



## XMLデータとWeb2.0の親和性

日本アイ・ビー・エム株式会社  
コンサルティング・テクノロジー・エバンジェリスト  
米持 幸寿      [pandrxbox@jp.ibm.com](mailto:pandrxbox@jp.ibm.com)



# XMLとは何でしょうか？ 答え：表現形式です

- 複数のデータの集まりを
- テキストだけで表現する方式

## CSV

```
"date","code","title","speaker"  
2006/07/06,F4,"Eclipse進化の課程とこれから","米持幸寿"  
2006/07/06,F5,"XMLデータベース本格活用元年","米持幸寿"
```

## Label and Value pare

```
[recture]  
date=2006/07/06  
code=F4  
title=Eclipse進化の課程とこれから  
speaker=米持幸寿  
  
[recture]  
date=2006/07/06  
code=F5  
title=XMLデータベース本格活用元年  
speaker=米持幸寿
```

## XML

```
<Rectures>  
  <recture>  
    <date>2006/07/06</date>  
    <code>F4</code>  
    <title>Eclipse進化の課程とこれから</title>  
    <speaker>米持幸寿</speaker>  
  </recture>  
  <recture>  
    <date>2006/07/06</date>  
    <code>F5</code>  
    <title>XMLデータベース本格活用元年</title>  
    <speaker>米持幸寿</speaker>  
  </recture>  
</Rectures>
```

# ティム・オライリー 「XMLはWeb2.0にとって重要」

## developerWorks インタビュー: Web 2.0 EXPO Tokyoにてティム・オライリー氏に聞く

XMLデータベースへの期待と日本のエンジニアへのメッセージを語る

レベル: 初級

米持 幸寿, テクノロジー・エバンジェリスト, IBM

2007年 11月 19日

### ページオプション

ページを印刷する

ページをeメールで送信

2007年11月15、16日、アジアで初となる [Web 2.0 EXPO Tokyo](#) が開催されました。私は、アドバイザー・ボード・メンバーとして、USからロッドスミスを迎え、キーノート・スピーチの準備をするなど、このイベントを裏で支えてきました。Web 2.0 という単語の生みの親とも言えるティム・オライリー氏と、2日目のキーノートの直後にちょっとした会話をする機会を得ました。ティムはとてもフランクに、そして、楽しそうに会話してくれました。そのときの様子をIBMのXMLデータベース/Info 2.0 エバンジェリストである中林紀彦とともにお送りします。

**米持:** ティム、Web 2.0を活用する上でXMLの重要性をきかせてくれる？

**ティム:** セマンティックWebやマイクロフォーマットに代表されるような、メタデータの埋め込みやフォークノミー (folksonomy) において、構造化されたデータやタギング (tagging) を処理することにおいて、XMLはとても重要であると確信してるよ。

それから、それぞれの分野でプロトコルとかのXML規格スタンダードが存在することが重要で、例えばSNSのFacebookのFacebook Markup Language (FBML)のような標準が存在することによってSNS間で標準ベースでのデータ交換が可能になり、XMLがWeb 2.0の世界でより自然に選択されていくだろうね。

**米持:** XMLの機能性についてデータベースに期待することは？

**ティム:** XMLを使うシステムにおいても、データベースはとても便利でパワフルな存在だよ。データベースがXMLを直接扱えるようになる事によって、XMLデータの格納や取出しなどがとてもシンプルになり、あまり面倒な処理を記述しなくても、簡単にWebサイトが構築できるようになるでしょ？ そういう点でXMLのサポートはデータベースにとって非常に重要であると思うし、Webサイトを開発する側にとっては大きな驚きとなることだろうね。



## Webに、XMLの未来はあるか？

- 答えは「Yes」
- ただし下降気味
  - JSONが台頭
  - XHTMLが失敗
- ブラウザーは問題を抱えている
- XHTMLはセキュリティー問題を解決しない

Yahoo!アーキテクト  
ダグラス・クロックフォード



### Douglas Crockford's Blog

I was invited to speak at XML 2007 last month. I was given the topic "Does XML have a future on the web?" My answer was "yes." As evidence, I offered that there are still people selling Cobol compilers. Once this stuff gets into the enterprise, it can take generations to get rid of it.

A more interesting question is "Is XML on the web trending up or trending down?" Clearly, it is trending down. For data transfer applications, XML is losing ground to JSON because JSON is simply a better data transfer format. And XHTML has failed to displace HTML in the marketplace. The benefit of clientside validation has proven to not be a benefit.

I think you can argue, and in fact I did argue, that because of W3C's adventures with XML, the web itself may not have a future. The browser has a lot of problems, the worst of which are the security problems that came with Netscape Navigator 2. That was 12 years ago, and there has been no progress since that time in fixing the fundamental problems. There have been lots of patches on top of patches. Nothing more.

The web has grown up from a document delivery system to an application delivery system. But the browser has not kept pace, so there are now new proprietary platforms from Adobe and Microsoft and others that are hoping to replace the web.

Michael Sperberg-McQueen was there, and has described my argument as a ~~truly~~ mystifying rhetorical move. Michael and I were on the same panel, at which he wished he had said

<http://xml.sys-con.com/read/481198.htm>

# じゃらんWebサービス(リクルート)

## -<Results>

<NumberOfResults>21</NumberOfResults>

<DisplayPerPage>1</DisplayPerPage>

<DisplayFrom>1</DisplayFrom>

<APIVersion>1.2</APIVersion>

## -<Hotel>

<HotelID>300000</HotelID>

<HotelName>ホテル日航千歳</HotelName>

<PostCode>066-8520</PostCode>

<HotelAddress>千歳市本町4-4-4</HotelAddress>

## -<Area>

<Region>北海道</Region>

<Prefecture>北海道</Prefecture>

<LargeArea>石狩・空知・千歳</LargeArea>

<SmallArea>支笏湖・千歳</SmallArea>

</Area>

<HotelType>ホテル</HotelType>

## -<HotelDetailURL>

[http://jws.jalan.net/Redirect/V1/?key=psc111594ba511&rd\\_key=MzAwMDAwICAgIDAwMDE=](http://jws.jalan.net/Redirect/V1/?key=psc111594ba511&rd_key=MzAwMDAwICAgIDAwMDE=)

</HotelDetailURL>

<HotelCatchCopy>北欧風レンガ造りの美しいホテル。隠れた千歳の自然を楽しめます</HotelCatchCo

## -<HotelCaption>

空の玄関、新千歳空港から車で8分。ホテルの近くには原始に息づく支笏湖をはじめ、雄大なゴルフ場、舞の近くにありながら、北海道の大自然を満喫できます。

</HotelCaption>

## -<PictureURL>

<http://www.jalan.net/jalan/images/pictM/00075221.jpg>



## 最近のWeb 2.0でのトレンド

### ■ XML

#### – RESTful サービス

- APML
- リクルート
- 価格コム

#### – ドキュメント

- ODF

### ■ XML以外

#### – HTML

#### – JSON (Ajax のプロトコル)

#### – micro format

#### – YAML

#### – RSD



## XMLにはあって、他にはないもの

- 基本技術
  - 属性
  - スキーマ言語(検証、型の変換、属性の生成、プログラムの自動生成)
  - データの型(XSD)
  - 名前空間
- 拡張技術
  - エンベロープ(プロトコル仕様)
    - セキュアプロトコル
    - トランザクション
  - デジタル署名
  - スタイルシート
  - ビジネス・ドキュメント(xhtml、NewsML、ContactXML、BPEL、ODF・・・)



## スキーマ言語の役割

- 検証(validation)
  - 意図しているものと合致しているか確認できる
- 動作の制御
  - 型(string、int など)の自動変換
  - 属性の規定値の生成(データの省略が可能)
- プログラムモジュールの自動生成
  - WSDL → スタブ生成
  - XSDBean
  - JAX-RPC、JAX-B、JAX-WS

## データ型 (*type*) の適用方法

name	string	米持幸寿
age	int	25

strongly typed

```
<name type="string">  
米持幸寿  
</name>  
<age type="int">  
25  
</age>
```

weakly typed

```
<name>  
米持幸寿  
</name>  
<age>  
25  
</age>
```

```
<element name="name" type="string"/>  
<element name="age" type="int"/>
```



## たとえばJSONと比較してみる

- JSONの優れている点
  - バイト数が少ない
  - JavaScript からアクセスするプログラミングがシンプル
  - JavaScript 上でのアクセス速度が速い(DOMに比べ)
- XMLの優れている点
  - スキーマ言語がある
  - データ型がある
  - 名前空間がある(ネスト構造が作りやすい=モジュール化)
  - JavaScript 以外でも使いやすい

## 結論

### ■ XML

- 外部にサービスするデータ
- 業界標準フォーマット
- 長期保存するデータ
- ややこしいプロトコル(ベンダーが使いたがる)

ずっと使うなら

### ■ XML以外

- コンテンツ(寿命が短い)
- 勝手に使うデータ(Ajaxの通信プロトコルなど)
- Ajax

ちょっと使うなら

**XML活用にはとにかく XML Schemaが重要**

