SmartSlides Project Wide Standard

T. Davidson
WaveMedia

11th March 2015

Contents

1 Rules

1.1 XML Structure

- The top level element is the slideshow element.
- The slideshow element must include both documentinfo and deafualtsettings elements. These elements must occur in the order documentinfo, deafultsettings.
- The slideshow element must include 1 or more slide elements.
- The documentinfo element must include author, version, comment and groupid elements. These elements may occur in any order.
- The defaultsettings element must include backgroundcolor, font, fontsize and fontcolor elements. These elements may occur in any order.
- The slide element can include any number of text, image, audio, video and graphics elements. These elements can occur in any order.
- The text element must include xstart and ystart attributes.
- The text element can include sourcefile, font, fontsize, fontcolor and duration attributes.
- The text element can be an empty element, if a source file is specified.
- The image element cannot include any child elements.
- The image element must include sourcefile, xstart and ystart attributes.
- The image element can include scale, duration and starttime attributes.
- The audio element cannot include any child elements.
- The audio element must include a sourcefile attribute.
- The audio element can include a starttime attribute.
- The video element cannot include any child elements.
- The video element must include sourcefile, xstart and ystart attributes.
- The graphics element must include type, xstart, ystart, xend, yend, solid and graphiccolor attributes.
- The graphics element can include a duration attribute.
- The graphics element can include a cyclicshading element.
- The cylclicshading element cannot include any elements.
- The cyclicshading element must include a shadingcolor attribute.

1.2 Additional Rules

- XML, and therefore the project wide standard, is case sensitive. All element names, attribute names and data must be written purely in lower case characters.
- Color is always spelt color, not colour.
- All colors are described as 8 digit hex strings in ARGB format. (The alpha channel is the first two digits).
- All position values are relative to screen size, measured from the top left corner of the screen, and describe the top left corner of the object. The value is a float data type, in the range 0.0 - 1.0.
- All duration values are given in seconds. The value is a float data type.
- Where a duration value is not given an infinite duration is implied.
- A group ID value of 0 specifies that the file uses the standard format, with no group's extensions.
- All font sizes are integer pt sizes.
- The standard supports both 'arial' and 'times new roman' fonts.
- Slides occur in the XML file in sequential order, first to last.
- Text source files should be of type '.txt'.
- Image source files should be of type '.png' or '.jpg'.
- Audio source files should be of type '.wav'.
- Video source files should be of type '.mkv'.
- Image scale attribute is relative to the dimensions of the source file. The value is a float data type greater than 0.
- The graphic 'type' attribute supports the following values: oval, rectangle, line.

2 Text Handler

Five different XML text standards need to be supported, which are shown below.

The hierarchy for settings is program defaults, slideshow defaults(if the parameter exists), text element, richtext element.

The hierarchy for what text is printed is text from sourcefile, then text within XML file.

2.1 Text included with a null string for sourcefile

```
<text sourcefile="null" xstart="0.5" ystart="0.5">
    Some Text.
</text>

    Some Text.

(If sourcefile is null, display chars inside text)
```

2.2 Text included with no sourcefile attribute

```
<text xstart="0.5" ystart="0.5">
    Some Text.
</text>

    Some Text.

(If sourcefile attribute is missing, display chars inside text)
```

2.3 Text from sourcefile

```
<text sourcefile="text.txt" xstart="0.5" ystart="0.5"/>

    (Text from sourcefile)

Perhaps wrap the thing.
```

2.4 Text from list of richtext elements

```
<text xstart="0.5" ystart="0.5">
    <richtext newline="true">Example text!</richtext>
    <richtext newline="false">Hello</richtext>
    <richtext newline="true">World!</richtext>
</text>

Example Text!
HelloWorld!
```

3 Graphics Handler

Below is a list of all shapes supported by the Graphics Handler, and the parameters that can be varied on each shape.

3.1 Common Parameters

Each shape requires these values specified.

- xStartPos - x coordinate of the top left corner of the bounding box around the shape.
- yStartPos - y coordinate of the top left corner of the bounding box around the shape.

3.2 Oval

- xEndPos - x coordinate of the bottom right corner of the bounding box around the oval.
- yEndPos - y coordinate of the bottom right corner of the bounding box around the oval.
- color - color of the oval. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the oval is an outline or a solid shape.
- outlineColor - outline color of the oval. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the oval outline.
- shadowType - the amount of shadow on the oval.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the oval.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.3 Circle

- radius - the radius of the circle.
- color - color of the circle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the circle is an outline or a solid shape.
- outlineColor - outline color of the circle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the circle outline.
- shadowType - the amount of shadow on the circle.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- shadingType - the type of shading to be applied to the circle.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.

- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.4 Rectangle

- xEndPos - x coordinate of the bottom right corner of the bounding box around the rectangle.
- yEndPos - y coordinate of the bottom right corner of the bounding box around the rectangle.
- arcWidth - the vertical diameter of the arc at the four corners of the rectangle.
- arcHeight - the horizontal diameter of the arc at the four corners of the rectangle.
- color - color of the rectangle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the rectangle is an outline or a solid shape.
- outlineColor - outline color of the rectangle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the rectangle outline.
- shadowType - the amount of shadow on the rectangle.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the rectangle.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.5 Square

- length - the length of each side of the square.
- color - color of the rectangle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the square is an outline or a solid shape.
- outlineColor - outline color of the square. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the square outline.
- shadowType - the amount of shadow on the square.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the square.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.6 Line

- xStartPos - x coordinate of the start of the line.
- yStartPos - y coordinate of the start of the line.
- xEndPos - x coordinate of the end of the line.
- yEndPos - y coordinate of the end of the line.
- color - color of the line. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- thickness - thickness of the line.
- shadingType - the type of shading to be applied to the line.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.7 Arrow

- xStartPos - x coordinate of the start of the arrow.
- yStartPos - y coordinate of the start of the arrow.
- xEndPos - x coordinate of the end of the arrow.
- yEndPos - y coordinate of the end of the arrow.
- color - color of the arrow. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- shadingType - the type of shading to be applied to the arrow.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.8 Equilateral Triangle

- length - the side length of the triangle
- color - color of the triangle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the triangle is an outline or a solid shape.
- outlineColor - outline color of the triangle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the triangle outline.
- shadowType - the amount of shadow on the triangle.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.

- shadingType - the type of shading to be applied to the equilateral triangle.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.9 Triangle

- triangleCoordinates - all 6 coordinates of the 3 corners of the triangle.
- color - color of the triangle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the triangle is an outline or a solid shape.
- outlineColor - outline color of the triangle. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the triangle outline.
- shadowType - the amount of shadow on the triangle.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the triangle.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.10 N Sided Regular Polygon

- width - the width of the polygon.
- height - the height of the polygon.
- numberOfSides - the number of sides of the polygon.
- color - color of the polygon. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the polygon is an outline or a solid shape.
- outlineColor - outline color of the polygon. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the polygon outline.
- shadowType - the amount of shadow on the polygon.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the polygon.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.11 Polygon

- polygonCoordinate - coordinates of one of the corners of the polygon.
- xStartPos - x coordinate of the top left corner of the bounding box around the polygon. All the x coordinates are relative to this position.
- yStartPos - y coordinate of the top left corner of the bounding box around the polygon. All the y coordinates are relative to this position.
- color - color of the polygon. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the polygon is an outline or a solid shape.
- outlineColor - outline color of the polygon. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the polygon outline.
- shadowType - the amount of shadow on the polygon.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the polygon.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.12 Star

- numberOfPoints - the number of points of the star.
- color - color of the star. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the star is an outline or a solid shape.
- outlineColor - outline color of the star. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the star outline.
- shadowType - the amount of shadow on the star.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the star.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.13 Chord

- width - the overall width (horizontal radius) of the full ellipse of which the arc in the chord is a partial section.
- height - the overall height (vertical radius) of the full ellipse of which the arc in the chord is a partial section.
- arcAngle - the starting angle of the arc in the chord in degrees.
- length - the angular extent of the arc in the chord in degrees.
- color - color of the chord. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the chord is an outline or a solid shape.
- outlineColor - outline color of the chord. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the chord outline.
- shadowType - the amount of shadow on the chord.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the chord.
  Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.
- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

3.14 Arc

- width - the overall width (horizontal radius) of the full ellipse of which this arc is a partial section.
- height - the overall height (vertical radius) of the full ellipse of which this arc is a partial section.
- arcAngle - the starting angle of the arc in degrees.
- length - the angular extent of the arc in degrees.
- color - color of the arc. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- solid - boolean value of if the arc is an outline or a solid shape.
- outlineColor - outline color of the arc. The accepted string format is a #, followed by a 8 bit hex number in ARGB format.
- outlineThickness - thickness of the arc outline.
- shadowType - the amount of shadow on the arc.
  Options: Shadow.NONE, Shadow.LIGHT, Shadow.NORMAL and Shadow.HEAVY.
- rotation - amount of rotation about the center in degrees.
- shadingType - the type of shading to be applied to the rectangle. Options: Shading.NONE, Shading.CYCLIC, Shading.HORIZONTAL, Shading.VERTICAL.

- shadingElement - a element that describes the color at a position in the shape. Contains a shadingColor in the regular color string format, and a offset that is a relative (0 to 1) distance through the shape that the color should appear. This element can occur multiple times per shape.

4 Image Handler

5 Example XML file

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

<slideshow>
    <documentinfo>
        <author>Tom Davidson</author>
        <version>1.0</version>
        <comment>this is a comment</comment>
        <groupid>2</groupid>
    </documentinfo>
    <defaultsettings>
        <backgroundcolor>#ff00ff00</backgroundcolor>
        <font>times new roman</font>
        <fontsize>24</fontsize>
        <fontcolor>#ffcccc00</fontcolor>
        <graphiccolor>#ffaaaa00</graphiccolor>
    </defaultsettings>
    <slide>
        <text sourcefile="null" xstart="0.5" ystart="0.5" font="arial" fontsize="24" fontcolor="#00112233"  duration="1.5">
            Some Text.
        </text>

        <text xstart="0.5" ystart="0.5" font="arial" fontsize="24" fontcolor="#00112233" duration="1.5" starttime="2.0">
            <richtext font="arial" fontsize="26" fontcolor="#11223344" newline="true" b="true" i="true" u="true" strikethrough="true" superscript="true" subscript="true" case="upper/lower/camel" alignment="left/right/centre">Example text!</richtext>
            <richtext font="arial" fontsize="26" fontcolor="#11223344" newline="true" b="true" i="true" u="true" strikethrough="true" superscript="true" subscript="true" case="upper/lower/camel" alignment="left">Example text!</richtext>
            <richtext font="arial" fontsize="26" fontcolor="#11223344" newline="true" b="true" i="true" u="true" strikethrough="true" superscript="true" subscript="true" case="upper/lower/camel" alignment="left">Example text!</richtext>
        </text>

        <text xstart="0.5" ystart="0.5" font="arial" fontsize="24" fontcolor="#00112233" duration="1.5" starttime="2.0">
            <richtext font="arial" fontsize="26" fontcolor="#11223344" newline="true" b="true" i="true" u="true" strikethrough="true" superscript="true" subscript="true" case="upper/lower/camel" alignment="left">Example text!</richtext>
            <richtext font="arial" fontsize="26" fontcolor="#11223344" newline="true" b="true" i="true" u="true" strikethrough="true" superscript="true" subscript="true" case="upper/lower/camel" alignment="left">Example text!</richtext>
        </text>

        <image sourcefile="cat.jpg" xstart="0.5" ystart="0.5" scale="1.0" duration="1.5" starttime="2.0" />

        <image sourcefile="selfie.jpg" xstart="0.5" ystart="0.5" scale="1.0" duration="1.5" starttime="2.0" rotate="90" fliphorizontal="true" flipvertical="true" cropx1="0.0" cropy1="0.0" cropx2="0.5" cropy2="0.5"/>

        <audio sourcefile="gavel.wav" starttime="2.5" />
        <video sourcefile="avengers.mkv" xstart="0.5" ystart="0.5" />

        <!-- PWS Shapes -->
        <graphic type="oval/rectangle/line" xstart="0.5" ystart="0.5" xend="0.7" yend="0.7" solid="true" graphiccolor="#11223344" duration="1.5">
            <cyclicshading shadingcolor="#88776655" />
        </graphic>


        <!-- Our Expanded PWS Shapes -->
        <graphic graphiccolor="#11223344" duration="1.5" starttime="2.0">
            <oval xstart="0.5" ystart="0.5" xend="0.7" yend="0.7" solid="true" outlinecolor="#00000000" outlinethickness="1.0" rotation="45"/>
            <shadow weight="heavy/normal/light"/>
            <cyclicshading shadingcolor="#88776655">
                <shading shadingcolor="#00aabbcc"/>
                <shading shadingcolor="#00aabbcc"/>
            <cyclicshading/>
        </graphic>

        <graphic graphiccolor="#11223344" duration="1.5" starttime="2.0">
            <rectangle xstart="0.5" ystart="0.5" xend="0.7" yend="0.7" arcwidth="10" archeight="10" solid="true" outlinecolor="#00000000" outlinethickness="1.0" rotation="45"/>
        </graphic>

        <graphic graphiccolor="#11223344" duration="1.5" starttime="2.0">
            <line xstart="0.5" ystart="0.5" xend="0.7" yend="0.7" outlinethickness="1.0"/> <!-- I’m happy with calling this outline thickness... -->
        </graphic>


        <!-- Our Added Shapes -->

        <graphic graphiccolor="#11223344" duration="1.5" starttime="2.0">
            <circle xstart="0.5" ystart="0.5" size="10" solid="true" outlinecolor="#00000000" outlinethickness="1.0"/>
            <shadow weight="heavy/normal/light"/>
            <cyclicshading shadingcolor="#88776655">
                <shading shadingcolor="#00aabbcc"/>
                <shading shadingcolor="#00aabbcc"/>
            <cyclicshading/>
        </graphic>

        <graphic graphiccolor="#11223344" duration="1.5" starttime="2.0">
```

```
            <square xstart="0.5" ystart="0.5" size="10" solid="true" outlinecolor="#00000000" outlinethickness="1.0">
        </graphic>

        <graphic graphiccolor="#11223344" duration="1.5" starttime="2.0">
            <itriangle xstart="0.5" ystart="0.5" xend="0.7" yend="0.7" solid="true"/>
        </graphic>
    </slide>
</slideshow>
```

6 Schema

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" attributeFormDefault="unqualified" elementFormDefault="qualified">
  <xs:element name="slideshow">
    <xs:complexType>
      <xs:sequence>
        <xs:sequence>
          <xs:element name="documentinfo">
            <xs:complexType>
              <xs:all>
                <xs:element type="xs:string" name="author" />
                <xs:element type="xs:string" name="version" />
                <xs:element type="xs:string" name="comment" />
                <xs:element type="xs:integer" name="groupid" />
              </xs:all>
            </xs:complexType>
          </xs:element>
          <xs:element name="defaultsettings">
            <xs:complexType>
              <xs:all>
                <xs:element name="backgroundcolor">
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:pattern value="[#]([a-fA-F0-9]{8})" />
                    </xs:restriction>
                  </xs:simpleType>
                </xs:element>
                <xs:element type="xs:string" name="font" />
                <xs:element type="xs:positiveInteger" name="fontsize" />
                <xs:element name="fontcolor">
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:pattern value="[#]([a-fA-F0-9]{8})" />
                    </xs:restriction>
                  </xs:simpleType>
                </xs:element>
                <xs:element name="graphiccolor">
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:pattern value="[#]([a-fA-F0-9]{8})" />
                    </xs:restriction>
                  </xs:simpleType>
                </xs:element>
              </xs:all>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
        <xs:sequence minOccurs="1" maxOccurs="unbounded">
          <xs:element name="slide">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="text">
                  <xs:complexType mixed="true">
                    <xs:choice minOccurs="0" maxOccurs="unbounded">
                      <xs:element name="richtext">
                        <xs:complexType>
                          <xs:simpleContent>
                            <xs:extension base="xs:string">
                              <xs:attribute type="xs:string" name="font" />
                              <xs:attribute type="xs:string" name="fontsize" />
                              <xs:attribute type="xs:string" name="fontcolor" />
                              <xs:attribute type="xs:boolean" name="newline" />
                              <xs:attribute type="xs:boolean" name="b" />
                              <xs:attribute type="xs:boolean" name="i" />
                              <xs:attribute type="xs:boolean" name="u" />
                              <xs:attribute type="xs:boolean" name="strikethrough" />
                              <xs:attribute type="xs:boolean" name="superscript" />
                              <xs:attribute type="xs:boolean" name="subscript" />
                              <xs:attribute type="xs:string" name="case" />
                              <xs:attribute type="xs:string" name="alignment" />
                            </xs:extension>
                          </xs:simpleContent>
                        </xs:complexType>
                      </xs:element>
                    </xs:choice>
                    <xs:attribute type="xs:string" name="sourcefile" />
                    <xs:attribute type="xs:float" name="xstart" use="required" />
                    <xs:attribute type="xs:float" name="ystart" use="required" />
                    <xs:attribute type="xs:string" name="font" />
                    <xs:attribute type="xs:integer" name="fontsize" />
                    <xs:attribute type="xs:string" name="fontcolor" />
                    <xs:attribute type="xs:float" name="duration" />
                    <xs:attribute type="xs:float" name="starttime" />
                  </xs:complexType>
                </xs:element>
                <xs:element name="image">
                  <xs:complexType>
                    <xs:simpleContent>
                      <xs:extension base="xs:string">
                        <xs:attribute type="xs:string" name="sourcefile" use="required" />
                        <xs:attribute type="xs:float" name="xstart" use="required" />
                        <xs:attribute type="xs:float" name="ystart" use="required" />
                        <xs:attribute type="xs:float" name="scale" />
                        <xs:attribute type="xs:float" name="duration" />
                        <xs:attribute type="xs:float" name="starttime" />
                        <xs:attribute type="xs:float" name="rotate" />
                        <xs:attribute type="xs:boolean" name="fliphorizontal" />
                        <xs:attribute type="xs:boolean" name="flipvertical" />
                        <xs:attribute type="xs:float" name="cropx1" />
                        <xs:attribute type="xs:float" name="cropy1" />
                        <xs:attribute type="xs:float" name="cropx2" />
                        <xs:attribute type="xs:float" name="cropy2" />
                      </xs:extension>
                    </xs:simpleContent>
                  </xs:complexType>
                </xs:element>
                <xs:element name="audio">
                  <xs:complexType>
                    <xs:simpleContent>
                      <xs:extension base="xs:string">
                        <xs:attribute type="xs:string" name="sourcefile" use="required" />
                        <xs:attribute type="xs:float" name="starttime" />
                      </xs:extension>
                    </xs:simpleContent>
                  </xs:complexType>
                </xs:element>
                <xs:element name="video">
                  <xs:complexType>
                    <xs:simpleContent>
                      <xs:extension base="xs:string">
                        <xs:attribute type="xs:string" name="sourcefile" use="required" />
                        <xs:attribute type="xs:float" name="xstart" use="required" />
                        <xs:attribute type="xs:float" name="ystart" use="required" />
                      </xs:extension>
                    </xs:simpleContent>
                  </xs:complexType>
                </xs:element>
                <xs:element name="graphic">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:choice minOccurs="0">
                        <xs:element name="oval">
                          <xs:complexType>
                            <xs:simpleContent>
                              <xs:extension base="xs:string">
                                <xs:attribute type="xs:float" name="xstart" use="required" />
                                <xs:attribute type="xs:float" name="ystart" use="required" />
                                <xs:attribute type="xs:float" name="xend" use="required" />
                                <xs:attribute type="xs:float" name="yend" use="required" />
                                <xs:attribute type="xs:boolean" name="solid" />
                              </xs:extension>
                            </xs:simpleContent>
                          </xs:complexType>
                        </xs:element>
                        <xs:element name="rectangle">
                          <xs:complexType>
                            <xs:simpleContent>
```

```
                              <xs:extension base="xs:string">
                                <xs:attribute type="xs:float" name="xstart" use="required" />
                                <xs:attribute type="xs:float" name="ystart" use="required" />
                                <xs:attribute type="xs:float" name="xend" use="required" />
                                <xs:attribute type="xs:float" name="yend" use="required" />
                                <xs:attribute type="xs:boolean" name="solid" />
                              </xs:extension>
                            </xs:simpleContent>
                          </xs:complexType>
                        </xs:element>
                        <xs:element name="line">
                          <xs:complexType>
                            <xs:simpleContent>
                              <xs:extension base="xs:string">
                                <xs:attribute type="xs:float" name="xstart" use="required" />
                                <xs:attribute type="xs:float" name="ystart" use="required" />
                                <xs:attribute type="xs:float" name="xend" use="required" />
                                <xs:attribute type="xs:float" name="yend" use="required" />
                              </xs:extension>
                            </xs:simpleContent>
                          </xs:complexType>
                        </xs:element>
                        <xs:element name="itriangle">
                          <xs:complexType>
                            <xs:simpleContent>
                              <xs:extension base="xs:string">
                                <xs:attribute type="xs:float" name="xstart" use="required" />
                                <xs:attribute type="xs:float" name="ystart" use="required" />
                                <xs:attribute type="xs:float" name="xend" use="required" />
                                <xs:attribute type="xs:float" name="yend" use="required" />
                                <xs:attribute type="xs:boolean" name="solid" />
                              </xs:extension>
                            </xs:simpleContent>
                          </xs:complexType>
                        </xs:element>
                      </xs:choice>
                      <xs:choice minOccurs="0">
                        <xs:element name="cyclicshading">
                          <xs:complexType>
                            <xs:simpleContent>
                              <xs:extension base="xs:string">
                                <xs:attribute type="xs:string" name="shadingcolor" use="required"
    />
                              </xs:extension>
                            </xs:simpleContent>
                          </xs:complexType>
                        </xs:element>
                      </xs:choice>
                    </xs:sequence>
                    <xs:attribute type="xs:string" name="type" />
                    <xs:attribute type="xs:float" name="xstart" />
                    <xs:attribute type="xs:float" name="ystart" />
                    <xs:attribute type="xs:float" name="xend" />
                    <xs:attribute type="xs:float" name="yend" />
                    <xs:attribute type="xs:boolean" name="solid" />
                    <xs:attribute type="xs:string" name="graphiccolor" />
                    <xs:attribute type="xs:float" name="duration" />
                    <xs:attribute type="xs:float" name="starttime" />
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```