

# 総称的動詞オブジェクトによる 画像システムの仕様記述

7K-6

川野辺 甲 橋本 雄治 村尾 洋 榎本 肇

芝浦工業大学

## 1. はじめに

ソフトウェア記述プロセス中において、仕様記述と手続き記述の間に多くの重複があったり、仕様と手続き間の関係が希薄であったりする。これを克服するためには仕様定義部と手続部分との間の相互関係を明確にし、ソフトウェア記述プロセスに適合するように、ソフトウェアの仕様から逐次的な実行手順のみでなく並行手続きをも意識した形で導出をおこなう仕様の記述方式を明確なものにし、ソフトウェア記述プロセス全体の流れの枠組みを定義する方法論を展開する必要がある。本論文では、画像システム記述言語(WELL-PPP<sup>[1]</sup>)上でインプリメントされた描画システム中のオブジェクトネットワークを取り上げ、そこの仕様をTELL-SCP<sup>[2]</sup>における仕様の記述方法にならって記述する。

## 2. TELL-SCPにおける仕様の表現

TELL-SCPでは、オブジェクトを実現するために考えられた表現として仕様とミッション(制御情報)がある。仕様は、属性構造等の制約条件や関数などの定義を行うものであり、ミッションは、制御・動作に関する制約条件を定義するものである。そして、これら二つを合わせてドキュメントと呼ぶ。

## 3. 動詞オブジェクトと名詞オブジェクト

オブジェクト間の関係を考える時、オブジェクトを動詞オブジェクトと名詞オブジェクトに分けて考えた。名詞オブジェクトは、普通名詞に対応する集合オブジェクトと、固有名詞に対応する固有オブジェクトから成る。動詞オブジェクトは、ある演算を行う関数であると考へ 1)演算動詞, 2)関係付加動詞, 3)制約演算動詞<sup>[3]</sup>の三つに分類できる。

また、オブジェクトの記述単位は項目と関数、静的と動的という性質にわけられ、それらは、直交性を持っている。名詞オブジェクトは項目にあたり、動詞オブジェクトが関数に当たるので、静的名詞オブジェクトをクラス、動的名詞オブジェクトをダイナミッククラス、静的動詞オブジェクトをファンクション、動的動詞オブジェクトをアクションという。

## 4. 総称的動詞オブジェクトと関数

一般動詞で表されるものを総称的動詞オブジェクトと言ひ、これは名詞オブジェクト間をつなぐ関数と見た場合、総称的関数という。総称的関数にある制約条件を加えると、実際に実行可能な関数である具体的関数ができる。

例えば draw up (何かを描く) という総称的関数に a point という名詞オブジェクトを制約条件として付加することによって、draw up a point という具体的関数ができ、点を描くという具体的な実行可能な動作が表現される。

このような関数の仕様を考え、定義するために、この関数の動作条件として、Precondition, Incondition, Postcondition という三つの条件を導入する。Precondition は、動作の前提条件であり、その関数の入力条件を示すものであって、この条件が揃わないと関数が実行されない。Incondition は、関数動作中条件であり、

関数を実行している時、その関数により作成される各データがその条件内に入っているかどうか、つまり、そのデータ値が妥当なものかどうかをチェックするものである。Postcondition は、動作の後堤条件であり、ある関数実行により作成されたデータがある条件をみたしているかのチェックを行うものであり、得られた結果が妥当なものかどうかという妥当性のチェックを行うものである。

この三つの条件を明確なものにすることによって、関数が定義できる。

## 5. WELL-PPPのオブジェクトネットワークを用いた仕様記述

### 5.1 オブジェクトネットワーク

WELL-PPP上でインプリメントされる描画システムは、オブジェクトネットワークを用いて表現されている。オブジェクトネットワークは、クラスとしての名詞オブジェクトをノードとし、その間を関数としての総称的動詞オブジェクトをブランチとしてつなぎ、名詞オブジェクトが階層的に定義されている(図1)。総称的動詞オブジェクトが、総称的関数となり、クラスが制約条件となって具体的関数が導出される。また、クラスの属性構造をテンプレートという枠組みで形式化した。

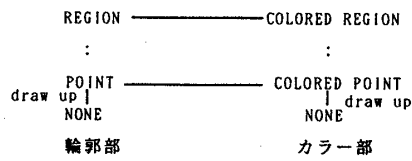


図1 オブジェクトネットワーク

### 5.2 画像描画システムの仕様の位置づけと記述形式

画像描画システムの仕様は、具体的な仕事の内容を表しており、総称的関数を用いて記述される。これは、具体的関数の仕様は記述量が多いため、総称的関数の仕様と、その仕様から具体的関数を生成する生成規則を仕様で記述する事で記述量を少なくすることができることによる。

これとは別に、クライアントとユーザーのリクエスト・レスポンドというインタラクションを記述したインタラクティブ仕様<sup>[4]</sup>がある。これらの仕様を統合することによって、協調処理システムの仕様ができる。

### 5.3 総称的関数の仕様

描画システムの輪郭作成の処理は、オブジェクトネットワークのノードに沿って順に行われるので、処理の順序が明確であるという事から、仕様のみでドキュメントが生成される。

名詞オブジェクト間をつなぐ総称的関数の仕様を示す(図2)。

Preconditionの1)では、静的なものとしてのクラス名を、ダイナミッククラスとして認識する事を表現している。例えば、点の座標といった動的な値をクラスに持たせる事を考えると、クラスは動的な値は持てない理由からクラスからダイナミッククラスへの状態変化が必要となる。2)では現クラスの関数を起動するには、前クラスのデータが満足されている事が必要であることが記述されている。

Inconditionの1)では、まず、データを入れるテンプレートが準備されたかどうかをチェックし、3)では、そのデータの情報がカラー部に送られたかどうかをチェックする。

Postcondition の1)は、テンプレートのセルに代入された値が妥当なものかどうかをチェックし、2)は、次の状態に移るのか、それとも引き続き現状態で関数を実行するのかを識別する条件である。

Function name: draw up

Precondition)

- 1) When the current class name is hit on the operation window, the current class name is recognized as a dynamic class.
- 2) Every value in every cell of template corresponding to the previous state is already satisfied.

Incondition)

- 1) When the mouse button is pushed on the data window, the template corresponding to the current state is prepared.
- 2) Every value of every cell in template corresponding to the current state is satisfied.
- 3) The information of frame and every value in the template corresponding to the current state are transmitted to the color section.

Postcondition)

- 1) Every value of every cell in template corresponding to the current state is already satisfied.
- 2) case 1
  - 1) When the next class name is hit on the operation window, the next class name is recognized as a dynamic class.
  - 2) Every value of every cell in a set of templates corresponding to the current state is already satisfied.
- case 2

The current state changes the state of 'the current class name is hit on the operation window'.

図2 draw up の仕様

### 5.4 具体的関数の仕様

具体的関数の仕様は総称的関数の仕様に対して詳細化規則である制約条件別の変換規則を適用することで構成される。draw up a point という具体的関数の仕様を示す。(図3)

Actual function name: draw up a point

Precondition)

- 1) When 'Point' is hit on the operation window, Point is recognized as a dynamic class.

Incondition)

- 1) When the mouse button on the data window is pushed, Tp is prepared.
- 2) Every value of I.D, Name, X, Y in Tp is satisfied.
- 3) The information of frame and every value of I.D, Name, X, Y in Tp are transmitted to the color section.

Postcondition)

- 1) Every value of I.D Name, X, Y in Tp is already satisfied.
- 2) case 1
  - 1) When 'Point sequence' is hit on the operation window, the point sequence is recognized as a dynamic class.
- 2) Every value of I.D, Name, X, Y in a set of Tp is already satisfied.
- case 2
- 1) The current state changes the state of 'Point is hit on the operation window'.

図3 具体的関数の仕様

### 5.5 名詞オブジェクトの属性定義

5.4節の具体的関数の仕様で、制約として名詞オブジェクトの属性構造、テンプレートの型を定義する必要がある。(図4)

Class name: 'Point'

- 1) Every principal point is a point.
- 2) Every point has the template, 'Tp'.
- 3) Tp has a sequence of cells.

Cell name in Tp: 'I.D', 'name', 'X', 'Y', 'Chi', 'Lumi', 'Lx', 'Ly', 'PPP', 'PCP', 'NPP', 'NCP'.

- 1) 'I.D' has a value of a index number.
- 2) 'Name' have twenty words.
- 3) 'X' has a value of a x coordinate.
- 4) 'Y' has a value of a y coordinate.
- 5) 'Chi' has a i element in the chrominamce diagram.
- 6) 'Chq' has a q element in the chrominance diagram.
- 7) 'Lumi' has a value of the luminance in the luminance bar.
- 8) 'Lx' has a x element of the luminance vector.
- 9) 'Ly' has a y element of the luminance vector.
- 10) 'PPP' has a value of the prior principal point.
- 11) 'PCP' has a value of the prior connecting point.
- 12) 'NPP' has a value of the next principal point.
- 13) 'NCP' has a value of the next connecting point.
- 14) Every value of 'I.D', 'X', 'Y', 'Lumi', 'PPP', 'PCP', 'NPP', 'NCP' is integer.
- 15) The value of 'Name' is character.
- 16) Every value of 'Chi', 'Chq', 'Lx', 'Ly' is vector.

図4 名詞オブジェクトの属性定義

### 6. 総称的関数から具体的関数への変換規則

詳細化規則としての総称的関数から具体的関数を生成するための変換規則を示す。ある総称的な表現がある具体的な表現に変化する事を制約条件別に記述する(図5)。

If 'draw up' has 'a point' as constraint,

- 1) 'the current class name' changes 'Point'.
- 2) 2) of the precondition is none.
- 3) 'the template corresponding to the current state' changes 'Tp'.
- 4) 'every cell' changes 'I.D, Name, X, Y'.
- 5) 'the next class name' changes 'Point Sequence'.

If 'draw up' has 'a point sequence' as constraint

図5 制約条件別の変換規則

### 7. まとめ

本論文では、具体的関数が総称的関数に制約条件を付加して作られる関係から、具体的な仕様が総称的関数の仕様と制約条件別の変換規則による生成やその記述形式を示した。また、この仕様は、サーバーとクライアントに分けてインタラクションの関係を示したインタラクティブな仕様の中で、描画システムが具体的に処理を表しており、これらの仕様が形式的に記述することができたので、それぞれの仕様間の統合が成されれば協調処理システムとしての仕様ができる。そして、この仕様にある規則を適用すれば、C言語などの実際に実行可能な言語に変換できる。

### 文献

- [1] 丹羽, 守屋, 関, 村尾, 榎本: " 並行実行型画像システム記述言語 (Concurrent WELL-PPP) の実現, 情報処理学会 第46回全国大会, 1993, 3
- [2] 森谷, 村尾, 榎本: " 内包的テンプレートによるソフトウェア記述プロセスの実現", 情報処理学会第46回全国大会, 1993, 3
- [3] 榎本, 鴨志田: " 分野記述言語の構造", 情報処理学会 第44回全国大会, 1992, 3
- [4] 榎本, 村尾, 榎本: " インタラクションの仕様記述とそれによる画像ソフトウェアへの応用", 情報処理学会第46回全国大会, 1993, 3
- [5] 榎本 監編: " ソフトウェア工学ハンドブック", オーム社, 1986, 4