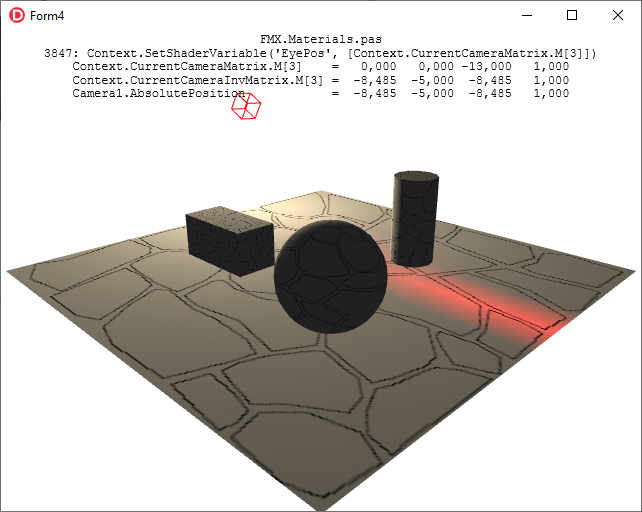
I see that FMX TLightMaterial **specular**is not correct, as report here since 2015:

<https://quality.embarcadero.com/browse/RSP-10000>

But until now (10.4.2) it has not been fixed. As screenshot below:

​

I think that the Camera (eye) Position is not set correct, so I look at unit ***FMX.Materials.pas*** and found this line:

procedure TLightMaterial.DoApply(const Context: TContext3D);

​...

​​​3847: Context.SetShaderVariable('EyePos', [Context.CurrentCameraMatrix.M[3]]);

But in this context, **CurrentCameraMatrix​.M[3]** is not **absolute position** of current camera, as code below:

Unit FMX.Controls3D; Line 2681:

​​​function TCamera.GetCameraMatrix: TMatrix3D;

begin

if FTarget <> nil then

Result := TMatrix3D.CreateLookAtDirRH(**TPoint3D(AbsolutePosition)**, TPoint3D(AbsolutePosition) -

TPoint3D(Target.AbsolutePosition), - TPoint3D(AbsoluteUp))

else

Result := TMatrix3D.CreateLookAtDirRH(**TPoint3D(AbsolutePosition)**, - TPoint3D(AbsoluteDirection),

- TPoint3D(AbsoluteUp));

end;

Unit System.Math.Vectors; Line 1295:

class function TMatrix3D.CreateLookAtDirRH(const **ASource**, ADirection, ACeiling: TPoint3D): TMatrix3D;

var

ZAxis, XAxis, YAxis: TPoint3D;

begin

ZAxis := ADirection.Normalize;

XAxis := ACeiling.CrossProduct(ZAxis).Normalize;

YAxis := ZAxis.CrossProduct(XAxis);

...

Result.m41 := - XAxis.DotProduct(**ASource**);

Result.m42 := - YAxis.DotProduct(**ASource**);

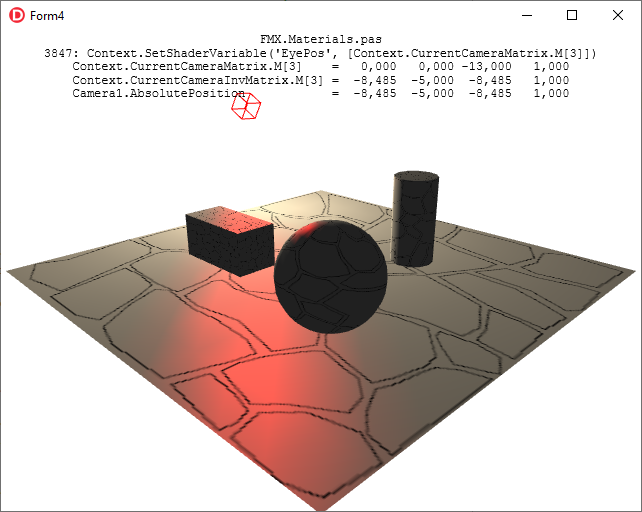
Result.m43 := - ZAxis.DotProduct(**ASource**);

end;

​​So I copy file *FMX.Materials.pas*to current Project folder, modify line 3847 as below:

3847: Context.SetShaderVariable('EyePos', [Context.CurrentCamera**Inv**Matrix.M[3]]);

Rebuild the project, and result look good:

​

My test code at this link: <https://github.com/thaivankhanh/Delphi>

I hope the problem will be resolved.

Thanks.

Why Context.**CurrentCameraInvMatrix**.M[3] is Camera Absolute Position (EyePos)?

We known:  
CameraAbsolutePosition = Vector(0,0,0,1) \* CurrentCameraInvMatrix

class operator TMatrix3D.Multiply(const AVector: TVector3D;  
const AMatrix: TMatrix3D): TVector3D;  
begin  
Result.X := (AVector.X \* AMatrix.m11) + (AVector.Y \* AMatrix.m21) + (AVector.Z \* AMatrix.m31) + (AVector.W \* AMatrix.m41);  
Result.Y := (AVector.X \* AMatrix.m12) + (AVector.Y \* AMatrix.m22) + (AVector.Z \* AMatrix.m32) + (AVector.W \* AMatrix.m42);  
Result.Z := (AVector.X \* AMatrix.m13) + (AVector.Y \* AMatrix.m23) + (AVector.Z \* AMatrix.m33) + (AVector.W \* AMatrix.m43);  
Result.W := (AVector.X \* AMatrix.m14) + (AVector.Y \* AMatrix.m24) + (AVector.Z \* AMatrix.m34) + (AVector.W \* AMatrix.m44);  
end;

So when AVector is (0,0,0,1), Result is (AMatrix.m41, AMatrix.m42, AMatrix.m43, AMatrix.m44), that is AMatrix.M[3]

CameraAbsolutePosition = Vector(0,0,0,1) \* CurrentCameraInvMatrix = CurrentCameraInvMatrix.M[3]