

POV-RAY

POV-Ray (Persistence of Vision Raytracer) ist ein frei verfügbares Raytracing Programm. Beim Raytracing werden durch jedes Pixel des Bildes Lichtstrahlen zurückverfolgt bis sie ein Objekt treffen. Dort nehmen sie Farbinformation auf und werden je nach Oberflächenbeschaffenheit absorbiert, reflektiert, gebrochen, in mehrere Strahlen aufgeteilt. Dieser Prozess wird rekursiv bis zu einer gewissen Anzahl von Kollisionen mit Objekten durchgeführt. Danach werden alle Farbinformationen zu einem Pixel aufgesammelt.

Die Beschreibung einer Szene erfolgt in einem eigenen POV-Ray spezifischen Format, das der Programmiersprache C ähnelt. Aus manchen Programmen, wie etwa Maple, könne POV-Ray Dateien direkt exportiert werden. Es gibt auch einige teilweise frei verfügbare Programme, die andere 3D-Fileformate ins POV-Ray-Format konvertieren.

Die Beschreibung wird in einer Textdatei mit der Erweiterung *.pov abgespeichert. Hier ist ein einfaches Beispiel einer Szene bestehend aus einem Würfel und einer auf ihm liegenden Kugel:

```
// Globale Einstellungen
global_settings { assumed_gamma 2.2 }

//Includieren von vordefinierten Objekten
#include "colors.inc"
#include "shapes.inc"

background {Grey} //Hintergrundfarbe

camera { //Kameraposition
    location <6,3,3>
    angle 30
    look_at <0, 0.5, 0>
}
light_source { <10,20,30> , rgb 1 } //Licht

// Würfel gegeben durch diagonal gegenüberliegende Punkte
box { <-0.5, -0.5, -0.5>, <0.5, 0.5, 0.5>
    // Textur, Farbe, Reflexionseigenschaften,..
    texture {
        pigment {Green}
    }
}

// Kugel gegeben durch Mittelpunkt und Radius
sphere {<0,0,0> 0.5
    texture {
        pigment {Blue}
    }
    translate <0, 1, 0>
}
```

Bei der Windows-Version des Programmes können über das Menü „Insert“ nach Auswahl in der Liste automatisch vorgefertigte Codeabschnitte eingegeben werden. Die dabei eingefügten Standardwerte können dann anhand der im Quelltext eingefügten Kommentare oder – nach Klicken mit der Maus auf das entsprechende Schlüsselwort und Drücken der Taste <F1> - anhand des Hilfetextes verändert werden.

.Koordinatensystem

Das für die Berechnungen zugrundegelegte Koordinatensystem ist ein Linkssystem, dabei weist Die x-Koordinate nach rechts und die y-Koordinate nach oben (in der Schirmebene) sowie die z-Achse nach hinten.

Boolesche Operationen

Werden Objekte durch Boolesche Operationen aus Grundkörpern erzeugt, so genügt es für den Raytracing-Algorithmus den sogenannten CSG-Baum (Constructiv Solid Geometry) zu kennen (vgl. auch den Feature-Manager in Microstation), eine plizite Berechnung der Raddarstellung des Objektes ist nicht notwendig. Daher können im Quelltext von POV-Ray (im Gegensatz zu VRML) Boolesche Operationen angegeben werden. Mit #declare können für Objekte Variablennamen vergeben werden. Die untenstehende Datei zeigt den POV-Ray Quelltext für das untenstehende Bild.

```
global_settings { assumed_gamma 2.2 }

#include "colors.inc"
#include "shapes.inc"
#include "textures.inc"

camera {
    orthographic
    location <4, 6, -3>
    //direction <1, 1, 0>
    look_at <0, 0, 4>
}

light_source {
    <30, 10, 5> color white
    // spotlight
    // point_at <0,0,0>
    // radius 100
}

light_source {
    <20, 15, -10> color white
}

sky_sphere {
    pigment {
        gradient y
        color_map {
            [ 0.8 color Grey]
            [ 1.0 color MidnightBlue ]
        }
    }
}
```

```

difference {
    union {
        box {
            <0, 0, 0>, <1, 4, 4>
            pigment {color Blue}
        }

        cylinder {
            <0, 0, 0>, <1, 0, 0>, 2
            pigment {color Blue}
            translate <0, 4, 2>
        }
    }
    cylinder {
        <-1, 0, 0>, <2,0,0>, 1
        pigment {color Blue}
        translate <0,4,2>
    }
}

```

```

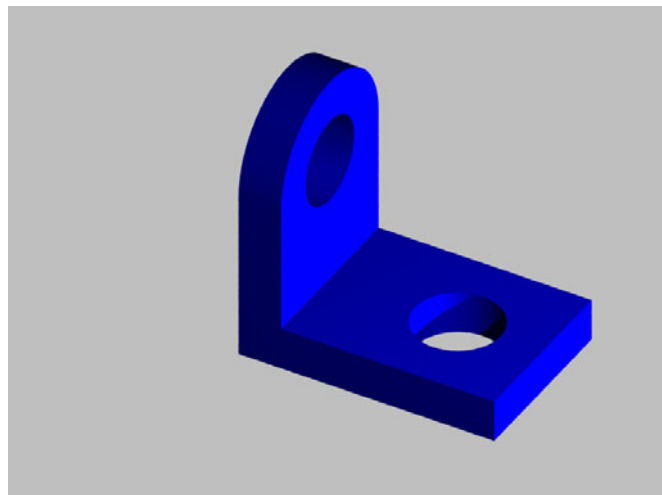
#declare box2 = object{
    box {
        <0, 0, 0>, <6, 1, 4>

        pigment {color Blue}
        translate <0, 0, 0>
    }
}

#declare cyl2 = object{
    cylinder {
        <0, -1, 0>, <0, 6, 0>,
1
        pigment {color Blue}
        translate <4, 0, 2>
    }
}

object{
    difference{
        object { box2}
        object { cyl2}
    }
}

```



Software und Informationen über POV-Ray

POV-Ray Homepage: <http://www.provray.org>

Internet Raytracing Competition: <http://www.irtc.org>