

# Porque ?



Se eu fosse um extraterrestre grato ...

# Era da informação

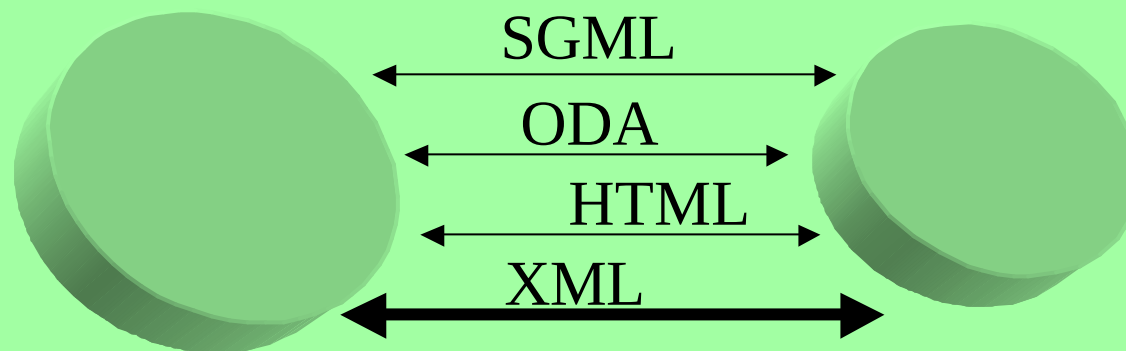
- Analistas estimam que organizações gastam por volta de 20% de sua renda bruta manipulando informações (armazenando e distribuindo digitalmente ou em papel)
- A maioria é organizada em documentos

# Padrões

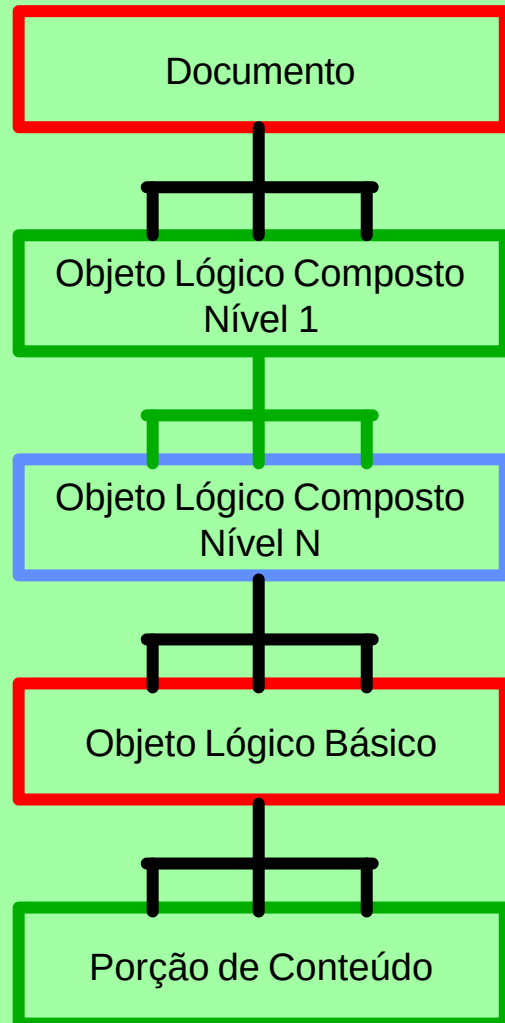
Padrões são necessários para permitir :

- ✓ O compartilhamento de informações entre ambientes heterogêneos.
- ✓ A reutilização destas informações.
- ✓ Empacotamento.

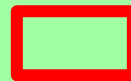
# Intercâmbio de Informações



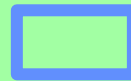
# Modelo ODA



## Legenda



Objeto Obrigatório



Objeto Opcional



Porção de Conteúdo Opcional



Possibilidade de Muitos Subordinados



Possibilidade de Níveis Intermediários com Múltiplos Subordinados

# ODA

- **NF ISO 8613**
  - Tratamento da informação
  - Escritório (“Bureautique”)
  - Arquitetura dos documentos de escritório
  - Formato de troca
- **Oito Partes:**
  - 1 – Introdução e princípios gerais
  - 2 – Estruturas do documento
  - 3 – Não foi usado
  - 4 – Perfil do documento
  - 5 – Formato de troca
  - 6 – Arquit. Conteúdo de caracteres
  - 7 – Arquit. Conteúdo gráfico em pontos
  - 8 – Arquit. Conteúdo gráfico geométrico

## **Formato de Representação de um documento ODIF**

- **Imagem**
- **Processável (edita conteúdo)**
- **Formatável (edita o lay-out de apresentação)**
- **Processável - Formatável**

# ODA - Office Document Architecture

- **Níveis de conformidades**

- Formatado (Formato Imagem)
- Processável (conteúdo e estrutura lógica)
- processável-formatável (conteúdo, estruturas lógicas e de apresentação)

|                           |                       |                                                                                                                                                                                                                                                   |
|---------------------------|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Mixed Mode</b><br>T.73 | <b>Formato Imagem</b> | <b>MIF 2</b> Estrutura Completa de Apresentação<br>Definições Opcionais de Apresentação<br>Todos CF para Caracteres. Imagens + T.6-CF<br>Data Stream B, Referências a atributos                                                                   |
| <b>G.4-FAX</b><br>T.6     |                       | <b>MIF 1</b> Estrutura Restrita de Apresentação<br>Definições Opcionais de Apresentação<br>T.61-CF + T6-CF, Data Stream B<br>Nenhuma Referência a atributos<br><br><b>TIF 0</b> Páginas; T.6-CF; Data Stream B;<br>Nenhuma Referência a atributos |



## ***Office Document Architecture - ODA***

Permite a descrição de documentos visando facilitar a sua compreensão.

- ✧ **Documento**

Agregado de texto que pode ser processado ou trocado como uma unidade.

- ✧ **Texto**

Informação para a compreensão humana. Pode conter elementos gráficos, geométricos ou fotográficos.

- ✧ **Conteúdo**

Parte da informação contida no documento que é independente de estrutura ou forma de apresentação.

## Estrutura Hierárquica

Documentos em ODA são compostos por duas estruturas:

- ✓ **Lógica**

Responsável pela divisão e subdivisão do documento em itens que tenham significado para o autor e o leitor como por exemplo seções e parágrafos.

- ✓ **Apresentação**

Responsável pela visualização do documento. Além de apresentar o documento permite que sua estrutura lógica seja destacada.

Ambas as estruturas podem ser representadas em forma de árvore onde as folhas são o conteúdo do documento.

# **Estrutura Hierárquica**

**Documentos em ODA são compostos por duas estruturas:**

❖ **Lógica**

**Responsável pela divisão e subdivisão do documento em itens que tenham significado para o autor e o leitor como por exemplo seções e parágrafos.**

# Estrutura Hierárquica

## ❖ Apresentação

**Responsável pela visualização do documento.  
Além de apresentar o documento permite que sua  
estrutura lógica seja destacada.**

**Ambas as estruturas podem ser representadas  
em forma de árvore onde as folhas são o  
conteúdo do documento.**

# Objetos Básicos e Compostos

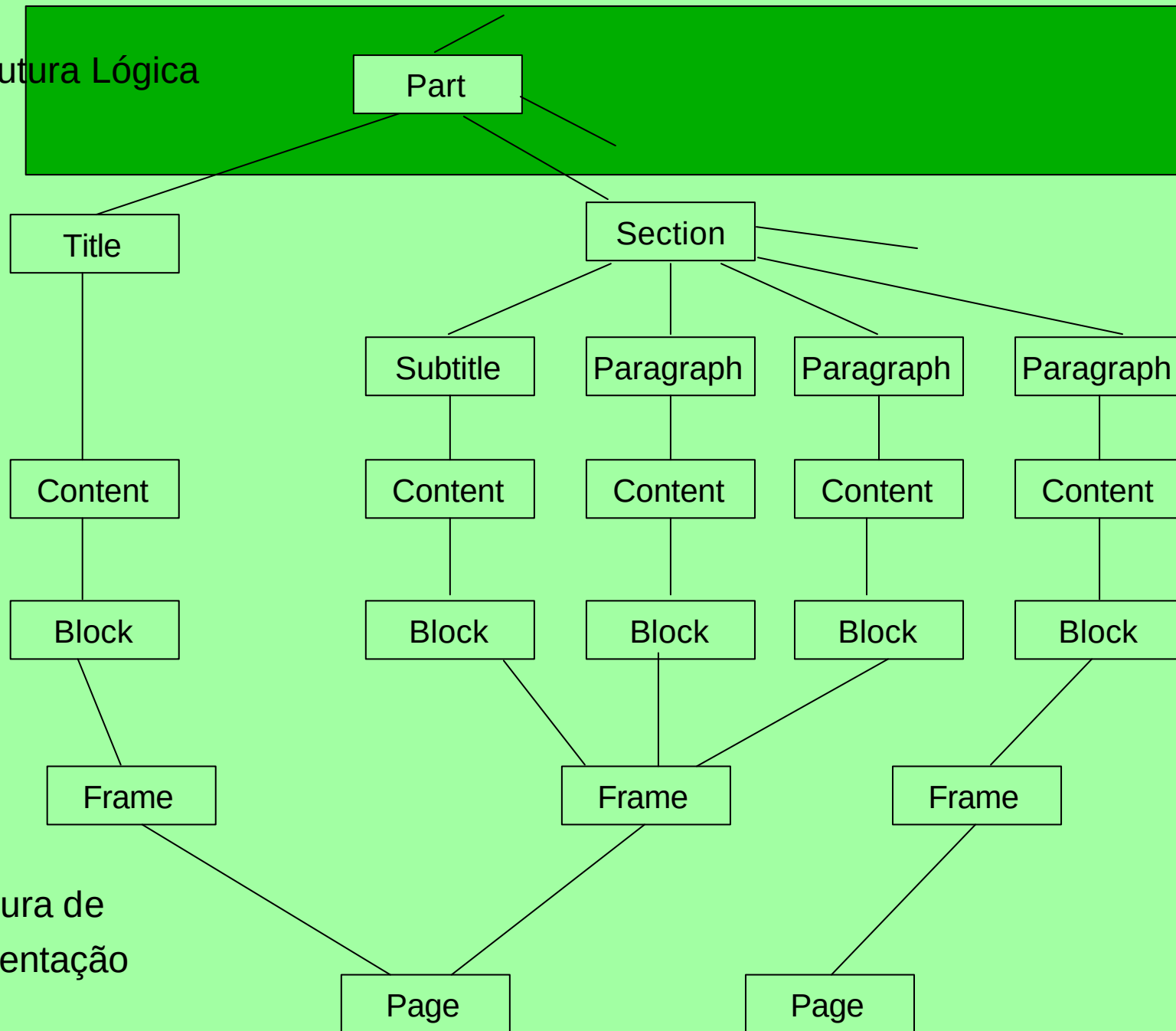
- **Objetos Básicos**

**Objetos que estão diretamente relacionados a porções de conteúdo.**

- **Objetos Compostos**

**Constituídos por objetos básicos ou por objetos compostos.**

Estrutura Lógica



Estrutura de  
Apresentação

Page

Page

# Objetos Básicos e Compostos

## ⇒ **Objetos Básicos**

Objetos que estão diretamente relacionados a porções de conteúdo.

## ⇒ **Objetos Compostos**

Constituídos por objetos básicos ou por objetos compostos.

Estrutura Lógica



Part

★ Objetos Lógicos Compostos

▀ Objetos Lógicos Básicos



Title



Section



Subtitle



Paragraph



Paragraph



Paragraph

Content

Content

Content

Content

Content

Block

Block

Block

Block

Block

Frame

Frame

Frame

Estrutura de  
Apresentação

Page

Page

Valdeni



Estrutura Lógica

Part

★ Objetos de Apresentação Compostos

➡ Objetos de Apresentação Básicos

Title

Section

Subtitle

Paragraph

Paragraph

Paragraph

Content

Content

Content

Content

Content

➡ Block

➡ Block

➡ Block

➡ Block

➡ Block

★

Frame

★

Frame

★

Frame

Estrutura de  
Apresentação

★

Page

★

Page

Valdeni

# Construtores

Especificam a ordem em que subobjetos devem formar um objeto.

- **Seqüencial**

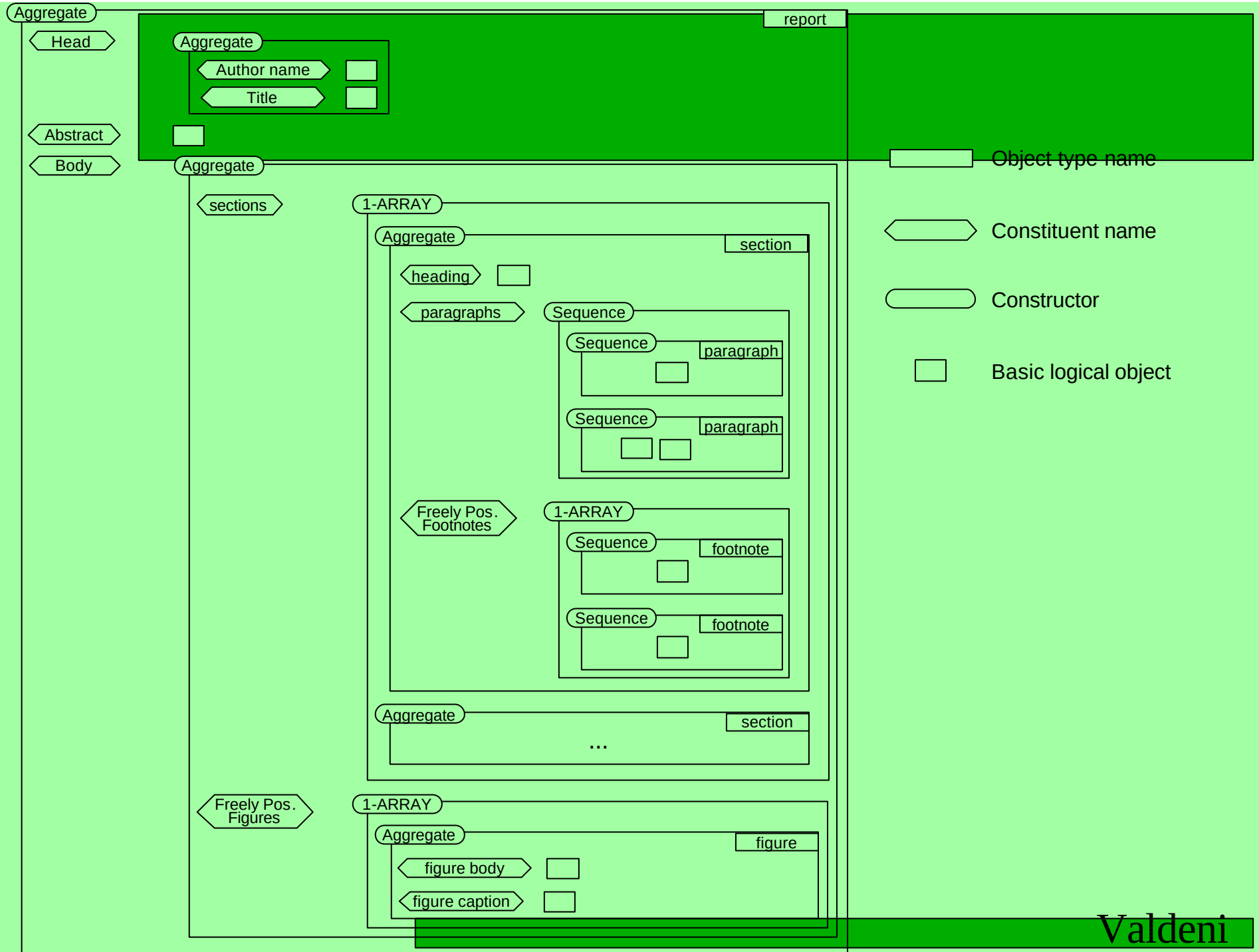
Especifica uma ordem seqüencial para os objetos que constituem um objeto.

- **Vetor**

Especifica uma ordem unidimensional ou bi-dimensional para os elementos de um objeto.

- **Agregado**

Especificam uma sequencia ou uma organização não particular de elementos que constituem um objeto composto.



## Referências

- **SGML(Standard Generalized Markup Language)**
  - ISO 8879 : 1986
  - NF EM 28879 : 1990
  - ISSO 9069 : 1988 (SDIF)
  - NF EM 29735 : 1990 (SDIF)
- **ODA/ODIF (Office Document Architecture / Office Document Interchange Format)**
- **CCITT T.73: 1984 (recomendação)**
- **ECMA 101:1985 (norma)**
- **NF ISO 8613: 1991**
- **TEI – Text Encoding and Interchange**

## ***Standard Generalized Markup Language - SGML***

### ✓ **Markup**

Termo utilizado por profissionais de editoração para referirem-se a marcas no texto a ser entregue para a gráfica.

### ✓ **Markup Language**

Conjunto de regras que permitem a especificação de como o texto deve ser processado.

### ✓ **Generalized Markup Language**

Linguagem que especifica a maneira como o texto é formado e não a maneira como ele deve ser exibido.

Através de SGML é informado ao sistema qual a informação existente e como ela está estruturada e como deve ser apresentada.

## Estrutura de um documento em SGML

- ♦ **Declaração**

Especifica o conjunto de caracteres do documento. Por exemplo os caracteres que delimitam as “marcas” são “<” e “>”.

- ♦ **Document Type Declaration**

Define as regras para a estruturação da informação mas não especifica como a mesma deve ser processada.

- ♦ **Document Instance**

É o documento propriamente dito. Geralmente um documento em SGML começa com o seu nome entre sinais “<” e “>”.

Exemplo:

`<trabconclusao>`

## Exemplo de um DTD

<!-- ELEMENTOS      CONTEÚDO      -->

<!ELEMENT            Msg            (Dsts, Subj?, MsgBody?, Attac\*)>

<!ELEMENT            Dsts            (Dst+) >

<!ELEMENT            Dst            (#PCDATA) >

<!ELEMENT            Subj            (#PCDATA) >

<!ELEMENT            MsgBody        (#PCDATA) >

<!ELEMENT            Attac            (#PCDATA) >

A tarefa de quem projeta um DTD é identificar os elementos que constituem um documento e definir a hierarquia estrutural desses e de outros elementos.

## Indicadores de Ocorrência

| Indicador | Obrigatoriedade/Núm. de Ocorrências                 |
|-----------|-----------------------------------------------------|
| ?         | Opcional: 0 ou 1 ocorrências                        |
| +         | Obrigatória e Repetitiva: pelo menos uma ocorrência |
| *         | Opcional e Repetitiva: 0 ou mais vezes              |

## Conectores

| Conector | Significado                                |
|----------|--------------------------------------------|
| ,        | Devem ocorrer na ordem especificada        |
|          | Somente um elemento do grupo deve ocorrer  |
| &        | Todos devem ocorrer, mas em qualquer ordem |



## Exemplo de um documento em SGML

```
<Msg>
<Dsts> <Dst> _____ </Dst>
<Dst> _____ </Dst>
<Dst> _____ </Dst> </Dsts>
<Subj> _____ </Subj>
<MsgBody>_____
_____ </MsgBody>
<Attac> _____ </Attac>
<Attac> _____ </Attac>
<Attac> _____ </Attac> </Msg>
```

## Minimização de Marcas

- ✓ Omissão de Marcas
- ✓ Marcas Vazias
- ✓ Referências curtas

## Omissão de Marcas

<Msg>

<Dsts> <Dst> \_\_\_\_\_

<Dst> \_\_\_\_\_

<Dst> \_\_\_\_\_

<Subj> \_\_\_\_\_

<MsgBody>\_\_\_\_\_

\_\_\_\_\_

<Attac> \_\_\_\_\_

<Attac> \_\_\_\_\_

<Attac> \_\_\_\_\_ </Msg>

## Marcas Vazias

<Msg>

<Dsts> <Dst> \_\_\_\_\_ </Dst>

< > \_\_\_\_\_ </>

< > \_\_\_\_\_ </> </Dsts>

<Subj> \_\_\_\_\_ </Subj>

<MsgBody>\_\_\_\_\_

\_\_\_\_\_ </MsgBody>

<Attac> \_\_\_\_\_ </Attac>

< > \_\_\_\_\_ </>

< > \_\_\_\_\_ </> </Msg>

## Referências Curtas

<Msg>

<Dsts> E-mail do primeiro Destinatário

E-mail do segundo Destinatário  
\_\_\_\_\_

</Dsts>

<Subj> \_\_\_\_\_ </Subj>

<MsgBody>\_\_\_\_\_

\_\_\_\_\_ </MsgBody>

<Attac> \_\_\_\_\_ </Attac>

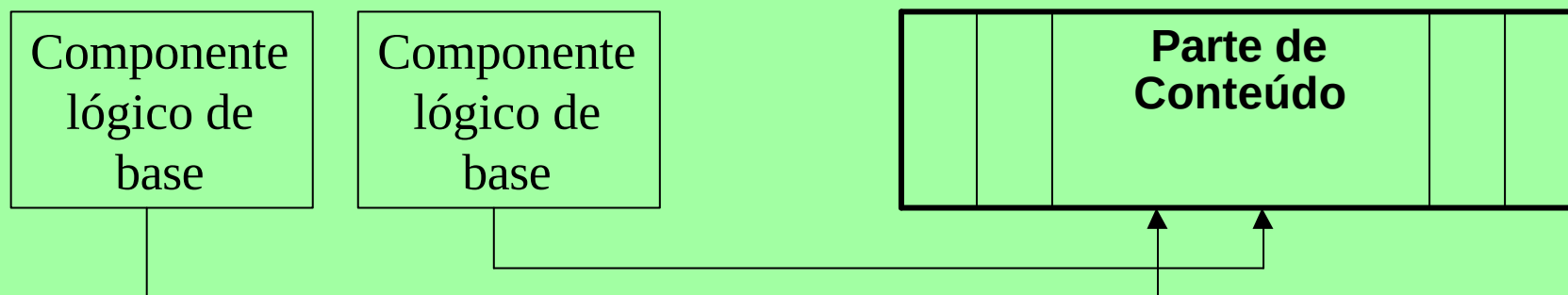
<Attac> \_\_\_\_\_ </Attac>

<Attac> \_\_\_\_\_ </Attac> </Msg>

# Estrutura Interna do SGML



# Estrutura Interna do ODA



## ***Hypertext Markup Language - HTML***

- ✦ É um *document type* de SGML com uma semântica genérica apropriada para a representação de informações de uma ampla gama de aplicações.
- ✦ Tem como objetivo permitir a criação de uma rede global de informações baseadas em hiperdocumentos onde os nós podem estar em diferentes continentes.
- ✦ Foi projetado para ser utilizado em WWW como um formato público para a distribuição de hiperdocumentos em redes de longo alcance.

# Hypertext Markup Language (HTML)

- É uma aplicação SGML
- O DTD HTML 4.01 é o último publicado (Mudanças dependem do W3C)
- Elementos começam com os tags de início `<tag>` e terminam pelos tags de fim `</tag>`
- Alguns elementos não requerem tags de fim : `<P>` `<IMG>`
- Outros não tem conteúdo : `<BR>` `<HR>`
- Os elementos são *insensitives*, logo as letras podem ser maiúsculas ou minúsculas



## Exemplo de um documento em HTML

```
<HTML>
<!-- ex001: Primeiro exemplo HTML -->
<HEAD>
<TITLE>Exemplo Documento HTML</TITLE>
</HEAD>
<BODY>
<h1 id="a1">Nível 1</h1>
<p>Este é texto normal. Você pode
enfatizar <em>partes </em> se
deseja.</p>
<p style = « fonte-size: 20pt; color:
#0000ff »>Novo parágrafo</p>
</BODY>
</HTML>
```

# Padrões TXT

- Visão ANSI com browser
- Visão UTF-8 com browser
- Visão UTF-8 + entidades com browser
- Visão unicode com browser
- Visão unicode big endian com browser

# Limitações do HTML

- **Perda da Extensibilidade**
  - Não pode criar novos *tags* ou atributos
  - Qualquer extensão deve ser aprovada pelo W3C
  - As extensões devem ser adotadas pelos vendedores de *browsers*
- **Perda de estruturação**
  - Sem mecanismos para especificar uma informação estruturada. Ex. Livros compostos por capítulos.
- **Perda de descrição dos dados**
  - Não pode incluir metadados
  - Sem mecanismos de inclusão de esquemas
- **Não suporta validação**
  - Ex. Falta do título do livro.

# Limitações do HTML

- Mistura conceitos de estruturação lógica com formatação

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<HTML version="4.01" encoding="ANSI">
<!-- ex001: Primeiro exemplo HTML -->
<HEAD>
<META http-equiv="Content-Type" content="text/html; charset=UTF-8">
<TITLE>Exemplo Documento HTML</TITLE>
</HEAD>
<BODY>
<h1 id="a1" style = "font-family: arial ">N&iacute;vel 1</h1>
<p>Este &eacute; texto normal. Voc&ecirc; pode enfatizar <em>partes
  </em> se deseja.</p>
<p style ="fonte-size: 40pt; color: #0000ff">Novo par&aacute;grafo</p>
</BODY>
</HTML>
```

[Ver no browser](#)

# XML

- *e***Extensible Markup Language**
- **Subset** do SGML
- 80% da funcionalidade do SGML com somente 20% de sua complexidade
- Padrão para descrever e trocar dados estruturados na Web
- Documento = Estrutura (XML) + Conteúdo (Usuário) + Estilo (XSL)

# Vantagens do XML

- **Extensibilidade**
  - Criação novos *tags* ou atributos
  - Reuso de um conjunto de *tags* através de *XML namespaces*
- **Estruturação**
  - Definição de estruturas aninhadas
  - Qualquer profundidade
- **Descrição dos dados**
  - Inclusão de metadados
  - Reuso de conjuntos de metadados através de *XML namespaces*
- **Validação**
  - Validade estrutural dos dados

# Documento XML

- Tem duas estruturas : Lógica e Física
- Estrutura física é composta de *entities*
- Estrutura lógica é composta de :
  - Prólogo
    - » Declarações
    - » Instruções de processamento (IPs)
  - Corpo
    - » Elementos (raiz e atributo, com texto, vazio, tag fim da raiz)
  - Epílogo
    - » Comentários
- Forma de uma IP `<?name ... ?>`  
Ex. `<?xml version="1.0" encoding="UTF-8"?>`

# Exemplo Documento XML

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE document [  
  <!ELEMENT document (titulo, nada)>  
  <!ELEMENT titulo (#PCDATA)>  
  <!ELEMENT nada EMPTY>  
  <!ATTLIST document  
    autor CDATA #IMPLIED  
    xmlns CDATA #FIXED http://www.exemplo.com/>  
<document autor="jósé">  
  <titulo>Título</titulo>  
  <nada/>  
</document>  
  
<!-- Exemplo simples de um documento XML -->  
Prólogo(Document Type Definition + IP) Corpo Epílogo  
Visualizar p/ browser IE
```



# Exemplo Documento XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE document [
  <!ELEMENT document (titulo, nada)>
  <!ELEMENT titulo (#PCDATA)>
  <!ELEMENT nada EMPTY>
  <!ATTLIST document
    autor CDATA #IMPLIED
    xmlns CDATA #FIXED http://www.exemplo.com/
]>
<?xml-stylesheet href="exemplo.css" type="text/css"?>
<document autor="jósé">
  <titulo>Título</titulo>
  <nada/>
</document>
```

<!-- Exemplo simples de um documento XML -->

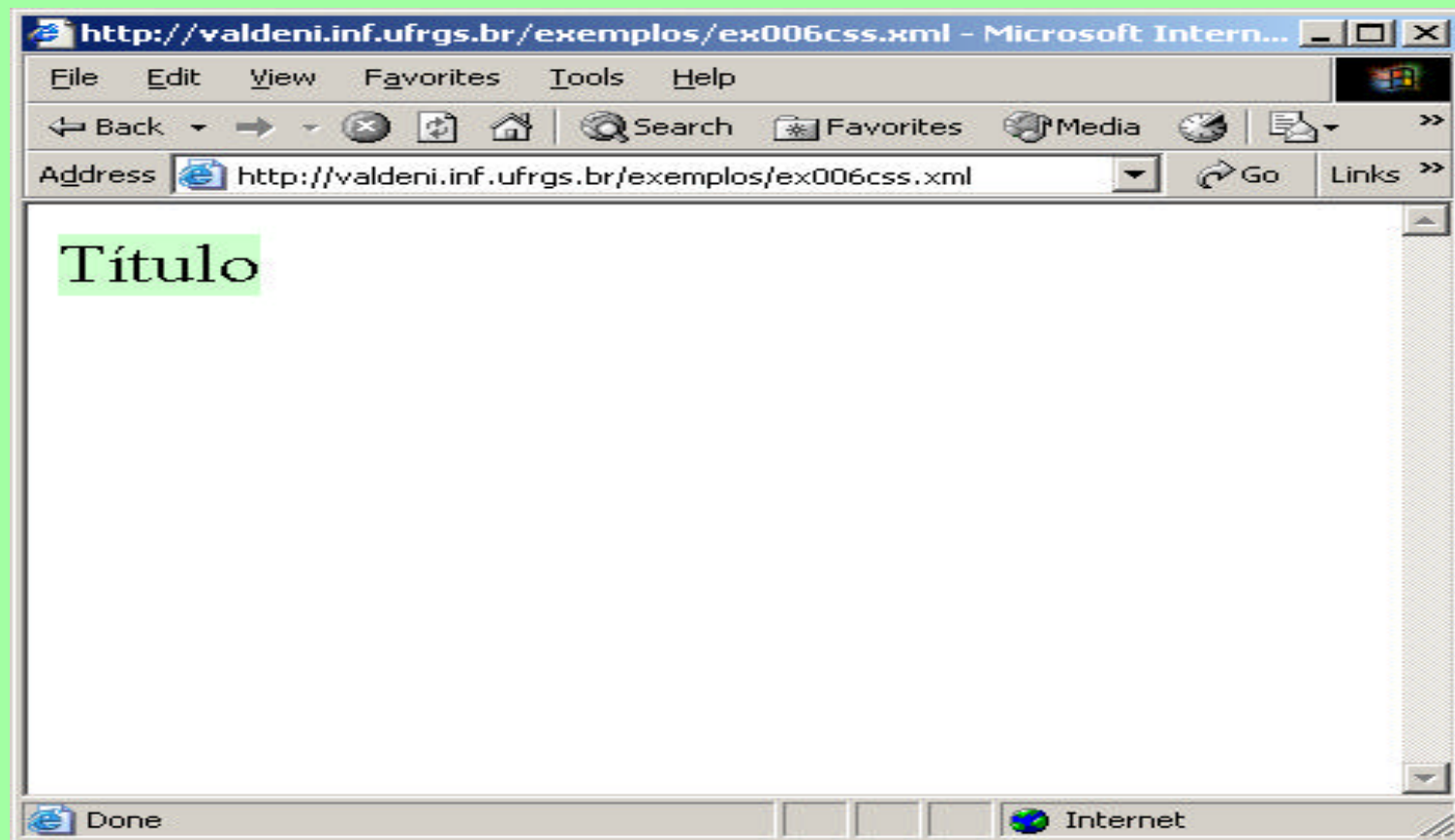
Prólogo(Document Type Definition + IP) Corpo Epílogo

Visualizar c/ CSS

# Exemplo.css

```
titulo {font-size: 20pt;  
        background-color: #ccffcc }
```

# Resultado



# *Document Type Definition*

- Contém declarações de tipo de documento ou aponta para recursos externos contendo declarações
- Ex. Documento XML com DTD :

```
<?xml version="1.0:?">  
<!DOCTYPE saudação [  
    <!ELEMENT saudação (#PCDATA)>  
<saudação>Alo, Pessoal</saudação>
```
- Documento XML apontando uma DTD :

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE saudação SYSTEM "Alo.dtd">  
<saudação>Alo, Pessoal</saudação>
```

# Tipo de Conteúdo de um Elemento

- **#PCDATA** : Parser Character Data –(<, >, &, etc. )
- **(#PCDATA | message)** : conteúdo misto
- **ANY** : Pode ser de qualquer tipo
- **EMPTY** : não contem caracteres nem filhos

## Seção CDATA

- **#CDATA** : Character Data – ([>])

# Tipos de Atributos e #DEFAULT

**<!ATTLIST X Y CDATA #REQUIRED>**

- **CDATA** : Tipo caracter – (<,>,&,',")
- **#REQUIRED** : Default indica que atributo deve está presente. Na falta não é doc válido.

**<ATTLIST X id IDREF #IMPLIED>**

- **IDREF** : Tipo “token” apontando para elementos com atributo ID
- **#IMPLIED** : Default avisa à aplicação que nenhum valor foi especificado

# Tipos de Atributos e #DEFAULT

- **#FIXED** : Default indicando que o valor especificado deve ser igual ao valor padrão senão o documento é inválido
- **ID** : Tipo “token” identifica unicamente um elemento
- **ENTITY** : Tipo “token” aponta para entidade geral externa declarada na DTD
- **NMTOKEN** : Tipo string que toma forma de um “token”

# Estrutura Lógica XML

- Declaração do tipo de elemento (restringe o conteúdo do elemento)
- Declaração da lista de atributos (pares nome-valor)

Ex. Declaração:

```
<!ELEMENT City - - (continent)>  
<!ATTLIST City  
    name CDATA #REQUIRED  
    country CDATA #IMPLIED>  
<!ELEMENT continent CDATA>  
...
```



# Ocorrência do Elemento

Ex.

```
<City name="Buenos Aires"  
  country="Argentina">  
  <continent>  
    América do Sul  
  </continent>  
</City>
```

# Declaração do tipo de elemento

Ex. Declaração:

```
<!ELEMENT City EMPTY>  
<!ATTLIST City  
    name CDATA #REQUIRED  
    country CDATA #IMPLIED>  
...
```

Ex. Ocorrência :

```
<City name="Buenos Aires"  
    country="Argentina"/>
```

# Estrutura Física XML

- Fisicamente um documento XML tem uma ou mais unidades de armazenamento chamadas entidades (*entities*)
- *Parsed entity* ou *unparsed entity*
- Dois tipos de *Parsed entity* : entidades gerais (conteúdo) e entidades de parâmetros (DTD)
- Referências às Entidades gerais : `&toc.xml`;
- Referências às Entidades de parâmetros : `%ações`;

# Exemplos de Referências

- Referência às entidades gerais

```
<!DOCTYPE livro SYSTEM "livro.dtd" [  
    <!ENTITY indice SYSTEM "indice.xml">  
]  
<livro><head>&indice;</head>  
</livro>
```

- Referência às entidades de parâmetros

```
<!ENTITY % pattern "CDATA">  
<!ENTITY % actions "#PCDATA">  
<!ELEMENT rule %actions;>  
<!ATTLIST rule  
    pattern %pattern; #REQUIRED>
```

# ArcWorld.dtd

```
<?xml version="1.0" encoding="UTF-8"?>  
<!-- ArcWorld.dtd -->
```

```
<!ELEMENT ArcWorld (Country*,City*,Building*,  
Church*)>
```

```
<!ELEMENT Country EMPTY>  
<!ATTLIST Country  
    name CDATA #REQUIRED>
```

```
<!ELEMENT City EMPTY>  
<!ATTLIST City  
    name CDATA #REQUIRED  
    country CDATA #IMPLIED>
```

```
<!ELEMENT Architect EMPTY>  
<!ATTLIST Architect  
    name CDATA #REQUIRED  
    nationality CDATA #IMPLIED>
```

# ArcWorld.dtd (cont.)

**<!ELEMENT Building EMPTY>**

**<!ATTLIST Buildingt**

**name CDATA #REQUIRED**

**type CDATA #IMPLIED**

**address CDATA #IMPLIED**

**city CDATA #IMPLIED**

**yearBuilt CDATA #IMPLIED**

**architect CDATA #IMPLIED**

**style CDATA #IMPLIED**

**description CDATA #IMPLIED>**

**<!ELEMENT Church EMPTY>**

**<!ATTLIST Buildingt**

**name CDATA #REQUIRED**

**type CDATA #IMPLIED**

**address CDATA #IMPLIED**

**city CDATA #IMPLIED**

**yearBuilt CDATA #IMPLIED**

**architect CDATA #IMPLIED**

**style CDATA #IMPLIED**

**description CDATA #IMPLIED**

**denomination CDATA #IMPLIED**

**pastor CDATA #IMPLIED>**

# ArcWorld.xml

```
<?xml version="1.0" ?>
<!-- ArcWorld.xml -->

<!ELEMENT ArcWorld (Country*,City*,Building*, Church*)>

<!DOCTYPE ArcWorld SYSTEM "ArcWorld.dtd">

<ArcWorld>
  <Country name="USA"/>
  <Country name="France"/>
  <Country name="China"/>

  <City name="Washington" country="USA">
    <City name="Paris" country="France">
      <City name="Beijing" country="China">

        <Building name="Lincoln Memorial" city="Washington"/>
        <Building name="National Gallery" city="Washington"/>
        <Building name="The Capitol" city="Washington"/>
        <Building name="Washington Monument" city="Washington"/>
        <Building name="Eiffel Tower" city="Paris"/>
      </City>
    </City>
  </City>
</ArcWorld>
```

## ArcWorld.xml (Cont)

```
<Building name="Arc de Triumph" city="Paris"/>
```

```
<Building name="Louvre" city="Paris"/>
```

```
<Building name="Great Wall" city="China"/>
```

```
<Building name="Tiananmen" city="China"/>
```

```
</ArcWorld>
```



# Reuso, Chave Primária, Hierarquia

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

```
<!-- ArcWorld00.dtd -->
```

```
<!ELEMENT ArcWorld00 (Country*)>
```

```
<!ELEMENT Country (City*)>
```

```
<!ATTLIST Country
```

```
    id ID #REQUIRED
```

```
    name CDATA #REQUIRED>
```

```
<!ELEMENT City (Building | Church)*>
```

```
<!ATTLIST City
```

```
    id ID #REQUIRED
```

```
    name CDATA #REQUIRED
```

```
    country IDREF #IMPLIED>
```

# Reuso, ...

```
<!-- Cont. ArcWorld00.dtd -->
```

```
<!ELEMENT Architect EMPTY
```

```
<!ATTLIST Architect
```

```
    id ID #REQUIRED
```

```
    name CDATA #REQUIRED
```

```
    nationality IDREF #IMPLIED>
```

```
<!ELEMENT Building EMPTY
```

```
<!ATTLIST Buildingt
```

```
    id ID #REQUIRED
```

```
    name CDATA #REQUIRED
```

```
    type CDATA #IMPLIED
```

```
    address CDATA #IMPLIED
```

```
    city IDREF #IMPLIED
```

```
    yearBuilt CDATA #IMPLIED
```

```
    architect IDREF #IMPLIED
```

```
    style CDATA #IMPLIED
```

```
    description CDATA #IMPLIED>
```

# Reuso, ...

```
<!-- Cont. ArcWorld00.dtd -->
```

```
<!ELEMENT Church EMPTY
```

```
<!ATTLIST Church
```

```
    id ID #REQUIRED
```

```
    name CDATA #REQUIRED
```

```
    type CDATA #IMPLIED
```

```
    address CDATA #IMPLIED
```

```
    city IDREF #IMPLIED
```

```
    yearBuilt CDATA #IMPLIED
```

```
    architect IDREF #IMPLIED
```

```
    style CDATA #IMPLIED
```

```
    description CDATA #IMPLIED
```

```
    denomination CDATA #IMPLIED
```

```
    pastor CDATA #IMPLIED>
```

# Reuso, ... (XML)

```
<?xml version="1.0" ?>
<!-- ArcWorld00.xml -->

<!DOCTYPE ArcWorld00 SYSTEM "ArcWorld00.dtd">

<ArcWorld00>

  <Country id C1 name="USA"/>
  <City id="C1c1" name="Washington" country="C1">
    <Building id="C1c1b1" name="Lincoln Memorial" city="C1c1"/>
    <Building id="C1c1b2" name="National Gallery" city="C1c1"/>
    <Building id="C1c1b3" name="The Capitol" city="C1c1"/>
    <Building id="C1c1b4" name="Washington Monument" city="C1c1"/>
  </City>
</Country>

  <Country id="C2" name="France"/>
  <City id="C2c1" name="Paris" country="C2">
    <Building id="C2c1b1" name="Eiffel Tower" city="C2"/>
  </City>
</Country>
</ArcWorld00>
```

# Reuso, ... (XML)

```
<Country id="C2" name="France"/>
<City id="C2c1" name="Paris" country="C2">
  <Building id="C2c1b1" name="Eiffel Tower" city="C2c1"/>
  <Building id="C2c1b2" name="Arc de Triumph" city="C2c1"/>
  <Building id="C2c1b3" name="Louvre" city="C2c1"/>
</City>
</Country>
```

```
<Country id="C3" name="China"/>
<City id="C3c1" name="Beijing" country="C3">
  <Building id="C3c1b1" name="Great Wall" city="C3c1"/>
  <Building id="C3c1b2" name="Tiananmen" city="C3c1"/>
</City>
</Country>
```

```
</ArcWorld>
```

# XML NameSpaces

- Reuso de esquemas
- Reuso de marcas de outros documentos XML
- Nomes únicos para tags e atributos através `prefixo.nome local`
- Ex.

```
<?xml version = "1.0"?>
<xsl:stylesheet version = "1.0"
  xmlns:xsl =
http://www.w3.org/1999/XSL/Transform>
  <xsl:template match = "/">
    . . .
```

# Document.xml

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

```
<?xml:stylesheet type = "text/xsl" href = "usage.xsl"?>
```

```
<document editor="jose">
```

```
<titulo>Titulo</titulo>
```

```
<autor>Joao</autor>
```

```
<nada/>
```

```
</document>
```

```
<!-- Exemplo simples de um documento XML -->
```

# Usage.xsl

```
<?xml version = "1.0" encoding="UTF-8"?>
<xsl:stylesheet version = "1.0"
  xmlns:xsl = "http://www.w3.org/1999/XSL/Transform">

  <xsl:template match = "document">

    <html>
      <body>
        <xsl:value-of select="@editor"/>
        <xsl:apply-templates/>
      </body>
    </html>

  </xsl:template>

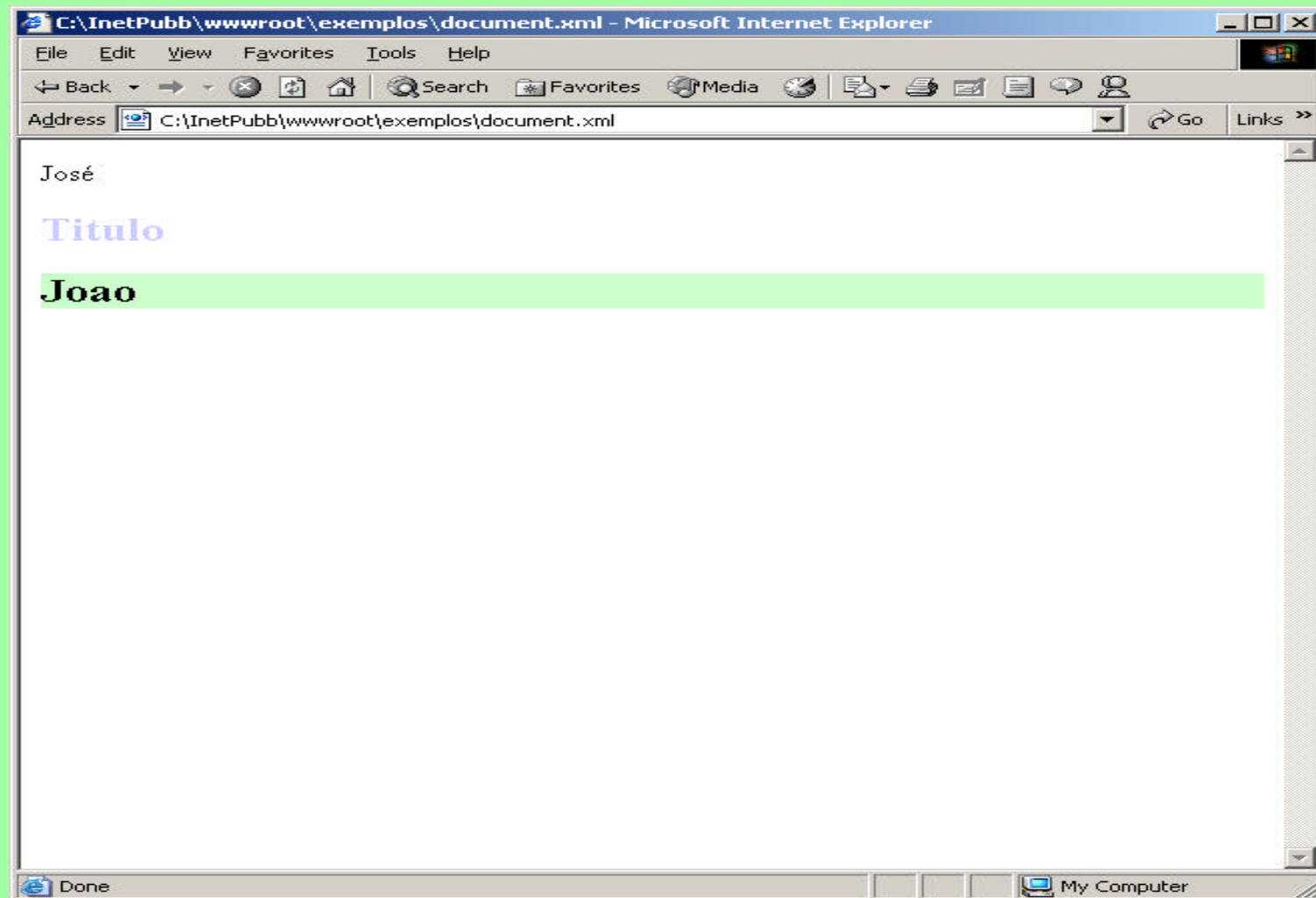
  <xsl:template match="titulo">
    <h2 style = "color:#ccccff">
      <xsl:value-of select="."/>
    </h2>
  </xsl:template>

  <xsl:template match="autor">
    <h2 style = "background-color: #ccffcc">
      <xsl:value-of select = "."/>
    </h2>
  </xsl:template>

  <xsl:template match = "*/text()"/>
</xsl:stylesheet>
```



# resultado



# XLINK (XML Linking Language)

- Links entre objetos ou porções de objeto
- Dois tipos : simples e estendidos
- Ex. de links simples (Unidirecional)

```
<!-- HTML link -->
```

```
<A xml:link="simple" href="http://www.w3.org/">The W3C</A>
```

```
<Citation xml:link="simple"
```

```
  href="http://www.xyz.com/xml/smith.xml"
```

```
  title="Smith" show="new"
```

```
  content-role="Reference">
```

```
    as discuted in Smith(1988)
```

```
</Citation>
```

# Estilo x Comportamento

- HTML mistura conceitos

**<-- Link bidirecional -->**

**<HEAD>**

**<LINK rel="stylesheet" href="http://www.w.com/style">**

**<LINK rev="made" href="valdeni@inf.ufrgs.br">**

**</HEAD>**

**<-- Estilos de link como parâmetro do corpo -->**

**<BODY link=#FF0000 alink=#00FF00 vlink=#0000FF>**

**<A href="http://www.w3.org/">The W3C</A>**

**</BODY>**

# Links estendidos

- Multidirecionais

**<-- Link bidirecional -->**

```
<Comentario xml:link="extended" inline="false"
  <locator href="smith2.1" role="Essay"/>
  <locator href="jones1.4" role="Rebuttal"/>
  <locator href="robin3.2" role="Comparison"/>
</Comentario>
```

- *inline* ou *out-of-line*

- Políticas de comportamento

- Show, Actuate e behavior

# Políticas de Comportamento dos Links

- **Show** (materializa ou processa)
  - *embed* indica que o recurso alvo será aninhado no recurso no mesmo ponto onde disparou o link para o propósito de materialização ou processamento
  - *replace* indica que o recurso que disparou o link será substituído para fins de materialização e processamento
  - *new* indica que o recurso alvo pode ser materializado ou processado em um novo contexto sem afetar o recurso que iniciou o link
- **Actuate** (quando)
  - *auto* indica que o recurso deveria ser recuperado quando qualquer um dos outros recursos do mesmo link é encontrado. Também o recurso inicial não é considerado completo até que isto aconteça
  - *user* depende de uma explícita requisição externa
- **Behavior**
  - detalhes do comportamento do link

# Endereçamento de partes de Documento XML

- ***Absolute location terms***
  - *root()* o elemento raiz do documento XML
  - *origin()* o diretório onde o usuário começou
  - *id(name)* o elemento contendo o ID declarado com o valor igual ao *name*
- ***Relative location terms***
  - child, descendant, ancestor, proceeding, following, psibling, fsibling
  - EX. descendant(-1, section)  
ancestor(1, #element, name, 3.2)
- ***Span location terms***
  - Ex. Id(a1).span(child(1),child(3))

# Endereçamento ...

- *Attribute location terms*

- Ex. `Ancestor(1,#element).attr(name)`

- `<-- fornece o valor do atributo name do  
mais perto elemento antecessor -->`

- *String location terms*

- Ex. `string(3,"")`

- `<-- fornece a posição antes do terceiro  
caractere -->`

- `string(5,"!",1,1)`

- `<-- seleciona a quinta ! e o caractere  
imediatamente depois -->`

# Metadados

- *RDF (Resource Description Framework)*



- **Aplicabilidade**
  - Descoberta de recursos
  - Catálogos de bibliotecas, sites e páginas
  - Agentes inteligentes compartilhando documentos
  - Reuso (múltiplas páginas com um único documento)
  - Direitos autorais
  - Privacidade



# Troca de Informações

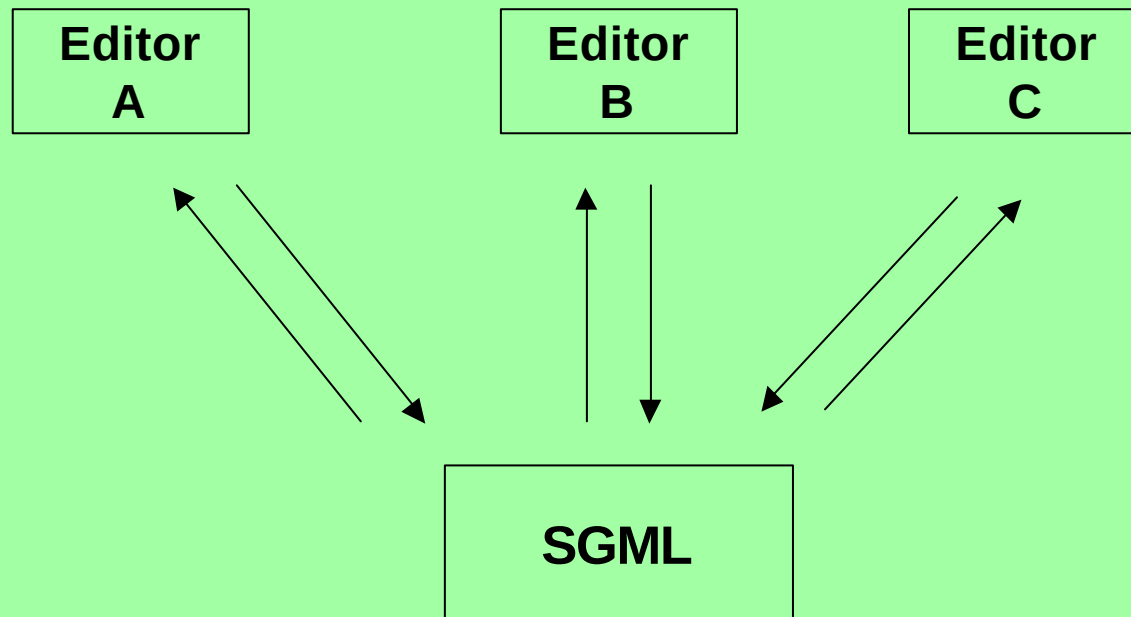
## ⇒ **Nível Pessoal**

- ♦ MIME - *Multipurpose Internet Mail Extensions*
- ♦ ODIF - *Office Document Interchange Format*
- ♦ SDIF - *SGML Document Interchange Format*

## ⇒ **Nível Global**

- ♦ Gopher
- ♦ WAIS - *Wide Area Information Servers*
- ♦ WWW - *World Wide Web*

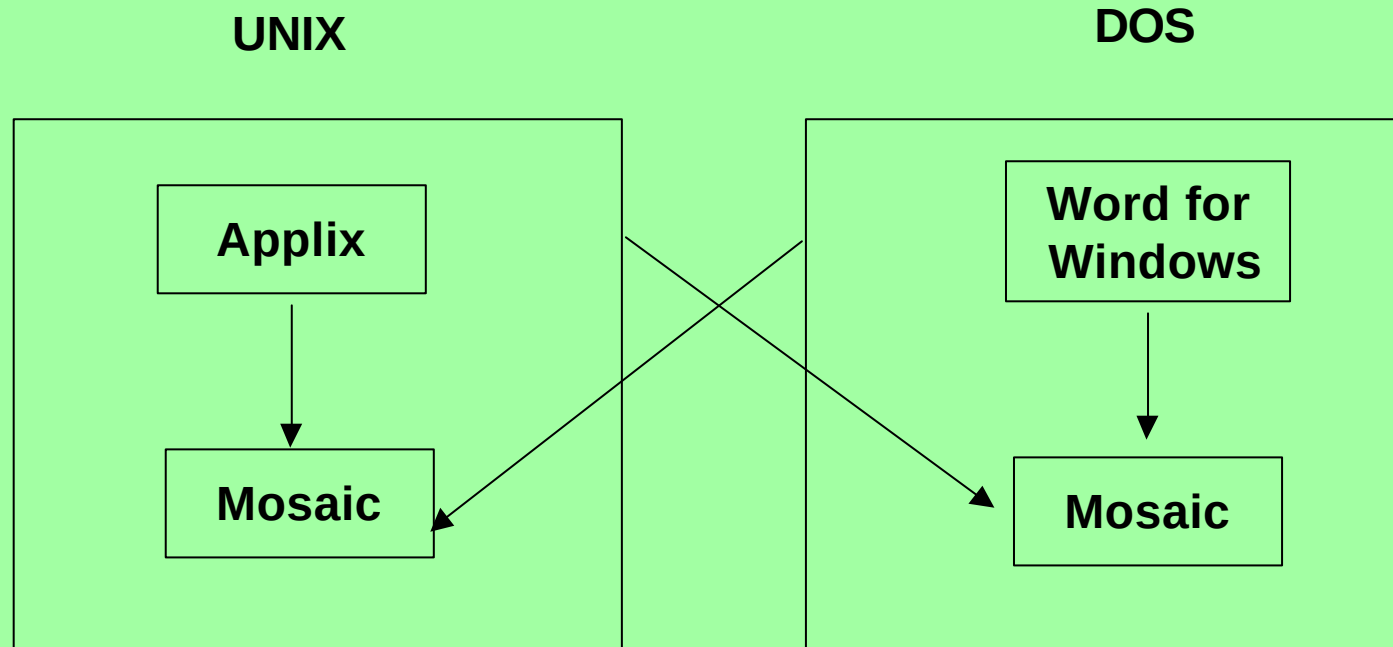
## Filtro para um padrão.



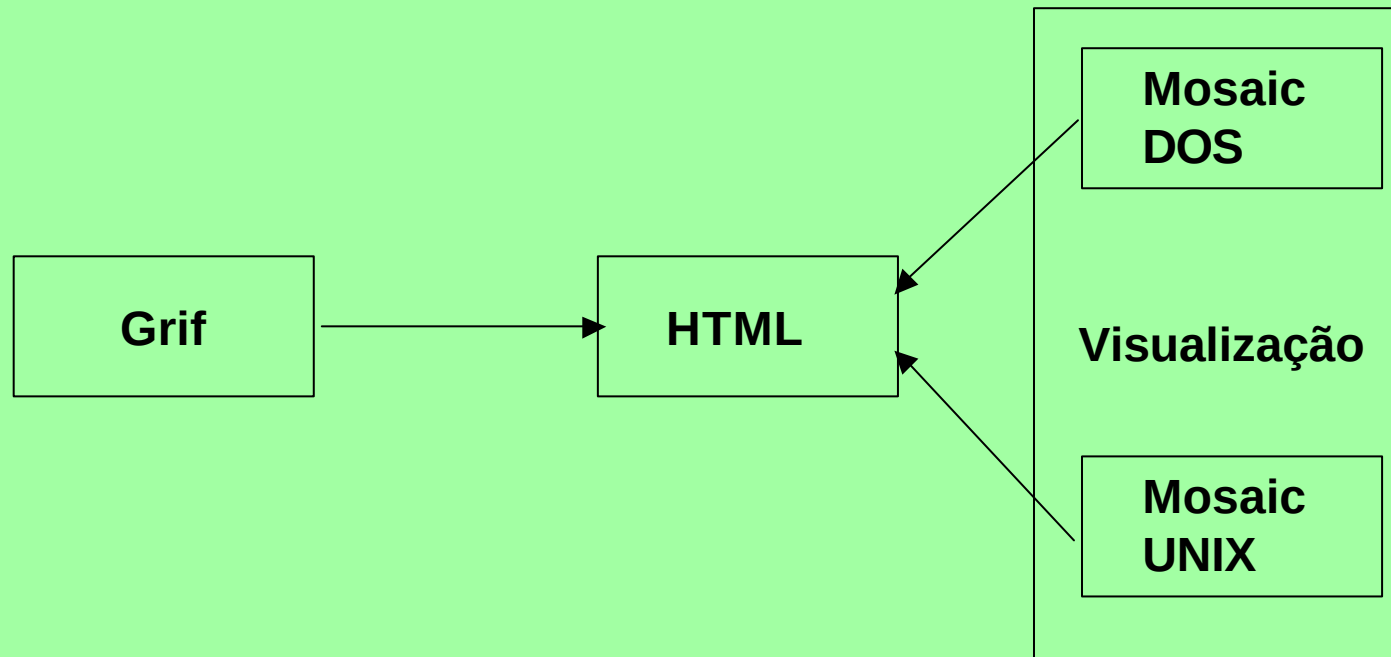
## Conversão de documentos Applix para HTML.



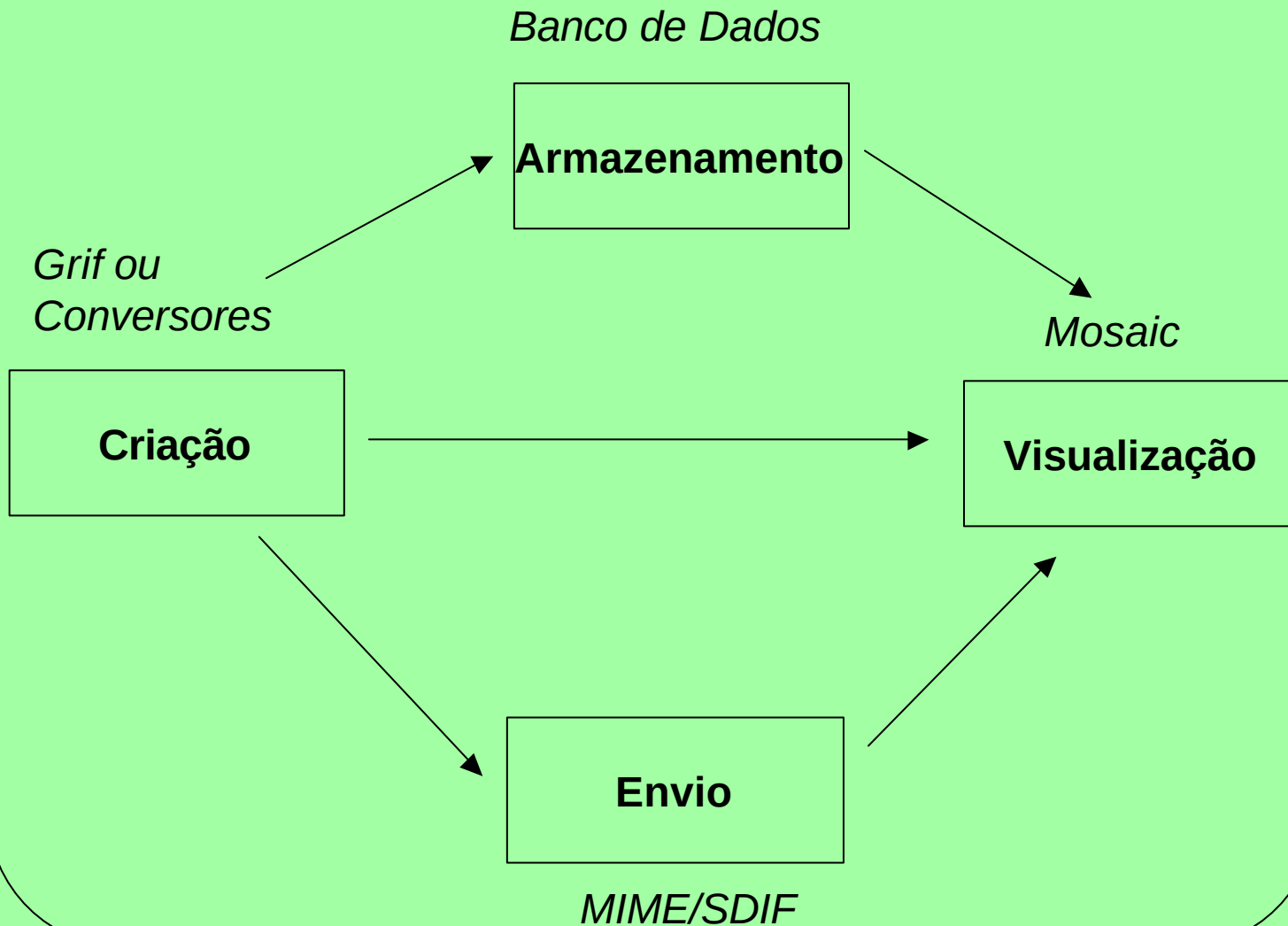
# Visualização de Documentos



## Criação de documentos em HTML através do Grif.



## Ciclo de Vida de um Documento



# Proposta de Intercâmbio de Documentos Estruturados.

