

学校情報教育におけるプッシュ配信技術利用の可能性

梅津 彰能

Akiyoshi Umezu

aumezu@cs.reitaku-u.ac.jp

(麗澤大学国際経済学部国際経営学科 4 年)

Abstract : 現在、文部省の指導や先進的な研究プロジェクトの取り組みにより学校へのネットワーク接続が急速に普及している。これによって、教室からネットワークに接続することで必要な情報が容易に Web 等で入手できるようになってきた。しかし、学校教育を考えると単に Web 検索による利用ではネットワークを十分に利用しているとは言いがたい。本稿ではネットワークの有効利用の一方策として、プッシュテクノロジーと、現在注目されている XML を使用し、簡単なシステムを実験的に構築し、その有用性を検討した。プッシュテクノロジーとは更新されたデータが即座に手元に届けられる技術で、情報提供の効率化が可能となる。さらに、データ形式として XML を用いることで情報共有も容易になる。学校に設置される教室用 PC 環境のトラフィック抑制の一方策となろう。

Keywords: Push-Technology, XML

1. はじめに

インターネットの急速な普及とともに、Web 上の情報も広い範囲に及び、活用性が高くなってきている。今や、必要な情報は Web 上で入手できるようになった。

インターネットの普及とともに学校のネットワーク接続の普及も進んでいる。文部省¹においても、「学校の情報通信ネットワークの整備、すべての学校をインターネットに接続する」とし、平成 13 年度までにすべての小・中・高等学校、特殊教育諸学校をインターネットへ接続するように提言している²。

しかし、ネットワーク整備が進むにつれ、ユーザ数が増加し、ネットワークトラフィックの急激な増加を招いてしまっている。これらの原因の一つとしてサーバへの不要なアクセスが考えられる。

本稿ではネットワーク有効利用のための手段として XML を用いたプッシュテクノロジーを用いた Web サービスの形態を提案する。

2. プッシュテクノロジーとは

プッシュテクノロジーについて Web ページを例にとって説明していくことにする。

2.1 Web ページの配送モデル

サーバからブラウザへ Web ページを配送するモデルは大きくプル型とプッシュ型に分けることができる。

この二つの大きな相違点は Web ページをいつ配送するかを決めるのは誰かという点にある。プル型ではブラウザであるのに対し、プッシュ型ではサーバがこれを決める。また、プル型はさらにノーマルプル型とスマートプル型の 2 つに分類することが出来る。

2.2 ノーマルプル型モデル

Web ページ配送の最初のモデルで、ユーザ側から見ると、ブラウザがサーバから Web ページを引っ張ってくるように見えることから、ノーマルプル型と呼ばれている。

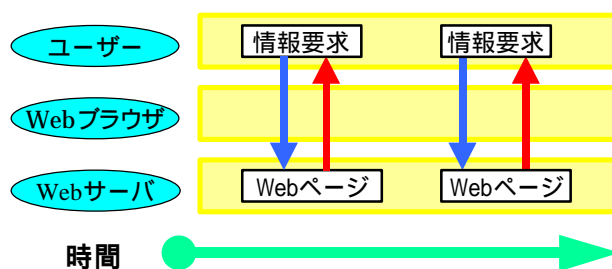


図 1. ノーマルプル型配送モデル

2.3 スマートプル型モデル

スマートプル型モデルはノーマルプル型モデルの次に登場した技術である。ノーマルプル型モデルとの違いは、ブラウザとユーザの関係である。ノーマルプル型モデルは、ユーザが Web ページを要求したときにブラウザはサーバへアクセスするが、スマートプル型モデルではユーザに要求されたときにサーバ

¹ <http://www.monbu.go.jp>

² <http://www.monbu.go.jp/news/00000307/>

にアクセスをするのではない。ユーザはあらかじめ、ブラウザの設定で時間間隔と URL を指定する。ブラウザは設定従って、定期的にサーバにアクセスする。ユーザが Web ページを見るときにはブラウザがサーバからすでに受け取っているの、直ちに参照することが可能になる。

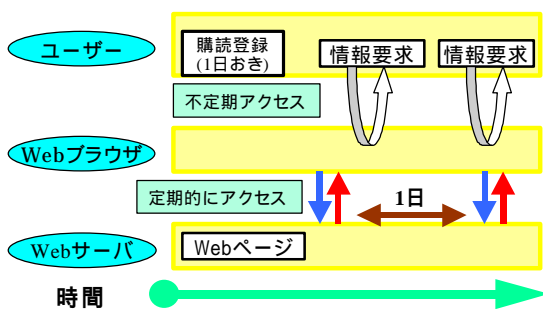


図2. スマートプル型配送モデル

2.4 スマートプル型の問題点

Web ページには 1 ヶ月ごとに更新されるものもあれば、1 日ごと、1 時間ごとに更新されるものもある。ユーザはどんなタイミングで Web ページが更新されるのかわからないので、適切な時間間隔をブラウザに指定することができない。

例えば、Web ページの更新間隔が 1 時間に 1 回であって、ブラウザがサーバにアクセスする時間間隔が 15 分だとする。この場合、1 時間に 4 回もブラウザはサーバへアクセスすることになり、ネットワーク資源の無駄となる。また、逆に Web ページの更新間隔が 15 分に 1 回あって、ブラウザからサーバへのアクセスする時間間隔が 1 時間だとすると、更新された Web ページを見逃すことになってしまう³。つまり、スマートプル型モデルには、指定した時間間隔で Web ページを要求する機能は持っているが、どんな時間間隔で Web ページが更新されるかを知るための機能はないのである。

2.5 プッシュ型モデル

プッシュ型では、指定された時間間隔によって Web ページが更新されたかどうかを確認する。情報配信ファイルに時間間隔と URL が

³ ISDN 間欠接続によるネットワーク内にこのようなアプリケーションが設定されると意図しない発呼が頻発することになる。スマートプル型のアプリケーションを利用するには注意が必要となる。

書いてあるので、スマートプル型と比べると、ネットワーク資源の無駄遣いはかなり抑える事ができ、更新された Web を見逃すという心配がなくなるのである。

また、更新時間が指定されているので、更新時にブラウザからユーザへ通知することも可能である。

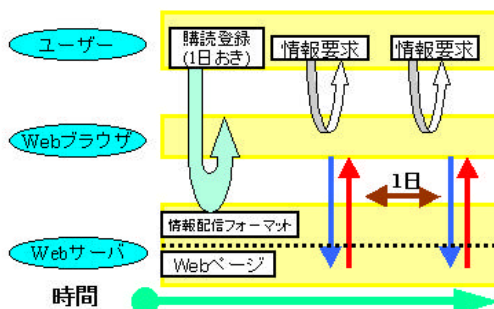


図3. プッシュ型配送モデル

しかし、常時、インターネットへ接続していなければ、その威力を発揮できないという欠点を持つ。ダイヤルアップ等でインターネットへ接続しているユーザーや間欠接続によるネットワーク間接続ではリアルタイムで情報を更新することができない⁴。

2.6 情報配信フォーマット

プッシュ型モデルを実現している情報配信ファイル。プッシュを実現させる為にブラウザや、サーバをメーカーが独自の形式で開発してしまうと互換性がなくなり、一般のユーザは情報の共有ができなくなる。また、プログラム資産の蓄積が困難となる。このため、独自形式ではプッシュ製品分のプッシュコンテンツ作成ツールや対応ブラウザが必要となり、ユーザにとっては好ましい状況ではない。

現在、利用できる情報配信フォーマットは次のとおりである。

- CDF(Channel Definition Format) チャンネル定義
- OSD(Open Software Distribution)⁵ ソフトウェア配布

2.7 CDF (Channel Definition Format)

CDFはMicrosoft社⁶からW3C⁷に提案された仕様で、XMLをベースとして設計されている。

⁴ 通常、間欠接続のネットワークではローカル側の利用要求によって始めてインターネットへ接続される。

⁵ 参考資料参照。

⁶ <http://microsoft.com>

⁷ <http://www.w3g.org/>

XML ベースの仕様を共有することによって互換性が生まれ、プログラム資産の蓄積が出来るようになってきている。

Microsoft 社の Internet Explorer のアクティブデスクトップにあるチャンネルや Point Cast 社⁸の Point Cast 2.8 は CDF を採用している。

3. プッシュテクノロジーの実際

ニュースの提供を始め、ソフトウェアやドキュメントの配布など、いろいろなアプリケーションにプッシュテクノロジーが利用され始めている。米国では Microsoft や Netscape に限らず、多数のプッシュシステム構築用製品が登場している⁹。

3.1 プッシュテクノロジーの形態

プッシュ技術は現在、大きく 3 種類の形態で利用が進んでいる。ニュースなどを提供するいわゆる「情報配信」、ソフトウェアのアップデートなどを行う「ソフトウェア配布」、ドキュメントや業務データなどを配布する「ドキュメント配布」の 3 種類である。

3.2 情報配信

情報配信アプリケーションには PointCast 社の PointCast2.8 がある(図 4.参照)。

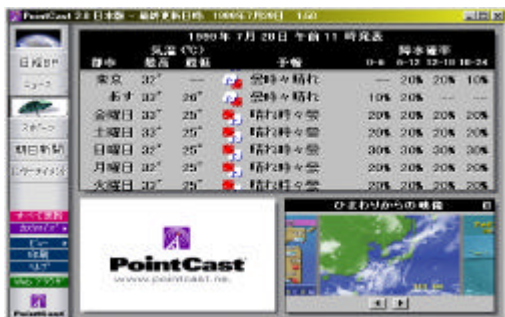


図 4. PointCast2.8

特定の企業情報カテゴリのチャンネルをユーザが選択設定しておけば自動的に新しいニュースがクライアントに届くというサービスである。現在この形態の利用が最も多い。

3.3 ソフトウェア配布

ソフトウェア配布は、プッシュ技術で新規ソフトウェアやアップデートソフトウェアを配布・インストールする使い方である。

3.4 ドキュメント配布

ドキュメント配布は既存アプリケーションのドキュメントやデータを配布して、グループウェアや業務に活用しようという使い方である。例えば、Excel 形式の価格リストをプッシュで顧客に配信し、製品の説明や価格などを合わせた製品カタログを自動的に最新に保つといった使い方である。

4. XML (eXtensible Markup Language)

4.1 XML とは

XML は 1998 年 2 月に W3C より勧告された、インターネット上で扱うデータを記述するための新しいマークアップ言語である。これまでの経緯を表 1 に示した。

表.1 XML 制定の経緯

1996 年 8 月	XML の検討を行う委員会が W3C で組織される。
1996 年 11 月	XML 構文規格の最初のドラフトが公表される。
1997 年 12 月	XML1.0 ¹⁰ 勧告案が公開される。
1998 年 2 月	XML1.0 が勧告される。

XML は SGML¹¹の流れを汲むデータ記述の技術である。そして、HTML と比べてみると「拡張可能」で「意味情報」を持たせることができるデータ形式なのである。XML ではタグを自由に決めることができる。HTML のようにあらかじめ決められたタグを使うのではなく、内容に合ったタグを自分で決めることができる。これにより、データ自身に意味をもたせることができ人間だけでなくコンピュータにも意味を理解させることが可能なのである。

また、XML のもうひとつの特徴としては特定のソフトウェアに依存していなく、テキストベースのデータ形式であるということである。現在、コンピュータのデータ形式はそのソフトウェアによって決まってしまう。例えば、MS-Word を使用して作成した文書は MS-Word でしか読めなく一太郎では読めない。XML では、データに意味を持たせ、オープンなデータ形式をもたせることができる。XML 文書を処理できるプログラムを作成することによって異なったシステムやソフトウェア間

⁸ <http://www.pointcast.co.jp>

⁹米 Marimba 社の Castanet1.1, 米 Fistfloor 社の Fistfloor 等。

¹⁰ <http://www.w3.org/TR/REC-xml>

¹¹ Standard Generalized Markup Language: XML と同じようなタグ定義を行う言語。1986 年に ISO8879 として制定された国際規格。

であっても正しく情報交換をすることが可能になるのである。

4.2 XML の基本構造

XML は全体として XML 宣言と XML インスタンス(タグ付けの行われるコンテンツ部分)に分かれている。以下 XML 宣言、XML インスタンスの順で説明をする。

XML 宣言、は XML のバージョンや使用する文字コードを記述するだけの簡単なものである。

XML インスタンスが、実際のコンテンツをタグ付けする部分である。XML のデータはデータを構成する要素をタグによって表現し、それらを積み重ねることによって、階層化された構造にすることができる。従って、階層型の構造を持つデータ形式であれば、XML で表現することが可能である。

- 要素の書き方

XML 文書は要素を階層的に作り上げることで、成り立っている。ある要素の開始タグと終了タグの対の中に、別の要素の開始タグと終了タグを入れ子にして書くことによって階層関係を作る。よって、XML は HTML とは異なり、要素は必ず開始タグと終了タグで囲まなければならない。HTML で使用されている、、
等のタグは終了タグが省略されているので XML では許されていない。

- 属性の書き方

HTML において、<hr width="300">という書き方がある。この中の width="300"という部分が属性にあたる。XML においても開始タグに同じような書き方が可能で、要素への付加情報を与えることができる。

例: <kiuforum date="2000/01/22">

- 空要素の書き方

要素には、その内容に下位の要素もテキストもない空要素がある。HTML においては 等が挙げられる。XML において、子のようなタグを使用する場合、という記法となる。

4.3 Web ブラウザで表示可能な XML

現在 XML の表示を確認している Web ブラウザは以下の通りである。

- Hybrick ver. 0.81 富士通¹²
- Internet Explorer5.0¹³ Microsoft 社

¹² <http://www.fujitsu.co.jp/hypertext/free/HyBrick/>

Hybrick ver. 0.81 は XLink¹⁴、XPointer¹⁵を表示可能な SGML/XML ブラウザである。しかし、スタイルシートに DSSSL¹⁶しか使用できない。Internet Explorer5.0 では XML のスタイルシートとして XSL を使用している。

XML をブラウザなどで表示する場合には、それぞれの要素に対してどの フォントを使ってどのフォントサイズで表示させるかといった表示スタイルをあらかじめ 決めておく必要がある。このようなスタイル情報を記述するために作られた言語を スタイル指定言語という。

XML に対するスタイル指定言語としては、この時点では XSL、DSSSL、CSS2¹⁷という3つの候補がある。DSSSL は先にも述べたように、IE5.0 に対応していない。CSS2 は XML ファイルに直接記述する方式をとっているが、それでは XML の有効利用を実現するのが難しく、XML を最大限に発揮することが出来ない。今回の実験では XSL の一部にしか使用しなかった。XSL は XML の為に考えられたスタイルシート言語なので、XML の持つ力を最大限に発揮させることが出来る。

5. 配送環境実装の実験

ブッシュテクノロジーを用いた環境を構築し、その動作および運用性を確認するため、小規模のサーバおよびクライアントによる実験システムを構築した(図5参照)。

5.1 装置

サーバ構築に使用した PC は FM/V CPU Pentium120MHz、メモリ 32MB、HDD 1.7GB である。クライアント側の PC は Windows95 マシンである。今回の実験においてクライアント側は Internet Explorer 5.0 が動作すればよく、特に必要な装置やアプリケーションは不要である。

5.2 Web サーバの構築

¹³ 現在は、5.01、5.5 版がダウンロード可能となっている。

¹⁴ <http://www.w3.org/TR/WD-xlink/> XLink, XPointer 共に XML でハイパーリンクを表現する仕様である。古い文献では XLL とも書かれている。

¹⁵ <http://www.w3.org/TR/WD-xptr/>

¹⁶ Document Style Semantics and Specification Language: ISO で標準化されたスタイルシート記述言語の国際標準。

¹⁷ Cascading Style Sheet, Level2

WebサーバのOSにはLinux¹⁸ (Slackware 3.6 カーネル 2.0.35) を用いた。OS インストール後ネットワークカードの認識のためのみにカーネルを再構築した。Webサーバには apache(version 1.3)を用いた。

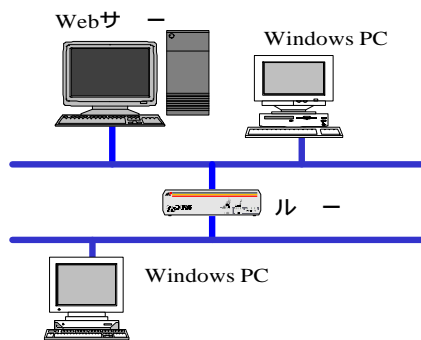


図 5. 実験ネットワーク構成図

5.3 CDF を解釈出来るブラウザ

CDF を解釈出来るブラウザとして、今回は Internet Explorer 5.0 を採用した。PointCast 2.8 は PointCast 社が定めた独自のチャンネル以外は指定できないため、実験で使用することを避けた。

5.4 CDF ファイルの作成

CDF の作成はテキストエディタでも作成できるが、あまりにも生産性が低い。そこで、今回は Microsoft の CDF 編集ツール CDF Generator を使用して CDF ファイルを作成した(図 6 参照)。

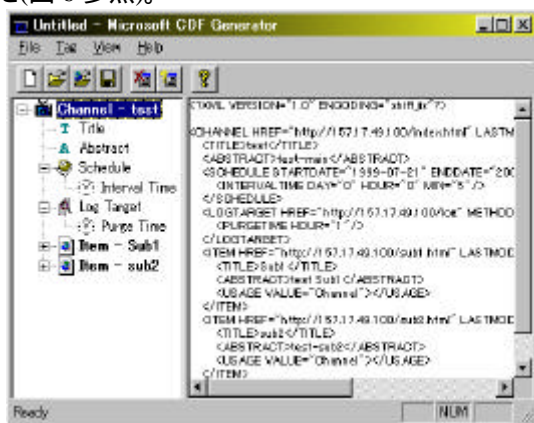


図 6. CDF Generator

CDF Generator では案内に従ってだけで、

¹⁸ 現在非常に注目を集めている Unix 互換 OS であり、Unix に求められる機能をすべて備えている。しかし、今回の実験においては BSDI や FreeBSD でもかまわない。

CDF ファイルを容易にかつ短時間で作成することが出来る。また、CDF ファイル構造を階層表示することができる。

5.5 確認手順

以下の手順で動作確認を行った。

1. UNIX のシェルスクリプト¹⁹によって定期的に更新される簡単な xml ページ(A) を配信ソースとして作成した
2. CDF ファイル(B)を作成し Web サーバへ格納した
3. 別に購読要求ページ(C)を html で作成し、B へリンクした
4. ブラウザから C を参照し、CDF ファイル(B)をダウンロードした
5. IE4.0 の購読要求ウィザードが表示され、チャンネルとして認識される
6. 自動的にページ(A)が CDF(B)の指定に従って随時更新されて表示された

5.6 結果

サーバで 5 分ごとに HTML を書きかえるプログラムを動かし、CDF も 5 分ごとの更新スケジュールにし、チャンネルが 5 分ごとに更新されているのを確認した。

6. XML を用いた

プッシュテクノロジの教育への応用

小中高等学校に PC が配備されてきているが、現状ではワープロをはじめとしたリテラシー教育での利用や Cu-SeeMe を使った遠隔授業、Web による調べ学習に重点が置かれている。専用線で学校が接続された場合のネットワークの利用方法では更に進んだ活用方法が検討されなければならない。また、そのような活用方法がなければ、急速なネットワークの専用線化は望めないであろう。

以下に、データ形式に XML を使用したプッシュテクノロジのネットワーク利用案を示す。

6.1 情報配信

現在、学校行事の案内や父兄への連絡等は、口頭かプリントによって行われているが、連絡が行き届かない可能性も多々考えられる。これを Web 化する試みはすでにいくつかの例²⁰があるが、プル型の転送では父兄が意図的に操作しなければ情報を入手できない。例えば

¹⁹ 参考資料参照

²⁰例えば <http://www.mz.reitaku-u.ac.jp/inf.htm> など

現在普及しつつある CATV インターネット等では常時接続が可能になるので、家庭の Web 端末にプッシュテクノロジーを利用して学校からの情報を的確に転送することも可能になる。

6.2 ソフトウェア配布

学校教育の現場で PC を利用しているとその数や導入されるアプリケーションの規模は大きなものになる。ソフトウェアのインストールやバージョンアップにプッシュテクノロジーを使用すれば管理コストの低減が期待できる。常時接続されているネットワーク環境では、複数の学校へ同時にソフトウェアの配信も可能となり、学校ごとに担当の管理者を育成したり、専門の管理者が巡回して設定するようなコストを削減可能となる。

すでに、プッシュテクノロジーを利用したウイルス対策ソフトもあり²¹、新型ウイルスに対処するためのパターンファイルを自動的にダウンロード、インストールというような利用が可能となっている。

6.3 授業コンテンツ配布

授業コンテンツのデータ形式として XML を使用する。XML は OS、ソフトウェアに依存しないデータ形式であるので、各学校に設置されている様々な PC や、授業に PC を用いることで持ちこまれることが予想されるノート PC にも対応することが可能となる。また、テキストベースなのでファイルサイズも小さい。授業コンテンツは教科書開発センターの様な共同研究機関が開発し、各学校へ配布する。時間割として CDF ファイルを利用する。例えば、配信スケジュールはその授業が始まる 10～45 分前²²に行う。つまり、その授業の前の授業中にダウンロードをすることになる。更新時の時間に幅をとることでホストへの短期間アクセス集中を解消する。

7. まとめ

本稿ではプッシュテクノロジーと XML についてまとめた。また、教育への応用の可能性について述べた。

XML は Microsoft 社や Oracle 社の XML サポート表明に加え、化学式、衛星デジタル放送、電子カルテや、道路情報等への採用も発表されている。プラットフォームを選ばない

データ形式の XML は今後様々な場面に利用されるであろう。また、現状のネットワークではプッシュテクノロジーを十分に発揮させることは出来ないであろう。高速な常時接続型ネットワークの整備が進み、プッシュテクノロジーがさらに発展すれば、学校教育への応用も多様性をもつものと思われる。平成 10 年度の文部省・郵政省の「先進的教育用ネットワークモデル事業」等の学校インターネット高度化計画による、ネットワーク整備に期待したい。

参考文献

- 村田真, 『XML 入門』, 日本経済新聞社, 1998.
 富士通 XML 推進チーム, 『初めての XML』, 日経 BP 社, 1997.
 Sean McGrath, 『XML 開発事例』, 株式会社アスキー, 1999.
 『月間ジャバワールド 1999 年 6 月号』, 株式会社 IDG コミュニケーションズ
 『Nikkei Internet Technology 1997 年 10 月号, 1999 年 5 月号』, 日経 BP 社
 『UNIX USER 1999 年 4 月号』, SOFT BANK
 『DB Magazine 2000 年 2 月号』, 翔泳社, 2000

参考 URL

- KIU(柏 インターネットユニオン)
<http://www.kiu.ad.jp><http://www.kiu.ad.jp/>
 文部省 <http://www.monbu.go.jp>
 W3C (World Wide Web Consortium) <http://www.w3.org/>
 Microsoft <http://www.microsoft.com>
 PROJECT KySS
http://user.shikoku.ne.jp/kyss/xml_xsl.html

²¹例えば米 McAfee 社の「VirusScan」など

²² 小学校の場合。中高等学校の場合は 10～50 分前だろう。