

解説



文書記述言語の標準化動向—II

文書記述言語 SGML とその動向†

田 中 洋 一†

1. はじめに

SGML (Standard Generalized Markup Language)¹⁾ は、共通符号化 (Generic Coding) を基本とする文書記述言語であり、文書を構造化することにより、各種の文書処理の合理化、文書交換及び入力インスタンス (入力文書データ) の再利用を促すものである。

近年はワードプロセッサ、DTP、コンピュータ組版により、大量の文書情報がデジタル化されている。しかしそのほとんどのものは、清書のために入力されたものである。そのため、この文書情報の中には貴重な情報が多く含まれているのにも関わらず、その再利用や情報交換についての考慮が十分されないまま、文書整形に比重をおいた文書処理が行われてきた。

これに対し文書の再利用、文書編集の合理化に重点を置いた SGML が開発され、欧米では公文書の発行やマニュアルの制作などを中心に SGML が使われるようになった。最近ではハイパテキストなどの電子化文書や、全文データベースの構築のため展開が検討されている。

SGML は、文書記述言語であり、文書入力、校正、編集といった文書処理を、人間系と機械系が協力しながら行うことに重点を置いて設計された、文書処理のための道具として位置付けられる。画像や音声情報についても SGML 文書の中で扱うことができ、複雑なマルチメディア情報を記述するにも十分な機能をもっている。

2. SGML の概要

2.1 SGML 関連の規格

ISO では SGML 関連規格として、図-1 に示すように、いくつかの独立した規格からなる文書処理体系²⁾ として開発が進められている。

(1) SGML: SGML 規格による SGML 入力インスタンスは、文書整形、CD-ROM、オンライン検索システムなど多様な応用のための、汎用的な文書データとして利用される。出力形式は定義されておらず、DTP などの応用に組み込まれて使用される。本解説は入力インスタンスとともに、その文書構造を記述する DTD について紹介する。

(2) DSSSL (Document Style Semantics and Specification Language)³⁾: 印刷のみならず、ディスプレイ表示などのあらゆるフォーマティングのための、スタイル指示を行う。実際に可視化処理を行うフォーマタ自体は規格化されず、既存の組版システムをそのまま利用することを想定している。

実際の処理系は、SGML パーサとフォーマタが組み込まれて提供されることになる。

(3) SPDL (Standard Page Description Language)⁴⁾: PostScript の ISO 版であり、フォーマット済みの情報を記述し、出力装置の標準インタフェース言語となる。

(4) フォントリソース^{5)~7)}: スタイル指示、フォーマティング、出力の各処理より参照され、おのおのに統一したフォント情報を供給する。DTP などの文書整形に対する要求が高度化するにつれ、重要度が増大している。

SGML は 1986 年に ISO 8879 として国際規格となった。現在は修正がなされ、AMENDMENT

† Standardization of SGML by Yoichi TANAKA (Toppan Printing Co., Ltd., Electronic Publishing Division).

† 凸版印刷(株)電子映像出版本部

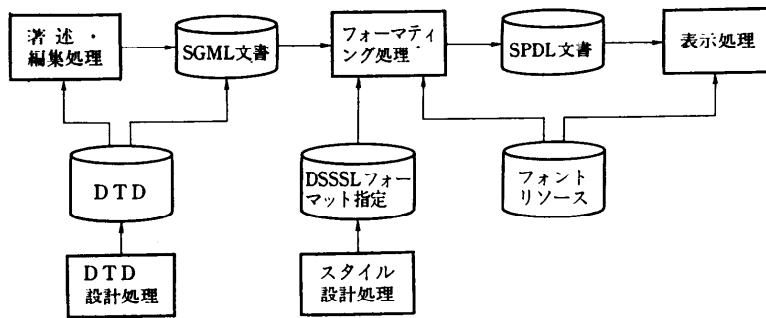


図-1 SGML 関連のオペレーショナル・モデル

1 が発行されているので、規格を参照する場合は ISO 8879 と AMENDMENT 1⁹⁾ の両方が必要である。現在は規格作成後 5 年ごとの見直しが行われており、ISO 8879 本体の改訂が行われる予定である。

さらに関連規格として、SDIF (SGML Document Interchange Format: ISO 9069)⁹⁾ がある。これは SGML 文書の交換フォーマットを定めている。また SGML に関連した各種のテクニカルレポートが ISO より発行されている。TR 9573 (Techniques for using SGML)¹⁰⁾ には、基本的な文書の例や、スプレッドシートの SGML 表現、数式、各種の言語での使用例など、SGML の応用についての解説がされている。現在の ISO 8879 の付録には大変分かりやすい解説が含まれているが、将来この TR に移されることになっている。

SGML 構造エディタについては、それに必要な機能などについて論じた Guidelines for SGML Syntax-Directed Editing System¹¹⁾ が発行されている。SGML 文書は、特別な機能をもたない通常のワードプロセッサやエディタでも効率良く入力できるように考慮されている。しかし、インテリジェントな SGML 構造エディタにより、正確な SGML 文書をより簡単に入力することができるようになる。SGML の本格的利用には必須の機能である。また CTS データなどの非 SGML 文書を、後で紹介する SHORTREF と呼ばれる機能を使って、SGML 文書として入力する手法については TR 9573 に記されている。

SGML 応用規格には、ハイパメディア情報を記述する言語として、HyTime (Hypermedia/Time-based Subset)¹²⁾ 及び SMDL (Standard Music Description Language)¹³⁾ がある。これらはリンク情報や時間情報を含むマルチメディア情報を記述

することが可能であり、SMDL はさらに音楽関連情報を扱うように拡張したものである。ANSI で規格化活動中であったが、昨年より ISO での検討が始まっている。

2.2 共通符号化 (Generic Coding) について

文書に対するマーク付けは、一般のワードプロセッサで行っているような、文書スタイルに関する指示命令と、SGML で行うような、一般的な要素の名前によるマーク付け、つまり共通符号化に分けることができる。

印刷用マークは処理システムに依存し、マークされた文書はその応用に固有なものとなるため、スタイルの変更や、他システムとの情報交換に大きな障害が生じる。

これに対し共通符号化は、その文書の要素のもつ構造的な特質、たとえば表題とか、章見出しというような一般的に共通な性質を表す要素名をマークする。スタイルの指示は、その要素ごとに、目的とするスタイルの指定を記述し、共通符号化された文書から変換することによって得る。したがってこの文書情報は、処理システムや応用から独立であるので、スタイルの変更や、他の応用への展開、システム間での情報交換などに大変有利である。

このような考え方をもちたフォーマタに Scribe がある。直接印刷命令を指定せずに、表題とか、著者、箇条書というように、文書の構造に基づく一般的なマーク、つまり共通符号化を行う。また Scribe にならって開発されたフォーマタに LaTeX がある。LaTeX は強力なマクロ言語をもちたフォーマタである TeX 上に、共通符号化の機能を付加したものである。整形指示は一般的なマークに対応するマクロ命令で指示されているため、一応文書情報とは分離することができる。

したがって、LaTeX を用いて共通符号化文書を交換することが可能である。また必要に応じてスタイルを指示するための命令を挿入することもできるため、文書整形を主目的とする文書交換には、たいへん有効である。しかし、文書構造を明示的に示す機構がないので、データベースが主目的となる場合は、SGML と組み合わせ、文書構造を解釈する機構を付加する必要がある。

2.3 文書構造の必要性

たとえば議事録は、会議名、開催日時、場所、出席者、議題、などの共通要素からなっており、その出現の順番なども一定である。この文書構造を利用して、文書インスタンスより報告書を自動的に印刷することができる。SGML では文書構造を利用して、SGML 入力インスタンスの解釈を行うようになっている。文庫本、論文、マニュアルなどの文書はそれぞれ固有の文書構造をもっているため、文書構造を明示的に記述し、その文書構造を共有することによって、個々の入力インスタンスの解釈方法を共有することができる。SGML ではこれを DTD (Document Type Definition) と呼んでいる。

3. SGML 開発の背景と動向

共通符号化は 1960 年代の終わりごろ GCA (Graphic Communications Association) で研究が始まり、GenCode が開発されたのが始まりとされている。

1969年に IBM の Charles Goldfarb により、GML (Generalized Markup Language) の開発が行われ、文書型定義の形式定義の概念が導入された¹⁰⁾。

IBM は、コンピュータシステムのマニュアルなどの出版で、世界で有数の出版社でもあるわけだが、現在ではその 90% 以上の出版物が GML によって出版されているといわれている。

1978年に、ANSI での文書記述言語の標準化活動が始まり、1980年に最初の SGML 規格原案が出された。名前のとおり、GML が基礎となっているが、C. Goldfarb らにより、short reference, link processes, concurrent document type などの概念が提案され、現在の SGML の骨格を作った。これは IRS (US Internal Revenue Service) 及び国防総省によって採用された。

1984年には ISO による標準化活動が始まり、

1985年に出された規格案は EC 出版局に採用されて、9カ国語に及ぶ言語での公報出版に使われている。そして翌年 1986年に ISO 国際規格となった。

SGML は言語であるので、実際の文書交換など、文書の共有を目的とするには、さらに文書構造の標準化を行う必要がある。このような応用レベルの標準化には、IEEE、米化学会、数学会、のほか多くの学会や出版社が参加して、米出版協会 AAP (Association of American Publishers) によって開発された EMP (Electronic Manuscript Project) がある。これは現在 ANSI 規格 (Z 39.59) となっており、内容としては、単行本、雑誌、論文、の基本的な文書型が規定してある。これには数式の記法や表組の記法についての規定も含まれている。

また大変大きなプロジェクトとして、米国防総省による CALS (Computer-aided Acquisition and Logistic Support) がある。これは国防総省の調達物資に関する膨大な情報を全て電子化しようというプロジェクトであり、1990年より段階的に実施されている。規格は、文書情報に SGML を採用し、画像情報に IGES 及び G 4, CGM, 交換媒体には磁気テープを用いる仕様 (AITI) からなっている (MIL-M-28001)。このため、納入業者はマニュアルを SGML で電子化して作る必要があり、アメリカ、及びこれに追従したカナダなどでは SGML を扱うことのできる DTD システムが開発、発売されている。

日本では欧米に比べ、普及が遅れたが、1989年に SGML 懇談会が発足し、SGML の普及活動を始めた。特許庁が昨年より始まった電子出願にともなう特許公報の電子化に、SGML を採用することが決まっているが、それ以外の本格的な応用はまだ行われていない。それでも学术论文の電子化について、SGML による論文の全文データベース化の検討が始まるなど、SGML 応用の動きが進みだしている。

日本の標準化活動としては、ISO 8879 の JIS 原案¹¹⁾が作成されており、現在工技院による審議が行われているのである。またこれより先に、前述の EMP に関してすでに JIS 原案¹²⁾が作成されているが、ISO 8879 の JIS 化を待って審議されることになっている。

4. SGML の規定内容

SGML は次のような内容を規定している。

- (1) 文書の構造及びその属性情報の記述方法を示す、抽象構文。
- (2) この抽象構文に特定の文字などを与え、具体構文を定義する方法。
- (3) この具体構文に適合した文書の記述方法。
- (4) この SGML 適合文書进行处理し、文章中のマークの誤りを指摘する能力をもつ適合システムについての規定。
- (5) 画像など、多様な情報を SGML に組み込むための記述方法。

SGML は文書情報を直接扱うため、対象とする言語や、文書内容に制約を生じないようにする必要があるので、つまり直接文書の記述方法を規定するのではなく、抽象構文を規定するメタ言語になっている。このため複雑で難解な規格になっているが、具体化された構文自体が複雑というわけではない。

以下に紹介する文法の表現は、具体化された構文を使用するが、ISO 8879 での実際の構文は、全て抽象的な表現でなされている。たとえば“<”は tago (TAG OPEN), “</”は tagc (TAG CLOSE) “<要素名>”は starttag というように呼ばれている。“<”や“</”は、ISO 8879 でデフォルトで決められた表現であり、実際に使用する記号は上記(2)で使用文字コードなどを宣言することになっている。

SGML 文書は、大きく分けると三つの部分に分けることができる。

- (1) SGML 入力インスタンス
- (2) DTD (Document Type Definition)
- (3) SGML 宣言

(1)はエンドユーザが直接記述すべき原稿である。(2)は文書構造を定義するもので、応用ごとに宣言するが、エンドユーザがこれを記述するようなものではない。(3)は使用する言語やシステムの環境などに応じるものであり、応用ごとに変わるものではない。実際の宣言は、(3)、(2)、(1)の順に行うが、直感的な理解を助けるため、(1)より説明を始める。(3)については簡単にふれるにとどめる。

4.1 文書情報の記法

図-2 は印刷物 (図-3) を得るための SGML による原稿である。実際に図-3 を得るための仕組みは 4.2 で説明する。ここで“<論文>”や“<題名>”は構造化された要素の開始を示している。“論文”は最上位の構造名であり、“題名”や“著者”、“章”などを含んでいる。これらの要素も 4.2 で説明する DTD で定義する。全ての要素は原則として開始タグと終了タグによって囲まれる。“<論文>”や“<題名>”は開始タグであり、“</>”や“</章>”は終了タグである。“</>”は直前の開始タグのタグ名を省略したものである。

“<著者 会員=yes>”の“会員”は“著者”の属性を示しており、値が“yes”であることを示している。“会員”属性は“著者”の下位要素としても良いのだが、一般に編集者が特定の処理のために

```

<論文>
<題名>SGMLによる文書処理</>
<著者 会員=yes>山田太郎</>
<章><章題>SGMLとは</>
  <段落>SGML<脚注>Standard generalized Markup Language:ISO 8879</>は文書情報を構造化し
  . . . . . </>
  <段落>したがって、&WP;のように. . . . .
  . . . . . </>
</章>
<章><章題>文書情報の. . . . .
. . . . .
. . . . .

```

図-2 SGML 入力インスタンス例

SGMLによる文書処理

山田太郎

- 1.SGMLとは
 - SGML¹⁾は文書情報を構造化し、.
 -
 - したがって、ワードプロセッサのように.
 -
- 2.文書情報の.
 -
 -
 -

¹⁾ Standard Generalized Markup Language: ISO 8879

図-3 印刷例

付加する情報は属性として扱い、オリジナル情報と区別されることが多い。属性指定を列記することにより、複数の属性を記述することができる。

文書要素の一般的な記法は
<要素名 属性指定 (複数)> 要素データ </要素名>

である。

エンドユーザにとって SGML の本質部分は、以上でほとんど全てであるといつてよい。もちろんエンドユーザは、著作物の構造について当然知っている必要がある。

タグは、本来編集者や印刷業者によって組版指示を付加していたものであるが、SGML では文書の一般的な要素名を指示するため、著者による入力も可能となる。ただしこれは、SGML によるタグ付けを必ずしも著者に要求するものではなく、従来どおり編集段階で付加してもかまわない。しかし文書構造のように著者が認識している情報は、なるべく創作時に入力したい。

タグ付けは SGML に限らずめんどうなものであるので、SGML には、多くのタグ付け省略に関する機能がある。ここでは詳細にはふれないが、図-4 のようにほとんどタグのない SGML 入力インスタンスを設定することもできる。要素の順序が明確であれば、終了タグあるいは開始タグは省略可能であるし、タグの代わりをする文字列を定義しておけば、通常 の原稿上の記法と同様な文書を SGML 入力インスタンスとすることができる。

ここでは属性をもつ要素のタグ “<著者>” が省略されているが、属性値がデフォルトの場合は省

略が許される。

このほか入力インスタンス中には、“&WP;” のように実体 (Entity) を表す一種の短縮記法が許される。ここでは “&” と “;” ではさまれた “WP” が、“ワードプロセッサ” の省略であることを示している。また大きな外部ファイルや、SGML 文書の部分を参照することも可能である。

文中に “<” や “>”, “/” などのマークに使用している文字がデータとして現れる場合は、SGML 宣言でエスケープ文字を指定することが可能である。また長い文章を操作するために、マーク指定区域 (Marked Section) と呼ばれる次のような構文で要素データの扱い指示を行うことができる。

<![扱い指示 [要素データ]]>

扱い指示には、マーク文字有効、マーク文字無効、実体のみ有効、データ無視などの指定が可能である。扱い指示は実体が使えるので、パラメタによる要素データの扱いを変更することが可能である。

このようにして、構造化を要する全ての要素に分解し、タグをつける。要素名は前述した一般的な名前である。この要素は次に説明する文書構造定義で設定する。したがって SGML 文書の交換は、タグのつけられた入力インスタンスとともに文書構造定義も交換する必要がある。ただし AAP による文書構造の標準化のように、文書構造をお互いに共有化できれば、その構造名を知らせるだけでよい。

4.2 文書構造

文書構造は図-5 のように ELEMENT 文によって階層的に定義され、#PCDATA によって示される終端要素である文字列まで分解される。“論

SGMLによる文書処理

山田太郎

SGMLとは

SGML(Standard generalized Markup Language:ISO 8879)は文書情報を構造化し、.....

したがって、&WP;のように.....

文書情報の.....

図-4 省略機能を使用した SGML 入力インスタンス例

<!ELEMENT 論文 0 0 (題名,著者,章+) >
<!ELEMENT 題名 - 0 (#PCDATA) >
<!ATTLIST 題名 会員 (yes|no) "yes" >
<!ELEMENT 著者 - 0 (#PCDATA) >
<!ELEMENT 章 - 0 (章題,段落*)+(脚注)>
<!ELEMENT 章題 - 0 (#PCDATA) >
<!ELEMENT 段落 - 0 (#PCDATA) >
<!ELEMENT 脚注 - - (#PCDATA) >
<!ENTITY WP "ワードプロセッサ">

図-5 文書構造宣言例

文は“題名”，“著者”，“章”の下位要素からなっていることを示し，“題名”は文字列からなっており，構造をもたないことを示している。“+”は要素の出現選択記号であり，

- ? : 任意選択 (0回または1回)
- + : 必須で反復可能 (1回以上)
- * : 任意選択で反復可能 (0回以上)

などがある。また順序記号として

- , : 全てが示した順序で出現する
- & : 全て出現するが，順序は任意
- | : いずれか一つが出現

などがある。

“章”にある“+(脚注)”の“+”は，出現選択記号ではなく包含を示している。この意味は“脚注”が“章”のどこに現れても良いことを示している。同様に“-”を使用して宣言した要素の内容から特定の要素の出現を排除することもできる。これらは少しだけ構造の異なる要素を，効率よく定義するのにたいへん有効である。

一般的な記法は次のようになる。

<!ELEMENT 要素名 -- (構成要素リスト)>

図-5の“<!ATTLIST 著者...”は属性の定義を行っている。図-2の“著者”要素に“会員”属性があり，その値は“yes”または“no”であり，デフォルト値は“yes”であることを示している。属性定義は複数指定できる。一般的な記法は次のようになる。

<!ATTLIST 要素名 属性定義リスト>

属性定義リストは，属性名，値のタイプ，既定値からなるリストである。

“<!ENTITY...”は実体の定義を行っており，図-2の“&WP;”で呼び出す。

文書情報を記述するには基本的にマーク付きのSGML文書と文書構造定義があればよい。しかし効率的な文書処理を行うために，前述したタグの入力を省略したり，また文書を印刷したりするために，いくつかの基本的な文書処理機能が必要である。このためSGMLには，次に説明するような機能が付加されている。

4.3 入力 (SHORT REFERENCE)

文書を厳密に記述すると，入力インスタンスはタグだらけのものになってしまう。このような文書を人間が直接書くことは大変面倒なことであるし，かえってタグのないほうが読みやすい。人間

は，タグがなくても，体裁を利用して文書を構造化している。

たとえば箇条書では行の先頭に“•”印があればそれと直感的に理解できる。SGMLにはこの解釈法を機械系に知らせる機能がある。SHORTREFと呼ばれる一種の変換表を用い，SGML文書の入力を省力化するとともに可読性を改善する。図-4は図-5の文書構造定義に図-6を付加することにより，タグの省略を行ったものである。

“<!USEMAP tmap 題名...”は要素“題名”中では，“tmap”の名前の付いた“<!SHORTREF...”を適用することを示している。“<!SHORTREF tmap '&#RE; &#RS;...'”では，改行 (&#RE; &#RS; で示されている)を“startau”のEntity名で示される“</題名><著者>”に変換することを示す。つまり題名中に改行が現れたら自動的に“題名”を終了して，“著者”を開始することを指示している。SHORTREFには複数の変換対を指定することができる。

この機能をうまく使えば，ほとんどタグのないSGML文書を作ることができる。ただし人間系がこの変換ルールに合った文書を書けばの話であって，あまり複雑なルールや著者の習慣に合わないルールは避けるべきである。文書入力の本質的な解決にはSGML構造エディタを使うべきであるが，既存の文書データをSGML文書として処理可能にするために，必要な機能である。

```
<!ENTITY startau "</題名><著者>" >
<!SHORTREF tmap "&#RE;&#RS;" startau >
<!USEMAP tmap 題名 >
```

```
<!ENTITY startsc "</著者><章><章題>" >
<!SHORTREF aumap "&#RE;&#RS;" startsc >
<!USEMAP aumap 著者 >
```

```
<!ENTITY startfn "<脚注>" >
<!ENTITY endfn "</脚注>" >
<!SHORTREF pmap "{" startfn
                "}" endfn >
<!USEMAP pmap 段落 >
```

図-6 SHORTREF 宣言例

4.4 文書型の操作 (LINK)

SGML 文書は多くの応用に展開することができる。通常その応用展開に必要な処理は、その応用に必要な、固有なマークの付加である。SGML 文書の全ての要素にマークが付けられていれば、基本的にはタグをその必要なマークに変換すればよい。SGML では LINK と呼ばれる機能を用いてこの変換を行う。たとえば組版の例では全ての要素は、最終的に、組版指示をもった矩形の領域 (BLOCK) に変換される。図-7 の例では、“題名”はフォントサイズ属性が 12 ポイントで、フォント属性がゴシック、配置属性が中央揃えの属性が与えられる (図-7)。

“<!LINK…”の宣言は複数指定することができるので、属性値や要素名によって変換指示を変えることができる。LINK 機能も、SHORTREF 機能と同じく文書構造を利用した強力な文書処理機能といえる。

SGML による文書処理システムの概略は図-8 のようになり、文書構造解析機能と解析済み内部文書データを中心に、各種の省略記法復元を行う入力部及び目的とする文書の構造に変換する出力部から構成されている。

SGML では出力フォーマットは規定されていないので、実際の応用のデータ形式は処理系にまかされることになる。DTP システムに組み込ま

```
<!LINKTYPE tex 論文 プリント
[<!LINK #INITIAL
題名 BLOCK [サイズ="12"
           フォント="ゴシック"
           揃え="中央"]
著者 BLOCK [サイズ="10"
           フォント="明朝"
           揃え="右"]
.
.
.
脚注 BLOCK [サイズ="8"
           フォント="明朝"
           揃え="左"] > >
```

図-7 LINK 宣言例

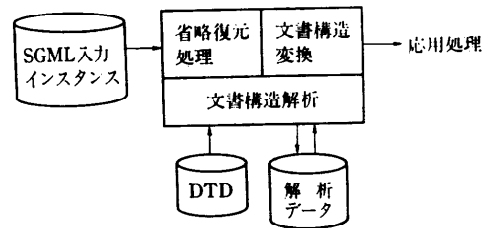


図-8 SGML による文書処理概要

れた SGML では、出力結果はその DTP システムのデータ形式となるものが多いようだが、単体で市販されている SGML パーサには、SGML 文書形式の出力が得られるものがある。この場合も簡単な変換プログラムを追加することによって応用に必要なデータを容易に作成することができる。

ところが元の SGML 文書構造と応に必要な文書構造は必ずしも一致していない。たとえば書籍の奥付けは、和書では最後のページにあるが、洋書では最初のページにある。このような要素データの位置の移動などの高度な文書構造変換機能は、SGML ではサポートしていないが、関連規格である DSSSL でサポートされている。

4.5 SGML 宣言

SGML 宣言では、マークとして使用した文字の宣言、データとして使用した文字の宣言、要素名など名前に使用する文字の宣言、特別の機能を指定した文字の宣言などを行う。また名前に使用する文字の長さなど、処理系に必要な各種のパラメータや、SGML 文書が必要とする各種機能の宣言なども行う。これらの宣言は、使用する言語による事情や、システムによる機能の違いなどを吸収するためのものである。したがってローカルに処理を行うには必要がないが、文書交換時に SGML 宣言も交換することによって、使用している言語や、環境の異なる処理系間でも確実な交換が保証される。

5. SGML 普及上の問題点

SGML は 1986 年に ISO 規格となったが、欧米での政府主導による SGML 応用以外は民間などの利用が遅れている。その理由の一つに、SGML の難解さと処理系の重たさが指摘されている。

SGML は文書構造と共通符号化を利用した、

明快な原理をもつが、言語やシステム、応用によらない汎用性と文書交換の保証やタグ入力の省略などのために、複雑な規格となっている。このため SGML のサブセットを定義して、簡易 SGML パーサを開発しようという動きも生じている。通産省の指導で SGML の普及啓蒙を行っている SGML 懇談会では、ビジネス文書用途で実用に耐えるような SGML サブセットのパーサを開発している¹⁷⁾。

また SGML は人間系での文書の扱いに重点をおいて設計されているが、機械系ではタグの省略などは不要である。全文データベースに SGML 文書を使用した場合、処理の効率上、タグを辿らなくても機械系は要素を直接アクセスできる必要がある。このため SGML の 2 値表現の規格が検討されており、現 SGML と完全に変換可能であることを前提に検討がされている。

SGML は原理的に使用する言語によらないように設計されている。日本語の使用についても、2 バイトコードが使用できる処理環境ならば、JIS コードが使用できる。しかし使いやすさの点からいうと、コードの扱いに不便な部分もある。SGML では、データとして使用する文字や名前文字の宣言に、10 進数の文字番号を使用するので、2 バイトコードのように文字数の多いものは指定がやっかいとなる。特に要素名に使用する名前文字の指定は、範囲指定ができないので SGML 宣言が大変大きくなってしまふ。SGML 宣言は一度作成すれば、あとは必要に応じて呼び出せば良いので、致命的な問題にはならないが、次回の規格改訂時に修正を検討することになっている。

運用面では、前述した AAP や CALS の文書構造のように、応用ごとの文書構造に関する取り決め、つまり DTD の標準化が必要である。今後オフィス環境での文書処理の比重がますます高まっていくと思われるが、SGML に限らず文書情報の共有化、交換、再利用、を促進するために、応用分野ごとに標準文書構造を取り決めておくことが重要である。

参 考 文 献

- 1) ISO 8879, Standard Generalized Markup Language (SGML) (1986).
- 2) ISO/IEC JTC 1/SC 18/WG 8 N 1078, Operational Model for Text Description and Processing Languages (1990).
- 3) ISO/IEC JTC 1/SC 18/WG 8 N 10179, Document Style Semantics and Specification Language (DSSSL) (1991).
- 4) ISO/IEC DIS 10180, Standard Page Description Language (SPDL) (1991).
- 5) ISO/IEC DIS 954-1. 2, Font information interchange—Part 1: Architecture (1990).
- 6) ISO/IEC DIS 954-1. 2, Font information interchange—Part 2: Interchange Format (1990).
- 7) ISO/IEC DIS 9541-3, Font information interchange—Part 3: Glyph Shape Technologies (1991).
- 8) ISO 8879/AMENDMENT 1 (1988).
- 9) ISO 9069, SGML Document Interchange Format (SDIF) (1988).
- 10) ISO/IEC TR 9573, SGML support facilities—Techniques for using SGML (1988).
- 11) ISO/IEC DTR 10037, SGML and Text-entry Systems—Guidelines for SGML Syntax-Directed Editing Systems (1989).
- 12) ISO/IEC CD 10744, Hypermedia/Timebased Structuring Language (HyTime) (1991).
- 13) ISO/IEC CD 10743, Standard Music Description Language (SMDL) (1991).
- 14) Goldfarb, C.: Standardization of SGML Future Perspectives, SGML セミナー SPRING '90 資料 (1990).
- 15) 日本規格協会: ニューメディア間インターフェースの標準化に関する調査報告書 (1991).
- 16) 日本電子工業振興協会: カラーデジタル画像システムの標準化に関する調査研究 (1990).
- 17) SGML 懇談会: SGML サブセット仕様 (1991).

(平成 3 年 5 月 21 日受付)



田中 洋一

1948 年生まれ。1972 年上智大学理工学部物理学科卒業。現在凸版印刷(株)電子映像出版本部システム開発部開発グループ担当課長。CDROM などマルチメディアデータベースの開発に従事。ISO/IEC SC 18/WG 8 国内委員会委員。