

Sistemas Distribuídos na WEB

(Plataformas para Aplicações Distribuídas)

XML (eXtensible Markup Language)

XML

Origens

- ✚ XML (eXtensible Markup Language – linguagem de marcação extensível)
- ✚ Criada em 1996 pelo W3C (World Wide Web Consortium)
- ✚ Subconjunto do SGML (Standard Generalized Markup Language) criada por Jon Bosak.
- ✚ XML foi criado para ser uma metalinguagem flexível, mas formal, para ser usada na Internet

HTML

Problemas

- ✚ HTML (Tim Berners Lee - CERN)
- ✚ HTML não é extensível.
- ✚ HTML é focado para exibição.
- ✚ HTML não possui estrutura semântica.
- ✚ HTML não permite múltiplas visualizações de um conjunto de dados.
- ✚ Conjunto pré-definido de elementos (tags) para especificação das dimensões de estrutura e apresentação de um documento

HTML

Exemplo

```
010010100101001001010111
1101010010101010100110
0011010101010010101000
<p><b>Mrs. Mary McGoon</b>
<br>
1401 Main Street
<br>
Anytown, NC 34829</p>
```

HTML

Exemplo

```
010010100101001001010111
1101010010101010100110
0011010101010010101000
<p><b>Mrs. Mary McGoon</b>
<br>
1401 Main Street
<br>
Anytown, NC 34829</p>
```

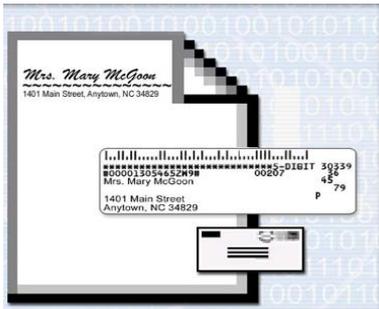
Mrs. Mary McGoon
1401 Main Street
Anytown, NC 34829

XML

Exemplo

```
010010100101001001010111
1101010010101010100110
0011010101010010101000
<address>
<name>
<title>Mrs.</title>
<first-name>Mary</first-name>
<last-name>McGoon</last-name>
</name>
<street>1401 Main Street</street>
<city>Anytown</city>
<state>NC</state>
<zipcode>34829</zipcode>
</address>
```

XML Exemplo



Prof. Carlos Paes

7

XML Aplicações

- ✦ Data Interchange
- ✦ B2B (Web Services)
- ✦ Permitir um melhor suporte para agentes inteligentes.
- ✦ Suportar buscas mais inteligentes.
- ✦ Descrição e configuração de objetos (Componentes J2EE).

Sistemas Distribuídos 2007
Prof. Carlos Paes

8

XML eXtensible Markup Language

- ✦ Linguagem de Tags:
 - Abertura: `<elem>`
 - Fechamento: `</elem>`
 - Abertura e Fechamento imediato: `<elem />`
 - Atributo: `<elem atrib="value" />`
- ✦ O seu objetivo é diferente do HTML
 - XML descreve os dados
 - HTML apresenta-os

Sistemas Distribuídos 2007
Prof. Carlos Paes

9

XML

Sintaxe

- ✦ Regras de construção simples mas rígidas:
 - Existe um elemento raiz único
 - Todos os elementos têm que fechar
 - Fecham pela ordem inversa em que abrem
 - Os nomes são case-sensitive
- ✦ Um documento que respeita as regras diz-se bem formado (well formed)
- ✦ Adicionalmente, pode-se definir uma gramática para validar documentos XML
 - Que seqüências de elementos são válidas
 - Que tipo de dados contém o elemento

XML

Exemplo

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- editado por App -->
<mensagem>
  <de>João</de>
  <para>Carla</para>
  <assunto>Reunião urgente</assunto>
</mensagem>
```

XML

Dimensões e Processamento

- ✦ XML
 - Dimensões de estrutura e conteúdo ²
 - Documentos bem formados! ¹
- ✦ Outras dimensões de um documento XML
 - Apresentação: CSS, XSL ³
 - Estrutura e Semântica: DTDs e XML Schemas ⁴
 - Metadados e mais semântica: RDF ⁵
 - Estrutura de hipertexto: XLink e XPointer ⁶
- ✦ Processamento de documentos XML
 - Parsers, APIs, DOM... *nossas aplicações...*

XML

Criando Aplicações

✦ Para criarmos uma aplicação são necessários os seguintes passos:

- Selecionar ou definir um DTD (Document Type Definition).
- Gerar documentos XML.
- Interpretar documentos XML.
- Exibir documentos XML.

XML

DTD (Document Type Definition)

✦ Um **DTD** define quais tags podem aparecer num documento, quais tags podem conter outras tags, a numeração e seqüência de tags, os atributos dos tags e valores opcionais.

✦ Segue a gramática estabelecida por uma **Document Type Definition (DTD)**. Um documento XML pode estar, opcionalmente, associado a uma gramática que rege sua estrutura e composição, denominada DTD. Essa é a gramática definida com a declaração de elementos (ELEMENT) e seus atributos (ATTLIST)

XML

DTD (Document Type Definition)

✦ Gramática que define

- Estrutura
- Elementos válidos

✦ Sintaxe própria diferente de XML

✦ Documento XML pode ter

- DTD embutido
- Referência externa

DTD (Document Type Definition) Declaração de elementos

- ✿ ", " : elementos devem ocorrer na ordem especificada;
- ✿ "1" : apenas um elemento deve ocorrer;
- ✿ "+": o elemento deve ocorrer 1 ou + vezes;
- ✿ "*": o elemento pode ocorrer zero ou + vezes;
- ✿ "?": o elemento é opcional;
- ✿ #PCDATA: uma seqüência de caracteres;
- ✿ EMPTY: indica que o elemento é vazio;
- ✿ ANY: o elemento pode conter qq conteúdo declarado.

```
<ELEMENT layout ANY>  
<!ATTLIST layout  
  id ID #IMPLIED  
  type CDATA "text/xml-basic"  
>
```

XML Exemplo DTD

- ✿ Exemplo de documento XML que referencia DTD:

```
<?xml version="1.0"?>  
<!DOCTYPE note SYSTEM "mensagem.dtd">  
<?xml version="1.0" encoding="ISO-8859-1"?>  
<mensagem>  
  <de>João</de>  
  <para>Carla</para>  
  <assunto>Reunião urgente</assunto>  
</mensagem>
```

XML Exemplo DTD

- ✿ Exemplo de DTD (mensagem.dtd):

```
<!ELEMENT mensagem (de,para,assunto)>  
<!ELEMENT de (#PCDATA)>  
<!ELEMENT para (#PCDATA)>  
<!ELEMENT assunto (#PCDATA)>
```

XML

Exemplo Receita

```
<?xml version="1.0"?>
<Receita>
  <Nome>Torta de limao</Nome>
  <Descricao>
    receita favorita de minha avó.
  </Descricao>
  <Ingredientes>
    <Ingrediente>
      <Qty unidade="caixa">1</Qty>
      <Item>gelatina de limao</Item>
    </Ingrediente >
    <Ingrediente>
      <Qty unidade="lata">2</Qty>
      <Item>leite condensado</Item>
    </Ingrediente>
    ....
  </Ingredientes>
</Receita>
```

XML

DTD da Receita

!- Este eh o DTD para o exemplo da Receita -->

```
<!ELEMENT Receita (Nome, Descricao?, Ingredientes?, Instrucoes?)>
  <!ELEMENT Nome (#PCDATA)>
  <!ELEMENT Descricao (#PCDATA)>
  <!ELEMENT Ingredientes (Ingredient)*>
  <!ELEMENT Ingrediente (Qty, Item)>
  <!ELEMENT Qty (#PCDATA)>
  <!ATTLIST Qty unidade CDATA #REQUIRED>
  <!ELEMENT Item (#PCDATA)>
  <!ATTLIST Item optional CDATA "0"
    isVegetarian CDATA "true">
  <!ELEMENT Instrucoes (Passos)*>
```

XML

Exercício DTD

👉 Seja um banco de dados escolar com diferentes cursos. (Cada curso com os seguintes atributos: nome, professor e número de sala) e em cada curso uma lista de alunos participantes (cada aluno com nome, endereço e RG). Defina um documento XML que descreva este banco de dados. Crie um DTD para descrever documentos deste tipo

- 👉 Resposta documento XML
- 👉 Resposta documento DTD

XML Schema Definition (XSD)

- ✦ Gramática tal como o DTD
- ✦ Escrita em XML
- ✦ Extensível
- ✦ Suporta tipos de dados
 - Decimais
 - Datas
 - etc
- ✦ Suporta espaços de nomes (namespaces)

Exemplo de XSD

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  >
  <xs:element name="shiporder">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="orderperson"
          type="xs:string"/>
        <xs:element name="shipto">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="name"
                type="xs:string"/>
              <xs:element name="address"
                type="xs:string"/>
              <xs:element name="city"
                type="xs:string"/>
              <xs:element name="country"
                type="xs:string"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="item"
          maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="title"
                type="xs:string"/>
              <xs:element name="note"
                type="xs:string"
                minOccurs="0"/>
              <xs:element name="quantity"
                type="xs:positiveInteger"/>
              <xs:element name="price"
                type="xs:decimal"/>
            </xs:sequence>
          </xs:complexType>
          </xs:element>
        </xs:sequence>
        <xs:attribute name="orderid"
          type="xs:string"
          use="required"/>
      </xs:complexType>
    </xs:element>
  </xs:schema>
```

Referência de documento a XSD

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<shiporder orderid="889923"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="shiporder.xsd">
  <orderperson>John Smith</orderperson>
  <shipto>
    <name>Ola Nordmann</name>
    <address>Langgt 23</address>
    <city>4000 Stavanger</city>
    <country>Norway</country>
  </shipto>
  <item>
    <title>Empire Burlesque</title>
    <note>Special Edition</note>
    <quantity>1</quantity>
    <price>10.90</price>
  </item>
  <item>
    <title>Hide your heart</title>
    <quantity>1</quantity>
    <price>9.90</price>
  </item>
</shiporder>
```

Exemplo de XSD (Escola)

- ✦ Duas formas de descrever uma escola (conforme o exercício anterior de DTD)
 - Forma 1
 - Forma 2

eXtensible Stylesheet Language (XSL)

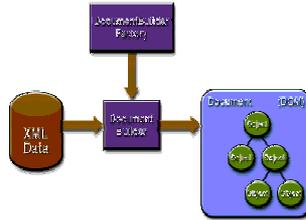
- ✦ Permite:
 - Transformar XML em HTML
 - Transformar XML em outro XML
 - Filtrar e ordenar dados XML
- ✦ Apresentar o mesmo documento de formas diferentes dependendo do dispositivo de destino (Palm, impressão, telefone celular, etc)

Exemplo de XSL

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
<h2>My CD Collection</h2>
<table border="1">
<tr bgcolor="#9acd32">
<th align="left">Title</th>
<th align="left">Artist</th>
</tr>
<xsl:for-each select="catalog/od">
<tr>
<td><xsl:value-of select="title"/></td>
<td><xsl:value-of select="artist"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

Document Object Model (DOM)

Manipulação de árvore em memória



Mais Informações...

- ✦ <http://www.w3schools.com>
 - Tutoriais Web, incluindo XML, DTD, Schema e XSL
- ✦ Java Web Services Developer Pack Tutorial 1.3
 - Capítulos 5 a 11
- ✦ Java Docs
 - `javax.xml.*`
 - `org.w3c.dom`
 - `org.xml.sax`
 - `javax.xml.transform`

Mais Informações...

- ✦ Recomendação disponível on-line em <http://www.w3.org/TR/REC-xml>
- ✦ Trata-se de um formato universal para documentos estruturados e dados na WWW (<http://www.w3c.org/XML>)
- ✦ XML em 10 pontos:
<http://www.w3.org/XML/1999/XML-in-10-points.html>
