

Webを支える情報通信技術 (2)

——Webでのデータの表現形式——

情報基盤センター
佐藤周行・中山雅哉・西村健

1. はじめに

今回の課題は Web でのデータの表現形式である。キーワードは HTML と XML。

ふだん、われわれは Web ブラウザを通して Web を見ている。目の前には派手なグラフィックスなど、視覚効果満載のウィンドウが出てくる。その裏側を少しだけのぞいてみよう。

2. HTML

まず、どれでもいいから、Web のページを開いてみる。たとえば、以下は 3 月 27 日の午後にとった <http://www.u-tokyo.ac.jp/> にアクセスすると出てくる画面である。

(<http://www.u-tokyo.ac.jp/> の画面)

ブラウザにもよるが、一般的に視覚に訴える形で表示されることがほとんどである。では、どのようなプロセスを経て表示されるのだろうか？

とりあえず、大枠として以下のことを押さえよう。

1. まずブラウザがある。
2. ブラウザが www.u-tokyo.ac.jp というサーバにアクセスする。
3. サーバはブラウザにデータを送りつける。
4. ブラウザは、もらったデータを「解釈する」¹⁾

問題：

- Q 1. <http://www.u-tokyo.ac.jp/> ってそもそも何？
- Q 2. 2での「アクセス」って何？
- Q 3. 3での送りつけられるデータって何？

ここでは、Q 3について考えることにする。Q 1 と Q 2 は大きな問題なので、5 月に入ってから徹底的に考察・実験をすることにしよう。まずは Q 3 である。

ブラウザは、アクセスした先から HTML 形式で書かれたデータを受け取る。データの生の形は、「ソースの表示」またはそれに類した命令で見ることができる。

¹⁾ 「解釈」の具体的な内容は後で触れるので、とりあえずこの抽象的なことばで我慢する。

(ソース画面)

HTML の形式

では、HTML とは、どのように書かれているのだろうか。表示と対応させながら見るとよい。ここでは、ごく単純に書かれたものをみてみよう。対象は <http://www.ipsj.or.jp/sig/pro/> とする。

(ソース画面)

ここで特別な意味を持つものは<と>でかこまれたものである。これをタグという。

例：<HTML>, <HEAD>

タグを置くことによって、それが「はじまる」ことを指定する。</と>で囲まれたものは、そのタグが終わることを示す。名前の次に、いろいろな「属性」を指定することができる。

例：関連情報

ここで href が属性名になる。タグに付随していろいろな属性が定められている。

HTML の意味

ブラウザはデータを HTML 形式で受け取った後、それを解釈して文書を整形²し、最終的にウィンドウに表示する。たとえば、同じソース中にあるタグとしてを考える。これを受け取ったらブラウザは、「その部分をクリックしたら <http://www.ipsj.or.jp/08edit/dc/index.html> にアクセスしに行く」ような表示をしなければならない。また、に続く内容を受け取ったら、それを箇条書きにして表示しなければならない。このようにタグにひとつひとつ意味を決めていくことにより、最終的な表示が得られるのである。

ところで、われわれは Web ページを自分で作る場合でも、HTML を意識することはほとんどなくなった。Web オーサリングツール(HTML エディタ)が関連することをほとんどすべてやってくれる。

問題：

Q 4. 自分の知っている(使ったことのある) Web オーサリングツールをあげてみよ。

3. XML

ところで、データの表現形式として標準的なのは HTML だけではない。

² ここでいう「文書の整形」とは、文章の配置や、フォントの大きさなどを調整してきれいにみせられる形にすること、程度の意味。

<http://www.w3.org/2000/08/w3c-synd/home.rss> にアクセスしてみる。

(<http://www.w3.org/2000/08/w3c-synd/home.rss> の画面)

ところがこれは HTML 形式ではない。このソースを見してみる。

問題

- Q 5. ソースコードを何とか表示してみよ。
- Q 6. ソースコードを何とかわかりやすい形で表示できないか試みよ。

ソースコードの形式をじっと見ると、HTML のものとは若干異なっていることに気づく。最初が<?xml ... ではじまる形式を XML 形式という。ほとんどすべてのブラウザは XML 形式を受け付ける。たとえば、<http://slashdot.jp/slashdot.rdf> をみてみよう。

(<http://slashdot.jp/slashdot.rdf> の画面)

とりあえず、このゼミナールでは XML 形式については深入りしない。ただし、今あげたものが RSS であること、RSS は、人間を直接相手にするのではなく、ソフトウェアを相手にすることをおぼえておこう。

問題：

- Q 7. 拡張子を xml にしてページをファイルに保存し（この意味わかりますか？）、そのファイルのアイコンがどうなるか観察せよ。
- Q 8. RSS って何？
- Q 9. RDF って何？

ところで、XML を使って計算だってできるんだよ。知ってた？

問題：

- Q 10. 添付の inp.xml と fac.xsl を自分のディレクトリにコピーし、inp.xml を表示してみよ。または <http://www-sato.cc.u-tokyo.ac.jp/PKI-project/Komaba/inp.xml> にアクセスしてみよ。以下になったら正解。

