

LispとGUIを用いたシソーラスエディタ

芝 勝徳 <shiba@kobe-cufs.ac.jp>

神戸市外国語大学図書館

シソーラスの生成・維持のための支援ツールとしてシソーラスエディタを試作した。エンジンとしてLispを選択し、GUIを使用して操作性を高めた。エディタの機能としてシソーラスの持つ基本的な語同士の関係の編集に対応した。約7千語をローディングしての実験においてある程度の結果が得られた。今後、基本機能の拡大やネットワーク上での共同作業が可能な方向に向かって開発を進める必要がある。

A development of thesaurus editor
with GUI based on Lisp engine

Masanori SHIBA

Kobe City University of foreign studies Library

I try to develop a thesaurus editor with user friendly GUI based on Lisp engine. This editor is specified basic inter-term relationship in a thesaurus. The result of experiment of this editor in practical use loading about 7,000 terms were good. I propose using a thesaurus editor under network environment.

1 はじめに

今回の試作の直接の目的は、図書館で伝統的につかわれてきた件名標目表をシソーラス構造に作り替える改版作業支援である。本エディタを使用するユーザ対象はシソーラスを構築する側のオペレータであるとし、シソーラスのユーザとしての索引語付与作業者や、検索補助手段としてシソーラスを使う一般の検索者を対象としていない。

本エディタの機能は人間の手作業では間違いをおこしやすいシソーラス内の語同士の関係のリンク構造の整合を機械的に補正し作業の効率化を図るもので、シソーラス構造を GUI を通して理解しやすく作業者にブラウジングする機能、語の生成や削除および編集、語間のリンクを生成・編集する3つの機能をもつ。

しかし、広い範囲でシソーラス生成・維持支援ツールの目的や機能を考える場合、今回の試作の範囲は部分的なものである。本来の本ツールの目標としては単なるシソーラスに特化したエディタ機能だけでなく、多様なシソーラスの人間対機械インターフェースの開発やシソーラス全体を媒体変換したりネットワークで配布するための交換フォーマットへの変形機能、さらにネットワーク上での共同作業としてのシソーラスの改訂・改版等を補助する機能を今後開発していく必要がある。そのための今後の課題についても考察した。

2 エディタ機能詳細

2.1 エディタ機能の範囲

シソーラスが生成されてから交換・伝達されユーザに利用されるまでの範囲を考える場合、3層構造を考えることができる。(図1参照)

最下位の層は通信・交換のための媒体を通じて伝達される交換フォーマットに固定されたシソーラスであり MARC と呼ばれる機械可読目録としての論理フォーマットとしての形式である。

中位の層は機械可読形式のシソーラスをデータベースにローディングしたデータベースのスキーマとして定義され、検索やリンクをたどることに最適化された機械内部のシソーラスである。

最上位の層は人と機械のインターフェースを通して人間が読める形としてのシソーラスの形態である。従来の印刷媒体に固定されたシソーラスはこの層でのみ存在しており、シソーラスの参照構造や階層構造を2次元の版面上で記号やインデント等によって表現している。今回この3層の構造のうち上位の2層を含むものとしてシソーラスエディタを作成した。

扱うシソーラスの論理構造は ISO 2788-1986 及び JIS X 0901-1991 に従う。ただし階層構造のうち全体部分関係と種類関係の分離はしていない。

データベース層については以下の理由により Lisp を使用した。

- DBMS の適性

シソーラス構造を表現するには RDBMS では無理があると判断した。複雑なリンク構造を RDBMS のモデルで表現するよりも、語をオブジェクトとしてとらえ、オブジェクト間のリンクで表現するほうが自然である。

- 語彙数

今回の試作の範囲においては扱う語彙数を最大約2万件までとするという条件から、すべての語オブジェクトを主記憶装置上に展開できるハードウェアが用意できる。

- ユーザ数

シングルユーザに限定し同時に複数のユーザからシソーラスへのアクセスはないとする。よって汎用 DBMS のもつ排他制御やトランザクションの整合性維持等のマルチユーザ環境での機能を必要としない。

表示層への GUI としては Motif を採用した。GUI 設計の意図としてはシソーラス全体の一覧性よりも、注目しているひとつの語を中心とした階層構造を認識しやすいインターフェースを目指した。シソーラス中の個々の語彙についてのスコープノートや分類記号等の記述は図書や論文の目録記述レコードに比較して少量であるためエディタとしての機能である記述の編集作業用のウィンドウは常時用意しない。あくまでも GUI の中心は階層ブラウザである。

もうひとつの重要な機能であるリンク操作については原則として、語同士は一度に1対1の関係でのみリンク操作ができることとし、片方の語はブラウザ上で現在注目している語に固定し、対になる語をスクリーン上でなるべく自由にポインティングできるようにした。さらにヒストリバッファを画面オブジェクトとして表示できるようにし、オペレータの思考がたどった経緯のリストからも語を選択できるようにしてある。

2.2 実装環境

- CPU R4400
- 主記憶 48Mbyte
- 必要ディスク容量 約 200Mbyte
- OS UNIX System V rel4.0

2.3 機能

- 語の生成・追加
- 語の削除
- 語の記述の編集
- 語同士のリンク作成 (USE ↔ UF NT ↔ BT RT)

2.4 組込規則

- ループチェック
- 不正リンク検出機能

- 非優先語同士の USE 関係
- 非優先語のチェーン
- 非優先語への階層関係リンク
- 階層関係をもつ優先語を非優先語に変換

2.5 GUI 及び操作

図 5-1-4 参照

2.6 実装したシソーラス

事前に基本件名表目標 第 3 版¹ 6,830 語をフラットテキストにして疑似シソーラス化した。これをさらにフィルタプログラムで処理をし Lisp 形式に変形した(図 2 4 参照)²。

3 使用結果

3.1 処理速度

起動時間には問題があるが、語彙の生成、語彙間のリンクの処理速度についてはほぼ問題がなかった。ループチェックについてもシソーラスの階層の最大の深さが最上位語から最下位語まで 13 程度で、かつ 5 つ程度の最上位語からの複数パスをもつ語彙を追加するという今回のシソーラスで最も負荷のかかる場合でも 0.5 秒以内に処理が可能で満足できるものであった。

3.2 不足した機能

リンク操作において、オペレータの試行錯誤が予想したよりも発生することがわかった。現状の仕様ではリンク操作のやり直しはできない。よって必要と感じたのは予報機能とやり直し機能である。特に予報機能については、操作の結果階層の深さがいくつになるとか、リンク削除の結果、語が孤立してしまうことのチェックをした結果の予報をオペレータに知らせて不用な試行を防ぐという意味で重要である。

4 課題

4.1 コンピュータ支援共同作業

実用的なシソーラス維持・管理については、ひとつの機関や図書館だけで対応できるものではなくってきている。増える語彙に対しての比較的短いインターバルでの更新や、単なる更新ではなく改訂・改版にかかわるシソーラスの構造や大幅な語彙の整理等の作業についてもコンピュータ上での共同作業(CSCW)が必須となるであろう。またそうしなければ、数年に一度の改訂し

¹ 日本図書館協会監修の件名表で一般に出版される図書に対して件名標目を付与するために編集されている

² 参照構造が単一のルールで構成されている時、単純なアルゴリズムで参照関係から階層関係を生成できる

かできないようなシソーラス、特に新しい分野の専門シソーラスの利用価値はなくなっていくともいえる。

4.2 ネットワーク対応

CSCW が必要となると、ネットワークに対応した CSCW もまた必須となる。ネットワーク上でシソーラスの利用が可能となればネットワークシソーラスとして (HTML への変形等) のネットワーク経由の利用も可能となるであろう。一つのマスターシソーラスへのアクセスの集中をさけるため、分散したネットワーク上でのシソーラスの整合性の維持が新たな課題となるであろう。

4.3 マルチメディアシソーラス

記述内容がマルチメディア化することが考えられる。例えば動植物の固有名の辞典類については既に一部マルチメディア化している。これらを取り込むもしくはこれらにリンクすることにより、文字による注記よりも効率よく認識できるであろう。

4.4 DBMS を使用した再実装

上のような課題を解決するためには、シングルタスクの Lisp をエンジンにした今回の方法とはとれない。また語彙の増加はシソーラス全体を主記憶上に一度に展開できなくなる可能性をもっている。本格的にネットワーク対応をも視野にいれた場合、汎用的なデータベースエンジンを使用する事は必然の解となる。

現在、試作した Lisp エンジンオブジェクト指向データベース (OODBMS) エンジンに置換する作業を行っている。これに伴い、語彙オブジェクトとリンクオブジェクトの分離、語彙単位のバージョン管理等が可能になる予定である。

図1

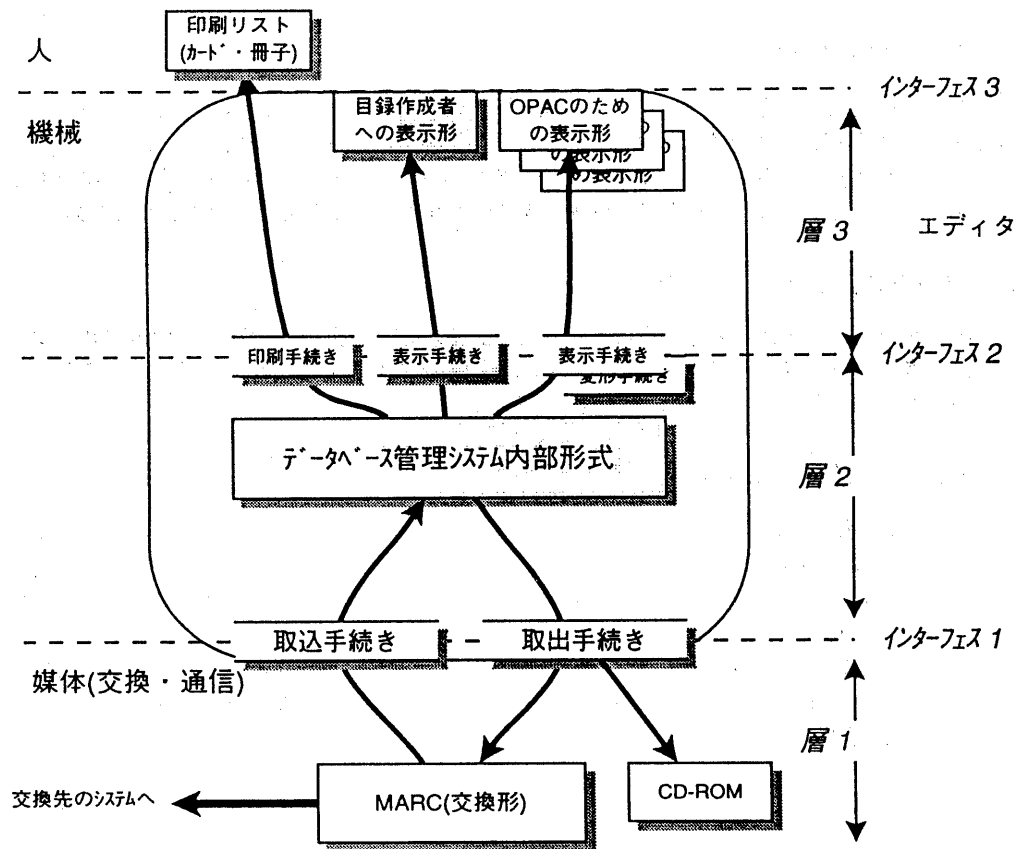


図2

- カンショウキョ 観賞魚 666.9 >:金魚, 熱帯魚 <:魚類
- キョウメイ 魚類 487.5;663.6;664.6 >:観賞魚, 金魚, 淡水魚, 熱帯魚, 個々の魚の名 (例:さけ)
<魚 (ウオ), 魚 (サカナ) <:水産資源, 水産動物, 水産物, 脊椎動物
- キンキョ 金魚 666.9 <:観賞魚, 魚類, 淡水魚
- タンシキョ 淡水魚 481.75;487.5;664.2;664.69;666.69
>:金魚, その他個々の淡水魚名 (例:こい)
<:魚類, 水産増殖, 淡水動物
- ネツキョ 熱帯魚 487.52;666.9 <:観賞魚, 魚類

図3

ID: BSH300108100
 HD: 観賞魚
 YM: カンショウキョ
 BT: 魚類 BSH300142300
 NT: 金魚 BSH300146400
 NT: 熱帯魚 BSH300522200
 NDC: 666.9

ID: BSH300142300
 HD: 魚類
 YM: キョイ
 UF: 魚 (ウオ)
 UF: 魚 (サカナ)
 BT: 水産資源 BSH300353000
 BT: 水産動物 BSH300353400
 BT: 水産物 BSH300353500
 BT: 脊椎動物 BSH300385600
 NT: 観賞魚 BSH300108100
 NT: 金魚 BSH300146400
 NT: 淡水魚 BSH300417200
 NT: 熱帯魚 BSH300522200
 SANOTE: 個々の魚の名 (例: さけ)
 NDC: 487.5
 NDC: 663.6
 NDC: 664.6

ID: BSH300146000
 HD: 金魚
 YM: キンキョ
 BT: 観賞魚 BSH300108100
 BT: 魚類 BSH300142300
 BT: 淡水魚 BSH300417800
 NDC: 666.9

ID: BSH300417800
 HD: 淡水魚
 YM: カスイキョ
 BT: 魚類 BSH300142300
 BT: 水産増殖 BSH300353300
 BT: 淡水動物 BSH300418100
 NT: 金魚 BSH300146400
 SANOTE: その他個々の淡水魚名 (例: こい)
 NDC: 481.75
 NDC: 487.5
 NDC: 664.2
 NDC: 664.69
 NDC: 666.69

ID: BSH300522200
 HD: 熱帯魚
 YM: ネットイキョ
 BT: 観賞魚 BSH300108100
 BT: 魚類 BSH300142300
 NDC: 487.52
 NDC: 666.9

図4

("観賞魚" "BSH300108100"
 (YM
 "カンショウキョ"
)
 (BT
 ("魚類" "BSH300142300")
)
 (NT
 ("金魚" "BSH300146400")
 ("熱帯魚" "BSH300522200")
)
 (NDC
 "666.9"
))

 ("魚類" "BSH300142300"
 (YM
 "キョイ"
)
 (UF
 ("魚 (ウオ)" "BSH300029600")
 ("魚 (サカナ)" "BSH300249700")
)
 (BT
 ("水産資源" "BSH300353000")
 ("水産動物" "BSH300353400")
 ("水産物" "BSH300353500")
 ("脊椎動物" "BSH300385600")
)
 (NT
 ("観賞魚" "BSH300108100")
 ("金魚" "BSH300146400")
 ("淡水魚" "BSH300417800")
 ("熱帯魚" "BSH300522200")
)
 (NDC
 "487.5"
 "663.6"
 "664.6"
)
 (SANOTE
 "個々の魚の名 (例: さけ)"
))

("金魚" "BSH300146400"
 (YM
 "キンキョ"
)
 (BT
 ("観賞魚" "BSH300108100")
 ("魚類" "BSH300142300")
 ("淡水魚" "BSH300417800")
)
 (NDC
 "666.9"
))

図5-1

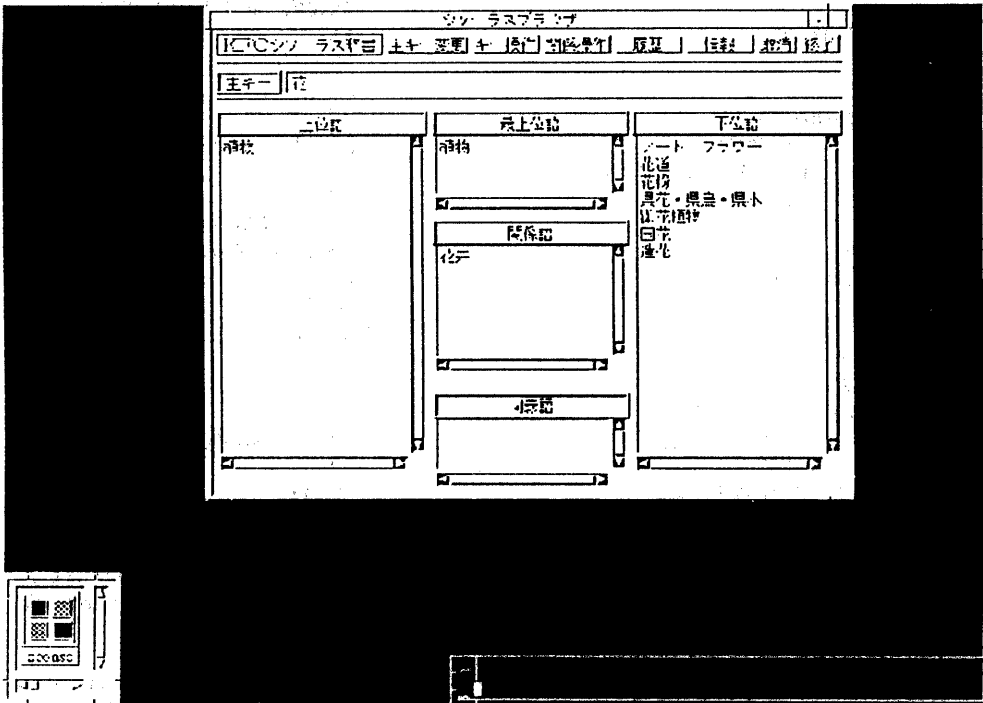


図5-2

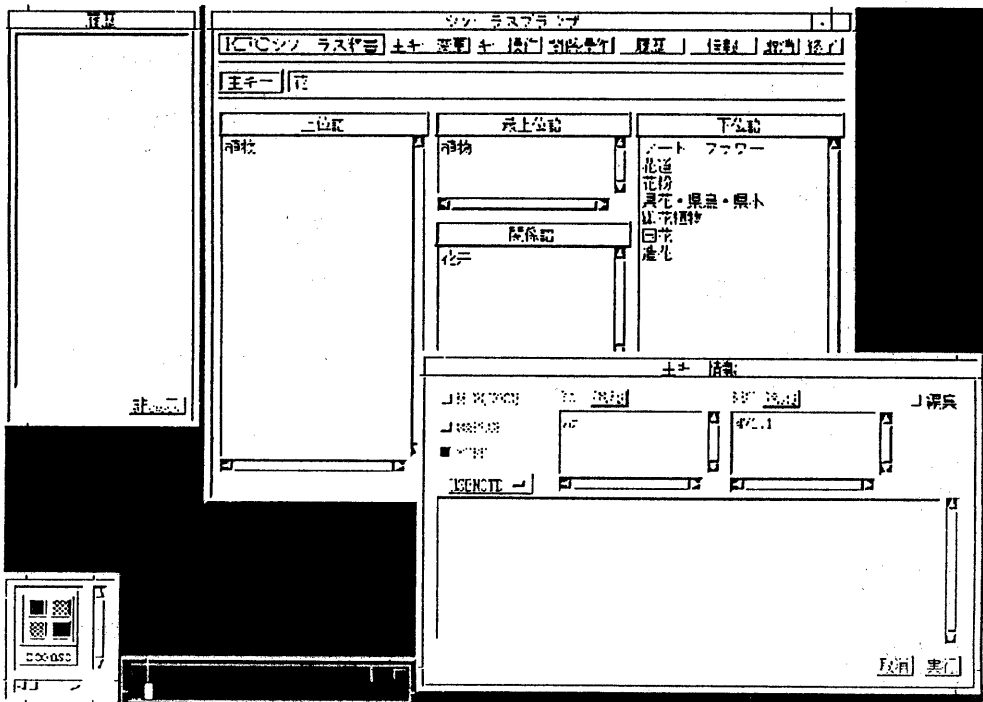


図5-3

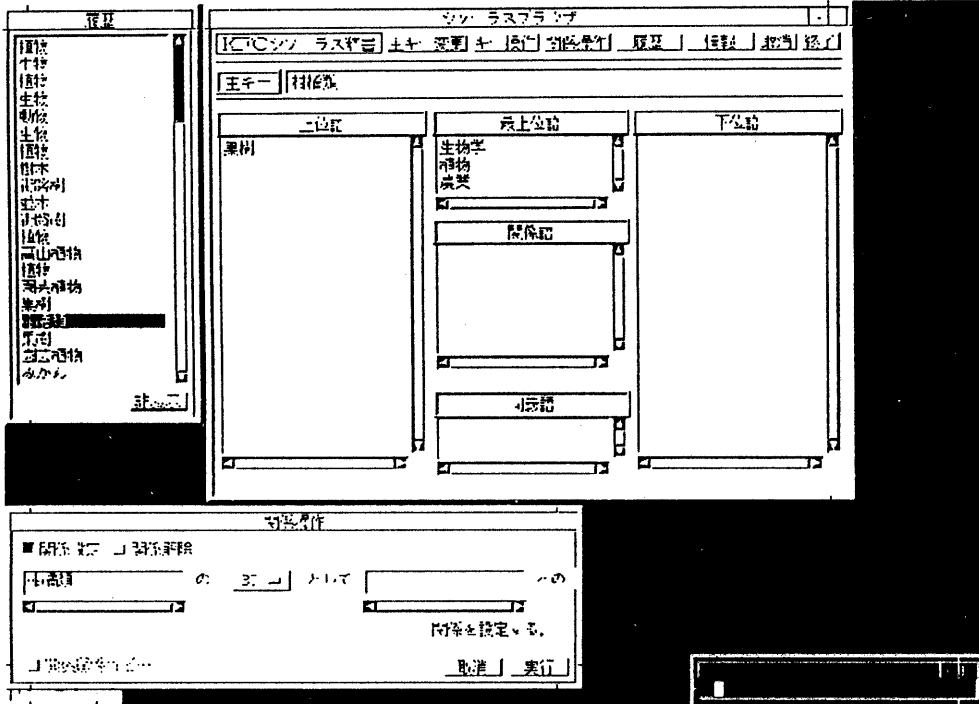


図5-4

