

Asset Dependencies

The following scripting define symbols are used to add ***paid*** or ***non-distributable*** asset add-on features to MRET. These symbols are automatically added to the project settings in the editor upon detection of the script assets and are managed using Unity's ConditionalCompilationUtility (CCU). Please download and configure the CCU prior to opening MRET in Unity for the first time. Refer the installation instructions in the Release Guide distributed with MRET for more information. The CCU is located here:

<https://github.com/Unity-Technologies/ConditionalCompilationUtility/tree/f364090bbda3728e1662074c969c2b7c3c34199b>

The MRET assembly definition is defined here:

```
Assets/MRET/Core/Editor/MRETExtensionSymbols.cs
```

The add-on dependency feature set is broken into several groups: automatic, recommended, and optional. They are defined as follows:

AUTOMATIC

- SicCity GLTFUtility
 - URL:** <https://github.com/Siccity/GLTFUtility>
 - Symbol:** MRET_EXTENSION_SICCITYGLTF
 - Description:** Used for loading GLTF files at MRET runtime
 - Scoped Registry:**

```
"scopedRegistries": [  
  {  
    "name": "Siccity",  
    "url": "https://github.com/siccity/glutfutility.git",  
    "scopes": [  
      "com.siccity.glutfutility"  
    ]  
  },  
]
```
- Newtonsoft JSON
 - URL:** <https://www.newtonsoft.com/json>
 - Symbol:** MRET_EXTENSION_NEWTONSOFTJSON
 - Description:** Used for JSON serialization. This is primarily used for storing and retrieving client and server certs and for parsing IoT telemetry.
- Virtual Data Explorer (VDE)
 - o **URL:** <https://coda.ee/vde>
 - o **Symbol:** MRET_EXTENSION_VDE

- **Description:** Visualize and explore your computer network topology as a set of 3D data-shapes in MRET. Included as part of the standard MRET distribution.

RECOMMENDED

- Digital Ruby Free Earth Planet

URL: <https://assetstore.unity.com/packages/vfx/shaders/free-earth-planet-the-best-planet-shader-in-the-asset-store-56841>

Symbol: MRET_EXTENSION_DIGITALRUBYEARTH

Description: Scripts support low tessellation sphere primitives and planets and provides textures/materials/shaders for a low-resolution Earth. If not added to MRET, predefined Earth and other prefabs that use low tessellation sphere meshes will not be available and errors may ensue for some MRET projects.

- Dummiesman Runtime OBJ Importer

URL: <https://assetstore.unity.com/packages/tools/modeling/runtime-obj-importer-49547#content>

Symbol: MRET_EXTENSION_OBJIMPORTER

Description: Used for loading the SteamVR controller model in the Cross-Platform Input System (CPIS)

Notes: This software has not been maintained but it still works with Unity. It does not work as is with the HDRP nor URP render pipelines. The following edits are required:

MTLLoader.cs

Add this method:

```
/// <summary>
/// Obtains the default material depending on the defined render pipeline
/// </summary>
/// <returns>The <code>Material</code> reference or NULL if not defined</returns>
protected Material GetRenderPipelineMaterial()
{
    Material result = null;

    if (QualitySettings.renderPipeline != null)
    {
        result = QualitySettings.renderPipeline.defaultMaterial;
    }
    else if (GraphicsSettings.defaultRenderPipeline != null)
    {
        result = GraphicsSettings.defaultRenderPipeline.defaultMaterial;
    }

    return result;
}
```

MTLLoader:

Modify this method:

```
public Dictionary<string, Material> Load(Stream input)
[
.....
    //newmtl
    if (splitLine[0] == "newmtl")
    {
        string materialName = processedLine.Substring(7);

        Material newMtl = GetRenderPipelineMaterial();
        try
        {
            newMtl = new Material(newMtl.shader) { name = materialName };
        }
        catch (System.Exception)
        {
            newMtl = null;
        }

        if (newMtl != null)
        {
            mtlDict[materialName] = newMtl;
            currentMaterial = newMtl;
        }

        continue;
    }
    .....
    //diffuse
    var KdTexture = TryLoadTexture(texturePath);
    currentMaterial.SetTexture("BaseMap", KdTexture);
    .....
    //specular map
    if (splitLine[0] == "map_Ks" || splitLine[0] == "map_ks")
    {
        string texturePath = GetTexPathFromMapStatement(processedLine, splitLine);
        if (texturePath == null)
        {
            continue; //invalid args or sth
        }

        currentMaterial.SetTexture("_SpecGlossMap", TryLoadTexture(texturePath));
        currentMaterial.SetFloat("_WorkflowMode", 0);
        continue;
    }
}
```

- Microsoft Mixed Reality Toolkit (MRTK) Foundation 2.8.3
URL: <https://github.com/microsoft/MixedRealityToolkit-Unity/releases>
Symbol: MRET_EXTENSION_MRTK
Description: A set of foundational components and features to accelerate mixed reality app development in Unity. Not currently used for the Windows standalone version of MRET, but required for the HoloLens 2/UWP
- Microsoft Mixed Reality OpenXR 1.8.1
URL: <https://github.com/microsoft/MixedRealityToolkit-Unity/releases>
Symbol: MRET_EXTENSION_MROPNXR
Description: Extension to Unity OpenXR Plugin to support a suite of features for HoloLens 2 and Windows Mixed Reality headsets.

OPTIONAL

- BestHTTP
URL: <https://assetstore.unity.com/packages/tools/network/best-http-2-155981>
Symbol: MRET_EXTENSION_BESTHTTP
Description: Used only by Virtual Data Explorer (VDE) to support communication with the backend and for synchronization of user sessions.
- Chart and Graph
URL: <https://assetstore.unity.com/packages/tools/gui/graph-and-chart-78488>
Symbol: MRET_EXTENSION_CHARTANDGRAPH
Description: Supports plotting of data points stored in the MRET DataManager
- Easy Build System
URL: <https://assetstore.unity.com/packages/templates/systems/easy-build-system-45394>
Symbol: MRET_EXTENSION_EASYBUILDSYSTEM
Description: Supports optional alignment/snapping of scene objects in MRET to other scene objects
- FinalIK
URL: <https://assetstore.unity.com/packages/tools/animation/final-ik-14290>
Symbol: MRET_EXTENSION_FINALIK
Description: Adds low-fidelity IK to MRET, including avatars and robotic arms (See RosSharp)
- OSGeo GDAL
URL: <https://gdal.org>
Symbol: MRET_EXTENSION_GDAL
Description: Used for Geospatial processing, specifically converting DEMs to 3D terrains
Scoped Registry:

```
"scopedRegistries": [  
  {  
    "name": "package.openupm.com",  
    "url": "https://package.openupm.com",
```

- ```
 "scopes": [
 "com.openupm",
 "com.virgis"
],
 },
]
```
- M2MQTT  
**nuget>** Install-Package M2Mqtt  
**Symbol:** MRET\_EXTENSION\_M2MQTT  
**Description:** Adds in scene object association with NASA IoT sources into MRET
  - Microsoft .NET JSON  
**URL:** <https://www.nuget.org/packages/System.Text.Json/>  
**Symbol:** MRET\_EXTENSION\_SYSTEMTEXTJSON  
**Description:** Used for JSON serialization. If present in the project, this JSON serialization will be used in place of the Newtonsoft JSON because of its high-performance and low memory allocation. It is not available as a Unity asset/plugin but is possible to get it to work with Unity. It is included as part of the Siemens RosSharp distribution.
  - NonConvexMeshCollider  
**URL:** <https://assetstore.unity.com/packages/tools/physics/non-convex-mesh-collider-84867>  
**Symbol:** MRET\_EXTENSION\_NONCONVEXMESHCOLLIDER  
**Description:** Supports optional non-convex colliders for scene object in MRET
  - MGear Point Cloud Viewer and Tools  
**URL:** <https://assetstore.unity.com/packages/tools/utilities/point-cloud-viewer-and-tools-16019#content>  
**Symbol:** MRET\_EXTENSION\_POINTCLOUDVIEWER  
**Description:** Currently used to load static point clouds in MRET from a file.
  - RockVR Video Capture  
**URL:** <https://assetstore.unity.com/packages/tools/video/video-capture-75653>  
**Symbol:** MRET\_EXTENSION\_ROCKVR  
**Description:** Supports optional interactive video capture recording during runtime of MRET
  - Siemens RosSharp  
**URL:** <https://github.com/siemens/ros-sharp/tree/master/ROS>  
**Symbol:** MRET\_EXTENSION\_ROSSHARP  
**Description:** Provides low-fidelity IK solutions for robotic arms in MRET using FinalIK
  - TerrainTools  
**Unity Preview Package:**  
<https://docs.unity3d.com/2020.3/Documentation/Manual/terrain-Tools.html>  
**Symbol:** MRET\_EXTENSION\_TERRAINTOOLS  
**Description:** Supports dynamic terrain creation
  - Vuplex 3D Webview  
**URL:** <https://developer.vuplex.com/webview/overview>

**Symbol:** MRET\_EXTENSION\_VUPLEX

**Description:** Add in a 2D/3D HTML browser into MRET

DEPRECATED

- UnityOctree

**URL:** <https://github.com/Nition/UnityOctree.git>

**Symbol:** MRET\_EXTENSION\_UNITYOCTREE

**Description:** Used during investigation of octrees for a point cloud implementation. Not currently used.