

分散環境におけるアプリケーション協調制御方式の一提案

2U-8

根尾 秀一 神谷 一隆 趙 玲 伊與田 光宏

千葉工業大学

1. はじめに

近年、コンピュータネットワークの発展に伴い情報資源や作業環境をネットワーク上の複数サイトに分散化する傾向にある。このような環境で複数の人間が互いに協調して作業を行うためには、分散する様々なアプリケーションプログラムやデータ資源に対してシームレスなアクセスを実現する必要がある。

本研究では分散環境における協調作業を柔軟かつ円滑なものとするため、リアルタイムメッセージ通信に基づくアプリケーション間通信モデルを定義し、スクリプト記述による協調制御方式を提案する。

2. 現状のアプリケーションにおける問題点

アプリケーションはコンピュータの高性能化により多彩な機能を備えるようになった。これによりPCやWSを利用する個人ユーザの作業環境は高レベルになり、コンピュータの応用分野も多様化している。しかし個々のアプリケーション動作は独立的で他のプログラムとの連携・協調を考慮されているものは少ない。よって複数のアプリケーションに渡る作業を実行するためにはユーザ自身がアプリケーション機能に関する知識を持ち、利用計画を立てなければならない。ユーザの作業対象領域がコンピュータネットワークへと拡大し様々な資源が分散する環境下では、このようなプロセスはさらに複雑化しユーザの負担を増加させている。以上の問題解決にはアプリケーションが協調してユーザ要求を達成するような動作環境を実現する必要がある。

3. アプリケーション間通信モデル

2章における問題解決の一手段として本研究ではアプリケーション間通信モデルを定義する。図1に本モデルの概要を示す。

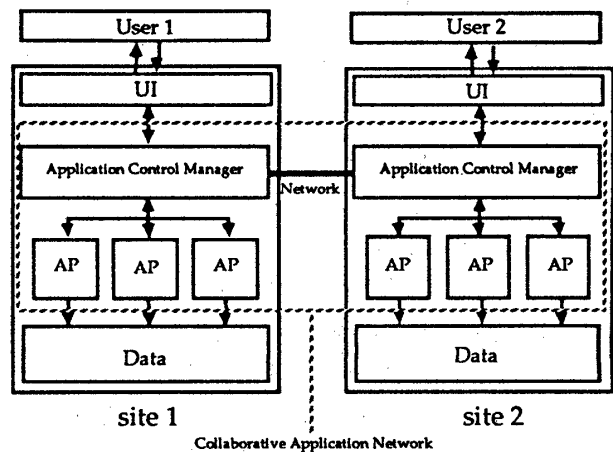


図1 アプリケーション間通信モデル

本研究ではアプリケーションを機能面から2つのセクションに分類してとらえる。1つはメッセージにより定められた命令に対して常に受動的な立場をとるセクションである。これはツールとして固有の機能を持ち、対象とするデータを特定し処理を実行する。一方、これらのツールを管理し利用計画を立てるセクションが考えられる。これはユーザからの作業要求メッセージに基づいて利用するツールやデータを特定し、各ツールに対して適切な動作命令を送信する役割をもつ。このような点を考慮し、本モデルではアプリケーションの統合管理機構としてApplication Control Manager（以下ACM）を定義する。ACMはローカルサイト内のアプリケーションを協調的に制御すると同時に分散するACMとネットワークを介して通信する。これによりアプリケーションレベルでの協調ネットワークが形成され、ユーザはUIを介してこのネットワークにアクセスすることによりサービスを受けることができる。

A study of collaborative control method of application based on distributed environment  
 Syuichi NEO, Kazutaka KAMIYA, Chou LEI,  
 Mitsuhiro IYODA  
 Chiba Institute of Technology

#### 4. 協調制御方式

前項で述べたモデルにおいてACMは協調制御の核となる存在である。本章ではACMの構成およびこれを実現する記述言語について述べる。

##### 4.1 ACMの構成

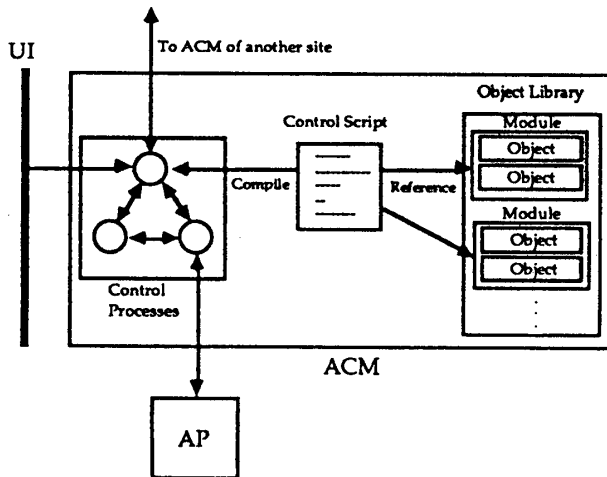


図2 ACMの構成

図2にACMの構成を示す。ACMは複数のオブジェクトと制御プロセスにより構成される。オブジェクトはカテゴリごとにモジュールを形成し、ライブラリとして統合化されている。制御プロセスはライブラリを参照して記述される制御スクリプトを実行形式に変換することにより生成される。ユーザからの作業要求はUIとの対話プロセスによりパラメータに分割されメッセージとして制御プロセスに送信される。制御プロセスはプロセス間およびアプリケーションとの協調を通じて処理を実行する。

##### 4.2 記述言語

ACMを具体化するためにはアプリケーション固有オブジェクトや協調的な通信制御を記述可能な言語が必要となる。このような要求に基づき本研究ではAppleScriptを採用した。この言語はアプリケーションのオブジェクトモデルに対応しハイレベルな制御記述が可能なスクリプト言語である。

```
script [scriptObjectVariable]
  [(property | prop) propertyLabel: expression ]
  [handlerDefinition]
  [statement]
end [script]
```

図3 スクリプトオブジェクトの記述仕様

前述したオブジェクトは、この言語のスクリプトオブジェクトの概念を用いて定義することができる(図3)。スクリプトオブジェクトは複数の制御プロセスに共有される汎用プログラムパーツである。また制御スクリプトは対象となる複数要素(アプリケーション、制御プロセス、スクリプトオブジェクト)に対してメッセージ送信とオブジェクトデータ交換の手順を明確にする。これにより一連の協調処理の流れを記述可能としている。図4に制御スクリプトの記述例を示す。

```
set TargetModule1 to load script Module1
global SharedVariable

tell TargetApplication1
  try
    set SharedVariable to word of paragraph of
    window
  on error
    [statement]
  end try
end tell

tell TargetApplication2
  try
    set cell of Record to SharedVariable
  on error
    tell TargetModule1
      tell ScriptObject1 to run
    end tell
  end try
end tell

⋮
```

図4 制御スクリプトの記述例

#### 5. おわりに

本稿では現状におけるアプリケーションの問題点を指摘し、この解決の手段としてアプリケーション間通信モデルを定義した。さらに、このモデルにおける協調制御をスクリプト言語を用いて記述する手法を提案した。今後はアプリケーションの自律的動作を考慮した拡張モデルを提案していきたいと考えている。

#### 参考文献

- [1]根尾 他：アプリケーション間通信による協同作業支援システム,電子情報通信学会秋季大会, D-142,September,1994.
- [2]根尾 他：アプリケーション間通信による協調作業環境の構築,情報処理学会春季大会,7M-3,March,1995.
- [3]片岡 雅憲：ソフトウェア・モデリング,日科技連,1988.