

コーディングレスでWebアプリケーションの作成を実現するGdForm

竹下俊・松尾真臣
神田俊之・丸田善彦

要旨

GdFormはExcelを用いることによってコーディングレスでWebアプリケーションを作成できるツールです。GdFormを使うことにより、システム開発工数の削減やアプリケーションメンテナンス性の向上などのメリットを得ることができます。本稿では、今回開発したGdFormの機能を紹介します。また、XMLやGdFrameといった、GdFormを実現した技術についても述べます。

キーワード

●情報システム ●コーディングレス ●業務アプリケーション ●XML ●Excel ●ASP.NET

1. はじめに

昨今の企業は事業環境の変化や他社との競争激化からビジネス革新を余儀なくされ、情報システムにも継続的な改善や変更が必要とされています。

一方で、企業の情報システム開発は、IT基盤やアプリケーション開発技術のめまぐるしい進化により、ITの専門性が一段と求められ、結果として開発運用コストを更に増大させるか、情報システムの追従をあきらめるしかない状況にあります。この問題は情報システムの開発に高度なスキルが要求されることに起因しており、解決するには情報システムをITの専門家に任せせる高度なスキルを必要とする部分と一般的な技術で実現できる部分に分離し、餅は餅屋のシステム構造にす

る必要があります。

このような問題を解決するツールとして、高度なプログラミングスキルを必要とせず、Excelを使ってコーディングレスでアプリケーション作成を実現する「GdForm」を開発しましたので、紹介します。

2. 特長

GdFormはコーディングレスでWebアプリケーション作成を実現するツールです。開発者は専用のExcelシート(GdFormDesigner)に画面仕様を記述します。

画面に表示する文字列や、データベースと表示項目のマッピングなどの情報をGdFormDesignerに入力し、サーバに送信

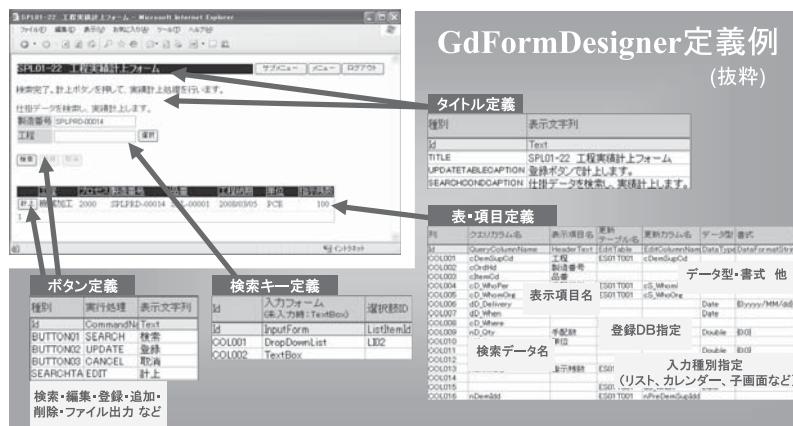


図1 GdFormDesigner定義例（抜粋）

すると、その内容がWeb画面に反映され、検索や登録などのデータベースアクセス処理を実行できるWebアプリケーションとなります。GdFormDesignerの記述と実際に表示されるWeb画面の例を図1に示します。

GdFormDesignerに記述する画面仕様は、一般的な開発におけるソースコードに該当します。本稿では、この「GdFormDesignerに記述する画面仕様」を「画面定義」と表します。GdFormDesignerの1シートが1画面定義となります。

以下は、GdForm使用時のメリットです。

(1) 開発フェーズでのメリット

コーディングの必要がないため、開発者はプログラミングの専門知識が不要となり、製造のコストを抑えることができます。単体テストでは、画面遷移などGdFormによって品質が担保される部分のテストを省略できるため、スクラッチ開発時と比較し、大きくコスト削減できます。

(2) 要件定義フェーズでのメリット

要件定義時に画面仕様をGdFormDesignerへ記述することで、プロトタイプ画面を即時に作成できます。実際に動作する画面を早期に確認できることで、上流工程の品質を高め、戻り作業のコストを削減します。作成したプロトタイプは製造工程にて流用することで、更にトータルコストを減らすことができます。

(3) 保守フェーズでのメリット

画面定義がそのままアプリケーションとして動作するため、画面と仕様の乖離を防ぐことができます。また、画面定義の記述はGdFormDesignerのフォーマットに則っていますので、属人性が排除され、形式は必然的に標準化されます。そのため、仕様変更の対応者が画面定義の作成者とは異なる場合であっても、アプリケーションの修正が容易です。

システムの再構築をするプロジェクトで、スクラッチ開発とGdForm使用時の開発生産性について比較検討したところ、開発画面全体のうちGdFormでの画面作成が約6割の場合で、概要設計から内部結合テストまでの間で25%の効率化となった事例があります。

3. システム構成

GdFormのシステム構成を図2に示します。画面定義情報は

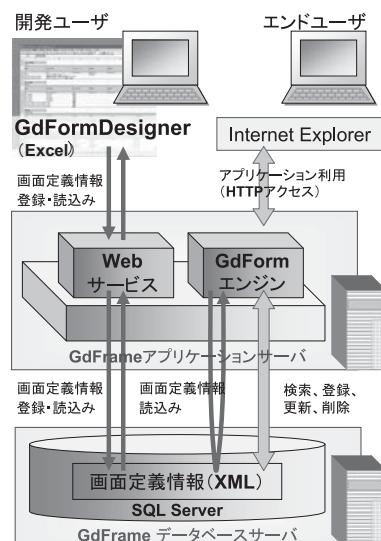


図2 システム構成図

GdFormDesignerからアプリケーションサーバのWebサービスを経由してデータベースへ格納されます。GdFormエンジンはエンドユーザからアクセスがあった時点で、必要な画面定義情報をデータベースから取得し、その内容に従って動的に画面を生成します。画面定義を変更すると、次回アクセス時に変更された画面が表示されます。

図2のGdFrameは、基幹業務システム構築のフレームワークであり、第5章第3節で後述します。

4. GdFormで実現できる機能

GdFormが実現するアプリケーションの主要機能を説明します。

(1) 検索、編集（登録、更新、削除）処理

検索で取得するデータは、GdFormDesignerにクエリを記述します。画面に表示する項目と、検索結果の項目及び編集対象テーブルの項目とのマッピングは、前述した図1の「表・項目定義」に示す形式で記述します。この定義に従い、GdFormエンジンは動的に型付きデータセット¹を作成し、検索結果や編集データの格納に利用します。編集処理には、対象テーブルのスキーマ情報が必要となりますが、

¹ データベースから読み込んだデータを保持するためのオブジェクト

普通論文

コーディングレスでWebアプリケーションの作成を実現するGdForm

GdFormDesignerはデータベースサーバのテーブルスキーマ情報をWebサービス経由で取得する機能を備えています。こうした機能により、開発に使用するツールをGdFormDesignerのみとすることが可能です。

(2) 標準的な画面遷移

GdFormでは、業務アプリケーションで用いられる画面レイアウト及び画面遷移を整理し、まとめています。

・ 画面レイアウト

入力欄の形式別の分類で、一覧表、単票、ヘッダ/明細(図3)の3種類があります。

・ 画面遷移

編集可能な状態までの画面遷移の違いにより、次の2種類があります。「」内はエンドユーザの動作を表します。

1) 「検索ボタン押下」 → 結果の一覧表示 → 「検索結果から編集対象を選択」 → 選択対象を編集可能な状態で表示

2) 「検索ボタン押下」 → 結果を編集可能な状態で表示

GdFormでは、この画面レイアウトと画面遷移の組み合わせを画面パターンと呼び、次の6種類があります。

1) 検索 → 検索結果 → 選択 → 一覧表入力

2) 検索 → 検索結果 → 選択 → 単票入力

3) 検索 → 単票入力

4) 検索 → 一覧表入力

5) 検索 → 検索結果 → 選択 → ヘッダ明細入力

6) 検索 → ヘッダ/明細入力

開発者は作成するアプリケーションの機能に合った画面パターンを選択します。

ヘッダ部

明細部

図3 ヘッダ/明細入力形式

(3) 入力フォーム

データの新規登録や更新時の入力フォームとしてテキストボックス、ドロップダウンリスト、チェックボックス、カレンダー、子画面検索、HTMLタグから選択できます。

子画面検索とは、画面から子画面（ウィンドウダイアログ）を開き、子画面にマスターデータなどの検索結果一覧を表示し、その中から選択したレコードを入力値とするものです。子画面に表示するデータをクエリ指定するだけで既定の子画面が使用可能ですが、GdFormで作成した画面や、スクラッチ開発した画面も子画面として用いることができます。また入力フォームをHTMLで記述可能です。このように一定のカスタマイズ性を確保することで、細かな要求仕様に対応できるようにしています。

(4) 入力チェック

基本的な入力チェックとして必須項目チェック、正規表現チェック、最大・最小値チェックを持つほか、GdForm独自のチェックとして行チェックの指定が可能です。行チェックとは同一レコード内の複数項目の値によるチェックです。例えば、2つの項目の入力値の和が100以下であること、といったチェック式を記述することができます。

(5) その他機能

自動採番、ログインユーザ情報やシステム時刻の取得、TSVやCSV形式でのファイル出力などの機能を持ちます。

5. GdFormの主要技術

5.1 XML

GdFormDesignerに記述した画面定義はXMLに変換してデータベースへ登録しています。画面生成時はGdFormエンジンでXMLの画面定義が読み込まれます。

このようにGdFormは画面定義情報をXMLで扱っていますが、その理由には主に次の2点があります。1点目は.NET FrameworkとXMLの高い親和性によるものです。XMLの画面定義情報は.NET Frameworkのライブラリでデータセットに簡単に取り込むことができ、扱いも容易です。2点目はXMLの優れたデータ拡張性にあります。GdFormの機能拡充によって新たに定義可能な項目をGdFormDesignerに追加したとしても、この拡張性のおかげで、過去に作成した画面定義には修正を加えることなく動作させることができます。

5.2 ASP.NETによるWebフォームの動的生成

画面定義情報に従ってWebフォームを動的に生成しているのは、ASP.NETアプリケーションであるGdFormエンジンです。ASP.NETアプリケーションには画面デザインと処理ロジックを完全に分離可能という特長があります。GdFormエンジンはその特長を生かし、画面デザイン部にはボタンやタイトル用ラベルなどを配置した画面雰形のみを持ち、処理ロジック部で実際のアプリケーション画面となるWebフォームを画面雰形から生成します。画面雰形は画面パターンごとに用意していますが、処理ロジックは共通化しています。

処理ロジック部では、画面定義に従ってボタン押下時の動作やラベルに表示する文字列などを画面雰形に当て込んで動的にWebフォームを生成する処理のほか、画面定義の検索クエリやデータマッピングに従ってデータベースアクセスを行う処理を実装しています。

5.3 GdFormを支える情報統合基盤「GdFrame」

GdFrameは基幹業務システム構築のためのフレームワーク製品です。5W1H型の構造を持つデータベースと、そのデータベースへのアクセス機能、アプリケーション作成に共通的に利用できる機能、ログイン画面やログ確認画面などの汎用アプリケーションで構成されています。

従来、情報システムは個別システムが多数乱立し、システムごとに個別のデータベースを構築することが多く、システム改造が複雑化し、業務の変化に追従し難いという問題がありました。GdFrameでは汎用的構造のデータベースを持つことで、この問題を解決しています。各システムのデータを汎用データベースに一元化することで、システム改造時にデータベースまで変更する必要がないため、業務変化に強い情報システムを実現します。

GdFormはGdFrameが提供するデータベースアクセスAPIを利用しています。このAPIによってデータベースアクセス時に詳細なログが outputされ、GdFrameの持つログ確認画面で処理内容を確認することができます。エンドユーザのデータ操作ログが自動収集されるほか、開発時のデバッグにも利用でき、開発効率の向上に貢献します。

GdFormとGdFrameを利用してすることで、画面機能とデータ処理機に分離した実装方式となり、生産性・保守性・品質を向

上させることができます。

6. おわりに

コーディングレスでのアプリケーション作成ツールに求められるのは、多くの要求仕様に対応できる機能性と、アプリケーション作成の容易さの両立と考えます。GdFormはこれまでさまざまな機能を実現しつつも、画面定義方法を煩雑にすることなく、より多くの業務アプリケーションを作成できるよう開発を行ってきました。GdFrameとGdFormによる情報システムのアプリケーション開発は、システム開発工数の削減やアプリケーションメンテナンス性の向上などのメリットを得ることができます。

今後はWebアプリケーションのみならず、ハンディターミナル用アプリケーションや携帯電話用アプリケーションにも作成対象を広げ、更に多くのシステム開発で適用を容易にすることを目指し、製品力向上を継続していきます。

*Excelは、米国Microsoft Corporationの米国、及びその他の国における登録商標または商標です。

執筆者プロフィール

竹下俊
NEC情報システムズ
ITソリューション事業部
マネージャー

松尾真臣
NEC情報システムズ
ITソリューション事業部
主任

神田俊之
NEC情報システムズ
ITソリューション事業部
主任

丸田善彦
NEC情報システムズ
ITソリューション事業部
主任

●本論文に関する詳細は下記をご覧ください。

関連URL

GdFrame製品に関するホームページ:
<http://www.nec.co.jp/GdFrame/>