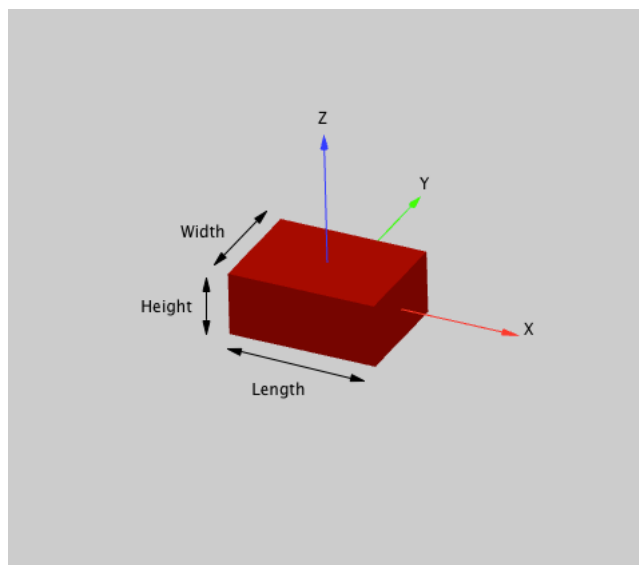


Box Objects

A box object is defined by length along the X direction, width along the Y direction, and height along the Z direction. Boxes are centered on the origin and are always drawn filled. The fill color is determined by the internal color attribute, an embedded color attribute, or a global color attribute, in that order of precedence.

This illustration shows a box with length = 1, width = 0.75, height = 0.5:



To position a box in orientations where the sides are not parallel to the axes or not centered on the origin, add rotate and/or translate operations to the display list before the box item.

Sphere Objects

A sphere is a quadric object, meaning that it can be generated by a quadratic polynomial. Internally a quadric object is composed of a list of triangles and quads that approximate the curved surface.

A newly-created sphere is centered on the origin and both poles are on the Z axis. The numbers of slices (subdivisions around the Z axis; divisions of longitude) and stacks (subdivisions along the Z axis; divisions of latitude) determine the smoothness of the sphere. The greater the numbers of subdivisions, the smoother the sphere's appearance. Small values produce other geometric objects (see illustrations below). The more

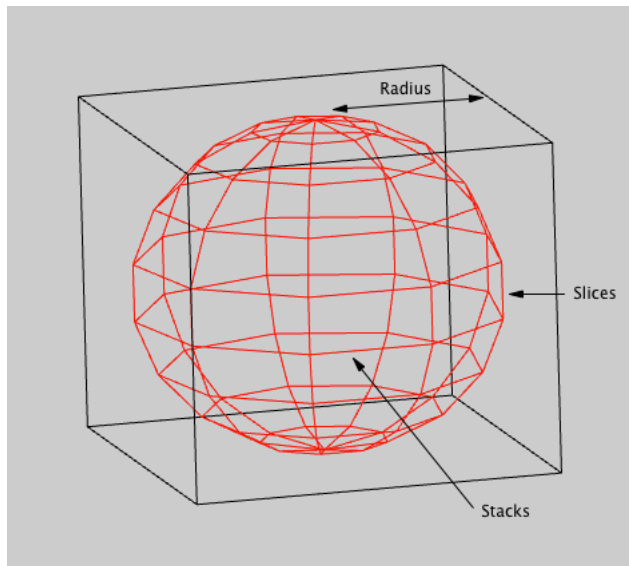
subdivisions you use, the more time it takes to draw the sphere, but in most applications this is not important unless the sphere is replicated many times, such as when used as a marker in a scatter plot.

By default, spheres are initially drawn filled. The fill color is determined by the internal color attribute, an embedded color attribute, or a global color attribute, in that order of precedence.

Use translate and rotate operations to position the sphere in other locations and orientations.

These commands generate a sphere with radius=1, stacks=10, and slices=10 and set its drawing style to lines:

```
AppendToGizmo/D sphere={1,10,10}
ModifyGizmo modifyObject=sphere0, objectType=Sphere,
    property={useGlobalAttributes,0}
ModifyGizmo modifyObject=sphere0, objectType=Sphere,
    property={drawStyle,100011}
```



Create other geometric shapes by using small values for the number of stacks and keeping slices = 2. These commands generate an octahedron with radius=1, stacks=4 and slices=2 and set its drawing style to lines:

```
AppendToGizmo/D sphere={1,4,2}
ModifyGizmo modifyObject=sphere0, objectType=Sphere,
    property={useGlobalAttributes,0}
ModifyGizmo modifyObject=sphere0, objectType=Sphere,
    property={drawStyle,100011}
```