

glTF-Tutorials

[Previous: Morph Targets](#)[Table of Contents](#)[Next: Skins](#)

A Simple Skin

glTF supports *vertex skinning*, which allows the geometry (vertices) of a mesh to be deformed based on the pose of a skeleton. This is essential in order to give animated geometry, for example of virtual characters, a realistic appearance. The core for the definition of vertex skinning in a glTF asset is the `skin`, but vertex skinning in general implies several interdependencies between the elements of a glTF asset that have been presented so far.

The following is a glTF asset that shows basic vertex skinning for a simple geometry. The elements of this asset will be summarized quickly in this section, referring to the previous sections where appropriate, and pointing out the new elements that have been added for the vertex skinning functionality. The details and background information for vertex skinning will be given in the next section.

```
{
  "scene" : 0,
  "scenes" : [ {
    "nodes" : [ 0, 1 ]
  }],
  "nodes" : [ {
    "skin" : 0,
    "mesh" : 0
  }, {
    "children" : [ 2 ]
  }, {
    "translation" : [ 0.0, 1.0, 0.0 ],
    "rotation" : [ 0.0, 0.0, 0.0, 1.0 ]
  }],
  "meshes" : [ {
    "primitives" : [ {
      "attributes" : {
        "POSITION" : 1,
        "JOINTS_0" : 2,
        "WEIGHTS_0" : 3
      },
      "indices" : 0
    } ]
  }],
}
```

```
"skins" : [ {
    "inverseBindMatrices" : 4,
    "joints" : [ 1, 2 ]
} ],


"animations" : [ {
    "channels" : [ {
        "sampler" : 0,
        "target" : {
            "node" : 2,
            "path" : "rotation"
        }
    } ],
    "samplers" : [ {
        "input" : 5,
        "interpolation" : "LINEAR",
        "output" : 6
    } ]
} ],


"buffers" : [ {
    "uri" : "data:application/gltf-buffer;base64,AAABAAMAAAADAAIAAgADAAUUAgAFAAQABAAFAAcAE",
    "byteLength" : 168
}, {
    "uri" : "data:application/gltf-buffer;base64,AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA/",
    "byteLength" : 320
}, {
    "uri" : "data:application/gltf-buffer;base64,AACAPwAAAAAAAAD8AAIA/AADAPwAAAEEAACBAAABAQAAA",
    "byteLength" : 128
}, {
    "uri" : "data:application/gltf-buffer;base64,AAAAAAAAD8AAIA/AADAPwAAAEEAACBAAABAQAAA",
    "byteLength" : 240
} ],


"bufferViews" : [ {
    "buffer" : 0,
    "byteLength" : 48,
    "target" : 34963
}, {
    "buffer" : 0,
    "byteOffset" : 48,
    "byteLength" : 120,
    "target" : 34962
}, {
    "buffer" : 1,
    "byteLength" : 320,
    "byteStride" : 16
}, {
    "buffer" : 2,
    "byteLength" : 128
}, {
    "buffer" : 3,
```

```
"byteLength" : 240
} ],

"accessors" : [ {
  "bufferView" : 0,
  "componentType" : 5123,
  "count" : 24,
  "type" : "SCALAR"
}, {
  "bufferView" : 1,
  "componentType" : 5126,
  "count" : 10,
  "type" : "VEC3",
  "max" : [ 0.5, 2.0, 0.0 ],
  "min" : [ -0.5, 0.0, 0.0 ]
}, {
  "bufferView" : 2,
  "componentType" : 5123,
  "count" : 10,
  "type" : "VEC4"
}, {
  "bufferView" : 2,
  "byteOffset" : 160,
  "componentType" : 5126,
  "count" : 10,
  "type" : "VEC4"
}, {
  "bufferView" : 3,
  "componentType" : 5126,
  "count" : 2,
  "type" : "MAT4"
}, {
  "bufferView" : 4,
  "componentType" : 5126,
  "count" : 12,
  "type" : "SCALAR",
  "max" : [ 5.5 ],
  "min" : [ 0.0 ]
}, {
  "bufferView" : 4,
  "byteOffset" : 48,
  "componentType" : 5126,
  "count" : 12,
  "type" : "VEC4",
  "max" : [ 0.0, 0.0, 0.707, 1.0 ],
  "min" : [ 0.0, 0.0, -0.707, 0.707 ]
} ],

"asset" : {
  "version" : "2.0"
```

```
}
```

The result of rendering this asset is shown in Image 19a.

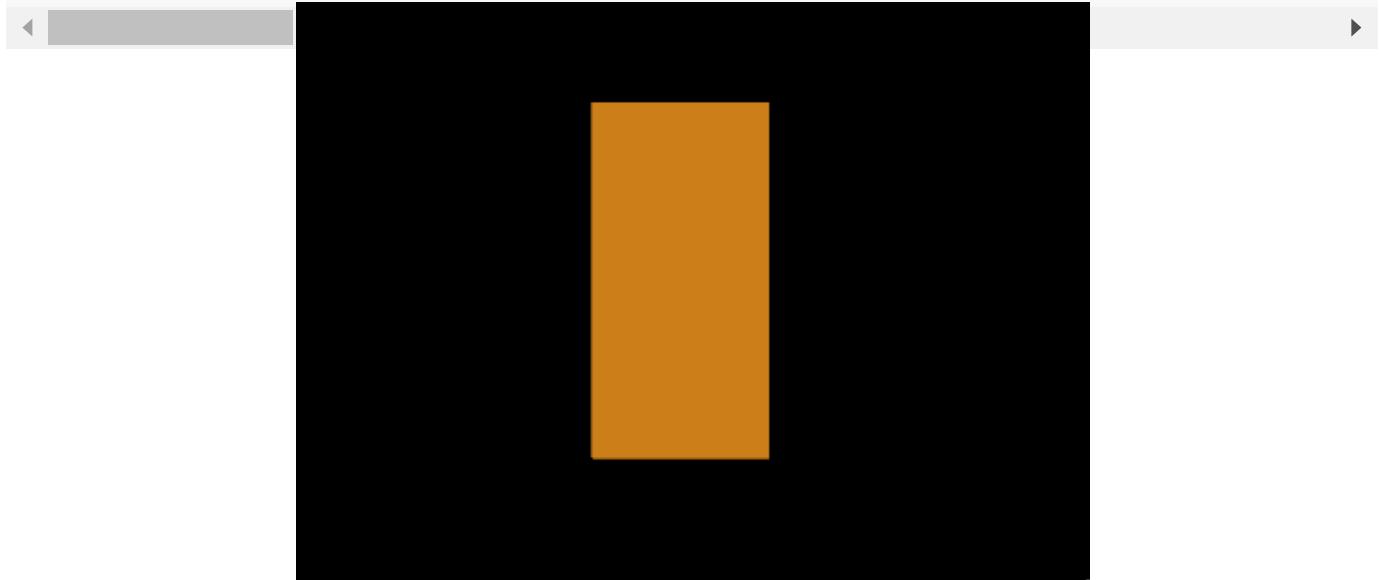


Image 19a: A scene with simple vertex skinning.

Elements of the simple skin example

The elements of the given example are briefly summarized here:

- The `scenes` and `nodes` elements have been explained in the [Scenes and Nodes](#) section. For the vertex skinning, new nodes have been added: the nodes at index 1 and 2 define a new node hierarchy for the *skeleton*. These nodes can be considered the joints between the “bones” that will eventually cause the deformation of the mesh.
- The new top-level dictionary `skins` contains a single skin in the given example. The properties of this skin object will be explained later.
- The concepts of `animations` has been explained in the [Animations](#) section. In the given example, the animation refers to the *skeleton* nodes so that the effect of the vertex skinning is actually visible during the animation.
- The [Meshes](#) section already explained the contents of the `meshes` and `mesh.primitive` objects. In this example, new mesh primitive attributes have been added, which are required for vertex skinning, namely the `"JOINTS_0"` and `"WEIGHTS_0"` attributes.
- There are several new `buffers`, `bufferViews`, and `accessors`. Their basic properties have been described in the [Buffers, BufferViews, and Accessors](#) section. In the given example, they contain the additional data required for vertex skinning.

Details about how these elements are interconnected to achieve the vertex skinning will be explained in the [Skins](#) section.

[Previous: Morph Targets](#)[Table of Contents](#)[Next: Skins](#)

This site is open source. [Improve this page.](#)

