

Unterlagen
zur Vorlesung
Neue Medien im DG-Unterricht

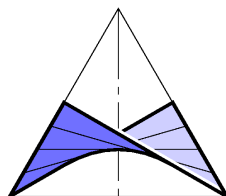
für die Studienrichtung Lehramt

Unterrichtsfach DARSTELLENDGEOMETRIE

von

Ao.Univ.Prof. Mag. Dr. Wolfgang RATH

Wien 2004



Institut für Diskrete Mathematik und Geometrie
Technische Universität Wien
Wiedner Hauptstraße 8-10/104, 1040 Wien
URL: <http://www.geometrie.tuwien.ac.at>

Einleitung

Ziel der Lehrveranstaltung "Neue Medien im DG-Unterricht" ist der Erwerb der Fähigkeit neue Medien zur Vorbereitung für den Unterricht sowie im Unterricht einzusetzen.

Folgende Inhalt werden behandelt: Erstellung von Prsentationen, realistischen Bildern, VRML-Modellen, Animationen und Arbeitsblättern mit geeigneter Software. Einübung an Beispielen, die für den Unterrichtseinsatz geeignet sind.

Die Leistungsfeststellung erfolgt durch drei bis vier Projekte, die auszuarbeiten bzw. in der Lehrveranstaltung zu präsentieren sind, sowie durch Abgabe eines Portfolios (Sammlung der Unterlagen und ausgearbeiteten Projekte in eventuell überarbeiteter und einheitlicher Gestaltung) bei einem mündlichen Prüfungsgespräch über den Stoff der Lehveranstaltung.

Folgende Softwareprodukte werden hauptsächlich verwendet:

- MicroStation (Version 2004)
- OpenOffice, StarOffice bzw. Word, PowerPoint
- Corel-Draw
- POV-Ray
- VRML-Viewer (Corona VRML-Client)
- Mediator

Hinsichtlich des Unterrichtes mit MicroStation beachten Sie auch die von Dr. Andreas Asperl gepflegte Internetseite <http://www.brg4.ac.at/acg/unterlagen>

Als Unterlagen zu dieser Vorlesung wird nicht ein durchgehendes Skriptum zur Verfügung gestellt, da sich die Erstellung von Modular aufgebauten Unterlagen hinsichtlich der Verwendung im Unterricht und hinsichtlich der Wiederverwendbarkeit bewährt hat.

ÜBERBLICK ZUR LEHRVERANSTALTUNG NEUE MEDIEN IM DG-UNTERRICHT WS 2004, W. Rath

Die Unterlagen der Lehrveranstaltung bestehen aus modular aufgebauten Arbeitsblättern. Dieser Überblick soll über die verwendeten Arbeitsblätter und die Abfolge der behandelten Themen informieren.

Beachten Sie auch die Internetseiten

<http://www.geometrie.tuwien.ac.at/rath/lva/nm> (Internetseite zur LVA von W.Rath)

<http://www.brg4.ac.at/acg/unterlagen> (Internetseite ACG von Andreas Asperl)

1. Erstellen von Arbeitsblättern mit MicroStation

Blattmodelle und Referenzen Erstellen
Plotkonventionen
Material
Licht
Arbeitsblatt Schatten und globale Beleuchtung
Konstruktion eines Parallelschattens – Einige Werkzeuge

2. Erstellen einer Powerpointpräsentation

Erstellen einer Powerpointpräsentation – Schattenkonstruktionen (Powerpointpräsentation)

3. Modellierung mit MicroStation

3D-Modellierung
Modeler – Features
Modeler – Features Ändern – Featuremanager
Tisch mit Gläsern

4. Animationen

Animation – Grundlagen
Animation – Keyframing
Animation – Definition von Pfaden
Animation mit Parametergleichungen
Animation eines Kardangelenks
Kamerafahrten
Eigene Materialien

5. VRML

VRML – X3D

6. POV-RAY

POV-RAY

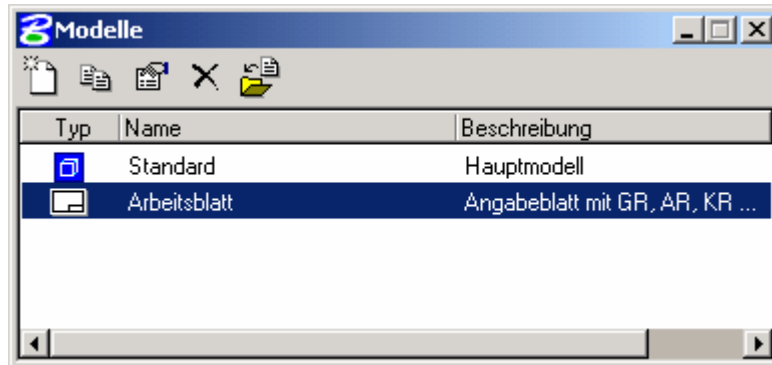
7. Arbeiten mit CorelDraw

Graphikbearbeitung
Farben
2D-Vektorgraphik
2D-Pixelgraphik

BLATTMODELLE UND REFERENZEN ERSTELLEN

Zum Erstellen eines Arbeitsblattes, das mehrere Risse, Bilder, Blattrand, Beschriftung etc. enthält verwenden wir Blattmodelle.

1. Wähle in der Symbolleiste **Primäre Funktionen** das Werkzeug **Modelle**.



2. **Erstelle ein neues Blattmodell "Arbeitsblatt"** durch Klicken auf die Ikone **Neues Modell erstellen** im Einstellungsfenster Modelle und fülle das Dialogfenster **Modell erstellen** folgendermaßen aus:

Modell erstellen

Typ: Blatt 3D

Seed-Modell: <Seed wird nicht verwendet>

Name: Arbeitsblatt

Beschreibung:

Logischer Ref.name:

Full Size 1=1 1.00000 : 1.00000

Blatteigenschaften

☒ Blatt-Layout darstellen

Größe: ANSI A H: 8.50000 W: 11.00000

Ursprung: X: 0.000000 Y: 0.000000

Drehung: 0.0000

Zelleigenschaften

☒ Kann als Zelle plziert werden

Zellentyp: Graphik

☒ Ansichtgruppe erstellen

OK Abbrechen

Bemerkung:

Wenn in der Seed-Datei bereits ein Blattmodell definiert war, kann man in dieses wechseln

- durch Doppelklick in die entsprechende Zeile des Einstellungsfensters Modelle
- durch Auswahl des Blattmodells im Einstellungsfenster Ansichtsgruppen

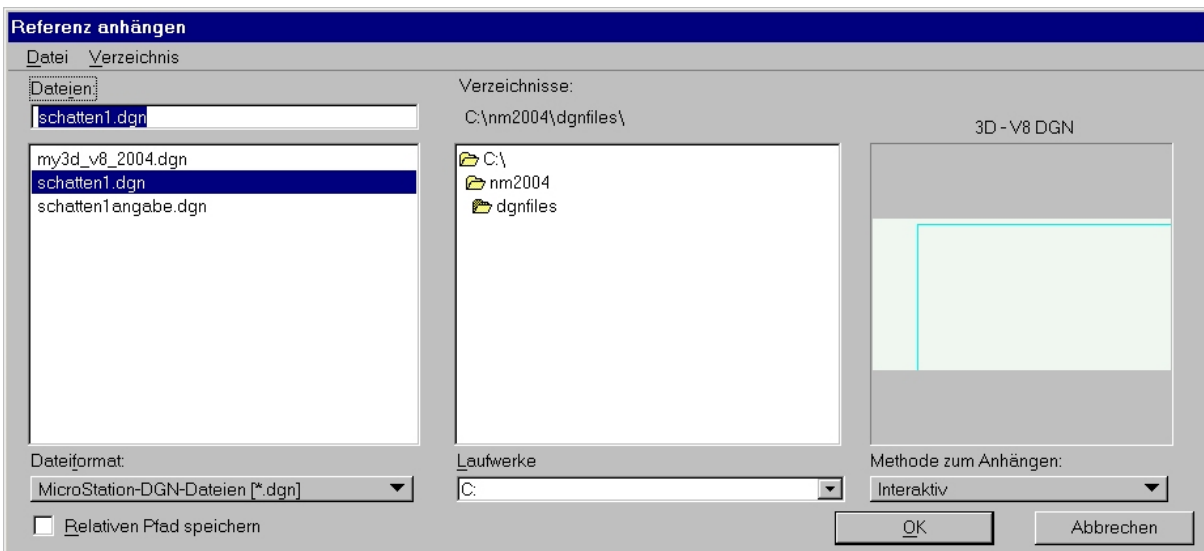
3. Einfügen von Referenzen:

Wähle im Einstellungsfenster Ansichtsgruppen das Blattmodell "Arbeitsblatt".

Wähle in der Symbolleiste **Primäre Funktionen** das Werkzeug **Referenzen**.



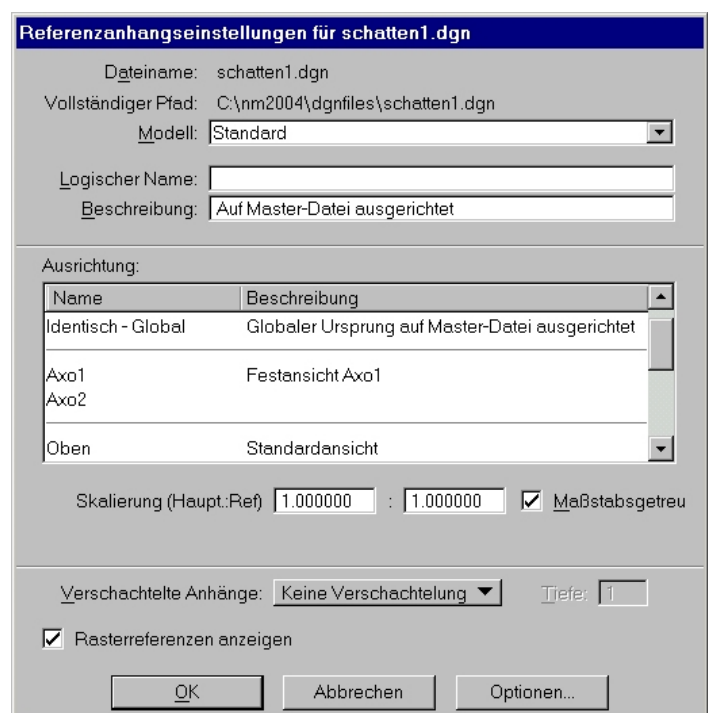
Wähle durch Anklicken der Ikone im Einstellungsfenster das Werkzeug **Referenz anhängen**. Im folgenden Einstellungsfenster ist die Datei auszuwählen, in der das Modell vorhanden ist, dessen Riss man auf dem Arbeitsblatt platzieren will.



Im nächsten Schritt sind im zugehörigen Einstellungsfenster folgende Einstellungen vorzunehmen:

Modell: Dort gibt man an welches Modell der Datei man anhängen will.

Ausrichtung: Dort gibt man an, ob man ein 3D-Modell (erster Abschnitt), eine selbstdefinierte Festansicht (zweiter Abschnitt) oder eine vordefinierte Ansicht (dritter Abschnitt) anhängen will



Nun ist die Ansicht mit der Maus in der Platzansicht zu platzieren



4. Einstellungen

Die Platzierungen und Einstellungen der angehängten Referenzen können später über die **Ikonen** in dem Einstellungsfenster Referenzen, den **Menüpunkt Einstellungen** oder im **Kontextmenü** (Klicken mit der rechten Maustaste auf eine Zeile der Liste) geändert werden

Zwei Werkzeuge sollen hervorgehoben werden:

Referenz falten (Seitenriss anhängen)

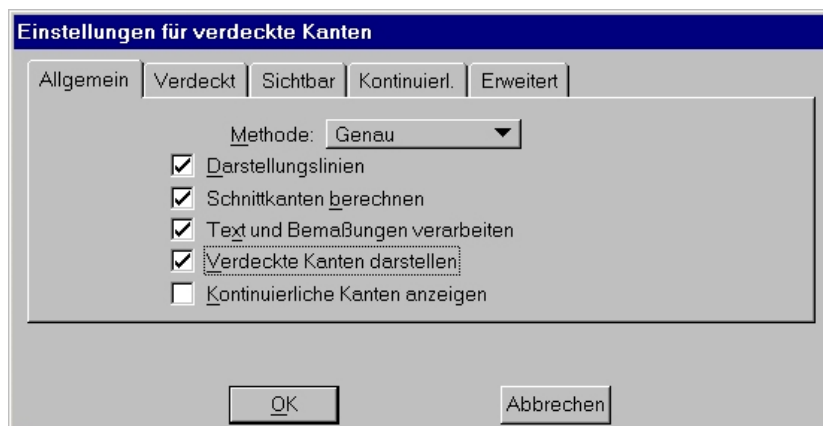
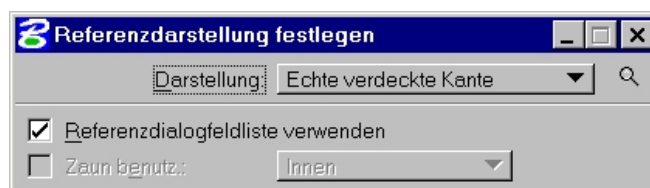


Durch Angabe einer "Rissachse" kann ein Seitenriss angehängt werden.

Darstellung



Hier kann festgelegt werden, wie die Ansicht des Objekts auf dem Blatt dargestellt wird. Die bestmögliche Qualität schattierter Bilder ist Phong (hier werden aber die Schatten nicht korrekt angezeigt). Will man ein Bild mit Methode Ray-Tracing auf dem Blatt platzieren, so muss man ein abgespeichertes Bild mit dem **Rastermanager (Primäre Funktionen)** anhängen.



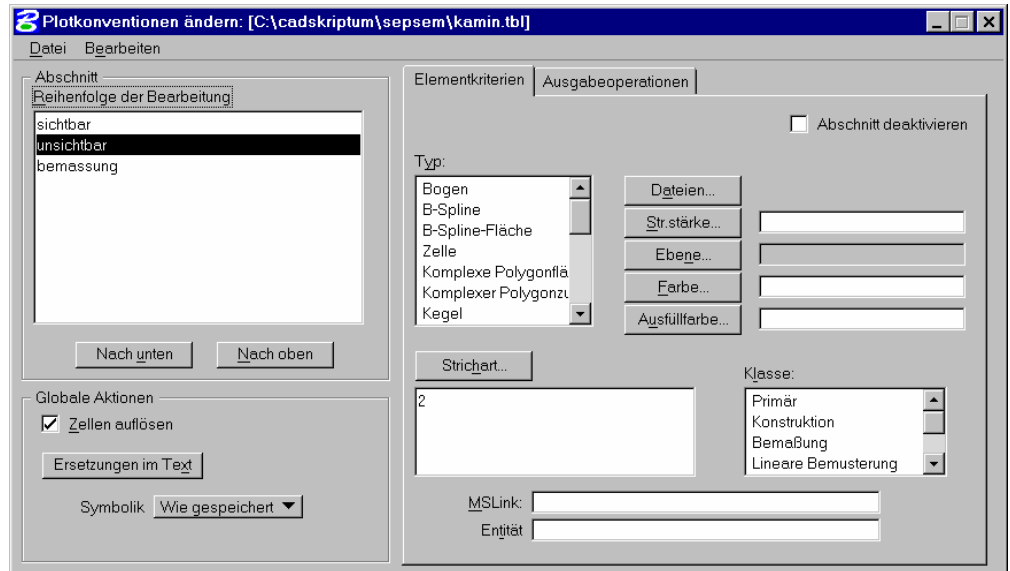
Mit der Einstellung "Echte verdeckte Kanten" wird die Sichtbarkeit für eine Liniengraphik korrekt berechnet, und unsichtbare Linien können auch dargestellt werden. Nach Klicken auf die Lupe wird das Einstellungsfenster für die detaillierten Einstellung der verdeckten Kanten geöffnet.

PLOTKONVENTIONEN

In den Plotkonventionen wird festgelegt, welche Objekte anders gedruckt werden sollen, als in der Zeichnung festgelegt bzw. am Bildschirm angezeigt. Dazu gehen wir folgendermaßen vor:

1. Im Funktionsfenster von Drucken das Menü "Plotkonventionen - Neu" wählen und in einem geeigneten Verzeichnis eine neue Datei anlegen (Erweiterung *.tbl ("Table") wird verwendet).

2. Mit dem Menü "Bearbeiten - Neuen Abschnitt einfügen" (bzw. umbenennen) einen **Abschnitt definieren**.
3. Danach geben wir in der Karteikarte "Elementkriterien" an, **welche Elemente abgeändert** werden sollen (etwa sichtbare und unsichtbare Linien, Bemaßung, ...).

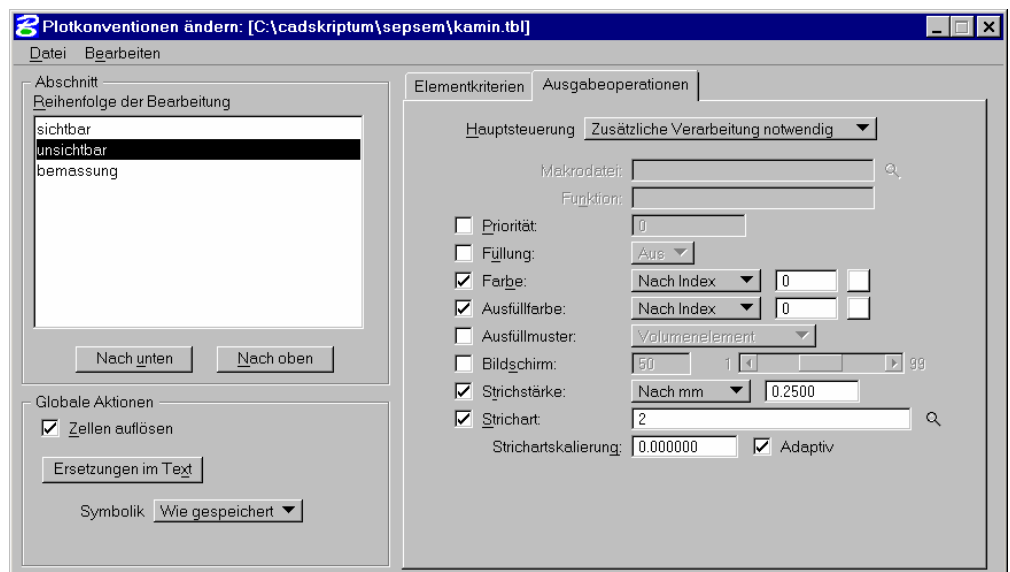


Falls in der Liste Typ bzw. Klasse kein

Element ausgewählt ist, wird die Definition auf Elemente aller Typen bzw. Klassen angewendet.

4. In der Karteikarte "Ausgabeoperationen" stellen wir dann ein, **was abgeändert** werden soll.

So können wir etwa definieren, dass alle unsichtbare Linien mit einer bestimmten Farbe 0 (angegeben durch MicroStation Farbnummer oder RGB-Wert), der Strichstärke 2.5mm (neben der Strichstärke mit dem Rollbalken von "Nach



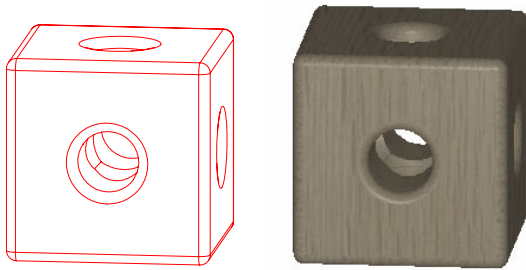
Index" auf "Nach mm" umstellen und 0.25 eingeben) und der Strichart 2 (strichliert) gezeichnet werden sollen. Durch Angabe der Strichstärke in den Plotkonventionen ist gewährleistet, dass diese nicht von Maßstab, Bildschirmauflösung etc. abhängt.

5. Die Abschnitte werden in der Reihenfolge der Liste von oben nach unten abgearbeitet und jeweils die letzte Definition ist gültig.

ACHTUNG: Nicht vergessen vor dem Verlassen des Fensters mit "Datei Speichern" die Definitionen abzuspeichern.

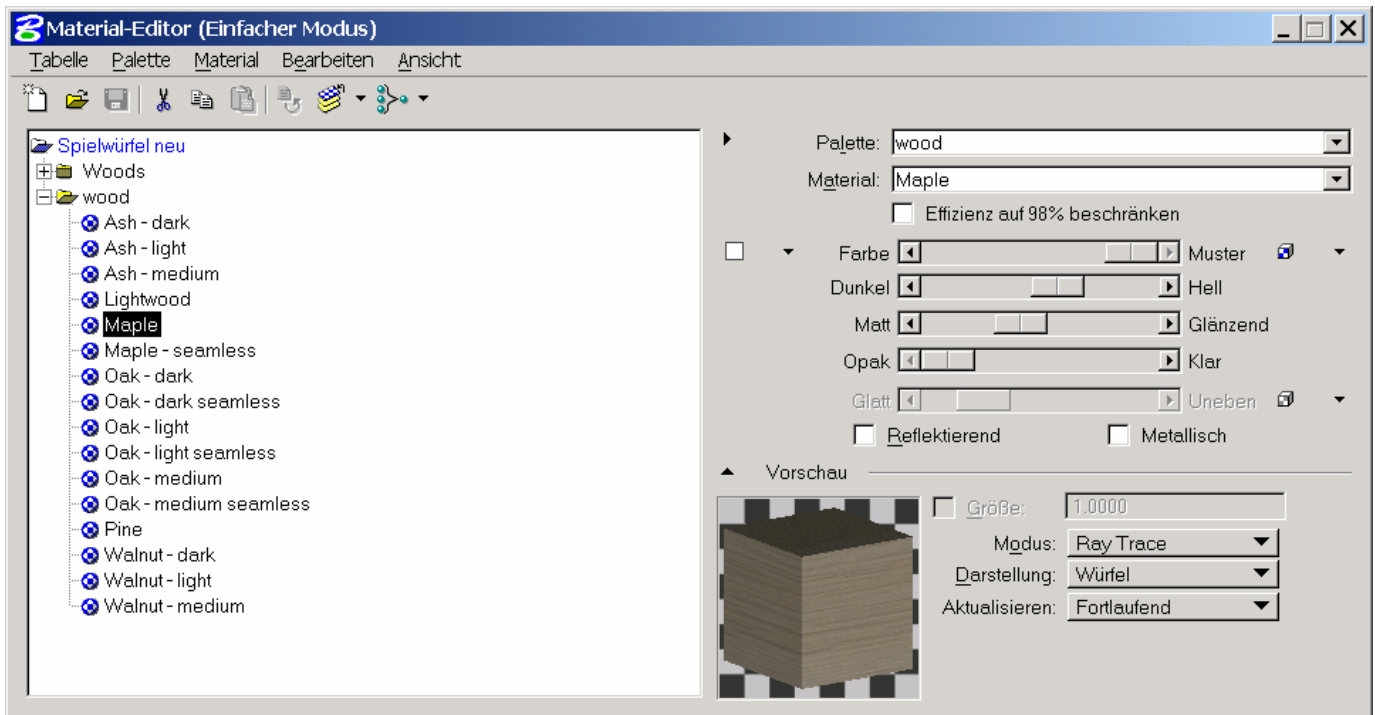
6. Dann das Fenster schließen und im Funktionsfenster Drucken mit dem Menüpunkt Plotkonventionen anhängen, die Definitionen aktivieren.

MATERIAL



Nun wollen wir das Material des Spielwürfels, etwa Holz, definieren. Dies geschieht mit dem Menüpunkt **Einstellungen – Rendern – Materialien**.

Im Funktionsfenster „Material-Editor“ öffnen wir mit **Palette - Öffnen** eine Palettendatei (*.pal), in der die Materialdefinitionen (Materialpaletten) gespeichert sind.

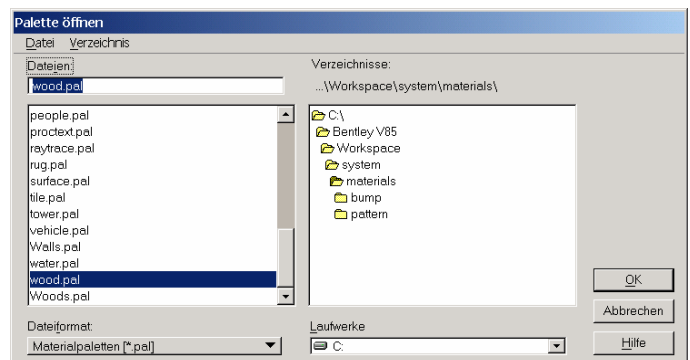


Im Verzeichnisfenster wählen wir die Datei wood.pal aus.

Hinweise: Für die Auswahl der Palettendatei befinden wir uns automatisch in dem Unterverzeichnis, in dem die mit MicroStation mitgelieferten Palettendateien liegen.

Paletten mit kleinem Anfangsbuchstaben stammen aus der Version 7 und früher.

Wir wählen nun durch Anklicken ein Material – etwa Maple (Ahorn) – aus. In der Vorschau können wir uns überzeugen, ob das gewählte Material passt.



Mit dem Menüpunkt **Material – Zuweisen** wird nach Anklicken des gewünschten Objektes allen Objekten einer Ebene mit derselben Farbe das ausgewählte Material zugewiesen. Diese Zuordnung kann mit dem Menüpunkt **Material – Zuweisungen bearbeiten** auch durch Eintippen der Ebenen- und Farbnummern erfolgen.

*Hinweis: Die Zuordnung des Materials geschieht also durch Ebenen- und Farbnummer und alle Objekte in derselben Ebene mit derselben Farbe haben dann dieses Material. Theoretisch ist es mit dem Menüpunkt **Funktionen – Material anhängen** auch möglich Objekten derselben Ebene verschiedene Materialien zuzuweisen, dies ist aber nicht empfehlenswert.*

Die Frage beim Schließen des Funktionsfensters, ob man Materialdefinitionen in der Datei *.MAT (mit demselben Namen wie die Zeichnungsdatei) gespeichert werden soll, beantworten wir mit JA.

Damit diese Definitionen nicht verlorengehen, muss man also beim Sichern von Zeichnungsdateien *.DGN immer auch die zugehörigen Materialdateien *.MAT sichern.

LICHT

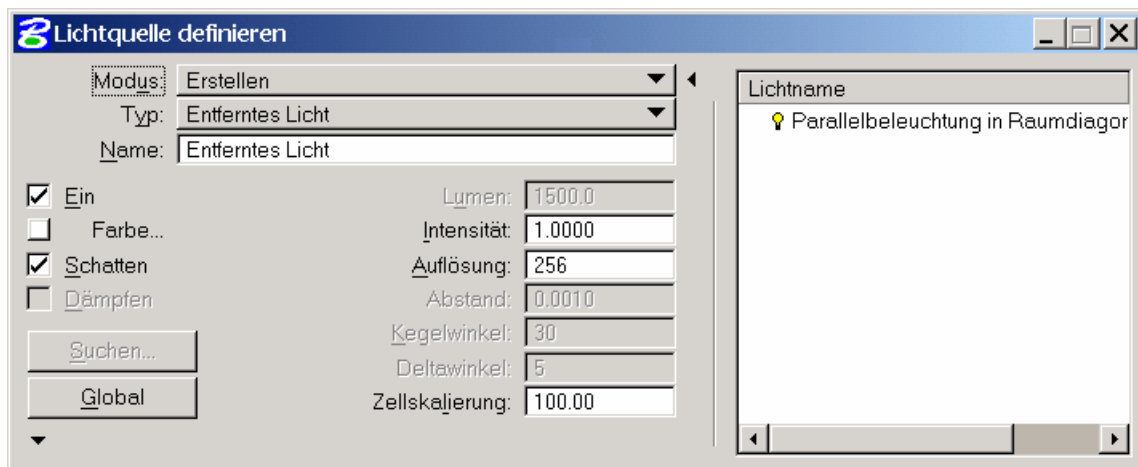
Um realistische Bilder zu erzeugen ist die Definition von Lichtquellen erforderlich. Standardmäßig ist aber ein Umgebungs- und Blitzlicht definiert, damit anschauliche Bilder ohne viel Aufwand erzeugt werden können.

Die Definition und Einstellung der Lichtquellen erfolgt mit den Menüpunkten **Einstellungen – Rendern – Globale Beleuchtung** sowie **Einstellungen – Rendern – Quellbeleuchtung**.



*Hinweis: Beide Menüpunkte können auch über die Werkzeugkiste **Rendern** (Funktionen – Visualisierung – Rendern) aufgerufen werden.*

Wir wollen zuerst mit **Einstellungen – Rendern – Quellbeleuchtung** eine neue Lichtquelle definieren:



Wir stellen zuerst im Einstellungsfenster alle nötigen Parameter ein. Als *Lichtquellentyp* stehen *Entferntes Licht* (Parallelbeleuchtung), *Punktlicht* (Zentralbeleuchtung), *Scheinwerfer*, *Flächenlicht* (zB Neonröhren) und *Himmelsöffnung* (für Radiositylösungen) zur Verfügung. Wir wählen vorerst Entferntes Licht und lassen alle anderen Parameter unverändert. Damit später auch Schatten angezeigt werden, überprüfen wir ob die Option Schatten aktiviert ist.



Dann starten wir mit dem Modus **Erstellen** die Eingabe der Lichtrichtung und geben diese durch Anfangs- und Endpunkt des Lichtvektors (etwa in Richtung der Raumdiagonalen des Würfels) an.

Die Lichtquelle wird durch ein Symbol samt den Lichtdaten angezeigt. Die Größe dieses Symbols wird über den Faktor **Zellskalierung** gesteuert.

Hinweis: Um das Lichtsymbol zu verbergen, ist es geschickt, sämtliche Lichter in eine eigene Ansichtsebene (Layer) zu legen und diesen dann auszublenden.

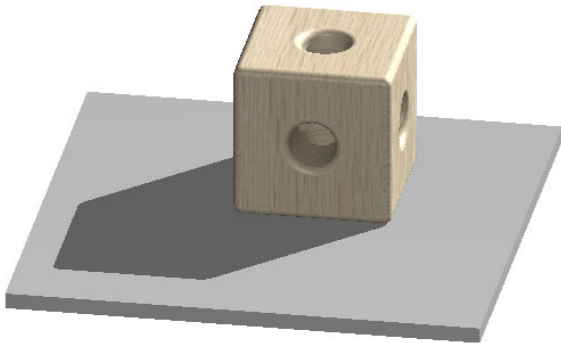
Damit das Objekt mit der gewählten Beleuchtung angezeigt wird, muss die Ansicht mit dem **Rendermodus Phong** oder **Ray-Trace** gerendert werden. Bei den „qualitativ schlechteren“ Rendermodi werden Lichtquellen nicht berücksichtigt.

SCHATTEN und GLOBALE BELEUCHTUNG

Schatten:

Wir stellen den Spielwürfel auf eine 30x30x10 große Platte.

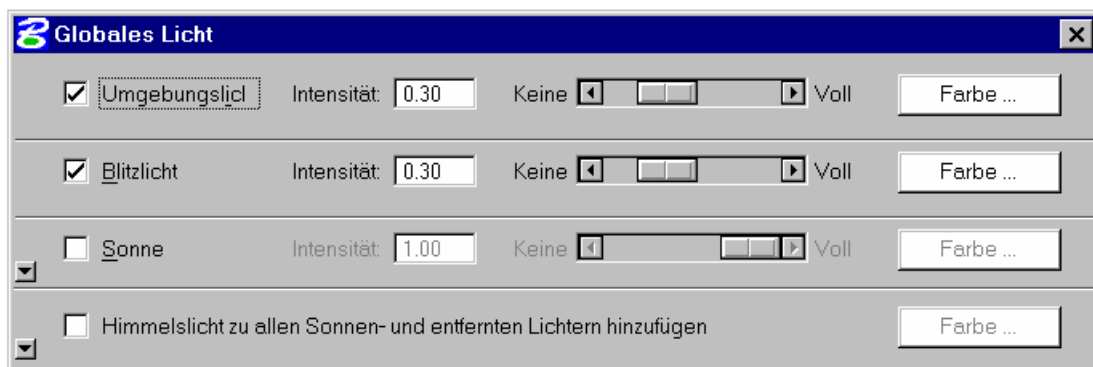
Damit beim Rendern der Schatten korrekt angezeigt wird, muss folgendes beachtet werden:



- Als Rendermodus muss **Ray-Trace** oder **besser** gewählt werden. (Bei Phong und aktiviertem Schatten bei Ansichtsattributen wird zwar der Schatten angezeigt, aber nicht korrekt!)
- Im Menüpunkt **Attribute zum Rendern der Ansicht** (Einstellungen – Rendern – Ansichtsattribute) müssen die Optionen Texturen/Schraffuren, Schatten, Transparenz aktiviert sein.
- Bei der **Definition der Lichtquelle** muss die Option **Schatten aktiviert** sein. Dies kann nachträglich im Menü **Einstellungen-Rendern-Quellbeleuchtung** mit dem Modus **Ändern** modifiziert werden.

Globale Beleuchtung:

Mit dem Menüpunkt **Einstellungen – Rendern – Globale Beleuchtung** lassen sich noch folgende Einstellungen treffen:



- **Umgebungslicht** (ambient light): diffuses Umgebungslicht, das keine Schatten wirft.
- **Blitzlicht** (head light): verhält sich wie das Blitzlicht eines Fotoapparates, mit dem die Szene betrachtet wird, oder das Licht einer Kopflampe, bei Betrachtung der Szene.

Hinweis: Durch Anklicken der kleinen Pfeile lassen sich Menüs mit weiteren Einstellungsmöglichkeiten für Sonne und Himmelslicht aufklappen.

- **Sonne:** Hier können der Einfallswinkel der Sonne eingegeben werden, oder zur Berechnung von Schattenbahnen das Sonnenlicht nach Eingabe von geographischer Länge und Breite sowie Uhrzeit der genaue Sonnenstand automatisch berechnet werden.
- **Himmelslicht:** Hier kann zusätzlich der Einfluß der „Wetterlage“ berücksichtigt werden.

KONSTRUKTION EINES PARALLELSCHATTENS – EINIGE WERKZEUGE

Anhand der Konstruktion eines Parallelschattens eines ebenflächig begrenzten Objektes im Raum sollen einige Strategien und Werkzeuge besprochen werden.

Gegeben sei

- ein Objekt als Volumsmodell,
- die Lichtrichtung durch eine Strecke s und
- die Schattenebene durch eine Rechteckfläche

1. Lichtstrahlen mit Schattenebene schneiden

Wir verlängern die Strecke s ausreichend und legen durch die Ecken des Objektes Lichtstrahlen, am besten durch Kopieren der Strecke s .



Mit dem Werkzeug **Volumenelement/Fläche mit linearem Element verschneiden** (3D-Haupt 2b-4) bestimmen wir die Schnittpunkte der Lichtstrahlen mit der Schattenebene.

2. Polygone parallel zur Schattenebene parallel verschieben

Der Schatten eines Polygons parallel zur Schattenebene geht durch Parallelverschiebung aus dem Original hervor.



Wir verwenden das Werkzeug **Flächen- oder Kantengeometrie extrahieren** (3D-Haupt 2b-3) um eine Seitenfläche der Oberfläche des Objektes zu extrahieren (dabei wird eine Kopie der Seitenfläche erzeugt) und verschieben das Polygon in die Schattenebene.

3. Schattenprisma als Extrusionsfläche erzeugen und trimmen

- Wie im Punkt 2 extrahieren wir eine Polygonfläche des Objektes (nicht notwendig parallel zur Schattenebene).
- Mit dem Werkzeug **Extrudieren** (3D-Haupt 1b-1) erzeugen wir eine Prismenfläche (Erzeugendenrichtung parallel zur Lichtrichtung). Dazu die Einstellung *Orthogonal* deaktivieren.
- Mit dem Werkzeug **Trimmung erstellen** (Fläche modellieren 1b-1) konstruieren wir das Schnittpolygon der Prismenfläche mit der Rechtecksfläche.

4. Schattenpolygonflächen erzeugen

Das Schattenpolygon erzeugen wir mit den Werkzeugen

- **Smartlinie** (Hauptwerkzeugkasten 2b-1)
- **Polygonfläche** (Hauptwerkzeugkasten 3b-2)
- **Bereich erstellen** (Hauptwerkzeugkasten 6a-4)

5. Schattenpolygonfläche schraffieren

Mit dem Werkzeug **Elementfläche schraffieren** (Hauptwerkzeugkasten 3a-1) (oder anderer Werkzeuge aus dem Werkzeugkasten 3a) schraffieren wir die Schattenpolygonflächen.

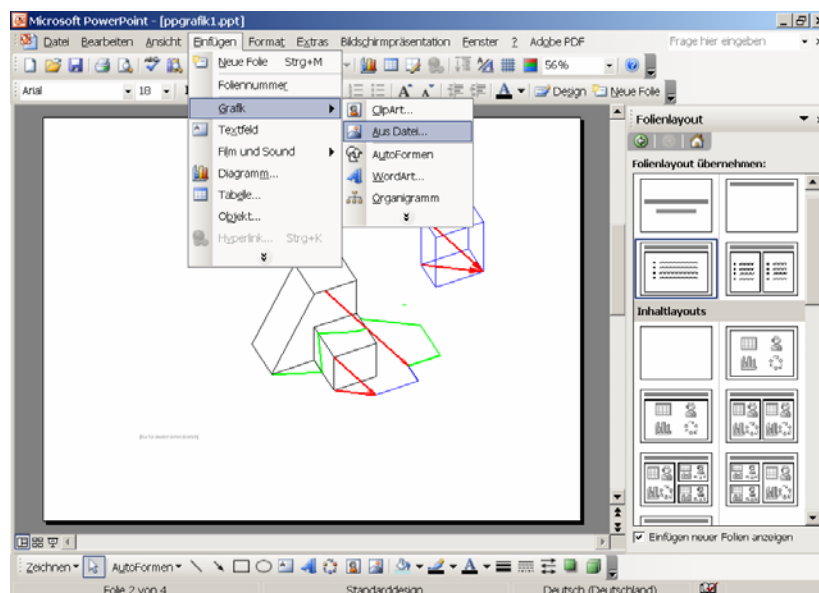
6. Schattenpolygonfläche ausfüllen

Alternativ zur Schraffur können wir die Polygonfläche auch mit einer Farbe ausfüllen:

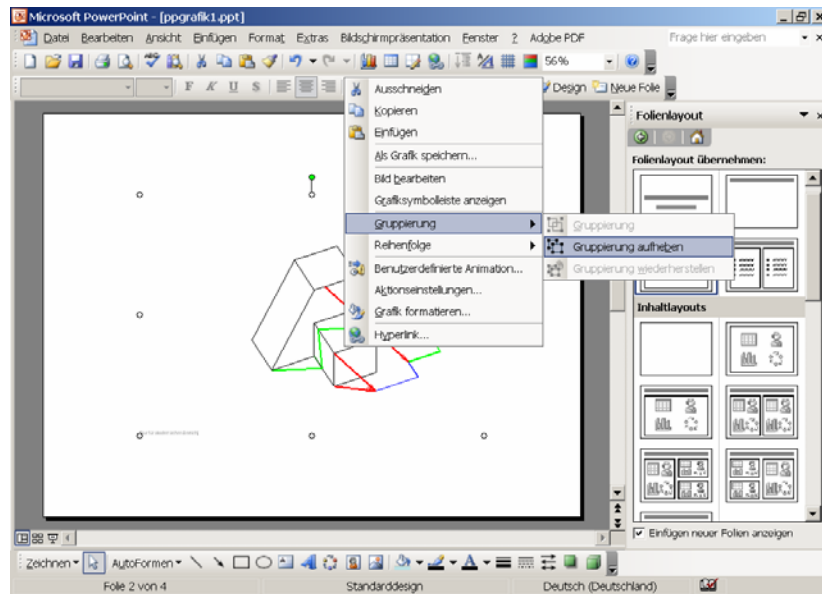
- bei der Konstruktion der Polygonfläche mit dem Werkzeug **Polygonfläche** (Hauptwerkzeugkasten 3b-2)
- nachträglich **Element auf aktive Ausfüllart einstellen** (Hauptwerkzeugkasten 8a-3)
*Beachte, dass für die Ansichtsattribute im Zeichenfenster **Ausfüllung aktiviert** ist!*

Erstellen einer Powerpointpräsentation

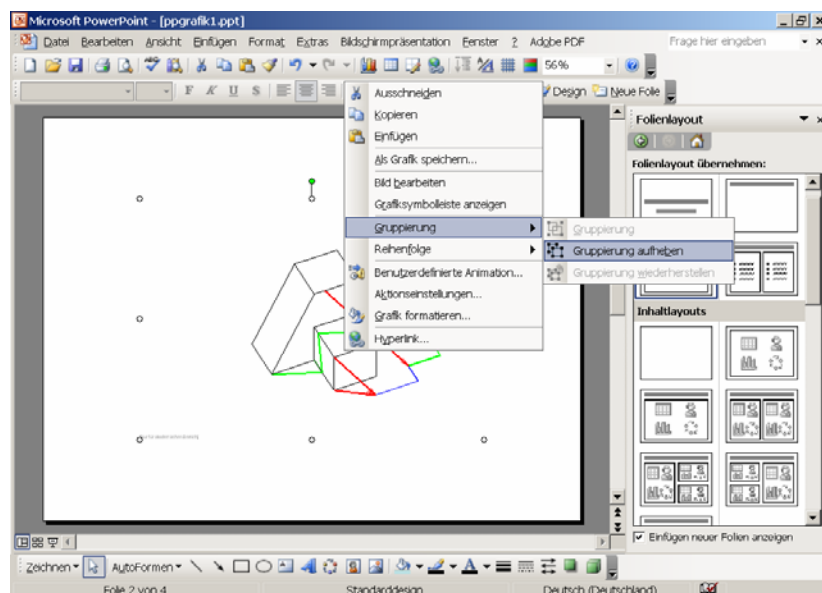
Schattenkonstruktionen



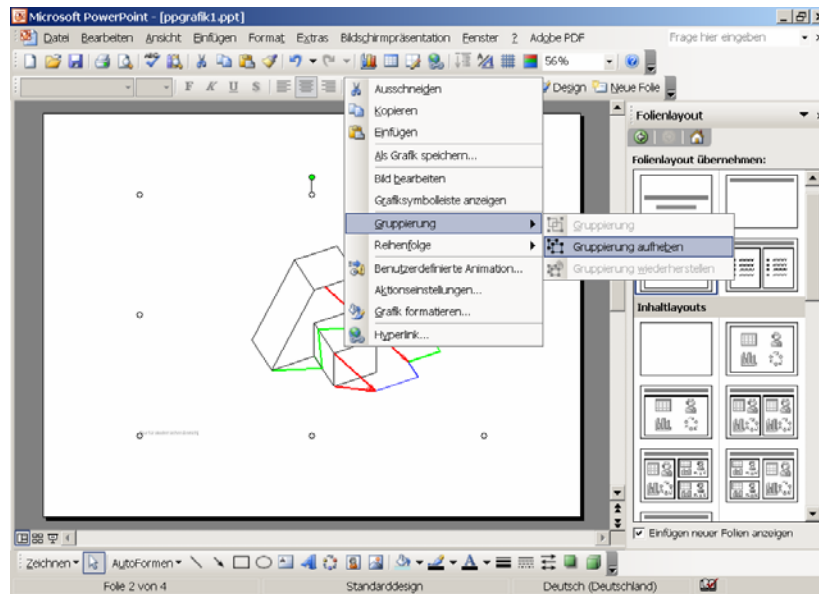
Grafik (z.B.) einfügen mit Einfügen – Grafik – Aus Datei



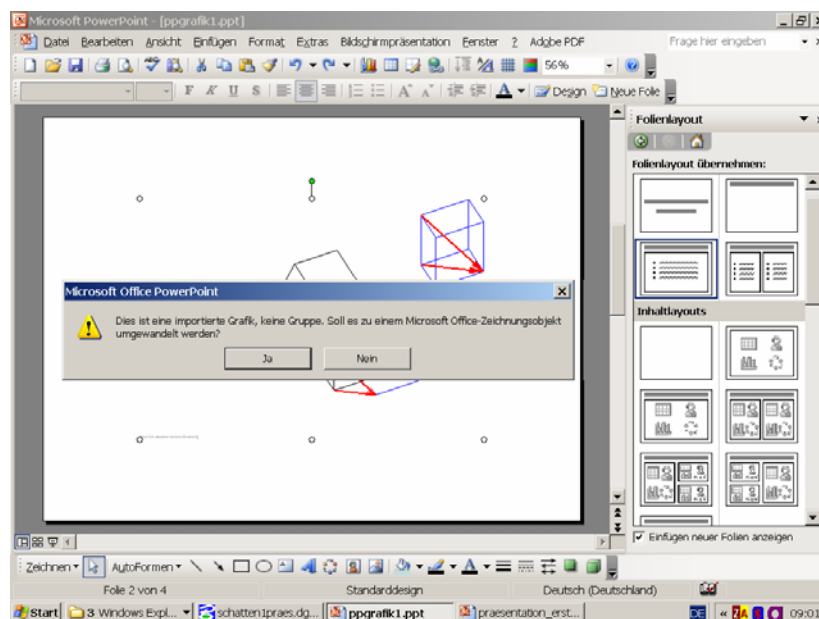
Auswählen – rechte Maustaste – Gruppierung – Gruppierung aufheben



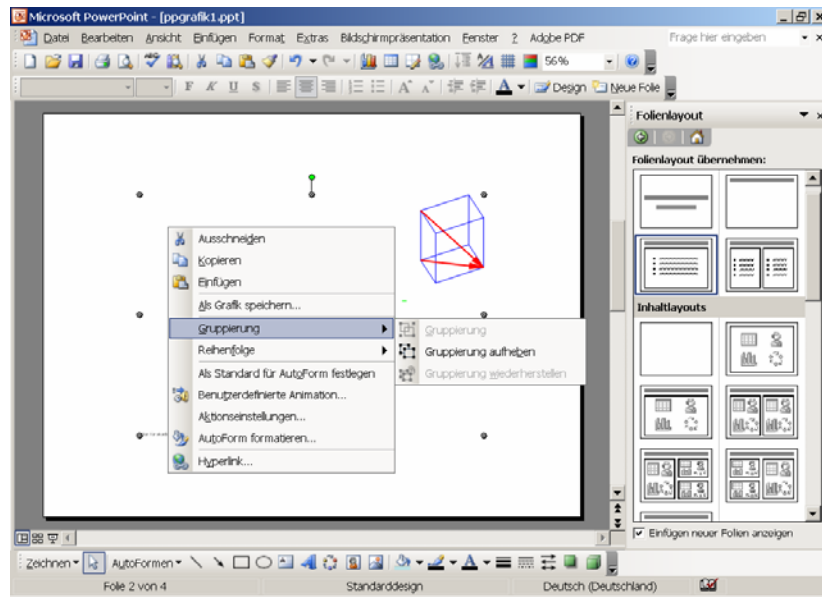
Auswählen – rechte Maustaste – Gruppierung – Gruppierung aufheben



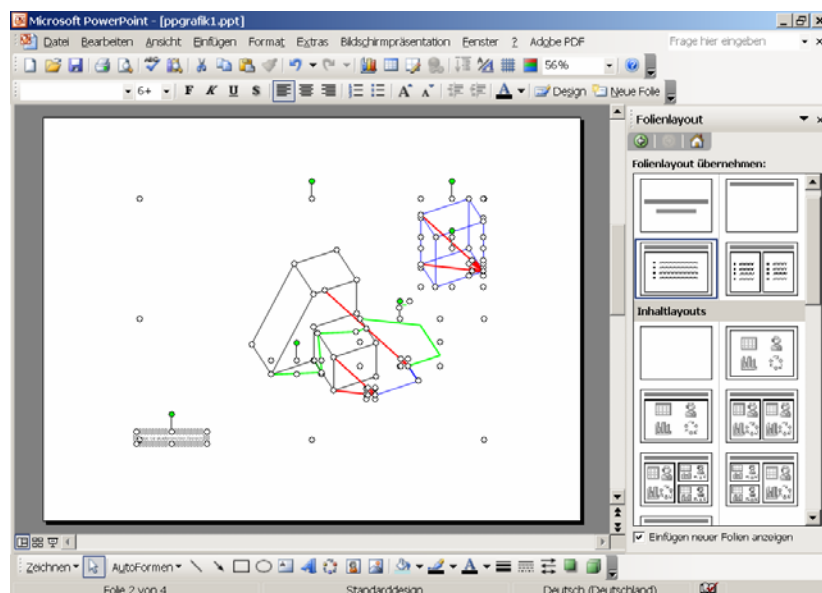
Auswählen – rechte Maustaste – Gruppierung – Gruppierung aufheben



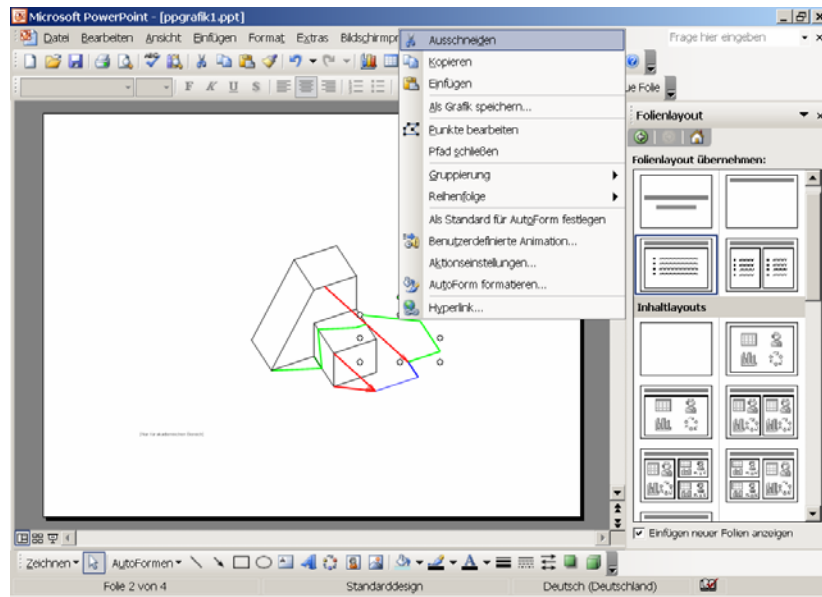
Grafik umwandeln? - Ja



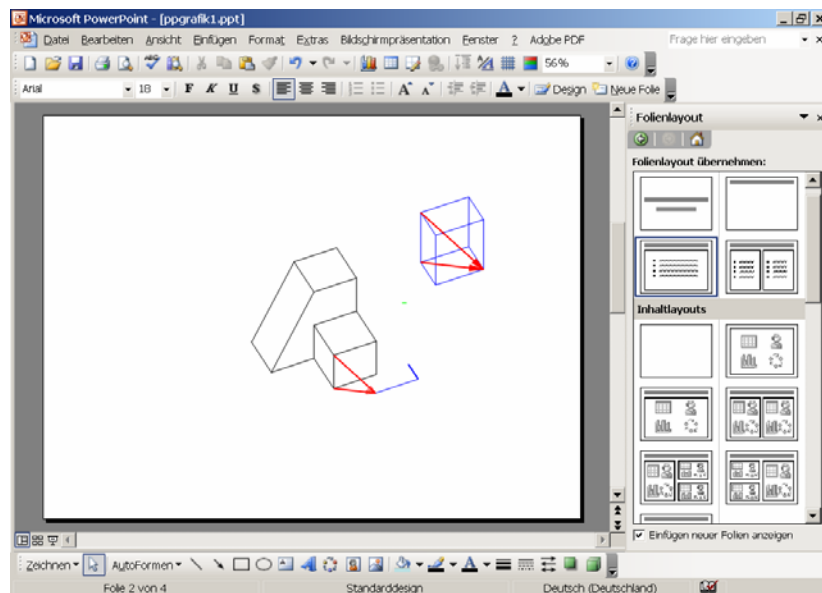
Gruppierung aufheben so oft wie nötig



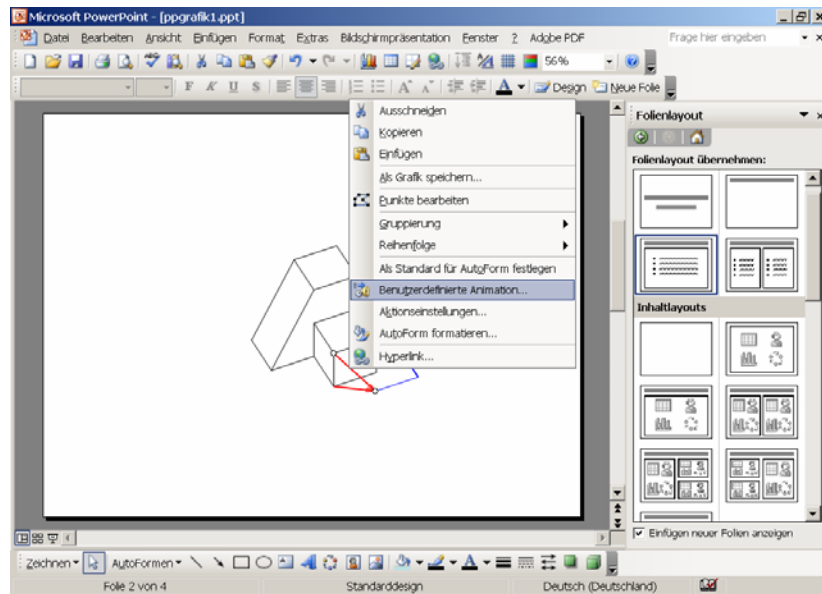
Gruppierung aufheben so oft wie nötig



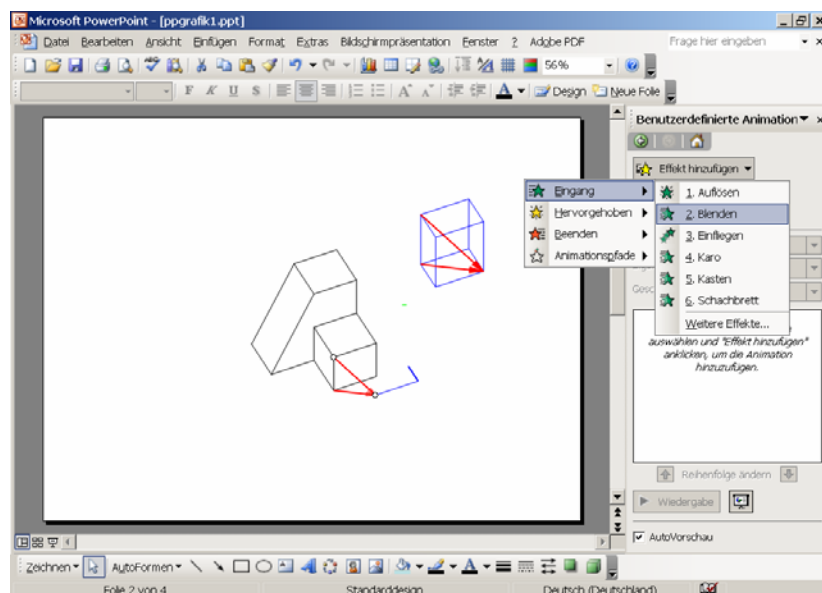
Nicht benötigtes Löschen bzw. Ausschneiden



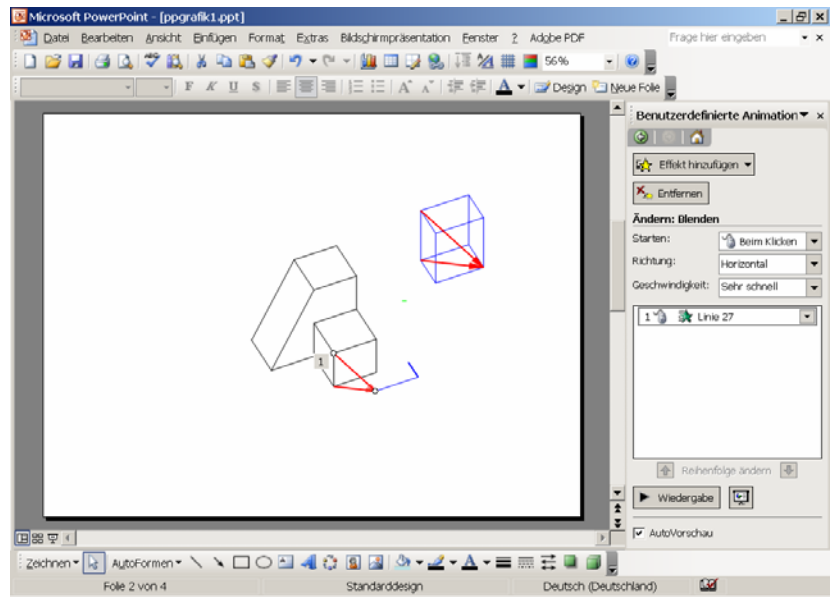
Animation planen und bearbeiten



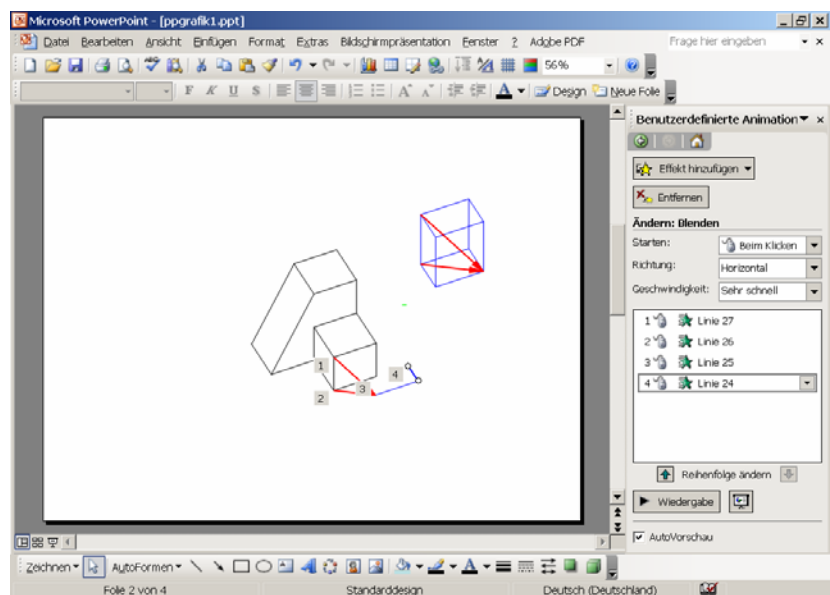
Element Anklicken – rechte Maustaste – Benutzerdefinierte Animation



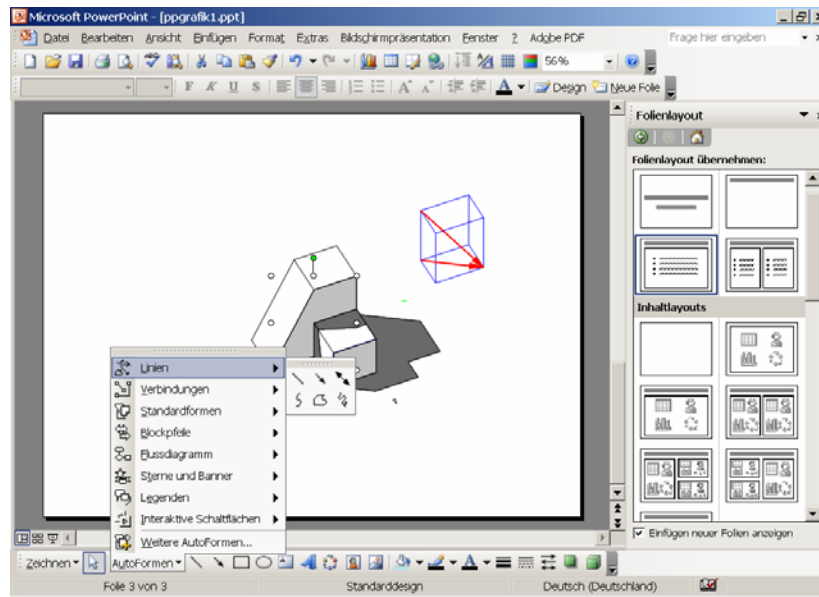
Effekt hinzufügen



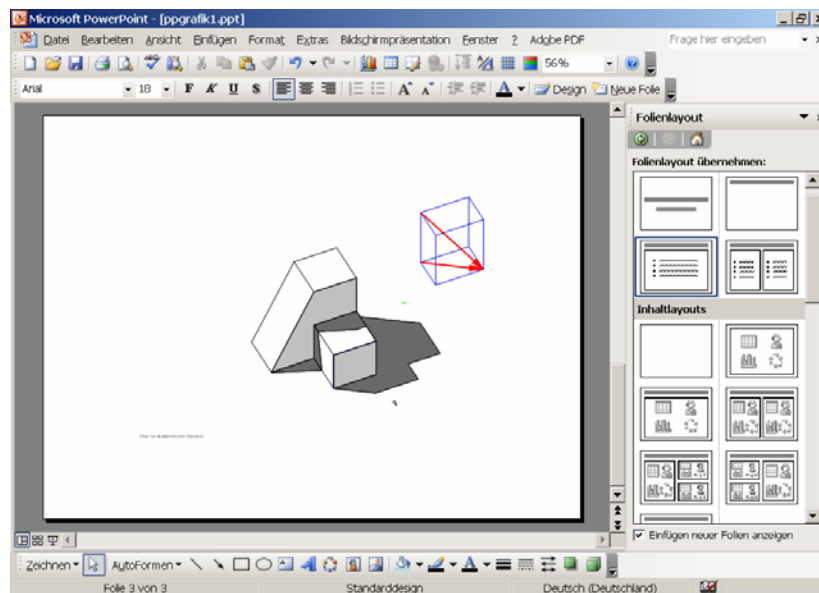
Effekt hinzufügen



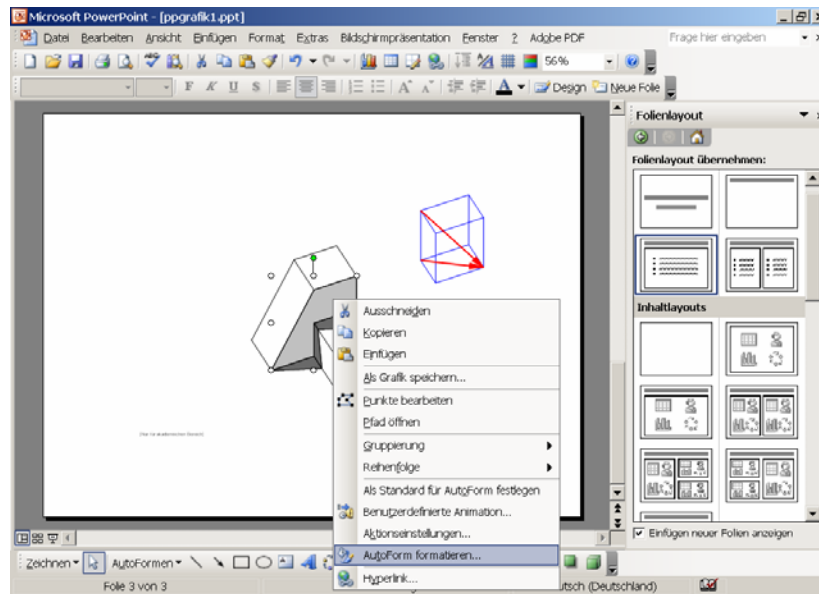
Der Reihe nach weitere Effekte hinzufügen



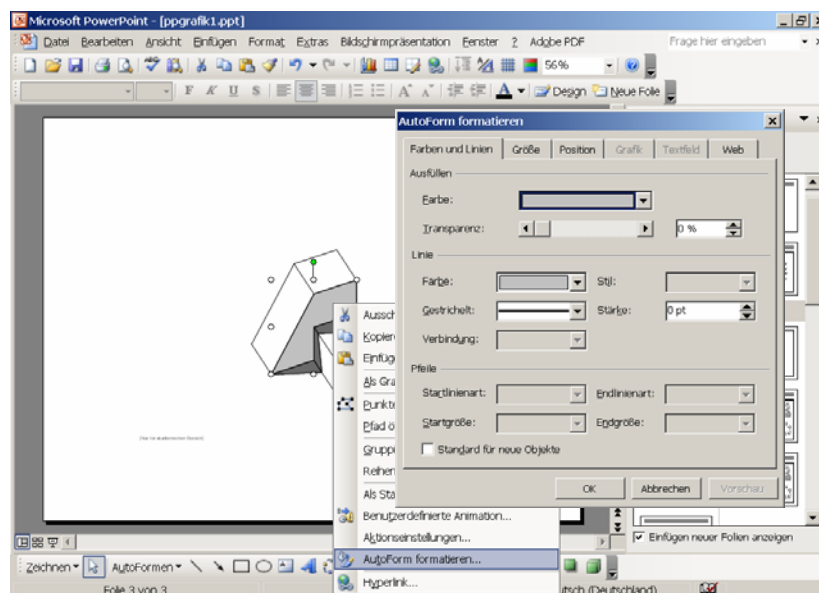
Eventuell mit Autoformen etwas dazuzichnen



Weitere Folien einfügen



Rechte Maustaste – AutoForm formatieren



AutoForm formatieren – Elemente modifizieren

3D-MODELLIERUNG

Zur Beschreibung dreidimensionaler Objekte werden i.a. folgende Modellbildungen verwendet:

- **Kantenmodell** (Drahtgittermodell, Wireframe): die Objekte werden durch Linien und Kurvelemente beschrieben. Es liegt keine Information vor, welche Flächen und Volumina von den Linienelementen eingeschlossen werden; eine automatische Sichtbarkeitsdarstellung des beschriebenen Körpers ist nicht möglich.
- **Flächenmodell**: nur die Oberfläche der Objekte wird generiert; dies entspricht dem Zugang der klassischen Darstellenden Geometrie.
- **Volumsmodell**: hier liegen auch Information über Volumina vor. Im allgemeinen werden Volumskörper aus Grundkörpern zusammengesetzt, etwa durch Boole'sche Operationen. Für die interne Darstellung von Volumsmodellen gibt es zwei unterschiedliche Verfahren:
 - **CSG = Constructive Solid Geometry**: Dabei wird die Entstehungsgeschichte, die Abfolge der Boole'sche Operationen wie Vereinigung, Durchschnitt und Differenz in einem binären Baum abgespeichert.
 - **Begrenzungsflächenmodell (B-Rep = Boundary Representation)**: Dabei werden die Körper durch ihre Oberflächen und zusätzliche Informationen über die Topologie, auf welcher „Seite“ Material vorhanden ist, dargestellt; B-Rep zeigt nur das Ergebnis einer Konstruktion, der Konstruktionsablauf wird nicht mit dem Modell abgespeichert. Trotzdem muss für gewisse Sichtbarkeitsalgorithmen eine B-Rep-Darstellung aus dem CSG-Baum (automatisch) generiert werden.
- **Feature**: Dies übersetzt man wohl am besten mit **Eigenschaft** bzw. **Funktionalität**. So ist etwa eine Bohrung ein Feature. Zusätzlich zur den erzeugenden Booleschen Operationen (hier etwa Differenz von Quader und Zylinder) werden auch weitere Eigenschaften abgespeichert, etwa die Tiefe der Bohrung oder ob ein Durchgangsloch gebohrt werden soll. Dies bietet Vorteile bei Variantenkonstruktionen: Ändert man etwa die Dicke einer Platte so wird bei Erzeugung eines Durchgangsloches als Feature nach Änderung wieder ein durchgebohrtes Loch vorhanden sein, während bei Erzeugung des Objektes durch Boolesche Operationen nach Änderung möglicherweise ein Sackloch entsteht.

Um alle Möglichkeiten der Volumsmodellierung auszunutzen verwenden wir das zusätzliche Modul „Modeler“ von Microstation. Dieses wird beim Programmstart durch Auswahl des Benutzers „Modeler“ aktiviert. Damit gibt es den zusätzlichen Hauptwerkzeugkasten **Funktionen – Modeler**.

Parametrisches Volumen

(M1a)

Parametrische Freiformvolumen

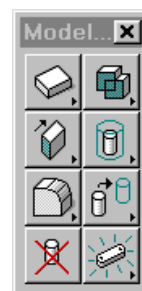
(M2a)

Feature erstellen

(M3a)

Feature löschen

(M4a)



(M1b) **Boolesche Operationen**

(M2b) **Feature ändern**

(M3b) **Feature manipulieren**

(M4b) **Baugruppen**

Zusätzlich werden auch manche Menüs und Werkzeuge erweitert.



Alle Booleschen Operationen und erzeugten Features werden in einem CSG-Baum abgespeichert. Dieser kann auch mit der Schaltfläche **Modeler Feature-Manager** in der Primär-Symbolleiste betrachtet und bearbeitet werden.

Wesentliche Vorteile des Modelers gegenüber den Standard-3D-Funktionen von MicroStation:

- Änderbarkeit der Abmessungen der erzeugenden Grundkörper eines Objektes
- Verwendung von Features
- Nachträgliche Änderbarkeit von Abmessungen, Eigenschaften und Lage von Features
- Manipulation des CSG-Baumes
- Parametrische Konstruktion in Abhängigkeit von globalen Variablen

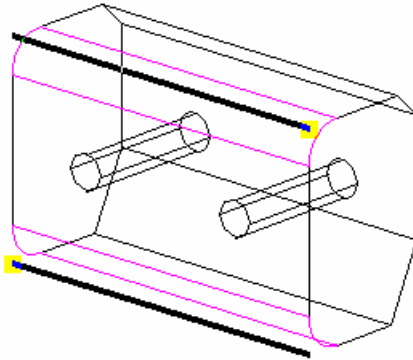
MODELER – FEATURES ÄNDERN – FEATUREMANAGER

Nun sollen die Features des abgerundeten und mit Bohrungen versehenen Quaders geändert werden.



Ändern der Abrundungsradien:

Mit dem Werkzeug **Parametrisches Volumen oder Feature ändern (M2b-1)** modifizieren wir die Abrundungsradien. Wir wählen die abgerundeten Kanten durch Anklicken aus.



Alle bei der Konstruktion in einem Schritt abgerundeten Kanten werden hervorgehoben. Im Einstellungsfenster kann nun der neue Radius eingegeben und nach Bestätigung geändert werden.

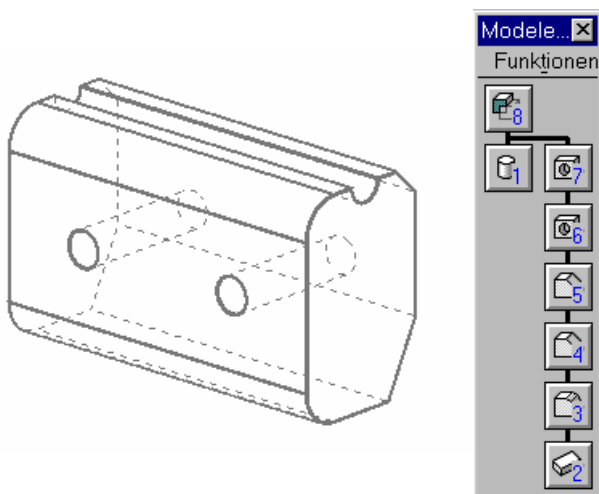
Hinweis: Analog können die Maße alle anderen Features geändert werden.



Feature Manager:

Mit Hilfe des Werkzeugs **Feature Manager** (in der **Primären Symbolleiste**) kann der CSG-Baum des Objektes angezeigt und modifiziert werden.

Um den Unterschied zwischen Feature und Boole'scher Operation zu zeigen, bringen wir in der oberen Seitenfläche des Quaders noch eine zylindrische Nut an, die wir als Boole'sche Differenz des Quaders und eines Drehzylinders konstruieren.



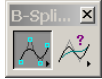

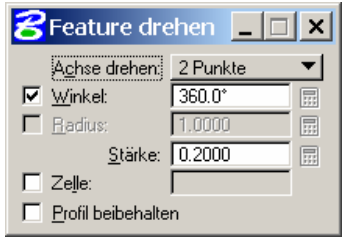
Der mit dem Feature-Manager angezeigte CSG-Baum zeigt den Unterschied zwischen der Booleschen Differenz und den vorher hinzugefügten Features.

Nach Anklicken eines Symbols des CSG-Baums wird das zugehörige geometrische Element in der Zeichnung hervorgehoben.

Klicken auf ein Symbol mit der rechten Maustaste öffnet ein Kontextmenü, das die Bearbeitung der einzelnen Features und Grundkörper erlaubt.

TISCH MIT GLÄSERN

Glas modellieren:

1. Erzeuge eine neue Datei mit dem Namen glaeser.dgn (keine Umlaute!)
2. Erzeuge ein neues 3D-Modell mit dem Namen glas1 und führe dort die folgenden Konstruktionen durch.
3. Erzeuge mit dem Werkzeug **Funktionen – B-Spline-Kurven – B-Spline-Kurve platzieren** den Meridian einer Drehfläche. 
4. Erzeuge mit dem Werkzeug **Funktionen – Feature-Modellierung – Feature drehen** ein Glas als Drehfläche mit den nebenstehenden Einstellungen  
5. Weise der Drehfläche (eventuell selbst definiertes) transparentes Material zu.
6. Erzeuge (eventuell später) weitere Modelle, wobei in einem Modell jeweils nur ein Glas modelliert wird.

Hinweis: Modifizierte Gläser (Drehflächen) eines gegebenen lassen sich leicht herstellen, indem man das erste Modell kopiert und dann mit den Griffen die Kontrollpunkte der Meridiankurve ändert.

7. Speichere die Zeichnung ab.

Tisch modellieren:

8. Erzeuge eine neue Datei mit dem Namen tisch.dgn und modelliere einen Tisch.

Gläser auf den Tisch stellen

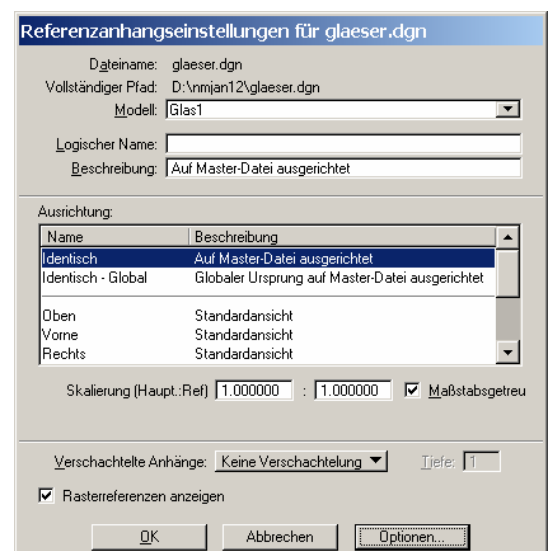
9. Im Folgenden werden wir mit Verwendung von 3D-Referenzen die in der Datei glaeser modellierten Objekte einbinden und auf dem Tisch platzieren.

Mit dem Werkzeug **Referenz – Referenz anhängen** wird das Modell Glas1 aus der Datei glaeser „identisch“ an die Datei tisch angehängt.

Hinweise:

*Wird ein Objekt als Referenz eingebunden, so werden die Materialeinstellungen der Ursprungsdatei verwendet. Wird dieses Objekt kopiert oder durch **Zusammenfügen mit Master** in die Datei kopiert, so gehen die Materialzuweisungen verloren. Man kann diese aus der Datei glaeser holen, indem man im Material-Editor mit Tabelle öffnen die Materialtabelle von glaeser öffnet und unter tisch.mat abspeichert. Macht man das mit mehreren Dateien muss man auf Konflikte achten.*

Wird ein Objekt als Referenz eingebunden, so können Änderungen in der eingebundenen Datei vorgenommen werden, die dann - eventuell nach aktualisieren - automatisch in der Datei tisch übernommen werden. Dies erleichtert etwa die Entwicklung von Bauteilen mit standardisierten Schnittstellen.



Gerendertes Bild erzeugen

10. Definiere geeignetes Licht

11. Rendere die Szene mit der Methode Ray-Tracing

Hinweis: Überprüfe folgende Einstellungen:

*Überprüfe ob bei „**Attribute zum Rendern der Ansicht**“ die Optionen „Textur“, „Schatten“, „Transparenz“ angehakt sind.*

*Überprüfe die Einstellungen im Einstellungsfenster „**Ray Tracing**“ insbesondere ob die Optionen „Schatten“, „Reflexionen“, „Transparenz“ aktiviert sind, die Einstellung für „Naturgetreue Beleuchtung“ und Antialiasing (auf höhere Qualität stellen und Antialiasing im übergeordneten Dialogfenster „**Rendern**“ aktivieren).*

12. Speichere mit dem Werkzeug **Extras – Bild – Speichern** ein gerendertes Bild (in ausreichend hoher Auflösung) ab.

Szene weiter ausgestalten

13. Gestalte weitere Objekte und stelle sie auf den Tisch bzw. gestalte die Umgebung des Tisches.

ANIMATION - GRUNDLAGEN

Bei CAD-Programmen gibt es meist folgende Verfahren zur Erzeugung von Animationen:

- Festlegung von Schlüsselszenen, aus denen das Programm automatisch Zwischenbilder generiert („Keyframing“)
- Festlegung von Objektwegen und/oder Kamerafahrten
- Skriptgesteuertes Animieren unter Einsatz mathematischer Formeln – dabei wird die Position aller beweglichen Objekte durch Parameterdarstellungen beschrieben

Zusätzlich gibt es in MicroStation noch die Möglichkeit, Licht- und Materialeigenschaften einer Änderung zu unterwerfen.

Der Ablauf zur Erzeugung von Animationen sieht im Groben etwa so aus:

- 1) Festlegung von Akteuren und deren Bewegungen durch Angabe von Schlüsselpositionen, Pfaden oder parametrischen Bewegungsgleichungen
- 2) Definition der Kamerastandpunkte und –zielpunkte
- 3) Kontrolle und Aufzeichnen der Animation

Die Werkzeuge zur Erstellung einer Animation befinden sich im Werkzeugkasten **Funktionen - Visualisierung – Animation**

mit den Werkzeugladen:



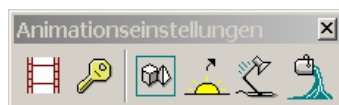
Animateure



Kameras



Einstellungen

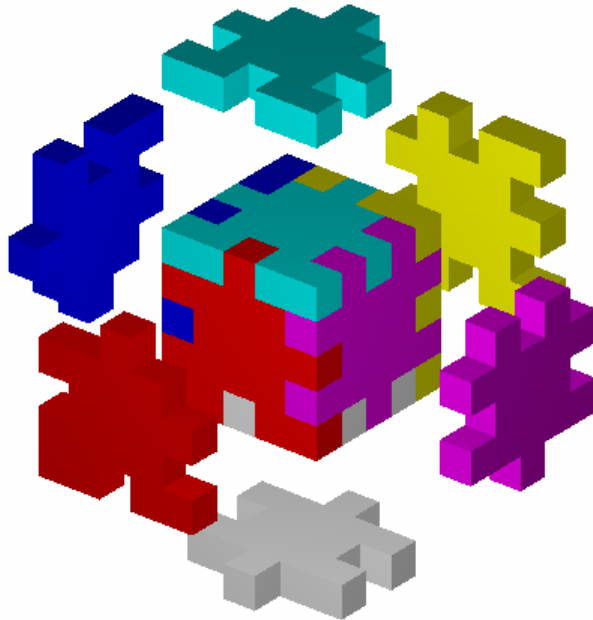


Vorschau



ANIMATION - KEYFRAMING

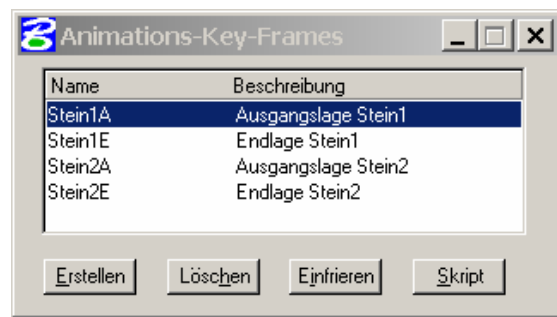
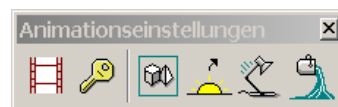
Erzeuge einen Film, der die Lösung für das Zusammensetzen eines "Moosgummiwürfels" zeigt. Verwende dabei die Datei ANIMATION_MOOSGUMMI.



Schritt 1: Herstellen der Schlüsselszenen

Wir verlagern jedes der zu bewegendenden Objekte in seine Anfangs- und Endlage (Zwischenlagen sind auch möglich), und speichern diese Schlüsselbilder ab.

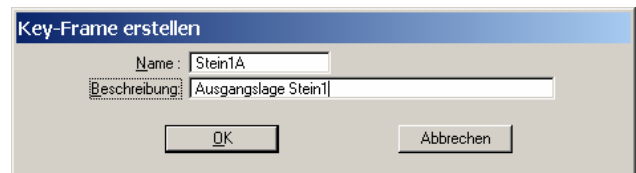
Dies geschieht mit dem Werkzeug **Animations-Key-Frames** (Animation 2a-2).



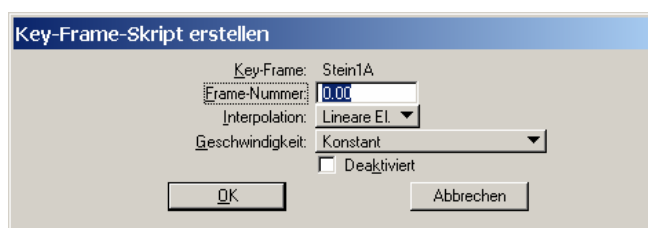
Wir wählen die Schaltfläche „Erstellen“, identifizieren das (in die richtige Position gebrachte) Objekt und vergeben Namen und Zusatzinformationen für das Schlüsselbild.

Analog verfahren wir mit Stein 2 etc.

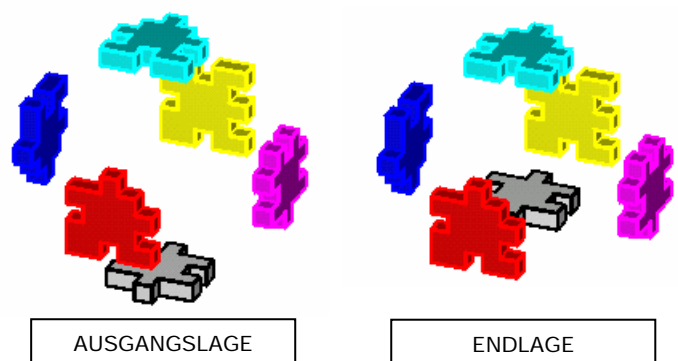
Bemerkung: Damit die Animation funktioniert, müssen zum Erstellen von Schlüsselbildern eines bewegten Objektes, verschiedenen Positionen **desselben** Objektes identifiziert werden, es darf nicht ein Objekt und eine an anderer Stelle liegende Kopie des Objektes identifiziert werden



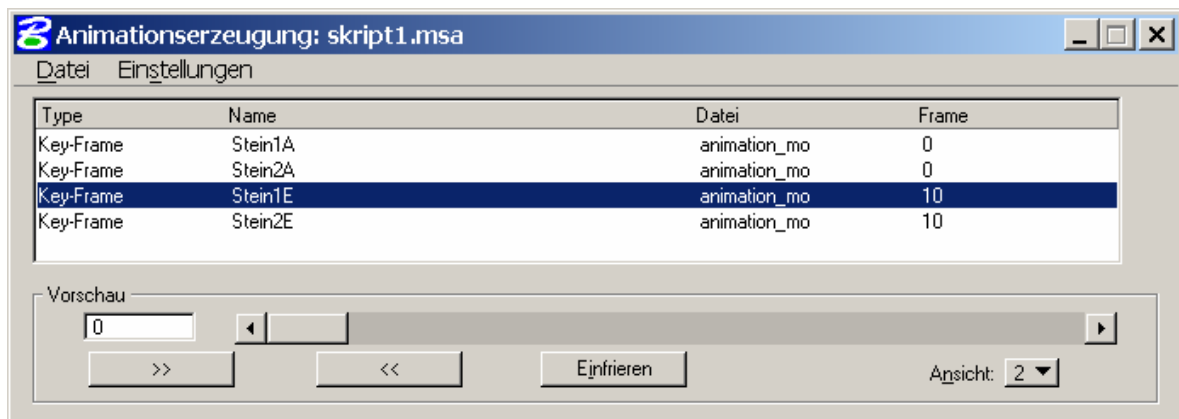
Schritt 2: Schlüsselbilder in ein Skript einfügen



Wir markieren im Dialogfenster **Animations-Key-Frames** ein Schlüsselbild und tragen es – durch Anklicken der Schaltfläche „Skript“ – in das aktuelle Animationsskript ein. Dabei muss die Nummer des Bildes im Feld „Frame-Nummer“ eingegeben werden (z.B. Ausgangslage = 0 und Endlage = 10).



Bemerkung: Einer Frame-Nummer (also einem Zeitpunkt der Animation) können mehrere Schlüsselbilder zugeordnet sein. In diesem Beispiel etwa Stein1, Stein2 etc. Die durch verschiedene Schlüsselbilder der jeweiligen Objekte definierten Bewegungen laufen dann gleichzeitig ab.

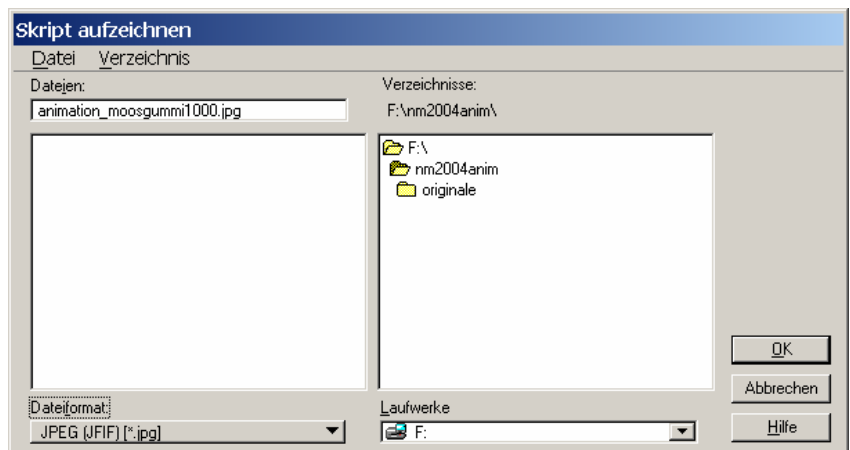
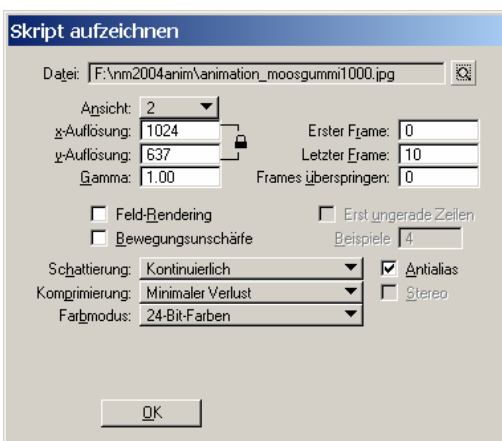


Mit dem Dialogfenster Animationserzeugung (Animation 2a-1) kann man alle Skripteinträge und die zugehörigen Bilder betrachten und bearbeiten, sowie eine Vorschau der Animation anzeigen.

Schritt 3: Aufnahme der Animation



Im Werkzeugkasten **Animationsvorschau** befinden sich ebenfalls Werkzeuge zur Kontrolle sowie zur Aufnahme der Animation.



Nach Betätigen der „Aufnahmetaste“ kann man im dazugehörigen Dialogfenster Qualität, Speicherort usw. angeben. Es werden dann die Frames einzeln berechnet und

- bei Angabe eines Dateiformates **Einzelbilder** im gewählten Format (z.B. jpeg) versehen mit einer laufenden Nummer am Ende des Dateinamens abgespeichert.
- oder beim expliziten Eintippen eines Dateinamens samt Erweiterung (avi, fli, flc) **eine Video-datei** im entsprechenden Format abgespeichert. (Dies ist möglich, obwohl in der Liste der Dateiformate, diese Formate nicht angeführt sind.)

Hinweis:

Die berechneten Einzelbilder können auch über den Menüpunkt **Extras – Bild – Animation** abgespielt, bearbeitet und in einer Animationsdatei gespeichert werden, die etwa mit dem Windows Media Player abspielbar ist. Dies geschieht mit dem Menüpunkt Datei speichern und Auswahl des Formats AVI. Ein weiteres empfehlenswertes Dateiformat ist FLI, eine MicroStation internes Mediendateiform – diese Dateien können anschließend weiter bearbeitet werden.

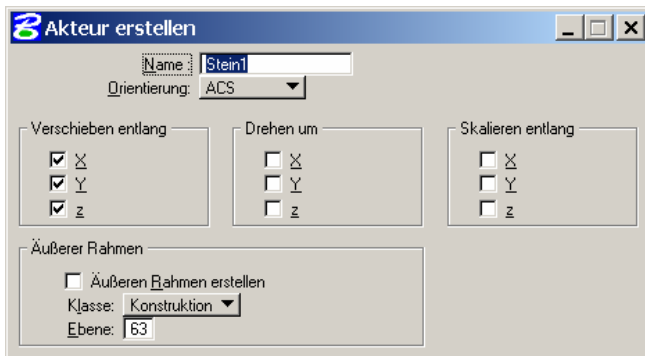
Zur Erzeugung komprimierter AVI-Dateien oder Animationsdateien in anderen Formaten empfiehlt sich die Verwendung anderer Software um diese aus den in MicroStation berechneten Einzelbildern zu generieren.

Bei dieser Art von Animation hat man keinen Einfluss auf Zwischenlagen. Dies führt meist zu unrealistischen Situationen (vgl. MOSSGUMMI3.AVI). Was fällt auf?

ANIMATION – DEFINITION VON PFADEN

Erzeuge eine Animation, die den Zerfall des Würfels zeigt und verwende dabei beliebige Bahnkurven. Sieh dir dazu den Film MOOSGUMMI 4.AVI an.

Schritt 1: Definition der Akteure

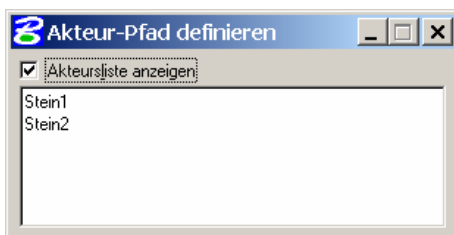


Mit dem Werkzeug „Akteur erstellen“ (Animation 1a-1) wird ein Objekt benannt, dann gewählt und mit einem lokalen Koordinatensystem (zur Festlegung seiner Bewegungsfreiheit) versehen. Sollen mehrere Objekte derselben Bewegung unterworfen werden, wählen wir zuerst alle Objekte aus (<STRG>-Taste) und rufen erst dann das Werkzeug „Akteur erstellen“ auf.

Bemerkungen:

- Das beim Erzeugen des Akteurs mit diesem verknüpfte lokale Koordinatensystem kann parallel zum internen absoluten Koordinatensystem oder zum aktuellen Hilfskoordinatensystem (ACS) ausgerichtet werden.
- Die Position des Akteurs kann mit dem Werkzeug „Akteur Manipulieren“ (Animation 2a-3) geändert werden. Dies kann man etwa auch dazu benutzen um ein Objekt mit dem man einen Akteur verknüpft hat in die gewünschten Positionen für Schlüsselbilder zu bringen.

Schritt 2: Angabe des Pfades

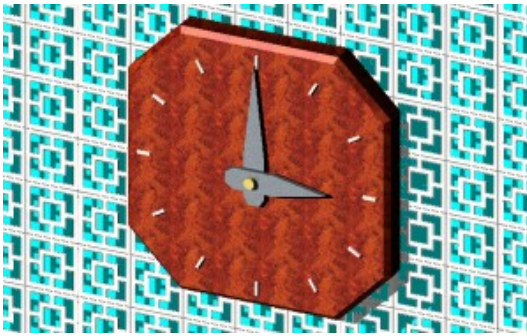


Nun geben wir die Bahn des Objektes – am besten als Konstruktionselement - vor (Linien, Kreise, B-Splinekurven, ...) und wählen das Werkzeug „Akteur-Pfad definieren“ (Animation 1a-6) aus. Im Dialogfenster wählen wir den gewünschten Akteur durch einen Doppelklick aus und klicken anschließend den Endpunkt der Bahnkurve an - Achtung der angeklickte Punkt gibt auch den Endpunkt der Animation an!

Schritt 3: Kontrolle und Aufnahme der Animation

Wie bei der Erstellung einer Animation aus Schlüsselszenen.

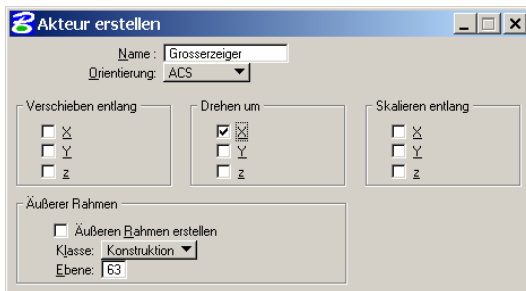
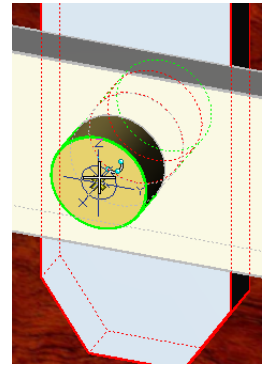
ANIMATION MIT PARAMETERGLEICHUNGEN



Eine analoge Uhr soll die Zeitspanne zwischen 15:00 und 18:00 im Zeitraffer wiedergeben. Verwende dazu die Datei *WIE DIE ZEIT VERGEHT.DGN* inklusive Materialdatei.

Schritt 1: Festlegung der Akteure

Da wir die beiden Zeiger um eine horizontale Achse drehen wollen, legen wir mit dem Werkzeug „Akteur erstellen“ (Animation 1a-1) den großen und den kleinen Zeiger als Akteure fest. Die x-Achse des lokalen Koordinatensystems des Akteurs muss dabei jeweils in der Drehachse der drehzylinderförmigen Nabe liegen. Weiters erlauben wir den beiden Akteuren nur eine Drehung um die x-Achse. Diese Einstellung kann auch im Nachhinein mit dem Werkzeug „Akteur ändern“ (Animation 1a-4) vorgenommen werden.



Schritt 2: Skript für die Akteure schreiben

Nun legen wir für jeden Zeitpunkt die Lage der beiden Zeiger mit Hilfe einer Parameterdarstellung in einem Skript fest. Dabei dient als Parameter die Variable „frame“. Nach Aufruf des Werkzeugs „Akteur-Skript“ (Animation 1a-7) wählen wir den Akteur aus der Liste mit einem Doppelklick aus und geben folgende Werte ein:

Grosserzeiger	Begin Frame:	0	End Frame:	360	X-Rotation:	$-3 * \text{frame}$
Kleinerzeiger	Begin Frame:	0	End Frame:	360	X-Rotation:	$-0.25 * \text{frame}$

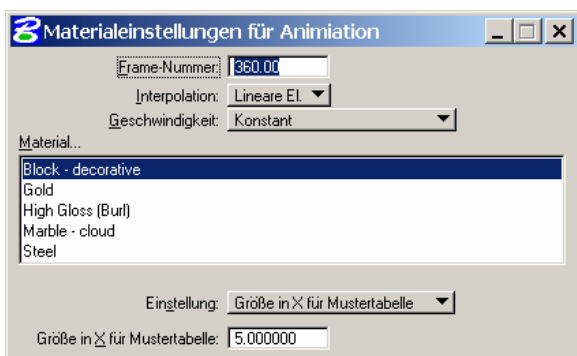
Dadurch legen wir 360 Bilder fest; der große Zeiger wird insgesamt um $3 \times 360^\circ$ gedreht – also genau 3 Umrundungen – und der kleine Zeiger bewegt sich um genau $0.25 \times 360^\circ = 90^\circ$ weiter. Das negative Vorzeichen ist notwendig, da der Uhrzeigersinn mathematisch negativ orientiert ist.

Schritt 3: Kontrolle und Aufnahme

Wie bei der Erstellung einer Animation aus Schlüsselszenen.

Schritt 4 : Änderung von Material und Kamera

Während die 3 virtuellen Stunden ablaufen, soll sich die **Hintergrundtextur** ändern.

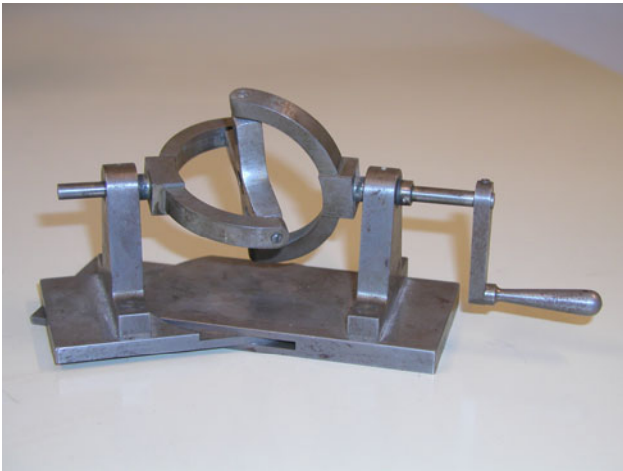


Im Dialogfenster „Materialeinstellungen für Animation“ (Werkzeug „Materialien“ (Animation 2a-6)) können jene Materialien ausgesucht werden, bei denen eine Eigenschaftsänderung durchgeführt werden kann. Im Feld „Frame-Nummer“ wird jene Bildnummer eingegeben, ab der die Einstellung gilt; unter „Einstellung“ kann man die veränderbaren Eigenschaften auswählen.

Hinweis: Besteht eine Textur aus einer nummerierten Bildfolge, dann können diese Bilder ebenfalls zur Animation verwendet werden

Zusätzlich kann auch die **Kamera** durch Angabe von Schlüsselbildern, durch Festlegung von Pfaden bzw. durch Beschreibung mittels Parametergleichungen bewegt werden.

ANIMATION EINES KARDANGELENKS



Foto

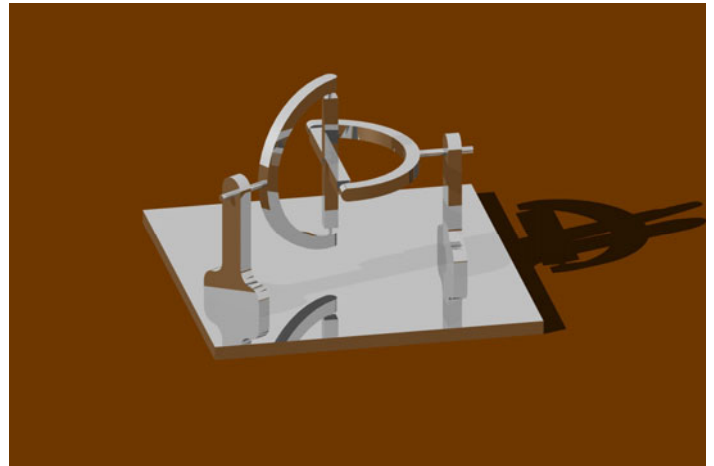


Bild erzeugt mit MicroStation

Ein Kardangelenke dient zur Übertragung einer Drehung um eine Achse a auf eine Drehung um eine schneidende Achse a . Das Verhältnis von Abtriebswinkel zu Antriebswinkel ist allerdings nicht konstant.

Konstruktion der Geometrie

Wir konstruieren zuerst das „Gerüst des Getriebes“ bestehend aus einem rechtwinkligen Kreuz, das in zwei Halbkreisbögen eingespannt ist (siehe Figur).

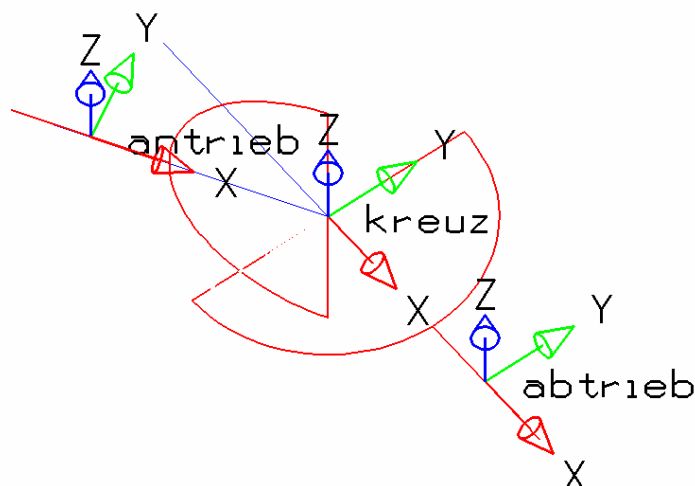
Dann gestalten wir das Getriebe aus, indem wir dieses Gerüst zu Quader- und Zylinderteilen „verdicken“.

Am besten konstruiert man zuerst einen Bogen und erzeugt dann den zweiten durch Kopieren und Anwenden geeigneter Transformationen.

Diese Konstruktionen sind in der Datei kardan_angabe.dgn vorbereitet.

Festlegen von Hilfskoordinatensystemen

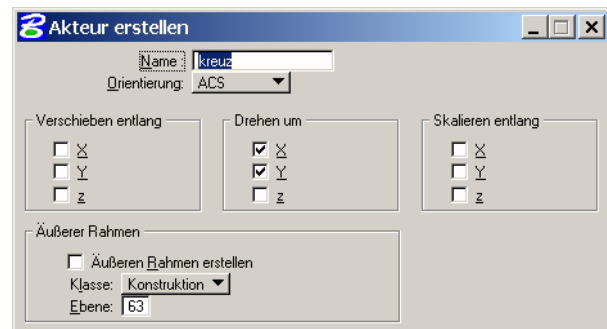
Wir definieren drei Hilfskoordinatensysteme gemäß untenstehender Figur.



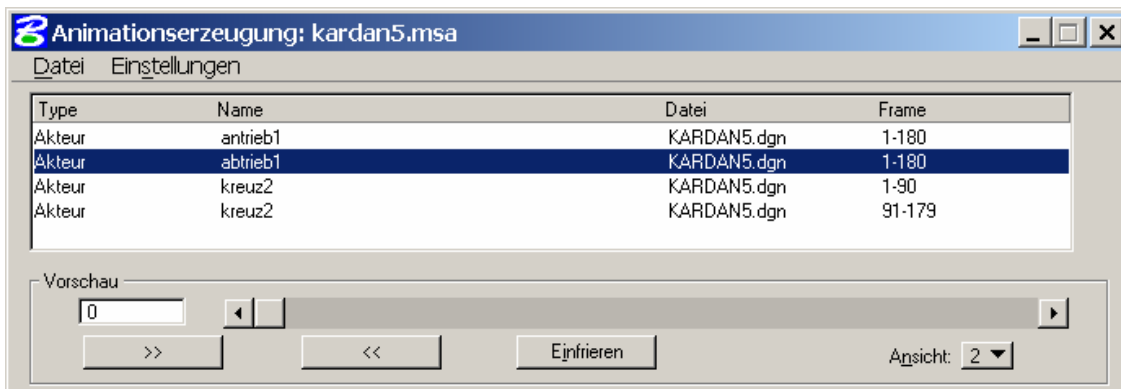
Definition der Akteure

Für den Antrieb und Abtrieb definieren wir je einen Akteur mit Ausrichtung „ACS“, der nur um die x-Achse drehbar ist.

Für das Kreuz definieren wir einen Akteur mit Ausrichtung „ACS“, der um x-Achse und y-Achse drehbar ist.



Mit dem Werkzeug **Akteur Skript erstellen** erzeugen wir nun ein Animationsskript, wobei wir die Drehwinkel gemäß dem Blatt „Berechnung zum Kardangelenk“ verwenden. Als Antriebswinkel wird die Variable „frame“ verwendet. Wir machen folgende Einträge:



Antrieb:

x-Drehung:	frame
y-Drehung:	0,0
z-Drehung:	0,0

Abtrieb:

x-Drehung:	$\text{atan}(\tan(\text{frame}) \cdot \cos(30))$
y-Drehung:	0,0
z-Drehung:	0,0

Kreuz:

x-Drehung:	$\text{atan}(\tan(\text{frame}) \cdot \cos(30))$
y-Drehung:	$\text{acos}(\sqrt{\text{pow}(\sin(\text{frame}), 2) \cdot \text{pow}(\cos(30), 2) + \text{pow}(\cos(\text{frame}), 2)})$
z-Drehung:	0,0

Für den Antriebswinkel zwischen 91 und 180 Grad ist beim Kreuz bei der y-Drehung das andere Vorzeichen als beim Winkel zwischen 0 und 180 Grad zu wählen.

Berechnung zum Kardangelenk

Ein Kardangelenk dient zur Übertragung einer Drehung um eine Achse a auf eine schneidende Achse b . Dabei ist allerdings das Übersetzungsverhältnis nicht konstant.

Der Koordinatenursprung O sei der Schnittpunkt von a und b mit den Richtungsvektoren

$$\mathbf{a} = \begin{pmatrix} \sin \gamma \\ -\cos \gamma \\ 0 \end{pmatrix} \text{ bzw. } \mathbf{b} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

d.h. der Winkel der Achsen hat das Maß γ .

Die Übertragung der Drehung geschieht durch ein rechtwinkeliges Kreuz mit Mitte im Schnittpunkt der Drehachsen a und b (Koordinatenursprung), wobei ein Schenkel PQ normal zu a steht und sich um a dreht, während sich der andere Schenkel $RS \perp PQ$, der normal auf b steht um b dreht.

In der Ausgangslage sei PQ parallel zur z-Achse. Dann liest man aus Aufriss und Seitenrissen, in denen a bzw. b projizierend erscheinen, ab, dass der zu einem Drehwinkel α um a (Antriebswinkel) gehörige Drehwinkel β um b (Abtriebswinkel) durch

$$\tan \beta = \tan \alpha \cos \gamma$$

bestimmt ist.

Eine zum Antriebswinkel α gehörige Lage P, Q, R, S des Kreuzes erhalten wir folgendermaßen: Durch eine Drehung um die y-Achse durch den Abtriebswinkel β bringen wir die Ausgangslage $P_0Q_0R_0S_0$ in die Lage $P_1Q_1R_1S_1$ und drehen dann das Kreuz um die Gerade $[R_1S_1]$ durch den (noch zu berechnenden) Winkel δ in die Lage $PQRS$.

Im folgenden bezeichne \mathbf{p} den Ortsvektor von P, usf.

δ ist der Winkel zwischen \mathbf{p} und \mathbf{p}_1 .

\mathbf{p} entsteht aus \mathbf{p}_0 durch eine Drehung um die y-Achse durch α und eine anschließende Drehung um die z-Achse durch γ und lautet daher:

$$\mathbf{p} = \begin{pmatrix} \sin \alpha \cos \gamma \\ \sin \alpha \sin \gamma \\ \cos \alpha \end{pmatrix}$$

\mathbf{p}_1 entsteht aus \mathbf{p}_0 durch Drehung um die y-Achse durch β und lautet daher:

$$\mathbf{p}_1 = \begin{pmatrix} \sin \beta \\ 0 \\ \cos \beta \end{pmatrix}$$

Da die Vektoren normiert sind erhalten wir den Winkel δ aus

$$\cos \delta = \begin{pmatrix} \sin \alpha \cos \gamma \\ \sin \alpha \sin \gamma \\ \cos \alpha \end{pmatrix} \cdot \begin{pmatrix} \sin \beta \\ 0 \\ \cos \beta \end{pmatrix}$$

Einsetzen und Vereinfachung mit Unterstützung eines Computeralgebraprogramms liefert:

$$\delta = \arccos(\sqrt{(\sin^2 \alpha) \cos^2 \gamma + \cos^2 \alpha})$$

KAMERAFAHRTEN

Aktionskameras (Augpunkte der einzelnen Perspektiven) und deren Zielpunkte sind spezielle Akteure und können daher mittels Schlüsselbilder (keyframes), Pfaden und Skripten kontrolliert werden.

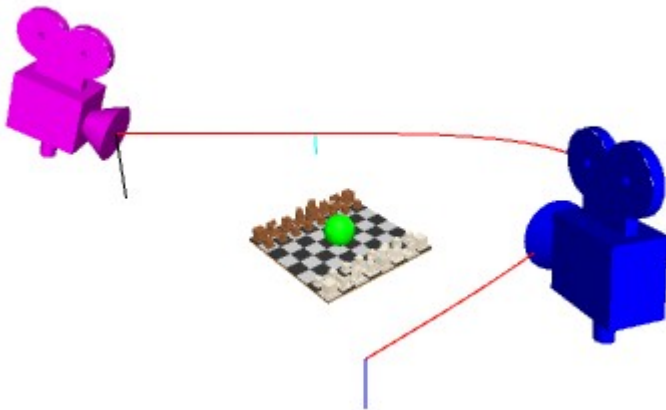
Aufgabe: Erstelle mit Hilfe der Datei KAMERAFAHRT.DGN eine virtuelle Schachpartie, bei der sich die Figuren bewegen. Gleichzeitig soll diese Szene von einer beweglichen Kamera aus verschiedenen Standpunkten gefilmt werden.



Mit dem Werkzeug **Animationskamera erstellen** (1) kannst du entweder aus der vorliegenden Ansicht oder durch Eingabe von Kamerastandpunkt bzw. -zielpunkt eine Animationskamera erzeugen. Diese wird in der Szene als Filmkamera in der aktiven Ansichtsebene angezeigt. In unserem Beispiel sind die Kamerastandpunkte als obere Endpunkte senkrechter Strecken in der Ansichtsebene „Kamerastandpunkte und Zielpunkte“ vorgegeben. Als Zielpunkt dient uns jeweils der höher liegende Endpunkt der blauen Strecke in der Mitte des Schachspiels.

Mit dem Werkzeug **Kameraansicht** (4) überprüfen wir, ob die gewählte Ansicht unseren Vorstellungen entspricht.

Hinweis: Bei Wahl eines Objektivwinkels von 0 Grad zeichnet die Kamera eine Parallelprojektion auf.

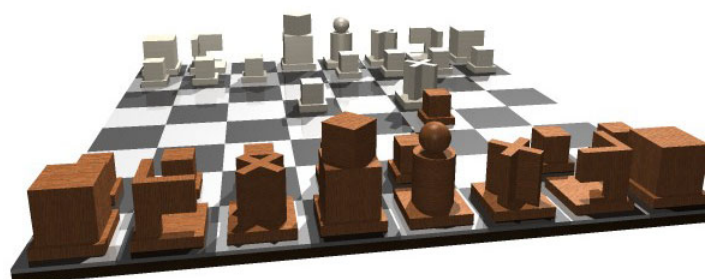
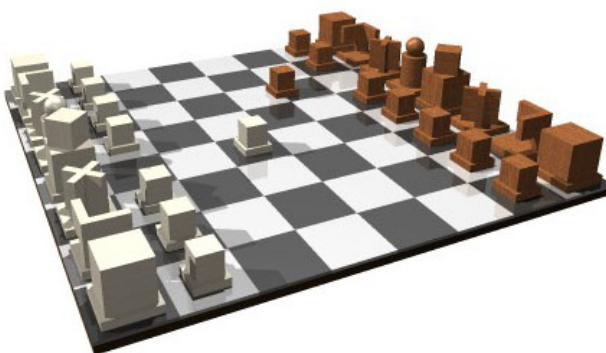


Die so erstellte Kamera ist nun als Akteur verfügbar und kann wie jeder andere Akteur behandelt werden. Wir legen daher eine B-Splinekurve durch die Kamerastandpunkte um die Kamerafahrt längs eines Pfades festzulegen und legen mit dem Werkzeug **Akteur-Pfad definieren** (Animation 1b-6) den Kameraweg fest (Achtung: Den Endpunkt der B-Splinekurve snapen!).

Zum Aktivieren der Kamera müssen wir mit dem Werkzeug **Kameraskript erstellen** (3) die Bildnummer angeben, ab der die Kameraeinstellung gültig ist.

Damit die Kamera immer in die Schachbrettmitte zeigt definieren wir noch einen Zielpunkt (Werkzeug **Zielobjekt erstellen** (5)) und setzen diesen mittels Skript (6) für die gesamte Filmdauer aktiv.

Nun fehlt nur noch die Animation der Schachfiguren, die in gewohnter Weise vor sich geht.

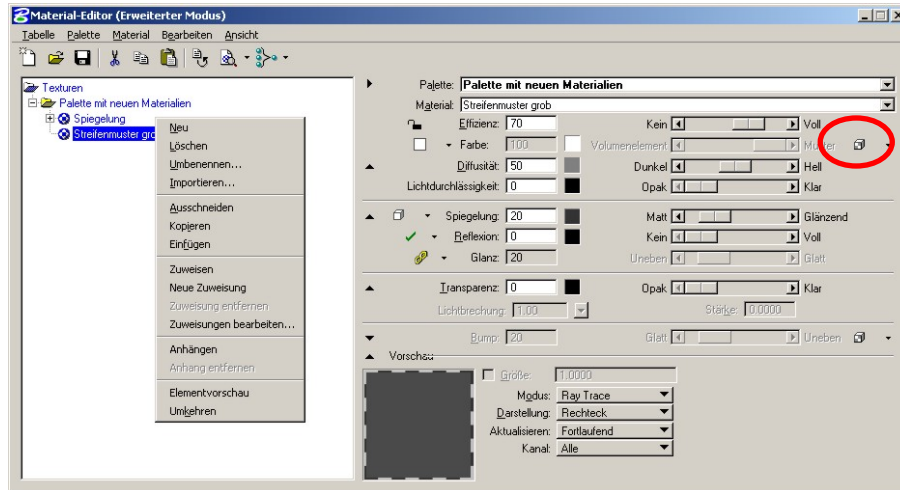


Hinweis: Will man mit Kameraschlüsselbildern arbeiten, so muss man zuerst Festansichten, die als Schlüsselansichten dienen, festlegen.

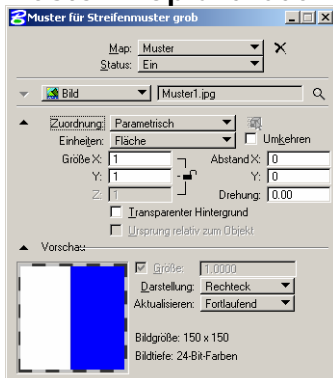
Tipp: Bei beweglichen Ansichten gibt es manchmal Schwierigkeiten mit der Darstellungstiefe; schalte daher in den Ansichtsattributen Abschneiden hinten und vorne aus.

EIGENE MATERIALIEN

Zur Erzeugung eigener Materialien mit dem Materialeditor erstellen wir zuerst eine eigene Palette (**Palette – Neu**) und speichern diese unter einem entsprechenden Namen ab (**Palette – speichern unter**). Nun können wir (**Material – Neu**) mit dem Entwurf eines Materials beginnen.



Ein Rechtsklick auf das Material öffnet dann ein kontextsensitives Menü, mit dessen Hilfe wir z.B. die Bezeichnung des Materials editieren können. Zum Zuordnen eines Bildes klicken wir auf das Symbol **Muster-map** und laden die gewünschte Grafik.



Die Werte für die Diffusität, Lichtdurchlässigkeit und Spiegelung wählen wir so, dass deren Summe (=Effizienz) kleiner als 100% ist. Die Einstellungen für Reflexion und Glanz wählen wir beliebig und die Werte für die Transparenz und Farbe setzen wir auf 0.

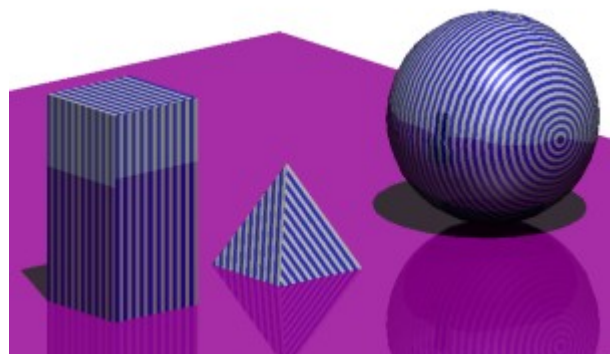
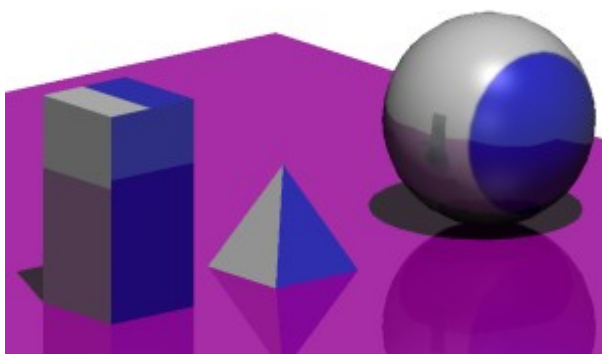
Durch Anklicken des Symbols **Muster-map** öffnen wir das Fenster zum Bearbeiten des Musters. Hier stellen wir – falls nicht schon vorhanden – die Einstellungen analog zur Abbildung ein.

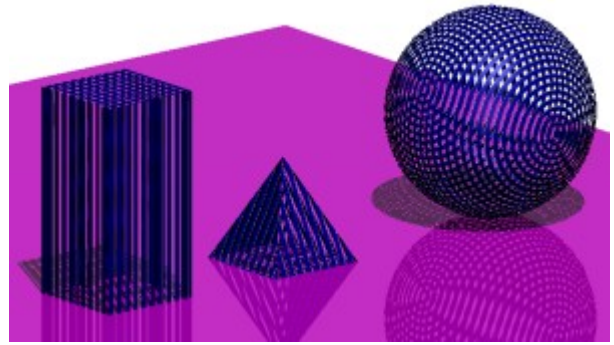
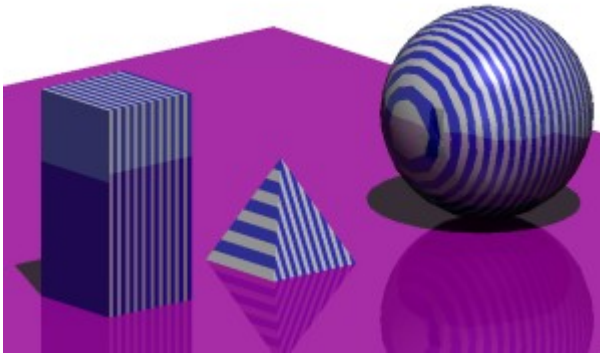
Experimentiere nun mit den angegebenen Werten und schau dir die Auswirkungen auf eine einfache geometrische Szene (Quader, Pyramide, Kugel) an. Beachte dabei, dass bedingt durch die Konstruktion der Objekte der „Ursprung“ der Textur bereits festgelegt ist.

Änderung der Einheiten: Werden Flächeneinheiten verwendet, dann wird das Muster (1/Größe)-fach auf das Objekt projiziert und zwar unabhängig von der Objektgröße. Verwendest du hingegen Haupteinheiten, dann wird das Bild auf ein Rechteck der Größe (X,Y) skaliert und sooft wie möglich auf das Objekt projiziert.

Zuordnung: Bei der Einstellung **Planar** wird das Muster auf jede Seite des Objekts projiziert, ohne auf die benachbarten Flächen Rücksicht zu nehmen. Die Einstellungen **Parametrisch** und **Höhenangaben drapieren** liefern prozedurale Texturen, die das Objekt als Ganzes berücksichtigen.

Aufgabe: Erzeuge folgende Szenen durch Erstellen und Ändern einer einfachen Textur.





Tipp: Falls du transparenten Hintergrund aktivierst, dann wird die transparente Farbe durchsichtig dargestellt – dies ist wichtig bei der Erstellung von Materialien mit Bildern von Personen oder Bäumen.

Untersuche anhand der Aufstellung auch, wie sich eine Drehung des Musters bei den diversen „Zuordnungseinstellungen“ auswirkt.

Im Materialeditor kannst du weitere Eigenschaften des Materials ändern:

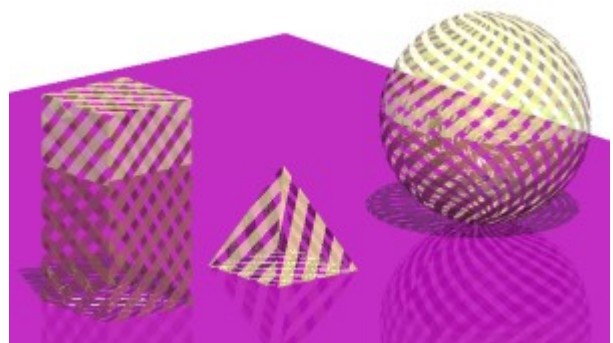
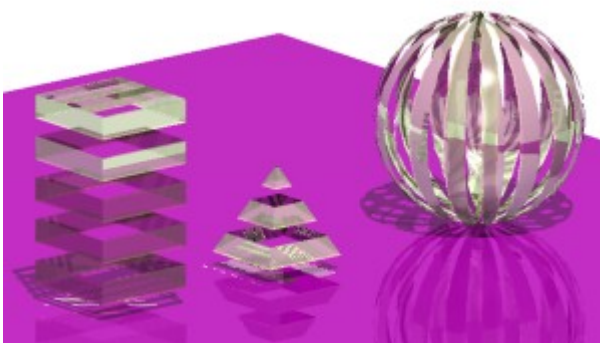
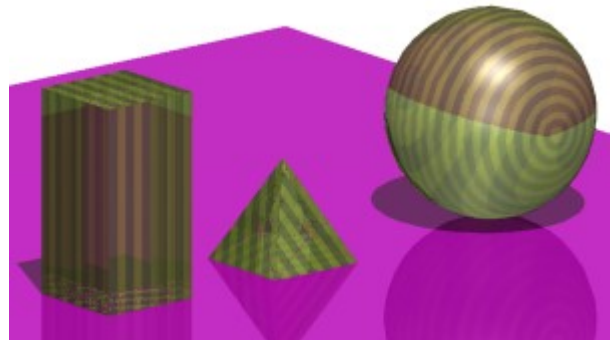
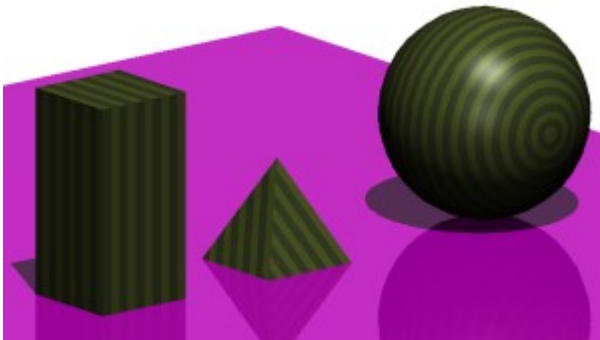
Farbe:

Hier stellst du die Mischung aus Objektfarbe und Farbe des Originalmusters ein. Du kannst aber auch eine eigene Farbe (benutzerdefiniert) zum Mischen verwenden.

Transparenz:

Je höher der Wert hier gewählt wird umso durchsichtiger ist das erzeugte Material. Denk daran, die Darstellung der Transparenz über **Einstellungen – Rendern – Ansichtsattribute** deiner Ansicht zuzuweisen. Bei aktivierter Transparenz kannst du auch einen Brechungskoeffizienten angeben oder aus der Liste auswählen.

Aufgabe: Erzeuge folgende Szenen ebenfalls durch weiteres Ändern deiner Textur.



VRML – X3D

VRML ist die Abkürzung für "Virtual Reality Modeling Language". Das ist ein Fileformat zur Beschreibung interaktiver und multimedialer 3D-Welten im Internet. Die erste von Silicon Graphics entwickelte Version VRML 1.0 wurde in Zusammenarbeit mit Sony und anderen Entwicklern zur Version VRML 2.0 erweitert. Neben geringen Veränderungen in der Sprachstruktur wurden insbesondere die Möglichkeit von Animationen, Multimediaeffekten und der Aufruf von externen Scripts hinzugefügt. Diese Version wurde in nur geringfügig geänderter Form 1997 als internationaler Standard ISO/IEC 14772-1:1997 beschlossen und wird kurz als VRML97 bezeichnet. Mit ISO/IEC:14772-2:2002 wurde dieser Standard um das „External authoring interface (EAI)“ erweitert, das Schnittstellen von VRML zu anderen Programmiersprachen regelt. Derzeit durchläuft eine Weiterentwicklung X3D („Extensible 3D“) den Standardisierungsprozess. Die weitere Entwicklung wird vom Web3D Consortium koordiniert. Die jeweils aktuellen Informationen findet man auf <http://www.web3d.org>. Da X3D insbesondere wegen des noch laufenden Standardisierungsprozesses noch nicht so verbreitet ist, behandeln wir hier nur VRML.

In VRML erfolgt die Beschreibung durch eine Hierarchie von sogenannten Knoten (Nodes), die in Feldern wieder Knoten enthalten. So ein Knoten beschreibt ein Geometrieobjekt als eine Gruppe von Objekten (die dann einer Transformation unterworfen werden können), Material- und Oberflächeneigenschaften sowie Sound. Seit VRML 2.0 gibt es auch Script-Knoten, die Scripts ausführen und externe Scripts aufrufen können. Um Animationen ausführen zu können, gibt es Knoten Ereignisse melden können, etwa Sensoren oder Timer. Diese werden dann mit einem ROUTE Befehl zu Eingabeparametern (etwa Größe eines Translationsvektors) anderer Knoten weitergeleitet.

Die Beschreibung wird in einer Textdatei mit der Erweiterung *.wrl („World“) abgespeichert. Ein ganz einfaches VRML-File, das einen Würfel generiert lautet:

```
#VRML V2.0 utf8

Shape {
  geometry Box {}

  appearance Appearance {
    material Material{
    }
  }
}
```

Die Werte für appearance sind zwar Leer, aber bei weglassen kommt es manchmal zu Darstellungsschwierigkeiten.

Die Möglichkeit den Standpunkt des Betrachters zu ändern und das Schattieren (Rendern) erfolgt dann automatisch durch einen VRML-Viewer, ein Programm zum Betrachten von VRML-Welten, das als Plug-In in den WWW-Browser integriert ist.

Software und weitere Informationen:

Plug-In zum Anzeigen von VRML-Files:

Cortona: <http://www.parallelgraphics.com/products/cortona>

Weitere Informationen:

WEB3D Consortium: <http://www.web3d.org>

Tutorial:

Lighthouse Interactive VRML Tutorial: <http://www.lighthouse3d.com/vrml/tutorial>

Ein interessanter Link:

Virtuelle Modelle: <http://www.uni-kl.de/AG-Leopold/VRML>

Ein etwas komplexeres VRML-File, das mittels eines Sensors die Mausbewegung abfragt, und danach dementsprechend die Kugel verschiebt, sieht so aus (Groß- und Kleinschreibung muss beachtet werden!):

```
#VRML V2.0 utf8

# In der ersten Zeile steht die Information über das Datenformat
# Alles was in einer Zeile hinter # steht ist ein Kommentar

NavigationInfo{
    type "EXAMINE"
}

# Knoten der Eigenschaften und Geometrie einer Kugel festlegt
Shape {
    geometry Box {}
    # Definition von Farbe, Material etc.
    appearance Appearance{
        material Material{
            diffuseColor 1 0 0
        }
    }
}

# Knoten der eine Gruppe von Elementen zusammenfasst,
# die einer Transformation unterworfen werden können.
# Die Vergabe eines Namens mit DEF kugel
# ist für die folgende Animation nötig

DEF kugel Transform {
    children[
        Shape{
            appearance Appearance{
                material Material{
                    diffuseColor 0 0 1
                }
            }
            geometry Sphere{
                radius 1
            }
        }
    ]

    # Hier wird ein Sensor definiert, der die Bewegung der
    # Maus in einer Ebene registriert

    DEF sensor PlaneSensor{
        minPosition 0 -2
        maxPosition 0 2
        offset 0 2 0
    }

    ]

    # Transformation aller children Objekte des Knotens
    translation 0 2 0
}

# Die Mausbewegung wird registriert und dementsprechend wird
# die Translation geändert

ROUTE sensor.translation_changed TO kugel.set_translation
```


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 Randdarstellung 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>
    }
}

```

```

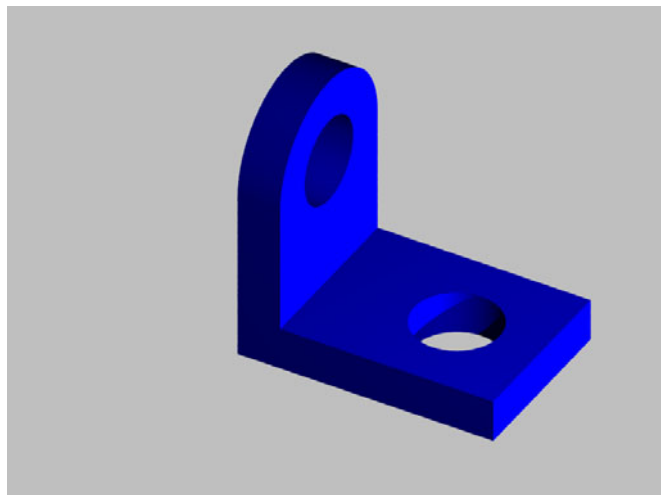
#declare cyl2 = object{
    cylinder {
        <0, -1, 0>, <0, 6, 0>,
1
        pigment {color Blue}
        translate <4, 0, 2>
    }
}

```

```

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>

GRAPHIKBEARBEITUNG

Grundsätzliches:

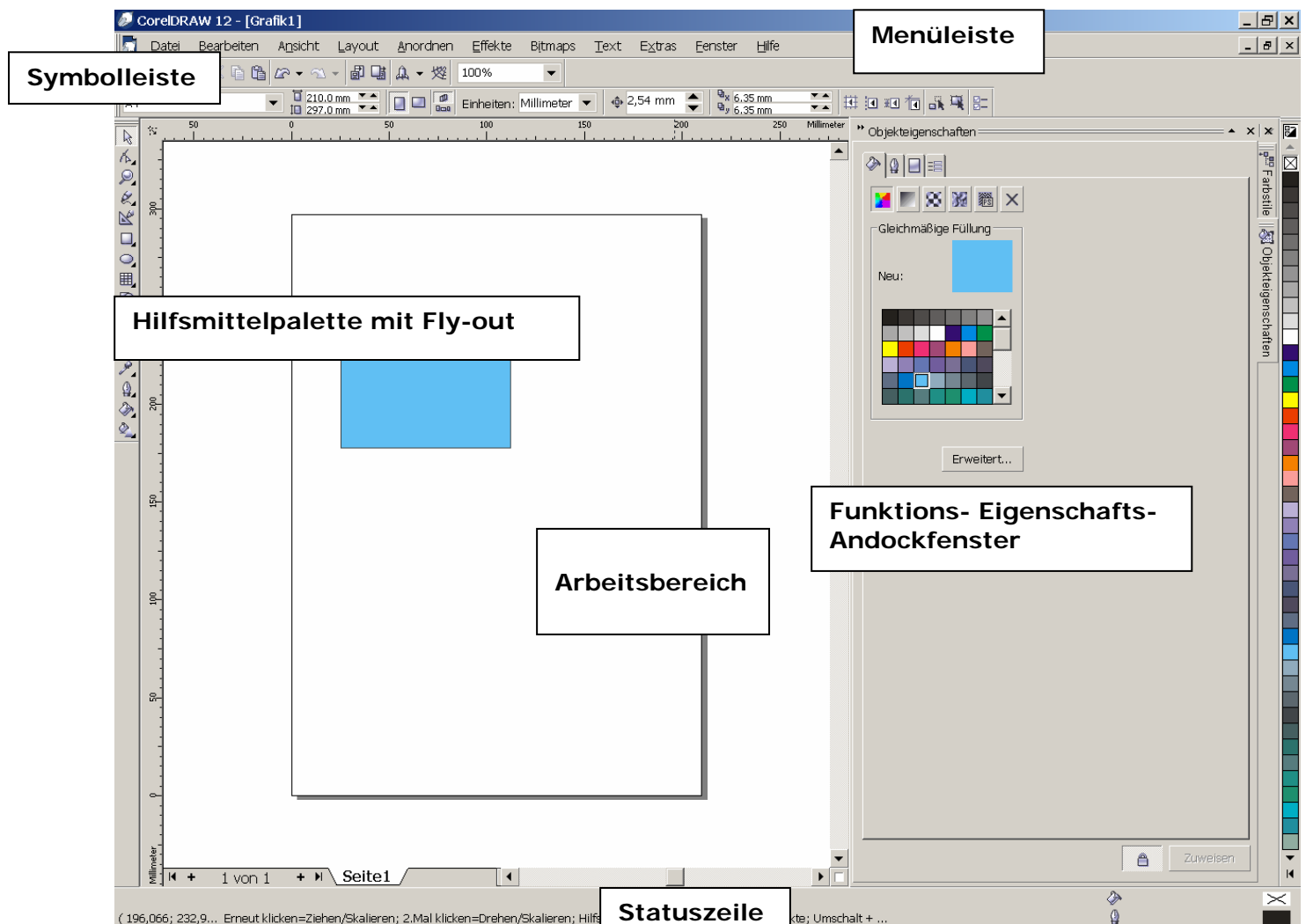
Zur Erstellung und Bearbeitung von Graphiken (Zeichnungen, Bilder und Text) benötigen wir grundsätzlich drei verschiedene Programme:

- Software zum Bearbeiten von **Vektorgraphiken** (z.B. Corel-Draw, Adobe Illustrator, Macromedia Freehand, Macromedia Fireworks)
- Software zum Bearbeiten von **Pixelgraphiken** (z.B. Corel-Photo-Paint, Adobe PhotoShop, Macromedia Fireworks, Gimp, IrfanView)
- Software zum **Abfotografieren** (Capture-Software) von Bereichen und Bildern am Schirm (z.B. Corel Capture, SnagIt, MWSnap (Freeware))

Hinsichtlich Free- und Shareware siehe etwa TUCOWS (tucows.univie.ac.at).

Viele Zeichen, Bearbeitungs und Textwerkzeuge sind bei Vektor- und Pixelbearbeitungsprogrammen ähnlich (etwa Corel-Draw und Corel-Photo-Paint) erzeugen jedoch entweder eine Vektorgraphik oder wandeln alles sofort in eine Pixelgraphik um.

Hier sollen kurz grundlegenden Werkzeug vorgestellt werden. Die konkreten Hinweise auf Werkzeuge und Menüs beziehen sich im Folgenden auf **Corel-Draw** und **Corel-Photo-Paint Version 12**.



Benutzeroberfläche:

Grundsätzlich ist die Benutzeroberfläche in Arbeitsbereich, Menüleiste, Symbolleisten, Statusleiste sowie Funktions- und Einstellungsfenster (eventuell Andockfenster) gegliedert. Zu den meisten Werkzeugen gibt es ein Einstellungsfenster (**Fenster – Symbolleiste – Eigenschaftsleiste**), das eventuell erst bei Bedarf geöffnet werden muss, in dem dann Parameter eingestellt werden können.

Werkzeuge für Vektor und Pixelgraphik:

- **Hilfslinien und Lineale:**

Mit **Ansicht – Lineale** werden die Lineale angezeigt und durch Herausziehen eines Lineals wird eine waagrechte oder senkrechte Hilfslinie erzeugt.

- **Fangfunktionen:**

Mit **Ansicht – An Gitter bzw. Hilfslinien bzw. Objekten ausrichten** und **Ansicht – Dynamische Hilfslinien** werden die automatischen Fang- und Ausrichtfunktionen ein- bzw. ausgeschaltet.

Mit **Extras – Optionen – Arbeitsbereich – An Objekten ausrichten bzw. Dynamische Hilfslinien** werden die Einstellungen für Fang- und Ausrichtfunktionen konfiguriert.

- **Zeichenwerkzeuge:**

In **Hilfsmittelpalette 4 (Corel-Draw)** bzw. **Hilfsmittelpalette 6, 7 (Corel-Draw)** befinden sich die Werkzeuge zum Zeichnen von Linien, Splinekurven, etc. bzw. von Grundfiguren.

- **Eigenschaften:**

Mit **Fenster – Symbolleiste – Eigenschaften** bzw. **Fenster – Andockfenster – Eigenschaften** öffnet man die Symbolleiste bzw. das Andockfenster zum Einstellen bzw. Ändern von Eigenschaften eines Objektes wie Strichart und Farbe des Umriss, Farbe und Muster der Füllung etc.

- **Füllung:**

Mit **Hilfsmittelpalette – Füllung bzw. Hilfsmittelpalette – Interaktive Füllung** können Farben bzw. Muster für einheitliche Füllungen bzw. Füllungen mit stetigen Übergängen erzeugt werden.

- **Pipette:**

Mit **Hilfsmittelpalette 12 (Corel-Draw)** bzw. **Hilfsmittelpalette 5 (Corel-Photo-Paint)** kann man mit einer Pipette auf ein Pixel zeigen, erhält damit Informationen (z.B. RGB-Wert) der Farbe und kann diese Farbe für die Verwendung mit anderen Werkzeugen übernehmen.

- **Textwerkzeuge:**

In **Hilfsmittelpalette 10 (Corel-Draw)** bzw. **Hilfsmittelpalette 7 (Corel-Photo-Paint)** kann Text eingefügt werden.

- **Gruppieren:**

Mit **Anordnen (Corel-Draw)** bzw. **Objekt – Anordnen (Corel-Photo-Paint)** können Gruppierungen von Objekten gebildet und aufgelöst werden bzw. Objekte in den Vordergrund bzw. Hintergrund verschoben werden.

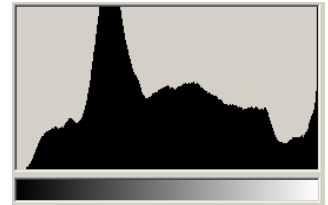
Werkzeuge für Pixelgraphik:

- **Beschneiden (Crop): (Hilfsmittelpalette 3 - Beschneiden)**

Durch Angabe eines Rechteckes wird der neue Rand festgelegt, und anschließend mit Doppelklick der außerhalb des Randes liegende Bereich gelöscht.

- **Histogramm:**

Um die Helligkeits-, Kontrast- und Farbverteilung eines Bildes zu beurteilen, kann man ein Histogramm anzeigen (**Bild – Histogramm**), oder wird bei den Werkzeugen für das Anpassen bzw. Bearbeiten von Bildern ein Histogramm angezeigt (**Bild – Anpassen**).



- **Anpassen, Ändern, Bearbeiten:**

Mit diesen Werkzeugen (**Bild – Anpassen**) werden Kontrast, Helligkeit, Sättigung, Farbbalance, Gamma, Tonkurve etc. des Bildes geändert bzw. automatisch ausgeglichen.

- **Effekte und Filter:**

Mit diesen Werkzeugen (**Effekte**) wird das Bild unschärfer gemacht (Rauschen hinzufügen, Blur), geschärft (Schärfen, Sharpen), Konturen extrahiert, künstlerische Effekte erzeugt bzw. (**Bild - Korrektur**) Staub- und Kratzer entfernt (Despeckle), d.h. es werden vereinzelte Pixel der Umgebung angepasst.

- **Retuschieren und Klonen:**

Mit einem Retuschierwerkzeug („Pinsel“, „Wedel“) (**Hilfsmittelpalette 8 – Retuschierpinsel**) kann in einem lokalen Bereich mit fließendem Übergang zur Umgebung - sowie beim Anpassen eines Bildes - Helligkeit, Kontrast, etc. geändert werden. Mit dem Werkzeug **Hilfsmittelpalette 8 – Klonen** kann ein Bereich eines Bildes auf einen anderen übertragen (geklont) werden.

- **Kanaltrennung und –kombination: (Bild – Kanäle teilen in, Kanäle kombinieren)**

Damit kann ein Bild zur besseren Bearbeitung oder zur Druckvorbereitung in die einzelnen Kanäle des gewünschten Farbmodells aufgeteilt werden. Zwei Anwendungsbeispiele:

- Zum Druck benötigt man die einzelnen Kanäle des CMYK-Modells.
- Zur Verbesserung des Kontrastes eines Bildes, ist es anstatt einfach den Kontrast zu erhöhen, oft besser, das Bild zuerst in die HLS-Kanäle zu zerlegen, dann beim L (Lightness)-Kanal den Kontrast zu erhöhen und dann die modifizierten Kanäle wieder zu kombinieren.

(siehe auch das Arbeitsblatt **Farben**)

- **Freistellen, Maske, Lasso, Zauberstab: (Hilfsmittelpalette 2 – Maske)**

Man wählt einen oder mehrere Bereiche des Bildes durch Zeichnen einer Kurve oder durch Auswahl mit einem Werkzeug, das nach Angabe eines Punktes alle Punkte in einem Bereiche um den angegebenen Punkt (Lasso) oder im Bild (Zauberstab) bestimmt, die sich vom Angabepunkt in Helligkeit und/oder Farbe durch einen im Einstellungsfenster angegebenen Toleranzwert unterscheidet.

Dieser Bereich kann dann mit anderen Werkzeugen bearbeitet werden.

- **Zahl der verwendeten Farben reduzieren bzw. erhöhen (Bild – Farbmodus)**

Eine Reduktion der Farben kann etwa für die Verkleinerung der Bilddatei oder die Erzeugung eines Schwarz-Weiss-Bildes nötig sein.

Eine (formale) Erhöhung der Anzahl der Farben kann nötig sein, um gewisse Werkzeuge anwenden zu können.

- **Bild neu berechnen (Bild – Bild neu aufbauen)**

Um Qualitätsverluste zu vermeiden ist es sinnvoll beim Einbinden eines Bildes in ein zu druckendes Dokument oder in eine Webseite, das Bild mit einem Bildbearbeitungsprogramm neu zu berechnen, sodass es bei vorgegebener Auflösung (dpi – dots per inch - etwa für den Druck 600 dpi, für eine Webseite zum Anzeigen am Schirm 72 dpi) die gewünschte Größe hat.

- **Ebenen Objekte (Fenster – Andockfenster – Objekte)**

Beim Erzeugen und Bearbeiten von Bildern können die Informationen auf mehrere Ebenen aufgeteilt werden, die dann einzeln bearbeitet werden können. Diese können dann – unter Angabe von Gewichtung und Transparenzfunktionen kombiniert werden.

FARBEN

Eine Farbe wird durch eine Intensitätsfunktion in Abhängigkeit von der Wellenlänge festgesetzt. Die Wellenlänge legt den Farbton (Hue) (die Position im Regenbogen) fest. Für die Darstellung im Bereich der Computergraphik, müssen einfachere Modelle verwendet werden.

Werte: Folgende Werte werden bei den verschiedenen Farbmodellen verwendet:

- **Hue (Farbton):** Festgelegt durch die Wellenlänge.
- **Brightness(Helligkeit):** Die wahrgenommene Intensität.
- **Luminance (Luminanz, Leuchtkraft):** Die ausgesandte Intensität.
- **Saturation (Sättigung):** Legt den "Schwarzanteil" einer Farbe fest.

Farbmodelle: Die häufigsten Farbmodelle sind:

- **sRGB (standard Red, Green, Blue)**

Wie oben beschrieben werden die Farben aus Rot, Grün und Blau zusammengesetzt. Diese Farbmodell beruht auf der additiven Farbmischung und wird etwa beim Monitor verwendet.

- **HSB (Hue, Saturation, Brightness)**

Bei diesem Modell wird die Art und Weise nachvollzogen, wie das menschliche Auge die Farben erfasst.

- **HSV (Hue, Saturation, Value)**

Wie bei HSB. Anstatt Brightness wird hier der Wert (Value) der Helligkeit angegeben.

- **HLS (Hue, Luminance, Saturation)**

- **CMYK (Cyan, Magenta, Yellow, Black)**

Bei diesem Farbmodell werden die Farben durch subtraktive Farbmischung aus den zu Rot, Grün, Blau komplementären Farben zusammengesetzt. Damit schwarze Flächen wirklich rein schwarz erscheinen wird noch die vierte Farbe schwarz benutzt. Dieses Farbmodell wird beim Druck verwendet.

- **LAB (CIE L*a*b)}**

Dieses Modell besteht aus einer Helligkeitskomponente ("L" für Luminanz) und zwei chromatischen Komponenten: der Komponente "a" (grün bis rot) und die Komponente "b" (blau bis gelb) und wurde festgelegt durch die Commission Internationale de l'Eclairage (CIE)

Die Umrechnung zwischen den verschiedenen Modellen können mit Werkzeugen von Bildbearbeitungsprogrammen visualisiert und durchgeführt werden (etwa bei Corel-Draw 12 mit dem Menü **Fenster – Andockfenster – Farbe bzw. Farbstile**).

Farbprofile

Damit bei Verwendung verschiedener Medien (Monitor, Farbdrucker mit unterschiedlichen Tinten) beim selben Bild derselbe Farbeindruck entsteht, werden ICC-Farbprofile zur Anpassung der geräte-spezifischen Eigenheiten an ein Standard-Farbmodell verwendet. Die Standardisierung erfolgt vom International Color Consortium (ICC) (<http://www.color.org>).

Gamut

Es ist zu beachten, dass nicht alle Medien denselben Bereich an Farben darstellen können. Den Bereich der darstellbaren Farben bezeichnet man als Gamut.

Gamma-Korrektur:

Das Auge registriert das Verhältnis von Helligkeitsintensitäten und nicht die Differenz. Auf einer Intensitätsskala zwischen 0 und 1 erscheint unseren Augen daher der Unterschied zwischen 0.10 und 0.11 sowie der Unterschied zwischen 0.50 und 0.55. Will man also den Bereich zwischen der Intensität I_0 und I_{256} in gleichmäßig erscheinende Abstufungen aufteilen, so muss gelten: $I_1 = r I_0$, $I_2 = r I_1$ usw. Da bei Ausgabe einer Graphik auf verschiedenen Medien (Schirm, Ausdruck auf Papier) die Helligkeitsunterschiede unterschiedlich erscheinen, muss eine Gammakorrektur durchgeführt werden, d.h. eine geeignete Funktion der eingegebenen und ausgegebenen Helligkeit angegeben werden.

Für weitere Infos siehe auch

die Internetseite der LVA „Farbe in der Computergraphik“
(Institut für Computergraphik und Algorithmen)

http://www.cg.tuwien.ac.at/courses/ColourInComputerGraphics/slides_2004.html

oder das Buch Foley, vanDam: Computer Graphics, Addison Wesley. 1990.

2D-VEKTORGRAPHIK

Bei einer Vektorgraphik werden Strecken, Linien in einem Koordinatensystem durch Koordinaten der Angabeelemente gespeichert und erst für das jeweilige Ausgabemedium durch geeignete Software in Bildpunkte umgesetzt. Daher ist es auch möglich eine Vektorgraphik ohne Qualitätsverlust zu skalieren.

Die gebräuchlichsten Datenformate zum Speichern von Vektorgraphiken sind:

- **PostScript (*.ps)**

Siehe auch <http://www.ghostscript.com>

- **WMF Windows Meta File (*.wmf)**

WMF setzen auf das "Graphic Device Interface" (GDI) von Windows auf.

- **EMF Enhanced Windows Meta File (*.emf)**

Erweiterung von WMF für die 32 Bit Versionen von Windows

- **CGM Computer Graphics Meta-File (*.cgm)**

Das ist das Standard Format des American National Standards Institute und ist unabhängig vom Betriebssystem.

- **HPGL HP-Graphic Language (*.plt)**

Die "alte Plottersprache" (z.B. HP 7475) der Firma HP

Insbesondere für das Internet:

- **Shockwave-Flash (*.swf)**

Derzeit ein "quasi Standard" für Vektordateien, aber kein OpenSource Produkt, da die Firma Macromedia die Entwicklung kontrolliert.

Siehe auch <http://www.macromedia.com>

- **SVG Scaleable Vector Grafics (*.svg)**

Vom W3C-Consortium vorgeschlagenes Format für die Darstellung von Vektorgrafiken. Dieses Format wird vor allem von Adobe und Corel unterstützt.

Siehe auch <http://www.adobe.com/svg/main.html>

2D-PIXELGRAPHIK

Bei einer Pixelgraphik wird für jedes Pixel eine Farbinformation gespeichert. Dabei fallen große Datenmengen an, daher werden die Informationen komprimiert gespeichert, wobei verlustfreie und verlustbehaftete Verfahren zu unterscheiden sind.

Die gebräuchlichsten Datenformate zum Speichern von Pixelgraphiken sind:

TIFF Tag Image File Format (*.tif):

Ein plattformunabhängiges verlustfreies Format

GIF Graphic Interchange Format (*.gif)

Ein verlustfreies Format, das sich besonders für Strichzeichnungen eignet, maximal 256 Farben, eine Transparenzfarbe. Wegen Lizenzproblemen und neuere Entwicklungen ein auslaufendes Format.

PNG Portable Network Graphics (*.png)

Ein verlustfreies Format mit verschiedenen Komprimierungsalgorithmen bis 64-bit Farbtiefe und einer Transparenzfarbe. Das aktuelle Format für das Internet.

JPG Joint Photographics Graphic (*.jpg)

Ein verlustbehaftetes Format, das sich besonders für Bilder eignet, 24-bit Farbtiefe, keine Transparenz, aktuelles Pixelformat für Photos.

JPEG 2000 (*.jpg)

Der Nachfolger von Jpeg. Ein verlustbehaftetes Format mit Wavelet Komprimierung, keine "Kastenbildung", wird JPG ablösen.