

Extensible WELLにおける協調処理の制約的実現

1 Y-6

青木 稔 村尾 洋 榎本 肇

芝浦工業大学

1 はじめに

拡張機能言語 Extensible WELL(Window-based ELaboration Language)は、分野記述言語でありある特定の分野を指定すると分野記述言語になりうる言語である。ウインド指向言語でありクライアントとのインタフェースはコモンプラットフォームと呼ばれるエリアである。またオブジェクト指向の言語であり、サービスを品詞を用いたオブジェクトで表現する。サービスは当事者間でのインタラクションによって実現し、各当事者はそれぞれ固有の役割を持っている。その役割に互いに依存しあいながら目的を達成する役割依存型システムである。また独立性を持つプロセスは並列実行される協調型並列実行システムであり、プロセスの進行、統合には制約処理がなされる。本言語ではデータと制約を統一的な管理方法によって制御する。本論文では、あるサービスにおける当事者間の協調処理の実現方法を制約を用いて管理する手法を示す。

2 エージェント機能の拡張

拡張機能言語 Extensible WELL には、エージェント機能が組み込まれてきた。このエージェント機能はより複雑なサービスを実現するため、必要な機能を集め知識ベースを格納したものであり、サービス計画を行っている。さらに高度なサービスの実現と操作の快適性を可能にするため、最優先当事者の概念とデータ欠落時におけるデータ探索機能をエージェント機能の拡張として追加する。

ある個別サービスを行う場合、そのサービスを誰が行うかという問題が起こる。そのサービスに関して知識を持っているのは誰なのか、誰がそのサービスを行うのが妥当であるのかを知っている必要がある。その適任者とそのサービスにおける最優先当事者何という。エキスパートは各個別サービスの最優先当事者を登録する。

ある点を画面上に書くという単純なサービスを考える。このサービスは Extensible WELL 上では「DRAW UP POINT」として表現される。「DRAW UP」という動詞オブジェクトに対して目的語が「POINT」という名詞オブジェクトで付随している。このためエージェントロールサーバは描画用のオブジェクトネットワークである要素ネットワークのNONEからPOINTまでを利用すればいいとエキスパートの登録に基づいて判断する。

Implementation of the cooperative Extensible WELL with constraint processing
Minoru AOKI Yo MURAO Hajime ENOMOTO
Shibaura Institute of Technology

要素ネットワークの一部を新たな点描画用のオブジェクトネットワークと見なす。さらにエージェントロールサーバはその新たなオブジェクトネットワークを実行しようとする。その際に必要な点の座標 (X, Y) が欠落しているのが分かると、誰がそのデータを入力すべきかを判断し、この場合どこに点を打てばいいのかわ知っているのはユーザのみであるので、ユーザに場所を指示するようにデータウインドがコモンプラットフォーム上に表示される。点の位置が指定されると、そのデータを元に作業を進めサービス結果をデータウインドウ上に表示しユーザに評価をゆだねる。

一方、あるオブジェクトに対して対称な点を書きたい場合は、何に対して対称なのかをユーザに指定してもらった後に、このサービスの最優先当事者をエージェントロールサーバが判断する。この場合、あるオブジェクト（例えば点）に対して対称な点を書くことになるので、誤差のあるユーザに入力してもらうよりは、正確な位置を計算できるサービスモジュールにした方が好ましいことになる。

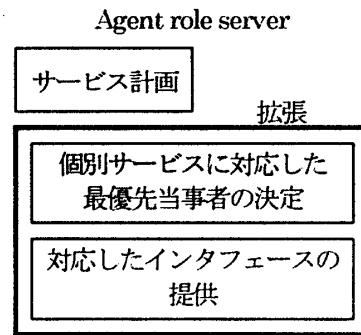


図. 1 エージェント機能の拡張

このようにエージェント機能では、サービスの内容とオブジェクトネットワークの対応とその部分定義、そして各サービスに対応した最優先当事者に対してインタフェースを提供している。

3 サービス分割

有用なエージェント機能を実現するためには、複雑なサービスを細かく分割していくが必要になる。その分割には大きく分けて2種類に分類される。層分割と並列分割である。

層分割ではネスト形式で表現され、3つの状態、プレ、イン、ポストをもつ。プレ状態においては必要なプリプロセスが実行され、イン状態では次の層に対する実際の

サービスが行われ、ポスト状態では行われたサービス結果の妥当性検査が行われる。

並列分割はサービスの相互排他性のあるものに対して行われる。分割されたサービスは並列実行が可能である。

4 制約分類

分割されたサービスはそれぞれ制約 α を用いて管理される。層分割に対しては時相制約、並列分割に対しては形態制約が用いられる。

時相制約では、必要な条件が AND 制約として扱われる。即ち必要な条件の全てがそろわなければ、次の状態へは移れない。先の「DRAW UP POINT」の例における点の座標データはこの操作における時相制約といえる。独立性を持つプロセスには、OR 制約独立性が当てはまり、並列実行が可能となる。

形態制約については、上記並列分割された際に、独立に出されたサービス結果を統合する場合に用いられる。例えば相互独立に書かれた画像を合成する場合、それぞれの相対的な大きさや向き、角度などがそれにあたる。また、プロセスに関する制約もこれに含まれ、階層制約という。ある下位オブジェクトの新規操作をするとそれに伴って上位オブジェクトにおいても整合比調整をする必要がある。さらに前述の役割機能の優先性についての制約として優先制約がある。これはプロセスにおける最優先当事者を決定するのに用いられる。

5 制約処理の実現

上記のような制約条件を用いて制約処理を行う場合、本言語でこれまで用いられてきた定義準備(Defining process)と定義操作(Define operation)を拡張して管理を行う。定義準備において対応するテンプレートを準備し、定義操作においてデータをテンプレートに格納するという基本概念は継承する。ただ、制約処理を実現するためには、テンプレート自身が動的に変更可能である必要がある。つまりテンプレート群、テンプレートリストを管理する必要がある。動的に変更されるテンプレートを決定するのが制約条件である。

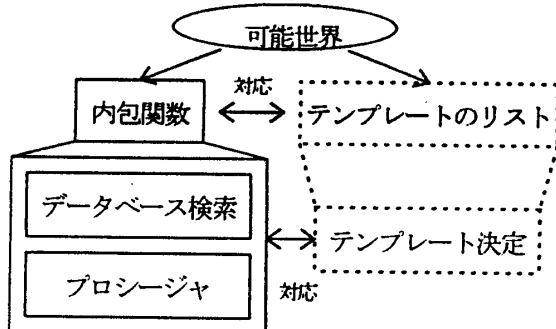


図2. 制約処理と内包論理

Extensible WELL においてデータとは値指向で定義される必要がありその値は既に定義されていなければならない。一方制約については、名称指向でありその値自身は定義されていない。名称参照のみが可能で、内包関

数に可能世界を与えることによって外延へととなり、値が定義され、データと等価になる。上記の動的なテンプレートの変更もこの内包関数のために必要になる。

先に述べたデータと制約を統一的に管理するという方法論は内包論理式によって実現される。データは α (β) で表現され、制約は α (β) で表される。この両者の関係式は Brace convention によって以下のようになる。

$$\alpha \{ \beta \} = \alpha (\beta)$$

オブジェクト β は属性 α を持っている。{} は可能世界を与えることによって属性 α を満たすようなオブジェクト β を生成することを表す。この生成が制約処理であり、2種類の生成方法がある。データベース検索とプロシージャである。

データベース検索の例としては HIGHLIGHT POINT の抽出がある。指定された水平捜査線上の点の中で、エキスパートによって指定された、ある一定以上もしくは最大のハイライト値を持つものを抽出する。この場合の HIGHLIGHT は名詞である POINT を修飾しているので形容詞と捉えることができ、ある特定の POINT に対する限定詞として使われている。

プロシージャの例としては、先のあるオブジェクトに対して対称に点を打つ例が当てはまる。この場合オブジェクト POINT は、あるオブジェクトに対して対称であるという属性を持ち、エージェントロールサーバはその条件を満たす POINT を生成するために必要なプロシージャを呼ぶ。エージェントロールサーバはこの作業に対する最優先当事者を知っているのでそちらへ依頼することになる。この場合の最優先当事者が人間であるユーザの場合と、機械であるサーバの場合とでは、インタフェースが異なる他は等価と見なされる。つまり、ユーザもサーバの一つであるということである。

こうして求められた値は定義操作として決定されたテンプレートに格納され処理が進んでいく。

6 まとめ

このように協調処理を進めるに当たって、必要となるエージェント機能の拡張とそれを可能にする制約処理の仕組みを示した。制約処理は内包論理式によって表され、求められた値はデータと等価な扱いとなり、データと制約の統一的な管理方法を実現した。またそれらに役割や最優先当事者の概念を導入することによって、単純な方法によって複雑なサービスを実現することを可能にした。全体として、拡張機能言語 Extensible WELL における協調処理の制約的実現を可能にしたといえる。

文献

- [1] 榎本, 村尾, "モデル駆動とプロセス間の結合", 情報処理学会 5 2 回全国大会, 11-5, 1993.3
- [2] 平井, 青木, 村尾, 榎本, "Extensible WELL における制約処理", 情報処理学会 5 0 回全国大会, 2L-2, 1995.3
- [3] Maeki, Y., Murao, H., Enomoto, "Constraint processing in our extensible language for cooperative imaging system", Electronic Imaging: Science & Technology: IS&T/SPIE 93 Vol. 2933
- [4] 榎本, "ソフトウェア工学ハンドブック", オーム社 1990, p.14~p.16
- [5] 榎本, 村尾, "インタラクションの形態分析", 情報処理学会 5 0 回全国大会, 2L-1, 1995.3