

パッケージフレーム技術実用化に向けての 問題と解決策

3D-6

三橋 彰彦、小坂 哲也

三菱電機（株）情報技術研究所

1. はじめに

「パッケージフレーム」とは、市販のアプリケーション（以下AP）を部品としてクラス化することにより、異なるAP間でハイパーリンクを用いた業務システムの構築、保守、拡張を容易にする技術[1]である。

「パッケージフレーム」は上記のような仕組みを提供するために、Microsoft® Windows®（以下Windows）環境でAPを制御しているが、この技術を実用化するためには、より多くの市販Windows APを制御できることが重要となる。ここでは、市販Windows APを制御するための問題と実際に行った解決策を述べる。

2. 問題点

APを制御する場合、共通のインタフェース（以下i/f）を制御する側とされる側で実装していなければならない。

Windows環境において、AP同士で（双方向・片方向）通信するためには以下のようなi/fがある。

1. DDE (Dynamic Data Exchange)
2. OLE (Object Linking and Embedding)
3. DLL (Dynamic Link Library) の API (Application Programming Interface) 呼び出し

これらのi/fを利用することにより、APとのデータ送受信や機能を実行することができる。「パッケージフレーム」では、この中でDDEを実装して

いる。したがって、DDEを実装しているAPについては、このi/fにより制御可能である。

しかしながら、APが他のi/fを実装していたり、i/fそのものを実装していない場合は、制御不可能である。

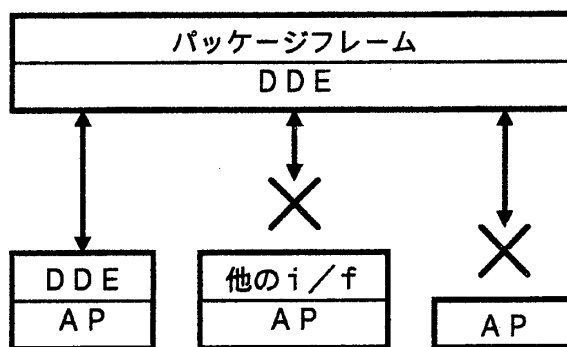


図1：DDE通信のみのAP制御

「パッケージフレーム」がこれらすべてのi/fを実装すれば、i/fを実装しているAPについては制御可能であると言える。しかし、i/fを実装していても、制御される側のAPが提供している機能が少なければ、制御できる範囲も限られてしまう。また、i/fを実装していないAPを制御する方法が一般化されていない。しかも、市場に流通しているAPの多くは、i/fを持たないAPである。

3. 解決策

3.1. i/fの実装方法

DDE以外のi/fとして、「AP制御部」を設けた。この「AP制御部」はi/fの無いAPや、DDE以外のi/fを持つAPの制御を受け持っている。

3.2. i/fのないAPの制御方法

i/fを持たないWindows APを制御するには、次のような方法が考えられる。

1. APの内部で使用しているウィンドウメッセー

Solutions to problems for the PACKAGEFRAME
Akihiko Mitsuhashi, Tetsuya Kosaka
Information Technology R&D Center, Mitsubishi
Electric Corp
5-1-1 Ofuna, Kamakura, Kanagawa 247, Japan

ジをフックして、制御するためのメッセージを送信する。

2. キーボードやマウスの入力をシミュレートすることにより、ユーザがAPを操作するのと同じような方法で制御する。

1の方法を用いれば、キーボードやマウス操作などのユーザの割り込み入力に依存しない制御が可能となる。反面、APのメッセージ仕様を把握しておく必要があるため、汎用的にAPを制御することは困難である。

2の方法を用いれば、汎用的にAPを制御することができる。しかし、制御を行っている過程でダイアログが表示されるなど、ウィンドウの切り替えが発生するので、どのタイミングでどのウィンドウに入力情報を送信するかが問題となる。また、ユーザの割り込み入力に依存してしまうので、結果が予想と異なる可能性がある。

「パッケージフレーム」では、APの汎用的な制御ができなければならないので、2の方法を選択することとした。ユーザの割り込み入力に対しては、一時的にユーザ入力の拒否をすることで回避することができた。ただし、マウス入力のシミュレートは、WindowsのAPがウィンドウ形式であるため、マウスのポジション指定方法が複雑になるので実現しなかった。多くのAPは、キーボードの操作でほとんどの機能を実行することが可能であることから、キーボード入力のシミュレートだけで制御できることになる。

Windows環境では、APにメッセージを送信するには、送信先のウィンドウやダイアログのハンドルが分からなければならないので、制御するウィンドウが変化する場合の処理は複雑になる。しかし、キーボードのシミュレートに関しては、ウィンドウのメッセージキューにキーボード入力のデータ挿入すると、その時アクティブになっているウィンドウに送信される。これにより、制御中に頻繁に切り替わるウィンドウに対して、キーボードのデータを入力することができた。

上記の拡張により、市場に流通しているほとんどのWindows APを制御することが可能となった。

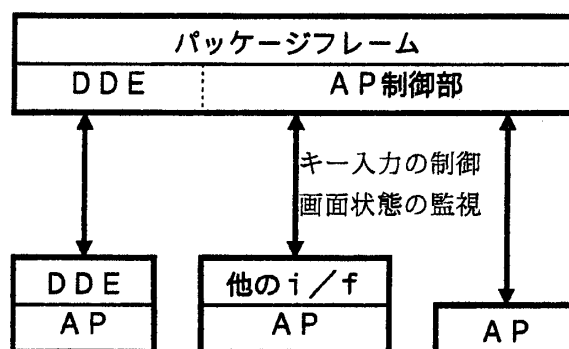


図2. APの制御

4. 課題

現在の仕様で、APが表示しているオブジェクトにアンカー（リンク元）を作成し、ハイパーリンクを実行するには、そのAP自身がアンカーを表示したり、アンカーをマウスでクリックしたときに他の処理を実行する機能を持っていないなければならない。そこで、今後は他のAPのオブジェクトにアンカーを作成する汎用的な方法を検討する予定である。そのためには、以下のような要素技術が必要となるであろう。

- オブジェクトへのアンカー表示
- アンカーに対するマウスの情報取得
- オブジェクトに関連付けされたアンカー情報の保存方法

参考文献

- [1] 小津、土田、佐々木：“既存アプリケーションの機能・情報の統合が可能なシステム構築環境「PACKAGEFRAME」”，電子情報通信学会全国大会,1995,D-160
- [2] 佐々木、阿部、土田：“ハイパーメディア機構を有したアプリケーション統合システム「HyperFrame」”，情報処理学会48回全国大会,1994,5E-4