

# ActiveX と COM

## ActiveX and COM

川崎 高志, 池田 大樹 (知的システムデザイン研究室)

Takashi KAWASAKI, Taiju IKEDA (Intelligent Systems Design Laboratory)

**Abstract** ActiveX is one of the most important technology of Microsoft that based on Component Object Model(COM), Distributed COM. It tremendously accelerates development of distributed computing applications and also make it easier to reuse or/and extend the pre-developed components. Now, we don't have to reinvent the wheels.

## 1 はじめに

近年のコンピュータの処理能力向上にはめざましいものがあり, それと並行してアプリケーションの規模も大規模化の傾向がある. また, アプリケーションの開発周期もも自覚なってきたため, アプリケーション開発の分野では, より直感的で簡単な開発というものが重視されるようになってきた. ActiveX とはその中で生まれたコンポーネント指向プログラミングへの Microsoft のソリューションである. この論文では, コンポーネント指向プログラミングの概要, ActiveX のベースである COM について述べ, 今後の展望に関しても言及する.

## 2 オブジェクト指向からコンポーネント指向へ

コンポーネント指向プログラミングとは, オブジェクト指向プログラミングから派生した新しいスタイルのプログラミングである. Windows 上の Visual Basic や, Delphi に代表される, このスタイルのプログラミング環境では, GUI を駆使し, オブジェクト指向の技術をより親しみやすいものとしているため, デベロッパは枝葉の部分に気を取られることなく, 根幹技術の開発に専念することができる.

ActiveX とは, 主に Windows で使われる, このコンポーネント指向プログラミングでのコンポーネントのことであり, それらは, COM(Component Object Model) や, OLE<sup>1</sup>といわれる技術をベースとしている.

<sup>1</sup>元は Object Linking and Embedding:オブジェクトのリンクと埋め込みに関する一般技術の総称であったが, 後に COM をベースとするいくつかのテクノロジーの総称として単に OLE となった.

## 3 COM

ActiveX の説明の中でしばしば登場する COM とは, Microsoft が提唱するオブジェクトモデルである. COM は, Windows 上でのバイナリ標準であり, 最近の Windows アプリケーションは当然のように COM ベースで実装されている. その背景には, その再利用可能性が非常に高いこと, そして, ほとんどの適応例において十分なパフォーマンスを確保できているということがある. 例えば, Internet Explorer の内部で実装されている DHTML という HTML レンダリングエンジンは COM コンポーネントとして実装されているため, Internet Explorer の一部として動作することはもちろん, Visual Basic, WSH<sup>2</sup>, その他の COM に対応するあらゆる言語から, そのコンポーネントをあたかも自分の一部であるかのように実行することができる.

### 3.1 堅牢なバージョン管理

COM は, 複数のインターフェイスという方法を用いて, それまで Windows で使われていた共有ランタイムライブラリ (DLL) の欠点であるバージョン管理の難しさを改善した. COM では, 新しい機能を追加する場合, 従来方式のようにそれまでのインターフェイスを変更したりはせず, 代わりに新しいインターフェイスを追加する. 従って, 古いインターフェイスに依存していたアプリケーションが動作しなくなるようなことはなく, また, 新しいアプリケーションは新しいインターフェイスを通じて新しい機能を活用することができる. また, インター

<sup>2</sup>Windows Scripting Host: Windows での標準スクリプティングエンジンインターフェイス. VBScript, JScript, ActivePerl などのさまざまなスクリプト環境がこの上で実装されている.

フェイスは、GUID(Global Unique Identifier)<sup>3</sup>といわれる 128 ビットの識別子によって世界中で一意に区別できる。そのため、ファイル名でライブラリを管理する場合のように、個々のコンポーネントのインターフェイス識別子 (通常は名前) が衝突し問題を生じるといったことはない。

### 3.2 位置透過性

COM の特徴として、インターフェイスと実装を分離することにより位置透過性を確保したことがある。それまでの Windows の DLL はファイル名のみによって位置を特定していたため、ディレクトリ構造に変化があった場合ライブラリを見つけれなかったことがあったが、COM ではそのようなことはなくなった。さらに、DCOM(Distributed COM) では、RPC<sup>4</sup>によってその概念を拡張し、位置透過性をネットワーク上にまで拡大した。COM の位置透過性によって、ユーザーはそのコンポーネントがどこに存在するのか、どこで実行されるのかをほとんど意識する必要がなくなる。

### 3.3 オブジェクト指向

COM はコンポーネント機能を公開するためにオブジェクト指向的な要素を取り入れている。これまでの Windows の DLL や、モジュール間通信 (DDE) では、現在 C++ プログラムが日常的に使用している典型的なオブジェクト指向要素が採用されていなかった。COM はカプセル化、継承<sup>5</sup>、ポリモーフィズムといった 3 つのオブジェクト指向要素を、言語に依存することなく取り入れている。

### 3.4 開発言語からの独立

前述のとおり、COM はオブジェクト指向要素を言語非依存を実現しながら取り入れているが、C++ だけではなく、Visual Basic や Java といった C++ 以外のオブジェクト指向言語でも COM コンポーネントを作成することができる。また、この違いは COM 自身に吸収されるため、利用する側で問題となることはない。

<sup>3</sup>INTERNET-DRAFT "UIDs and GUIDs"  
ftp://ftp.isi.edu/internet-drafts/draft-leach-uuids-guids-01.txt

<sup>4</sup>Remote Procedure Call : 分散コンピューティング環境 (DCE) 標準の他のマシン上でプログラム実行を制御する技術。

<sup>5</sup>実際には、COM はインターフェイスレベルでの継承しかサポートしていないという非難もある。

## 4 分散オブジェクトとしての一面

COM には位置透過性でも述べたように分散オブジェクトとしての側面がある。この場合、ネットワーク上でのコンポーネント間の通信を取り持つのではあるが、この分野でいえば、COM よりも、OMG が提唱、推進するオープンな規格 CORBA(Common Object Request Broker Architecture) の方が注目されており、Windows 以外の世界ではむしろこちらの方が有望視されている。

その理由は、基本的に ActiveX が Microsoft の閉鎖的なテクノロジーであり、CORBA は逆に完全にオープンであるということであろう。メーカーにとって、特定の技術を実装する上で移植性や可搬性を考慮することは当然のことであり、一社独占状態のテクノロジー上での実装を行うことにはやはり抵抗がある。従って、この分野では、ActiveX よりも CORBA の方が有利になるであろう。<sup>6</sup>

## 5 まとめ

技術的に見ると本来、コンポーネント指向は COM の利点でもあり、欠点でもある。簡単にいえば、オブジェクト指向から継承などの難しい概念を省き、簡略化したものであるため、継承などの強力なオブジェクト指向技術は使うことができないのである。これは、Microsoft によれば実装レベルでの継承は悪しき風習であり、得る物はないためとされている。確かに、この理論はある意味では正しく、COM は修得の難しいオブジェクト指向の概念の一部を省略し、コンポーネント指向プログラミングを全面に出したことによって、Windows の開発者人口を増やしたという意味では、ある程度の評価を得ることはできる。今後、この ActiveX がどのように発展していくかが Windows が有益なアプリケーションのプラットフォームになり得るか、そして、Windows の行く末を決定するであろう。

## 参考文献

- [1] Tom Armstrong 著, 豊田 孝 監訳, ATL プログラミング, SOFTBANK, 1999
- [2] 豊田 孝, COM アーキテクチャと実践スクリプティング, 翔泳社, 1998

<sup>6</sup>Windows 上での有力な CORBA プラットホームとしては、Inprise 社の Delphi, C++Builder などがある。