

「XMLを利用したシステムの実例」

大分県情報サービス産業協会
技術研究会
「XML部会」

平成14年2月15日

メンバー紹介

部会長	平井 英二	(株)シーエイシー システム部
副部会長	村田 昭一郎	(株)富士通大分ソフトウェアラボラトリ 第二ソリューション営業部
	河野 裕一	大分シーイーシー(株) 第一システム部
	小野 万尚	(株)オーイーシー Webビジネス推進部
	後藤 順子	大銀コンピュータサービス(株) 業務課
	佐藤 みゆき	新日鉄ソリューションズ(株) システム第二部
	矢須田 剛士	(株)アトムス事務器 ソフト開発課
技術委員	築城 久敏	システムトレンド(有)

- 目次 -

はじめに	5
1 . Web の現状と XML の可能性	6
1.1. Web の発展と HTML	6
1.2. Web 上での情報検索	6
1.3. 意味的な情報を規定する XML	7
1.4. XML 文書の活用	7
2 . XML を利用した検索システム	8
2.1. システム概要	8
2.2. 開発環境	9
2.3. データ構造	9
3 . パソコン情報検索システムの実現について	12
3.1. 店舗について	12
3.1.1. 概要	12
3.1.2. 構成	12
3.2. サーチャエンジン (サーバーサイドプログラム) について	15
3.2.1. 概要	15
3.2.2. サーバーサイドプログラミング	15
3.2.3. ASP	15
3.2.4. サーチャエンジンプログラムの概要	16
3.2.5. 実行画面	17
3.3. サーチャエンジン (クライアントサイドプログラム) について	19
3.3.1. 概要	19
3.3.2. クライアントサイドプログラミング	20
3.2.3. JavaScript	20
3.2.4. サーチャエンジンプログラムの概要	21
3.4. 評価・考察	22
3.4.1. 店舗について	22
3.4.2. サーバーサイドプログラムについて	22
3.4.3. クライアントサイドプログラムについて	23
3.4.4. ASP について	23

3.4.5. XML について	23
4 . XML での開発を終えて	25
4.1. XML の可能性等	25
4.2. 従来の検索システムとの比較	25
4.3. XML によるシステム構築の利点	25
4.4. XML がソフトウェア産業にもたらす変革	26
4.5. XML を取り巻く周辺技術	26
4.6. XML 技術者認定制度	27
4.7. 最後に	27
略語・用語	28
参考文献	31
「XML 技術研究会」に参加して	33

はじめに

現在、インターネットの Web サイト上はもちろんのこと、ネットワークにつながっていないところでも日々膨大な量の情報が生み出されている。それらは、たとえば Web 上にある HTML 文書であれば閲覧という形で共有することができる。また、ワープロ文書などはアプリケーションを統一することでデータ交換を可能としている。しかし、情報の共有・交換・再利用という点では極めて限定的なものでしかない。Web 上の HTML 文書を再利用するのはそれほど簡単なものではないし、ワープロ文書もアプリケーションやバージョンの違いにより再利用できないことがしばしばである。

こうした現状で XML が登場し、情報・データの共有や再利用を十分な形で実現する文書として期待されている。XML は簡単な仕様であるとともに、文書内容と書式情報を分離することで応用範囲を格段に広げることが可能にした。

インターネットを通じて世界中がつながっている現代において、XML による情報の共有・交換・再利用は非常に大きなメリットをもたらすといわれている。

現在、Web ページの多くは HTML によって記述されている。これらが XML で記述されていれば、具体的にどのようなことが可能となり得るであろうか。我々は、Web 上の XML 文書を利用して、HTML 文書では作成不可能なシステムを作成することで、XML に対する理解を深めていくことにした。

1. Web の現状と XML の可能性

1.1. Web の発展と HTML

1969 年のアメリカ国防総省高等研究計画の出資による ARPAnet に始まるインターネットの歴史は、1994 年頃からの WWW ブームにより、研究者たちだけのものから、一般大衆のものへと大きな変化を遂げた。WWW によって Web ページが誕生し、それを活用して個人も企業も、あるいは行政や学校も、実にさまざまな情報を発信している。

現在、Web ページはおもに HTML というマークアップ言語を使用して記述されている。HTML は SGML というマークアップ言語から派生したものである。仕様が複雑で扱うのが困難な SGML に比べ、HTML はシンプルで習得しやすいだけでなく、ほかのファイルやページへの関連付け(リンク)ができる。また、HTML によるホームページは、OS が Windows や Macintosh、さらに最近隆盛の Linux であろうと、Web ブラウザさえあれば見ることができる。このように HTML は、作る場合も見る場合も簡単であるという特徴を持っている。このことにより、ビジネスの世界はもちろん、個人のプライベートな生活においてもインターネットが急速に普及していった。

膨大な量の情報がインターネット上に出現し、なにか分からないことがあったらインターネットで調べるという方法も定着してきた。

1.2. Web 上での情報検索

我々は、希望する情報にたどり着くために、Web ブラウザの検索機能を多々利用する。しかし、あまり要求に沿わない結果が返ってきたり、検索結果のリストが膨大になりすぎたりして、目的の文書を絞り込むのに多くの時間が必要となる。HTML では、満足する検索結果が得られないのが現状である。

このような事態となる原因は HTML の構造にある。HTML のタグは規約によりあらかじめ決められており、新しいタグを増やすことは出来ない。そして、その機能の大部分が文書のデザインやレイアウトのために使われている。HTML の文法がコンテンツの意味的な構造を規定していないため、利用者が Web ブラウザに映し出されたコンテンツを見て内容を判読できても、プログラムが認識することはできないのである。Web ブラウザの検索機能は、本質的にはコンテンツに含まれる単語を機械的に全文検索しているだけで、せいぜい特定の単語の使用頻度などから内容を憶測しているにすぎない。したがって、HTML で記述された複数のページの中から、もっとも希望に添うデータを捜し出すためには、利用者が Web ページをひとつひとつ自分で確認・比較するというアナログ的な検索方法をとらなければならない。

1.3. 意味的な情報を規定する XML

HTML での情報の共有・交換・再利用の限界等もあり、現在の HTML にコンテンツの内容に関する情報を書き込んだり、意味的な構造を規定できる「メタ情報」を取り入れようとする動きが活発になってきた。ここで、HTML の元でもあり、文書のデータのみを電子的に永久保存する目的で開発された SGML が注目を集めた。SGML は自由に定義したタグによって文書の意味を構造化できるものであるが、仕様の複雑さやマルチメディア環境への対応の不十分さという問題があった。そこで、SGML の複雑すぎる部分やあまり使用されない部分をカットして簡潔な仕様にし、URL の使用・リンク記述・名前空間の使用といった SGML にはない機能を付加した XML が登場した。XML は W3C により標準化が行われ、1998 年 2 月に XML 1.0 の仕様書が勧告として公開された。

XML は HTML が持つ Web ブラウザの操作性を保ちつつ、SGML が持つコンテンツをデータとして扱うための特質を備えるため、情報検索の手法を根本的に変える可能性を持つといわれる。XML ではタグを自由に増やすことが出来、このタグに意味を定義する。このことにより、プログラムにも内容が把握できるようになる。

1.4. XML 文書の活用

XML ではデータに意味を持たせる為、プログラムが内容を認識できる。Web ページが XML を使用して記述されていれば、HTML では限界があった細かい条件による速やかな検索が可能となるであろう。また、インターネット上に蓄積された情報をデータベースとして活用することや、データをそのまま再利用することも可能となりそうである。我々は、その第一歩として、Web ページとして利用されているインターネット上の XML 文書から、複合条件にてデータを抽出・表示するような検索システムを作成し、その調査研究をまとめた。

2. XML を利用した検索システム

2.1. システム概要

我々が開発した検索システムは、複数の仮想店舗の Web ページよりパソコン情報を検索し、条件に該当するデータを一覧表示する、というものである。ユーザーは、自分の要望を満たすスペックのパソコンがどの店舗にあるか、或いは特定のパソコンを買う場合、どの店舗が一番安い、ということを知ることができる。仮想店舗でのパソコンデータの保存形式として、XML を利用した。

開発は、2 通りの方法で行った。1 つ目は、VBScript を用いて作成したサーバーサイドで動くシステムであり、もう 1 つは、JavaScript を用いて作成したクライアントサイドで動くシステムである。

また、システムを実現するためには、各店舗で扱うパソコンデータのタグ名等のデータ構造が統一されている必要がある。そこで、このシステムを開発する上での前提として、各店舗でデータ構造が統一されていることを掲げた。現実の店舗の Web ページでの実現は難しいため、今回は、前提を満たした仮想店舗の Web ページを作成することとなった。

システムのイメージ図を、図 2.1-1 に示す。

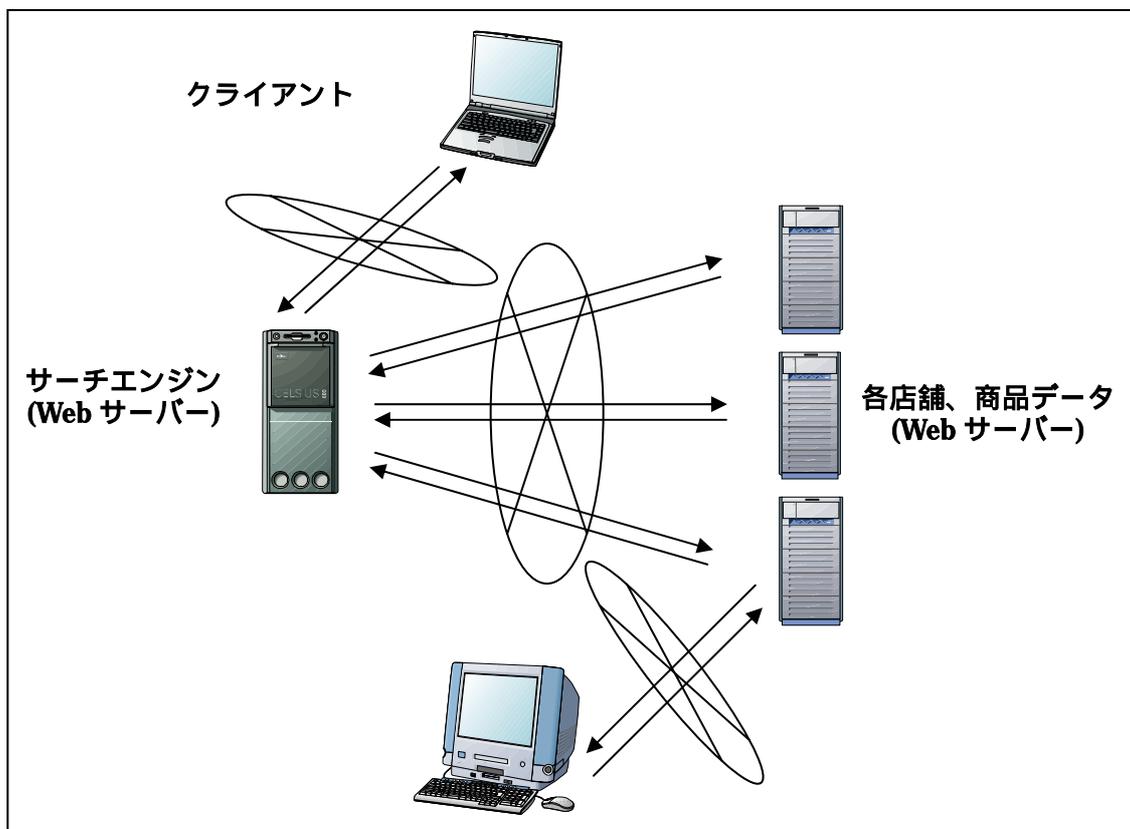


図 2.1-1 検索システムイメージ図

2.2. 開発環境

開発は検索システム、各店舗共に以下の環境で行った。

OS	Windows 98, Windows 2000
ブラウザ	Microsoft Internet Explorer(IE) 5.0 以上
XML パーサー	IE5.0 以上に標準搭載されている MSXML Parser を使用
Web サーバー	(Win 98) Microsoft Personal Web Server 4.0 (Win 2000) Microsoft Internet Information Services 5.0

2.3. データ構造

パソコンデータを格納する XML の構造決定にあたり、まず、検索システムの検索結果で表示したい項目と検索条件を指定したい項目とを挙げた。また、検索条件の内容は、()内の項目から選択可能とした。

< 検索結果 >

- ・品名
- ・型番
- ・価格(売価)
- ・スペック(CPU、メモリー、ディスプレイ、OS)
- ・リンク(各メーカーの製品紹介 URL)
- ・店舗名

< 検索条件 >

- ・店舗(複数の仮想店舗) 入力必須項目
- ・分類(指定なし、デスクトップ、液晶デスクトップ、A4 ノート、B5 ノート、その他)
- ・メーカー(指定なし、富士通、Sony、Compaq、NEC)
- ・価格(指定なし 10 万円以下、10 ~ 40 万円の間で 5 万円区切りの範囲指定、40 万円以上)
- ・クロック
(指定なし、500MHz 以下、500M ~ 1.5GHz の間で 500MHz 区切りの範囲指定、1.5GHz 以上)
- ・HDD(指定なし、20GB 以下、20 ~ 60GB の間で 20GB 区切りの範囲指定、60GB 以上)
- ・メモリー(指定なし、64MB 以下、64 ~ 128MB、128MB 以上)

次に、データ構造を決定した。決定に至るまでには、以下のような事項について検討した。

【その1】データは複数のファイルに分けて持つか？1つのファイルにまとめるか？

検索条件としてメーカーを持たせることに決まったので、最初はメーカー毎にファイルを分けた方が検索が速いと考えていた。しかし、メーカーを指定しなかった場合にはファイルを探すのに時間がかかってしまうため、全てのデータを1つのファイルにまとめて持つことにした。

【その2】スペックの情報の持ち方。

パソコンのスペックに関する情報の持ち方として、「スペック」タグを用意してはどうか、という検討もした。そこに様々なスペックの情報をまとめて記述しておき、検索時には「スペック」タグについて全文検索を行う、という方法である。しかし、同じ数値が検索条件以外の値として書かれていた場合にミスヒットしてしまうため、この方法は良くない、という結論に達した。そこで、「CPU」、「メモリー」、「HDD」などのタグを用意し、それぞれの値を個別に検索する仕組みにした。

これらの検索条件、表示項目、構造などを検討した結果、各店舗および検索システムで扱うXMLデータの構造は、図2.3-1のように決定した。

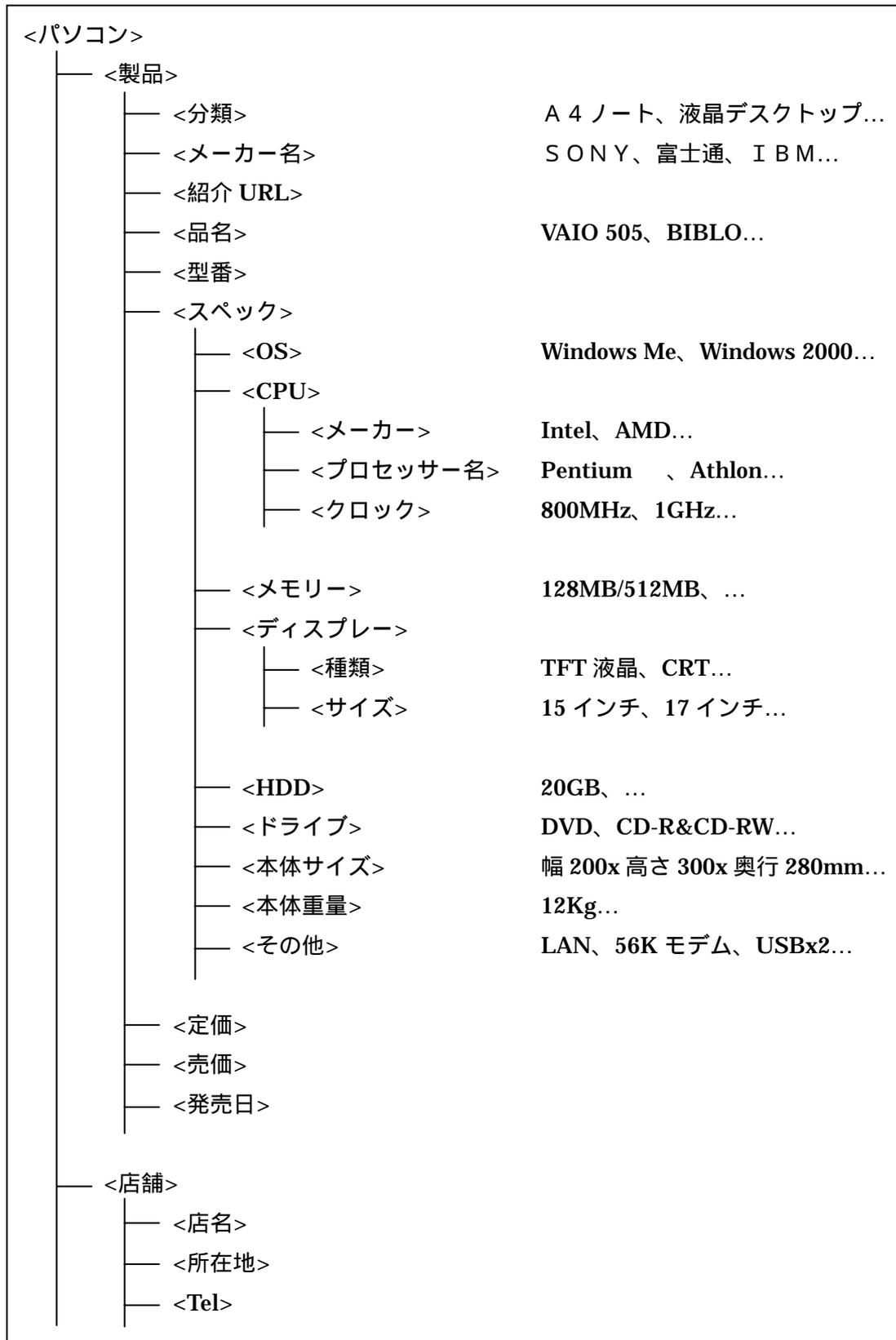


図 2.3-1 XML データ構造

3. パソコン情報検索システムの実現について

本研究では、オンラインショッピングで利用されることを想定した、パソコンの情報検索システムを作成した。ここでは、店舗と、検索システムをサーバーサイド、クライアントサイドという2つの観点から技術的に論じる。

3.1. 店舗について

パソコンの小売り店が共通の XML 構造で商品データを XML 文書として記述しており、そのファイルを使用して各店舗は価格情報を Web 上で公開しているとする。

3.1.1. 概要

この店舗のサイトは XML 文書と XSLT スタイルシートを組み合わせることでサーバーサイドで HTML 文書に変換し、クライアントサイドには HTML 文書を返している。

3.1.2. 構成

XML 文書と XSLT スタイルシートを組み合わせることで HTML 文書を作成することは、サーバーサイドで行うことも、クライアントで行うこともできる。XML 化されたデータを、XSLT スタイルシートを使って加工して HTML 文書へと変換するには、XSLT に対応した XML パーサーが必要になる。逆に言えば、XML 文書进行操作するマシンには XML パーサーがインストールされていなければ、XML の操作はできない。XML と XML パーサー、そして Web ブラウザの関係を図 3.1.2-1 に示す。

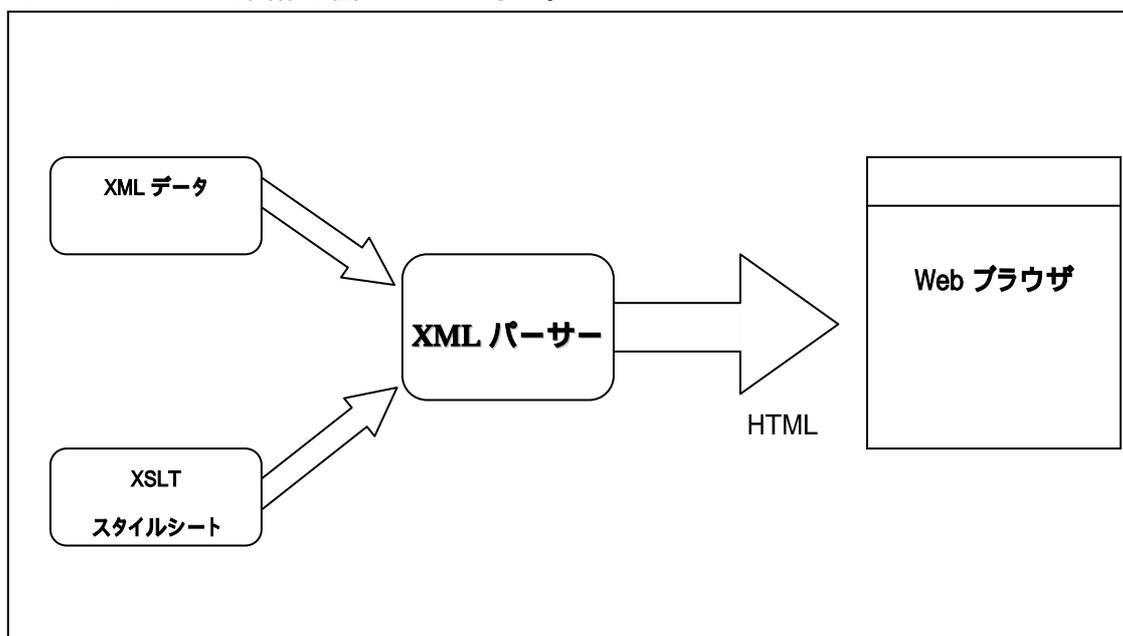


図 3.1.2-1

XML 文書をブラウザで表示する仕組み

いずれにせよ、XML 文書进行操作する側には XML パーサーがインストールされていなければならない。しかし、単に Web ブラウザに XML 文書とスタイルシート(この場合は XSLT スタイルシート)を送信し、Web ブラウザ側で XML 文書を変換して表示する方法には現在、多くの問題がある。例えば、すべてのプラットフォームですべての Web ブラウザが、XML 文書とスタイルシートに同じように対応しているとはいえない点である。そもそも HTML だけに対応し、XML には対応していない Web ブラウザも数多く存在する。このような現状では、データを XML 化し、Web ブラウザ側でスタイルシートを利用して表示するという方法は利用しにくい。これを避けるには、クライアントとして利用する Web ブラウザとして XML に対応した特定のものを使用することを規定することになる。現在主流のブラウザでは Microsoft Internet Explorer (IE) と Netscape Navigator (NN) が挙げられるが、それぞれバージョン 5.x、6.x 以上であれば XML をサポートしている。しかし、NN では XML の技術のサポートが不完全であるため、上記のような目的では実質 IE に特定するしかない。また、IE は OS が Windows、Macintosh 用しか存在しない。しかし、それではブラウザをクライアントとする Web アプリケーションとしての利点が損なわれてしまう。そのため、XML をデータとして使い、このデータをすべてのプラットフォーム上のあらゆる Web ブラウザで表示できるような Web アプリケーションを作成するには、現時点では XML 文書をサーバーサイドで HTML へと変換する処理を行い、Web ブラウザには HTML 文書を送るようになる必要がある。そのため、この店舗ではサーバーサイドで変換処理を行う構成とした(図 3.1.2-2)。

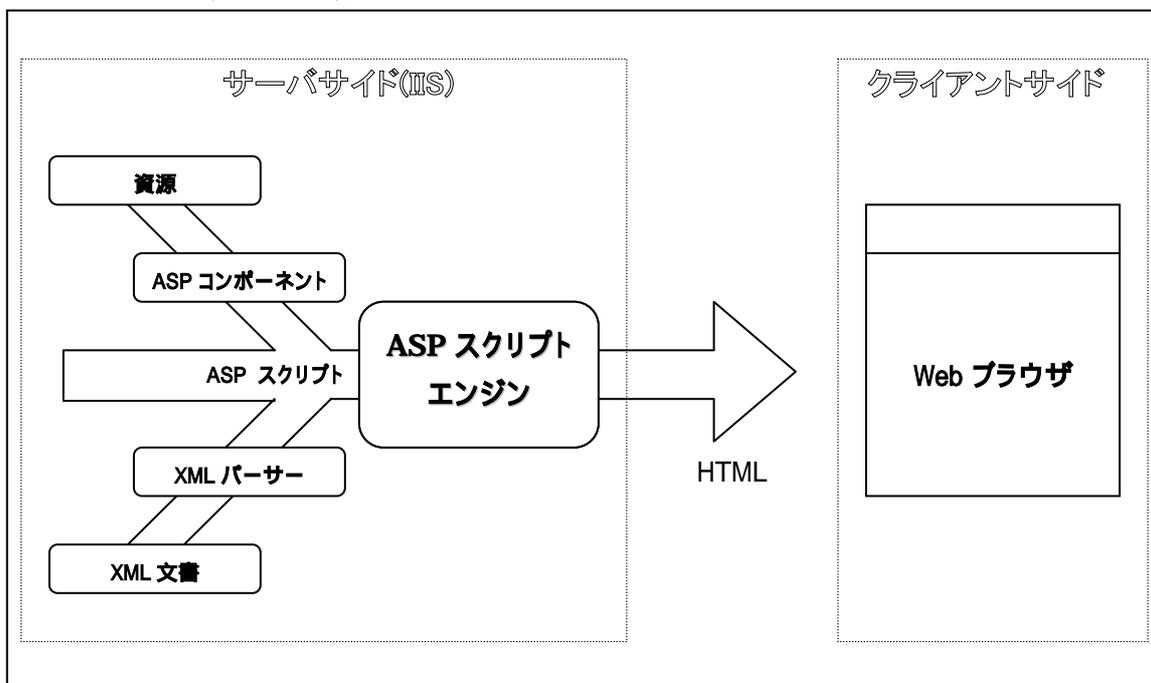


図 3.1.2-2

ASP を使用したサーバーサイド変換処理のイメージ



図上：XML をサポートした Web ブラウザでクライアントサイドで変換したものを表示した様子

図下：XML をサポートしていない Web ブラウザでサーバーサイドで HTML に変換したものを表示した様子



図 3.1.2-3.

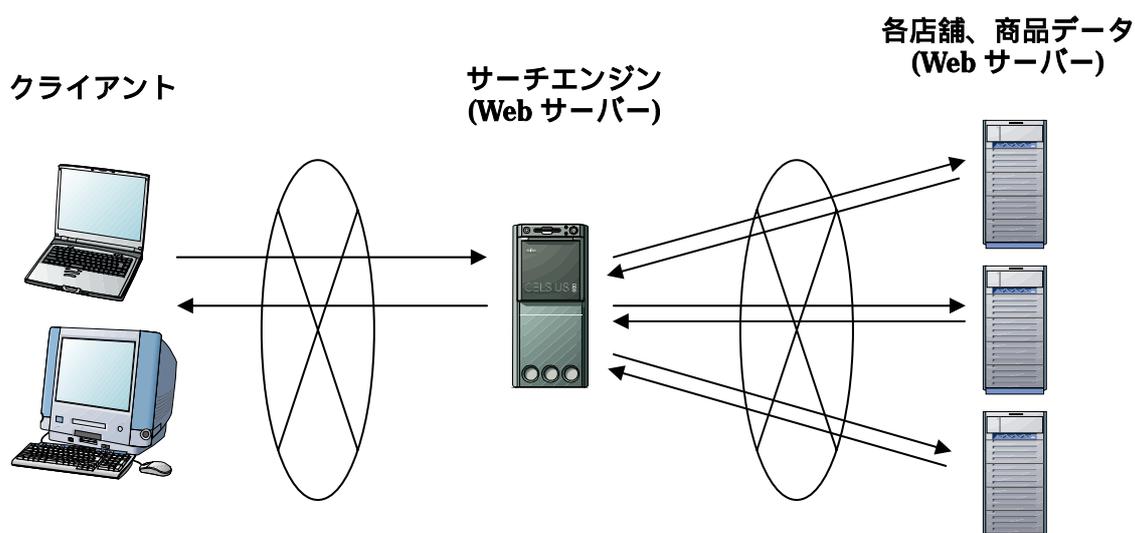
同一の XML 文書をサーバーサイド、クライアントサイドで変換処理したものを表示した様子

3.2. サーチエンジン(サーバーサイドプログラム)について

3.2.1 概要

本研究では、顧客(クライアント)の端末から各店舗に存在している商品データを検索するために、サーチエンジンプログラムを作成した。

ここでは、検索プログラムが1つのサーバー上で動作するサーチエンジンを解説する。このプログラムは以下のような動作を行う。



クライアントが商品データの検索を要求する。<送信データは条件フォーム>
エンジンの Web サーバーが各店舗に商品データを要求する。

各店舗は商品データを Web サーバーに送る。<送信データは XML>

店舗から送信された XML データから で取得した条件に合致するものを抽出する。
結果を送信する。<送信データは HTML>

3.2.2 サーバーサイドプログラミング

サーバーサイドプログラミングとは、前節のように、クライアントからの要求を Web サーバーで受けて、Web サーバーではデータの取得と検索を行うような方式をいう。

したがって、検索プログラムの負荷がかかるのはサーチエンジンの Web サーバーだけである。クライアントは結果を表示するだけなので負荷はかからない。各店舗の Web サーバーもデータを送信するだけなので負荷はかからない。

3.2.3 ASP

ASP はサーバーサイドプログラミングを実現する方法の一つである。ASP は Microsoft 社が開発した処理環境で、VisualBasic に似た特徴を持つ言語(VBScript)で処理を記述し、

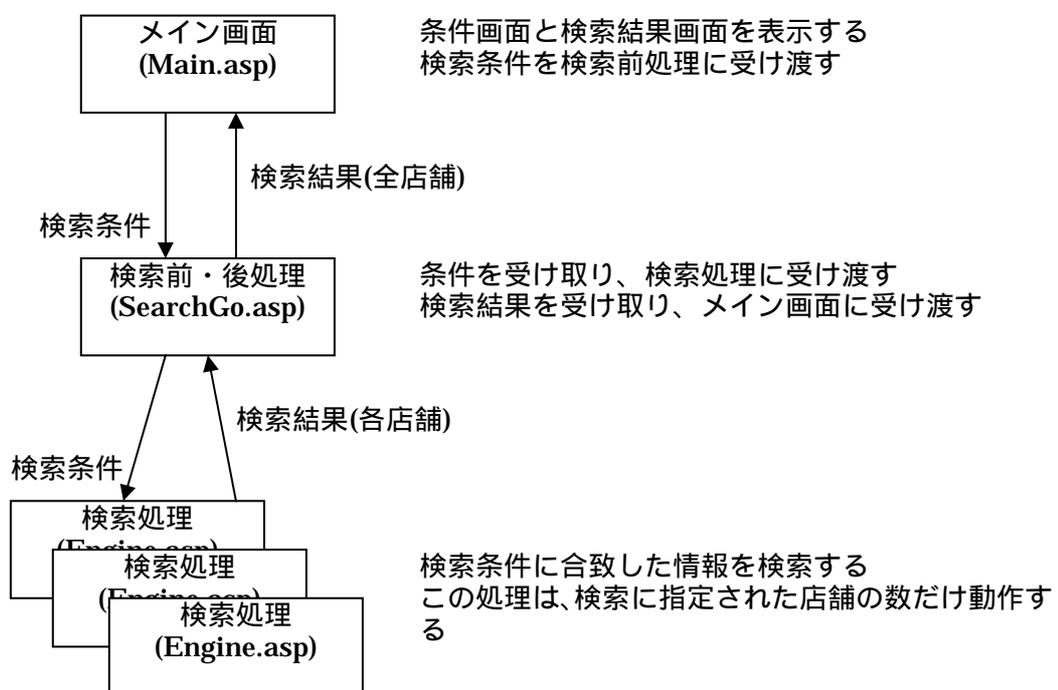
Microsoft 社の Web サーバー「Internet Information Server」「Personal Web Server」上で動作する。

また、ActiveX 技術で作成されたコンポーネントを使用することができるので、外部プログラムの連携も比較的容易に行うことができる。

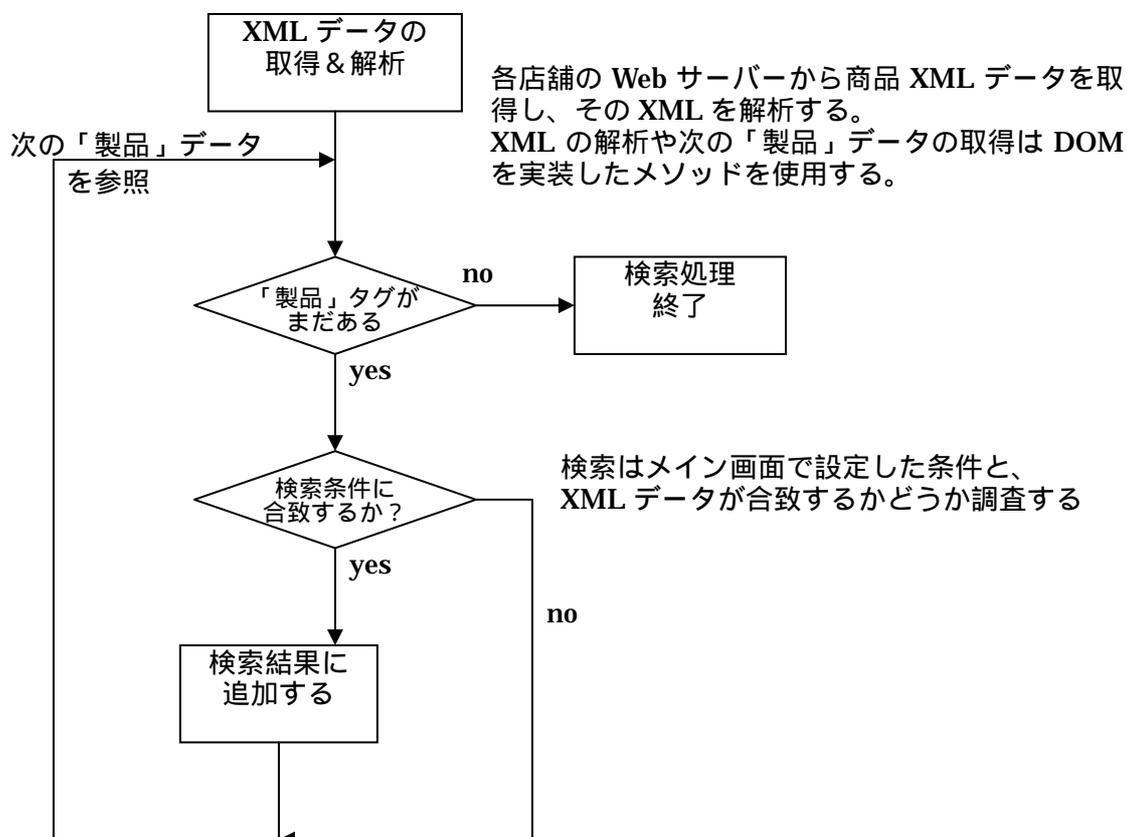
今回の開発は Microsoft 社が提供している XML 解析コンポーネント(MSXML)を利用して行った。MSXML は XML のプログラミングインターフェース DOM を実装している。

3.2.4 サーチエンジンプログラムの概要

サーチエンジンは以下のような処理を行っている。()内は実際のモジュール名

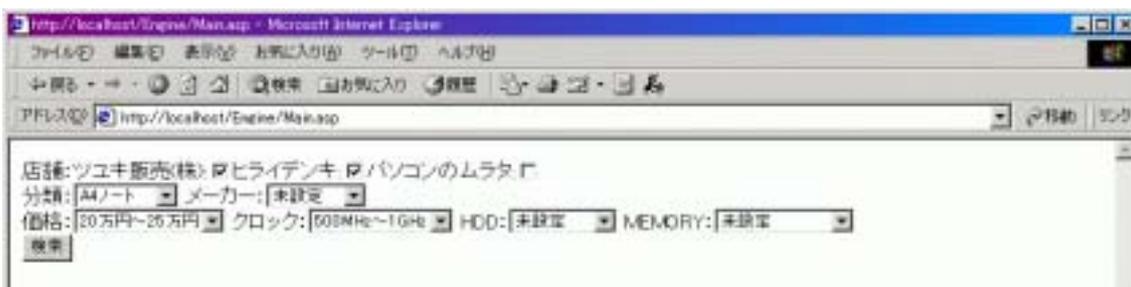


検索処理(Engine.asp)は以下のような処理を行っている。



3.2.5 実行画面

実際に作成した検索画面は以下のようなものである。



「店舗」は、検索の対象となる XML ファイルを提供している店舗を複数選択できる。

「分類」「メーカー」などは検索条件であり、この検索条件を基にして XML ファイルからデータを取得する。「未設定」は条件をつけない設定である。

「検索」ボタンを押下すると、検索が実行される。

http://localhost/Engine/Main.asp - Microsoft Internet Explorer

アドレス http://localhost/Engine/Main.asp

店舗: ツユキ販売(株) | ヒライデンキ | パソコンのムラタロ

分類: ノート | メーカー: [未設定]

価格: 20万円~25万円 | クロック: 500MHz~1GHz | HDD: [未設定] | MEMORY: [未設定]

検索

検索結果は6件です。

店舗名	メーカー	品名	型番	価格	スペック	リンク
ツユキ販売(株)	hp	omnibook 500 P600 12X 128/10 CD 98S CX	F2165W#ABJ	218000	Windows 2000 Professional Service Pack 2, Intel製 Pentium III 600MHz, 128MB, 12.1インチ液晶, 10GB, CD-ROM幅280×高さ25×奥行220mm, 1.6kg インテルR SpeedStep?テクノロジー対応低電圧版モバイル PentiumR III 600MHz, 納期2週間。	
ツユキ販売(株)	Sony	VAIO Note FX	PCG-FX77G/BP	229800	Windows Me, Intel製 Mobile Pentium III 850MHz, 128MB, 15インチ TFT液晶, 30GB, CD-RW/DVD-ROM幅324×高さ38.5×奥行265.5mm, 3.2Kg	
ツユキ販売(株)	TOSHIBA	DynaBook A2/590PMCモデル	PAA2580PMC	218000	Windows 98 Millennium Edition, Intel製 Pentium III 800MHz, 128MB, 15インチ液晶, 30GB, CD-RW/DVD-ROMマルチドライブ幅328×高さ277×奥行44.7mm, 3.5kg Intel SpeedStepテクノロジー対応モバイル Pentium III プロセッサ, Fine SuperView液晶: 15型FLサイドライト付き TFT液晶 1,024×768ドット: 1,677万色, 納期1週間。	
ヒライデンキ	TOSHIBA	DynaBook A2/590PMCモデル	PAA2580PMC	218000	Windows 98 Millennium Edition, Intel製 Pentium III 800MHz, 128MB, 15インチ液晶, 30GB, CD-RW/DVD-ROMマルチドライブ幅328×高さ277×奥行44.7mm, 3.5kg Intel SpeedStepテクノロジー対応モバイル Pentium III プロセッサ, Fine SuperView液晶: 15型FLサイドライト付き TFT液晶 1,024×768ドット: 1,677万色, 納期1週間。	
ヒライデンキ	hp	omnibook 500 P600 12X 128/10 CD 98S CX	F2165W#ABJ	218000	Windows 2000 Professional Service Pack 2, Intel製 Pentium III 600MHz, 128MB, 12.1インチ液晶, 10GB, CD-ROM幅280×高さ25×奥行220mm, 1.6kg インテルR SpeedStep?テクノロジー対応低電圧版モバイル PentiumR III 600MHz, 納期2週間。	
ヒライデンキ	Sony	VAIO Note FX	PCG-FX77G/BP	229800	Windows Me, Intel製 Mobile Pentium III 850MHz, 128MB, 15インチ TFT液晶, 30GB, CD-RW/DVD-ROM幅324×高さ38.5×奥行265.5mm, 3.2Kg	

ページが表示されました

インターネット

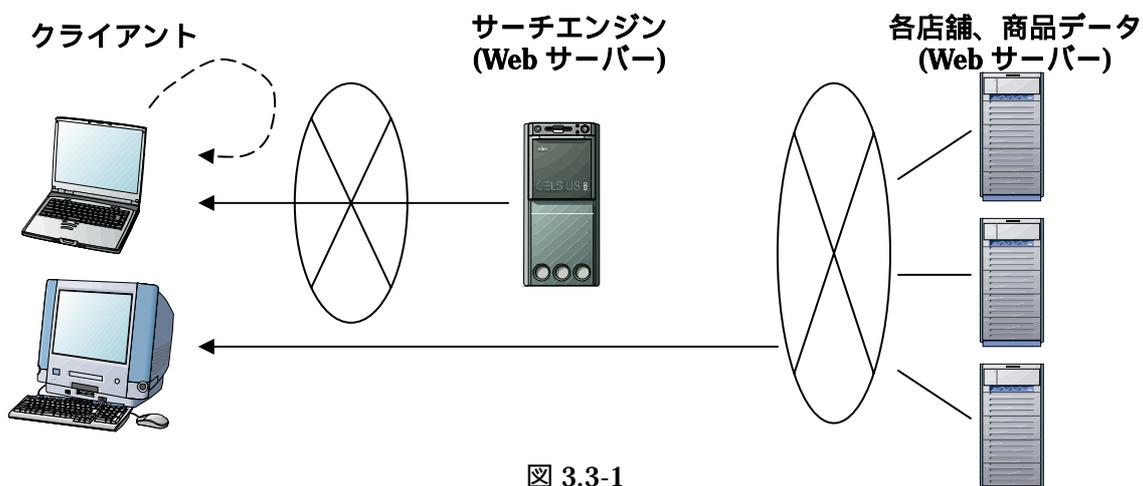
検索結果は、条件画面の下方に挿入される。

「リンク」ボタンはメーカーが提供しているパソコンの Web ページへのリンクである。

3.3. サーチエンジン(クライアントサイドプログラム)について

3.3.1 概要

クライアントサイドの検索では、データの流りは下図の様になる。



検索画面の表示

サーバー上の HTML ファイルをブラウザで表示する。

この時検索用の JavaScript はクライアントのメモリ上にロードされる。



図 3.3-2

検索条件の受け渡し

クライアントの画面上で選択されている条件を検索プログラムが受け取る。



図 3.3-3

検索

XML を格納するオブジェクトをクライアントのメモリ上に生成し、生成されたオブジェクトにサーバー上の XML ファイルを読みこむ。

検索結果の表示

JavaScript から直接 HTML を書き出し、画面に検索結果を表示する。



図 3.3-4

3.3.2. クライアントサイドプログラミング

JavaScript によるクライアントサイドプログラミングでは、ユーザーからの要求はサーバーを介することなくブラウザ上で解釈されて実行される。

3.3.3. JavaScript

Netscape Communications によって開発されたスクリプト言語。Java とはまったく別物。

クライアントでの JavaScript は HTML の<Script>タグ内に処理を記述し、ページを読み込む際に Web ブラウザに搭載されたスクリプトエンジンで実行することになる。条件分岐によってそれぞれ異なった HTML を表示させたり、値の入力や選択などフォームに対する操作に対して、特定の関数を実行させたりすることができる。

3.3.4. サーチエンジンプログラムの概要

1. XML を格納するオブジェクトをクライアントのメモリ上に生成する。

```
DomObj = new ActiveXObject("Microsoft.XMLDOM");
```

2. 生成されたオブジェクトにサーバー上の XML ファイルを読みこむ。

```
DomObj.load("http://www.hoge.com/foo.xml");
```

3. "パソコン/製品/*"の NodeList オブジェクト(ノードの子要素リスト)を取得する。

```
selnode=DomObj.selectNodes("パソコン/製品/*");
```

4. 各 NodeList の要素と値を取得する。

要素 : selnode(x).nodeName

値 : selnode(x).text

5. 一製品分の全要素を取得したところで、画面から入力された検索条件と比較を行い、条件を満たすものは検索結果画面に表示、満たさない場合には次の製品を読みこむ。

6. NodeList の最後まで読みこんだら終了。

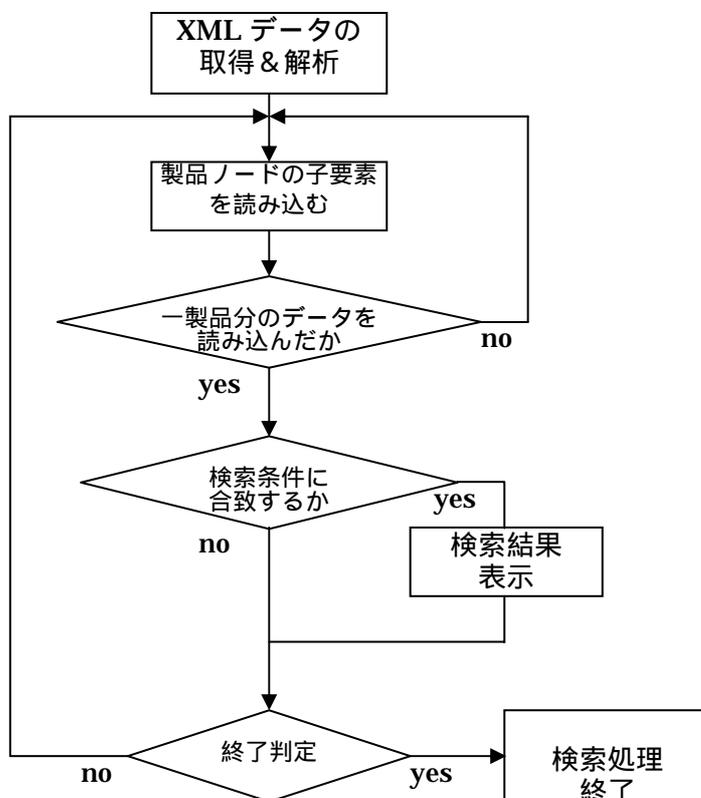


図 3.3-5

3.4. 評価・考察

今回のシステムの作成について、以下の考察を行った。

3.4.1. 店舗について

今回の XML でデータを管理する店舗のケースの利点としては、HTML と異なり、データ部分(XML)と見た目(レイアウト)(XSL(T))部分を別に分けることにより、一度レイアウトを作成しておけば、価格の更新があった場合はデータ部分のみの更新で済み、またサイトのデザインを変えたい場合は、レイアウト部分の更新で済む。データとレイアウトの分離は HTML でも、HTML4.01 と CSS の様に奨められているが、HTML そのものが文章を公開するためのレイアウトを目的とした SGML アプリケーションであることから分離にも限界がある。

従来の販売店の Web サイトは HTML 文書で公開されている。しかし、内部的にどのように商品データ、価格データを管理しているかは販売店毎のシステムによって異なるはずであり、データベース、表計算文書、または、紙の帳簿で管理しているかもしれない。また、HTML 文章もそれらの管理している情報から、アクセスがある度にデータベースから最新の情報を元に ASP、PHP、Perl 等で動的に作成しているかもしれないし、目で読み取りながらエディタやオーサリングソフトで作成しているかもしれない。

紙での管理は別にして、上記利点はデータを電子的に管理していることが前提となるが、システム全体を作り替えるのではなく、そういった従来のシステムを元にデータを外部とのやり取り・公開する部分に XML を導入することを考えた。XML 対応データベースと呼ばれるものは大きく分けて 2 種類あり、オブジェクト指向の実装がされた XML のツリー構造のイメージでデータを格納するタイプの XML データサーバー、入出力する形式に XML を使用すると内部的に表形式に変換して格納してくれる XML 対応リレーショナルデータベース(RDBMS)があるが、XML データサーバーは発展途中であり、現時点でもデータの管理はトランザクションやパフォーマンス、今までの遺産の面でも RDBMS が優れている。また、XML には適する分野とそうでない分野があり、XML にはデータ保存だけでなく、今回の様な表示や、転送にも使う形式であることから、XML とデータベースの連携システムの方が実際には利用されることが多くなると考えられる。

3.4.2. サーバーサイドプログラムについて

サーバーサイドプログラミングの利点

- ・ クライアント(顧客)は Web ブラウザだけで、サービスを受けることができる。
- ・ クライアントや、XML データが存在する Web サーバー(各店舗)に負荷がかからない。
- ・ クライアントに XML データが直接到達しないので、各店舗に存在するデータの隠蔽性がある。

サーバーサイドプログラミングの欠点

- ・ Web サーバーに負荷が集中することがある。ただし、性能の良いサーバーを使用する、複数の Web サーバーを使用して負荷分散を行う、などの解決策が存在する。

3.4.3. クライアントサイトプログラムについて

クライアントサイドプログラミングの利点

- ・ クライアントサイドで検索する場合、検索条件をサーバーに送信する必要が無い。
- ・ Web サーバーに負荷がかからない。
- ・ JavaScript が記述された HTML ファイル(サーチエンジン)が最初からクライアントに存在する場合、サーチエンジンを置く Web サーバー自体必要なくなる。

クライアントサイドプログラミングの欠点

- ・ クライアントにかかる負荷が大きい。
- ・ クライアントの環境に依存する割合が大きい。(例：ブラウザの種類やバージョン)

3.4.4. ASP について

ASP を利用する利点

- ・ サーバーサイドプログラミングを、比較的容易な言語(VBScript)で記述することができる。
- ・ Microsoft 社の IIS に付属しているので、容易に入手することができる。
- ・ 技術者が多いので、問題が発生したときにネット上の情報が検索しやすい。

ASP を利用する欠点

- ・ Microsoft 社の IIS でしか動作しないので、Web サーバーを Windows 以外で構築する場合は使用できない。Windows と UNIX 系 OS 双方で運用するには Web 版 Java アプリケーションである、Servlet などの使用が適当である。
- ・ VBScript なので、Java や C++のようなオブジェクト的なプログラミングができない。そのため、多人数の技術者が平行して開発し、メンテナンスすることが難しい。IIS を利用するのは検索エンジン部分のみで、各店舗は IIS でなくてもかまわない。

3.4.5. XML について

XML を使用する利点

- ・ 今回は、商品データとして各店舗で統一されたフォーマットを使用していたが、統一されていなくても、サーバーサイドの検索が可能である。
XSLT などのデータ形式変換などを經由する必要がある

XML を使用する欠点

- XML はテキストファイルであり、読み込みは XML 解析を行わなければならないので、商品データが大量の場合は、性能に問題があると考えられる。
そのような場合はデータベースを使用する必要がある。
- XML データがインターネット上に存在しているので、セキュリティを考える必要がある。

4. XML での開発を終えて

4.1. XML の可能性等

現在、XML が抱える問題点や今後の課題、将来性等に関しては既に書籍等のさまざまなメディアで述べられている通りである。

本論文の結論として、今回我々が XML を使用したアプリケーションとして『検索システム』を開発した際に感じた点や XML の可能性等を考察する。

4.2. 従来の検索システムとの比較

今回、我々が作成した検索システムと同処理を、従来の Web 検索サービスで行う場合はどうするか。サーチエンジンを使用したりして Web サイトにアクセスし、目的の情報が載った Web ページを探し、ページを見て少しずつ自分に必要な情報を集めていく。

また、検索サイトの多くが、複数の Web サイトにアクセスして必要な情報を収集してくる巡回エージェントやロボットと呼ばれるプログラムを使用している。

だが、ロボットを使って収集したデータは、「意味」まで分析できるわけではない。

例えば、「A4 ノートパソコン」の価格調査をしようとして検索エンジン等を利用して、「A4 ノートパソコン並みの性能を持った B5 ノート...」や「A4 ノートパソコン用パソコンバッグ...」等を紹介したサイトの URL までもが表示される。

結局、本当に必要な情報を手に入れるには、最終的には利用者が検索結果を見なければならぬ。

それに比べて、今回我々が作成した検索システムでは、XML 文書を検索しているため、ユーザーが要求した結果のみを返す事が可能である。これは、XML 文書のデータには「意味」を持たせることができるためである。

実際に、メーカーやショップの Web ページが XML で書かれ、<製品>や<型番>等の共通なタグが使われるようになれば、現実の Web 上でも実現可能なシステムである。

更に、XML を使用すれば、Web ページを「見る」ものから「自動的に処理する」ものへ変えることもできる。今回は実現できなかったが、「検索結果より購入希望商品を店舗に発注する」といった商取引システム(BtoC)に発展することも可能である。

4.3. XML によるシステム構築の利点

今回のサンプルでは、インターネット環境での使用を前提に作成したが、システム要件によっては、LAN 環境でハードディスクが共有されて、データベースや表計算ソフト等を使用すれば XML と同じ事は実現可能な場合もある。だが、XML の利点はその導入のし易さにある。

複数のマシンにデータベースや表計算ソフトを実装するためのコスト面で有利である事は当然であるとして、データベースや表計算ソフトでは、HTML と同様にデータとレイア

ウトの情報が一体化しているという欠点がある。そのため、ファイルのレイアウト変更の都度、ファイル同士の連携が取れなくなることが想定できる。

どの方法を選ぶのが正解であるかは、システム要件によって異なるので一概には言えないが、XML ベースのアプリケーション開発は、「実現可能であるが、ちょっと面倒」と思っていた仕組みを、比較的容易に実現できる点が利点であると言える。

4.4. XML がソフトウェア産業にもたらす変革

XML がソフトウェア産業にもたらすインパクトは、3 点に大別される。

1 点目は、「アプリケーションの選択方法」が変わる点。従来は、あるアプリケーションを使い始めたら、それをいつまでも使い続けるのが大半であった。これは、多くのアプリケーションが、専用のデータ形式を備えることに起因するが、データ形式が XML で標準化されたら、アプリケーションがデータを束縛することは無くなる。その結果、ユーザーは使い勝手等でアプリケーションを選択できるようになる。

2 点目は、「データ交換の方法」が変わる点。データを XML 化すると、アプリケーション・レベルだけでなく、システム間や企業間でデータを交換する場合も、データを変換する必要が無くなる。

3 点目は、「アプリケーションの作り方」が変わる点。XML を使用すれば、従来人間の判断が必要だったアプリケーションでも、データに意味を持たせる事によって自動化する事ができる。これを実現するには、クライアントからサーバーまであらゆるプログラムを XML に対応させなければならない。「XML データを認識してそれを処理し、また XML データに返す」というインターフェースを実装する必要がある。

4.5. XML を取り巻く周辺技術

また、XML を本格的に活用するための、データ構造の変換、ビジネスプロセスの実装、データベースとの連携技術等の周辺技術も整いつつある。

XSLT は、異なるスキーマ間でデータ変換を行う周辺技術である。XML はユーザーが自由に定義できるが、これは XML を利用する上で大きなリスクになる可能性もある。それぞれのユーザーが勝手に文書構造を決めてしまうと、相互のやり取りが難しくなってしまう。この問題はすでに現実のものになっている。業界内で XML を標準言語として利用するには、業界標準の文書定義が必要になるが、業界標準ができる前に XML の活用を始めているケースがあるからである。XSLT の最大の目的は、このような問題を解決することにある。スキーマ間でデータ交換ができれば、新たに業界標準が策定されても、既存データを XSLT で変換することで対応できる。このような柔軟性は、XML への取り組みを加速する要因となっている。

企業間取引を行うには、プロトコルの標準化も重要になる。XML は単なるデータ記述言語であり、これだけでは、企業間取引を行うことはできない。取引に必要なドキュメント

の構造は勿論のこと、メッセージの送受信手順等を明確にしておかなければならない。このような目的で利用されるのが、標準ビジネスプロトコルである。ビジネスプロトコル策定の動きは 1999 年から本格化しており、各種業界団体や公的機関、有力企業による標準プロトコル策定が進められてきた。代表的なものとしては、情報機器と電子部品業界団体による標準フレームワーク「RosettaNet」や、UN/CEFACT と OASIS によって策定されている電子ビジネス向け汎用プロトコル「ebXML」等がある。

データベースとの連携技術に関してであるが、XML ドキュメントを保管するための最もシンプルな方法は、XML ドキュメントをそのままファイルシステムに格納する方法である。しかしこの方法は、扱うドキュメントの量が増大した場合には問題が発生する。テキストファイルは検索に時間がかかるため、レスポンスが著しく悪化する可能性がある。データアクセスを高速化するには RDBMS が向いているが、RDBMS が提供するリレーショナル構造は、XML の階層構造とは大きく異なる。そこで XML と RDBMS を連携させるための仕組みが必要となる。一昨年从去年にかけて、XML 対応機能を実装した RDBMS がリリースされている。

4.6. XML 技術者認定制度

XML は通常のコンピュータ言語とは異なり、データ構造の設計・定義等の要素が含まれ、技術者の技量によって、その成果物の品質に影響がでる可能性があることから、以前から「ベンダーニュートラルな XML 技術者制度」発足の要望が高かった。そこで、2001 年 10 月より「XML マスター」が発足・運営が始まった。「XML マスター」は標準的な XML の技術力を測る指針となるであろう。

4.7. 最後に

前述した通り、代表的なビジネスプロトコルはすでに実用期に入っており、各社からリリースされるソフトウェア製品も XML 対応のものが増えている。各企業の XML への取り組みも積極化してきた。一時に比べれば、「XML にたいする注目度が下がっている」といった論評もあるが、実際には XML 関連のニュースは増える一方であり、XML 採用の事例も依然として増加傾向にある。つまり、XML の普及は緩やかに加速しつつ、堅実に進行中といえる。

XML の本質と周辺技術に対する理解を深めておくことが、我々 IT 業界関係者にとっての今後の課題である。

略語・用語

・ ARPAnet (Advanced Research Projects Agency network)

インターネットの原点ともいえるネットワーク。1969年に米国国防総省（U.S. Department Of Defense : DOD）の高等研究計画局(Advanced Research Projects Agency : ARPA)が、大学のコンピュータを結んで実験的に始めた。当時は東西の冷戦時代であり、核攻撃によっても情報の伝達が途絶えない通信経路を研究する目的があった。ARPAnetで使用されていたネットワークプロトコルであるTCP/IPは、その後、国防総省によって改良が加えられ、UNIXワークステーションでは標準プロトコルになっている。また、インターネットの標準プロトコルでもある。

・ ASP (Active Server Pages)

IIS 3.0でサポートされた、WWWサーバー側でJScriptやVBScriptを実行するスクリプティングエンジン。

・ BtoC (Business to Consumer)

電子商取引の1つの形態を示す言葉である。BはBusinessの略、CはConsumerの略である。企業間の取引引き(BtoB)ではなく、消費者向けの小売りを意味する。インターネット上のオンライン店舗などによるさまざまなサービスや物品の販売を示す。

・ DOM (Document Object Model)

W3Cが策定している、プログラムやスクリプトを使って、コンテンツを動的に変更するためのインターフェース。

・ HTML (HyperText Markup Language)

HTMLは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WWWサーバーで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。WWWサーバーのドキュメントでは、このHTMLにより文書の構造や体裁などの要素を定義し、イメージやURLを貼り込んで、ハイパーテキストを実現している。

・ JScript

MicrosoftによるECMA 262規格仕様の言語 (ECMAScript Edition 3)。以前のバージョンとの互換性を含むいくつかの例外事項を除いては、JScriptはECMA規格に完全に準拠している。

・ **OASIS (Organization for the Advancement of Structured Information Standards)**

OASIS は、ビジネスにおける情報交換用技術標準を作成する国際的な団体である。XML や SGML などの標準技術をベースに活動している。組織でも個人でも参加することができる。

情報交換においては、国際的に統一された標準の存在が非常に重要である。それがあって初めて、不特定多数の個人や組織間での情報交換が実現する。そういった、既存の XML や SGML などの標準技術の上で、実際のビジネスアプリケーションに適合し、相互運用を実現するための標準を定める団体として、OASIS はある。

インターネット上の標準化団体としては W3C も存在するが、W3C は XML や HTML のような、Web に関する基礎技術を扱い、OASIS はその上位のアプリケーション寄りの標準化を扱う。

・ **SGML (Standard Generalized Markup Language)**

マークアップ言語のひとつで、汎用マークアップ言語規約ともいう。ISO8879 として規格化されている。文書の中にタグを埋め込んで文書の構造を記述する。複数で分担してひとつの文書を作成する場合、スタイルの統一が図りやすく効率がよい。各国の政府機関や米国防総省の公文書フォーマットとしても採用されている。

・ **UN/CEFACT (United Nations Centre for Trade Facilitation and Electronic Business)**

国連による行政、商業、運輸のための実務と手続き簡易化センター

・ **URL (Uniform Resource Locator)**

インターネット上のリソースの位置を示す、リソースへアクセスするためのスキーム(通常は通信プロトコル名)とリソースの名前をセットにした表記法(仕様)のこと。URL で指定されるプロトコルとしては、http、ftp、gopher などがある。RFC1738、1808 で規定され、その後の標準化の過程で、より包括的な URI の一部となっている。

・ **VBScript (Visual Basic Scripting Edition)**

プログラミング言語である Visual Basic 言語シリーズの最新作で、広範囲の環境下でアクティブなスクリプト処理を実現する。Microsoft Internet Explorer は、Web クライアント スクリプト処理を行い、Microsoft Internet Information Service は、Web サーバー スクリプト処理を行う。

・ **W3C (World Wide Web Consortium)**

WWW に関する標準化組織。(http://www.w3.org/)

- ・ Web ページ (Web Page)

Web ブラウザで表示されるコンテンツのこと。ホームページと呼ばれることが多いが、インターネットの WWW サービスにおいて、あるサイトあるいはひとまとまりのコンテンツを代表する Web ページ。通常はそのサイトのトップページやコンテンツの目次ページがホームページになる。

- ・ WWW (World Wide Web)

欧州核物理学研究所 (European Center for Nuclear Research, CERN) で開発されたドキュメントシステム。ドキュメント中に他のドキュメントへのジャンプコマンドを埋め込むことができる。このジャンプコマンドとしては、URL (Uniform Resource Locator) を使用し、インターネットに参加している世界中の WWW サーバのどのドキュメントにもジャンプすることができる。

- ・ XML (eXtensible Markup Language)

Internet 上での SGML の利用を容易にすることを目的として設計されたマークアップ言語。HTML のような固定のマークアップ方法だけでなく、文書独自のマークアップ方法を定義できるようになる。1998 年に W3C により標準化勧告され、現在はインターネットのさまざまな分野での応用が進められている。

- ・ XSL (eXtensible Stylesheet Language)

XML 専用のスタイルシート言語であり、変換言語としての機能と、実際にテキストをフォーマットする言語の 2 種類の言語が含まれていた。この中で、変換言語の部分は非常に有用で、XSL 以外の目的でも役に立つことから、分離独立し XSLT(XSL Transformations) となり、現在 2002 年 1 月現在、XSLT1.0 が W3C の勧告、XSLT2.0 がドラフトとなっている。

- ・ XSLT (eXtensible Stylesheet Language Transformations)

XSLT は、任意の XML 文書を読み込んで、それを加工して出力する簡易なスクリプト言語として使用することができる。たとえば、ビジネスデータを読み込んで、それを SVG 形式で書き出すことでグラフを自動作成する、といった使い方も可能である。

また、出力は XML 文書とは限らず、プレーンテキストや HTML を出力することができる。そのため、XML 文書を HTML に変換するために使用されることも多い。

しかし、XSLT は万能ではなく、あくまで、スタイルシート変換用の言語である。すべてのニーズに対応することはできない。

参考文献

- 参考図書 -

(図解入門)XML技術

著 株式会社テクノエスト 協力 引地一男
株式会社アスキー

(まるごと図解)最新XMLがわかる

著 池田実/小野寺尚希
技術評論社

日経ソフトウェア

日経BP社

標準ASPテクニカルレファレンス

著 山田祥寛
SOFTBANK Publishing

ステップバイステップで学ぶXML実践講座

著 Michael J. Young
日経BPソフトプレス

XMLハンドブック

著 渡辺 竜生
ソフトバンクパブリッシング株式会社

XML+XSLによるWebサイトの構築と活用

著 PROJECT KySS 宮川 雅輝
ソフトバンクパブリッシング株式会社

お気楽Q&A XML

株式会社日本ユニテック おおた じゅん
株式会社IDGジャパン

- 参考 Web -

"情報検索の謎と不安"

<http://www11.nikkeibp.co.jp/jp/articles/features/970804/kaku7.html>

"MSDN Online - XML Developer Center"

Microsoft Corporation (<http://www.microsoft.com/japan/>)

<http://www.microsoft.com/japan/developer/xml/default.asp>

"Active Server Scripting Guide"

http://activex.adsp.or.jp/japanese/active_server/index.htm

"ASP Developers Network"

<http://www.aspdeveloper.net/>

"USERS GROUP Web Site"

<http://www.users.gr.jp/>

"たのしい XML "

<http://www.cityfujisawa.ne.jp/~yanai/xml/>

"MSDN Online - XML Developer Center"

Microsoft Corporation (<http://www.microsoft.com/japan/>)

<http://www.microsoft.com/japan/developer/xml/default.asp>

"@IT XML 技術者のための情報提供 / 情報交換フォーラム"

<http://www.atmarkit.co.jp/fxml/>

「XML 技術研究会」に参加して

今までメインフレーム系の業務経験が主であった私にとって、XML 部会での研究活動は大変有意義なものでした。

Web 技術の経験が殆ど無い私にとって XML という最新技術の調査・研究・開発を行う事は大変貴重な経験になりました。また、文献や Web の情報を調査するだけでなく、XML での開発経験のあるメンバーの話を聞くのは大変参考になりました。

約半年間という短い期間で、業務との調整を行いながら研究活動を行うのは大変な面もありましたが、研究会メンバーや職場の皆様の協力によって何とか最後までやり遂げる事が出来ました。関係者の皆様に感謝申し上げます。

私は、実際の業務で XML をデータとして使っていました。これは独自の簡易的な XML パーサーと簡易的な XML でした。

業務でも、将来的には DOM など一般的なインターフェースを使用することになるのがわかっていたので、今回の研究は実務に直接利用できると思います。

DOM を使用することでインターネットを経由して XML 文章を管理する具体的な方法を学べたのも大きかったです。

今後は、ASP だけでなくいろいろな言語から DOM を使用できるくらいのスキルを身につけていきたいと思います。

研究会という形で複数の人と新しいことを調べるということは、私自身初めての経験でとても新鮮でした。

遠隔での作業ということで、メーリングリストや Web 上の共有フォルダを利用した作業でしたが、その便利な点や不便な点を知ることができたのも貴重な経験になりました。今回学んだ技術や経験を今後の業務で活かせるようにしていきたいと思います。

最後に、限られた時間の中での共同作業という厳しい条件の中でしたが、皆様の協力によって何とか最後までやり遂げることができました。本研究会メンバーの皆様及び関係者の方々に深く感謝いたします。

この研究会に参加することになるまで、「XML」という言葉さえ知らず全く 1 からのスタートでした。研究会で、実際に XML を扱うプログラムを作成してみるなどして、XML とはどのようなものかがだんだんと分かってきました。また、この研究会ではメーリングリストを活用するなど初めての体験も出来ました。

これまで、仕事においても私生活においてもインターネットにふれる機会がほとんど無かったので、とても良い勉強になりました。ありがとうございました。

XML についての知識を何も持たずに臨んだ XML 部会。最初はとにかく分からないことだらけで、「XML とは文書の保存形式のことである」ということを理解するまでにもかなりの時間を要した。

それでも、半年強の間 XML に触れたことで理解が深まったし、IIS や ASP などについての知識も得ることができた。今回の部会への参加は、自分にとって非常に有益なものであったと思う。

お世話になった皆様に、この場をお借りして御礼を申し上げたい。