

1.1 なぜ標準でなければならないのか

昔から同じスタイルでHTMLページやWebサイトを作成してきた人は、「なぜ標準でなければならないのか?」という疑問を持つかもしれません。特に標準に従わなくても、要求されたWebページを制作して見た目が顧客/ユーザの要望に沿っていれば、安定して結果が得られるテーブルレイアウトでよいのではないか、という考えもあります。

それでも、標準に従ってWebページを作成する方がメリットは大きいと言えます。そこでまず、身近な例で標準であるメリット、標準ではないデメリットを確認しておきましょう。

今、あなたがこの本を読んでいるということは、「日本語」の文章が読めるということです。例えば、以下の文字は「あ」としてわかります。

あ

同様に、以下の文字も「い」としてわかるはずで

い

このように日本語の文字が読めるのは、文字の基本的な形状が決まられており、それをどう読むかを学習したからです。このような決まりがなく、各個人がバラバラに文字を決めたとしたら、この本を読むことは可能でしょうか?

各自が書く文字や読み方が異なっていたら、本の内容は書いた人にしか理解できません。それではまるで暗号文です。自分以外の人にも読めるようになって、はじめて情報と知識を共有できるのです。それはWebでも同じことが言えます。

また、他人が読めるようにするには標準的な文字と読み方を決めて、それを学習しなければなりません。Web関連のさまざまな仕様は、主としてW3C(World Wide Web Consortium)^{*1-1}という組織が決めています。W3Cの仕様に沿うことで、自分が作成したWebサイトは他の人にも有益なものになります。「Web標準」に従ったからといって、Webサイトのクリエイティブな部分が損なわれるわけではありません。その逆に、本当にやりたい部分に注力できるようになります。

それでは、Web標準に従うことにはどのようなメリットがあるのでしょうか? 例えば、以下のようなメリットが考えられます。

- ①新しいWebブラウザ/デバイスに対応できる(多様なデバイスへの対応)
- ②一度覚えれば長年使える技術になる(技術/プログラムの再利用)
- ③他のWebサイトのデータも利用できる(データの再利用)
- ④他人と技術を共有できる(教育時間の軽減)

*1-1 公式サイト<http://www.w3.org/>

ここではメリットだけ挙げましたが、人によっては「新たに学習する時間/費用が必要になる」というデメリットが生じます。しかし、Webサイト開発に関わるのであれば、Web標準に関する技術は知っておいた方がよいでしょう。特に本書で解説しているDOM(Document Object Model: 文書オブジェクトモデル)およびDOMスクリプティングは用意されている命令が少なく、考え方も非常にシンプルなので知っておいて損はありません。

また、DOMを操作するスクリプトであるJavaScriptは、ECMA(European Computer Manufacturer Association: ヨーロッパ電子計算機工業会)^{*1-2}においてECMA Scriptと決められています。

それでは、①~④のメリットについて説明しましょう。

①新しいWebブラウザ/デバイスに対応できる(多様なデバイスへの対応)

Webが登場してから間もなく、複数のWebブラウザが登場しました。初期のWebブラウザとしてはMosaic(モザイク)が有名です。その後、Netscape Navigator(ネットスケープナビゲータ)やInternet Explorer(インターネットエクスプローラ)などが登場しました。

Webの発展

Webの発展は、人間の脳が子供から大人へと成長していく過程と似ています。最初は単体で物事を記憶します。1つ記憶することは、Webで言えばHTMLページが1つできたのと同じと言えます。そして、記憶した複数の言葉が1つのもを示すこととなります。Webに言い換えると、複数のHTMLページが集まったWebサイトと同じです。

しかし、多くのWebサイトは自分のサイト内にしかデータを保持せず、再利用を考慮していません。このため、他のWebサイトと結びつけることはできません。

例えば「リンゴ」に関するWebサイトがあるとします。このWebサイトは「ふじ」や「津軽」など、リンゴの種類に関する膨大なデータを持っています。しかし単体で完結しているので、同じ果物なのに「リンゴ」と「ミカン」の関連性を結びつけられないのです。他のWebサイトからは、このWebサイトにある「リンゴ」の情報から「ふじ」と命名されているものを検索するのは難しく、また検索で得られたデータを再利用することも困難です。

Web 2.0/Ajaxなどにより、Webサイト上のXML/JSONデータは利用できるようになりましたが、それはWebサイトのごく一部に過ぎません。もし、Webサイト上のHTMLデータが仕様に沿って標準化されており、プログラムで手軽に取り出せるようになれば、このような結びつきは容易に実現できるはずで、そこから新たな発想が生まれる可能性があるのではないか、と筆者は考えます。

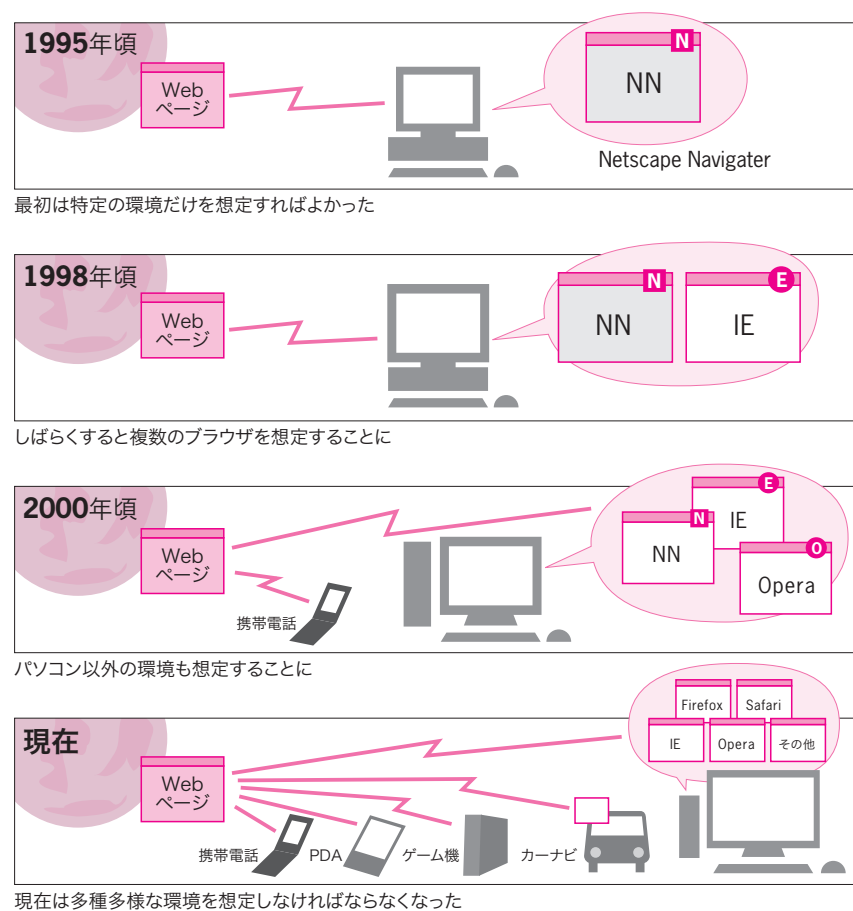
*1-2 公式サイト<http://www.ecma-international.org/>

当時のWebブラウザは現在ほど高性能ではありませんが、テキストと画像を同時に表示できるもの、テキストだけ表示できるもの(Lynxなど)がありました。また、最初はパソコン上でしかWebを閲覧できませんでしたが、現在は携帯電話やカーナビ、ゲーム機など、多くのデバイスで利用できるようになりました。

Webが普及しはじめた頃はNetscape Navigatorのシェアが大きく、これを前提にWebページを作成する傾向がありました。しかし、Internet Explorer 4が登場すると、Netscape NavigatorとInternet Explorerの両方でうまく表示されるように作成するか、あるいはJavaScriptを使ってWebブラウザごとにレイアウトされたWebページを表示するようになりました。多くの場合、Internet Explorerと他のWebブラウザでWebページを分けていました。

Netscape NavigatorとInternet Explorerは機能だけではなく、DOMの実装やスタイルシートの実装も異なるため、互換性が乏しい状態でした。Internet Explorerはバージョン3からCSS(Cascading Style Sheets:カスケードスタイルシート)を実装し、バージョン4で

▼図1.1 1995年以降のインターネット環境の変化



大幅に強化されました。Netscape Navigatorはバージョン4でDOMやCSS, JSS(JavaScript Style Sheet)などを実装しましたが、不具合が多くて利用は困難でした。また、DOMとDynamic HTML(動的に処理できるHTML)も現在とは異なり、独自の仕組みになっていました。この実装のまずさによりDOMやCSSがうまく機能せず、普及が遅れる一因になったとも言われます。

当時のWebブラウザはNetscape NavigatorとInternet Explorerがシェアの大半を占めていたため、Webページの制作者は両者で安定した結果が得られる「テーブルレイアウト」で作成していきました。これならば手間がかからず、コストを抑えられたためです。

しかし、新しいWebブラウザの登場やバージョンアップのたびにWebページを作り直すのは大変です。場合によっては、JavaScriptなどのプログラムを書き直す必要もあるでしょう。Web業界の変化は早いので、その間に新しいデバイスやWebブラウザが登場すると、時代の要求には応えられず、余計なコストも増えてしまいます。

その対策として、共通の仕様に従ってWebページを作成するのです。これによってWebブラウザに依存せず、新しく登場したデバイスにも素早く対応でき、コストも削減できるようになります。これが現在に至る標準化の大きな流れです。

この場面において、HTML/CSSではこれまでに培ったノウハウを利用できます。もちろん、JavaScriptでも従来の手法がHTML DOM Level 2として仕様化されているので、過去のノウハウが全く使えないわけではありません。ただし、ノード処理やイベントなどは、従来とは違う方法でプログラムを作成・実装しなければならない場合があります。

また、JavaScriptは操作対象となる(X)HTML文書の構造とCSSにも依存するため、(X)HTML/CSSとも標準に沿って作成されていない場合、スクリプトで操作する手間が増えてしまいます。プログラムが簡単になるかどうかは、XML/(X)HTMLの作り方に大きく依存することになります。

②一度覚えれば長年使える技術になる(技術/プログラムの再利用)

Web業界は変化が速く、次から次へと新しい技術や考え方が登場します。新しいプログラミング言語がリリースされるだけでなく、既存の言語が何かの拍子で注目されるようになり、急速に普及することもあります。近年注目されるようになった言語にはRubyやPythonなどがあります。Ajaxのおかげで注目されたJavaScriptも、その1つかもかもしれません。

新しい技術や言語が登場するたびに、それらを学習するのはよいことです。しかし、文書の処理方法が言語ごとに異なっていたら、学習するのは大変です。「新しい言語を習得したら、以前の言語で文書処理する方法や呼び出す関数名などを忘れてしまった」ということもありえます。

その点、文書処理において標準化されたDOMを使うことには大きなメリットがあります。その1つが「文書の処理方法が決められている」ことです。文書内のテキストやタグを抽出す