

## XFORM SkalierungX, NeigungY\_x, NeigungZ\_x, VerschiebungX, NeigungX\_y, SkalierungY, NeigungZ\_y, VerschiebungY, NeigungX\_z, NeigungY\_z, SkalierungZ, VerschiebungZ

### Erläuterung

Definiert eine komplette Transformationsmatrix mit folgender Abbildungsvorschrift:

$$x' = \text{SkalierungX} \cdot x + \text{NeigungY\_x} \cdot y + \text{NeigungZ\_x} \cdot z + \text{VerschiebungX}$$

$$y' = \text{NeigungX\_y} \cdot x + \text{SkalierungY} \cdot y + \text{NeigungZ\_y} \cdot z + \text{VerschiebungY}$$

$$z' = \text{NeigungX\_z} \cdot x + \text{NeigungY\_z} \cdot y + \text{SkalierungZ} \cdot z + \text{VerschiebungZ}$$

Findet fast ausschließlich Verwendung bei der automatischen Generierung des GDL Codes. Der Befehl ist eigentlich vollständig mit ROT, MUL und ADD-Befehlen zu ersetzen. Einzig durch den Befehl XFORM läßt sich der 3D-Raum verzerren, sodaß die Achsen X,Y und Z nicht mehr orthogonal zueinander stehen.

Für einen angenommenen Neigungswinkel der X-Achse von WinkelX lautet der Befehl:

$$tx = \tan(\text{WinkelX})$$

$$\text{XFORM } 1,0,0,0, 0,1,0,0, tx,0,1,0$$

Hier die Transformationsmatrix (siehe Tipps+Tricks) als Grundlage für den Befehl:

$$1 \ 0 \ tx \ 0$$

$$0 \ 1 \ 0 \ 0$$

$$0 \ 0 \ 1 \ 0$$

$$0 \ 0 \ 0 \ 1$$

Für die Neigung der Y-Achse um WinkelY mit

$$ty = \tan(\text{WinkelY})$$

lautet die Transformationsmatrix:

$$1 \ ty \ 0 \ 0$$

$$0 \ 1 \ 0 \ 0$$

$$0 \ 0 \ 1 \ 0$$

$$0 \ 0 \ 0 \ 1$$

Für die Neigung der Z-Achse um WinkelZ mit

$$tz = \tan(\text{WinkelZ})$$

lautet die Transformationsmatrix:

$$1 \ 0 \ 0 \ 0$$

$$0 \ 1 \ 0 \ 0$$

$$tz \ 0 \ 1 \ 0$$

$$0 \ 0 \ 0 \ 1$$

Ein sinnvoller Einsatz erschließt sich nicht sofort. Eine Möglichkeit ist ein schnelles Schrägstellen horizontaler Elemente, wie James Willmot im GDL-Talk gepostet hat: **XFORM 1,0,0,0, 0,1,0,0,tan(Neigungswinkel), 0,1,0**  
Dieser Befehl am Beginn des 3D-skriptes des Objektes "Geländer\_horizontal 70.GSM" (aus der Bibliothek 7.0: 1. Basisbibliothek/ 1.04 Bauelemente/ Zäune und Geländer) neigt dieses um den angegebenen Winkel. Zwar ohne korrekte Wiedergabe der Profilquerschnitte, dafür aber eine sehr einfache Programmierung.  
So können auch einfach trapezförmige Fenster erzeugt werden.

### Tipps und Tricks

Die Transformation von Vektoren kann mit der algebraischen Matrizenrechnung abgebildet werden. Dabei wird bei der Verwendung homogener Koordinaten folgende Matrize verwendet:

$$1 \ 0 \ 0 \ 0$$

$$0 \ 1 \ 0 \ 0$$

$$0 \ 0 \ 1 \ 0$$

$$0 \ 0 \ 0 \ 1$$

Zur Übertragung der Matrix werden die Spalten v.l.n.r in die Zeilen des Befehls übertragen. Die letzte Spalte wird weggelassen. so finden sich die Parameter des Befehls an folgenden Stellen in der Matrix:

$$\text{SkalierungX} \ \text{NeigungX\_y} \ \text{NeigungX\_z} \ 0$$

$$\text{NeigungY\_x} \ \text{SkalierungY} \ \text{NeigungY\_z} \ 0$$

$$\text{NeigungZ\_x} \ \text{NeigungZ\_y} \ \text{SkalierungZ} \ 0$$

$$\text{VerschiebungX} \ \text{VerschiebungY} \ \text{VerschiebungZ} \ 1$$

Die Umsetzung der herkömmlichen Transformationsbefehle im Einzelnen:

#### ADD - Verschiebungen

Der Translationsanteil des XFORM Befehls findet sich in der untersten Zeile der Matrix. Der Verschiebungsvektor dx/dy/dz eines adäquaten Befehls ADD dx,dy,dz findet sich in der Matrix wie folgt:

```
1 0 0 0
0 1 0 0
0 0 1 0
dx dy dz 1
```

Der Befehl lautet:

```
XFORM 1,0,0,dx, 0,1,0,dy, 0,0,1,dz
```

#### MUL - Skalierung

Die Skalierungsfaktoren finden sich auf der Diagonalen der Matrix von links oben nach rechts unten. Die Faktoren fx/fy/fz eines adäquaten Befehls MUL fx,fy,fz findet sich in der Matrix wie folgt:

```
fx 0 0 0
0 fy 0 0
0 0 fz 0
0 0 0 1
```

Der Befehl lautet:

```
XFORM fx,0,0,0, 0,fy,0,0, 0,0,fz,0
```

#### ROT - Rotationen

Die Matrix für eine Rotation ist schon etwas schwieriger, weil sie in ihre achsialen Komponenten aufgespalten werden muß. Drehungen um alle drei Achsen sind daher auch nicht so einfach wie bei der Verschiebung und der Skalierung anzugeben. In der Folge nun die 3 Matrizen für die 3 Achsen.

Der übersichtlichen Darstellung halber werden in den Matrizen die Variablen verwendet, die zuvor aus dem Rotationswinkel errechnet werden:

#### ROTX - Drehung um die X-Achse

Äquivalente Matrix zum Rotationsbefehl ROTX WinkelX:

```
sx = sin(WinkelX)
cx = cos(WinkelX)
1 0 0 0
0 cx sx 0
0 -sx cx 0
0 0 0 1
```

Der Befehl lautet:

```
XFORM 1,0,0,0, 0,cx,-sx,0, 0,sx,cx,0
```

#### ROTY - Drehung um die Y-Achse

Äquivalente Matrix zum Rotationsbefehl ROTY WinkelY:

```
sy = sin(WinkelY)
cy = cos(WinkelY)
cy 0 -sy 0
0 1 0 0
sy 0 cy 0
0 0 0 1
```

Der Befehl lautet:

```
XFORM cy,0,sy,0, 0,1,0,0, -sy,0,cy,0
```

#### ROTZ - Drehung um die Z-Achse

Äquivalente Matrix zum Rotationsbefehl ROTZ WinkelZ:

```
sz = sin(WinkelZ)
cz = cos(WinkelZ)
cz sz 0 0
-sz cz 0 0
0 0 1 0
0 0 0 1
```

Der Befehl lautet:

```
XFORM cz,-sz,0,0, sz,cz,0,0, 0,0,1,0
```

Mehrere Transformationen lassen sich beliebig zu einer kombinieren, wenn man die Matrizen miteinander multipliziert.

Das ist jedoch nicht ganz so einfach, denn jetzt wird etwas analytische Geometrie notwendig (siehe "Links").

Zur Erläuterung kehren wir zu der im GDL-Handbuch verwendeten Notation der Parameter zurück, sonst wird es unübersichtlich.

Unser Befehl lautet:

```
XFORM a11 , a12 , a13 , a14 ,
a21 , a22 , a23 , a24 ,
a31 , a32 , a33 , a34
```

Schreibt man nun die Parameter in die Matrix ergibt sich als erste Transformation **A**:

```
a11 a21 a31 0
a12 a22 a32 0
a13 a23 a33 0
a14 a24 a34 1
```

Eine zweite, nennen wir sie **B**, sieht so aus:

```
b11 b21 b31 0
b12 b22 b32 0
b13 b23 b33 0
b14 b24 b34 1
```

Die Zielmatrix, nennen wir diese **C**, die beide Transformationen enthält ist das Produkt aus **AxB** und hat die gleichen Abmessungen, also 4x4 Stellen.

```
c11 c21 c31 c41
c12 c22 c32 c42
c13 c23 c33 c43
c14 c24 c34 c44
```

Jede dieser Stellen berechnet sich aus dem Skalarprodukt der zugehörigen Zeile aus A mit der zugehörigen Spalte aus B.

Ein Beispiel:

$$c32 = (a12 \ a22 \ a32 \ a42) \cdot (b31 \ b32 \ b33 \ b34)$$

Also zweite Zeile aus A mit dritter Spalte aus B.

Das entspricht ausmultipliziert:

$$c32 = a12 \ b31 + a22 \ b32 + a32 \ b33 + a42 \ b34$$

Da die Stellen  $a41$  ,  $a42$  ,  $a43$  ,  $b41$  ,  $b42$  und  $b43$  immer 0 und  $a44$  und  $b44$  immer 1 sind, gilt gleiches für die letzte Spalte der Zielmatrix analog.

Berücksichtigt man diese Besonderheit ergibt sich folgender, resultierender Befehl für die beiden Transformationen **A** und **B**:

```
XFORM a11 * b11 + a21 * b12 + a31 * b13 , != c11
a12 * b11 + a22 * b12 + a32 * b13 , != c12
a13 * b11 + a23 * b12 + a33 * b13 , != c13
a14 * b11 + a24 * b12 + a34 * b13 + b14 , != c14
a11 * b21 + a21 * b22 + a31 * b23 , != c21
a12 * b21 + a22 * b22 + a32 * b23 , != c22
a13 * b21 + a23 * b22 + a33 * b23 , != c23
a14 * b21 + a24 * b22 + a34 * b23 + b24 , != c24
a11 * b31 + a21 * b32 + a31 * b33 , != c31
a12 * b31 + a22 * b32 + a32 * b33 , != c32
a13 * b31 + a23 * b32 + a33 * b33 , != c33
a14 * b31 + a24 * b32 + a34 * b33 + b34 != c34
```

### Copyright

Ohne Garantiefür Vollständigkeit und Richtigkeit.  
Hamburg, März 2005, Frank Beister

### Links

<http://www.mathematik.net/matrizen/21.htm>  
<http://www.mathematik.net/vektoral/vak8s8.htm>  
[http://www.matheboard.de/matrizen\\_multiplikation.php](http://www.matheboard.de/matrizen_multiplikation.php)

[Thanks to Karl Ottenstein, who enlightened the members of the GDL-Alliance in July 2001 in this issue.]