XMLスキーマ設計と スキーマリポジトリの研究

XMLDB部会

矢野 令（東芝ソリューション株式会社）

XML Consortium Week 2009.5.20.

目次

- 動機
- 目指すところ
- XMLDBが最適な理由
- スキーマリポジトリのユースケース
- デモイメージ
- まとめ

Copyright © XMLコンソーシアム 2009 All rights reserved.

動機1:
「”会員企業”はどのようなタグ名がいい？」

- XML設計技術勉強会での一コマ
  - XMLコンソーシアムの会員登録情報を題材に、XMLスキーマ設計のハンズオンを実施
  - 参加メンバでXMLスキーマの設計をディスカッション
  - 「”会員企業”はどのようなタグ名がいい？」
    - WebでW3C等の仕様書を検索して参考になるタグを探した

簡単にタグが探せる検索エンジンがあると便利！

※XML設計技術勉強会
2008年6月～にXMLコンソーシアムで実施された、XML設計技術の普及と啓発を目指した勉強会

Copyright © XMLコンソーシアム 2009 All rights reserved.

動機2:
「独自のXMLタグを作るべきではない！」

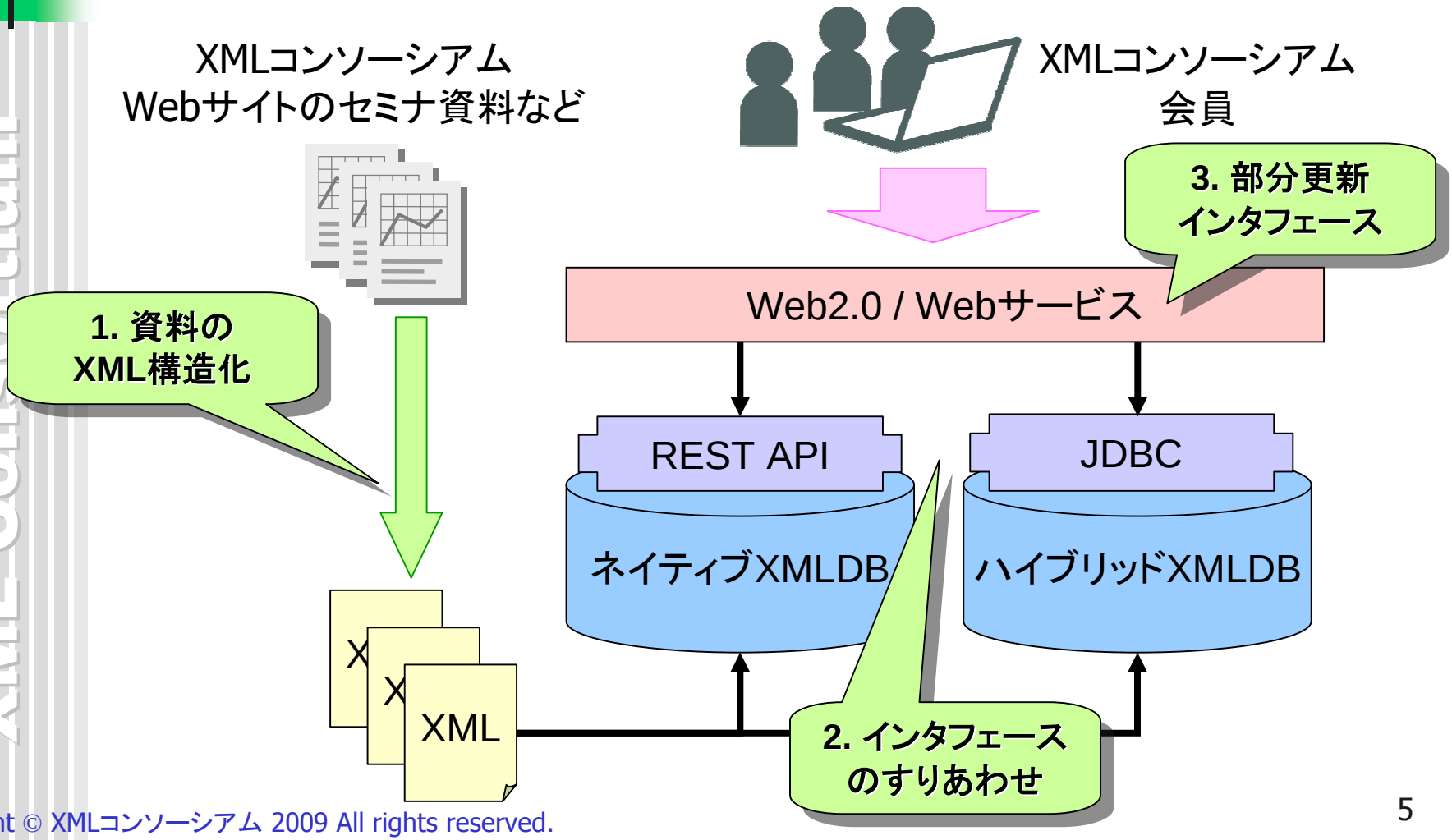
- XMLDB部会での一コマ
  - 第7回 XMLコンソーシアムWeek
    「フューチャWebサイト構想PJ」の課題検討
  - XML構造化にどのようなスキーマを使うべきだったか激論
  - 意見:「独自のXMLタグを作るべきではない！」
    - 新しく作るのはコストがかかる
    - 独自のスキーマだと、相手に渡した時に意味が伝わらない

既存のスキーマが簡単に参照できれば、
似て非なるタグが作られるのを避けられるのでは？

Copyright © XMLコンソーシアム 2009 All rights reserved.

※第7回 XMLコンソーシアム Week発表資料より引用
「フューチャWebサイト構想PJ XMLDB的、今後への展望」
矢野 令(東芝ソリューション)

(ご参考)フューチャWebサイト構想PJ 全体構成



Copyright © XMLコンソーシアム 2009 All rights reserved.

※第7回 XMLコンソーシアム Week発表資料より引用
「フューチャWebサイト構想PJ XMLDB的、今後への展望」
矢野 令(東芝ソリューション)

(ご参考) 資料のXML構造化

- 資料のXML構造化はツールにより半自動化
  - メタデータは手動で追加
- XMLは単純な構造を採用
  - テキスト情報のみを抽出して
    スライドごとに<page></page>で囲む

Dublin Core +
独自タグを
利用

- 最初から完全な構造を目指すのではなく、
  粗い構造から段階的に構造を詳細化する
- 将来的にメタデータの自動抽出に対応した
  XML構造化ができると便利

Copyright © XMLコンソーシアム 2009 All rights reserved.

目指すところ

- XMLスキーマの格納庫があり、
それをタグの粒度で自由に検索できると
課題の解決に繋がるのでは？

XMLスキーマをXMLDBで管理する
スキーマリポジトリ構想

コンテンツとしてXMLスキーマを
用いた、XMLDBの活用例

Copyright © XMLコンソーシアム 2009 All rights reserved.

XMLDBが最適な理由(1/2)

- XMLスキーマ自体はXML形式
- XMLスキーマの書式には様々なものがある[*]
  - Russian Doll
  - Salami Slice
  - Venetian Blind
  - Garden of Eden
    - それぞれ長所短所がある

[*]JavaOne Sanfrancisco 2006
"BOF-2224 What You Need to Know About Schema Design Patterns and Java Technology"

Copyright © XMLコンソーシアム 2009 All rights reserved.

(ご参考)Russian Dollの例

```
<xsd:schema>
  <xsd:element name="Person">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Name" type="xsd:string" />
        <xsd:element name="Lab" type="xsd:string" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

※XMLインスタンスの例：

```
<Person>
  <Name>東芝太郎</Name>
  <Lab>XMLDB</Lab>
</Person>
```

入れ子構造
で定義

Copyright © XMLコンソーシアム 2009 All rights reserved.

(ご参考)Garden of Edenの例

```
<xsd:schema>
  <xsd:element name="Person" type="PersonType" />
  <xsd:complexType name="PersonType">
    <xsd:element ref="Name" />
    <xsd:element ref="Lab" />
  </xsd:complexType>
  <xsd:element name="Name" type="xsd:string" />
  <xsd:element name="Lab" type="LabType" />
  <xsd:simpleType name="LabType">
    <xsd:restriction base="xsd:element">
      <xsd:enumeration value="XMLDB" />
      <xsd:enumeration value="Web2.0" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

要素も型も
個別にグローバ
ルに定義

Copyright © XMLコンソーシアム 2009 All rights reserved.

XMLDBが最適な理由(2/2)

- スキーマによって用いている書式が違う
  - 例えばTravelXMLはSalami Sliceを採用
  - 混在しているものもある
- それぞれ長所短所があるので、
  いずれかに統一するのは困難

そのまま柔軟に管理できるXMLDBが最適！

XML Consortium

Copyright © XMLコンソーシアム 2009 All rights reserved.

スキーマリポジトリのユースケース

- C:登録
  - スキーマのimport/includeの自動解決
- R:参照/検索
  - タグの検索、型で検索、階層で検索
  - annotationに書かれた説明文なども対象に全文検索
  - スキーマに対応するインスタンスの検索
- U:更新
  - スキーマのバージョン管理
- D:削除
  - XMLデータの論理削除

今回の
フォーカス

Copyright © XMLコンソーシアム 2009 All rights reserved.

参照/検索のユースケース(1/2)

- 要素名や属性名で検索する
  - 名前に”company”を含む要素または属性を検索
- 型(simpleType/complexType)を検索
  - xsd:string型を拡張した型を検索
  - その型を持つ要素や属性を検索
- 子要素/親要素にXXを持つ要素の検索
  - “Name”または”Lab”を子要素に持つ要素の検索
  - 親要素が”Person”である要素の子孫要素を全て検索

Copyright © XMLコンソーシアム 2009 All rights reserved.

参照/検索のユースケース(2/2)

- annotationにキーワードを持つ要素の全文検索
  - 例:「会員企業」で検索すると、”company”要素がヒット

```
<xsd:element name=“company” type=“xsd:string”>
  <xsd:annotation><xsd:documentation>
    登録されている会員企業を表す。
  </xsd:documentation></xsd:annotation>
</xsd:element>
```

- ある要素に該当するXMLインスタンスの検索
  - Lab要素として使われている値の例を検索

```
<Lab>XMLDB</Lab>
<Lab>Web2.0</Lab>
<Lab>SOA</Lab>
…
```

Copyright © XMLコンソーシアム 2009 All rights reserved.

デモイメージ

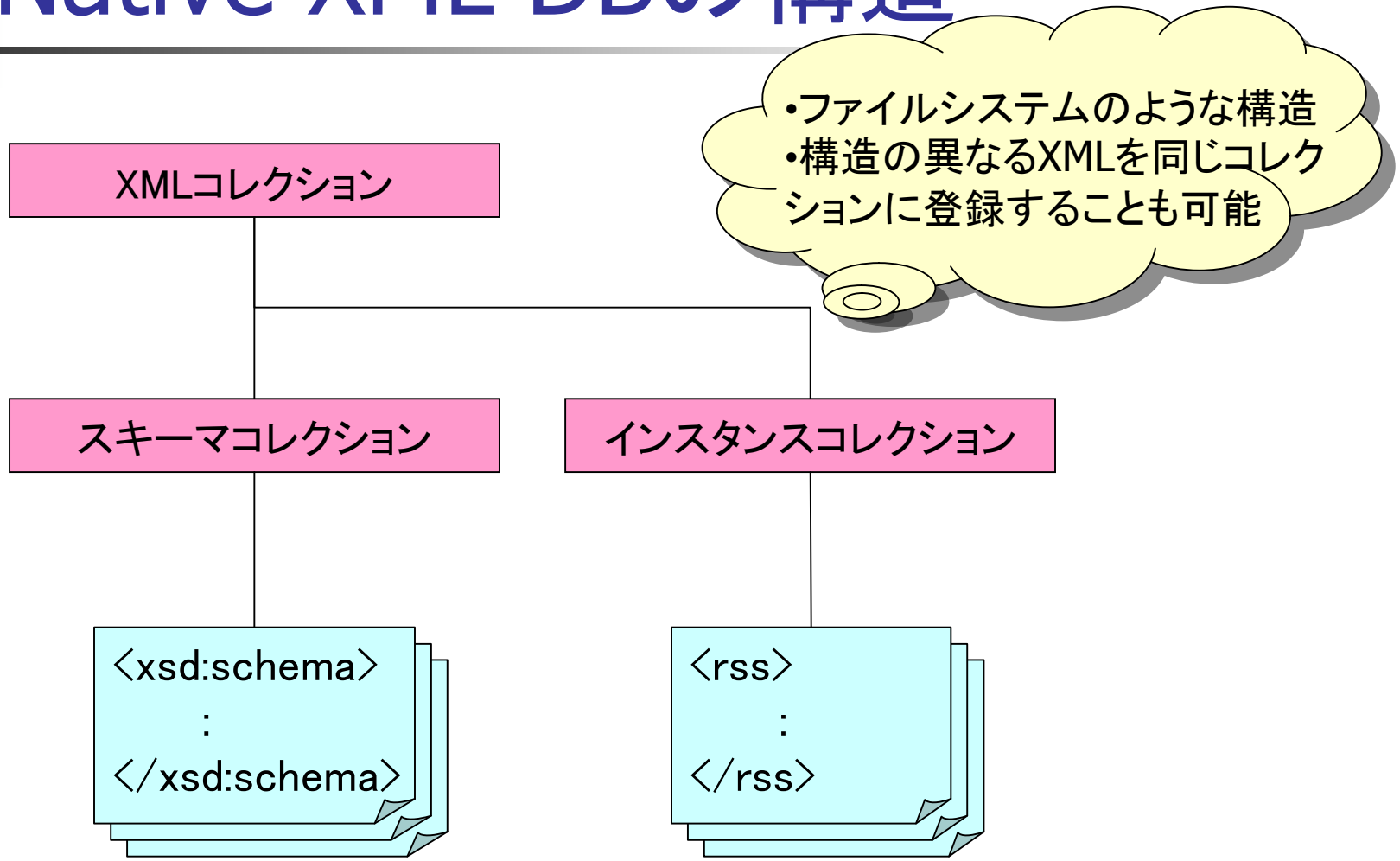
- 参照/検索のユースケースの一部を試作
  - 要素と属性の検索
  - annotationの全文検索
  - XMLインスタンスの検索
- XMLスキーマは一般公開されている下記を対象
  - TravelXML 1.4[*]
  - RSS 2.0
  - WSDL 2.0
  - UDDI 3.0

[*]TravelXML 仕様公開ページ
http://www.xmlconsortium.org/wg/TravelXML/TravelXML_index.html
Copyright (c) XML コンソーシアム 2006 All rights reserved.
Copyright (c) 社団法人日本旅行業協会 2006 All rights reserved.

- XMLDBには東芝ソリューションのTX1を利用
  - http://xml.toshiba-sol.co.jp/

Copyright © XMLコンソーシアム 2009 All rights reserved.

Native XML DBの構造



Copyright © XMLコンソーシアム 2009 All rights reserved.

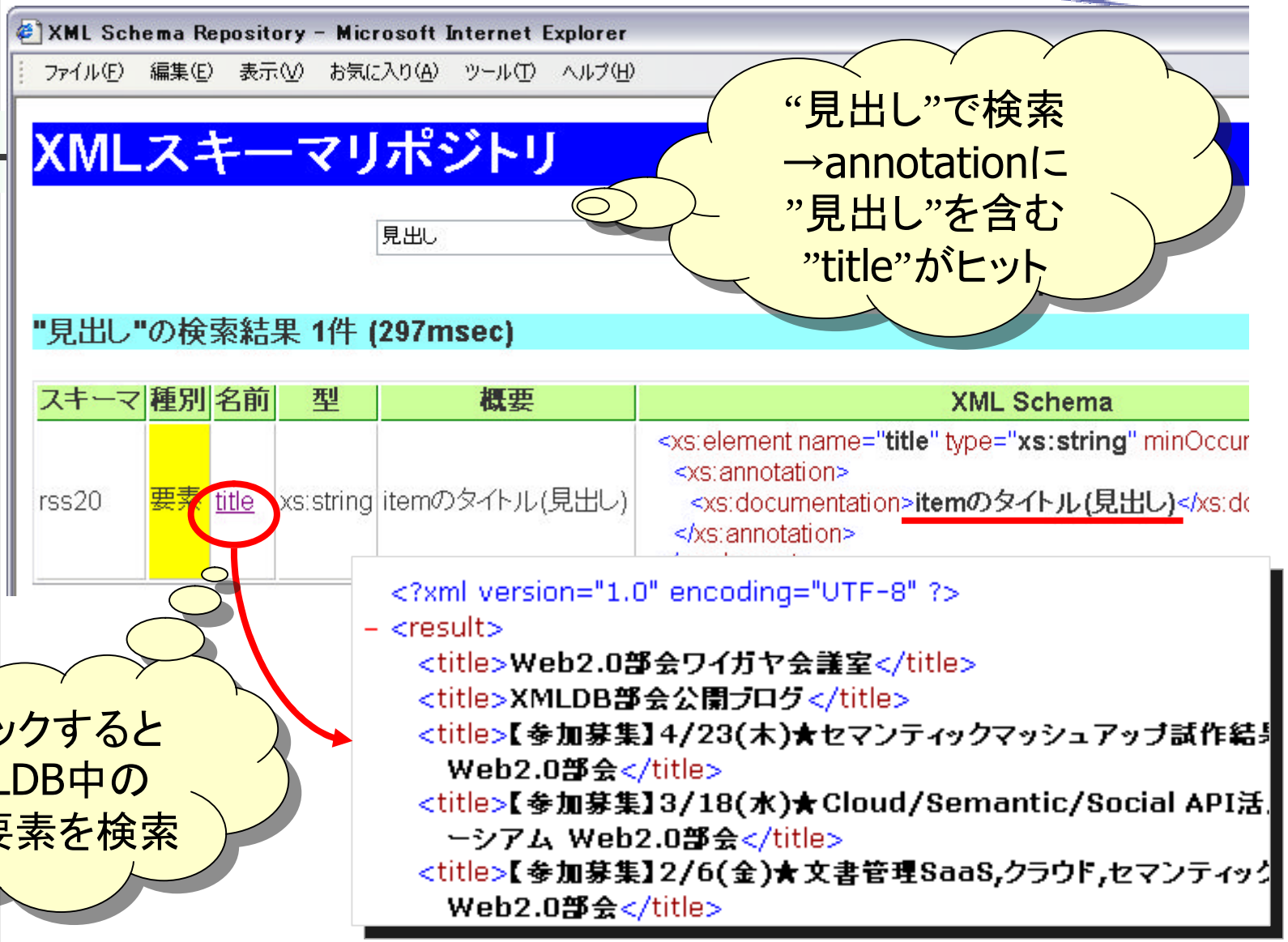
例：要素や属性名の検索

スキーマ	種別	名前	型	概要	XML Schema
rss20	要素	author	tEmailAddress	Email address of the author of the item.	<xs:element name="author" type="tEmailAddress" minOccurs="0"> <xs:annotation> <xs:documentation>Email address of the author of the item.</xs:documentation> </xs:annotation> </xs:element>
travelxml	要素	CreditCardAuthority	xs:string		<xs:element name="CreditCardAuthority" type="xs:string"></xs:element>
travelxml	要素	CreditCardAuthority	xs:string		<xs:element name="CreditCardAuthority" type="xs:string"></xs:element>
travelxml	要素	CreditCardAuthority	xs:string		<xs:element name="CreditCardAuthority" type="xs:string"></xs:element>
travelxml	要素	CreditCardAuthority	xs:string		<xs:element name="CreditCardAuthority" type="xs:string"></xs:element>
uddi30	要素	authorizedName	uddi:authorizedName		<xs:element name="authorizedName" type="uddi:authorizedName" final="restriction"></xs:element>

“author”で検索
→要素名に”author”
を含むものがヒット

Copyright © XMLコンソーシアム 2009 All rights reserved.

例：全文検索 / XMLインスタンスの検索



“見出し”で検索
→annotationに
”見出し”を含む
”title”がヒット

クリックすると
XMLDB中の
title要素を検索

※XMLコンソーシアム 部会公開ブログより

Copyright © XMLコンソーシアム 2009 All rights reserved.

まとめ

- もともとの課題を解決できそう！
- annotationがしっかり記述されていれば全文検索も有効
  - 自由に要素を追加することで高度な検索も可能になる
- スキーマがどのような書式かを気にする必要がない

コンテンツの柔軟な管理、コンテンツを育てる、XMLDBの本領発揮

- 課題
  - Russian Dollのようにツリーで定義されたものはXPath/XQueryで処理しやすい
  - 一方、Garden of Edenのように細切れで参照関係を辿るものは処理が手間

Copyright © XMLコンソーシアム 2009 All rights reserved.

ご静聴ありがとうございました

TX1は東芝ソリューション株式会社の登録商標です。
その他、本書に記載されている会社名、商品名、製品名は、一般に各社の登録商標または商標です。

Copyright © XMLコンソーシアム 2009 All rights reserved.