

Extensible WELLにおけるサービス・マネージャ

4G-8

平井 郁雄 宮本 泰秀 村尾 洋 榎本 肇

芝浦工業大学

1. はじめに

Extensible WELL(Window-based ELaboration Language)は、その中に持っているサービス・マネージャを上手に組み合わせることによって拡張機能言語としての特徴を持つことが出来る。特にコミュニケーション・マネージャは、コモン・プラットフォーム [2]、[3] と Extensible WELL kernelおよび他のマネージャとの円滑なコミュニケーションを実現し、その重要度は高い。

2. Extensible WELLのサービス・マネージャ

Extensible WELLは、サービス・マネージャとして、(1) コミュニケーション・マネージャ、(2) ウインド・マネージャ、(3) ディスプレイ・マネージャ、(4) ファンクション・マネージャを持っている。それらは、コモン・プラットフォームを通して入ってきたクライアントの意図 (intention)を実現するために存在するが、Extensible WELLでは、出来るだけ自由度と拡張性を維持するための構成を考慮している。

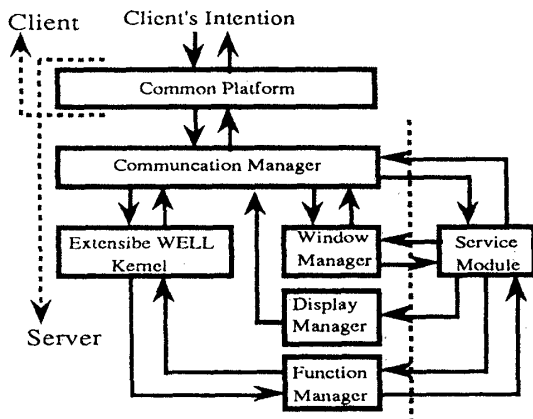


図1. マネージャの相関関係

その構成の特徴は、クライアントの意図がシステムに伝えられるときには、必ずコミュニケーション・マネージャを経由してカーネルや各マネージャに伝えられるということである。このため、コミュニケーション・マネージャはコモン・プラットフォームと直結している唯一のマネージャとなり、対話プロセス [1] を集中的に管理している。

2-1. コミュニケーション・マネージャ

コミュニケーション・マネージャは、Extensible WELLの拡張性を維持するためにデータを間接的に管理する。

そのためデータは、テンプレートとして一元的に管理され、そこに属性を自由に設定できるようになっている。

index	X	Y	attributes for Point (X,Y)
-------	---	---	----------------------------

図2 テンプレート概念図

テンプレートは、サービス・モジュール内ではインデックス [3] によって管理されるが、コモン・プラットフォーム (データ・ウインド)を通してクライアントから与えられるものは、データ・ウインド上の座標値である。サービス・モジュールは、テンプレートのインデックス検索を行えないので、ここで座標値をインデックスに変換し、それをサービス・モジュールに引き渡す。更に、コミュニケーション・マネージャは、Extensible WELLの持つ対話プロセスを集中管理する。コモン・プラットフォームは、クライアントにオペレーションを伝えるオブジェクト・ネットワーク、メッセージや選択項目を伝えるメッセージ・ウインド、実際に描画などの作業を行い、画像データなどを表示するデータ・ウインドなどを持つが、それらのウインドを通して得られたクライアントの意図を実現するために、コミュニケーション・マネージャは、その意図の入力に応じて、それを各マネージャやカーネルへ伝える。コミュニケーション・マネージャのこのような対話プロセスの一元管理により、データが必ずここを通過することになり、テンプレートのチェックを行ったり、制約条件のチェックを行うことも可能になってくる。

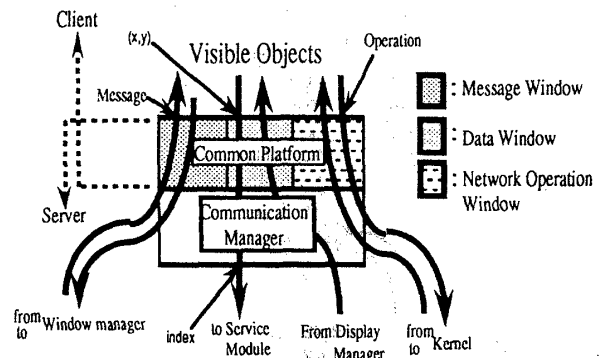


図3. コミュニケーション・マネージャ

3-2. ウインド・マネージャ

Extensible WELLは、コモンプラットフォームを生成するためにウインド・マネージャを有している。また、クライアントとシステムの間きめ細かい協調処理を実現するためのメッセージ・ウインドもここで生成される。

3-3. ディスプレイ・マネージャ

サービス・モジュールがテンプレートの中に持ってい

Service Managers of Extensible WELL
 Ikuo HIRAI, Yasuhide MIYAMOTO, Yo MURAO, Hajime ENOMOTO
 Shibaura Institute of Technology

る値は、ディスプレイ・マネージャによってコモン・プラットフォームを通して可視化対象 (visible object) として表現される。

3-4. ファンクション・マネージャ

Extensible WELLは、サービス・モジュールの集中的な管理を実現するためにファンクション・マネージャを有している。クライアントがコモンプラットフォーム (オペレーション・ウインド) を通して出した意図は、Extensible WELL Kernelを経てファンクション・マネージャへと渡される。

4. Extensible WELLの制約動作

クライアントは、オブジェクト・ネットワークに従った作業の流れの中で、自分がデータ・ウインド上に生成していったデータの正当性を確認したいと思うことがあると考えられる。そのようなクライアントが確認したいと思う事柄を一つの制約条件として捉えることが出来る。このような制約条件は、オブジェクト・ネットワークのそれぞれの名詞オブジェクト (テンプレートで表されるデータの状態) に付属的なものとして課され、サーバ側では、その制約条件とテンプレートの内容を比較することで確認作業を行うことが出来る。ここで問題にされるのは、サーバーが、処理対象とされるテンプレートの特定方法 (identify) をどのように持つべきかの問題である。Extensible WELLでは、コミュニケーション・マネージャの機能を利用してこれを実現できる。コミュニケーション・マネージャは、オブジェクト・ネットワークの持つテンプレートを間接的に管理しているので、クライアントがマウスでヒットした座標値から処理対象を特定するための情報をテンプレートから引き出して、チェック・モジュール (具体的に制約条件をテンプレートに照らし合わせるサービス・モジュール) に渡すことが出来る。チェック・モジュールでは、その情報を頼りに、その制約条件が満足されているかどうかを確認し、その結果をメッセージ・ウインドを通してクライアントに知らせる。また、このような制約動作も、Extensible WELLの持つマネージャがあれば、他のサービス・マネージャと同様な方法で実現される。

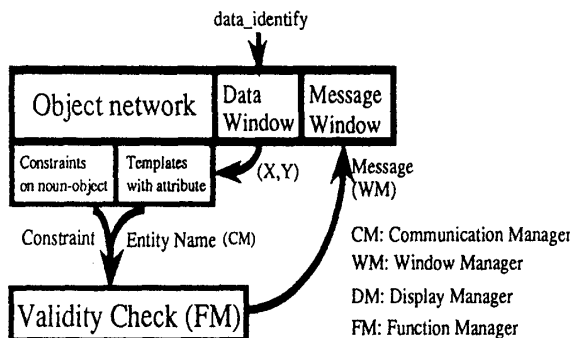


図4. 制約動作

4. サービス・マネージャの動作例

ここでは、図形描画によって描かれたラインにどのような制約動作がなされるかを1つの例として、各マネージャの動きを示す。ラインは、それ自体がデータの状態を表しているため、名詞オブジェクトとして捉えることが出来る。ラインという状態に対して課され得る制約条

件として考えられるのは、「連続性」や「対称性」といったものである。名詞オブジェクトが持つ、そのような制約条件は、ウインド・マネージャがメッセージ・ウインドを通してクライアントに知らせる。

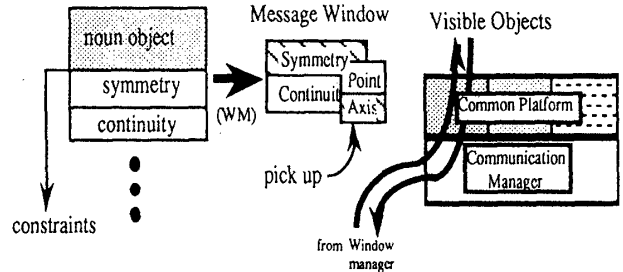


図5. 制約条件の表示

チェック・モジュールは、メッセージ・ウインドを通して起動要求をクライアントから受けるとファンクション・マネージャによって起動される。チェック・モジュールは、必要があればクライアントに対してアイデンティファイ作業の要求をメッセージ・ウインドを用いて行う。クライアントは、それに対してマウスでデータ・ウインド上の可視化対象をヒットすることでコモン・プラットフォームを通じたコミュニケーション・マネージャへの座標値の引き渡しを行う。座標値は、コミュニケーション・マネージャによってインデックスに変換されるとチェック・モジュールへ送られ、このインデックスを使ってチェック・モジュールは、テンプレートから具体的な属性値を引き出してチェックを行う。チェックされた結果は、メッセージ・ウインドによってクライアントに知らされる。

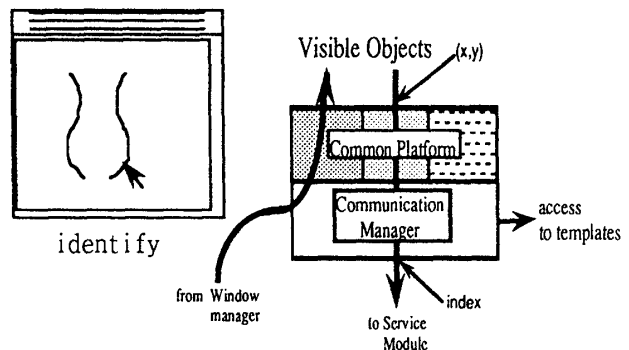


図6. アイデンティファイ作業

5. まとめ

以上、Extensible WELLの持つサービス・マネージャについての説明と具体的な動きについて示した。コミュニケーション・マネージャを中心にどのように名詞オブジェクトに制約条件を課していくかは、そこに感性的な処理を実現できる可能性も含んでおり、Extensible WELLの今後の発展が期待できる。

文献

[1] 榎本肇 要求意図の対話的詳細化プロセス 情報処理学会第48回全国大会 1994.3
 [2] 宮本、平井、村尾、榎本 拡張機能言語Extensible WELLの体系化 情報処理学会第48回全国大会 1994.3
 [3] 青木、平井、宮本、村尾、榎本 拡張機能言語Extensible WELLの操作環境 情報処理学会第48回全国大会 1994.3