

Part 3

オラクルの XML関連機能を 総括する

Feature*1

検索・更新からマルチデバイス対応まで

日本オラクル株式会社
秋本 尚吾
AKIMOTO, Shougo

OracleとXML

XMLはどのようにデータベースを使用し、データベースはどのようにXMLを利用するのでしょうか。オラクルではOracle8iを核として、XMLへの積極的な取り組みを行なっています。XMLというインターネット上での情報交換の標準と、Oracle8iを含めたOracle Internet Platformという情報を管理し、インターネットアプリケーションを配置する製品を組み合わせることで、これまでより効率的で堅牢なシステムを作ることができます。そのために、オラクルは次のような方向性でXMLに取り組んでいます^注。

XMLを利用することで、信頼性が高くかつスケラブルなインターネット・アプリケーションを、開発者が効率よく構築し、コスト効率よく配備できるよう、最高のプラットフォームを提供する

それでは、Oracle8iのこういった場所でどのような方法でXMLを使用し、効率的で堅牢なシステムを作るのか順を追って見ていきます。

Oracle8iでのXMLの利用場所

実行環境の点から区別すると、Oracle8iでXMLを利用する場所として次のふたつが考えられます。

1. Java VM以外で使用する
2. Java VMで使用する

Java VM以外で使用する場合は、CやPerlなどから使用可能なインターフェイスを持った各社のXML Parserを、それまでに構築していたシステムの一部に組み込んで使用する方法や、後述のOracle8i *interMedia Text*を使用するといった方法があります。

Java VMで使用する場合には、Javaコンポーネントを実行する環境として次の3つが考えられます。

- Oracle JVM
- Oracle9iAS
- そのほかのJava VM

Oracle JVM

Oracle8iでは次に挙げるような特徴を持った、サーバーサイドJavaに最適化されたOracle JVMをデータベースの内部に持っています。このJVM上で各種のXML関連ツールを使用することができます(図1)

- データベースデータとXMLを同一サーバーで処理することで、データアクセスのためのネットワークトラフィックを削減
- 各DBセッション専用に変換的なJava VMが割り当てられるため、クライアント間の競合がなくスケーラブル
- CORBA、EJB、Javaサーブレットなどに対応していることで、CORBAクライアント、EJBクライアント、Webクライアントからアクセスできる

Oracle9iAS

Oracle9iASはOracle8iと親和性の高いアプリケーションサーバーです(図2)

Oracle9iAS上の Oracle JVMやOracle Servlet Engine (OSE) でOracle XML Developer's Kit (XDK) などのコンポーネントを使用することができます。

データベースでXMLを利用するとは?

Oracle8iに限らず、データベースでXMLを利用するとはどのようなことでしょうか。データベース上でのXML文書の流れを考えると次のようになります。

- 1 XML Parserなどを使用して、XML文書の内容を理解します。
- 2 データベースにXML文書を何らかの形(XML文書そのまま、列データに変換など)でXML文書の持っている「付加価値情報」と共に格納します。
- 3 データベースに格納しているデータを「付加価値情報」と共にXML文書として配信します。

XML文書のデータと「付加価値情報」を同時に扱う点が重要になります。次にOracle8iでどうやってXMLを利用し、この「付加価値情報」を扱っていくか見ていきます。

Oracle8iでのXMLの利用方法

XML文書をOracle8iに格納することで、XML文書だけでは実現の難しい次のような問題をクリアすることが可能です。

図1 : Oracle JVM

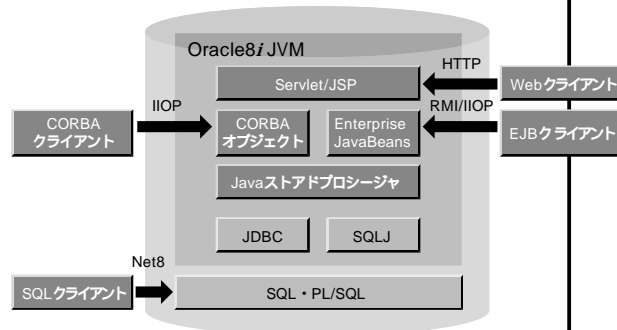
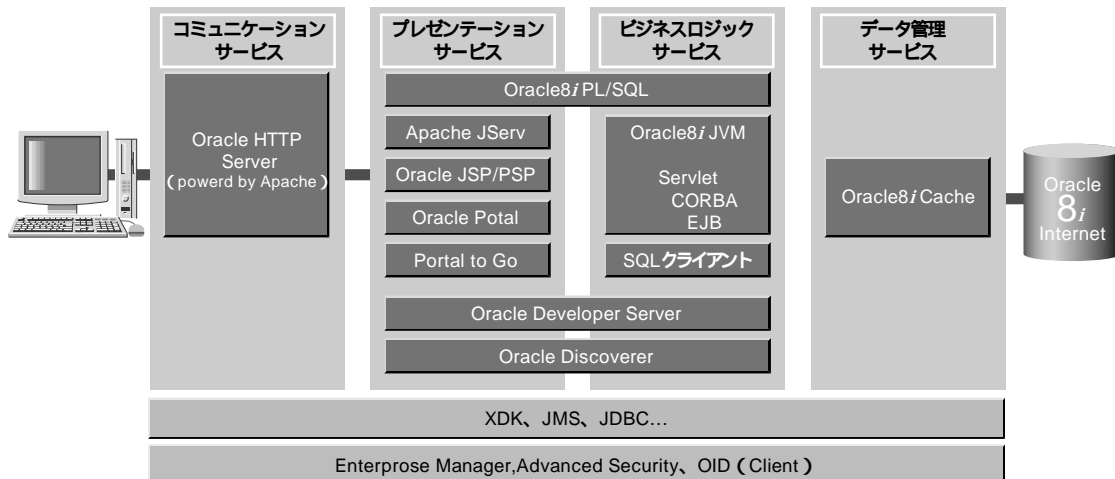


図2 : Oracle9iAS



Feature*1

- Oracle Advanced Securityなどを使用したセキュリティの確保
- 複数ユーザーによる排他制御
- トランザクション処理

また、XML文書をOracle8iに格納するにはふたつの方法があります。

- XML文書をそのまま文書として保存する
Oracle8i *interMedia Text*を使用する
- XML文書をタグ無しでデータとして保存する
Oracle XML Developer's Kitを使用する

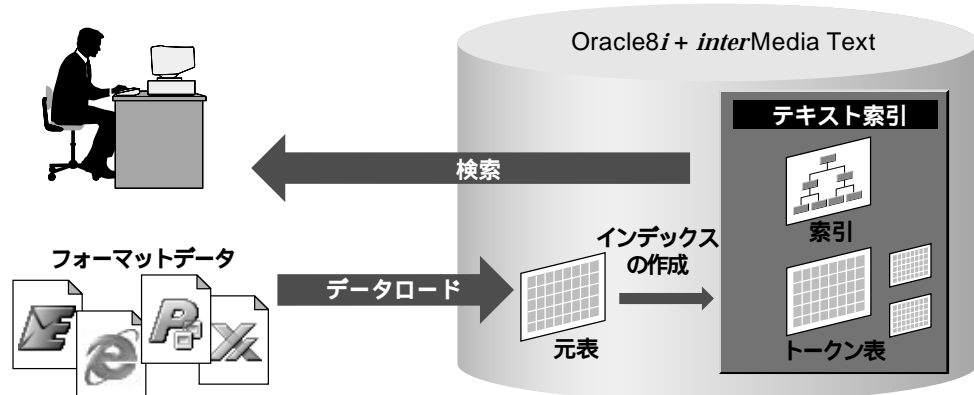
それぞれの方法に関して簡単に紹介します。

Oracle8i *interMedia Text*

XML文書をVARCHAR2/CLOB/BLOB/BFILE/CHARの列にそのまま保存することで、Oracle8i *interMedia Text* (以下*interMedia Text*) を使用してXMLタグ内のデータを検索することが可能です。こういった用途には、法律や雑誌の記事のように、特定のタグだけが更新されるのではなく、文書全体が更新されることが多いものに対して非常に有効です。*interMedia Text*はデータベースアーキテクチャに統合された全文検索エンジンで、XMLタグを使用した検索以外に次のような特徴を持っています(図3)

- SQLやPL/SQLのみで検索可能

図3 : *interMedia Text*



- 多くのフォーマットの文書データへの対応
- Microsoft Office (Word、Excelなど) や一太郎など、文書データ内の文字列を検索対象とすることが可能
- 多くの文書データの格納場所への対応
- 文書データの格納先をデータベース内外にすることができ、OS上の外部ファイルやURLで指定することも可能

検索例

実際に*interMedia Text*を使用した例を見てみましょう。前提として、*interMedia Text*を既にインストールし、使用可能な状態とします。

1) 検索対象テーブル作成

```
CREATE TABLE doc (
  num NUMBER(10),
  text VARCHAR2(100)
);
```

2) 検索対象データ作成

```
INSERT INTO doc VALUES (
  1, '<名前>XML Magazine 4</名前><著者>
  <名前>オラクル</名前><年齢>10</年齢>
  </著者><出版社>翔泳社</出版社>');
INSERT INTO doc VALUES (
  2, '<名前>オラクル</名前><著者>
  <名前>秋本 尚吾</名前><年齢>28</年齢>
  </著者><出版社></出版社>');
commit;
```

3) Oracle8i *interMedia* Textの準備

```

begin
  ctx_ddl.create_preference('my_lexer',
    'JAPANESE_VGRAM_LEXER');
  ctx_ddl.create_section_group('xml_group',
    'XML_SECTION_GROUP');
end;

begin
  ctx_ddl.add_zone_section('xml_group',
    'タイトル', 'タイトル');
  ctx_ddl.add_zone_section('xml_group',
    '著者', '著者');
  ctx_ddl.add_zone_section('xml_group',
    '名前', '名前');
  ctx_ddl.add_zone_section('xml_group',
    '年齢', '年齢');
  ctx_ddl.add_zone_section('xml_group',
    '出版社', '出版社');
end;

CREATE INDEX doc_idx ON doc(text)
  indextype is ctxsys.context
  parameters(
    'lexer my_lexer section group xml_group');

```

4.) 検索

```

SQL> SELECT count(*) FROM doc WHERE
  CONTAINS(text, 'オラクル within 名前') > 0;

COUNT(*)
-----
         2

SQL> SELECT count(*) FROM doc WHERE
  CONTAINS(text, '(オラクル within 名前)
  within 著者') > 0;

COUNT(*)
-----
         1

```

検索結果を見ると分かるように、WHERE句に「CONTAINS(text, '(オラクル within 名前) within 著者) > 0」と指定した場合には、「著者」タグ内の「名前」タグ内に「オラクル」というデータがある場合のみを検索することができます。

Oracle XML Developer's Kit

XML文書とOracle8iのテーブルやビューと結び付けを行なってXML文書の読み込み、操作、変換および表示を行なうためのコンポーネントとして、Oracle XML Developer's Kit (XDK) があります。Oracle XDKには次のようなコンポーネントがあります。

- Oracle XML Parser for Java
 - Oracle XML Parser for PL/SQL
XML文書の構造が文法的に正しいかを解析。またXSLTをサポートし、XML文書からほかの形式のXML文書などに変換することができます。
 - Oracle XML Class Generator for Java
DTDに基づいてクラスを生成し、そのクラスを用いることで、DTDに従った妥当なXML文書を生成するJavaアプリケーションを作成することができます。
 - Oracle XML Transviewer Beans
JavaアプリケーションやアプレットのためのXMLコンポーネントで、JDeveloperなどの開発環境と統合することでXMLを用いたデータベースアプリケーションの作成が容易になる。
 - Oracle XSQL Servlet
WebサーバーのServletとして動作するツールで、<xsql:query>?</xsql:query>というタグの要素の内容としてSQL文を書き、問い合わせ結果をXMLで出力したり、スタイルシートを適応してHTMLなどに変換して出力することができます。
- XMLに対応したOracle Internet PlatformであるOracle8iのOracle JVMに、Oracle XML ParserやOracle XML SQL Utility for Javaが組み込まれ、PL/SQLパッケージとして使用できます。
- また、Oracle8i 8.1.7からはOracle XDKの関連コンポーネントとしてOracle XML SQL Utility for Javaを使用することができます。
- Oracle XML SQL Utility for Java
データベースに問い合わせを渡し、結果からXML文書(テキストまたはDOM)を生成したり、XML



データをデータベース表に書き込むことができる。

ネストされたタグとのマッピング

XML文書として、Oracle8i *interMedia Text*の例で使用した次のようなタグの構成を想定して、実際の検索および書き込みの例を見えます。

```
<書籍>
<名前>書籍名</名前>
<著者>
  <名前>著者名</名前>
  <年齢>著者年齢</年齢>
</著者>
<出版社>出版社名</出版社>
</書籍>
```

Oracle8iのテーブルとXML文書を結び付ける場合に、XML文書のタグがネストしている場合（上記の例では<著者>タグ）にはOracle8iのオブジェクト型やビューを使用します。ここでは、オブジェクト型を使用して、次のようにテーブルを作成します。

```
create type 著者T as object (
  名前 varchar2(20),
  年齢 number(3)
);

create table 書籍 (
  名前 varchar2(20),
  著者 著者T,
  出版社 varchar2(20)
);
```

検索例

1. 検索対象データ作成

```
insert into 書籍 values ('XML Magazine 4',
  著者T('オラクル', 10), '翔泳社');
commit;
```

2. 検索を行なうJavaアプリケーションを作成

```
import oracle.jdbc.driver.*;
import oracle.xml.sql.query.OracleXMLQuery;
import java.lang.*;
import java.sql.*;
```

```
class XMLQueryDB {
  public static void main(String [ ] argv)
  {

    try {
      DriverManager.registerDriver(
        new oracle.jdbc.driver.OracleDriver());
      Connection conn =DriverManager.getConnection(
        "jdbc:oracle:thin:@<SERVER>:<PORT>:<SID>",
        "scott", "tiger");

      OracleXMLQuery qry =
        new OracleXMLQuery(conn,
          "select * from 書籍");

      qry.setEncoding("Shift_JIS");
      // 生成されたXML文書を構造化します
      qry.setRowsetTag("書籍情報");// ROWSETの
          // タグ名
      qry.setRowTag("書籍"); // ROWのタグ名
      qry.setRowIdAttrName(""); // ROWにID属性を
          // 付けない

      // XML文書を文字列形式で取得します
      String str = qry.getXMLString();
      System.out.println(" [ 生成された XML 文書 ] \n"
        +str);

      qry.close();
    } catch(SQLException e) {
      System.out.println(e.toString());
    }
  }
}
```

3. 実行結果

【生成されたXML文書】

```
<?xml version = '1.0' encoding = 'Shift_JIS'?>
<書籍情報> ROWSETのタグ名
<書籍> ROWのタグ名
  <名前>XML4</名前>
  <著者>
    <名前>オラクル</名前>
    <年齢>10</年齢>
  </著者>
  <出版社>オラクル</出版社>
</書籍>
</書籍情報>
```

書き込み例

1. 書き込むXML文書「doc.xml」を作成

```
<?xml version = '1.0' encoding = 'Shift_JIS'?>
<書籍情報>
<書籍>
  <名前>オラクル</名前>
  <著者>
    <名前>秋本 尚吾</名前>
    <年齢>28</年齢>
  </著者>
  <出版社></出版社>
</書籍>
<書籍>
  <名前>XML Magazine 4</名前>
  <著者>
    <名前>秋本 尚吾</名前>
    <年齢>28</年齢>
  </著者>
  <出版社>翔泳社</出版社>
</書籍>
</書籍情報>
```

2. 書き込みを行なうJavaアプリケーションを作成

```
import oracle.jdbc.driver.*;
import oracle.xml.sql.dml.OracleXMLSave;
import java.lang.*;
import java.sql.*;

class XMLWriteDB {
  public static void main(String [ ] argv)
  {
    try {
      // XML文書ファイル名
      String fileName = "doc.xml";
      DriverManager.registerDriver(
        new oracle.jdbc.driver.OracleDriver());
      Connection conn =
        DriverManager.getConnection(
"jdbc:oracle:thin:@<SERVER>:<PORT>:<SID>",
        "scott", "tiger");

      // XMLデータをファイルfileNameから
      // テーブルに書き込む
      OracleXMLSave save =
        new OracleXMLSave(conn, "書籍");
      save.setRowTag("書籍");      ROWのタグ名

      int rowCount =
```

```
save.insertXML(save.getURL(fileName));
```

```
System.out.println(rowCount +
  "件の書き込み成功");
```

```
conn.close();
} catch(SQLException e) {
  System.out.println(e.toString());
}
}
```

3. 実行結果

2件の書き込み成功

Oracle8からのXMLの利用

ここまでは、主にXML文書をデータベースへ「付加価値情報」と共に格納する方法について確認してきました。ここでは、データベースの情報を（付加価値情報と共に）XML文書として配信する場面について確認してみます。

この場面でXMLを有効に効率的に使用する手段として、同一のXML文書を、参照するデバイスによって形を変えて配信する方法があります。これに最も適しているオラクルのコンポーネントとしてOracle XSQL Servletが挙げられます（図4）。

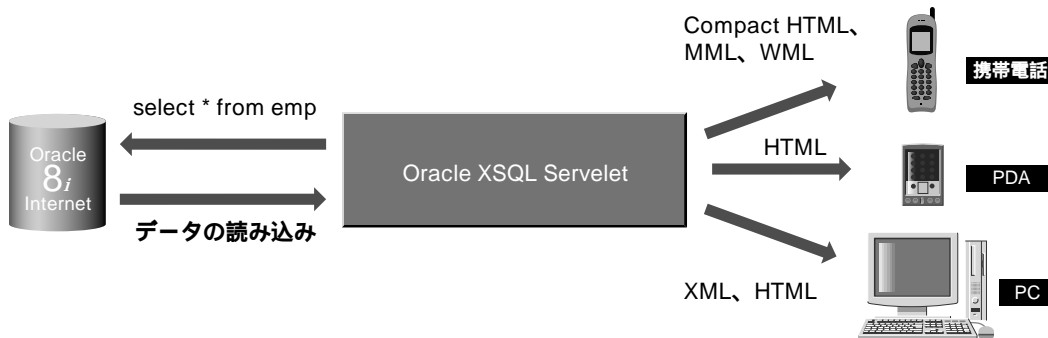
Oracle XSQL Servletの例

実際に、デバイス毎に配信する文書の形式を変更する例を挙げます。この例で使用するファイルは次の通りです。

- dept.xsql
最初に参照されるXSQLファイル。
- dept_html.xml, dept_ckptml.xml
dept.xsqlファイルから呼び出されるXSLファイルで、それぞれHTMLおよびCompact HTMLのフォーマットに変換する。
- emp.xsql
詳細表示時に参照されるXSQLファイル。
- emp_html.xml, emp_ckptml.xml
emp.xsqlファイルから呼び出されるXSLファイルで、それぞれHTMLおよびCompact HTMLのフォ



図4 : Oracle XSQL Servletとデバイス



フォーマットに変換する。

- rowcol.css、rowcol.xml

このふたつはemp_html.xmlが使用するファイルで、`<Oracle XSQL Servletをダウンロードして解凍したディレクトリ>/demo/common`にあるファイルをそのまま使用したもの。

これらのファイルの中で、デバイス毎に別のXSLファイルを適応する方法を見てください。

デバイス毎に区別させる

まず、dept.xsqlです (リスト1)

`<?xml-stylesheet?>` 処理命令のmedia属性でデバイス毎のWebブラウザが使用しているUser Agent毎に適応するXSLファイルを指定しています。

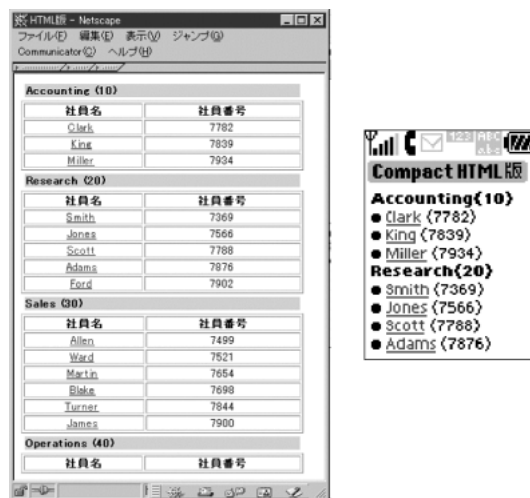
次にCompact HTMLおよびHTML用のXSLファイルを見てください。まず、HTML用のdept_html.xml (リスト2) そしてCompact HTML用のdept_chtml.xml (リスト3) です。

Compact HTMLの場合には、`<table>`タグを使用できないために、代わりに``タグを使用し、表現しています (図5)。

JDeveloper

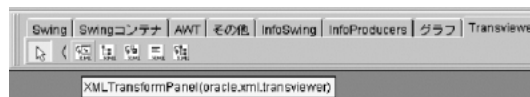
これまでに紹介したオラクルのXMLに対応したコンポーネントに加えて、これらのコンポーネントをさらに効率よく効果的に使用できる開発環境として次のような特徴を持ったOracle JDeveloperを挙げることができます。

図5 : ブラウザでの表示の違い (左がHTMLブラウザでの結果で右がCompact HTMLブラウザの結果)



- Oracle XML Parser for Java、Oracle XML SQL Utility for JavaおよびOracle XSQL Servletが同梱されている
- Oracle XML Transfer BeanなどのJava Beansを簡単にアドインすることができる (図6)
- XML、XSLやXSQL用の構文チェック機能
- Web Serverを起動しなくても、Oracle XSQL Servletをローカルで実行し確認することができる

図6 : Oracle XML Transfer Bean



リスト1 : dept.xsql

```
<?xml version="1.0" encoding="Shift_JIS"?>
<?xml-stylesheet type="text/xsl" media="DoCoMo"
  href="dept_imode.xsl" ?>          i-mode用
<?xml-stylesheet type="text/xsl"
  href="dept_html.xsl" ?>

<xsql:query connection="demo"
xmlns:xsql="urn:oracle-xsql" null-indicator="yes" >
  SELECT DEPTNO, initcap(DNAME) AS 部門名,
  CURSOR(SELECT EMPNO AS 社員番号,
  initcap(ENAME) AS 社員名
  FROM EMP
  WHERE DEPTNO =
  CURR.DEPTNO)
  AS EMP
  FROM DEPT CURR
</xsql:query></xsql:query>
```

おわりに

Oracleは、次のようにインターネット・アプリケーション構築に役立つXMLを利用し、統一されたソリューションを提供します。

- 堅牢なデータベース・プラットフォームである Oracle8i
- Javaサーバー・プラットフォームとしてのOracle JVMおよびOracle9i AS
- インターネット・コンテンツへと対応を広げるOracle8i *interMedia* Text
- 堅牢なXMLサポートを行なうOracle XML Developer's Kit
- 統合ツールとしてのOracle JDeveloper

これらのソリューションによって、一度XMLベースの情報をOracle8iのデータベースに保存すると、その情報はこれまでにみなさんが慣れ親しまれた強力な方法で検索し、分析し、処理することができます。これからも、オラクルでは更なるXMLへの対応をOracleの製品全般に対して行なうよう進めています。

注：XMLへの取り組みに関してはOracle Technology Network Japan (OTN Japan、<http://otn.oracle.co.jp/>)にてテクノロジー情報として公開していません。プロフィール(=ユーザー)登録していただくことで、これらの技術情報、製品のトライアル版やマニュアルなどを参照できるようになっています。

リスト2 : HTML用のスタイルシート

```
<xsl:for-each select="ROWSET/ROW">
  <table width="100%"><tr><td bgcolor="#CCCCCC">
    <b><xsl:value-of select="部門名"/>
      (<xsl:value-of select="DEPTNO"/>)</b>
    </td></tr></table>

  <table border="1" width="100%">
    <tr><th>社員名</th><th>社員番号</th></tr>

    <xsl:for-each select="EMP/EMP_ROW">
      <tr>
        <td align="center">
          <xsl:element name="A">
            <xsl:attribute name="HREF">emp.xsql?p_empno=
              <xsl:value-of select="社員番号"/>
            </xsl:attribute>
            <xsl:value-of select="社員名"/>
          </xsl:element>
        </td>
        <td align="center">
          <xsl:value-of select="社員番号"/></td>
        </tr>
      </xsl:for-each>
    </table>
```

リスト3 : Compact HTML用のスタイルシート

```
<xsl:for-each select="ROWSET/ROW">
  <ul>
    <b><xsl:value-of select="部門名"/>
      (<xsl:value-of select="DEPTNO"/>)</b>
    <br/>
    <xsl:for-each select="EMP/EMP_ROW">
      <li>
        <xsl:element name="A">
          <xsl:attribute name="HREF">
            emp.xsql?p_empno=
              <xsl:value-of select="社員番号"/>
          </xsl:attribute>
          <xsl:value-of select="社員名"/>
        </xsl:element>
        (<xsl:value-of select="社員番号"/>)
      </li>
    </xsl:for-each>
    </ul>
  </xsl:for-each>
</xsl:stylesheet>
```


緊急インタビュー

XML Standards

Graphics アクティビティリード クリス・リリー氏、SVGについて語る

XML Magazine 編集部

本誌Vol.2の記事「XSLとSVGによるXMLプレゼンテーション」ではSVGの解説を掲載した。

今回、そのSVGの仕様策定に携わったW3C Graphics アクティビティリード兼SVGワーキンググループリードのクリス・リリー氏が来日するとの情報をキャッチし急きょ取材に向かった。

クリス氏の経歴

まずはクリス・リリー氏の略歴から紹介しよう。

クリス氏は、英国のマンチェスター大学でコンピュータグラフィックスを勉強し、卒業後はライターとしてCG関連の教則本やイラスト制作に携わる。

後に、IETFに参加するようになり、HTML 2.0^注の策定に関わるようになる。

94年、W3Cに参加しPNGの策定にあたった（PNGはW3Cが結成されてからRecommendationとして策定された初めての仕様）。96年にW3Cチームに正式に参加。Webグラフィックスに関わるようになり、CSSのワーキンググループのチェアに就任。

注：HTMLの現在のバージョンは4.01でありW3Cが策定したもののだが、2.0はIETFのワーキンググループが策定した。

SVGとは何か

SVGの誕生

CSS2の仕様策定にあたる議論のなかで、ワーキンググループ外からの異論があり、なかなか先に進めずにいた。

そんなある時、W3Cの総会でひとりの講演者がブラウザを使ってプレゼンテーションを行なった。そのブラウザは黒い背景の上に白い文字の画像が表示され、大変に見づらいものだった。その画像を見てクリス氏は思った。「グラフィックスもスタイルシートを使って見やすく表現できないか」と。

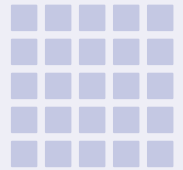
SVGとは

この「グラフィックスを見やすく修正できないか」という発想から生まれたのがSVGである。

SVGとは、Scalable Vector Graphicsの略で、ベクター画像をXMLベースで表現するためのタグセットである。

従来ならば、Webサイトのロゴを作りたい場合、Photoshopなどの画像作成ツールを使用していたが、画像データなので容量が多く、修正も煩わしい。ところが、テキストデータであるSVGであれば、容量が少ないので縦や横への移動、回転などが簡単にできるわけだ。

また、XLink、XPointerを使ってハイパーリンクを埋め込んだり、XML Namespaceを使ってXML文書にSVG文



書を埋め込んだり、XSLTを使ってXML文書からSVG文書に変換するなど、ほかの仕様と組み合わせることで、いろいろな用途が実現する。

実装状況

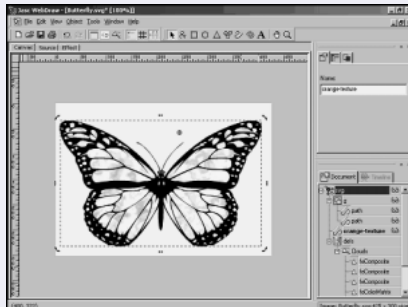
SVGワーキンググループは単に仕様を決めるだけでなく、実装にも積極的に取り組んでいる。Adobe社が提供するIllustratorをはじめ、GNOMEからフリーで提供され、SVGデータをインポート・アウトプットできるGill、また、JASCのWebDrawというエディタツールを使えばグラフィックスを自由にハンドリングすることが可能だ(詳しくは「SVGインプリメンテーションリスト」を参照。http://www.w3.org/Graphics/SVG/SVG-Implementations)

SVGが数多く実装されているのは、各企業、団体がSVGを高く評価しているからこそであるが、「私だけの力ではなく、SVGワーキンググループに優秀なエキスパートがいたからだ」とクリス氏は語っている。

SVGの使用例

今後、SVGデータをPDAやモバイル機器でも表示できるようにしたい、とク

WebDrawを使って表示。ソースからの編集も可能



リス氏自身は考えている。さまざまな使用例が想定されるが、ここではふたつの使用例を紹介する。

SVGを使ったアニメーション

SVGワーキンググループとSMILのワーキンググループ(SYMM)は、SMIL Animationという仕様を共同で策定している。このSMIL Animationは現在、ラストコールワーキングドラフト(ワーキングドラフトの最終チェックの段階)まで、こぎ着けている。

既存のアニメーション技術ではJavaScriptなど、相互運用性の確保が困難な技術が用いられているが、この仕様が完成すれば、Aというオーサリングソフトで作成したデータを、Bというオーサリングソフトでも再利用可能となり、相互運用が確保される。

SVGフォント

既存のフォントは同じユニコードであるにもかかわらず、国それぞれの言語によって見た目が微妙に違っている。SVGフォントというSVGベースのフォントフォーマットを使えば、たとえ見た目が違っていても修正が簡単にできる。これは再三説明したように、SVGが画像データではなくテキストデータであるためだ。



Recommendationはいつか?

SVGは現在、Candidate Recommendationの段階である。

W3Cでは、ある仕様からCandidate RecommendationからProposed Recommendationに前進するためには、ひとつ以上の実装が存在していなければならない。この決まりは前述通り問題なくクリアしている。

「SVGの今後は、SMIL Animationの仕様策定がカギとなるでしょう」とクリス氏は言っている。実は、SVGの仕様はSMIL Animationを参照しており、SMIL Animationの仕様がラストコールワーキングドラフトから先に進まないことにはSVGも先に進むことができないからだ。

SVGが標準化されれば画像をテキストデータとして検索・交換することが可能となり、SVGフォントが台頭すれば、PSフォントのようにデバイスに依存せずに出力することも夢ではない。SVGがRecommendationとなるのはいつだろうか? 今後のSVGワーキンググループの動向から目を離せない。



Webと取材で体験する

W3Cの歩き方

XML Magazine編集部

「W3Cとは？」と聞かれたら、みなさんはどう答えますか。
 「何を今さら」と思った人、本当に大丈夫ですか。
 知っているようで知られていない、W3C。
 今回、W3Cの実態を聞きに、
 運営組織がある慶應義塾大学湘南藤沢キャンパスまで
 行くことになりました。

いざ取材へ

私たちXML Magazine編集部は今回慶應義塾大学湘南藤沢キャンパス内にあるW3C (World Wide Web Consortium) を取材しました。

W3CというとWebに携わっている方ならみなさん1度は聞いたことがあるで

棟の外観



しょうが、組織体系や所在地についてご存じの方はそう多くないでしょう。

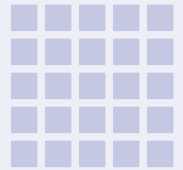
ということで、「W3Cとはどのような組織なのかを取材したいのですが...」とW3Cに依頼したところ、快いお返事をいただき、早速、現地に伺うことになりました。

湘南藤沢キャンパスまでの交通手段を簡単に説明しますと、JR東海道線辻堂駅からバスか、小田急江ノ島線・相模鉄道いずみ野線・横浜市営地下鉄の湘南台駅からバスか、のいずれかです。電車からバスに乗り継いで、湘南藤沢キャンパスに着いた私たちは、まず自然に囲まれたキャンパスの広さに驚き、(タウ)棟と呼ばれる大学院棟内にあるW3Cオフィスまで、地図をたよりに向かいました。

W3Cとは

W3CとはWebの発展と相互運用性の確保を目的に共通の仕様を策定している会員制の国際産業コンソーシアムです。「メンバー」と呼ばれる企業や大学などの組織が出資し、「ホスト」という3つの組織が中心となって共同運営されています。3つのホストは、今回お伺いした慶應義塾大学湘南藤沢キャンパスのほか、米国のMIT/LCS (マサチューセッツ工科大学計算機科学研究所)、フランスのINRIA (国立情報処理自動化研究所) にあり、「チーム」と呼ばれるスタッフによって運営されています。

チームは60人程度の規模で構成されており、仕様の策定に携わる技術スタッフ



だけでなく、広報担当者やシステム管理担当者などもいます。

W3Cは基本的にメンバー組織を中心に議論を行いますが、チームからメンバーに意見や提案を求め、技術に関する全体像や将来性について議論をすることもあります。

ワーキンググループの立ち上げから Recommendationに至るまで

チーム内で議論の結果、仕様の開発が決定され、メンバーの承認を得るといよいよ開発段階に入ります。そのとき実際の開発にあたるのは「ワーキンググループ」と呼ばれるグループです。

ワーキンググループでは、作業を取りまとめるチェア（議長）を決定した後、メンバーからワーキンググループへの参加者を募ります。ひとつのワーキンググループに大抵20程度のメンバー組織が参加しますが、制限はありません。

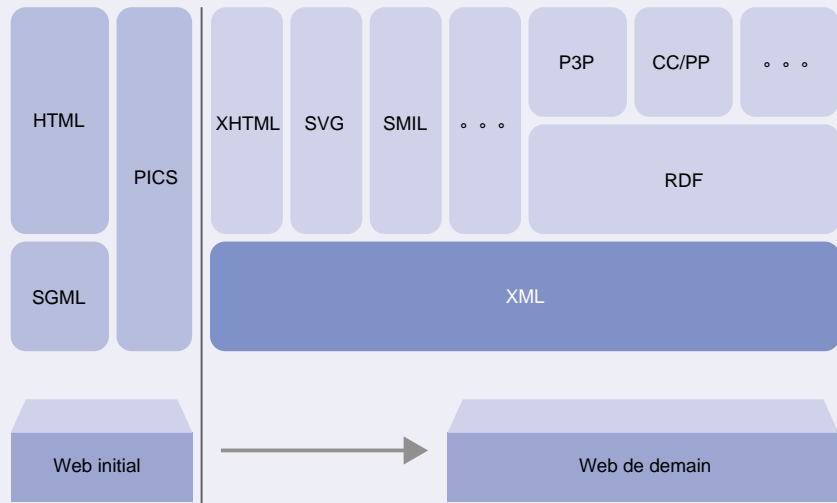
ワーキンググループが行なう開発プロセスについて、W3Cでは4つの段階に分けて一般公開しています。

Working Draft（ドラフト作業）

仕様策定にあたって、必要なことや結果、見込みなどを考慮しながら議論します。ワーキンググループに参加していないメンバー組織やほかのワーキンググループからも意見を集め、問題点を発見し実装に向けて話し合います。

Candidate Recommendation(勧告候補) Working Draftで議論した仕様が要求を満たしているのかテストします。この時に、いくつかの実装を行ないます。

W3CではXMLをベースに各仕様が策定されている



Proposed Recommendation（勧告案） Candidate Recommendationで決められた仕様をメンバー組織に確認してもらいます。

Recommendation（勧告）メンバー組織から「正式な仕様にしましょう」というGOサインが出れば、晴れて正式な仕様として公開されます。

一度Recommendationになった仕様は変更されませんが、間違いの修正一覧を公開することもありますし、機能を追加して、新しい仕様として策定し直すこともあります。また、Recommendationになる前の段階では、ひとつ前の段階に戻ることもあります。実際に、XPointerの仕様策定では、Candidate RecommendationからWorking Draftに戻りました。W3Cでは早く策定することよりも、良いものを作りたいという考えがあるからです。

W3Cが直接関わる作業ではありませんが、Recommendationにまつわるものとして、仕様書の翻訳作業があります。

W3CがWebに掲載する仕様書は全て英語で書かれていますが、策定する仕様書はたくさんあるので、全てを翻訳するとなると、どうしてもボランティアに頼らざるを得ません。このため、正式な仕様書は英語版で、英語以外の言語で書かれた仕様書は参考訳とされています。

W3Cの基本方針

XML Magazineに寄せられる読者がきの質問に、「実現したいシステムを構築するために数多くある仕様のうち、どれを組み合わせるのが、よく分かりません」というものがあります。

W3Cの方針は、あくまでWebに関する基盤技術を作ることで、応用技術は関与していません。「必ずしも技術を順番通りに使うという制限をしない」と考えているからです。

また、システム環境によっていろいろなやり方がありますし、Web技術は日々進化しています。こういった事情からもこれでやればよいとは言えません。従って、どうビジネスに結び付けるか、どの

ように応用していくかは自分で考えなければなりません。重要なのは、仕様がどういったもので、どのように使えるかを知ることです。そして、いつでもシステムを変更できるように複数の仕様を導入し、システム全体を見据えた柔軟な設計を考えることです。

メンバー組織、他団体との関わり

“Submission”とはメンバー組織からWeb技術をW3Cに提案することを言いますが、必ずしも提案が認可されるとは限りません。ほかのメンバー組織からの反応が悪く、受け入れられないケースもあります。

最近の例では、昨年12月に“SOAP Messages with Attachments”、11月には“XTND (XML Transition Network Definition)”、“SpeechObjects Specification V1.0”、“Shop Log File Format”が、SubmissionとしてW3Cに受理されました。

W3Cと他団体との関わりについては、IETF (The Internet Engineering Task Force) とHTTPプロトコルを共同で開発したり、OASIS (The Organization for the Advancement of Structured Information Standards) とはXML test suiteで付き合いがあります。また、PNG^[注1]をISO (International Organization for Standardization)^[注2]に提出しました。ほかにもテレビ業界や携帯電話関連でも多くの団体と関わっています。

このように、多くのメンバー組織や他団体と交流があるわけですが、日本の組織に限って言うと、外国に比べてメンバー数が少ないのが現状です。欧米の金融、印刷・出版業界の組織が参加していても

日本ではまだ例がありません。

もちろん、日本の全ての組織が貢献していないわけではありません。アクセスや松下電器産業はCompact HTMLをSubmissionとして提案し、初めてのコンパクトなHTMLとして仕様策定に貢献したという例もあります。

「もっと日本の組織が参加していただけるといいなあと思います」

とHTML アクティビティリードの石川雅康氏は言います。もう少し石川氏の意見に耳を傾けてみると、

「日本の組織は積極的に議論に参加してこないことの方が多いです。いろんな意見を言うてはくるんですけど、公開の場で自分の意見を述べるディスカッションに参加してこないのです」

ここで言う公開の場とは、W3C内で行なわれるメーリングリストを指します。W3Cでは、公開の場で議論した記録を基にして仕様を策定していきますので、いくら公開の場以外で意見を述べても反映されません。

日本の組織がW3Cへ積極的に参加しないのは組織内の事情、外国との時差の違いなどありますが、このような話を聞くと日本人の特質が問題となっている側面もあるようです。Device Independence アクティビティリードの北川和

裕氏は、日本人の特質についてさらにこんなことをおっしゃっていました。

「W3Cは、ワークショップの結果やsubmissionの内容を基にメンバー主導で策定作業を進めます。もしかしたら、こういう文化はもともと日本にはあまりなかったのかもしれないね」

注1: Portable Network Graphics。画像フォーマットのひとつでW3Cが仕様を定めている。

注2: 国際標準化機構。国際標準を取り決めるための国際機関。

W3Cの気になる仕様

W3Cで策定している仕様は数えると切りがありませんが、今回みなさん気になっているXML Schemaと最近脚光を浴びているRDFのふたつの仕様について聞いてみました。

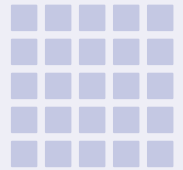
XML Schema

XML Schemaが標準化されればデータのやり取りがスムーズになるだけでなく、DTDとは違ってXML Schema自体の仕様はXMLベースなので、バージョンアップしても旧バージョンから新バージョンに対応するように、XSLTで変更することも可能になります。

Internationalizationアクティビティリードのマーティン・J・テュールス



Device Independence
アクティビティリードの
北川和裕氏



Internationalization アクティビティ
リードのマーティン・J・テュールスト氏

ト氏は、
「XML Schemaの仕様策定作業はなかなか大変な仕事で、予定より遅れています
が、少しずつ進んでいます。今週末ロ
ンドンでミーティングがあります（1月
18日現在）」
とおっしゃっていました。

1回目の提案でたくさんの人から多くの
要求があり、マーティンさん自身も現
段階のXML Schemaにはまだ足りない
ところがあると言います。

XML Schemaの策定は少しずつ進ん
ではいるものの、完成にはまだまだ時間
がかかるといったところです。

RDF
(Resource Description Framework)

RDFとはメタデータ記述言語です。
メタデータとはデータのデータのこと、
つまり、そのデータが何であるかを詳細
に記述するものです。

RDFは一時期忘れられていましたが、
ここに来て再び脚光を浴びています。こ
の現象についてW3C側ではどのように
見ているのでしょうか。

「XMLほどではないですが、RDFは少
しずつ浸透しています」

とマーティンさんは言います。それでは
なぜ、XMLほど浸透しないのでしょ

うか。その問いにマーティンさんはこう答
えています。

「RDFは頭で仕様を考えなければいけな
いから」

つまり、RDFはメタデータを記述す
るものなので、ほかのデータ自体を記述
する言語に比べると、ちょっととつき
にくいかもしれないということです。ま
た、

「いろいろな使い方があるから何に使っ
たらいいかわからないという質問もあり
ます」

と北川氏は言います。いろいろな応用の
仕方があるがゆえに、ユーザー側からす
ればどのように使ったらいいかわかってし
まうということです。

しかしながら、W3Cで仕様策定中の
端末能力記述方式であるCC/PPなどに
も用いられており、その応用範囲は広が
ってきています。

RDFはまだ徐々に浸透している段階
にあるわけですから、今後思いも寄らな
い使い方が生まれるかもしれません。

W3Cが望んでいること

今後、ビジネスはパートナーといっし
よに行なうことがさらに増えてくるでし
ょう。その時にビジネス戦略を考えなけ

ればなりませんが、その前にデータ交換
をスムーズに行なうためのインフラを整
備しなければ、情報を共有することはで
きません。また、業界内、さらに業界同
士のデータ交換も行なわれつつありま
す。

各企業では、使っているツールもOS
もさまざまで、データ交換の際には記述
や定義、システムの相性などが障害とな
っていました。

W3Cでは、XML Schemaのように、
まだまだ決まっていない仕様が多いわけ
ですが、お互いに利用できるようなWeb
技術の仕様を策定しています。

「私たちは共通の基盤を作り、使う側は
形式的にならないように応用してくれた
らいいと思います」

とマーティンさんは言いますが、それは
マーティンさん自身の望みでもあり、
W3Cの望みでもあるのです。

取材を終えて

取材前、正直言うと、開発室で黙々と
コンピュータに向かってキーボードを叩
いている研究員の方々の姿を想像してい
ましたが、実際はイメージとは全く違い
アットホームな雰囲気の中インタビュー
を行なうことができました。

どうしたら面白いもの、便利なものが
作れるのだろうか、と常に考えているの
だと、お話をうかがったチームの方々の
見て思いました。

W3Cのチームの方々、お忙しいところ
お時間をいただき、本当にありがとう
ございました。

W3Cの歩き方

座談会

Webの将来について語る

慶應義塾大学湘南藤沢キャンパスにあるW3Cのチームの方々に、「Webはこれからどう変わっていくのか」という大きなテーマについて、座談会という形で語っていただきました。

座談会メンバー

Internationalization アクティビティリード /
マーティン・J・テュールスト氏

Device Independence アクティビティリード / 北川 和裕氏

HTML アクティビティリード / 石川 雅康氏

システム管理担当 / 遠山 緑生氏 平川 泰之氏

Web全体がどう変わっていくのかそれぞれどんな見解を持っていらっしゃいますか？

北川：個人的にはデスクトップPCやノートPCがどんどん減って欲しいと思っています。みんながコンピュータを持つのも変かという感じがします。いろいろなところにWebブラウザがあっても別にかまわないと思うのです。

それはなぜですか？

北川：それはXMLが基盤になって、例えばリモコンでXMLをコントロールしてもいいわけですよ。

マーティン：XMLをリモコンでというのはちょっとねえ。

北川：良くないかもしれないけど、作る側は楽ですよ。手でテストができますから。XMLで全部記述すればブラウザひとつである程度までテストできますし、別にHTTPを使う必要は全然なくて赤外線でも何でもいい。とにかくデータが行けばいいんですよ。携帯ではあれだけ若者が速く打てるんだったらキーボードも打たなくてもいいんじゃないかと。

マーティン：でも、いろんなところでWebが使えたらいいですねえ。

北川：レシートもXMLデータにしてポ

タンを押せば家計簿がつくれちゃう、というのいいかもしれません。

マーティン：出張に行ったとき領収書を会社に回すように電子化されるといいでしょうねえ。でも、技術的になかなかできないんですよ。

石川：どうなって欲しいということ言えば、みんなが規格を意識しなくてすむようになればいいと思います。「なんでこのブラウザは文字化けするんですか」といった質問がなくなればいいですね。

マーティン：少しずつ共通化が進んではいます。でも、急いで開発しても使えないものができてしまっはいけないし、難しいところだと思います。

遠山さんはWebのこれからについてどう思いますか？

遠山：Webというのは、最近ビジネス系の話が多いじゃないですか。昔はパーソナルコミュニケーションツールとしての意味合いが強くて面白かったと思うんです。どうしてもマーケットベースになると、どうやって儲けるかという話を中心になるじゃないですか。もうちょっと初心に戻って欲しいですね。最近のPeer to PeerもXMLやRDFといったメタデータを使って、うまく取りまとめる

ような形で新しいネットワークが形成されるといいんでしょうね。

平川さんは？

平川：今までですと、電話なら通信、テレビなら放送というように、それぞれ分野別に独自の方法でやってきたじゃないですか。今はちょうどそれらを同じ規格に統合するいいチャンスだと思うんですよ。それぞれが持っている能力や利点をひとつにまとめて、あるものから全然違うものを意識せずに使えるように目指せばいいと思うんです。

それでは最後に、Webの開発者に対して要望がありましたら。

マーティン：そうですね、いっぱいありますからね。

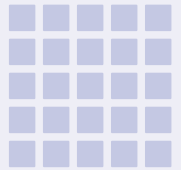
遠山：やはり、正しいHTMLを書いてくださいということです。

マーティン：標準に戻りましょうということですかね。

北川：いきなりアクセシビリティを考える前に正しいHTMLを覚えて備えると。

遠山：「イバク」のトップページとか。

マーティン：いろんな仕様適合度に従って点数を付けるソフトウェアもありますね（詳細はXML Magazine 02「XHTMLの



概要」を参照)。

石川：あの点数の付け方はいろいろと思うところもあるのですが...

マーティン：W3Cのトップページも90点くらいしか付いていないんだけど、実際にチェックして点数を付けると、そのページの問題が発見できたりするんじゃないですか。

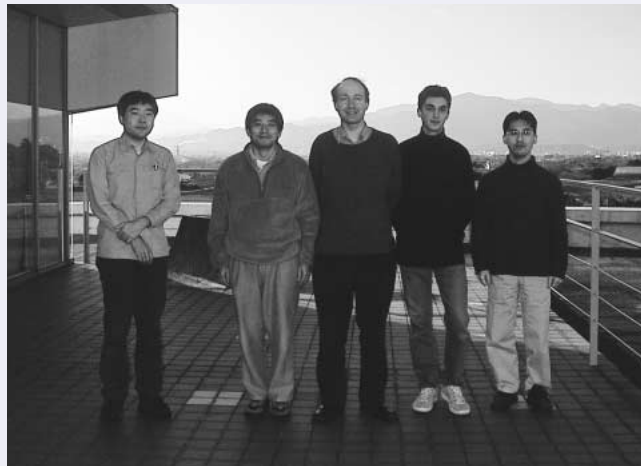
石川：あれは減点法ですから、別にマイナス50点でもマイナス100点でも、大して差はないでしょう。問題があるという意味では。

北川：50歩100歩ってやつ。

石川：しょせん、ダメなものはダメで。

北川：せめてバリデータはかけて欲しいですよ（バリデータについては、<http://validator.w3.org/>を参照)

石川：そんなことを意識しないといけない時点でおかしい。私はできれば正しいHTMLを書かなきゃいけないと、みんなが考えなくていいようになって欲しいですね。



慶應義塾大学湘南藤沢キャンパスのW3Cチームのみなさん。

遠山：オーサリングツールを上手に使って、正しいHTMLを出力できるといいですね。

そのようなツールを出すご予定はありますか？

マーティン：予定は以前からあるんです。Amayaの国際化の進行状況は必ず

しも順調とは言えません。あんまり期待しないでください。

北川：いやいや、みんなに期待してもらってプレスリリースしないと。

マーティン：いやー、それはやりますよ。

本日はお忙しいところ、お集まりいただきありがとうございました。

W3C「PAGE2001」に参加

印刷やDTPなどのグラフィックスビジネスに焦点を合わせた「PAGE2001」(社団法人日本印刷技術協会主催)が、2月7日から9日までの3日間にわたり開催された。

池袋サンシャインシティ コンベンションセンターTOKYOにて行われた今回は、今年で14回目を迎え、全37セッション、出展社数124社に上る大規模なカンファレンスとなった。

W3Cもこのカンファレンスに参加し、「Webパブリッシング技術 W3Cの取り組み」というテーマで、各講師による講演を行なった。

各講演内容は以下の通り。

「W3Cの技術仕様における概要」

慶應義塾大学湘南藤沢キャンパス W3Cチーム
遠山緑生氏

「ベクターグラフィックス SVG」

Graphics アクティビティリード
クリス・リリー氏

「XML変換マークアップ XSLT」

インフォテリア株式会社社長
平野洋一郎氏

「組版用マークアップ XSL Formatting Object」

アンテナハウス株式会社
古畑広氏

社団法人日本印刷技術協会のWebサイト
(<http://www.jagat.or.jp/>)



本誌03号の93ページにSOAPのTRに関する記載について誤りがあります。

[誤] <http://www.w3.org/tr/soap>

[正] <http://www.w3.org/TR/SOAP>

また138ページに「W3C.org(海外)」と記載していますが、正式名称は「W3C」であり、また、今回ご紹介した通り、アメリカやヨーロッパなど海外だけでなく、日本にも拠点を持つ国際的な標準化団体です。読者ならびに関係者の方々には大変なご迷惑をおかけいたしました。

ここで、お詫びし訂正させていただきます。

W3Cの歩き方

サイト編

XMLを学ぶためには、W3Cの情報収集が欠かせません。いざW3CのWebサイトを覗いてみると、膨大な情報量にかなり戸惑うことでしょう。途中で迷子にならないように、W3Cサイトをご案内します。

1 W3Cを知らない人は

2 最新情報を知りたい人は

3 W3C公開の技術を
すぐを知りたい人は

4 ソースコードや連絡先
などを入手したい人は

1 W3Cを知らない人は
W3Cサイトに全く足を踏み入れたことのない人は手始めにここからじっくり見るとよいでしょう。情報が多いため、右の5つのカテゴリーに分けて掲載しています。

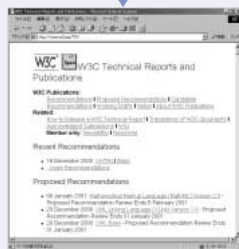


Activities | Technical Reports | Site Index | About W3C | Contact



Activities

W3Cの活動に関するページです。Webに関する技術やツールごとに、それぞれ活動状況を報告しています。



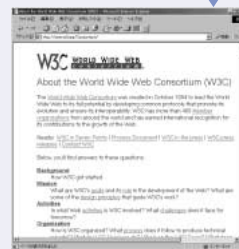
Technical Reports

Notes, Working Drafts, Candidate Recommendations, Proposed Recommendations, Recommendationsの5つのカテゴリに分けて各技術仕様書を掲載しています。



Site Index

W3Cのキーワードを集めたインデックス。簡易検索でキーワードを入力すると、W3Cサイト内の情報をリストアップしてくれます。



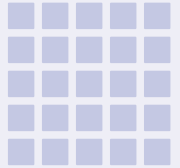
About W3C

設立の背景や組織体系など、W3Cの組織に関する情報が掲載されています。W3Cがどういった組織が分からない人はこのページが便利です。



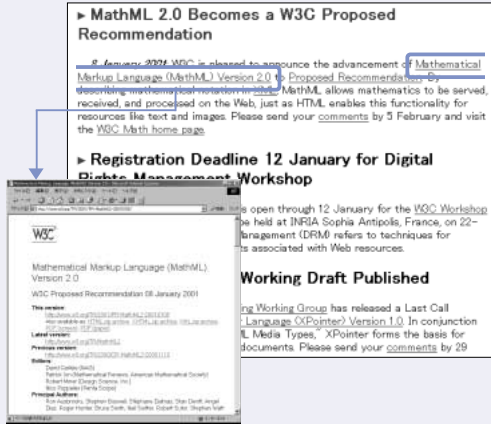
Contact

ここにはW3C各部門の連絡先が掲載されています。W3Cに質問や意見がある場合、ここからメールで問い合わせができます。



② 最新情報を知りたい人は

各技術仕様の公開情報や活動状況が次々と更新されています。ちょっと見ないうちに「Recommendation」の情報が掲載されていることが多々あるので、定期的にチェックしましょう。



最新情報の詳細について読むことができますので、足を踏み入れてみましょう。

③ W3C公開の技術をすぐに知りたい人は

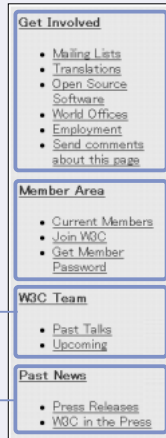
「W3C関連のこの技術を知りたい」と決まっているのなら、ここを訪ねてみましょう。この「W3C A to Z」のリンク先は先に説明した「Site Index」と同じですが、初めから知りたいことが分かっているれば、トップページから目的のページに飛ぶことができます。



あらかじめ知りたいことが分かっているなら、その詳細をすぐに調べることができます。

④ 各種情報を入手したい人は

ツールのソースコードはここからダウンロードできます。また、翻訳仕様書、過去の記事、各連絡先などW3Cに関する細かな情報を見ることもできます。ここでは4つのカテゴリーに分類されています。



Get Involved
ここからツールのソースコードをダウンロードできます。メーリングリスト、翻訳仕様書、W3Cへの問い合わせのメール、といった情報も見ることができます。



Member Area
W3Cメンバーに関するページです。各メンバーの紹介、メンバーへの参加方法が掲載されています。



W3C Team
W3Cチームの紹介ページです。91年からのチームスタッフによる発言やプレゼンテーション、W3Cに関わるイベントの記録が掲載されています。



Past News
W3Cからのプレスリリースや新聞・雑誌に掲載された記事をここでは集めています。過去の情報を知りたい人はこのページが便利です。

英語で読むのはちょっと... と考えているあなたへ

英語は苦手だから... という人は、仕様書翻訳リンク集を見てみましょう。トップページの右にある「Translations」が、上段にある「About W3C」内の「list of translations of technical reports」をクリックすると「Translations at W3C」というページに飛ぶことができます。このページからたどれる日本語翻訳集は、W3Cのメーリングリスト上で報告された日本語翻訳文書へのリンクを集めたものですが、正式な仕様書は英語版ですので見比べる必要があります。また、日本語だけでなく、ドイツ語、フランス語、イタリア語、中国語などの翻訳も掲載されています。

