

# 電子ジャーナルにおけるXML技術応用性研究のための試作開発

清水卓彦 上羽清彦

NTTラーニングシステムズ株式会社

〒103-0004 東京都中央区東日本橋 1-11-4 NTT浜町ビル 6F

Tel:03-5822-4371 Fax:03-5822-4373 E-mail:shimizu@stage.gr.jp

白木澤佳子

科学技術振興事業団

〒102-0081 東京都千代田区四番町 5-3

Tel:03-5214-8402 Fax:03-5214-8420 E-mail:shirokiz@jst.go.jp

## 概要

学術論文をインターネットにより発信する電子ジャーナル化の動きが盛んであるが、現在一般的に用いられているデータ形式では、学術論文、研究報告書等の情報発信・交換・流通には、限界があると言われている。そのような中でXML (eXtensible Markup Language) 技術は、現在一般的に用いられている記述形式に比べて、データの流通・交換、データベース管理といったデータの統合的利用の点で多くの優れた特徴を持つとして注目を集めている。そのXML技術を応用した、学術論文の流通のための総合的モデルシステムの試作開発を行ったのでその内容について報告する。

## キーワード

学術論文、学術雑誌、XML、インターネット、電子ジャーナル、電子出版、情報システム

## An experimental prototype to research the applications of XML technologies in distributing electronic journals

Takahiko SHIMIZU, Kiyohiko UEBA

NTT Learning Systems Corporation

NTT Hamacho bld. 6F 1-11-4

Higashi Nihonbashi Chuo-ku Tokyo 103-0004

Yoshiko SHIROKIZAWA

Japan Science and Technology Corporation

5-3 Yonbancho Chiyoda-ku Tokyo 102-0081

## Abstract

There is a strong trend towards providing electronic journals over the Internet. However, there is a limit in providing, exchanging, and distributing articles and research reports with the currently used data formats.

Meanwhile XML (eXtensible Markup Language) technologies draw attention as the means of utilizing the integrated data in data distribution, data exchange and database management. This report is about the experimental model of integrated electronic journal distribution system that we developed by applying XML technologies.

## Keywords

academic articles, academic journals, XML, Internet, electronic journals, electronic publishing, information system

## 1. はじめに

近年、学術論文を電子データ化し、データベースシステムやネットワークシステムと組み合わせて処理する電子ジャーナルシステムの開発・運用の動きが盛んである。特に、インターネットによる世界的規模のネットワーク構築が実現されつつある今日においては、情報の流通が様々な枠組みを超えて行なわれるようになってきている。現に科学技術振興事業団（JST）においてもそのような機能を実現するものとしてJ-STAGEシステムを開発、1999年10月より稼動を開始している。J-STAGEシステムは論文の投稿から公開に至るプロセスをWWWサーバーとデータベースを中心としたシステムによりオンライン化したもので、各学会のジャーナルを電子化しインターネット上で各種のサービス提供を行なっている。そのような中でドキュメントデータの蓄積、流通、交換を行なうシステムには新たな要求が発生しつつある。例えば、他のネットワークシステムとデータ交換を効率的に行なうことや一度電子化したデータを低コストで多目的に応用するといった機能である。これらの機能は、データ形式に高い互換性が要求され、その処理を行なうための様々なリソースが体系的に用意されている必要がある。今回は、この要件を現時点で最もよく満たしていると思われるXML技術を用いて、電子ジャーナル流通システムに対するその応用性を調査実験するためのモデルシステムを開発した。本システムにおいては、学術論文の執筆から投稿、データベースへの格納、公開、検索といった一連の電子ジャーナル処理機能をXML技術を用いて構築することにより、その利用法、問題点などについて調査を行なった。特にXML技術の特徴である他システムとのデータ交換、単一データの多目的利用と言った技術の調査が重点的に行なえるサービス形態を想定し試作及び運用実験を行なった。その他にも、既存データとの互換性を確保するための試みとして各種コンバータの試作開発や数式データのレンダリング機能実験など周辺機能の調査も行なっている。これら一連の機能により構成されたモデルシステムを先進的XML応用技術調査実験：XML Advanced and Applied Technologies Experiment とし、その頭文字を取り、Xa<sup>2</sup>T（ザート）と命名した。Xa<sup>2</sup>Tシステムは図1のような構成になっている。

仮に学術論文の出版公開および流通の業務形態の移行プロセスを三段階に分けてみると。

第一段階：紙、印刷雑誌を中心としたもの。

第二段階：オンライン化・電子化を行なったもの。独立したサービス提供。

第三段階：ネットワーク環境、標準化データに対応した分散・共有システム。

冒頭で述べたJ-STAGEはこの第二段階にあたるものである。Xa<sup>2</sup>Tシステムはこの第三段階を担うシステムを開発するための調査実験プロジェクトである。現行のシステムの問題点などを考慮し、それを解決するための様々なサービスやツールを開発・提供し、従来の資産を生かしつつ、その目的を果たすために最も適切なシステム形態と応用技術について調査を行なった。

## 2. Xa<sup>2</sup>Tシステムを構成する技術要素

Xa<sup>2</sup>Tシステムは図1に示されるように、論文の蓄積と提供を行なうサーバー Xa<sup>2</sup>T-1を中心に様々な調査実験の要素が組み込まれている。表1に今回の開発の項目と実験の内容を記す。

### 2.1 メタデータ文書定義

メタデータはあるデータの性質を表すデータであり、Xa<sup>2</sup>Tシステムにおいては個々の論文データの属性情報や書誌事項がそれにあたる。メタデータはシステム内で情報を管理することや外部との情報交換をおこなうことを目的として用いられる。たとえば、検索機能などは全文を検索する場合を除いて、ほとんど

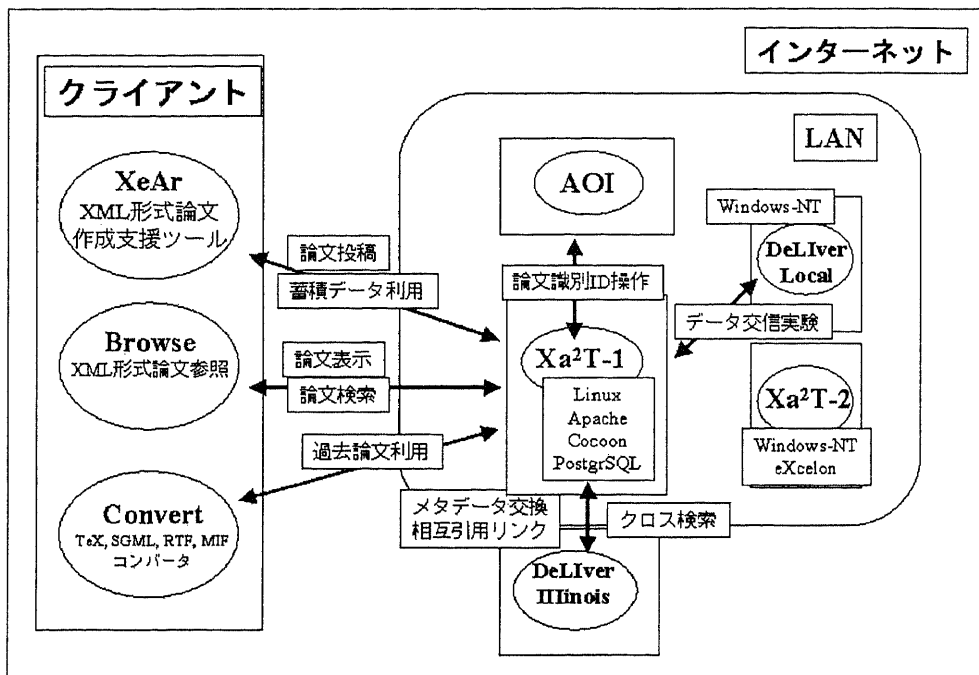


図1 Xa2Tシステム全体構成

はこのメタデータを対象として処理を行なうことで機能を実現することが可能である。特にこのメタデータの様式を外部にも公開し、その仕様に基づいてアクセス可能なインターフェースを提供することにより外部からの検索を可能としたり、それを自動引用リンク生成のために利用することが可能になる。

Xa<sup>2</sup>Tシステムにおけるメタデータ定義は2種類ある。一つは、サーバー間通信の実験を行なうパートナーである米国イリノイ大学の電子図書館システム「DeLiver」の開発チームが設計したメタデータセットを参考にして作成したものである。その開発呼称を「Bronco」とし、その内容は表2に示されるようなエレメントセットで構成されている。「Bronco」は広範囲な素材に対応可能なように数多くの項目を含んでおり、その利用法においても柔軟な対応が取れるように構造定義も特に行なっていない。さらに「Bronco」はドキュメント系メタデータの標準とも言えるダブリンコアを包含しており、それに基づく処理にも対応できる。

もう一つのメタデータ定義は論文検索機能に必要とされる最小限の構成にまとめたエレメントセットで、その開発呼称を「Unicorn」とした。「Unicorn」は表3に示すエレメントを有し、図2に示すような構造を定義付けている。「Unicorn」はXa<sup>2</sup>Tシステム内部での論文管理、情報検索、外部との情報交換という明確な目的に基づいて設計されている。そのため、エレメント構成を必要なもののみを残してスリム化し、その意味付けもより具体的に決めてある。

## 2. 2 文書定義

文書定義はXa<sup>2</sup>Tシステムの根幹となるものである。XML形式のデータを取り扱うシステムにおいて文書定義はデータベースの構造を定義するものであり、情報の流通・交換の処理仕様を決定するものである。

文書定義（メタデータ）	検索機能、自動引用リンク生成機能などで応用される汎用性の高い抽出データのXML文書定義の開発。書誌事項を中心とした情報で構成されており、目的に応じた2種類のテーブルを開発。
文書定義（論文本体）	電子ジャーナル用のXML文書定義を開発。
XML形式論文作成支援ツール開発	上記開発におけるXML文書定義に従った論文データの作成を行なうためのアプリケーションプログラム。
XSL試作開発	XML形式の論文データを表示、変換するためのXSL開発。複数のXSLを開発することにより、一つのXML形式論文を様々な様式で出力が可能になっている。
数式処理機能開発	学術論文における数式処理、特にMathML形式のデータの処理に関する機能の開発。MathML形式データの画像レンダリング機能の開発など。
各種コンバータ開発	TeX、MIF、RTF、SGMLの各データとXMLとの双方向変換アプリケーションプログラムの開発。但し、今回は調査試作ということで限られた範囲のデータを対象として開発作業を行なった。
XML分散データ処理システム開発	XML技術の大きな特徴の一つであるデータ交換機能に関する開発。主にXa2T-1サーバーの開発を行なった。Xa2T-1を通じて外部のサーバーへ横断検索を行ったり、蓄積された論文の引用表記から外部のサーバーの蓄積論文へリンクを自動生成したりする。また、XSLをダイナミックに切り替える仕組みや論文を特定するための新たな仕組みも組み込んだ。

表1 Xa2Tシステムを構成する技術要素

今回開発した文書定義はISO12083XML - DTDを参考に基本構造を設計、Xa<sup>2</sup>Tシステムを構成する各種のツールや処理機能に合わせて構造を調整、そして現行の論文を処理するにあたり、問題が将来にわたっても発生することがないかどうか検証をし改良を行った。文書定義の開発呼称は「WhiteHorse」とした。「WhiteHorse」は約120種類のタグから構成されている。これらのタグは可読性の向上を図るために可能な限り表記をフルスペルを用いるようにしてある。（例：ISO12083では<aff>をしているのを「WhiteHorse」では<affiliation>とする）「WhiteHorse」の第二階層までの概略構造図を図3に示す。「WhiteHorse」では最上位を表すタグである<article>の直下に5つのタググループを有する。

1. <front>

<front>タグはその論文の書誌事項、ヘッダデータが所属する。論文のタイトルや著者に関する情報、アブストラクト、キーワードなどがここに格納される。

2. <textbody>

ここには論文本体が格納される。段落分けなどにも対応。

3. <appendixgrp>

ここには追加情報が格納される。

4. <resource>

文字以外のデータグループ。論文内の画像、表、数式、化学データなどに関する情報を格納する。

5. <back>

謝辞や文献情報を保持するグループ。

これら5つのタグはXML形式で論文を記述する上での基本構造であり、それぞれがさらに細かな情報エレメントによって構成される。

## 2. 3 XML 形式論文作成支援ツール開発

XMLは”<>”で囲まれたタグにより実際のデータを意味付けして用いるいわゆるマークアップ言語の一つである。XMLではそのタグの構成と内容を自由に定義できるのが特徴の一つであるが、Xa<sup>2</sup>Tシステムにおいては前述した「WhiteHorse」がその定義を行なっている。XML形式の論文を記述する場合、基本的にXMLはテキストベースであるのでテキストエディタを用いてタグを挿入しながらXa<sup>2</sup>Tシステムで処理可能なXML形式の論文を記述することも可能ではある。しかし、その場合は文書定義「WhiteHorse」の仕様に精通していなくてはならない。そこで、今回は文書定義などの内部処理仕様を意識すること無くXML形式の論文の作成を行なえるように支援するアプリケーションプログラム、XML形式論文作成支援ツール（XeAr:XML Editor for Article。以下XeArと記す。）を開発した。このツールの画面表示を図4に示す。XeArの画面はコマンド指定を行なう[メニュー部]、文書のエレメント構造を表示する[ツリー部]、項目に従って入力を行なう[エディット部]、編集したデータをブラウザ表示としてチェック表示させる[HTML部]の四つのパートから成っている。

このエディタの基本的な操作方法は、

1. あらかじめ別のソフトで作成した論文草稿をコピーペーストし、[エディット部]の所定の位置にあてはめてゆく。または一括して草稿を読み込みそれを各項目に振り分けて行くという方法によりXML形式論文の作成を進める。
2. 著者名のように複数個のエレメントが存在する項目の追加は、[ツリー部]でマウスを右クリックし、それを行なう。
3. 入力の表示は[メニュー部]のリロードボタンをクリックすることで[HTML部]に表示される。
4. [メニュー部]のファイル保存を選ぶと「WhiteHorse」文書定義に従ったXML形式の論文データが保存される。

論文の作成そのものをまったく執筆開始の段階から行なうことを主流としてとらえず、あらかじめ草稿レベルのデータを他のアプリケーションにより作成させ、最後にこの支援ツールによりXML形式論文を作成するというのがXeArの狙いである。つまり、あくまでも論文にXa<sup>2</sup>T用XML文書定義「WhiteHorse」を適用させることを支援するのが目的であり、XML形式論文の汎用ワードプロセッサではない。

XeArはいくつかの特徴的な機能を有している。例えば、引用文献の末尾リストと本文内の参照位置のリンクを自動生成する機能や、ユーザーが任意に作成した属性タグの挿入、外部参照データ（画像データなど）の直接挿入などである。将来的には執筆から査読審査、修正、出版社など論文の最終的な公開までの作業の流れの中で係る作業すべてに対応可能なアプリケーションに発展させることも検討する。

## 2. 4 X S L 試作開発

XML形式のデータは一つのソースを多目的に利用できることも特徴の一つである。その手法のひとつとしてXSLがある。XSLはXSLプロセッサを通じてXMLソースに働きかけ別のデータフォーマットへ変換したり、表示整形を行なうものである（図5）。このXSLをそれぞれの目的により入れ替えることにより、一つのXMLデータからいくつもの結果を得ることが可能になる。

W3C（World Wide Web Consortium）ではXSLの仕様を、データ変換を行なうXSLT（XSL Transformation）と表示整形を行なうXSL-FO（XML Formatting Object）の2つから構成するとしている。本システムにおいてはその両方の機能を検証するために電子ジャーナルを表示するためのXSLを試作した。今回試作したXSLは以下の3点である。

1. 論文の全体を整形してHTML化してブラウザで表示する。(図6)
2. 論文の抄録情報のみを抜き出しHTML化してブラウザで表示する。
3. 論文の全体を整形してプリンタ印字用にPDF形式で出力する。

このうち、1および2はXMLソースをHTMLに変換するXSLTの技術により処理を行なう。また3はXSLの表示整形機能であるXSL-FOによって処理を行なう。いずれもHTTPサーバーを通じてインターネット経由でクライアントへ電子ジャーナルデータを送信する仕組みの中で動作することを想定して試作をおこなった。この場合、基本的にXSLTとXSL-FOの処理の流れはそれぞれ図7に示すような形となる。

サーバー側の構成は、LINUX コンピュータおよびHTTPサーバープログラムとしてApacheを導入、それに加えてXML関連の様々な処理をおこなうServletであるCocoonとServlet動作に必要なモジュールJDK(JAVA2SDK)およびApacheJServを導入してある。XMLデータはクライアント側からの読み出し要求に従い所定のXSLを呼び出して変換を行ないクライアントへその変換結果を送信する。XSLはCocoonに含まれるXSLプロセッサであるXalanによって処理される。XMLをHTML化して表示する2種類のXSL(論文全体表示および論文抄録表示)はCocoonを通じて動作し変換結果をクライアントに送信する。また、PDF形式の出力に関しては、一度XSL-FOプロセッサによって処理可能な形式のXMLソースを生成し、それをXSL-FOプロセッサがPDF形式のデータに変換しクライアントへ送信する。なお、この仕組みではXSLはダイナミックに選択されることになるが、通常XSLに対する宣言はXML内に静的に記述されていなくてはならない。そのため、今回は静的に宣言されるXSLファイルには渡された引数に従い、指定されたXSLファイルを用いるように再宣言するという動作を構成した。

## 2.5 数式処理機能開発

電子ジャーナルの表示系機能の中で非常に重要な位置を占めるのは数式である。数式は通常の文字列処理と図形画像処理の中間的な要素を持ち、数式情報はそれを含む論文の中で独立した処理を要する。通常XMLデータにはMathMLという数式処理専用の文書定義がある。例えば、図8はMathMLのサンプルリストとそのレンダリング結果である。

このようにMathMLはXa<sup>2</sup>Tシステムの中で全く別の処理体系の文書定義を有している。このため、MathMLで記述された情報が含まれているXML文書を処理するためには、それ専用の処理を付加する必要がある。その手法としては、

1. HTMLおよびCSS(Cascading Style Sheet)、JavaScriptなどを用いてブラウザ上で表示可能なデータを供給する。
2. クライアント側でMathMLをサポートした拡張機能(Plug-INなど)をブラウザに組み込む。(IBM techexplorerなど)

これらの方法の問題点は、複雑な数式処理をCSSやJavaScriptで対応するには限界があることやクライアント側の環境の統一性に裏付けが乏しく拡張機能の導入を前提としたサービスの提供の有効性に疑問があることである。いずれ数式処理はクライアント側で標準的に可能になるものと思われるが、将来に渡っても旧環境への対応を考慮するならばほぼすべてのブラウザが表示を可能としているJPEG画像にMathMLを変換することによりそれは解決できる。今回は、MathMLをJPEG画像に変換する仕組みを調査し、その機能を共有サーバーを通じて共同利用できるオンライン数式画像生成機能を試作した。本機能の構成と処理の手順を図9に示す。

本システムが動作する Xa<sup>2</sup>T-2 サーバーの構成は、Windows NT4.0、Microsoft Internet Information Server、そしていくつかの ActiveX component から成っている。この図に示されるように、MathML をレンダリングしているのは Xa<sup>2</sup>T-2 サーバー側の ActiveX コンポーネントである。MathML のソースが最終的に JPEG 画像となってクライアントに返される手順は以下の通りである。

1. クライアント側のブラウザ画面から MathML の部分のみを入力し、サーバーへ送信する。
2. サーバー側の CGI がそれを受け取り TechExplorer の ActiveX コンポーネントを用いて数式レンダリングを行なう。その結果は仮想的にサーバー側のスクリーンに表示される。
3. サーバー側のスクリーンを同じく ActiveX コンポーネントを用いてメモリに取り込みそれをファイルに記録する。
4. CGI プログラムは最終的にそのファイルへの表示リンクをした HTML ファイルを生成しクライアント側に返す。

以上のような行程により MathML はレンダリングされた数式 JPEG 画像として出力される。

## 2. 6 各種コンバータ開発

新たなシステムの構築と運営を目指すとき、重要になってくるのは既存システムの資産引継ぎである。現行の文書データシステムの用いているデータ形式については、TeX、MIF、RTF、SGML の 4 つの形式を対象にそれらのデータと XML 形式データとの双方向コンバータアプリケーションを試作した。これらの形式のデータの特徴はそれぞれ以下の通りである。

1. TeX : TeX とは、スタンフォード大学の Donald E. Knuth 氏によって作成された文書組版ソフトウェア体系である。テキストベースのデータで構成され、その内容はドキュメントの実データと体裁情報が含まれている。論文の執筆者には古くから愛用され現存のデータの量も抜きんで多い。
2. MIF : **M**aker **I**nterchange **F**ormat の略。Adobe 社の製品 FrameMaker+SGML などを用いられる文書の保存形式。独自のスタイルではあるが大部分をテキストデータ形式で記述し、基本的に実データとそれを表示する際の体裁情報によって構成する。
3. RTF : **R**ich **T**ext **F**ormat の略。Microsoft 社が提唱した文書ファイルのフォーマット。MIF と同じく実データと体裁情報から成るファイル形式。テンプレートと呼ばれる体裁情報と意味付けの定義を対応させる仕様を用いることにより単に体裁情報だけではなく固有の意味付けをデータに与えることが可能である。
4. SGML : **S**tandard **G**eneralized **M**arkup **L**anguage の略。マークアップ言語のひとつで、汎用マークアップ言語規約ともいう。ISO8879 として規格化されている。XML は SGML の文書定義の曖昧さを排除しオンライン流通の仕様を加えたものであり、ほぼ兄弟関係にあるといってもよい。

XML データと各種フォーマットデータとの変換アプリケーションプログラムの開発にあたって、最も問題となるのは、SGML を除く 3 つのデータフォーマットが、表示・印字を主目的とした体裁情報中心の仕様であるということである。XML はその特徴として、データ本体から体裁情報を分離することにより

データの性質に基づいた処理を可能とし必要に応じた体裁処理を随時あてはめることが可能となった。このようなXML形式のデータとTeX、MIF、RTFの各データように体裁情報を中心に扱うデータとの変換を行なうのは非常に困難である。無論、体裁情報からデータの性質を推測することも不可能ではない、例えば、「一番最初に出てくる24ポイント、ボールド表記の行を論文タイトルとする。」というような条件設定に基づいて変換を行なうというような手法が無いわけではない。しかしこの手法ではそのような条件が元来データの性質を決めることを目的として設定されている保証はなく、同じ体裁条件が別の意味で用いられている可能性は否定できない。RTFのテンプレート仕様のように、中には各体裁情報に名前をつけることで属性情報を持たせることが可能なものもある。しかし、残念ながら既存データでそれらが有効に活用されている例はさほど多くはない。このような観点から今回開発したアプリケーションプログラムの最終的な形態を下記にまとめる。尚、すべてのプログラムに必要な条件としては以下の3点である。

- ・ コマンドラインで入力し動作させる方式を取り、GUIなどを持たせないこと。  
これにより単一アプリケーションとして実行させるだけでなく、CGIなどからも呼び出すことができる。
- ・ XMLデータは本プロジェクトで策定した文書定義「WhiteHorse」に従ったデータを対象とする。
- ・ 開発言語は互換性を考慮してPerl言語を用いるのが望ましいがそれ以外の言語を用いる場合は、Windows環境での動作を保証する。

#### 1. TeXコンバータ

コンバータアプリケーションの仕様範囲は、約5000件の論文サンプルの中から無作為に抽出した40論文を対象に完全な変換が双方向で可能であることとした。吸収できない仕様の情報が紛れていた場合はコメントアウトし無効情報とする。但し、何らかの形で再利用する可能性もあるので、削除はしない。TeXは一定の表示出力に対する記述が何通りもありえる。そのため一度独立した中間フォーマットであるDVI (DeVice Independent) ファイルに変換し、それを変換の対象とすれば少なくとも記述を一本化することが可能である。本件は実験終盤に明らかになったので実際には適用していない。

#### 2. MIF・RTFコンバータ

基本的にこの2つのデータ形式の場合は、仮想的な学会誌の投稿規程に従ったテンプレートを作成、それに則り記述された論文を変換の対象とした。これらの情報には画像データなども含まれている。その場合は切り出して別のファイルとして保管する。

#### 3. SGMLコンバータ

SGMLはXMLとの親和性が高く、変換の難易度はそれほど高くない。今回はXML側は「WhiteHorse」に従ったものとなっているが、SGMLコンバータでは変換の中心にXSLTを用いるため、XSLTを各種用意することにより様々な形式のSGMLを入力して変換することが可能になる。



## 2. 7 XML分散データ処理システム

XML分散データ処理システムは、XMLの持つ先進的な特徴の一つであるメタデータ交換技術を用いたものである。今回はその技術に基づいたサービスとして、他のネットワークシステムにあるデータベースに対する相互の検索機能および引用リンク自動生成機能を実現した。また、引用リンク生成については引用された側の論文に引用した側へのリンクを生成する逆引用リンクという新たな機能も組み込まれている。これらの機能はこれまで述べてきた Xa<sup>2</sup>T システムの中核をなすサーバー Xa<sup>2</sup>T-1 に搭載した。

Xa<sup>2</sup>T-1 は投稿を受付、それを蓄積し公開する。Xa<sup>2</sup>T-1 サーバーを通じてクライアントは Xa<sup>2</sup>T-1 内に蓄積された論文を検索、表示させることができる。さらに、外部のネットワークにあるサーバーを検索することも可能である。今回はXMLによるドキュメントデータベースの研究プロジェクトのパートナーである米国イリノイ大学グレンジャー技術図書館の研究チームが開発中の DeLIver サーバーに対して同様に検索をかけることを可能とした。

また、引用リンクの自動生成の場合は、投稿から蓄積まで以下の手順で行なう。(図10)

1. XML形式論文作成支援ツール (X e A r) から作成された論文が投稿される。  
このとき、投稿された論文からは書誌事項であるメタデータが抽出されデータベース (PostgreSQL) に格納される。実際の論文データはXMLファイルのまま保存される。
2. XMLファイルから引用文献情報を抽出する。
3. 米イリノイ大学 DeLIver システムの検索用クエリとして抽出データを整形する。
4. クエリを送信する。
5. 見つかった場合はリンクを張るための情報を送信してもらう。
6. 受信した情報を論文に格納。
7. 引用したことを DeLIver に通知。
8. DeLIver からメタデータの要求を受信。
9. リンクを張った側の論文のメタデータを送信。

また、DeLIver 側から引用のための問い合わせがあった場合にも双方の受動の立場が入れ替わるのみで同様の動作を行なう。

このように外部サーバとのメタデータの交換により、相互のリソースを効率的に、かつ自動的に関連付けることが可能である。

## 3. 成果並びに問題点と今後の取り組み

近年における電子ジャーナルシステムへの取り組みは冒頭で述べたとおり、多数の学会、出版社で既に開発・運用が盛んになりつつある。明らかに従来までの紙印刷主体の業務体系から電子データ、ネットワーク流通のシステムへの移行が始まっているといえる。無論これは従来の紙印刷の存在を否定するものではない。印刷物による情報の提供には低価格で安定した情報の確保が可能であるという特徴がある。しかし、オンラインシステムには高速で大量の情報を処理できるという利点がある。そのような状況にあって、現在の情報のオンライン化、ネットワーク上での分散および共有処理を組み込んだシステムの開発を考慮した場合、データの互換性・柔軟性、それを処理するための大量の利用可能リソースの存在、といった要素が重要になってくる。XMLのシステム体系はまさにこの条件に適合していると言える。たしかに、XMLには多くのものを犠牲にしている面もある。例えばテキストベースのデータであるために処理データ量が肥大する傾向にある。情報の定義をシステム開発レベルに開放していることにより処理系が複雑になり

開発コストがかさむ。と言うようなことである。しかし、現時点で他の選択肢はあまり多くない。今のところXMLは完全ではないがXML以上のものが存在しているとも言い難い。さらに、今後ハードウェアやネットワークの機能、および処理能力は向上しコストも下がって行くと予測される。XMLの処理効率に関する問題は時間と共に解決される可能性が高い。

今回の調査およびモデルシステムの開発と運用についてXMLを主たるテーマとして選択した理由はここに述べるようにジャーナルデータの統合化された流通システムを考慮した場合にそれに最も適合した技術だったからである。X a 2 Tシステムは今回の試作において特に基礎的な部分でのモデル化を図った。たとえば、投稿の手順は執筆してそれをサーバーにアップロードした瞬間から公開されるようになっているが現実にはそのようなことは有り得ず、査読や審査などのいくつかのプロセスを経て公開される。今後はそのような業務フローも十分に調査し、XML技術のさらに高度な応用技術も導入して行くことを考慮したい。冒頭で述べた電子ジャーナルシステム「J-STAGE」は既に多数の学会誌の格納を果たし、安定したサービスを供給している。しかしながらその将来性について考察すると外部のシステムとの情報交換や効率のよい業務フローの反映といったことが低コスト・短期稼働で実現が可能かどうかという面で、限界点がさして速くない将来に訪れる可能性が高いと言わざるを得ない。今回行われたX a 2 Tシステムの開発と運用実験で得られた技術は今後「J-STAGE」システムにも積極的に応用して行く予定である。

## 参考文献

- [1] 科学技術情報発信・流通総合システム J-STAGE. <http://www.jstage.jst.go.jp>
- [2] イリノイ大学の電子図書館システム DeLIver. <http://dli.grainger.uiuc.edu/deliver/about.htm>
- [3] University of Illinois Digital Library Initiative (IDLI) Metadata Schema Description. <http://dli.grainger.uiuc.edu/iso11179/IDLIISO11179.xml>
- [4] The World Wide Web Consortium(W3C). <http://www.w3.org>
- [5] ISO 12083 Information from XMLxperts Web site. <http://www.xmlxperts.com/12083.htm>
- [6] Dublin Core 参照記述. <http://www.DL.ulis.ac.jp/DC/>
- [7] Dublin Core Metadata Initiative. <http://purl.org/DC/>
- [8] 小原満穂「米国の電子ジャーナル事情」、情報管理、42(11), 942-950, 2000. <http://johokanri.jst.go.jp/e-journal/data/v42/n11/a04/index.htm>
- [9] 時実象一「電子ジャーナルの現状と動向」、情報管理、43(5), 391-410, 2000. <http://johokanri.jst.go.jp/e-journal/data/v43/n05/a02/index.htm>
- [10] 時実象一「学術系電子雑誌の現状」、情報管理、41(5), 343-354, 1998. <http://johokanri.jst.go.jp/e-journal/data/v41/n05/a02/index.htm>
- [11] 白木澤佳子「学術情報のためのXML ツールの開発」情報科学技術研究集会, 37, 7-12, 2000.
- [12] 吉田幸二「J-STAGE:『科学技術情報発信・流通総合システム』電子ジャーナル作成とインターネットによる流通」デジタル図書館. No.16. [http://www.dl.ulis.ac.jp/DLjournal/No\\_16/7-k3yoshid/7-k3yoshid.html](http://www.dl.ulis.ac.jp/DLjournal/No_16/7-k3yoshid/7-k3yoshid.html)
- [13] 杉本重雄「メタデータについて-Dublin Core を中心として-」情報の科学と技術 49(1), 3-10, 1999.
- [14] ハーントク, デレク J 「電子出版で百倍になる可能性-5年後にはもう冊子体なし(?) 学術出版の世界-」情報管理, 42(10), 887-891, 2000. <http://johokanri.jst.go.jp/e-journal/data/v42/n10/b09/index.htm>

メタデータエレメント名	意味	ダブリンコアとの関連
Abstract	アイテムの要約テキスト。	Description
Alternate	論文の代替フォームへのレファレンス。	Relation
author_info	個人著者名、所属団体、組織著者名。	Creator, external
author_name	個人著者名、一般に姓、名。author_infoに含まれる	
Occ	アイテムの著作権許可コード。	Rights
Controlled_term	アイテムを記述する統制サブジェクト言語。	Subject
Copyright	論文の著作権文書。	Rights
Date	オブジェクトに関する重要な日付。	Date
External	論文に関連する外部ドキュメントやオブジェクトへのレファレンス。	Relation
first_page	紙のソースの論文の最初のページ。Publicationに含まれる	
Identifier	公式識別番号またはコード。論文やジャーナルの識別に使われる。DOI, PMID, CODEN, ISSNなど。	Identifier, publication, external
Issue	論文が掲載されているジャーナルのイシューナンバー。Publicationに含まれる。	
item_title	論文中の論文タイトル。	Title
Language	オブジェクトの言語を識別するRFC1766によるコード。ENIは英語など。	Language
last_update	メタデータレコードの最終更新日。Preparationに含まれる	
Link	関連実体のURL。External, alternate, partに含まれる	
Mime	アイテムのmimeタイプ。	format
organization_name	論文の著者組織名や出版者名を含む。	Publisher, author_info
Pageination	ソースに現れるアイテムの総ページ数。publicationに含まれる	
part	論文の一部になっているスタンドアロンオブジェクトへのレファレンス。	Relation
Place	論文の適用範囲を記述する地理的名称。	Coverage
Preparation	このメタデータレコード準備に関する情報。作成準備者の名前、email、最終更新日	
preparer_email	このメタデータレコードの最終更新の責任者のemailアドレス。Preparationに含まれる	
preparer_name	このメタデータレコードの最終更新の責任者の名前。Preparationに含まれる	
Publication	論文のソースである出版物に関する情報。ジャーナル名、ボリューム、最初のページ等を含んでもよい。	
publication_date	関連アイテムの出版日。Publicationに含まれる	
publication_title	出版物のタイトル。ジャーナルタイトルなど。Publicationに含まれる	
publication_title_abbreviation	出版物のタイトルの一般的省略形。'Journal of Applied Physics'が'J.Appl.Phys'など。publicationに含まれる	
Size	論文のバイト数によるファイルサイズ。	format
Subtype	オブジェクトのタイプを十分に記述するためにタイプと共に使用してもよい。	type
Time	論文の一時的適用を記述する期間。	coverage
Title	論文の完全表示タイトル、関連論文の記述、論文の一部。引用にも使用可。著者、ジャーナルタイトル、ページ、図表に使われたテキストを含んでもよい。externalやpartに含まれる。	
Type	オブジェクトのタイプ。画像やテキストなど。	type
Volume	論文が出版されている出版物のボリュームナンバーやボリューム識別要素。Publicationに含まれる	
volume_title	論文が出版されているシリーズの中の各ボリュームの公式タイトル。	

表2 メタデータ「Bronco」の元素表

メタデータエレメント名	意 味
aoi_record	新規論文のメタデータレコードルート。
timestamp	レコード生成およびモディファイ時の時刻記録。
aoi_data	aoi情報セット
aoi	aoi番号
url	aoi番号に対応するリソースのロケーション。(URL)
journal_article_metadata	メタデータセット
article	論文識別情報。
title	論文タイトル。
author	論文著者。すべての著者を列記し、姓名間は空白、各著者名間はカンマで区切る。
given_name	著者名前(姓以外すべて)。
surname	著者姓
date	日付
month	出版公開月
day	出版公開日。
year	出版公開年
enumeration	論文目録情報
volume	論文ボリューム番号
issue	論文イシュー番号
first_page	第一ページ番号(主に論文開始ページ番号)
last_page	第二ページ番号(主に終了ページ番号)
article_number	論文番号(出版社指定の場合)
special_numbering	論文番号(補遺、特別号などの場合)
identifier	各種識別子。aoi以外の論文識別子(例: PII、DOIなど)
journal	ジャーナル識別情報
full_title	ジャーナル正式名
abbrev_title	略式ジャーナル名
issn	ISSN番号
coden	CODENコード

表3 メタデータ「Unicorn」のエレメント表

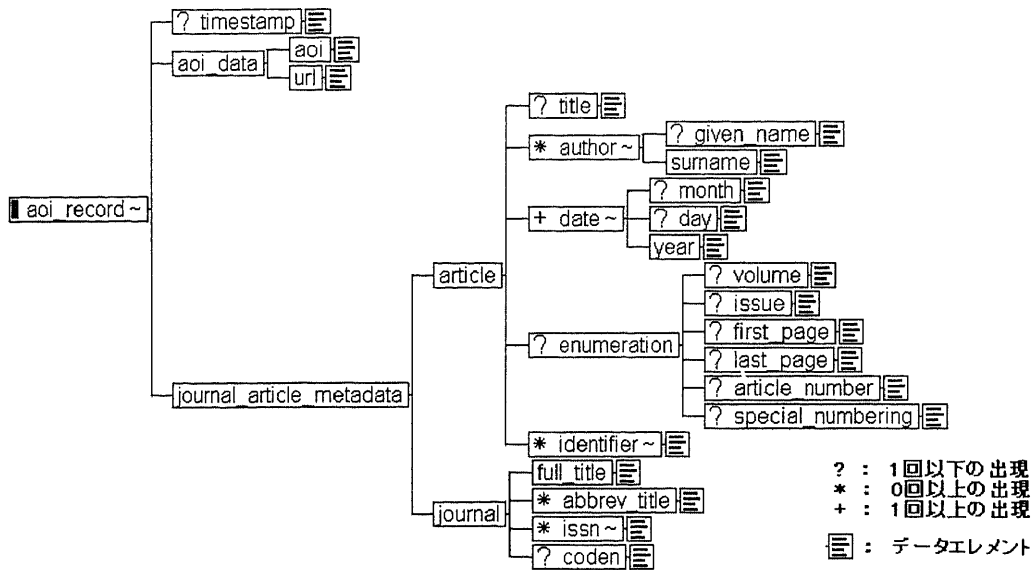


図2メタデータ「Unicorn」の構造概念図

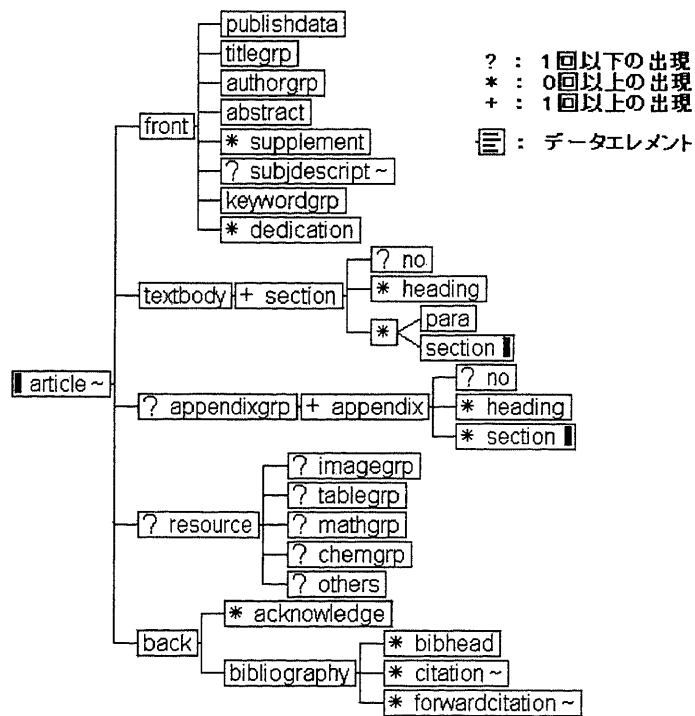


図3 XML形式論文文書定義「WhiteHorse」概念図

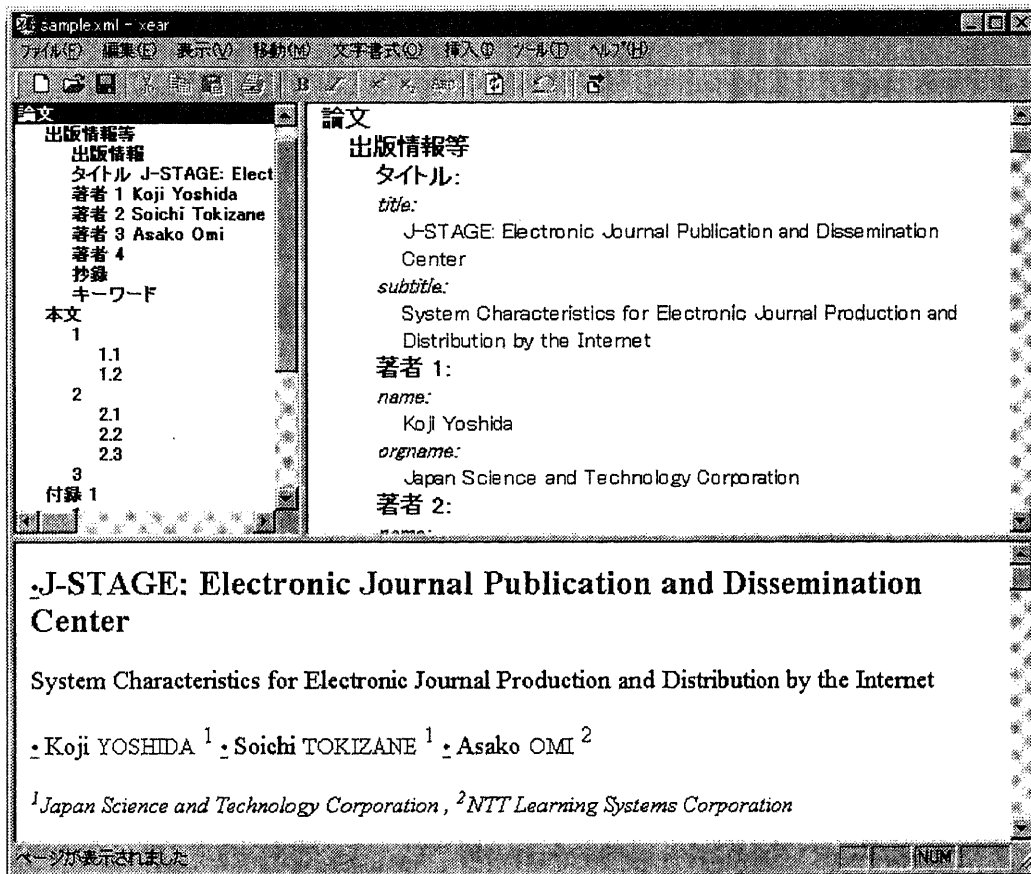


図4 XML形式論文作成支援システム 動作画面

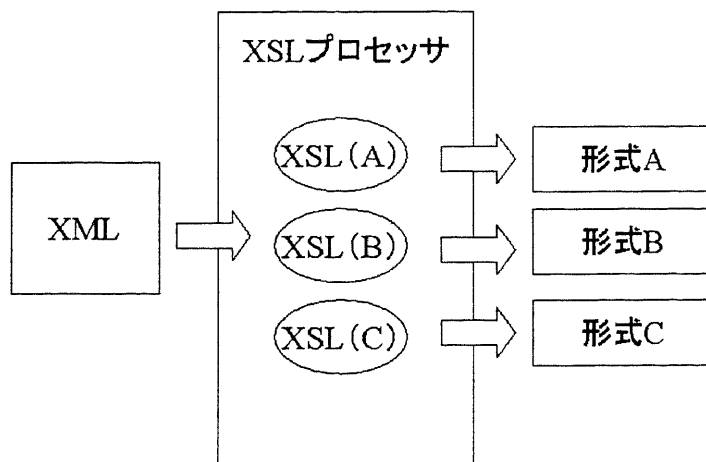


図5 XSLの動作概念図

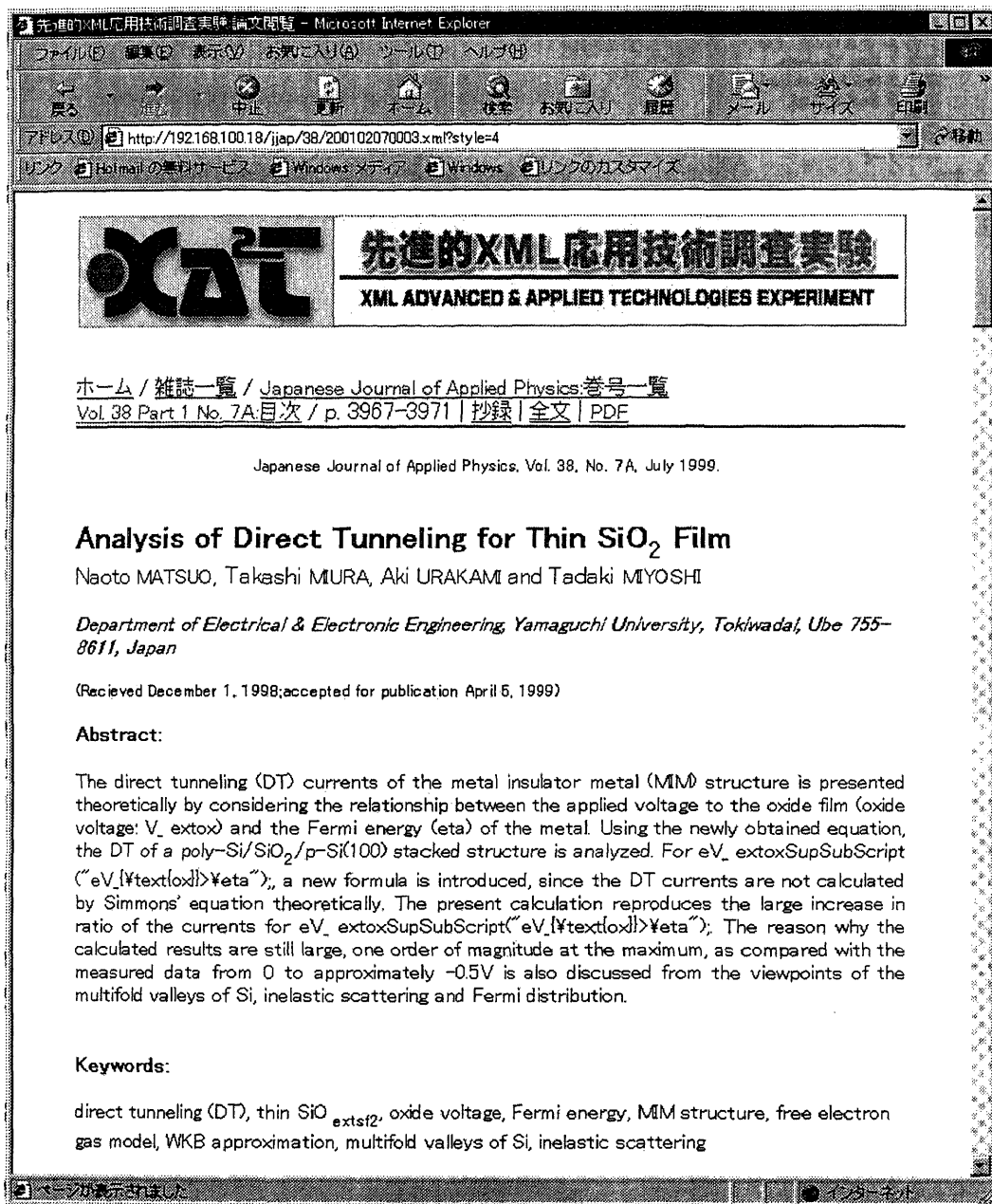


図6 論文表示の例

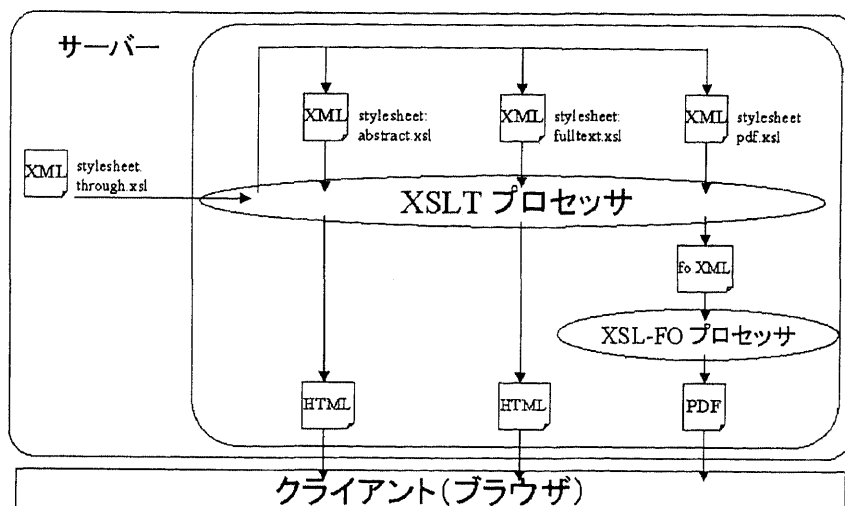


図7 Xa2TシステムにおけるXSL処理体系

```

<math>
  <mi>E</mi>
  <no>-</no>
  <mi>n</mi>
  <msup>
    <mi>c</mi>
    <mn>2</mn>
  </msup>
</math>

```

$$E=mc^2$$

図8 MathMLのサンプルリスト(左)およびレンダリング結果(右)



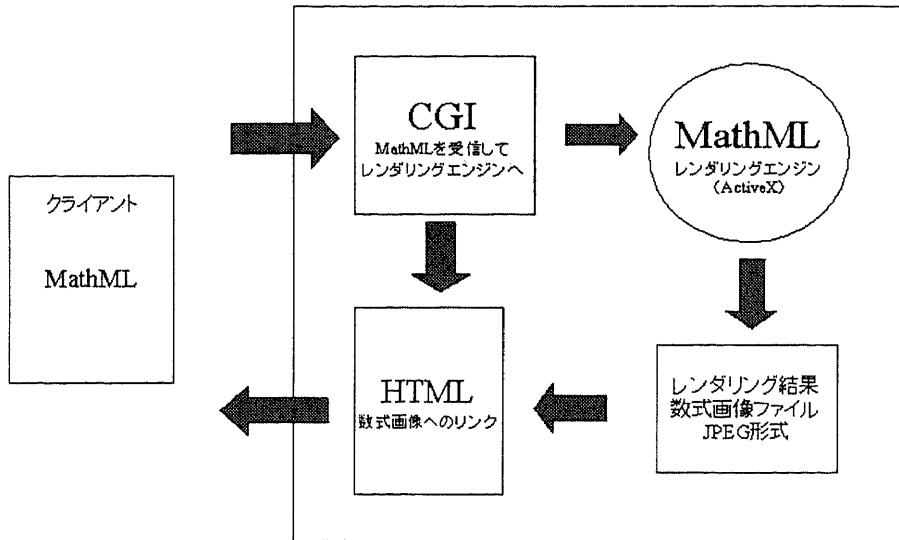


図9 オンライン数式画像生成機能の構成と処理手順

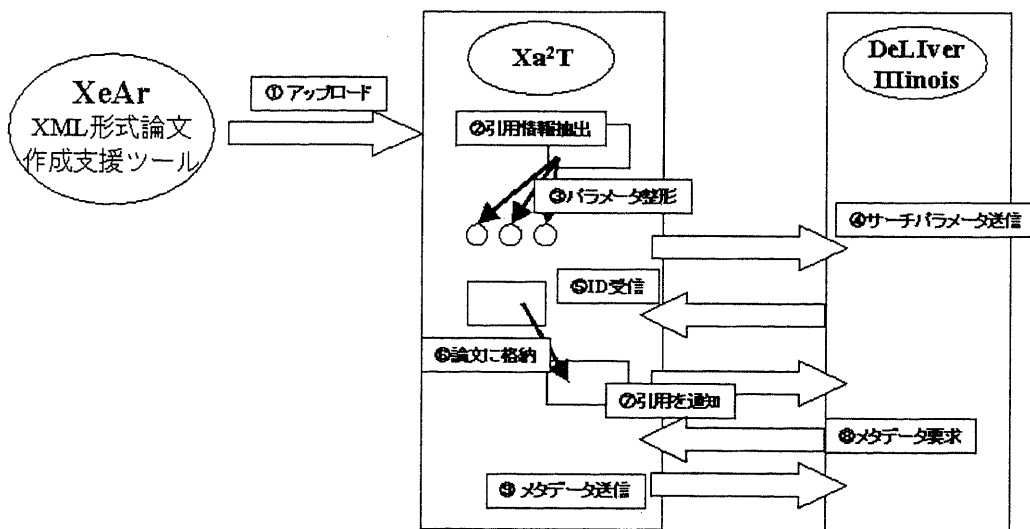


図10 Xa2T-1サーバーにおける相互リンク作成の手順