

# XSL スタイルシートによる XML の組版と印刷

2001 年 7 月



アンテナハウス株式会社



# はじめに

---

産業界の様々な分野で XML の利用が進むにつれ XML 文書、XML データの表示と印刷の需要が高まっています。XML は情報表現を従来のアプリケーションに依存した閉じた形式から、アプリケーションから独立したオープンな形式に転換できるという大きな特徴をもっています。ところが、XML はテキスト形式であるとはいえ、そのままでは人間にとって必ずしも見やすいとはいえません。XML を見やすい形で表示・印刷することは、それ自身の発展にとっても欠かすことができない技術分野です。

私どもアンテナハウスでは、2000 年 12 月より XSL-FO 仕様にもとづいた日本で初めての XML 組版ソフトである「XSL Formatter」を発売、国内だけでなく海外からも御好評をいただいています。

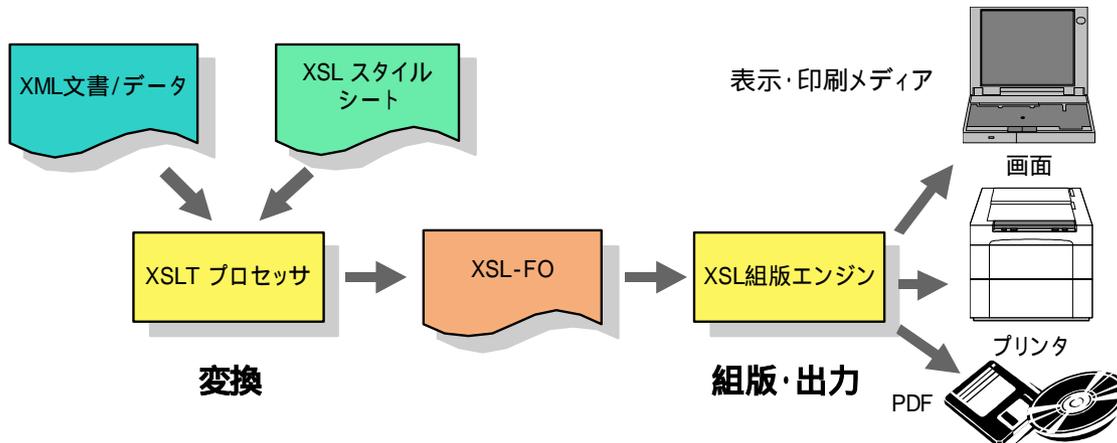
本稿では、XSL スタイルシートと XSL-FO を使用した XML 文書の表示と印刷について簡単に解説し、実例を紹介します。まず始めに XSL Formatter を用いて、XML の表示と印刷をどのように行うかを御覧ください。

1. XSL Formatter を起動します。
2. ドキュメント (D) に XML 文書/XML データを指定します。
3. スタイルシート (S) に適用する XSL スタイルシートを指定します。
4. 組版 (O) - 実行 (E) で組版が実行され、結果が表示されます。
5. 組版結果はプリンタに出力することができます。



# XML の表示/印刷のステップ

XSL スタイルシートを使った XML の表示/印刷は次のステップで行われます。



## XML の表示/印刷のステップ

1. 表示/印刷する XML 文書を用意します。
2. 表示/印刷する仕様に基づき XSL スタイルシートを作成します。
3. XSL プロセッサにより、XML 文書に XSL スタイルシートを適用して XSL-FO を作成します。
4. XSL-FO を組版エンジンに処理させ、結果を表示・印刷メディアに出力します。

各項目について簡単に説明します。

### **XML 文書/データ**

表示/印刷する対象です。DTD は必要ありませんが整形形式の XML である必要があります。

### **XSL スタイルシート**

XSLT 仕様に基づいて作成されたスタイルシートです。XML 文書・データを XSL-FO に変換する XSLT プロセッサに対する指示を記述します。

### **XSLT プロセッサ**

スタイルシートの指示に基づいて、XML 文書/データを Transform (変換) するソフトウェアです。オープンソースや商用の製品も含め多くのインプリメンテーションが存在しています。XSL Formatter では内部から MSXML3 を呼び出しています。

### **XSL-FO**

XSLT プロセッサの出力結果であり、組版エンジンの入力となります。XSL-FO 仕様に基づいた「フォーマッティングオブジェクト」の集まりの XML 文書です。

### **XSL 組版エンジン**

XSL-FO 仕様に基づいた XML 文書を入力とし、組版処理を行って様々なメディアに出力するソフトウェアです。

XML 文書・データは、それぞれのボキャブラリ (語彙) を持っています。例えば、文書であるならば章や節などの要素があるでしょうし、各々の見出しを表すタイトルの属性があるかもしれません。受注データならば、顧客の名前、住所などの要素や、受注した商品のコードの要素もあるでしょう。一般的にこれら入力 XML データ (contents) には、表示/印刷に関する情報はありません。

スタイルシートには、これらの様々なボキャブラリの入力 XML 文書・データを、表示・印刷の表現形式 (style) を表した XSL-FO のボキャブラリに変換するルールが記述されています。



# XSL-FO とは？

XSL-FO は、Web の普及 / 標準化を進める W3C(World Wide Web Consortium)が、XML 文書を綺麗にレイアウトして組版するための仕様として、策定を進めてきたものです。

XSL-FO は「フォーマットオブジェクト」と呼ばれ、56 個のオブジェクトと、246 種類のプロパティから構成されています。オブジェクトにはページ書式を表現するオブジェクトや、文書の段落、インライン、表、リストなどの組版のためのオブジェクトなどがあります。プロパティは、ページサイズやフォントサイズ、ボーダー、パディングなど、オブジェクトの属性を表します。

さて XSL という言葉は若干混乱していますが、W3C の解説では XSL は次の 3 つの部分から構成されています。最後の、Extensible Stylesheet Language の中で定義されているのが XSL-FO です。

分野	最新仕様	内容
XSLT Transformations (XSLT)	Version 1.0 1999/11/16 (REC <sup>(1)</sup> ) <a href="http://www.w3.org/TR/1999/REC-xslt-19991116">http://www.w3.org/TR/1999/REC-xslt-19991116</a>	XML 文書を変換する言語
XML Path Language (XPath)	Version 1.0 1999/11/16 (REC) <a href="http://www.w3.org/TR/1999/REC-xpath-19991116">http://www.w3.org/TR/1999/REC-xpath-19991116</a>	XML 文書の目的部分にアクセスしたり、参照するために XSLT の式で使用される言語。
Extensible Stylesheet Language (XSL)	Version 1.0 2000/11/21 (CR <sup>(2)</sup> ) <a href="http://www.w3.org/TR/2000/CR-xsl-20001121/">http://www.w3.org/TR/2000/CR-xsl-20001121/</a>	組版の仕様を表すための言語

XSLT は本来は XSL を作成するための変換言語でしたが、現在は汎用の XML 変換用途に幅広く利用されています。もっともポピュラーな使用法は、XML から HTML への変換でしょう。

(1) Recommendation

(2) Candidate Recommendation



# 実際の XSL-FO の例

---

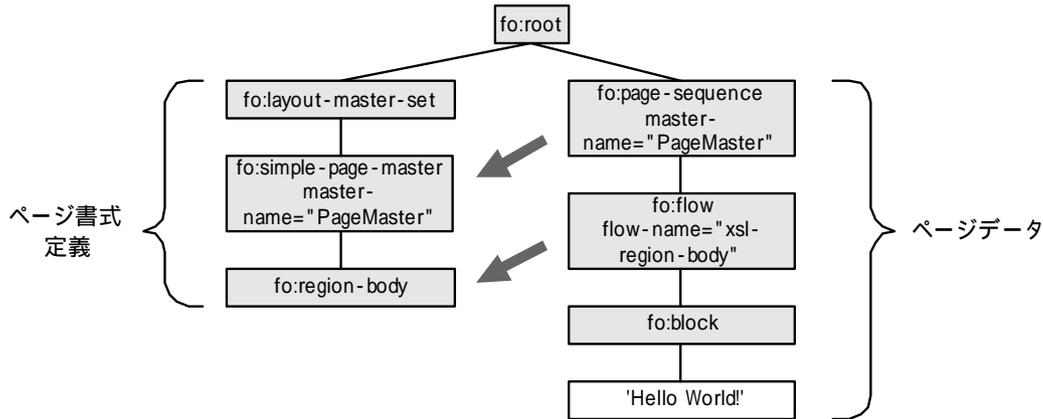
例えば Hello World!を表示するための最低限の XSL-FO は次のようなものです .

## 簡単な XSL-FO の例(Hello.fo)

```
<?xml version="1.0" encoding="UTF-16"?>
<fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">
  <fo:layout-master-set>
    <fo:simple-page-master page-height="297mm" page-width="210mm" margin="5mm 25mm 5mm
25mm" master-name="PageMaster">
      <fo:region-body margin="20mm 0mm 20mm 0mm"/>
    </fo:simple-page-master>
  </fo:layout-master-set>
  <fo:page-sequence master-name="PageMaster">
    <fo:flow flow-name="xsl-region-body">
      <fo:block>Hello World!</fo:block>
    </fo:flow>
  </fo:page-sequence>
</fo:root>
```

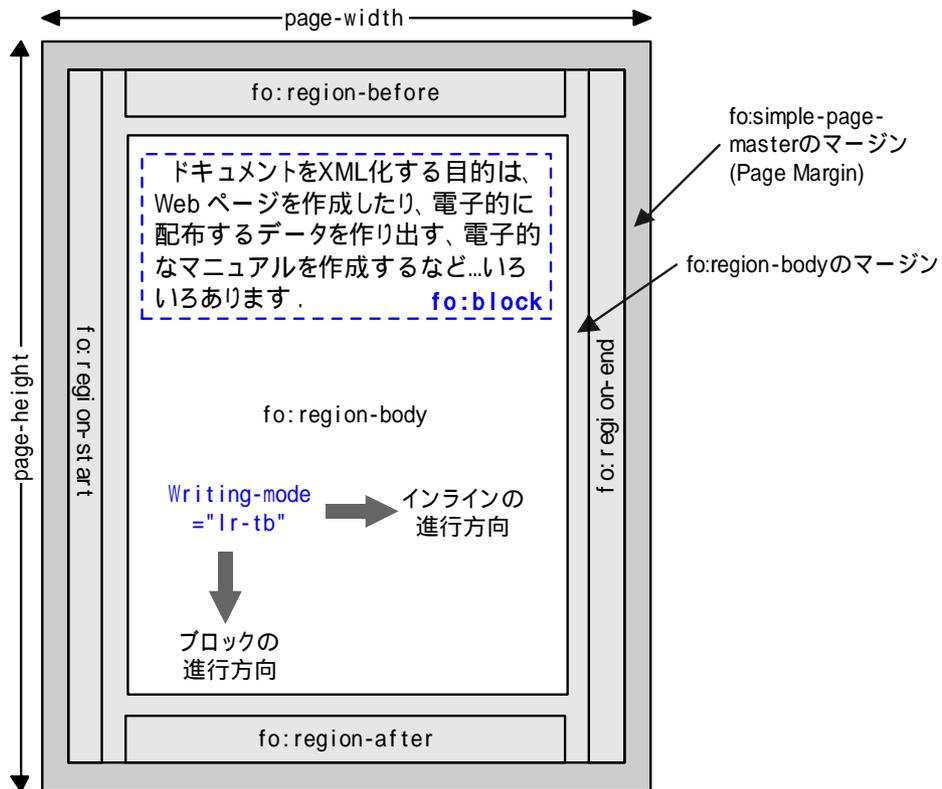
XSL-FO のネームスペース URI として"<http://www.w3.org/1999/XSL/Format>"を指定します . ネームスペースプリフィックスは慣習的に"fo"が使用されています .

hello.fo の構造を簡単に解説します .



### Hello.fo のツリー構造

- ルート要素は fo:root です . 左側にページ書式の fo:layout-master-set、右側に実際の組版文字データを擁する fo:page-sequence があります .
- fo:layout-master-set には、master-name を変えて、複数のページ書式 fo:simple-page-master を定義できます . (例 : 表紙用、目次用のページ書式) ページの中には本文領域の fo:region-body だけでなく、ヘッダー・フッターを配置する fo:region-before や fo:region-after を定義することができます .
- fo:root の下には複数の fo:page-sequence を持つことができます . この中にページを構成する実際のデータを子として配置してゆきます . fo:flow は子の要素の内容をページ書式で定義されたどの領域に流し込むかを flow-name で指定します .
- fo:block は、その子ノードを組版して出力し、終了タグで組版上改行を発生させるオブジェクトです . すなわち段落を表します . これに対して、同じように文字を含むが終了タグで改行を発生させないものが fo:inline です .

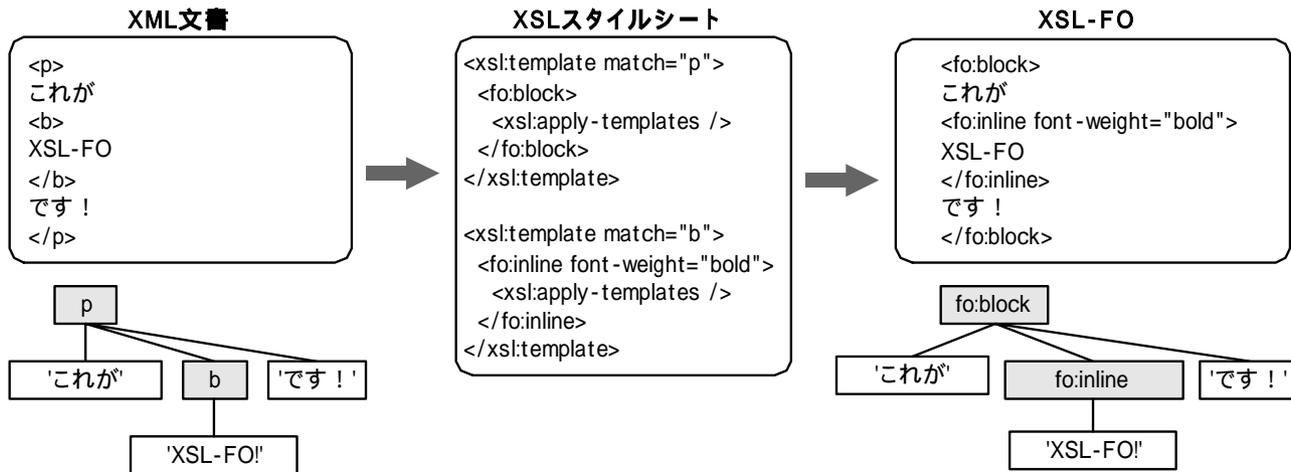


XSL-FO のページレイアウト



# XML 文書を処理するスタイルシート

XML 文書から XSL-FO を生成するスタイルシートの例を示します。

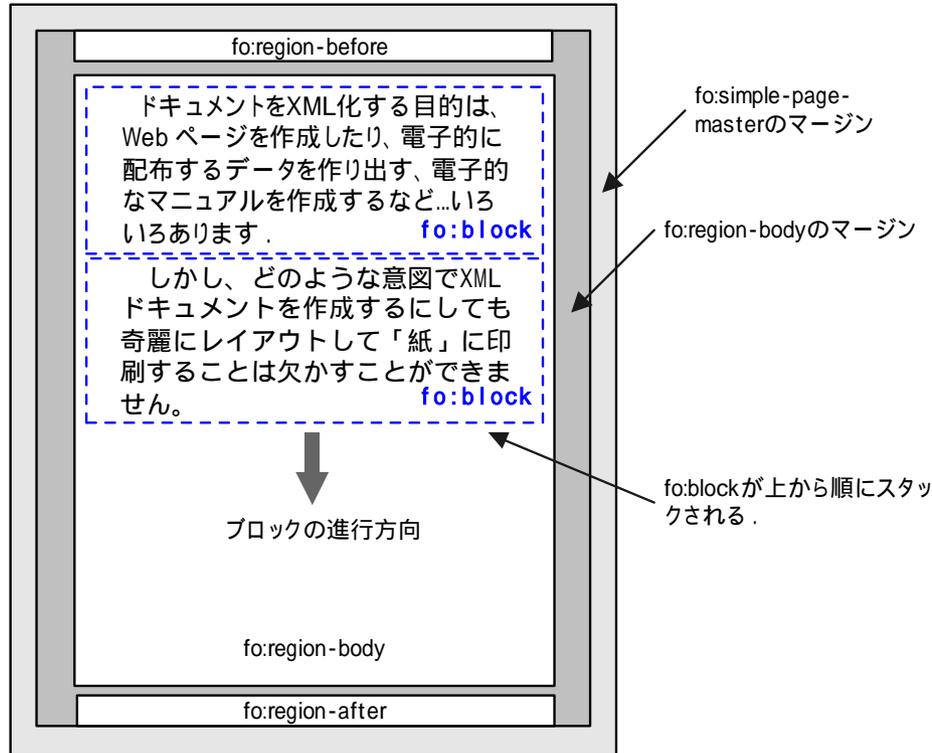


## XML 文書から XSL-FO を生成するスタイルシート例

- ・ スタイルシートは、「XML 文書のボキャブラリを XSL-FO のボキャブラリに変換するテンプレートの集まり」です。
- ・ XML 文書の p 要素は段落を表します。p 要素 fo:block にマッピングします。
- ・ XML 文書の b 要素は強調を表します。b 要素は fo:inline にマッピングし、font-weight="bold"を指定します。

XML 文書を変換するテンプレートは一般的に<xsl:template match="xxx">~</xsl:template>の**データ駆動型**が中心です．組版結果のオブジェクト配置は XML 文書の内容・順序に応じて変化します．

組版処理では本文領域内に上から下に順に fo:block が使用されて文書内容がスタックされてゆきます．1 ページに収まらない場合は、次のページに送られます．

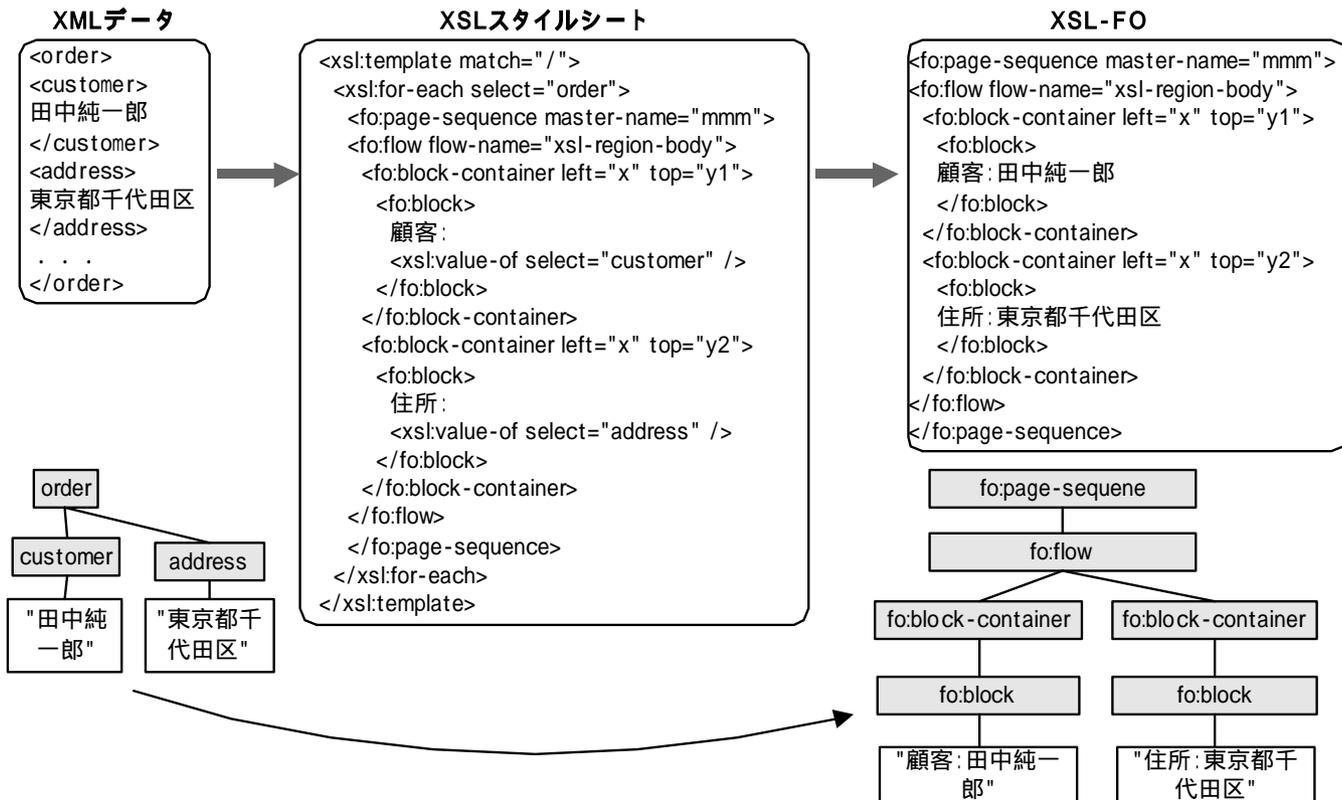


**XML 文書の組版処理**



# XML データを処理するスタイルシート

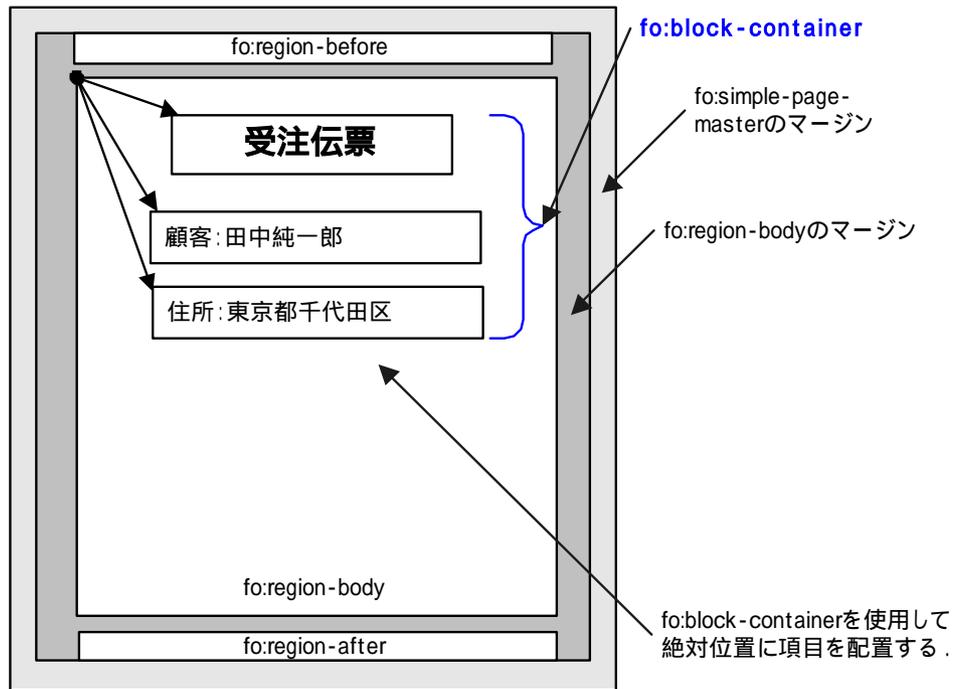
受発注データなど様々な XML データも XSL-FO に変換することができます。



オーダーから 1 ページの帳票を作成するスタイルシート例

XML データを変換するテンプレートは一般的に<xsl:for-each select="xxx">を使用するスタイルシート駆動型です。スタイルシート側から積極的にデータを選択しページ内に配置します。

XML 文書と異なり、ページ内の特定の位置に項目を配置する場合があります。この用途には fo:block-container を使用して位置決めを行います。<sup>(3)</sup>



### XML データの組版処理：絶対位置への項目配置

<sup>(3)</sup> 図ではスペースの関係上指定してありませんが、fo:block-container に absolute-position="absolute" を指定する必要があります。



# XSL-FO 組版エンジン

---

## Antenna House XSL Formatter

- 2000年11月21日のW3C Candidate Recommendation に準拠しています。(4)
- プラットフォームは Windows95/98/Me/NT4.0/2000 です。
- ActiveX コントロールの組版エンジンと、それを制御するユーザーインタフェースプログラムから構成されています。
- 組版エンジン部はコマンドラインや VBScript から COM 経由で使用できるインタフェースを備えています。サーバーでも使用できます。
- 組版結果は Windows の画面かプリンタです。Adobe Acrobat を使用すれば PDF として結果を保存することができます。
- XSLT プロセッサとしてマイクロソフトの MSXML3 を使用しています。他の XSLT プロセッサで作成した XSL-FO を処理させることもできます。

## XSL Formatter の評価

2001年4月に発売された Wrox 社の「Professional XSL」という本で、XSL Formatter が3ページにわたって取り上げられました。組版結果をすぐにその場で確認でき、XSL-FO を使う開発者に新しい可能性を開いたとして高い評価が与えられています。

---

(4) 基本部分はほぼ実装していますが、拡張部分で実装していないオブジェクトやプロパティがあります。

# 世界の組版エンジン

名前	特徴
FOP	Apache Software Foundation が開発するオープンソースの Java 版の XSL-FO 組版エンジン．組版結果は PDF．日本語フォントも使用できるようになってきている．
XEP	米国のベンチャー Render-X の Java 版 XSL-FO 組版エンジン．FOP 同様に PDF を出力する．CJK への対応を強めている．
UFO	Unicorn 社の XSL-FO を TeX のマクロに変換する開発ツール．従って組版は TeX を使用する．
Passive TeX	XSL-FO を処理する TeX のマクロライブラリ．

## XSL Formatter との比較

- ・ 世界のインプリメントは、Java で実装して出力は PDF というものと、XSL-FO Tex への変換という流れに大別できます．PDF 出力は基本的にはサーバープラットフォームを想定していると考えられます．
- ・ XSL Formatter はこのような中では特異な存在です．XSL-FO を簡単に検証・確認できるツールとしては最も使いやすいでしょう．また C++ で実装されているため Java に比べて組版速度が速い点、組版エンジンが ActiveX で実現されており Windows 環境でアプリケーションに簡単に組み込める点も優位性を持っています．



# XSL-FO による組版の実例

ここでは XML 文書の組版と XML データの組版の実例を御覧いただきます。

## XML 文書の組版

私どもアンテナハウスが今年の 6 月から開催している「XSL School」のテキストです。テキストは、XML で作成して XSL Formatter で組版して使用しています。XML 文書は、浅海智晴氏が作成した PureSmartDoc を講習会用に簡略化した文書フォーマットです。

### 注目点

- ・ 目次：目次に入れる要素を XML 文書から抜き出して自動作成します。(fo:page-number-citation, fo:leader)
- ・ 表・箇条書きなどの文書構成要素：対応する XSL-FO のオブジェクトにマッピングして作成できます。(fo:table, fo:list-block)
- ・ 脚注：脚注を簡単に作成できます。(fo:footnote)
- ・ 画像：JPEG, PNG, BMP などの画像を埋め込むことができます。(fo:external-graphic)
- ・ その他スタイルシートのつくりこみ次第で、見栄えの良い印刷物を作成することができます。

## XML データの組版

ブラウザで受注データの入力を受け、サーバー側で組版して結果を PDF に出力する業務アプリケーションの雛型です。ここでは結果だけ紹介します。実際のデモンストレーションはブースの展示で行っています。

## 注目点

WEB サーバーは IIS を使用します . IIS の Active Server Page の VBScript から XSL Formatter の組版機能を使用します .

VBScript から XSL Formatter を使用するコードは非常に簡単です .

## VBScript からの XSL Formatter 組版エンジンの起動例

```
dim objAXF

On Error Resume Next
set objAXF=CreateObject("AXFOSVR.XFOObj")
objAXF.Source="C:\My Documents\XML\data\sample.xml"
objAXF.Stylesheet="C:\My Documents\XML\data\sample.xsl"
objAXF.Option="C:\My Documents\XML\data\option.ini "
objAXF.PrinterName="Acrobat PDFWriter"
objAXF.OutputFile=" C:\My Documents\XML\data\sample.pdf "
objAXF.Execute()
if Err.Number then MsgBox("Format Failed!", mbOKOnly)
set objAXF=nothing
```



# XSL-FO の利用を考える上でのポイント

XSL-FO はまだ新しい技術です．現状ではメリットもありますし、デメリットもあります．ここではお客様から寄せられている質問や意見と、私どもの考え方を述べたいと思います．

## XSL-FO について

### **XSL-FO 仕様はまだ固まっていない．**

昨年2回の Working Draft と Candidate Recommendation が出されました．仕様として安定してきており、実装状況を見ながら次へ進む検証を行う段階です．細部の調整はあるものの、実際に使用することができるでしょう．まもなく次の Proposed Recommendation に進むのではないかと見られます．

### **XSL-FO 仕様は難解である．**

W3C の他の Recommendation は様々な形で邦訳が出ていますが、XSL-FO に関しては存在しません．難解であることは事実ですが、Recommendation そのものでも述べられているように、基本的に組版エンジンのインプリメンターのための仕様です．スタイルシート作成者はすべてを理解する必要はありません．オブジェクトとプロパティの概要を押さえれば、スタイルシートは作成できます．既存のスタイルシートを使って、XSL-FO を作成してみることが理解への近道となります．

### **スタイルシートの作成が難しい．**

スタイルシートの作成は一定のスキルを必要とします．求められる知識は XSLT と XSL-FO の両方に及びます．しかし、XML にスタイルシートを適用して HTML+CSS を作成することは広く行われています．XSL-FO は CSS の仕様を引き継いでいる個所が多く、CSS を理解している技術者ならば比較的簡単に XSL-FO のスタイルシートを理解することはできるでしょう．

### スタイルシートの作成講習会はないか？

私どもアンテナハウスではスタイルシートの重要性を認識し、2001年6月から XSL School というセミナーを開催しています。おかげさまで第1回は定員一杯のお客様に参加いただきました。私たちはソフトウェア製品だけでなく、セミナーの分野からもスタイルシート作成技術向上の支援ができるよう努力してゆきたいと考えています。

### スタイルシートが GUI で作成できない。

これはそのとおりです。XSLT の変換過程をデバッグできるスタイルシートエディタは存在しますが、XSL-FO への変換スタイルシートを GUI で作成できる開発環境はまだ存在しません。XML の世界は非常に進歩が早いので、早晚ツールが登場するでしょう。

## XSL Formatter について

### PDF 出力ができないか？

現在のバージョン (V1.2) では、Adobe PDF Writer や Distiller を使用すれば、PDF に出力することができます。PDF への直接出力は次バージョンで実現する予定です。

さらに次のバージョンでは、XSL-FO 仕様のオブジェクトとプロパティの実装を更に進め、様々な組版の要求に答えられるようにしてゆく予定です。

### XSL Formatter はどのような分野で使用されているか？

日本では企業内の文書管理部門、特に技術マニュアルの組版に利用している例が多いようです。海外では、XML を利用した様々なシステムに部品として組み込まれるケースが多くなっています。

現在の XSL-FO 仕様はまだバージョン 1.0 です。高度な組版はできませんが、文書データではマニュアル関係などは十分利用できる水準と言って良いでしょう。むしろ、今後利用が見込まれるのは、XML をデータとしてやりとりするシステムにおいて、動的に XML データを組版するような機動性を生かした利用形態が増えるのではないかと考えられます。



# 参考資料

---

XSLT, XSL-FO に関する参考文献を紹介します。

## XML バイブル

Elliotte Rusty Harold / 日経 BP

XML 全般の解説書です。XSL-FO について解説している章があります。2001 年 2 月の発売ですが、すでに海外では 2001 年 4 月に第 2 版が出版されています。

## PROFESSIONAL XSL

Kurt Cagle 他 / WROX

XSL 全般の解説書です。内容は XSLT, XSL-FO, CSS, SVG など様々な分野にわたります。XSLT で XML データからグラフの SVG を生成する例など、興味深い内容も収録されています。第 9 章の CSS and XSL-FO には弊社の XSL Formatter が 3 ページにわたって紹介されています。

## XSL Companion

Neil Bradley / Addison-Wesley

XSLT, XSL-FO, HTML などが全般にわたってわかりやすく、平易に解説されています。XSL-FO も第 17 章で多く取り上げられています。邦訳がピアソン・エデュケーションから「XSLT 完全活用マニュアル」として出版されています。

## XSLT Programmer's Reference

Michael Kay / WROX

XSLT の解説書です。非常に内容が豊富です。筆者の Michael Kay 氏は XSLT プロセッサの SAXON の開発者として有名です。2001 年 4 月に第 2 版が出版されました。日本でもインプレスから 10 月にこの邦訳が出る予定とのことです。