

XML 文書ワークフロー構築支援方式

- インターネットを用いた設計支援システムにおける XML 文書設計支援方式 -

今村 誠 森口 修 鈴木 克志

E-mail: {imamura, mog, suzuki}@isl.melco.co.jp

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

従来のワークフローシステムは、売上伝票や旅費精算などの主に固定的な事務帳票を対象としており、設計仕様書のように製品仕様に応じて文書構造が動的に変わるような基幹業務向け文書を扱うことができないという問題があった。本稿では、この問題を解決するために、ワークフローシステムの対象文書形式として XML(eXtensible Markup Language)を採用し、文書の論理設計情報に相当する文書型定義(DTD)を基にして担当分野毎の入力画面を設計していく XML 文書設計支援方式を提案する。本方式には、以下の特徴がある。(1)DTD では記述できなかった「ある記載項目の内容と、ある記載項目の繰り返し数との関係制約」、「記載項目の内容間の制約」、および「記載項目の内容と、前 WEB 画面入力値との制約」の記述により、入力チェック機能の開発が容易になる。(2)タグの階層構造と表の構造との対応関係パターンの類型化により、画面スタイルの開発が容易になる。(3)タグ毎にワークフロー担当分野でのアクセス権限を設定するだけで、担当分野毎に応じた画面を自動生成できる。

A Supporting Method for XML Document Workflow

- A Method of Supporting XML Document Design for Product Design Support System on the Internet -

Makoto IMAMUMA Osamu MORIGUCHI Katsushi SUZUKI

E-mail: {imamura, mog, suzuki}@isl.melco.co.jp

Mitsubishi Electric Corporation

Information Technology R & D Center Human Media Technology Dept.

Existing Workflow Systems mainly target fixed office forms such as an account of sails and travelling expenses, so they can not cope with dynamic engineering forms, such as product specification, whose document structure changes depending on the document contents. This paper presents a method of supporting XML document design, in which we design input forms depending on roles in the workflow system by putting document constraints and document style on its DTD (Document Type Definition). The features of this method are the following. (1) It can describe DTD unrepresentable constrains, such as "the relation between document contents and document structure", "the relation among document contents", and "the relation between document contents and inputted date on the former WEB form" for efficient development of input check functions. (2) It provides types of the relation between the structure of XML elements and that of table for efficient development of document style. (3) It can automatically generate forms depending on roles in the workflow system by setting access authorization for each XML element.

1 はじめに

製造企業においては、企業全体、さらには協業企業を含めてグローバルな生産効率化が求められている。そして、部門内では、ネットワークを利用して電子化文書を担当者間で伝送するワークフローシステムが導入されているが、従来のワークフローシステムは、売上伝票や旅費精算などの主に固定的な事務帳票を対象としており、設計仕様書のように製品仕様に応じて文書構造が動的に変わるような基幹業務向けの文書(プロダクティブ文書と呼ぶ)を扱うことができないという問題があった。

本稿では、この問題を解決するために、ワークフローシステムの対象文書形式として、インターネットでの次世代文書形式であるXML[1]を採用し、文書の論理設計情報に相当する文書型定義(DTD:Document Type Definition)を基にして担当分野毎の入力画面を設計していくXML文書設計支援方式を提案する。

ワークフローの対象文書形式としてのXML採用の主な理由は、「タグの追加・削除により、製品機種仕様改訂などに伴うフォーマット変更が容易であること」、「設計仕様書などのデータを、設計支援システムや工程管理システムなどの企業情報システムでの再利用が容易であること」、および「不特定多数の営業所や工場での利用を想定して、WEBブラウザからの文書参照が容易であること」の3点である。また、従来の問題点であったプロダクティブ文書への対応については、DTDは生成文法を規定できるので、「タグAまたはタグBのどちらかが出現する」、あるいは、「タグAは、1回以上繰り返し出現する」といった文書中のタグの出現順序を抽象的に規定できる柔軟性をもつことに注目した。すなわち、この柔軟性を生かすことによって、「データ入力時に業務内容に応じて入力項目を追加/削除するといった動的なデータ項目変更に対応しやすい」という従来の固定帳票にない特徴を提供していくことが容易になると考えた。

提案するXML文書設計は、文書の構造や内容に関する制約を記述する「文書制約設計」、文書制約を反映した画面のユーザインタフェースを規定する「文書スタイル設計」、およびワークフローでの文書データ活用に必要な要求を反映させる「ワークフロー文書活用設計」の3つの部分からなり、各々の設計手順を支援するためのツールを開発した。ツ

ルの特徴は、以下の3点である。

(1) DTDでは記述できなかった「ある記載項目の内容と、ある記載項目の存在有無や繰り返し数との関係制約」、「文書中の記載項目の内容間の制約」、および「記載項目の内容と、前WEB画面入力値との制約」を設定できるようにすることにより、入力チェック機能の開発が容易になる。

(2)XMLのタグの階層構造と、画面上の表の構造との対応関係のパターンを類型化することにより、多数の記載項目を表形式で提示する画面スタイルの開発が容易になる。

(3)タグ毎に、ワークフロー担当分野での参照権限や編集権限などのアクセス権の有無を設定するだけで、担当分野毎に応じた画面を自動生成できる。

本稿の構成は以下の通りである。2章では、設計支援ワークフローにおける文書処理への要求を分析した後、関連するW3C(World Wide Web Consortium)のXML標準や、従来ワークフロー用入力フォーム作成支援ツールの技術課題を検討する。3章では、2章で述べた要求や課題の解決方針、提案方式の全体構成、そして、提案方式の主要構成物である「フォーム記述言語」と「XML文書設計支援ツール」について述べる。4章では、提案方式の想定効果について述べ、5章と6章で、考察とまとめを述べる。

2 設計支援ワークフローにおける文書設計支援方式の課題

本章では、2.1節で、設計支援ワークフローにおける文書処理への要求を分析し、2.2節では、W3Cの関連XML標準や、固定的な事務帳票を対象とする従来ワークフロー用入力フォーム作成支援ツールが2.1節の要求を満たすかどうかを検討することにより、技術課題を抽出する。

2.1 設計支援ワークフローにおける文書処理への要求

エンジニアリング部門の設計支援ワークフローにおける文書処理では、設計対象物の構成や設計業務の内容を反映して、設計仕様書やデータシートなどの対象文書を処理するための要求機能が複雑になる。その複雑さは、文書の構造や内容の制約の複雑さ、文書の画面IFの複雑さ、ワークフローでの文書データ活用の複雑さの3点に集約できる。以下、3つの複雑さについて順に説明する。

(1) 構造や内容の制約の複雑さ(文書制約設計の問題)

対象文書の文書構造(DTD)が規定する文書中の記

載項目の出現順序や繰返し出現数などが設計対象物の構成を反映して階層木の構造をもつだけでなく、「ある記載項目の内容と、ある記載項目の存在有無や繰返し数との関係制約(内容・構造間制約)」、「文書中の記載項目の内容間の制約(内容間制約)」、および「記載項目の内容と、前 WEB 画面入力値との制約(文書外制約)」をもつという特徴がある。本稿では、これらの特徴をもつ文書を、従来のワークフローシステムが対象としていた売上传票や支払精算書などの単純な表や固定フォーマットをもつ事務帳票との違いを明確にする目的で、プロダクティブ文書と呼ぶ。ちなみに、プロダクティブとは基幹業務のことであり、売上传票や支払精算書などの管理業務(アドミニストラティブ)に対する用語である。設計仕様書の作成を支援する WEB 入力画面の開発では、ある仕様項目の値に応じて記載項目の数を動的に変えるなどの機能が必要になる。

図 1 に、プロダクティブ文書の具体例として、昇降機的设计仕様書の文書構造の一部を示す。昇降機的设计仕様書では、仕様が基本仕様、機械仕様、電気仕様、および意匠仕様などの仕様区分をもち、例えば以下の制約をもつ。

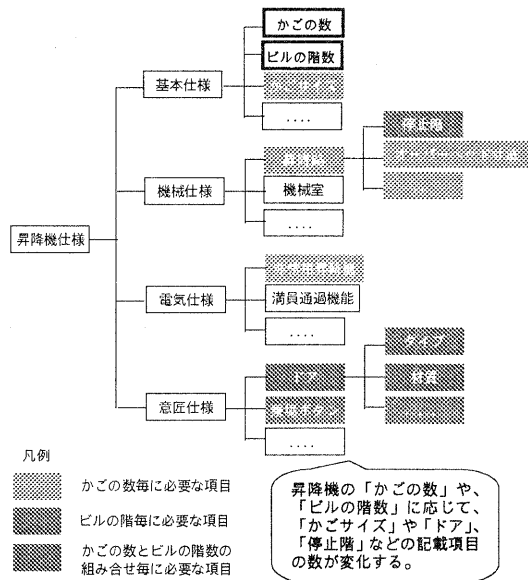


図 1 プロダクティブ文書の例

(i) 内容・構造間制約の例：昇降機のかごの数やビルの階数に応じて、基本仕様のかごサイズ、機械仕様の停止階、電気仕様の非常用昇降機、および意匠仕様のドアタイプなどの仕様項目の数が動的に変わらなくてはならない。

(ii) 内容間制約の例：「停止階数と階高を変えると、レールの本数が変わる」、また「自動音声は、MBS か SR(光電装置)かいずれかのセンサが無いと取り付けられない」など。

(iii) 文書外制約の例：「担当者名」や「製品番号」は、前の WEB 画面で入力され、それらの値を仕様書の対応する項目の内容に挿入しなければならない。本稿では、構造や内容の制約の複雑さに対処するための文書設計上の問題を「文書制約設計の問題」と呼ぶ。

(2) 画面 IF の複雑さ(文書スタイル設計の問題)

文書の構造や内容がもつ制約が複雑になるのに伴って、見やすさや入力チェック制御のための画面 IF への要求を満足する必要がある。

(i) 見やすさへの要求(階層構造の表による表示)

文書が階層構造を目次や表形式で見やすく表示したり、文書中のデータ項目間の関係をわかりやすく画面に提示する必要がある。ここでの主な問題は、「階層構造の表による表示」の問題である。これは、XML 文書中のデータを表形式で見やすく表示するためには、XML 文書タグの階層構造(木構造)を行と列からなる表の構造への写像を指定する必要があるが、この指定は必ずしも容易でない。

(ii) 入力チェック制御

文書入力画面では、(1)で述べた文書制約を満たしているかどうかを検証する入力チェック機能の起動タイミング、警告メッセージ表示、および警告メッセージから警告箇所へのハイパーリンク付与などの機能が必要になる。

本稿では、画面 IF の複雑さに対処するための文書設計上の問題を「文書スタイル設計の問題」と呼ぶ。

(3) ワークフローでの文書データ活用の複雑さ(ワークフロー文書活用設計の問題)

ワークフローシステムでは、「担当分野毎・アクション毎の画面提示」や「文書データの業務システム連携」の要求がある。

(i) 担当分野毎・アクション毎の画面提示

ワークフローシステムの文書画面では、文書にアクセスする人が担当者か管理者か、あるいは、機械設計者か電気設計者かといった担当分野に応じて、どの項目を見せるか、またどの項目を編集させるかが異なる場合がある。また、同じ担当分野でも、表示画面や入力画面によっても異なり、また、人手の内容チェックを支援するために、前版の文書との差分をわかりやすく表示する機能も要求される。

(ii) 文書データの業務システム連携

作成された文書データは、後工程の設計業務用システムなどのレガシーシステムに受け渡すために、データ変換が必要とされることがある。

本稿ではワークフローでの文書データ活用の複雑さに対処するための文書設計上の問題を「ワークフロー文書活用設計の問題」と呼ぶ。

2.2 XML 文書処理の技術課題

本節では、前節で述べた要求を実現するための関連技術として、既存の W3C 標準とワークフロー用の XML 入力フォーム作成支援ツールの現状と課題について述べる。以下では、フォームを「入力画面を生成するために必要な文書スタイルや文書制約を記述するもの」という意味で用いる。

2.2.1 W3C 標準

文書制約設計に関係する W3C 標準として、DTD と XML Schema について述べ、文書スタイル設計に関係する W3C 標準として、XSLT について述べる。

(1) DTD と XML Schema

XML 文書の構造や内容の制約を記述する言語として、DTD[1]と XML Schema([2] [3])がある。本稿では、DTD や XML Schema などの XML 文書の構造や内容の制約を記述する言語を文書スキーマと呼ぶ。XML Schema は、DTD に対して、データ型の記述や継承を導入などの制約記述能力が向上しているが、2.1 節(1)で述べた「内容・構造間制約」、「内容間制約」、および「文書外制約」は記述できない。従って、2.1 節の要求に応えるためには、その要求を実現する機能をプログラム等により実現する必要がある。

(2) XSLT(XSL Transformations)

XSLT[4]は、XML 文書の要素毎に変換規則を記述する言語であり、XML 文書から WEB ブラウザで表示する HTML 文書への変換規則を記述するためによく用いられる。本稿では、この HTML への変換規則記述をスタイルシートと呼ぶ。また、XML 文書とスタイルシートを入力として、XSLT の規則を解釈実行することにより出力文書を出力する処理系を XSLT プロセッサと呼ぶ。

XSLT を用いて XML 文書入力フォームを作成するには、2.1 節(1)で述べた文書制約をチェックするための JavaScript や JavaApplet などのチェックプログラムを参照し、また文書スタイルを実現するための HTML への変換規則を記述すればよい。しかし、XSLT 自体は、基盤となる構文を与えているだけで、

チェックプログラムの作成や、HTML への変換規則の作成を支援するしくみは提供していない。特に、XML 文書を表形式で表示する場合には、XML のタグと対応する HTML タグを指定する必要があり、これは必ずしも容易でない。

2.2.2 XML 入力フォーム作成支援ツール

ワークフロー用の XML 文書の入力フォーム作成支援ツールとしては、JetForm 社の FormFlow ([5])がよく知られている。本ツールでは、事務帳票を主な対象文書とし、紙印刷機能を重要視しており、入力画面イメージに対して文書制約などを付与していく設計手順(表示中心設計)を採用している。そのために、2.1 節で述べた複雑な文書処理要求に対応しにくいという問題がある。以下では、表示中心設計を概説した後、2.1 節の要求毎に技術課題を抽出する。

(1) 表示中心設計

表示中心設計では、文書スキーマ設計をした後、以下の順序で文書設計を進める。

① 文書スタイル設計

紙帳票と同じレイアウトをもった入力画面を作成する。

② スタイルと文書構造との写像

①の入力画面中の各々の入力枠に対して、XML 文書中のタグとの対応関係を指定する。

③ 文書制約設計

入力画面中の各々の入力枠に対して、入力チェックに必要な文書制約を指定する。

④ ワークフロー活用設計

担当分野に応じて画面をカスタマイズする。

(2) 技術課題

2.1 節の要求毎に、表示中心の文書設計の問題点について述べる。

(i) 文書制約設計での問題

・記載項目があらかじめ固定的に決まっている事務帳票ではなく、設計仕様書のように製品仕様に応じて記載項目が動的に変わるようなプロダクティブ文書に対する入力画面を作成できないという問題がある。

・入力画面中の入力枠に対して、入力チェックに必要な内容検証規則を設定していくので、設計仕様書のようにデータ項目が多くなると、データ項目間の関係が捉えにくくなるという問題がある。

(ii) 文書スタイル設計での問題

GUI インタフェースにより入力枠や選択メニュー

などのオブジェクトを貼り付けながら入力画面を作成していくので、表のように類似の記載項目が繰り返し規則的に出現する場合にはコピー貼り付けする手間が大きいという問題がある。

(iii) ワークフロー文書活用設計での問題

担当分野毎に入力画面を作成するので、共通部分が変わった場合でも、担当分野毎画面の数だけ修正作業が必要になり、担当分野毎に異なる入力画面の作成や保守が手間であるという問題がある。

3 XML 文書設計支援方式

本章では、3.1節で、2章で述べた文書処理への要求と技術課題を解決方針を説明し、続く、3.2節で、提案方式の全体構成について述べる。そして、3.3節と 3.4節で、提案方式の主要構成物である「フォーム記述言語」と「XML 文書設計支援ツール」の各々について述べる。

3.1 文書処理への要求と技術課題の解決方針

2章で述べた文書処理への要求と技術課題の解決方針は、「文書スキーマ中心設計の導入」、「XML フォーム記述言語の開発」、および「XML 文書設計支援ツールの開発」の3点である。

(1) 文書スキーマ中心設計の導入

プロダクティブ文書に対応するために、文書設計を表示中心ではなく、以下に示すような、「DTD や XML Schema が規定する文書構造の定義に対して、文書制約、文書スタイル、およびワークフロー活用に関する設計情報を付与していく」文書スキーマ中心の手順を進める。文書スキーマ中心設計とすることで、記載項目が動的に変わるようなプロダクティブ文書に対応できるようになるだけでなく、

文書制約が複雑な技術分野の文書設計や保守が容易になることが期待できる。

① 文書スキーマ設計

紙帳票がもつデータ項目として何があるかを XML のタグとして抽出し、「タグの出現順序」や「タグの繰り返し出現の有無」といった文書のデータ設計情報を DTD や XML Schema で表現する。

② 文書制約設計

文書スキーマ中の XML タグ毎に入力チェックに必要な文書制約を指定する。

③ 文書スタイル設計

XML タグ毎に、文書内容の配置、色、選択メニューやラジオボタンの選択肢、および見出しなどの入力画面スタイルとの対応関係を付与指定する。

④ ワークフロー活用設計

担当分野に応じて画面をカスタマイズする。

(2) XML フォーム記述言語の開発

2.1節の要求を満足する XML 文書設計情報を記述するための言語(XML フォーム記述言語)を開発し、その言語をもとに、WEB の入力フォームを生成するしくみ(WEB フォーム生成ソフトウェア)を開発する。

(3) XML 文書設計支援ツールの開発

文書制約設計、文書スタイル設計、ワークフロー文書活用設計からなる文書スキーマ中心の XML 文書設計を支援するツール(XML 文書設計支援ツール)を開発する。

3.2 提案方式の全体構成

以下、図 2 に従って、提案方式における XML 文書設計から WEB 入力画面生成までのデータと処理の流れを説明する。

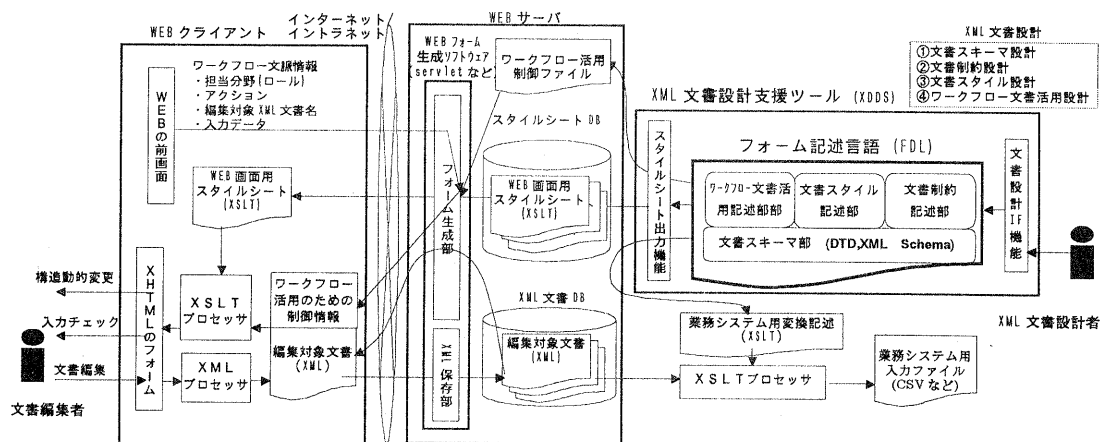


図 2 XML 文書設計支援方式の全体図

(1) XML 文書設計

XML 文書設計者は、XML 文書設計支援ツール(以後、XDDS ツールと呼ぶ)を用いて、文書スキーマ設計、文書制約設計、文書スタイル設計、ワークフロー文書活用設計を行う。XDDS ツールは、文書設計情報をフォーム記述言語(以後、FDL と呼ぶ)で表現し、スタイルシート出力機能により、WEB 画面用スタイルシート、業務システム用変換記述、およびワークフロー活用制御ファイルを出力する。ここで、ワークフロー活用制御ファイルとは、文書編集者の担当分野やアクション(表示か編集かなど文書へのアクセス種別)と、その担当分野やアクションに対応する WEB 画面用スタイルシートとの対応関係などを格納するファイルである。

(2) WEB サーバ側の入力フォーム生成処理

WEB サーバの WEB フォーム生成ソフトウェアは、WEB の前画面から、ワークフロー文脈情報として、文書編集者の担当分野、アクション、編集対象 XML 文書名、および製品 ID や担当者名などの次画面に受け渡すべき入力データを受け取る。そして、このワークフロー文脈情報とワークフロー活用制御ファイルから今回使用する WEB 画面用スタイルシートと編集対象 XML 文書を選定し、WEB クライアント側に送る(ただし、送付 XML 文書には、前画面入力データや前版との差分表示用情報などのワークフロー活用のための制御情報が付与されている)。

(3) WEB クライアント側の入力フォーム生成処理と文書編集

WEB クライアント側では、XSLT プロセッサは、XML 文書と WEB 画面用スタイルシートを入力として、XHTML 形式の入力フォームを出力し、WEB ブラウザが入力フォームを画面上に表示する。ここで、XML 文書中の記載項目の内容に応じた記載項目の動的変更や、入力チェックなどは、XDDS ツールが自動生成または参照する XHTML 上の JavaScript または JavaApplet によって実現されている。

また、WEB ブラウザ上の入力フォームの編集結果は、XML プロセッサを通じて編集対象 XML 文書の内容に反映され、編集が終了すると、WEB サーバ側に返送され、WEB サーバの XML 保存部によって、XML 文書ファイルとして、WEB サーバに保存される。

3.3 XML フォーム記述言語(FDL)

開発した XML フォーム記述言語は、文書スキーマ

記述部、文書制約記述部、文書スタイル記述部、およびワークフロー文書活用記述部からなる。以下では、順に説明する。

(1) 文書スキーマ記述部

DTD や XML Schema で記述できる XML 文書の構造や内容の制約を記述する部分である。市販の DTD や XML Schema 設計支援ツールとの併用を可能にするために、FDL は DTD や XML Schema をインポートできるようにしている。

(2) 文書制約記述部

現在の DTD や XML Schema で記述できない 2.1 節(1)で述べた「内容・構造間制約」、「内容間制約」、および「文書外制約」を記述する部分である。

「内容・構造間制約」は、要素の繰り返し数を変数やある要素の内容で指定することにより記述する。また、「内容間制約」は、要素名のパスを用いて内容を指し示すパス式と数値や文字列からなる等式や不等式をブール演算子(論理和、論理積、否定)で結合された制約式で記述する。「文書外制約」の記述は、「内容・構造間制約」や「内容間制約」の記述中で WEB の前画面の入力データを変数として参照できるようにすることにより実現する。

(3) 文書スタイル記述部

2.1 節(2)の要求に対応させて、「階層構造の表による表示」と「入力チェック制御」を中心に説明する。

(i) 「階層構造の表による表示」の問題への対処

「階層構造の表による表示」の問題に対処するために、表として表示する XML 文書タグの下位構造の「階層の深さ」と「タグの繰り返し出現の有無」に応じて、タグと表の行や列との対応関係を類型化することにより、その類型を指定するだけで、XML 文書を表形式で表示する XSLT 形式のスタイルシートを自動生成できるようにした[6]。類型としては、以下の4つがある。

表 1 XML タグと表との対応指定の類型

類型の名称	説明
1 階型	XML 文書タグが 1 階層の場合(あるいは、階層を無視して末端のタグのみを対象とする場合)
2 階線 0 型	XML 文書タグが 2 階層で、タグの繰り返し出現がない場合
2 階線 1 型	XML 文書タグが 2 階層で、1 階層目または 2 階層目のタグが繰り返し出現する場合
2 階線 2 型	XML 文書タグが 2 階層で、1 階層目も 2 階層目もタグが繰り返し出現する場合

(ii)入力チェック制御

警告メッセージ表示や、警告箇所と警告メッセージとのハイパーリンクなどのユーザインタフェース機能を実現するために、「(2)文書制約記述部」の各々の制約に対して、入力チェックの起動条件、起動タイミング、警告メッセージ内容、および警告メッセージ表示先などを指定する構文を提供した。

(4)ワークフロー文書活用記述部

2.1節(3)の要求に対応させて、「担当分野毎・アクション毎の画面提示」と「文書データの業務システム連携」を中心に説明する。

(i)担当分野毎・アクション毎の画面提示

XML 文書のタグ毎に、担当分野とアクションに応じて、そのタグの内容が「編集可」、「表示のみ」、「非表示」を指定するための構文を提供した。

(ii)文書データの業務システム連携

XML 文書のタグ毎に、業務システムの入力データフォーマットの対応する項目名を指定するための構文を提供した。

3.4 XML 文書設計支援ツール

XML 文書設計設計支援ツールは、文書スキーマ中心の XML 文書設計を支援するためのツールであり、図 3 の GUI インタフェースにより、XML のタグ毎に、文書制約の設定、文書スタイルの設定、およびワークフロー文書活用情報の設定をすることができる。

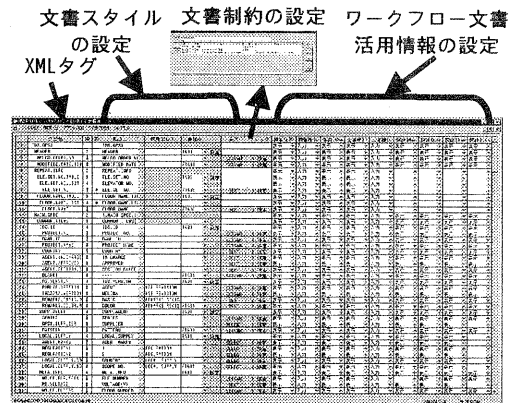


図 3 XML 文書設計支援ツールの画面例

以下、画面例を示す図に従って、各々の設計支援機能について説明する。

3.4.1 文書制約の設計支援機能

図 4は、図 3のチェックという列をクリックすると表示される画面であり、XML のタグ毎に、3.3 節(2)で述べた「内容・構造間制約」、「内容間制約」、および「文書外制約」を設定すると共に、3.3 節(3)(ii)で述べた「入力チェックの起動条件」、「起動タイミング」、「警告メッセージ内容」、「警告メッセージ表示先」などを指定できる。



図 4 文書制約設定インタフェースの画面例

3.4.2 文書スタイルの設定支援機能

図 5は、図 3において 3.3 節(3)の「タグと表との対応関係の種類」を指定した後に、文書スタイルの詳細を設定するインタフェースである。表の「枠線の太さ」、「セル間スペース」、「セル間マージン」、「見出し背景色」などを設定することができる。

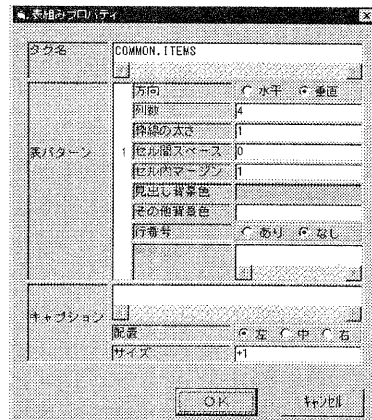


図 5 文書スタイル設定インタフェースの画面例

3.4.3 ワークフロー文書活用の設計支援機能

図 3の「ワークフロー文書活用情報の設定」の部分の列では、「営業設計」や「工事設計」といった担当分野名を表示している。ツール画面の行に対応する XML タグ毎に、列に対応する担当分野の人に対する内容へのアクセス権を「編集可」、「表示のみ」、「非表示」というメニューで指定することができる。

4 想定される効果

本稿で提案した XML 文書設計支援方式の全体構成

(主に 3.2 節で説明)と、XML 文書設計支援ツール(主に 3.4 節で説明)は、以下の効果を想定している。実システムによる 2.1 節の要求機能の実現性検証や、既存ツールとの開発工数定量比較などによる想定効果の実証は、今後の課題である。

(1) XML 文書設計支援方式の全体構成の効果

従来ツールが主な対象としていた売上传票や支払精算などの主に固定的な事務帳票ではなく、設計仕様書のように製品仕様に応じて文書構造が動的に変わるような基幹業務向けの文書(プロダクティブ文書)を扱うことができる。

(2) XML 文書設計支援ツールの想定効果

・文書制約の設計支援機能により、XML Schema で記述できなかった「内容・構造間制約」、「内容間制約」、および「文書外制約」を設定することができ、XML 文書入力画面の入力チェック機能の開発が容易になる。

・文書スタイルの設計支援機能により、XML 文書の内容の内容を表形式で表示するためのスタイルシートの開発が容易になる。

・ワークフロー文書活用の設計支援機能により、営業所の設計者、管理者、工場の設計者、および管理者毎に異なる入力画面や表示画面の開発が容易になる。

5 考察

本章では、関係形式データベースや帳票テーブルモデルによる方式と本稿提案方式との比較考察を行う。

(1) 関係形式データベース(RDB)中心方式との比較

本稿では、ワークフローシステムが伝送対象とする文書データは XML 形式で管理する方式(XML 中心方式)であるが、文書データを RDB で管理する方式(RDB 中心方式)が採用されることも多い。しかし、XML 中心方式では、以下に示すような意味で文書のデータ設計と表示処理に関して利点があると思われる。開発効率の定量的な比較は今後の課題である。

・データ設計に関しては、RDB 中心方式での正規化などの作業を含むテーブル設計より簡単であり、また、文書中の項目の追加削除などが容易である。

・表示処理に関しては、RDB 中心方式では正規化の結果、1つの文書中のデータ複数のテーブルにまたがるのに対して、XML は人間が参照する文書の形態に比較的近いので、XML を HTML に変換して表示するなどの表示処理の開発が容易である。

(2) 既存ワークフロー構築ツールにおける帳票テ

ブルモデルに基づく方式との比較

既存のワークフロー構築ツールの入力フォーム作成支援ツールには、GroupMax Form[7]のように、売上传票や旅費精算書などの事務帳票がもつべきデータ構造を帳票テーブルモデルとして抽象化して、帳票テーブルの項目定義を行うと帳票画面を自動生成するものがある。これらのツールは、(1)の RDB 中心方式と比較すると、対象とする帳票機能を限定することで、データ設計を簡単にするだけでなく、データ設計に対応した画面を簡単に作るしきみを提案してたものと位置づけられる。この方式だと、表の行を追加/削除するといった意味での、動的な処理が可能である。

本稿提案方式は、対象文書がもつべきデータ構造を表ではなく、DTD としている点で異なっており、2.1 節で述べたような複雑なエンジニアリングでの設計業務支援の場合に適していると思われる。

6 おわりに

設計仕様書のように製品仕様に応じて文書構造が動的に変わるような基幹業務向け文書を扱うことを特徴とする XML 文書設計支援方式を提案した。本方式の主な特徴は、「DTD や XML Schema では記述できなかった内容・構造間制約、内容間制約、および文書外制約記述の導入」、「タグの階層構造と表の構造との対応関係パターンの類型化」、および「タグ毎にワークフロー担当分野でのアクセス権限 IF の提供」により、WEB での XML 入力フォームの開発を容易にした点にある。

[参考文献]

- [1] Bray T., Paoli J. and Sperberg-McQueen C. M., "Extensible Markup Language (XML) 1.0", <http://www.w3.org/TR/1998/REC-xml-19980210> (1998)
- [2] Thompson H. S.他,XML Schema Part 1: Structures, <http://www.w3.org/TR/xmlschema-1/> (2000)
- [3] Biron P.V. and Malhotra A., XML Schema Part 2: Datatypes, <http://www.w3.org/TR/xmlschema-2/> (2000)
- [4] James Clark : XSL Transformations (XSLT) Version 1.0, <http://www.w3.org/TR/xslt> (1999)
- [5] JetForm 社 FORMFLOW99 ホームページ, <http://www.jetform.co.jp/nwcompu1.html>(2000)
- [6] 森口 修, 今村 誠, 鈴木 克志: XML 文書のスタイルシート生成方式, 情報処理学会デジタルドキュメント研究会資料 17-2, pp9-16(1999)
- [7] 日立製作所: Groupmax Form Version 3 ユーザーズガイド(1998)