

# クラス構成演算によるクラス構造自動構成ツール Class Builder の試作

橋本 亮† 小林 悠介† 大木 幹雄†

日本工業大学 情報工学科†

## 1. はじめに

現代の情報システム開発において、オブジェクト指向設計の中核を成すクラス構造の分析設計作業は、後続作業の進行速度や完成するシステムの品質および拡張性を左右する重要な工程である。しかし、クラス構造の設計手法は明確な定式化が行われておらず、現在でも設計者が持つ経験や勘などといった不確定な要素により設計の詳細が決定されており、生成されるクラス図の品質が保証されているとは言い難い。

この問題を受け、本稿では経験や勘などといった設計者固有の能力を問わず、誰にでも一定の品質のクラス図を生成出来ることを目標に、クラス構造の構成演算による導出論理 [1] を用いたクラス図の自動構成ツール Class Builder を試作し、その検証を行った。

## 2. Class Builder の概要

Class Builder は、クラス構造の自動構成に必要な各種データの入力機能、変換機能、クラス図描画・編集機能の 3 つの機能を持つモジュールから構成される。ツール内で扱う各種のデータは、Microsoft Access データベース上に保存し必要に応じて読み書きを行う。

本ツールにおけるクラス構造の自動構成は、以下の機能を順次実行していくことによって行われる。

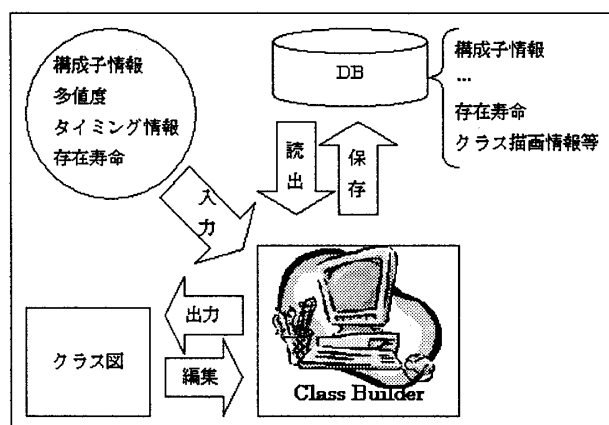


図 1 Class Builder の機能イメージ

Class structural automatic operation composition tool  
Class Builder through class composition operation

† Ryou Hashimoto, Yusuke Kobayashi, Mikio Ohki

‡ Nippon Institute of Technology

## 2.1 データ入力機能

クラス構造の自動生成に必要なデータの入力、および編集を行う機能である。クラス構造の生成に必要なデータは以下の通りである。

### (1) 構成子情報

本稿ではクラスに所属する属性およびメソッドのことを、クラスを構成する要素として構成子と呼ぶ。

構成子情報とは構成子の名称や、その構成子が属性とメソッドのどちらか等、個々の構成子が持つ基本的なデータを指す。

### (2) 多値度

その構成子のインスタンスが生成された時に、同時に決定されるインスタンスの個数を示したデータである。

### (3) タイミング情報

構成子の生成・削除が行われるタイミング、および入力されたタイミングの前後関係を整理した時系列のデータである。

### (4) 存在寿命 (Lifetime)

その構成子が発生してから利用を終えて不要な情報となり、削除が行われるまでの期間を表すデータである。構成子情報とタイミング情報の組み合わせで表現される。一つの構成子が持つ存在寿命は一つとは限らず、複数の寿命を持つ場合がある。

クラス構造の生成を行う構成演算は、これらの情報に着目してクラスやクラス間のリレーションを生成し、生成されたクラスに構成子を配置していくことによって、最終的なクラス構造を生成する。

## 2.2 変換機能

入力されたデータをツールが内蔵する 13 種類の構成演算式に従って変換し、クラス構造を得るための機能である。ツール上に入力されたデータは、それぞれ構成演算の適用条件を満たすと次々に変換処理が行われていく。その結果得たクラス構造は、クラス描画情報として保存される。

## 2.3 クラス図表示・編集機能

構成演算によって得たクラス描画情報を用いて、クラス図の描画を行い画面上に表示するための機能である。表示中のクラス図に対する編集機能も有する。

## 3. 評価実験

### 3.1 実験概要

Class Builder の評価実験として、GOF デザインパターン [2] の中から構造に関する数パターンを対象に、本ツールによって出力されたクラス構造と合致するか

の検証を行った。デザインパターンの出力を評価対象とした理由は、デザインパターンがクラス構造設計において最も標準的で正解に近い設計と考えたからである。

本稿では構造に関するデザインパターンと同等のクラス構造が自動生成されるかの実験として、Adapterパターンと Bridgeパターンを取り上げ、実験に用いた入力データおよび出力例を示す。

### 3.2 Adapter パターン生成実験

Adapterパターン生成用の入力データには、入力構成子として、Request()と SpecifiedRequest()の2種のメソッドが存在する。Request()と SpecifiedRequest()は、それぞれ一部が重なり合う存在寿命 A および存在寿命 B を持ち、更に存在寿命の論理積  $A \cap B$  に内包される存在寿命 C を持つ形で定義される。以上の入力データを表 1 において示す。

以上のデータをツール上に入力し、自動生成実験を行った結果、図 2 に示されるクラス図が生成された。これは Adapter パターンのクラス構造に合致する。

表 1 Adapter パターン生成用入力データ

構成子情報		タイミング情報			
構成子名	多値度	タイミング1	タイミング2	タイミング3	タイミング4
Request()	1				
SpecifiedRequest()	1				

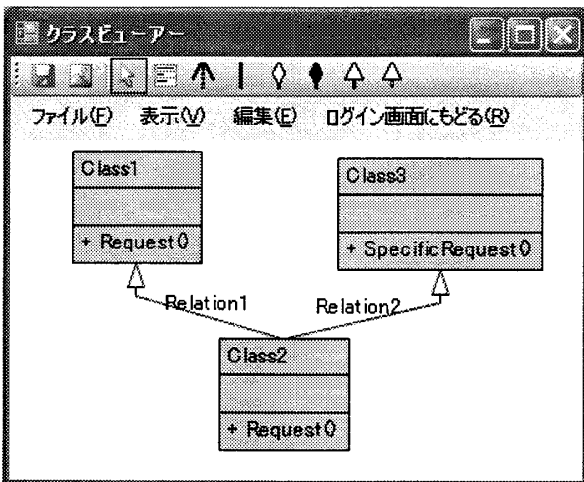


図 2 Adapter パターンに合致するクラス構造出力結果

### 3.3 Bridge パターン生成実験

Bridgeパターンの生成用の入力データには、入力構成子として多値度 1 で実装されるメソッド Operation() が 1 つ、同じ識別名を持つメソッドであり多値度が複数 (表 2 内では\*と表記) で実装される Operation() が 2 つ、合計 3 つの構成子が存在する。複数の多値度を持つ 2 つの Operation() は、それぞれ存在寿命 A および存在寿命 B を持ち、存在寿命 A と存在寿命 B は一部が重なり合っている。多値度が単数の Operation() は存在寿命  $A \cup B = C$  の関係にある存在寿命 C を持つ形で定義される。以上の入力データを表 2 において示す。

以上のデータをツール上に入力し、自動生成実験を行った結果、図 3 に示されるクラス図が生成された。これは Bridge パターンのクラス構造に合致する。

表 2 Bridge パターン生成用入力データ

構成子情報		タイミング情報			
構成子名	多値度	タイミング1	タイミング2	タイミング3	タイミング4
Operation()	1				
Operation()	*				
Operation()	*				

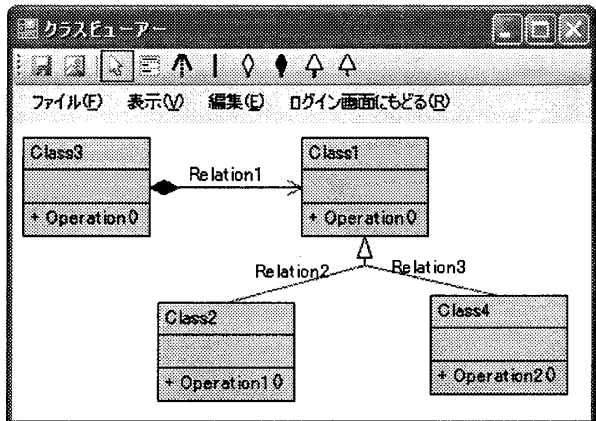


図 3 Bridge パターンに合致するクラス構造出力結果

## 4. 考察と今後の課題

本稿ではクラス構成演算によるクラス構造自動構成ツール Class Builder の試作および検証を行い、いくつかの構造に関するデザインパターンの出力に成功した。デザインパターンの出力に成功したということは、本ツールにおいてそのパターンに対応した典型的なクラス構造が出力可能ということであり、本稿が目標とした一定品質のクラス図が生成できることを示したものである。

構造に関するデザインパターンは本稿で取り上げた Adapterパターンと Bridgeパターンの他に、Compositeパターン等がある。Compositeパターン等の再帰的なクラス構造の生成には、クラスを構成子として扱う必要があるため、存在寿命の指定が困難である。そのため、再帰構造の生成に対応させるための変換機能の強化、および複雑な存在寿命の入力を容易にするためのデータ入力画面のユーザインタフェース強化等が今後の課題となる。

## 参考文献

- [1]大木 幹雄, ライフタイム分析に基づくクラス構造抽出の定式化と構造に関するデザインパターンの抽出実験, 情報処理学会 論文 45 巻 6 号 PP.1554-1568, (2004)
- [2]Erich Gamma, Ralph Johnson, Richard Helm, John Vlissides 共著/本位出 新一, 吉田 一樹訳, オブジェクト指向における再利用のためのデザインパターン 改訂版, ソフトバンクパブリッシング, (1999)