#####

Table of contents

1.

(-- ############# --)

umsCodeGenerator###################### ############################## ####mapping definition ####################### #umsCodeGenerator################### ############################# ################ #######################

1. #########
2. mapping definition ###
3. ############
4. ###########
5. ####

######### mapping definition #####UMS################## ################## ### ############ mapping definition #############################

2. 1.

###########mapping definition ############### ################UMS#########################

3. 2. mapping definition

###mapping definition ######## ##################### mapping definition ######

```
<grammar xmlns="http://ums.isas.jaxa.jp/0.4">
  <start>
    <java:class scope="abstract" name="XmlSample"
xmlns:java="http://ums.isas.jaxa.jp/0.4/java">
      <defineVariables>
        <java:var name="rec" class="Record"/>
      </defineVariables>
    </java:class>
  </start>
</grammar>
```

Note:
####trang ####schema/*.rng #compact syntax ############# Emacs/nxml-mode ############ mapping definition # validate ###########

4. 3.

#####

############# decode#encode #################### #####mapping definition ############mapping definition ########## ################### ######XSLT ##################### ##############datatypeLibrary#################

############## main ############################

#########################

5. 4.

##################### mapping definition ########### #### mapping definition ############################## ########## ####XSLT ###################mapping definition # XSL ######### ## ########### RELAX NG ##################### XSLT ###########

```
<element name="a"/>
# <xsl:foreach select="a"/>
######a#################XSLT #a#################
###varidation#########
```

```
<zeroOrMore>
  <element name="a"/>
</zeroOrMore>
# <xsl:foreach select="a"/>
```

```
<optional>
  <attribute name="@a"/>
</optional>
# <xsl:if test="@a">#####</xsl:if>
```

```
<choice/>
# <xsl:choose/>
```

```
<define name="xxx"/> # <ref name="xxx"/> ##
# <template name="xxx"/> # <call-template name="xxx"/> ##
```

XSLT

```
<element name="a">
  <element name="b"/>
  <element name="c"/>
</element>
# <xsl:for-each select="a"><xsl:for-each
select="b"/></xsl:for-each>
# <xsl:for-each select="a"><xsl:for-each
```

```
select="c"/></xsl:for-each>
```

############# XSLT #######

```
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format"
xmlns:ums="http://ums.isas.jaxa.jp/0.4"
xmlns:java="http://ums.isas.jaxa.jp/0.4/java">
  <xsl:output method="text"/>
  <xsl:template match="/">
    <xsl:for-each select="ums:grammar">
      <xsl:for-each select="ums:start">
        <xsl:for-each select="java:class"><xsl:text></xsl:text>
          <xsl:if test="@scope">
            <xsl:value-of select="@scope"/>
            <xsl:text> </xsl:text>
          </xsl:if>class <xsl:value-of select="@name"/> {
          <xsl:for-each select="ums:defineVariables">
            <xsl:for-each select="java:var">
              <xsl:choose>
                <xsl:when test="@type">
                  <xsl:value-of select="@type"/>
                </xsl:when>
                <xsl:when test="@class">
                  <xsl:value-of select="@class"/>
                </xsl:when>
              </xsl:choose>
              <xsl:text> </xsl:text>
              <xsl:value-of select="@name"/>;
            </xsl:for-each>
          </xsl:for-each>
}
        </xsl:for-each>
      </xsl:for-each>
    </xsl:for-each>
  </xsl:template>
</xsl:stylesheet>
```

6. 5.

XSLT ##### XSLT ######mapping definition
###