

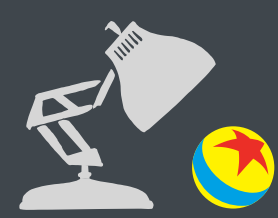


---

# MaterialBindingAPI Requirements and Updates

---

Varun Talwar  
[varun@pixar.com](mailto:varun@pixar.com)



# MaterialBindingAPI Evolution

- Applied API schema to begin with, but not strictly enforced in core usdShade code.
- Realization of performance benefits and Hydra2.0 requirements

```
1 #usda 1.0
2 (
3   defaultPrim = "Mesh"
4   endFrame = 10
5   metersPerUnit = 1
6   startFrame = 1
7   upAxis = "Y"
8 )
9 def Material "Material"
10 {
11   token outputs:surface.connect = </Material/Surface.outputs:surface>
12
13   def Shader "Surface"
14   {
15     color3f inputs:diffuseColor.connect = </Material/DiffuseColor.outputs:rgb>
16     token outputs:surface
17   }
18
19   def Shader "DiffuseColor"
20   {
21     float3 outputs:rgb
22   }
23 }
24
25 def "Mesh"
26 {
27   rel material:binding = </Material>
28 }
29
```

```
1 #usda 1.0
2 (
3   defaultPrim = "Mesh"
4   endFrame = 10
5   metersPerUnit = 1
6   startFrame = 1
7   upAxis = "Y"
8 )
9 def Material "Material"
10 {
11   token outputs:surface.connect = </Material/Surface.outputs:surface>
12
13   def Shader "Surface"
14   {
15     color3f inputs:diffuseColor.connect = </Material/DiffuseColor.outputs:rgb>
16     token outputs:surface
17   }
18
19   def Shader "DiffuseColor"
20   {
21     float3 outputs:rgb
22   }
23 }
24
25 def "Mesh" (
26   prepend apiSchemas = ["MaterialBindingAPI"]
27 )
28 {
29   rel material:binding = </Material>
30 }
31
```



# Plan of Action

- Modify `UsdShadeMaterialBindingAPI::ComputeBoundMaterial(s)` to include a “supportLegacyBinding” parameter.

```
pxr/usd/usdShade/materialBindingAPI.h

public:

781  /// UsdShadeMaterialBindingAPI::GetResolvedTargetPathFromBindingRel to get
782  /// the path of the resolved target identified by the winning bindingRel.
783  ///

784  /// See \ref UsdShadeMaterialBindingAPI_MaterialResolution "Bound Material Resol-
785  /// for details on the material resolution process.
786  ///
787  /// The python version of this method returns a tuple containing the
788  /// bound material and the "winning" binding relationship.
789  USDSHADE_API
790  UsdShadeMaterial ComputeBoundMaterial(
791      BindingsCache *bindingsCache,
792      CollectionQueryCache *collectionQueryCache,
793      const TfToken &materialPurpose=UsdShadeTokens->allPurpose,
794      UsdRelationship *bindingRel=nullptr) const;

781  /// UsdShadeMaterialBindingAPI::GetResolvedTargetPathFromBindingRel to get
782  /// the path of the resolved target identified by the winning bindingRel.
783  ///
784  /// In order for backward compatibility with old assets not having
785  /// MaterialBindingAPI applied, \p supportLegacyBindings defaults to true.
786  /// Though its recommended for clients to update the assets to have
787  /// MaterialBindingAPI applied for optimized computation of bound material.
788  ///
789  /// Note: In a future release the default for \p supportLegacyBindings will
790  /// be updated to "false".
791  ///
792  /// See \ref UsdShadeMaterialBindingAPI_MaterialResolution "Bound Material Resol-
793  /// for details on the material resolution process.
794  ///
795  /// The python version of this method returns a tuple containing the
796  /// bound material and the "winning" binding relationship.
797  USDSHADE_API
798  UsdShadeMaterial ComputeBoundMaterial(
799      BindingsCache *bindingsCache,
800      CollectionQueryCache *collectionQueryCache,
801      const TfToken &materialPurpose=UsdShadeTokens->allPurpose,
802      UsdRelationship *bindingRel=nullptr,
803      bool supportLegacyBindings=true) const;
```



# Plan of Action

- Update UsdImaging 1.0 to always allow legacy behavior.

pxr/usdImaging/usdImaging/resolvedAttributeCache.h

```
typedef UsdImaging_ResolvedAttributeCache<UsdImaging_MaterialStrategy,
```

```
699     UsdImaging_MaterialBindingImplData>  
700     UsdImaging_MaterialBindingCache;  
701
```

```
702 struct UsdImaging_MaterialStrategy {  
703     // inherited path to bound target  
704     // depending on the load state, override, etc bound target path might not →
```

```
struct UsdImaging_MaterialStrategy {
```

```
721     query_type MakeQuery(  
722         UsdPrim const& prim,  
723         ImplData *implData)
```

```
724     {  
725         UsdRelationship bindingRel;  
726         UsdShadeMaterial materialPrim =  
727             UsdShadeMaterialBindingAPI(prim).ComputeBoundMaterial(  
728                 &implData->GetBindingsCache(),  
729                 &implData->GetCollectionQueryCache(),  
730                 implData->GetMaterialPurpose(),  
731                 &bindingRel);
```

```
732  
733         if (materialPrim) {  
734             return materialPrim.GetPath();
```

```
699     UsdImaging_MaterialBindingImplData>  
700     UsdImaging_MaterialBindingCache;
```

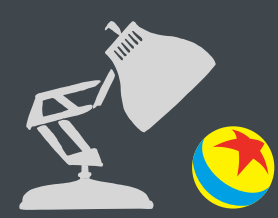
```
702 struct UsdImaging_MaterialStrategy {  
703     // inherited path to bound target  
704     // depending on the load state, override, etc bound target path might not →
```

```
722     query_type MakeQuery(  
723         UsdPrim const& prim,  
724         ImplData *implData)
```

```
725     {  
726         UsdRelationship bindingRel;  
727         UsdShadeMaterial materialPrim =  
728             UsdShadeMaterialBindingAPI(prim).ComputeBoundMaterial(  
729                 &implData->GetBindingsCache(),  
730                 &implData->GetCollectionQueryCache(),  
731                 implData->GetMaterialPurpose(),  
732                 &bindingRel,  
733                 true /*supportLegacyBindings*/);
```

```
734  
735         if (materialPrim) {  
736             return materialPrim.GetPath();
```

- Updates to Hydra2.0 to always enforce MaterialBindingAPI is applied.



# Tools - Validation and Fixing

- usdchecker: MaterialBindingAPI rule
- usdfixbrokenpixarschemas: Applied fixes to a single USD layer or an entire usdz package

```
islay <UNIT> <PROD> /home/varun/dev/materialBindingAPI-aswf-usd-wg :  
> delta badMaterialBindingAPIApplied.usda cleanMaterialBindingAPIApplied.usda
```

**comparing: badMaterialBindingAPIApplied.usda → cleanMaterialBindingAPIApplied.usda**

```
23     }  
24 }  
25  
26 def "Mesh"  
  
27 {  
28     rel material:binding = </Material>  
29 }
```

```
23     }  
24 }  
25  
26 def "Mesh" (  
27     prepend apiSchemas = ["MaterialBindingAPI"]  
28 )  
29 {  
30     rel material:binding = </Material>  
31 }
```

```
islay <UNIT> <PROD> /home/varun/dev/materialBindingAPI-aswf-usd-wg :
```

```
> usdchecker badMaterialBindingAPIApplied.usda
```

```
Found material bindings but no MaterialBindingAPI applied on the prim </Mesh>. (fails 'MaterialBindingAPIAppliedChecker')
```

```
Failed!
```

```
islay <UNIT> <PROD> /home/varun/dev/materialBindingAPI-aswf-usd-wg :
```

```
> usdfixbrokenpixarschemas badMaterialBindingAPIApplied.usda
```

```
Warning: in _GetUnitNameOrFallback at line 160 of bedrock/prod/pr/pathResolverUtils.cpp -- Empty unit provided - using fallback
```

```
islay <UNIT> <PROD> /home/varun/dev/materialBindingAPI-aswf-usd-wg :
```

```
> usdchecker badMaterialBindingAPIApplied.usda
```

```
Success!
```

```
islay <UNIT> <PROD> /home/varun/dev/materialBindingAPI-aswf-usd-wg :
```

```
> delta badMaterialBindingAPIApplied.usda cleanMaterialBindingAPIApplied.usda
```

```
islay <UNIT> <PROD> /home/varun/dev/materialBindingAPI-aswf-usd-wg :
```