
VRML 1

Martin Klíma



Computer Graphics Group



Charakteristika VR systémů

- 3D prostor modelovaný v paměti počítače
- Interakce a navigace v reálném čase
 - vyžaduje množství ústupků kvůli urychlení
- Multimediální obsah
 - audio, video, obrázky, text
 - odkazy
- Interakce modelu s uživatelem
- Animace

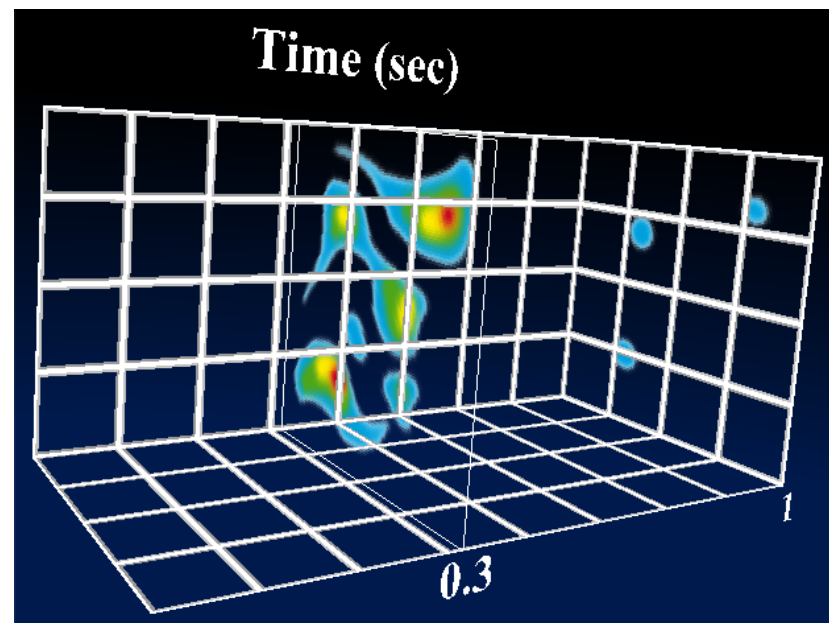
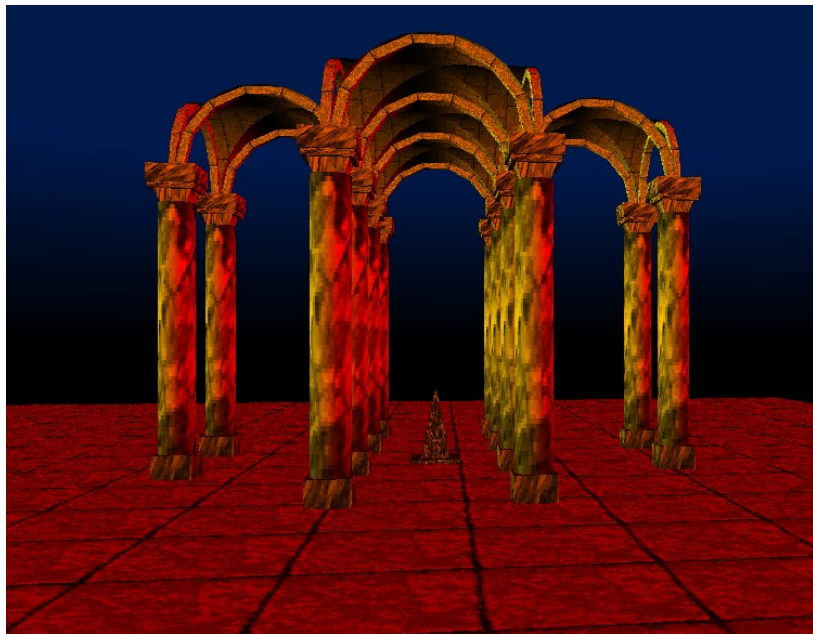


Co to je VRML?

- Virtual Reality Modeling Language
 - jazyk pro modelování virtuálních světů
 - speciálně určen pro použití na webu
 - textově založený
 - soubory s příponou .wrl
 - prohlížeče jako rozšíření běžných prohlížečů
 - Cortona
 - ve virtuálním 3D světě je 3D model a interakce objektů



Ukázky



Prohlížeče a podpora

- Většina 3D modelovacích nástrojů umožňuje export do VRML
- Volně dostupné (zdarma) prohlížeče
- Vazba na WWW
- Plug-in do IE, FF,...
 - Cortona, Blacksun, Cosmoplayer,
 - Podpora v Javě: XJ3D, loader



Chození
aplikuje se
gravitace

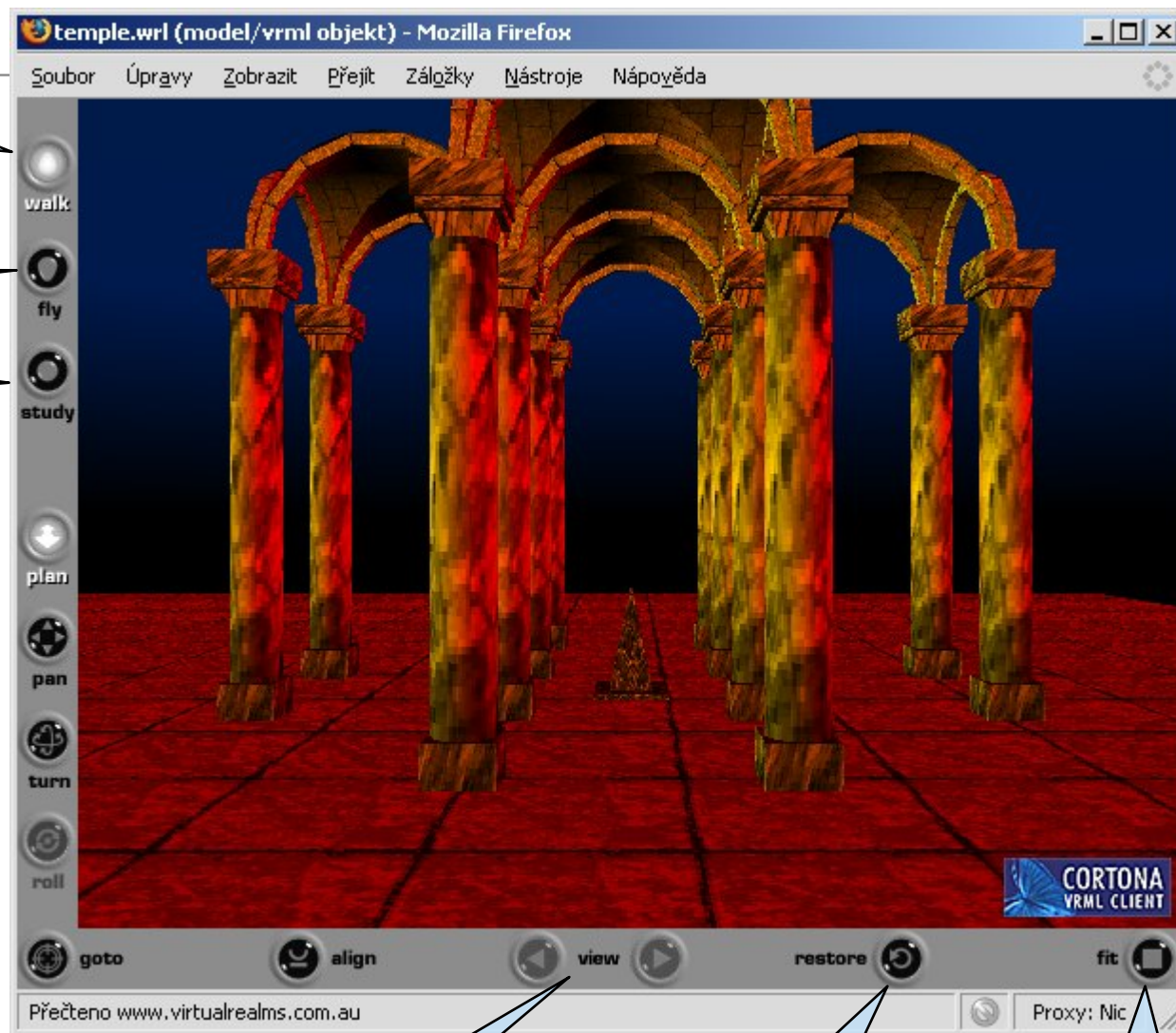
Létání

Prohlížení modelu
ze všech stran

Vhodné pro
chodicí režim

Posun avatara

Otáčení avatara



Přepínání kamer

Znovunastavení
pohledu kamery

Pohled na celý
model

Historie

- VRML 1.0, formát odvozen od Open Inventor formátu vyvinutého firmou Silicon Graphics Inc. (SGI)
- VRML 1.1 specifikace rozšíření, kterou se nepovedlo implementovat ☹ zrušeno
- VRML 2.0 (podzim 1996) mnoho vylepšení a rozšíření
 - animace, interakce, zvuky, mlha, pozadí, jazyková rozšíření
- VRML 97 velmi malá modifikace VRML 2.0 pro potřebu standardizace ISO

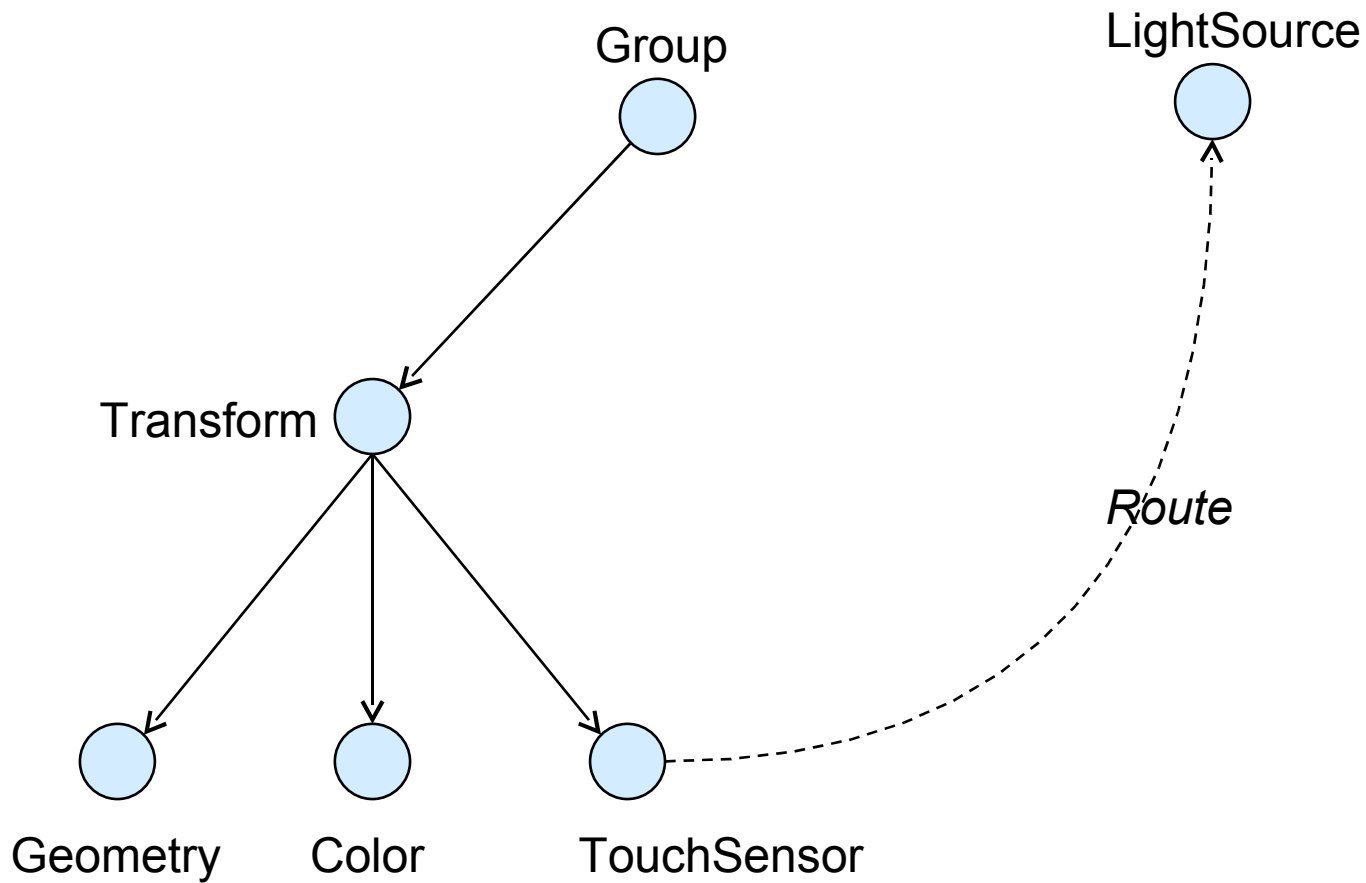


Základní pojmy

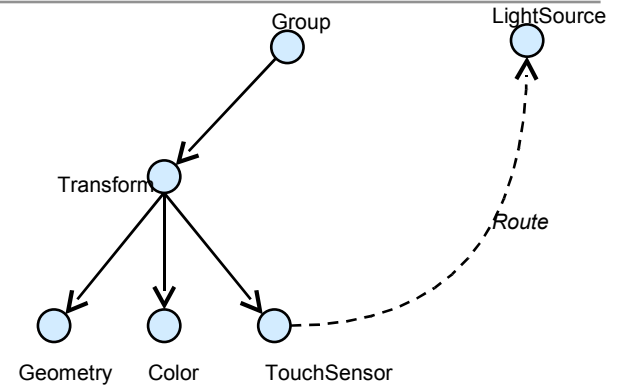
- Avatar
 - virtuální postava reprezentující uživatele ve virtuálním 3D světě
 - má definované rozměry
 - 3D svět vidíme očima svého avatara
- Scene graph
 - datová struktura reprezentující 3D svět
 - hierarchická struktura
 - obsahuje uzly různých typů



Scene graph



Scene graph



```
#VRML V2.0 utf8
Group { children [
  Shape { geometry Sphere { } }
  DEF TS TouchSensor { }
]
}
DEF PL PointLight { location 2 4 4
on FALSE }
ROUTE TS.isActive TO PL.on
```

Obecná struktura wrl souboru

```
#VRML V2.0 utf8
```

```
Viewpoint { ...  
}
```

```
Transform { ...  
}
```

```
ROUTE ... TO ...
```

Hlavička

Globální vlastnosti scény

VRML strom

Ošetření událostí

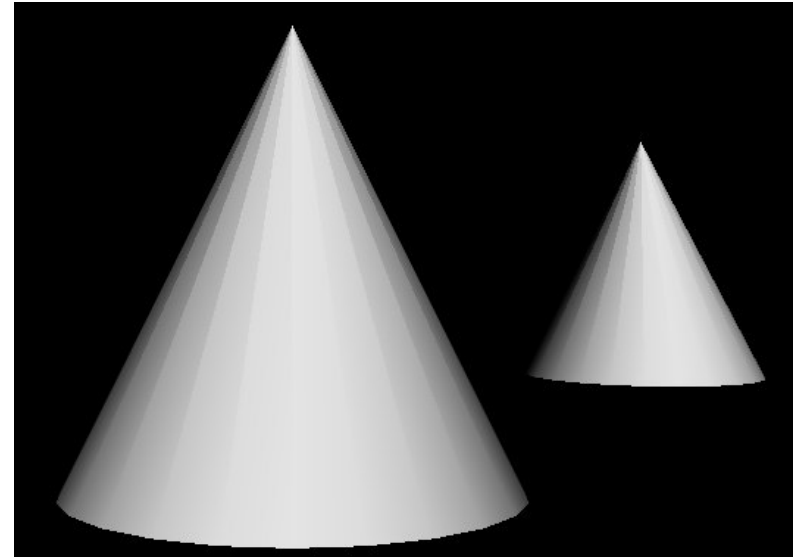
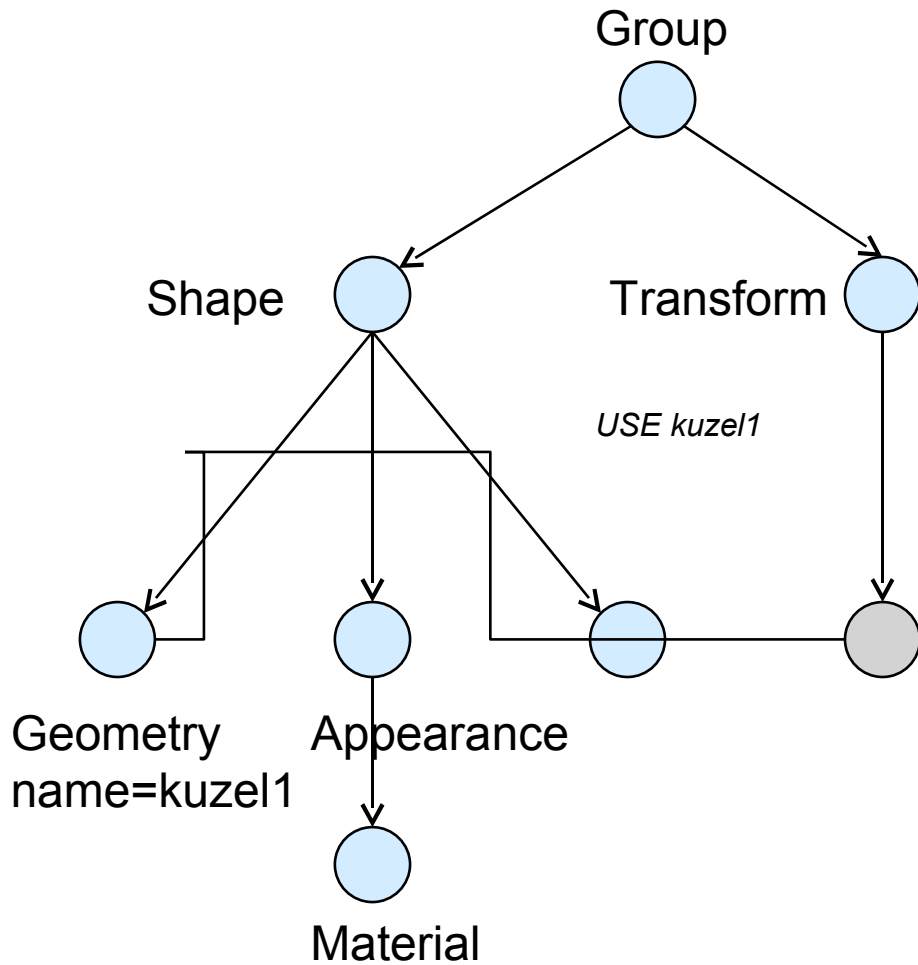


Základní stavební kameny – uzly (Nodes)

- VRML definuje sadu základních uzlů
 - koule, kužel, kvádr, mlha, ...
- Každý uzel má
 - typ
 - 0 až N parametrů
 - může mít jméno



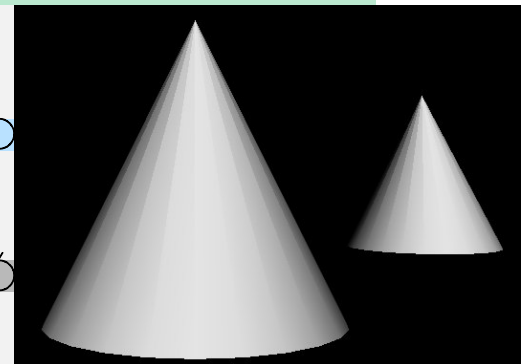
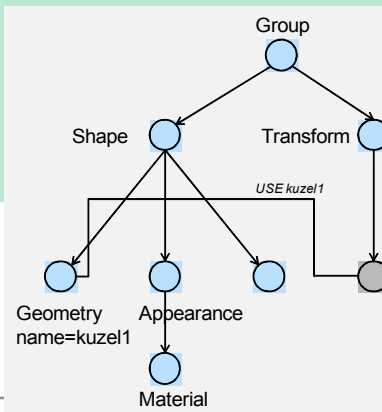
Scene graph



```

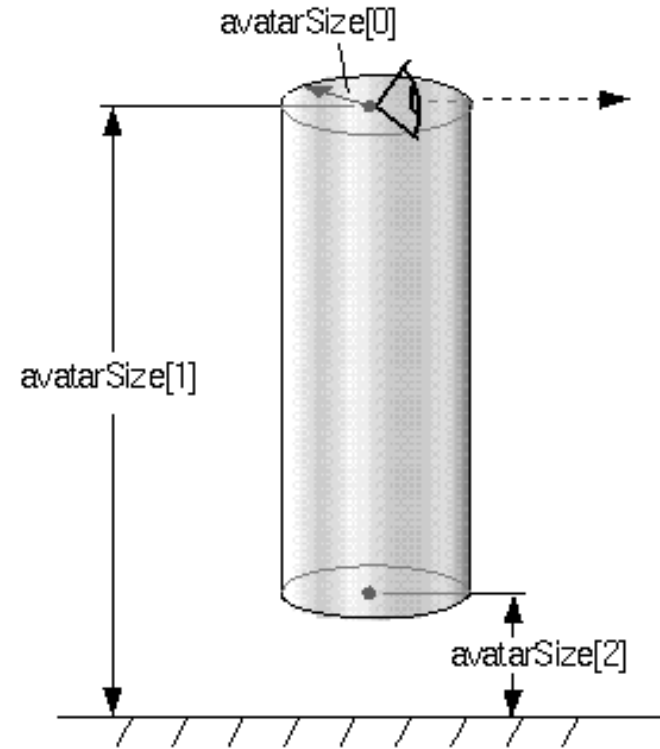
#VRML V2.0 utf8
Group {
  children [
    DEF kuzel1 Shape {
      geometry Cone { bottomRadius 2 height 4 bottom FALSE }
      appearance Appearance {
        material Material { diffuseColor 1 1 1 }
      }
    }
    Transform { scale .5 .5 .5
      translation 3 0 0
      children USE kuzel1 }
  ]
}

```



Vlastnosti světa a avatara

- Uzel WorldInfo
 - title
 - info []
- NavigationInfo
 - avatarSize [0.25, 1.8, 0.75]
 - headlight TRUE
 - speed 1.0
 - type ["WALK", "EXAMINE", "FLY", "ANY", "NONE"]
 - visibilityLimit 0.0



```
#VRML V2.0 utf8
```

```
WorldInfo {
```

```
  title "Dve barevne koule"
```

```
  info [ "pro potreby vyuky",
```

```
    "zdroj: http://www.cs.vu.nl/~eliens/documents/vrml/reference",
```

```
    "uzivejte s opatrne :-)"
```

```
  ]
```

```
}
```

```
NavigationInfo {
```

```
  avatarSize [0.25, 1.8, 0.9]
```

```
  headlight TRUE
```

```
  speed 1.0
```

```
  type ["WALK"]
```

```
  visibilityLimit 10.0
```

```
}
```

```
Group {
```

```
  children [
```

```
    DEF kuzel1 Shape {
```

```
      geometry Cone { bottomRadius 2 height 4 bottom FALSE }
```

```
      appearance Appearance {
```

```
        material Material { emissiveColor 1 0 0 }
```

```
      }
```

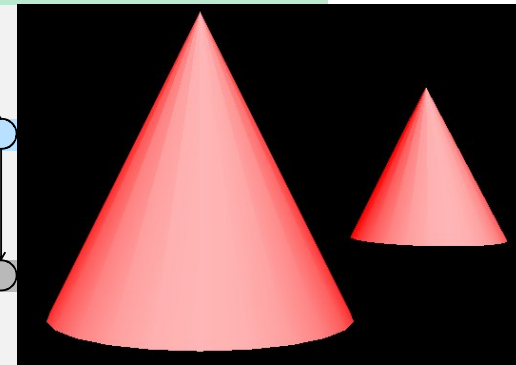
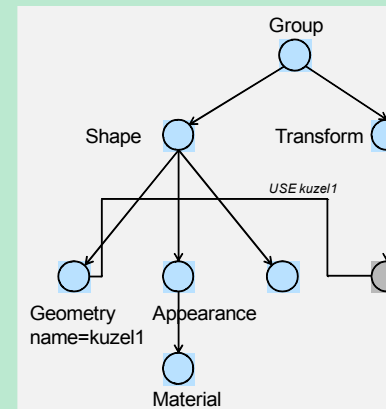
```
    }
```

```
    Transform {
```

```
      scale .5 .5 .5
```

```
      translation 3 0 0
```

```
      children USE kuzel1 }
```



Viewpoint

- Uzel definující pozici kamery
- Může jich být více
 - **fieldOfView** **0.785398** **# (0, π)**
 - **position** **0 0 10**
 - **orientation** **0 0 1 0**
 - **jump** **TRUE**
 - **description** **"muj první viewpoint"**



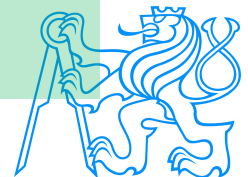
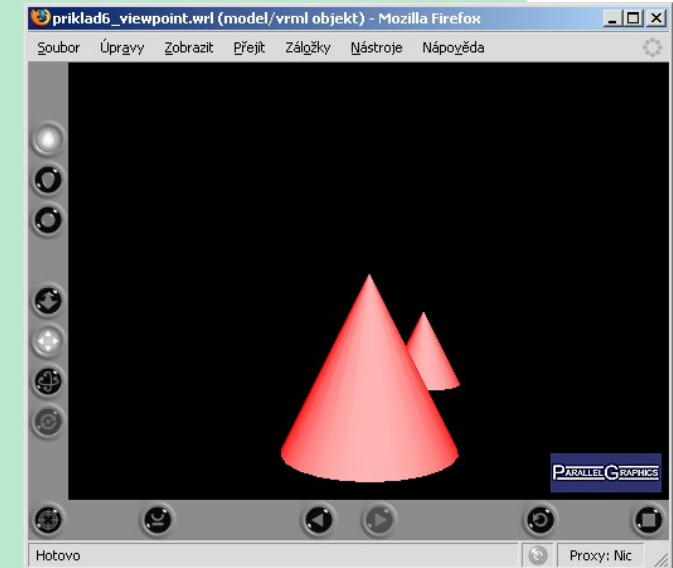
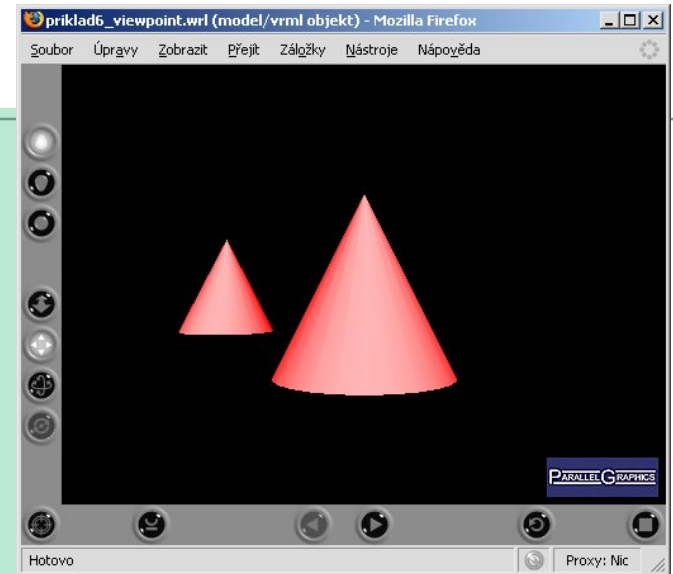
Pohledy

```
#VRML V2.0 utf8
```

```
DEF V1 Viewpoint {  
  fieldOfView 0.76  
  position 0 0 -12  
  orientation 0 1 0 3.1416  
  description "Pohled 1"  
}
```

```
DEF V2 Viewpoint {  
  position -10.4 1.8 6  
  orientation 0 1 0 -1.047  
  description "Pohled 2"  
}
```

```
Group {  
  children [  
    DEF kuzel1 Shape {  
      geometry Cone { bottomRadius 2 height 4 bottom FALSE }  
      appearance Appearance {  
        material Material { emissiveColor 1 0 0 }  
      }  
    }  
    Transform {  
      scale .5 .5 .5  
      translation 3 0 0  
      children USE kuzel1 }  
  ]  
}
```



Geometrie

- 10 uzlů
 - Box
 - Cone
 - Cylinder
 - Sphere
 - IndexedFaceSet
 - IndexedLineSet
 - PointSet
 - Extrusion
 - ElevationGrid
 - Text



Skupinové uzly

- Group
 - skupina uzlů
- Transform
 - transformace
 - translace
 - rotace
 - změna velikosti
- Shape
 - jeden objekt a jeho vzhled (geometry + appearance)
- Inline
 - externí soubory



Group

- vlastnoti
 - children []
 - pole potomků
 - bboxSize -1 -1 -1
 - nápověda pro prohlížeč, velikost bounding boxu = ohraničujícího kvádru
 - bboxCenter 0 0 0
 - střed bounding boxu



K čemu se Group hodí

- k ovládání skupiny uzlů jako celku
- transformace na celou skupinu



Transform

- transformace aplikované na potomky
- skludají se z
 - změny velikosti
 - rotace
 - translace



Transform


- **scale** # 1st transf.
- **scaleOrientation** # rotation before scaling
- **rotation** # rotation (2nd transf.)
- **center** # pivot point for rotation
- **translation** # 3rd transf.



Transform

```
Transform {  
  center C  
  rotation R  
  scale S  
  scaleOrientation SR  
  translation T  
  children [...]  
}
```

Ekvivalenty



```
Transform {  
  translation T  
  children Transform {  
    translation C  
    children Transform {  
      rotation R  
      children Transform {  
        rotation SR  
        children Transform {  
          scale S  
          children Transform {  
            rotation -SR  
            children Transform {  
              translation -C  
              children [...]  
            }  
          }  
        }  
      }  
    }  
  }  
}
```



Ukázka Translation

- jednoduchý stolek
 - <http://www.cgg.cvut.cz/LaskavyPruvodce/kap3/p-3-03.wr>



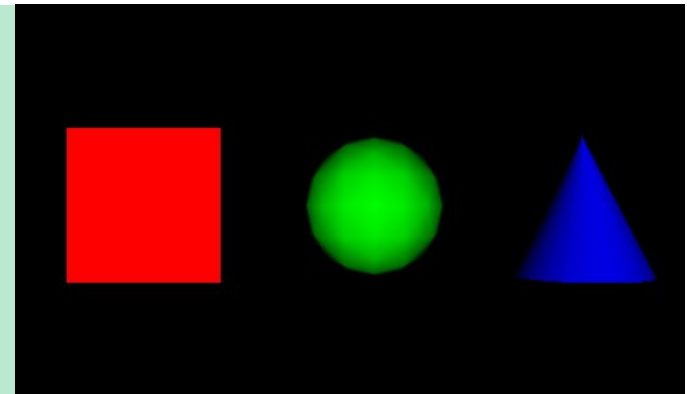
Shape

- **appearance**
 - **vzhled**
- **geometry**



Shape příklad

```
#VRML V2.0 utf8
Group { children [
  Transform {
    translation -3 0 0
    children Shape {
      geometry Box {}
      appearance Appearance {
        material Material { diffuseColor 1 0 0 }
      }
    }
  }
  Transform {
    children Shape {
      geometry Sphere {}
      appearance Appearance {
        material Material { diffuseColor 0 1 0 }
      }
    }
  }
  Transform {
    translation 3 0 0
    children Shape {
      geometry Cone {}
      appearance Appearance {
        material Material { diffuseColor 0 0 1 }
      }
    }
  }
]
```



Textury

- Existují 3 druhy textury
 - PixelTexture - pole pixelů
 - ImageTexture - obrázek PNG, JPG nebo GIF
 - MovieTexture - MPEG nebo animovaný GIF
- Textury jsou potomky uzlu Appearance
- texture nahrazuje parametr diffuseColor v uzlu Material



PixelTexture

- 4 typy:
 - 1: gray
 - 2: gray + α (alpha)
 - 3: RGB
 - 4: RGB + α (alpha)
- exposedField image 0 0 0 }
- width, height, type + hexadecimal values

- ukázka:

<http://www.cs.vu.nl/~eliens/documents/vrml/reference/EGS/PIXTEXE.W>

```
Shape {  
  appearance Appearance {  
    texture PixelTexture { # One component (gray scale)  
      image 4 4 1 0x00 0xDD 0xAA 0xFF  
                                0xDD 0x00 0xDD 0x00  
                                0xAA 0xDD 0x00 0x00  
                                0xFF 0x00 0x00 0x00  
    }  
    # Notice how the diffuseColor darkens the texture  
    material DEF M Material { diffuseColor .7 .7 .7 }  
  }  
}
```



ImageTexture

- url []
- repeatS
- repeatT

- ukázka:

<http://www.cg.cvut.cz/laskavyPruvodce/kap3/p-3-07.wrl>

```
#VRML V2.0 utf8

WorldInfo {title "Jiri Zara - Obrazek: Textura bez opakovani" }

Background {skyColor 1 1 1}

Viewpoint {
  position 28 0 50
  orientation 0 1 0 0.51
  description "Mirne zboku"
  fieldOfView 0.06
}

Transform {
  rotation 1 0 -0.59 0.3
  children
  Shape {
    geometry Box {}
    appearance Appearance {
      texture ImageTexture {
        url "obr-rgb.gif"
        repeatS FALSE
        repeatT FALSE
      }
    }
    textureTransform TextureTransform {
      scale 2 2
      translation -0.25 -0.25
    }
  }
}
```



Materiály a barvy

- Material
 - definuje jednu barvu pro celý bojekt
 - je to potomek Appearance
- ```
Material {
 ambientIntensity 0.2
 diffuseColor 0.8 0.8 0.8
 emissiveColor 0 0 0
 shininess 0.2
 specularColor 0 0 0
 transparency 0
}
```
- Color
    - pro jednotlivé plochy





# Material - parametry

---

- *ambientIntensity*
  - kolik ambientního světla bude odraženo
- *diffuseColor*
  - odráží světlo z definovaných zdrojů, záleží na směru
- *emissiveColor*
  - pro modelování zářících objektů
- *specularColor*
  - přidává se tehdy, když úhel pohledu je skoro stejný jako úhel dopadu, způsobuje odlesk
- *transparency*
  - průhlednost objektu



# Materiály

---

- doporučení: nastavte co nejvíce parametrů na 0
- výkonnostní problémy



# Zdroje světla

---

- Existuje několik zdrojů světla
  - pozor, materiály s emissiveColor nejsou zdroji světla
3. Čelovka avatara (headlight)
  4. Směrové světlo (DirectionalLight)
    - svítí podle pozice ve sceneGraph.
    - svítí jen na svoje potomky
  5. Všesměrové bodové světlo
    - svítí všemi směry stejně
    - má svůj horizont a definovaný útlum
  6. Směrové bodové světlo



# Světlo

---

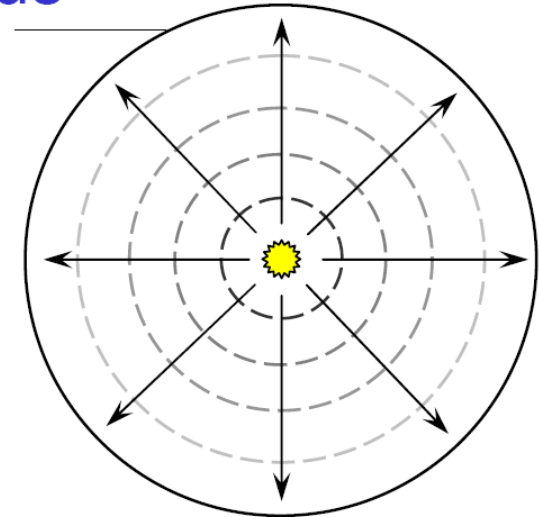
- `ambientIntensity`
  - všudypřítomné nesměrové světlo
- `color`
- `intensity`
- `on`



# Zdroje světla – všesměrové bodové světlo

```
PointLight {
 ambientIntensity 0
 attenuation 1 0 0
 color 1 1 1
 intensity 1
 location 0 0 0
 on TRUE
 radius 100
}
```

radius



attenuation: útlum, výpočet

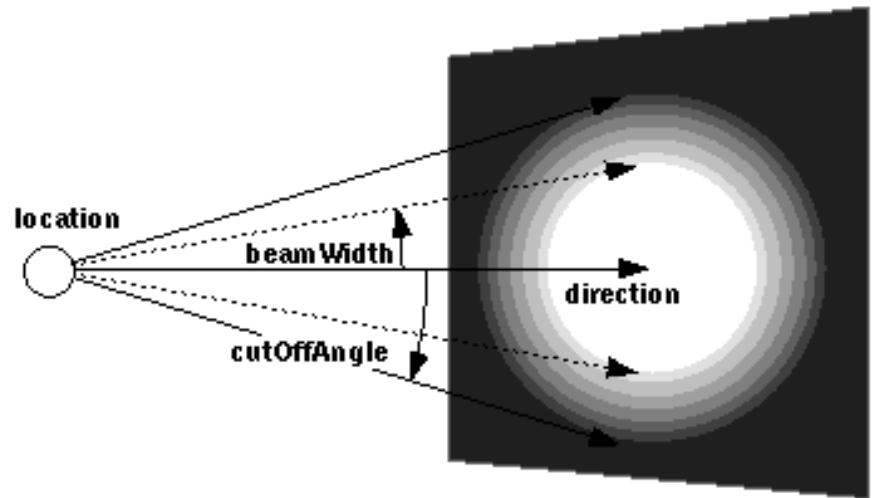
1

$$\max(\text{attenuation}[0] + \text{attenuation}[1] * r + \text{attenuation}[2] * r^2, 1)$$

# Směrové bodové světlo

- Svítí určitým směrem
- má rozptyl
- definovaná oblast s plným osvětlením

```
SpotLight {
 ambientIntensity 0
 attenuation 1 0
 beamWidth 1.570796
 exposedField 1 1 1
 cutOffAngle 0.785398
 direction 0 0 -1
 intensity 1
 location 0 0 0
 on TRUE
 radius 100
}
```



ukázka: <http://www.cs.vu.nl/~eliens/documents/vrml/reference/EGS/SPTLT.WRL>

# Jak dostat VRML model do HTML stránky

---

1. přímý odkaz na .wrl soubor
2. `<EMBED src="soubor.wrl" width="640" height="480" align="middle">`



# Odkazy

---

- Annotate VRML97 manual
  - <http://www.cs.vu.nl/~eliens/documents/vrml/reference/>
- Předmět x36MUS – přednášky
  - <http://service.felk.cvut.cz/courses/36MUS/>
- Laskavý průvodce virtuálními světy (Jiří Žára)
  - <http://service.felk.cvut.cz/courses/36MUS/>
- Předmět x36MUS – galerie
  - <http://service.felk.cvut.cz/courses/36MUS/galerie.html>

