

## O. 用語知識ベースのための編集機能の試作：用語属性・階層構造の編集

後藤 智範

神奈川大学 理学部 情報科学科

### 1. はじめに

現在、多くの辞書がCD-ROM化され、そのGUIソフトによって広く利用されているのに対し、検索ツールとしてのシソーラスは、未だに冊子体での利用が主であり、一般利用者には有効に活用されていない。本研究室では、シソーラスおよび階層構造をもつ用語データを多面的なアクセスを可能とし、特定のシソーラスに限定されない汎用性の高いBrowserとその応用ソフトウェアの開発を行ってきた[1][2][3]。

また、通常、シソーラス等のように用語間に狭義語や広義語の関係がある用語木を編集する際に編集者は、辞書データの入ったファイルを直接編集する方法が一般的である。しかしこの方法は、データファイルにおいて編集対象となる用語についての用語情報や用語関係等の詳細情報をあらかじめ調べておく必要がある。しかし、入力ミス等で編集が失敗した際、編集した用語間の関係が崩れてしまう恐れがある。

当研究は、編集者が辞書、専門用語辞典シソーラスなどの用語知識ベースを編集する際、視覚的な木構造表示を可能とする上述の browser 上で、用語属性および用語の上位-下位関係を直接編集する機能の実装を意図するものである。を試作した。この機能により、利用者は用語間の関係を視覚的に把握した上で用語木を編集することができ、編集後の用語関係をリアルタイムに確認できることが実現できる。本報では、この木構造上表示上の編集機能の詳細・問題点について報告する。

### 2. 階層構造を有する用語データのブラウジング機能

用語間に階層構造を有するデータを多角的に、言い換えればその構造からアクセスするためのブラウザーは、下記に挙げる機能を有することが必要とされる。

- (1) 見出し語の迅速かつ簡便な検索・表示。
- (2) 特定された見出し語の上位語-下位語のグラフィカル表示。
- (3) 特定された見出し語の上位-下位以外の関係の表示。
- (4) 履歴表示
- (5) 複数シソーラスの同時使用
- (6) マウス操作主体

また、後で説明する編集機能についてもこのブラウジング機能を元に GUI 上で編集可能と

なる。まずは元となるブラウジング機能について説明する。

## 2.1 Browser の利用事例

上記の要件・機能の満たすべく Java(JDK1.4.0)で実装を行った。動作の検証のためのシソーラスとして下記の3種類を用いた。

- (1)1987年度版INSPECシソーラス 見出し語数: 約9,000語(ディスクリプタ:約5500語)
- (2)1987年度版JICSTシソーラス 見出し語数: 約50,000語(ディスクリプタ:約40000語)
- (3)1995年度版EDR概念辞書 見出し語数: 約400,000語

## 2.2 Browser の構成

本ブラウザーは、4つの window コンポーネントから構成される。

- (1) メイン window
- (2) 用語リスト window
- (3) 用語木 window
- (4) 関連 window

(1)は最初に表示される window で、シソーラスの選択および、履歴ファイルの指定による過去の使用用語の履歴表示のために用いられる。図 2.1 にその例を示す。頻繁に使用する可能性の高い用語を(1)に保存をしておくことにより(2)を用いて用語を探索する手間を省くことができる。

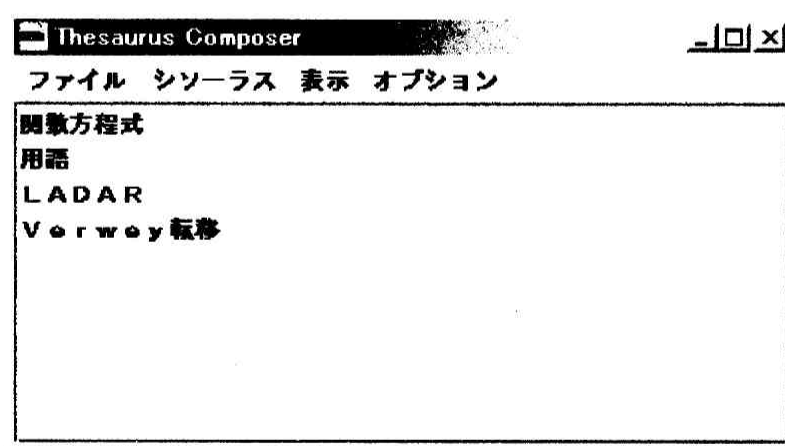


図 2.1 メイン window

(2)は見出し語を探索、表示するための window である。EDR 概念辞書を用いた実例を図 1.2 に示す。利用者が(1)で EDR 概念辞書を選択すると、用語リスト window は空白状態で表示される。次に、利用者が探したい用語の先頭文字を含む行(「0」～「わ」)のボタンをマウスでクリックすると、当該行の文字(濁音などを含む)のコンボボックスを表示する(この例で

は、「ラ」行)。利用者が文字のボタンを選択すると、現在探索中の文字列を表示するテキストフィールドにその文字が追加され、その文字列から始まる見出し語が表示される。

現在探索中の文字列を表示するテキストフィールドはキーボードからの直接入力も可能であり、探索したい語がすでに判明している場合などは一文字ずつマウスで入力する手間を省くことができる。

対象用語の総数が数万語以上になると、当該用語を見つけるためのスクロールが必要とされ、これが利用者に多大な手間と時間を課することが最初のbrowserの試作時に判明した。用語探索に用いる文字列を徐々に長くしていくことにより、見つけたい用語の範囲が徐々に狭められるためにスクロールの負担が軽減されようになった。

また、用語リストwindowに表示される用語を指定し、上の「表示」ボタンを押すか、または右クリックのポップアップメニューにより、「NT」、「GR」ボタンが現れる(図2.2では「投げる」を指定)。「NT」ボタンが押されると、用語木windowが表示され、「GR」ボタンを押すと関連windowが表示される。

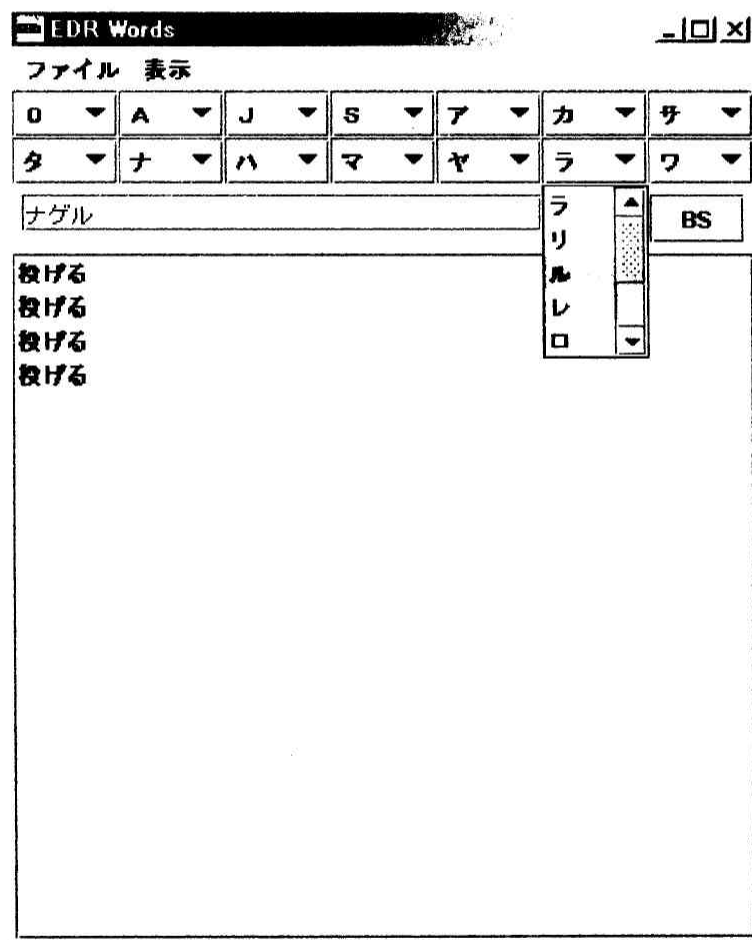


図 2.2 用語リスト window

(3)の用語木windowは、(a)階層表示、(b)二次元カーソル(左上のボックス)、(c)構成用語リスト、(d)用語木情報、の4つのコンポーネントから構成されている。図2.3にこの例を示す。

シンソーラスを構成する木の大きさ(用語数)、木の高さは様々であり、通常のディスプレイの解像度(1024×768～1600×1200)の範囲を超えるような木も数多く存在する。用語木を構成する個々の用語を閲覧するためには、縦横のスクロールバーによる操作が必須となるが、木が大きい(100語以上)となると、この操作は利用者に多大な負荷を課す。(b)および(c)はこの操作を軽減するために導入したコンポーネントである。

(a)の階層表示は木構造を表示させた用語が木全体の何番目の改装に存在するか、用語木の最大の高さ、用語木に含まれる総用語数がデータとして表示される。

(b)二次元カーソルは、Xtoolkitにも含まれるコンポーネントであり、これにより縦横のスクロールバーは不必要となった。通常のスクリールバーに比べ木全体の中における現在の表示領域の大きさと、表示位置を視覚的に理解しやすくなり、長い距離のスクロールも瞬時に行うことができる。しかし、用語木が非常に巨大になると、相対的にカーソルが小さくなり短い距離の細かなスクロールを行うことが困難になる。そのために、今回は(d)の用語木を直接ドラッグすることにより夜スクロールを可能にした。これにより、巨大な用語木の場合でも細やかなスクロールを行うことが可能になった。

また、左に位置する(c)構成用語windowの用語をクリックすると、階層構造コンポーネントはその用語が中央に位置するように用語木を再描画する。

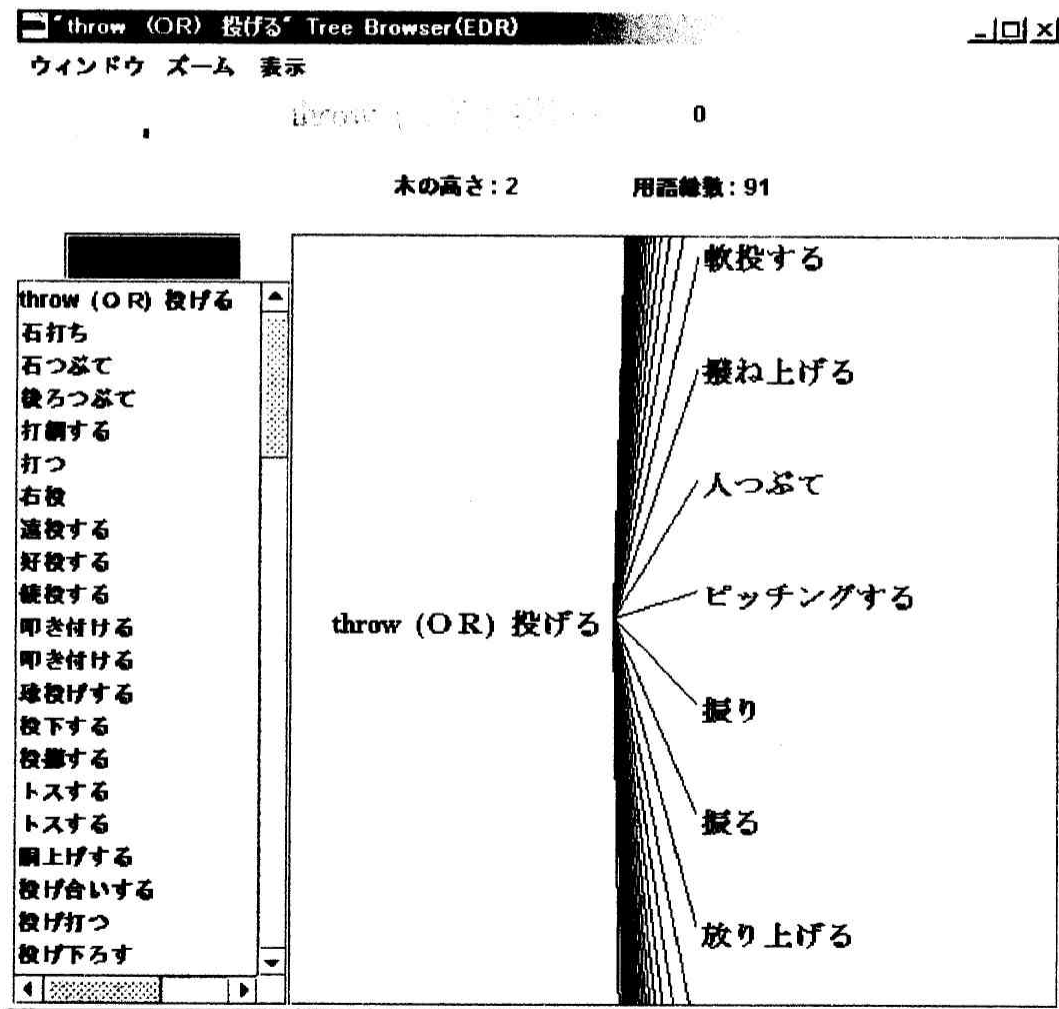


図 2.3 用語木 window

この(b)(c)2つのコンポーネントの導入により、100語以上からなる大きな用語木についても、自在にその構成用語およびその上位語一下位語を知ることが可能となる。

さらに、(d)用語木情報表示部分については表示する木の大きさを拡大縮小することが可能になっている。メニューバーの「ズーム」から「拡大」もしくは「縮小」を選択することで実行できる。加えて、メニューバーの「表示」から「横」もしくは「縦」を選択することにより木構造を横に表示するか縦に表示するかを切り替えることもできる。この二つの機能により、木構造の全体像が視覚的に理解しやすくなる。

上位一下位語の表示、関連語の表示、および用語の保存は、用語が示されるどの window からでも表示できる。これは、用語木 window 中の用語木情報表示部分に表示されている用語ノードからでも可能である。例えば図 2.3 において、「振る」のノードでマウスの右クリックを行えば、ポップアップ window が表われ、「GR」を選択すると関連 window が表示される。このように、どの window からでも必要な情報を相互に呼び出せることにより、用語間の関連をたどり必要な用語を探し出すことが可能となっている。

(4)の関連 window は用語同士が、どのような関係詞で関連しているかを表示する window である。図 2.4 にこの例を示す。

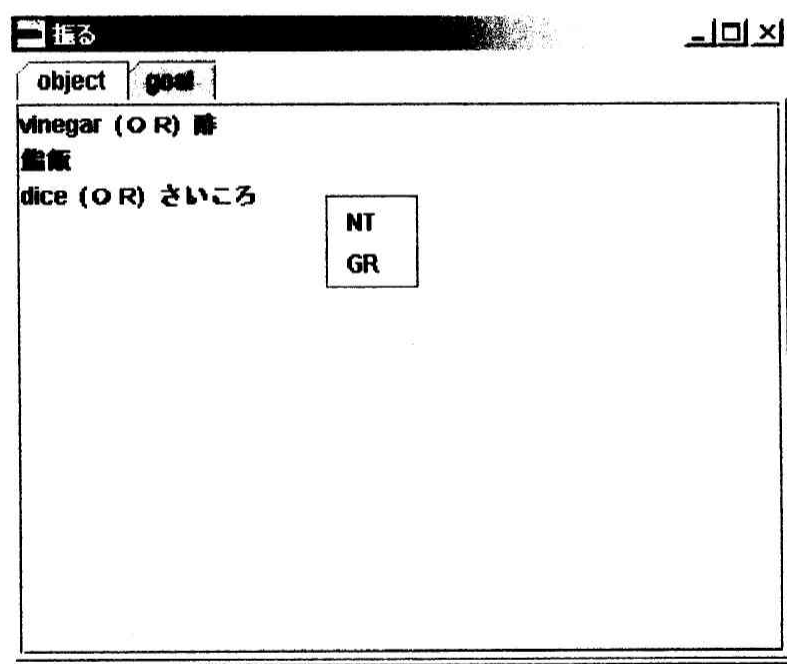


図 2.4 関連 window

ここで、先に用語間の関係について説明する。まず、2つの概念は関係子による2項関係として下記のように表現される。

概念 関係子 概念

EDR 概念辞書における関係子を用いた用語関係の記述例を下記に示す。

りんごを食べる → 食べる object りんご  
 ナイフで切る → 切る implement ナイフ

ここでは「object」と「implement」がそれぞれ関係詞となる。

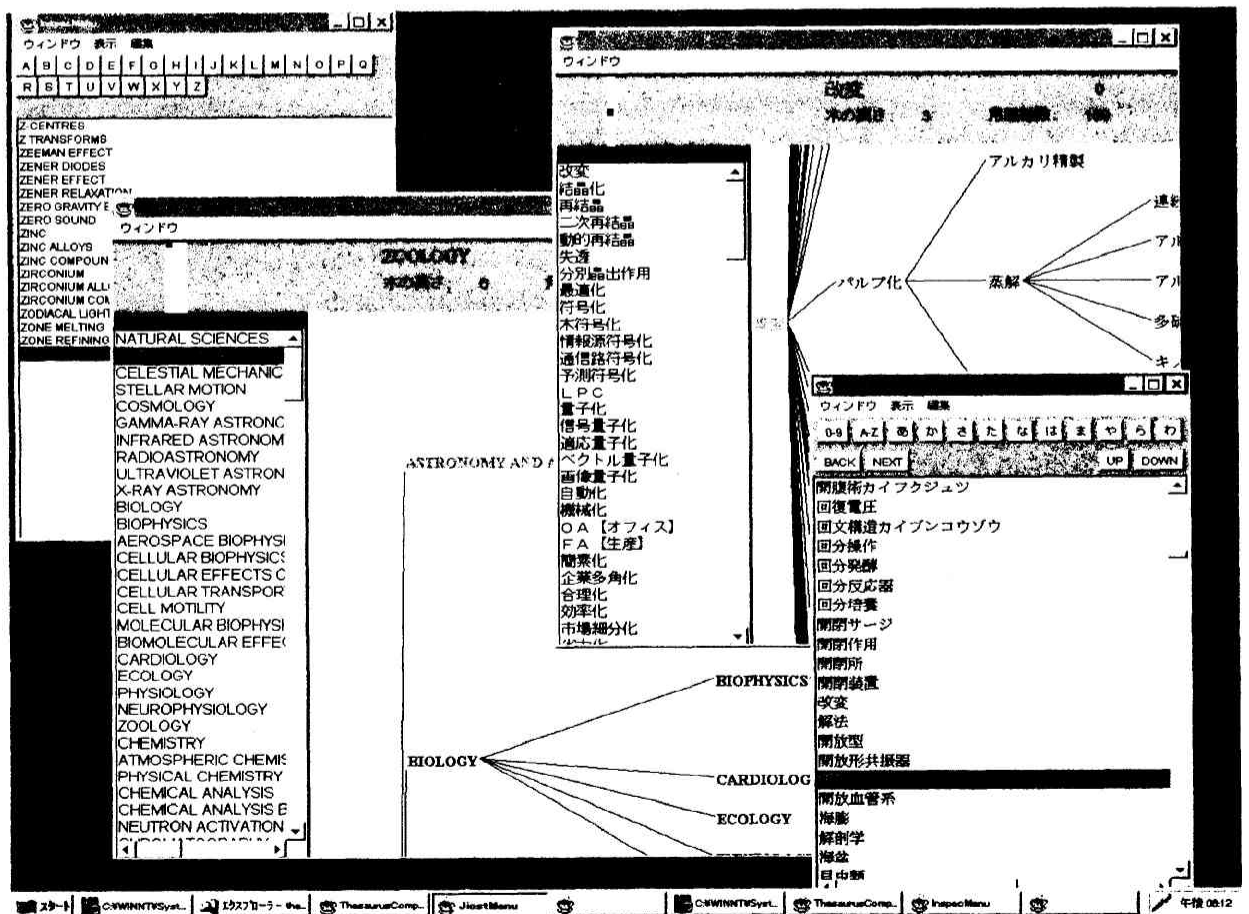
EDR 概念辞書の場合全部で 27 の関係詞によって各用語間が関係づけられている。以下がその例である。

- agent&(動作主体)
- a-object&(属性をもつ対象)
- material&(材料)
- goal&(主体の最後の位置)
- object&(対象)
- implement&(手段)
- source&(主体の最初の位置)
- place&(場所)

関係 window では window 上部のタブの部分に關係詞を表示し、そのしたの用語リストにその關係詞によって關係づけられている用語の一覧を表示している。

### 2.3 複数辞書間での使用例

本ブラウザは、ブラウザが稼動するPC(windows95/98/NT), WS(UNIX)のメモリーの容量が許す限り、同時に(1つのブラウザで)複数のシソーラスを利用することができる。図 2.5は、INSPEC、JICST、EDRの各辞書を同時に利用している例を示している。



## 図2.5 複数シソーラスの利用例

本ブラウザは、複数シソーラスを統一されたインターフェースで利用することができる。これに加え、次に説明する編集機能を用いることにより、複数の用語階層構造の統合や、複数シソーラスを元とした用語木データの編集などが容易に行えるようになった。

### 3. 用語編集機能

次に、当ブラウザに追加された編集機能について報告する。

用語編集の流れとしては、編集対象となる用語に関する用語木を表示させ、この用語木中で編集対象の用語を選択する。その上で編集機能を選択するとその用語に対する編集 window が表示され、それについての各種編集を行っていく。編集が確定されると、その編集情報に基づき(データ構造としての)用語木を操作し、変更後の木構造を再表示させる。

用語木を編集する際に必要な機能として以下の機能が考えられる。

- (1) 用語の追加
- (2) 用語の削除
- (3) 用語の属性変更
- (4) 用語(部分木)のコピー
- (5) 用語の詳細情報表示

これらの機能の詳細について以下に解説する。また、編集機能は用語木ブラウザ中で右クリックし表示されるポップアップメニューから「編集」を選択することにより行うことができる。

#### 3.1 用語の追加

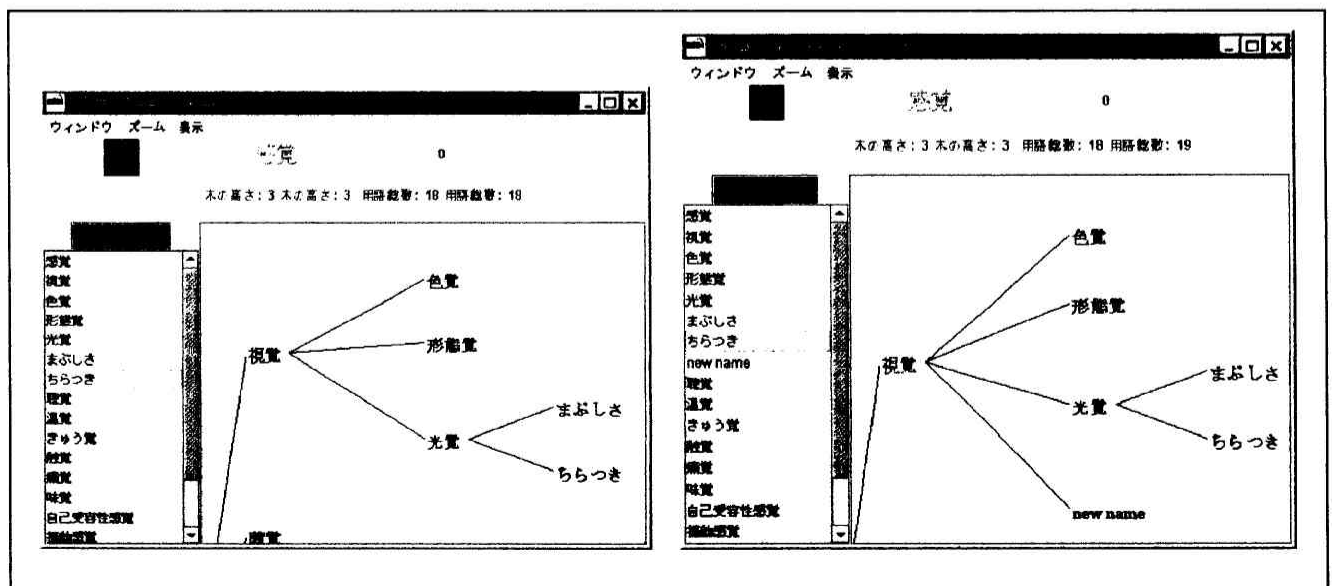


図 3.1 用語の追加例

この機能は、用語の狭義語として新規の用語を追加する時に用いる。編集機能で用語の追加を選択し、追加する用語名を記入すると選択された用語の狭義語としてその用語が追加される。図 3.1 において、左側は追加前、右側は、用語”new name”が追加された後の画面を示している。

実際にはポップアップメニューの「編集」から「ノードの追加」を選択することで実行できる。

### 3.2 用語の削除

この機能は、選択された用語を削除する機能である。用語木中から用語を削除したい場合には、その用語だけを削除するだけではなく、その用語の下位用語の処理方法についても考える必要がある。以上のことを踏まえ、今回は用語の削除方法について3つの方法を提案する。方法は以下の通りである。

- (1) 選択した用語以下の部分木の削除
- (2) 選択した用語のみを削除する
- (3) 選択した用語のみを削除し、その用語の狭義語のどれかと入れ替える

図 3.2a は削除操作の選択画面を示している。図 3.2a の右側、ノードの編集 window の「ノードの削除」部分で、上に提案した3つの削除方法を選択できるようになっている。

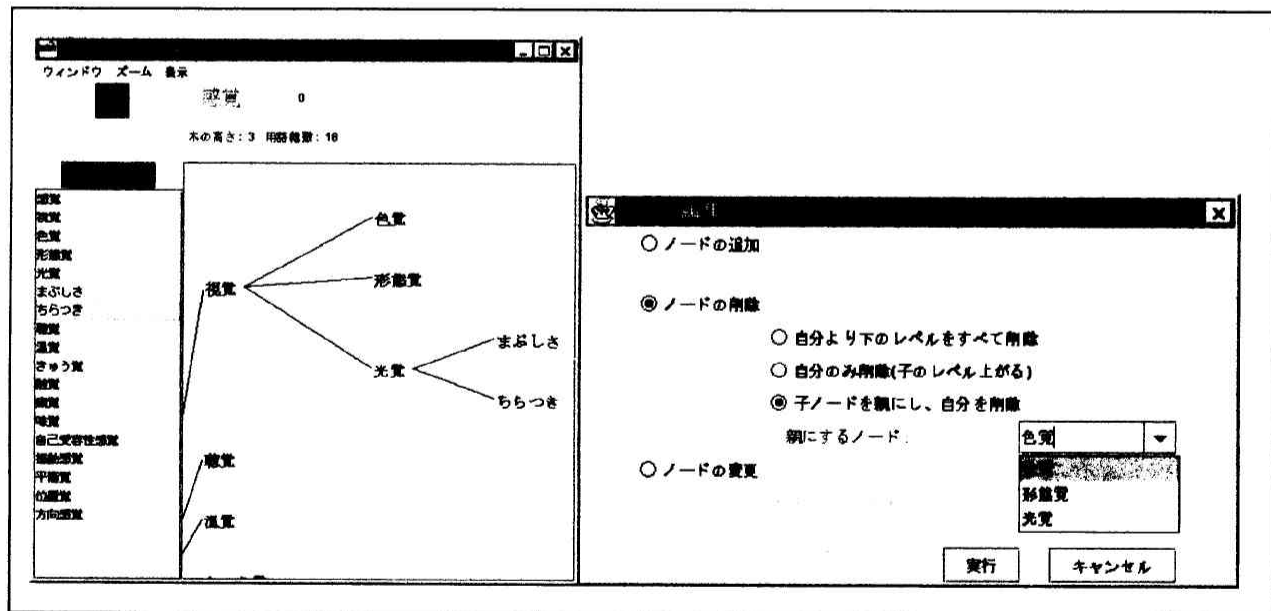


図 3.2a 削除操作の選択画面

次に、実際に削除を行った際の例が図 3.2b である。今回は(3)の操作を選択した場合の例を示している。左側が削除前、右側が「感覚」を削除し、下位語「形感覚」と入れ替えられた状態を示している。



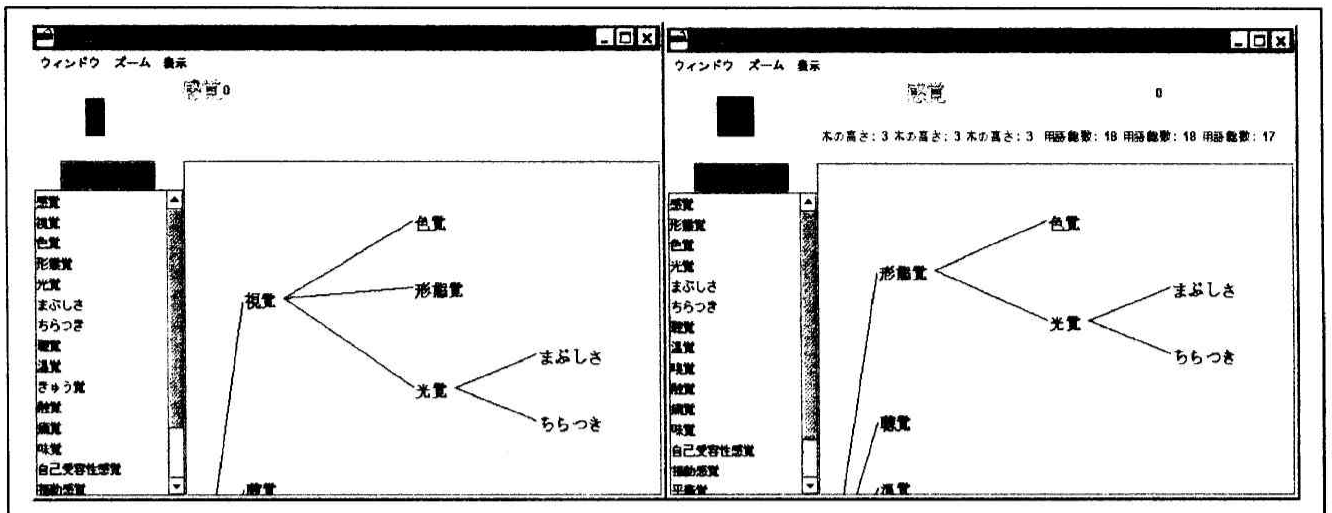


図 3.2b 用語削除の例

### 3.3 見出し語の変更

この機能は用語名などを変更する際に使用する機能であり、編集対象となる用語名を変更するときに用いる。図 3.3 にこの操作例を示す。

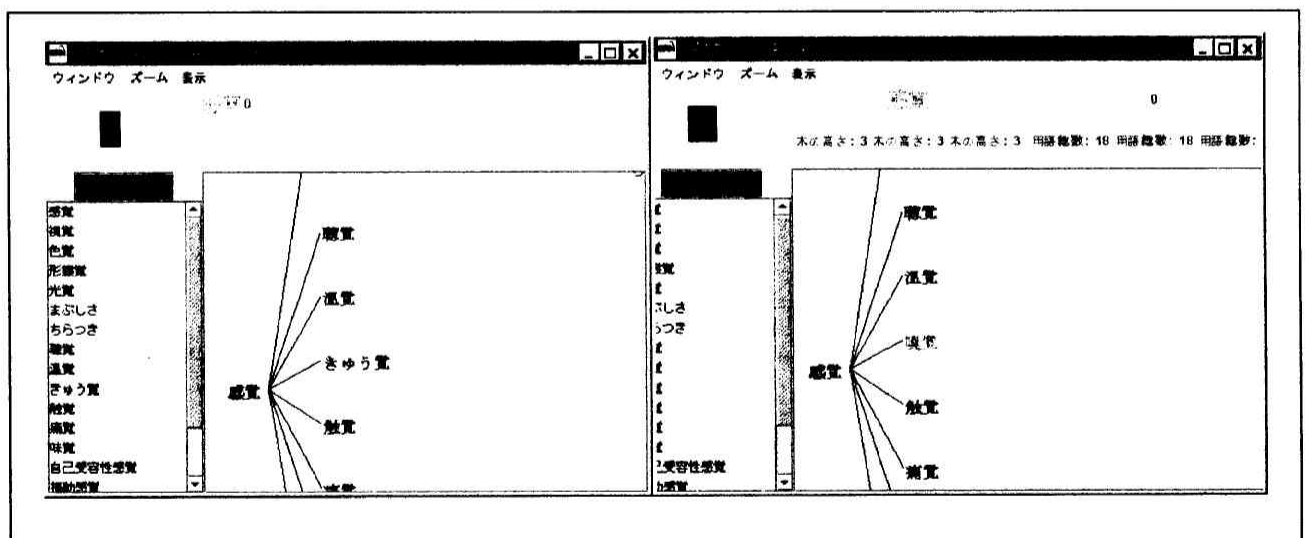


図 3.3 見出し語の変更例

### 3.4 用語属性の詳細表示・変更

用語木上では、そのノードの見出し語とその上下関係しか表示されていないため、その他のデータ(レコード番号・用語の読み・木番号など)の詳細な情報を参照することができない。用語の関係のみが記されているデータであればよいが、EDR 概念辞書のように、見出し語が同じでも概念が異なるようなもの、また概念見出しのないような概念をブラウザで表現する際に、用語の説明がないと用語木の表示・編集が不便になる。このような辞書を用語木上で

表示するために、用語の詳細を表示する機能を追加することにより、用語情報を参照することが可能になった。

感覚のプロパティ	
見出し語:	感覚
読み:	カンカク【セイリ】
木番号:	369
最上位語:	
下位語:	視覚, 聴覚, 温覚, きゅう覚, 触覚, 痛覚, 味覚, 自己受
関連語:	視認

OK

図 3.4 用語属性の詳細表示

### 3.5 用語(部分木)のコピー

指定した用語以下の部分木のコピー・貼り付け機能をも実装した。この機能により、ある用語の下位に他の用語以下の部分木をコピーすることができる。

## 4. 考察

上記の編集機能は、既存の木、部分木の編集に対して有効である。全く新規に木構造を生成する、あるいは木構造生成の対象となる用語リストのみが存在するような状況では、上記の機能では木の構築はその操作性において不十分と考えられる。木構造の生成は、用語の上位/下位用語を追加することであるが、これを1つ1つ行うのではなく、1回の操作で複数の用語を下位語化するような機能も必要であろう。

また、異なった属性群をもつ複数の用語木間での編集では、木構造以外の属性をどのようにするか、という問題が生ずる。このような編集状況に対処するためには、用語知識ベースのXMLデータベース化が有効であると思われる。

## 5. おわりに

当研究室では、数年来、用語知識ベースの操作ツールを開発・実装している。今回の報告では、従来の二次元視覚化表示に加え、同様の視覚化-操作感覚での編集機能を開発した。今後は、考察で指摘した問題の解決手段、さらにEDR電子化辞書<sup>[4]</sup>やNTT日本語語彙体系<sup>[5]</sup>のような意味関係で結合されたグラフ構造データの表示、編集機能を開発する予定である。

## 参考文献

- [1] 下村 央人 他.『汎用シソーラスブラウザの試作』.第 33 回情報科学技術研究集会発表論文集,pp.99-105(1997).
  
- [2] 下村 央人 他.『JAVA による索引支援ブラウザの試作 2』.第 35 回情報科学技術研究集会予稿集,pp.133-137(1998).
  
- [3] 後藤貴信、鈴木祐介、後藤智範. 階層構造をもつ用語データのためのBrowsing Tool. 情報知識学会第7回(1999年)研究報告会講演論文集. pp.57-60(1999).
  
- [4] EDR 電子化辞書. 日本電子化辞書研究所. 1995 年.
  
- [5] NTT コミュニケーション研究所 監修. 日本語語彙体系. 岩波書店. 1999 年.