

# モデル駆動によるオブジェクトとプロセス間結合

1Y-5

榎本 肇      村尾 洋

芝浦工業大学

## 1 はじめに

マルチメディア開発の進展に伴って、その特徴に立脚した大規模ソフトウェアを効率よく生産する方法論の重要性が、多くの人々によって指摘されている。このような観点に立って、多種多様な分野<sup>[1]</sup>に対して複数のサーバが役割<sup>[2]</sup>に応じてインタラクション<sup>[3]</sup>を効果的に行うようなシステムの持つべき特性について論じ、その方法論をもとに拡張機能言語 Extensible WELL (Window-based Elaboration Language) の体系を画像についてのケーススタディとして開発を進め、生産効率性を示してきた。

本論文では、データモデルを基礎としてオブジェクトとプロセスのモデルを設定し、これらのモデル間を継承性及び合成性を利用して有機的に結合する。そして Extensible WELL と、部分的ではあるが実際に実現し画像描画に利用している、エージェント機能とによる協調処理について基礎付けを行う。

## 2 データと制約<sup>[4]</sup>

データは関数の入力引数、出力引数として数学的に表現され、システム内では、名詞オブジェクトの属性名としてデータ名、値としては属性値が与えられる。WELLの体系内ではデータ定義過程に入る時、名詞オブジェクト名に対応するテンプレートを用意する定義準備プロセス (defining process) と値を定めテンプレート内の対応位置に値を入れる定義操作 (define operation) とに段階を分割して取り扱う。動詞オブジェクトは名詞オブジェクトに対し、ある関数操作を行う。名詞格には普通名詞と固有名詞の格が存在し、限定詞 {a, the} によって、参照指示やデータ値の確定について一方性が与えられ、総称性と具体性が決定される。

修飾については形容詞や副詞で行うが、総称性についての修飾には制約優先性によって修飾の半順序が存在する。例えば featured と colored とを比べると、この順序で制約優先性が定まる。

名前  $\alpha$  で表現されたオブジェクトが  $\gamma$  で表記された性質を持つことを、内包論理学の記法に従って  $\gamma\{\alpha\}$  のように結合支援記法として書く。すると、それは  $\alpha$  で表記されたオブジェクトは  $\gamma$  で表記された性質を持つことを示している。従って、 $P\{j\}$  の述語の集合が  $\{j\}$  というオブジェクトを規定する。例えば line segment のリストが line を define することが、図1のオブジェクトネットワークに示されるような形でなされ、現在描画作業が行われている時 painted by {the identified colored line} = [the list of colored line seg.] であり、対応する作業過程で現れる the colored line seg. が答えとして求められる。制約の性質は内包的等式で定義されるが、外延的性質を持つ要素としては、次のように考えられる。

例えば symmetrical element {an identified element} により内包的には対称点/線についての相手要素の持つ

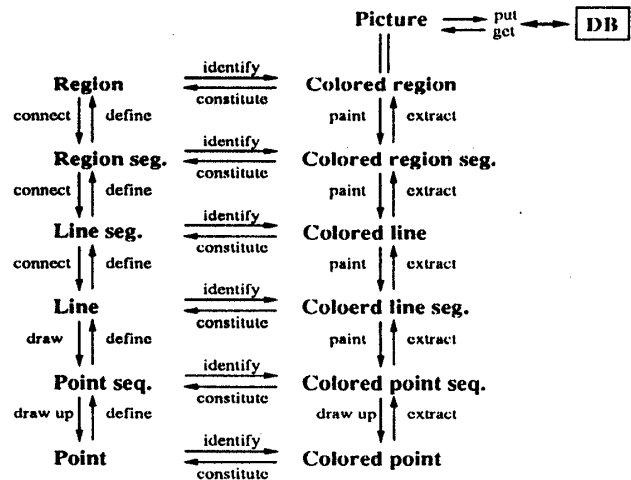


図1: 画像描画処理用オブジェクトネットワーク

べき性質が与えられるのに対し、外延的には、対応する要素がデータベースより検索されなければならない。

以上のように、データは具体的な属性の値として考え、制約は総称的オブジェクトの持つ性質と考える。処理システムの実現としてデータはクライアントが決定し、制約はサービスとしてその内容が決定されるようにし、同格的取扱いを行う。すなわち表記的には小括弧と中括弧の相違として処理を行う。

## 3 モデルの分類と構造

名詞、動詞及び形容詞・副詞の修飾詞はそれぞれテンプレートを持ち、語句について修飾内容のためのテンプレートを接続して統合化が行なわれる。そのために以下のようなモデルが設定される。

### (1) データ・モデル:

WELLの言語系では属性値構造をテンプレート形式のスキーマを用いて表現する。このデータ・モデルを通じて、オブジェクトの表記形式、データ操作などの管理を行い、データ制御を行わせ、各オブジェクトをプロセスと関連させる。

データの格として名詞オブジェクト、動詞オブジェクト、それらの修飾詞オブジェクト (形容詞、副詞) など考え、準自然言語形式の TELL と関連させる。

### (2) オブジェクト・モデル:

オブジェクト・モデルは、データ・モデルに従って形式モデルと特徴モデルに階層的に分割、表現する。

#### (a) 形式モデル:

画像でいえば点、線、領域から立体構造、動き構造などをデータモデルとしてテンプレートにより複合形式化し、すべてのパターン項目の形式表記を行う。複

数の単体間の関係、例えば前後関係による隠蔽や反射、屈折による新規オブジェクトもテンプレート表記し形式モデル化する。すなわち形式モデルによって構文的表現が行われ画像の場合は、幾何モデルに対応する。

動詞オブジェクトの形式モデルは、データ操作の実行開始のための precondition として継承すべき名詞オブジェクト名を状態制約としてもつのはじめ、制約名を指すテンプレート部、実行中制約としての incondition、実行終了後の制約としての postcondition を持つ。これらの制約は、名詞オブジェクトに名前として付加され指示される制約と結合して総称オブジェクトから具体オブジェクトへの関係が定義される。

#### (b) 特徴モデル:

名詞オブジェクトについての属性値をもとに特徴が表現され、それは環境についてある程度の不変性を持っている。その例は、分割線であり、顔画像の分割線で区切られた領域に目、鼻、口などの個別構成要素が配置される。ハイライト線も類似した性質を持ち、立体表現に大きな役割を果たす。

動きについての特徴モデルは、動きのための動詞オブジェクトと動き対象となる名詞オブジェクトとを中心に構成される。

#### (c) 感性モデル:

感性モデルは人間の感性を形式モデル、特徴モデルに埋め込むためのモデルであり感性語として名前と修飾パラメータとを随時指定し変更しうるようにする。

#### (3) プロセスモデル:

プロセスとして名詞や動詞オブジェクトが表現され、オブジェクト・ネットワークとしてグラフ表現される。この設計はクライアントのうちエキスパートによって行われ、WELL のグラフ構造エディタによってシステムが駆動可能とする。

さらに agent role server と specific role server が担当するプロセス間の協調および統合をはかるモデルが必要であり、形式モデルとしては各個別プロセス間のインタラクションの形態が定義される。各サービスは request, respond コマンドによって継承データと制約とをもって実行依頼し、合成データと制約とが実行結果として実行報告を行うことを基本として、より効果的に協調作業を行うように役割概念を用いて形式化される。ここでも上記、形式モデルとともに、下記の特徴モデルが可能である。

特徴モデルは、役割機能について環境に応じて制約機能を付加するためのモデルで、合目的演算モデルとして規定する。

## 4 モデル間結合

プロセス実行処理は、データ・モデルをもとにして行う。データは原則としてクライアントから与えられ、制約の内容はモデル内で定義されているものと、クライアントが環境に応じて規定するものがあるが、その種類としては次のものがある。

#### (a) And 制約: 名詞オブジェクトの属性値間、

ある名詞名への複数個の動詞オブジェクト間には次の制約規則がある。

(b) Or 制約 (独立性): ある名詞オブジェクト名に属する複数項目への指示間、ある名詞オブジェクト名から出発する動詞オブジェクト間

(c) 階層制約: オブジェクト・ネットワークでの処理手順上の上位オブジェクトと下位オブジェクトの関係から、現時点の上位オブジェクトの属性値集合は、下位オブジェクトの新規操作によって、整合化操作を受ける。ある動詞オブジェクト操作によって名詞オブジェクトの属性値を変化させた時、それより上位の名詞オブジェクト群について整合化修整を行う。

(d) 優先制約: 修飾詞間の優先規定、プロセスなどにおける役割機能の優先性を規定する。

プロセスの進行過程は、インタラクションとしての対話過程と考えると、

(a) プロセス動作の可視化→オブジェクト・ネットワーク

(b) クライアントとの対話性→コモンプラットフォームに情報集約化

(c) クライアントのユーザーとエキスパートの役割分担

(d) 協調過程の効率的表記→役割機能の制御

を満足させる形でモデル間結合を行わせることになる。そのためにコモンプラットフォームにユーザと関係するデータと制約が与えられ、表示されているオブジェクト・ネットワークで進行が管理される。オブジェクト・ネットワーク内の実行処理は request-respond ベースで行うが、その相手先でプロセスからデータまでの継承および合成についての制御規則がプロセスに関する優先性を含めた形で図2のように管理される。

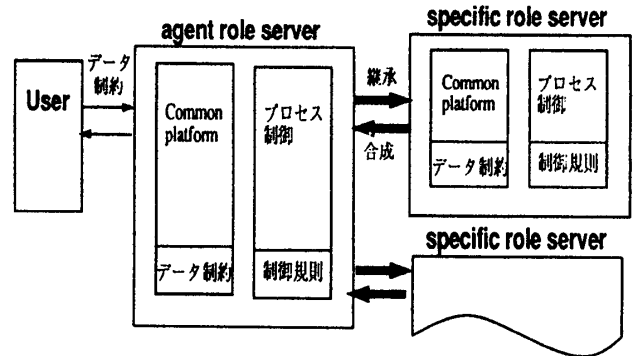


図 2: 実行処理の進行

以上のようにモデル間結合はデータ、オブジェクトおよびプロセスの各種モデル間の制約規則をもとに行われ、それが具体的に実行管理をすすめる。

## 5 まとめ

モデルとしてデータ、オブジェクト、プロセスを設定し、これらのモデル間の結合が制約を中心とした規則によって行われることを述べた。この結合方式によってモデル駆動が実行処理過程を規定し、ユーザ・フレンドリなシステム使用およびエキスパートによるシステム設計が可能となることを述べた。

これは研究室の協同研究の方針を示したもので、関係諸氏および各種援助を与えられた KDD、富士通の関係者に感謝します。

## 文献

- [1] 榎本、村尾: "インタラクションの形態分析", 情報処理学会第 50 回全国大会 2L-1 1995.3
- [2] 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 6F-3, 1992.3
- [3] H.Enomoto, Y.Murao: "Interactive Feature Coding System for Image Painting", Proc. of Advanced Image and Video Communications and Storage Technologies, Vol.2451, Mar.1995.
- [4] M.Aoki, Y.Murao, H.Enomoto: "Constraint processing in our extensible language for cooperative imaging system", Proc. of Electronic Imaging Science & Technology, IS&T/SPIE96 Vol.2663, Jan/Feb.1996