

# Hilfe zum Nostruktur für EEP

## Kapitel 1

### Grundsätzliches

Um mit dem Nostruktur ein Modell zu erstellen sind 3 Arbeitsschritte, respektive 3 Dateiformate zu erstellen.

Als erstes erfolgt die Erstellung der einzelnen Bauteile im „Konstruktionsfenster“.

Damit erstellte Dokumente werden als \*.kon Dateien gespeichert.

Der zweite Schritt erfolgt im „Nostruktur“ Fenster (3D Ansicht). Hier werden die \*.kon Dateien zum Modell zusammengefügt und positioniert. Diese Dateien werden als \*.Mod2 gespeichert.

Der letzte Schritt zum fertigen Modell erfolgt im „MoveModell“ Fenster.

Die damit erstellten Dateien werden als \*.GSB gespeichert. In dieser Datei sind alle Informationen zum eigentlichen Modell also zur Mod2 Datei enthalten.

### Arbeiten mit Kon-Dateien

#### **Erstellen einer Kondatei**

Im Gegensatz zum Konstruktor (EEEC) liegt der Nullpunkt der X Y Z-Achsen in der Mitte des Dokuments. Dies erleichtert die Positionierung der einzelnen Elemente zueinander.


Da Modelle grundsätzlich im Koordinatenmittelpunkt stehen sollten, ist es Vorteilhaft wenn schon beim zeichnen darauf geachtet wird.


Beginnen wir mit einer einfachen Lektion.

Schon beim Öffnen des Nostruktors wählen wir „Konstruktion“ aus.



Wir vergrößern das Dokument, mit den Pfeilbuttons in der Werkzeugleiste oder mit den Pfeiltasten soweit, bis wir ein kleines Raster sehen. Wir suchen nun den Dokumentenmittelpunkt welcher mit einem roten XY Koordinatenkreuz gekennzeichnet ist.

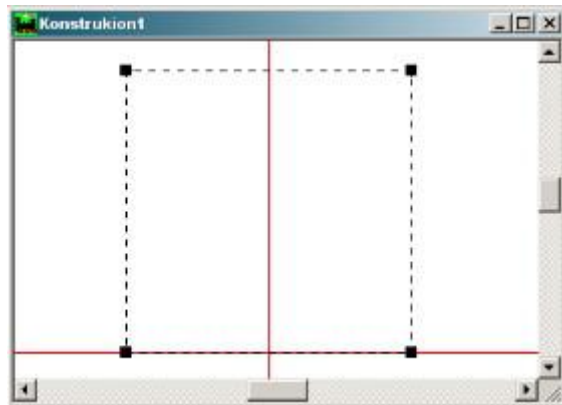
Hier zeichnen wir mit dem Werkzeug  ein Polygon (Rechteck), beginnend auf der Nullkoordinate der X-Achse (waagrechte) mit den Maßen X 400, Y 400 (4x4 m). Dabei sollte die Y-Koordinate (Senkrechte) in der Mitte des Polygons verlaufen. Hierbei können wir auch die Koordinatenanzeige im linken untern Fensterrand **x:200,y:0** zu Hilfe nehmen.

Mit einem Klick auf das Werkzeug  verbinden wir die Linien durch „Vertexe“ und erhalten ein geschlossenes Polygon.

Mit  in der oberen Fensterleiste öffnen wir das 3D Fenster (Nostruktur).

Nun ist der Button  aktiv und wir können das Polygon mit einem Klick darauf ins 3D Fenster einfügen.

Im 3D Fenster können wir nun das Polygon mit Hilfe der Pfeiltasten des Ziffernblocks oder durch anklicken und festhalten der linken Maustaste in jede gewünschte Richtung drehen.




Da im 3D Fenster die Z und Y Achsen im Gegensatz zum Konstruktionsfenster vertauscht sind, muss das Polygon erst mal um 90° nach hinten gekippt werden. Den Mauszeiger etwas über der Mitte platzieren, linke Maustaste gedrückt halten bis das Polygon senkrecht steht.

Zurück zum Kon-Fenster doppelklicken wir auf  Vertexeigenschaften in der Werkzeugleiste.  
„Die Werkzeugleiste lässt sich wie jedes Windowsfenster an jeden Bildrand verschieben“

Mit dem nun veränderten Mauszeiger gehen wir zum linken oberen Vertex des Polygons und doppelklicken mit der linken Maustaste. Im folgenden Fenster gehen wir zu Farbwahl-dialog und wählen eine Farbe. Farbexperten können Farbwerte auch direkt eingeben, was aber sehr gute Farbkenntnisse voraussetzt.

Nachdem wir im Farbwahlfenster die Farbe erstellt haben klicken wir auf schliessen wir Dieses mit Okay und schliessen das Vertex Eigenschaftenfenster ebenfalls mit Okay.

Ein Klick auf  entfernt das Polygon aus dem 3D Fenster und ein nochmaliger Klick fügt das geänderte Polygon wieder ein.

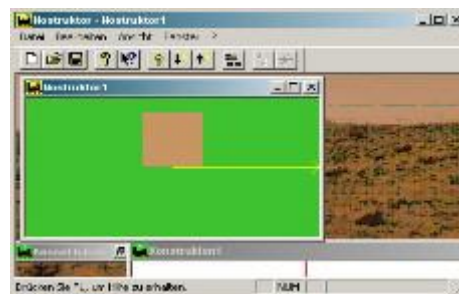
### Positionieren von Polygonen im 3D Fenster

Um immer einen Überblick zu behalten hier noch ein Tip wie ich arbeite:

Das 3D Fenster habe ich so gross wie möglich links oben, das Kon Fenster mit dem ich gerade arbeite ist dabei über den unteren Bildrand hinausgeschoben, alle anderen Kon Fenster sind Minimiert ebenfalls am unteren Bildrand. Welches Fenster gerade aktiv ist zeigt die farbige obere Fensterleiste. Will ich nun im Kon Fenster arbeiten vergrössere ich es durch einen Doppelklick auf die obere Fensterleiste.




Konstruktionsfenster ist aktiv



3D Fenster ist aktiv

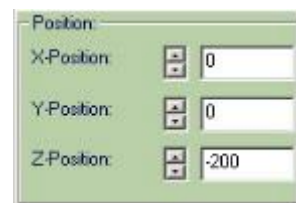
Stellen wir uns vor wir wollen ein kleines Häuschen mit einer Grundfläche von 4x4 Meter und einer Höhe von 4 Meter, mit einem Satteldach. Wir brauchen also die gerade Wand zweimal und eine Giebelwand ebenfalls zweimal. Eine Wand haben wir bereits. Diese setzen wir zunächst einmal um die halbe Breite der späteren Giebelwand nach vorne also um 2 Meter auf der Z Achse.

Dazu gehen wir über  in den Dialog „Kopien anlegen“. Jede Kopie eines oder mehrerer Polygone innerhalb eines Dokumentes wird als Instanz bezeichnet. Die erste Instanz Trägt dabei die Nummer 0. Wir klicken auch „gehe zu...“ und befinden uns im Instanzen Menü.


Hier erfolgt die Eingabe der Werte für die Position der Polygone im Raum, also im 3D Fenster, Die Werte für eine evtl. Skalierung, falls die Polygone kleiner oder grösser als gezeichnet dargestellt werden sollen, sowie die werte für evtl. Drehungen aller drei Achsen.



Wir geben für unsere „Wand“ im Fenster Position bei Z – Position den Wert –200 ein. Mit „Übernehmen“ gehen wir zurück und erstellen eine neue Instanz. Wir wählen mit den Richtungstasten neben dem Fenster „Aktuelle Instanz“ Die 1, falls nicht automatisch nach dem Erstellen angezeigt, und klicken auf „gehe zu...“ Diese Instanz soll die Rückwand werden, welche 400 cm Hinter der Vorderwand stehen soll, also geben wir bei Z-Position den Wert +200 ein. Allerdings darf hier das Vorzeichen + nicht mit angegeben werden. Lediglich Minuszeichen müssen angegeben werden. Zahlen ohne Vorzeichen sind automatisch +Werte.





Wir schliessen das Fenster mit „Übernehmen“ und


anschliessend mit „Okay“ und klicken Zweimal auf  um die Wände ins 3D Fenster einzufügen.






Nun haben wir eine Vorder- und eine Rückwand für unser Häuschen.  
In der gleichen Weise erstellen wir nun in einem neuen Dokument die Seitlichen Giebelwände.  
Dazu klicken wir auf „Datei“ in der oberen Bildleiste und speichern die Kondatei unter dem Namen „Wand-1-2.kon“ ab. Spätestens jetzt sollten wir einen Namen für unser Objekt haben, um die Datei in einem entsprechenden Verzeichnis zu speichern wo wir alle zu diesem Objekt gehörenden Dateien speichern um sie später mal wiederzufinden.

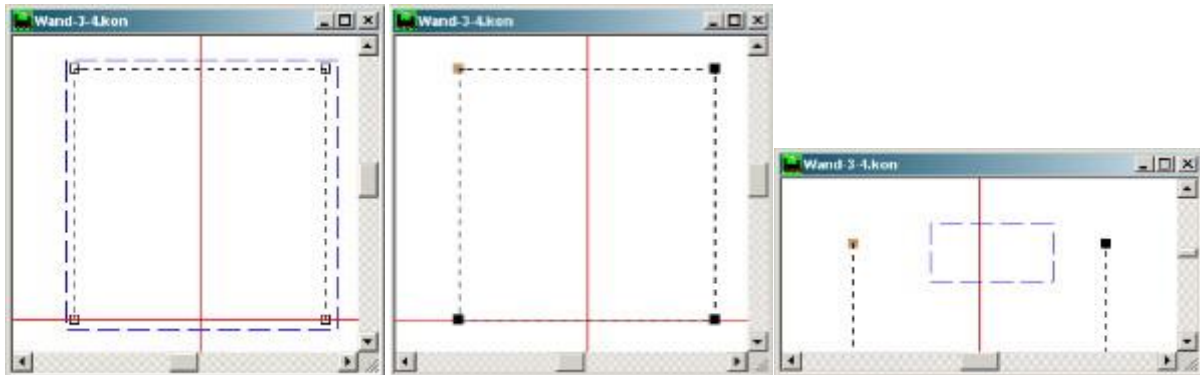
### ***Kopieren von Polygonen in ein anderes Dokument***

Mit dem Werkzeug  „Kennzeichnen“ ziehen wir ein Rechteck um die Zeichnung, darauf achten daß alle Vertexe innerhalb des markierten Bereiches sind, und klicken auf .

Eine Kopie der Zeichnung befindet sich nun in der Zwischenablage. Wir minimieren das Dokument „Wand-1-2“ wieder und öffnen als nächstes wir über  ein neues Konstruktionsdokument.

Mit  fügen wir die Kopie aus der Zwischenablage in das neue Dokument ein und positionieren es auf der X-Achse mittig zur Y-Achse. Mit dem Werkzeug  kann man markierte Polygone oder Teile davon auf dem Dokument verschieben, auch wenn darunter ein anderes Polygon ist. Vorhandene Polygone werden beim Einfügen nicht automatisch mit markiert.

Anschließend löschen wir mit dem gleichen Werkzeug  die obere waagrechte Linie des Polygons. Markieren der Linie, wobei der Mittelpunkt innerhalb der Markierung liegen muss, mit der „Entf.“ Taste löschen. Markierung aufheben durch Klick auf eine nicht markierte Stelle.






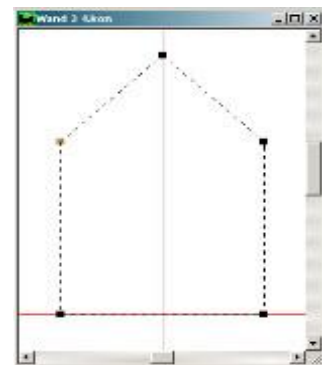
markieren und kopieren



einfügen in neues Dokument

Linie markieren und löschen.

Nun zeichnen wir den Giebel auf die neue Wand.

Wir setzen mit dem Werkzeug  am linken oberen Vertex an und ziehen mit gehaltener Maustaste eine Linie nach oben rechts bis zu den Koordinaten Y 600 X 0, lassen die Maustaste los und ziehen eine weitere Linie zum rechten oberen Vertex. Mit  verbinden wir die Linien. Mit  doppelklicken wir auf den oberen mittleren Vertex und geben diesem die gleiche jedoch etwas aufgehellte oder abgedunkelte Farbe wie bei der ersten Wand.



*Hinweis: man kann mehreren Vertexen die gleiche Farbe zuordnen indem man zunächst mit  auf den Vertex mit der richtigen Farbe doppelklicken, mit Okay schließen und mit dem Werkzeug  die Farbe auf die anderen Vertexe durch anklicken überträgt. Nun kann man die Farbe in diesen Vertexen leichter abdunkeln oder aufhellen.*

Da wir ja wieder zwei Wände brauchen wiederholen wir den Schritt wie bei der ersten Wand, indem wir wieder innerhalb des Dokumentes eine zweite Instanz anlegen. Wers nicht mehr weiß – oben nachlesen.

Nun müssen diese beiden Wände ja um 90° zur Y-Achse gedreht und entlang der X-Achse positioniert werden.

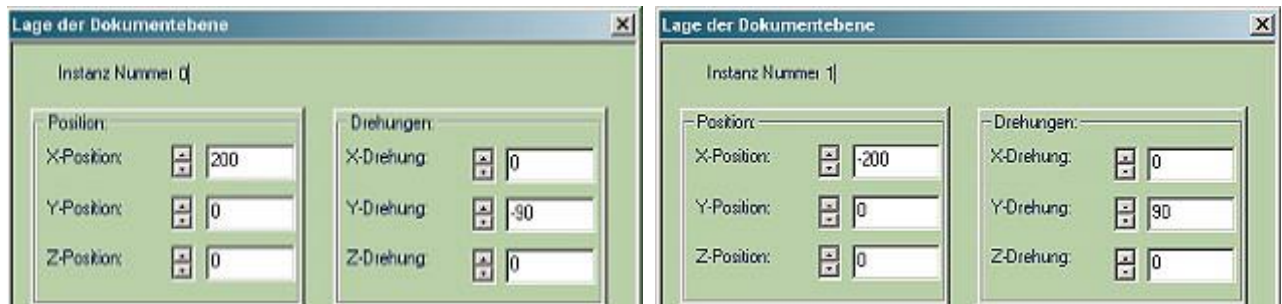
Wie das geht wissen wir inzwischen. Hier aber nochmal die richtigen Werte zur Kontrolle.

Die Werte für die Drehung sind einmal als Minuswert und einmal als Pluswert einzugeben.

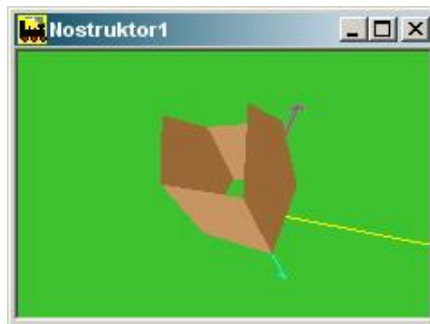
Dies erleichtert später das Ausblenden der richtigen Seiten. Dies aber weiter unten.

Die rechte Wand (Instanz Nr.0) hat bei Y-Drehung den Wert –90 und bei X-Position den Wert 200.

Die linke Wand (Instanz Nr.1) bei Y-Drehung den Wert 90 und bei X-Position den Wert –200.



Nach dem Einfügen in das 3D Fenster muss es dann so aussehen:



### ***Arbeiten mit der Z-Achse (Tiefe) im Konstruktionsfenster***

Nun machen wir das Dach und lernen dabei die „Dritte Dimension“, die Z-Achse kennen.

Ein Dach besteht aus zwei Hälften. Dazu müssen wir aber nicht 2 Dokumente erstellen oder ein Polygon über Instanzen kopieren, sondern wir arbeiten hier gleich mit der Z-Achse.


Wir zeichnen das Dach von der Aufsicht. Die Giebelkante verläuft hier längs der X-Achse.

Also zeichnen wir auch zwei zusammenhängende Polygone deren Mittellinie parallel und mittig zur X-Achse verläuft.


Da das Dach in diesem Beispiel nirgends überstehen soll, können wir die Maße der Wände nehmen. Dabei geben wir den Vertexen bereits die endgültige Höhe, so daß das fertige Dach nur noch gedreht werden muss. Das Ganze sieht dann so aus:


Wir haben zwei zusammenhängende Polygone mit den Maßen: X= 400, Y= je 200. Das Koordinatenkreuz liegt genau in der Mitte.

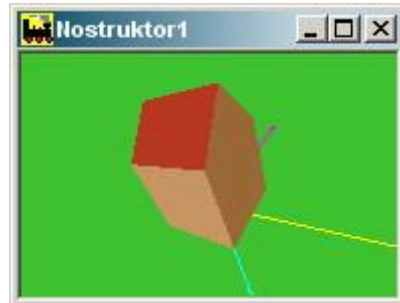


Mit  doppelklicken wir auf den oberen linken Vertex und geben dort folgende Werte ein: Höhe Z = 400 (höhe der Wände)

Bevor wir das Fenster schliessen geben wir dem Vertex noch eine andere Farbe. Z.B. rot.

Anschliessend klicken wir auf  und übertragen diesen Wert auf die anderen 4 Eckvertexe durch anklicken.

Wir nehmen nochmal das Werkzeug  und doppelklicken auf den mittleren linken Vertex. Wir ändern auch hier zunächst die Farbe in rot und geben bei Höhe Z den Wert –600 ein. Dies ist die Höhe der Giebelspitze. Bei Drehungen muss nun noch der Wert –90 eingegeben werden und das Dach kann in das 3D Fenster eingefügt werden:



Das Modell würde nun jedoch in EEP auf der Seite liegen, da im 3D Fenster die Z und Y Achsen vertauscht sind.

Es gibt nun 2 Wege das Modell auf die richtige Seite zu stellen. Der etwas umständliche Weg ist – gleich bei der Erstellung der Kondateien jedes Polygon und jede Instanz um 90° zu drehen.

Da ich dazu zu faul bin nutze ich persönlich den 2. bequemeren Weg. Wir haben im Nostruktur die Möglichkeit komplette Objekte im 3D Fenster auf allen drei Achsen zu drehen, zu verschieben und sogar die gesamte Grösse zu ändern, zu Skalieren.

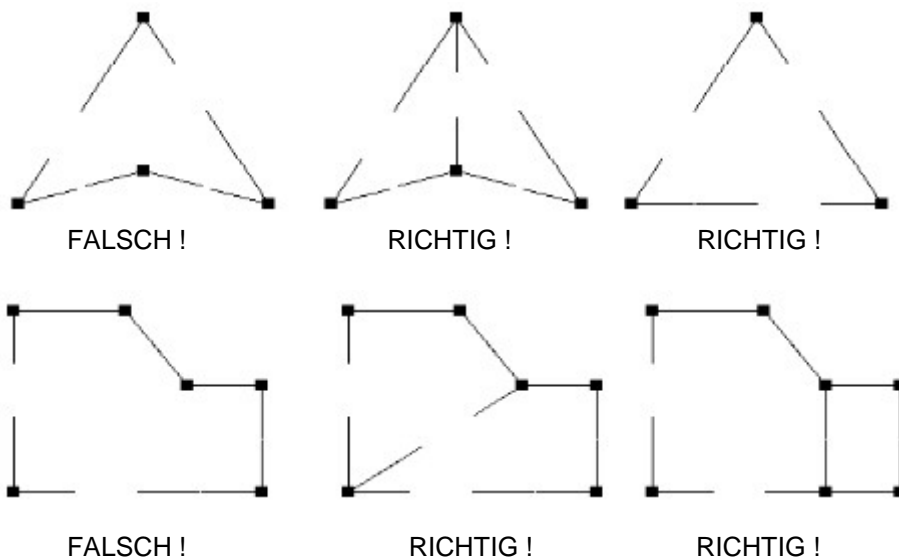
Ihr müsst selber herausfinden welche Methode für euch die bessere ist.

Bevor ich das Kapitel *Arbeiten mit Kondateien* schliesse, noch einige nützliche Hinweise. Was bedeuten die 3 unterschiedlichen Linien.



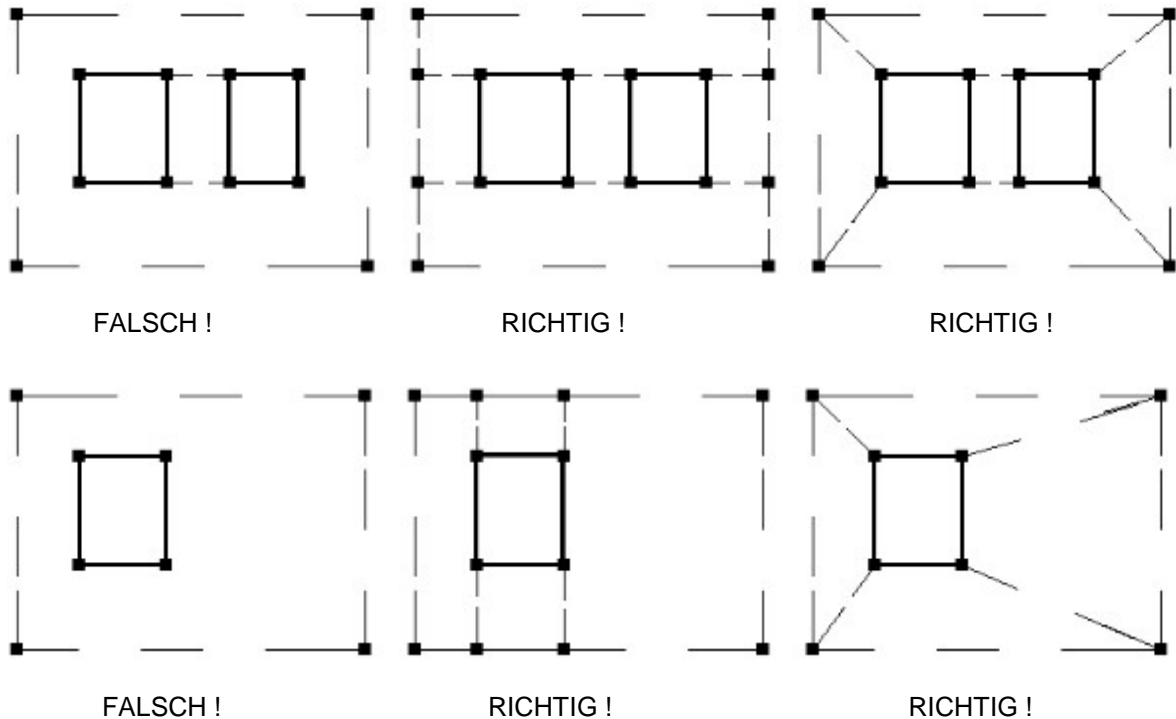
Das Haupt-Linienwerkzeug. Mit dieser Linie werden alle geschlossenen Polygone sowie Unterteilungen von Polygonen gemacht.

*Polygone müssen immer in sich geschlossen sein.* Hier einige Beispiele:



Begrenzungslinie. Diese Linie verwendet man für Ausschnitte (Löcher) in einem Polygon. Diese Linie stellt die absolute Aussenkante eines Polygons dar. Will man z.B. in einer Hauswand offene, also durchsichtige Fenster darstellen, dann werden diese Ausschnitte mit dieser Linie gemacht. Es muss aber darauf geachtet werden, daß die Wand, also das umschliessende Polygon in sich geschlossen ist. Beispiel:





Die Einzellinie. Ein Relikt aus EEEC-Zeiten. Ursprünglich wurden damit Linien in EEEC dargestellt, z.B. Stäbe oder auch Oberleitungsdrähte usw. DX kann keine Linien darstellen. dies ist aber trotzdem ein nützliches Werkzeug, wenn man mal eine Hilfslinie zum zeichnen komplizierter Polygone benötigt.


Beispiel:

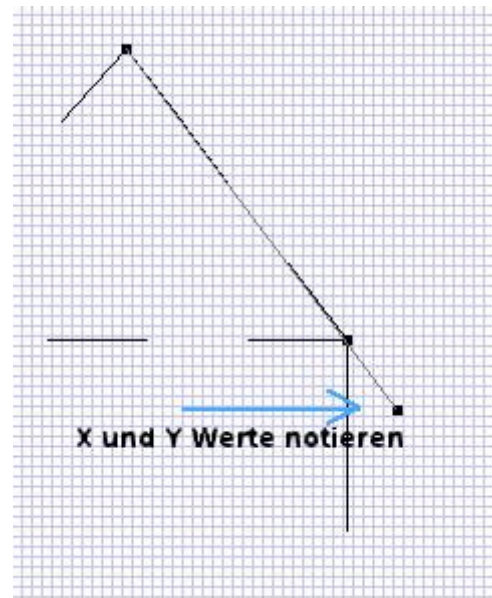
wir wollen die Dachkante überstehen lassen und müssen dazu die Werte für die X und Y Achse finden. Dazu zeichnen wir von der oberen Giebelspitze mit diesem Werkzeug eine Linie über einen der unteren Vertexe hinaus. Dabei sollte sich die Linie mit dem unteren Vertex möglichst genau decken.

Nun können wir am unteren Ende der Linie die Werte für die X und Y Achsen in der Statuszeile ablesen. Anschliessend kann die Hilfslinie gelöscht werden.

Die neuen Werte übertragen wir wie folgt in unser Dokument Dach.

Dabei ist der Y-Wert bei den Vertexeigenschaften Bei Höhe Z einzutragen und für den X-Wert sind die vier Eckpolygone in Y Richtung entsprechend zu verschieben.

Vertexe können ganz einfach mit dem Werkzeug  verschoben werden.



Noch eine Anmerkung zur Position von Vetexen:

*Vertexe sollten immer auf einer geraden Koordinate liegen. Liegt ein Vertex auf einer Ungeraden, z.B. X 101 oder Y -13 wird dieser beim abspeichern auf die nächsthöhere Koordinate gesetzt.*



Ein weiteres nützliches Werkzeug ist das Lineal mit beweglichem Winkelmesser. Damit lassen sich einfach exakte Kreise zeichnen.

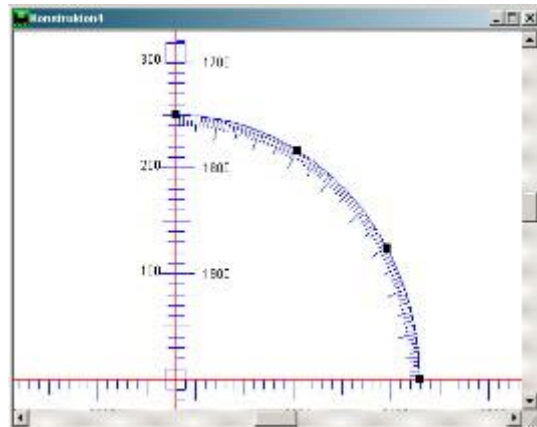
Am Beispiel eines Zylinders will ich mal eine der vielen Möglichkeiten demonstrieren.

Wir öffnen ein neues Konstruktionsdokument und fügen das Lineal hinzu.

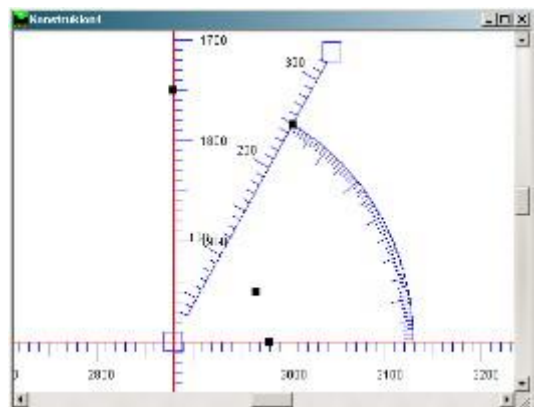
Abhängig von der Grösse (Durchmesser) des Zylinders vergrössern oder verkleinern wir die Arbeitsfläche. Da das Lineal dabei meist aus dem Blickwinkel gerät (die Position wird leider nicht gespeichert) doppelklicken wir im Schnittpunkt der x-Y Achse, halten die Maustaste gedrückt und vergrössern die Ansicht. Am besten ziehen wir den Schnittpunkt genau auf das Koordinatenkreuz des Dokuments. Den Bildausschnitt verschieben wir dabei mit Hilfe der Pfeiltasten.

Im Beispiel wurde die Ansicht in der Grösse so eingestellt, daß der Kreis des Lineals einen Radius von 250 cm hat. Da ich die Möglichkeit habe die Grösse später über die Instanzen zu variieren, kann ich das so lassen. anders wenn ich hier schon eine andere Grösse will. Dann nehme ich den verstellbaren Winkel zu Hilfe.

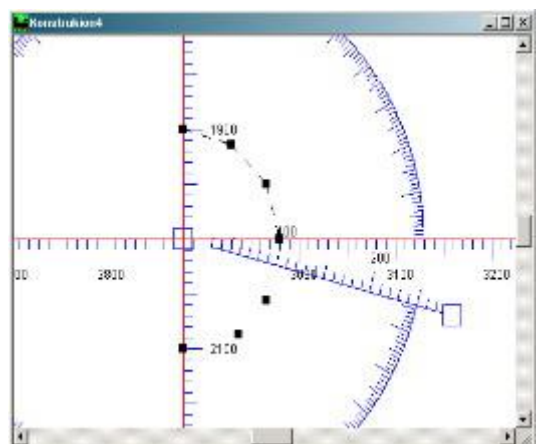
Ich möchte eine Ausgangsgrösse des Radius von 100 cm. Um die Polygonanzahl so klein wie möglich zu halten, wähle ich eine 30° Teilung.




Ich stelle den Winkel jeweils um 30 ° und verschiebe die Vertexe bis zur Markierung 100 des beweglichen Schenkels.



Auf diese Weise setze ich alle 30° einen Vertex vom Kreismittelpunkt aus bei jeweils 100. Dann verbinde ich die Vertexe durch Linien.



Nun gehe ich nacheinander mit  in die Vertexeigenschaften jedes einzelnen Vertexes, wobei die beiden Vertexe auf der 0-Linie der Y-Achse unberührt bleiben, und ändere die Werte bei Höhe Z

Dabei übernehme ich jeweils die Werte der horizontale X auf die Höhe Z und zwar als Minus Wert.

Anschliessend schiebe ich mit  alle Vertexe waagrecht bis zur Mitte also an die Y-Achse bis alle genau Senkrecht

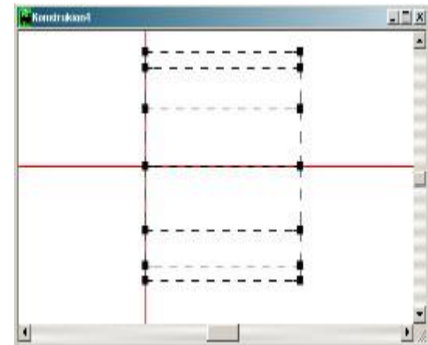
untereinander stehen. Dabei ist zu beachten daß die Vertexe in der Höhe nicht verschoben werden dürfen.

Anschliessend kopiere ich das ganze und füge es erneut ins Dokument ein, wobei die Kopie nach Rechts auf der X-Achse um 200 cm nach rechts verschoben wird.

Nun verbinde ich alle Vertexe waagrecht miteinander.

Wir haben nun einen liegenden halben Zylinder mit einem Durchmesser von 200cm und einer Länge von 200 cm.

Diesen Zylinder kann man nun beliebig in die Länge ziehen oder auch den Durchmesser ändern.



Da nun ein Zylinder mit nur einer Farbe nicht aus jeder Position als solcher erkennbar wäre, müssen wir die Polygone noch unterschiedlich einfärben.

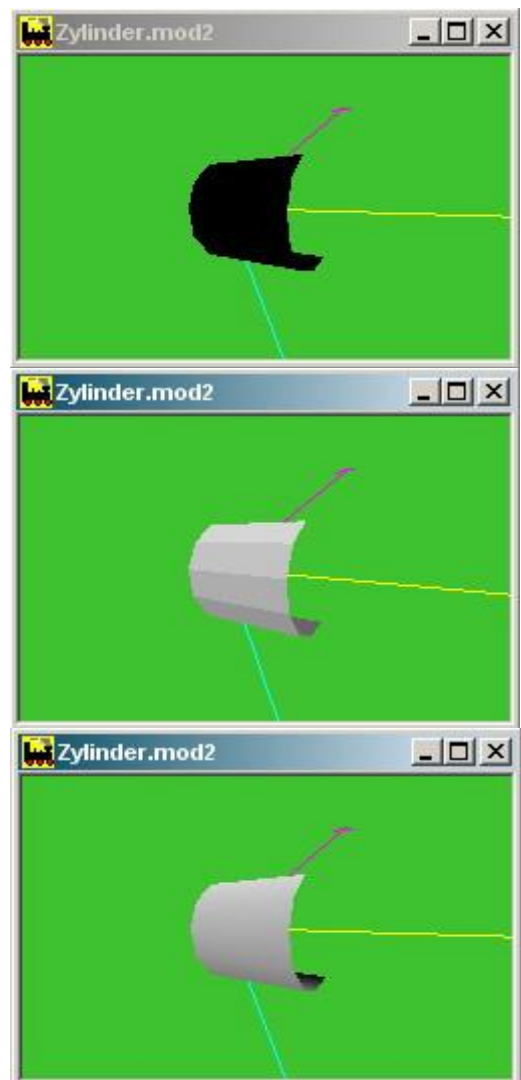
Dazu erstellen wir uns im Farbenmenü eine Reihe in der Helligkeit abgestufter Farben, von Hell nach Dunkel und färben unsere Polygone damit. Jetzt ist der Zylinder aus jeder Perspektive auch optisch als solcher zu erkennen.

Da wir auch die Möglichkeit eines Farbverlaufes haben, kann zwischen den Polygonen ein sanfter Übergang von Hell nach Dunkel geschaffen werden.

Diese Möglichkeit wird in einem anderen Kapitel nochmal genauer erklärt.

Dieses „Smoothshading“ genannte Verfahren ist auch deshalb nützlich, weil man hier eine Menge Polygone sparen kann, was der Performance sehr zuträglich ist, und dabei den selben Effekt, oder sogar noch einen besseren erzielt, wie mit sehr vielen Polygonen.

Um ohne das „Smoothshading“ einen annähernd gleichen Effekt zu erzielen, bräuchte man mindestens die 10fache Zahl an Polygonen.





## Werkzeuge bei aktivem Konstruktionsfenster



löschen/ausschneiden markierter Objekte (nur aktiv wenn Objekte markiert sind)



markierte Objekte in die Zwischenablage kopieren



Objekte aus der Zwischenablage ins Dokument einfügen



Linienwerkzeug (Standard Linie)



Linienwerkzeug (Aussenlinie – Polygonbegrenzung)



Vertex setzen



Einzellinie (Hilfslinie)



Rasierklinge – löschen einzelner Linien oder Vertexe



Zeichenwerkzeug - Lineal



Seitenwechsel der Linealteilung



Verschieben einzelner Vertexe (verbundene Linien werden mitverschoben)



Linien verbinden (verbindet alle offenen Linienenden und setzt Vertex)



Vertexeigenschaften



Kopien anlegen (Instanzen Menü)



Farbbestimmender Vertex



mehreren Vertexen die gleiche Farbe zuweisen



mehreren Vertexen den gleichen Lichtwert zuweisen



mehreren Vertexen die gleiche Höhe zuweisen (Höhe Z Achse)



Konstruktion ins 3D Fenster einfügen (nur Aktiv wenn ein Konstruktorfenster geöffnet ist)



Ansicht vergrößern (oder Pfeiltaste >nach oben< der Tastatur)



Ansicht verkleinern (oder Pfeiltaste >nach unten< der Tastatur)



Objekte markieren



# Hilfe zum Nostruktur für EEP

## Kapitel 2

### Arbeiten mit Texturen

#### Grundsätzliches

Texturen müssen in EEP in zwei Formaten vorliegen. Da der Nostruktur unter OpenGL läuft und dieses Programm nur BMP in 24 Bit Farbtiefe lesen kann brauchen wir zum Konstruieren von jeder Textur eine \*.BMP

EEP selbst dagegen läuft unter DirectX, wofür wir wiederum ein DDS-Format benötigen.

*Noch eine Anmerkung:*

*eigene Texturen können nur erstellt werden, wenn entsprechende Textur ID's vergeben wurden. Textur ID's werden grundsätzlich von SU vergeben und müssen für jeden Konstrukteur registriert werden. Vorhandene Texturen dürfen nicht verändert werden bzw. eine Änderung geschieht auf eigene Gefahr. Modelle mit willkürlich veränderten Texturen dürfen nicht verbreitet werden.*

Auch die Größe einer Textur kann nicht willkürlich gewählt werden. Das Programm liest nur Texturen mit bestimmten Größen bzw. Seitenverhältnissen.

Wir kennen das von unserem Arbeitsspeichern, hier gibt es keine Größe von 34 oder 72 (um mal ein Beispiel zu nennen). Die Größenabstufungen sind 1, 2, 4, 8, 16, 32 usw. also immer das doppelte bis zur maximalen Größe von 512 (bei Texturen).

Eine Texturgröße bis 32x32 ist eher die Ausnahme. Allerdings sollte die maximale Größe 512x512 eher vermieden werden. Denkbar und eher die Regel sind Größen wie 64x64, 128x128, 256x256 wobei auch Größen wie z.B. 128x256 oder 64x128 usw. möglich sind. Wir sehen – die Anzahl der Pixel einer Seite ist immer doppelt so hoch wie die nächst kleinere oder umgekehrt um die Hälfte kleiner als die nächst höhere.

#### Erstellen von Texturen

Mit jedem 24Bit Malprogramm lassen sich Texturen erstellen. Meistens hat man ein Foto oder eine andere gezeichnete oder gedruckte Vorlage welche sich dafür eignet.

Es können aber auch selber Texturen Pixel für Pixel „gemalt“ werden. Was allerdings nur geübten Usern zu empfehlen ist. Auch ist dies sehr zeitaufwendig.

Hier mal die Vorgehensweise am Beispiel eines Fensters. Wir haben eine Fotografie einer Hauswand und möchten daraus nur ein Fenster für unsere Zwecke nutzen.

Zunächst einmal erstellen wir eine neue Datei in der Größe 128x128, wobei die Hintergrundfarbe zunächst mal egal ist. Sie sollte so gewählt werden dass sich das Objekt gut davon abhebt.

Nun schneiden wir aus dem Original (ich habe im Beispiel einen Papier-Bastelbogen genommen), unser Objekt aus, verkleinern es notfalls und fügen es in unser neues Dokument ein.

Anschließend können wir das Objekt noch mit diversen Werkzeugen unseres Malprogrammes bearbeiten, bis es unseren Wünschen entspricht.

Wir positionieren das Objekt in der Fläche, so dass wir später noch andere Elemente einfügen können.

Ihr werdet nun feststellen, dass ich nur ein halbes Fenster in die Textur eingefügt habe. Mehr brauchen wir auch nicht. Ein ganzes Fenster wäre Platzverschwendung, da wir später die Hälften in zwei Polygonen spiegeln können und so wieder ein ganzes Fenster erhalten.



Original



fertige Fensterhälfte

( 2 )

Die Textur wäre nun soweit, dass wir das Fenster schon mal verwenden können. Hier muss ich aber noch auf eine weitere Eigenschaft von Texturen unter DX bzw. OpenGL hinweisen. Alle Farben werden angezeigt außer reinem Schwarz. Unsere Farben setzen sich ja aus den Grundfarben Rot, Grün und Blau zusammen. Wenn nun bei allen drei Grundfarben der Wert auf Null gesetzt wird, so entsteht reines Schwarz und wird vom Programm als durchsichtig interpretiert. Sobald auch nur in einer der Grundfarben ein Höherer Wert als Null gesetzt ist, ist dies kein reines Schwarz mehr und also nicht mehr durchsichtig. Dies wird dann als Schwarz dargestellt. Deshalb müssen wir in unserer Textur alles, was nicht dargestellt werden soll, also hier der Hintergrund um das Fenster, auf die RGB-Werte 0 einstellen. Die fertige Textur sieht nun so aus:



Nun muss die Textur nur noch einen Namen bekommen und kann erst mal gespeichert werden. Dabei kann ein beliebiger Name verwendet werden, oder auch die (von SU vergebene) ID Nummer als Name verwendet werden.

*Textur ID's können nicht wahllos genommen werden. Diese werden von Seiten SU an jeden Konstrukteur vergeben und müssen dort registriert sein.*

Eigentlich könnte nun schon mal mit dieser Textur im Nostruktur gearbeitet werden. Aber um unsere Texturen auch in EEP sichtbar zu machen, benötigen wir noch eine zweite Textur im DDS-Format. Außerdem müssen sogenannte Mipmap-Level erstellt werden.

Mipmap-Level sind in der DDS enthaltene proportional verkleinerte Texturen bis zur Größe 1x1 Pixel.

Eine Textur mit der Ausgangsgröße 128x128 enthält in der DDS also die Texturen von 128x128, 64x64, 32x32, 16x16, 8x8, 4x4, 2x2, 1x1 Pixel.

Dies hat folgenden Grund: Ein Objekt in EEP wird nicht immer in voller Größe angezeigt. Je weiter man sich vom Objekt entfernt, umso weniger Details sind zu erkennen. Somit muss auch die Textur nicht mehr alle Details enthalten. Dazu dienen die Mipmap-Level. Je kleiner also ein Objekt auf dem Bildschirm dargestellt wird um so kleiner ist der Texturlevel den sich das Programm aus der DDS zur Darstellung holt.

*Das Programm sagt sich: Ein Gegenstand der 20x20 cm groß ist, ist in 100 Meter Entfernung nicht mehr zu erkennen, also muss ich ihn nicht darstellen.*

Und noch eine Eigenheit von Texturen.

Alle Inhalte einer Textur sind verkleinerte Abbildungen der späteren tatsächlichen Größe im Modell. Das heißt, die Inhalte werden am Modell immer vergrößert. Bei einer solchen Vergrößerung werden alle Pixel mit den danebenliegenden Pixeln „vermischt“. Es entstehen somit je nach der Stärke der Vergrößerung zusätzliche Pixel. Dabei werden vom Programm Zwischenwerte der jeweils benachbarten Pixel sowohl in der Farbe als auch in der Helligkeit gebildet.

Dies gilt natürlich auch für die benachbarten Pixel des schwarzen Hintergrundes. Hier würden die zusätzlich bei der Vergrößerung geschaffenen Pixel einen dunklen Rand verursachen.

Um das zu verhindern, müssen die Objekte, die dargestellt werden sollen, strikt vom Hintergrund getrennt werden. Dies geschieht mit einer sogenannten Schablone. Damit sagen wir dem Programm, hier ist die Textur zu Ende, die Pixel außerhalb sind tabu.

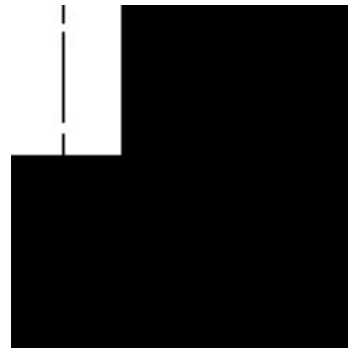
Diese Schablone machen wir, indem wir alles, was dargestellt werden soll, weiß färben.

Dabei müssen alle drei Grundfarben den Wert 255 aufweisen.

( 3 )

Diese „Schablone“ speichern wir unter dem gleichen Namen wie unsere „normale“ Textur und versehen sie noch mit \_a, z.B. Fenst\_a.bmp

Den Rest machen wir mit dem Tool „DxTex.exe“. Dort erstellen wir zunächst die Mipmap-Level der Originaltextur und fügen dann die SW Schablone hinzu. Die DDS enthält nun neben den Levels der Textur auch die entsprechenden Level der Schablone.

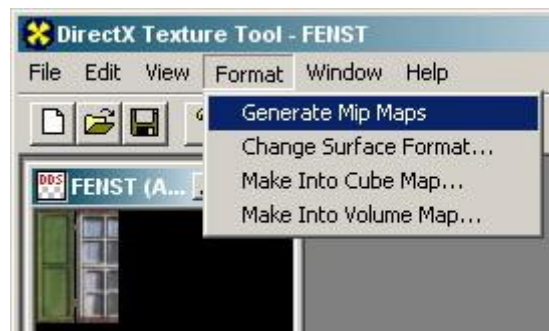


Und so geht's:

Wir öffnen das Programm DXTEX.exe und laden unsere Textur.

Anschließend generieren wir die Mipmap-Level.

>Format< >generate Mip Maps<



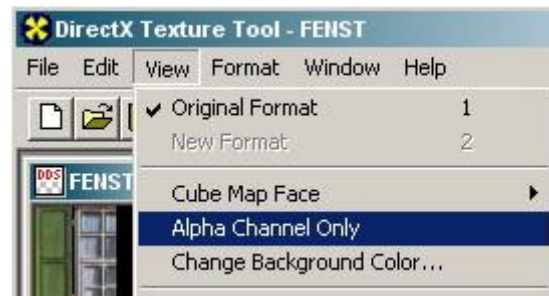
Dann öffnen wir die „Texturschablone“

>File< >Open Onto Alpha Channel.....<

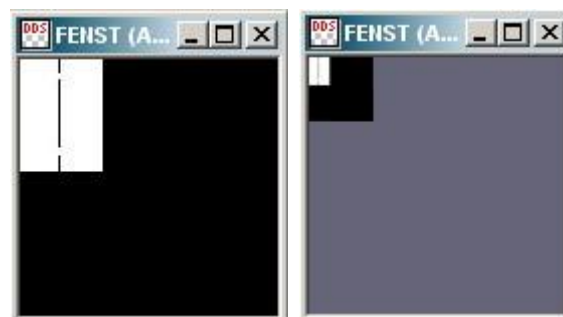


Um die korrekte Ausführung zu überprüfen, checken wir das mit

>View< Alpha Channel Only<



Mit den Pfeiltasten „auf“ – „ab“ können wir auch die MipMap-Level überprüfen.



Nun müssen wir nur noch die Textur als \*.DDS speichern. Beide Texturen, sowohl die BMP als auch die DDS, kommen ins Verzeichnis „Parallels“.



Der letzte Schritt, um die Textur ins Programm einzubinden, besteht nun noch darin, diese ins Texturen Skript einzutragen, damit das Programm die Textur auch findet. Im gleichen Verzeichnis wie alle Texturen befindet sich die Datei Texturen.txt. Hier werden alle Texturen sowie ihre Eigenschaften eingetragen. Die Zeile für unsere Textur müsste so aussehen:

```
Textur{ id(xxx)   name(Fenster)  billboard()  automipmap()  magfilter(1)  minfilter(5) }
```

Dabei steht xxx für die Textur ID-Nummer bei der die ersten Nullen weggelassen werden. Wenn die Textur ID 0024 ist, dann wird hier nur 24 eingetragen.

Bei Name steht der Dateiname ohne die Endung.

Eine Bedeutung der einzelnen Begriffe am Ende des Kapitels in einer Zusammenfassung.

### **Verwendung von Texturen im Nostruktur**

Zunächst möchte ich mal die Verwendung von Texturen erklären.

Texturen werden mit Hilfe von Koordinaten in die jeweiligen Vertices eines Polygons übertragen.

Man muss sich über einer Textur ein Raster vorstellen, dessen Teilung 100 oder besser 1000 ist. Dabei spielt es keine Rolle ob die Seitenverhältnisse gleich oder ungleich sind.

Man muss sich dies so vorstellen:

*Anmerkung: um eine bessere Darstellung zu erreichen, habe ich hier nur eine Teilung von 10 vorgenommen.*

Die Koordinaten für Y, also die Höhe beginnen am unteren Rand mit Null und enden am oberen Rand mit 1

Wollten wir die komplette Textur in ein Polygon einfügen, müssten wir bei den Vertices für Y folgendes eintragen:

oben links Y= 1

unten links Y= 0

oben rechts Y= 1

unten rechts Y= 0

Die Koordinaten für die X Richtung, also die Waagrechte beginnen links mit 0 und enden rechts mit 1.

Im Falle der kompletten Verwendung der Textur müssten dann die X Werte in den Vertices so aussehen:

oben links X= 0

unten links X= 0

oben rechts X= 1

unten rechts X= 1

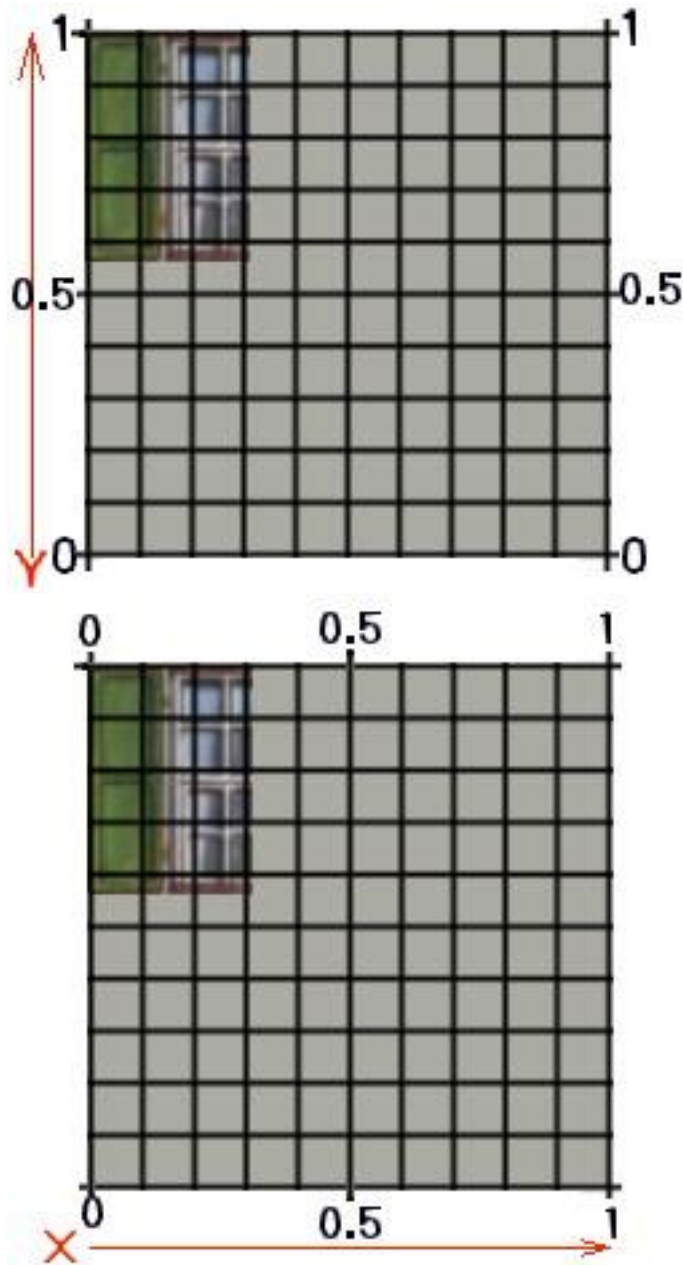
Wir möchten aber nur das Fenster im Polygon haben.

Die obere linke Ecke des Fensters befindet sich sowohl für die X als auch die Y Achse bei 0

Die Unterkante in Y Richtung bei ca. 0.58 (hier nur geschätzt).

Die rechte Fensterkante in X Richtung bei ca. 0.31

*Zu bemerken wäre hier noch, dass das Programm keine Kommas akzeptiert. An Stelle eines Kommas wird hier ein Punkt gesetzt.*




( 5 )

Die Einträge in den Vertices für die Texturkoordinaten des Fensters sind also folgende:

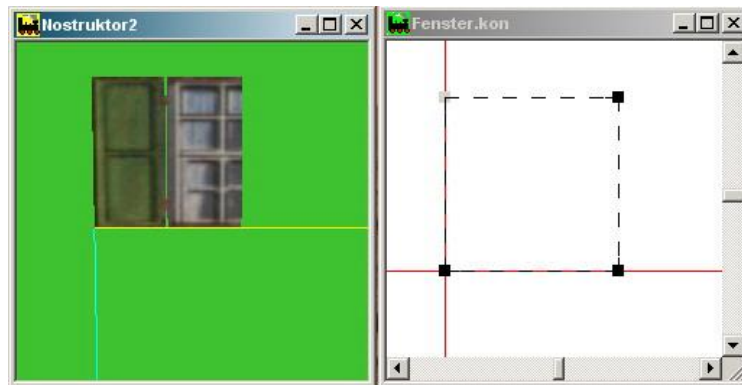
oberer linker Vertex:  $X=0$   $Y=1$   
unterer linker Vertex:  $X=0$   $Y=0.58$   
oberer rechter Vertex:  $X=0.31$   $Y=1$   
unterer rechter Vertex:  $X=0.31$   $Y=0.58$

*Die Dezimalwerte wurden hier nur grob geschätzt (pi mal Auge). Genauere Ergebnisse erzielt man, indem man die Teilung 100 oder besser 1000 nimmt.*

Die Größe und Proportion des Polygons ist natürlich abhängig von der Größe, die man dem Fenster geben möchte. Wie groß ein Fenster ist, dürfte bekannt sein.

Dass die Eingaben über  gemacht werden, dürfte aus dem ersten Kapitel bekannt sein.

So haben wir nun ein halbes Fenster.



Erwähnen sollte ich noch, dass natürlich auch die Textur ID eingetragen werden muss. Dies geschieht gleichzeitig mit der Eingabe der X Y Werte und zwar im linken oberen Vertex. Alle anderen Vertices bekommen bei Textur ID eine 0.

*Wichtig: Vertices dürfen nie die Textur-ID -1 haben, auch wenn dieser Wert im Nostruktur erscheint. Die Eingabe ist auf 0 bzw. die ID-Nummer zu setzen.*

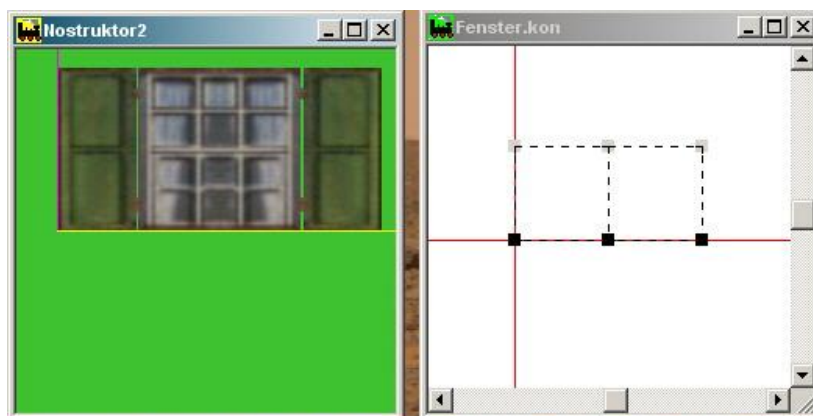
Auch muss, um die Textur sichtbar zu machen, der linke obere Vertex mit einer anderen Farbe versehen werden. Ich nehme für den ersten Test immer weiß. Das kann dann später, wenn die Textur OK ist, je nach Geschmack geändert werden. Bleibt der Vertex schwarz, so sieht man nur ein schwarzes Rechteck.

*Übrigens, für die Fensterhöhe und Breite habe ich die Maße 120 x 120 genommen.*

*Dabei ist dann das eigentliche halbe Fenster 120 x 60, der Rest ist Fensterladen.*

Um nun die andere Fensterhälfte zu konstruieren, markieren wir die linken beiden Vertices und kopieren diese in die Zwischenablage, fügen diese wieder ein und schieben diese Kopie nach rechts bis X 240. Wir löschen die beiden Linien, die nach rechts schauen, und verbinden die oberen und unteren beiden Vertices. Dann müssen wir noch im oberen mittleren Vertex die Textur ID eintragen und die gleiche Farbe zuweisen wie beim linken oberen Vertex.

Fertig ist unser Fenster mit Fensterladen.



( 6 )

Auf diese Weise lassen sich beliebige Bereiche einer Textur in ein beliebiges Polygon einfügen. Dabei ist es auch möglich, Texturen innerhalb eines Polygons zu spiegeln, zu drehen oder zu verzerren.

Noch ein Wort zu ungleichseitigen Polygonen am Beispiel eines Dreiecks.

Wir wissen aus dem ersten Kapitel, dass immer der linke obere Vertex bestimmend für die Farbe ist, und wissen nun auch dass dieser die Textur bestimmt. Wie aber ist das bei einem Dreieck? Auch hier bestimmt der obere Vertex Farbe und Textur, wenn dieses mit der Spitze nach oben zeigt. Zeigt die Spitze nach unten dann bestimmt der obere linke wieder Farbe und Textur.

Die Texturkoordinaten sind hier auch leicht zu finden.

Der Einfachheit halber nehme ich für das Beispiel an, dass die ganze Textur in das Polygon soll.

Als Textur wähle ich die ID 1, eine einfache Brettertextur.

Man muss das betreffende Polygon einfach Gedanklich in die Textur projizieren und sich noch die Koordinaten hinzudenken:

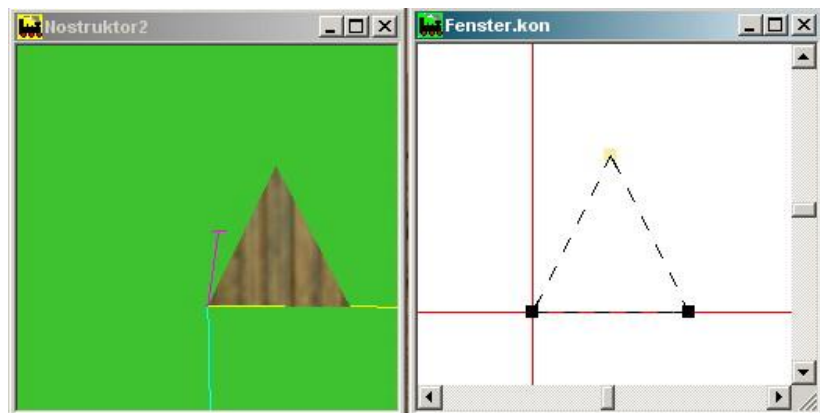


Dabei sind die Werte für die X Y Koordinaten in den einzelnen Vertices wie folgt:

oberer Vertex  
Textur ID = 1, X= 0.5, Y= 1

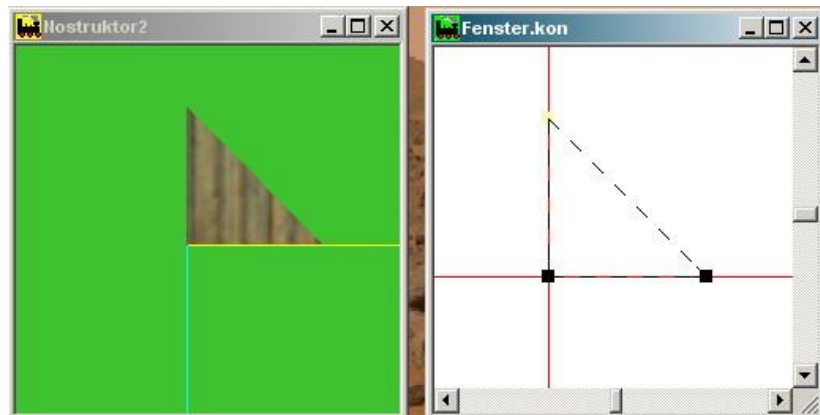
*Zur Erläuterung: der obere Vertex liegt hier in der Mitte der Textur am oberen Rand.*

unterer linker Vertex  
Textur ID 0, X= 0, Y= 0.  
unterer rechter Vertex  
Textur ID 0, X= 1, Y= 0.



Das gleiche hier nochmals :

Oberer Vertex:  
Textur ID= 1, X= 0, Y=1  
unterer linker Vertex :  
Textur ID= 0, X= 0, Y= 0  
unterer rechter Vertex :  
Textur ID= 0, X= 1, Y= 0



### Wiederholung von Texturen in Polygonen

Wenn wir beispielsweise eine Hauswand mit Ziegelsteinen gestalten möchten, dann muss nicht die ganze Hauswand in einer Textur enthalten sein. Da sich Texturen innerhalb eines Polygons wiederholen lassen, sofern diese Option im Texturskript ausgewiesen ist, genügt es, nur einen kleinen Ausschnitt eines Mauerwerkes in der Textur darzustellen. Dadurch kann die Textur sehr klein gehalten werden.

Nehmen wir z.B. die Textur ID 41:

Hierbei handelt es sich um den Ausschnitt einer Backsteinmauer in der Größe 64x64 Pixel. Gehen wir davon aus, dass ein Stein ca. 12,5 cm hoch und 25 cm breit ist, dann handelt es sich um eine Fläche von 1x1 Meter.



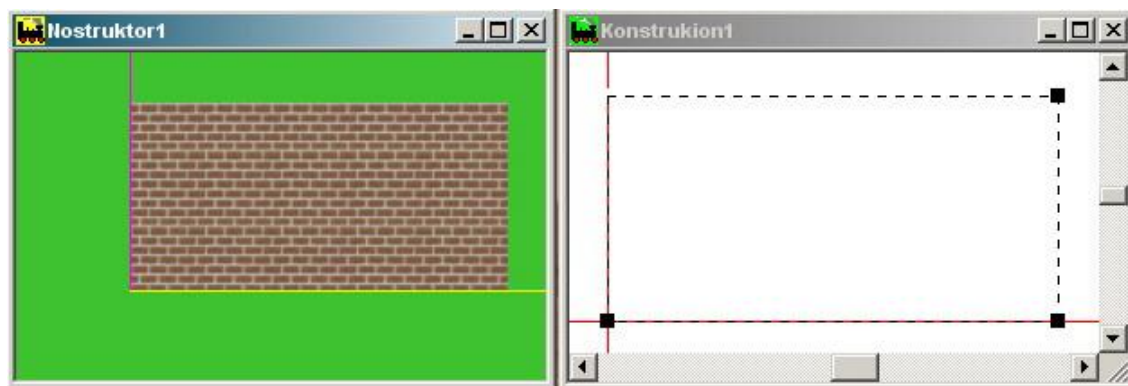
Wenn wir also eine Hauswand mit den Maßen H 2,50m und B 5m machen möchten, so müssen wir die Textur in diesem Polygon in der X-Achse 5x wiederholen und in der Y-Achse 2,5 x. Allerdings muss im Texturskript diese Textur als wiederholbar gekennzeichnet sein.

In der Zeile der Textur 41 muss folgender Eintrag stehen:

`repeat_t()` `repeat_s()` - dabei steht t für Wiederholung in Y-Richtung und s für Wiederholung in X-Richtung.

Sehen wir uns das an einem Beispiel an:

Unsere Mauer ist 2,5 m hoch und 5 m breit



Die Werte in den einzelnen Vertices sehen dann so aus:

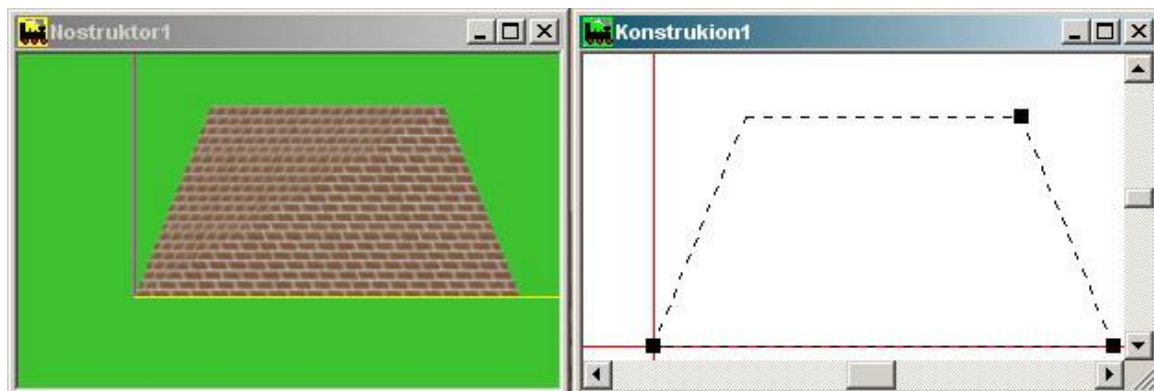
oberer linker Vertex: Textur ID= 41, X= 0, Y= 2.5

unterer linker Vertex: Textur ID= 0, X= 0, Y= 0

oberer rechter Vertex: Textur ID= 0, X= 5, Y= 2.5

unterer rechter Vertex: Textur ID= 0, X= 5 Y= 0

Verändern wir mal unser Polygon in die Form eines Trapezes. Wir ziehen die beiden oberen Vertices um je 1 m nach innen, ohne die X Y Werte zu verändern:



Wir erkennen dass die Textur nun sehr verzerrt ist.



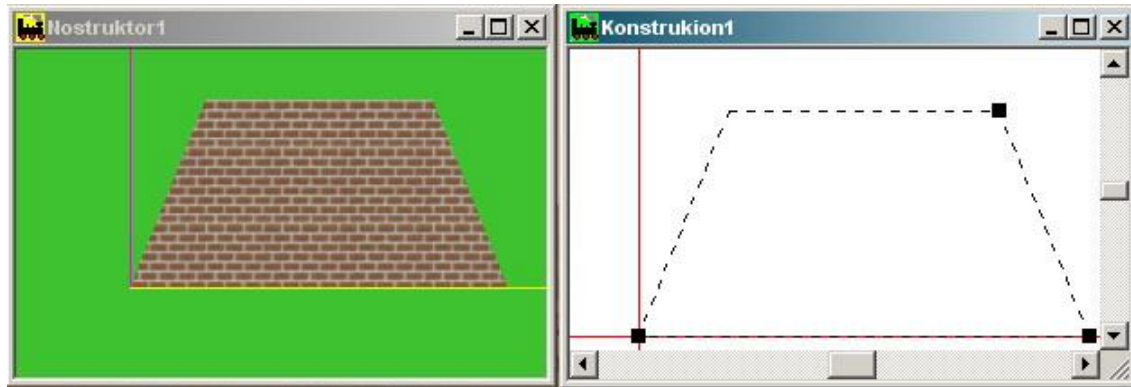
Um diese Verzerrung aufzuheben, müssen die Texturkoordinaten für die X-Achse korrigiert werden:

Der linke Vertex wurde um 1 Meter nach rechts verschoben, also beginnt hier die Texturwiederholung nicht bei 0, sondern ist sie hier bei 1. Auch der rechte Vertex wurde um 1 Meter nach innen also nach links verschoben, somit endet hier die Wiederholung nicht bei 5, sondern schon bei 4.

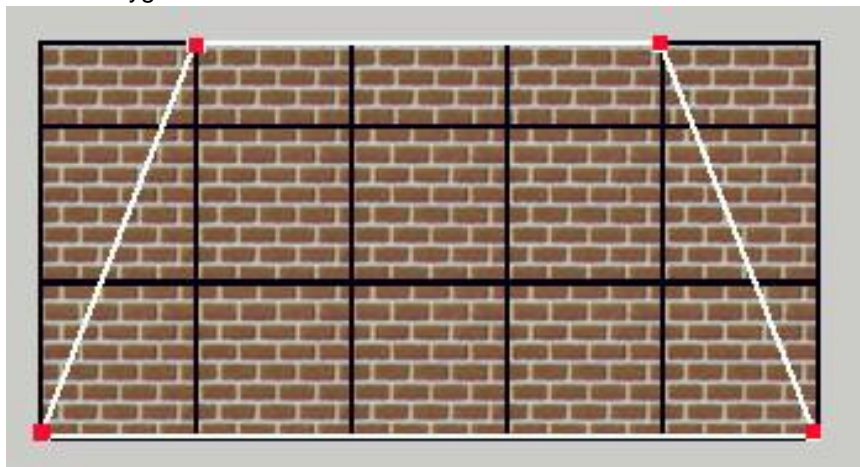
Bei den unteren Vertices ändert sich nichts.

Vertex oben links: X= 1 Y= 2.5

Vertex oben rechts X= 4 Y= 2.5



Um das nochmals zu verdeutlichen, denken wir uns einfach die mehrfach wiederholte Textur über dem Polygon:



In der Waagrechten (X) haben wir die Textur 5 x und in der Senkrechten (Y) 2,5 x. Dabei bildet die weiße Linie unser Polygon.

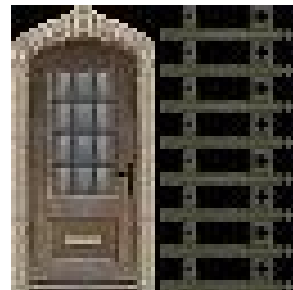
Ich denke, nun dürfte das System wiederholter Texturen in einer Fläche allen klar sein.

*Wichtig: Teile aus Texturen können nicht wiederholt werden, da immer die ganze Textur wiederholt wird. Allerdings kann man in größere Texturen wiederholbare Teile einfügen, wenn diese von einem Textur Rand zum anderen reichen.*

Beispiel:

Diese Textur enthält eine Tür und ein Stück eines Zaunes. Der Zaun soll wiederholt werden. In der Textur wäre das die Y Richtung. Also steht in der Texturen.txt in der Zeile dieser Textur nur der Eintrag: `repeat_t()`

Da der Zaun in der Textur aber hochkant steht, müssen in den Vertices auch die Werte vertauscht (gedreht) werden. Das heißt: der linke obere Vertex erhält die Koordinaten der rechten oberen Ecke der Textur. Der obere rechte Vertex des Polygons erhält die Koordinaten der rechten unteren Ecke des Polygons usw. Die unteren Vertices erhalten bei Y die X Werte der Textur. Man muss sich in Gedanken nur das gedrehte Polygon auf der Textur vorstellen.





## Texturenskripterstellung und Bedeutung der Parameter (erstellt von Software Untergrund)

### Einleitung:

In der ASCII - Textdatei im Verzeichnis Parallels für die Texturen wird die Verknüpfung von Dateinamen zu sog. Textur IDs hergestellt. Diese IDs können im Nostruktur benutzt werden, um Flächen Texturen zuzuordnen. Ergänzend zur ID können einige Eigenschaften der Textur festgelegt werden, diese werden im Folgenden beschrieben.

### *Schlüsselwort Basis:*

Der gesamte Skripttext muss von einem Block aus geschweiften Klammern eingerahmt sein, der mit dem Schlüsselwort Basis bezeichnet ist. Die Parameter dieses Blockes bestehen ausschließlich aus Blöcken mit dem Schlüsselwort Textur.

### *Schlüsselwort Textur:*

Für jede Textur wird ein Block mit Namen Textur angelegt. Die Reihenfolge dieser Blöcke ist egal. In folgender Tabelle werden die einzelnen Parameter eines Texturblocks kurz erklärt. Es folgen Behandlungen einiger Themen, wie Texturfilter, Mipmap und Texturwiederholung.

- id** Unter diesem Parameter wird die eindeutige Textur ID angegeben, und zwar als ganze Zahl, Null ist kein gültiger Wert. Muss unbedingt angegeben werden.
- name** Name der Datei mit dem Texturbild ohne Endung .bmp Muss unbedingt angegeben werden.
- automipmap** Gibt an, dass Mipmaplevel mit verringerter Auflösung automatisch erstellt werden sollen. Es muss kein Wert angegeben werden. Um das Mipmapping tatsächlich einzuschalten, siehe Texturfilter. Default: aus
- billboarding** Gibt an, dass Texel mit schwarzer Farbe, also RGB - Werten 0,0,0 wegzulassen sind. Eine Fläche kann so durchbrochen werden. Default: aus
- magfilter** Gibt den Magnification - Filter an. Gültige Werte sind : 0, 1. Für eine Beschreibung siehe Abschnitt Texturfilter. Default: 0
- minfilter** Gibt den Minification - Filter an. Gültige Werte sind : 0, 1, 2, 3, 4, 5 Für eine Beschreibung siehe Abschnitt Texturfilter. Default: 5
- repeat\_s** Wenn gesetzt, werden Texturen in X-Richtung wiederholt, wenn die Texturkoordinaten den Bereich 0.0f - 1.0f überschreiten. Es muss kein Wert übergeben werden. Default: aus
- repeat\_t** Wenn gesetzt, werden Texturen in Y-Richtung wiederholt, wenn die Texturkoordinaten den Bereich 0.0f - 1.0f überschreiten. Es muss kein Wert übergeben werden. Default: aus
- opacity** Gibt an, dass und in welchem Grade eine Textur eine Fläche durchsichtig erscheinen lassen soll. Gültige Werte sind 0.0 (unsichtbar) bis 1.0 (opak) Default: aus
- MipMapLevel** Schlüsselwort gefolgt von einem Block mit dem Parameter "name", der eine Datei angibt, in der eine Textur mit halbierten Abmessungen steht, die als Mipmap - Level benutzt werden kann. Es müssen sämtliche Mipmaplevel in absteigender Reihenfolge angegeben werden. Für eine Textur mit Abmessungen 128 \* 32 wären also gegebenenfalls Mipmaplevel mit Abmessungen 64\*16, 32\*8, 16\*4, 8\*2, 4\*1, 2\*1 und 1\*1 anzugeben. Default: keine

### **Mipmapping:**

Mipmapping bezeichnet eine Technik, bei der Texturen in größeren Entfernungen mit verringerter Auflösung dargestellt werden. Dies kann Flimmereffekte beseitigen und gibt bessere Kontrolle über die Darstellung von Objekten in größerer Entfernung. Mit dem Parameter `automipmap()` werden die Mipmaplevel automatisch berechnet, indem die Farbwerte jeweils vier Texel zu einem verrechnet werden (Einfache Mittelung). Mit dem Schlüsselwort `MipMapLevel` hat man die Möglichkeit, von Hand erzeugte Mipmaplevel anzugeben. Dies ist zum Beispiel im Zusammenhang mit Billboarding sinnvoll. (siehe unten)

### **Texturfilter:**

Grundsätzlich steht der Renderer vor dem Problem, dass er entscheiden muss, welches Bildelement (Texel) einer durch die 3D -> 2D Projektion unter einem Pixel zu liegen kommenden texturierten Fläche die Farbwerte für das Pixel liefern soll. Es gibt zwei Situationen, in der einen sind die texturierten Flächen soweit weg, dass mehrere Texel in ein Pixel fallen: hier wird der sog. Minifikation - Filter aktiv. In der anderen überdeckt ein Texel mehrere Pixel, hier wird die Darstellung durch den Magnifikationfilter beeinflusst.

#### Werte für den Minifikationfilter:

0: `GL_NEAREST`, das dem Pixelmittelpunkt zunächst liegende Texel bestimmt die Farbe des Pixels.

1: `GL_LINEAR`, die vier dem Pixelmittelpunkt zunächst liegenden Texel werden zur Farbe des Pixels verrechnet.

2: `GL_NEAREST_MIPMAP_NEAREST`, wie `GL_NEAREST`, nur werden abhängig von der Entfernung Mipmaplevel zur Farbbestimmung benutzt.

3: `GL_LINEAR_MIPMAP_NEAREST`, es wird abhängig von der Entfernung ein Mipmaplevel benutzt, die endgültige Farbe des Pixels jedoch aus den vier zunächstliegenden Texeln in dem Mipmaplevel gemittelt.

4: `GL_NEAREST_MIPMAP_LINEAR`, es werden die beiden am besten zu der Entfernung passenden Mipmaplevel benutzt. Relevant ist aber immer nur das Texel, was dem Pixelmittelpunkt am nächsten liegt.

5: `GL_LINEAR_MIPMAP_LINEAR`, es werden die beiden am besten zu der Entfernung passenden Mipmaplevel benutzt. Die Farbwerte, die jeder Mipmaplevel beisteuert, werden aus den dem Pixelzentrum nächstliegenden Texeln gemittelt.

Wenn für eine Textur `automipmap()` oder mit `MipMapLevel` eigene Texturlevel angegeben werden, muss man noch einen der `MIPMAP` - Level für den Minifikationfilter bestimmen, um das Mipmapping auch wirklich einzuschalten.

#### Werte für den Magnifikationfilter:

0: `GL_NEAREST`, es wird das Texel die Farbe eines Pixels bestimmen, dessen Zentrum dem Pixelzentrum am nächsten liegt.

1: `GL_LINEAR`, es werden die vier einem Pixel zunächst liegenden Texel zur Farbbestimmung herangezogen.

### **Texturwiederholung:**

Mit den Parametern `repeat_s`, `repeat_t` gibt man, an, dass die Textur in X bzw. Y Richtung wiederholt werden soll, wenn die Texturkoordinaten den Bereich zwischen 0 und 1 überschreiten. Dies kann im Zusammenhang mit den Texturfiltern zu Problemen führen. Wenn beim Filtern Texelwerte herangezogen werden müssten, die jenseits der Texturgrenzen liegen, können Streifen entstehen, die die Farbe der gegenüberliegenden Texturseite einmischen. Ohne Repeat hätten die verwendeten nicht existierenden Texel die Farben der letzten existierenden.

Auch beim Billboarding kann es passieren, dass unsichtbare Texel an einem Texturrand wieder sichtbar werden. In solchen Fällen kann man entweder die Filtermethoden auf `NEAREST` stellen, oder aber auf `REPEAT` verzichten.

### **Billboarding:**

Texel mit der Farbe RGB: 0,0,0 (schwarz) werden nicht gezeichnet. Man sieht dann an diesen Stellen die Szenerie dahinter. Beim Auto - Mipmapping können hier Probleme auftreten, da vollautomatisch nicht immer korrekt bestimmt werden kann, ob ein Texel jetzt unsichtbar ist oder nicht. Etwa, wenn zwei unsichtbare und zwei sichtbare zu einem verrechnet werden. In solchen Fällen kann es sinnvoll sein, spezielle Mipmaplevel von Hand anzugeben.

# Hilfe zum Nostruktur für EEP

## Kapitel 3

### Arbeiten mit dem 3D Fenster (mod2 Dateien)

#### Grundsätzliches

Im 3D Fenster (mod2) werden alle Komponenten eines Modells zusammengefügt, positioniert, und evtl. noch skaliert. Auch werden hier die Einstellungen für Z-Offset, Face Culling und Smooth Shading vorgenommen. Die Erläuterung der einzelnen Begriffe findet ihr in den einzelnen Abschnitten dieses Kapitels.

#### Noch eine Anmerkung:

Manche Modelle benötigen mehrere mod2 Dateien, zum Beispiel Modelle mit beweglichen Teilen wie Rollmaterial (Räder, Fahrgestelle), bewegliche Türen oder Auslegerarme (Kräne). Die Zugehörigkeit der einzelnen Komponenten (mod2) werden in der GSB festgelegt. Doch davon später.

Wie schon erwähnt findet im 3D Fenster der „Zusammenbau“ der einzelnen Kon-Dokumente zum kompletten Modell statt. Hier können auch noch Korrekturen bezüglich der Grösse und Position des gesamten Modelles vorgenommen werden.

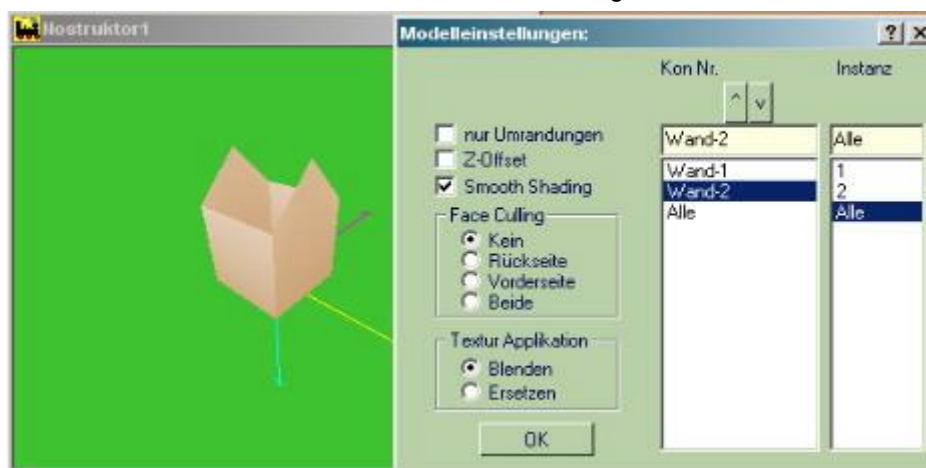
Die Optionen im 3D Fenster im Einzelnen:

#### Smooth Shading

Mit der Option Smooth Shading gibt man den Flächen zum einen ein realistischeres Aussehen, und zum andern kann damit der 3D Effekt betont bzw. erhöht werden. Hauswände sind durch Schmutz etc. unten immer etwas dunkler als nach oben hin. Gibt man nun bei einer Hauswand den unteren beiden Vertexen einen etwas dunkleren Farbton und den oberen einen etwas helleren, so werden diese Farben kontinuierlich miteinander verrechnet und es entsteht ein Farbverlauf.

Dazu gehen wir bei aktiviertem 3D Fenster über  in das Menü „Einstellungen“

Im folgenden Menü sehen wir im linken der beiden Fenster eine Auflistung aller ins 3D Fenster eingefügten Kon Dateien, im rechten Fenster sehen wir die einzelnen Instanzen der jeweils aktiven Kon Dateien. In unserem Beispiel haben wir die Wand 1 und 2 eingefügt, wobei Wand-2 aktiviert ist. Im rechten Fenster sehen wir, daß Wand-2 zwei Instanzen enthält. Da wir den Smooth Shading Effekt an beiden Instanzen wollen, markieren wir „Alle“ und machen ein Häkchen bei „Smooth Shading“. Da aber auch die anderen Wände der Wand-1 mit Smooth Shading versehen werden sollen, können wir auch im linken Fenster „Alle“ markieren und Smooth Shading einstellen.



Dabei muss bei „Textur Applikation“ „Blenden“ gesetzt sein.

### Textur Applikation

Hier zunächst eine Erläuterung zu „Textur Applikation“

In der Regel werden Flächen mit Texturen belegt. Eine Textur hat jedoch meist eine Eigenfarbe. Soll diese alleine für die Farbgebung maßgebend sein, so wird „Ersetzen“ gesetzt. Die Farbe des Polygons (Vertexfarbe) wird in diesem Falle nicht angezeigt, sie wird vollständig unterdrückt. Wird „Blenden“ gesetzt, so werden die Farben der Textur und die Farben des Polygons addiert. Für das Smooth Shading muss „Blenden“ gesetzt sein, da die Textur nicht schattiert werden kann. Diesen Umstand kann man nutzen, indem man Texturen welche nur aus Grauwerten besteht, also keine „Farben“ enthalten, mehrfach einsetzen kann. So kann z.B. eine Türe welche in der Textur nur aus Grauwerten besteht, in allen erdenklichen Farben dargestellt werden, ohne jedesmal eine neue Textur zu machen. Je heller dabei die Textur ist, umso mehr Möglichkeiten hat man bei der nachträglichen Farbgestaltung und Schattierung.

Ein Beispiel an der Textur 1441 (Mauersteine) auf unserer Hauswand:

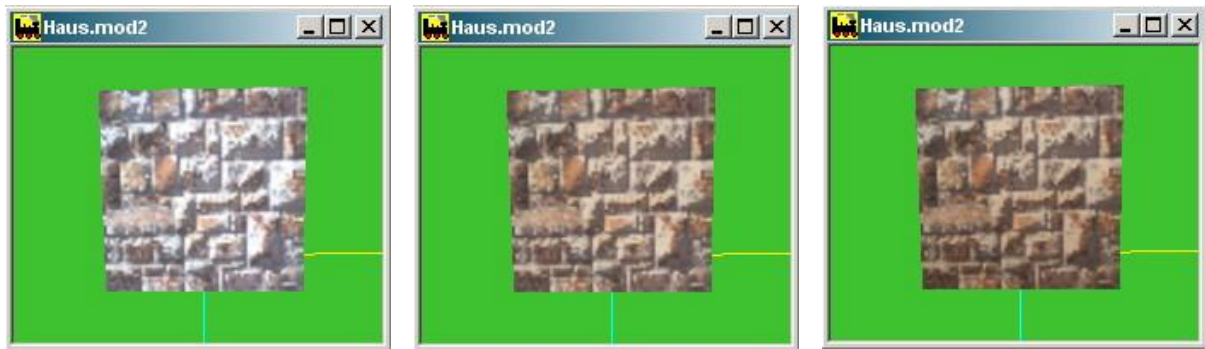


Bild 1,  
Einstellung in Textur Applikation  
„Ersetzen“

Bild 2,  
Einstellung in Textur Applikation  
„Blenden“

Bild 3,  
Einstellung in Textur Applikation  
„Blenden“ und zusätzlich  
„Smooth Shading“

Da diese Textur nur wenige Farbtöne enthält, können mit einer entsprechenden Färbung des Polygons, verschiedene Mauerfarben dargestellt werden.

### Face Culling

Betrachten wir uns mal ein Haus – wir sehen immer nur maximal 2 Seiten. Auch die Wände die wir gerade sehen haben eine Rückseite die wir NIE sehen, egal von welcher Seite wir das Haus betrachten. Aber auch die Seiten die wir im Augenblick nicht sehen sind ja vorhanden, auch für den Computer, also muss er sie auch berechnen. Mit Face Culling jedoch können wir beliebige Seiten einer Fläche unsichtbar machen.

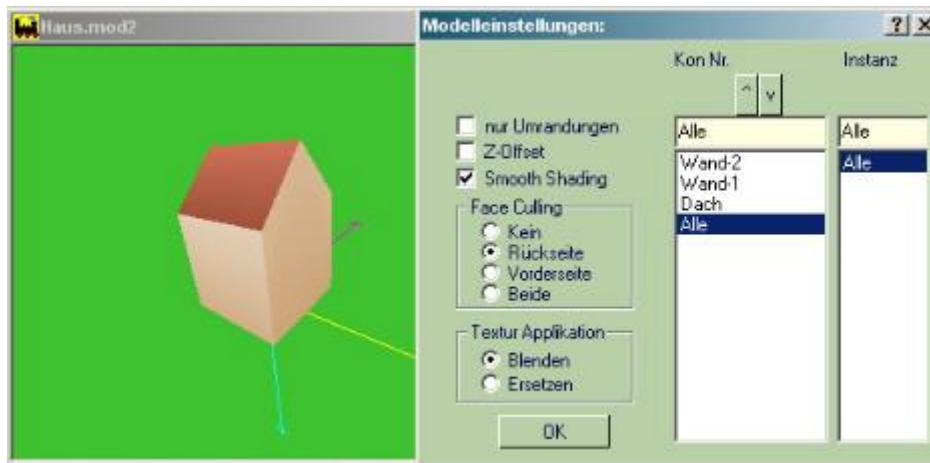
*Für den Computer heisst das: was nicht sichtbar ist, ist auch nicht vorhanden und was nicht vorhanden ist muss ich nicht berechnen.*

Folglich blenden wir alle Seiten der Flächen die wir sowieso nie sehen einfach aus. Wir müssen nur darauf achten, immer die richtigen Seiten auszublenden. Jedes Polygon hat ja eine Vorder- und eine Rückseite.

Am Beispiel der vorderen und hinteren Hauswand sieht das so aus:

Die erste Instanz unserer Wand-1 ist vorne. Hier wird die Rückseite ausgeblendet. Die zweite Instanz der Wand-1 ist hinten. Haben wir diese Wand um 180° gedreht, so muss auch hier die Rückseite ausgeblendet werden. Haben wir diese aber nicht gedreht, sondern nur nach hinten gesetzt, so zeigt hier die Rückseite der Wand nach hinten, ist also aussen. Somit muss hier die Vorderseite ausgeblendet werden. In diesem Falle muss ich die Instanzen einzeln bearbeiten. Deshalb achte ich schon beim Einsetzen der einzelnen Instanzen darauf, daß sie mit der „richtigen“ Seite nach aussen zeigen. Es kann sehr unübersichtlich werden, wenn sehr viele Polygone im 3D Fenster sind und man das Modell ständig drehen muss um herauszufinden ob nun auch wirklich überall die richtigen Seiten ausgeblendet wurden.

Gehen wir in unserem Beispiel davon aus, daß alle Wände richtig gesetzt wurden, so sehen die Einstellungen wie folgt aus (dabei habe ich auch schon das Dach eingesetzt),



Markiert sind Alle, Smooth Shading ist aktiviert, bei Textur Applikation ist Blenden gesetzt. Kippen wir nun das Modell nach hinten, so bemerken wir, da alle hinteren Wände von dieser Seite aus nicht sichtbar sind. Drehen wir aber das Modell um die eigene Achse, so sind die Wände sichtbar. Oder andersrum, zoomen wir uns in das Haus hinein, also betrachten wir alle Wände von innen, so dürften wir bei richtiger Anwendung nichts sehen.

### **Z-Offset**

Z-Offset dient dazu dicht voreinander liegende Polygone optisch voneinander klar zu trennen. Es verhindert den „Flimmereffekt“ an Modellen, bei denen Flächen mit nur sehr geringem Abstand voreinander liegen.

Nehmen wir unser Haus als Beispiel. Wir möchten nun Fenster in unserer Hauswand. Klar – wir könnten nun die Fenster gleich in unsere Hauswand zeichnen. Damit hätten wir keine Probleme, da ja dann Hauswand und Fenster eine Fläche bilden. Damit würden wir aber gleich zwei neue Probleme schaffen. Zum einen wäre es nicht möglich sowohl für die Hauswand als auch gleichzeitig für die Fenster eine Textur zu verwenden. Die Texturwerte für die Fenster würden die Texturwerte für die Hauswand auf den Kopf stellen. Also müssen wir für die Fenster ein neues Dokument erstellen und diese dann auf die Hauswand „kleben“. Diese Polygone müssen aber dann etwas vor den Wänden stehen, damit sie nicht mit diesen „verschmelzen“.

Damit ich nun aber bei den Fenstern nicht alle Koordinaten neu eingeben muss (ich bin eben faul), gehe ich so vor:

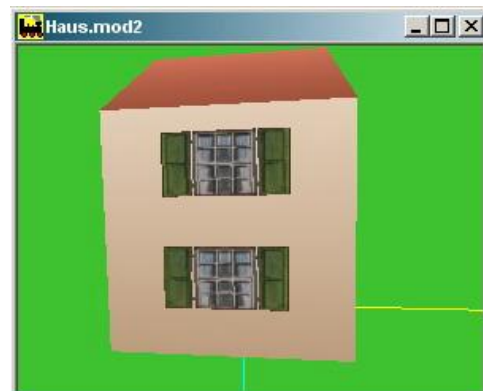
Ich entferne die Wand-1 aus dem 3D Fenster, speichere die Kondatei unter dem Namen „Fenster-1“ erneut ab, öffne wieder die Datei Wand-1 und füge sie wieder ins 3D Fenster ein. Natürlich gehen dabei die Einstellungen für Face Culling usw. verloren. Aber diese Einstellungen nimmt man in der Regel erst am fertigen Modell vor, bevor dieses dann endgültig gespeichert wird.

Nun nehme ich die Datei Fenster-1 und zeichne dort meine Fenster ein. Bin ich mit der Position und der Einteilung fertig, dann lösche ich alle Linien und Vertexe der Wand. Damit aber die Fenster nicht auf gleicher Ebene mit der Wand liegen, gebe ich allen Vertexen bei Höhe Z einen Wert zwischen 4 und 8.

So steht nun das Fenster 4 cm vor der Hauswand.

Das ganze hat aber noch einen kleinen Haken. wenn wir uns vom Objekt entfernen bzw. weg-zoomen, dann kommt es zum sogenannten Flimmern.

Um dies zu verhindern müssten wir das Fenster noch weiter von der Wand entfernen. Aber selbst bei 20 cm würde irgendwann dieser Effekt eintreten und aus der Nähe würde das ja auch nicht gut aussehen, wenn ein Fenster fast einen halben Meter von der Wand weg ist.

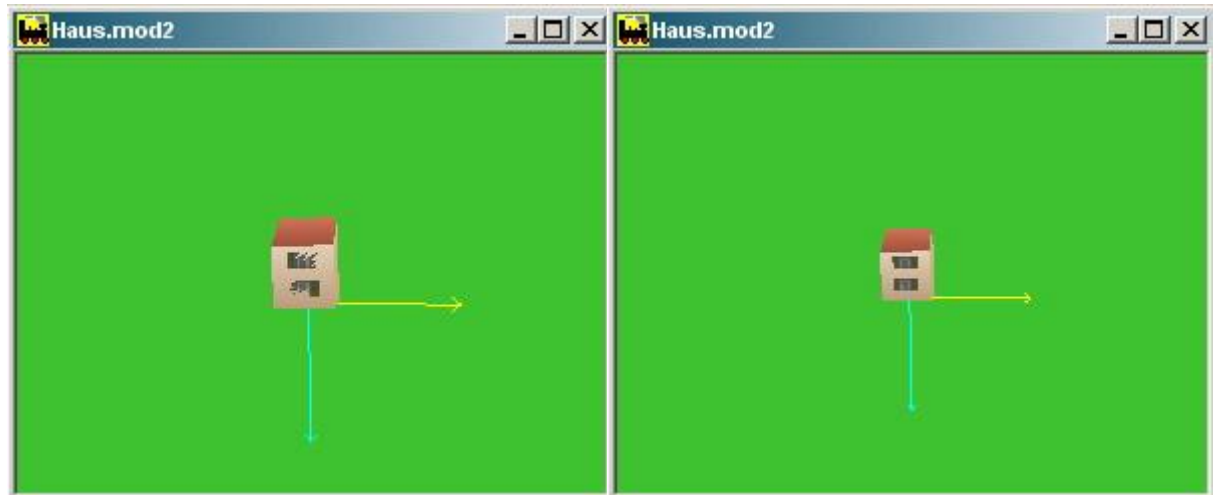




Das kommt daher, weil mit steigender Entfernung Proportional auch der Abstand zwischen Fenster und Wand abnimmt. Das Programm sieht hier wie das menschliche Auge. Halte ein Lineal waagrecht von deinem Auge weg – die vorderen Teilungsstriche kann man noch erkennen, aber je weiter weg die Teilungsstriche sind um so stärker „verschmelzen“ sie miteinander. Der Abstand erscheint immer kürzer je weiter die Striche weg sind.

Genauso „sieht“ das Programm. Wir müssen also das Programm dazu zwingen zuerst die Wand zu zeichnen und dann das Fenster, so daß das Fenster „immer“ vor der Wand ist, auch in grösster Entfernung. Das macht Z-Offset. Wir sagen dem Computer – hier ist eine Wand, davor ist ein Fenster.

Hier das Beispiel mit und ohne Z-Offset:



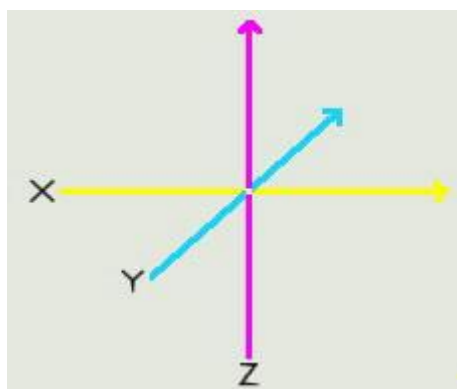
Im linken Bild ist Z-Offset ausgeschaltet. Es ist deutlich zu erkennen, dass die Fenster so langsam mit der Hauswand verschmelzen. Würde man das Ganze aus der Bewegung betrachten, aus einem fahrenden Zug z.B. dann würde das zu einem Flimmern führen.

Im rechten Bild ist Z-Offset eingeschaltet. Hier könnten wir noch weiter weg gehen und würden trotzdem erkennen, daß vor der Hauswand noch etwas ist. Ohne Flimmern !!

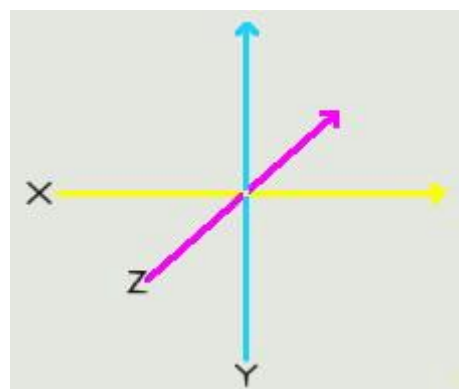
### ***Drehen von Modellen im 3D Fenster***

Wie bereits erwähnt, ist in Bezug auf das Konstruktionsfenster, die Y mit der Z Achse im 3D Fenster vertauscht.

Wenn wir unsere Datei also so Speichern würden, dann läge das Modell auf der Seite.



Lage der Koordinaten im 3D Fenster



Lage der Koordinaten im Konstr.-Fenster

Deshalb muss alles um 90° nach hinten gekippt werden. Es gibt zwei Möglichkeiten das Modell in die richtige Lage zu bringen. Die Erste wäre, gleich alle Kon-Dokumente einschliesslich aller Instanzen zu drehen ( Drehung= X -90),

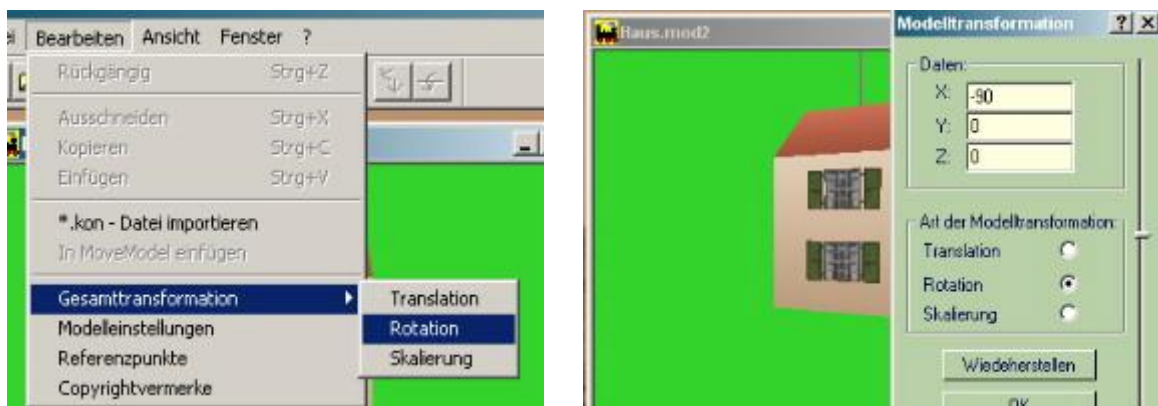
Das hat aber zwei Nachteile. Wenn man einem Polygon auf Grund einer bestimmten Lage bereits für die Y und/oder Z Achse eine Drehung geben muss und dann zusätzlich noch 90° in der X-Achse, so kommt das Koordinatensystem ins Schleudern. Zumindest habe ich es noch nicht geschafft Drehungen über alle 3 Achsen zu erreichen. Bei gleichen Gradangaben heben sich die Drehungen sogar auf.

Der zweite Nachteil, ich muss für jedes Dokument in jeder Instanz die Drehung eingeben.

Das ist mir zu umständlich.

Ich gehe so vor:

Zu Beginn meiner Arbeit kippe ich das Koordinatenkreuz im 3D Fenster nach hinten. Nun füge ich meine Kon's ein. Wenn das Modell fertig ist und ich keine Änderungen mehr vornehmen muss, dann gehe ich zu „Bearbeiten/Gesamttransformation/Rotation“ und gebe dort bei X -90 ein, mit OK bestätigen und das Modell ist in der richtigen Position und kann nun gespeichert werden.



Sollte es vorkommen, daß ich zu einem bereits gespeicherten und erneut geöffneten Modell weitere Kon's hinzufügen muss, so Drehe ich zunächst das erneut geöffnete Modell wieder in die alte Lage, füge die neuen Dateien hinzu, drehe es wieder und kann es nun erneut speichern.

Aber wie gesagt – dies ist meine Methode, jeder muss für sich selbst entscheiden wie er vorgeht.

### ***Verschieben von Modellen im 3D Fenster (Translation)***

Die Modelle im 3D Fenster lassen sich im gleichen Menü auch komplett auf allen 3 Achsen verschieben.

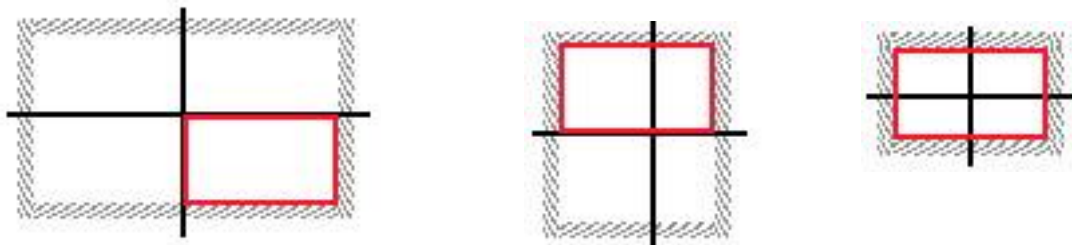
Zur Erläuterung:

Bei grösseren Modellen oder wenn man mehrere Objekte zu einer Mod-Datei vereint, kann es vorkommen, daß das komplette Modell nicht in der Mitte des Koordinatenkreuzes steht.

Dies sollte aber unbedingt beachtet werden. Und zwar aus folgendem Grund:

Im Planfenster von EEP wird jedes Modell mit einem Rahmen versehen welcher die absoluten Aussenmaße, also den am weitesten von der Mitte entfernten Eckpunkt zur Berechnung der übrigen 4 Eckpunkte nimmt. Das sieht so aus:

(schraffierte Linie = Markierung im Planfenster, Rote Linie = tatsächliches Modell (nicht sichtbar im Planfenster von EEP), schwarze Linie = Koordinaten Mitte Z-X Achse)



Wir sehen, das Modell wird im Planfenster nicht in der tatsächlichen Grösse angezeigt, wenn es nicht im Mittelpunkt des 3D Fensters gespeichert ist. Dabei kommt es nicht auf einige Pixel (cm) hin oder her an.

### **Skalieren (vergrössern / verkleinern) von kompletten modellen im 3D Fenster**

Mit dieser Option lassen sich komplette Modelle nachträglich vergrössern oder verkleinern. Dabei hat dies jedoch keine Auswirkung auf die Konstruktionsdatei. Diese Option wird eher selten genutzt, weshalb ich hier nur kurz darauf eingehen möchte.

Ein Beispiel an einem meiner Modelle:

Ich möchte eine Tannengruppe mit drei Tannen als ein Modell machen, wobei jede weitere Tanne doppelt so gross sein soll wie die andere. Um nun nicht alle drei Tannen einzeln über die Instanzen der Kon-Datei mühsam zu erstellen, zumal ja eine Tanne bereits aus mehreren Instanzen besteht, gehe ich wie folgt vor.

Ich mache eine Tanne, setze diese ins 3D Fenster und schiebe sie etwas aus der Mitte, so daß die folgende Tanne, welche ja nun in die Mitte kommt, genug Platz hat.

Dann setze die Tanne erneut ein und Skaliere beide mit dem Faktor 2 bei X, Y und Z.

Ich verschiebe nun beide Tannen wieder um Platz für die dritte zu erhalten und setze nun diese ein.

Jetzt skaliere ich die Gruppe wieder mit dem Faktor 2 bei allen 3 Achsen.

Ich habe nun eine Tannengruppe mit drei unterschiedlich grossen Tannen.

### **Besonderheiten von Mod2 Dateien**

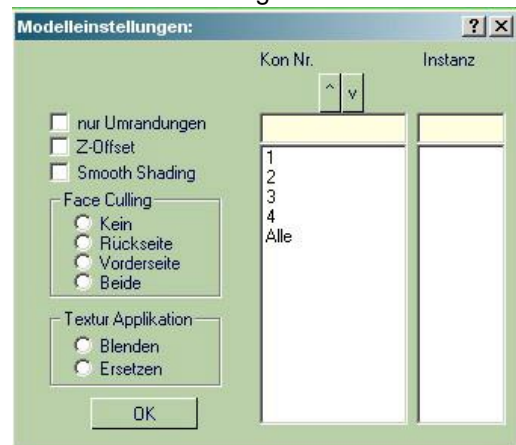
Wir wissen, daß eine Konstruktionsdatei nach dem speichern, schliessen und erneutem öffnen immer wieder verändert werden kann.

Dies ist bei einer mod2 Datei nicht möglich. Zu einer gespeicherten und einmal geschlossenen mod2 Datei lassen sich nach erneutem Öffnen wohl Kondateien hinzufügen, jedoch lassen sich keine Objekte daraus entfernen. Und zwar aus folgendem Grund:

Während der Arbeit wird im linken Fenster im Menü Modelleinstellungen der Dateiname jeder eingefügten Datei angezeigt. Wird das mod2 Dokument gespeichert, geschlossen und erneut geöffnet, so tritt an Stelle der Dateinamen eine fortlaufende Nummerierung.



Anzeige vor dem Schliessen



Anzeige nach dem Wiederöffnen

Fügt man nun weitere Kon's hinzu, so erscheint zunächst der Dateiname. Nach erneutem speichern, schliessen und öffnen werden diese dann ebenfalls in die fortlaufende Nummerierung einbezogen. Deshalb ist es im Nachhinein auch etwas schwierig Einstellungen wie Smooth Shading oder Faceculling usw. vorzunehmen. Oft besteht ja ein Modell aus mehr als 5 Dateien, wobei ja auch noch bei den ein oder anderen mehrere Instanzen hinzukommen.

Wir sehen – bevor wir ein Modell speichern, sollten wir alle Einstellung gemacht haben.

Drehungen, Skalierungen und Verschiebungen können Jederzeit geändert werden.

### **Nachwort**

Alle Optionen die die Beleuchtung betreffen werden in dieser Anleitung nicht behandelt.

Zum einen ist dies zur Zeit der Erstellung dieser Anleitung nicht relevant, da in EEP (noch) nicht möglich, zum anderen habe ich diesbezüglich kaum Erfahrung. Sollte dies jedoch für den ein oder anderen von Interesse sein, so möge er sich bitte an einen Konstrukteur mit entsprechender Erfahrung wenden. Ich bitte um Nachsicht.

# Hilfe zum Nostruktur für EEP

## Kapitel 4 (von Steffen Estler)

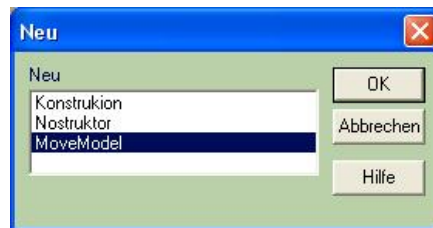
### Arbeiten im Move Model Fenster (GSB)

#### *Vorbereitung:*

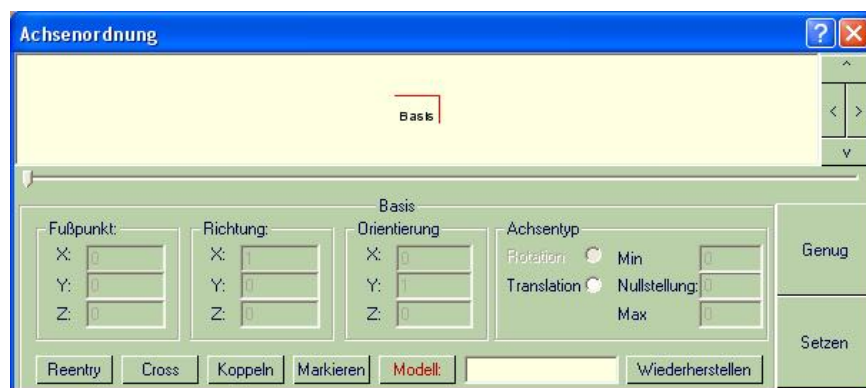
Alle mod2-Dateien sollten ins Verzeichnis „Ressourcen\Modelle\XX1“ kopiert werden. XX steht hierbei für das Konstruktorskürzel, die 1 für die Nummer, falls es schon einen Konstrukteur mit gleichen Initialien geben sollte. Unterverzeichnisse können beliebig angelegt werden. Leerzeichen, Dezimalpunkt und sonstige Sonderzeichen und Umlaute sollten im Pfad- und im Dateinamen keine Verwendung finden, da dies zu unerwarteten Fehlern führt.

### Erstellen eines einfachen Modells als GSB-Datei

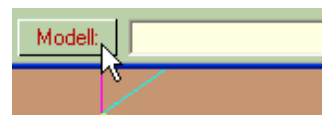
**Neue Datei** erstellen >Datei \ Neu< Move Model wählen



Dialog zur Achsenschnittung aufrufen

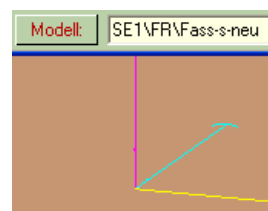


Auswahldialog für die Einzelteile aufrufen...

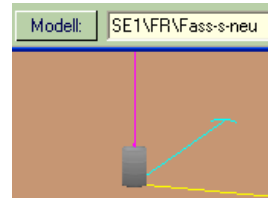


...und das gewünschte Modell (mod2-Datei - in unserem Beispiel ein Fass) auswählen. Die ausgewählte Datei erscheint in der Eingabezeile...

...aber das Modell ist im Konstruktionsfenster noch nicht sichtbar.



Hierzu einfach auf den Button „Setzen“ klicken und schon ist unser Modell als Basis eingebaut.



Durch einen Klick auf den Button „Genug“ beenden wir den Dialog und können unser erstes Modell im Verzeichnis „EEP\Resourcen\Immobilien\.....“ speichern.

### **Wichtig!**

*Die endgültige Position des Modells muß bereits in der Mod2 Datei festgelegt und gespeichert sein, da dieses die Basis darstellt. Die Position der Basis lässt sich im Move Model Fenster (GSB) nicht mehr verändern. Besteht also unser Modell aus nur einer Mod2 Datei muß dieses schon in der Mod2 Mittig stehen.*

*Besteht dagegen das Modell aus 2 oder Mehr Mod2 (in der Regel bei Modellen mit beweglichen Teilen, oder Rollmaterial), so muss nur das Basismodell in der Mod2 auf Mitte gespeichert sein. Alle anderen Mod2 Objekte lassen sich im Move Model Fenster noch positionieren.*

### **Erstellen von Rollmaterial**

Als erstes sollte man sich klar werden welche Bewegungen soll das Rollmaterial im einfachen Gebrauch als Rollmaterial ausführen. Um die Vielfalt des Nostruktors darzustellen werden wir ein Modell kreieren welches ein starre und ein bewegliche Achse mit rotierenden Rädern besitzt.

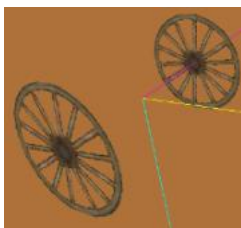
Alle benötigten Einzelteile werden als einzelne mod2-Dateien erstellt und im eigenen Konstrukteursverzeichnis abgespeichert. z.B:..Resourcen\Modelle\XX1

Hierbei ist die richtige Lage der Teile wichtig – bei beweglichen Teilen ist die pinkfarbige Z-Achse die Drehachse um welche sich das Teil bewegen/drehen soll.

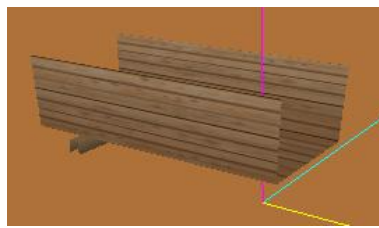
Falls die Modellteile (mod2) noch nicht die richtige Lage/Position besitzen, kann man dies nachträglich im Menüpunkt: Bearbeiten \ Gesamttransformation ändern.

Im Beispiel die Mod2-Dateien eines einfachen Karrens mit drehbarer Vorderachse und beweglichen Rädern.

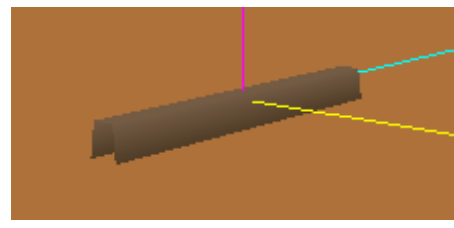
1 x Radsatz.mod2



1 x Aufbau.mod2



1 x Achse.mod2



Da Mod2 Dateien in das MoveModel Fenster mehrfach eingefügt werden können, muss der Radsatz nur einmal als Mod2 gespeichert werden.

Die Hinterachse wird mit dem Aufbau zusammen als Mod2-Datei gespeichert, da diese nicht lenkbar sein muss.

Die Vorderachse soll lenkbar sein, muss also als einzelne Mod2-Datei gespeichert werden

Sind alle Mod2 Dateien gespeichert und befinden sich im richtigen Verzeichnis, können wir mit dem Zusammenbau des Modells beginnen. Hier am Beispiel des einfachen Karrens.

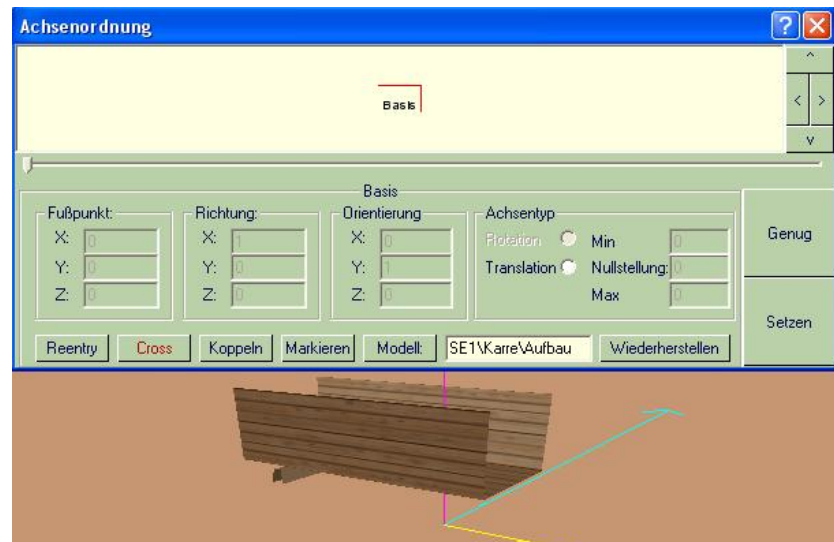


**Neue Datei** erstellen >Datei \ Neu< Move Model wählen

Die nächsten Schritte sind die gleichen wie oben bei einer einfachen Immobilie:

*Dialog der Achsenschnittelung aufrufen – Auswahldialog für Einzelteile (Modell) aufrufen – Datei „Aufbau.mod2“ auswählen – und auf setzen klicken.*

Der Aufbau des Karrens mit der Hinterachse steht nun als „Basis“ im MoveModel Fenster.

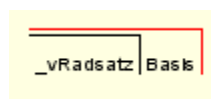


Damit EEP unser Modell auch als Rollmaterial erkennt und richtig in der Spur hält, benötigen wir 2 neue Bewegungsachsen welche die Namen „**vRadsatz**“ und „**hRadsatz**“ tragen. Bitte **keine anderen Namen verwenden** – dies führt im Spiel zu Fehlermeldungen und Abstürzen. Diese beiden Bewegungsachsen sind für jedes Rollmaterial Pflicht!

„**Cross**“ anklicken um die neue Bewegungsachse „**vRadsatz**“ für die Vorderachse anzulegen



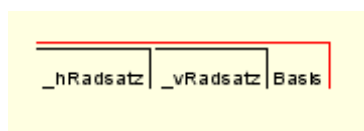
im Achsenordnungsdialog erscheint



„**Cross**“ anklicken um die 2. Bewegungsachse „**hRadsatz**“ für die Hinterachse anzulegen



im Achsenordnungsdialog erscheint

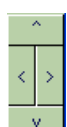


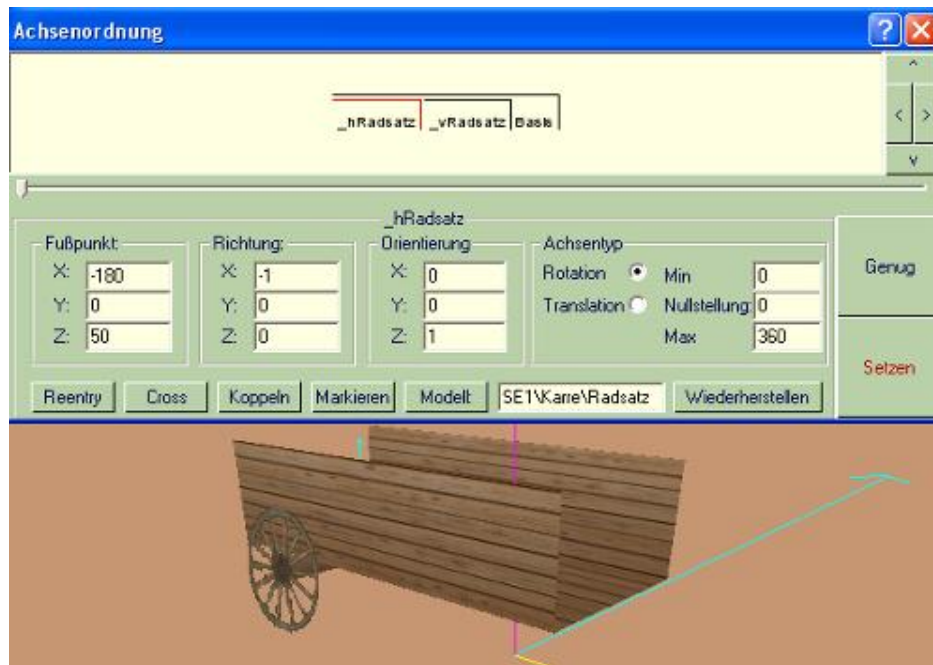
An der hinteren Bewegungsachse sollen nun die Räder montiert werden.

Um die einzelnen Bewegungsachsen auszuwählen, werden die Richtungsbuttons benutzt.

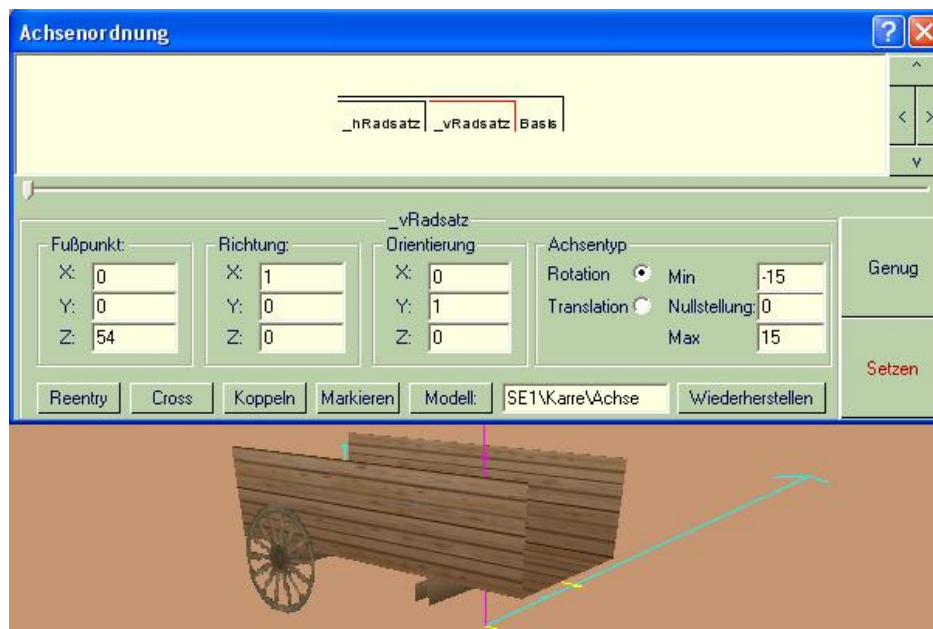
Die aktive Bewegungsachse wird rot dargestellt.

Wir wählen „**hRadsatz**“ aus und geben die Daten wie folgt ein:

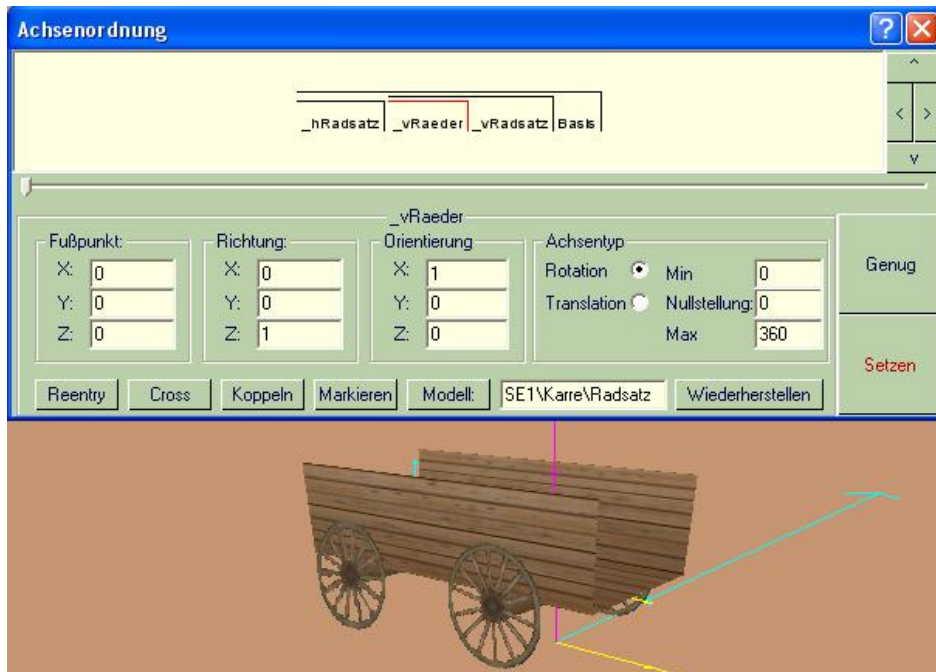




Ähnlich verfahren wir mit der Vorderen Bewegungsachse „**vRadsatz**“ nur mit einigen Unterschieden. Bei „**vRadsatz**“ fügen wir diesmal keinen Radsatz sondern die Wagenachse (Achse.mod2) ein. Daten wie in folgender Abb. eingeben



Damit die Räder der vorderen Achse die Bewegungen (Drehung um Z-Achse) der Bewegungsachse „**vRadsatz**“ übernimmt und eine eigene Bewegung (Rotation um die eigene Achse ausführen kann, müssen wir dem „**vRadsatz**“ eine weitere Bewegungsachse unterkreuzen. Also „**vRadsatz**“ mit Hilfe der Richtungsbuttons auswählen („**vRadsatz**“ wird rot markiert) Klick auf „Cross“ – neue Bewegungsachse Namens „**vRaeder**“ erstellen, „**vRaeder**“ aktivieren und Daten wie folgt eingeben.



Hiermit ist unsere Karre fast fertig Konstruiert. Damit sich unsere Räder während der Fahrt auch wirklich drehen ist noch eine weitere Bewegungsachse nötig – die Geschwindigkeit. Wir erstellen eine weitere Bewegungsachse Namens „**Geschwindigkeit**“ welche unter die Basis gekreuzt wird.

Unsere beiden Radsätze „**hRadsatz**“ und „**vRaeder**“ müssen nun an die Bewegungsachse „**Geschwindigkeit**“ gekoppelt werden.

Dazu wählen wir die Bewegungsachse „**hRadsatz**“ aus und Klicken auf „**Koppeln**“.

Im folgenden Dialog „Kopplungen“ wählen wir „**Geschwindigkeit**“ aus und geben als Kopplungswert = 1 ein.

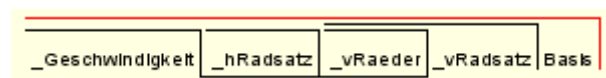
Klick auf „OK“ und der hintere Radsatz ist mit der Geschwindigkeit 1 zu 1 gekoppelt.



Analog koppeln wir „**vRaeder**“ an die Geschwindigkeit.

Will man eine vorhandene Kopplung wieder lösen, muss man als Kopplungswert = 0 eingeben.

Eine Kopplung von Bewegungsachsen erkennt man an der unteren Verbindungslinie



Unserem Modell wurden nun alle benötigten Achsen eingefügt und miteinander gekoppelt. Wir beenden den Achsenordnung - Dialog indem wir auf „Genug“ klicken.

Wir speichern unser Modell im endgültigen Verzeichnis. In unserem Falle „...\\EEP\\Ressourcen\\ Rollmaterial\\ Strasse\\Landwirtschaft\\Karre.gsb“

Zum testen der Achsen klicken wir auf



Hier können wir alle Bewegungsachsen einzeln auswählen und mit dem Schieberegler auf ihre ordnungsgemäße Funktion prüfen.



Sind alle Achsen an ihrer richtigen Position und drehen sich alle Räder in die richtige Richtung speichern wir unser Modell ab. Fertig.  
Sollten sich einige Teile nicht an der gewünschten Lage befinden, ist Nacharbeit an der jeweiligen Achse nötig.

### Erstellen einer gleisbegleitenden Immobilie

Prinzipiell wird eine Immobilie mit Gleisen genau so erstellt wie eine einfache Immobilie, nur die Referenzpunkte der Gleise müssen in der Mod2-Datei noch eingegeben werden.

Hier als Beispiel ein Bahnsteigmodul mit einer Länge von 1000 Punkten.

Das Gleis soll 365 Einheiten von der Mitte des Bahnsteigmodul entfernt sein und eine Höhe von 26 Einheiten (normales Gleis) haben.

Wir öffnen die fertige mod2-Datei.

Öffnen den Referenzpunkt-Dialog mit dem Menüpunkt [Bearbeiten/ Referenzpunkte]

Hier klicken wir auf [Neu] um einen neuen Referenzpunkt anzulegen und geben die Koordinaten wie in der Abb. ein.

Dies ist unser Anfangspunkt des Gleises bzw. Straße.

Nun benötigen wir noch einen Endpunkt für unser Gleis. Dazu wählen wir Referenzpunkt Nr.: 2 aus, klicken auf [Neu] und geben die Koordinaten ein.



Fußpunkt X: 500

Fußpunkt Y: -365

Fußpunkt Z: 26

Für jedes weitere Gleis müssen immer 2 Referenzpunkte neu angelegt werden.

Modell speichern.

– EEP starten – Modell einfügen – am Modell erfreuen oder fluchen weil Nacharbeit nötig ist.

# Anleitung zum Nostruktur für EEP

## Kapitel 1

### Vorwort

Es sind mehrere Ausführungen des Nostruktur im Umlauf, welche sich vor allem durch den Funktionsumfang und der Bedienbarkeit unterscheiden:

1. Trend-NOS: Der offizielle NOS des Trendverlags
2. Home-NOS: wie 1., jedoch mit festem Texturbereich. Die hiermit erstellten Modelle dürfen nicht in Umlauf gebracht werden.
3. NOSPlus: Durch Anpassungen und Erweiterungen optimierte Version des NOS (ausschl. für CTR-Konstrukteure)

Bleibt zu erwähnen, dass mit allen NOS-Versionen das selbe Resultat erzielt werden kann, der Weg zum fertigen Modell aber mit dem NOSPlus durch neue und verbesserte Funktionen am einfachsten ist.

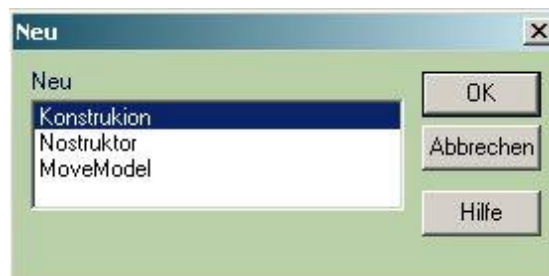
Zusammenfassend bietet der NOSPlus folgende wesentliche Verbesserungen:

- Anpassungen der Dialogfenster und der Menustruktur
- Funktionen zur Vereinfachung der Zylindererstellung
- Verbesserte Eingabegenauigkeit und Cursorsteuerung
- Frei Skalierbarer Winkelkreis
- Neue Funktion Textur-ID zuweisen
- Neuer Dialog für Copyright
- Editor für die Modell-INI-Dateien
- Erweiterte Rasterfunktionen und Eigenschaften
- Neues Maßwerkzeug

Da ich mit dem NOSPlus arbeite, bezieht sich die Anleitung auf diesen - auf Unterschiede zu den beiden anderen Versionen wird jeweils hingewiesen. Die beschriebenen Funktionen lassen sich dann nur auf manuellem Wege durchführen oder bedürfen externer Programme.

Das Konstruktions-Programm **Nostruktur** beinhaltet 3 Arbeitsbereiche, welche in der Folge einzeln beschrieben werden. Jeder dieser Arbeitsbereiche hat spezifische Werkzeuge oder Menüs, welche nur bei entsprechend aktiviertem Fenster aktiv bzw. verfügbar sind.  
(In der Folge sind Arbeitsfenster verkleinert dargestellt) Alle Screens wurden in NOSPlus gemacht.

Wenn sie das Programm aufrufen, so erscheint zunächst ein Auswahlfenster.

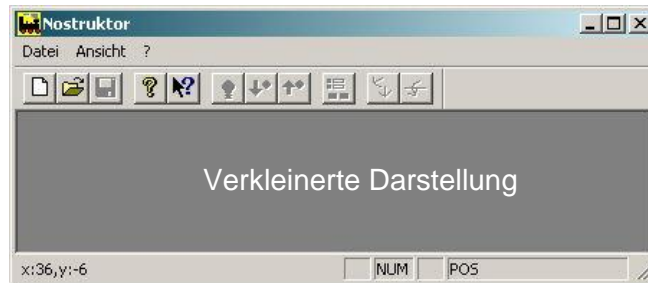


Hier wählen sie den Arbeitsbereich mit dem sie zunächst arbeiten möchten. In der Regel ist dies eine Konstruktions-Datei.


Möchten sie mit einem neuen Dokument beginnen, so markieren sie das entsprechende im Auswahlfenster und klicken auf **OK**






Wenn sie an einer bereits vorhandenen Datei weiterarbeiten möchten, so klicken sie auf **Abbrechen** und das Hauptfenster wird geöffnet.



Unterhalb der Menüleiste befinden sich Buttons welche teilweise inaktiv sind. Diese werden erst aktiv, wenn die entsprechenden Arbeitsbereiche (Fenster) geöffnet bzw. aktiv sind. Die Funktion dieser Buttons werde ich in den entsprechenden Abschnitten erläutern.

 Button öffnet ein neues Dokument im eigenen Arbeitsfenster  
Nach dem anklicken erscheint zunächst wieder das Auswahlfenster (s.o.)

 Button öffnet bereits vorhandene Dokumente.  
Nach dem anklicken erscheint ein Dateiauswahlfenster

  Button zum schnellen speichern eines Dokuments.  
Nur aktiv wenn ein Dateifenster geöffnet ist



Hier wählen sie zunächst den *Dateityp* (Standarteinstellung ist Konstruktion), markieren die entsprechende Datei im Auswahlfenster und klicken auf öffnen.

Zur Kontrolle welche Datei geöffnet wird, dient das Fenster „Dateiname“

### Das Konstruktionsfenster

Die Kopfleiste des Konstruktionsfensters trägt zunächst die Bezeichnung **Konstruktion1**

Nach dem speichern steht hier der Name der Datei (z.B. Wand-1 usw). Dieser Name wird auch in der Kopfleiste des Hauptfensters angezeigt, wodurch immer eine Kontrolle über das gerade aktive Dokument (Fenster) besteht.

In der Statusleiste wird die genaue Position des Mauszeigers innerhalb des Dokuments angezeigt.




















## Die Werkzeuge des Konstruktionsfensters




















Nach dem öffnen einer Konstruktionsdatei, wird eine Werkzeugleiste in das Fenster eingeblendet. Diese ist zunächst Quer am unteren Fensterrand, kann jedoch in jede beliebige andere Position verschoben werden. Ich bevorzuge diese senkrecht am rechten Fensterrand. (ein Monitor ist eben mal breiter als hoch)

Die einzelnen Werkzeuge und ihre Funktion

**Hinweis !** einige Buttons sind erst aktiv bzw. verfügbar, wenn entsprechende andere Funktionen ausgeführt wurden bzw. ein zusätzliches Fenster geöffnet ist.

Mit einigen Werkzeugen ändert sich auch das Aussehen des Mauszeigers. Diesen stelle ich zur Kontrolle neben die entsprechenden Buttons. Das Aussehen kann von der von ihnen verwendeten Version abweichen. Bei den abgebildeten handelt es sich um die verbesserten Varianten.

Button inaktiv / aktiv	Bezeichnung	Mauszeiger (NosPlus)	Funktion
 	<b>ausschneiden</b>		löscht <b>markierte</b> Elemente. Achtung! Linien welche mit markierten Vertexen verbunden sind, werden mitgelöscht Markierte Elemente können auch mittels <b>Entf-Taste</b> gelöscht werden
 	<b>kopieren</b>		kopiert ausgewählte Elemente in die Zwischenablage
 	<b>einfügen</b>		kopiert die Zwischenablage in das aktive Dokument
	<b>Strichlinie</b>	 <b>Teiler</b>	Linie zum zeichnen und teilen von Polygonen, auch für Aussenlinien verwendbar
	<b>Aussenlinie</b>	 <b>Rand</b>	Aussenlinie, auch für Ausschnitte (Löcher) in geschlossenen Polygonen
	<b>Hilfslinie</b>	 <b>HLinie</b>	ursprünglich als Einzel-Linie z.B. für Oberleitung wird jedoch nicht immer korrekt in EEP dargestellt und sollte daher nur als Hilfslinie Verwendung finden
	<b>Vertex</b>	 <b>Vertex</b>	Vertex auf ausgewählte X Y Koordinate setzen
	<b>verbinden</b>		setzt einen Vertex an Linien-Enden bzw. verbindet zwei, sich an den Enden berührende Linien
	<b>Vertex Dialog</b>		durch Doppelklick auf einen Vertex öffnet sich das Dialogfenster für die Vertex Eigenschaften

	<b>Hilfslineal</b>		projiziert ein beliebig positionierbares Hilfslineal mit verstellbarem Winkelmesser auf die Arbeitsfläche
	<b>Skala vertauschen</b>		spiegelt die Skala des Lineals
	<b>verschieben</b>		verschiebt Vertexe. Angeschlossene Linien werden mit verschoben, behalten jedoch am jeweils anderen Ende ihre Position
	<b>bestimmender Vertex</b>		markiert einen Vertex welcher Farbbestimmend für das Polygon sein soll. Nicht für Smoothshading
	<b>Höhe</b>		Überträgt die <b>Höhe ( Z )</b> des zuletzt aufgerufenen Vertex auf andere Vertexe durch anklicken
	<b>Vertexfarbe</b>		überträgt die <b>Farbwerte</b> des zuletzt aufgerufenen Vertex auf andere Vertexe durch anklicken
	<b>Textur</b>		überträgt die Textur-ID des zuletzt aufgerufenen Vertex auf andere Vertexe <b>(Nur NOSPlus)</b>
	<b>Rasierklinge</b>		löscht Vertexe oder Linien. Markierte Objekte können auch mit der Entf-Taste gelöscht werden
	<b>Lichtwerte</b>		überträgt Lichtwerte des zuletzt aufgerufenen Vertex auf andere Vertexe durch anklicken
	<b>einfügen</b>		fügt die gerade aktive Kon-Datei in das 3D Fenster Nur aktiv wenn ein 3D (Mod2) Fenster offen ist
	<b>zoom in</b>		Fensterinhalt des aktiven Fensters vergrößern
	<b>zoom out</b>		Fensterinhalt des aktiven Fensters verkleinern
	<b>markieren</b>		markieren von Objekten oder Bereichen im Kon-Dokument.

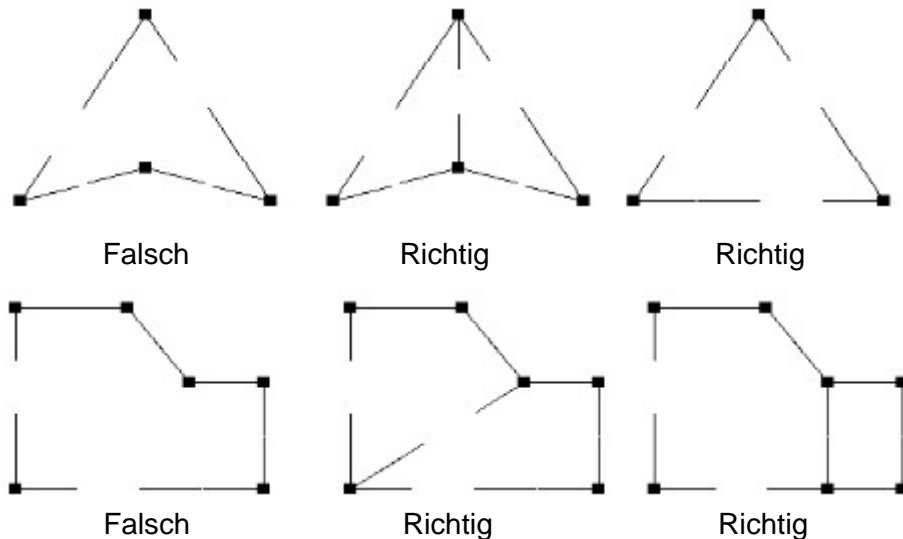
## Arbeiten mit Polygonen

Ein 3D Modell besteht im Prinzip aus einzelnen Flächen, Polygone genannt, welche alle erdenklichen Formen aufweisen können, jedoch immer eine in sich geschlossene Fläche sein müssen um dargestellt werden zu können. Die Polygone werden mit Linien gezeichnet die dann mittels Punkten, Vertexe genannt, miteinander verbunden werden.

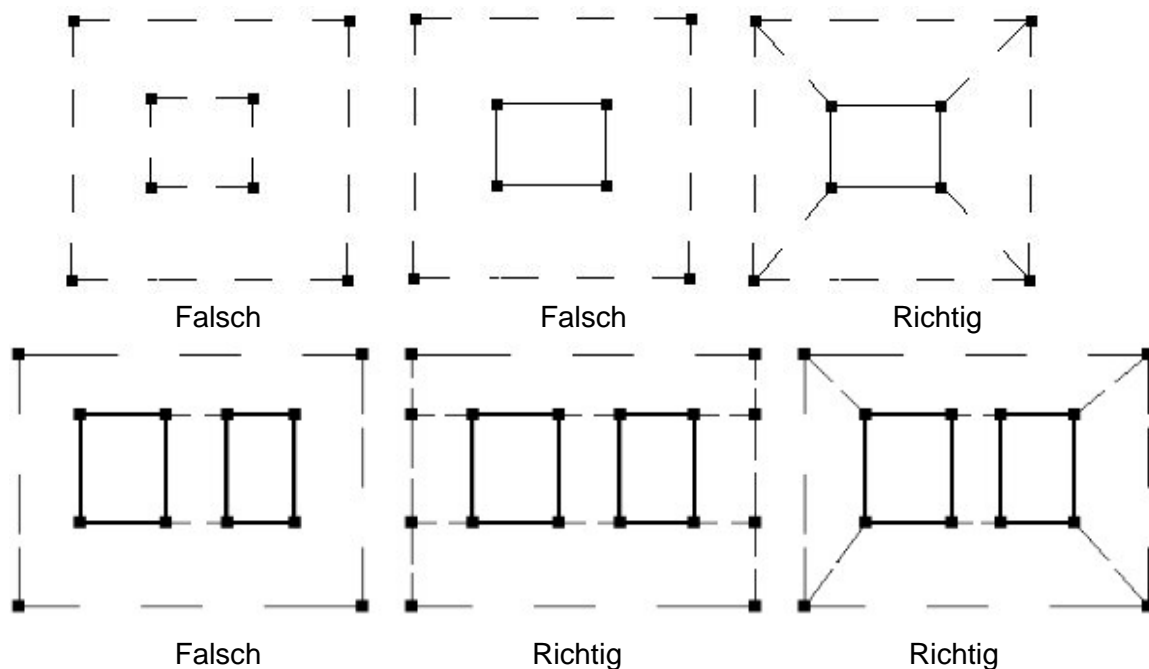
Diese Flächen werden dann in einem zweiten Arbeitsbereich (3D Fenster) zu komplexen dreidimensionalen Körpern zusammengefügt.

Polygone welche unregelmässige Seiten haben, müssen mit Linien so unterteilt werden, daß wiederum in sich geschlossene Polygone entstehen.

Hier einige Beispiele für falsche und richtige Polygone.



Polygone können auch Aussparungen z.B. für Fenster usw. enthalten, wenn diese nicht bereits in einer Textur vorhanden sind. Der Ausschnitt wird mit der **Aussenlinie** gezeichnet wobei hier wiederum mit **Teilungslinien** in sich geschlossene Polygone erzeugt werden müssen. Hier wieder ein Beispiel

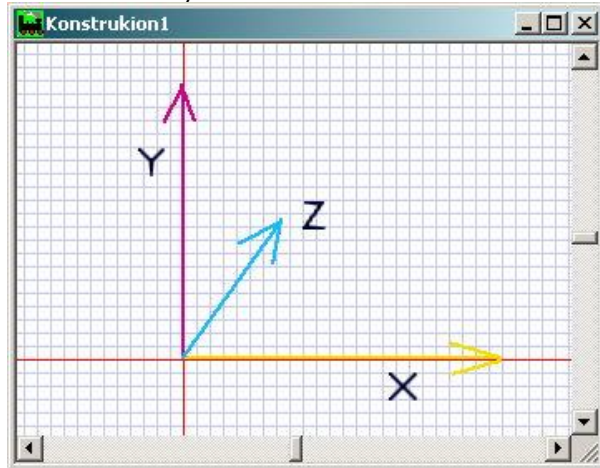


Zum Zeichnen der Aussenlinien geschlossener Polygone sowie für Ausschnitte verwendet man normalerweise die durchgezogene **Aussenlinie**. Ich selbst verwende diese lediglich für Ausschnitte. Um mir während des Zeichnens das Umschalten zu ersparen und auch weil es für mich übersichtlicher ist, verwende ich auch für äussere Begrenzungen eines Polygons die gestrichelte Linie. Lediglich für Ausschnitte **muss** die **Aussenlinie** verwendet werden. Sie stellt praktisch die absolute Aussenkante eines Polygons dar.

#### Hinweis!

Obwohl wir im Konstruktionsfenster in der Draufsicht arbeiten, also unsere Objekte eigentlich von oben betrachten, muss z.B. eine Hauswand von der Seite gezeichnet werden. Dieser Umstand resultiert aus der Tatsache, daß wir keine Vertexe oder Linien übereinander zeichnen können. Die Polygone werden später im 3D Fenster, wo alle Kon-Dateien zusammengebaut werden, in die richtige Lage gedreht.

Koordinatensystem im Konstruktionsdokument:



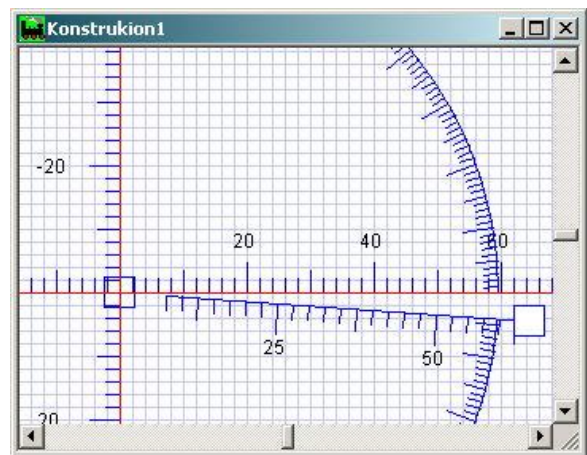
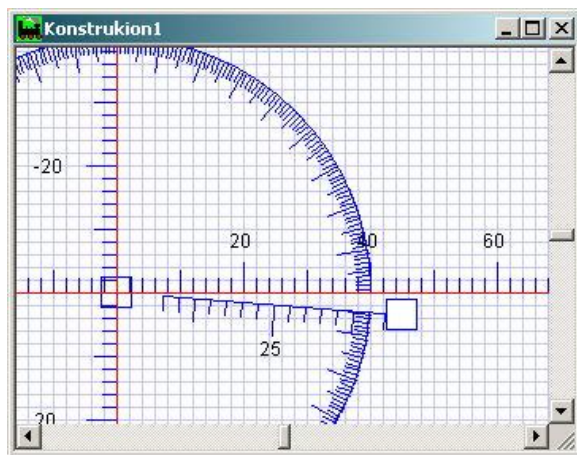
Die Arbeitsfläche im Kon-Dokument zeigt ein Fadenkreuz, wobei im Vergleich zum 3D Fenster hier die Y und Z Achse visuell vertauscht ist.

Ich betrachte hier also die Y Achse als die Höhe und die Z-Achse als die Tiefe.

Doch dazu später mehr.

Der Schnittpunkt des roten Fadenkreuzes liegt auf der Null-Koordinate. Werte in Pfeilrichtung sind jeweils Pluswerte und sind in Dialogfenstern ohne Vorzeichen einzugeben. Minuswerte dagegen werden in Dialogfenstern mit Vorzeichen – eingegeben.

#### Raster und Zeichenwerkzeug



Der Nostruktur verfügt in den beiden grössten Zoomstufen über ein Raster, welches das Zeichnen erleichtert. Der NOSPlus verfügt über ein Raster in den letzten 3 Zoomstufen.

Im Standart NOS sowie im Home-NOS liegen die Linien des Fadenkreuzes sowie die Linien des Lineals zwischen den Rasterlinien. Auch gesetzte Vertexe und Polygonlinien liegen zwischen dem Raster.

Dies wurde im NOSPlus verbessert, wodurch ein noch genaueres und effizienteres Arbeiten möglich ist. Ausserdem ist im NOSPlus der Winkelkreis in der Grösse veränderlich, im Standart NOS hat dieser einen festen Radius.



Nachdem wir nun schon die wichtigsten Funktionen kennen, wenden wir uns der praktischen Seite zu und beginnen mit einem einfaches Modell.

Am anschaulichsten wird das Prinzip des Modellbaus wenn wir ein einfaches Haus bauen.

Bevor wir mit unserem Modell beginnen, wissen wir natürlich wie gross es werden soll.

In unserem Beispiel nehmen wir eine Grundfläche von 10 x 5 m und eine Höhe von 4 m.

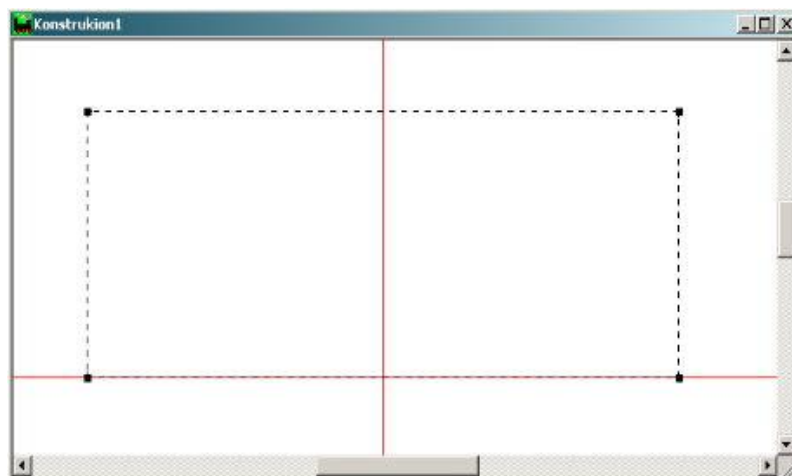
Wir öffnen eine neue Kon-Datei und zeichnen zunächst ein Viereck welches in der X-Achse 1000 cm lang und in der Y-Achse 300 cm hoch ist. Dabei sollte das Polygon mittig im Dokument auf der Grundlinie stehen. Die Grundlinie im Kon-Fenster entspricht der Gundfläche in EEP. Dieses Polygon stellt die beiden langen Hauswände dar.

#### Hinweis!

Der Nostruktur akzeptiert in der X und Y Achse nur gerade Werte. Vertexe welche auf eine ungerade Koordinate gesetzt werden, werden beim abspeichern automatisch auf die nächst höhere gerade Koordinate gesetzt. (Im NOSPlus ist diese Funktion ausser Kraft gesetzt, d.h. Vertexe und Linien Anfänge bzw. Enden werden automatisch nur auf gerade Koordinaten gesetzt).

Sind alle 4 Linien gezogen, wobei darauf zu achten ist, daß sich alle Linienenden an den Eckpunkten auf dem selben Koordinatenpunkt befinden, dann verbinden wir das Viereck zu

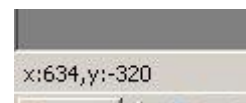
einem geschlossenen Polygon mit Hilfe des **Verbinden** Werkzeuges. 



Die Linien sind verbunden und wir haben ein geschlossenes Polygon.

#### Tip:

Wenn sie Polygone erstellen welche grösser sind als der sichtbare Bereich im Dokumentfenster, dann orientieren sie sich an den Koordinatenangaben in der Statusleiste des Hauptfensters (unten links) Die Angaben hier sind in cm.



Noch einfacher geht es wenn sie die Ansicht vergrössern (entweder Bild auf Taste der Tastatur oder mit dem Button **vergrössern**).




Setzen sie den Mauscursor auf der Null-Linie bei -500 an, halten die linke Maustaste gedrückt und verschieben das Bild mit den Pfeiltasten der Tastatur nach rechts bis sie bei (+) 500 sind. Die Maustaste loslassen und die Linie ist fertig. Ein Linienwerkzeug muss ausgewählt sein.

Nachdem unsere erste Hauswand als Polygon fertig ist, werden wir dieser erst mal eine andere Farbe geben. Da die Vertexe die Farbe eines Polygons bestimmen und diese Standartmässig schwarz sind, müssen wir erst mal die Farbe ändern.




Dies geschieht über den Dialog „**Vertexeigenschaften**“

Dazu nehmen wir das entsprechende Werkzeug  und Doppelklicken auf den linken oberen Vertex. Im Dialogfenster der Vertexeigenschaften klicken wir auf Farbwahldialog. In der Windows Farbpalette wählen wir die gewünschte Farbe und übernehmen diese mit **OK**




#### Hinweis!

In der Regel ist der linke obere Vertex Farbbestimmend für das Polygon. Allerdings, wenn man alle anderen Polygone in ihrer Ursprungsfarbe schwarz lässt, kann es bei manchen Grafikkarten zu Problemen führen, besonders wenn das Modell in Teilen Smoothshading enthält. Deshalb ist es ratsam, bei einfarbigen Polygonen immer ALLE Vertexe mit gleicher Farbe zu versehen, ausgenommen bei der Verwendung von Smoothshading. (dies wird in einem der nächsten Kapitel behandelt)

Um allen Vertexen die gleiche Farbe zuzuordnen, muss nicht für jeden Vertex das „Eigenschaftenfenster“ geöffnet werden. Wenn wir die Farbe des ersten Vertex bestätigt haben, übertragen wir die Farbe mit  durch einfachen Klick auf die anderen Vertexe.

Jetzt haben wir eine Hauswand, wir brauchen diese aber 2mal.

Dazu müssen wir diese nicht neu zeichnen. Wir öffnen mit  das Dialogfenster „**Kopien anlegen**“.



Die Kopie einer Kondatei wird als Instanz bezeichnet. Die Nummerierung der Instanzen beginnt immer bei 0. Somit hat die erste Instanz die Nummer 0. Nun legen wir eine Kopie der Wand an, also eine zweite Instanz. Dazu klicken wir auf „**Neue Instanz**“ und haben nun zwei Hauswände nachdem wir dies mit **OK** bestätigen..

Wir haben jetzt zwei Hauswände, diese liegen aber praktisch übereinander. Also müssen wir die beiden Wände richtig in Position bringen.

Wir wissen – das Haus hat eine Grundfläche von 10 x 5 Meter. Unsere beiden Wände bilden also die lange Seite des Hauses. Folglich müssen wir diese auch auf einen Abstand zur Breite von 5 Metern bringen.

#### Hinweis!

Modelle für EEP sollten möglichst auf Mitte konstruiert sein. Dies erleichtert das spätere Positionieren in EEP.


Da die langen Wände entlang der X-Achse stehen und wir dies erst mal so lassen wollen, müssen wir diese in der Z-Achse um jeweils 2,5 Meter verschieben (wie bereits erwähnt sollten Modelle möglichst auf Mitte gebaut sein). Dabei sollte die Rückseitige Wand um 180 Grad gedreht werden, denn auch im NOS hat ein Polygon zwei Seiten, eine Vorderseite und eine Rückseite. Doch dazu später mehr. Zunächst speichern wir unsere Kon-Datei unter dem Namen „**Wand-1**“ ab.

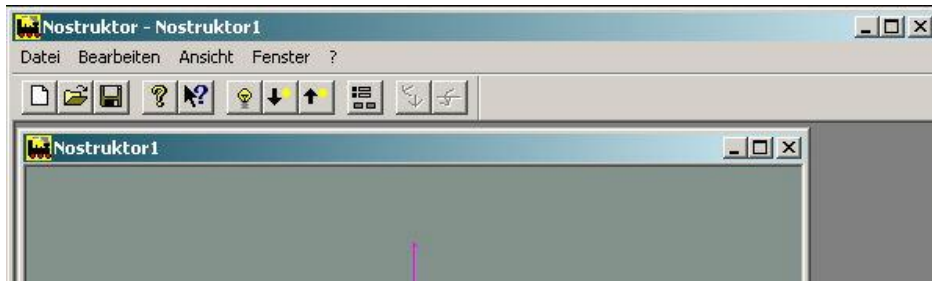
#### Tip


Legen sie sich einen eigenen Ordner für ihre eigenen Konstruktionen an. Dies kann auch während der Arbeit mit ihrem NOS geschehen. Am besten legen sie für jedes Modell einen eigenen Ordner an. Ich habe meine Arbeitsdateien Grundsätzlich auf einer zweiten Festplatte im Ordner „Konstruktionen“ und darunter für jedes Modell einen eigenen Unterordner.

Es gibt nun zwei Möglichkeiten die beiden Wände im richtigen Abstand zueinander zu positionieren. Doch dazu wollen wir uns erst mal dem Nostruktur-Fenster, in Folge 3D-Fenster, befassen.

Im 3D-Fenster wird das eigentliche Modell aus mehreren Kon-Dateien zusammengebaut.

Mit  rufen wir das Auswahlfenster auf, markieren **Nostruktur** und klicken auf **OK**.



Der Button „Einfügen“  in der Werkzeugleiste ist nun verfügbar, jedoch nur wenn ein Konstruktionsfenster aktiv ist.

Folgende Optionen sind bei geöffnetem und aktiviertem 3D-Fenster zusätzlich verfügbar:



Lichtschalter schaltet Lichter ein oder aus, sofern Polygone Lichtwerte enthalten




- Dimmer verdunkelt Schrittweise das 3D-Fenster um Lichtwerte zu beurteilen

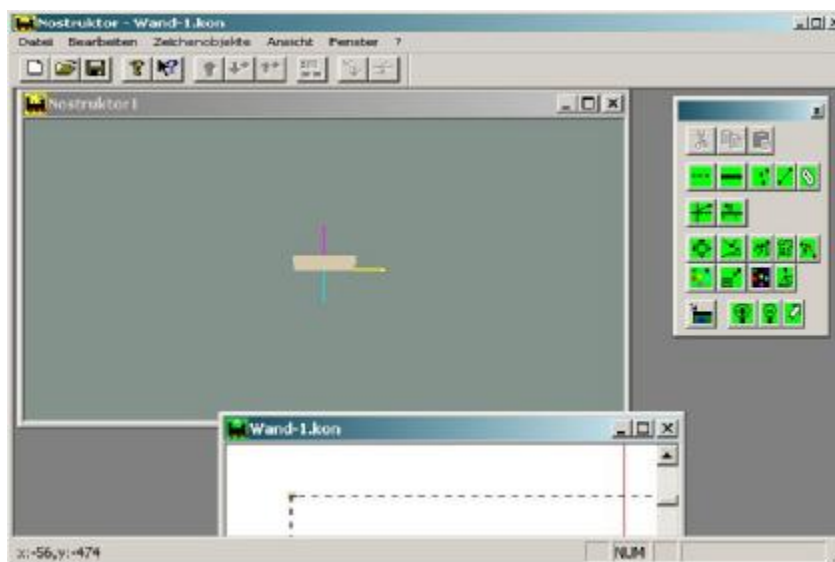


+ Dimmer erhellt Schrittweise das 3D-Fenster



Modelleinstellungen Hier werden besondere Einstellungen wie Smoothshading, Z-Offset, Faceculling usw. vorgenommen. Diese Einstellungen werden wir später im Einzelnen noch kennenlernen.

Wir haben nun unsere Kon-Datei „Wand-1“ sowie das 3D-Fenster geöffnet und können die Hauswände einfügen. Dazu aktivieren wir das Kon-Fenster so daß die Werkzeugleiste sichtbar ist und fügen die Wände mit einem Klick auf  in das 3D-Fenster ein.  
(verkleinerte Darstellung)

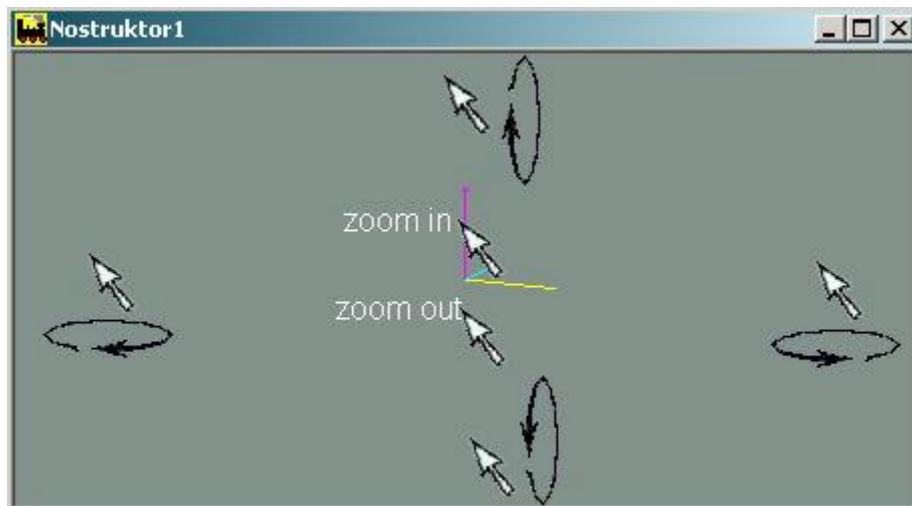


Wie Anfangs bereits erwähnt, sind die Z und Y Achse visuell vertauscht. Wir müssen also die Ansicht im 3D-Fenster zunächst um die X-Achse drehen, damit wir unsere Wand sehen.

### Drehen der Ansicht im 3D-Fenster:

Innerhalb des 3D-Fensters befinden sich sensitive Bereiche zum drehen der Ansicht.

Setzen sie den Mauszeiger innerhalb des 3D-Fensters auf die entsprechende Position und halten sie die linke Maustaste gedrückt bis die gewünschte Position erreicht ist.



Die einzelnen Achsen werden im 3D-Fenster durch verschiedenfarbige Pfeile dargestellt.

**Roter Pfeil** stellt die Z-Achse dar und ist identisch mit der Z-Achse im Kon-Dokument.

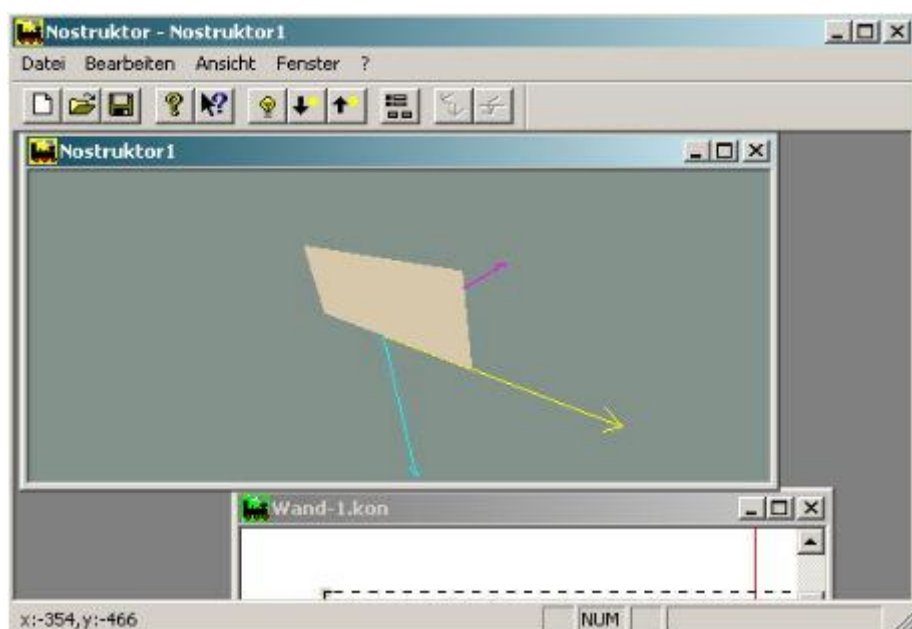
**Gelber Pfeil** stellt die X-Achse oder auch Länge dar und verläuft entlang der Grundlinie.

**Blauer Pfeil** ist die Y-Achse also die Tiefe oder auch Breite.


Mit den Pfeiltasten der 10er Tastatur (der rechte Zahlenblock der Tastatur) lässt sich die Ansicht in alle Richtungen zusätzlich verschieben.

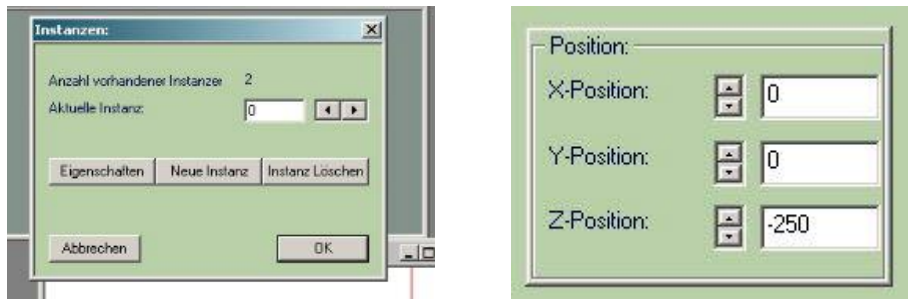
Nachdem wir unsere beiden Wände im 3D-Fenster haben, drehen wir die Ansicht.

Wir haben ja bereits 2 Wände, diese liegen jedoch in der Z-Achse auf gleicher Höhe, so daß wir sie als eine wand sehen. Um nun die Wände in die richtige Position zu bringen gibt es zwei Möglichkeiten.



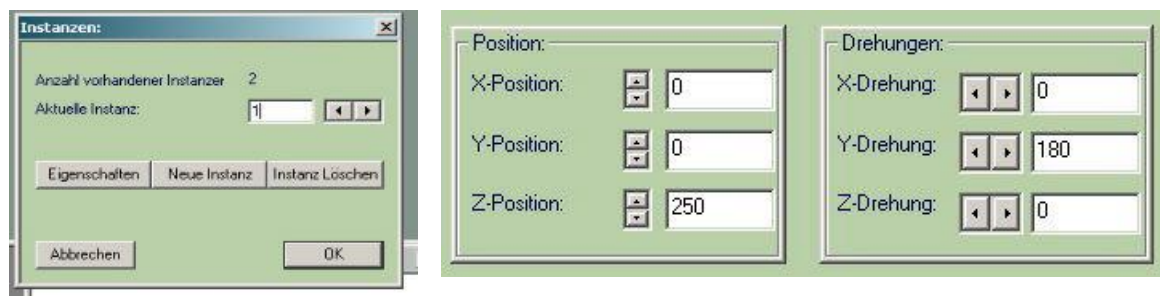
## Möglichkeit 1

Wir aktivieren das Kon-Fenster (Wand-1) und gehen mit  in das Instanzen-Menü, (Standardmässig ist hier die Instanz 0 eingestellt.) dort klicken wir auf „**Eigenschaften**“ und geben bei **Z-Position** den Wert -250 ein.



Wir erinnern uns – das Haus soll 5 m breit werden und ein Modell sollte immer auf Mitte gebaut sein. Deshalb erhalten die Wände den gleichen Abstand von der Mitte aus; jeweils 250 cm, ergibt einen Abstand von 5 m.

Die Einstellungen bestätigen wir mit „**Übernehmen**“ und gehen in die nächste Instanz, entweder durch Eingabe der Instanznummer oder durch klicken auf den Rechtspfeil neben dem Eingabefenster. Die 2. Instanz ist nun aktiv und wir klicken wieder auf „**Eigenschaften**“




Hier geben wir bei „**Z-Position**“ den Wert 250 ein und bei „**Y-Drehung**“ den Wert 180. Warum die Drehung erfahren wir später.

### Hinweis:

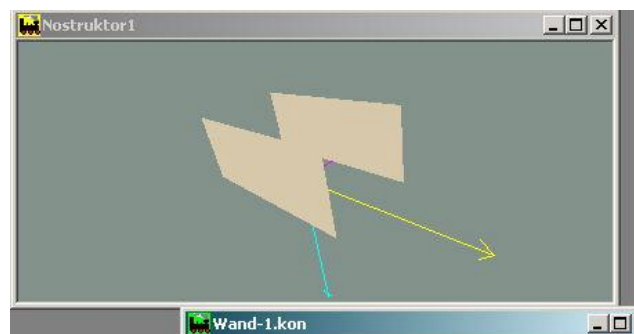
Minuswerte werden mit -Vorzeichen eingegeben, Pluswerte ohne Vorzeichen.

Nach dem „**Übernehmen**“ schliessen wir den Instanzendialog mit **OK**

Wir haben nun zwar die Position der beiden Wände verändert, sehen diese Änderung zunächst aber nicht im

3D-Fenster. Dazu klicken wir einmal auf  und entfernen dadurch die Wände aus dem 3D-Fenster, um sie mit einem weiteren Klick auf diesen Button erneut einzufügen.

Die beiden langen Wände unseres Hauses sind nun an der richtigen Position.



### Hinweis!

Korrekterweise müssten wir die Wände auch um die X-Achse um 90 Grad nach hinten drehen, da ja die Z-Achse die Höhe des Modells darstellt. Viele Konstrukteure machen dies. Ich jedoch möchte mir unnötige Eintragungen in den Instanzen sparen und kippe während des Zusammenbaus meiner Modelle zunächst nur die Ansicht. Später, wenn das Modell komplett ist, drehe ich das komplette Modell auf einmal um 90 Grad. Dann steht unser Modell auch wieder aufrecht, ich erspare mir aber viele Drehungen der einzelnen Kon's.

## Möglichkeit 2

Wie wir wissen arbeiten wir in den Kon-Dokumenten zweidimensional. Trotzdem können wir bereits hier die 3. Dimension nämlich die Z-Achse für die Tiefe nutzen. Diese können wir zwar nicht sehen, aber sie lässt sich erzeugen.


Als erstes nehmen wir die Kondatei aus dem 3D-Fenster und schliessen es vollständig ohne die neuen Einstellungen zu speichern.

### Hinweis!

Wenn man eine Kondatei schliesst ohne diese aus dem 3D-Fenster zu nehmen, kann diese nach dem Wiederöffnen der betreffenden Kondatei nicht mehr aus dem 3D-Fenster genommen werden. Das gleiche gilt, wenn sie ein 3D-Fenster mit eingebetteten Kon-Dateien schliessen und speichern, können die eingebetteten Kondateien nicht mehr entfernt werden wenn das 3D-Fenster erneut geöffnet wird. Es gibt jedoch ein Tool welches auch nachträglich das entfernen bzw. austauschen von Kondateien im 3D-Fenster ermöglicht.

Unser 3D-Fenster ist nun leer. Wir öffnen wieder die Kondatei „**Wand-1**“.

Die beiden Wände stehen ja nun wieder auf gleicher „Höhe“ in der Z-Achse und ich will demonstrieren, wie wir diese an die richtige Stelle bekommen, ohne die lästigen Eintragungen für Position im Instanzenmenü.


Wir maximieren das Kon-Fenster, nehmen das Werkzeug „**Vertexeigenschaften**“  und Doppelklicken auf einen beliebigen Vertex.


Im Koordinatenfenster geben wir bei Höhe (Z) –250 ein.

Warum –250? Unser Haus soll ja 500cm breit (tief) werden und in der Mitte stehen. Also müssen die beiden Wände jeweils um 250 cm in der Z-Achse nach vorne bzw. nach hinten versetzt werden.



Wir bestätigen die Änderung mit **Okay**.


Nun klicken wir auf das Werkzeug  **Höhe**, klicken anschliessend auf alle übrigen Vertexe und übertragen so die Höhe Z auf alle anderen Vertexe. Im „**Instanzenmenü**“ tragen wir lediglich für die 2. Instanz, also die Rückseitige Wand bei „**Drehungen**“ den Wert 180 ein.

Wir erinnern uns; Mit  das Instanzenmenü aufrufen, die zweite Instanz (hier die Instanznummer 1) anwählen, auf „**Eigenschaften**“ gehen und bei Drehungen Y den Wert eingeben.

Setzen wir nun die Kondatei in das 3D-Fenster sehen wir die beiden Wände wieder an der richtigen Position, jedoch haben wir uns einige Eintragungen im Instanzenmenü erspart. Das mag nun bei zwei Instanzen nicht die grosse Zeitersparnis darstellen, aber nicht jede Kondatei besteht nur aus zwei Instanzen. Letztendlich muss jeder für sich entscheiden welche der beiden Möglichkeiten er nutzt.

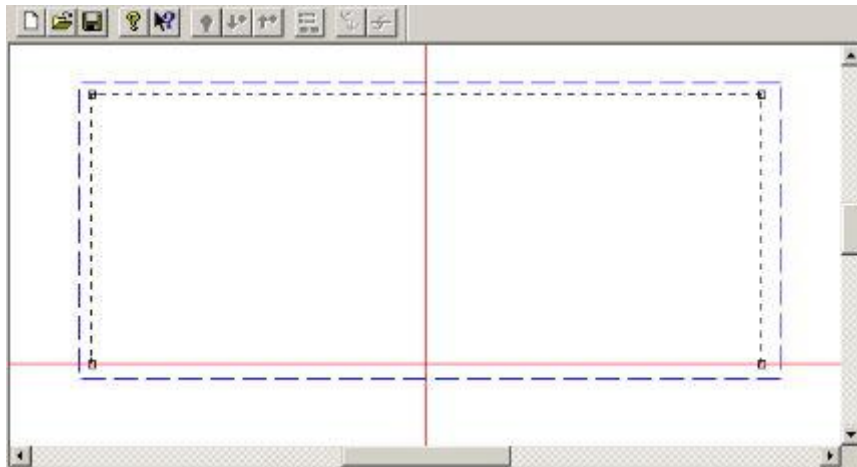
Kommen wir nun zu den beiden „Seitenwänden“. Nun mag der Laie denken, dazu müssen zwei neue Rechtecke gezeichnet werden. Im Prinzip ja. Nicht jedoch wenn man die Kopierfunktion und die Windows Zwischenablage nutzt.

Wir stellen die Ansicht unseres Kondokumentes so ein, daß wir die ganze „Wand“ im Bild haben.


Nun nehmen wir das Werkzeug  „**Markieren**“ und markieren die Wand im Dokument. Wir setzen den Mauszeiger links oben ausserhalb des Rechteckes an, halten die linke Maustaste gedrückt und ziehen ein Markierungsrechteck bis ausserhalb des rechten unteren Vertex. Es müssen alle Teile des Rechtecks markiert sein.




## Verkleinerte Darstellung




Alle Teile innerhalb der blauen gestrichelten Linie sind nun markiert. Markierte Vertexe erkennt man auch daran, daß diese nur als Umriss dargestellt sind.

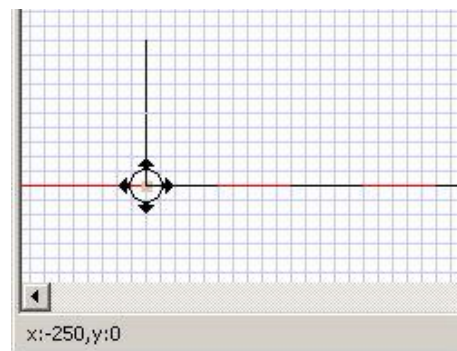
Wir klicken nun auf das Werkzeug  „kopieren“ in der Werkzeugleiste und unsere Wand befindet sich nun in der Zwischenablage. Wir heben nach dem kopieren die Markierung auf, indem wir mit der Maus ausserhalb der Markierung einmal klicken.

Nun öffnen wir eine neue Kondatei. Wir erinnern uns: auf  klicken und im Auswahlfenster „Konstruktion“ wählen. Mit OK ist ein neues Konfenster offen und wir fügen die datei aus der

Zwischenablage mit  „Einfügen“ in unser neues Kondokument ein.

Allerdings ist die Wand nun nicht in der richtigen Position und auch etwas zu breit. Wir vergrössern die Ansicht bis wir ein Raster sehen (Bild auf Taste) und verschieben das Bild entweder mit den Schiebeleisten oder den Pfeiltasten der Tastatur, bis wir den linken unteren Vertex im Bild haben. Das Rechteck sollte dabei immer markiert bleiben.

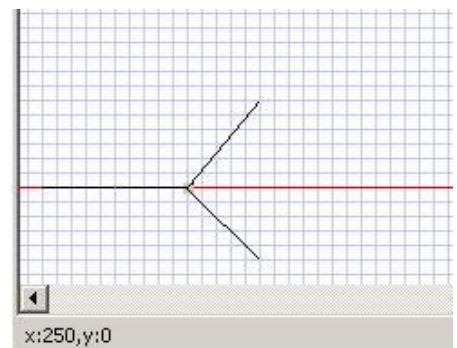
Nun wählen wir das Werkzeug  „Verschieben“ gehen mit dem Mauszeiger auf den Vertex drücken die linke Maustaste, halten dies gedrückt und schieben das markierte Rechteck soweit, daß der Vertex auf die Koordinaten X -250 und Y 0 kommt und lassen dann die Maustaste los. Liegen die betreffenden Koordinaten auf Grund des vergrößerten Bildausschnittes ausserhalb des sichtbaren Bereiches, so können wir den Fensterinhalt mit den Pfeiltasten Schrittweise solange verschieben, bis die Position erreicht ist. Hilfreich beim Verschieben ist auch die Koordinatenangabe am linken unteren Bildrand.



## Hinweis

Das Raster ist innerhalb des markierten Bereiches nur sichtbar während des Verschiebens mittels der Pfeiltasten der Tastatur bei gedrückt gehaltener linker Maustaste.

Die linke Seite der „Wand“ ist nun an der richtigen Position, jedoch ist unsere Wand immer noch zu lang. Dies ändern wir, indem wir die Ansicht mit Hilfe der Pfeiltasten soweit nach links schieben, bis der rechte untere Vertex sichtbar ist. Diesen nehmen wir mit dem Verschiebewerkzeug wieder auf und verschieben, bei gedrückter linker Maustaste, das Bild mittels Pfeiltasten, bis die Position X 250 erreicht ist.





Das Gleiche machen wir mit dem rechten oberen Vertex, wobei wir darauf achten, daß er auf der Y-Koordinate -300 bleibt.

Dies alles mag nun für den Laien etwas umständlich erscheinen, aber es liest sich nur so. Jedenfalls ist diese Methode schneller als die zwei Seitenwände neu zu zeichnen und alle Vertexdaten neu einzugeben. Versuchen sie es mal mit dem anderen Weg, das Polygon für die beiden schmalen Wände neu zu zeichnen.

### Hinweis!


Während des Verschiebens einzelner Vertexe ändert der Mauszeiger sein Aussehen zu einer Pfeilspitze welche jeweils in die Schieberichtung zeigt. Während des Verschiebens sind Linien unsichtbar bzw. behalten scheinbar ihre Position. Erst wenn sie die Maustaste loslassen, werden die Linien wieder an der neuen Position angezeigt.


### Tip


Verschieben sie Vertexe oder markierte Bereiche mittels den Pfeiltasten der Tastatur, so bewegen sie die Maus dabei leicht hin und her, jedoch ohne die linke Maustaste loszulassen. So werden in der Statusleiste immer die aktuellen Koordinaten angezeigt.

Wenn sie die Maus während eines Verschiebevorganges mittels Pfeiltasten ruhig halten, so ist die Aktualisierung der Statusanzeige kurzfristig ausser Kraft und zeigt erst nach loslassen der Maustaste die aktuellen Koordinaten des Mauszeigers an.


Was wir nun noch machen müssen bevor wir die Wand ins 3D-Fenster einfügen, ist die Werte der Vertexe in Z-Achse zu ändern und die Wände etwas dunkler machen, da ja Hauswände auf Grund des Lichteinfalls nie gleich hell sind und so auch der Kontrast bzw. die 3D-Wirkung verstärkt wird. Auch müssen wir noch eine zweite Instanz anlegen, da wir ja auch hier die Wand zweimal brauchen.

Zunächst ändern wir die Z-Werte der Vertexe. Wir erinnern uns: mit  Doppelklicken wir auf einen

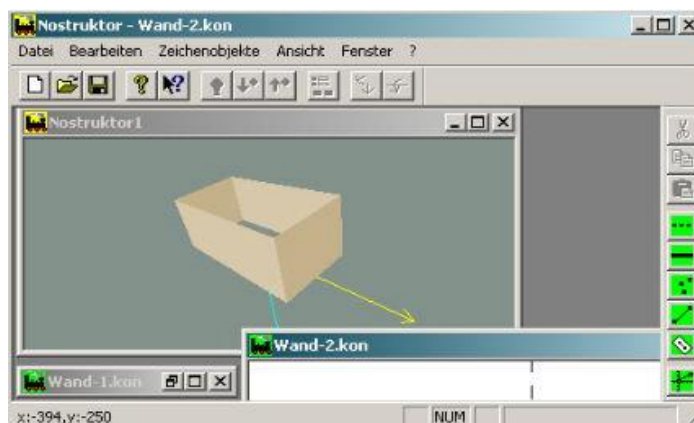
Vertex, ändern die Werte im Koordinatenfeld bei Höhe (Z) in -500 und übertragen den Wert mit  auf alle anderen Vertexe. Wenn wir uns im Menü „**Verteaxeigenschaften**“ befinden, so ändern wir auch gleich die Helligkeit der Farbe. Dabei kann die Farbe der Seitenwände ruhig um einiges dunkler sein als die der Vorder- und Rückwand. Nachdem wir die Höhe übertragen haben klicken wir auf das

Werkzeug  „**Farbe**“ und übertragen auch diese auf alle anderen Vertexe.

Das Thema Farbe wird in einem anderen Kapitel noch näher behandelt.

Nun erstellen wir eine „Kopie“ indem wir wieder über  ins Instanzenmenü gehen, und dort eine neue Instanz anlegen. Bei der für die linke Seite vorgesehenen Wand (Instanz Nr. 0) geben wir im Fenster Instanz-Eigenschaften bei Drehung Y den Wert 90 ein, bei der für die rechte Seite vorgesehenen Wand (Instanz Nr. 1) ebenfalls bei Drehung Y den Wert -90.

Die Rückseiten der Polygone zeigen so nach innen und die Vorderseiten nach aussen, was später noch von Bedeutung sein wird. Nun fügen wir das Dokument „Wand-2“ in das 3D-Fenster ein.



Im 3D-Fenster haben wir nun die vier Aussenwände unseres Hauses.  
Die Kondatei der Seitenwände speichern wir als „**Wand-2**“ ab.

**Hinweis !**

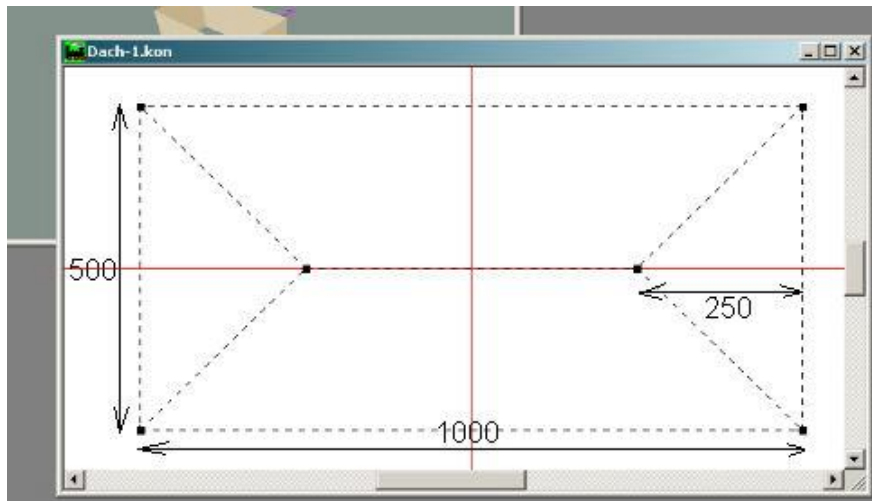
Wer die Positionsvorgaben der einzelnen Kondatei lieber wie im Abschnitt „**Möglichkeit 1**“ Vornehmen möchte, kann dies natürlich auch machen. Dabei bleiben die Werte der Vertexe bei Position (Z) bei 0 und die Positionen der Kondatei im 3D-Fenster werden über das Instanzenmenü eingestellt.

Jetzt soll unser Haus ein Dach bekommen.


Da alle vier Wände gleich hoch sind, also die Seitenwände keinen Giebel besitzen, müssen wir für unser Haus ein vierseitiges Dach machen. Das Dach soll zunächst einmal bündig auf den Wänden aufliegen. Andere Dachformen werde ich später noch ausführlich behandeln.

Wir öffnen eine neue Kondatei und zeichnen ein Rechteck mittig im Dokument mit den Grundmaßen unseres Hauses, also Länge 1000cm (X-Achse) und Breite 500 cm (Y-Achse).

Von den vier Ecken ausgehend zeichnen wir noch zwei Dreiecke wie in der folgenden Abbildung.




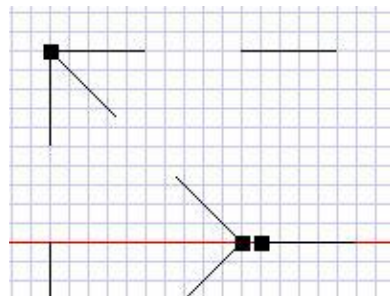
Wichtig!! Beim zeichnen von Linien ist immer darauf zu achten, daß die Linienenden und Anfänge auf der gleichen Koordinate liegen, sich also berühren müssen. Die Linien verbinden wir wieder durch

einen Klick auf  „**Linien zusammenfügen**“ wodurch die Linien mittels Vertex verbunden werden. Nun ziehen wir noch eine Linie zwischen den beiden Vertexen auf der Null-Linie der X-Achse. Auch hier ist darauf zu achten, daß man mit dem zeichnen der Linie genau auf einem Vertex beginnt und auch genau auf einem Vertex endet.

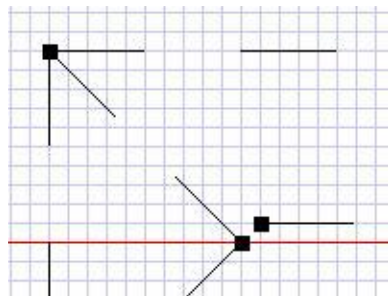
**Tip !**

Um zu überprüfen ob alle Linien richtig zusammengefügt sind, klickt man nach dem zeichnen einer

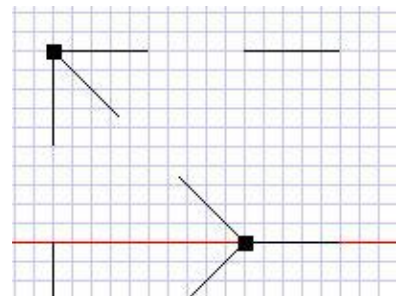
Linie von Vertex zu Vertex noch einmal auf  Befinden sich nun zwei Vertexe dicht nebeneinander, so wurden Linienanfang bzw. Ende nicht korrekt gesetzt. Beispiel:



Falsch




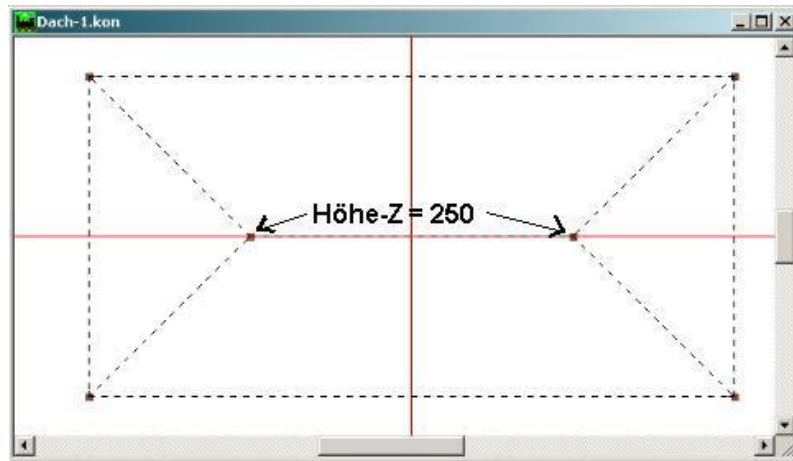
Falsch






Richtig

Nun haben wir zwar das Dach soweit fertig, aber es wäre flach, wenn wir es so belassen würden. Deshalb müssen wir noch die Dachhöhe eingeben. Hier arbeiten wir wieder mit der Tiefe Z.

Wir Doppelklicken mit  „**Vertexeigenschaften**“ auf einen der Vertexe auf der Null-Linie und geben bei **Höhe (Z)** den Wert **-250** ein. Da wir uns schon mal in diesem Menü befinden, stellen wir auch gleich eine zum Dach passende Farbe ein. Ich nehme hier mal ein mittleres Rotbraun.



Mit  übertragen wir den Z-Wert auf den zweiten Vertex auf der Null-Linie und anschliessend die Farbe mit dem Button  die Farbe auf alle übrigen Vertexe. Würden wir das Dach nun in das 3D-Fenster einfügen, würde es auf der Null-Linie hochkant stehen. Deshalb müssen wir im Instanzenmenü noch entsprechende Werte eingeben.

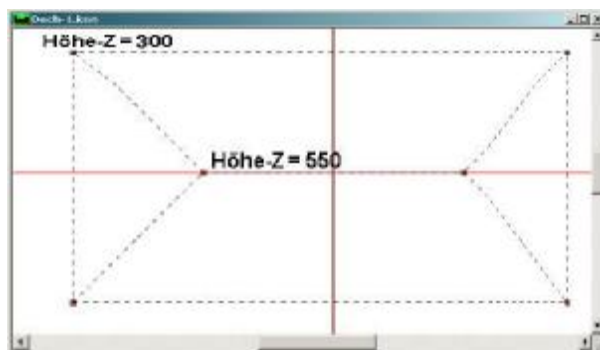
Wir öffnen mit  das Instanzenmenü, klicken auf „**Eigenschaften**“ und geben dort die folgenden Werte ein:



Position:		Drehungen:	
X-Position:	<input type="text" value="0"/>	X-Drehung:	<input type="text" value="-90"/>
Y-Position:	<input type="text" value="-300"/>	Y-Drehung:	<input type="text" value="0"/>
Z-Position:	<input type="text" value="0"/>	Z-Drehung:	<input type="text" value="0"/>

Das Dach soll auf den Hauswänden aufsitzen welche ja 3 m hoch sind, deshalb bei Position Y der Wert **-300**.

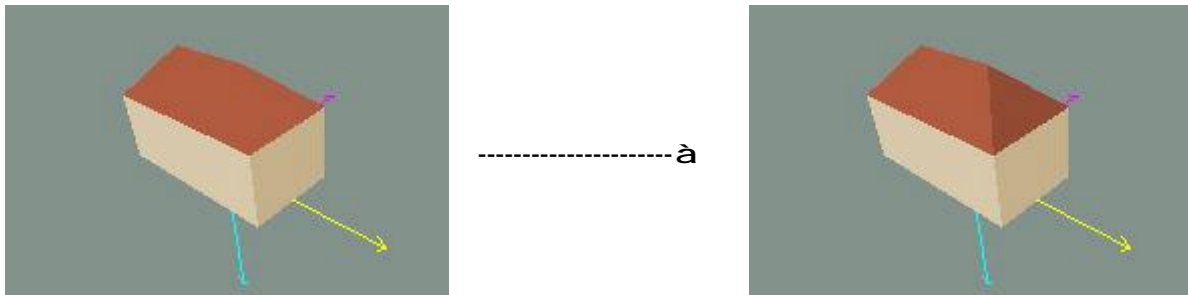
### Tip !

Auch hier können wir uns, wie schon bei den Wänden des Hauses, einige Eingaben im Instanzenmenü sparen.

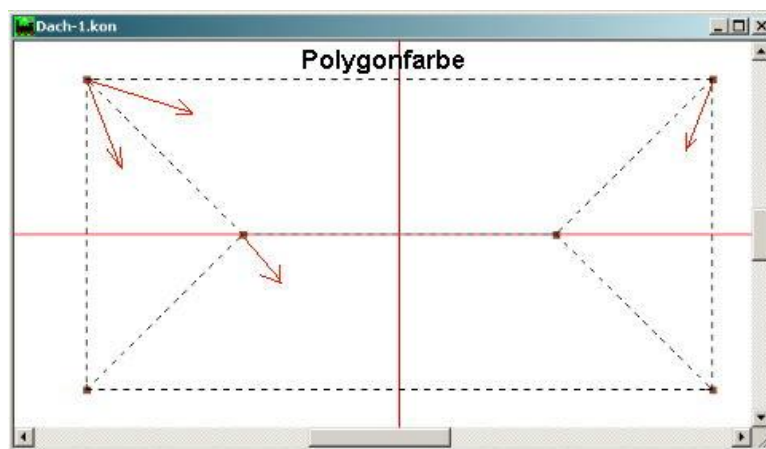


Wir klicken mit  auf einen der äusseren Vertexe (z.B. li oben) und geben bei Höhe-Z den Wert **-300** ein und übertragen diesen mit  auf die anderen äusseren Vertexe. Bei den beiden mittleren Vertexen geben wir den Wert **-550** ein (Wandhöhe 300 plus Dachhöhe 250). Nun muss im Instanzenmenü lediglich bei Drehung X **-90** eingegeben werden. Der Eintrag bei Position Y im Instanzenmenü ist dabei 0.

Nun können wir das Dach in das 3D-Fenster einfügen.



Nun sieht unser Dach allerdings etwas „flach“ aus. Um die 3D-Wirkung hervorzuheben müssen wir die beiden seitlichen Flächen (Dreiecke) ebenso wie die beiden seitlichen Wände, etwas dunkler machen. Wir erinnern uns – für die Farbgebung eines Polygons ist immer der linke obere bzw. bei unregelmässigen Polygonen der oberste Vertex bestimmend. Wollen wir nur die beiden dreieckigen Polygone dunkler machen und die beiden anderen Dachflächen mit gleicher Farbe belassen, so stoßen wir auf ein Problem.





Die roten Pfeile in der Abbildung zeigen welcher Vertex für welches Polygon Farbbestimmend ist. Wir erkennen, daß der linke obere Vertex sowohl für die linke dreieckige, als auch für die hintere Dachfläche die Farbe vestimmt. Wir möchten aber die hintere Dachfläche in gleicher Farbe wie die vordere um so auch aus der hinteren Ansicht den 3D-Effekt zu sehen.

Dies erreichen wir nur dadurch, daß wir dieses Dach mittels zweier Kondateien erstellen, also die Polygone voneinander trennen.


Dazu gehen wir wie folgt vor.

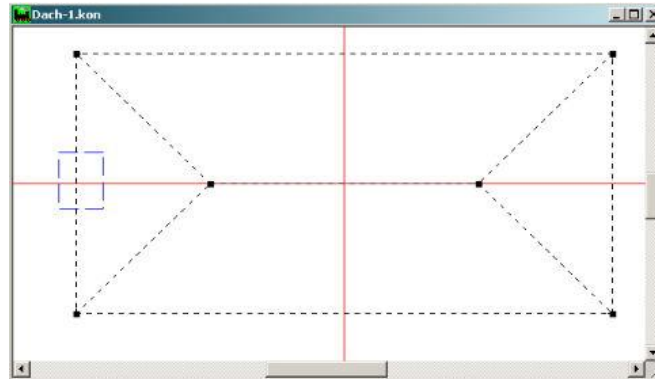
Wir speichern unser dach zunächst unter dem namen „**Dach-1**“ ab. Dann entfernen wir das dach mit


einem Klick auf  aus dem 3D-Fenster. Nun speichern wir die Datei nocheinmal, allerdings unter dem Namen „**Dach-2**“ ab. Dabei verwenden wir nicht den Button  sondern gehen über „Datei“ in der Menüleiste des Hauptfensters „**Speichern unter...**“. (siehe Abb.)



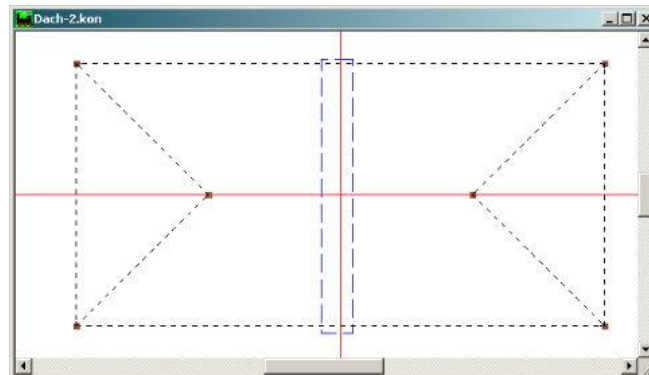
Anschliessend öffnen wir wieder das Dokument „**Dach-1**“ und führen folgende Schritte aus:


Wir markieren die linke senkrechte Linie (etwa in Linienmitte) mit dem Werkzeug  und drücken die „Entf“ Taste der Tastatur. Diesen Vorgang wiederholen wir bei der rechten senkrechten Linie.




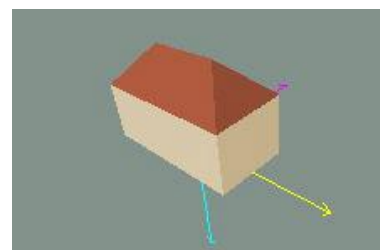
Nun haben wir die vordere und hintere Dachfläche und speichern diese mit einem Klick auf  ab.

Nun aktivieren wir das Dokument „**Dach-2**“, markieren die drei Waagrechten Linien und drücken ebenfalls die „Entf“ Taste.




Anschliessend wählen wir mit  durch Doppelklick einen der Vertexe, gehen im Menü „**Eigenschaften von Vertex**“ in den „**Farbwahldialog**“ und machen die Farbe etwas dunkler. Dies geschieht am besten mit dem kleinen Schieberegler rechts am Rande der Windows Farbpalette.

Auch dieses Dokument können wir nun mit  abspeichern. Wir fügen nun beide Kondateien in unser 3D-Fenster ein und haben ein Dach dessen vordere und hintere lange Dachflächen etwas heller sind als die beiden Seitenflächen. So wird die 3D-Wirkung erst ins „rechte Licht“ gesetzt.



Fassen wir zusammen: wir haben 2 mal 2 Hauswände und ein Dach für das wir insgesamt 4 Kondokumente angelegt haben. Aber was ist ein Haus ohne Fenster und Türen? Beschäftigen wir uns also mit diesen.

#### Hinweis !

Um Linien zu löschen müssen diese in der Mitte oder an einem der Enden markiert werden. Löschen ist auch mit  möglich, jedoch ist es etwas schwierig genau die Mitte der Linie zu treffen. Ich lösche deshalb lieber mit Markieren.

Bevor wir jedoch damit beginnen noch einige Worte zur Übersichtlichkeit während unserer Arbeiten und wie ich gewohnt bin zu arbeiten. Dazu einige Beispiele und Tips.

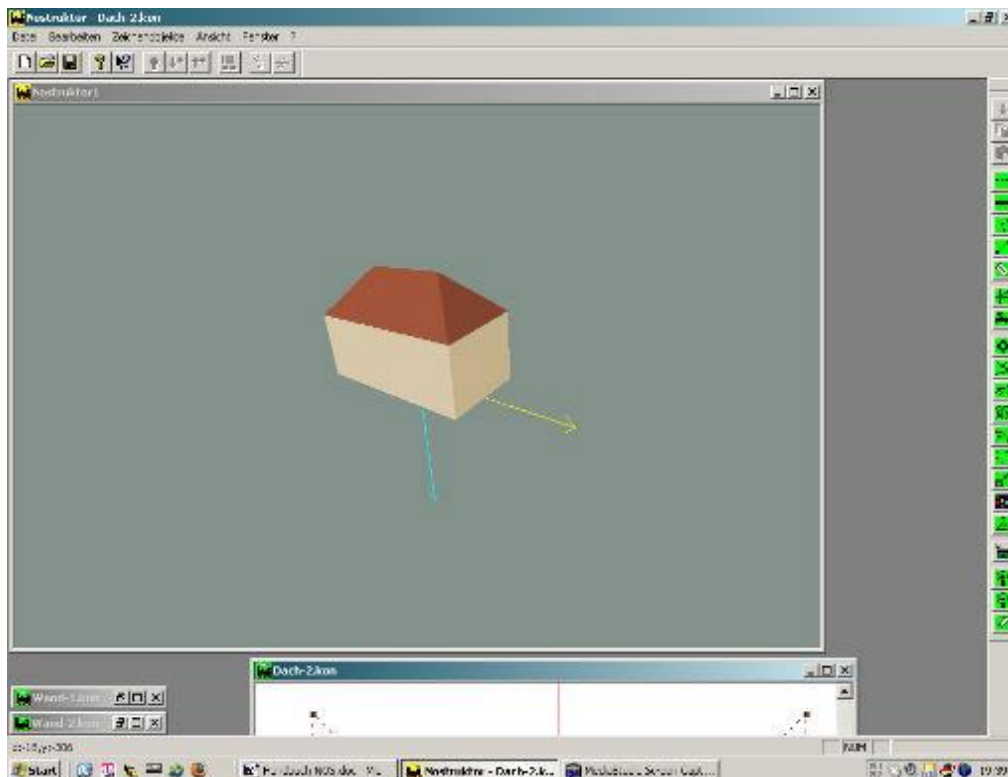



## Hinweis!

Wie wir bereits wissen können wir aus dem 3D-Fenster keine Kondateien entfernen bzw. austauschen, wenn zwischenzeitlich die betreffende Kon-Datei geschlossen und erneut geöffnet wurde. Ebenso ist es nicht möglich aus einem 3D-Fenster Kon-Dateien zu entfernen oder nachträglich zu bearbeiten. Aber – man kann in eine gespeicherte, zuvor geschlossene Mod-Datei neue Kon-Dateien einfügen.

Deshalb ist es Sinnvoll z.B. Kondateien nicht zu schliessen bevor man das Modell fertig hat. Natürlich ist es bei grösseren Modellen, für welche sehr viele einzelne Kondateien benötigt werden, diese an einem Tag fertig zu stellen. Man sollte sich also wohl überlegen, wenn man die Arbeit für längere Zeit unterbricht oder am nächsten Tag fortsetzen möchte, in welchem Zusatznd man die Mod-Datei speichert. Sind noch nicht allzu viele Kondateien in der Mod verbaut, so ist der Aufwand, von vorne mit dem Einfügen zu beginnen, noch nicht allzu gross. Bei umfangreicheren Modellen lohnt es sich jedoch, die Mod zu speichern und am nächsten Tag, oder später, dort weiter zu machen.

Damit ich immer alles im Blick und eine Kontrolle darüber habe welches Fenster gerade aktiv ist, platziere ich die einzelnen Arbeitsfenster schon zu Beginn meiner Arbeit so, daß ich immer die Übersicht behalte.



Das offene Nostruktur-Fenster (3D-Fenster) ziehe ich soweit auf, daß unten unten rechts etwas Platz bleibt. Die Werkzeugleiste des Konstruktionsfensters setze ich an den rechten Fensterrand. Man kann diese aber auch frei beweglich an jede beliebige Stelle setzen. (siehe Seite 9 dieser Anleitung) Konstruktionsfenster welche bereits fertig sind oder solche die ich evtl. erst später noch einmal benötige, minimiere ich. Diese werden automatisch an den unteren Rand des hauptfensters gesetzt, immer von links nach rechts und von unten nach oben. Lediglich das Nos-Fenster welches ich gerade bearbeite, lasse ich geöffnet, schiebe es aber an den unteren Bildrand. Wenn ich nun in diesem Fenster arbeiten möchte, maximiere ich es mit einem Doppelklick auf die Kopfleiste. Möchte ich es wieder verkleinern, so geschieht das mit dem Button  und es ist wieder in seiner ursprünglichen Grösse und Position. So habe ich immer freien Blick auf mein Modell im 3D-Fenster und kann durch Mausklick (auch bei minimierten Fenstern) hin und her schalten. Übrigens – ein Kon-Fenster muss nicht vollständig geöffnet sein um den Inhalt in das 3D-Fenster einzufügen. Dies geht auch mit minimierten Fenstern (im Bild oben in der li. u. Ecke). Man muss dazu nur das entsprechende Fenster mit der Maus einmal anklicken um es zu aktivieren. Ob ein Fenster, ob offen oder minimiert, gerade aktiv erkennt man an der andersfarbigen Kopfleiste.

Diese Farbe ist jedoch von User zu User unterschiedlich und hängt von der individuellen Einstellung der Windows-Oberfläche jedes einzelnen ab.



Selbstverständlich bleibt es jedem einzelnen überlassen wie er mit dem NOS arbeitet. Vielleicht findet der ein oder andere ja auch eine Vorgehensweise die ihm mehr zusagt.

Jetzt wollen wir uns um die Fenster und Türen unseres Hauses kümmern. Hier gibt es natürlich die unterschiedlichen Möglichkeiten, wobei wir uns in dieser Anleitung zunächst mal mit der althergebrachten befassen, diese als separate Polygone auf die Wand „aufzusetzen“.

In EEP ist es (fast) wie im richtigen Leben – die Proportionen sollten schon in etwa stimmen. Das trifft auch auf den Hausbau in EEP zu. Die Stockwerkhöhe eines einfachen Wohnhauses beträgt in der Regel ca. 250 bis 280 cm, eine Haustüre hat in der Regel eine Höhe von 200 cm und die Breite kann zwischen 100 und 120 cm variieren. Die Fenster beginnen meist bei einer Höhe zw. 80 und 100 cm und enden meist so um 30 bis 40 cm unterhalb der Zimmerdecke.

Wenn man diese Maße kennt, dürfte es nicht schwer sein, seinem Haus einigermaßen realistische Proportionen zu geben.

Wir wollen hier keine Luxusvilla oder ein Haus mit Grossraumbüros bauen, so daß wir uns an diesen Maßen orientieren können.

### Tip !

Um Fenster und Türen nicht ins Leere zu bauen und um mir lästiges Hin und Herschieben im 3D-Fenster zu ersparen, missbrauche ich die Kondateien als Schablonen. Das geht so:

Ich öffne die Datei „Wand-1“ und speichere sie unter dem Namen „Fenster-1“ ab.

**Vorsicht !** die Kon-Datei „Wand-1“ vorher aus dem 3D-Fenster nehmen und nach dem speichern der neuen Datei wieder öffnen und erneut einfügen.

Nun kann ich in das Polygon meine Fenster und Türen zeichnen, diese Positionieren und anschliessend die Linien und Vertexe welche die Wand darstellen löschen.


Wir zeichnen also innerhalb des Wandpolygons unsere Türen und Fenster.

Die Türe mache ich 220 cm Hoch und 140 cm breit. (ich rechne die Breite eines Türrahmens hier mit je 10cm zur Türhöhedazu).



Für die Fenster nehme ich ein Maß von 120 cm Höhe und 80 cm Breite.

Die Türe setze ich mit der Unterkante auf die Null-Linie, die Fenster mit der Unterkante auf Y –100cm

Da die Fenster ja „vor“ der Wand liegen müssen um sie zu sehen, müssen wir noch die Höhe Z in den Vertexen eingeben, wobei diese ca. 4 – 6 cm vor der Wand stehen sollten, um einen unerwünschten Flimmereffekt zu vermeiden. Dieser Flimmereffekt tritt auf, wenn zwei Polygone direkt „aufeinander“ liegen. Die Farbe ist zunächst unerheblich, sie sollte sich nur von der Hauswand abheben. Ich nehme dazu das dunkle Grau aus der Windows Farbpalette.


Wir erinnern uns: Mit  auf einen Vertex doppelklicken, bei Position Höhe (Z) –254 eingeben, im Farbwahl-Dialog die Farbe wählen und jeweils mit Okay bestätigen.

Anschliessend übertragen wir die Werte auf die anderen Vertexe sowohl die des Fensters als auch

der Türe.  für Höhe Z  für die Farbe.

Das Fenster brauchen wir mehrfach, müssen dies aber nicht mehrfach zeichnen.

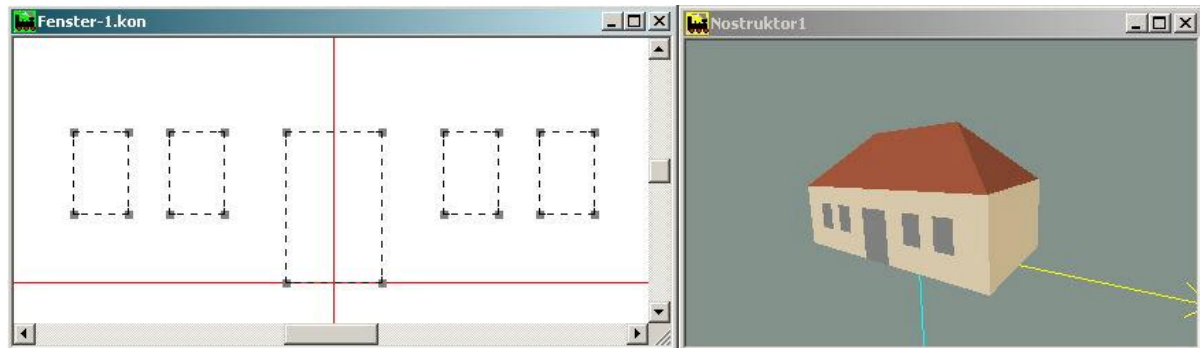
Wir markieren das Fenster mit  kopieren es mit  in die Zwischenablage und fügen die Kopie

anschliessend mit  in das Dokument ein. Dabei ist es unerheblich wenn die markierte Kopie über einem anderen Polygon zu liegen kommt, solange es markiert bleibt. Das markierte Polygon kann ich nun in Position schieben und anschliessend alle Linien und Vertexe der Wand löschen.

Dazu markieren wir die Vertexe des Wandpolygons und löschen mit der „Entf“ – Taste. Die Linien werden dabei automatisch mitgelöscht.


Wenn sie alles richtig gemacht haben müsste das Ganze jetzt so aussehen. Linkes Bild die Kondatei mit den Fenstern und der Türe (wobei Anzahl und Position der Fenster ihnen überlassen bleibt) und im Rechten Bild wie das Ganze im 3D-Fenster aussehen müsste.

Wenn sie nun die Ansicht im 3D-Fenster drehen, werden sie sehen, daß Fenster und Türe auch auf der Rückseite vorhanden sind. Warum? Wir haben die Kondatei „Wand-1“ als Schablone benutzt und somit auch alle Instanzen und Einstellungen übernommen.




Nun wird der eine oder andere sagen was soll die Türe an der Rückseite? Ich will da keine Türe. Dem kann geholfen werden. Wir gehen genauso vor wie bereits beim Dach.

Wir nehmen die Datei „**Fenster-1**“ aus dem 3D-Fenster und speichern sie nochmal mit neuem Namen ab. Für den Namen wählen wir „**Fenster-2**“

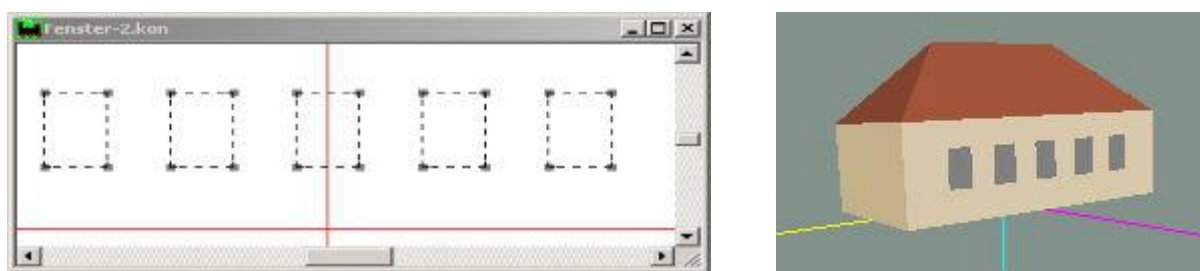
Danach öffnen wir die Datei „**Fenster-1**“ erneut und löschen die zweite Instanz (Instanz Nr. 1), welche ja die Rückseite darstellt. Wir können diese Datei nun mit  speichern.

Nun löschen wir die erste Instanz (Instanz Nr. 0) im Dokument „**Fenster-2**“ (die Fenster für die Vorderseite sind ja bereits in „**Fenster-1**“ enthalten.) und löschen auch die Türe.

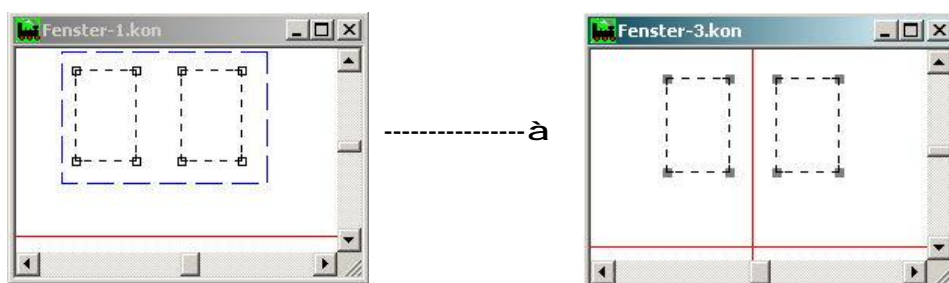
Dann fügen wir das Fenster aus der Zwischenablage in das Dokument, positionieren es an die Stelle an der sich vorher die Türe befand und korrigieren die Abstände. Nun noch mit  speichern.

Sollte sich die Kopie nicht mehr in der Zwischenablage befinden, so kopieren wir einfach eines der vorhandenen Fenster erneut.

Die fertige Kondatei „**Fenster-2**“ sieht dann so aus.





Jetzt brauchen wir nur noch Fenster für die Seitenwände. Dazu kopieren wir zwei Fenster aus einer der Kondateien und fügen diese in ein neues Kon-Dokument.



Zwei Fenster markieren und kopieren, in ein neues Dokument einfügen und als „**Fenster-3**“ speichern

Nachdem wir die Fenster ausgerichtet haben, müssen wir noch die Höhe Z korrigieren, da der Abstand der beiden Seitenwände grösser ist als der Abstand der vorderen zur hinteren Wand.

Wir klicken mit  auf einen der Vertexe, ändern den Wert bei Höhe Z auf -502 (halber Abstand der Wände plus 2cm für den Abstand der Fenster zur Wand.)

und übertragen diesen Wert mit  auf die übrigen Vertexe. (Die Werte für X und Y bleiben unberührt)

Koordinaten:	
Horizontale (X):	-110
Vertikale (Y):	-220
Höhe (Z):	-502

Anschliessend erstellen wir eine zweite Instanz, wir wollen ja Fenster auf beiden Seiten.

In der ersten Instanz geben wir bei Drehung Y den Wert 90 ein, in der zweiten Instanz (Nr 1) bei Drehung-Y den Wert -90. Nun fügen wir die Fenster in das 3D-Fenster ein.



Nun kann man bereits erkennen daß es sich bei unserem Modell um ein Haus handelt. Es ist zwar bei Weitem noch nicht perfekt, es fehlen noch viele Details wie Texturen, Dachrinne, Kamin usw. aber Dazu kommen wir später.

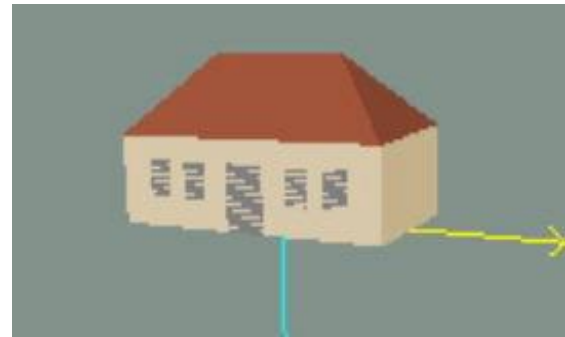
Zunächst einmal möchte ich sie noch mit einigen Einstellungsmöglichkeiten für die Ansicht vertraut machen. Wenn sie die Ansicht im 3D-Fenster verkleinern, also wegzoomen (siehe auch Seite 10), werden sie bemerken, daß die Fenster mit zunehmendem Abstand zu „flackern“ anfangen und letztlich auch ganz mit den Wänden „verschmelzen“.

Dies liese sich dadurch vermeiden, indem man den Abstand der Fenster zu den Wänden vergrößert. Das würde aber wiederum aus der Nähe nicht so gut aussehen, besonders wenn man das Haus seitlich betrachtet.

Deshalb gibt es die Funktion Z-Offset.


Z-Offset zwingt die Grafikengine die damit belegten Polygone als letzte zu zeichnen. Somit sind diese immer vor einem anderen Polygon sichtbar, selbst wenn sie auf gleicher Ebene liegen.

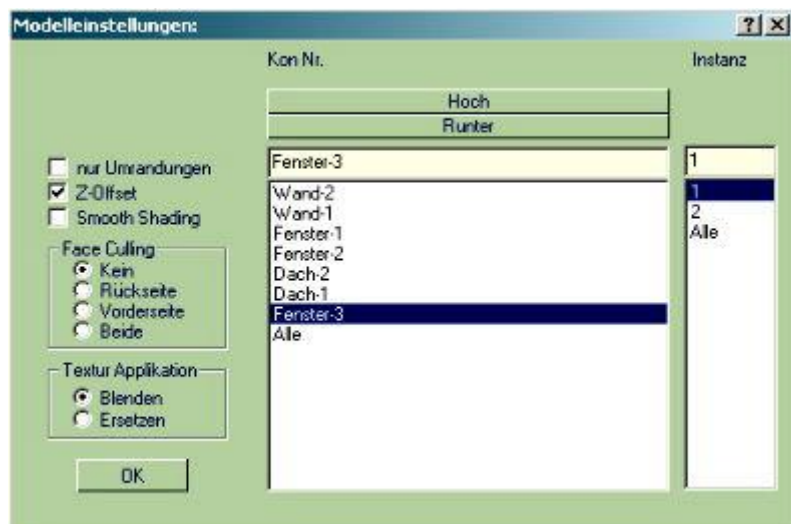
Sicherheitshalber sollten aber solche Polygone trotzdem einen geringen Abstand zum dahinterliegenden Polygon erhalten. In unserem Falle haben wir ja einen Abstand von 2 cm.



Und so stellen wir Z-Offset ein:

Wenn unser 3D-Fenster aktiv ist,

gehen wir mit  in das Menü „**Modelleinstellungen**“  
Im grösseren der beiden Fenster sind alle im 3D-Fenster enthaltenen Kondokumente aufgelistet. Im kleineren, rechts werden die Instanzen des jeweils markierten Kondokumentes aufgelistet, sofern mehr als eine Instanz enthalten ist.



Wir markieren nun nacheinander im grossen Fenster die Kondateien „**Fenster-1**“ und „**Fenster-2**“ Und machen jeweils bei Z-Offset (links neben dem grossen Fenster) ein Häkchen. Bei der Kondatei „**Fenster-3**“ müssen wir zusätzlich im rechten Fenster nacheinander die beiden Instanzen 1 und 2 markieren und jeweils bei Z-Offset ein Häkchen machen.

#### Hinweis !

Sind mehrere Instanzen vorhanden und sollen diese alle die gleichen Einstellungen erhalten, so genügt es, im kleinen Fenster „**Alle**“ zu markieren und bei Z-Offset ein Häkchen zu machen. Hier ist jedoch zu beachten, daß im Feld „**Textur Applikation**“ (links unten) wieder auf „**Blenden**“ gesetzt wird. Die Bedeutung „**Blenden**“ und „**Ersetzen**“ wird später näher erläutert.


#### Achtung!




Wird eine gespeicherte Mod2 Datei geschlossen und erneut geöffnet, so sind an Stelle der Dateinamen im grossen Fenster lediglich Nummern. Deshalb sollten Mod2 Dateien erst geschlossen werden, wenn alle Einstellungen vorgenommen wurden. Ansonsten ist es bei vielen Kondateien mit sehr viel Aufwand verbunden die entsprechenden herauszufinden um Einstellungen vorzunehmen.

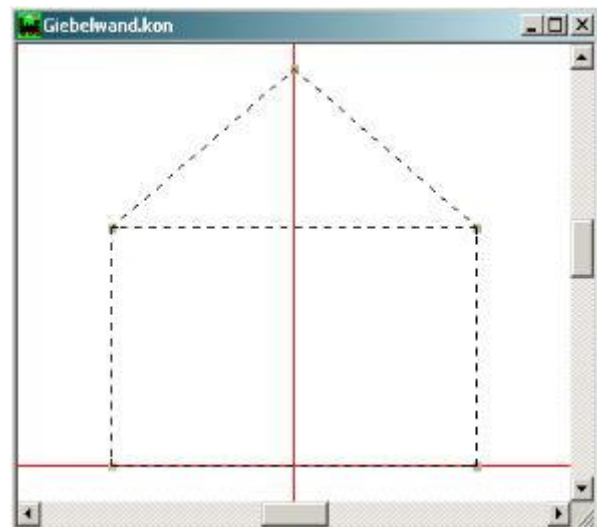
Ist bei allen Fensterdateien und deren Instanzen Z-Offset eingestellt, so bestätigen wir mit **OK** Nun sind die Fenster auch aus grösserer Entfernung zu erkennen, da wir ja mit Z-Offset das Programm zwingen, diese Polygone immer als letzte zu zeichnen.

Bevor wir nun unserem Modell Texturen verleihen und für EEP fertigstellen, möchte ich noch eine andere Dachvariante zeigen.

Dazu nehmen wir die Datei „**Wand-2**“ und speichern diese unter einem neuen Namen ab. Ich nehme hier den Namen „**Giebelwand**“ Nun zeichnen wir, ausgehend von den oberen beiden Vertexen, ein Dreieck, welches mit seiner Spitze bei Y -550 und X 0 liegt. Wenn die

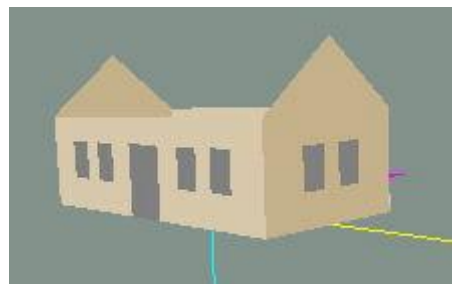
beiden Linien mit einem Vertex ( mit  ) verbunden sind, übertragen wir noch die Werte für Höhe Z und die Farben von einem der vorhanden Vertexe.

Mit  auf einen Vertex doppelklicken und mit den entsprechenden Werkzeugen  und  die Werte für Höhe Z und Farbe übertragen.




Weitere Einstellungen brauchen wir hier nicht mehr vorzunehmen, da wir ja ein vorhandenes Dokument, welches bereits alle Einstellungen enthielt, nur umbenannt und Inhaltlich verändert haben.

Bevor wir die neue Kondatei in unser 3D-Fenster einfügen, entfernen wir zunächst das Dach „**Dach-1**“ und „**Dach-2**“ und tauschen nun die alten Seitenwände durch die neuerstellten aus. Diesmal setzen wir das Dach nicht auf den Hauswänden bündig auf, sondern wollen dies etwas realistischer machen und lassen das Dach an allen Seiten etwas überstehen.





Ich möchte dabei einmal meine Vorgehensweise zeigen, wie man am einfachsten die entsprechenden Maße und Werte ermittelt. Natürlich bleibt es jedem selbst überlassen ob er sich die gleiche Arbeitsweise aneignet oder evtl. eine einfachere Möglichkeit findet.



Ich maximiere zunächst das Fenster des Dokumentes „Giebelwand“ und nehme die Hilfslinie,  im Programm als „**Einzellinie**“ bezeichnet.

Damit ziehe ich eine Linie von der Spitze des Giebels über den rechten oberen Vertex der Wand hinaus. Dabei sollte die „Hilfslinie“ mit der gestrichelten Linie des Giebels Deckungsgleich sein. Dazu vergrößere ich die Ansicht bis das erste, bzw. im NOSPlus das zweite, Raster sichtbar ist. Mit den Pfeiltasten der Tastatur verschiebe ich die Ansicht bis der obere Vertex der Giebelspitze sichtbar ist und setze den Mauszeiger auf den Vertex, halte die linke Maustaste gedrückt und verschiebe die Ansicht mittels der Pfeiltasten bis der rechte obere Vertex der Wand im Bild ist.

Etwas unter- und rechts des Vertex lasse ich die Maustaste los und setze mit  einen Vertex.

Nun kann ich mit  den Vertex der Hilfslinie aufnehmen und solange verschieben, bis ich den gewünschten Dachüberstand (Abstand zur Wand) erreicht habe. Dabei kommt es weniger auf den exakten Abstand als auf die Deckungsgleichheit der beiden Linien.

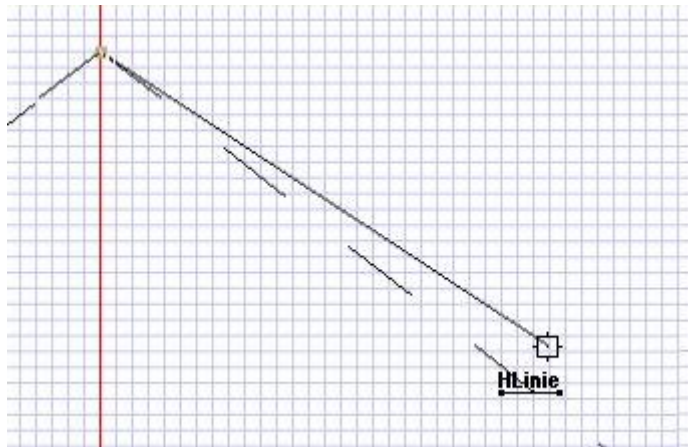


Bild 1  
Mauszeiger am oberen Vertex  
(Giebelspitze) ansetzen.....

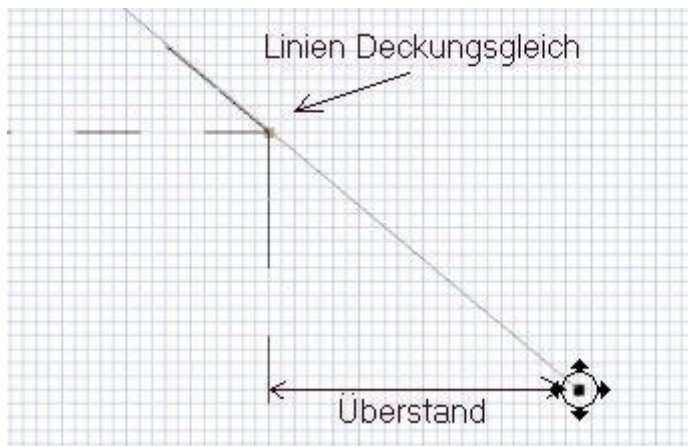



Bild 2  
..... die Ansicht bei gedrückt gehaltener Maustaste mit den Pfeiltasten verschieben bis zum rechten oberen Vertex des Wandpolygons, Maustaste loslassen, am Linienende Vertex setzen und Vertex verschieben bis dieser an der Position X. 280 und Y -270 ist. Der Überstand beträgt hier 30 cm. Größere oder kleinere Überstände ergeben natürlich auch andere X und Y Werte.

Zur Orientierung wann ich den von mir gewünschten Überstand erreicht habe, dient mir die Koordinatenangabe in der Statusleiste (unten links). Dabei ist, wie bereits erwähnt, beim Abstand unter Umständen ein Kompromiss zu machen, da es mehr auf genaue Deckung der Linien ankommt.

Nun kann ich die Werte für mein eigentliches Dach einfach ablesen und notieren.

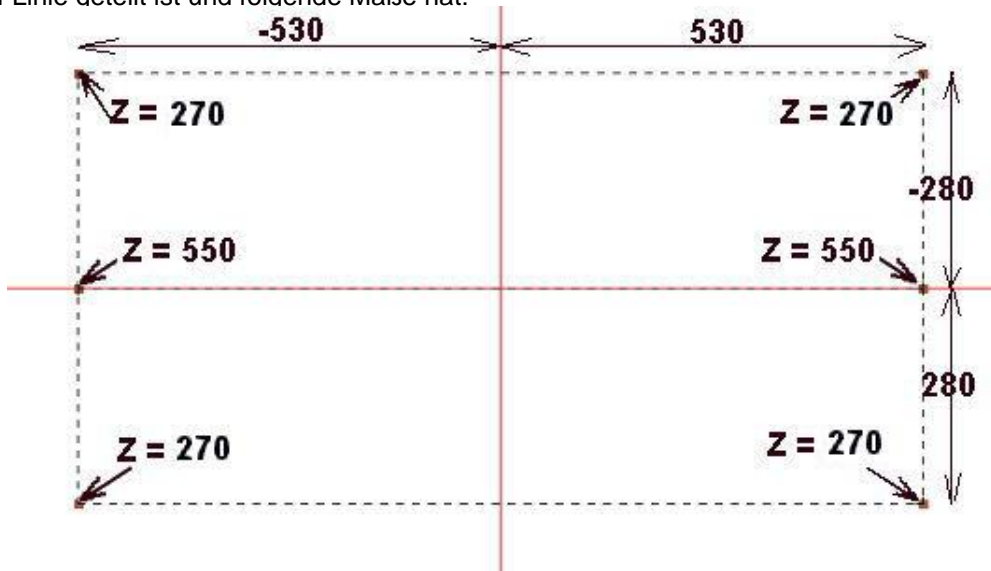
Dazu führe ich den Mauszeiger genau auf den Vertex und notiere mir die X und Y Werte in der Statusleiste. In diesem Falle X 280 und Y -270.

Der X-Wert stellt dabei den Überstand der Dachunterkante dar, der Y-Wert die Höhe der Dachkante. Ausserdem notiere ich mir noch den Wert der Giebelspitze. Haben wir uns alle Werte notiert, können

wir die Hilfslinie entfernen. Mit  den Vertex am unteren Ende der Hilfslinie markieren, und mit der „Entf“ Taste löschen. Die Linie wird mit gelöscht.

Diese Werte übertragen wir später in die Vertexe unseres Daches. Hierzu öffnen wir eine neues Kondokument.

Ich zeichne also im neuen Dokument zwei zusammenhängende Polygone, wobei die mittlere Linie, welche die beiden Dachhälften trennt entlang der X-Achse in Höhe 0 verlaufen muss. Auch an den Giebelseiten lasse ich das Dach um 30 cm überstehen. Somit muss ich, genau mittig des Dokumentes, ein Polygon zeichnen, welches in der Mitte, entlang der Null-Linie geteilt ist und folgende Maße hat:



Im Menü „Verteigenschaften“ setze ich noch die Werte für Höhe (Z) wie in der Zeichnung oben angegeben. Nun noch im Menü „Instanzen“ bei Drehung (X) -90 eingeben und wir können das Dach in das 3D-Fenster einsetzen.

### TIP !

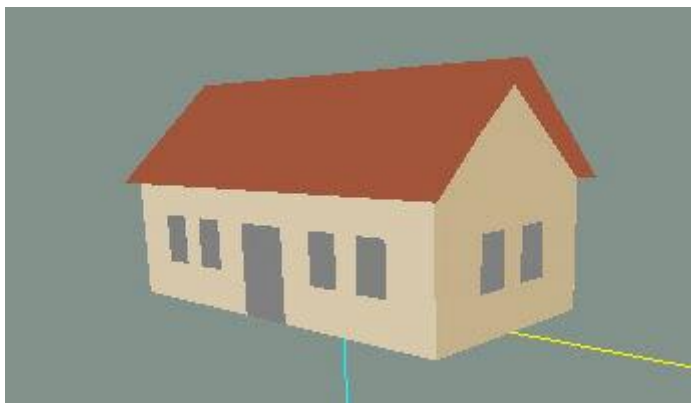
Wer sich das vorhergehende erst mal durchgelesen hat und noch nicht angefangen hat das Dach zu zeichnen, kann es sich etwas leichter machen.

Da ich, wie man bereits feststellen konnte, immer den bequemen Weg gehe, habe ich das neue Dach natürlich nicht neu gezeichnet, sondern die vorhandene Datei „Dach-1“ einfach erneut unter dem

Namen „Dach-3“ abgespeichert und lediglich die Vertexe an die neue Position verschoben **J**

Somit habe ich mir das Neuzeichnen erspart und musste nur die Werte für Höhe (Z) der Vertexe ändern. Auch den Wert für Drehung im Menü „Instanzen“ habe ich bereits.

Das Ganze sieht dann im 3D-Fenster so aus:



Wir haben nun bereits, mit mehr oder weniger Aufwand, zwei Häuser mit unterschiedlichen Dachformen. Gut – die sehen etwas bescheiden aus – so ganz ohne Texturen und Details. Deshalb werden wir uns jetzt dem Kapitel 2 „arbeiten mit Texturen“ zuwenden.

Wer jedoch sein Modell erst mal im „Rohbau“ in EEP betrachten möchte, kann gleich zu Kapitel 4 wechseln und die Kap. 2 und 3 später nachholen.

## Kapitel 2

### Arbeiten mit Texturen

Texturen bewirken zweierlei; zum einen verleihen sie unseren Modellen einen Hauch von Realität, zum anderen müssen wir beim Einsatz von Texturen nicht jedes einzelne Detail als Polygon zeichnen.

Was sind Texturen. Texturen sind nichts anderes als Bilddateien die je nach Bildgrösse eine unterschiedliche Zahl von Bildpunkten, Pixel genannt, aufweisen.

Texturen für EEP-Modelle müssen allerdings ein ganz bestimmtes Datei-Format sowie vorgegebene Grössen bzw. Seitenverhältnisse haben.

EEP z.B. kann nur das \*.DDS Format lesen, da es unter DirectX läuft. Der Nostruktur hingegen läuft unter OpenGL und kann nur das \*.bmp Format lesen. Daraus folgt, daß wir zum Konstruieren eine BMP-Dateien brauchen.

Wie man Texturen erstellt und in das jeweils richtige Format bringt, werden wir in einem anderen Abschnitt erfahren. Zunächst gilt es, unser Haus mit bereits vorhandenen Texturen aufzuwerten.

Mit der Grundversion von EEP bzw. dem Nostruktur werden etliche Texturen gleich als BMP mitgeliefert. Es gibt auch Teilweise die Möglichkeit Texturen anderer Kollegen zu verwenden. Meist genügt eine kurze Anfrage per Mail bei dem betreffenden.

Auf unserer Website: <http://www.vmb-mroth.de/Sonderseite.htm> kann eine Liste von Texturen geladen werden, die frei verwendet werden dürfen.

Wenden wir uns nun wieder unserem kleinen Haus zu.

Zunächst werden wir uns darüber klar, welche „Struktur“ unsere Hauswand bekommen soll.

Wir können hier bereits auf eine Vielzahl fertiger, frei verfügbarer Texturen zugreifen.

Da es wenig Sinn machen würde, wenn wir in einer Textur, z.B. Ziegelsteine, eine komplette Hauswand abbilden würden, verwenden wir nur einen kleinen Ausschnitt und wiederholen diese Textur mehrfach auf unserem Polygon.

Wie geht das.

Als Beispiel nehmen wir die farblich neutrale Ziegelsteintextur aus der Grundausstattung von EEP. Farblich neutrale Texturen haben den Vorteil, daß man beliebig viele verschiedenfarbige Modelle damit erstellen kann, ohne jedesmal eine neue Textur-ID dafür zu „verschwenden“

Der Dateiname dieser Textur ist „64x64y2“, wir arbeiten jedoch mit der ID-Nummer, diese ist hier **85**.



Abb. in Originalgrösse

#### Hinweis!

Es hat sich schon seit längerem bei den meisten Konstrukteuren durchgesetzt, für die Dateinamen gleich die ID-Nummer zu verwenden.

Dies erleichtert die Suche nach bestimmten Texturen. Lediglich ältere Texturen, teilweise noch Standart-Texturen aus EEEC-Zeiten haben einen anderen Dateinamen als ihre ID-Nummer.

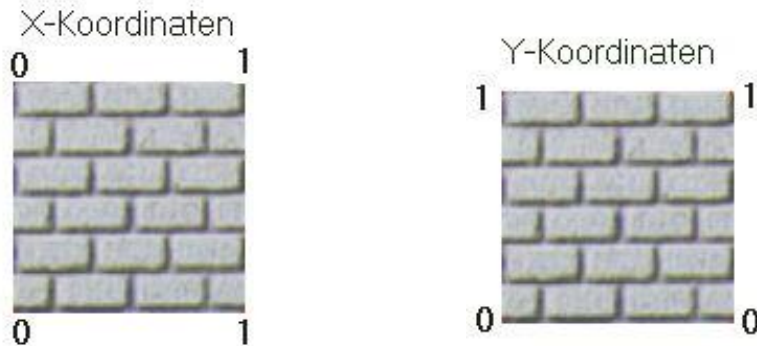
Gehen wir davon aus, daß in der Realität ein Ziegelstein ca. 10x20 cm gross ist, so stellt die Textur einen Ausschnitt von ca. 60x60cm dar. (Natürlich stimmt das nicht ganz genau auf den cm, aber in EEP kann man durchaus etwas schummeln J )

Folglich müssen wir diese Textur auf unseren Hauswänden entsprechend oft wiederholen.


Dazu müssen wir in jedem einzelnen Vertex die ID sowie die Koordinaten der betreffenden Textur eingeben und zwar in jedem Vertex die entsprechende X und Y Koordinate.

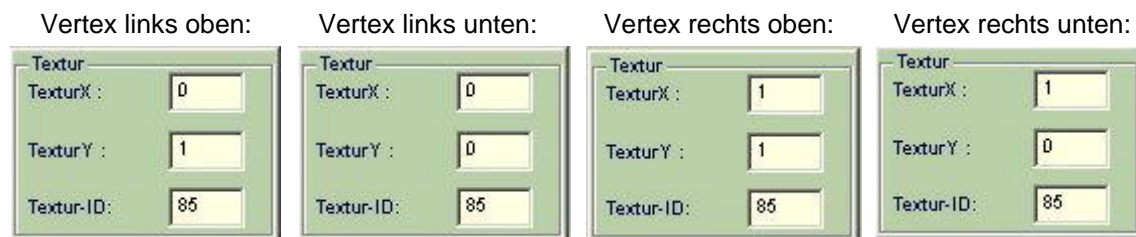
Eine Textur hat X und Y Koordinaten. Die X-Koordinaten beginnen immer links mit 0 und enden rechts mit 1 und die Y Koordinaten beginnen unten mit 0 und enden oben mit 1, egal wie gross eine Textur ist.

Beispiel: (Textur vergrössert)

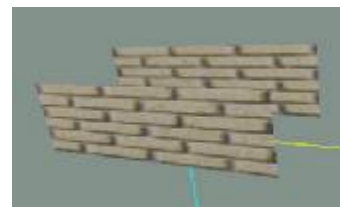


Wir müssten also, um diese Textur einmal auf die Ganze Fläche unseres Polygons zu „kleben“ diese Werte in die einzelnen Vertexe unseres Wandpolygons übernehmen.

Öffnen wir also zunächst die Kondatei „Wand-1“ unseres Hauses und doppelklicken wir mit  nacheinander auf jeden einzelnen Vertex, um in das Eigenschaftenmenü zu gelangen. Hier tragen wir im Feld „Textur“ zunächst die Textur-ID im unteren Fenster ein und anschliessend die entsprechenden Texturkoordinaten:



Mit diesen Werten haben wir die Textur einmal auf die gesamte Fläche gelegt was dann so aussieht:

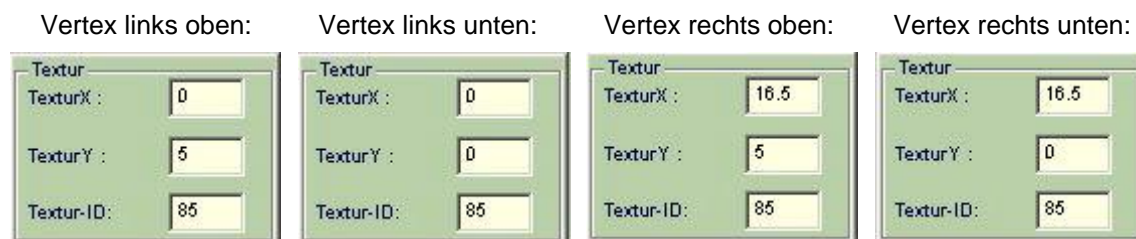


#### Hinweis !

Der NOS akzeptiert kein Komma! Müssen Bruchstellen eingegeben werden, so sind diese mit einem Punkt zu trennen

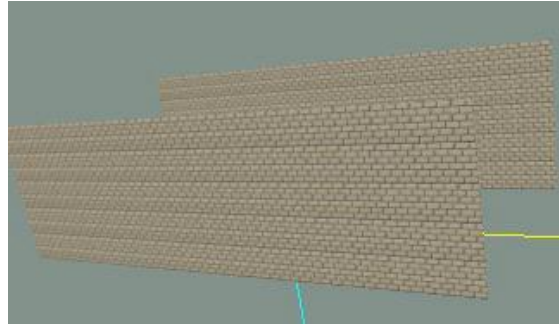
Die Proportionen der Ziegelsteine stimmen hier nicht. Wir müssen die Textur also sowohl in X-Richtung als auch in Y-Richtung vervielfachen um diese korrekt darzustellen. Wir erinnern uns; die Textur stellt einen Ausschnitt in der Realität von ca. 60 x 60 cm dar, unsere Hauswand ist X 10 m und Y 3 m. Wir teilen also jeweils die Höhe und Länge unserer Wand um den Faktor 0.6 und bekommen so für X ( $1000 : 60 = 16.666...$ ) abgerundet den Wert 16.5 und für Y ( $300 : 60 = 5$ ) den Wert 5.

Da wir in der Regel Texturwiederholungen von links nach rechts und von unten nach oben vornehmen, (anders müssten wir mit minus Werten arbeiten) ergeben sich folgende neuen Werte:






Nun stimmen die Proportionen unserer Ziegelsteine



#### Hinweis !

Der NOS-Plus wie ich ihn verwende ist mit einer zusätzlichen Funktion ausgestattet. Mit dem

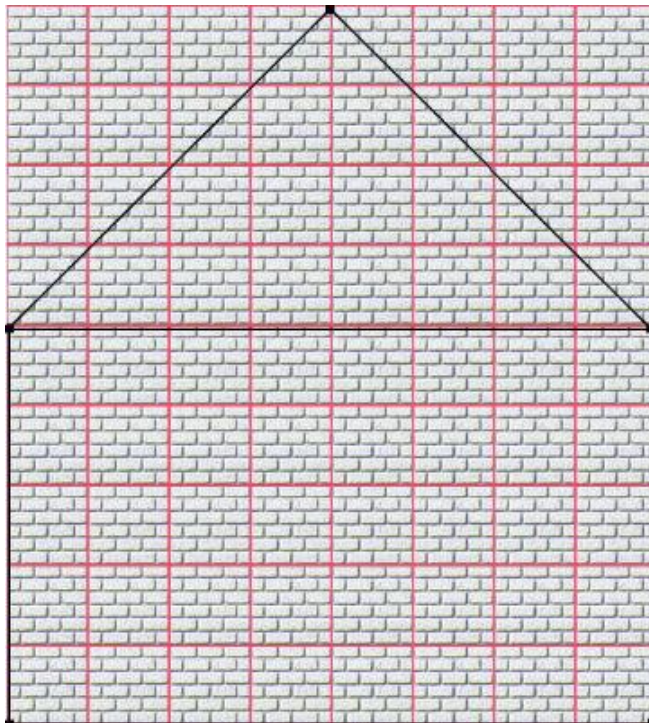
Button  können Textur-ID's durch einfachen Mausklick auf alle anderen Vertexe übertragen werden. Obwohl, wie bei der Polygon Farbe, immer der linke obere Vertex auch für die Textur bestimmend für das Polygon ist, hat die Erfahrung gezeigt, daß es ratsam ist die Textur-ID auf alle Vertexe anzuwenden.

Für die Seiten wände gilt im Prinzip das Gleiche. Wenn wir die einfache Seitenwand unserer ersten Dachform wählen, dann sind die Koordinatenwerte für Y die gleichen, nämlich für 5 für die beiden oberen Vertexe.

Die Koordinaten für X entsprechend weniger also ( $X = 500 : 60 = 8,33333...$ ) abgerundet 8 für die beiden rechten Vertexe.

Nehmen wir jedoch die Giebelwand, so sieht es etwas anders aus. Hier haben wir zusätzlich für die obere Hälfte der Hauswand ein dreieckiges Polygon. Um das mal Bildlich zu verdeutlichen, stellen wir uns die Textur einfach mal mehrfach über- und nebeneinander vor und darauf unser Polygon.

Dies würde dann etwa so aussehen: (vereinfachte Darstellung)



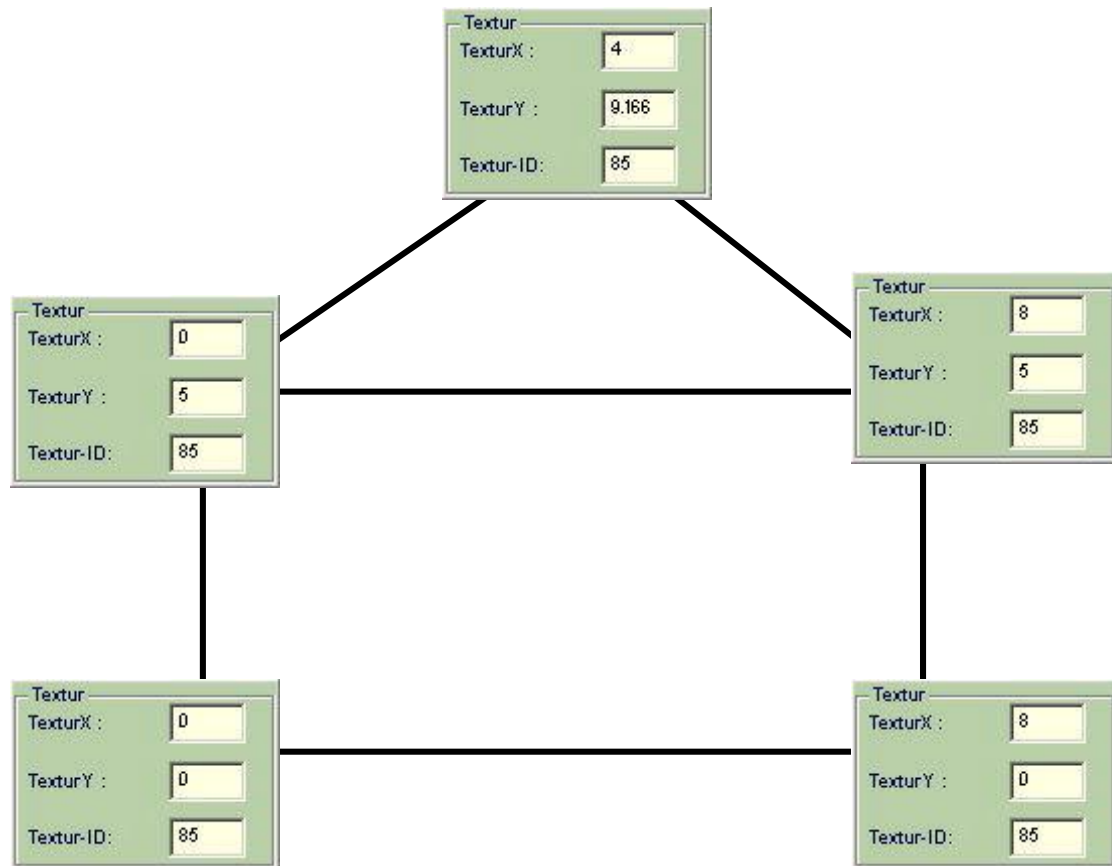
Immer davon ausgehend, daß die Textur in der Realität einen Ausschnitt von 60 x 60 cm darstellt und die Wand in unserem Falle 5 m breit und 5,5 meter hoch ist (einschl. Giebelspitze), ergeben sich Wiederholungen der Textur auf die ganze Wand von ca. Y (Höhe) 9.166mal und X (Breite) 8mal

Dabei muss ich allerdings berücksichtigen, daß ich zwei Polygone habe.

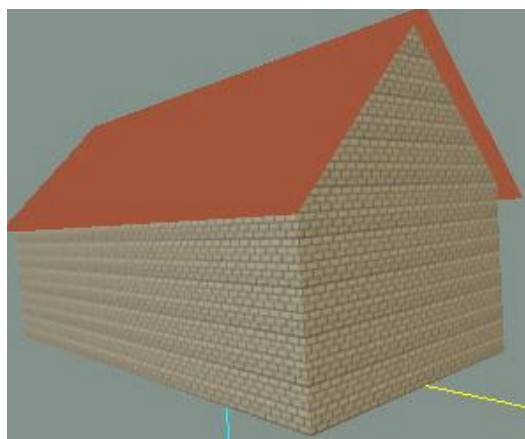
Das untere rechteckige Polygon hat die gleiche Höhe wie unsere Wand-1 (vordere und hintere Wand).

Übertragen in die Vertexe sieht das dann wie folgt aus. Ich zeige dies an Hand einer einfachen grafischen Darstellung, wobei die Eingabefenster des Vertex-Eigenschaftenmenüs jeweils an der Position der entsprechenden Vertexe stehen.





Wenn wir uns nun die Koordinatenwerte der Textur in den beiden zusammenhängenden Polygonen, nämlich dem unteren rechteckigen und dem oberen dreieckigen, genau ansehen, stellen wir fest, daß wir beim oberen Polygon unten mit den Y-Koordinaten nicht bei 0 beginnen, sondern mit den gleichen Werten mit denen unser unteres Polygon in Y-Richtung endet, nämlich bei 5. Da unser oberes dreieckiges Polygon eine Höhe von 250 cm hat, müsste theoretisch der Y-Wert eigentlich 9,16666.... sein ( $300 : 60 = 5$  Plus  $250 : 60 = 4,166666..$  also  $9,166666...$ ). Jedoch können wir derart geringe Dezimalstellen durchaus bis auf 4 oder 3 Stellen abrunden ohne daß dies gross auffallen würde. Setzen wir nun die Giebelwand ebenfalls in das 3D-Fenster:



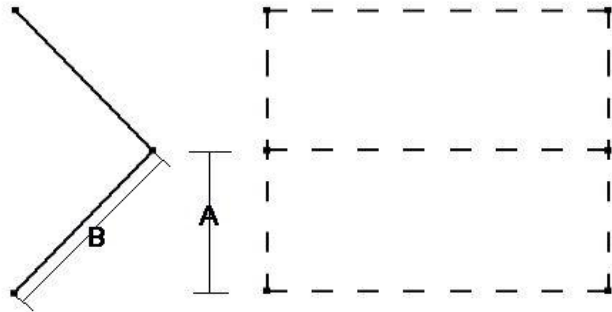
Je nach eigener Vorstellung können die Wände noch in jeder beliebigen Farbe dargestellt werden. Man sollte jedoch darauf achten, daß sich die Seitenwände in der Helligkeit etwas von der vorderen und der hinteren Wand abheben um den 3D Effekt zu verstärken. Ich belasse es erst einmal bei der ursprünglich gewählten Farbe der Polygone.




Ähnlich wie bei den Wänden gehen wir bei der Textur für das Dach vor. Nehmen wir zunächst mal das zuletzt verwendete einfache Dach. Dies besteht aus zwei zusammenhängenden Polygonen. Möchte man die Fläche nun genau berechnen, müssen wir uns die Zeichnung genauer betrachten. Wir haben ja im Kondokument das Dach in der Draufsicht gezeichnet. Wir sehen also nicht die tatsächliche Höhe der Dachfläche. Ich möchte dies mal grafisch vereinfacht darstellen.

Rechts in der Zeichnung das Dach wie wir es im Kondokument sehen, links der seitliche Querschnitt (um 90 Grad gedreht).

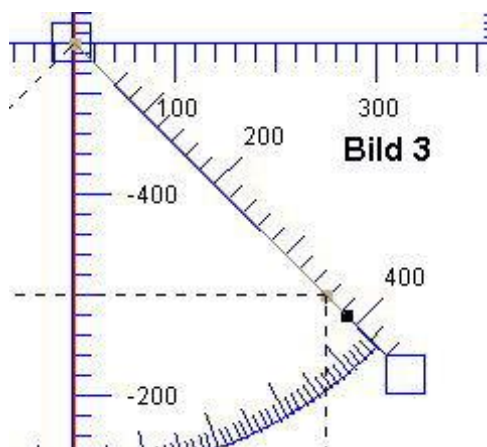
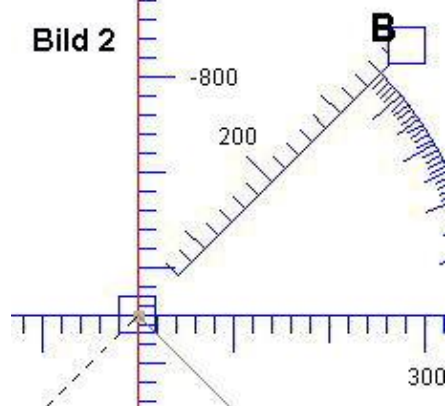
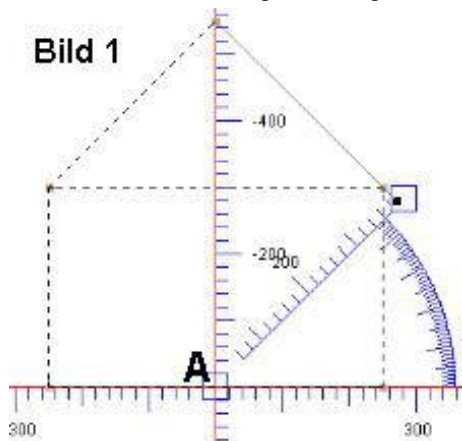
**A** bezeichnet das Maß der Polygonhöhe (Y) in der Kondatei, **B** ist das Maß der tatsächlichen Höhe des Polygons (jeweils eine Dachhälfte).

Wie nun ermittle ich die Tatsächliche Höhe der Dachhälfte? Erinnern wir uns wie wir die Werte für die einzelnen Vertexe für die Höhe (Z) ermittelt haben, nämlich durch eine Hilfslinie. Genauso ermitteln wir auch die tatsächliche Grösse der Dachfläche.



Wir öffnen die Datei „**Giebelwand**“ und maximieren das Fenster. Mit dem Werkzeug  ziehen wir wieder eine Hilfslinie nach rechts unten bis zu den Koordinaten X 270 Y 280 und setzen mit  einen Vertex. Dann klicken wir auf  und das Hilfslinial ist eingeschaltet. Dieses liegt Standardmässig auf den Nullkoordinaten von X und Y.

Wir doppelklicken nun in das Rechteck am Schnittpunkt (**A** in Bild 1), halten die Maustaste gedrückt und schieben das Lineal zum oberen Vertex, so daß der Schnittpunkt genau auf dem Vertex liegt (Bild 2). Dann doppelklicken wir in das Rechteck am Ende des beweglichen Winkelmessers (**B** in Bild 2) und ziehen diesen bei gedrückt gehaltener Maustaste über den Vertex unserer Hilfslinie (Bild 3).



Hier können wir nun am beweglichen Winkel-Lineal die tatsächliche Höhe ablesen. Dies ist natürlich bei dieser Ansicht sehr ungenau, weshalb wir dazu die Ansicht mittels der Bild-auf Taste vergrößern. (Bild 4 Seite 31)

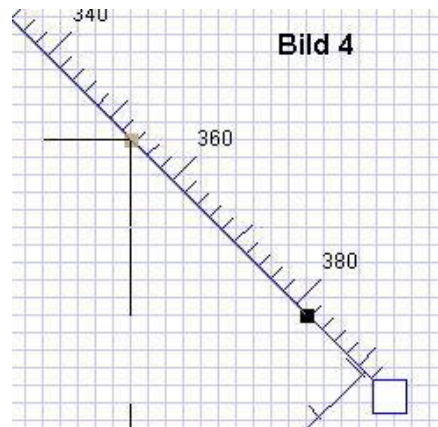
**Hinweis !**

Das Hilfslineal hat die Eigenschaft beim vergrößern oder verkleinern der Ansicht mit dem Kreuzungspunkt wieder auf seine ursprüngliche Position zu springen. Dies erschwert das Auffinden des Kreuzungsmittelpunktes bei vergrößerter Ansicht, um diesen wieder auf den richtigen Vertex zu legen. (Der eingestellte Winkel des beweglichen Winkelmessers wird jedoch beibehalten.) Um dies zu verhindern wenden wir folgenden Trick an.

**Tip !**

Bevor wir mit der Bild-auf Taste die Ansicht vergrößern, doppelklicken wir im Rechteck am Kreuzungspunkt der X-Y Linien (**A** Bild-1), halten die Maustaste gedrückt und vergrößern nun die Ansicht mittels der Bild-auf Taste der Tastatur. Auf diese Weise bleibt der Kreuzungspunkt im Bild sichtbar und wir müssen diesen nur noch auf den oberen Vertex setzen. Sollte dieser nun ausserhalb der Ansicht sein, so können wir diesen, bevor wir die Maustaste loslassen, mit den Pfeiltasten ins Bild holen.

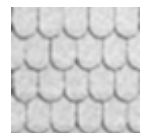
Ist der Schnittpunkt der X-Y Linien des Lineals auf dem oberen Vertex, lassen wir die Maustaste los und verschieben mittels der Pfeiltasten die Ansicht bis wir den unteren Vertex der Hilfslinie im Bild haben. Hier können wir nun Zentimetergenau die Länge ablesen. Sollte das verstellbare Lineal durch das Vergrößern seine Länge verändert haben (was meist der Fall ist), nehmen wir diesen mit Doppelklick noch mal auf und ziehen diesen wieder bis über den Vertex der Hilfslinie. (Bild 4)



Sicher gibt es mit Hilfe von Formeln auch die Möglichkeit die tatsächliche Höhe der Dachflächen zu errechnen. Nur – ich war noch nie ein Freund von Formeln und so arbeite ich eben auf diese Weise.

Wir haben also nun die tatsächliche Höhe der Dachfläche, wobei wir hier auch wieder eine runde Zahl nehmen, nämlich 380 cm.

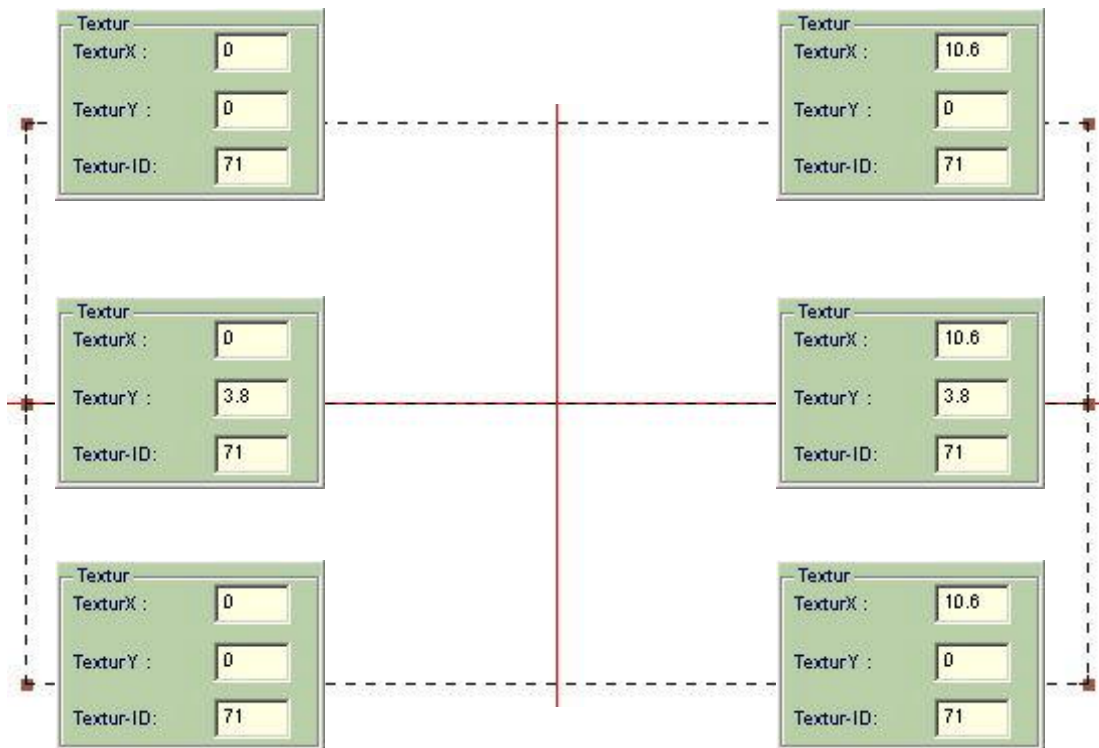
Jetzt können wir für das Dach unsere Textur berechnen. Dafür nehmen wir ebenfalls eine Textur welche als Standart in EEP vorhanden ist und auch als \*.bmp vorliegt. Ich wähle für unser Dach die Farbneutrale 64x64r2 **ID 71**. Auch bei dieser handelt es sich um eine Wiederholbare Textur.



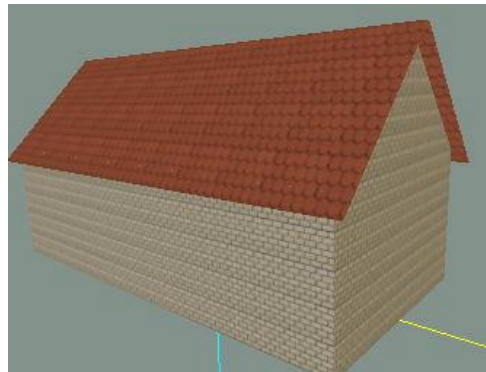
Auch bei den Dachflächen sollten wir versuchen die Textur so zu berechnen, daß die Proportionen einigermaßen der Realität entsprechen. Gehen wir einfach mal davon aus, daß die Textur einen Ausschnitt von 1 x 1 Meter in der Realität entspricht und halten wir uns die Maße einer unserer Dachflächen vor Augen, Länge (X) = 1060 cm und Höhe (Y) = 380.

Wir müssen also unsere Textur in der X-Richtung 10,6 mal wiederholen und in der Y-Richtung 3,8 mal und zwar gilt dies für beide Dachflächen. Dabei sind die Vertexe der Trennlinie der beiden Dachhälften für beide maßgebend. Das Polygon für die hintere (obere) Dachfläche, wenn wir es in Verbindung mit den Texturkoordinaten betrachten, wäre um 180 gedreht, steht also auf dem Kopf.

Wie wir nun die Textur-Koordinaten für die einzelnen Vertexe eingeben wissen wir ja bereits. In der folgenden vereinfachten Skizze zeige ich die entsprechenden Eintragungen im jeweiligen Eingabefenster des Vertex-Dialoges. Dabei steht das entsprechende Fenster neben dem Vertex zu dem es gehört.



Sind alle Eintragungen in den Vertexen gemacht, können wir das Dach erneut in das 3D-Fenster einfügen. Nun sieht ja das ganze schon etwas realistischer aus. Fehlen noch Fenster und Türen. Doch vorher nehmen wir uns der ersten Dachform an. Diese besteht ja aus zwei Kondateien, wobei ja die beiden seitlichen Dachflächen Dreiecke bilden welche um 90 Grad gedreht sind.



Nehmen wir das Dach wieder aus dem 3D-Fenster und öffnen zunächst die Datei „Dach-1“. Die Berechnung der Textur ist hier im Prinzip genauso einfach wie bei dem Dach-3. Die tatsächliche Höhe (Y) ist hier annähernd gleich die des Daches 3, so daß wir für die Y-Koordinaten den gleichen Wert nehmen können. (Der geringe Unterschied fällt letztendlich nicht auf) Allerdings müssen wir die X-Koordinaten anders berechnen.

Beim linken unteren Vertex ändert sich nichts – die Werte sind wie beim Dach-3  $X=0$   $Y=0$ . Ebenso beim linken oberen Vertex. Der mittlere linke Vertex beginnt 250 cm rechts von den beiden anderen linken Vertexen. Wenn also die linken äusseren Vertexe bei  $-500$  liegen, der linke mittlere Vertex bei  $-250$  liegt und die Textur einen Ausschnitt von  $1 \times 1$  Meter darstellt, dann muss bei diesem Vertex die X-Koordinate  $0 + 250 = 250$  sein. Wir tragen also im Vertex-Dialog beim linken mittleren Vertex als Texturkoordinaten bei  $X$  2.5 ein. Bei  $Y$  tragen wir 3.8 ein.

In den rechten Vertexen können wir berücksichtigen, daß dieses Dach etwas kürzer ist als das Dach-3.

Dort hatten wir einen Wiederholwert für  $X$  von 10,8. Für dieses Dach nehmen wir den Wert 10. Wie gesagt – etwas schummeln ist erlaubt und es fällt auch nicht auf.

Wir geben also folgende Texturkoordinaten im rechten oberen Vertex ein:  $X$  10,  $Y$  0 und ebenso im rechten unteren Vertex. Da der rechte mittlere Vertex 250 cm vor den beiden anderen in  $X$ -Richtung liegt, müssen wir diesen Wert abziehen. Wir geben also hier bei  $X$   $10 - 2,5 = 7,5$  ein und bei  $Y$  3,8.

Hier nochmal die Werte für die Texturkoordinaten wie sie im Menü „**Eigenschaften von Vertex**“ im Fenster „**Textur**“ eingegeben werden müssen:

Vertex links oben:

X = 0  
Y = 0  
ID = 71

Vertex links mitte

X = 2,5  
Y = 3,8  
ID = 71

Vertex links unten

X = 0  
Y = 0  
ID = 71

Vertex rechts oben

X = 10  
Y = 0  
ID = 71

Vertex rechts mitte

X = 7,5  
Y = 3,8  
ID = 71

Vertex rechts unten

X = 10  
Y = 0  
ID = 71

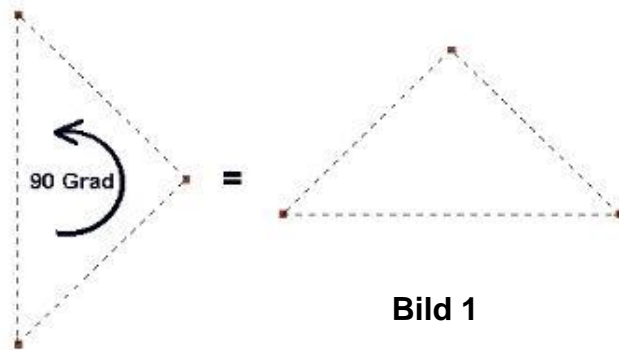
Nochmal zur Erinnerung, der NOS akzeptiert kein Komma – also immer einen Punkt an Stelle des Komma eingeben. Sollte man dies mal vergessen kommt eine entsprechende Fehlermeldung. Ich selbst habe mich schon so sehr daran gewöhnt, daß es mir oft schwer fällt mich bei anderen Programmen wieder umzugewöhnen. **J**

Haben wir alle Eintragungen gemacht, öffnen wir die datei „**Dach-2**“.

Hier müssen wir uns die Polygone jeweils um 90 Grad gedreht vorstellen, da ja hier die Spitzen, welche ja im Modell oben sind, nach links bzw. rechts zeigen.

Denken wir uns also das Dreieck der linken Seiten um 90 Grad nach links gedreht, dann ist der obere Vertex links unten, der rechte mittlere oben, lediglich der rechte untere behält die Position bei.

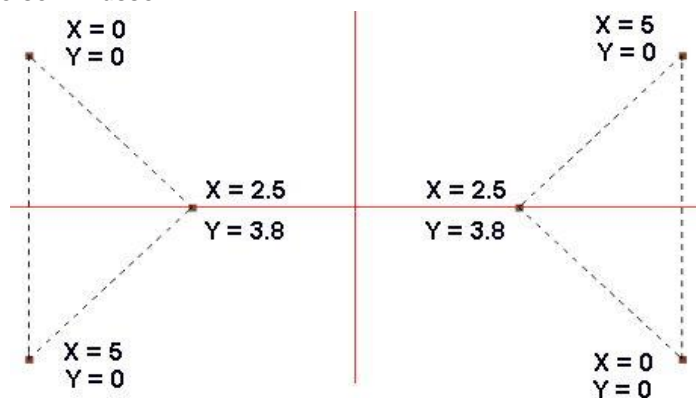
Beispiel:



**Bild 1**

Wie gesagt – wir drehen das Polygon nicht wirklich, sondern stellen und dies nur vor, um eine Orientierung zu haben wie die Koordinateneingabe der Textur sein muss. Würden wir die Y-Koordinaten der Textur nach der visuellen Stellung der Vertexe eingeben, so würden später unsere Dachziegel falsch liegen.

Um das Ganze abzukürzen zeige ich nachfolgend auf, wie die Koordinaten-Eingaben in den Vertexen der beiden Polygone sein müssen.





Wie komme ich zu diesen Koordinaten?

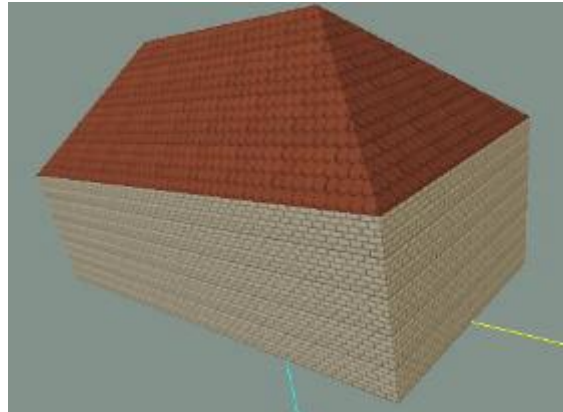
Wie bereits auf Seite 31 erwähnt, gehe ich davon aus, daß die Textur einen Ausschnitt von 1x1 Meter darstellt. Die Null-Koordinaten für X und Y einer Textur sind immer links unten. Da unsere Polygone jedoch um 90 Grad gedreht gezeichnet sind, muss auch die Textur entsprechend auf dem Polygon „gedreht“ werden. Somit ist, bezogen auf das linke Polygon, der Vertex oben auch die linke untere Ecke in der Textur, also  $X = 0$  und  $Y = 0$ . Da das Polygon eine Länge von ca. 5 Meter hat, muss die Textur auch in der X-Richtung 5 mal wiederholt werden. Dies ist der untere Vertex des linken Polygons. Die tatsächliche Höhe der Dachfläche haben wir bereits für die vordere Dachfläche ermittelt und kann somit für die seitlichen Dachflächen übernommen werden. Der rechte Vertex, also die Höhe des Daches stellt der mittlere Vertex dar und befindet sich auch genau in der Mitte der X-Achse, also die Hälfte von 500 = 250, oder bezogen auf die Textur 2,5

Für das rechte Polygon sind die Daten die gleichen, nur eben Spiegelverkehrt.

Das fertige Dach sieht nun so aus:

Natürlich müssen für diese Dachform die beiden Seitenwände „Giebelwand“ gegen die ursprünglichen Wände „Wand-2“ ausgetauscht werden.

Für die nächsten Schritte, das Texturieren der Fenster und der Türe, tauschen wir die beiden Seitenwände und das Dach jedoch wieder aus und verwenden hierzu die „**Giebelwand**“ und das „**Dach-3**“



Für die Gestaltung von Türen und Fenstern gibt es mittlerweile sehr viele schöne, von Konstrukteuren erstellte Texturen, welche Teilweise auch frei verwendbar sind. Allerdings sollte man den Autor der jeweiligen Textur vor deren Verwendung fragen.

Ich verwende jedoch für unser Beispiel erst mal eine Textur aus der Grundausstattung von EEP, da von diesen auch die BMP's vorhanden ist, und wir nur mit dieser im NOS arbeiten können.

Diese ist zwar alles andere als schön, aber wir wollen uns ja zunächst einmal mit den Grundkenntnissen vertraut machen. Wie sie eigene Texturen erstellen, zeige ich in einem anderen Kapitel.

Ich wähle für unsere Fenster die Textur ID **62** (64x64n). Sie eignet sich deshalb am besten weil sie zum einen nicht gross ist, also wenig Ressourcen braucht, und zum anderen zwei verschiedene Fenster enthält. (Abb. in Originalgröße)



Nun – Texturkoordinaten sind ganz ohne Hilfsmittel nicht so einfach zu ermitteln. Wie bereits an anderer Stelle erwähnt, muss man die Textur immer als Ganzes sehen, also Texturkoordinaten von 0 bis 1, wobei es keine Rolle spielt wie gross eine Textur ist. Am besten man stellt sich das so vor, daß die Textur in der X-Richtung von links ausgehend am rechten Rand 100 % ist und in der Y-Richtung vom unteren aus, der obere Rand ebenfalls 100 % ist. Will man aus einer Textur nur einen bestimmten Bereich auf das Polygon „kleben“, so müsste man sich dieses Gedanklich auf die Textur projizieren und zwar passend auf den Ausschnitt der auf das Polygon soll. In unserem Falle wäre das eines der Fenster in der Textur **62**. Um sich das besser vorzustellen, habe ich diese Textur etwas vergrößert und eines der Fensterpolygone darübergelegt. Natürlich muss dabei das Seitenverhältnis dem Teil der Textur angepasst werden, das auf das Polygon soll.

Die grüne Linie stellt unser Fensterpolygon dar.



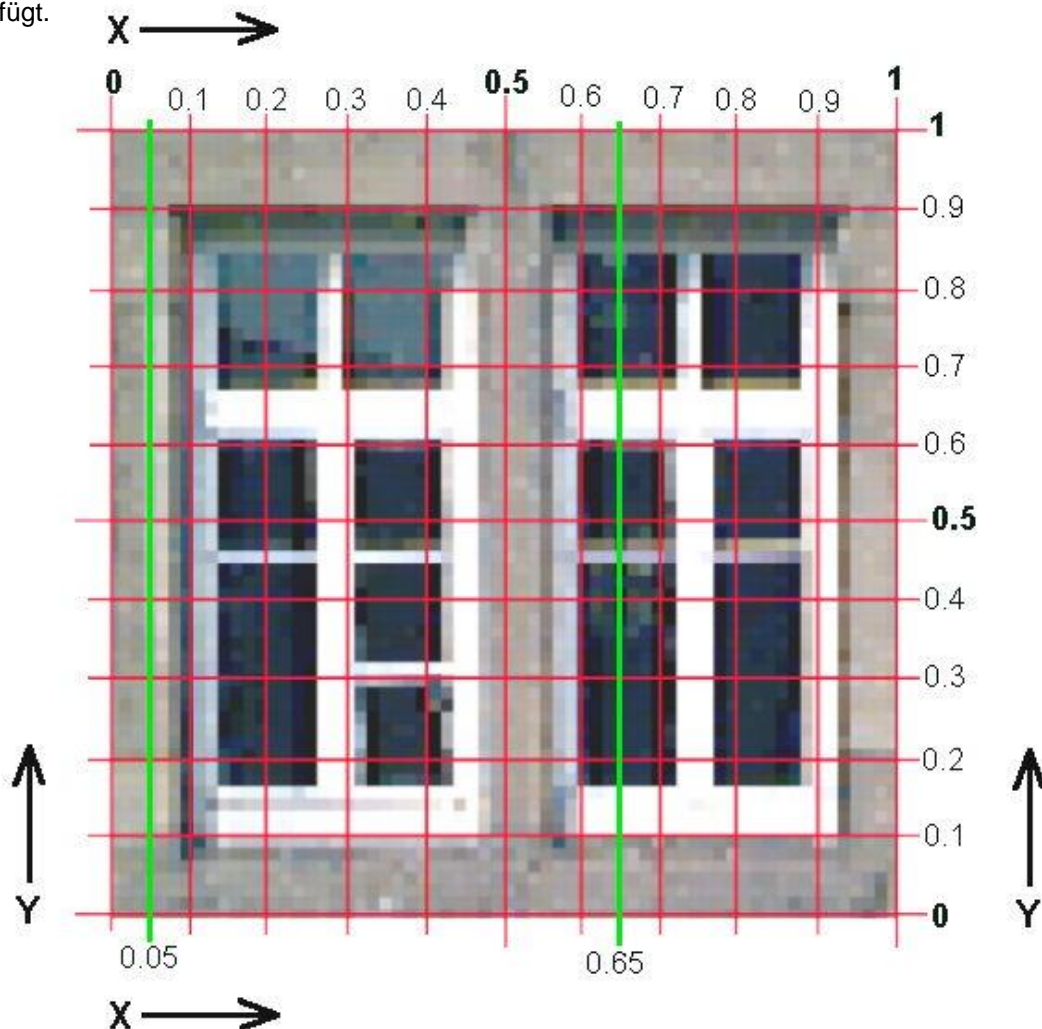
Wie aber ermittle ich die Koordinaten für unsere Fenster?

#### Hinweis !

Es gibt mittlerweile einige praktische Tools bzw. kleine Programme, teilweise von Konstrukteuren geschrieben, mit welchen sich die Texturkoordinaten schnell und einfach berechnen lassen. Diese Tools sind jedoch nicht frei zugänglich. Ich selbst arbeite seit längerem mit dem Tool EEPV, das von CTR entwickelt wurde und nicht nur das Ermitteln der Koordinaten erleichtert, sondern auch die Auswahl von eigenen oder Fremdtexturen. Leider ist die Installation dieses Tools nicht ohne tiefgreifende PC-Kenntnisse möglich. Auch wurde das Tool, aus Gründen die ich hier nicht näher ausführen möchte, nicht weiterentwickelt. Ich werde jedoch am Schluss dieses Tool kurz beschreiben.

Wie ich in meiner Anfangszeit Texturkoordinaten ermittelt hab möchte ich hier mal etwas ausführlicher beschreiben.

Ich habe mir die Textur auf DIN A4 ausgedruckt und zwar so, daß die Textur genau das Maß 10 x 10 cm auf dem Papier hatte. Darüber habe ich mir ein Raster gezeichnet, das die Textur in 10 x 10 Felder einteilte. Nun konnte ich mit einem Lineal mit Milimeteerteilung die jeweiligen X und Y Koordinaten ermitteln. Im Beispiel habe ich noch die Koordinaten bis zur ersten Dezimalstelle hinzugefügt.



Das rote Raster markiert praktisch die erste Dezimalstelle. Ist die Abbildung nun genau 10 x 10 cm, so kann man mit einem Lineal mit Milimeter-Einteilung auch noch die zweite Dezimalstelle ermitteln. Wenn man noch genauere Koordinaten braucht, z.B. bei sehr grossen Texturen mit sehr kleinen Details, so muss man sich hier durch probieren Schritt für Schritt an die richtigen Koordinaten herantasten.

Bevor wir jedoch die Texturkoordinaten eintragen ändern wir noch die Polygonfarbe über die Vertexe, wobei wir reines Weiß vermeiden sollten. Ein ganz helles Grau wäre hier von Vorteil. Wer sich in der Farbgebung unsicher ist, kann sich dies bei den beigefügten Dateien ansehen.

Öffnen wir zunächst einmal die Datei „**Fenster-1**“ und nehmen uns das linke äussere Polygon vor. Hier möchte ich das linke Fenster aus der Textur übernehmen. Die Koordinaten sehen wie folgt aus:

Links oben  
 Textur X = 0.0703  
 Textur Y = 0.914  
 Textur ID = 62

Rechts oben  
 Textur X = 0.49  
 Textur Y = 0.914  
 Textur ID = 62

Links unten  
 Textur X = 0.0703  
 Textur Y = 0.0703  
 Textur ID = 62

Rechts unten  
 Textur X = 0.49  
 Textur Y = 0.0703  
 Textur ID = 62



Spielen sie ruhig mal etwas mit den Texturkoordinaten und sehen sie sich die entsprechenden Ergebnisse im 3D-Fenster an. So bekommen sie schnell ein Gefühl dafür und sie werden feststellen, daß es doch nicht ganz so schwer ist wie es zu Anfang den Anschein hat.

Jetzt tragen wir noch die Textur-Koordinaten in das daneben liegende Polygon ein.

Links oben  
 Textur X = 0.5234  
 Textur Y = 0.914  
 Textur ID = 62

Rechts oben  
 Textur X = 0.96  
 Textur Y = 0.914  
 Textur ID = 62

Links unten  
 Textur X = 0.5234  
 Textur Y = 0.0703  
 Textur ID = 62

Rechts unten  
 Textur X = 0.96  
 Textur Y = 0.0703  
 Textur ID = 62



Sehen wir uns mal die Y Koordinaten der beiden Fenster an so stellen wir fest, daß diese in den oberen Vertexen der beiden Polygone immer identisch ist, ebenso in den unteren Vertexen. Ebenso haben die beiden linken sowie die beiden rechten Vertexen in jedem Polygon identische Koordinaten. Liegen doch diese senkrecht bzw. waagrecht immer auf einer Linie.

Um nun auch die übrigen Fenster mit der Textur zu belegen, müssen wir nicht nocheinmal jeden Vertex bearbeiten. Wir ersparen uns diese Arbeit indem wir die Fenster jeweils kopieren und die noch nicht mit Textur versehenen Polygone damit ersetzen. Dabei bleibt es jedem selbst überlassen welches der beiden Fenster er kopiert. Ich habe für die Datei „**Fenster-1**“ das rechte der beiden Fenster gewählt und ersetze damit die beiden Polygone auf der rechten Seite.

Genauso verfahren wir mit den übrigen beiden Kondateien „**Fenster-2**“ und „**Fenster-3**“.

Wir kopieren die beiden linken Fenster in die Zwischenablage und öffnen nacheinander die beiden anderen Kon-Dateien, um auch dort die Polygone auszutauschen.

**Achtung !!** Wenn wir die Polygone in der Datei „**Fenster-3**“ ersetzt haben müssen wir hier noch die Werte für **Höhe Z** ändern. Diese sind ja bei den Fenstern der Vorder- und Rückseite jeweils **-252**, für die Fenster der beiden Seitenwände jedoch hatten den Wert **-502**.

Beim Kopieren von Polygonen bleiben alle Einstellungen erhalten und werden mit übertragen.



Bis auf die Türe sieht unser Häuschen schon recht ansehnlich aus.



Suchen wir uns also unter den Standard-Texturen eine geeignete Haustüre.  
 Ich wähle für unser Beispiel die Textur ID 19 (0019) da ich hier keinen Ausschnitt berechnen muss, sondern die Textur komplett in das Polygon übernehmen kann.  
 (Abb. in Originalgröße)



Das Polygon der Haustüre ist ja in der Datei **“Fenster-1”** enthalten und somit öffnen wir diese und geben die Koordinaten für die Textur wie folgt ein:

Vertex links oben	Vertex rechts oben
Textur X = 0	Textur X = 1
Textur Y = 1	Textur Y = 1
Textur ID = 19	Textur ID = 19
 Vertex links unten	 Vertex rechts unten
Textur X = 0	Textur X = 1
Textur Y = 0	Textur Y = 0
Textur ID = 19	Textur ID = 19



Unser Haus ist nun mit Fenstern und einer Türe ausgestattet und somit fast fertig. FAST !!  
 Was macht ein richtiges Haus ausser Fenster und Türe noch aus? Da wären Dachrinnen, ein Schornstein, Schneefanggitter und evtl. noch Dachfenster oder noch besser eine Dachgaube.

Doch bevor wir uns damit beschäftigen möchte ich zeigen wie man eigene Texturen erstellen und diese in EEP einbinden kann.

Dazu sollten wir uns aber zuerst einmal anschauen wie Texturen für EEP beschaffen sein müssen.  
 Zunächst einmal das Dateiformat. Dieses ist für den Nostruktur das Windows-Bitmap kurz bmp, nicht zu verwechseln mit OS2, welches ebenfalls ein BMP-Format ist.

Die Farbtiefe hier ist 24 Bit Echtfarben, also 72 Pixel/Zoll.

Dann ist noch die Bildgröße maßgebend, welche nicht xbeliebig sein darf, sondern, beginnend vom kleinstmöglichen Maß ausgehend immer das doppelte der Seitenlänge betragen muss.

Rein theoretisch wäre das kleinste Maß 1 X 1 Pixel. Die kleinste, von mir verwendete Textur hat eine Seitenlänge von 16 X 16 Pixel, was jedoch die Ausnahme darstellt. Dazu später mehr.

Hier mal eine Reihe möglicher Texturgrößen, ausgehend von 16 X 16:

32 X 32, 64 X 64, 128 X 128, 256 X 256, 512 X 512 usw. wobei auch Formate wie 64 X 128, 128 X 256, 256 X 512 usw.

Ich unterscheide noch zwei Arten von Texturen und zwar „Sammeltexturen“, das sind solche Texturen mit verschiedenen Inhalten, wobei diese jedoch möglichst thematischen Inhalts sein sollten, das heißt es wäre nicht sinnvoll in einer Textur Teile eines Rollmaterials und Teile für eine Immobilie zusammen zu fassen. Und dann gibt es noch die sogenannten „Wiederholbaren“ Texturen.

Im Prinzip kann jede Textur wiederholbar sein, da dies über das „Texturescript“ gesteuert wird.

Es macht aber keinen Sinn z.B. eine Sammeltextur, welche verschiedene Elemente enthält, zu wiederholen. Ein Beispiel für eine wiederholbare Textur wäre die bereits für die Hauswand verwendete Ziegelsteintextur sowie die Textur der Dachziegel. Für derartige Oberflächen oder Strukturen genügt ein kleiner Ausschnitt, der dann auf beliebig grossen Flächen wiederholt wird. Solche Texturen können relativ klein gehalten werden, womit wir Ressourcen sparen.

Auch die Textur der beiden Fenster ist eine wiederholbare Textur.

Hier ein Beispiel:

Ich habe ein Polygon mit den Maßen  
 X = 1200 und Y = 500 cm  
 Darauf wiederhole ich die Textur ID 62  
 X = 4 x und Y = 2 x



Wiederholbare Texturen müssen die Fläche komplett ausfüllen, d.h. es darf nirgends ein Pixel mit den Farbwerten R0, G0, B0 sein, da schwarz immer durchsichtig dargestellt wird, es sei denn dies wäre beabsichtigt. Dazu aber später mehr.

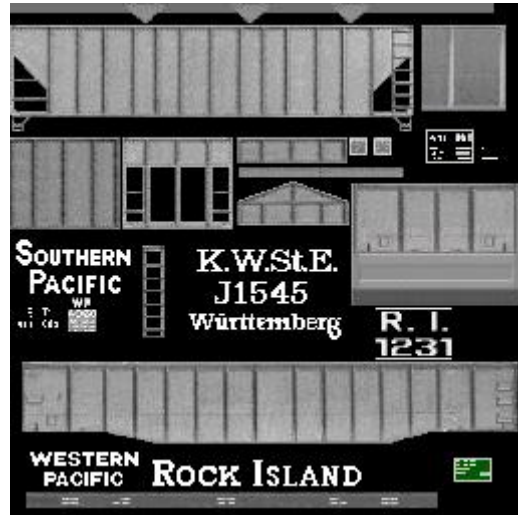
Bei Sammeltexturen muss alles was nicht dargestellt werden soll, also der Hintergrund schwarz sein, also die Werte R0, G0, B0 haben.

Hier ein Beispiel einer solchen Textur:

Diese Textur hat die Grösse 256 X 256 mit einer Farbtiefe von 24 Bit im \*.bmp-Format.

Wie der Betrachter unschwer erkennen kann, enthält die Textur Teile verschiedener US-Güterwagen. Ausserdem werden alle Elemente nur in Grauwerten dargestellt. Dies hat den Vorteil, daß man damit, mit nur dieser einen Textur beliebig viele Farbvarianten erstellen kann, denn wie wir bereits bei der Textur für unsere Hauswand und die Dachziegel gesehen haben, kann die Farbe auch über die Vertexe bestimmt werden.

Diese \*.bmp brauchen wir für die Arbeit im Nostruktur. Um aber auch damit arbeiten zu können, muss sie noch in der Texturen.txt registriert werden.



### Tip!

Ich erstelle neue Texturen grundsätzlich nie im Original Ressourcen/Parallels-Ordner, sondern in einem extra dafür angelegten Arbeitsverzeichnis auf einer zweiten Festplatte. Dies hat den Vorteil, daß man zum einen die Übersicht über seine eigenen Texturen behält und zum anderen immer eine Kopie hat auf die man zurückgreifen kann wenn mal was daneben geht.

Ich kopiere also zunächst die Textur in den Ordner „Ressourcen/Parallels“. Im gleichen Ordner befindet sich auch die Datei „Texturen.txt“. Dabei handelt es sich, wie man an der Endung \*.txt erkennt um eine einfache Text-Datei.

Hier tragen wir die Daten unserer Textur ein. Die Zeile für diese Textur würde wie folgt aussehen:

```
Textur{id(945) name(0945) automipmap() billboard( ) magfilter(1) minfilter(5) }
```

(Natürlich kommt bei Textur „ID“ und bei „Name“ eine eigene freie ID-Nummer zum Einsatz.)

## Bedeutung der Parameter in der ASCII-Datei „Texturen.txt“

### Einleitung:

In der ASCII - Textdatei im Verzeichnis Parallels für die Texturen wird die Verknüpfung von Dateinamen zu sog. Textur IDs hergestellt. Diese IDs können im Nostruktur benutzt werden, um Flächen Texturen zuzuordnen. Ergänzend zur ID können einige Eigenschaften der Textur festgelegt werden, diese werden im Folgenden beschrieben.

### Basis:

Der gesamte Skripttext muss von einem Block aus geschweiften Klammern eingerahmt sein, der mit dem Schlüsselwort Basis bezeichnet ist. Die Parameter dieses Blockes bestehen ausschließlich aus Blöcken mit dem Schlüsselwort Textur.

### Textur:

Für jede Textur wird ein Block mit Namen Textur angelegt. Die Reihenfolge dieser Blöcke ist egal. In folgender Tabelle werden die einzelnen Parameter eines Texturblocks kurz erklärt. Es folgen Behandlungen einiger Themen, wie Texturfilter, Mipmap und Texturwiederholung.



- id** Unter diesem Parameter wird die eindeutige Textur ID angegeben, und zwar als ganze Zahl, Null ist kein gültiger Wert. Muss unbedingt angegeben werden.
- name** Name der Datei mit dem Texturbild ohne Endung .bmp Muss unbedingt angegeben werden.
- automipmap** Gibt an, dass Mipmaplevel mit verringerter Auflösung automatisch erstellt werden sollen. Es muss kein Wert angegeben werden. Um das Mipmapping tatsächlich einzuschalten, siehe Texturfilter. Default: aus
- billboarding** Gibt an, dass Texel mit schwarzer Farbe, also RGB - Werten 0,0,0 wegzulassen sind. Eine Fläche kann so durchbrochen werden. Default: aus
- magfilter** Gibt den Magnification - Filter an. Gültige Werte sind : 0, 1. Für eine Beschreibung siehe Abschnitt Texturfilter. Default: 0
- minfilter** Gibt den Minification - Filter an. Gültige Werte sind : 0, 1, 2, 3, 4, 5 Für eine Beschreibung siehe Abschnitt Texturfilter. Default: 5
- repeat\_s** Wenn gesetzt, werden Texturen in X-Richtung wiederholt, wenn die Texturkoordinaten den Bereich 0.0f - 1.0f überschreiten. Es muss kein Wert übergeben werden. Default: aus
- repeat\_t** Wenn gesetzt, werden Texturen in Y-Richtung wiederholt, wenn die Texturkoordinaten den Bereich 0.0f - 1.0f überschreiten. Es muss kein Wert übergeben werden. Default: aus
- opacity** Gibt an, dass und in welchem Grade eine Textur eine Fläche durchsichtig erscheinen lassen soll. Gültige Werte sind 0.0 (unsichtbar) bis 1.0 (opak) Default: aus

#### Mipmapping:

Mipmapping bezeichnet eine Technik, bei der Texturen in größeren Entfernungen mit verringerter Auflösung dargestellt werden. Dies kann Flimmereffekte beseitigen und gibt bessere Kontrolle über die Darstellung von Objekten in größerer Entfernung. Mit dem Parameter automipmap() werden die Mipmaplevel automatisch berechnet, indem die Farbwerte jeweils vier Texel zu einem verrechnet werden (Einfache Mittelung). Mit dem Schlüsselwort MipMapLevel hat man die Möglichkeit, von Hand erzeugte Mipmaplevel anzugeben. Dies ist zum Beispiel im Zusammenhang mit Billboarding sinnvoll. (siehe unten)

#### Texturfilter:

Grundsätzlich steht der Renderer vor dem Problem, dass er entscheiden muss, welches Bildelement (Texel) einer durch die 3D -> 2D Projektion unter einem Pixel zu liegen kommenden texturierten Fläche die Farbwerte für das Pixel liefern soll. Es gibt zwei Situationen, in der einen sind die texturierten Flächen soweit weg, dass mehrere Texel in ein Pixel fallen: hier wird der sog. Minifikation - Filter aktiv. In der anderen überdeckt ein Texel mehrere Pixel, hier wird die Darstellung durch den Magnifikationfilter beeinflusst.

##### Werte für den Minifikationfilter:

- 0: GL\_NEAREST, das dem Pixelmittelpunkt zunächst liegende Texel bestimmt die Farbe des Pixels.
- 1: GL\_LINEAR, die vier dem Pixelmittelpunkt zunächst liegenden Texel werden zur Farbe des Pixels verrechnet.
- 2: GL\_NEAREST\_MIPMAP\_NEAREST, wie GL\_NEAREST, nur werden abhängig von der Entfernung Mipmaplevel zur Farbbestimmung benutzt.
- 3: GL\_LINEAR\_MIPMAP\_NEAREST, es wird abhängig von der Entfernung ein Mipmaplevel benutzt, die endgültige Farbe des Pixels jedoch aus den vier zunächstliegenden Texeln in dem Mipmaplevel gemittelt.
- 4: GL\_NEAREST\_MIPMAP\_LINEAR, es werden die beiden am besten zu der Entfernung passenden Mipmaplevel benutzt. Relevant ist aber immer nur das Texel, was dem Pixelmittelpunkt am nächsten liegt.
- 5: GL\_LINEAR\_MIPMAP\_LINEAR, es werden die beiden am besten zu der Entfernung passenden Mipmaplevel benutzt. Die Farbwerte, die jeder Mipmaplevel beisteuert, werden aus den dem Pixelzentrum nächstliegenden Texeln gemittelt.

#### Werte für den Magnifikationfilter:

0: GL\_NEAREST, es wird das Texel die Farbe eines Pixels bestimmen, dessen Zentrum dem Pixelzentrum am nächsten liegt.

1: GL\_LINEAR, es werden die vier einem Pixel zunächst liegenden Texel zur Farbbestimmung herangezogen.

#### Texturwiederholung:

Mit den Parametern repeat\_s, repeat\_t gibt man, an, dass die Textur in X bzw. Y Richtung wiederholt werden soll, wenn die Texturkoordinaten den Bereich zwischen 0 und 1 überschreiten. Dies kann im Zusammenhang mit den Texturfiltern zu Problemen führen. Wenn beim Filtern Texelwerte herangezogen werden müssten, die jenseits der Texturgrenzen liegen, können Streifen entstehen, die die Farbe der gegenüberliegenden Texturseite einmischen. Ohne Repeat hätten die verwendeten nicht existierenden Texel die Farben der letzten existierenden.

Auch beim Billboarding kann es passieren, dass unsichtbare Texel an einem Texturrand wieder sichtbar werden. In solchen Fällen kann man entweder die Filtermethoden auf NEAREST stellen, oder aber auf REPEAT verzichten.

#### Billboarding:

Texel mit der Farbe RGB: 0,0,0 (schwarz) werden nicht gezeichnet. Man sieht dann an diesen Stellen die Szenerie dahinter. Beim Auto - Mipmapping können hier Probleme auftreten, da vollautomatisch nicht immer korrekt bestimmt werden kann, ob ein Texel jetzt unsichtbar ist oder nicht. Etwa, wenn zwei unsichtbare und zwei sichtbare zu einem verrechnet werden. In solchen Fällen kann es sinnvoll sein, spezielle Mipmaplevel von Hand anzugeben.

Einige der vorgenannten Parameter sind für die \*.BMP, also den Nostruktur ohne Bedeutung, jedoch für die \*.DDS, welche für EEP benötigt wird, um so mehr. Die Parameter für repeat und opacity jedoch nur im Bedarfsfalle.

Um nun aus unserer \*.BMP eine für EEP notwendige \*.DDS zu erzeugen, müssen wir zusätzlich zu unserer Textur einen sogenannten AlphaChannel erstellen. Dieser stellt im Prinzip eine Schwarz-weiß Schablone unserer Textur dar, wobei alle Flächen die dargestellt werden sollen weiß sind, also die Farbwerte R255 – G255 – B255 haben und alles was nicht dargestellt werden soll, also durchsichtig sein soll den Farbwert R0 – G0 – B0.

Am Beispiel einer meiner Texturen möchte ich zeigen wie ich selbst einen solchen AlphaChannel mit meinem Bildbearbeitungsprogramm erstelle. (Abb. zeigen Ausschnitte der Textur in Originalgröße)

Ich öffne die Textur **0940.bmp** in meinem Malprogramm und erstelle ein Duplikat. Dann setze den Farbwert des Markierungswerkzeuges auf 1 (oder 0), markiere alle schwarzen Flächen welche durchsichtig sein sollen und kehre die Markierung um.



Anschliessend fülle ich die so markierten Flächen mit dem Füllwerkzeug mit der Farbe weiß.

Diese Datei speichere ich unter dem gleichen Namen ab wie die Original Textur, allerdings mit dem Zusatz

**\_a.bmp**

Wichtig ist hier der **Unterstrich** und **a**, sonst erkennt DxTex.exe diese nicht als AlphaChannel.

Diese Textur hat den Dateinamen **0940\_a.bmp**



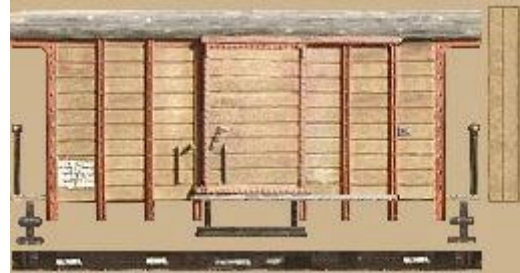
Der ganze Vorgang dauert nur wenige Sekunden falls man ein Malprogramm mit den entsprechenden Werkzeugen zur Verfügung hat.

**Hinweis !**

Trotz AlphaChannel kann es gelegentlich vorkommen, daß Objekte, welche über unregelmässige Aussenkanten verfügen, jedoch auf ein rechteckiges Polygon projiziert werden (Beispiel Rad), mit dunklen oft schwarzen Rändern dargestellt werden. Dies resultiert aus der Tatsache, daß das Programm nebeneinanderliegende Pixel miteinander verechnet. (siehe auch Seite 40/41) Es hat sich deshalb bei den meisten Konstrukteuren durchgesetzt, für die \*.BMP aus welcher die \*.DDS erstellt wird, ein Duplikat der Textur mit angepasstem Hintergrund zu verwenden. Die Farbe kann sogar die gleiche sein wie die des Objektes, da ja das tatsächliche Objekt durch den AlphaChannel wieder „herausgestanzt“ wird.

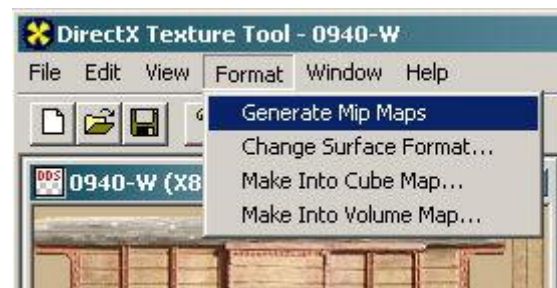
Diese bmp wird im Prinzip genauso erstellt wie der AlphaChannel, nur daß hier der markierte Hintergrund eingefärbt wird. Im Prinzip würde es reichen, wenn nur einen 2-3 Pixel breiten Rand um jedes Element einfärbt. Dies mache ich aber nur wenn in einer Textur sehr unterschiedliche Farben verwendet werden.

Dieses Duplikat speichere ich unter dem selben Namen ab wie meine Original bmp, jedoch mit einem zusätzlichen Buchstaben (im Beispiel -w) 0940-w.bmp



Um nun aus diesen beiden bmp's eine DDS zu konvertieren, benötigen wir das Tool DxTex.exe. Dieses ist Freeware und man kann es sich auf der Website von MicroSoft downloaden.

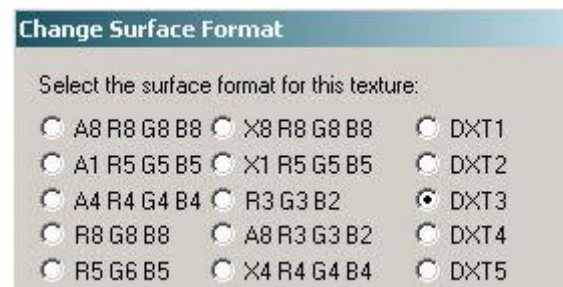
Sind also alle unsere BMP's gespeichert, öffnen wir DxTex.exe und laden zunächst unsere BMP, im Beispiel die oben genannte 0940-w.bmp und Erstellen zunächst die MipMap Level. (Bild) (siehe auch Seite 39, automipmap)



Anschliessend komprimieren wir die Textur. Dazu gehen wir ins Menü (Bild)

„Change Surface Format.....“ und wählen eine Komprimierungsstufe.

Sollten sie über eine neuere Version von DxTex.exe verfügen, so wählen sie aus der linken oder der rechten Spalte. Komprimierungsstufen der mittleren Spalten akzeptieren keinen AlphaChannel. Ich wähle für diese Textur DXT3



Nun gehe ich über „File“ ins nächste Menü, um den AlphaChannel hinzuzufügen.

Wichtig! Nur über die im Bild markierte Option, „Open Onto Alpha Channel Of This Texture....“ (Bild)



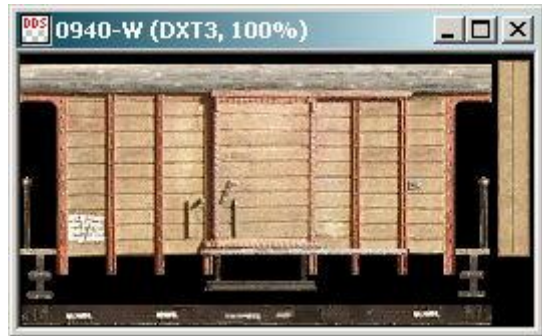
Im Dateifenster suche ich die Datei 0940\_a.bmp  
Wähle sie aus und füge sie mit OK ein.



Die so fertiggestellte Textur speichere ich als **0940.dds** ab.

Obwohl ich hier eine BMP mit farbigem, also den Objekten angepassten Hintergrund verwende, wird dieser nicht mehr angezeigt.

Die Farbe wird durch den AlphaChannel praktisch eliminiert und wieder durchsichtig.



Zum Schluss dieses Kapitels noch ein Wort zu den verschiedenen Komprimierungsstufen in DxDex.exe.

Halten wir uns vor Augen, daß wir für eine DDS Datei zwei BMP's verwenden, die eigentliche Textur und den AlphaChannel. Da beide Dateien mit 24 Bit Farbtiefe erstellt sind, ist die Dateigrösse bei beiden gleich. Das heisst also die Dateigrösse der DDS verdoppelt sich. Bei meiner Textur 0940.bmp und der 0940\_a.bmp sind das bei einer Grösse von (Originalgrösse) 256 X 256 2 mal 193 KB. Die DDS hätte also eine Dateigrösse von 342 KB, wenn wir diese im Standard-Format lassen. Dies wäre die Einstellung **A8 R8 G8 B8** in der linken Spalte. Mit dieser Einstellung findet also keine Komprimierung statt und sollte daher nur für kleinere und sehr Detailreiche Texturen Verwendung finden.

Um die jeweils passende Komprimierung zu finden, ist es auf jeden Fall unerlässlich, die jeweilige Stufe in EEP für jede neu erstellte Textur zu testen.

Einen Anhaltspunkt geben die Formate die ich und auch die meisten anderen Konstrukteure verwenden.

(In der Aufstellung gibt die Zahl in **Klammer** die Dateigrösse einer Textur 256 x 256 an)

**A8 R8 G8 B8** – 32 Bit- 8 Bit je Kanal R G B und Alpha.....( **342 KB**)  
Sollte nur für sehr kleine Texturen oder wenn es auf sehr grosse Wiedergabequalität ankommt, verwendet werden

**A1 R5 G5 B5** – 16 Bit – 1 Bit Alpha, 5 Bit je Kanal R G B .....( **171 KB**)  
Gängige Komprimierung bei kleinen bis mittleren oder auch Detailreichen Texturen

**DXT3** – 4 Bit Alpha .....( **86 KB**)  
Von mir die am meisten verwendete Komprimierung für mittlere bis grosse Texturen

**DXT1** – 1 Bit Alpha .....( **43 KB**)  
Von mir bei grossen bis sehr grossen Texturen verwendet, wenn wenig oder keine feinen Details in der Textur enthalten sind.

Dies sind wie gesagt nur Vorschläge. Letztendlich sollte man selbst bei neuen Texturen den optimalen Kompromiss zwischen akzeptabler Wiedergabe und Kompressionsstufe durch probieren ermitteln.



## Kapitel 3

Wenden wir uns nun wieder unserem zuletzt konstruierten Haus zu. Bei oberflächlicher Betrachtung könnte man es schon als fertiges Modell betrachten, auch wenn noch einige Kleinigkeiten wie Kamin, Dachrinnen usw. fehlen. Doch sehen wir uns zunächst die beiden folgenden Bilder an und vergleichen die optische Wirkung.



Im linken Bild sehen wir unser Haus wie wir es zuvor konstruiert haben. Das vorstehende Dach zeigt auf der Unterseite die gleiche Textur, nämlich die Ziegel wie auf der Oberseite. Dies ist nicht sehr realistisch. Ebenso sieht das Dach aus als ob es nur aus bedruckter Pappe wäre. Auch fehlt hier der Schatten, welcher das Ganze erst realistisch wirken lässt und dem Modell die richtige 3D Wirkung verleiht.


Werden wir also unser Modell auf, indem wir dies an unserem Haus Schritt für Schritt in die Tat umsetzen.

Um das Dach bzw. die Dach-Unterseite realistischer zu gestalten gibt es zwei Möglichkeiten, einmal die Polygonspar-Methode und die etwas aufwendigere Möglichkeit mit zusätzlichen Polygonen für die Dachkante unten und an der Giebelseite.

Zunächst die Spar-Methode.

Ich gehe mal davon aus, daß wir den NOS noch geöffnet und alle Teile im 3D-Fenster vorhanden sind. Wenn ja entfernen wir das **Dach-3** aus dem 3D-Fenster und speichern es mit dem neuen Namen **Dach-3-u** ab. So haben wir ein Duplikat. Das Dach-3 öffnen wir wieder und fügen es unverändert in die 3D-Ansicht ein. Wir sollten alle Kondokumente ausser **Dach-3-u** minimieren, um die Übersicht zu behalten. Ist das Dokument **Dach-3-u** offen bzw. aktiv setzen wir die Textur zurück.

Dazu klicken wir auf „Bearbeiten“, dann auf „Alle Texturen weg“ und bestätigen mit „Ja“.


Anschliessend geben wir allen Vertexen noch eine dunkelbraune Farbe und speichern mit .

Auf eine Textur verzichten wir zunächst einmal, da es hier nur um eine vereinfachte Variante geht.

Nun fügen wir das **Dach-3-u** ebenfalls in die 3D-Ansicht.



Wir werden hierbei zunächst keine Veränderung in der 3D-Ansicht feststellen. Um das **Dach-3-u** welches ja die dunkle Dachunterseite darstellen soll, sichtbar zu machen, müssen wir die folgenden Schritte ausführen.

Wir aktivieren das 3D-Fenster (mit Klick auf die obere Leiste oder irgendeine Stelle des Fensters) und klicken auf  „Besondere Einstellungen“ um in das Menü „Modelleinstellungen“ zu gelangen.



Hier machen wir die folgenden Einstellungen bei „Face Culling“

Im Rechten Fenster wählen wir „Dach-3“ und bei „Face Culling“ markieren wir „**Rückseite**“



Das gleiche wiederholen wir bei „Dach-3-u“ hier markieren wir jedoch „**Vorderseite**“



Damit blenden wir alle Seiten der Polygone aus, welche wir sowieso nie sehen, das heisst sie sind nur von einer Seite sichtbar. Man kann dies kontrollieren, indem man zuerst das untere Dach einfügt und die Vorderseite ausblendet.

Und so sieht das dann aus

à



#### Hinweis !

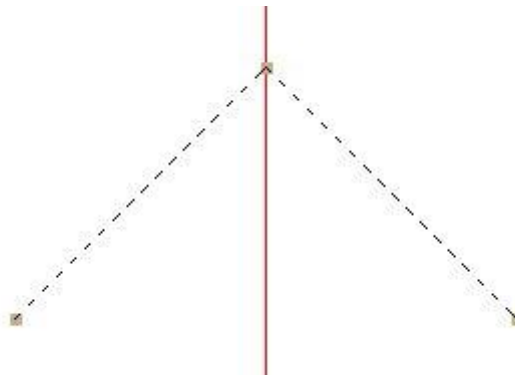
Modelleinstellungen in der 3D-Ansicht (Mod2-Datei) werden nicht in die Kondatei übernommen, also dort nicht gespeichert. Nimmt man eine Kon aus dem 3D-Fenster und fügt sie später wieder ein, so müssen die Einstellungen erneut vorgenommen werden.

Das war die „Sparvariante“ um das Dach etwas realistischer zu machen. Kommen wir zur etwas aufwendigeren Variante.

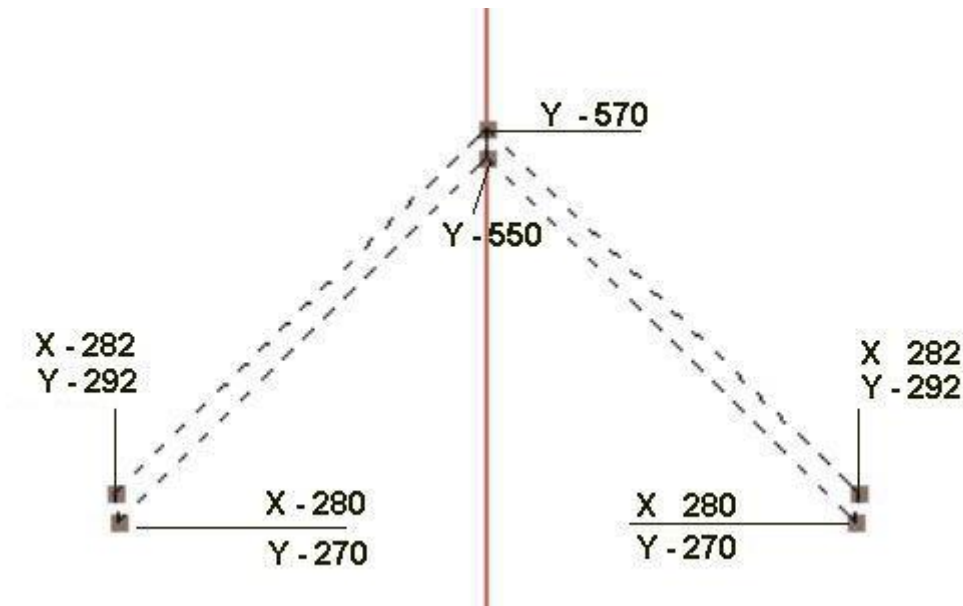
Hier werden wir das Dach mit der Ziegeltextur um 20cm über das „Unterdach“ heben, so daß hier eine realistische Dicke des Daches erreicht wird. Dabei wird die Blende an der Unterkante zu beiden Seiten des Daches sowie die Abdeckung an der Giebelseite und die Dachunterseite mit einer Holztextur versehen. Ausserdem werden die beiden Bretter der unteren Dachkanten in der gleichen Kon wie die Dachunterseite gezeichnet, also in der Kondatei „**Dach-3-u**“ .

Damit später alles Nahtlos zusammenpasst, kopiere ich zunächst das obere Dreieck aus der Datei „Giebelwand“ und lösche alle Linien, so daß nur ein unten offenes Dreieck übrigbleibt. (siehe Bild)

Anmerkung: die obere Spitze des Dreiecks muss die gleiche Position haben wie in der Datei „Giebelwand“.



Dann zeichne ich ein zweites Dreieck das um jeweils 20 cm über den bereits vorhandenen Vertexen liegen und verbinde auch die Vertexe durch senkrechte Linien. Siehe Bild unten.  
(die Zahlen geben jeweils die X – Y Koordinaten der Vertexe an)



Sind alle Vertexe und Linien verbunden, gebe ich den Vertexen noch eine mittelbraune Farbe und setze bei allen die Höhe (**Z**) auf – 530 (Halbe Länge des Daches).

Da wir dies auf beiden Seiten des Daches brauchen, legen wir noch eine zweite Instanz an.

In den Instanzen müssen wir lediglich bei Drehung (Y) folgende Werte eingeben:

Instanz Nr. 0 Drehung (Y) = 90, Instanz Nr. 1 Drehung (Y) = - 90

Wenn wir die Datei nun ins 3D Fenster einfügen muss sich folgendes Bild ergeben:

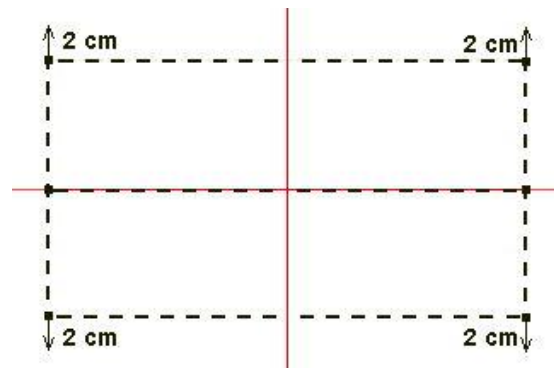
Wir sehen, daß das Giebelbrett bereits sauber an die Dachunterseite anschliesst.

Die Dachfläche mit der Ziegeltextur müssen wir nun noch um 20 cm heben und ausserdem auf jeder Seite in Y-Richtung um 2 cm verbreitern.

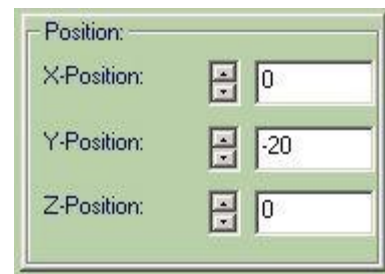


Wir öffnen bzw. maximieren die Datei „Dach-3“ und ziehen die Eckvertexe um jeweils 2 cm nach unten bzw. nach oben (siehe Maße Bild oben X 282 = Y 282)

(vereinfachte Darstellung)

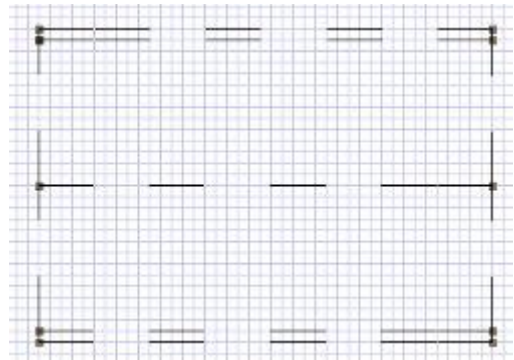


Anschliessend gehen wir mit  ins Instanzen Menü und geben dort für „Position“ bei Y-Position den Wert -20 ein.



Fügen wir nun das Dach wieder in das 3D-Fenster ein, so schliessen die seitlichen Kanten exakt mit den Oberkanten des Giebelbrettes ab. Nun fehlt nur noch das Abdeckbrett zwischen der unteren Kante des Ziegeldaches und des Unterdaches.

Dazu öffnen wir die Datei „**Dach-3-u**“ und fügen jeweils an den beiden Längskanten oben und unten ein 2 cm breites Polygon an. (Bild – verkleinerte Darstellung)  
Die 4 neu hinzugekommenen Vertexe bekommen Für die Höhe Z den Wert – 292  
(siehe Zeichnung am Seitenanfang)



Fügen wir nun die Datei „**Dach-3-u**“ erneut in das 3D-Fenster ein, müssten alle Polygone genau passen. (Bild)



Den letzten „Schliff“ bekommt das Ganze noch einer passend Bretter-Textur.

Ich nehme dafür die Textur-ID 965. Diese Textur stellt den Ausschnitt einer Wand aus verwitterten Brettern dar und ist für jeden frei verwendbar. Die Textur ist in X- und auch in Y-Richtung wiederholbar und eignet sich daher auch für grosse Flächen.

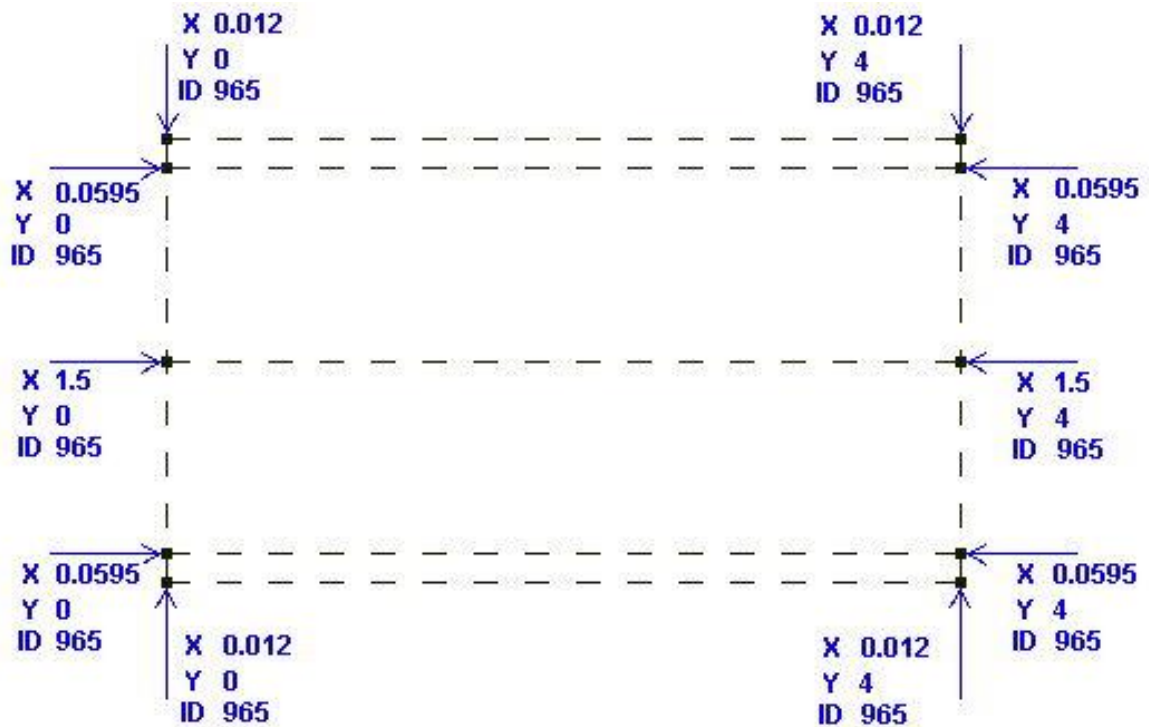
(Abb. in Originalgrösse)



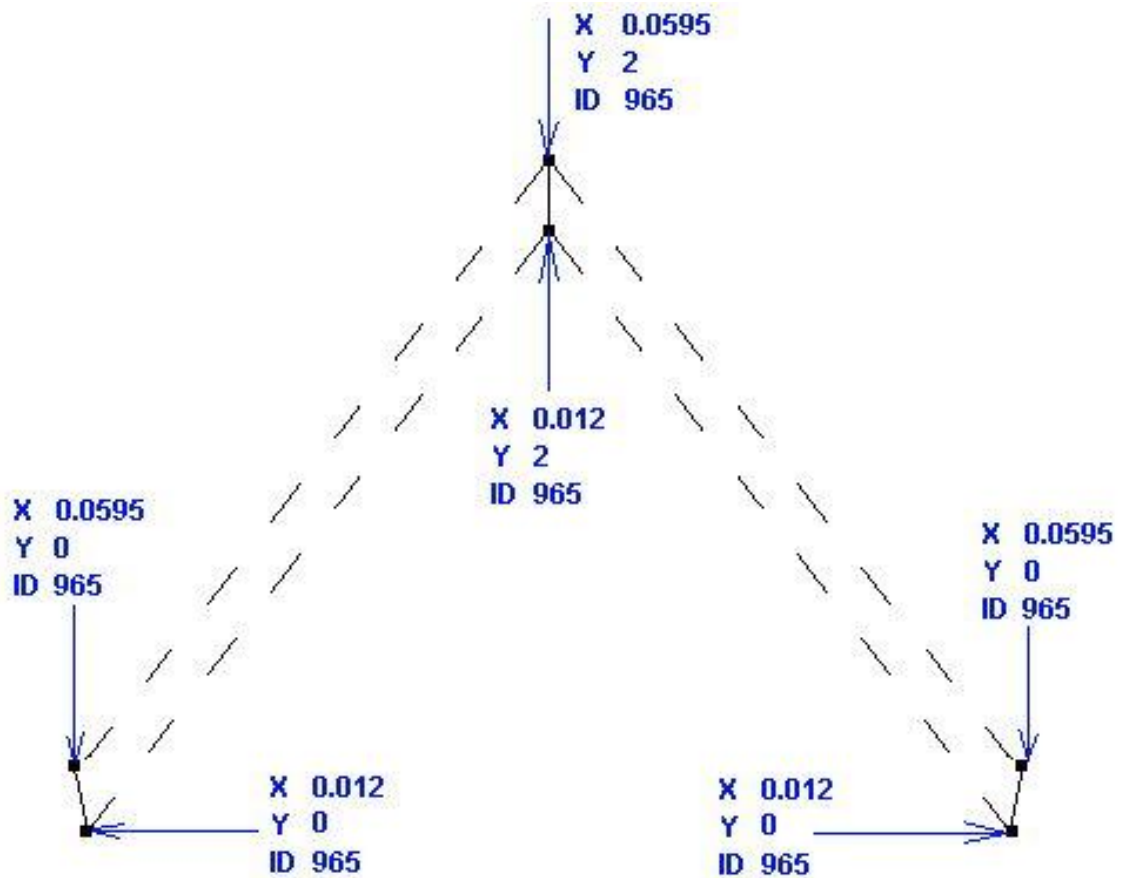
Die Textur soll auf der Dachunterseite längs zum Dach verlaufen und die Breite der dargestellten Bretter soll dabei einigermaßenrealistisch sein. Ausserdem soll auf den Polygonen welche die beiden Dachkanten bilden, nur ein Brett dargestellt werden.

Die entsprechenden Texturkoordinaten ermittle ich mit dem bereits erwähnten Tool EEPV. Um das Ganze hier etwas abzukürzen zeige die Werte für die einzelnen Vertexe an Hand einer einfachen schematischen Zeichnung. Ausserdem können die Werte sowie die von mir gewählten Farben in der mitgelieferten Kondatei überprüft werden.

Texturkoordinaten für die Datei „Dach-3-u“



Texturkoordinaten für die Datei „Giebelbrett“



Wenden wir uns nun den Schatten an den Hauswänden zu, welche ja durch das allseitig überstehende Dach auch in der Realität entstehen. Es muss uns allerdings klar sein, daß es kaum möglich ist solch einen Schatten zu erzeugen wie er an einem sonnigen Tag in der Realität durch Sonneneinstrahlung entsteht. Wir können nur einen „diffusen“ Schatten erzeugen, so wie er bei bedecktem Himmel auch in der Realität entsteht.

Wir haben ja bereits die schmalen Seiten unseres Hauses etwas dunkler gemacht, um zum einen eine bessere 3D-Wirkung zu erzielen und zum anderen schon einen Schatten anzudeuten.

Obwohl ich es mir angewöhnt habe den „Schatten“ bei meinen Modellen immer auf der gleichen Seite nämlich den Quer zur X-Achse stehenden Wänden bzw. Flächen zu setzen, so können hier trotzdem keine Rückschlüsse auf den Stand einer imaginären Sonne in EEP geschlossen werden, da nicht jedes Modell so stehen bleibt wie es eingestzt wird. Die meisten Immobilien werden ja doch mehr oder weniger gedreht. Trotz allem jedoch sollte man bei allen Modellen einen gewissen Schattenwurf berücksichtigen und anwenden. Dies wertet jedes Modell enorm auf und es wirkt realistischer.

Genug der Vorrede – machen wir uns an die Arbeit und beginnen mit den Seitenwänden.

Wir wollen einen Schatten unterhalb der überstehenden Dachkante. Dafür gibt es im Menü „Besondere Modelleinstellungen“ die Option „Smooth Shading“. Diese Funktion verechnet Farben und/oder Helligkeitswerte gegenüberliegender Vertexe miteinander und schafft einen weichen Übergang.

Würden wir nun nur die oberen beiden Vertexe dunkler machen und Smooth Shading aktivieren, so wäre unser Schatten kontinuierlich über die gesamte Höhe verrechnet. Unsere Hauswand würde einfach nur „schmutzig“ von oben her aussehen. (Bild)



Damit der Schatten realistisch wirkt, wenden wir hier einen Trick an. Wir unterteilen unsere Wand in zwei Polygone und zwar so, da das obere Polygon etwa  $\frac{1}{4}$  der Höhe hat.


Ein Viertel wären bei 300cm genau 75cm. Da der NOS keine ungeraden Zahlen akzeptiert und ich es mir bei der Berechnung der Textur leicht machen will, runde ich auf 80 cm auf.

Ich muss also bei Y –220m das Polygon waagrecht teilen.

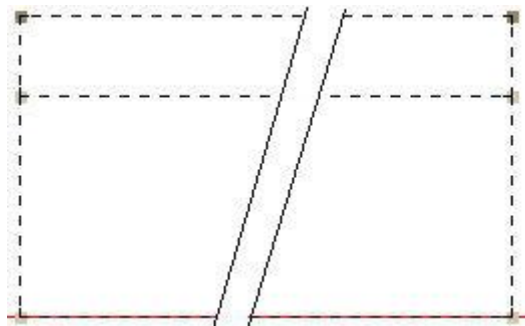
### Hinweis!

Wenn man eine vorhandene Linie nachträglich teilen möchte, so kann man dies nicht dadurch daß man ein Vertex auf die Linie setzt. Vertexe können nur am Linien Anfang bzw. Ende gesetzt werden. Wird ein Vertex auf eine beliebige Stelle einer Linie platziert, so wird dieser vom Programm ignoriert, eine Verbindung mit der Linie findet nicht statt.


Wir löschen also die beiden senkrechten Linien der **Wand-3** und ziehen neue Linien. Wir beginnen am linken uneren Vertex und ziehen die erste Linie bis Y – 220, setzen neu an und ziehen eine zweite Linie von Y –220 bis zum oberen linken Vertex Y –300. Wir

verbinden die Linien mit  und wiederholen das Ganze an der rechten Seite. Ansließend verbinden wir die beiden Vertexe waagrecht mit einer Linie.

(Bild – verkürzte Darstellung)



Nun übernehmen wir noch die Farbe eines der bereits farbigen Vertexe und geben im Menü „Verteigenschaften“ bei Höhe (Z) den Wert –250 bei den neu hinzugekommenen Vertexen ein.

Nochmal zur Erinnerung: mit  Klick auf einen Vertex gelangt man ins Menü „Eigenschaften von Veretex“

Und wenn wir uns schon im Vertexmenü befinden, so setzen wir auch gleich die Koordinaten der Texturen in den beiden neuen Vertexen.



Die Eintragungen dort sehen für die neuen Vertexe wie folgt aus:

Linker neuer Vertex		Rechter neuer Vertex	
Koordinaten:		Koordinaten:	
Horizontale (X):	-500	Horizontale (X):	500
Vertikale (Y):	-220	Vertikale (Y):	-220
Höhe (Z):	-250	Höhe (Z):	-250
Textur:		Textur:	
TexturX :	0	TexturX :	16.5
TexturY :	3.666	TexturY :	3.666
Textur-ID:	85	Textur-ID:	85

Wie ermittle ich die Textur-Koordinaten für die neuen Vertexe?

Wir erinnern uns – davon ausgehend, daß die Textur in der Realität einen Ausschnitt von 60x60cm darstellt und wir für das untere der beiden Polygone eine Höhe von 220 cm haben, ergeben sich die Koordinaten für  $Y = 220 : 60 = 3.666666.....$

Wir können diese Werte auf bis zu 3 Stellen abrunden, also 3.666

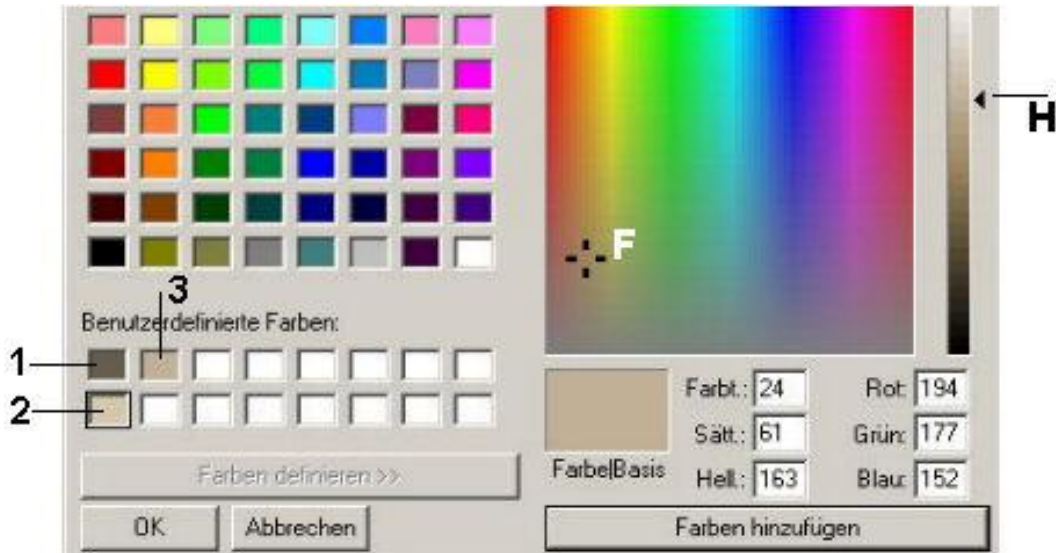
Die Werte für X sind mit denen des jeweils oberen bzw. unteren Vertex auf jeder Seite identisch, da sich ja in der Wiederholung der Textur in der X – Achse nichts ändert. Auch die Y Werte der oberen und unteren beiden Vertexe bleiben gleich, da sich ja die Gesamthöhe der Wand nicht ändert.

Noch ein Wort zu den Farben.

Solange ich im NOS arbeite lege ich eigene Farben in der Windows Farbpalette in den Feldern für „Benutzerdefinierte Farben“ ab. In unserem Beispiel habe ich drei Farben abgelegt:


Erläuterung zur Abbildung unten:

- 1 = Farbe der oberen beiden Vertexe
- 2 = Farbe der unteren beiden Vertexe
- 3 = Farbe der mittleren beiden Vertexe
- F = Auswahl für Farbtton und Sättigung
- H = Regler für Helligkeit



#### Hinweis !

In der Windows Farbpalette können bis zu 16 eigene Farben gespeichert werden.(Bild)  
Achtung!! Beim schliessen des Nostruktors gehen die gespeicherten Farben in den Feldern „Benutzerdefinierte Farben“ verloren. Wenn man die Arbeit im NOS unterbricht, also das Programm schliesst und später evtl. auf bereits mühsam ermittelte Farben zurückgreifen möchte, so empfiehlt es sich die Farbwerte zu notieren. Ansonsten besteht die Möglichkeit eigene Farben erneut über den Aufruf bereits mit diesen Farben belegter Vertexe in „Benutzerdefinierte Farben“ zu übertragen.

Um die Wirkung zu testen und um evtl. Korrekturen vorzunehmen, fügen wir die Wand wieder in das 3D-Fenster ein. Anschliessend gehen wir (bei aktiviertem 3D-Fenster) über  in das Menü „Besondere Einstellungen“.

Wir markieren „Wand-1“ im linken Fenster und „Alle“ im rechten Fenster, machen ein Häkchen bei Smooth Shading und markieren im Feld „Textur Applikation“ Blenden. Face Culling lassen wir erst mal unberücksichtigt.

Das Ergebnis ist sofort im 3D-Fenster sichtbar.



ohne Smooth Shading





mit Smooth Shading



Sie können nun selbst etwas mit den Schatten experimentieren. Wenn der Schatten etwas weiter nach unten gehen soll, so zieht man die Linie welche die beiden Polygone trennt mit den Vertexten etwas nach unten, dazu müssen jedoch die Texturkoordinaten neu berechnet werden.

Das gleiche wiederholen wir nun an der „Giebelwand“. Wir öffnen bzw. maximieren die Kondatei „Giebelwand“ und übertragen die Farbe welche wir für die oberen beiden Vertexe der „Wand-1“ verwendet haben, auf die oberen 3 Vertexe der Giebelwand, machen diese aber ein wenig dunkler, da ja die Wand insgesamt etwas dunkler werden soll wie die vordere Wand. Dann löschen wir die Waagerechte Trennlinie und die beiden senkrechten Aussenlinien.

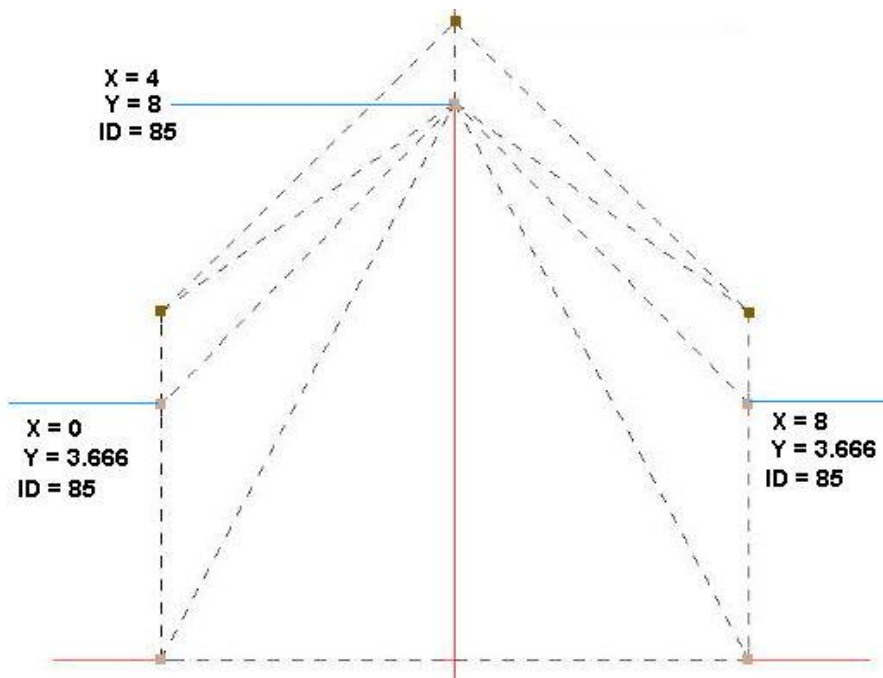
Als nächstes ziehen wir mit dem Linienwerkzeug  vom unteren linken Vertex ausgehend (X -250 / Y 0) eine senkrechte Linie nach oben bis Y -220, lassen die Maustaste los, setzen am gleichen Punkt nochmal an und ziehen die Linie weiter bis zum Vertex X -250 / Y -300.

Mit Klick auf  setzen wir einen Vertex und verbinden somit die Linie. Das gleiche machen wir auf der rechten Seite, wieder am unteren Vertex ansetzen, eine Linie bis Höhe Y -220, neu ansetzen und die Linie am oberen Vertex X 250 / Y -300 beenden.

Anschliessend ziehen wir noch eine Linie ausgehend vom oberen mittleren Vertex ( X 0 / Y -550) Eine kurze Senkrechte Linie bis X = / Y -480. Wir setzen wieder einen Vertex und verbinden nun die Vertexe durch Diagonale Linien wie in der folgenden Zeichnung zu sehen.

### Wichtig !

Linien müssen immer auf einem Vertex beginnen bzw. enden. Ansonsten findet keine Verbindung statt und das Polygon ist nicht geschlossen und kann auch nicht dargestellt werden.





Jetzt müssen wir noch den Farbwert eines der unteren Vertexe auf die 3 neuen übertragen ebenso die Höhe Z auf  $-500$  setzen.  
Zum Schluss noch in die neuen Vertexe die Texturkoordinaten wie in der Zeichnung oben angegeben eintragen.  
Nun überprüfen wir die Wirkung im 3D Fenster und korrigieren gegebenenfalls noch die Farb und /oder Helligkeitswerte.

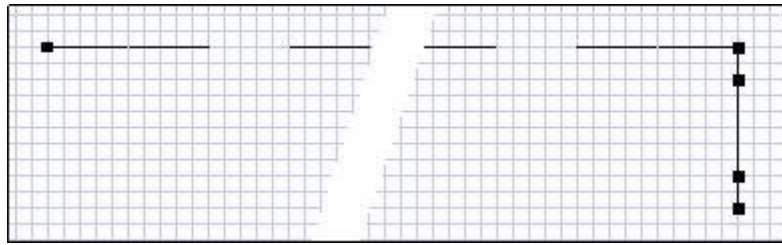
Experimentieren sie ruhig mal etwas mit der Breite der Schatten oder auch mit den Farb- und Helligkeitswerten, um ein Gefühl für die richtigen Einstellungen zu bekommen.



Wir sehen an diesem Beispiel wie man mit besonderen Einstellungen ein Modell nahezu realistisch, oder zumindest ansehnlich gestalten kann. Mit der Einstellung „Smooth Shading“ lassen sich jedoch auch noch andere erstaunlich Effekte erzielen.  
Dies zeige ich am Beispiel einer Dachrinne mit Fallrohr.



Wir erstellen dazu eine neue Kon-Datei und zeichnen 3 Zusammenhängende Polygone.  
Dabei berücksichtigen wir bereits die Maße des Daches, also Länge und Lage in der Z-Achse sowie die Höhe in Bezug auch die Dachkante über der Null-Linie. Somit ersparen wir uns umständliches verschieben und drehen später im 3D-Fenster.  
Wir ziehen eine Waagerechte Linie, unterhalb der Dokumentenmitte, beginnend bei den Koordinaten X  $-530$  Y 282 nach rechts bis Koordinaten X 530 Y 282 und setzen mit  Vertexe an den Linien-Enden. Wir bleiben am rechten Linienende und setzen mit  Vertexe auf die Koordinaten X 530 / Y 286, X 530 / Y 298, X 530 – Y 302 und verbinden diese mit Linien und zwar immer von Vertex zu Vertex. Siehe Bild nächste Seite.  
(Linienenden und Anfänge müssen immer auf einem Vertex liegen)

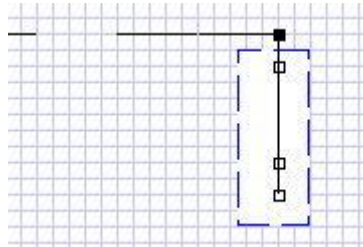
(Alle Abb. in verkürzter Darstellung)




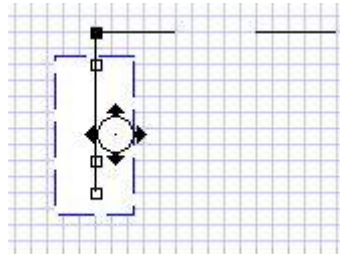
Jetzt geben wir im Menü  
Verteigenschaften die Höhen für  
Z ein wie in der nebenstehenden Skizze



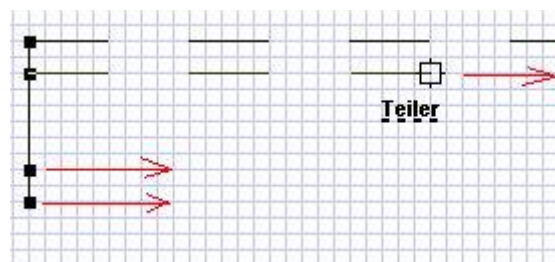
Mit  markieren wir die unteren 3 Vertexe  
und kopieren sie mit  in die  
Zwischenablage.



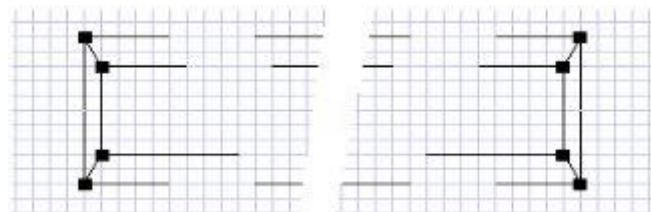
Wir verschieben die Ansicht bis zum Vertex  
am linken Linienend, fügen die  
Zwischenablage wieder ein und verschieben  
die Auswahl mit  daß sich das offene  
Linienende mit dem oberen Vertex  
verbindet.



In den Verteigenschaften des linken  
Oberen Vertex müssen wir noch die  
Höhe Z (-286) eintragen.  
Dann verbinden wir alle Vertexe  
durch waagerechte Linien, so daß 3  
schmale zusammenhängende Polygone  
entstehen.



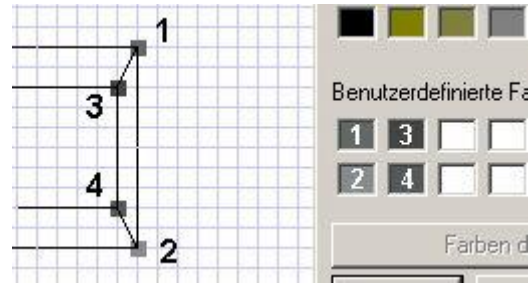
Zum Schluss verschieben wir die beiden  
Vertexe auf jeder Seite um je 2 cm nach  
innen, und verbinden die Äusseren beiden  
Vertexe auf jeder Seite mit je einer  
Senkrechten Linie. Somit haben wir eine Art  
Wanne deren Stirnseiten geschlossen sind.





Um mit Hilfe der Funktion „Smooth Shading“ die 3D Wirkung zu erzielen und die Dachrinne „runder“ erscheinen zu lassen, müssen wir die Polygone noch über die Vertexe einfärben.


Als Beispiel soll nebenstehende Skizze dienen. Dabei sind die Benutzerdefinierten Farben an Hand der Zahlen zuzuordnen. Dies gilt sowohl für die Vertexe der rechten als auch der linken Seite.



Selbstverständlich können sie auch eigene Farben zusammenstellen und verwenden.

Bevor wir nun die Dachrinne „anbauen“, gehen wir noch in das Menü „Instanzen“ und geben dort bei Drehungen **X –90** ein, erstellen eine neue Instanz, geben dort ebenfalls bei Drehung **X –90** ein und zusätzlich bei **Y 180**.

Jetzt fügen wir die Dachrinnen in das 3D-Fenster ein, aktivieren dieses und gehen

mit  in das Menü „Besondere Einstellungen“. Dort machen wir die Einstellungen wie auf dem Bild zu sehen.



Nun hat unser Haus auf beiden Seiten eine Dachrinne, die, obwohl eigentlich nur als eine Art eckige Wanne konstruiert ist, doch dank Smooth Shading rund wirkt.

Ich bezeichne das Smooth Shading als eine der wichtigsten Funktionen um Modelle nahezu realistisch zu gestalten.



Dies machen wir uns auch bei der folgenden Arbeit, den Fallrohren, zunutze. Um ein rundes Fallrohr (Regenablaufrohr) darzustellen, müssten wir eine Röhre aus unzähligen Polygonen machen, und sie wäre trotzdem noch nicht rund.

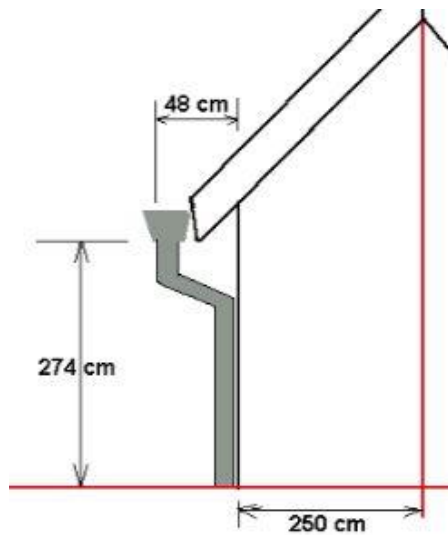
Mit Hilfe von Smooth Shading brauchen wir, um eine solche Röhre „darzustellen“ lediglich zwei Polygone. Da jedoch die Dachrinne etwas von der Hauswand absteht, das Fallrohr jedoch direkt auf der Hauswand verlaufen soll, brauchen wir noch einen Knick.

Bevor wir anfangen zu zeichnen, ermitteln wir die Position des Fallrohres im 3D-Fenster, die Höhe und Ausladung sowie den Durchmesser.

Ich zeige mal wie ich das mache. Ich mache mir eine grobe Skizze und notiere mir dort die entsprechenden Maße aus den Kondateien „Dachrinne“ und „Wand-1“



(schematische Darstellung)



Nun wäre es möglich unser Fallrohr von der Seite her zu zeichnen. Das jedoch würde bedeuten wir müssten dafür 4 Instanzen anlegen, damit auch von vorne gesehen das Rohr als Körper dargestellt wird.

Ich gehe da einen anderen Weg, nämlich das Ganze von vorne zu zeichnen und dabei die Rohrdicke (Z) und die Abknickung zu berücksichtigen.

Das klingt kompliziert, würde auch in einer Detaillierten Beschreibung kompliziert werden, weshalb es besser ist ich zeige dies in einer groben Skizze und gebe dort alle Maße für X, Y und Z vor.

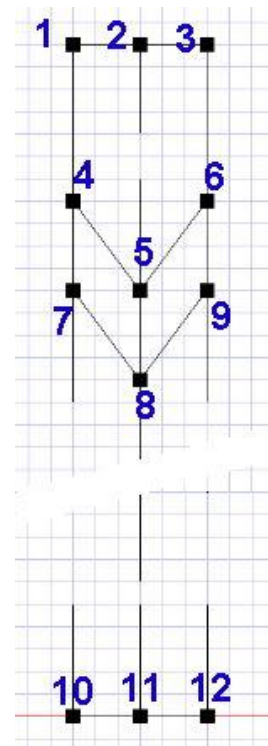
Ausserdem kann sich jeder zusätzlich die beigefügte fertige Kondatei ansehen.

Die nebenstehende Skizze zeigt die Zeichnung des Fallrohres im Konstruktionsdokument mit Zoomstufe 5 (etwas vergrößert) jedoch in der Höhe stark verkürzt.

Damit es ihnen leichter fällt die Position der einzelnen Vertexe im Dokument nachzuvollziehen, habe ich diese mit einer Nummer versehen und die entsprechenden Daten in der nachfolgenden Liste dokumentiert.

Dabei müssen die Z-Koordinaten wie wir ja bereits wissen, über den Vertex „Eigenschaften-Dialog“ eingegeben werden.

Nr.	Vertex Koordinaten			
1	X 458	Y -274	Z -288	
2	X 464	Y -274	Z -298	
3	X 470	Y -274	Z -288	
4	X 458	Y -258	Z -288	
5	X 464	Y -250	Z -298	
6	X 470	Y -258	Z -288	
7	X 458	Y -250	Z -252	
8	X 464	Y -242	Z -262	
9	X 470	Y -250	Z -252	
10	X 458	Y 0	Z -252	
11	X 464	Y 0	Z -262	
12	X 470	Y 0	Z -252	



Jetzt geben wir den Vertexen noch eine entsprechende Farbe und zwar die linke und rechte senkrechte Reihe mit einem sehr dunklen Grau (etwa RGB jeweils ca. 56) und die mittlere rechte Reihe mit einem Mittelgrau (etwa RGB jeweils ca. 130), und legen noch eine zweite Instanz an, in der lediglich für Drehungen bei Y der Wert 180 eingetragen wird.

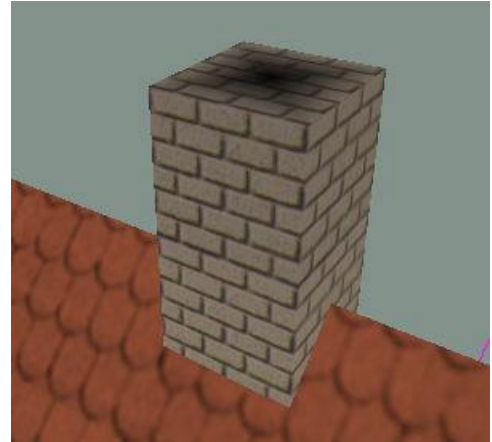
Haben wir das Fallrohr im 3D-Fenster, so fehlt noch die Aktivierung von Smooth Shading.

Und so müsste das dann aussehen -----à



Um das Ganze abzurunden und unser Modell für EEP fertig zu stellen, machen wir noch einen Kamin. Da sie jetzt schon etwas Übung haben, versuchen sie sich doch selbst mal. Der Kamin sollte Proportional schon zum Haus passen. Meine Vorschlag für die Grösse: Querschnitt 60x60 und eine Höhe von ca. 120 cm. Für die Textur die gleiche wie für die Hauswände, nur in einer anderen Farbe.

Wer sich dies noch nicht zutraut, kann auf die Dateien „Kamin-a“ und „Kamin-b“ zurückgreifen. Die Dateien finden sie im mitgelieferten Verzeichnis „Modelldateien\Haus-3“



Bevor wir nun weitere Funktionen des Nostruktors kennenlernen, wollen wir nun endlich mal unser bisheriges Werk in EEP begutachten.

Wenn wir so arbeiten wie ich es gewohnt bin, also während unserer Konstruktion die Y Achse im 3D-Fenster als die Höhe betrachten, müssen wir zunächst das Modell in der X-Achse um 90 Grad nach oben (hinten) drehen bevor wir den nächsten Schritt vornehmen.

(Ich gehe hierbei davon aus, daß wir unser Haus komplett im 3D-Fenster haben. Sollten sie die Arbeit unterbrochen haben, ohne die Mod2 zu speichern, so bauen sie das Haus nocheinmal zusammen. )

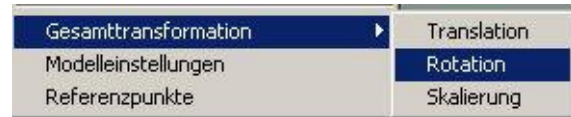
Steht unser Haus komplett im 3D-Fenster, so klicken wir mit der **rechten** Maustaste **in** das Fenster, so daß der rote Pfeil der die Z-Achse, also die tatsächliche Höhenachse darstellt, nach oben zeigt.

(Den Klick mit der rechten Maustaste ins 3D-Fenster kann man auch als Panikschalter bezeichnen. Immer dann wenn man durch zahlreiche Drehungen und hin und her zoomen mal nicht mehr weiß wo oben und unten ist, bringt man so die Ansicht wieder in die Grundstellung)

Das Haus liegt nun praktisch auf der Seite und wir müssen es in die richtige Lage bringen.

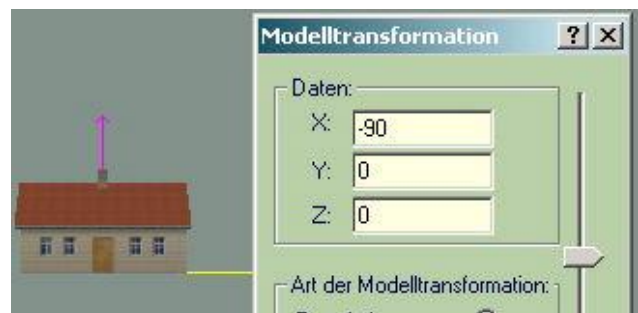
Dazu klicken wir in der Menüleiste oben links auf

**Bearbeiten** gehen zum Punkt „Gesamttransformation“ und klicken auf „Rotation“



Im darauffolgenden Fenster ist Standardmässig das Eingabefenster für die X-Achse markiert. Wir können nun mittels des Schiebereglers den Wert auf -90 setzen, was beim Home-Nos evtl. etwas ungenau ist. Geben sie den Wert deshalb lieber manuell ein.

Beim NosPlus arbeitet der Schieberegler in 5 Grad Schritten, was die Arbeit erleichtert.



Das Haus steht nun in der entgültigen Position aufrecht und mittig im 3D-Fenster.

### Hinweis !

Modelle wie Immobilien oder Gleisobjekte sollten möglichst mittig stehen, da EEP im Planfenster ein Markierungsrechteck zeichnet, welches bei Objekten die ausserhalb der Mitte stehen nicht die tatsächliche Grösse und/oder Position des Objektes in selbigem zeigen.

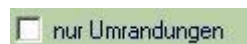
Jetzt können wir das Haus als Mod2 Datei speichern. Um Missverständnisse im weiteren Verlauf der Instruktionen zu vermeiden, wählen sie als dateinamen bitte „Haus-3“. Ich speichere meine Mod2 Dateien zunächst im gleichen Arbeitsverzeichnis wie meine Kondateien und kopiere sie später in die Ressourcen von EEP. Somit habe ich auch immer eine Sicherungskopie.

Damit EEP das Modell auch findet, muss eine \*.gsb Datei von unserem Modell erstellt werden. EEP sucht nicht nach der Mod2 sondern liest zunächst die Daten aus der \*.gsb. In dieser sind lediglich alle Daten erfasst die zum eigentlichen Modell, also der Mod2 führen.

Ein Modell kann auch aus mehreren Mod2 bestehen, nämlich dann wenn es über bewegliche Teile wie Türen usw. Verfügt. Diese Teile müssen getrennt vom eigentlichen Modell als eigenständige Mod2 erstellt und gespeichert werden. Auch diese Information ist in der gsb gespeichert. Dazu später mehr.

Da wir im Verlaufe unserer Arbeit immer wieder Kon-Dateien aus dem 3D-Fenster entfernt, bearbeitet und wieder eingefügt haben, sind auch alle Einstellungen wie Smooth Shading, Z-Offset usw. verloren gegangen und wir müssen diese nun wieder herstellen bevor wir uns an das Erstellen der GSB machen.

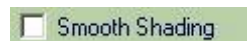
Erläuterung der Einstellungen im Menü „Modelleinstellungen“



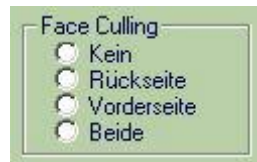
Wenn aktiviert werden die Polygone nur als Umriss angezeigt. Die Linienfarbe ist dabei von der jeweiligen Vertexfarbe abhängig. Diese Funktion ist u.U. hilfreich wenn eine Mod2 Datei bereits gespeichert und geschlossen wurde, jedoch für ein bestimmtes Polygon die Einstellungen geändert werden sollen, da in der Liste der Kondateien nach einem erneuten öffnen einer Mod2 an Stelle der Dateinamen Nummern treten.



Weißt das Programm an das damit markierte Polygon als letztes zu zeichnen. Bei Polygonen die sehr dicht aufeinander liegen kommt es bei zunehmender Entfernung zu Flimmereffekten. Z-Offset zwingt das Programm daß Polygone mit dieser Einstellung immer als das vordere erkannt werden.



Verechnet Farben und Helligkeitswerte von gegenüberliegenden Vertexen auf das gesamte Polygon fließend miteinander und sorgt so für sanfte Übergänge zwischen unterschiedlichen Helligkeits- bzw. Farbwerten



Blendet eine oder beide Seiten eines Polygons aus.

Sinnvoll ist die Ausblendung einer Seite eines Polygons, wenn diese nie sichtbar ist. Z.B. die Innenseiten eines Hauses, die Rückseiten von Fenstern und Türen wenn diese vor einem anderen Polygon (Wand) liegen.

In der Regel müssen alle Polygone vom Programm berechnet werden, auch die nicht sichtbaren Rück- bzw. Innenseiten. Werden diese Flächen ausgeblendet sind sie für das Programm nicht vorhanden und müssen somit auch nicht berechnet werden, was sich positiv auf die Performance auswirkt.



Ist hier Blenden gesetzt, so wirkt zusätzlich zur evtl. Farbe der Textur auch die Polygonfarbe.

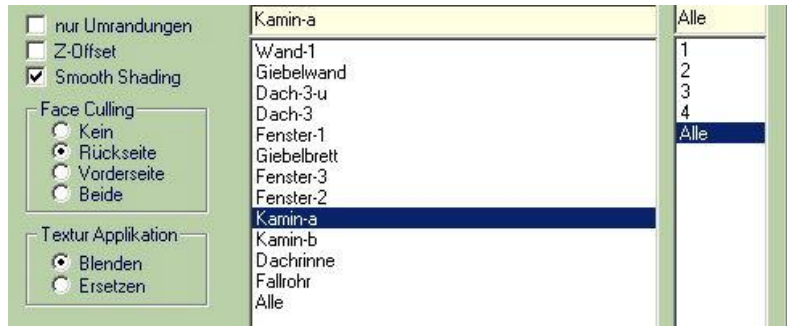
Ist Ersetzen gesetzt, so wird die Farbe des Polygons (Vertex-Farben) ignoriert. Nachteil – Die Polygone leuchten im Nachtmodus (ab EEP3).

Es sollte also Grundsätzlich darauf geachtet werden, daß bei allen Polygonen immer Blenden aktiviert ist.

In der Regel versuche ich schon beim Zeichnen von Polygonen diese so zu konstruieren bzw. in das 3D-Fenster zu setzen, daß ich beim setzen von Face Culling in der Regel bei allen „Rückseite“ ausblenden kann. So muss ich bei mehreren Instanzen nicht jede Instanz einzeln auswählen. Dies ist jedoch nicht immer möglich wie wir in der nachfolgenden Liste für unser Haus sehen werden.

Vorab am Beispiel der Datei „Kamin-a“ wie ich in Regel vorgehe die letzten Einstellungen eines Modells zu machen.

Datei im linken Fenster von Modelleinstellungen markieren, im rechten Fenster „Alle“ markieren, „Blenden“ aktivieren, Smooth Shading aktivieren, Face Culling „Rückseite“




Und hier die kompletten Einstellungen für unser Haus:  
Leere Klammer ( ) = nicht aktiviert bzw. Default)

Wand-1	(alle)	Z-Offset ( )	Smooth Shading (ein)	Face Culling (Rückseite)
Giebelwand	(alle)	Z-Offset ( )	Smooth Shading (ein)	Face Culling (Rückseite)
Dach-3-u	( )	Z-Offset ( )	Smooth Shading ( )	Face Culling (Vorderseite)
Dach-3	( )	Z-Offset ( )	Smooth Shading ( )	Face Culling (Rückseite)
Fenster-1	( )	Z-Offset (Rückseite)	Smooth Shading ( )	Face Culling (Rückseite)
Fenster-2	( )	Z-Offset (Rückseite)	Smooth Shading ( )	Face Culling (Rückseite)
Fenster-3	(alle)	Z-Offset (Rückseite)	Smooth Shading ( )	Face Culling (Rückseite)
Giebelbrett	( )	Z-Offset ( )	Smooth Shading ( )	Face Culling (Rückseite)
Kamin-a	(alle)	Z-Offset ( )	Smooth Shading (ein)	Face Culling (Rückseite)
Kamin-b	(alle)	Z-Offset ( )	Smooth Shading (ein)	Face Culling (Rückseite)
Dachrinne	(alle )	Z-Offset ( )	Smooth Shading (ein)	Face Culling (Kein)
Fallrohr	(alle )	Z-Offset ( )	Smooth Shading (ein)	Face Culling (Kein)

Wie bereits erwähnt sollte bei allen Polygonen immer „Blenden“ aktiviert sein.

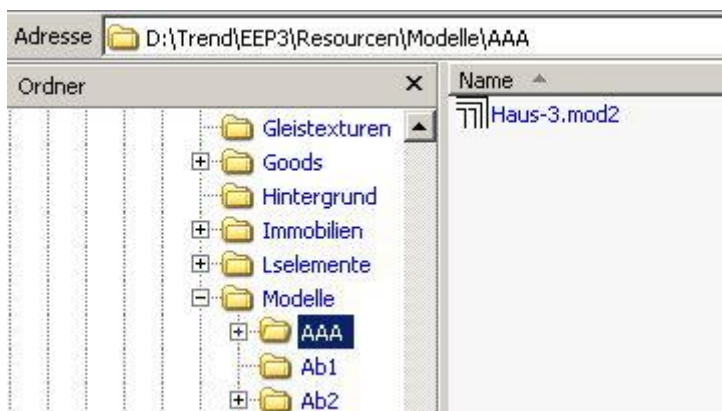
Bevor wir die Datei endgültig speichern, überzeugen wir uns ob alle Einstellungen richtig und alle Polygone die sichtbar sein sollen auch sichtbar sind. Dazu drehen wir die Ansicht im 3D-Fenster und betrachten das Modell von allen Seiten.

Wenn alles okay ist, dann speichern sie die Datei mit  endgültig ab. Sie können nun alle \*.kon sowie die Haus-3.mod2 schließen.



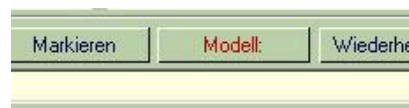
Um nun unser Modell für EEP aufzubereiten, müssen wir die Datei „**Haus-3.mod2**“ zunächst in den Ressourcenordner von EEP kopieren.

Jedoch nicht irgendwohin, sondern wir müssen uns einen eigenen Ordner unter .....\\Ressourcen\\Modelle\\.. erstellen. Dabei spielt es keine Rolle, welchen Namen wir für diesen Ordner verwenden. Theoretisch könnten sie auch einen der vorhandenen Ordner für ihre Modelle verwenden, was jedoch nicht Sinnvoll wäre. Ich nenne „Ihren“ Modell-Ordner für unser Beispiel einfach mal AAA

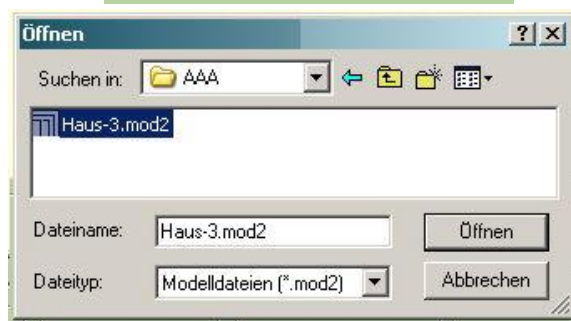


Ist dies erledigt, öffnen sie ein „**MoveModel**“ – Fenster und klicken dort auf 

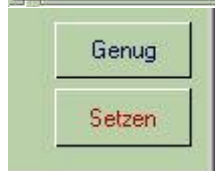
Im Menüfenster klicken wir auf den Button „Modell“ worauf ein Dateiauswahlfenster zur Verfügung steht.



Dort öffnen wir unseren Modellordner und markieren die Datei „Haus-3.mod2“ und klicken auf öffnen.



Anschliessend klicken wir auf „Setzen“ rechts unten und dann noch auf „Genug“, um das Fenster zu schliessen.



Das Haus müsste nun im MoveModel-Fenster aufrecht stehen.

Bevor wir ein Modell (\*.gsb) irgendwohin speichern, sollten wir uns überlegen in welche Kategorie dieses Modell am besten passt und wo wir es in EEP am ehesten suchen.

Da es sich hier um ein sehr kleines, eher ländlich anmutendes Modell handelt, würde es meiner Meinung am besten auch in einen Ordner mit ähnlichen Modellen passen.

Wir speichern die Datei also unter .....\\Immobilien\\Domizil\\Land als Testhaus-3 ab.

Sie können natürlich auch einen anderen Namen für dieses Modell verwenden.

### Hinweis !

Ab EEP 3 benötigen Modelle zusätzlich INI-Dateien. Hierbei handelt es sich um einfache Textdateien, welche mit der Endung \*.ini gespeichert werden. Diese Datei wird zwar beim nächsten Modell-Scan von EEP automatisch angelegt, jedoch enthält die Datei dann nur Standard-Daten welche die Suchfunktion stark einschränkt. Es ist zu empfehlen die \*.ini selbst anzulegen.

Wem die Suchfunktion als unwichtig erscheint, kann die \*.ini auch von EEP erstellen lassen.

Der **NosPlus** öffnet beim Speichern der Datei automatisch ein Fenster, in der die Benutzer- und Modellspezifischen Angaben ausgewählt und gespeichert werden können.



Nachfolgend das Beispiel einer korrekten mit dem NosPlus erstellten \*.ini für unser Modell

```
[FileInfo]
Name_Eng="Testhaus-3"
Name_Ger="Testhaus-3"
Name_Fra="Testhaus-3"
Name_Pol="Testhaus-3"
Icon=7
GENERATION=0
DEPLOYMENT_START=1900
DEPLOYMENT_END=2005
COUNTRY="D"
CREATOR_MODEL="MR1"
CREATOR_ORIGINAL="Others"
VERSION="1.0"
LANGUAGE="GER"
DESCRIPTION="Testhaus-3"
DESCRIPTION_ENG="Testhaus-3"
DESCRIPTION_GER="Testhaus-3"
DESCRIPTION_FRA="Testhaus-3"
DESCRIPTION_POL="Testhaus-3"
```

Zum Vergleich die \*.ini wie sie von EEP erstellt wird.

```
[FileInfo]
Name_ENG = "Testhaus-3"
Name_GER = "Testhaus-3"
Name_FRA = "Testhaus-3"
Name_POL = "Testhaus-3"
Icon = 7
GENERATION = 0
DEPLOYMENT_START = 1900
DEPLOYMENT_END = 1999
COUNTRY = "Germany"
CREATOR_MODEL = "Unknown"
CREATOR_ORIGINAL = "Unknown"
VERSION = "1.0"
LANGUAGE = "GER"
DESCRIPTION = "Non language specific description"
DESCRIPTION_ENG = "Description of model marked with language"
DESCRIPTION_GER = "Description of model marked with language"
DESCRIPTION_FRA = "Description of model marked with language"
DESCRIPTION_POL = "Description of model marked with language"
```

#### Hinweis !

Die \*.ini Datei muss den gleichen Namen wie die dazugehörige \*.gsb haben. In unserem Beispiel Testhaus-3.ini



# Kleine Texturhilfe

Hier will ich dir mal einige Grundkenntnisse der Texturenverwendung aufzeigen.

## Texturgrößen

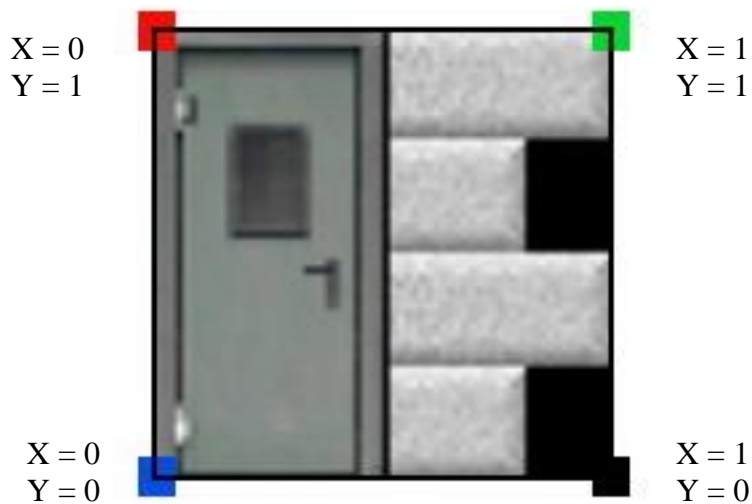
Texturen müssen eine bestimmte Grösse haben, z.B. **32x32**, **32x64**, **64x128**, **128x128**, usw. Als Minimum ist 16x16 anzunehmen wobei diese Grösse eher selten brauchbar ist. Als Maximum 256x256, wobei diese Grösse eher vermieden werden sollte. Theoretisch wäre eine Grösse von 512x512 möglich, darauf sollte aber ganz verzichtet werden.

## Texturen-Darstellung

Unter DirectX als auch unter OpenGL werden alle Farben sichtbar dargestellt ausser reines Schwarz, wenn alle 3 Farbwerte (Rot, Grün, Blau) den Wert Null aufweisen. Hat nur einer dieser Farbwerte einen höheren Wert als Null, also z.B. R=0, G=1, B=0, so wird diese Farbe sichtbar dargestellt. Reines Schwarz wird also „durchsichtig“. Diese Eigenschaft können wir uns zunutze machen.

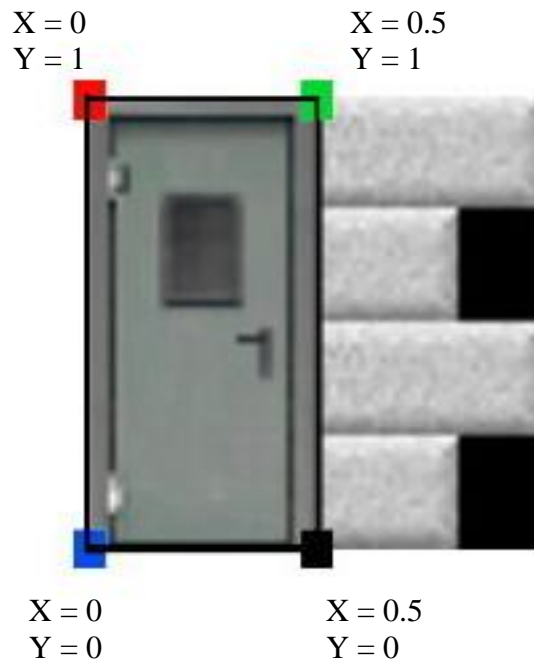
## Texturkoordinaten

Um eine Textur oder Teile daraus auf ein Polygon zu „kleben“, muss man zunächst die Koordinaten ermitteln. Will man eine Textur komplett in ein Polygon ist das einfach. Der Koordinaten Nullpunkt für die Y Richtung, also für die Senkrechte ist unten, der Nullpunkt für die X Richtung, also die Waagrechte ist links. Somit würden, bei Anwendung der ganzen Textur in einem Polygon, die Werte der einzelnen Vertexe wie folgt aussehen:



Wir möchten aber nur die Türe dieser textur verwenden. Dazu müssen wir uns ein Koordinatengitter über der Textur denken. Teilen wir also (in Gedanken) die Textur in 10 Koordinatenfelder auf. Dann wäre die Türe in Y Richtung die ganze Textur, also 100%. In der X Richtung aber nur ein teil davon. Bei dem Beispiel haben wir es einfach. Die Türe Belegt genau die Hälfte der Textur in X Richtung. Also 50%. Um diese Türe darzustellen muss unser Polygon erst mal das richtige Format aufweisen. Also nicht Quadratisch sondern Rechteckig.

Die Koordinatenwerte der Vertexe sind dann wie folgt:



Nun – das ist recht einfach. Wie aber ist das wenn ich aus einer Textur nur einen Kleinen Teil brauche? Ganz genauso.

Nehmen wir die gleiche Textur aus der wir diesmal nur den Mauerstein nehmen. Legen wir unser Polygon auf den teil der Textur den wir brauchen und betrachten es mal genauer.

Wir erkennen, daß der linke obere Vertex des Polygons genau im Mittelpunkt der Textur befindet, der linke untere Vertex in X Richtung ebenfalls in der Mitte und in Y Richtung bei einem viertel, also bei 25% der Gesamthöhe.

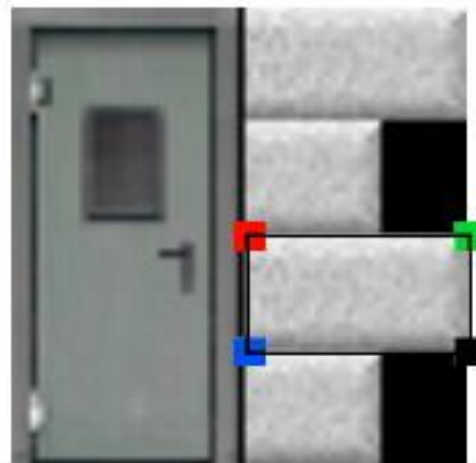
Daraus ergeben sich folgende Koordinaten:

Links oben  $X = 0.5$   
 $Y = 0.5$

Links unten  $X = 0.5$   
 $Y = 0.25$

Rechts oben  $X = 1$   
 $Y = 0.5$

Rechts unten  $X = 1$   
 $Y = 0.25$



Die Grösse des Steins wird dann natürlich über die Grösse des Polygons bestimmt.

## Kleine Anleitung zur Texturenerstellung

Ich zeige dir hier mal gaaanz grob wie ich vorgehe um z.B. aus einem Foto oder ähnlichem eine fertige Textur zu machen.

Ich habe mir erstmal eine Arbeitsfläche als neue Datei mit Namen „Arbeit.bmp“ erstellt. Auf diese übertrage ich zunächst mal das Objekt.



In diesem Fall nehme ich es aus einer bereits erstellten Textur. Nun überlege ich mir die entgültige Grösse der Textur. In diesem Falle 128x128. Ausserdem will ich hier zwei unterschiedliche Fensterarten einbringen. Da eine Wiederholung hier nicht notwendig ist, kann ich die Fenster beliebig positionieren.



Ich setze also die Fenster in der linken oberen ecke an und kann so auch die Grösse in der Statuszeile (links unten) ablesen. Um die Proportionen ohne Platzverlust in der Textur einigermaßen beizubehalten, wähle ich ähnliche Fenster. So kann ich von jeder Art 4 Fenster in dieser Textur unterbringen. Wenn mir aber von jeder Fensterart 2 verschiedene genügen, so könnte ich die Textur auch nur 64x128 machen. In diesem Falle aber wollen wir von jeder Sorte eine grössere Auswahl. Sollten wir dies nicht benötigen, könnte man später immer noch zwei oder mehr durch andere, ähnliche austauschen.

Sind die Fenster alle positioniert und ist die endgültige Grösse der Textur festgelegt, geht es an die Feinheiten.

Ich bearbeite einzelne Bereiche indem ich sie erst mal ausschneide und auf eine freie Stelle der Arbeitsfläche ziehe.



So kann ich z.B. die Fenstergläser, Rahmenteile usw. nachbearbeiten ohne erst mal das Original zu übermalen.



Wenn ich alle Scheiben bearbeitet habe, muss ich noch Helligkeit, Kontrast, Schärfe und Farbsättigung einstellen. Eine leichte Grundfärbung sollte bleiben, der Kontrast sollte nicht zu stark sein. Bei der Helligkeit kann man ruhig etwas höher gehen, da man über die Vertexe diese ja wieder dunkler machen kann. Auch leichte Farbnuancen lassen sich dort einstellen. Heller machen geht allerdings nicht mehr.

Am folgenden Beispiel zeige ich noch wie ich selber noch verschiedene Details ausarbeite. Z.B. eine zerbrochene Scheibe.



Ich schneide mir wieder eine der Scheiben aus und ziehe sie auf eine freie Stelle auf der Arbeitsfläche. dann ich nehme das Freihand-Linienwerkzeug und markiere damit die Stelle, an der ein Loch im Glas sein soll.





Dann nehme ich die Helligkeit für den markierten Bereich soweit zurück, bis der Eindruck entsteht hier ist ein Loch im Glas.



Da dies nun noch nicht so echt aussieht markiere ich die ganze Scheibe und bearbeite den markierten Bereich mit dem Werkzeug „verschwommen“. Hier muss man etwas experimentieren welche Stufe am besten wirkt. Und so sieht das fertige Ergebnis aus.



Möchte man andeuten, daß hinter dem zerbrochen Fenster eine Pappe steht, so muss das Glas erst entsprechend eingefärbt werden. Dabei ist dort wo noch Glas ist die Pappe dann etwas dunkler als da wo man die Pappe direkt sieht. Man kann also hier einen sehr realistischen Eindruck erzeugen. Siehe die Fenster weiter oben.

Hier nun alle Möglichkeiten zu erläutern wäre etwas übertrieben. Ich denke dies war ein kleiner Anstoss zum selber experimentieren.

# Texturen für EEP-Modelle richtig gemacht

Mit dieser kleinen Anleitung möchte ich dem Anfänger als auch dem schon etwas Fortgeschrittenen die Erstellung und den Umgang mit Texturen nahe bringen.

Es gibt für Modelle in EEP 2 Arten von Texturen; die \*.bmp welche ausschliesslich zum konstruieren im Nostruktur benötigt wird und die \*.dds welche von EEP benötigt wird.

## Das Erstellen von BMP Dateien

Texturen werden in einer durch 2 teilbaren Grösse erstellt. Dies beginnt (theoretisch) bei 1x1 Pixel bis 512x512 oder grösser. Die kleinste mir bekannte in EEP verwendete Grösse ist 32x32 Pixel. Es sind aber auch Grössen mit einem Seitenverhältnis von 2:1 oder in seltenen Fällen auch 4:1 möglich. Das optimale Seitenverhältnis ist jedoch immer 1:1 also ein Quadrat.

Eine Textur enthält in der Regel sichtbare und unsichtbare Bereiche. Sichtbare Bereiche sind die Teile welche im Modell dargestellt werden sollen, unsichtbare sind diejenigen welche nicht dargestellt werden sollen, also unsichtbar sein müssen. Sichtbare Bereiche können die Farbwerte für rot, grün und blau von 1 bis 255 aufweisen. Unsichtbare Bereiche müssen die Werte R0 G0 B0 haben, also reines schwarz.

Eine Textur kann jedoch auch vollflächige Elemente beinhalten also kein durchsichtigen Bereiche aufweisen. Beispiel der Ausschnitt einer Backsteinwand welcher dann auf einer Grösseren Fläche mehrfach dupliziert, also wiederholt wird.

## Nun zur Praxis

Ausgangsmaterial für eine Textur ist in der Regel ein Foto. Wer schon etwas Übung hat kann eine Textur aber auch von Grund auf selber zeichnen wenn es sich um eine einfaches Motiv handelt.

Zunächst mal eine einfache Textur welche den Ausschnitt eines Mauerwerks darstellt der vielfach wiederholt werden kann. Es dürfte einleuchten, daß wenn ich eine Mauer in der Grösse von z.B. 10 mal 10 meter machen möchte, nicht die komplette Mauer in einer Textur darstellen muss. Hier genügt ein Ausschnitt von wenigen Mauersteinen, welcher die Fläche komplett ausfüllt. Dieser kann dann beliebig oft wiederholt werden. Auch muss hierzu die Textur nicht übermässig gross sein, was Speicherplatz spart.

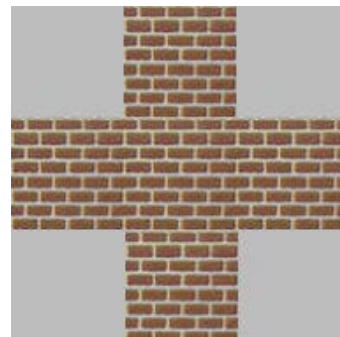
## Wiederholbare Texturen

Beispiel: Backsteinmauer mit hellen Fugen

Zunächst öffne ich eine Datei mit einem neutralen Hintergrund. Ich nehme dazu meist ein mittleres Grau. Nun schneide ich aus einem Foto eine Fläche die etwa 1x1 Meter in der Realität entspricht aus. Die Texturgrösse soll 64x64 Pixel betragen.



Da die Textur nach allen Seiten wiederholt werden soll, müssen nun alle 4 Seiten aneinander angepasst werden. Dies mache indem ich den Ausschnitt kopiere und auf meiner Arbeitsfläche eine Kopie an allen 4 Seiten anlege. Sind die Übergänge sauber kann ich die Textur fertigstellen indem ich den korrigierten Ausschnitt kopiere und in einer eigenen Datei speichere.



## Sammeltexturen

Als Sammeltexturen bezeichne ich Texturen in denen mehrere Elemente auf einer gemeinsamen Fläche arrangiert sind. Die Inhalte solcher Texturen sollten nach Möglichkeit immer Themenbezogen oder einem bestimmten Modelltyp zugeordnet sein. Immer gehts ja nicht **J**  
Beispiel, bestimmte Rollmaterialtypen, oder nur Fenster und Türen, oder Teile eines grösseren Objektes bzw. eines Themas. (Beispiel: Serie Wiesental)

Für die Erstellung von Texturen habe ich mir eine „Arbeitsdatei“ in der Grösse 512x512 Pixel gemacht. Als Hintergrund kann ich einen jeweils passenden Farbton wählen, so daß sich die Inhalte gut davon abheben.

Die Vorgehensweise für das eigentliche Erstellen der Inhalte muss ich hier nicht extra behandeln. Hier weichen auch die unterschiedlichen Malprogramme mit ihren Werkzeugen und Möglichkeiten oft stark voneinander ab.

## Fertigstellen der Texturen

Als Beispiel für die weiteren Schritte nehme ich meine Textur „Ladegut“.

Habe ich alle Inhalte der Textur auf meinem „Arbeitsblatt“ positioniert und dadurch auch die entgültige Grösse festgelegt (im Beispiel 256x256 Pixel) schneide ich diese aus und erstelle damit eine neue Datei. Diese Textur hat einen den Inhalten angepassten Hintergrund. Diese Datei wird im Prinzip nur zur Erstellung der DDS verwendet, weshalb ich diese im Dateinamen besonders kennzeichne.

Da ich grundsätzlich als Dateiname die Textur-ID verwende, würde diese Datei z.B. so heißen: **2398-w.bmp**

Das **-w** sagt mir später daß ich diese Datei zur Erstellung Der DDS verwenden muss.



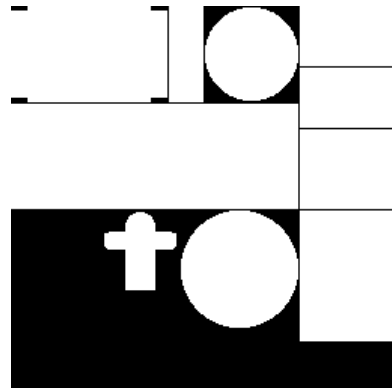
Damit ich diese Textur im Nostrukturor richtig verwenden kann, muss ich noch den Hintergrund, also alles was nicht sichtbar sein darf, mit den Farbwerten R0 G0 B0 (schwarz) versehen. So ist es z.B. möglich ein Rundes Element in ein rechteckiges Polygon einzufügen und trotzdem ist später nur das runde Element sichtbar. (Im Beispiel die Fassböden). Mit dem Malprogramm das ich verwende geht das sehr einfach. Ich nehme das werkzeug „Zauberstab“ stelle den Farbwert auf >0< und markiere so durch einfachen Klick alle Flächen die später durchsichtig, also schwarz sein müssen. Die markierten Flächen fülle mit schwarz (R0 G0 B0) Diese Datei speichere in diesem Beispiel als **2398.bmp**



Nun muss noch eine SW-Schablone, der sogenannte AlphaCannel erstellt werden. Dieser legt aufs Pixel genau Fest wo die Grenzen der darzustellenden Elemente sind. Gleichzeitig eliminiert er in der DDS den farbigen Hintergrund. Wenn die schwarzen Flächen in der vorhergehenden Datei Noch markiert sind, dann kann ich mit der Funktion „Markierung umkehren“ mit einem Klick die Darzustellenden Inhalte markieren und anschliessend mit weiß füllen.

Diese Datei speichere ich unter **2398\_a.bmp**

Unterstrich \_a ist wichtig, damit DxTex.exe diese Datei als AlphaChannel erkennt.



Das alles liest sich vielleicht etwas lang und umständlich. Aber diese Schritte sind mit einem dafür geeigneten Malprogramm in Minutenschnelle erledigt. Mehr Aufwand ist das eigentliche „malen“.

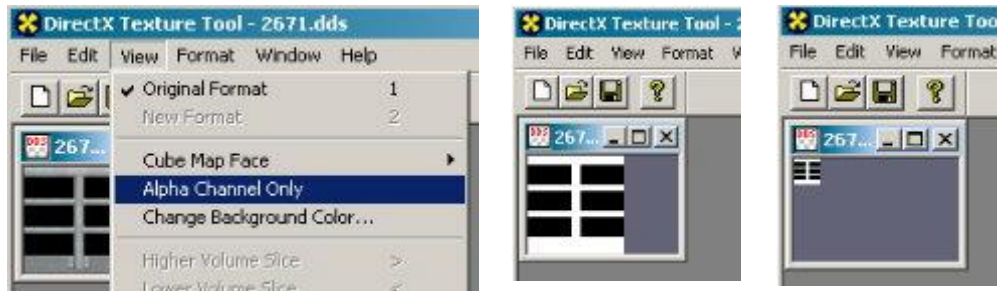
# Erstellen einer \*.DDS Datei

Soll eine vorhandene DDS durch eine Neue ersetzt werden sind zunächst folgende Schritte notwendig:

## Wichtig !

Zunächst muss überprüft werden ob das Original über **Alphachannel** verfügt, mit welcher Kompressionsstufe die Datei **komprimiert** ist und ob sie **MipMapLevel** enthält.

Die Original DDS in DxTex.exe öffnen und im Menü „**View**“ den AlphaChannel sichtbar machen.



Anschließend überprüfen ob **MipMapLevel** vorhanden sind. Dies geschieht mit der „Bild auf“ und „Bild ab“ Taste.

Anschließend die Kompressionsrate ermitteln. Dazu geht man ins Menü „**Format**“ auf die Option „**ChangeSurface Format**“.



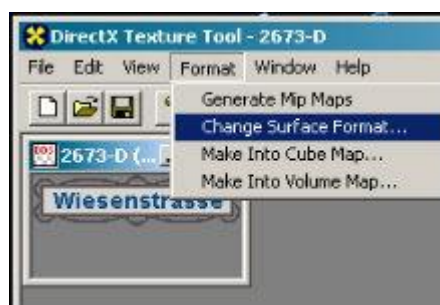
Die so ermittelten **Werte und Einstellungen** notiert man sich. Diese sind wichtig für eine korrekte Darstellung der Texturen in den jeweiligen Modellen.

Am besten sichert man die Original DDS durch umbenennen oder verschieben in einen anderen Ordner

## Das Erstellen einer neuen DDS

In DxTex die neu erstellte oder geänderte \*.BMP laden. Wenn vorhanden nimmt man hier die BMP mit dem farblich angepassten Hintergrund.

Im Beispiel ist es die 2673-D.bmp. Ich verwende für den Dateinamen meiner Texturen die ID Nummer und um die bmp's zu unterscheiden erhält die angepasste einen Buchstaben hinter der ID.



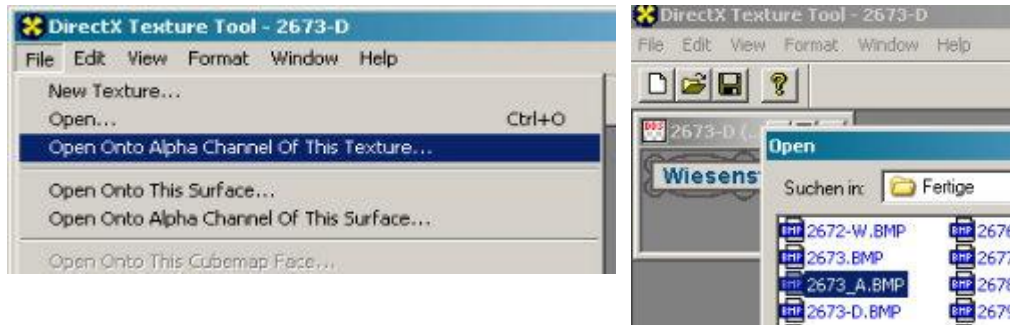
Nun geht man wieder in das Menü „**Format**“ und gibt dort den Komprimierungsfaktor bei „**Change Surface Format**“ ein. Für eine bereits bestehende DDS den dort notierten.



Anschliessend das Menü „**File**“ aufrufen und „**Open Onto Alpha Channel of this Textur....**“ aufrufen.

Im DateiBrowser sucht man nun die BMP mit dem AlphaChannel, in diesem Falle die Datei „2573\_a.bmp“, markiert diese und fügt sie durch OKAY ein.

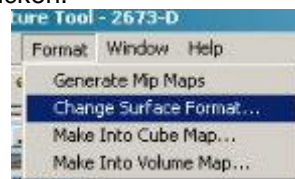
**Wichtig!** Bei der BMP mit dem AlphaChannel muss hinter dem Dateinamen unbedingt >(Unterstrich)\_a< stehen. Ansonsten wird die Datei nicht akzeptiert.



Durch den Alpha Channel wird der angepasste Hintergrund wieder neutralisiert, das heisst er hat wieder die Farbwerte R0 G0 B0.



Anschliessend geht man wieder in das Menü „**Format**“ und erstellt dort die **MipMapLevel**. Auf „**Generate Mip Maps**“ klicken.



Als letzter Schritt muss die neue DDS gespeichert werden. Dabei muss unbedingt der Dateiname der Originaldatei verwendet werden wenn es sich um eine Änderung handelt. Bei einer neu zu erstellenden DDS, was aber für den Konstrukteur von Bedeutung ist, kann auch ein beliebiger Name verwendet werden. Die Kennung der ID erfolgt in der Texturen.txt – Es hat sich aber bewährt, für den dateinamen gleich die Textur-ID zu verwenden.





# Ändern von Texturen für eigene Bedürfnisse

## Vorwort.

Texturen für Modelle in EEP lassen sich unter bestimmten Voraussetzungen für eigene Bedürfnisse ändern. Insbesondere wenn Texte wie Schilderbeschriftungen etc. angepasst werden sollen.

Da Texturen einem Copyright unterliegen und ohne die Zustimmung des Autors nicht verändert werden dürfen, wäre es sinnvoll vor einer solchen Änderung die Zustimmung des Autors einzuholen.

**Wichtig !** geänderte Texturen dürfen nicht in Umlauf gebracht werden

## Was braucht man um eine Textur zu ändern.

Ein Mal- oder Grafikprogramm das Pixelbearbeitung zulässt, die entsprechende \*. bmp Datei im Windows Bitmap Format (nicht OS2 BMP) der Textur die zu ändern ist und, falls die DDS einen Alpha Channel beinhaltet, was meistens der Fall ist, auch die \*\_a.bmp, also den AlphaChannel. Der AlphaChannel ist eine SW Schablone der darzustellenden Inhalte einer Textur.

**Achtung!** Einige Konstrukteure verwenden zur Erzeugung der DDS Datei eine BMP bei welcher der Hintergrund farblich den Inhalten angepasst wurde. (Dies verhindert schwarze Ränder an bestimmten Objekten). Dieser Hintergrund wird durch den AlphaChannel in der DDS wieder eliminiert. Wurde eine solche BMP zur Erstellung der DDS verwendet, so ist diese für die Änderung zu verwenden.

**Anmerkung:** ich rate dringend davon ab die \*.bmp aus einer \*.DDS zu konvertieren. Eine damit erstellte neue DDS entspricht u.U.nicht mehr dem Original.

Hier nun ein praktisches Beispiel an Hand einer von mir erstellten Textur (Ausschnitt):



Hinweis: „Name“ steht für den Dateinamen der Textur.

„Name . bmp “

Diese Datei findet ausschliesslich im Nostruktur Verwendung, da hier der Hintergrund (Farbwerte R0 G0 B0) nicht dargestellt wird.



„Name-d . bmp “

Diese Datei dient der Erstellung der DDS. Der Hintergrund wurde farblich den darzustellenden Elementen angepasst um evtl. auftretende schwarze Ränder zu eliminieren.

Hier kann nun eine entsprechende Änderung vorgenommen werden. Dabei darf aber weder die Grösse noch die Form des Objektes geändert werden.



„Name\_a . bmp “

Diese Datei stellt den Alphachannel dar. Die Umriss der weissen Schablone entsprechen auf das Pixel genau den Umrissen des im Modell darzustellenden Inhaltes der Textur. Diese „Schablone“ zwingt das Programm nur die innerhalb der weissen Fläche befindlichen Teile (Pixel) darzustellen.

Wie muss man nun eine solche Änderung bewerkstelligen.

Zunächst wird das Feld in dem sich der Text befindet mit dem Texthintergrund übermalt. Nun wird ein neuer Text eingefügt und entsprechend positioniert. Evtl. muss Pixel für Pixel nachgearbeitet werden. Die Länge des neuen Textes sollte dabei die des ursprünglichen Textes nicht wesentlich unter- bzw. überschreiten. Dies ist auch zu beachten, wenn der Text frei, also vor dem schwarzen Hintergrund steht, da sich die Koordinaten im Polygon der entsprechenden Modelldatei nicht ändern lassen.

Auch die Grösse der Textur darf nicht verändert werden.

Eine Änderung der Gesamtgrösse des Textes ist nur durch Neuerstellung einer Textur und dem Neuaufbau des Modells möglich.

War die Änderung erfolgreich, muss nun die DDS mit Hilfe von „DxTex.exe“ neu erstellt werden.

**Wichtig !** Enthielt die Originaltextur **MipMap Level**, so müssen diese wieder hergestellt werden.

Enthielt die Originaltextur einen **Alphachannel**, muss auch dieser in der DDS eingebunden werden.

War die DDS in einer der möglichen Stufen komprimiert, so muss diese auch in der neuerstellten DDS wieder verwendet werden. Näheres dazu ist in DxTex.exe ersichtlich.

**Von Original Texturen vor jeder Änderung ein Duplikat anfertigen.** (Falls was schief geht)

Wünsche gutes Gelingen.