

オブジェクトモデリング環境 SOME の日本語化・多層表記化とそれに基づく発想支援システム KJS の開発

Japanization of SOME and KJ editor based on SOME

石堂 竹彦(15800010) 白川 淳平(15800041) 戸田 成彦(15800051)

TAKEHIKO Ishidou JUNPEI Shirakawa NARUHIKO Toda

原田研究室 指導教員 原田 実 教授 韓 東力 助手

1. はじめに

現在まで構造化オブジェクトモデリング環境 SOME[1]でクラス名、属性名、操作名、関連名等を入力する場合、半角英数字による入力しか許されていなかったため、ユーザは入力する名前の全てをローマ字で入力しなければならなかった。そのため、入力における自由度や図の見辛さや入力における手間やミスが問題視されていた。また、集約の視覚化表現にも改良の余地があった。そこで、以下のことを研究テーマとした。

- (1) SOME において日本語入力を可能にする。
- (2) 集約をより直感的に捉えられる多層表記 MLV のクラス図を追加する。
- (3) MLV を利用した KJ 図エディタ機能を SOME に追加する。

2. 日本語入力

全角ひらがな、カタカナ、漢字による入力を日本語入力と呼ぶことにする。従来の SOME でも日本語入力クラス名等を入力できるが、ファイルに保存するとそれは失われてしまう。また C 言語のコードを生成する際に全角文字を使用するとエラーになるので、事実上日本語入力は不可能であった。そこで本年度の研究では日本語入力により得られた文字列を半角英数字に変換し、それらをコード生成に渡すというプロセスを辿ることにより表示上で日本語利用を可能にした。日本語入力から半角英数字への変換にはフリーソフトウェアである KAKASI を使用した。すなわち、クラス名、属性名、操作名、関連名等は日本語入力により得られる日本語名と KAKASI により変換されて得られる半角英数字名の 2 つを保持することにした。尚、この日本語化によって図 1 に示されたようなわかりやすいクラス図から EOS による自動設計、LOLA による自動プログラミングを経て生成された C++ プログラムは VC++ でコンパイルされ正しく実行された。これで、SOME の従来の自動設計・自動プログラミング機能を保持しながら日本語化によるインタフェースの改善を行うことができた。

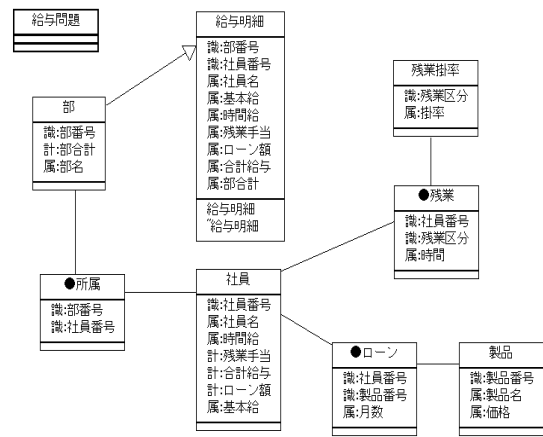


図 1. 日本語化された SOME の画面例

3. 新しいクラス図の表記法 MLV

これまでユーザはクラス間の集約関係を示す場合、クラス同士を線で結んで表現していた。しかしそれでは関係線が大量にあるときなどに交差などにより関係を正しく把握できなかつたりした。そこで再帰グラフ理論[2]にもあるように、クラス間の集約を図 2 に示す入れ子構造として直感的に理解できる表記法を SOME に導入した。この表記を多層表記 MLV (Multi Level View) と名づけて新しくクラス図の表記法とする。MLV 表記でユーザはクラスを集約先のクラス内に持っていきただけで集約関係を設定でき、関係を変更する時でも関係線の削除、再構築などの手間は省かれ直接その集約先にクラスを出し入れするだけで変更できる。クラス図を見たときも囲いの中にあるクラスを見ていくことで集約関係を理解でき、見間違ふこともない。このようにして直感によるクラス図の作成が可能になる。MLV 表記では継承は表現できない。そこで、実際に設計を行うときは、MLV クラス図でクラス宣言と関連と集約の関係の設定に重点を置いて設計し、その後自動変換機能を用いて UML 表記に変換し、ここで継承関係の設定を行う。

平成15年度卒業論文要旨

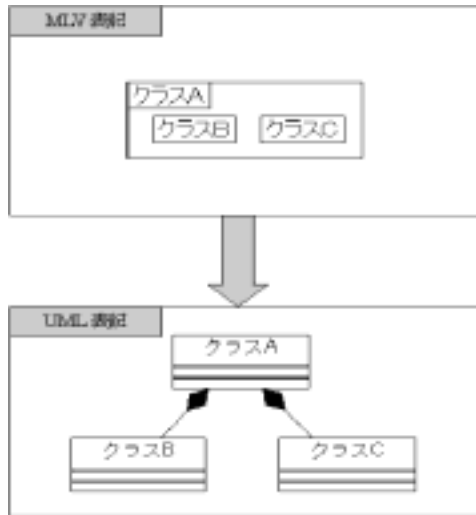


図2 . MLV 表記から UML 表記へ

エディタとしても有用なものであった。

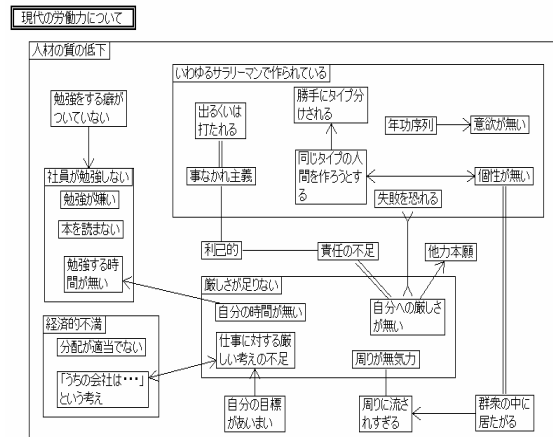


図3 . KJ 表記面例

4 . KJ 法エディタ KJS

最近の創造性開発(創造的問題解決)の技法にはブレイン・ストーミング、KJ法[3]、チェックリスト法、強制関連法、焦点法、アナロジー技法などがある。本研究では、この中で日本の文化人類学者川喜田二郎氏が考案したKJ法に焦点を当てた。KJ法の手順はブレイン・ストーミングなどで出されたアイデアや意見、または各種の調査の情報を1枚ずつ小さなカードに書き込み、関連のあるものを2,3枚ずつ集めてグループ化していき、図解化していく。それにより、テーマの解決に役立つヒントやひらめきを生み出すといったものである。意見などが書かれた小さなカードを名札といい、グループ化で名札を束ねておくものを表札という。

本研究では、SOMEにおいて新しく作成したMLV表記によるクラス図をベースにしてKJエディタを開発した。クラスはKJ表記における名札に、表札はクラスの集約先のクラスにという考えに基づいて、MLVクラス図エディタをKJ図エディタとして転用する。本来KJ法では表札は名札を束ねておくものであり、1つも名札を含まない表札は存在しない。しかしSOMEでのKJエディタ機能では名札と表札はどちらもクラスであるので名札を表札からすべて出したり、削除したときには元の名札の姿に戻る。この考えは、ある過程で用いられたグループ的概念を概念単体として一時的に保持するのに有効である効果を発想法にもたらした。

SOME上でKJ法を行うと図3のようになる。KJエディタの操作性を高めるために、ドラッグで範囲を指定し複数の名札を一度に入れられる機能、含んでいる名札により自動的に表札に囲み部分のリサイズを行う機能、Shiftキーを押しながら動かすことで含まれている名札の配置を自由に設定できる機能、ユーザの意図する通りに階層の構築/削除を行える機能などが必要になり、これらはMLV表記クラス図のエディタ機能として実現された。これらの機能はクラス図

5 . おわりに

本研究でSOMEを日本語化したことにより、見やすいクラス図の表現ができた。従来どおりコード生成も可能であった。また、MLV表記の導入により集約を視覚化できた。その結果、KJ法エディタを容易に開発できた。

6 . 参考文献

[1] 野口貴, 原田実: "ドメインモデルの再利用を支援するオブジェクトモデリング環境SOME/D", オブジェクト指向最前線 2003, 近代科学社, pp.191-192(2003).
 [2] Minoru Harada, Toshiyasu L. Kunii: "A Design Process Formalization", Proceedings of IEEE Computer Software & Application Conference, COMPSAC'79, pp.361-371, Chicago(1979.11)
 [3] "続・発想法", 川喜田二郎著, 中央公論新社, 2000