SGML 文書の内容検証について

3S－4

今村 誠　森口 修　鈴木 克志
三菱電機（株）情報技術総合研究所 音声・言語インタフェース技術部

1．はじめに

CALS(Commerce At Light Speed)の進展に伴い、形式を標準化した文書を統合管理することにより、文書情報を既存の情報システムから利用しやすくするための技術が要求されている。この要求に応えるための文書形式が SGML(Standard Generalized Markup Language)である。SGML は、文書型定義(DTD:Document Type Definition)によって、業務に応じて、文書の論理構造を厳密に規定できるので、文書中から必要な情報を機械的に抽出しデータベースに自動登録したり、仕様書から必要な項目を抽出し EDI(Electronic Data Interchange)メッセージに変換するといった処理が容易になる。

しかし、SGML 文書による機械処理を円滑に進めるためには、DTD による構文的な文書構造のチェックだけでなく、アプリケーションに応じた意味的な文書内容のチェック(文書の内容検証)が必要になる。例えば、SGML 文書のデータベース自動登録の際には、格納すべき情報の属性の名称、データ型、データ長、及びデータ単位等がデータ辞書の条件を満たしていることをチェックする必要がある。

本稿では、SGML 文書の内容検証処理を簡潔に記述するための言語について考察する。

2．SGML 文書の内容検証の例

SGML 文書の内容検証では、「文書の部分構造間の比較」や「複数文書間の処理」等のグローバルな文書構造の操作が必要になる。以下、図 1 の購入伺い書を基にして、内容検証すべき条件の例を示す。

(1) 購入伺い書の〈氏名〉と〈会員番号〉タグで囲まれる各々の内容(以下「タグの内容」と記す)が、社員名簿(図 2)の〈氏名〉と〈番号〉タグの内容に矛盾しない。

(2) 購入伺い書の購入品の〈価格〉タグの内容の合計が〈合計〉タグの内容に等しい。

(3)〈合計〉タグの数字が 200000 以上の場合には、〈検印〉タグの内容が“あり”でなければならない。

```
<購入伺い書>
<申請者><氏名>森口 太郎</氏名>
        <社員番号>31415</社員番号></申請者>
<購入品><項目><名称>パソコンA</名称>
                <価格>150000</価格>
                <特記>タワータイプにしてください</特記>
        <項目><名称>モニタB</名称>
                <価格>70000</価格>
        ..... </購入品>
<合計> 325000</合計>
<検印>あり</検印>
</ 購入伺い書>
```

図 1 SGML 文書の例 (購入伺い書)

```
<社員名簿>
<社員><氏名>森口 太郎</氏名>
      <番号>31415</番号></社員>
<社員><氏名>今村 次郎</氏名>
      <番号>26535</番号></社員>
      ..... </社員名簿>
```

図 2 SGML 文書の例 (社員名簿)

3．SGML 文書の内容検証処理の記述言語の要件

内容検証処理では、以下に示すような、SGML 文書を DTD に従って解析した結果得られる「タグでラベル付けされた木構造」の操作の記述が重要になる。

(1)タグの内容の簡単な取り出し

例えば、図 2 の社員名簿から〈社員〉タグの下位の〈氏名〉タグの内容を取り出す。取り出した内容は、部分木(文字列も部分木とみなす)のリストとなる。また、部分木は、データ構造で言えばレコードに相当する。例えば、〈社員〉タグの内容は、〈氏名〉と〈番号〉というラベルをもったレコードとみなせる。

(2)リストとレコードの操作の簡潔な記述

タグの内容は、レコードのリストとみなせるので、リストとレコードを操作するための簡潔な記述が必要となる。

(3)複数の SGML 文書へのアクセス

2. の例のように複数の SGML 文書にまたがる制約関係を検証する場合がある。

Content Test of SGML Documents
Makoto Imamura, Osamu Moriguchi, Katsushi Suzuki
Mitsubishi Electric Corporation. Information Technology R & D CENTER. Human Media Technology Dept
5-1-1 Ofuna, Kamakura, Kanagawa 247, JAPAN

4．SGML 文書の内容検証のための記述言語

4.1 従来の SGML 文書変換言語

SGML 文書の文書構造操作を記述する言語の先行技術としては、OmniMark[1]や DSSSL(Document Style Semantics and Specification Language)[2]がある。

OmniMark では変換のための命令をタグ毎に記述する。また、タグの出現文脈や属性値に応じた条件記述が可能である。しかし、ローカルな文書構造の操作が主目的なので、SGML 文書の内容検証で必要とされる「文書の部分構造同士の比較」や「複数文書にまたがる処理」の記述が困難、ないし複雑となる。

国際規格 DSSSL では、SGML 文書を構文解析した結果得られる木構造を操作する関数ライブラリをもった Scheme 風(LISP の一方言)の言語 Standard Document Query Language(SDQL)を規定している。「木構造を操作する能力の高さ」と「リスト処理の強力さ」に特徴がある。しかし、ライブラリは基本関数に限定されており、プログラマは、SDQL を習得した上で、その基本関数を使いこなす必要がある。

4.2 SGML 文書の内容検証のための記述言語の提案

3.であげた要件を満たす記述言語を提案する。特徴は 3.の(1)(2)に対応する以下の２点である。

(1) パス表現の導入

タグの列を「.」でつないだ表現(パス表現)により、タグの内容を簡単に取り出せるようにする。

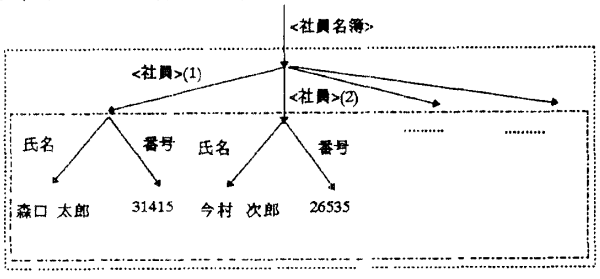

図 3 社員名簿の構文解析結果の木構造

例えば、図２の社員名簿を構文解析した結果図３に示す木構造が得られるが、パス表現によって指し示される部分木は表１のようになる。表１の左欄がパス表現で、右欄が対応する部分木(または部分木のリスト)である。また、パス表現の拡張として、「パス変数(例：*.<氏名>)によるタグの出現文脈のパターンマッチ」や「属性の指定(例：<社員 ID=1>.<番号>)による属性値に応じた条件指定」が可能になる。

表 1 パス表現が指し示す部分木

パス表現例	パス表現が指し示す部分木
<社員名簿>	図２の点線の部分木
<社員名簿>.<社員>	図３の破線の部分木のリスト
<社員名簿>.<社員>.<氏名>	[森口 太郎,今村 次郎,.....]
<社員名簿>.<社員>.<番号>	[31315, 26535,.....]

(2) パス表現により得られるリストに対する操作

「要素がリストに含まれるかの判定」や「要素毎の手続きの適用(Lisp でいう map 関数)」を記述する枠組みを提供する。特に、map 関数により、OmniMark でのタグ毎の命令記述を表現できる。

4.3 記述例

記述例を図４に示す。(a)(b)(c)の記述が 2.の(1)(2)(3)の内容検証処理に対応している。(d)はリストの包含関係の判定例であり、購入伺い書の氏名が社員名簿にあるかどうかをチェックしている。

```
x := sgml_parse(購入伺い書.sgm);
y := sgml_parse(社員名簿.sgm);                          (a)
x1 := x.<購入伺い書>.<申請者>;
z := y.<社員> st (y.<社員名簿>.<社員>.<番号>
                  = x1.<社員番号>);
if(x1.<氏名> != z.<氏名>){printf("社員番号エラー");}

if(x.<合計> !=
  sum(x.<購入伺い書>.<購入品>.<項目>.<価格>))          (b)
  {printf("購入品合計エラー");}

if((x.<合計> >= 200000) && (x.<検印> != "あり")       (c)
  {printf("検印エラー");}

if(x1.<氏名> !∈ y.<社員名簿>.<氏名>)                  (d)
  {printf("%s は社員ではありません",x1.<氏名>);}
```

図 4 内容検証処理の簡易言語による記述例

5．おわりに

パス表現とリスト処理を特徴とする SGML 文書の内容検証用の記述言語について検討した。今後の課題は、複数のアプリケーションを想定し、内容検証処理を記述実験し、言語仕様を評価することである。

参考文献

[1]OmniMark Programmer's Guide Version 2,Exoterica Corporation (1993).

[2] DSSSL:Document Style Semantics and Specification Language , ISO/IEC 10179 (1996).