

5D-1

MOMOCLIプロジェクトにおけるプログラミング環境

垂水浩幸、阿草清滋、大野 豊

(京都大学工学部情報工学教室)

1. はじめに

MOMOCLIプロジェクトはオブジェクト指向型言語におけるソフトウェア部品である「クラス」を蓄積したクラスライブラリから必要なクラスを検索・再利用することが容易なプログラミング環境の構築を目的としている。その考え方については文献[1]に詳述した。本稿では、プロジェクトで開発する再利用支援環境の概要と再利用支援システムの構成を、[1]執筆後の設計変更を含めて述べる。

2. 再利用支援環境

本プロジェクトの特徴はクラスライブラリに蓄積されているクラスについて予備知識を持っていないユーザを対象としていることである。すなわち、ユーザはライブラリ中のクラス、メッセージの名前を知らないものと仮定している。そのようなユーザがプログラムを書こうとする場合、ライブラリについて調査して利用可能なクラスを見つけださねばならない。しかしライブラリが大きい場合は調査が困難であり、ライブラリ検索支援環境が必要となる。

本プロジェクトで構築する環境では、まずユーザが作成しようとしているプログラム中に登場する各オブジェクトについて、以下の項目を記述してもらう。

- (1) オブジェクトの名前。
- (2) 他のオブジェクトとの関係。
- (3) 形容詞によるオブジェクトの性質の表現。
- (4) オブジェクト上で実行されるオペレーションの名前。
- (5) 各オペレーションと他のオブジェクトとの関係。

(1)、(3)、(4)はそれぞれ自然言語の名詞、形容詞、動詞を用いてユーザが自由に記述してよい。ただし、再利用支援システムはこれらの単語を手掛かりとして検索を行うので、例えばオブジェクトの名前として「Data1」などの無意味な単語を用いるのは望ましくない。

一方、(2)と(5)はあらかじめ決められた数種

類の関係名を用いて記述してもらう。(2)で用いられる関係には、クラスの継承関係、他のオブジェクトを「知っている」関係、及び「持っている」関係がある。(5)で用いられる関係には、オペレーション実行中に他のオブジェクトに「作用する」関係、「依存する」関係、他のオブジェクトを「取り込む」関係などがある。これらの関係は「知人/構成変数モデル」[1]で定義されている。

再利用支援システムは、上に述べた要領によって記述されたものを読み込み、ユーザの記述したオブジェクトを実現するためのクラスの候補をクラスライブラリMOMOCLIから順位付きで数個選び出し、ユーザに提示する。

ユーザがオブジェクトについての最初に行った記述は、オブジェクトの仕様ではなく特徴の一部を表現しているに過ぎないので、システムが提示したクラスは必ずしもユーザが使えるものではない。したがって、ユーザは提示されたクラス群の説明を読み、再利用し得るクラスを選択する必要がある。多重継承を利用するために複数のクラスを選択することも可能である。

次に、システムはユーザが選択したクラスを用いるときに同時に用いるべきクラスを、MOMOCLI内で定義されているクラス間の関係を利用して探し出し、ユーザに提示する。例えば、ポップアップメニューを表現するクラスを利用するときには必ずマウスを表現するクラスを同時に使わなければならない。ここで用いられる関係も前述のモデルによって定義されているものである。

システムによって提示されたクラスがユーザの記述したオペレーションをすべて実現しているわけではないので、ユーザはクラスの継承を用いて不足している機能を補い、新しいクラスを作る。このとき、多重継承を用いることができる。

以上の作業によってプログラミングが行われるわけであるが、ユーザとシステムの会話に使われる情報(オブジェクトについてのユーザの記述や検索されたクラスの説明など)が複雑なものであるため、ユーザインターフェースとしては優れたものを準備することが前提となっている。

3. 再利用支援システムの構成

再利用支援システムはSUNワークステーション上で現在構築中である。図1にその構成を示す。

検索サブシステムがプロジェクトの核となる部分である。これはオブジェクトについてのユーザの記述を入力とし、候補クラス群を出力するサブシステムである。まずユーザの記述したオブジェクト名、オペレーション名、形容詞から辞書による単純な対応によって候補クラス群、候補オペレーション群を候補順位付きで選び出す。次にこれらの組み合わせの中からユーザの記述した関係をより多く満たし、ユーザの記述したオペレーションをより多く実現しているものを選択し、出力する。

辞書は部品作成者によって作られるものである。辞書作成は次のようにして行われる。まず、部品作成者は作成した部品（クラス）とそこで実現されている各オペレーションの性質を、自然言語の名詞、形容詞、動詞の単語を列挙することによって記述する。この記述は辞書作成ツール（図には示されていない）によって自然言語の名詞、形容詞、動詞をそれぞれ見出し語とする3つの辞書に変換される。自然言語の選択は自由であるが、本システムでは英語を用いている。現在、

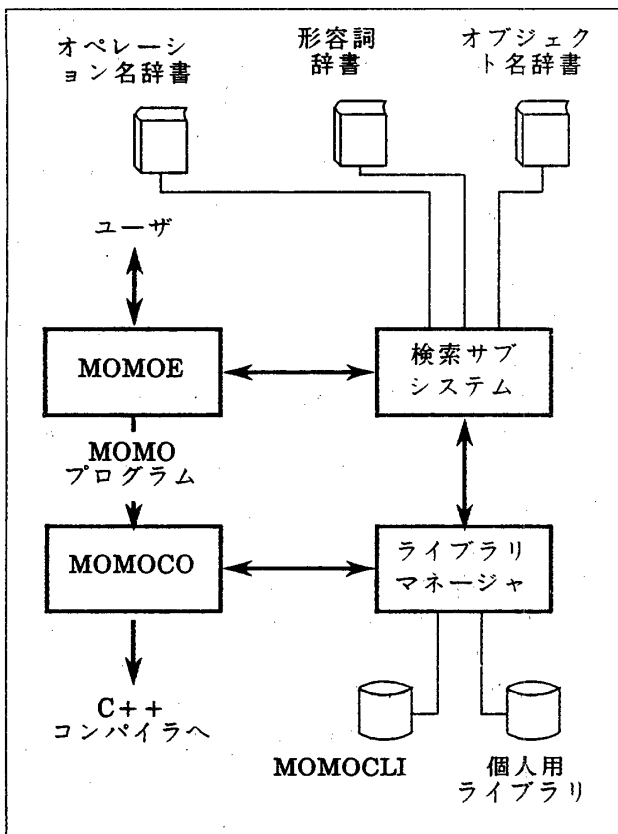


図1 MOMOCLIプロジェクトで作成するシステム

辞書はUNIXの簡易データベース構築パッケージDBMによって実現されている。

ライブラリマネージャはMOMOCLIと個人用ライブラリを管理・検索する。前者は再利用を意識した公的で質の高いライブラリであるのに対し、後者は個人で開発したクラスを蓄積するだけのものである。本マネージャは検索サブシステムやMOMOCOからの種々の問い合わせに解答する他、クラスの登録・削除などの作業を行うものである。MOMOCLIはネットワークモデルによるデータベースとして構成されている。MOMOCLIには実際に部品として利用されるクラスその他、検索に利用される抽象クラスや、オペレーションとクラスの関係のみを示すオペレーションフレームというデータが蓄積されている。

MOMOCOは「知人/構成変数モデル」を実現する言語MOMOの処理系である。UNIX上で簡単に実現するために、MOMOのプログラムはいったんC++に変換され、市販のC++トランスレータによって実行可能コードまで落とされる。

MOMOEは本システムとユーザとの窓口である。現在はプロトタイプのためシェルスクリプトで既存のエディタや上述のサブシステム群を呼び出しているに過ぎないが、将来はプログラムやクラスの構造を階層的に表示したり、オブジェクトについてのユーザの記述を容易にするようなウィンドウシステムとして実現する。

文献[1]以後の主な変更点は形容詞による記述の導入である。オブジェクト指向言語中に形容詞に直接対応する概念はない。しかし例えば「赤い」と形容されるオブジェクトは「赤いもの」を表現するクラスがあればそのサブクラスから生成できるはずであるし、もしなければ、「何色か」というメッセージに反応するクラスを候補にあげることができる。形容詞辞書は形容詞をクラス名またはメッセージ名に対応させるものとなっている。

4. おわりに

紙面の都合で概略説明にとどまった。候補クラス選択の手法など、本システムの詳細については、別の機会に発表する予定である。

参考文献

- [1] 垂水、岡村、阿草、大野：クラス再利用支援のためのオブジェクトモデル、コンピュータソフトウェア、Vol. 3、No. 3、61-70、1986