クライアントマシン上でのWWW情報統合システム

北村泰彦　野田知哉　辰巳昭治
大阪市立大学工学部情報工学科
E-Mail: kitamura@info.eng.osaka-cu.ac.jp

Abstract

分散して存在するWWW情報源から情報を抽出しクライアントマシン上で統合するWWW情報統合システムの構想について述べる．システムはWWW情報源から情報抽出を行う仮想WWWオブジェクト，情報統合を行う統合オブジェクト，ファイルアクセスなどを行う入出力オブジェクトから構成される．利用者はGUI上でこれらを組み合わせたり，それぞれの入力パラメータを変化させることで柔軟な情報統合が可能になる．

1　はじめに

近年インターネットは急速に社会に浸透し，今やわれわれの日常生活を支えるために不可欠なインフラストラクチャの一つとなりつつある．インターネットが提供するサービスの中でもWWW (World Wide Web)は最も人気の高いサービスの一つであり，学術研究，ビジネス取引，コミュニティ形成などを支援する様々な情報源がインターネット上に提供されている．これまでWWW情報源はブラウザを介してアクセスされ，それぞれ単独で利用されることが一般的であったが，これらの情報源を組み合わせることができれば，以下の事例のように，その付加価値をさらに高めることが可能になる．

ワシントン大学のOren Etzioniによるインターネットソフトボットの研究はインターネット上にある既存の情報源を統合することにより，より高度な情報提供を可能にしている．例えば，MetaCrawler[1]は複数の汎用検索エンジンの検索結果を統合することで，検索結果の質を向上させている[12]．Ahoy![2] はMetaCrawlerの検索結果を電子メールデータベースから得られたアドレスをもとにフィルタリングするなどして個人のホームページを検索する専門検索エンジンである[13]．またShopBotはインターネット上にある複数のバーチャルショップから価格情報を抽出し，比較ショッピングを可能にするサービスである[1].

また学術研究の分野においてもWWW情報源の統合に関する要求が高まっている．例えばヒトゲノム計画においては現在，DNAや蛋白質の配列，三次元構造，機能，遺伝病，文献などに関するさまざまなデータベースがWWW上で公開されている．これらのデータベースは単体利用よりも相互利用できればより有用であり，DBGET[3][4] やSRS[4][2] などではハイパーリンクを用いてデータの相互参照を可能にしている．

しかしながら以上のWWW情報統合は専用サーバ上でなされ，情報統合の仕様は設計者により決定された固定的なものである．したがって利用者は統合された情報を通常の WWWブラウザでアクセスするだけであり，様々なWWW情報源を自由に組み合わせたり，カスタマイズすることができないという制限がある．これに対して，ヒトゲノム計画などの分野では様々なデータベースに格納されたデータを，研究者自身が自由に組み合わせて新しい知見を得ることが今後更に要求されるようになる．そこで本稿ではクライアントマシン上で利用者が柔軟にWWW情報源を統合したりカスタマイズできるようなシステムの構想を提案する．

[1] http://www.metacrawler.com/
[2] http://ahoy.cs.washington.edu:6060/
[3] http://www.genome.ad.jp/dbget/dbget.html
[4] http://www.embl-heidelberg.de/srs/srsc

2 仮想WWWオブジェクトに基づくWWW情報統合

WWW情報源は一般的に$HTML \leftarrow f(URL, INPUT)$という関数で表すことができる．ここで$URL$は情報が存在する場所を表すURL (Universal Resource Locator)，$INPUT$はFORM形式のWWWページで用いられる入力パラメータリストである．これらの入力パラメータはGETあるいはPOSTメソッドによりWWWサーバに送られる[5]．FORM形式でないWWWページに関しては$INPUT$の値は空である．$HTML$はURLと$INPUT$により得られるHTML(Hyper Text Markup Language)文書である．例えば検索エンジンではURLは検索エンジンのURL，$INPUT$は検索キーワード，$HTML$は検索結果を示すHTML文書である．

WWW情報源はインターネット上に分散して存在しており，通常はブラウザを介して利用者はURLと$INPUT$を入力し，その出力となる$HTML$をブラウザに表示して，情報を得るものである．本稿で対象とするWWW情報統合は，複数のWWW情報源からの出力($HTML$)を合成したり，あるWWW情報源からの出力($HTML$の一部)を他のWWW情報源の入力とすることを目的とする．これは従来の検索エンジンがキーワードにより関連付けられたページ(URL)レベルの情報統合を行っていると見なせるのに対して，本研究における情報統合はページレベルよりも粒度の細かいパラグラフあるいは単語レベルの統合を目的としている．

しかしながらこのようなWWW情報統合を実現するためには以下の課題を解決する必要がある．

(P1) 自立分散的な情報源．WWW情報源はインターネット上に分散して存在し，それぞれのサーバ内で独自に管理運用されている．それらはHTTPを介したブラウザからの1対1のアクセスしか想定しておらず，複数のWWW情報源を統合するためには共通の統合インタフェースを用意する必要がある．

(P2) 情報の混在．WWWページは情報提供者がHTMLを用いて自由に記述することができ，情報利用者が必要としないものも含めて様々な情報が混在する場合がある．例えば新聞社のWWWページには新聞記事だけではなく，インデックスや広告なども含まれる．したがって効果的な情報統合を行うためにはWWWページから利用者が必要する情報だけを抽出する機能が必要である．

(P3) 準構造情報．WWWページを記述するHTMLは視覚的な構造を表すことはできるが，意味的な構造を表すことができない．WWW情報統合において関連する情報を集めることを可能にするためには，HTML文書を意味情報が付加された構造情報に変換する必要がある．

(P4) 情報統合．複数のWWW情報源をパラグラフレベルで柔軟に組み合わせる手段が必要になる．

(P5) 利用者インタフェース．計算機の非専門家でも情報統合を容易に行なえる利用者インタフェースが必要になる．

(P6) スケーラビリティ．インターネット上には膨大な数のWWW情報源が存在する．これらの情報源を統合するためには分散している多数の開発者が共同して情報統合を実現できるような仕組みが必要になる．

(P7) 動的な情報源．WWW情報源はその内容や構造が頻繁に変更される可能性があり，その対処が必要になる．

以上の問題に対して本研究の示す解決策は以下のとおりである．

(A1) インターネット上に分散するWWW情報資源をクライアント上で操作可能なように仮想WWWオブジェクト化する．

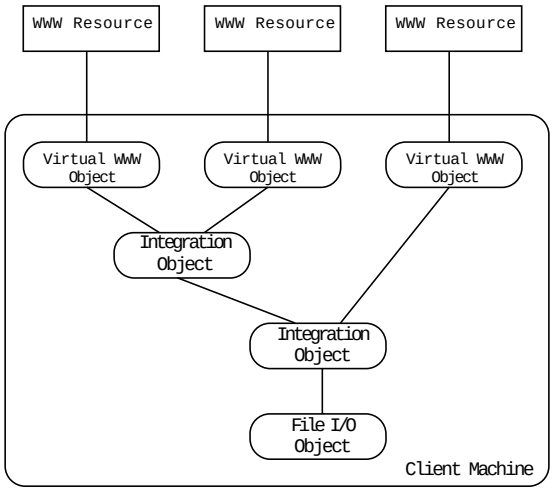

図 1: 仮想WWWオブジェクトによる情報統合

(A2,A3) HTML文書を分解し，意味付けされた構造化情報に変換する．この記述言語としてはXMLを採用する．

(A4) 仮想WWWオブジェクト間のメッセージ交換を可能にすることでその統合を実現する．さらに高度な統合を実現するためにメッセージ変換，フィルタリング，ファイル入出力などを行う特殊オブジェクトを用意する．

(A5) オブジェクトの統合を行なうGUI環境を提供する．

(A6,A7) オブジェクトは利用者／開発者間で共有可能とすることにより，スケーラビリティや動的な情報源の問題に対処する．

本稿では特に (A1) から (A4) の問題に対処する試みとして，図1に示すようなWWW情報統合システムを提案する．インターネット上に存在する個別のWWW情報源 (WWW Resource) はクライアントマシン上で抽象化された仮想WWWオブジェクト (Virtual WWW Object) として扱われる．仮想WWWオブジェクトはHTTPを介してWWWサーバにアクセスを行ない，HTML文書を得る．そして得られたHTML文書を分解して，情報の構造化，意味情報の付加を行なう．この記述言語としてはXML[10]を採用する．情報統合は仮想WWWオブジェクト間でメッセージ通信を可能にすることにより実現される．さらに高度な情報統合を実現するためにファイル入出力やデータ操作を行なう特殊オブジェクトを用意する．ファイル入出力オブジェクト (File I/O Object) は情報統合の結果をファイルに保存するオブジェクトであり，統合オブジェクト (Integration Object) はメッセージ変換やフィルタリングなどのデータ操作を行うオブジェクトである．

3 仮想WWWオブジェクト

仮想WWWオブジェクトはインターネット上に存在するWWW情報源を抽象化したもので，クライアントマシン上で他のオブジェクトとのインタラクションを行なうインタフェースを提供する．仮想WWWオブジェクトは入力パラメータを受け取ると，WWW情報源に対してHTTPを介してアクセスし，得られたHTML文書を基に意味タグを付加した構造化情報を出力とする．

```
 <HTML>
<HEAD><TITLE>検索結果</TITLE></HEAD>
<BODY >
<!-- 広告1 == -->
<A HREF="http://www.ad1.com"><IMG SRC="ad1.gif" ></A>
<!-- 検索ヒット数 -->
<P>検索結果10

<!-- 第1項目 -->
<TABLE>
<!-- 項目番号とハイパーリンク -->
<TR><TD>1</TD><TD><A HREF="http://www.osaka-cu.ac.jp/">大阪市立大学のホームページ</A></TD></TR>
<!-- スコアとURL -->
<TR><TD>99%</TD><TD>http://www.osaka-cu.ac.jp/</TD></TR>
<!-- 文書サイズと最終更新日 -->
<TR><TD></TD><TD>3549 bytes, 1998/04/19</TD></TR>
<!-- 内容記述 -->
<TR><TD></TD><TD>2月2日より大学の郵便番号が 558-8585(杉本キャンパス), 545-8585(阿倍野キャンパス)に変
わりました. 大阪市立大学学長からのメッセージ 大阪市立大学の場所 大阪市立大学の組織 大阪市立大学の歴史 学
内の公式....</TD></TR>
</TABLE>

<!-- 第2項目以降が続く -->

<!-- 広告2 -->
<A HREF="http://www.ad2.com"><IMG SRC="ad2.gif" ></A>
</BODY>
</HTML>
```

図 2: HTML 文書の例

例えば検索エンジンの典型的な出力例は図2のようなものである.

ここでは検索ヒット数, ヒットしたHTML 文書の項目番号, ハイパーリンク, スコア, URL, 文書サイズ, 最終更新日, 内容記述などがHTML を用いて記述されている. また検索結果以外の情報として広告情報も付加されている. 仮想WWW オブジェクトはこのようなHTML 文書から (広告などの) 不要なデータを除去し, 必要なデータを抽出して構造化し, 意味タグを付加する. その結果は図3のようになる.

ここではデータの意味を表すタグとしてhits (検索ヒット数), item (検索項目), title (タイトル), score (スコア), url, size (文書サイズ), date (最終更新日), description (内容記述) が導入されている. この意味付けされた構造情報全体をメッセージと呼びmessage タグで囲っている. メッセージは後述する統合ネットワークにおいてオブジェクト間で情報交換される.

さて, HTML 文書からメッセージへの変換は, スクリプト [8] やプログラミング言語を用いてハードコーディングする方法がもっとも直接的であるが, プログラミングが煩雑になったり, 情報源の更新に対して頑健でないという問題点がある.. その対処としては, テンプレートを用いて自動的に情報抽出するような試み [7] も考えられる. またWWW データに対してどのような意味タグを付加するか, あるいはその一般化は重要な研究課題 [6] であるが, ここではそのような問題には立ち入らず, 仮想WWW オブジェクトの設計者が自由に設定できるものとする.

```
<message>

<hits>10</hits>

<item>
<title>大阪市立大学のホームページ</title>
<score>99%</score>
<url>http://www.osaka-cu.ac.jp/</url>
<size>3549</size>
<date>1998/04/19</date>
<description>2月2日より大学の郵便番号が 558-8585(杉本キャンパス), 545-8585(阿倍野キャンパス)に変わりました. 大阪市立大学学長からのメッセージ 大阪市立大学の場所 大阪市立大学の組織 大阪市立大学の歴史 学内の公式....</description>
</item>

<!-- 第2項目以降が続く -->

</message>
```

図 3: メッセージの例

4 統合ネットワーク

仮想WWWオブジェクトはURLや入力パラメータを入力として，メッセージと呼ばれる意味タグ付き構造情報を出力とする．WWW情報統合は仮想WWWオブジェクトや後述する特殊オブジェクト間でメッセージを介した情報交換を可能とすることにより実現される．オブジェクト間のメッセージ交換は形式的には

$$\text{in@x} = \text{out@y}$$

として記述される．ここでx,yは仮想WWWオブジェクトであり，in@xは仮想WWWオブジェクトxの入力パラメータを指定し，out@yはyの出力(の一部)を指定する．仮想WWWオブジェクトの出力は入れ子構造を許すXML記述であるので，その要素を特定するためにa.bというような記法を用いる．これはデータオブジェクトbの内部オブジェクトaを指定している．データオブジェクトが複数存在する場合はその値はリストとなる．例えば，図3において検索項目のURLを指定する場合にはurl.item.messageとなる．

仮想WWWオブジェクトの入力がリストである場合には，そのそれぞれの要素に対して処理を行い，出力はそれぞれの結果をマージしたものとなる．例えば検索エンジンに該当する仮想WWWオブジェクトsearchはある仮想WWWオブジェクトobjectから入力keywordとしてキーワード列を受け付け，その出力はHTML文書であるとする．すなわち

$$\text{keyword@search} = \text{keywords@object}$$

とする．ここで，入力として

```
<message>
<keywords>abc def</keywords>
<keywords>pqr xyz</keywords>
</message>
```

を与えると，その出力は

```
<message>
<result>
<keywords>abc def</keywords>
<html>abc defの検索結果を示すHTML文書</html>
</result>
<result>
<keywords>pqr xyz</keywords>
<html>pqr xyzの検索結果を示すHTML文書</html>
</result>
</message>
```

となる．

5 特殊オブジェクト

高度な情報統合を行うためにはファイル入出力やメッセージの操作を行なう特殊オブジェクトが必要となる．代表的な特殊オブジェクトとしてはファイル入出力オブジェクトとデータ操作オブジェクトがあり，それぞれについて述べる．

ファイル入力オブジェクト：ファイル入力オブジェクトはローカルファイルの内容をメッセージに変換するオブジェクトである．HTTPはローカルファイルへのアクセスをサポートしているので，ファイルの内容がHTMLで記述されていないことを除いては，通常の仮想WWWオブジェクトと同様に開発できる．

ファイル出力オブジェクト：ファイル出力オブジェクトは情報統合により得られた情報をファイルに保存するために利用するオブジェクトである．ファイル出力に関してはXML形式あるいはHTML形式のいずれかで出力される．HTML形式の場合はそれを通常のWWWブラウザで表示することができるが，XML形式の場合は変換が必要になる．

集合演算オブジェクト：二つのオブジェクトからのメッセージ出力に対して集合演算を行なうオブジェクトである．集合演算には和集合，差集合演算，共通集合演算がある．

関係演算オブジェクト：二つのオブジェクトからの出力に対してリレーショナル演算を行なうオブジェクトである．これには直積演算，射影演算，選択演算が可能である．

これ以外にも様々な特殊オブジェクトを定義することが可能であるが，汎用的なオブジェクトの方が情報統合のために用意するオブジェクトの数が少なくてすむ．

6 具体例

ここで図4に示されるような検索エンジンの統合を具体例として示そう．検索すべきキーワードはファイルに保存されているとする．ファイル入力オブジェクト(File Input Object)はファイルからキーワードを読みだし，検索エンジンA(Search Engine A)と検索エンジンB(Search Engine B)に送る．検索エンジンAでは得られた検索結果をフィルタリングオブジェクトに送る．フィルタリングオブジェクトは検索結果のうち，スコアが80%以上のもののみを統合オブジェクト（UNION Object）に送る．検索エンジンBは検索結果をそのまま統合オブジェクトに送る．統合オブジェクトは二つの検索エンジンの検索結果に対

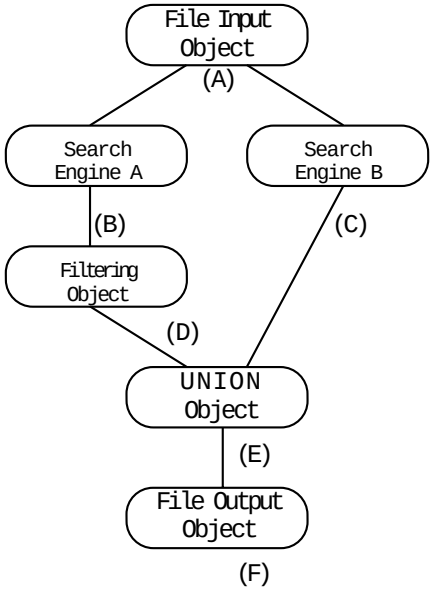

図 4: 検索エンジンの統合

して，URLが両方に含まれるもののみを取り出し，ファイル出力オブジェクトに送る．ファイル出力オブジェクトはそれをファイルに保存する．

7　まとめ

クライアントマシン上での WWW 情報統合システムの構想について述べた．利用者はそれぞれの WWW 情報源を一つの部品と見なし，それらを組み合わせることにより WWW 情報統合を行うことができる．また組み合わせる情報源を変えたり，それぞれの情報源の入力パラメータを変化させることで柔軟な情報統合が可能になる．現在は Java アプリケーションとしてプロトタイプを実装中であり，いくつかの仮想 WWW オブジェクトと特殊オブジェクト，そしてそれらを統合する GUI 環境 [11] を開発した．

今後の課題としては複数の利用者間でのオブジェクトの共有があげられる．それぞれの利用者で開発されたオブジェクトを共有することができればさらに広い範囲の WWW 情報統合が可能になる．また WWW 情報源の問題点としては内容や構造が頻繁に更新されることがあげられる．WWW 情報源からの情報抽出がハードコーディングなどの方法を用いていれば，構造の変更により正しく情報抽出できないようなことも生じる．したがって情報源の変更に応じて迅速にオブジェクトを再配布する仕組みが必要である．現在，オブジェクトは Java のクラスとして実装されているが，オブジェクトの共有に関しては Castanet[9] のようなプッシュ技術を用いて配布を行うことが有効であると考えられる．

謝辞

本研究の一部は文部省科学研究補助金特定研究領域 A(1)「ゲノムサイエンス」（課題番号 08283103）によるものである．

参考文献

[1] Robert B. Doorenbos, Oren Etzioni, and Daniel S. Weld, A Scalable Comparison-Shopping Agent for the World-Wide Web, *Proc. 1st International Conference on Autonomous Agents*, 39–48 (1997). ftp://ftp.cs.washington.edu/pub/etzioni/softbots/agents97.ps

[2] Thure Etzold and Patrick Argos, SRS – an indexing and retrieval tool for flat file data libraries, *CABIOS*, 9(1):49–57 (1993).

[3] Oren Etzioni, Moving Up the Information Food Chain, *AI Magazine*, 18(2):11-18 (1997).

[4] W. Fujibuchi et al., DBGET/LinkDB: an Integrated Database Retrieval System, *Pacific Symposium on Biocomputing*, 3:683–694 (1997).

[5] Shishir Gundavaram, CGIプログラミング，オライリー・ジャパン (1996).

[6] 橋田浩一，GDA 意味的修飾に基づく多用途の知的コンテンツ，人工知能学会誌，13(4):528–535 (1998).

[7] J. Y. Hsu and W. Yih, Template-based information mining from HTML documents; *Proceedings of 14th National Conference on Artificial Intelligence*, 256–262 (1997)

[8] 北村泰彦，野崎哲也，辰巳昭治，スクリプトに基づく WWW 情報統合支援システムとゲノムデータベースへの応用，電子情報通信学会論文誌，J81-D-I(5):451-459 (1998).

[9] Laura Lemay, Castanet, プレンティスホール出版 (1997).

[10] 村田真，XML 入門，日本経済新聞社 (1998).

[11] 野田知哉，北村泰彦，辰巳昭治，WWW 情報資源の仮想オブジェクト化と統合を支援する GUI システムの試作，情報処理学会第 56 回全国大会，4Z-5 (1998).

[12] E. Selberg and O. Etzioni, The MetaCrawler architecture for resource aggregation on the web, *IEEE Expert*, 12(1):11–14 (1997). http://www.cs.washington.edu/homes/speed/papers/ieee/ieee-metacrawler/ieee-metacrawler.html

[13] J. Shakes, M. Langheinrich and O. Etzioni, Dynamic reference sifting: a case study in the homepage domain, *Proceedings of 6th International World Wide Web Conference* (1997). http://www.cs.washington.edu/homes/jshakes/ahoy_paper/paper.html