

必ずできる!

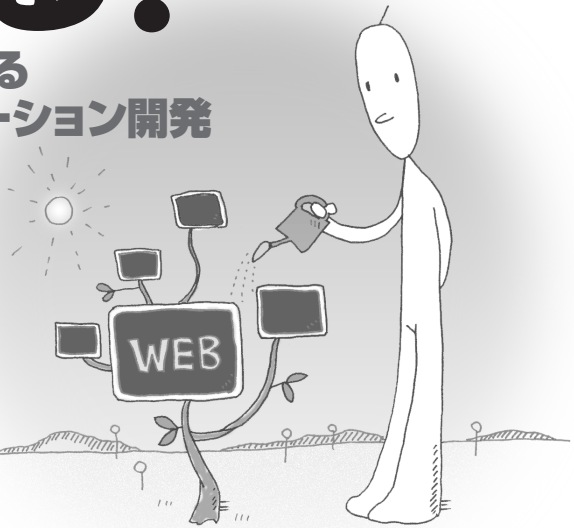
基礎固め ゼロからはじめる
Webアプリケーション開発

ASP.NET

新連載

ASP.NETの基本を学ぶ
～概要から環境設定まで

山田 祥寛 YAMADA, Yoshihiro
<http://www.wings.msn.to/>



Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2002
- ASP.NET
- Internet Information Services
- Other:

Level



Samples

ASP.NETってなんだろう

今月から12回にわたって、ASP.NET + Visual Studio .NETによるWebアプリケーションの開発について一緒に勉強してゆきましょう。ASP.NETにせよ Visual Studio .NET (以下VS.NET) にせよ、さまざまな機能を提供する膨大な技術ではありますが、恐れることはありません。

重要なことは基本的な考え方の理解です。細かい構文や機能を記憶することに振り回されないでください。「そんなもの」は基本概念を理解してさえいれば、必要な時に必要に応じてドキュメントを調べればそれで良いのです。この連載では、そのドキュメントを「自分で調べる」ための、最初の取っかかりの部分を知っていただければと思っています。

まず具体的な環境構築、開発に入るのに先立って、「ASP.NETとはなにか?」——どういう技術なのかという点について、簡単におさえておくこと

にしましょう。

※ サーバースайдで動作する

ASP.NETは、クライアント (多くがブラウザ) からのリクエストに応じて、Webサーバー上で処理を行ない、その処理結果を (一般的には) HTMLとしてクライアントに回答する技術です (図1)。「技術」ということは、ASP.NET自体が使用する言語を規定するわけではありません。Visual Basic .NETやC#、JScript .NETなどなど、ASP.NETに対応したプログラム言語ならば、なんでも自分の好みに応じて自由に使用することができます。

ASP.NETとは、Webアプリケーションを作成するための「便利な部品の集まり」であり、「基本的なルールを規定したもの」であると思っていただければよいでしょう。

※ イベントドリブンモデル

実は、このようなサーバースайд処理の仕組みそのものは決してASP.NET特有のものというわけでも、目新しい

必ずできる!

基礎固め ゼロからはじめる
Webアプリケーション開発

ASP.NET



図1：ASP.NETはサーバーサイド技術

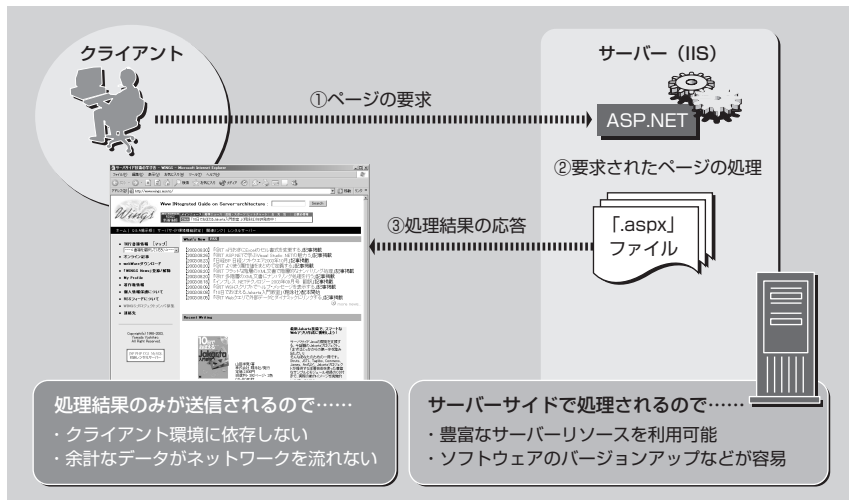
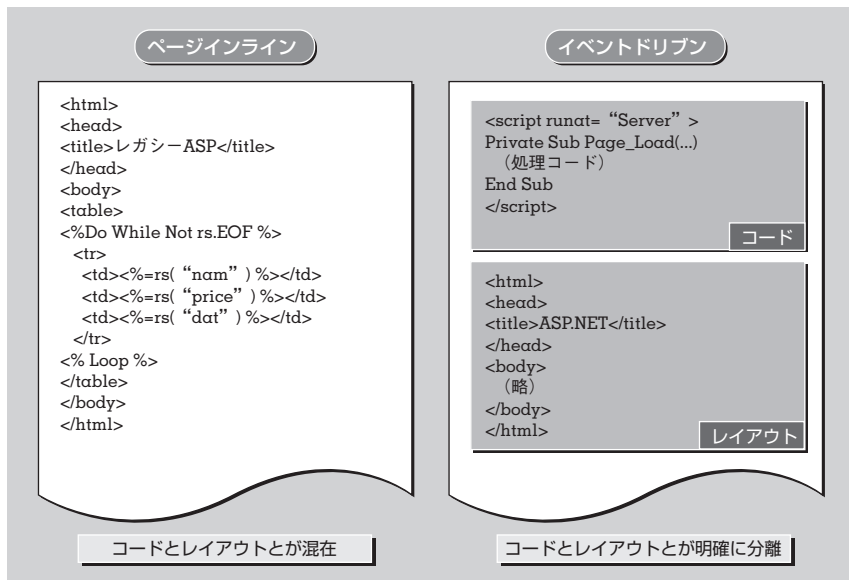


図2：イベントドリブンとページインライン



ものというわけでもありません。ほかにもJava言語をベースとしたJSP (Java Server Pages) & サーブレットや、関数(マクロ)を主体とした簡易なスクリプト環境PHP (PHP: Hypertext Preprocessor) などが代表的です。

ASP.NETがこれらの技術と異なる点は、開発モデルとして「イベントドリブン(駆動型)」を採用しているという

点です。

イベントドリブンがなにかという点に話を移す前に、従来のJSP&サーブレットやPHPで採用されている開発モデルについて、少々お話ししておく必要があるでしょう。JSP&サーブレットやPHPで採用されているモデルは、「HTML埋め込み型」あるいは「ページインラインモデル」と呼ばれるもので

す。ページインラインモデルとは、HTMLで作成されたレイアウトに対して、部分的なプログラムを埋め込んでゆくイメージです(図2左)。

しかし、Webアプリケーションが次第と複雑化し、時には基幹業務システムにも採用されるようになると、レイアウトとプログラムとが複雑に絡み合っているページインラインモデルの限界が見えてきました。

ページインラインモデルでは、プログラムが少し複雑になると、コード全体の可読性が著しく悪くなるという決定的な短所があったのです。Webアプリケーションに要求されるものが次第に高度になるのに応じて、プログラマとコンテンツ作成者、デザイナーによる分業も進んできましたが、ページインラインモデルではひとつのソースコードを前に、複数の作業者が突つき合わなければならないという事情もありました。

しかし、イベントドリブンモデルにおいては、まず画面設計がありきです。画面上に配置された「ボタンがクリックされた」「選択ボックスの内容が変更された」、あるいは「ページそのものが呼び出された」などのイベントをトリガー(きっかけ)として行なうべき一連の処理手続きを記述する方式なので、おのずとレイアウトとプログラムとが分離します(図2右)。

しかも、ASP.NETには「コードビハインド」という機能が用意されており、コードの部分だけを別ファイルとして完全に分離することが可能となっているのです。これによって、上に挙げたような「開発の分業」が促進できるだ