```
In[1]:=  << GA30`;

In[98]:= ClearAll[i, j, k, b1, b2, b3, ej, x, xr, xt, xp, r]
         i = Bivector[1, 1, 2];
         k = Bivector[1, 3, 1];
         b1 = Vector[1, 1];
         b2 = Vector[1, 2];
         b3 = Vector[1, 3];
         j[phi_] = k ** (Cos[phi] + i Sin[phi]);
         ej[t_, p_] = Cos[t] + j[p] Sin[t];
         x[r_, t_, p_] = r (b3 ** ej[t, p]);

         xr[r_, theta_, phi_] = D[x[a, theta, phi], a] /. a → r;
         xt[r_, theta_, phi_] = D[x[r, t, phi], t] /. t → theta;
         xp[r_, theta_, phi_] = D[x[r, theta, p], p] /. p → phi;

         {x[r, θ, ϕ],
           xr[r, θ, ϕ],
           xt[r, θ, ϕ],
           xp[r, θ, ϕ]} // Column
```

```
         r Cos[ϕ] Sin[θ] e1 + r Sin[θ] Sin[ϕ] e2 + r Cos[θ] e3
         Cos[ϕ] Sin[θ] e1 + Sin[θ] Sin[ϕ] e2 + Cos[θ] e3
Out[110]=
         r Cos[θ] Cos[ϕ] e1 + r Cos[θ] Sin[ϕ] e2 - r Sin[θ] e3
         -r Sin[θ] Sin[ϕ] e1 + r Cos[ϕ] Sin[θ] e2
```

```
          xr[r, θ, ϕ] **
           xt[r, θ, ϕ] **
           xp[r, θ, ϕ]

         {e1, e2, e3} = IdentityMatrix[3];

         (*probably a better way to do this with Map or MapThread*)
         jacobean = {
            {xr[r, θ, ϕ] . b1, xr[r, θ, ϕ] . b2, xr[r, θ, ϕ] . b3},
            {xt[r, θ, ϕ] . b1, xt[r, θ, ϕ] . b2, xt[r, θ, ϕ] . b3},
            {xp[r, θ, ϕ] . b1, xp[r, θ, ϕ] . b2, xp[r, θ, ϕ] . b3}};

         Det[ jacobean ] // Simplify
```

```
Out[111]= r^2 Sin[θ] e123

Out[114]= r^2 Sin[θ]
```

```
In[125]:= ClearAll[a, b, c, e, f, g]
          Map[foo, {{a, b, c}, {e, f, g}}, 1]

          ClearAll: Symbol e is Protected.

Out[126]= {foo[{a, b, c}], foo[{e, f, g}]}
```