

# Extensible WELLにおける制約処理

2L-2

平井 郁雄 青木 稔 村尾 洋 榎本 肇

芝浦工業大学

## 1 はじめに

Extensible WELLを用いたインタラクション・プロセス<sup>[1]</sup>に於いて、並列協調型<sup>[2]</sup>のものを実現するためには、データ構造を表す名詞オブジェクト、また、関数を定義する動詞オブジェクトのそれぞれについて、その妥当性を実行中に常に評価していかななくてはならない。このような仕組みは、client-server間における対話の効率化を図るために構成されるものであり、制約処理と呼ばれる。

## 2 エージェント

Extensible WELLにおいては、clientとserver間のコミュニケーションの手段としてcommon platform<sup>[3] [4]</sup>を定義し、その上に可視的に明示されるオブジェクト・ネットワークによって対話が効率的におこなわれる。このようなオブジェクト・ネットワークは、1つの分野のサービスに対して、1つのネットワークが定義されるものであるが、サービス同志が相互に関連し合い、協調的に作業を行って、clientの意図実現を図ろうとする場合には、その作業手順が複雑化し、ユーザ・フレンドリー性の確保が難しくなってくる。このようなことを考慮するならば、クライアントには、作業を手助けしてくれるAgentと呼べる機能の存在が有益になってくる。この場合、Agent機能が必要とする知識は以下ようになる。

- (1) オブジェクト・ネットワークが定義するサービス内容、機能についての知識
- (2) 複数のサービスによる協調処理時における相互関係についての知識
- (3) サービスからサービスへ名詞オブジェクトが受け渡されるときに発生するデータ構造の整合性についての知識

また、Extensible WELLによるシステム設計を行う際には、このような知識を持ったExpertと呼ばれるclientによってAgent機能にこれらが組み入れられ、userと呼ばれる、実際の作業者によって利用される。この時、設計とは、サービス間の時相制約関係、形態制約関係の設定であり、それらの制約に乗った形でuserは、複雑な作業を進め、意図を実現できる。

## 3 時相制約

複数のサービスを利用して協調処理を行う場合には、その間にある相互関係を無視することは出来な

い。例えば、Agentによって立てられたサービス計画は、その時間的關係を明確しておくかねばならない。サービス間の関係には、同時に処理可能なものと順番的な優先度があるものが存在し、それらは並列性とタイミングを含んだ束構造で表される。この場合、並列性とは、同じタイミングで実行可能なサービス群のことを意味し、下図のように示される。

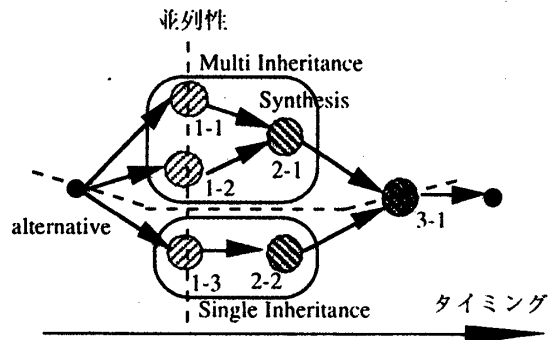


図-1 協調処理における束構造

図中において、縦に並んだサービスに関しては、別々に実行が可能であり、並列性を有している。また、データの継承という観点から見た場合、multiple inheritanceとsingle inheritanceという2つが考えられる。例えば、図-1の2-1での処理が1-1、1-2の処理結果を用いて行われる時(synthesis)などは、multiple inheritanceの場合であり、2-1が持つ制約条件によって同期が管理される。このようにサービス同志の結束性は连接的に表現され、この関係を時相制約として一元的に管理する必要がある。図-2は、束構造で表されるサービス間の関係を管理するのに用いられるテンプレート及び制約条件の例である。

### Data Template

Index	X	Y	Attributes for Point(X,Y)	●
-------	---	---	---------------------------	---

### Temporal Constraint

Name	Exe Flag	Pre-Indexes Flags	In-Indexes	Post-Indexes	●
------	----------	-------------------	------------	--------------	---

### Modal Constraint

等式形式

図-2 Constraint と Template

このようなテンプレートは、束構造に組み込まれる個別オブジェクトの1個1個に与えられ、互いを連

Constraint processing in the Extensible WELL  
 Ikuo HIRAI, Minoru AOKI, Yo MURAO,  
 Hajime ENOMOTO  
 Shibaura Institute of Technology

接させ、制約条件を付加する。数者択一の場合において、clientの選択前には、Generic Constraintとして総称的に捉えられていた制約条件は、その選択後のパラメータ入力によって制約条件の限定が行われ、Actual Constraintとして具体的な時相制約処理を行うことができる。つまり、clientが、選択したサービスのIndexをパラメータとして、それに対応するテンプレートからConstraint\_pointerを取り出した後にConstraintを参照し、次のサービス管理に必要なConstraintの書き換え作業を行う。これは、束構造の動的な変化に対応するのに有効である。例えば、clientがサービス束の途中でalternativeな立場に立ち、選択を行った場合には、Constraintを表すIn-Indexes内の記述に従って、選択されなかったサービスの実行許可について調整処理を行う。その他、multiple inheritanceにおける合成(synthesis)においても、table上のPre-Indexes内のFlagsの関係が制約条件として働きサービス間の関係を管理する。それに加え、clientによる作業のやり直し、又は、数者選択で選んだサービスの内容が気に入らず、選択をやり直したいときにも、Agentはclientによって知らされたIndexをもとにサービスの再計画を制約処理の1つとして行う。これは、Constraintを元的に管理している各Index、Pre-indexs、Post-Indexesを書き換えることによって容易に行うことができる。また、このような、束構造の定義は、サービス間の相互関係を熟知したExpertによってグラフ構造エディタ<sup>[3]</sup>を用いてなされる。

#### 4 形態制約

Agentが立てたサービス計画を実行する際には、サービスからサービスへの名詞オブジェクトの受け渡し、つまり、データの継承が行われることになる。その時に、それらを持つデータ構造についての整合性を保持しておかなければ、円滑なコミュニケーションをはかることは出来ない。つまり、Agentは、名詞オブジェクトをサービス間で伝達してゆく役割を持っており、そこで、問題が生じないようにオブジェクト・ネットワークが定義しているサービスの間で形態調整を行う必要がある。個々のオブジェクト・ネットワークには、それぞれテンプレートが定義され、サービス間でデータが継承される際には、これらのテンプレートがインターフェースとなって名詞オブジェクトが伝達される。Agentは、このようなテンプレートを手がかりにして、オブジェクト・ネットワークによってなされたサービスの結果を検証し、サービス計画に沿ったものかどうかを確かめなくてはならない。テンプレート内に用意されているConstraint\_pointerという領域は、どの様な妥当性チェックを行う必要があるのかをAgentに知らせるために存在している。また、図-3は、このようなサービス結果内容チェック<sup>[5]</sup>の位置付けを表している。サービス結果内容チェックは、実行結果を受け取ると、その妥当性チェックの要求を出す。まず、必要なのは、Constraint\_pointerを形態制約用のConstraint<sup>[4]</sup>に参照して、一体どの様な制約条件が課されているのかを知り、それを検証するためのプロセスへ処理の要求を出すことである。そして、そこでの検証

結果をサービス結果内容チェックに返す。このようなConstraint\_pointerとConstraintの関係は、時相制約について述べたものと同じ形態をとっており、全体的に見た場合、階層構造が構成されていることを示している。

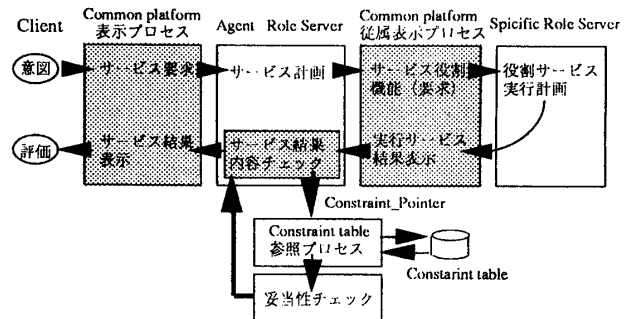


図-3 サービス結果内容チェック

また、このような形態制約処理を利用した例として、顔画像の生成などが挙げられる。顔は、その部品に影響を及ぼす支配領域によって表され、その領域間の関係を形態的な制約として、Constraintに登録し、clientが変化させた領域のConstraint\_pointerとマッチングすることで制約処理を行う。このようなものは、Expertによって設計段階で設定されることになる。

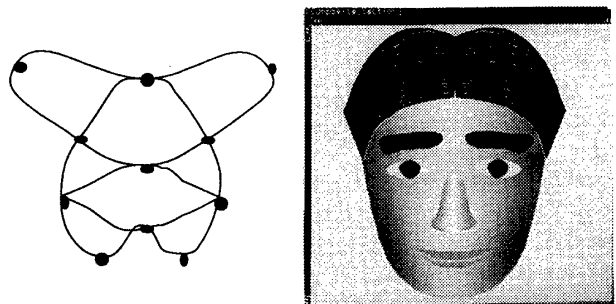


図-4 顔画像における形態制約

#### 5 まとめ

本論文では、Extensible WELLによる協調処理に必要な制約処理に関して、時相制約、形態制約といった観点から、その内容について述べた。このような制約処理を用いた対話プロセスのポイントは、clientの負担を減らすこと、拡張性を確保することであった。

#### 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50回全国大会,2L-1,1995.3
- [2] 青木,平井,村尾,榎本, "Extensible WELLにおける協調処理", 情報処理学会50回全国大会,2L-3,1995.3
- [3] 宮本,平井,青木,村尾,榎本, "拡張機能,言語Extensible WELLの体系化", 情報処理学会48回全国大会,4G-08,1994.3
- [4] I.Hirai, Y.Miyamoto, Y.Muraio, H.Enomoto, "Service management in the picture processing and painting system having extensible functions", Visual Communications and Image Processing'94, SPIE Vol.2308
- [5] 榎本, "要求意図の対話的詳細化プロセス", 情報処理学会48回全国大会,4G-06,1994.3