



LexML Brasil
Parte 3 – LexML XML Schema

Versão 0.7
Brasília – Agosto de 2008

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

Versão 0.7 (agosto/2008)

Parte 3 – LexML XML Schema	2
1. Introdução	3
2. Esquemas Rígido e Flexível	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.....	12
11.1. Editores.....	12
11.2. Grupo de Trabalho LexML (em Agosto de 2008).....	13
Anexo 1 – Esquema lexml07-base.xsd.....	14
Anexo 2 - Esquema lexml07-br-rigido.xsd	33
Anexo 3 - Esquema lexml07-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 de *XML* e de *XML Schema*.

2. Esquemas Rígido e Flexível

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 da LCP 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. Serve também 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/0.7” 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 cegos ou pessoas com baixa visão ou, ainda, erro na utilização de software 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 sequência, 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 – sequência 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 “lexml07-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 “lexml07-flexivel.xsd”. Os esquemas rígido e flexível fazem inclusão do esquema base, localizado no arquivo “lexml07-base.xsd”.

Na sequência, 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 um 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 sequencial 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
<Alinea>	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: epigrafe, 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 sequencial 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 sequência 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, além do texto 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, foram utilizados dois tipos de construtores para conter o texto propriamente dito:

- `<Texto>` - elemento tipo *container*, que admite vários blocos (`<p>`, `<table>`, ``, `` etc)
- `<TextoSimples>` - elemento tipo *bloco*, permitindo conteúdo equivalente ao de um parágrafo `<p>`.

O texto propriamente dito é marcado utilizando-se um subconjunto de elementos HTML.

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 do elemento `<Texto>` ou `<TextoSimples>` 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:sp: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 conterá 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 ser referenciados em outros 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]

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

10. Lista de Abreviaturas e Siglas

CSS – Cascade Styling Sheets

DTD – Document Type Definition

FRBR – Functional Requirements for Bibliographical Records

NIR – Norme in Rete

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 Agosto de 2008)

Alfredo Luiz Campos Júnior (Câmara dos Deputados / CENIN)

Carlos Corrêa Gonçalves (Tribunal Superior Eleitoral)

Cláudio Morale (Senado Federal / Interlegis)

Dalva Luca (Ministério da Justiça)

Fernando Teixeira (Câmara dos Deputados / CENIN)

Flávia Lacerda (Tribunal de Contas da União)

Flávio Henrique Rocha e Silva (Supremo Tribunal Federal)

Flávio Heringer (Senado Federal)

Manuel de Medeiros Dantas (Advocacia Geral da União)

Jean Rodrigo Ferri (Senado Federal / Interlegis)

João Alberto de Oliveira Lima (Senado Federal / Prodasen)

João Batista de Holanda Neto (Senado Federal / Prodasen)

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

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

Paulo Martins Inocêncio (Conselho da Justiça Federal)

Ricardo Bravo (Tribunal de Contas da União)

Sérgio Falcão (Câmara dos Deputados / CENIN)

Virgínia Azevedo (Supremo Tribunal Federal)

Anexo 1 – Esquema lexml07-base.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="http://www.lexml.gov.br/0.7"
  xmlns="http://www.lexml.gov.br/0.7"
  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 - joaolima@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.

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

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

```

```

        <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: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>
    =====

    Tipos Simples
    =====
  </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="TextoSimples" 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="Texto" minOccurs="0" maxOccurs="1" />
  </xsd:sequence>
  <xsd:attributeGroup ref="corereq"/>
</xsd:complexType>

<xsd:complexType name="textoSimplesType">
  <xsd:sequence>
    <xsd:element ref="TextoSimples" minOccurs="0" maxOccurs="1" />
  </xsd:sequence>
</xsd:complexType>
```

```

        </xsd:sequence>
        <xsd:attributeGroup ref="corereq"/>
    </xsd:complexType>

    <xsd:complexType name="textoSimplesOptType">
        <xsd:sequence>
            <xsd:element ref="TextoSimples" 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: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>

<!-- ArticulaÃ§Ã£o (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: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: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>

<!-- Jurisprudencia ( elementos especÃ-ficos ) -->

  <xsd:element name="EmentaAcordao" type="textoType"/>

  <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:complexType>
    </xsd:element>

    <xsd:element name="Acordao">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="CabecalhoAcordao" 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:element ref="EmentaAcordao" minOccurs="1" maxOccurs="1"/>
            </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: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="Texto" minOccurs="0" maxOccurs="1"/>
        <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="TextoSimples" 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="textoSimplesOptType"/>

<xsd:element name="AspasTexto" type="inline" /> <!-- nÃ£o se aplica Ã 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="TextoSimples" 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:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```



```

        <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:extension>
        </xsd:complexContent>
    </xsd:complexType>
    <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: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: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: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"/>
          </xsd:choice>
          <xsd:attributeGroup ref="date"/>
        </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 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>
```



```

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

<xsd:schema targetNamespace="http://www.lexml.gov.br/0.7"
  xmlns="http://www.lexml.gov.br/0.7"
  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="lexml07-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\})_inc\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\d+((-[0-9]{1,3})?)\{0,3\})_ali\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\})_ali\d+((-[0-9]{1,3})?)\{0,3\})_ite\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|agh)\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|agh)\d+((-[0-9]{1,3})?)\{0,3\}_(liv|agh)\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|agh)\d+((-[0-9]{1,3})?)\{0,3\}_(liv|agh)\d+((-[0-9]{1,3})?)\{0,3\})_(tit|agh)\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|agh)\d+((-[0-9]{1,3})?)\{0,3\}_(liv|agh)\d+((-[0-9]{1,3})?)\{0,3\})_(tit|agh)\d+((-[0-9]{1,3})?)\{0,3\})_(cap|agh)\d+((-[0-9]{1,3})?)\{0,3\})_(sec|agh)\d+((-[0-9]{1,3})?)\{0,3\})_(sub|agh)\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>

```

```

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

    <xsd:element name="Parte">
        <xsd:complexType>
            <xsd:complexContent>
                <xsd:extension base="hierarchy">
                    <xsd:sequence minOccurs="0" maxOccurs="unbounded" >
                        <xsd:element ref="Artigo" minOccurs="0"
maxOccurs="unbounded" />
                        <xsd:element ref="Livro" 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:sequence>
                            <xsd:element ref="Artigo" minOccurs="0"
maxOccurs="unbounded" />
                            <xsd:element ref="Titulo" minOccurs="0"
maxOccurs="unbounded" />
                        </xsd:sequence>
                    </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:sequence>
                            <xsd:element ref="Artigo" minOccurs="0"
maxOccurs="unbounded" />
                            <xsd:element ref="Capitulo" minOccurs="0"
maxOccurs="unbounded" />
                        </xsd:sequence>
                    </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:sequence>
                            <xsd:element ref="Artigo" minOccurs="0"
maxOccurs="unbounded" />
                            <xsd:element ref="Secao" minOccurs="0"
maxOccurs="unbounded" />
                        </xsd:sequence>
                    </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:sequence>

```

```

maxOccurs="unbounded" />
<xsd:element ref="Artigo" minOccurs="0"
maxOccurs="unbounded" />
<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="Texto" minOccurs="0" maxOccurs="1"/>
<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>

```

```

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

<xsd:schema targetNamespace="http://www.lexml.gov.br/0.7"
  xmlns="http://www.lexml.gov.br/0.7"
  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="lexml07-base.xsd"/>

  <xsd:simpleType name="idArtigo">
    <xsd:annotation>
      <xsd:documentation>
        A regras abaixo tratam dos IDs de dispositivos no nível de artigo para baixo.
        TBD: flexibilizar regras de validação dos ids
      </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}) _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}) _inc(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}) _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}) _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.
        TBD: flexibilizar regras de validação dos ids
      </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|agh)\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|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(liv|agh)\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|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(liv|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(tit|agh)\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|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(liv|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(tit|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(cap|agh)\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|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(liv|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(tit|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(cap|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(sec|agh)\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|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(liv|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(tit|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(cap|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(sec|agh)\d+((-[0-9]{1,3})?) {0,3}) _?(sub|agh)\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>

```

```

=====
</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: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="Texto" minOccurs="0" maxOccurs="1"/>
        <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>

```