デジタル・ドキュメント　7－2
（１９９７．　５．　２３）

既存論文の構造化支援ツールの試作と評価

飯尾和彦　木村聡宏　小林透　忠海均
{iio, kimura, kobayasi, tadaumi }@slab.ntt.co.jp
NTT ソフトウェア研究所

既存の非 SGML 文書を SGML 文書に変換するためには DTD に従って多くのタグを人手で文書に埋め込む必要がある。これは、コストがかかり、SGML 普及上の大きな問題点となっている。本論文では、既存の非 SGML 文書を SGML 文書に変換する方式とその評価結果について述べる。

変換方法は、以下の通りである。

(i)テキスト文書に簡単なタグつけを施す。この結果できた文書を一次変換文書と呼ぶ。

(ii)これらのタグと一次変換文書に現れる位置や特徴を手がかりとして、SGML 文書へ自動変換する。

本方式により、SGML の経験者が SGML エディタを利用して SGML 文書に変換するような場合と比較しても 1/4～1/3 の時間でタグつけできることが実験から明らかになった。

Implement and Evaluation of Conversion Tool for Legacy Papers

Kazuhiko Iio　Akihiro Kimura　Toru Kobayashi　Hitoshi Tadaumi

NTT Software Laboratories

It is needed to be put many tags into papers when we want to convert non SGML document into SGML document. However

it is costly and is one of main problems for utilizing SGML. This paper describes the method to convert non SGML document into SGML document and the evaluation of it.

The method is followings:

(i) Insert small tag set into text. The resulting document is called first translated document.

(ii) Translate first translated document into SGML document automatically by analyzing these tag, location and features.

It is found that even SGML expert can convert document three or four times effectively than using SGML editor by using this method.

1．はじめに

SGML(Standard Generalized Markup Language)[1]化された文書は、長期間の保存や電子出版や再利用のために向いている。

DTD(Document Type Definition)に従って、SGML文書を新規に作成することは、SGML文書作成の支援機能を持ったワープロを利用する等によって近年容易になりつつある。

しかし、既存の非SGML文書をSGML文書に変換するためにはDTDに従って多くのタグを人手で文書に埋め込む必要がある。これは、コストがかかり、SGML普及上の大きな問題点となっている。

本論文では、既存の非SGML文書をSGML文書に変換する方式とその評価結果について述べる。

2．用語の定義

ここでは、使用する用語の意味を定義する。

SGML文書

SGML文書インスタンス。章、節、リストなど文書の構造が定義されており、機械(コンピュータ)がその構造を認識できるようになっている文書を指す。

スタイル情報

細明朝やゴシックなどの書体、10pt,18ptなどの文字の大きさ、あるいは、紙に印刷した時に章タイトルは紙の左端から2cmの位置から印刷されるなど、印刷および表示の際の位置情報などのデータを総称してスタイルと呼ぶ。

3．SGML文書への変換の問題点

既存文書からSGML文書への変換において、変換対象の文書に共通するルールが多い場合、ほとんど人手を加えずにSGML化することが可能である。例えば、法規文書など、紙上のスタイルがほぼ一定で、スタイル情報から構造を指定することが可能なものであれば、スタイル情報を構造情報に変換することにより、SGML化することが可能である[2]。

しかし、非SGML文書からSGML文書への変換は、文書の意味や特徴に基づいた情報を付加する作業であり、一般的に完全な自動変換は不可能である[3]。変換対象の既存文書が、たとえ構造的に記述されていたとしても、機械が自動的に構造を認識することは難しい。

また、人手でSGMLタグを付与するにも、変換作業者に、SGMLやDTDに関する知識が必要である。

今回は対象とした学術論文は、人によって記述スタイルはまちまちで、自動変換は難しい[4]。

4．変換方式と変換ツール

4．1　論文用DTD

論文を共有するための論文用のDTDを作成した。DTDの要素数はエレメントのパラメータエンティティが7通り、一般の要素が60通りである。その内訳は章、節、リスト、強調など一般文書で使われる要素と共通のものが35通り、<paper>、<title>など論文構造特有の要素が25通りである。

4．2　一次変換文書の仕様

4．2．1　ヘッダ情報

いくつかの論文を比較すると、いくつかの共通の特徴が存在することに気がつく。例えば、論文のタイトルが必ず先頭にあるとか、著者名や概要が必ず存在するなどである。

そこで、論文サンプルから確定的な情報を抽出し、ヘッダ情報にしてSGMLへ変換した。例えば、一行目にあるものを<title>タグに変換する。これにより、明示的に論文タイトルを表わすタグをつけなくとも、変換ツールが論文タイトルと認識し、SGMLファイルへと変換できる。

以下にそれらの項目を示す。

タイトル	＃1行目
サブタイトル	＃2行目
著者	＃3行目
所属組織	＃4行目
所属組織住所	＃5行目
電話番号	＃6行目
電子メールアドレス	＃7行目

4．2．2 簡易タグ

簡易タグが必要な条件として、以下の内容があげられる。

（1）情報がオプションの場合

オプション情報のためその情報が文書中に存在するか否かがわからない場合、その存在を明示的に示すために、タグが必要となる。

例えば、今回のような論文を対象とした場合だと、キーワードや参考文献がこれに対応する。

（2）テキストに存在しない情報を付与したい場合

テキストを仲介することにより、脱落する情報がある。例えば、強調がこれに対応する。これらは、タグとして明示的に情報を付与する必要がある。簡易タグとしては以下のような種類を定めた。

<keyword>	キーワード
<abstract>	概要
<english>	英語用ヘッダ情報(タイトルなど)
< 図>	図の位置を表す
<表>	表の位置を表す
	番号なし箇条書き
	番号付き箇条書き
<dl>	定義型リスト
<bibliog>	参考文献
<acknowl>	謝辞
<sub>	下付け
<super>	上付け
<highlight>	強調
<note>	注釈

4．3 変換手順

テキスト形式の論文から、手作業により、一次変換文書への変換を行なう。次にそれを、今回試作したDocMakerによりSGML文書へ変換する。

Step 1

テキスト形式から、一次変換文書へ手作業による変換。テキスト文書にタグを入力する作業。

Step 2

一次変換文書からSGML文書への自動変換。運用ではバッチにより同時にHTML文書も生成。

Step 3

修正作業。変換がうまく行われていない場合、一次変換文書の修正を行う。

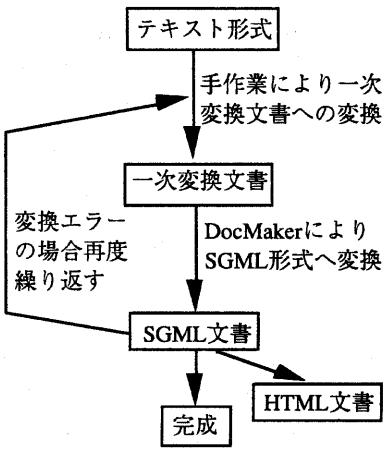

図1．変換手順

4．4 自動変換

DocMakerでは、主に、次のルールに従ってSGML文書へ変換する。

（1）スタイル情報

章、節を表す"1. "とか、"3.4 "などが文頭にくる、あるいは、"・"で始まる行が連続するなど、スタイル上の特徴から文書構造要素を認識し、変換する。

（2）論文特有の構造

4.2.1で述べたような記述ルールにより、タグを付与することなく、定められたSGML構造へと変換する。

3）ユーザが付与したタグ

4.2.2で述べた簡易タグに基づいて変換する。

4．5　特別に考慮すべき事項

4．5．1　図

図に関しては、ワープロで通常にテキスト形式に保管すると、保管対象外となることが多いので、特殊な処理が必要となる。そこで、図2のような処理で変換を行う。

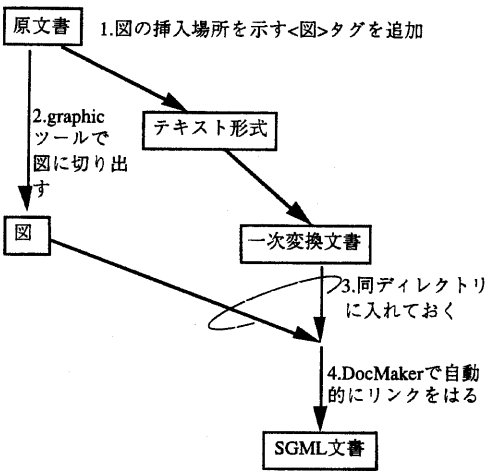

図2．図の変換手順

4．5．2　表

一太郎™、OASYS™などのワープロファイルを、テキスト形式で保管すると、罫線は罫線を表す絵文字に変換して保管される。これをDocMakerではSGMLで用いる表タグに自動的に置き換えるため、手作業は必要ない。表のDTDはHTML3.2[5]と同様のものを利用している。

しかし、マルチカラムなどには対応していないため、複雑な表の変換の場合、図と同様の方法で外部GIFファイルへ変換する。

4．5．3　式

数学記号のSGML化には現状では対処していない。現状では、GIF情報として変換している。これは最終的には、HTMLに変換し、ユーザに提示しているので、SGMLのマークアップを施しても、現状のブラウザでは表示できないという理由もある。

4．6　本方式の長所と欠点

本方式では、テキスト文書を仲介とするため、複雑な変換はできない。また、一次変換文書への変換では、人手による処理を必要とするという欠点がある。

しかし、汎用的な形式であるテキスト形式からの変換という特徴から、ワープロの種類にも左右されず、ワープロのバージョンアップしても無関係で利用できるという利点がある。

4．7　具体例

原テキスト文書の例

○○○システム設計方法の提案
鈴木一郎、山田太郎

1．はじめに
　ハードウェアとネットワークの飛躍的な性能向上を背景として、・・クライアント・サーバアーキテクチャによる情報処理システムが・・

1．1　業務分析
　この作業は従来のシステム設計の分析作業と・・

図1　ベースとなるC／Sシステム

＊サービス処理のプロセスのI/Fは、・・
＊仮想端末プロセスは。・・

一次変換文書

○○○システム設計方法の提案
鈴木一郎、山田太郎

1．はじめに
　ハードウェアとネットワークの飛躍的な性能向上を背景として、・・クライアント・サーバアーキテクチャによる情報処理システムが・・

1．1　業務分析
　この作業は従来のシステム設計の分析作業と・・
<図>
図1　ベースとなるC／Sシステム
</図>
<UL>
＊サービス処理のプロセスのI/Fは、・・
＊仮想端末プロセスは。・・
</UL>

ＳＧＭＬ文書

```
<paper>
<front>
<title>○○○システム設計方法の提案
<author>鈴木一郎、山田太郎</author>
</front>
<section>
<title0>１．はじめに
<para>ハードウェアとネットワークの飛躍的な性能向上を背景として、・・クライアント・サーバアーキテクチャによる情報処理システムが・・
<subsec1>
<title1>１．１　業務分析
<para>　この作業は従来のシステム設計の分析作業と・・
<figure ID="図1">
<graphic fn="zu1">
<figtitle>図１　ベースとなるＣ／Ｓシステム
</figure>
<randlist>
<li><para>サービス処理のプロセスのI/Fは、・・
<li><para>仮想端末プロセスは。・・
</randlist>
```

図３．文書変換の例

５．評価実験

５．１　評価方法

今回提案した（１）のDocMaker方式が、従来のSGML変換方式である（２）および（３）の方式と比較して、SGML変換時の容易性、効率性を検証する。

（１）DocMaker方式

「４．変換方式と変換ツール」で説明した方法。

（２）直接タグ付け方式

DTDまたはそれに相当するものを参照しながら（今回はタグ付けマニュアルを作成してある）、一般的なテキストエディタを使って文書中にSGMLのタグを埋め込む方法。

（３）SGMLエディタ方式

SGML専用エディタはDTDに基づいて、SGMLのタグの候補を表示できる。その機能を利用して、必要なタグを選択してそのタグに対応する文書の内容を、カット&ペーストなどで入力する方法。

５．２　評価実験

５．２．１　被験者

A　SGML熟練者(経験１年半)

B　SGML経験者(経験半年)

C　SGML未経験者

A、B、C各一名ずつ

被験者は、コンピュータ関連の開発・Webコンテンツの作成などの業務に通常従事している。被験者は、文書の内容はほとんど理解していない。

また、Bは、本実験で利用したSGMLエディタを半年間利用した経験がある。

５．２．２　対象論文

論文のテキスト文書を２種類を変換の対象文書とする。

・論文１

文書の構造は比較的単純な論文。文書量は4768バイト。図表類は含まれていない。

・論文２

文書量は3238バイトとやや少ないが、節やリスト構造などを含んでおり文書の構造はやや複雑である。図表類は含まれていない。

５．２．３　実験手順

（１）被験者に変換の方法を説明する。

一次変換文書への変換やSGMLタグ付けなどのマニュアル類は、測定直前に説明した。説明時間25分。

・DocMaker　方式

一次変換文書の作成方法に関して説明する。特別な知識がなく編集できるように、一次変換文書作成マニュアルを作成してある。

・直接タグ付け方式

直接SGMLのタグ付けのためのタグ付与方法を説明する。直接タグつけを行なうためのマニュアルを作成してある。

・SGML　エディタ方式

SGMLエディタ(今回利用したエディタはInContext™)の操作方法を説明する。

被験者A、BはSGMLエディタに慣れていたが、被験者Cは利用したことがないので、時間の測定時に操作方法を教えた。

（2）各文書を3通りの方式で変換実験

被験者に2種の論文を、それぞれ3通りの方式を使って変換してもらう(全6通り)。

5．2．4　変換にあたっての留意事項

タグを付与したあと、DocMaker を通した時にタグの付与が適切でないと、エラー、もしくは予期せぬ結果に変換される場合がある。そのようなエラー発生時は、自己解決が原則であるが、自己解決ができない場合は、AまたはBのものに聞く。

5．3　比較項目

5．3．1　時間の短縮

（1）測定対象

被験者が SGML 文書に変換するのに要した時間に関して、比較を行なう。測定した項目は以下の通りである。

・完成までの所要時間

（2）結果の判定

・変換に要する時間が少ないほど、効率的な変換が行われており、方式が優れていると判定する。

・A、B、Cの人の順に差があるかを確認する。

5．3．2　変換作業の学習容易性

（1）測定対象

A、B、Cの論文1と論文2で入力に要した時間と編集に要した時間の比をとる。

（2）結果の判定

編集に要した比率が少ないほど、入力が効率よくできてきて、変換作業の学習が容易であると判定する。（対象が変わるので厳密な学習効果は期待できないが、参考として測定）

5．3．3　タグの入力量

（1）測定対象

変換中、被験者の手入力によるタグの付与量を調べるために、以下の3ファイルに対してファイル容量を調べた。なお、個人差があるため、A、B、Cの三人のファイルを調べた。

・原ファイル（変換前ファイル）

・一次変換文書

・SGML 文書（手入力）

（2）結果の判定

原ファイルに比べてファイル量の増加分が少ないほど、被験者によるタグの入力量が少なく、作成が容易であると判定する。具体的には、一次変換文書／原ファイル及びＳＧＭＬ文書／原ファイルの値を求める。

6．評価結果

6．1　時間の短縮

被験者 A, B, C に対し、SGML 文書への変換に要した時間を縦軸にとる。

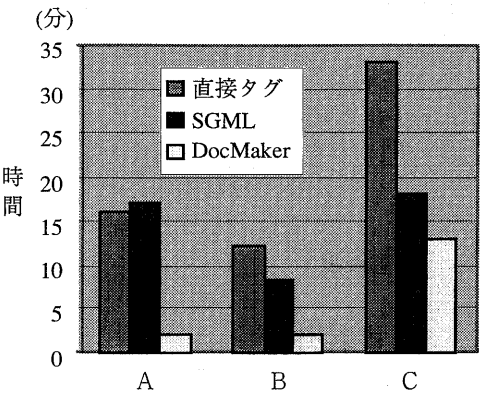

図4．所要時間(論文1)

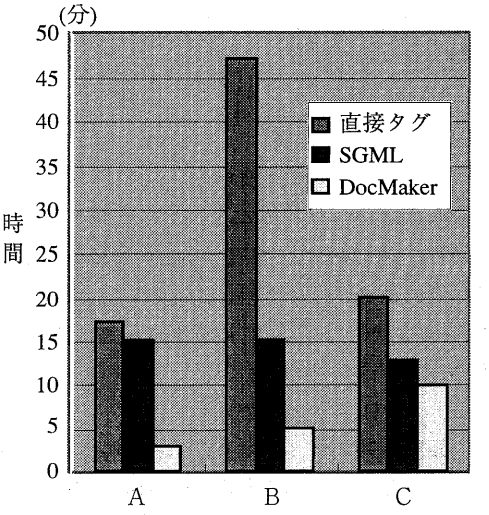

図5．所要時間(論文2)

図4,図5から、以下のことが読み取れる。

・簡易タグ方式での SGML 変換の所要時間は、他の2方式より、非常に少ないといえる。(3/4～1/8)

・手入力の場合は、SGML の経験者でも簡易タグ方式の約7倍の所要時間がかかる。

・SGML 専用エディタを利用した変換では、簡易タグ方式に比べて、1.3～6 倍の所要時間がかかる(ただし、手入力、SGML エディタ、DocMaker の順に変換作業を行ったため、学習効果があると考えられる。)

・DocMaker 方式では、最低限の SGML の知識を取得すれば、被験者 A のレベルにすぐに近づける可能性がある。

6．2　変換作業のしやすさ

DocMaker 方法の変換の論文1と論文2で入力と修正に要した時間比を個人別にまとめた。

表1．論文1と論文2の比較
(全体を 100%とした場合の比率)

内訳	所要時間の比率(%)					
	A		B		C	
	論1	論2	論1	論2	論1	論2
入力	88	76	47	83	45	75
修正	12	24	53	17	55	25
合計	100	100	100	100	100	100

被験者 A の修正時間は論文1と論文2で 12%から 24%へ増加している。被験者 A については SGML と DTD については熟知しているので学習効果はほとんどなく、論文1より論文2の方が構造が複雑な分だけ、修正時間が増加したと思われる。

被験者 B、C については、論文2の方が、デバッグ時間の割合が大幅に減少している。このことから、本方式の学習効果が高いと予測できる。

6．3　タグの量

SGML エディタでは、タグはエディタが付与するので、ユーザ自身はタグは入力しない。そこで、ユーザの入力量に直接影響する DocMaker 方式の一次変換文書の量と直接タグ付け方式の SGML 文書の量を比較する。

表2．SGML 文書の大きさ

被験者	原テキスト文書	一次変換文書	SGML文書(手入力)
A	8006(1)	8099(1.012)	8991(1.123)
B	8006(1)	8105(1.012)	9025(1.127)
C	8006(1)	8128(1.015)	9053(1.131)
平均	8006(1)	8111(1.013)	9023(1.127)

表2から、以下のことがいえる

・原テキスト文書に対して、一次変換文書では 1.3%、SGML 文書では 12.7%、それぞれファイル量が増加している。すならち、タグの量は、一次変換文書は SGML 文書の 1/10 となっている。DocMaker 方式の方が挿入すべき情報が少なくて済むので、SGML 化の所要時間も少なくて済むといえる。

7．考察

7．1　SGML 化方式の比較(定量的)

DocMaker 方式、SGML エディタ方式、直接タグ付け方式の3方式について、6章で述べた定量的な評価をまとめる。

1）タグ(or マーク)の入力量

6.3 節の結果より、DocMaker 方式は直接タグ付け方式に比べ、1/10 の入力量である。

2）完成までの所要時間

6.1 節の結果よりを基に、各方式で要した所要時間を比較する。

以上の評価結果を表3にまとめる。

表3．ＳＧＭＬ化方式の比較

項目	DocMaker方式	SGMLエディタ方式	直接タグ付け方式
タグ(orマーク)の入力量	1	10	10
完成までの所要時間	1	4	6

7．2　SGML化方式の比較(定性的)

DocMaker方式、SGMLエディタ方式、直接タグ付け方式の3方式について、定性的な比較を行なう。

1）タグ(orマーク)選択の容易さ

DocMaker方式は、4.2.2で示した14種のみで選択しやすい。SGMLエディタ方式は、選択タグの種類は全種であるが、エディタがタグの入力をサポートしてくれる。

2）複雑な構成の記述

DocMaker方式は、簡易タグがネスト構造になるなど複雑な構成の論文には使えない。

3）SGMLの予備知識

DocMakerは14種の単純なタグ付けのみのため、SGMLに関する知識は、ほとんど不要。エディタ方式は、DTD構造の図示やタグ入力サポートなどの機能がある。

表4．ＳＧＭＬ化方式の比較

項目	DocMaker方式	SGMLエディタ方式	直接タグ付け方式
タグ(orマーク)選択の容易さ	○	△	×
複雑な構成の記述が可能	×	○	○
SGMLの予備知識が不要	○	△	×

7．3　適性

1．DocMaker方式

複雑な変換はできないが、SGMLの予備知識がなくとも効率よく変換できる方式といえる。

2．直接タグ付け方式

SGML専門家にとっては、SGMLエディタ方式に匹敵する効率が得られる。しかし、SGML専門家以外には不向きの方式といえる。

3．SGMLエディタ方式

SGMLエディタの操作法の学習は必要となるが、対象とするDTDの概要を理解すれば、SGMLの専門家でなくとも使用可能な方式といえる。

7．4　さらなる改善

他にワープロからSGML化の話題として注目すべきものは、Rainbow DTDの取り組みである。これは、スタイル情報を表現するDTDである。今回はテキスト文書からの変換を行ったが、各ワープロメーカがRainbow DTDに対応しれくれば、このDTDを仲介することで、4.2.2の(2)で説明したようなスタイル情報も含めた自動変換ができ、自動変換率の向上が望まれる。

参考文献

[1]ISO8879, Information processing -Text and office systems - Standard Generalized Markup Language (SGML)

[2]岡本卓哉，里佳史，村田英子，樋野匡利、紙法規文書からSGML文書への変換システムの開発、情報処理学会第53回全国大会

[3]SGML実践ガイド、Brian E. Travis, Dale C. Waldt著、学研／スリーエーシステムズ訳、pp. 131-133

[4]飯尾和彦、伊藤光恭、情報共有のための論文作成環境の検討、情報処理学会第53回全国大会

[5]http://www.w3.org/pub/WWW/TR/REC-html32.html

[6]ftp://ftp.ebt.com/pub/nv/dtd/rainbow/rainbow.why