

Webアプリケーションの イベントドリブンを理解する

サーバーでは何が行なわれているのか

山田 祥寛
YAMADA, Yoshihiro

Technology Tools

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

Level



Samples

ページインライン モデルの問題点

従来のASP3.0（レガシーASP）をはじめ、PHP（PHP：Hypertext Preprocessor）やJSP（Java Server Pages）など従来のサーバーサイド技術における開発モデルは、HTMLを主体にロジックを適宜ページ中に埋め込む「ページインライン」と呼ばれるモデルです。見慣れたHTMLをベースに記述できる、必要な箇所に適宜断片的なスクリプト（スクリプトレット）を記述することのできる——その簡易さはサーバーサイド技術の普及を支えた重要な一要素でもありました。

■スパゲッティなコード

しかし一方で、「ページインライン」モデルの場合、コードがどうしても表示順や位置に左右されやすく、また、複雑なレイアウトになった場合、おのずとロジック自体がページ内に散在してしまい、複雑に絡み合ってしまう、という側面があるのも事実です（コードの「スパゲッティ状態」といいます）。

スパゲッティ状態となったコードは、ページレイアウトのちょっとした変更にもセンシティブに影響を受け、メンテナンスが煩雑になるだけではなく、分業／細分化が進んできたWeb開発の分野において、デザイナーとプログラマーが並行作業を行なうことを困難にしました。

■ページインラインこそが特殊 ～IDEの導入を阻むもの～

また、HTMLタグにロジックが密に組み込まれていることから、IDE環境での構築にも比較的不向きであったという事情があります。事実、IDE環境での開発の最たる長所は、GUIによって直感的に画面イメージを把握しながら、ビジネスロジックを実装してゆける点にあります。しかし、いわゆるプレゼンテーション（デザイン）部分にロジックが混在したページインライン方式においては、IDEが表示する画面イメージはきわめて断片的なものにならざるを得ません。

このような事情から、ますます高度化しつつあるWeb開発の分野におい

て、開発効率性、保守性という2つの側面から「ページインライン」モデルによる開発は限界に達していたといえます。

■ イベントドリブンの導入

ASP.NETでは旧来の「ページインライン」方式を部分的に残しつつも、方式の主体は「イベントドリブン（イベント駆動型）」に移すことで、既知の問題の解決を図っています。

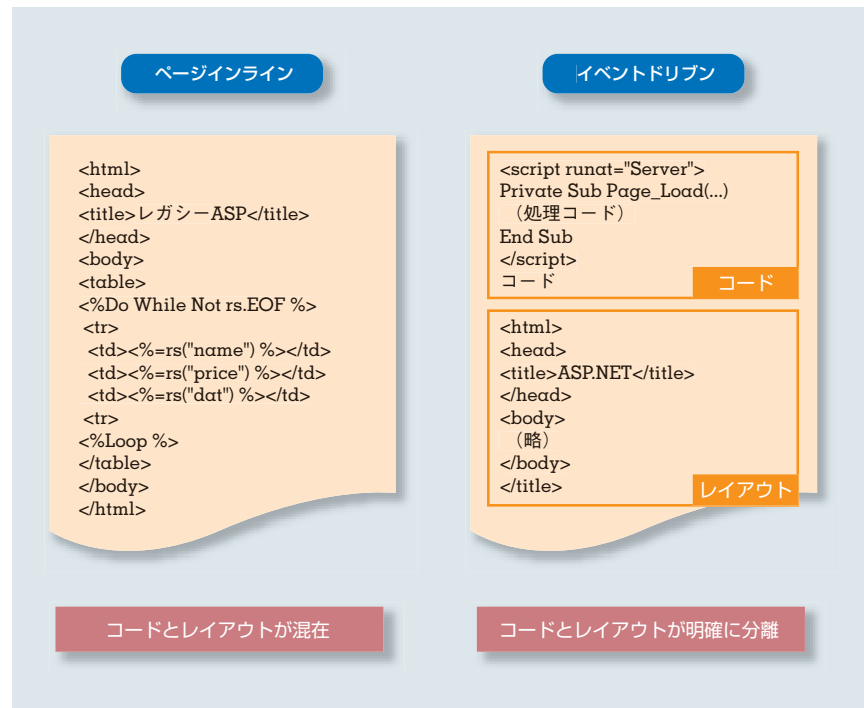
「イベントドリブン」とは、フォーム（ページ上）で発生した事象（ページがロードされた、ボタンがクリックされたなど）をすべて「イベント」という形で捉え、これをトリガーとして一連の動作（ロジック）を制御する方式です。

レガシーASPはじめ、PHPやJSPなどでのみプログラミングを行ってきた方にとっては、やや違和感を感じるかもしれませんが、伝統的なWindowsプログラミング（Visual BasicやVisual C++などによる）にあっては、実は、きわめて典型的な手法でもあります。

イベントドリブン方式は、原則として、ページレイアウトありきで、ページの各所で発生したイベントに基づいて、ビジネスロジックが走るという発想ですから、レイアウトとコードの分離が明確です。

つまり、ASP.NETにおいては、新たにイベントドリブン方式を採用したことによって、レガシーASPの問題点を解消したと同時に、WindowsプログラミングとWebプログラミングとをまったく同じパラダイム（概念）のもとで構築できるようになったのです（図1）。

図1：ページインラインとイベントドリブン



また、あとで述べますが、イベントドリブン方式は、従来はどうしても相対する概念となりがちであったサーバーサイド、クライアントサイドという2つの対象を、互いに補完し合う——限りなく「融合の関係」に近づけたといえます。

ドキュメント登録システム (アップロード)

概念だけを見ても、なかなかイメージがつかみにくいかもしれません。ここでひとつ、具体的なサンプルを作成してみることにしましょう。

本稿で紹介する「ドキュメント登録システム」は、任意のファイルをサーバー上にアップロードし、一元的に管理するWeb版「ファイルサーバー」です。ローカル環境で作成したファイル

資産をサーバー上にアップロードし、公開するという、きわめて単純なしくみではありますが、ファイル公開にあたって、いちいちHTMLを記述しなくても良い点、ファイル情報を一元的にデータベースで管理できる点は、煩雑なドキュメント管理を容易にします。

■ GUI部品を

Webフォーム上に配置する

イベントドリブン方式のプログラミングにおいては、まずイベントを発生する元となるGUI部品（テキストボックスやコンボボックス、コマンドボタンなど）がなければ話になりません。たとえば、内容が変更された（Server Change イベント）、マウスでクリックされた（ServerClick イベント）などイベントの多くは、これらGUI部品上で発生するものだからです。これらGUI