

1 Szenario

Gegeben sind 2 Koordinatensysteme $V(\vec{v}^1, \vec{v}^2, \vec{v}^3)$ und $W(\vec{w}^1, \vec{w}^2, \vec{w}^3)$. Ziel ist es nun einen Vektor \vec{v} aus V nach W zu transformieren. Aus vorherigem Tutorial ist bereits bekannt, dass dazu zuerst die Matrix M bestimmt werden muss, die Translation \vec{p} und Rotation R zwischen beiden Koordinatensystemen enthält. Nun kann der Vektor \vec{w} wie folgt bestimmt werden:

$$\vec{w} = M \cdot \vec{v}$$

Wobei zu berücksichtigen ist, dass M eine 4x4 Matrix darstellt, daher müssen die Vektoren \vec{v} und \vec{w} auch 4-Dimensional sein.

2 Implementierung

Wie oben beschrieben kann dies nun in den Code einfließen:

```
Vector3D v1(1,0,0);
Vector3D v2(0,1,0);
Vector3D v3(0,0,1);
Vector3D w1(1,2,3);
Vector3D w2(0,0,4);
Vector3D ov(0,0,0);
Vector3D ow(1,1,1);

CoordinateFrame3D V(v1,v2,v3,ov); //Erstelle V
CoordinateFrame3D W(w1,w2,ow);    //Erstelle W
Matrix4D M = V.mapTo(W);          //M enthält p und R

Vector3D v(1337,1,1);
Vector4D v4D = v.to4D();           // 4 dimensional
Vector4D w4D = v4D.matVecMult(M); //w4d = M*v4D
Vector3D w = w4D.to3D();           // ergebnis
```

3 Hinweis

Sind nicht beide Koordinatensysteme bekannt, sondern nur Vektoren im jeweiligen Koordinatensystem, so verweise ich hier auf das TRIAD-Tutorial