

.NET

ってなに？

と聞かれても、もう慌てない

北澤 誠 KITAZAWA, Makoto

はじめに

「.NET」という言葉が世に出てから、もう2年が経ちました……。

2000年の6月、主に各報道機関向けに米国にて開催されたマイクロソフト社のForum 2000において、Microsoft .NETという新たな全体構想が発表されました。Microsoft .NET自体は大変広範な構想を指すものであり、それらの技術の中核は、当初、NGWS (Next Generation Windows Services) と名付けられる予定で開発されていました。それ以後、.NET Frameworkと名付けられたこの技術は、現在ではすでにそのすべてが利用できるようになっています。もちろん、最も要である、開発ツールVisual Studio .NETも、2002年3月に日本語版が無事リリースされており、今すぐにも.NET Frameworkを利用するソフトウェアの開発を始められる状態にあります。

ソフトウェア開発の進化

ご存知、Bill Gates氏も、「DOSからWindowsの変化に匹敵する」と表現したように、.NET Frameworkを用いた開発は、従来からのWindowsソフトウェア開発を大きく進化させるものだと思います。そして、マイクロソフト社自身も、その変化にユーザーが対応できるように非常に早い時期から日本においても、.NET Framework上で動作するソフトウェアを開発するための開発環境「Visual Studio.NET」のベータプログラム配布を実施し、さまざまな技術資料やセミナーなどを提供していたように思います。

しかしながら、本誌をご覧になっているみなさん、そしてみなさんの身のまわりにいる方々は、.NET Frameworkについて、どこまで理解されているでしょうか？

ぜひ近くにいる方に、「.NET Frameworkとはなんですか？」という質問を

投げかけてみてください。さまざまな答えが返ってくると思います。恐らく、その中には「インターネットを使わなければ関係ないよ」とか、「XML Webサービスを使わなければ関係ない」といった回答もあるのではないのでしょうか？でも、それらはまったくの誤りです。ただし、残念ながら、このような状況が市場全体の大勢であると言わざるを得ないのかもしれないかもしれません。少なくとも、.NET Frameworkのメリットがきちんと理解されているわけではなく、そのために誤解が生じていることも確かでしょう。

筆者が見る限り、.NET Frameworkが提供する新しい世界は、開発に携わる方から、最終的には導入したすべての企業やユーザーに多くのメリットを提供できるだけのポテンシャルを兼ね備えています。もちろん、開発に携わる方々には、新しい開発プラットフォームとして、相応の勉強が必要となることは否めません。しかし、それら努力を上回るメリットを享受できることも

また事実とも言えます。

本稿では、本誌がdotNETマガジンという新しい名前を授かったという区切り、まさに“始まり”の意味も込め、今まであまり簡潔な言葉で語られることのなかった、.NETすなわち.NET Frameworkを利用することのメリットを本誌の読者すべての方に理解していただけるように整理してゆきます。

なぜ、.NET Frameworkは登場したのか?

.NET Frameworkが登場した理由は、さまざまな側面から考えることができます。政治的側面から見れば確かに「Javaの脅威」といったこともあるのかもしれませんが、それらは記事のネタとしては面白いかもしれませんが、利用する側にはまったく関係のないことです。

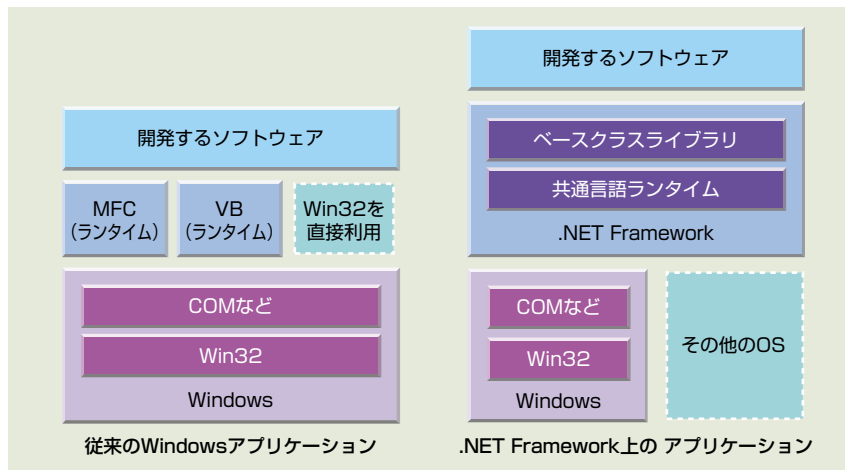
そこで、ここでは純粋に提供されている機能から.NET Frameworkの特徴を考えてみます。特徴としては、以下のような5つの点を挙げるができます。

.NET Frameworkの特徴

- ・ 強固なセキュリティ機構
- ・ システム全体の安定性の向上
- ・ プログラミングモデルの統一
- ・ 開発生産性の向上
- ・ 簡便な展開／導入

上記を見るとみなさんが日々、開発の現場で求められていることもあると思います。セキュリティや、安定性に

図1：開発環境のこれまでとこれから



についてはインターネットが登場してから言われて続けていることであり、とくに目新しいことはないように感じます。

今までの開発環境と .NET Frameworkの違い

では、一体、今までの開発環境と.NET Frameworkは何が異なるのでしょうか？ たとえば、従来の方法では、セキュリティも安定性も開発者のスキルに任される部分がとても大きくなっていました。セキュリティのさまざまな仕組みを組み込むのも、プログラム自身の安定性を確保するのも、大部分は設計者も含めた開発担当者の腕にかかっていました。ある意味、「開発者は高いスキルがあり、絶対ミスを犯さない」という性善説に基づいていたのです。

それに対して、.NET Frameworkの設計思想は若干異なっているようです。さまざまなセキュリティの仕組みをシステムレベルでもち、開発者はその機能を利用してより簡単に強固なセキュリティを実現することができます。ま

た、メモリ管理をはじめとした機能の強化によって、実質的なプログラム上のミスをできる限り減らし、プログラムの安定性を増すことができます。

従来であれば、非常に高いスキルをもった開発者でなければ実現できなかったさまざまなセキュリティ機能の利用や、安定度の高いプログラムの作成を、.NET Framework側の機能で助けてくれる分、「さまざまな層の開発者がセキュリティや安定性を実現しやすいように考えられている」と考えるとわかりやすいと思います (図1)。

また、大きな特徴として「開発言語／手法の統一」があります。既存の開発環境では、Windowsアプリケーションを作るとき、Webアプリケーションを作るときでは、使う開発言語も開発手法も異なりました。しかし、.NET Frameworkの環境下では、これらの統一を実現しています。

要するに、Windowsアプリケーションも、Webアプリケーションも、そしてXML Webサービスも、すべて同じ開発言語、同じような開発手法で作成