- Definition of TGeoHMatrix

In[2]:=
```
RM[phi_, tht_, psi_, x_, y_, z_] := {
   {Cos[psi] Cos[phi] - Cos[tht] Sin[phi] Sin[psi],
    -Sin[psi] Cos[phi] - Cos[tht] Sin[phi] Cos[psi],
    Sin[tht] Sin[phi],
    0},
   {Cos[psi] Sin[phi] + Cos[tht] Cos[phi] Sin[psi],
    -Sin[psi] Sin[phi] + Cos[tht] Cos[phi] Cos[psi],
    -Sin[tht] Cos[phi],
    0},
   {Sin[psi] Sin[tht],
    Cos[psi] Sin[phi],
    Cos[tht],
    0},
   {x, y, z, 1}
  };
```

- Definition of TGeoHMatrix multiplication: ML * MR

In[3]:=
```
MPRD[ML_, MR_] := {
   {ML[[1, 1]] * MR[[1, 1]] + ML[[1, 2]] * MR[[2, 1]] + ML[[1, 3]] * MR[[3, 1]],
    ML[[1, 1]] * MR[[1, 2]] + ML[[1, 2]] * MR[[2, 2]] + ML[[1, 3]] * MR[[3, 2]],
    ML[[1, 1]] * MR[[1, 3]] + ML[[1, 2]] * MR[[2, 3]] + ML[[1, 3]] * MR[[3, 3]], 0},
   {ML[[2, 1]] * MR[[1, 1]] + ML[[2, 2]] * MR[[2, 1]] + ML[[2, 3]] * MR[[3, 1]],
    ML[[2, 1]] * MR[[1, 2]] + ML[[2, 2]] * MR[[2, 2]] + ML[[2, 3]] * MR[[3, 2]],
    ML[[2, 1]] * MR[[1, 3]] + ML[[2, 2]] * MR[[2, 3]] + ML[[2, 3]] * MR[[3, 3]], 0},
   {ML[[3, 1]] * MR[[1, 1]] + ML[[3, 2]] * MR[[2, 1]] + ML[[3, 3]] * MR[[3, 1]],
    ML[[3, 1]] * MR[[1, 2]] + ML[[3, 2]] * MR[[2, 2]] + ML[[3, 3]] * MR[[3, 2]],
    ML[[3, 1]] * MR[[1, 3]] + ML[[3, 2]] * MR[[2, 3]] + ML[[3, 3]] * MR[[3, 3]], 0},
   {ML[[4, 1]] + ML[[1, 1]] * MR[[4, 1]] + ML[[1, 2]] * MR[[4, 2]] + ML[[1, 3]] * MR[[4, 3]],
    ML[[4, 2]] + ML[[2, 1]] * MR[[4, 1]] + ML[[2, 2]] * MR[[4, 2]] + ML[[2, 3]] * MR[[4, 3]],
    ML[[4, 3]] + ML[[3, 1]] * MR[[4, 1]] +
     ML[[3, 2]] * MR[[4, 2]] + ML[[3, 3]] * MR[[4, 3]], 1}
  };
```

- Unity matrix

In[4]:=
```
U = {{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}};
```

- Transformation TGeoHMatrix matrices, usually we use RI = Inverse[R]

In[5]:=
```
R = {{r0, r1, r2, 0}, {r3, r4, r5, 0}, {r6, r7, r8, 0}, {t0, t1, t2, 1}};
RI = {{ri0, ri1, ri2, 0}, {ri3, ri4, ri5, 0}, {ri6, ri7, ri8, 0}, {ti0, ti1, ti2, 1}};
```

- Input parameters defining delta TGeoHMatrix : alignment increment. We assume that the parameters are so small
  that cos(x)->1 and sin(x)->x approximation is valid

In[7]:=
```
tau = RM[dphi, dtht, dpsi, dtx, dty, dtz];
rule1 = {Cos[dphi] → 1, Cos[dpsi] → 1, Cos[dtht] → 1,
   Sin[dphi] → dphi, Sin[dpsi] → dpsi, Sin[dtht] → dtht};
rule2 = {dphi * dpsi → 0, dpsi * dtht → 0, dphi * dtht → 0};
tauS = tau /. rule1 /. rule2;
```

- We need to compute transformation of delta matrix (tau) from its frame to another frame (vectors trasform as V = R*v) and take its
  component linear in tau input paramets. The final aim is to have the sum of transformations of child volumes to be unity matrix in
  their parent's frame, i.e. $\Sigma R_i \tau$ $\mathrm{RInv}_i = \mathbf{I}$, hence we can require $\Sigma R_i(\tau - \mathbf{I})$ $\mathrm{RInv}_i = 0$.

```
In[11]:= tauSU = tauS - U;

In[12]:= TAUU = MPRD[R, MPRD[tauSU, RI]]

Out[12]= {{(-dphi - dpsi) r0 ri3 + dphi r2 ri3 + r1 ((dphi + dpsi) ri0 - dtht ri6),
   (-dphi - dpsi) r0 ri4 + dphi r2 ri4 + r1 ((dphi + dpsi) ri1 - dtht ri7),
   (-dphi - dpsi) r0 ri5 + dphi r2 ri5 + r1 ((dphi + dpsi) ri2 - dtht ri8), 0},
  {(-dphi - dpsi) r3 ri3 + dphi r5 ri3 + r4 ((dphi + dpsi) ri0 - dtht ri6),
   (-dphi - dpsi) r3 ri4 + dphi r5 ri4 + r4 ((dphi + dpsi) ri1 - dtht ri7),
   (-dphi - dpsi) r3 ri5 + dphi r5 ri5 + r4 ((dphi + dpsi) ri2 - dtht ri8), 0},
  {(-dphi - dpsi) r6 ri3 + dphi r8 ri3 + r7 ((dphi + dpsi) ri0 - dtht ri6),
   (-dphi - dpsi) r6 ri4 + dphi r8 ri4 + r7 ((dphi + dpsi) ri1 - dtht ri7),
   (-dphi - dpsi) r6 ri5 + dphi r8 ri5 + r7 ((dphi + dpsi) ri2 - dtht ri8), 0},
  {t0 + r2 (dtz + dphi ti1) + r0 (dtx + (-dphi - dpsi) ti1) +
    r1 (dty + (dphi + dpsi) ti0 - dtht ti2), t1 + r5 (dtz + dphi ti1) +
    r3 (dtx + (-dphi - dpsi) ti1) + r4 (dty + (dphi + dpsi) ti0 - dtht ti2),
   t2 + r8 (dtz + dphi ti1) + r6 (dtx + (-dphi - dpsi) ti1) +
    r7 (dty + (dphi + dpsi) ti0 - dtht ti2), 1}}

In[13]:= MatrixForm[TAUU]

Out[13]//MatrixForm=
            (-dphi - dpsi) r0 ri3 + dphi r2 ri3 + r1 ((dphi + dpsi) ri0 - dtht ri6)
            (-dphi - dpsi) r3 ri3 + dphi r5 ri3 + r4 ((dphi + dpsi) ri0 - dtht ri6)
            (-dphi - dpsi) r6 ri3 + dphi r8 ri3 + r7 ((dphi + dpsi) ri0 - dtht ri6)
 t0 + r2 (dtz + dphi ti1) + r0 (dtx + (-dphi - dpsi) ti1) + r1 (dty + (dphi + dpsi) ti0 - dtht ti2

In[18]:= MatrixForm[D[TAUU, dphi]]

Out[18]//MatrixForm=
 r1 ri0 - r0 ri3 + r2 ri3  r1 ri1 - r0 ri4 + r2 ri4  r1 ri2 - r0 ri5 + r2 ri5  0
 r4 ri0 - r3 ri3 + r5 ri3  r4 ri1 - r3 ri4 + r5 ri4  r4 ri2 - r3 ri5 + r5 ri5  0
 r7 ri0 - r6 ri3 + r8 ri3  r7 ri1 - r6 ri4 + r8 ri4  r7 ri2 - r6 ri5 + r8 ri5  0
 r1 ti0 - r0 ti1 + r2 ti1  r4 ti0 - r3 ti1 + r5 ti1  r7 ti0 - r6 ti1 + r8 ti1  0

In[19]:= MatrixForm[D[TAUU, dpsi]]

Out[19]//MatrixForm=
 r1 ri0 - r0 ri3  r1 ri1 - r0 ri4  r1 ri2 - r0 ri5  0
 r4 ri0 - r3 ri3  r4 ri1 - r3 ri4  r4 ri2 - r3 ri5  0
 r7 ri0 - r6 ri3  r7 ri1 - r6 ri4  r7 ri2 - r6 ri5  0
 r1 ti0 - r0 ti1  r4 ti0 - r3 ti1  r7 ti0 - r6 ti1  0

In[20]:= MatrixForm[D[TAUU, dtht]]

Out[20]//MatrixForm=
 -r1 ri6  -r1 ri7  -r1 ri8  0
 -r4 ri6  -r4 ri7  -r4 ri8  0
 -r7 ri6  -r7 ri7  -r7 ri8  0
 -r1 ti2  -r4 ti2  -r7 ti2  0
```

In[21]:= `MatrixForm[D[TAUU, dtx]]`

Out[21]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ r0 & r3 & r6 & 0 \end{pmatrix}$$

In[22]:= `MatrixForm[D[TAUU, dty]]`

Out[22]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ r1 & r4 & r7 & 0 \end{pmatrix}$$

In[23]:= `MatrixForm[D[TAUU, dtz]]`

Out[23]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ r2 & r5 & r8 & 0 \end{pmatrix}$$