XML と WWW の発展

技術本部 技術研究室 室長 稲垣陽一

１．はじめに

1998年２月、XML（eXtensible Markup Language；拡張可能なマーク付け言語）がＷ３Ｃ（World Wide Web Consortium）の勧告[*1]となって以来、多様な領域でその応用が試みられている[*2]。XML の応用領域は、BS データ放送から Java の GUI 生成ツールまでほとんど何でもありという様相を呈している。中でも EDI(Electrical Data Interchange)、EC（Electronic Commerce）への応用は、われわれのビジネスと直結したトピックである。

しかし、あえて本稿では以上のような文脈から一歩離れ、XML に関わる多様な動きの背後にある導因を探究したい。特に、Tim Berners-Lee（HTTP の設計者）をはじめとする WWW（World Wide Web）アーキテクトたちの考えるシナリオを検討しつつ、世界中を巻き込みながら進展するデジタル化のひとつの潮流として XML を捉え、その方向性を考察する。

２．WWW の技術拡張

2.1 自由拡張の不文律

WWW の急速な技術拡張を可能にしたのは、実は電子メール開発者たちのコミュニティから継承された暗黙のルールであった。それは「自分の知らない要素は無視せよ」という不文律である。

たとえば、メールソフトは自分の処理できない電子メール・ヘッダがあれば、ただ単に無視する。同じく Web ブラウザは理解できないタグを無視する。このような不文律のおかげで、ソフトウェア開発者は独自の技術拡張をインターネット上で実地に試すことができる。この暗黙の原則は、WWW が誕生した1989年から現在に至るまで、WWW の分散的な技術開発に適用され、その急速な技術革新を背後から支えてきた。

2.2 原則の危機

しかし、この不文律は同時に問題も引き起こす。たとえば、独自拡張された HTML のテーブル・タグ（データを表形式で表示するためのマーク）をブラウザが処理できない、つまり不文律により無視してしまう場合、われわれが目にするのは意味不明の文字列である。自由拡張の不文律は分散的な技術革新に寄与しながら、その一方で Web の信頼性を低下させる原因にもなってきた。拡張が最終的なコンセンサスに至らない間は相互運用性が保証されず、最悪の場合は半永久的な WWW の断片化をもたらしてしまう。

2.3 拡張ニーズのさらなる増大

ところが、WWW の急速な普及は、拡張ニーズをよりいっそう増大させている。たとえば、数学者のコミュニティ

＊１）XML 勧告は以下から得られる：
http：//www.w3.org/TR/REC-xml（原文）
http：//www.fxis.co.jp/DMS/sgml/xml/rec-xml.html（邦訳）

＊２）多様な応用事例の紹介は以下を参照：
http：//www.oasis-open.org/cover/xml.html#xml-webCollections

は数式表示のための言語拡張を必要とし、電子商取引では受発注帳票や製品カタログなどの様々なビジネス・データを Web を介してやり取りしたい。これら特定用途のためにみんなで個々バラバラに HTML を拡張していったならば、WWW の断片化は必至である。断片化すれば、誰もが安価に接続できるという、WWW のインフラとしての最大の長所は消滅してしまう。一方、拡張を管理する中央機関を設置するという方法は、脱中心をポリシーとするインターネットの世界とは融和せず、また急速な分散技術開発というメリットを失うことにもなるだろう。

３．WWW の断片化を防ぐ XML

3.1 拡張可能な言語への要求

拡張ニーズが増大する中で、自由拡張の不文律が破綻しつつある。この問題を解決するためには、次のような要求を満たすアーキテクチャを導入する必要がある。

要求Ａ：特定用途のための言語拡張が可能であること
企業、産業、団体などに特化したローカルな言語拡張を許容し、またその拡張を WWW 上で実地に試験できること。

要求Ｂ：拡張言語間で情報の相互利用が容易であること
ローカルな拡張が可能であるとともに、そのリソースを外部からも容易に利用できることが要求される。なぜなら利用が困難であれば WWW は断片化してしまうからだ。

上記２つの要求から、次のような特性が要請される。

（α）拡張が明確に定義できる
（∵拡張が入り交じってはカオス状態）

（β）拡張を容易に処理できる
（∵容易でなければ断片化）

この２つの特性を合わせ持つアーキテクチャは、様々なコミュニティにおいて独自の言語拡張を可能にしながら、同時にコミュニティ外のコンポーネントはコスト・エフェクティブに拡張を取り込むことができる。コミュニティ外のリソースを漸次、組み合わせて発展していくことが可能となる。分散的に実行された拡張が効率的に統合できるならば、WWW の断片化は阻止される。

3.2 オブジェクト指向？

DCOM、CORBA などのオブジェクト指向環境は、インタフェースが明確に定義されたシステムであり、特性αを満たす。オブジェクト指向システムでは利用できる API が事前に厳密に定義され、曖昧性の生じる余地はない。しかし、システムの構成要素は事前に定義されたインタフェース全体に対してコミットしなければならない。このことは特性βと適合しない。求められているのは、「拡張の厳密な定義」と「拡張の容易なハンドリング」の両方が可能な機構である。

3.3 Enabler－XML

このような背景の中、前述の要求 A と B を満たすことができる機構として、XML は登場した。XML は、SGML（Structured Generalized Markup Language, ISO8879）のサブセットであり、インターネット上でのデータ交換を効率的に実現するために SGML を簡略化した標準である。XML の仕様作成では「SGML の20%の仕様で80%の機能を実現する」ことが目的とされた。付属的な機能を削除することにより、XML ソフトウェアの開発は極めて容易になっている。

3.4 XML と HTML

一方、現在 Web の標準記述言語である HTML と XML との関係はどのようなものであろうか。HTML は SGML のひとつのインスタンス（SGML 規格に則り開発されたマークアップ言語）である。そういう意味で XML は HTML と次元の異なる規格であり、実際、HTML4.0は XML のインスタンスとして標準化されている。つまり、HTML4.0は拡張可能なマーク付け言語 XML のひとつの応用なのである。以上の関係を図１に示す。

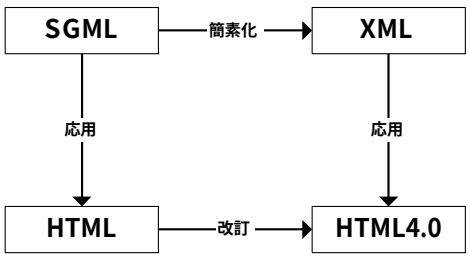

図１　XML と SGML、 HTML の関係

HTML は、人間による文書理解を容易化するためのマークアップ言語である。文書の表示構造をマーク付けする。そのスコープはグローバルであり、Web 上のすべての人々により共有される。このように、HTML が人間による理解を目的としているのに対して、XML にはそのような制約はない。現在試みられている XML 応用の多くは、特定のコミュニティを適用範囲として、データの内容をマークする。内容に基づくマーク付けは、文書（SGML／XML ではデータのことをすべて文書と呼ぶ）のソフトウェアに

表１　HTML と XML インスタンス言語

記述言語	マーク付けの対象	スコープ	主目的
HTML	表示構造	グローバル	文書の人間理解
XML	データの内容	コミュニティ	自動処理など

よる自動処理という大きなアドバンテージをもっている（表１参照）。

3.5　言語拡張の仕組み

ここで、XML ではどのように言語拡張を行うのかを簡単に見てみる。XML では、独自の言語拡張をおこなうためにタグ集合を文書型定義（DTD：Document Type Definition）として規定する[*3]。

DTD は、対象となるタイプの文書内部に現われる要素およびその木状構造（包含関係、順序関係、出現回数など）を規定する（図２）。

たとえば、「名刺」という１つの文書を考えたとき、次のような要素が抽出される。

・氏名
・所属（企業、部署、役職）
・連絡先（住所、電話、電子メール）

DTD では、これらの要素とその文書中での包含・順序関係を記述する（図３）。

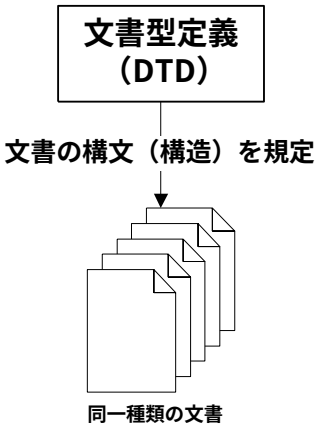

図２　DTD と文書

```
<!ELEMENT 名刺（氏名，所属，連絡先）＞
<!ELEMENT 所属（企業，部署，役職？）＞
<!ELEMENT 連絡先（住所，電話，電子メール＊）＞
<!ELEMENT 氏名（＃PCDATA）＞
・・・etc.
```

（注１）「役職」の後ろの ？ ’は０または１回の出現制約を意味する。たとえば役職者の場合とそうでない場合に対応するため。
（注２）「電子メール」の後ろの ＊ ’は０回以上の繰り返しを意味する。
（注３）（＃PCDATA）はその要素が文字列であることを示す。

図３　DTD による記述の例

この名刺 DTD に適合する XML 文書は、たとえば図４のようになる。

DTD に則ってマークアップされた文書は、解析・情報抽出・自動生成・変換などを実行するソフトウェアをかなり容易に実装できる。実際、JavaScript、Perl、Java、C++など多様なプログラミング言語が、XML のためのライブラリを用意している。

以上、概観してきたように、Web の記述言語として XML が導入されることにより、自由拡張によるカオス化の危険なしに、特定用途向けの言語拡張を安全に行うことが可能となる。XML とその関連規格は、コミュニティ・ワイドなアプリケーション開発の触媒となり、領域に特化した様々な応用が今後試みられていくと予測される。

４．XML の次にくるもの

4.1　DTD を越えてオントロジーへ

XML による情報共有・交換の効率化の鍵は、文書型定義 DTD である。人工知能系の研究では、あるコミュニティの成員が一定の理解を共有し、知識の交換をそれに基づいて行うような諸概念の辞書をオントロジー（Ontology）と呼んでいる。視点を変えると、実は XML の文書型定義 DTD は（簡素ではあるが）オントロジーそのものとみなすことができる。XML　DTD は特定領域のオントロジーをコスト・エフェクティブに開発する手段であり、DTD（＝簡易オントロジー）が開発されることで、従来は人間の理解を仲介しなければならなかった様々なタスクが自動化され、情報の共有・伝達が効率化するわけである。

では、次のステップは何だろうか。先に DTD は構文スキーマ（要素の構造、データ型・出現制約など）を記述することができると述べた。しかし、DTD は意味を定義す

```
＜名刺＞
  ＜氏名＞稲垣陽一＜／氏名＞
  ＜所属＞
    ＜企業＞㈱シーエーシー＜／企業＞
    ＜部署＞技術研究室＜／部署＞
  ＜／所属＞
  ＜連絡先＞
    ＜住所＞〒112 東京都文京区・・・＜／住所＞
    ＜電話＞03-3263-7241＜／電話＞
    ＜電子メール＞yoichi@inagaki.com＜／電子メール＞
 ＜／連絡先＞
＜／名刺＞
```

図４　XML 文書の例

＊３）実際には、SGML では必須である DTD が、XML では必須ではない。これは整形式（well-formedness）という概念の導入による。しかし本式に XML を利用する場合にはやはり DTD が不可欠であろう。

ることができるわけではない。次のステップは、意味記述による本格的なオントロジー作成ではないかと筆者は推測する。

4.2 概念スキーマ言語

意味的関係を記述する言語は概念スキーマ言語と呼ばれるが、実は概念スキーマ言語それ自身がXMLの応用マークアップ言語として開発されつつある[*4]。DTDがボキャブラリと構文スキーマを定義するのに対して、概念スキーマ言語は概念およびその間の意味的関係を記述する。先の名刺の例でいえば、「人、企業、部署、住所、電話、電子メール」といった文書要素（＝概念）について、概念スキーマ言語は「企業は部署から構成される」「人は企業に所属する」などの宣言的な知識を定義する。

意味的関係が形式的に定義され、それが容易に共有できた場合のメリットは計り知れない。そのため、米国はKIF（Knowledge Interchange Format）[*5]やcyc[*6]など知識表現システム（KR System, Knowledge Representation System）の研究に多大な投資をしてきた。しかし普及は容易ではないようだ。KR Systemが有効な推論をするためには十分な知識が不可欠であるが、現在は知識の収集にボトルネックが存在する。しかし、XMLベースの概念スキーマ言語により知識ベースの共有・再利用が容易化する可能性がある。

たとえば、HTTPの作成者Tim Berners-Leeにより提唱されるSemantic Web[*7]というビジョンは、非常にシンプルなXML応用スキーマ言語により、WWWが地球規模の分散知識ベースに発展することを目指している（図５）。

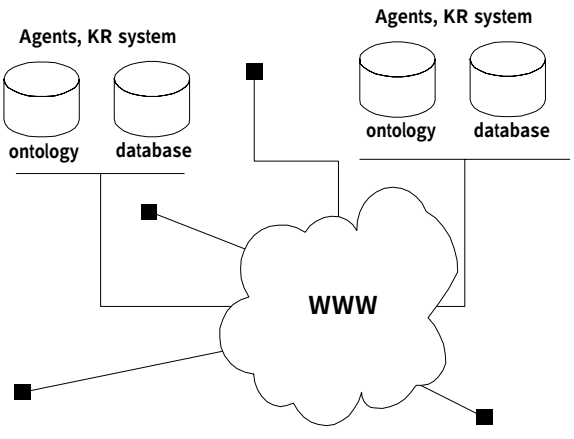

図５　Near Future WWW

５．XMLとWWW発展のシナリオ

本稿では、XMLを軸とするWWW発展のシナリオを概観した。その要点をまとめると次のようになる。

- XMLにより特定用途向けのWebアプリケーションが可能となる。
- XMLスキーマ言語が開発され、WWWは地球規模の分散知識ベースへ発展する。

かつて、HTTP／HTMLという非常にシンプルな情報共有の仕組みが全世界に急激に普及し、様々な驚くべき応用が生まれた。同様にXMLとスキーマ言語は次世代のWebアプリケーションの触媒となるだろう。特に、HTMLからXMLへの移行が起こる今後数年間は、WWWにとって本当にクリティカルな期間となるだろう。

参考文献

１．Editors：Tim Berners-Lee, Dan Connolly, W３C Note 10 Feb 1998
Web Architecuture：Extensible Language
http：//www.w3.org/DesignIssues/Extensible.html

２．Dan Connolly, Rohit Khare, Adam Rifkin, 1998/01/15
The Evolution of Web Documents：The Ascent of XML
http：//www.cs.caltech.edu/~adam/papers/xml/ascent-of-xml.html

３．Tim Berners-Lee, March 1998（modified1998/10/14）
Evolvability
http：//www.w3.org/DesignIssues/Evolution.html

４．Tim Berners-Lee, September1998（modified：1998/11/15）
“Web Architecture from 50,000 feet”
http：//www.w3.org/DesignIssues/Architecture.html

＊４）Resource Description Framework, XML-data, Ontology Markup Languageなど。表現能力はそれぞれ非常に異なる。

＊５）http：//logic.stanford.edu/kif/kif.html

＊６）http：//www.cyc.com/index.html

＊７）Semantic Web Road Map：http：//www.w3.org/DesignIssues/Semantic.html