データベースシステム 109－51
（1996. 7. 26）

構造化文書データベースに対するラッピング手法の提案

古舘丈裕 岡安光彦 石川佳治 植村俊亮

奈良先端科学技術大学院大学 情報科学研究科

独立して管理される様々な情報源に存在するデータをネットワークを通して広く相互利用するために，異種分散情報源に対する統一的なアクセスを提供するラッピング技術が注目されている．本研究では，構造化文書を対象として，文書リポジトリを抽象化するラッパーを構築する手法について提案する．標準的な言語による問合せが与えられた時，対象とする情報源がそれに対応できるかどうかを，領域代数 (region algebra) の枠組を用いて判別する．情報源が対応できる問合せを記述するテンプレートと，構造化文書の文書構造の情報，情報源で利用できる機能の情報などからラッパーの半自動的な生成を目指す．

A Wrapping Scheme for Structured Document Databases

Takehiro FURUDATE, Mitsuhiko OKAYASU, Yoshiharu ISHIKAWA and Shunsuke UEMURA

Graduate School of Information Science,
Nara Institute of Science and Technology (NAIST)

A *wrapper* is a software component that converts data and queries to provide access to heterogeneous information sources. In this paper, we propose a scheme to construct wrappers which extract content information from structured document repositories. When a query is given to a wrapper, it examines whether the underlying structured document repository can *support* the query or not. For the basis of the decision, we use the *subsumption* relationship between *region algebra* expressions. We aim at semi-automatic generation of wrappers. The *wrapper generator* receives declarative information such as query templates and document structure definitions and generates a wrapper.

1. はじめに

文書データは企業や大学で日々電子的に作成され大量に蓄積されているが，それらは応用システムごとに独立している場合が多い．また形式に互換性があっても他のシステムからネットワークを通じて統一的に参照できるようになっているケースは少なく，相互利用が求められている．このような要望への対応は CALS[6] のテーマの一つであり，関連する要素技術の研究・開発が盛んに行われている．また，インターネット技術を活かして WWW とデータベース管理システムを結び付ける製品もいくつか出回ってきている．しかし，文書情報は形式が一様でなく構造が複雑なためデータベースで管理しにくく，単に WWW とデータベースを結び付けるだけでは不十分である．

また，文書情報の相互利用の枠組を情報を提供する側が用意するのでは，その技術の導入コストが高くなってしまう．それぞれの部署や研究室で独立して運営されている様々な形態の情報源に対して，内部に手を加えること無く統一的なアクセスを可能にするようなシステムが望ましい．

本研究室では考古学データを対象として異種分散情報資源の統合に関する研究を行っている．本稿では，分散環境上の情報源として考古学に関する構造化文書データベースを対象とし，統一的なアクセスを提供するラッパーを構築する手法について述べる．

2. 背景

2.1 分散情報資源の統合

情報の統合を目的とする研究は数多く行われており，ARPA による I^3(Intelligent Integration of Information) プログラム [1] では情報資源の知的統合のための参照アーキテクチャが提案されている (図 1)．図の最下段にある情報源は分散して管理されており，それぞれ独立している．

全体のアーキテクチャは，異種情報源の抽象化を行うラッパー (wrapper)，情報の意味的統合を受け持つメディエータ (mediator)，そして情報の加工を行うファシリテータ (facilitator) というように機能によって分割されている．

情報資源としては，関係データベース，オブジェクトデータベース，情報検索システム，ファイルシステム上の文書などが考えられる．情報の統合を行うメディエータは，情報源の内容に SQL，OQL[3] などの一般的な問合せ言語によりアクセスできることを仮定している．ラッパーはメディエータからの問合せを情報源への問合せに変換し，問合せ結果を共通のデータモデルに合わせ加工する．

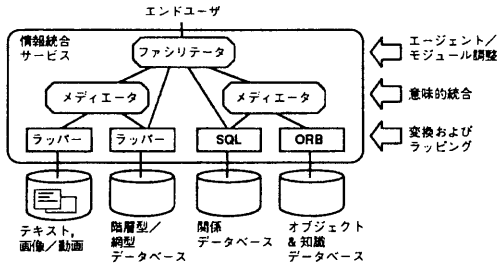

図 1 I^{3}参照アーキテクチャ

2.2 構造化文書とデータベース

SGML を代表とする構造化文書とデータベースの融合に関しても数多くの研究がなされている [4, 5, 8, 9, 10]．一般的には，構造化文書を要素ごとに分解してデータベースに格納する文書分解アプローチと，文書を格納するための新たなデータ型をデータベースシステムに導入する文書データ型導入アプローチに大きく分類される [10]．

これらの研究では構造化文書とデータベースとの密な結合を指向し，高度な問合せ機能の提供などを行っている．本研究とは構造化文書のリポジトリを外部情報源と見なしている点が異なる．本研究における構造化文書のリポジトリは，メディエータによる支配を受けない (non-proprietary な)[7] 情報源である．

構造化文書データベースに対する問合せ処理に関連して，[5] は領域代数 (*region algebra*) と呼ばれる代数に基づく研究を行っている．領域代数の表現 (expression) は次のような構文を有する．

$$\begin{aligned} e \;\rightarrow\; & R_i \mid e \cup e \mid e \cap e \mid e - e \mid \sigma_w(e) \mid \\ & e \supset e \mid e \subset e \mid e \supset_d e \mid e \subset_d e \mid (e) \end{aligned}$$

ここで R_i はテキスト中の領域 (region) 名を表す．領域の集合 R, S について，各演算は

$$\begin{aligned} R \subset S &= \{r \in R : \exists s \in S,\ r \supset s\} \\ R \supset S &= \{r \in R : \exists s \in S,\ s \supset r\} \\ R \supset_d S &= \{r \in R : \exists s \in S,\ r \supset s \wedge \\ &\qquad \neg\exists t \in T, T \in \mathcal{I},\ r \supset t \supset s\} \\ R \subset_d S &= \{r \in R : \exists s \in S,\ s \supset r \wedge \\ &\qquad \neg\exists t \in T, T \in \mathcal{I},\ s \supset t \supset r\} \end{aligned}$$

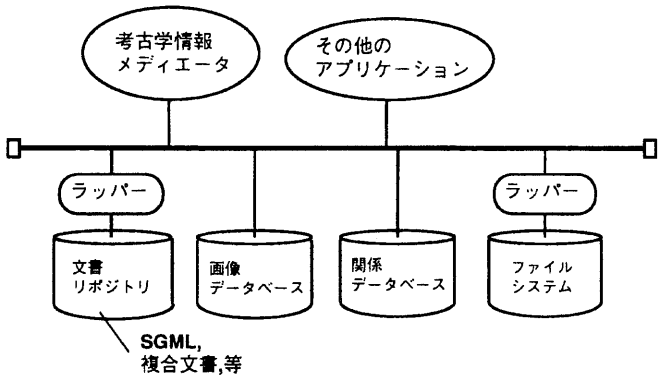

図 2 考古学情報システムの全体像

と定義される．$\supset_d, \subset_d$は直接的な包含関係に基づく演算である．$\sigma[w](R)$ は，領域の集合 R のうちで，単語 wを含む領域の集合を表す．$\cup, \cap, -$は，それぞれ和集合，積集合，差集合演算である．また，領域索引 (*region index*)$\mathcal{I}$は，領域名の集合 $\{R_1, \ldots, R_n\}$ である。

[5] では，特に，演算が$\supset, \subset, \supset_d, \subset_d$のみからなる包含表現 (*inclusion expression*) に関し，問合せの最適化を議論している．

本研究ではこの包含表現を SGML 文書インスタンスに適用し，ラッパーにおける問合せとのマッチングの処理のために利用する．

3. 構造化文書データの情報統合

3.1 考古学情報システム

考古学の分野では，現在，これまでの遺跡発掘情報の電子化が盛んに行われている．しかし，その作業は各地の調査団体ごとに独立に行われ，データの重複や，形式の違い，ネットワークへの未対応等，情報の共有に関して様々な課題を抱えている．

考古学者からは，これまでに提出された報告書からの情報の抽出や報告書の内容を元にしたデータベース検索が，各地に散在している調査団体の保有するデータに対して行え，さらには重複する内容のマージや表現だけが異なる情報の画一化をサポートするシステムが期待されている．

このようなシステムが構築されることにより，考古学者や調査団体にとって，各団体の自律性が保たれながらも個々の変更が全ての参照先に即座に反映されるというメリットが生じる．図2に想定する考古学情報システムの全体像を示す．

3.2 報告書のサンプル

一般に考古学の報告書類は図表や写真を豊富に含んでおり，複雑な構造を持っている．報告書から画像を取り出したり，逆に画像から関連する報告書を検索する要求も考えられる．ここでは ISO や JIS で標準化されている SGML 文書 [2] を想定する．図 3は非常に簡略化した報告書の DTD の構造を表している．この DTD に従った文書インスタンスは例えば図 4のようになる．各情報源にはこのような文書が多数格納されているものとする．

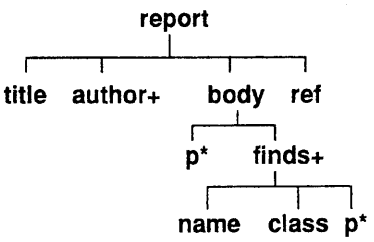

図 3 サンプル DTD の構造

SGML に基づく報告書類は，一部のサイトでは ODBMS や文書検索システム等と統合され管理されると考えられる．しかし，一方ではファイルシステム上で直接管理される場合もある．本研究では著者名等による簡単な検索を支援する SGML 文書のリポジトリが存在することを想定する．

本稿では，構造化文書のラッパーに範囲を限定して，メディエータから発行される OQL 風の問合せを下位の SGML リポジトリへの問合せに変換する部分を扱う．

```
<report>
<title>城山1号墳発掘報告</title>
<author>古舘</author>
<author>岡安</author>
<body>
<p>出土品一覧
<finds><name>菱形留金具</name>
        <class>馬具</class>
        <p>鉄地金銅装で大型鋲を伴う.
</finds>
<finds><name>蓋杯</name>
        <class>須恵器</class>
</finds></body>
<ref>日本古代文化研究,1984</ref>
</report>
```

図 4 文書インスタンス

4. ラッパーの構築手法

4.1 ラッパーの役割

2.1節で触れたように，情報を統合するには情報を表現する共通のデータモデルが必要となり，表現力の高いオブジェクトモデルが適している．ここではODMG-93[3] のモデルを想定する．情報統合の枠組が分散オブジェクトアーキテクチャで実装されるならば，各情報源はオブジェクトとして表現されることになる．ラッパーは情報源を抽象化する役割を果たす．ODL(Object Definition Language) インタフェースで簡単にラッパーを記述すると図5のように書ける．OQL 文はラッパーオブジェクトへの問合せメッセージの引数になっており，その結果は新たに生成したオブジェクトとして返される．

ラッパーの機能を一から定義し実装するのは負荷の高い作業である．ラッパーのコードの多くは共通するものであることから，問合せやデータの変換を宣言的に記述することで，ラッパーが生成されるようなジェネレータの機能が求められる．

4.2 ラッパーの動作

ラッパーの動作は，1) メディエータから問合せを受け取る，2) 情報源特有の問合せに変換する，3) 問合せ結果の加工，4) メディエータへ結果を返す，という流れになり，図6のように表せる．3) における加工には，例えば情報源特有の問合せの

```
interface DocDB {
  attribute integer doc_no;
  attribute Set<Ref<Doc>> doc_set;
  integer   get_doc_num(void);
  Set<Object>
      query(in string OQL-sentence);
}
```

図 5 ラッパーのインタフェース記述例

結果がSGML ファイル全体を返し，一方でメディエータからの問合せがタイトルのみを要求しているような場合に，その切り出しを行う処理が考えられる．そのためにラッパーはDTD を参照しながら，SGML パーザの機能を呼び出す．

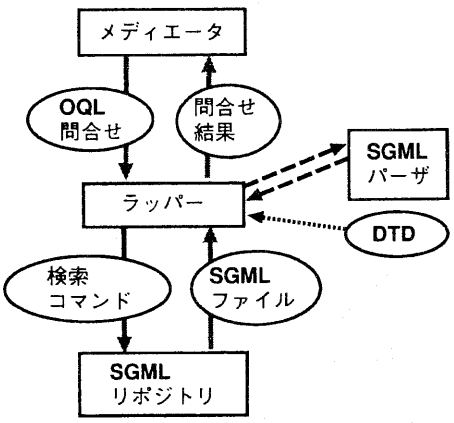

図 6 ラッパーの動作

4.3 想定する情報源

ここでは，ラッピングの対象とする情報源として，図4 のようなSGML 文書が多数格納されていて，<author>と<title>の内容についてのみ索引が用意されているシステムを想定する．すなわち，

```
% search -author ' 岡安'
% search -title  ' 城山1号'
```

といったコマンドが提供されているとする．この例では著書またはタイトルに含まれる文字列を指定して，ヒットしたものがあれば対応する文書ファイルが返される．

このシステムに対して，例えば

```
Q1:  select r.title
     from   report r
     where  r.author = '岡安'
```

という問合せが与えられた場合，ラッパーはwhere句の内容に従ってシステム特有の命令を用いて検索を行い，結果として得られた文書の中からタイトルだけを抜き出して返すことになる．

4.4 問合せの検証

このような処理において，問合せに情報源の情報を用いて回答できるかどうかを判定する処理はラッパーの役割である．

問合せの検証には2.2節でも述べたように領域代数を利用する．本研究では，ラッピング対象の情報源が提供する問合せ機能の記述に領域代数を用いる．情報源が提供する機能を記述したものをテンプレートと呼ぶ．ユーザから与えられた問合せも領域代数に変換され，テンプレートとのマッチングが行われ，与えられた問合せが情報源により支援可能であるかという判定が領域代数表現の包摂 (subsumption) 関係に基づいて決定される．

4.3節の情報源で提供されている検索機能を，最適化した領域代数表現でテンプレートとして表現すると，

$$t_1 = report \supset \sigma[x](author)$$
$$t_2 = report \supset \sigma[x](title)$$

となる．また，Q1 は次の領域代数に変換できる．

$$q_1 = title \subset (report \supset \sigma['岡安'](author))$$

これは，テキスト中の著者の領域に'岡安'を含むような全ての報告書の中からタイトルの部分を全て抜き出すことを意味する[1]．

この問合せがテンプレート t_1 にマッチすることは，領域代数において変数 x に適切な値('岡安')を代入した時，$t_1 \supset q_1 \neq \phi$が成立することより検証できる．

次に，複数の領域を要求する問合せの例として，

```
Q2: select r.title, r.body.finds.name
    from   report r
    where  r.author = '岡安'
```

が与えられた場合を考える．この問合せは，

$$\begin{aligned} q_2 &= (title \cup (name \subset_d finds \subset_d body)) \\ &\quad \subset_d (report \supset_d \sigma['岡安'](author)) \\ &= (title \cup name) \\ &\quad \subset (report \supset \sigma['岡安'](author)) \end{aligned}$$

と変形できる [5]．このような∪を含む問合せについては，

$$q_{21} = title \subset (report \supset \sigma['岡安'](author))$$
$$q_{22} = name \subset (report \supset \sigma['岡安'](author))$$

という二つの部分問合せに分解し，

$$t \supset q_{21} \neq \phi \quad かつ \quad t \supset q_{22} \neq \phi$$

を満たすようなテンプレート t が存在するかどうかを調べる．$t \supset q_{21}, t \supset q_{22}$はそれぞれ内部に$\supset, \supset_d$しか含まない包含表現であるため，多項式時間で一意な正規形に変換でき [5]，結果が空であるかの判定が行える．

一般には，上のように∪を含んだ問合せ表現 Q を ∪を含まない包含表現の集合 $\{q_1, \ldots, q_i, \ldots, q_n\}$ に分解したとき，テンプレートの集合 $\{t_1, \ldots, t_j, \ldots, t_m\}$ について

$$\forall q_i, \exists t_j, \ t_j \supset q_i \neq \phi$$

が成立すれば，その情報源が問合せ Q に回答できることになる．

4.5 ラッパーの生成

ラッパーの構築に必要な情報は，問合せテンプレートの他に，情報源が用意している実際の検索を行うコマンドとテンプレートとのマッピング情報，およびSGML文書の構造を表すDTD，そしてラッパーのODLインタフェースの記述である．ラッパージェネレータはこれらを入力としてラッパーを生成するが，完全に自動化できない部分は実装者がコードを提供することになる．

4.6 関連研究

[7] では，ラッパー構築のためのツールキットについて述べている．与えられた問合せにラッパーが回答できるかという問題を，演繹データベースにおける問合せの包摂関係に帰着させている．ただし，特に構造化文書を対象にしているわけではなく，オブジェクト一般を扱っている．構造化文書では，文書中の領域に基づく検索要求があり，本研究とは領域に基づいた問合せをサポートしていない点が異なる．

[1] [5] と異なり，演算σは部分一致もサポートしていることを想定する．

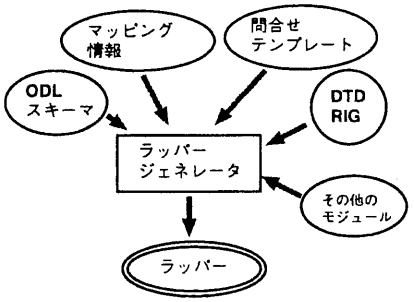

図 7 ラッパーの生成

5. まとめ

構造化文書データベースに対するラッパーを生成する手法についてその構想を提案した．本手法では領域代数を用いて，与えられた問合せに個々の情報源が対応できるかどうかの判定を行う．ラッパーはテンプレートと情報源とのマッピング情報，DTD を用いて半自動的に生成される．

今後の課題として，複雑な問合せとテンプレートとのマッチングの検証の方法，ラッパージェネレータの記述の仕方，ODMG オブジェクトへのマッピングについて検討する必要がある．

謝辞

奈良先端大の吉川正俊助教授，博士課程学生の加藤弘之氏には構造化文書とデータベースについて有益な助言を頂きました．また，本研究を進めるにあたって植村研究室の皆様には有意義な御指導・御討論を頂きました．ここに記して感謝致します．

参考文献

[1] Yigal Arens, Richard Hull, and Roger King (eds.). Reference Architecture for the Intelligent Integration of Information version 2.0, August 1995.

[2] Martin Bryan, 山崎 俊一監訳, 福島 誠訳. SGML 入門. アスキー出版, 1991.

[3] R. G. G. Cattell, editor. *The Object Database Standard: ODMG-93.* Release 1.1. Morgan Kaufmann, 1994.

[4] V. Christophides, S. Abiteboul, S. Cluet, and M. Scholl. From Structured Documents to Novel Query Facilities. In *Proc. ACM SIGMOD International Conference on Management of Data*, pp. 313–324, May 1994.

[5] Mariano P. Consens and Tova Milo. Optimizing Queries on Files. In *Proc. ACM SIGMOD International Conference on Management of Data*, pp. 301–312, May 1994.

[6] 後藤龍男. CALS: 21 世紀における企業情報システムの国際標準確立と企業統合に向けて. 情報処理, Vol. 36, No. 1, pp. 1–7, January 1995.

[7] Yannis Papakonstantinou, Ashish Gupta, Hector Garcia-Molia, and Jeffrey Ullman. A Query Translation Scheme for Rapid Implementation of Wrappers. In *Proc. of the Fourth International Conference on Deductive and Object-Oriented Databases (DOOD'95), Lecture Notes in Computer Science*, pp. 161–186, Singapore, December 1995.

[8] Ron Sacks-Davis, Timothy Arnold-Moore, and Justin Zobel. Database Systems for Structured Documents. In *Proc. of the International Symposium on Advanced Database Technologies and Their Integration*, pp. 272–283, October 1994.

[9] 加藤弘之, 吉川正俊, 植村俊亮. 構造化文書とデータベース間の汎用リンク機構の実現法. 情報処理学会アドバンスト・データベース・シンポジウム, December 1995.

[10] 吉川正俊. 構造化文書とデータベース. 情報処理学会アドバンスト・データベース・シンポジウム, December 1995.