

# XML 技術を用いた Web システムの管理

菊地修一 吉田龍司 塚本享治

東京工科大学メディア学部

## 1. はじめに

インターネットが普及し、キラーアプリとして Web システムが広く利用されている。そのコンテンツの多くは、HTML で記述されている。HTML は記述しやすい反面、レイアウト情報やリンク情報を本文に埋め込むために大規模なコンテンツの管理がしにくいという欠点がある。情報処理学会情報規格調査会[1] (ITSCJ) の Web サイトを取り上げ、その HTML で記述されたコンテンツを XML[2]形式に作り直し、Web システムの管理を XML 技術でおこなったので報告する。

## 2. 既存 Web コンテンツの分析と変換

ITSCJ では、今まで HTML 文書を作成してきたが、NetscapeNavigator6 からレイヤータグをサポートしなくなり、HTML 文書が正しく表示できなくなってしまった。このような技術の変化に柔軟に対応するために HTML コンテンツを XML 化し、この XML コンテンツから HTML ページを始めとする様々なコンテンツを作成することになった。

まず、ITSCJ で作成した約 600 ページの HTML 文書の中から、15 カテゴリ(国際会議報告書、技術委員会議事録、声のページなど)400 ページを対象に、使用する DTD と XSLT[3]の数を減らすことを念頭において分析をおこなった。その結果、文書型、表型、文書型と表型の併用型、の3種類に分類できることがわかった。

しかし、この3種類に対応する DTD と XSLT だけでは、要素名が一般的すぎて、DTD や XSLT が複雑になる。その結果、今後カテゴリを新設する際にシステム全体を理解することが必要になり、かえって保守性が減少すると推測した。そこで、DTD と XSLT を図1のように共通部分と15カテゴリの固有部分に分離することにした。

ここで共通部分とは、管理用のメタ情報(タイトル、著者、カテゴリ名、キーワードなど)であり、固有部分とは文章本体のことである。

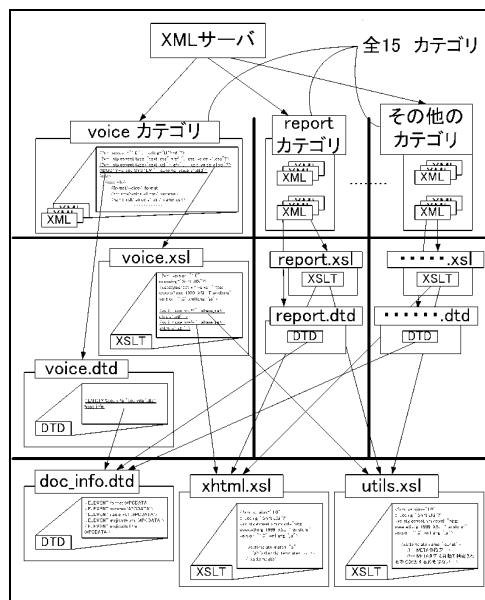


図1. カテゴリにおける DTD と XSLT の階層関係

コンテンツの変換にあたり、HTML 文書のソースから、必要な部分の情報を抜き出して XML 文書を作り、DTD を用いてその妥当性を検証する方法を用いた。

なお、コンテンツ作成を容易にするために、DTD をベースにした XML 構文エディタ (XXE[5]) を日本語化した。

## 3. XML 技術を用いた Web システムの再構築

新しい Web システムは、図2のように HTML 文書をインターネットに公開する既存の Web サーバと、その HTML 文書を XML 文書から生成する XML サーバ(新規に作成)に分けて、構築した。

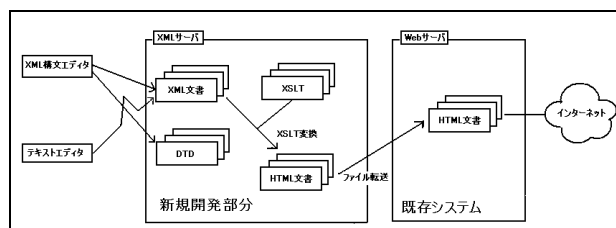


図2. XML サーバと Web サーバの関係

### 3.1 Web サーバと XML サーバの分離

XML 文書から HTML 文書に変換するサーバとして、Cocoon[6]を用いる方法も考えられるが、本

システムでは、XML 文書を管理する XML サーバと HTML 文書を公開する Web サーバとに分離することにした。

その理由を以下に挙げる。

(1) Web サーバの稼働を極力止めることなく容易に移行できるようにするため。

(2) コンテンツの管理を実現するソフトウェアは、複雑になるが、Web サーバに処理性能やセキュリティの面で影響を及ぼさないため。

XML サーバは、ファイアウォール内に配置した。その結果、インターネットから HTML 文書をブラウザする場合は、XML サーバにアクセスするのではなく、XML サーバが作成し Web サーバに置かれた HTML 文書にアクセスすることになる。これによって、XML サーバはインターネットから隔離され、セキュリティが向上する。

### 3.2 XML サーバにおけるコンテンツ管理

XML サーバは、コンテンツ管理用に以下の機能を提供する。

(1) XML 文書から HTML 文書への変換

(2) レイアウトの統一的な変換

(3) リンク情報の管理

XML 文書中から、要素名で検索する XSLT スクリプトを作成した。これによって、リンク切れの検出が容易になった。

(4) インデクスページの自動生成

カテゴリで管理されている XML 文書の中から、特定の情報を XPath[4] を用いて抽出し、インデクスページを自動生成することが可能になった。

これらの機能は、Apache と Tomcat 上で動作する Java サブレットと、XML を処理するために Xerces と Xalan を用いて実現した。

### 4. XML サーバの運用

XML サーバの運用を図 3 に示す。図中の数字の内容は以下のとおりである。

(1) コンテンツ作成者が XML 構文エディタを用いて、XML 文書を作成し、ブラウザを通して XML サーバにアップロードする。

(2) コンテンツ管理者が XML サーバに対して、HTML 文書に変換するようにブラウザを通して命令する。

(3) 命令を受けた XML サーバは、Web サーバに変換後の HTML 文書をファイル転送する。

(4) コンテンツ閲覧者は、インターネットを経て Web サーバ上の HTML 文書を閲覧する。

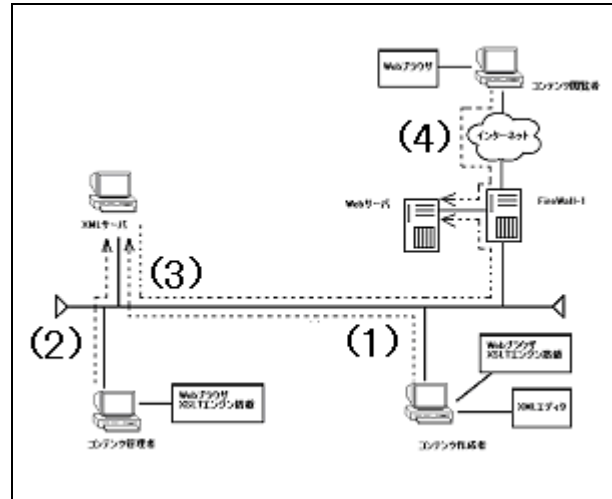


図 3. システムの運用

### 5. 本システムの可能性

コンテンツを XML 化したことによって、今後どのような新規のサービスの提供が可能になるか検討してみた。

(1) コンテンツ検索

意味による検索や、因果関係検索などが可能になる。しかし、高度な検索のためには XML 文書に付加情報を記述する必要があり、コンテンツ作成者の仕事量は増える。

(2) Web パブリッシング

利用者の要求に応じ、XML 文書から集めて PDF に変換して送付することが可能になる。そのためには、PDF 用の XSLT スクリプトが必要になる。

(3) コンテンツ配信

XML サーバが管理する XML 文書、あるいはその一部を Web サービスを用いて他のコンピュータに配信することも可能になる。

### 6. おわりに

HTML ベースの Web システムを再構築して、コンテンツを XML ベースで管理し、それを HTML 文書に変換して公開するシステムを実現した。このシステムは 2003 年 1 月後半から実地に運用されることになっている。

### 参考文献

- [1] 情報処理学会情報規格調査会：  
<http://www.itscj.ipsj.or.jp/>
- [2] XML 1.0: <http://www.w3.org/XML/>
- [3] XSLT 1.0: <http://www.w3.org/TR/xslt>
- [4] XPath 1.0: <http://www.w3.org/TR/xpath>
- [5] XEXE: <http://www.xmlmind.com/xmleditor/>
- [6] Cocoon: <http://xml.apache.org/cocoon/>