

INTERNET

● インターネット最新テクノロジー : 第33回

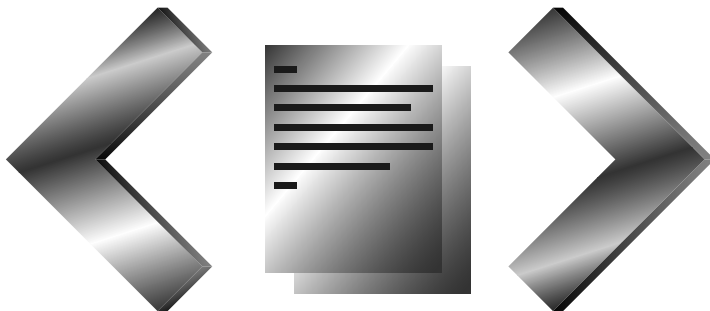
次世代HTMLの標準となる

XHTML

(eXtensible HyperText Markup Language)

W3Cは、新しいHTML仕様であるXHTML1.0(eXtensible HyperText Markup Language1.0)を1月26日に正式に公開した。HTML4.0と互換性を持たせながら、完全にXML(eXtensible Markup Language)ベースとなった新時代のHTMLである。XHTMLは今後次々とする予定の関連仕様と合わせて、今後のウェブに大きなインパクトを与える技術である。

檜山 正幸



XHTML1.0 登場

2000年1月26日、XHTML1.0がW3C勧告となった。XHTMLは、1998年の末からW3C内で精力的に開発されてきた新時代のHTMLで、1年以上の議論と作業の結果、正式なウェブ標準として最近公開されたものだ。XHTML1.0勧告の意義は大きい。これで、ウェブは新しい時代を迎えたといっても過言ではない。

まず、XHTML1.0にいたる経緯を簡単に述べておこう。インターネットにWorld Wide Webが登場してから現在まで、それを支えてきた技術はHTMLというマークアップ言語とHTTPというプロトコルであった。ウェブ初期の時代にはHTMLはシンプルなものだったが、

その後のいわゆる「ブラウザ戦争」でHTMLは大きく進化した。表現力は劇的に高まり、インターネット上のリソースに対する標準的かつ統合的なインターフェイスとしての地位を獲得したのはご存じのとおりだ。しかし一方、特定のブラウザ専用のタグが横行して、非互換性の弊害をもたらした。こちらのブラウザでは綺麗に表示できるが、あちらのブラウザではメチャクチャといった現象を経験された方は多いだろう。

ウェブの相互運用性を高め、ユーザーにとって快適な環境を維持するためには、ブラウザベンダーによらない中立なHTML仕様が必要だった。W3Cはこのための作業を積極的に行い、HTML3.2(1997年7月)、そしてHTML4.0(1997年12月、1998年4月に改

版)を生み出した(図1)。現在では、HTML4.0は信頼できる基準とみなされている。非互換性が完全になくなったわけではないが、HTML4.0の範囲内でHTML文書を作れば、たいいていブラウザで満足のいく表示が得られる。

紆余曲折があったとはいえ、多彩な機能と能力を獲得しつつ標準化も進められ、HTMLは順調に進化してきたと言えるだろう。とはいえ、HTML4.0からさらに先に進むには困難な状況になっている。HTML4.0は、上手にまとめ上げられた仕様だが、それでもツギハギであることは否めないし、そもそものHTMLの基礎技術であるSGMLがあまりしっかりしたものではない。それに、これ以上機能を詰め込むと巨大な言語となり、ブラウザを作るのも、HTMLページを作るのも大変なことになる。

HTML4.0の作業とほぼ時を同じくして、W3Cでは、新しいマークアップ基礎技術の開発も行っていた。1998年2月、その基礎技術が公開された。そう、XML1.0である。XML技術を使えば、HTMLは新しい力と可能性を獲得するだろう、多くの人がそう考えた。そして、このプランを実行に移した結果がXHTML1.0なのである。

XMLとHTML、XHTMLの関係

XMLとHTMLの関係は、いまだにハッキリとしないと感じている方も多いだろう。まず、「XMLがHTMLに取って代わる」とか「XMLは次世代のHTMLだ」とかいう言い方は(それを責める気はないが)間違いであることを断っておこう。HTMLは元来、SGMLと呼ばれるマークアップ基礎技術(メタ言語)の応用である。そしてXMLは、SGMLをリニューアルしたマークアップ基礎技術である。SGML、HTML、XMLの三者の関係がわかれば、XHTMLの位置付けは必然的に明らかになる。SGML:HTML=XML:XHTMLという等式で考えるとわかりやすいだろう(図

TECHNOLOGY

2) つまりXHTMLは、HTMLをXMLベースで定義し直したマークアップ言語なのだ。

XHTMLはHTMLの対応物であるから、今までHTMLが果たしてきた役割を、XML時代にはXHTMLが担うことになる。つまり、本当の意味で「HTMLの次世代版」なのである。もちろん、HTML4.0との互換性は注意深く考慮されている。また、さまざまなXML技術との連携に向けての第一歩も踏み出したことになる。いままで、HTMLがウェブを引っ張ってきたのと同様に、XHTMLは今後のウェブの中心的存在となるだろう。

タグは小文字で省略せずに書く

HTML4.0にある程度慣れてる人なら、XHTML1.0への移行は簡単だ。XHTML1.0で新たに追加されたタグや属性は一切なく、HTML4.0とXHTML1.0は高い互換性を持っている。HTMLとXHTMLの相違点は以下のようなになる。

1. [XML宣言]

先頭に<?xml version="1.0" encoding="文字コード"?>を入れる。文字コードは、UTF-8、Shift_JIS、ISO-2022-JPなどが入る。

2. [小文字使用]

HTMLでは、タグ名や属性名は大文字で書いても小文字で書いてもよかったが、XHTMLでは必ず小文字を使う。

3. [タグを省略しない]

<p>やなどの終了タグ(</p>、)を省略しない。また、<head>、<body>などのタグもすべて書く。

4. [空要素タグ]

内容を持たないタグは、
、<hr />のように書く。属性が付く場合にも、<meta name="Author" content="M.HIYAMA" />となる。

5. [引用符使用]

XHTMLでは属性値に必ず引用符を付ける。

(誤)

(正)

XHTML対応のソフトウェア

先に述べた規則を守れば、テキストエディターを使ってもXHTML文書を作れる。いまでも特別なツールを使わずにHTMLタグを書いてきた人なら、たいした苦労はないだろう。現時点ではHTMLエディター（ホームページ

```
<?xml version="1.0" encoding="utf-8"?>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>XHTML 1.0 Document</title>
<meta name="Author" content="M.HIYAMA" />
</head>
<body>
<h1>XMLのここが好き</h1>
<hr />
<p>XMLのいい点</p>
<ul>
<li>仕様がコンパクト</li>
<li>シッカリしている</li>
<li>何にでも使える汎用性</li>
<li>単純だが奥が深い</li>
</ul>
</body>
</html>
```

リスト2 XHTML文書の例

```
<HTML>
<TITLE>HTML4.0 Document</TITLE>
<META name=Author content=M.HIYAMA>
<H1>XMLのここが好き</H1>
<HR>
<P>XMLのいい点
<UL>
<LI>仕様がコンパクト
<LI>シッカリしている
<LI>何にでも使える汎用性
<LI>単純だが奥が深い
```

リスト1 HTML文書の例

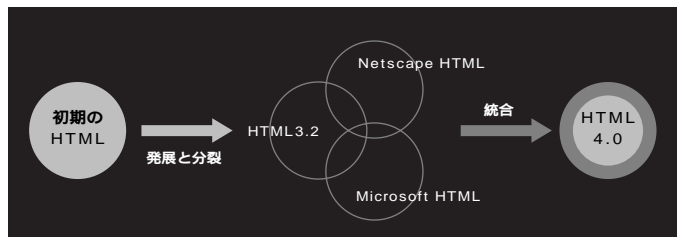
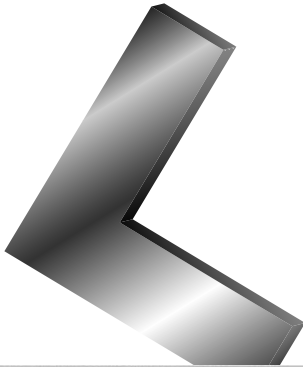


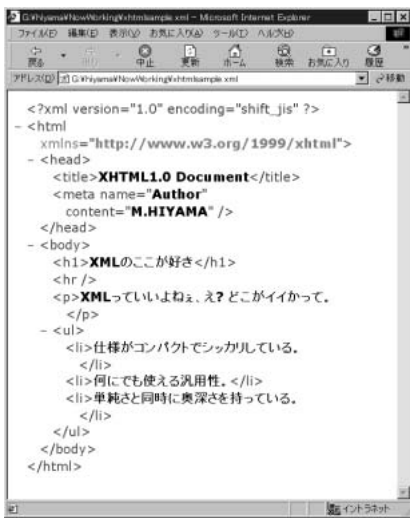
図1 HTMLのこれまでの経緯



図2 SGMLとHTML、XMLとXHTML



画面 3 : 「一太郎 Ark」による HTML XHTML 変換



画面 4 : 「.xml」拡張子を付けると、IE5 はツリー表示してしまう

作成ソフト)でXHTMLに対応したものは少ないが、とりあえず既存のHTMLエディターでHTMLファイルを作り、それを変換プログラムを使ってXHTMLにする方法もある。

1999年末にリリースされたジャストシステムの「一太郎 Ark」(Jump01)は、デフォルトでXHTMLファイルを扱うようになっている。名前に「一太郎」と付いているが、従来のワープロソフト「一太郎」とはまったく別物である。100% Pure Java アプリケーションで、XMLエンジンにXHTMLワープロとしてのGUIを被せたソフトウェアである。いまのところ、XHTMLを扱うソフトウェアとしては最も実用性が高いものだろう。「一太郎 Ark」は、HTML4.0のファイルを読み込んでXHTML1.0形式で保存できる。文字コードの変換も自由に行えるので、HTML4.0 XHTML1.0変換ツールとして利用することもできる(画面3)。メニューとダイアログを操作することになるので、大量のファイルの変換には手間取る。しかし現在、一太郎Arkの内部構造とプラグイン作成法がArkホームページで公開されつつあるので、Javaプログラミングに自信があれば、バッチ処理も可能かもしれない。

また、HTML文書をXHTMLに変換する「HTML TIDY」(Jump02)というソフトウェアもある。TIDYはウィンドウズ用のコマンドラインツールなので、MS-DOSプロンプトから次のようなコマンドで変換できる。

```
tidy -indent -asxml MyHTML.html > MyXHTML.xhtml
```

HTML TIDYは、オプションや設定ファイルでより細かい変換の指定もできる。たとえば、--word2000 yes というオプションで、Word2000で出力したHTMLファイルに対処しようと努力する(完全にうまく変換できるわけではない)。ただし、HTML TIDYは、UTF-8とISO-2022-JP以外の日本語文字コードに対応してないので、シフトJISなどの場合には文字コード変換プログラムと組み合わせ

せる必要がある。

作ったXHTMLを表示するには、IE5を使うえばよい。ただし、メディアタイプを「text/html」とするか、ファイルの拡張子を「.html」または「.htm」にしておかないと、XMLファイルとしてツリー表示をしてしまう(画面4)。もともとIEはXMLの表示機能を持っているので、はっきりとHTMLであると教えてあげないと、HTML表示機能がオフになってしまうのだ。古いブラウザを考慮したXHTMLの書き方もW3Cは発表している。このガイドラインに従ったXHTML文書なら、ネットスケープナビゲーターやIE4、あるいは他のブラウザでも問題なくXHTMLを表示できる。

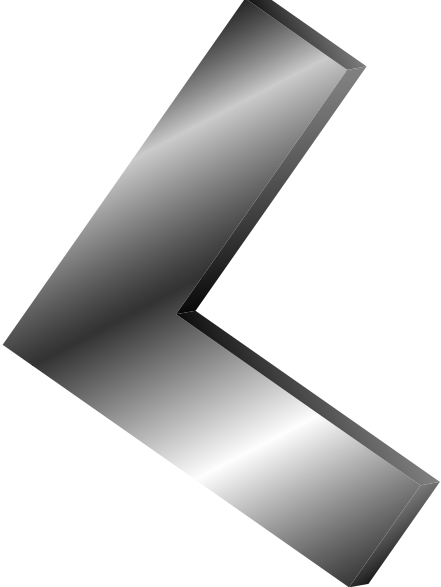
Jump01 www.justsystem.co.jp/ark/

Jump02 www.w3.org/People/Raggett/tidy/

モジュール化が今後のキーワード

XHTML1.0はHTML4.0とよく似ていることがわかった。本質的に新しい機能もない。それでは、XHTMLにどんな意義があるのだろうか。この疑問に答えるヒントがW3Cのページ(Jump)にある。今回勧告となったXHTML1.0以外にも、次のようなタイトルの技術資料(Technical Report)が出ている(図5)。これは、XHTML1.0勧告は出発点にすぎず、今後XHTMLをめぐる新技術がいろいろと予定されていることを示している。

このタイトル一覧を見るとモジュール(module)という言葉が目につく。実際、モジュールはこれからのマークアップ技術のキーワードである。モジュールと呼ばれる比較的小さな部品単位でタグセットを作る。それらのモジュールを組み合わせてマークアップ言語を作ろうという発想である。モジュールとしては、たとえば、箇条書きのモジュール、表のモジュール、ハイパーリンクのモジュールなどがある。XHTMLをモジュールベースにすることにより、次に示すようなさまざまな応用が考えられる(図6)。



1. XHTMLのモジュールを他のマークアップ言語のために利用する。たとえば、社内文書用の独自マークアップ言語に、XHTMLの表モジュールとハイパーリンクモジュールを使う、など。
2. いろいろなモジュールをXHTMLに付け加えて、XHTMLを拡張する。たとえば、数式を記述するためのMathMLモジュールを足して、数学論文用XHTMLとする、など。
3. 不必要なXHTMLモジュールを取り除いて、XHTMLのサブセットを作る。たとえば、日本人のメンバーが中心となって開発しているXHTML Basicはそのようなサブセットである。

現在、HTMLは、ウェブの標準形式であるにとどまらず、オフィスアプリケーションでもHTMLで入出力するものが増えているし、電子帳票（ビジネスフォーム）、データベースフロントエンド、グループウェアなどのデータとしても活用されている。さらには、携帯電話やテレビなどにもHTML類似のマークアップ言語が応用される。このような時代になると、HTML4.0のような単一の標準でHTML応用をすべてカバーするのは難しくなる。そこで、モジュールベースXHTMLの出番となる。モジュールベースXHTMLは、いわばXHTMLファミリーを作り出すための枠組みと素材を提供するのだ。そこからは、卒業論文用のHTMLとか、商取引用のHTMLとか、携帯電話向けHTMLとかが生み出されることになる。

 www.w3.org/TR/

XHTML 1.1、そしてその先へ

XHTML 1.0が勧告になったばかりだが、すでにXHTML 1.1がドラフトとして出ている。XHTML 1.1の仕様自体は短いものだが、ここから別な仕様を参照することで実質的にはモジュールベースXHTMLについて記述している。

XHTML 1.1が完成すれば、XHTML 1.0で基礎付けた方向へとさらに前進することになる。マークアップ技術がモジュールベースにな

れば、ブラウザなどのXHTMLを扱うプログラムも、マークアップモジュールと並行した構造を持つことになるだろう。たとえば、基本的なHTML機能を持つブラウザに図形表示とタイミング制御のモジュールを付け加えてアニメーション機能を実現することなどが可能だろう。

HTML大成功の歴史、そしてXML技術の奥深さを考えると、XHTMLがオフィスアプリケーションや情報システム一般にまで広く利用されるのは間違いのない。現在でもすでに、多くのことがHTMLブラウザだけでできるようになっている。XHTMLでは、拡張、縮小（サブセット）、カスタマイズが自由にできるから「HTMLブラウザがアプリケーションだ」という傾向がさらに強まるだろう。しかし、単一のHTMLブラウザが何でも担当するわけではなく、マークアップモジュールとプログラムモジュールの組み合わせが必要な機能を提供する、そんな世界が実現すると期待できる。

- XHTML Basic
- XHTML 1.1 - Module-based XHTML
- Modularization of XHTML
- Building XHTML Modules
- XHTML Events module
- XHTML Document Profile Requirements
- XHTML Extended Forms Requirements

図5 XHTML関連の技術資料

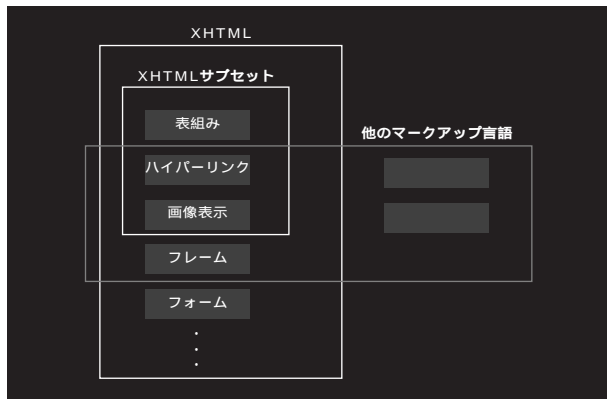


図6 モジュールベースXHTMLの応用





[インターネットマガジン バックナンバーアーカイブ] ご利用上の注意

このPDFファイルは、株式会社インプレスR&D(株式会社インプレスから分割)が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

<http://i.impressRD.jp/bn>

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容(技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど)は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者(執筆者、写真の撮影者、イラストの作成者、編集部など)が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元(株式会社インプレス R&D)、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D(当時は株式会社インプレス)と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接のおよび間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先

株式会社インプレスR&D

All-in-One INTERNET magazine 編集部

im-info@impress.co.jp