X3D Volume Rendering Component

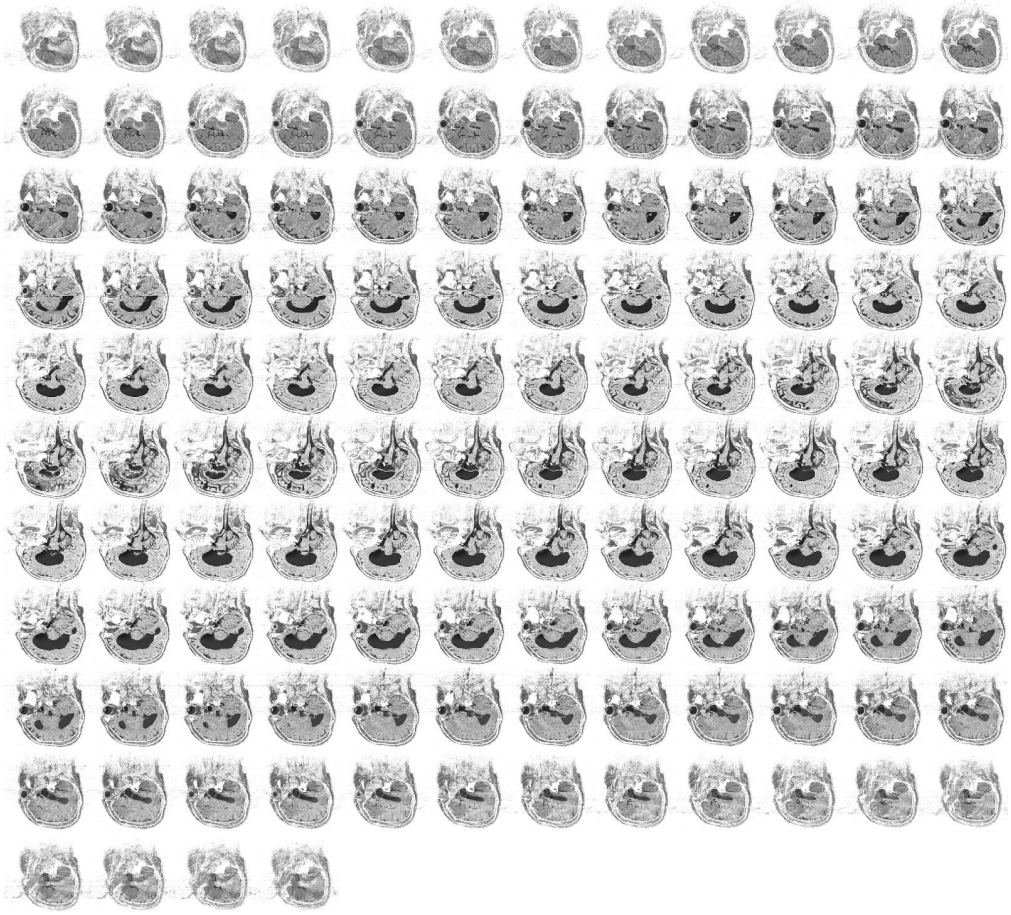
Extension Proposals

New extension proposals

1. ImageTextureAtlas node
2. MPRVolumeStyle node
3. allowViewpointInside attribute

ImageTextureAtlas node

- A data structure for volume data which is compatible with WebGL 1.0
- Easily transferable between client-server as an Image (PNG, JPEG, etc.)
- Can be composed in the browser using 2D <canvas>
- It could be used as fallback solution for browsers/devices which lack support of WebGL 2.0



ImageTextureAtlas node

```
ImageTextureAtlas : X3DTexture2DNode
      SFNode [in,out] metadata NULL [X3DMetadataObject]
      SFBool [in,out] repeatS TRUE
      SFBool [in,out] repeatT TRUE
      SFNode [in,out] textureProperties NULL [TextureProperties]
      MFString [in,out] url [] [URI]
      SFInt32 [in,out] numberOfSlices 0 [0,∞)
      SFInt32 [in,out] slicesOverX 0 [0,∞)
      SFInt32 [in,out] slicesOverY 0 [0,∞)
      SFBool [in,out] hideChildren TRUE
      SFString [in,out] channels "R"
      SFString [in,out] sortOrder "NONE" ["ROW", "CHANNEL", "NONE"]
}
```

MPRVolumeStyle node

- Conceptually:
  - Define an arbitrary plane through the volume data
  - Reconstruct the data at the given plane
- Usually the volume data is stored as image slices in a given direction, the idea is to allow to visualize the data in other direction.
- Although this is usually seen as not necessary real-time. The idea of this node is to use re-use the 3D rendering of the volume to also offer this plane

MPRVolumeStyle node

```
MPRVolumeStyle : X3DVolumeRenderStyleNode  {
     SFBool [in,out] enabled TRUE
     SFNode [in,out] metadata NULL [X3DMetadataObject]
     SFNode [in,out] transferFunction NULL [X3DTexture2DNode, X3DTexture3DNode]
     SFBool [in,out] forceOpaque TRUE
     MFNode [in,out] planes NULL [MPRPlane]
}

MPRPlane : X3DNode {
     SFBool [in,out] enabled TRUE
     SFNode [in,out] metadata NULL [X3DMetadataObject]
     SFVec3 [in,out] normal 0 0 1
     SFFloat [in,out] position 0.0 [0,1]
}
```

MPRVolumeStyle node

```
<VolumeData id="volume" dimensions="4.0 4.0 4.0">
        <ImageTextureAtlas id="atlas" containerField="voxels" url="" numberOfSlices="96" slicesOverX="10" slicesOverY="10">
        </ImageTextureAtlas>
        <MPRVolumeStyle forceOpaque="true">
                <MPRPlane id="planeA" normal="0.0 1.0 0.0" position="0.5"></MPRPlane>
                <MPRPlane id="planeB" normal="1.0 0.0 0.0" position="0.5"></MPRPlane>
                <MPRPlane id="planeC" normal="0.0 0.0 1.0" position="0.5"></MPRPlane>
        </MPRVolumeStyle>
</VolumeData>
```

Demo: https://sunrise-lunge.glitch.me/

allowViewPointInside attribute

- Define the behaviour of the rendering algorithm when the camera is located inside the volume data
  - Render the data when the camera is inside the volume bounding box

- **Reason**: If inside exploration is not needed some computation can be avoided

allowViewPointInside attribute

```
X3DVolumeDataNode : X3DChildNode, X3DBoundedObject {
    SFVec3f [in,out] dimensions 1 1 1 (0,∞)
    SFNode [in,out] metadata NULL [X3DMetadataObject]
    SFVec3f [] bboxCenter  0 0 0 (-∞,∞)
    SFVec3f [] bboxSize -1 -1 -1 [0,∞) or -1 -1 -1
    SFBool [in,out] allowViewPointInside TRUE
}
```

Demo: https://mountain-verse.glitch.me