

VRML-Export und Animationen (GAM 14e)

Definition aus Wikipedia: Virtual Reality Modeling Language (VRML) ist eine Beschreibungssprache für 3D-Szenen, deren Geometrien, Ausleuchtungen, Animationen und Interaktionsmöglichkeiten.

VRML wurde ursprünglich als 3D-Standard für das Internet entwickelt.
VRML-Dateien erkennt man an der Dateierweiterung „.wrl“ (world), sie sind im Klartext (ASCII bzw. UTF-8) geschrieben und können auch in einem einfachen Texteditor erstellt werden.

Um sich eine VRML-Datei im Internet anzuschauen zu können, benötigt man ein Plugin (Ergänzungsmodul) für den Browser. Gratis zum Download zur Verfügung steht zum Beispiel Cortona VRML Client – <http://www.parallelgraphics.com/products/cortona/> für Windows, Mac OS X, Mac OS 9, Pocket PC

Würfel als VRML-Modell

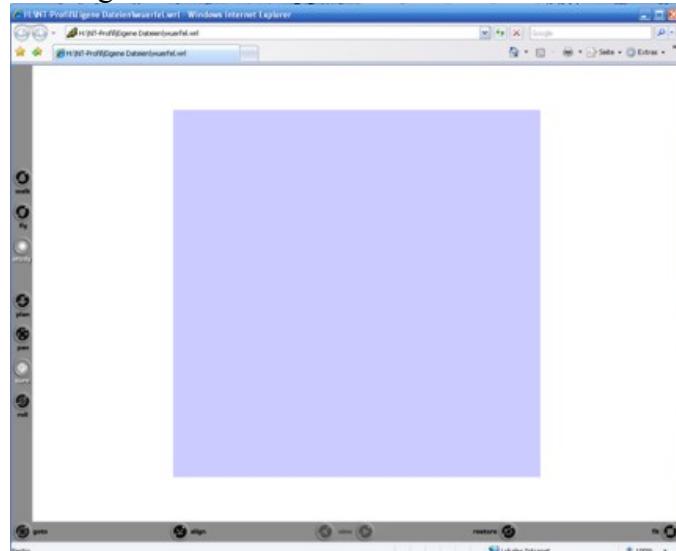
- ⊕ Konstruiere einen farbigen Würfel in GAM.
- ⊕ Gehe zu Datei/ exportieren/ VRML und speichere den Würfel unter dem Namen wuerfel an einen Platz deiner Wahl.
- ⊕ Bestätige die nächste Meldung, dass es bei Sonderzeichen Probleme geben kann, mit ja.
- ⊕ Verändere im nächsten Fenster nur die Farbe des Himmels durch Einfachklick auf den Button und Regelung mit den Schiebereglern. Wenn du in Betracht ziehest, einen Screenshot des Objekts anzufertigen, ist die Farbe weiß am besten.
- ⊕ Beantworte die Frage „VRML zeigen?“ mit ja.
- ⊕ Nun könnte das VRML-Bild schon angezeigt werden, oder es kommt eine Sicherheitsabfrage (Je nach Einstellung am PC). Sollte die Sicherheitsabfrage kommen, dann klick auf die INFO-Leiste und lass die geblockten Inhalte zu.

Die VRML-Datei sieht so aus. Es ist eine Draufsicht. Mit gehaltener linker Maustaste kannst du das Objekt jetzt bewegen.

Lass den Button „**study**“ aktiv.

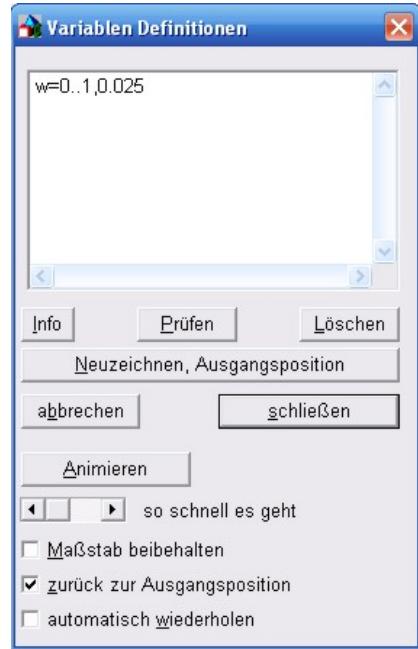
Ist zusätzlich „**turn**“ oder „**roll**“ aktiv, dann kannst du mit der LMT drehen.

Ist zusätzlich der Button „**pan**“ aktiv, dann kannst du verschieben und ist der Button „**plan**“ aktiv, dann kannst du zoomen.



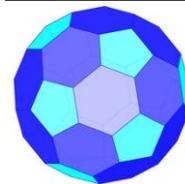
Rotierender Würfel als VRML Modell

- Konstruiere einen farbigen Würfel in GAM.
- Gehe zu Transformieren/ Drehen/ drehe den Würfel um die z-Achse um 360° und gib einen Haken bei animiert hinein. Es tut sich nichts.
- Gehe zu Bearbeiten/ Variable-Animation
Gam hat eine Variable w hinzugefügt, die von einem Minimalwert 0 bis zu einem Maximalwert 1 mit der Schrittweite 0.025 geht.
Kontrollierst du das Protokoll unter Bearbeiten/ Protokoll/ editieren, dann siehst du, dass GAM die Drehung mit 360^*w gespeichert hat.
- Unter dem Button Animieren hast du einige Einstellungen:
Lass den Schieberegler auf der langsamsten Einstellung. GAM passt den Bildschirm immer der Größe der Zeichnung an. D.h. GAM wird während der Animation die Objektgröße ändern. Willst du das nicht, musst du einen Haken in das Kästchen „Maßstab beibehalten“ machen.
Zurück zur Ausgangsposition heißt, dass GAM nach der Animation selbstständig das Objekt wieder zurück in die ursprüngliche Position bringt.
Automatisch wiederholen heißt, dass die Animation nach ihrem Ende immer wieder von vorne beginnt.
- Triff deine Einstellungen und klick auf den Button Animieren. Klick, wenn du dein Werk begutachtet hast und die Animation beenden möchtest, auf die Escape-Taste und klick danach auf „Neuzeichnen, Ausgangsposition“, um das Objekt in seine ursprüngliche Position zurück zu versetzen. Das ist vor allem für den VRML-Export wichtig.
- Gehe zu Datei/ exportieren/ VRML und speichere den Würfel unter dem Namen drehwuerfel an einen Platz deiner Wahl. Verwende im Dateinamen keine Sonder- oder Leerzeichen.
- Bestätige die nächste Meldung, dass es bei Sonderzeichen Probleme geben kann, mit ja.
- **Gib im nächsten Fenster einen Haken bei Animation exportieren hinein.** Das Fenster wird jetzt größer. Überlege genau, wie du die Animation haben willst, und setze entsprechende Häkchen. Verändere eventuell noch die Farbe des Himmels durch Einfachklick auf den Button und Regelung mit den Schiebereglern. Wenn du in Betracht ziehst einen Screenshot des Objekts anzufertigen, ist die Farbe weiß am besten.
- Beantworte die Frage „VRML zeigen?“ mit ja.
Nun könnte das VRML-Bild schon angezeigt werden, oder es kommt eine Sicherheitsabfrage (Je nach Einstellung am PC). Sollte die Sicherheitsabfrage kommen, dann klick auf die INFO-Leiste und lass die geblockten Inhalte zu.
Zuletzt sollte der animierte, sich drehende Würfel auftauchen. Er kann nun genau so behandelt werden, wie der stehende oben.



Erzeugung und Bewegen eines Fußballs mit GAM:

- ⊕ Stelle die Zeichenfarbe auf irgendeine Farbe außer schwarz, denn schwarz wird nicht schattiert und wirkt im VRML Modell nicht.
- ⊕ Wähle: 3D-Objekte/ weitere/ reguläre Polyeder.
- ⊕ Wähle dort dann Ikosaeder. Das ist ein platonischer Körper (ein Körper, der aus lauter gleichen regelmäßigen Vielecken besteht und keine einspringenden Ecken hat). Ein Ikosaeder ist ein „20-Flächner“ Er besteht aus 20 gleichseitigen Dreiecken. in jeder seiner Ecken stoßen 5 solche gleichseitigen Dreiecke zusammen.
- ⊕ Gib als Kantenlänge 3 ein.
- ⊕ Stelle nun eine andere Zeichenfarbe ein.
- ⊕ Gehe dann zu: Modellieren/ Fasen/ alle Ecken und wähle als Abstand 1.
Das nun entstehende Gebilde ist ein abgestumpftes Ikosaeder – allgemein als Fußball bekannt.



1. Animation:

Erzeuge in Gam einen zweifarbigen Fußball, der sich fortlaufend um sich selbst - um die z-Achse – dreht. Exportiere diese Animation als VRML-Modell.

2. Animationen:

Erzeuge in GAM einen zweifarbigen Fußball, verschiebe und kopiere diesen um 10 Einheiten nach rechts.

Animiere den ersten so, dass er sich fortlaufend um die y-Achse dreht und animiere den zweiten so, dass er den ersten und die z-Achse umkreist. Exportiere diese Animation als VRML-Modell.

3. Animationen:

Erzeuge in GAM einen zweifarbigen Fußball mit der Ikosaederseitenlänge 3. Verschiebe diesen Fußball T(0/ 5.5/ 0). Konstruiere einen Drehzylinder ($r = 3, h = 15$). Kopiere und verschiebe diesen Drehzylinder T(0/ 40/ 0).

Verschiebe den Fußball animiert T(0/ 29/ 0) so, dass er zwischen den beiden Zylindern hin und her pendelt. Exportiere diese Animation als VRML-Modell.

4. Animationen:

Erzeuge in GAM einen zweifarbigen Fußball mit der Ikosaederseitenlänge 3. Verschiebe diesen Fußball T(0/-20/15). Konstruiere einen Drehzylinder ($r = 3, h = 15$). Verschiebe diesen Drehzylinder T(0/-20/0). Verschiebe und kopiere diesen Drehzylinder um T(0/40/0) Drehe den Fussball um die x-Achse animiert um -98° . Exportiere diese Animation als VRML-Modell.

5. Animation (Für Experten):

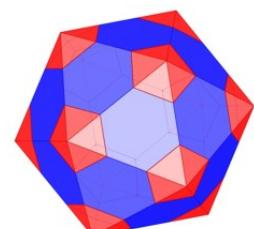
Erzeuge in GAM einen zweifarbigen Fußball mit der Ikosaederseitenlänge 3.

Gehe zu Bearbeiten/ Objekt duplizieren – der Fußball ist jetzt 2 Mal übereinander vorhanden. Ändere die Zeichenfarbe.

Wähle abermals: 3D-Objekte/ weitere/ reguläre Polyeder/ Ikosaeder.

Behalte die Kantenlänge bei. Gehe zu Modellieren/ Differenz und klick auf eine Ikosaederkante danach auf eine Fußballkante. Man sieht dann schön die fünfseitigen Pyramiden, die vorher beim Fasen abgeschnitten wurden. Zeichne 3D Objekte/ Koordinatenachsen in einer anderen Farbe. Verschiebe den Fußball und die Pyramiden so, dass der Körpermittelpunkt im Koordinatenursprung ist (erw. Punktfang/ wähle Objekte/ wähle Schiebevektor/ Mittelpunkt des Fußballs auf Koordinatenursprung). Die Schiebung sollte dann, -0.26728394223 in z-Richtung sein.

Wähle dann Transformieren/ Skalieren/ wähle eine Pyramide, gib als Skalierungsfaktor S(2/ 2/ 2) ein und mach einen Haken bei animiert.



Uhr

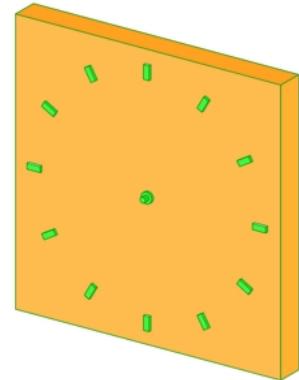
Ziffernblatt:

Erzeuge einen Quader ($10 \times 100 \times 100$) mit einer Farbe deiner Wahl und verschiebe diesen $T(-10 / -50 / -50)$ damit die x- Achse in der Mitte des Ziffernblattes ist.

Ändere die Objektfarbe, erzeuge einen weiteren Quader ($1 \times 2 \times 5$) und verschiebe ihn $T(0 / -1 / 40)$. Drehe diesen Quader um die x- Achse um 30° und kopiere ihn 11 Mal. Vereinige die Zeiten einzeln mit dem Quader.

Erzeuge einen Zylinder ($r = 3; h = 1$), erzeuge einen zweiten Zylinder ($r = 1; h = 3$) und vereinige die beiden Zylinder. Drehe danach die beiden Zylinder um 90° um die y- Achse, vereinige sie mit dem Quader.

Speichere die Datei unter „meinName_Ziffernblatt“ ab.



Großer Zeiger:

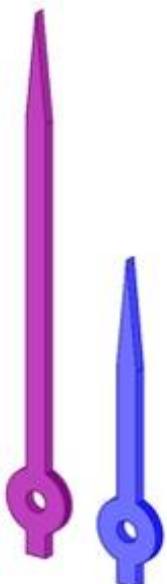
Öffne eine neue Datei.

Erzeuge einen Quader ($1 \times 2 \times 40$) und verschiebe ihn $T(1 / -1 / -5)$.

Erzeuge ein Satteldach [3D Objekte/ weitere/ Dächer] ($1 \times 2 \times 10$; **Kästchen bei Winkel muss leer** bleiben). Verschiebe das Satteldach auf den Quader und vereinige die beiden Objekte.

Erzeuge einen Zylinder ($r = 3; h = 1$), drehe ihn um 90° um die y- Achse, verschiebe ihn $T(1 / 0 / 0)$ und vereinige ihn mit dem Quader.

Erzeuge noch einen Zylinder ($r = 1; h = 5$), drehe ihn um 90° um die y- Achse und bilde die Differenz Quader\ Zylinder. Speichere die Datei unter „meinName_grosserZeiger“ ab.



Kleiner Zeiger:

Öffne einen neuen Datei:

Erzeuge einen Quader ($1 \times 2 \times 20$) und verschiebe ihn $T(2 / -1 / -5)$.

Erzeuge ein Satteldach ($1 \times 2 \times 10$; **Kästchen bei Winkel muss leer** bleiben). Verschiebe das Satteldach auf den Quader und vereinige die beiden Objekte.

Erzeuge einen Zylinder ($r = 3; h = 1$), drehe ihn um 90° um die y- Achse, verschiebe ihn $T(2 / 0 / 0)$ und vereinige ihn mit dem Quader.

Erzeuge noch einen Zylinder ($r = 1; h = 5$), drehe ihn um 90° um die y- Achse und bilde die Differenz Quader\ Zylinder. Speichere die Datei unter „meinName_kleinerZeiger“ ab.

Zusammenbau und Animation:

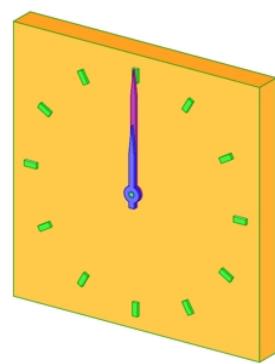
Öffne die Datei Ziffernblatt, füge den großen Zeiger hinzu und füge zuletzt auch den kleinen Zeiger hinzu.

Gehe dann zu Transformieren/ Drehen/ wähle den großen Zeiger und gib als Drehwinkel -360° animiert um die x- Achse ein.

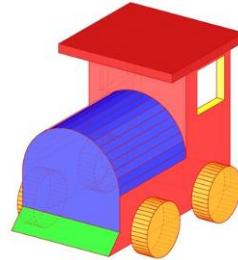
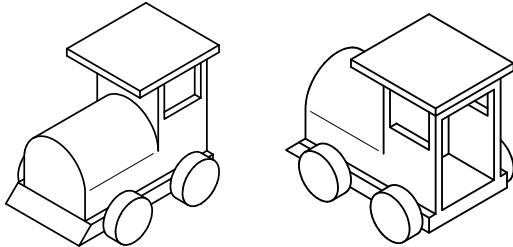
Gehe dann zu Transformieren/ Drehen/ wähle den kleinen Zeiger und gib als Drehwinkel -30° animiert um die x- Achse ein und speichere unter „meinName_Uhr“

Exportiere die animierte Uhr als VRML-Datei und speichere sie unter „meinName_Uhr“

Tipp: Beim Zeichnen deiner eigenen Uhr achte von Beginn an darauf, dass die x-Achse durch den Mittelpunkt des Ziffernblatts geht.

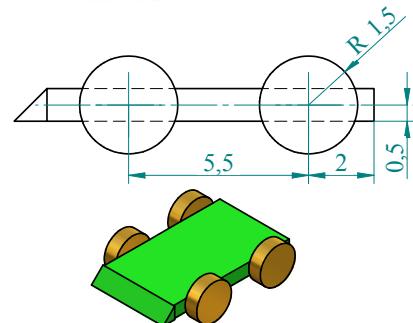


Lokomotive



Bodenplatte

Stelle eine Farbe deiner Wahl ein. Zeichne einen Quader ($10 \times 6 \times 1$) und einen Keil ($6.2 \times 1 \times 1$). Positioniere den Keil richtig (erw. Punktfang – Halbierungspunkt! Siehe Raumtransformtionen Seite 7 Wehrturm) und vereinige ihn mit dem Quader.



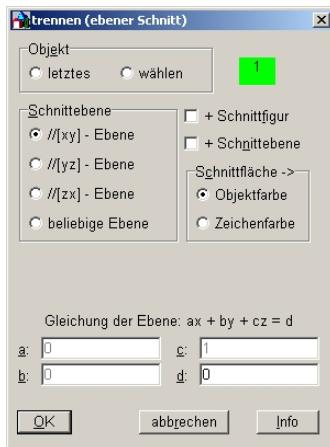
Räder

Zeichnen einen Zylinder ($r = 1.5 ; h = 1$) und drehe den ihn um 90° um die x-Achse. Verschiebe den Zylinder ($2 / 0 / 0.5$). Verschiebe und kopiere den Zylinder passend um das vordere Rad zu bekommen.

Spiegle und kopiere zuletzt die beiden Räder an der Symmetrieebene der Bodenplatte (Erw. Punktfang – Halbierungspunkt). Vereinige die 5 Objekte und verschiebe danach das Fahrgestell um $(-5.5 / 13 / 1)$. Speichere das erhaltene Fahrgestell unter „meinName_Fahrgestell“ ab. Öffne eine neue Datei.

Motorblock

Ändere die Zeichenfarbe. Zeichne einen Zylinder ($r = 3 ; h = 6$), drehe ihn um die y-Achse um 90° und gehe zu Modellieren/ trennen/ebener Schnitt, wähle die Schnittebene $\//xy$ und gib bei $d = 0$ ein (d.h. es wird parallel zur xy-Ebene in der Höhe $z = 0$ durchgesägt).



Gehe zu Bearbeiten/ Objekt löschen und lösche die untere Zylinderhälfte.

Zeichne einen Quader ($6 \times 6 \times 2$).

Füge Quader und Zylinder richtig zusammen (wähle Schiebevektor) und vereinige die beiden Teile Speichere das erhaltene Objekt unter „meinName_Motorblock“ ab. Öffne eine neue Datei.

Führerhaus

Ändere die Zeichenfarbe. Zeichne Quader 1 ($4 \times 6 \times 7$)

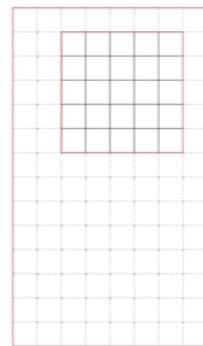
Raster in xz-Ebene ($0 / 4 / 0.5 / 0 / 7 / 0.5$)

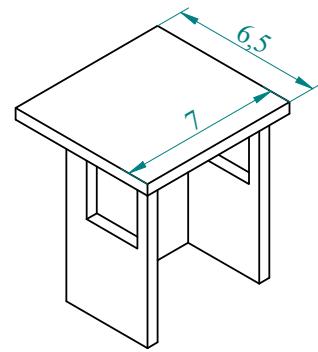
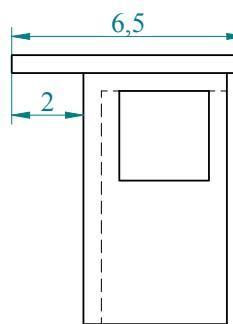
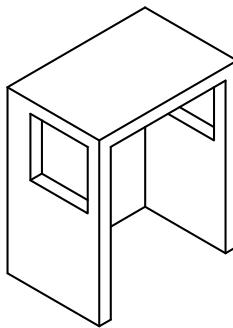
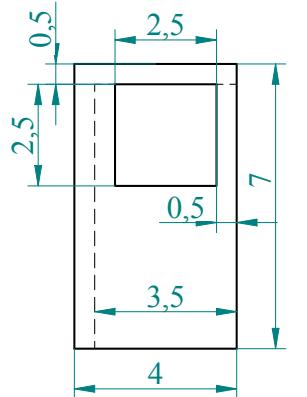
Schalte auf die Ansicht von rechts K(reuzriss) um.

Wähle Modellieren/ Bohrung prismatisch, klick den Quader außen an und zeichne das zu bohrende Rechteck. (Beachte, dass GAM vom letzten Eckpunkt zum Anfangspunkt das Bohrprisma selbstständig schließt!) Es soll 2.5 Einheiten breit und 2.5 Einheiten hoch sein, von links muss es 2 Kästchen entfernt sein! (1 Kästchen = 0.5×0.5)

Zeichne Quader 2 ($4 \times 5 \times 6.5$), verschiebe ihn ($-0.5 / 0.5 / 0$) und bilde die Differenz Quader 1\ Quader 2. Lösch das Raster.

Zeichne Quader 3 ($6.5 \times 7 \times 0.5$) und positioniere ihn, wie unten abgebildet.





Vereinige die beiden Objekte und speichere sie unter „meinName_Fuehrerhaus“ ab.

Öffne eine neue Datei. Konstruiere einen Zylinder mit $r = 20$ und $h = -0.1$

Füge die Datei Fahrgestell hinzu, füge danach die Datei Motorblock hinzu und positioniere den Motorblock auf dem Fahrgestell. Füge zuletzt die Datei Fuehrerhaus hinzu und positioniere sie richtig auf dem Fahrgestell. Vereinige die 3 Lokomotiventeile nicht, da in einem VRML Modell die Seitenflächen, sie sind flächenbündig, nicht richtig angezeigt würden. Speichere die Datei unter „meinName_Lokomotive“.

Gehe nun zu Transformieren/ Drehen/ wähle die 3 Lokteile gib als Winkel -360° ein und als Drehachse die z-Achse. Mach einen Haken bei animiert. Speichere die Datei unter „meinName_Lokbewegt“.

Wähle Datei exportieren/ VRML.....

Verschönere deine Animation in der Mitte durch einen Baum.

Hier einige Beispiele von Animationen, die Schüler erstellt haben:

<http://www.bglerch asn-ktn.ac.at/dg/vrml-4s/plieschner.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4s/salzerchristian.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4s/stuppacher.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/alfaiz-flugzeug.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/brunner-stadt.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/doppler-service.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/doppler-roboter.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/mahrl-baeume.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/paulitsch-flugzeug.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/paulitsch-service.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/privas-auto.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/wess-panzer.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/wolbang-auto.wrl>
<http://www.bglerch asn-ktn.ac.at/dg/vrml-4c/wolbang-service.wrl>

Aufgabe: Erzeuge ein eigenes Fahrzeug, das auf einer kreisförmigen Unterlage ein anderes Objekt (Busch, Baum, Statue, Haus....) umkreist