LexML Brasil

Parte 4 – Coleta de Metadados

Versão 0.7

Brasília – Agosto de 2008

LexML Brasil

Parte 4 – Coleta de Metadados

Versão 0.7 (agosto/2008)

1. Introdução

A coleta de metadados tem por objetivo reunir os metadados de documentos legislativos e jurídicos disponíveis nos sítios dos diversos órgãos governamentais. Serão coletados, inicialmente, apenas metadados de identificação (epígrafe, apelidos, identificadores, etc.) e a ementa (metadado descritivo).

Como forma de facilitar e automatizar o processo de coleta de metadados foi escolhido o Protocolo OAI-PMH (*Open Archives Information – Protocol for Metadata Harvest*).

As próximas seções apresentam os principais componentes da arquitetura OAI-PMH, a aplicação deste protocolo para o Projeto LexML e algumas explicações sobre o *XML Schema* lexml_oai.xsd (Anexo 1).

2. Protocolo OAI-PMH

A arquitetura de uma rede de informações que utiliza o Protocolo OAI-PMH para intercâmbio de metadados é formada por nodos de três tipos (Figura 1):

- Provedor de Dados (*Data Provider*) – serviço responsável pela exposição de metadados;

- Provedor de Serviço (*Service Provider*) – serviço responsável pela comunicação com os nodos provedores e agregadores de dados, pelo processamento dos dados coletados e pela oferta de serviços de pesquisa.

- Agregador de Dados (*Data Aggregator*) – serviço responsável por agregar metadados coletados de Provedores de Dados e disponibilizá-los para um Provedor de Serviço.

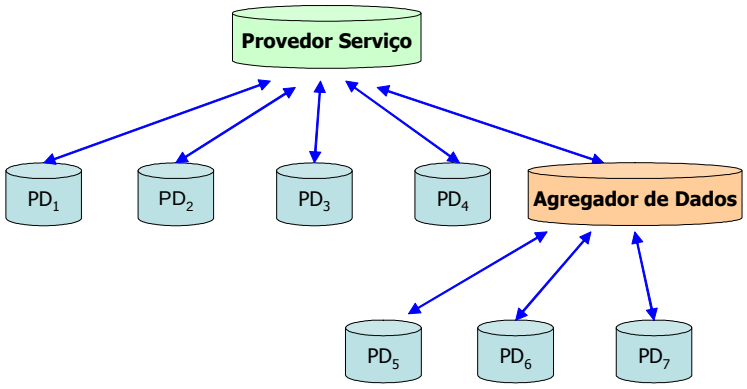

Figura 1. Tipos de Nodos de uma rede OAI-PMH.

O OAI-PMH caracteriza-se pela simplicidade dos comandos (apenas 6 verbos) e pela fácil integração a qualquer ambiente computacional pois é baseado apenas no protocolo HTTP (*Hypertext Transfer Protocol*) e no formato XML.

Cada registro de metadado é composto por um cabeçalho (dados de identificação do protocolo), um corpo (metadado propriamente dito) e, opcionalmente, uma seção com informações de proveniência do registro.

3. Protocolo OAI-PMH aplicado ao Projeto LexML

Para atender aos requisitos do Projeto LexML foram definidas algumas convenções para a implementação do protocolo OAI-PMH. Essas convenções são de várias naturezas e tem como motivação a simplificação do processo de intercâmbio de metadados ao mesmo tempo em que se preocupa com questões como a persistência dos recursos disponibilizados.

3.1. Papéis no Provedor de Dados

O nodo provedor de dados pode possuir, além do administrador responsável pela monitoração do fluxo da coleta de metadados (normalmente alguém com o conhecimentos de informática), vários publicadores que são responsáveis pelo recurso disponibilizado e pela garantia de sua persistência (associação entre URLs válidas com URNs). Os publicadores são normalmente os responsáveis pelos sistemas que disponibilizam informações legislativas e jurídicas.

A cada publicador é associado um perfil onde são identificados os possíveis tipos de documentos, com respectivas autoridades emitentes, passíveis de publicação.

3.2. Cabeçalho do Registro

A Figura 2 apresenta um exemplo de cabeçalho de registro compatível com o protocolo OAI-PMH.

```
<header>
     <identifier>oai:ssinf.senado.gov.br:njur/102415</identifier>
     <datestamp>2008-07-08-10:20:20:002221</datestamp>
 </header>
```

Figura 2. Cabeçalho Registro OAI – PMH.

O elemento `<identifier>` é um identificador unívoco de um recurso disponibilizado no sistema de origem. O LexML convencionou o seguinte formato para esse campo:

"oai:" [orgao .] domínio ":" sistema "/" identificador interno [";" detalhe]

Após a constante “oai:”, o campo órgão identifica, opcionalmente, o publicador. Caso existe apenas um responsável pela publicação das informações, esse campo poderá ser omitido. Na seqüência, é identificado o domínio ao qual o provedor de dados está relacionado. Após o caractere “:” é identificado o sistema de informações origem do recurso e, após a “/” o identificador interno utilizado pelo sistema. Como veremos a seguir, esse identificador interno poderá ser detalhado para indicar recurso complementares (ex.: anexos, retificações) que estão localizados junto ao recurso principal.

O elemento `<datestamp>` identifica a data e hora da última atualização do registro na base do provedor de dados. Por exemplo, caso o publicador perceba que a ementa de um documento está com erro, ao corrigir a ementa no seu sistema (no exemplo, NJUR), a atualização deverá ser refletida (de forma automática) no registro respectivo do provedor de dados. Nesse caso, o `datestamp` é alterado para a data/hora dessa atualização.

3.3. Corpo do Registro de Metadado

A Figura 3 apresenta um exemplo de corpo de registro de metadados segundo as convenções do LexML.

```
<metadata>
  <LexML xmlns="http://www.lexml.gov.br/">

     <Item formato="text/html">
        http://www6.senado.gov.br/legislacao/ListaPublicacoes.action?id=102415
     </Item>

     <DocumentoIndividual>
       urn:lex:br:federal:lei:1990-09-11;8078@1990-09-12!1990-09-12~texto;pt-br
     </DocumentoIndividual>

     <Epigrafe>Lei nº 8.078, de 11 de setembro de 1990</Epigrafe>
     <Apelido>Código de Defesa do Consumidor</Apelido>
     <Apelido>Código de Proteção e Defesa do Consumidor</Apelido>
     <Apelido xml:lang="es">
        Código de Protección y Defensa del Consumidor
     </Apelido>

     <Ementa>
        Dispõe sobre a proteção do consumidor e dá outras providências
     </Ementa>

    <Relacionamento tipo="publicacao.oficial">
urn:lex:br:imprensa.nacional:publicacao.oficial;diario.oficial.uniao;secao.1:1990-09-
12;123:pag1
    </Relacionamento>

  </LexML>
</metadata>
```

Figura 3. Corpo do Registro de Metadados.

O elemento `<Item>` possui a URL do recurso disponibilizado na Internet sob a responsabilidade do publicador. A cada `<Item>` deve-se, obrigatoriamente, relacionar uma URN no elemento `<DocumentoIndividual>`. Esse relacionamento posiciona o recurso publicado no espaço de nomes definidos pela Parte 2 das especificações LexML. Na seqüência, são relacionados a Epígrafe, os Apelidos registrados para o documento e a Ementa. Por fim, sempre que possível, deve-se relacionar ao Documento Individual identificado a URN da publicação oficial que o veiculou.

Os elementos textuais (Epigrafe, Apelido e Ementa) possui o atributo `xml:lang` para especificação da língua. Esse atributo tem por default o valor “pt-BR”.

Caso uma URL contenha mais de um Documento Individual (por exemplo, a norma e seus anexos), deve-se criar um registro para cada Documento Individual relacionado. Nesse caso, pode-se acrescentar ao `<Identifier>` um detalhamento que diferencie esse registro dos demais.

3.4. Tipos de Relacionamentos

Além do relacionamento “publicação.oficial” apresentado no exemplo anterior, é possível definir ainda o relacionamento “sucessor.logico.de” e “equivalente.a”.

O relacionamento “sucessor.logico.de” deve ser utilizado para estabelecer relacionamentos entre documentos complexos. Por exemplo, no caso de processos que tramitam em diferentes tribunais é possível relacionar qual o identificador do processo no tribunal de origem. Dessa forma, caso cada publicador faça a relação com o Documento Complexo imediatamente anterior, é possível montar todas as relações entre os processos de todas as instâncias de um determinado caso.

3.5. Publicador Oficial

O órgão “Publicador Oficial” de um documento, seja a publicação realizada em papel ou em meio digital, ao se integrar à Rede de Informações LexML deverá preencher os metadados indicando a URN do Documento Individual como sendo a da publicação oficial. A Figura 4 apresenta um exemplo deste caso.

```
<metadata>
  <LexML xmlns="http://www.lexml.gov.br/">
     <Item formato="application/pdf" paginas="140">
https://www.in.gov.br/imprensa/visualiza/index.jsp?jornal=do&secao=1&pagina=1&data=10/01
/2007
     </Item>

     <DocumentoIndividual>
 urn:lex:br:imprensa.nacional:publicacao.oficial;diario.oficial.uniao;secao.1:2007-01-
10;133
     </DocumentoIndividual>

  </LexML>
</metadata>
```

Figura 4. Registro de Metadado do órgão Publicador Oficial.

4. Sobre o Esquema lexml_oai.xsd

O esquema lexml_oai.xsd está organizado em duas partes.

Na parte inicial, são definidos os elementos e atributos que compõem a instância de um registro. Na parte final, são definidos os tipos utilizados pelas definições da parte inicial.

5. Referências

5.1. Bibliográficas

5.2. Sites

http://www.openarchives.org/OAI/openarchivesprotocol.html - OAI-PMH

6. Lista de Abreviaturas e Siglas

OAI-PMH – Open Archives Information – Protocol for Metadata Harvest

HTTP – Hypertext Transfer Protocol

7. Créditos

7.1. Editores

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

Fernando Ciciliati (Senado Federal / Interlegis)

7.2. Grupo de Trabalho LexML (em Agosto de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda (Tribunal de Contas da União)

Flávio Henrique Rocha e Silva (Supremo Tribunal Federal)

Flávio Heringer (Senado Federal)

Manuel de Medeiros Dantas (Advocacia Geral da União)

Jean Rodrigo Ferri (Senado Federal / Interlegis)

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

João Batista de Holanda Neto (Senado Federal / Prodasen)

João R. Kramer Santana (Tribunal de Contas da União)

Paulo André Mattos de Carvalho (Tribunal de Contas da União)

Paulo Martins Inocêncio (Conselho da Justiça Federal)

Ricardo Bravo (Tribunal de Contas da União)

Sérgio Falcão (Câmara dos Deputados / CENIN)

Virgínia Azevedo (Supremo Tribunal Federal)

Anexo 1 – Esquema lexml_oai.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:lexml="http://www.lexml.gov.br/oai_lexml"
    xmlns:xml="http://www.w3.org/XML/1998/namespace"
    targetNamespace="http://www.lexml.gov.br/oai_lexml"
    elementFormDefault="qualified" attributeFormDefault="unqualified" xml:lang="PT">
      <xs:import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="http://www.w3.org/2001/xml.xsd"/>
       <!-- Parte 1  - Definição dos Elementos →
      <xs:element name="LexML" type="lexml:LexMLType">
        <xs:annotation>
         <xs:documentation>
             XML Schema - OAI LexML
          </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:complexType name="LexMLType">
        <xs:annotation>
         <xs:documentation>
                  Cada instância deste elemento corresponde a um Item que poderá ser relacionado
                  a um identificador de documento (principal ou anexo) e a outros identificadores (publicação oficial,
                   sucesso lógico, equivalente a).
                        </xs:documentation>
        </xs:annotation>
        <xs:sequence>
         <xs:element name="Item">
              <xs:annotation>
                        <xs:documentation>URL do Item com atributo Formato (subconjunto dos valores da tabela de mime-types)
</xs:documentation>
              </xs:annotation>
              <xs:complexType>
                        <xs:simpleContent>
                                  <xs:extension base="xs:anyURI">
                                            <xs:attribute name="formato" type="lexml:FormatoIdentificadorItemType"
use="required"/>
                                  </xs:extension>
                        </xs:simpleContent>
              </xs:complexType>
         </xs:element>
         <xs:element name="DocumentoIndividual">
              <xs:annotation>
                        <xs:documentation>URN no padrão URN LexML
                        Informar a URN do documento individual contido no Item.
                        No caso do Item conter mais de um documento individual, deve-se gerar um registro para cada um.
                        </xs:documentation>
              </xs:annotation>
         </xs:element>
         <xs:element name="Relacionamento" type="lexml:RelacionamentoType" minOccurs="0" maxOccurs="unbounded">
              <xs:annotation>
```

```
                    <xs:documentation>URN no padrão URN LexML adicionada de atributo TipoRelacionamento para indicar se é uma "publicacao.oficial", "sucessor.logico" ou "equivalente.a"</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="Epigrafe" type="lexml:CampoTextoComIdiomaType" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element name="Apelido" type="lexml:CampoTextoComIdiomaType" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element name="Ementa" type="lexml:CampoTextoComIdiomaType" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>

        <!-- Parte 2  - Definição dos Tipos →
        <xs:complexType name="RelacionamentoType">
          <xs:simpleContent>
            <xs:extension base="lexml:URNType">
                <xs:attribute name="tipo" type="lexml:TipoRelacionamentoType"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
        <xs:complexType name="CampoTextoComIdiomaType" abstract="false">
          <xs:annotation>
            <xs:documentation>Campo Texto com atributo de idioma default</xs:documentation>
          </xs:annotation>
          <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute ref="xml:lang" use="optional" default="pt-BR"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
        <xs:simpleType name="FormatoIdentificadorItemType">
          <xs:annotation>
            <xs:documentation>
            Subconjunto mais frequente de tipos mime.
            IANA - MIME MEDIA TYPES
            http://www.iana.org/assignments/media-types/
            </xs:documentation>
          </xs:annotation>
          <xs:restriction base="xs:string">
            <xs:enumeration value="application/mpeg4-generic"/>
            <xs:enumeration value="application/mp4"/>
            <xs:enumeration value="application/msword"/>
            <xs:enumeration value="application/ogg"/>
            <xs:enumeration value="application/pdf"/>
            <xs:enumeration value="application/postscript"/>
            <xs:enumeration value="application/rtf"/>
            <xs:enumeration value="application/sgml"/>
            <xs:enumeration value="application/xhtml+xml"/>
            <xs:enumeration value="application/xml"/>
            <xs:enumeration value="application/zip"/>
            <xs:enumeration value="application/vnd.oasis.opendocument.chart"/>
            <xs:enumeration value="application/vnd.oasis.opendocument.image"/>
```

```
        <xs:enumeration value="application/vnd.oasis.opendocument.text-web"/>
        <xs:enumeration value="application/vnd.oasis.opendocument.text-web"/>
        <xs:enumeration value="audio/ac3"/>
        <xs:enumeration value="audio/mp4"/>
        <xs:enumeration value="audio/mpeg"/>
        <xs:enumeration value="audio/mpeg4-generic"/>
        <xs:enumeration value="audio/ogg"/>
        <xs:enumeration value="image/gif"/>
        <xs:enumeration value="image/png"/>
        <xs:enumeration value="image/jpeg"/>
        <xs:enumeration value="image/tiff"/>
        <xs:enumeration value="image/bmp"/>
        <xs:enumeration value="text/plain"/>
        <xs:enumeration value="text/html"/>
        <xs:enumeration value="text/xml"/>
        <xs:enumeration value="text/sgml"/>
        <xs:enumeration value="text/rtf"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TipoRelacionamentoType">
      <xs:annotation>
       <xs:documentation>Enumerações possíveis para o atributo TipoRelacionamento</xs:documentation>
      </xs:annotation>
      <xs:restriction base="xs:string">
       <xs:enumeration value="sucessor.logico.de"/>
       <xs:enumeration value="publicacao.oficial"/>
       <xs:enumeration value="equivalente.a"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="URNType">
      <xs:annotation>
       <xs:documentation>Tipo URN</xs:documentation>
      </xs:annotation>
      <xs:restriction base="xs:anyURI">
       <xs:pattern value="urn:lex(:\S)+(:\S)+"/>
      </xs:restriction>
    </xs:simpleType>
</xs:schema>
```