

A-031

オブジェクト指向プログラミングにおけるオブジェクトの自動可視化 — Nigari への実装を例として

Automatic visualization of objects in an object-oriented programming — an enhancement to Nigari

鈴木 健司[†] 小田嶋 祐介[†] 長 慎也[†] 甲斐 宗徳[‡] 筧 捷彦[†]

Kenji Suzuki Yuusuke Odashima Shinya Cho Munenori Kai Katsuhiko Kakehi

1. はじめに

高度情報通信社会の到来に対応した教育の環境設備が叫ばれており、初等中学教育の科目として「情報」の採用も取り入れられたところである。しかし情報教育の重要性が叫ばれている中、それに適した環境、システムの設備の遅れが目立つ。そこで我々は、初学者にプログラミング学習の導入部において用いるのに適したオブジェクト指向プログラミング言語 Nigari とその環境である NigariSystem[1] を開発した。

初学者がオブジェクト指向プログラミングを学習する場合、オブジェクト、オブジェクトの参照関係、データ構造といったものが視覚的に表示されると、プログラムの動きを理解する上で大きな手助けになる。そこで本研究では、実行時にシステムがそれらを自動的に可視化し、画面上に表示する手法を提案する。この手法を我々が開発した言語 Nigari とその環境である NigariSystem に実装した。(以下、去年までの NigariSystem を NigariSystem2003、今年拡張した NigariSystem を NigariSystem2004 とする)

2. Nigari の概要

オブジェクト指向プログラミング言語 Nigari とそのプログラミング環境である NigariSystem2003 は次のような特徴をもっている。

2.1 簡素な言語仕様

Nigari は、クラスやメソッドの宣言を書かずともプログラムが動作するようになっている。よって、初学者に「おまじない」として教えておかななくてはならない項目がほとんどない。また Nigari の変数は、宣言することなしに使用が可能であり、変数にどんな型でも代入することが可能である。

2.2 Java に近い言語仕様

学習が進行すると、実用的な言語の習得も必要になる。Nigari の言語仕様は、式や制御文の構造などをほとんど Java と同じにして、実用言語として学ぶ Java への移行の際に混乱をきたさないようにしている。

2.3 オブジェクトの可視化

NigariSystem2003 は、プログラムが正常に動作しているか確認するために、作成したオブジェクトを自動的にグラフィックが出力するという方法を採用している。

これにより、プログラムの動作をイメージしやすいものとなる。

3. NigariSystem2004

NigariSystem2003 の可視化機能はオブジェクトを可視化するのみで、オブジェクト指向プログラミングを学習する際教えるべき、配列や、オブジェクト間の参照といった、基本的な部分を網羅していないといえる。この問題を解決すべく、可視化機能を拡張した NigariSystem2004 を実装した。

3.1 可視化機能の設計

NigariSystem2003 のオブジェクトを可視化する仕組みでは、プログラム上で生成されたオブジェクトをそのまま利用することで画面上にオブジェクトを表示していた。

今回 NigariSystem2004 では、2つの段階に分け可視機能を実装する。

- Visible 変換機能

プログラム上で生成されたオブジェクトから描写に必要な要素を取り出して、それを元に Visible と呼ばれる可視化するためのオブジェクトを生成する

- 描写機能

Visible のみを用いて画面上に描写をする

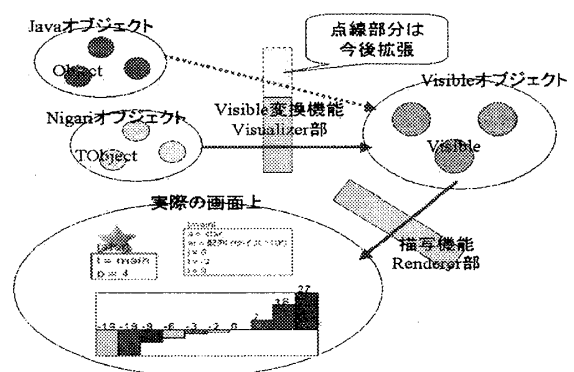


図 1: 可視化機能の全体像

このように実装することで、他のオブジェクト指向プログラミング言語で書かれたプログラムでも同様な描写が可能になる。図 1 で示すように、言語ごとに Visible

[†]早稲田大学 理工学研究科 - Graduate School of Science and Engineering, Waseda University

[‡]成蹊大学 経営・情報工学科 - Department of Industrial Engineering and Information Sciences, Seikei University

変換機能さえ拡張していけば、実際に描写する描写機能の部分を共有化することが出来る。

3.2 可視化機能追加項目

すべてのオブジェクトのすべての値を画面上に表示する

NigariSystem2003では、オブジェクトが画面上に表示されるのみで、オブジェクトの値の変化は画面上に表示されなかった。今回、オブジェクトがもつすべての値をオブジェクトと共に画面上に表示する。利用者は値の変更をプログラムの実行中に常に見てとれるようになる。

オブジェクト間の参照関係を矢印を使い表示する

オブジェクトが参照する他のオブジェクトを矢印をつないで表示する。プログラムの実行中、オブジェクトの位置が変更されたときにも、この矢印での参照は自動的に追跡する。

配列の要素を文字またはグラフで表示する

オブジェクトと同様に、配列を画面上に表示できるようにする。今回、配列を表示する上で2つの表示形式を採った。配列の要素を文字ベースで表示する形式とグラフを用いて表示する形式である。こうすることで、プログラムでの配列の用途に合わせて選択が可能である。

コンソール出力を画面上に表示する

NigariSystem2003では、コンソール出力をオブジェクトが表示される画面とは別の画面で表示をしていた。今回、コンソール出力をオブジェクトが表示される画面に重ねて表示するという方法を採用した。コンソール出力は、画面の左側面に表示することにした。

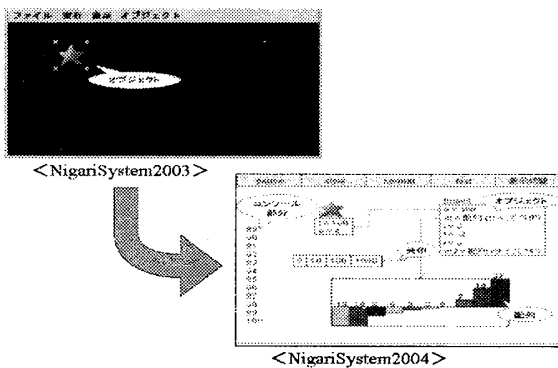


図 2: NigariSystem2003 と NigariSystem2004 の描写機能比較

4. アンケート検証

NigariSystem2004 を、早稲田大学工学部 4 年情報学科 22 名を評価対象にし、アンケートによる検証を行った。まず NigariSystem2004 を用いてプログラムの演習問題を解かせ、その後アンケートに答えてもらうという方法を採用した。問題には、拡張した可視化機能の仕組みが有効に働くよう、配列、オブジェクト間の参照、コンソール部分への出力といったものを含んだものを採用した。

アンケートの集計結果のうち、可視化機能の評価にあたる部分を図 3 に示す。

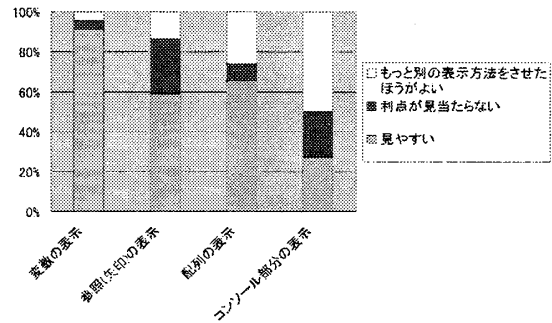


図 3: アンケート結果

配列、オブジェクトの参照関係、変数等が表示できるようになったことで、プログラムの動きをより理解しやすくなり、高評価を得られた。しかし、コンソール出力の表示に関しては、別の表示方法を求める意見が多く、表示方法を検討する必要がある。

5. まとめと今後の課題

今回、初学者がオブジェクト指向プログラミングを学習する際の手助けになることを目的として、プログラムの実行時にシステムがオブジェクト等を自動的に可視化し、画面上に表示する手法を提案した。プログラムを視覚的に表示する NigariSystem2004 の可視化機能について、プログラムがある程度かける者を対象にアンケート検証を行った結果、プログラムが視覚的に表示されることの有効性が示された。

現在 NigariSystem2004 を用いて、大学 1 年生向けのプログラミングの授業が行われている。その成果を見て、初学者に対する効果を計る予定である。更に、Java 言語で書かれたプログラムに対しても、今回の可視化の仕組みをそのまま利用できるようにシステムの機能拡張を進めている。

参考文献

- [1] 長慎也：プログラミング環境 Nigari - 初学者が Java を習うまでの案内役 第 47 回情報処理学会プログラミング研究会 (2004)
- [2] 長慎也, 川合晶, 日野孝昭, 前島真一：NigariSystem <http://taurus.kake.info.waseda.ac.jp/Nigari/>
- [3] 長慎也, 川合晶, 日野孝昭, 前島真一：NigariSystem 仕様書 <http://taurus.kake.info.waseda.ac.jp/nspec/>
- [4] Gray, K.E. and Flatt, M.: ProfessorJ: a gradual introduction to Java through language levels, Companion of the 18th annual ACM SIGPLAN conference of Object-oriented programming, systems, languages, and applications, ACM Press, pp. 170-177(2003)