

6F-3

分野記述言語の構造

榎本 肇 鴨志田 稔

芝浦工業大学

1. はじめに

ある特定応用分野のために、システム開発をするに当たって、システムの効率をあげるために、ハードウェアではASIC(Application Specific IC)を設計し、システムの核部分、あるいはインタフェース部分に採用し、コスト・パフォーマンスをあげている。

これに対応して、ソフトウェアでも、特定応用分野のための言語を設計製作し、それを総合的に利用することが考えられる。その場合、特定応用が多様、多岐である時、関連分野に適合するような特別なソフトウェア体系を構築することによって、

- (1)ソフトウェア生産の効率化
- (2)カスタマイズ容易なソフトウェア・パッケージ
- (3)ユーザ・フレンドリシステムの構築

を達成しようと考えられる。

以上の観点から、総合的に分野記述型言語のあるべき姿をさぐり、画像処理・描画システムを例にとりて考察する。

2. 表現世界とモデル

ある分野についてのソフトウェア設計を行うにあたって、その分野に属する種々の応用の核部分の構造を考える必要がある。それは応用対象が属する特定の実世界について対象構造をオブジェクト・ネットワークとして表現し、感性に訴えうる仮想対象として、実際と比較評価しうるように、中核構造として具体化されなければならないことによる。この過程では、図1のように、

(1)実世界→感性対象世界:

実世界から具体的興味に動かされて、感性的に対象に着目し、実世界中の対象集合を感性対象世界に移入する。これによって応用対象の属する分野が特定される。

(2)感性対象世界→オブジェクト論理表現:

感性対象集合の特徴を、対象世界中に存在する環境の変化に対し、独立性のある属性構造としてとらえ、理解処理を行うものとする。その結果特徴構造はオブジェクト・ネットワークとして論理表現化され、形式的論理の世界に写像することが可能となる。

この段階で、分野に属する対象についての仕様の記述が終わり、次にその実行処理を行う。

(3)オブジェクト論理表現→実行処理表現:

仕様から平行処理手続など実行処理表現の誘導を行う段階であり、実行処理に関するインタフェース構造および、目的としての実行処理を行うため合目的構造モデルをつくることによってオブジェクト・ネットワーク表現が実行処理表現に変換される。

合目的構造モデルは、TellシステムでのMissionのように下層レベル対象についてのオブジェクト・ネットワークとなるが、そのため特化した用語が必要となる。この例でも見られるように特化した分野記述のための語を必要とし、この語群が上層のネットワークと結合する規則によって制御される。

これによって、オブジェクト・ネットワークとしての階層化構造が生成される。

(4)実行処理表現→個別対象仮想表現:

実行処理表現は、手続的表現であり、実行インタフェースがこれに作用して、結果としての対象が具体的に得られるが、これはあくまでも仮想的なものである。ここに至るために、具体的オペレーションとしては、オブジェクト・ネットワーク表現から実行処理表現に変換する部分に、オペレーションとして関数、または動作表現として記述されており、そのためのインタフェース表現も当然含まれている。

実行インタフェースは、それを具体的実行に移すためのインタフェースとしての操作であるので、オブジェクト・ネットワークのみが見え、その具体化操作が、実行インタフェース部に伝達される。

(5)個別対象仮想表現→感性的仮想対象世界:

個別対象が仮想的結果として得られたならば、それらが集積され、レビューされて一つの世界が形成される。ここでは感性的なレビューが中心であり、部分的に理論的検証が行われる。レビューは感性的に妥当性の明確なサンプルもしくは理論にもとずかなければならない。

この段階以前において妥当性は、可能な限りオブジェクト・ネットワーク中の関数/動作表現の制約条件として、関門が設けられている。

(6)感性的仮想対象世界→感性対象世界:

両者の比較は、実行処理の結果として得られた仮想対象世界を、実際の感性的対象世界と、ユーザあるいはレビュー者が比較評価し、オブジェクト・ネットワーク表現された論理表現の適否を判定しなければならない。

(7)モデルと対象世界

ある意思によって、ある分野の対象構造を、オブジェクト・ネットワークとして論理表現するためには、特徴構造が必要で、これは対象についての部分的な論理モデルとなっている。同様にこれを仕様としてとらえ、実行処理のために平行手続などの変換操作を実行するためには、実行方針に適合するように、合目的性モデルを構成する必要がある。

合目的モデルは、目的達成をとげるために動作構造体や、文についての代入順序、同期や動作のための準備状況テストなどの動作タイミングなどを規定し、その規定と仕様とを統合し、実行処理のために変換しうるモデルとしての役割をはたし、仕様が平行手続に変換される。

対象の感性モデルは、実行処理の結果得られた個別対象についての仮想表現レビューをし、感性的仮想対象世界として集積し、さらにこれを感性対象世界と比較評価しうるために、構成される。したがって、感性モデルは、インタフェースのためのプロトコルの基礎の上に構築される。感性モデルは、また適用分野についてユーザの持つイメージの表現であり、比較評価支援を行う。

3. オブジェクト・ネットワーク

論理表現の基本は、オブジェクト指向でありその構成要素として、名詞オブジェクトと動詞オブジェクトおよび、それらを修飾する制約条件が、存在する。また各オブジェクトを具体化するための環境ツールが存在し、オ

ブジェクト名が付けられる。名詞については、普通名詞名と固有名詞名があり前方参照機能が存在するようなオブジェクト環境を考える。

(1)名詞オブジェクト

カラー画像描画を例にとるとpoint, point sequence, line segment, line, region segment, regionの名詞オブジェクトが存在し、それらを参照できる。

さらに名詞オブジェクトにはluminance value, chrominance vectorのような属性値が付加され、属性オブジェクト構造ができる。たとえば、regionの集まりとしてのデッサンに属性値が与えられると、painted region⇒カラー画像、として描画が行われたことになる。

(2)動詞オブジェクト(関数的取扱)

a.名詞オブジェクト→名詞オブジェクト

b.名詞オブジェクト→属性付加名詞オブジェクト

が存在し、前者には合成的関数と分解的関数とがある。たとえばpoint sequenceが合成されてline segmentとなる。後者にも属性値を名詞オブジェクトに与える、すなわち描画としての関数と、逆に属性値関係から、名詞オブジェクトを抽出する画像解析のように逆関数関係(双対性)が存在する。

動詞オブジェクトの具体化には、

総称的関数+制約条件→具体的関数

の形式の採用によって、覚えることを少なくして、ユーザフレンドリ性を増加しうる。制約条件には、現在実行中の名詞オブジェクト名がシステムの現状をあらわすことを積極的に利用すると共に、制約条件集合によって、具体化の内容を決定する。

また動詞オブジェクトの具体的実行に際しては、

a.前提条件(pre):動作前の制約条件

b.実行中条件(in):動作中の制約条件

c.終了条件(post):終了時成立の制約条件

によって、システム動作の妥当性検定機能や推論機能に役立てる。

動詞オブジェクトは、名前前で特性づけられた名詞オブジェクトを、関係あるいは属性値付加をつけた名詞オブジェクトに変換する。演算目的対象となるオブジェクトは、単一である時と複数である時とがある。さらに対象オブジェクトが、目的語と、能動としての修飾あるいは環境指定でもありうる。たとえばplace A into Bはその例であって、Bについて制約について、intoに対応した妥当性や推論可能性などの支援ツールを必要とする。

名詞オブジェクトで構成される節点間は、動詞オブジェクト名をもつ枝によってネットワーク化される。したがってネットワークの実現は、オブジェクト関係の記述によって行いうる。

図2にオブジェクト・ネットワークの構成要素の例を示す。

4. 分野記述型言語とインタフェース

オブジェクトの記述は、分野特有のもので、専門用語的色彩をもっており、専門家が自由に使用しうると共に、非専門家に対しても使用可能とするような支援環境を備えていなければならない。

そのための言語アーキテクチャーとしては、

- (1)オブジェクト・ネットワークもオブジェクトであり、すべてのオブジェクトはウインド表記され、標準テンプレートをもっている。
 - (2)ウインドに表記された内容へのユーザ・インタラクションがすぐれたGUIを与える。
 - (3)動詞オブジェクトのもつ特性によって、妥当性検定機能や推論機能をそなえている。
 - (4)動詞と名詞に対応する変換過程や実行結果が明確であり、各種変更が容易である。
 - (5)データベース機能が十分である。
- などが重要である。これを具体的に記述し、システムを

実現するための言語を分野記述型言語のための核言語として規定されなければならない。言語としてのインタフェース機能としては、サービス工学的発想にもとずいて動詞オブジェクトはサービスモジュールであり、名詞オブジェクトは、サービス対象であって、両者の間では、標準テンプレートによってサービスの受渡しを行う。このため、原始的動詞名として(Request, Respond) 対のサービス機能を持ち、さらにイベント駆動とデータ駆動によってオブジェクト・ネットワーク中を走査して、データの完備性を獲得させる。画像描画のようなインタラクション・システムでは、実行処理表現は実行インタフェースによって実行順序が具体化される。

したがってアーキテクチャーとして、ネットワーク中をトークンが走ることは、実行であり、これを基本的に平行処理を行わせることは、ペトリネットワークによって肉づけすることに対応する。

分野記述型言語として、以上のような実行処理の他にオブジェクト・ネットワーク中に、分野特有の感性情報をモデル化し、生成物に対して実世界と比較評価しうる感性モデルを、組込むことに適応しなければならない。

5. まとめ

分野記述型言語の発想は、カラー画像処理描画のための言語Window Elaboration Language for Picture Processing and Painting(WELL-PPP)の開発に関連して得たもので、ソフトウェア・プロセスの研究のための良いケーススタディとなり、ソフトウェア生産の効率化、とくに平行システムに有用であることが半明しつつある。

芝浦工業大学の情報構造工学研究所として、院生、卒業生諸君の協力のもとに行った。重点領域「高度ソフトウェア」の補助を受け、富士通、KDDの支援のもとに研究を行ったことを記して謝意を表します。

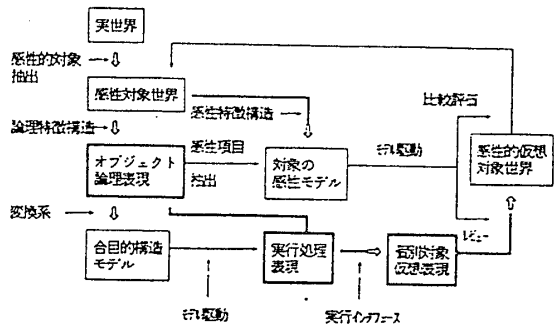


図1 世界と表現間関係

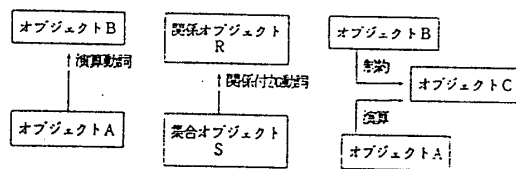


図2 オブジェクト・ネットワークの構成要素

1. 榎本、鴨志田、宮村：カラー画像処理描画システムの研究開発、情報処理学会第42回大会、4Q-1、1991、3
 2. M. Kamoshida, H. Enomoto, I. Miyamura : Window-based elaboration language for picture processing and painting, SPIE Visual Communication and Image Processing '91, vol. : 1606 pp951-960, 1991, 11