XLink 標準に従ったリンク情報共有システムの構築

川畑　英明　清水　將吾　大場　充
広島市立大学情報科学部情報メディア工学科

1. はじめに

HTML ファイルにリンクを作成する際，アンカーテキストとそれに対応させる URL が決まっている場合やページ作成者が知らない有益なリンクを定義したい場合等，同一サイト内，コミュニティ内でリンク情報を共有したいことがある．リンク情報を個別に管理するとリンク先の URL 変更の検出や相互参照等のリンクの構成管理が難しいという問題がある．

特定の URL に対して，同一のアンカーテキストが使用されることが多いとき，アンカーテキストと URL の組合せを共有することにより，Web ページの作成者が個別に URL を調べてリンク先を記述する手間を省くことができる．実際，Google のバックリンク機能を用いて，広島市立大学の日本語トップページ http://www.hiroshima-cu.ac.jp/japanese/へのリンクを調査した結果，「広島市立大学」という文字列を含むアンカーテキストからのリンクが全体の 69%を占めており，英語表記である「Hiroshima City University」を含めれば全体の 76%を占める．

リンク情報共有の例として，生物学データベースの統合検索システムである DBGET/LinkDB[3]が挙げられる．LinkDB では，核酸配列とそれを翻訳したアミノ酸配列，配列データとそれが記載された文献等，異なるデータベース間のエントリ(キー)の関連を二項関係の表として作成しており，これをたどることにより多くの関連データを容易に求めることができる．

我々は，リンク情報の記述形式として W3C が提案する XLink[4]を採用し，ページ作成者がリンク情報サーバに対して問合せを行うことによりリンク情報の共有を可能にするシステムの構築を行った．XLink はリンク対象データとして XML 文書の任意の範囲を指定でき，またリンク情報を内容とは別のファイルに記述できる．

2.XLink リンク情報共有システム

2.1 XLink

XLink は W3C が提案する，多方向，双方向リンクといった高機能なリンクを表現するための仕様である．これらの高機能なリンクは拡張リンクと呼ばれる．本稿では，簡単のため，HTML リンクと同機能の単純リンクのみを扱う．単純リンクは Mozilla，Amaya 等の Web ブラウザでサポートされている．また，リンク情報はリンクを付与する文書とは独立に記述でき，これはリンクベースと呼ばれる．図 1 にリンクベースの記述例を示す．この例では一番目の locator 要素の href 属性に「広島市立大学」という文字列を指定する XPointer 式が記述されている．ここで，href 属性に URL が記述されずに XPointer 式のみが記述されている場合は任意の URL を対象とするように解釈を拡張する．二番目の locator 要素で実際の広島市立大学の URL を定義している．そして go 要素では，「広島市立大学」という文字列から広島市立大学の URL へのリンクを定義している．

```
<link xlink:type = ”simple”>
    <locator xlink:type = ”locator”
        xlink:href =
          ”#xpointer(string-range(//*,”広島市立大学”))”
         xlink:label = ”hcu_anchor” />
    <locator xlink:type = ”locator”
            xlink:href =
             ”http://www.hiroshima-cu.ac.jp/japanese/”
              xlink:label = “hcu” />
    <go xlink:type = ”arc”
          xlink:from = ”hcu_anchor”
         xlink:to = “hcu” />
</link>
```

図 1.リンクベースの記述例

2.2 システムの概要

本システムでは，リンク情報をリンクベースの集まりとして Web 上に配置したデータベースにより集中的に管理する．各サービスは Web サービスとして実現し，API を WSDL により公開する．クライアントは作成したリンクベースを登録 API を用いてデータベースに登録する．また，クライアントは必要なリンク情報をデータベースから検索 API を用いて取得する． API の詳細は次節で述べる．その後，取得したリンクベースを用いてファイルの該当箇所にリンクを追加する．図 2 にシステムの構成図を示す．以下，図中の番号に従って説明する．

登録処理の流れは以下の通りである．

(1) クライアントは登録 API に従って登録を要求するリンクベースとパラメータをサーバに送信する．
(2) サーバは受け取ったリンクベースをデータベースに登録する．

検索処理の流れは以下の通りである．

(3) クライアントは検索 API に従って検索に必要なパラメータをサーバに送信する．

Implementation of a Link Sharing System Conforming to the XLink Standard

Hideaki KAWABATA
Shougo SHIMIZU
Mitsuru OHBA

Hiroshima City University
Faculty of Information Sciences
Dept. of Computer and Media Technologies

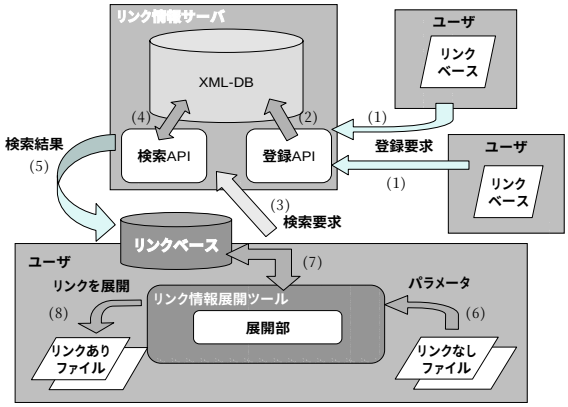

図 2.システム構成図

(4) サーバは受け取ったパラメータから検索式を作成し，データベースを検索する．
(5) 検索結果をクライアントに返す．

また，リンクを展開する場合は以下のようになる．

(6) リンクの付与されていない文書を展開部に渡す．
(7) ローカルにあるリンクベースの集まりから必要なリンク情報を取得する．
(8) 展開部は取得したリンク情報を文書に付与する．

3．API の提供機能

3.1 登録 API

登録 API は，クライアントからの登録要求を受け，ユーザが作成したリンクベースをデータベースに登録する処理を行う．各リンクベースには登録者を一意に識別するための登録者 URI をパラメータとして付与する．

サービス名は registerDataBase とし，Message 型サービスとして配置する．登録 API のパラメータを以下に示す．

(1) 登録者 URI，リンクベース：リンクベースに登録者 URI を属性値として追加し，データベースに登録する．

3.2 検索 API

検索 API は，クライアントからの検索要求を受け，データベースからリンク情報を取り出しクライアントに返す処理を行う． サービス名は searchDataBase とし，Message 型サービスとして配置する．検索 API のパラメータを以下に示す．以下，パラメータの値が指定されていなければ，任意の値にマッチングさせることとする．

(1) 登録者 URI：指定された登録者 URI をもつリンクベースを返す．
(2) 登録者 URI，URL：指定された登録者 URI をもつリンクベースから，指定された URL を href 属性値としてもつようなノードの集合をリンクベースとして返す．
(3) 登録者 URI，ラベルテキスト，type 属性の種類：指定された登録者 URI をもつリンクベースから，指定されたラベルテキストを label 属性値としてもち，かつ指定された type 属性値をもつようなノードの集合をリンクベースとして返す．
(4) 登録者 URI，リンク元 URI，リンク先 URI：指定された登録者 URI をもつリンクベースから，指定されたリンク元 URI を from 属性値としてもち，かつ指定されたリンク先 URI を to 属性値としてもつようなノードの集合をリンクベースとして返す．

4.実装

本システムは Java 言語で実装した．データベースには，JVM 上で動作する無償のネイティブ XML データベースである Apache Xindice[1] を採用した．Xindice は XPath[5]を検索言語として採用している．また，サーブレットコンテナには Jakarta Tomcat，SOAP エンジンには Apache Axis を用いた．

検索処理では，受け取ったパラメータから XPath 式を作成し，データベースへの検索を行う．パラメータの種類が 3.2 節の (3)の場合の XPath 式の作成例を以下に示す．サーバが受け取ったパラメータ {userURI，ラベルテキスト，type 属性の種類} が {kawabata，hcu，locator} のとき，サーバで作成される XPath 式は

//locator[@xlink:userURI='kawabata'][@xlink:label='hcu'][@xlink:type='locator']

となる．この XPath 式は userURI 属性値が kawabata であり，label 属性値が hcu，かつ type 属性値が locator であるような locator 要素の集合を指定している．

クライアントは，検索用パラメータを属性値としてもつ DOM の Document オブジェクトを作成し，Axis のサービス呼出しメソッドを実行する．検索結果は DOM の Document オブジェクトとして受け取った後，それをファイル化する．

5.まとめと今後の課題

本研究では，XLink リンクベースを用いたリンク情報共有システムの構築を行った．これにより，コミュニティ内で有益なリンクの共有や相互参照の構成管理が容易に行えるようになる．

現在の方式では，最新のリンク情報を得るためにユーザが定期的にサーバに問い合わせる必要がある．そこで，登録されているページが移動されたときに，同ページを参照しているサイトに更新通知を行う，といった機能の追加が必要であると考えられる．

本システムはコミュニティ内でリンク情報を共有するための基礎となるシステムである．今後は Gene Ontology[2]等のドメイン固有のオントロジや XLink の拡張リンク機能を活用した，より高度なリンク情報共有システムを開発する予定である．

参考文献

[1] The Apache Software Foundation, Xindice.
http://xml.apache.org/xindice/
[2] Gene Ontology Consortium.
http://www.geneontology.org/
[3] GenomeNet, DBGET/LinkDB.
http://www.genome.ad.jp/dbget-bin/www_linkdb
[4] World Wide Web Consortium (W3C), XML Linking Language (XLink) Version 1.0.
http://www.w3.org/TR/xlink/
[5] World Wide Web Consortium (W3C), XML Path Language (XPath) Version 1.0.
http://www.w3.org/TR/xpath/