

Web 技術と WebGIS

山田 敏博

Web Technology and WebGIS

Toshihiro Yamada

Abstract

The basis of Web Technology and its application to GIS are described in this technical note. As an example of WebGIS, OYO's developing system is introduced.

キーワード：Web, クライアント, サーバー, GIS

1. はじめに

これまでネットワークコンピューティングの主流であったクライアント/サーバーシステム^{*1}は、クライアント側に専用のソフトウェアをインストールする必要があるため、システムを更新するたびにすべてのクライアントのソフトウェアをインストールし直さなければならず、管理が大変であった。

一方、インターネットに代表的される Web (ウェブ) システム^{*2}においては、クライアント側に必要なのは Web ブラウザ^{*3}だけであるため、システムの更新がサーバー側だけで行うことができ、管理の負担が大幅に軽減されることから、近年、様々な分野で急速に普及している。

ここでいう WebGIS は、この Web 技術を利用した GIS 環境を指している。

以下では、まず WebGIS の基礎となっている Web 技術について整理し、次に現在開発しているシステムを例に WebGIS について紹介する。

なお、本稿には多くの情報関連の用語が出てくるが、煩雑さを避けるために本文中の説明は最小限にとどめている。興味のある方は、文末の用語集を参照されたい。

2. Web 技術の概要

2.1 Web 技術の進化¹⁾

1) Web の起源

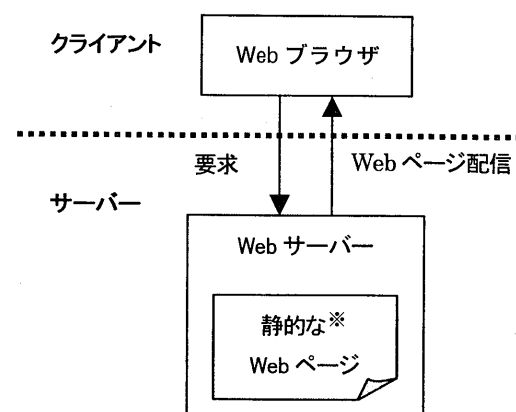
Web はもともと欧州原子核研究機構 (CERN) に参加する研究者間の文書情報のやり取りを目的として開発されたものであるが、その仕様 (表-1 参照) が一般公開されたことから全世界に急激に広がった。

初期の Web システムは、クライアントの要求に応じ

てあらかじめ用意された (=静的な, Static) Web ページ (HTML^{*4}で書かれた文書ファイル) を Web サーバー^{*5}が配信するという仕組みであった (図-1 参照)。

表-1 Web の仕様
Table.1 Specification of Web

<記述言語> HTML <書式情報の表示プログラム> Web ブラウザ <要求に応じて書式情報を配信するプログラム> Web サーバーソフトウェア <コンピュータ間のやりとりの手順> HTTP ^{*6}



※静的=Static

図-1 初期の Web システムの構成
Fig. 1 Structure of Primitive Web system

2) DHTML への拡張

HTML の登場により、多くの人間がインターネットで情報を発信できるようになったが、Web に対するニーズが一方的な情報の配信から「双方向のコミュニケーション」へと変化していくにつれ、HTML だけでは対応しきれない課題が表面化し、HTML にさまざまな機能が追加されるようになった。この HTML の拡張仕様を DHTML (DynamicHTML) ^{*7} という。

DHTML では、イラストを回したり、背景色をフェードアウトさせたり、といった操作を HTML に埋め込んだスクリプト ^{*8} というプログラムによって制御している。

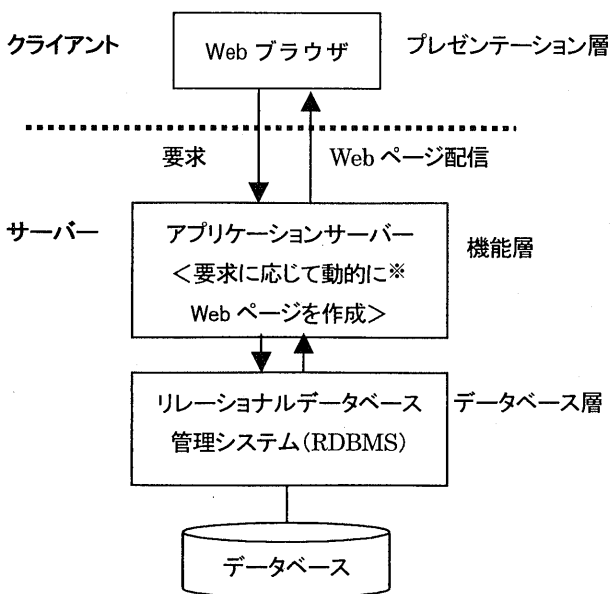
3) CGI の利用

データベース検索などさらに「複雑なサービス」が必要になってくると、CGI(Common Gateway Interface) ^{*9} という仕組みを使って、Web サーバー外部のプログラムに複雑な処理をさせ、その結果を表示するといった方法がとられるようになってきた。

ただし、CGI を使うとクライアントの要求ごとにプログラム処理の実行単位であるプロセスが発生するため、利用者が多くなるとサーバーに負荷がかかり、処理が遅くなるという新たな問題が発生した。

4) アプリケーションサーバー

こうしたサーバー側の負荷を分散し、処理を「高速化」するためにマルチスレッドという技術が生まれ、これを組み込んだアプリケーションサーバー ^{*10} が登場した。図-2 は現在主流となっているアプリケーションサーバーを使った 3 階層の Web システムの構成を示している。



※動的:Dynamic

図-2 3階層のWebシステム

Fig. 2 Structure of 3 tiered Web system

2.2 最近のWeb技術 ^{1),2)}

サーバーマシンのOSは、マイクロソフト社のWindows系OSと商用UNIXやLinuxなどのUNIX系OSに大別される。Windows環境で作ったWebシステムはWindows環境でしか動かないが、Java ^{*11} を使ったシステムはWindows系とUNIX系の両環境で動かすことができる。ここでは、Javaを使ったシステムを例に最近のWeb技術を概観する。

2.2.1 サーバー側の技術

サーバー側のもっとも基本的な機能は、図-1に示すようにWebブラウザの要求にしたがって、HTMLファイルをWebブラウザに配信するWebサーバーの機能である。

一方、図-2に示したアプリケーションサーバーは、Webサーバー機能とJavaプログラムを実行する環境を統合したものであり、この上で様々なユーザープログラムが動作し、複雑な処理を実現する。アプリケーションサーバー上で動くプログラムには次の3種類がある。

<サーブレット(Servlet) ^{*12}>

Webサーバー上で動くJavaプログラム。WebサーバーはWebブラウザから要求を受けると、サーブレットを起動して処理を実行し、その結果をWebブラウザに返す。

<JSP(Java Server Pages) ^{*13}>

Webブラウザの要求に応じて、動的(Dynamic)にHTMLファイルを作成する。

<EJB(Enterprise Java Beans) ^{*14}>

再利用可能なように部品化したJavaプログラム。

簡単なシステムの場合は、これらの内のいずれか一つを使って開発することも可能であるが、大規模なシステムの場合は管理を容易にするために、これらの特徴を生かして使い分けることが多い。この代表的な方法がMVCモデル(Model-View-Control)とよばれるもので、次のような使い分けを行う。

- ① “EJP” が “データの保持と処理 (Model)”
- ② “JSP” が “画面表示 (View)”
- ③ “サーブレット” が “入力情報の受け取り、モデルとビューの制御 (Control)”

2.2.2 クライアント側の技術

クライアント側で高度な処理をする場合やサーバー側の負荷を減らしたい場合は、Javaアプレット ^{*15} やBean ^{*16} などのプログラムをWebブラウザ上で実行する。

この場合でも、これらのプログラムはサーバー側からWebブラウザに自動的にインストールされるので、ク

クライアントがインストール作業をする必要はない。

3. WebGIS の構築

ここでは WebGIS の例として社内で開発しているシステムを紹介する。

3.1 WebGIS の開発方針

WebGIS の利点は、前述のようにネットワークに接続しているすべての人が自分のマシンにソフトウェアやデータをインストールすることなく GIS が利用できる点である。

一方で、Web 上で全てのことをしようとすると、システムが大きくなり、「動きが遅くなる」、「開発に時間がかかる」といった問題がある。

開発にあたっては、これらのことを総合的に勘案して以下のことを基本方針とした。

- 1) WebGIS によって社内で整備している地理情報を共有する。
- 2) 社内ネットワークにつながるすべてのマシンから利用できるシステムとする。このため、わかりやすいユーザーインターフェイスとする。
- 3) 機能は GIS の基本機能に絞り、できるだけ軽く動くようにする。
- 4) 高度な処理は、別途設備しているデスクトップ GIS (MapInfo) で実現する。このため、デスクトップ GIS に対してデータを配信できる仕組みとする。
- 5) できるだけ短時間で WebGIS の運用を開始できるよう、開発は 2 段階に分けて行う。すなわち、第 1 フェーズ (平成 13 年度) では最小限のシステム構

成とし、第 2 フェーズで、パフォーマンスの向上等、最終的な仕上げを行う (図-3 参照)。

3.2 WebGIS の開発環境

利用環境に柔軟性を持たせるため、Windows 系と UNIX 系の両環境で使える Java システムを採用した。

また、開発エンジンはデスクトップ GIS (MapInfo) とデータの互換性がある MapXtream (Java Edition) とした。MapXtream は、Web 上で GIS 機能を実現するためのプログラム製品群で、前述のサーブレットや Bean の形でプログラムが提供されている。

3.3 WebGIS におけるマッピングの方式

MapXtream には次の 3 種類のマッピング方式がある。社内システムでは、操作性と速い地図表示、工期短縮の点から Medium 方式を採用している。

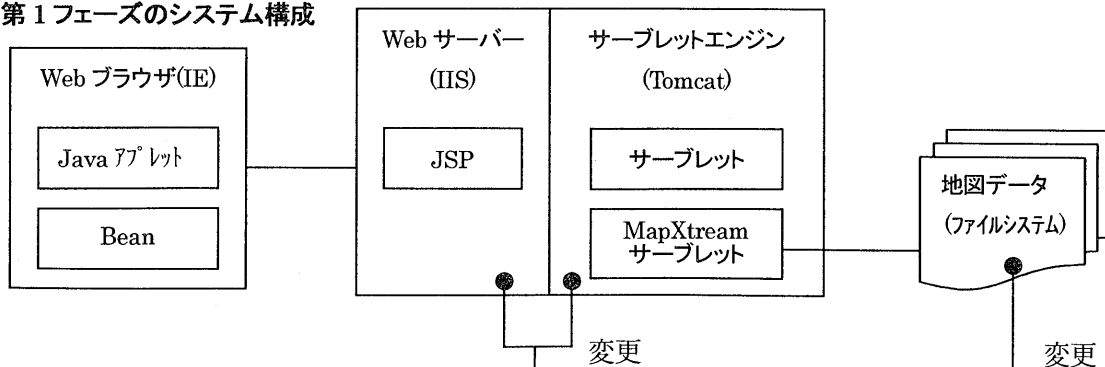
<Thin 方式>

地図を GIF^{*17} で作成しこれを HTML ファイルに埋め込んで配信する方式。拡大やレイヤ管理といった GIS 機能も使えるが、簡単な機能に限られる。ただし、配信するものが HTML ファイルだけなので地図の表示が速いのが利点である。

<Medium 方式>

地図を GIF で作成しこれを HTML ファイルに埋め込んで配信する点は Thin 方式と同様であるが、Java アプレットや Bean を Web ブラウザに組み込むことにより、良好な操作性を実現できる。Java アプレットや

a) 第 1 フェーズのシステム構成



b) 第 2 フェーズのシステム構成

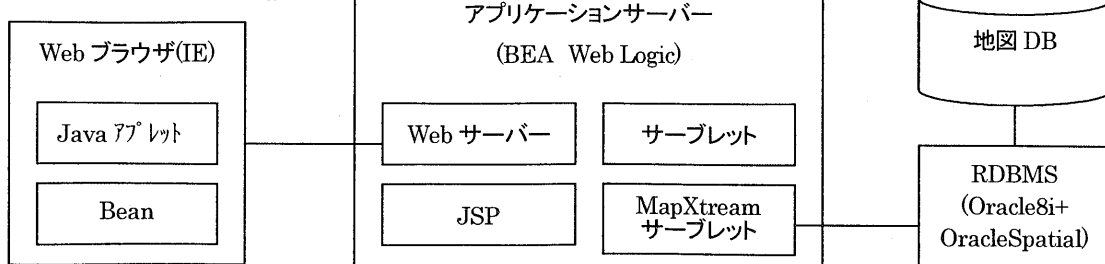


図-3 社内 WebGIS のシステム構成
Fig. 3 Structure of in-house WebGIS

Bean を読み込む際に時間がかかる。

〈Thick 方式〉

地図をベクトルデータで配信し、Java アプレットや Bean を使って Web ブラウザ上で地図を処理する方式。デスクトップの GIS ソフトウェアと同じ多くの機能を実現できるが、データを送信するたびに時間がかかるという問題がある。また他の方式に比べて開発に時間がかかる。

3.4 WebGIS のシステム構成

図-3(a)が第1フェーズのシステム構成、(b)が第2フェーズで予定しているシステム構成である。

第1フェーズでは、サーブレットの実行環境としてアプリケーションサーバーではなくフリーソフトウェアのサーブレットエンジン^{※18}（ツール名：Tomcat）を使用している。また、地図データはファイル形式のまま使用している。

第2フェーズでは、システムのパフォーマンスを向上させるとともにデータの管理を強化するために、アプリケーションサーバー（製品名：BEA WebLogic）を使用し、地図データの管理をRDBMS（製品名：Oracle8i + OracleSpatial）に切り替える予定である。

また、現状はデータを含めたすべてのシステムを1台のマシン上に乗せているが、アクセス量が増えパフォーマンスが落ちてきた場合には、データベースや GIS 機能をアプリケーションサーバーと別のマシンに分離する予定である。こうしたシステムの拡張はアプリケーションサーバーなどの最新の Web 技術によってソフトの変更作業等を伴わずに実施できる。こうした拡張性はシステムのスケラビリティとよばれており Web 技術の重要な要素となっている。

3.5 WebGIS の機能

本システムで利用できる機能を表-2に整理する。このうち拡張機能が、業務を支援するために追加した機能である。

4. まとめ

GIS の普及を阻害している要因としては、一般にソフトウェアが高価なことや地図データの整備に手間がかかるといったことがあげられよう。この点 WebGIS は、Web ブラウザさえあればソフトウェアもデータもインストールせずに使えるのでユーザーにとっては利便性が高いものと考えられる。

こうした意味で、現在開発を進めている WebGIS も社内における GIS の普及に大きな効果があるものと期待している。

ただし、現状のシステムは GIS の基本機能に絞っているので、このシステムをさらに広い分野で使えるようにするためには、以下のような点について検討を進める

表-2 社内 WebGIS の機能

Table.2 Function of in-house WebGIS

項目	機能
ナビゲーション	拡大
	縮小
	移動
	前画面
	全体表示
	サーチ
	案内図
状態表示	スケール
	座標値
基本機能	凡例
	情報表示
	計測（距離・面積）
	レイヤ管理
	検索
	主題図
拡張機能	書込み機能
	印刷（案内図等のテンプレート）
	エクスポート（デスクトップ GIS への配信）
管理機能	ユーザー管理
	データ管理
	プロジェクト管理
	アクセス管理

必要がある。

- 1) 点検調査などへの応用を目的としたモバイル環境や GPS との統合
- 2) 全庁型 GIS への拡張
また、最近では技術的な問題よりはコンテンツがより重要視されるようになってきており、
- 3) WebGIS を使った独自の情報サービスについても検討していきたいと考えている。

参 考 文 献

- 1) 小泉修(2001)：図解でわかる Web 技術のすべて、日本実業出版社
- 2) 日経 BP 社(2001)：日経ソフトウェア、第4巻、第12号

用 語 集

※1 クライアント/サーバー

ネットワークに接続されている他のコンピュータの要求に応じてサービスを行う側のコンピュータをサーバー、ネットワーク上のサーバーからサービスを受ける側のコンピュータをクライアントという。

※2 Web : World Wide Web の略

インターネットでのサービスシステムのひとつ。ハイパーテキストの概念を応用した分散型の情報システム。WW

W, W3 などとも呼ばれる。Web コンテンツ（インターネット上の Web サーバーに掲載されているテキストやグラフィックなど）を閲覧するには Web ブラウザが必要。

※3 Web ブラウザ : Web Browser

Web ページを閲覧するためのソフトウェア。Netscape Communications 社の Netscape Navigator / Communicator と Microsoft 社の Internet Explorer が有名。

※4 HTML : Hyper Text Markup Language

SGML (Standard Generalized Markup Language) の書式を踏襲したマークアップ言語の 1 つ。HTML は Web サーバーのドキュメントを記述するための言語として広く知られている。

※5 Web サーバー : Web Server

Web ブラウザで閲覧するコンテンツを提供するコンピュータ。またそのために稼働しているソフトウェア。Windows 環境では Internet Information Server (IIS), UNIX 系では Apache などが有名。

※6 HTTP : HyperText Transfer Protocol

Web ブラウザと Web サーバー間において、HTML ファイルなどの文書を転送するために用いられるアプリケーションレベルの手順。

※7 DHTML : Dynamic HTML

Web ページ上で動きのあるコンテンツを表示するために HTML を拡張した技術。ポインタをあてることで画像を表示させたり、イラストを回したり、背景色をフェードアウトさせたり、といった動的な Web コンテンツを実現する。HTML4.0 とスタイルシート、スクリプト言語がコアになっている。処理はクライアント側の Web ブラウザで行う。

なお、DHTML は開発元が独自に Web ブラウザに機能を追加していったため Web ブラウザ間で互換性がないという問題があり、現在標準化が図られている。

※8 スクリプト : Script

スクリプト言語の仕様に沿ってテキストで記述されたプログラム。スクリプト言語は Basic などと同じインタプリタ言語の一つ。DHTML では JavaScript や Jscript, VBScript などのスクリプト言語が使われる。後述の CGI では、JavaScript や VBScript の他に Perl や Ruby などのスクリプト言語が使われる。

※9 CGI : Common Gateway Interface

Web サーバーから外部プログラムを利用するための標準インターフェース。CGI を利用すると外部プログラム（ゲートウェイ）を起動し、処理結果を Web ブラウザに送り返すことができる。そのため、HTML だけでは不可能な高度な処理を行う Web ページの作成が可能

になる。

※10 アプリケーションサーバー : Application Server

3 階層情報システムの中に位置し、ユーザーの要求とバックエンド（データベースなどの業務システム）の処理を橋渡しするミドルウェア。最近では、インターネットの普及に伴って、Web ブラウザと業務システムを連携させる Web システムをベースにしたものを指す。この場合は特に、「Web アプリケーションサーバー」と呼ぶこともある。

※11 Java

Sun Microsystems 社が 1995 年に開発したオブジェクト指向言語。Java の最大の特徴はプラットフォームに依存しない動作が保証されている点である。Java アプリケーションは、Java Virtual Machine (Java VM) と呼ばれる動作環境をもつコンピュータなら、ハードウェアや OS が異なっても変更なしに動作させることができる。Netscape Navigator / Communicator 2.0 以降や Microsoft Internet Explorer 3.0 以降では、Java で書かれた小さなアプリケーション (Java アプレット) を Web サーバーから自動的にダウンロードして実行する機能をもつ。

※12 サブレット : Servlet

サブレットはサーバー上で Web サーバーと連携して動く Java プログラム。Web サーバーは Web ブラウザから要求を受けると、サブレットを起動して処理を依頼し、その結果をクライアントに返す。これは Web サーバーで以前より使われている CGI の仕組みに似ているが、サブレットは CGI よりパフォーマンスや安全性に優れている。サブレットは JavaVM で動作するサブレットエンジンと呼ばれる実行環境で動作する。実際には、Web サーバーの機能とサブレットエンジンを統合したアプリケーションサーバー製品が使われる。

※13 JSP : Java Server Pages

JSP は、JSP ファイルの中に HTML に加えて Java のプログラムコードや JavaBeans のコンポーネントを埋め込んで、動的に Web ページを作成する技術。JSP はマイクロソフト社の Web アプリケーション実行環境である ASP と似ているが、ASP は同社の Web サーバーである IIS でしか使えないのに対し、JSP は JavaVM およびサブレットエンジンが稼働していれば、Windows 以外にも Unix や Linux 上の他の Web サーバーでも動く。

※14 EJP : Enterprise Java Beans

Java プログラムを再利用可能なソフトウェア部品にする JavaBeans を拡張した、サーバーサイド向けのソフトウェア部品化技術の規約。

※15 Java アプレット : Java Applet

小規模なアプリケーションをアプレットという。通常は、Web ブラウザ上（より正確には、ブラウザに組み込まれた Java VM）で作動する Java アプリケーションを Java アプレットという。

※ 16 *Bean*

Java アプリケーション・プログラムを構成する個々の機能をコンポーネント化（部品化）して、共通のインターフェースを通して動作できるように考えられたコンポーネント・モデル（規約）が JavaBeans。個々の JavaBeans コンポーネントを Bean と呼ぶ。

※ 17 *GIF*

Graphics Interchange Format の略。アメリカの大手パソコン通信ネットワークの CompuServe で開発された画像データのフォーマット。モノクロは 256 階調、カラーは 256 色まで扱える。インターネットで JPEG とならんでよく使われる。

※ 18 *サーブレットエンジン*

Servlet API モジュールとともに、Apache などの Web サーバーをサーブレットが動作するように拡張するためのモジュール。