-FORM <<レベル>> [-範囲 [+]範囲] )  
<< 変数指定 >> ::= (RETURN <<リターン変数 >>+ )  
<< 統計 >> ::= (RETURN-STAT << 変数 >>+ )
```

```
レベル ::= 以下の何れかのキーワード  
  a) 単語、b) 文節 c) 文 d) 発話  
- 範囲 ::= 同一レベル内での開始位置を指定する  
[+]範囲 ::= 同一レベル内での終了位置を指定する
```

```
リターン変数 ::= <<変数>>|<<疑似マルチ変数>>|<<ID変数>>  
ID変数 ::= (#VAR リテラル)|(#$VAR リテラル)
```

```
モード文 ::= (MODE << モード指定 >>+ )  
モード指定 ::= <<言語モード>> | <<統計モード>> |  
  <<ユニークモード>> | <<ソートモード>>  
言語モード ::= (LANGUAGE 英語|日本語)  
統計モード ::= (STAT T|NIL)  
ユニークモード ::= (UNIQUE T|NIL)  
ソートモード ::= (SORT T|NIL)
```

(注) モード文はリターン形式がレベル指定の場合のみ有効。
(STAT T)または(UNIQUE T)を指定すると(SORT T)のモードとなる

(注)

- ① 同じ変数は同じ値 (オブジェクト)
- ② << スロット値条件 >> の? は任意の1つの値、* は0個以上の任意の値を指す。

2. 2 検索パターン

検索パターンは
(DEFSEARCH
で始まり、続いて、
パターン名
任意のコメント
が続き、その後、
1つ以上のオブジェクト条件列
があり、最後に
リターン形式
があつて、
)
で終わる。

2. 3 オブジェクト条件

オブジェクト条件は
 (OBJECT-PATTERN
で始まり、続いて、
 オブジェクト変数
が続き、その後、
 (INSTANCE-OF クラス名)
 スロット条件列
があつて、
)
で終わる。

クラス名はデータベースのテーブル名に対応するもので、そのオブジェクト条件がどのデータクラスに対するものかを意味する。

オブジェクト変数というのはそのオブジェクト条件を代表する変数である。

2. 5 スロット値条件

スロット値条件は、即値データをもつスロット（出現単語など）、シングルポインター値を持つスロット（NEXT-ISなど）、疑似マルチポインターをもつスロット（COMPOSED-OFなど）により記述できるものが異なる。（2.1でBNF的としたのはこのあたりが不正確であるからである。）

なお、データベースでは「意味関係名」は疑似マルチスロットであるが、DEFSEARCH文では即値データでマッチングができる。（どちらか一方が成功すればマッチングしたとする）

即値データを持つスロットの場合、スロット値条件は1つしか記述できない。

それは、即値型の条件、OR条件、NOT条件、変数（オブジェクト変数は記述できない）の何れかである。

（例）

即値型：

（出現単語 が）

（品詞 名詞）

OR、NOT：

（標準表現（OR する 為る））

（品詞（NOT 助動詞））

変数：

（品詞（VAR X））

シングルポインター型の場合、スロット値条件として記述できるのはオブジェクト変数のみで、しかも1つしか記述できない。

（例）

(NEXT-IS (VAR 単語1))

疑似マルチポインター型のスロットの場合は、

複数個（1個のみでもよい）のオブジェクト変数、
メタキャラクター（*,?）

または

疑似マルチ変数

である。

疑似マルチ変数とオブジェクト変数などの混在はできない。

疑似マルチ変数は、そのスロットの疑似マルチデータを一括処理するためのもので、リターン形式のパラメータとして使用する。詳細はリターン形式で説明する。

（例）

(COMPOSED-OF * (VAR 単語1) * (VAR 単語1) *)

(COMPOSED-OF (MVAR 単語1))

2. 6 リターン形式

リターン形式として、次の3種類がある。

- (1) レベル指定 (RETURN-FORM)
- (2) 変数指定 (RETURN)
- (3) 統計 (RETURN-STAT)

(1) レベル指定

これは、その検索パターン中に記述されている全てのオブジェクト変数にマッチングしたデータを包含するように、指定したレベルに統一して結果を返す。具体例を示した方が理解しやすいと思われるので、具体例で示す。次のようなDEFSEARCH文を考える。

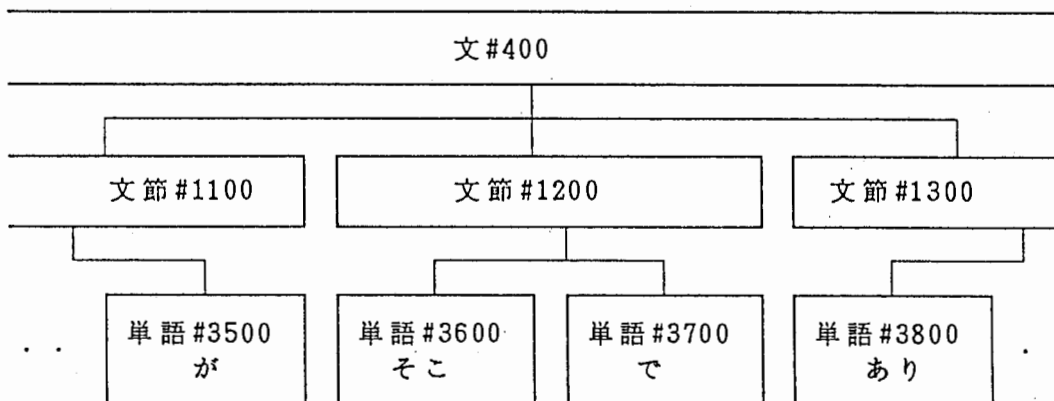
```
(DEFSEARCH SAMPLE1 "サンプル"  
  (OBJECT-PATTERN (VAR 単語1)  
    ((INSTANCE-OF 単語)  
     (AFTER (VAR 単語2))  
     (出現単語 が)))  
  (OBJECT-PATTERN (VAR 単語2)  
    ((INSTANCE-OF 単語)  
     (標準表現 有る)))  
  (RETURN-FORM 文))
```

このDEFSEARCH文で検索を行った結果の1組として、単語1、単語2にそれぞれ次のようなデータがマッチングしたとする。

単語1 (#3500) 単語2 (#3800)

"#3500" はデータIDとする。

また、これらの単語が、次のような文節、文の要素であったとする。



リターンレベルが文の場合、単語#3500、単語#3800を包含する文#400をテキストにして結果を返す。

結果： 3月13日に国際会議がそこありますか？

リターンレベルが文節の場合、単語#3500、単語#3800を包含する文節#1100～文節#1300をテキストにして結果を返す。

結果： 国際会議がそこありますか？

なお、パターン中で使用する変数のレベルより、リターンで指定するレベルの方が低い場合、結果は保証されない。

(2) 変数指定

これは、返したいオブジェクト変数、または一般変数を直接指定して返す方法である。

パラメータには、検索パターン中で使用する任意の変数を記述できる。
(複数個の変数を並べて指定できる。)

テキストに変換できるもの(会話、発話、文、文節、単語の変数など)はテキストに変換して返す。

また疑似マルチ変数を指定した場合、その変数にマッチングした疑似マルチをかえす。(テキストに変換できるものはテキストに変換)

また、テキストに変換せず、IDをそのまま返すこともできる。

これは変数" (VAR"、" (MVAR" の代わりに、" #" を前につけた変数、" (#VAR"、" (#MVAR" を使用する。

会話IDを返したいことがしばしばあるので、

(#VAR 会話)

は予約語として用意されている。

(3) 統計

これは、データベース中の出現頻度を調べたいときに使用する。

変数指定と同じように、頻度を取りたい変数を並べる。

複数の変数を並べた場合、その組合せ毎に頻度をとる。

3. 検索パターンのコーディングと結果の例

(例1) "格助詞「は」の含まれる文

<検索パターン>

```
(DEFSEARCH HA "格助詞[は]"
  (OBJECT-PATTERN (VAR 単語1)
    ((INSTANCE-OF 単語)
      (出現単語 は)
      (品詞 格助詞)))
  (RETURN-FORM 文 0 0))
```

<結果>

1. 太郎は本を読んだ。
2. 次郎は絵を描いた。

(例3) "に+++ (標準表現) する" が含まれる文

<検索パターン>

```
(DEFSEARCH に+する "に+++ (標準表現) する"  
  (OBJECT-PATTERN (VAR 単語1)  
    ((INSTANCE-OF 単語)  
      (AFTER (VAR 単語2))  
        (出現単語 に)  
        (標準表現 に)  
        (品詞 格助詞)))  
  (OBJECT-PATTERN (VAR 単語2)  
    ((INSTANCE-OF 単語)  
      (標準表現 する)))  
  (RETURN-FORM 文 0 0))
```

<結果>

1. . . . 会議に参加する。
2. . . . そちらに発送します。

(例4) "わかりました"が含まれる文節(変数指定)

<検索パターン>

```
(DEFSEARCH WAKARIMASHITA "わかりました"  
  (OBJECT-PATTERN (VAR 単語1)  
    ((INSTANCE-OF 単語)  
     (PART-OF (VAR 文節1))  
     (NEXT-IS (VAR 単語2))  
     (出現単語 わかり)))  
  (OBJECT-PATTERN (VAR 単語2)  
    ((INSTANCE-OF 単語)  
     (PART-OF (VAR 文節1))  
     (NEXT-IS (VAR 単語3))  
     (出現単語 まし)))  
  (OBJECT-PATTERN (VAR 単語3)  
    ((INSTANCE-OF 単語)  
     (PART-OF (VAR 文節1))  
     (出現単語 た)))  
  (OBJECT-PATTERN (VAR 文節1)  
    ((INSTANCE-OF 文節)))  
  (RETURN (#VAR 会話) (#VAR 単語1) (VAR 単語1) (VAR 文節1)))
```

<結果>

1. 会話:#3045/単語1:#3400/単語1:わかり/文節1:わかりました。
2. 会話:#3045/単語1:#5400/単語1:わかり/文節1:わかりました。

(例 5) 標準表現の度数 (統計)

<検索パターン>

```
(DEFSEARCH 度数-1 "標準表現の度数"  
  (OBJECT-PATTERN (VAR 単語1)  
    ((INSTANCE-OF 単語)  
      (標準表現 (VAR X))  
    ))  
  (RETURN-STAT (VAR X)))
```

<結果>

1. 12 の
2. 8 ます