



XML e Cia

XML, Esquema XML, Espaço de Nomes XML

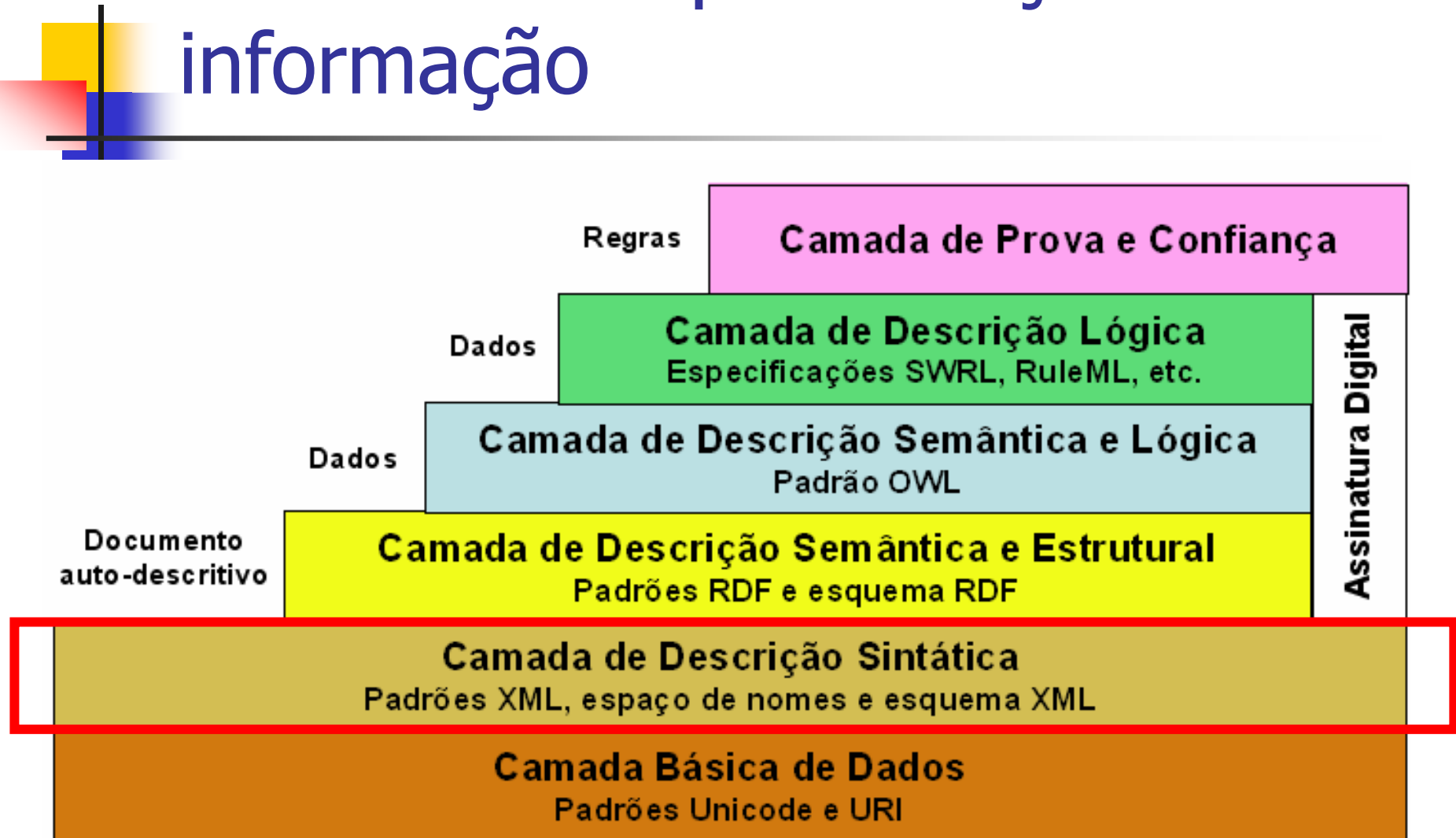
Renato Bulcão Neto
Cássio Vinicius Prazeres
Maria da Graça Pimentel

`{rbulcao,prazeres,mgp}@icmc.usp.br`



Departamento de SCC
Ciências de Computação

Padrões de representação de informação





Linguagem XML



Documento HTML

```
<h1>CDs</h1>
```

```
<h2>Sou Chicleteiro</h2>
```

```
<b>Músicas</b>
```

```
<ol>
```

```
  <li>100% Você</li>
```

```
  <li>Do nosso jeito</li>
```

```
</ol>
```



Documento XML

```
<?xml version="1.0"?>
<cd year="2004">
  <title>Sou Chicleteiro</title>
  <company>Sony Music</company>
  <price>10.90</price>
  <country>Brasil</country>

  <song duration="360">
    <title>100% Você</title>
    <composer>Beto Garrido</composer>
    <composer>Alexandre Peixe</composer>
    <artist>Chiclete com Banana</artist>
    <lyrics href="http://www.cdstore.fake/chiclete100voce.txt"/>
  </song>

  <song duration="335">
    <title>Do Nosso Jeito</title>
    <composer>Lucas Monteiro</composer>
    <artist>Chiclete com Banana</artist>
    <lyrics href="http://www.cdstore.fake/chicletedonossojeito.txt"/>
  </song>
</cd>
```



Similaridades entre XML e HTML

- Os dois usam marcações (ou tags)
 - `<h1>` e `<title>`
- As marcações podem ser aninhadas
- Pessoas podem ler e interpretar ambos
 - E as máquinas?



Informação estrutural (1)

- Documentos HTML não contêm informação estrutural (ou muito pouca)
 - Não há relacionamentos entre partes do documento
- XML é facilmente processado por aplicações
 - Todo trecho de informação é descrito
 - Relações são definidas a partir de estrutura de aninhamento



Informação estrutural (2)

- Exemplo
 - Se <autor> aparece dentro de <livro>, então este descreve uma propriedade de um livro
- Um software pode processar o documento XML e poderia deduzir que
 - <autor> refere-se ao elemento <livro> imediatamente acima dele
- XML permite a definição de restrições em valores
 - Exemplo: ano precisa ter quatro dígitos



Formatação

- HTML provê formatação
 - O principal uso de HTML é para exibir informação
- XML separa conteúdo da apresentação
 - A mesma informação pode ser mostrada de diferentes formas



HTML vs XML: Outro exemplo

- Em HTML

```
<h2>Relação matéria-energia</h2>
```

```
<i>E = M × c2</i>
```

- Em XML

```
<equacao>
```

```
  <significado>Relação matéria-energia </significado>
```

```
  <ladoEsquerdo>E</ladoEsquerdo>
```

```
  <ladoDireito>M × c2</ladoDireito>
```

```
</equacao>
```



HTML vs XML: Diferentes tags

- Em HTML as marcações são sempre as mesmas
- Em XML as marcações são completamente diferentes
- Marcações HTML definem apresentação: cores, fontes, tamanho de fontes ...
- Marcações XML não são fixas: são definidas pelo usuário



Vocabulários XML

- XML como meta-linguagem
- Aplicações Web devem utilizar vocabulários comuns para interoperarem
- Alguns vocabulários têm sido definidos por comunidades
 - MathML (matemática)
 - BSML (bioinformática)
 - HRML (recursos humanos)



Árvore de documentos XML (1)

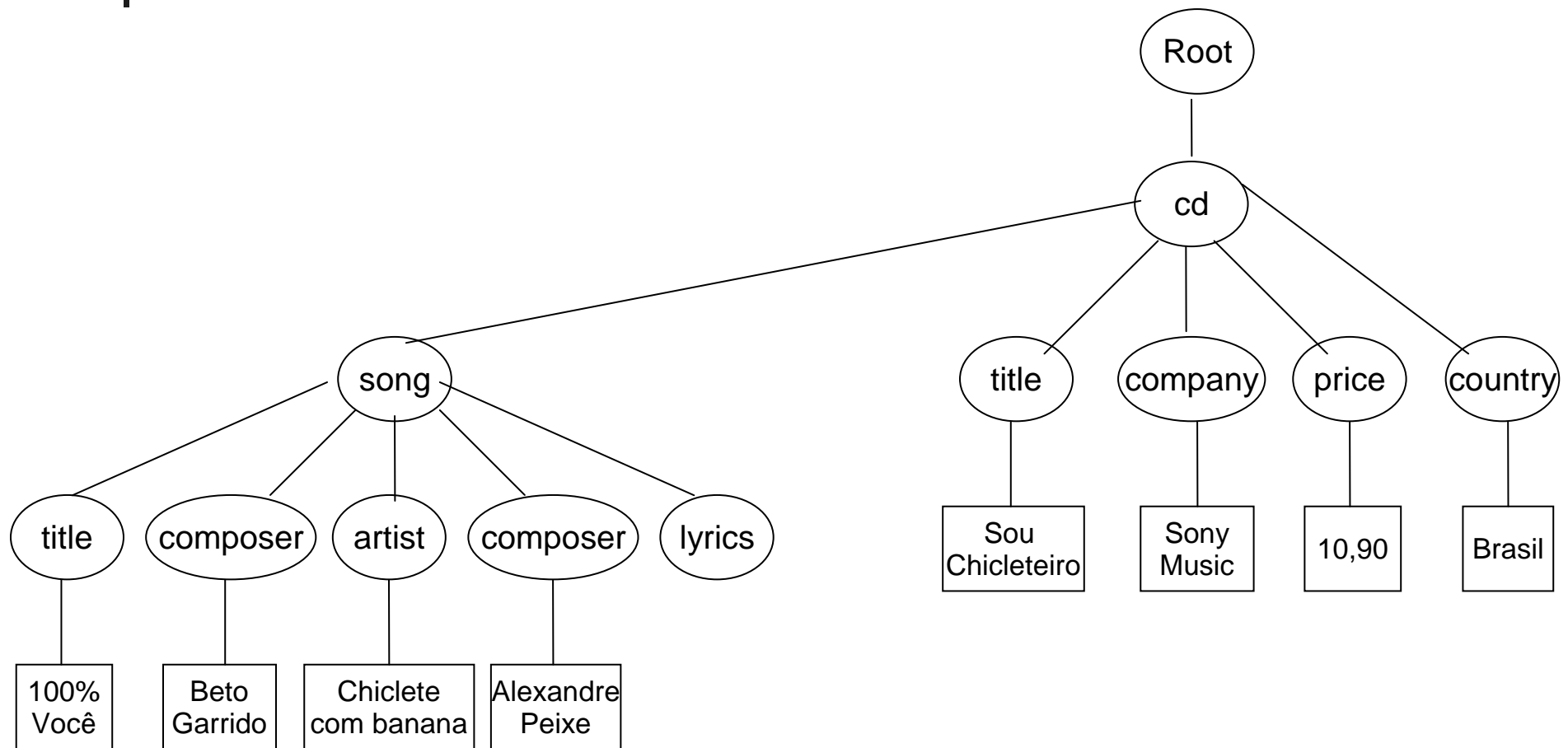
```
<?xml version="1.0"?>
<cd year="2004">
  <title>Sou Chicleteiro</title>
  <company>Sony Music</company>
  <price>10.90</price>
  <country>Brasil</country>

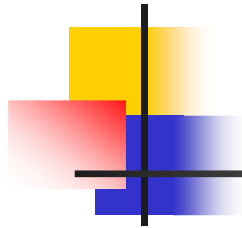
  <song duration="360">
    <title>100% Você</title>
    <composer>Beto Garrido</composer>
    <composer>Alexandre Peixe</composer>
    <artist>Chiclete com Banana</artist>
    <lyrics href="http://www.cdstore.fake/chiclete100voce.txt"/>
  </song>

  <song duration="335">
    <title>Do Nosso Jeito</title>
    <composer>Lucas Monteiro</composer>
    <artist>Chiclete com Banana</artist>
    <lyrics href="http://www.cdstore.fake/chicletedonossojeito.txt"/>
  </song>
</cd>
```



Árvore de documentos XML (2)



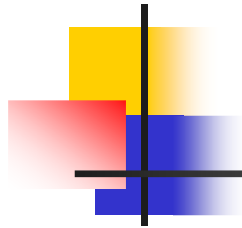


Árvore de documentos XML (3)

- A representação em árvore dos documentos XML é uma árvore ordenada e rotulada
 - Exatamente uma raiz
 - Não possui ciclos
 - Cada nó (exceto o raiz) tem exatamente um pai
 - A ordem dos elementos é importante
 - A ordem dos atributos não importa



Esquema XML



Esquema XML

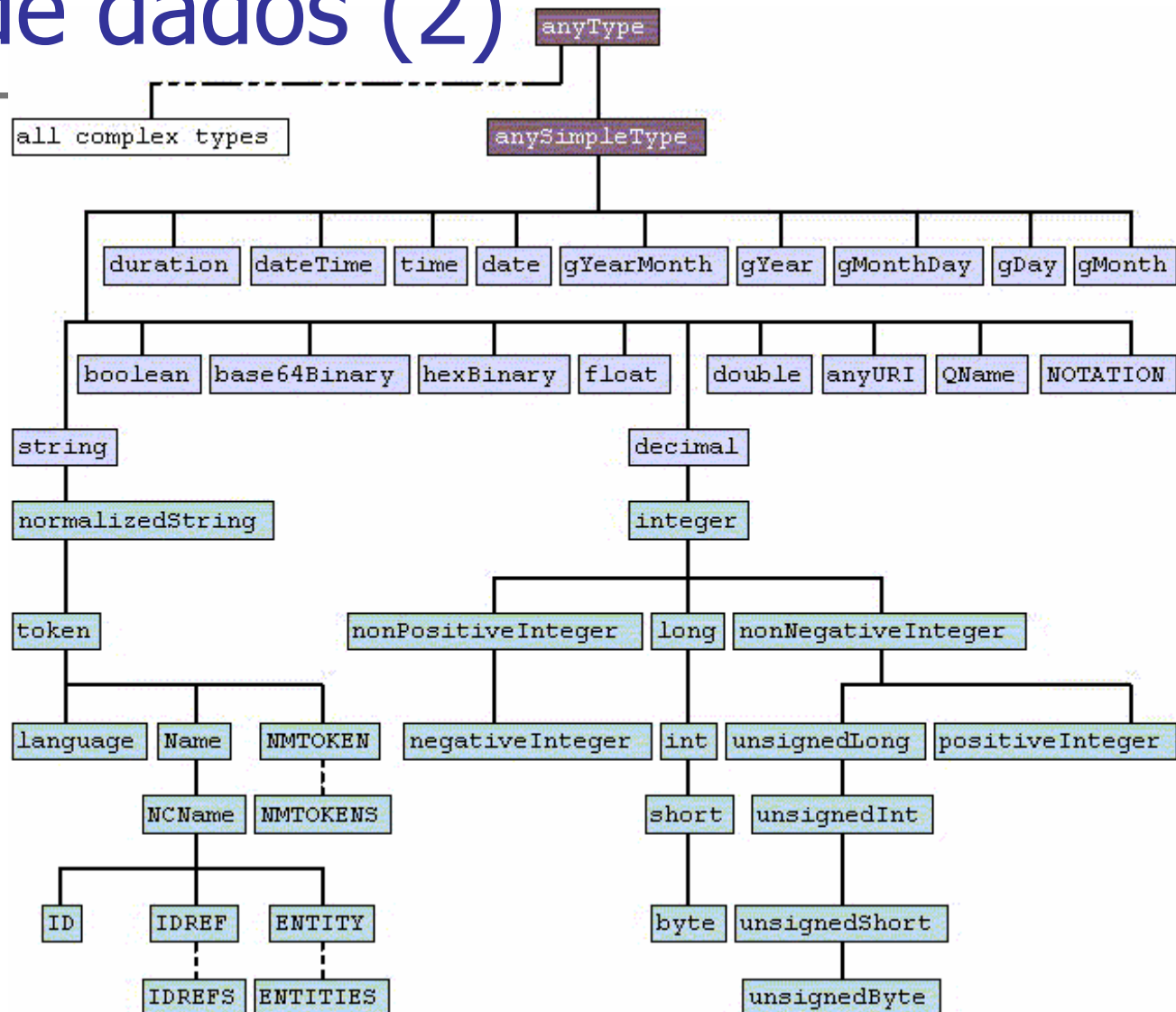
- Linguagem para definição da estrutura de documentos XML
- Sintaxe baseada no próprio XML
- Permite reuso de esquemas existentes
- Conjunto sofisticado de tipos de dados



Tipos de dados (1)

- Existe uma variedade de tipos de dados nativos
 - Numéricos: integer, short, decimal, ...
 - String
 - Data e tempo: time, date, gYear, gMonth, ...

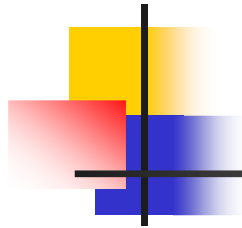
Tipos de dados (2)





Tipos de dados (3)

- Tipos de dados podem ser definidos pelo usuário
 - Tipos de dados simples (não contém outros elementos nem atributos)
 - Tipos de dados complexos
 - Definidos a partir dos tipos básicos contendo atributos e/ou outros elementos



Tipos de dados (4)

```
<complexType name="lecturerType">
  <sequence>
    <element name="firstname" type="string"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="lastname" type="string"/>
  </sequence>
  <attribute name="title" type="string" use="optional"/>
</complexType>
```



Esquema XML: Exemplo

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

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
  <xs:element name="cd">
```

```
    <xs:complexType>
```

```
      <xs:sequence>
```

```
        <xs:element name="title" type="xs:string"/>
```

```
        <xs:element name="company" type="xs:string"/>
```

```
        <xs:element name="price" type="xs:decimal"/>
```

```
        <xs:element name="country" type="xs:string"/>
```

```
      <xs:element name="song" maxOccurs="unbounded">
```

```
        <xs:complexType>
```

```
          <xs:sequence>
```

```
            <xs:element name="title" type="xs:string"/>
```

```
            <xs:element name="composer" type="xs:string" maxOccurs="unbounded"/>
```

```
            <xs:element name="artist" type="xs:string"/>
```

```
            <xs:element name="lyrics">
```

```
              <xs:complexType>
```

```
                <xs:attribute name="href" type="xs:anyURI"/>
```

```
              </xs:complexType>
```

```
            </xs:element>
```

```
          </xs:sequence>
```

```
        </xs:complexType>
```

```
        <xs:attribute name="duration" type="xs:duration"/>
```

```
      </xs:element>
```

```
    </xs:sequence>
```

```
    <xs:attribute name="year" type="xs:gYear"/>
```

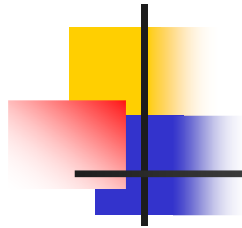
```
  </xs:complexType>
```

```
</xs:element>
```

```
</xs:schema>
```



Espaço de nomes XML



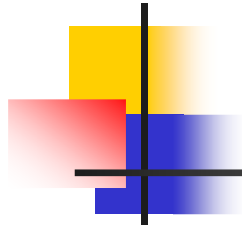
Espaço de nomes XML

- Um documento XML pode usar mais que um esquema
- Cada esquema pode ter sido desenvolvido separadamente
 - Podem existir nomes conflitantes
- A solução é utilizar prefixos diferentes para cada esquema
 - `prefixo:nome`



Espaço de nomes XML: Declaração

- Uma declaração de espaço de nomes XML tem a seguinte forma
 - `xmlns:prefix="localização"`
 - localização é o endereço do Esquema XML que descreve a estrutura do documento XML em questão



Espaço de nomes XML: Exemplo

```
<vu:instructors    xmlns:vu="http://www.vu.com/empDTD"
                  xmlns:gu="http://www.gu.au/empDTD"
                  xmlns:uky="http://www.uky.edu/empDTD">
```

```
  <uky:faculty    uky:title="assistant professor"
                 uky:name="John Smith"
                 uky:department="Computer Science"/>
```

```
    <gu:academicStaff    gu:title="lecturer"
                       gu:name="Mate Jones"
                       gu:school="Information Technology"/>
```

```
</vu:instructors>
```



Pontos para discussão

- Motivação para próximos slides
 - Aninhamento de tags não tem significado padrão
 - Semântica de documentos XML não é acessível por software, apenas por pessoas
 - Colaboração e intercâmbio são apoiados se existir um entendimento compartilhado de um vocabulário
 - XML é mais adequada para colaboração de baixa granulosidade (comunidade, empresa, etc.)
 - Não é adequada para comunicação global

XML e Cia

XML, Esquema XML, Espaço de Nomes XML

Renato Bulcão Neto

Cássio Vinicius Prazeres

Maria da Graça Pimentel

`{rbulcao,prazeres,mgp}@icmc.usp.br`



Departamento de SCC
Ciências de Computação