

A Linguagem XML **(*& Markup Languages*)**

Renata Pontin M. Fortes

(renata@icmc.usp.br)

PAE: Willian Watanabe (watinha@gmail.com)

Instituto de Ciências Matemáticas e de Computação

ICMC-USP S.Carlos, 2010

Roteiro



- Linguagens de marcação
- Introdução a XML
- Diferenças entre HTML e XML
- Documentos XML
 - bem-formados
 - válidos
- DTD para definir novos tipos de documentos
- Aplicações de XML
- Dimensões dos Documentos: conteúdo, estrutura e apresentação

Markup Languages

O que é linguagem de marcação?

“uma linguagem de marcação (**markup language**) combina texto e informações extra sobre o texto.

Exemplos de markup languages:

SGML

Standard Generalized Markup Language

ISO 8879:1986

HTML

Hypertext Markup Language

RFC 1866 (HTML 2.0)

Markup Languages

De que constitui uma linguagem de marcação?

Tags (etiquetas)

descriptive markups

Marcação que delimita um elemento (SGML)

Elemento

Um componente da estrutura de um documento



Markup Language - HTML

O que é HTML?

HTML

- primeiro padrão amplamente utilizado: **HTML 2.0** que foi liberada em 1995
- **HTML 3.2**: recomendação em 1997
- **HTML 4.01**: recomendação em 1999
- **em 2000, HTML foi atualizada para incluir recursos da XML: **XHTML** (*EXtensible Hypertext Markup Language*)**
- HTML originalmente projetada para disseminar dados estruturados, passou a ser também um modo de apresentar dados, MAS não oferece marcação significativa do conteúdo desses dados!!!
- Muitos problemas para busca e recuperação de informações



```
<HTML>
```

```
<!-- Here's a good place to put a comment. -->
```



```
<HEAD>
```

```
<TITLE>Structural Example</TITLE>
```

```
</HEAD>
```



```
<BODY>
```

```
<H1>First Header</H1>
```

```
<P>This is a paragraph in the example HTML file.  
Keep in mind that the title does not appear in the  
document text, but that the header (defined by H1)  
does.</P>
```

```
<OL> <LI>First item in an ordered list.
```

```
    <LI>Second item in an ordered list.
```

```
    <LI>Third item in an ordered list.
```

```
</OL>
```

```
<P>This is an additional paragraph. Technically,  
end tags are not required for paragraphs, although  
they are allowed. You can include character  
highlighting in a paragraph. <EM>This sentence of the  
paragraph is emphasized.</EM> Note that the &lt;/P&gt;  
end tag has been omitted.
```

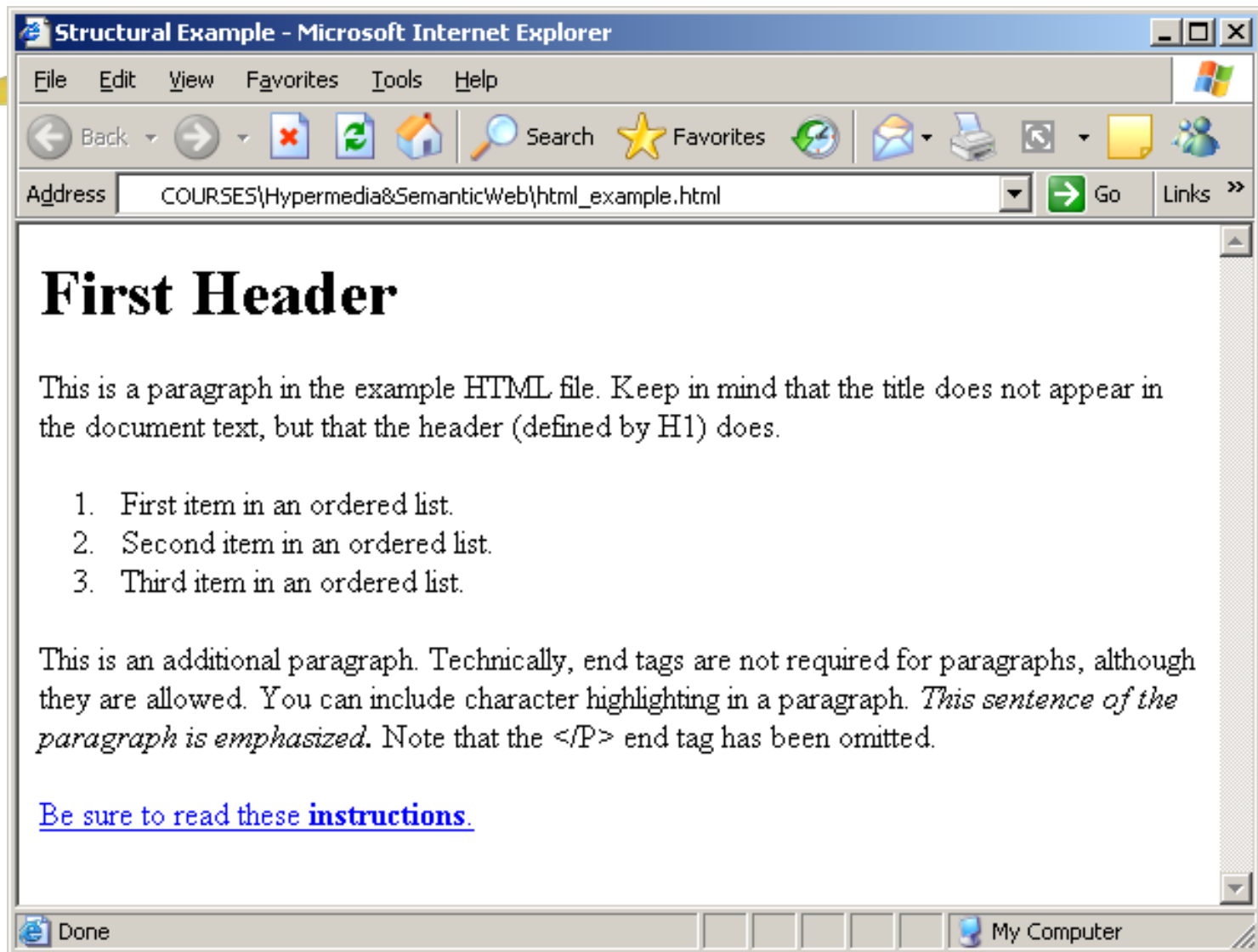
```
<P> <A HREF="instructions.html"> Be sure to read  
these <b>instructions</b>.</A>
```



```
</BODY>
```



```
</HTML>
```



Markup Language - HTML

HTML

Define uma classe simples de documentos

- › Cabeçalhos, parágrafos, listas, tabelas e imagens

Suporte mínimo a hipertexto

- › Ligação unidirecional especificada integralmente dentro do documento

Outras limitações

- A linguagem não é extensível
- Um documento não pode ser reutilizado
- Pouca, ou quase nenhuma semântica pode ser extraída de um documento

Markup Language - SGML

O que é SGML?

SGML

Standard Generalized Markup Language é uma metalinguagem através da qual se pode definir linguagens de marcação.

Linguagem bastante extensa e complexa - dificulta a criação de aplicações (*tools*) no ambiente Web

HTML é uma aplicação da SGML

Vantagens:

- Flexibilidade
- Definição dos elementos necessários em cada aplicação
- Definição de novas aplicações quando necessárias

Desvantagem

- Custo do processamento no ambiente Web

Markup Language - XML

O que é XML?

XML

foi criada em 1996, em 1998 tornou-se uma recomendação de **W3C**

Extensible Markup Language

Um *application profile* (subconjunto) da SGML

Omite todas as partes opcionais da SGML

Omite diversas partes mais complexas e menos usadas da SGML

Vantagens:

- ✓ Facilidade
- ✓ Definir tipos de documentos
- ✓ Escrever programas para manipular os documentos

Markup Language - XML

O que é W3C?

W3C

World Wide Web Consortium é um consórcio de empresas de tecnologia, atualmente com cerca de 500 membros.

Fundado por Tim Berners-Lee em 1994 para levar a Web ao seu potencial máximo, por meio do desenvolvimento de protocolos comuns e fóruns abertos que promovem sua evolução e asseguram a sua interoperabilidade.

Desenvolve tecnologias denominadas **padrões da web** para a criação e a interpretação dos conteúdos para a Web.

É um dever de todo desenvolvedor Web respeitar e seguir os padrões do W3C, para não impor barreiras tecnológicas às pessoas, desestimulando e até mesmo impedindo o acesso a suas páginas.

Markup Language - XML

Recomendação W3C?

W3C

De acordo com o Documento de Processo do W3C, a evolução de uma especificação até que seja uma recomendação progride por 5 níveis de maturidade:

1. **Working Draft (WD)**
2. Last Call Working Draft
3. **Candidate Recommendation (CR)**
4. Proposed Recommendation (PR)
5. **W3C Recommendation (REC)**

Trechos da especificação da XML



“tornar **fácil** e direto o uso de SGML na Web:
fácil para definir tipos de documentos,
fácil para autoria e gerenciamento de documentos
definidos como SGML, e
fácil para transmitir e compartilhar na Web.”

*“an **extremely simple dialect of SGML** which is completely described in the XML Specification. The goal is to enable generic SGML to be served, received, and processed on the Web in the way that is now possible with HTML.”*

“For this reason, XML has been designed for ease of implementation, and for interoperability with both SGML and HTML.”

Markup Language: XML

- “termo” XML é utilizado para fazer referência a: um **padrão aberto** (uma *W3C recommendation*) que fornece:
 - Um formato de dados
 - Uma linguagem de modelagem de dados
 - O uso de dados formatados em XML em uma aplicação
 - Uma metalinguagem para criar linguagens de marcação
 - Um conjunto de recomendações e especificações associadas: ligação, estilo, transformação, consulta, APIs, etc.

XML



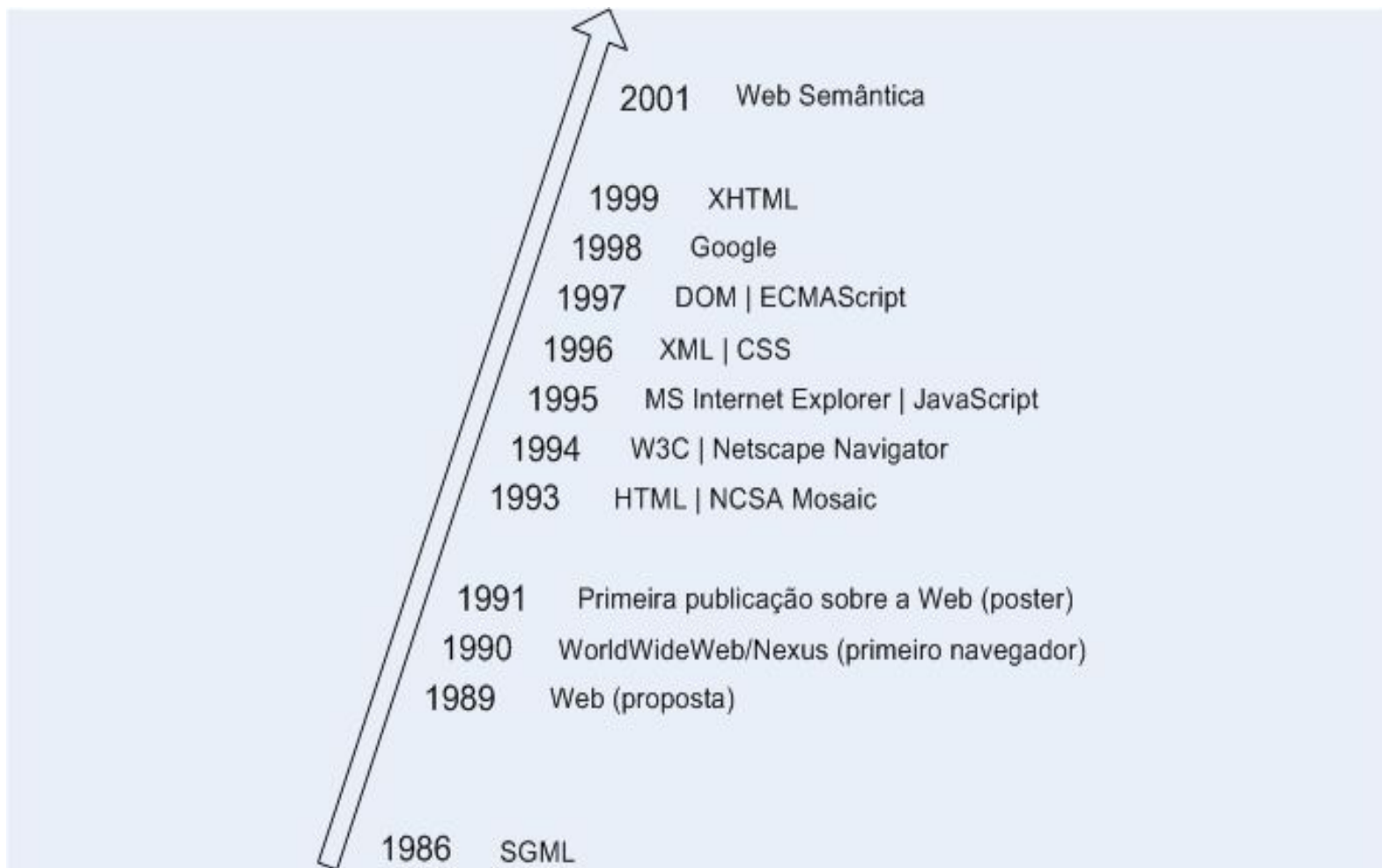
XML e suas tecnologias relacionadas oferecem uma arquitetura robusta para integração, manipulação, intercâmbio e apresentação de documentos

Baseada nas seguintes recomendações do W3C:

- XML 1.0
- XML *namespaces*
- APIs XML - DOM ou SAX
- XSL - XSLT - XPATH

XML

E o histórico das linguagens de marcação?



XML

Documentos XML

- › Dados do usuário ou aplicação
- › Um documento é uma redação que contém informação

Como a linguagem XML “vê” os dados?

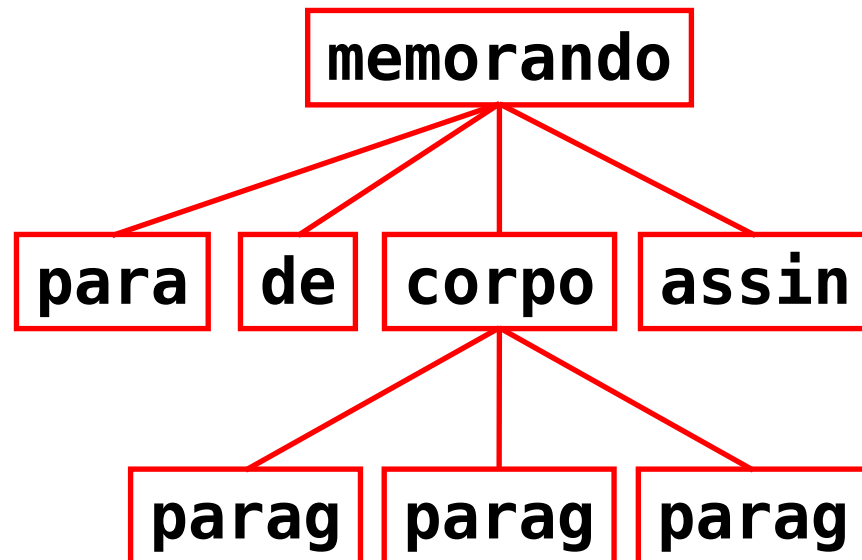
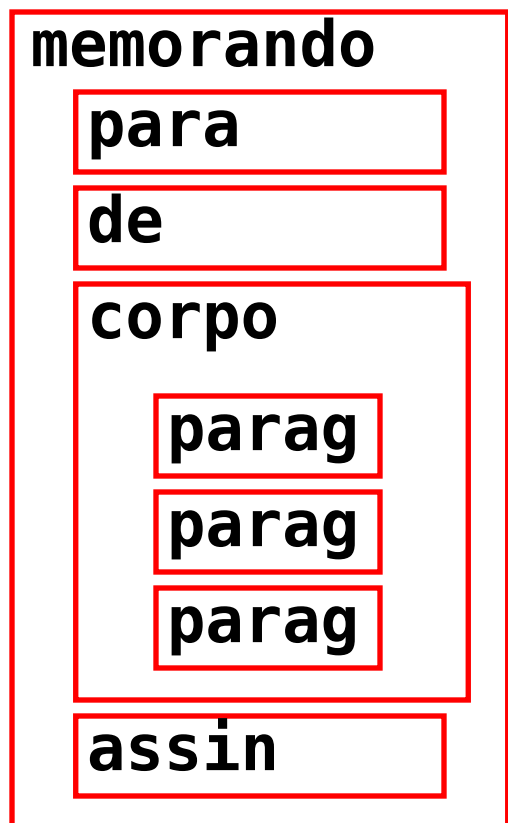
Documentos: compostos por elementos

Elementos: consistem de marcações (*tags*) e de conteúdo

- Nome do elemento
- Conteúdo: dados, outros elementos

XML

Elementos podem estar aninhados



XML

- **Elementos** podem ter **atributos**
- Localizados dentro das *start tags*
- Informação sobre os dados (metadados)
- Acrescentam informação ao elemento

```
<numeroTelefone tipo="fixo" uso="residencial" plano="basico"  
    instalacao="10-03-2008">16/3371-7777</numeroTelefone>
```

XML

XML é um **formato de dados**

Exemplo: registro de funcionário

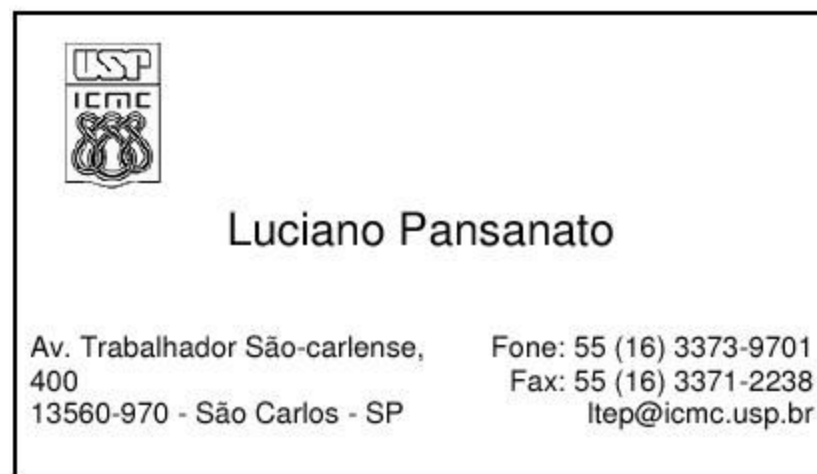
```
<registro-funcionario tipo="ativo" nfunc="15">  
<nome>  
<primeiro>Luciano</primeiro>  
<ultimo>Pansanato</ultimo></nome>  
<afiliacao>  
<titulo>Analista de sistemas</titulo>  
<empresa>ICMC-USP</empresa>  
<localizacao><cidade>São Carlos</cidade>  
<estado>SP</estado><cep>13560-970</cep></localizacao>  
<email>ltep@icmc.usp.br</email-name>  
</afiliacao>  
<peso unidade="Kg">70</peso><altura unidade="m">1,76</altura>  
</registro-funcionario>
```

XML

HTML



Aplicação impressa



XML

Base de dados

Key: 00095ABC

NFUNC: 015

001: Luciano


002: Pansanato

008: 1,76

014: 70

020: Analista de sistemas

FUNCIONÁRIO NOVO!



Luciano Pansanato
foi admitido recentemente no ICMC-USP
como Analista de sistemas

Bem-vindo à nossa equipe, Luciano!

Dados iguais, aplicações diferentes!

XML

Finalidade da XML

- “Codificar” (*mark up*) dados somente uma vez
- Gerar diversos produtos a partir dessa marcação
- Permitir busca complexa (semanticamente)
- Reusar dados (no todo ou em parte) diversas vezes
- Realizar intercâmbio de dados
- Permitir comunicação entre computadores

XML vs HTML

XML: marcação de conteúdos

HTML: marcação de apresentações

XML: permite elementos definidos pelo usuário

HTML: somente elementos pré-definidos

XML: exige validação

HTML: vale praticamente qualquer coisa

XML: orientada para dados

HTML: orientada para telas

XML: permite troca de dados entre aplicações

HTML: projetada para apresentação visual

XML: rigidamente definida e interpretada

HTML: fracamente interpretada

XML

XML não faz nada!

XML é um formato de dados

O software pode fazer muito com marcações

Tipos de marcações

→ descreve o conteúdo dos dados

Tipo de informação

→ descreve a estrutura do documento

Parte do documento

→ descreve com informação de valor agregado

Localização e navegação / Metadados

→ descreve com informação de apresentação e formatação

XML

XML é uma metalinguagem

Uma “linguagem” XML: conjunto de tags

- › Domínios: jornal, livro texto

Aplicações comuns: comércio eletrônico, gerência de conteúdo

Coleções de informação: biografias, leis, dicionários

Exemplos

- ♦ **CML** - *Chemical Markup Language* <http://www.xml-cml.org/>
- ♦ **MathML** <http://www.w3.org/Math/>
- ♦ **SVG** - *Scalable Vector Graphics*
<http://www.w3.org/Graphics/SVG/>

XML

Componentes lógicos de uma aplicação XML

- Documento XML = documento de texto Unicode
- DTD ou XML Schema
- Especificações de saída
- Transformações

XML

Documento XML

As tags (marcações) e o texto (conteúdo)

Dois tipos de documentos

- **Bem-formatado** (*well-formed*):
- **Válido** (*valid*): **em conformidade com o modelo**

Criação de um documento XML

Autoria: usando um editor XML

Programação, a partir de:

Uma base de dados

Um outro documento XML (transformação)

Um outro formato, por exemplo, ms-word (conversão)

XML

Modelo de Documento

DTD (**D**ocument **T**ype **D**efinition)

- Mecanismo de modelagem da especificação XML
- Modela um tipo/classe de documento
- Conjunto de regras que descrevem como os documentos desse tipo/classe podem ser formados
- Escrito na sintaxe formal da XML

DTDs especificam regras que definem quais as tags que podem ser usadas em um documento XML e quais os valores válidos.

Por que usar um modelo de documento?

Para compartilhar a informação, deve-se compartilhar o modelo de documento

XML

Especificações de saída

Stylesheet

- Separação de conteúdo do formato (de apresentação) e comportamento
- Define uma aparência ou comportamento
 - Para cada elemento
 - Em cada um dos contextos do documento

Uma *stylesheet*, diversos documentos

Manter consistência do formato ao longo dos documentos

É fácil desenvolver, manter e aplicar

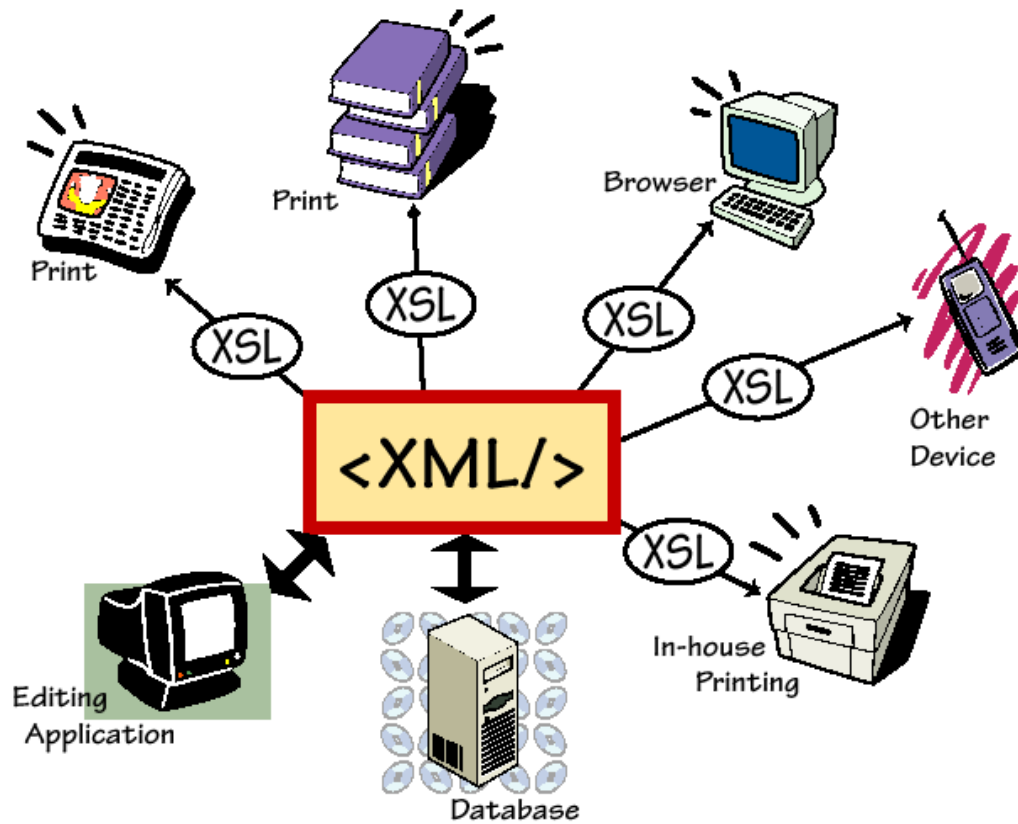
Um documento, diversas *stylesheets*

Criar documentos para tipos de mídia diferentes: impressa, on-line, etc.

É fácil produzir documentos derivados

XML

Transformações



XML

Transformações

- utiliza-se Folhas de Estilo = conjunto de regras para formatar OU transformar as informações de um documento XML
- **CSS** (*Cascading Style Sheets*)
 - transformações visando apresentação visual
 - aplicação do estilo em tempo de execução no cliente
- **XSLT** (*eXtensible Stylesheet Language*)
 - transformação em texto, HTML ou outro formato
 - aplicação em tempo real ou prévia (servidor)

XML

Exemplo

Documento XML

```
<film code="A010" type="DVD">  
  <title>The Lord of the Rings - The Two Towers</title>  
  <director>Peter Jackson</director>  
  <actor>Elijah Wood</actor>  
  <actor>Ian McKellen</actor>  
</film>
```

DTD

```
<!ELEMENT film (title, director, actor+)>  
<!ATTLIST film code ID #REQUIRED>  
<!ATTLIST film type (DVD|VHS) #REQUIRED>  
<!ELEMENT title (#PCDATA)>  
<!ELEMENT director (#PCDATA)>  
<!ELEMENT actor (#PCDATA)>
```

XML

Outro exemplo

Documento XML

```
<CATALOGO>
  <FILME>
    <TITULO tipo="DVD">Indiana Jones</TITULO>
    <GENERO>Aventura</GENERO>
    <DISTRIBUIDORA>CIC</DISTRIBUIDORA>
    <ANO>1989</ANO>
    <PRECO>14.28</PRECO>
  </FILME>
  <FILME>
    <TITULO tipo="VHS">Dinossauros</TITULO>
    <GENERO>Infantil</GENERO>
    <DISTRIBUIDORA>Walt Disney</DISTRIBUIDORA>
    <ANO>2001</ANO>
    <PRECO>16.90</PRECO>
  </FILME>
</CATALOGO>
```

Sintaxe XML

Notação básica

Um documento XML é composto por **tags** (marcações) e **texto** (conteúdo)

As tags identificam o **elemento** e seu conteúdo

Atributos de elemento

Aninhamento de elementos

Elemento

Atributo

Referência a entidade

Seção CDATA

➤ Character Data

Comentários

Instrução de processamento

Declaração XML

Sintaxe XML

Elemento

Cada elemento tem um nome, e pode ter um conjunto de atributos

Elemento vazio

```
<director>Peter Jackson</director>
```

```
<br/>
```

```
<version status="draft" target="private"/>
```

Um documento XML deve ter um elemento raiz

que contém todos os outros elementos

>> Bem-formatado (*well-formed*)

Sintaxe XML

Um documento XML é **bem-formado** (*well-formed*) se:

- possui um só elemento raiz que inclui todos os demais,
- todos os elementos devem estar devidamente aninhados
- se presente, o que é recomendado, a instrução de processamento declaração XML deve literalmente iniciar o documento. Essa declaração pode indicar a versão de *character set* utilizada
- todos os elementos não vazios devem ter *start-tag* e *end-tag*
- atributos não se repetem num mesmo elemento
- todas as entidades analisadas, referenciadas direta ou indiretamente, devem ser também entidades bem-formadas
- A recomendação é explícita em proibir que *parsers* XML tentem corrigir e interpretar documentos que não sejam bem formados ³⁷

Sintaxe XML

Atributo

Um par (nome, valor) presente na *start tag* do elemento, logo após o seu nome

```
<publication volume="2" number="4"  
  pages="29-66">World Wide Web Journal</publication>
```

- Os valores dos atributos devem estar entre aspas
- Um atributo não pode aparecer mais de uma vez no mesmo elemento

Sintaxe XML

Referência a entidade

Formato: **&nome_entidade;**

Entidade interna e externa

```
<!ENTITY www "World Wide Web">
<!ENTITY filmes SYSTEM "filmes.xml">

<publication volume="2" number="4"
  pages="29-66">&www; Journal</publication>

<description>Este &eacute; um teste
  de descri&ccedil;&atilde;o.</description>

<catalogo>
  &filmes;
  &discos;
  &livros;
</catalogo>
```

Sintaxe XML

Seção CDATA

Permite a inclusão de trechos que devem ser interpretados como caracteres e não como elementos de marcação

Seções CDATA não podem ser aninhadas

```
<![CDATA[  
*p = &q;  b = (i <= 3);  
]]>
```

&	&amp;
<	&lt;
>	&gt;
'	&apos;
=	&quot;

Sintaxe XML

Comentários

Não fazem parte do conteúdo do documento

```
<!-- falta pouco ... -->
```

Sintaxe XML

Instrução de processamento

Os *parsers* XML apenas repassam essas informações para a aplicação

Um documento XML deve iniciar com uma **declaração XML**

Bem-formatado (*well-formed*)

```
<?xml version="1.0"?>
```

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<?xml-stylesheet type="text/xml" href="5-2.xsl"?>
```

Padrão de codificação definido pela ISO consiste de 191 caracteres, contendo os do alfabeto latino

Parser é o programa de computador que serve para analisar a estrutura gramatical de um input, manipulando os *tokens*, que são segmentos de texto ou símbolos que podem ser manipulados.

Sistemas interativos Web



Sobre a Definição de Tipo do Documento
DTD

DTD

Document Type Definition

Corresponde à declaração que define a gramática do documento

Palavras reservadas: *devem aparecer sempre em letra maiúscula*

- **ELEMENT**
- **ATTLIST**
- **CDATA**
- **PCDATA**
- **NOTATION**
- **SYSTEM**

DTD

Document Type Definition

ELEMENT

ATTLIST

PCDATA

CDATA

NOTATION

SYSTEM

os nomes de elementos, atributos, entidades e notações devem começar com uma letra, underscore (“_”) ou dois-pontos (“:”), seguido de letras, dígitos ou dos sinais “_ - . :”, mas não podem começar com as letras “xml”

DTD

Declaração de Tipo de Documento

Indica que o documento deve seguir um DTD

DTDs internos

```
<?xml version="1.0" encoding="ISO-8859-1"?>  
<!DOCTYPE HELLO [ <!ELEMENT hello (#PCDATA)>]>  
<hello>  
  Olá XML!  
</hello>
```

DTD

Declaração de Tipo de Documento

Indica que o documento deve seguir um DTD

DTDs externos

```
<?xml version="1.0" encoding="ISO-8859-1"?>  
<!DOCTYPE HELLO SYSTEM "hello.dtd">
```

```
<hello>  
  Olá XML!  
</hello>
```

DTD

Declaração de elementos

Identifica o nome do elemento e a natureza de seu conteúdo

#PCDATA

➤ *Parseable Character Data*

EMPTY

Operadores

,	seqüência (and)
	escolha (ou)
?	opcional
*	zero ou mais filhos
+	um ou mais filhos
()	agrupamento

```
<!ELEMENT thesis (abstract, chapter+)>  
<!ELEMENT abstract (#PCDATA)>  
<!ELEMENT chapter (#PCDATA)>
```


DTD

Declaração de atributos

Valor

Nome do atributo, tipo de dados e valor padrão

<code>#REQUIRED</code>	obrigatório
<code>#IMPLIED</code>	opcional
<code>#FIXED < value ></code>	fixo
<code><default value ></code>	padrão

Tipos de dados:

CDATA	Texto (Character Data)
ID	Um nome exclusivo no documento
IDREF	Uma referência para outro elemento
IDREFS	Diversas referências separadas por espaços
ENTITY	Uma entidade declarada
ENTITIES	Diversas entidades separadas por espaços
NMTOKEN	Um XML name token
NMTOKENS	Diversos XML name tokens
NOTATION	Um nome de notação declarado

```
<!ATTLIST thesis author CDATA #REQUIRED>
```

DTD

Declaração de entidades

Interna

- › Associa uma seqüência de caracteres a um nome

Externa

- › Permite a inclusão de um arquivo no documento

```
<!ENTITY W3C "World Wide Web Consortium">  
<!ENTITY chapter1 SYSTEM "chapter1.xml">
```

DTD

Declaração de notação

Identifica, tipicamente, tipos de dados binários externos

```
<!DOCTYPE image [  
  <!NOTATION jpeg SYSTEM "http://www.example.org/jpeg">  
  <!NOTATION png SYSTEM "http://www.example.org/png">  
  <!ENTITY mypic SYSTEM "foo.jpg" NDATA jpeg>  
  <!ELEMENT image (#PCDATA)*>  
  <!ATTLIST image src ENTITY (jpeg|png) #REQUIRED>  
>
```

```
<image src="mypic">my picture</image>
```

Não use notação!

DTD

Limitações

A verificação de que os valores contidos no documento correspondem ao esperado é responsabilidade da aplicação

Por exemplo, não existem mecanismos para especificar que:

- Um elemento <data> deve conter apenas valores válidos para dia-mês-ano
- Valores de <custo> devem conter apenas valores positivos em alguma moeda corrente
- Um elemento <titulo> deve conter entre 1 e 255 caracteres

DTD

Limitações

Utiliza uma sintaxe especial que não é parecida com a sintaxe XML usual

Extensibilidade e escalabilidade

Compatibilidade com namespaces

Outras limitações, ??

Documentos XML válidos



- Seguem regras estabelecidas por **gramáticas**, como DTDs e Esquemas XML
- Documentos XML podem estar, **opcionalmente**, associados a uma gramática que rege sua estrutura e composição

Bem-formados vs. Válidos

- Aplicação que utilize documentos XML **DEVE**:
 - processar os documentos;
 - verificar se seu conteúdo está **de acordo** com as regras de formação de um documento XML, em geral;
 - se for o caso, validar sua **estrutura** e **conteúdo** frente à gramática correspondente definida no DTD ou Esquema XML
- Aplicação **DEVE** implementar (ou reusar) as funções de verificação e validação para XML

Parsers XML



- Validadores para DTD (*VALIDATOR*)
 - Aelfred – <http://saxon.sourceforge.net/aelfred.html>
 - Expat (James Clark em C) – <http://expat.sourceforge.net/>
 - MSXML 2.5 (IE 5.5) e MSXML 3.0, 4.0 (IE 6.0)
 - Xerces-J – <http://xerces.apache.org/xerces-j/>
- Não-validadores
 - XP (James Clark em Java)
 - MSXML 2.0 (IE 5.0)
- O que eles fazem??? – http://www.xml.com/pub/rg/XML_Parsers
 - *TAREFA!!*

XML Schema

XML Schema

Suporte a tipo de dados

Range checking

Validação baseada em URIs de namespace, em vez de prefixos

Extensibilidade e escalabilidade

XML Schema em mais detalhes, **mais tarde** no curso

Schemas não substituem DTDs!

Aplicações de XML

XML é ótima solução para compartilhar dados. Para implementar soluções em gestão de informações usando XML, pode-se usar: **DTD** ou **XSchema** para especificar o modelo de dados e validar as informações

As **APIs DOM** ou **SAX** para extrair dados dos documentos, gerar documentos, ler e gravar em bancos de dados

XSLT e **XPath** para transformar os dados em outros formatos

XLink, **XPointer** e **XQuery** para criar vínculos lógicos entre os documentos e localizar seus componentes

XSL-FO ou **XHTML** para formatar os dados para impressão ou visualização na tela (PDF, Word ou Web)

SVG para gerar informações em forma de gráfico vetorial

Aplicações de XML



Exemplos de aplicações

XML para publicação impressa e na Web

XML entre camadas da aplicação

XML para intercâmbio de informação

Aplicações



XML para publicação impressa e na Web

Diversas saídas diferentes, uma fonte gerenciável

- Diversos tipos de mídia/equipamentos (Web, CD-ROM, PDAs, sintetizador de voz)
- Diversos estilos de impressão/apresentação

Hardware, software, SO diferentes para entrada, manipulação, apresentação

Publicação sob demanda/saída personalizada

Aplicações

XML entre camadas da aplicação

Modelo de sistema “*three-tier*”:

- Camada de apresentação/Interface de usuário
- Camada de processamento ou “lógica do negócio”
- Camada de armazenamento

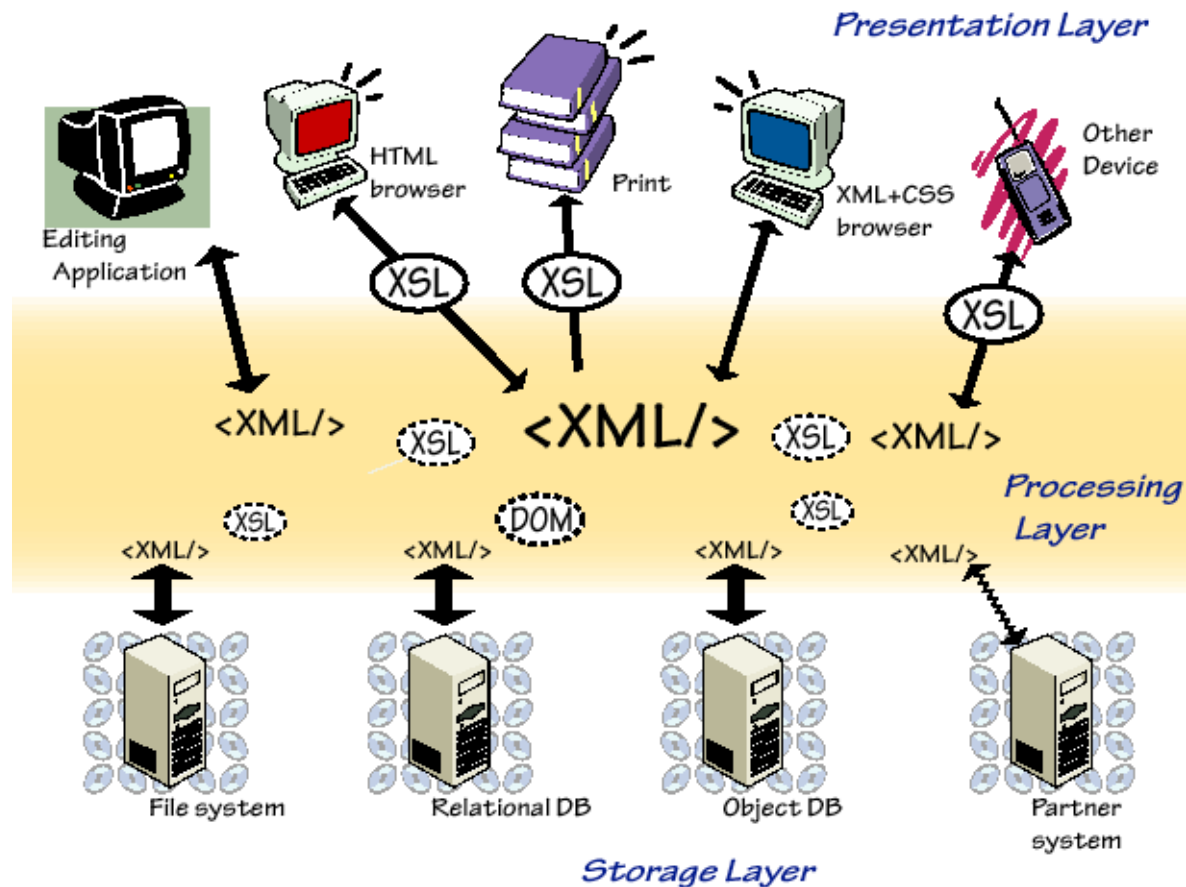
XML pode ser usada em qualquer uma das três camadas, principalmente na camada do meio

XSL (*Extensible Stylesheet Language*) pode ser usada para qualquer processamento

- Dentro da camada do meio, e entre as camadas

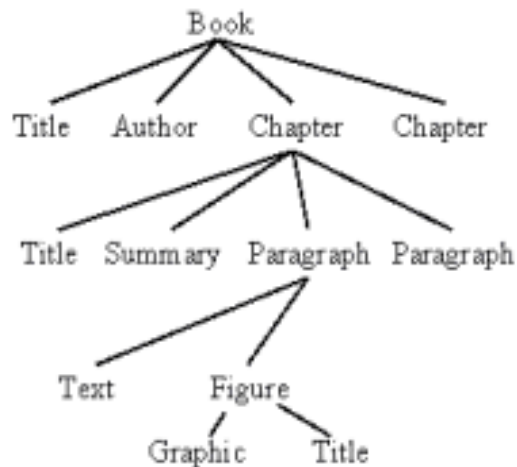
Aplicações

Modelo “three-tier” típico

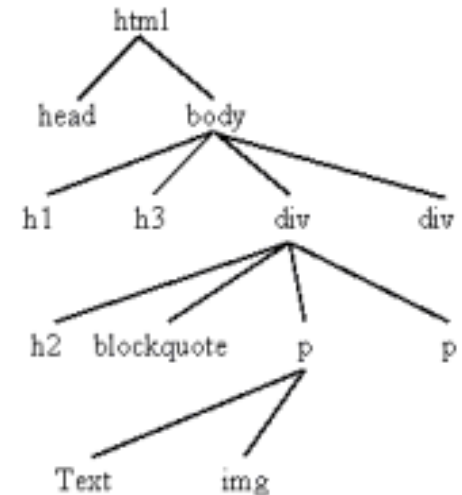


Aplicações

XML Source Tree



XHTML Result Tree



```
<html>
<head>...</head>
<body>
<h1></h1>
<h3></h3>
.....
</body>
</html>
```

Aplicações

XML para intercâmbio de informação

- Entre “*data aggregators*”
(websites de jornais e de comunidades científicas)
- Através do ciclo de vida de um produto (entre divisões)
- Transferência direta entre computadores

- Entre formatos proprietários
- Entre parceiros de negócios
(transações de comércio eletrônico B2B, B2C,
substituição dos formatos proprietários de EDI)

XML: dimensões e processamento

XML

Dimensões de estrutura e conteúdo

Documentos bem formados!

Outras dimensões de um documento XML

Apresentação: CSS, XSL

Estrutura e Semântica: DTDs e XML Schemas

Metadados e mais semântica: RDF

Estrutura de hipertexto: XLink e XPointer

Processamento de documentos XML

Parsers, APIs, DOM...

*nossas
aplicações...*

Referências

W3C (2004) Extensible Markup Language (XML) 1.0 (Third Edition). W3C Recommendation. <http://www.w3.org/TR/REC-xml>

Extensible Markup Language (XML).
<http://www.w3.org/XML/>

Cover Pages: Extensible Markup Language (XML).
<http://xml.coverpages.org/xml.html>

The XML FAQ v4.0 (2005-01-01)
<http://www.ucc.ie/xml/>

Harold, E. R. (2001) XML Bible. Second Edition, Hungry Minds.

Slides inicialmente elaborados por Luciano T. E. Pansanato

Tarefa:



- Re-escreva a página HTML (sua grade-horária), a partir daquela previamente construída, em **XML**.
- Escreva um **DTD** para esta sua grade horária em XML.
- Utilize dois validadores a partir da lista vista em sala de aula e apresente um relatório sobre os resultados da nova página.

Resumo



Visto:

Markup Languages

SGML

HTML

XML e Sintaxe XML

DTD