

Übung 9 – Lösungsvorschlag

Prof. Dr. Arjan Kuijper

Max von Buelow, M.Sc., Volker Knauthe, M.Sc.

Darya Nikitina, B.Sc. Alexander Stichling, Kai Li

Aufgabe 9.1: Rendering

Nennen Sie vier Informationen, welche benötigt werden, um eine Szene in 3D zu rendern. Nennen Sie für jede Information zusätzlich ein Beispiel, welches nicht in der Vorlesung genannt wurde. (1P)

Aufgabe 9.1: Rendering

Antwort: Vier der folgenden Informationen können genannt werden.

- Objekt Geometrie (z.B. die Form eines Schrankes)
- Transformationen (z.B. die Positionierung des Schrankes in einem Raum)
- Materialien (z.B. die Farbe und die Textur eines Apfels)
- Kameras (vordefinierte Ansichten z.B. die Vogelperspektive)
- Lichter (Verschiedene Arten von Lichtquellen, Farben z.B. gelbe und grüne Scheinwerfer in einem Raum)
- Spezial-Effekte (z.B. Nebel oder Schatten innerhalb eines Raumes)

Bewertung: 0.5 Punkte für 4 richtige Informationen und 0.5 Punkte für 4 Beispiele (es können auch andere Beispiele genannt werden)

Aufgabe 9.2: Szenengraphstruktur

- a) Nennen Sie vier Eigenschaften, welcher ein Szenengraph erfüllen muss und erklären sie diese (1P)

Antwort: Ein Szenengraph ist ein **gerichteter, azyklischer Graph**

gerichtet: Jede Kante hat eine Richtung

azyklisch: Es gibt keine Zyklen im Graph

zusätzlich: Szenengraph hat einen Wurzelknoten

Kein Baum: Jeder Knoten kann mehrere Elternknoten haben (Ausnahme: Wurzelknoten)

Bewertung: 0.25 Punkte jeweils für eine Eigenschaft und eine Erklärung

Aufgabe 9.2: Szenengraphstruktur

- b) Nennen Sie zwei Vorteile bei der Verwendung von 3D-Szenengraphen und erklären Sie diese anhand eines Beispiels, das in der Vorlesung nicht vorgestellt wurde. (1P)

Aufgabe 9.2: Szenengraphstruktur

Antwort:

- Wiederverwendbarkeit der Objektdaten, z.B. man erstellt ein Fenster-Objekt für ein Gebäude und benutzt es mehrfach
- Semantische Gruppierung der Objektdaten, z.B. gemeinsames Ein-/und Ausblenden der Fenster
- Transformationshierarchie ermöglicht Transformation von kompletten Gruppen, ohne diese explizit ändern zu müssen, z.B. kann man das Gebäude inklusive der Fenster manipulieren

Bewertung: 0.25 Punkte pro Erklärung und 0.25 Punkte pro Beispiel

Aufgabe 9.3: Szenengraph

Erstellen Sie einen Szenengraphen für das folgende Bild. In dem Szenengraphen sollen **mindestens 3** Gruppierungsknoten verwendet werden. Zudem sollen für **mindestens 2** der Gruppierungsknoten die Transformations- und Objektknoten dargestellt werden. (3P)



Quelle: <https://pixabay.com/photos/heaven-carousel-entertainment-3279551/>

```

graph TD
    Root((G „Kettenkarussell“)) --> S((G „Sitze“))
    Root --> K((G „Ketten“))
    Root --> D((G „Drehkranz“))
    Root --> Ellipsis1[...]
    S --> T1((T))
    S --> T2((T))
    K --> T3((T))
    K --> T4((T))
    D --> Ellipsis2[...]
    D --> Ellipsis3[...]
    D --> Ellipsis4[...]
    T1 --> O1((O))
    T2 --> O1
    T3 --> O2((O))
    T4 --> O2
  
```

„Kettenkarussell“

G

G „Sitze“

G „Ketten“

G „Drehkranz“

[...]

[...]

[...]

[...]

T

T

T

T

O

O

Transformationen der einzelnen Ketten

Geometrie und Textur für eine...

Der gegebene Szenengraph ist nur eine mögliche Darstellung.

Bewertung: 0.5 Punkte pro richtigem Knoten (Transformationsknoten zählen pro Gruppierungsknoten als ein einzelner Knoten)

Aufgabe 9.4: Anwendungsaufgabe X3DOM

Erstellen Sie eine X3DOM-Szene basierend auf den folgenden Anforderungen (wenn nicht anders angegeben, verwenden Sie die Standardgrößen für die Objekte). (4P)

Aufgabe 9.4: Anwendungsaufgabe X3DOM

Bauen Sie das notwendige HTML-Grundgerüst sowie eine Szene auf. Die Szene soll vorerst nur einen roten Zylinder enthalten. Nutzen Sie die DEF-Funktion, um den Zylinder zu definieren. (1P)

VC Übung X3DOM



```
<!--Red cylinder in the middle-->  
<group DEF="RED_CYLINDER">  
  <shape>  
    <appearance>  
      <material diffuseColor='1 0 0'></material>  
    </appearance>  
    <cylinder></cylinder>  
  </shape>  
</group>
```

Aufgabe 9.4: Anwendungsaufgabe X3DOM

Fügen Sie der Szene nun oben auf dem Zylinder eine rote Kugel hinzu.
Die Kugel soll um den Wert 1 auf der y-Achse bewegt werden. (1P)

VC Übung X3DOM



```
<!--Red sphere (head)-->  
<transform translation='0 1 0'>  
  <group DEF='RED_SPHERE'>  
    <shape>  
      <appearance>  
        <material diffuseColor='1 0 0'></material>  
      </appearance>  
      <sphere></sphere>  
    </shape>  
  </group>  
</transform>
```

Beachte: DEF ist hier nicht notwendig

Aufgabe 9.4: Anwendungsaufgabe X3DOM

Nutzen Sie die USE-Funktion, um zwei weitere rote Zylinder der Szene hinzuzufügen. Skalieren Sie beide Zylinder mit dem Wert 0,4. Den ersten Zylinder bewegen Sie um den Wert -0,5 auf der x-Achse und um -1,4 auf der y-Achse. Den zweiten Zylinder bewegen Sie um den Wert 0,5 auf der x-Achse und um -1,4 auf der y-Achse. (1P)

VC Übung X3DOM



```
<!--Two cylinders on the bottom-->
<transform translation="-0.5 -1.4 0">
  <transform scale="0.4 0.4 0.4">
    <group USE="RED_CYLINDER"></group>
  </transform>
</transform>

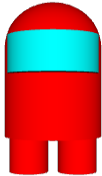
<transform translation="0.5 -1.4 0">
  <transform scale="0.4 0.4 0.4">
    <group USE="RED_CYLINDER"></group>
  </transform>
</transform>
```

Aufgabe 9.4: Anwendungsaufgabe X3DOM

Fügen Sie anschließend einen weiteren, türkisen Zylinder der Szene hinzu. Diesen skalieren Sie für x um den Wert 0,9 und für y um den Wert 0,4. Bewegen Sie diesen anschließend um 0.5 entlang der z-Achse und um den Wert 2 entlang der y-Achse.

(1P)

VC Übung X3DOM



```
<!--Cyan Cylinder (Visor)-->
<transform scale='0.9 0.4 1'>
  <transform translation='0 2 0.5'>
    <group DEF='CYAN_CYLINDER'>
      <shape>
        <appearance>
          <material diffuseColor='0 1 1'></material>
        </appearance>
        <cylinder></cylinder>
      </shape>
    </group>
  </transform>
</transform>
```

Beachte: DEF ist hier nicht notwendig

NB: In der Aufgabe hieß es

„Fügen Sie anschließend einen weiteren, türkisen Zylinder der Szene hinzu. Diesen bewegen Sie um 0.5 entlang der z-Achse und um den Wert 2 entlang der y-Achse. Skalieren Sie diesen anschließend für x um den Wert 0,9 und für y um den Wert 0,4. (1P) “

Das ergibt (langweilig!):

