(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第4868733号
(P4868733)

(45)発行日　平成24年2月1日(2012.2.1)　(24)登録日　平成23年11月25日(2011.11.25)

(51)Int.Cl.　F I
G06F 17/21 (2006.01)　G06F 17/21 570L
G06F 17/21 501T

請求項の数 6　(全 15 頁)

(21)出願番号　特願2004-340802(P2004-340802)
(22)出願日　平成16年11月25日(2004.11.25)
(65)公開番号　特開2006-154952(P2006-154952A)
(43)公開日　平成18年6月15日(2006.6.15)
審査請求日　平成19年11月14日(2007.11.14)

(73)特許権者　000001007
キヤノン株式会社
東京都大田区下丸子３丁目３０番２号
(74)代理人　100076428
弁理士　大塚　康徳
(74)代理人　100112508
弁理士　高柳　司郎
(74)代理人　100115071
弁理士　大塚　康弘
(74)代理人　100116894
弁理士　木村　秀二
(74)代理人　100130409
弁理士　下山　治
(74)代理人　100134175
弁理士　永川　行光

最終頁に続く

(54)【発明の名称】構造化文書処理装置及び構造化文書処理方法、プログラム

(57)【特許請求の範囲】
【請求項１】
第１構造化文書、第２構造化文書、それぞれが前記第１構造化文書に含まれる要素の要素名を指定する複数の第１ロケーション情報、及び、それぞれが前記第２構造化文書に含まれる要素の要素名を指定する複数の第２ロケーション情報に基づいて、第３構造化文書を構築する構造化文書処理装置であって、
前記第１構造化文書、前記第２構造化文書、前記複数の第１ロケーション情報、及び、前記複数の第２ロケーション情報を、前記複数の第１ロケーション情報のそれぞれと前記複数の第２ロケーション情報のそれぞれを対応付けて入力する入力手段と、
前記入力手段により入力された前記第１構造化文書に含まれる要素のうち、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を抽出する抽出手段と、
前記複数の第１ロケーション情報のそれぞれに対応付けられた前記複数の第２ロケーション情報のそれぞれで指定された要素名と要素名が一致する前記第２構造化文書中の要素のテキスト要素値として、前記抽出手段により前記第１構造化文書から抽出された、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を挿入して、前記第３構造化文書を構築する構造構築手段と
を備えることを特徴とする構造化文書処理装置。
【請求項２】
第１構造化文書、第２構造化文書、それぞれが前記第１構造化文書に含まれる要素の要

素名を指定する複数の第１ロケーション情報、及び、それぞれが前記第２構造化文書に含まれる要素の要素名を指定する複数の第２ロケーション情報に基づいて、第３構造化文書を構築する構造化文書処理装置における構造化文書処理方法であって、

入力手段が、前記第１構造化文書、前記第２構造化文書、前記複数の第１ロケーション情報、及び、前記複数の第２ロケーション情報を、前記複数の第１ロケーション情報のそれぞれと前記複数の第２ロケーション情報のそれぞれを対応付けて入力する入力工程と、

抽出手段が、前記入力工程で入力された前記第１構造化文書に含まれる要素のうち、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を抽出する抽出工程と、

構造構築手段が、前記複数の第１ロケーション情報のそれぞれに対応付けられた前記複数の第２ロケーション情報のそれぞれで指定された要素名と要素名が一致する前記第２構造化文書中の要素のテキスト要素値として、前記抽出工程で前記第１構造化文書から抽出された、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を挿入して、前記第３構造化文書を構築する構造構築工程と

を有することを特徴とする構造化文書処理方法。

【請求項３】

第１構造化文書、第２構造化文書、それぞれが前記第１構造化文書に含まれる要素の要素名を指定する複数の第１ロケーション情報、及び、それぞれが前記第２構造化文書に含まれる要素の要素名を指定する複数の第２ロケーション情報に基づいて、第３構造化文書を構築する構造化文書処理装置における構造化文書処理方法をコンピュータに実行させるプログラムであって、前記プログラムは、

前記第１構造化文書、前記第２構造化文書、前記複数の第１ロケーション情報、及び、前記複数の第２ロケーション情報を、前記複数の第１ロケーション情報のそれぞれと前記複数の第２ロケーション情報のそれぞれを対応付けて入力する入力手順と、

前記入力手順で入力された前記第１構造化文書に含まれる要素のうち、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を抽出する抽出手順と、

前記複数の第１ロケーション情報のそれぞれに対応付けられた前記複数の第２ロケーション情報のそれぞれで指定された要素名と要素名が一致する前記第２構造化文書中の要素のテキスト要素値として、前記抽出手順で前記第１構造化文書から抽出された、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を挿入して、第３構造化文書を構築する構造構築手順と

をコンピュータに実行させることを特徴とするプログラム。

【請求項４】

第１構造化文書、第２構造化文書、それぞれが前記第１構造化文書に含まれる要素の要素名を指定する複数の第１ロケーション情報、及び、それぞれが前記第２構造化文書に含まれる要素の要素名を指定する複数の第２ロケーション情報に基づいて、第３構造化文書を構築する構造化文書処理装置であって、

前記第１構造化文書、前記第２構造化文書、前記複数の第１ロケーション情報、及び、前記複数の第２ロケーション情報を、前記複数の第１ロケーション情報のそれぞれと前記複数の第２ロケーション情報のそれぞれを対応付けて入力する入力手段と、

前記入力手段により入力された前記第１構造化文書に含まれる要素のうち、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素の属性値を抽出する抽出手段と、

前記複数の第１ロケーション情報のそれぞれに対応付けられた前記複数の第２ロケーション情報のそれぞれで指定された要素名と要素名が一致する前記第２構造化文書中の要素の属性値として、前記抽出手段により前記第１構造化文書から抽出された、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素の属性値を挿入して、前記第３構造化文書を構築する構造構築手段と

を備えることを特徴とする構造化文書処理装置。

【請求項５】
第１構造化文書、第２構造化文書、それぞれが前記第１構造化文書に含まれる要素の要素名を指定する複数の第１ロケーション情報、及び、それぞれが前記第２構造化文書に含まれる要素の要素名を指定する複数の第２ロケーション情報に基づいて、第３構造化文書を構築する構造化文書処理装置における構造化文書処理方法であって、
入力手段が、前記第１構造化文書、前記第２構造化文書、前記複数の第１ロケーション情報、及び、前記複数の第２ロケーション情報を、前記複数の第１ロケーション情報のそれぞれと前記複数の第２ロケーション情報のそれぞれを対応付けて入力する入力工程と、
抽出手段が、前記入力工程で入力された前記第１構造化文書に含まれる要素のうち、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素の属性値を抽出する抽出工程と、
構造構築手段が、前記複数の第１ロケーション情報のそれぞれに対応付けられた前記複数の第２ロケーション情報のそれぞれで指定された要素名と要素名が一致する前記第２構造化文書中の要素の属性値として、前記抽出工程で前記第１構造化文書から抽出された、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素の属性値を挿入して、前記第３構造化文書を構築する構造構築工程と
を有することを特徴とする構造化文書処理方法。
【請求項６】
第１構造化文書、第２構造化文書、それぞれが前記第１構造化文書に含まれる要素の要素名を指定する複数の第１ロケーション情報、及び、それぞれが前記第２構造化文書に含まれる要素の要素名を指定する複数の第２ロケーション情報に基づいて、第３構造化文書を構築する構造化文書処理装置における構造化文書処理方法をコンピュータに実行させるプログラムであって、前記プログラムは、
前記第１構造化文書、前記第２構造化文書、前記複数の第１ロケーション情報、及び、前記複数の第２ロケーション情報を、前記複数の第１ロケーション情報のそれぞれと前記複数の第２ロケーション情報のそれぞれを対応付けて入力する入力手順と、
前記入力手順で入力された前記第１構造化文書に含まれる要素のうち、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素の属性値を抽出する抽出手順と、
前記複数の第１ロケーション情報のそれぞれに対応付けられた前記複数の第２ロケーション情報のそれぞれで指定された要素名と要素名が一致する前記第２構造化文書中の要素の属性値として、前記抽出手順で前記第１構造化文書から抽出された、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素の属性値を挿入して、第３構造化文書を構築する構造構築手順と
をコンピュータに実行させることを特徴とするプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、第１構造化文書、第２構造化文書に基づいて、第３構造化文書を構築する構造化文書処理装置、構造化文書処理方法、及びプログラムに関する。
【背景技術】
【０００２】
電子的な構造化文書情報の効果的な情報利用技術は、インターネットを含む広範な情報の交換・流通にとってきわめて重要な位置を占めている。例えば、XML（eXtensible Markup Language)に代表されるこれらの技術は，WEBによる情報環境に向けて開発されたものであり、構造化言語として標準化されている。
【０００３】
しかしながら、構造化文書構造の自動解析及び他の構造化文書への変換に関する体系的な言語処理技術は欠如している。従来、入力される構造化文書情報から必要な情報を抽出し、それらをまとめて、別の構造を持った構造化文書として出力するためには、あらかじ

め入力される構造化文書の構造と、出力する構造化文書の構造とを把握した上で、XSLT（XML Transformations)の作成や構造化文書から情報を抽出し、新たな別の構造をもった構造化文書として出力するプログラミングを行っていた。

【０００４】

上述の従来技術として、例えば、以下の特許文献１及び特許文献２に示されるものがある。

【特許文献１】特開２００４－３０５８２号公報

【特許文献２】特開２００４－３８３３４号公報

【発明の開示】

【発明が解決しようとする課題】

【０００５】

しかしながら、入力される構造化文書と、出力する構造化文書の構造があらかじめ分からなければ、入力される構造と出力する構造を意識したXSLTの作成や、プログラミングができず、入力される構造化文書から、必要な情報を抽出し、新たな別の構造をもった構造化文書として出力することは困難であった。

【０００６】

そこで、本発明は、上記の問題点を鑑みて、構造化文書を別の構造化文書に変換処理を行う際、予め、構造化文書の属性を把握してプログラミングを行うことなく、ロケーションパスの対応付けに基づいて変換処理を行うことを可能にする構造化文書処理技術を提供することを目的とする。

【課題を解決するための手段】

【０００７】

上記の目的を達成するべく、本発明にかかる構造化文書処理装置は、第１構造化文書、第２構造化文書、それぞれが前記第１構造化文書に含まれる要素の要素名を指定する複数の第１ロケーション情報、及び、それぞれが前記第２構造化文書に含まれる要素の要素名を指定する複数の第２ロケーション情報に基づいて、第３構造化文書を構築する構造化文書処理装置であって、

前記第１構造化文書、前記第２構造化文書、前記複数の第１ロケーション情報、及び、前記複数の第２ロケーション情報を、前記複数の第１ロケーション情報のそれぞれと前記複数の第２ロケーション情報のそれぞれを対応付けて入力する入力手段と、

前記入力手段により入力された前記第１構造化文書に含まれる要素のうち、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を抽出する抽出手段と、

前記複数の第１ロケーション情報のそれぞれに対応付けられた前記複数の第２ロケーション情報のそれぞれで指定された要素名と要素名が一致する前記第２構造化文書中の要素のテキスト要素値として、前記抽出手段により前記第１構造化文書から抽出された、前記複数の第１ロケーション情報のそれぞれで指定された要素名と要素名が一致する要素のテキスト要素値を挿入して、前記第３構造化文書を構築する構造構築手段と

を備えることを特徴とする。

【発明の効果】

【０００８】

本発明により、第１構造化文書から第２構造化文書に挿入するテキスト要素値、属性値を簡単に変更することが可能になる。

【発明を実施するための最良の形態】

【０００９】

以下、本発明の実施の形態を、具体例を用いて詳細に説明する。

【００１０】

（実施形態１）

図１Ａは、本発明の実施形態にかかる構造化文書処理装置の構成を示すブロック図である。図１Ａにおいて、１０１は、構造化文書処理装置における解析変換処理部を示す。解

析変換処理部１０１は、不図示のＣＰＵ等の制御ユニットの全体的な制御の下、解析変換処理を実行することが可能であり、処理結果を不図示のメモリに格納し、必要に応じてメモリからデータを読み出して処理を実行することが可能である。この解析変換処理部１０１の内部には、構造化文書（inputA.xml）１０２、構造化文書（outputB.xml）１０３やロケーションパスＡ、Ｂ（１０４、１０５）を入力処理するための入力部１０６、構造自動解析部１０７、ロケーションパス対応付け及びデータ変換処理部１１３、構造構築部１１４、出力部１１５が含まれる。

【００１１】

１０２は、入力部１０６に入力される構造化文書（inputA.xml）であり、１０３は、１０２の構造化文書を変換して出力するための文書の構造を記述する出力用の構造化文書（outputB.xml）（共にXMLデータであり、以下、「XMLデータ」と表現する）である。１０４、１０５は、それぞれの構造化文書１０２、１０３の内部データを指し示すためのロケーションパスを示す。

【００１２】

入力部１０６は、インターネットを含んだネットワークと接続し、ネットワークを介して構造化文書（例えば、XML、SGML、HTML等）及びロケーション情報（ロケーションパス）を受信することができるものとする。入力部１０６が、構造化文書（XMLデータ）１０２、１０３とロケーションパスＡ，Ｂ（１０４、１０５）を受信すると、構造自動解析部１０７は処理を開始する。

【００１３】

１０７は、構造自動解析部であり、入力部１０６が受信した構造化文書（１０２、１０３）及びロケーションパスＡ、Ｂ（１０４、１０５）の構造解析を行い、その構造解析に基づいて、データの選択、構造の再構築を行う。

【００１４】

図１Ｂは、構造自動解析部１０７の内部を具体的に示すブロック図である。図１Ｂ中、１０９は、構造解析・分解部であり、例えば、SAX(The Simple API for XML)エンジンを利用して、入力されたXMLデータをパースしていき、不図示のメモリ上にパースによって順番に検出されるXMLデータ構造に含まれる要素ごとにリストを作成し、要素の名前、要素タグの接続関係（親子関係、兄弟関係）、要素タグと要素タグで囲まれる"値"、属性の名前、属性の値を１つの要素に対して付加情報として対応づけ、XMLデータ解析リストとしてXMLデータの構造を分解した情報を格納する。

【００１５】

１０８はロケーションパス解析部であり、入力されたロケーションパスＡ，Ｂ（１０４、１０５）をそれぞれ解析し、不図示のメモリ上に作成したXMLデータ解析リストの内容とリンクして指し示すことが可能な形式でXMLデータ解析リストに格納する。

【００１６】

１１１はデータ選択・抽出部であり、XMLデータ解析リストから、格納してあるロケーションパスが指し示すデータのリスト番号と、そのリスト番号と親子関係あるいは兄弟関係など関連するリスト番号をすべて選択し、抽出する処理を行う。

【００１７】

１１２は構造構築部であり、XMLデータ解析リストから、リスト番号に基づいて、一度分解したXMLデータのみを抽出し、構築しなおして、例えば、XMLデータとして出力する処理を行う。

【００１８】

説明を図１Ａに戻し、１１３は、ロケーションパス対応付け・データ変換処理部であり、入力部１０６に入力されたロケーションパスＡ１０４とロケーションパスＢ１０５を１対1で対応付け、入力用XMLデータ(inputA.xml)１０２と、出力用XMLデータ（outputB.xml）１０３とのそれぞれのXMLデータ解析リストに格納されているデータを、ロケーションパスの対応付けによって変換し、変換した出力用XMLデータのXMLデータ解析リストを出力する。

【００１９】
１１４は構造構築部であり、XMLデータ解析リストから再びXMLデータとして構築するという１１２と同様の処理を行い、出力部１１５から、構築したXMLデータ（output.xml）１１６を出力する。
【００２０】
出力部１１５は、インターネットを含んだネットワークと接続し、ネットワークを介して構築した構造化文書（例えば、XML、SGML、HTML等）を他の装置に配信することができるものとする。
【００２１】
図２は、構造解析・分解部１０９の処理を説明する図である。図２（ａ）において、２０６は、入力用XMLデータ(inputA.xml)１０２の具体例を示す図である。このXMLデータ２０６が構造解析・分解部１０９に入力されると、図２（ｂ）に示すフローチャートが構造解析・分解部１０９により実行される。
【００２２】
図２（ｂ）のステップＳ２０１でXMLデータを入力し、ステップＳ２０２でSAXエンジンによって、XMLデータ２０６を先頭のタグからパースしていく。ステップＳ２０３で、メモリ上に、パースによって順番に出現するXMLデータの構造における要素ごとに番号を振ってリストを作成し、ステップＳ２０４で、要素の階層番号、要素の名前、namespace名、要素タグと要素タグで囲まれる"値"であるテキスト要素値、要素タグの親を示す親要素番号、要素タグの兄弟を表す1つ前兄弟番号、同じ階層で同じ要素名がある場合に判別できるよう、要素が出現した順番を示す要素出現順番号、要素が持っている属性の数、要素が持っている属性の名前と属性の値を全て、１つの要素に対して付加情報として対応づけを行い、ステップＳ２０５で、全ての要素に対し分解処理が終了したか否かを判断し、終了しない場合は、ステップＳ２０４の処理を続行し、分解処理が終わり次第パースを終了する（Ｓ２０５）。
【００２３】
図２（ｃ）の２０９は、XMLデータ解析リストの例を示す図であり、不図示のメモリ上にXMLデータの構造を分解した情報がXMLデータ解析リストとして格納される。
【００２４】
図３は、ロケーションパス解析部１０８の処理を説明する図であり、３１０は入力用XMLデータ１０２中のデータを指し示すロケーションパス１０４の具体例を示す図である。このロケーションパス３１０がロケーションパス解析部１０８に入力されると、図３（ｂ）に示すフローチャートがロケーションパス解析部１０８により実行される。
【００２５】
図３（ｂ）のステップＳ３０１で、ロケーションパス３１０を入力し、ステップＳ３０２で、ロケーションパス解析部１０８は、与えられたロケーションパスを１から順に解析して分解し、不図示のメモリ上にリストとして保存する。
【００２６】
図３（ｃ）の３０４は、分解されて、リストとして保存されているロケーションパスの例を示す図である。ロケーションパス３１０中において、例えば、"student[position()="2"]"という表現があれば、これは、同一階層番号上に同じ要素名が存在し、それを区別するために、その中でも2番目の要素という指定がされていることを示し、この場合、"student#2"という形でメモリ上にロケーションパスが格納される。また、例えば、"class/@number"という表現があれば、class@numberという形でロケーションパスが格納される。
【００２７】
図４Ａ、Ｂは、データ選択・抽出部１１１の処理内容を説明する図である。
【００２８】
図４Ａにおいて、４０１は構造解析・分解部１０９から出力されるXMLデータ解析リストを例示しており、また、図４Ｂはデータ選択・抽出部１１１の処理の流れを説明するフローチャートである。

【００２９】
まず、図４Ｂのステップ４０１で、構造解析・分解部１０９は、先に説明した図３（ｃ）の３０４のロケーションパスのリストを順に取得する。
【００３０】
次いで、ステップＳ４０２で、XMLデータ解析リストから、要素番号順にリスト（図４Ａの４００）を取得し、取得したリストから要素名と、取得したロケーションパスリストの最下層の要素名address（図３（ｃ）では、リストの１番目ならaddress、リストの２番目ならtel、リストの３番目ならname）を比較していき、XMLデータ解析リスト４０１内の要素名がロケーションパスで指定された要素名と同じ要素名であるか検索する。
【００３１】
ステップＳ４０３において、要素名が同じでなければ（Ｓ４０３-Ｎｏ）、次の要素番号のリストを取得するステップＳ４０２の処理まで戻る。同じであれば（Ｓ４０３-Ｙｅｓ）、処理をステップＳ４０４に進め、その要素番号を保存する。
【００３２】
ステップＳ４０５で、その要素番号に対するリスト内から親要素番号を検索する。
【００３３】
ステップＳ４０６において、親要素番号が「-1」であれば（Ｓ４０６-Ｙｅｓ）、ステップＳ４１９に処理を進め、ヒットしたリスト番号と、それに関係する保存したリスト番号を出力する。１番最初から親要素番号が「-１」であれば、ルートの要素名であり、関係する保存したリスト番号は無いことになる。
【００３４】
一方、ステップＳ４０６において、親要素番号が-1でなければ（Ｓ４０６-Ｎｏ）、処理をステップＳ４０７に進め、親要素番号が示すリストを取得する。
【００３５】
ステップＳ４０８で、取得してあるロケーションパスに次の要素名、例えば、最下層の要素名の親の要素名(リスト１であればaddressの前のsecurity)、が指定されているか検索する。
【００３６】
ステップＳ４０９において、指定の有無を判定し、指定されていなければ（Ｓ４０９－Ｎｏ）、その親要素番号の示すリストは必要なデータであるので、処理をステップＳ４１７に進め、取得したリスト番号を保存する。
【００３７】
ステップＳ４０９の判定において、指定されていれば（Ｓ４０９－Ｙｅｓ）、処理をステップＳ４１０に進め、親要素番号の示すリストの要素名と、そのロケーションパスで指定されている要素名を順（例security→teacher）に比較していく。
【００３８】
ステップＳ４１１で、ロケーションパスで指定された要素名と、XMLデータ解析リスト内の親要素番号を辿っていき、ひとつでも要素名の不一致があれば（Ｓ４１１-Ｎｏ）、その親要素番号で指定したリストデータは必要が無いと判断し、次の要素番号のリストを取得する処理に戻る。要素名が一致していれば（Ｓ４１１-Ｙｅｓ）、ステップＳ４１２でロケーションパスで指定されている要素名に、データとして「#N(N:数値)」、が付加されているか確認する（Ｓ４１２）。
【００３９】
ステップＳ４１３で、「#N」が付加されているか判断し、付加されていれば（Ｓ４１３－Ｙｅｓ）、#NのNで指定された番号と、ステップＳ４１４で要素出現順番号を検索した番号をステップＳ４１５で比較する。
【００４０】
ステップＳ４１６で番号が一致しているか判断し、一致していなければ（Ｓ４１６-Ｎｏ）、その親要素番号で指定したリストデータは必要が無いと判断し、次の要素番号のリストを取得する処理に戻る。

【００４１】
一致していれば（Ｓ４１６－Ｙｅｓ）、その親要素番号を保持し（Ｓ４１７）、そのリストデータの親要素番号を取得（Ｓ４１８）する。
【００４２】
そして、再びステップＳ４０６に処理を戻し、親要素番号が-1でないかの判断の処理からまた繰り返す。
【００４３】
最終的にロケーションパスで指定されたパスで、XMLデータ解析リスト内でヒットした要素番号とそれに関係する要素番号を保持する。具体的には、ロケーションパス（図３（ｃ）の３０４）で１番目のリストのロケーションパス（teacher/security/address）にヒットする番号は、全部で、５→４→２→１→０の順でヒットし、これらが抽出される番号となる。１つのロケーションパスに対して処理が終われば、2番目のロケーションパス（class/student[position="2"]/security/tel）を取得し、同様の処理を繰り返す。2番目のリストのロケーションパスにヒットする番号は、１６→１４→１２→１→０の順でヒットする。３番目のリストのロケーションパス(name)にヒットする番号は、nameという指定だけなので、３→２→１→０、８→７→１→０、１３→１２→１→０という順番でヒットする。
【００４４】
すべての処理が終わった時点で、抽出した番号で重なるものがあれば、重複した番号は1つだけ残し、その他はすべて削除し、最終的に残った番号を出力する。その結果、データ選択・抽出部１１１が選択する要素番号のリストは（１）式のようになる。
【００４５】
要素番号のリスト：0,1,2,3,4,5,7,8,12,13,14,16　・・・（１）
これが、すべてのロケーションパスで指定されたデータに関連する要素番号となる。
【００４６】
次に、構造構築部１１２の処理を図５Ａ、Ｂを参照しつつ説明する。
【００４７】
図５Ａは、構造構築部１１２の処理内容を説明するフローチャートである。まず、ステップＳ５０１において、XMLデータ解析リスト（図４Ａを参照）を取得する。
【００４８】
そして、ステップＳ５０２で、XMLデータ解析リストから抽出する要素番号を保持したリスト（（１）式を参照）を取得する。
【００４９】
ステップＳ５０３で、XMLデータ解析リスト４０１（図４Ａ）から要素番号のリストデータ（（１）式）を順に取得する。
【００５０】
ステップＳ５０４で、抽出したリストデータの階層番号を取得し、保持する。
【００５１】
ステップＳ５０５で、前回保持した階層番号と今回保持した階層番号を比較する。ステップＳ５０６で、初回の処理でない、もしくは、前回保持した階層番号＞＝今回取得した階層番号の判断が偽ならば（Ｓ５０６－ＮＯ）、処理をステップＳ５０７に進め、リストデータから要素名を取得し、「＜要素名＞」の開始タグの形にして（'＜''＞'記号を付加して）文字列（図５Ｂの５２０）として格納する。
【００５２】
ステップＳ５０８で、リストデータ（（１）式）に基づいて、XMLデータ解析リスト中に属性名、属性値で該当するものがあれば順に取得し、開始タグの中に追加する形で、文字列（図５Ｂの５２０）に追加する。
【００５３】
ステップＳ５０９で、リストデータ（（１）式））に基づいて、XMLデータ解析リスト中にテキスト要素値で該当するものがあれば取得し、開始タグの後に、文字列（図５中５

２０）として追加する。

【００５４】

ステップＳ５１０で、取得した要素名を「＜/要素名＞」　の終了タグの形にして、スタック（図５Ｂの５２１）に格納(push)する。

【００５５】

ステップＳ５０６の判断で真ならば（Ｓ５０６－Ｙｅｓ）、処理をステップＳ５１５に進め、スタック（図５Ｂの５２１）に格納してある終了タグを１回取り出して（pop）、文字列（図５Ｂの５２０）に格納する。

【００５６】

ステップＳ５１１で、XML解析リスト（図４Ａ）から要素番号のリスト（（１）式）に記述されている要素番号のデータ分、データの抽出をしていなければ（Ｓ５１１－Ｙｅｓ）、ステップＳ５０３に処理を戻し、抽出していないデータに対して同様の処理を繰り返す。

【００５７】

一方、リストデータ（（１）式）に記述された要素番号のデータをすべて抽出し終わったら（Ｓ５１１－Ｎｏ）、処理をステップＳ５１２に進め、図５Ｂのスタック５２１に終了タグがまだ残っているか確認する。

【００５８】

ステップＳ５１３で、終了タグがスタックに残っていれば（Ｓ５１３－Ｙｅｓ）、スタックから終了タグを取得(pop)し、文字列（図５Ｂの５２０）に追加する。

【００５９】

終了タグが残っていなければ（Ｓ５１３－Ｎｏ）、処理をステップＳ５１４に進め、格納した文字列（図５Ｂの５２０）を出力する。

【００６０】

データ選択・抽出部１１１は、ＸＭＬ解析リスト（図４Ａの４０１）から、要素番号のリスト（（１）式）に記述されている要素番号のデータだけを抽出して、文字列を組み立て最終的に出力することができ、その結果、データ選択・抽出部１１１は、入力されたＸＭＬデータの構造を崩さず、ロケーションパスによって選択されたデータだけを抽出して、図５ＢにおけるＸＭＬデータ５２２を出力することができる。

【００６１】

図６は、ロケーションパス対応付け・データ変換部１１３の処理内容を説明する図である。これまでは、入力されたＸＭＬデータが１つで、そのデータからロケーションパスで選択したデータだけを抽出してＸＭＬデータとして構造を崩さずに構築しなおして出力するところまで述べたが、前記ロケーションパス対応付け＆データ変換処理部を、前記構造構築部の処理が行われる前に加えることで、前記入力データの入力用としたXMLデータ（以下入力用ＸＭＬデータとする）、前記入力データの出力用としたXMLデータ（以下出力用ＸＭＬデータとする）をそれぞれ読み込み、入力用ＸＭＬデータからロケーションパスで選択したデータを、出力用ＸＭＬデータのロケーションパスで選択した要素に挿入することで、異なる構造をしたＸＭＬデータによるデータの変換が可能となる。以下その流れを説明する。

【００６２】

図６の６０１は、入力用ＸＭＬデータであり、これまでの説明の中で使用してきた入力用XMLデータ(inputA.xml)２０６と同様の構造とデータ内容とを有する。６０２は、出力用ＸＭＬデータであり、既にテキスト要素値が一部格納してあり、入力用ＸＭＬデータとは構造が異なっている。

【００６３】

６０３は、入力用ＸＭＬデータに対するロケーションパスを示しており、６０４は、出力用ＸＭＬデータに対するロケーションパスである。

【００６４】

ロケーションパス対応付け・データ変換部１１３では、この両者のロケーションパスを

６０５のようにリストの一番目同士、二番目同士という形で、1対1で対応付ける。このロケーションパスの対応付けの情報を使って、ロケーションパス対応付け・データ変換部１１３におけるデータ変換処理は、不図示のメモリ上に保持してある入力用ＸＭＬデータ６０１を読み込み、構造解析の結果得られるＸＭＬデータ解析リスト６０６から、それぞれ、ロケーションパスで選択されたデータを示す要素番号を検索し、出力用ＸＭＬデータ６０２の解析から得られるＸＭＬデータ解析リスト６０７へコピーする（６０８）。その結果、ロケーションパス対応付け・データ変換部１１３は、ＸＭＬデータ解析リスト６０９を求め、出力する。

【００６５】

そして、ロケーションパス対応付け・データ変換部１１３は、このデータの要素番号をすべてリストデータ６１０として、構造構築部１１２に入力する。構造構築部１１２は、ＸＭＬデータ解析リスト６０９と、要素番号のリストデータ６１０を受け取り、図５Ａで説明したリストデータに対応する抽出処理を行うことで、ＸＭＬデータ６１２（図６（ｅ））を出力する。具体的には、入力用ＸＭＬデータ６０１内のデータである、「Ａさん」、「東京都」、「045-3333-3333」というテキスト要素値が、出力用ＸＭＬデータ６０２のそれぞれ、ロケーションパスで選択されたテキスト要素値へ格納される。

【００６６】

この結果、構造の異なるＸＭＬデータ６０１からロケーションパスで指定したデータを、別の構造を持ったＸＭＬデータ６０２のロケーションパスで指定した領域にデータを挿入することができ、結果的に、ＸＭＬの構造を変換して出力することが可能になる。従って、予め入力用ＸＭＬデータと出力用ＸＭＬデータの構造を把握していなくても、ＸＭＬデータの解析から自動的に変換処理を行うことが可能になるので、動的なデータの交換、動的なＸＭＬの構造変換が可能になる。

【００６７】

尚、本実施形態では、テキスト要素値の変換しか行っていないが、属性値に対してもロケーションパスの指定で、構造の異なるデータ間で同様の変換処理を行うことが可能になる。

【００６８】

また、ロケーションパスの関連付けを示すデータをネットワークを介して、他の装置と交換することも可能であり、異なる構造を有する構造化文書のやり取りが、ネットワーク上でも可能になる。

【００６９】

以上説明したように本実施形態によれば、構造化文書を別の構造化文書に変換処理を行う際、予め、構造化文書の属性を把握してプログラミングを行うことなく、ロケーションパスの対応付けに基づいて変換処理を行うことが可能になる。

【００７０】

（他の実施形態）

本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、上述の構造化文書処理装置に供給し、その装置のコンピュータ（またはＣＰＵやＭＰＵ）が記憶媒体に格納されたプログラムコードを読出し実行することによっても、達成されることは言うまでもない。プログラムコードの格納は、クライアントコンピュータに限定されるものではなく、例えば、サーバとして機能するコンピュータに記憶されておくことも可能である。

【００７１】

この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【００７２】

プログラムコードを供給するための記憶媒体としては、例えば、フレキシブルディスク，ハードディスク，光ディスク，光磁気ディスク，ＣＤ－ＲＯＭ，ＣＤ－Ｒ，ＤＶＤ、磁

気テープ，不揮発性のメモリカード，ＲＯＭなどを用いることができる。

【００７３】

また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているＯＳ（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【００７４】

さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるＣＰＵなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【図面の簡単な説明】

【００７５】

【図１Ａ】本発明の実施形態にかかる構造化文書処理装置の構成を示すブロック図である。

【図１Ｂ】構造自動解析部１０７の内部を具体的に示すブロック図である。

【図２】構造解析・分解部１０９の処理を説明する図である。

【図３】ロケーションパス解析部１０８の処理を説明する図である。

【図４Ａ】構造解析・分解部１０９から出力されるXMLデータ解析リストを例示する図である。

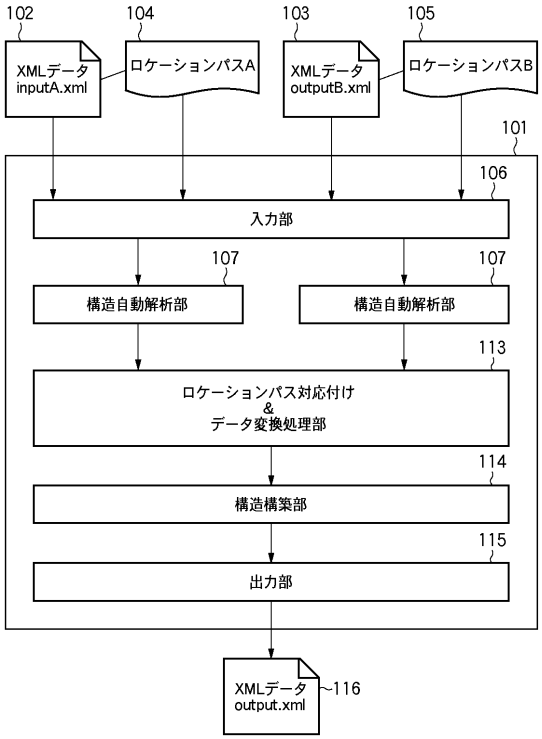
【図４Ｂ】データ選択・抽出部１１１の処理の流れを説明するフローチャートである。

【図５Ａ】構造構築部１１２の処理内容を説明するフローチャートである。

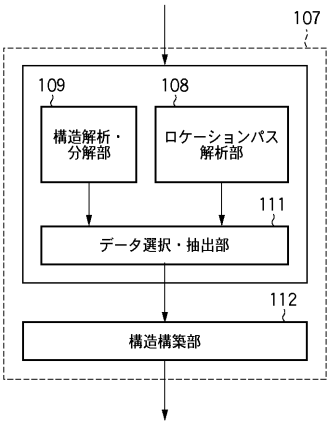
【図５Ｂ】構造構築部１１２の処理の内容を説明する図である。

【図６】ロケーションパス対応付け・データ変換部１１３の処理内容を説明する図である。

【図１Ａ】



【図１Ｂ】



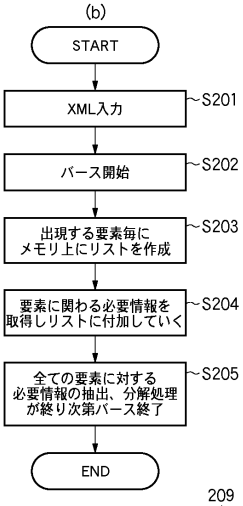
【図２】

(a)

206

```
<?xml version="1.0" encoding="Shift_jis"?>
<list>
 <class number="1">
  <teacher>
   <name>Aさん</name>
    <security>
     <address>東京都</address>
     <tel>03-1111-1111</tel>
    </security>
  </teacher>
  <student>
   <name>Bさん</name>
    <security>
     <address>神奈川県</address>
     <tel>045-2222-2222</tel>
    </security>
  </student>
  <student>
   <name>Cさん</name>
   <security>
    <address>神奈川県</address>
    <tel>045-3333-3333</tel>
   </security>
  </student>
 </class>
</list>
```

(b)



(c)

209

要素番号	階層番号	要素名	name space名	テキスト要素値	親要素番号	1つ前兄弟番号	要素出現順番号	属性数	属性名	属性値
0	0	list	null		−1	0	1	0		
1	1	class	null		0	0	1	1	number	1
2	2	teacher	null		1	0	1	0		
3	3	name	null	Aさん	2	0	1	0		
4	3	security	null		2	3	1	0		
5	4	address	null	東京都	4	0	1	0		
6	4	tel	null	03-1111-1111	4	5	1	0		
7	2	student	null		1	2	1	0		
8	3	name	null	Bさん	7	0	1	0		
9	3	security	null		7	8	1	0		
10	4	address	null	神奈川県	9	0	1	0		
11	4	tel	null	045-2222-2222	9	10	1	0		
12	2	student	null		1	7	2	0		
13	3	name	null	Cさん	12	0	1	0		
14	3	security	null		12	13	1	0		
15	4	address	null	神奈川県	14	0	1	0		
16	4	tel	null	045-3333-3333	14	15	1	0		

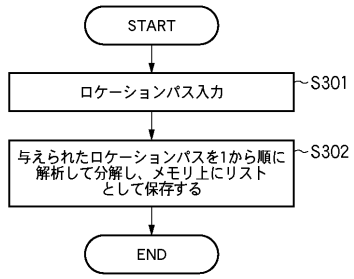
【図３】

(a)

310

```
1 teacher/security/address
2 class/student[position()='2']/security/tel
3 name
```

(b)



(c)

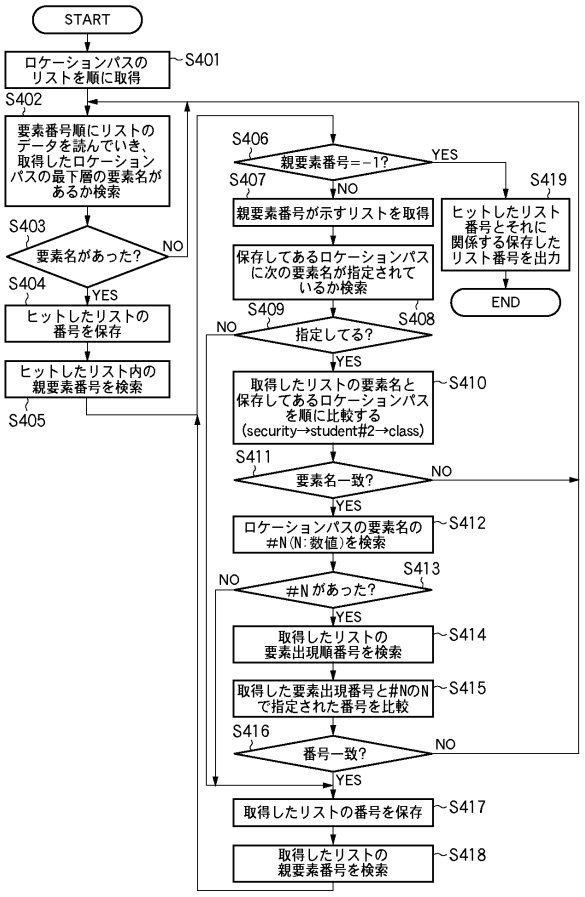
304

```
1 teacher, security, address
2 class, student#2, security, tel
3 name
```

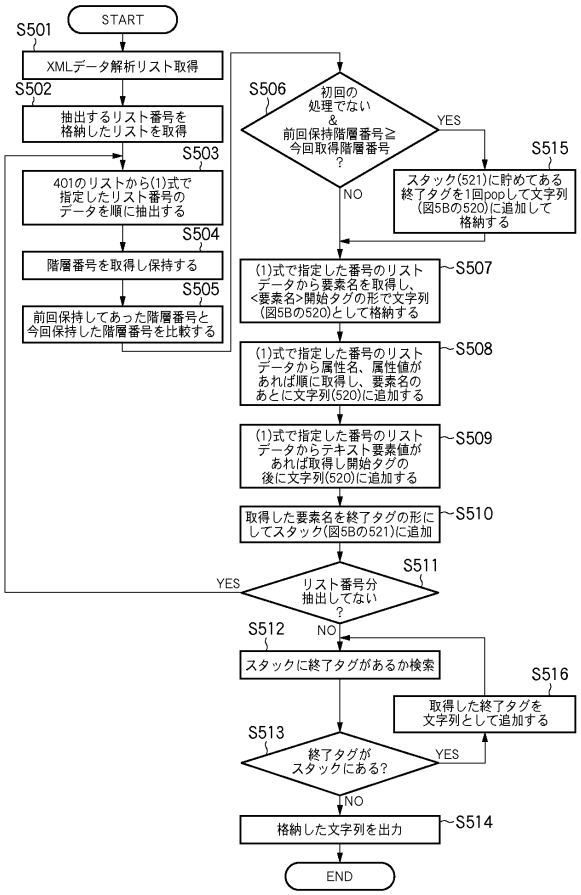
【図４Ａ】

要素番号	階層番号	要素名	name_space名	テキスト要素値	親要素番号	1つ前の兄弟番号	要素出現順番号	属性数	属性名	属性値
0	0	list	null		−1	0	1	0		1
1	1	class	null		0	0	1	1	number	
2	2	teacher	null		1	0	1	0		
3	3	name	null	Aさん	2	0	1	0		
4	3	security	null		2	3	1	0		
5	4	address	null	東京都	4	0	1	0		
6	4	tel	null	03−1111−1111	4	5	1	0		
7	2	student	null		1	2	1	0		
8	3	name	null	Bさん	7	0	1	0		
9	3	security	null		7	8	1	0		
10	4	address	null	神奈川県	9	0	1	0		
11	4	tel	null	045−2222−2222	9	10	1	0		
12	2	student	null		1	7	2	0		
13	3	name	null	Cさん	12	0	1	0		
14	3	security	null		12	13	1	0		
15	4	address	null	神奈川県	14	0	1	0		
16	4	tel	null	045−3333−3333	14	15	1	0		

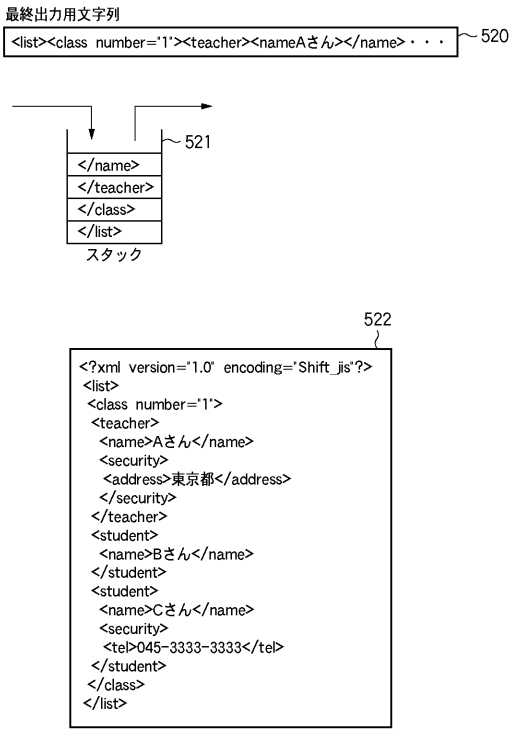
【図４Ｂ】



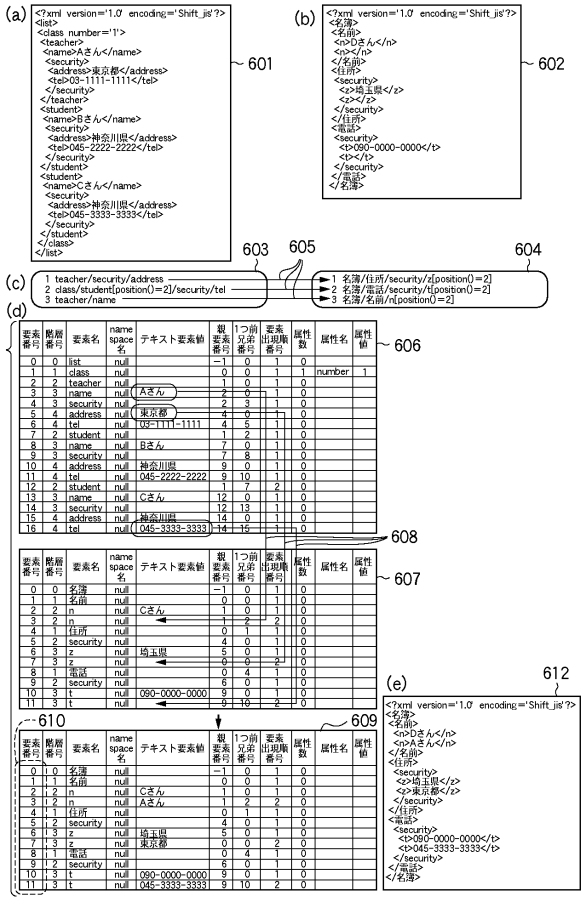
【図５Ａ】



【図５Ｂ】



【図６】


(a)
601
(b)
602
(c)
1 teacher/security/address
2 class/student[position()=2]/security/tel
3 teacher/name
1 名簿/住所/security/z[position()=2]
2 名簿/電話/security/t[position()=2]
3 名簿/名前/n[position()=2]
603
604
605
(d)
606
607
608
609
610
(e)
612

フロントページの続き

(72)発明者　岩崎　晋吾
東京都大田区下丸子３丁目３０番２号　キヤノン株式会社内

審査官　岩田　淳

(56)参考文献　特開２００２－１８３１１６（ＪＰ，Ａ）
吉田　晃伸，Ｂｌｏｇサイト構築で理解するXMLデータベース，日経Ｌｉｎｕｘ，日本，日経ＢＰ社，２００４年１１月　８日，第6巻，第11号，P.133-138
薬師寺　国安，外1名，動いてなんぼのXML，ＯＰＥＮ　ＤＥＳＩＧＮ，日本，ＣＱ出版株式会社，２００１年１２月　１日，第8巻，第12号，P.144-159

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１７／２１
ＪＳＴＰｌｕｓ（ＪＤｒｅａｍＩＩ）