

3 文書の電子化手法の調査

3.1 電子化の目的

現在、施工計画書も帳票も、紙に印刷されたものであるという理由だけで、ともに同じ様な「文書」として扱われている。しかもパソコン(ワープロ)を使えば、両方とも電子化できることは一応理解されている。それでは、パソコン(ワープロ)を使う彼らに電子化する目的は何ですかと質問すると、たいていの場合、「きれいに印刷するため」、「データが再利用出来るため」、との返答がかえってくる。「きれいに印刷するため」は、電子化の最終目標を印刷に置いている。「データが再利用出来るため」は、表面的にみれば、情報の蓄積を考えているような思えるが、雛形を保管しているだけで、その雛形は次に同様な「文書」を作るときに役立てることが目的である。これが電子化の第1の目的である。また、文書データの形で提出する場合もあるが、実際に印刷するしないに関わらず、印刷できることを前提とした形で、電子化されている。電子化の第1の目的は、「印刷」を前提にしているにすぎない。

しかし、帳票*1には電子化の別の目的も合わせ持っている。紙に印刷して提出すれば良いのであれば、それを作成する側の電子化は、「きれいに印刷するため」と雛形として「データが再利用出来るため」だけで良い。それ以上の電子化目的は特に見いだせないと言える、しかし、帳票を受け取った側が、帳票から特定の項目を抽出し整理することを試みた場合、「印刷」を前提にした形式の文書データでは役に立たない。ここに至って、初めて、「印刷」や「印刷のイメージ」*2に束縛されない文書データの形式(新たな文書の電子化手法)が必要であると考えられるようになるのである。受け取った文書データから、情報を最初に引き出すのはコンピュータであり、そして、コンピュータに集積されたデータを用いれば、帳票がバラバラにあるのとは異なり、データの集計や検索が容易にできる。これが電子化の第2の目的である。

例えば、施工計画書の場合、技術的内容を記述した部分については、人間が内容を読んで理解することが是非必要とされる。特定の項目だけ取り出しても意味は通じない。印刷するしないに関わらず、印刷のイメージは大事にすべきであろう。内容を読みとる対象が常に人であるため「きれいに印刷する」目的と編集のしやすさや編集後のデータとしての「集計や検索を容易にする」目的の両者を兼ね備えた電子化が求められているのである。

以上は、厳密な承認行為を含まない場合であるが、これらに電子的な承認機能を加えれば、さらに追加して異なる技巧が必要とされる。これまで印鑑で処理していたものを電子化することは、文書データとその管理システムに、ワークフローに対応した機能を持たせることになると考える。

3.2 成果物による電子化の目的と手法の違い

(1) 施工計画書

目的

一つの本として捉え、記述する内容とその順番と位置(章節項)の標準化(木構造の標準化)を目指す。そして、その標準化に沿った執筆を支援する目的での電子化を考える。「印刷」や「印刷のイメージ」も重視される。

電子化手法

現在のワープロでの作業の延長として捉え、標準化に沿った執筆を支援できるツールがそこに加わる形での電子化を行う。ワープロであるので、印刷やWeb表示も重視した電子化を行う。さらに、文書データとして木構造を保ったままで、交換や保管できる様な電子化を行う。上記に対応した電子化手法を提案する。

(2) 帳票

目的

必要な項目が電子データの形で提出できれば良い。「印刷」や「印刷のイメージ」は重視されない場合もあり得る。

電子化手法

「書類の達人」の様な入力システムを採用する。最初に、作成側の効率化を図る。紙での提出にも対応する。次に、紙に印刷するのでは無く、電子データ(必要項目の羅列)で提出できるを進める。このためには、受領側に電子データで受け取るメリットを提案する。

注) *1:帳票

書式が一定で所定の枠内に必要事項を記入する書類を指す。定型書類とも呼ばれる。説明のために資料を添付することもあるが、本体は、必須項目に対して作成時に記載漏れや内容の間違い、受領後の確認漏れが無いように工夫された書式(例えば、罫線で囲む、選択する、コード化する)を採る。情報の伝達に際し意味を持つ最小単位は各項目である。必須項目は、個別に抽出しても意味を持つ場合がある。(施工(彰国社):1998年6月:「定型提出書類と電子申請/電子提出」から引用)

*2:「印刷」や「印刷のイメージ」

ワープロデータだけではなく、HTMLで記述された文書データも含む。

3.3 電子化手法の紹介

ここでは、HTMLの問題点と対比させたSGMLの特長とその手順、SGMLの問題点からみたXML誕生までの流れとXMLの特長、急速に注目を集めているPDFの特徴と役割について述べる。

(1) HTMLの問題点

HTMLのスタイルを対象とした共通マーク付けは、特定の処理システムに依存していない利点を持つが、文書内容と共通マーク付けからなる適切な構造を持ったデータとして記述されていることを保証している訳ではない。例えば、<H1>タグは、最大見出し文字を指定するための共通マーク付けであるため、表題や章名などで利用されることが多いが、論理的に文書構造上の「表題」を指定するものではない。表現目的として、他の文字でも利用される可能性もあり、さらに、「表題」の文字だけを確実に抽出することもできない。すなわち、スタイルを対象とした共通マーク付けは、文書データの変換や再利用を行おうとした場合には問題が多い。また、文書体裁と不可分な体系であるが故に、文書体裁の変更も容易ではない。スタイルを対象とした共通マーク付けが内包する問題点を解決するためには、もう一段階の形式化や抽象化が必要となる。文書構造の記述法を、印刷や表示などの処理系から完全に切り離し、独立した言語として定義する必要がある。このような背景の上で、論理構造を対象とし処理系非依存の共通マーク付けを行う文書記述言語として生まれたのが、SGMLである。

参考文献 高橋亨,「SGML/XMLによる文書記述」,Computer Today,1998.9

(2) SGMLとは

SGML(Standard Generalized Markup Language)は、1986年ISOが規格化した「情報の記述ルール」であり、直訳すれば、「標準汎用マーク付け言語」となる。

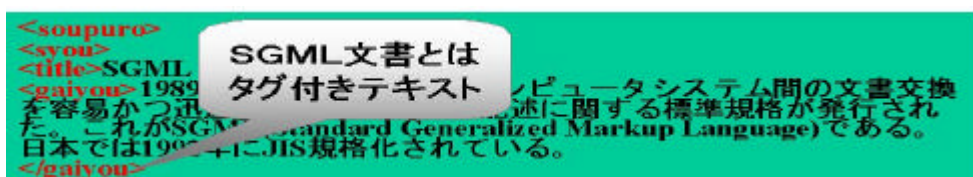


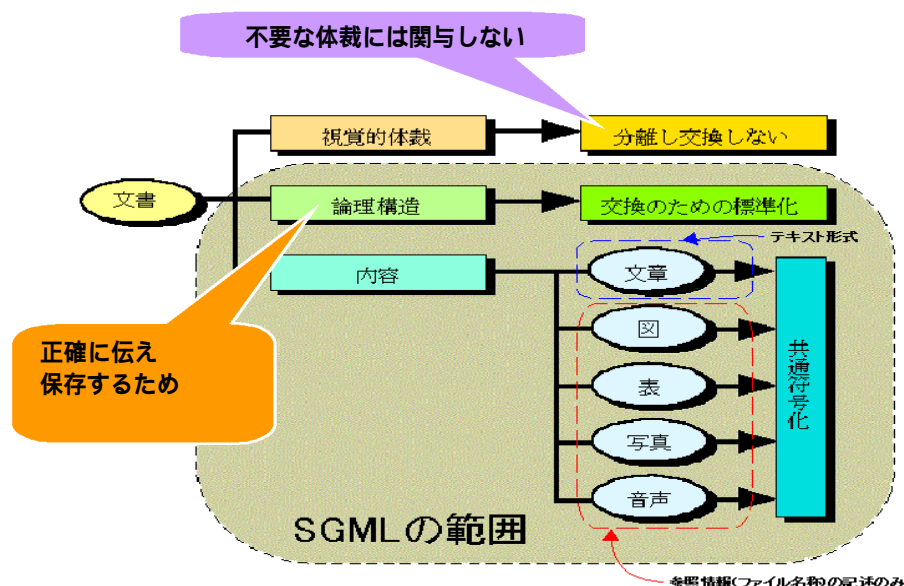
図-3.1 SGMLとは(1)

SGMLとはなんだろうか。文書情報を電子化して、発注者と受注者が文書をやり取りすることを想定してもらいたい。この場合、ワープロを使って文書を作成し電子データを相手側に送り、受け取った側はそのデータを自分のパソコンで見る。果たして見ることができる

であろうか。そう、同じアプリケーションを両者が使っていれば簡単である。しかし、私は一太郎を使っている、いや Word だ、というように異なるアプリケーションでは文書情報の伝達は難しい(異なるアプリケーションで作成したものを見る機能を設けたものもあるが)。これは、それぞれのアプリケーションで文書情報の保存の方法が異なっているため、言い換えれば標準化がなされていないための混乱である。ましてや、頻繁にアプリケーションがバージョンアップされる今日、同じアプリケーションでもこのような問題が生じている。これでは、情報を正確に伝えることができず、「みんなで情報を共有しましょう」という CALS とはほど遠いものとなってしまふ。もちろん「メモ帳」に代表されるテキスト文書であれば、異なるアプリケーションを使用していても問題なくやり取りできる。しかし、我々が目指すべきものは、単なるファイルのやり取りだけでなく、文章を構成するパーツがそれぞれ「タイトル」、「段落」、あるいはもっと具体的には打ち合わせ簿の「参加者」などのように何を意味するものであるかの情報も同時に持っている付加価値のある表現形式である。

これを達成するための方法に、文書情報の標準化技術として SGML が CALS 標準として採用されている。SGML は、文書情報を正確に伝えること、正確に保存すること、表示などの不要な体裁には関与しないことが特徴としてあげられる。

図-3.2 SGMLとは(2)



最近の SGML に関する動向として、最も顕著で影響力のある規格として XML(eXtensible Markup Language)が制定されている。これは、SGML で制定された数ある規約のうち使われなくなったものを廃止し、一部の情報の付加形態に関してある程度の制約をつけ、インターネット上で使用できるようにしたものである。

(3) 文書の SGML 化の手順

文書を SGML 化するためには、対象文書の構造化を行い、文書型定義 (DTD) を作成する。文書型定義とは、文書の章、節、箇条書きの取り決めを行うことで、通常我々が作成する文書の構造がどのようになっているか把握する必要がある。我々は、ワープロなどで文書を作成する時、どうも見栄え(体裁)ばかり気にして、どの章、節に、どのような内容を記述するか、あまり意識しなでいることが多い。例えば、帳票などのように定型化された文書でも、項目の書く順番が違って提出したりすると、提出先が受け取らないなどという経験があるのではなからうか。帳票の AA という項目には、工事名を書きなさいと決まっていれば、多少順番が異なっても問題はないはずである。この項目の取り決めをすることが、文書型定義を行うことになる。

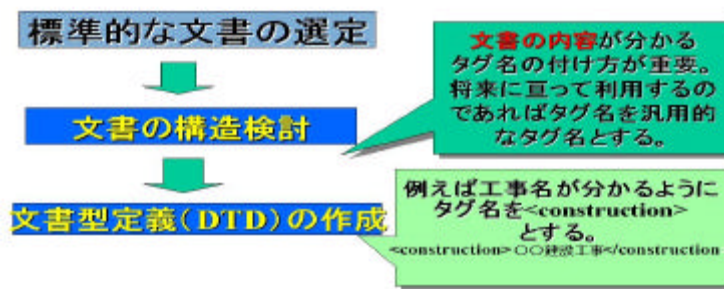


図-3.3 文書型定義

(4) 代表的な文書の SGML 化

標準化対象文書の調査・整理の結果から効果が高いとされた「工事打合簿」と「施工計画書」の文書型定義 (DTD) を作成した。

「工事打合簿」は、建設工事における各種文書の使用状況調査の結果、使用頻度が最も多く、また、施工時において施工方法などを決定する重要な情報である。この工事打合簿について、異なるワープロソフトで SGML 文書を介してデータのやり取りが可能ないようにプロトタイプを作成した。工事打合簿の書式は、将来に亘ってデータベースとしての利用を図るため「件名」を付加した。工事打合簿の作成ツールは、代表的なワープロソフトである「一太郎 8」と「Microsoft Word97」を利用した。一太郎 8 では、通常のユーザーが DTD を基に入力帳票を作成することが可能であり、簡単な帳票の SGML 化には便利である。さらに「一太郎 9 エクステンション」では、外部実体を外部ビューアの起動を必要とせずインライン表示する機能や表を作成表示する機能を追加する予定である。一方、Word97 の入力帳票の作成は、マクロ機能を利用したもので、通常のユーザーでは入力ツールの作成は難しいと考えられる。しかしながら、様々な DTD の要求に対してマクロ機能で対応が図れるので、より複雑な文書については、Word97 による方がよいと思われる。

施工計画書作成ツールは、複雑な文書となるため「Microsoft Word97」を基に、Word97 のテンプレート機能を利用して文書を入力し、マクロ機能で SGML のタグ変換を行い文書インスタンスを作成するプロトタイプを作成した。

この作成ツールは、フリーウェアとして平成 9 年度報告書の CD に使用説明書とともに載せている。

(5) SGML から XML へ

ここでは、SGML の問題点から見た XML 誕生までの流れと、XML が特徴をまとめる。SGML の問題点は、一言でいえば、「理論だけが先行し、実装を意識していない」ことに尽きる。例えば、タグの省略など実装時の自由度が大きいため、DTD を参照しなければ正しく解析できない。このために対応ソフトウェアの開発を難しくしている。

SGML が開発された時代は、PC(個人向けのコンピュータ)や GUI(Graphical User Interface)が普及する以前であり、手作業でプログラムを書く感覚で、文書データをコンピュータに入力する作業が当然のように思われていた。タグの省略など、入力手間の削減を許したのも、このような時代背景がもとになっている。また、ネットワークを通じ、システムから切り離された電子データが、独立して存在する運用環境が想像されなかった時代でもある。DTD 無しでは文書の内容を表示することさえもできない電子データである SGML は、通信機能を持つ PC が当然視される時代にあっては、ネットワークの負荷を高め効率的な処理を妨げる要因を内包している。

そして、SGML のスリム化し、さらに、データ交換を重視した共通マーク付けへの拡張したものとして、XML が誕生した。SGML でもその試行はあったが、タグで項目データが識別できることに着目して、単なる文書のデータではなく、可変長のデータを扱えるフォーマットとして利用できることが考え出された。ここに至り、「共通マーク付け」の考え方は、単なる文書記述言語の域を越え、データ交換に適用できるメタ言語を生み出した。

そして、前述した SGML の問題点を、インターネット上の利用を前提にして思想で見直し、再構築したものが XML である。この特徴を一言でいえば、「現状の処理系を睨んだ特定の処

理系非依存の共通マーク付け」となる。

参考文献 高橋亨,「SGML/XMLによる文書記述」,Computer Today,1998.9

(6) 電子の紙「PDF」

特定のワープロで作成した文書データは、同じ種類のワープロで新たな文書を作成するための雛形として配布するには適している。しかし、保管/提出のための電子データ、ここでは、承認を受けた文書データを想定した場合、それに要求される機能は単に見るだけで十分である。内容を修正できる機能などは必要とされない。「電子の紙」としてここで紹介するPDF(Portable Document Format)は、データ再利用の機能を持たず、印刷イメージを再現する機能に特化した電子データの形式である。

PDFはファイル形式の呼び名である。文書や図表などを作るアプリケーションを用いて普段と同じように作成して印刷、プリンターとしてAcrobatを選択すると、普通はプリンターに送られるはずの印刷のためのデータが、PDF形式のファイルとして保存される。このPDFファイルは、無償で配布の閲覧(表示/印刷)用ソフト「Acrobat Reader 3.0J」で、自由に読むことができる。Acrobat Readerは、Windows用だけでなく、Mac用も配布されており、プラットフォームの違いを乗り越えて文書や図表の交換ができる。まとめると、PDFの出現により、各自が使い慣れたアプリケーションや目的にあったアプリケーションを用いて、誰もが読める「電子の紙」を作れるようになった。

具体例として、定型提出書類でPDFを利用することを取り上げ、二つのステップに分けて説明する。

1)第1ステップ --- 用紙の配布

提出書類の書式は提出先固有の場合が多く、官公庁の場合には所管区域ごとに書式が異なり、また、知らない間に書式が変わるなどといった問題点がある。単に作成だけではなく、用紙入手についても煩雑感が高い。紙の形式による情報の受け渡しを前提にした上で、第1ステップとして、定型提出書類の用紙をPDFファイルとして作成し、このファイルをインターネットを通じて配布することを考える。

この用紙に手書きで必要な項目を書き込み、捺印/押印を行い、紙の形式で提出する。

2)第2ステップ --- 用紙の回収/保管

PDFファイルには、基本的に改ざんや書き込みができないものであるが、元データと別の領域にデータを書き込むことはできる仕様となっている。第2ステップとして、この領域を効果的に使うことによる、定型提出書類の用紙の回収と保管、すなわち、記載項目の電子的な回収と保管までを考える。

実際の建設生産の場には、電子データと手書き資料と印刷物など、さまざまな形式で情報が存在している。また、電子データは、作成するアプリケーションごとにさまざまなファイル形式があり、見るだけでも、作成時のアプリケーションとその環境が必要とされる。これらを、電子データとして一つの形式でまとめるとすれば、どのような方法を採用すべきか。電子データの利用目的を再現性の確保に限定すると、現状、PDFファイルとしてまとめることが最適な方法であろう。

データ項目を厳格に定義するSGMLと印刷イメージのみに特化したPDFは、相反するものではない。PDFで発信するとは、成果物として完成した段階である。共同で元データを作成するに際し、また、厳密な改定履歴を求める場合には、文書データ作成のためのガイドラインが必要となる。この時点では、PDFは何の役にも立たない、SGMLの出番となる。逆に、定型提出書類の内、受領する側が見るだけで終わる書類、すなわち、転記などの対象とならない書類はPDFに向いている。この書類を変更が不要な電子データであるとして捉えれば、電子認証と組み合わせたPDFファイルの形式が最適であろう。

参考文献：施工9807「電子の紙 --- PDF」