

## 構造化文書編集ツールの開発と共同文書作成への適用

3S-7

大坪 基秀 藤田 悟 山之内 徹

NEC C&amp;C 研究所

## 1 はじめに

SGML[1]は、構造化文書を表現し、流通する目的で広く用いられつつある。このSGMLをグループの文書管理に利用する場合、特に、グループの業務報告書を作成するような場合には、複数の構造化文書を新たな別の構造化文書に編集する作業が必要となる。本稿では、上記構造化文書編集ツールの開発と業務報告書作成への適用例を報告する。

## 2 共同文書作成における構造編集

グループリーダーが各メンバから電子メールで送られてきた報告書をグループ全体の報告書としてまとめるなど、共同で1つの文書を作成する場合（共同文書作成）、送られてきた個々の文書の一部分を切り取り、グループ報告書のテンプレートに貼り付けた後に、若干の修正や補足を行うというプロセスで、目的とする文書が作成される。例えば、本ツールの適用対象の1つである業務報告書作成[2]の場合には、グループリーダーは、メンバの作成した個人報告書から、（報告対象時期における）各人の“目標”や“成果”などの項目を切り取ってそれぞれマージし、グループ報告書のフォーマット上の対応する箇所に張り付けるといった作業が必要となる（図1）。

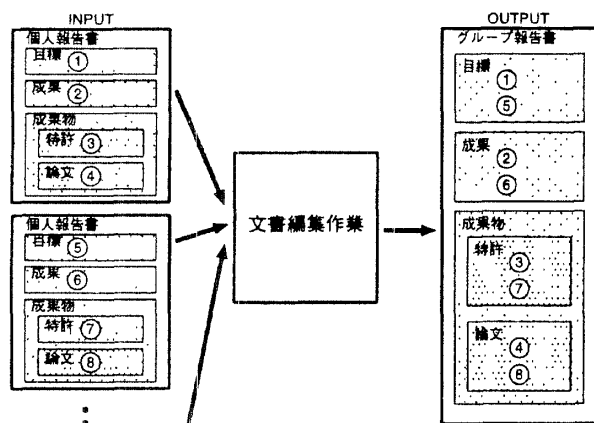


図1: 報告書作成のための編集作業

このような、複数の構造化文書を対象とした文書編集に必要な作業は、次の処理に分類できる。

- 特定の文書構造の抽出とマージ: 例えば、各メンバの目標や成果などの項目をまとめる。

A pattern scanning and processing tool for SGML document  
Motohide OTSUBO, Satoru FUJITA, Toru YAMANOCHI  
C&C Research Laboratories, NEC Corporation

- 抽出した文書構造のソート: 例えば、成果物一覧を、個人毎に件数毎にソートする。
- 特定の文書構造の出現回数のカウント: 例えば、グループ全体の論文の件数をカウントする。

編集対象として、SGMLで記述された構造化文書を扱う場合には、タグを手がかりとすることにより、抽出箇所となる文書構造を特定したり、特定の文書構造の出現数をカウントすることが比較的容易に実現できる。そこで、SGML文書を扱うための独自の文書パターンを定義し、これに従って文書を編集する構造化文書編集ツールを開発した。

## 3 構造化文書抽出のための文書パターン

SGML文書から抽出すべき文書構造を特定するための文書パターンについて説明する。特にSGML文書の処理を念頭におき、文書構造間の位置関係を表現する記号として、順序接続子や階層接続子を用意している。文書パターンの構成要素を以下に示す。

## 基本パターン

1. 構造指定子: <タグ名>  
マッチさせたい文書構造を指定する。
2. 文書内容: "文字列"  
マッチさせたい文書を指定する。
3. 万能構造指定子: <\*>  
任意の複数段の文書構造とマッチする。

## 接続記号

1. 順序接続子: パターンA , パターンB  
パターンAおよびパターンBとマッチする文書構造が、同一の構造内に、この順に出現することを指定する。
2. 階層接続子: パターンA パターンB  
パターンAとマッチする文書構造のすぐ内側に、パターンBとマッチする文書構造があることを指定する。
3. AND 接続子: パターンA & パターンB  
パターンAおよびパターンBとマッチする文書構造が、同一の構造内に存在することを指定する（出現順序は任意）。
4. OR 接続子: パターンA | パターンB  
パターンAもしくはパターンBのいずれかの文書パターンとマッチする文書構造が存在することを指定する。

## 修飾記号

1. 引用指定子: % 構造指定子  
切り出したい文書構造を指定する。1つのパターン中に複数記述することができる。
2. 否定指定子: ! 構造指定子  
指定した文書構造以外の文書構造とマッチ。
3. 括弧: ( パターン )  
優先してマッチ処理すべきパターンを指定する。

例えば、<論文><参考文献>%<項目>という文書パターンは、“論文”という構造の中にある“参考文献”という構造の中の全ての“項目”という構造にマッチする。さらに、“項目”に引用指定子が付けられていることから、マッチした“項目”という構造を全て抽出することができる。

## 4 文書編集のプロセス

構造化文書の編集方法は、前述の文書パターンに編集アクションを組み合わせた編集ルールによって指定する。編集ルールは複数与えることができ、各編集ルールは互いに独立に発火する。また、複数の入力文書各々に対して、全てのルールが評価される。

編集アクションの記述には、汎用のスクリプト言語(Tcl)を用いている。これにより、文書構造の切り貼りだけでなく、文書構造のソートや出現回数のカウント、編集した文書の出力動作など様々な動作の記述が可能である。なお、引用された文書構造は、編集アクションの中では特別な変数の値として参照できる。特に、1つのパターン中に引用指定子を複数記述することを考慮し、引用指定子の出現する順番に、%*n* という名前 (*n* は出現順序を表す整数) の各々異なる変数を用いて参照することができる。

例えば、次のような編集ルールを実行すると、文書パターンでマッチした“項目”という構造が、構造化文書編集ツールの内部変数**bunken**の中にリストとして追加されていく。

文書パターン: <論文><参考文献>%<項目>

編集アクション: lappend bunken \${%1}

ただし、lappend は第一引数の変数の内容に第二引数の値をリストとして追加するコマンドであり、\${%1} は、%1 という名前の変数からその格納値を得るためのオペレーションである。

さらに、構造化文書編集ツールでは、各文書に対して編集ルールを適用する処理の前後、および、全体処理の開始時と終了時に、初期化および後処理を行うための編集アクションを記述することができる。例えば、特定の文書構造の出現回数をカウントするための変数の初期化などに使われる。

## 5 報告書作成への適用

以上の機能を持つ構造化文書編集ツールをTcl言語を用いて作成し、前述の業務報告書作成へ適用した。図2は、記述した編集操作の一部であり、各メンバの報告書から“論文”の項目を抽出しマージするとともに、“論文”の項目の出現回数をカウントするものである。

```
#
# prolog action: 論文, 論文件数を格納する変数の初期化
#
(set paper NIL; set papCnt 0; ..... )
#
# begin action: 個人毎の論文を格納する変数の初期化
#
(set kojIn.pap NIL; ..... )
#
# main rules
#
{
## pattern1: 個人報告から論文情報を抜き出すパターン
( [<個人報告><成果物><論文>%<論文情報> ]
## action1: 抜き出した論文情報を変数に蓄積し、論文件数を1つ増やすアクション
(lappend kojIn.pap ${%1}; incr papCnt )
## 次のパターン、アクション...
( ..... )
}
#
# end action: 個人毎に集めた論文をソートして全体の論文へ反映
#
(lappend $paper [sortPaper $kojIn.pap]; ..... )
#
# epilog action: 変数に蓄積した値からグループ報告書を完成し、出力
#
(outputReport $paper $papCnt ..... )
```

図2: 報告書作成のための編集操作記述の一部

適用の結果、業務報告書作成に含まれる文書のマージ作業は、全て本稿で説明した編集操作記述を用いて記述できることを確認した。また、文書マージ作業の自動化により業務報告書作成作業全体の効率化が実現でき、特に、グループリーダーは、単調な定型的作業から解放され、編集後の文書の言い回しの修正や要約などのより創造的な作業だけに専念できるという効果があった。

## 6 おわりに

構造化文書に適した文書パターンを整理し、構造化文書から文書構造を切り出して新たな文書を作成する構造化文書編集ツールを開発した。本ツールを業務報告書作成作業に適用して、ツールの編集操作記述の記述性を確認し、また、作業を効率化できるという効果を得た。

今後、本ツールをより多くの事例に適用して評価実績を増やしていく予定である。

## 参考文献

- [1] 吉岡誠 編著, SGML のススメ, オーム社, 1992
- [2] 小山, 藤田, 山之内, エージェントによる電子メール業務支援方式, 第54回情報処全国大会, 1997