

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5114547号
(P5114547)

(45) 発行日 平成25年1月9日(2013.1.9)

(24) 登録日 平成24年10月19日(2012.10.19)

(51) Int. Cl. F I
G 0 6 F 17/30 (2006.01)
 G 0 6 F 17/30 1 4 0
 G 0 6 F 17/30 3 4 0 D
 G 0 6 F 17/30 1 1 0 F

請求項の数 7 (全 20 頁)

<p>(21) 出願番号 特願2010-500851 (P2010-500851) (86) (22) 出願日 平成20年9月2日(2008.9.2) (65) 公表番号 特表2010-522394 (P2010-522394A) (43) 公表日 平成22年7月1日(2010.7.1) (86) 国際出願番号 PCT/KR2008/005140 (87) 国際公開番号 W02009/031791 (87) 国際公開日 平成21年3月12日(2009.3.12) 審査請求日 平成21年9月24日(2009.9.24) (31) 優先権主張番号 10-2007-0088758 (32) 優先日 平成19年9月3日(2007.9.3) (33) 優先権主張国 韓国(KR)</p>	<p>(73) 特許権者 599028364 電子部品研究院 KOREA ELECTRONICS T ECHNOLOGY INSTITUTE 大韓民国京畿道城南市盆唐区野塔洞68番 地 68 Yatap-dong, Bund ang-gu, Seongnam-si , Gyeonggi-do 463-8 16, Republic of Kor ea (74) 代理人 100089266 弁理士 大島 陽一</p>
--	--

最終頁に続く

(54) 【発明の名称】 SOAPオペレーションを用いた問合せコンテンツサービス方法

(57) 【特許請求の範囲】

【請求項1】

サーバ及びクライアントを含むネットワークにおいて、シンプル・オブジェクト・アクセス・プロトコル(以下、「SOAP」と言う)オペレーションを用いて検索コンテンツを提供する方法であって、

前記クライアントが、算術演算を含む複合演算で多様なコンテンツ検索を要求可能にするための要求エレメントフィールド類型が追加された前記SOAP問合せオペレーションの要求メッセージを伝送するステップと、

前記要求メッセージが問合せ要求結果値を指定する前記要求エレメントフィールド類型を含む場合に、

前記サーバが、前記要求エレメントフィールド類型において指定されたテーブルフィールドに該当する問合せ結果表現値を抽出するステップと、

前記サーバが、前記問合せ結果表現値を問合せ結果エレメントに追加してSOAPオペレーションの応答メッセージにより伝送するステップと、
 を含み、

前記要求エレメントフィールド類型は、算術演算の被演算子値と該被演算子値のタイプ情報とを含むコンスタントパリュウ類型を含み、

前記抽出するステップは、前記サーバが前記被演算子値の前記タイプ情報をチェックするステップを含むことを特徴とするSOAPオペレーションを用いた問合せコンテンツサービス方法。

【請求項 2】

前記応答メッセージは、maxPage、maxItem、freeText、resource、outputName、Field、AggregateID、Field Combination、SortBy、GroupBy、GroupbyField、Aggregate、Thumbnail 及び typeName の内少なくとも一つの間合せ結果エレメントを含むことを特徴とする請求項 1 に記載の SOAP オペレーションを用いた間合せコンテンツサービス方法。

【請求項 3】

前記応答メッセージは、間合せ要求したコンテンツによる間合せ結果をソーティング及び/またはグルーピングすることを特徴とする請求項 1 に記載の SOAP オペレーションを用いた間合せコンテンツサービス方法。

【請求項 4】

前記応答メッセージは、次のように定義されることを特徴とする請求項 1 に記載の SOAP オペレーションを用いた間合せコンテンツサービス方法。

```
<complexType name="OutputDescription" >
  <sequence >
    <element name="Field" type="mpqf:FieldType" minOccurs="0" maxOccurs="unbounded" / >
    <element name="AggregateID" type="IDREF" minOccurs="0" maxOccurs="unbounded" / >
    <element name="GroupBy" minOccurs="0" >
      <complexType >
        <sequence >
          <element name="GroupByField" type="mpqf:FieldType" maxOccurs="unbounded" / >
          <element name="Aggregate" type="mpqf:AggregateExpressionType" minOccurs="0" maxOccurs="unbounded" / >
        </sequence >
      </complexType >
    </element >
    <element name="SortBy" type="mpqf:AbstractSortByType" minOccurs="0" maxOccurs="unbounded" / >
  </sequence >
  <attribute name="maxPage" type="positiveInteger" use="optional" / >
  <attribute name="maxItem" type="positiveInteger" use="optional" / >
  <attribute name="freeText" type="Boolean" use="optional" / >
  <attribute name="thumbnail" type="Boolean" use="optional" / >
  <attribute name="mediaResource" type="Boolean" use="optional" / >
  <attribute name="outputName" type="anyURI" use="required" / >
  <!--This indicates that if I want to receive a link to THE actual media resource.-- >
</complexType >
<complexType name="FieldType" >
  <simpleContent >
    <extension base="mpeg7:xPathRefType" >
      <attribute name="typeName" type="string" use="optional" / >
    </extension >
  </simpleContent >
</complexType >
```

【請求項 5】

前記複合演算は、高いレベルの間合せコンテンツが可能であり間合せ結果を表現可能にする特定の演算であることを特徴とする請求項 1 に記載の SOAP オペレーションを用いた間合せコンテンツサービス方法。

【請求項 6】

前記複合演算は、比較演算をさらに含むことを特徴とする請求項 1 に記載の SOAP オ

10

20

30

40

50

ペレージョンを用いた問合せコンテンツサービス方法。

【請求項7】

請求項1に記載されたSOAPオペレーションを用いた問合せコンテンツサービス方法の各ステップを行う命令語を含むコンピュータ読取可能な記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、コンテンツ提供サービスに関するものであって、特にSOAPオペレーションを用いて問合せされたコンテンツをサービスする方法に関する。

【背景技術】

10

【0002】

近年、デジタル放送サービスの本格化するにつれてマルチチャンネル、マルチメディア環境においてオンデマンド型の放送サービスを提供するための技術に関する研究が盛んに進められている。一例として、民間の国際標準であるTV-Anytimeは、コンテンツの記述(description)情報を表現するメタデータに基づいて、ユーザが自身の嗜好(preference)情報と前述したメタデータとをマッチングすることにより求めるコンテンツを格納して、自由な時間に視聴可能にするエニタイム(Anytime)サービスを提供するための標準規格である。

【0003】

メタデータは前述したようにコンテンツに関する記述情報として、TV-Anytimeでは、MPEG-7で定義された内容基盤(content-based)の記述とEPG(電子プログラムガイド)情報を含み、ユーザの求めるコンテンツを容易に探索・選択可能にする。メタデータ標準は、2つのパートから構成され、パートAは、メタデータを記述するためのフォーマット、即ち、スキーマ(schema)を定義したものであって、XML(extendible Markup Language)基盤のMPEG-7 DDL(Description Definition Language)(ISO/IEC15938-2)を活用する。パートBは、メタデータの伝送に関するものであって、バイナリフォーマット[MPEG-7 BiM(Binary Format for MPEG-7)](ISO/IEC15938-1)、断片化(fragmentation)モデル、カプセル化(encapsulation)、及び索引(indexing)技法を含んでいる。

20

30

【0004】

図1は、一般的なTV-Anytimeメタデータの構成を示すものであって、プログラム記述メタデータ(Program Description MetaData)及びユーザ記述メタデータ(User Description MetaData)を含んでおり、プログラム記述メタデータは、コンテンツ記述メタデータとインスタンス(instance)記述メタデータから構成されている。一つのプログラムに対するメタデータは、CRID(Content Reference Identifier、以下“CRID”と言う)と呼ばれるコンテンツ識別子で相互に関連付けられる。

【0005】

40

コンテンツ記述メタデータは、コンテンツ制作者(Content Creator)によって生成され、プログラムタイトル、ジャンル、要約、批評家レビューなどを含む。インスタンス記述メタデータは、コンテンツプロバイダ(Content Provider)によって生成され、ロケーション(放送時間、チャンネル、URLなど)、使用規則(usage rule)、伝送パラメータ(delivery parameter)などを含む。最後に、ユーザ記述メタデータは、ユーザ嗜好(user preference)、使用履歴(usage history)、個人ブックマーク(personal bookmark)などを含み、ユーザによって生成される。

【0006】

TV-Anytime標準は、リターンパスを通じた双方向メタデータサービスのため

50

に、2つのタイプのメタデータウェブサービスを定義しており、これは、well-defined behaviorと入出力セットに対するリモートプロシージャ(remote procedure)である。XML(extendible Markup Language; 拡張性生成言語)基盤のWSDL(Web Service Description Language; ウェブサービス記述言語)標準において、前述したリモートプロシージャ(remote procedure)は、シンプル・オブジェクト・アクセス・プロトコル(SOAP; Simple Object Access Protocol、以下“SOAP”と言う)オペレーションの形態で定義されており、メタデータ検索のための“get__Data”オペレーションと、ユーザ記述提出(user description submission)のための“submit__Data”オペレーションとがある。なお、前述したSOAPプロトコルは分散環境において客体(Object)にアクセス可能とするXML通信プロトコルである。

10

【0007】

TV-Anytimeメタデータサービスにおいて使用される要求(Request)/応答(Response)のタイプは、“urn;tva;transport:2002”の名前空間に定義される。この名前空間は種々のメッセージを検証するためのツールとして提供される。メタデータ仕様とコンテンツ参照(Content Referencing)標準によって定義されるタイプは、伝送(transport)名前空間において参照される。スキーマ断片(Schema fragment)は、前述した名前空間で定義される。そして、名前空間プロバイダは、スキーマ断片において“tns:”と定義される。完全なXMLスキーマファイルはtva__transport__types__v10.xsdである。

20

【0008】

1. get__Dataオペレーション

get__Dataオペレーションは、クライアントがプログラムまたはプログラムグループに対し、TV-Anytimeデータをサーバから検索する機能を提供する。TV-Anytimeメタデータプロバイダが、get__Dataオペレーションを使用して提供可能な機能を例示すると、以下の通りである。

【0009】

- CRIDリストを用いてCRIDに対するコンテンツ参照データを返すこと。
- CRIDリストを用いてCRIDに対するTV-Anytimeメタデータを返すこと。
- 特定のメタデータ属性(Attribute)(例えば、ジャンル、俳優など)に対する問合せを受信し、これに該当するプログラムを返すこと。
- 特定の時間または特定のチャンネルに対する問合せに回答し、該当プログラムを返すこと。

30

【0010】

get__Dataオペレーションは、原則的に全ての問合せタイプに対応し、メタデータ制限条件に対し広範な問合せを提供する。

【0011】

イ. 要求フォーマット(Request Format)

図2ないし図4に示すように、get__Dataオペレーションにおいて、要求フォーマットは、クライアントに3つのタイプのパラメータを指定し、問合せ(検索)結果値として返還されるエレメントタイプを要求テーブル(Requested Tables)タイプに指定する。

40

【0012】

ロ. 応答フォーマット(Response Format)

get__Dataオペレーションの応答フォーマットは、図5に示すように、エレメント(TVAMain、ContentReferencingTable、InvalidFragments)に対し、0または一つ以上のXMLインスタンスドキュメントを

50

含んでおり、要求フォーマットにおいて要求した要求テーブル (Requested Tables) タイプに応じて問合せ結果値を返す。

【 0 0 1 3 】

見てみた通り、現在定義された TV - Anytime オペレーションは検索問合せを要求する時、検索を求める特定の要素 (element)、属性 (attribute) などのフィールド値に対する文字列比較検索を行う。

【 0 0 1 4 】

このような従来の SOAP オペレーションを用いたメタデータ提供方法は、基本的なメタデータ検索形式と構造によってタイトルまたはジャンルによるコンテンツを検索する方法 (例えば、特定のフィールド値に対するテキスト比較) から成る。しかし、多様な検索結果 (Output Description) を表現するための形式は定義されておらず、多様な問合せに対応する問合せコンテンツをサービスできないという問題点がある。

【 発明の概要 】

【 発明が解決しようとする課題 】

【 0 0 1 5 】

本発明は、前述した問題を解決するために案出されたものであって、SOAP 問合せオペレーションにより具体的なコンテンツ問合せが要求される場合に、該当コンテンツの検索結果を集合処理してサービスできるようにした SOAP オペレーションを用いた問合せコンテンツ提供方法を提供する。

【 0 0 1 6 】

また本発明の他の目的は、SOAP 問合せオペレーションにおいて比較演算及び/または算術演算子を用いて複合的な演算結果を満たす該当コンテンツの問合せ結果をソーティング (sorting) 及びグルーピング (grouping) してサービスできるようにした SOAP オペレーションを用いた問合せコンテンツ提供方法を提供する。

【 課題を解決するための手段 】

【 0 0 1 7 】

前述した目的を達成するため、本発明の一面に係る SOAP オペレーション (演算) を用いた問合せコンテンツサービス方法は、サーバ及びクライアントを含むネットワークにおいて、SOAP 問合せオペレーションを用いて問合せされたコンテンツを提供する方法において、クライアントが、複合演算で多様なコンテンツ検索を要求可能に要求要素フィールドタイプを追加した前記 SOAP 問合せオペレーションの要求メッセージを送るステップと、前記要求メッセージが問合せ要求結果値を指定する前記要求要素フィールドタイプを含む場合に、サーバが、前記要求フィールドタイプ要素で指定されたテーブルフィールドに該当する問合せ結果表現値を抽出し、その問合せ表現値を問合せ結果要素に追加して SOAP オペレーションの応答メッセージにより送るステップとを含んで成ることを特徴とする。

【 0 0 1 8 】

前記応答メッセージは、maxPage、maxItem、freeText、resource、outputName、Field、AggregateID、Field Combination、SortBy、GroupBy、GroupField、Aggregate と Thumbnail 及び typeName のうち少なくとも一つの問合せ結果要素を含んで成ることを特徴とする。

【 0 0 1 9 】

前記応答メッセージは、問合せ要求したコンテンツによる問合せ結果をソーティング及び/またはグルーピングすることを特徴とする。

【 0 0 2 0 】

前記複合オペレーションは、高いレベルの問合せコンテンツが可能であり、問合せ結果を表現可能にする特定のオペレーションであることを特徴とする。

【 0 0 2 1 】

また前述した目的を達成するため、本発明の他の態様では、SOAP オペレーションを

10

20

30

40

50

用いた問合せコンテンツサービス方法の各ステップを行う命令語を含むコンピュータ読取可能記録媒体が提供される。

【発明の効果】

【0022】

前述した課題解決手段により、本発明は、SOAP問合せオペレーション問合せ要求フィールドに算術演算及び/または比較演算に関する演算子を追加してコンテンツの問合せ要求時、問合せされたコンテンツの抽出順序とともにソーティング及び/またはグルーピングして、より正確で且つ多様なコンテンツの問合せ結果をサービスすることによって、ユーザに対する選好(嗜好)情報に基づいてより正確で且つ多様なコンテンツの問合せ結果をユーザにサービスしてユーザの便宜性を増大させる効果がある。

10

【図面の簡単な説明】

【0023】

【図1】一般的なTV-Anytimeメタデータの構成図である。

【図2】get_Dataオペレーションの一般的な動作概念図である。

【図3】従来のget_Dataオペレーションの要求フォーマットを示す例示図である。

【図4】従来のget_Dataオペレーションの要求フォーマットを示す例示図である。

【図5】従来のget_Dataオペレーションの要求フォーマットを示す例示図である。

20

【図6】従来のget_Dataオペレーションの応答フォーマットを示す例示図である。

【図7】本発明に適用されるConstant Valueタイプを示す例示図である。

【図8】本発明に適用されるField Combinationタイプを示す例示図である。

【図9】本発明に適用されるField Combinationタイプを示す例示図である。

【図10】本発明に適用されるField Combinationタイプを示す例示図である。

【図11】本発明に適用されるField Combinationタイプを示す例示図である。

30

【図12】本発明に適用されるComparison Typeを示す例示図である。

【図13】本発明に適用されるComparison Typeを示す例示図である。

【図14】本発明に適用されるComparison Typeを示す例示図である。

【図15】本発明のSOAPオペレーションの応答メッセージタイプを示す例示図である。

【発明を実施するための形態】

【0024】

以下、本発明の好ましい実施例を図面に沿って説明する。

本発明のSOAPオペレーションを用いた問合せコンテンツサービス方法は、SOAP問合せオペレーションにより問合せ要求したコンテンツに対し、問合せ結果をソーティング及びグルーピングして問合せコンテンツをサービス可能にする技術的な意味を有する。この実施例による方法は、SOAP問合せオペレーションに算術演算(オペレーション)及び/または比較演算(オペレーション)で多様なコンテンツ検索を要求可能に要求エレメントフィールドタイプを追加する。それによって、要求エレメントフィールドタイプが追加されたSOAP問合せオペレーションによるSOAP応答メッセージに問合せ結果フィールドに拡張された形態の問合せ結果エレメントを追加して問合せ結果に対する集合処理された問合せコンテンツをサービスする。

40

【0025】

本発明の明細書全般に亘って複合演算は、高いレベルの問合せコンテンツを表現可能に

50

する演算を意味するものと定義される。

【0026】

以下、本発明に係る好ましい実施例を添付の図面を参照して詳細に説明し、本発明に係る動作及び作用を理解するのに必要な部分を中心として説明する。

【0027】

以下の説明において、本発明のSOAPオペレーションを用いた問合せコンテンツサービス方法の特定の詳細事項は、本発明をより全般的に理解できるように示されているが、これら特定の詳細事項無しに、またこれらの変形によっても本発明が容易に実施できるということはこの技術分野において通常の知識を持つ者であれば自明である。

【0028】

図6は、本発明のメタデータ配布オペレーションの概念を示している。TV-Anytimeサービスにおいてクライアント(Client)はインターネット網(IP Network)などを通じてget_DataオペレーションによるSOAP要求メッセージ[get_Data()Request]をメタデータサービスサーバ(Metadata Service Server)に伝送する。この際、get_Dataオペレーションは、原則的に全ての問合せタイプに対応し、メタデータ制限条件に対し広範な問合せを提供する。

【0029】

特に、get_Dataオペレーションの特定のフィールドに比較演算の内在的演算を通じた複合的な演算表現が可能ないように要求エレメントフィールドタイプを追加する。そうすると、メタデータサービスサーバ(Metadata Service Server)はSOAP応答メッセージ[get_Data()Request]の問合せ結果フィールドにより前記要求エレメントフィールドタイプが追加されたSOAP要求メッセージに対する問合せ結果値に該当するメタデータをソーティング及び/またはグルーピングして返す。

【0030】

このようにコンテンツ問合せ時、問合せ結果値をソーティング及び/またはグルーピングして返すようにするためには、複合演算(算術演算及び/または比較演算)でコンテンツを問合せ可能にSOAP問合せオペレーションの要求エレメントフィールドを追加しなければならない。

【0031】

次は、SOAPオペレーションにおいて算術演算で多様なコンテンツ検索を要求可能に要求エレメントフィールドタイプを追加する場合に対して詳細に説明する。

【0032】

1. Constant Value

Constant Valueタイプ(コンスタントバリュー類型)は、算術演算の被演算子を常数に割り当てるために定義されたタイプである。Constant Valueタイプは、MPEG7とxml基本タイプにおいて算術演算と関連付けられた全てのタイプを選択的な要素として定義し含んでいる。

【0033】

Constant Valueタイプに登場するタイプ要素は、演算子を適用する際に、被演算子のタイプとして選択されて、被演算子値とタイプ情報を含んでいる。Constant Valueタイプから一つのタイプを選択することによって、正しい演算のために演算子の使用中に即時的な被演算子のタイプチェックが可能である。

【0034】

このようなConstant Valueタイプは図7に示し、算術演算子のタイプは、次の表1に定義された通りである。

10

20

30

40

【表1】

名前	定義
Constant Value Type	この型は FieldCombinationType 型と ComparisonOperatorType 型において被演算子で使用可能な型を選択的要素として含むことによる説明し規定する。
unsigned 1	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 1 であることを示す。
unsigned 2	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 2 であることを示す。
unsigned 3	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 3 であることを示す。
unsigned 4	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 4 であることを示す。
unsigned 5	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 5 であることを示す。
unsigned 6	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 6 であることを示す。
unsigned 7	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 7 であることを示す。
unsigned 8	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 8 であることを示す。
unsigned 9	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 9 であることを示す。
unsigned 10	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 10 であることを示す。
unsigned 11	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 11 であることを示す。
unsigned 12	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 12 であることを示す。
unsigned 13	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 13 であることを示す。
unsigned 14	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 14 であることを示す。
unsigned 15	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 15 であることを示す。
unsigned 16	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 16 であることを示す。
unsigned 17	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 17 であることを示す。
unsigned 18	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 18 であることを示す。
unsigned 19	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 19 であることを示す。
unsigned 20	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 20 であることを示す。
unsigned 21	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 21 であることを示す。
unsigned 22	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 22 であることを示す。
unsigned 23	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 23 であることを示す。
unsigned 24	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 24 であることを示す。
unsigned 25	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 25 であることを示す。
unsigned 26	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 26 であることを示す。
unsigned 27	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 27 であることを示す。
unsigned 28	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 28 であることを示す。
unsigned 29	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 29 であることを示す。
unsigned 30	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 30 であることを示す。
unsigned 31	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 31 であることを示す。
unsigned 32	この要素は該被演算子の型が MPEG7 基本データ型にある unsigned 32 であることを示す。
Integer	この要素は該被演算子の型が MPEG7 基本データ型にある Integer であることを示す。
float	この要素は該被演算子の型が MPEG7 基本データ型にある float であることを示す。
double	この要素は該被演算子の型が MPEG7 基本データ型にある double であることを示す。
decimal	この要素は該被演算子の型が MPEG7 基本データ型にある decimal であることを示す。
long	この要素は該被演算子の型が MPEG7 基本データ型にある long であることを示す。
int	この要素は該被演算子の型が MPEG7 基本データ型にある int であることを示す。
short	この要素は該被演算子の型が MPEG7 基本データ型にある short であることを示す。
nonPositiveInteger	この要素は該被演算子の型が MPEG7 基本データ型にある nonPositiveInteger であることを示す。

10

20

30

40

2. 算術演算子

コンテンツ情報抽出過程において、抽出結果を生成するだけでなく抽出条件を明示するために算術値を越す演算子の定義は必要である。よって、コンテンツ問合せ形式においても、次のような幾つか重要な算術演算子を定義する必要がある。

- Field Combination

Field Combinationタイプは算術値を越す全ての演算子の父母クラスと定義されている。全ての算術演算子は、Field Combinationを抽象型として、承継しなければならない。算術演算子はまた演算後に結果値を越さなければならない。このようなField Combinationタイプは、図8ないし図11に示されており、問合せ形式に適用するために表2のような算術演算子のタイプを規定している

【表 2】

名前	定義
FieldCombination	算術演算子を定義するための基本類型
sum	FieldCombinationを基盤にして問合せにおいて“sum”算術演算子を表現するために導出された類型。被演算子がこの演算子に配置される場合、この演算子は全ての被演算子の和を意味する。
average	FieldCombinationを基盤にして問合せにおいて“average”算術演算子を表現するために導出された類型。被演算子がこの演算子に配置される場合、この演算子は全ての被演算子の平均を意味する。
count	FieldCombinationを基盤にして問合せにおいて“count”算術演算子を表現するために導出された類型。被演算子がこの演算子に配置される場合、この演算子は全ての被演算子の個数を意味する。
max	FieldCombinationを基盤にして問合せにおいて“max”算術演算子を表現するために導出された類型。被演算子がこの演算子に配置される場合、この演算子は全ての被演算子のうち最大被演算子を意味する。
min	FieldCombinationを基盤にして問合せにおいて“min”算術演算子を表現するために導出された類型。被演算子がこの演算子に配置される場合、この演算子は全ての被演算子のうち最小被演算子を意味する。
add	FieldCombinationを基盤にして問合せにおいて“add”算術演算子を表現するために導出された類型。被演算子がこの演算子に配置される場合、この演算子は被演算子の和を意味する。
multiply	FieldCombinationを基盤にして問合せにおいて“multiply”算術演算子を表現するために導出された類型。被演算子がこの演算子に配置される場合、この演算子は全ての被演算子の乗算を意味する。
subtract	FieldCombinationを基盤にして問合せにおいて“subtract”算術演算子を表現するために導出された類型。二つの被演算子が演算子に配置される場合、この演算子は第一被演算子から第二被演算子を引いた値を意味する。
divide	FieldCombinationを基盤にして問合せにおいて“divide”算術演算子を表現するために導出された類型。二つの被演算子が演算子に配置される場合、この演算子は第一被演算子から第二被演算子を割り算した値を意味する。
modulus	FieldCombinationを基盤にして問合せにおいて“modulus”算術演算子を表現するために導出された類型。二つの被演算子が演算子に配置される場合、この演算子は第一被演算子から第二被演算子を割り算した残りを意味する。
round	FieldCombinationを基盤にして問合せにおいて“round”算術演算子を表現するために導出された類型。一つの被演算子がこの演算子に配置される場合、この演算子は被演算子に最も近い整数を意味する。
floor	FieldCombinationを基盤にして問合せにおいて“floor”算術演算子を表現するために導出された類型。一つの被演算子がこの演算子に配置される場合、この演算子は被演算子に最も小さく且つ近い整数を意味する。
ceil	FieldCombinationを基盤にして問合せにおいて“ceil”算術演算子を表現するために導出された類型。一つの被演算子がこの演算子に配置される場合、この演算子は被演算子に最も大きく且つ近い整数を意味する。
Field	この要素は各算術演算のための被演算子を表現する。
FieldCombination	この要素は各算術演算のための内在された形式の被演算子を表現する。
GroupBy	この要素は各演算は規定された特定条件によって集められた後に行わなければならないということを示す。

10

20

30

40

【0036】

次は、SOAP問合せオペレーションにおいて、算術演算と或いは単独の比較演算で多様なコンテンツ検索を要求できるように要求エレメントフィールドタイプを追加する場合に対し、詳細に説明する。

【0037】

1. Comparison Operator Type

Comparison Operator Typeは、Operator Typeを拡張して定義され、比較演算を説明する全ての演算子タイプの上位クラスになる抽象型である。全ての比較演算子タイプはComparison Operator Typeを抽象型として承継しなければならない。本提案の比較演算子は、結果としてブール(B

50

o o l e a n) 値を越さなければならない。このような C o m p a r i s o n O p e r a t o r T y p e は、図 1 2 ないし図 1 4 に示している。

【 0 0 3 8 】

一方、図 1 2 ないし図 1 4 の比較演算子は表 3 に記載されたように規定される。

【 表 3 】

名 前	定 義
C o m p a r i s o n T y p e	論理演算子を定義するための基本タイプ。
G T E	C o m p a r i s o n T y p e を基盤にして問合せにおいて“G T E”演算を定義するクラスである。二つの被演算子がこの演算に配置される場合、この演算子は二つの被演算子間の大きいか等しい関係が真であるか偽であるかを判別することを意味する。
G T	C o m p a r i s o n T y p e を基盤にして問合せにおいて“G T”演算を定義するクラスである。二つの被演算子がこの演算に配置される場合、この演算子は二つの被演算子間の大きいか等しい関係が真であるかを判別することを意味する。
L T E	C o m p a r i s o n T y p e を基盤にして問合せにおいて“L T E”演算を定義するクラスである。二つの被演算子がこの演算に配置される場合、この演算子は二つの被演算子間の小さいか等しい関係が真であるかを判別することを意味する。
L T	C o m p a r i s o n T y p e を基盤にして問合せにおいて“L T”演算を定義するクラスである。二つの被演算子がこの演算に配置される場合、この演算子は二つの被演算子間の小さい被演算子の関係が真であるかを判別することを意味する。
E Q	C o m p a r i s o n T y p e を基盤にして問合せにおいて“E Q”演算を定義するクラスである。二つの被演算子がこの演算に配置される場合、この演算子は二つの被演算子間の大きさの等しい関係が真であるかを判別することを意味する。
N E	C o m p a r i s o n T y p e を基盤にして問合せにおいて“N E”演算を定義するクラスである。二つの被演算子がこの演算に配置される場合、この演算子は二つの被演算子間の大きさが異なるという関係が真であるかを判別することを意味する。
c o n t a i n	C o m p a r i s o n T y p e を基盤にして問合せにおいて“c o n t a i n”演算を定義するクラスである。c o n t a i n 演算は先立つストリングに追うストリングが登場するかどうかを判断する演算である。従って、この命令は二つのストリングを含むようになる。

10

20

“ G T E ”、“ G T ”、“ L T E ”、“ L T ”“ E Q ”“ N E ”“ c o n t a i n ”のような比較演算子は、二つの被演算子を表現する要素が必要である。また、比較演算子は被演算子として C o n s t a n t V a l u e T y p e により制限される算術的な整数値を有する。全ての演算は F i e l d C o m b i n a t i o n T y p e の内在された演算表現を有する。

30

【 0 0 3 9 】

一方、前述した算術演算及び/または比較演算により問合せ要求した要求フィールドタイプエレメントにおいて指定されたテーブルフィールドに該当する問合せ結果値を抽出し、その問合せ結果値を S O A P オペレーションの応答メッセージにより伝送する技術に対し詳細に説明すると、以下の通りである。

【 0 0 4 0 】

T V - A n y t i m e の g e t _ D a t a オペレーションはクライアントがプログラムまたはプログラムグループに対し T V - A n y t i m e データをサーバへ検索する機能を提供する。T V - A n y t i m e M e t a d a t a プロバイダが g e t _ D a t a オペレーションを使用して提供可能な機能を例示すると、次の通りである。

40

【 0 0 4 1 】

- C R I D リストを用いて C R I D に対するコンテンツ参照データを返すこと。
 - C R I D リストを用いて C R I D に対する T V - A n y t i m e M e t a d a t a を返すこと。
 - オペレーションは特別な M e t a d a t a 属性 (A t t r i b u t e) (例えば、ジャンル、俳優など) に対する問合せを受信し、これに該当されるプログラムを返すこと。
- g e t _ D a t a オペレーションは原則的に全ての問合せタイプに対応し、メタデータ制限条件に対して広範な問合せを提供し、このような問合せの結果は数個のメタデータ

50

ーブルで表示される。メタデータテーブルは特定記述条件がなければ抽出された順に表示される。しかし、ユーザの可読性及びプログラムの効率化のために抽出された順を明示する方法が必要であり、既存のデータベース及びウェブ検索などで使用される Group By、Sort By などの集合処理方法が必要である。

【0042】

ここに本発明では、マルチメディア検索の結果に対する集合処理のための方法を提案する。

問合せ結果 (Output Description) タイプは、入力要素において Output Description 要素を規定するための形式であって、算術演算及び / または比較演算を通じた結果抽出過程を提供するとともに整列機能と併せて多様な算術演算と合算機能に対応するように拡張された形態の問合せ検索結果を提案する。

10

【0043】

また SOAP オペレーションの応答メッセージは、問合せ結果として Group By、Sort By エlement を有する。Group By Element は、問合せ結果セットに対する集合分類を示す。このために、分類基準となるフィールドタイプをサブ Element として有する。Sort By Element は整列基準を示す。

【0044】

このような SOAP オペレーションの応答メッセージタイプは、図 8 に示されており、表 4 のように定義される問合せ結果 Element で成される。

20

【表 4】

名前	定義
OutputDescription	マルチメディア検索の出力結果において越す、選択されたデジタルアイテム(Digital item)と関連付けられた情報を表現する方法を明示している。各アイテムメタデータ内にはデータ経路を明示している。
maxPage	最大結果ページを表現する(選択的)。与えられた問合せの結果アイテムは他の結果ページ間に分けられる。
maxItem	結果アイテムの最大個数を表現する(選択的)。
freeText	各々の結果アイテムに freeText 領域が現れることが期待されるかどうかを明示するブール要素である(選択的)。若しこの要素値が false であれば、freeText 領域は結果アイテムに現れてはいけない。若しこの要素が明示されなければ、ユーザは freeText が現れたかどうかは関係ないということを仮定する。
resource	各々の結果アイテムに resource 領域が現れることが期待されるかどうかを明示するブール要素である(選択的)。若しこの要素値が false であれば、freeText 領域は結果アイテムに現れてはいけない。若しこの要素が明示されなければ、ユーザは resource が現れたかどうかは関係ないということを仮定する。
outputName	各々の結果アイテムの Description 要素に現れるように期待されるメタデータ形式と関連付けられたネームスペース(namespace)の URI を明示する。outputName は MPEG7 または他のメタデータ形式と関連付けるネームスペースとなり得る。
Field	アイテムのメタデータ内でデータ経路を明示する。経路はアイテムのメタデータのルート(root)を指す絶対 XPath と表現し、或いは選択的に MPEG7 スキーマの導出された形式を指す相対的な XPath 表現を使用することができる。
AggregateID	ユーザが所望する operation の ID。
FieldCombination	アイテムのメタデータにおいてフィールド(field)を取る合算機能或いは算術的関数を明示する。合算機能と算術関数は算術演算者の下位節において規定する。
SortBy	整列と順序のための主要フィールドを明示する(昇順或いは降順)。
GroupBy	この要素はこの演算は規定された特定条件によって集めた後に行わなければならないことを示す。
GroupByField	グループ化過程のキー(key)となるフィールド。
Aggregate	ユーザが具体的なグルーピング演算の結果に適用しようとする演算の ID。
thumbnail	プレビュー機能を使用するかどうかをブールで表現
typeName	MPEG7 や他のメタデータ形式のスキーマの名前を具体化。XPath の相対経路と表現

30

40

【産業上の利用可能性】

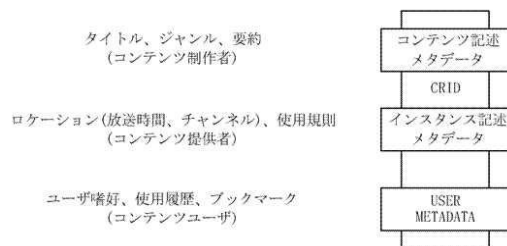
【0045】

以上説明した通り、SOAP 問合せオペレーションの問合せ要求フィールドに算術演算及び / または比較演算に関する演算子を追加してコンテンツの問合せ要求時、問合せされたコンテンツの抽出順序とともにソーティング及び / またはグルーピングしてより正確で且つ多様なコンテンツの問合せ結果をサービスすることによって、ユーザに対する選好 (

50

嗜好)情報に基づいてより正確で且つ多様なコンテンツの問合せ結果をユーザにサービスしてユーザの便宜性を増大させることができる。

【図1】



【図2】

```

<element name="get_Data" type="tns:get_Data"/>
<complexType name="get_Data">
  <sequence>
    <element name="QueryConstraints">
      <complexType>
        <choice>
          <element name="PredicateBag" type="tns:PredicateBagType"/>
          <element name="BinaryPredicate" type="tns:BinaryPredicateType"/>
          <element name="UnaryPredicate" type="tns:UnaryPredicateType"/>
        </choice>
      </complexType>
    </element>
    <element name="RequestedTables" type="tns:RequestedTablesType"/>
  </sequence>
  <attribute name="maxPrograms" type="unsignedInt"/>
</complexType>

```

【 3 】

```

<complexType name="PredicateBagType">
  <sequence maxOccurs="unbounded">
    <choice>
      <element name="PredicateBag" type="tns:PredicateBagType"/>
      <element name="BinaryPredicate" type="tns:BinaryPredicateType"/>
      <element name="UnaryPredicate" type="tns:UnaryPredicateType"/>
    </choice>
  </sequence>
  <attribute name="contextNode" type="tns:contextNodeIDType"/>
  <attribute name="negate" type="boolean" default="false"/>
  <attribute name="type" type="tns:PredicateBagTypeType"/>
</complexType>
<simpleType name="PredicateBagTypeType">
  <restriction base="string">
    <enumeration value="AND"/>
    <enumeration value="OR"/>
  </restriction>
</simpleType>
<complexType name="BinaryPredicateType">
  <attribute name="fieldID" type="tns:fieldIDType" use="required"/>
  <attribute name="fieldValue" type="string" use="required"/>
  <attribute name="test" default="equals"
    type="tns:BinaryPredicateTestType"/>
</complexType>
<complexType name="UnaryPredicateType">
  <attribute name="fieldID" type="tns:fieldIDType" use="required"/>
  <attribute name="test" default="exists"
    type="tns:UnaryPredicateTestType"/>
</complexType>

```

【 4 】

```

<simpleType name="BinaryPredicateTestType">
  <restriction base="string">
    <enumeration value="equals"/>
    <enumeration value="not_equals"/>
    <enumeration value="contains"/>
    <enumeration value="greater_than"/>
    <enumeration value="greater_than_or_equals"/>
    <enumeration value="less_than"/>
    <enumeration value="less_than_or_equals"/>
  </restriction>
</simpleType>
<simpleType name="UnaryPredicateTestType">
  <restriction base="string">
    <enumeration value="exists"/>
  </restriction>
</simpleType>

```

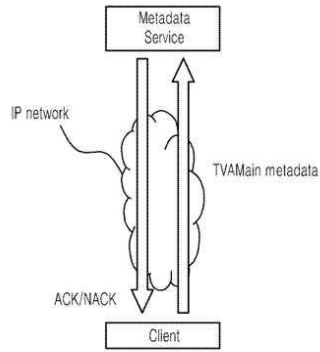
【 5 】

```

<element name="get_Data_Result" type="tns:get_Data_ResultType"/>
<complexType name="get_Data_ResultType">
  <sequence>
    <element name="TableSortingInformation"
      type="tns:RequestedTablesType" minOccurs="0"/>
    <element ref="tva:TVAMain" minOccurs="0"/>
    <element ref="cr:ContentReferencingTable" minOccurs="0"/>
    <element name="InvalidFragments"
      type="tns:InvalidFragmentsType" minOccurs="0"/>
  </sequence>
  <attribute name="serviceVersion" type="unsignedInt" use="required"/>
  <attribute name="truncated" type="boolean"/>

```

【 6 】



【 7 】

```
<complexType name="ConstantValue">
  <choice>
    <element name="unsigned1" type="mpeg7:unsigned1"/>
    <element name="unsigned2" type="mpeg7:unsigned2"/>
    <element name="unsigned3" type="mpeg7:unsigned3"/>
    <element name="unsigned4" type="mpeg7:unsigned4"/>
    <element name="unsigned5" type="mpeg7:unsigned5"/>
    <element name="unsigned6" type="mpeg7:unsigned6"/>
    <element name="unsigned7" type="mpeg7:unsigned7"/>
    <element name="unsigned8" type="mpeg7:unsigned8"/>
    <element name="unsigned9" type="mpeg7:unsigned9"/>
    <element name="unsigned10" type="mpeg7:unsigned10"/>
    <element name="unsigned11" type="mpeg7:unsigned11"/>
    <element name="unsigned12" type="mpeg7:unsigned12"/>
    <element name="unsigned13" type="mpeg7:unsigned13"/>
    <element name="unsigned14" type="mpeg7:unsigned14"/>
    <element name="unsigned15" type="mpeg7:unsigned15"/>
    <element name="unsigned16" type="mpeg7:unsigned16"/>
    <element name="unsigned17" type="mpeg7:unsigned17"/>
    <element name="unsigned18" type="mpeg7:unsigned18"/>
    <element name="unsigned19" type="mpeg7:unsigned19"/>
    <element name="unsigned20" type="mpeg7:unsigned20"/>
    <element name="unsigned21" type="mpeg7:unsigned21"/>
    <element name="unsigned22" type="mpeg7:unsigned22"/>
    <element name="unsigned23" type="mpeg7:unsigned23"/>
    <element name="unsigned24" type="mpeg7:unsigned24"/>
    <element name="unsigned25" type="mpeg7:unsigned25"/>
    <element name="unsigned26" type="mpeg7:unsigned26"/>
    <element name="unsigned27" type="mpeg7:unsigned27"/>
    <element name="unsigned28" type="mpeg7:unsigned28"/>
    <element name="unsigned29" type="mpeg7:unsigned29"/>
    <element name="unsigned30" type="mpeg7:unsigned30"/>
    <element name="unsigned31" type="mpeg7:unsigned31"/>
    <element name="integer" type="integer"/>
    <element name="float" type="float"/>
    <element name="double" type="double"/>
    <element name="decimal" type="decimal"/>
    <element name="long" type="long"/>
    <element name="int" type="int"/>
    <element name="short" type="short"/>
    <element name="nonPositiveInteger" type="nonPositiveInteger"/>
    <element name="negativeInteger" type="negativeInteger"/>
    <element name="nonNegativeInteger" type="nonNegativeInteger"/>
    <element name="unsignedLong" type="unsignedLong"/>
    <element name="unsignedShort" type="unsignedShort"/>
    <element name="positiveInteger" type="positiveInteger"/>
    <element name="unsignedInt" type="unsignedInt"/>
    <!--add more!-->
  </choice>
</complexType>
```

【 図 8 】

```

<complexType name='FieldCombination' abstract='true'>
  <complexContent>
    <restriction base='anyType' />
  </complexContent>
</complexType>
<complexType name='Sum'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence maxOccurs='unbounded'>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
        </choice>
      </sequence>
      <element name='GroupBy' type='mp7qf:FieldType' minOccurs='0' />
    </extension>
  </complexContent>
</complexType>
<complexType name='Average'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence maxOccurs='unbounded'>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
        </choice>
      </sequence>
      <element name='GroupBy' type='mp7qf:FieldType' minOccurs='0' />
    </extension>
  </complexContent>
</complexType>
<complexType name='Count'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence maxOccurs='unbounded'>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
        </choice>
      </sequence>
      <element name='GroupBy' type='mp7qf:FieldType' minOccurs='0' />
    </extension>
  </complexContent>
</complexType>
<complexType name='Max'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence maxOccurs='unbounded'>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
        </choice>
      </sequence>
      <element name='GroupBy' type='mp7qf:FieldType' minOccurs='0' />
    </extension>
  </complexContent>
</complexType>

```

【 図 9 】

```

<complexType name='Min'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence maxOccurs='unbounded'>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
        </choice>
      </sequence>
      <element name='GroupBy' type='mp7qf:FieldType' minOccurs='0' />
    </extension>
  </complexContent>
</complexType>
<complexType name='Add'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence minOccurs='2' maxOccurs='unbounded'>
        <choice>
          <element name='Field' type='mp7qf:Field' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
          <element name='ConstantValue' type='mp7qf:ConstantValueType' />
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<complexType name='Multiply'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence minOccurs='2' maxOccurs='unbounded'>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
          <element name='ConstantValue' type='mp7qf:ConstantValue' />
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<complexType name='Subtract'>
  <complexContent>
    <extension base='mp7qf:FieldCombination'>
      <sequence>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
          <element name='ConstantValue' type='mp7qf:ConstantValue' />
        </choice>
        <choice>
          <element name='Field' type='mp7qf:FieldType' />
          <element name='FieldCombination' type='mp7qf:FieldCombination' />
          <element name='ConstantValue' type='mp7qf:ConstantValue' />
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

```

【 10 】

```

<complexType name="Divide">
  <complexContent>
    <extension base="mp7qf:FieldCombination">
      <sequence>
        <choice>
          <element name="Field" type="mp7qf:FieldType"/>
          <element name="FieldCombination" type="mp7qf:FieldCombination"/>
          <element name="ConstantValue" type="mp7qf:ConstantValue"/>
        </choice>
        <choice>
          <element name="Field" type="mp7qf:FieldType"/>
          <element name="FieldCombination" type="mp7qf:FieldCombination"/>
          <element name="ConstantValue" type="mp7qf:ConstantValue"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="Modulus">
  <complexContent>
    <extension base="mp7qf:FieldCombination">
      <sequence>
        <choice>
          <element name="Field" type="mp7qf:FieldType"/>
          <element name="FieldCombination" type="mp7qf:FieldCombination"/>
          <element name="ConstantValue" type="mp7qf:ConstantValue"/>
        </choice>
        <choice>
          <element name="Field" type="mp7qf:FieldType"/>
          <element name="FieldCombination" type="mp7qf:FieldCombination"/>
          <element name="ConstantValue" type="mp7qf:ConstantValue"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="Round">
  <complexContent>
    <extension base="mp7qf:FieldCombination">
      <sequence>
        <choice>
          <element name="Field" type="mp7qf:FieldType"/>
          <element name="FieldCombination" type="mp7qf:FieldCombination"/>
          <element name="ConstantValue" type="mp7qf:ConstantValue"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

```

【 11 】

```

<complexType name="Floor">
  <complexContent>
    <extension base="mp7qf:FieldCombination">
      <sequence>
        <choice>
          <element name="Field" type="mp7qf:FieldType"/>
          <element name="FieldCombination" type="mp7qf:FieldCombination"/>
          <element name="ConstantValue" type="mp7qf:ConstantValue"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="Ceil">
  <complexContent>
    <extension base="mp7qf:FieldCombination">
      <sequence>
        <choice>
          <element name="Field" type="mp7qf:FieldType"/>
          <element name="FieldCombination" type="mp7qf:FieldCombination"/>
          <element name="ConstantValue" type="mp7qf:ConstantValue"/>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="Abs">
  <complexContent>
    <extension base="mpqf:ArithmeticExpressionType">
      <choice>
        <element name="Field" type="mpqf:FieldType"/>
        <element name="mp7qf:FieldCombination" type="mpqf:ArithmeticExpressionType"/>
      </choice>
    </extension>
  </complexContent>
</complexType>

```


【 14 】

```

<complexType name="NE">
  <complexContent>
    <extension base="mpqt:ComparisonOperationType">
      <choice>
        <group ref="mpqt:ArithmeticOperands"/>
        <group ref="mpqt:BooleanOperands"/>
        <group ref="mpqt:DateTimeOperands"/>
        <group ref="mpqt:DurationOperands"/>
        <group ref="mpqt:StringOperands"/>
      </choice>
    </extension>
  </complexContent>
</complexType>
<complexType name="contain">
  <complexContent>
    <extension base="mpqt:ComparisonOperationType">
      <sequence>
        <group ref="mpqt:StringOperations"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<group name="BooleanOperands">
  <sequence>
    <choice minOccurs="2" maxOccurs="2">
      <element name="BooleanValue" type="boolean"/>
      <element name="BooleanField" type="mpqt:FieldType"/>
      <element name="BooleanExpression" type="mpqt:ConditionType"/>
    </choice>
  </sequence>
</group>
<group name="StringOperands">
  <sequence>
    <choice minOccurs="2" maxOccurs="2">
      <element name="StringField" type="mpqt:FieldType"/>
      <element name="StringExpression" type="mpqt:StringExpressionType"/>
    </choice>
  </sequence>
</group>
<group name="DateTimeOperands">
  <sequence>
    <choice minOccurs="2" maxOccurs="2">
      <element name="DateTime" type="dateTime"/>
      <element name="Date" type="date"/>
      <element name="Time" type="time"/>
      <element name="DateTimeField" type="mpqt:FieldType"/>
    </choice>
  </sequence>
</group>
<group name="DurationOperands">
  <sequence>
    <choice minOccurs="2" maxOccurs="2">
      <element name="Duration" type="duration"/>
      <element name="DurationField" type="mpqt:FieldType"/>
    </choice>
  </sequence>
</group>
</group>

```

【 15 】

```

<complexType name="OutputDescription">
  <sequence>
    <element name="Field" type="mpqt:FieldType" minOccurs="0" maxOccurs="unbounded"/>
    <element name="AggregateID" type="IDREF" minOccurs="0" maxOccurs="unbounded"/>
    <element name="GroupBy" minOccurs="0">
      <complexType>
        <sequence>
          <element name="GroupByField" type="mpqt:FieldType" maxOccurs="unbounded"/>
          <element name="Aggregate" type="mpqt:AggregateExpressionType" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
      </complexType>
    </element>
    <element name="SortBy" type="mpqt:AbstractSortByType" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="maxPage" type="positiveInteger" use="optional"/>
  <attribute name="maxItem" type="positiveInteger" use="optional"/>
  <attribute name="hasText" type="boolean" use="optional"/>
  <attribute name="thumbnail" type="boolean" use="optional"/>
  <attribute name="mediaResource" type="boolean" use="optional"/>
  <attribute name="outputName" type="anyURI" use="required"/>
  <!-- This indicates that I want to receive a link to the actual media resource. -->
</complexType>
<complexType name="FileType">
  <simpleContent>
    <extension base="mpqt:URIRefType">
      <attribute name="typeName" type="string" use="optional"/>
    </extension>
  </simpleContent>
</complexType>

```

フロントページの続き

- (72)発明者 イ ジョンソル
大韓民国 キョンギド ヨンインシ キフンク マブクドクン ヨンウォンマウル ビョクサン
アパート125 - 1503
- (72)発明者 イ ソクピル
大韓民国 キョンギド ソンナムシ ブンダンク クムゴクドクン180 ファイン アパート2
03 - 1704
- (72)発明者 リン テボム
大韓民国 キョンギド ヨンインシ ギフンク ボジヨンドクン ポスホームタウン304 - 601
- (72)発明者 ユン キョンロ
大韓民国 ソウル特別市 カンナムク ドゴクドクン895 - 8 ヨクサムハンシン アパート3
- 1205
- (72)発明者 シン サイム
大韓民国 ソウル特別市 ウンピョンク ロクバンドクン ジンロ アパート104 - 1705

審査官 久々宇 篤志

- (56)参考文献 特開2006 - 085698 (JP, A)
米国特許出願公開第2004 / 0210500 (US, A1)
田中 哲朗, SPARCの特徴を生かしたUtilisp / Cの実現法, 情報処理学会論文誌,
日本, 社団法人情報処理学会, 1991年 5月15日, 第32巻 第5号, pp.684-690

- (58)調査した分野(Int.Cl., DB名)
G06F 17/30