# 木簡研究支援データベースシステム
## —知見と仮説に基づく再構造化
# Scientific Database Management System for Research of Wooden Slips
## — Restructuring of Data based on Hyposeses and Viewpoints

森下淳也，大月一弘 (神戸大学国際文化学部)，上島紳一 (関西大学総合情報学部)，
大庭脩 (皇学館大学文学部)，杉山武司 (姫路獨協大学情報科学センター)
**Jun-ya Morishita, Kazuhiro Ohtsuki(Kobe University),**
**Shinichi Ueshima(Kansai University), Osamu Ohba(Kohgakukan University),**
**Takeshi Sugiyama(Himeji Dokkyo University).**

神戸大学国際文化学部　〒657 神戸市灘区鶴甲 1-2-1
**Faculty of Cross-cultural Studies, Kobe University**
**1-2-1 Tsurukabuto, Nada Kobe, Hyogo 657 Japan.**

**あらまし:** 木簡研究支援システムのために，半構造化データの視点に基づく再構造化について，利用者の操作モデルを階層構造グラフによるグラフ操作で表すことを検討する．これらの操作は従来のデータベースシステムにはないものであり，データと視点の意味に基づいた対応を必要とする．問題点と幾つかの適応例を提示する．

**Summary:** Restructuring the semi-structured data on scientific database system for research of wooden slips, we examine the graph operations based on the hierarchical graph model so as to give researchers' operation schema based on their viewpoints. The operation schema are not supported by ordinary database management systems and depend on semantics of the data and viewpoints. Problems and some examples are also given.

**キーワード:** 科学技術データベース，木簡，半構造化データ，オブジェクト指向，オブジェクトビュー
**Keywords:** scientific database, wooden slips, semi-structured data, object oriented, object view.

## 1　はじめに

近年，文献情報データや環境情報データなどの科学技術データベースが注目されてきている [1, 2]．これらはデータが収集された時点で，データに対して大雑把な構造しか与えられていない場合が多い．言わば半構造化の状態のまま，構築されているという特徴がある．また，確定したスキーマを持つ場合でも，そのデータを利用する段階で，視点の違いや仮説などを盛り込んだ思考実験のような自由度を与えたいと考えられる．

木簡研究支援システムは，中国敦煌遺跡から出土した木簡一千本，中国居延遺跡の木簡一万本を対象にしたシステムである [3]．これらの木簡 (図1) は，発掘された時点で基本的なデータを収集し，その後，木簡の釋読が進むにつれて文字データが収集される．さらに，釋読文から個々の意味が汲み取られるという過程を経て，データベースに格納される．このようなデータは，

図 1. 木簡

- 文字も分からない状態のまま，長い期間保存される．

- データの解釈が後から発見される.
- データの意味が一般に確定するには，長い期間の研究が必要である.

など，特異な振舞いが見られる.

我々はこのような大雑把なスキーマしか持たない半構造化状態のデータに対して，元のデータベースのデータを保持しつつ，利用者の視点に応じて，属性付けを行うことで様々な構造化を同時に表現するデータモデル，階層構造グラフを考案した [4]. そして、そのデータモデルを用いて木簡研究支援システムで必要な属性付け機構を検討してきた [8].

本稿では、階層構造グラフを用いてデータを再構造化するために、利用者の行うデータ操作を検討し、それをデータモデルにおけるグラフ構造の更新によって実現することを議論する。利用者に基づくデータの再構造化の過程は、通常のデータベースの操作（大量のデータから必要な情報を取り出す、或いは特定の情報を更新する）とは、概念的に大きく異なるものである。その仕組みを実現するには通常のデータベースの考え方とは異なるアプローチを必要とする. そのため我々は木簡研究者や科学者のデータを扱う利用者の操作モデルを考えることにする. そのモデルに従って，データモデルの操作を定義し，円滑にデータベースシステムとの関係を形成することを吟味する.

従来のデータベースシステムの特徴を掲げれば，

- 大量のデータを対象として高速に処理が行える.
- データベース自身は常にどのような時点で見ても正しく矛盾を含まない.
- データベースのスキーマは固定されていて，これが変更される時にはデータベースを再構築する必要がある.

これに対して，科学者が資料，データを吟味する作業は，

- 特定の内容を注意深く調べるため，一つ乃至少数のデータを大きくクローズアップする.
- 自分の関心のあるデータのみを収集して，自分の世界を構築する.
- 自分の仮説や知見を検証する際に，より大きなデータ集合に目を向ける.

というデータベースとは異なる特徴を持っている.

階層構造グラフでは，データベースのスキーマを随時変更できるようにするために，固定的なスキーマを持たず，属性集合を保持できるオブジェクトをデータの基本構成要素として採用している（スキーマレス）. この入れ物は全てが同じメタな構造を持っているため，データベースとしての再構築の必要のない環境を提供できる. 無論，データベースの本来持っている高速性は損なわれる. この構成要素をグラフでつなぎあわせることで，データベースから見れば，全てがデータで構成される構造体が出来上がる（インスタンスベース）. この構造体の活用方法が，上で述べた利用者モデルによる再構造化である.

以下に階層構造グラフを紹介し，再構造化のデータ操作を論ずる. なお，この内容は一部，[6, 7, 8] に基づいている.

## 2 階層構造グラフ

階層構造グラフは，ノードと枝からなるサイクルのない有向グラフである [4]. 各ノードと枝には属性集合を持つオブジェクトへのリンクが張られている. リンクされるオブジェクトはオブジェクト・アイデンティティと属性集合からなる. 次のような形式のものである.

$$O \equiv \langle Oid, \{a_1 : v_1, \ldots, a_n : v_n\}\rangle. \qquad (1)$$

ここで，$Oid$はオブジェクト・アイデンティティであり，$a_i$は属性，$v_i$は値である. この属性集合の$a_i$と$v_i$はそれぞれ次の様に⟨attribute⟩と⟨values⟩で定義されるデータである.

```
⟨attribute⟩  ::=  symbol,
⟨values⟩     ::=  ⟨value⟩                          (2)
              |   ⟨value⟩,⟨values⟩,
⟨value⟩      ::=  string|int|...|oid.
```

オブジェクトは3種類あり，それぞれ基本オブジェクト，カテゴリ，関係オブジェクトと呼ばれる. 基本オブジェクトは元々の半構造化状態のデータを格納するもので，グラフの最下位のリーフに置かれる. カテゴリはノードに置かれ，視点を表す. 関係オブジェクトは枝に置かれ，視点とデータの間に固有の情報が書き込まれる. 図2に階層構造グラフの例を示す. ノードと枝は，リンクされたオブジェクトの属性集合$\alpha_i, \beta_j, \rho_{kl}$を各々持つ（四角で表される）.

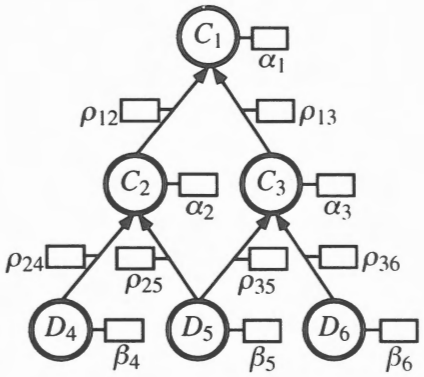


図 2. 階層構造グラフ

このグラフからカテゴリを定めて，下位のノードまでのサブグラフを取り出すと，その下位のノードのデータだけでなく，間に付与されたすべての属性を取り出すことができる．また，同じ下位ノードを対象としてもカテゴリが異なれば，全体として異なる属性集合を取り出せる．このサブグラフを仮想的なオブジェクトとして捉えることで，視点に基づくいくつものビューをデータに与えることができる．

# 3　再構造化のデータ操作

研究活動を支援する機能を考えるに際して，通常のデータベースの利用を振り返ってみると，データベースシステム自身の持っている機能は少なく，多くはデータベースを操作するアプリケーションによって実現されるとの見方が大勢を占めている．

データベース本来の機能とは，大量なデータからの高速な検索と新規データの作成や更新に対する円滑で矛盾のない処理である．これらは全てデータベースという大きな一つのものに対してメッセージを送り，その全体からの応答として，検索結果や更新結果を得るものである．これは本質的に非手続的で大域的な操作である．勿論，例外はあるにしろ，データベースとはデータ集合に対する処理を規定するものであり，ここで我々が念頭に置いているような，特定のデータを対象とした逐次的な細かいデータ操作などは本来のデータベース処理の対象とはならない．

そのため，利用者が手作業でデータを付与しながら，徐々にデータベースを構築していくような，逐次的な処理の管理という概念は，従来のデータベースシステムの枠組みには当てはまらない（勿論，データを追記した場合の無矛盾性などの機構は存在するが）．半構造化データにおける利用者の自由で動的なデータ生成を支えるためには，全体の保全といった消極的な立場ではなく，より積極的な操作モデルをシステムが備えている必要があると思われる．

ここでは，階層構造グラフに基づいて，利用者の操作をモデル化して実際のグラフ操作をそれに当てはめることを考える．

## 3.1　操作モデル

本簡研究者のような利用者が資料を扱う時には，全体を眺めるというよりも，自分の思い付くままに，いくつかのデータに当りを付けながら，自らの着眼を得るというような過程を経ると思われる．重要なのはデータベース全体ではなく個々のデータであり，その中を自由に散策することからインスピレーションを得ると思うのは，それ程不思議なことではない．また，そのような眼鏡に適ったデータを集めて，自分の世界を構築する．そこで自らの知見を記してみて，もう一度，眺め直すといった過程を繰り返すものと思われる．

このような過程を階層構造グラフに従って，たどっていくと次のようになる．

1. **散策**
   元々の半構造化データの中を自在に眺めながら，自分の必要とする（或いは琴線に触れる）データを選ぶ．そのためには，データ自身を自由に閲覧できる機能が必要である．また，従来のデータベースの操作として，検索によって対象を選ぶこともできる．このため検索結果に加えて利用者が閲覧して選んだものも同等に扱えるようにシステムを設計しなければならない．

2. **集合化**
   選ばれたデータを自分の手元に置くことは，従来ならデータベースの外側に取り出すことを意味するが，階層構造グラフではカテゴリオブジェクトを生成し，その下にデータを収容することで，データベース自身に自分の視点を格納することができる．これは従来のデータベースにはない支援環境に必要な機構である（図3）．

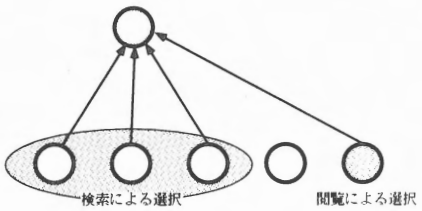


図3. 集合化

3. **属性付け**
選ばれたデータに対して補足すべき情報を付け加える．これはカテゴリオブジェクトとデータオブジェクトの間のエッジに関係オブジェクトとして格納する．この補足情報は，カテゴリとデータの間に付与され，他の利用者からのデータの参照には影響を与えない（図4）．

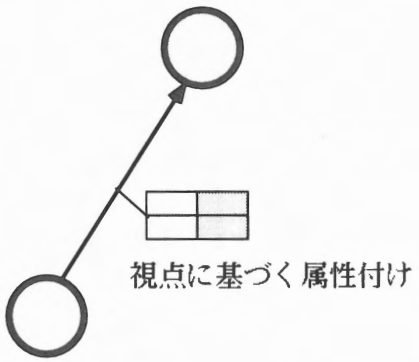


図4. 属性付け

4. **仮想オブジェクトによる確認**
カテゴリとデータのサブグラフ（仮想オブジェクト）を単位としてオブジェクトを見ることで，属性伝播による補足情報も含めたデータを仮想的に見ることができる．これをデータと同一視することで，新しいデータを得るのと同等なことが実現する．

5. **検証**
この仮想オブジェクトをデータとして，そこで得た知識を利用して更に他のデータの閲覧，検索を行い，検証を重ねる．

6. **発展**
このようにしてできたカテゴリは一つの視点を表しているが，検証によって更にこれが階層化されることが考えられる．階層化には大きく2つのパターンがある（図5）．

- 一般化
調べていたものが，より一般化されることで，他のカテゴリが想起され，幾つものカテゴリをまとめ上げる上位概念がカテゴリの上に置かれるようになる．
- 詳細化
よく調べることで，カテゴリの下に複数のサブカテゴリが生み出されて階層化される．

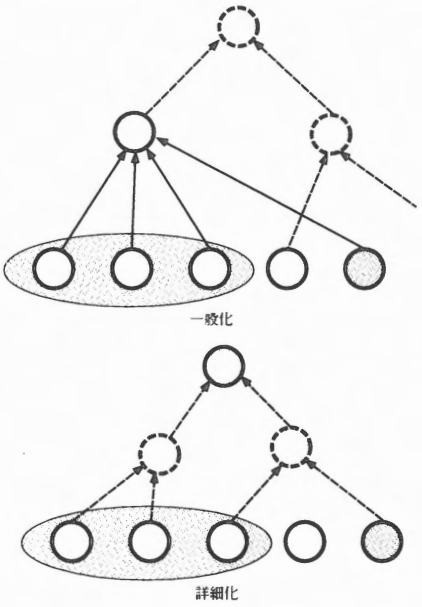


図5. 一般化と詳細化

このような過程を経て，階層構造が徐々に構成されていくと考えられる．そして，この形で構成されたグラフ自身がデータの再構造化の帰結であり，ここから仮想オブジェクトを用いて取り出されるデータが具体的な半構造化データの構造化された姿である．

本来，分類や集合化といった操作はデータベースに保存されるものではないが，それを保存することで明確な属性が付与され，その結果，構造が確立すると考えている．その過程をデータベースが内蔵することでデータの再構造化の過程をデータベースが支援できるようになっている．

## 3.2 典型的な利用形態

この再構造化の仕組みとグラフの特性を使って，幾つかの利用の形を考えることができる．ここでは協調作業環境と履歴管理について説明する．ま

た，具体的な活用として実際のメーリングリストの構造化について検討する．ここには面白い難点が現れる．

### 3.2.1 協調作業環境

一つの半構造化データに着目する．これは例えば，共著論文の下書きのようなものを想定している．これに対して，幾つかの視点から補足がなされるとしよう．この場合の視点は共著者がそれぞれ論文の下書きに朱を入れたもの，コメントを付与したものと考える．この視点を全て束ねるカテゴリを生成して，そのカテゴリからデータを見た場合（図6），仮想オブジェクトは全てのコメントを含むデータとして取り出すことができる．これは，視点の独立性とグラフの多重な属性伝播を用いた応用例である．

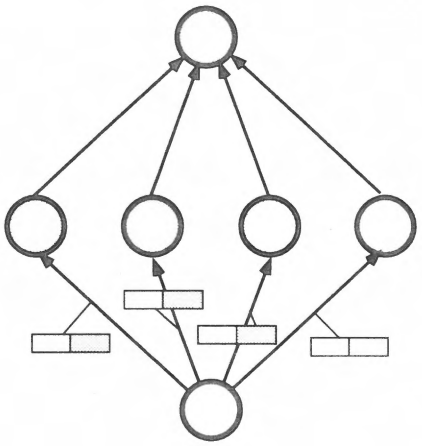


図6. 協調作業

### 3.2.2 履歴管理

法律の条文やコンピュータプログラムのようなデータを考える．これらに新たな更新がかかる場合，新しい改定データやプログラムの修正データを直接，元データに対して更新するのではなくカテゴリを用いて関係オブジェクトに記述することで，全ての履歴が保存されたデータベースを作ることができる（図7）．特定のカテゴリからの仮想オブジェクトによって各々のバージョンのデータを取り出すことができる．

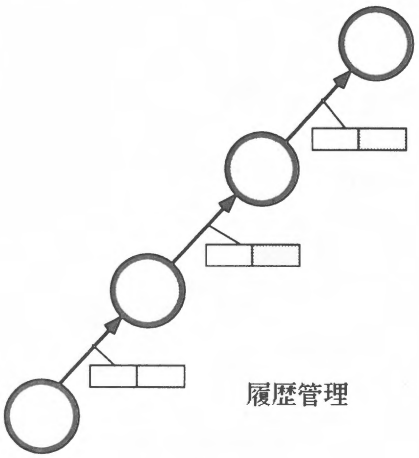


図7. 履歴管理

### 3.2.3 メーリングリストの例

この例は電子メールの実際にメーリングリストとして交わされたデータに基づいている例である．内容は特定の大学の講義に関して，まとめや質問を学生が交換しているものである．メーリングリストは電子メールを複数の人間で共有するシステムである．このデータから内容に関する情報を整理するためにシステムを用いている（図8）．最下層のデータは個々の電子メールである．対象となる話題に応じてカテゴリが生成され，その下に格納され補足されている．また，話題が講義内容であるため，更にその講義を表すカテゴリが上位に付与されている．

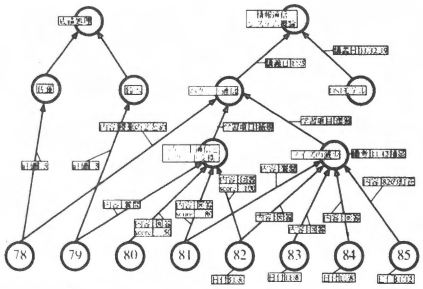


図8. メーリングリスト

この例で注目すべきは，話題にまたがって入っているデータである．複数のカテゴリに属することは問題ない（図8の「佐藤」と「パケット通信」に属するケースを参照）．本来，それが許されるようにシステムが構成されている．しかしながら，「パケット通信」のサブカテゴリの双方に含まれる「81」や「82」のデータは,「パケット通信」からの仮想オブジェクトによって，異なる2つの値が同じ属性“学習項目”に付与されている．これ

をそのまま伝播させたとするとこの内容が何を表しているのかわからなくなってしまう．

この難点は元々，話題という視点に対して「81」と「82」のデータが一度に2つの話題を含んでいることに起因している．これは元データの不完全さ（話題という側面についての）から分離すべき2つのものが一つのデータに含まれてしまったと考えるのが妥当である．従って，元データの部分を参照する，或いは断片化するデータ操作を何らかの形で導入する必要がある．また，この逆のケースも有り得る．即ち，複数の半構造化データを合成することで一つのデータと見なす場合である．これらについては，現在，検討中である．

半構造化データの不完全さはその視点に強く依存している．従って，どのようなものを対象とするかで，既に完成されたデータベースのデータもまた，我々の操作すべき対象となる．しかしながら，視点に基づいた再構造化は，その内容に強く依存している．これについて一般的な対応法を導入することは困難であると考えられる．それよりも，矛盾や冗長性を関知して通報する監視システムや，断片化や合成を容易に行える支援機構を導入することで対処できないものかと考察を進めている．

## 4 プロトタイプシステム

プロトタイプシステムは2つの部分から構成されている。

- **データベースカーネル**
  データオブジェクトやカテゴリ，階層構造グラフを収納し，データベースに関する諸操作を行なう核の部分である。この部分は，lispの一種であるscheme言語の拡張の，ELKというシステムを使用している[9]。

  このようなlispシステムをプロトタイプに採用した理由は，(1) インタープリタであるため，変更をすぐにシステムに反映させて，即座に結果が得られる点でプロトタイプとして優れている。(2) Lisp言語は内部構造を全てリスト形式で保持していて，殆ど全ての要素がリファレンスであるため，我々のオブジェクトを擬似的に表現するのに適している。また，インタープリタとしてメモリ，ディスクの区別なくリファレンスを表現できるオブジェクト指向データベースを容易に真似られる。等が挙げられる。

  このlispによるシステムだけでも，我々のシステムはその結果の入出力を除いて，動作する。操作のインターフェースはコマンドラインユーザインターフェース(CUI)となる。

- **ユーザインターフェース**
  利用者がデータベースと対話をおこない，結果を適当な形で表現する部分を受け持つ。ここは，コマンド言語として著名なtcl/tkの拡張版であるExpectkを用いて記述している[10]。

  このシステムも，インタープリタである。Expectkは対話的にウィンドウシステムを構築でき，マウス操作をコマンドラインの入力に変換できる機能を持っている。この為，細かなデザインをシステムの再構築をおこなわずに，その場で反映することができる。また，コマンドラインの入出力を内部で処理できるので，どのようなものにもグラフィカルユーザインターフェース(GUI)を与えることができる。

このプロトタイプシステムでは，格納されているデータはテキストと画像であるが，階層構造グラフ上を自在にtraverseでき，カテゴリの生成/消去ができる。カテゴリを選んで仮想オブジェクトを閲覧でき，属性について検索が可能である。本モデルが，データに種々の属性を付与し多様な属性構造を表現できることとデータが持っている意味に従った構造化に適していることをこのシステムで確認できる。但し，このプロトタイプシステムでは，同時に複数の利用者が操作することは実現できていない。

図9及び10に木簡と上で述べたメーリングリストの動作例を示す．

## 5 終わりに

階層構造グラフを用いた視点に基づくデータの再構造化の過程を考察した．この過程を支援するためには，データモデルの標準的な操作体系に，意味に応じて半自動的に利用者を助けるグラフ操作の監視機構のようなものを考案する必要があ

る．また，半構造化データの断片化や合成も実現する必要が有る．或いはグラフ自身に特別な要素を導入して，役割を明示することも考えなければならないかも知れない．更なる検討を必要としている．

木簡システムのようなデータの特徴(大まかに格納されたデータが認識された属性として取り出されるような場合)は，木簡独自のものではなく，科学技術データを利用する立場で考えた場合，より広い範囲に存在すると思われる．グラフ構造とデータをリンクすることで，グラフからの伝播ビューによって自在なデータの描像が得られることから，データの再構築のインターフェースとしてこのデータモデルが活用できるものと期待される．

## 参考文献

[1] IEEE Computer Society, "Special Issue on Scientific Databases," Bulletin of the Technical Committee on Data Engineering, Vol.16, No.1, Mar. 1993.

[2] Zdonik, S., "Incremental Database Systems: Databases from the Ground Up," Proc. of the 1993 ACM SIGMOD International Conference on Management of Data, Washington DC, USA, pp.408–412, May 1993.

[3] Ueshima, S., Ohtsuki, K., Morishita, J., Qian, Q., Oiso, H. and Tanaka, K., "Incremental Data Organization for Ancient Document Databases,", Proc. of the Fourth International Conference on Database Systems for Advanced Applications(DASFAA'95), pp.457–466, Singapore, Apr. 1995.

[4] 森下淳也, 上島紳一, 大月一弘, 杉山武司, "階層構造グラフを用いた半構造化データの段階的構造化手法に関する検討," 情処研報, Vol.97, No.7, DBS96–111,pp.9–16, Jan.1997.

[5] 森下淳也, 上島紳一, 大月一弘, 杉山武司, "階層構造グラフにおける属性の取り扱い方に関する検討," 信学技報, Vol.96, No.469, DE96–79, pp.31–36, Jan.1997.

[6] 森下淳也, 上島紳一, 大月一弘, 大庭脩, "視点に依存した属性付け機構を持つ木簡研究支援データベースシステムの開発," 科研重点領域「人文科学とコンピュータ」1996年度研究成果報告書, Mar.1997.

[7] 杉山武司, 森下淳也, 大月一弘, 上島紳一, "グラフを用いたオブジェクトの多重ビューの実現に関する一考察," 第55回情報処理学会全国大会, 4AA–8, 福岡, Sep.1997.

[8] 上島紳一, 森下淳也, 大月一弘, 杉山武司, 田中克己, "木簡研究支援システムにおける視点に依存した属性付け機構に関する検討," 第55回情報処理学会全国大会, 2M–8, 福岡, Sep.1997.

[9] Laumann,O.,Bormann,C.,"ELK: The Extension Language Kit," USENIX Computing Systems, Vol.7–4, pp.419–449, Apr. 1994.

[10] Libes, D., "Exploring Expect: A Tcl-Based Toolkit for Automating Interactive Programs," O'Reilly & Associates, Inc., Jan. 1995.

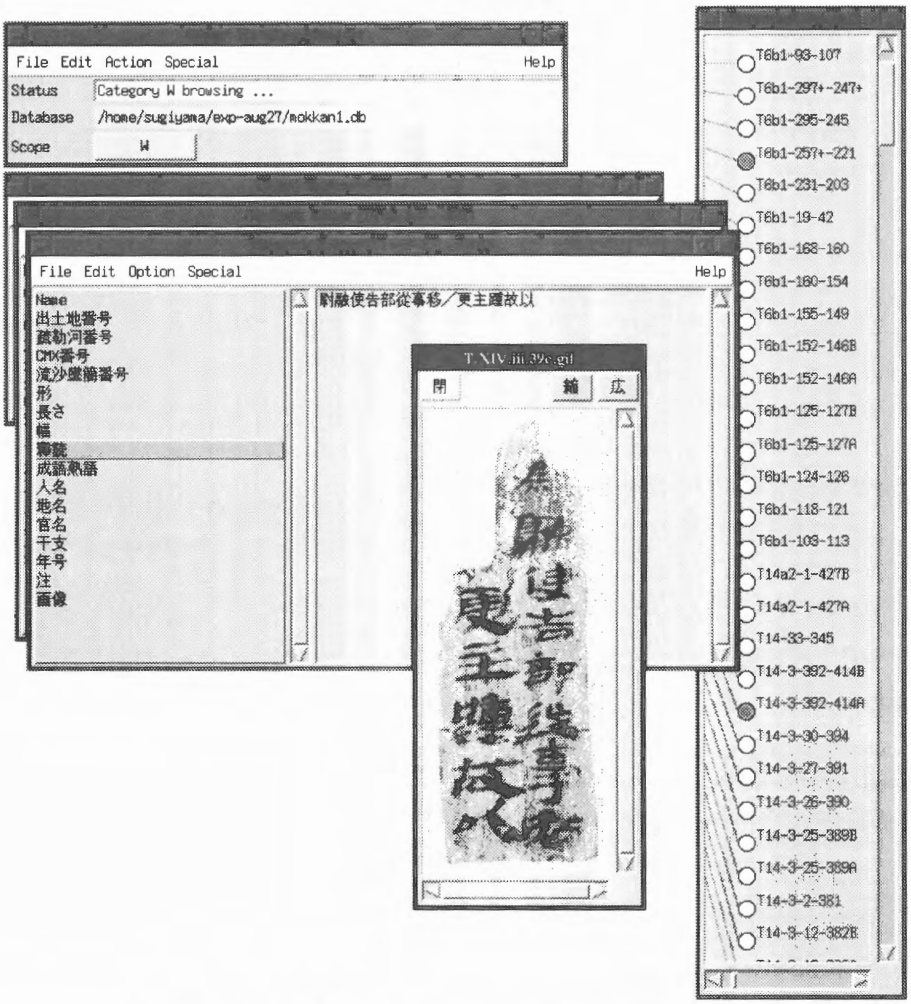


図9. 木簡プロトタイプシステムの動作例

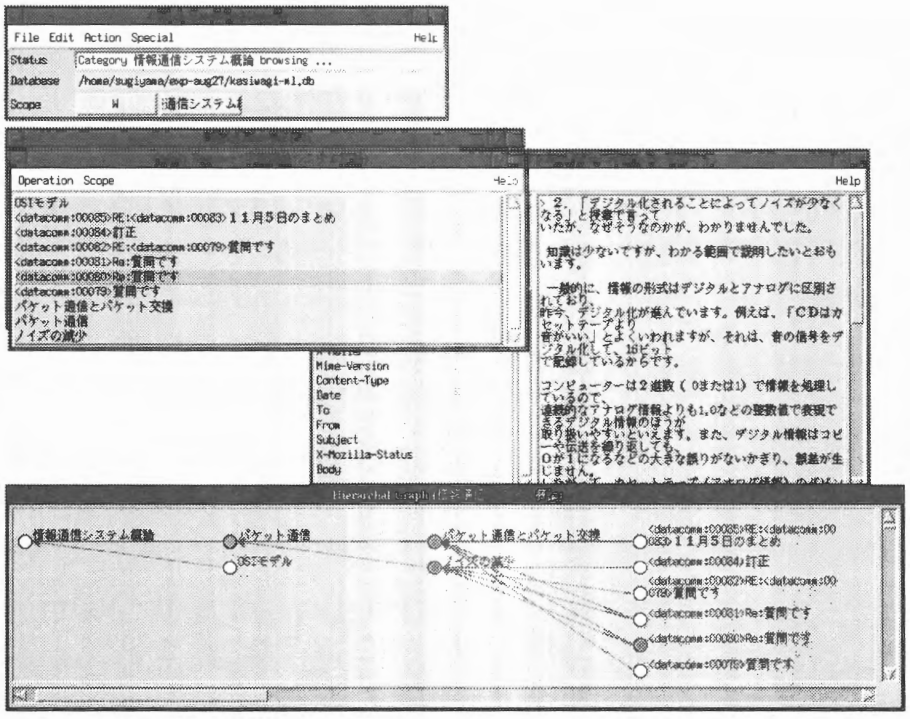


図10. メーリングリストの動作例