

選択問題システム「サッと選択！」のJavaScriptへの移植

西 村 則 久

Porting “Satto Sentaku” (Selection Problem System) to JavaScript

Norihisa NISHIMURA

要 旨

選択問題システム「サッと選択！」は、選択問題をコンピュータが繰り返し出題するシステムである。安田女子大学においては、英語英米文学科の専門科目の授業や共通教育科目の英語の授業などで利用されている。このたびシステムの一部を改良（移植）したので報告する。

I. 選択問題システム「サッと選択！」の概要

選択問題システム「サッと選択！」は2002年から安田女子大学で運用実験を繰り返しながら実用化された、選択問題をコンピュータが出題するシステムである。ソフトウェアの形態はWebアプリケーションである。開発の技術面は筆者が担当している。北辰映電株式会社によって商用化されており、安田女子大学以外の教育機関でも利用されている。

利用の流れを大まかに説明する。学習を開始すると、ブラウザ上に問題文およびその答えとして複数の選択肢（ボタン）が表示される（図1）。

学習者は選択肢の中から正解と思うものをクリックで選択する。クリックされた選択肢は、橙色に変わる（図2）。



図1 出題画面



図2 答えを選択した状態

次に、学習者の選択が正解か不正解かに応じて、正解／不正解の表示がなされる。正解の場合、学習者が選んだ選択肢は緑色に変わる。不正解の場合、学習者が選んだ選択肢は橙色のまま、正解の選択肢が緑色に変わる（図3）。



図3 正解／不正解の表示（不正解の例）

続けて、次の問題へ遷移するためのボタンが表示されるので、学習者はそれをクリックして次の問題へと進む。以上が、本システムを学習者が利用する際の基本的な流れである。

Web上で選択問題を解いていくシステムは本システムが開発されるよりもインターネット上でよく見られたが、それらの多くはWebブラウザが標準的に用意しているHTML言語のGUI部品を使用して作られており、機能も外観も簡素なもので、HTML言語のソースコードを表示させればどれが正解かわかってしまうという欠点もあった。本システムは授業で本格的に利用することを考えたため、不正防止の仕組みや、グラフィカルな表示に基づく高い操作性を追求して

いる。また、画像の表示と音声の再生もできるようにしており、教師用の管理ページには、問題作成、問題の階層化、ヒント表示、コメント表示、出題パターンのカスタマイズ、確認テスト、学習履歴表示など、様々な設定・機能がある。ただし本報告はシステムの移植が主眼のため、それらについては述べない。

II. 移植に至る経緯

本システムは2002年に安田女子大学で運用され始めた。当時、ブラウザ上にグラフィックの表示を行う手段にはJavaScriptとJavaがあったが、その頃のJavaScriptは機能が豊富でなかったり、ブラウザによって言語仕様が異なったりしたのに対し、Javaは機能が比較的豊富で、また異なるブラウザでも比較的安定して動作していたことから、ほぼ必然的にJavaで本システムのグラフィック表示部を実装した。

しかしその後、ブラウザ上でアニメーションを行うソフトとして普及していたFlashが、プログラミングの機能を大幅に強化したため、グラフィック表示の有力な手段となった。それに対してJavaは進化に乏しく、Flashのほうが動作も安定していたことから、Javaに代わってFlashが主流となっていった。安田女子大学の教室のパソコンにもJava実行環境はインストールされなくなることが決まったので、本システムの利用が継続されるためには、グラフィック表示部をFlashに移植することが必然かつ急務となった。2010年に、Javaで行っていたグラフィック表示を全てFlashで行うように移植をした。以降、Flashの普及率は全世界で99%以上と言われるまでになり、本システムは安定的に運用されたが、近年になって再度新たな局面を迎えた。

1. スマートホンの台頭

近年、携帯電話が普及してiPhoneやAndroidといったスマートホンでインターネットのブラウザを簡便に利用することができるようになると、従来であればパソコンのブラウザで行っていたことをスマートホンで済ませる人が増えた。当然、本システムもスマートホンで利用できれば良いのだが、Flashは動作が重いと言われておりスマートホンには全く普及しなかったため、本システムはほとんど全てのスマートホンで利用できない状態であった。

2. Flashの退廃

インターネットの利用の大半がスマートホンで行われるようになると、Flashで作られたものは動作しないので、Flashの優位性が急速に損なわれることとなった。ブラウザとして最大のシェアを獲得しているGoogle Chromeにおいては、内蔵のFlashプレイヤーが無効化されることが決まり、Flashの製造元であるAdobeも潔くFlashプレイヤーの開発中止を決定し、アンインストーラーをリリースするなどした。こうして、本システムはパソコン上でさえ動作しなくなる見通しとなってしまった。

Flashが急速に廃れたその陰には、代替技術としてHTML5という新たなプラットフォームが提案されてきたことがある。JavaやFlashは単独のメーカーが開発したものであるためブラウザに追加型のプラグインをインストールしなければならないが、HTML5は共通化の呼び声のもと、ブラウザ自体が対応する仕様となっており、スクリプト言語にJavaScriptを採用している。

JavaScriptはかつては低機能・低互換性のためあまり採用されていなかったが、近年はHTML5推進の流れにより高機能化と異ブラウザ間の高い互換性が実現しつつある。

以上のことから、本システムのグラフィック表示部をFlashからJavaScriptへと移植することにした。HTML5はブラウザが対応するので、JavaScriptに移植すればスマートフォンでも動作することになる。本システムをスマートフォンで利用できないことは懸案となりつつあったことから、ここへ来てこの移植は必須の選択であったと言える。

Ⅲ. 移 植 の 概 要

本システムは、サーバー側で実行されるプログラムとクライアント側で実行されるプログラムからなる。クライアント側で実行されるプログラムは、クライアントにインストールするわけではなく、クライアントがブラウザでサーバーにアクセスする度にサーバーからクライアントへと送られて実行される。このクライアントで実行されるプログラムが、今回FlashからJavaScriptへ移植したものである。このプログラムの役割は、サーバーから問題データを受信して表示（出題）し、学習者に答えを選ばせ、正解／不正解を表示してその情報をサーバーへ送信することである。

なお、サーバー側のプログラムはC言語とPerl言語で作られており、クライアント側へ問題データを送り、正解／不正解の情報を履歴として保存したり、出題パターンに従って次の問題データを送信したりしている。教師用の管理ページもサーバー側のプログラムである。

Ⅳ. 移 植 の 詳 細

今回の移植を行うのに先立って、本システムと類似のシステムで同様の移植を行っていたため、そこで得た知見を元にして、以下のように移植を進めた。

JavaScriptとFlashのプログラミング言語であるActionScriptはどちらもECMA-262仕様に準拠しており、文法的にも似ているうえ、変数や文字列を操作するライブラリ関数などにもほとんど違いがない。そのため、Flash版のソースコードの大部分はそのまままたは単純な置換をするだけで流用することができた。

よって、移植に必要な開発の大部分は、ボタンなどのUIコンポーネントや文字列を表示するサブルーチンをJavaScriptで作ることであった。Flash版のものと同仕様のサブルーチンを作れば、それをコールするメインルーチンは変更せずそのままよいことになる。

UIコンポーネントはJavaScriptのクラスを利用することも考えられるが、そうすると実行するブラウザによって異なる形状になる可能性もある。ボタンについては、既にFlash版で作っていた独自ボタン作成ルーチンを移植することにした。

Ⅴ. 移 植 上 の 難 点

移植を行う上で難易度が高かったことや、工数を要したことについて述べる。

1. 文字コードの問題

移植前のFlash版において、サーバー側プログラムとクライアント側プログラムとの間で問題データなどの送受信を行う際に、そのまま送受信を行うとサイバー側とクライアント側とで扱える文字コードが異なるために文字化けが発生するので、まず送信側で文字化けの心配のない文字だけで構成される安全な文字列へとエンコードしておいてから送信し、受信側ではそれをデコードして使用するようにはしていた。そのエンコードとデコードのアルゴリズムをそのままJavaScriptでも行うようにしたところ、文字化けが発生した。文字化けが起こらない文字の集合がFlashとJavaScriptとでは異なったためと考えられる。エンコードとデコードのアルゴリズムを試行錯誤的にいろいろ変化させて実験し、文字化けが起こらない送受信方法を見出した。

2. Internet Explorer用の調整

移植の具体的な作業は、HTML5への対応が最も進んだブラウザの一つであるGoogle Chrome (Windows版) で動作確認を行いながら進めたが、一つのブラウザで動作するだけでは不十分であり、シェアが高いブラウザを中心として、なるべく多種多様なブラウザ・OS・スマートホンで動作するようにする必要がある。

本来は、HTML5やJavaScriptの仕様に則って開発しさえすればどの環境でも問題なく動作して然るべきなのであるが、各ブラウザは、段階的に定められた仕様に対して少しずつ追いかけるように対応しているため、最新の仕様通りに開発したのでは一部のブラウザでしか動作しないという現象が起こるのである。

シェアが高いブラウザの中では、Internet Explorerは、HTML5への対応が遅れたまま開発が終了している。そのため、本システムの移植中においても、Internet Explorerだと動作しないというケースが多発した。

なお、動作しないという状態は、細かく分ければ、プログラムがフリーズするケースや、プログラムはフリーズしておらず操作を受け付けるものの画面の一部が表示されていないようなケースがある。

例えばInternet Explorerのバージョン10以前では定数が使えず、実行中のスクリプトに定数が出現したところでプログラムがフリーズした。このケースでは、定数を変数で代用することで、フリーズしなくなった。また、eventという変数名が使用できず、これも実行中にスクリプトが出てきたところでフリーズした。これは別の変数名に置き換えることでフリーズしなくなった。

また、Internet Explorerのバージョン11ではアロー関数が使えず、これはアロー関数の出現時ではなく実行開始後すぐにフリーズしたので、原因の特定が困難であった。アロー関数を通常の関数の表記に変えることでフリーズしなくなった。

このように、動作しない場合にプログラムの組み方を変えて問題を回避することができる場合もあるが、そもそも目的の機能がブラウザに用意されていないような場合は回避することができない。

例えば、Internet Explorerでは、(HTML5の枠組では) 音声を再生することができない。従って音声の再生は断念せざるを得なかった。また、Internet Explorerでは音声を再生させる関数がコールされるとフリーズしていたが、単にこの関数のコールを止めてしまうと他のブラウザでも音声再生されなくなるので、音声データの準備段階でコールされるコールバック関数がコ

ールされている場合に限り音声を再生させる関数をコールするように変更した。これにより、他のブラウザでは音声为正しく再生されるまま、Internet Explorerでのフリーズを回避した。

VI. 結 果

選択問題システム「サッと選択！」の表示部をFlashプログラムからJavaScriptへと移植した。これにより、本システムを近い将来パソコンでは利用できなくなる不安が解消されたのと同時に、新たにスマートホンで利用することができるようになった。

一部のブラウザでは細部において正常に動作しないこともあるが、ブラウザのHTML5対応がメインストリームとなっていることから、本システムに強引な調整を加えなくても不具合は無理なく解消されていくと予想される。

ただし、Internet Explorerは開発が終了している。これについて開発元であるマイクロソフト社は後継ブラウザのMicrosoft Edgeへの乗り換えを推奨している。Microsoft EdgeではHTML5対応が進んでいるため、本システムは完全に動作する。

VII. 今 後 の 課 題

このたび選択問題システム「サッと選択！」をJavaScriptに移植してHTML5対応したことにより、これまでできなかったスマートホンの環境でも本システムを利用できるようになったことは大きな前進である。

しかし今回の移植では、スマートホンの環境でもこれまでパソコンで利用していたのと全く同じ画面構成となっており、スマートホンでの使い易さを考えて再デザインされているわけではない。

一般的にスマートホン上の指でのタップはパソコンのマウスによるクリックよりも位置を正確に合わせるのが難しいので、スマートホンでの表示は大きくあるべきであるが、スマートホンの画面はパソコンよりも解像度が高いため、むしろ小さく表示されてしまっている。ピンチアウト（タッチパネル上で二本の指を広げるように動かす操作）により表示内容を拡大させることはできるが、そうすると表示の全体を一度には見ることはできないので、更にスクロールなどをさせる必要が出て、結果としてパソコンで利用する場合よりも操作性は低下する。

スマートホンの画面にはスマートホンに適した画面構成で表示されるように修正することが今後の課題である。

参 考 文 献

1. 西村則久, 英作文自動添削システム「サッと英作！」のJavaScriptへの移植, 安田女子大学紀要 45, 239-245 2017.

[2018. 9. 27 受理]

コントリビューター：山下 明博 教授（造形デザイン学科）