

非対称協同作業のための

3M-3

既存アプリケーションの共有方式

坂入 隆、篠崎 雅英、小林 真
日本アイ・ビー・エム（株） 東京基礎研究所

1. はじめに

複数の計算機を接続し、アプリケーションを共有することによって協同作業を行うシステムが多く開発されている。これらのシステムは、目的に応じて様々な仕組みで実現されている。

共同作業には、会議のようにそれぞれのユーザが同じような役割を持つ対称な協同作業と、顧客と顧客対応係のように異なった役割を持つ非対称な協同作業がある。非対称な協同作業では、それぞれのユーザの役割に応じて許される操作に違いがある。そのため、非対象な協同作業のためのシステムは、対象な協同作業のためのシステムとは違った機能が必要となる。

本稿では、既存のアプリケーションを共有しながら、非対称な協同作業を可能とする方式についてWWWブラウザの共有を例にして述べる。

2. 例: WWWブラウザの共有

WWWブラウザを複数のユーザ間で同期させることができる機能としては、(1) ウィンドウの大きさ、(2) スクロールの位置、(3) ウィンドウ上での注釈、(4) ポインタ、(5) URLの変更、(6) フォームの入力などが考えられる。これらのうち、(1) から (4) まではアプリケーションの共有に一般に必要な機能であり、(5) と (6) はWWWブラウザの共有に固有な機能である。

また、それぞれの機能ごとにユーザ間で同期させたり、同期させなかったりといった使い方もありうる。例えば、(5) と (6) の機能だけをユーザ間で同期させるといった使い方である。

さらに、それぞれの機能ごと、それぞれのユーザごとに、操作可能であるか操作不可能であるかを設定することも考えられる。このようなことができれば、共有WWWブラウザを非対称な協同作業に使うことが可能となる。

3. 従来技術

WWWブラウザを協同作業に使うためのアプローチには、次のようなものがある。

2.1 コラボレーションアンアウェア

共用ウィンドウシステムなどを使うことによって、既存のWWWブラウザをそのまま協同作業に使う[2]。このようなアプローチをコラボレーションアンアウェアと呼ぶ。

このアプローチの利点は、次の通りである。

1. ユーザは、普段使っているWWWブラウザをそのまま協同作業に使うことができる。
2. 協同作業用のWWWブラウザの開発の手間が不要である。

このアプローチの欠点は、次の通りである。

1. 協同作業を支援するプログラムは、WWWブラウザに関する知識がないので、WWWブラウザの特性にあった協同作業を支援することができない。例えば、特定のユーザに対してURLの変更は許すがフォームの入力を禁止するといったことはできない。
2. 低レベルのメッセージをWWWブラウザ間で送受信するために、ウィンドウの状態を厳格なWYSIWIS (What You See Is What I See) にする必要がある。例えば、URLの変更とフォームの入力は同期させるが、ウィンドウの大きさは同期させないといったことはできない。また、通信量は多くなる。

2.2 コラボレーションアウェア

協同作業のためにマルチユーザ用のWWWブラウザを開発する[1]。このようなアプローチをコラボレーションアウェアと呼ぶ。

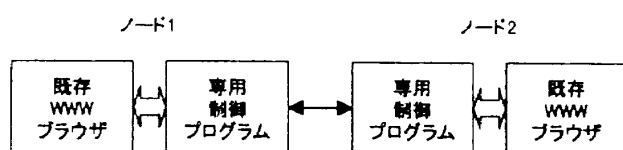


図1: 外部からの制御による共有の仕組み

このアプローチの利点は、次の通りである。

1. WWWブラウザの特性にあった協同作業の支援を行うことができる。
2. 高レベルのメッセージを送受信するために、ウィンドウの状態を厳密なWYSIWISにする必要がない。また、通信量を少なくできる。

このアプローチの欠点は、次の通りである。

1. ユーザは、新しいWWWブラウザの使い方を覚える必要がある。
2. 協同作業用のWWWブラウザの開発の手間が必要である。

4. 外部からの制御による共有

我々は、コラボレーションアンウェアとコラボレーションウェアの両方のアプローチを組み合わせることによって、両方の利点を併せ持つ既存アプリケーションの共有方式を考案した。そして、このアプローチをWWWブラウザであるNetscape Navigatorの共有に適用した。

我々のアプローチの特徴は、次の通りである。

1. コラボレーションアンウェアと同じく既存のWWWブラウザをそのまま協同作業に使う。
2. 既存のWWWブラウザをそのWWWブラウザ専用の外部プログラムから制御することによって、協同作業の支援を行う。外部プログラムからの制御には、ウィンドウメッセージのフックやDDE (Dynamic Data Exchange) などを用いる。

このアプローチの仕組みを図1に示す。それぞれのノードで既存のWWWブラウザ及びそのWWWブラウザ専用の制御プログラムを実行する。そして、制御プログラム同士が通信し、協同作業を支援する。WWWブラウザの状態が変わったことを検知した制御プログラムは、別のノードの制御プログラムにメッセージを送ることによって通知する。そのようなメッセージを受け取った制御プログラムは、そのノードのWWWブラウザの状態を変更する。

このアプローチの利点は、次の通りである。

1. ユーザは、普段使っているWWWブラウザをそのまま協同作業に使うことができる。

2. WWWブラウザ本体の開発は不要である。
3. WWWブラウザ専用の制御プログラムを用いるため、WWWブラウザの特性にあった協同作業の支援を行うことができる。
4. 高レベルのメッセージを送受信するために、ウィンドウの状態を厳密なWYSIWISにする必要がない。また、通信量を少なくできる。

このアプローチの欠点は、次の通りである。

1. 外部から協調作業のための制御を行うために十分なAPIをアプリケーションが持っていなければならない。最低限必要なAPIは、アプリケーションの状態が変更されたときに外部に通知することと、外部からの指示でアプリケーションの状態を変更することである。Netscape Navigatorの場合には、このためのAPIとしてNetscape Client APIを使うことができた。また、最近のアプリケーションはこのようなAPIを持っていることが多いので、そのようなアプリケーションをこのアプローチを用いて共有することができる。
2. アプリケーションごとに専用の制御プログラムを開発する必要がある。これは、アプリケーションの特性にあった共同作業を支援したいということに起因している。しかし、このような制御プログラムのためのツールキットを用いることによって開発を効率的に行えるようにできる[3]。

5. おわりに

コラボレーションアンウェアとコラボレーションウェアの両方のアプローチを組み合わせることによって、両方の利点を併せ持つ既存アプリケーションの共有方式を考案した。また、この方式をWWWブラウザの共有化に適用し、有効性を確認した。

参考文献

1. Greenberg, S. and Roseman, M., GroupWeb: A WWW Browser as Real Time Groupware, ACM SIGCHI'96 Conference on Human Factors in Computing System, Companion Proceedings, pp. 271 - 272 (1996).
2. Microsoft Corporation, NetMeeting. Available at <http://www.microsoft.com>
3. Sakairi, T., Shinozaki, M., and Kobayashi, M., CollaborationFramework: A Toolkit for Sharing Existing Single-User Applications without Modification, Proceedings of APCHI 98 Asia Pacific Computer Interaction, pp. 183 - 188 (1998).