
Flash を GUI に用いた

VC++によるアプリケーション開発事例

富士ソフト ABC 株式会社

■ 執筆者 Profile ■



森 内 信 公

2003 年 富士ソフト ABC (株)
技術開発業務担当
2004 年 現在 システム事業本部
通信制御部所属



高 橋 卓 央

2003 年 富士ソフト ABC (株)
技術開発業務担当
2004 年 現在 システム事業本部
通信制御部所属



永 谷 隆 宏

2003 年 富士ソフト ABC (株)
技術開発業務担当
2004 年 現在 システム事業本部
通信制御部所属



勝 木 靖 子

2003 年 富士ソフト ABC (株)
技術開発業務担当
2004 年 現在 システム事業本部
通信制御部所属

■ 論文要旨 ■

今日、多くのアプリケーション開発が行われており、ここ数年を見ても、GUI の発達には目まぐるしいものがある。画面デザインは、ユーザーがコンピュータと直接触れる部分であり、これはシステムを作る上で非常に重要で、日々進化を続けている。

優れたアプリケーションを開発するためには、重要な課題がある。それは、ユーザーにとって直感的に利用できる操作性、デザイン性であり、開発者にとってはコストと開発期間の縮小である。

我々は、VC++を用いたアプリケーション開発でこの課題に取り組み考察していく。

この論文をこれからのアプリケーション開発の一例として考えていただければ幸いである。

■ 論文目次 ■

1. はじめに	《 4》
1. 1 当社概要	
1. 2 概要	
2. VC++で開発されるアプリケーションの現状と問題点	《 6》
2. 1 VC++で開発されるアプリケーションの利点	
2. 2 VC++で開発されるアプリケーションの問題点	
3. VC++で開発されるアプリケーションの問題への取り組み	《 6》
3. 1 Flashを導入する利点	
3. 2 Flash導入に際しての注意点	
3. 3 VC++とFlashの開発手法	
3. 4 VC++とFlashの連携手法	
3. 5 VC++とFlashの連携によって生じた問題点と改善点	
3. 6 VC++とFlashの連携による成果	
4. 考察	《 15》
5. おわりに	《 15》

■ 図表一覧 ■

図 1 VC++の開発	《 7》
図 2 VC++とFlashの開発	《 7》
図 3 GUIイメージ	《 8》
図 4 クライアントアプリケーションの処理の流れ	《 8》
図 5 fscommandを用いた処理の流れ	《 9》
図 6 SetVariableを用いた処理の流れ	《 10》
図 7 GetVariableを用いた処理の流れ	《 11》
図 8 VC++で作成した画面	《 14》
図 9 Flashで作成した画面	《 14》
表 1 fscommandの説明	《 9》
表 2 SetVariableの説明	《 10》
表 3 GetVariableの説明	《 11》

1. はじめに

1. 1 当社概要

富士ソフト ABC 株式会社は 1970 年（昭和 45 年）5 月に設立され、以来「国内最大の独立系 SE 集団」として、ひのき（「品質」、「納期」、「機密保持」）を開発ポリシーに、特定のプラットフォームにとらわれることなく、お客様に最適な環境とは何かを考えている。大規模エンタープライズシステム構築においては、業務分析と IT コンサルティングの段階から、お客様の視点に立った柔軟な発想で最適なソリューションを提案している。高度な知識と豊富な経験に基づき、お客様を取り巻くシステム環境をあらゆる角度から精査しながら、要件定義・システム設計、プログラミングからテスト・納品まで、一貫した開発理念が高品質・高信頼性を提供している。保守、ヘルプデスク、教育といったシステム運用に至るまで、IT に関わるすべての技術サービスをお届けする「総合システムビルダー」として IT のベストパートナーとなる企業である。

1970 年 5 月 会社設立
1992 年 10 月 東証二部上場
1995 年 6 月 ISO9001 認証取得
1998 年 8 月 ISO14001 認証取得
1998 年 9 月 東証一部上場
2002 年 5 月 プライバシーマーク認証取得

詳しくは当社ホームページ（URL <http://www.fsi.co.jp/>）を参照。

1. 2 概要

今日、様々なアプリケーション開発が行われている。そして、アプリケーションを使用するユーザー層に目を向けると、システムのハイテク化が進み小学校の授業でもコンピュータが導入され裾野が広がっている。また高齢化が進んでいることもあり、アプリケーションを利用するユーザー層は、以前に比べると格段に幅広くなっている。ユーザー層の拡大に伴い、使いやすさが追求され、グラフィカルユーザーインタフェース（以下 GUI という）が目まぐるしく発達している。例えば、Windows XP では「LUNA」と呼ばれる全く新しい GUI を採用することで、従来より作業を快適化し、その進化は立証済みである。

画面デザインと GUI は、ユーザーがコンピュータと直接触れる部分である。これはシステムを作る上で非常に重要なものであり、その技術は日々進化を続けている。

アプリケーション開発における代表的な開発環境の一つに Visual C++（*1）（以下 VC++ という）が挙げられる。VC++ は開発用のライブラリ、ツールが充実し、複雑な内部処理の実装に長けている。しかし、アニメーションといった GUI 系処理の実装を考えた場合、多くの時間を要する上に、デザイン性に優れているものを作れるとは言い難い。

アプリケーション開発には、ユーザーにとって直感的に利用できる操作性と、高いデザイン性が求められる。また、開発者にとってはコストと開発期間の縮小が重要な課題である。これらの課題を克服するためには、どうしたらよいのだろうか。

この論文ではオンラインサービスを提供するアプリケーションの開発事例とともに、上記の課題に対する開発手法を紹介する。

2. VC++で開発されるアプリケーションの現状と問題点

我々が開発したオンラインサービスは、サーバからユーザー端末（PC）へ情報を提供するサービスである。本論文では、ユーザー端末となるクライアントアプリケーションについて取り上げる。なお、クライアントアプリケーションは Web ブラウザに依存しない独立したアプリケーションであり、開発では主に VC++を利用する。

2. 1 VC++で開発されるアプリケーションの利点

VC++はオブジェクト指向開発ツールや Microsoft Foundation Class (*2) (以下 MFC という) といったクラスライブラリにより、優れた生産性を提供する。MFC は様々な処理を抽象化しており、複雑なロジックを手軽に実現することができる。

また、VC++の特徴である多重定義、動的メモリ取得・開放、例外処理によって、高機能かつ安定した内部処理を実装することができる。

2. 2 VC++で開発されるアプリケーションの問題点

クライアントアプリケーションで提供されるサービスの内容から、アプリケーションの GUI には優れたデザイン性が求められる。また不特定多数のユーザーが利用するため、直感的に理解できる操作性も求められる。

これらを考慮するにあたり、以下の三つの問題点が挙げられる。

(1) GUI のデザイン性の問題

VC++で標準コントロールを用いて GUI を作成すると、ボタンやウィンドウなどは Windows 準拠のため、新しさやユニーク性に欠けてしまう。加えて、フリースタイルでアニメーション機能を持つといったデザイン性の高いボタンを実装するには、新たに設計やコーディングを行う必要があり、開発のために膨大な時間を要する。

(2) アプリケーションの操作性の問題

VC++で GUI を作成すると、ボタンやウィンドウは Windows 準拠の GUI 部品が存在する。

しかし、多種多様なシステムへの対応や幅広い年齢のユーザーが操作することを考慮すると、限られた部品だけでは直感的に操作が出来ないという問題がある。

(3) 開発手法としての問題

画面デザインは、利用者及び技術面からも変更要求の頻度が高い部分である。画面部分の変更が原因で、複数の担当者が同一のソースに対してメンテナンスを行うことが多くなる。また、他の処理にも影響が発生するため、品質の低下や対応の遅れの原因となり、開発手法として問題である。

3. VC++で開発されるアプリケーションの問題への取り組み

3. 1 Flashを導入する利点

VC++で開発されるアプリケーションの問題点を改善するためにクライアントアプリケーションではGUI部分にFlash(*3)を導入し、VC++とFlashの連携による開発を行った。

現在、Flashは主にWebコンテンツを中心に利用されており、これまでのGUIの操作性や表現力を圧倒するものである。しかしブラウザに依存しない独立したアプリケーションへの利用はまだ少なく、Flashを利用することによるメリットは魅力的である。そこで我々の開発に際し、ブラウザに依存しないクライアントアプリケーションに次のようなFlashの特徴を利用した。

(1) インパクトのある表現が可能

Flashは動画や音声の挿入、そしてFlashのモーショントゥーイン機能(*4)やシェイプトゥーイン機能(*5)を用いることにより、ユーザーによるマウスやキーボードの入力に対してボタンが飛び出すなどのアクションを、容易かつシステムの負荷を抑えて実現する。これにより従来までの静的なGUIとは一線を画すインパクトのある表現が可能である。

(2) ユニークなGUI部品を作成し操作性を向上させることが可能

Flashはアプリケーションごとに開発者達が独自にGUI部品を開発することができる。

例えば、Flashのボタンシンボル機能を用いることにより、ボタンの点滅表示や拡大縮小、更にはボタンの形さえも自由に作り上げてユーザーが直感的に操作しやすい画面を開発することが可能である。

(3) 機能ごとの独立

GUIの部分をFlashで開発することにより、画面側と制御側の分離が明確になる。

これにより、それぞれの独立性が確保され、開発においては分業が容易になり、設計、実装、デバッグが効率的に行える。

3. 2 Flash導入に際しての注意点

VC++とFlashの連携による開発を進めていく際に、以下の点について注意が必要である。

(1) VC++とFlashの連携に前例がない

クライアントアプリケーションではVC++とFlashの連携を試みるが、我々の身近ではこのような開発は行われておらず参考のできる開発事例がなかった。そのため、開発を進めていく上でどのような技術的問題点が発生するか、またそれに対してどのように解決したらよいかなど不明な点が多かった。

(2) 不要なアニメーションの乱用

Flashの特徴としてアニメーションなどの複雑な表現が可能である。インパクトの

ある表現が可能となる反面、乱用すると処理速度の低下や操作性の悪化の原因となる。そのため効果的に利用するかが重要である。

(3) 実行ファイルの統合による動作の負担

Flash の特徴としてファイルサイズの軽減が挙げられるが、画面ごとの実行ファイルを一つの実行ファイルにまとめると、結局はファイルサイズが拡大してしまい、読み込みに時間がかかってしまう。したがって細かく画面ごとに実行ファイルを分割し、必要な場合に必要な実行ファイルのみを読み込むことが重要である。

3. 3 VC++とFlashの開発手法

下図はVC++における開発とFlashとの連携を行なった開発の構成比較である。

現在、VC++のアプリケーション開発では、画面側、制御側共にVC++を用いて開発が行われているのが主流である(図1を参照)。今回のクライアントアプリケーションでは、画面側(Flash)と制御側(VC++)を独立させて、生産性の向上を図った(図2を参照)。

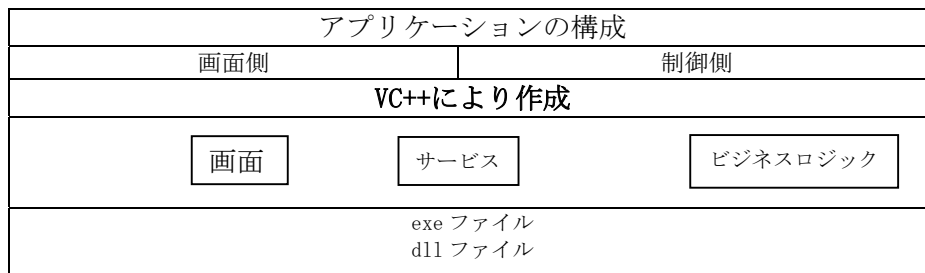


図1 VC++の開発

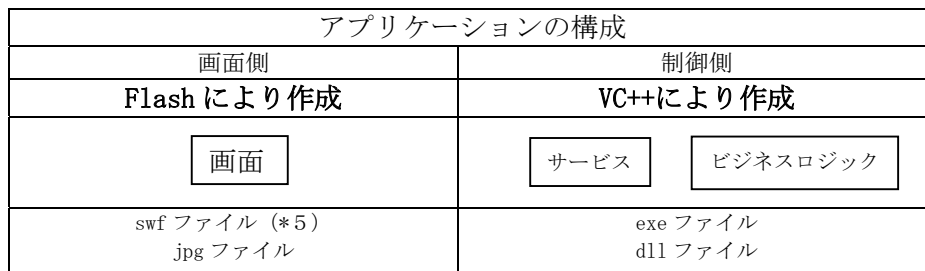


図2 VC++とFlashの開発

3.4 VC++とFlashの連携手法

Flash で作成された画面側と VC++で作成された制御側との連携手法について記述する。クライアントアプリケーションでは、VC++のダイアログの上に Flash で作成した画面を貼り付けている（図3を参照）。また、VC++とFlash が互いの処理結果を設定、及び取得することにより連携を図っている（図4を参照）。

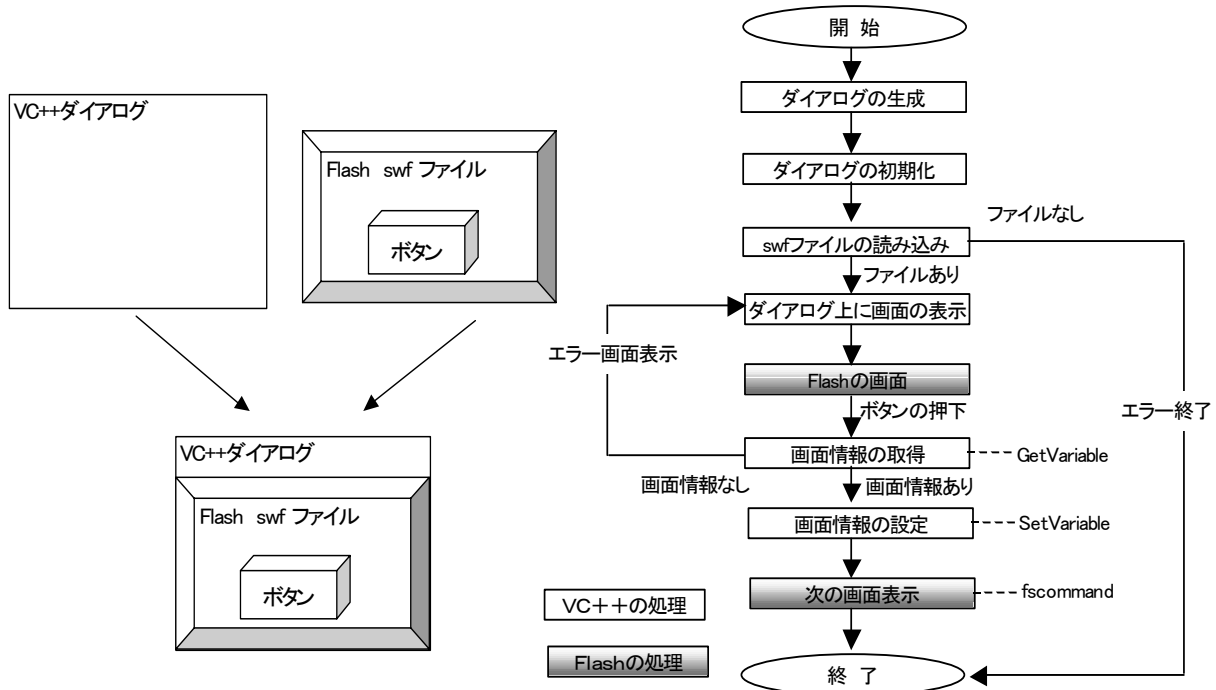


図3 GUI イメージ

図4 クライアントアプリケーションの処理の流れ

クライアントアプリケーションでは、VC++とFlash間でデータの受け渡しをする。Flash側では、ActionScript (*7) の fscommand メソッドを画面表示（画面遷移）に用いる。VC++側では、CShockwaveFlash クラスの SetVariable, GetVariable メソッドを画面情報設定と画面情報取得に用いる。次にクライアントアプリケーションで用いたメソッドの使い方について詳しく説明する。

(1) fscommand の使い方

画面上にあるボタンが押下された時の処理として、ActionScript の fscommand メソッドを用いる。Flash は、パラメータとして画面名とボタン番号を VC++ に渡す。VC++ は渡されたパラメータをもとに画面遷移やデータ取得、終了といった処理を実行する。VC++ がパラメータを取得した旨を伝えるために、処理同期フラグを Flash に渡す（**図 5**を参照）。fscommand メソッドの引数の説明を**表 1**に記す。

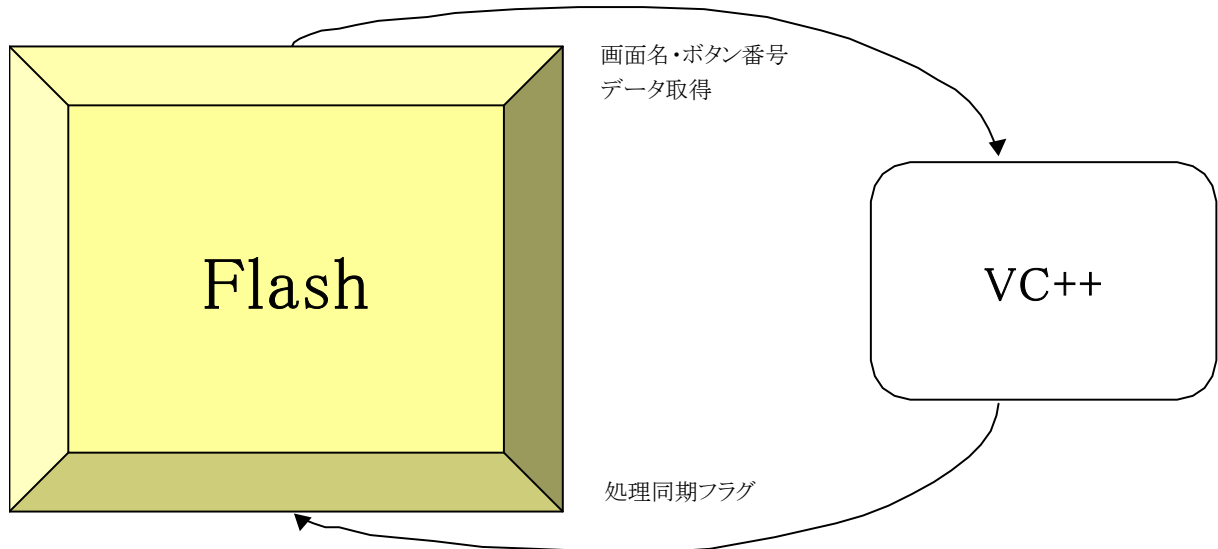


図 5 fscommand を用いた処理の流れ

表 1 fscommand の説明

メソッド名	No.	型	説明
fscommand	1	String	画面名
	2	String	ボタン番号

(2) SetVariable の使い方

VC++が Flash のデータを設定する時の処理として、SetVariable メソッドを用いる。VC++が画面に表示させたい画面名とボタン番号をパラメータとして Flash に渡す。Flash はパラメータを取得し、画面のタイトルやボタン番号を設定する（図6を参照）。その他に、画面数や画面に表示する情報を設定している。SetVariable メソッドの引数の説明を表2に記す。

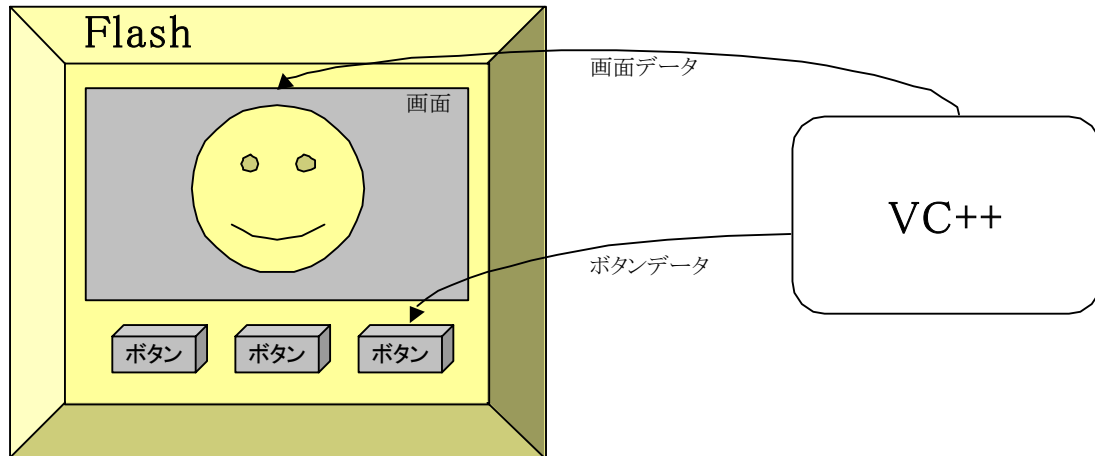


図6 SetVariable を用いた処理の流れ

表2 SetVariable の説明

メソッド名	No.	型	説明
SetVariable	1	String	画面名またはFlash内での変数名
	2	String	ボタン番号または設定値

(3) GetVariable の使い方

VC++が Flash のデータを取得する時の処理として、GetVariable メソッドを用いる。ボタンを押下時に VC++が Flash の画面データ（ユーザーが入力した項目）や画面のタイトルなどの画面に表示されているすべてのデータを取得する時に用いる（**図7**を参照）。f_scommnad とは違い、画面遷移の場合ではなく、画面内の情報を取得する場合に用いる。GetVariable メソッドの引数の説明を**表3**に記す。

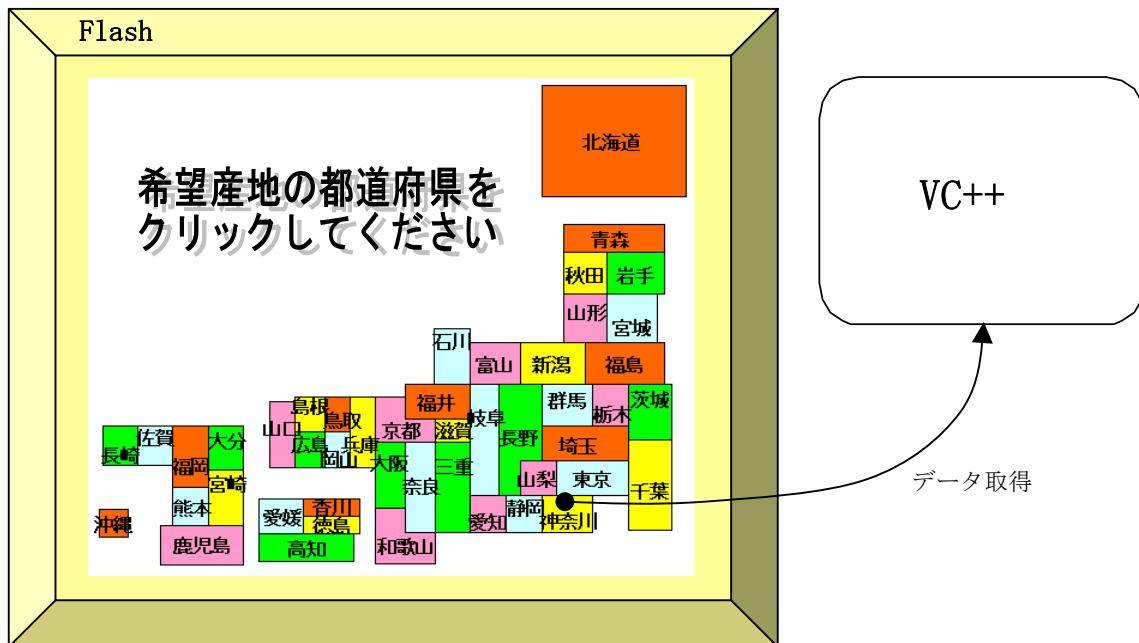


図7 GetVariable を用いた処理の流れ

表3 GetVariable の説明

メソッド名	No.	型	説明
GetVariable	1	String	データ変数名

3. 5 VC++とFlashの連携によって生じた問題点と改善点

クライアントアプリケーションでは VC++と Flash を連携することにより、VC++単独の開発では発生しなかった問題が生じた。

(1) データ受け渡しに関する問題点と改善点

VC++と Flash がデータをやり取りする場合、処理の流れのタイミングが異なるために問題が生じた。例として、VC++から任意のデータを Flash 側が受け取り、その受け取ったデータを画面に表示するとした場合を挙げる。この処理は「VC++が任意のデータを Flash に渡し、Flash は VC++からデータを受け取り、Flash は受け取ったデータを画面に表示する。」というものである。しかし、実際に VC++と Flash を連携し、動作確認を行ったが、予期せぬ動きとなった。これは、VC++が Flash にデータを渡す時に Flash がデータを受け取らずに処理を実行させてしまうということである。この問題を調査するために、ActionScript 内にログを埋め込み、Flash が VC++からデータを取得するタイミングを検証した。すると、VC++がデータを渡すタイミングは期待通りだったが、Flash のデータを受け取るタイミングが、想定したタイミングと異なっていることが判明した。その結果より、VC++と Flash は独立したタイムラインで処理を行っていると推測した。この問題を解決するために双方の同期を取る必要がある。そこで VC++から Flash へデータとともに同期を取るための通知フラグを渡し、Flash は通知フラグを取得後に処理を実行するようにした。この方法ならば、勝手に処理が進むこともなく、確かなデータの受け渡しが可能となり、問題を解決することができた。このように双方の同期を取ることが VC++と Flash を連携させる開発において最も考慮すべき点である。

(2) ActionScript に型宣言の制約がない問題と改善点

型宣言の制約のない ActionScript において VC++から取得したデータをそのまま利用すると、誤動作する問題が生じた。これは VC++から Flash へデータを数字の文字列として渡した場合に起きた問題である。つまり、ActionScript はプラス記号 (+) を用いて文字列連結を行うため、VC++から取得したデータを文字列と解釈し、和演算ではなく文字列連結を行っていた。

この問題の解決策としては、C 言語や Java 言語のように型宣言を行うのが一番よいのだが、先に述べたように ActionScript には特に型宣言の制約がない。このために、VC++から文字列として送られてきたデータを Number 関数でデータに変換後、和演算を行うことによりこの問題を解決することができた。

3. 6 VC++とFlashの連携による成果

(1) GUIのデザイン性

従来からの VC++により作成された画面は、現在多くのアプリケーションで Windows 準拠のボタンやウィンドウのオブジェクトを利用しているため斬新なデザイン性というものはあまり感じられない。

一方、クライアントアプリケーションで Flash により作成した画面は、VC++の画面と比べると斬新なデザインや独自に作成したボタンを配置している。両者を比較すると Flash で作成した画面の方が独創性が高く、クライアントアプリケーションに相応しいといえる。

(2) 操作性

VC++を用いた例として図8と、Flash を用いた例として図9の操作性を比較する。図8は、ボタンを押下するとボタンがへこんだような単調な動きしか表現できず、どのボタンを選択して押下したのかわかりにくい。対して、図9はマウスによりフォーカスが当たった部分を前面に強調して表示し、ボタンを押下するとそれに伴って画面が拡大されるといった様々な特殊効果的アクションを組み込むことができ、ユーザーの操作に多彩な演出が加わる。これなら、今までパソコンを使ってきたユーザー達の興味を引くことができ、またパソコンを扱うことに苦手意識のあるユーザーも直感的に操作することができるだろう。

(3) 開発手法

クライアントアプリケーションにおいて画面側 (Flash) と制御側 (VC++) にコンポーネントを独立することで、コンポーネント間の依存性が最小限に抑えられ、設計や製造、大幅な仕様変更が発生した場合も比較的柔軟な対応が可能となり、作業効率の向上に繋がった。Flash は、VC++よりも短時間で画面を作成することができる。また、製造後のメンテナンスも各機能の開発者が専念すればよいので保守性が確保できた。

従来の VC++単独の手法では画面の仕様変更が発生した場合、全ソースのコンパイルが必須であり、多くの時間と労力を費やしていた。本開発の手法では、画面のコンポーネントを独立させているので、Flash で作成された画面の実行ファイルのみを差し替えることで、画面変更を行い、制御側のコンポーネントへの影響を最小限に抑え品質の低下や対応の遅れを防げるようになった。



図8 VC++で作成した画面

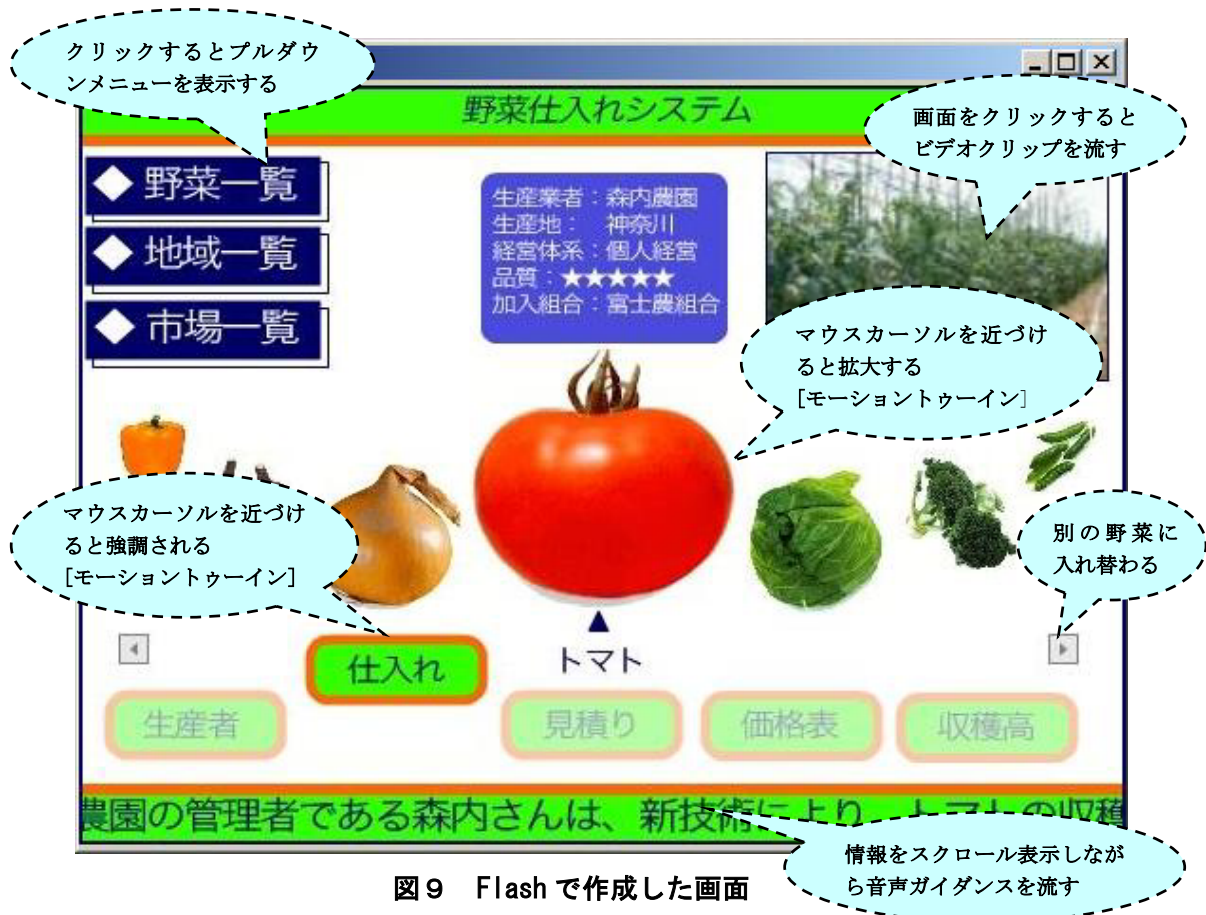


図9 Flashで作成した画面

4. 考察

クライアントアプリケーションで VC++ と Flash の連携手法を進めるにあたり、前例がなく発展途上の段階のため、いくつもの問題に直面し、その調査に多大なる労力を費やした。連携をさせることで従来までの問題を解決できるメリットを我々もわかっていたが、連携することによる特有の問題にも遭遇した。しかし、開発したアプリケーションはデザイン性や操作性が向上し、コンポーネントを独立させることで作業効率の向上や保守性を確保することが可能となった。

以上より、異なる開発環境を連携することによって、今までにない可能性を持つ新たなアプリケーションの開発と作業効率の向上を実現できる結果は明らかである。

今回は VC++ と Flash という開発環境を選択したが、今回のクライアントアプリケーションで利用できた Flash 特有の機能はまだまだ少ないと思われる。今後の課題として、動画や音声の挿入やトゥーイン機能をはじめとした様々な機能を効果的に用いることにより、VC++ と Flash という異なる開発環境のメリットを取り入れたインタラクティブなアプリケーションの開発を進めていきたい。

5. おわりに

GUI は、ユーザーがコンピュータと直接触れる部分であるため、デザイン性の向上、操作性の容易さが今後も追及され、機能の充実にとどまることはないだろう。

開発者は、これらの問題に立ち向かうためにも、より充実した開発環境を構築し、かつ設計から製造後の保守面でも柔軟な対応ができるようにならなければならない。

今回は VC++ と Flash の連携だったが、開発環境の組合せは様々である。

今後、ユーザーの視点に立ったアプリケーションの開発を進める上で、この論文がこれからのアプリケーションの一例として考えていただければ幸いである。

[注釈]

- *1 Microsoft, Windows, Visual C++ は、米国Microsoft Corporationの米国及びその他の国における登録商標または商標である。
- *2 MFC (Microsoft Foundation Class) とは、米国Microsoftが開発したWindows用アプリのプログラミングを効率的に行うためのC++用のクラスライブラリである。
- *3 Flashは、米国Macromedia, Inc. の商標である。
- *4 モーショントゥーインは、ユーザ側で初期フレームと終了フレームを作成することによりFlashが中間フレームの内容を自動作成する。例：ボタンのズームインやボタンの移動
- *5 シェイプトゥーインは、モーショントゥーインと同様に初期フレームと終了フレームの間を自動作成するが、一つの形から徐々に別の形へとモーフィングの効果を實現できる。例：ボタンの変形
- *6 swf (small Web format) とは、Flash Playerで表示する外部ムービーファイルである。swfファイルを読み込む場合は、URL (相対パス, 絶対パス) を指定する必要がある。
- *7 Macromedia社のWeb向けアニメーションツール「Macromedia Flash」に内蔵されている、コンテンツの再生制御などを行うためのスクリプト言語である。