Grupo de Especificação e Processamento de Linguagens

Constraining Content

XCSL –XML Constraint Specification Language

José Carlos Ramalho
jcr@di.uminho.pt

Grupo de Especificação e Processamento de Linguagens

Contents

- Behind the scene
- Examples
- Constraint
- System Arq
- Example solutions
- Future work

CODE
+
other inner parts

Grupo de Especificação e Processamento de Linguagens

Behind the scene

- SGML96 (Boston)
  - Semantic Validation: two approaches
- SGML97 (Washington)
  - Where does quality go?
- Markup Languages Journal
  - Processing constraints
- Markup Technologies 98 (Chicago)
  - Attribute Grammar approach

Grupo de Especificação e Processamento de Linguagens

Behind the scene (2)

- Phd thesis – 2000
  - Low-level functional approach
  - Formal specification of XCSL:
    - Attribute grammar
    - XAM specification
  - Processor: had to be written in the host functional language (SSL, Haskell)
- Desired goal
  - To have everything in the same declarative paradigm

Grupo de Especificação e Processamento de Linguagens

Motivation

- To meet some particular project needs
- Sometimes we do not want a Schema or a DTD
  - We just want to constraint certain document parts
- Requirements:
  - The solution should work with existing tools and standards
  - The solution should work across every platform

Grupo de Especificação e Processamento de Linguagens

Type inference

Problem: how to process ... ?

Constraint: **latitude > 39 and latitude < 42.5**

Document: **...<latitude>41.32</latitude>...**

Answer:

...<latitude type="float">41.32</latitude>...

Can be transparent to users: #FIXED attributes

Grupo de Especificação e Processamento de Linguagens

Value normalization

Problem: How can I identify ...?

... King <name>Affonso</name> proclaimed several ...

... And his soldiers battled against <name>Afonso</name>.

...and that church was built in the <date>XVIII century</date>.

...it all happened on <date> the fifth October</date>...

Answer

...King <name value="Afonso">Affonso</name>...

...it all happened on <date value="xxxx.10.05">the fifth...

User awareness is needed!

Grupo de Especificação e Processamento de Linguagens

Programs ⇔ XML Documents

- Have a support language formally defined
- Processing - compiler
  - lexical analysis
  - syntactic analysis
  - semantic analysis
    - complex: type checking; type inference, ...
    - Can be formally specified: Attribute Grammars
- Have a support markup language defined in XML
- Processing - parser
  - lexical analysis
  - syntactic analysis
  - semantic analysis
    - very simple: ID - IDREF coupling

Constraints: what type?

i. Domain range checking
   - Normally data is of type numeric or date

ii. Dependencies between two elements or attributes

iii. Pattern matching against a Regular Expression
   - Enforcing content to follow a certain format
   - Example: [0-9]{4}\.[0-9]{2}\.[0-9]{2}

iv. Quantified constraints … (???)

Example1: students

```
<?xml version="1.0"?>
<students>
 <student>
  <name>Peter Weird</name>
  <grades>
   <grade1>12</grade1>
   <grade2>8</grade2>
   <grade3>15</grade3>
  </grades>
 </student>
 <student>
  <name>Jose Almeida</name>
  <grades>
   <grade1>9</grade1>
   <grade2>18</grade2>
   <grade3>7</grade3>
  </grades>
 </student>
</students>
```

We want:
- to ensure that each grade is greater than 10
- to give a warning message for each grade below 10

domain range (i)

Grupo de Especificação e Processamento de Linguagens

Example2: linguistics

```
<?xml version="1.0"?>
<doc>
<sentence>
  <noun number="s" genre="f">Alice</noun>
  <verb time="present" number="s“ person="3">drinks</verb>
</sentence>
<sentence>
  <noun number="p">Dogs</noun>
  <verb time="present" number="s" person="3">barks</verb>
</sentence>
<sentence>
  <noun genre="f" number="s">Diana</noun>
  <verb number
  <adj genre="
</sentence>
</doc>
```

In portuguese:

- verbs must agree with correspondent nouns in person and number
- adjectives must agree with correspondent nouns in genre and number

Grupo de Especificação e Processamento de Linguagens

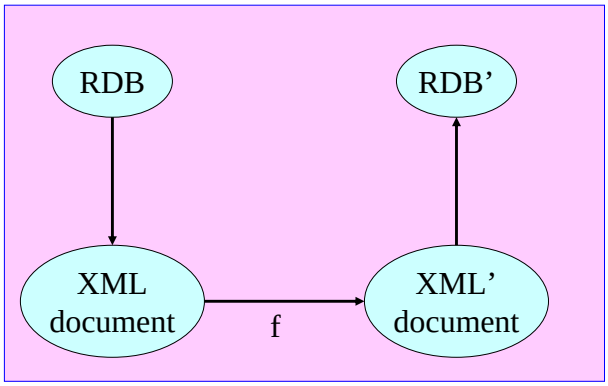
Example3: XDBML

```
<?xml version="1.0"?>
<DB>
 <STRUCTURE>
  <TABLE NAME="ite
   <COLUMNS>
    <COLUMN NAME
    <COLUMN NAME
    ...
   </COLUMNS>
   <KEYS>
    <PKEYS TYPE="s
     <PKEY NAME="
    </PKEYS>
   </KEYS>
  </TABLE>
 </STRUCTURE>
```

```
<DATA>
  <items>
   <items-REG>
    <code>a111</code>
    <description>leite agros meio-gordo</description>
   </items-REG>
   <items-REG>
    <code>a115</code>
    <description>leite agros chocolatado</description>
   </items-REG>
   <items-REG>
    <code>a112</code>
    <description>leite agros meio-gordo</description>
   </items-REG>
   <items-REG>
    <code>a115</code>
    <description>leite agros chocolatado</description>
   </items-REG>
   ...
</DATA>
</DB>
```

Grupo de Especificação e Processamento de Linguagens

XDBML(2): the problem



Grupo de Especificação e Processamento de Linguagens

XDBML(2): the problem

- We want to transform the database in the XML axis (with XSLT).
- We want to upload the new database into a DBMS
- We must ensure:
  - Primary keys are still primary keys (challenging)
    - ID/IDREF attributes are globally unique
  - Other constraints are easy to deal with
- Do we need quantifiers?
  - For each key, test if its value is unique.
  - Are quantifiers already there?

Grupo de Especificação e Processamento de Linguagens

Constraint Spec. Lang.

- From author's Phd
  - Subset of XSLT
  - Composition: set of triples

```
ConstraintSpec = Constraint+
Constraint = (ContextSelector,ContextCondition,Action)
```

We can use a XSLT processor to handle Const.Processing

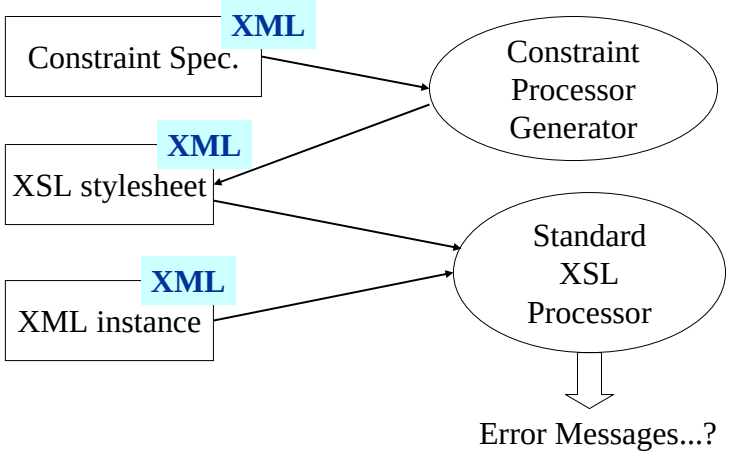
XPath

Xpath predicates

?? code

Grupo de Especificação e Processamento de Linguagens

Architecture



Grupo de Especificação e Processamento de Linguagens

CSL with a XML wrapper

DTD: V1.0

```
<!ELEMENT cs (constraint)+>
<!ELEMENT constraint (selector,cc,action)>
<!ELEMENT selector EMPTY>
<!ELEMENT cc       EMPTY>
<!ELEMENT action   (message*)>
<!ELEMENT message  (#PCDATA|value)*>
<!ELEMENT value EMPTY>
<!ATTLIST cs
              dtd CDATA #IMPLIED
              date CDATA #IMPLIED
              version CDATA #IMPLIED >
<!ATTLIST selector
              selexp CDATA #REQUIRED>
<!ATTLIST cc
              cond CDATA #REQUIRED>

<!ATTLIST value
              selexp CDATA #REQUIRED>
```

- **action is a message list**
- **value is used to refer document nodes inside messages**

Grupo de Especificação e Processamento de Linguagens

XCSL1: Students

```
<?xml version="1.0"?>
<CS>
 <CONSTRAINT>
  <SELECTOR SELEXP="//student/grades/*"/>
  <CC>.&gt;10</CC>
  <ACTION>
   <MESSAGE>WARNING:
    <VALUE SELEXP="name(.)"/> of
    <VALUE SELEXP="../../name"/> is below minimum!</MESSAGE>
  </ACTION>
 </CONSTRAINT>
</CS>
```

Spotting grades lower than 10.

Grupo de Especificação e Processamento de Linguagens

XCSL2: linguistics

```
<?xml version="1.0"?>
<!DOCTYPE CS SYSTEM "csl.dtd">
<CS>
  <CONSTRAINT>
    <SELECTOR SELEXP="//sentence"/>
    <CC>noun/@number=verb/@number</CC>
    <ACTION>
      <MESSAGE>Attribute number of "<VALUE SELEXP="noun"/>"
       does not agree with attribute number of
       <VALUE SELEXP="verb"/>"!!!</MESSAGE>
    </ACTION>
  </CONSTRAINT>
  <CONSTRAINT>
    <SELECTOR SELEXP="//sentence"/>
    <CC>noun/@genre=adj/@genre</CC>
    <ACTION>
      <MESSAGE>ERROR:<VALUE SELEXP="noun"/> e
        <VALUE SELEXP="adj"/> do not agree in genre!!!</MESSAGE>
    </ACTION>
  </CONSTRAINT></CS>
```

Testing agreements...

Grupo de Especificação e Processamento de Linguagens

XCSL processors

- Implemented with XML::DT:
  - Perl module on top of XML::Parser
  - Presented at XML Europe'99
  - It can be found on:
    - http://natura.di.uminho.pt
    - Comphreensive Perl Archive Network
- XSL processing being done with:
  - Saxon – Windows platforms
  - Xalan – Linux like platforms

CSL processor: V1.0

```
...
%handler=(
     '-outputenc' => 'ISO-8859-1',
     '-default'   => sub{"<$q>$c</$q>"},
     'SELECTOR' => sub{"$v{SELEXP}"},
     'CONSTRAINT' => sub{"<xsl:template
                          match=\"$c\n</xsl:template> \n"},
     'MESSAGE' => sub{"\n  ERROR: $c\n"},
     'VARIABLE' => sub{"<xsl:value-of select=\'$v{SELEXP}\'/>"},
     'CS' => sub{"$c"},
     'ACTION' => sub{"$c"}
     'CC' => sub{"[
);
...
```

Algorithm:
- convert each constraint into a template
- use SELEXP for MATCH attribute
- convert CC into a predicate: [...]
- put MESSAGE contents inside template body
- filter text nodes

Example1: students

```
<?xml version="1.0"?>
<CS>
 <CONSTRAINT>
  <SELECTOR SELEXP="//student/grades/*"/>
  <CC>.&gt;10</CC>
  <ACTION>
   <MESSAGE>WARNING:
    <VALUE SELEXP="name(.)"/> of
    <VALUE SELEXP="../../name"/> is below minimum!</MESSAGE>
  </ACTION>
 </CONSTRAINT>
</CS>
```

XCSL document

Grupo de Especificação e Processamento de Linguagens

Example1: students

```
<xsl:template match="/">
   <xsl:apply-templates/>
</xsl:template>

<!--########## NEW CONSTRAINT ##########-->
<xsl:template match="//student/grades/*[not(.&gt;10)]">
   WARNING:
      <xsl:value-of select='name(.)'/> of
      <xsl:value-of select='../../name'/> is below minimum!
     <xsl:apply-templates/>
</xsl:template>


<xsl:template match="text()" priority="-1"/>
<xsl:template match="//text()" priority="-1">
</xsl:template>
```

Generated XSL stylesheet

Grupo de Especificação e Processamento de Linguagens

Problems with V1.0

- Optional elements
- Non-filled elements
  - Will cause errors
  - “You will catch more than you want, ...”

➢ similar to DB when we have SQL queries over Attributes that can be empty

Solution: CSL V2.0

```
<!ELEMENT cs          (constraint)+>
<!ELEMENT constraint (selector,cc,action)>
<!ELEMENT selector    EMPTY>
<!ELEMENT cc          (#PCDATA|variable)*>
<!ELEMENT action      (message*)>
<!ELEMENT message     (#PCDATA|value)*>
<!ELEMENT variable    EMPTY>
<!ELEMENT value       EMPTY>
<!ATTLIST cs
              dtd CDATA #IMPLIED
              date CDATA #IMPLIED
              version CDATA #IMPLIED >
<!ATTLIST selector
              selexp CDATA #IMPLIED >
<!ATTLIST variable
              selexp CDATA #IMPLIED >
<!ATTLIST value
              selexp CDATA #IMPLIED >
```

Changes:

- CC has a mixed content
- user has a stamp to mark suspicious paths: VARIABLE
- “user must be aware ...”

CSL processor: V2.0

```
%handler=(
     '-outputenc' => 'ISO-8859-1',
     '-default'   => sub{"<$q>$c</$q>"},
     'SELECTOR'   => sub{"$v{SELEXP}"},
     'CONSTRAINT' => sub{@ccvars=();
                         $predicates="";
                         "<xsl:template
                              match=\"$c</xsl:template>";},
     'MESSAGE'    => sub{"...ERROR: $c"},
     'VALUE
     'CS'
     'ACTIO
     'CC'
     'VARIA
);
```

Algorithm:

- convert each constraint into a template
- use SELEXP for MATCH attribute
- convert CC into a predicate: [...]
- each “stamped” path will be converted into a predicate
- put MESSAGE contents inside template body
- filter text nodes

Grupo de Especificação e Processamento de Linguagens

Example2: linguistics

```
<CS>
  <CONSTRAINT>
    <SELECTOR SELEXP="//sentence"/>
    <CC><VARIABLE SELEXP="noun/@number"/>
              =<VARIABLE SELEXP="verb/@number"/>
    </CC>
    <ACTION>
       <MESSAGE>Attribute number of "<VALUE SELEXP="noun"/>"
       does not agree
        with attribute number of <VALUE SELEXP="verb"/>"!!!</MESSAGE>
    </ACTION>
  </CONSTRAINT>
  <CONSTRAINT>
    <SELECTOR SELEXP="//sentence"/>
    <CC><VARIABLE SELEXP="noun/@genre"/>=<VARIABLE SELEXP="adj/@genre"/></CC>
    <ACTION>
        <MESSAGE>:<VALUE SELEXP="noun"/> e <VALUE SELEXP="adj"/>
         do not  agree in genre!!!</MESSAGE>
    </ACTION>
  </CONSTRAINT>
</CS>
```

Grupo de Especificação e Processamento de Linguagens

Example2: linguistics

```
...
<xsl:template match="//sentence
 [not(noun/@number=verb/@number)][noun/@number][verb/@number]">
      ERROR: Attribute number of
      "<xsl:value-of select='noun'/>" does no
      with attribute number of <xsl:value-of
</xsl:template>

<xsl:template match="//sentence
     [not(noun/@genre=adj/@genre)][noun/@genre][adj/@genre]">
      ERROR: :<xsl:value-of select='noun'/> e
      <xsl:value-of select='adj'/> do not agree in genre!!!
</xsl:template>

<xsl:template match="//text()" priority="-1">
</xsl:template>
...
```

Ambiguity problem

Generated XSL stylesheet

XCSL: versions 2.1 and 2.2

- Version 2.1
  - Each CONSTI
    template
  - Templates wil
- Version 2.2
  - Added suppor
  - Enables two d

```
<xsl:template match="/">
<xsl:apply-templates
mode="const1"/>
<xsl:apply-templates
mode="const2"/>
<xsl:apply-templates
mode="const3"/>
<xsl:apply-templates
mode="const4"/>
</xsl:template>
```

Version 2.1 main skeleton

Example3: XDBML

Maintaining primary key invariant

```
<?xml version="1.0"?>
<CS>
 <CONSTRAINT>
  <SELECTOR SELEXP="//items/items-REG"/>

  <LET NAME="key" VALUE="code"/>

  <CC>count(//items/items-REG[code = $key]) = 1</CC>

  <ACTION>
   <MESSAGE>WARNING:
    code: <VALUE SELEXP="code"/> is not unique!</MESSAGE>
  </ACTION>
 </CONSTRAINT>
</CS>
```

XCSL v2.2 document

Grupo de Especificação e Processamento de Linguagens

Example3: XDBML

```
<xsl:template mode="const1" match="//items/items-REG">

  <xsl:variable name="key">
    <xsl:value-of select="code"/>
  </xsl:variable>

  <xsl:if test="not(count(//items/items-REG
                          [code = $key]) = 1)">
    WARNING:
        code: <xsl:value-of select='code'/> is not
        unique!
  </xsl:if>

  <xsl:apply-templates mode="const1"/>
</xsl:template>
```

Generated XSL stylesheet: main template

Grupo de Especificação e Processamento de Linguagens

Conclusions

- XML data and XCSL constraints can be joined in the same document
- Schemas or DTDs are not always necessary
- Other ideas coming from this presentation:
  - Do it simple
  - With existing technology

Grupo de Especificação e Processamento de Linguagens

Future Work

- To create a XSL processor generator
- To generate Perl (XML::DT) instead of XSL
  - Would allow to have Perl actions
  - Would allow to specify pattern matching constraints
- To do some reverse engineering on top of all:
  - Finding a suitable abstract representation for these contraints and the whole model (HOAG)

Grupo de Especificação e Processamento de Linguagens

Other resources

- My personnal webpage:
  - http://www.di.uminho.pt/~jcr
- My XML page with drafts, thesis, presentations, software, ... (always under construction!):
  - http://orunner.di.uminho.pt/PED/ped.html

Grupo de Especificação e Processamento de Linguagens

Q&A?

jcr@di.uminho.pt