

# XLink を利用した XML ドキュメントブラウザ

後藤正智

(株)富士通研究所 情報サービス研究部

Web の世界のマークアップ記述言語として HTML が利用されたことで、構造化文書と、ハイパーリンクの概念が急速に広がり一般的なものとなった。しかしながら、HTML のリンクは、ただ単にリンクを記述した場所と指定したアドレス先とを関係づけるだけのシンプルなものであるため、より洗練された複雑なリンク（例：外部からある文書中のオブジェクト間のリンクを貼ることや、リンクを往き来したり、複数オブジェクト間のリンクなど）を提供することができない。そこで XML 上で実現されるハイパーリンクの仕様として XLink (XML Linking Language: XLL) がドラフトとして策定中である。今回はこの XLink の処理系の設計方法とその処理系を利用したアプリケーションについて考察する。

## XLink Aware XML Document Browser

Masatomo Goto

Information Service Architecture Laboratory,  
Fujitsu Laboratories Ltd,

With growing the markup technologies: SGML, HTML and XML, we thought that hyperlinking is also important technology to manipulate any type of information. But the hyperlink which HTML provides is very simple so that it cannot represent some useful hyperlinking functions: link objects from out of that document, multidirectional link traversal, etc. Then, XLink, which is the hyperlinking language on the XML is now discussed in the W3C. This paper describes an implementation design of XLink and some application examples using this XLink engine.

## 1 概要

本論文は、HyTime をベースとした XLink エンジンの設計・実装方法ならびに、そのエンジンを利用したアプリケーションの設計について記述したものである。

構造化文書の表現法に SGML が国際規格として制定され、文書の論理構造とスタイル情報を区別する重要性が示された。また WWW の分野において HTML がマークアップ言語として利用されたことで文書を構造化することやハイパーリンクの便利さや重要さが示された。そして其々短所を補う XML が登場し、様々な分野で研究がはじまっている。そんな中、ハイパーリンクを利用したオブジェクト間の関連付け（ドキュメントの相互参照や、ホットスポットティングなど）が IETM など で文書管理方法として注目されている。

しかし、従来実現されている HTML のリンクでは、ただ単に、リンクを記述した場所と指定したアドレス先とを関係づけるだけのシンプルなものであるため、表現力に乏しい。本研究のねらいは、より洗練された複雑なリンク（例：外部からある文書中のオブジェクトとあるオブジェクトのリンクを貼ることや、リンクを往き来したり、複数オブジェクト間のリンクなど）を提供することで有用な文書処理システムを実現することにある。

この様々なリンク表現を実現するため、我々はリンクの表現方法に HyTime を利用しようと考えた。しかし、HyTime はその高機能さゆえ、例えば有限座標空間の定義など、現在のコンピュータのリソースではハイパフォーマンスな動作が望めない。そこで internet 上で動作させること

を目的とし、XML をベースにしたリンク規約である XLink に着目した。XLink 規約は ISO である HyTime のサブセットと捉えることができる。従って、将来の機能拡張を考慮し、HyTime 規約をベースにした設計、実装を行った。

このため、本 XLink エンジンは XML 文書だけでなく、SGML 文書上でも利用できるようになった。つまり、XLink で提供されるリンク機能を正當に SGML 文書上で記述することが可能になった。

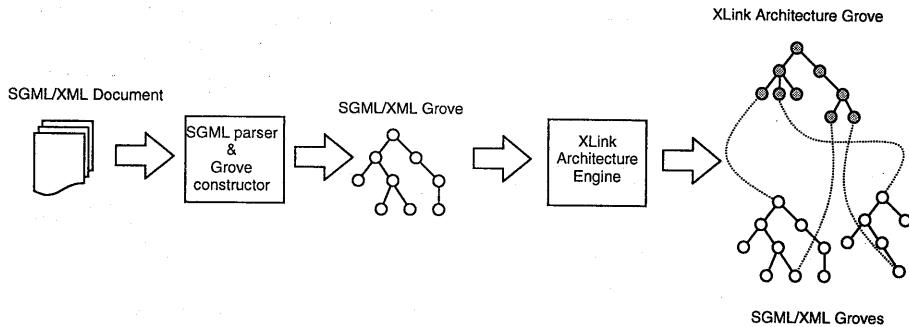
今回は、この XLink エンジンの実装技術ならびにアプリケーション例について報告する。

まず、2章でエンジンの設計・実装方法について説明し、HyTime ベースの XLink がどのように設計されるか示す。3章ではそのエンジンを用いたアプリケーション例について紹介し、この XLink エンジンの有効性を証明する。そして4章で将来の展望、エンジンの効果についてまとめる。

## 2 XLink エンジンの実装

### 2.1 エンジンの設計について

XML Linking Language(XLink) は HTML のリンクより洗練された表現力の高いリンク機能を XML 文書や SGML 文書上に記述できるようにする言語である。その機能として、HTML では一方向へしか行けなかったハイパージャンプを双方向可能にしたり、複数のオブジェクト間のリンクを表現したり、外部文書からある文書中のオブジェクト間のリンクを表現することなどが可能になる。



[図 2-1] XLink 処理の流れ

今回この XLink 処理部の設計に際し、アプリケーションの開発効率、汎用性の向上をめざし、アプリケーションに直接埋め込むのではなく、モジュール化し簡単な API を提供する構成とした。さらに HyTime へ将来拡張できるように、HyTime 同様、AFDR に沿った形で XLink アーキテクチャとして動作するように設計を行なった。

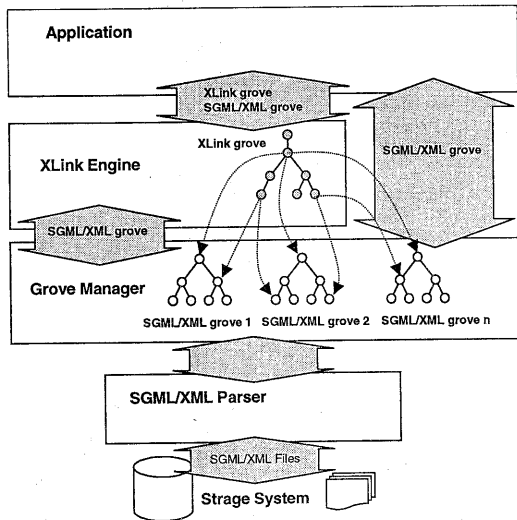
AFDR(Architectural Form Definition Requirement)とは、HyTime 第二版の ANNEX で定義されている、SGML 文書上にアーキテクチャを定義するための定義方法を定めたものである。

アーキテクチャとしての XLink 処理の流れは図 2-1 のようになる。まず Xlink 記述されているオリジナルの SGML/XML 文書を構文解析してグローブ（文書構造の表現形式の一つ）が生成される。そして XLink を処理するためのアーキテクチャとして必要な部分を抽出する。そして XLink 特有のリンク情報を表現するグローブを生成する。そして生成されたグローブにアクセスすることによって、XLink の動作を定義することである。

グローブを利用しているものとしては、SGML、HyTime の他に SGML の構造変換およびスタイル情報を指定する DSSSL (Document Style Semantics and Specification Language) がある。このように SGML の分野では、構造化文書記述、変換およびスタイル、リンクにおいてグローブという共通の文書表現形式を利用している。従ってグローブベースで動作する XLink エンジン設計することで、XML 処理をサポートする SGML システムを構築することが可能になる。さらに XLink で利用する XML 文書中の位置を特定するアドレッシングの言語として XPointer が定義されているが、これはグローブのようなツリー構造を利用した言語であるため実装が容易である。

XPointer を用いてリンク先を記述することで、HTML のリンクではファイル単位、もしくは #NAME アトリビュートを持つ <A> エレメントしかさせなかったが、あらゆる粒度で任意の箇所をさせるようになる。(パラグラフ中の特定の文字列など)

図 2-2 に、上記の方針（あるいは SGML グローブベース）で設計した XLink エンジンと、そのエンジンを含むシステム構成図を示す。



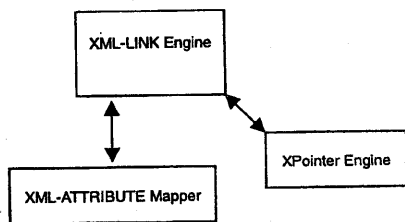
[図 2-2] システム構成図

グローブマネージャから得られる SGML グローブを元に XLink エンジンが XLink グローブを生成する。アプリケーションは SGML グローブとハイパーリンク情報を表現する XLink グローブを元に所望の処理をおこなうことになる。

## 2.2 エンジンの構成

この章では XLink エンジンのシステム構成について説明する。

図 2-3 に今回実装した XLink エンジンの構成図を示す。



[図 2-3] リンクエンジンの構成

この図に示されているようにエンジンは大きく分けて3つの構成要素からなる。

### XML-Link モジュール：

SGML/XML グローブから、XLink の情報を

持つエレメントを探し出し独自の XLink グローブを生成するモジュール。エンジンとの API は全てこのモジュールに存在する。

### XML-ATTRIBUTE モジュール：

XLink の xml:attribute 属性で指定されたアトリビュート名のリマップを認識してその情報を保持するモジュール。XLink モジュールからのみ利用される。

### XPointer モジュール：

href 属性中に記述された XPointer を解析、SGML グローブ中を探索して指定されたノードを探し出すモジュール。XLink モジュールが XLink グローブを生成する時、トラバース(リンクにより関係づけられたオブジェクト間を移動する)時などで利用される。

XPointer モジュールで提供するアドレスリング機能は、外部からもクエリとして利用されることを前提として設計した。

## 2.3 XLink プロパティセットの説明

HyTime の Annex の SGML の拡張機能で定義されているように、グローブは PSDR (Property Set Definition Requirement : プロパティセットを定義するための定義方法を定めたもの)に従って作成されたプロパティセットに基づき生成される。

プロパティセットは、グローブを構築するために必要な情報を表現するデータ表現である。グローブ中のノードが持つデータのタイプをあらわすプロパティと、ノードの型をあらわすクラスから構成され、そのプロパティやクラスはモジュールによってカテゴライズされる。

従って XLink グローブを生成する際にも、PSDR に沿って専用のプロパティセットを定義する必要がある。また、HyTime に拡張できる XLink 処理エンジンとして設計するために、HyTime のプロパティセットをベースに XLink のプロパティセットを設計した。つまり、HyTime のプロパ

ティセットのうち、XLink を表現するために必要なリンク情報に関するプロパティ、クラスが定義されている以下のモジュールを利用して XLink プロパティセットを設計した。

### 1. Bounded object set module

HyTime の BOS を表現するためのプロパティが定義されているモジュール。このモジュールから、拡張リンクグループを表現するクラス、プロパティを定義した。

### 2. Enabling architecture module

HyTime のアーキテクチャを表現するためのプロパティが定義されているモジュール。このモジュールから、XLink のリンクのタイプを表現するための列挙値のプロパティを定義した。

### 3. HyTime document module

HyTime 文書表現するモジュール。このモジュールから、XLink が記述されている XML/SGML 文書表現するクラスやプロパティを定義した。

### 4. Effective SGML document module

HyTime が定義されている文書の SGML グループを表現するモジュール。XLink でも同様に XLink が定義されている XML 文書表現するプロパティを定義した。

### 5. Hyperlink module

HyTime の様々なハイパーリンクを統一的に表現するクラスやプロパティが定義されているモジュール。このモジュールから simple, extended リンクを統一的に表現するようなクラス、プロパティを定義した。

### 6. Location address targets module

アドレッシングに関係するプロパティが定義されているモジュール。このモジュールから、href 属性中の XPointer で記述されたアドレス指定部分の解析結果を保持するためのクラスやプロパティを定義した。

### 7. Location address path module

HyTime では Location Address を表現するために利用されているモジュール。このモジュールから、href で指定されたノードをアドレス情報として格納するためのクラスとプロパティを定義した。

次にこれらのモジュール内のクラスやプロパティをベースに、新規に定義された XLink のプロパティセットについてクラスをベースに説明する。

#### *HyperXmlDoc*

Hyperdoc クラスに相当する。XLink のグローバルルート。ハイパーグループ内の XLink 文書のプロパティや、アドレッシングに関するプロパティ、リソースとなるノードに関するプロパティ、文書グループ情報を表現するプロパティを持つ。

#### *GrpDoc*

bosmem に相当する。拡張リンクグループの情報を表現する。ハイパーグループ内に存在するドキュメントとその steps 値を保持するプロパティを持つ。

#### *XmlDoc*

HyDoc クラスに相当する。XLink 文書表現。ハブドキュメントかどうかのフラグであるプロパティ、文書中のリンクエレメントをリストで持つプロパティ、該文書の SGML グループへの情報を表現するプロパティを持つ。

#### *ResObj*

anchobj クラスに相当する。このハイパーグループ空間内でリンクのリソースになる SGML ノードを指すプロパティとそのリソースのロケータ情報のリストのプロパティを持つ。

#### *XmlLink*

hylink クラスに相当する。XLink のリンク情報を表現。リンクのタイプ情報を示すプロパティとエレメント名や SGML 文書の該当ノードを示すプロパ

ティ、該リンク内のロケータのリストであるプロパティを持つ。

### Locator

anchor クラスに相当する。リンク、ロケータエレメントで記述された様々な属性値 (show, actuate, title など) を表現するプロパティ群とロケータにより指されているオブジェクトへのポインタ等をプロパティとして持つ。

### Reftargs

reftargs クラスに相当する。アドレス指定元の位置情報、アドレス後の位置情報、XPointer 情報を表現するプロパティを持つ。

### Href

locaddr クラスに相当する。HREF 属性の情報を、URL 名を保持するプロパティ、XPointer データを保持するプロパティ、さらにはコネクタタイプの情報を保持するプロパティとして表現する。

この XLink プロパティセットを基にアーキテクチャグループが生成されることになる。

## 3 XLink アプリケーション例

### 3.1 アプリケーション API

アプリケーションに必要な全ての情報は、構築された XLink グループ内にデータとして格納される。従って、リンク情報を取得するには SGML グループにアクセスするのと同じ方法で XLink グループにアクセスするだけでよい。それ以外に、エンジンとして提供すべき機能は、以下に示す 4 つのメソッドだけである。

#### 1. グループの構築を指示するメソッド

現在存在する XLink グループを破棄し、新規に XLink グループを生成するメソッド。

#### 2. リソースかどうかの確認メソッド

SGML グループ中のノードがリンクのリ

ソースであるかどうか調べるメソッド

### 3. リンクのロケータ情報取得メソッド

ある SGML グループのノードのロケータ情報を記した XLink グループのノードを取得するメソッド

### 4. リンクのトラバースメソッド

現在存在するハイパーグループ内でのノードの移動、またはこの空間内に存在しない文書を新たにメンバとして取り込み所定のノードへ移動するメソッド。

この API を用いることにより、アプリケーションは自由に、構築された XLink グループと、この XLink グループによってリンクされた複数の SGML グループ群にアクセスできるようになる。

つまり、XML 文書をハブ文書とするハイパーグループ空間：複数のグループがその空間内に存在し、それぞれがリンクされているような空間が構築されるのである。アプリケーションは、そこから必要なグループやグループの一部などを取り出して、合成することで仮想 XML 文書を生成・表示することになる。

### 3.2 アプリ例

XLink を利用したサンプルアプリに我々はドキュメントブラウザ、ならびにアニメーション機能を考えている。

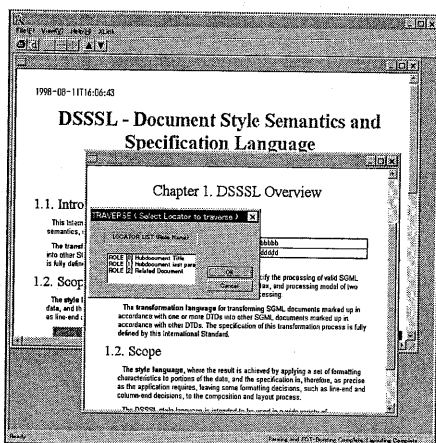
#### 1) ドキュメントブラウザ

SGML や XML で記述された文書を、DSSSL スタイルシートを利用して整形し印刷・表示するドキュメントブラウザに XLink で可能になった様々なハイパージャンプを提供する機能を追加する。

つまりこのドキュメントブラウザによって SGML や XML で記述された文書に HTML と同等のハイパージャンプ機能を提供できる。さらに、CDROM 内の文書間のハイパージャンプを可能にしたり、複数のリソ

ース間のハイパージャンプを可能にするなどの多様なリンク表現を提供し、文書間つまり情報のあらゆるナビゲーションを提供することができるようになる。

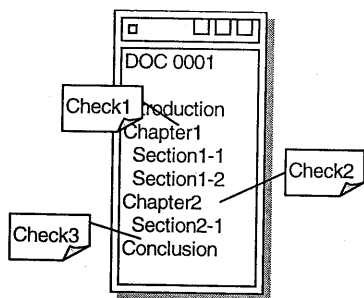
以下の図が、3.1 で説明した API を利用して XLink 処理エンジンを、DSSSL スタイルシートを利用した SGML/XML ドキュメントブラウザに統合したイメージである。



[図 3-1] ドキュメントブラウザ

## 2) アノテーション機能

ドキュメントブラウザにてメモが貼り付けられ、ユーザに提示し注意を促す機能。次の図のようなイメージになる。



[図 3-2] アノテーション機能

オリジナルドキュメント、注釈メモ双方からトラバース可能で、かつ双方とも XML

文書であり、双方ともリンク情報を含まない。

このアノテーション機能を実現するため、XLink の HTML のリンクにはない 2 つの機能を利用している。

1. 外部から文書中の一部をリンクのリソースとすることができる。
2. 双方向のトラバースが可能になる。

この機能を現在の WWW の環境で実現しようとするならば、HTML のリンクは一方向で双方向性がないため、アノテーション情報と文書とのトラバースは完全にアプリケーション依存になる。また、外部からオリジナル文書中にリンクを表現できないのでアノテーションへオリジナル文書からトラバースすることができない。

また、XLink にて記述することで、アノテーション情報は文書中にリンクとして表現されることになる。従ってアノテーション情報を解釈し専用のサポート機能を提供することができないアプリも XLink をサポートしていれば、そのままリンクとして表現することができる。つまり、アノテーション情報は保持され、結果として様々なブラウザでその情報が失われることなく表示できることになる。

## 4 結論

本論文では、ドキュメント管理において様々な種類のハイパーリンク機能を提供するために HyTime 的に XLink エンジン、XLink プロパティセットを設計し、そのエンジンを利用したアプリケーションについて紹介した。

HyTime 的に XLink エンジンを実装したことにより、容易に HyTime へ拡張することが可能である。また、このことは SGML の拡張機能を利用したことを意味するため、SGML 文書上にも XLink を構築するこ

とが可能になる。

さらに、この実装はグローブをベースとしたつくりになっているが、XML の文書表現のひとつであり、現在策定中の DOM をベースとして XLink エンジンを作成する場合もこの考えを応用できる。

また、このようなエンジン化により、アプリケーション側での負荷を軽減し、様々な分野でリンクエンジンを利用することが可能になる。さらに、このオリジナルの文書構造に直接的に手を加えず、グローブの集合の中から、必要なサブツリーを取得してアプリケーションが処理できるという link の特性により、SGML の正当性を損なわずにアプリ側で様々なことができるようになる。

しかしながら、描画に関する情報を実際に解釈して処理するのは、アプリケーションになるために、リンク表現をどのようにユーザに提示するかなどはアプリ独立になってしまう。つまり DSSSL などのスタイルシートで文書を表示するときなどは多方向リンクのときに、どのようにそのリソースをリンクとして表示するか決定することができないのである。現状ではアプリ依存になってしまうので、この状況にどうするか統一的な方法を定める必要がある。

XLink エンジン自体は、現在機能確認レベルの実装であるため、今後は性能向上につとめていく予定である。また XLink の仕様が現在ドラフトであるため今後のアップデートに対処していき、エンジンとしての特徴をふまえ、様々なアプリで使用できるように機能拡張をおこなっていく予定である。

## 参考文献

- [1] "Extensible Markup Language (XML) 1.0", W3C Recommendation, REC-xml-19980210, <http://www.w3.org/TR/REC-xml>
- [2] "XML Linking Language (XLink)" W3C Working Draft, WD-xlink-19980303, <http://www.w3.org/TR/WD-xlink>
- [3] "XML Pointing Language (XPointer)" W3C Working Draft, WD-xptr-19980303, <http://www.w3.org/TR/WD-xptr>
- [4] Information Processing - Text and office systems - Standard Generalized Markup Language (SGML) ISO 8879:1986
- [5] Information technology - Hypermedia/Time-based Structuring Language (HyTime) ISO/IEC 10744:1997
- [6] Document Style Semantics and Specification Language (DSSSL). ISO/IEC 10179:1996