



Геометрические объекты X3D

1. А.В. Аксенов. «Интерактивная компьютерная графика». Учебно-методическое пособие. СПб.: ГУАП. 2020г. 89с.
2. Аксенов А.В. Каталог примеров по X3D.[Электронный ресурс] URL: <https://aksenov.in/guap/x3dom/> (дата обращения 14.09.2023)
3. Официальная X3DOM документация. Fraunhofer [Электронный ресурс] URL: <https://www.x3dom.org/> (дата обращения 14.09.2023)

Примечание. Полный список функций X3D можно посмотреть в описаниях компонентов в разделах 7 – 40 ISO/IEC 19775-1:2013:
<https://www.web3d.org/documents/specifications/19775-1/V3.3/Part01/Architecture.html>

Web 3D Consortium (W3C) - <https://www.web3d.org/standards/version/V3.3>

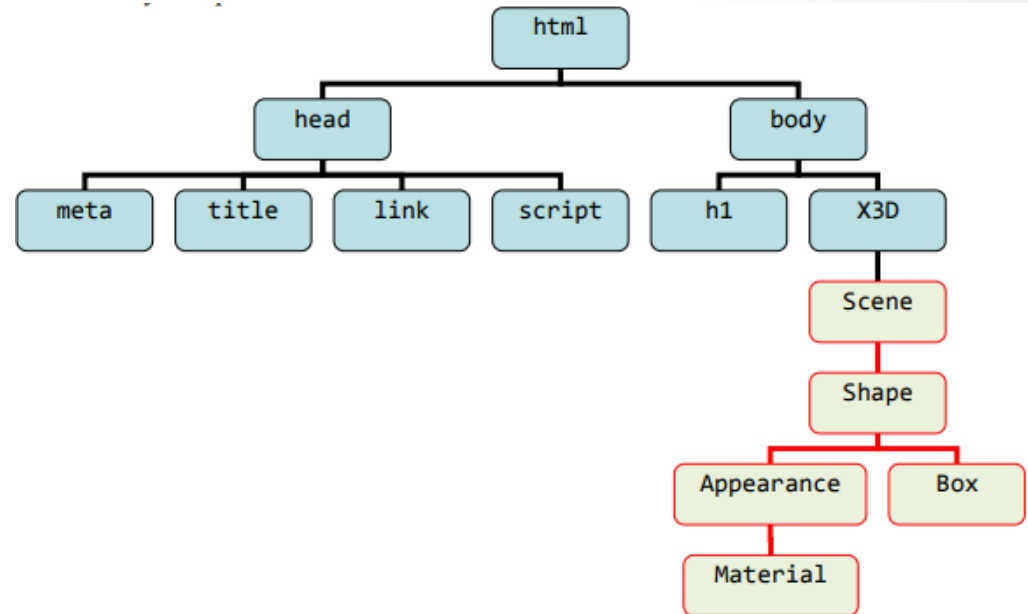
Сцена X3DOM представляет собой встроенный напрямую в код HTML-страницы элемент, в котором размещен XML-код, описывающий граф сцены.

Граф сцены встроен в DOM.

DOM – древовидное представление HTML-документа.

Все узлы документа образуют древовидную иерархическую структуру, при этом вложенные узлы являются дочерними по отношению к узлу, в который они вложены.

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <meta charset="utf-8" />
    <title>Пример 1</title>
    <link rel="stylesheet" href="x3dom.css" />
    <script src="x3dom.js"></script>
  </head>
  <body>
    <h1>Пример 1</h1>
    <X3D>
      <Scene>
        <Shape>
          <Appearance>
            <Material></Material>
          </Appearance>
          <Box size="1 2 3"></Box>
        </Shape>
      </Scene>
    </X3D>
  </body>
</html>
```





Создание сцены X3DOM



При описании каждой HTML-страницы задается код типа документа и ресурса, необходимого для публикации Web: **<!DOCTYPE html> ...</html>**

После описания типа документа следует тег заголовка, в котором нужно указать служебную информацию: **<head>....</head>**.

Например:

```
<meta charset="utf-8" />  
<title> Пример X3DOM </title>  
<link rel="stylesheet" href="https://www.x3dom.org/download/dev/x3dom.css" />  
<script src="https://www.x3dom.org/download/dev/x3dom-full.js"></script>  
<style>  
  x3d { border: 4px solid darkorange; }  
</style>  
</head>
```

Далее необходимо определить основную часть(тело) HTML-страницы: **<body>...</body>**

```
<body>  
  <h1>BlueBox </h1>
```

Затем начинается корневой элемент X3D-сцены: **<X3D>.....</X3D>**

```
<X3D width='800px' height='400px'>
```

После того, как выполнено описание информации для браузера, объявляется сцена: **<Scene>... </Scene>**

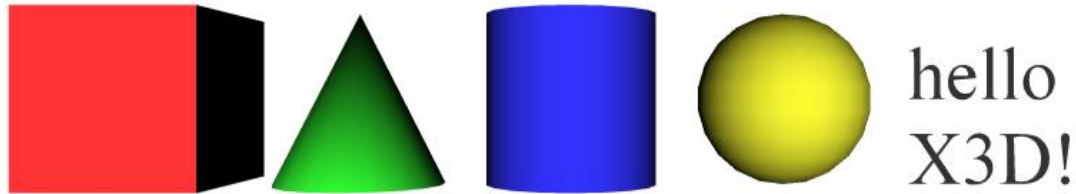
Между тегами Scene располагаются все геометрические объекты, сенсоры, скрипты и все необходимое для получения конечного результата.

```
<Scene>  
  <Shape>  
    <Box size="2 3 4"></Box>  
    <Appearance>  
      <Material diffuseColor='0 0 1' />  
    </Appearance>  
  </Shape>
```





Геометрические примитивы



Box	size (размеры x y z) solid (признак запрета рендеринга обратной стороны)	2 2 2 true
Cylinder	height (высота) radius (радиус основания) top (признак запрета рендеринга верхнего основания) side (признак запрета рендеринга боковой поверхности) bottom (признак запрета рендеринга нижнего основания) solid (признак запрета рендеринга обратной стороны)	2 1 true true true true
Cone	height (высота) bottomRadius (радиус основания) side (признак запрета рендеринга боковой поверхности) bottom (признак запрета рендеринга основания) solid (признак запрета рендеринга обратной стороны)	2 1 true true true
Sphere	radius (радиус) solid (признак запрета рендеринга обратной стороны)	1.5 true



Геометрические примитивы - Text

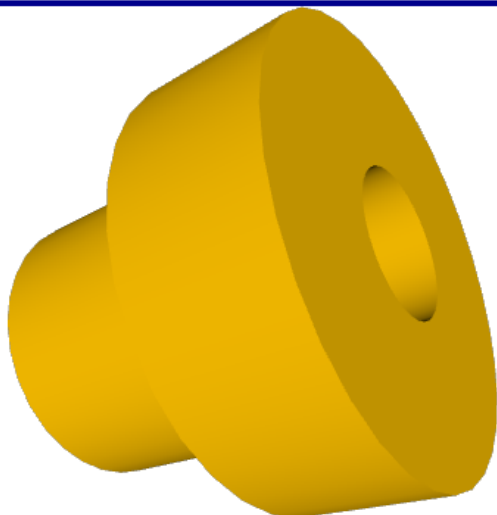
```
Text_X3D.html | Export_Blender_X3D.html | HTMLEventsCSS_3.html | LR3_Ost.html | untitled

<html>
<head>
  <meta charset="utf-8" />
  <title> Текст X3DOM </title>
  <link rel="stylesheet" href="https://www.x3dom.org/download/dev/x3dom.css" />
  <script src="https://www.x3dom.org/download/dev/x3dom-full.js"></script>
  <style>
    x3d {
      border: 4px solid darkblue;
    }
  </style>
</head>
<body>
  <h1> Текст X3DOM!</h1>
  <X3D width='800px' height='400px'>
    <Scene>
      <Shape>
        <Text string='Text 2D' solid='false'>
          <FontStyle family="TYPEWRITER" size='24' spacing='0.1' language='en'>
          </FontStyle>
        </Text>
        <Appearance>
          <Material diffuseColor='1 0.2 .5'></Material>
        </Appearance>
      </Shape>
      <transform translation='-20 -20 10'>
        <Shape>
          <Text string='X3DOM' solid='false'>
            <FontStyle family="TYPEWRITER" size='32' spacing='0.1' language='en'>
            </FontStyle>
          </Text>
          <Appearance>
            <Material diffuseColor='1 0.5 .2'></Material>
          </Appearance>
        </Shape>
      </transform>
    </Scene>
  </X3D>
</body>
</html>
```

Text 2D
X3DOM



Геометрический примитив - Noozle

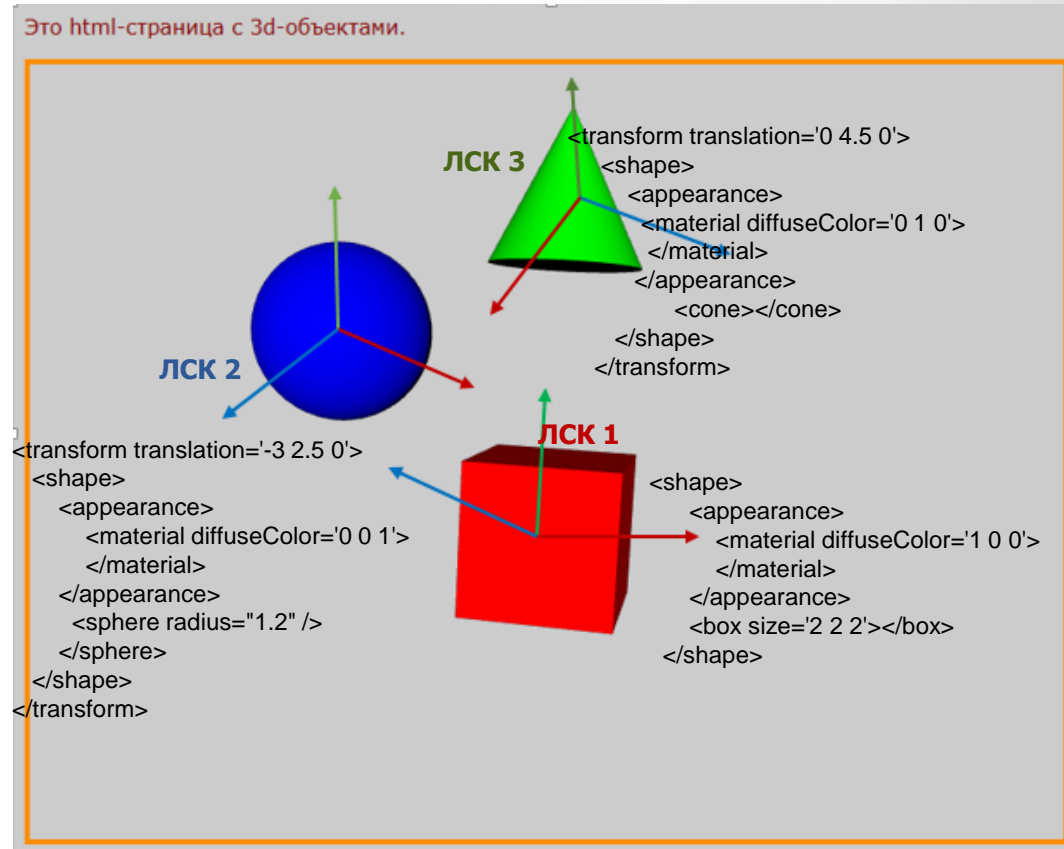


```
1 <html>
2 <head>
3   <meta charset="utf-8" />
4   <title> Конно X3DOM </title>
5   <link rel="stylesheet" href="https://www.x3dom.org/download/dev/x3dom.css" />
6   <script src="https://www.x3dom.org/download/dev/x3dom-full.js"></script>
7   <style>
8     x3d {
9       border: 4px solid darkblue;
10    }
11  </style>
12 </head>
13 <body>
14   <h1> Nozzle X3DOM!</h1>
15   <X3D width='800px' height='400px'>
16     <Scene>
17       <Shape>
18         <Appearance>
19           <Material diffuseColor='darkorange'></Material>
20         </Appearance>
21         <Nozzle outerRadius="2" innerRadius="1" nozzleRadius="3" nozzleHeight="2" height="5">
22         </Nozzle>
23       </Shape>
24     </Scene>
25   </X3D>
26 </body>
27 </html>
```

<X3D> Преобразование объектов в 3D сцене (узел Transform)

Для размещения и группировки объектов применяется узел `<Transform>`, в котором задается информация об изменении положений геометрических объектов, заключенных в него. Transform обладает следующими атрибутами:

- **translation** 'x y z' отвечает за перемещение;
- **rotation** 'x y z angle_radianes' – вращение объекта относительно выбранной оси;
- **scale** 'x y z' – масштабирование.



Узел Transform может иметь вложенные узлы Transform, что создаёт последовательности трансформаций. Порядок, в котором выполняются преобразования, влияет на результат. Поворот после перемещения и перемещение после поворота – это не одно и то же. Внутри одного узла Transform преобразования осуществляются в строго заданном порядке: масштабирование, затем поворот, затем перемещение.

Также для группирования объектов сцены применяется тег `<Group>` `</Group>`, образуя узел группы объектов.

С помощью атрибута DEF можно задавать уникальное имя для объекта сцены и впоследствии использовать этот объект повторно с помощью указания заданного имени в качестве значения атрибута USE узла того же типа. Это сокращает код и вычислительные затраты, т.к. объект создается единожды. Помимо этого, осмысленные DEF имена позволяют легче ориентироваться в коде сцены. Имя, задаваемое с помощью DEF, должно быть цифробуквенным и начинаться с - буквы. Допускается только латиница. Имена регистрозависимы.

Пример:

```
<<Scene>  
  <Transform DEF="LeftCube" translation="-2 0 0">  
    <Shape DEF="MyCube">  
      <Appearance>  
        <Material diffuseColor="1 0 0"></Material>  
      </Appearance>  
      <Box></Box>  
    </Shape>  
  </Transform>  
  <Transform DEF="RightCube" translation="2 0 0">  
    <Shape USE="MyCube"></Shape>  
  </Transform>  
</Scene>
```


<X3D> Группирующие узлы относятся Group, Anchor, Transform

Группирующие узлы способны включать в себя в качестве дочерних, наборы других узлов сцены. При этом все дочерние элементы объединяются в единую группу. Группирование позволяет достичь следующих целей:

- структурирование отдельных элементов сцены логическим образом;
- объединение связанных элементов для упрощения работы с ними;
- поддержание общей для группы системы координат, в пределах которой объекты могут быть легко позиционированы и ориентированы друг относительно друга;
- упрощение тиражирования групп связанных объектов.
- Группирующие узлы могут быть вложены друг в друга:

```
<Group DEF="Gr1">
```

```
...
```

```
<Transform DEF="Tr">
```

```
...
```

```
<Group DEF="Gr2">
```

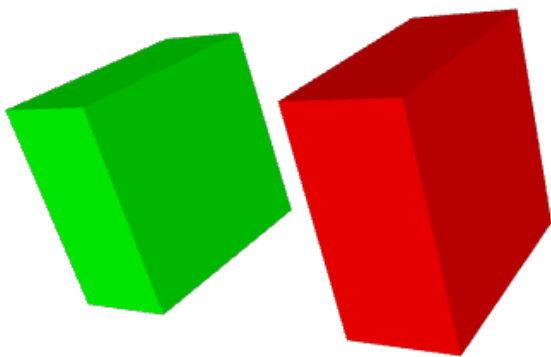
```
...
```

```
</Group>
```

```
</Transform>
```

```
</Group>
```

Клонирование объектов X3DOM!



```
<html>
<head>
  <meta charset="utf-8" />
  <title> DEF/USE X3DOM </title>
  <link rel="stylesheet" href="https://www.x3dom.org/download/dev/x3dom.css" />
  <script src="https://www.x3dom.org/download/dev/x3dom-full.js"></script>
  <style>
    x3d {
      border: 4px solid darkolivegreen;
    }
  </style>
</head>
<body>
  <h1> Клонирование объектов X3DOM!</h1>
  <X3D width='800px' height='400px'>
    <Scene>
      <Anchor url="http://guap.ru" parameter="target='_self'">
        <Shape>
          <Box DEF="Cube" size='1 2 2'>
            </Box>
            <Appearance>
              <Material diffuseColor='1 0 0' ></Material>
            </Appearance>
          </Shape>
        </Anchor>
        <Transform translation="3 0 0">
          <Shape>
            <Appearance>
              <Material diffuseColor='0 1 0' ></Material>
            </Appearance>
            <Box USE="Cube">
              </Box>
            </Shape>
          </Scene>
        </X3D>
      </body>
    </html>
```

Anchor - гиперссылки

К объектам сцены или их группам можно привязать гиперссылки, которые активируются при нажатии на объект.

Для этого служит узел Anchor. Он группирует дочерние узлы и связывает с ними гиперссылку. Описание полей:

parameter – параметры перехода.

Например, значение `parameter="target='_self'"` указывает, что переход осуществляется в текущей вкладке (по умолчанию открывается новая).

url – URL, на который осуществляется переход.

В примере у Anchor два дочерних узла, переход на сайт осуществляется при нажатии на любой из них.

<Scene>

<Anchor url="http://guap.ru" parameter="target='_self'">

<Shape DEF="Cube">

<Box>

</Box>

<Appearance>

<Material diffuseColor='1 0 0'></Material>

</Appearance>

</Shape>

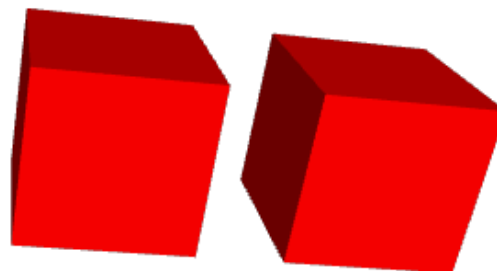
<Transform translation="3 0 0">

<Shape USE="Cube">

</Shape>

</Anchor>

● </Scene>



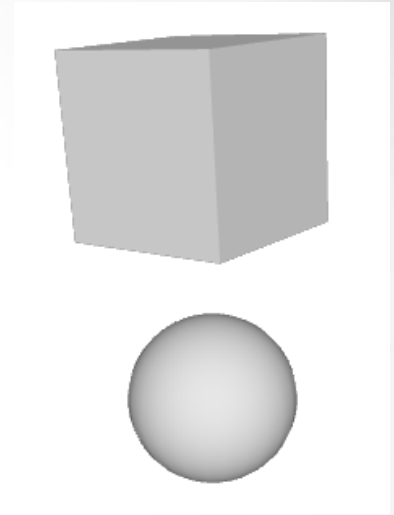
Inline - встраивание

В сцену X3DOM можно встроить внешний X3D-файл. Объекты из встроеной сцены появятся в основной, как если бы они были описаны в текущем файле. Для этого используется узел Inline. Описание полей:

url – набор путей к файлу сцены. Путь может быть несколько, в этом случае достигается большая надежность загрузки файла, поскольку по некоторым из них он может быть недоступен.

load – загружать ли файл сцены автоматически (true/false). Атрибут может быть выставлен в true динамически по какому-то событию, тогда подгрузка произойдет в этот момент, а не при начальной загрузке сцены.

```
<Scene>  
  <Shape>  
    <Sphere></Sphere>  
    <Appearance>  
      <Material></Material>  
    </Appearance>  
  </Shape>  
  <Transform translation="0 3 0">  
    <Inline url="'cube.x3d','http://mysite.com/x3d/cude.x3d'">  
  </Transform>  
</Scene>
```

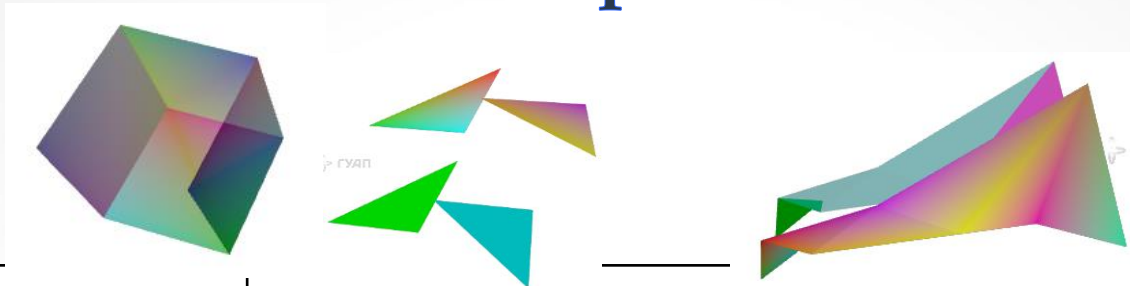


Для приведенного примера файл **cube.x3d** может иметь вид:

```
<Scene>  
  <Shape>  
    <Box></Box>  
    <Appearance>  
      <Material></Material>  
    </Appearance>  
  </Shape>  
</Scene>
```

<https://aksenov.in/guap/x3dom/examples/3/inline.html>

Сложные геометрические объекты



IndexedFaceSet	<p>coordIndex (порядок применения индексов координат, для отделения индексов одного полигона используется «-1»)</p> <p>solid (признак запрета рендеринга обратной стороны, по умолчанию имеет значение true)</p> <p>colorIndex (порядок применения цветов)</p>
IndexedLineSet	<p>coordIndex (порядок применения индексов координат, для отделения индексов каждой полилинии используется «-1»)</p> <p>colorIndex (порядок применения цветов)</p>
IndexedTriangleSet	<p>index – перечисление троек индексов координат из массива дочернего узла Coordinate. Каждая тройка индексов образует новый треугольник. Разделитель (-1) не требуется. Данный индекс также индексирует цвета из массива дочернего узла Color (или ColorRGBA)</p>
IndexedTriangleStripSet	<p>index – перечисление индексов координат из массива дочернего узла Coordinate. Когда формирование полосы закончено, ставится разделитель (-1), после чего можно определить новую полосу. Данный индекс также индексирует цвета из массива дочернего узла Color (или ColorRGBA)</p>

Для использования сложных объектов необходимо объявить внутри конструкции тег <Coordinate>, который описывает обход координат вершин полигонов.

<X3D> Список сложных геометрических объектов X3D

В X3D определены следующие сложные геометрические узлы:

- **PointSet** - создает набор несвязанных между собой точек.
- **IndexedLineSet** - позволяет определить набор ломаных линий определённого цвета в пространстве.
- **IndexedFaceSet** - служит для создания сложных геометрических фигур, состоящих из набора полигонов.
- **ElevationGrid** - служит для создания поверхностей из набора четырехугольников, определяя сетку возвышенностей над горизонтальной плоскостью.
- **IndexedTriangleSet** - служит для создания сложных геометрических фигур, состоящих из треугольников.
- **IndexedTriangleStripSet** - служит эффективным способом создания объектов, состоящих из полосы треугольников.
- **Extrusion** - геометрический узел, который позволяет перемещать 2D-поперечное сечение вдоль линии пути в локальной системе координат. Масштабирование и повороты поперечных сечений могут создавать различные сложные 3D-формы геометрических объектов.

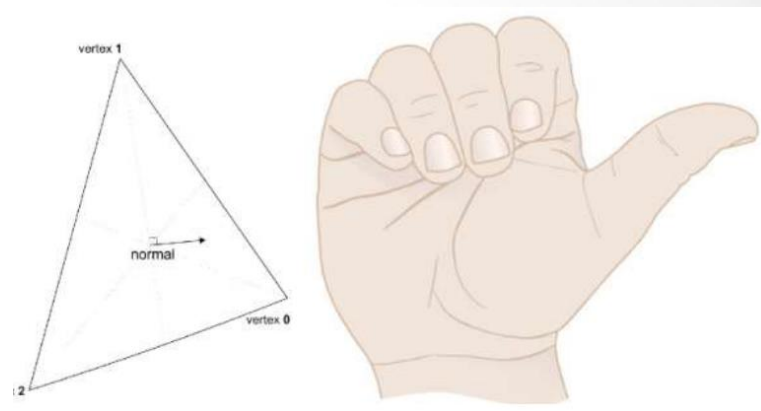
<X3D> Общие поля сложных геометрических узлов X3D

Нумерация элементов массивов сложных геометрических узлов X3D начинается с 0.

ccw (counterclockwise, против часовой стрелки) определяет порядок описания вертексов для полигонов. По умолчанию имеет значение true, что соответствует правилу правой руки.

Если четыре пальца правой руки указывают порядок описания вертексов, то отогнутый большой палец указывает направление вектора нормали

лицевой стороны данного полигона.



Значение поля **colorPerVertex**. определяет, применяются ли цвета к вертексам (true) или к полигонам/линиям (false). В первом случае для интервалов между вертексами осуществляется линейная (в случае закрашивания полигона – билинейная) интерполяция цвета, в массиве **colorIndex** количество цветов должно соответствовать количеству вертексов. Во втором случае цвета не интерполируются и применяются к полигонам/линиям, в массиве **colorIndex** количество цветов должно соответствовать количеству полигонов/линий.



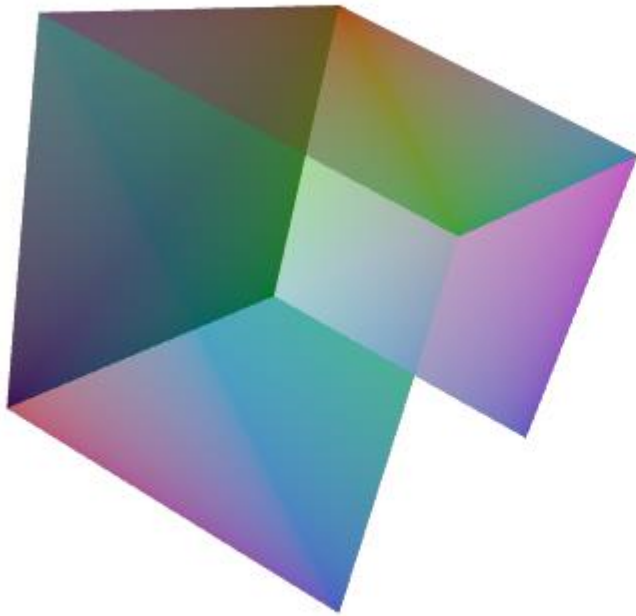
IndexedFaceSet colorPerVertex true, false

IndexedLineSet colorPerVertex true, false



Геометрический объект - IndexedFaceSet

IndexedFaceSet X3DOM!



```
<html>
<head>
  <meta charset="utf-8" />
  <title> IndexedFaceSet X3DOM </title>
  <link rel="stylesheet" href="https://www.x3dom.org/download/dev/x3dom.css" />
  <script src="https://www.x3dom.org/download/dev/x3dom-full.js"></script>
  <style>
    x3d {
      border: 4px solid darkblue;
    }
  </style>
</head>
<body>
  <h1> IndexedFaceSet X3DOM!</h1>
  <X3D width='800px' height='400px'>
    <Scene>
      <Shape>
        <Appearance>
          <Material transparency="0.3"></Material>
        </appearance>
        <IndexedFaceSet ccw="true" colorPerVertex="true" convex="true"
          creaseAngle="0" solid="false"
          coordIndex="0 4 7 3 -1 5 6 2 1 -1 4 5 6 7 -1 6 7 3 2 -1"
          colorIndex="0 1 2 3 -1 4 5 6 0 -1 1 2 3 4 -1 5 6 0 1 -1">
          <Coordinate point="-2 0 2, 2 0 2, 2 0 -2, -2 0 -2,
            -2 4 2, 2 4 2, 2 4 -2, -2 4 -2"></Coordinate>
          <Color color="0 0 1, 0 1 0, 0 1 1, 1 0 0, 1 0 1, 1 1 0, 1 1 1">
          </Color>
        </IndexedFaceSet>
      </Shape>
    </Scene>
  </X3D>
</body>
</html>
```




ElevationGrid – Карта высот

Служит для создания поверхностей из набора четырехугольников, определяя сетку возвышенностей над горизонтальной плоскостью.

```
ElevationGrid : X3DGeometryNode {  
  MFNode [in,out] attrib [] [X3DVertexAttributeNode]  
  SFNode [in,out] color NULL [X3DColorNode]  
  SFNode [in,out] fogCoord NULL [FogCoordinate]  
  SFNode [in,out] normal NULL [X3DNormalNode]  
  SFNode [in,out] texCoord NULL [X3DTextureCoordinateNode]  
  SFBool [] ccw true  
  SFBool [] colorPerVertex true  
  SFFloat [] creaseAngle 0 [0,∞)  
  MFFloat [] height [] (-∞,∞)  
  SFBool [] normalPerVertex true  
  SFBool [] solid true  
  SFInt32 [] xDimension 0 [0,∞)  
  SFFloat [] xSpacing 1.0 (0,∞)  
  SFInt32 [] zDimension 0 [0,∞)  
  SFFloat [] zSpacing 1.0 (0,∞)  
}
```

Описание полей:

xDimension - | - эти переменные определяют размерность сетки возвышенностей.

zDimension- Соответственно массив height должно содержать

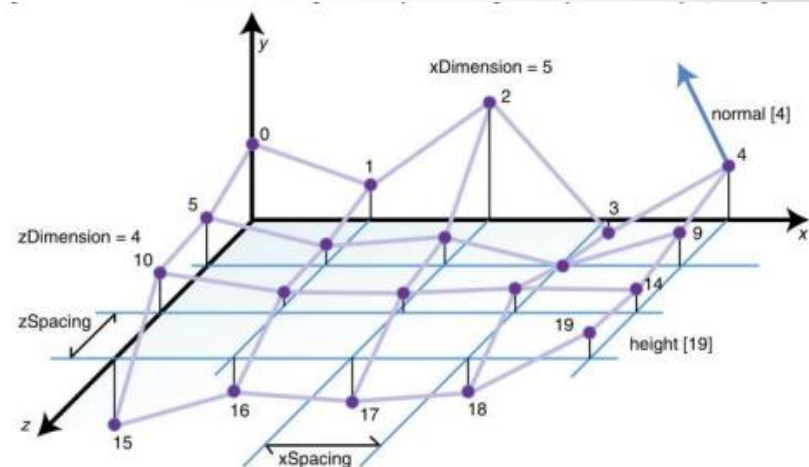
xDimension*zDimension | элементов

xSpacing- | -здесь определяется расстояние между соседними вершинами в ме

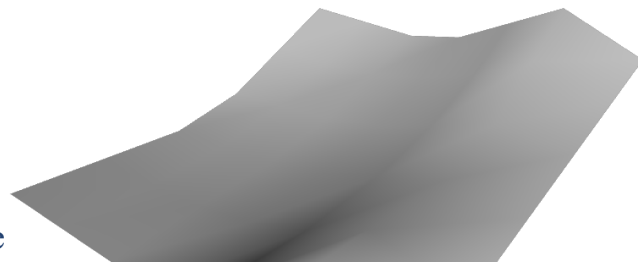
zSpacing- | соответственно по осям X и Z

height - определяет массив возвышенностей в метрах над горизонтальной плоскостью

Y=0. Если смотреть на поверхность сверху в направлении оси -Z, то вершины перечисляются от левого верхнего угла к правому нижнему (построчно)

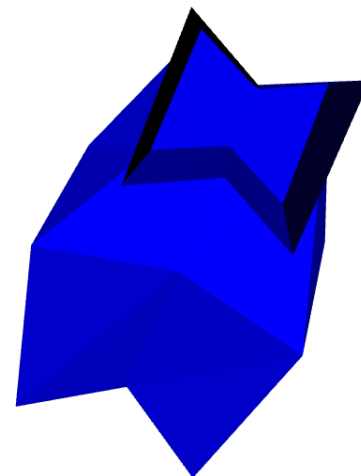
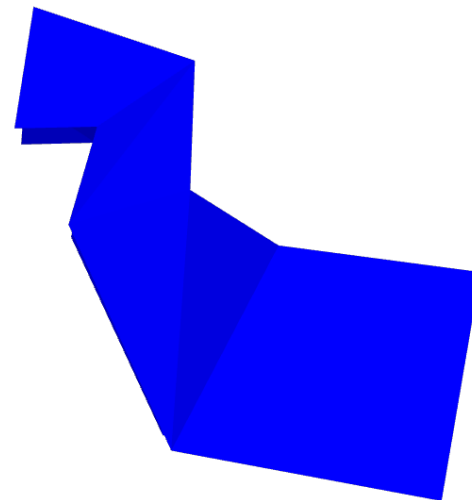


```
<ElevationGrid xDimension="7" zDimension="6"  
  height="1.5, 10, 0.5, 0.5, 1, 1.5, 0,  
    1, 0.5, 0.25, 0.25, 0.5, 1, 0,  
    0.5, 0.25, 0, 0, 0.25, 0.5, 0,  
    0.5, 0.25, 0, 0, 0.25, 0.5, 0,  
    1, 0.5, 0.25, 0.25, 0.5, 1, 0,  
    1.5, 1, 0.5, 0.5, 1, 1.5, 0"  
  xSpacing="5.0" zSpacing="5.0" solid="false">  
</ElevationGrid>
```



Extrusion

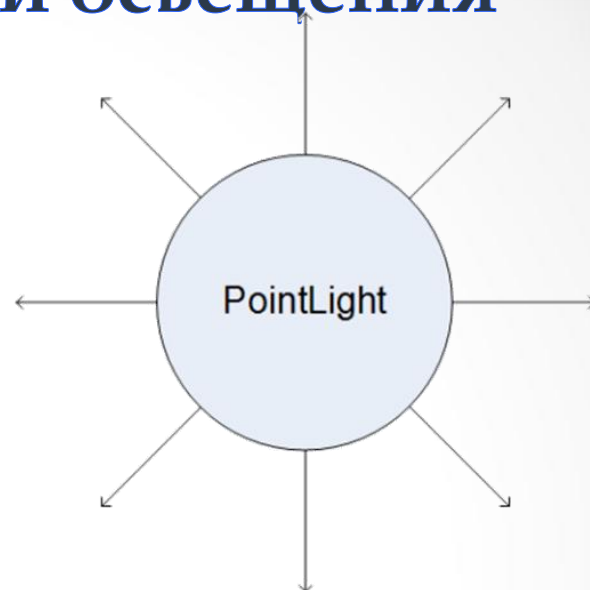
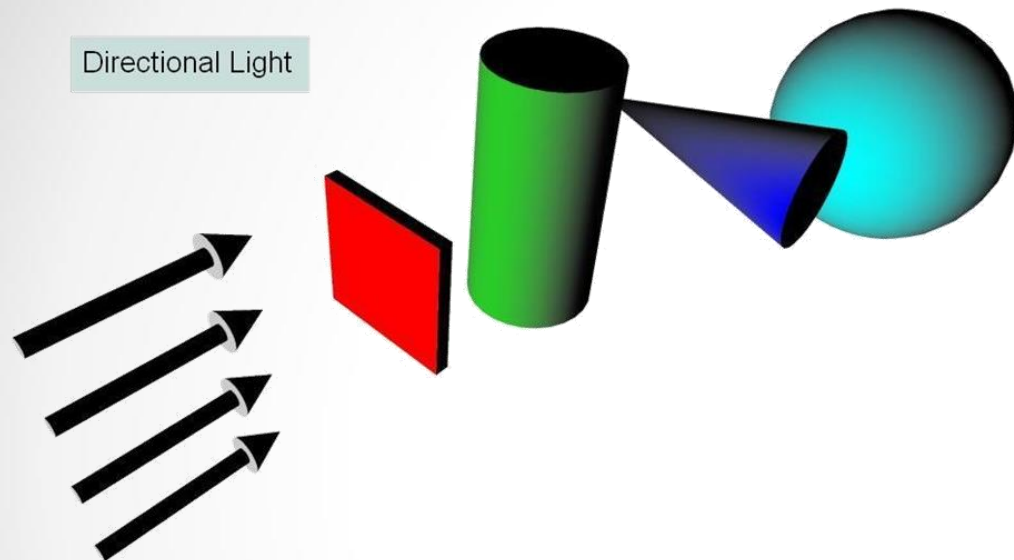
```
<html>
<head>
  <meta charset="utf-8" />
  <title> Экструзия X3DOM </title>
  <link rel="stylesheet" href="https://www.x3dom.org/download/dev/x3dom.css" />
  <script src="https://www.x3dom.org/download/dev/x3dom-full.js"></script>
  <style>
    x3d {
      border: 4px solid darkblue;
    }
  </style>
</head>
<body>
  <h1>Extrusion</h1>
  <X3D id="x3d" xmlns="https://www.x3dom.org/x3dom">
    <Scene>
      <Shape>
        <Appearance>
          <Material diffuseColor='blue'></Material>
        </Appearance>
        <Extrusion beginCap='false' endCap='false' solid='false'
          crossSection='[(1, 1), (0.5, 0), (1, -1), (-1, -1), (-0.5, 0), (-1, 1), (1, 1)]'
          scale='[1 1, 1 0.8, 0.7 1, 0.5 0.5]'
          spine='[0 -1 0, 0 1 0, 1 2 0, 2 2 0, 2 4 0]'\>
        </Extrusion>
      </Shape>
    </Scene>
  </X3D>
</body>
</html>
```



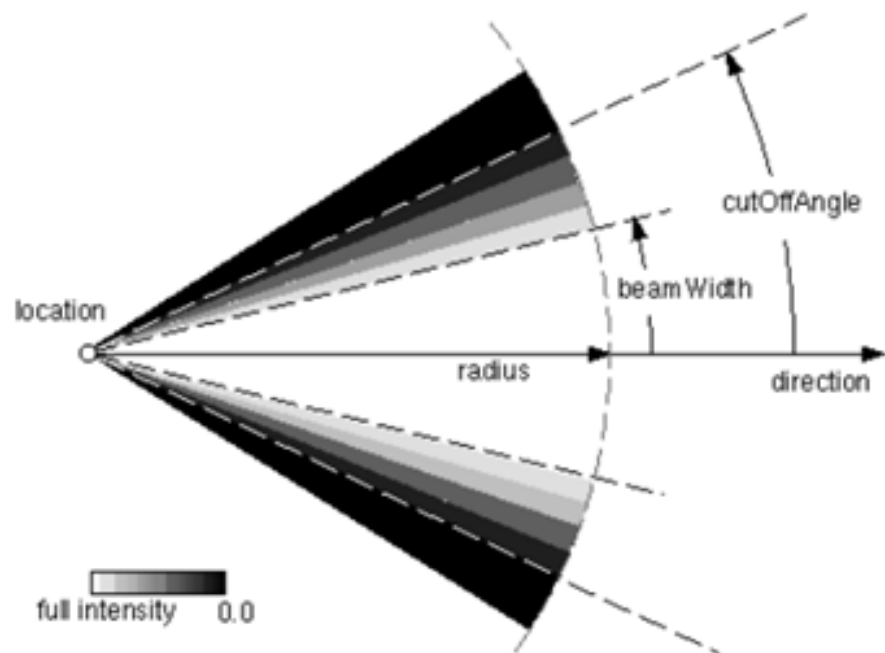
Extrusion - геометрический узел, который позволяет перемещать 2D-поперечное сечение вдоль линии пути в локальной системе координат. Масштабирование и повороты поперечных сечений могут создавать различные сложные 3D-формы геометрических объектов.

Камеры и Источники освещения

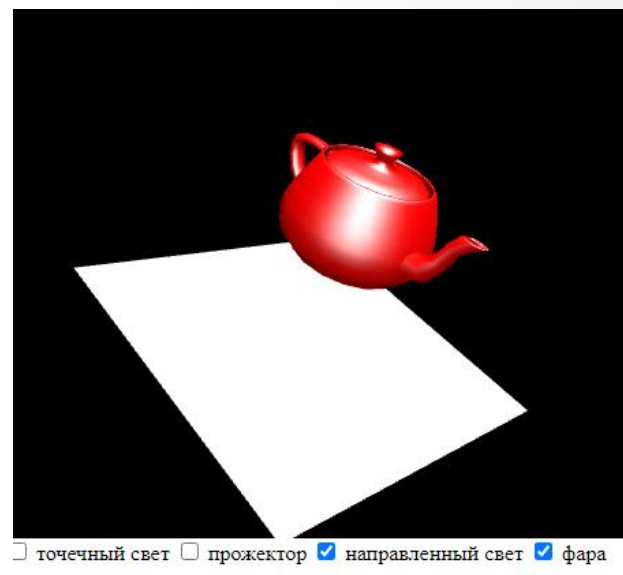
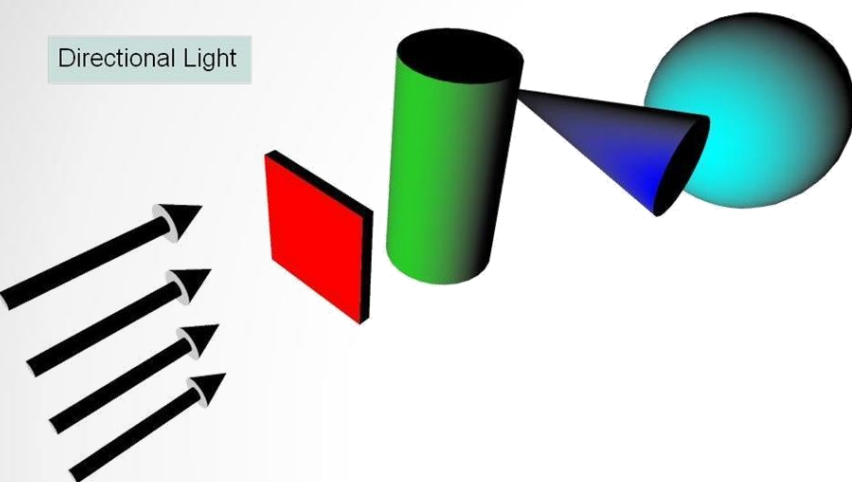
Directional Light



- камеры (Viewpoint)
- источники освещения:
 - направленный свет (DirectionLight)
 - точечный свет(PointLight)
 - прожекторный свет(SpotLight)



Источники освещения



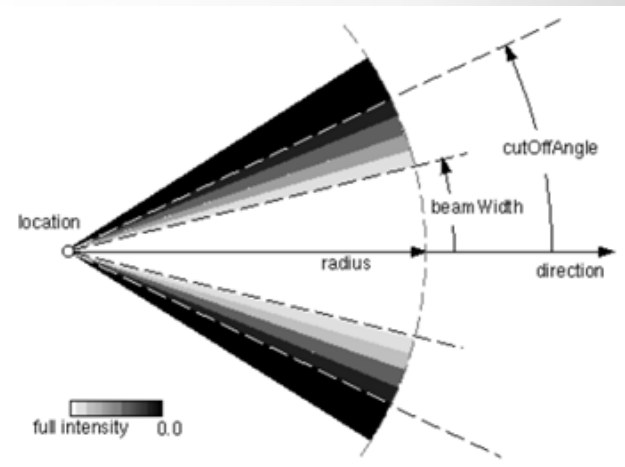
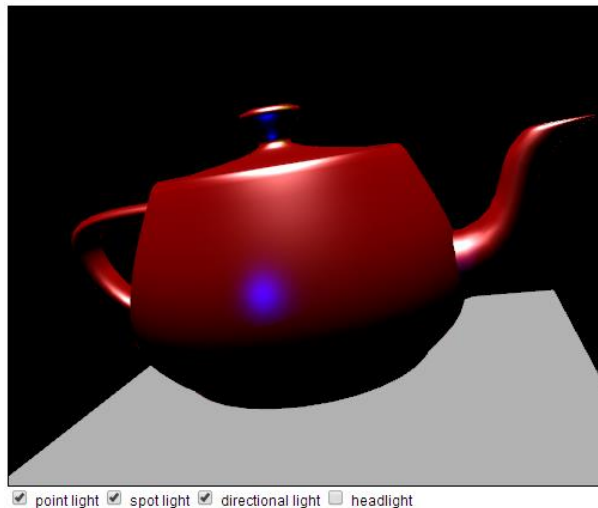
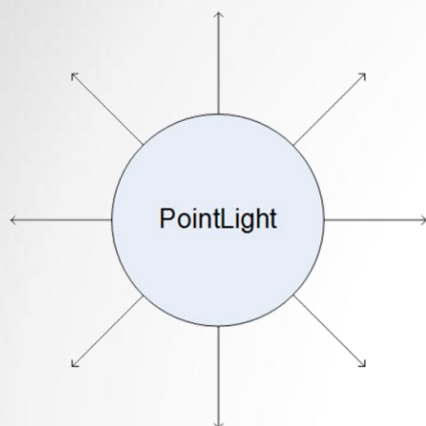
<https://doc.x3dom.org/tutorials/lighting/lights/example.html>

источники освещения:

- направленный свет (DirectionalLight)

```
<DirectionalLight id="directional" direction='0 0 1' on ="TRUE" intensity='1.0' shadowIntensity='0.0'>  
</DirectionalLight>
```

Источники освещения



ИСТОЧНИКИ ОСВЕЩЕНИЯ:

- точечный свет(PointLight)

```
<PointLight id='point' on='TRUE' intensity='0.9000' color='0.0 0.6 0.0' location='0 10 0.5 '
radius='5.0000' > </PointLight>
```

- прожекторный свет(SpotLight)

```
<SpotLight id='spot' on ="TRUE" beamWidth='0.9' color='0 0 1' cutOffAngle='0.78'
location='0 0 12' radius='22' > </SpotLight>
```



Создание сцены X3D (CAD Component Editor > scene.json)

https://examples.x3dom.org/editor/component_editor/

Сервисы Google Mail.Ru: почта, пои... Результат поиска... видео-уроки по 3D... Разработка комп... переводчик - Пои... Как войти в свой г... <https://www.youtub...> Списание для чтения

New Load Store UI Style: lightness x3dom

Scene Explorer

- Scene
 - Nozzle(id:21)
 - Dish(id:24)
 - SlopedCylinder(id:27)
 - Snout(id:30)

Focus Clone Group Delete

Element Catalogue

- Box Cone Cylinder
- Circular Torus Rectangular Torus Dish
- Snout Pyramid Slope-Bottom Cylinder
- Nozzle Extrusion Solid of Revolution
- Origin Reference Point

Top Free FR TP LT BK BT RT

Object Transformation

Position: 0 0 0

Rotation: 0 0 0

Editor Configuration

- Show Grid ☒
- Show Axes ☒
- Show Axis Labels ☒
- Show Height Cues ☒
- Show Connection Point Labels ☐
- Show Snap Points/Vectors ☒
- Show Reference Frames ☒
- Show Debug Stats ☐
- Show Debug Log ☐

Reference Unit: meters

Snap to grid ☒

Turntable navigation ☒

Scene Properties

Grid Cell Size: 0.2 m

Grid Size (Cells): 26 x 26

Scene Size: 5.20 x 5.20 m

Compute Grid Dynamically ☐

Adjust Grid ...

Element Properties

Align ...

Height 1

RInside 0.4

ROutside 0.5

Nozzle Height 0.1

Nozzle Radius 0.6

Positive Element ☒

scene is not saved

no reference points added

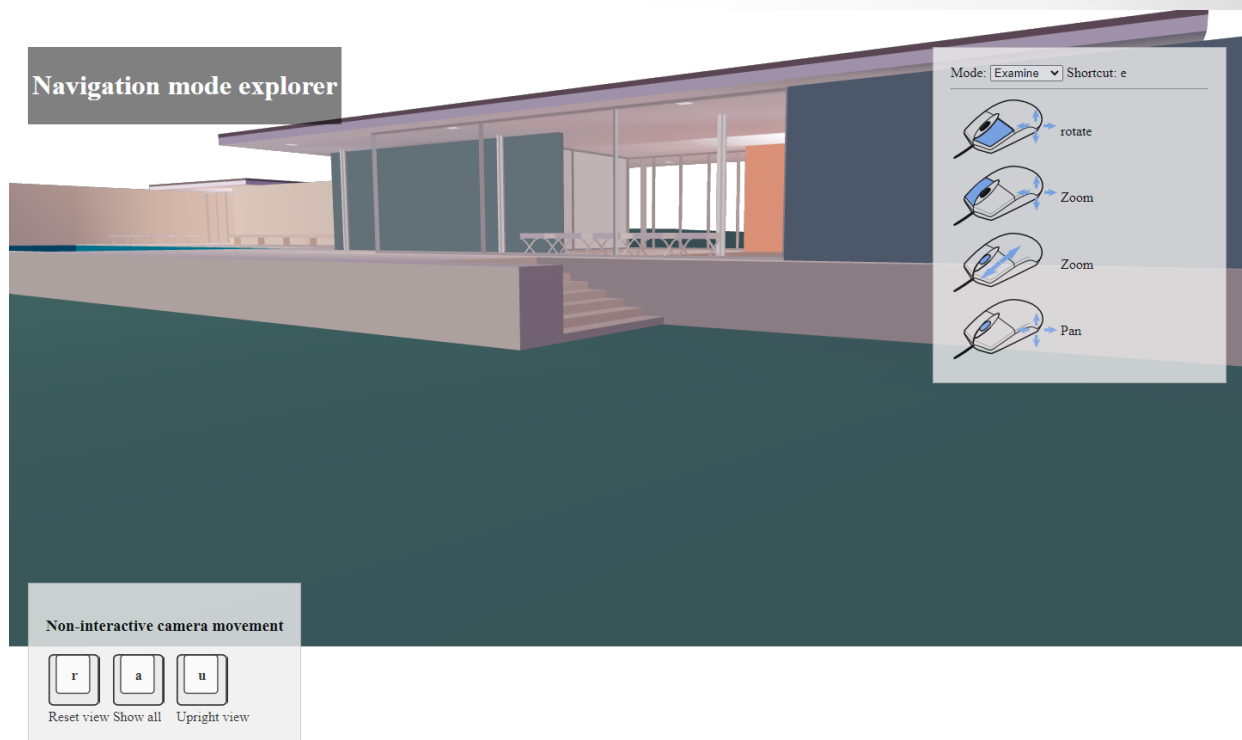
no origin added



Навигация в X3D-сцене

<https://doc.x3dom.org/tutorials/animationInteraction/navigation/example.html>

В настоящее время
X3DOM
поддерживает
следующие
интерактивные
режимы навигации:
Examine, Walk, Fly,
Look-at, Look-around,
Turntable, Game and
Helicopter.



Синтаксис узла NavigationInfo:

```
<x3d width='500px' height='400px'>
```

```
<scene>
```

```
  <navigationInfo type="walk" "any" id="navType"></navigationInfo>
```

```
</scene>
```

```
</x3d>
```

Тип может быть одним режимом или также содержать массив, включающий дополнительные резервные режимы, если основной режим недоступен.

Можно изменить значение, установив новый тип для этого узла.

```
document.getElementById('navType').setAttribute("type", currentMode);
```