



LexML Brasil
Apresentação

Versão 1.0 (RC1)
Brasília – Dezembro de 2008

LexML Brasil

Apresentação

Versão 1.0 (Dezembro/2008)

LexML Brasil	2
1. Introdução.....	3
2. Projeto LexML Brasil.....	4
2.1. Objetivo Geral	4
2.2. Objetivos Específicos	4
2.3. Princípios Regentes	4
2.4. Projetos Similares no Exterior.....	4
2.5. Histórico do Projeto LexML Brasil.....	6
3. Identificadores de Normas, Julgados e Projetos de Norma.....	7
3.1. Exemplo de Identificador Persistente	8
4. Estrutura de Normas Jurídicas.....	8
5. Articulação segundo a Lei Complementar nº 95.....	10
5.1. Agrupamento de Artigos e Artigo	10
5.2. Detalhamento de Artigo	11
6. Especificação do Projeto LexML Brasil.....	12
7. Referências	12
7.1. Bibliográficas	12
7.2. Sítios da Internet.....	13
8. Lista de Abreviaturas e Siglas	13
9. Créditos.....	13
9.1. Editores.....	13
9.2. Grupo de Trabalho LexML (em Dezembro de 2008).....	14
9.3. Outras Contribuições	14
9.4. Criação da Logomarca LexML	15

1. Introdução

A Constituição Federal de 1988, no art. 5º, inciso XIV, define o “acesso à informação” como um direito fundamental do cidadão. Vários órgãos dos três poderes e das três esferas de governo disponibilizam informações legislativas e jurídicas em sítios da Internet. A criação de uma infra-estrutura que permita organizar e integrar a grande quantidade de informações legislativas e jurídicas existente facilitará o acesso à informação e contribuirá para a realização desse direito fundamental. Nesse sentido, foi criado o Projeto LexML Brasil que é uma iniciativa conjunta de diversos órgãos participantes de um Grupo de Trabalho da Comunidade TI Controle em busca do estabelecimento de padrões abertos, integração de processos de trabalho e compartilhamento de dados de interesse comum, que permitam a *identificação* e a *estruturação* de informações legislativas e jurídicas.

Na organização de um acervo (conjunto de itens) é necessário definir, entre outras coisas, um critério de identidade, ou seja, que tipos de itens devem ser considerados e que parâmetros identificam univocamente cada item. Por exemplo, no caso de normas jurídicas, é possível considerar os seguintes tipos de itens: a) a norma, partindo de uma perspectiva histórica, considerando toda a sua evolução no tempo; b) a versão do texto de uma norma para uma determinada data; c) o dispositivo de uma versão específica da norma. Para cada um destes três níveis de granularidade listados, é possível definir um grupo de elementos (datas, tipos, números sequenciais etc.) para a composição de um identificador unívoco. Raciocínio análogo aplica-se às proposições legislativas e aos julgados.

As novas tecnologias da informação e comunicação permitem uma melhor organização e gerência de informações textuais. Com a criação, em 1998, da metalinguagem XML (*eXtensible Markup Language*), passou a ser possível a definição de tipos de documentos textuais estruturados com a possibilidade de validação da estrutura do texto em relação a um conjunto de regras. Em diversas áreas surgiram linguagens de estruturação de documentos para atender a necessidades específicas como, por exemplo, nos casos da Matemática (MathML - *Mathematical Markup Language*), da Química (ChemML - *Chemical Markup Language*) e da Geografia (GML - *Geography Markup Language*). No caso de normas jurídicas, utilizando esta tecnologia, é possível verificar, por exemplo, se um determinado texto está articulado de acordo com a técnica legislativa. Em alguns países surgiram linguagens de marcação baseadas em XML para estruturar o texto de normas jurídicas e proposições legislativas. Outra importante característica da metalinguagem XML é a interoperabilidade, ou seja, a capacidade de intercâmbio de dados e informações de forma independente da tecnologia ou plataforma utilizada pelos participantes de uma comunidade.

O novo Portal da Rede de Informação Legislativa e Jurídica (<http://www.lexml.gov.br>) viabiliza um novo ponto de acesso unificado às diversas fontes de informação legislativa e jurídica do governo.

2. Projeto LexML Brasil

2.1. Objetivo Geral

O Projeto LexML Brasil tem por objetivo organizar a informação legislativa e jurídica disponibilizada em forma digital pelos vários órgãos dos Poderes Executivo, Legislativo e Judiciário, Advocacia Geral da União e Ministério Público, nas esferas Federal, Estadual, Municipal e Distrital, considerando o acervo de informações do passado, melhorando o processo de geração de novas informações no presente e se preocupando com a preservação da informação digital para o futuro.

2.2. Objetivos Específicos

- **Identificar**, de forma unívoca e persistente, os recursos de informação legislativa e jurídica.
- **Estruturar** o inteiro teor dos documentos utilizando o formato XML.

2.3. Princípios Regentes

O desenvolvimento da Rede de Informações LexML irá respeitar os seguintes princípios:

- A consulta e a referência ao acervo da Rede de Informações LexML será sempre gratuita.
- Apenas entidades e órgãos da administração pública direta e indireta nas esferas federal, estadual, municipal e distrital poderão integrar a Rede LexML como provedores de informações e serviços.
- Os integrantes da Rede de Informações LexML terão acesso ao acervo de informações consolidado e agrupado pelos identificadores uniformes.

2.4. Projetos Similares no Exterior

As recomendações do Projeto LexML reúnem as melhores práticas de iniciativas para identificação e estruturação de documentos legislativos e jurídicos, adaptando as experiências bem sucedidas do exterior às necessidades do nosso ordenamento jurídico e da nossa técnica legislativa. A Tabela 1 apresenta diversas iniciativas de aplicação do XML à informação legislativa e jurídica no mundo.

Devido à influência do Direito Romano na organização do nosso ordenamento jurídico, as iniciativas de países que adotam o sistema *civil law* são mais aplicáveis à nossa realidade do que as de países que adotam outros sistemas legais. A Itália é um dos países com maior evolução na aplicação de novas tecnologias para a estruturação e organização de normas jurídicas no formato XML. O Projeto *Norme in Rete* (NIR), coordenado pelo CNIPA (*Centro Nazionale per Informatica nella Pubblica Amministrazione*), e realizado com o suporte do ITTIG (*Istituto di Teoria e Tecniche dell'Informazione Giuridica*) e da Universidade de Bolonha, é um projeto implantado desde 2002 com resultados expressivos.

Basicamente, no projeto italiano NIR, foram definidos a estrutura do documento utilizando a sintaxe DTD (*Document Type Definition*), a estrutura dos identificadores utilizando o padrão URN (*Uniform Resource Name*) e um portal de informações (<http://www.nir.it>). O padrão DTD do NIR trata da definição da articulação de documentos, estruturando o inteiro teor dos documentos no formato XML. O padrão URN do NIR trata da identificação de documentos, requisito essencial para permitir a criação de remissões entre documentos. O portal do projeto NIR oferece um serviço de pesquisa e, no resultado, relaciona os *links* disponíveis para um determinado documento jurídico nos órgãos governamentais da Itália.

Tabela 1. Iniciativas similares em outros países.

Localidade	Projeto	Observações
África	Akoma Ntoso	Projeto promovido pelas Nações Unidas no contexto do projeto “Fortalecimento dos Sistemas de Informação dos Parlamentos na África”. Define um esquema para a estruturação do texto integral de normas e projetos de lei.
Áustria	eLaw	Consiste de um sistema de <i>workflow</i> que acompanha todo o processo de formação da lei, desde a iniciativa até a publicação em diário oficial eletrônico.
Dinamarca	LexDania	Utiliza camadas estratificadas que permitem a montagem de múltiplos esquemas, de acordo com uma metodologia que garante a construção de esquemas compatíveis e consistentes.
Europa	Metalex CEN	O Centro Europeu de Normatização (CEN) está definindo um meta-esquema que servirá de denominador comum em relação aos diversos esquemas nacionais.
Holanda	BWB SDU	Desenvolvido, inicialmente, por uma empresa que publicava os atos oficiais, é mantido atualmente pelo governo.
	Metalex	Esquema flexível e extensível para a codificação de normas em XML, influenciou fortemente o Metalex CEN.
Itália	Norme in Rete	Consiste na definição de um identificador uniforme (URN), um formato para estruturar o inteiro teor de normas (DTD e <i>XML Schema</i>) e um portal para a resolução de identificadores (tradução de URN em URLs).
Suíça	CHLexML	Define <i>XML Schema</i> para ser utilizado na publicação de todos os atos legislativos, nos níveis federal, estadual (27 cantões) e local. O sistema lida com o problema de múltiplas línguas oficiais e múltiplos sistemas de classificação (1 para cada cantão).
Tasmânia (Austrália)	EnAct	Implementação de sistema que permite a recuperação do texto vigente da norma para uma determinada data.

A experiência da criação do padrão DTD do NIR subsidiou a definição do *XML Schema* para documentos legislativos e normativos do projeto AKOMA NTOSO (<http://www.akomantoso.org/>). O Akoma Ntoso é um projeto promovido pelas Nações

Unidas e realizado com o suporte de pesquisadores da Universidade de Bolonha com o objetivo de capacitar os parlamentos da África com novas tecnologias de informação e comunicação

O Projeto LexML Brasil é fortemente baseado nas experiências bem sucedidas dos projetos *Norme in Rete* e *Akoma Ntoso*.

2.5. Histórico do Projeto LexML Brasil

2.5.1. Antecedentes

O uso de XML para estruturar normas passou a ser investigado no ano de 2000 no PRODASEN (Processamento de Dados do Senado Federal). No VI CONIP (Congresso Nacional de Informática Pública), evento realizado no período de 23-25 de maio de 2000 em São Paulo, foi apresentada a palestra “Legislação Brasileira na Internet” (Lima & Souza, 2000) com uma proposta de estruturação de textos normativos utilizando XML.

No ano de 2002, foi disponibilizada no sítio do Senado Federal a Constituição Federal de 1988 (<http://www.senado.gov.br/sf/legislacao/const/>) utilizando XML. Neste sítio, o cidadão pode escolher, em uma linha do tempo, qual a edição do texto constitucional que deseja consultar. A compilação do texto de cada edição foi realizada de forma automática a partir do texto original promulgado em 1988 e dos textos das emendas constitucionais.

2.5.2. Início do Projeto LexML Brasil

A partir de novembro de 2005, por ocasião do X ENIAL (Encontro Nacional de Informática Aplicada ao Legislativo), evento realizado em Domingos Martins (ES), a proposta de uso do XML para documentos legislativos e jurídicos no Brasil passou a ser denominada “Projeto LexML Brasil”. Nessa época o projeto contava com a participação do PRODASEN e do Interlegis.

O termo “LexML” é uma fusão da palavra *lex* (do latim, lei) e do acrônimo XML (*eXtensible Markup Language*). Em alguns países da Europa, tais como Alemanha (www.lexml.de) e Itália (www.lexml.it), o termo “LexML” denomina a comunidade de interesse no assunto XML e Legislação.

No segundo semestre de 2006, foi formado um Grupo de Trabalho com representantes de diversos órgãos da Comunidade TI Controle (www.ti.controle.gov.br).

Os anos de 2006 e 2007 foram determinantes para o desenvolvimento do modelo conceitual destinado à organização da informação legislativa e jurídica do Projeto LexML Brasil. Nesse período, foi intensificado o relacionamento com professores da Universidade de Bolonha (Dr. Fabio Vitali e Dr^a. Monica Palmirani), e pesquisadores do ITTIG (Dr. Pierluigi Spinosa e Dr. Enrico Francesconi).

Em fevereiro de 2007, a convite do Supremo Tribunal Federal, o Dr. Pierluigi Spinosa apresentou a experiência de identificadores uniformes do projeto *Norme In Rete* no evento I SIJED (Seminário de Gestão da Informação Jurídica em Espaços Digitais).

Em abril de 2008, as reuniões do Grupo de Trabalho LexML foram retomadas com periodicidade mensal.

Em agosto de 2008, com o apoio da Secretaria de Logística e Tecnologia da Informação do Ministério do Planejamento, Orçamento e Gestão, foi realizada uma Consulta Pública da especificação do Projeto LexML no Portal do Governo Eletrônico (<http://www.governoeletronico.gov.br/consulta-publica>).

3. Identificadores de Normas, Julgados e Projetos de Norma

A remissão textual é muito utilizada nos documentos legislativos e jurídicos. Por exemplo, uma norma jurídica pode fazer referências a outras normas, seja para realizar alterações, revogar um dispositivo ou regulamentar uma matéria; um acórdão pode referenciar normas jurídicas bem como outros acórdãos e súmulas.

As remissões textuais podem ser classificadas de diversas formas:

- Quanto à localização do alvo de uma remissão, podemos identificar dois tipos: as remissões internas e as externas. As remissões internas têm como alvo o próprio documento. As externas referenciam outros documentos.
- Quanto à granularidade de uma remissão externa, podemos identificar dois tipos: a remissão ao documento no todo e a remissão a uma parte do documento.
- Quanto à dimensão temporal de uma remissão externa, podemos identificar dois tipos: a remissão a uma versão específica de uma norma e a remissão à norma de uma forma genérica, sem especificar uma versão específica.

Os identificadores servem também para estabelecer referências não explícitas no documento de origem. Por exemplo, é possível definir, na seção de metadados, um relacionamento entre duas normas correlatas, mesmo que não exista remissão textual entre as normas.

Para representar corretamente as remissões é necessário criar um identificador que qualifique unicamente cada possível alvo de uma remissão textual. Na impossibilidade de criar um sistema único de numeração para todo o acervo de normas, julgados e proposições legislativas do Brasil, fez-se necessária a criação de um identificador que seja compatível com os diversos sistemas de numeração existentes.

O Projeto LexML define um identificador unívoco e persistente para os documentos legislativos e jurídicos. A persistência possibilita que os alvos sejam encontrados, mesmo que ocorra mudança de endereços da Internet (URLs – *Universal Resource Locator*), evitando assim o conhecido Erro 404 (Recurso não encontrado). Para que a

persistência se realize é necessário que os provedores de dados informem a nova localização do recurso quando da mudança de localização.

3.1. Exemplo de Identificador Persistente

O identificador LexML permite a integração entre as fontes de informações legislativas e jurídicas pois define um referencial único para esse tipo de documento.

A lista abaixo relaciona endereços da Internet (URLs) onde é possível consultar o texto da Lei nº 8.666/1993 nos sítios do Senado Federal, da Câmara dos Deputados e da Presidência da República, respectivamente. É possível notar que cada identificador possui regras de formação diferentes e está diretamente relacionado à solução tecnológica adotada pela instituição em um determinado momento.

- <http://www6.senado.gov.br/legislacao/ListaPublicacoes.action?id=102466>
- <http://www2.camara.gov.br/internet/legislacao/legin.html/textos/visualizarTexto.html?ideNorma=322221&seqTexto=1&PalavrasDestaque=>
- <http://www.planalto.gov.br/ccivil/Leis/L8666cons.htm>

Caso ocorra mudança na tecnologia utilizada ou alteração na convenção de nomes por parte dos órgãos gestores desses recursos, todos os links dos documentos que referenciavam esses identificadores irão apresentar Erro 404 (recurso não encontrado) ao serem ativados.

O identificador LexML utiliza a notação URN que permite a criação de identificadores persistentes, pois estes não estão atrelados a detalhes de tecnologia ou à convenção de nomes definidos de forma unilateral. As URNs seguem um esquema previamente acordado entre os participantes da rede de informações. Veja abaixo a URN da Lei nº 8.666/1993.

urn:lex:br:federal:lei:1993-06-21;8666

Ao referenciar um recurso, o cidadão poderá omitir detalhes, como, por exemplo, o mês e o dia da data de assinatura, ou ainda poderá utilizar apelidos atribuídos às normas, como nos exemplos abaixo.

urn:lex:br:federal:lei:1993;8666 (*apenas ano*)
urn:lex:br:federal:lei:lei.licitacoes (*apelido*)

4. Estrutura de Normas Jurídicas

A preocupação com a técnica legislativa no Brasil remonta ao início do século passado. Aurelino Leal, autor da obra “*Technica Constitucional Brasileira*” (1914), aponta falta de consistência na numeração de dispositivos da própria Constituição Federal de sua época e defende uma melhor técnica legislativa.

A promulgação, em 1988, da Constituição Federal previu uma lei complementar que trataria da “elaboração, redação, alteração e consolidação das leis”. Esta lacuna foi

preenchida dez anos após, com a publicação da Lei Complementar nº 95/1998, alterada pela Lei Complementar nº 107/2001. Essas normas representaram um grande avanço na uniformização da técnica legislativa. No entanto, mesmo com a ampla aceitação, é fácil encontrar textos de leis não aderentes às regras de articulação de dispositivos da referida lei complementar. Por exemplo, o recente Código Civil de 2002 utiliza “SUBTÍTULO” como dispositivo para agrupar artigos, elemento este não previsto no rol de elementos para agrupar artigos da Lei Complementar nº 95. Outro exemplo é a Lei de Diretrizes Orçamentárias para o Orçamento de 2007 (LDO – Lei nº 11.439/2006), que lista alíneas subordinadas ao § 3º do art. 5º, quando o correto seria tratar esta enumeração utilizando incisos. Apesar de não comprometer a matéria, este equívoco da forma passou despercebido por todos os atores do processo orçamentário, desde a mensagem enviada pela Presidência da República até a publicação da norma jurídica. Este e outros tipos de erros formais podem ser facilmente evitados com a validação automática da estrutura de documentos textuais utilizando a tecnologia XML.

É importante ressaltar que os benefícios da adoção da tecnologia XML para modelar o texto de normas, julgados e matérias legislativas vão muito além da simples validação da estrutura. É possível realizar operações sofisticadas, tais como a apresentação de um mesmo conteúdo em vários formatos, a síntese de voz com a correta vocalização das remissões textuais e a comparação entre diferentes versões de um texto.

A legislação de referência da técnica legislativa federal é apresentada na Tabela 2.

Tabela 2. Legislação Federal sobre Técnica Legislativa.

Norma	Ementa
Lei Complementar nº 95/1998	Dispõe sobre a elaboração, a redação, a alteração e a consolidação das leis, conforme determina o parágrafo único do art. 59 da Constituição Federal, e estabelece normas para a consolidação dos atos normativos que menciona.
Lei Complementar nº 107/2001	Altera a Lei Complementar nº 95, de 26 de fevereiro de 1998.
Decreto nº 4.176/2002	Estabelece normas e diretrizes para a elaboração, a redação, a alteração, a consolidação e o encaminhamento ao Presidente da República de projetos de atos normativos de competência dos órgãos do Poder Executivo Federal, e dá outras providências.

As unidades da federação e os municípios podem adotar regras próprias para a técnica legislativa. Alguns estados, como Minas Gerais e Santa Catarina, seguem as regras da Lei Complementar nº 95/1998, enquanto que outros, como São Paulo, utilizam um sistema diferente para articulação dos dispositivos de uma norma. Essa diversidade de técnicas legislativas é considerada pelo Projeto LexML Brasil, na medida em que define dois tipos de validação. A “validação flexível” é o denominador comum a todos os ordenamentos jurídicos, existindo apenas um esquema, e a “validação rígida”, que se atém às regras da técnica legislativa de um ordenamento jurídico específico. Por exemplo: no nível federal o Projeto LexML define um esquema para validar a estrutura

da articulação de dispositivos, de acordo com as regras da Lei Complementar nº 95/1998; de forma semelhante, poderão existir também esquemas rígidos para validações de acordo com cada sistema de técnica legislativa estadual ou municipal, caso esses sistemas sejam diferentes do especificado na Lei Complementar nº 95/1998.

5. Articulação segundo a Lei Complementar nº 95.

Como forma de melhorar a articulação do texto de normas com muitos dispositivos, é possível utilizar dispositivos para agrupamentos de artigos (Parte, Livro, Título, Capítulo, Seção e Subseção). Essa sistematização é criada segundo critérios que visam à organização lógica e hierárquica da matéria normativa.

O artigo é a unidade básica de articulação e contém no seu *caput* a regra normativa propriamente dita. Os parágrafos servem para expressar os aspectos complementares bem como as exceções à regra. As discriminações e enumerações necessárias ao *caput* e aos parágrafos são feitas de forma hierárquica, utilizando incisos, alíneas e itens.

As duas seções seguintes mostram como os elementos para agrupar ou detalhar um artigo foram sistematizados no Projeto LexML, de acordo com a Lei Complementar nº 95/1998.

5.1. Agrupamento de Artigos e Artigo

A partir do elemento *Articulação*, pode-se ir diretamente ao elemento *Artigo* quando se trata de uma norma com poucos artigos ou pode-se utilizar um dos elementos para agrupar artigos. Apesar de a Lei Complementar nº 95/1995 não proibir o início da articulação pelo elemento *Subseção*, consideramos esse elemento apenas como um desdobramento do elemento *Seção*.

A Figura 1 ilustra as diversas possibilidades de combinação dos elementos para agrupar *Artigos*.

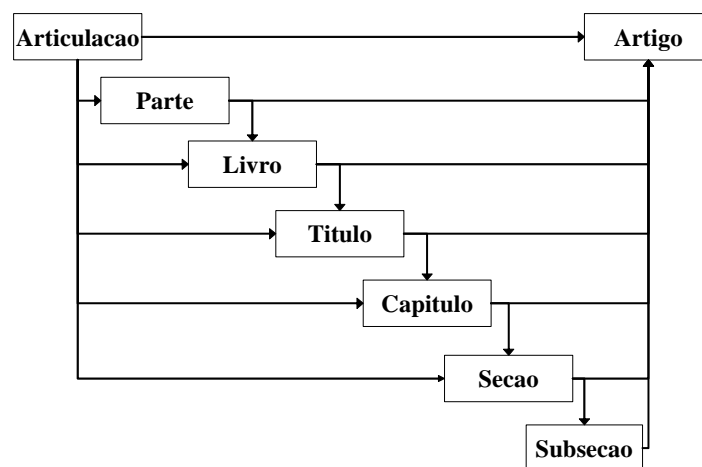


Figura 1. Agrupamento de Artigos e Artigo.

5.2. Detalhamento de Artigo

O detalhamento de artigo consiste do elemento *Caput*, podendo ser seguido opcionalmente por uma ou mais ocorrências do elemento *Parágrafo*. Pode parecer estranho para alguns a presença do elemento *Caput*, pois, normalmente, ele, por ser obrigatório, é omitido nas remissões levando a uma primeira impressão de que o inciso está relacionado diretamente com o Artigo. Após análise criteriosa da melhor estratégia para modelar o Artigo, definimos um elemento específico para tratar o texto do *caput* e seus desdobramentos. Entre os motivos, podemos citar a situação em que se revoga ou se veta apenas o *caput* e não o artigo como um todo. Além disso, do ponto de vista lógico, de acordo com a Lei Complementar nº 95, art. 11, III, os incisos, alíneas e itens servem para discriminações e enumerações do *caput* e dos parágrafos. A Figura 2 apresenta os desdobramentos do elemento Artigo, de acordo com a modelagem do Projeto LexML.

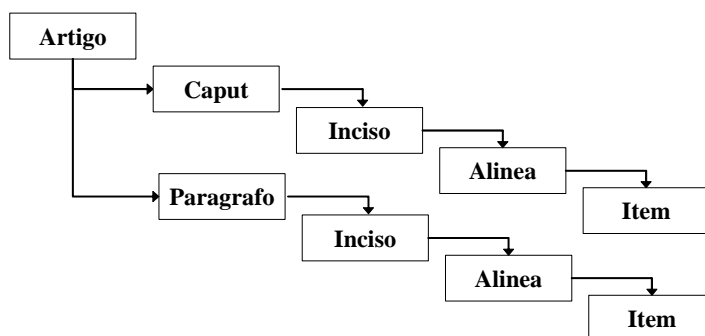


Figura 2. Detalhamento de Artigo.

5.2.1. Exemplo de Articulação no LexML

A Figura 3 apresenta a estruturação do art. 1º da Constituição Federal de 1988 utilizando a notação LexML.

```

<Artigo id="art1">
  <Rotulo>Art. 1º</Rotulo>
  <Caput id="art1_cpt"><p>A República Federativa do Brasil, formada pela união indissolúvel dos Estados e Municípios e do Distrito Federal, constitui-se em Estado democrático de direito e tem como fundamentos:</p>
    <Inciso id="art1_cpt_inc1"><Rotulo>I - </Rotulo><p>a soberania;</p></Inciso>
    <Inciso id="art1_cpt_inc2"><Rotulo>II - </Rotulo><p>a cidadania;</p></Inciso>
    <Inciso id="art1_cpt_inc3"><Rotulo>III - </Rotulo><p>a dignidade da pessoa humana;</p></Inciso>
    <Inciso id="art1_cpt_inc4"><Rotulo>IV - </Rotulo><p>os valores sociais do trabalho e da livre iniciativa;</p></Inciso>
    <Inciso ID="art1_cpt_inc5"><Rotulo>V - </Rotulo><p>o pluralismo político.</p></Inciso>
  </Caput>
  <Paragrafo ID="art1_par1"><Rotulo>Parágrafo único.</Rotulo><p>Todo o poder emana do povo, que o exerce por meio de representantes eleitos ou diretamente, nos termos desta Constituição.</p></Paragrafo>
</Artigo>
  
```

Figura 3. Art 1º da Constituição Federal de 1988 utilizando notação XML.

6. Especificação do Projeto LexML Brasil.

Os documentos de especificação do Projeto LexML Brasil estão organizados conforme Figura 4.

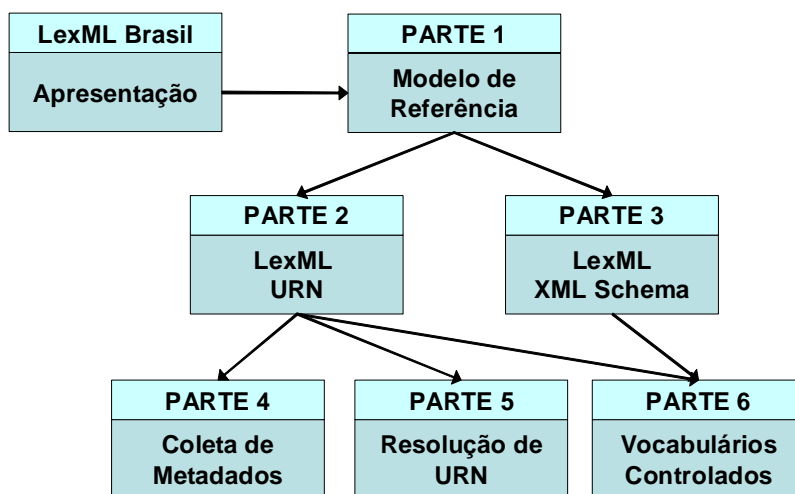


Figura 4. Especificação do Projeto LexML Brasil

- Parte 1 – Modelo de Referência – Detalha o modelo conceitual utilizado pelo Projeto LexML Brasil. Contém informações técnicas que são pré-requisitos para o entendimento dos demais documentos.
- Parte 2 – LexML URN – Descreve a gramática para formação de nomes uniformes de Normas, Julgados e Projetos de Normas.
- Parte 3 – LexML XML Schema – Explica os princípios e conceitos utilizados na criação do XML Schema para Normas, Julgados e Projetos de Normas.
- Parte 4 – Coleta de Metadados – Explica como o protocolo de coleta de metadados é aplicado ao projeto LexML Brasil.
- Parte 5 – Resolução de URN - Apresenta o serviço de resolução de URNs, detalhando sua funcionalidade.
- Parte 6 – Vocabulários Controlados – Apresenta os tipos de vocabulários controlados utilizados na formação de um nome uniforme e na especificação da seção de metadados do XML Schema.

7. Referências

7.1. Bibliográficas

LEAL, A. *Technica Constitucional Brasileira*. Rio de Janeiro: Typ. Do Jornal do Commercio, de Rodrigues & C. 1914. 81 p.

LIMA, J. A. O; SOUZA, M. A. M. *Legislação Brasileira na Internet*. VI CONIP (Congresso Nacional de Informática Pública). São Paulo. 2000. Disponível em:

<http://www.interlegis.gov.br/Members/monicaco/2006/legislacao-brasileira-na-internet>.
Acessado em: dezembro de 2008.

7.2. Sítios da Internet

<http://www.nir.it/> - Projeto Norme in Rete (Itália)

<http://www.akomantoso.org/> - Projeto Akoma Ntoso (Nações Unidas/África)

<http://www.ticontrole.gov.br/> - Comunidade TI Controle

<http://www.senado.gov.br/sf/legislacao/const/> - Constituição Federal de 1988

<http://www.governoeletronico.gov.br/consulta-publica> - Governo Eletrônico

<http://www.lexml.gov.br> – Portal LexML

<http://www.lexml.it> – Comunidade LexML na Itália

<http://www.lexml.de> – Comunidade LexML na Alemanha

8. Lista de Abreviaturas e Siglas

CEN - Centro Europeu de Normatização

ChemML - Chemical Markup Language

CNIPA - Centro Nazionale per Informatica nella Pubblica Amministrazione

CONIP - Congresso Nacional de Informática Pública

GML - Geography Markup Language

ITTIG - Istituto di Teoria e Tecniche dell'Informazione Giuridica

LDO – Lei de Diretrizes Orçamentárias

MathML - Mathematical Markup Language

NIR – Norme in Rete

PRODASEN - Processamento de Dados do Senado Federal

SIJED - Seminário de Gestão da Informação Jurídica em Espaços Digitais

URL - Universal Resource Locator

URN – Uniform Resource Name

XML – Extensible Markup Language

9. Créditos

9.1. Editores

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

Fernando Ciciliati (Senado Federal / Interlegis)

9.2. Grupo de Trabalho LexML (em Dezembro de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Cláudio Mori (Tribunal Superior Eleitoral)

Claudson dos Santos Melo (Tribunal Superior do Trabalho)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda Oliveira de Macedo (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)

Jonatas Rodrigues (Procuradoria Geral da República)

Luís Henrique Raja G. Mitchell (Tribunal de Contas da União)

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

Paulo de Tarso Penna da Costa (Superior Tribunal de Justiça)

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)

9.3. Outras Contribuições

Carlos Magno Cataldi Santoro (Senado Federal / Prodasen)

Cláudio Muniz Machado Cavalcanti (Ministério do Planejamento)

Davi Lima (Senado Federal / Interlegis)

Marcelo Silva Cunha (Senado Federal / Prodasen)

Márcio Iorio Aranha (Universidade de Brasília)

Neide De Sordi (Conselho da Justiça Federal)

Peter de Pádua Krauss (Prefeitura Municipal de São Carlos (SP))

Raimundo de Oliveira (Tribunal Regional do Trabalho da 21ª Região)

9.4. Criação da Logomarca LexML

Bruno Sartório (Senado Federal / Interlegis)

Diogo Pohl (Senado Federal / Interlegis)

Rafael Abadio (Senado Federal / Interlegis)



LexML Brasil
Parte 1 – Modelo de Referência

Versão 1.0 (RC1)
Brasília – Dezembro de 2008

LexML Brasil

Parte 1 – Modelo de Referência

Versão 1.0 (dezembro/2008)

Parte 1 – Modelo de Referência	2
1. Introdução.....	3
2. Modelo de Referência.....	3
2.1. FRBR _{oo}	3
3. Modelo de Referência no Projeto LexML Brasil	6
3.1. Criação e Publicação de uma Norma Jurídica	6
3.2. Norma e Tempo.....	7
4. Metadados e Identificação de Versões	8
5. Referências	9
5.1. Bibliográficas	9
5.2. Sites	9
6. Lista de Abreviaturas e Siglas	9
7. Créditos.....	10
7.1. Editores.....	10
7.2. Grupo de Trabalho LexML (em Dezembro de 2008).....	10

1. Introdução

A informação legislativa e jurídica pode ser percebida em diversos níveis de abstração. A norma jurídica pode ser percebida como um objeto físico ou digital, como, por exemplo, no documento original assinado pelo Presidente da República (autógrafo) ou no exemplar do Diário Oficial em formato PDF (*Portable Document Format*). Pode ser identificada ainda como um objeto lingüístico (texto veiculado por esses suportes), ou ainda como um objeto semântico (teor normativo veiculado nos textos).

Este documento apresenta o modelo de referência que define as entidades de interesse do Projeto LexML Brasil considerando os níveis de abstração apresentados.

2. Modelo de Referência

O Modelo de Referência do Projeto LexML Brasil é baseado na ontologia FRBR_{OO} (*Functional Requirements for Bibliographical Records*) (Bekiari et. al, 2008). O modelo FRBR_{OO} é uma extensão da ontologia CIDOC CRM (*Comité International pour la Documentation - Conceptual Reference Model*) (ICOM, 2004) com conceitos do modelo FRBR (IFLA, 1998), que originalmente utilizava a modelagem entidade relacionamento.

Uma das principais vantagens da adoção desse modelo é a percepção dos diversos níveis de abstração dos recursos de informação. Por exemplo, quando referenciamos uma determinada norma jurídica, de qual entidade estamos falando? Do conteúdo original promulgado após o processo legislativo? Ou da versão consolidada desta norma para a data corrente que foi editada por alguém? Ou ainda estamos tratando de uma versão traduzida desta norma para outra língua? Um sistema que tente modelar todos estes exemplos de entidades como uma única entidade não permitirá a correta representação dos diversos tipos de relacionamentos existentes entre estas entidades.

Após a apresentação de um resumo do modelo FRBR_{OO} (Seção 2.1), mostraremos como as entidades do universo legislativo e jurídico são representadas neste modelo de referência (Seção 3).

2.1. FRBR_{OO}

O modelo FRBR_{OO} é uma extensão do modelo CIDOC CRM. Desenvolvido desde 1994 de forma independente da iniciativa FRBR, o modelo CIDOC CRM foi aprovado como padrão internacional ISO 21127:2006. Em 2003, foi criado um grupo de trabalho composto por representantes da IFLA (*International Federation of Library Associations and Institutions*) e do ICOM (*International Council of Museums*) com o objetivo de harmonizar o modelo FRBR_{ER} com a ontologia CIDOC CRM. Apesar de o trabalho de integração ainda não estar concluído, a atual versão do modelo FRBR_{OO} representa um avanço considerável, pois corrige inconsistências do antigo modelo além de agregar a ele a dimensão temporal. Com esta nova abordagem é possível representar, no novo modelo, tanto o processo de criação de uma obra (documento) e sua respectiva

expressão manifestada em um original (Manifestação Única / *Manifestation Singleton*), quanto o processo de produção que, a partir do original, cria diversos exemplares (*Itens*) de uma obra.

O trabalho de harmonização consistiu na análise dos conceitos presentes em cada entidade e atributo do modelo FRBR_{ER} e o posicionamento das novas classes e propriedades na ontologia CIDOC CRM. As classes e propriedades no modelo CIDOC CRM são identificadas pelas letras “E” e “P”, respectivamente, seguidas de um número e de um nome. As classes e propriedades resultantes do processo de harmonização são identificadas pelas letras “F” e “R”, respectivamente, seguidas de um número e de um nome. No processo de harmonização, em alguns casos, uma entidade deu origem a uma só classe, como, por exemplo, no caso da entidade “Item”, que gerou a classe “F5 Item”, subclasse de “E84 Information Carrier”. Em outros casos, foi necessário criar várias classes como forma de modelar cada aspecto que antes estava aglutinado em uma única entidade ou que não era considerado no antigo modelo. As seções seguintes apresentam as principais classes de interesse para o Projeto LexML Brasil.

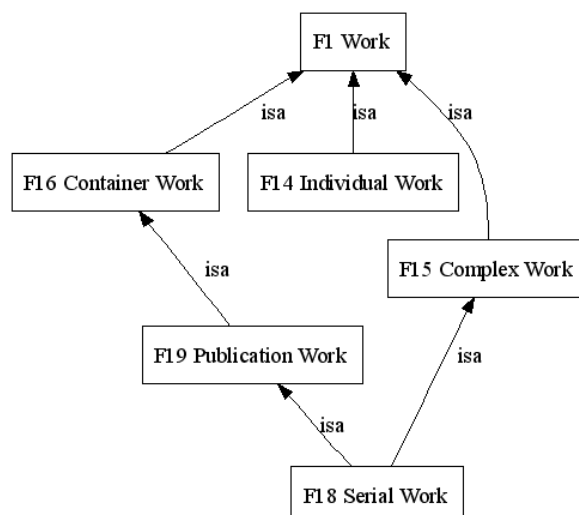
2.1.1. FRBR_{OO} – Classe ‘F1 Work’

A classe ‘F1 Work’ é uma das principais classes do modelo FRBR_{OO}, pois representa o conteúdo conceitual de uma obra. Normalmente o usuário está interessado em uma obra e não em um tipo de manifestação específica.

A classe ‘F1 Work’ é especializada em várias subclasses, detalhadas a seguir:

- ‘F14 Individual Work’ – representa os conceitos associados com um conjunto específico de símbolos realizados em uma ‘F22 Self Contained Expression’.
- ‘F15 Complex Work’ – representa um conjunto que possui outras instâncias de ‘F1 Work’ como membros. Serve para representar a evolução de uma obra no tempo e/ou para representar obras que são compostas por outras obras.
- ‘F16 Container Work’ – representa obras cuja essência é melhorar ou adicionar valor a expressões de outras obras, sem alterá-las.
- ‘F18 Serial Work’ – representa obras que foram planejadas para resultar em uma seqüência de manifestações com características comuns.
- ‘F19 Publication Work’ – representa obras que foram planejadas para resultar em um ‘Manifestation Product Type’ (tipo de produto) e que veicula expressões de outras obras.

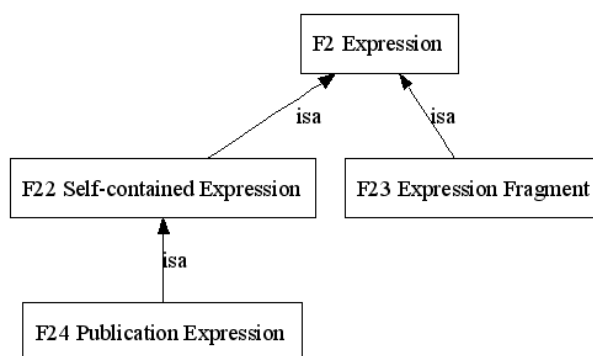
O diagrama hierárquico da classe ‘F1 Work’ é apresentado na Figura 1.


 Figura 1. Hierarquia da Classe ‘*F1 Work*’

2.1.2. FRBR_{OO} – Classe ‘*F2 Expression*’

Enquanto que a substância de um ‘*F1 Work*’ são conceitos, a substância de uma ‘*F2 Expression*’ são símbolos. Quando um ‘*F1 Work*’ é realizado de forma completa por um conjunto de símbolos, tem-se uma instância da classe ‘*F22 Self-Contained Expression*’. No outro caso, quando este conjunto está incompleto, ocorre uma instância da classe ‘*F23 Expression Fragment*’. Estas classes são subclasses de ‘*F2 Expression*’.

A classe ‘*F22 Self-Contained Expression*’ é especializada pela classe ‘*F24 Publication Expression*’ que representa o layout e conteúdo completo fornecido pelo editor para publicação. Pode conter elementos adicionais (além das expressões das obras veiculadas) tais como tabelas de conteúdo (sumário), cabeçalhos e notas do editor. O diagrama hierárquico da classe ‘*F2 Expression*’ é apresentado na Figura 2.


 Figura 2. Hierarquia da Classe ‘*F2 Expression*’

2.1.3. FRBR_{OO} – Classe ‘*F3 Manifestation Product Type*’

A classe ‘*F3 Manifestation Product Type*’ contém a definição dos objetos que serão produtos de um processo de produção (ou publicação). Trata-se apenas de um tipo (objeto abstrato).

2.1.4. FRBR_{OO} – Classe ‘F4 Manifestation Singleton’

A classe ‘F4 Manifestation Singleton’ possui como instância objetos físicos que possuem uma ‘F2 Expression’ e que foram produzidos como um objeto único, como no caso dos originais de uma obra. Trata-se de um artefato (objeto físico).

3. Modelo de Referência no Projeto LexML Brasil

Esta seção mostra como os processos de criação e publicação de uma norma jurídica são modelados utilizando o modelo de referência apresentado na seção anterior. Além disso, é apresentada também a modelagem da evolução da norma no tempo. Apesar de o exemplo ser ilustrado com uma norma jurídica, o raciocínio pode ser aplicado, com as devidas adaptações, aos outros tipos de documentos de interesse do Projeto LexML Brasil: julgados e projetos de normas.

3.1. Criação e Publicação de uma Norma Jurídica

A Tabela 1 relaciona classes e instâncias que podem ser identificadas na manifestação original (documento físico ou digital assinado pela autoridade competente) de uma norma jurídica. Trata-se de uma obra criada pelo processo legislativo cuja autoria deve ser atribuída a todos os atores envolvidos no processo legislativo, desde o momento da iniciativa legislativa, passando pelas etapas de discussão, deliberação e votação, considerando também a análise de veto que pode modificar o texto normativo promulgado.

Tabela 1 – Documento Original Assinado

Classe	Instância
<i>F4 Manifestation Singleton</i>	O documento original assinado (artefato).
<i>F22 Self Contained Expression</i> (especialização de <i>F2 Expression</i>)	Texto normativo resultante do processo legislativo (após a análise de veto).
F14 Individual Work	Conceitos associados com o documento original.

A norma jurídica, resultado do processo legislativo, é criada no momento da sua assinatura pela autoridade competente. No entanto, é só com a promulgação e publicação que a norma passa a ser executória. A edição específica de uma publicação oficial agrega várias entidades que coexistem em um mesmo suporte. Em uma primeira análise, pode-se perceber a publicação em si, resultado de um processo industrial que gera uma quantidade de exemplares de acordo com um plano de produção. Abstraindo a publicação, é possível perceber entidades relacionadas às normas jurídicas veiculadas.

A Tabela 2 relaciona classes e instâncias que podem ser identificadas em uma página de uma publicação oficial. Um exemplar de uma publicação oficial é uma instância da classe ‘F5 Item’ pois carrega uma ‘F24 Publication Expression’ e foi produzida por um processo industrial. A expressão textual desta manifestação é composta pela expressão textual dos documentos oficiais além de conteúdos originais criados pelo editor, tais como tabela de conteúdo, cabeçalhos etc. No caso de normas jurídicas, a expressão textual é originada a partir do original de documentos oficiais.

Tabela 2 – Periódico Oficial - Exemplar

Classe	Instância
<i>F5 Item</i>	Exemplar da Publicação Oficial
<i>F3 Manifestation Product Type</i>	Tipo de publicação do produto “Diário Oficial da União – Seção 1”, publicado em 6 Agosto de 2007 pela Impressa Nacional.
<i>F18 Serial Work</i>	Periódico intitulado “Diário Oficial”
<i>F19 Publication Work</i> <i>F17 Aggregation Work</i>	Conceitos associados à edição de 6 de Agosto de 2007 do Diário Oficial da União.
<i>F41 Publication Expression</i> (especialização de <i>F22 Self Contained Expression</i>)	Layout completo e conteúdo definido pelo editor (incluindo tabela de conteúdo, cabeçalhos, notas e expressão de outras obras).
<i>F22 Self Contained Expression</i>	Texto da norma publicada.
<i>F14 Individual Work</i>	Conceitos associados à norma publicada.

A Figura 3 apresenta, em um único diagrama, os processos de concepção de uma norma e criação da sua expressão realizados pelo autor, bem como os processos que levam à produção de exemplares de uma publicação oficial.

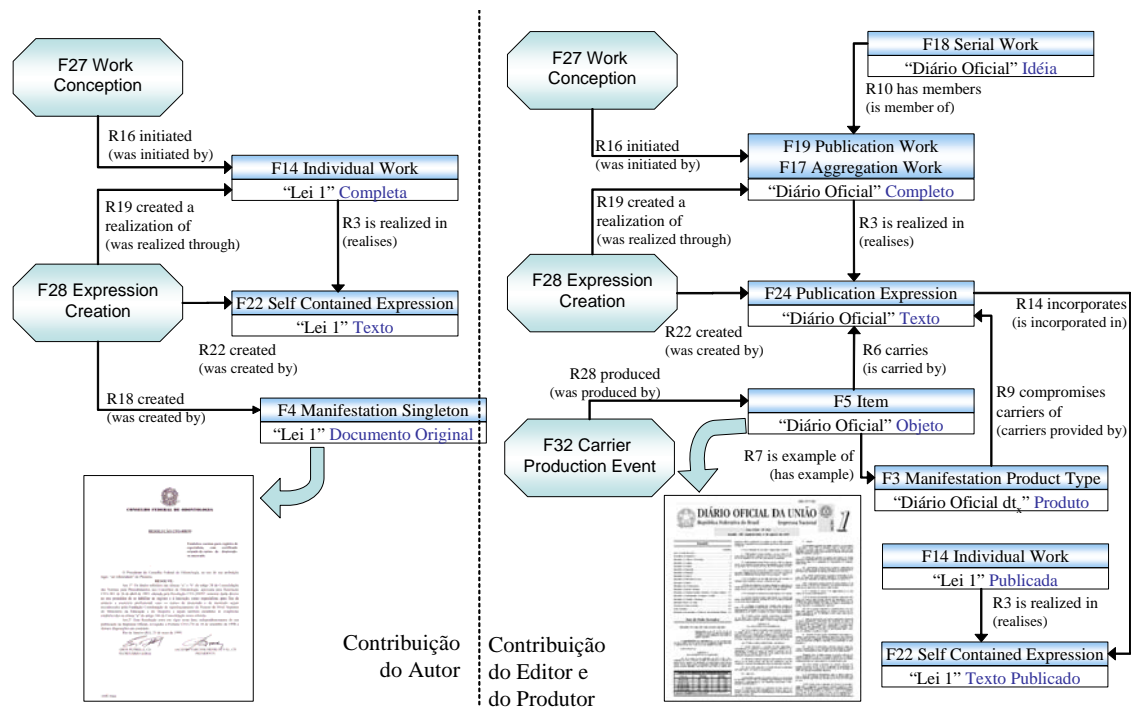


Figura 3. Processo de Criação, Edição e Publicação de Documentos Oficiais.

Normalmente o texto publicado reflete o texto do original. No entanto, por problemas no processo de publicação, podem ocorrer inconsistências entre o texto do original e o texto publicado. Nesse caso é necessário realizar a publicação de uma comunicação oficial com as devidas retificações.

3.2. Norma e Tempo

Durante o ciclo de vida de uma norma jurídica, vários eventos podem afetar o seu conteúdo, tanto em relação a sua forma (expressão textual) quanto em relação a sua matéria (teor normativo). Por exemplo, a expressão textual de uma norma pode ser afetada por ações de inclusão, alteração ou revogação de dispositivos. Esta seção mostra

como o novo modelo FRBR₀₀ permite representar, de forma precisa, a evolução da norma no tempo.

A seção anterior mostrou que cada norma é modelada como uma instância da classe ‘*F14 Individual Work*’ com a respectiva instância da classe ‘*F22 Self Contained Expression*’. No momento em que ocorre a publicação de uma norma deve-se criar também uma instância da classe ‘*F15 Complex Work*’ com o objetivo de referenciar as duas instâncias da classe ‘*F14 Individual Work*’: uma referente ao documento original assinado e outra contida no documento oficial publicado.

No momento da entrada em vigor da norma que altera uma outra já existente, cria-se uma obra derivada que é representada por uma nova instância da classe ‘*F14 Individual Work*’ com a respectiva instância da classe ‘*F22 Self Contained Expression*’. Na maioria das vezes, os eventos de publicação e entrada em vigor ocorrem em uma mesma data. Essa dinâmica é ilustrada pela Figura 4 que mostra o que ocorre com a “Lei 1” quando é alterada pela “Lei 8”. Neste exemplo, o período de *vacatio legis* da norma alteradora está representado pelo intervalo t_2 - t_3 .

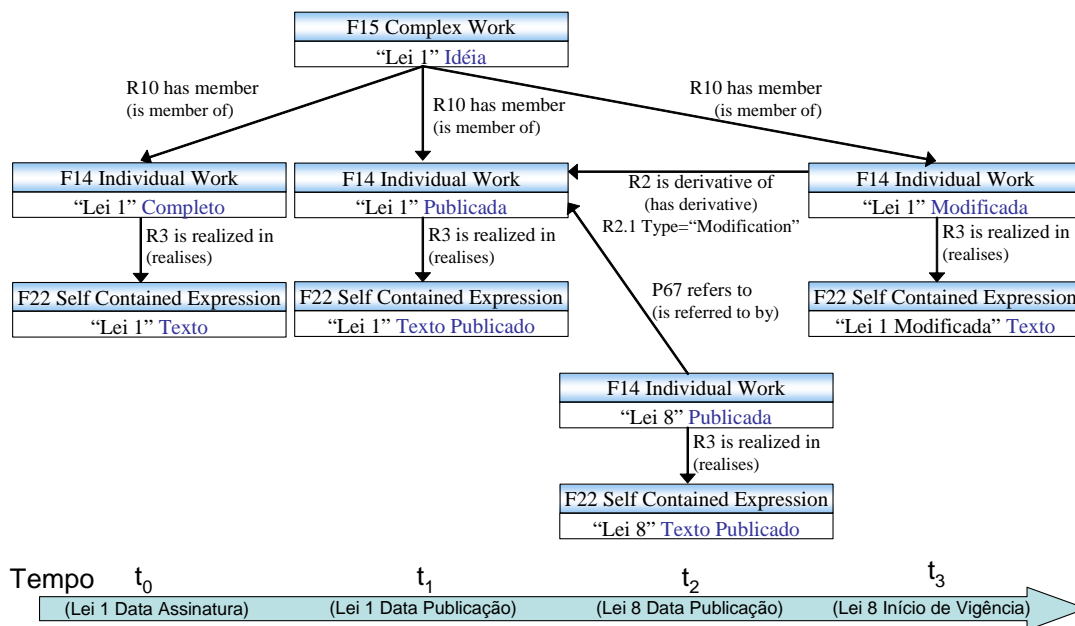


Figura 4. Evolução da Norma no Tempo.

4. Metadados e Identificação de Versões

Como apresentado, o modelo FRBR₀₀ permite identificar as entidades envolvidas tanto no processo de criação (originais) como no processo de publicação, ou ainda as entidades abstratas resultantes das modificações das normas. A definição de um identificador que permita endereçar as entidades citadas deve considerar, além de outros campos de metadados, três datas:

- o Data Representativa – data da assinatura (normas), iniciativa (proposição legislativa) ou julgamento (julgados) para a identificação de uma instância de um ‘*F15 Complex Work*’.

- Data da Versão – data de início de vigência de uma nova norma ou de sua alteração por uma outra norma (ver Seção 3.2 - Norma e Tempo). No caso de proposições legislativas e julgados, este atributo está relacionado ao evento de publicação de uma versão.
- Data da Visão – data de um evento que gere uma nova versão ou uma variante de uma versão. Por exemplo, os eventos de retificação, derrubada de veto parcial ou declaração de inconstitucionalidade geram uma nova visão para versões já existentes de normas jurídicas.

As datas de versão e de visão são utilizadas na identificação de uma versão específica de um documento. A Parte 2 (URN) da especificação LexML apresenta exemplos utilizando as datas de versão e visão.

5. Referências

5.1. Bibliográficas

BEKIARI, C.; DOERR, M.; LE BOEUF P.; International Working Group on FRBR and CIDOC CRM Harmonisation. *FRBR Object-Oriented Definition and Mapping to FRBR_{ER}*. (version 0.9 draft) 2008. Disponível em : <http://cidoc.ics.forth.gr/docs/frbr_oo/frbr_docs/FRBR_oo_V0.9.pdf>. Acesso em: 8 julho 2008.

ICOM/CIDOC Documentation Standards Group; CIDOC CRM Special Interest Group. *Definition of the CIDOC Conceptual Reference Model: version 4.0*, 2004. Heraklion, Greece, 2004. Disponível em: <http://cidoc.ics.forth.gr/docs/cidoc_crm_version_4.0.pdf> Acesso em: 17 julho 2008.

IFLA. Study Group on the Functional Requirements for Bibliographic Records. *Functional Requirements for Bibliographic Records*. München: K . G. Saur, 1998. 144 p.

5.2. Sites

<http://cidoc.ics.forth.gr/> - CIDOC CRM

<http://www.ifla.org/VII/s13/frbr/> - IFLA / FRBR

6. Lista de Abreviaturas e Siglas

CIDOC - Comité International pour la Documentation

CRM – Conceptual Reference Model

FRBR - Functional Requirements for Bibliographical Records

FRBR_{OO} - Functional Requirements for Bibliographical Records (Object Oriented)

FRBR_{ER} - Functional Requirements for Bibliographical Records (Entity Relationship)

ICOM - International Council of Museums

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 Dezembro de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Cláudio Mori (Tribunal Superior Eleitoral)

Claudson dos Santos Melo (Tribunal Superior do Trabalho)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda Oliveira de Macedo (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)

Jonatas Rodrigues (Procuradoria Geral da República)

Luís Henrique Raja G. Mitchell (Tribunal de Contas da União)

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

Paulo de Tarso Penna da Costa (Superior Tribunal de Justiça)

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)



LexML Brasil
Parte 2 – LexML URN

Versão 1.0 (RC1)
Brasília – Dezembro de 2008

LexML Brasil

Parte 2 – LexML URN

Versão 1.0 (Dezembro/2008)

Parte 2 – LexML URN	2
1. Introdução.....	4
1.1. Objetivo do Nome Uniforme do Recurso.....	5
1.2. Associação do Nome Uniforme ao Documento	5
1.3. Utilização do Nome Uniforme nas Referências	5
1.4. Tipos de URN.....	6
1.5. Princípios-base do Nome Uniforme (URN)	6
1.6. Características Gerais do Sistema de Identificação e Referência a Documentos	7
2. Sintaxe Utilizada neste Documento.....	7
3. Espaço de nomes (namespace)	8
3.1. Internacionalização do espaço de nomes "lex"	8
4. Sintaxe Geral do Nome-Específico Brasileiro	9
4.1. Caracteres Permitidos e Proibidos.....	9
4.2. Caracteres Reservados.....	10
4.3. Casos Especiais	10
4.4. Formato da Data	12
5. Construção do Nome-Específico (NSS-lex-br)	12
5.1. Estrutura do Nome-específico <NSS-lex-br>	12
5.2. Nomes Alternativos (alias)	13
6. Elemento <local>.....	14
7. Elemento <documento>.....	15
8. Elemento <autoridade>	16
8.1. Indicação da Autoridade.....	16
8.2. Pluralidade dos Sujeitos	16
8.3. Modalidade de Indicação do Sujeito	16
9. Elemento <tipo-documento>	18
9.1. Critérios para a Indicação do Tipo do Documento.....	18
9.2. Especificação Adicional do Tipo de Documento	18
9.3. Especificação da Publicação Oficial.....	19
10. Elemento <descriptor>	19

10.1.	Indicação do Descritor.....	20
10.2.	Elemento <identificadores>.....	21
10.3.	Identificador de Componentes de um Documento	23
10.4.	Descritor da Publicação Oficial.....	25
11.	Elemento <fragmento>	25
11.1.	Fragmento da Publicação Oficial	26
12.	Elemento <versao>.....	27
12.1.	As Versões e Visões de um Texto	27
12.2.	Casos Especiais	28
12.3.	Identificação da Versão	28
12.4.	Exemplo - CDC	29
12.5.	Valores <i>Default</i> para os Elementos da Versão	30
13.	Elemento <forma>.....	30
14.	Referências	31
14.1.	Bibliográficas	31
14.2.	Sites	31
15.	Lista de Abreviaturas e Siglas	31
16.	Créditos.....	32
16.1.	Editores (LexML Brasil)	32
16.2.	Grupo de Trabalho LexML (em Dezembro de 2008).....	32
16.3.	Outras Contribuições	33
Anexo A – Gramática da URN Canônica.....		34
Anexo B – Gramática da URN de Referência.....		38

1. Introdução

Os nomes uniformes de recursos - *Uniform Resource Names* (URNs) – foram concebidos para fornecer identificadores unívocos e persistentes para recursos publicados na Internet, independente da sua localização na rede.

Com este documento define-se a identificação de normas, julgados e projetos de normas por meio de um nome uniforme para o ordenamento jurídico brasileiro. Tal nome será utilizado como instrumento para representar as referências (e, mais genericamente, quaisquer tipos de relações) entre os documentos, com a finalidade de tornar disponível um ambiente hipertextual global sobre os recursos de informação, operacional em um ambiente distribuído.

Em especial, este documento descreve as regras para a atribuição de nomes uniformes e delinea os critérios principais para que os serviços de resolução possam associar a um nome uniforme a localização do documento correspondente na rede.

Nos documentos jurídicos são muito freqüentes, e extremamente importantes, as referências a outros documentos; portanto, a possibilidade de poder consultar imediatamente o documento referenciado constitui um auxílio de grande relevância, e, às vezes, indispensável para a plena compreensão do texto.

Os links hipertextuais da web certamente respondem a essa necessidade, mas mostram-se inadequados a um emprego em larga escala para referenciar documentos normativos e legislativos, pois a referência ao recurso relacionado baseia-se, na maioria das vezes, em sua localização física, expressa de modo uniforme por meio de uma URL (*Uniform Resource Locator*), que apresenta problemas já amplamente conhecidos, quais sejam:

- a dificuldade de se conhecer a localização do recurso citado;
- com o passar do tempo, a perda de validade dos links (URL) nas referências;
- a impossibilidade de referenciar recursos ainda não publicados;

Esses problemas tornam a rede de links entre documentos extremamente limitada com relação às suas potencialidades e cada vez menos confiável com o passar do tempo.

Para remediar tais inconvenientes, utiliza-se um sistema de referências baseado na atribuição a cada documento de um nome uniforme (URN: *Uniform Resource Name*), ou identificador unívoco, e um método de resolução (RDS: *Resolver Discovery Service*) capaz de encontrar o objeto correspondente. Essa abordagem está de acordo com as recomendações do IETF (*Internet Engineering Task Force*), definidas por um grupo de trabalho específico (*URN Working Group*), e descritas em vários documentos - dos padrões oficiais (*RFC: Request For Comments*) aos *drafts* - aos quais é, e deverá ser, garantido o alinhamento também no futuro.

1.1. Objetivo do Nome Uniforme do Recurso

A atribuição de um nome uniforme tem como objetivo associar a cada documento jurídico, legislativo ou parte dele um identificador unívoco, em um formato padronizado que dependa apenas das características do próprio documento (metadados) e que, por isso, seja independente da disponibilidade em rede, da sua localização física e das modalidades de acesso.

Tal identificador é utilizado como instrumento para representar as referências - e mais genericamente qualquer tipo de relação - entre os documentos. O seu emprego facilita, em um ambiente de rede com recursos (documentos) distribuídos entre diversas fontes eletrônicas de informação, a construção de um hipertexto global entre os documentos jurídicos, legislativos e de bases de conhecimento contendo as relações entre os mesmos.

1.2. Associação do Nome Uniforme ao Documento

A associação do nome uniforme ao documento dá-se por meio de uma meta-informação, que pode ser:

- inserida no interior do próprio documento: é a solução natural a ser aplicada com a adoção do formato XML (*eXtensible Markup Language*), mas é factível também em outros formatos tais como HTML (*HyperText Markup Language*), PDF (*Portable Document Format*), DOC (Documento do Microsoft Word) ou ODF (*Open Document Format*);
- externa ao documento, mas estreitamente relacionada a este, de diversos modos: mesmo nome do arquivo e extensão predefinida, link hipertextual particular, atributo específico no banco de dados etc.

Tais modalidades são fortemente recomendadas no que se refere à implementação e à atualização automática dos catálogos (distribuídos ou centralizados), necessários ao processo de resolução, que contem, para todo documento, o nome uniforme e sua localização física.

Em perspectiva, deve-se tender à atualização automática, evitando-se a atualização manual dos catálogos, por causa do esforço necessário à sua manutenção e da intrínseca perda de validade com o tempo.

Em tais catálogos podem estar presentes também outras informações, como: metadados descritivos (ementa, título, assunto, dados da publicação oficial etc), relacionamentos com outros documentos e período de vigência.

1.3. Utilização do Nome Uniforme nas Referências

O nome uniforme será utilizado em larga escala nas referências de hipertexto realizadas a partir do texto integral de documentos ou a partir de sistemas de informações.

Tal vínculo pode ser realizado das seguintes maneiras:

- inserindo-se manualmente, no texto do documento que realiza a referência, o link com o nome uniforme: operação onerosa, sobretudo para os documentos já publicados em rede;
- construindo-se automaticamente (de maneira permanente ou provisória) o link com o nome uniforme, por meio de analisadores (*parser*) das referências no interior do texto: operação mais econômica mas sujeita a um certo percentual de imprecisão e incompletude. As referências não são de fato sempre formalmente corretas e completas, e devem por isso ser complementadas pelo operador ou interpretadas pelo usuário.
- construindo-se automaticamente (de maneira permanente ou provisória) o link com o nome uniforme, a partir de metadados (dados descritivos) existentes nos sistemas de informações dos diversos órgãos.

1.4. Tipos de URN

Considerando os cenários apresentados de uso da URN, podemos definir dois tipos de URN:

- URN Canônica – É a URN normalizada segundo o vocabulário e regras mais restritas, existindo apenas uma para cada entidade identificável (Documento, Documento Complexo e Fragmento). Ela é utilizada obrigatoriamente pelo provedor de informações (ver LexML Parte 4 – Coleta de Metadados), relacionada a um documento específico, devendo ser codificada sempre na forma completa.
- URN de Referência – É a URN utilizada pelos usuários ou sistemas de informação nas referências a documentos, podendo utilizar a forma compacta, omitir elementos que assumirão valores *default* ou ainda referenciar nomes alternativos.

A URN de Referência facilita a referência a documentos pois permite múltiplas formas de endereçamento. Por exemplo, para referenciar uma lei não é necessário informar o mês e o dia da data de assinatura (data representativa), bastando informar o ano; ao referenciar a norma “Lei Maria da Penha” não é necessário informar números e datas (ex: urn:lex:br:federal:lei:lei.maria.penha).

1.5. Princípios-base do Nome Uniforme (URN)

O nome uniforme deve ser unívoco, isto é, deve identificar uma e apenas uma entidade, e é construído de modo a ser, tanto quanto possível, alinhado com os princípios enumerados a seguir.

1.5.1. Princípios Gerais da URN

- auto-explicativo para os usuários;
- dedutível por meio de regras simples e claras;

- alinhado constantemente aos demais padrões do projeto.

1.5.2. Princípios da URN Canônica

- unívoco para o espaço das URNs canônicas;
- registrável com o mínimo de informação (o suficiente para ser unívoca)
- representativo dos aspectos formais do documento;
- em conformidade, à data de sua emissão, com a estrutura/organização da autoridade emitente e com a tipologia do documento;

1.5.3. Princípios da URN de Referência

- representativo dos aspectos formais e/ou substanciais do documento;
- mapeável em URNs canônicas cabíveis;
- compatível com a prática em uso para criar referências;
- reduzido ao essencial, para simplificar os links com outros documentos;
- gerado automaticamente por analisadores das referências no texto;

1.6. Características Gerais do Sistema de Identificação e Referência a Documentos

O sistema de referenciamento baseado em nomes uniformes deve prever:

- um esquema de atribuição dos nomes capaz de representar de maneira unívoca qualquer documento normativo, emitido por qualquer autoridade a qualquer tempo (passado, presente e futuro);
- um mecanismo para a resolução de um nome uniforme para a localização dos recursos correspondentes na rede.

O presente documento trata em detalhes apenas o primeiro dos requisitos indicados, isto é, a definição do esquema de atribuição dos nomes uniformes. A “Parte 5 – Serviço de Resolução” trata das diretrizes utilizadas pelo serviço de resolução.

2. Sintaxe Utilizada neste Documento

No presente documento foi utilizada a sintaxe típica de muitos RFCs da Internet, baseada na meta-linguagem BNF (*Backus-Naur Form*), e, em particular, naquela do documento RFC 2141 "URN Syntax".

As suas principais características são:

- os elementos são contidos entre os sinais de menor e maior (" $<$ " e " $>$ ");
- um elemento é separado da sua especificação com a *string* ":: $=$ ";
- os elementos em alternativa são separados por uma barra vertical (" $|$ ");
- as *strings* de caracteres são contidas entre aspas (" $''$ ");

- as partes opcionais são contidas entre colchetes ("[" e "]");
- um grupo de elementos é contido entre parênteses ("(" e ")");
- uma expressão que precede um elemento, ou um grupo de elementos, indica um fator de repetição, com os seguintes formatos:
 - `<n>` : `<n>` vezes;
 - `<n>*` : `<n>` ou mais vezes;
 - `*` : zero ou mais vezes.

Em cada diagrama de sintaxe, será indicado se as regras são específicas da URN Canônica ou de Referência ou se elas se aplicam a ambos os casos, indicado, nesse caso, com a palavra “Comum”.

3. Espaço de nomes (*namespace*)

O documento RFC 2141 "URN Syntax" (RFC 2141) define a sintaxe geral de um nome uniforme, que deve ter a forma:

Comum	<code><URN> ::= "urn:" <NID> ":" <NSS></code>
-------	---

onde `<NID>` é o identificador do Espaço de Nomes (*namespace*) e `<NSS>` é um nome-específico qualquer pertencente ao Espaço precedente.

O Espaço de Nomes identifica o contexto no qual estes são válidos e significativos. Para os documentos jurídicos define-se o Espaço de Nomes Uniformes com o identificador "lex":

Comum	<code><NID> ::= "lex"</code>
-------	------------------------------------

e assim as URNs relativas têm a forma:

Comum	<code><URN> ::= "urn:lex:" <NSS-lex></code>
-------	---

3.1. Internacionalização do espaço de nomes "lex"

A fim de permitir a adoção do Espaço de nomes "lex" em nível internacional, respeitando-se a autonomia e as especificidades de cada país, foi definido o elemento-base `<local>`, colocado dessa forma na URN:

Comum	<code><NSS-lex> ::= <local> ":" <NSS-lex-local></code>
-------	--

Esse elemento, que deve ser composto pelo menos pelo código IETF do país ou da super-região ("eu", no caso da União Européia), podendo ainda, a critério de cada país, ser subdividido de forma hierárquica de modo a contemplar o mapeamento de regiões, estados, municípios ou de outras unidades político-administrativas inerentes à sua organização. Em resumo:

Comum	<code><local> ::= <codigo-pais> [<detalhamento-local>]</code>
-------	---

	<code><codigo-pais> ::= 2<minuscula></code>
--	---

O código IETF para o Brasil é a *string* "br", e assim as URNs de documentos brasileiros têm a forma:

Comum	<code><URN> ::= "urn:lex:br:" <NSS-lex-local></code>
-------	--

Para o Brasil, definiu-se ainda que o elemento `<local>` possui uma estrutura hierárquica que reflete a jurisdição da autoridade emitente, desdobrando-se, quando necessário, em unidades da federação e municípios ou em regiões de órgãos do Poder Judiciário. A forma de composição desse elemento será vista adiante.

Apesar de a forma do elemento `<NSS-lex-local>` ser de livre definição pelo país usuário do espaço de nomes, optamos por utilizar para o `<NSS-lex-br>` um esquema similar ao modelo do projeto italiano NIR (*Norme in Rete*), aproveitando-nos da experiência e de conceitos já desenvolvidos, testados e estabelecidos.

4. Sintaxe Geral do Nome-Específico Brasileiro

4.1. Caracteres Permitidos e Proibidos

São definidos de acordo com a RFC 2141 "*URN Syntax*", em especial figuram como não admitidos, além de uma série de caracteres especiais, todos aqueles nacionais (código ASCII superior ao 127), que devem ser convertidos na sequência de escape ("% <hex> <hex>") correspondente.

Por razões de várias naturezas, ilustradas a seguir, no nome-específico (`NSS-lex`) do namespace "lex" é permitido apenas este subconjunto de caracteres:

Comum	<pre> <NSS-lex-local> ::= 1*<aceitos-lex> <aceitos-lex> ::= <normal> <reservado> <detalhaID> <outro> <normal> ::= <minuscula> <numeral> "." <normalID> ::= <normal> <detalhaID> <minuscula> ::= "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n" "o" "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z" <numeral> ::= "0" "1" "2" "3" "4" "5" "6" "7" "8" "9" <reservado> ::= ":" "@" ";" "," "[" "]" "!" "~" <detalhaID> ::= "_" "-" <outro> ::= "=" "\$" </pre>
-------	---

Os outros caracteres são eliminados ou convertidos, de acordo com as regras que serão apresentadas nas próximas seções.

A sintaxe completa dos nomes uniformes do Espaço "lex-br" é descrita nos Anexos A e B, onde são apresentadas as gramáticas da URN Canônica e da URN de Referência respectivamente.

4.2. Caracteres Reservados

São os caracteres que têm um significado especial na formação das URI (*Uniform Resource Identifier*) ou na formação de nomes específicos no Espaço "lex", e, portanto, devem ser empregados sempre e apenas para o escopo definido. Pertencem à primeira categoria os caracteres "%", "/", "?", "#" que conservam o significado típico das URLs, com exceção do caractere "#", que, no âmbito do LexML, não deve ser utilizado para indicar um "fragmento" do documento no nome uniforme, pois, para isso, existe um caractere específico ("\$").

Por sua vez, são reservados no nome-específico do Espaço "lex" os caracteres apresentados na Tabela 1.

Tabela 1. Caracteres Reservados.

Caractere(s)	Aplicação
" : "	separador dos elementos principais do nome.
" ; "	marcador da introdução de uma especificidade ou de um nível hierarquicamente inferior do elemento.
" @ "	separador de versão.
" ~ "	separador para indicar a forma da expressão do conteúdo (ex: texto, imagem etc.) e língua.
" ! "	separador de fragmento.
" [" , "] "	delimitadores utilizados para especificar intervalo de valores. Os valores de limite inferior e superior são separados pelo caractere " , " .
" , "	É utilizado nas seguintes situações: <ul style="list-style-type: none">▪ separador de múltiplas ocorrências de valores para um mesmo elemento principal, ao mesmo nível de especificidade (ex.: numeração e autoridade múltipla);▪ separador dos valores limites de um intervalo; e▪ separador do identificador de componente do título do componente (ex: anexo.1, tratado.cooperacao.amazonica).

4.3. Casos Especiais

4.3.1. Caracteres Maiúsculos e Minúsculos

O nome-específico <NSS> da URN, como as URLs, é sensível à grafia maiúscula ou minúscula.

A resolução dos nomes pertencentes ao Espaço "lex" deve ser independente da grafia, mas, para desfrutar plenamente das possibilidades de *caching* (memorização temporária para acesso rápido) de recursos, o nome-específico é construído sempre em caracteres minúsculos.

(ex.: "Ministério" é representado como "ministerio")

4.3.2. Caracteres Nacionais e Sinais Diacríticos

Mesmo sendo possível representar os caracteres nacionais e os sinais diacríticos por meio de uma seqüência de escape ("%<hex> <hex>"), por razões de simplicidade, de gestão e de comunicação estes caracteres especiais são excluídos do Espaço "lex" e, portanto, devem ser convertidos aos caracteres base.

(ex.: "saúde" é representada como "saude")

4.3.3. Substituição de Espaços, Conectivos e Pontuação

Na construção do nome-específico são eliminados da sentença correspondente todos os conectivos lingüísticos (artigos, preposições simples e com artigos, conjunções etc.), a pontuação e os caracteres especiais como apóstrofos, hífens, aspas etc.

As palavras significativas restantes são ligadas entre si por um ponto (".") em substituição ao espaço.

(ex.: "Ministério do Planejamento, Orçamento e Gestão" é representado como "ministerio.planejamento.orcamento.gestao")

4.3.4. Expansão das Abreviações

No nome-específico os nominativos são representados por extenso, e, portanto, devem-se expandir as abreviações de quaisquer tipos presentes na indicação das instituições (ex.: Min.), das estruturas (ex.: Gov.) ou das normas (ex.: circ.).

(ex.: "Min." é representado como "ministerio")

4.3.5. Expansão das Siglas

Para evitar possíveis homônimos provocados pelo uso de siglas na indicação das instituições, das estruturas ou das normas, no nome-específico essas devem ser explicitadas com as correspondentes formas extensas.

(ex.: "SRF" é representada por "secretaria.receita.federal")

4.3.6. Elementos e Números Ordinais

Na construção do nome-específico os eventuais números ordinais presentes nos elementos são sempre representados por algarismos arábicos, independente da grafia

que assumam: número romano, alíneas, adjetivo, letras, número arábico com ápice etc. (IV, alínea b, terceiro, 1º, 2ª etc.).

Identificadores de fragmentos de documento (ex.: artigos, incisos, capítulos etc.) possuem sintaxe compacta própria, conforme será apresentado nas seções seguintes.

(ex.: "Anexo III" é representado como "anexo.3")

"Anexo D" é representado como "anexo.4"

"Art. 11" é representado como "art11"

"Art. 12-B" é representado como "art12-2"

"Inciso III do Art. 12-B" é representado como "art12-2_inc3")

4.4. Formato da Data

Para tornar o nome uniforme mais facilmente ordenável cronologicamente e mais legível aos operadores, as datas são representadas sempre em forma numérica e no formato:

Comum	<pre><data> ::= <ano> "-" <mes> "-" <dia> <ano> ::= 4<numeral> <mes> ::= 2<numeral> <dia> ::= 2<numeral></pre>
-------	--

(ex.: "2 de setembro de 99" é representado por "1999-09-02")

4.4.1. Tratamento de Datas Múltiplas e Período

Alguns atos, mesmo sendo únicos, são identificados por mais de uma data; nesse caso, no campo <datas> devem ser relacionadas todas as datas presentes, separadas por “,”.

Caso seja necessário especificar um intervalo fechado entre duas datas, utiliza-se os delimitadores “[” e “]” com as datas separadas por “,”.

Dessa forma, o elemento <datas> possui a seguinte definição:

Comum	<pre><datas> ::= (<data> *["," <data>]) <periodo> <periodo> ::= "[" <data> "," <data> "]"</pre>
-------	--

(ex.: "1999-03-02,1999-04-15,2000-08-01" e

"[1999-03-02,2000-08-01]")

5. Construção do Nome-Específico (NSS-lex-br)

5.1. Estrutura do Nome-específico <NSS-lex-br>

Considerando o Modelo de Referência do Projeto LexML Brasil (Parte 1), são três os tipos de entidades que podem ser endereçadas por uma URN:

- Documento Complexo (*F15 Work Complex*) – representa o mais alto nível de abstração do modelo e referencia o conteúdo do documento (matéria) no tempo, não se preocupando em especificar detalhes de versão ou forma de expressão. Normalmente é esta a entidade referenciada pelo usuário quando da pesquisa de uma norma.
- Documento Individual (*F14 Individual Work*) – representa o conteúdo abstrato (matéria) da versão específica de um documento. Quando um Documento Individual é base para derivação de um outro Documento Individual (como no caso da publicação), ambas as instâncias passam a ser membros de um Documento Complexo. Cada instância de Documento Individual possui uma expressão auto-contida relacionada (*F22 Self Contained Expression*) que pode conter, de forma recursiva, outras expressões, denominadas fragmentos.
- Fragmento de Documento Individual (*F22 Self Contained Expression*) – representa a partição de um documento individual. No caso de normas jurídicas ou proposições legislativas, pode ser um artigo, um dispositivo que agrupa artigos (ex. título, capítulo, seção etc.) ou ainda um dispositivo de detalhamento de um artigo (ex. alínea, inciso etc.). No caso de acórdãos, pode ser um dos seus elementos constituintes (ex. relatório, voto etc.)

Em nosso ordenamento jurídico, o conteúdo normativo expressa-se como um texto na língua portuguesa. No entanto, em alguns casos, pode haver variação tanto na língua utilizada quanto na forma de expressão (ex: mapas, figuras etc).

Portanto, a estrutura do nome-específico de cada entidade é representada por:

Comum	<pre> <NSS-lex-local> ::= <documentoComplexo> (<documentoIndividual> ["!" <fragmento>]) <documentoIndividual> ::= <documentoComplexo> ["@" <versao>] ["~" <forma>] <documentoComplexo> ::= <documento> </pre>
-------	---

5.2. Nomes Alternativos (alias)

A um mesmo documento podem ser atribuídos diversos nomes uniformes, a fim de favorecer a ligação – por referenciamento e por pesquisa - com outros documentos. Essa facilidade pode ser utilizada para aqueles documentos que, devido à sua individualidade, acabam sendo referenciados na prática sob ângulos diversos, como, por exemplo, a forma de promulgação e o conteúdo específico (ex.: Regimento Interno veiculado por uma Resolução).

É muito comum também a referência de normas utilizando apelidos, tais como ‘Código Civil’, ‘Código de Defesa do Consumidor’, ‘Lei do Audiovisual’ etc. A codificação de nomes alternativos será realizada utilizando valores especiais para os elementos <tipo-documento> e <descriptor>, que serão apresentados nas próximas seções.

6. Elemento <local>

O elemento <local> delimita a localidade na qual será definida a autoridade emitente do documento. É importante ressaltar que esse elemento não está relacionado ao conteúdo e sim à autoridade emitente. Por exemplo, para a autoridade “camara.municipal” o elemento <local> indicará o nome do município no qual a câmara municipal está localizada.

O elemento <detalhamento-local> tem a seguinte estrutura:

URN Canônica	<pre> <local> ::= <codigo-pais> [<detalhamento-local>] <detalhamento-local> ::= ([";" <unidade-federacao> [";" <municipio>]]) ([";" <unidade-federacao>] [";" <local-judiciario>]) <detalhe-ramo> ::= (("regiao." "zona.") 1* <numeral>) ("secao.judiciaria" "comarca") (<unidade-federacao> *[", " <unidade-federacao>]) (<municipio> *[", " <municipio>])) </pre>
URN de Referência	<pre> <detalhamento-local> ::= ([";" <unidade-federacao-ou-uf> [";" <municipio>]]) ([";" <unidade-federacao-ou-uf>] [";" <local-judiciario>]) <detalhe-ramo> ::= (("regiao." "zona.") 1* <numeral>) ("secao.judiciaria" "comarca") (<unidade-federacao-ou-uf> *[", " <unidade-federacao-ou-uf>]) (<municipio> *[", " <municipio>])) <unidade-federacao-ou-uf> ::= <unidade-federacao> 2<minúscula> </pre>
Comum	<pre> <unidade-federacao> ::= 1*(<minuscula> ".") <municipio> ::= 1*(<minuscula> ".") <local-judiciario> ::= <ramo-justica> *[";" <detalhe-ramo>] <ramo-justica> ::= "justica." ("federal" "trabalho" "eleitoral" "militar" "estadual") </pre>

Os elementos <unidade-federacao> e <municipio> representam o nome da unidade da federação e do município, respectivamente, ambos grafados por

extenso em minúsculas, sem abreviaturas, conectivos, apóstrofo, sinais de pontuação ou espaços. Nas URNs de Referência, poderá ser informada a sigla da unidade da federação, ao invés do nome por extenso. Na URN Canônica, o nome da unidade da federação deve ser grafado por extenso.

A segunda forma de composição do elemento <detalhamento-local> especifica a jurisdição dos órgãos do Poder Judiciário, considerando os vários ramos de especialização da justiça brasileira. É importante ressaltar que o nome da unidade da federação precede o nome do ramo da justiça apenas no caso da “justiça estadual”.

Exemplos de valores para o elemento <local>:

Jurisdição:	Representação (elemento <local>):
-----	-----
Brasil	br
Pindamonhangaba-SP	br;sao.paulo;pindamonhangaba
João Pessoa-PB	br;paraiba;joao.pessoa
Estado de São Paulo	br;sao.paulo
Distrito Federal	br;distrito.federal
São Paulo-SP	br;sao.paulo;sao.paulo
São João do Meriti-RJ	br;rio.janeiro;sao.joao.meriti
TSE	br;justica.eleitoral
TRE da Paraíba	br;justica.eleitoral;paraiba
TRT da 13ª Região	br;justica.trabalho;regiao.13
Zona 22 do TRE-PB	br;justica.eleitoral;paraiba;zona.22
Comarca de Macapá-AP	br;amapa;justica.estadual;comarca;macapa
Seção Judiciária Goiás	br;justica.federal;regiao.1;secao.judiciaria;goias

7. Elemento <documento>

O elemento <documento> deve conter informações adequadas a identificar univocamente as entidades referentes aos documentos jurídicos e legislativos, as quais são essencialmente: a autoridade emissora, o tipo de provimento e o descritor (datas, números, etc.).

A estrutura do identificador do documento é composta por três elementos fundamentais, claramente distintos entre si e de acordo com uma ordem que identifica domínios e competências sempre mais restritas:

Comum	<documento> ::= <autoridade> ":" <tipo-documento> ":" <descriptor>
-------	--

Os elementos principais do nome uniforme são geralmente articulados em diversos componentes elementares, para os quais são fixadas regras de representação (critérios, modalidade, sintaxe e ordem).

8. Elemento <autoridade>

8.1. Indicação da Autoridade

O elemento <autoridade> indica, de acordo com o caso:

- a efetiva autoridade emissora da norma, isto é, aquela que realmente adotou ou da qual emanou o ato;
- a instituição por meio da qual o ato é registrado, conhecido e referenciado, ainda que emitido ou de alguma forma produzido por outros sujeitos: é o caso, por exemplo, dos projetos de leis que são identificadas com a Casa Legislativa na qual são apresentadas e não com seus autores e subscritores (parlamentares, poder executivo etc.);
- a instituição regulamentada (e indicada nas referências) pela norma, ainda que emitida por outra autoridade: é o caso, por exemplo, do estatuto de um ente. Nesta hipótese o nome uniforme será um alias do ato de adoção ou emanção.

8.2. Pluralidade dos Sujeitos

Alguns atos têm diversas autoridades emittentes, quando emanadas de uma pluralidade de sujeitos (ex.: portaria interministerial, acordos etc.). Neste caso, no elemento <autoridade> são representados, adequadamente separados, todos os sujeitos envolvidos segundo a forma:

Comum	<autoridade> ::= <sujeito> *["," <sujeito>]
-------	--

(ex.: "ministerio.fazenda,ministerio.justica").

Cabe ressaltar que, no caso de diversas autoridades emittentes, de forma independente da seqüência em que os nomes aparecem no documento, deve-se utilizar a ordem alfabética dos nomes codificados dos sujeitos na composição do nome uniforme. Em uma URN de referência, podem-se especificar os nomes dos sujeitos em qualquer ordem, pois, no processo de resolução, eles serão convertidos para a ordem alfabética.

(ex.: "ministerio.justica,ministerio.fazenda"

e "ministerio.fazenda,ministerio.justica" referenciam, de forma equivalente, a mesma autoridade emittente).

8.3. Modalidade de Indicação do Sujeito

Cada autoridade emissora (aquela que produz o ato) ou referenciada (aquela que é regulamentada pelo ato) consiste essencialmente em um cargo institucional (ex.: Presidente da República) ou em uma instituição (ex.: Presidência da República). Quando for necessária, para uma identificação precisa da norma, a indicação de um órgão no interior da instituição, esta é efetuada segundo a ordem hierárquica do geral para o particular, partindo da instituição, seguindo os vários níveis da estrutura

(Secretaria, Departamento etc.) e terminando com o cargo relativo (Presidente, Diretor, Secretário etc.). Portanto, a estrutura do sujeito emitente fica sendo:

Comum	<pre> <sujeito> ::= (<instituicao> *[";" <orgao>] [";" <funcao>]) <cargo> <instituicao> ::= 1*<normal> <orgao> ::= 1*<normal> <funcao> ::= 1*<normal> <cargo> ::= 1*<normal> </pre>
-------	---

(ex.: "ministerio.fazenda;secretaria.receita.federal;secretario")

8.3.1. Indicação do órgão

Em alguns tipos de norma, mesmo a função da autoridade do órgão emitente sendo univocamente determinada (ex.: Secretário da SRF), geralmente esta não é indicada nas referências. Nesses casos, em analogia ao que é adotado na prática, a indicação da autoridade emitente é limitada ao mínimo indispensável em função do tipo do ato.

(ex.: "ministerio.fazenda;secretaria.receita.federal:instrucao.normativa")

e não "ministerio.fazenda;secretaria.receita.federal;secretario:instrucao.normativa")

8.3.2. Indicação da função

Em linhas gerais o componente <funcao>:

- o é indicado, às vezes no lugar do próprio órgão, quando se trata de um cargo do tipo "político", representativo ou eletivo, ou caso seja natural ao tipo de ato

(ex.: "universidade.brasilia;reitor:oficio")

e não "universidade.brasilia;reitoria:oficio");

- o é indicado quando se trata de uma figura maior da instituição (ex.: diretor geral, secretário geral, diretor administrativo etc.), à qual nem sempre é possível associar uma estrutura interna específica

(ex.: "ministerio.justica;departamento.policia.federal;diretor.geral");

- o não é indicado quando corresponde à responsabilidade de uma estrutura (tipicamente preenchida por um funcionário ou por um dirigente); nesse caso indica-se apenas a estrutura e não o dirigente

(ex.: "ministerio.justica;departamento.administracao.penitenciaria")

e não "ministerio.justica;departamento.administracao.penitenciaria;chefe").

Por outro lado, a função deverá ser indicada quando:

- o não coincide com a direção ou a responsabilidade da estrutura (ex.: subsecretário, vice-diretor etc.);
- o o tipo de ato possa ser de origem tanto monocrática quanto colegiada: em tal caso a indicação do cargo serve para eliminar a ambigüidade.

8.3.3. Convenções para a autoridade de documentos normativos

Por convenção, as normas do tipo constituição, lei orgânica, emendas, leis complementares, leis delegadas, medidas provisórias, decretos-leis e decretos têm a autoridade indicada apenas com os termos "federal", "estadual" ou "municipal", dependendo da esfera pública.

(ex.: "urn:lex:br:**federal**:emenda.constitucional"

e não "urn:lex:br:congresso.nacional:emenda.constitucional",

"urn:lex:br;sao.paulo;campinas:**municipal**:decreto"

e não "urn:lex:br;sao.paulo;campinas:prefeitura:decreto")

"urn:lex:br;sao.paulo;campinas:**municipal**:lei"

e não "urn:lex:br;sao.paulo;campinas:camara.municipal:lei")

As resoluções, portarias, instruções normativas e outras normas de hierarquia inferior devem especificar a autoridade emitente de forma unívoca.

9. Elemento <tipo-documento>

9.1. Critérios para a Indicação do Tipo do Documento

A indicação obrigatória da autoridade emitente nos nomes uniformes torna supérflua, na grande maioria dos casos, a qualificação adicional da norma (lei municipal, portaria ministerial etc.), mesmo se não sistematicamente, pelo menos na prática.

Quando a combinação autoridade-norma identifica claramente o documento em questão, o tipo de ato não é qualificado por meio de atributos que se refiram à autoridade emitente.

(ex.: "urn:lex:br;sao.paulo;ribeirao.preto:municipal:lei"

e não "urn:lex:br;sao.paulo;ribeirao.preto:municipal:lei.municipal")

9.2. Especificação Adicional do Tipo de Documento

Para a identificação do ato no elemento <tipo-documento>, geralmente é suficiente a indicação da sua tipologia. Todavia, é bastante difundida na prática a utilização de siglas que identificam um subtipo do ato (ex.: PEC – Proposta de Emenda Constitucional; ADI – Ação Direta de Inconstitucionalidade). Nesses casos, o tipo de documento pode ser seguido por especificações, de modo a viabilizar a identificação, conforme apresentado a seguir:

Comum	<pre> <tipo-documento> ::= ((<tipo-norma> <tipo-jurisprudencia> <tipo-projeto-norma>) [";" <nome-subtipo-sequenciador >]) ("publicacao.oficial;" <nome-periodico-oficial> [";" <nome-secao-periodico-oficial>] [";" <detalhe-extra-suplemento>]) <tipo-projeto-norma> ::= 1*<normal> <tipo-norma> ::= 1*<normal> <tipo-jurisprudencia> ::= 1*<normal> <nome-subtipo-sequenciador> ::= 1*<normal> <nome-periodico-oficial> ::= 1*<normal> <nome-secao-periodico-oficial> ::= 1*<normal> <detalhe-extra-suplemento> ::= ("edicao.extra" "suplemento") ["." 1*<numeral>] </pre>
-------	---

Os possíveis valores para `<tipo-norma>`, `<tipo-jurisprudencia>` e `<tipo-projeto-norma>` são definidos em vocabulários específicos.

O detalhamento `<nome-subtipo-sequenciador>` identifica o subtipo do documento que foi utilizado na seqüência de numeração dos documentos no contexto dos sistemas de informação da autoridade emitente. Os valores são especificados no vocabulário controlado utilizando o contexto da localidade e da autoridade emitente. Por exemplo, o valor “plc”, no contexto do Senado Federal, é definido como um subtipo de “projeto.lei.complementar”. Nesse caso específico, pode-se utilizar a sigla diretamente na codificação da URN Canônica.

9.3. Especificação da Publicação Oficial

O tipo "publicacao.oficial" permite identificar um periódico oficial que publica os textos das normas, julgados e proposições legislativas. Ao codificar informações sobre a publicação oficial, deve-se ser o mais específico possível, devendo-se codificar o nome da publicação oficial por extenso e, se existente, a identificação da seção. No caso de edições-extras e suplementos, deve-se utilizar o elemento `<detalhe-extra-suplemento>`.

(ex: “publicacao.oficial;diario.oficial.uniao;secao.1” ou

“publicacao.oficial;diario.justica.eletronico” ou

“publicacao.oficial;diario.oficial.uniao;secao.1;suplemento”)

10. Elemento `<descriptor>`

10.1. Indicação do Descritor

A codificação do descritor do documento geralmente constitui-se de uma data representativa (assinatura no caso de normas, iniciativa para proposições legislativas e data de julgamento para os julgados) e um número de identificação (seqüencial, número de protocolo etc.). No caso de componentes adicionais do documento, como, por exemplo, anexos, adicionam-se os descritores destes componentes separados por “;”.

No que se refere à data representativa, deve-se observar que algumas normas apresentam datas múltiplas. Em relação ao número do documento, este pode não existir de fato (ex.: atos não numerados) ou ainda ser múltiplo (ex.: atos unificados). Por essa razão, prevê-se que os elementos de codificação possam ter valores múltiplos. Nestes casos, utiliza-se o caractere “,” para delimitar cada valor.

Para efeito de implantação da Rede de Informações LexML, considerando que muitos sistemas de informações não possuem o registro da data na sua forma completa, será permitido informar apenas o ano da data representativa na URN Canônica. No futuro, essa permissão poderá ser revista, tornando, por exemplo, obrigatória a especificação da data completa para os documentos cuja data representativa seja maior do que uma data específica. Recomendamos o uso da data completa, sendo esta obrigatória nos casos dos documentos não numerados. No caso das URNs de Referência, sempre será possível abstrair o dia e o mês da data representativa.

De acordo com a estrutura global do nome uniforme, do geral ao particular, define-se como formato de indicação a seqüência “data número”, na forma:

URN Canônica	<pre> <descriptor> ::= <datasOuAno> ";" <identificadores> *[";" <id-componente> ["," <titulo-componente>]] [";"retificacao." <seq-retificacao>] </pre>
URN de Referência	<pre> <descriptor> ::= ((<datas> [";" <identificadores>]) (<datasOuAno> ";" <identificadores>) ([<datasOuAno> ";"] <apelido-documento>)) *[";" <id-componente> ["," <titulo-componente>]] [";"retificacao." <seq-retificacao>] </pre>
Comum	<pre> <id-componente> ::= 1*<normal> <titulo-componente> ::= 1*<normal> <seq-retificacao> ::= 1*<numeral> <datasOuAno> ::= <datas> <ano> </pre>

(ex.: "2000-12-06;126",

"2003-10-01;10741",

```
"2003-10-01;10741;anexo.1",
"2003-10-01;estatuto.idoso",
"1993-07-20;8685" e
"1993-07-20;lei.audiovisual" )
```

10.2. Elemento <identificadores>

O número do documento é representado pelo elemento <id-documento>. Alguns documentos podem ser identificados por uma série de números. Nesse caso, relacionam-se no campo <identificadores> todos os números.

O elemento <identificadores> possui a seguinte estrutura:

Comum	<pre><identificadores> ::= (<id-documento> *["," <id-documento>]) (<numero-lex>) (<numero-seq>) <numero-lex> ::= "lex-" 1*<numeral> <numero-seq> ::= "seq-" <sigla-orgao> "-" 1*<numeral> <sigla-orgao> ::= 1*<normal> <id-documento> ::= 1*<normalID></pre>
-------	--

(ex. "2000-06-12;c-10-97,c-11-97,c-12-97")

Os caracteres não permitidos (ex.: "/") ou reservados (ex.: ":") , incluindo a vírgula, não podem estar presentes no interior do <id-documento>, e portanto são convertidos em "-".

Os elementos <numero-lex> e <numero-seq> são utilizados para identificação de documentos sem numeração, conforme especificação apresentada na próxima seção.

10.2.1. Documentos sem Numeração

Os documentos sem numeração têm uma referência não unívoca, já que podem existir diversos documentos do mesmo tipo, emitidos no mesmo dia pela mesma autoridade.

Dado que um nome uniforme deve identificar univocamente cada um deles, o campo <descriptor> deve conter um elemento discriminante. A unicidade deste número seqüencial será gerada pelo sistema LexML considerando a localidade, a autoridade emitente, o tipo de documento e a data de assinatura, além dos metadados descritivos disponíveis, considerando, sempre que possível, a ordem em que o documento aparece na publicação oficial. Para tornar este identificador gerado claramente distinguível, tal número é precedido pela *string* "lex-".

Comum	<pre><numero-lex> ::= "lex-" 1*<numeral></pre>
-------	--

(ex. "br:federal:decreto:1999-12-21;lex-16")

Caso o provedor de dados utilize um seqüencial de identificação diverso daquele utilizado pelo LexML, poderá indicar o número do descritor próprio, qualificado pela sigla do órgão (conforme vocabulário controlado), utilizando a seguinte codificação:

Comum	<pre><numero-seq> ::= "seq-" <sigla-orgao> "-" 1*<numeral> <sigla-orgao> ::= 1*<normal></pre>
-------	---

(ex. "urn:lex:br:federal:decreto:1999-12-21;seq-sf-32")

Rotinas de conciliação poderão relacionar os descritores `<numero-lex>` e `<numero-seq>` correspondentes, tendo por base os metadados descritivos.

Caso ocorra, por erro, a atribuição de múltiplos identificadores `<numero-lex>` para um mesmo documento, deve-se utilizar o recurso de equivalência entre identificadores para aglutinar os múltiplos registros de identificação do documento.

10.2.2. Indicação do Número de Identificação do Documento

Para facilitar a construção automática do nome uniforme a partir das referências, o número de identificação do ato é representado exatamente na forma que foi atribuído pela autoridade emitente, considerando as conversões necessárias de caracteres especiais.

(ex.: um ofício com a numeração AIPA/CR/35 de 06/11/2001 teria como elemento `<descriptor>` do nome uniforme: "2001-11-06;aipa-cr-35").

De acordo com as regras de numeração adotadas pelo emanante, este pode aceitar a presença de informações redundantes (ex.: autoridade, ano de emissão etc.) já presentes em outras partes do nome uniforme (na mesma forma ou em forma diversa).

Por outro lado, a eliminação automática de tais redundâncias a partir das citações que trazem o número de maneira completa (ex.: ofício AIPA/CR/35) não é uma operação confiável, já que existe uma variabilidade enorme nas regras de numeração adotadas pelas autoridades emitentes.

Tendo em conta que as referências podem ser efetuadas também de forma reduzida (ex.: ofício AIPA no. 35 de 2001), o que faz com que o nome uniforme construído também seja reduzido, no final do processo de resolução o número identificador é considerado como sendo formado por diversos componentes elementares, cada um deles delimitado por caracteres especiais.

10.2.3. Esclarecimentos Sobre os Números de Identificação do Documento

Na sintaxe dos nomes uniformes, o número singular (`<id-documento>`) do elemento `<identificadores>` é, no seu todo, uma *string* opaca que, combinada com os outros elementos, forma um identificador unívoco.

Na numeração, são relacionadas na mesma ordem todas as informações presentes no documento que caracterizam a sua identificação, mesmo que algumas (ex.: ano) já estejam contidas em outros elementos (ex.: data).

Assim, por exemplo, se o identificador de um ofício é 409/98, utiliza-se a numeração completa "409-98" e não apenas 409, mesmo que o ano 1998 já seja um conteúdo da data.

Não existe regra pela qual os componentes devam ser representados no interior de uma numeração em uma determinada ordem (ex.: primeiro o número sequencial e depois o ano, ou vice-versa): assim, por exemplo, o nº 409/98 é reportado como "409-98", o nº 98/409 como "98-409". Os dois números "409-98" e "98-409" são ambos válidos e identificam, considerando a paridade dos outros elementos, dois documentos distintos.

Assim, utiliza-se exatamente a forma adotada pela autoridade emitente, segundo a prática difundida no seu interior, para uma classe (ou período) de atos, eventualmente distinta de outras classes ou períodos (ex.: para os ofícios: número/ano; para os memorandos: ano/número).

No caso de normas jurídicas que utilizem as autoridades emittentes convencionadas para cada esfera administrativa ("federal", "estadual" ou "municipal"), não se deve codificar o ano da norma no elemento `<id-documento>`.

É prevista ainda a presença de diversos números identificadores do documento; por princípio, tais números seriam da mesma natureza, atribuídos, assim, com base em um registro comum (ex.: reunificação de projetos de lei, de casos etc.) e, desse modo, congruentes entre si. Em tal hipótese, a ordem não tem relevância, mas é o conjunto dos números singulares que distingue o ato.

Se o documento tem diversos números, estes podem ser representados na mesma seqüência em que aparecem no documento: no esquema é irrelevante a ordem na qual os números singulares são representados (crescente, decrescente, nenhuma) para a unicidade do nome uniforme (ex.: "135,409" e "409,135" são ambos válidos, e identificam, considerando-se a paridade dos outros elementos, o mesmo documento).

10.3. Identificador de Componentes de um Documento

O elemento `<id-componente>` deve ser utilizado nos casos em que um documento é acompanhado de outros documentos (ex.: ocorrência de anexos). Este identificador é formado por um descritor curto acompanhado de um número sequencial (ex.: anexo.1) seguido, sempre que possível, de um título (`<titulo-componente>`) pelo qual o documento é conhecido.

(ex: decreto:1980-08-18;85080;anexo.1,tratado.cooperacao.amazonica
e não decreto:1980-08-18;85080;anexo.1)

10.3.1. Anexos Dependentes Funcionais e Formais

A presença de anexos de várias naturezas (anexos, tabelas, mapas etc.) é muito freqüente nos documentos legislativos e jurídicos e não é raro o caso no qual esses assumem a forma de textos articulados (ex.: regulamentos). Apesar de quase sempre serem partes integrantes do documento normativo principal, os anexos podem ser referenciados (e mesmo sofrer modificações) de maneira independente do ato a que pertencem. Torna-se, assim, necessário identificar univocamente não apenas o documento principal, mas também cada um dos seus anexos.

Os anexos dependentes formais são subdivisões efetivamente previstas no documento principal com o objetivo de tornar o seu texto mais fluente e organizado; esses são claramente nominados ou facilmente identificáveis no ato (ex.: “Anexo I”) e geralmente referenciados no mesmo texto (ex.: "no Anexo I encontra-se a lista ...").

Já os anexos dependentes funcionais são, ao contrário, partes integrantes do próprio documento principal, que têm, no entanto, uma forma de registro não-textual e que por isso, na representação eletrônica do documento, precisam ser forçosamente segregados de forma a constituir objetos separados. O documento completo será recomposto na fase de apresentação. Isso acontece regularmente na presença de objetos multimídia (ex.: mapas geográficos, imagens, fotografias etc.), mas podem apresentar-se também como textos registrados em formato eletrônico não textual (por ex.: a imagem digitalizada da versão em língua estrangeira de um acordo internacional). É necessário indicar, no elemento que trata da forma da expressão, a natureza do conteúdo, de acordo com o respectivo vocabulário controlado.

10.3.2. Identificação do Anexo

Ao anexo de qualquer natureza é sempre atribuído um nome uniforme distinto, independente do seu valor autônomo e da sua forma de registro (junto ou separado do documento principal). O seu nome uniforme é formado pelo nome uniforme do documento ao qual está vinculado e por uma desinência identificadora do próprio anexo.

Para os anexos dependentes formais, a desinência traz a denominação oficial do anexo e eventuais especificações adicionais (ex.: título), para facilitar a recuperação nos casos de falta do identificador.

(ex.: "urn:lex:br:ministerio.fazenda;secretaria.receita.federal:instrucao.normativa:2007-02-12;25;anexo.a,tabela.aliquota.imposto.renda.pessoa.fisica")

Para os anexos dependentes funcionais, na falta de uma denominação oficial, utilizam-se diretamente as informações que ajudam na sua identificação, como, por exemplo, a legenda.

(ex.: se no interior da instrução normativa anterior fosse apresentado um gráfico com a evolução da arrecadação, seria criado um anexo funcional com o nome uniforme "urn:lex:br:ministerio.fazenda;secretaria.receita.federal:instrucao.normativa:2007-02-12;25;grafico.evlucao.arrecadacao")

Os caracteres não admitidos (ex.: "/") ou reservados (ex.: ":") não podem estar presentes dentro do <id-componente>, e, portanto, deverão ser substituídos por ".".

10.3.3. Anexos de anexos.

Nos casos em que um anexo contenha um ou mais anexos, os identificadores correspondentes serão construídos adicionando-se ao identificador do anexo-pai os dos anexos vinculados.

(ex.: Supondo-se que o gráfico do exemplo anterior estivesse contido em um anexo específico da Instrução Normativa nº 25 que tratasse apenas da justificativa para a correção da tabela do IRPF, o nome uniforme para esse gráfico seria: "urn:lex:br:ministerio.fazenda;secretaria.receita.federal:instrucao.normativa:2007-02-12;25;anexo.b.justificativa.correcao;grafico.evlucao.arrecadacao").

10.3.4. Descritor de Retificação

Nos casos do texto da retificação de um documento, deve-se, após a qualificação do documento no elemento descritor, informar o termo “retificacao.” seguido de um seqüencial de ordem da retificação do documento. Por sua vez, o elemento <versao> deve apresentar três sub-elementos:

1. a data da versão do ato a que a retificação se refere, que, no caso de normas, é a data de início da vigência;
2. o nome do evento “publicacao”; e
3. a data da publicação da retificação.

(ex.: lei:1990-09-11;8078;retificacao.1@1990-09-12;publicacao;2007-01-10).

10.4. Descritor da Publicação Oficial

O elemento descritor, no caso do identificador da publicação oficial, contém o número da edição, conforme exemplo abaixo:

(ex.:urn:lex:br:imprensa.nacional:publicacao.oficial;diario.oficial.uniao;secao.1:2008-07-22;139)

No caso de não existir o número da edição, deve-se aplicar a regra especificada para documentos não numerados.

(ex.:urn:lex:br:congresso.nacional:publicacao.oficial;diario.congresso.nacional:2003-02-01;seq-cn-0)

11. Elemento <fragmento>

A possibilidade de identificar univocamente o fragmento de um documento principal ou de seu anexo é importante, pois permite estabelecer relacionamentos mais precisos, como nos casos dos relacionamentos entre dispositivos e normas. Por exemplo, a regulamentação do artigo de uma lei pode ser realizada por um decreto.

Nesse caso, ao invés de relacionar as normas deve-se estabelecer relacionamento entre o dispositivo da lei e o decreto no todo.

O elemento `<fragmento>` deve ser utilizado apenas na identificação das partições internas de uma expressão, como, por exemplo, no caso de uma norma jurídica, na identificação de artigos, dispositivos que agrupam artigos (títulos, capítulos etc.) ou detalhamento de artigos (parágrafo, inciso, alínea etc.). Os elementos de identificação das partições internas devem seguir as regras de nomenclatura previstas no XML Schema do Projeto LexML (Parte 3).

Comum	<pre> <fragmento> ::= (<id-particao> <intervalo-ids>) *[" , " <fragmento>] <intervalo-ids> ::= "[" <id-particao> " , " <id-particao> "]" <id-particao> ::= 1* <normalID> </pre>
-------	--

(ex.: "lex:br:federal:lei:2000-12-06;126!art5_par2",
 "lex:br:federal:lei:2000-12-06;126![art6,art10]",
 "lex:br:federal:lei:2000-12-06;126![art6,art10],art12,[art20,art30]",
 "lex:br:federal:lei:1993-07-20;lei.audiovisual!tit1")

A semântica desses fragmentos é fixada pela semântica XPointer (DeRose et al. 2002) no seguinte mapeamento (esboço de algoritmo para mapeamento da sintaxe LexML-fragmento em sintaxe XPointer):

1. todo `<id-particao>` é convertido para `"id('strIdParticao')"`
2. todo `<intervalo-ids>` é convertido para `"id('strId1')/range-to(id('strId2'))"`
3. o conjunto com mais de um fragmento (separado por vírgula) é convertido em um conjunto separado por `" | "` (*pipe*).
4. a *string* resultante é convertida para `"xpointer(strResultante)"`

Aplicando este algoritmo ao fragmento "art5_par2", gera como resultado o valor `"xpointer(id('art5_par2'))"`.

O mapeamento de `"[art6,art10],art12,[art20,art30]"` resulta em `"xpointer(id('art6')/range-to(id('art10'))|id('art12')|id('art20')/range-to(id('art30')))"`.

11.1. Fragmento da Publicação Oficial

O elemento `<fragmento>` permite especificar, no caso da publicação oficial, a página inicial (prefixo "pag") e, se necessário, separado por sublinhado ("_"), a coluna (prefixo "col") onde o documento foi publicado.

(ex.: urn:lex:br:imprensa.nacional:publicacao.oficial;diario.oficial.uniao;secao.1:1997-07-17;222:pag15081_col1).

Caso a publicação oficial possua um identificador único do documento visível na própria publicação, pode-se utilizar o prefixo “doc” seguido do número de identificação do documento.

(ex.: urn:lex:br:superior.tribunal.justica:publicacao.oficial;diario.justica.eletronico;2008-11-05;251;doc4312835).

12. Elemento <versao>

12.1. As Versões e Visões de um Texto

Podem existir diversas versões do texto de um documento, cada uma delas caracterizada pelo intervalo temporal no qual o texto pode ser considerado referenciável ou válido. O período de vigência ou validade de uma versão termina com o início da versão sucessiva.

O conceito de versão está ligado ao evento de entrada em vigor do texto no ordenamento jurídico (no caso de normas) ou de validade (no caso de proposições legislativas e julgados). O conceito de visão, por sua vez, está ligado ao evento que gerou uma nova versão ou uma nova visão de uma versão existente.

Em alguns casos, como na publicação de uma retificação, passam a existir duas visões de uma mesma versão do texto do documento: a visão anterior e a visão posterior ao evento de retificação. Além da retificação, outros eventos, que serão detalhados a seguir, podem gerar novas visões para uma mesma versão.

A data da versão é originada a partir da data do evento:

- o de entrada em vigor de um novo ato normativo;
- o de entrada em vigor das modificações realizadas por outros atos;
- o de republicação por erro, realizada após o *vacatio legis*.
- o de publicação de uma iniciativa de projeto de norma;
- o de consolidação de uma emenda ao projeto de norma;

A data da visão é originada a partir da data do evento:

- o de publicação de um novo ato normativo ou projeto de norma;
- o de publicação de correções devido a erros na publicação (retificações ou erratas);
- o de publicação de decisão judicial com reflexo na norma jurídica;
- o de entrada em vigor das modificações realizadas por outros atos normativos antes da entrada em vigor do ato alterado;
- o de republicação por erro realizada antes do *vacatio legis*.
- o de publicação da derrubada de veto parcial.
- o (qualquer) que gere alguma variante do texto de uma versão.

12.2. Casos Especiais

A criação do texto atualizado de uma norma pode considerar uma ou mais datas de vigência, de acordo com os seguintes tipos:

- multivigente: quando, no interior de um único texto, são adicionadas marcações adequadas para identificar as partes modificadas (inseridas, substituídas e suprimidas) e os relativos períodos de vigência. Um documento desse tipo pode ser capaz, de maneira dinâmica, de se apresentar de formas distintas, de acordo com a data de vigência desejada;
- monovigente: quando, por outro lado, para cada variação sofrida pelo texto a uma certa data, é criado um novo texto distinto. Cada texto é, portanto, caracterizado por um período próprio de validade ou vigência.

12.3. Identificação da Versão

Para identificar, no tempo, as diversas versões de um mesmo documento, o nome uniforme de cada versão é construído agregando-se uma desinência particular àquela do documento original.

Os elementos que qualificam o documento individual são definidos conforme as seguintes regras:

URN Canônica	<pre> <versao> ::= [(<datas> "versao.original" "inicio.vigencia" "multivigente") [";" <evento> [";" <visao>]]] </pre>
URN de Referência	<pre> <versao> ::= [(<datas> "versao.original" "inicio.vigencia" "multivigente" "versao.vigente.em;" <data> "versao.eficaz.em;" <data> "versao.consultada.em;" <data>) [";" <evento> [";" <visao>]]] </pre>
Comum	<pre> <evento> ::= 1*<normal> <visao> ::= <datas> </pre>

A especificação da <versao> inicia com a data do início de período de vigência ou validade do documento. Caso o sistema de informação não possua esta data,

pode-se especificar, no caso de normas, o valor “início.vigência” e, para os demais casos, o valor “versão.original”. No caso do texto multiversão, pode-se especificar uma lista de datas de vigência separadas por “,” (vírgula), um período entre duas datas delimitadas por “[” e “]” (ex.: “[2003-05-01,2005-04-02]”) ou ainda o valor “multivigente”.

No caso das URNs de Referência, é possível referenciar o texto vigente ou eficaz em uma determinada data, ou ainda o texto consultado em uma data específica.

O elemento <evento> indica o evento que define a visão de um documento. Caso não seja informado, considera-se a publicação como o evento default. Um vocabulário específico (ver Parte 6 – Vocabulário Controlado) define os possíveis valores dos eventos.

O elemento <visão> reporta a data do evento que define uma visão do documento. No caso do texto multivisão, pode-se especificar uma lista de datas de visões separadas por “,” (vírgula) ou um período entre duas datas delimitadas por “[” e “]” (ex.: “[2003-05-01,2005-04-02]”).

12.4. Exemplo - CDC

Para exemplificar a relação entre os componentes do elemento visão, vamos analisar como os eventos de publicação, alteração (neste exemplo consideramos apenas as duas primeiras alterações desta norma) e retificação do CDC (Código de Defesa do Consumidor) são modelados. Os eventos citados são apresentados na Tabela 2 e são a base para a construção das URNs de cada versão, apresentadas na Tabela 3.

Tabela 2. Eventos do ciclo de vida do CDC (Lei 8078/1990).

Evento	Nome do Evento	Versão	Visão
E1	Criação da norma com entrada em vigor em 11/03/1991 (180 dias da publicação)	Vigência* 11/03/1991	Assinatura 11/09/1990
E2	Publicação da norma	Vigência 11/03/1991	Publicação 12/09/1990
E3	Entrada em vigor da norma alteradora, Lei 8656/1993 (publicada em 22/05/1993, com entrada em vigor na publicação)	Entrada em vigor da alteração. 22/05/1993	Publicação 22/05/1993
E4	Entrada em vigor da norma alteradora, Lei 8703/1993 (publicada em 08/09/1993, com entrada em vigor na publicação)	Entrada em vigor da alteração. 08/09/1993	Publicação 08/09/1993
E5	Retificação do texto publicado (E2)	[Todas as vigências desde evento E2]	Publicação 10/01/2007

Tabela 3. URNs do CDC (em ordem da ocorrência de eventos)

Ref	Tipo URN	Sufixo da URN (“urn:lex:br:federal:lei” + [sufixo da urn])
E1	Doc. Individual	1990-09-11;8078@1991-03-11;assinatura;1990-09-11
E2	Doc. Individual	1990-09-11;8078@1991-03-11;publicacao;1990-09-12
E3	Doc. Individual	1990-09-11;8078@1993-05-22;alteracao;1993-05-22
E4	Doc. Individual	1990-09-11;8078@1993-09-08;alteracao;1993-09-08
E5	Doc. Individual	1990-09-11;8078@1991-03-11;retificacao;2007-01-10
E5	Doc. Individual	1990-09-11;8078@1993-05-22;retificacao;2007-01-10
E5	Doc. Individual	1990-09-11;8078@1993-09-08;retificacao;2007-01-10

12.5. Valores *Default* para os Elementos da Versão

Por convenção, iremos considerar os seguintes valores *default* para as URNs de documentos individuais que não especifiquem valores para a data da versão, evento e visão:

- Data da Versão (*default*) – A versão do início de vigência (norma) ou validade de um texto (julgados e proposições legislativas), representado respectivamente pelo valores *default* “inicio.vigencia” e “versao.original”.
- Evento (*default*) - O evento da primeira publicação do documento, representado pelo valor *default* “publicacao”.
- Data da Visão (*default*) – A data do evento que tornou a versão pública (normalmente, a data da primeira publicação).

13. Elemento <forma>

O elemento <forma> identifica a forma de expressão do recurso endereçado. Caso a forma seja textual, deve-se informar o código da língua em que o texto está codificado. Nos casos de elementos não textuais, deve-se informar a natureza do recurso de acordo com o vocabulário controlado de tipo de conteúdo.

Comum	<pre> <forma> ::= <tipo-forma> [";" <lingua>] *["," <lingua>]] <tipo-forma> ::= 1*<normal> <lingua> ::= <lingua-codigo> ["-" <lingua-script>] ["-" <lingua-regiao>] <lingua-codigo> ::= 1*<minuscula> <lingua-script> ::= 1*<minuscula> <lingua-regiao> ::= 1*<minuscula> </pre>
-------	---

(ex.: federal:lei:1992-05-11;8421@1992-05-12;assinatura;1992-05-12~texto;pt-br

```
federal:lei:1992-05-11;8421;anexo.1,desenho.bandeira.nacional  
@1992-05-12;assinatura;1992-05-12~imagem )
```

No caso de um documento que possui seu conteúdo expresso em múltiplas línguas, deve-se especificar cada código separado por vírgulas.

(ex.: `federal:lei:1992-05-11;8421@1992-05-12;assinatura;1992-05-12~texto;pt-br,es-ar`)

Por convenção, considera-se o valor “`texto;pt-br`” como *default* para o elemento `<forma>` de uma URN de documento individual.

14. Referências

14.1. Bibliográficas

Moats, R; Sollins, K. R. *URN Syntax*, RFC 2141, Maio 1997.

Berners-Lee, T; Fielding, R; Masinter, L. *Uniform Resource Identifiers (URI): Generic Syntax*, RFC 2396, Agosto 1998.

DeRose, S; Maler, E; Daniel Jr., Ron. *XPointer xpointer() Scheme*. W3C Working Draft, Dezembro de 2002. Disponível em: <http://www.w3.org/TR/xptr-xpointer/>. Acessado em: 4 de Novembro de 2008.

14.2. Sites

<http://cidoc.ics.forth.gr/> - CIDOC CRM

<http://www.ifla.org/VII/s13/frbr/> - IFLA / FRBR

15. Lista de Abreviaturas e Siglas

BNF - Backus-Naur Form

CDC – Código de Defesa do Consumidor

HTML – Hypertext Markup Language

IETF - Internet Engineering Task Force

ITTIG - Istituto di Teoria e Tecniche dell'Informazione Giuridica

NIR – Norme in Rete

ODF – Open Document Format

PDF – Portable Document Format

RDS - Resolver Discovery Service

RFC - Request For Comments

URI - Uniform Resource Identifier

URL - Uniform Resource Locator

URN - Uniform Resource Name

XML – Extensible Markup Language

16. Créditos

16.1. Editores (LexML Brasil)

Fernando Ciciliati (Senado Federal / Interlegis)

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

16.2. Grupo de Trabalho LexML (em Dezembro de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Cláudio Mori (Tribunal Superior Eleitoral)

Claudson dos Santos Melo (Tribunal Superior do Trabalho)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda Oliveira de Macedo (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)

Jonatas Rodrigues (Procuradoria Geral da República)

Luís Henrique Raja G. Mitchell (Tribunal de Contas da União)

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

Paulo de Tarso Penna da Costa (Superior Tribunal de Justiça)

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)

16.3. Outras Contribuições

Neide De Sordi (Conselho da Justiça Federal)

Peter de Pádua Krauss (Prefeitura Municipal de São Carlos (SP))

Anexo A – Gramática da URN Canônica

```
*-----
* Estrutura de um nome uniforme (URN) do espaço "lex"
*-----
<URN> ::= "urn:" <NID> ":" <NSS-lex>
<NID> ::= "lex"
*-----
* Estrutura de um nome uniforme do espaço "lex" internacionalizado
*-----
<NSS-lex> ::= <local> ":" <NSS-lex-local>
<local> ::= <codigo-pais> [ <detalhamento-local> ]
*-----
* Estrutura do elemento <local> para o Brasil
*-----
<codigo-pais> ::= "br"
<detalhamento-local> ::=
    ( [ ";" <unidade-federacao> [ ";" <municipio> ] ] )
  | ( [ ";" <unidade-federacao> ] [ ";" <local-judiciario> ] )
<detalhe-ramo> ::= ( ( ("regiao." | "zona.") 1* <numeral> )
    | ( "secao.judiciaria" | "comarca" )
    | ( <unidade-federacao> *[ ", " <unidade-federacao> ] )
    | ( <municipio> *[ ", " <municipio> ] ) ) )
<unidade-federacao> ::= 1*( <minuscula> | "." )
<municipio> ::= 1*( <minuscula> | "." )
<local-judiciario> ::= <ramo-justica> *[ ";" <detalhe-ramo> ]
<ramo-justica> ::= "justica."
    ( "federal" | "trabalho" | "eleitoral" | "militar" | "estadual" )
*-----
* Estrutura de um nome específico do "lex-br"
*-----
<NSS-lex-local> ::= <documentoComplexo>
    | ( <documentoIndividual> [ "!" <fragmento> ] )
<documentoIndividual> ::= <documentoComplexo> [ "@" <versao> ] [ "~" <forma> ]
<documentoComplexo> ::= <documento>
```

```
*-----  
* Estrutura do elemento <documento>  
*-----  
<documento> ::= <autoridade> ":" <tipo-documento> ":" <descriptor>  
*-----  
* Estrutura do elemento <autoridade>  
*-----  
<autoridade> ::= <sujeito> *["," <sujeito> ]  
<sujeito> ::= ( <instituicao> *[";" <orgao> ] [ ";" <funcao> ] )  
                | <cargo>  
<instituicao> ::= 1*<normal>  
<orgao>       ::= 1*<normal>  
<funcao>      ::= 1*<normal>  
<cargo>       ::= 1*<normal>  
*-----  
* Estrutura do elemento <tipo-documento>  
*-----  
<tipo-documento> ::= ( ( <tipo-norma>  
                        | <tipo-jurisprudencia>  
                        | <tipo-projeto-norma>  
                        ) [ ";" <nome-subtipo-sequenciador> ] )  
                | ( "publicacao.oficial;" <nome-periodico-oficial>  
                    [ ";" <nome-secao-periodico-oficial> ]  
                    [ ";" <detalhe-extra-suplemento> ] )  
<tipo-projeto-norma> ::= 1*<normal>  
<tipo-norma>         ::= 1*<normal>  
<tipo-jurisprudencia> ::= 1*<normal>  
<nome-subtipo-sequenciador> ::= 1*<normal>  
<nome-periodico-oficial>   ::= 1*<normal>  
<nome-secao-periodico-oficial> ::= 1*<normal>  
<detalhe-extra-suplemento> ::=  
    ( "edicao.extra" | "suplemento" ) [ "." 1*<numeral> ]  
*-----  
* Estrutura do elemento <descriptor>  
*-----
```

```
<descriptor> ::= <datasOuAno> ";" <identificadores>
                * [ ";" <id-componente> [ "," <titulo-componente> ] ]
                [ ";"retificacao." <seq-retificacao> ]
<identificadores> ::= ( <id-documento> * [ "," <id-documento> ] )
                    | ( <numero-lex> )
                    | ( <numero-seq> )
<id-documento> ::= 1* <normalID>
<numero-lex> ::= "lex-" 1* <numeral>
<numero-seq> ::= "seq-" <sigla-orgao> "-" 1* <numeral>
<sigla-orgao> ::= 1* <normal>
<id-componente> ::= 1* <normal>
<titulo-componente> ::= 1* <normal>
<seq-retificacao> ::= 1* <numeral>
<datasOuAno> ::= <datas> | <ano>
*-----
* Estrutura do elemento <fragmento>
*-----
<fragmento> ::= ( <id-particao> | <intervalo-ids> ) * [ "," <fragmento> ]
<intervalo-ids> ::= "[" <id-particao> "," <id-particao> "]"
<id-particao> ::= 1* <normalID>
*-----
* Estrutura do elemento <versao>
*-----
<versao> ::= [ ( <datas>
                | "versao.original"
                | "inicio.vigencia"
                | "multivigente" )
                [ ";" <evento> [ ";" <visao> ] ] ]
<evento> ::= 1* <normal>
<visao> ::= <datas>
*-----
* Estrutura do elemento <forma>
*-----
<forma> ::= <tipo-forma> [ ";" <lingua> ] * [ "," <lingua> ] ]
<tipo-forma> ::= 1* <normal>
```

```
<lingua> ::= <lingua-codigo>
           [ "-" <lingua-script> ] [ "-" <lingua-regiao> ]

<lingua-codigo> ::= 1*<minuscula>
<lingua-script> ::= 1*<minuscula>
<lingua-regiao> ::= 1*<minuscula>

*-----
* Estrutura do elemento <datas>
*-----

<datas> ::= ( <data> *["," <data> ] ) | <periodo>
<periodo> ::= "[" <data> "," <data> "]"
<data> ::= <ano> "-" <mes> "-" <dia>
<ano> ::= 4<numeral>
<mes> ::= 2<numeral>
<dia> ::= 2<numeral>

*-----
* Caracteres admitidos
*-----

<aceitos-lex> ::= <normal> | <reservado> | <detalhaID> | <outro>
<normal> ::= <minuscula> | <numeral> | "."
<normalID> ::= <normal> | <detalhaID>
<minuscula> ::= "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" |
               "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" |
               "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"
<numeral> ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
<reservado> ::= ":" | "@" | ";" | "," | "[" | "]" | "!" | "~"
<detalhaID> ::= "_" | "-"
<outro> ::= "=" | "$"
```

Anexo B – Gramática da URN de Referência

```
*-----
* Estrutura de um nome uniforme (URN) do espaço "lex"
*-----
<URN> ::= "urn:" <NID> ":" <NSS-lex>
<NID> ::= "lex"
*-----
* Estrutura de um nome uniforme do espaço "lex" internacionalizado
*-----
<NSS-lex> ::= <local> ":" <NSS-lex-local>
<local> ::= <codigo-pais> [ <detalhamento-local> ]
*-----
* Estrutura do elemento <local> para o Brasil
*-----
<codigo-pais> ::= "br"
<detalhamento-local> ::=
    ( [ ";" <unidade-federacao-ou-uf> [ ";" <municipio> ] ] )
  | ( [ ";" <unidade-federacao-ou-uf> ] [ ";" <local-judiciario> ] )
<detalhe-ramo> ::= ( ( ("regiao." | "zona.") 1* <numeral> )
  | ( "secao.judiciaria" | "comarca" )
  | ( <unidade-federacao-ou-uf> *[ "," <unidade-federacao-ou-uf> ] )
  | ( <municipio> *[ "," <municipio> ] ) )
<unidade-federacao-ou-uf> ::= <unidade-federacao> | 2<minuscula>
<unidade-federacao> ::= 1*(<minuscula> | ".")
<municipio> ::= 1*(<minuscula> | ".")
<local-judiciario> ::= <ramo-justica> *[ ";" <detalhe-ramo> ]
<ramo-justica> ::= "justica."
    ("federal" | "trabalho" | "eleitoral" | "militar" | "estadual")
*-----
* Estrutura de um nome específico do "lex-br"
*-----
<NSS-lex-local> ::= <documentoComplexo>
    | ( <documentoIndividual> [ "!" <fragmento> ] )
<documentoIndividual> ::= <documentoComplexo> [ "@" <versao> ] [ "~" <forma> ]
```

```
<documentoComplexo> ::= <documento>
*-----
* Estrutura do elemento <documento>
*-----
<documento> ::= <autoridade> ":" <tipo-documento> ":" <descriptor>
*-----
* Estrutura do elemento <autoridade>
*-----
<autoridade> ::= <sujeito> *["," <sujeito> ]
<sujeito> ::= ( <instituicao> *[";" <orgao> ] [ ";" <funcao> ] )
| <cargo>
<instituicao> ::= 1*<normal>
<orgao> ::= 1*<normal>
<funcao> ::= 1*<normal>
<cargo> ::= 1*<normal>
*-----
* Estrutura do elemento <tipo-documento>
*-----
<tipo-documento> ::= ( ( <tipo-norma>
| <tipo-jurisprudencia>
| <tipo-projeto-norma>
) [ ";" <nome-subtipo-sequenciador> ] )
| ( "publicacao.oficial;" <nome-periodico-oficial>
[";" <nome-secao-periodico-oficial> ]
[";" <detalhe-extra-suplemento> ] )
<tipo-projeto-norma> ::= 1*<normal>
<tipo-norma> ::= 1*<normal>
<tipo-jurisprudencia> ::= 1*<normal>
<nome-subtipo-sequenciador> ::= 1*<normal>
<nome-periodico-oficial> ::= 1*<normal>
<nome-secao-periodico-oficial> ::= 1*<normal>
<detalhe-extra-suplemento> ::= "edicao.extra" | "suplemento"
*-----
* Estrutura do elemento <descriptor>
*-----
```



```
<descriptor> ::= ( ( <datas> [ ";" <identificadores> ] )
                    | ( <datasOuAno> ";" <identificadores> )
                    | ( [ <datasOuAno> ";" ] <apelido-documento> )
                    )
                *[";" <id-componente> ["," <titulo-componente> ] ]
                [ ";"retificacao." <seq-retificacao> ]

<identificadores> ::= ( <id-documento> *[";" <id-documento> ] )
                    | ( <numero-lex> )
                    | ( <numero-seq> )

<apelido-documento> ::= 1*<normal>
<id-documento>       ::= 1*<normalID>
<numero-lex>        ::= "lex-" 1*<numeral>
<numero-seq>        ::= "seq-" <sigla-orgao> "-" 1*<numeral>
<sigla-orgao>       ::= 1*<normal>
<id-componente>     ::= 1*<normal>
<titulo-componente> ::= 1*<normal>
<seq-retificacao>   ::= 1*<numeral>
<datasOuAno>        ::= <datas> | <ano>

*-----
* Estrutura do elemento <fragmento>
*-----
<fragmento> ::= ( <id-particao> | <intervalo-ids> ) *[";" <fragmento> ]
<intervalo-ids> ::= "[" <id-particao> "," <id-particao> "]"
<id-particao> ::= 1*<normalID>

*-----
* Estrutura do elemento <versao>
*-----
<versao> ::= [ ( <datas>
                | "versao.original"
                | "inicio.vigencia"
                | "multivigente"
                | "versao.vigente.em;" <data>
                | "versao.eficaz.em;" <data>
                | "versao.consultada.em;" <data> )
              [ ";" <evento> [ ";" <visao> ] ] ]
```

```
<evento> ::= 1*<normal>
<visao> ::= <datas>
*-----
* Estrutura do elemento <forma>
*-----
<forma> ::= <tipo-forma> [ ";" <lingua> ] *[ "," <lingua> ] ]
<tipo-forma> ::= 1*<normal>
<lingua> ::= <lingua-codigo>
                [ "-" <lingua-script> ] [ "-" <lingua-regiao> ]
<lingua-codigo> ::= 1*<minuscula>
<lingua-script> ::= 1*<minuscula>
<lingua-regiao> ::= 1*<minuscula>
*-----
* Estrutura do elemento <datas>
*-----
<datas> ::= ( <data> *["," <data> ] ) | <periodo>
<periodo> ::= "[" <data> "," <data> "]"
<data> ::= <ano> "-" <mes> "-" <dia>
<ano> ::= 4<numeral>
<mes> ::= 2<numeral>
<dia> ::= 2<numeral>
*-----
* Caracteres admitidos
*-----
<aceitos-lex> ::= <normal> | <reservado> | <detalhaID> | <outro>
<normal> ::= <minuscula> | <numeral> | "."
<normalID> ::= <normal> | <detalhaID>
<minuscula> ::= "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" |
                "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" |
                "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"
<numeral> ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
<reservado> ::= ":" | "@" | ";" | "," | "[" | "]" | "!" | "~"
<detalhaID> ::= "_" | "-"
<outro> ::= "=" | "$"
```



LexML Brasil
Parte 3 – LexML XML Schema

Versão 1.0 (RC1)
Brasília – Dezembro de 2008

LexML Brasil**Parte 3 – LexML XML Schema**

Versão 1.0 (Dezembro/2008)

Parte 3 – LexML XML Schema	2
1. Introdução.....	3
2. Esquemas Rígido e Flexível para Texto Articulado.....	3
3. Namespace.....	4
4. Nome e Conteúdo de Elementos e Atributos	4
4.1. <i>Encoding</i> e UTF-8.....	5
4.2. Cuidados na Codificação de Alguns Caracteres.....	5
5. Visão Geral do Esquema LexML	6
5.1. LexML - Parte Inicial	6
5.2. LexML - Parte Principal.....	6
5.3. Identificadores de Agrupamento de Artigo	7
5.4. Identificadores de Artigos e Dispositivos de Artigo	8
5.5. Outros Identificadores	9
6. Estrutura de Acórdão em LexML.....	9
7. Tratamento do Texto	10
7.1. Fórmulas Matemáticas.....	11
8. Metadados.....	11
9. Referências	12
9.1. Bibliográficas	12
9.2. Sites	12
10. Lista de Abreviaturas e Siglas	12
11. Créditos.....	13
11.1. Editores.....	13
11.2. Grupo de Trabalho LexML (em Dezembro de 2008).....	13
Anexo 1 – Esquema lexml-base.xsd.....	14
Anexo 2 - Esquema lexml-br-rigido.xsd	33
Anexo 3 - Esquema lexml-flexivel.xsd	37

1. Introdução

Este documento apresenta princípios e conceitos utilizados na definição do esquema XML para a estruturação dos textos de normas, julgados e projetos de normas.

O esquema do Projeto LexML Brasil é baseado nos esquemas dos Projetos NIR (*Norme in Rete*) e AKOMA NTOSO, ambos editados pelo Prof. Dr. Fabio Vitali, da Universidade de Bolonha. Serão enfatizados nas próximas seções aspectos referentes ao ordenamento jurídico brasileiro e à técnica legislativa federal. Em algumas seções, haverá adaptação e/ou tradução de trechos do documento de referência do projeto AKOMA NTOSO.

Na época da implantação do projeto NIR, a linguagem *XML Schema* ainda estava em processo de definição. Atualmente, o esquema do Projeto NIR está disponível tanto na notação DTD (*Document Type Definition*) quanto em *XML Schema*. O Projeto LexML Brasil utiliza apenas a linguagem *XML Schema* para modelagem dos documentos textuais por essa linguagem possuir mais recursos de validação, quando comparado com a DTD.

A leitura deste documento requer o conhecimento dos padrões *XML* (W3C 2006) e de *XML Schema* (W3C 2004).

2. Esquemas Rígido e Flexível para Texto Articulado

Devido à falta de padronização da técnica legislativa em relação às três esferas e à existência de normas federais com articulação incompatível com a Lei Complementar nº 95/1998 (principalmente normas anteriores a 1998), foram criados dois esquemas, a exemplo da estratégia do Projeto NIR, conforme abaixo:

- Esquema Rígido, de acordo com a Lei Complementar nº 95/1998 – Normas e projetos de normas que seguem as regras de articulação definidas pela Lei Complementar nº 95/1998.
- Esquema Flexível – Permite mais combinações na articulação dos dispositivos. Aplica-se às normas e proposições legislativas que não seguem a Lei Complementar nº 95/1998 além de servir como um denominador comum entre os esquemas rígidos definidos para cada técnica legislativa.

As definições comuns fazem parte do Esquema Base que é incluído nos Esquemas Rígido e Flexível.

Todo documento válido no Esquema Rígido é também válido no Esquema Flexível. No entanto, o inverso não é verdadeiro.

Estados e municípios que não utilizam as regras de estruturação conforme a Lei Complementar nº 95/1998, devem definir seu próprio Esquema Rígido. Por exemplo, será necessário definir um Esquema Rígido de acordo com a técnica legislativa do Estado de São Paulo, definida pela Lei Complementar Estadual nº 863/1999. As

instâncias de documentos XML válidas no Esquema Rígido da LCP-SP 863/1999 também serão válidas no Esquema Flexível do LexML.

Normas federais anteriores a 1998 que não consigam ser validadas no Esquema Rígido LCP 95/1998 devem utilizar o Esquema Flexível.

3. Namespace

As instâncias de documentos LexML são completamente qualificadas, isto é, utiliza-se o *namespace* “http://www.lexml.gov.br/1.0” como *namespace default* para todos os elementos.

Apesar de alguns elementos utilizarem o mesmo nome de elementos HTML, para simplificar, decidiu-se incorporar alguns elementos do *namespace* HTML ao *namespace* LexML. Como resultado, é possível especificar o *namespace LexML* como *namespace default* minimizando o uso de prefixos no documento da instância, ao mesmo tempo em que se mantém a qualificação completa dos elementos.

O LexML utiliza três atributos que devem ser codificados com o prefixo do *namespace* explícito, conforme a seguir:

- “xml:lang” e “xml:base” do *namespace* <http://www.w3.org/XML/1998/namespace>, e
- “xlink:href” do *namespace* <http://www.w3.org/1999/xlink>.

O atributo *xml:lang* deve ser utilizado no elemento *inline* para delimitar expressões escritas em outras línguas, tais como: latim (“la”), inglês (“en”), espanhol (“es”) e alemão (“de”).

O atributo *xml:base* deve ser utilizado no elemento <Alteracao> ou <RemissaoMultipla> para identificar a URN da norma que está sendo alterada ou referenciada. Todos os atributos *xlink:href* que fazem parte dos elementos contidos nos elementos <Alteracao> e <RemissaoMultipla> são relativos à URN indicada no atributo *xml:base*.

4. Nome e Conteúdo de Elementos e Atributos

Uma das principais características de um documento XML é a sua capacidade de se auto-descrever, isto é, cada segmento do conteúdo textual é delimitado por marcações (*tags*) que possuem nomes significativos.

Todos os elementos do LexML (exceto os “importados” de outros vocabulários – ex. HTML) utilizam a notação *UpperCamelCase*. Os atributos utilizam a notação *lowerCamelCase*, isto é, iniciam com letra minúscula e cada nova palavra é iniciada por letra maiúscula.

Ao se definir uma nova linguagem na notação XML é comum surgirem dúvidas quanto à escolha entre o construtor elemento e atributo em uma determinada situação. Utilizou-se a seguinte diretriz no LexML: o conteúdo textual dos elementos deve ser

suficiente para reproduzir o texto constante do documento oficial (em papel ou arquivo digitalmente assinado) na ordem em que ele se apresenta; os atributos codificam informações de controle do conteúdo texto, tais como identificadores, referências, indicação da língua etc. No caso de acórdãos, onde, em alguns casos, há repetição do cabeçalho, decidiu-se codificar estas informações em um elemento <CabecalhoAcordao> separado dos outros componentes textuais do acórdão (relatório, voto etc) com o objetivo de minimizar a redundância de informações.

Não são utilizados diacríticos nem a letra “ç” nos nomes de elementos, atributos ou no conteúdo de atributos “id”. Por serem construtores voltados principalmente para o processamento técnico do documento, faz parte da prática da informática omitir estes sinais que nem sempre estão disponíveis nos teclados.

4.1. **Encoding e UTF-8**

As instâncias de documentos do LexML utilizam o *encoding* “UTF-8”. Esta é a codificação *default* para o XML e foi escolhida porque oferece o armazenamento em um único *byte* para os caracteres mais comuns e permite a utilização dos caracteres definidos no padrão Unicode. A codificação ISO-8859-1 (muito difundida no Brasil) sempre utiliza um único byte por caractere, e, por isso, é bem mais limitada que a UTF-8. Por exemplo, a codificação ISO-8859-1 não possui as aspas angulares: “ (abre aspas) e ” (fecha aspas). Seria necessário utilizar entidades ou referências numéricas para codificá-las em um documento XML caso fosse utilizada a codificação ISO-8859-1.

Outra vantagem da UTF-8 é a possibilidade de armazenamento de documentos textuais de outros países que utilizam conjuntos de caracteres não latinos.

4.2. **Cuidados na Codificação de Alguns Caracteres**

É muito comum encontrar na Internet textos de normas codificados com caracteres errados. Um dos casos mais comuns é a codificação errada do Símbolo Ordinal “º” (código 186), utilizado na identificação de normas e dispositivos, como por exemplo: “Lei nº 8.112 ...” ou “§ 2º do art. 6º”. Este símbolo algumas vezes é codificado como “^o” (letra “o” minúscula sobrescrita) ou como “°” (Símbolo Grau – código 176).

A palavra abreviada “números” deve ser codificada como “n^{os}” (letra “n”; símbolo de ordinal (código 186); letra “s” sem marcação de estilo sobrescrito ou sublinhado).

De acordo com o texto da Lei Complementar nº 95/1998, os incisos são codificados em números romanos e separados do texto por hífen “-” (código 45). Neste caso não devem ser utilizados os símbolos *ene* “ – ” (código 150) ou *eme* “ — ” (código 151).

A codificação errada de caracteres pode gerar vários problemas, como, por exemplo, erro na pronúncia por parte dos softwares de síntese de voz utilizados por

deficientes visuais ou pessoas com baixa visão ou, ainda, erro na utilização de software de recuperação da informação ou de verificação ortográfica.

5. Visão Geral do Esquema LexML

A Parte Inicial apresenta construtores (grupos, tipos etc.) que são utilizados pela Parte Principal. Como o XML Schema do LexML foi derivado do XML Schema do Projeto AKOMA NTOSO, foram preservados a forma de organização e alguns comentários originais. Alguns construtores da parte inicial estão definidos com nome em inglês. No entanto, esses nomes são utilizados apenas internamente no esquema e não são visíveis nas instâncias de documentos LexML.

5.1. LexML - Parte Inicial

A parte inicial do esquema relaciona alguns grupos de elementos (`<xsd:groups>`) e atributos (`<xsd:attributeGroup>`) utilizados na definição dos modelos de conteúdos e tipos em todo o documento. Na seqüência, são definidos tipos simples (`<xsd:simpleType>`), basicamente para enumeração de valores *string*, e tipos complexos (`<xsd:complexType>`). Esses tipos complexos correspondem a *Design Patterns* de modelos de conteúdo referenciados na definição dos elementos conforme (Vitali et. al, 2005) e (Vitali, 2007).

Hierarchy – elementos de estrutura hierárquica (ex: `<Livro>`, `<Parte>`, etc.).

Block – seqüência de elementos bloco (ex: `<p>`).

Inline – trata elementos de conteúdo misto (ex: conteúdo de `<p>`).

Marker – elemento de conteúdo vazio (ex: `<NotaReferenciada/>`).

O *Design Pattern Container* não possui um tipo complexo associado, a exemplo dos outros. Cada elemento dessa categoria organiza os elementos na ordem desejada, utilizando os elementos construtores da linguagem XML Schema.

5.2. LexML - Parte Principal

Todos os documentos LexML compartilham o mesmo elemento raiz `<LexML>`, cujo conteúdo inicial é obrigatoriamente o elemento `<Metadado>` seguido do elemento que identifica o tipo de documento. A utilização de um único elemento raiz segue o *Design Pattern* “*Universal Root*”.

O LexML prevê inicialmente os seguintes tipos de documentos:

Norma – Constituição, Emendas, Leis, Decretos etc.

ProjetoNorma – Proposições legislativas.

Jurisprudencia – Súmulas e Acórdãos.

DocumentoGenerico – Para outros tipos de documentos.

Anexo – Anexos dos documentos acima.

A Norma e o Projeto de Norma utilizam o tipo `HierarchicalStructure`; os Anexos podem ser definidos utilizando tanto uma estrutura hierárquica quanto uma estrutura genérica, dependendo do caso específico. A Jurisprudência utiliza a estrutura genérica para cada um dos seus componentes textuais (Relatório, Voto, etc).

Na seção seguinte do esquema, são definidos *containers* referenciados na definição dos tipos de documento. Por exemplo, é possível encontrar nesta seção os elementos `ParteInicial`, `Articulacao` e `ParteFinal`, referenciados na definição do elemento `Norma`.

Os elementos que definem a hierarquia de um documento articulado, de acordo com a Lei Complementar nº 95/1998, são apresentados no arquivo “lexml-br-rigido.xsd”. Inicialmente são apresentados os elementos que agrupam artigos (`Parte`, `Livro`, `Titulo`, `Capitulo`, `Secao` e `Subsecao`), em seguida a unidade básica de articulação (`Artigo`), e, por fim, os dispositivos de artigo (`Caput`, `Inciso`, `Alinea`, `Item`, `Paragrafo`). Os elementos correspondentes, no esquema flexível, são apresentados no arquivo “lexml-flexivel.xsd”. Os esquemas rígido e flexível fazem inclusão do esquema base, localizado no arquivo “lexml-base.xsd”.

Na seqüência, são apresentados os elementos genéricos correspondentes a cada um dos cinco *patterns* de modelo de conteúdo: `Agrupamento` (*Container*), `AgrupamentoHierarquico` (*Hierarchy*), `Bloco` (*Block*), `EmLinha` (*Inline*), `Marca` (*Marker*). Ao utilizar estes elementos genéricos é necessário especificar o atributo “nome” significativo.

Em seguida, o esquema apresenta os elementos “importados” do vocabulário HTML que tratam da codificação do texto corrido, de listas, de tabelas e de outros elementos de formatação tais como negrito, itálico, sublinhado, subscrito e sobrescrito. Trata-se de uma simplificação do vocabulário HTML.

Por fim, são apresentados os elementos de metadados que tratam de vários aspectos, tais como identificação, contexto, indexação, ciclo de vida e informações sobre a publicação.

5.3. Identificadores de Agrupamento de Artigo

A definição de identificadores para os dispositivos tem por objetivo identificar partes de um documento. No esquema LexML, os identificadores são permitidos em todos os elementos, e, em alguns, devido a sua importância, são atributos obrigatórios. Eles podem ser utilizados, por exemplo, nas remissões textuais de um dispositivo ou na indicação de qual dispositivo está sendo alterado.

Os identificadores de elementos que agrupam artigos são criados a partir da concatenação do identificador de cada nível hierárquico desde o nível mais alto até o nível que se está identificando. Por exemplo, `id="tit1_cap2-1_sec3"` é o identificador da “Seção III do Capítulo II-A do Título I”. Note que os algarismos romanos e as letras são convertidos em números arábicos. A letra do seqüencial de inclusão também é convertida para o número arábico. No caso do elemento `<Parte>`,

normalmente identificado por um nome (Parte Geral, Parte Especial), utiliza-se um seqüencial a partir do número 1.

A Tabela 2 apresenta os identificadores de agrupamento de artigo previstos na Lei Complementar nº 95/1998 bem como o elemento genérico <AgrupamentoHierarquico> que poderá ser utilizado nos casos não previstos. Por exemplo, o novo Código Civil utiliza o elemento “Subtítulo” não previsto na Lei Complementar nº 95/1998.

Em algumas situações, será necessário utilizar o Esquema Flexível que permite mais combinações dos elementos que agrupam artigos.

Tabela 2. Identificadores para agrupamento de artigos.

Elemento	id	Exemplo	id do exemplo
<Parte>	prtN	Parte Geral do Código Civil	prt1
<Livro>	livN	Livro I	liv1
<Titulo>	titN	Título II	tit2
<Capitulo>	capN	Título IX, Capítulo IV-A	tit9_cap4-1
<Secao>	secN	Seção III do Capítulo I	cap1_sec3
<Subsecao>	subN	Parte Geral do Código Civil, Livro I, Título VI, Capítulo I, Seção II, Subseção I	prt2_liv1_tit6_cap1_sec2_sub1
<AgrupamentoHierarquico>	aghN	Subtítulo I do Título II do Livro II da Parte Especial do Código Civil	prt2_liv2_tit2_agh1

5.4. Identificadores de Artigos e Dispositivos de Artigo

A lógica de formação dos identificadores de artigos e seus dispositivos é similar à dos elementos que agrupam artigos, atentando-se ao fato da presença do elemento <Caput> que não possui o seqüencial numérico por ele ser único e obrigatório.

A Tabela 3 apresenta os identificadores de artigo e seus dispositivos previstos na Lei Complementar nº 95/1998 bem como o elemento genérico <DispositivoGenerico> que poderá ser utilizado nos casos de dispositivos não previstos ou em desacordo com os da lei complementar. Em algumas situações será necessário utilizar o Esquema Flexível que permite mais combinações entre os dispositivos de artigos. A tabela mostra também exemplos em que as letras das alíneas são convertidas em números arábicos.

Tabela 3. Identificadores de Artigo e Dispositivos de Artigo.

Elemento	Id	Exemplo(s)	id do exemplo
<Artigo>	artN	Art. 5º	art5
<Caput>	cpt	caput do art. 5º	art5_cpt
<Paragrafo>	parN	Art. 8-A, parágrafo único	art8-1_par1
<Inciso>	incN	inciso III do art 8º inciso II do § 2º do art 20	art8_cpt_inc3 art20_par2_inc2
<Alínea>	aliN	Alínea “a” do inciso IX do parágrafo único do art. 3º-A-A.	art3-1-1_par1_inc9_ali1
<Item>	iteN	item 1, da alínea “b”, do inciso I, do art. 39	art39_cpt_inc1_ali2_ite1
<DispositivoGenerico>	dpgN	alínea “b” do § 3º do art. 5º	art5_par3_dpg2

Na seção de definição dos tipos simples do esquema, foram definidos *patterns* com expressões regulares para validação dos identificadores.

5.5. Outros Identificadores

Os identificadores servem de “gancho” para a localização de determinado partição do documento. No caso de normas jurídicas, foram definidos alguns identificadores para rápida localização de outros segmentos de texto, tais como: epígrafe, ementa, preambulo, localdata e fecho.

Os identificadores de dispositivos alvo de uma alteração (normalmente o novo texto destes dispositivos aparece entre aspas) são precedidos pelo prefixo *CONT_aspN_*, onde *CONT* (*contexto*) representa o id do dispositivo da alteração na norma publicada e *N* representa o seqüencial das aspas referente à alteração. Por exemplo, o identificador *art14_cpt_asp1_art4_cpt_inc3* identifica o elemento com a alteração do Inciso III do art. 4º da norma identificada no elemento <Alteracao>. Este elemento está localizado na primeira seqüência de aspas do *caput* do art 14 da norma alteradora.

6. Estrutura de Acórdão em LexML

O inteiro teor de um Acórdão é formado por seis elementos conforme a seguir:

- <CabecalhoAcordao> - contém dados estruturados comuns como epígrafe, data de julgamento, órgão julgador e partes.
- <EmentaTexto> - texto livre da ementa.
- <AcordaoTexto> - texto livre com o acórdão.
- <RelatorioTexto> - texto livre com o relatório.
- <VotoTexto> - texto livre com o voto.
- <DebateTexto> - texto livre opcional com debates.

- `<ExtratoAtaTexto>` - texto livre com o Extrato da Ata.

A estrutura para texto livre do LexML permite a codificação de texto com sofisticadas características, tais como definição de links persistentes para normas (de qualquer esfera) e julgados.

As notas de rodapé são tratadas em cada componentes de texto do acórdão utilizando o elemento `<div>` e links bidirecionais. Dessa forma é possível isolar este elemento ao mesmo tempo em que é permitida a navegação entre a nota e o texto principal.

O elemento “*class*” pode ser codificado em cada parágrafo para tratar as questões de apresentação em uma folha de estilo CSS.

7. Tratamento do Texto

Com o intuito de facilitar o desenvolvimento de ferramentas para redação de documentos no formato LexML, definimos a utilização de um subconjunto de elementos HTML para a estruturação da parte textual do documento.

Os elementos importados desempenham o mesmo papel daquele definido para o HTML. Acrescenta-se a esta regra apenas uma exceção: o elemento `<div>`, que no HTML desempenha o papel de bloco genérico, no LexML desempenha o papel de *container* genérico, pois o elemento `<p>` já desempenha o papel de bloco genérico.

Os elementos `<div>`, `<p>` e `` são considerados elementos genéricos de *container*, *bloco* e *inline*, respectivamente, sendo equivalentes a `<Agrupador>`, `<Bloco>` e `<EmLinha>`.

Dentro de um elemento `<p>` é possível utilizar o elemento *inline* `<Remissao>` para criar referências *href* para outros documentos do LexML.

As referências poderão ser absolutas ou relativas. Este último caso ocorre quando o elemento `<Remissao>` está codificado internamente ao elemento `<Alteracao>` ou `<RemissaoMultipla>`: no elemento remissão será codificada a referência relativa para o *id* do dispositivo apontado (`href="art1_par1"`), enquanto que no elemento `<Alteracao>` haverá a referência absoluta para a norma (`xml:base="urn:lex:br:sao.paulo:lei:2004-02-11;123:"`).

Para otimizar a marcação de referências, nos casos em que muitos dispositivos de uma mesma norma são referenciados, é possível utilizar o elemento `<RemissaoMultipla>` que conterà mais de um elemento `<Remissao>`. O primeiro utiliza o atributo `xml:base`, e, o segundo, atributo `xlink:href` com referências relativas.

Todos os elementos do esquema LexML (inclusive os “importados”) podem ser marcados com atributos *class* e *style*, permitindo a associação de estilos CSS com definições precisas de apresentação.

7.1. Fórmulas Matemáticas

Para incluir fórmulas matemáticas deve-se utilizar o elemento inline `<Formula>` que inclui o elemento `<math>` do vocabulário MathML (namespace <http://www.w3.org/1998/Math/MathML>). A fórmula poderá ser codificada em um arquivo externo ou no próprio documento LexML.

8. Metadados

O elemento Metadado contém informações sobre o documento para atender a diversas necessidades tais como recuperação da informação, preservação e gerência do ciclo de vida do documento.

Como regra, todo conteúdo feito por um editor (e não pelo autor) deve fazer parte desta seção. Vice versa, todo conteúdo do autor não deve ser codificado na seção de metadados.

A Seção `<Metadados>` está estruturada em oito subseções, conforme a seguir:

- Identificação – contém elementos de metadados que identificam unicamente o documento corrente.
- Contexto – identifica a posição do arquivo corrente na hierarquia FRBR. A posição é indicada pelo atributo `'id="self"'` do elemento `FRBRItem`.
- CicloDeVida – registra os eventos relacionados ao arquivo corrente. No caso de textos multivigentes, deve-se registrar o histórico de eventos do período relativo à vigência do documento.
- EventosGerados – registra os eventos de alteração que afetam outros documentos. Por exemplo, a alteração e revogação de dispositivos de outras normas.
- Notas – registra notas do editor ou do responsável pelo *markup*. As notas podem ser visíveis, nos casos em que registre informações sobre a versão corrente, ou invisíveis, nos casos em que faz-se um registro técnico sobre o *markup*.
- Recursos – lista alguns recursos auxiliares que podem referenciados em outras pontos da seção de metadados.
- MetadadosProprietario – ponto de extensão para registro de metadados não considerados no atual esquema.

Para o detalhamento dos campos da Seção Identificação, deve-se consultar o documento URN (LexML – Parte 2) e para o entendimento da hierarquia FRBR no contexto do Projeto LexML, deve-se consultar o documento Modelo de Referência (LexML – Parte 1).

9. Referências

9.1. Bibliográficas

VITALI, F; Di IORIO, A.; GUBELLINI, D.; Design Patterns for Descriptives Document Substructures. *In Extreme Markup Language 2005*. Montreal, Québec. 2005. Acessado em 03 de Janeiro de 2007. Disponível em: [www.mulberrytech.com/Extreme/Proceedings/html/2005/Vitali01/EML2005Vitali01.html]

VITALI, F. *Akoma Ntoso 1.0 Release Notes*. (versão 23/11/2006). 2006. Acessado em 3 de fevereiro de 2007. Disponível em: [http://www.akomantoso.org/AKOMA_NTOSO/downloads-1/Release_Notes_20112006.pdf]

W3C-2004. XML Schema Part 0: Primer Second Edition. W3C recommendation. Disponível em : <http://www.w3.org/TR/xmlschema-0/>. Acessado em 01 de dezembro de 2008.

W3C-2006. Extensible Markup Language (XML) 1.0: W3C recommendation. Disponível em: <http://www.w3.org/TR/2006/REC-xml-20060816/>. Acesso em 01 de dezembro de 2008.

9.2. Sites

<http://www.nir.it/>

Projeto Norme in Rete (Itália)

<http://www.akomantoso.org/>

Projeto Akoma Ntoso (Nações Unidas/África)

<http://www.xmlpatterns.com/>

Design Patterns para XML

<http://www.w3c.org>

Recomendações do W3C

10. Lista de Abreviaturas e Siglas

CSS – Cascade Styling Sheets

DTD – Document Type Definition

FRBR – Functional Requirements for Bibliographical Records

HTML – Hypertext Markup Language

NIR – Norme in Rete

XML – Extensible Markup Language

W3C – World Wide Web Consortium

11. Créditos

11.1. Editores

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

Fernando Ciciliati (Senado Federal / Interlegis)

11.2. Grupo de Trabalho LexML (em Dezembro de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Cláudio Mori (Tribunal Superior Eleitoral)

Claudson dos Santos Melo (Tribunal Superior do Trabalho)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda Oliveira de Macedo (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)

Jonatas Rodrigues (Procuradoria Geral da República)

Luís Henrique Raja G. Mitchell (Tribunal de Contas da União)

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

Paulo de Tarso Penna da Costa (Superior Tribunal de Justiça)

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-base.xsd

```

<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema targetNamespace="http://www.lexml.gov.br/1.0"
  xmlns="http://www.lexml.gov.br/1.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:mathml="http://www.w3.org/1998/Math/MathML"
  elementFormDefault="qualified"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:import namespace="http://www.w3.org/1999/xlink" schemaLocation="Xlink.xsd" />
  <xsd:import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xsd:import namespace="http://www.w3.org/1998/Math/MathML" schemaLocation="mathml2/mathml2.xsd"/>

  <xsd:annotation>
    <xsd:documentation>
      =====
      O esquema do projeto LexML Brasil é uma adaptação dos schemas
      Akoma Ntoso (1.0) (http://www.akomantoso.org/)
      e
      Norme in Rete (2.0) (http://www.nir.it)
      para a técnica legislativa brasileira. Esses esquemas foram
      originalmente criados por Fabio Vitali ( Universidade of Bolonha )
      com a supervisão (Direito) de Monica Palmirani ( Universidade de Bolonha ).

      Adaptado por João Alberto de Oliveira Lima (Senado Federal - joalima@senado.gov.br)
      e revisado por Fernando Ciciliati (Interlegis - ciciliati@interlegis.gov.br)

      O presente esquema usa XML design patterns de forma sistemática, tal como
      descrito em http://www.xmlpatterns.com/
      Alguns deles são mencionados explicitamente, quando o seu uso
      não é suficientemente claro ou natural. Outros são usados sem nenhuma
      referência.

      Versão 1.0 - dezembro/08 - exclusão dos elementos Texto e TextoSimples

      Versão 0.7 - agosto/08 - ajuste no patterns de ids de dispositivos para trocar o complemento alfa para num
      - inclusão do elemento inline Formula com conteúdo MathML.

      Versão 0.6 - abril/08 - alteração nos ids das subdivisões de artigo retirando o 'alt' e incluindo o
      prefixo do dispositivo pai.
      - inclusão do elemento Pena após o Texto dos dispositivos de artigo.
      - alteração de nome da tag "Justificativa" para "Justificacao"
      - inclusão do elemento EmConformidadeCom
      - inclusão de elemento Apelido no nível ObraIndividual
      e ObraIndividualComplementar

      =====
    </xsd:documentation>
  </xsd:annotation>

  <xsd:annotation>
    <xsd:documentation>
      =====
    </xsd:documentation>
  </xsd:annotation>

  Principais Grupos de Elementos
  - LexML (Hierarchy, Inline, Marker)
  - HTML (Container, Block, Inline, Marker)
  - LexML + HTML + generic elements

  =====
    </xsd:documentation>
  </xsd:annotation>

  <xsd:group name="LXhier">
    <xsd:choice>
      <xsd:element ref="Parte"/>
      <xsd:element ref="Livro"/>
      <xsd:element ref="Titulo"/>
      <xsd:element ref="Capitulo"/>
    </xsd:choice>
  </xsd:group>
    
```



```

        <xsd:element ref="Secao"/>
        <xsd:element ref="Artigo"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="LXhierCompleto">
    <xsd:choice>
        <xsd:group ref="LXhier"/>
        <xsd:element ref="Subsecao"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="LXcontainers">
    <xsd:choice>
        <xsd:element ref="Caput"/>
        <xsd:element ref="Paragrafo"/>
        <xsd:element ref="Inciso"/>
        <xsd:element ref="Alinea"/>
        <xsd:element ref="Item"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="LXinline">
    <xsd:choice>
        <xsd:element ref="Remissao"/>
        <xsd:element ref="RemissaoMultipla"/>
        <xsd:element ref="Alteracao"/>
        <xsd:element ref="Formula"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="LXmarker">
    <xsd:choice>
        <xsd:element ref="NotaReferenciada"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="HTMLcontainers">
    <xsd:sequence>
        <xsd:element ref="div"/>
    </xsd:sequence>
</xsd:group>

<xsd:group name="HTMLblock">
    <xsd:choice>
        <xsd:element ref="p"/>
        <xsd:element ref="ul"/>
        <xsd:element ref="ol"/>
        <xsd:element ref="table"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="HTMLinline">
    <xsd:choice>
        <xsd:element ref="span"/>
        <xsd:element ref="b"/>
        <xsd:element ref="i"/>
        <xsd:element ref="a"/>
        <xsd:element ref="sub"/>
        <xsd:element ref="sup"/>
        <xsd:element ref="ins"/>
        <xsd:element ref="del"/>
        <xsd:element ref="dfn"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="HTMLmarker">
    <xsd:sequence>
        <xsd:element ref="img"/>
    </xsd:sequence>
</xsd:group>

<xsd:group name="hierElements">
    <xsd:choice>

```

```

        <xsd:group ref="LXhier" />
        <xsd:element ref="AgrupamentoHierarquico"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="hierElementsCompleto">
    <xsd:choice>
        <xsd:group ref="LXhierCompleto" />
        <xsd:element ref="AgrupamentoHierarquico"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="containerElements">
    <xsd:choice>
        <xsd:group ref="HTMLcontainers" />
        <xsd:element ref="Agrupamento"/>
        <xsd:element ref="AspasEstrutura"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="blockElements">
    <xsd:choice>
        <xsd:group ref="HTMLblock" />
        <xsd:element ref="ConteudoExterno"/>
        <xsd:element ref="Bloco"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="inlineElements">
    <xsd:choice>
        <xsd:group ref="LXinline" />
        <xsd:group ref="HTMLinline" />
        <xsd:element ref="EmLinha"/>
    </xsd:choice>
</xsd:group>

<xsd:group name="markerElements">
    <xsd:choice>
        <xsd:group ref="LXmarker" />
        <xsd:group ref="HTMLmarker" />
        <xsd:element ref="Marcador"/>
    </xsd:choice>
</xsd:group>

<xsd:annotation>
    <xsd:documentation>
        =====
        Grupos de Atributos
        =====
    </xsd:documentation>
</xsd:annotation>

<xsd:attributeGroup name="nome">
    <xsd:attribute name="nome" type="xsd:string" use="required"/>
</xsd:attributeGroup>

<xsd:attributeGroup name="source">
    <xsd:attribute name="fonte" use="required" type="xsd:anyURI"/>
</xsd:attributeGroup>

<xsd:attributeGroup name="date">
    <xsd:attribute name="data" type="xsd:date" use="required"/>
</xsd:attributeGroup>

<xsd:attributeGroup name="link">
    <xsd:attribute ref="xlink:href" use="required"/>
</xsd:attributeGroup>

<xsd:attributeGroup name="linkID">
    <xsd:attribute name="nota" type="xsd:IDREF" use="required"/>
</xsd:attributeGroup>

<xsd:attributeGroup name="linkURN">

```

```

        <xsd:attribute name="URN" type="xsd:anyURI" use="required"/>
        <xsd:attribute name="showAs" type="xsd:string"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="linkopt">
        <xsd:attribute ref="xlink:href"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="optvalue">
        <xsd:attribute name="value" type="xsd:string"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="period">
        <xsd:attribute name="eventoInicial" type="xsd:anyURI"/>
        <xsd:attribute name="eventoFinal" type="xsd:anyURI"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="enactment">
        <xsd:attributeGroup ref="period"/>
        <xsd:attribute name="situacao" type="TipoSituacao"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="cellattrs">
        <xsd:attribute name="rowspan" type="xsd:integer" default="1" />
        <xsd:attribute name="colspan" type="xsd:integer" default="1" />
    </xsd:attributeGroup>

    <xsd:attributeGroup name="HTMLattrs">
        <xsd:attribute name="class" type="xsd:string"/>
        <xsd:attribute name="style" type="xsd:string"/>
        <xsd:attribute name="title" type="xsd:string"/>
        <xsd:attribute ref="xml:lang" default="pt-BR"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="HTMLname">
        <xsd:attribute name="name" type="xsd:string"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="idreq">
        <xsd:attribute name="id" use="required" type="xsd:ID"/>
        <xsd:attribute name="seqOrdem" type="xsd:int"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="idreqArt">
        <xsd:attribute name="id" use="required" type="idArtigo"/>
        <xsd:attribute name="seqOrdem" type="xsd:int"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="idopt">
        <xsd:attribute name="id" type="xsd:ID"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="corereq">
        <xsd:attributeGroup ref="HTMLattrs"/>
        <xsd:attributeGroup ref="enactment"/>
        <xsd:attributeGroup ref="idreq"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="coreopt">
        <xsd:attributeGroup ref="HTMLattrs"/>
        <xsd:attributeGroup ref="enactment"/>
        <xsd:attributeGroup ref="idopt"/>
    </xsd:attributeGroup>

    <xsd:attributeGroup name="corereqArt">
        <xsd:attributeGroup ref="HTMLattrs"/>
        <xsd:attributeGroup ref="enactment"/>
        <xsd:attributeGroup ref="idreqArt"/>
    </xsd:attributeGroup>

    <xsd:annotation>
        <xsd:documentation>
            =====

```

```

=====
</xsd:documentation>
</xsd:annotation>

<xsd:simpleType name="TipoSituacao">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="omissis" />
    <xsd:enumeration value="revogado" />
    <xsd:enumeration value="suspensao" />
    <xsd:enumeration value="vetado"/>
    <xsd:enumeration value="superado" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:annotation>
  <xsd:documentation>
    =====

    Tipos Complexos

    =====
  </xsd:documentation>
</xsd:annotation>

<xsd:complexType name="hierarchy" >
  <xsd:sequence>
    <xsd:element ref="Rotulo" minOccurs="0" maxOccurs="1" />
    <xsd:element ref="NomeAgrupador" minOccurs="0" maxOccurs="1" />
    <xsd:sequence>
      <xsd:element ref="AgrupamentoHierarquico" minOccurs="0"
maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:sequence>
  <xsd:attributeGroup ref="corereq"/>
</xsd:complexType>

<xsd:complexType name="blocksreq" >
  <xsd:sequence minOccurs="1" maxOccurs="unbounded" >
    <xsd:group ref="blockElements" />
  </xsd:sequence>
  <xsd:attributeGroup ref="corereq"/>
</xsd:complexType>

<xsd:complexType name="blocksopt">
  <xsd:sequence minOccurs="1" maxOccurs="unbounded" >
    <xsd:group ref="blockElements" />
  </xsd:sequence>
  <xsd:attributeGroup ref="coreopt"/>
</xsd:complexType>

<xsd:complexType name="inline" mixed="true">
  <xsd:choice minOccurs="0" maxOccurs="unbounded" >
    <xsd:group ref="inlineElements" />
    <xsd:group ref="markerElements" />
  </xsd:choice>
  <xsd:attributeGroup ref="coreopt"/>
</xsd:complexType>

<xsd:complexType name="markerreq" >
  <xsd:attributeGroup ref="corereq"/>
</xsd:complexType>

<xsd:complexType name="markeropt" >
  <xsd:attributeGroup ref="coreopt"/>
</xsd:complexType>

<xsd:complexType name="anyOther" >
  <xsd:choice>
    <xsd:any namespace="##other"/>
  </xsd:choice>
</xsd:complexType>

```

```

<xsd:complexType name="participaType" >
  <xsd:sequence>
    <xsd:element ref="Rotulo" minOccurs="1" maxOccurs="1" />
    <xsd:element ref="Tratamento" minOccurs="0" maxOccurs="1" />
    <xsd:element ref="NomeAgente" minOccurs="1" maxOccurs="1" />
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="textoType">
  <xsd:sequence>
    <xsd:element ref="p" minOccurs="0" maxOccurs="unbounded" />
  </xsd:sequence>
  <xsd:attributeGroup ref="corereq"/>
</xsd:complexType>

<xsd:complexType name="textoSimplesType">
  <xsd:sequence>
    <xsd:element ref="p" minOccurs="0" maxOccurs="1" />
  </xsd:sequence>
  <xsd:attributeGroup ref="corereq"/>
</xsd:complexType>

  <xsd:complexType name="textoSimplesOptType">
    <xsd:sequence>
      <xsd:element ref="p" minOccurs="0" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attributeGroup ref="coreopt"/>
  </xsd:complexType>

<xsd:annotation>
  <xsd:documentation>
    =====
    Elemento Raiz
    Comentário Um elemento genérico raiz contendo todos os tipos de documentos
    Pattern: Universal Root
    Referência: http://www.xmlpatterns.com/UniversalRootMain.shtml
    =====
  </xsd:documentation>
</xsd:annotation>

<xsd:complexType name="DocumentTypes" >
  <xsd:sequence >
    <xsd:element ref="Metadado" />
    <xsd:choice >
      <xsd:element ref="Norma" />
      <xsd:element ref="ProjetoNorma" />
      <xsd:element ref="Jurisprudencia" />
      <xsd:element ref="DocumentoGenerico" />
      <xsd:element ref="Anexo" />
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="LexML" type="DocumentTypes" />

<xsd:annotation>
  <xsd:documentation>
    =====
    Tipos de Documento
    Comentário: Todos os tipos de documentos compartilham dois modelos de conteúdo (Hierárquico, Aberto)
    Pattern: Consistent Element Set
    Referência: http://www.xmlpatterns.com/ConsistentElementSetMain.shtml
    =====
  </xsd:documentation>
</xsd:annotation>

<xsd:complexType name="HierarchicalStructure" >
  <xsd:sequence >
    <xsd:element ref="ParteInicial" minOccurs="0" maxOccurs="1" />
    <xsd:element ref="Articulacao" />
    <xsd:element ref="ParteFinal" minOccurs="0" maxOccurs="1" />
  </xsd:sequence>

```

```

        <xsd:element ref="Anexos" minOccurs="0" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:complexType name="OpenStructure" >
    <xsd:sequence >
        <xsd:element ref="PartePrincipal" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="Anexos" minOccurs="0" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attributeGroup ref="coreopt"/>
</xsd:complexType>

    <xsd:element name="Norma" type="HierarchicalStructure"/>

<xsd:element name="ProjetoNorma">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="Norma"/>
            <xsd:element name="Justificacao" type="OpenStructure" minOccurs="0"
maxOccurs="unbounded" />
            <xsd:element name="AutorProjeto" type="xsd:string" minOccurs="0"
maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="Jurisprudencia">
    <xsd:complexType>
        <xsd:choice>
            <xsd:element ref="Sumula"/>
            <xsd:element ref="Acordao"/>
        </xsd:choice>
    </xsd:complexType>
</xsd:element>

<xsd:element name="DocumentoGenerico" type="OpenStructure" />

    <xsd:element name="Anexo">
        <xsd:complexType>
            <xsd:choice>
                <xsd:element name="DocumentoArticulado" type="HierarchicalStructure"/>
                <xsd:element ref="DocumentoGenerico"/>
            </xsd:choice>
        </xsd:complexType>
    </xsd:element>

<xsd:annotation>
    <xsd:documentation>
        =====
        LX common containers elements
        =====
    </xsd:documentation>
</xsd:annotation>

    <!-- Articulação (Normas e Projetos de Normas) -->

<xsd:element name="ParteInicial" >
    <xsd:complexType >
        <xsd:sequence>
            <xsd:element ref="Epigrafe" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="Ementa" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="Preambulo" minOccurs="0" maxOccurs="1" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="ParteFinal" >
    <xsd:complexType >
        <xsd:sequence>

```

```

        <xsd:element ref="LocalDataFecho" minOccurs="0" maxOccurs="1" />
        <xsd:choice>
            <xsd:element ref="AssinaturaGrupo" minOccurs="0"
maxOccurs="unbounded" />
            <xsd:element ref="Assinaturas" minOccurs="0"
maxOccurs="unbounded"/>
            <xsd:element ref="Assinatura" minOccurs="0" maxOccurs="unbounded"/>
        </xsd:choice>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>

<xsd:element name="Epigrafe" type="textoSimpleType"/>
<xsd:element name="Ementa" type="textoSimpleType"/>
<xsd:element name="LocalDataFecho" type="textoSimpleType"/>
<xsd:element name="Observacao" type="textoSimpleType"/>
<xsd:element name="Preambulo" type="textoType"/>
<xsd:element name="AssinaturaGrupo">
    <xsd:complexType >
        <xsd:sequence>
            <xsd:element name="NomeGrupo" type="xsd:string" minOccurs="0"
maxOccurs="1"/>
            <xsd:choice>
                <xsd:element ref="Assinaturas" minOccurs="0"
maxOccurs="unbounded"/>
                <xsd:element ref="Assinatura" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:choice>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="Assinaturas">
    <xsd:complexType >
        <xsd:sequence >
            <xsd:element ref="Assinatura" minOccurs="1" maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="Assinatura">
    <xsd:complexType >
        <xsd:sequence >
            <xsd:element ref="NomePessoa" minOccurs="1" maxOccurs="unbounded" />
            <xsd:element ref="Cargo" minOccurs="0" maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="NomePessoa" type="xsd:string"/>
<xsd:element name="Cargo" type="xsd:string"/>

    <xsd:element name="ReferenciaAnexo" type="refURN" />

    <xsd:element name="Anexos">
        <xsd:complexType >
            <xsd:sequence >
                <xsd:element ref="ReferenciaAnexo" minOccurs="1" maxOccurs="unbounded" />
            </xsd:sequence>
            <xsd:attributeGroup ref="coreopt"/>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="Articulacao">
        <xsd:complexType >
            <xsd:sequence minOccurs="1" maxOccurs="unbounded" >
                <xsd:group ref="hierElements" />
            </xsd:sequence>
            <xsd:attributeGroup ref="coreopt"/>
        </xsd:complexType>
    </xsd:element>

```

```

</xsd:element>

<!-- Jurisprudencia ( elementos especificos ) -->

<xsd:element name="Tratamento" type="xsd:string"/>

<xsd:element name="NomeAgente" type="xsd:string"/>

<xsd:element name="Sumula">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="Epigrafe" minOccurs="1" maxOccurs="1"/>
      <xsd:element ref="Ementa" minOccurs="1" maxOccurs="1"/>
      <xsd:element ref="Observacao" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="coreopt"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Acordao">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="CabecalhoAcordao" minOccurs="1" maxOccurs="1"/>
      <xsd:element name="EmentaTexto" type="OpenStructure" minOccurs="1"
maxOccurs="1"/>
      <xsd:element name="AcordaoTexto" type="OpenStructure" minOccurs="1"
maxOccurs="1"/>
      <xsd:element name="RelatorioTexto" type="OpenStructure" minOccurs="1"
maxOccurs="1"/>
      <xsd:element name="VotoTexto" type="OpenStructure" minOccurs="1"
maxOccurs="1"/>
      <xsd:element name="DebateTexto" type="OpenStructure" minOccurs="0"
maxOccurs="1"/>
      <xsd:element name="ExtratoAtaTexto" type="OpenStructure" minOccurs="1"
maxOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="CabecalhoAcordao">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="Epigrafe" minOccurs="1" maxOccurs="1"/>
      <xsd:element name="DataJulgamento" type="xsd:date" minOccurs="1"
maxOccurs="1"/>
      <xsd:element name="OrgaoJulgador" type="xsd:string" minOccurs="1"
maxOccurs="1"/>
      <xsd:element ref="Relator" minOccurs="1" maxOccurs="1"/>
      <xsd:element ref="PartesProcesso" minOccurs="2" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Relator" type="participaType"/>

<xsd:element name="PartesProcesso">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="ParteProcesso" type="participaType" minOccurs="1"
maxOccurs="unbounded"/>
      <xsd:element name="AdvogadosParte" type="participaType" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<!-- Open Structure -->

<xsd:element name="PartePrincipal">
  <xsd:complexType>
    <xsd:choice minOccurs="1" maxOccurs="unbounded">
      <xsd:element ref="AgrupamentoHierarquico" />
      <xsd:group ref="containerElements"/>
      <xsd:group ref="blockElements" />
    </xsd:choice>
  </xsd:complexType>
</xsd:element>
    
```



```

        <xsd:attributeGroup ref="coreopt"/>
    </xsd:complexType>
</xsd:element>

<xsd:element name="DispositivoGenerico">
    <xsd:complexType>
        <xsd:complexContent>
            <xsd:extension base="DispositivoType">
                <xsd:attributeGroup ref="nome"/>
            </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>

<xsd:element name="Pena">
<xsd:complexType>
    <xsd:sequence >
        <xsd:element ref="Rotulo" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="p" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="DispositivoGenerico" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="coreopt"/>
    <xsd:attributeGroup ref="linkopt"/>
</xsd:complexType>
</xsd:element>

<xsd:element name="TituloArtigo" type="xsd:string" />

<xsd:element name="Rotulo" type="xsd:string" />

<xsd:element name="NomeAgrupador" type="inline" />

<xsd:element name="Texto" type="blocksopt"/>

<xsd:annotation>
    <xsd:documentation>
        =====
        LX elementos inline
        =====
    </xsd:documentation>
</xsd:annotation>

<xsd:element name="Remissao">
    <xsd:complexType mixed="true">
        <xsd:complexContent>
            <xsd:extension base="inline">
                <xsd:attributeGroup ref="link"/>
            </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>

<xsd:element name="RemissaoMultipla">
    <xsd:complexType mixed="true">
        <xsd:complexContent>
            <xsd:extension base="inline">
                <xsd:attribute ref="xml:base" use="required"/>
            </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>

<xsd:element name="NotaReferenciada">
    <xsd:complexType>
        <xsd:complexContent>
            <xsd:extension base="markeropt">
                <xsd:attributeGroup ref="linkID"/>
            </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>

```

```

        <xsd:element name="Alteracao">
            <xsd:complexType mixed="true">
                <xsd:complexContent>
                    <xsd:extension base="inline">
                        <xsd:sequence>
                            <xsd:choice >
                                <xsd:element ref="AspasTexto" minOccurs="1"
maxOccurs="unbounded" />
                                <xsd:element ref="AspasEstrutura" minOccurs="1"
maxOccurs="unbounded" />
                            </xsd:choice>
                        </xsd:sequence>
                        <xsd:attribute ref="xml:base"/>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>

        <xsd:element name="NotaAlteracao" type="inline"/>

        <xsd:element name="AspasTexto" type="inline" /> <!-- não se aplica à LCP 95 -->

        <xsd:element name="AspasEstrutura">
            <xsd:complexType >
                <xsd:sequence>
                    <xsd:element ref="AbreAspas"/>
                    <xsd:choice minOccurs="1" maxOccurs="unbounded" >
                        <xsd:group ref="hierElementsCompleto" />
                        <xsd:group ref="blockElements" />
                        <xsd:group ref="containerElements" />
                        <xsd:group ref="LXcontainers"/>
                        <xsd:element ref="Omissis"/>
                    </xsd:choice>
                    <xsd:element ref="FechaAspas"/>
                    <xsd:element ref="NotaAlteracao" minOccurs="0" maxOccurs="1"/>
                </xsd:sequence>
                <xsd:attributeGroup ref="corereq"/>
                <xsd:attributeGroup ref="linkopt"/>
            </xsd:complexType>
        </xsd:element>

        <xsd:element name="AbreAspas" fixed="&#8220;/>
        <xsd:element name="FechaAspas" fixed="&#8221;/>

        <xsd:element name="Omissis">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element ref="Rotulo" minOccurs="0" maxOccurs="1" />
                    <xsd:element ref="p" minOccurs="0" maxOccurs="1" />
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>

        <xsd:element name="Formula">
            <xsd:complexType>
                <xsd:sequence minOccurs="0" maxOccurs="1">
                    <xsd:element ref="mathml:math" />
                </xsd:sequence>
                <xsd:attribute name="fonte" type="xsd:string"/>
                <xsd:attribute name="tipo" type="xsd:string"/>
            </xsd:complexType>
        </xsd:element>
    
```

```

        <xsd:annotation>
            <xsd:documentation>
                =====
    
```

Elementos Genéricos

Comentário: Cada elementos deste esquema corresponde a um dos cinco modelos de conteúdo: Hierarchy, Container, Block, Inline e Mark. Além destes elementos, o esquema provê um elemento genérico para cada um dos modelos. Estes elementos genéricos podem ser utilizados para atender necessidades específicas ou situações não previstas no modelo original. O atributo nome do elemento genérico será uma indicação do seu significado.

Pattern: Generic Document + Role Attribute
 Referência: <http://www.xmlpatterns.com/GenericDocumentMain.shtml> +
<http://www.xmlpatterns.com/RoleAttributeMain.shtml>

```

=====
</xsd:documentation>
</xsd:annotation>

<xsd:element name="AgrupamentoHierarquico">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="hierarchy">
        <xsd:sequence>
          <xsd:group ref="LXhierCompleto" minOccurs="1"
maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:attributeGroup ref="nome"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Agrupamento">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="blocksreq">
        <xsd:attributeGroup ref="nome"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Bloco">
  <xsd:complexType mixed="true">
    <xsd:complexContent>
      <xsd:extension base="inline">
        <xsd:attributeGroup ref="nome"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="EmLinha">
  <xsd:complexType mixed="true">
    <xsd:complexContent>
      <xsd:extension base="inline">
        <xsd:attributeGroup ref="nome"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Marcador">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="markerreq">
        <xsd:attributeGroup ref="nome"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="ConteudoExterno" type="anyOther" />

</xsd:annotation>
<xsd:documentation>
=====

```

HTML elements

Comment: Many elements are drawn directly from HTML 4.0

Pattern: Reuse Document Types (partial)

Reference: <http://www.xmlpatterns.com/ReuseDocumentTypesMain.shtml>

=====

```

        </xsd:documentation>
    </xsd:annotation>

    <xsd:element name="div" type="blocksreq" />

    <xsd:element name="p" type="inline" />

    <xsd:element name="li">
        <xsd:complexType mixed="true">
            <xsd:complexContent>
                <xsd:extension base="inline">
                    <xsd:choice minOccurs="0" maxOccurs="unbounded" >
                        <xsd:element ref="ul" />
                        <xsd:element ref="ol" />
                        <xsd:element ref="p"/>
                    </xsd:choice>
                    <xsd:attributeGroup ref="optvalue"/>
                </xsd:extension>
            </xsd:complexContent>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="span" type="inline"/>

    <xsd:element name="b" type="inline" />

    <xsd:element name="i" type="inline" />

    <xsd:element name="sub" type="inline" />

    <xsd:element name="sup" type="inline" />

    <xsd:element name="ins" type="inline" />

    <xsd:element name="del" type="inline" />

    <xsd:element name="dfn" type="inline" />

    <xsd:element name="a">
        <xsd:complexType mixed="true">
            <xsd:complexContent>
                <xsd:extension base="inline">
                    <xsd:attributeGroup ref="link"/>
                    <xsd:attributeGroup ref="HTMLname"/>
                    <xsd:attribute name="target" type="xsd:string"/>
                </xsd:extension>
            </xsd:complexContent>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="img">
        <xsd:complexType>
            <xsd:complexContent>
                <xsd:extension base="markeropt">
                    <xsd:attribute name="src" type="xsd:anyURI" use="required"/>
                    <xsd:attribute name="alt" type="xsd:string"/>
                    <xsd:attribute name="width" type="xsd:integer"/>
                    <xsd:attribute name="height" type="xsd:integer"/>
                </xsd:extension>
            </xsd:complexContent>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="ul">
        <xsd:complexType >
            <xsd:sequence >
                <xsd:element ref="li" minOccurs="1" maxOccurs="unbounded" />
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="ol">
        <xsd:complexType >
            <xsd:sequence >
                <xsd:element ref="li" minOccurs="1" maxOccurs="unbounded" />
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

```

```

        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="table">
    <xsd:complexType >
        <xsd:sequence >
            <xsd:element ref="tr" minOccurs="1" maxOccurs="unbounded" />
        </xsd:sequence>
        <xsd:attributeGroup ref="HTMLAttrs"/>
        <xsd:attribute name="width" type="xsd:integer"/>
        <xsd:attribute name="border" type="xsd:integer"/>
        <xsd:attribute name="cellspacing" type="xsd:integer"/>
        <xsd:attribute name="cellpadding" type="xsd:integer"/>
        <xsd:attributeGroup ref="idreq"/>
    </xsd:complexType>
</xsd:element>

<xsd:element name="tr">
    <xsd:complexType >
        <xsd:choice minOccurs="1" maxOccurs="unbounded" >
            <xsd:element ref="th" />
            <xsd:element ref="td" />
        </xsd:choice>
        <xsd:attributeGroup ref="HTMLAttrs"/>
    </xsd:complexType>
</xsd:element>

<xsd:element name="th">
    <xsd:complexType mixed="true">
        <xsd:complexContent>
            <xsd:extension base="inline">
                <xsd:attributeGroup ref="cellattrs"/>
            </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>

<xsd:element name="td">
    <xsd:complexType mixed="true">
        <xsd:complexContent>
            <xsd:extension base="inline">
                <xsd:attributeGroup ref="cellattrs"/>
            </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>

<xsd:annotation>
    <xsd:documentation>
        =====
        Metadados
        =====
    </xsd:documentation>
</xsd:annotation>

<!-- Algumas definições de tipos preliminares -->

<xsd:simpleType name="MetadadoLiteralTexto">
    <xsd:restriction base="xsd:string">
        <xsd:minLength value="1"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="MetadadoLiteralTextoURN">
    <xsd:simpleContent>
        <xsd:extension base="MetadadoLiteralTexto">
            <xsd:attribute name="URN" type="xsd:anyURI"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="MetadadoData">

```

```

        <xsd:restriction base="xsd:date">
            <xsd:minInclusive value="1500-01-01"/>
            <xsd:maxInclusive value="2500-01-01"/>
        </xsd:restriction>
    </xsd:simpleType>

    <xsd:complexType name="MetadadoLingua">
        <xsd:attribute ref="xlink:href"/>
    </xsd:complexType>

    <xsd:complexType name="MetadadoRefTipo">
        <xsd:attribute ref="xlink:href"/>
    </xsd:complexType>

    <xsd:complexType name="MetadadoRefNome">
        <xsd:attribute ref="xlink:href"/>
    </xsd:complexType>

    <xsd:complexType name="MetaSection" >
        <xsd:sequence >
            <xsd:element ref="Identificacao" />
            <xsd:element ref="Contexto" />
            <xsd:element ref="CicloDeVida" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="EventosGerados" minOccurs="0" maxOccurs="1"/>
            <xsd:element ref="Notas" minOccurs="0" maxOccurs="unbounded" />
            <xsd:element ref="Recursos" minOccurs="0" maxOccurs="1"/>
            <xsd:element ref="MetadadoProprietario" minOccurs="0" maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>

    <xsd:element name="Metadado" type="MetaSection" />

    <xsd:complexType name="IdentificacaoType" >
        <xsd:sequence >
            <xsd:element ref="JurisdicaoAutoridade" minOccurs="1" maxOccurs="1" />
            <xsd:element ref="Autoridade" minOccurs="1" maxOccurs="1" />
            <xsd:element ref="TipoDocumento" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="IdentificadorDocumento" minOccurs="1" maxOccurs="1" />
            <xsd:element ref="DataVersao" minOccurs="1" maxOccurs="unbounded" />
            <xsd:element ref="DataVersaoFinal" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="DataVisao" minOccurs="1" maxOccurs="unbounded" />
            <xsd:element ref="DataVisaoFinal" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="FormaConteudo" minOccurs="1" maxOccurs="1" />
        </xsd:sequence>
        <xsd:attribute ref="xml:base"/>
        <xsd:attribute name="FRBRitemRef" type="xsd:IDREF" use="required"/>
    </xsd:complexType>

    <xsd:element name="Identificacao" type="IdentificacaoType"/>

    <xsd:element name="JurisdicaoAutoridade" type="MetadadoRefNome"/>
    <xsd:element name="Autoridade" type="MetadadoRefNome"/>
    <xsd:element name="TipoDocumento" type="MetadadoRefTipo"/>
    <xsd:element name="IdentificadorDocumento" type="IdentificadorDocumentoType"/>
    <xsd:element name="DataVersao" type="MetadadoData"/>
    <xsd:element name="DataVersaoFinal" type="MetadadoData"/>
    <xsd:element name="DataVisao" type="MetadadoData"/>
    <xsd:element name="DataVisaoFinal" type="MetadadoData"/>
    <xsd:element name="FormaConteudo" type="FormaConteudoType"/>

    <xsd:complexType name="IdentificadorDocumentoType">
        <xsd:sequence>
            <xsd:element ref="DataRepresentativa" minOccurs="1" maxOccurs="1"/>
            <xsd:element ref="Identificador" minOccurs="1" maxOccurs="1"/>
            <xsd:element ref="IdentificadorComponente" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="FormaConteudoType">
        <xsd:sequence>
            <xsd:element ref="TipoConteudo" minOccurs="1" maxOccurs="1"/>
            <xsd:element ref="Lingua" minOccurs="1" maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
    
```

```

        </xsd:sequence>
    </xsd:complexType>

    <xsd:element name="DataRepresentativa" type="MetadadoData"/>
    <xsd:element name="Identificador" type="MetadadoLiteralTexto"/>
    <xsd:element name="IdentificadorComponente" type="MetadadoLiteralTexto"/>

    <xsd:element name="TipoConteudo" type="MetadadoRefTipo"/>
    <xsd:element name="Lingua" type="MetadadoLingua"/>

    <xsd:element name="Contexto">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="FRBRObraComplexa" minOccurs="1" maxOccurs="unbounded"
/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="FRBRObraComplexa">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="Apelido" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="EquivalenteA" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="SucessorLogicoDe" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="FRBRObraIndividual" minOccurs="1" maxOccurs="1"/>
                <xsd:element ref="Indexacao" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:sequence>
            <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="FRBRObraIndividual">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="Apelido" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="PublicacaoOficial" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="EmConformidadeCom" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="FRBRItem" minOccurs="1" maxOccurs="unbounded"/>
                <xsd:element ref="FRBRObraIndividualComplementar" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="Indexacao" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:sequence>
            <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="FRBRItem">
        <xsd:complexType>
            <xsd:attributeGroup ref="link"/></xsd:attributeGroup>
            <xsd:attribute name="mimetype" type="xsd:string"/>
            <xsd:attributeGroup ref="idopt"/>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="FRBRObraIndividualComplementar">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="Apelido" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="EmConformidadeCom" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="FRBRItem" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="FRBRObraIndividualComplementar" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:sequence>
            <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="Apelido">
        <xsd:complexType>
            <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
        </xsd:complexType>
    </xsd:element>

```

```

<xsd:element name="PublicacaoOficial">
  <xsd:complexType>
    <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
  </xsd:complexType>
</xsd:element>

<xsd:element name="EquivalenteA">
  <xsd:complexType>
    <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
  </xsd:complexType>
</xsd:element>

<xsd:element name="SucessorLogicoDe">
  <xsd:complexType>
    <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
  </xsd:complexType>
</xsd:element>

<xsd:element name="EmConformidadeCom">
  <xsd:complexType>
    <xsd:attributeGroup ref="linkURN"/></xsd:attributeGroup>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Indexacao" type="IndexacaoType"/>

  <xsd:complexType name="IndexacaoType">
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="vocabulario" type="xsd:IDREF"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

<xsd:element name="CicloDeVida">
  <xsd:complexType >
    <xsd:sequence>
      <xsd:element ref="Evento" minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="EventosGerados">
  <xsd:complexType >
    <xsd:sequence>
      <xsd:element ref="Evento" minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

  <xsd:element name="Evento">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="markerreq">
          <xsd:choice minOccurs="1" maxOccurs="unbounded">
            <xsd:element ref="Criacao"/>
            <xsd:element ref="Publicacao"/>
            <xsd:element ref="EntradaEmVigor"/>
            <xsd:element ref="Retificacao"/>
            <xsd:element ref="Republicacao"/>
            <xsd:element ref="RevogacaoTotal"/>
            <xsd:element ref="AnulamentoTotal"/>
            <xsd:element ref="AlteracaoFragmento"/>
            <!-- AlteraçãoFragmento inclui modificação (Substituicao Textual),
Revogacao Parcial,
Anulamento Parcial e Derrubada de Veto Parcial -->
          </xsd:choice>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="refURN" >
    <xsd:attributeGroup ref="coreopt"/>

```



```

        <xsd:attribute name="AlvoURN" type="xsd:anyURI"/>
        <xsd:attribute name="FonteURN" type="xsd:anyURI"/>
    </xsd:complexType>

    <xsd:complexType name="refURNSimples" >
        <xsd:attributeGroup ref="coreopt"/>
        <xsd:attribute name="URN" type="xsd:anyURI"/>
    </xsd:complexType>

    <xsd:element name="Publicacao" type="refURN" />

    <xsd:element name="EntradaEmVigor" type="refURN"/>

    <xsd:element name="Republicacao" type="refURN" />

    <xsd:element name="Retificacao" type="refURN" />

    <xsd:element name="ProjetoNormaOrigem" type="refURN" />

    <xsd:element name="JulgadoOrigemAnulacao" type="refURN" />

    <xsd:element name="NormaFonte" type="refURN"/>

    <xsd:element name="MensagemVetoAplicado" type="refURN" />

    <xsd:element name="RevogacaoTotal" type="refURN"/>

    <xsd:element name="AnulamentoTotal" type="refURN"/>

    <xsd:element name="Criacao">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="ProjetoNormaOrigem" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="JulgadoOrigemAnulacao" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="MensagemVetoAplicado" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:complexType name="refURNAlvo" >
        <xsd:attributeGroup ref="coreopt"/>
        <xsd:attribute name="FonteURN" type="xsd:anyURI"/>
        <xsd:attribute name="AlvoURN"/>
        <xsd:attribute name="AlvoLocal" type="xsd:IDREFS"/>
        <xsd:attribute name="FonteLocal" type="xsd:IDREFS"/>
    </xsd:complexType>

    <xsd:complexType name="URNtype" >
        <xsd:attributeGroup ref="coreopt"/>
        <xsd:attribute name="URN" type="xsd:anyURI"/>
    </xsd:complexType>

    <xsd:element name="AlteracaoFragmento">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="ModificacaoTextual" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="RevogacaoParcial" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="AnulamentoParcial" minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="DerrubadaVetoParcial" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="ModificacaoTextual" type="refURNAlvo"/>
    <xsd:element name="RevogacaoParcial" type="refURNAlvo"/>
    <xsd:element name="AnulamentoParcial" type="refURNAlvo"/>
    <xsd:element name="DerrubadaVetoParcial" type="refURNAlvo"/>

    <xsd:element name="Notas">
        <xsd:complexType >
            <xsd:sequence minOccurs="1" maxOccurs="unbounded" >
                <xsd:element ref="Nota" />
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

```

```

</xsd:element>

<xsd:element name="Nota">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="textoType">
        <xsd:attribute name="exporta" type="xsd:boolean"/>
        <xsd:attribute name="dataInclusao" type="xsd:date"/>
        <xsd:attribute name="autor" type="xsd:IDREF"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Recursos">
  <xsd:complexType >
    <xsd:sequence minOccurs="1" maxOccurs="unbounded" >
      <xsd:element ref="Recurso" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Recurso" type="markerreq" />

<xsd:element name="MetadadoProprietario">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="xsd:anyType">
        <xsd:attributeGroup ref="source"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:annotation>
  <xsd:documentation>
    =====
    End of Schema
    =====
  </xsd:documentation>
</xsd:annotation>

</xsd:schema>
    
```

Anexo 2 - Esquema lexml-br-rigido.xsd

```

<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema targetNamespace="http://www.lexml.gov.br/1.0"
  xmlns="http://www.lexml.gov.br/1.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  elementFormDefault="qualified"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:import namespace="http://www.w3.org/1999/xlink" schemaLocation="Xlink.xsd" />
  <xsd:import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xsd:include schemaLocation="lexml-base.xsd"/>

  <xsd:simpleType name="idArtigo">
    <xsd:annotation>
      <xsd:documentation>
        A regras abaixo tratam dos IDs de dispositivos no nível de artigo para baixo.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:ID">
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?art\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_(inc\dpg)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_(inc\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_(ali\dpg)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_(inc\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_(ali\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_(ite\dpg)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="idAgregador">
    <xsd:annotation>
      <xsd:documentation>
        A regras abaixo tratam dos IDs de dispositivos no nível de artigo para cima.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:ID">
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?(prt|lgh)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?(prt|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(liv|lgh)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?(prt|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(liv|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(tit|lgh)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?(prt|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(liv|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(tit|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(cap|lgh)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?(prt|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(liv|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(tit|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(cap|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(sec|lgh)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
      <xsd:pattern value="(art\d+((-[0-9]{1,3})?)\{0,3\}_cpt_(par\dpg)\d+((-[0-9]{1,3})?)\{0,3\})_asp\d+_)?(prt|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(liv|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(tit|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(cap|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(sec|lgh)\d+((-[0-9]{1,3})?)\{0,3\})_(sub|lgh)\d+((-[0-9]{1,3})?)\{0,3\}"></xsd:pattern>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:annotation>
    <xsd:documentation>
      =====
      Elements for hierarchical documents
      =====
    </xsd:documentation>
  </xsd:annotation>

```



```

maxOccurs="unbounded" />
maxOccurs="unbounded" />
<xsd:element ref="Artigo" minOccurs="0"
<xsd:element ref="Subsecao" minOccurs="0"
</xsd:sequence>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="Subsecao">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="hierarchy">
<xsd:sequence>
<xsd:element ref="Artigo" minOccurs="0"
maxOccurs="unbounded" />
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:complexType name="ArticleType" >
<xsd:sequence >
<xsd:element ref="TituloArtigo" minOccurs="0" maxOccurs="1" />
<xsd:element ref="Rotulo" minOccurs="1" maxOccurs="1" />
<xsd:choice>
<xsd:sequence >
<xsd:element ref="Caput" minOccurs="1" maxOccurs="1" />
<xsd:element ref="Paragrafo" minOccurs="0" maxOccurs="unbounded" />
</xsd:sequence>
<xsd:element ref="DispositivoGenerico" minOccurs="0" maxOccurs="unbounded"/>
</xsd:choice>
</xsd:sequence>
<xsd:attributeGroup ref="corereqArt"/>
<xsd:attributeGroup ref="linkopt"/>
</xsd:complexType>
<xsd:element name="Artigo" type="ArticleType" />
<xsd:complexType name="DispositivoType" >
<xsd:sequence >
<xsd:element ref="Rotulo" minOccurs="0" maxOccurs="1" />
<xsd:element ref="p" minOccurs="0" maxOccurs="unbounded"/>
<xsd:element ref="Alteracao" minOccurs="0" maxOccurs="1"/>
<xsd:element ref="DispositivoGenerico" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attributeGroup ref="corereqArt"/>
<xsd:attributeGroup ref="linkopt"/>
</xsd:complexType>
<xsd:element name="Caput">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="DispositivoType">
<xsd:sequence>
<xsd:element ref="Inciso" minOccurs="0"
maxOccurs="unbounded" />
<xsd:element ref="Pena" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="Inciso">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="DispositivoType">
<xsd:sequence>
<xsd:element ref="Alinea" minOccurs="0"
maxOccurs="unbounded" />
<xsd:element ref="Pena" minOccurs="0" maxOccurs="1"/>

```

```

        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Alinea">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="DispositivoType">
        <xsd:sequence>
          <xsd:element ref="Item" minOccurs="0"
maxOccurs="unbounded" />
          <xsd:element ref="Pena" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Paragrafo">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="DispositivoType">
        <xsd:sequence>
          <xsd:element ref="Inciso" minOccurs="0"
maxOccurs="unbounded" />
          <xsd:element ref="Pena" minOccurs="0" maxOccurs="1"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Item" type="DispositivoType"/>

<xsd:annotation>
  <xsd:documentation>
    =====
    End of Schema
    =====
  </xsd:documentation>
</xsd:annotation>
</xsd:schema>

```



```

=====
</xsd:documentation>
</xsd:annotation>

<xsd:element name="Parte">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="hierarchy">
        <xsd:sequence minOccurs="0" maxOccurs="unbounded" >
          <xsd:group ref="LXhierCompleto" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Livro">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="hierarchy">
        <xsd:sequence>
          <xsd:group ref="LXhierCompleto" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Titulo">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="hierarchy">
        <xsd:sequence>
          <xsd:group ref="LXhierCompleto" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Capitulo">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="hierarchy">
        <xsd:sequence>
          <xsd:group ref="LXhierCompleto" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Secao">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="hierarchy">
        <xsd:sequence>
          <xsd:group ref="LXhierCompleto" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Subsecao">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="hierarchy">
        <xsd:group ref="LXhierCompleto" minOccurs="0"

```



```

maxOccurs="unbounded"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
</xsd:element>

<xsd:complexType name="ArticleType" >
    <xsd:sequence >
        <xsd:element ref="TituloArtigo" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="Rotulo" minOccurs="1" maxOccurs="1" />
        <xsd:choice>
            <xsd:sequence >
                <xsd:group ref="LXcontainers" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:sequence>
            <xsd:element ref="DispositivoGenerico" minOccurs="0" maxOccurs="unbounded"/>
        </xsd:choice>
        <xsd:attributeGroup ref="corereqArt"/>
        <xsd:attributeGroup ref="linkopt"/>
    </xsd:complexType>

<xsd:element name="Artigo" type="ArticleType" />

<xsd:complexType name="DispositivoType" >
    <xsd:sequence >
        <xsd:element ref="Rotulo" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="p" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="Alteracao" minOccurs="0" maxOccurs="1"/>
        <xsd:group ref="LXcontainers" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="DispositivoGenerico" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="Pena" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="corereqArt"/>
    <xsd:attributeGroup ref="linkopt"/>
</xsd:complexType>

<xsd:element name="Caput" type="DispositivoType"/>

<xsd:element name="Inciso" type="DispositivoType"/>

<xsd:element name="Alinea" type="DispositivoType"/>

<xsd:element name="Paragrafo" type="DispositivoType"/>

<xsd:element name="Item" type="DispositivoType"/>

<xsd:annotation>
    <xsd:documentation>
        =====
        End of Schema
        =====
    </xsd:documentation>
</xsd:annotation>

</xsd:schema>
    
```



LexML Brasil
Parte 4 – Coleta de Metadados

Versão 1.0 (RC1)
Brasília – Dezembro de 2008

LexML Brasil**Parte 4 – Coleta de Metadados**

Versão 1.0 (Dezembro/2008)

Parte 4 – Coleta de Metadados	2
1. Introdução.....	3
2. Protocolo OAI-PMH	3
3. Protocolo OAI-PMH aplicado ao Projeto LexML	4
3.1. Provedor de Dados - Papéis.....	4
3.2. Cabeçalho do Registro.....	4
3.3. Corpo do Registro de Metadado.....	5
3.4. Corpo do Registro de Metadados do Publicador Oficial	6
3.5. Tipos de Relacionamentos.....	6
4. Sobre o Esquema lexml_oai.xsd.....	7
5. Sobre o Esquema provedor_profile_lexml.xsd	7
6. Referências	8
6.1. Sites	8
7. Lista de Abreviaturas e Siglas	8
8. Créditos.....	8
8.1. Editores.....	8
8.2. Grupo de Trabalho LexML (em Dezembro de 2008).....	8
8.3. Outras Contribuições	9
Anexo 1 – Esquema lexml_oai.xsd	10
Anexo 2 – Esquema provedor_profile_lexml.xsd	15

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, metadados de identificação (epígrafe, apelidos, identificadores, etc.) e metadados descritos (ementa e relacionamentos).

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.

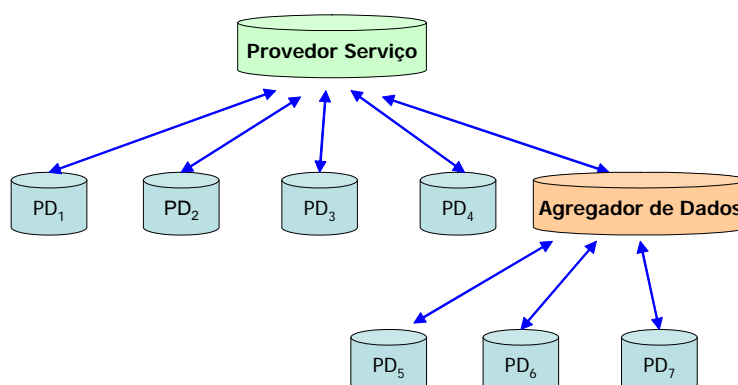


Figura 1. Tipos de Nodos de uma rede OAI-PMH.

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

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. Provedor de Dados - Papéis

No nodo provedor de dados, além do administrador responsável pela monitoração do fluxo da coleta de metadados (normalmente alguém com o conhecimentos de informática), são definidos um ou mais publicadores que são os responsáveis pelos recursos disponibilizados e pela garantia de sua persistência (associação entre URLs válidas com URNs). Normalmente eles são os gestores dos 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 e localidades.

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>
  <timestamp>2008-07-08-10:20:20:002221</timestamp>
</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, a sigla do órgão do publicador. Caso existe apenas um responsável pela publicação das informações, esse campo poderá ser omitido. Na seqüência, é identificado o domínio Internet ao qual o provedor de dados está relacionado. Após o caractere “:” é identificado a sigla do sistema de informações origem do recurso e, após a “/” o identificador interno utilizado por esse sistema. Como veremos a seguir, esse identificador interno poderá ser detalhado para indicar registros complementares (ex.:

anexos, retificações) que estão compartilhando a mesma identificação do registro 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` também 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 sequência, 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) possuem 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. Corpo do Registro de Metadados do Publicador Oficial

O órgão que publica oficialmente os documentos (Imprensa Nacional, Imprensa Oficiais dos Estados), 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" qtdPaginas="140">
      https://www.in.gov.br/imprensa/visualiza/index.jsp?jornal=do&secao=1&pagina=${pagina}&data=10/01/2007
    </Item>

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

    <Epigrafe>Diário Oficial da União. Seção 1. 10/01/2007</Epigrafe>

  </LexML>
</metadata>
```

Figura 4. Registro de Metadado do órgão Publicador Oficial.

3.5. Tipos de Relacionamentos

A Tabela 1 apresenta os tipos de relacionamentos que podem ser codificados no registro de metadados a ser enviado para o LexML.

Tabela 1. Tipos de Relacionamento.

Relacionamento	Descrição
altera	Alteração entre documentos.
correlato.a	Correlação entre documentos.
declara.inconstitucional	Declaração de inconstitucionalidade de uma norma no todo.
declara.inconstitucional.dispositivo	Declaração de inconstitucionalidade de um dispositivo de uma norma.
emenda.de	Relação entre a emenda e a proposição

	legislativa.
equivalente.a	Equivalência entre identificadores de documentos.
membro.de	Relação de pertinência.
parecer.de	Relação entre um parecer e o documento objeto do parecer.
parte.de	Relação todo-parte.
peticao.inicial.de	Relação entre a petição inicial e o processo.
publicacao.oficial	Relação entre o documento e o identificador da publicação oficial.
referencia	Referência entre documentos.
regulamenta	Relação de regulamentação.
revoga	Revogação de uma norma no todo.
revoga.dispositivo	Revogação de um dispositivo de uma norma.
sucessivo.ou.precedente.de	Relação de seqüência entre acórdãos.
sucessor.logico.de	Relacionamentos entre documentos que se sucedem logicamente.

4. Sobre o Esquema *lexml_oai.xsd*

O esquema *lexml_oai.xsd*, apresentado no Anexo 1, está organizado em duas partes:

- Parte inicial: são definidos os elementos e atributos que compõem a instância de um registro.
- Parte final: são definidos os tipos utilizados pelas definições da Parte Inicial.

5. Sobre o Esquema *provedor_profile_lexml.xsd*

O esquema *provedor_profile_lexml.xsd*, apresentado no Anexo 2, é organizado de forma hierárquica contendo os seguintes níveis:

- Elemento *ConfiguracaoProvedor*: elemento raiz;
- Elemento *Provedor*: contém informações sobre o tipo de provedor (borda ou agregador) e a URL Base do OAI (atributo *baseURL*); o primeiro

elemento filho identifica o responsável pelo funcionamento do provedor; Um Provedor possui um ou mais Publicadores.

- Elemento Publicador: contém informações sobre o publicador; O primeiro elemento filho identifica o responsável pela informação publicada. Um Publicador possui um ou mais perfis.
- Elemento Perfil: identifica uma combinação de localidade, autoridade e tipo de documento, além de informar se esta combinação pode ser referenciada no Documento Individual, no Relacionamento ou em ambos.

6. Referências

6.1. Sites

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

7. Lista de Abreviaturas e Siglas

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

HTTP – Hypertext Transfer Protocol

8. Créditos

8.1. Editores

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

Fernando Ciciliati (Senado Federal / Interlegis)

8.2. Grupo de Trabalho LexML (em Dezembro de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Cláudio Mori (Tribunal Superior Eleitoral)

Claudson dos Santos Melo (Tribunal Superior do Trabalho)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda Oliveira de Macedo (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)

Jonatas Rodrigues (Procuradoria Geral da República)

Luís Henrique Raja G. Mitchell (Tribunal de Contas da União)

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

Paulo de Tarso Penna da Costa (Superior Tribunal de Justiça)

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)

8.3. Outras Contribuições

Gabriel Franklin Araújo Mathias (Senado Federal / Interlegis)

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="unqualified"
attributeFormDefault="unqualified" xml:lang="PT">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
schemaLocation="xml.xsd"/>
  <xs:element name="LexML" type="lexml:LexMLType">
    <xs:annotation>
      <xs:documentation>
XML Schema for the OAI LexML Standard
Standard Version 1.0, 2008-12-13

João Oliveira
Prodasen

History:
2008-08-25: openoffice2 and and Sun StarOffice8 mime types added
2008-08-24: allowing tags p,b,i,u,s,sup,sub inside Ementa,
tag Item is now optional and unbounded
2008-08-13 release: v0.7

References:
- Formato de Item utiliza mime-type: http://www.iana.org/assignments/media-types/
- Códigos Idiomas: http://www.ietf.org/rfc/rfc3066.txt
</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" minOccurs="0" maxOccurs="unbounded">
          <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:attributeGroup
ref="lexml:ItemAttributeGroup"/>
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
        <xs:element name="DocumentoIndividual"
type="lexml:DocumentoIndividualType">
          <xs:annotation>
            <xs:documentation>URN no padrão URN LexML
```



```

        <xs:element name="b"
type="lexml:CampoTextoComTagsComIdiomaType" form="unqualified"/>
        <xs:element name="p"
type="lexml:CampoTextoComTagsComIdiomaType" form="unqualified"/>
        <xs:element name="i"
type="lexml:CampoTextoComTagsComIdiomaType" form="unqualified"/>
        <xs:element name="sup"
type="lexml:CampoTextoComTagsComIdiomaType" form="unqualified"/>
        <xs:element name="sub"
type="lexml:CampoTextoComTagsComIdiomaType" form="unqualified"/>
    </xs:choice>
    <xs:attributeGroup ref="lexml:IdiomaAttributeGroup"/>
</xs:complexType>
<xs:complexType name="DocumentoIndividualType">
    <xs:complexContent>
        <xs:extension base="lexml:URNType"/>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="URNType" abstract="false">
    <xs:annotation>
        <xs:documentation>Tipo URN</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
        <xs:extension base="lexml:REGEXURNType"/>
    </xs:simpleContent>
</xs:complexType>
<xs:attributeGroup name="IdiomaAttributeGroup">
    <xs:attribute ref="xml:lang" use="optional" default="pt-BR"/>
</xs:attributeGroup>
<xs:attributeGroup name="ItemAttributeGroup">
    <xs:attribute name="tipo" type="lexml:TipoItem" use="optional"
default="conteudo"/>
    <xs:attribute name="formato" type="lexml:FormatoIdentificadorItem"
use="optional"/>
    <xs:attribute name="idPublicador" type="lexml:idPublicador" use="required"/>
    <xs:attribute name="qtdPaginas" type="xs:integer" use="optional"/>
</xs:attributeGroup>
<xs:simpleType name="FormatoIdentificadorItem">
    <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/vnd.sun.xml.writer"/>
        <xs:enumeration value="application/vnd.sun.xml.impress"/>
        <xs:enumeration value="vnd.oasis.opendocument.graphics"/>
        <xs:enumeration value="vnd.oasis.opendocument.image"/>
        <xs:enumeration value="vnd.oasis.opendocument.presentation"/>
        <xs:enumeration value="vnd.oasis.opendocument.spreadsheet"/>
        <xs:enumeration value="vnd.oasis.opendocument.text"/>
        <xs:enumeration value="vnd.oasis.opendocument.text-master"/>
        <xs:enumeration value="vnd.oasis.opendocument.text-web"/>
        <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:restriction>
</xs:simpleType>

```

```

        <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="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="TipoItem">
    <xs:restriction base="xs:string">
        <xs:enumeration value="metadado"/>
        <xs:enumeration value="conteudo"/>
        <xs:enumeration value="conteudo. anotado"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="TipoRelacionamento">
    <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:enumeration value="parecer.de"/>
        <xs:enumeration value="parte.de"/>
        <xs:enumeration value="peticao.inicial.de"/>
        <xs:enumeration value="emenda.de"/>
        <xs:enumeration value="declara.inconstitucional"/>
        <xs:enumeration value="declara.inconstitucional.dispositivo"/>
        <xs:enumeration value="referencia"/>
        <xs:enumeration value="membro.de"/>
        <xs:enumeration value="sucessivo.ou.precedente.de"/>
        <xs:enumeration value="revoga"/>
        <xs:enumeration value="revoga.dispositivo"/>
        <xs:enumeration value="regulamenta"/>
        <xs:enumeration value="altera"/>
        <xs:enumeration value="correlato.a"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="REGEXURNT">
    <xs:restriction base="xs:string">
        <xs:pattern value="urn:lex(:[a-zA-Z0-9;!\\-@/$~\.\_\\[\],]+)"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="ItemURNT"/>

```

```
<xs:simpleType name="ItemanyURNTType">  
  <xs:restriction base="xs:anyURI">  
    <xs:minLength value="6"/>  
  </xs:restriction>  
</xs:simpleType>  
<xs:simpleType name="idPublicador">  
  <xs:restriction base="xs:integer"/>  
</xs:simpleType>  
</xs:schema>
```

Anexo 2 – Esquema provedor_profile_lexml.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:profile="http://www.lexml.gov.br/profile_lexml"
xmlns:xml="http://www.w3.org/XML/1998/namespace"
targetNamespace="http://www.lexml.gov.br/profile_lexml" elementFormDefault="qualified"
attributeFormDefault="unqualified" xml:lang="PT">
  <import namespace="http://www.w3.org/XML/1998/namespace"
schemaLocation="xml.xsd"/>
  <element name="ConfiguracaoProvedor" type="profile:ConfiguracaoProvedorType"/>
  <complexType name="ConfiguracaoProvedorType">
    <sequence>
      <element name="Provedor" type="profile:ProvedorType"
maxOccurs="unbounded"/>
    </sequence>
    <attribute name="dataGeracao" type="dateTime" use="required"/>
  </complexType>
  <complexType name="ProvedorType">
    <sequence>
      <element name="Administrador" type="profile:ResponsavelType"/>
      <element name="Publicador" type="profile:PublicadorType"
minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
    <attribute name="idProvedor" type="integer" use="required"/>
    <attribute name="nome" use="optional">
      <simpleType>
        <restriction base="string">
          <maxLength value="255"/>
          <whiteSpace value="preserve"/>
          <minLength value="1"/>
        </restriction>
      </simpleType>
    </attribute>
    <attribute name="tipo" use="optional" default="Provedor">
      <simpleType>
        <restriction base="string">
          <enumeration value="Provedor"/>
          <enumeration value="Agregador"/>
        </restriction>
      </simpleType>
    </attribute>
    <attribute name="baseURL" type="anyURI"/>
  </complexType>
  <complexType name="PublicadorType">
    <sequence>
      <element name="Responsavel" type="profile:ResponsavelType"/>
      <element name="Perfil" type="profile:PerfilType"
maxOccurs="unbounded"/>
    </sequence>
    <attribute name="idPublicador" type="integer" use="required"/>
    <attribute name="nome" use="required">
      <simpleType>
        <restriction base="string">
          <maxLength value="255"/>
          <whiteSpace value="preserve"/>
        </restriction>
      </simpleType>
    </attribute>
  </complexType>
```



```
<attribute name="sigla" use="optional">
  <simpleType>
    <restriction base="string">
      <maxLength value="25"/>
      <whiteSpace value="preserve"/>
    </restriction>
  </simpleType>
</attribute>
</complexType>
<complexType name="ResponsavelType">
  <attribute name="idResponsavel" type="integer" use="required"/>
  <attribute name="email" use="optional">
    <simpleType>
      <restriction base="string">
        <whiteSpace value="replace"/>
        <pattern value="^[^@]+@[^\.]+\.[^\.]+(\.[^@]+)"/>
      </restriction>
    </simpleType>
  </attribute>
  <attribute name="senha" type="string" use="optional"/>
</complexType>
<complexType name="RepositorioOAILexMLType">
  <attribute name="baseURL" type="anyURI" use="required"/>
</complexType>
<complexType name="PerfilType">
  <attribute name="localidade" use="required">
    <simpleType>
      <restriction base="string">
        <whiteSpace value="replace"/>
      </restriction>
    </simpleType>
  </attribute>
  <attribute name="autoridade" use="required">
    <simpleType>
      <restriction base="string">
        <whiteSpace value="replace"/>
      </restriction>
    </simpleType>
  </attribute>
  <attribute name="tipoDocumento" use="required">
    <simpleType>
      <restriction base="string">
        <whiteSpace value="replace"/>
      </restriction>
    </simpleType>
  </attribute>
  <attribute name="tipoPerfil" use="optional" default="T">
    <simpleType>
      <restriction base="string">
        <enumeration value="T"/>
        <enumeration value="R"/>
        <enumeration value="D"/>
      </restriction>
    </simpleType>
  </attribute>
</complexType>
</schema>
```



LexML Brasil
Parte 5 – Serviço de Resolução de URN

Versão 1.0 (RC1)
Brasília – Dezembro de 2008

LexML Brasil**Parte 5 – Serviço de Resolução de URN**

Versão 1.0 (Dezembro/2008)

Parte 5 – Serviço de Resolução de URN	2
1. Introdução.....	3
2. Princípios do Serviço de Resolução	3
2.1. Arquitetura geral do Sistema	3
2.2. O Catálogo para a Resolução	3
2.3. Fases do Processo de Resolução.....	3
2.4. Comportamento do Serviço de Resolução.....	4
3. Referências	5
3.1. Bibliográficas	5
3.2. Sites	5
4. Lista de Abreviaturas e Siglas	5
5. Créditos.....	5
5.1. Editores.....	5
5.2. Grupo de Trabalho LexML (em Dezembro de 2008).....	5

1. Introdução

Este documento apresenta requisitos para a implementação do Serviço de Resolução de URNs do projeto LexML.

O processo de resolução não se confunde com o processo de pesquisa no qual o usuário informa argumentos (palavras, datas etc) que serão pesquisadas em todos os campos de metadados (apelidos, ementa etc). Em um resultado de pesquisa, ao selecionar um item de interesse, invoca-se o processo de resolução. O serviço de resolução pode ser invocado diretamente, caso o usuário concatene ao prefixo “<http://www.lexml.gov.br/urn/>” a URN de interesse, conforme exemplo abaixo:

http://www.lexml.gov.br/urn/urn:lex:br:federal:lei:1993;8666

2. Princípios do Serviço de Resolução

2.1. Arquitetura geral do Sistema

O serviço de resolução tem a função de listar todas as ocorrências (URLs) de um documento a partir do nome uniforme (URN) fornecido.

O serviço de resolução é composto por três elementos:

- vocabulários controlados (Parte 6 da especificação LexML);
- catálogo de informações sobre a ocorrência dos documentos resultado do processo de coleta de metadados especificado na Parte 4;
- software de normalização e resolução de URNs.

2.2. O Catálogo para a Resolução

A implementação de um catálogo, construído sobre uma base de dados relacional, oferece uma maior flexibilidade para a resolução de nomes uniformes incompletos ou imprecisos, evento provável no campo jurídico, e ainda mais frequente se o nome uniforme do documento referenciado for construído automaticamente (por meio de um analisador léxico/sintático ou *parser*) a partir das referências em linguagem natural (português).

Cada autoridade emitente será responsável pela especificação dos termos do vocabulário controlado relativo aos seus próprios documentos, bem como a atribuição a cada recurso do nome uniforme adequado.

Além das localizações na rede (URL), o catálogo contém metadados básicos de interesse para o usuário tais como informações sobre ementa, apelidos de norma e formato de arquivo.

2.3. Fases do Processo de Resolução

Em geral o processo de resolução possui as seguintes fases:

- normalização do nome uniforme a ser resolvido, consistindo na transformação para a forma canônica de alguns componentes. Esta fase, particularmente útil no caso de construção automática do nome, ocupa-se, por exemplo, de expandir as siglas e as abreviações, de uniformizar as nomenclaturas das instituições, de padronizar os tipos de documentos etc.;
- interrogação ao catálogo com o nome uniforme normalizado e apresentação das informações relacionadas a um documento.

2.4. Comportamento do Serviço de Resolução

O serviço de resolução deve buscar no catálogo o nome uniforme com correspondência completa àquele informado. Nesse sentido, caso o usuário informe uma URN no nível de documento complexo, o resultado será composto por todos os documentos individuais relacionados ao documento complexo; caso o usuário informe uma URN no nível de documento individual, o resultado será composto por todas as ocorrências da versão específica do documento requerido; caso o usuário informe uma URN no nível de fragmento de um documento individual, o resultado será composto por todas as ocorrências do fragmento especificado.

Caso o resultado da resolução gere apenas uma URL, o sistema deve apresentar os metadados juntamente com o documento da URL encontrada, podendo, para isso, utilizar frames.

Caso o resultado da resolução não encontre nenhum resultado no nível de detalhe especificado (documento individual ou fragmento), mas existam uma ou mais ocorrências em níveis superiores, o sistema de resolução deve apresentar o resultado alternativo com um aviso informando sobre o processo de ajuste da resolução. Por exemplo, se o usuário especificou na URN o Art 1º de uma determinada norma, e o sistema de resolução não encontrou ocorrências específicas deste artigo, o sistema poderá realizar a tentativa de encontrar ocorrências no nível de documento individual, e, caso não encontre a versão desejada, poderá tentar encontrar ocorrências no nível de documento complexo que relaciona todas as versões. Nesses casos, não haverá redirecionamento automático mesmo que seja encontrada apenas 1 ocorrência nesses níveis superiores.

Considerando a possibilidade de ocorrência de nomes imprecisos e/ou incompletos provenientes das referências, o serviço de resolução deve, no caso de falta de correspondência, requisitar ao serviço de pesquisa um conjunto resultado que considere os elementos da URN especificada (*substrings*). O item selecionado pelo usuário em um resultado de pesquisa pode ser argumento para um novo processo de resolução.

Este comportamento é extremamente útil para o sucesso na busca dos documentos objeto de remissões, posto que, freqüentemente, a citação não contém todos os elementos para construir o nome uniforme canônico: isso se dá, por exemplo, com os atos citados apenas com número e ano; com atos numerados ou não, citados apenas pela

data etc. Tal funcionamento permite a simplificação e a automatização da construção do nome uniforme a partir da remissão.

De fato, os elementos do nome são considerados como se fossem compostos por diversos campos elementares, com valores posicionais internos (ex.: autoridades, datas) ou sem (ex.: números identificadores do ato); o confronto deveria assim se dar campo a campo, e resultar como positivo se todos os campos fornecidos encontrarem correspondência em um ou mais nomes uniformes do catálogo. Por ex.: o registro do catálogo com data do ato "2000-06-13" deveria resultar em confronto (posicional) positivo, mesmo com as datas informadas "2000" e "2000-06".

3. Referências

3.1. Bibliográficas

3.2. Sites

4. Lista de Abreviaturas e Siglas

URN – Uniform Resource Name

5. Créditos

5.1. Editores

Fernando Ciciliati (Senado Federal / Interlegis)

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

5.2. Grupo de Trabalho LexML (em Dezembro de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Cláudio Mori (Tribunal Superior Eleitoral)

Claudson dos Santos Melo (Tribunal Superior do Trabalho)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda Oliveira de Macedo (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)

Jonatas Rodrigues (Procuradoria Geral da República)

Luís Henrique Raja G. Mitchell (Tribunal de Contas da União)

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

Paulo de Tarso Penna da Costa (Superior Tribunal de Justiça)

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)



LexML Brasil
Parte 6 – Vocabulários Controlados

Versão 1.0 (RC1)
Brasília – Dezembro de 2008

LexML Brasil**Parte 6 – Vocabulários Controlados**

Versão 1.0 (Dezembro/2008)

Parte 6 – Vocabulários Controlados	2
1. Introdução.....	3
1.1. O Comitê Central para a Atribuição de Nomes	3
1.2. Atribuição dos Nomes e Inserção no Registro	4
2. Vocabulários Básicos	4
2.1. Natureza do Conteúdo	4
2.2. Língua.....	5
2.3. Evento.....	5
3. Vocabulários Específicos	6
3.1. Localidade	7
3.2. Autoridade.....	7
3.3. Tipo de Documento	8
4. Publicação do Vocabulários	9
5. Referências	9
5.1. Bibliográficas	9
5.2. Sítios da Internet.....	9
6. Lista de Abreviaturas e Siglas	9
7. Créditos.....	9
7.1. Editores.....	9
7.2. Grupo de Trabalho LexML (em Dezembro de 2008).....	9

1. Introdução

O controle de vocabulários desempenha papel fundamental na organização da informação legislativa e jurídica. Ele permitirá, entre outras coisas, a realização de pesquisas sofisticadas, a execução do processo de normalização da URN e a integração dos vocabulários utilizadas pelas diversas casas.

Por exemplo, o usuário poderá informar o prefixo “urn:lex:br:stf” na URN de um acórdão do Supremo Tribunal Federal. A expansão da sigla “stf” para o termo uniforme “supremo.tribunal.federal”, realizado pelo processo de normalização, será baseada no vocabulário de autoridades emitentes.

Outro importante uso dos vocabulários é na indicação do subtipo do documento emitido por uma autoridade. Nesse caso, cada entidade emitente poderá relacionar todos os subtipos utilizados no seu contexto como detalhamento do tipo principal. Por exemplo, no contexto do “Senado Federal” o tipo/subtipo “projeto.lei;plc” identifica os projetos de lei originários da Câmara dos Deputados. No contexto de uma determinada assembleia legislativa, o tipo/subtipo “projeto.lei;plc”, caso exista, terá certamente significado diverso.

Alguns elementos dos identificadores persistentes do Projeto LexML (Parte 2 - URN) referenciam termos dos vocabulários controlados.

Enquanto que alguns vocabulários são básicos, podendo ser utilizados por todos os tipos de documentos (ex.: natureza do conteúdo, língua e evento), outros deverão ser construídos na medida em que novos recursos são integrados à Rede de Informações LexML (ex.: localidades, autoridades emitentes e tipos de documentos).

O Projeto LexML identifica um conjunto de valores iniciais para os vocabulários básicos. Para os outros vocabulários, essa especificação indica um conjunto de valores iniciais e diretrizes que devem ser consideradas quando da inclusão de novos valores.

O vocabulário de assuntos, essencial para a efetiva recuperação da informação, não faz parte da atual versão do Projeto LexML. No futuro, será realizado um estudo para saber qual a melhor abordagem para tratar a indexação dos assuntos das informações legislativas e jurídicas considerando a diversidade dos vocabulários em uso pelas diversas instituições.

1.1. O Comitê Central para a Atribuição de Nomes

Deve ser constituído um Comitê Central que tenha autoridade sobre a atribuição dos elementos primários dos nomes uniformes, associados, respectivamente, à autoridade emitente e ao tipo de ato ou norma, de acordo com as orientações indicadas no presente documento.

O Comitê tem a tarefa de uniformizar as formas de emprego dos elementos, de efetuar as eventuais normalizações e, enfim, de resolver eventuais homônimas, além de manter um registro das autoridades e dos tipos de atos e de comunicados aprovados.

O registro de nomes será publicado de modo a difundir o conhecimento dos nomes uniformes e a favorecer o seu uso correto nos outros documentos.

1.2. Atribuição dos Nomes e Inserção no Registro

O Comitê Central deve atribuir um primeiro núcleo de nomes uniformes de autoridades emitentes, de seus órgãos e documentos normativos, e efetua a inserção no registro on-line.

As autoridades emitentes não presentes no registro (ou não presentes no grau de detalhes necessário) requerem a atribuição do nome relativo à instituição e/ou aos seus órgãos e funções (e ainda dos eventuais tipos de atos não ainda registrados) ao Comitê Central utilizando formulário disponível na Internet.

O Comitê, depois das verificações adequadas e de eventuais ajustes, comunica o nome aprovado e autoriza a criação de registros utilizando os formatos aprovados.

2. Vocabulários Básicos

2.1. Natureza do Conteúdo

Os valores deste vocabulário são baseados nos valores de categorias de conteúdo especificada no *draft* do padrão RDA (*Resource Description and Access*). Os termos são apresentados em ordem alfabética.

A codificação do termo na notação URN, apresentada na última coluna, segue as regras da Parte 2 da especificação LexML.

A natureza do conteúdo de um documento não se confunde com a natureza do seu suporte. Por exemplo, um “texto” poderá ter como suporte um arquivo de imagem. Esse vocabulário se preocupa com a forma original de expressão de um documento.

Categoria do Recurso	Descrição	Codificação (URN)
Imagem	Conteúdo expresso através de linhas, formatos, tons, etc. com a intenção de ser percebido visualmente como uma imagem parada ou imagem em duas dimensões. Inclui desenhos, pinturas, diagramas, fotografias, etc.	imagem
Imagem em Movimento	Conteúdo expresso através de imagens com a intenção de ser percebido visualmente como uma imagem em movimento, podendo ser ou não acompanhadas de áudio.	imagem.movimento

Música	Conteúdo expresso através de música em forma audível. Inclui registros de execuções musicais, musicas geradas por computador, etc.	musica
Notação Musical	Conteúdo expresso através de um sistema de notação para música com a intenção de ser percebido visualmente. Inclui todas as formas de notação musical.	notacao.musical
Texto	Conteúdo expresso através de um sistema de notação para linguagem com a intenção de ser percebido visualmente. Inclui todos os tipos de notação de linguagem exceto aqueles que permitem a percepção tátil.	texto
Texto Falado	Conteúdo expresso através de uma linguagem em forma audível. Inclui registro de leituras, recitações, falas, etc., voz sintetizada por computador, etc.	texto.falado

2.2. Língua

Inicialmente serão definidas as seguintes línguas:

Língua	Descrição	Codificação (URN)
de	Alemão	de
fr	Francês	fr
em	Inglês	en
Es	Espanhol	es
it	Italiano	it
pt-BR	Português (Brasil)	pt-br

2.3. Evento

O vocabulário de eventos é utilizado para identificar um ou mais eventos que deram origem a uma determinada versão de documento. Inicialmente serão definidos os seguintes eventos:

Evento	Descrição	Codificação (URN)
Iniciativa	Evento de iniciativa de uma proposição legislativa.	iniciativa
Assinatura	Evento de assinatura de documentos normativos.	assinatura
Julgamento	Evento de julgamento que dará origem a um acórdão ou a uma decisão monocrática. Inclui também o evento que deu origem a uma súmula (ex. sessão administrativa).	julgamento
Publicação	Evento de publicação oficial de documento.	publicacao
Retificação	Evento de retificação de uma publicação oficial.	retificacao
Re-publicação	Evento de re-publicação oficial de documento.	republicacao
Anulação	Evento de anulação de um documento oficial.	anulacao
Alteração	Alteração de um documento por outro, gerando uma nova versão. Inclui também as revogações (total ou parcial) e as inclusões de novo texto.	alteracao
Derrubada de Veto Parcial	Evento de derrubada de veto parcial.	derrubada.veto.parcial
Derrubada de Veto Total	Evento de derrubada de veto total.	derrubada.veto.total
Declaração de Inconstitucionalidade	Evento de declaração de inconstitucionalidade de uma norma.	declaracao.inconstitucionalidade

3. Vocabulários Específicos

3.1. Localidade

Os valores iniciais do vocabulário Localidade incluem as seguintes entidades:

- “Brasil”
- “Distrito Federal”
- 1 entrada para cada Estado com informações obtidas a partir das bases de dados do IBGE;
- 1 entrada para cada Município com informações obtidas a partir das bases de dados do IBGE;
- 1 entrada para cada região dos Tribunais Regionais Federais; e
- 1 entrada para cada região dos Tribunais Regionais do Trabalho.

Cada localidade terá no máximo um nome preferido para cada língua e poderá possuir diversos nomes alternativos que pode considerar, por exemplo, as alterações toponímicas.

Se determinado município passou a pertencer a uma outra unidade da federação (ex. no evento de criação do Estado de Tocantins), deve-se criar uma nova localidade.

Caso ocorra mudança na categoria da localidade (ex. de Território para Estado), deve-se criar uma nova localidade no vocabulário.

3.2. Autoridade

O vocabulário de Autoridade é inicialmente preenchido com apenas três valores, conforme a seguir:

Autoridade	Descrição	Codificação (URN)
Federal	Normas de hierarquia superior da esfera federal (constituição, emendas, leis, leis complementares, medidas provisórias, decretos-leis e decretos)	federal
Estadual	Normas de hierarquia superior da esfera estadual (constituição, emendas, leis, leis complementares, medidas provisórias, decretos-leis e decretos)	Estadual
Municipal	Normas de hierarquia superior da esfera municipal (lei orgânica, emendas, leis, leis complementares, medidas provisórias, decretos-leis e decretos)	Municipal

Os demais valores serão preenchidos com a adesão das entidades à Rede de Informações LexML. Cada publicador, ligado a um provedor de dados, irá informar

qual a combinação de tipos de localidade, de autoridades e de documento que irá prover ao sistema.

3.3. Tipo de Documento

O vocabulário Tipo de Documento é inicialmente preenchido com os seguintes valores:

Categoria	Tipo de Documento	Codificação (URN)
Proposição Legislativa	Projeto de Lei	projeto.lei
	Projeto de Lei Complementar	projeto.lei.complementar
	Proposta de Emenda Constitucional	proposta.emenda.constitucional
	Projeto de Resolução	projeto.resolucao
	Projeto de Lei de Conversão	projeto.lei.conversao
	Projeto de Decreto Legislativo	projeto.decreto.legislativo
	Emenda	emenda
Normas	Constituição	constituicao
	Emenda Constitucional	emenda.constitucional
	Lei Complementar	lei.complementar
	Lei Delegada	lei.delegada
	Lei Ordinária	lei
	Decreto-Lei	decreto.lei
	Medida Provisória	medida.provisoria
	Decreto	decreto
	Resolução	resolucao
	Portaria	portaria
	Instrução Normativa	instrucao.normativa
Julgados	Acórdão	acordao
	Súmula	sumula
	Súmula Vinculante	sumula.vinculante
	Decisão Monocrática	decisao.monocratica

Os demais valores serão preenchidos com a adesão das entidades à Rede de Informações LexML. Cada publicador, ligado a um provedor de dados, irá informar qual a combinação de tipos de localidade, de autoridades e de documento que irá prover ao sistema.

Cada autoridade emitente poderá definir subtipos de documentos, conforme apresentado na Introdução deste documento.

4. Publicação do Vocabulários

Os vocabulários estão publicados no sítio LexML (<http://www.lexml.gov.br/vocabularios>) utilizando o formato W3C SKOS (*Simple Knowledge Organization System*).

5. Referências

5.1. Bibliográficas

5.2. Sítios da Internet

<http://www.collectionscanada.gc.ca/jsc/docs/5rda-parta-categorization.pdf>- RDA

<http://www.w3.org/2004/02/skos/> - SKOS

6. Lista de Abreviaturas e Siglas

RDA – Resource Description and Access

IBGE – Instituto Brasileiro de Geografia e Estatística

SKOS - *Simple Knowledge Organization System*

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 Dezembro de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Cláudio Mori (Tribunal Superior Eleitoral)

Claudson dos Santos Melo (Tribunal Superior do Trabalho)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)
Flávia Lacerda Oliveira de Macedo (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)
Jonatas Rodrigues (Procuradoria Geral da República)
Luís Henrique Raja G. Mitchell (Tribunal de Contas da União)
Paulo André Mattos de Carvalho (Tribunal de Contas da União)
Paulo de Tarso Penna da Costa (Superior Tribunal de Justiça)
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)