

ウェブアプリケーション開発のための UI プロトタイピング支援環境の構築

赤羽 祐亮[†] 井垣 宏[†] 田胡 和哉[†]

[†]東京工科大学コンピュータサイエンス専攻@

1 はじめに

クラウドコンピューティングの発展にともない、小規模な Web アプリケーションを短納期で多数開発できる様にする必要性が高まっている。このためには、エンドユーザーの開発への参加が重要になるが、アプリケーションをエンドユーザーだけで開発することは現実的ではない。ここでは、エンドユーザーや営業担当者が、プロトタイピングを通じて設計を行えるようにする手法について検討する。これにより、短期間で手戻りの少ない設計が行えることが期待出来る。

この目的のためには、プロトタイピングの際に、アルゴリズムを手続きの形式で記述する方法は現実的でない。そこで、遷移ごとの画面を既存の画面設計ツールで設計し、その遷移を起こすために必要な手続きをシステムが推測する、一種の宣言的 [1] な手法をとることを考える。これによって、実際に動作するプロトタイプが実現できれば、UML 等の形式で設計を出力することは比較的容易に行う事が出来る。また、仕様の確認の精度も高くすることが出来る。本稿では、提案方式の概要について述べる。

2 方式の概要

図 1 に、提案方式によるプロトタイピングシステムを示す。マウス操作等によって遷移する画面を、それぞれ html の形式で作成する。ここでは、まったく別の画面に遷移する場合ばかりでなく、動的に画面の一部に変化が起きる場合も結果の画面の例を作成する。たとえば、表に行が追加される場合は、1 行追加された画面を作成しておく。

このようにして作成された複数の画面定義の html をパースし、それぞれを DOM の形式に変換し、差分を判定することにより、システムは、

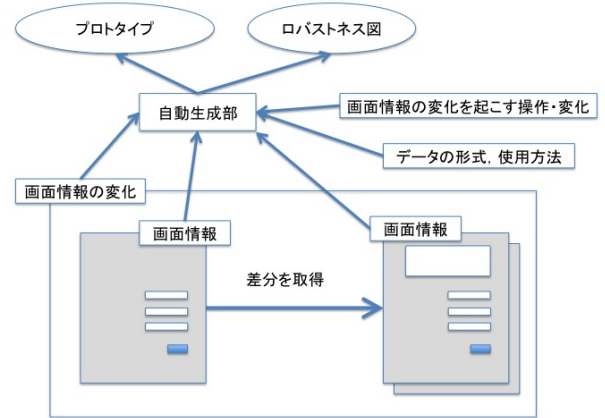


図 1: システム構成図

どのような画面変化が起きたのかを判定できる。したがって、利用者は、

- 遷移ごとの画面設計
- 変化が起きる条件（イベント）の設定
- 画面変化の際に利用されるデータのソース

を指定することによってプロトタイピングが行えるようになることを期待できる。

変化が起きる条件の設定は、例示によって行う。具体的には、設計された html ファイルを表示する際、そのまま表示するのではなく、システムが自動的に起こりうるすべてのイベントに仮の定義を埋め込む。利用者が画面を表示し、イベントを起こすための動作、たとえば、ボタンクリック等の動作をとると、イベント定義のためのウィザード画面が起動する。これをもちいて、遷移先の画面を指定することによって、イベントの定義を行う。この際、同時に、その際に利用されるデータのソースも指定することができる。データのソースが指定された場合、システムは、パラメトリックな遷移であると推定する。ここで、パラメトリックな遷移とは、データの内容によって、遷移後の表示が一部変化する画面遷移を示すことにする。

Building a UI prototyping support environment for Web Development

[†] Yusuke AKABANE(y.akabane@y-ref.com)

[†] Hiroshi IGAKI(igaki@cs.teu.ac.jp)

[†] Kazuya TAGO(ktago@cs.teu.ac.jp)

Department of Computer Science, Tokyo University of Technology (†)

1404-1 Katakura, Hachioji, Tokyo 192-0982, Japan

パラメトリックな遷移に用いられるデータの供給源は、オブジェクトの形式で指定する。オブジェクトは、サーバ機能やDOMの値を抽象化したものである。REST等の方法で提供されるサーバ機能の仕様は、オブジェクト定義の形式であらかじめシステムが保持しているものと仮定する。

パラメトリックな遷移を定義する際、システムは、遷移先の画面が指定されると、遷移元の画面との差分を検出して、定義ウィザードに画面変化の候補を表示する。たとえば、表の内容に差分があれば、既存の行に対する項目の書き換えか、行の追加が変化の候補となる。このとき、例として作成した画面の表部分に遷移形式を指示するアノテーションを記入しておけば、比較的容易に画面遷移の形式を推定することができる。たとえば、@update、@add等の記号を表の項目中に記入しておけば、表の書き換えか、行の追加であるかが区別できる。

画面遷移を定義することができれば、実装することができる。具体的には、JavaScriptのライブラリを用意し、システムがそれへの呼び出しコードを埋め込むことによって実現する。また、この情報からUMLを生成する。これによって、利用者は、JavaScriptの知識なしに、画面設計を行うだけでプロトタイプを行うことが可能になる。使用するツールによっては、htmlに関する知識もほとんど必要としない。

3 設計例

宣言的な方法によってプロトタイプの設計が行えるかは、自明ではない。利用者が指定したい、典型的なケースについて、事例を検討してみる。

3.1 ユーザーによる操作の指定

ボタンを押すといった単純な操作は一つのイベントによって構築されているが、ドラッグ&ドロップはドラッグする対象にドロップされる領域を加えた二つの要素によって構築されている。その為、ユーザーによる操作の対象は、実際に操作に関わる要素の数だけ、対象を指定する必要がある。

具体的に設定を行う項目としては、行いたい操作の種別と、その操作に関わる対象の全てを指定する。

3.2 操作による画面変化の指定

操作による画面変化の指定は、操作の種別に関係なく、操作の結果として指定される動的データの形式以外の差は無いと考えられる。

動的データを元に表示する場合は、どのようなデータを、どのような形で表示できるかを定義する必要がある。単一データの場合はidを指定してのマッピングし、そのデータをどのように表示するか指定を行う。単一データ以外は単一データ同様の指定に加え、テーブルの様なレコード表示をテンプレートとして定義しておくことが考えられる。設定される項目としては、影響を受ける対象の指定、対象の変化に影響するデータ、データによる影響の与え方を指定する必要がある。

設定される具体的な例として、テーブルデータを使用した場合を挙げる(図2)。項目としては影響を受ける対象として、実際に変化するテーブルを、対象の変化に影響するデータとして、別途定義されているデータオブジェクトを、データによる影響の与え方として、データの向き、テーブルのサイズ、オブジェクトidによるデータのマッピングを指定する必要がある。

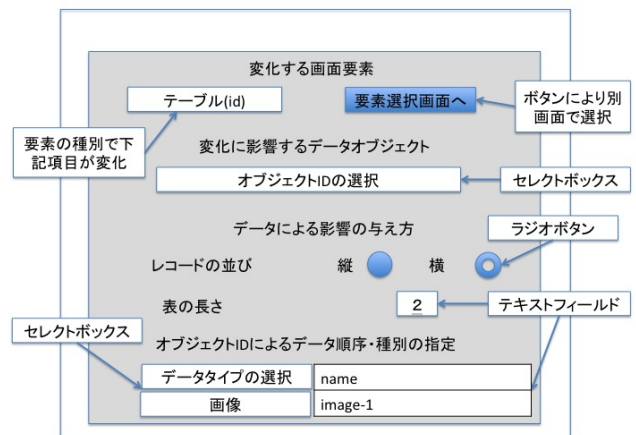


図2: 操作による画面変化設定ウィザードの具体例

4 むすび

プログラミングの知識が少ない利用者を主体として、Webアプリケーションを設計する手法について提案した。現在、クラウドシステムの構築を行っているため、ここで利用することにより有効性に関する知見を収集してゆきたい。

参考文献

[1] Frans Coenen. Characteristics of declarative programming languages