



Programação para Web

XML

Ivo Calado

Instituto Federal de Educação, Ciência e Tecnologia de Alagoas

22 de Fevereiro de 2016



Roteiro

- 1 Introdução
- 2 Especificações XML
- 3 XML a fundo
- 4 Desenvolvendo aplicações com XML
- 5 Validação de XMLs



Definição

- EXtensible Markup Language
- É uma linguagem de marcação bastante semelhante à HTML



Definição

- EXtensible Markup Language
- É uma linguagem de marcação bastante semelhante à HTML
- Porém não descende e nem originou o HTML
- É originária do SGML



Definição

- EXtensible Markup Language
- É uma linguagem de marcação bastante semelhante à HTML
- Porém não descende e nem originou o HTML
- É originária do SGML
- É bem mais flexível que o HTML
- É uma recomendação da W3C



Diferença entre XML e HTML

Qual seria a diferença básica entre XML e HTML?



Diferença entre XML e HTML

Qual seria a diferença básica entre XML e HTML?

XML foi projetado para armazenar dados e não para exibí-los!

- XML não é um substituto do HTML
- Diferentes objetivos:
 - XML foi projetado para **descrever** dados e com foco no **que o dado é**
 - HTML foi projetado para **exibir** dados e com foco em como a informação será **apresentada**



Exemplo de XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```





Porque XML é importante?

Texto Plano

- Fácil de editar
- Útil para armazenar pequenas quantidades de dados
- Possibilita o armazenamento eficiente de uma grande quantidade de dados através de um *XML front end* e de um banco de dados

Identificação dos Dados

- Descreve o tipo de informação armazenado (Metadado)
- Pode ser usado de diferentes formas por diferentes aplicações



Porque XML é importante?

Estilo de exibição

- Desassocia o armazenamento da informação do seu processamento



Porque XML é importante?

Estilo de exibição

- Desassocia o armazenamento da informação do seu processamento
- Para isso, usa-se a linguagem XSL
- XSL possibilita a apresentação do conteúdo armazenado no XML sob diferentes estilos

Reusabilidade inline

- Pode ser composto de diferentes entidades
- É possível modularizar o documento e fazer referencia entre eles



Porque XML é importante?

Facilmente processável

- Possui uma notação regular e consistente
- Notação independente de fabricante

Formato baseado em hierarquia

- Rápido de acessar
- Fácil de rearranjar



Especificações XML

Diversas especificações fazem parte do padrão XML

- **XML 1.0:** define a estrutura da sintaxe de documento XML
- **XPointer e XLink:** define uma abordagem padrão para representação referência entre recursos
- **XSL:** Define uma abordagem padrão para representação de estilos a serem aplicações em documentos XML



Elementos

- Delimitados por colchetes angulares (< e >)
- Identifica a natureza do conteúdo que circunda

Formato geral:

```
<element> ... </element>
```

Elementos vazios:

```
<emptyelement />
```



Atributos

- Pares chave-valor
- Em geral usa-se como metadado sobre os elementos
- Os valores devem estar entre aspas

```
<element attribute="value">
```



Sintaxe XML

- Todos elementos devem ter uma tag de fechamento
- Tags XML são *case sensitive*
- Todo os elementos XML devem estar corretamente aninhados



Sintaxe XML

- Todos elementos devem ter uma tag de fechamento
- Tags XML são *case sensitive*
- Todo os elementos XML devem estar corretamente aninhados

```
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder
    <body>Don't forget me this weekend!
  </heading>
  </body>
</note>
```



Sintaxe XML

- Espaços em branco são preservados
- Comentários seguem a mesma estrutura do HTML



Conclusões a partir da sintaxe

- Os elementos são extensíveis
- Há um relacionamento entre os elementos
- Os elementos tem conteúdo (que podem ser inclusive outros elementos)



Dúvida

Quando usar elementos e quando usar atributos?



Dúvida

Quando usar elementos e quando usar atributos?

Metainformação sobre os elementos devem ser armazenados como atributos e o dado por si só deve ser armazenado como elemento.



Desenvolvendo aplicações com XML

- Diversas APIs estão disponíveis para o desenvolvimento de aplicações que fazem uso de XML
- Nessas APIs é possível a criação de uma árvore XML, sendo possível inclusive a validação com base em DTD ou XMLSchema
- Vamos trabalhar com a API JDOM



Por que é necessário validar um documento XML?



Por que é necessário validar um documento XML?

- Apesar do arquivo estar sintaticamente correto poderemos querer que ele siga alguma estrutura



Por que é necessário validar um documento XML?

- Apesar do arquivo estar sintaticamente correto poderemos querer que ele siga alguma estrutura
- Suponha que desejamos criar a estrutura de uma entidade Pessoa, poderemos querer forçar que a pessoa tenha um elemento informando o CPF e ter apenas um elemento endereço



Por que é necessário validar um documento XML?

- Apesar do arquivo estar sintaticamente correto poderemos querer que ele siga alguma estrutura
- Suponha que desejamos criar a estrutura de uma entidade Pessoa, poderemos querer forçar que a pessoa tenha um elemento informando o CPF e ter apenas um elemento endereço
- Para isso usamos estruturas de validação DTD ou XMLSchema



DTD

- Acrônimo de *Document Type Definition*
- Especifica a estrutura a ser utilizada pelo documento XML
- Pode ser definido tanto dentro do XML quanto em um documento externo

```
<!DOCTYPE note [  
  <!ELEMENT note (to , from , heading , body)>  
  <!ELEMENT to (#PCDATA)>  
  <!ELEMENT from (#PCDATA)>  
  <!ELEMENT heading (#PCDATA)>  
  <!ELEMENT body (#PCDATA)>  
>]
```





DTD

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE note SYSTEM "Note.dtd">
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

```
<!DOCTYPE root-element SYSTEM "filename">
```



O que é?

- Trata-se de uma alternativa ao DTD
- Oferece uma abordagem baseada em XML
- Também referenciado como XSD



O que possibilita?

- Define os elementos e atributos que podem aparecer no documento
- Define a ordem, o número e quais são os elementos filhos de um certo elemento
- Define se um elemento pode ser vazio ou se contém texto
- Define o tipo de dado de cada elemento e atributo
- Possibilita definir valores padrões para elementos e atributos



Porque XMLSchema são sucessores dos DTDs?

- São mais ricos e mais poderosos que os DTDs
- São escritos em XML
- Suportam tipos de dados



Exemplo XSD

```

<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://mynamespace/2011/empresa"
xmlns="http://www.w3schools.com"
elementFormDefault="qualified">
<xs:element name="note">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="to" type="xs:string"/>
      <xs:element name="from" type="xs:string"/>
      <xs:element name="heading" type="xs:string"/>
      <xs:element name="body" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```




Exemplo XML

```
<?xml version="1.0"?>
<note xmlns="http://www.w3schools.com"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.w3schools.com note.xsd">
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```



O que são namespaces

- Provém um método de evitar conflitos em nomes de elementos
- É útil quando é necessário unir documentos de dois documentos XML



Exemplo Namespace

Forma de usar: **xmlns:prefix="URI"**

```
<root>
<h:table xmlns:h="http://www.w3.org/TR/html4/">
  <h:tr>
    <h:td>Apples</h:td>
    <h:td>Bananas</h:td>
  </h:tr>
</h:table>
<f:table xmlns:f="http://www.w3schools.com/furniture">
  <f:name>African Coffee Table</f:name>
  <f:width>80</f:width>
  <f:length>120</f:length>
</f:table>
</root>
```



Trabalho I

Implementar mecanismo de serialização e desserialização de objetos. Fornecer suporte aos seguintes mecanismos

- Tipos primitivos
- Tipos complexos definidos pelo usuário
- Arrays de tipos complexos
- Objetos do tipo ArrayList

Implementar mecanismo de XMLSchema que valide o arquivo XML