

X3D Graphics for Web Authors

Chapter 2

Geometry 1: Primitive Shapes

Dorothy in Oz: "Toto, I've a feeling we're not in Kansas anymore."

—L. Frank Baum, Wizard of Oz, 1939

Table of Contents

Chapter Overview and Concepts

X3D Nodes and Examples

Additional Resources

Chapter Summary and Suggested Exercises

References

Chapter Overview

Overview: Geometry 1, Primitive Shapes

Common pattern for Shape nodes

- Shape contains geometry node
- Appearance and Material nodes

Five nodes for primitive geometry in this chapter

- Box, Cone, Cylinder, Sphere, Text
- Text node is flat, not extruded
- FontStyle modifies Text node parameters

X3D tooltips and specifications are helpful to use

Concepts

Shape and geometry

Shape nodes can contain a single geometry node

- For example, one of the five geometry primitive nodes
- Alternatively contains a more-advanced geometry node
 - Chapter 2: Geometric primitives
 - Chapter 6: Points Lines and Polygon nodes
 - Chapter 10: Geometry2D nodes
 - Chapter 13: Triangle nodes

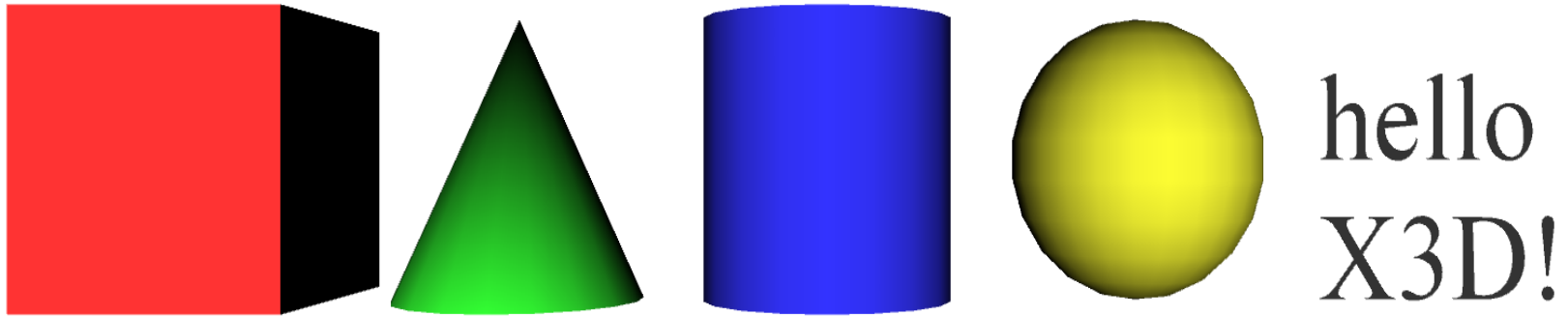
Shape nodes can also contain an Appearance node

- Which in turn contains a Material node for coloring
- Covered in Chapter 3

Why is this pattern fundamental?

- Common design pattern throughout X3D:
 - **Shape**
 - **GeometryNode**
 - **Appearance**
 - **Material** (optional) for colors
 - **ImageTexture** (optional) for wrapping an image file
- Top three priorities in graphics design:
performance performance performance!!!
- This pattern is repeated in order to directly represent geometry and appearance together for maximum graphics-card performance

Geometry Primitives



Primitives are simple geometric constructs

Shape, geometry, Appearance, Material pattern

Browsers decide implementation details, including tessellation (polygon count) and thus quality

Common field: *solid*

In 3D graphics, all triangles have 2 sides

- Graphics term: backface culling only draws front sides

The *solid* field defines whether a geometry node has an inside or not, with a default value of true

- *solid*='true' means do not render (draw) the inside
- *solid*='false' means render both inside and outside

This approach reduces the number of polygons needing to be drawn, thus improving performance

Confusing if user gets lost inside invisible geometry

- **Hint:** set *solid*='false' to draw both sides

X3D Nodes and Examples

Shape parent with geometry child

```
<Shape>  
  <Box size='1 2 3'/>  
  <Appearance>  
    <Material/>  
  </Appearance>  
</Shape>
```

Shape must be parent node, can only hold one geometry node
Appearance and Material nodes define colors, transparency, etc.

```
<Shape>  
  <Sphere radius='1'/>  
  <Appearance>  
    <Material/>  
  </Appearance>  
</Shape>
```

Primitives have simple dimensions

- Typical volume ~ 1 m radius

All units are in meters
Note parent-child relationships

Box node

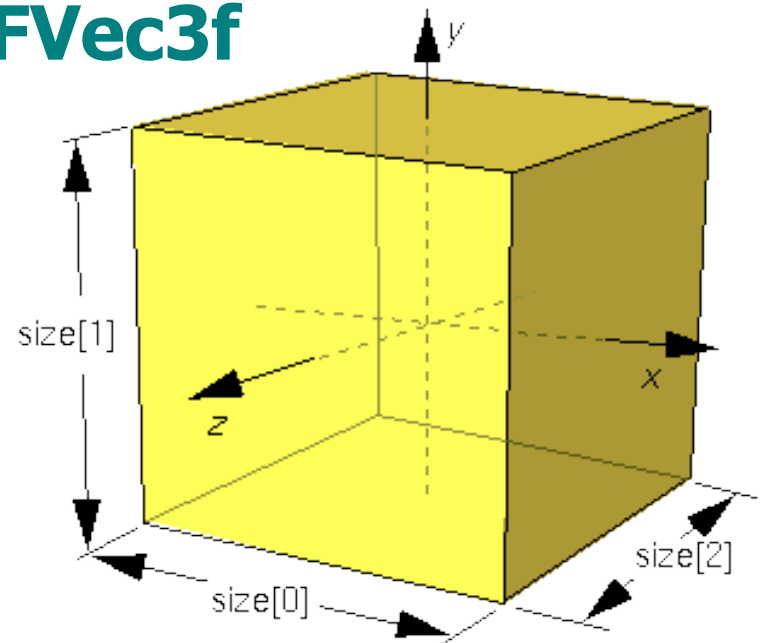
Six-sided rectangular parallelepiped

- meaning: not necessarily a cube, but it can be
- Three non-zero non-negative *size* dimensions for x y z

Centered at local origin

size field has X3D data type **SFVec3f**

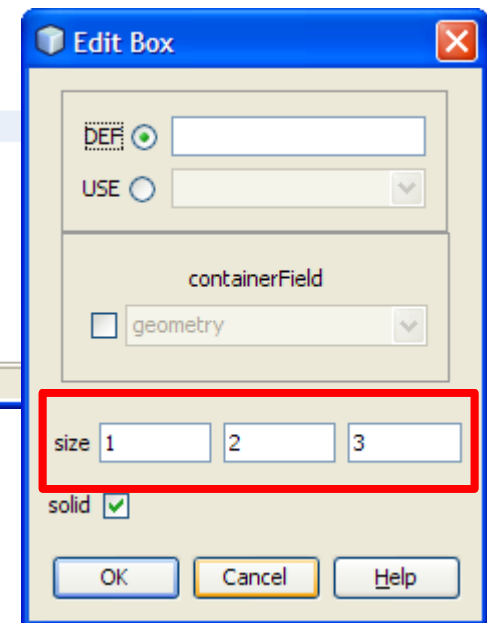
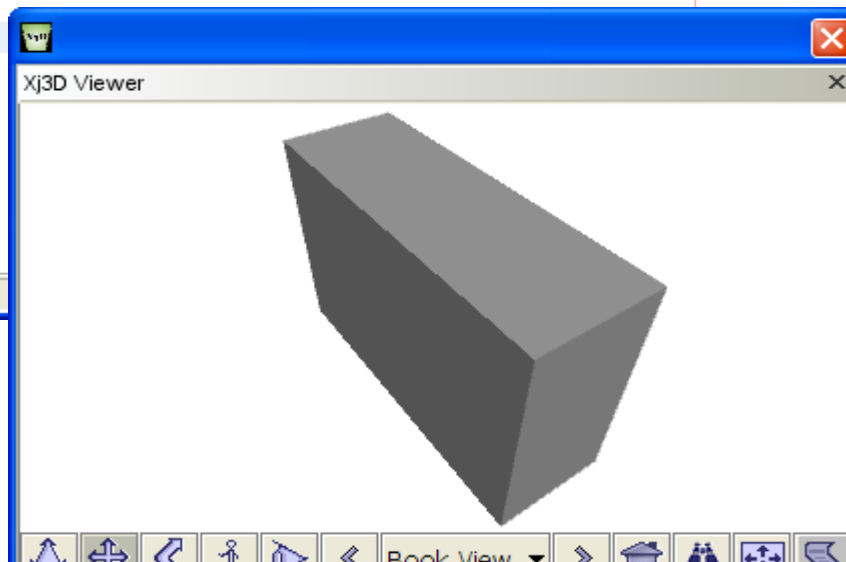
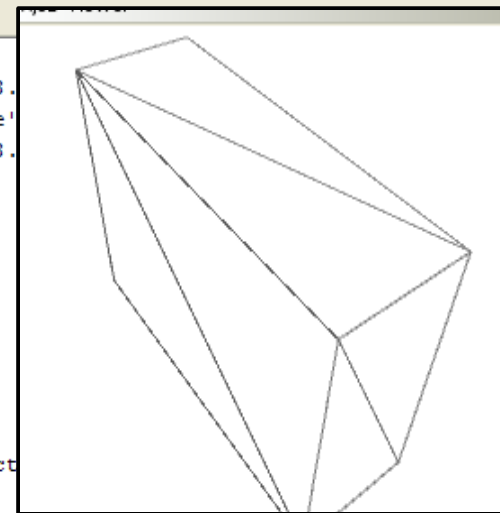
- **SF Vec** = Single-field vector
- array length of 0 or 1 only
- **3f** = 3 floating-point values
- Default *size*='2 2 2'

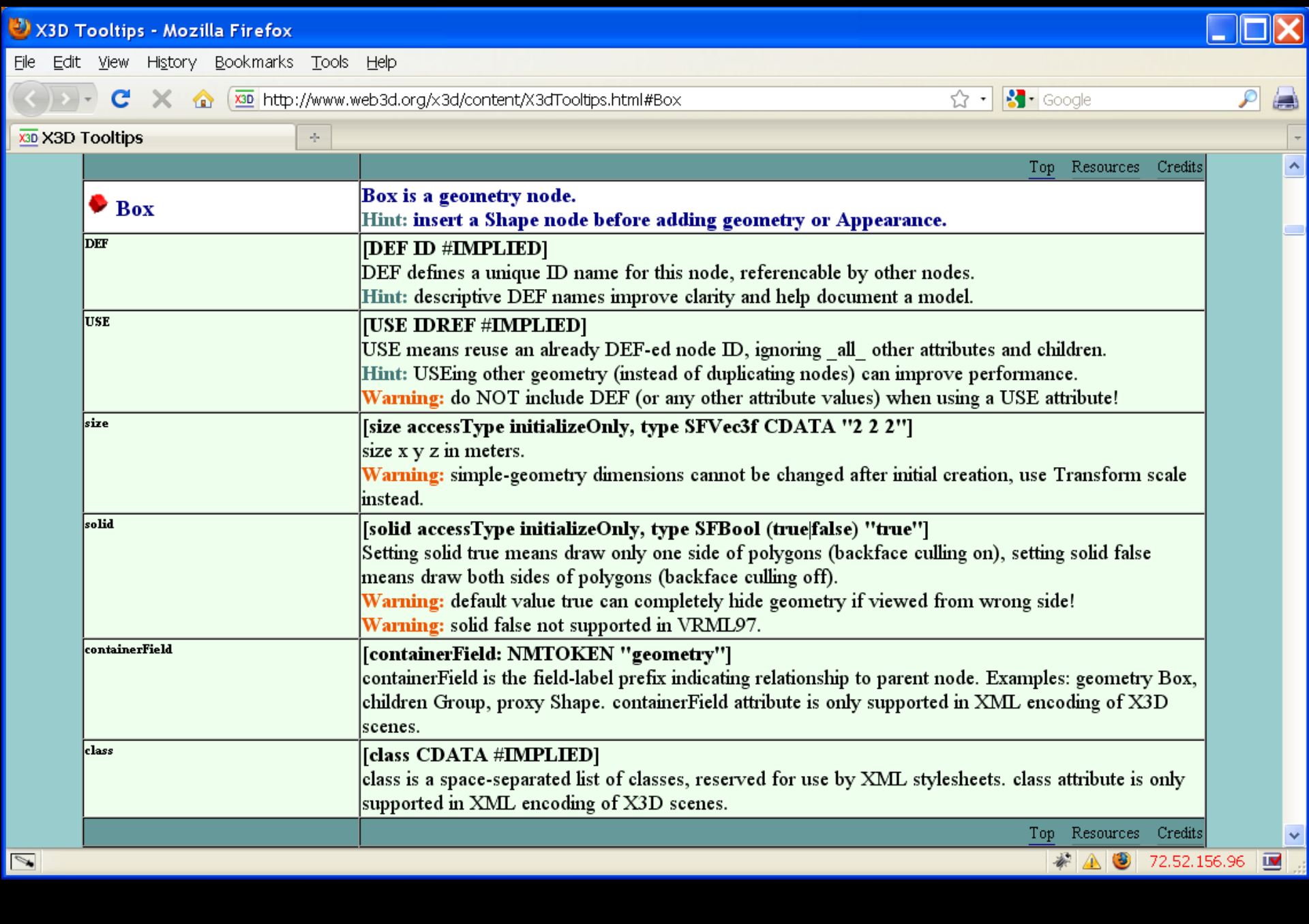



```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.1//EN" "http://www.web3d.org/specifications/x3d-3.1" >
<X3D profile='Immersive' version='3.1' xmlns:xsd='http://www.w3.org/2001/XMLSchema-instance'
      xsd:noNamespaceSchemaLocation='http://www.web3d.org/specifications/x3d-3.1' >
  <head>
    <meta content='Box.x3d' name='title' />
    <meta content='Box geometric primitive node.' name='description' />
    <meta content='Leonard Daly' name='creator' />
    <meta content='1 January 2007' name='created' />
    <meta content='27 March 2007' name='modified' />
    <meta content='http://X3dGraphics.com' name='reference' />
    <meta content='http://www.web3d.org/x3d/content/examples/help.html' name='reference' />
    <meta content='Copyright Don Brutzman and Leonard Daly 2007' name='rights' />
    <meta content='X3D book, X3D graphics, X3D-Edit, http://www.x3dGraphics.com' name='subject' />
    <meta name='identifier'
          content='http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Box.x3d' />
    <meta content='X3D-Edit, https://savage.nps.edu/X3D-Edit' name='generator' />
    <meta content='../license.html' name='license' />
  </head>
  <Scene>
    <Background skyColor='1 1 1' />
    <Viewpoint description='Book View' position='-1.81 3.12 2.59'
              orientation='-0.747 -0.624 -0.231 1.05' />
    <Shape>
      <Box size='1 2 3' />
      <Appearance>
        <Material />
      </Appearance>
    </Shape>
  </Scene>
</X3D>

```





 Box	<p>Box is a geometry node. Hint: insert a Shape node before adding geometry or Appearance.</p>
DEF	<p>[DEF ID #IMPLIED] DEF defines a unique ID name for this node, referencable by other nodes. Hint: descriptive DEF names improve clarity and help document a model.</p>
USE	<p>[USE IDREF #IMPLIED] USE means reuse an already DEF-ed node ID, ignoring <u>all</u> other attributes and children. Hint: USEing other geometry (instead of duplicating nodes) can improve performance. Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!</p>
size	<p>[size accessType initializeOnly, type SFVec3f CDATA "2 2 2"] size x y z in meters. Warning: simple-geometry dimensions cannot be changed after initial creation, use Transform scale instead.</p>
solid	<p>[solid accessType initializeOnly, type SFBool (true false) "true"] Setting solid true means draw only one side of polygons (backface culling on), setting solid false means draw both sides of polygons (backface culling off). Warning: default value true can completely hide geometry if viewed from wrong side! Warning: solid false not supported in VRML97.</p>
containerField	<p>[containerField: NMTOKEN "geometry"] containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.</p>
class	<p>[class CDATA #IMPLIED] class is a space-separated list of classes, reserved for use by XML stylesheets. class attribute is only supported in XML encoding of X3D scenes.</p>

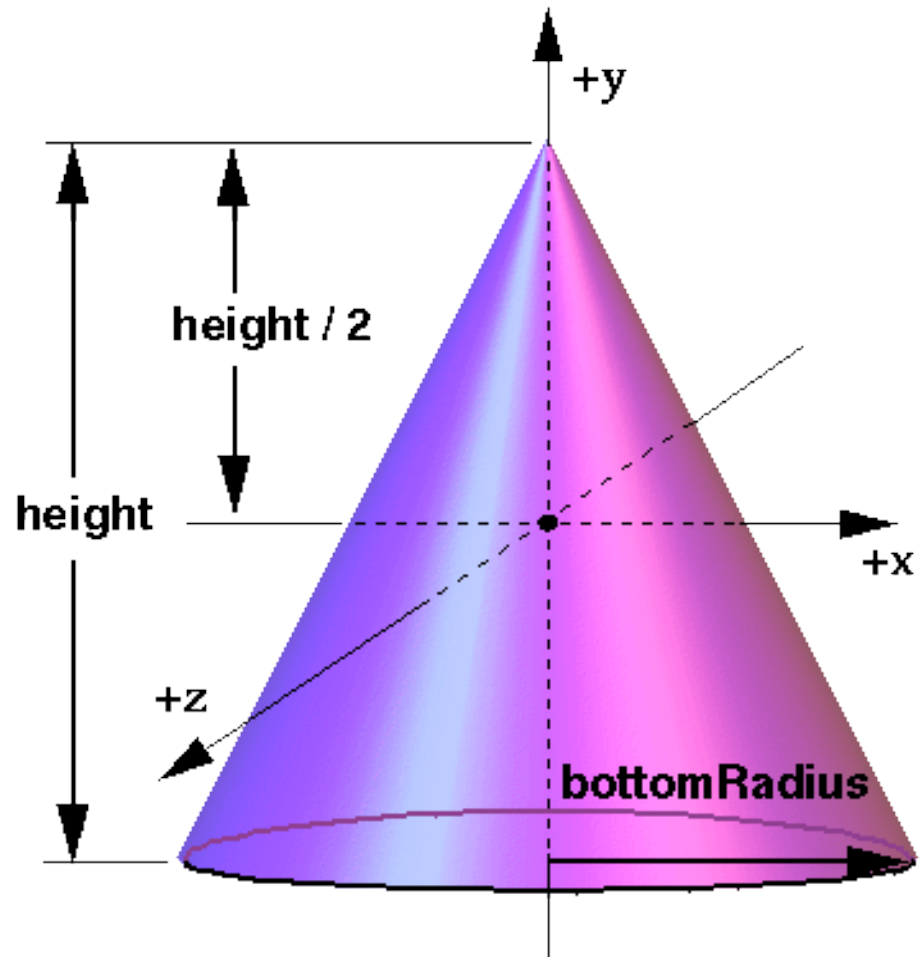
Cone node

Circular *bottomRadius*
non-zero non-negative
height above bottom
Centered at local origin
Can hide different parts

- *side*='false'
- *bottom*='false'

Default *height*='2'
bottomRadius='1'

Set *side*='false' (for bottom only) to define flat circle



Cone.x3d - Editor

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.1//EN" "http://www.web3d.org/specifications/x3d-3.1.dtd">
<X3D profile='Immersive' version='3.1' xmlns:xsd='http://www.w3.org/2001/XMLSchema-instance'
  xsd:noNamespaceSchemaLocation='http://www.web3d.org/specifications/x3d-3.1.xsd' >
  <head>
    <meta content='Cone.x3d' name='title' />
    <meta content='Cone geometric primitive node.' name='description' />
    <meta content='Leonard Daly and Don Brutzman' name='creator' />
    <meta content='1 January 2007' name='created' />
    <meta content='26 December 2007' name='modified' />
    <meta content='http://X3dGraphics.com' name='reference' />
    <meta content='http://www.web3d.org/x3d/content/examples/help.html' name='reference' />
    <meta content='Copyright Don Brutzman and Leonard Daly 2007' name='rights' />
    <meta content='X3D book, X3D graphics, X3D-Edit, http://www.x3dgraphics.com' name='subject' />
    <meta name='identifier'
      content='http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Cone.x3d' />
    <meta content='X3D-Edit, https://savage.nps.edu/X3D-Edit' name='generator' />
    <meta content='../license.html' name='license' />
  </head>
  <Scene>
    <Background skyColor='1 1 1' />
    <Viewpoint description='Side View' position='0 0 4' />
    <Viewpoint description='Book View' orientation='-1 0 0 0.68' position='0 1.11 1.9' />
    <Shape>
      <Cone bottom='true' bottomRadius='1' height='1' side='true' />
      <Appearance>
        <Material />
      </Appearance>
    </Shape>
  </Scene>
</X3D>

```

25:68 | INS

Cone.x3d

Edit Cone

DEF

USE

containerField

geometry

bottom

bottomRadius

height


side

solid

OK Cancel Help

X3D Viewer

Side View

 Cone	<p>Cone is a geometry node. Hint: insert a Shape node before adding geometry or Appearance.</p>
DEF	<p>[DEF ID #IMPLIED] DEF defines a unique ID name for this node, referencable by other nodes. Hint: descriptive DEF names improve clarity and help document a model.</p>
USE	<p>[USE IDREF #IMPLIED] USE means reuse an already DEF-ed node ID, ignoring <code>_all_</code> other attributes and children. Hint: USEing other geometry (instead of duplicating nodes) can improve performance. Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!</p>
height	<p>[height: accessType initializeOnly, type SFFloat CDATA "2"] Size in meters. Warning: simple-geometry dimensions cannot be changed after initial creation, use Transform scale instead.</p>
bottomRadius	<p>[bottomRadius: accessType initializeOnly, type SFFloat CDATA "1"] Size in meters. Warning: simple-geometry dimensions cannot be changed after initial creation, use Transform scale instead.</p>
side	<p>[side: accessType initializeOnly, type SFBool (true false) "true"] Whether to draw sides (other inside faces are not drawn). Warning: cannot be changed after initial creation.</p>
bottom	<p>[bottom: accessType initializeOnly, type SFBool (true false) "true"] Whether to draw bottom (other inside faces are not drawn). Warning: cannot be changed after initial creation.</p>
solid	<p>[solid: accessType initializeOnly, type SFBool (true false) "true"] Setting solid true means draw only one side of polygons (backface culling on), setting solid false means draw both sides of polygons (backface culling off). Warning: default value true can completely hide geometry if viewed from wrong side! Warning: solid false not supported in VRML97.</p>
containerField	<p>[containerField: NMTOKEN "geometry"] containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.</p>
class	<p>[class CDATA #IMPLIED] class is a space-separated list of classes, reserved for use by XML stylesheets. class attribute is only supported in XML encoding of X3D scenes.</p>

Cylinder node

Right-angle cylinder with top and bottom caps

Non-zero non-negative *height* above bottom

Circular *radius*

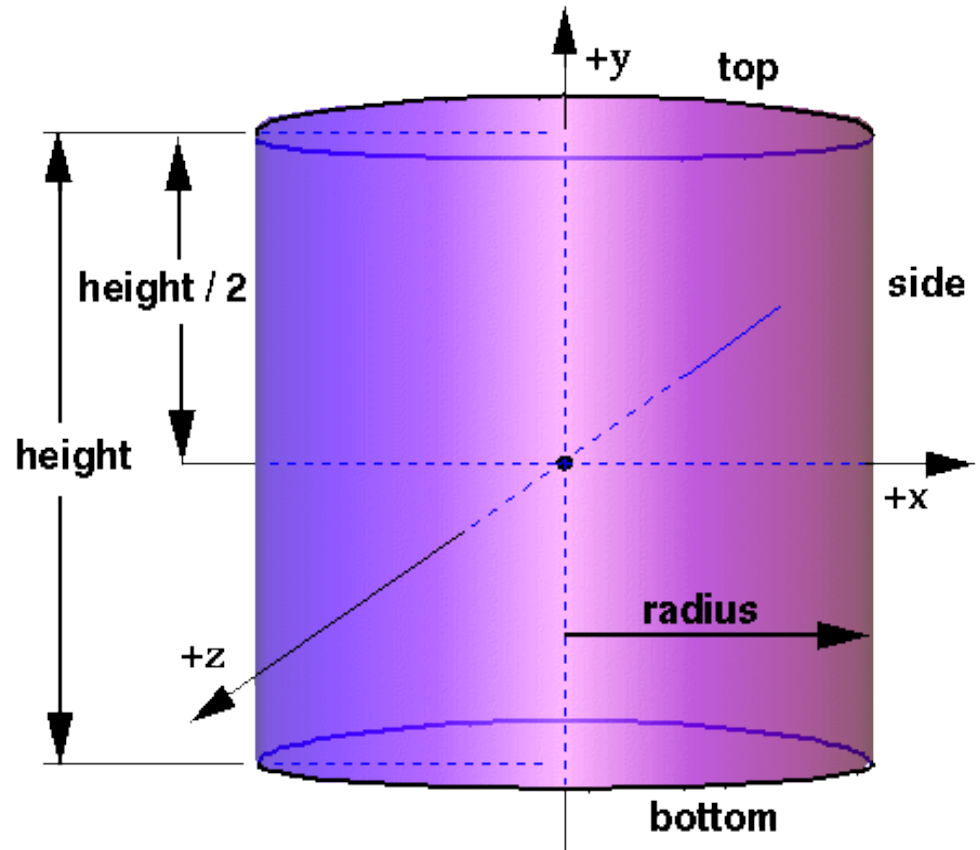
Centered at local origin

Can hide different parts

- *side*='false'
- *top*='false'
- *bottom*='false'

Default values are

height='2' *radius*='1'

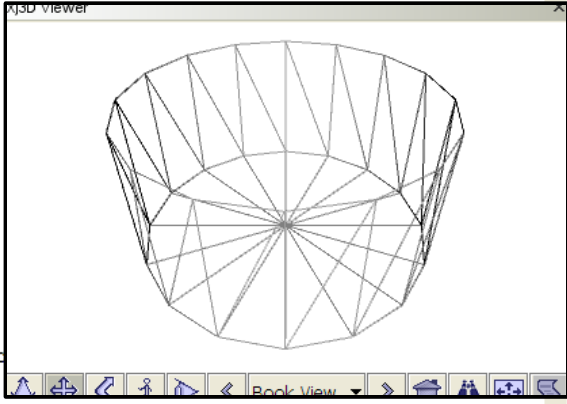


Cylinder.x3d - Editor

Cylinder.x3d x


```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.1//EN" "http://www.web3d.org/specifications/x3d-3.1.dtd">
<X3D profile='Immersive' version='3.1' xmlns:xsd='http://www.w3.org/2001/XMLSchema-instance'
xsd:noNamespaceSchemaLocation='http://www.web3d.org/specifications/x3d-3.1.xsd'>
  <head>
    <meta content='Cylinder.x3d' name='title' />
    <meta content='Cylinder geometric primitive node.' name='description' />
    <meta content='Leonard Daly and Don Brutzman' name='creator' />
    <meta content='1 January 2007' name='created' />
    <meta content='26 December 2007' name='modified' />
    <meta content='http://X3dGraphics.com' name='reference' />
    <meta content='http://www.web3d.org/x3d/content/examples/help.html' name='reference' />
    <meta content='Copyright Don Brutzman and Leonard Daly 2007' name='rights' />
    <meta content='X3D book, X3D graphics, X3D-Edit, http://www.x3dGraphics.com' name='subject' />
    <meta name='identifier'
      content='http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Cylinder.x3d' />
    <meta content='X3D-Edit, https://savage.nps.edu/X3D-Edit' name='generator' />
    <meta content='../license.html' name='license' />
  </head>
  <Scene>
    <Background skyColor='1 1 1' />
    <Viewpoint description='Book View' orientation='-1 0. 0 0.68' position='0 2.9 4.83' />
    <Shape>
      <Cylinder bottom='true' radius='2' side='true' solid='false' top='false' />
      <Appearance>
        <Material />
      </Appearance>
    </Shape>
  </Scene>
</X3D>
```

Xj3D viewer



24:81 | INS

Xj3D Viewer



Book View

Edit Cylinder

DEF []
USE []

containerField
 geometry

bottom
height 2
radius 2
side
solid
top

OK Cancel Help

Cylinder.x3d



Cylinder

Cylinder is a geometry node.

Hint: insert a Shape node before adding geometry or Appearance.

DEF
[DEF ID #IMPLIED]
 DEF defines a unique ID name for this node, referencable by other nodes.
Hint: descriptive DEF names improve clarity and help document a model.

USE
[USE IDREF #IMPLIED]
 USE means reuse an already DEF-ed node ID, ignoring `_all_` other attributes and children.
Hint: USEing other geometry (instead of duplicating nodes) can improve performance.
Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!

height
[height: accessType initializeOnly, type SFFloat CDATA "2"]
 Size in meters.
Warning: simple-geometry dimensions cannot be changed after initial creation, use Transform scale instead.

radius
[radius: accessType initializeOnly, type SFFloat CDATA "1"]
 Size in meters.
Warning: simple-geometry dimensions cannot be changed after initial creation, use Transform scale instead.

top
[top: accessType initializeOnly, type SFBool (true|false) "true"]
 Whether to draw top (inside faces are never drawn).
Warning: cannot be changed after initial creation.

side
[side: accessType initializeOnly, type SFBool (true|false) "true"]
 Whether to draw sides (inside faces are never drawn).
Warning: cannot be changed after initial creation.

bottom
[bottom: accessType initializeOnly, type SFBool (true|false) "true"]
 Whether to draw bottom (inside faces are never drawn).
Warning: cannot be changed after initial creation.

solid
[solid: accessType initializeOnly, type SFBool (true|false) "true"]
 Setting solid true means draw only one side of polygons (backface culling on), setting solid false means draw both sides of polygons (backface culling off).
Warning: default value true can completely hide geometry if viewed from wrong side!
Warning: solid false not supported in VRML97.

containerField
[containerField: NMTOKEN "geometry"]
 containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.

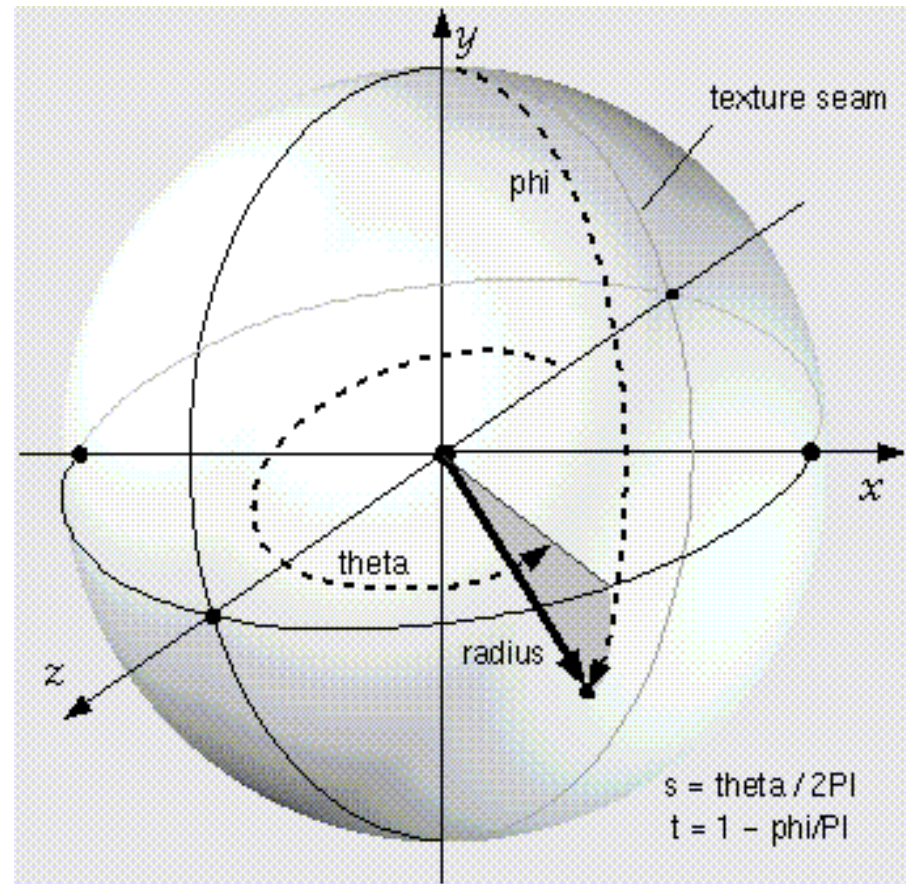
class
[class CDATA #IMPLIED]
 class is a space-separated list of classes, reserved for use by XML stylesheets. class attribute is only supported in XML encoding of X3D scenes.

Sphere node

Circular *radius*

Centered at local origin

- phi and theta are implicit
- not defined by author

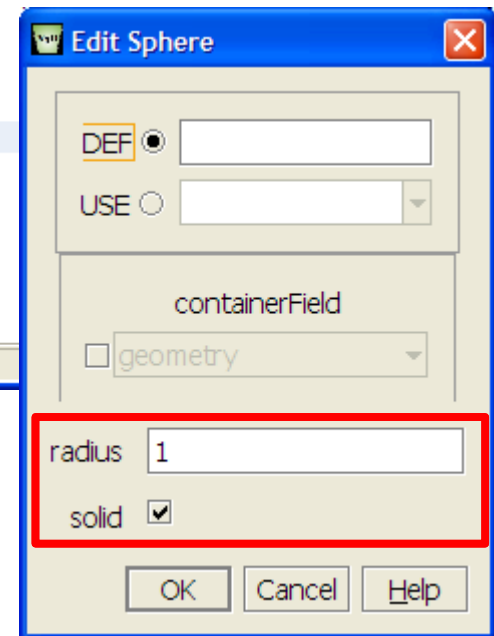
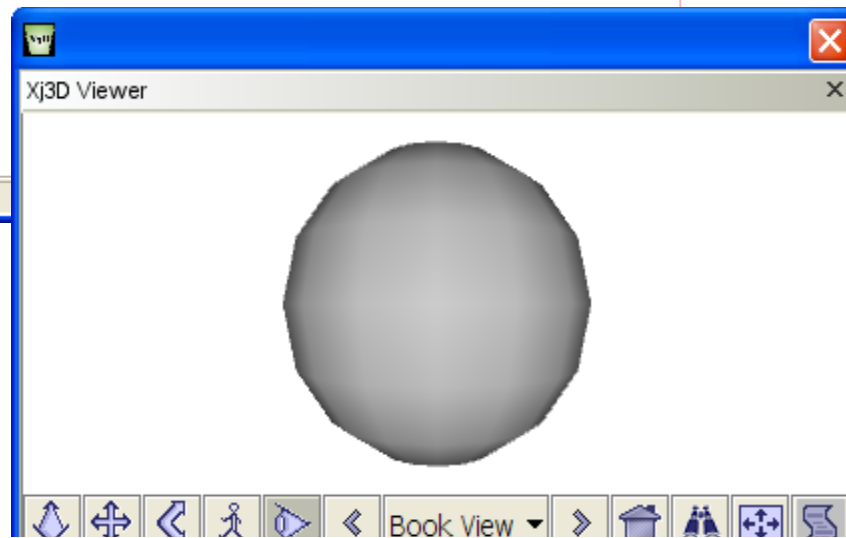
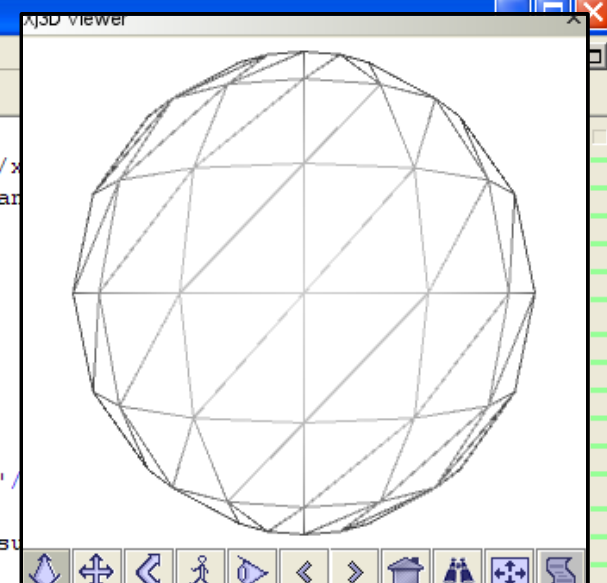



```

Sphere.x3d x
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.1//EN" "http://www.web3d.org/specifications/x
<X3D profile='Immersive' version='3.1' xmlns:xsd='http://www.w3.org/2001/XMLSchema-instan
  xsd:noNamespaceSchemaLocation='http://www.web3d.org/specifications/x3d-3.1.xsd'>
  <head>
    <meta content='Sphere.x3d' name='title' />
    <meta content='Sphere geometric primitive node.' name='description' />
    <meta content='Leonard Daly and Don Brutzman' name='creator' />
    <meta content='1 January 2007' name='created' />
    <meta content='23 March 2007' name='modified' />
    <meta content='http://X3dGraphics.com' name='reference' />
    <meta content='http://www.web3d.org/x3d/content/examples/help.html' name='reference' />
    <meta content='Copyright Don Brutzman and Leonard Daly 2007' name='rights' />
    <meta content='X3D book, X3D graphics, X3D-Edit, http://www.x3dGraphics.com' name='st
    <meta name='identifier'
      content='http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Sphere.x3d' />
    <meta content='X3D-Edit, https://savage.nps.edu/X3D-Edit' name='generator' />
    <meta content='../license.html' name='license' />
  </head>
  <Scene>
    <Background skyColor='1 1 1' />
    <Viewpoint description='Book View' orientation='0 0 1 0' position='0 0 3' />
    <Shape>
      <Sphere />
    </Shape>
  </Scene>
</X3D>

```

24:16 | INS



Sphere.x3d



Sphere

Sphere is a geometry node.

Hint: insert a Shape node before adding geometry or Appearance.

DEF

[DEF ID #IMPLIED]

DEF defines a unique ID name for this node, referencable by other nodes.

Hint: descriptive DEF names improve clarity and help document a model.

USE

[USE IDREF #IMPLIED]

USE means reuse an already DEF-ed node ID, ignoring `_all_` other attributes and children.

Hint: USEing other geometry (instead of duplicating nodes) can improve performance.

Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!

radius

[radius: accessType initializeOnly, type SFFloat CDATA "1"]

Size in meters.

Warning: simple-geometry dimensions cannot be changed after initial creation, use Transform scale instead.

solid

[solid: accessType initializeOnly, type SFBool (true|false) "true"]

Setting solid true means draw only one side of polygons (backface culling on), setting solid false means draw both sides of polygons (backface culling off).

Warning: default value true can completely hide geometry if viewed from wrong side!

Warning: solid false not supported in VRML97.

containerField

[containerField: NMTOKEN "geometry"]

containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.

class

[class CDATA #IMPLIED]

Text node

Produce readable flat, 2D text strings in X3D world
string field is MFString array of “quoted strings”

- Each “quoted string” appears on a separate line

length field is MFFloat array of lengths for each line

- Can shrink or stretch size of each line if needed

maxExtent is maximum length for all substring lines

Note characters have no 3D depth

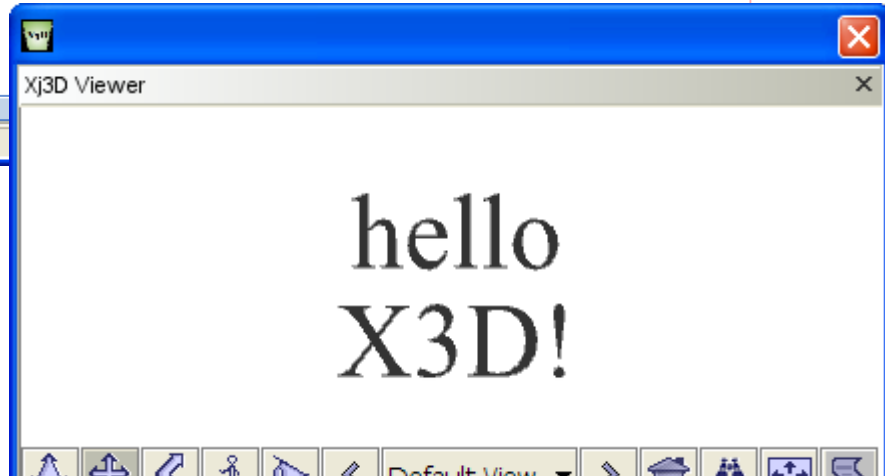
- Flat when viewed from alongside
- Typically viewable from behind since default is *solid*='false'
- **Hint:** use Billboard to face user




```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.1//EN" "http://www.web3d.org/specifications/x3d-3.1.dtd">
<X3D profile='Immersive' version='3.1' xmlns:xsd='http://www.w3.org/2001/XMLSchema-instance'
      xsd:noNamespaceSchemaLocation='http://www.web3d.org/specifications/x3d-3.1.xsd'>
  <head>
    <meta content='Text.x3d' name='title' />
    <meta content='Simple Text node: hello X3D!' name='description' />
    <meta content='Don Brutzman' name='creator' />
    <meta content='25 March 2005' name='created' />
    <meta content='29 December 2007' name='modified' />
    <meta content='Copyright (c) Don Brutzman and Len Daly, 2005' name='rights' />
    <meta name='identifier'
          content='http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitive
    <meta content='X3D-Edit, https://savage.nps.edu/X3D-Edit' name='generator' />
    <meta content='../license.html' name='license' />
  </head>
  <Scene>
    <Background skyColor='1 1 1' />
    <Viewpoint description='Default View' position='0 0 3' />
    <Viewpoint description='Book View' position='0.89 -1.11 2.33' />
    <Shape>
      <Text DEF='DefaultText' string='hello" "X3D!' />
      <FontStyle DEF='CenteredFontStyle' justify='MIDDLE' MIDDLE' />
    </Text>
    <Appearance>
      <Material DEF='DefaultMaterial' ambientIntensity='0.2' diffuseColor='.2 .2 .2' emissiveColor='0 0 0' shininess='0.2' specularColor='0
    </Appearance>
    </Shape>
  </Scene>
</X3D>

```



Edit Text

containerField DEF: DefaultText

geometry USE: [dropdown]

string array

individual SFString values

hello from "Monterey"

length [input field]

maxExtent 0

solid

OK Cancel Help

Text.x3d

Inserting apostrophes, ampersands, and quotation marks into Text strings

Character entity definitions are XML encodings

- Character entities are also known as *escape characters*
- apostrophe ' is ' & is & " is "
- <http://www.w3.org/TR/REC-html40/sgml/entities.html>

Precede embedded "quote marks" with backslash (\" to differentiate from line-delimiting quote marks

Suggested XML to escape Text node's string field:

- single quote (apostrophe) as XML attribute delimiter
- string=' "Hello from \"Monterey\"" ' or
- string=' "Hello from \""Monterey"" ' or
- string=' "A friend's new car" "just arrived" ' or

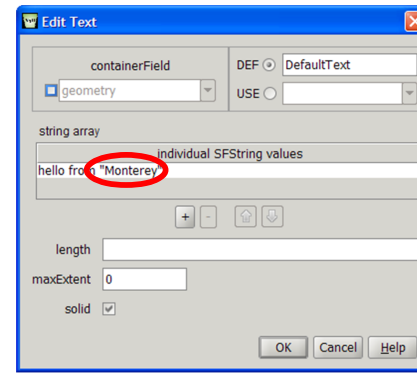
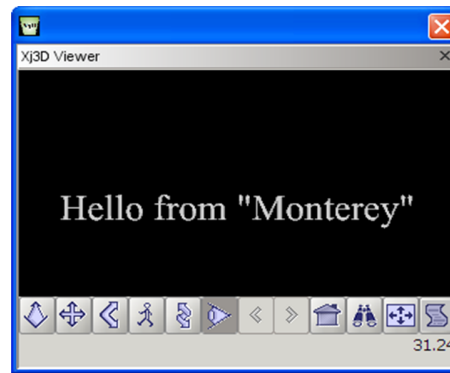
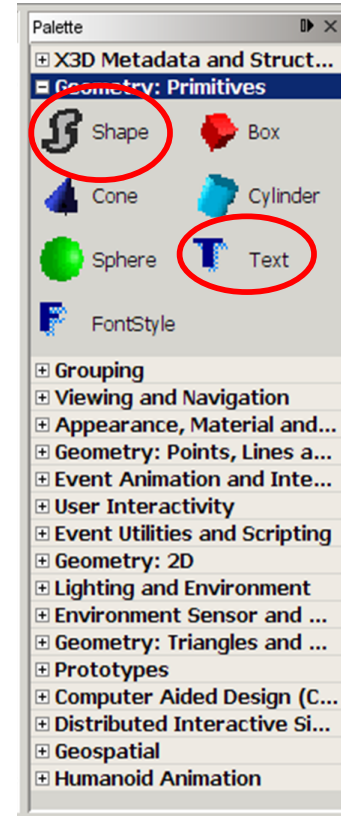
Try it yourself



1. Create a new scene by clicking the New X3D Scene button, or else select using menus (*File, New X3D, New X3D Scene*)
2. Open the palette for *Geometry: Primitives*
3. Drag a new Shape node into the scene graph where XML comment says
`<!-- Scene graph nodes are added here -->`
7. Drag a new Text node into the scene graph where the XML comment says
`<!-- Add a single geometry node here -->`
9. Right click on the Text node, cut and paste the text into the `string` field. Be sure to include all of the double quotes, don't paste the single quote delimiters.

```
<Text string='Hello from \"My Home Town\" ' />
```

11. Right-click the context menu to refresh or redraw in Xj3D:



```
TextSpecialCharacters.x3d - Editor
TextSpecialCharacters.x3d x
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.1//EN" "http://www.web3d.org/specifications/x3d-3.1.dtd">
<X3D profile='Immersive' version='3.1' xmlns:xsd='http://www.w3.org/2001/XMLSchema-instance' xsd:noNamespaceSchemaLocation='http://www.web3d.org/sp
<head>
  <meta content='TextSpecialCharacters.x3d' name='title' />
  <meta content='Text node demonstration of quotation, apostrophe, ampersand and backslash characters using X3D MFString escaping for XML charact
  <meta content='Don Brutzman' name='creator' />
  <meta content='12 July 2008' name='created' />
  <meta content='22 January 2008' name='modified' />
  <meta content='Character entity references in HTML 4' name='reference' />
  <meta content='http://www.w3.org/TR/REC-html40/sgml/entities.html' name='reference' />
  <meta content='Copyright (c) Don Brutzman and Len Daly, 2008' name='rights' />
  <meta content='http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/TextSpecialCharacters.x3d' name='identifier' />
  <meta content='X3D-Edit, https://savage.nps.edu/X3D-Edit' name='generator' />
  <meta content='../license.html' name='license' />
</head>
<Scene>
  <Background skyColor='1 1 1' />
  <Viewpoint description='Default View' position='0 0 15' />
  <Shape>
    <!-- Empty string "" means to skip a line -->
    <!-- The ampersand escape characters are based on XML rules -->
    <!-- apostrophe ' is &apos; and needs to be escaped in single-quote delimiters used for string='value' attribute -->
    <!-- ampersand & is &amp; and needs to be escaped -->
    <!-- quotation " is &quot; and isn't needed if single-quote delimiters used for string='value' attribute -->
    <!-- quotation " can be used as part of X3D string if escaped with backslash: \" -->
    <!-- backslash \ is used as escape character for " (and itself) in X3D -->
    <!-- character entities are listed in HTML specification and are good for any XML -->
    <Text DEF='DefaultText' string='Character entity substitutions:
      "empty string \"\" skips a line:"
      ""
      "apostrophe &apos; is &amp;apos;"
      "ampersand &amp; is &amp;amp;"
      "quote mark \" is &amp;quot;"
      "backslash \\ is X3D escape character"
      "Pi &#928; is &amp;#928; XML character entity">
      <FontStyle DEF='CenteredFontStyle' justify='MIDDLE' "MIDDLE" />
    </Text>
    <Appearance>
      <Material DEF='DefaultMaterial' diffuseColor='0.2 0.2 0.2' />
    </Appearance>
  </Shape>
</Scene>
</X3D>
```

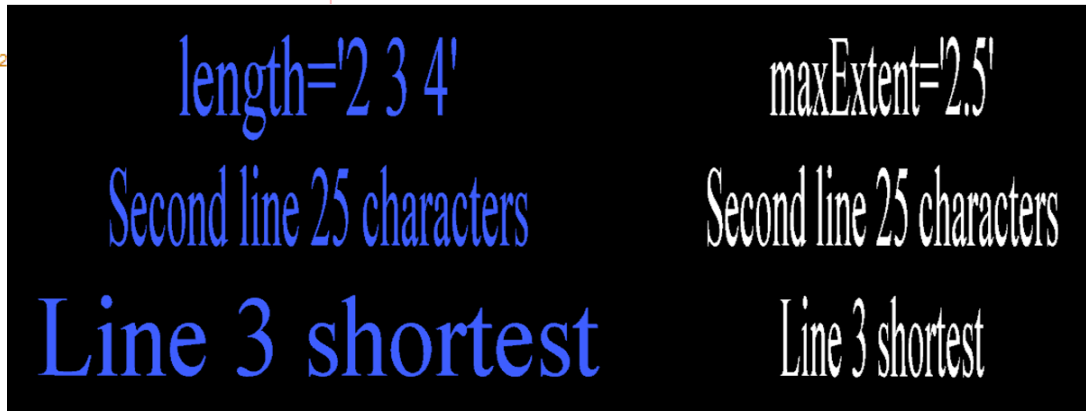
Character entity substitutions:
empty string "" skips a line:
apostrophe ' is '
ampersand & is &
quote mark " is "
backslash \ is X3D escape character
Pi Π is Π XML character entity



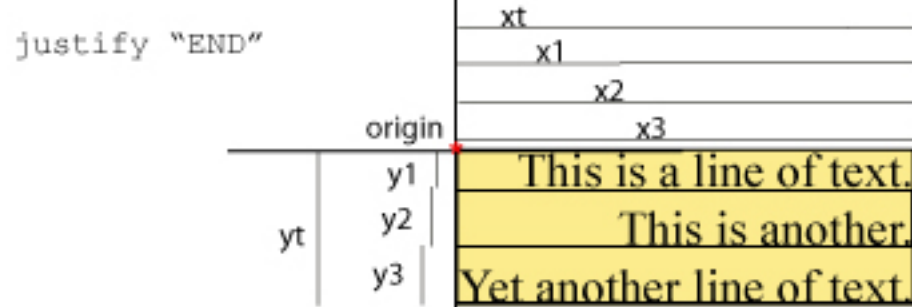
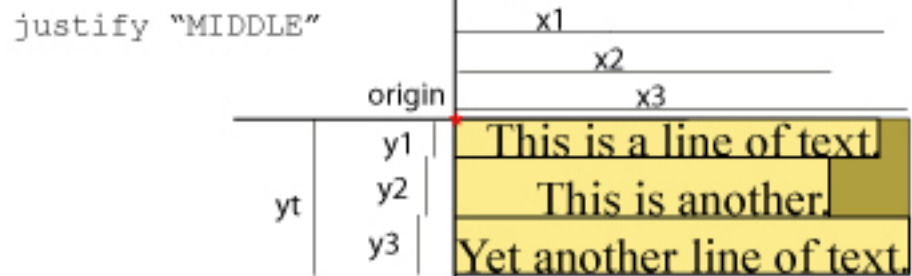
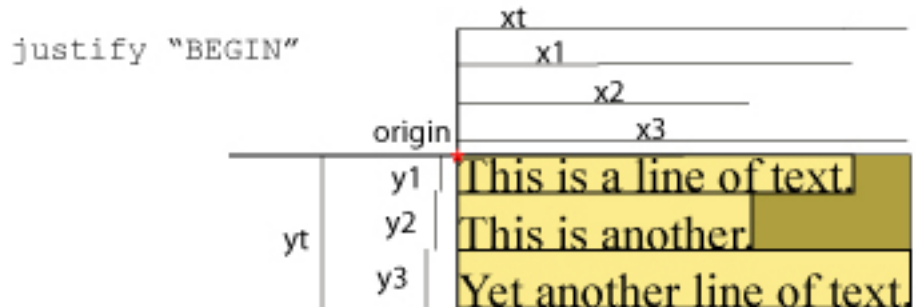
```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.1//EN" "http://www.web3d.org/specifications/x3d-3.1.dtd">
3 <X3D profile='Immersive' version='3.1' xmlns:xsd='http://www.w3.org/2001/XMLSchema-instance' xsd:noNamespaceSchemaLocation='http://www.web3d.org/s
4 <head>
5 <meta content='TextLengthMaxExtent.x3d' name='title' />
6 <meta content='Simple Text node, illustrating length array and maxExtent field' name='description' />
7 <meta content='Don Brutzman' name='creator' />
8 <meta content='25 January 2009' name='created' />
9 <meta content='25 January 2009' name='modified' />
10 <meta content='TextLengthMaxExtent.png' name='reference' />
11 <meta content='Copyright (c) Don Brutzman and Len Daly, 2005' name='rights' />
12 <meta content='http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/TextLengthMaxExtent.x3d' name='identifier' />
13 <meta content='X3D-Edit, https://savage.nps.edu/X3D-Edit' name='generator' />
14 <meta content='../license.html' name='license' />
15 </head>
16 <Scene>
17 <Viewpoint description='Default View' position='0 0 6' />
18 <Viewpoint description='Book View' position='0.89 -1.11 2.33' />
19 <Transform translation='-2 0 0'>
20 <Shape>
21 <Text DEF='DefaultText' length='2 3 4' string='"length=&apos;2 3 4&apos;" "Second line 25 characters" "Line 3 shortest"'>
22 <FontStyle DEF='CenteredFontStyle' justify="MIDDLE" MIDDLE" />
23 </Text>
24 <Appearance>
25 <Material DEF='BlueMaterial' diffuseColor='0.2
26 </Appearance>
27 </Shape>
28 </Transform>
29
30
31
32
33
34
35
36
37 <Transform translation='2 0 0'>
38 <Shape>
39 <Text maxExtent='2.5' string='"maxExtent=&apos;2.5&apos;" "Second line 25 characters" "Line 3 shortest"'>
40 <FontStyle USE='CenteredFontStyle' />
41 </Text>
42 </Shape>
43 </Transform>
44 </Scene>
45 </X3D>

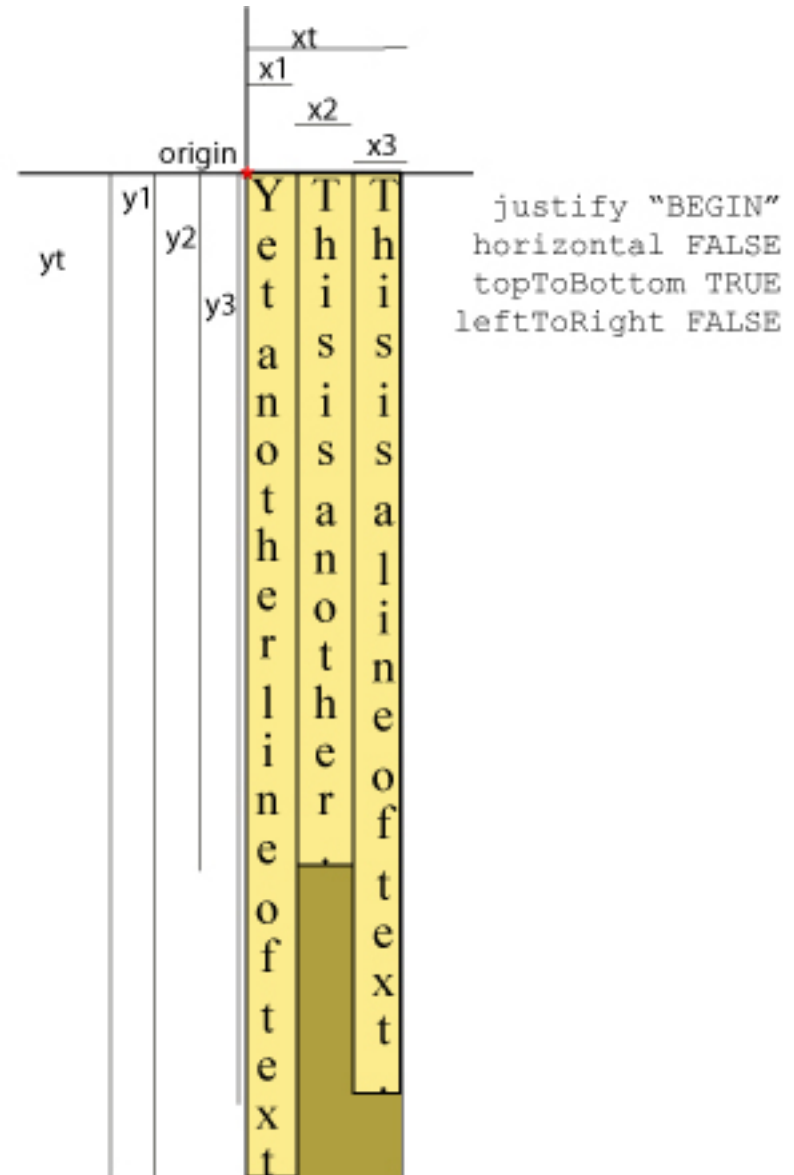
```




Horizontal or vertical adjustments



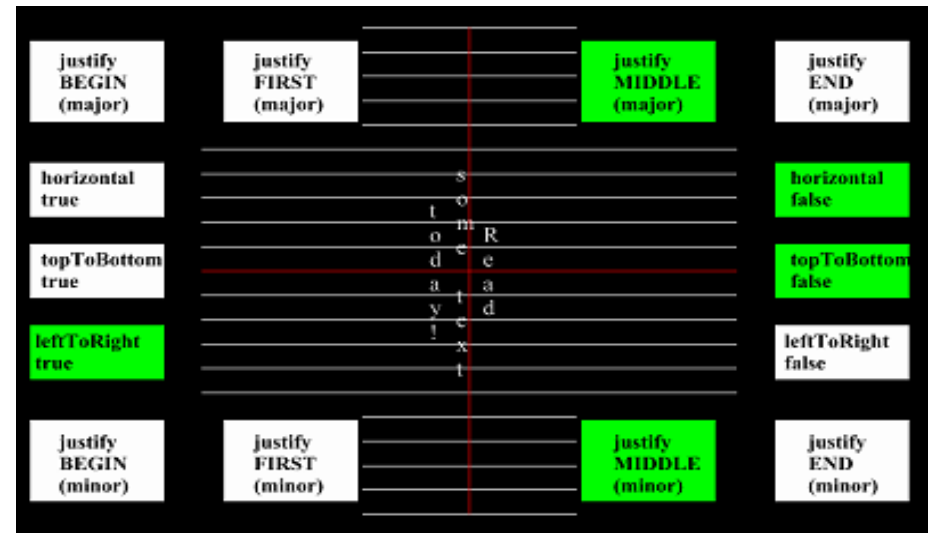
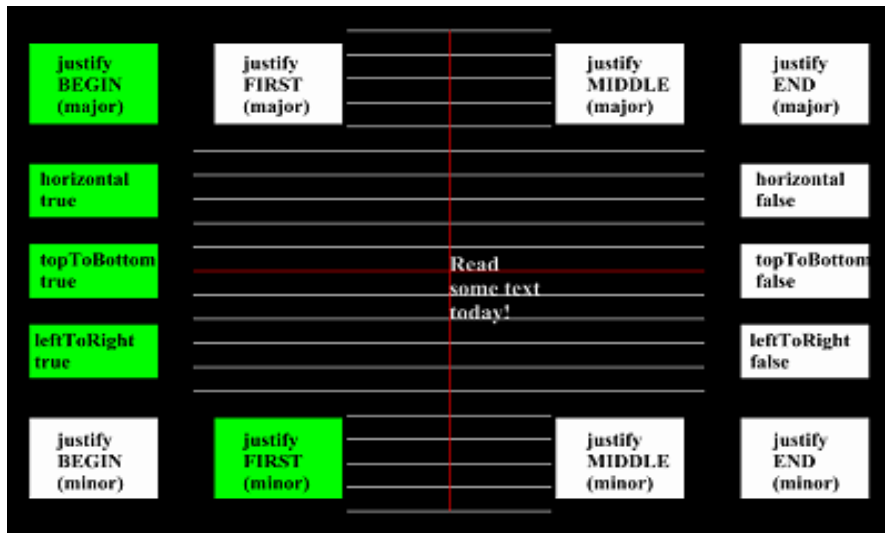
lineBounds [x1 y1, x2 y2, x3 y3]
 textBounds [xt yt]



 Text	Text is a geometry node that can contain a FontStyle node. Hint: insert a Shape node before adding geometry or Appearance. You can also substitute a type-matched ProtoInstance.
DEF	[DEF ID #IMPLIED] DEF defines a unique ID name for this node, referencable by other nodes. Hint: descriptive DEF names improve clarity and help document a model.
USE	[USE IDREF #IMPLIED] USE means reuse an already DEF-ed node ID, ignoring <code>_all_</code> other attributes and children. Hint: USEing other geometry (instead of duplicating nodes) can improve performance. Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!
string	[string: accessType inputOutput, type MFString CDATA #IMPLIED] Single or multiple string values to present as Text. Hint: Strings can have multiple values, so separate each string by quote marks Hint: Strings can contain quote marks by first escaping them with a backslash example: "say \"hello\" please" Hint: many XML tools substitute XML character references automatically if needed (like <code>&#38;</code> for <code>&</code> or <code>&#34;</code> for <code>"</code>).
length	[length: accessType inputOutput, type MFFloat CDATA #IMPLIED] Array of length values for each text string in the local coordinate system. Each string is stretched or compressed to fit.
maxExtent	[maxExtent: accessType inputOutput, type SFFloat CDATA "0.0"] Limits/compresses all text strings if max string length is longer than maxExtent, as measured in local coordinate system.
solid	[solid: accessType initializeOnly, type SFBool (true false) "true"] Setting solid true means draw only one side of polygons (backface culling on), setting solid false means draw both sides of polygons (backface culling off). Warning: default value true can completely hide geometry if viewed from wrong side! Warning: solid false not supported in VRML97.
lineBounds	[lineBounds: accessType outputOnly, type MFVec2f CDATA #IMPLIED] Array of 2D bounding box values for each line of text in the local coordinate system.
textBounds	[textBounds: accessType outputOnly, type SFVec2f CDATA #IMPLIED] 2D bounding box value for all lines of text in the local coordinate system.
containerField	[containerField: NMTOKEN "geometry"] containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.
class	[class CDATA #IMPLIED] class is a space-separated list of classes, reserved for use by XML stylesheets. class attribute is only supported in XML encoding of X3D scenes.

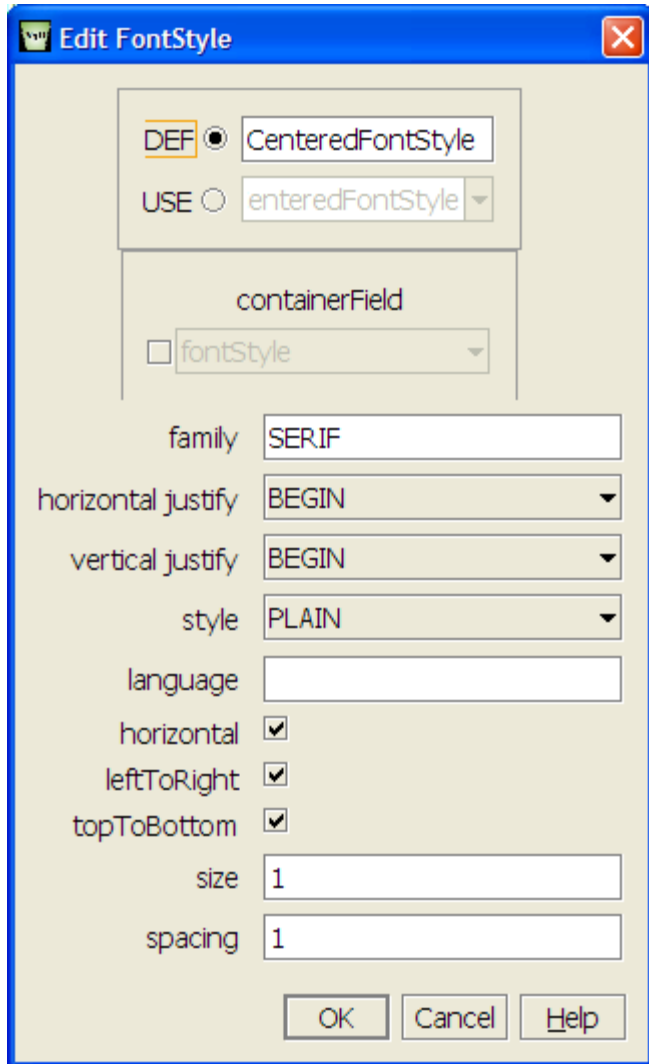
FontStyle node

Defines *size*, font *family*, layout directions and justification, language, and style for Text strings



<http://www.web3d.org/x3d/content/examples/ConformanceNist/Appearance/FontStyle/driver.x3d>


```
<FontStyle justify=' "MIDDLE" "MIDDLE" ' />
```



FontStyle is only allowed as child of a Text node

- FontStyle modifies that parent

Other supported default font family values are SANS (serif) and TYPEWRITER

- Additional font families require special browser support

Other field values support internationalization (I18N) and localization (L10N)

- DEF, USE for consistent look

FontStyle values, X3D Specification

Table 15.2 – Major Alignment, *horizontal* = TRUE

<i>justify</i> Enumerant	<i>leftToRight</i> = TRUE	<i>leftToRight</i> = FALSE
FIRST	Left edge of each line	Right edge of each line
BEGIN	Left edge of each line	Right edge of each line
MIDDLE	Centred about X-axis	Centred about X-axis
END	Right edge of each line	Left edge of each line

Table 15.3 – Major Alignment, *horizontal* = FALSE

<i>justify</i> Enumerant	<i>topToBottom</i> = TRUE	<i>topToBottom</i> = FALSE
FIRST	Top edge of each line	Bottom edge of each line
BEGIN	Top edge of each line	Bottom edge of each line
MIDDLE	Centred about Y-axis	Centre about Y-axis
END	Bottom edge of each line	Top edge of each line

FontStyle values, X3D Specification

Table 15.4 – Minor Alignment, *horizontal* = TRUE

<i>justify</i> Enumerant	<i>topToBottom</i> = TRUE	<i>topToBottom</i> = FALSE
FIRST	Baseline of first line	Baseline of first line
BEGIN	Top edge of first line	Bottom edge of first line
MIDDLE	Centred about Y-axis	Centred about Y-axis
END	Bottom edge of last line	Top edge of last line

Table 15.5 – Minor Alignment, *horizontal* = FALSE

<i>justify</i> Enumerant	<i>leftToRight</i> = TRUE	<i>leftToRight</i> = FALSE
FIRST	Left edge of first line	Right edge of first line
BEGIN	Left edge of first line	Right edge of first line
MIDDLE	Centred about X-axis	Centred about X-axis
END	Right edge of last line	Left edge of last line

X3D Specification

Tables 15.6 and 15.7

Key			
	minor = "FIRST"		minor = "BEGIN"
	minor = "MIDDLE"		minor = "END"

		major = "BEGIN" or "FIRST"		major = "MIDDLE"		major = "END"	
		leftToRight		leftToRight		leftToRight	
		TRUE	FALSE	TRUE	FALSE	TRUE	FALSE
topToBottom	TRUE						
	FALSE						

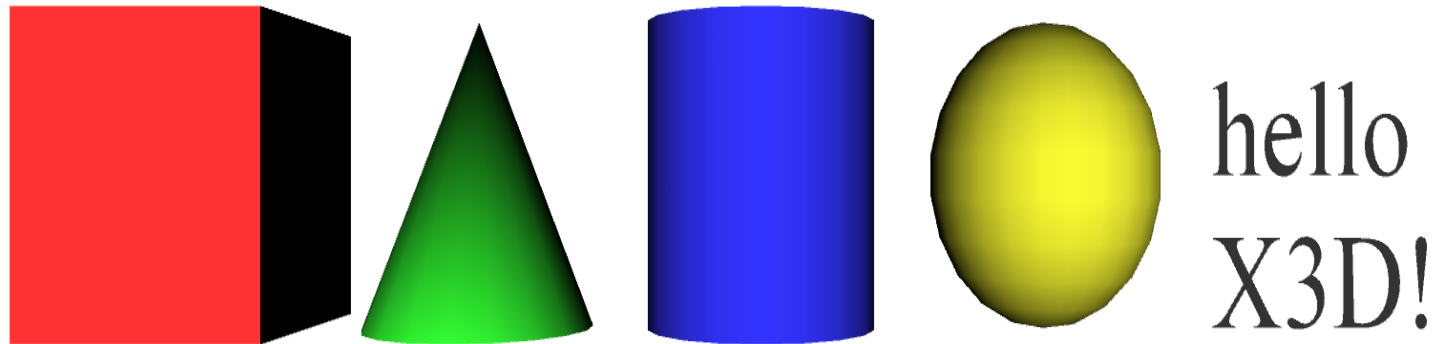
		major = "BEGIN" or "FIRST"		major = "MIDDLE"		major = "END"	
		leftToRight		leftToRight		leftToRight	
		TRUE	FALSE	TRUE	FALSE	TRUE	FALSE
topToBottom	TRUE						
	FALSE						

Note: In every case, the "FIRST" minor axis marker is coincident with the "BEGIN" minor axis marker (and is offset for presentation purposes only).

X3D Tooltips - Mozilla Firefox	
File Edit View History Bookmarks Tools GUtil Help	
http://www.web3d.org/x3d/content/X3dTooltips.html#FontStyle	
top help credits	
F FontStyle	FontStyle defines the size, family, and style used for Text nodes. Hint: first add a Text node as parent.
DEF	[DEF ID #IMPLIED] DEF defines a unique ID name for this node, referencable by other nodes. Hint: descriptive DEF names improve clarity and help document a model.
USE	[USE IDREF #IMPLIED] USE means reuse an already DEF-ed node ID, ignoring <code>_all_</code> other attributes and children. Hint: USEing other geometry (instead of duplicating nodes) can improve performance. Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!
family	[family: accessType initializeOnly, type MFString CDATA "SERIF"] Sequence of font family names in preference order - browsers use first supported family. Supported values include "SERIF" "SANS" "TYPEWRITER". Hint: SERIF and SANS are variable-width fonts (for example, Roman and Arial). Hint: TYPEWRITER is a fixed-width font (for example, Courier). Hint: Strings can have multiple values, so separate "each string" "by" "quote marks".
style	[style: accessType initializeOnly, type SFString CDATA (PLAIN BOLD ITALIC BOLDITALIC) "PLAIN"] Pick one of four values for text style.
justify	[justify: accessType initializeOnly, type MFString CDATA "BEGIN"] Two string values are provided for major and minor axis alignment, possible values are "FIRST" "BEGIN" "MIDDLE" "END"] Example: "MIDDLE" "MIDDLE". Hint: Strings can have multiple values, so separate "each string" "by" "quote marks".
size	[size: accessType initializeOnly, type SFFloat CDATA "1.0"] Nominal height (in local coordinate system) of text glyphs Also sets default spacing between adjacent lines of text.
spacing	[spacing: accessType initializeOnly, type SFFloat CDATA "1.0"] Adjustment factor for line spacing between adjacent lines of text.
language	[language: accessType initializeOnly, type SFString CDATA #IMPLIED] Language codes consist of a primary code and a (possibly empty) series of subcodes. [language-code = primary-code ("-" subcode)*] Two-letter primary codes are reserved for language abbreviations. [RFC1766, http://www.ietf.org/rfc/rfc1766.txt] Two-letter primary codes include en (English), fr (French), de (German), it (Italian), nl (Dutch), el (Greek), es (Spanish), pt (Portuguese), ar (Arabic), he (Hebrew), ru (Russian), zh (Chinese), ja (Japanese), hi (Hindi), ur (Urdu), and sa (Sanskrit). Any two-letter subcode is understood to be a country code. [ISO3166 or http://www.oasis-open.org/cover/iso639a.html]
horizontal	[horizontal: accessType initializeOnly, type SFBool (true false) "true"] Whether text direction is horizontal (true) or vertical (false).
leftToRight	[leftToRight: accessType initializeOnly, type SFBool (true false) "true"] Whether text direction is left-to-right (true) or right-to-left (false).
topToBottom	[topToBottom: accessType initializeOnly, type SFBool (true false) "true"] Whether text direction is top-to-bottom (true) or bottom-to-top (false).
containerField	[containerField: NMTOKEN "fontStyle"] containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.
class	[class CDATA #IMPLIED] class is a space-separated list of classes, reserved for use by XML stylesheets. class attribute is only supported in XML encoding of X3D scenes.

Review

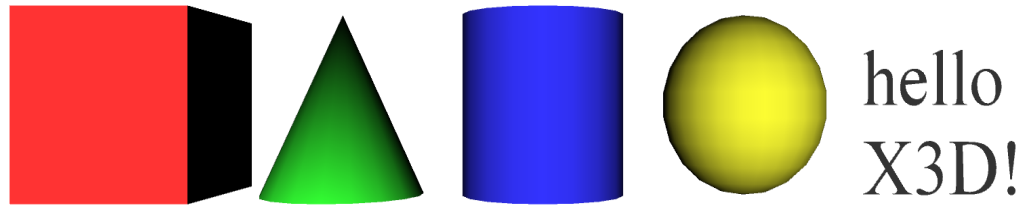
Geometry Primitives



Primitives are simple geometric constructs
Shape, geometry, Appearance, Material pattern
Browsers decide implementation details,
including quality of tessellation resolution

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "ISO//Web3D//DTD X3D 3.0//EN" "http://www.web3d.org/specifications/x3d-3.0.dtd">
X3D: version: 3.0, profile: Immersive, xmlns:xsd: http://www.w3.org/2001/XMLSchema-instance, xsd:noNamespaceSchemaLocation: http://www.web3d.org/specifications/x3d-3.0.xsd
head
  meta: name: filename, content: GeometryPrimitiveNodes.x3d
  meta: name: description, content: Geometry Primitive Nodes: Shape, Box, Cone, Cylinder, Sphere, Text, FontStyle
  meta: name: creator, content: Don Brutzman
  meta: name: created, content: 25 March 2005
  meta: name: modified, content: 25 March 2005
  meta: name: rights, content: Copyright (c) Don Brutzman and Len Daly, 2005
  meta: name: identifier, content: GeometryPrimitiveNodes.x3d
  meta: name: generator, content: X3D-Edit, http://www.web3d.org/x3d/conte
  meta: name: license, content: ../../license.html
Scene
  Transform: translation: -5 0 0
    Shape: DEF: DefaultShape, bboxCenter: 0 0 0, bboxSize: -1 -1 -1, containerField: children
      Box: DEF: DefaultBox, size: 2 2 2, containerField: geometry
        Appearance: DEF: DefaultAppearance, containerField: appearance
          Material: diffuseColor: 1 0.2 0.2
      Cone: DEF: DefaultCone, height: 2, bottomRadius: 1, side: true, bottom: true, containerField: geometry
        Appearance
          Material: diffuseColor: 0.2 1 0.2
      Cylinder: DEF: DefaultCylinder, height: 2, radius: 1, top: true, side: true, bottom: true, containerField: geometry
        Appearance
          Material: diffuseColor: 0.2 0.2 1
      Sphere: DEF: DefaultSphere, radius: 1, containerField: geometry
        Appearance
          Material: diffuseColor: 1 1 0.2
      Transform: translation: 4 0 0
        Shape
          Text: DEF: DefaultText, string: "hello" "X3D!", maxExtent: 0.0, containerField: geometry
            FontStyle: DEF: DefaultFontStyle, family: "SERIF", style: PLAIN, justify: "BEGIN", size: 1.0, spacing: 1.0, horizontal: true, leftToRight: true, topToBottom: true, containerField: fontStyle
          Appearance
            Material: DEF: DefaultMaterial, diffuseColor: 0.8 0.8 0.8, emissiveColor: 0 0 0, specularColor: 0 0 0, shininess: 0.2, ambientIntensity: 0.2, transparency: 0, containerField: material
  
```



Transform nodes
position each Shape
so that they do not
obscure each other

Additional Resources

Multilingual X3D tooltips

X3D tooltips are available online and bundled in the X3D-Edit help pages

X3D tooltips are available in the following languages

- English
- German
- Portuguese
- Chinese
- Italian
- Spanish
- French
- Korean

Translations into other languages are welcome

- <http://www.web3d.org/x3d/content/X3dTooltips.html>

X3D spec excerpt for Shape node

```
Shape : X3DShapeNode {
  SFNode [in,out] appearance NULL [X3DAppearanceNode]
  SFNode [in,out] geometry NULL [X3DGeometryNode]
  SFNode [in,out] metadata NULL [X3DMetadataObject]
  SFVec3f [] bboxCenter 0 0 0 (-∞,∞)
  SFVec3f [] bboxSize -1 -1 -1 [0,∞) or -1 -1 -1
}
```

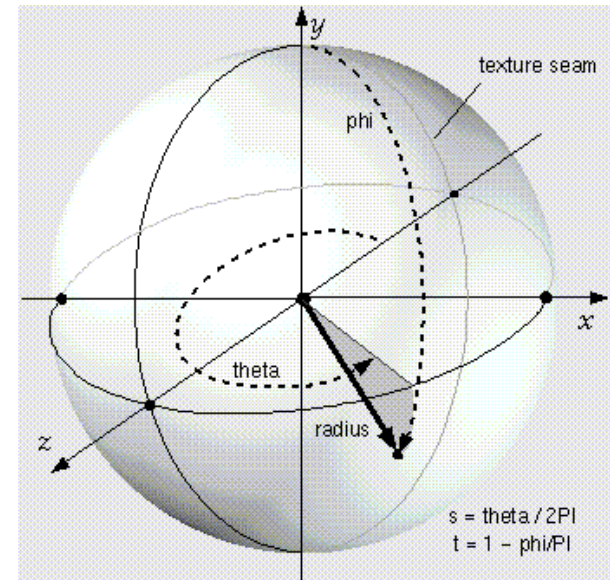
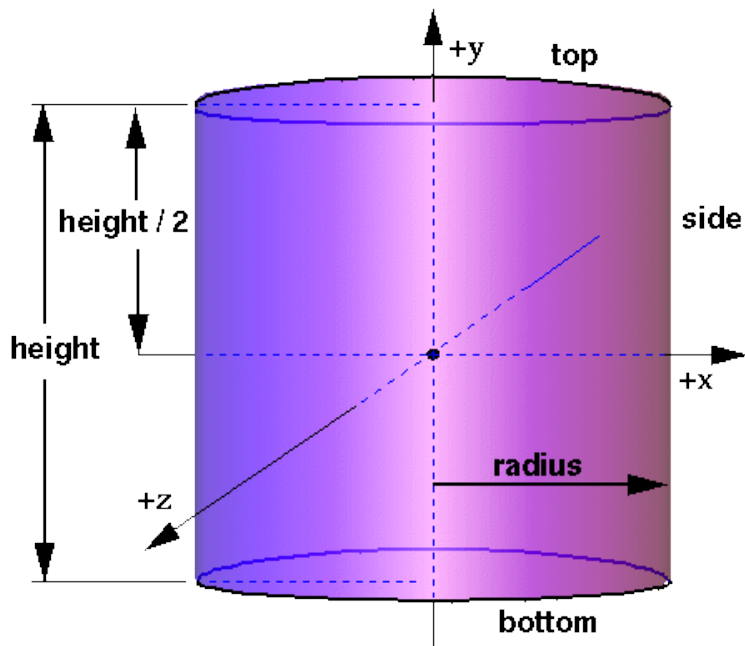
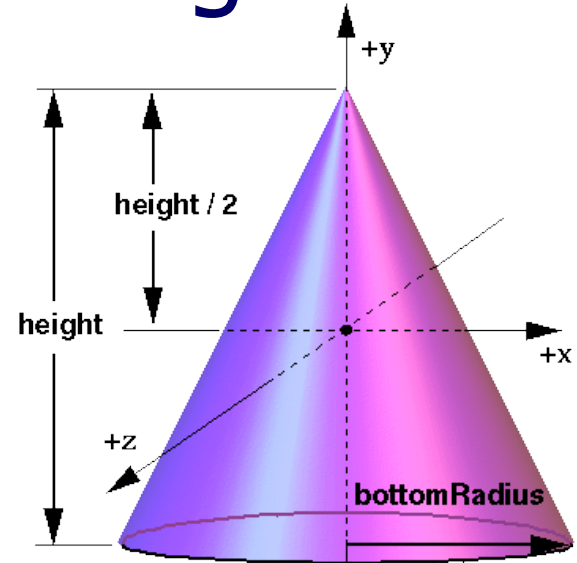
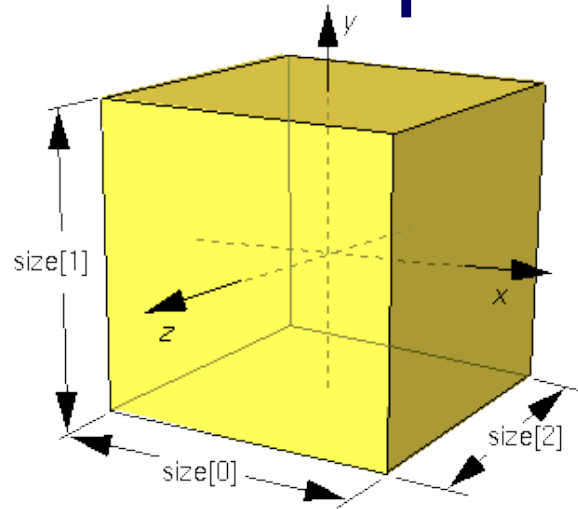
The Shape node has two fields, *appearance* and *geometry*, which are used to create rendered objects in the world. The *appearance* field contains an Appearance node that specifies the visual attributes (e.g., material and texture) to be applied to the geometry. The *geometry* field contains a geometry node. The specified geometry node is rendered with the specified appearance nodes applied. See [12.2 Concepts](#) for more information.

[17 Lighting component](#) contains details of the X3D lighting model and the interaction between Appearance nodes and geometry nodes.

If the *geometry* field is `NULL`, the object is not drawn.

The *bboxCenter* and *bboxSize* fields specify a bounding box that encloses the Shape node's geometry. This is a hint that may be used for optimization purposes. The results are undefined if the specified bounding box is smaller than the actual bounding box of the geometry at any time. A default *bboxSize* value, (-1, -1, -1), implies that the bounding box is not specified and, if needed, is calculated by the browser. A description of

X3D Specification Diagrams



Related concepts

DEF/USE nodes: chapter 3

Transform node: chapter 3

Viewpoint node: chapter 4

Appearance node: chapter 5

Material node: chapter 5

Background node: chapter 11

Bounding boxes: chapter 12

Geometry nodes

Chapter 2, Primitives

- Box, Cone, Cylinder, Sphere, Text / FontStyle

Chapter 6, Points Lines and Polygons

- PointSet, IndexedLineSet, IndexedFaceSet, ElevationGrid, Extrusion

Chapter 10, Geometry2D

- Arc2D, ArcClose2D, Circle2D, Disk2D, Polyline2D, Polypoint2D, Rectangle2D, TriangleSet2D

Chapter 13, Triangles and Quadrilaterals

- TriangleSet, TriangleStripSet, TriangleFanSet, QuadSet
- Both regular and Indexed versions

Advanced geometry nodes

Geospatial component

- GeoElevationGrid

NURBS component

- NurbsCurve, NurbsPatchSurface, NurbsSweptSurface, NurbsSwungSurface, NurbsTrimmedSurface

Programmable shaders component

- ComposedShader, PackagedShader, ProgramShader

Further information available in X3D Specification

- <http://www.web3d.org/x3d/specifications>

Chapter Summary

Chapter Summary

Shape is a container node for a single piece of geometry

Geometry primitive nodes

- Box, Cone, Cylinder, Sphere, Text
- FontStyle supports Text

Numerous additional resources

- X3D tooltips, with multilingual versions available
- X3D specifications
- Each are integrated within X3D-Edit authoring tool

Suggested exercises

Modify an existing example scene to make it into another object. Become familiar with editing, XML validation, and reloading a scene to refresh the 3D rendering.

Create a simple object using only primitive geometric shapes

Demonstrate use of internationalization (I18N) text

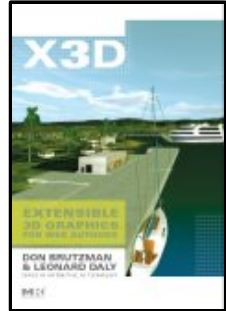
Look at other examples for modeling ideas

- <http://www.web3d.org/x3d/content/examples/Vrml2.0Sourcebook>
- <http://www.web3d.org/x3d/content/examples/Basic/StudentProjects>

References

References 1

X3D: Extensible 3D Graphics for Web Authors
by Don Brutzman and Leonard Daly, Morgan
Kaufmann Publishers, April 2007, 468 pages.



- Chapter 2, Geometry: Primitive Shapes
- <http://x3dGraphics.com>
- <http://x3dgraphics.com/examples/X3dForWebAuthors>

X3D Resources

- <http://www.web3d.org/x3d/content/examples/X3dResources.html>

References 2

X3D-Edit Authoring Tool

- <https://savage.nps.edu/X3D-Edit>

X3D Scene Authoring Hints

- <http://x3dgraphics.com/examples/X3dSceneAuthoringHints.ntml>



X3D Graphics Specification

- <http://www.web3d.org/x3d/specifications>
- http://www.web3d.org/x3d/specifications/spec_feedback
- Available as help pages from within X3D-Edit



References 3

VRML 2.0 Sourcebook by Andrea L. Ames, David R. Nadeau, and John L. Moreland, John Wiley & Sons, 1996.

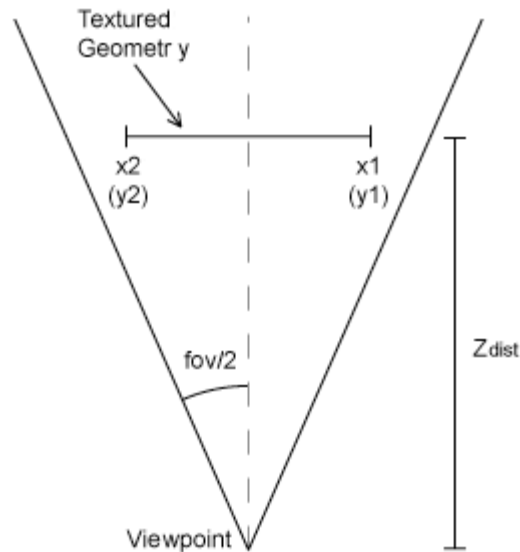


- <http://www.wiley.com/legacy/compbooks/vrml2sbk/cover/cover.htm>
- <http://www.web3d.org/x3d/content/examples/Vrml2.0Sourcebook>
- Chapter 02 – Introduction
- Chapter 03 – Shapes
- Chapter 04 – Text

References 4

Pixel Perfect Text by David Frerichs

- Overcome poor pixelation of Text nodes by creating a texture image of the desired text, along with a matching Viewpoint at the right distance
- http://www.frerichs.net/vrml2/pp/pixel_perfect.html



Contact

Don Brutzman

brutzman@nps.edu

http://faculty.nps.edu/brutzman

Code USW/Br, Naval Postgraduate School

Monterey California 93943-5000 USA

1.831.656.2149 voice

CGEMS, SIGGRAPH, Eurographics

The Computer Graphics Educational Materials Source(CGEMS) site is designed for educators

- to provide a source of refereed high-quality content
- as a service to the Computer Graphics community
- freely available, directly prepared for classroom use
- <http://cgems.inesc.pt>

X3D for Web Authors recognized by CGEMS! 😊

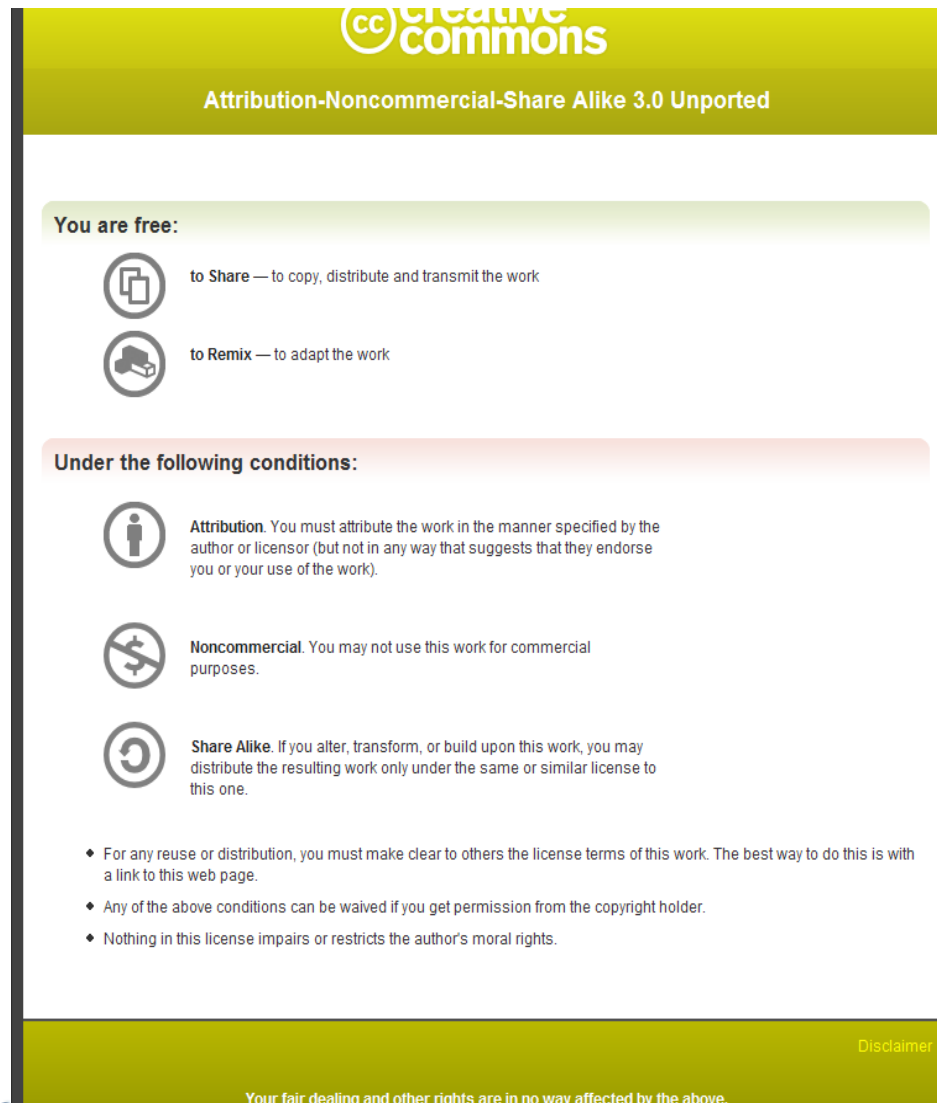
- Book materials: X3D-Edit tool, examples, slidesets
- Received jury award for Best Submission 2008

CGEMS supported by SIGGRAPH, Eurographics



Creative Commons open-source license

<http://creativecommons.org/licenses/by-nc-sa/3.0>





The image shows a Creative Commons license card for Attribution-Noncommercial-Share Alike 3.0 Unported. The card has a green header with the Creative Commons logo and the license name. Below the header, there are two main sections: 'You are free:' and 'Under the following conditions:'. The 'You are free:' section includes icons for 'Share' (two overlapping squares) and 'Remix' (a hand holding a pencil), with corresponding text explaining these permissions. The 'Under the following conditions:' section includes icons for 'Attribution' (a person), 'Noncommercial' (a dollar sign with a slash), and 'Share Alike' (a circular arrow), with corresponding text explaining these conditions. At the bottom, there are three bullet points providing additional information about reuse, waiver, and moral rights. A green footer contains the text 'Your fair dealing and other rights are in no way affected by the above.' and a 'Disclaimer' link.




cc creative commons

Attribution-Noncommercial-Share Alike 3.0 Unported

You are free:

-  to **Share** — to copy, distribute and transmit the work
-  to **Remix** — to adapt the work

Under the following conditions:

-  **Attribution.** You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
-  **Noncommercial.** You may not use this work for commercial purposes.
-  **Share Alike.** If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

- ♦ For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.
- ♦ Any of the above conditions can be waived if you get permission from the copyright holder.
- ♦ Nothing in this license impairs or restricts the author's moral rights.

Disclaimer

Your fair dealing and other rights are in no way affected by the above.

Open-source license for X3D-Edit software and X3D example scenes

<http://www.web3d.org/x3d/content/examples/license.html>

Copyright (c) 1995-2013 held by the author(s). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the names of the Naval Postgraduate School (NPS) Modeling Virtual Environments and Simulation (MOVES) Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

X3D Graphics for Web Authors

Chapter 2

Geometry 1: Primitive Shapes

Dorothy in Oz: "Toto, I've a feeling we're not in Kansas anymore."

—L. Frank Baum, Wizard of Oz, 1939



Memorable quotes for *The Wizard of Oz*: <http://www.imdb.com/title/tt0032138/quotes>

Table of Contents

Chapter Overview and Concepts

X3D Nodes and Examples

Additional Resources

Chapter Summary and Suggested Exercises

References



Chapter Overview



Overview: Geometry 1, Primitive Shapes

Common pattern for Shape nodes

- Shape contains geometry node
- Appearance and Material nodes

Five nodes for primitive geometry in this chapter

- Box, Cone, Cylinder, Sphere, Text
- Text node is flat, not extruded
- FontStyle modifies Text node parameters

X3D tooltips and specifications are helpful to use

[back to Table of Contents](#)

Concepts



Shape and geometry

Shape nodes can contain a single geometry node

- For example, one of the five geometry primitive nodes
- Alternatively contains a more-advanced geometry node
 - Chapter 2: Geometric primitives
 - Chapter 6: Points Lines and Polygon nodes
 - Chapter 10: Geometry2D nodes
 - Chapter 13: Triangle nodes

Shape nodes can also contain an Appearance node

- Which in turn contains a Material node for coloring
- Covered in Chapter 3



Since every individual piece of geometry to be drawn must have a parent Shape node, expect to see a lot of Shape nodes in your X3D scenes.

The structure provided by having many Shape nodes helps keep a scene organized and clearly separates capabilities that might otherwise get unintentionally mixed up.

Why is this pattern fundamental?

- Common design pattern throughout X3D:
 - **Shape**
 - **GeometryNode**
 - **Appearance**
 - **Material** (optional) for colors
 - **ImageTexture** (optional) for wrapping an image file
- Top three priorities in graphics design:
performance performance performance!!!
- This pattern is repeated in order to directly represent geometry and appearance together for maximum graphics-card performance

It is helpful to remember that maximizing the performance of graphics cards to render geometry is of fundamental importance. Performance issues explain many default values and design choices in X3D as well as other graphics languages.

For example, an excellent way to optimize for high performance is to do one thing extremely well, and apply it generally. As a result, graphics cards have long been optimized to render loooong piles of triangles as quickly as possible. Thus X3D geometric shapes are typically converted into triangles by the X3D browser when it takes your model and preprocesses it for the the graphics card. This general approach is called *polygon-based rendering*.

Old piano-teacher joke follows. [Carnegie Hall](#) is in New York City, and performing there is considered the pinnacle of a musical career.

- Pedestrian: how can I get to Carnegie Hall?
- Cab driver: well, there are only three ways to get there.
- Pedestrian: OK then, what are the three ways to get to Carnegie Hall?
- Cab driver: **practice, practice, practice!**

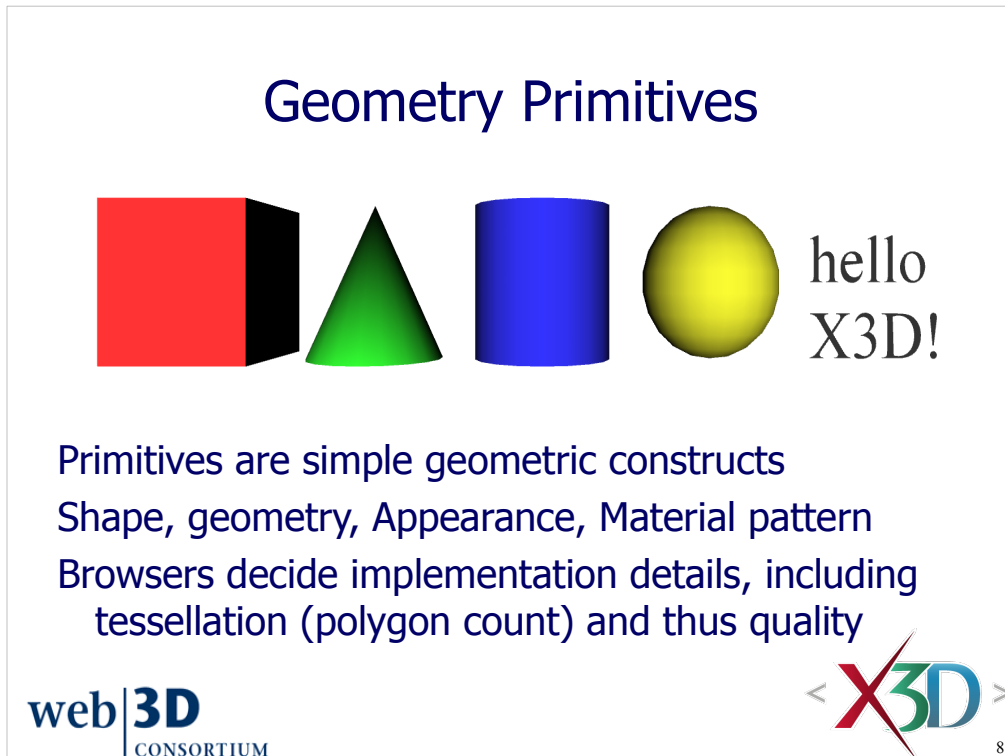


Figure 2.2b, page 39, *X3D for Web Authors*

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/GeometryPrimitiveNodes.x3d>

There are five primitive geometry nodes: Box Cone Cylinder Sphere and Text

Why is this very simple pattern repeated so many times? Because graphics cards like it!

Improving the polygon count of primitive geometry is a frequently requested X3D feature.

- Some browsers (e.g. Xj3D) allow setting a parameter for primitive quality.
- Maybe a new field will eventually be added to the X3D specification, or maybe not. Probably this won't ever happen, because X3D browsers like to be small and fast. Adding non-essential features is usually referred to as “code bloat.”
- Authors can generate their own geometry (e.g. IndexedFaceSet) if they do not want to live with the uncertainty of browser quality when drawing geometry primitives.

Tesselation means how do we convert an arbitrary shape into triangles for fast rendering

Common field: *solid*

In 3D graphics, all triangles have 2 sides

- Graphics term: backface culling only draws front sides

The *solid* field defines whether a geometry node has an inside or not, with a default value of true

- *solid*='true' means do not render (draw) the inside
- *solid*='false' means render both inside and outside

This approach reduces the number of polygons needing to be drawn, thus improving performance

Confusing if user gets lost inside invisible geometry

- **Hint:** set *solid*='false' to draw both sides
- web|3D
CONSORTIUM



9

Mnemonic device: **solid like a brick!** We don't need to draw the insides of a brick, so only single-sided rendering of the outside is needed. You can't go inside a brick.

Let's check to see what the X3D Specification says about the solid field for the nodes in this chapter. It is the final authority on what the correct behavior is for X3D Scenes.

[X3D Resources](#), [Authoring Support](#), [X3D Specifications](#),

[X3D Architecture and base components Edition 2](#)

<http://www.web3d.org/x3d/specifications/ISO-IEC-19775-1.2-X3D-AbstractSpecification/Part01/Ar>

For each node, we look in Geometry3D and Text Components for default values

<http://www.web3d.org/x3d/specifications/ISO-IEC-19775-1.2-X3D-AbstractSpecification/Part01/co>

- Box *solid*='true' for 1-sided rendering
- Cone *solid*='true'
- Cylinder *solid*='true'
- Sphere *solid*='true'

But:

- Text *solid*='false' for 2-sided rendering
 - Why? for human performance! Invisible text is not much use, so we made a special case for the Text node

[back to Table of Contents](#)

X3D Nodes and Examples



10

Shape parent with geometry child

```
<Shape>  
  <Box size='1 2 3'/>  
  <Appearance>  
    <Material/>  
  </Appearance>  
</Shape>
```

Shape must be parent node, can only hold one geometry node
Appearance and Material nodes define colors, transparency, etc.

```
<Shape>  
  <Sphere radius='1'/>  
  <Appearance>  
    <Material/>  
  </Appearance>  
</Shape>
```

Primitives have simple dimensions

- Typical volume ~ 1 m radius

All units are in meters
Note parent-child relationships

Box node

Six-sided rectangular parallelepiped

- meaning: not necessarily a cube, but it can be
- Three non-zero non-negative *size* dimensions for x y z

Centered at local origin

size field has X3D data type **SFVec3f**

- **SF Vec** = Single-field vector
- array length of 0 or 1 only
- **3f** = 3 floating-point values
- Default *size*='2 2 2'

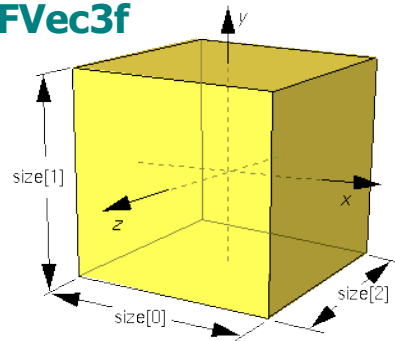
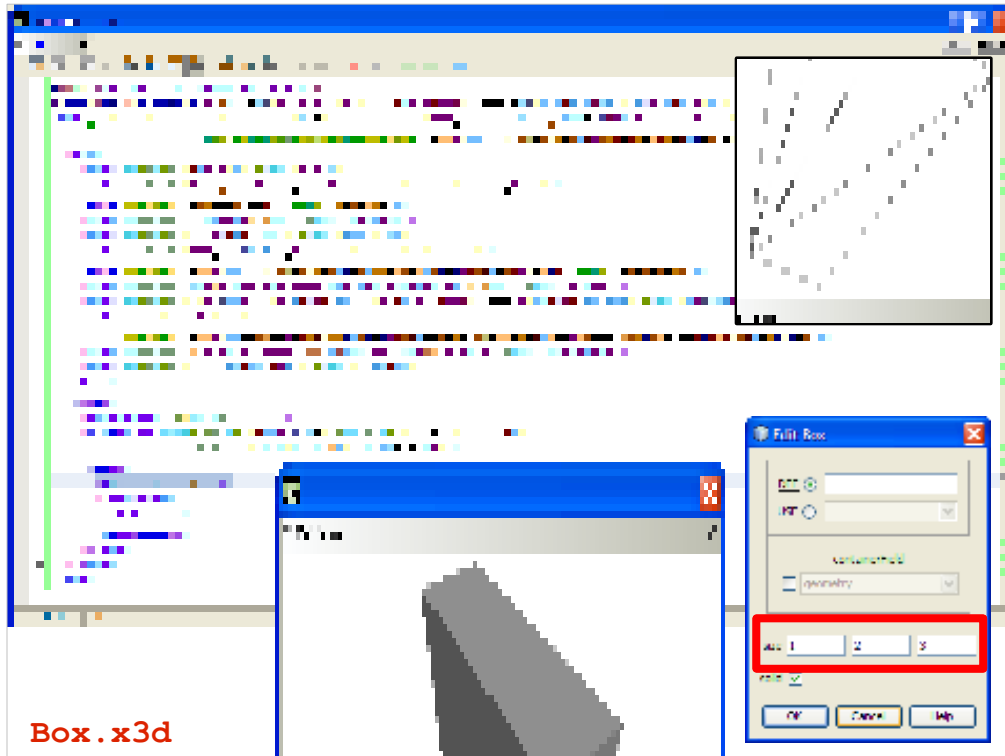


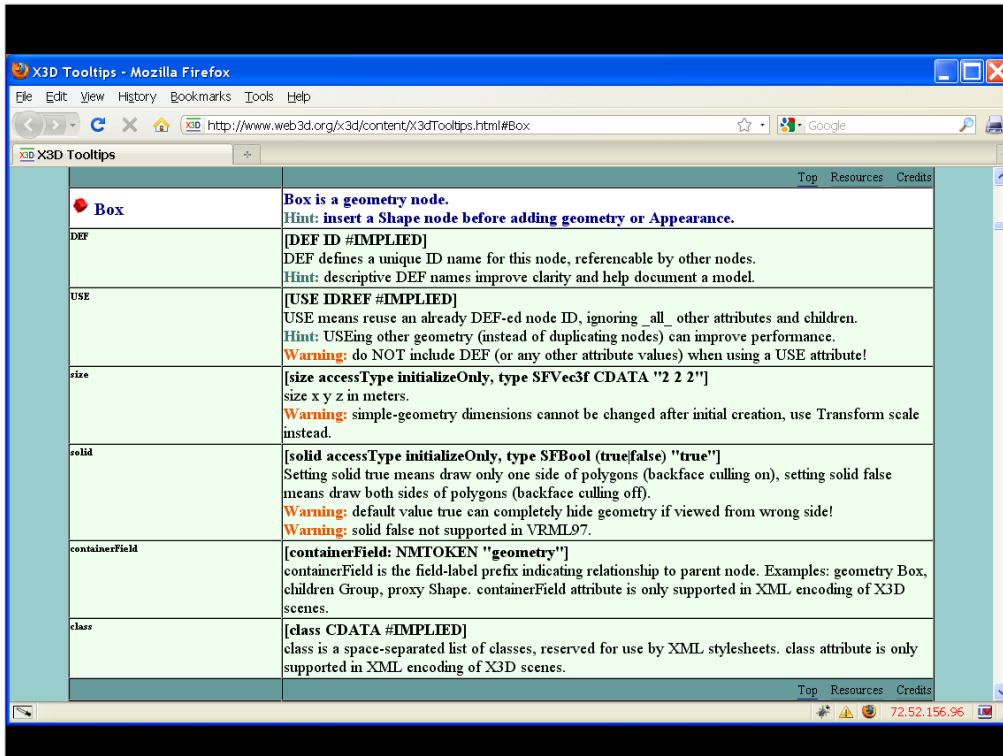
Image used with permission from X3D Abstract Specification, ISO/IEC 19775-1:2008

- Figure 13.1 — Box node
- <http://www.web3d.org/x3d/specifications>



From Xj3D help: Alt-W toggles Wireframe rendering mode. Wireframe rendering of geometry is a special feature offered by some browsers and cannot be set by an X3D author.

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Box.x3d>



<http://www.web3d.org/x3d/content/X3dTooltips.html#Box>

Cone node

Circular *bottomRadius*
non-zero non-negative
height above bottom
Centered at local origin
Can hide different parts

- *side*='false'
- *bottom*='false'

Default *height*='2'
bottomRadius='1'

Set *side*='false' (for bottom only) to define flat circle

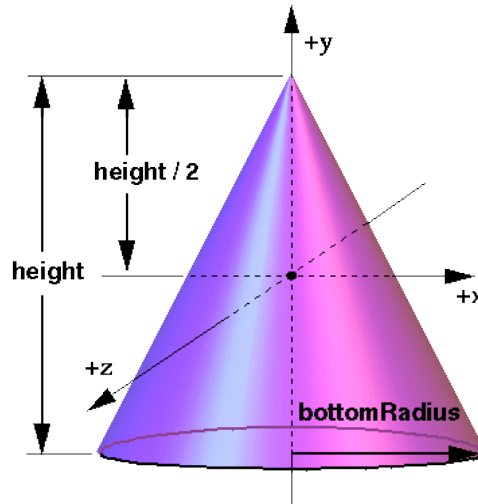
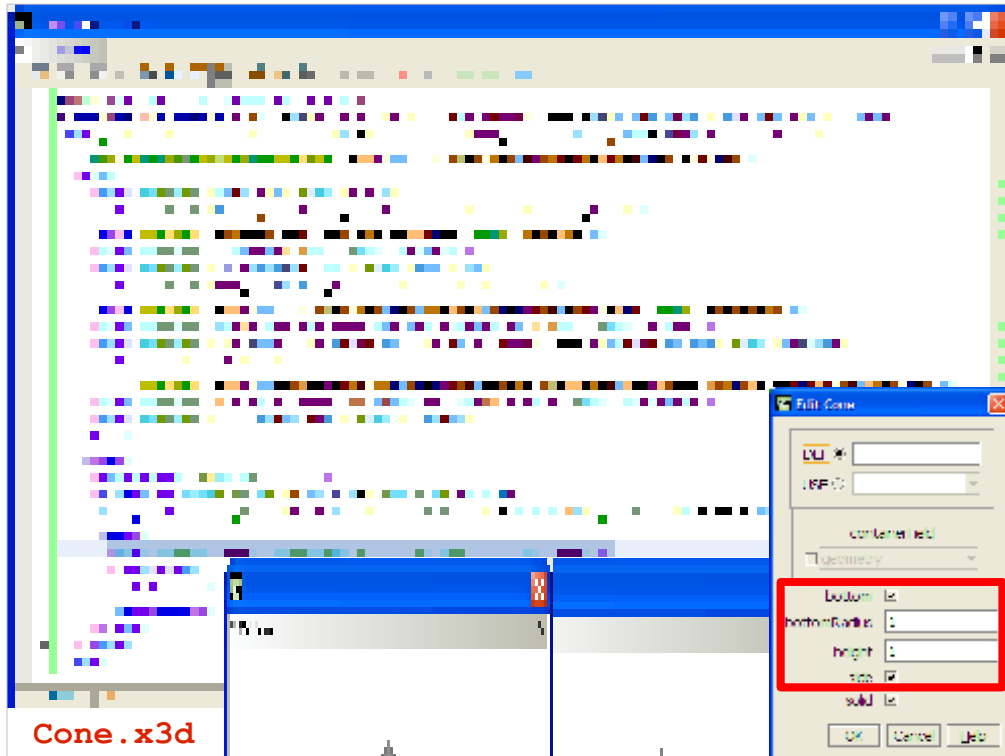


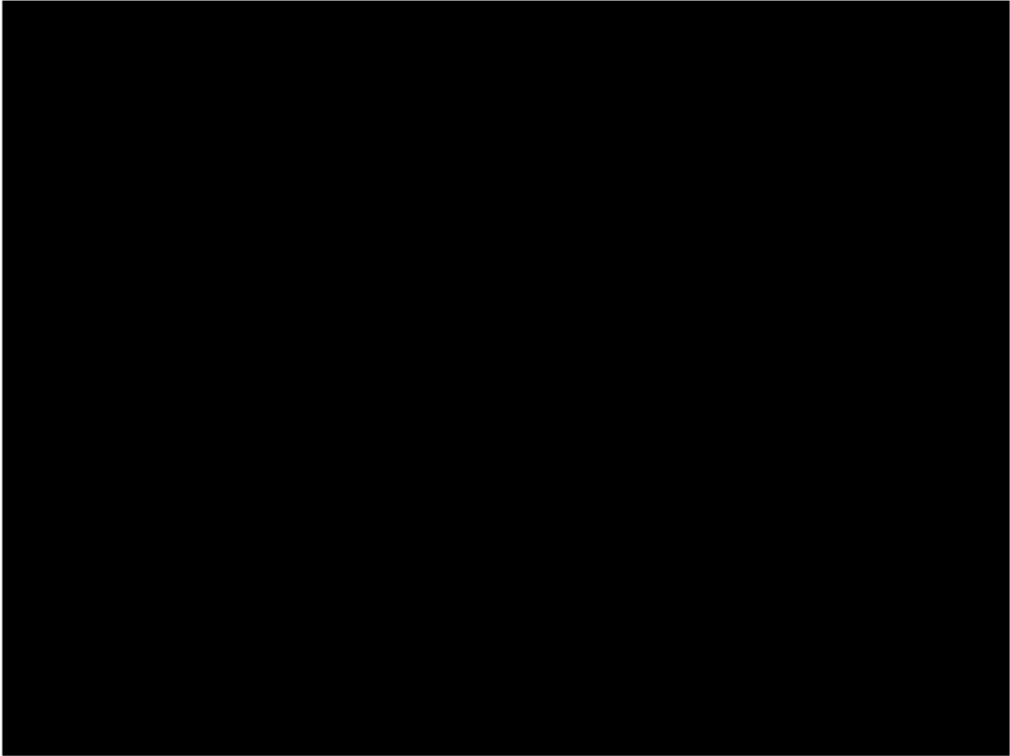
Image used with permission from X3D Abstract Specification, ISO/IEC 19775-1:2008

- Figure 13.2 — Cone node
- <http://www.web3d.org/x3d/specifications>



From Xj3D help: Alt-W toggles Wireframe rendering mode. Wireframe rendering of geometry is a special feature offered by some browsers and cannot be set by an X3D author.

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Cone.x3d>



<http://www.web3d.org/x3d/content/X3dTooltips.html#Cone>

Cylinder node

Right-angle cylinder with top and bottom caps

Non-zero non-negative *height* above bottom

Circular *radius*

Centered at local origin

Can hide different parts

- *side*='false'
- *top*='false'
- *bottom*='false'

Default values are

height='2' *radius*='1'

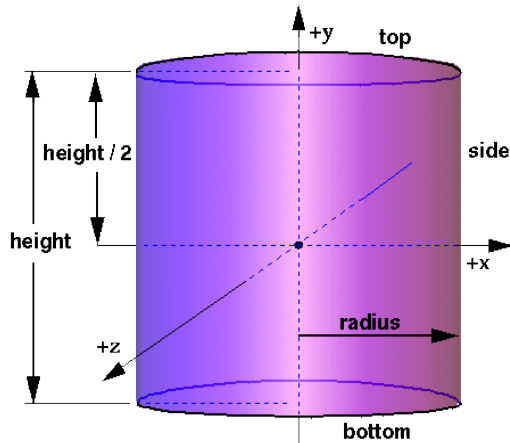
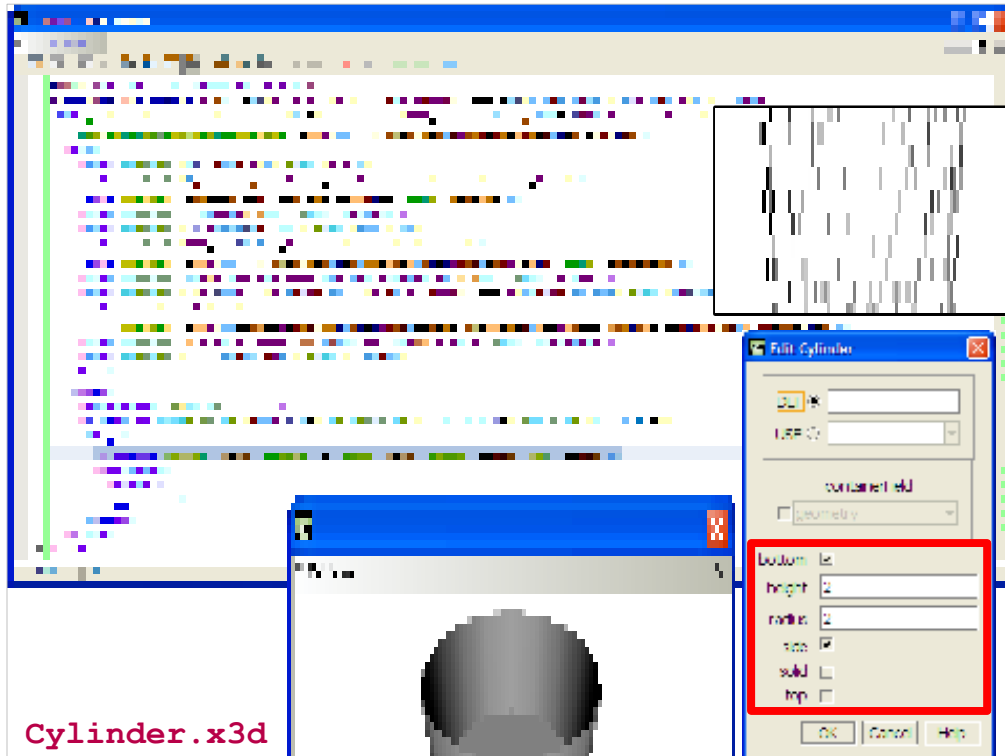


Image used with permission from X3D Abstract Specification, ISO/IEC 19775-1:2008

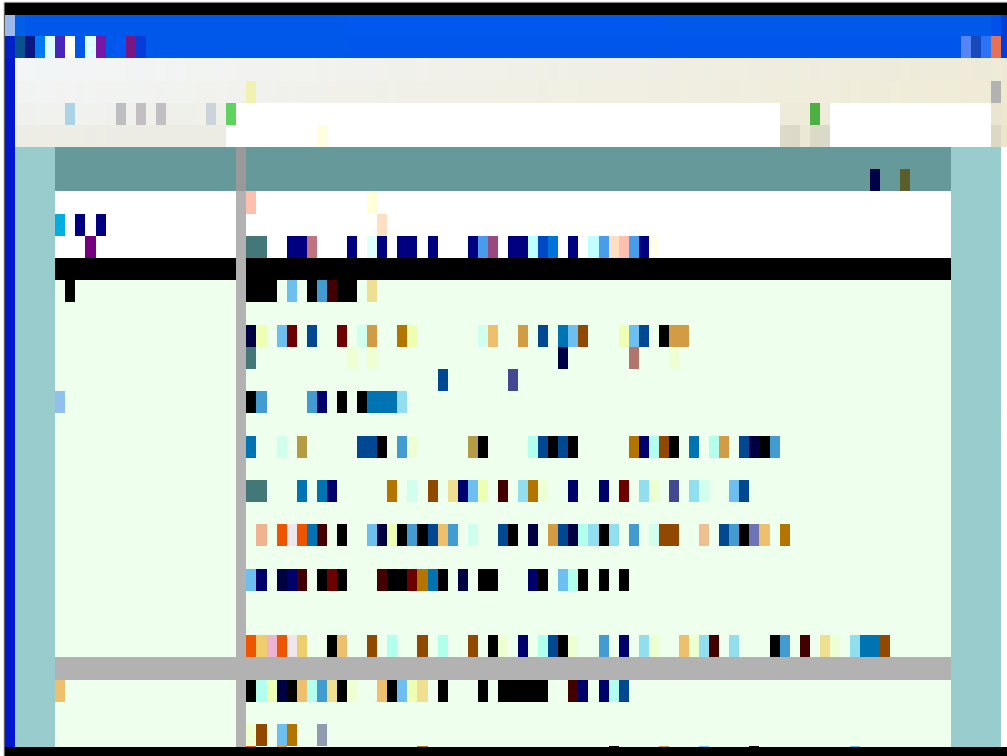
- Figure 13.3 — Cylinder node
- <http://www.web3d.org/x3d/specifications>



From Xj3D help: Alt-W toggles wireframe rendering mode. Wireframe rendering of geometry is a special feature offered by some browsers and cannot be set by an X3D author.

Note that wireframe mode can play tricks with your sense of perspective because all line segments are drawn exactly 1-pixel wide.

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Cylinder.x3d>



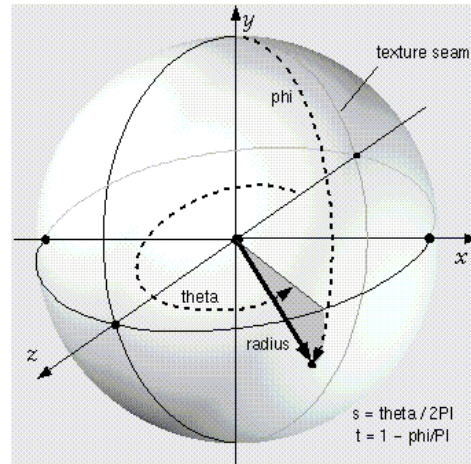
<http://www.web3d.org/x3d/content/X3dTooltips.html#Cone>

Sphere node

Circular *radius*

Centered at local origin

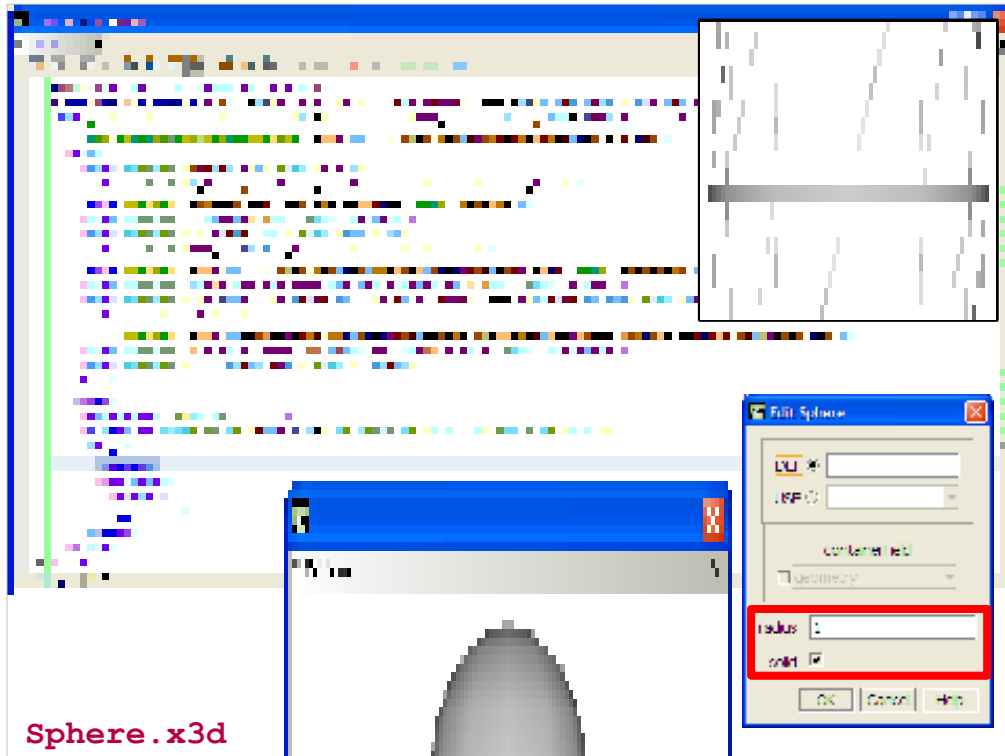
- phi and theta are implicit
- not defined by author



Note that angles are defined in the figure, but these really can't be referenced within the Sphere node. So this specification figure is really much more complex than it needs to be.

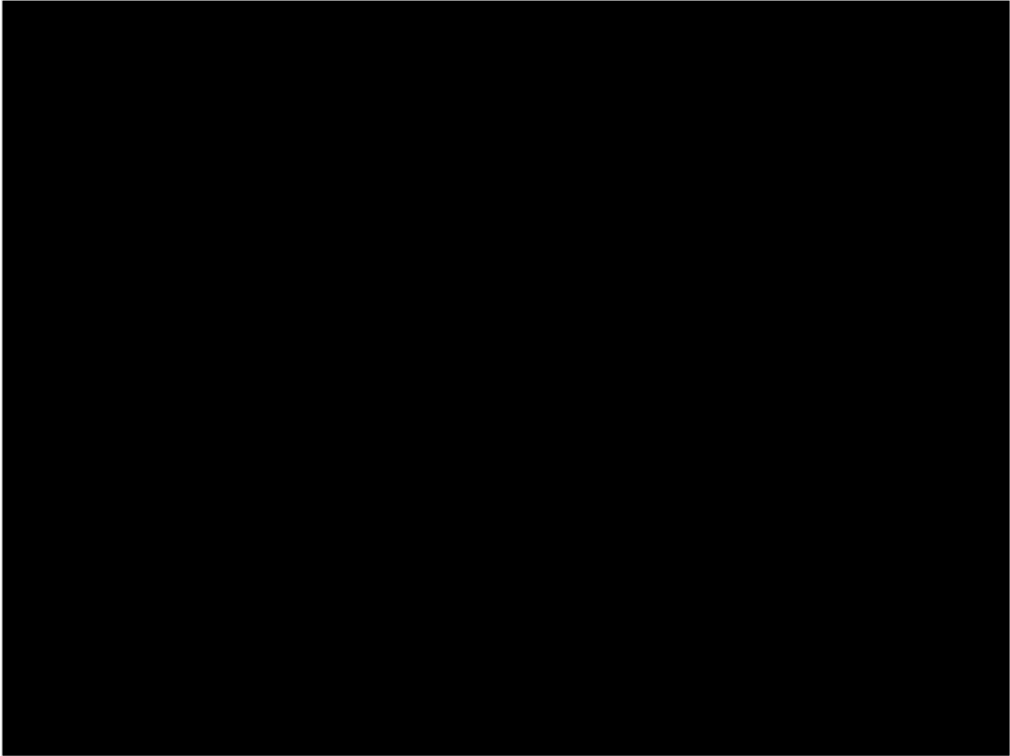
Image used with permission from X3D Abstract Specification, ISO/IEC 19775-1:2008

- Figure 13.8 — Sphere node
- <http://www.web3d.org/x3d/specifications>



From Xj3D help: Alt-W toggles Wireframe rendering mode. Wireframe rendering of geometry is a special feature offered by some browsers and cannot be set by an X3D author.

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Sphere.x3d>



<http://www.web3d.org/x3d/content/X3dTooltips.html#Sphere>

Text node

Produce readable flat, 2D text strings in X3D world

string field is MFString array of “quoted strings”

- Each “quoted string” appears on a separate line

length field is MFFloat array of lengths for each line

- Can shrink or stretch size of each line if needed

maxExtent is maximum length for all substring lines

Note characters have no 3D depth

- Flat when viewed from alongside
- Typically viewable from behind since default is *solid*='false'
- **Hint:** use Billboard to face user



Commonly we don't have to worry about the *length* and *maxExtent* fields. They are computed automatically and internally by the X3D view. Defining values for these two fields is a specialty technique when needed for very precise authoring.

The other primitive geometry nodes have default *solid*='true' for 1-sided rendering.

Billboard node is covered in Chapter 4, Viewing and Navigation. Use of Billboard as a parent node keeps the front face of Text facing users in order to maintain readability.

Interesting reference describing in detail how to make text and other information comprehensible and readable despite self-occluding clutter in detailed scenes:

- *Information Visualization in Information-Rich Virtual Environments*, dissertation, Nick Polys at Virginia Tech.



From Xj3D help: Alt-W toggles Wireframe rendering mode. Wireframe rendering of geometry is a special feature offered by some browsers and cannot be set by an X3D author.

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Text.x3d>

Inserting apostrophes, ampersands, and quotation marks into Text strings

Character entity definitions are XML encodings

- Character entities are also known as *escape characters*
- apostrophe ' is ' & is & " is "
- <http://www.w3.org/TR/REC-html40/sgml/entities.html>

Precede embedded "quote marks" with backslash \" to differentiate from line-delimiting quote marks

Suggested XML to escape Text node's string field:

- single quote (apostrophe) as XML attribute delimiter
- string=' "Hello from \"Monterey\" ' ' or
- string=' "Hello from \""Monterey"" ' ' or
- string=' "A friend's new car" "just arrived" ' ' or

HTML spec lists all character entities at

<http://www.w3.org/TR/REC-html40/sgml/entities.html>

Although alphabetic abbreviations are provided for most numeric code, support for the abbreviations is an optional feature for browsers.

Note that decimal character-code values are expressed as &#___; while hexadecimal character-code values are expressed as &#x___;

Support for 3D rendering of special fonts and special characters in the Text node is dependent on the X3D viewer utilized by the end user.

Special characters can usually be cut/pasted from other applications into an X3D file or into the *string* array on the X3D-Edit pane for editing a Text node.

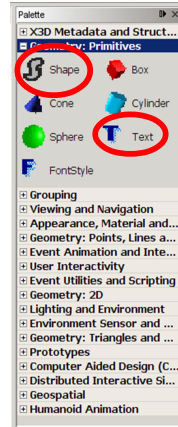
TODO: add X3D-Edit features to

- select and insert special characters
- Convert unicode symbology into escape codes
- Additional Unicode references

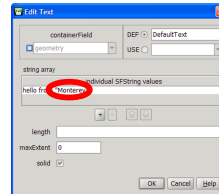
Try it yourself



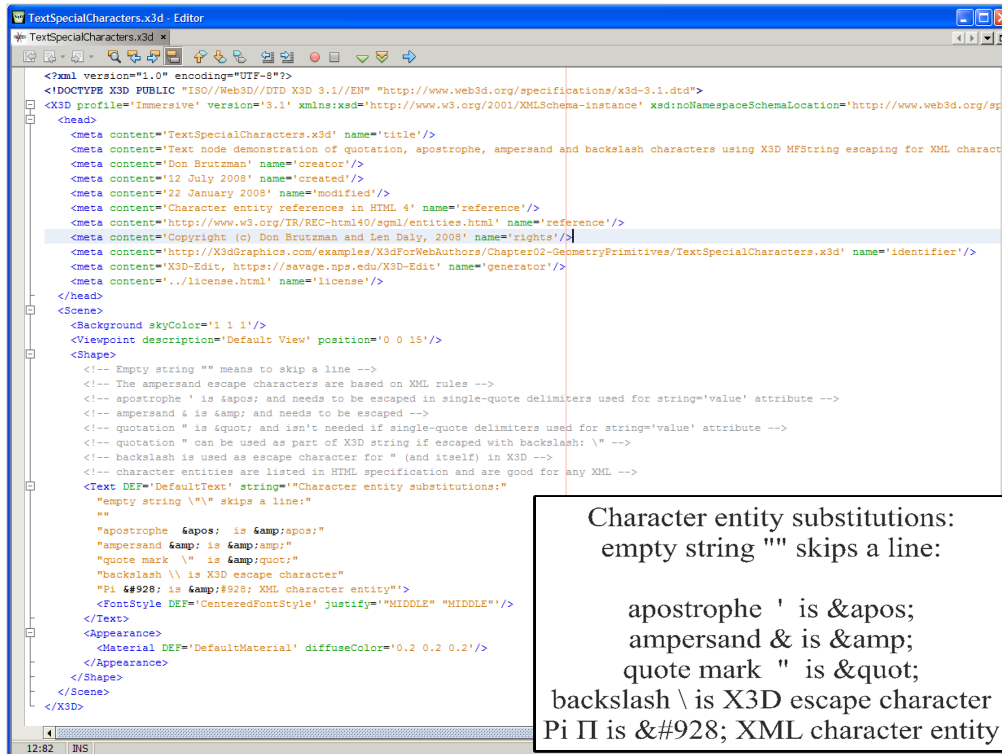
1. Create a new scene by clicking the New X3D Scene button, or else select using menus (*File, New X3D, New X3D Scene*)
2. Open the palette for *Geometry: Primitives*
3. Drag a new Shape node into the scene graph where XML comment says
`<!-- Scene graph nodes are added here -->`
7. Drag a new Text node into the scene graph where the XML comment says
`<!-- Add a single geometry node here -->`
9. Right click on the Text node, cut and paste the text into the `string` field. Be sure to include all of the double quotes, don't paste the single quote delimiters.
`<Text string='Hello from "My Home Town"' />`
11. Right-click the context menu to refresh or redraw in Xj3D:



web|3D
CONSORTIUM



27



<http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/TextSpecialCharacters.x3d>

Note that this slide's Text box is actual X3D output, captured as a screen snapshot.

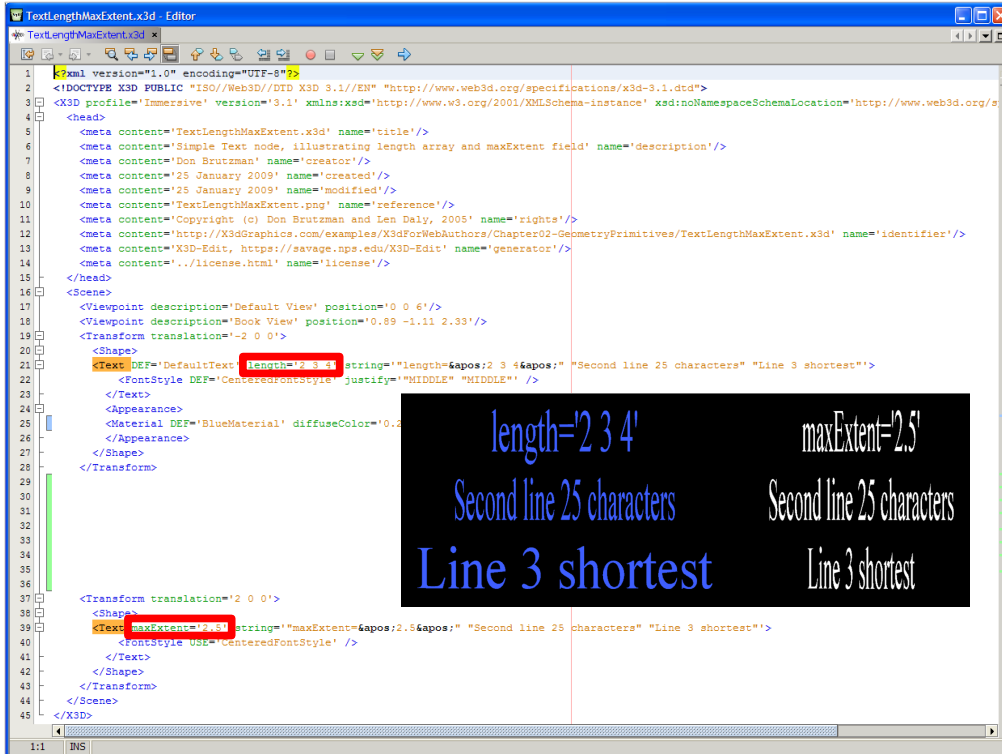
Usually special escape characters aren't needed for most scenes.

Character entities are crucial for creating multilingual scenes using the internationalization (I18N) capabilities of XML.

<http://www.w3.org/International>

TODO:

- Joke about thermos bottle
- Escape-character conundrum: how to represent escape characters themselves?
- Qui custodiet custodies? Who watches the watchers?



Example TextLengthMaxExtent.x3d online at

<http://X3dGraphics.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/TextLengthMaxExtent.x3d>

From X3D Specification, 15.4.2 Text:

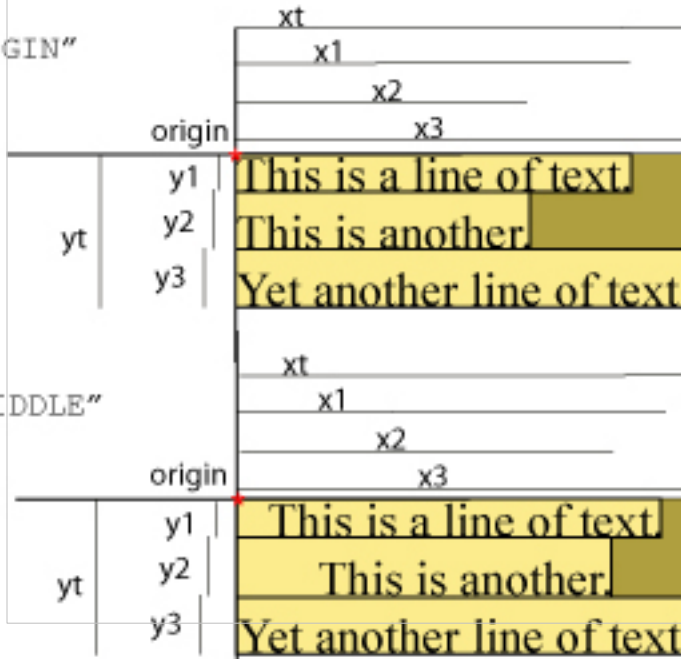
<http://www.web3d.org/x3d/specifications/ISO-IEC-FDIS-19775-1.2-X3D-AbstractSpecification/Part01/components/text.html#Text>

“The *length* field contains an MFFloat value that specifies the length of each text string in the local coordinate system.”

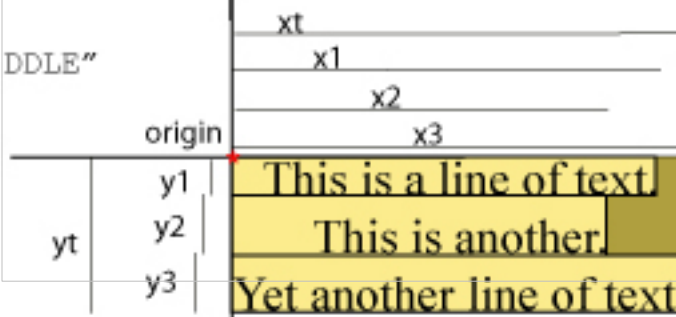
“The *maxExtent* field limits and compresses all of the text strings if the length of the maximum string is longer than the maximum extent, as measured in the local coordinate system. If the text string with the maximum length is shorter than the *maxExtent*, then there is no compressing.”

Horizontal or vertical adjustments

justify "BEGIN"



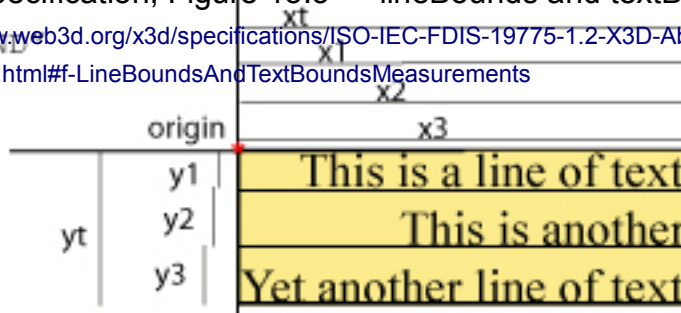
justify "MIDDLE"



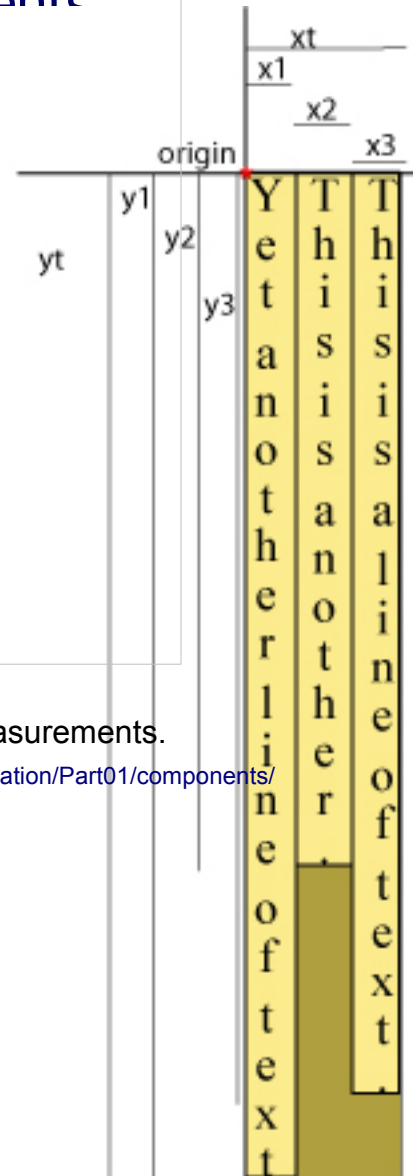
X3D Specification, Figure 15.3 — lineBounds and textBounds measurements.

<http://www.web3d.org/x3d/specifications/ISO-IEC-FDIS-19775-1.2-X3D-AbstractSpecification/Part01/components/text.html#f-LineBoundsAndTextBoundsMeasurements>

justify "END"



lineBounds [x1 y1, x2 y2, x3 y3]
textBounds [xt yt]

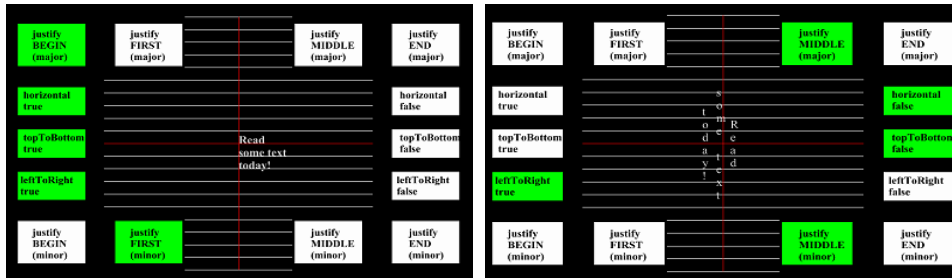


T Text	Text is a geometry node that can contain a FontStyle node. Hint: insert a Shape node before adding geometry or Appearance. You can also substitute a type-matched ProtoInstance.
DEF	[DEF ID #IMPLIED] DEF defines a unique ID name for this node, referencable by other nodes. Hint: descriptive DEF names improve clarity and help document a model.
USE	[USE IDREF #IMPLIED] USE means reuse an already DEF-ed node ID, ignoring all other attributes and children. Hint: USEing other geometry (instead of duplicating nodes) can improve performance. Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!
string	[string: accessType inputOutput, type MFString CDATA #IMPLIED] Single or multiple string values to present as Text. Hint: Strings can have multiple values, so separate each string by quote marks Hint: Strings can contain quote marks by first escaping them with a backslash example: "say \"hello\" please" Hint: many XML tools substitute XML character references automatically if needed (like & for & or " for ").
length	[length: accessType inputOutput, type MFFloat CDATA #IMPLIED] Array of length values for each text string in the local coordinate system. Each string is stretched or compressed to fit.
maxExtent	[maxExtent: accessType inputOutput, type SFFloat CDATA "0.0"] Limits/compresses all text strings if max string length is longer than maxExtent, as measured in local coordinate system.
solid	[solid: accessType initializeOnly, type SFBool (true/false) "true"] Setting solid true means draw only one side of polygons (backface culling on), setting solid false means draw both sides of polygons (backface culling off). Warning: default value true can completely hide geometry if viewed from wrong side! Warning: solid false not supported in VRML97.
lineBounds	[lineBounds: accessType outputOnly, type MFVec2f CDATA #IMPLIED] Array of 2D bounding box values for each line of text in the local coordinate system.
textBounds	[textBounds: accessType outputOnly, type SFVec2f CDATA #IMPLIED] 2D bounding box value for all lines of text in the local coordinate system.
containerField	[containerField: NMTOKEN "geometry"] containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.
class	[class CDATA #IMPLIED] class is a space-separated list of classes, reserved for use by XML stylesheets. class attribute is only supported in XML encoding of X3D scenes.

<http://www.web3d.org/x3d/content/X3dTooltips.html#Text>

FontStyle node

Defines *size*, font *family*, layout directions and justification, language, and style for Text strings



<http://www.web3d.org/x3d/content/examples/ConformanceNist/Appearance/FontStyle/driver.x3d>



32

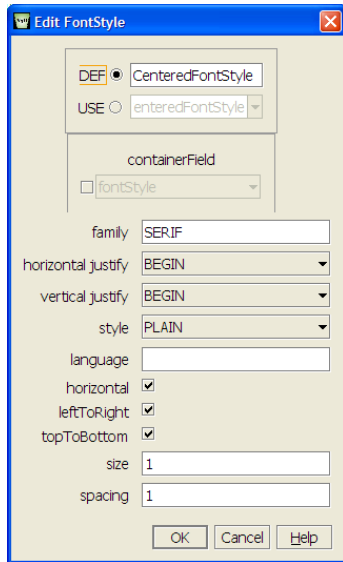
These images correspond to Figures 2.11, 2.12 on pp. 57-58 in *X3D for Web Authors*. Note that the image provided for Figure 2.11 in the published book is incorrect.

These example snapshots are taken from the FontStyle demo scene in the NIST Conformance Suite.

<http://www.web3d.org/x3d/content/examples/ConformanceNist>

<http://www.web3d.org/x3d/content/examples/ConformanceNist/Appearance/FontStyle/driver.x3d>

```
<FontStyle justify=' MIDDLE MIDDLE ' />
```



FontStyle is only allowed as child of a Text node

- FontStyle modifies that parent

Other supported default font family values are SANS (serif) and TYPEWRITER

- Additional font families require special browser support

Other field values support internationalization (I18N) and localization (L10N)

- DEF, USE for consistent look

Dialog box is from the same Text.x3d scene:

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/Text.x3d>

It is possible to define a single font style for consistent use throughout an X3D scene. Create and define (DEF) a single FontStyle node near the top of a scene, then USE it whenever a Text node appears elsewhere. This makes it easy to maintain or change consistent font styles throughout a document. Example:

```
<Shape>
  <Text string='Hello from \"Monterey\"'>
    <FontStyle DEF="MiddleJustify" justify='\"MIDDLE\" \"MIDDLE\"' />
  </Text>
</Shape>
<!-- ... and then later in the scene: -->
<Shape>
  <Text string='Hello again from \"somewhere else\"'>
    <FontStyle USE='MiddleJustify' />
  </Text>
</Shape>
```

DEF and USE are covered in Chapter 3, Grouping nodes.

FontStyle values, X3D Specification

Table 15.2 – Major Alignment, *horizontal* = TRUE

<i>justify</i> Enumerant	<i>leftToRight</i> = TRUE	<i>leftToRight</i> = FALSE
FIRST	Left edge of each line	Right edge of each line
BEGIN	Left edge of each line	Right edge of each line
MIDDLE	Centred about X-axis	Centred about X-axis
END	Right edge of each line	Left edge of each line

Table 15.3 – Major Alignment, *horizontal* = FALSE

<i>justify</i> Enumerant	<i>topToBottom</i> = TRUE	<i>topToBottom</i> = FALSE
FIRST	Top edge of each line	Bottom edge of each line
BEGIN	Top edge of each line	Bottom edge of each line
MIDDLE	Centred about Y-axis	Centre about Y-axis
END	Bottom edge of each line	Top edge of each line

There is no need to memorize this information. Rather, the key part is to remember that this information is available in the book and in the specification when needed for internationalization (non-English) text or special layouts.

From X3D abstract specification, Tables 15.2 and 15.3

<http://www.web3d.org/x3d/specifications/ISO-IEC-FDIS-19775-1.2-X3D-AbstractSpecification/Part01/components/text.html#-MajorAlignhorizTRUE>

FontStyle values, X3D Specification

Table 15.4 – Minor Alignment, *horizontal* = TRUE

<i>justify</i> Enumerant	<i>topToBottom</i> = TRUE	<i>topToBottom</i> = FALSE
FIRST	Baseline of first line	Baseline of first line
BEGIN	Top edge of first line	Bottom edge of first line
MIDDLE	Centred about Y-axis	Centred about Y-axis
END	Bottom edge of last line	Top edge of last line

Table 15.5 – Minor Alignment, *horizontal* = FALSE

<i>justify</i> Enumerant	<i>leftToRight</i> = TRUE	<i>leftToRight</i> = FALSE
FIRST	Left edge of first line	Right edge of first line
BEGIN	Left edge of first line	Right edge of first line
MIDDLE	Centred about X-axis	Centred about X-axis
END	Right edge of last line	Left edge of last line

There is no need to memorize this information. Rather, the key part is to remember that this information is available in the book and in the specification when needed for internationalization (non-English) text or special layouts.

From X3D abstract specification, Tables 15.4 and 15.5

<http://www.web3d.org/x3d/specifications/ISO-IEC-FDIS-19775-1.2-X3D-AbstractSpecification/Part01/components/text.html#-MinorAlignhorizTRUE>

X3D Specification Tables 15.6 and 15.7

Key			
⊕	minor = "FIRST"	+	minor = "BEGIN"
⊞	minor = "MIDDLE"	*	minor = "END"

	major = "BEGIN" or "FIRST"		major = "MIDDLE"		major = "END"		
	leftToRight		leftToRight		leftToRight		
	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	
topToBottom	TRUE	⊕ Read ⊕ ⊞ some text ⊞ ⊕ today! ⊕	dae⊕ bret emos⊞ !yadot⊞	Read⊕ some text⊞ today!⊕	dae⊕ bret emos⊞ !yadot⊞	Read⊕ some text⊞ today!⊕	dae⊕ bret emos⊞ !yadot⊞
	FALSE	⊕ today! ⊕ ⊞ some text ⊞ ⊕ Read ⊕	!yadot⊞ bret emos⊞ dae⊕	⊕ today! ⊕ ⊞ some text ⊞ ⊕ Read ⊕	!yadot⊞ bret emos⊞ dae⊕	⊕ today! ⊕ ⊞ some text ⊞ ⊕ Read ⊕	!yadot⊞ bret emos⊞ dae⊕

		major = "BEGIN" or "FIRST"		major = "MIDDLE"		major = "END"	
		leftToRight		leftToRight		leftToRight	
		TRUE	FALSE	TRUE	FALSE	TRUE	FALSE
topToBottom	TRUE	⊕ Read ⊕ ⊞ some text ⊞ ⊕ today! ⊕ ⊕ text ⊕	* today! * * some text * * Read *	s o m e t o d a y t e x t	s o m e t o d a y t e x t	s o m e t o d a y t e x t	s o m e t o d a y t e x t
	FALSE	t x e t ! y a d o t d a e m o o s	! y a d o t ! y a d o t ! y a d o t	t x e t ! y a d o t d a e m o o s	t x e t ! y a d o t d a e m o o s	⊕ dae⊕ ⊞ text⊞ ⊕ today!⊕	* today! * * text * * Read *

Note: In every case, the "FIRST" minor axis marker ⊕ is coincident with the "BEGIN" minor axis marker + (and is offset for presentation purposes only).

There is no need to memorize this information. Rather, the key part is to remember that this information is available in the book and in the specification when needed for internationalization (non-English) text or special layouts.

From X3D abstract specification, Tables 15.6 and 15.7

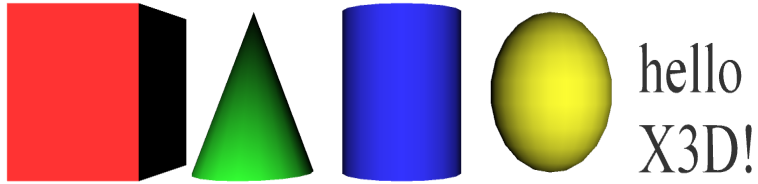
<http://www.web3d.org/x3d/specifications/ISO-IEC-FDIS-19775-1.2-X3D-AbstractSpecification/Part01/components/text.html#-KeyforTables>

FontStyle	
	FontStyle defines the size, family, and style used for Text nodes. Hint: first add a Text node as parent.
DEF	[DEF ID #IMPLIED] DEF defines a unique ID name for this node, referencable by other nodes. Hint: descriptive DEF names improve clarity and help document a model.
USE	[USE IDREF #IMPLIED] USE means reuse an already DEF-ed node ID, ignoring all other attributes and children. Hint: USEing other geometry (instead of duplicating nodes) can improve performance. Warning: do NOT include DEF (or any other attribute values) when using a USE attribute!
family	[family: accessType initializeOnly, type MFString CDATA "SERIF"] Sequence of font family names in preference order - browsers use first supported family. Supported values include "SERIF" "SANS" "TYPEWRITER". Hint: SERIF and SANS are variable-width fonts (for example, Roman and Arial). Hint: TYPEWRITER is a fixed-width font (for example, Courier). Hint: Strings can have multiple values, so separate "each string" by " " quote marks".
style	[style: accessType initializeOnly, type SFString CDATA (PLAIN BOLD ITALIC BOLDITALIC) "PLAIN"] Pick one of four values for text style.
justify	[justify: accessType initializeOnly, type MFString CDATA "BEGIN"] Two string values are provided for major and minor axis alignment, possible values are "FIRST" "BEGIN" "MIDDLE" "END" Example: "MIDDLE" "MIDDLE". Hint: Strings can have multiple values, so separate "each string" by " " quote marks".
size	[size: accessType initializeOnly, type SFFloat CDATA "1.0"] Nominal height (in local coordinate system) of text glyphs Also sets default spacing between adjacent lines of text.
spacing	[spacing: accessType initializeOnly, type SFFloat CDATA "1.0"] Adjustment factor for line spacing between adjacent lines of text.
language	[language: accessType initializeOnly, type SFString CDATA #IMPLIED] Language codes consist of a primary code and a (possibly empty) series of subcodes. [language-code = primary-code ("-" subcode) *] Two-letter primary codes are reserved for language abbreviations. [RFC1766, http://www.ietf.org/rfc/rfc1766.txt] Two-letter primary codes include en (English), fr (French), de (German), it (Italian), nl (Dutch), el (Greek), es (Spanish), pt (Portuguese), ar (Arabic), he (Hebrew), ru (Russian), zh (Chinese), ja (Japanese), hi (Hindi), ur (Urdu), and sa (Sanskrit). Any two-letter subcode is understood to be a country code. [ISO3166 or http://www.oasis-open.org/cover/iso639a.html]
horizontal	[horizontal: accessType initializeOnly, type SFBool (true false) "true"] Whether text direction is horizontal (true) or vertical (false).
leftToRight	[leftToRight: accessType initializeOnly, type SFBool (true false) "true"] Whether text direction is left-to-right (true) or right-to-left (false).
topToBottom	[topToBottom: accessType initializeOnly, type SFBool (true false) "true"] Whether text direction is top-to-bottom (true) or bottom-to-top (false).
containerField	[containerField: NMTOKEN "FontStyle"] containerField is the field-label prefix indicating relationship to parent node. Examples: geometry Box, children Group, proxy Shape. containerField attribute is only supported in XML encoding of X3D scenes.
class	[class CDATA #IMPLIED] class is a space-separated list of classes, reserved for use by XML stylesheets. class attribute is only supported in XML encoding of X3D scenes.

<http://www.web3d.org/x3d/content/X3dTooltips.html#FontStyle>

Review

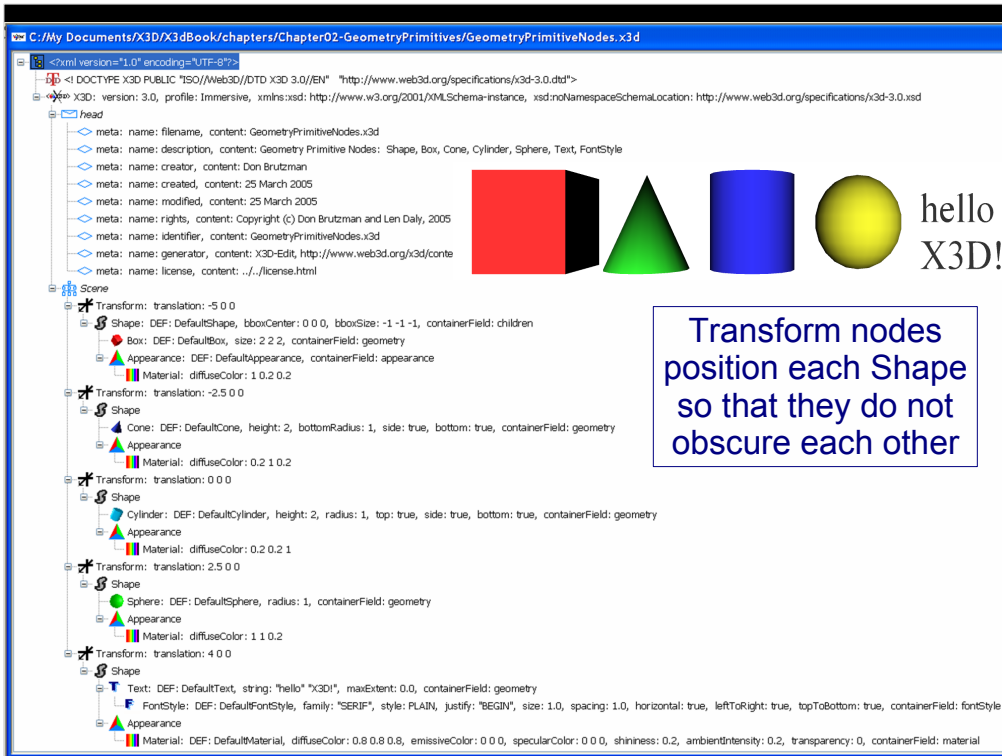
Geometry Primitives



Primitives are simple geometric constructs
Shape, geometry, Appearance, Material pattern
Browsers decide implementation details,
including quality of tessellation resolution

web|3D
CONSORTIUM





Figures 2.1 and 2.2, page 39, *X3D for Web Authors*

<http://www.x3dbook.com/examples/X3dForWebAuthors/Chapter02-GeometryPrimitives/GeometryPrimitiveNodes.x3d>

This scene-graph screen snapshot was taken using X3D-Edit 3.1.

[back to Table of Contents](#)

Additional Resources



40

Multilingual X3D tooltips

X3D tooltips are available online and bundled in the X3D-Edit help pages

X3D tooltips are available in the following languages

- English
- German
- Portuguese
- Chinese
- Italian
- Spanish
- French
- Korean

Translations into other languages are welcome

- <http://www.web3d.org/x3d/content/X3dTooltips.html>



41

X3D tooltip links:

<http://www.web3d.org/x3d/content/X3dTooltipsChinese.html>

<http://www.web3d.org/x3d/content/X3dTooltips.html> (English)

<http://www.web3d.org/x3d/content/X3dTooltipsFrench.html>

<http://www.web3d.org/x3d/content/X3dTooltipsGerman.html>

<http://www.web3d.org/x3d/content/X3dTooltipsItalian.html>

<http://www.web3d.org/x3d/content/X3dTooltipsPortuguese.html>

<http://www.web3d.org/x3d/content/X3dTooltipsSpanish.html>

X3D spec excerpt for Shape node

```
Shape : X3DShapeNode {
  SFNode [in,out] appearance NULL [X3DAppearanceNode]
  SFNode [in,out] geometry NULL [X3DGeometryNode]
  SFNode [in,out] metadata NULL [X3DMetadataObject]
  SFVec3f [] bboxCenter 0 0 0 (-*,*)
  SFVec3f [] bboxSize -1 -1 -1 [0,*] or -1 -1 -1
}
```

The Shape node has two fields, *appearance* and *geometry*, which are used to create rendered objects in the world. The *appearance* field contains an Appearance node that specifies the visual attributes (e.g., material and texture) to be applied to the geometry. The *geometry* field contains a geometry node. The specified geometry node is rendered with the specified appearance nodes applied. See [12.2 Concepts](#) for more information.

[17 Lighting component](#) contains details of the X3D lighting model and the interaction between Appearance nodes and geometry nodes.

If the *geometry* field is `NULL`, the object is not drawn.

The *bboxCenter* and *bboxSize* fields specify a bounding box that encloses the Shape node's geometry. This is a hint that may be used for optimization purposes. The results are undefined if the specified bounding box is smaller than the actual bounding box of the geometry at any time. A default *bboxSize* value, (-1, -1, -1), implies that the bounding box is not specified and, if needed, is calculated by the browser. A description of

All of the current approved X3D specifications are available via X3D-Edit help system.

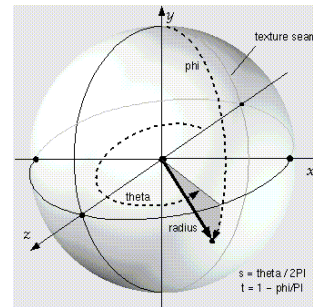
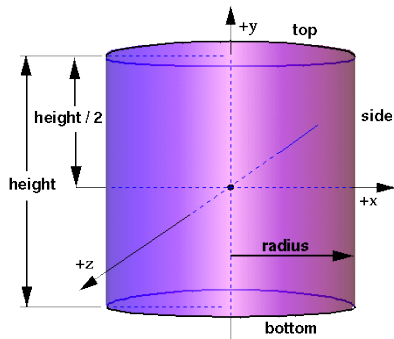
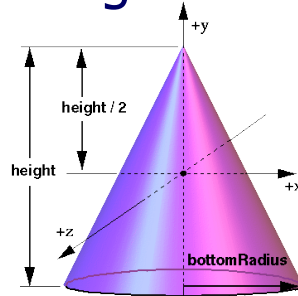
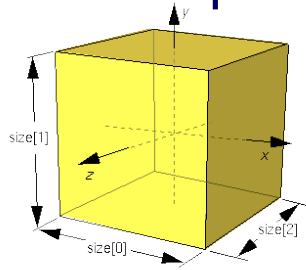
X3D specifications are online at

<http://www.web3d.org/x3d/specifications>

X3D specification feedback can be submitted at

http://www.web3d.org/x3d/specifications/spec_feedback

X3D Specification Diagrams



Images used with permission from X3D Abstract Specification, ISO/IEC 19775-1:2008

- Clause 13 : Geometry3D component
- <http://www.web3d.org/x3d/specifications>

Related concepts

DEF/USE nodes: chapter 3
Transform node: chapter 3
Viewpoint node: chapter 4
Appearance node: chapter 5
Material node: chapter 5
Background node: chapter 11

Bounding boxes: chapter 12

Geometry nodes

Chapter 2, Primitives

- Box, Cone, Cylinder, Sphere, Text / FontStyle

Chapter 6, Points Lines and Polygons

- PointSet, IndexedLineSet, IndexedFaceSet, ElevationGrid, Extrusion

Chapter 10, Geometry2D

- Arc2D, ArcClose2D, Circle2D, Disk2D, Polyline2D, Polypoint2D, Rectangle2D, TriangleSet2D

Chapter 13, Triangles and Quadrilaterals

- TriangleSet, TriangleStripSet, TriangleFanSet, QuadSet
- Both regular and Indexed versions

The principle that one geometry node goes inside each Shape, and next to each Appearance, is consistent for all the different geometry nodes available in X3D.

Advanced geometry nodes

Geospatial component

- GeoElevationGrid

NURBS component

- NurbsCurve, NurbsPatchSurface, NurbsSweptSurface, NurbsSwungSurface, NurbsTrimmedSurface

Programmable shaders component

- ComposedShader, PackagedShader, ProgramShader

Further information available in X3D Specification

- <http://www.web3d.org/x3d/specifications>



46

[back to Table of Contents](#)

Chapter Summary



47

Chapter Summary

Shape is a container node for a single piece of geometry

Geometry primitive nodes

- Box, Cone, Cylinder, Sphere, Text
- FontStyle supports Text

Numerous additional resources

- X3D tooltips, with multilingual versions available
- X3D specifications
- Each are integrated within X3D-Edit authoring tool

Suggested exercises

Modify an existing example scene to make it into another object. Become familiar with editing, XML validation, and reloading a scene to refresh the 3D rendering.

Create a simple object using only primitive geometric shapes

Demonstrate use of internationalization (I18N) text

Look at other examples for modeling ideas

- <http://www.web3d.org/x3d/content/examples/Vrml2.0Sourcebook>
- <http://www.web3d.org/x3d/content/examples/Basic/StudentProjects>



49

Advanced study:

Polys, Nicholas R., *Display Techniques in Information-Rich Virtual Environments*, Ph.D. Dissertation, Virginia Tech University, Blacksburg Virginia, June 2006. Available at <http://scholar.lib.vt.edu/theses/available/etd-06152006-024611>

Across domains, researchers, engineers, and designers are faced with large volumes of data that are heterogeneous in nature - including spatial, abstract, and temporal information. There are numerous design and technical challenges when considering the unification, management, and presentation of these information types. Most research and applications have focused on display techniques for each of the information types individually, but much less is known about how to represent the relationships between information types. This research explores the perceptual and usability impacts of data representations and layout algorithms for the next-generation of integrated information spaces.

We propose Information-Rich Virtual Environments (IRVEs) as a solution to challenges of integrated information spaces. In this presentation, we will demonstrate the application requirements and foundational technology of IRVEs and articulate crucial tradeoffs in IRVE information design. We will present a design space and evaluation methodology to explore the usability effects of these tradeoffs. Experimental results will be presented for a series of empirical usability evaluations that increase our understanding of how these tradeoffs can be resolved to improve user performance. Finally, we interpret the results through the models of Information Theory and Human Information Processing to derive new conclusions regarding the role of perceptual cues in determining user performance in IRVEs. These lessons are posed as a set of design guidelines to aid developers of new IRVE interfaces and specifications.

[back to Table of Contents](#)

References



50

References 1

X3D: Extensible 3D Graphics for Web Authors
by Don Brutzman and Leonard Daly, Morgan
Kaufmann Publishers, April 2007, 468 pages.



- Chapter 2, Geometry: Primitive Shapes
- <http://x3dGraphics.com>
- <http://x3dgraphics.com/examples/X3dForWebAuthors>

X3D Resources

- <http://www.web3d.org/x3d/content/examples/X3dResources.html>



51

References 2

X3D-Edit Authoring Tool

- <https://savage.nps.edu/X3D-Edit>

X3D Scene Authoring Hints

- <http://x3dgraphics.com/examples/X3dSceneAuthoringHints.ntml>



X3D Graphics Specification

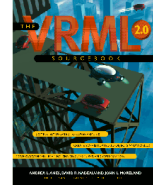
- <http://www.web3d.org/x3d/specifications>
- http://www.web3d.org/x3d/specifications/spec_feedback
- Available as help pages from within X3D-Edit



52

References 3

VRML 2.0 Sourcebook by Andrea L. Ames, David R. Nadeau, and John L. Moreland, John Wiley & Sons, 1996.

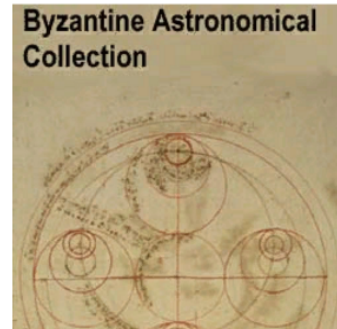
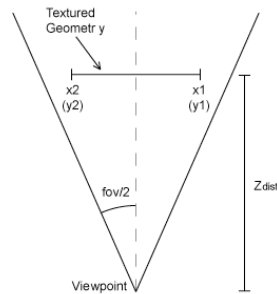


- <http://www.wiley.com/legacy/compbooks/vrml2sbk/cover/cover.htm>
- <http://www.web3d.org/x3d/content/examples/Vrml2.0Sourcebook>
- Chapter 02 – Introduction
- Chapter 03 – Shapes
- Chapter 04 – Text

References 4

Pixel Perfect Text by David Frerichs

- Overcome poor pixelation of Text nodes by creating a texture image of the desired text, along with a matching Viewpoint at the right distance
- http://www.frerichs.net/vrml2/pp/pixel_perfect.html



Texture images are covered in Chapter 5: Appearance, Materials and Textures.

Pixel Perfect Text scene available directly at
http://www.frerichs.net/vrml2/pp/pixel_perfect.wrl

Contact

Don Brutzman

brutzman@nps.edu

<http://faculty.nps.edu/brutzman>

Code USW/Br, Naval Postgraduate School
Monterey California 93943-5000 USA
1.831.656.2149 voice

web|**3D**
CONSORTIUM



55

CGEMS, SIGGRAPH, Eurographics

The Computer Graphics Educational Materials Source(CGEMS) site is designed for educators

- to provide a source of refereed high-quality content
- as a service to the Computer Graphics community
- freely available, directly prepared for classroom use
- <http://cgems.inesc.pt>

X3D for Web Authors recognized by CGEMS! ☺

- Book materials: X3D-Edit tool, examples, slidesets
- Received jury award for Best Submission 2008

CGEMS supported by SIGGRAPH, Eurographics



From the CGEMS home page:

- <http://cgems.inesc.pt>

Welcome to CGEMS - Computer Graphics Educational Materials Source. The CGEMS site is designed for educators to provide a source of refereed high-quality content as a service to the Computer Graphics community as a whole. Materials herein are freely available and directly prepared for your classroom.

List of all published modules:

- <http://cgems.inesc.pt/authors/ListModules.aspx>



CGEMS Editorial Policy:

- <http://cgems.inesc.pt/EditorialPolicy.htm>

Creative Commons open-source license

<http://creativecommons.org/licenses/by-nc-sa/3.0>



Attribution-Noncommercial-Share Alike 3.0 Unported

You are free:

- * to Share — to copy, distribute and transmit the work
- * to Remix — to adapt the work

Under the following conditions:

* Attribution. You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Attribute this work: What does "Attribute this work" mean?

The page you came from contained embedded licensing metadata, including how the creator wishes to be attributed for re-use. You can use the HTML here to cite the work. Doing so will also include metadata on your page so that others can find the original work as well.

- * Noncommercial. You may not use this work for commercial purposes.
- * Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.
- * For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.
- * Any of the above conditions can be waived if you get permission from the copyright holder.
- * Nothing in this license impairs or restricts the author's moral rights.

Open-source license for X3D-Edit software and X3D example scenes

<http://www.web3d.org/x3d/content/examples/license.html>

Copyright (c) 1995-2013 held by the author(s). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the names of the Naval Postgraduate School (NPS) Modeling Virtual Environments and Simulation (MOVES) Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

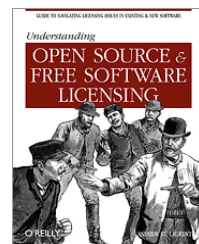
License available at

<http://www.web3d.org/x3d/content/examples/license.txt>

<http://www.web3d.org/x3d/content/examples/license.html>

Good references on open source:

Andrew M. St. Laurent, *Understanding Open Source and Free Software Licensing*, O'Reilly Publishing, Sebastopol California, August 2004. <http://oreilly.com/catalog/9780596005818/index.html>



Herz, J. C., Mark Lucas, John Scott, *Open Technology Development: Roadmap Plan*, Deputy Under Secretary of Defense for Advanced Systems and Concepts, Washington DC, April 2006. <http://handle.dtic.mil/100.2/ADA450769>

