

産業用 Web アプリケーション構築技術 ～ 監視系と情報系の混在した Web アプリケーションの実現 ～

寺岡 照彦 大崎 雅代 柳原 慎太郎 中田 秀男

三菱電機 (株) 先端技術総合研究所

1. はじめに

Web は情報の閲覧だけでなく、ネットワークを利用したシステムのプラットフォームとして広く利用されるようになってきた。産業分野においては、業務支援システムなどデータベースと連携した情報系のシステムだけでなく、機器・設備の監視、防災情報や道路情報の表示といった、刻々と変化する情報を提示する監視系のシステムへの適用が進んでいる。このような Web アプリケーションの開発では、さまざまなプログラミングの専門知識が必要であり、開発支援と生産性の向上が急務となっている。本稿では、このような産業用の Web アプリケーションのための構築技術について説明する。

2. Web アプリケーション構築技術

2.1 Web アプリケーションの開発

従来の Web アプリケーションの開発環境は、データベース等と連携した情報系のアプリケーションの開発支援 [1] が主流である。一方で、監視系のアプリケーションではグラフィックスが多用されるため、画面は Java Applet や ActiveX、Flash などの専用モジュールとして作成されることが多い。これらは自由度が高い反面、専用の開発環境が必要であり、汎用性が低く、読込みや起動に時間がかかる、等の問題がある。

以上のような背景から、監視系と情報系が混在した産業用 Web アプリケーションの生産性向上のための、構築環境を開発した [2, 3]。特長を以下に示す。

1. 高度なプログラミング知識を必要としない視覚的エンジニアリング環境を提供
2. Web 画面のデザインに市販ツールを利用可能
3. 状態遷移図をベースに、アプリケーションの振る舞いを直感的に記述可能
4. 画面デザイン、状態遷移定義、データ(変数)定義、ビジネスロジックを明確に分離
 - 画面デザインとビジネスロジック間のパラメータの不整合を抑制
 - 画面遷移の容易な変更を実現
5. ビジネスロジックを共有した多種端末対応のアプリケーションを効率よく実現

6. 表示項目の自動更新を行う監視系画面を、SVG(Scalable Vector Graphics)により実現

開発した構築環境の概要を図 1 に示す。構築環境は、「エディタ」、「ジェネレータ」、「実行エンジン」からなる。これらについて説明した後、監視画面の実現について説明する。情報系に必要な、データベース等と連携した画面の実現は、文献 [4] を参照のこと。

2.2 エディタ

エディタは、「状態遷移定義部」、「ビジネスロジック定義部」、「バインディング定義部」、「データ定義部」からなる。状態遷移定義部では、市販ツールで作成した雛形の Web 画面を読み込み、状態遷移図を用いてアプリケーションの状態遷移を記述する。ビジネスロジック定義部では、サーバサイドで実行されるビジネスロジックを、プログラマブルコントローラ向けプログラミング言語の標準規格 IEC-61131-3 におけるファンクションブロックを用いて、視覚的に記述する。ビジネスロジックの入出力変数と、画面上の入力/表示変数との対応づけは、バインディング定義部で記述する。ビジネスロジックに用いる変数の型や名称はデータ定義部にて行う。エディタで定義した記述結果は、Web アプリケーションモデル記述として出力する [2]。

2.3 ジェネレータ

ジェネレータは、エディタが出力した Web アプリケーションモデル記述から、実行可能な Web アプリケーションを生成する。具体的には、必要となる Java プログラムの自動生成やコンパイルと、雛形の Web 画面に JSP カスタムタグやページ遷移に必要な情報を埋め込んで JSP ファイルを生成する。

2.4 実行エンジン

実行エンジンは、MVC モデル 2 アーキテクチャに基づいて実装されており、エディタで記述した状態遷移定義に従って、Web アプリケーションを動作させる。

2.5 監視画面の実現

表示項目が刻々と変化する監視画面を実現する単純な方法は、ユーザが閲覧中の画面を再読込みすることである。しかしこの方法では、クライアントとサーバ間の通信量が多い、再読込みの頻度をあげると負荷が高い、スクロールなどのユーザ操作が取消される、画面のちらつきが発生する、などの問題がある。これらを解決するために、HTML のフレームとスクリプト機能を利用した手法を提案した [5]。この手法では、見えている Web 画面の再読みを行わずに、(1) 表示サイズ

“Industrial Web application development technology – Development of a Web application with which the monitoring system and the information system integrated –”, Teruhiko Teraoka, Masayo Osaki, Shintarou Yanagihara, Hideo Nakata, Mitsubishi Electric Corporation

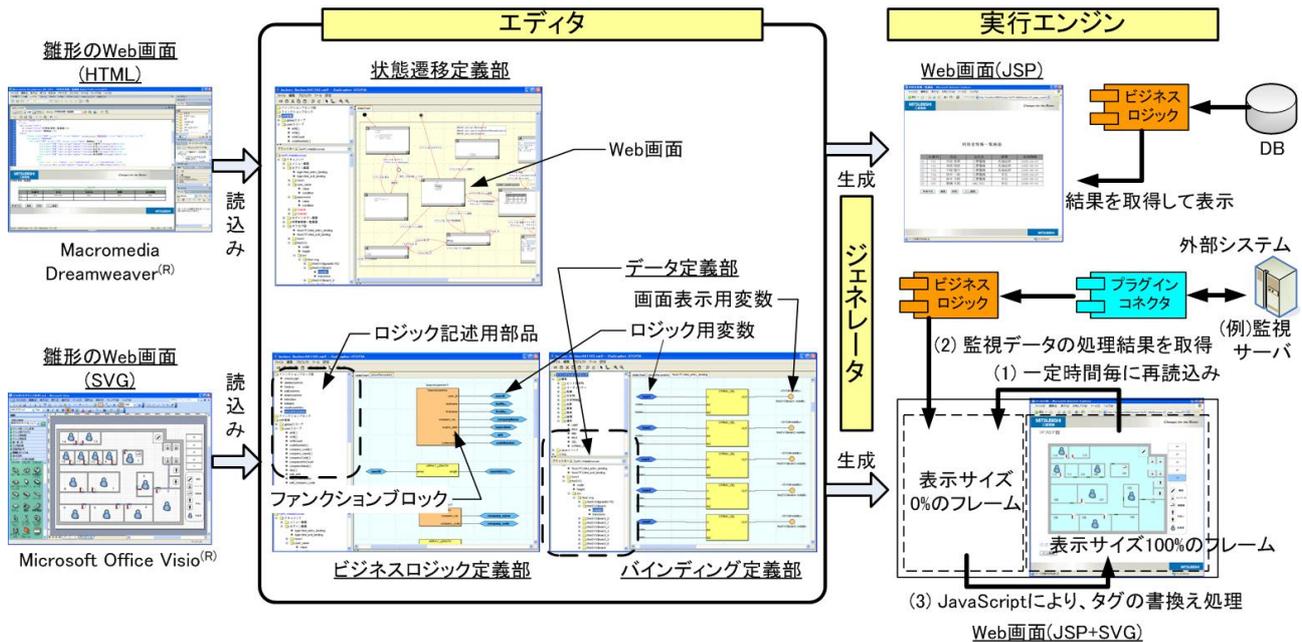


図 1: Web アプリケーション構築環境の概要

が 0% の Web 画面の再読み込みをスクリプトのタイマ機能で一定時間毎に行い、(2) 監視データの処理結果を取得し、(3) この画面のスクリプトによって見えている画面の変更箇所を書換える (図 1 の右下参照)。

開発においては、エディタの状態遷移定義部で、当該画面の更新間隔や外部システムの指定などのパラメータを定義し、データ定義部で、画面内で自動更新する表示変数の定義を行う。さらに、ビジネスロジック定義部とバインディング定義部で、データ値に応じた表示項目の変更ロジックを定義する。ジェネレータでは、これらの定義を基にマルチフレーム画面の生成ならびに、自動更新に必要なとなるスクリプト (JavaScript) を JSP に埋め込む。監視サーバなどの外部システムとの連携は、実行エンジンのコントローラであるサーブレットにプラグイン可能なコネクタを介して実現する。

3. 適用例

図 2 は、ビルの各部屋の入室状況の監視を行う Web アプリケーションの実現例である。フロア図上で、部屋毎の入室者の有無や、ドアの開閉情報をグラフィックスによって監視することができる。グラフィックスのデザインは、Microsoft Office Visio®2003 で行った。また、部屋の入室はカードで制御され、その利用者情報はデータベースで一元管理される。このような監視系と情報系が混在した Web アプリケーションを、本構築環境で容易に実現できることを確認した。

4. おわりに

本稿では、業務支援システムなどの情報系のシステムと、刻々と変化する情報を提示する監視系のシステムが混在した Web アプリケーションのための構築技術について説明した。今後は、汎用フレームワークへの対応や、試験/デバッグ機能の充実を図る予定である。

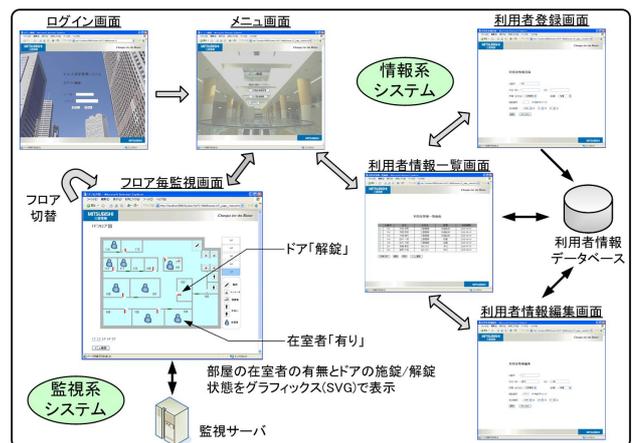


図 2: 監視系と情報系が混在したアプリケーションの例

参考文献

- [1] 田井, 根路銘, 安部, 堀, “モデルに基づく Web アプリケーション開発支援環境”, 情処論, Vol.44, No.6, pp.1498-1508 (2003)
- [2] 大崎, 寺岡, 中田, “状態マシンに基づく Web アプリケーションのモデル化”, 信学技報 ソフトウェアサイエンス, SS2004-37 (2004)
- [3] 寺岡, 大崎, 中田, “状態遷移図に基づく Web アプリケーション構築環境と産業用システムへの適用”, 電気学会情報システム研究会資料, IS-04-58 (2004)
- [4] 石原, 中田, 他, “ユビキタスユーザインタフェースミドルウェア (1)~(4)”, 第 66 回情処全大, 6G-1~6G-4 (2004)
- [5] 大崎, 寺岡, “外部データと連携して自動更新する要素を含む Web 画面の実現”, FIT2004, B-045 (2004)