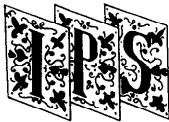


解説

World-Wide Web (WWW)[†]益岡竜介^{††} 木庭袋圭祐^{†††}

1. はじめに

本稿では World-Wide Web (WWW) について、その特徴、歴史、技術、ホームページのデザイン、課題と将来を解説する。

現在 WWW の解説はすでに数多くあり、それらの情報が溢れている。他の文献でも多く紹介されているような WWW の一般的なアーキテクチャについての解説は必要最小限に抑えているため、詳細な情報については文献 1), 2) などを参照されたい。

なおこの文書中で現在とは 1995 年 2 月 28 日を意味している。WWW の変化の勢いを考えてあえて明示しておく。またこの解説は HTML 文書に変換したものを以下の URL に置く予定である。是非一度アクセスして見ていただきたい。
http://www.fujitsu.co.jp/people/masuoka/www_paper/index.html

2. 現状と特徴

近年のインターネットおよびその中心的な枠組の 1 つである WWW などの盛り上がり方には目覚ましいものがある。コンピュータ雑誌はもちろんのこと、新聞、テレビ、一般雑誌、週刊誌の多くがインターネットや WWW の特集を組んでいる。もちろん各種メディアがただ騒いでるだけでなく、学術関係者のネットワークであったインターネットそして WWW が着実に一般に広がり始めている。インターネットへのアクセスを提供する Network Service Provider (NSP) が増え

ている。WWW のホームページを持つものも、以前のコンピュータ関係の大学の学科や企業から一般の大学や企業、そして個人へと裾野が広がっている。WWW はこのように大きく成長しつつあり、同時にそれを取り巻く環境にも大きな影響を与えている。

ではこの WWW とは何か。一言でいえば以下のようなになる^{*}。WWW は 1989 年に CERN (Organization [旧 Conseil] Européenne pour la Recherche Nucléaire) の Tim Berners-Lee 氏らによって提案されたハイパテキストの概念に基づく分散型情報システムである。主な特徴として次のものがあげられる。

- インターネットに散在する情報をハイパテキストの概念で整理する。
- ハイパテキスト情報をやりとりするための手段を提供する。
- 様々なリソースを指定するための統一的手段を提供する。

以下では WWW の表面的な特徴、すなわち WWW クライアント側から見た特徴を WWW の実際の操作の雰囲気とあわせて紹介する。

WWW の表面的な主な特徴は netsurf の感覚とマルチメディア性とオープンであることにまとめられると考える。たとえば、図-1 は富士通のホームページ (URL は <http://www.fujitsu.co.jp/>) に WWW クライアントである Infomosaic でアクセスした状態である。この中のアンダーラインが引かれている青い文字列をマウスでクリックすると、その文字列に関連した情報が表示される。たとえば「富士通紹介」の文字列をクリックすれば、「富士通紹介」のページが表示される。

[†] World-Wide Web (WWW) by Ryusuke MASUOKA (Netmedia Laboratory, Multimedia Systems Laboratories, FUJITSU LABORATORIES LTD.) and Keisuke KIBAKURA (Research & Planning Division, IT Strategy & Planning Group, FUJITSU Ltd.).

^{††} (株)富士通研究所マルチメディアシステム研究所ネットメディア研究センター

^{†††} 富士通(株)企画本部企画部

^{*} 言葉上の理解だけで WWW の可能性を掴み切るのは難しい。まだ実際に WWW に触れたことがない方がいたら、これを機にぜひ触れてみていただきたい。

文字列だけではなく、青い枠で囲まれた画像*あるいは画像の一部分をクリックすることにより、その画像あるいは画像の部分に関係した情報が表示される。これらの青で示された部分が、他の情報への“リンク”になっている**。リンクをたどり、次の情報を表示している部分に再びリンクがあれば、またそのリンクを選ぶことにより、情報から情報へと、次々とアクセスしていくことができる。このようにインターネットの情報の海をsurfingしているような感覚がnetsurfと呼ばれている**。

リンクの先は必ずしもWWWサーバ上の文書でなくてもよい。音声、静止画、動画、科学データ、MIDIデータなど文書以外のデータや、Gopher, FTP, WAISなどネットワーク上の他のサービスを指定することもできる。ユーザ側に各々のデータに対応したビューアを準備することで、これらのすべてのリソースをWWWという統一された世界で扱える。マウスクリック1つで音楽が始まったり、映像が始まったりするところは「まさにマルチメディア」である。

さらにWWWサーバへのアクセスはプラットフォームを選ばない。Unix Workstation, PC, MacintoshのいずれにもWWWクライアントがあり、イリノイ大学のNCSA (National Center for Supercomputing Applications) で開発されたMosaicやNetscape Communications社のNetscapeが有名である。またキャラクタベースのWWWクライアントも多くあり、状況によってはグラフィカルなものより便利である。

以上のnetsurfの感覚とマルチメディア性、オープンであるなどのWWWの表面的な特徴に加え、以下で説明するWWWの強力な枠組により、従来個々の特定の計算機に閉じ込められてきた情報を簡単にインターネットに提供することが可能になった。これらのことが組み合わさりWWWの急速な発展につながったと筆者らは考えている。

3. 歴 史

個人的な見解により現在までのWWWの歴史を以下のような3つの期間に分類する。まず1989年にCERNのTim Berners-LeeがWorld-Wide Web Projectを提案してから、NCSAがMosaic 1.0 for X windowsをリリースする1993年6月までを黎明期と呼ぶ。その1993年6月から、NCSAからその開発の中心的なメンバを引き抜いてMosaic Communications社(現在Netscape Communications社)が設立される1994年4月までを膨張期と呼ぶ。そして1994年4月以降を戦国期と呼ぶ。以上の分類にしたがって、WWWの発展の歴史を見ていく。

WWWの黎明期においてはCERNがWWW発展の中心的な存在であった。もちろんWWW以前にもハイパテキストを使ったシステムは考えられていたが、ほとんど研究レベルに留まっていた。WWWが発展した理由は世界中がちょうどインターネットで結ばれはじめていたという時期がよかったということとともに、CERNでのWWWプロジェクトが以下のように具体的かつ明確な目的を持っていたことであると考えている。

CERNのように加速器を使って高エネルギー物理学を研究するようなBig Scienceでは、世界中から研究者が集まって実験を行い、実験が終了すれば研究者たちは世界中に散らばってしまう。このように世界中に分散する研究者たちの間の共同研究を円滑に進めるための枠組として、1989年Tim Berners-LeeによってWorld-Wide Webプロジェクトは提案された*。

WWWは徐々にだが確実に成長していった。黎明期ではもっぱらLine Mode(すわなちキャラクタベース)のWWWクライアントが中心であった。telnetで“info.cern.ch”にアクセスしてCERNで走っているLine ModeのWWWクライアントを使うこともかなり一般的であった**。

1992年の秋頃にはNCSAの人たちがWWWに注目し、Mosaicの開発をはじめた。NCSAは

* たとえば「富士通紹介」の文字列の左にある地球に“FUJITSU”と書いてある画像が青い枠で囲まれている。

** リンクの表現はWWWクライアントによって異なる。また一度アクセスしたリンクの表現を変える場合が多い。

** しかしあまりに大きな海に溺れるものも続出している。

* ちなみに日本の最も古いWWWのサーバの1つは筑波にある高エネルギー物理学研究所(KEK)のWWWサーバ(<http://www.kek.jp/>)である。

** 現在はこの方法でWWWにアクセスすることはできなくなっている。

全米に5つある Supercomputer Center の1つとして Supercomputer を遠隔地から使ってもらおうという形で存在していて、CERN と似た立場にあった。NCSA では各種ツールを作って*分散した人々の共同研究を進めようとしていた。WWW はそのような目的にちょうど合致し、Mosaic の開発に至った。

そして1993年6月、Mosaic の最初の正式なリリースである Mosaic 1.0 for X Window が発表され、膨張期に入る。Mosaic はそのグラフィカルなインタフェースと幅広い UNIX ワークステーションのプラットフォームをサポートしたことにより、一気にインターネットのコミュニティに広がっていった。さらに1993年の秋には Windows 版、Macintosh 版をリリースし、さらに WWW の裾野を広げていった。この頃にはインターネットのコミュニティだけではなく、一般の注目も集めるようになっていった。これ以降 WWW の世界は急速に膨張していった。

Mosaic が出てからは WWW の発展の中心はその重心を NCSA に大きく移した。もちろん CERN も中心の1つであったが、NCSA は次々と新しいバージョンの Mosaic を出し、この頃の WWW 発展の原動力となった。

しかし Netscape Communications 社にその開発の中心的なメンバを引き抜かれた1994年春以降、NCSA はその求心力を徐々に失っていった。Mosaic のサポートを続けてはいるが、新しいソフトの出るペースも落ちてしまった。やはりこういった技術をリードするためには、新しい技術がある程度素早く形にする開発力も必要である。NCSA が求心力を失って以来、WWW に関してはいろいろな動きが出て、いわば戦国時代のような様相を呈している。

著者らは現在では CERN と MIT が1994年10月に設立した W3C (World - Wide Web Consortium)³⁾ と IETF (Internet Engineering Task Force) がこれからの WWW の発展の方向を握るのではないかと見ている。ちなみに W3C の Director は CERN で最初に WWW を提案した Tim Berners-Lee であり、この組織は X Consortium をモデルにしている。IETF はイ

ンターネットにおける標準化に関して中心的な役割を果たしており、HTTP, HTML, URI の標準化を進めている。これら以外に企業としては Netscape Communications 社や、CommerceNet* の中心的な企業の1つである EIT (Enterprise Integration Technologies) 社などが WWW に深く関わっている。

4. 技 術

4.1 WWW の枠組

WWW の枠組は中心的な概念として URL, HTTP, HTML, CGI などによって形作られている**。WWW はこれらの強力な枠組によってインターネット上における情報の共有を可能にした。以下では WWW の枠組の全体を概観した後、個々の要素を説明する。CGI については特に WWW の動的な側面として章を分けて解説する。

WWW システムの全体の構成は図-2 のようになる。URL はインターネット上の様々な資源を統一的に表記するための記法である。HTTP は WWW のクライアントとサーバがやりとりするためのプロトコルである。HTML はハイパテキストを表現するための言語である。

WWW クライアントは URL によって指定された情報を、HTTP を用いて、HTML 文書を取得して、画面に表示する。その画面上でさらにユーザがリンクを選択すると、そのユーザのアクションは URL という形で WWW クライアントに伝わり、続くサーバとの対話を引き起こす。

ユーザによって選択された URL が指す情報が他のサービスのテキスト系のデータであった場合には、それらのサーバが使う通信プロトコル (FTP サーバなら FTP) を使ってその情報を取得する。情報はそのサービス固有の形式を持って

* CommerceNet とは、米国 San Francisco の近辺 (Bay Area) の企業を中心となり始めた競争力強化の組織 Smart Valley 公社の活動の一環として、インターネット上の高速企業間ネットワークを使い電子市場を作る計画のことである。将来は企業と個人間のネットワークにも展開する予定である。

** ここでは解説はしないが、その他の WWW の重要な要素として proxy server がある。これは fire wall を通じた外部へのアクセス、キャッシュによるネットワークの負荷軽減、日本語のコード変換などを実現する。

* Macintosh からインターネットで Supercomputer にアクセスできるようにした NCSA telnet などがある。

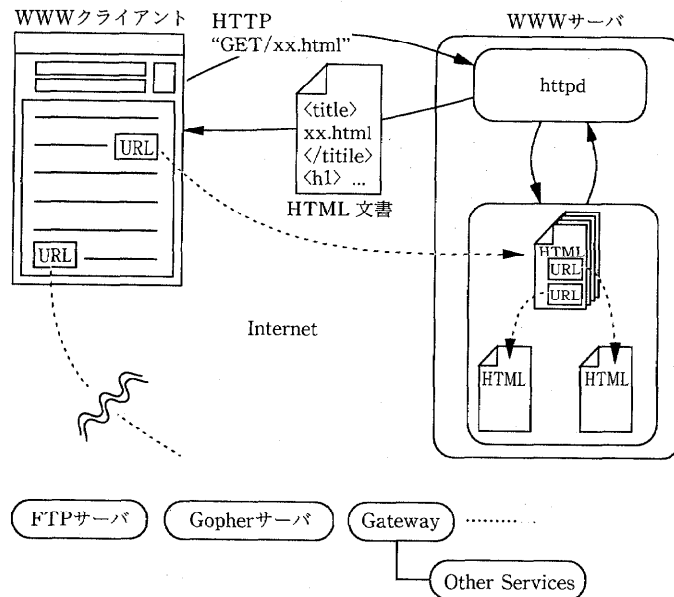


図-2 WWWの全体の構成

いるが、HTMLに変換され^{*}、画面に表示される。

URLが指す情報が画像や音声などのバイナリデータの場合には、そのデータを表示できる外部ビューアに渡して起動する。この仕組みによってユーザがリンクを選択すると動画や音楽が始まったりするのである。

一部のサービスはゲートウェイによって実現されている。実際の情報は別のサーバに格納されており、問合せと応答がゲートウェイで中継されるという仕組みであり、CGIで実現されていることが多い。クライアント側からはHTTPでアクセスできるため、ユーザにはWWWに統合されたサービスのように見える。

URL (Uniform Resource Locator)

URLとはインターネット上の様々なリソースを統一的に記述するための記法である。URLはリソースを特定するために、ホスト名、ファイル名、アクセスに用いるプロトコル、アクセス時のコマンドなどの要素を取り込んだ名前空間を作ろうというものであり、以下のような形式を持っている。

`scheme://host.domain[:port]/path[?key]`

^{*} ここでハイパテキストであるHTMLの表現力が生きる。計算機で使われているたいの情報の形式はHTMLで表現できる。

この *scheme* 部には http, gopher, ftp, wais, nntp, telnet, file, などの様々なプロトコルを指定することができる。具体的なURLの記述例を次にあげるので参考にされたい。

- WWWサーバ `www.ncsa.uiuc.edu` 上のホームページ `index.html` を示すURL:

`http://www.ncsa.uiuc.edu/index.html`

- FTPサーバ `ftp.iij.ad.jp` 上のファイル `/pub/ls-lR.Z` を示すURL:

`ftp://ftp.iij.ad.jp/pub/ls-lR.Z`

HTTP (HyperText Transfer Protocol)

WWWではクライアントとサーバが通信するためのプロトコルをHTTPとして規定している。ハイパテキストは主にこのプロトコルにしたがってやりとりされる。基本的には、クライアントがサーバへリクエストを送信し、それに対してサーバがクライアントへレスポンスを返信するという形式になっている。

リクエストはたとえば次のような形をしている。

`GET /index.html HTTP/1.0`

最初のGETがメソッドと呼ばれるもので、対象となる情報に対して行われる処理を示す。他のメソッドはHEAD, PUT, POST, DELETEなどがあるが、実装の度合はサーバにより異なるのが現状である。2番目の“/index.html”がクライア

ントがそのサーバに求める情報を指す URL である*。最後の“HTTP/1.0”は使用する HTTP のバージョンを示す。リクエストには、この後に RFC 822 および RFC 1521 (MIME) のフォーマットに従ったリクエストヘッダがついたり、POST メソッドを使うときにはデータが付属する。

サーバがクライアントに返すレスポンスは、たとえば次のような形をしている。

```
HTTP/1.0 200 Document follows
MIME-Version: 1.0
Server: CERN/3.0
Date: Friday, 24-Feb-95 11:25:26 GMT
Content-Type: text/html
Content-Length: 1808
```

```
<HTML>
```

```
... (中略) ...
```

```
</HTML>
```

最初の一行がステータスラインと呼ばれるものでリクエストを解釈した結果をクライアントに知らせている。上の例ではリクエストが成功したので、この行の後にリクエストされた情報が返送されることを示している。その後ろには返送される情報に関する MIME ヘッダが付けられており、さらにその後ろには空行に続けて実際のデータ（この場合は HTML 文書）が転送される。

HTML (HyperText Markup Language)

WWW で用いられるハイパテキストを記述するための言語が HTML である。HTML は SGML** の DTD (Document Type Definition) によって定義された言語であり、文章の要素を表すのに SGML の記法に従ったタグを用いる。すなわち `<element-name>...</element-name>` のような表現形式である。以下に HTML 文書の簡単な例をあげる。

```
<HTML>
```

```
<HEAD>
```

```
<TITLE> A simple example</TITLE>
```

```
</HEAD>
```

* proxy server に対してでない通常のアクセスでは、このように URL は *scheme* や *host* を省略した相対的な記法で与える。

** Standard Generalized Markup Language: 文書の論理構造を記述するための言語。1986年に ISO 8879として制定された。

```
<BODY>
```

```
<H1> A simple example</H1>
```

```
Welcome to<EM> WWW</EM> world.
```

```
<P>
```

```
<UL>
```

```
<LI><A HREF="http://www.fujitsu.co.jp/">
```

```
FUJITSU LIMITED. Home Page</A>
```

```
<LI><A HREF="http://www.ntt.jp/">
```

```
NTT Home Page</A>
```

```
</UL>
```

```
</BODY>
```

```
</HTML>
```

これを WWW クライアントで表示させると、タグの情報をもとにフォーマットされた文書として見るができる。各クライアントにより差異があるが、Infomosaic では図-3 のように表示される。タグに用いられるアルファベットは大小文字の区別が無視される。ここではもっとも重要なタグ `<A>`、アンカー (Anchor) について説明する。このアンカータグは、そのタグで示された文字列 (あるいは画像) をアンカーとするものであるが、その“HREF”属性に URL を指定することにより、そのアンカーを“リンク”とすることができる。このアンカータグによってはじめて HTML 文書はハイパテキストたり得る。

4.2 CGI: WWW の動的な側面

この章では、WWW の動的な側面としての CGI (Common Gateway Interface) を具体的な例を使って紹介していく。

HTTP で WWW サーバに

```
GET /cgi-bin/command/path HTTP/1.0
```

とリクエストを渡すと WWW サーバでは、環境変数“PATH_INFO”に“/path”が設定されて、“command”が実行される*。その command を実行して標準出力に出てきた結果をアクセスしてきた WWW クライアントにデータとして返す。このコマンドと WWW サーバの間のやりとりを決めたものが CGI である。

この CGI により WWW サーバにアクセスしてきた WWW のクライアントに対して、“path”に

* 最初はコマンドに対する引数で“/path”を渡すようにしていたが、“/path”以外にもいろいろな種類の情報を渡すために、結局環境変数でコマンドに情報を渡すようになった。



図-1 富士通の Home Page



図-3 簡単な HTML 文書の表示

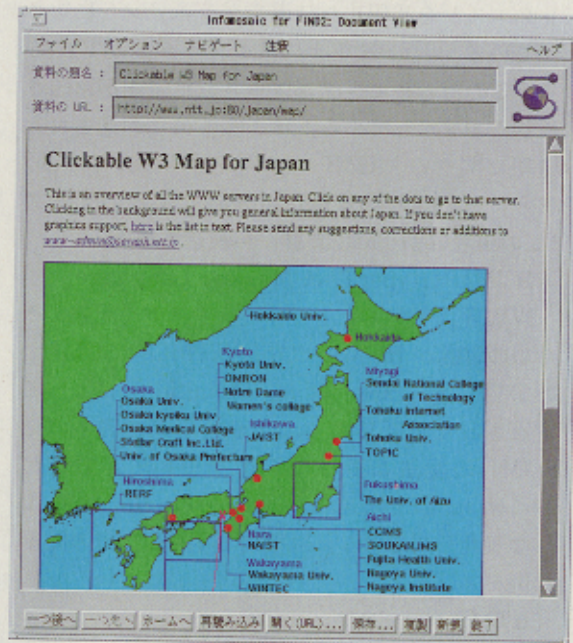


図-4 “Clickable W3 Map for Japan” のページ



図-5 15 Puzzle のページ



図-6 FAX ゲートウェイのページ

対応する実在のファイルとは関係なく、“path”を WWW サーバ側で解釈して、その解釈した結果で動的に HTML 文書を生成してクライアント側に返すことができる。このことを本稿では WWW の動的な側面と呼んでいる。WWW では

こういったことが容易にでき⁹、それが WWW を面白くし、可能性を開いた大きな理由の一つだと考えている。

Image Map と URL Redirection

図-4 は NTT の WWW サーバにある“Clickable W3 Map for Japan”のページ (URL は <http://www.ntt.jp/japan/map>) である。この日本地図が Image Map になっていて、クリックする位置によって表示されるページが異なる。たとえば、ユーザが北海道大学の文字 (“Hokkaido Univ.”) の部分をクリックすると、WWW クライアントは北海道大学のホームページを表示する¹⁰。

これは以下のような仕組みになっている。クライアントは Image Map の URL の後ろに “?” とマウスでクリックされた位置の座標を付加して、次のようなリクエストを送信する

```
GET /cgi-bin/imagemap/www-in-JP?307,59 HTTP/1.0
```

サーバでは CGI コマンドである imagemap が起動され、その座標を解析し、ページが別の場所に移ったことを示す Location フィールドをヘッダに含んだメッセージをクライアントに返信する。クライアントはヘッダの Location フィールドを見て、北海道大学のホームページにアクセスする。

シーケンスの管理 (15 Puzzle)

CGI と Image Map を使うと 15 Puzzle といわれるものが作成できる。図-5 のようなページで、最終的に絵全体を復元することが目的であり、空いているグレーの枠に接しているいずれかの枠内をクリックすると、次はクリックした枠が空いている枠に移動したページが表示される。

このようなゲームでは、複数のクライアントが同時にアクセスする可能性があるため、アクセスする個人ごとのアクセスのシーケンスを管理する必要がある。その管理の方法は、サーバが返す HTML 文書中にそれぞれの状態に対応する数を入れておくものである。この場合は Image Map である絵をクリックしたら、その数がサーバに

⁹ Gopher でも可能であるが、WWW ほど容易ではない。

¹⁰ URL Redirection をサポートしていない WWW クライアントでは、代わりにハイパーテキストが表示される。この場合、ユーザはそこからリンクをたどって行くかたになる。

“path”の一部として返ってくるようにした。

具体的にはリンクになっている絵の部分はたとえば次のようになっている。(以下の例でwwwは仮のホスト名である。)

```
<A HREF="http://www/cgi-bin/puzzle/14035">
```

```
<IMG SRC="http://www/cgi-bin/puzzle_gif?14035" ISMAP></A>
```

ユーザが絵をマウスでクリックするとそのクライアントはwwwに対して、マウスがクリックされた座標を付加して以下のようなリクエストを送る。

```
GET /cgi-bin/puzzle/14035?231,80
HTTP/1.0
```

サーバ側では“14035”に対応する状態で座標(231,80)でクリックがあったということで次の状態のページを作り、クライアントに返す。もちろん次のページも状態に対応する番号が前のページと同じようにURLの中に入っている。

POST メソッド

WWWを情報をブラウズするだけの受動的なメディアから、より能動的なメディアにするには、クライアント側からもサーバに情報を送れることが本質的に必要である。

WWWの初期には、クライアントからサーバへ情報を送るためには、GETメソッドを使って、URLの中にその情報をエンコーディングしてサーバに送っていた。この手法は検索のキーワード程度を送るには十分であったが、やがてFORM*が出てきて文書まで送るようになるとURL自体がとてつもなく長くなり、実用上不便であった。このような背景があり、POSTメソッドが使われるようになった。このPOSTメソッドは、通常CGIコマンドと組み合わせて使われる。

以下にFORMとPOSTメソッドを使ったWWWとFAXのゲートウェイの例を説明する。FORMを使うとたとえば、図-6のようなFAXの送信紙のようなページができる。このそれぞれのフィールドに適当に入力をして、“Send”のボタンをクリックすると、WWWクライアントは

HTTPでWWWサーバに

```
POST /cgi-bin/fax-www-post
```

といった行から始まるリクエストを送る。リクエストヘッダの後にそれぞれのフィールドの情報をデータとしてWWWサーバに渡す。

WWWサーバによって実行されるCGIコマンドであるfax-www-postは、WWWクライアントから送られたデータを標準入力から以下のような形で受けとる*。

```
subject=Test+Fax+Message
```

```
&r-name=Keisuke+Kibakura
```

```
&r-affiliation=Fujitsu+Ltd.
```

```
&r-fax=044-XXX-XXXX
```

```
&s-name=Ryusuke+Masuoka
```

...

CGIコマンドであるfax-www-postはこの入力から必要な情報を抽出して、FAXプログラムを使ってFAXを送信する。

5. ホームページのデザイン

著者らとしてはぜひとも、より多くの方々にWWWを通じて情報発信を行っていただきたいと考えている。最近ではNSPを中心としてWWWのホームページを持たせてくれるサービスや、ホームページのデザインを手伝ってくれるサービスが出ている。個人向けのサービスならかなり安い金額で提供してくれるところも多い。こういったサービスを利用すれば、容易に情報発信を試みる事が可能である。

しかし現在すでに数多くの個人や組織のホームページがあり、ホームページを持つだけでは差別化は難しい。特に企業のホームページなどでは、かなりのコミットメントが要求され、中途半端に立ちあげるくらいなら、いっそ立ちあげない方がましなくらいである。個人のホームページでもあまりアクセスがないと寂しいものである。

以下に個人や組織がホームページを持つ際の要点をいくつかあげる。ここではホームページにアクセスする人のことをユーザと呼んでいる。

英語のページを持つ

世界へ情報発信するのだという気持ちで、ぜひ英語のページを持っていただきたいと考えて

* FORMはHTMLの要素である。さらにネストしたHTMLの要素(タグ)を持つことができ、それらのタグにより、ボタンや、ユーザがテキストを入れる入力フィールドや、文書を入力するテキストフィールドを提供することができる。

* 実際には1行で来ているが、文章の都合上分かりやすいところに改行を入れている。

いる。

全ページの統一性

一番最初のページがかなり凝っていても、他のページにいったらテキストだけという状態では貧弱な感じを与えてしまう。全ページに凝った絵を入れる必要はないが、すべてのページに統一的なデザインを与えることによって、ユーザが受ける感じは非常に変わる。

ユーザを引き付けるコンテンツ

ユーザは移ろいやすい。面白くなければ、あるいは役に立たなければ、すぐにアクセスを止めてしまう。そうすればホームページはただそこにあるだけで、さびれた街のやけに派手な看板のようなものである。ユーザを引き付けるための要点を以下にまとめた。特に最初の2つはリピータを確保するのに重要である。

- 常に新しいことを付け加えていく
- 役に立つ本物の情報を載せる
- 組織のある街の紹介などの遊びがある

情報発信の仕組みを作る

大きな組織で本格的なホームページを持ち、継続的にいい情報を発信し続けるためには、ページを统一的に管理し、各部署から情報を提供する、情報発信の仕組みを作る必要がある。

画像データ

画像を使ってホームページを作ることができるのが、WWWによる情報提供の大きなメリットであるが、適切な大きさの画像を使うように注意する必要がある。大きな画像は転送に時間がかかり、ネットワークにも負荷がかかる。また256色程度しか表示できないコンピュータも多いので、使う色数は抑えておいた方がよい。

キャラクタベースのWWWクライアントでアクセスしてくる人のためにキャラクタベースのページを用意する、あるいは画像がなくても理解できるようにしておくことが重要である。説明をなどとしてIMGタグのALT属性として入れておくと、ALTに指定した文字列がそのようなクライアントでは絵の代わりに表示される。

検索

ホームページ、あるいはそこからのリンクで、そのサーバにある情報が、タイトル、あるいはHTML文書などの全文にわたって、キーワード

などで検索できるようにしておくと、ユーザにとって非常に便利である。

6. WWWの課題と将来

以下ではWWWの課題と将来に関して、標準化の動向⁴⁾を中心に概観していく。

セキュリティ

インターネットの商業利用が企業の関心を集めているが、WWWを電子取引などの商業利用するためには、セキュリティを確保したHTTPを実現するのが本質的に必要である。この分野では、EIT社の提案するSHTTP (Secure HTTP) と Netscape Communications社の提案するSSL (Secure Socket Layer) が有望であるようだ。これらのどちらかあるいは両者を組み合わせたものがW3Cなどで採用されればそれが実質的な標準になると考えている。ただし暗号化の技術に関するアメリカの輸出規制などもあり、最終的にどうなるかを予測するのは難しい。

URI

URLに関してはIETFのURIワーキンググループで検討され、現在はRFC 1738⁵⁾にまとめられている。しかしURLでは文書の位置(どのマシンのどの"path"にあるか)やどのプロトコルでとってくるかに依存している。このことは、URLで指定されると、まったく同じ文書が近くにあっても、指定されたところにとりにいってしまうとか、置かれる場所が変わってしまうとアクセスできなくなってしまうなどの問題を持つ。

そこで文書の位置やプロトコルに依存しないURI (Uniform Resource Identifier) を実現しようとIETFのURIワーキンググループでURN (Uniform Resource Name) などが検討されている⁶⁾。具体的にはDNS (Domain Name System) と似た以下のような形で実現するようである。WWWクライアントなどでURNで文書にアクセスするとき何らかのサーバに対してURNをURLに具体的に解決してもらい、そのURLで文書を取得する。ここでURLがIP Address, URNがDomain Nameに相当する。このURNとURLを結びつけるリソースエンコーディングの情報を記述する方法はURC (Uniform Resource Characteristics) として検討されている。

HTML3.0

HTMLの初期のバージョンではインタフェースを損なわないために、処理に時間のかかる高性能な要素は少なかった。しかしWWWもこれだけ広がりいろいろな機能に関する要望も出てきたことや、計算機の処理速度の向上などにより、高機能化に向かう様子を見せている。

現在一般的に使われているHTMLであるHTML 2.0がInternet-Draft⁷⁾として公開されている。IETFではさらにHTMLに関する議論が続けられており、図、表、数式などがサポートされているHTML 3.0*のInternet-Draft⁸⁾も公開されている。

MBone との関係

最近MBone (Multicast Backbone) という新しい技術が出てきている⁹⁾。これはインターネット上の仮想的なmulticastのネットワークであり、今までインターネット上では難しいと思われていた動画や音声による放送や、多人数による会議などを実現した。このMBoneは今までコンピュータが取り扱うには不向きであると思われていた動画や音声のような冗長な情報にむしろ向いており、新たな展開を見せている。

著者らは、ある側面から見ると情報の形態には2種類のものがある。ちょうどWWWとMBoneはそれら2種類の形態の情報のやりとりに関してうまく互いを補いあう形になっていくのではないかと考えている。

1つ目の情報の形態は、テキストなどを含む、冗長性が少なく、厳密に取り扱う必要がある情報形態である。この形態の情報は冗長性が少なく蓄積にも向いている。WWWの枠組は、このような情報を他から容易にアクセスしやすいように蓄積し、やりとりすることを可能にする。

もう1つの情報の形態は、音声や動画などを主とする冗長性の高い、人間が日常生活で使っているような情報形態である。この形態の情報は冗長性が高いので蓄積には向かないが、厳密に取り扱わなくてもよく、多くの場所に届ける場合にも問題が少ない。MBoneのような枠組は、放送型や会議型のアプリケーションの動画や音声などのデ

ータなどのように「流れていく」ような情報のやりとりを可能にする。

すでにWWWとMBoneとの興味深い関係の例がいくつか現れており*、これからもさらにWWWとMBoneのそれぞれの特徴をうまく生かしたような組合せが出てくるのではないかと考えている。

インフラストラクチャ

WWWの将来は、このような標準化などに加えてネットワークのインフラストラクチャにも関わってくると考えている。WWWを含むインターネットが健全に成長していくには、ネットワークの回線幅が十分あることが重要である。残念ながら、日本においてはその回線幅が十分であるとはいえない。その1つの原因は、日本ではネットワーク、すなわち通信にかかる費用が電話代を含めて高すぎることにありのように感じている。しかしこの問題は現在注目が集まっており、将来的にはかなりよくなるのではないかと著者らは楽観的に見ている。

7. まとめ

World-Wide Web (WWW) について、その特徴、歴史、技術、ホームページのデザイン、課題と将来などを解説した。その技術については特にその動的な側面に焦点をあてて解説した。またホームページのデザインについていくつかの提言を行い、最後のWWWの課題や将来についてはその標準化動向を中心に解説した。

WWWは計算機業界はもちろん今までは計算機とあまり関係のなかった分野からも注目を集めており、変化も激しい。どんどん新たな情報が入ってきて、執筆は難しい作業であった。そのような状況のもと、単なる事実や情報の紹介ではなく、自らの視点からの説明や、オリジナリティのある古くならない情報を提供しようと心がけた**。読者の皆さんがWWWの新たな側面を1つでも見い出せることができたなら幸いである。

* 1つの例はURLをmulticastで分配して、分散したWWWクライアントたちで同じページを見るようにするものである。

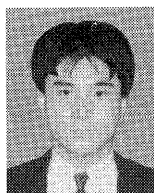
** これはホームページにおく情報にも通じる。すなわち情報を自ら作り出して情報発信しないだけでもそのホームページに見向いてくれなくなる。

* 以前はHTML+と呼ばれていたものに相当する。

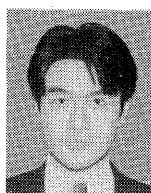
参考文献および参考 URL

- 1) Krol, E.: The Whole Internet: User's Guide & Catalog, Second Edition, 543 p., O'Reilly & Associates, Inc. (1994). (日本語訳がオーム社より出ている。)
- 2) 木庭袋圭祐, 益岡竜介: World Wide Web によるメディア統合, 第 100 回データベースシステム研究会記念合同研究会論文集, pp.185-192 (1994).
- 3) World-Wide Web Consortium (W3C), <http://www.w3.org/hypertext/WWW/Consortium/Prospectus/Overview.html>
- 4) IETF の WWW 関連ページ,
<http://www.ics.uci.edu/pub/ietf/html>
<http://www.ics.uci.edu/pub/ietf/http>
<http://www.ics.uci.edu/pub/ietf/uri>
- 5) RFC 1738, "Uniform Resource Locators (URL)" (1994).
- 6) RFC 1737: Functional Requirements for Uniform Resource Names (1994).
- 7) Internet Draft: HyperText Markup Language Specification-2.0<draft-ietf-html-spec-01.txt> (1995).
- 8) Internet Draft: HyperText Markup Language Specification Version 3.0<draft-ietf-html-specv3-00.txt> (1995).
- 9) Eriksson, H.: Mbone: The Multicast Backbone, Communications of the ACM, pp.54-60 (Aug.1994).

(平成 7 年 3 月 3 日受付)

**益岡 竜介 (正会員)**

昭和 37 年生。昭和 60 年東京大学理学部数学科卒業。昭和 62 年同大学修士課程卒業。理学修士。昭和 63 年(株)富士通研究所入社。平成 3 年カーネギーメロン大学客員研究員。平成 5 年より富士通研究所に戻り、現在マルチメディアシステム研究所ネットメディア研究センター研究員。ニューラル・ネットワークなどの研究に携わる。

**大庭袋圭祐**

昭和 45 年生。平成 5 年東北大学工学部精密工学科卒業。同年富士通(株)入社。現在に至る。主に文書形式、セキュリティ、World Wide Web などに関連する技術動向の調査に従事。

訂 正

本誌第 36 卷 12 号 pp.1155-1165 に掲載されました解説「World-Wide Web (WWW)」の著者紹介文 (p.1165) に以下のような誤りがありました。

誤→“大庭袋圭祐”

正→“木庭袋圭祐”

本誌第 37 卷 1 号 pp.11-17 に掲載されました特集「計算物理学と超並列計算機－ CP-PACS 計画－ 1. 計算物理学と CP-PACS 計画」の岩崎洋一氏および梅村雅之氏のご所属 (p.11) に以下のような誤りがありました。

誤→“〒 筑波大学物理各系および計算物理学研究センター”

正→“〒 筑波大学物理学系および計算物理学研究センター”

また、本誌第 37 卷 1 号 pp.29-37 に掲載されました特集「計算物理学と超並列計算機－ CP-PACS 計画－ 3. 超並列計算機 CP-PACS のソフトウェア」の中田育男氏の著者紹介文 (p.37) に以下のような誤りがありました。

誤→“1958 年東京大学理学部数学教授。”

正→“1958 年東京大学理学部数学科卒業。”

執筆者の方々および関係者の皆様にご迷惑をおかけしたことをお詫びして訂正いたします。